Hitachi Single-Chip Microcomputer

H8/3857 Series

H8/3857	HD6433857, HCD6433857
H8/3856	HD6433856, HCD6433856
H8/3855	HD6433855, HCD6433855
H8/3857F-ZTAT™	HD64F3857, HCD64F3857

H8/3854 Series

H8/3854	HD6433854, HCD6433854
H8/3853	HD6433853, HCD6433853
H8/3852	HD6433852, HCD6433852
H8/3854F-ZTAT™	HD64F3854, HCD64F3854

Hardware Manual

HITACHI

ADE-602-175A
Rev. 2.0
2/15/00
Hitachi, Ltd.

Hitachi semiconductor

Cautions

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

Preface

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

The H8/3857 Series has the following on-chip peripheral functions required for system configuration: a maximum 1,280-dot display LCD controller, four types of timers, a 14-bit PWM, a 2-channel serial communication interface, and an 8-channel A/D converter.

The H8/3854 Series has the following on-chip peripheral functions required for system configuration: a maximum 640-dot display LCD controller, three types of timers, a single-channel serial communication interface, and a 4-channel A/D converter.

Both series can be used as embedded microcomputers in systems requiring LCD display.

The H8/3857, H8/3856, H8/3855, H8/3854, H8/3853, and H8/3852 are available in mask ROM versions, and the H8/3857 and H8/3854 are also available in an F-ZTAT™* version which allows programs to be written after the chip is mounted on a board.

Note: * F-ZTAT (Flexible Zero Turn-Around Time) is a trademark of Hitachi, Ltd.

This manual describes the hardware of the H8/3857 Series and H8/3854 Series. For details on the H8/3857 Series and H8/3854 Series instruction set, refer to the H8/300L Series Programming Manual.

HITACHI

List of Functions

Series		H8/3857 Series				H8/3854 Series			
Product Code		**F-ZTAT Version**	**Mask ROM Version**			**F-ZTAT Version**	**Mask ROM Version**		
		H8/3857F	**H8/3857**	**H8/3856**	**H8/3855**	**H8/3854F**	**H8/3854**	**H8/3853**	**H8/3852**
ROM size (kbytes)		60	60	48	40	60*	32*	24	16
RAM size (kbytes)		2	2	2	2	2*	1*	1*	1*
I/O ports	Input/output ports	35	35	35	35	24	24	24	24
	Input ports	9	9	9	9	5	5	5	5
Interrupts	External interrupts	13 sources	13 sources	13 sources	13 sources	12 sources	12 sources	12 sources	12 sources
	Internal interrupts	16 sources	16 sources	16 sources	16 sources	14 sources	14 sources	14 sources	14 sources
Timer A (for realtime clock)		○	○	○	○	○	○	○	○
Timer B (8 bits)		○	○	○	○	○	○	○	○
Timer C (8 bits)		○	○	○	○	—	—	—	—
Timer F (16 bits)		○	○	○	○	○	○	○	○
Watchdog timer		○	—	—	—	○	—	—	—
14-bit PWM		○	○	○	○	—	—	—	—
Serial communication interface (SCI)		× 2	× 2	× 2	× 2	× 1	× 1	× 1	× 1
A/D converter		8 ch	8 ch	8 ch	8 ch	4 ch	4 ch	4 ch	4 ch
LCD controller	Max. display dots	1280 dots	1280 dots	1280 dots	1280 dots	640 dots	640 dots	640 dots	640 dots
	Display RAM size	2048 bits	2048 bits	2048 bits	2048 bits	640 bits	640 bits	640 bits	640 bits
Packages	Pins	144	144	144	144	100	100	100	100
	Shipping form	FP-144H (20 × 20 mm) TFP-144 (16 × 16 mm) Die (F-ZTAT version: 7.08 × 7.31 mm / mask ROM version: 6.21 × 6.21 mm)				FP-100B (14 × 14 mm) TFP-100G (12 × 12 mm) Die (F-ZTAT version: 6.34 × 6.34 mm / mask ROM version: 4.69 × 4.69 mm)			

Note: * Note that the H8/3854F (F-ZTAT version) and H8/3854 (mask ROM version) have different ROM and RAM sizes.

When carrying out program development using the H8/3854F with the intention of mask ROM implementation, care must be taken with ROM and RAM sizes since the maximum sizes for the mask ROM version are 32 kbytes of ROM and 1 kbyte of RAM.

List of Items Revised or Added for This Version

Page	Item	Description
All	H8/3854 Series added	
6	Figure 1.1 H8/3857 Series Internal Block Diagram	Changed
10	Figure 1.5 Pad Layout of HCD64F3857 (F-ZTAT Version)	Pad numbers changed
11, 12	Table 1.2 HCD64F3857 Pad Coordinates	Pad numbers changed
13	Figure 1.6 Pad Layout of HCD6433855, HCD6433856, and HCD6433857 (Mask ROM Version)	Model names amended
14, 15	Table 1.3 HCD6433855, HCD6433856, and HCD6433857 Pad Coordinates	Model names amended
18	Figure 1.8 Pad Layout of HCD6433854, HCD6433853, and HCD6433852 (Mask ROM Version)	Added
19	Table 1.5 HCD6433852, HCD6433853, and HCD6433854 Pad Coordinates	Added
37	Table 2.2 Effective Address Calculation	Amended
60	Figure 2.16 (a) H8/3857 Series Memory Map	Changed
96	4.2 System Clock Generator Inputting an External Clock	Added
109	5.3.5 Notes on External Input Signal Changes before/after Standby Mode	Added
111	5.4.4 Notes on External Input Signal Changes before/after Watch Mode	Added
118	5.8.3 Notes on External Input Signal Changes before/after Direct Transition	Added
153	Figure 6.16 (a) HD64F3857 Socket Adapter Pin Interconnections	“Notation” amended
168	6.10 Notes when Converting the F-ZTAT Application Software to the Mask-ROM Versions	Added
172	Table 8.1 (a) H8/3857 Series Port Functions	Port 3 pins changed
197	8.5.2 Register Configuration and Description Port Data Register 4	Changed
235	9.5.2 Register Descriptions Timer Control/ Status Register F (TCSRF)	Name of bit 3 amended
256	10.2.2 Register Descriptions Serial Control/Status Register 1	Description added
257	10.2.2 Register Descriptions Serial Control/Status Register 1	Bit 0 description changed

Page	Item	Description
261	10.2.5 Application Notes	Description changed
292	Table 10.5 Receive Error Conditions and Received Data Processing	Amended
347	13.3.2 CPU Interface Notes on Use of Chip-Internal I/O Ports 2	Description amended
350	Table 13.6 Pin Functions According to Display Mode and Display Duty	Title added
363	Figure 13.12 Module Standby Mode and Standby Mode Initiation and Clearing Procedures	Title corrected
405	Table 15.1 Absolute Maximum Ratings	Note amended
415	Table 15.5 Serial Interface (SCI1) Timing of H8/3855, H8/3856, and H8/3857	Amended
417	Table 15.7 A/D Converter Characteristics of H8/3855, H8/3856, and H8/3857	Note 1 amended
422	Figure 15.7 SCK3 Input Clock Timing	Title corrected
497	B.2 Register Descriptions PDRB-Port data register B	Changed
525	Figure C.6 Port 9 Block Diagram	Amended

Contents

Section 1 Overview

1.1 Overview

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

Within the H8/300L Series, the H8/3857 Series and H8/3854 Series feature on-chip liquid crystal display (LCD) controllers. Other on-chip peripheral functions include a LCD controller, timers, serial communication interface, and an analog-to-digital (A/D) converter. Together these functions make the H8/3857 Series and H8/3854 Series ideally suited for embedded control of systems requiring an LCD display.

The H8/3857 Series comprises the H8/3855, with 40 kbytes of ROM and 2 kbytes of RAM on-chip, the H8/3856, with 48 kbytes of ROM and 2 kbytes of RAM, and the H8/3857, with 60 kbytes of ROM and 2 kbytes of RAM. H8/3854 Series mask ROM versions are the H8/3852, with 16 kbytes of ROM and 1 kbyte of RAM on-chip, the H8/3853, with 24 kbytes of ROM and 1 kbyte of RAM, and the H8/3854, with 32 kbytes of ROM and 1 kbyte of RAM.

Two F-ZTAT versions—the H8/3857F and H8/3854F—are also available, with user-programmable on-chip flash ROM. These models have 60 kbytes of ROM and 2 kbytes of RAM.

Note that the H8/3854 mask ROM and F-ZTAT versions have different ROM and RAM sizes.

Table 1.1 summarizes the features of the H8/3857 Series and H8/3854 Series.

Table 1.1 Features

Item	Description
CPU	High-speed H8/300L CPU • General-register architecture General registers: Sixteen 8-bit registers (can be used as eight 16-bit registers) • Operating speed — Max. operating speed: 5 MHz — Add/subtract: 0.4 µs (operating at 5 MHz) — Multiply/divide: 2.8 µs (operating at 5 MHz) — Can run on 32.768 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 registers Typical instructions • Multiply (8 bits × 8 bits) • Divide (16 bits ÷ 8 bits) • Bit accumulator • Register-indirect designation of bit position
Interrupts	H8/3857 Series: 29 interrupt sources • 13 external interrupt sources: IRQ_4 to IRQ_0, WKP_7 to WKP_0 • 16 internal interrupt sources H8/3854 Series: 26 interrupt sources • 12 external interrupt sources: IRQ_4, IRQ_3, IRQ_1, IRQ_0, WKP_7 to WKP_0 • 14 internal interrupt sources
Clock pulse generators	Two on-chip clock pulse generators • System clock pulse generator: 1 to 10 MHz • Subclock pulse generator: 32.768 kHz
Power-down modes	Six power-down modes • Sleep mode • Standby mode • Watch mode • Subsleep mode • Subactive mode • Active (medium-speed) mode

Item	Description
Memory	H8/3857 Series • H8/3855: 40-kbyte ROM, 2-kbyte RAM • H8/3856: 48-kbyte ROM, 2-kbyte RAM • H8/3857: 60-kbyte ROM, 2-kbyte RAM • H8/3857F: 60-kbyte ROM, 2-kbyte RAM H8/3854 Series • H8/3852: 16-kbyte ROM, 1-kbyte RAM • H8/3853: 24-kbyte ROM, 1-kbyte RAM • H8/3854: 32-kbyte ROM, 1-kbyte RAM • H8/3854F: 60-kbyte ROM, 2-kbyte RAM Note that the H8/3854 (mask ROM version) and H8/3854F (F-ZTAT version) have different ROM and RAM sizes.
I/O ports	H8/3857 Series: 44 I/O port pins • I/O pins: 35 • Input pins: 9 H8/3854 Series: 29 I/O port pins • I/O pins: 24 • Input pins: 5
Timers	Four on-chip timers (three in the H8/3854 Series) • Timer A: 8-bit timer Count-up timer with selection of eight internal clock signals divided from the system clock (ϕ)*1 and four clock signals divided from the watch clock (ϕ_w)*1 • Timer B: 8-bit timer — Count-up timer with selection of seven internal clock signals or event input from external pin — Auto-reloading • Timer C: 8-bit timer (in H8/3857 Series only) — Count-up/count-down timer with selection of seven internal clock signals 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 signals or event input from external pin — Compare-match function with toggle output

Item	Description
Serial communication interface	H8/3857 Series: Two channels on chip • SCI1: synchronous serial interface Choice of 8-bit or 16-bit data transfer • SCI3: 8-bit synchronous or asynchronous serial interface Built-in function for multiprocessor communication H8/3854 Series: One channel on chip • SCI3: 8-bit synchronous or asynchronous serial interface Built-in function for multiprocessor communication
14-bit PWM (in H8/3857 Series only)	Pulse-division PWM output for reduced ripple • Can be used as a 14-bit D/A converter by connecting to an external low-pass filter.
A/D converter	H8/3857 Series • Successive approximations using a resistance ladder resolution: 8 bits • 8-channel analog input port • Conversion time: 31/ϕ or 62/ϕ per channel H8/3854 Series • Successive approximations using a resistance ladder resolution: 8 bits • 4-channel analog input port • Conversion time: 31/ϕ or 62/ϕ per channel
LCD controller	H8/3857 Series: Up to 64 segment pins and 32 common pins • Choice of three duty cycles (1/8, 1/16, 1/32) With 1/8 duty selected: 64 SEG $\times$ 8 COM, 40 SEG $\times$ 8 COM With 1/16 duty selected: 56 SEG $\times$ 16 COM, 40 SEG $\times$ 16 COM With 1/32 duty selected: 40 SEG $\times$ 32 COM • Built-in 2048-bit display data RAM • Built-in 2$\times$ or 3$\times$ LCD step-up circuit • Built-in contrast control circuit • Built-in LCD power supply bleeder resistance and voltage follower op-amp circuits H8/3854 Series: 40 segment pins and up to 16 common pins • Choice of two duty cycles (1/8, 1/16) With 1/8 duty selected: 40 SEG $\times$ 8 COM With 1/16 duty selected: 40 SEG $\times$ 16 COM • Built-in 640-bit display data RAM • Built-in LCD power supply bleeder resistance

Item	Description

Product lineup

H8/3857 Series

Product Code			
Mask ROM Version	**F-ZTAT Version**	**Package**	**ROM/RAM Size**
HD6433855FQ	—	144-pin QFP (FP-144H)	ROM: 40 kbytes
HD6433855TG	—	144-pin TQFP (TFP-144)	RAM: 2 kbytes
HCD6433855	—	Die	
HD6433856FQ	—	144-pin QFP (FP-144H)	ROM: 48 kbytes
HD6433856TG	—	144-pin TQFP (TFP-144)	RAM: 2 kbytes
HCD6433856	—	Die	
HD6433857FQ	HD64F3857FQ	144-pin QFP (FP-144H)	ROM: 60 kbytes
HD6433857TG	HD64F3857TG	144-pin TQFP (TFP-144)	RAM: 2 kbytes
HCD6433857	HCD64F3857	Die	

H8/3854 Series

Product Code			
Mask ROM Version*[2]	**F-ZTAT Version**	**Package**	**ROM/RAM Size**
HD6433852H	—	100-pin QFP (FP-100B)	ROM: 16 kbytes
HD6433852W	—	100-pin TQFP (TFP-100G)	RAM: 1 kbyte
HCD6433852	—	Die	
HD6433853H	—	100-pin QFP (FP-100B)	ROM: 24 kbytes
HD6433853W	—	100-pin TQFP (TFP-100G)	RAM: 1 kbyte
HCD6433853	—	Die	
HD6433854H	—	100-pin QFP (FP-100B)	ROM: 32 kbytes
HD6433854W	—	100-pin TQFP (TFP-100G)	RAM: 1 kbyte
HCD6433854	—	Die	
—	HD64F3854H	100-pin QFP (FP-100B)	ROM: 60 kbytes
—	HD64F3854W	100-pin TQFP (TFP-100G)	RAM: 2 kbytes
—	HCD64F3854	Die	

Notes: 1. ϕ and ϕ_w are defined in section 4, Clock Pulse Generators.
2. Under development

1.2 Internal Block Diagram

Figures 1.1 and 1.2 show internal block diagrams of the H8/3857 Series and H8/3854 Series.

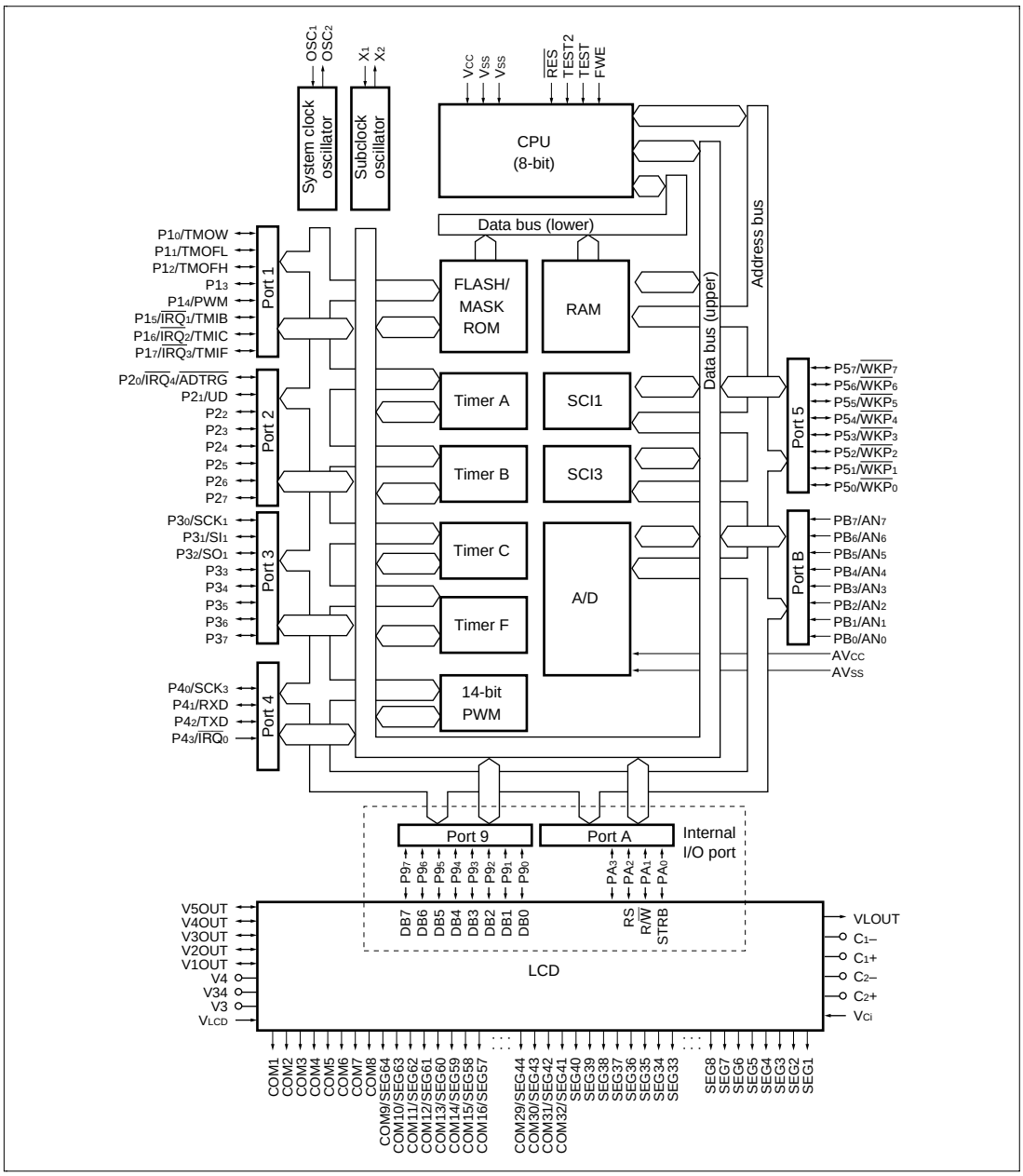


Figure 1.1 H8/3857 Series Internal Block Diagram

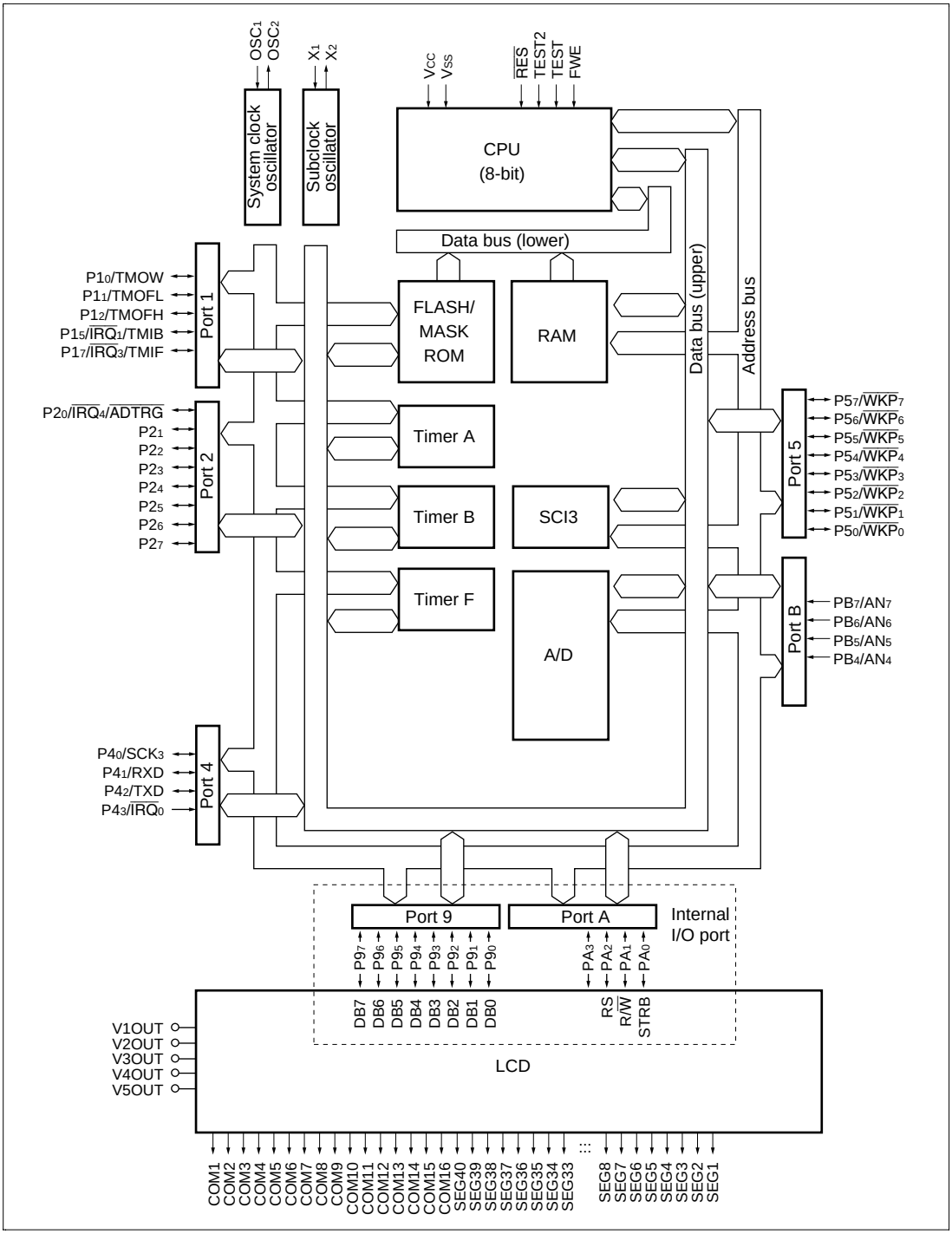


Figure 1.2 H8/3854 Series Internal Block Diagram

1.3 Pin Arrangement and Functions

1.3.1 Pin Arrangement

The pin arrangements of the H8/3857 Series and H8/3854 Series are shown in figures 1.3 and 1.4. The HCD64F3857 pad layout is shown in figure 1.5, and the pad coordinates in table 1.2; the HCD6433855, HCD6433856, and HCD6433857 pad layout is shown in figure 1.6, and the pad coordinates in table 1.3; the HCD64F3854 pad layout is shown in figure 1.7, and the pad coordinates in table 1.4; and the HCD6433852, HCD6433853, and HCD6433854 pad layout is shown in figure 1.8, and the pad coordinates in table 1.5.

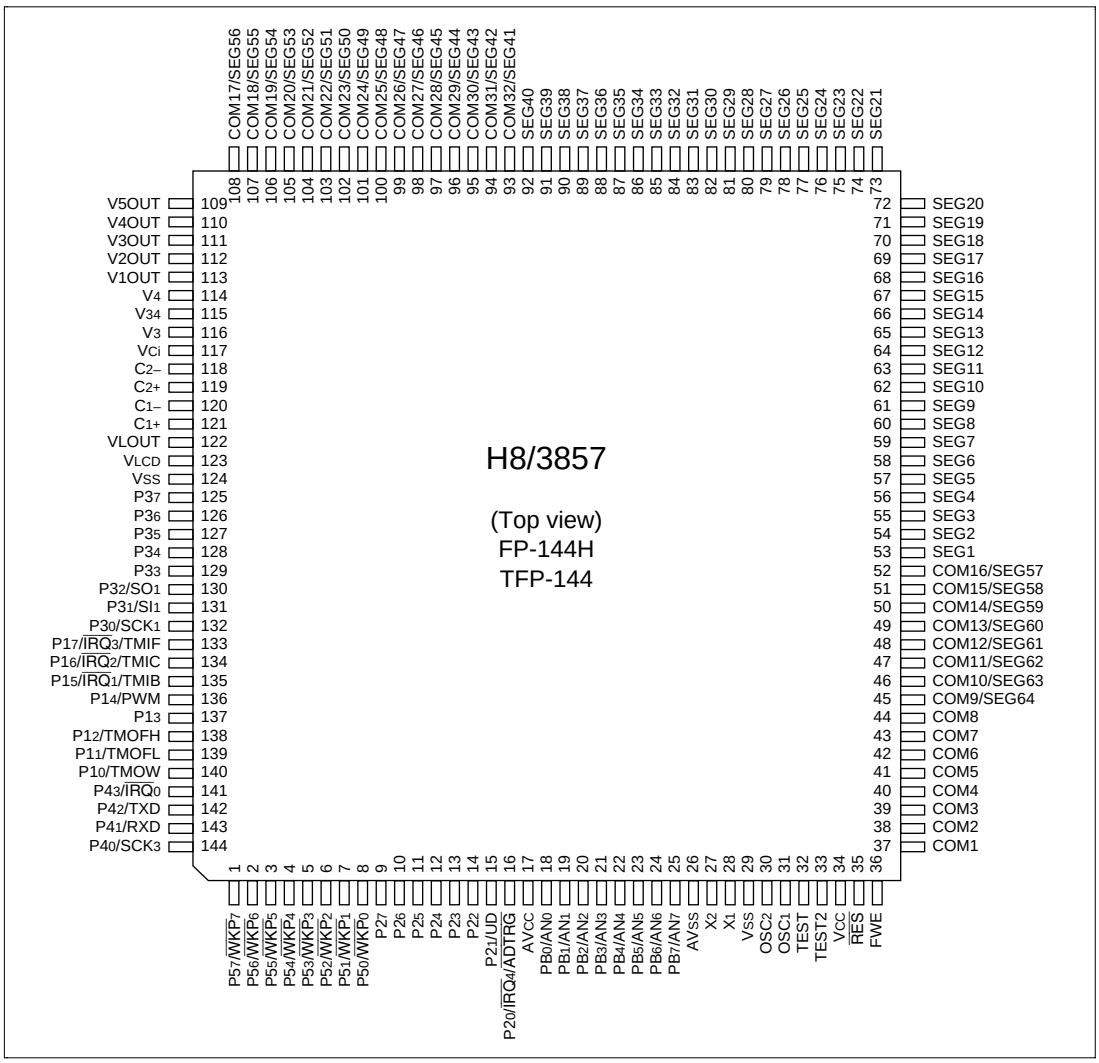


Figure 1.3 H8/3857 Series Pin Arrangement (FP-144H, TFP-144: Top View)

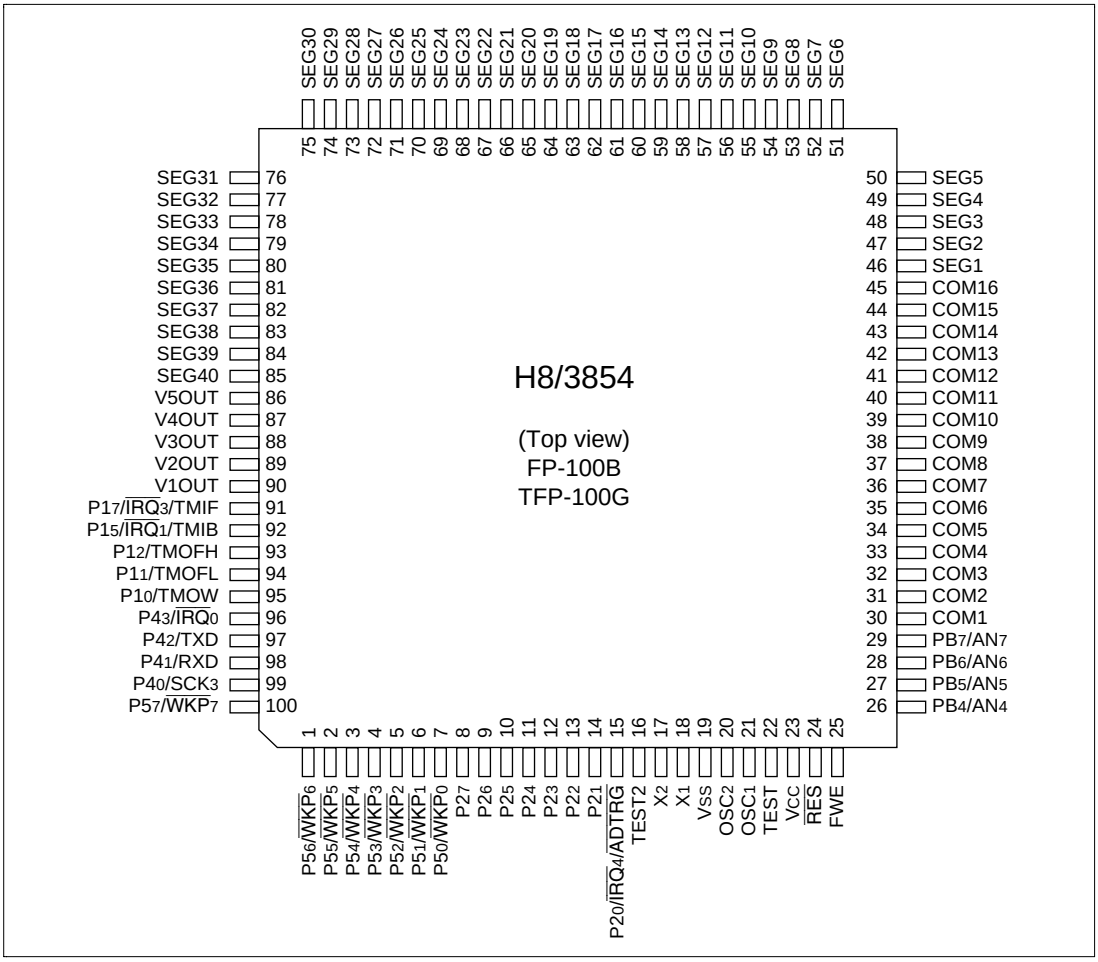


Figure 1.4 H8/3854 Series Pin Arrangement (FP-100B, TFP-100G: Top View)

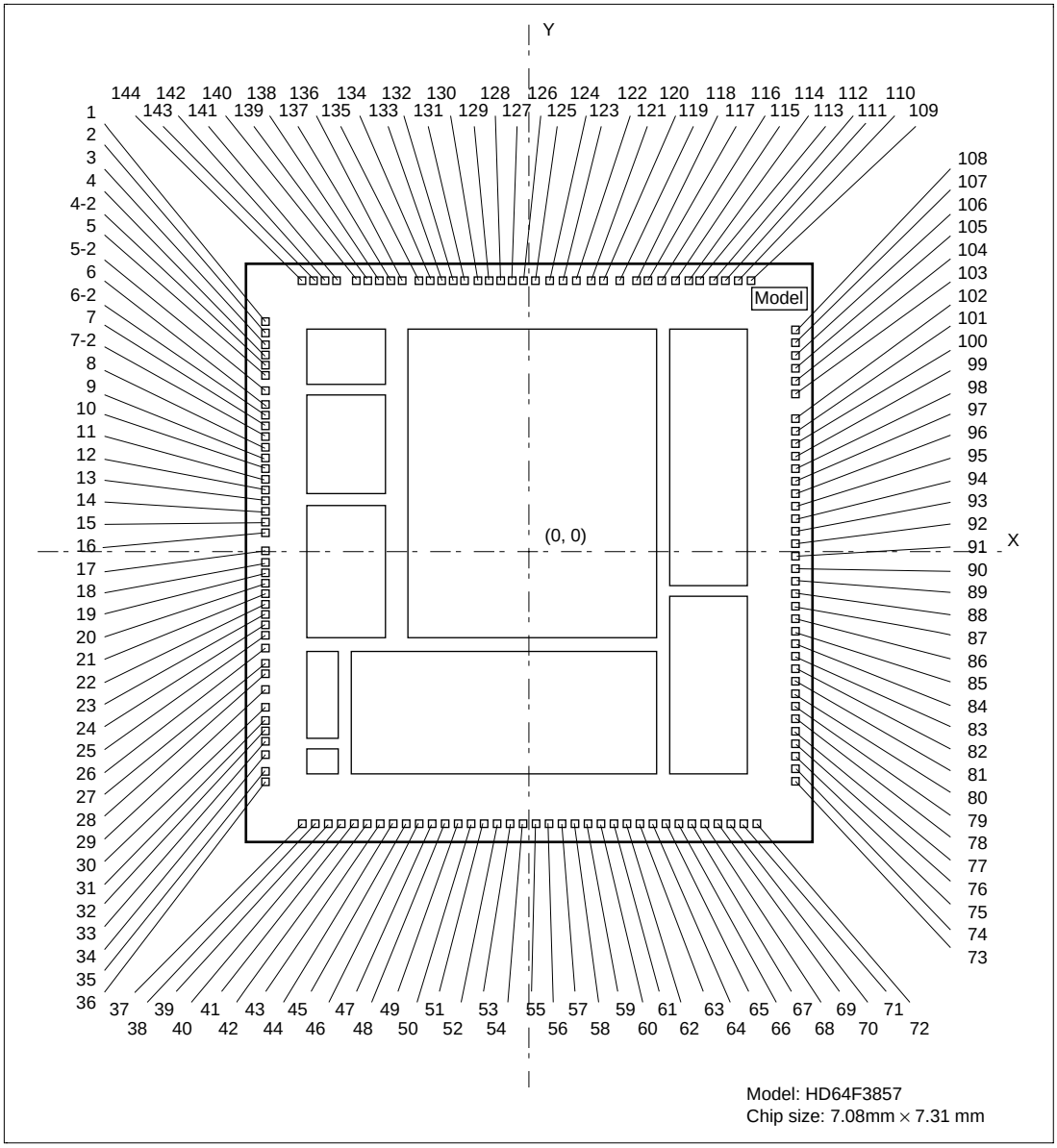


Figure 1.5 Pad Layout of HCD64F3857 (F-ZTAT Version) (Top View)

Table 1.2 HCD64F3857 Pad Coordinates

Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)	Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)	Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)
1	$P5_7/\overline{WKP_7}$	-3348	2928	32	TEST	-3348	-2264	67	SEG15	2032	-3463
2	$P5_6/\overline{WKP_6}$	-3348	2784	33	TEST2	-3348	-2404	68	SEG16	2196	-3463
3	$P5_5/\overline{WKP_5}$	-3348	2640	34	V_{CC}	-3348	-2599	69	SEG17	2360	-3463
4	$P5_4/\overline{WKP_4}$	-3348	2506	35	$\overline{RES}$	-3348	-2794	70	SEG18	2522	-3463
4-2	NC4*2	-3348	2372	36	FWE	-3348	-2944	71	SEG19	2686	-3463
5	$P5_3/\overline{WKP_3}$	-3348	2238	37	COM1	-2862	-3463	72	SEG20	2848	-3463
5-2	NC3*2	-3348	2044	38	COM2	-2700	-3463	73	SEG21	3348	-2907
6	$P5_2/\overline{WKP_2}$	-3348	1854	39	COM3	-2536	-3463	74	SEG22	3348	-2747
6-2	NC2*2	-3348	1720	40	COM4	-2374	-3463	75	SEG23	3348	-2587
7	$P5_1/\overline{WKP_1}$	-3348	1590	41	COM5	-2210	-3463	76	SEG24	3348	-2427
7-2	NC1*2	-3348	1456	42	COM6	-2046	-3463	77	SEG25	3348	-2267
8	$P5_0/\overline{WKP_0}$	-3348	1316	43	COM7	-1884	-3463	78	SEG26	3348	-2107
9	$P2_7$	-3348	1173	44	COM8	-1720	-3463	79	SEG27	3348	-1947
10	$P2_6$	-3348	1039	45	COM9/SEG64	-1556	-3463	80	SEG28	3348	-1787
11	$P2_5$	-3348	905	46	COM10/SEG63	-1394	-3463	81	SEG29	3348	-1627
12	$P2_4$	-3348	771	47	COM11/SEG62	-1231	-3463	82	SEG30	3348	-1467
13	$P2_3$	-3348	637	48	COM12/SEG61	-1069	-3463	83	SEG31	3348	-1307
14	$P2_2$	-3348	503	49	COM13/SEG60	-905	-3463	84	SEG32	3348	-1148
15	$P2_1$/UD	-3348	369	50	COM14/SEG59	-741	-3463	85	SEG33	3348	-988
16	$P2_0/\overline{IRQ_4}/\overline{ADTRG}$	-3348	235	51	COM15/SEG58	-579	-3463	86	SEG34	3348	-828
17	AV_{CC}	-3348	22	52	COM16/SEG57	-415	-3463	87	SEG35	3348	-668
18	PB_0/AN_0	-3348	-143	53	SEG1	-251	-3463	88	SEG36	3348	-508
19	PB_1/AN_1	-3348	-273	54	SEG2	-89	-3463	89	SEG37	3348	-348
20	PB_2/AN_2	-3348	-403	55	SEG3	75	-3463	90	SEG38	3348	-188
21	PB_3/AN_3	-3348	-533	56	SEG4	237	-3463	91	SEG39	3348	-28
22	PB_4/AN_4	-3348	-663	57	SEG5	401	-3463	92	SEG40	3348	132
23	PB_5/AN_5	-3348	-793	58	SEG6	565	-3463	93	COM32/SEG41	3348	292
24	PB_6/AN_6	-3348	-923	59	SEG7	727	-3463	94	COM31/SEG42	3348	452
25	PB_7/AN_7	-3348	-1053	60	SEG8	891	-3463	95	COM30/SEG43	3348	612
26	AV_{SS}	-3348	-1228	61	SEG9	1055	-3463	96	COM29/SEG44	3348	772
27	X_2	-3348	-1438	62	SEG10	1217	-3463	97	COM28/SEG45	3348	932
28	X_1	-3348	-1568	63	SEG11	1380	-3463	98	COM27/SEG46	3348	1092
29	V_{SS}	-3348	-1763	64	SEG12	1542	-3463	99	COM26/SEG47	3348	1252
30	OSC_2	-3348	-1981	65	SEG13	1706	-3463	100	COM25/SEG48	3348	1411
31	OSC_1	-3348	-2134	66	SEG14	1870	-3463	101	COM24/SEG49	3348	1571

Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)	Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)	Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)
102	COM23/SEG50	3348	1731	117	V_{ci}	1471	3463	132	$P3_0/SCK_1$	-982	3463
103	COM22/SEG51	3348	2063	118	C_2-	1341	3463	133	$P1_7/\overline{IRQ}_3$/TMIF	-1116	3463
104	COM21/SEG52	3348	2223	119	C_2+	1125	3463	134	$P1_6/\overline{IRQ}_2$/TMIC	-1250	3463
105	COM20/SEG53	3348	2383	120	C_1-	925	3463	135	$P1_5/\overline{IRQ}_1$/TMIB	-1383	3463
106	COM19/SEG54	3348	2543	121	C_1+	747	3463	136	$P1_4$/PWM	-1611	3463
107	COM18/SEG55	3348	2703	122	VLOUT	569	3463	137	$P1_3$	-1761	3463
108	COM17/SEG56	3348	2863	123	V_{LCD}	395	3463	138	$P1_2$/TMOFH	-1911	3463
109	V5OUT	2776	3463	124	V_{SS}	226	3463	139	$P1_1$/TMOFL	-2045	3463
110	V4OUT	2616	3463	125	$P3_7$	74	3463	140	$P1_0$/TMOW	-2180	3463
111	V3OUT	2456	3463	126	$P3_6$	-78	3463	141	$P4_3/\overline{IRQ}_0$	-2447	3463
112	V2OUT	2296	3463	127	$P3_5$	-234	3463	142	$P4_2$/TXD	-2587	3463
113	V1OUT	2136	3463	128	$P3_4$	-386	3463	143	$P4_1$/RXD	-2737	3463
114	V_4	1976	3463	129	$P3_3$	-538	3463	144	$P4_0/SCK_3$	-2888	3463
115	V_{34}	1816	3463	130	$P3_2/SO_1$	-690	3463				
116	V_3	1656	3463	131	$P3_1/SI_1$	-848	3463				

Notes: 1. Numbers indicate coordinates at the center of the pad area, with an accuracy of ±5 µm.
The origin is the center of the chip, and the center is at a point halfway between pads, horizontally and vertically.
2. NC1 to NC4 are test pads; they should be left open.

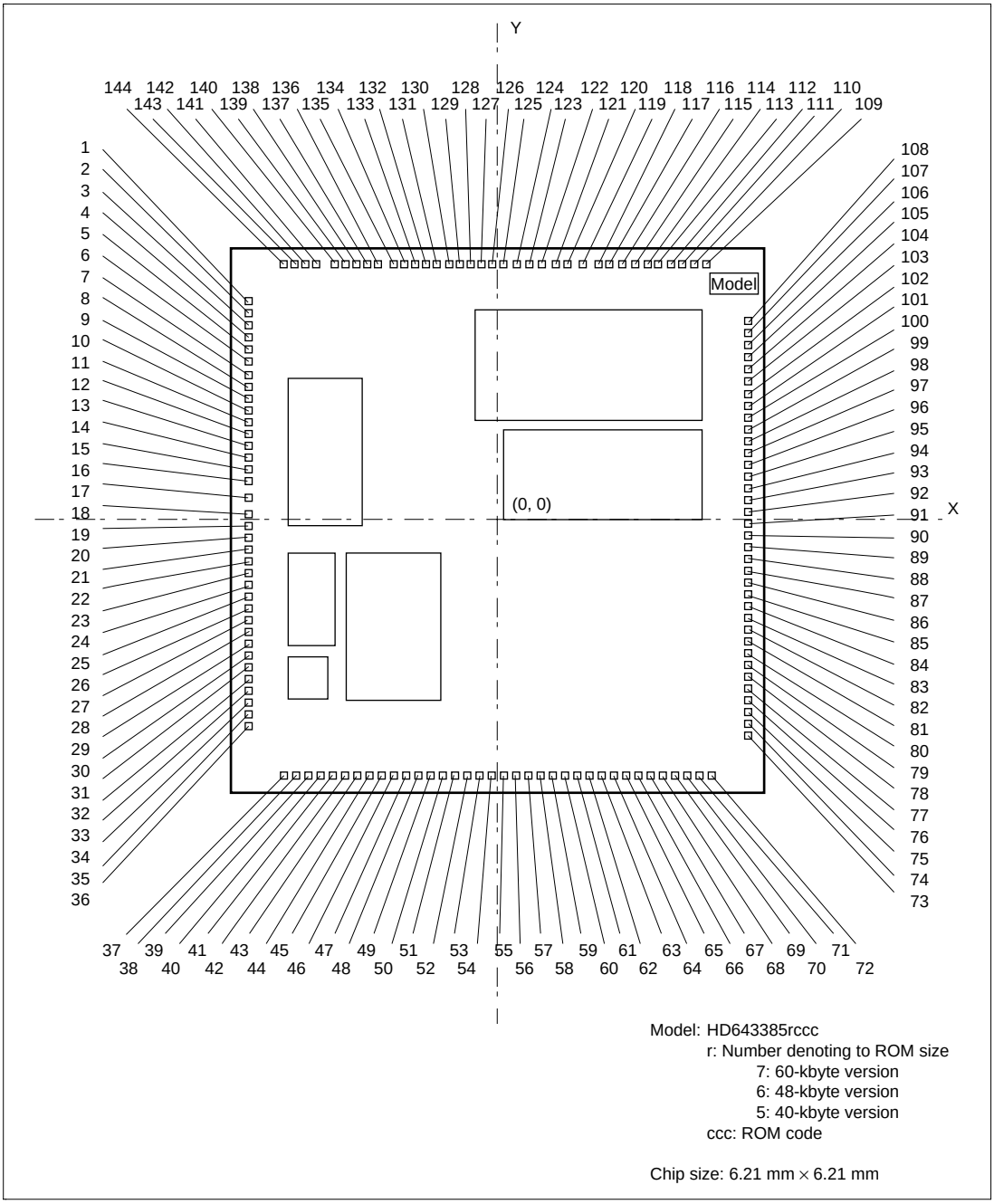


Figure 1.6 Pad Layout of HCD6433855, HCD6433856, and HCD6433857 (Mask ROM Version) (Top View)

Table 1.3 HCD6433855, HCD6433856, and HCD6433857 Pad Coordinates

Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)	Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)	Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)
1	$P5_7/\overline{WKP}_7$	-2913	2515	36	FWE*2	-2913	-2534	71	SEG19	2305	-2913
2	$P5_6/\overline{WKP}_6$	-2913	2365	37	COM1	-2495	-2913	72	SEG20	2495	-2913
3	$P5_5/\overline{WKP}_5$	-2913	2215	38	COM2	-2305	-2913	73	SEG21	2913	-2495
4	$P5_4/\overline{WKP}_4$	-2913	2070	39	COM3	-2125	-2913	74	SEG22	2913	-2305
5	$P5_3/\overline{WKP}_3$	-2913	1930	40	COM4	-1955	-2913	75	SEG23	2913	-2125
6	$P5_2/\overline{WKP}_2$	-2913	1795	41	COM5	-1795	-2913	76	SEG24	2913	-1955
7	$P5_1/\overline{WKP}_1$	-2913	1660	42	COM6	-1645	-2913	77	SEG25	2913	-1795
8	$P5_0/\overline{WKP}_0$	-2913	1530	43	COM7	-1505	-2913	78	SEG26	2913	-1645
9	$P2_7$	-2913	1400	44	COM8	-1365	-2913	79	SEG27	2913	-1505
10	$P2_6$	-2913	1271	45	COM9/SEG64	-1235	-2913	80	SEG28	2913	-1365
11	$P2_5$	-2913	1141	46	COM10/SEG63	-1105	-2913	81	SEG29	2913	-1235
12	$P2_4$	-2913	1011	47	COM11/SEG62	-975	-2913	82	SEG30	2913	-1105
13	$P2_3$	-2913	881	48	COM12/SEG61	-845	-2913	83	SEG31	2913	-975
14	$P2_2$	-2913	751	49	COM13/SEG60	-715	-2913	84	SEG32	2913	-845
15	$P2_1$ /UD	-2913	621	50	COM14/SEG59	-585	-2913	85	SEG33	2913	-715
16	$P2_0/\overline{IRQ}_4/\overline{ADTRG}$	-2913	491	51	COM15/SEG58	-455	-2913	86	SEG34	2913	-585
17	AV_{CC}	-2913	290	52	COM16/SEG57	-325	-2913	87	SEG35	2913	-455
18	PB_0/AN_0	-2913	125	53	SEG1	-195	-2913	88	SEG36	2913	-325
19	PB_1/AN_1	-2913	-5	54	SEG2	-65	-2913	89	SEG37	2913	-195
20	PB_2/AN_2	-2913	-135	55	SEG3	65	-2913	90	SEG38	2913	-65
21	PB_3/AN_3	-2913	-265	56	SEG4	195	-2913	91	SEG39	2913	65
22	PB_4/AN_4	-2913	-395	57	SEG5	325	-2913	92	SEG40	2913	195
23	PB_5/AN_5	-2913	-525	58	SEG6	455	-2913	93	COM32/SEG41	2913	325
24	PB_6/AN_6	-2913	-655	59	SEG7	585	-2913	94	COM31/SEG42	2913	455
25	PB_7/AN_7	-2913	-785	60	SEG8	715	-2913	95	COM30/SEG43	2913	585
26	AV_{SS}	-2913	-960	61	SEG9	845	-2913	96	COM29/SEG44	2913	715
27	X_2	-2913	-1169	62	SEG10	975	-2913	97	COM28/SEG45	2913	845
28	X_1	-2913	-1299	63	SEG11	1105	-2913	98	COM27/SEG46	2913	975
29	V_{SS}	-2913	-1428	64	SEG12	1235	-2913	99	COM26/SEG47	2913	1105
30	OSC_2	-2913	-1581	65	SEG13	1365	-2913	100	COM25/SEG48	2913	1235
31	OSC_1	-2913	-1734	66	SEG14	1505	-2913	101	COM24/SEG49	2913	1365
32	TEST	-2913	-1874	67	SEG15	1645	-2913	102	COM23/SEG50	2913	1505
33	TEST2	-2913	-2024	68	SEG16	1795	-2913	103	COM22/SEG51	2913	1645
34	V_{CC}	-2913	-2189	69	SEG17	1955	-2913	104	COM21/SEG52	2913	1795
35	$\overline{RES}$	-2913	-2384	70	SEG18	2125	-2913	105	COM20/SEG53	2913	1955

Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)	Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)	Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)
106	COM19/SEG54	2913	2125	119	C_2+	995	2913	132	$P3_0$ /SCK_1	-715	2913
107	COM18/SEG55	2913	2305	120	C_1–	865	2913	133	$P1_7$/$\overline{IRQ}_3$/TMIF	-845	2913
108	COM17/SEG56	2913	2495	121	C_1+	735	2913	134	$P1_6$/$\overline{IRQ}_2$/TMIC	-975	2913
109	V5OUT	2435	2913	122	VLOUT	605	2913	135	$P1_5$/$\overline{IRQ}_1$/TMIB	-1105	2913
110	V4OUT	2275	2913	123	V_{LCD}	475	2913	136	$P1_4$/PWM	-1235	2913
111	V3OUT	2125	2913	124	V_{SS}	325	2913	137	$P1_3$	-1365	2913
112	V2OUT	1975	2913	125	$P3_7$	195	2913	138	$P1_2$/TMOFH	-1505	2913
113	V1OUT	1825	2913	126	$P3_6$	65	2913	139	$P1_1$/TMOFL	-1645	2913
114	V_4	1675	2913	127	$P3_5$	-65	2913	140	$P1_0$/TMOW	-1795	2913
115	V_{34}	1520	2913	128	$P3_4$	-195	2913	141	$P4_3$/$\overline{IRQ}_0$	-1955	2913
116	V_3	1385	2913	129	$P3_3$	-325	2913	142	$P4_2$/TXD	-2125	2913
117	V_{ci}	1255	2913	130	$P3_2$/SO_1	-455	2913	143	$P4_1$/RXD	-2305	2913
118	C_2–	1125	2913	131	$P3_1$/SI_1	-585	2913	144	$P4_0$/SCK_3	-2495	2913

Notes: 1. Numbers indicate coordinates at the center of the pad area, with an accuracy of ±5 µm. The origin is the center of the chip, and the center is at a point halfway between pads, horizontally and vertically.

2. Connect FWE to V_{SS}.

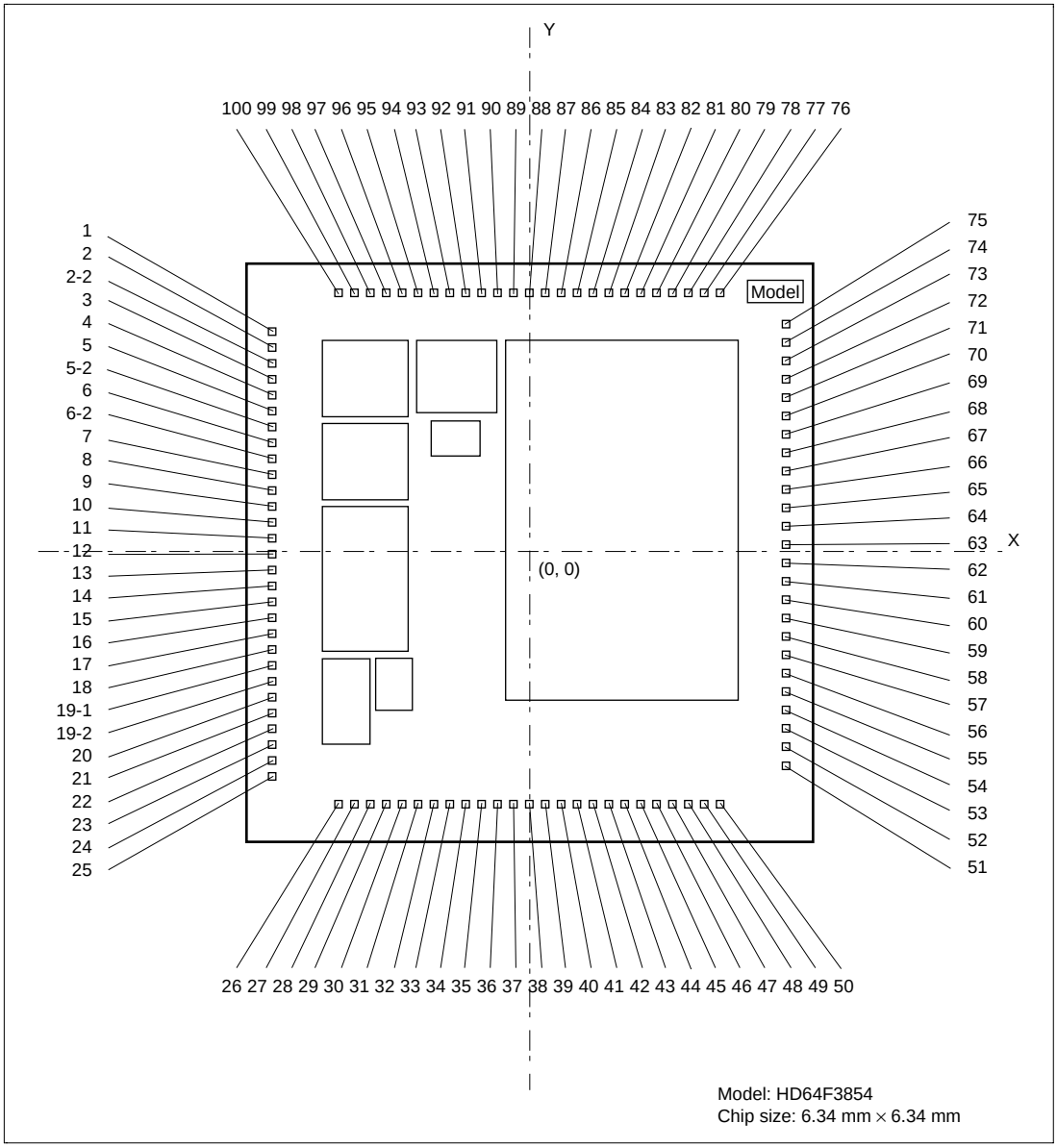


Figure 1.7 Pad Layout of HCD64F3854 (F-ZTAT Version) (Top View)

Table 1.4 HCD64F3854 Pad Coordinates

Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)	Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)	Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)
1	$P5_6/\overline{WKP_6}$	-2985	2494	31	COM2	-1413	-2985	66	SEG21	2985	627
2	$P5_5/\overline{WKP_5}$	-2985	2333	32	COM3	-1210	-2985	67	SEG22	2985	831
2-2	NC1*2	-2985	2139	33	COM4	-1006	-2985	68	SEG23	2985	1036
3	$P5_4/\overline{WKP_4}$	-2985	1950	34	COM5	-801	-2985	69	SEG24	2985	1240
4	$P5_3/\overline{WKP_3}$	-2985	1788	35	COM6	-597	-2985	70	SEG25	2985	1443
5	$P5_2/\overline{WKP_2}$	-2985	1626	36	COM7	-393	-2985	71	SEG26	2985	1647
5-2	NC2*2	-2985	1419	37	COM8	-189	-2985	72	SEG27	2985	1851
6	$P5_1/\overline{WKP_1}$	-2985	1215	38	COM9	16	-2985	73	SEG28	2985	2055
6-2	NC3*2	-2985	1054	39	COM10	219	-2985	74	SEG29	2985	2259
7	$P5_0/\overline{WKP_0}$	-2985	897	40	COM11	423	-2985	75	SEG30	2985	2463
8	$P2_7$	-2985	735	41	COM12	627	-2985	76	SEG31	2435	2985
9	$P2_6$	-2985	573	42	COM13	831	-2985	77	SEG32	2234	2985
10	$P2_5$	-2985	412	43	COM14	1035	-2985	78	SEG33	2032	2985
11	$P2_4$	-2985	250	44	COM15	1240	-2985	79	SEG34	1830	2985
12	$P2_3$	-2985	88	45	COM16	1443	-2985	80	SEG35	1629	2985
13	$P2_2$	-2985	-73	46	SEG1	1647	-2985	81	SEG36	1427	2985
14	$P2_1$	-2985	-234	47	SEG2	1851	-2985	82	SEG37	1226	2985
15	$P2_0/\overline{IRQ_4}/\overline{ADTRG}$	-2985	-396	48	SEG3	2055	-2985	83	SEG38	1025	2985
16	TEST2	-2985	-558	49	SEG4	2259	-2985	84	SEG39	823	2985
17	X_2	-2985	-716	50	SEG5	2463	-2985	85	SEG40	621	2985
18	X_1	-2985	-873	51	SEG6	2985	-2433	86	V5OUT	434	2985
19-1	V_{SS}*3	-2985	-1031	52	SEG7	2985	-2229	87	V4OUT	233	2985
19-2	V_{SS}*3	-2985	-1267	53	SEG8	2985	-2025	88	V3OUT	31	2985
20	OSC_2	-2985	-1526	54	SEG9	2985	-1821	89	V2OUT	-171	2985
21	OSC_1	-2985	-1707	55	SEG10	2985	-1617	90	V1OUT	-372	2985
22	TEST	-2985	-1864	56	SEG11	2985	-1413	91	$P1_7/\overline{IRQ_3}$/TMIF	-574	2985
23	V_{CC}	-2985	-2101	57	SEG12	2985	-1210	92	$P1_5/\overline{IRQ_1}$/TMIB	-780	2985
24	$\overline{RES}$	-2985	-2336	58	SEG13	2985	-1005	93	$P1_2$/TMOFH	-985	2985
25	FWE	-2985	-2494	59	SEG14	2985	-801	94	$P1_1$/TMOFL	-1191	2985
26	PB_4/AN_4	-2448	-2985	60	SEG15	2985	-597	95	$P1_0$/TMOW	-1396	2985
27	PB_5/AN_5	-2244	-2985	61	SEG16	2985	-393	96	$P4_3/\overline{IRQ_0}$	-1618	2985
28	PB_6/AN_6	-2040	-2985	62	SEG17	2985	-189	97	$P4_2$/TXD	-1820	2985
29	PB_7/AN_7	-1836	-2985	63	SEG18	2985	15	98	$P4_1$/RXD	-2022	2985
30	COM1	-1617	-2985	64	SEG19	2985	219	99	$P4_0/SCK_3$	-2234	2985
				65	SEG20	2985	423	100	$P5_7/\overline{WKP_7}$	-2446	2985

Notes: 1. Numbers indicate coordinates at the center of the pad area, with an accuracy of ±5 µm. The origin is the center of the chip, and the center is at a point halfway between pads, horizontally and vertically.
2. NC1 to NC3 are test pads; they should be left open.
3. Connect both 19-1 and 19-2 to V_{SS}.

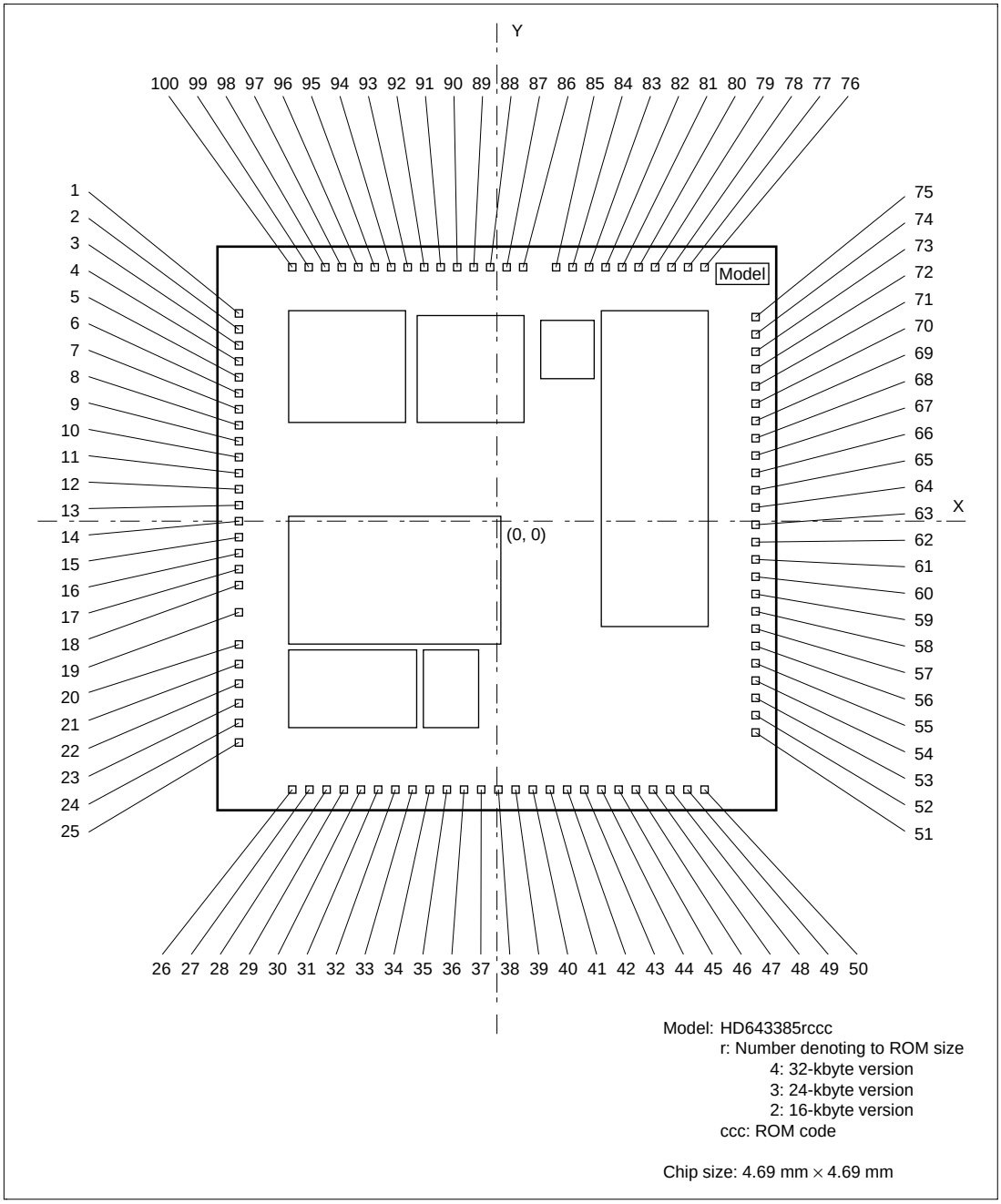


Figure 1.8 Pad Layout of HCD6433854, HCD6433853, and HCD6433852 (Mask ROM Version) (Top View)

Table 1.5 HCD6433852, HCD6433853, and HCD6433854 Pad Coordinates

Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)	Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)	Pad No.	Pad Name	Coordinates*1 X (µm)	Y (µm)
1	$P5_6/\overline{WKP}_6$	-2161	1738	35	COM6	-390	-2161	68	SEG23	2161	650
2	$P5_5/\overline{WKP}_5$	-2161	1590	36	COM7	-260	-2161	69	SEG24	2161	780
3	$P5_4/\overline{WKP}_4$	-2161	1445	37	COM8	-130	-2161	70	SEG25	2161	910
4	$P5_3/\overline{WKP}_3$	-2161	1305	38	COM9	0	-2161	71	SEG26	2161	1060
5	$P5_2/\overline{WKP}_2$	-2161	1171	39	COM10	130	-2161	72	SEG27	2161	1213
6	$P5_1/\overline{WKP}_1$	-2161	1041	40	COM11	260	-2161	73	SEG28	2161	1383
7	$P5_0/\overline{WKP}_0$	-2161	911	41	COM12	390	-2161	74	SEG29	2161	1551
8	$P2_7$	-2161	781	42	COM13	520	-2161	75	SEG30	2161	1721
9	$P2_6$	-2161	651	43	COM14	650	-2161	76	SEG31	1716	2161
10	$P2_5$	-2161	521	44	COM15	780	-2161	77	SEG32	1565	2161
11	$P2_4$	-2161	391	45	COM16	910	-2161	78	SEG33	1422	2161
12	$P2_3$	-2161	261	46	SEG1	1060	-2161	79	SEG34	1282	2161
13	$P2_2$	-2161	131	47	SEG2	1213	-2161	80	SEG35	1150	2161
14	$P2_1$	-2161	1	48	SEG3	1383	-2161	81	SEG36	1020	2161
15	$P2_0/\overline{IRQ}_4/\overline{ADTRG}$	-2161	-129	49	SEG4	1551	-2161	82	SEG37	890	2161
16	TEST2	-2161	-259	50	SEG5	1721	-2161	83	SEG38	760	2161
17	X_2	-2161	-389	51	SEG6	2161	-1721	84	SEG39	630	2161
18	X_1	-2161	-519	52	SEG7	2161	-1551	85	SEG40	500	2161
19	V_{SS}	-2161	-764	53	SEG8	2161	-1383	86	V5OUT	197	2161
20	OSC_2	-2161	-1020	54	SEG9	2161	-1213	87	V4OUT	67	2161
21	OSC_1	-2161	-1173	55	SEG10	2161	-1060	88	V3OUT	-63	2161
22	TEST	-2161	-1312	56	SEG11	2161	-910	89	V2OUT	-193	2161
23	V_{CC}	-2161	-1497	57	SEG12	2161	-780	90	V1OUT	-323	2161
24	$\overline{RES}$	-2161	-1657	58	SEG13	2161	-650	91	$P1_7/\overline{IRQ}_3$/TMIF	-453	2161
25	FWE*2	-2161	-1821	59	SEG14	2161	-520	92	$P1_5/\overline{IRQ}_1$/TMIB	-583	2161
26	PB_4/AN_4	-1722	-2161	60	SEG15	2161	-390	93	$P1_2$/TMOFH	-713	2161
27	PB_5/AN_5	-1552	-2161	61	SEG16	2161	-260	94	$P1_1$/TMOFL	-843	2161
28	PB_6/AN_6	-1395	-2161	62	SEG17	2161	-130	95	$P1_0$/TMOW	-973	2161
29	PB_7/AN_7	-1236	-2161	63	SEG18	2161	0	96	$P4_3/\overline{IRQ}_0$	-1104	2161
30	COM1	-1060	-2161	64	SEG19	2161	130	97	$P4_2$/TXD	-1244	2161
31	COM2	-910	-2161	65	SEG20	2161	260	98	$P4_1$/RXD	-1383	2161
32	COM3	-780	-2161	66	SEG21	2161	390	99	$P4_0/SCK_3$	-1534	2161
33	COM4	-650	-2161	67	SEG22	2161	520	100	$P5_7/\overline{WKP}_7$	-1698	2161
34	COM5	-520	-2161								

Notes: 1. Numbers indicate coordinates at the center of the pad area, with an accuracy of ±5 µm. The origin is the center of the chip, and the center is at a point halfway between pads, horizontally and vertically.

2. Connect FWE to V_{SS}.

1.3.2 Pin Functions

Table 1.6 outlines the pin functions of the H8/3857 Series.

Table 1.6 Pin Functions

		H8/3857 Series		H8/3854 Series			
		Pin No.		Pin No.			
Type	Symbol	FP-144H TFP-144	Pad No.	FP-100B TFP-100G	Pad No.	I/O	Name and Functions
Power source pins	V_{CC}	34	34	23	23	Input	**Power supply:** All V_{CC} pins should be connected to the system power supply (+5 V)
	V_{SS}	29, 124	29, 124	19	19-1, 19-2	Input	**Ground:** All V_{SS} pins should be connected to the system power supply (0 V)
	AV_{CC}	17	17	—	—	Input	**Analog power supply (H8/3857 Series only):** This is the power supply pin for the A/D converter. When the A/D converter is not used, connect this pin to the system power supply (+5 V).
	AV_{SS}	26	26	—	—	Input	**Analog ground (H8/3857 Series only):** This is the A/D converter ground pin. It should be connected to the system power supply (0 V).
Clock pins	OSC_1	31	31	21	21	Input	This pin connects to a crystal or ceramic oscillator.
	OSC_2	30	30	20	20	Output	See section 4, Clock Pulse Generators, for a typical connection diagram.
	X_1	28	28	18	18	Input	This pin connects to a 32.768-kHz crystal oscillator, or can be used to input an external clock.
	X_2	27	27	17	17	Output	See section 4, Clock Pulse Generators, for a typical connection diagram.

		H8/3857 Series		H8/3854 Series			
		Pin No.		Pin No.			
Type	Symbol	FP-144H TFP-144	Pad No.	FP-100B TFP-100G	Pad No.	I/O	Name and Functions
System control	$\overline{RES}$	35	35	24	24	Input	**Reset:** When this pin is driven low, the chip is reset
	FWE	36	36	25	25	Input	**Flash write enable:** This pin enables or disables flash memory programming. In the mask ROM version, ground this pin to the V_{SS} potential.
	TEST	32	32	22	22	Input	**Test:** This is a test pin, not for use in application systems. It should be connected to V_{SS}.
	TEST2	33	33	16	16	Input	**Test pin:** This pin sets the flash memory operating mode. In the mask ROM version, ground this pin to the V_{SS} potential.
Interrupt pins	$\overline{IRQ_0}$ $\overline{IRQ_1}$ $\overline{IRQ_2}$ $\overline{IRQ_3}$ $\overline{IRQ_4}$	141 135 134 133 16	141 135 134 133 16	96 92 — 91 15	96 92 — 91 15	Input	**External interrupt request 4 to 0 (H8/3857 Series)** **External interrupt request 4, 3, 1, 0 (H8/3854 Series)** These are input pins for external interrupts for which there is a choice between rising and falling edge sensing.
	$\overline{WKP_7}$ to $\overline{WKP_0}$	1 to 8	1 to 8	100, 1 to 7	100, 1 to 7	Input	**Wakeup interrupt request 0 to 7:** These are input pins for external interrupts that are detected at the falling edge
Timer pins	TMOW	140	140	95	95	Output	**Clock output:** This is an output pin for waveforms generated by the timer A output circuit
	TMIB	135	135	92	92	Input	**Timer B event counter input:** This is an event input pin for input to the timer B counter
	TMIC	134	134	—	—	Input	**Timer C event counter input (H8/3857 Series only):** This is an event input pin for input to the timer C counter

		H8/3857 Series		H8/3854 Series			
		Pin No.		Pin No.			
Type	Symbol	FP-144H TFP-144	Pad No.	FP-100B TFP-100G	Pad No.	I/O	Name and Functions
Timer pins	UD	15	15	—	—	Input	**Timer C up/down select (H8/3857 Series only):** This pin selects whether the timer C counter is used for up- or down-counting. At high level it selects up-counting, and at low level down-counting.
	TMIF	133	133	91	91	Input	**Timer F event counter input:** This is an event input pin for input to the timer F counter
	TMOFL	139	139	94	94	Output	**Timer FL output:** This is an output pin for waveforms generated by the timer FL output compare function
	TMOFH	138	138	93	93	Output	**Timer FH output:** This is an output pin for waveforms generated by the timer FH output compare function
14-bit PWM pin	PWM	136	136	—	—	Output	**14-bit PWM output (H8/3857 Series only):** This is an output pin for waveforms generated by the 14-bit PWM
I/O ports	PB_7 to PB_4 PB_3 to PB_0	25 to 22 21 to 18	25 to 22 21 to 18	29 to 26 —	29 to 26 —	Input	**Port B:** This is an 8-bit input port in the H8/3857 Series, and a 4-bit input port in the H8/3854 Series
	$P4_3$	141	141	96	96	Input	**Port 4 (bit 3):** This is a 1-bit input port
	$P4_2$ to $P4_0$	142 to 144	142 to 144	97 to 99	97 to 99	I/O	**Port 4 (bits 2 to 0):** This is a 3-bit I/O port. Input or output can be designated for each bit by means of port control register 4 (PCR4).
	$P1_7$, $P1_6$, $P1_5$, $P1_4$, $P1_3$, $P1_2$ to $P1_0$	133 134 135 136, 137 138 to 140	133 134 135 136, 137 138 to 140	91 — 92 — 93 to 95	91 — 92 — 93 to 95	I/O	**Port 1:** This is an 8-bit I/O port in the H8/3857 Series, and a 5-bit I/O port in the H8/3854 Series. Input or output can be designated for each bit by means of port control register 1 (PCR1).

Type	Symbol	H8/3857 Series Pin No. FP-144H TFP-144	H8/3857 Series Pin No. Pad No.	H8/3854 Series Pin No. FP-100B TFP-100G	H8/3854 Series Pin No. Pad No.	I/O	Name and Functions
I/O ports	$P2_7$ to $P2_0$	9 to 16	9 to 16	8 to 15	8 to 15	I/O	**Port 2:** This is an 8-bit I/O port. Input or output can be designated for each bit by means of port control register 2 (PCR2).
	$P3_7$ to $P3_0$	125 to 132	125 to 132	—	—	I/O	**Port 3 (H8/3857 Series only):** This is an 8-bit I/O port. Input or output can be designated for each bit by means of port control register 3 (PCR3).
	$P5_7$ to $P5_0$	1 to 8	1 to 8	100, 1 to 7	100, 1 to 7	I/O	**Port 5:** This is an 8-bit I/O port. Input or output can be designated for each bit by means of port control register 5 (PCR5).
Internal I/O ports	PA_3 to PA_0	—	—	—	—	I/O	**Port A:** This is a 4-bit I/O port. Input or output can be designated for each bit by means of port control register A (PCRA). PA_2 to PA_0 are connected internally to LCD pins RS, R/$\overline{W}$, and STRB.
	$P9_7$ to $P9_0$	—	—	—	—	I/O	**Port 9:** This is an 8-bit I/O port. Input or output can be designated for each bit by means of port control register 9 (PCR9). $P9_7$ to $P9_0$ are connected internally to LCD pins DB_7 to DB_0.
Serial communication interface (SCI)	SI_1	131	131	—	—	Input	**SCI1 receive data input (H8/3857 Series only):** This is the SCI1 data input pin
	SO_1	130	130	—	—	Output	**SCI1 send data output (H8/3857 Series only):** This is the SCI1 data output pin
	SCK_1	132	132	—	—	I/O	**SCI1 clock I/O (H8/3857 Series only):** This is the SCI1 clock I/O pin
	RXD	143	143	98	98	Input	**SCI3 receive data input:** This is the SCI3 data input pin

		H8/3857 Series		H8/3854 Series			
		Pin No.		Pin No.			
Type	Symbol	FP-144H TFP-144	Pad No.	FP-100B TFP-100G	Pad No.	I/O	Name and Functions
Serial communication interface (SCI)	TXD	142	142	97	97	Output	**SCI3 send data output:** This is the SCI3 data output pin
	SCK_3	144	144	99	99	I/O	**SCI3 clock I/O :** This is the SCI3 clock I/O pin
A/D converter	AN_7 to AN_4 AN_3 to AN_0	25 to 22 21 to 18	25 to 22 21 to 18	29 to 26 —	29 to 26 —	Input	**Analog input (channels 7 to 0 in H8/3857 Series, channels 7 to 4 in H8/3854 Series):** These are analog data input channels to the A/D converter
	$\overline{ADTRG}$	16	16	15	15	Input	**A/D converter trigger input:** This is the external trigger input pin to the A/D converter
LCD controller	COM32 to COM17 COM16 to COM1	93 to 108 52 to 37	93 to 108 52 to 37	— 45 to 30	— 45 to 30	Output	**LCD common output:** These are LCD common output pins. The maximum number of pins is 32 in the H8/3857 Series, and 16 in the H8/3854 Series. In standby mode and module standby mode, all pins output the V_{SS} level.
	SEG64 to SEG41 SEG40 to SEG1	45 to 52 108 to 93, 92 to 53	45 to 52 108 to 93, 92 to 53	— 85 to 46	— 85 to 46	Output	**LCD segment output:** These are LCD segment output pins. The maximum number of pins is 64 in the H8/3857 Series, and 40 in the H8/3854 Series. In standby mode and module standby mode, all pins output the V_{SS} level.
	V3, V4	116, 114	116, 114	—	—	Input	**LCD bias setting pins (H8/3857 Series only):** 1/4 bias drive is selected when V3 and V4 are shorted, and 1/5 bias drive when V3 and V4 are left open.
	V34	115	115	—	—	Input	**LCD test pin (H8/3857 Series only):** This is the LCD built-in resistance test pin. V3 and V34 must be shorted.

		H8/3857 Series		H8/3854 Series			
		Pin No.		Pin No.			
Type	Symbol	FP-144H TFP-144	Pad No.	FP-100B TFP-100G	Pad No.	I/O	Name and Functions
LCD controller	C1+, C1–, C2+, C2–	121 to 118	121 to 118	—	—	—	**LCD step-up circuit capacitance connection pins (H8/3857 Series only):** These pins connect external capacitances for LCD step-up. Connect capacitances according to the step-up factor.
H8/3857 Series LCD power supply	V1OUT to V5OUT	113 to 109	113 to 109	—	—	I/O	**LCD drive power supply level (H8/3857 Series):** When the OPON bit is set high, these bits output LCD drive power supply levels V1 to V5. If the built-in op-amp drive capacity is inadequate, connect a capacitor to provide stabilization. If levels V1 to V5 are input from an external source, set the OPON bit low.
	V_{ci}	117	117	—	—	Input	**LCD step-up circuit reference power supply (H8/3857 Series only):** This pin doubles as the LCD step-up circuit reference input voltage and step-up circuit power supply, and provides the LCD drive voltage. Set 1.6 V ≤ V_{ci} ≤ V_{CC}. If the step-up circuit is not used, connect V_{ci} to V_{CC}.
	VLOUT	122	122	—	—	Output	**LCD step-up voltage output (H8/3857 Series only):** This is the LCD step-up voltage output pin. Connect a capacitance.
	V_{LCD}	123	123	—	—	Input	**LCD drive power supply (H8/3857 Series only):** This is the LCD drive power supply input pin. If the built-in step-up circuit output voltage is used for the LCD drive power supply, short this pin to VLOUT. Set V_{CC} ≤ V_{LCD} ≤ 7.0 V.

		H8/3857 Series		H8/3854 Series			
		Pin No.		Pin No.			
Type	Symbol	FP-144H TFP-144	Pad No.	FP-100B TFP-100G	Pad No.	I/O	Name and Functions
H8/3854 Series LCD power supply	V1OUT to V5OUT	—	—	90 to 86	90 to 86	I/O	**LCD drive power supply level (H8/3854 Series):** When the LPS1 and LPS0 bits are set high, these pins output LCD drive power supply levels V1 to V5. If the drive capacity is inadequate, connect a capacitor to provide stabilization. If levels V1 to V5 are input from an external source, set the LPS1 and LPS0 bits low. The V1 to V5 levels must not exceed V_{CC}. When driving with a 1/4 bias, short V3OUT and V4OUT.

Section 2 CPU

2.1 Overview

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

2.1.1 Features

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

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

Note: * These values are at ϕ = 5 MHz.

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

2.1.2 Address Space

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

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

2.1.3 Register Configuration

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

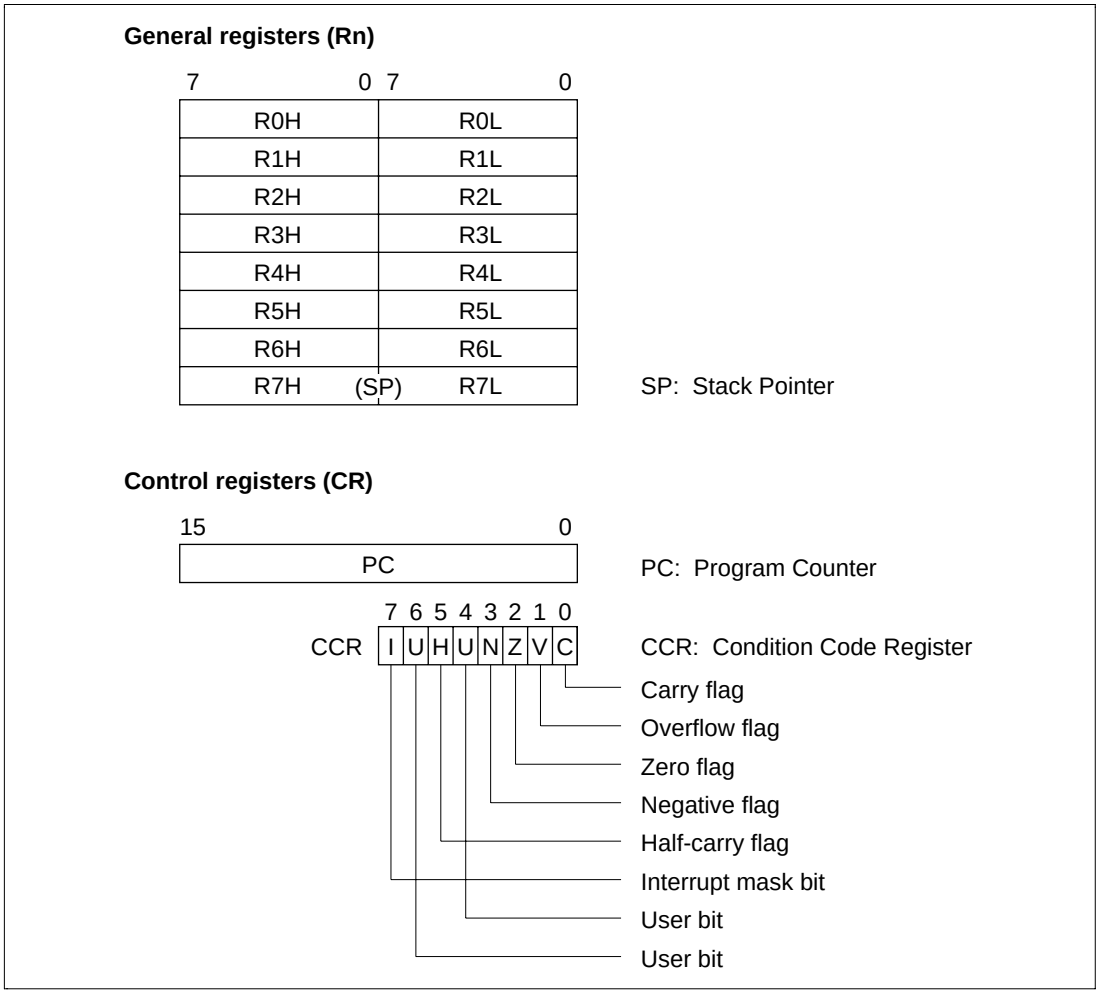


Figure 2.1 CPU Registers

2.2 Register Descriptions

2.2.1 General Registers

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

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

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

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

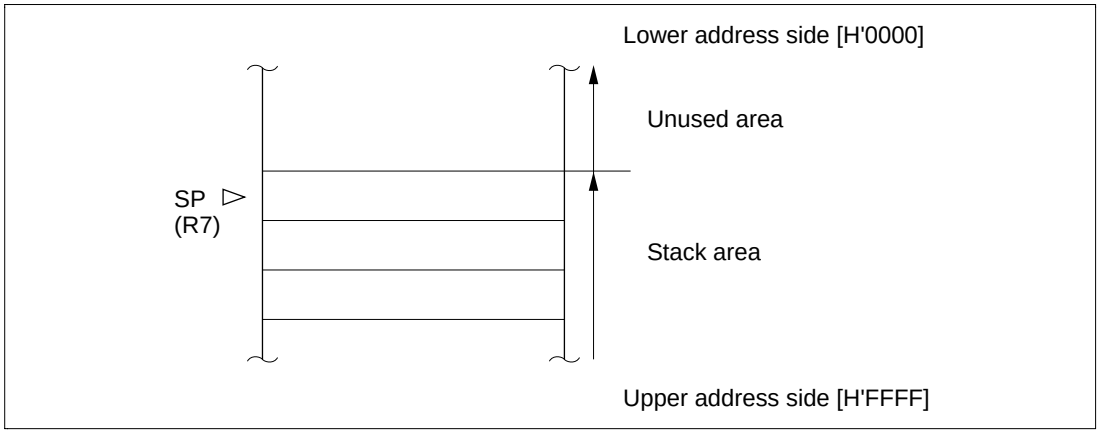


Figure 2.2 Stack Pointer

2.2.2 Control Registers

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

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

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

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

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

Bit 5—Half-Carry Flag (H): When the ADD.B, ADDX.B, SUB.B, SUBX.B, CMP.B, or NEG.B instruction is executed, this flag is set to 1 if there is a carry or borrow at bit 3, and is cleared to 0 otherwise.

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

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

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

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

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

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

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

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

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

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

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

2.2.3 Initial Register Values

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

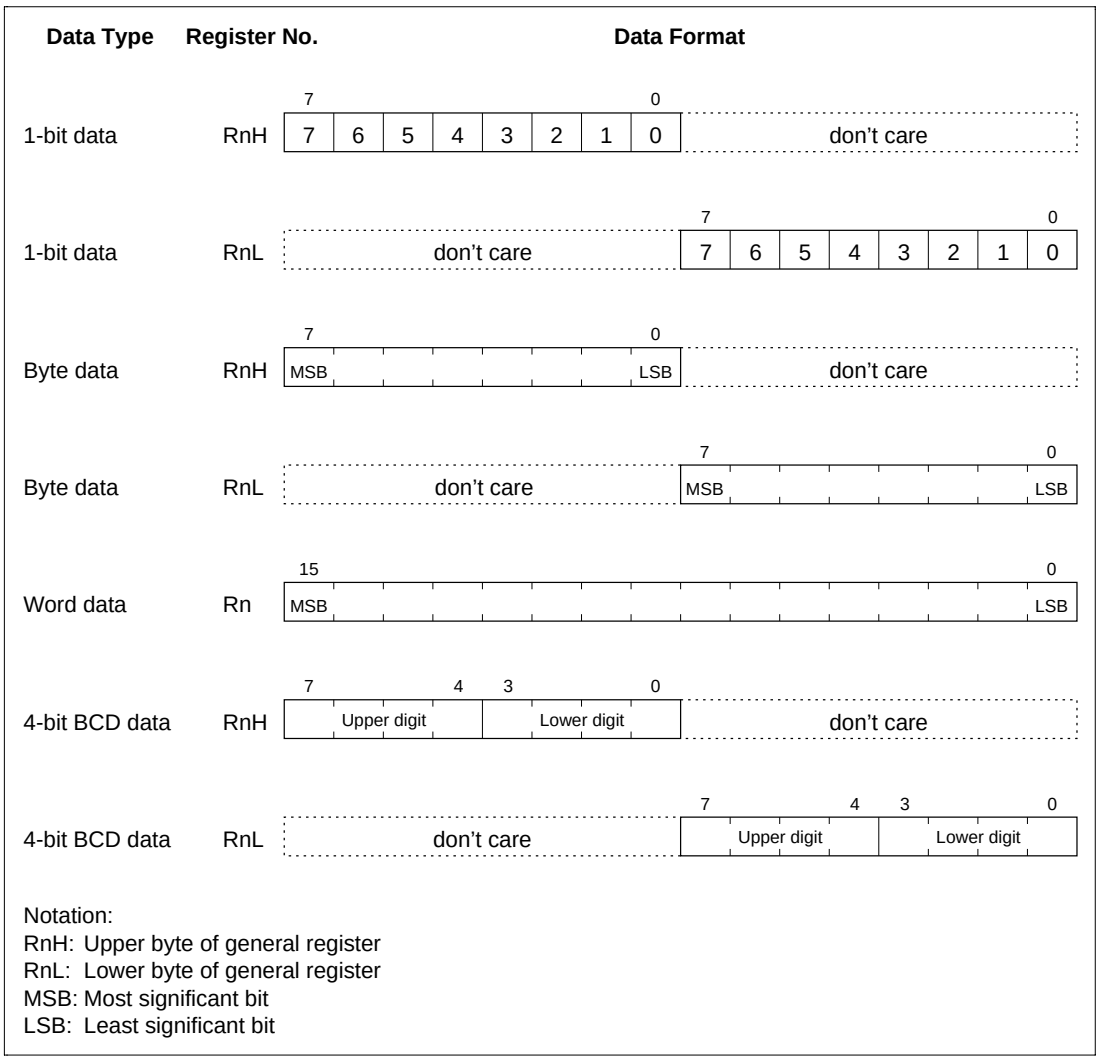
2.3 Data Formats

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

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

2.3.1 Data Formats in General Registers

The general register data formats are shown in figure 2.3.



Notation:
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

2.3.2 Memory Data Formats

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

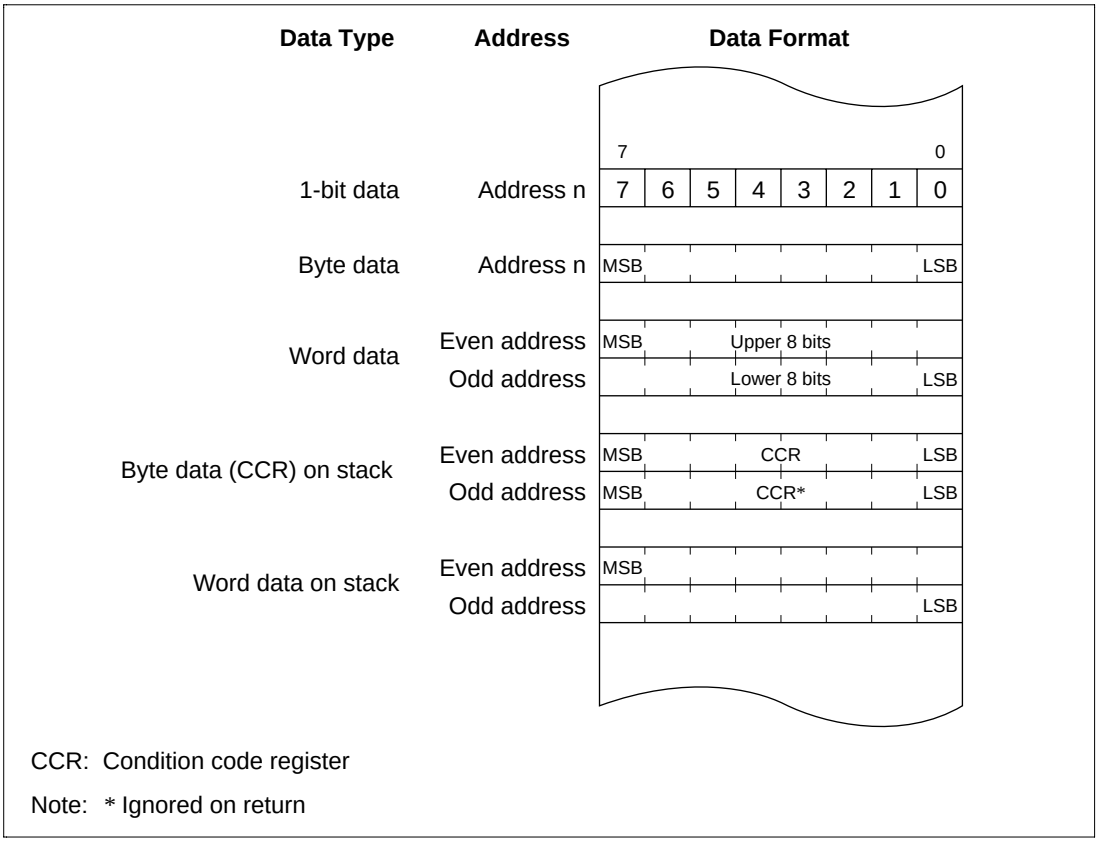


Figure 2.4 Memory Data Formats

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

2.4 Addressing Modes

2.4.1 Addressing Modes

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

Table 2.1 Addressing Modes

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

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

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

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

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

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

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

- Register indirect with post-increment—@Rn+

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

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

- Register indirect with pre-decrement—@–Rn

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

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

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

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

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

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

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

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

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

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

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

2.4.2 Effective Address Calculation

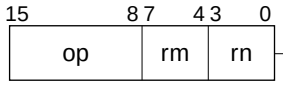
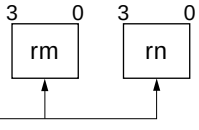
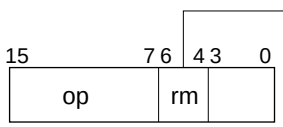
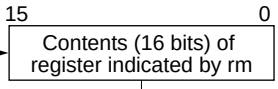
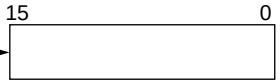
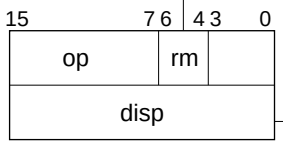
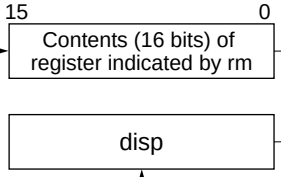
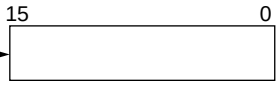
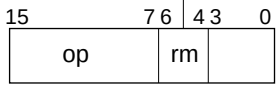
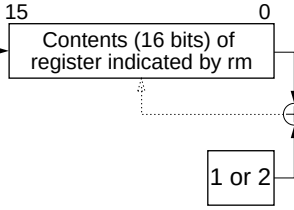
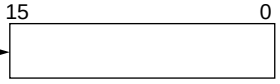
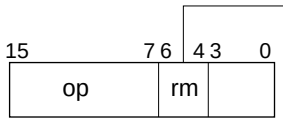
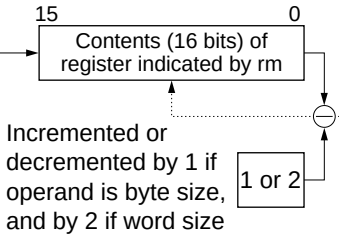
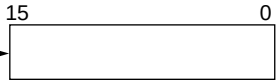
Table 2.2 shows how effective addresses are calculated in each of the addressing modes.

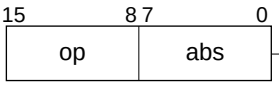
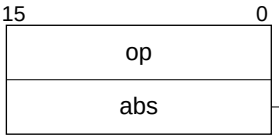
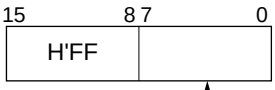
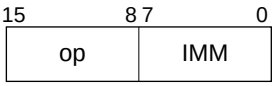
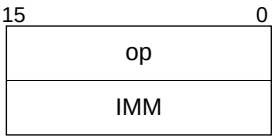
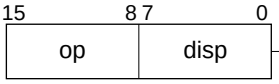
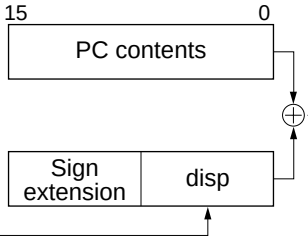
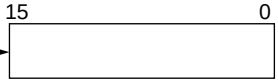
Arithmetic and logic instructions use register direct addressing (1). The ADD.B, ADDX, SUBX, CMP.B, AND, OR, and XOR instructions can also use immediate addressing (6).

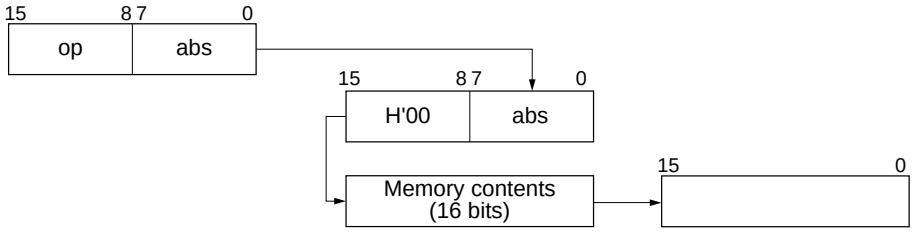
Data transfer instructions can use all addressing modes except program-counter relative (7) and memory indirect (8).

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

Table 2.2 Effective Address Calculation

No.	Addressing Mode and Instruction Format	Effective Address Calculation Method	Effective Address (EA)
1	Register direct, Rn 		Operand is contents of registers indicated by rm/rn
2	Register indirect, @Rn 		
3	Register indirect with displacement, @(d:16, Rn) 		
4	Register indirect with post-increment, @Rn+ 		
	Register indirect with pre-decrement, @–Rn 	 Incremented or decremented by 1 if operand is byte size, and by 2 if word size	

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

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

Notation:

rm, rn: Register field

op: Operation field

disp: Displacement

IMM: Immediate data

abs: Absolute address

2.5 Instruction Set

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

Table 2.3 Instruction Set

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

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

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

Notation

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

2.5.1 Data Transfer Instructions

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

Table 2.4 Data Transfer Instructions

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

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

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

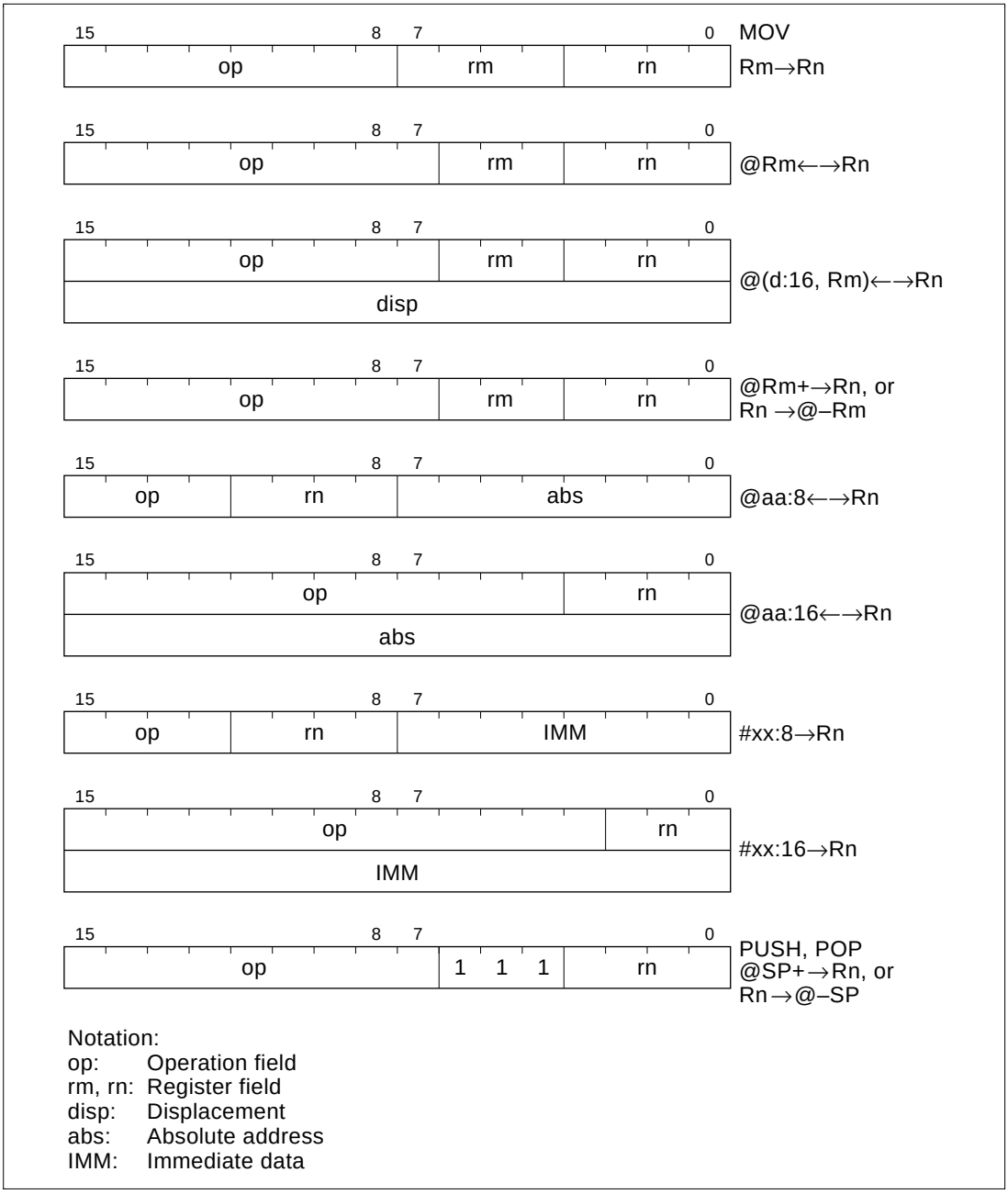


Figure 2.5 Data Transfer Instruction Codes

2.5.2 Arithmetic Operations

Table 2.5 describes the arithmetic instructions.

Table 2.5 Arithmetic Instructions

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

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

2.5.3 Logic Operations

Table 2.6 describes the four instructions that perform logic operations.

Table 2.6 Logic Operation Instructions

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

Note: * Size: Operand size
B: Byte

2.5.4 Shift Operations

Table 2.7 describes the eight shift instructions.

Table 2.7 Shift Instructions

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

Note: * Size: Operand size
B: Byte

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

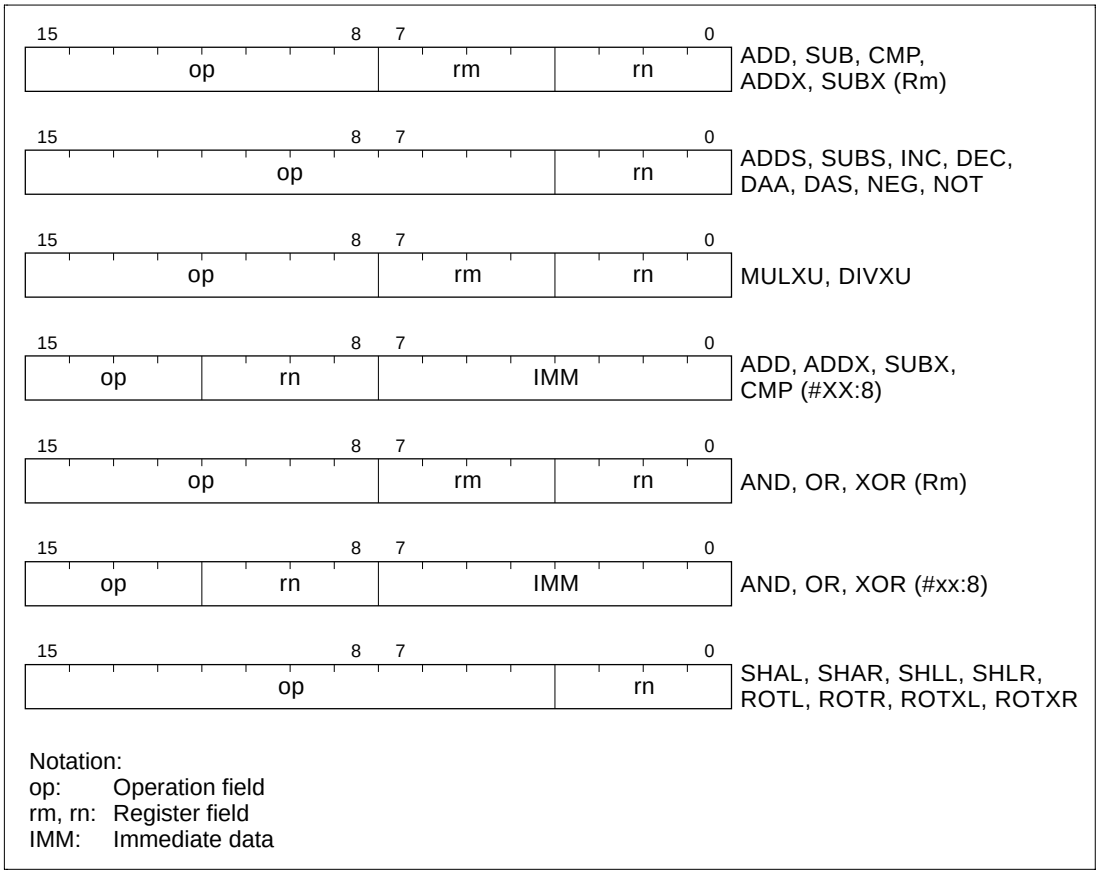


Figure 2.6 Arithmetic, Logic, and Shift Instruction Codes

2.5.5 Bit Manipulations

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

Table 2.8 Bit-Manipulation Instructions

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

Note: * Size: Operand size
B: Byte

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

Note: * Size: Operand size
B: Byte

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

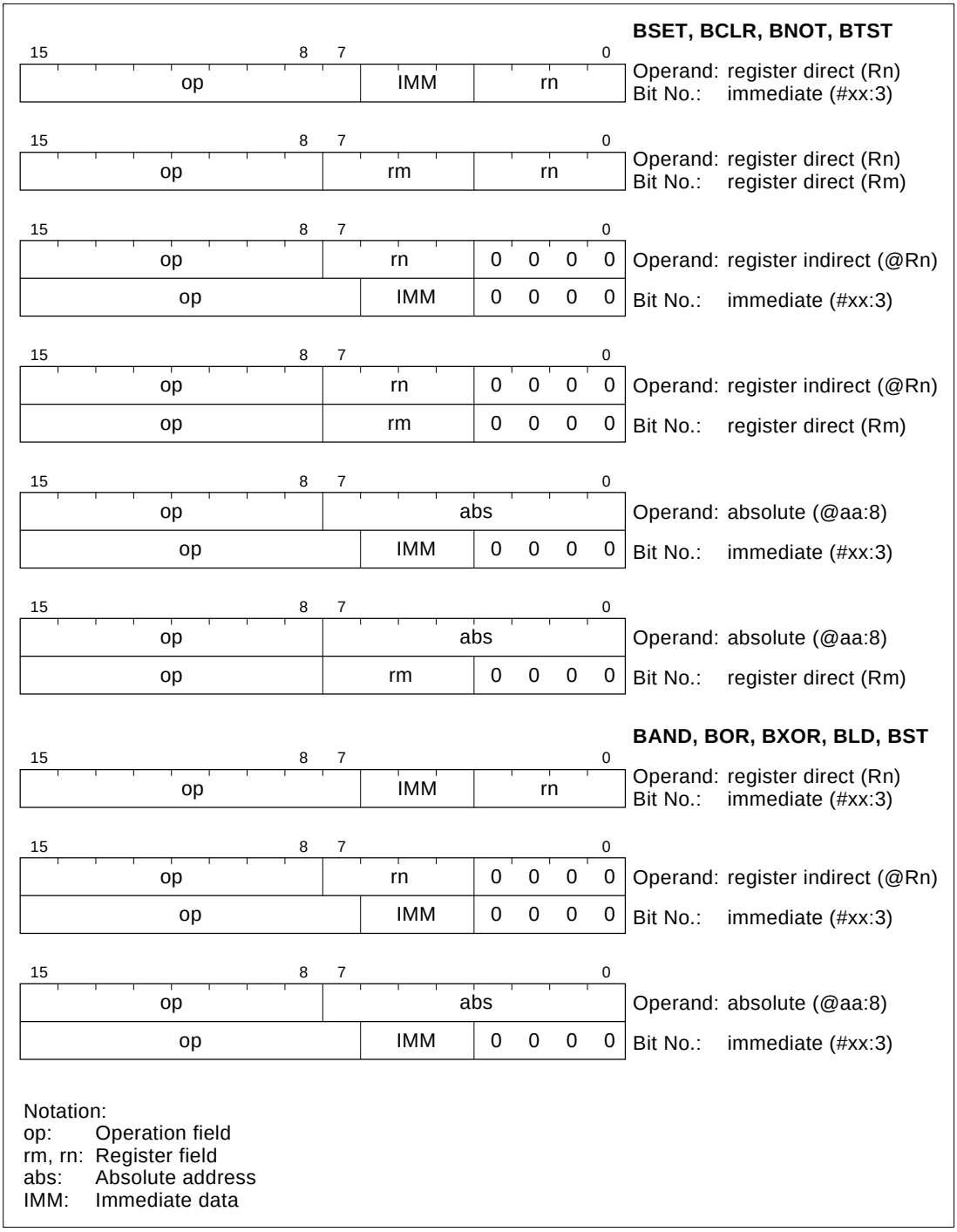


Figure 2.7 Bit Manipulation Instruction Codes

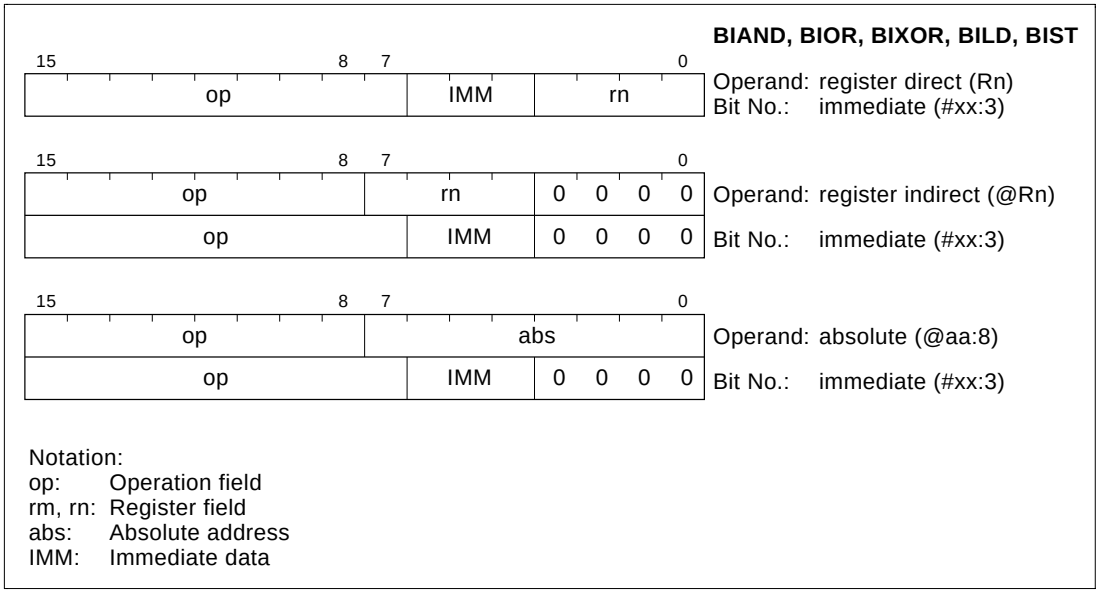


Figure 2.7 Bit Manipulation Instruction Codes (cont)

2.5.6 Branching Instructions

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

Table 2.9 Branching Instructions

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

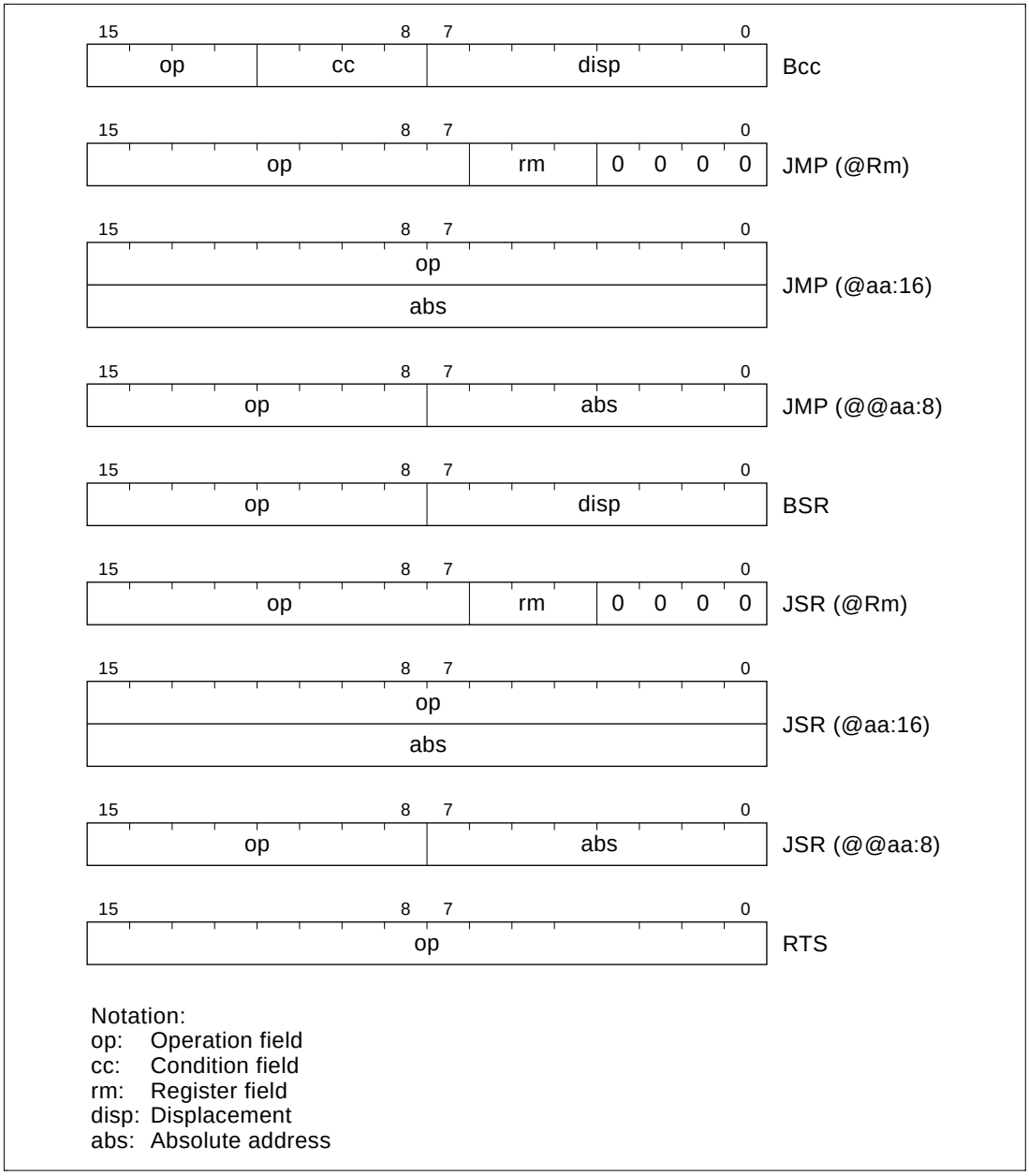


Figure 2.8 Branching Instruction Codes

2.5.7 System Control Instructions

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

Table 2.10 System Control Instructions

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

Note: * Size: Operand size
B: Byte

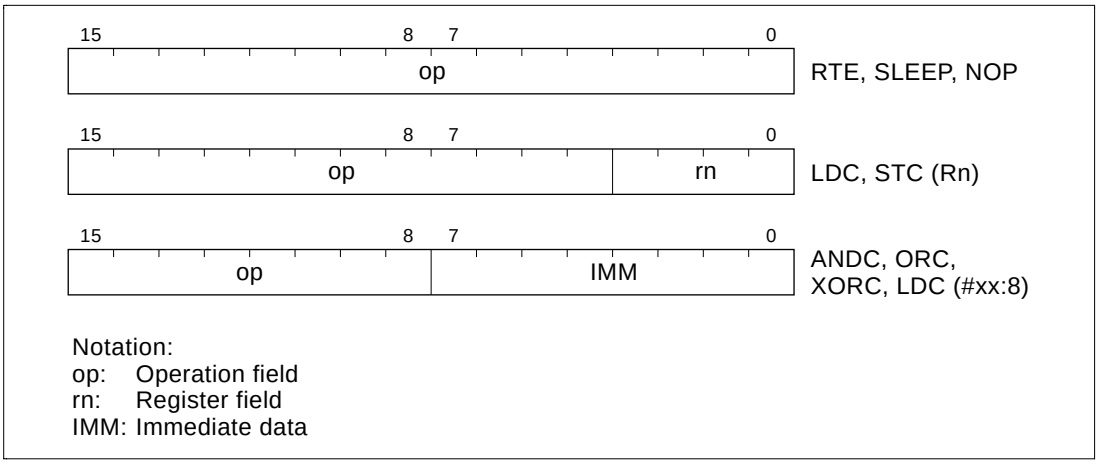


Figure 2.9 System Control Instruction Codes

2.5.8 Block Data Transfer Instruction

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

Table 2.11 Block Data Transfer Instruction

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

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

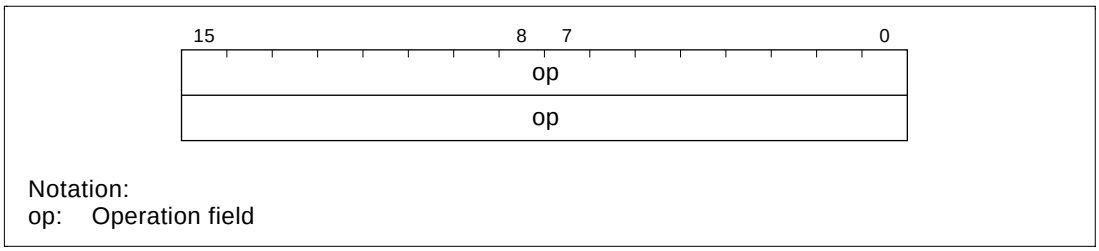


Figure 2.10 Block Data Transfer Instruction Code

2.6 Basic Operational Timing

CPU operation is synchronized by a system clock (ϕ) or a subclock (ϕ_{SUB}). For details on these clock signals see section 4, Clock Pulse Generators. The period from a rising edge of ϕ or ϕ_{SUB} to the next rising edge is called one state. A bus cycle consists of two states or three states. The cycle differs depending on whether access is to on-chip memory or to on-chip peripheral modules.

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

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

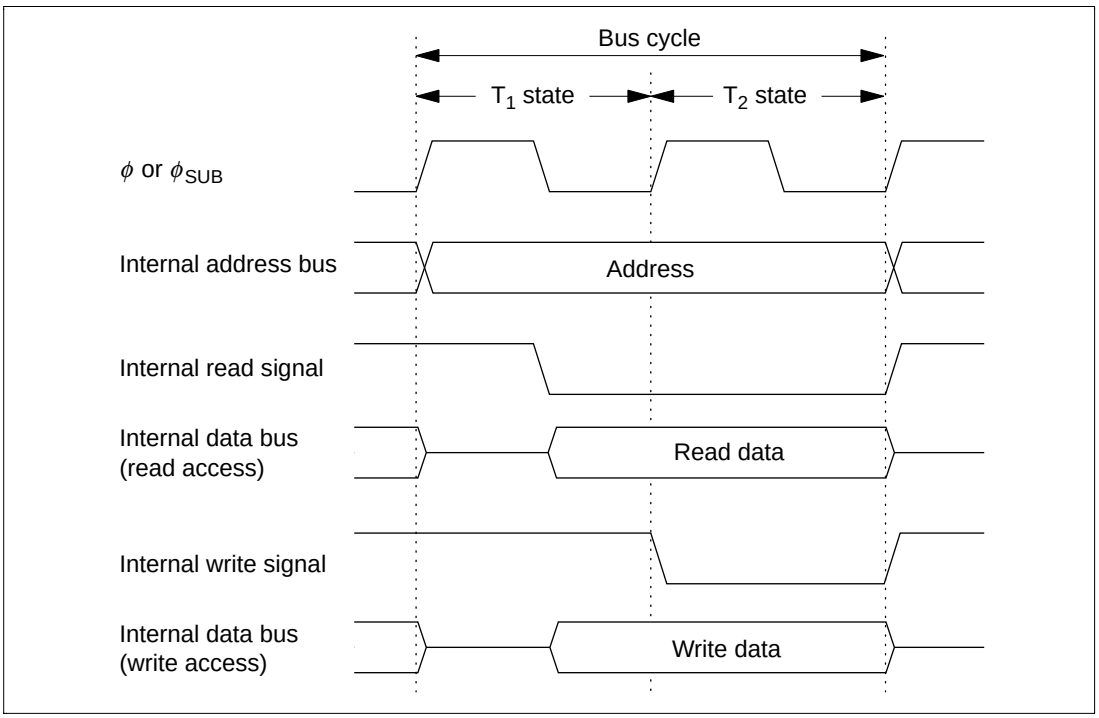


Figure 2.11 On-Chip Memory Access Cycle

2.6.2 Access to On-Chip Peripheral Modules

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

Two-State Access to On-Chip Peripheral Modules

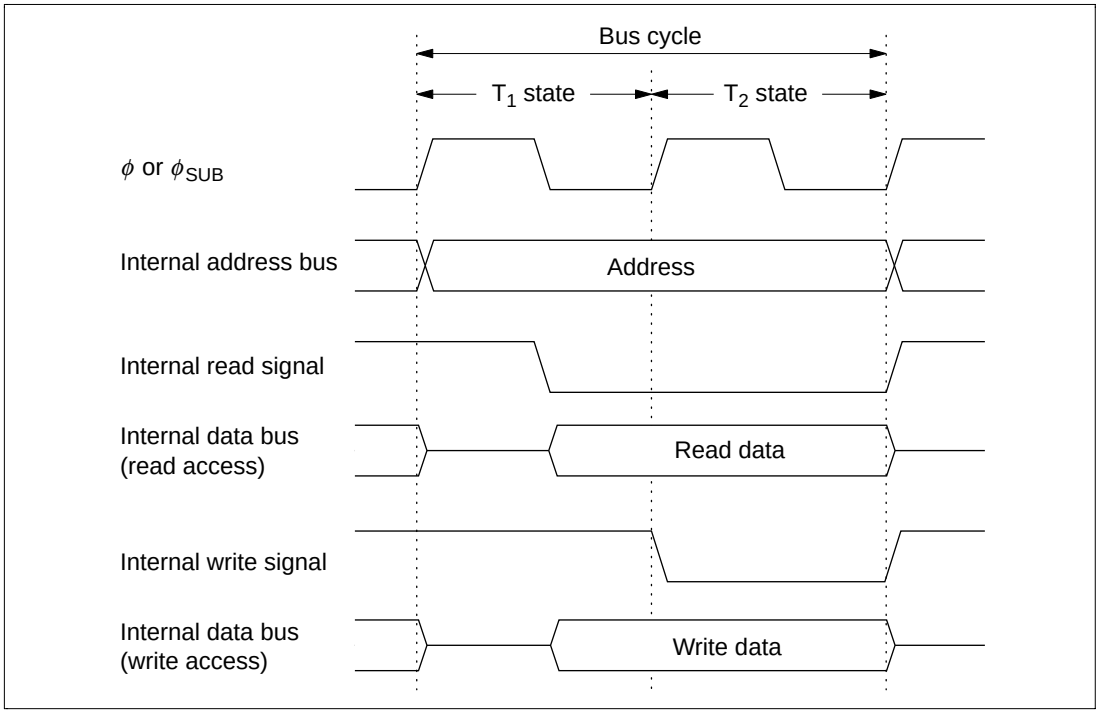


Figure 2.12 On-Chip Peripheral Module Access Cycle (2-State Access)

Three-State Access to On-Chip Peripheral Modules

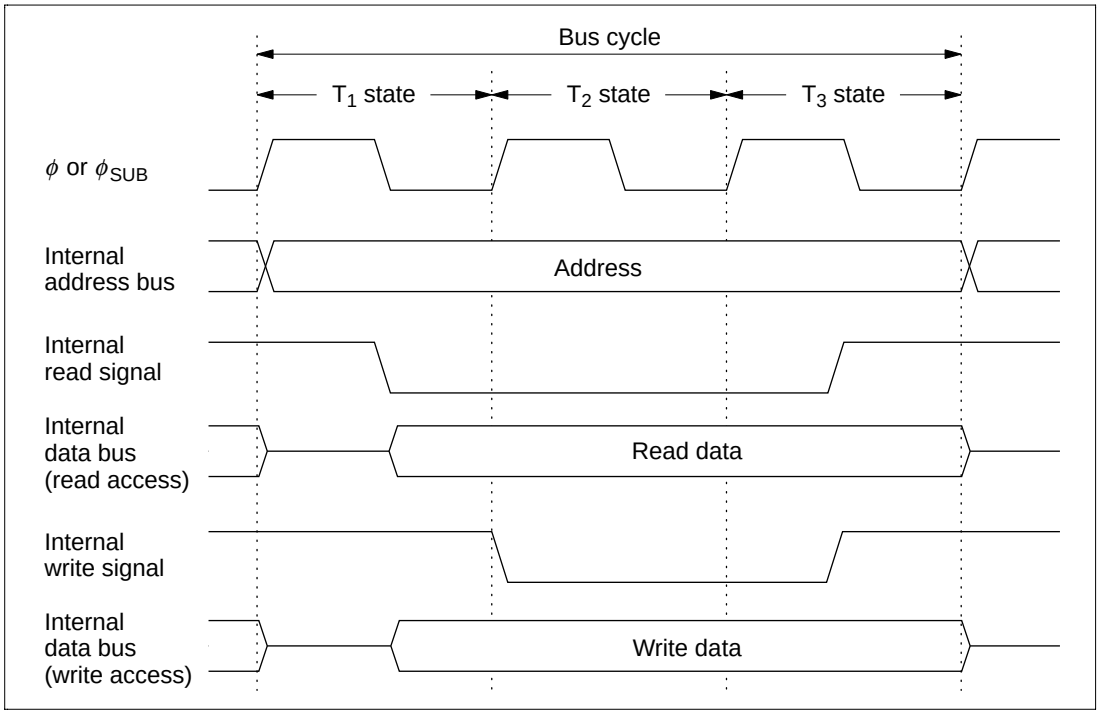


Figure 2.13 On-Chip Peripheral Module Access Cycle (3-State Access)

2.7 CPU States

2.7.1 Overview

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

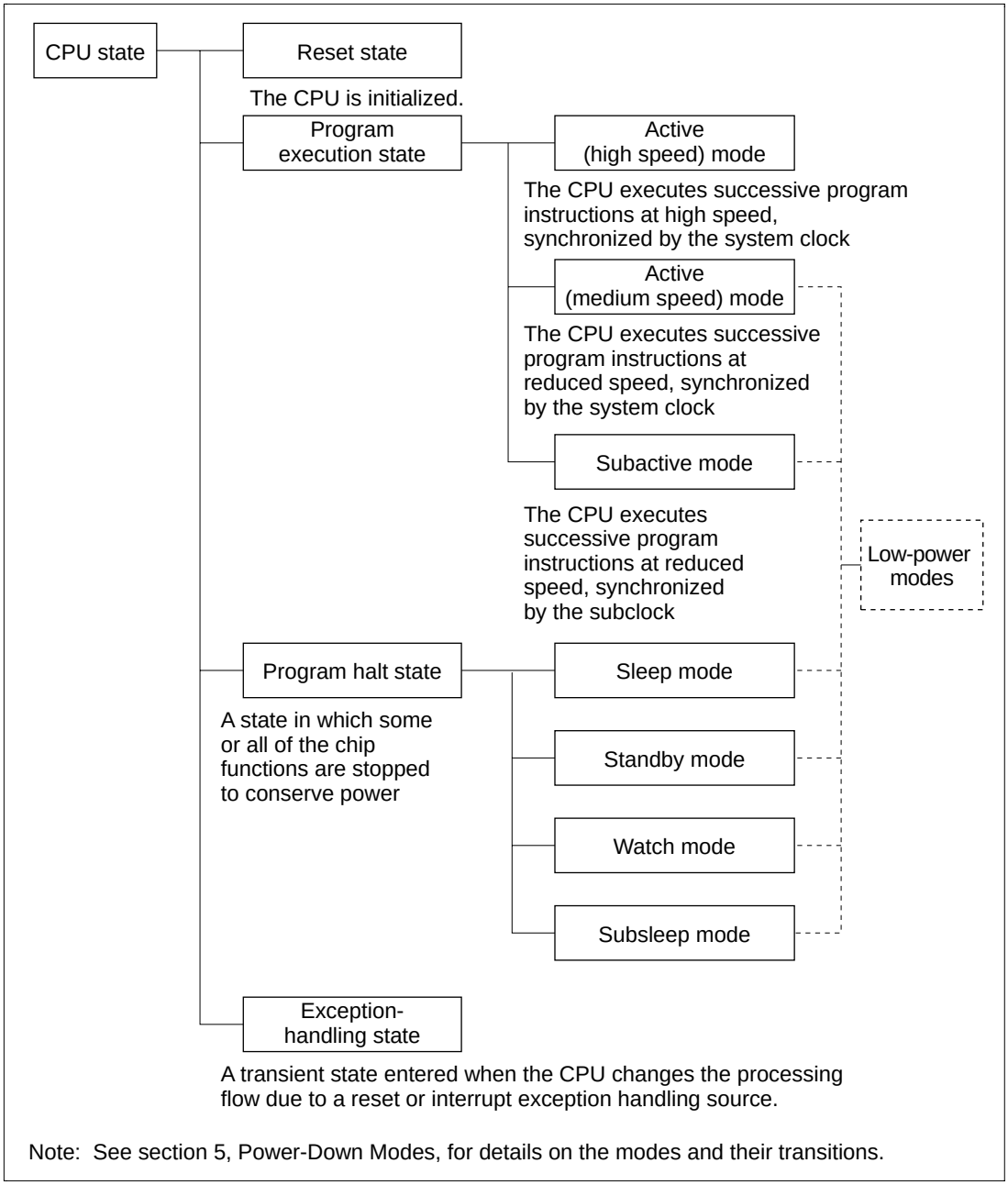


Figure 2.14 CPU Operation States

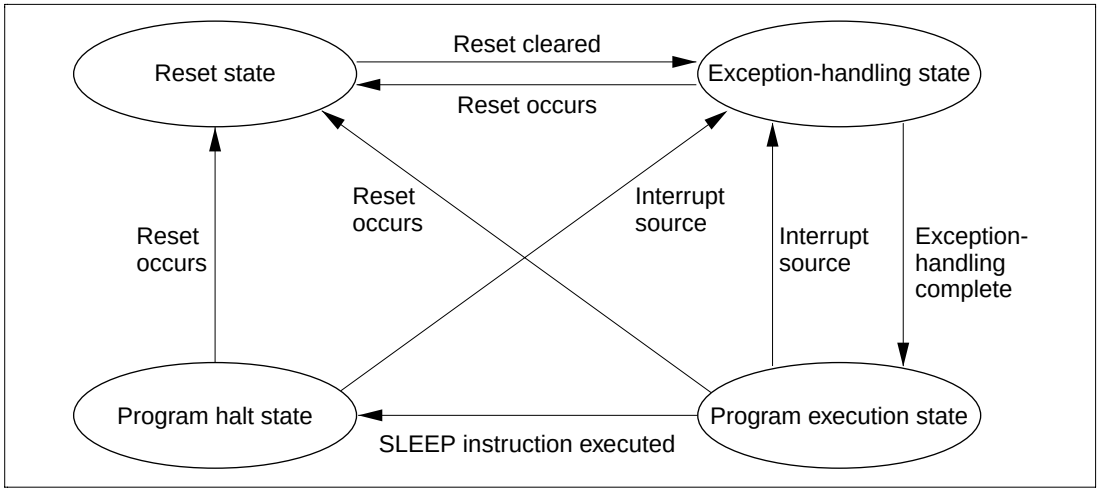


Figure 2.15 State Transitions

2.7.2 Program Execution State

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

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

2.7.3 Program Halt State

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

2.7.4 Exception-Handling State

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

For details on interrupt handling, see section 3.3 Interrupts.

2.8 Memory Map

2.8.1 Memory Map

The memory maps of the H8/3857 Series and H8/3854 Series are shown in figures 2.16 (a) and (b).

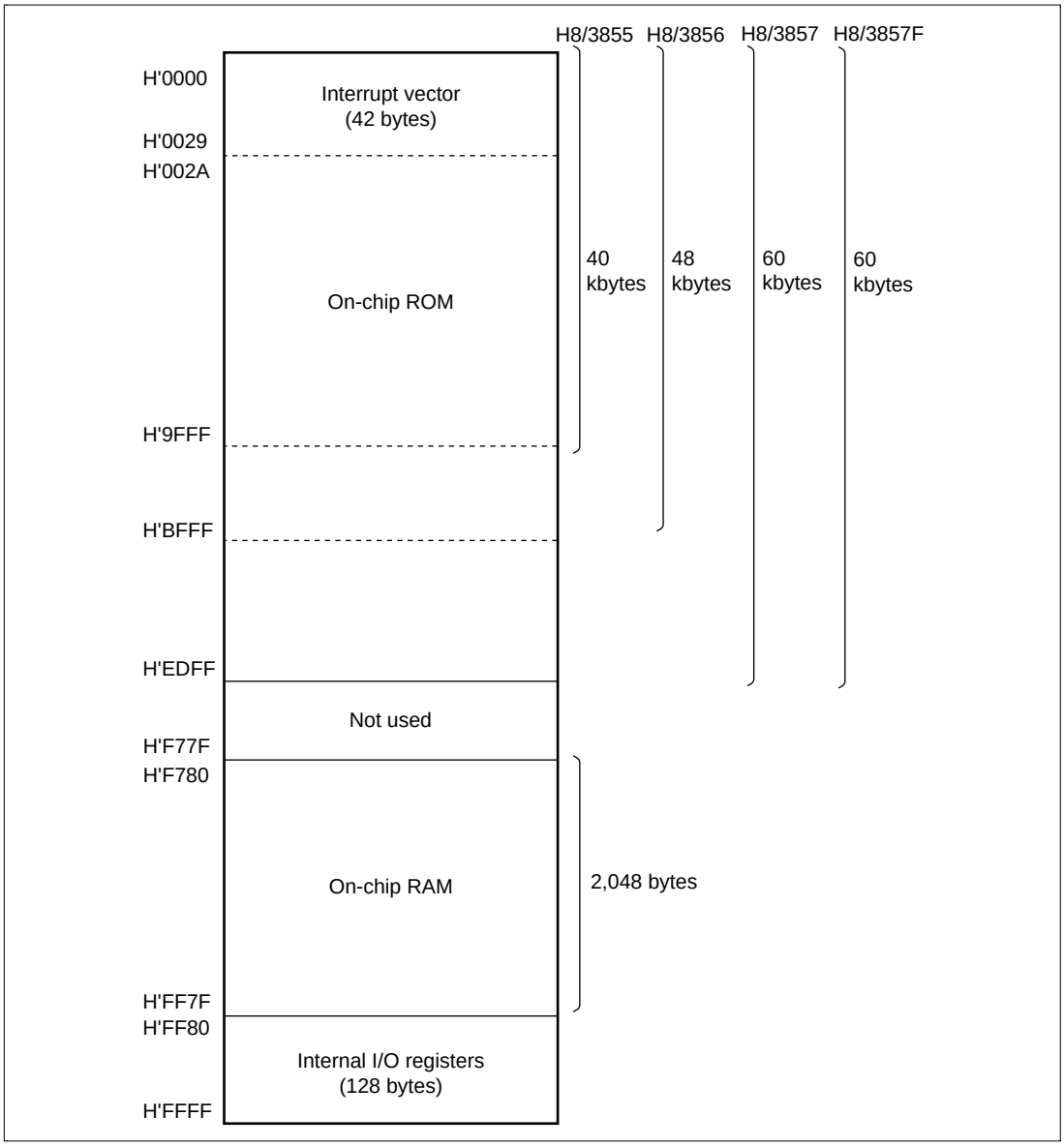
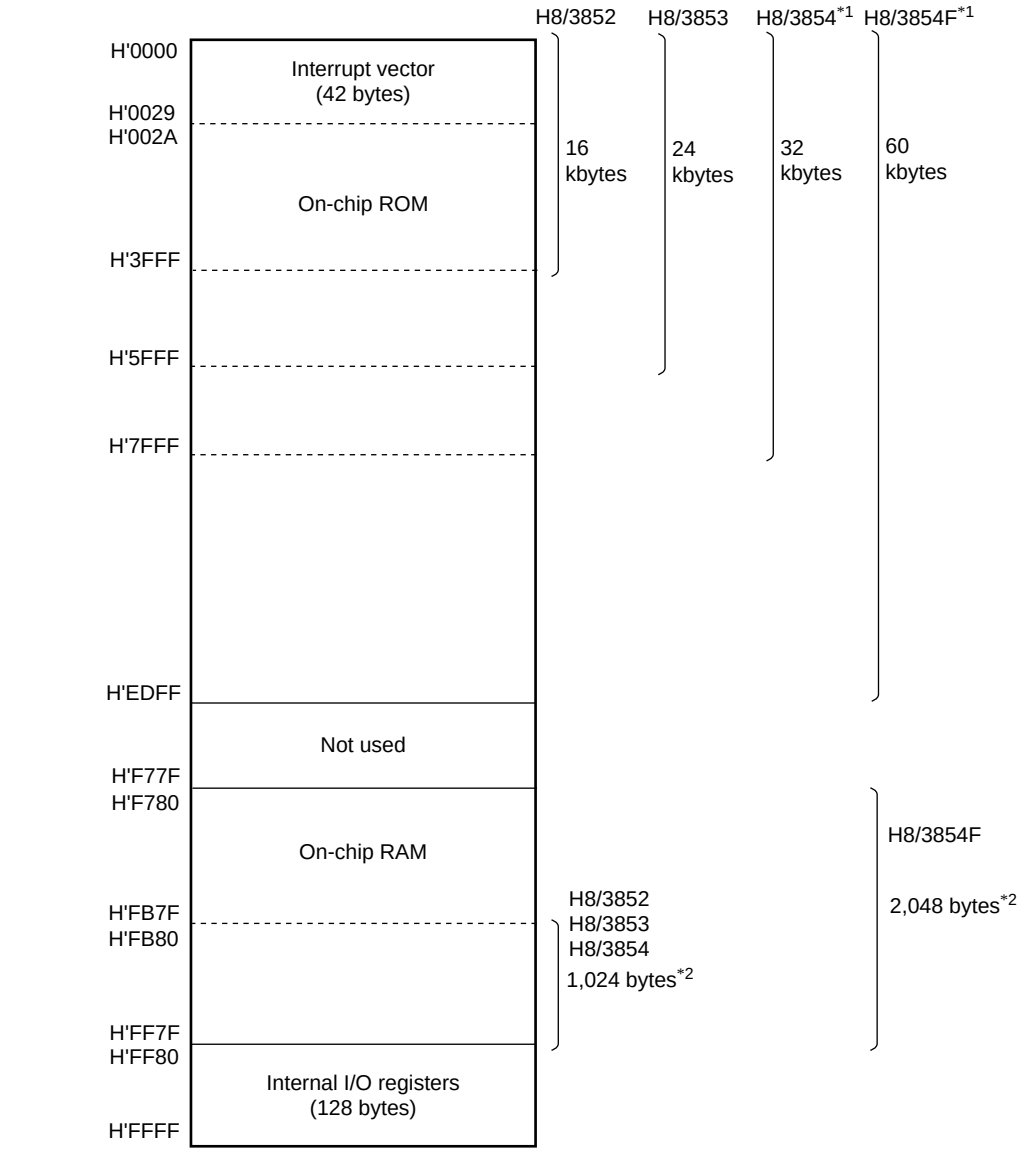


Figure 2.16 (a) H8/3857 Series Memory Map



Notes: 1. Note that the H8/3854 (mask ROM version) and H8/3854F (F-ZTAT version) have different ROM and RAM sizes.
2. The start address for 2-kbyte RAM is H'F780, and the start address for 1-kbyte RAM is H'FB80.

Figure 2.16 (b) H8/3854 Series Memory Map

2.9 Application Notes

2.9.1 Notes on Data Access

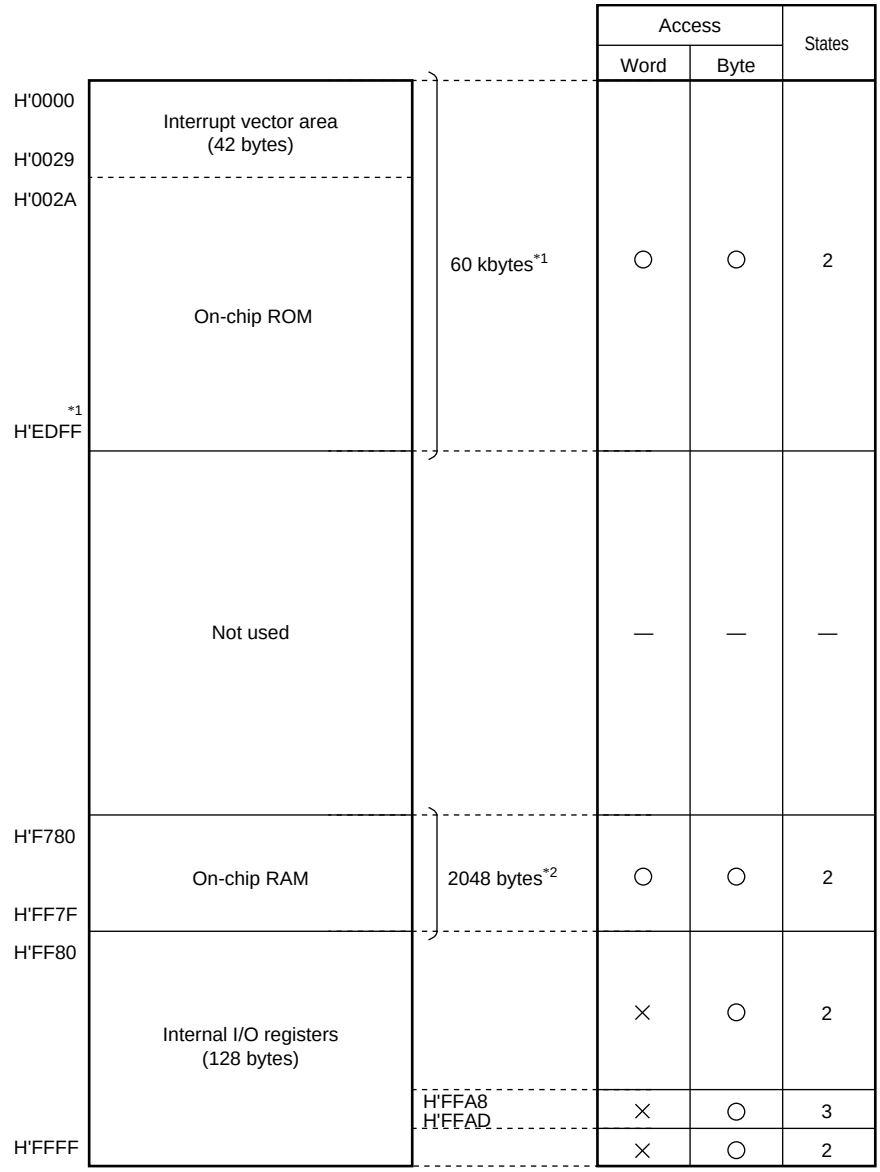
Access to Empty Areas: The address space of the H8/300L CPU includes empty areas in addition to the RAM, registers, and ROM areas available to the user. If these empty areas are mistakenly accessed by an application program, the following results will occur.

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

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

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

Byte size instructions should therefore be used when transferring data to or from I/O registers other than the on-chip ROM and RAM areas. Figure 2.17 shows the data size and number of states in which on-chip peripheral modules can be accessed.



Notes: The above example is a description of the H8/3857, H8/3857F, and H8/3854F.

1. The H8/3855 has 40 kbytes of on-chip ROM, ending at address H'9FFF, the H8/3856 has 48 kbytes, ending at address H'BFFF, the H8/3852 has 16 kbytes, ending at address H'3FFF, the H8/3853 has 24 kbytes, ending at address H'5FFF, and the H8/3854 (mask ROM version) has 32 kbytes, ending at address H'7FFF.
2. The H8/3857 Series and the H8/3854F have 2,048 bytes of on-chip RAM, and the H8/3854 Series (mask ROM version) has 1,024 bytes, starting at address H'FB80.

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

2.9.2 Notes on Bit Manipulation

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

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

Bit Manipulation in Two Registers Assigned to the Same Address

Example 1: Timer load register and timer count bit manipulation

Figure 2.18 shows an example in which two timer registers share the same address. When a bit manipulation instruction accesses the timer load register and timer counter of a reloadable timer, since these two registers share the same address, the following operations take place.

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

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

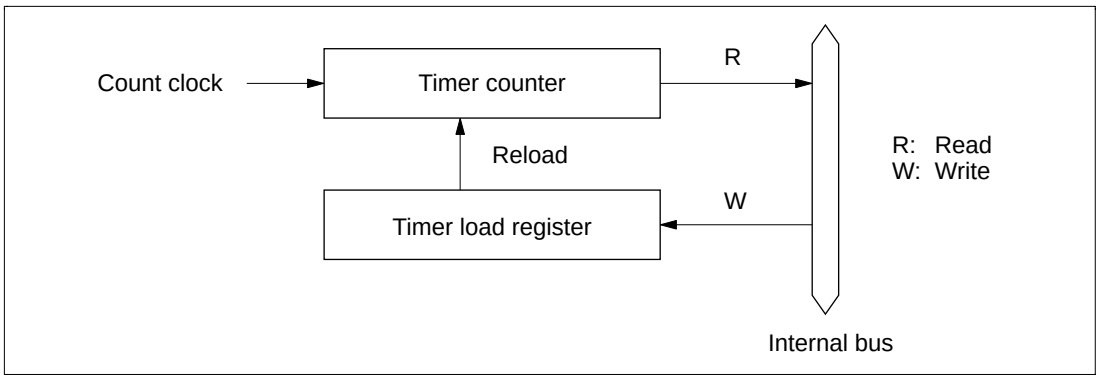


Figure 2.18 Timer Configuration Example

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

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

[A: Prior to executing BSET]

	$P3_7$	**$P3_6$**	**$P3_5$**	**$P3_4$**	**$P3_3$**	**$P3_2$**	**$P3_1$**	**$P3_0$**
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	Low level
PCR3	0	0	1	1	1	1	1	1
PDR3	1	0	0	0	0	0	0	0

[B: BSET instruction executed]

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

The BSET instruction is executed designating port 3.

[C: After executing BSET]

	$P3_7$	**$P3_6$**	**$P3_5$**	**$P3_4$**	**$P3_3$**	**$P3_2$**	**$P3_1$**	**$P3_0$**
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	High level
PCR3	0	0	1	1	1	1	1	1
PDR3	0	1	0	0	0	0	0	1

[D: Explanation of how BSET operates]

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

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

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

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

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

[A: Prior to executing BSET]

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

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

	$P3_7$	**$P3_6$**	**$P3_5$**	**$P3_4$**	**$P3_3$**	**$P3_2$**	**$P3_1$**	**$P3_0$**
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	Low level
PCR3	0	0	1	1	1	1	1	1
PDR3	1	0	0	0	0	0	0	0
RAM0	1	0	0	0	0	0	0	0

[B: BSET instruction executed]

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

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

[C: After executing BSET]

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

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

	$P3_7$	**$P3_6$**	**$P3_5$**	**$P3_4$**	**$P3_3$**	**$P3_2$**	**$P3_1$**	**$P3_0$**
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	High level
PCR3	0	0	1	1	1	1	1	1
PDR3	1	0	0	0	0	0	0	1
RAM0	1	0	0	0	0	0	0	1

Bit Manipulation in a Register Containing a Write-Only Bit

Example 3: When a BCLR instruction is executed on PCR3 of port 3

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

[A: Prior to executing BCLR]

	$P3_7$	**$P3_6$**	**$P3_5$**	**$P3_4$**	**$P3_3$**	**$P3_2$**	**$P3_1$**	**$P3_0$**
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	Low level
PCR3	0	0	1	1	1	1	1	1
PDR3	1	0	0	0	0	0	0	0

[B: BCLR instruction executed]

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

The BCLR instruction is executed designating PCR3.

[C: After executing BCLR]

	$P3_7$	**$P3_6$**	**$P3_5$**	**$P3_4$**	**$P3_3$**	**$P3_2$**	**$P3_1$**	**$P3_0$**
Input/output	Output	Output	Output	Output	Output	Output	Output	Input
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	High level
PCR3	1	1	1	1	1	1	1	0
PDR3	1	0	0	0	0	0	0	0

[D: Explanation of how BCLR operates]

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

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

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

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

[A: Prior to executing BCLR]

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

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

	$P3_7$	**$P3_6$**	**$P3_5$**	**$P3_4$**	**$P3_3$**	**$P3_2$**	**$P3_1$**	**$P3_0$**
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	Low level
PCR3	0	0	1	1	1	1	1	1
PDR3	1	0	0	0	0	0	0	0
RAM0	0	0	1	1	1	1	1	1

[B: BCLR instruction executed]

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

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

[C: After executing BCLR]

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

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

	$P3_7$	**$P3_6$**	**$P3_5$**	**$P3_4$**	**$P3_3$**	**$P3_2$**	**$P3_1$**	**$P3_0$**
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	High level
PCR3	0	0	1	1	1	1	1	0
PDR3	1	0	0	0	0	0	0	0
RAM0	0	0	1	1	1	1	1	0

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

Table 2.12 Registers with shared addresses

Register Name	Abbreviation	Address
Timer counter B and timer load register B	TCB/TLB	H'FFB3
Timer counter C and timer load register C*[2]	TCC/TLC	H'FFB5
Port data register 1*[1,3]	PDR1	H'FFD4
Port data register 2*[1]	PDR2	H'FFD5
Port data register 3*[1,2]	PDR3	H'FFD6
Port data register 4*[1]	PDR4	H'FFD7
Port data register 5*[1]	PDR5	H'FFD8
Port data register 9*[1]	PDR9	H'FFDC
Port data register A*[1]	PDRA	H'FFDD

Notes: 1. These port registers are used also for pin input.
2. A function of the H8/3857 Series only; not provided in the H8/3854 Series.
3. Some bits are not present in the H8/3854 Series.

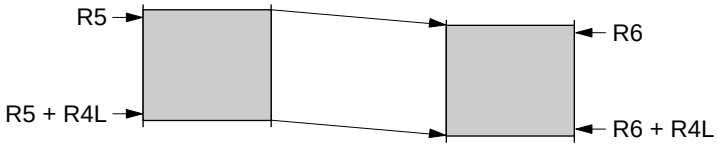
Table 2.13 Registers with write-only bits

Register Name	Abbreviation	Address
Port control register 1*[1]	PCR1	H'FFE4
Port control register 2	PCR2	H'FFE5
Port control register 3*[2]	PCR3	H'FFE6
Port control register 4	PCR4	H'FFE7
Port control register 5	PCR5	H'FFE8
Port control register 9	PCR9	H'FFEC
Port control register A	PCRA	H'FFED
Timer control register F	TCRF	H'FFB6
PWM control register*[2]	PWCR	H'FFD0
PWM data register U*[2]	PWDRU	H'FFD1
PWM data register L*[2]	PWDRL	H'FFD2

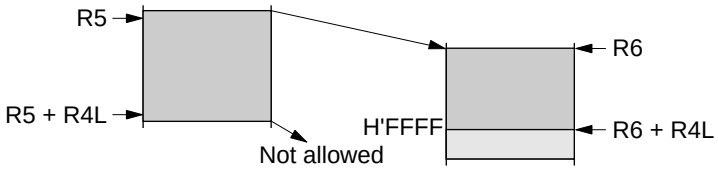
Notes: 1. Some bits are not present in the H8/3854 Series.
2. A function of the H8/3857 Series only; not provided in the H8/3854 Series.

2.9.3 Notes on Use of the EEPMOV Instruction

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



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



Section 3 Exception Handling

3.1 Overview

Exception handling is performed in the H8/3857 Series when a reset or interrupt occurs. Table 3.1 shows the priorities of these two types of exception handling.

Table 3.1 Exception Handling Types and Priorities

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

3.2 Reset

3.2.1 Overview

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

3.2.2 Reset Sequence

As soon as the $\overline{\text{RES}}$ pin goes low, all processing is stopped and the H8/3857 enters the reset state.

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

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

When the $\overline{\text{RES}}$ pin goes high again after being held low for a given period, reset exception handling begins. Reset exception handling takes place as follows:

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

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

Figure 3.1 shows the reset sequence.

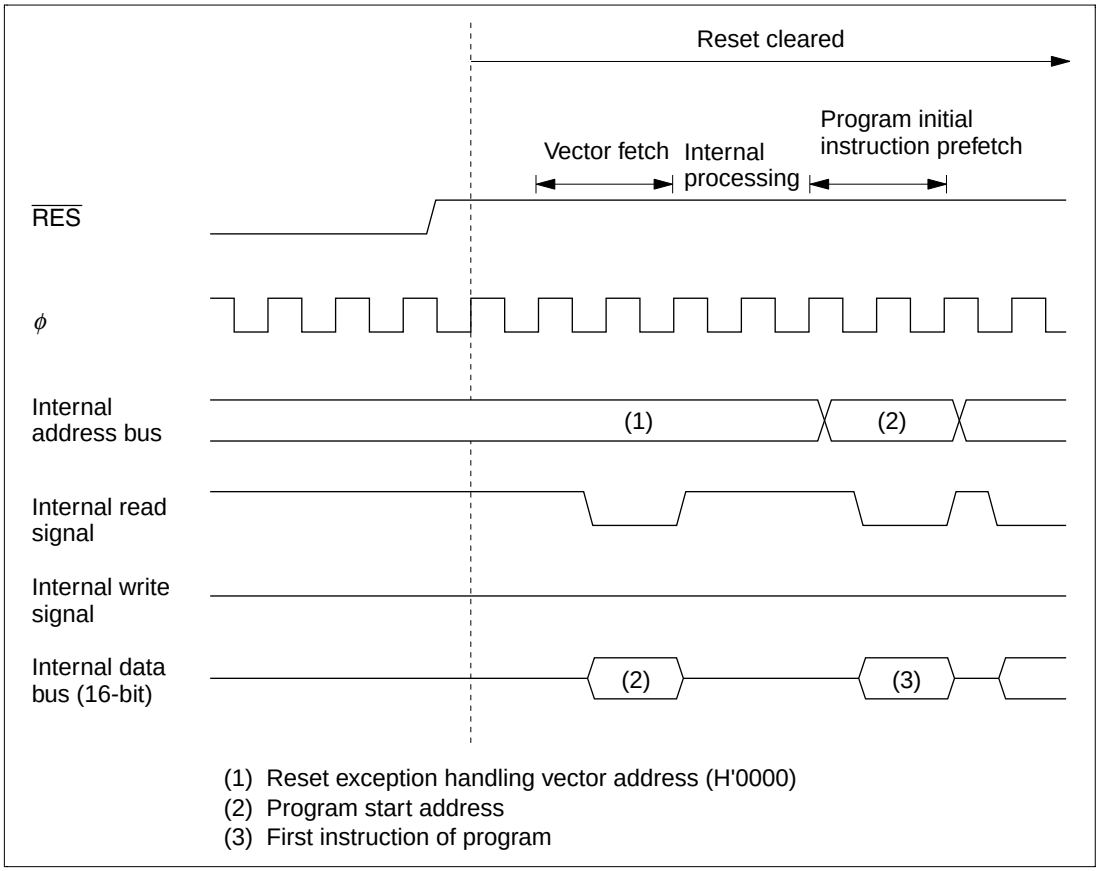


Figure 3.1 Reset Sequence

3.2.3 Interrupt Immediately after Reset

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

3.3 Interrupts

3.3.1 Overview

In the H8/3857 Series, sources that initiate interrupt exception handling include 13 external interrupts (WKP_7 to WKP_0, and IRQ_4 to IRQ_0), and 16 internal interrupts from on-chip peripheral modules. In the H8/3854 Series, sources that initiate interrupt exception handling include 12 external interrupts (WKP_7 to WKP_0, IRQ_4, IRQ_3, IRQ_1, and IRQ_0), and 14 internal interrupts from on-chip peripheral modules. Table 3.2 shows the interrupt sources, their priorities, and their vector addresses. When more than one interrupt is requested, the interrupt with the highest priority is processed.

The interrupts have the following features:

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

Table 3.2 Interrupt Sources and Priorities

Priority	Interrupt Source	Interrupt	Vector Number	Vector Address*[1]
High	$\overline{RES}$	Reset	0	H'0000 to H'0001
	$\overline{IRQ_0}$	IRQ_0	4	H'0008 to H'0009
	$\overline{IRQ_1}$	IRQ_1	5	H'000A to H'000B
	$\overline{IRQ_2}$*[2]	IRQ_2	6	H'000C to H'000D
	$\overline{IRQ_3}$	IRQ_3	7	H'000E to H'000F
	$\overline{IRQ_4}$	IRQ_4	8	H'0010 to H'0011
	$\overline{WKP_0}$	WKP_0	9	H'0012 to H'0013
	$\overline{WKP_1}$	WKP_1		
	$\overline{WKP_2}$	WKP_2		
	$\overline{WKP_3}$	WKP_3		
	$\overline{WKP_4}$	WKP_4		
	$\overline{WKP_5}$	WKP_5		
	$\overline{WKP_6}$	WKP_6		
	$\overline{WKP_7}$	WKP_7		
	SCI1*[2]	SCI1 transfer complete	10	H'0014 to H'0015
	Timer A	Timer A overflow	11	H'0016 to H'0017
	Timer B	Timer B overflow	12	H'0018 to H'0019
	Timer C*[2]	Timer C overflow or underflow	13	H'001A to H'001B
	Timer FL	Timer FL compare match	14	H'001C to H'001D
		Timer FL overflow		
	Timer FH	Timer FH compare match	15	H'001E to H'001F
		Timer FH overflow		
	SCI3	SCI3 transmit end	18	H'0024 to H'0025
		SCI3 transmit data empty		
		SCI3 receive data full		
		SCI3 overrun error		
		SCI3 framing error		
		SCI3 parity error		
	A/D converter	A/D conversion end	19	H'0026 to H'0027
Low	(SLEEP instruction executed)	Direct transfer	20	H'0028 to H'0029

Notes: 1. Vector addresses H'0002 to H'0007 and H'0020 to H'0023 are reserved and cannot be used.

2. Applies to the H8/3857 Series. In the H8/3854 Series, these vector addresses are reserved.

3.3.2 Interrupt Control Registers

Table 3.3 lists the registers that control interrupts.

Table 3.3 Interrupt Control Registers

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

Notes: 1. Write is enabled only for writing of 0 to clear a flag.
2. There are some differences in functions between the H8/3857 Series and the H8/3854 Series. For details, see the individual register descriptions.

IRQ Edge Select Register (IEGR)

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

Note: * Applies to the H8/3857 Series. In the H8/3854 Series, this bit must always be cleared to 0.

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

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

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

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

Bit 3—$\overline{IRQ}_3$ Edge Select (IEG3): Bit 3 selects the input sensing of pin $\overline{IRQ}_3$/TMIF.

Bit 3: IEG3	Description	
0	Falling edge of $\overline{IRQ}_3$/TMIF pin input is detected	(initial value)
1	Rising edge of $\overline{IRQ}_3$/TMIF pin input is detected	

Bit 2—$\overline{IRQ}_2$ Edge Select (IEG2): Bit 2 is used in the H8/3857 Series to select the input sensing of pin $\overline{IRQ}_2$/TMIC. In the H8/3854 Series, this bit must always be cleared to 0.

Bit 2: IEG2	Description	
0	Falling edge of $\overline{IRQ}_2$/TMIC pin input is detected	(initial value)
1	Rising edge of $\overline{IRQ}_2$/TMIC pin input is detected	

Bit 1—$\overline{IRQ}_1$ Edge Select (IEG1): Bit 1 selects the input sensing of pin $\overline{IRQ}_1$/TMIB.

Bit 1: IEG1	Description	
0	Falling edge of $\overline{IRQ}_1$/TMIB pin input is detected	(initial value)
1	Rising edge of $\overline{IRQ}_1$/TMIB pin input is detected	

Bit 0—$\overline{IRQ}_0$ Edge Select (IEG0): Bit 0 selects the input sensing of pin $\overline{IRQ}_0$.

Bit 0: IEG0	Description	
0	Falling edge of $\overline{IRQ}_0$ pin input is detected	(initial value)
1	Rising edge of $\overline{IRQ}_0$ pin input is detected	

Interrupt Enable Register 1 (IENR1)

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

Note: * Applies to the H8/3857 Series. In the H8/3854 Series, this bit must always be cleared to 0.

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

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

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

Bit 6—SCI1 Interrupt Enable (IENS1): Bit 6 is used in the H8/3857 Series to enable or disable SCI1 transfer complete interrupt requests. In the H8/3854 Series, this bit must always be cleared to 0.

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

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

Bit 5: IENWP	Description	
0	Disables interrupt requests from $\overline{WKP}_7$ to $\overline{WKP}_0$	(initial value)
1	Enables interrupt requests from $\overline{WKP}_7$ to $\overline{WKP}_0$	

Bits 4, 3, 1, and 0—IRQ_4, IRQ_3, IRQ_1, and IRQ_0 Interrupt Enable (IEN4, IEN3, IEN1, IEN0): Bits 4 to 0 enable or disable $\overline{IRQ}_4$, $\overline{IRQ}_3$, $\overline{IRQ}_1$, and $\overline{IRQ}_0$ interrupt requests.

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

(n = 4, 3, 1, or 0)

Bit 2—$\overline{IRQ}_2$ Interrupt Enable (IEN2): Bit 2 is used in the H8/3857 Series to enable or disable $\overline{IRQ}_2$ interrupt requests. In the H8/3854 Series, this bit must always be cleared to 0.

Bit 2: IEN2	Description	
0	Disables interrupt request $\overline{IRQ}_2$	(initial value)
1	Enables interrupt request $\overline{IRQ}_2$	

Interrupt Enable Register 2 (IENR2)

Bit	7	6	5	4	3	2	1	0
	IENDT	IENAD	—	—	IENTFH	IENTFL	IENTC*	IENTB
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: * Applies to the H8/3857 Series. In the H8/3854 Series, this bit must always be cleared to 0.

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

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

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

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

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

Bits 5 and 4—Reserved Bits: Bits 5 and 4 are reserved; they should always be cleared to 0.

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

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

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

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

Bit 1—Timer C Interrupt Enable (IENTC): Bit 1 is used in the H8/3857 Series to enable or disable timer C overflow or underflow interrupt requests. In the H8/3854 Series, this bit must always be cleared to 0.

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

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

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

SCI3 interrupt control is covered in 10.4.2, in the description of serial control register 3 (SCR3).

Interrupt request register 1 (IRR1)

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

Notes: 1. Only a write of 0 for flag clearing is possible.
2. Applies to the H8/3857 Series. In the H8/3854 Series, this bit must always be cleared to 0.

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

Bit 7—Timer A Interrupt Request Flag (IRRTA)

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

Bit 6—SCI1 Interrupt Request Flag (IRRS1): Bit 6 is used in the H8/3857 Series. In the H8/3854 Series, this bit must always be cleared to 0.

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

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

Bits 4, 3, 1, and 0—IRQ_4, IRQ_3, IRQ_1, and IRQ_0 Interrupt Request Flags (IRRI4, IRRI3, IRRI1, IRRI0)

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

(n = 4, 3, 1, or 0)

Bit 2—IRQ_2 Interrupt Request Flag (IRRI2): Bit 2 is used in the H8/3857 Series. In the H8/3854 Series, this bit must always be cleared to 0.

Bit 2: IRRI2	Description
0	Clearing conditions: When IRRI2 = 1, it is cleared by write 0 to IRRI2 (initial value)
1	Setting conditions: IRRI2 is set when pin $\overline{IRQ}_2$ is set to interrupt input, and the designated signal edge is detected

Interrupt Request Register 2 (IRR2)

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

Notes: 1. Only a write of 0 for flag clearing is possible.
2. Applies to the H8/3857 Series. In the H8/3854 Series, this bit must always be cleared to 0.

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

Bit 7—Direct Transfer Interrupt Request Flag (IRRDT)

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

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

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

Bits 5 and 4—Reserved Bits: Bits 5 and 4 are reserved; they should always be cleared to 0.

Bit 3—Timer FH Interrupt Request Flag (IRRTFH)

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

Bit 2—Timer FL Interrupt Request Flag (IRRTFL)

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

Bit 1—Timer C Interrupt Request Flag (IRRTC): Bit 1 is used in the H8/3857 Series. In the H8/3854 Series, this bit must always be cleared to 0.

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

Bit 0—Timer B Interrupt Request Flag (IRRTB)

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

Wakeup Interrupt Request Register (IWPR)

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

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

IWPR is an 8-bit read/write register, in which the corresponding bit is set to 1 when pins $\overline{WKP}_7$ to $\overline{WKP}_0$ are set to wakeup input and a pin receives a falling edge input. The flags are not cleared automatically when an interrupt is accepted. It is necessary to write 0 to clear each flag.

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

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

(n = 7 to 0)

3.3.3 External Interrupts

The H8/3857 Series has 13 external interrupt sources, WKP_7 to WKP_0, and IRQ_4 to IRQ_0. The H8/3854 Series has 12 external interrupt sources, WKP_7 to WKP_0, IRQ_4, IRQ_3, IRQ_1, and IRQ_0.

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

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

Interrupts IRQ_0 to IRQ_4: Interrupts IRQ_0 to IRQ_4 are requested by into pins inputs to $\overline{IRQ_0}$ to $\overline{IRQ_4}$. These interrupts are detected by either rising edge sensing or falling edge sensing, depending on the settings of bits IEG0 to IEG4 in the edge select register (IEGR). The IRQ_2 interrupt is a function of the H8/3857 Series only, and is not provided in the H8/3854 Series.

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

When IRQ_0 to IRQ_4 interrupt exception handling is initiated, the I bit is set to 1. Vector numbers 4 to 8 are assigned to interrupts IRQ_0 to IRQ_4. The order of priority is from IRQ_0 (high) to IRQ_4 (low). Table 3.2 gives details. In the H8/3854 Series, exception handling vector number 6 is reserved.

3.3.4 Internal Interrupts

There are 16 internal interrupts that can be requested by the on-chip peripheral modules in the H8/3857 Series, and 14 in the H8/3854 Series. When a peripheral module requests an interrupt, the corresponding bit in IRR1 or IRR2 is set to 1. Individual interrupt requests can be disabled by clearing the corresponding bit in IENR1 or IENR2 to 0. All interrupts can be masked by setting the I bit in CCR to 1. When an internal interrupt request is accepted, the I bit is set to 1. Vector numbers 10 to 20 are assigned to these interrupts. Table 3.2 shows the order of priority of interrupts from on-chip peripheral modules. In the H8/3854 Series, exception handling vector numbers 10 and 13 are reserved.

3.3.5 Interrupt Operations

Interrupts are controlled by an interrupt controller. Figure 3.2 shows a block diagram of the interrupt controller. Figure 3.3 shows the flow up to interrupt acceptance.

Interrupt operation is described as follows.

- When an interrupt condition is met while the interrupt enable register bit is set to 1, an interrupt request signal is sent to the interrupt controller.
- When the interrupt controller receives an interrupt request, it sets the interrupt request flag.
- From among the interrupts with interrupt request flags set to 1, the interrupt controller selects the interrupt request with the highest priority and holds the others pending. (Refer to table 3.2 for a list of interrupt priorities.)
- The interrupt controller checks the I bit of CCR. If the I bit is 0, the selected interrupt request is accepted; if the I bit is 1, the interrupt request is held pending.
- If the interrupt is accepted, after processing of the current instruction is completed, both PC and CCR are pushed onto the stack. The state of the stack at this time is shown in figure 3.4. The PC value pushed onto the stack is the address of the first instruction to be executed upon return from interrupt handling.
- The I bit of CCR is set to 1, masking all further interrupts.
- The vector address corresponding to the accepted interrupt is generated, and the interrupt handling routine located at the address indicated by the contents of the vector address is executed.

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

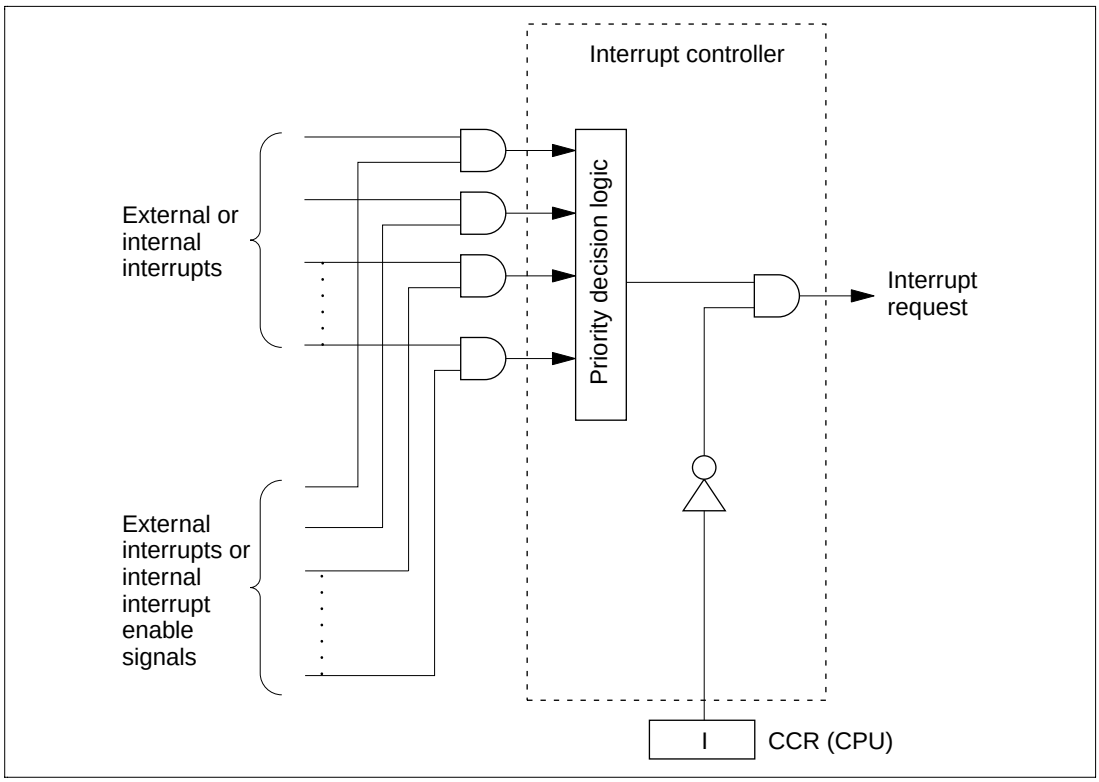
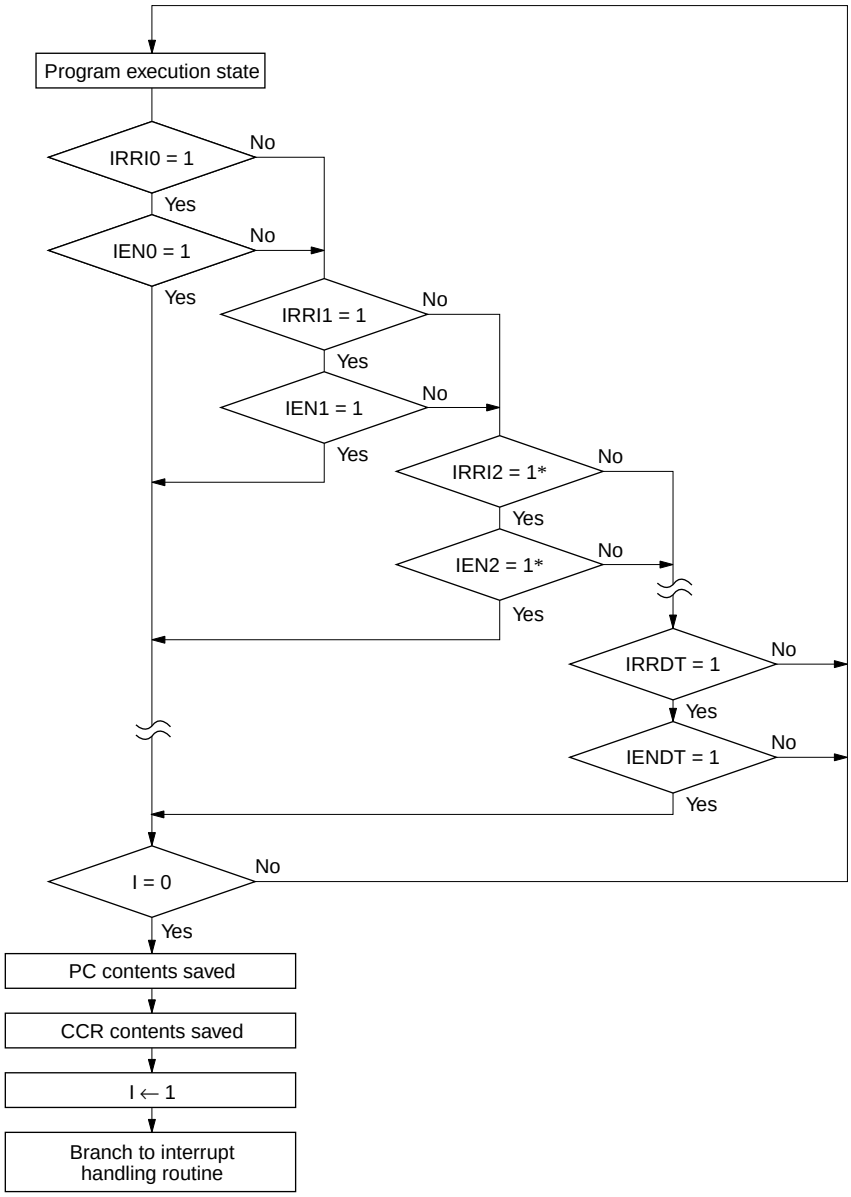


Figure 3.2 Block Diagram of Interrupt Controller



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

Note: * The IRQ_2, SCI1, and timer C interrupts are functions of the H8/3857 Series only, and are not provided in the H8/3854 Series.

Figure 3.3 Flow up to Interrupt Acceptance

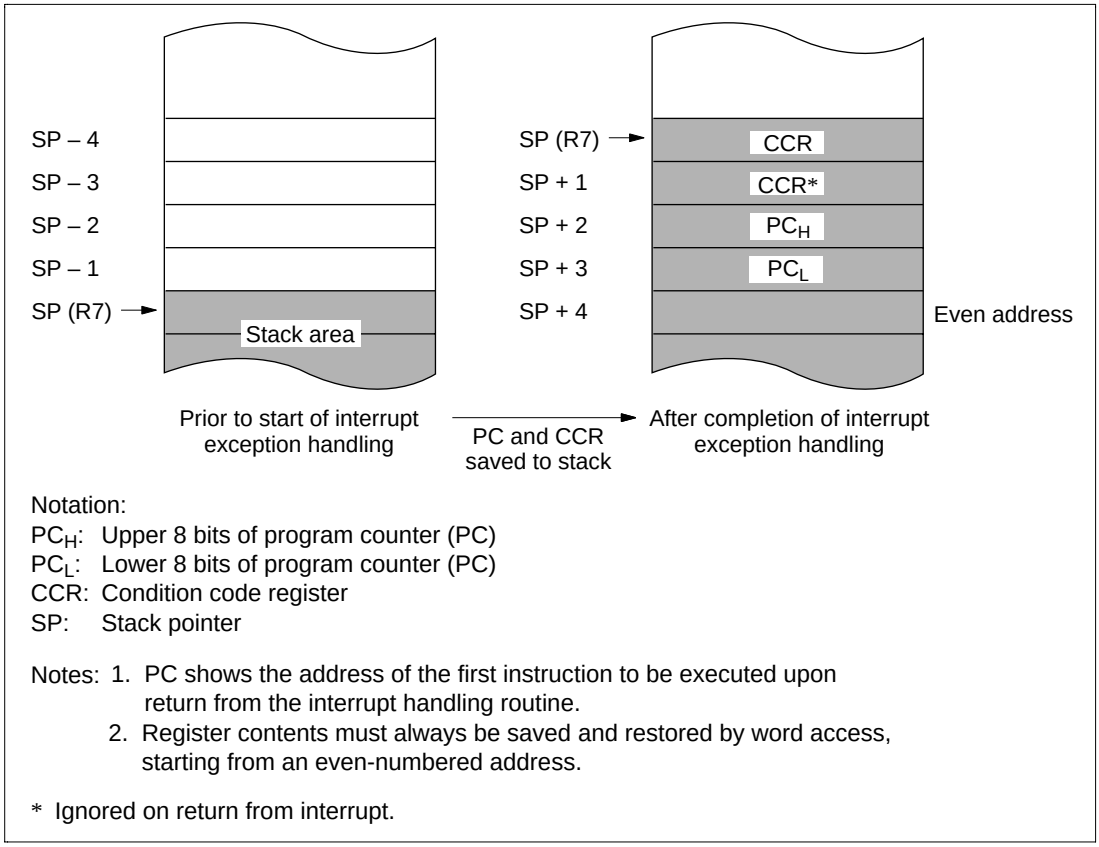


Figure 3.4 Stack State after Completion of Interrupt Exception Handling

Figure 3.5 shows a typical interrupt sequence where the program area is in the on-chip ROM and the stack area is in the on-chip RAM.

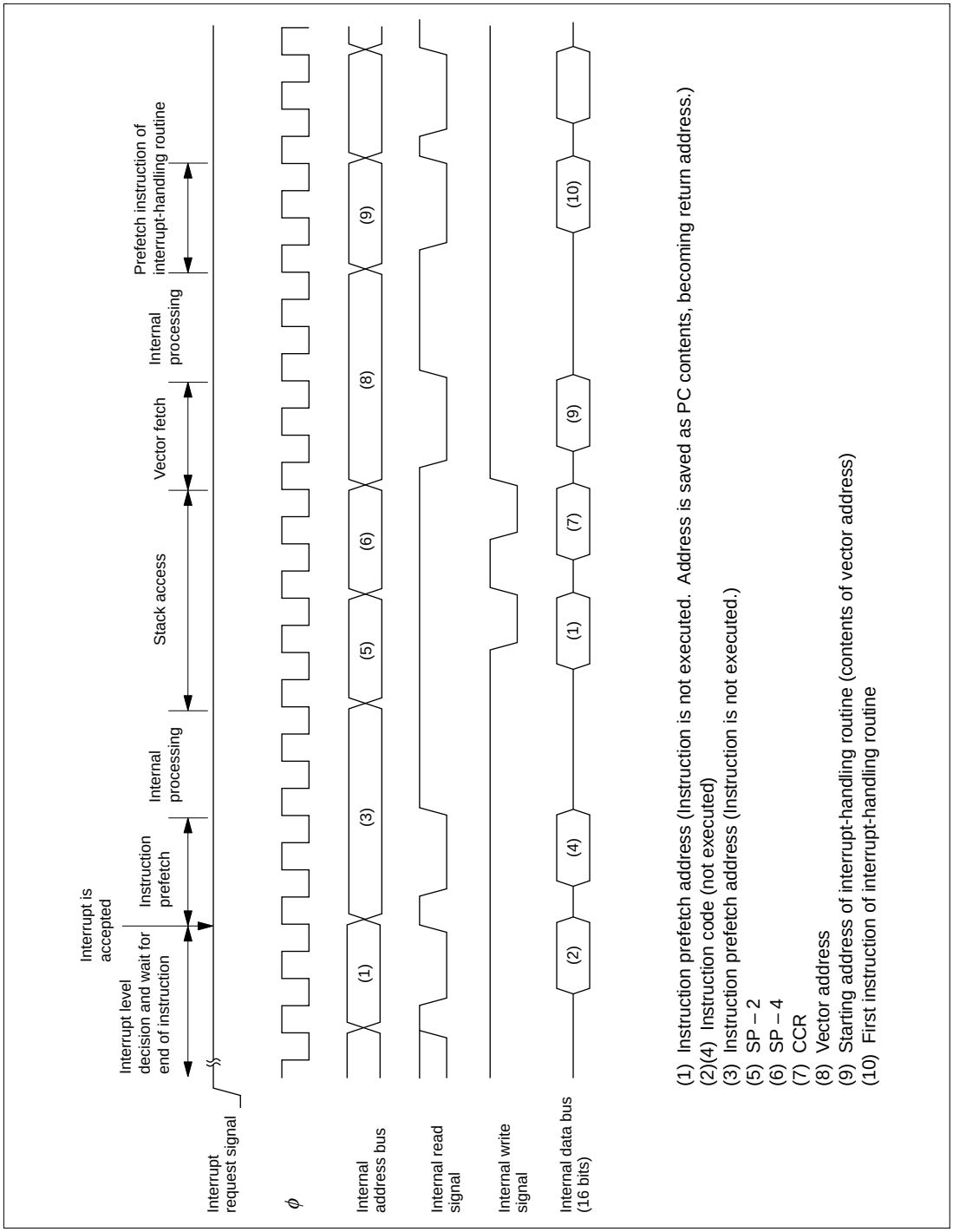


Figure 3.5 Interrupt Sequence

3.3.6 Interrupt Response Time

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

Table 3.4 Interrupt Wait States

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

Note: * Not including EEPMOV instruction.

3.4 Application Notes

3.4.1 Notes on Stack Area Use

When word data is accessed in the H8/3857 Series, the least significant bit of the address is regarded as 0. Access to the stack always takes place in word size, so the stack pointer (SP: R7) should never indicate an odd address. Use PUSH Rn (MOV.W Rn, @–SP) or POP Rn (MOV.W @SP+, Rn) to save or restore register values.

Setting an odd address in SP may cause a program to crash. An example is shown in figure 3.6.

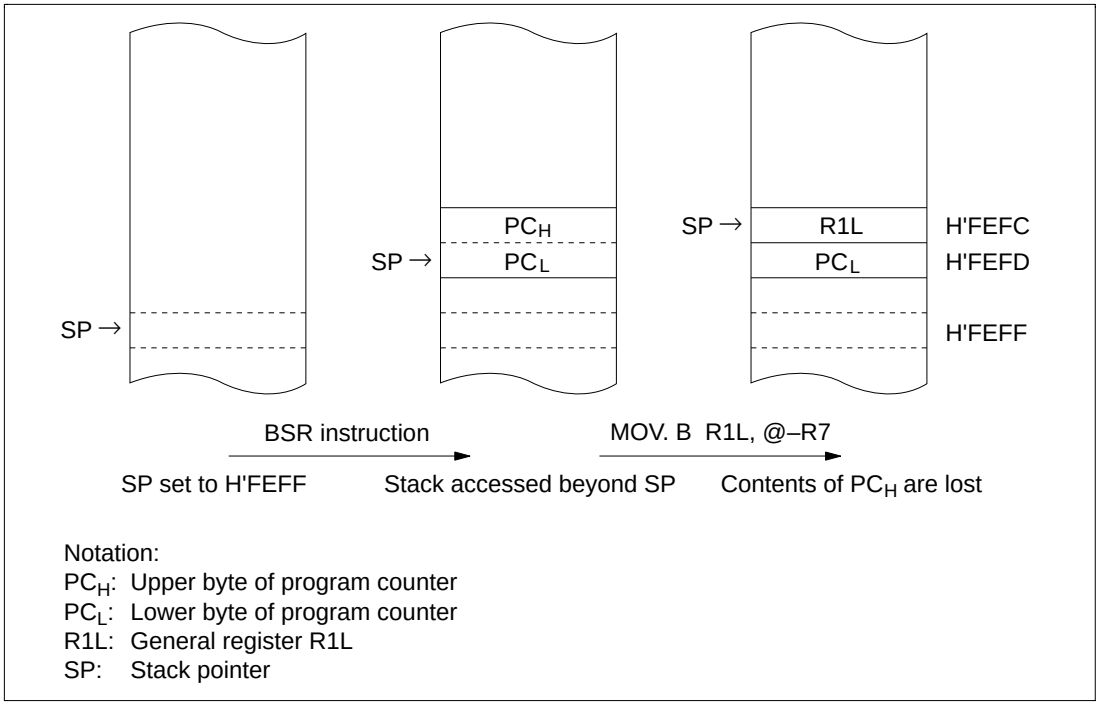


Figure 3.6 Operation when Odd Address is Set in SP

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

3.4.2 Notes on Rewriting Port Mode Registers

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

When an external interrupt pin function is switched by rewriting the port mode register that controls these pins ($\overline{IRQ}_4$, $\overline{IRQ}_3$, $\overline{IRQ}_2$*, $\overline{IRQ}_1$, $\overline{IRQ}_0$, and $\overline{WKP}_7$ to $\overline{WKP}_0$), the interrupt request flag may be set to 1 at the time the pin function is switched, even if no valid interrupt is input at the pin. Be sure to clear the interrupt request flag to 0 after switching pin functions. Table 3.5 shows the conditions under which interrupt request flags are set to 1 in this way.

Note: * Applies to the H8/3857 Series; not provided in the H8/3854 Series.

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

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

Note: * Applies to the H8/3857 Series. In the H8/3854 Series, this flag must always be cleared to 0.

Figure 3.7 shows the procedure for setting a bit in a port mode register and clearing the interrupt request flag.

When switching a pin function, mask the interrupt before setting the bit in the port mode register. After accessing the port mode register, execute at least one instruction (e.g., NOP), then clear the interrupt request flag from 1 to 0. If the instruction to clear the flag is executed immediately after the port mode register access without executing an intervening instruction, the flag will not be cleared.

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

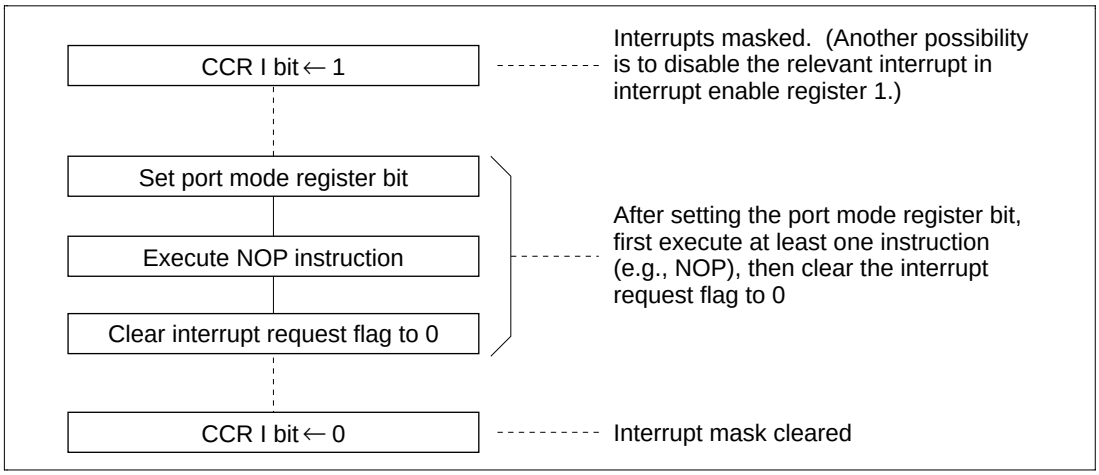


Figure 3.7 Port Mode Register Setting and Interrupt Request Flag Clearing Procedure

Section 4 Clock Pulse Generators

4.1 Overview

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

4.1.1 Block Diagram

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

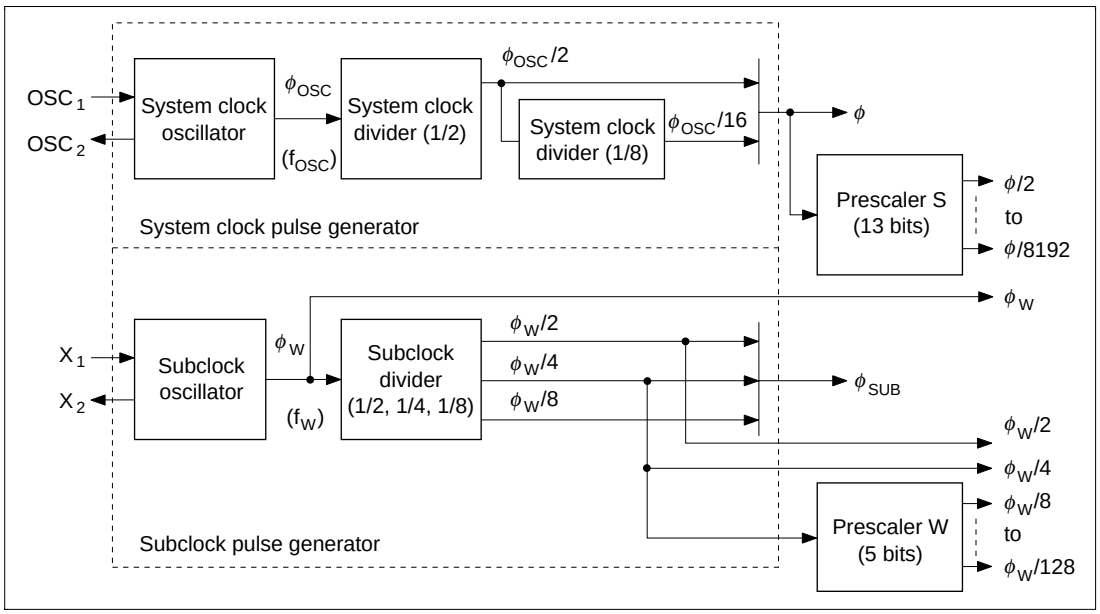


Figure 4.1 Block Diagram of Clock Pulse Generators

4.1.2 System Clock and Subclock

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

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

4.2 System Clock Generator

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

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

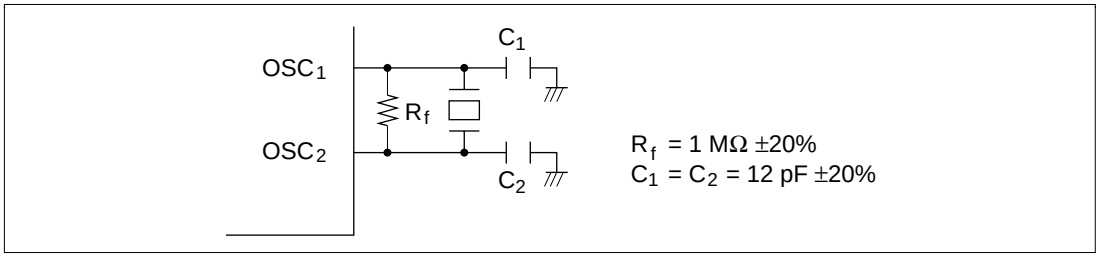


Figure 4.2 Typical Connection to Crystal Oscillator

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

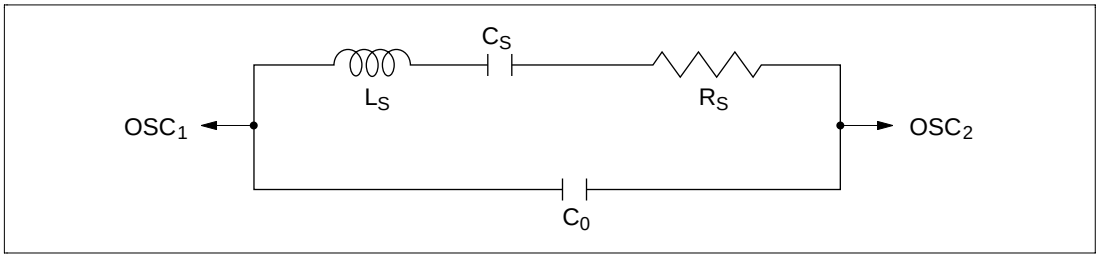


Figure 4.3 Equivalent Circuit of Crystal Oscillator

Table 4.1 Crystal Oscillator Parameters

Frequency (MHz)	2	4	8	10
R_s (max)	500 Ω	100 Ω	50 Ω	30 Ω
C_0 (max)	7 pF	7 pF	7 pF	7 pF

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

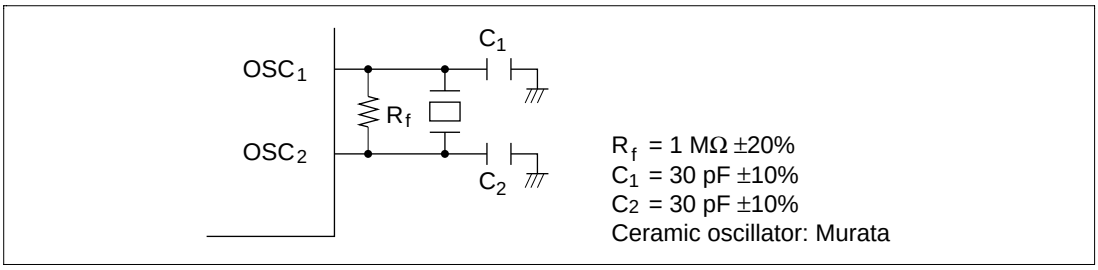


Figure 4.4 Typical Connection to Ceramic Oscillator

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

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

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

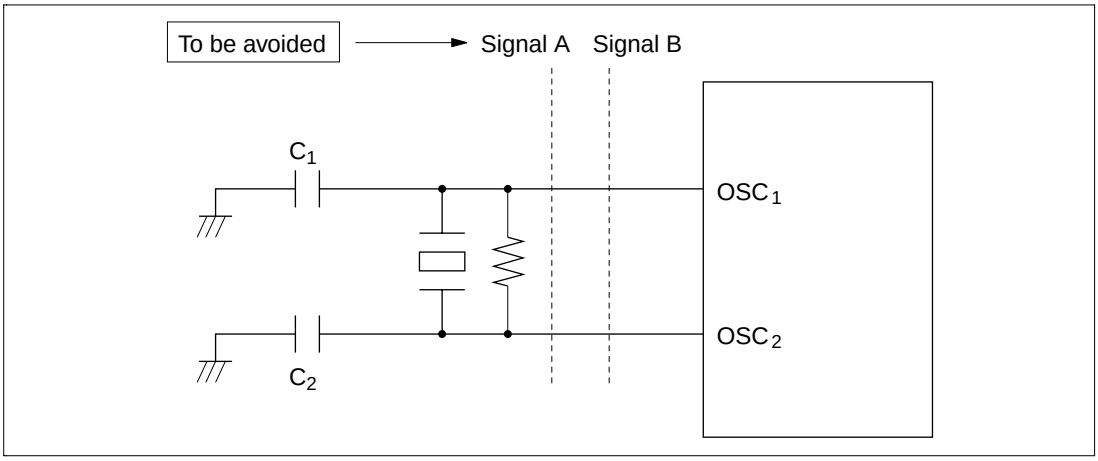


Figure 4.5 Board Design of Oscillator Circuit

Inputting an External Clock: When inputting an external clock, connect it to the OSC_1 pin via a resistance R, and leave the OSC_2 pin open.

An example of the connection in this case is shown in figure 4.6.

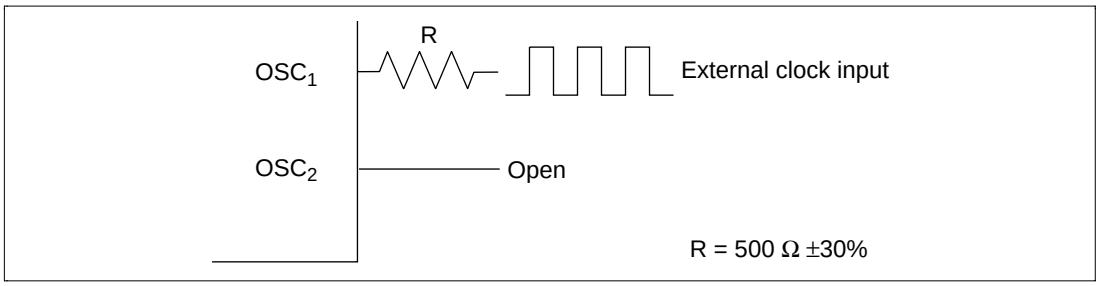


Figure 4.6 Example of Connection when Inputting an External Clock

Frequency	OSC clock (ϕ_{osc})
Duty	45% to 55%

4.3 Subclock Generator

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

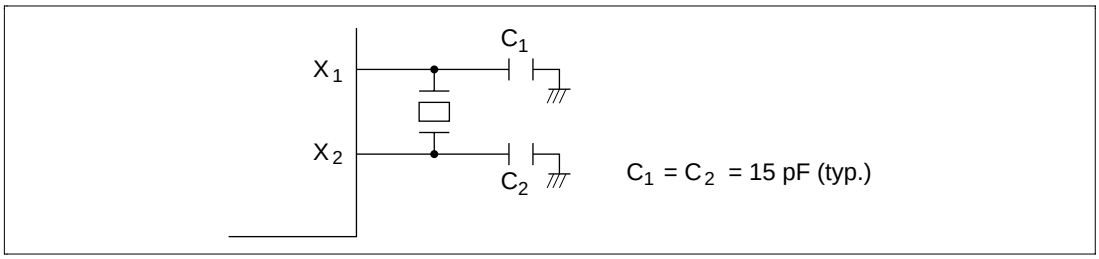


Figure 4.7 Typical Connection to 32.768-kHz Crystal Oscillator

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

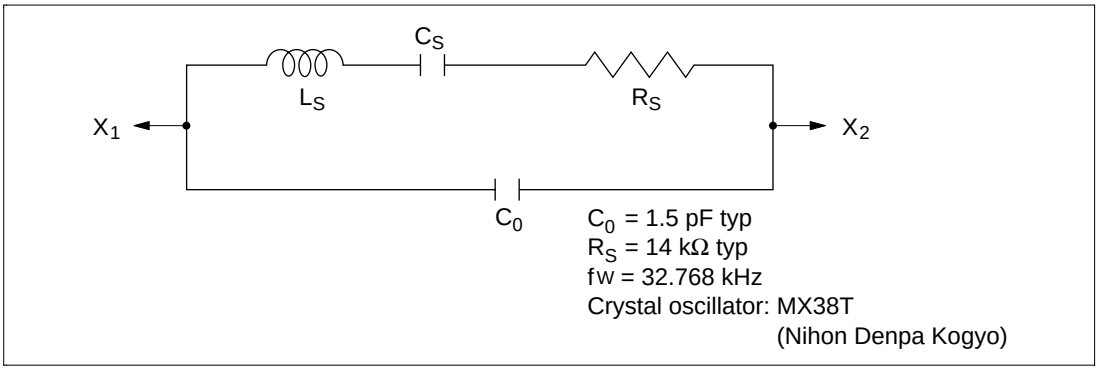


Figure 4.8 Equivalent Circuit of 32.768-kHz Crystal Oscillator

Inputting an External Clock

- Circuit configuration
  An external clock is input to the X_1 pin. The X_2 pin should be left open.
  An example of the connection in this case is shown in figure 4.9.

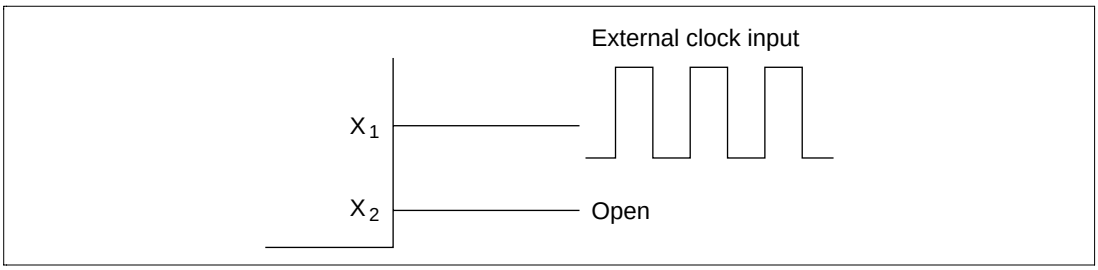


Figure 4.9 Example of Connection when Inputting an External Clock

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

Note: * This is a function of the H8/3857 Series only, and is not provided in the H8/3854 Series.

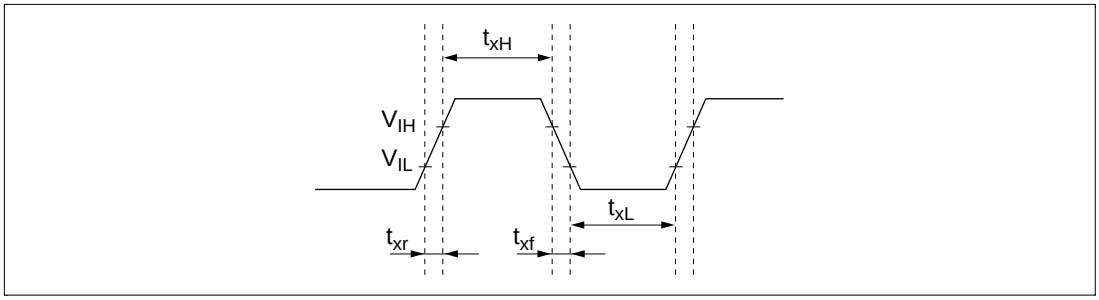


Figure 4.10 External Subclock Timing

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

Table 4.2 DC Characteristics and Timing

(V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, V_{SS} = AV_{SS} = 0.0 V, T_a = –20°C to + 75°C*, unless otherwise specified, including subactive mode)

Item	Symbol	Applicable Pin	Test Conditions	Values Min	Values Typ	Values Max	Unit	Notes
Input high voltage	V_{IH}	X_1		V_{CC} –0.3	—	V_{CC} +0.3	V	Figure 4.10
Input low voltage	V_{IL}			–0.3	—	0.3		
External subclock rise time	t_{xr}			—	—	100	ns	Figure 4.10
External subclock fall time	t_{xf}			—	—	100		
External subclock oscillation frequency	f_x		fx = 32.768kHz	—	32.768	—	kHz	
External subclock high width	t_{xH}			12.0	—	—	µs	Figure 4.10
External subclock low width	t_{xL}			12.0	—	—		
External subclock oscillation frequency	f_x		fx = 38.4kHz	—	38.4	—	kHz	
External subclock high width	t_{xH}			10.0	—	—	µs	Figure 4.10
External subclock low width	t_{xL}			10.0	—	—		

Note: * The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

4.4 Prescalers

The H8/3857 Series and H8/3854 Series are equipped with two on-chip prescalers having different input clocks (prescaler S and prescaler W). Prescaler S is a 13-bit counter using the system clock (ϕ) as its input clock. Its prescaled outputs provide internal clock signals for on-chip peripheral modules. Prescaler W is a 5-bit counter using a 32.768-kHz signal divided by 4 ($\phi_W/4$) as its input clock. Its prescaled outputs are used by timer A as a time base for timekeeping.

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

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

In standby mode, watch mode, subactive mode, and subsleep mode, the system clock pulse generator stops. Prescaler S also stops and is initialized to H'0000.

The CPU cannot read or write prescaler S.

The output from prescaler S is shared by timer A, timer B, timer C*, timer F, SCI1*, SCI3, the A/D converter, LCD controller, and 14-bit PWM*. The divider ratio can be set separately for each on-chip peripheral function.

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

Note: * This is a function of the H8/3857 Series only, and is not provided in the H8/3854 Series.

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

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

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

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

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

4.5 Note on Oscillators

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

Section 5 Power-Down Modes

5.1 Overview

The H8/3857 Series and H8/3854 Series have seven modes of operation after a reset. These include six power-down modes, in which power dissipation is significantly reduced.

Table 5.1 gives a summary of the seven operation modes.

Table 5.1 Operation Modes

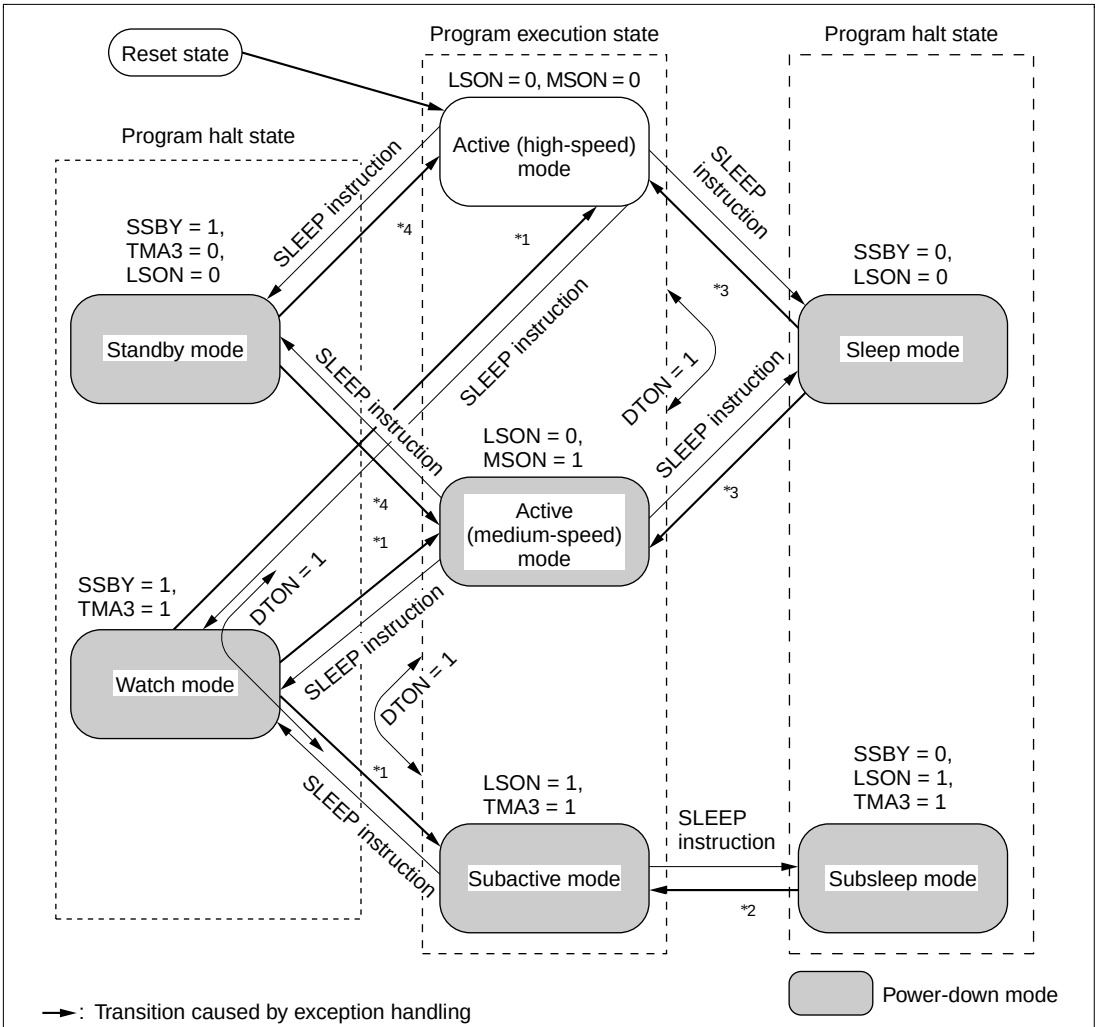
Operating Mode	Description
Active (high-speed) mode	The CPU runs on the system clock, executing program instructions at high speed
Active (medium-speed) mode	The CPU runs on the system clock, executing program instructions at reduced speed
Subactive mode	The CPU runs on the subclock, executing program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

Note: * This is a function of the H8/3857 Series only, and is not provided in the H8/3854 Series.

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

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

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



1. A transition between different modes cannot be made to occur simply because an interrupt request is generated. Make sure that the interrupt is accepted and interrupt handling is performed.
2. Details on the mode transition conditions are given in the explanations of each mode, in sections 5.2 through 5.8.
3. The module standby mode for the LCD controller is initiated by setting a register within the LCD controller itself, and so is not shown in this diagram.

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

*: The timer C, SCI1, and IRQ_2 interrupts are functions of the H8/3857 Series only, and are not provided in the H8/3854 Series.

Figure 5.1 Operation Mode Transition Diagram

Table 5.2 Internal State in Each Operation Mode

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

Notes:
1. Register contents held; high-impedance output.
2. Functions only if external clock or $\phi_W/4$ internal clock is selected; otherwise halted and retained.
3. Functions when timekeeping time-base function is selected.
4. External interrupt requests are ignored. The interrupt request register contents are not affected.
5. In module standby mode, only the clock supplied to the LCD controller is stopped. Register values are retained, and all outputs go to the V_{SS} potential.
6. This is a function of the H8/3857 Series only, and is not provided in the H8/3854 Series.

5.1.1 System Control Registers

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

Table 5.3 System Control Register

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

System Control Register 1 (SYSCR1)

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

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

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

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

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

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

Note: * Don't care

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

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

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

System Control Register 2 (SYSCR2)

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

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

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

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

Bit 4: NESEL	Description	
0	Sampling rate is ϕ_{OSC}/16	(initial value)
1	Sampling rate is ϕ_{OSC}/4	

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

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

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

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

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

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

Note: * Don't care

5.2 Sleep Mode

5.2.1 Transition to Sleep Mode

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

Note: * This is a function of the H8/3857 Series only, and is not provided in the H8/3854 Series.

5.2.2 Clearing Sleep Mode

Sleep mode is cleared by an interrupt (timer A, timer B, timer C*, timer F, IRQ_0, IRQ_1, IRQ_2*, IRQ_3, IRQ_4, WKP_0 to WKP_7, SCI1*, SCI3, A/D converter) or by input at the $\overline{RES}$ pin.

Note: * The timer C, SCI1, and IRQ_2 interrupts are functions of the H8/3857 Series only, and are not provided in the H8/3854 Series.

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

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

5.3 Standby Mode

5.3.1 Transition to Standby Mode

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

5.3.2 Clearing Standby Mode

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

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

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

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

5.3.3 Oscillator Settling Time after Standby Mode is Cleared

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

- When a Crystal Oscillator is Used

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

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

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

Note: * Don't care

- When an External Clock is Used

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

5.3.4 Transition to Standby Mode and Port Pin States

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

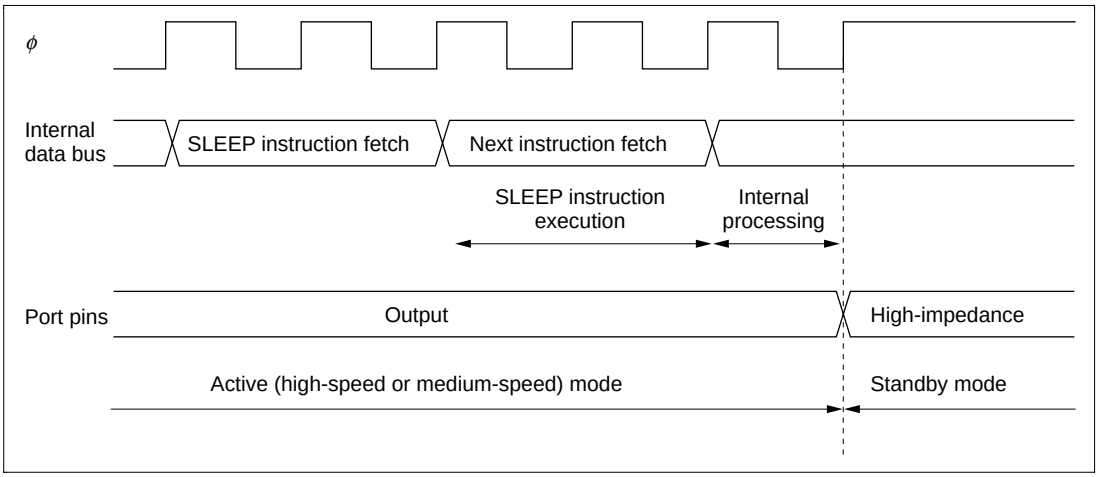


Figure 5.2 Transition to Standby Mode and Port Pin States

5.3.5 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-level widths of the signal must be at least two cycles of system clock ϕ or subclock ϕ_{SUB} (referred to together in this section as the internal clock). As the internal clock stops in standby mode and watch mode, the width of external input signals requires careful attention when a transition is made via these operating modes. Ensure that external input signals conform to the conditions 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 falls immediately after a transition to active (high-speed or medium-speed) mode or subactive mode, after oscillation is started by an interrupt via a different signal, the external input signal 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 widths of at least 2 t_{cyc} or 2 t_{subcyc} are necessary before a transition is made to standby mode or watch mode, as shown in "Capture possible: case 1."

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

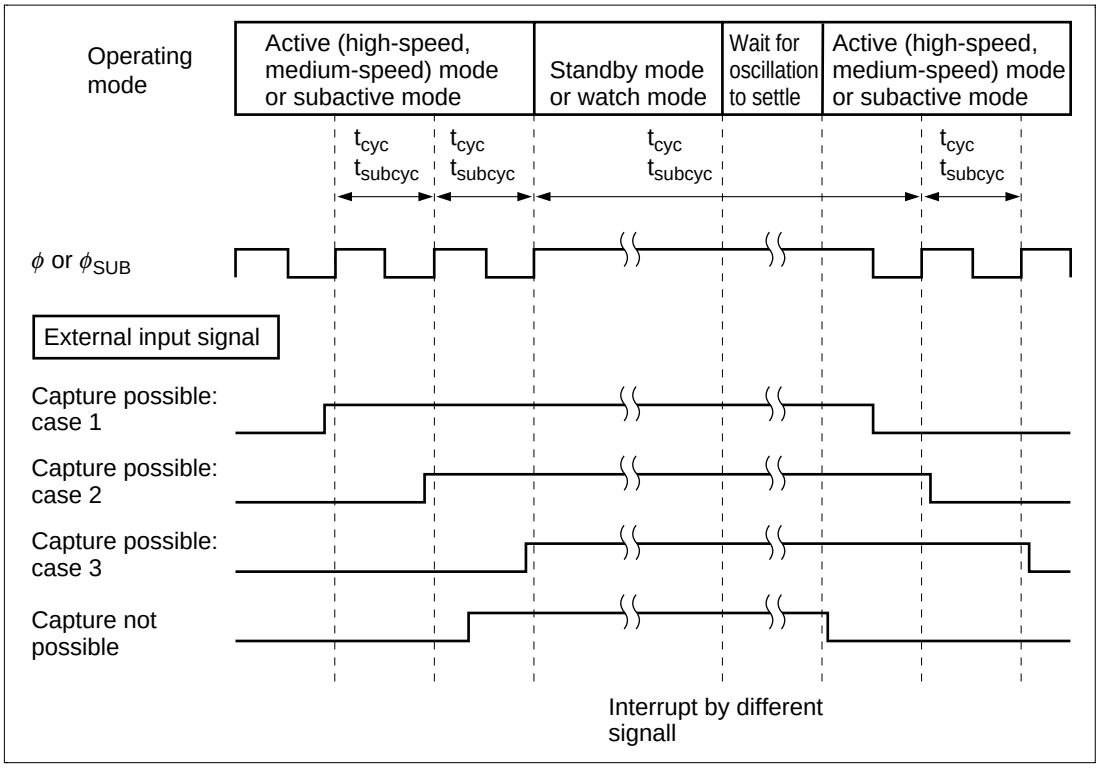


Figure 5.3 External Input Signal Capture when Signal Changes before/after Standby Mode or Watch Mode

4. Input pins to which these notes apply:
   $\overline{IRQ_4}$, $\overline{IRQ_3}$, $\overline{IRQ_2}$*, $\overline{IRQ_1}$, $\overline{IRQ_0}$, $\overline{WKP_7}$ to $\overline{WKP_0}$, $\overline{ADTRG}$, TMIB, TMIC*, TMIF

Note: * H8/3857 Series pin, not provided in the H8/3854 Series.

5.4 Watch Mode

5.4.1 Transition to Watch Mode

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

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

5.4.2 Clearing Watch Mode

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

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

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

5.4.3 Oscillator Settling Time after Watch Mode is Cleared

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

5.4.4 Notes on External Input Signal Changes before/after Watch Mode

See section 5.3.5, Notes on External Input Signal Changes before/after Standby Mode.

5.5 Subsleep Mode

5.5.1 Transition to Subsleep Mode

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

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

Note: * This is a function of the H8/3857 Series only, and is not provided in the H8/3854 Series.

5.5.2 Clearing Subsleep Mode

Subsleep mode is cleared by an interrupt (timer A, timer C*, IRQ_0, IRQ_1, IRQ_2*, IRQ_3, IRQ_4, WKP_0 to WKP_7) or by a low input at the $\overline{RES}$ pin.

Note: * The timer C and IRQ_2 interrupts are functions of the H8/3857 Series only, and are not provided in the H8/3854 Series.

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

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

5.6 Subactive Mode

5.6.1 Transition to Subactive Mode

Subactive mode is entered from watch mode if a timer A, IRQ_0, or WKP_0 to WKP_7 interrupt is requested while the LSON bit in SYSCR1 is set to 1. From subsleep mode, subactive mode is entered if a timer A, timer C*, IRQ_0, IRQ_1, IRQ_2*, IRQ_3, IRQ_4, or WKP_0 to WKP_7 interrupt is requested. A transition to subactive mode does not take place if the I bit of CCR is set to 1 or the particular interrupt is disabled in the interrupt enable register.

Note: * The timer C and IRQ_2 interrupts are functions of the H8/3857 Series only, and are not provided in the H8/3854 Series.

5.6.2 Clearing Subactive Mode

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

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

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

5.6.3 Operating Frequency in Subactive Mode

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

5.7 Active (medium-speed) Mode

5.7.1 Transition to Active (medium-speed) Mode

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

5.7.2 Clearing Active (medium-speed) Mode

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

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

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

5.7.3 Operating Frequency in Active (medium-speed) Mode

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

5.8 Direct Transfer

5.8.1 Direct Transfer Overview

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

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

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

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

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

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

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

Direct Transfer from Subactive Mode to Active (Medium-Speed) Mode: When a SLEEP instruction is executed in subactive mode while the SSBY bit in SYSCR1 is set to 1, the LSON bit in SYSCR1 is cleared to 0, the MSON bit in SYSCR2 is set to 1, the DTON bit

in SYSCR2 is set to 1, and TMA3 bit in TMA is set to 1, a transition is made directly to active (medium-speed) mode via watch mode after the waiting time set in SYSCR1 bits STS2 to STS0 has elapsed.

5.8.2 Calculation of Direct Transfer Time before Transition

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

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

Example: Direct transfer time for the H8/3857 Series and H8/3854 Series
= (2 + 1) × 2tosc + 14 × 16tosc = 230 tosc

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

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

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

Example: Direct transfer time for the H8/3857 Series and H8/3854 Series
= (2 + 1) × 16tosc + 14 × 2tosc = 76 tosc

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

Time Required before Direct Transfer from Subactive Mode to Active (High-Speed) Mode: A direct transfer is made from subactive mode to active (high-speed) mode when a SLEEP instruction is executed in subactive mode while the SSBY bit in SYSCR1 is set to 1, the LSON bit in SYSCR1 is cleared to 0, the MSON bit in SYSCR2 is cleared to 0, the DTON bit in SYSCR2 is set to 1, and the TMA3 bit in TMA is set to 1. A direct transfer time, that is, the time from SLEEP instruction execution to interrupt exception handling completion is calculated by expression (3) below.

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

Example: Direct transfer time for the H8/3857 Series and H8/3854 Series (when CPU clock frequency is ϕw/8 and wait time is 8192 states)
= (2 + 1) × 8tw + (8192 + 14) × 2tosc = 24tw + 16412tosc

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

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

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

Example: Direct transfer time for the H8/3857 Series and H8/3854 Series (when CPU clock frequency is ϕw/8 and wait time is 8192 states)
= (2 + 1) × 8tw + (8192 + 14) × 16tosc = 24tw + 131296tosc

Notation:

tosc:	OSC clock cycle time
tw:	Watch clock cycle time
tcyc:	System clock (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 5.3.5, Notes on External 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 5.3.5, Notes on External 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 5.3.5, Notes on External 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 5.3.5, Notes on External Input Signal Changes before/after Standby Mode.

Section 6 ROM

6.1 Overview

The H8/3857 has 60 kbytes of on-chip flash memory or mask ROM, while the H8/3856 has 48 kbytes, and the H8/3855 40 kbytes, of on-chip mask ROM. The H8/3854 has 60 kbytes of on-chip flash memory or 32 kbytes of on-chip mask ROM, while the H8/3853 has 24 kbytes, and the H8/3852 16 kbytes, of on-chip mask ROM. Note that the H8/3854 F-ZTAT and mask ROM versions have different ROM sizes. The ROM is connected to the CPU by a 16-bit data bus, allowing high-speed 2-state data access for both byte data and word data.

With the flash memory versions (H8/3857F, H8/3854F), programs can be written and erased and programmed either with a general-purpose PROM programmer or on-board.

When carrying out program development using the H8/3854F with the intention of mask ROM implementation, care must be taken with ROM and RAM sizes since the maximum sizes for the mask ROM version are 32 kbytes of ROM and 1 kbyte of RAM.

6.1.1 Block Diagram

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

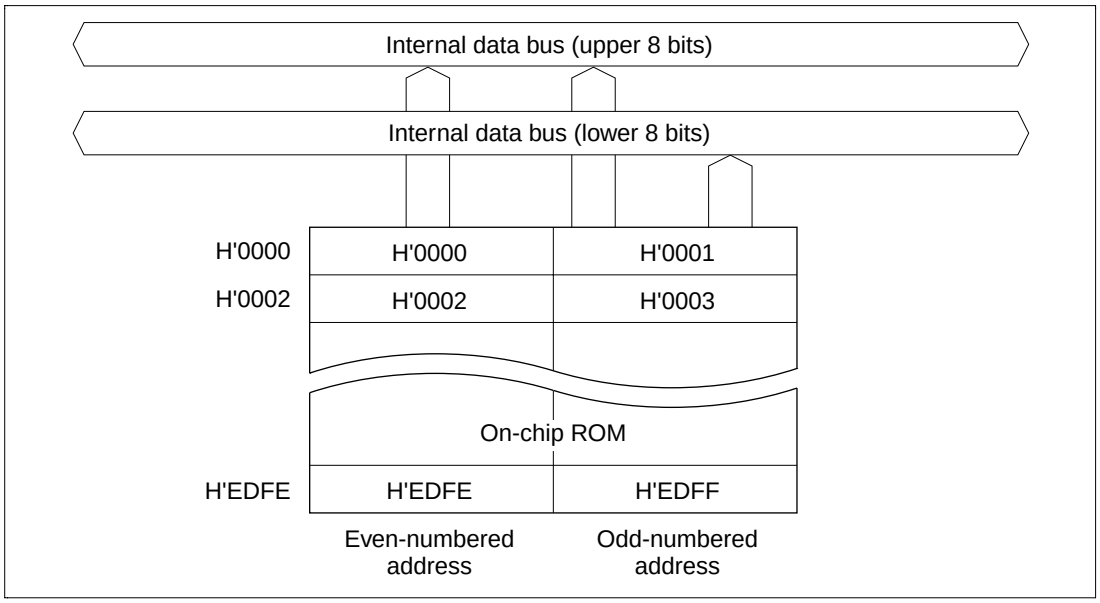


Figure 6.1 ROM Block Diagram (60 kbytes)

6.2 Overview of Flash Memory

6.2.1 Features

Features of the flash memory are summarized below.

- Four flash memory operating modes
  - — Program mode
  - — Erase mode
  - — Program-verify mode
  - — Erase-verify mode
- Programming/erase methods

  The flash memory is programmed 32 bytes at a time. Erasing is performed in block units. To erase multiple blocks, each block must be erased in turn. In block erasing, 1-kbyte, 28-kbyte, 16-kbyte, and 11-kbyte blocks can be set arbitrarily.
- Programming/erase times

  The flash memory programming time is 10 ms (typ.)*[1] for simultaneous 32-byte programming, equivalent to 300 µs (typ.)*[1] per byte, and the erase time is 100 ms (typ.)*[2] per block.
- Reprogramming capability

  The flash memory can be reprogrammed up to 100 times.
- On-board programming modes

  There are two modes in which flash memory can be programmed/erased/verified on-board:
  - — Boot mode
  - — User program mode
- Automatic bit rate adjustment

  For data transfer in boot mode, the chip's bit rate can be automatically adjusted to match the transfer bit rate of the host (9600, 4800, or 2400 bps).
- Protect modes

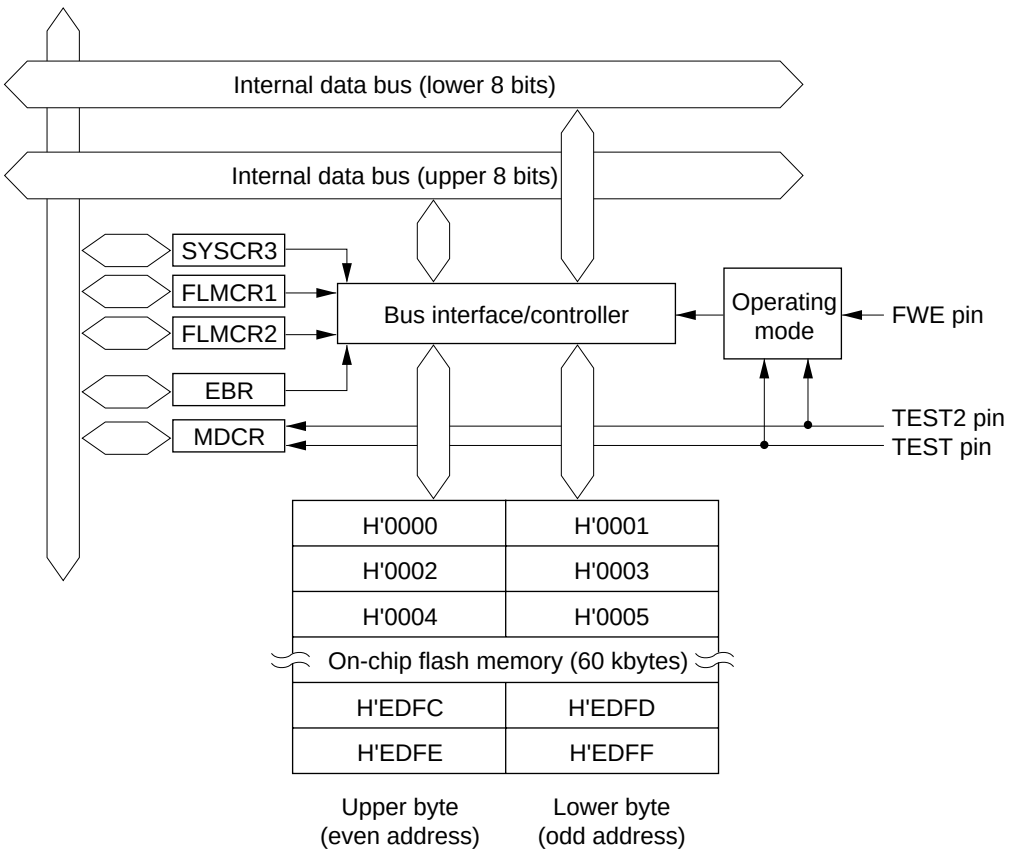
  There are three protect modes—hardware, software, and error—which allow protected status to be designated for flash memory program/erase/verify operations.
- Writer mode

  Flash memory can be programmed/erased in Writer mode, using a PROM programmer, as well as in on-board programming mode.

Notes: 1. Shows the total time during which the P bit in flash memory control register 1 (FLMCR1) is set. The program-verify time is not included.

2. Shows the total time during which the E bit in flash memory control register 1 (FLMCR1) is set. The erase-verify time is not included.

6.2.2 Block Diagram



Notation:
SYSCR3 : System control register 3*
FLMCR1 : Flash memory control register 1*
FLMCR2 : Flash memory control register 2*
EBR : Erase block register*

Note: * The registers that control the flash memory (FLMCR1, FLMCR2, EBR, and MDCR) are for use exclusively by the flash memory version, and are not provided in the mask ROM version. In the mask ROM version, a read access to the address of a register other than MDCR will always return 0, a read access to the address (H'FF89) corresponding to MDCR will return an undefined value, and writes are invalid.

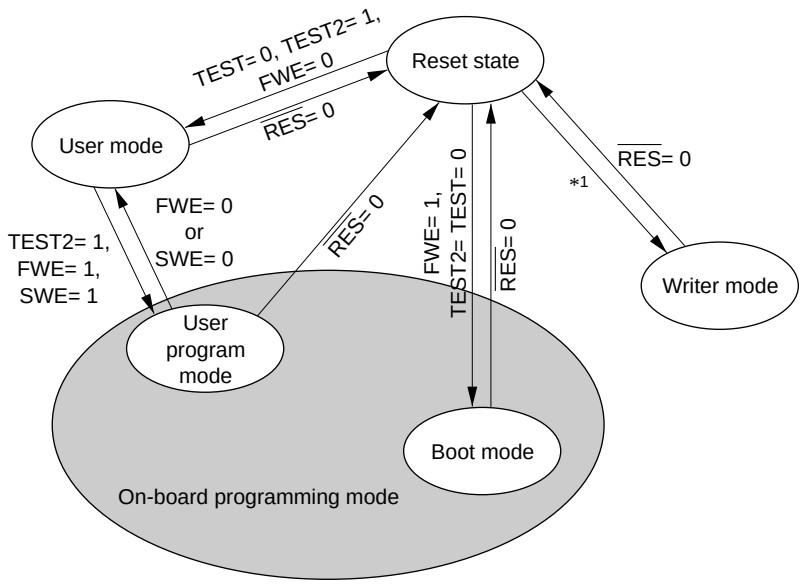
Figure 6.2 Block Diagram of Flash Memory

6.2.3 Flash Memory Operating Modes

(1) Mode Transition Diagram

When the $TEST_2$, TEST, and FWE pins are set in the reset state and a reset start is effected, the chip enters one of the operating modes shown in figure 6.3. In user mode, the flash memory can be read but cannot be programmed or erased.

Modes in which the flash memory can be programmed and erased are boot mode, user program mode, and Writer mode.



Notes: Transitions between user mode and user program mode should only be made when the CPU is not accessing the flash memory.

1. TEST2= 0, TEST= 1
   PB6= PB5= 1
   PB4= 0

Figure 6.3 Flash Memory Related State Transitions

(2) On-Board Programming Modes

(a) Boot Mode

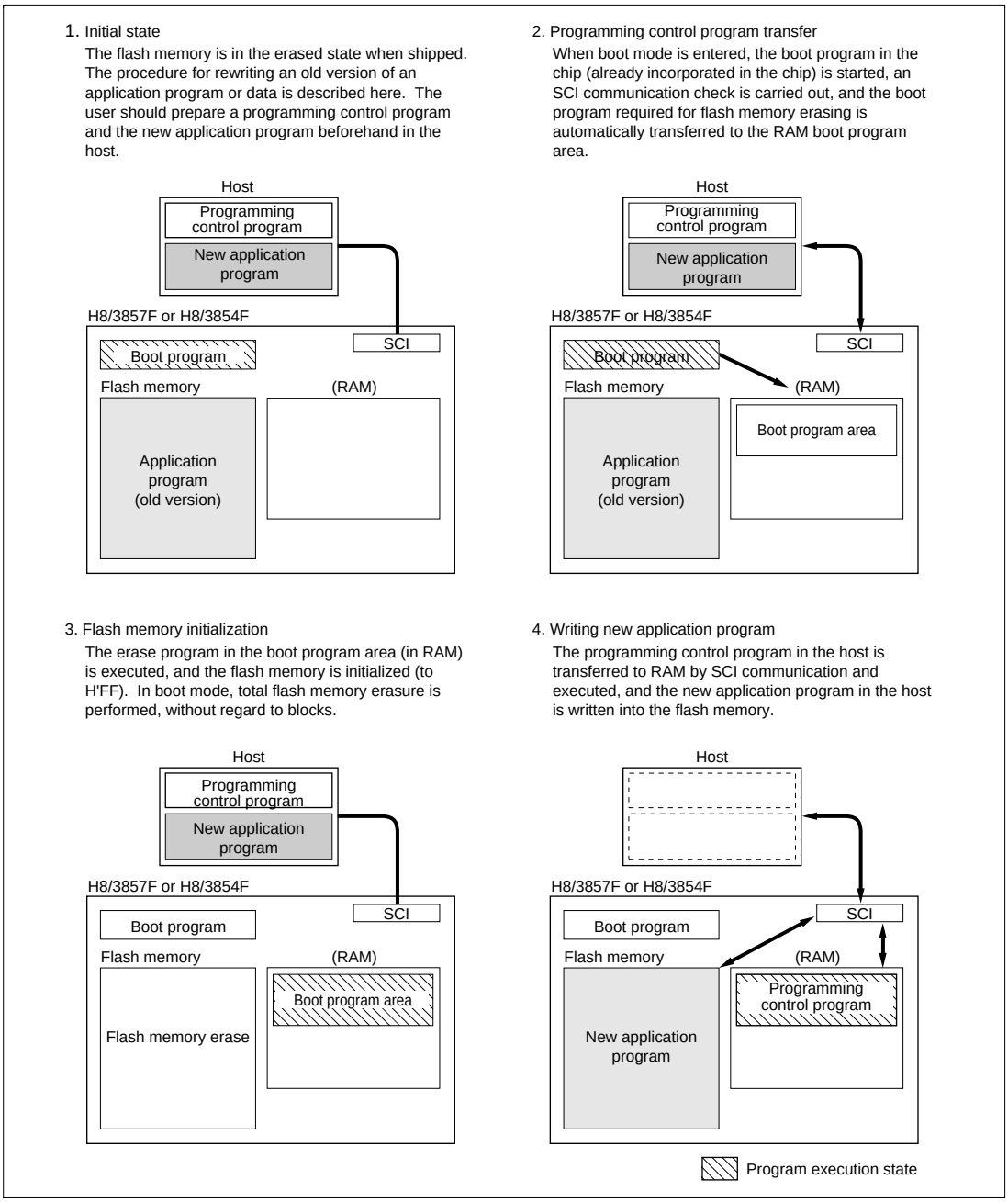
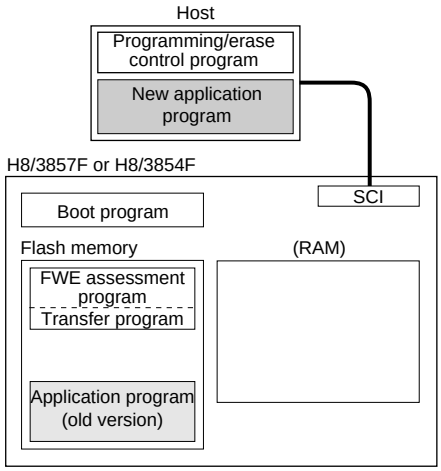


Figure 6.4 Boot Mode

(b) User Program Mode

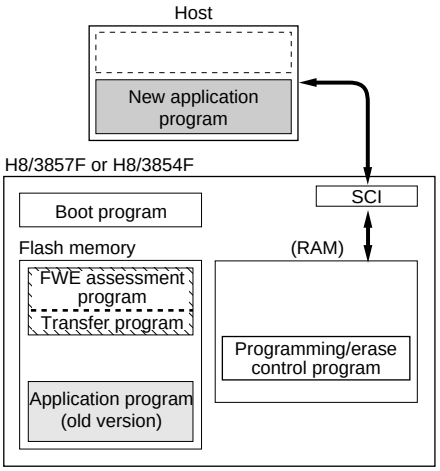
1. Initial state

The FWE assessment program that confirms that a high level has been applied to the FWE pin, and the program that will transfer the programming/erase control program from flash memory to on-chip RAM, should be written into the flash memory by the user beforehand. The programming/erase control program should be prepared in the host or in the flash memory.



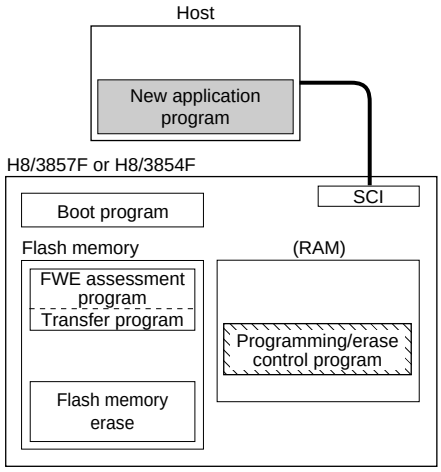
2. Programming/erase control program transfer

When a high level is applied to the FWE pin, user software confirms this fact, executes the transfer program in the flash memory, and transfers the programming/erase control program to RAM.



3. Flash memory initialization

The programming/erase program in RAM is executed, and the flash memory is initialized (to H'FF). Erasing can be performed in block units, but not in byte units.



4. Writing new application program

Next, the new application program in the host is written into the erased flash memory blocks. Do not write to unerased blocks.

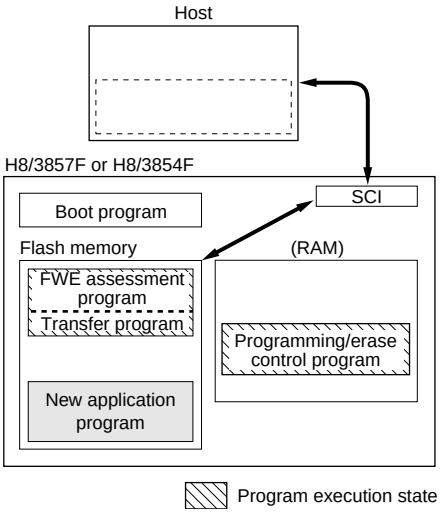


Figure 6.5 User Program Mode (Example)

(3) Differences between Boot Mode and User Program Mode

Table 6.1 Differences between Boot Mode and User Program Mode

	Boot Mode	User Program Mode
Total erase	Possible	Possible
Block erase	Not possible	Possible
Programming control program*	Program/program-verify	Erase/erase-verify Program/program-verify

Note: * To be provided by the user, in accordance with the recommended algorithm.

(4) Block Configuration

The flash memory is divided into one 12-kbyte block, one 16-kbyte block, one 28-kbyte block, and four 1-kbyte blocks.

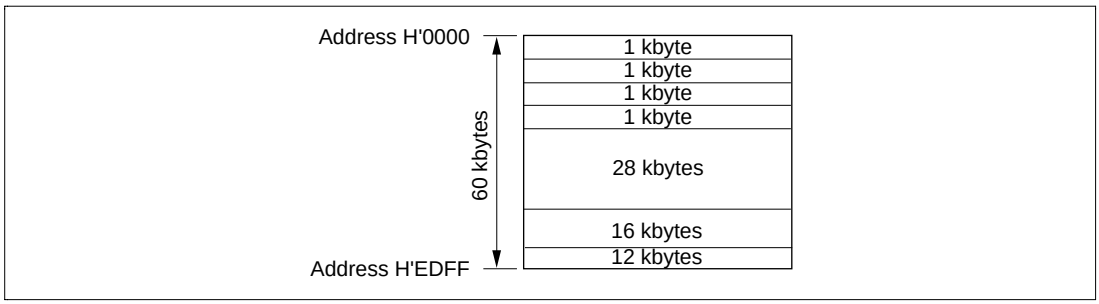


Figure 6.6 Flash Memory Blocks

6.2.4 Pin Configuration

The flash memory is controlled by means of the pins shown in table 6.2.

Table 6.2 Flash Memory Pins

Pin Name	Abbrev.	I/O	Function
Reset	$\overline{\text{RES}}$	Input	Reset
Flash write enable	FWE	Input	Flash program/erase protection by hardware
Test 2	TEST2	Input	Sets H8/3857F operating mode
Test	TEST	Input	Sets H8/3857F operating mode
Transmit data*	TXD	Output	SCI3 transmit data output
Receive data*	RXD	Input	SCI3 receive data input

Note: * The transmit data pin and receive data pin are used in boot mode.

6.2.5 Register Configuration

The registers used to control the on-chip flash memory when enabled are shown in table 6.3. In order to access these registers, the FLSHE bit in SYSCR3 must be set to 1.

Table 6.3 Flash Memory Registers

Register Name	Abbrev.	R/W	Initial Value	Address
Flash memory control register 1	FLMCR1*[5]	R/W*[2]	H'00*[3]	H'FF80*[1]
Flash memory control register 2	FLMCR2*[5]	R/W*[2]	H'00*[4]	H'FF81*[1]
Erase block register	EBR*[5]	R/W*[2]	H'00*[4]	H'FF83*[1]
Mode control register	MDCR	R	Undefined	H'FF89
System control register 3	SYSCR3	R/W	H'00	H'FF8F

Notes: 1. Flash memory register selection is performed by means of the FLSHE bit in system control register 3 (SYSCR3).
2. When the FWE bit in FLMCR1 is cleared to 0, writes are invalid.
3. When a high level is input to the FWE pin, the initial value is H'80.
4. When a low level is input to the FWE pin, or if a high level is input and the SWE bit in FLMCR1 is not set, these registers are initialized to H'00.
5. FLMCR1, FLMCR2, and EBR are 8-bit registers. Only byte accesses are valid for these registers, the access requiring 2 states.

The registers shown in table 6.3 are for use exclusively by the flash memory version. In the mask ROM version, a read access to the address of a register other than MDCR will always return 0, a read access to the MDCR address will return an undefined value, and writes are invalid.

6.3 Flash Memory Register Descriptions

6.3.1 Flash Memory Control Register 1 (FLMCR1)

Bit	7	6	5	4	3	2	1	0
	FWE	SWE	—	—	EV	PV	E	P
Initial value	—*	0	0	0	0	0	0	0
Read/Write	R	R/W	—	—	R/W	R/W	R/W	R/W

Note: * Determined by the state of the FWE pin.

FLMCR1 is an 8-bit register used for flash memory operating mode control. Program-verify mode or erase-verify mode is entered by setting SWE to 1 when FWE = 1, then setting the corresponding bit. Program mode is entered by setting SWE to 1 when FWE = 1, then setting the PSU bit in FLMCR2, and finally setting the P bit. Erase mode is entered by setting SWE to 1 when FWE = 1, then setting the ESU bit in FLMCR2, and finally setting the E bit. FLMCR1 is initialized by a reset and in standby mode. Its initial value is H'80 when a high level is input to the FWE pin, and H'00 when a low level is input.

Writes to the SWE bit in FLMCR1 are enabled only when FWE = 1; writes to the EV and PV bits only when FWE = 1 and SWE = 1; writes to the E bit only when FWE = 1, SWE = 1, and ESU = 1; and writes to the P bit only when FWE = 1, SWE = 1, and PSU = 1.

Bit 7—Flash Write Enable (FWE): Bit 7 sets hardware protection against flash memory programming/erasing. See section 6.9, Flash Memory Programming and Erasing Precautions, for more information on the use of this bit.

Bit 7

FWE	Description
0	When a low level is input to the FWE pin (hardware-protected state)
1	When a high level is input to the FWE pin

Bit 6—Software Write Enable (SWE)*[1, *2]: Bit 6 enables or disables flash memory programming and erasing. (This bit should be set before setting bits ESU, PSU, EV, PV, E, P, and EB6 to EB0, and should not be cleared at the same time as these bits.)

Bit 6

SWE	Description	
0	Programming/erasing disabled	(initial value)
1	Programming/erasing enabled [Setting condition] When FWE = 1	

Bits 5 and 4—Reserved Bits: Bits 5 and 4 are reserved; they are always read as 0 and cannot be modified.

Bit 3—Erase-Verify (EV)*[1]: Bit 3 selects erase-verify mode transition or clearing. (Do not set the SWE, ESU, PSU, PV, E, or P bit at the same time.)

Bit 3

EV	Description	
0	Erase-verify mode cleared	(initial value)
1	Transition to erase-verify mode [Setting condition] When FWE = 1 and SWE = 1	

Bit 2—Program-Verify (PV)*[1]: Bit 2 selects program-verify mode transition or clearing. (Do not set the SWE, ESU, PSU, EV, E, or P bit at the same time.)

Bit 2

PV	Description	
0	Program-verify mode cleared	(initial value)
1	Transition to program-verify mode [Setting condition] When FWE = 1 and SWE = 1	

Bit 1—Erase (E)*[1, *3]: Bit 1 selects erase mode transition or clearing. (Do not set the SWE, ESU, PSU, EV, PV, or P bit at the same time.)

Bit 1

E	Description	
0	Erase mode cleared	(initial value)
1	Transition to erase mode [Setting condition] When FWE = 1, SWE = 1, and ESU = 1	

Bit 0—Program (P) *[1, *3]: Bit 0 selects program mode transition or clearing. (Do not set the SWE, ESU, PSU, EV, PV, or E bit at the same time.)

Bit 0

P	Description	
0	Program mode cleared	(initial value)
1	Transition to program mode [Setting condition] When FWE = 1, SWE = 1, and PSU = 1	

Notes: 1. Do not set multiple bits simultaneously. Do not cut V_{CC} while a bit is set.
2. The SWE bit must not be set or cleared at the same time as other bits (bits EV, PV, E, and P in FLMCR1, and bits ESU and PSU in FLMCR2).
3. P bit and E bit setting should be carried out in accordance with the program/erase algorithms shown in section 6.5, Flash Memory Programming/Erasing. Before setting either of these bits, a watchdog timer setting should be made to prevent program runaway. See section 6.9, Flash Memory Programming and Erasing Precautions, for more information on the use of these bits.

6.3.2 Flash Memory Control Register 2 (FLMCR2)

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

FLMCR2 is an 8-bit register used for monitoring of flash memory program/erase protection (error protection) and for flash memory program/erase mode setup. FLMCR2 is initialized to H'00 by a reset. The ESU and PSU bits are cleared to 0 in standby mode, hardware protect mode, and software protect mode.

Bit 7—Flash Memory Error (FLER): Bit 7 indicates that an error has occurred during an operation on flash memory (programming or erasing). When FLER is set to 1, flash memory goes to the error-protection state.

Bit 7

FLER	Description
0	Flash memory is operating normally Flash memory program/erase protection (error protection) is disabled [Clearing condition] Reset (initial value)
1	An error occurred during flash memory programming/erasing Flash memory program/erase protection (error protection) is enabled [Setting condition] See section 6.6.3, Error Protection

Bits 6 to 2—Reserved Bits: Bits 6 to 2 are reserved; they are always read as 0 and cannot be modified.

Bit 1—Erase Setup (ESU)*: Bit 1 prepares for a transition to erase mode. Set this bit to 1 before setting the E bit in FLMCR1. (Do not set the SWE, PSU, EV, PV, E, or P bit at the same time.)

Bit 1

ESU	Description	
0	Erase setup cleared	(initial value)
1	Erase setup [Setting condition] When FWE = 1 and SWE = 1	

Bit 0—Program Setup (PSU)*: Bit 0 prepares for a transition to program mode. Set this bit to 1 before setting the P bit in FLMCR1. (Do not set the SWE, ESU, EV, PV, E, or P bit at the same time.)

Bit 0

PSU	Description	
0	Program setup cleared	(initial value)
1	Program setup [Setting condition] When FWE = 1 and SWE = 1	

Note: * Do not set multiple bits simultaneously.
Do not cut V_{CC} while a bit is set.

6.3.3 Erase Block Register (EBR)

Bit	7	6	5	4	3	2	1	0
	—	EB6	EB5	EB4	EB3	EB2	EB1	EB0
Initial value	0	0	0	0	0	0	0	0
Read/Write	—	R/W	R/W	R/W	R/W	R/W	R/W	R/W

EBR is a register that specifies the flash memory erase area, block by block. Bits 6 to 0 of EBR are read/write bits. EBR is initialized to H'00 by a reset, in standby mode, when a low level is input to the FWE pin, and when a high level is input to the FWE pin while the SWE bit in FLMCR1 is cleared to 0. When a bit in EBR is set to 1, the corresponding block can be erased. Other blocks are erase-protected. As erasing is carried out on a block-by-block basis, only one bit in EBR should be set at a time (more than one bit must not be set).

The flash memory block configuration is shown in table 6.4. To erase the entire flash memory, individual blocks must be erased in succession.

Table 6.4 Flash Memory Erase Blocks

Block (Size)	Addresses
EB0 (1 kbyte)	H'0000 to H'03FF
EB1 (1 kbyte)	H'0400 to H'07FF
EB2 (1 kbyte)	H'0800 to H'0BFF
EB3 (1 kbyte)	H'0C00 to H'0FFF
EB4 (28 kbytes)	H'1000 to H'7FFF
EB5 (16 kbytes)	H'8000 to H'BFFF
EB6 (12 kbytes)	H'C000 to H'EDFF

6.3.4 Mode Control Register (MDCR)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSDS2	TSDS1
Initial value	0	0	0	0	0	0	—*	—*
Read/Write	—	—	—	—	—	—	R	R

Note: * Determined by the TEST2 and TEST pins.

MDCR is an 8-bit read-only register used to monitor the current operating mode of the H8/3857F.

Bits 7 to 2—Reserved Bits: Bits 7 to 2 are reserved; they are always read as 0 and cannot be modified.

Bits 1 and 0—Test Pin Monitor 2 and 1 (TSDS2, TSDS1): Bits 1 and 0 show values that reflect the input levels at the test pins (TEST2 and TEST) (i.e. they indicate the current operating mode). Bits TSDS2 and TSDS1 correspond to pins TEST2 and TEST, respectively. These bits are read-only, and cannot be modified.

6.3.5 System Control Register 3 (SYSCR3)

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

SYSCR3 is an 8-bit read/write register that controls the on-chip flash memory.

SYSCR3 is initialized to H'00 by a reset.

Bits 7 to 4—Reserved Bits: Bits 7 to 4 are reserved; they are always read as 0 and cannot be modified.

Bit 3—Flash Memory Control Register Enable (FLSHE): Bit 3 controls CPU access to the flash memory control registers (FLMCR1, FLMCR2, and EBR). When the FLSHE bit is set to 1, the flash memory control registers can be read and written to. When FLSHE is cleared to 0, the flash memory control registers are unselected. In this case, the contents of the flash memory control registers are retained.

Bit 3	
FLSHE	**Description**
0	Flash memory control registers are unselected for addresses H'FF80 to H'FF83 (initial value)
1	Flash memory control registers are selected for addresses H'FF80 to H'FF83

Bits 2 to 0—Reserved Bits: Bits 2 to 0 are reserved; they are always read as 0 and cannot be modified.

6.4 On-Board Programming Modes

When an on-board programming mode is selected, the on-chip flash memory can be programmed, erased, and verified. There are two on-board programming modes: boot mode and user program mode. Table 6.5 shows the pin settings for transition to each mode. A state transition diagram for flash memory related modes is shown in figure 6.3.

Table 6.5 On-Board Programming Mode Selection

Mode	Pins		
MCU Mode	**FWE**	**TEST2**	**TEST**
Boot mode	1*[2]	0	0
User program mode*[1]	1	1	0

Notes: 1. The FWE pin should normally be set to 0. Before performing programming, erasing, or verifying, set the FWE pin to 1 and make a transition to user program mode.
2. For the high level application timing, see items (f) and (g) under (3) Notes on Use of Boot Mode in section 6.4.1.

6.4.1 Boot Mode

To use boot mode, a user program for programming and erasing the flash memory must be provided in advance in the host. SCI3 is used in asynchronous mode.

When a reset start is executed after the chip's pins have been set to boot mode, the built-in boot program is activated, and the programming control program provided in the host is transferred sequentially to the chip using SCI3. The chip writes the programming control program received via SCI3 to the programming control program area in the on-chip RAM. After the transfer is completed, execution branches to the start address (H'FB80) of the programming control program area, and the programming control program execution state is entered (flash memory programming is performed).

Therefore, a routine conforming to the programming algorithm described later must be provided in the programming control program transferred from the host.

Figure 6.7 shows the system configuration in boot mode, and figure 6.8 shows the boot mode execution procedure.

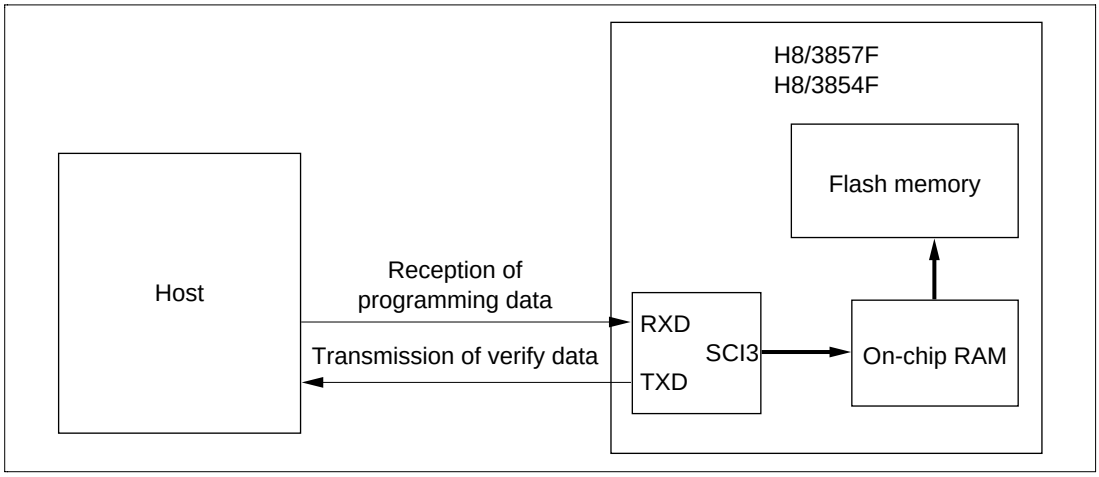
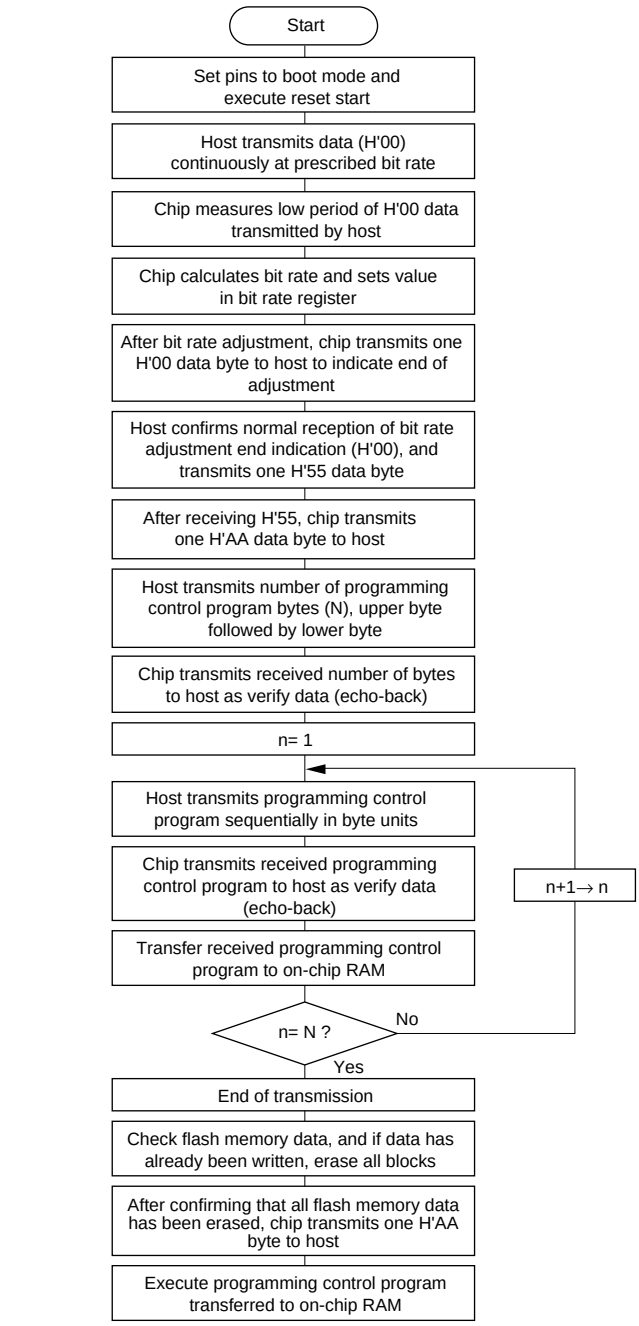


Figure 6.7 System Configuration When Using Boot Mode



Note: If a memory cell does not operate normally and cannot be erased, one H'FF byte is transmitted as an erase error, and the erase operation and subsequent operations are halted.

Figure 6.8 Boot Mode Execution Procedure

(1) Automatic SCI Bit Rate Adjustment

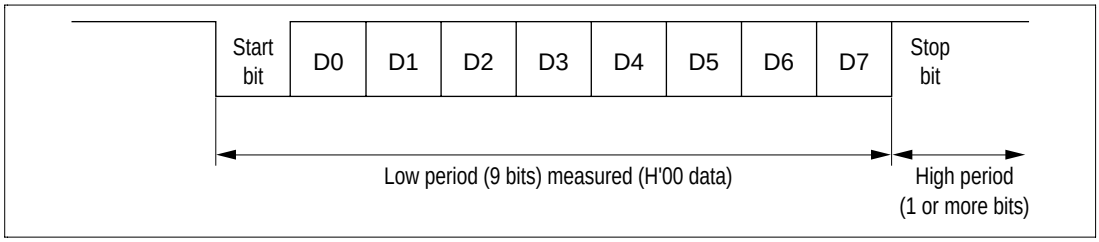


Figure 6.9 RXD Input Signal in Automatic SCI Bit Rate Adjustment

When boot mode is initiated, the chip measures the low period of the asynchronous SCI communication data (H'00) transmitted continuously from the host . The SCI transmit/receive format should be set as 8-bit data, 1 stop bit, no parity. The chip calculates the bit rate of the transmission from the host from the measured low period, and transmits one H'00 byte to the host to indicate the end of bit rate adjustment. The host should confirm that this adjustment end indication (H'00) has been received normally, and transmit one H'55 byte to the chip. If reception cannot be performed normally, initiate boot mode again (reset), and repeat the above operations. Depending on the host's transmission bit rate and the chip's system clock frequency, there will be a discrepancy between the bit rates of the host and the chip. To ensure correct SCI operation, the host's transfer bit rate should be set to 2400, 4800, or 9600 bps*[1].

Table 6.6 shows typical host transfer bit rates and system clock frequencies for which automatic adjustment of the chip's bit rate is possible. The boot program should be executed within this system clock oscillation frequency range*[2].

Table 6.6 System Clock Oscillation Frequencies for which Automatic Adjustment of Chip's Bit Rate is Possible

Host Bit Rate	System Clock Oscillation Frequencies (f_{osc}) for which Automatic Adjustment of Chip's Bit Rate is Possible
9600bps	1.2288 MHz, 2.4576 MHz, 4.9152 MHz, 6 MHz to 10 MHz
4800bps	1.2288 MHz, 2.4576 MHz, 4 MHz to 10 MHz
2400bps	1.2288 MHz, 2 MHz to 10 MHz

Notes: 1. Use a host bit rate setting of 2400, 4800, or 9600 bps only. No other setting should be used.

2. Although the chip may also perform automatic bit rate adjustment with bit rate and system clock combinations other than those shown in table 6.6, a degree of error will arise between the bit rates of the host and the chip, and subsequent transfer will not be performed normally. Therefore, only a combination of bit rate and system clock oscillation frequency within one of the ranges shown in table 6.6 can be used for boot mode execution.

(2) On-Chip RAM Area Divisions in Boot Mode

In boot mode, the 1-kbyte area from H'F780 to H'FB7F is reserved as an area for use by the boot program, as shown in figure 6.10. The area to which the programming control program is transferred comprises addresses H'FB80 to H'FF7F. The boot program area becomes available when the programming control program transferred to RAM switches to the execution state. A stack area should be set up as necessary.

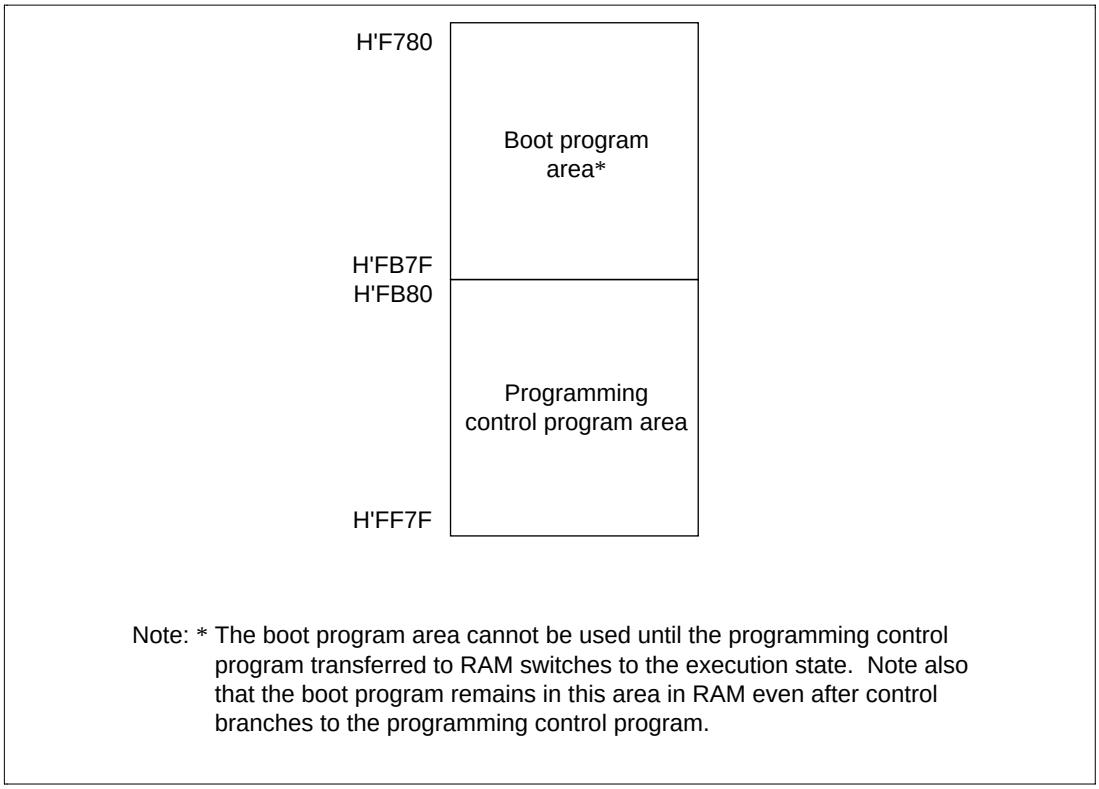


Figure 6.10 RAM Areas in Boot Mode

(3) Notes on Use of Boot Mode

(a) When the chip comes out of reset in boot mode, it measures the low period of the input at the SCI3's RXD pin. The reset should end with RXD high. After the reset ends, it takes about 100 states for the chip to get ready to measure the low period of the RXD input.

(b) In boot mode, if any data has been programmed into the flash memory (if all data is not 1), all flash memory blocks are erased. Boot mode is for use when user program mode is unavailable, such as the first time on-board programming is performed, or if the program activated in user program mode is accidentally erased.

(c) Interrupts cannot be used while the flash memory is being programmed or erased.

(d) The RXD and TXD lines should be pulled up on the board.

(e) Before branching to the programming control program (RAM area address H'FB80), the chip terminates transmit and receive operations by the on-chip SCI3 (by clearing the RE and TE bits to 0 in SCR3), but the adjusted bit rate value remains set in BRR. The transmit data output pin, TXD, goes to the high-level output state ($PCR4_2$ = 1 in port control register 4, $P4_2$ = 1 in port data register 4).

The contents of the CPU's internal general registers are undefined at this time, so these registers must be initialized immediately after branching to the programming control program. In particular, since the stack pointer (SP) is used implicitly in subroutine calls, etc., a stack area must be specified for use by the programming control program.

The initial values of other on-chip registers are not changed.

(f) Boot mode can be entered by making the pin settings shown in table 6.5, and then executing a reset start.

Boot mode can be exited by waiting at least 10 system clock cycles after driving the reset pin low*[2], then setting the FWE, TEST2, and TEST pins to execute reset release*[1]. Boot mode can also be exited when a WDT overflow reset occurs.

Do not change the input levels at the FWE, TEST2, and TEST pins while in boot mode. The FWE pin must not be driven low while the boot program is running or flash memory is being programmed or erased*[3].

(g) If the input level of the TEST2, TEST, or FWE pin is changed (for example, from low to high) during a reset, the MCU's operating mode will change, and as a result, the port states will also change. Therefore, care must be taken to make pin settings to prevent these pins from becoming output signal pins during a reset, or to prevent collision with signals outside the MCU.

Notes: 1. TEST2, TEST, and FWE pin input must satisfy the mode programming setup time (t_{MDS} = 4 states) with respect to the reset release timing.
2. See section 3.2.2, Reset Sequence, and section 6.9, Flash Memory Programming and Erasing Precautions.
3. For further information on FWE application and disconnection, see section 6.9, Flash Memory Programming and Erasing Precautions.

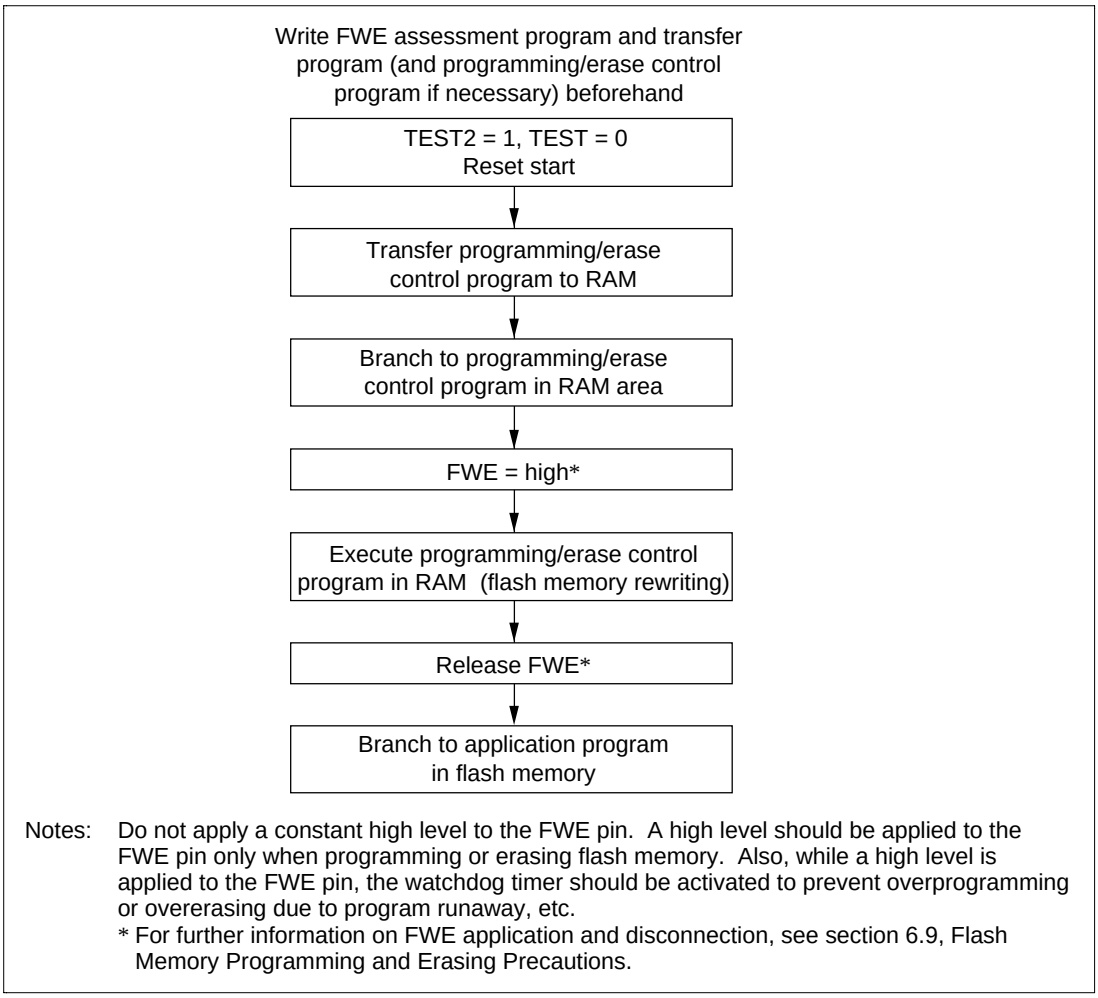
6.4.2 User Program Mode

When set to user program mode, the chip can program and erase its flash memory by executing a user programming/erase control program. Therefore, on-board reprogramming of the on-chip flash memory can be carried out by providing on-board means of FWE control and supply of programming data, and incorporating a programming/erase control program in part of the program area as necessary.

To select user program mode, start up in user program mode (TEST2 = 1, TEST = 0), and apply a high level to the FWE pin. In this mode, on-chip supporting modules other than flash memory operate as they normally would in user mode.

The flash memory itself cannot be read while the SWE bit is set to 1 to carry out flash memory programming or erasing, so the control program that performs programming and erasing must be executed in on-chip RAM.

Figure 6.11 shows the execution procedure when the programming/erase control program is transferred to on-chip RAM.



Notes: Do not apply a constant high level to the FWE pin. A high level should be applied to the FWE pin only when programming or erasing flash memory. Also, while a high level is applied to the FWE pin, the watchdog timer should be activated to prevent overprogramming or overerasing due to program runaway, etc.

* For further information on FWE application and disconnection, see section 6.9, Flash Memory Programming and Erasing Precautions.

Figure 6.11 User Program Mode Execution Procedure

6.5 Flash Memory Programming/Erasing

A software method, using the CPU, is employed to program and erase flash memory in the on-board programming modes. There are four flash memory operating modes: program mode, erase mode, program-verify mode, and erase-verify mode. Transitions to these modes are made by setting the PSU and ESU bits in FLMCR2, and the P, E, PV, and EV bits in FLMCR1.

The flash memory cannot be read while being programmed or erased. Therefore, the program that controls flash memory programming/erasing (the programming control program) should be placed in on-chip RAM, and executed there.

See section 6.9, Flash Memory Programming and Erasing Precautions, for points to note concerning programming and erasing, and section 14.2.6, Flash Memory Characteristics, for the wait times after setting or clearing FLMCR1 and FLMCR2 bits.

Notes: 1. Operation is not guaranteed if setting/resetting of the SWE, EV, PV, E, and P bits in FLMCR1 and the ESU and PSU bits in FLMCR2 is executed by a program in flash memory.
2. When programming or erasing, set FWE to 1 (programming/erasing will not be executed if FWE = 0).
3. Programming should be performed in the erased state. Do not perform additional programming on addresses that have already been programmed.

6.5.1 Program Mode

When writing data or programs to flash memory, the program/program-verify flowchart shown in figure 6.12 should be followed. Performing programming operations according to this flowchart will enable data or programs to be written to flash memory without subjecting the device to voltage stress or sacrificing programming data reliability. Programming should be carried out 32 bytes at a time.

The wait times (x, y, z, α, ß, γ, ε, η) after bits are set or cleared in flash memory control register 1 (FLMCR1) and flash memory control register 2 (FLMCR2), and the maximum number of programming operations (N), are shown in table 14.16 in section 14.2.6, Flash Memory Characteristics.

Following the elapse of (x) µs or more after the SWE bit is set to 1 in FLMCR1, 32-byte programming data is stored in the programming data area and the reprogramming data area, and the 32 bytes of data in the reprogramming data area in RAM are written consecutively to the write addresses. The lower 8 bits of the first address written to must be H'00, H'20, H'40, H'60, H'80, H'A0, H'C0, or H'E0. Thirty-two consecutive byte data transfers are performed. The programming address and programming data are latched in the flash memory. A 32-byte data

transfer must be performed even if writing fewer than 32 bytes; in this case, H'FF data must be written to the extra addresses.

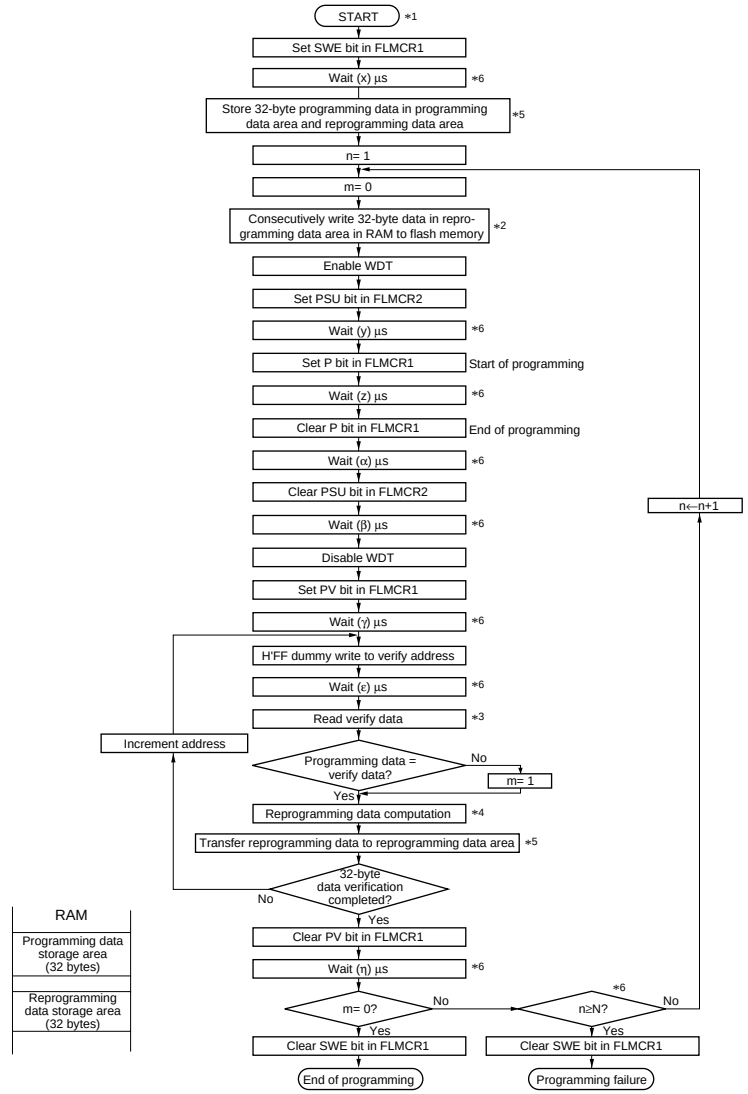
Next, the watchdog timer is set to prevent overprogramming in the event of program runaway, etc. Set a value greater than (y + z + α + ß) µs as the WDT overflow period. After this, preparation for program mode (program setup) is carried out by setting the PSU bit in FLMCR2, and after the elapse of (y) µs or more, the operating mode is switched to program mode by setting the P bit in FLMCR1. The time during which the P bit is set is the flash memory programming time. Make a program setting so that the time for one programming operation is within the range of (z) µs.

6.5.2 Program-Verify Mode

In program-verify mode, the data written in program mode is read to check whether it has been correctly written in the flash memory.

After the elapse of the given programming time, the programming mode is exited (the P bit in FLMCR1 is cleared, then the PSU bit in FLMCR2 is cleared at least (α) µs later). The watchdog timer is cleared following the elapse of more than (y + z + α + ß) µs after being set, and the operating mode is switched to program-verify mode by setting the PV bit in FLMCR1. Before reading in program-verify mode, a dummy write of H'FF data should be made to the addresses to be read. The dummy write should be executed after the elapse of (γ) µs or more. When the flash memory is read in this state (verify data is read in 16-bit units), the data at the latched address is read. Wait at least (ε) µs after the dummy write before performing this read operation. Next, the originally written data is compared with the verify data, and reprogramming data is computed (see figure 6.12) and transferred to the reprogramming data area. After 32 bytes of data have been verified, exit program-verify mode, wait for at least (η) µs, then clear the SWE bit in FLMCR1. If reprogramming is necessary, set program mode again, and repeat the program/program-verify sequence as before. However, ensure that the program/program-verify sequence is not repeated more than (N) times on the same bits.

Note: A 32-byte area for storing programming data and a 32-byte area for storing reprogramming data must be provided in RAM.



Notes: 1. Programming should be performed in the erased state. Do not perform additional programming on addresses that have already been programmed. (Perform 32-byte programming on memory after all 32 bytes have been erased.)
2. Data transfer is performed by byte transfer (word transfer is not possible). The lower 8 bits of the first address written to must be H'00, H'20, H'40, H'60, H'80, H'A0, H'C0, or H'E0. A 32-byte data transfer must be performed even if writing fewer than 32 bytes; in this case, H'FF data must be written to the extra addresses.
3. Verify data is read in 16-bit (word) units.
4. Reprogramming data is determined by the operation shown in the table below (comparison between the data stored in the programming data area and the verify data). Bits for which the reprogramming data is 0 are programmed in the next programming loop. Therefore, even bits for which programming has been completed will be subjected to programming once again if the result of the subsequent verify operation is No.
5. A 32-byte area for storing programming data and a 32-byte area for storing reprogramming data must be provided in RAM. The contents of the latter are rewritten in accordance with reprogramming data computation.
6. The values of x, y, z, α, β, γ, ε, η, and N are shown in section 14.2.6, Flash Memory Characteristics.

Programming Data	Verify Data	Reprogramming Data	Comments
0	0	1	Programmed bits are not reprogrammed
0	1	0	Programming incomplete; reprogram
1	0	1	—
1	1	1	Still in erased state; no action

Note: The memory erased state is "1". Programming is performed on "0" reprogramming data.

Figure 6.12 Program/Program-Verify Flowchart

6.5.3 Erase Mode

To erase an individual flash memory block, follow the erase/erase-verify flowchart (single-block erase) shown in figure 6.13.

The wait times (x, y, z, α, ß, γ, ε, η) after bits are set or cleared in flash memory control register 1 (FLMCR1) and flash memory control register 2 (FLMCR2), and the maximum number of erase operations (N), are shown in table 14.16 in section 14.2.6, Flash Memory Characteristics.

To perform data or program erasure, make a 1-bit setting for the flash memory area to be erased in the erase block register (EBR) at least (x) µs after setting the SWE bit to 1 in flash memory control register 1 (FLMCR1). Next, set up the watchdog timer to prevent overerasing in the event of program runaway, etc. Set a value greater than (y + z + α + ß) µs as the WDT overflow period. After this, preparation for erase mode (erase setup) is carried out by setting the ESU bit in FLMCR2, and after the elapse of (y) µs or more, the operating mode is switched to erase mode by setting the E bit in FLMCR1. The time during which the E bit is set is the flash memory erase time. Ensure that the erase time does not exceed (z) ms.

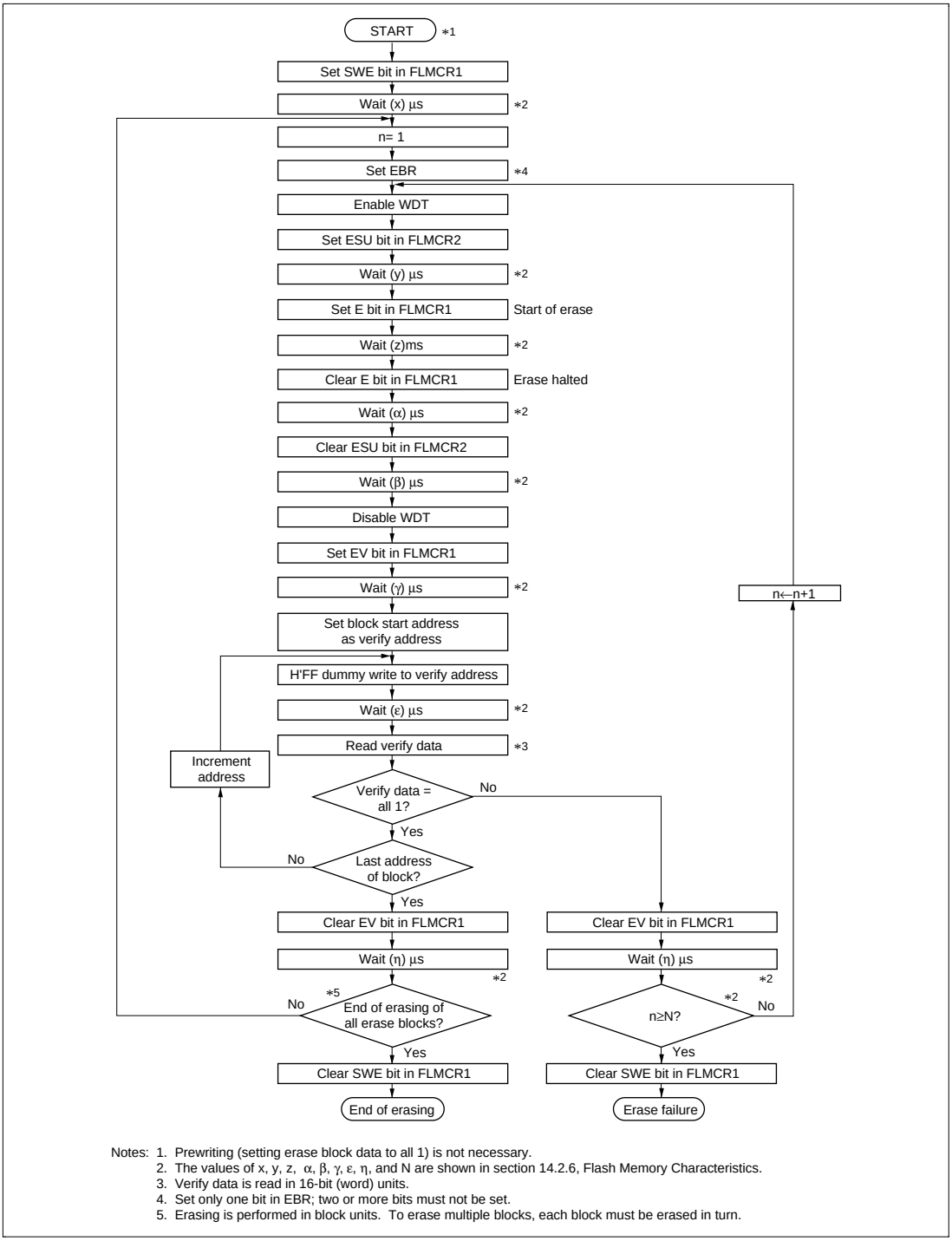
Note: With flash memory erasing, prewriting (setting memory data in the memory to be erased to all 0) is not necessary before starting the erase procedure.

6.5.4 Erase-Verify Mode

In erase-verify mode, data is read after memory has been erased to check whether it has been correctly erased.

After the elapse of the erase time, erase mode is exited (the E bit in FLMCR1 is cleared, then the PSU bit in FLMCR2 is cleared at least (α) µs later). The watchdog timer is cleared following the elapse of more than (y + z + α + ß) µs after being set, and the operating mode is switched to erase-verify mode by setting the EV bit in FLMCR1. Before reading in erase-verify mode, a dummy write of H'FF data should be made to the addresses to be read. The dummy write should be executed after the elapse of (y) µs or more. When the flash memory is read in this state (verify data is read in 16-bit units), the data at the latched address is read. Wait at least (ε) µs after the dummy write before performing this read operation. If the read data has been erased (all 1), execute a dummy write to the next address, and perform an erase-verify. If the read data has not been erased, select erase mode again and repeat the erase/erase-verify sequence as before. However, ensure that the erase/erase-verify sequence is not repeated more than (N) times.

When verification is completed, exit erase-verify mode, and wait for at least (η) µs. If erasure has been completed on all the erase blocks, clear the SWE bit in FLMCR1. If there are any unerased blocks, make a 1-bit setting in EBR for the flash memory block to be erased, and repeat the erase/erase-verify sequence as before.



Notes: 1. Prewriting (setting erase block data to all 1) is not necessary.
2. The values of x, y, z, α, β, γ, ε, η, and N are shown in section 14.2.6, Flash Memory Characteristics.
3. Verify data is read in 16-bit (word) units.
4. Set only one bit in EBR; two or more bits must not be set.
5. Erasing is performed in block units. To erase multiple blocks, each block must be erased in turn.

Figure 6.13 Erase/Erase-Verify Flowchart (Single-Block Erase)

6.6 Flash Memory Protection

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

6.6.1 Hardware Protection

Hardware protection refers to a state in which programming/erasing of flash memory is forcibly disabled or aborted. In this state, the settings in flash memory control registers 1 and 2 (FLMCR1, FLMCR2) and the erase block register (EBR) are reset. (See table 6.7.)

Table 6.7 Hardware Protection

		Functions		
Item	**Description**	**Program**	**Erase**	**Verify*[1]**
FWE pin protection	• When a low level is input to the FWE pin, FLMCR1, FLMCR2 (except the FLER bit), and EBR are initialized, and the program/erase-protected state is entered.*[3]	Not possible	Not possible*[2]	Not possible
Reset/standby protection	• In a reset (including a WDT overflow reset) and in standby mode, FLMCR1, FLMCR2, and EBR are initialized, and the program/erase-protected state is entered. • In a reset via the $\overline{\text{RES}}$ pin, the reset state is not entered unless the $\overline{\text{RES}}$ pin is held low for a minimum of 40 ms (oscillation stabilization time)*[4] after powering on. In the case of a reset during operation, hold the $\overline{\text{RES}}$ pin low for a minimum of 10 system clock cycles (10ø).	Not possible	Not possible*[2]	Not possible

Notes: 1. Two modes: program-verify and erase-verify.
2. All blocks are unerasable and block-by-block specification is not possible.
3. For details see section 6.9, Flash Memory Programming and Erasing Precautions.
4. For details see the AC characteristics in section 14, Electrical Characteristics.

6.6.2 Software Protection

Software protection can be implemented by setting the SWE bit in flash memory control register 1 (FLMCR1), and the erase block register (EBR). With software protection, setting the P or E bit in FLMCR1 does not cause a transition to program mode or erase mode. (See table 6.8.)

Table 6.8 Software Protection

		Functions		
Item	**Description**	**Program**	**Erase**	**Verify*[1]**
SWE bit protection	• Clearing the SWE bit to 0 in FLMCR1 sets the program/erase-protected state for all blocks. (Execute in on-chip RAM.)	Not possible	Not possible	Not possible
Block protection	• Individual blocks can be protected from erasing and programming by settings in the erase block register (EBR)*[2]. • If H'00 is set in EBR, all blocks are protected from erasing and programming.	—	Not possible	Possible

Notes: 1. Two modes: program-verify and erase-verify.
2. When not erasing, clear all EBR bits to 0.

6.6.3 Error Protection

In error protection, an error is detected when MCU runaway occurs during flash memory programming/erasing*[1], or operation is not performed in accordance with the program/erase algorithm, and the program/erase operation is aborted. Aborting the program/erase operation prevents damage to the flash memory due to overprogramming or overerasing.

If the MCU malfunctions during flash memory programming/erasing, the FLER bit is set to 1 in FLMCR2 and the error protection state is entered. FLMCR1, FLMCR2, and EBR settings*[2] are retained, but program mode or erase mode is aborted at the point at which the error occurred. Program mode or erase mode cannot be re-entered by re-setting the P or E bit. However, PV and EV bit setting is enabled, and a transition can be made to verify mode.

FLER bit setting conditions are as follows:

(1) When flash memory is read*[3] during programming/erasing (including a vector read or instruction fetch)

(2) Immediately after the start of exception handling (excluding a reset) during programming/erasing*[4]

(3) When a SLEEP instruction (including software standby) is executed during programming/erasing

Error protection is released only by a reset.

Figure 6.14 shows the flash memory state transition diagram.

Notes: 1. This is the state in which the P bit or E bit is set to 1 in FLMCR1.
2. FLMCR1, FLMCR2, and EBR can be written to. However, registers will be initialized if a transition is made to software standby mode in the error protection state.
3. The read value is undefined.
4. Before exception handling stack and vector read operations are performed.

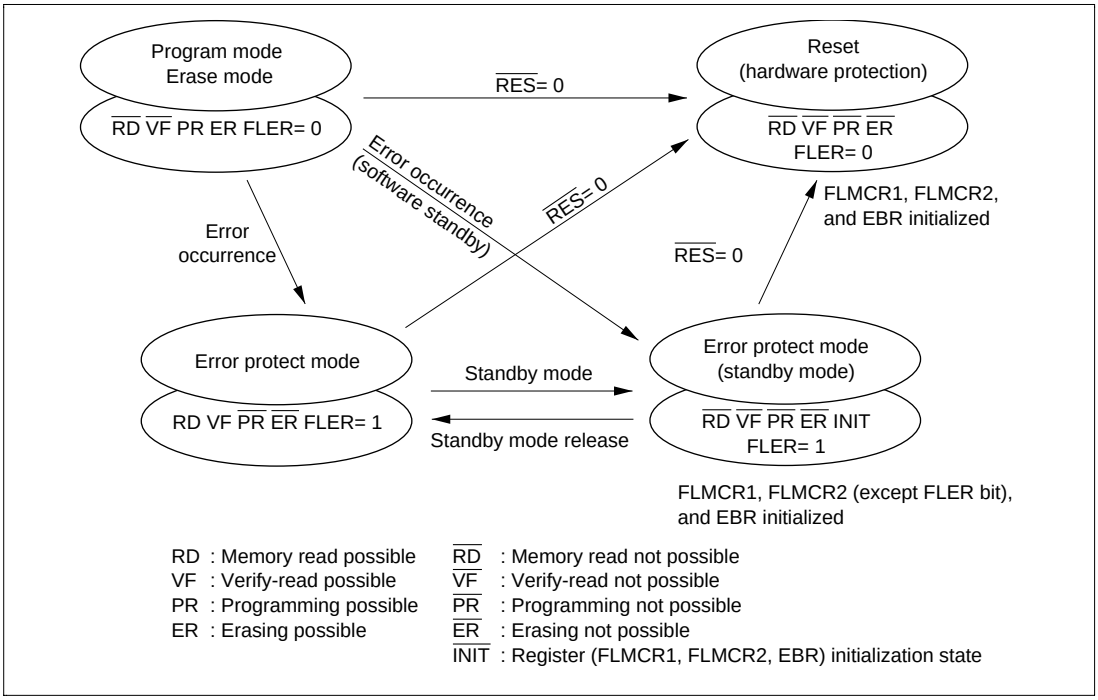


Figure 6.14 Flash Memory State Transitions

The error protection function is invalid for abnormal operations other than the FLER bit setting conditions. Also, if a certain time has elapsed before this protection state is entered, damage may already have been caused to the flash memory. Consequently, this function cannot provide complete protection against damage to flash memory.

To prevent such abnormal operations, therefore, it is necessary to ensure correct operation in accordance with the program/erase algorithm, with the flash write enable (FWE) voltage applied, and to conduct constant monitoring for MCU errors, internally and externally, using the watchdog timer or other means. There may also be cases where the flash memory is in an erroneous programming or erroneous erasing state at the point of transition to the protect mode, or where programming or erasing is not properly carried out because of an abort. In cases such as these, a

forced recovery (program rewrite) must be executed using boot mode. However, it may also happen that boot mode cannot be normally initiated because of overprogramming or overerasing.

6.7 Interrupt Handling during Flash Memory Programming and Erasing

All interrupts should be disabled when flash memory is being programmed or erased (while the P or E bit is set in FLMCR1) and while the boot program is executing in boot mode*[1], to give priority to the program or erase operation. There are three reasons for this:

(1) Interrupt occurrence during programming or erasing might cause a violation of the programming or erasing algorithm, with the result that normal operation could not be assured.
(2) In the interrupt exception handling sequence during programming or erasing, the vector would not be read correctly*[2], possibly resulting in MCU runaway.
(3) If an interrupt occurred during boot program execution, it would not be possible to execute the normal boot mode sequence.

For these reasons, there are conditions for disabling interrupts in the on-board programming modes alone, as an exception to the general rule. However, this provision does not guarantee normal erasing and programming or MCU operation.

All interrupt requests must therefore be disabled inside and outside the MCU when flash memory is programmed or erased.

Notes: 1. Interrupt requests must be disabled inside and outside the MCU until programming by the programming control program has been completed.
2. The vector may not be read correctly in this case for the following two reasons:
   - If flash memory is read while being programmed or erased (while the P or E bit is set in FLMCR1), correct read data will not be obtained (undefined values will be returned).
   - If a value has not yet been written in the interrupt vector table, interrupt exception handling will not be executed correctly.

6.8 Flash Memory Writer Mode

6.8.1 Writer Mode Setting

Programs and data can be written and erased in Writer mode as well as in the on-board programming modes. In Writer mode, the on-chip ROM can be freely programmed using a PROM programmer that supports the Hitachi microcomputer device type with 64-kbyte on-chip flash memory. Flash memory read mode, auto-program mode, auto-erase mode, and status read mode are supported with this device type. In auto-program mode, auto-erase mode, and status read mode, a status polling procedure is used, and in status read mode, detailed internal signals are output after execution of an auto-program or auto-erase operation.

6.8.2 Socket Adapter and Memory Map

In Writer mode, a socket adapter for the relevant kind of package is attached to the PROM programmer. The socket adapter performs 144-pin to 32-pin conversion for the HD64F3857, and 100-pin to 32-pin conversion for the HD64F3854. The socket adaptor ptoduct code is given in table 6.9.

Figure 6.15 shows the memory map in Writer mode, and figures 6.16 (a) and (b) show the socket adapter pin interconnections for the HD64F3857 and the HD64F3854.

Table 6.9 Socket Adapter Product Code

Product Code	Package	Socket Adapter Product Code
HD64F3857	144-pin TQFP (TFP-144)	Details available from Hitachi Sales
	144-pin QFP (FP-144H)	
HD64F3854	100-pin TQFP (TFP-100G)	
	100-pin QFP (FP-100B)	

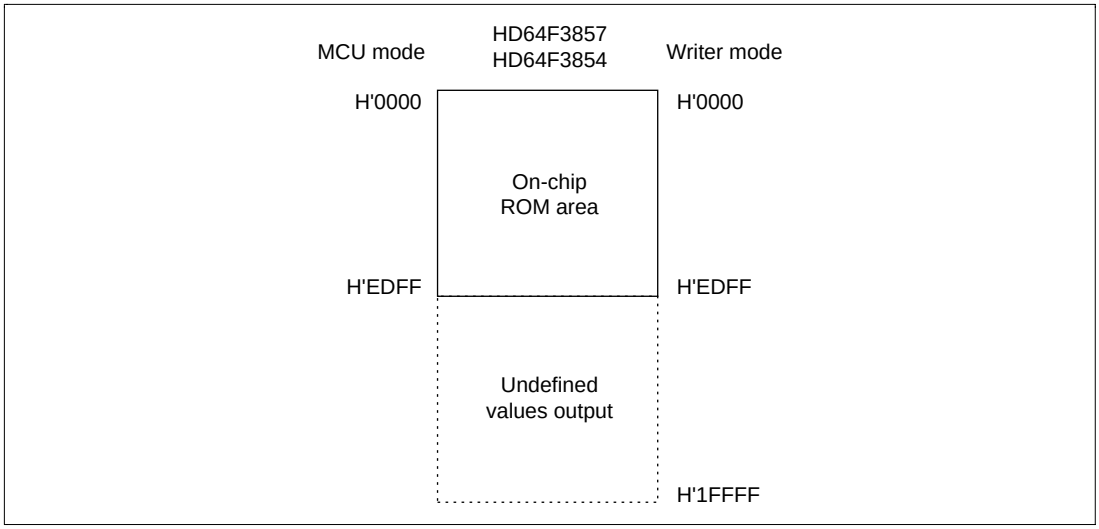


Figure 6.15 Memory Map in Writer Mode

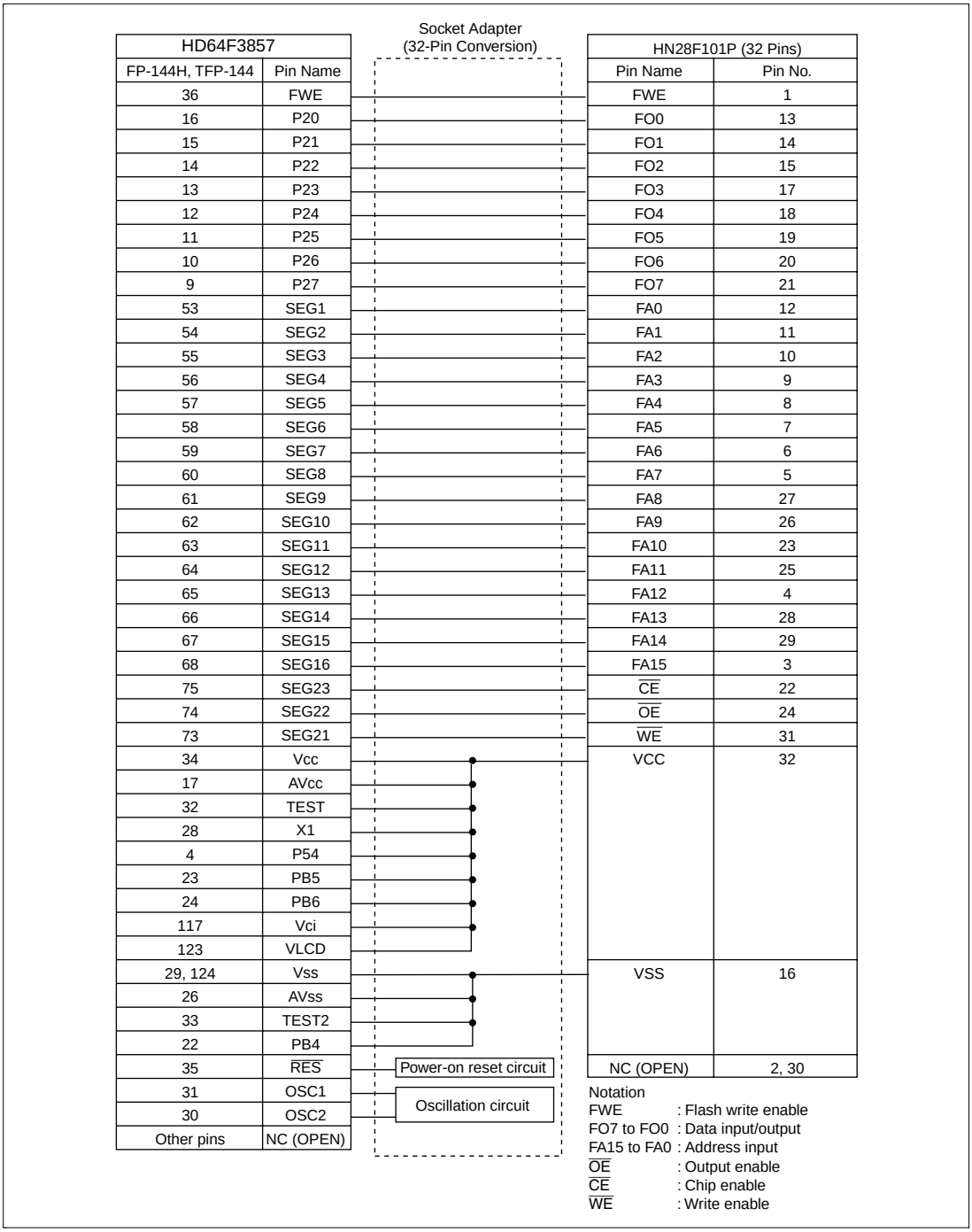


Figure 6.16 (a) HD64F3857 Socket Adapter Pin Interconnections

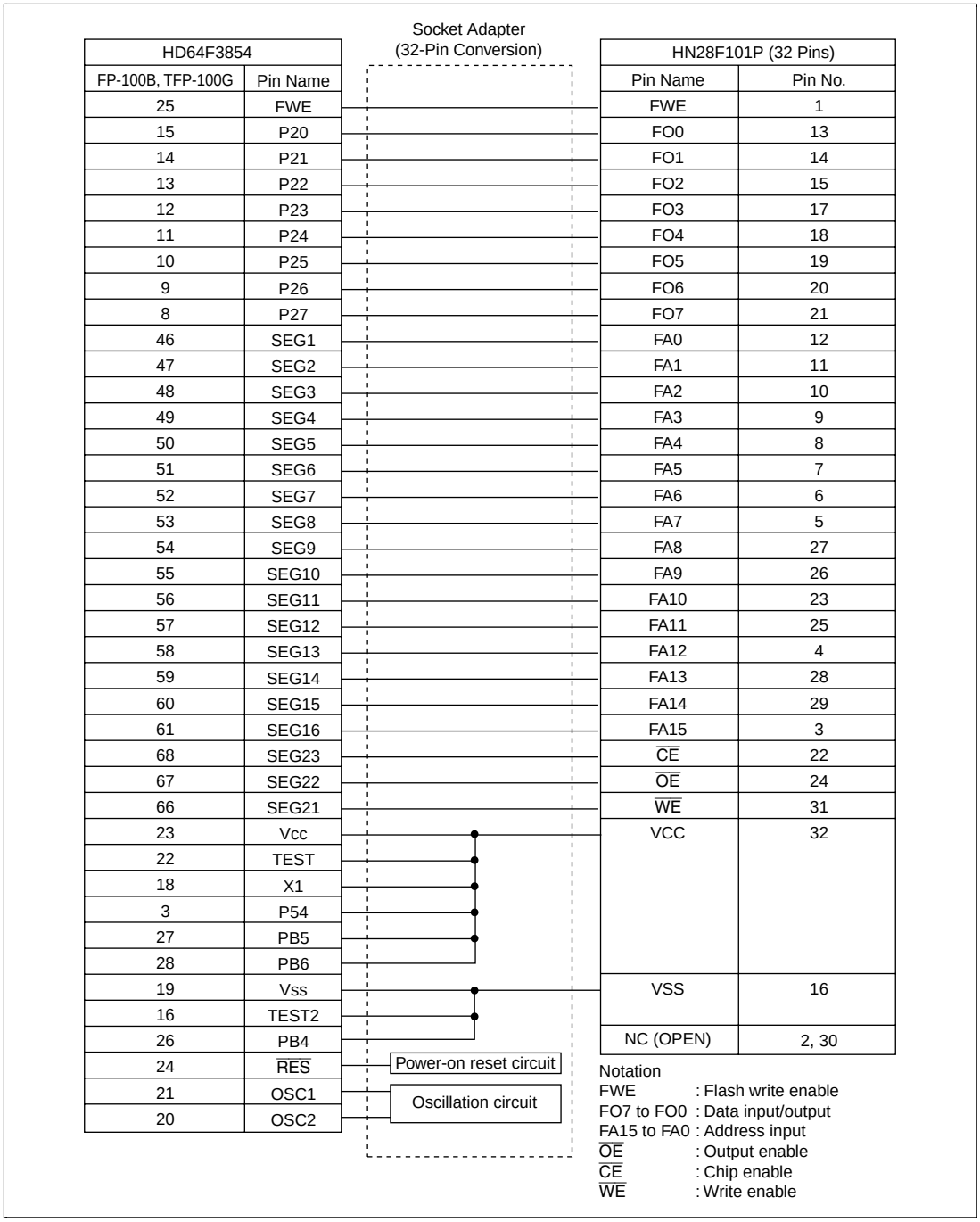


Figure 6.16 (b) HD64F3854 Socket Adapter Pin Interconnections

6.8.3 Writer Mode Operation

Table 6.10 shows how the different operating modes are set when using Writer mode, and table 6.11 lists the commands used in Writer mode. Details of each mode are given below.

Memory Read Mode

Memory read mode supports byte reads.

Auto-Program Mode

Auto-program mode supports programming of 128 bytes at a time. Status polling is used to confirm the end of auto-programming.

Auto-Erase Mode

Auto-erase mode supports automatic erasing of the entire flash memory. Status polling is used to confirm the end of auto-erasing.

Status Read Mode

Status polling is used for auto-programming and auto-erasing, and normal termination can be confirmed by reading the FO_7 signal. In status read mode, error information is output if an error occurs.

Table 6.10 Settings for Operating Modes In Writer Mode

	Pin Names*[4]					
Mode	**FWE**	**$\overline{CE}$**	**$\overline{OE}$**	**$\overline{WE}$**	**FO0 to FO7**	**FA0 to FA15**
Read	H/L	L	L	H	Data output	Ain
Output disable	H/L	L	H	H	Hi-z	X
Command write	H/L*[3]	L	H	L	Data input	Ain*[2]
Chip disable*[1]	H/L	H	X	X	Hi-z	X

Notation

L: Low level

H: High level

X: Undefined

Hi-z: High impedance

Notes: 1. Chip disable is not a standby state; internally, it is an operation state.

2. Ain indicates that there is also address input in auto-program mode.

3. For command writes in auto-program and auto-erase modes, input a high level to the FWE pin.

4. Pin names are those assigned in Writer mode. See figure 6.16 (a) for the H8/3857F, and figure 6.16 (b) for the H8/3854F.

Table 6.11 Writer Mode Commands

Command Name	Number of Cycles	1st Cycle			2nd Cycle		
		Mode	Address	Data	Mode	Address	Data
Memory read mode	1+n	write	X	H'00	read	RA	Dout
Auto-program mode	129	write	X	H'40	write	WA	Din
Auto-erase mode	2	write	X	H'20	write	X	H'20
Status read mode	3	write	X	H'71	write	X	H'71

Notation

RA: Read address

WA: Programming address

Dout: Read data

Din: Programming data

Notes: 1. In auto-program mode, 129 cycles are required for command writing by means of a simultaneous 128-byte write.

2. In memory read mode, the number of cycles depends on the number of address write cycles (n).

6.8.4 Memory Read Mode

(1) After the end of an auto-program, auto-erase, or status read operation, the command wait state is entered. To read memory contents, a transition must be made to memory read mode by means of a command write before the read is executed.

(2) Command writes can be performed In memory read mode, just as in the command wait state.

(3) Once memory read mode has been entered, consecutive reads can be performed.

(4) After power-on, memory read mode is entered.

(5) Do not make a setting outside the valid address range.

Table 6.12 AC Characteristics in Memory Read Mode

(Conditions: V_{CC} = 5.0 V ±10%, V_{SS} = 0 V, T_a = 25°C ±5°C)

Item	Symbol	Min	Max	Unit	Notes
Command write cycle	t_{nxtc}	20	—	µs	
$\overline{CE}$ 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	

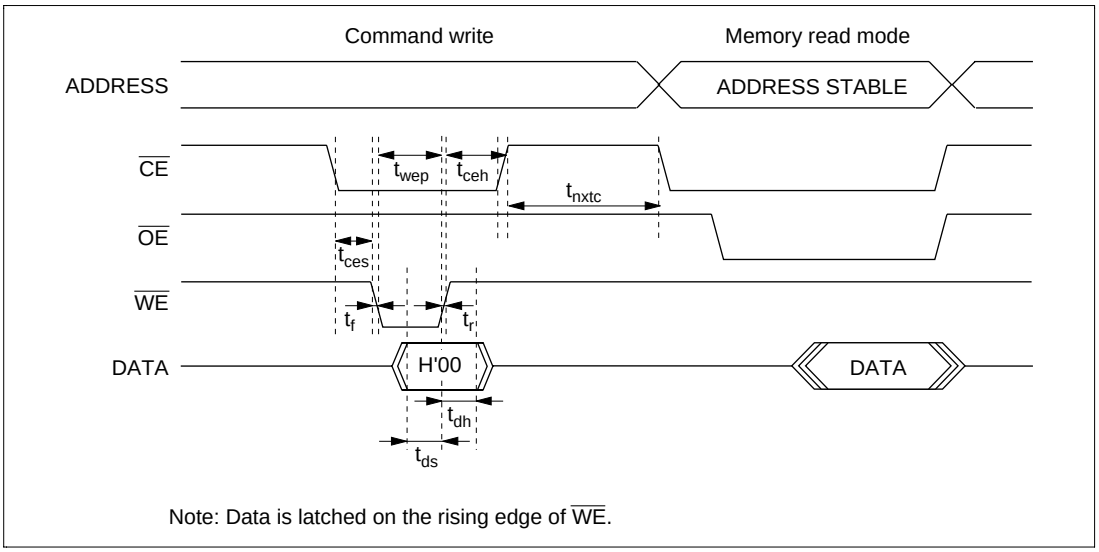


Figure 6.17 Timing Waveforms for Memory Read after Command Write

Table 6.13 AC Characteristics in Transition from Memory Read Mode to Another Mode

(Conditions: V_{CC} = 5.0 V ±10%, V_{SS} = 0 V, T_a = 25°C ±5°C)

Item	Symbol	Min	Max	Unit	Notes
Command write cycle	t_{nxtc}	20	—	µs	
$\overline{CE}$ 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	

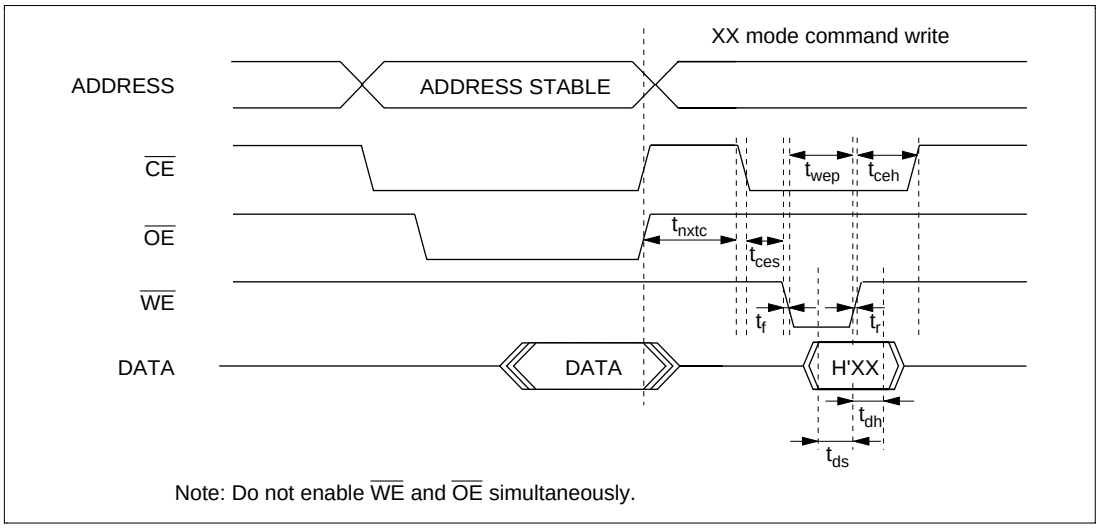


Figure 6.18 Timing Waveforms in Transition from Memory Read Mode to Another Mode

Table 6.14 AC Characteristics in Memory Read Mode (2)

(Conditions: V_{CC} = 5.0 V ±10%, V_{SS} = 0 V, T_a = 25°C ±5°C)

Item	Symbol	Min	Max	Unit	Notes
Access time	t_{acc}	—	20	µs	
$\overline{CE}$ output delay time	t_{ce}	—	150	ns	
$\overline{OE}$ output delay time	t_{oe}	—	150	ns	
Output disable delay time	t_{df}	—	100	ns	
Data output hold time	t_{oh}	5	—	ns	

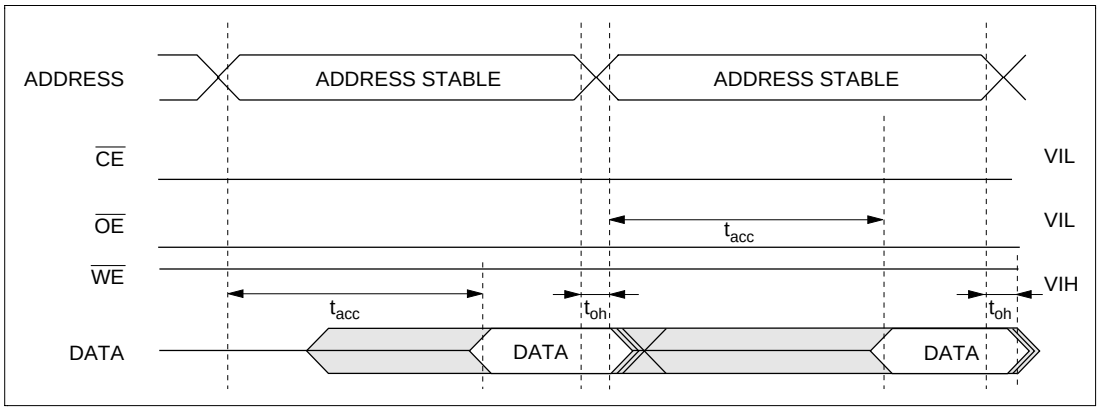


Figure 6.19 Timing Waveforms for $\overline{CE}$, $\overline{OE}$ Enable State Read

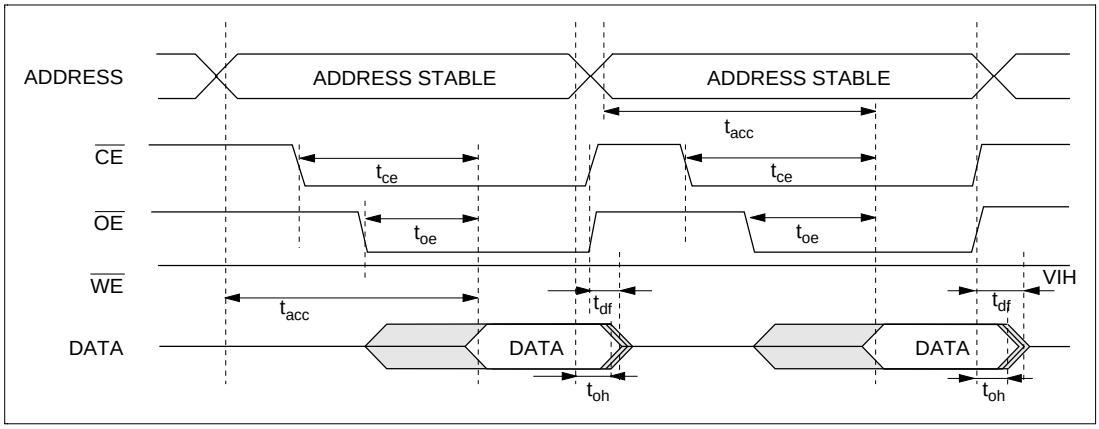


Figure 6.20 Timing Waveforms for $\overline{CE}$, $\overline{OE}$ Clocked Read

6.8.5 Auto-Program Mode

(1) AC Characteristics

Table 6.15 AC Characteristics in Auto-Program Mode

(Conditions: V_{CC} = 5.0 V ±10%, V_{SS} = 0 V, T_a = 25°C ±5°C)

Item	Symbol	Min	Max	Unit	Notes
Command write cycle	t_{nxtc}	20	—	µs	
$\overline{CE}$ 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_{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	
Write setup time	t_{pns}	100	—	ns	
Write end setup time	t_{pnh}	100	—	ns	

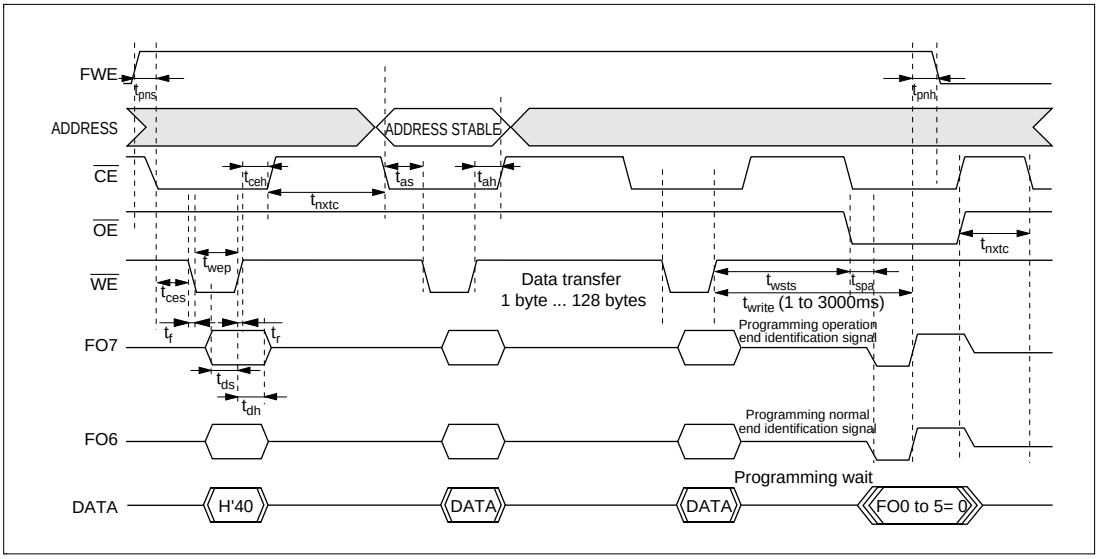


Figure 6.21 Auto-Program Mode Timing Waveforms

(2) Notes on Use of Auto-Program Mode

(a) In auto-program mode, 128 bytes are programmed simultaneously. This should be carried out by executing 128 consecutive byte transfers.

(b) A 128-byte data transfer is necessary even when programming fewer than 128 bytes. In this case, H'FF data must be written to the extra addresses.

(c) The lower 8 bits of the transfer address must be H'00 or H'80. If a value other than a valid address is input, processing will switch to a memory write operation but a write error will be flagged.

(d) Memory address transfer is performed in the second cycle (figure 6.20). Do not perform transfer after the second cycle.

(e) Do not perform a command write during a programming operation.

(f) Perform one auto-programming operation for a 128-byte block for each address. Characteristics cannot be guaranteed for two or more programming operations.

(g) Confirm normal end of auto-programming by checking FO_6. Alternatively, status read mode can also be used for this purpose (the FO_7 status polling pin is used to identify the end of an auto-program operation).

(h) Status polling FO_6 and FO_7 pin information is retained until the next command write. As long as the next command write has not been performed, reading is possible by enabling $\overline{CE}$ and $\overline{OE}$.

6.8.6 Auto-Erase Mode

(1) AC Characteristics

Table 6.16 AC Characteristics in Auto-Erase Mode

(Conditions: V_{CC} = 5.0 V ±10%, V_{SS} = 0 V, T_a = 25°C ±5°C)

Item	Symbol	Min	Max	Unit	Notes
Command write cycle	t_{nxtc}	20	—	µs	
$\overline{CE}$ 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	
Erase setup time	t_{ens}	100	—	ns	
Erase end setup time	t_{enh}	100	—	ns	

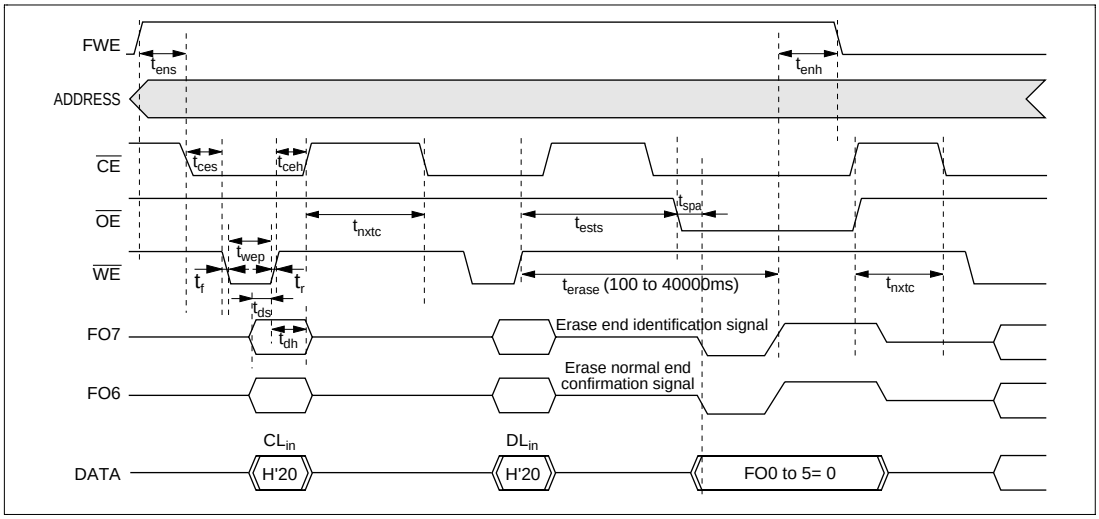


Figure 6.22 Auto-Erase Mode Timing Waveforms

(2) Notes on Use of Auto-Erase Mode

(a) Auto-erase mode supports only total memory erasing.

(b) Do not perform a command write during auto-erasing.

(c) Confirm normal end of auto-erasing by checking FO_6. Alternatively, status read mode can also be used for this purpose (the FO_7 status polling pin is used to identify the end of an auto-erase operation).

(d) Status polling FO_6 and FO_7 pin information is retained until the next command write. As long as the next command write has not been performed, reading is possible by enabling $\overline{CE}$ and $\overline{OE}$.

6.8.7 Status Read Mode

(1) Status read mode is used to identify what type of abnormal end has occurred. Use this mode when an abnormal end occurs in auto-program mode or auto-erase mode.

(2) The return code is retained until a command write for other than status read mode is performed.

Table 6.17 AC Characteristics in Status Read Mode

(Conditions: V_{CC} = 5.0 V ±10%, V_{SS} = 0 V, T_a = 25°C ±5°C)

Item	Symbol	Min	Max	Unit	Notes
Command write cycle	t_{nxtc}	20	—	µs	
$\overline{CE}$ 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	
$\overline{OE}$ 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	

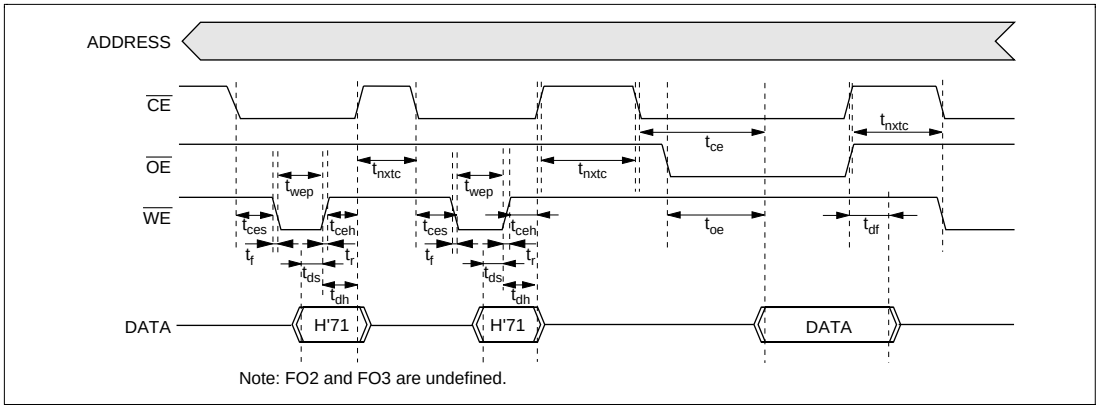


Figure 6.23 Status Read Mode Timing Waveforms

Table 6.18 Status Read Mode Return Codes

Pin Name	FO7	FO6	FO5	FO4	FO3	FO2	FO1	FO0
Attribute	Normal end identification	Command error	Programming error	Erase error	—	—	Programming or erase count exceeded	Valid address error
Initial value	0	0	0	0	0	0	0	0
Indications	Normal end: 0 Abnormal end: 1	Command error: 1 Otherwise: 0	Programming error: 1 Otherwise: 0	Erase error: 1 Otherwise: 0	—	—	Count exceeded: 1 Otherwise: 0	Valid address error: 1 Otherwise: 0

Note: FO_2 and FO_3 are undefined.

6.8.8 Status Polling

(1) The FO_7 status polling flag indicates the operating status in auto-program or auto-erase mode.

(2) The FO_6 status polling flag indicates a normal or abnormal end in auto-program or auto-erase mode.

Table 6.19 Status Polling Output Truth Table

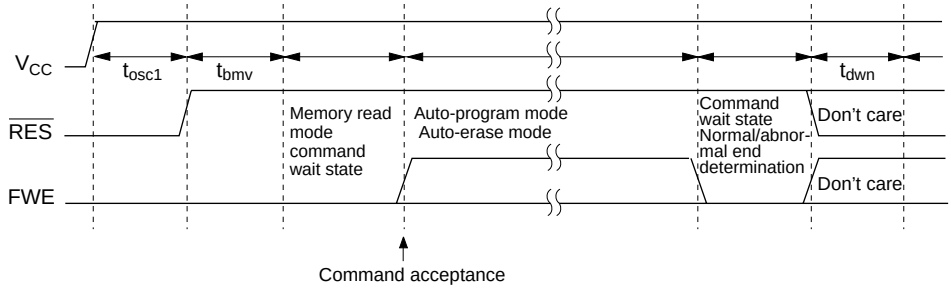
Pin Names	Internal Operation in Progress	Abnormal End	Normal Status Indication	Normal End
FO7	0	1	0	1
FO6	0	0	1	1
FO0 to FO5	0	0	0	0

6.8.9 Writer Mode Transition Time

Commands cannot be accepted during the oscillation stabilization period or the Writer mode setup period. A transition is made to memory read mode after the Writer mode setup time.

Table 6.20 Stipulated Transition Times to Command Wait State

Item	Symbol	Min	Max	Unit	Notes
Standby release (oscillation stabilization time)	t_{osc1}	40	—	ms	
Writer mode setup time	t_{bmv}	10	—	ms	
V_{CC} hold time	t_{dwn}	0	—	ms	



Note: When not in auto-program mode or auto-erase mode, the FWE input pin should be driven low.

Figure 6.24 Oscillation Stabilization Time, Writer Mode Setup, and Power-Down Sequence

6.8.10 Notes on Memory Programming

(1) When programming addresses which have previously been programmed, carry out auto-erasing before auto-programming.

(2) When performing programming using a PROM programmer on a chip that has been programmed/erased in an on-board programming mode, auto-erasing is recommended before carrying out auto-programming.

Notes: 1. The memory is initially in the erased state when the device is shipped by Hitachi. For other chips for which the erasure history is unknown, it is recommended that auto-erasing be executed to check and supplement the initialization (erase) level.

2. Auto-programming should be performed once only on a particular address block.

6.9 Flash Memory Programming and Erasing Precautions

Precautions concerning the use of on-board programming mode and Writer mode are summarized below.

(1) Use the specified voltages and timing for programming and erasing.

Applying a voltage in excess of the rating can permanently damage the device. Use a PROM programmer that supports the Hitachi microcomputer device type with 64-kbyte on-chip flash memory.

Do not select the HN28F101 setting for the PROM programmer, and only use the specified socket adapter. Incorrect use may damage the device.

(2) Powering on and off

Do not apply a high level to the FWE pin until V_{CC} has stabilized. Also, drive the FWE pin low before turning off V_{CC}.

When applying or disconnecting V_{CC}, fix the FWE pin low and place the flash memory in the hardware protection state.

The power-on and power-off timing requirements should also be satisfied in the event of a power failure and subsequent recovery. Failure to do so may result in overprogramming or overerasing due to MCU runaway, and loss of normal memory cell operation.

(3) FWE application/disconnection

FWE application should be carried out when MCU operation is in a stable condition. If MCU operation is not stable, fix the FWE pin low and set the protection state.

The following points must be observed concerning FWE application and disconnection to prevent unintentional programming or erasing of flash memory:

(a) Apply FWE when the V_{CC} voltage has stabilized within its rated voltage range.

(b) Apply FWE when oscillation has stabilized (after the elapse of the oscillation stabilization time).

(c) In boot mode, apply and disconnect FWE during a reset.

(d) In user program mode, FWE can be switched between high and low level regardless of the reset state. FWE input can also be switched during program execution in flash memory.

(e) Do not apply FWE if program runaway has occurred.

(f) Disconnect FWE only when the SWE, ESU, PSU, EV, PV, P, and E bits in FLMCR1 and FLMCR2 are cleared.

Make sure that the SWE, ESU, PSU, EV, PV, P, and E bits are not set by mistake when applying or disconnecting FWE.

(4) Do not apply a constant high level to the FWE pin

To prevent erroneous programming or erasing due to program runaway, etc., apply a high level to the FWE pin only when programming or erasing flash memory. A system configuration in which a high level is constantly applied to the FWE pin should be avoided. Also, while a high level is applied to the FWE pin, the watchdog timer should be activated to prevent overprogramming or overerasing due to program runaway, etc.

(5) Use the recommended algorithm when programming and erasing flash memory

The recommended algorithm enables programming and erasing to be carried out without subjecting the device to voltage stress or sacrificing program data reliability. When setting the P or E bit in FLMCR1, the watchdog timer should be set beforehand as a precaution against program runaway, etc.

(6) Do not set or clear the SWE bit during program execution in flash memory

Clear the SWE bit before executing a program or reading data in flash memory. When the SWE bit is set, data in flash memory can be rewritten, but flash memory should only be accessed for verify operations (verification during programming/erasing).

(7) Do not use interrupts while flash memory is being programmed or erased

All interrupt requests should be disabled during FWE application to give priority to program/erase operations.

(8) Do not perform additional programming. Erase the memory before reprogramming.

In on-board programming, perform only one programming operation on a 32-byte programming unit block. In Writer mode, perform only one programming operation on a 128-byte programming unit block. Programming should be carried out with the entire programming unit block erased.

(9) Before programming, check that the chip is correctly mounted in the PROM programmer.

Overcurrent damage to the device can result if the index marks on the PROM programmer socket, socket adapter, and chip are not correctly aligned.

(10) Do not touch the socket adapter or chip during programming.

Touching either of these can cause contact faults and write errors.

6.10 Notes when Converting the F-ZTAT Application Software to the Mask-ROM Versions

Please note the following when converting the F-ZTAT application software to the mask-ROM versions.

The values read from the internal registers for the flash ROM or the mask-ROM version and F-ZTAT version differ as follows.

		Status	
Register	**Bit**	**F-ZTAT Version**	**Mask-ROM Version**
FLMCR1	FWE	0: Application software running 1: Programming	0: Application software running 1: (Not read)

Note: This difference applies to all the F-ZTAT versions and all the mask-ROM versions that have different ROM size.

Section 7 RAM

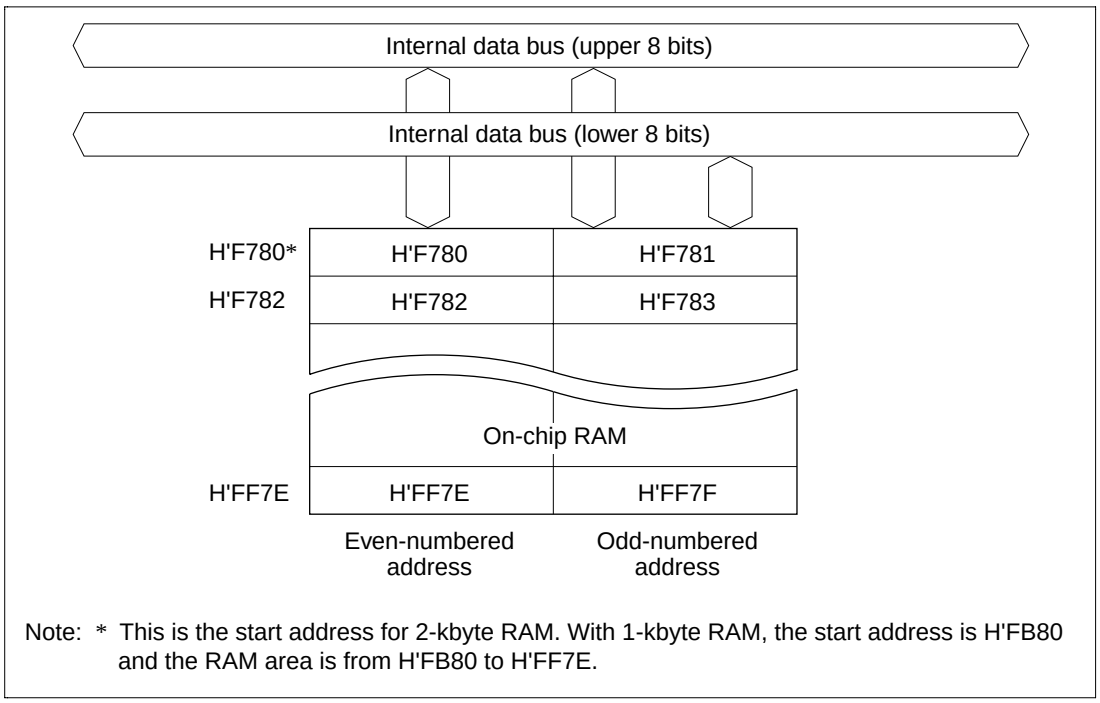
7.1 Overview

The H8/3857 Series and the H8/3854 flash memory version have 2 kbytes of high-speed on-chip static RAM, and the H8/3854 Series mask ROM version has 1 kbyte. The RAM is connected to the CPU by a 16-bit data bus, allowing high-speed 2-state access for both byte data and word data.

Note that the H8/3854 flash memory and mask ROM versions have different ROM and RAM sizes.

7.1.1 Block Diagram

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



Note: * This is the start address for 2-kbyte RAM. With 1-kbyte RAM, the start address is H'FB80 and the RAM area is from H'FB80 to H'FF7E.

Figure 7.1 RAM Block Diagram

Section 8 I/O Ports

8.1 Overview

The H8/3857 Series is provided with four 8-bit I/O ports, one 3-bit I/O port, one 8-bit input-only port, and one 1-bit input-only port. The H8/3854 Series is provided with two 8-bit I/O ports, one 3-bit I/O port, one 5-bit I/O port, one 4-bit input-only port, and one 1-bit input-only port. In addition, both series have an I/O port capable of interfacing with the on-chip LCD controller.

H8/3857 Series port functions are listed in table 8.1 (a), and H8/3854 Series port functions in table 8.1 (b).

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

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

Table 8.1 (a) H8/3857 Series Port Functions

Port	Description	Pins	Other Functions	Function Switching Register
Port 1	• 8-bit I/O port • Input pull-up MOS option	$P1_7$ to $P1_5$/ $\overline{IRQ_3}$ to $\overline{IRQ_1}$/ TMIF, TMIC, TMIB	External interrupts 3 to 1 Timer event input TMIF, TMIC, TMIB	PMR1 TCRF, TMC, TMB
		$P1_4$/PWM	14-bit PWM output	PMR1
		$P1_3$	None	
		$P1_2$, $P1_1$/ TMOFH, TMOFL	Timer F output compare	PMR1
		$P1_0$/TMOW	Timer A clock output	PMR1
Port 2	• 8-bit I/O port	$P2_7$ to $P2_2$	None	
	• Open drain output option	$P2_1$/UD	Timer C count-up/down selection	PMR2
	• High-current port	$P2_0$/$\overline{IRQ_4}$/ $\overline{ADTRG}$	External interrupt 4 and A/D converter external trigger	PMR2 AMR
Port 3	• 8-bit I/O port	$P3_7$ to $P3_3$	None	
	• Input pull-up MOS option • High-current port	$P3_2$/SO_1 $P3_1$/SI_1 $P3_0$/SCK_1	SCI1 data output (SO_1), data input (SI_1), clock input/output (SCK_1)	PMR3
Port 4	• 1-bit input-only port	$P4_3$/$\overline{IRQ_0}$	External interrupt 0	PMR2
	• 3-bit I/O port	$P4_2$/TXD $P4_1$/RXD $P4_0$/SCK_3	SCI3 data output (TXD), data input (RXD), clock input/output (SCK_3)	SCR3 SMR3
Port 5	• 8-bit I/O port • Input pull-up MOS option	$P5_7$ to $P5_0$/ $\overline{WKP_7}$ to $\overline{WKP_0}$	• Wakeup input ($\overline{WKP_7}$ to $\overline{WKP_0}$)	PMR5
Port 9*	• 8-bit I/O port	$P9_7$ to $P9_0$	None	
Port A*	• 4-bit I/O port	PA_3 to PA_0	None	
Port B	• 8-bit input port	PB_7 to PB_0/ AN_7 to AN_0	A/D converter analog input	AMR

Note: * This I/O port is used to interface to the LCD controller.

Table 8.1 (b) H8/3854 Series Port Functions

Port	Description	Pins	Other Functions	Function Switching Register
Port 1	• 5-bit I/O port • Input pull-up MOS option	$P1_7$, $P1_5$/ $\overline{IRQ_3}$, $\overline{IRQ_1}$/ TMIF, TMIB	External interrupts 3, 1 Timer event input TMIF, TMIB	PMR1 TCRF, TMB
		$P1_2$, $P1_1$/ TMOFH, TMOFL	Timer F output compare	PMR1
		$P1_0$/TMOW	Timer A clock output	PMR1
Port 2	• 8-bit I/O port	$P2_7$ to $P2_1$	None	
	• Open drain output option • High-current port	$P2_0$/$\overline{IRQ_4}$/ $\overline{ADTRG}$	External interrupt 4 and A/D converter external trigger	PMR2 AMR
Port 4	• 1-bit input-only port	$P4_3$/$\overline{IRQ_0}$	External interrupt 0	PMR2
	• 3-bit I/O port	$P4_2$/TXD $P4_1$/RXD $P4_0$/SCK_3	SCI3 data output (TXD), data input (RXD), clock input/output (SCK_3)	SCR3 SMR3
Port 5	• 8-bit I/O port • Input pull-up MOS option	$P5_7$ to $P5_0$/ $\overline{WKP_7}$ to $\overline{WKP_0}$	• Wakeup input ($\overline{WKP_7}$ to $\overline{WKP_0}$)	PMR5
Port 9*	• 8-bit I/O port	$P9_7$ to $P9_0$	None	
Port A*	• 4-bit I/O port	PA_3 to PA_0	None	
Port B	• 4-bit input port	PB_7 to PB_4/ AN_7 to AN_4	A/D converter analog input	AMR

Note: * This I/O port is used to interface to the LCD controller.

8.2 Port 1

Some port 1 functions differ between the H8/3857 Series and the H8/3854 Series.

The $P1_6/\overline{IRQ}_2$/TMIC, $P1_4$/PWM, and $P1_3$ pins are provided only in the H8/3857 Series, and not in the H8/3854 Series.

8.2.1 Overview

Port 1 is an 8-bit I/O port. The H8/3857 Series port 1 pin configuration is shown in figure 8.1 (a), and the H8/3854 Series port 1 pin configuration in figure 8.1 (b).

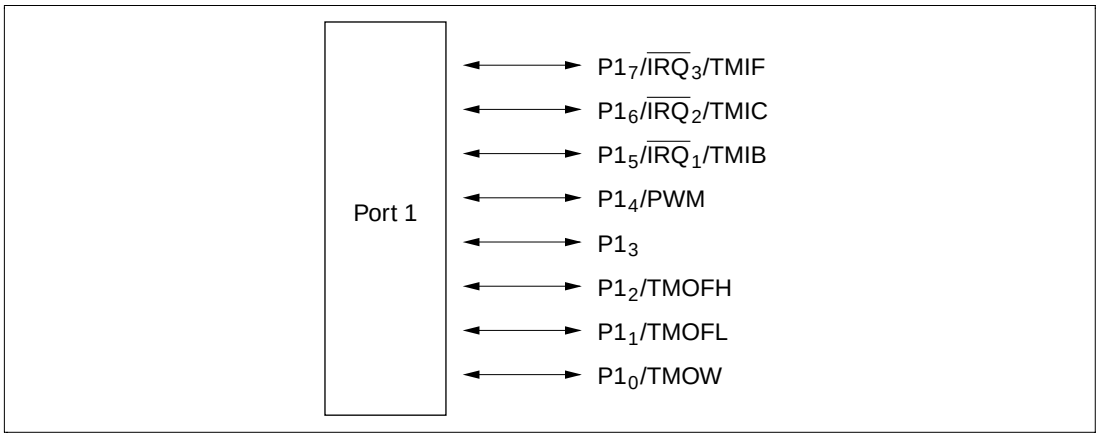


Figure 8.1 (a) H8/3857 Series Port 1 Pin Configuration

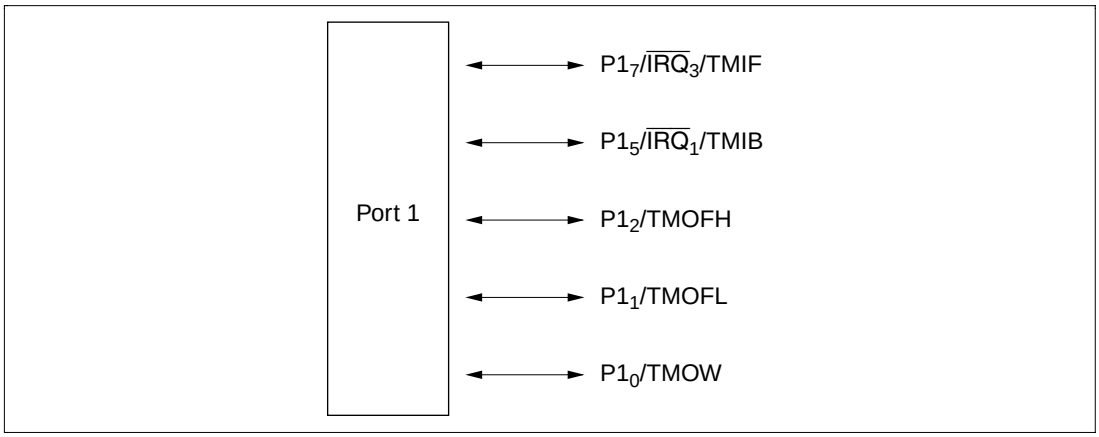


Figure 8.1 (b) H8/3854 Series Port 1 Pin Configuration

8.2.2 Register Configuration and Description

Table 8.2 shows the port 1 register configuration.

Table 8.2 Port 1 Registers

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

Port Data Register 1 (PDR1)

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

Upon reset, PDR1 is initialized to H'00 (H8/3857 Series) or H'58 (H8/3854 Series).

H8/3857 Series

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

H8/3854 Series

Bit	7	6	5	4	3	2	1	0
	$P1_7$	—	$P1_5$	—	—	$P1_2$	$P1_1$	$P1_0$
Initial value	0	1	0	1	1	0	0	0
Read/Write	R/W	—	R/W	—	—	R/W	R/W	R/W

In the H8/3854 Series, bits 6, 4, and 3 are reserved, and must always be set to 1.

Port Control Register 1 (PCR1)

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

Upon reset, PCR1 is initialized to H'00.

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

H8/3857 Series

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

H8/3854 Series

Bit	7	6	5	4	3	2	1	0
	$PCR1_7$	—	$PCR1_5$	—	—	$PCR1_2$	$PCR1_1$	$PCR1_0$
Initial value	0	0	0	0	0	0	0	0
Read/Write	W	—	W	—	—	W	W	W

In the H8/3854 Series, bits 6, 4, and 3 are reserved, and must always be set to 0.

Port Pull-Up Control Register 1 (PUCR1)

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

Upon reset, PUCR1 is initialized to H'00.

H8/3857 Series

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

H8/3854 Series

Bit	7	6	5	4	3	2	1	0
	$PUCR1_7$	—	$PUCR1_5$	—	—	$PUCR1_2$	$PUCR1_1$	$PUCR1_0$
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	—	R/W	—	—	R/W	R/W	R/W

In the H8/3854 Series, bits 6, 4, and 3 are reserved, and must always be set to 0.

Port Mode Register 1 (PMR1)

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

Upon reset, PMR1 is initialized to H'00.

H8/3857 Series

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

H8/3854 Series

Bit	7	6	5	4	3	2	1	0
	IRQ3	—	IRQ1	—	—	TMOFH	TMOFL	TMOW
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	—	R/W	—	—	R/W	R/W	R/W

In the H8/3854 Series, bits 6, 4, and 3 are reserved, and must always be set to 0.

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

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

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

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

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

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

In the H8/3854 Series, bit 6 is reserved, and must always be cleared to 0.

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

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

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

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

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

In the H8/3854 Series, bit 4 is reserved, and must always be cleared to 0.

Bit 3—Reserved Bit: Bit 3 is reserved; it should always be cleared to 0.

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

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

Bit 1—$P1_1$/TMOFL Pin Function Switch (TMOFL): This bit selects whether pin $P1_1$/TMOFL is used as $P1_1$ or as TMOFL.

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

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

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

8.2.3 Pin Functions

H8/3857 Series port 1 pin functions are shown in figure 8.3 (a), and H8/3854 Series port 1 pin functions in figure 8.3 (b).

Table 8.3 (a) H8/3857 Series Port 1 Pin Functions

<table>
<tr><th>Pin</th><th>Pin Functions and Selection Method</th></tr>
<tr><td>P1$_7$/$\overline{\text{IRQ}}_3$/TMIF</td><td>The pin function depends on bit IRQ3 in PMR1, bits CKSL2 to CKSL0 in TCRF, and bit PCR1$_7$ in PCR1.
<table>
<tr><td>IRQ3</td><td colspan="2">0</td><td colspan="2">1</td></tr>
<tr><td>PCR1$_7$</td><td>0</td><td>1</td><td colspan="2">*</td></tr>
<tr><td>CKSL2 to CKSL0</td><td colspan="2">***</td><td>Not 0**</td><td>0**</td></tr>
<tr><td>Pin function</td><td>P1$_7$ input pin</td><td>P1$_7$ output pin</td><td>$\overline{\text{IRQ}}_3$ input pin</td><td>$\overline{\text{IRQ}}_3$/TMIF input pin</td></tr>
</table>
Note: When using as TMIF input pin, clear bit IEN3 in IENR1 to 0, disabling IRQ$_3$ interrupts.</td></tr>
<tr><td>P1$_6$/$\overline{\text{IRQ}}_2$/TMIC</td><td>The pin function depends on bit IRQ2 in PMR1, bits TMC2 to TMC0 in TMC, and bit PCR1$_6$ in PCR1.
<table>
<tr><td>IRQ2</td><td colspan="2">0</td><td colspan="2">1</td></tr>
<tr><td>PCR1$_6$</td><td>0</td><td>1</td><td colspan="2">*</td></tr>
<tr><td>TMC2 to TMC0</td><td colspan="2">***</td><td>Not 111</td><td>111</td></tr>
<tr><td>Pin function</td><td>P1$_6$ input pin</td><td>P1$_6$ output pin</td><td>$\overline{\text{IRQ}}_2$ input pin</td><td>$\overline{\text{IRQ}}_2$/TMIC input pin</td></tr>
</table>
Note: When using as TMIC input pin, clear bit IEN2 in IENR1 to 0, disabling IRQ$_2$ interrupts.</td></tr>
<tr><td>P1$_5$/$\overline{\text{IRQ}}_1$/TMIB</td><td>The pin function depends on bit IRQ1 in PMR1, bits TMB2 to TMB0 in TMB, and bit PCR1$_5$ in PCR1.
<table>
<tr><td>IRQ1</td><td colspan="2">0</td><td colspan="2">1</td></tr>
<tr><td>PCR1$_5$</td><td>0</td><td>1</td><td colspan="2">*</td></tr>
<tr><td>TMB2 to TMB0</td><td colspan="2">***</td><td>Not 111</td><td>111</td></tr>
<tr><td>Pin function</td><td>P1$_5$ input pin</td><td>P1$_5$ output pin</td><td>$\overline{\text{IRQ}}_1$ input pin</td><td>$\overline{\text{IRQ}}_1$/TMIB input pin</td></tr>
</table>
Note: When using as TMIB input pin, clear bit IEN1 in IENR1 to 0, disabling IRQ1 interrupts.</td></tr>
</table>

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

Note: * Don’t care

Table 8.3 (b) H8/3854 Series Port 1 Pin Functions

Pin	Pin Functions and Selection Method
$P1_7$/$\overline{IRQ_3}$/TMIF	The pin function depends on bit IRQ3 in PMR1, bits CKSL2 to CKSL0 in TCRF, and bit $PCR1_7$ in PCR1.

IRQ3	0		1	
$PCR1_7$	0	1	*	
CKSL2 to CKSL0	***		Not 0**	0**
Pin function	$P1_7$ input pin	$P1_7$ output pin	$\overline{IRQ_3}$ input pin	$\overline{IRQ_3}$/TMIF input pin

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

$P1_5$/$\overline{IRQ_1}$/TMIB — The pin function depends on bit IRQ1 in PMR1, bits TMB2 to TMB0 in TMB, and bit $PCR1_5$ in PCR1.

IRQ1	0		1	
$PCR1_5$	0	1	*	
TMB2 to TMB0	***		Not 111	111
Pin function	$P1_5$ input pin	$P1_5$ output pin	$\overline{IRQ_1}$ input pin	$\overline{IRQ_1}$/TMIB input pin

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

$P1_2$/TMOFH — The pin function depends on bit TMOFH in PMR1 and bit $PCR1_2$ in PCR1.

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

$P1_1$/TMOFL — The pin function depends on bit TMOFL in PMR1 and bit $PCR1_1$ in PCR1.

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

$P1_0$/TMOW — The pin function depends on bit TMOW in PMR1 and bit $PCR1_0$ in PCR1.

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

Note: * Don't care

8.2.4 Pin States

H8/3857 Series port 1 pin states in each operating mode are shown in table 8.4 (a), and H8/3854 Series port 1 pin states in each operating mode in table 8.4 (b).

Table 8.4 (a) H8/3857 Series Port 1 Pin States

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

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

Table 8.4 (b) H8/3854 Series Port 1 Pin States

Pins	Reset	Sleep	Subsleep	Standby	Watch	Subactive	Active
$P1_7$/$\overline{IRQ_3}$/TMIF	High-impedance	Retains previous state	Retains previous state	High-impedance*	Retains previous state	Functional	Functional
$P1_5$/$\overline{IRQ_1}$/TMIB							
$P1_2$/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. When a PCR1 bit is cleared to 0, setting the corresponding PUCR1 bit to 1 turns on the MOS input pull-up for that pin. The MOS input pull-up function is in the off state after a reset.

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

Note: * Don't care
H8/3857 Series: n = 7 to 0
H8/3854 Series: n = 7, 5, 2 to 0

8.3 Port 2

Some port 2 functions differ between the H8/3857 Series and the H8/3854 Series.

The UD function multiplexed with the $P1_2$ pin is provided only in the H8/3857 Series, and not in the H8/3854 Series.

POF1 in PMR2 is also a function of the H8/3857 Series only.

8.3.1 Overview

Port 2 is an 8-bit I/O port. The H8/3857 Series port 2 pin configuration is shown in figure 8.2 (a), and the H8/3854 Series port 2 pin configuration in figure 8.2 (b).

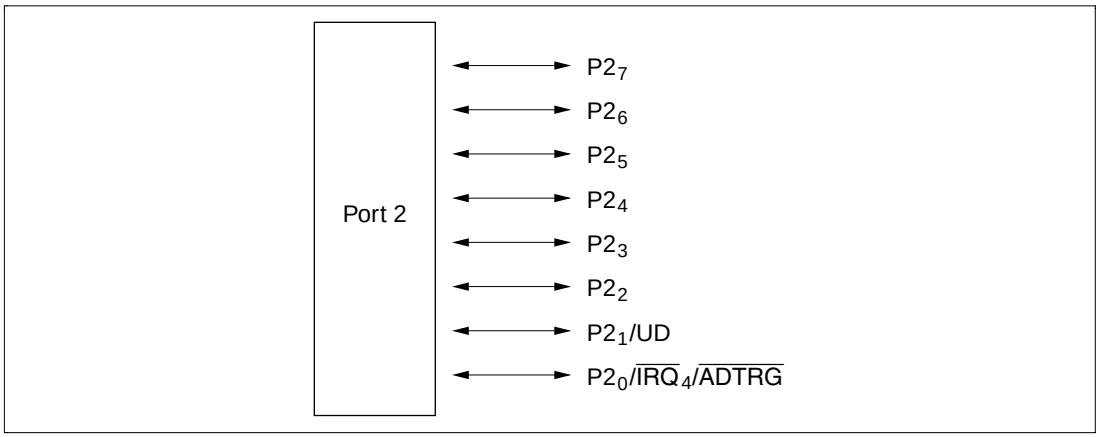


Figure 8.2 (a) H8/3857 Series Port 2 Pin Configuration

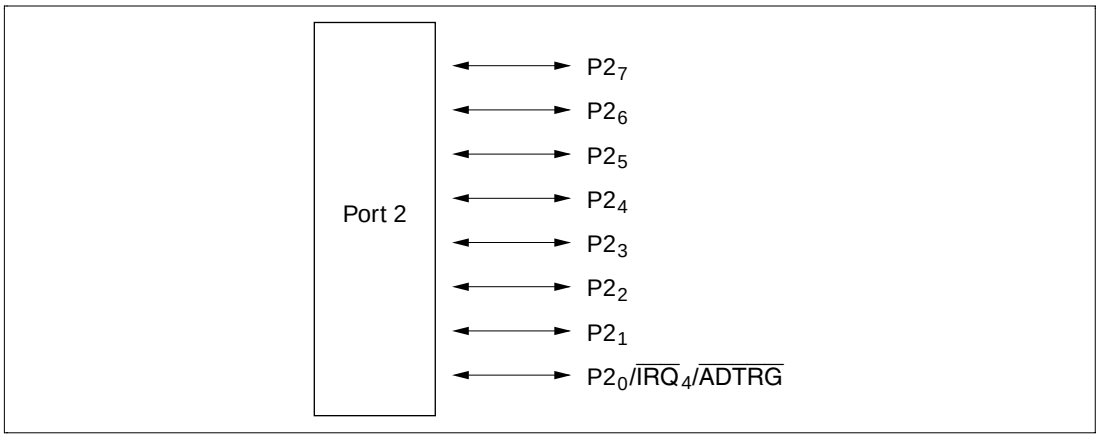


Figure 8.2 (b) H8/3854 Series Port 2 Pin Configuration

8.3.2 Register Configuration and Description

Table 8.5 shows the port 2 register configuration.

Table 8.5 Port 2 Registers

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

Port Data Register 2 (PDR2)

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

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

Upon reset, PDR2 is initialized to H'00.

Port Control Register 2 (PCR2)

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

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

Upon reset, PCR2 is initialized to H'00.

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

Port Mode Register 2 (PMR2)

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

Upon reset, PMR2 is initialized to H'C0.

Note: * $P2_1$ pin function switching and the $P3_2/SO_1$ pin are H8/3857 Series functions only, and are not provided in the H8/3854 Series.

H8/3857 Series

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

H8/3854 Series

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	IRQ0	—	—	IRQ4
Initial value	1	1	0	0	0	0	0	0
Read/Write	—	—	—	—	R/W	—	—	R/W

In the H8/3854 Series, bits 2 and 1 are reserved, and must always be cleared to 0.

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

Bits 5 and 4—Reserved Bits: Bits 5 and 4 are reserved; they should always be cleared to 0.

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

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

Note: Rising or falling edge sensing can be selected for the $\overline{IRQ_0}$ pin.

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

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

In the H8/3854 Series, bit 2 is reserved, and must always be cleared to 0.

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

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

In the H8/3854 Series, bit 1 is reserved, and must always be cleared to 0.

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

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

Note: Rising or falling edge sensing can be selected for the $\overline{IRQ_4}$ pin.
See 12.3.2, Start of A/D Conversion by External Trigger Input, for the $\overline{ADTRG}$ pin setting.

Port Mode Register 4 (PMR4)

Bit	7	6	5	4	3	2	1	0
	NMOD7	NMOD6	NMOD5	NMOD4	NMOD3	NMOD2	NMOD1	NMOD0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Upon reset, PMR4 is initialized to H'00.

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

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

(n = 7 to 0)

8.3.3 Pin Functions

H8/3857 Series port 2 pin functions are shown in figure 8.6 (a), and H8/3854 Series port 2 pin functions in figure 8.6 (b).

Table 8.6 (a) H8/3857 Series Port 2 Pin Functions

<table>
<tr><th>Pin</th><th>Pin Functions and Selection Method</th></tr>
<tr><td>$P2_7$ to $P2_2$</td><td>Input or output is selected as follows by the bit settings in PCR2.
(n = 7 to 2)
<table>
<tr><td>PCR2n</td><td>0</td><td>1</td></tr>
<tr><td>Pin function</td><td>$P2_n$ input pin</td><td>$P2_n$ output pin</td></tr>
</table></td></tr>
<tr><td>$P2_1$/UD</td><td>The pin function depends on bit UD in PMR2 and bit $PCR2_1$ in PCR2.
<table>
<tr><td>UD</td><td colspan="2">0</td><td>1</td></tr>
<tr><td>$PCR2_1$</td><td>0</td><td>1</td><td>*</td></tr>
<tr><td>Pin function</td><td>$P2_1$ input pin</td><td>$P2_1$ output pin</td><td>UD input pin</td></tr>
</table></td></tr>
<tr><td>$P2_0$/$\overline{IRQ_4}$/$\overline{ADTRG}$</td><td>The pin function depends on bit IRQ4 in PMR2, bit TRGE in AMR, and bit $PCR2_0$ in PCR2.
<table>
<tr><td>IRQ4</td><td colspan="2">0</td><td colspan="2">1</td></tr>
<tr><td>$PCR2_0$</td><td>0</td><td>1</td><td colspan="2">*</td></tr>
<tr><td>TRGE</td><td colspan="2">*</td><td>0</td><td>1</td></tr>
<tr><td>Pin function</td><td>$P2_0$ input pin</td><td>$P2_0$ output pin</td><td>$\overline{IRQ_4}$ input pin</td><td>$\overline{IRQ_4}$/$\overline{ADTRG}$ input pin</td></tr>
</table>
Note: When using as $\overline{ADTRG}$ input pin, clear bit IEN4 in IENR1 to 0, disabling IRQ_4 interrupts.</td></tr>
</table>

Note: * Don't care

Table 8.6 (b) H8/3854 Series Port 2 Pin Functions

Pin	Pin Functions and Selection Method
$P2_7$ to $P2_1$	Input or output is selected as follows by the bit settings in PCR2. (n = 7 to 1)

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

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

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

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

Note: * Don't care

8.3.4 Pin States

H8/3857 Series port 2 pin states in each operating mode are shown in table 8.7 (a), and H8/3854 Series port 2 pin states in each operating mode in table 8.7 (b).

Table 8.7 (a) H8/3857 Series Port 2 Pin States

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

Table 8.7 (b) H8/3854 Series Port 2 Pin States

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

8.4 Port 3 (H8/3857 Series Only)

Port 3 is a function of the H8/3857 Series only, and is not provided in the H8/3854 Series.

8.4.1 Overview

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

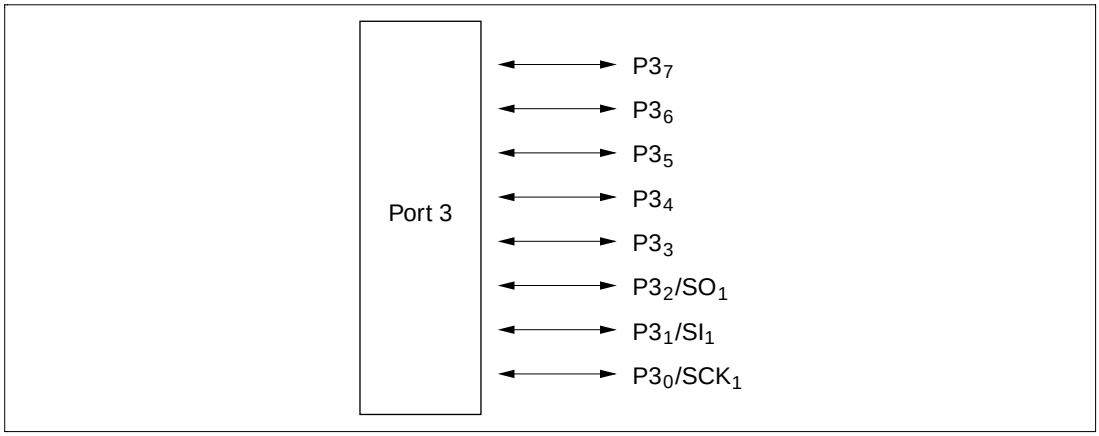


Figure 8.3 Port 3 Pin Configuration

8.4.2 Register Configuration and Description

Table 8.8 shows the port 3 register configuration.

Table 8.8 Port 3 Registers

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

Port Data Register 3 (PDR3)

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

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

Upon reset, PDR3 is initialized to H'00.

Port Control Register 3 (PCR3)

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

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

Upon reset, PCR3 is initialized to H'00.

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

Port Pull-Up Control Register 3 (PUCR3)

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

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

Upon reset, PUCR3 is initialized to H'00.

Port Mode Register 3 (PMR3)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	SO1	SI1	SCK1
Initial value	0	0	0	0	0	0	0	0
Read/Write	—	—	—	—	—	R/W	R/W	R/W

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

Upon reset, PMR3 is initialized to H'00.

Bits 7 to 3—Reserved Bits: Bits 7 to 3 are reserved; they should always be cleared to 0.

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

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

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

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

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

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

8.4.3 Pin Functions

Table 8.9 shows the port 3 pin functions.

Table 8.9 Port 3 Pin Functions

Pin	Pin Functions and Selection Method			
$P3_7$ to $P3_3$	The pin function depends on the corresponding bit in PCR3. (n=7 to 3)			
	$PCR3_n$	0	1	
	Pin function	$P3_n$ input pin	$P3_n$ output pin	
$P3_2/SO_1$	The pin function depends on bit SO1 in PMR3 and bit $PCR3_2$ in PCR3.			
	SO1	0	1	
	$PCR3_2$	0	1	*
	Pin function	$P3_2$ input pin	$P3_2$ output pin	SO_1 output pin
$P3_1/SI_1$	The pin function depends on bit SI1 in PMR3 and bit $PCR3_1$ in PCR3.			
	SI1	0	1	
	$PCR3_1$	0	1	*
	Pin function	$P3_1$ input pin	$P3_1$ output pin	SI_1 input pin
$P3_0/SCK_1$	The pin function depends on bit SCK1 in PMR3, bit CKS3 in SCR1, and bit $PCR3_0$ in PCR3.			
	SCK1	0	1	
	CKS3	*	0	1
	$PCR3_0$	0	1	*	*
	Pin function	$P3_0$ input pin	$P3_0$ output pin	SCK_1 output pin	SCK_1 input pin

Note: * Don't care

8.4.4 Pin States

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

Table 8.10 Port 3 Pin States

Pins	Reset	Sleep	Subsleep	Standby	Watch	Subactive	Active
$P3_7$ to $P3_3$ $P3_2/SO_1$ $P3_1/SI_1$ $P3_0/SCK_1$	High-impedance	Retains previous state	Retains previous state	High-impedance*	Retains previous state	Functional	Functional

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

8.4.5 MOS Input Pull-Up

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

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

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

8.5 Port 4

Port 4 functions are common to the H8/3857 Series and H8/3854 Series.

8.5.1 Overview

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

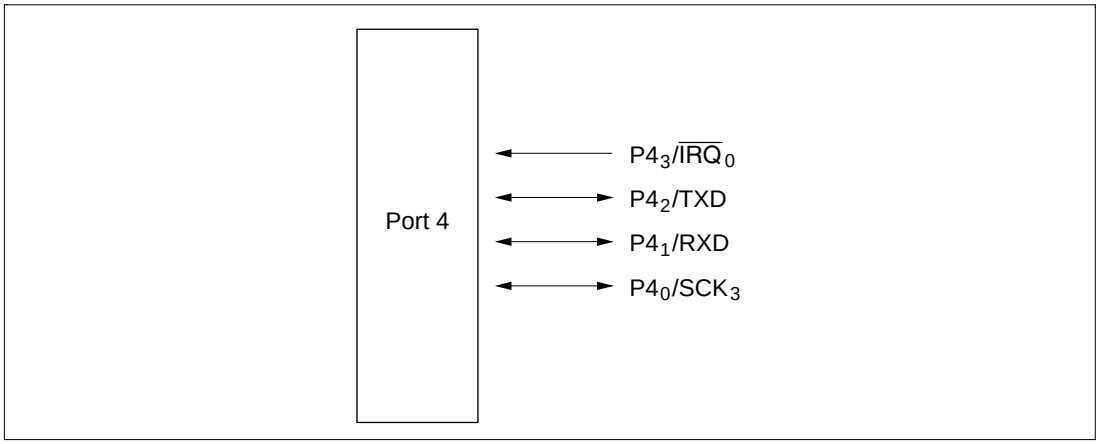


Figure 8.4 Port 4 Pin Configuration

8.5.2 Register Configuration and Description

Table 8.11 shows the port 4 register configuration.

Table 8.11 Port 4 Registers

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

Port Data Register 4 (PDR4)

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

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

The pin state is always read from bit 3 ($P4_3$).

Upon reset, PDR4 is initialized to H'F8.

Port Control Register 4 (PCR4)

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

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

Upon reset, PCR4 is initialized to H'F8.

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

8.5.3 Pin Functions

Table 8.12 shows the port 4 pin functions.

Table 8.12 Port 4 Pin Functions

Pin	Pin Functions and Selection Method			
$P4_3/\overline{IRQ_0}$	The pin function depends on the IRQ0 bit setting in PMR2.			
	IRQ0	0		1
	Pin function	$P4_3$ input pin		$\overline{IRQ_0}$ input pin
$P4_2$/TXD	The pin function depends on bit TE in SCR3 and bit $PCR4_2$ in PCR4.			
	TE	0		1
	$PCR4_2$	0	1	*
	Pin function	$P4_2$ input pin	$P4_2$ output pin	TXD output pin
$P4_1$/RXD	The pin function depends on bit RE in SCR3 and bit $PCR4_1$ in PCR4.			
	RE	0		1
	$PCR4_1$	0	1	*
	Pin function	$P4_1$ input pin	$P4_1$ output pin	RXD input pin
$P4_0/SCK_3$	The pin function depends on bits CKE1 and CKE0 in SCR3, bit COM in SMR, and bit $PCR4_0$ in PCR4.			

CKE1	0				1
CKE0	0			1	*
COM	0		1	*	*
$PCR4_0$	0	1	*		*
Pin function	$P4_0$ input pin	$P4_0$ output pin	SCK_3 output pin		SCK_3 input pin

Note: * Don't care

8.5.4 Pin States

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

Table 8.13 Port 4 Pin States

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

8.6 Port 5

Port 5 functions are common to the H8/3857 Series and H8/3854 Series.

8.6.1 Overview

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

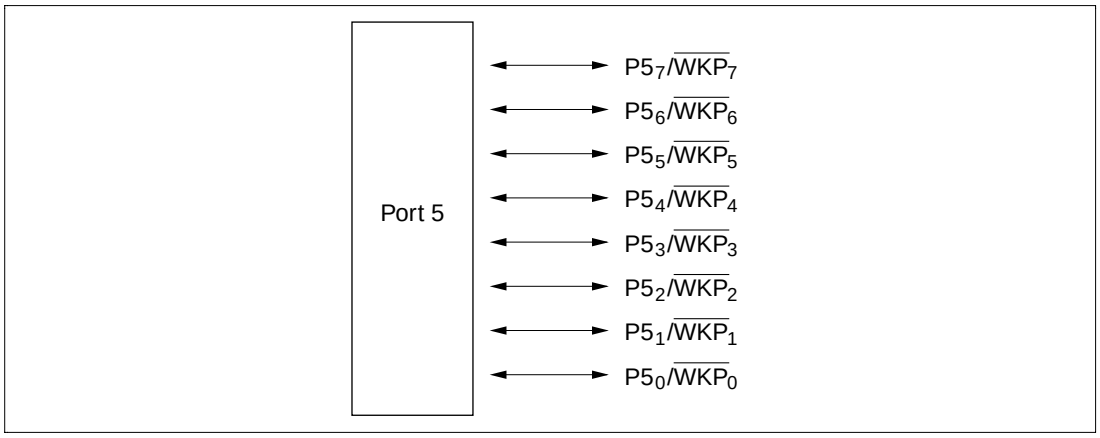


Figure 8.5 Port 5 Pin Configuration

8.6.2 Register Configuration and Description

Table 8.14 shows the port 5 register configuration.

Table 8.14 Port 5 Registers

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

Port Data Register 5 (PDR5)

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

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

Upon reset, PDR5 is initialized to H'00.

Port Control Register 5 (PCR5)

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

PCR5 is an 8-bit register for controlling whether each of the port 5 pins $P5_7$ to $P5_0$ functions as an input pin or output pin. Setting a PCR5 bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin.

Upon reset, PCR5 is initialized to H'00.

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

Port Pull-up Control Register 5 (PUCR5)

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

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

Upon reset, PUCR5 is initialized to H'00.

Port Mode Register 5 (PMR5)

Bit	7	6	5	4	3	2	1	0
	WKP_7	WKP_6	WKP_5	WKP_4	WKP_3	WKP_2	WKP_1	WKP_0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Upon reset, PMR5 is initialized to H'00.

Bit n—$P5_n/\overline{WKP}_n$ Pin Function Switch (WKPn): This bit selects whether pin $P5_n/\overline{WKP}_n$ is used as $P5_n$ or as $\overline{WKP}_n$.

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

(n = 7 to 0)

8.6.3 Pin Functions

Table 8.15 shows the port 5 pin functions.

Table 8.15 Port 5 Pin Functions

Pin	Pin Functions and Selection Method
$P5_7/\overline{WKP}_7$ to $P5_0/\overline{WKP}_0$	The pin function depends on bit WKPn in PMR5 and bit $PCR5_n$ in PCR5. (n = 7 to 0)

WKPn	0		1
$PCR5_n$	0	1	*
Pin function	$P5_n$ input pin	$P5_n$ output pin	$\overline{WKP}_n$ input pin

Note: * Don't care

8.6.4 Pin States

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

Table 8.16 Port 5 Pin States

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

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

8.6.5 MOS Input Pull-Up

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

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

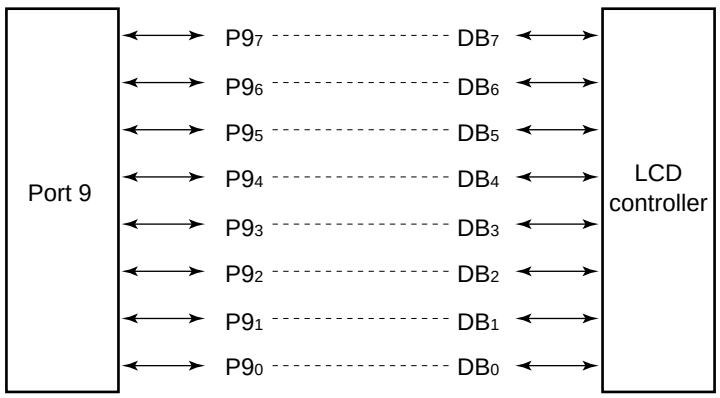
Note: * Don't care
n = 7 to 0

8.7 Port 9 [Chip-Internal I/O port]

Port 9 functions are common to the H8/3857 Series and H8/3854 Series.

8.7.1 Overview

Port 9 is an 8-bit I/O port that interfaces to the on-chip LCD controller. The port 9 pin configuration is shown in figure 8.6.



Note: Dotted lines indicate connections inside the chip.

Figure 8.6 Port 9 Pin Configuration

8.7.2 Register Configuration and Description

Table 8.17 shows the port 9 register configuration.

Table 8.17 Port 9 Registers

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

Port Data Register 9 (PDR9)

Bit	7	6	5	4	3	2	1	0
	$P9_7$	$P9_6$	$P9_5$	$P9_4$	$P9_3$	$P9_2$	$P9_1$	$P9_0$
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PDR9 is an 8-bit register that stores data for port 9 pins $P9_7$ to $P9_0$. If port 9 is read while PCR9 bits are set to 1, the values stored in PDR9 are read. If port 9 is read while PCR9 bits are cleared to 0, the pin states are read.

Upon reset, PDR9 is initialized to H'00.

Port Control Register 9 (PCR9)

Bit	7	6	5	4	3	2	1	0
	$PCR9_7$	$PCR9_6$	$PCR9_5$	$PCR9_4$	$PCR9_3$	$PCR9_2$	$PCR9_1$	$PCR9_0$
Initial value	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W

PCR9 is an 8-bit register for controlling whether each of the port 9 pins $P9_7$ to $P9_0$ functions as an input or output pin. Setting a PCR9 bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin.

Upon reset, PCR9 is initialized to H'00.

PCR9 is a write-only register. All bits are read as 1.

8.7.3 Pin Functions

Table 8.18 shows the port 9 pin functions.

Table 8.18 Port 9 Pin Functions

Pin	Pin Functions and Selection Method		
$P9_7$ to $P9_0$	The pin function depends on the corresponding bit in PCR9. (n=7 to 0)		
	$PCR9_n$	0	1
	Pin function	$P9_n$ input pin	$P9_n$ output pin

8.7.4 Pin States

Table 8.19 shows the port 9 pin states in each operating mode.

Table 8.19 Port 9 Pin States

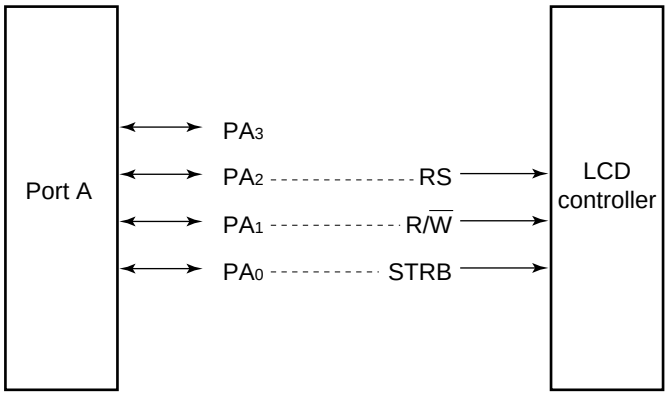
Pins	Reset	Sleep	Subsleep	Standby	Watch	Subactive	Active
$P9_7$ to $P9_0$	High-impedance	Retains previous state	Retains previous state	High-impedance	Retains previous state	Functional	Functional

8.8 Port A [Chip-Internal I/O port]

Port A functions are common to the H8/3857 Series and H8/3854 Series.

8.8.1 Overview

Port A is a 4-bit I/O port that interfaces to the on-chip LCD controller. The port A pin configuration is shown in figure 8.7.



Note: Dotted lines indicate connections inside the chip.

Figure 8.7 Port A Pin Configuration

8.8.2 Register Configuration and Description

Table 8.20 shows the port A register configuration.

Table 8.20 Port A Registers

Name	Abbrev.	R/W	Initial Value	Address
Port data register A	PDRA	R/W	H'F0	H'FFDD
Port control register A	PCRA	W	H'F0	H'FFED

Port Data Register A (PDRA)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	PA_3	PA_2	PA_1	PA_0
Initial value	1	1	1	1	0	0	0	0
Read/Write	—	—	—	—	R/W	R/W	R/W	R/W

PDRA is an 8-bit register that stores data for port A pins PA_3 to PA_0. If port A is read while PCRA bits are set to 1, the values stored in PDRA are read. If port A is read while PCRA bits are cleared to 0, the pin states are read.

Upon reset, PDRA is initialized to H'F0.

Port Control Register A (PCRA)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	$PCRA_3$	$PCRA_2$	$PCRA_1$	$PCRA_0$
Initial value	1	1	1	1	0	0	0	0
Read/Write	—	—	—	—	W	W	W	W

PCRA is an 8-bit register for controlling whether each of the port A pins PA_3 to PA_0 functions as an input or output pin. Setting a PCRA bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin.

Upon reset, PCRA is initialized to H'F0.

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

8.8.3 Pin Functions

Table 8.21 gives the port A pin functions.

Table 8.21 Port A Pin Functions

Pin	Pin Functions and Selection Method		
PA_3 to PA_0	The pin function depends on the corresponding bit in PCRA. (n=3 to 0)		
	$PCRA_n$	0	1
	Pin function	PA_n input pin	PA_n output pin

8.8.4 Pin States

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

Table 8.22 Port A Pin States

Pins	Reset	Sleep	Subsleep	Standby	Watch	Subactive	Active
PA_3 to PA_0	High-impedance	Retains previous state	Retains previous state	High-impedance	Retains previous state	Functional	Functional

8.9 Port B

Some port B functions differ between the H8/3857 Series and the H8/3854 Series.

Pins PB_3 to PB_0/AN_3 to AN_0 are provided only in the H8/3857 Series, and not in the H8/3854 Series.

8.9.1 Overview

Port B is an 8-bit input-only port. The H8/3857 Series port B pin configuration is shown in figure 8.8 (a), and the H8/3854 Series port B pin configuration in figure 8.8 (b).

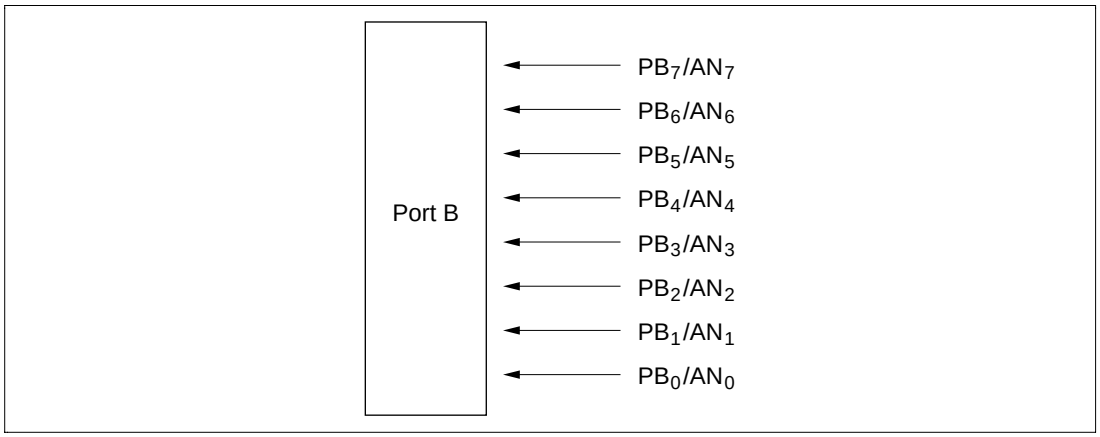


Figure 8.8 (a) H8/3857 Series Port B Pin Configuration

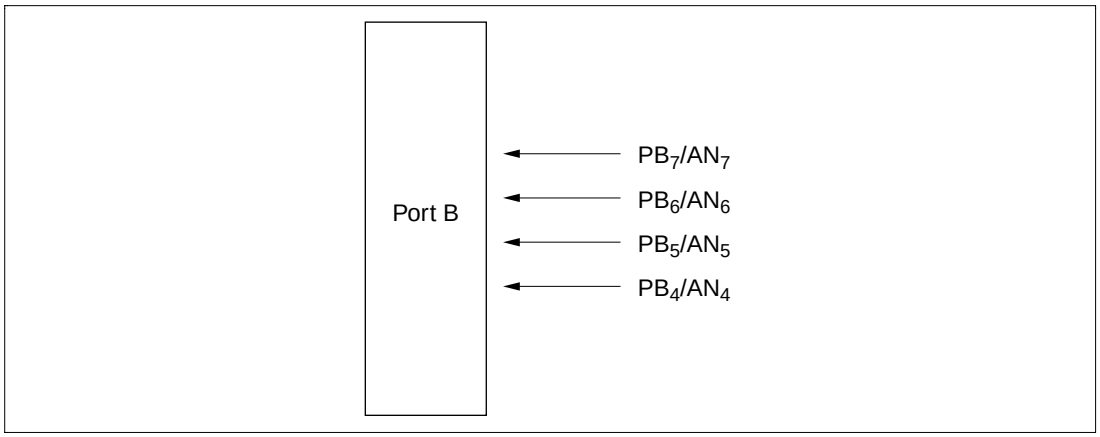


Figure 8.8 (b) H8/3854 Series Port B Pin Configuration

8.9.2 Register Configuration and Description

Table 8.23 shows the port B register configuration.

Table 8.23 Port B Register

Name	Abbrev.	R/W	Address
Port data register B	PDRB	R	H'FFDE

Port Data Register B (PDRB)

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

H8/3857 Series

Bit	7	6	5	4	3	2	1	0
	PB_7	PB_6	PB_5	PB_4	PB_3	PB_2	PB_1	PB_0
Read/Write	R	R	R	R	R	R	R	R

H8/3854 Series

Bit	7	6	5	4	3	2	1	0
	PB_7	PB_6	PB_5	PB_4	—	—	—	—
Read/Write	R	R	R	R	—	—	—	—

In the H8/3854 Series, bits 3 to 0 are reserved.

Section 9 Timers

9.1 Overview

The H8/3857 Series is provided with four timers (timers A, B, C, and F), and the H8/3854 Series with three (timers A, B, and F). The H8/3857F and H8/3854F also have an on-chip watchdog timer for flash memory programming control.

Table 9.1 outlines the functions of timers A, B, C, F, and the watchdog timer.

Table 9.1 Timer Functions

Name	Functions	Internal Clock	Event Input Pin	Waveform Output Pin	Remarks
Timer A	• 8-bit timer • Interval timer	ϕ/8 to ϕ/8192 (8 choices)	—	—	
	• 8-bit timer • Time base	ϕ_w/128 (choice of 4 overflow periods)	—	—	
	• 8-bit timer • Clock output	ϕ/4 to ϕ/32, ϕ_w/4 to ϕ_w/32 (8 choices)	—	TMOW	
Timer B	• 8-bit timer • Interval timer • Event counter	ϕ/4 to ϕ/8192 (7 choices)	TMIB	—	
Timer C*1	• 8-bit timer • Interval timer • Event counter • Choice of up- or down-counting	ϕ/4 to ϕ/8192, ϕ_w/4 (7 choices)	TMIC	—	Counting direction can be controlled by software or hardware
Timer F	• 16-bit timer • Event counter • Can be used as two independent 8-bit timers • Output compare	ϕ/2 to ϕ/32 (4 choices)	TMIF	TMOFL TMOFH	
Watchdog timer*2	• Generates reset signal on overflow of 8-bit counter	ϕ/64 to ϕ/8192 (8 choices)	—	—	Provided only in H8/3857F and H8/3854F

Notes: 1. Timer C is a function of the H8/3857 Series only, and is not provided in the H8/3854 Series.
2. The watchdog timer is used by the flash memory programming control program.

9.2 Timer A

9.2.1 Overview

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

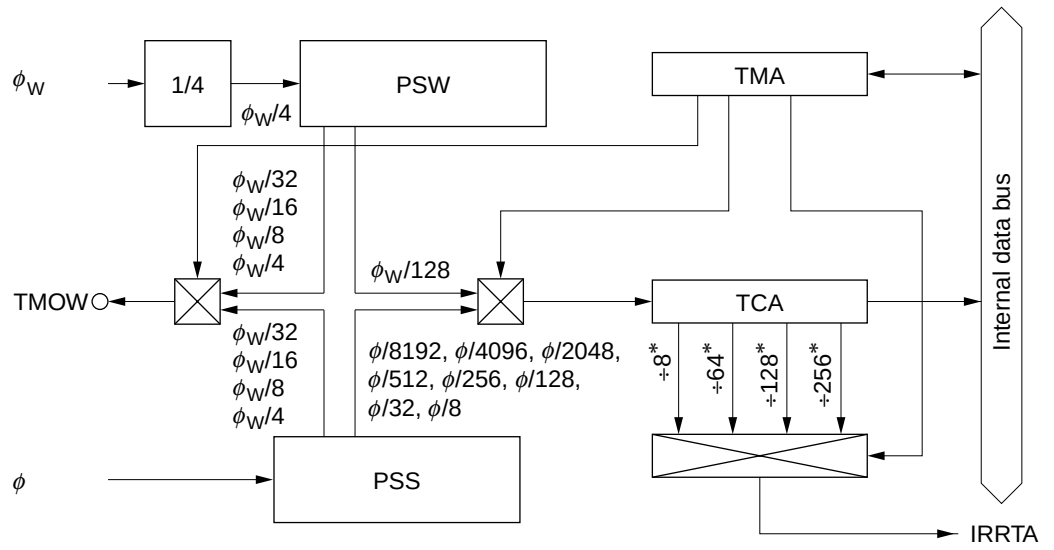
Features

Features of timer A are given below.

- Choice of eight internal clock sources (ϕ/8192, ϕ/4096, ϕ/2048, ϕ/512, ϕ/256, ϕ/128, ϕ/32, ϕ/8).
- Choice of four overflow periods (1 s, 0.5 s, 0.25 s, 31.25 ms) when timer A is used as a clock time base (using a 32.768 kHz crystal oscillator).
- An interrupt is requested when the counter overflows.
- Any of eight clock signals can be output from pin TMOW: 32.768 kHz divided by 32, 16, 8, or 4 (1 kHz, 2 kHz, 4 kHz, 8 kHz), or the system clock divided by 32, 16, 8, or 4.

Block Diagram

Figure 9.1 shows a block diagram of timer A.



Notation:
TMA: Timer mode register A
TCA: Timer counter A
IRRTA: Timer A overflow interrupt request flag
PSW: Prescaler W
PSS: Prescaler S

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

Figure 9.1 Block Diagram of Timer A

Pin Configuration

Table 9.2 shows the timer A pin configuration.

Table 9.2 Pin Configuration

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

Register Configuration

Table 9.3 shows the register configuration of timer A.

Table 9.3 Timer A Registers

Name	Abbrev.	R/W	Initial Value	Address
Timer mode register A	TMA	R/W	H'10	H'FFB0
Timer counter A	TCA	R	H'00	H'FFB1

9.2.2 Register Descriptions

Timer Mode Register A (TMA)

Bit	7	6	5	4	3	2	1	0
	TMA7	TMA6	TMA5	—	TMA3	TMA2	TMA1	TMA0
Initial value	0	0	0	1	0	0	0	0
Read/Write	R/W	R/W	R/W	—	R/W	R/W	R/W	R/W

TMA is an 8-bit read/write register for selecting the prescaler, input clock, and output clock.

Upon reset, TMA is initialized to H'10.

Bits 7 to 5—Clock Output Select (TMA7 to TMA5): Bits 7 to 5 choose which of eight clock signals is output at the TMOW pin. The system clock divided by 32, 16, 8, or 4 can be output in active mode and sleep mode. A 32.768 kHz signal divided by 32, 16, 8, or 4 can be output in active mode, sleep mode, and subactive mode.

Bit 7: TMA7	Bit 6: TMA6	Bit 5: TMA5	Clock Output	
0	0	0	ϕ/32	(initial value)
		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	

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

Bits 3 to 0—Internal Clock Select (TMA3 to TMA0): Bits 3 to 0 select the clock input to TCA.

				Description		
Bit 3: TMA3	**Bit 2: TMA2**	**Bit 1: TMA1**	**Bit 0: TMA0**	**Prescaler and Divider Ratio or Overflow Period**		**Function**
0	0	0	0	PSS, ϕ/8192	(initial value)	Interval timer
			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		Clock time base
			1	PSW, 0.5 s		
		1	0	PSW, 0.25 s		
			1	PSW, 0.03125 s		
	1	0	0	PSW and TCA are reset		
			1			
		1	0			
			1			

Timer Counter A (TCA)

Bit	7	6	5	4	3	2	1	0
	TCA7	TCA6	TCA5	TCA4	TCA3	TCA2	TCA1	TCA0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R	R

TCA is an 8-bit read-only up-counter, which is incremented by internal clock input. The clock source for input to this counter is selected by bits TMA3 to TMA0 in timer mode register A (TMA). TCA values can be read by the CPU in active mode, but cannot be read in subactive mode. When TCA overflows, the IRRTA bit in interrupt request register 1 (IRR1) is set to 1.

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

Upon reset, TCA is initialized to H'00.

9.2.3 Timer Operation

Interval Timer Operation: When bit TMA3 in timer mode register A (TMA) is cleared to 0, timer A functions as an 8-bit interval timer.

Upon reset, TCA is cleared to H'00 and bit TMA3 is cleared to 0, so up-counting and interval timing resume immediately. The clock input to timer A is selected by bits TMA2 to TMA0 in TMA; any of eight internal clock signals output by prescaler S can be selected.

After the count value in TCA reaches H'FF, the next clock signal input causes timer A to overflow, setting bit IRRTA to 1 in interrupt request register 1 (IRR1). If IENTA = 1 in interrupt enable register 1 (IENR1), a CPU interrupt is requested.*

At overflow, TCA returns to H'00 and starts counting up again. In this mode timer A functions as an interval timer that generates an overflow output at intervals of 256 input clock pulses.

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

Real-Time Clock Time Base Operation: When bit TMA3 in TMA is set to 1, timer A functions as a real-time clock time base by counting clock signals output by prescaler W.

The overflow period of timer A is set by bits TMA1 and TMA0 in TMA. A choice of four periods is available. In time base operation (TMA3 = 1), setting bit TMA2 to 1 clears both TCA and prescaler W to their initial values of H'00.

Clock Output: Setting bit TMOW in port mode register 1 (PMR1) to 1 causes a clock signal to be output at pin TMOW. Eight different clock output signals can be selected by means of bits TMA7 to TMA5 in TMA. The system clock divided by 32, 16, 8, or 4 can be output in active mode and sleep mode. A 32.768 kHz signal divided by 32, 16, 8, or 4 can be output in active mode, sleep mode, and subactive mode.

9.2.4 Timer A Operation States

Table 9.4 summarizes the timer A operation states.

Table 9.4 Timer A Operation States

Operation Mode		Reset	Active	Sleep	Watch	Sub-active	Sub-sleep	Standby
TCA	Interval	Reset	Functions	Functions	Halted	Halted	Halted	Halted
	Clock time base	Reset	Functions	Functions	Functions	Functions	Functions	Halted
TMA		Reset	Functions	Retained	Retained	Functions	Retained	Retained

Note: When real-time clock time base function is selected as the internal clock of TCA in active mode or sleep mode, the internal clock is not synchronous with the system clock, so it is synchronized by a synchronizing circuit. This may result in a maximum error of 1/ϕ (s) in the count cycle.

9.3 Timer B

9.3.1 Overview

Timer B is an 8-bit timer that increments each time a clock pulse is input. This timer has two operation modes, interval and auto reload.

Features

Features of timer B are given below.

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

Block Diagram

Figure 9.2 shows a block diagram of timer B.

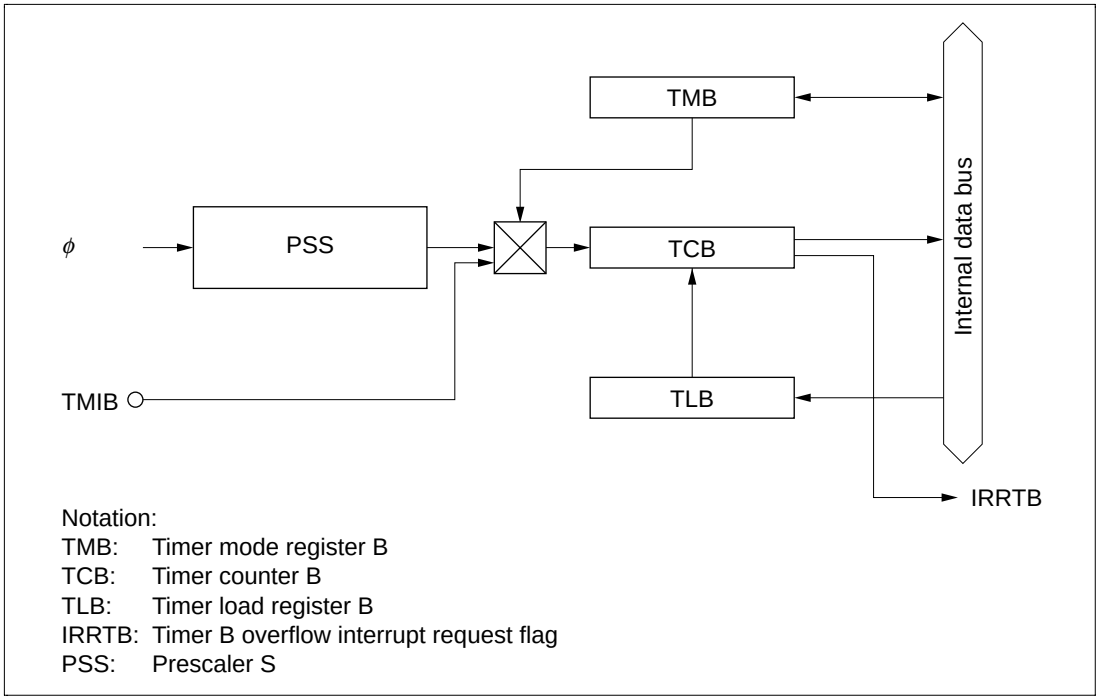


Figure 9.2 Block Diagram of Timer B

Pin Configuration

Table 9.5 shows the timer B pin configuration.

Table 9.5 Pin Configuration

Name	Abbrev.	I/O	Function
Timer B event input	TMIB	Input	Event input to TCB

Register Configuration

Table 9.6 shows the register configuration of timer B.

Table 9.6 Timer B Registers

Name	Abbrev.	R/W	Initial Value	Address
Timer mode register B	TMB	R/W	H'78	H'FFB2
Timer counter B	TCB	R	H'00	H'FFB3
Timer load register B	TLB	W	H'00	H'FFB3

9.3.2 Register Descriptions

Timer Mode Register B (TMB)

Bit	7	6	5	4	3	2	1	0
	TMB7	—	—	—	—	TMB2	TMB1	TMB0
Initial value	0	1	1	1	1	0	0	0
Read/Write	R/W	—	—	—	—	R/W	R/W	R/W

TMB is an 8-bit read/write register for selecting the auto-reload function and input clock.

Upon reset, TMB is initialized to H'78.

Bit 7—Auto-Reload Function Select (TMB7): Bit 7 selects whether timer B is used as an interval timer or auto-reload timer.

Bit 7: TMB7	Description	
0	Interval timer function selected	(initial value)
1	Auto-reload function selected	

Bits 6 to 3—Reserved Bits: Bits 6 to 3 are reserved; they always read 1, and cannot be modified.

Bits 2 to 0—Clock Select (TMB2 to TMB0): Bits 2 to 0 select the clock input to TCB. For external event counting, either the rising or falling edge can be selected.

Bit 2: TMB2	Bit 1: TMB1	Bit 0: TMB0	Description	
0	0	0	Internal clock: ϕ/8192	(initial value)
		1	Internal clock: ϕ/2048	
	1	0	Internal clock: ϕ/512	
		1	Internal clock: ϕ/256	
1	0	0	Internal clock: ϕ/64	
		1	Internal clock: ϕ/16	
	1	0	Internal clock: ϕ/4	
		1	External event (TMIB): rising or falling edge*	

Note: * The edge of the external event signal is selected by bit IEG1 in the IRQ edge select register (IEGR). See 3.3.2, Interrupt Control Registers, for details on the IRQ edge select register. Be sure to set bit IRQ1 in port mode register 1 (PMR1) to 1 before setting bits TMB2 to TMB0 to 111.

Timer Counter B (TCB)

Bit	7	6	5	4	3	2	1	0
	TCB7	TCB6	TCB5	TCB4	TCB3	TCB2	TCB1	TCB0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R	R

TCB is an 8-bit read-only up-counter, which is incremented by internal clock or external event input. The clock source for input to this counter is selected by bits TMB2 to TMB0 in timer mode register B (TMB). TCB values can be read by the CPU at any time.

When TCB overflows from H'FF to H'00 or to the value set in TLB, the IRRTB bit in interrupt request register 2 (IRR2) is set to 1.

TCB is allocated to the same address as timer load register B (TLB).

Upon reset, TCB is initialized to H'00.

Timer Load Register B (TLB)

Bit	7	6	5	4	3	2	1	0
	TLB7	TLB6	TLB5	TLB4	TLB3	TLB2	TLB1	TLB0
Initial value	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W

TLB is an 8-bit write-only register for setting the reload value of timer counter B.

When a reload value is set in TLB, the same value is loaded into timer counter B (TCB) as well, and TCB starts counting up from that value. When TCB overflows during operation in auto-reload mode, the TLB value is loaded into TCB. Accordingly, overflow periods can be set within the range of 1 to 256 input clocks.

The same address is allocated to TLB as to TCB.

Upon reset, TLB is initialized to H'00.

9.3.3 Timer Operation

Interval timer Operation: When bit TMB7 in timer mode register B (TMB) is cleared to 0, timer B functions as an 8-bit interval timer.

Upon reset, TCB is cleared to H'00 and bit TMB7 is cleared to 0, so up-counting and interval timing resume immediately. The clock input to timer B is selected from seven internal clock signals output by prescaler S, or an external clock input at pin TMIB. The selection is made by bits TMB2 to TMB0 of TMB.

After the count value in TCB reaches H'FF, the next clock signal input causes timer B to overflow, setting bit IRRTB to 1 in interrupt request register 2 (IRR2). If IENTB = 1 in interrupt enable register 2 (IENR2), a CPU interrupt is requested.*

At overflow, TCB returns to H'00 and starts counting up again.

During interval timer operation (TMB7 = 0), when a value is set in timer load register B (TLB), the same value is set in TCB.

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

Auto-Reload Timer Operation: Setting bit TMB7 in TMB to 1 causes timer B to function as an 8-bit auto-reload timer. When a reload value is set in TLB, the same value is loaded into TCB, becoming the value from which TCB starts its count.

After the count value in TCB reaches H'FF, the next clock signal input causes timer B to overflow. The TLB value is then loaded into TCB, and the count continues from that value. The overflow period can be set within a range from 1 to 256 input clocks, depending on the TLB value.

The clock sources and interrupts in auto-reload mode are the same as in interval mode.

In auto-reload mode (TMB7 = 1), when a new value is set in TLB, the TLB value is also set in TCB.

Event Counter Operation: Timer B can operate as an event counter, counting rising or falling edges of an external event signal input at pin TMIB. External event counting is selected by setting bits TMB2 to TMB0 in timer mode register B to all 1s (111).

When timer B is used to count external event input, bit IRQ1 in port mode register 1 (PMR1) should be set to 1, and bit IEN1 in interrupt enable register 1 (IENR1) should be cleared to 0 to disable IRQ_1 interrupt requests.

9.3.4 Timer B Operation States

Table 9.7 summarizes the timer B operation states.

Table 9.7 Timer B Operation States

Operation Mode		Reset	Active	Sleep	Watch	Sub-active	Sub-sleep	Standby
TCB	Interval	Reset	Functions	Functions	Halted	Halted	Halted	Halted
	Auto reload	Reset	Functions	Functions	Halted	Halted	Halted	Halted
TMB		Reset	Functions	Retained	Retained	Retained	Retained	Retained

9.4 Timer C (H8/3857 Series Only)

9.4.1 Overview

Timer C is an 8-bit timer that increments or decrements each time a clock pulse is input. This timer has two operation modes, interval and auto reload.

Timer C is a function of the H8/3857 Series only, and is not provided in the H8/3854 Series.

Features

The main 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, ϕ_W/4) or an external clock (can be used to count external events).
- An interrupt is requested when the counter overflows.
- Can be switched between up- and down-counting by software or hardware.
- When ϕ_W/4 is selected as the internal clock source, or when an external clock is selected, timer C can function in subactive mode and subsleep mode.

Block Diagram

Figure 9.3 shows a block diagram of timer C.

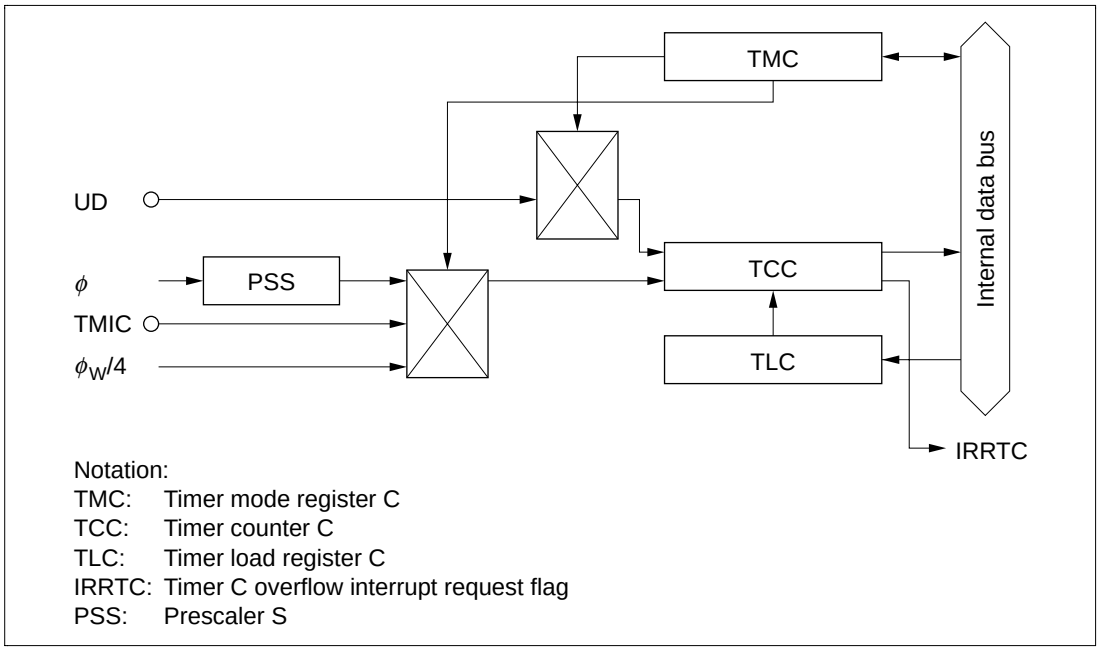


Figure 9.3 Block Diagram of Timer C

Pin Configuration

Table 9.8 shows the timer C pin configuration.

Table 9.8 Pin Configuration

Name	Abbrev.	I/O	Function
Timer C event input	TMIC	Input	Event input to TCC
Timer C up/down control	UD	Input	Selection of counting direction

Register Configuration

Table 9.9 shows the register configuration of timer C.

Table 9.9 Timer C Registers

Name	Abbrev.	R/W	Initial Value	Address
Timer mode register C	TMC	R/W	H'18	H'FFB4
Timer counter C	TCC	R	H'00	H'FFB5
Timer load register C	TLC	W	H'00	H'FFB5

9.4.2 Register Descriptions

Timer Mode Register C (TMC)

Bit	7	6	5	4	3	2	1	0
	TMC7	TMC6	TMC5	—	—	TMC2	TMC1	TMC0
Initial value	0	0	0	1	1	0	0	0
Read/Write	R/W	R/W	R/W	—	—	R/W	R/W	R/W

TMC is an 8-bit read/write register for selecting the auto-reload function, counting direction, and input clock.

Upon reset, TMC is initialized to H'18.

Bit 7—Auto-Reload Function Select (TMC7): Bit 7 selects whether timer C is used as an interval timer or auto-reload timer.

Bit 7: TMC7	Description	
0	Interval timer function selected	(initial value)
1	Auto-reload function selected	

Bits 6 and 5—Counter Up/Down Control (TMC6 and TMC5): These bits select the counting direction of timer counter C (TCC), or allow hardware to control the counting direction using pin UD.

Bit 6: TMC6	Bit 5: TMC5	Description	
0	0	TCC is an up-counter	(initial value)
	1	TCC is a down-counter	
1	*	TCC up/down control is determined by input at pin UD. TCC is a down-counter if the UD input is high, and an up-counter if the UD input is low.	

Note: * Don't care

Bits 4 and 3—Reserved Bits: Bits 4 and 3 are reserved; they are always read as 1, and cannot be modified.

Bits 2 to 0—Clock Select (TMC2 to TMC0): Bits 2 to 0 select the clock input to TCC. For external clock counting, either the rising or falling edge can be selected.

Bit 2: TMC2	Bit 1: TMC1	Bit 0: TMC0	Description	
0	0	0	Internal clock: ϕ/8192	(initial value)
		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: ϕ_W/4	
		1	External event (TMIC): rising or falling edge*	

Note: * The edge of the external event signal is selected by bit IEG2 in the IRQ edge select register (IEGR). See 3.3.2, for details on the IRQ edge select register. Be sure to set bit IRQ2 in port mode register 1 (PMR1) to 1 before setting bits TMC2 to TMC0 to 111.

Timer Counter C (TCC)

Bit	7	6	5	4	3	2	1	0
	TCC7	TCC6	TCC5	TCC4	TCC3	TCC2	TCC1	TCC0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R	R

TCC is an 8-bit read-only up-/down-counter, which is incremented or decremented by internal or external clock input. The clock source for input to this counter is selected by bits TMC2 to TMC0 in timer mode register C (TMC). TCC values can be read by the CPU at any time.

When TCC overflows (from H'FF to H'00 or to the value set in TLC) or underflows (from H'00 to H'FF or to the value set in TLC), the IRRTC bit in interrupt request register 2 (IRR2) is set to 1.

TCC is allocated to the same address as timer load register C (TLC).

Upon reset, TCC is initialized to H'00.

Timer Load Register C (TLC)

Bit	7	6	5	4	3	2	1	0
	TLC7	TLC6	TLC5	TLC4	TLC3	TLC2	TLC1	TLC0
Initial value	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W

TLC is an 8-bit write-only register for setting the reload value of TCC.

When a reload value is set in TLC, the same value is loaded into timer counter C (TCC) as well, and TCC starts counting up or down from that value. When TCC overflows or underflows during operation in auto-reload mode, the TLC value is loaded into TCC. Accordingly, overflow and underflow periods can be set within the range of 1 to 256 input clocks.

The same address is allocated to TLC as to TCC.

Upon reset, TLC is initialized to H'00.

9.4.3 Timer Operation

Interval Timer Operation: When bit TMC7 in timer mode register C (TMC) is cleared to 0, timer C functions as an 8-bit interval timer.

Upon reset, timer counter C (TCC) is initialized to H'00 and TMC to H'18. After a reset, the counter continues uninterrupted incrementing as an interval up-counter. The clock input to timer C is selected from seven internal clock signals output by prescalers S and W, or an external clock input at pin TMIC. The selection is made by bits TMC2 to TMC0 in TMC.

Either software or hardware can control whether TCC counts up or down. The selection is made by TMC bits TMC6 and TMC5.

After the count value in TCC reaches H'FF (H'00), the next clock signal input causes timer C to overflow (underflow), setting bit IRRTC to 1 in interrupt request register 2 (IRR2). If IENTC = 1 in interrupt enable register 2 (IENR2), a CPU interrupt is requested.*

At overflow or underflow, TCC returns to H'00 or H'FF and starts counting up or down again.

During interval timer operation (TMC7 = 0), when a value is set in timer load register C (TLC), the same value is set in TCC.

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

Auto-Reload Timer Operation: Setting bit TMC7 in TMC to 1 causes timer C to function as an 8-bit auto-reload timer. When a reload value is set in TLC, the same value is loaded into TCC, becoming the value from which TCC starts its count.

After the count value in TCC reaches H'FF (H'00), the next clock signal input causes timer C to overflow (underflow). The TLC value is then loaded TCC, and the count continues from that value. The overflow (underflow) period can be set within a range from 1 to 256 input clocks, depending on the TLC value.

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

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

Event Counter Operation: Timer C can operate as an event counter, counting an event signal input at pin TMIC. External event counting is selected by setting TMC bits TMC2 to TMC0 to all 1s (111). TCC counts up or down at the rising or falling edge of the input at pin TMIC.

When timer C is used to count external event inputs, bit IRQ2 in port mode register 1 (PMR1) should be set to 1, and bit IEN2 in interrupt enable register 1 (IENR1) should be cleared to 0 to disable IRQ_2 interrupt requests.

TCC Up/Down Control by Hardware: The counting direction of timer C can be controlled by input at pin UD. When bit TMC6 in TMC is set to 1, high-level input at the UD pin selects down-counting, while low-level input selects up-counting.

When using input at pin UD for this control function, set the UD bit in port mode register 2 (PMR2) to 1.

9.4.4 Timer C Operation States

Table 9.10 summarizes the timer C operation states.

Table 9.10 Timer C Operation States

Operation Mode		Reset	Active	Sleep	Watch	Sub-active	Sub-sleep	Standby
TCC	Interval	Reset	Functions	Functions	Halted	Functions/ Halted*	Functions/ Halted*	Halted
TCC	Auto reload	Reset	Functions	Functions	Halted	Functions/ Halted*	Functions/ Halted*	Halted
TMC		Reset	Functions	Retained	Retained	Functions	Retained	Retained

Note: * When $\phi_W/4$ is selected as the internal clock of TCC in active mode or sleep mode, the internal clock is not synchronous with the system clock, so it is synchronized by a synchronizing circuit. This may result in a maximum error of $1/\phi$ (s) in the count cycle.

When timer C is operated in subactive mode or subsleep mode, either an external clock or the $\phi_W/4$ internal clock must be selected. The counter will not operate in these modes if another clock is selected. If the internal $\phi_W/4$ clock is selected when $\phi_W/8$ is being used as the subclock ϕ_{SUB}, the lower 2 bits of the counter will operate on the same cycle, with the least significant bit not being counted.

9.5 Timer F

9.5.1 Overview

Timer F is a 16-bit timer with an output compare function. Compare match signals can be used to reset the counter, request an interrupt, or toggle the output. Timer F can also be used for external event counting, and can operate as two independent 8-bit timers, timer FH and timer FL.

Features

Features of timer F are given below.

- Choice of four internal clock sources ($\phi/32$, $\phi/16$, $\phi/4$, $\phi/2$) or an external clock (can be used as an external event counter).
- Output from pin TMOFH is toggled by one compare match signal (the initial value of the toggle output can be set).
- Counter can be reset by the compare match signal.
- Two interrupt sources: counter overflow and compare match.
- Can operate as two independent 8-bit timers (timer FH and timer FL) in 8-bit mode.

Timer FH

- 8-bit timer (clocked by timer FL overflow signals when timer F operates as a 16-bit timer).
- Choice of four internal clocks (ϕ/32, ϕ/16, ϕ/4, ϕ/2).
- Output from pin TMOFH is toggled by one compare match signal (the initial value of the toggle output can be set).
- Counter can be reset by the compare match signal.
- Two interrupt sources: counter overflow and compare match.

Timer FL

- 8-bit timer/event counter
- Choice of four internal clocks (ϕ/32, ϕ/16, ϕ/4, ϕ/2) or event input at pin TMIF.
- Output from pin TMOFL is toggled by one compare match signal (the initial value of the toggle output can be set).
- Counter can be reset by the compare match signal.
- Two interrupt sources: counter overflow and compare match.

Block Diagram

Figure 9.4 shows a block diagram of timer F.

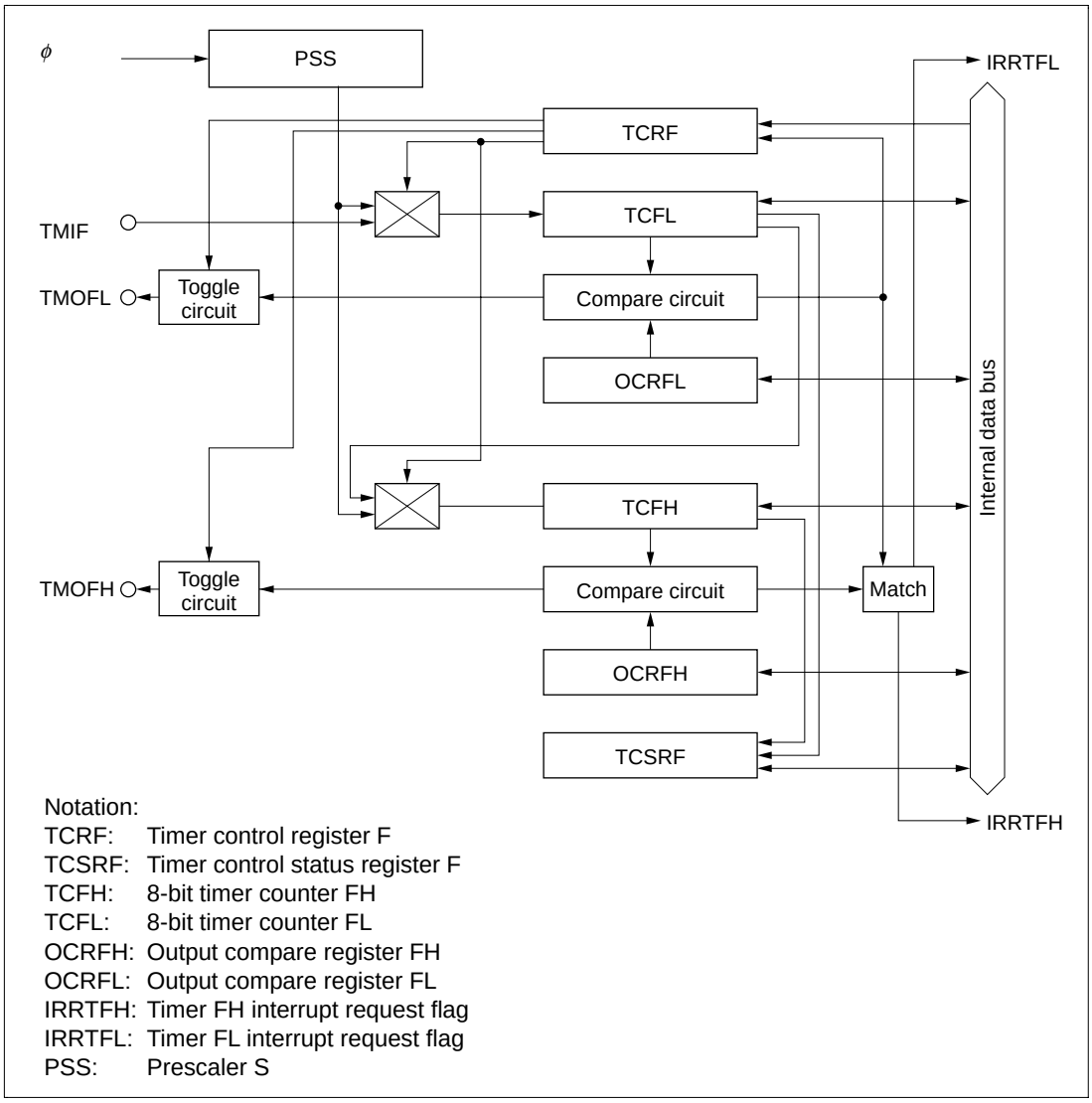


Figure 9.4 Block Diagram of Timer F

Pin Configuration

Table 9.11 shows the timer F pin configuration.

Table 9.11 Pin Configuration

Name	Abbrev.	I/O	Function
Timer F event input	TMIF	Input	Event input to TCFL
Timer FH output	TMOFH	Output	Timer FH toggle output
Timer FL output	TMOFL	Output	Timer FL toggle output

Register Configuration:

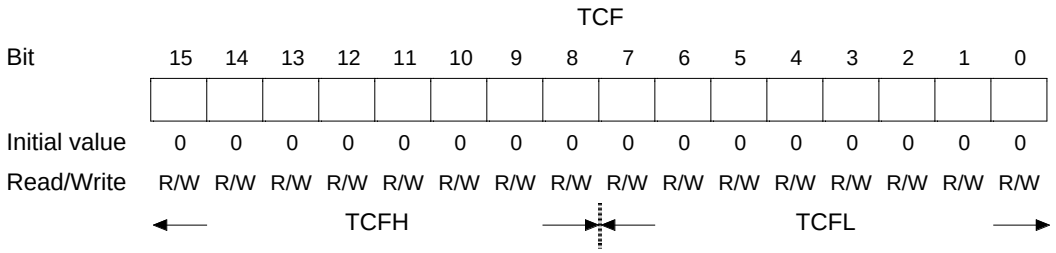
Table 9.12 shows the register configuration of timer F.

Table 9.12 Timer F Registers

Name	Abbrev.	R/W	Initial Value	Address
Timer control register F	TCRF	W	H'00	H'FFB6
Timer control/status register F	TCSRF	R/W	H'00	H'FFB7
8-bit timer counter FH	TCFH	R/W	H'00	H'FFB8
8-bit timer counter FL	TCFL	R/W	H'00	H'FFB9
Output compare register FH	OCRFH	R/W	H'FF	H'FFBA
Output compare register FL	OCRFL	R/W	H'FF	H'FFBB

9.5.2 Register Descriptions

16-Bit Timer Counter (TCF)
8-Bit Timer Counter (TCFH)
8-Bit Timer Counter (TCFL)



TCF is a 16-bit read/write up-counter consisting of two cascaded 8-bit timer counters, TCFH and TCFL. TCF can be used as a 16-bit counter, with TCFH as the upper 8 bits and TCFL as the lower 8 bits of the counter, or TCFH and TCFL can be used as independent 8-bit counters.

TCFH and TCFL can be read and written by the CPU, but in 16-bit mode, data transfer with the CPU takes place via a temporary register (TEMP). For details see 9.5.3, Interface with the CPU.

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

- 16-bit mode (TCF)

  16-bit mode is selected by clearing bit CKSH2 to 0 in timer control register F (TCRF). The TCF input clock is selected by TCRF bits CKSL2 to CKSL0.

  TCFH can be cleared by a compare match signal. This designation is made in bit CCLRH in TCSRF.

  When TCF overflows from H'FFFF to H'0000, the overflow flag (OVFH) in TCSRF is set to 1. If bit OVIEH in TCSRF is set to 1 when an overflow occurs, bit IRRTFH in interrupt request register 2 (IRR2) will be set to 1; and if bit IENTFH in interrupt enable register 2 (IENR2) is set to 1, a CPU interrupt will be requested.

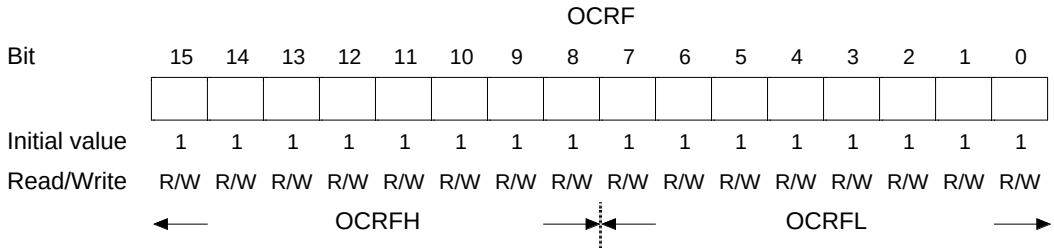
- 8-bit mode (TCFH, TCFL)

  When bit CKSH2 in timer control register F (TCRF) is set to 1, timer F functions as two separate 8-bit counters, TCFH and TCFL. The TCFH (TCFL) input clock is selected by TCRF bits CKSH2 to CKSH0 (CKSL2 to CKSL0).

  TCFH (TCFL) can be cleared by a compare match signal. This designation is made in bit CCLRH (CCLRL) in TCSRF.

  When TCFH (TCFL) overflows from H'FF to H'00, the overflow flag OVFH (OVFL) in TCSRF is set to 1. If bit OVIEH (OVIEL) in TCSRF is set to 1 when an overflow occurs, bit IRRTFH (IRRTHL) in interrupt request register 2 (IRR2) will be set to 1; and if bit IENTFH (IENTFL) in interrupt enable register 2 (IENR2) is set to 1, a CPU interrupt will be requested.

16-Bit Output Compare Register (OCRF)
8-Bit Output Compare Register (OCRFH)
8-Bit Output Compare Register (OCRFL)



OCRF is a 16-bit read/write output compare register consisting of two 8-bit read/write registers OCRFH and OCRFL. It can be used as a 16-bit output compare register, with OCRFH as the upper 8 bits and OCRFL as the lower 8 bits of the register, or OCRFH and OCRFL can be used as independent 8-bit registers.

OCRFH and OCRFL can be read and written by the CPU, but in 16-bit mode, data transfer with the CPU takes place via a temporary register (TEMP). For details see 9.5.3, Interface with the CPU.

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

- 16-bit mode (OCRF)

  16-bit mode is selected by clearing bit CKSH2 to 0 in timer control register F (TCRF). The OCRF contents are always compared with the 16-bit timer counter (TCF). When the contents match, the compare match flag (CMFH) in TCSRF is set to 1. Also, IRRTFH in interrupt request register 2 (IRR2) is set to 1. If bit IENTFH in interrupt enable register 2 (IENR2) is set to 1, a CPU interrupt is requested.

  Output for pin TMOFH can be toggled by compare match. The output level can also be set to high or low by bit TOLH of timer control register F (TCRF).

- 8-bit mode (OCRFH, OCRFL)

  Setting bit CKSH2 in TCRF to 1 results in two 8-bit registers, OCRFH and OCRFL.

  The OCRFH contents are always compared with TCFH, and the OCRFL contents are always compared with TCFL. When the contents match, the compare match flag (CMFH or CMFL) in TCSRF is set to 1. Also, bit IRRTFH (IRRTFL) in interrupt request register 2 (IRR2) set to 1. If bit IENTFH (IENTFL) in interrupt enable register 2 (IENR2) is set to 1 at this time, a CPU interrupt is requested.

  The output at pin TMOFH (TMOFL) can be toggled by compare match. The output level can also be set to high or low by bit TOLH (TOLL) of the timer control register (TCRF).

Timer Control Register F (TCRF)

Bit	7	6	5	4	3	2	1	0
	TOLH	CKSH2	CKSH1	CKSH0	TOLL	CKSL2	CKSL1	CKSL0
Initial value	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W

TCRF is an 8-bit write-only register. It is used to switch between 16-bit mode and 8-bit mode, to select among four internal clocks and an external clock, and to select the output level at pins TMOFH and TMOFL.

Upon reset, TCRF is initialized to H'00.

Bit 7—Toggle Output Level H (TOLH): Bit 7 sets the output level at pin TMOFH. The setting goes into effect immediately after this bit is written.

Bit 7: TOLH	Description	
0	Low level	(initial value)
1	High level	

Bits 6 to 4—Clock Select H (CKSH2 to CKSH0): Bits 6 to 4 select the input to TCFH from four internal clock signals or the overflow of TCFL.

Bit 6: CKSH2	Bit 5: CKSH1	Bit 4: CKSH0	Description
0	*	*	16-bit mode selected. TCFL overflow signals are counted (initial value)
1	0	0	Internal clock: ϕ/32
		1	Internal clock: ϕ/16
	1	0	Internal clock: ϕ/4
		1	Internal clock: ϕ/2

Note: * Don't care

Bit 3—Toggle Output Level L (TOLL): Bit 3 sets the output level at pin TMOFL. The setting goes into effect immediately after this bit is written.

Bit 3: TOLL	Description	
0	Low level	(initial value)
1	High level	

Bits 2 to 0—Clock Select L (CKSL2 to CKSL0): Bits 2 to 0 select the input to TCFL from four internal clock signals or external event input.

Bit 2: CKSL2	Bit 1: CKSL1	Bit 0: CKSL0	Description
0	*	*	External event (TMIF). Rising or falling edge is counted*[1] (initial value)
1	0	0	Internal clock: ϕ/32
		1	Internal clock: ϕ/16
	1	0	Internal clock: ϕ/4
		1	Internal clock: ϕ/2

* Don't care

Note: 1. The edge of the external event signal is selected by bit IEG3 in the IRQ edge select register (IEGR). See 3.3.2, for details on the IRQ edge select register. Note that switching the TMIF pin function by changing bit IRQ3 in port mode register 1 (PMR1) from 0 to 1 or from 1 to 0 while the TMIF pin is at the low level may cause the timer F counter to be incremented.

Timer Control/Status Register F (TCSRF)

Bit	7	6	5	4	3	2	1	0
	OVFH	CMFH	OVIEH	CCLRH	OVFL	CMFL	OVIEL	CCLRL
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W*	R/W*	R/W	R/W	R/W*	R/W*	R/W	R/W

Note: * Only 0 can be written, to clear flag.

TCSRF is an 8-bit read/write register. It is used for counter clear selection, overflow and compare match indication, and enabling of interrupts caused by timer overflow.

Upon reset, TCSRF is initialized to H'00.

Bit 7—Timer overflow flag H (OVFH): Bit 7 is a status flag indicating TCFH overflow (H'FF to H'00). This flag is set by hardware and cleared by software. It cannot be set by software.

Bit 7: OVFH	Description
0	Clearing conditions: After reading OVFH = 1, cleared by writing 0 to OVFH (initial value)
1	Setting conditions: Set when the value of TCFH goes from H'FF to H'00

Bit 6—Compare Match Flag H (CMFH): Bit 6 is a status flag indicating a compare match between TCFH and OCRFH. This flag is set by hardware and cleared by software. It cannot be set by software.

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

Bit 5—Timer Overflow Interrupt Enable H (OVIEH): Bit 5 enables or disables TCFH overflow interrupts.

Bit 5: OVIEH	Description	
0	TCFH overflow interrupt disabled	(initial value)
1	TCFH overflow interrupt enabled	

Bit 4—Counter Clear H (CCLRH): In 16-bit mode, bit 4 selects whether or not TCF is cleared when a compare match occurs between TCF and OCRF.

In 8-bit mode, bit 4 selects whether or not TCFH is cleared when a compare match occurs between TCFH and OCRFH.

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

Bit 3—Timer Overflow Flag L (OVFL): Bit 3 is a status flag indicating TCFL overflow (H'FF to H'00). This flag is set by hardware and cleared by software. It cannot be set by software.

Bit 3: OVFL	Description	
0	Clearing conditions: After reading OVFL = 1, cleared by writing 0 to OVFL	(initial value)
1	Setting conditions: Set when the value of TCFL goes from H'FF to H'00	

Bit 2—Compare Match Flag L (CMFL): Bit 2 is a status flag indicating a compare match between TCFL and OCRFL. This flag is set by hardware and cleared by software. It cannot be set by software.

Bit 2: CMFL	Description	
0	Clearing conditions: After reading CMFL = 1, cleared by writing 0 to CMFL	(initial value)
1	Setting conditions: Set when the TCFL value matches the OCRFL value	

Bit 1—Timer Overflow Interrupt Enable L (OVIEL): Bit 1 enables or disables TCFL overflow interrupts.

Bit 1: OVIEL	Description	
0	TCFL overflow interrupt disabled	(initial value)
1	TCFL overflow interrupt enabled	

Bit 0—Counter Clear L (CCLRL): Bit 0 selects whether or not TCFL is cleared when a compare match occurs between TCFL and OCRFL.

Bit 0: CCLRL	Description	
0	TCFL clearing by compare match disabled	(initial value)
1	TCFL clearing by compare match enabled	

9.5.3 Interface with the CPU

TCF and OCRF are 16-bit read/write registers, whereas the data bus between the CPU and on-chip peripheral modules has an 8-bit width. For this reason, when the CPU accesses TCF or OCRF, it makes use of an 8-bit temporary register (TEMP).

In 16-bit mode, when reading or writing TCF or writing OCRF, always use two consecutive byte size MOV instructions, and always access the upper byte first. Data will not be transferred properly if only the upper byte or only the lower byte is accessed. In 8-bit mode there is no such restriction on the order of access.

Write Access: When the upper byte is written, the upper-byte data is loaded into the TEMP register. Next when the lower byte is written, the data in TEMP goes to the upper byte of the register, and the lower-byte data goes directly to the lower byte of the register. Figure 9.5 shows a TCF write operation when H'AA55 is written to TCF.

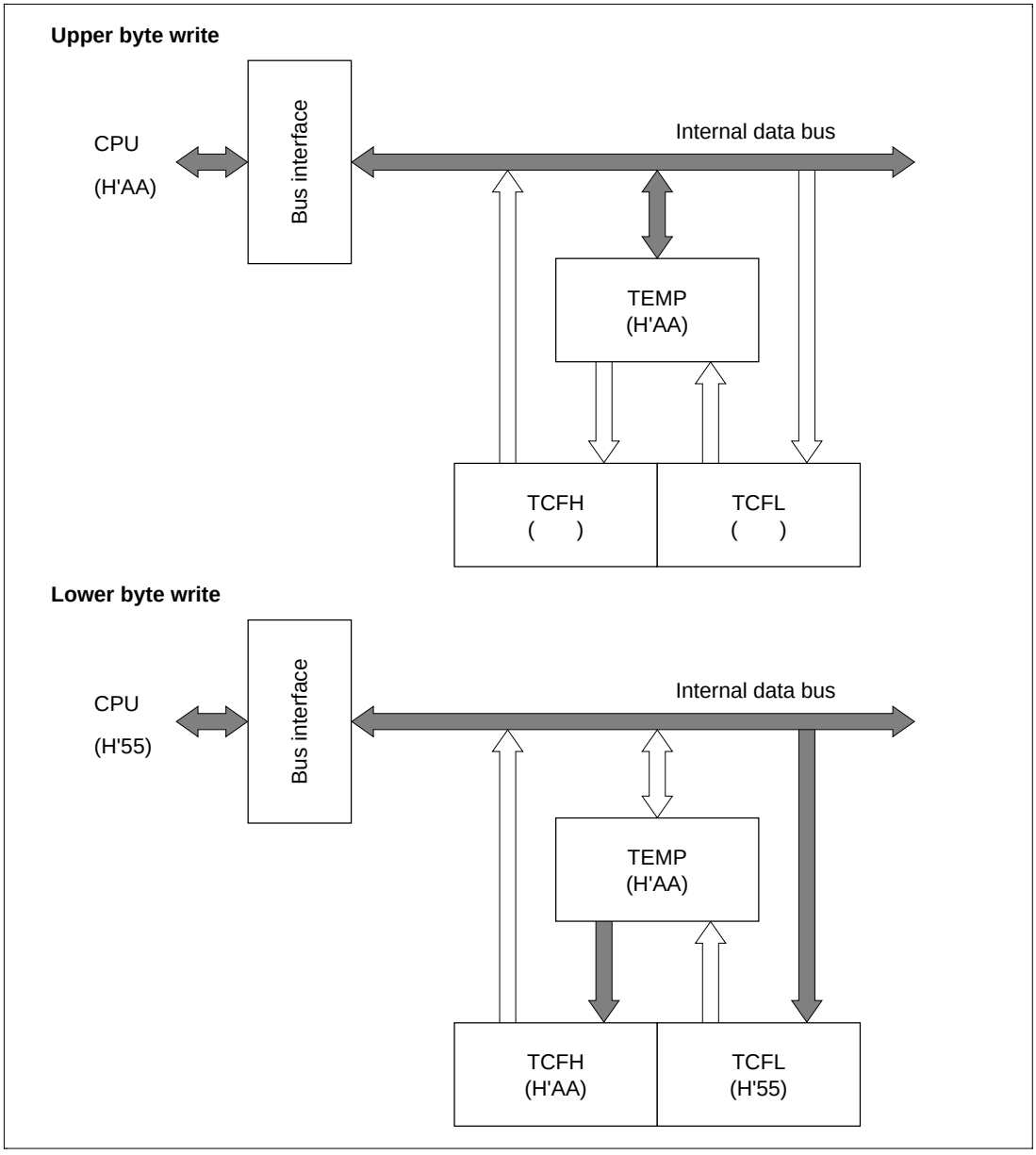
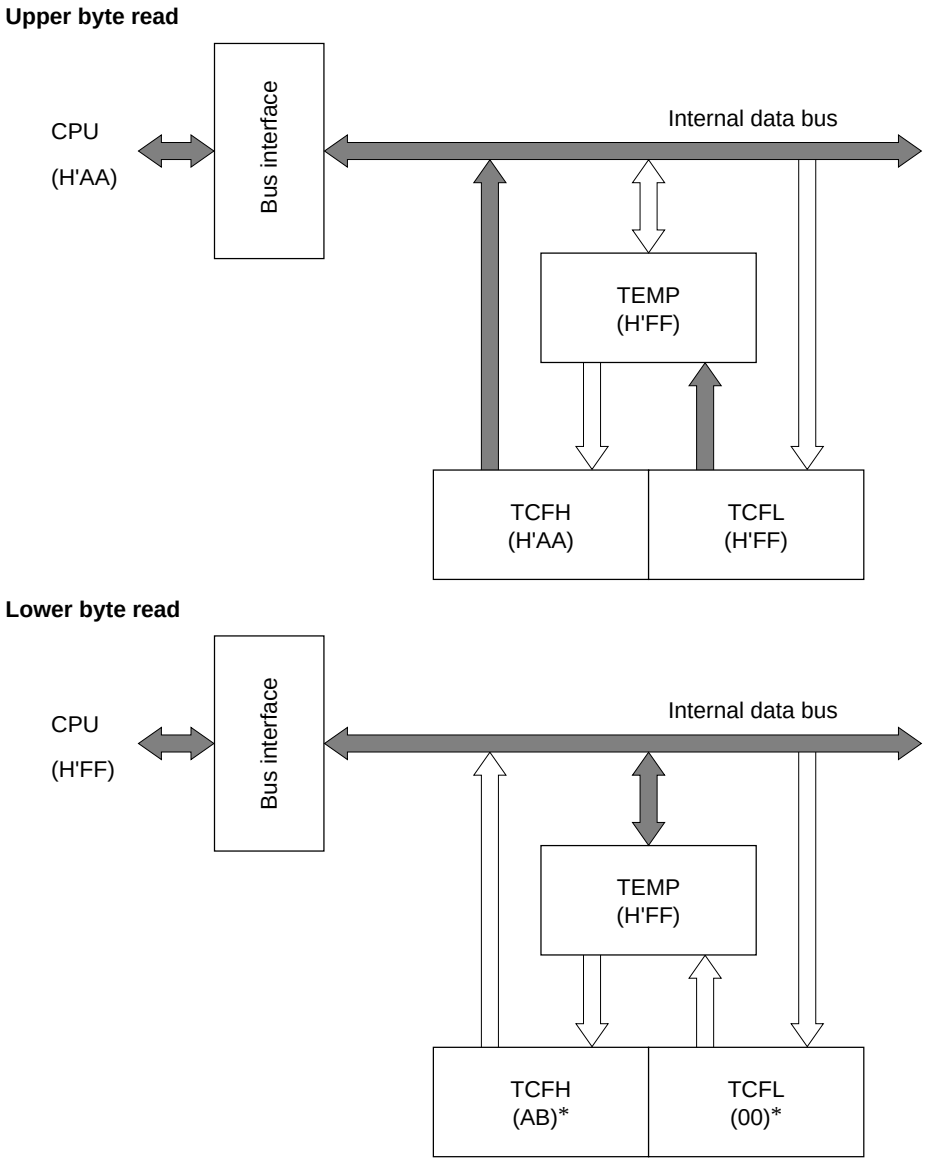


Figure 9.5 TCF Write Operation (CPU → TCF)

Read Access: When the upper byte of TCF is read, the upper-byte data is sent directly to the CPU, and the lower byte is loaded into TEMP. Next when the lower byte is read, the lower byte in TEMP is sent to the CPU.

When the upper byte of OCRF is read, the upper-byte data is sent directly to the CPU. Next when the lower byte is read, the lower-byte data is sent directly to the CPU.

Figure 9.6 shows a TCF read operation when H'AAFF is read from TCF.



Note: * Becomes H'AB00 if counter is incremented once.

Figure 9.6 TCF Read Operation (TCF → CPU)

9.5.4 Timer Operation

Timer F is a 16-bit timer/counter that increments with each input clock. The value set in output compare register F is constantly compared with the value of timer counter F, and when they match the counter can be cleared, an interrupt can be requested, and the port output can be toggled. Timer F can also be used as two independent 8-bit timers.

Timer F Operation: Timer F can operate in either 16-bit timer mode or 8-bit timer mode. These modes are described below.

- 16-bit timer mode

  Timer F operates in 16-bit timer mode when the CKSH2 bit in timer control register F (TCRF) is cleared to 0.

  A reset initializes timer counter F (TCF) to H'0000, output compare register F (OCRF) to H'FFFF, and timer control register F (TCRF) and timer control status register F (TCSRF) to H'00. Timer F begins counting external event input signals (TMIF). The edge of the external event signal is selected by the IEG3 bit in the IRQ edge select register (IEGR).

  Any of four internal clocks output by prescaler S, or an external clock, can be selected as the timer F operating clock by bits CKSL2 to CKSL0 in TCRF.

  TCF is continuously compared with the contents of OCRF. When these two values match, the CMFH bit in TCSRF is set to 1. At this time if IENTFH of IENR2 is 1, a CPU interrupt is requested and the output at pin TMOFH is toggled. If the CCLRH bit in TCSRF is 1, timer F is cleared. The output at pin TMOFH can also be set by the TOLH bit in TCRF.

  If timer F overflows (from H'FFFF to H'0000), the OVFH bit in TCSRF is set. At this time, if the OVIEH bit in TCSRF and the IENTFH bit in IENR2 are both 1, a CPU interrupt is requested.

- 8-bit timer mode

  When the CKSH2 bit in TCRF is set to 1, timer F operates as two independent 8-bit timers, TCFH and TCFL. The input clock of TCFH/TCFL is selected by bits CKSH2 to CKSH0/CKSL2 to CKSL0 in TCRF.

  When TCFH/TCFL and the contents of OCRFH/OCRFL match, the CMFH/CMFL bit in TCSRF is set to 1. If the IENTFH/IENTFL bit in IENR2 is 1, a CPU interrupt is requested and the output at pin TMOFH/TMOFL is toggled. If the CCLRH/CCLRL bit in TCRF is 1, TCFH/TCFL is cleared. The output at pin TMOFH/TMOFL can also be set by the TOLH/TOLL bit in TCRF.

  When TCFH/TCFL overflows from H'FF to H'00, the OVFH/OVFL bit in TCSRF is set to 1. At this time, if the OVIEH/OVIEL bit in TCSRF and the IENTFH/IENTFL bit in IENR2 are both 1, a CPU interrupt is requested.

TCF Count Timing: TCF is incremented by each pulse of the input clock (internal or external clock).

- Internal clock

  The settings of bits CKSH2 to CKSH0 or bits CKSL2 to CKSL0 in TCRF select one of four internal clock signals divided from the system clock (ϕ), namely, ϕ/32, ϕ/16, ϕ/4, or ϕ/2.
- External clock

  External clock input is selected by clearing bit CKSL2 to 0 in TCRF. Either rising or falling edges of the clock input can be counted. The edge of an external event is selected by bit IEG3 in the interrupt controller's IEGR register. An external event pulse width of at least two system clock (ϕ) cycles is necessary for correct operation of the counter.

TMOFH and TMOFL Output Timing: The outputs at pins TMOFH and TMOFL are the values set in bits TOLH and TOLL in TCRF. When a compare match occurs, the output value is inverted. Figure 9.7 shows the output timing.

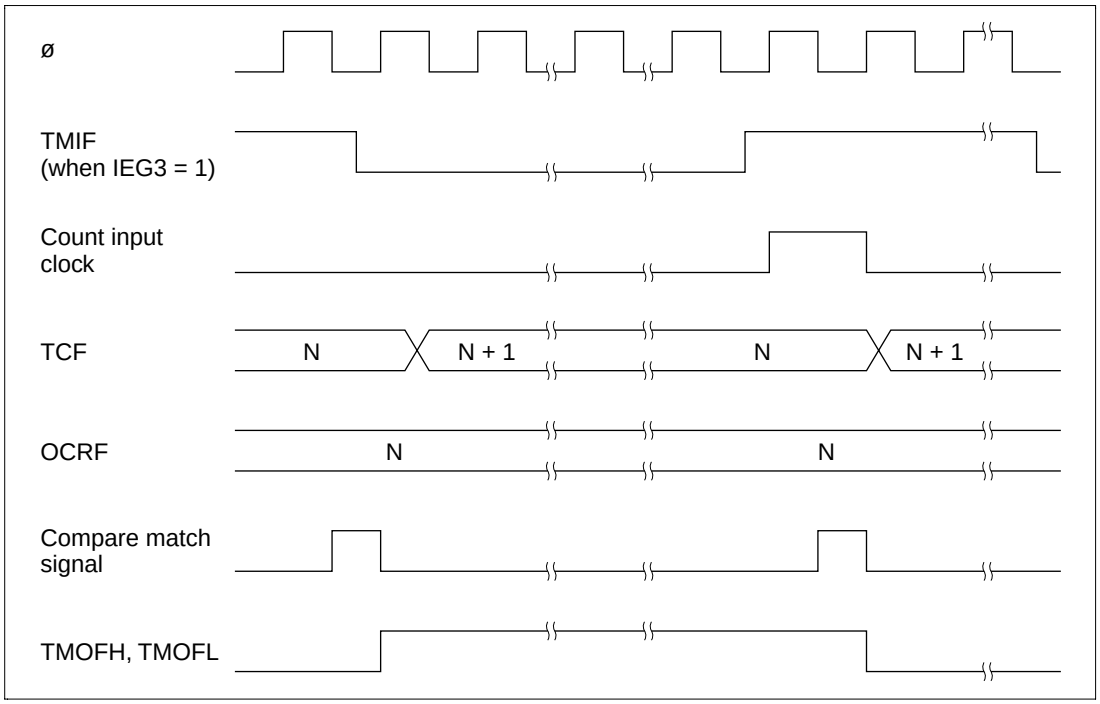


Figure 9.7 TMOFH, TMOFL Output Timing

TCF Clear Timing: TCF can be cleared at compare match with OCRF.

Timer Overflow Flag (OVF) Set Timing: OVF is set to 1 when TCF overflows (goes from H'FFFF to H'0000).

Compare Match Flag Set Timing: The compare match flags (CMFH or CMFL) are set to 1 when a compare match occurs between TCF and OCRF. A compare match signal is generated in the final state in which the values match (when TCF changes from the matching count value to the next value). When TCF and OCRF match, a compare match signal is not generated until the next counter clock pulse.

Timer F Operation States: Table 9.13 summarizes the timer F operation states.

Table 9.13 Timer F Operation States

Operation Mode	Reset	Active	Sleep	Watch	Sub-active	Sub-sleep	Standby
TCF	Reset	Functions	Functions	Halted	Halted	Halted	Halted
OCRF	Reset	Functions	Retained	Retained	Retained	Retained	Retained
TCRF	Reset	Functions	Retained	Retained	Retained	Retained	Retained
TCSRF	Reset	Functions	Retained	Retained	Retained	Retained	Retained

9.5.5 Application Notes

The following conflicts can arise in timer F operation.

- 16-bit timer mode

  The output at pin TMOFH toggles when all 16 bits match and a compare match signal is generated. If the compare match signal occurs at the same time as new data is written in TCRF by a MOV instruction, however, the new value written in bit TOLH will be output at pin TMOFH. The TMOFL output in 16-bit mode is indeterminate, so this output should not be used. Use the pin as a general input or output port.

  If an OCRFL write occurs at the same time as a compare match signal, the compare match signal is inhibited. If a compare match occurs between the written data and the counter value, however, a compare match signal will be generated at that point. The compare match signal is output in synchronization with the TCFL clock, so if this clock is stopped no compare match signal will be generated, even if a compare match occurs.

  Compare match flag CMFH is set when all 16 bits match and a compare match signal is generated; bit CMFL is set when the setting conditions are met for the lower 8 bits.

  The overflow flag (OVFH) is set when TCF overflows; bit OVFL is set if the setting conditions are met when the lower 8 bits overflow. If a write to TCFL occurs at the same time as an overflow signal, the overflow signal is not output.

- 8-bit timer mode
  - TCFH and OCRFH

    The output at pin TMOFH toggles when there is a compare match. If the compare match signal occurs at the same time as new data is written in TCRF by a MOV instruction, however, the new value written in bit TOLH will be output at pin TMOFH.

    If an OCRFH write occurs at the same time as a compare match signal, the compare match signal is inhibited. If a compare match occurs between the written data and the counter value, however, a compare match signal will be generated at that point. The compare match signal is output in synchronization with the TCFH clock.

    If a TCFH write occurs at the same time as an overflow signal, the overflow signal is not output.

  - TCFL and OCRFL

    The output at pin TMOFL toggles when there is a compare match. If the compare match signal occurs at the same time as new data is written in TCRF by a MOV instruction, however, the new value written in bit TOLL will be output at pin TMOFL.

    If an OCRFL write occurs at the same time as a compare match signal, the compare match signal is inhibited. If a compare match occurs between the written data and the counter value, however, a compare match signal will be generated at that point. The compare match signal is output in synchronization with the TCFL clock, so if this clock is stopped no compare match signal will be generated, even if a compare match occurs.

    If a TCFL write occurs at the same time as an overflow signal, the overflow signal is not output.

9.6 Watchdog Timer [H8/3857F and H8/3854F Only]

9.6.1 Overview

The watchdog timer (WDT) is equipped with an 8-bit counter that is incremented by an input clock. An internal chip reset can be executed if the counter overflows because it is not updated normally due to a system crash, etc.

This watchdog timer is used by the flash memory programming control program.

Features

Features of the watchdog timer are given below.

- Choice of eight internal clock sources (ø/64, ø/128, ø/256, ø/512, ø/1024, ø/2048, ø/4096, ø/8192)

- Reset signal generated on counter overflow
  An overflow period of 1 to 256 times the selected clock can be set.

Block Diagram

Figure 9.8 shows a block diagram of the watchdog timer.

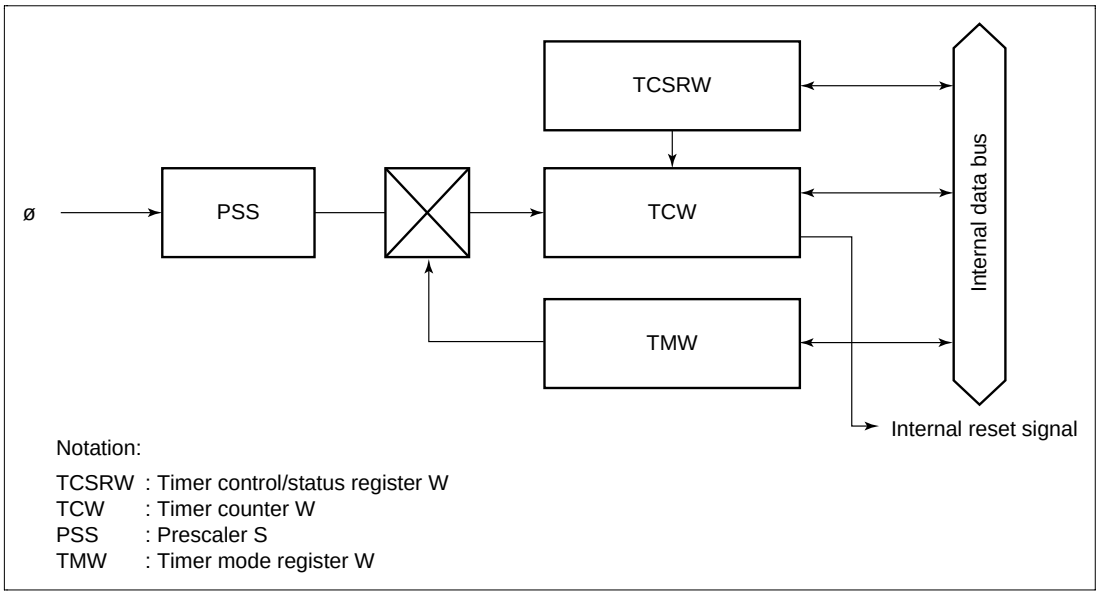


Figure 9.8 Block Diagram of Watchdog Timer

Register Configuration

Table 9.14 shows the watchdog timer register configuration. These registers are valid only in the F-ZTAT version. In the mask ROM version, read accesses to the corresponding addresses will always return 1, and writes are invalid.

Table 9.14 Watchdog Timer Registers

Name	Abbrev.	R/W	Initial Value	Address
Timer control/status register W	TCSRW	R/W	H'AA	H'FF90
Timer counter W	TCW	R/W	H'00	H'FF91
Timer mode register W	TMW	R/W	H'FF	H'FF92

9.6.2 Register Descriptions

Timer Control/Status Register W (TCSRW)

Bit	7	6	5	4	3	2	1	0
	B6WI	TCWE	B4WI	TCSRWE	B2WI	WDON	BOWI	WRST
Initial value	1	0	1	0	1	0	1	0
Read/Write	R	R/(W)*	R	R/(W)*	R	R/(W)*	R	R/(W)*

Note: * Can be written to only when the write condition is satisfied. For the write conditions, see the individual bit descriptions.

TCSRW is an 8-bit read/write register that performs TCSRW and TCW write control and watchdog timer operation control, and indicates the operation status.

Bit 7—Bit 6 Write Inhibit (B6WI): Bit 7 controls writing of data to bit 6 of TCSRW.

Bit 7

B6WI	Description	
0	Writing to bit 6 is enabled	
1	Writing to bit 6 is disabled	(initial value)

This bit is always read as 1. Data is not stored if written to this bit.

Bit 6—Timer Counter W Write Enable (TCWE): Bit 6 controls writing of 8-bit data to TCW.

Bit 6

TCWE	Description	
0	Writing of 8-bit data to TCW is disabled	(initial value)
1	Writing of 8-bit data to TCW is enabled	

Bit 5—Bit 4 Write Inhibit (B4WI): Bit 5 controls writing of data to bit 4 of TCSRW.

Bit 5

B4WI	Description	
0	Writing to bit 4 is enabled	
1	Writing to bit 4 is disabled	(initial value)

This bit is always read as 1. Data is not stored if written to this bit.

Bit 4—Timer Control/Status Register W Write Enable (TCSRWE): Bit 4 controls writing of data to bits 2 and 0 of TCSRW.

Bit 4		
TCSRWE	**Description**	
0	Writing to bits 2 and 0 is disabled	(initial value)
1	Writing to bits 2 and 0 is enabled	

Bit 3—Bit 2 Write Inhibit (B2WI): Bit 3 controls writing of data to bit 2 of TCSRW.

Bit 3		
B2WI	**Description**	
0	Writing to bit 2 is enabled	
1	Writing to bit 2 is disabled	(initial value)

This bit is always read as 1. Data is not stored if written to this bit.

Bit 2—Watchdog Timer On (WDON): Bit 2 controls watchdog timer operation.

Bit 2		
WDON	**Description**	
0	Watchdog timer operation is disabled [Clearing condition] In a reset, or when 0 is written to WDON while writing 0 to B2WI when TCSRWE = 1	(initial value)
1	Watchdog timer operation is enabled [Setting condition] When 1 is written to WDON while writing 0 to B2WI when TCSRWE = 1	

The count-up starts when this bit is set to 1, and stops when it is cleared to 0.

Bit 1—Bit 0 Write Inhibit (B0WI): Bit 1 controls writing of data to bit 0 of timer control/status register W.

Bit 1		
B0WI	**Description**	
0	Writing to bit 0 is enabled	
1	Writing to bit 0 is disabled	(initial value)

This bit is always read as 1. Data is not stored if written to this bit.

Bit 0—Watchdog Timer Reset (WRST): Bit 0 indicates that TCW has overflowed and an internal reset signal has been generated. The internal reset signal generated by the overflow resets the entire chip.

WRST is cleared by a reset via the $\overline{\text{RES}}$ pin or by a 0 write by software.

Bit 0

WRST	Description
0	[Clearing conditions] (initial value) • Reset by $\overline{\text{RES}}$ pin • When 0 is written to WRST while writing 0 to B0WI when TCSRWE = 1
1	[Setting condition] When TCW overflows and an internal reset signal is generated

Timer Counter W (TCW)

Bit	7	6	5	4	3	2	1	0
	TCW7	TCW6	TCW5	TCW4	TCW3	TCW2	TCW1	TCW0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TCW is an 8-bit read/write up-counter that is incremented by an input internal clock. The TCW value can be read or written by the CPU at any time.

When TCW overflows (from H'FF to H'00), an internal reset signal is generated and WRST in TCSRW is set to 1. Upon reset, TCW is initialized to H'00.

Timer Mode Register W (TMW)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	CKS2	CKS1	CKS0
Initial value	1	1	1	1	1	1	1	1
Read/Write	—	—	—	—	—	R/W	R/W	R/W

TMW is an 8-bit read/write register that selects the input clock.

Upon reset, TMW is initialized to H'FF.

Bits 7 to 3—Reserved Bits: Bits 7 to 3 are reserved; they are always read as 1 and cannot be modified.

Bits 2 to 0—Clock Select 2 to 0 (CKS2 to CKS0): Bits 2 to 0 select the clock to be input to TCW.

Bit 2	Bit 1	Bit 0		
CKS2	**CKS1**	**CKS0**	**Description**	
0	0	0	Internal clock: ø/64	
		1	Internal clock: ø/128	
	1	0	Internal clock: ø/256	
		1	Internal clock: ø/512	
1	0	0	Internal clock: ø/1024	
		1	Internal clock: ø/2048	
	1	0	Internal clock: ø/4096	
		1	Internal clock: ø/8192	(initial value)

9.6.3 Operation

The watchdog timer is provided with an 8-bit counter that increments with each input clock pulse. If 1 is written to WDON while writing 0 to B2WI when TCSRWE in TCSRW is set to 1, TCW begins counting up. When a clock pulse is input after the TCW count value has reached H'FF, the watchdog timer overflows and an internal reset signal is generated one base clock (ø)cycle later. The internal reset signal is output for a period of 512 $ø_{osc}$ clock cycles. TCW is a writable counter, and when a value is set in TCW, the count-up starts from that value. An overflow period in the range of 1 to 256 input clock cycles can therefore be set, according to the TCW value.

Figure 9.9 shows an example of watchdog timer operation.

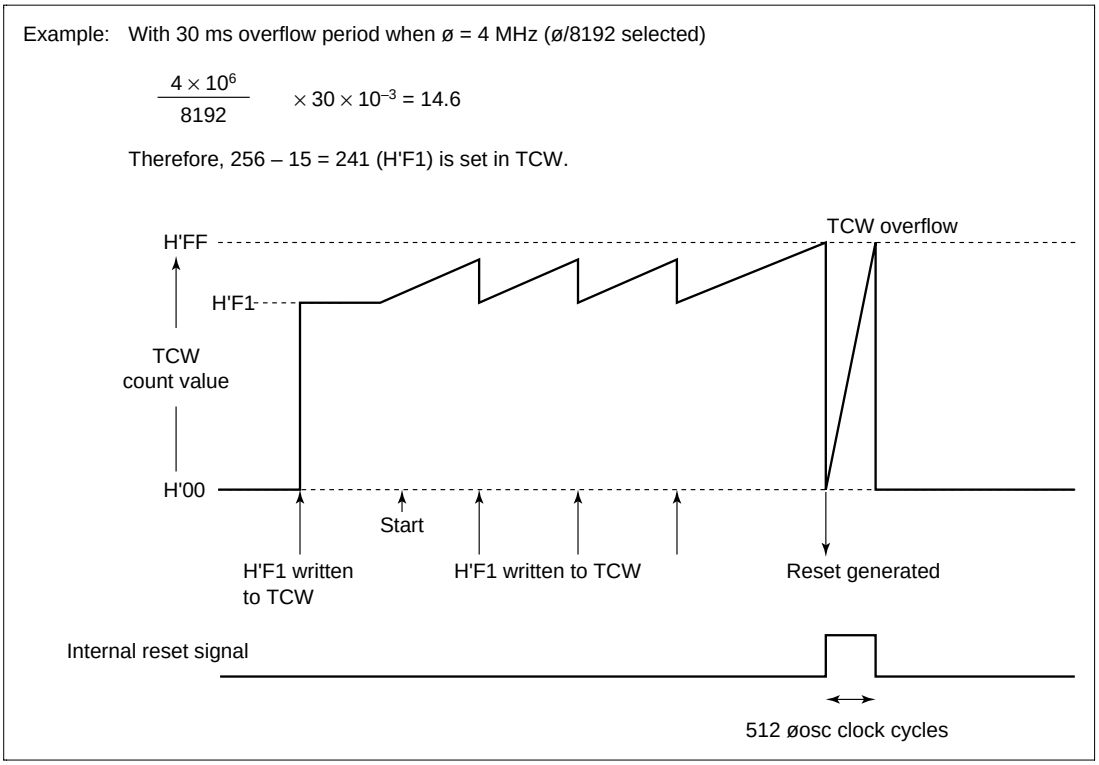


Figure 9.9 Example of Watchdog Timer Operation

9.6.4 Watchdog Timer Operating Modes

Watchdog timer operating modes are shown in table 9.15.

Table 9.15 Watchdog Timer Operating Modes

Operating mode	Reset	Active	Sleep	Watch	Subactive	Subsleep	Standby
TCW	Reset	Functions	Functions	Halted	Halted	Halted	Halted
TCSRW	Reset	Functions	Functions	Retained	Retained	Retained	Retained
TMW	Reset	Functions	Retained	Retained	Retained	Retained	Retained

Section 10 Serial Communication Interface

10.1 Overview

The H8/3857 Series is provided with a two-channel serial communication interface (SCI), and the H8/3854 Series with a single-channel SCI. Table 10.1 summarizes the functions and features of the SCI channels.

Table 10.1 Serial Communication Interface Functions

Channel	Functions	Features
SCI1*	Synchronous serial transfer • Choice of 8-bit or 16-bit data length • Continuous clock output	• Choice of 8 internal clocks (ϕ/1024 to ϕ/2) or external clock • Open drain output possible • Interrupt requested at completion of transfer
SCI3	Synchronous serial transfer • 8-bit data transfer • Send, receive, or simultaneous send/receive Asynchronous serial transfer • Multiprocessor communication function • Choice of 7-bit or 8-bit data length • Choice of 1-bit or 2-bit stop bit length • Parity addition	• Built-in baud rate generator • Receive error detection • Break detection • Interrupt requested at completion of transfer or error

Note: * SCI1 is a function of the H8/3857 Series only, and is not provided in the H8/3854 Series.

10.2 SCI1 (H8/3857 Series Only)

10.2.1 Overview

Serial communication interface 1 (SCI1) performs synchronous serial transfer of 8-bit or 16-bit data.

SCI1 is a function of the H8/3857 Series only, and is not provided in the H8/3854 Series.

Features

Features of SCI1 are as follows.

- Choice of 8-bit or 16-bit transfer data length
- Choice of eight internal clock sources (ϕ/1024, ϕ/256, ϕ/64, ϕ/32, ϕ/16, ϕ/8, ϕ/4, ϕ/2) or an external clock
- Interrupt requested at completion of transfer

Block Diagram

Figure 10.1 shows a block diagram of SCI1.

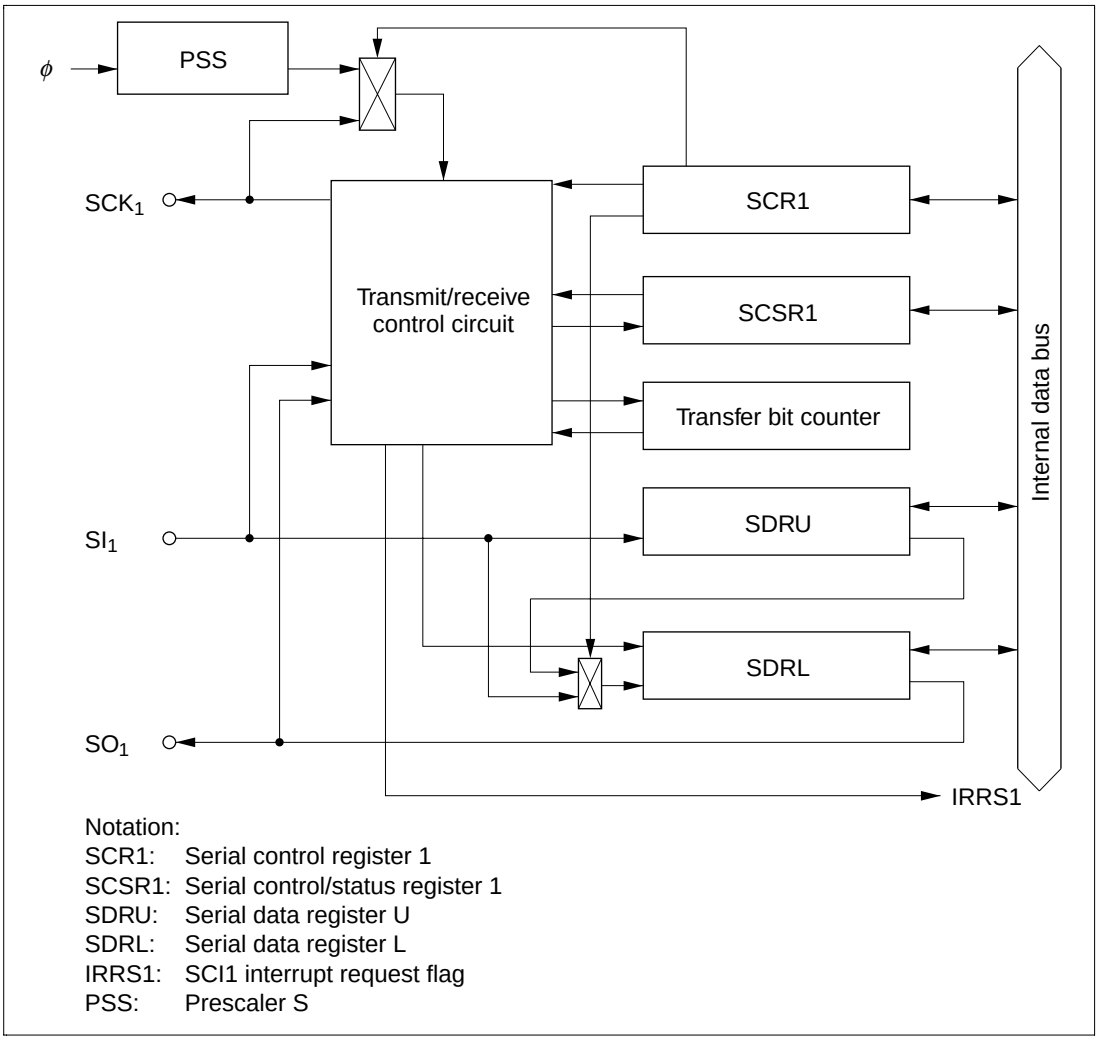


Figure 10.1 SCI1 Block Diagram

Pin Configuration

Table 10.2 shows the SCI1 pin configuration.

Table 10.2 Pin Configuration

Name	Abbrev.	I/O	Function
SCI1 clock pin	SCK_1	I/O	SCI1 clock input or output
SCI1 data input pin	SI_1	Input	SCI1 receive data input
SCI1 data output pin	SO_1	Output	SCI1 transmit data output

Register Configuration

Table 10.3 shows the SCI1 register configuration.

Table 10.3 SCI1 Registers

Name	Abbrev.	R/W	Initial Value	Address
Serial control register 1	SCR1	R/W	H'00	H'FFA0
Serial control status register 1	SCSR1	R/W	H'80	H'FFA1
Serial data register U	SDRU	R/W	Undefined	H'FFA2
Serial data register L	SDRL	R/W	Undefined	H'FFA3

10.2.2 Register Descriptions

Serial Control Register 1 (SCR1)

Bit	7	6	5	4	3	2	1	0
	SNC1	SNC0	—	—	CKS3	CKS2	CKS1	CKS0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SCR1 is an 8-bit read/write register for selecting the operation mode, the transfer clock source, and the prescaler division ratio.

Upon reset, SCR1 is initialized to H'00. Writing to this register during a transfer stops the transfer.

Bits 7 and 6—Operation Mode Select 1, 0 (SNC1, SNC0): Bits 7 and 6 select the operation mode.

Bit 7: SNC1	Bit 6: SNC0	Description	
0	0	8-bit synchronous transfer mode	(initial value)
	1	16-bit synchronous transfer mode	
1	0	Continuous clock output mode*[1]	
	1	Reserved*[2]	

Notes: 1. Pins SI_1 and SO_1 should be used as general input or output ports.
2. Don't set bits SNC1 and SNC0 to 11.

Bits 5 and 4—Reserved Bits: Bits 5 and 4 are reserved; they should always be cleared to 0.

Bit 3—Clock Source Select 3 (CKS3): Bit 3 selects the clock source and sets pin SCK_1 as an input or output pin.

Bit 3: CKS3	Description	
0	Clock source is prescaler S, and pin SCK_1 is output pin	(initial value)
1	Clock source is external clock, and pin SCK_1 is input pin	

Bits 2 to 0—Clock Select 2 to 0 (CKS2 to CKS 0): When CKS3 = 0, bits 2 to 0 select the prescaler division ratio and the serial clock cycle.

				Serial Clock Cycle	
Bit 2: CKS2	Bit 1: CKS1	Bit 0: CKS0	Prescaler Division	ϕ = 5 MHz	ϕ = 2.5 MHz
0	0	0	ϕ/1024 (initial value)	204.8 µs	409.6 µs
		1	ϕ/256	51.2 µs	102.4 µs
	1	0	ϕ/64	12.8 µs	25.6 µs
		1	ϕ/32	6.4 µs	12.8 µs
1	0	0	ϕ/16	3.2 µs	6.4 µs
		1	ϕ/8	1.6 µs	3.2 µs
	1	0	ϕ/4	0.8 µs	1.6 µs
		1	ϕ/2	—	0.8 µs

Serial Control/Status Register 1 (SCSR1)

Bit	7	6	5	4	3	2	1	0
	—	SOL	ORER	—	—	—	—	STF
Initial value	1	0	0	0	0	0	0	0
Read/Write	—	R/W	R/(W)*	—	—	—	R/W	R/W

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

SCSR1 is an 8-bit read/write register indicating operation status and error status.

Upon reset, SCSR1 is initialized to H'80.

Do not read or write to SCSR1 during a transfer operation, as this will cause erroneous operation.

Bit 7—Reserved Bit: Bit 7 is reserved; it is always read as 1, and cannot be modified.

Bit 6—Extended Data Bit (SOL): Bit 6 sets the SO_1 output level. When read, SOL returns the output level at the SO_1 pin. After completion of a transmission, SO_1 continues to output the value of the last bit of transmitted data. The SO_1 output can be changed by writing to SOL before or after a transmission. The SOL bit setting remains valid only until the start of the next transmission. To control the level of the SO_1 pin after transmission ends, it is necessary to write to the SOL bit at the end of each transmission. Do not write to this register while transmission is in progress, because that may cause a malfunction.

Bit 6: SOL	Description	
0	Read: SO_1 pin output level is low	(initial value)
	Write: SO_1 pin output level changes to low	
1	Read: SO_1 pin output level is high	
	Write: SO_1 pin output level changes to high	

Bit 5—Overrun Error Flag (ORER): When an external clock is used, bit 5 indicates the occurrence of an overrun error. If a clock pulse is input after transfer completion, this bit is set to 1 indicating an overrun. If noise occurs during a transfer, causing an extraneous pulse to be superimposed on the normal serial clock, incorrect data may be transferred.

Bit 5: ORER	Description	
0	Clearing conditions: After reading ORER = 1, cleared by writing 0 to ORER	(initial value)
1	Setting conditions: Set if a clock pulse is input after transfer is complete, when an external clock is used	

Bits 4 to 2—Reserved Bits: Bits 4 to 2 are reserved; they are always read as 0, and cannot be modified.

Bit 1—Reserved Bit: Bit 1 is reserved; it should always be cleared to 0.

Bit 0—Start Flag (STF): Bit 0 controls the start of a transfer. Setting this bit to 1 causes SCI1 to start transferring data.

This bit remains set to 1 during transfer or while waiting for a start bit, and is cleared to 0 upon completion of the transfer.

Bit 0: STF	Description	
0	Read: Indicates that transfer is stopped	(initial value)
	Write: Invalid	
1	Read: Indicates transfer in progress	
	Write: Starts a transfer operation	

Serial Data Register U (SDRU)

Bit	7	6	5	4	3	2	1	0
	SDRU7	SDRU6	SDRU5	SDRU4	SDRU3	SDRU2	SDRU1	SDRU0
Initial value	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SDRU is an 8-bit read/write register. It is used as the data register for the upper 8 bits in 16-bit transfer (SDRL is used for the lower 8 bits).

Data written to SDRU is output to SDRL starting from the least significant bit (LSB). This data is then replaced by LSB-first data input at pin SI_1, which is shifted in the direction from the most significant bit (MSB) toward the LSB.

SDRU must be written or read only after data transmission or reception is complete. If this register is written or read while a data transfer is in progress, the data contents are not guaranteed.

The SDRU value upon reset is not fixed.

Serial Data Register L (SDRL)

Bit	7	6	5	4	3	2	1	0
	SDRL7	SDRL6	SDRL5	SDRL4	SDRL3	SDRL2	SDRL1	SDRL0
Initial value	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SDRL is an 8-bit read/write register. It is used as the data register in 8-bit transfer, and as the data register for the lower 8 bits in 16-bit transfer (SDRU is used for the upper 8 bits).

In 8-bit transfer, data written to SDRL is output from pin SO_1 starting from the least significant bit (LSB). This data is than replaced by LSB-first data input at pin SI_1, which is shifted in the direction from the most significant bit (MSB) toward the LSB.

In 16-bit transfer, operation is the same as for 8-bit transfer, except that input data is fed in via SDRU.

SDRL must be written or read only after data transmission or reception is complete. If this register is read or written while a data transfer is in progress, the data contents are not guaranteed.

The SDRL value upon reset is not fixed.

10.2.3 Operation

Data can be sent and received in an 8-bit or 16-bit format, synchronized to an internal or external serial clock. Overrun errors can be detected when an external clock is used.

Clock

The serial clock can be selected from a choice of eight internal clocks and an external clock. When an internal clock source is selected, pin SCK_1 becomes the clock output pin. When continuous clock output mode is selected (SCR1 bits SNC1 and SNC0 are set to 10), the clock signal (ϕ/1024 to ϕ/2) selected in bits CKS2 to CKS0 is output continuously from pin SCK_1. When an external clock is used, pin SCK_1 is the clock input pin.

Data Transfer Format

Figure 10.2 shows the data transfer format. Data is sent and received starting from the least significant bit, in LSB-first format. Transmit data is output from one falling edge of the serial clock until the next falling edge. Receive data is latched at the rising edge of the serial clock.

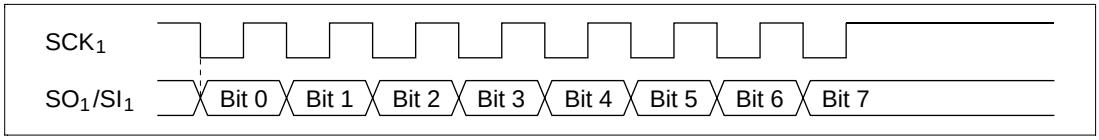


Figure 10.2 Transfer Format

Data Transfer Operations

Transmitting: A transmit operation is carried out as follows.

- Set bits SO1 and SCK1 in PMR3 TO 1 so that the respective pins function as SO_1 and SCK_1. If necessary, set bit POF1 in port mode register 2 (PMR2) for NMOS open drain output at pin SO_1.
- Clear bit SNC1 in SCR1 to 0, and set bit SNC0 to 1 or 0, designating 8- or 16-bit synchronous transfer mode. Select the serial clock in bits CKS3 to CKS0. Writing data to SCR1 initializes the internal state of SCI1.
- Write transmit data in SDRL and SDRU, as follows.
  - 8-bit transfer mode: SDRL
  - 16-bit transfer mode: Upper byte in SDRU, lower byte in SDRL
- Set the SCSR1 start flag (STF) to 1. SCI1 starts operating and outputs transmit data at pin SO_1.
- After data transmission is complete, bit IRRS1 in interrupt request register 1 (IRR1) is set to 1.

When an internal clock is used, a serial clock is output from pin SCK_1 in synchronization with the transmit data. After data transmission is complete, the serial clock is not output until the next time the start flag is set to 1. During this time, pin SO_1 continues to output the value of the last bit transmitted.

When an external clock is used, data is transmitted in synchronization with the serial clock input at pin SCK_1. After data transmission is complete, an overrun occurs if the serial clock continues to be input; no data is transmitted and the SCSR1 overrun error flag (bit ORER) is set to 1.

While transmission is stopped, the output value of pin SO_1 can be changed by rewriting bit SOL in SCSR1.

Receiving: A receive operation is carried out as follows.

- Set bits SI1 and SCK1 in PMR3 TO 1 so that the respective pins function as SI_1 and SCK_1.
- Clear bit SNC1 in SCR1 to 0, and set bit SNC0 to 1 or 0, designating 8- or 16-bit synchronous transfer mode. Select the serial clock in bits CKS3 to CKS0. Writing data to SCR1 initializes the internal state of SCI1.
- Set the SCSR1 start flag (STF) to 1. SCI1 starts operating and receives data at pin SI_1.
- After data reception is complete, bit IRRS1 in interrupt request register 1 (IRR1) is set to 1.

- Read the received data from SDRL and SDRU, as follows.
  — 8-bit transfer mode: SDRL
  — 16-bit transfer mode: Upper byte in SDRU, lower byte in SDRL
- After data reception is complete, an overrun occurs if the serial clock continues to be input; no data is received and the SCSR1 overrun error flag (bit ORER) is set to 1.

Simultaneous transmit/receive: A simultaneous transmit/receive operation is carried out as follows.

- Set bits SO1, SI1, and SCK1 in PMR3 to 1 so that the respective pins function as SO_1, SI_1, and SCK_1. If necessary, set bit POF1 in port mode register 2 (PMR2) for NMOS open drain output at pin SO_1.
- Clear bit SNC1 in SCR1 to 0, and set bit SNC0 to 1 or 0, designating 8- or 16-bit synchronous transfer mode. Select the serial clock in bits CKS3 to CKS0. Writing data to SCR1 initializes the internal state of SCI1.
- Write transmit data in SDRL and SDRU, as follows.
  — 8-bit transfer mode: SDRL
  — 16-bit transfer mode: Upper byte in SDRU, lower byte in SDRL
- Set the SCSR1 start flag (STF) to 1. SCI1 starts operating. Transmit data is output at pin SO_1. Receive data is input at pin SI_1.
- After data transmission and reception are complete, bit IRRS1 in IRR1 is set to 1.
- Read the received data from SDRL and SDRU, as follows.
  — 8-bit transfer mode: SDRL
  — 16-bit transfer mode: Upper byte in SDRU, lower byte in SDRL

When an internal clock is used, a serial clock is output from pin SCK_1 in synchronization with the transmit data. After data transmission is complete, the serial clock is not output until the next time the start flag is set to 1. During this time, pin SO_1 continues to output the value of the last bit transmitted.

When an external clock is used, data is transmitted and received in synchronization with the serial clock input at pin SCK_1. After data transmission and reception are complete, an overrun occurs if the serial clock continues to be input; no data is transmitted or received and the SCSR1 overrun error flag (bit ORER) is set to 1.

While transmission is stopped, the output value of pin SO_1 can be changed by rewriting bit SOL in SCSR1.

10.2.4 Interrupts

SCI1 can generate an interrupt at the end of a data transfer.

When an SCI1 transfer is complete, bit IRRS1 in interrupt request register 1 (IRR1) is set to 1. SCI1 interrupt requests can be enabled or disabled by bit IENS1 of interrupt enable register 1 (IENR1).

For further details, see 3.3, Interrupts.

10.2.5 Application Notes

Note the following points when using SCI1.

When an External Clock is Input to the SCK_1 Pin: When SCK_1 is designated as an input pin and an external clock is selected as the clock source, do not input the external clock before writing 1 to the STF bit in SCSR1 to start the transfer operation.

Confirming the End of Serial Transfer: Do not read or write to SCSR1 during serial transfer. The following two methods can be used to confirm the end of serial transfer:

- Using SCI1 interrupt exception handling
  Set the IENS1 bit to 1 in IENR1 and execute interrupt exception handling.
- Using IRR1 polling
  With SCI1 interrupts disabled (IENS1 = 0 in IENR1), confirm that the IRRS1 bit in IRR1 has been set to 1.

10.3 SCI3

10.3.1 Overview

Serial communication interface 3 (SCI3) has both synchronous and asynchronous serial data communication capabilities. It also has a multiprocessor communication function for serial data communication among two or more processors.

Features

SCI3 features are listed below.

- Selection of asynchronous or synchronous mode
  - — Asynchronous mode

    Serial data communication is performed using an asynchronous method in which synchronization is established character by character.

    SCI3 can communicate with a UART (universal asynchronous receiver/transmitter), ACIA (asynchronous communication interface adapter), or other chip that employs standard asynchronous serial communication. It can also communicate with two or more other processors using the multiprocessor communication function. There are twelve selectable serial data communication formats.
    - Data length: seven or eight bits
    - Stop bit length: one or two bits
    - Parity: even, odd, or none
    - Multiprocessor bit: one or none
    - Receive error detection: parity, overrun, and framing errors
    - Break detection: by reading the RXD level directly when a framing error occurs
  - — Synchronous mode

    Serial data communication is synchronized with a clock signal. SCI3 can communicate with other chips having a clocked synchronous communication function.
    - Data length: eight bits
    - Receive error detection: overrun errors
- Full duplex communication

  The transmitting and receiving sections are independent, so SCI3 can transmit and receive simultaneously. Both sections use double buffering, so continuous data transfer is possible in both the transmit and receive directions.
- Built-in baud rate generator with selectable bit rates.
- Internal or external clock may be selected as the transfer clock source.
- There are six interrupt sources: transmit end, transmit data empty, receive data full, overrun error, framing error, and parity error.

Block Diagram

Figure 10.3 shows a block diagram of SCI3.

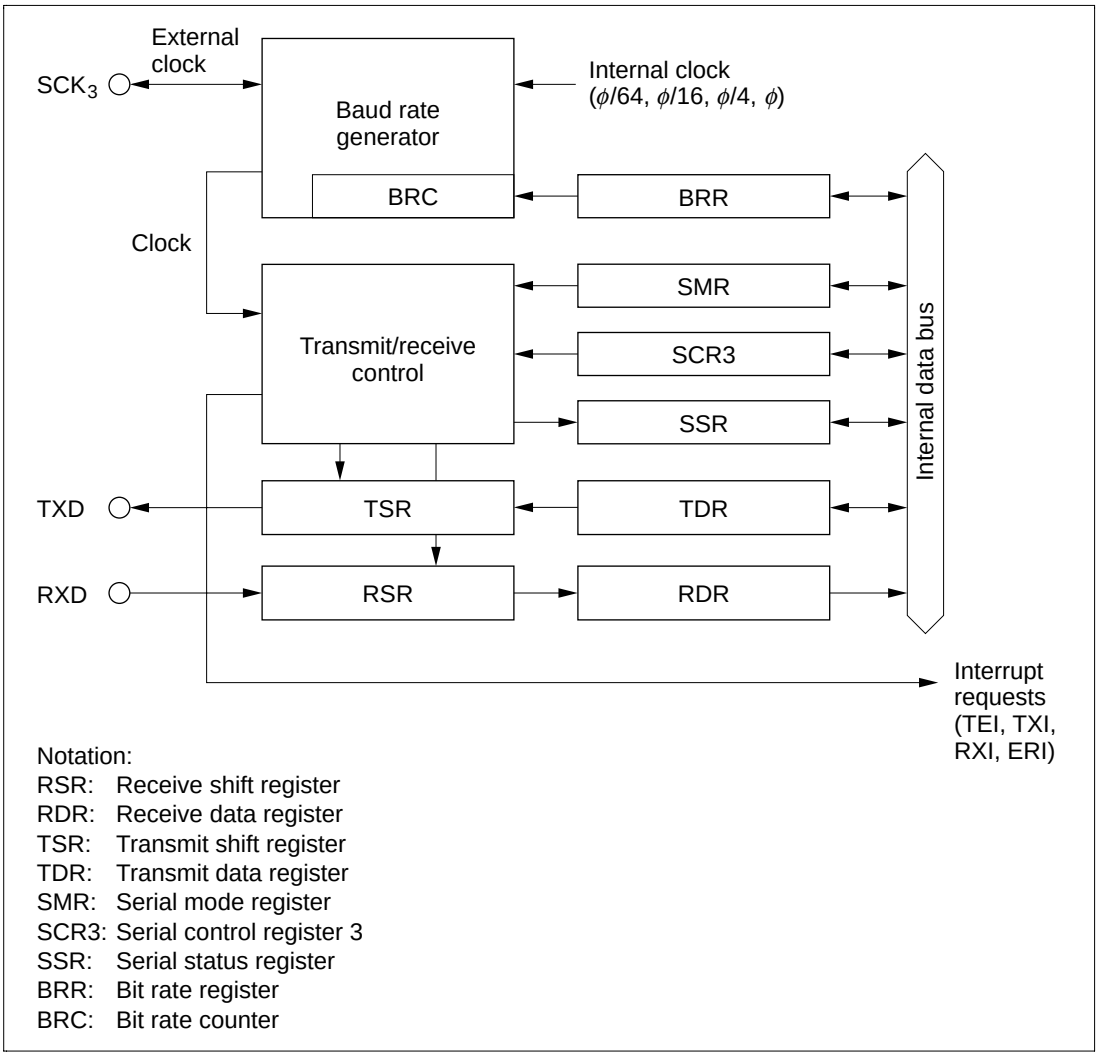


Figure 10.3 SCI3 Block Diagram

Pin Configuration

Table 10.4 shows the SCI3 pin configuration.

Table 10.4 Pin Configuration

Name	Abbrev.	I/O	Function
SCI3 clock	SCK_3	I/O	SCI3 clock input/output
SCI3 receive data input	RXD	Input	SCI3 receive data input
SCI3 transmit data output	TXD	Output	SCI3 transmit data output

Register Configuration

Table 10.5 shows the SCI3 internal register configuration.

Table 10.5 SCI3 Registers

Name	Abbrev.	R/W	Initial Value	Address
Serial mode register	SMR	R/W	H'00	H'FFA8
Bit rate register	BRR	R/W	H'FF	H'FFA9
Serial control register 3	SCR3	R/W	H'00	H'FFAA
Transmit data register	TDR	R/W	H'FF	H'FFAB
Serial status register	SSR	R/W	H'84	H'FFAC
Receive data register	RDR	R	H'00	H'FFAD
Transmit shift register	TSR	*	—	—
Receive shift register	RSR	*	—	—
Bit rate counter	BRC	*	—	—

Note: —: Cannot be read or written.

10.3.2 Register Descriptions

Receive Shift Register (RSR)

Bit	7	6	5	4	3	2	1	0
Read/Write	—	—	—	—	—	—	—	—

The receive shift register (RSR) is for receiving serial data.

Serial data is input in LSB (bit 0) order into RSR from pin RXD, converting it to parallel data. After each byte of data has been received, the byte is automatically transferred to the receive data register (RDR).

RSR cannot be read or written directly by the CPU.

Receive Data Register (RDR)

Bit	7	6	5	4	3	2	1	0
	RDR7	RDR6	RDR5	RDR4	RDR3	RDR2	RDR1	RDR0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R	R

The receive data register (RDR) is an 8-bit register for storing received serial data.

Each time a byte of data is received, the received data is transferred from the receive shift register (RSR) to RDR, completing a receive operation. Thereafter RSR again becomes ready to receive new data. RSR and RDR form a double buffer mechanism that allows data to be received continuously.

RDR is exclusively for receiving data and cannot be written by the CPU.

RDR is initialized to H'00 upon reset or in standby mode, watch mode, subactive mode, or subsleep mode.

Transmit Shift Register (TSR)

Bit	7	6	5	4	3	2	1	0
Read/Write	—	—	—	—	—	—	—	—

The transmit shift register (TSR) is for transmitting serial data.

Transmit data is first transferred from the transmit data register (TDR) to TSR, then is transmitted from pin TXD, starting from the LSB (bit 0).

After one byte of data has been sent, the next byte is automatically transferred from TDR to TSR, and the next transmission begins. If no data has been written to TDR (1 is set in TDRE), there is no data transfer from TDR to TSR.

TSR cannot be read or written directly by the CPU.

Transmit Data Register (TDR)

Bit	7	6	5	4	3	2	1	0
	TDR7	TDR6	TDR5	TDR4	TDR3	TDR2	TDR1	TDR0
Initial value	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The transmit data register (TDR) is an 8-bit register for holding transmit data.

When SCI3 detects that the transmit shift register (TSR) is empty, it shifts transmit data written in TDR to TSR and starts serial data transmission. While TSR is transmitting serial data, the next byte to be transmitted can be written to TDR, realizing continuous transmission.

TDR can be read or written by the CPU at all times.

TDR is initialized to H'FF upon reset or in standby mode, watch mode, subactive mode, or subsleep mode.

Serial Mode Register (SMR)

Bit	7	6	5	4	3	2	1	0
	COM	CHR	PE	PM	STOP	MP	CKS1	CKS0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The serial mode register (SMR) is an 8-bit register for setting the serial data communication format and for selecting the clock source of the baud rate generator. SMR can be read and written by the CPU at any time.

SMR is initialized to H'00 upon reset or in standby mode, watch mode, subactive mode, or subsleep mode.

Bit 7—Communication Mode (COM): Bit 7 selects asynchronous mode or synchronous mode as the serial data communication mode.

Bit 7: COM	Description	
0	Asynchronous mode	(initial value)
1	Synchronous mode	

Bit 6—Character Length (CHR): Bit 6 selects either 7 bits or 8 bits as the data length in asynchronous mode. In synchronous mode the data length is always 8 bits regardless of the setting here.

Bit 6: CHR	Description	
0	8-bit data	(initial value)
1	7-bit data*	

Note: * When 7-bit data is selected as the character length in asynchronous mode, the MSB (bit 7) in the transmit data register is not transmitted.

Bit 5—Parity Enable (PE): In asynchronous mode, bit 5 selects whether or not a parity bit is to be added to transmitted data and checked in received data. In synchronous mode there is no adding or checking of parity regardless of the setting here.

Bit 5: PE	Description	
0	Parity bit adding and checking disabled	(initial value)
1	Parity bit adding and checking enabled*	

Note: * When PE is set to 1, then either odd or even parity is added to transmit data, depending on the setting of the parity mode bit (PM). When data is received, it is checked for odd or even parity as designated in bit PM.

Bit 4—Parity Mode (PM): In asynchronous mode, bit 4 selects whether odd or even parity is to be added to transmitted data and checked in received data. The PM setting is valid only if bit PE is set to 1, enabling parity adding/checking. In synchronous mode, or if parity adding/checking is disabled in asynchronous mode, the PM setting is invalid.

Bit 4: PM	Description	
0	Even parity*[1]	(initial value)
1	Odd parity*[2]	

Notes: 1. When even parity is designated, a parity bit is added to the transmitted data so that the sum of 1s in the resulting data is an even number. When data is received, the sum of 1s in the data plus parity bit is checked to see if the result is an even number.

2. When odd parity is designated, a parity bit is added to the transmitted data so that the sum of 1s in the resulting data is an odd number. When data is received, the sum of 1s in the data plus parity bit is checked to see if the result is an odd number.

Bit 3—Stop Bit Length (STOP): Bit 3 selects 1 bit or 2 bits as the stop bit length in asynchronous mode. This setting is valid only in asynchronous mode. In synchronous mode a stop bit is not added, so this bit is ignored.

Bit 3: STOP	Description	
0	1 stop bit*[1]	(initial value)
1	2 stop bits*[2]	

Notes: 1. When data is transmitted, one 1 bit is added at the end of each transmitted character as the stop bit.
2. When data is transmitted, two 1 bits are added at the end of each transmitted character as the stop bits.

When data is received, only the first stop bit is checked regardless of the stop bit length. If the second stop bit value is 1 it is treated as a stop bit; if it is 0, it is treated as the start bit of the next character.

Bit 2—Multiprocessor Mode (MP): Bit 2 enables or disables the multiprocessor communication function. When the multiprocessor communication function is enabled, the parity enable (PE) and parity mode (PM) settings are ignored. The MP bit is valid only in asynchronous mode; it should be cleared to 0 in synchronous mode.

See 10.3.6, Multiprocessor Communication Function for details on the multiprocessor communication function.

Bit 2: MP	Description	
0	Multiprocessor communication function disabled	(initial value)
1	Multiprocessor communication function enabled	

Bits 1 and 0—Clock Select 1, 0 (CKS1, CKS0): Bits 1 and 0 select the clock source for the built-in baud rate generator. A choice of ϕ/64, ϕ/16, ϕ/4, or ϕ is made in these bits.

See 8, Bit rate register (BRR), below for information on the clock source and bit rate register settings, and their relation to the baud rate.

Bit 1: CKS1	Bit 0: CKS0	Description	
0	0	ϕ clock	(initial value)
	1	ϕ/4 clock	
1	0	ϕ/16 clock	
	1	ϕ/64 clock	

Serial Control Register 3 (SCR3)

Bit	7	6	5	4	3	2	1	0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Serial control register 3 (SCR3) is an 8-bit register that controls SCI3 transmit and receive operations, enables or disables serial clock output in asynchronous mode, enables or disables interrupts, and selects the serial clock source. SCR3 can be read and written by the CPU at any time.

SCR3 is initialized to H'00 upon reset or in standby mode, watch mode, subactive mode, or subsleep mode.

Bit 7—Transmit Interrupt Enable (TIE): Bit 7 enables or disables the transmit data empty interrupt (TXI) request when data is transferred from TDR to TSR and the transmit data register empty bit (TDRE) in the serial status register (SSR) is set to 1. The TXI interrupt can be cleared by clearing bit TDRE to 0, or by clearing bit TIE to 0.

Bit 7: TIE	Description	
0	Transmit data empty interrupt request (TXI) disabled	(initial value)
1	Transmit data empty interrupt request (TXI) enabled	

Bit 6—Receive Interrupt Enable (RIE): Bit 6 enables or disables the receive error interrupt (ERI), and the receive data full interrupt (RXI) requested when data is transferred from RSR to RDR and the receive data register full bit (RDRF) in the serial status register (SSR) is set to 1. There are three kinds of receive error: overrun, framing, and parity. RXI and ERI interrupts can be cleared by clearing SSR flag RDRF, or flags FER, PER, and OER to 0, or by clearing bit RIE to 0.

Bit 6: RIE	Description	
0	Receive data full interrupt request (RXI) and receive error interrupt request (ERI) disabled	(initial value)
1	Receive data full interrupt request (RXI) and receive error interrupt request (ERI) enabled	

Bit 5—Transmit Enable (TE): Bit 5 enables or disables the start of a transmit operation.

Bit 5: TE	Description	
0	Transmit operation disabled*[1] (TXD is a general I/O port)	(initial value)
1	Transmit operation enabled*[2] (TXD is the transmit data pin)	

Notes: 1. The transmit data register empty bit (TDRE) in the serial status register (SSR) is fixed at 1.

2. In this state, writing transmit data in TDR clears bit TDRE in SSR to 0 and starts serial data transmission.

   Before setting TE to 1 it is necessary to set the transmit format in SMR. When performing simultaneous transmission and reception in synchronous mode, TE and RE should be set to 1 simultaneously by a single instruction when they are both cleared to 0.

Bit 4—Receive Enable (RE): Bit 4 enables or disables the start of a receive operation.

Bit 4: RE	Description	
0	Receive operation disabled*[1] (RXD is a general I/O port)	(initial value)
1	Receive operation enabled*[2] (RXD is the receive data pin)	

Notes: 1. When RE is cleared to 0, this has no effect on the SSR flags RDRF, FER, PER, and OER, which retain their states.

2. Serial data receiving begins when, in this state, a start bit is detected in asynchronous mode, or serial clock input is detected in synchronous mode.

   Before setting RE to 1 it is necessary to set the receive format in SMR. When performing simultaneous transmission and reception in synchronous mode, TE and RE should be set to 1 simultaneously by a single instruction when they are both cleared to 0.

Bit 3—Multiprocessor Interrupt Enable (MPIE): Bit 3 enables or disables multiprocessor interrupt requests. This setting is valid only in asynchronous mode, and only when the multiprocessor mode bit (MP) in the serial mode register (SMR) is set to 1. This bit is ignored when COM is set to 1 or when bit MP is cleared to 0.

Bit 3: MPIE	Description
0	Multiprocessor interrupt request disabled (ordinary receive operation) (initial value) Clearing condition: Multiprocessor bit receives a data value of 1
1	Multiprocessor interrupt request enabled*

Note: * SCI3 does not transfer receive data from RSR to RDR, does not detect receive errors, and does not set status flags RDRF, FER, and OER in SSR. Until a multiprocessor bit value of 1 is received, the receive data full interrupt (RXI) and receive error interrupt (ERI) are disabled and serial status register (SSR) flags RDRF, FER, and OER are not set. When the multiprocessor bit receives a 1, the MPBR bit of SSR is set to 1, MPIE is automatically cleared to 0, RXI and ERI interrupts are enabled (provided bits TIE and RIE in SCR3 are set to 1), and setting of the RDRF, FER, and OER flags is enabled.

Bit 2—Transmit End Interrupt Enable (TEIE): Bit 2 enables or disables the transmit end interrupt (TEI) requested if there is no valid transmit data in TDR when the MSB is transmitted.

Bit 2: TEIE	Description
0	Transmit end interrupt (TEI) disabled (initial value)
1	Transmit end interrupt (TEI) enabled*

Note: * A TEI interrupt can be cleared by clearing the SSR bit TDRE to 0 and clearing the transmit end bit (TEND) to 0, or by clearing bit TEIE to 0.

Bits 1 and 0—Clock Enable 1, 0 (CKE1, CKE0): Bits 1 and 0 select the clock source and enable or disable clock output at pin SCK_3. The combination of bits CKE1 and CKE0 determines whether pin SCK_3 is a general I/O port, a clock output pin, or a clock input pin.

Note that the CKE0 setting is valid only when operation is in asynchronous mode using an internal clock (CKE1 = 0). This bit is invalid in synchronous mode or when using an external clock (CKE1 = 1). In synchronous mode and in external clock mode, clear CKE0 to 0. After setting bits CKE1 and CKE0, the operation mode must first be set in the serial mode register (SMR).

See table 10.12 in 10.3.3, Operation, for details on clock source selection.

Bit 1: CKE1	Bit 0: CKE0	Communication Mode	Clock Source	SCK_3 Pin Function
0	0	Asynchronous	Internal clock	I/O port*[1]
		Synchronous	Internal clock	Serial clock output*[1]
0	1	Asynchronous	Internal clock	Clock output*[2]
		Synchronous	Reserved	Reserved
1	0	Asynchronous	External clock	Clock input*[3]
		Synchronous	External clock	Serial clock input
1	1	Asynchronous	Reserved	Reserved
		Synchronous	Reserved	Reserved

Notes: 1. Initial value
2. A clock is output with the same frequency as the bit rate.
3. Input a clock with a frequency 16 times the bit rate.

Serial Status Register (SSR)

Bit	7	6	5	4	3	2	1	0
	TDRE	RDRF	OER	FER	PER	TEND	MPBR	MPBT
Initial value	1	0	0	0	0	1	0	0
Read/Write	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R/W

Note: * Only 0 can be written for flag clearing.

The serial status register (SSR) is an 8-bit register containing status flags for indicating SCI3 states, and containing the multiprocessor bits.

SSR can be read and written by the CPU at any time, but the CPU cannot write a 1 to the status flags TDRE, RDRF, OER, PER, and FER. To clear these flags to 0 it is first necessary to read a 1. Bit 2 (TEND) and bit 1 (MPBR) are read-only bits and cannot be modified.

SSR is initialized to H'84 upon reset or in standby mode, watch mode, subactive mode, or subsleep mode.

Bit 7—Transmit Data Register Empty (TDRE): Bit 7 is a status flag indicating that data has been transferred from TDR to TSR.

Bit 7: TDRE	Description
0	Indicates that transmit data written to TDR has not been transferred to TSR Clearing conditions: After reading TDRE = 1, cleared by writing 0 to TDRE. When data is written to TDR by an instruction.
1	Indicates that no transmit data has been written to TDR, or the transmit data written to TDR has been transferred to TSR (initial value) Setting conditions: When bit TE in SCR3 is cleared to 0. When data is transferred from TDR to TSR.

Bit 6—Receive Data Register Full (RDRF): Bit 6 is a status flag indicating whether there is receive data in RDR.

Bit 6: RDRF	Description
0	Indicates there is no receive data in RDR (initial value) Clearing conditions: After reading RDRF = 1, cleared by writing 0 to RDRF. When data is read from RDR by an instruction.
1	Indicates that there is receive data in RDR Setting condition: When receiving ends normally, with receive data transferred from RSR to RDR

Note: If a receive error is detected at the end of receiving, or if bit RE in serial control register 3 (SCR3) is cleared to 0, RDR and RDRF are unaffected and keep their previous states. An overrun error (OER) occurs if receiving of data is completed while bit RDRF remains set to 1. If this happens, receive data will be lost.

Bit 5—Overrun Error (OER): Bit 5 is a status flag indicating that an overrun error has occurred during data receiving.

Bit 5: OER	Description
0	Indicates that data receiving is in progress or has been completed*[1] (initial value) Clearing condition: After reading OER = 1, cleared by writing 0 to OER
1	Indicates that an overrun error occurred in data receiving*[2] Setting condition: When data receiving is completed while RDRF is set to 1

Notes: 1. When bit RE in serial control register 3 (SCR3) is cleared to 0, OER is unaffected and keeps its previous state.
2. RDR keeps the data received prior to the overrun; data received after that is lost. While OER is set to 1, data receiving cannot be continued. In synchronous mode, data transmitting cannot be continued either.

Bit 4—Framing Error (FER): Bit 4 is a status flag indicating that a framing error has occurred during asynchronous receiving.

Bit 4: FER	Description
0	Indicates that data receiving is in progress or has been completed*[1] (initial value) Clearing condition: After reading FER = 1, cleared by writing 0 to FER
1	Indicates that a framing error occurred in data receiving Setting condition: The stop bit at the end of receive data is checked for a value of 1 and found to be 0*[2]

Notes: 1. When bit RE in serial control register 3 (SCR3) is cleared to 0, FER is unaffected and keeps its previous state.
2. When two stop bits are used only the first stop bit is checked, not the second. When a framing error occurs, receive data is transferred to RDR but RDRF is not set. While FER is set to 1, data receiving cannot be continued. In synchronous mode, data transmitting cannot be continued either.

Bit 3—Parity Error (PER): Bit 3 is a status flag indicating that a parity error has occurred during asynchronous receiving.

Bit 3: PER	Description
0	Indicates that data receiving is in progress or has been completed*[1] (initial value) Clearing condition: After reading PER = 1, cleared by writing 0 to PER
1	Indicates that a parity error occurred in data receiving*[2] Setting condition: When the sum of 1s in received data plus the parity bit does not match the parity mode bit (PM) setting in the serial mode register (SMR)

Notes: 1. When bit RE in serial control register 3 (SCR3) is cleared to 0, PER is unaffected and keeps its previous state.
2. When a parity error occurs, receive data is transferred to RDR but RDRF is not set. While PER is set to 1, data receiving cannot be continued. In synchronous mode, data transmitting cannot be continued either.

Bit 2—Transmit End (TEND): Bit 2 is a status flag indicating that TDRE was set to 1 when the last bit of a transmitted character was sent. TEND is a read-only bit and cannot be modified directly.

Bit 2: TEND	Description
0	Indicates that transmission is in progress Clearing conditions: After reading TDRE = 1, cleared by writing 0 to TDRE. When data is written to TDR by an instruction.
1	Indicates that a transmission has ended (initial value) Setting conditions: When bit TE in SCR3 is cleared to 0. If TDRE is set to 1 when the last bit of a transmitted character is sent.

Bit 1—Multiprocessor Bit Receive (MPBR): Bit 1 holds the multiprocessor bit in data received in asynchronous mode using a multiprocessor format. MPBR is a read-only bit and cannot be modified.

Bit 1: MPBR	Description
0	Indicates reception of data in which the multiprocessor bit is 0* (initial value)
1	Indicates reception of data in which the multiprocessor bit is 1

Note: * If bit RE is cleared to 0 while a multiprocessor format is in use, MPBR retains its previous state.

Bit 0—Multiprocessor Bit Transmit (MPBT): Bit 0 holds the multiprocessor bit to be added to transmitted data when a multiprocessor format is used in asynchronous mode. Bit MPBT is ignored when synchronous mode is chosen, when the multiprocessor communication function is disabled, or when data transmission is disabled.

Bit 0: MPBT	Description	
0	The multiprocessor bit in transmit data is 0	(initial value)
1	The multiprocessor bit in transmit data is 1	

Bit Rate Register (BRR)

Bit	7	6	5	4	3	2	1	0
	BRR7	BRR6	BRR5	BRR4	BRR3	BRR2	BRR1	BRR0
Initial value	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The bit rate register (BRR) is an 8-bit register which, together with the baud rate generator clock selected by bits CKS1 and CKS0 in the serial mode register (SMR), sets the transmit/receive bit rate.

BRR can be read or written by the CPU at any time.

BRR is initialized to H'FF upon reset or in standby mode, watch mode, subactive mode, or subsleep mode.

Table 10.6 gives examples of how BRR is set in asynchronous mode. The values in table 10.6 are for active (high-speed) mode.

Table 10.6 BRR Settings and Bit Rates in Asynchronous Mode

	OSC (MHz)											
	2			2.4576			4			4.194304		
Bit Rate (bits/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	1	70	+0.03	1	86	+0.31	1	141	+0.03	1	148	–0.04
150	0	207	+0.16	0	255	0	1	103	+0.16	1	108	+0.21
300	0	103	+0.16	0	127	0	0	207	+0.16	0	217	+0.21
600	0	51	+0.16	0	63	0	0	103	+0.16	0	108	+0.21
1200	0	25	+0.16	0	31	0	0	51	+0.16	0	54	–0.70
2400	0	12	+0.16	0	15	0	0	25	+0.16	0	26	+1.14
4800	—	—	—	0	7	0	0	12	+0.16	0	13	–2.48
9600	—	—	—	0	3	0	—	—	—	0	6	–2.48
19200	—	—	—	0	1	0	—	—	—	—	—	—
31250	0	0	0	—	—	—	0	1	0	—	—	—
38400	—	—	—	0	0	0	—	—	—	—	—	—

	OSC (MHz)											
	4.9152			6			7.3728			8		
Bit Rate (bits/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	1	174	–0.26	1	212	+0.03	2	64	+0.70	2	70	+0.03
150	1	127	0	1	155	+0.16	1	191	0	1	207	+0.16
300	0	255	0	1	77	+0.16	1	95	0	1	103	+0.16
600	0	127	0	0	155	+0.16	0	191	0	0	207	+0.16
1200	0	63	0	0	77	+0.16	0	95	0	0	103	+0.16
2400	0	31	0	0	38	+0.16	0	47	0	0	51	+0.16
4800	0	15	0	0	19	–2.34	0	23	0	0	25	+0.16
9600	0	7	0	0	9	–2.34	0	11	0	0	12	+0.16
19200	0	3	0	0	4	–2.34	0	5	0	—	—	—
31250	—	—	—	0	2	0	—	—	—	0	3	0
38400	0	1	0	—	—	—	0	2	0	—	—	—

	OSC (MHz)					
	9.8304			10		
Bit Rate (bits/s)	n	N	Error (%)	n	N	Error (%)
110	2	86	+0.31	2	88	–0.25
150	1	255	0	2	64	+0.16
300	1	127	0	1	129	+0.16
600	0	255	0	1	64	+0.16
1200	0	127	0	0	129	+0.16
2400	0	63	0	0	64	+0.16
4800	0	31	0	0	32	–1.36
9600	0	15	0	0	15	+1.73
19200	0	7	0	0	7	+1.73
31250	0	4	–1.70	0	4	0
38400	0	3	0	0	3	+1.73

Notes: 1. Settings should be made so that error is within 1%.

2. BRR setting values are derived by the following equation.

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

B: Bit rate (bits/s)
N: BRR baud rate generator setting ($0 \leq N \leq 255$)
OSC: Value of ϕ_{OSC} (MHz)
n: Baud rate generator input clock number (n = 0, 1, 2, 3)

3. The error values in table 10.6 were derived by performing the following calculation and rounding off to two decimal places.

$$\text{Error (\%)} = \frac{B - R}{R} \times 100$$

B: Bit rate found from n, N, and OSC
R: Bit rate listed in left column of table 10.6

The meaning of n is shown in table 10.7.

Table 10.7 Relation between n and Clock

		SMR Setting	
n	**Clock**	**CKS1**	**CKS0**
0	ϕ	0	0
1	ϕ/4	0	1
2	ϕ/16	1	0
3	ϕ/64	1	1

Table 10.8 shows the maximum bit rate for selected frequencies in asynchronous mode. Values in table 10.8 are for active (high-speed) mode.

Table 10.8 Maximum Bit Rate at Selected Frequencies (Asynchronous Mode)

		Setting	
OSC (MHz)	**Maximum Bit Rate (bits/s)**	**n**	**N**
2	31250	0	0
2.4576	38400	0	0
4	62500	0	0
4.194304	65536	0	0
4.9152	76800	0	0
6	93750	0	0
7.3728	115200	0	0
8	125000	0	0
9.8304	153600	0	0
10	156250	0	0

Table 10.9 shows typical BRR settings in synchronous mode. Values in table 10.9 are for active (high-speed) mode.

Table 10.9 Typical BRR Settings and Bit Rates (Synchronous Mode)

Bit Rate (bits/s)	OSC (MHz) 2		4		8		10	
	n	N	n	N	n	N	n	N
110	—	—	—	—	—	—	—	—
250	1	249	2	124	2	249	—	—
500	1	124	1	249	2	124	—	—
1K	0	249	1	124	1	249	—	—
2.5K	0	99	0	199	1	99	1	124
5K	0	49	0	99	0	199	0	249
10K	0	24	0	49	0	99	0	124
25K	0	9	0	19	0	39	0	49
50K	0	4	0	9	0	19	0	24
100K	—	—	0	4	0	9	—	—
250K	0	0*	0	1	0	3	0	4
500K			0	0*	0	1	—	—
1M					0	0*	—	—
2.5M								

Note: Blank: Cannot be set
—: Can be set, but error will result
*: Continuous transfer not possible at this setting

BRR setting values are derived by the following equation.

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

B: Bit rate (bits/s)
N: BRR baud rate generator setting ($0 \leq N \leq 255$)
OSC: Value of ϕ_{OSC} (MHz)
n: Baud rate generator input clock number (n = 0, 1, 2, 3)

The meaning of n is shown in table 10.10.

Table 10.10 Relation between n and Clock

		SMR Setting	
n	**Clock**	**CKS1**	**CKS0**
0	ϕ	0	0
1	ϕ/4	0	1
2	ϕ/16	1	0
3	ϕ/64	1	1

10.3.3 Operation

SCI3 supports serial data communication in both asynchronous mode, where each character transferred is synchronized separately, and synchronous mode, where transfer is synchronized by clock pulses.

The choice of asynchronous mode or synchronous mode, and the communication format, is made in the serial mode register (SMR), as shown in table 10.11. The SCI3 clock source is determined by bit COM in SMR and bits CKE1 and CKE0 in serial control register 3 (SCR3), as shown in table 10.12.

Asynchronous Mode:

- Data length: choice of 7 bits or 8 bits
- Options include addition of parity bit, multiprocessor bit, and one or two stop bits (transmit/receive format and character length are determined by this combination of options).
- Framing error (FER), parity error (PER), overrun error (OER), and line breaks can be detected when data is received.
- Clock source: Choice of internal clocks or an external clock

  When an internal clock is selected: Operates on baud rate generator clock. A clock can be output with the same frequency as the bit rate.

  When an external clock is selected: A clock input with a frequency 16 times the bit rate is required (internal baud rate generator is not used).

Synchronous Mode:

- Transfer format: 8 bits
- Overrun error can be detected when data is received.
- Clock source: Choice of internal clocks or an external clock

  When an internal clock is selected: Operates on baud rate generator clock, and outputs a serial clock.

  When an external clock is selected: The internal baud rate generator is not used. Operation is synchronous with the input clock.

Table 10.11 SMR Settings and SCI3 Communication Format

SMR Setting						Communication Format			
Bit 7: COM	**Bit 6: CHR**	**Bit 2: MP**	**Bit 5: PE**	**Bit 3: STOP**	**Mode**	**Data Length**	**Multipro-cessor Bit**	**Parity Bit**	**Stop Bit Length**
0	0	0	0	0	Asynchronous mode	8-bit data	No	No	1 bit
				1					2 bits
			1	0				Yes	1 bit
				1					2 bits
	1		0	0		7-bit data		No	1 bit
				1					2 bits
			1	0				Yes	1 bit
				1					2 bits
	0	1	*	0	Asynchronous mode (multiprocessor format)	8-bit data	Yes	No	1 bit
			*	1					2 bits
	1		*	0		7-bit data			1 bit
			*	1					2 bits
1	*	0	*	*	Synchronous mode	8-bit data	No		None

Note: * Don’t care

Table 10.12 SMR and SCR3 Settings and Clock Source Selection

SMR	SCR3			Transmit/Receive Clock	
Bit 7: COM	**Bit 1: CKE1**	**Bit 0: CKE0**	**Mode**	**Clock Source**	**Pin SCK_3 Function**
0	0	0	Asynchronous mode	Internal	I/O port (SCK_3 function not used)
		1			Outputs clock with same frequency as bit rate
	1	0		External	Clock should be input with frequency 16 times the desired bit rate
1	0	0	Synchronous mode	Internal	Outputs a serial clock
	1	0		External	Inputs a serial clock
0	1	1	Reserved	(illegal settings)	
1	0	1			
1	1	1			

Continuous Transmit/Receive Operation Using Interrupts: Continuous transmit and receive operations are possible with SCI3, using the RXI or TXI interrupts. Table 10.13 explains this use of these interrupts.

Table 10.13 Transmit/Receive Interrupts

Interrupt	Flag	Interrupt Conditions	Remarks
RXI	RDRF RIE	When serial data is received normally and receive data is transferred from RSR to RDR, RDRF is set to 1. If RIE is 1 at this time, RXI is enabled and an interrupt occurs. (See figure 10.4.)	The RXI interrupt handler routine should read the receive data from RDR and clear RDRF to 0. Continuous receiving is possible if these operations are completed before the next data has been completely received in RSR.
TXI	TDRE TIE	When TSR empty (previous transmission complete) is detected and the transmit data set in TDR is transferred to TSR, TDRE is set to 1. If TIE is 1 at this time, TXI is enabled and an interrupt occurs. (See figure 10.5.)	The TXI interrupt handler routine should write the next transmit data to TDR and clear TDRE to 0.Continuous transmission is possible if these operations are completed before the data transferred to TSR has been completely transmitted.
TEI	TEND TEIE	When the last bit of the TSR transmit character has been sent, if TDRE is 1, then 1 is set in TEND. If TEIE is 1 at this time, TEI is enabled and an interrupt occurs. (See figure 10.6.)	TEI indicates that, when the last bit of the TSR transmit character was sent, the next transmit data had not been written to TDR.

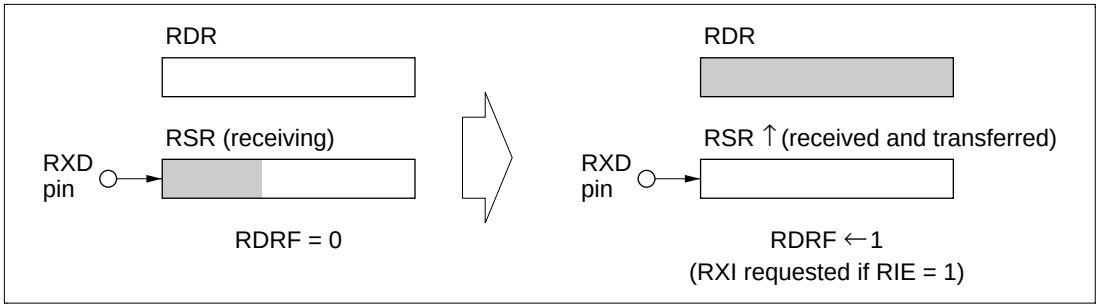


Figure 10.4 RDRF Setting and RXI Interrupt

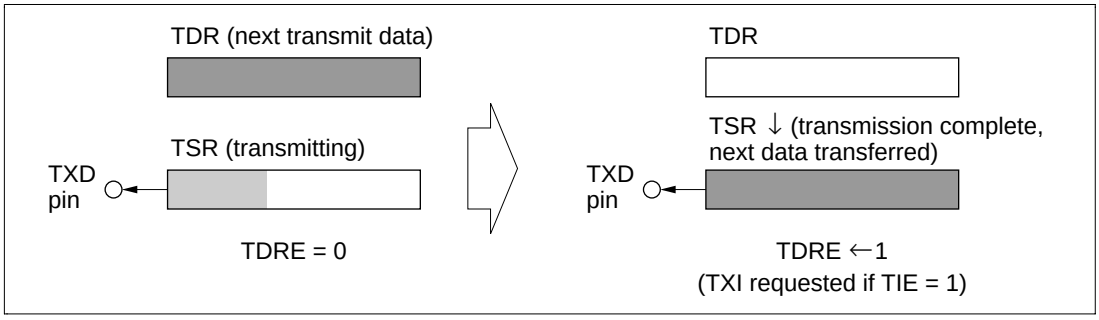


Figure 10.5 TDRE Setting and TXI Interrupt

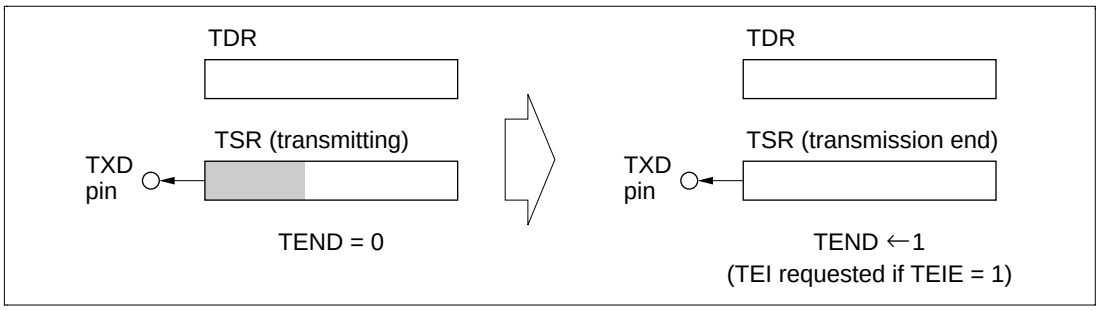


Figure 10.6 TEND Setting and TEI Interrupt

10.3.4 Operation in Asynchronous Mode

In asynchronous communication mode, a start bit indicating the start of communication and a stop bit indicating the end of communication are added to each character that is sent. In this way synchronization is achieved for each character as a self-contained unit.

SCI3 consists of independent transmit and receive modules, giving it the capability of full duplex communication. Both the transmit and receive modules have a double-buffer configuration, allowing data to be read or written during communication operations so that data can be transmitted and received continuously.

Transmit/Receive Formats

Figure 10.7 shows the general format for asynchronous serial communication.

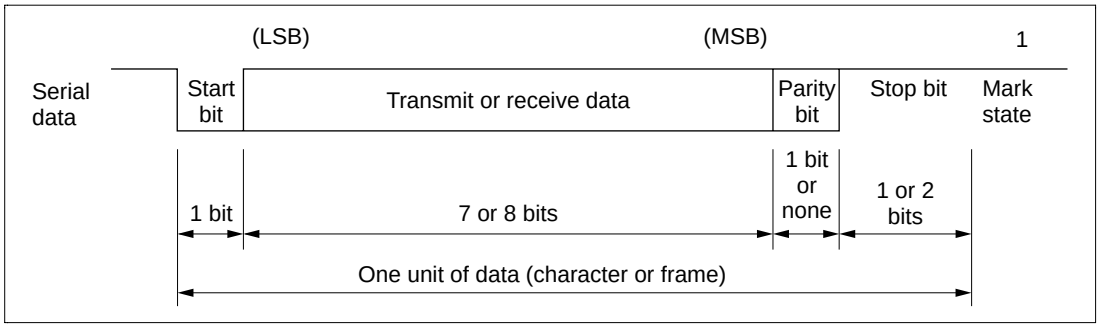


Figure 10.7 Data Format in Asynchronous Serial Communication Mode

The communication line in asynchronous communication mode normally stays at the high level, in the “mark” state. SCI3 monitors the communication line, and begins serial data communication when it detects a “space” (low-level signal), which is regarded as a start bit.

One character consists of a start bit (low level), transmit/receive data (in LSB-first order: starting with the least significant bit), a parity bit (high or low level), and finally a stop bit (high level), in this order.

In asynchronous data receiving, synchronization is with the falling edge of the start bit. SCI3 samples data on the 8th pulse of a clock that has 16 times the frequency of the bit rate, so each bit of data is latched at its center.

Table 10.14 shows the 12 transmit/receive formats formats that can be selected in asynchronous mode. The format is selected in the serial mode register (SMR).

Table 10.14 Serial Communication Formats in Asynchronous Mode

SMR Settings				Serial Transfer Format and Frame Length											
CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	S	8-bit data								STOP		
0	0	0	1	S	8-bit data								STOP	STOP	
0	1	0	0	S	8-bit data								P	STOP	
0	1	0	1	S	8-bit data								P	STOP	STOP
1	0	0	0	S	7-bit data							STOP			
1	0	0	1	S	7-bit data							STOP	STOP		
1	1	0	0	S	7-bit data							P	STOP		
1	1	0	1	S	7-bit data							P	STOP	STOP	
0	*	1	0	S	8-bit data								MPB	STOP	
0	*	1	1	S	8-bit data								MPB	STOP	STOP
1	*	1	0	S	7-bit data							MPB	STOP		
1	*	1	1	S	7-bit data							MPB	STOP	STOP	

Note: * Don’t care

Notation:

S: Start bit
STOP: Stop bit
P: Parity bit
MPB: Multiprocessor bit

Clock

The clock source is determined by bit COM in SMR and bits CKE1 and CKE0 in serial control register 3 (SCR3). See table 10.12 for the settings. Either an internal clock source can be used to run the built-in baud rate generator, or an external clock source can be input at pin SCK_3.

When an external clock is input at pin SCK_3, it should have a frequency 16 times the desired bit rate.

When an internal clock source is used, SCK_3 is used as the clock output pin. The clock output has the same frequency as the serial bit rate, and is synchronized as in figure 10.8 so that the rising edge of the clock occurs in the center of each bit of transmit/receive data.

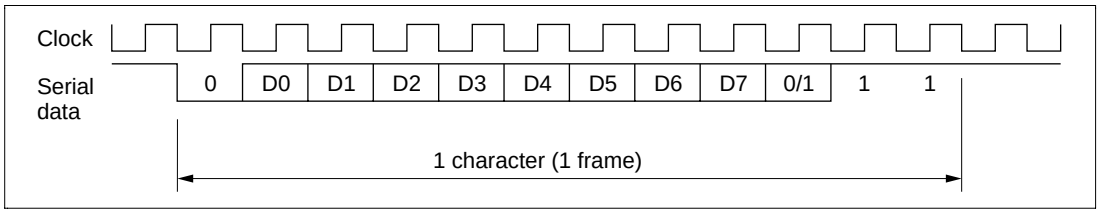


Figure 10.8 Phase Relation of Output Clock and Communication Data in Asynchronous Mode (8-Bit Data, Parity Bit Added, and 2 Stop Bits)

Data Transmit/Receive Operations

SCI3 Initialization: Before data is sent or received, bits TE and RE in serial control register 3 (SCR3) must be cleared to 0, after which initialization can be performed using the procedure shown in figure 10.9.

Note: When modifying the operation mode, transfer format or other settings, always be sure to clear bits TE and RE first. When TE is cleared to 0, bit TDRE will be set to 1. Clearing RE does not clear the status flags RDRF, PER, FER, or OER, or alter the contents of the receive data register (RDR).

When an external clock is used in asynchronous mode, do not stop the clock during operation, including during initialization. When an external clock is used in synchronous mode, do not supply the clock during initialization.

Figure 10.9 shows a typical flow chart for SCI3 initialization.

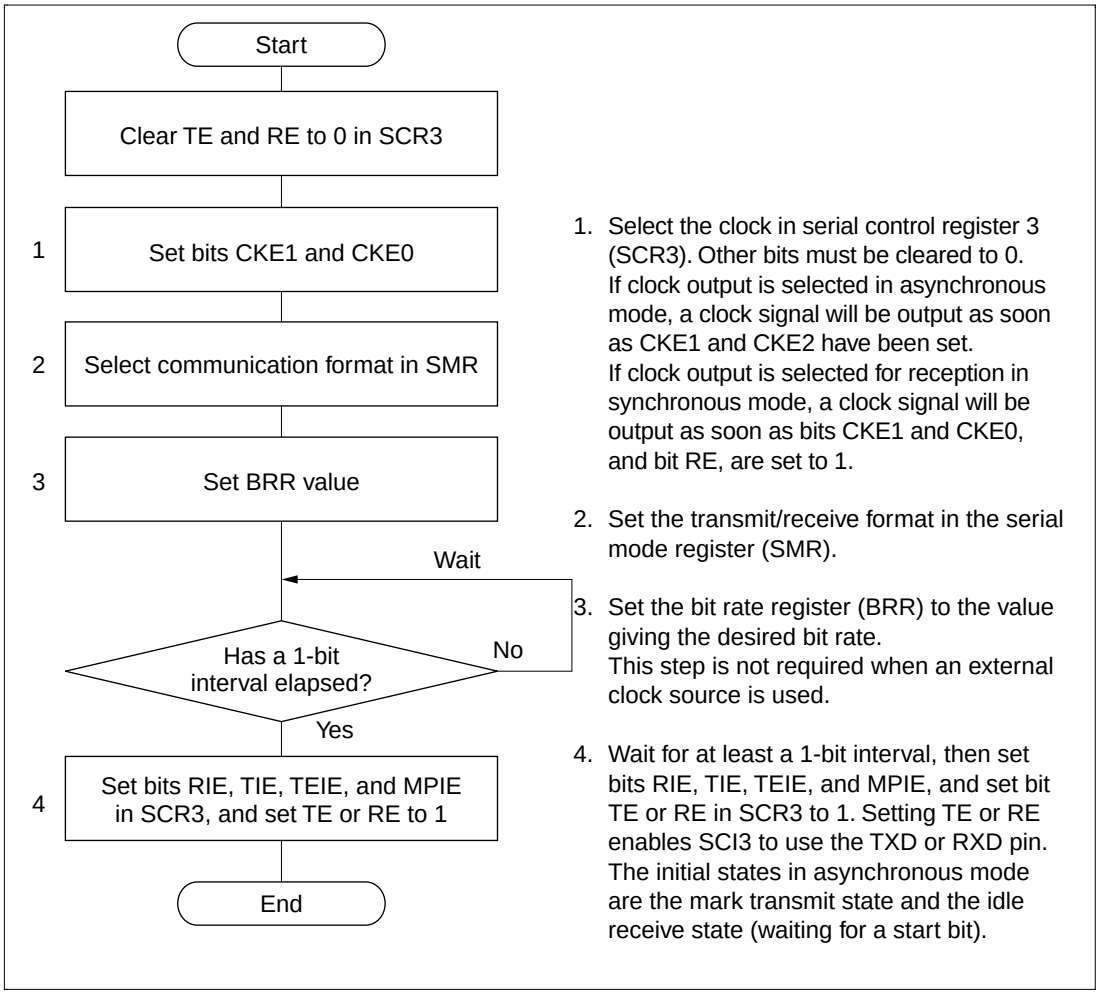


Figure 10.9 Typical Flow Chart when SCI3 Is Initialized

Transmitting: Figure 10.10 shows a typical flow chart for data transmission. After SCI3 initialization, follow the procedure below.

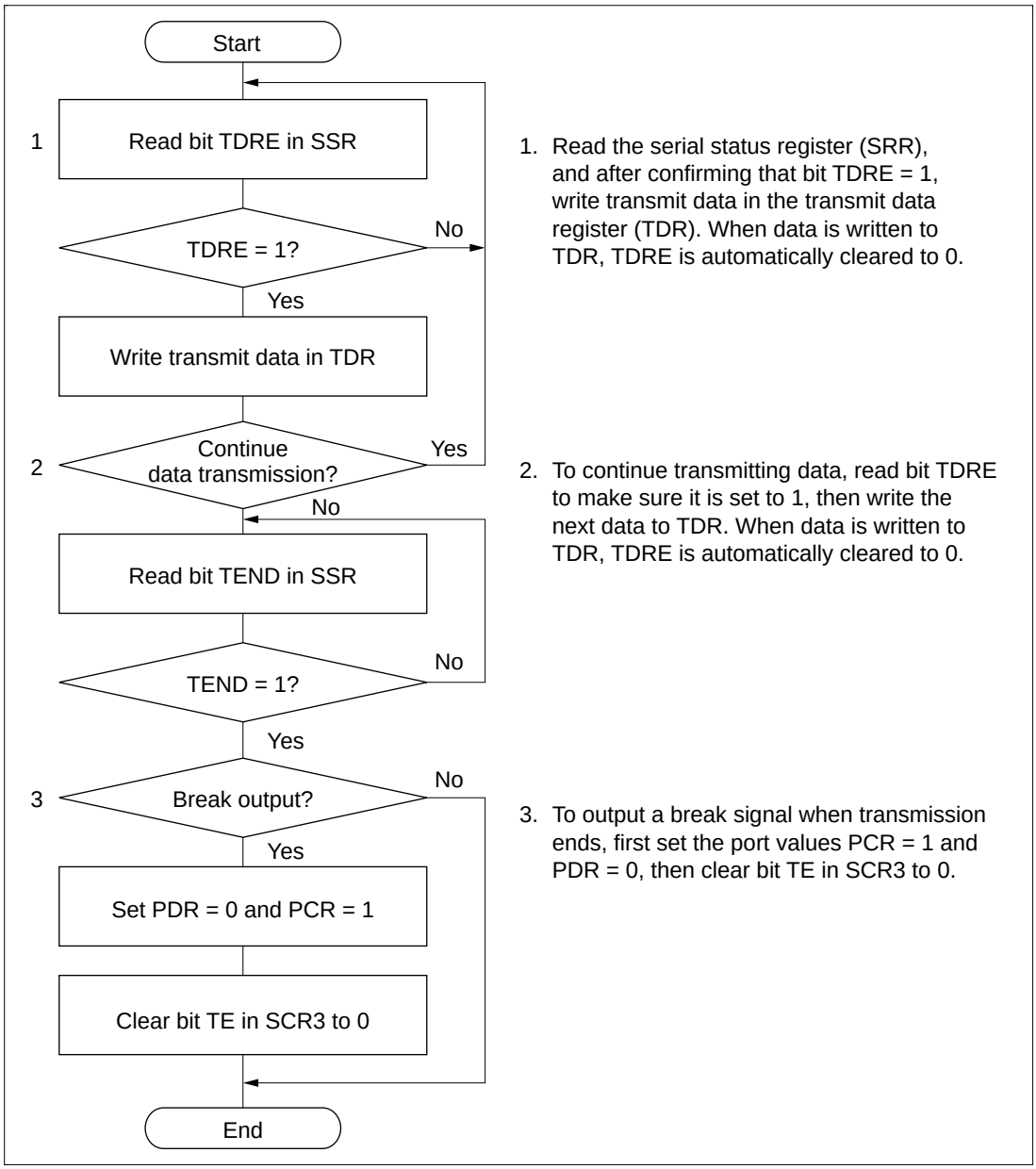


Figure 10.10 Typical Data Transmission Flow Chart (Asynchronous Mode)

SCI3 operates as follows during data transmission.

SCI3 monitors bit TDRE in SSR. When this bit is cleared to 0, SCI3 recognizes that there is data written in the transmit data register (TDR), which it transfers to the transmit shift register (TSR). Then TDRE is set to 1 and transmission starts. If bit TIE in SCR3 is set to 1, a TXI interrupt is requested.

Serial data is transmitted from pin TXD using the communication format outlined in table 10.14. Next, TDRE is checked as the stop bit is being transmitted.

If TDRE is 0, data is transferred from TDR to TSR, and after the stop bit is sent, transmission of the next frame starts. If TDRE is 1, the TEND bit in SSR is set to 1, and after the stop bit is sent the output remains at 1 (mark state). A TEI interrupt is requested in this state if bit TEIE in SCR3 is set to 1.

Figure 10.11 shows a typical operation in asynchronous transmission mode.

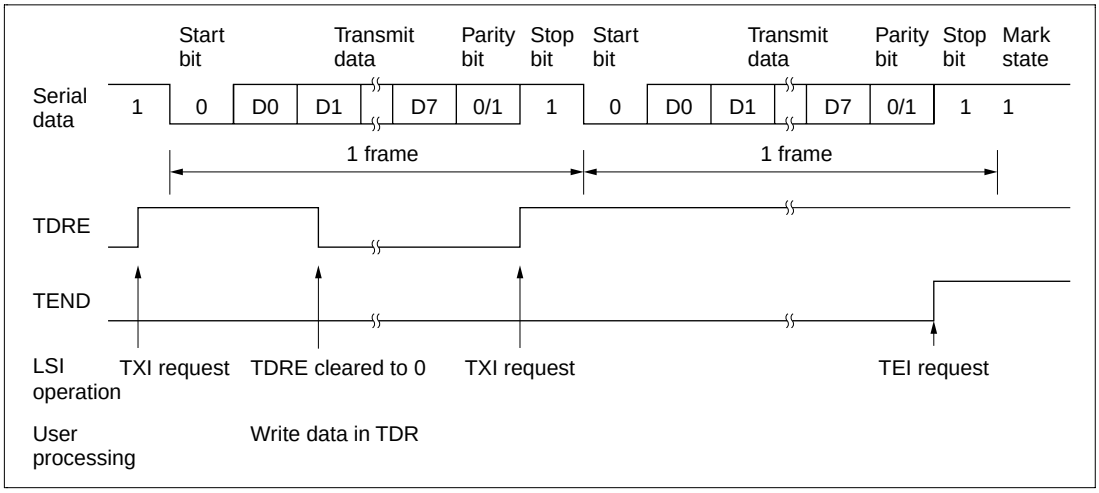


Figure 10.11 Typical Transmit Operation in Asynchronous Mode (8-Bit Data, Parity Bit Added, and 1 Stop Bit)

Receiving: Figure 10.12 shows a typical flow chart for receiving serial data. After SCI3 initialization, follow the procedure below.

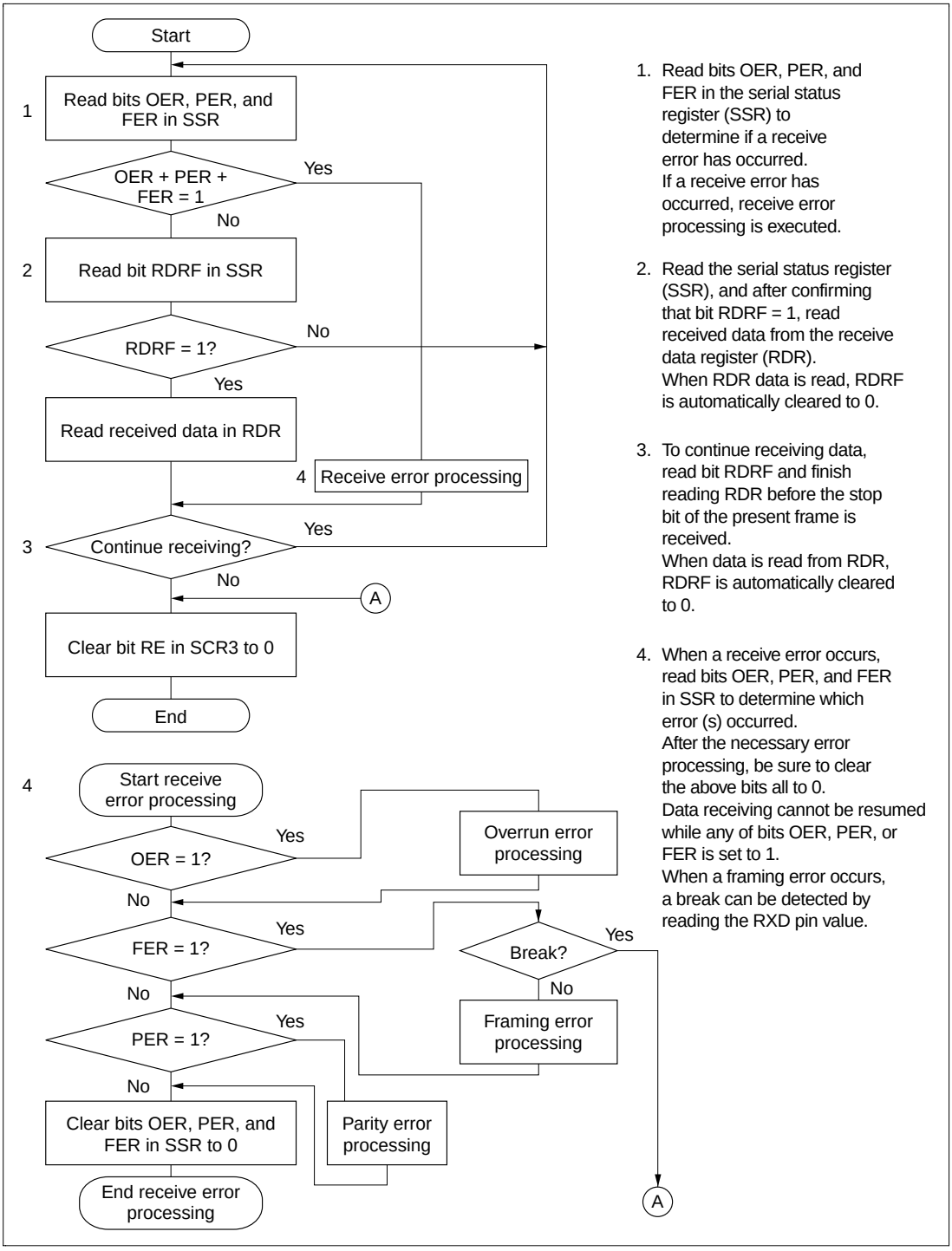


Figure 10.12 Typical Serial Data Receiving Flow Chart in Asynchronous Mode

SCI3 operates as follows when receiving serial data in asynchronous mode.

SCI3 monitors the communication line, and when a start bit (0) is detected it performs internal synchronization and starts receiving. The communication format for data receiving is as outlined in table 10.14. Received data is set in RSR from LSB to MSB, then the parity bit and stop bit(s) are received. After receiving the data, SCI3 performs the following checks:

- Parity check: The number of 1s received is checked to see if it matches the odd or even parity selected in bit PM of SMR.
- Stop bit check: The stop bit is checked for a value of 1. If there are two stop bits, only the first bit is checked.
- Status check: The RDRF bit is checked for a value of 0 to make sure received data can be transferred from RSR to RDR.

If no receive error is detected by the above checks, bit RDRF is set to 1 and the received data is stored in RDR. At that time, if bit RIE in SCR3 is set to 1, an RXI interrupt is requested. If the error check detects a receive error, the appropriate error flag (OER, PER, or FER) is set to 1. RDRF retains the same value as before the data was received. If at this time bit RIE in SCR3 is set to 1, an ERI interrupt is requested.

Table 10.15 gives the receive error detection conditions and the processing of received data in each case.

Note: Data receiving cannot be continued while a receive error flag is set. Before continuing the receive operation it is necessary to clear the OER, FER, PER, and RDRF flags to 0.

Table 10.15 Receive Error Conditions and Received Data Processing

Receive Error	Abbrev.	Detection Conditions	Received Data Processing
Overrun error	OER	Receiving of the next data ends while bit RDRF in SSR is still set to 1	Received data is not transferred from RSR to RDR
Framing error	FER	Stop bit is 0	Received data is transferred from RSR to RDR
Parity error	PER	Received data does not match the parity (odd/even) set in SMR	Received data is transferred from RSR to RDR

Figure 10.13 shows a typical SCI3 data receive operation in asynchronous mode.

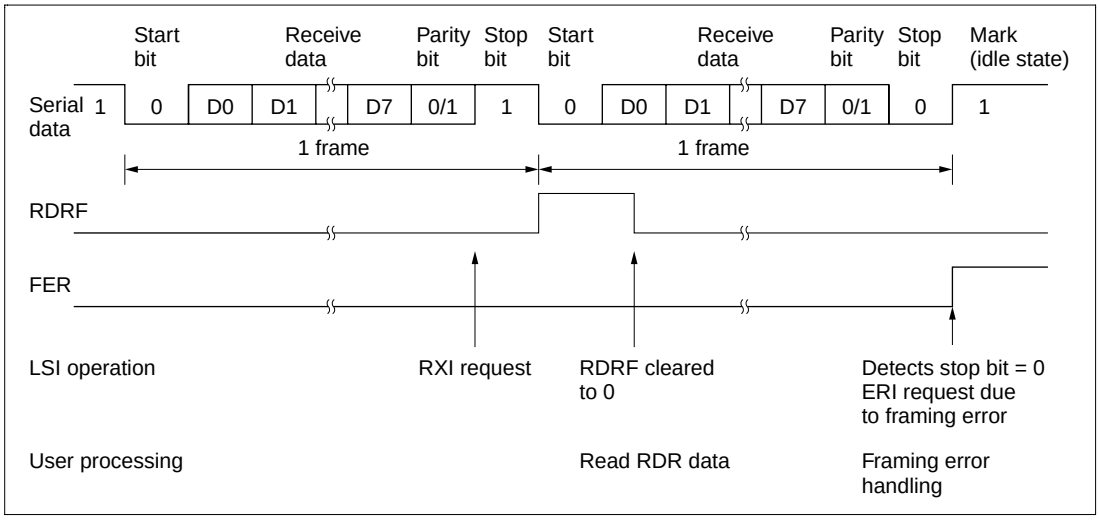


Figure 10.13 Typical Receive Operation in Asynchronous Mode (8-Bit Data, Parity Bit Added, and 1 Stop Bit)

10.3.5 Operation in Synchronous Mode

In synchronous mode, data is sent or received in synchronization with clock pulses. This mode is suited to high-speed serial communication.

SCI3 consists of independent transmit and receive modules, so full duplex communication is possible, sharing the same clock between both modules. Both the transmit and receive modules have a double-buffer configuration. This allows data to be written during a transmit operation so that data can be transmitted continuously, and enables data to be read during a receive operation so that data can be received continuously.

Transmit/Receive Format

Figure 10.14 shows the general communication data format for synchronous communication.

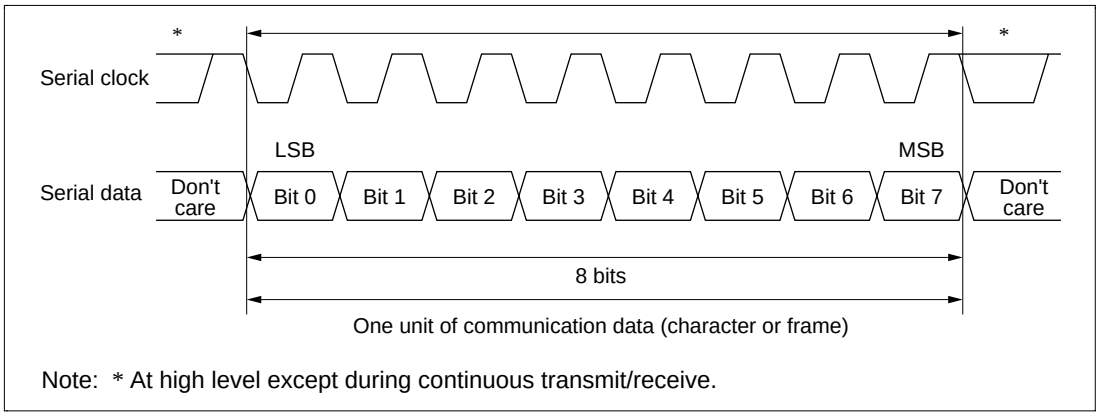


Figure 10.14 Data Format in Synchronous Communication Mode

In synchronous communication, data on the communication line is output from one falling edge of the serial clock until the next falling edge. Data is guaranteed valid at the rising edge of the serial clock.

One character of data starts from the LSB and ends with the MSB. The communication line retains the MSB state after the MSB is output.

In synchronous receive mode, SCI3 latches receive data in synchronization with the rising edge of the serial clock.

The transmit/receive format is fixed at 8-bit data. No parity bit or multiprocessor bit is added in this mode.

Clock

Either an internal clock from the built-in baud rate generator is used, or an external clock is input at pin SCK_3. The choice of clock sources is designated by bit COM in SMR and bits CKE1 and CKE0 in serial control register 3 (SCR3). See table 10.12 for details on selecting the clock source.

When operation is based on an internal clock, a serial clock is output at pin SCK_3. Eight clock pulses are output per character of transmit/receive data. When no transmit or receive operation is being performed, the pin is held at the high level.

Data Transmit/Receive Operations

SCI3 Initialization: Before transmitting or receiving data, follow the SCI3 initialization procedure explained under 10.3.4, SCI3 Initialization, and illustrated in figure 10.9.

Transmitting: Figure 10.15 shows a typical flow chart for data transmission. After SCI3 initialization, follow the procedure below.

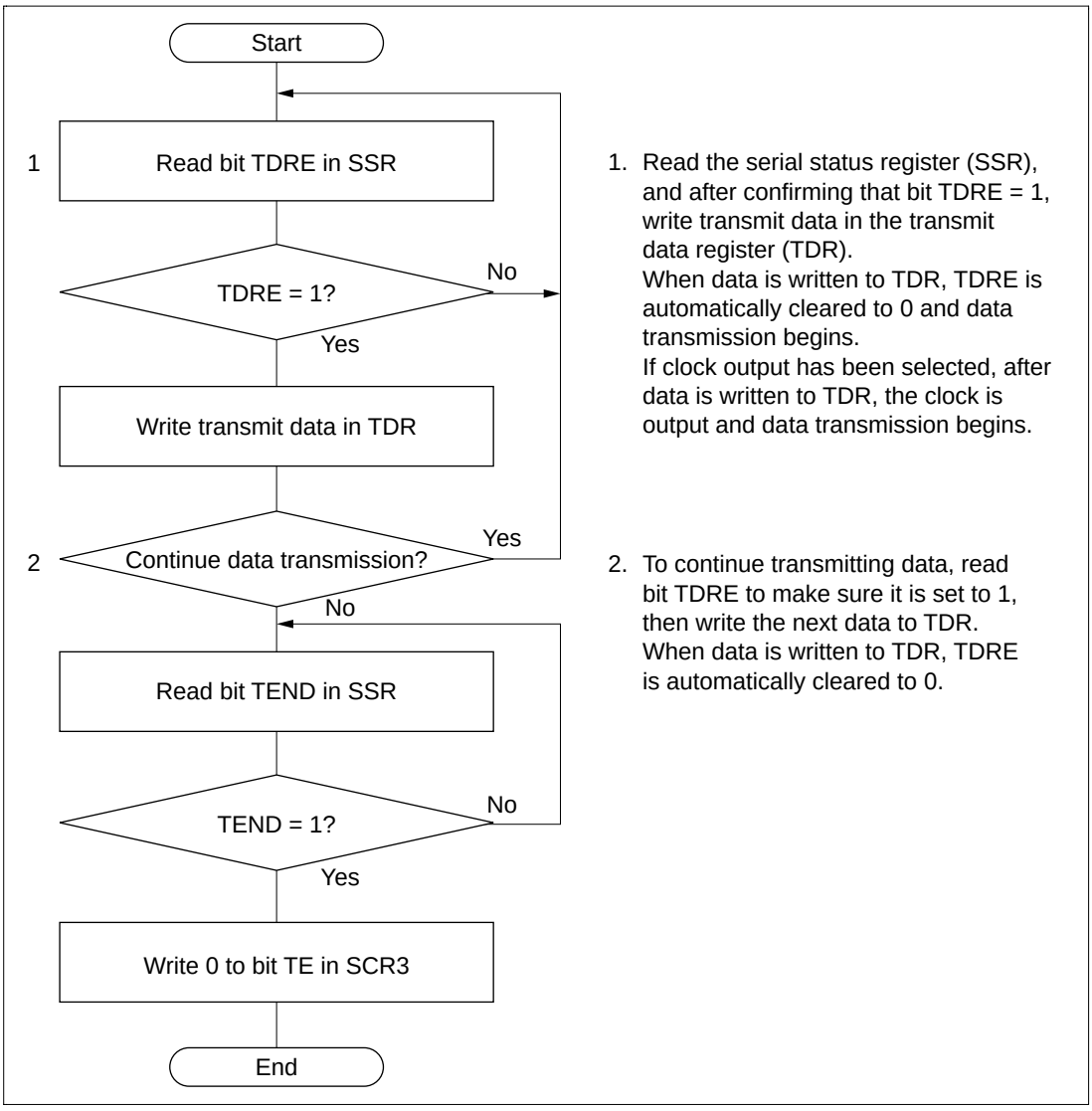


Figure 10.15 Typical Data Transmission Flow Chart in Synchronous Mode

SCI3 operates as follows during data transmission in synchronous mode.

SCI3 monitors bit TDRE in SSR. When this bit is cleared to 0, SCI3 recognizes that there is data written in the transmit data register (TDR), which it transfers to the transmit shift register (TSR). Then TDRE is set to 1 and transmission starts. If bit TIE in SCR3 is set to 1, a TXI interrupt is requested.

If clock output is selected, SCI3 outputs eight serial clock pulses. If an external clock is used, data is output in synchronization with the clock input.

Serial data is transmitted from pin TXD in order from LSB (bit 0) to MSB (bit 7).

Then TDRE is checked as the MSB (bit 7) is being transmitted. If TDRE is 0, data is transferred from TDR to TSR, and after the MSB (bit 7) is sent, transmission of the next frame starts. If TDRE is 1, the TEND bit in SSR is set to 1, and after the MSB (bit 7) has been sent, the MSB state is maintained. A TEI interrupt is requested in this state if bit TEIE in SCR3 is set to 1.

After data transmission ends, pin SCK_3 is held at the high level.

Note: Data transmission cannot take place while any of the receive error flags (OER, FER, PER) is set to 1. Be sure to confirm that these error flags are cleared to 0 before starting transmission.

Figure 10.16 shows a typical SCI3 transmit operation in synchronous mode.

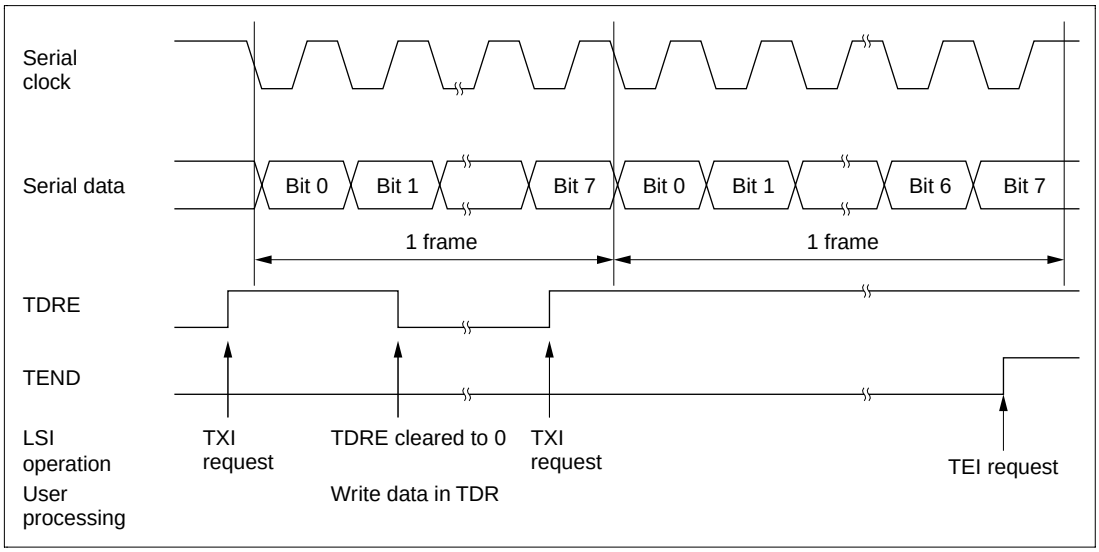


Figure 10.16 Typical SCI3 Transmit Operation in Synchronous Mode

Receiving: Figure 10.17 shows a typical flow chart for receiving data. After SCI3 initialization, follow the procedure below.

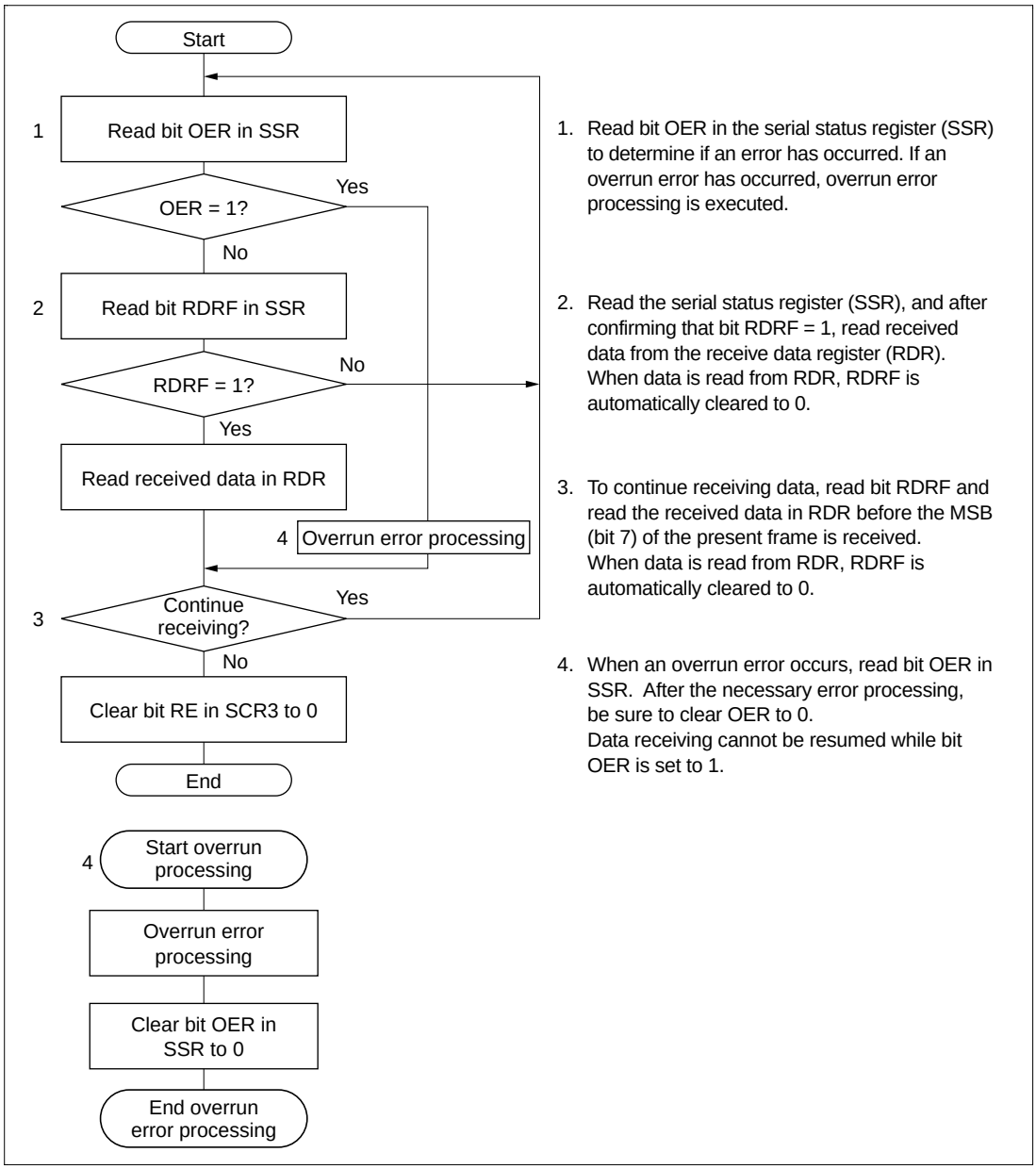


Figure 10.17 Typical Data Receiving Flow Chart in Synchronous Mode

SCI3 operates as follows when receiving serial data in synchronous mode.

SCI3 synchronizes internally with the input or output of the serial clock and starts receiving. Received data is set in RSR from LSB to MSB.

After data has been received, SCI3 checks to confirm that the value of bit RDRF is 0 indicating that received data can be transferred from RSR to RDR. If this check passes, RDRF is set to 1 and the received data is stored in RDR. At this time, if bit RIE in SCR3 is set to 1, an RXI interrupt is requested. If an overrun error is detected, OER is set to 1 and RDRF remains set to 1. Then if bit RIE in SCR3 is set to 1, an ERI interrupt is requested.

For the overrun error detection conditions and receive data processing, see table 10.15.

Note: Data receiving cannot be continued while a receive error flag is set. Before continuing the receive operation it is necessary to clear the OER, FER, PER, and RDRF flags to 0.

Figure 10.18 shows a typical receive operation in synchronous mode.

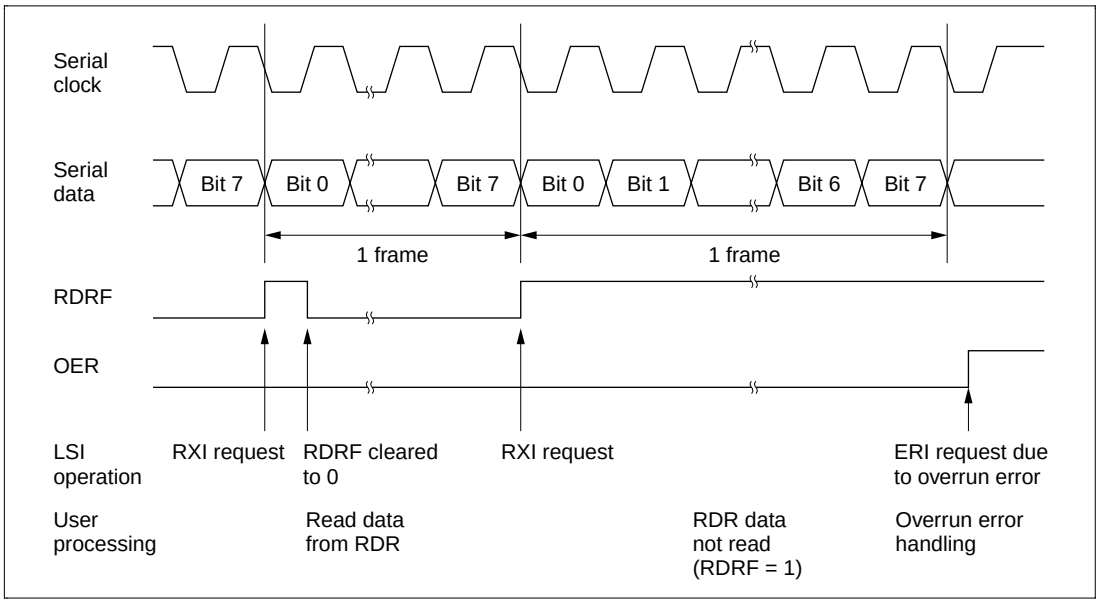
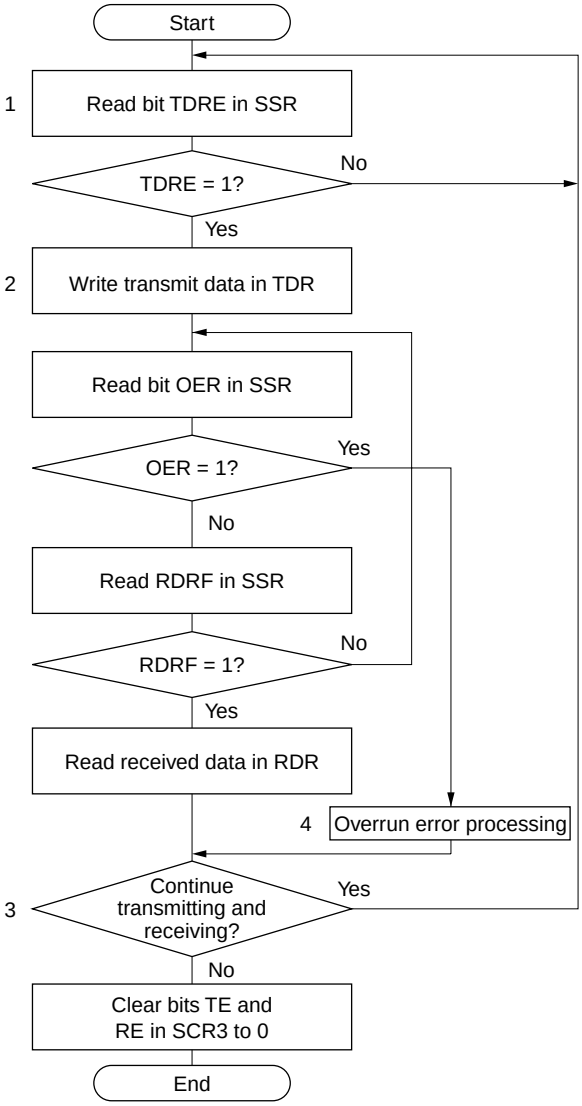


Figure 10.18 Typical Receive Operation in Synchronous Mode

Simultaneous Transmit/Receive: Figure 10.19 shows a typical flow chart for transmitting and receiving simultaneously. After SCI3 synchronization, follow the procedure below.



1. Read the serial status register (SSR), and after confirming that bit TDRE = 1, write transmit data in the transmit data register (TDR). When data is written to TDR, TDRE is automatically cleared to 0.
2. Read the serial status register (SSR), and after confirming that bit RDRF = 1, read the received data from the receive data register (RDR). When data is read from RDR, RDRF is automatically cleared to 0.
3. To continue transmitting and receiving serial data, read bit RDRF and finish reading RDR before the MSB (bit 7) of the present frame is received. Also read bit TDRE and check that it is set to 1, indicating that data can be written, then write the next data in TDR, before the MSB (bit 7) of the current frame is transmitted. When data is written to TDR, TDRE is automatically cleared to 0; and when data is read from RDR, RDRF is automatically cleared to 0.
4. When an overrun error occurs, read bit OER in SSR. After the necessary error processing, be sure to clear OER to 0. Data transmission and reception cannot take place while bit OER is set to 1. See figure 10.17 for overrun error processing.

Figure 10.19 Simultaneous Transmit/Receive Flow Chart in Synchronous Mode

Notes: 1. To switch from transmitting to simultaneous transmitting and receiving, use the following procedure.
   - First confirm that TDRE and TEND are both set to 1 and that SCI3 has finished transmitting. Next clear TE to 0. Then set both TE and RE to 1.

2. To switch from receiving to simultaneous transmitting and rceiving, use the following procedure.
   - After confirming that SCI3 has finished receiving, clear RE to 0. Next, after confirming that RDRF and the error flags (OER FER, PER) are all 0, set both TE and RE to 1.

10.3.6 Multiprocessor Communication Function

The multiprocessor communication function enables several processors to share a single serial communication line. The processors communicate in asynchronous mode using a format with an additional multiprocessor bit (multiprocessor format).

In multiprocessor communication, each receiving processor is addressed by an ID code. A serial communication cycle consists of two cycles: an ID-sending cycle that identifies the receiving processor, and a data-sending cycle. The ID-sending cycle and data-sending cycle are differentiated by the multiprocessor bit. The multiprocessor bit is 1 in an ID-sending cycle, and 0 in a data-sending cycle.

The transmitting processor starts by sending the ID of the receiving processor with which it wants to communicate as data with the multiprocessor bit set to 1. Next the transmitting processor sends transmit data with the multiprocessor bit cleared to 0. When a receiving processor receives data with the multiprocessor bit set to 1, it compares the data with its own ID. If the data matches its ID, the receiving processor continues to receive incoming data. If the data does not match its ID, the receiving processor skips further incoming data until it again receives data with the multiprocessor bit set to 1. Multiple processors can send and receive data in this way.

Figure 10.20 shows an example of communication among different processors using a multiprocessor format.

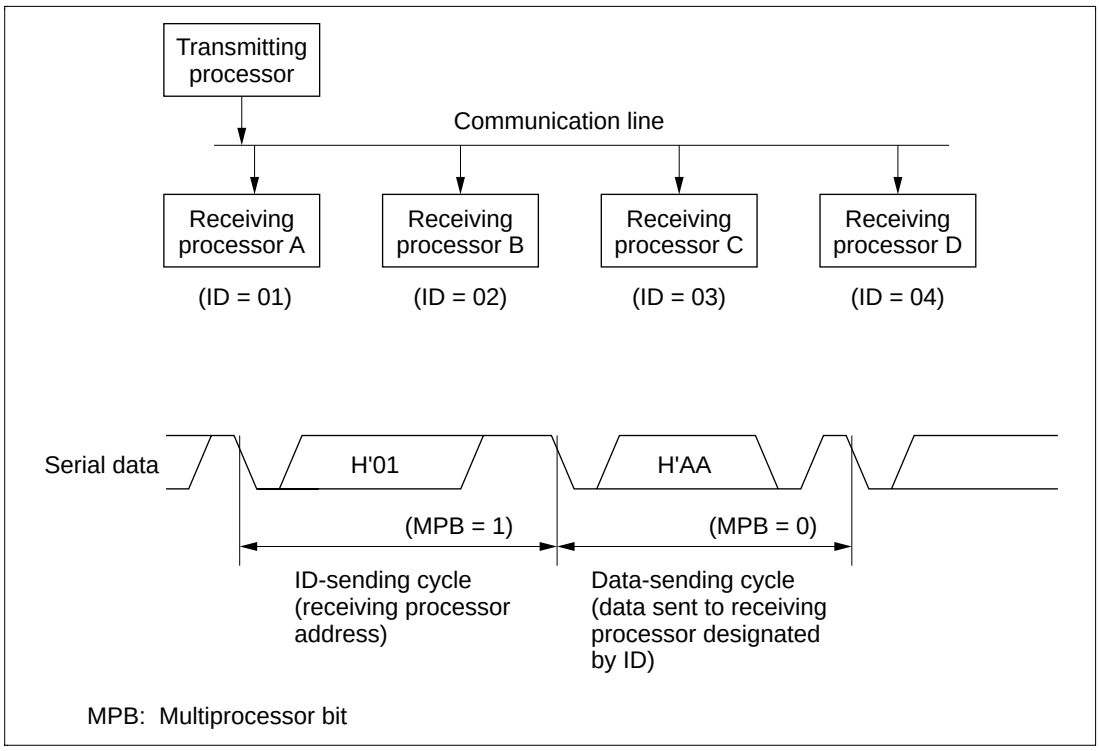


Figure 10.20 Example of Interprocessor Communication Using Multiprocessor Format (Data H'AA Sent to Receiving Processor A)

Four communication formats are available. Parity-bit settings are ignored when a multiprocessor format is selected. For details see table 10.14.

For a description of the clock used in multiprocessor communication, see 10.3.4, Operation in Asynchronous Mode.

Transmitting Multiprocessor Data: Figure 10.21 shows a typical flow chart for multiprocessor serial data transmission. After SCI3 initialization, follow the procedure below.

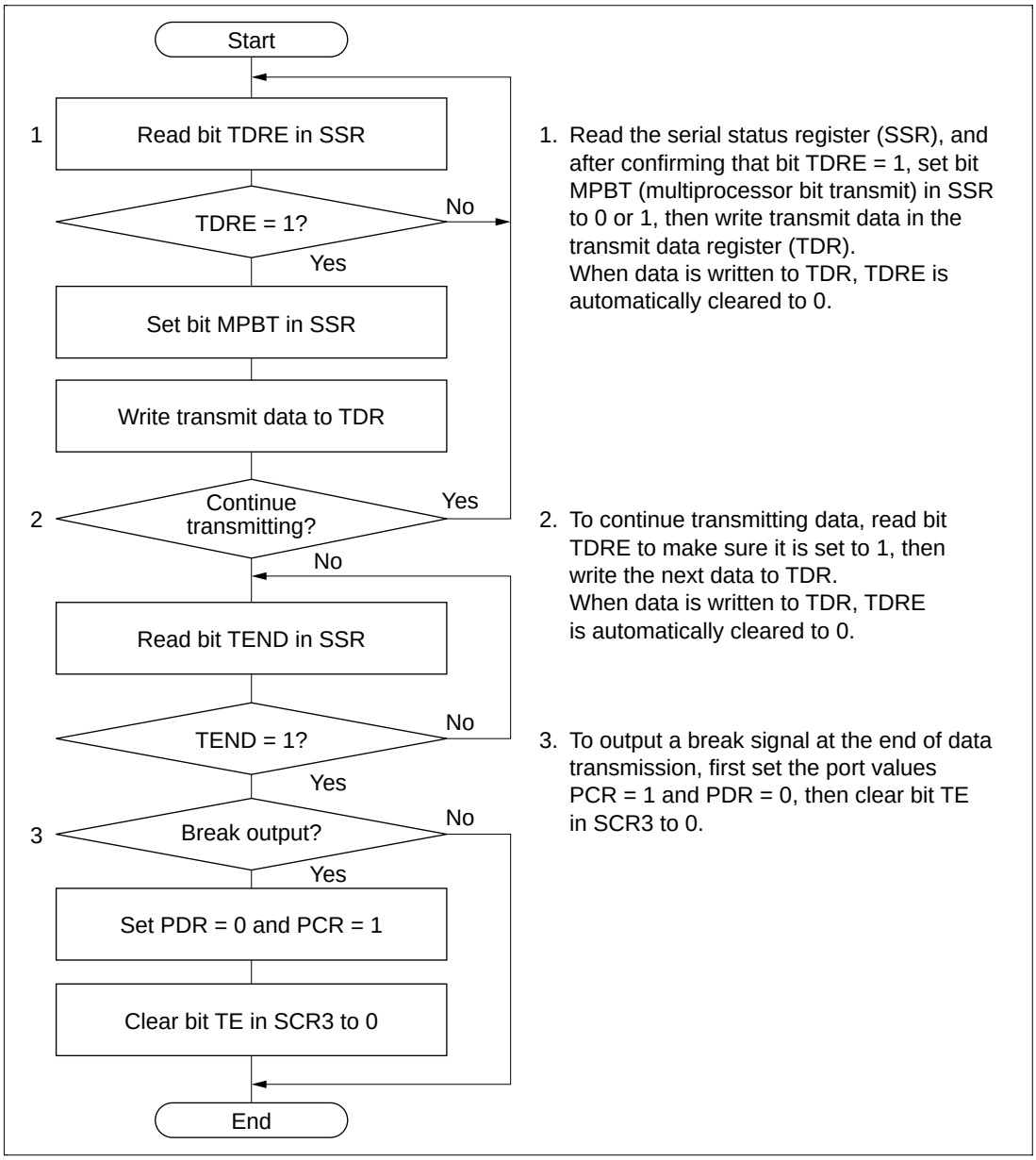


Figure 10.21 Typical Multiprocessor Data Transmission Flow Chart

SCI3 operates as follows during data transmission using a multiprocessor format.

SCI3 monitors bit TDRE in SSR. When this bit is cleared to 0, SCI3 recognizes that there is data written in the transmit data register (TDR), which it transfers to the transmit shift register (TSR). Then TDRE is set to 1 and transmission starts. If bit TIE in SCR3 is set to 1, a TXI interrupt is requested.

Serial data is transmitted from pin TXD using the communication format outlined in table 10.14.

Next, TDRE is checked as the stop bit is being transmitted. If TDRE is 0, data is transferred from TDR to TSR, and after the stop bit is sent, transmission of the next frame starts. If TDRE is 1, the TEND bit in SSR is set to 1, and after the stop bit is sent the output remains at 1 (mark state). A TEI interrupt is requested in this state if bit TEIE in SCR3 is set to 1.

Figure 10.22 shows a typical SCI3 operation in multiprocessor communication mode.

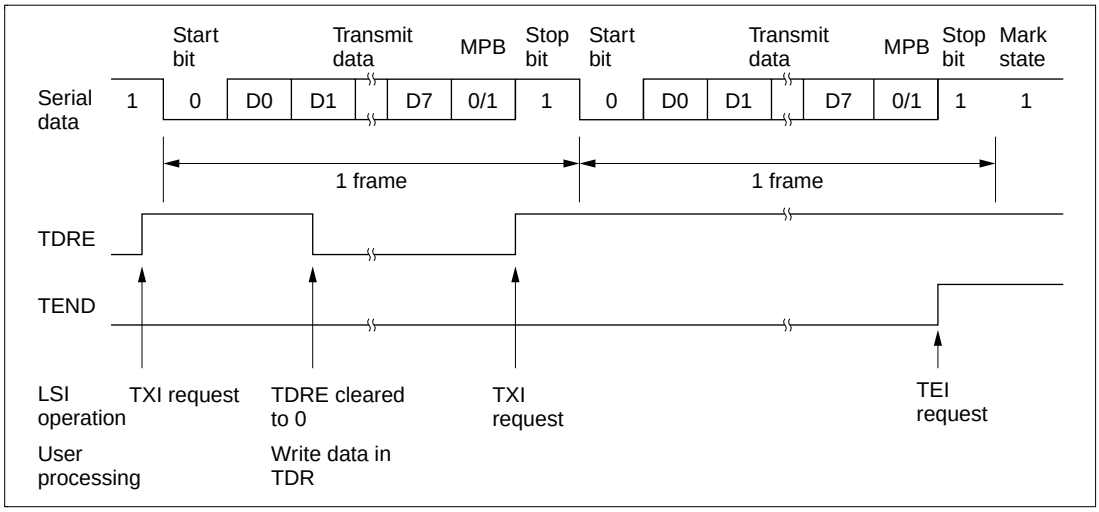


Figure 10.22 Typical Multiprocessor Format Transmit Operation (8-Bit Data, Multiprocessor Bit Added, and 1 Stop Bit)

Receiving Multiprocessor Data: Figure 10.23 shows a typical flow chart for receiving data using a multiprocessor format. After SCI3 initialization, follow the procedure below.

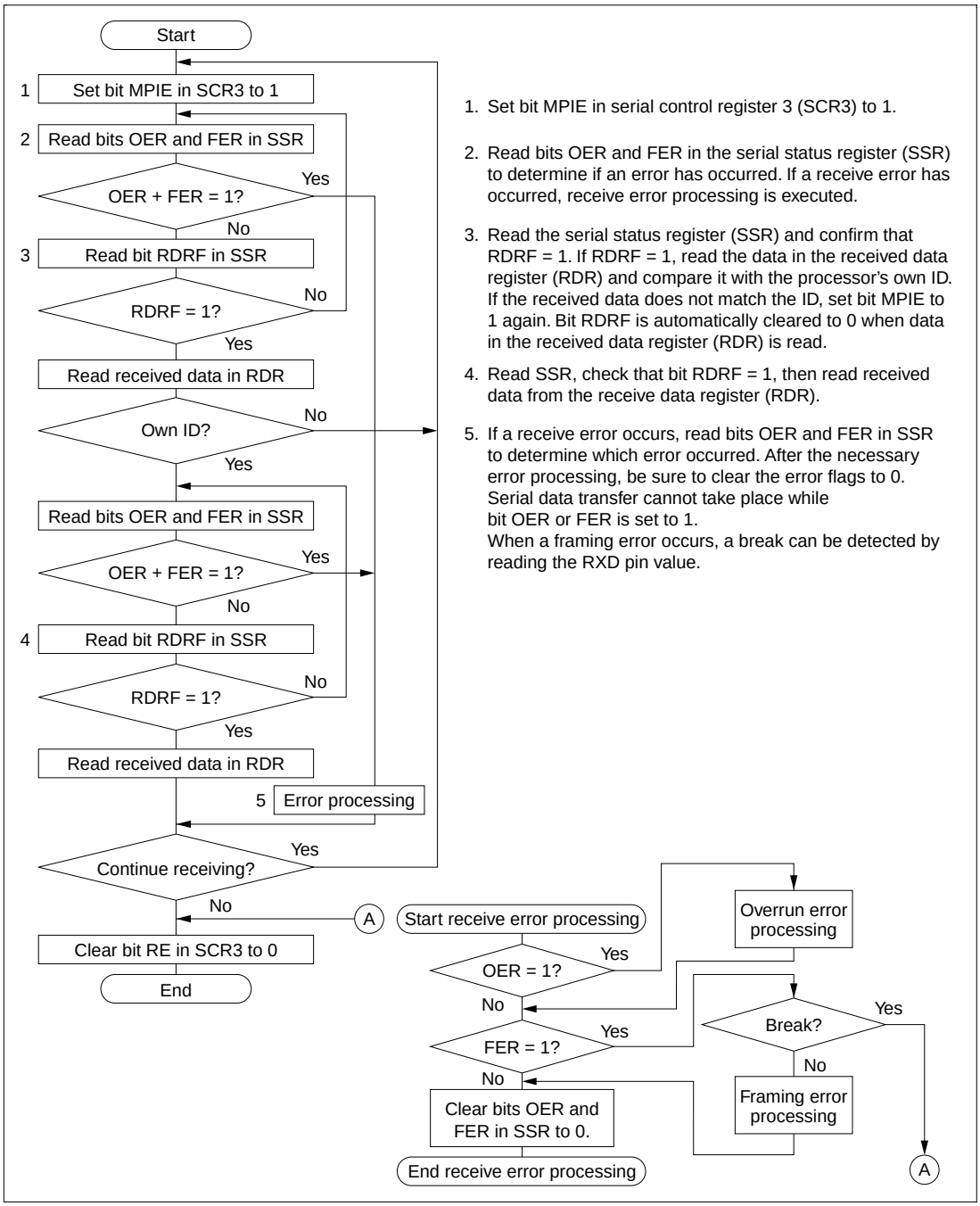


Figure 10.23 Typical Flow Chart for Receiving Serial Data Using Multiprocessor Format

Figure 10.24 gives an example of data reception using a multiprocessor format.

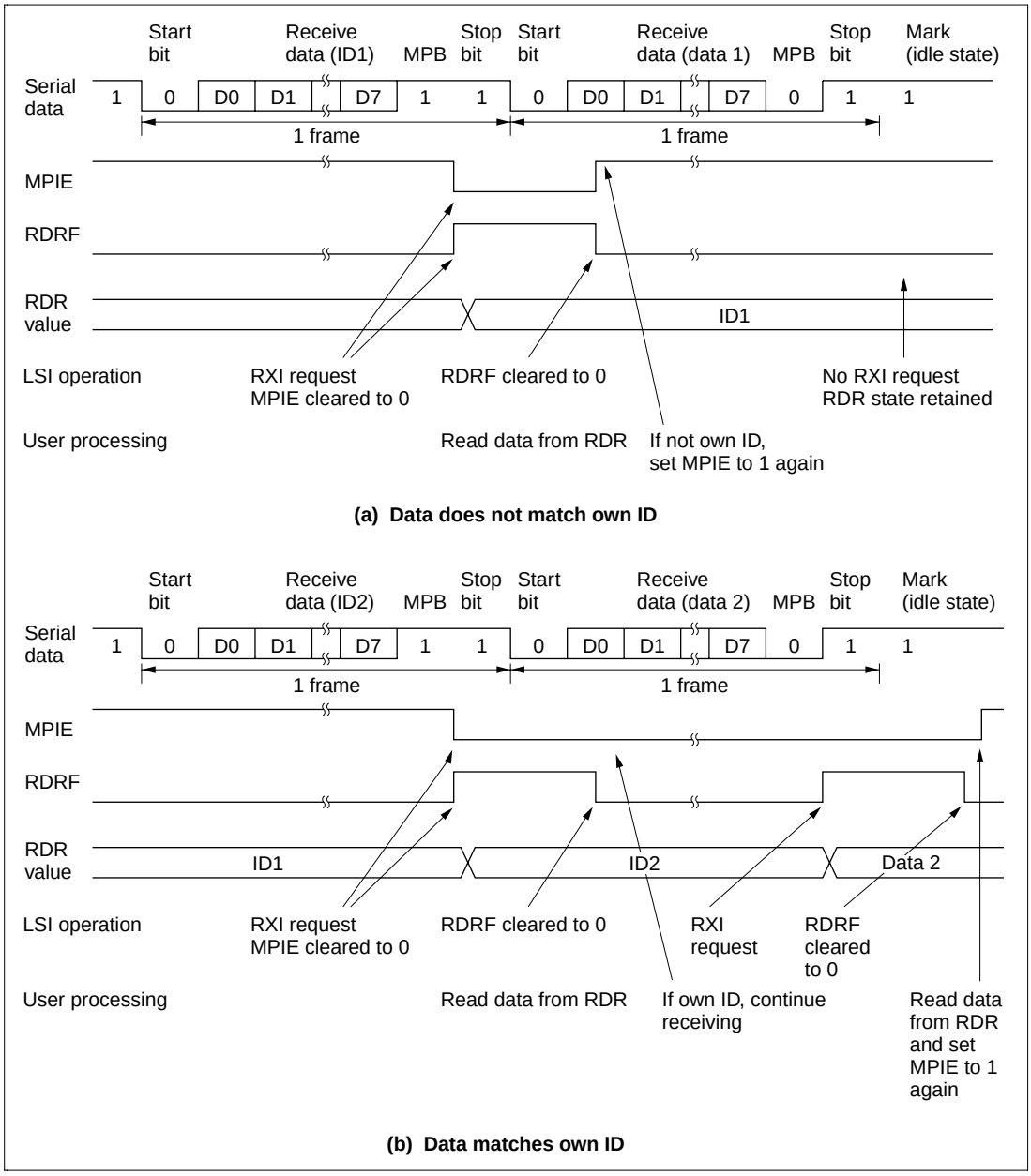


Figure 10.24 Example of Multiprocessor Format Receive Operation (8-Bit Data, Multiprocessor Bit Added, and 1 Stop Bit)

10.3.7 Interrupts

SCI3 has six interrupt sources: transmit end, transmit data empty, receive data full, and the three receive error interrupts (overrun error, framing error, and parity error). All share a common interrupt vector. Table 10.16 describes each interrupt.

Table 10.16 SCI3 Interrupts

Interrupt	Description	Vector Address
RXI	Interrupt request due to receive data register full (RDRF)	H'0024
TXI	Interrupt request due to transmit data register empty (TDRE)	
TEI	Interrupt request due to transmit end (TEND)	
ERI	Interrupt request due to receive error (OER, FER, or PER)	

The interrupt requests are enabled and disabled by bits TIE and RIE of SCR3.

When bit TDRE in SSR is set to 1, TXI is requested. When bit TEND in SSR is set to 1, TEI is requested. These two interrupt requests occur during data transmission.

The initial value of bit TDRE is 1. Accordingly, if the transmit data empty interrupt request (TXI) is enabled by setting bit TIE to 1 in SCR3 before placing transmit data in TDR, TXI will be requested even though no transmit data has been readied.

Likewise, the initial value of bit TEND in SSR is 1. Accordingly, if the transmit end interrupt request (TEI) is enabled by setting bit TEIE to 1 in SCR3 before placing transmit data in TDR, TEI will be requested even though no data has been transmitted.

These interrupt features can be used to advantage by programming the interrupt handler to move the transmit data into TDR. When this technique is not used, the interrupt enable bits (TIE and TEIE) should not be set to 1 until after TDR has been loaded with transmit data, to avoid unwanted TXI and TEI interrupts.

When bit RDRF in SSR is set to 1, RXI is requested. When any of SSR bits OER, FER, or PER is set to 1, ERI is requested. These two interrupt requests occur during the receiving of data.

Details on interrupts are given in 3.3, Interrupts.

10.3.8 Application Notes

When using SCI3, attention should be paid to the following matters.

Relation between Bit TDRE and Writing Data to TDR: Bit TDRE in the serial status register (SSR) is a status flag indicating that TDR does not contain new transmit data. TDRE is automatically cleared to 0 when data is written to TDR. When SCI3 transfers data from TDR to TSR, bit TDRE is set to 1.

Data can be written to TDR regardless of the status of bit TDRE. However, if new data is written to TDR while TDRE is cleared to 0, assuming the data held in TDR has not yet been shifted to TSR, it will be lost. For this reason it is advisable to confirm that bit TDRE is set to 1 before each write to TDR and not write to TDR more than once without checking TDRE in between.

Operation when Multiple Receive Errors Occur at the Same Time: When two or more receive errors occur at the same time, the status flags in SSR are set as shown in table 10.17. If an overrun error occurs, data is not transferred from RSR to RDR, and receive data is lost.

Table 10.17 SSR Status Flag States and Transfer of Receive Data

SSR Status Flags				Receive Data Transfer	
RDRF*	OER	FER	PER	(RSR → RDR)	Receive Error Status
1	1	0	0	Not transferred	Overrun error
0	0	1	0	Transferred	Framing error
0	0	0	1	Transferred	Parity error
1	1	1	0	Not transferred	Overrun error + framing error
1	1	0	1	Not transferred	Overrun error + parity error
0	0	1	1	Transferred	Framing error + parity error
1	1	1	1	Not transferred	Overrun error + framing error + parity error

Note: * RDRF keeps the same state as before the data was received. However, if due to a late read of received data in one frame an overrun error occurs in the next frame, RDRF is cleared to 0 when RDR is read.

Break Detection and Processing: Break signals can be detected by reading the RXD pin directly when a framing error (FER) is detected. In the break state the input from the RXD pin consists of all 0s, so FER is set and the parity error flag (PER) may also be set. In the break state SCI3 continues to receive, so if the FER bit is cleared to 0 it will be set to 1 again.

Sending a Mark or Break Signal: When TE is cleared to 0 the TXD pin becomes an I/O port, the level and direction (input or output) of which are determined by the PDR and PCR bits. This feature can be used to place the TXD pin in the mark state or send a break signal.

To place the serial communication line in the mark (1) state before TE is set to 1, set the PDR and PCR bits both to 1. Since TE is cleared to 0, TXD becomes a general output port outputting the value 1.

To send a break signal during data transmission, set the PCR bit to 1 and clear the PDR bit to 0, then clear TE to 0. When TE is cleared to 0 the transmitter is initialized, regardless of its current state, so the TXD pin becomes an output port outputting the value 0.

Receive Error Flags and Transmit Operation (Sysnchronous Mode Only): When a receive error flag (ORER, PER, or FER) is set to 1, SCI3 will not start transmitting even if TDRE is cleared to 0. Be sure to clear the receive error flags to 0 when starting to transmit. Note that clearing RE to 0 does not clear the receive error flags.

Receive Data Sampling Timing and Receive Margin in Asynchronous Mode: In asynchronous mode SCI3 operates on a base clock with 16 times the bit rate frequency. In receiving, SCI3 synchronizes internally with the falling edge of the start bit, which it samples on the base clock. Receive data is latched at the rising edge of the eighth base clock pulse. See figure 10.25.

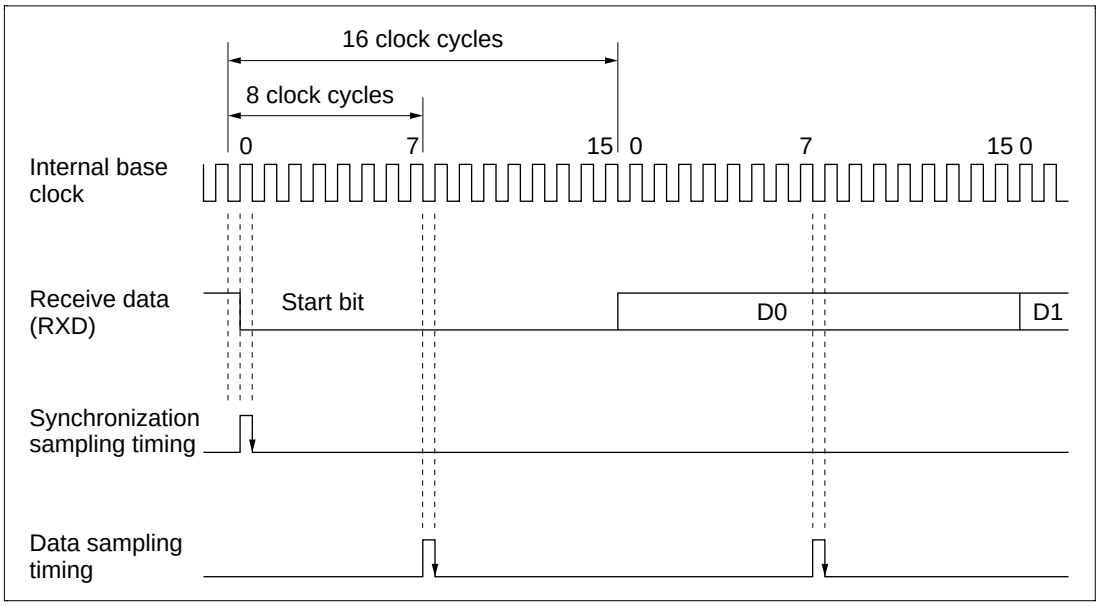


Figure 10.25 Receive Data Sampling Timing in Asynchronous Mode

The receive margin in asynchronous mode can therefore be derived from the following equation.

$$M = \{(0.5 - 1/2N) - (D - 0.5) / N - (L - 0.5) F\} \times 100\% \quad \text{..... Equation (1)}$$

M: Receive margin (%)
N: Ratio of clock frequency to bit rate (N = 16)
D: Clock duty cycle (D = 0.5 to 1)
L: Frame length (L = 9 to 12)
F: Absolute value of clock frequency error

In equation (1), if F (absolute value of clock frequency error) = 0 and D (clock duty cycle) = 0.5, the receive margin is 46.875% as given by equation (2) below.

When D = 0.5 and F = 0,

$$M = \{0.5 - 1/(2 \times 16)\} \times 100\% = 46.875\% \quad \text{..... Equation (2)}$$

This value is theoretical. In actual system designs a margin of from 20 to 30 percent should be allowed.

Relationship between Bit RDRF and Reading RDR: While SCI3 is receiving, it checks the RDRF flag. When a frame of data has been received, if the RDRF flag is cleared to 0, data receiving ends normally. If RDRF is set to 1, an overrun error occurs.

RDRF is automatically cleared to 0 when the contents of RDR are read. If RDR is read more than once, the second and later reads will be performed with RDRF cleared to 0. While RDRF is 0, if RDR is read when reception of the next frame is just ending, data from the next frame may be read. This is illustrated in figure 10.26.

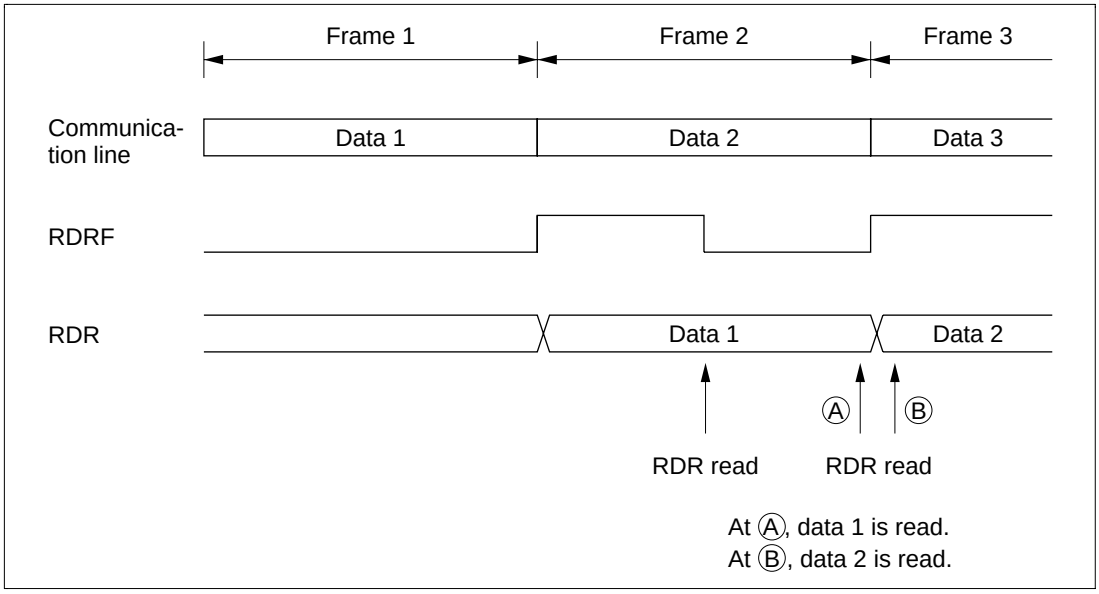


Figure 10.26 Relationship between Data and RDR Read Timing

To avoid the situation described above, after RDRF is confirmed to be 1, RDR should only be read once and should not be read twice or more.

When the same data must be read more than once, the data read the first time should be copied to RAM, for example, and the copied data should be used. An alternative is to read RDR but leave a safe margin of time before reception of the next frame is completed. Specifically, reading of RDR should be completed before bit 7 is transferred in synchronous mode, or before the stop bit is transferred in asynchronous mode.

Caution on Switching of SCK_3 Function: If pin SCK_3 is used as a clock output pin by SCI3 in synchronous mode and is then switched to a general input/output pin (a pin with a different function), the pin outputs a low level signal for half a system clock (ϕ) cycle immediately after it is switched.

This can be prevented by either of the following methods according to the situation.

1. When an SCK_3 function is switched from clock output to non clock-output
   When stopping data transfer, issue one instruction to clear bits TE and RE in SCR3 to 0 and to set bits CKE1 and CKE0 to 1 and 0, respectively. In this case, bit COM in SMR should be left 1. The above prevents SCK_3 from being used as a general input/output pin. To avoid an intermediate level of voltage from being applied to SCK_3, the line connected to SCK_3 should be pulled up to the V_{CC} level via a resistor, or supplied with output from an external device.

2. When an SCK_3 function is switched from clock output to general input/output

   When stopping data transfer,

   a. Issue one instruction to clear bits TE and RE in SCR3 to 0 and to set bits CKE1 and CKE0 to 1 and 0, respectively.
   b. Clear bit COM in SCR3 to 0
   c. Clear bits CKE1 and CKE0 in SCR3 to 0

Note that special care is also needed here to avoid an intermediate level of voltage from being applied to SCK_3.

Caution on Switching TxD Function: If pin TXD is used as a data output pin by SCI3 in synchronous mode and is then switched to a general input/output pin (a pin with a different function), the pin outputs a high level signal for one system clock (ϕ) cycle immediately after it is switched.

Section 11 14-Bit PWM (H8/3857 Series Only)

11.1 Overview

The H8/3857 Series is provided with a 14-bit PWM (pulse width modulator), which can be used as a D/A converter by connecting a low-pass filter. The H8/3854 Series does not have this module.

11.1.1 Features

Features of the 14-bit PWM are as follows.

- Choice of two conversion periods
  A conversion period of 32,768/ϕ, with a minimum modulation width of 2/ϕ (PWCR0 = 1), or a conversion period of 16,384/ϕ, with a minimum modulation width of 1/ϕ (PWCR0 = 0), can be chosen.
- Pulse division method for less ripple

11.1.2 Block Diagram

Figure 11.1 shows a block diagram of the 14-bit PWM.

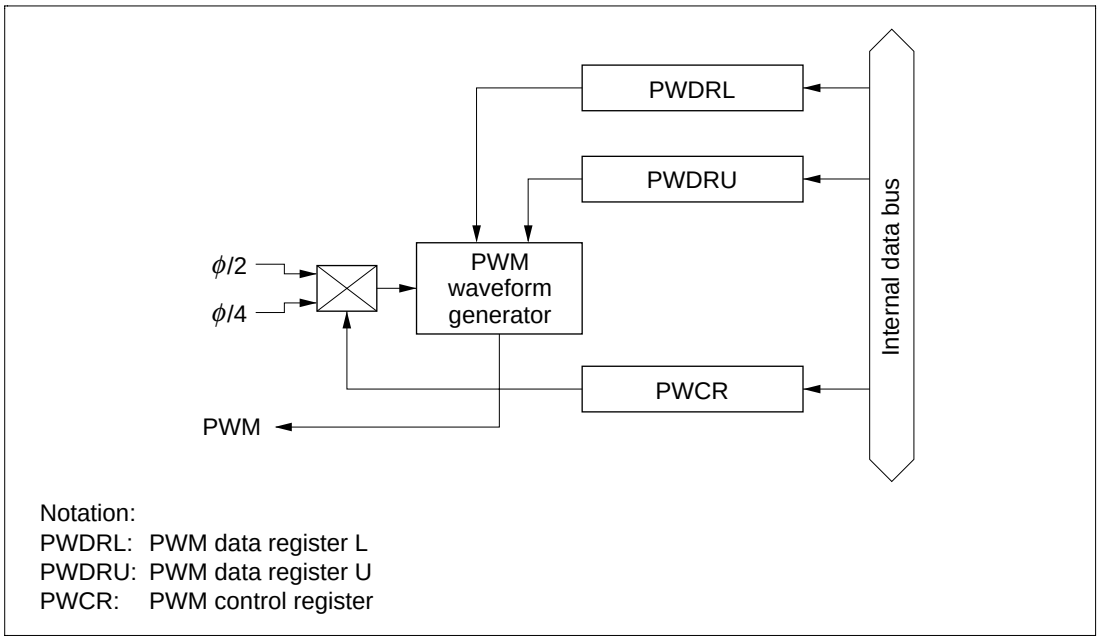


Figure 11.1 Block Diagram of the 14 bit PWM

11.1.3 Pin Configuration

Table 11.1 shows the output pin assigned to the 14-bit PWM.

Table 11.1 Pin Configuration

Name	Abbrev.	I/O	Function
PWM output pin	PWM	Output	Pulse-division PWM waveform output

11.1.4 Register Configuration

Table 11.2 shows the register configuration of the 14-bit PWM.

Table 11.2 Register Configuration

Name	Abbrev.	R/W	Initial Value	Address
PWM control register	PWCR	W	H'FE	H'FFD0
PWM data register U	PWDRU	W	H'C0	H'FFD1
PWM data register L	PWDRL	W	H'00	H'FFD2

11.2 Register Descriptions

11.2.1 PWM Control Register (PWCR)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PWCR0
Initial value	1	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'FE.

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

Bit 0—Clock Select 0 (PWCR0): Bit 0 selects the clock supplied to the 14-bit PWM. This bit is a write-only bit; it is always read as 1.

Bit 0: PWCR0	Description
0	The input clock is ϕ/2 (t_{ϕ}* = 2/ϕ). The conversion period is 16,384/ϕ, with a minimum modulation width of 1/ϕ. (initial value)
1	The input clock is ϕ/4 (t_{ϕ}* = 4/ϕ). The conversion period is 32,768/ϕ, with a minimum modulation width of 2/ϕ.

Note: t_{ϕ}: Period of PWM input clock

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

Bit	7	6	5	4	3	2	1	0
PWDRU	—	—	PWDRU5	PWDRU4	PWDRU3	PWDRU2	PWDRU1	PWDRU0
Initial value	1	1	0	0	0	0	0	0
Read/Write	—	—	W	W	W	W	W	W

Bit	7	6	5	4	3	2	1	0
PWDRL	PWDRL7	PWDRL6	PWDRL5	PWDRL4	PWDRL3	PWDRL2	PWDRL1	PWDRL0
Initial value	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W

PWDRU and PWDRL form a 14-bit write-only register, with the upper 6 bits assigned to PWDRU and the lower 8 bits to PWDRL. The value written to PWDRU and PWDRL gives the total high-level width of one PWM waveform cycle.

When 14-bit data is written to PWDRU and PWDRL, the register contents are latched in the PWM waveform generator, updating the PWM waveform generation data. The 14-bit data should always be written in the following sequence, first to PWDRL and then to PWDRU.

1. Write the lower 8 bits to PWDRL.
2. Write the upper 6 bits to PWDRU.

PWDRU and PWDRL are write-only registers. If they are read, all bits are read as 1.

Upon reset, PWDRU and PWDRL are initialized to H'C000.

11.3 Operation

When using the 14-bit PWM, set the registers in the following sequence.

1. Set bit PWM in port mode register 1 (PMR1) to 1 so that pin $P1_4$/PWM is designated for PWM output.

2. Set bit PWCR0 in the PWM control register (PWCR) to select a conversion period of either 32,768/ϕ (PWCR0 = 1) or 16,384/ϕ (PWCR0 = 0).

3. Set the output waveform data in PWM data registers U and L (PWDRU/L). Be sure to write in the correct sequence, first PWDRL then PWDRU. When data is written to PWDRU, the data in these registers will be latched in the PWM waveform generator, updating the PWM waveform generation in synchronization with internal signals.

   One conversion period consists of 64 pulses, as shown in figure 11.2. The total of the high-level pulse widths during this period (T_H) corresponds to the data in PWDRU and PWDRL. This relation can be represented as follows.

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

   where $t\phi$ is the PWM input clock period, either 2/ϕ (bit PWCR0 = 0) or 4/ϕ (bit PWCR0 = 1).

   Example: Settings in order to obtain a conversion period of 8,192 µs:

   When bit PWCR0 = 0, the conversion period is 16,384/ϕ, so ϕ must be 2 MHz. In this case t_{fn} = 128 µs, with 1/ϕ (resolution) = 0.5 µs.

   When bit PWCR0 = 1, the conversion period is 32,768/ϕ, so ϕ must be 4 MHz. In this case t_{fn} = 128 µs, with 2/ϕ (resolution) = 0.5 µs.

   Accordingly, for a conversion period of 8,192 µs, the system clock frequency (ϕ) must be 2 MHz or 4 MHz.

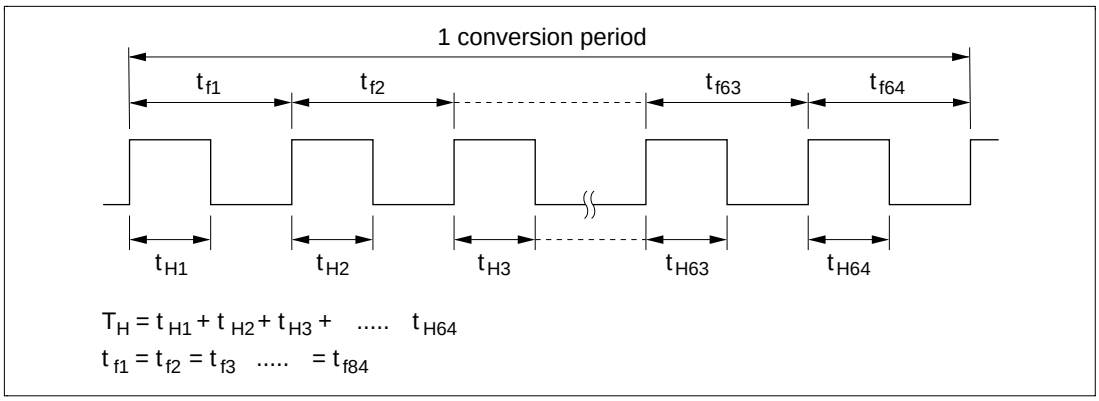


Figure 11.2 PWM Output Waveform

Section 12 A/D Converter

12.1 Overview

The H8/3857 Series and H8/3854 Series include a resistance-ladder-based successive-approximation analog-to-digital converter. The maximum number of analog input channels is eight in the H8/3857 Series and four in the H8/3854 Series.

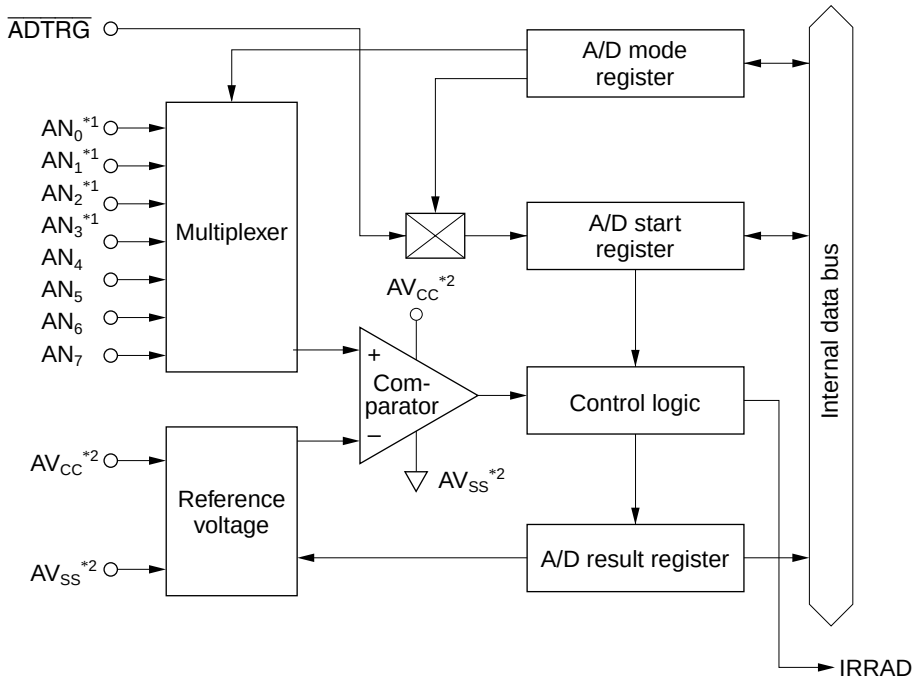
12.1.1 Features

The A/D converter has the following features.

- 8-bit resolution
- Input channels
  - 8 in H8/3857 Series
  - 4 in H8/3854 Series
- Conversion time: approx. 12.4 µs per channel (at 5 MHz operation)
- Built-in sample-and-hold function
- Interrupt requested on completion of A/D conversion
- A/D conversion can be started by external trigger input

12.1.2 Block Diagram

Figure 12.1 shows a block diagram of the A/D converter.



Notes: 1. AN_0 to AN_3 are functions of the H8/3857 Series only, and are not provided in the H8/3854 Series.
2. AV_{CC} and AV_{SS} are functions of the H8/3857 Series only. In the H8/3854 Series, $AV_{CC} = V_{CC}$ and $AV_{SS} = V_{SS}$.

Figure 12.1 Block Diagram of the A/D Converter

12.1.3 Pin Configuration

Table 12.1 shows the A/D converter pin configuration.

Table 12.1 Pin Configuration

Name	Abbrev.	I/O	Function
Analog power supply pin*	AV_{CC}	Input	Power supply and reference voltage of analog part
Analog ground pin*	AV_{SS}	Input	Ground and reference voltage of analog part
Analog input pin 0*	AN_0	Input	Analog input channel 0
Analog input pin 1*	AN_1	Input	Analog input channel 1
Analog input pin 2*	AN_2	Input	Analog input channel 2
Analog input pin 3*	AN_3	Input	Analog input channel 3
Analog input pin 4	AN_4	Input	Analog input channel 4
Analog input pin 5	AN_5	Input	Analog input channel 5
Analog input pin 6	AN_6	Input	Analog input channel 6
Analog input pin 7	AN_7	Input	Analog input channel 7
External trigger input pin	$\overline{\text{ADTRG}}$	Input	External trigger input for starting A/D conversion

Note: * The analog power supply pin, analog ground pin, and analog input pins 0 to 3 are functions of the H8/3857 Series only, and are not provided in the H8/3854 Series.

In the H8/3854 Series, the analog power supply pin is the power supply pin (V_{CC}), and the analog ground pin is the ground pin (V_{SS}).

12.1.4 Register Configuration

Table 12.2 shows the A/D converter register configuration.

Table 12.2 Register Configuration

Name	Abbrev.	R/W	Initial Value	Address
A/D mode register	AMR	R/W	H'30	H'FFC4
A/D start register	ADSR	R/W	H'7F	H'FFC6
A/D result register	ADRR	R	Undefined	H'FFC5

12.2 Register Descriptions

12.2.1 A/D Result Register (ADRR)

Bit	7	6	5	4	3	2	1	0
	ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0
Initial value	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
Read/Write	R	R	R	R	R	R	R	R

The A/D result register (ADRR) is an 8-bit read-only register for holding the results of analog-to-digital conversion.

ADRR can be read by the CPU at any time, but the ADRR values during A/D conversion are not fixed.

After A/D conversion is complete, the conversion result is stored in ADRR as 8-bit data; this data is held in ADRR until the next conversion operation starts.

ADRR is not cleared on reset.

12.2.2 A/D Mode Register (AMR)

Bit	7	6	5	4	3	2	1	0
	CKS	TRGE	—	—	CH3	CH2	CH1	CH0
Initial value	0	0	1	1	0	0	0	0
Read/Write	R/W	R/W	—	—	R/W	R/W	R/W	R/W

AMR is an 8-bit read/write register for specifying the A/D conversion speed, external trigger option, and the analog input pins.

Upon reset, AMR is initialized to H'30.

Bit 7—Clock Select (CKS): Bit 7 sets the A/D conversion speed.

		Conversion Time	
Bit 7: CKS	**Conversion Period**	**ϕ = 2 MHz**	**ϕ = 5 MHz**
0	62/ϕ (initial value)	31 µs	12.4 µs
1	31/ϕ	15.5 µs	—*

Note: * Operation is not guaranteed if the conversion time is less than 12.4 µs. Set bit 7 for a value of at least 12.4 µs.

Bit 6—External Trigger Select (TRGE): Bit 6 enables or disables the start of A/D conversion by external trigger input.

Bit 6: TRGE	**Description**	
0	Disables start of A/D conversion by external trigger	(initial value)
1	Enables start of A/D conversion by rising or falling edge of external trigger at pin $\overline{\text{ADTRG}}$*	

Note: * The external trigger ($\overline{\text{ADTRG}}$) edge is selected by bit INTEG4 of the IRQ edge select register (IEGR). See 3.3.2, Interrupt Edge Select Register (IEGR), for details.

Bits 5 and 4—Reserved Bits: Bits 5 and 4 are reserved; they are always read as 1, and cannot be modified.

Bits 3 to 0—Channel Select (CH3 to CH0): Bits 3 to 0 select the analog input channel.

The channel selection should be made while bit ADSF is cleared to 0.

Bit 3: CH3	**Bit 2: CH2**	**Bit 1: CH1**	**Bit 0: CH0**	**Analog Input Channel**	
0	0	*	*	No channel selected	(initial value)
	1	0	0	AN_0*[1]	
			1	AN_1*[1]	
		1	0	AN_2*[1]	
			1	AN_3*[1]	
1	0	0	0	AN_4	
			1	AN_5	
		1	0	AN_6	
			1	AN_7	
1	1	*	*	Reserved	

Notes: * Don't care

1. Channels AN_0 to AN_3 are functions of the H8/3857 Series only, and must not be selected in the H8/3854 Series.

12.2.3 A/D Start Register (ADSR)

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

The A/D start register (ADSR) is an 8-bit read/write register for starting and stopping A/D conversion.

A/D conversion is started by writing 1 to the A/D start flag (ADSF) or by input of the designated edge of the external trigger signal, which also sets ADSF to 1. When conversion is complete, the converted data is set in the A/D result register (ADRR), and at the same time ADSF is cleared to 0.

Bit 7—A/D Start Flag (ADSF): Bit 7 controls and indicates the start and end of A/D conversion.

Bit 7: ADSF	Description	
0	Read: Indicates the completion of A/D conversion	(initial value)
	Write: Stops A/D conversion	
1	Read: Indicates A/D conversion in progress	
	Write: Starts A/D conversion	

Bits 6 to 0—Reserved Bits: Bits 6 to 0 are reserved; they are always read as 1, and cannot be modified.

12.3 Operation

12.3.1 A/D Conversion Operation

The A/D converter operates by successive approximations, and yields its conversion result as 8-bit data.

A/D conversion begins when software sets the A/D start flag (bit ADSF) to 1. Bit ADSF keeps a value of 1 during A/D conversion, and is cleared to 0 automatically when conversion is complete.

The completion of conversion also sets bit IRRAD in interrupt request register 2 (IRR2) to 1. An A/D conversion end interrupt is requested if bit IENAD in interrupt enable register 2 (IENR2) is set to 1.

If the conversion time or input channel needs to be changed in the A/D mode register (AMR) during A/D conversion, bit ADSF should first be cleared to 0, stopping the conversion operation, in order to avoid malfunction.

12.3.2 Start of A/D Conversion by External Trigger Input

The A/D converter can be made to start A/D conversion by input of an external trigger signal. External trigger input is enabled at pin $\overline{\text{ADTRG}}$ when bit IRQ4 in port mode register 2 (PMR2) is set to 1, and bit TRGE in AMR is set to 1. Then when the input signal edge designated in bit IEG4 of the IRQ edge select register (IEGR) is detected at pin $\overline{\text{ADTRG}}$, bit ADSF in ADSR will be set to 1, starting A/D conversion.

Figure 12.2 shows the timing.

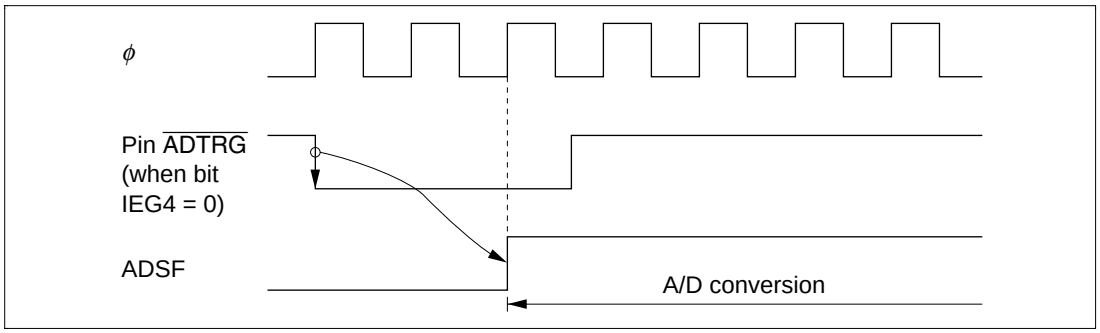


Figure 12.2 External Trigger Input Timing

12.4 Interrupts

When A/D conversion ends (ADSF changes from 1 to 0), bit IRRAD in interrupt request register 2 (IRR2) is set to 1.

A/D conversion end interrupts can be enabled or disabled by means of bit IENAD in interrupt enable register 2 (IENR2).

For further details see 3.3, Interrupts.

12.5 Typical Use

An example of how the A/D converter can be used is given below, using channel 1 (pin AN1) as the analog input channel. Figure 12.3 shows the operation timing.

- Bits CH3 to CH0 of the A/D mode register (AMR) are set to 0101, making pin AN_1 the analog input channel. A/D interrupts are enabled by setting bit IENAD to 1, and A/D conversion is started by setting bit ADSF to 1.
- When A/D conversion is complete, bit IRRAD is set to 1, and the A/D conversion result is stored in the A/D result register (ADRR). At the same time ADSF is cleared to 0, and the A/D converter goes to the idle state.
- Bit IENAD = 1, so an A/D conversion end interrupt is requested.
- The A/D interrupt handling routine starts.
- The A/D conversion result is read and processed.
- The A/D interrupt handling routine ends.

If ADSF is set to 1 again afterward, A/D conversion starts and steps 2 through 6 take place.

Figures 12.4 and 12.5 show flow charts of procedures for using the A/D converter.

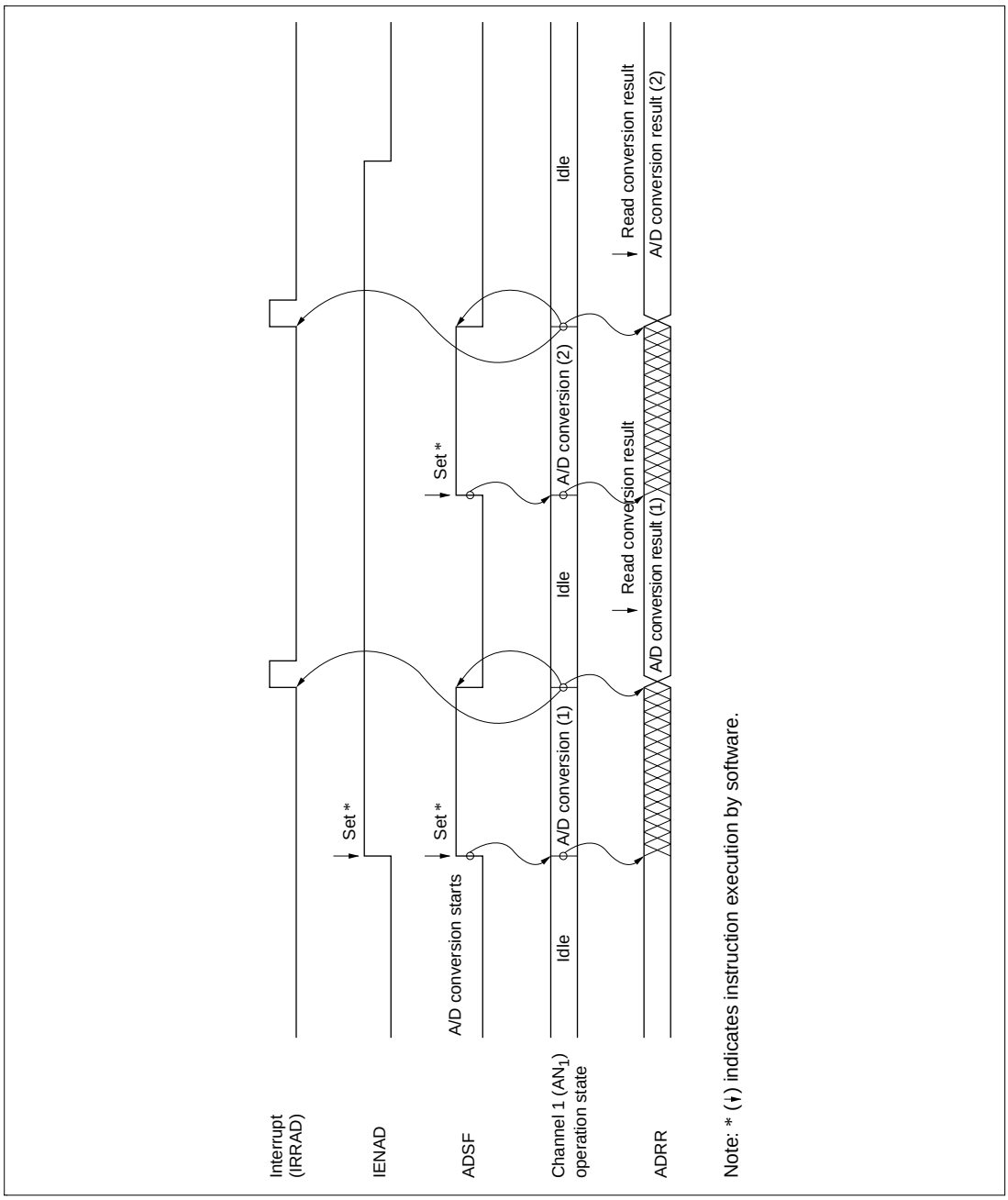


Figure 12.3 Typical A/D Converter Operation Timing

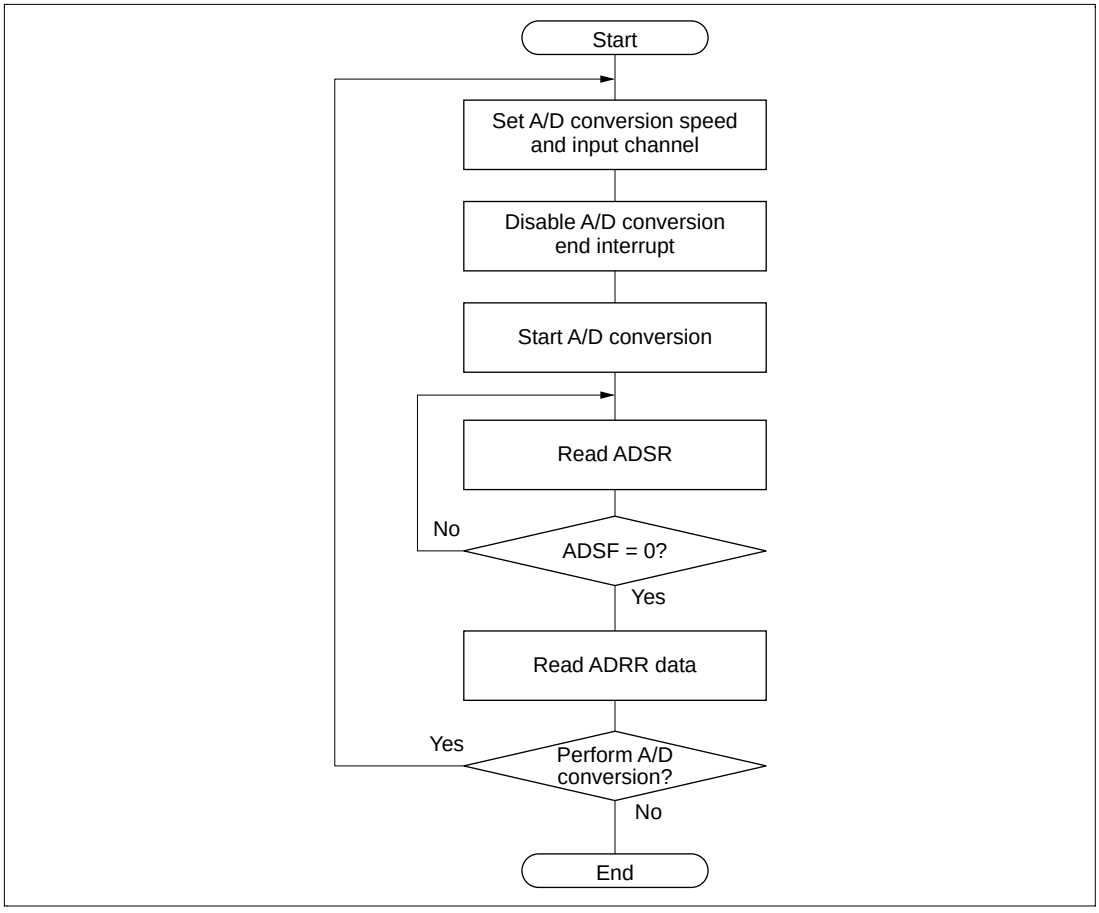


Figure 12.4 Flow Chart of Procedure for Using A/D Converter (1) (Polling by Software)

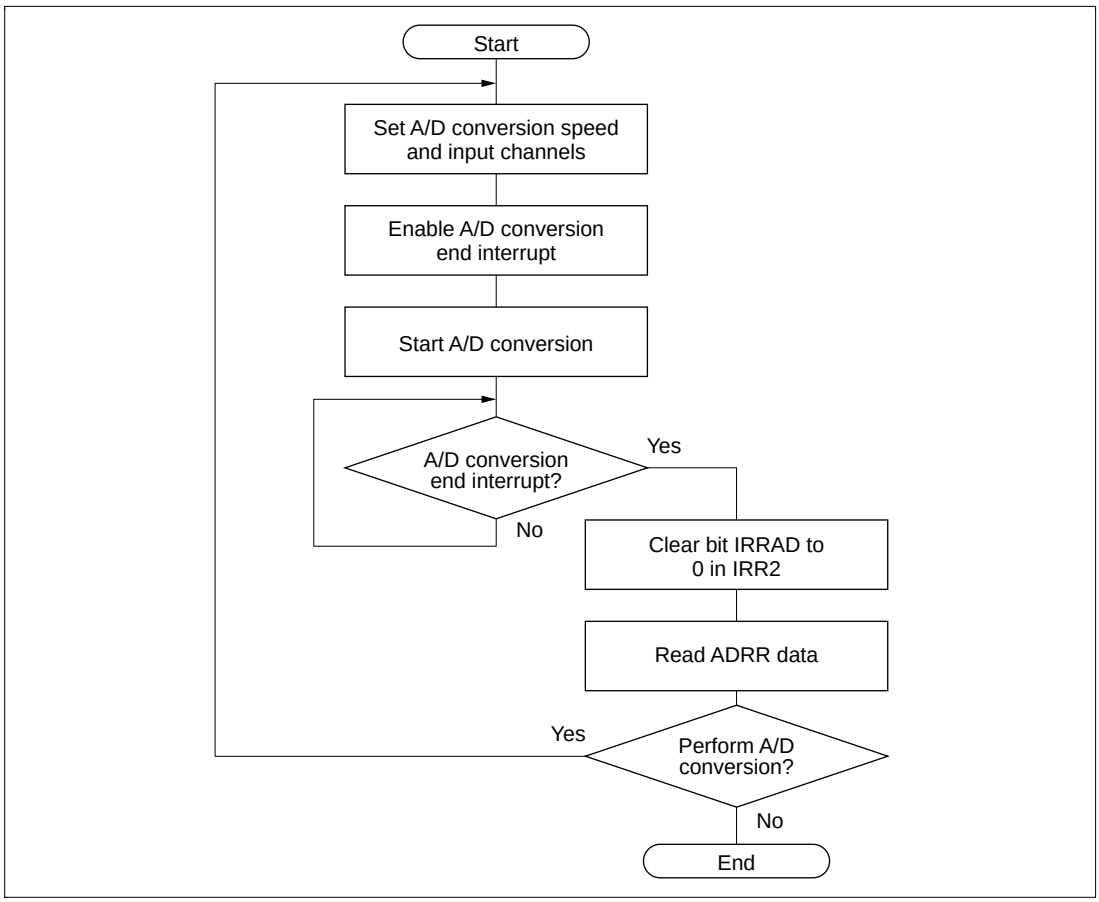


Figure 12.5 Flow Chart of Procedure for Using A/D Converter (2) (Interrupts Used)

12.6 Application Notes

- Data in the A/D result register (ADRR) should be read only when the A/D start flag (ADSF) in the A/D start register (ADSR) is cleared to 0.
- Changing the digital input signal at an adjacent pin during A/D conversion may adversely affect conversion accuracy.

Section 13 Dot Matrix LCD Controller (H8/3857 Series)

13.1 Overview

The LCD controller has built-in display RAM, and performs dot matrix LCD display. One bit of display RAM data corresponds to illumination or non-illumination of one dot on the LCD panel, making possible displays with an extremely high degree of freedom.

The LCD controller incorporates all the functions required for LCD display, allowing a dot matrix display of up to 40 × 32 dots.

I/O ports are used for the interface with the CPU, offering excellent software heritability when using a combination of MPU and LCD driver.

This module operates on the subclock, making it ideal for use in small portable devices.

13.1.1 Features

- Built-in bit-mapped display RAM (2084 bits)
  Maximum of 1280 display bits (selectable from 40 × 32 bits, 56 × 16 bits, 64 × 8 bits, 40 × 16 bits, or 40 × 8 bits)
- Choice of 1/8, 1/16, or 1/32 duty
- Low power consumption enabling extended drive on battery power
  Subclock operation
  Module standby
- Built-in 2X or 3X LCD power supply step-up circuit
- Comprehensive display control functions
  Display data read/write, display on/off control, vertical display scrolling, arbitrary area clipping, read-modify-write
- CPU interface
  I/O port interface
- Built-in contrast control circuit
- Built-in LCD power supply bleeder resistances and voltage follower type op-amp circuits

13.1.2 Block Diagram

Figure 13.1 shows a block diagram of the LCD controller.

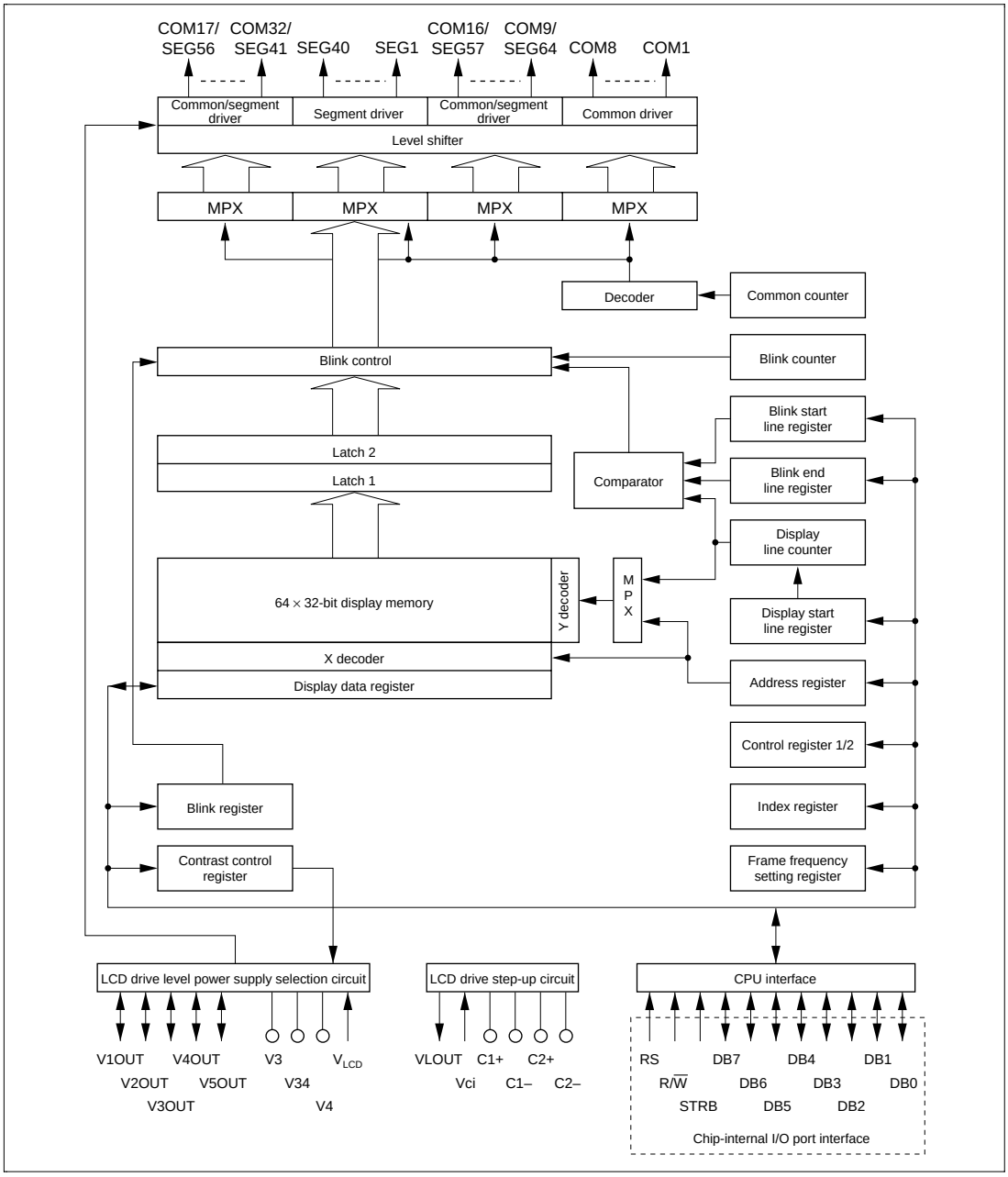


Figure 13.1 Block Diagram of LCD Controller

13.1.3 Pin Configuration

Table 13.1 shows the pins assigned to the LCD controller.

Table 13.1 Pin Configuration

Pin Name	Abbrev.	I/O	Function
Common output pins	COM1 to COM32	Output	LCD common drive pins
Segment output pins	SEG1 to SEG64	Output	LCD segment drive pins
LCD bias setting pins	V3, V4	Input	LCD bias setting
LCD test pin	V34	Input	Internal resistance test pins, shorted to V3
LCD step-up capacitance connection pins	C1+, C1– C2+, C2–	—	For connection of external capacitances for LCD step-up
LCD drive power supply level	V1OUT to V5OUT	I/O	LCD drive power supply level input/output pins
LCD step-up circuit reference power supply	V_{Ci}	Input	Reference input voltage for LCD step-up circuit, also functioning as step-up circuit power supply
LCD step-up power supply output pin	VLOUT	Output	LCD step-up voltage output pin
LCD drive power supply	V_{LCD}	Input	LCD drive power supply input pin

13.1.4 Register Configuration

The LCD controller has one index register and ten control registers, all of which are accessed via an I/O port interface. Except for the display data register (LR4), these registers cannot be read. The LCD controller register configuration is shown in table 13.2.

Table 13.2 Register Configuration

				Index Register			
Name	**Abbrev.**	**R/W**	**RS**	**IR3**	**IR2**	**IR1**	**IR0**
Index register	IR	W	0	—	—	—	—
Control register 1	LR0	W	1	0	0	0	0
Control register 2	LR1	W		0	0	0	1
Address register	LR2	W		0	0	1	0
Frame frequency setting register	LR3	W		0	0	1	1
Display data register	LR4	R/W		0	1	0	0
Display start line register	LR5	W		0	1	0	1
Blink register	LR6	W		0	1	1	0
Blink start line register	LR8	W		1	0	0	0
Blink end line register	LR9	W		1	0	0	1
Contrast control register	LRA	W		1	0	1	0

13.2 Register Descriptions

13.2.1 Index Register (IR)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	IR3	IR2	IR1	IR0
Initial value	—	—	—	—	0	0	0	0
Read/Write	—	—	—	—	W	W	W	W

IR is an 8-bit write-only register that selects one of the LCD controller's ten control registers. IR is selected when RS is 0.

Upon reset, IR is initialized to H'00.

Bits 7 to 4—Reserved Bits: Bits 7 to 4 are reserved; they should always be cleared to 0.

Bits 3 to 0—Index Register (IR3 to IR0): Bits 3 to 0 are used to select one of the LCD controller's ten control registers. The correspondence between the settings of IR3 to IR0 and the selected registers is shown in table 13.2. Other settings are invalid.

13.2.2 Control Register 1 (LR0)

Bit	7	6	5	4	3	2	1	0
	—	—	LSBY	PWR	—	SOB	DDTY1	DDTY0
Initial value	—	—	0	0	—	0	0	0
Read/Write	—	—	W	W	—	W	W	W

LR0 is an 8-bit write-only register that performs LCD module standby mode setting, step-up circuit control, switching between character display and graphic display, and drive duty selection.

Upon reset, LR0 is initialized to H'00.

Bits 7 and 6—Reserved Bits: Bits 7 and 6 are reserved; they should always be cleared to 0.

Bit 5—Module Standby (LSBY): Bit 5 is the module standby setting bit. When LSBY is set to 1, the LCD controller enters standby mode. At this time, the state of the PWR bit is not affected, but the DISP and OPON bits in LR1 are reset.

Bit 5

LSBY	Description	
0	LCD controller operates normally	(initial value)
1	Step-up and internal operations halt, display is turned off, and LCD controller enters standby mode	

Bit 4—Step-Up Circuit Operation Setting (PWR): Bit 4 selects operation or halting of the step-up circuit.

Bit 4

PWR	Description	
0	Step-up circuit halts	(initial value)
1	Step-up circuit operates	

Bit 3—Reserved Bit: Bit 3 is reserved; it should always be cleared to 0.

Bit 2—Display Mode Select (SOB): Bit 2 selects either character display mode or graphic display mode.

Bit 2	
SOB	**Description**
0	Character display mode Bits 4 to 0 of one display memory data byte are output to the segment pins (initial value)
1	Graphic display mode All bits in one display memory data byte are output to the segment pins The X address that can be output is in the range H'0 to H'4 in the case of 1/32 duty, H'0 to H'6 in the case of 1/16 duty, and H'0 to H'7 in the case of 1/8 duty

Bits 1 and 0—Display Duty Select (DDTY1, DDTY0): Bits 1 and 0 select a display duty of 1/32, 1/16, or 1/8.

Bit 1	Bit 0		
DDTY1	**DDTY0**	**Description**	
0	0	1/32 duty selected	(initial value)
	1	1/16 duty selected Y address H'10 to H'1F display data is invalid	
1	*	1/8 duty selected Y address H'8 to H'1F display data is invalid	

Note: * Don’t care

13.2.3 Control Register 2 (LR1)

Bit	7	6	5	4	3	2	1	0
	—	DISP	—	OPON	RMW	—	INC	BLK
Initial value	—	0	—	0	0	—	0	0
Read/Write	—	W	—	W	W	—	W	W

LR1 is an 8-bit write-only register that selects operation or halting of LCD display and the op-amp circuits, performs read-modify-write mode setting, and selects the address to be incremented in the display memory.

Upon reset, LR1 is initialized to H'00.

Bit 7—Reserved Bit: Bit 7 is reserved; it should always be cleared to 0.

Bit 6—LCD Operation Setting (DISP): Bit 6 selects operation or halting of the LCD display. When the LSBY bit in LR0 is set to 1, DISP is cleared.

Bit 6

DISP	Description	
0	LCD is turned off. All LCD outputs go to the V_{SS} level	(initial value)
1	LCD is turned on	

Bit 5—Reserved Bit: Bit 5 is reserved; it should always be cleared to 0.

Bit 4—Op-Amp Circuit Operation Setting (OPON): Bit 4 selects operation or halting of the op-amp circuits. When the LCD drive power supply level is applied to V1OUT to V5OUT from an external source, OPON must be cleared to 0.

When the LSBY bit in LR0 is set to 1, OPON is cleared.

Bit 4

OPON	Description	
0	Built-in op-amps are halted, and output becomes high-impedance. LCD drive voltage can be input from external source	(initial value)
1	Built-in op-amps operate	

Bit 3—Read-Modify-Write Setting (RMW): Bit 3 selects whether display memory X or Y address incrementing is carried out after a write/read access, or only after a write access (read-modify-write mode).

Bit 3

RMW	Description
0	Address is incremented after write/read access to display memory (initial value)
1	Read-modify-write mode is set In this mode, address is incremented only after write access to display memory

Bit 2—Reserved Bit: Bit 2 is reserved; it should always be cleared to 0.

Bit 1—Increment Address Select (INC): Bit 1 selects either the X address or the Y address as the address to be incremented after the display memory access specified by the RMW bit. The selected address is cleared after a display memory access with the maximum value for the valid display data area; in this case the other address is incremented.

Bit 1

INC	Description
0	Incrementing of display memory Y address has priority; X address is incremented after Y address overflow (initial value)
1	Incrementing of display memory X address has priority; Y address is incremented after X address overflow

Bit 0—Blink Operation Setting (BLK): Bit 0 enables or disables the blink function. If BLK is set to 1 while the DISP bit is set to 1 and LCD display is operating, the blink function is enabled and blinking operates in the range set by BK7 to BK0 in LR6, BSL4 to BSL0 in LR8, and BEL4 to BEL0 in LR9.

Bit 0

BLK	Description
0	Blinking is disabled (initial value)
1	Blinking is enabled

13.2.4 Address Register (LR2)

Bit	7	6	5	4	3	2	1	0
	XA2	XA1	XA0	YA4	YA3	YA2	YA1	YA0
Initial value	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W

LR2 is an 8-bit write-only register that sets the display memory X- and Y-direction addresses accessed by the CPU.

Upon reset, LR2 is initialized to H'00.

Bits 7 to 5—X Address Setting (XA2 to XA0): Bits 7 to 5 set the display memory X-direction address. A value from H'0 to H'7 can be set, but if the SOB bit in LR0 is set to 1, display data H'7 is invalid with 1/16 duty, and display data from H'5 to H'7 is invalid with 1/32 duty.

When the INC bit in LR1 is set to 1, the address is automatically incremented after the access specified by the RMW bit in LR1, and is cleared after an access with the maximum value for the valid display data area. When INC is 0 and YA4 to YA0 represent the maximum value for the valid display data area, the address is incremented after the access specified by RMW.

Bits 4 to 0—Y Address Setting (YA4 to YA0): Bits 4 to 0 set the display memory Y-direction address. A value from H'00 to H'1F can be set, but display data from H'10 to H'1F is invalid with 1/16 duty, and display data from H'08 to H'1F is invalid with 1/8 duty.

When the INC bit in LR1 is cleared to 0, the address is automatically incremented after the access specified by the RMW bit in LR1, and is cleared after an access with the maximum value for the valid display data area. When INC is 1 and XA2 to XA0 represent the maximum value for the valid display data area, the address is incremented after the access specified by RMW.

13.2.5 Frame Frequency Setting Register (LR3)

Bit	7	6	5	4	3	2	1	0
	—	—	FS5	FS4	FS3	FS2	FS1	FS0
Initial value	—	—	0	0	0	0	0	0
Read/Write	—	—	W	W	W	W	W	W

LR3 is an 8-bit write-only register that sets the frame frequency.

Upon reset, LR3 is initialized to H'00.

Bits 7 and 6—Reserved Bits: Bits 7 and 6 are reserved; they should always be cleared to 0.

Bits 5 to 0—Frame Frequency Setting (FS5 to FS0): Bits 5 to 0 control the subclock division ratio and set the LCD frame frequency. The relationship between the LCD frame frequency f_F (Hz), the subclock frequency f_W (Hz), the division ratio r, and the LCD duty 1/N is as follows:

$$f_F = \frac{f_W}{r \times N}$$

Set a division ratio suitable for the characteristics of the LCD panel used. The correspondence between register settings and division ratios is shown in table 13.3.

Table 13.3 Register Settings and Division Ratios

FS						Division	FS						Division	FS						Division	FS						Division
5	4	3	2	1	0	Ratio r	5	4	3	2	1	0	Ratio r	5	4	3	2	1	0	Ratio r	5	4	3	2	1	0	Ratio r
0	0	0	0	0	0	2	0	1	0	0	0	0	34	1	0	0	0	0	0	66	1	1	0	0	0	0	98
0	0	0	0	0	1	4	0	1	0	0	0	1	36	1	0	0	0	0	1	68	1	1	0	0	0	1	100
0	0	0	0	1	0	6	0	1	0	0	1	0	38	1	0	0	0	1	0	70	1	1	0	0	1	0	102
0	0	0	0	1	1	8	0	1	0	0	1	1	40	1	0	0	0	1	1	72	1	1	0	0	1	1	104
0	0	0	1	0	0	10	0	1	0	1	0	0	42	1	0	0	1	0	0	74	1	1	0	1	0	0	106
0	0	0	1	0	1	12	0	1	0	1	0	1	44	1	0	0	1	0	1	76	1	1	0	1	0	1	108
0	0	0	1	1	0	14	0	1	0	1	1	0	46	1	0	0	1	1	0	78	1	1	0	1	1	0	110
0	0	0	1	1	1	16	0	1	0	1	1	1	48	1	0	0	1	1	1	80	1	1	0	1	1	1	112
0	0	1	0	0	0	18	0	1	1	0	0	0	50	1	0	1	0	0	0	82	1	1	1	0	0	0	114
0	0	1	0	0	1	20	0	1	1	0	0	1	52	1	0	1	0	0	1	84	1	1	1	0	0	1	116
0	0	1	0	1	0	22	0	1	1	0	1	0	54	1	0	1	0	1	0	86	1	1	1	0	1	0	118
0	0	1	0	1	1	24	0	1	1	0	1	1	56	1	0	1	0	1	1	88	1	1	1	0	1	1	120
0	0	1	1	0	0	26	0	1	1	1	0	0	58	1	0	1	1	0	0	90	1	1	1	1	0	0	122
0	0	1	1	0	1	28	0	1	1	1	0	1	60	1	0	1	1	0	1	92	1	1	1	1	0	1	124
0	0	1	1	1	0	30	0	1	1	1	1	0	62	1	0	1	1	1	0	94	1	1	1	1	1	0	126
0	0	1	1	1	1	32	0	1	1	1	1	1	64	1	0	1	1	1	1	96	1	1	1	1	1	1	128

Examples of subclock frequency, LCD duty, and division ratio settings, and frame frequencies, are shown in table 13.4.

Table 13.4 Sample Frame Frequency Settings

		Subclock Frequency (kHz)	
Display Duty 1/N		**32.768**	**38.4**
1/8	Division ratio r	48	56
	Frame frequency f_F (Hz)	85.3	85.7
1/16	Division ratio r	24	28
	Frame frequency f_F (Hz)	85.3	85.7
1/32	Division ratio r	12	14
	Frame frequency f_F (Hz)	85.3	85.7

13.2.6 Display Data Register (LR4)

Bit	7	6	5	4	3	2	1	0
	D7	D6	D5	D4	D3	D2	D1	D0
Initial value	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

LR4 is an 8-bit read/write register used to perform read/write access to the display memory specified by XA2 to XA0 and YA4 to YA0 in LR2.

In a write to display memory, the write is performed directly to the display memory via this register. In a read, the data is temporarily latched into this register before being output to the bus.

After a reset, the display memory and LR4 contents are undefined.

13.2.7 Display Start Line Register (LR5)

Bit	7	6	5	4	3	2	1	0
	—	—	—	ST4	ST3	ST2	ST1	ST0
Initial value	—	—	—	0	0	0	0	0
Read/Write	—	—	—	W	W	W	W	W

LR5 is an 8-bit write-only register that specifies the line at which display starts.

Upon reset, LR5 is initialized to H'00.

Bits 7 to 5—Reserved Bits: Bits 7 to 5 are reserved; they should always be cleared to 0.

Bits 4 to 0—Display Start Line Setting (ST4 to ST0): Bits 4 to 0 specify the line at which display starts. Set a value of [display start line – 1].

Changing the setting in this register enables vertical scrolling to be implemented. The possible settings are 0 to 31 for 1/32 duty, 0 to 15 for 1/16 duty, and 0 to 7 for 1/8 duty. Display will not be performed normally if these ranges are exceeded.

13.2.8 Blink Register (LR6)

Bit	7	6	5	4	3	2	1	0
	BK7	BK6	BK5	BK4	BK3	BK2	BK1	BK0
Initial value	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W

LR6 is an 8-bit write-only register that specifies blink areas. An area is made to blink by writing 1 to the corresponding bit in this register. There are no restrictions on areas that can blink simultaneously, and the entire screen can be made to blink by writing 1 to all the bits. The setting in this register is valid only when the BLK bit in LR1 is set to 1.

The blink areas corresponding to the register bits depend on the value of the SOB bit in LR0, as shown below.

SOB	BK7	BK6	BK5	BK4	BK3	BK2	BK1	BK0
0	SEG36 to SEG40	SEG31 to SEG35	SEG26 to SEG30	SEG21 to SEG25	SEG16 to SEG20	SEG11 to SEG15	SEG6 to SEG10	SEG1 to SEG5
1	SEG57 to SEG64	SEG49 to SEG56	SEG41 to SEG48	SEG33 to SEG40	SEG25 to SEG32	SEG17 to SEG24	SEG9 to SEG16	SEG1 to SEG8

Upon reset, LR6 is initialized to H'00.

13.2.9 Blink Start Line Register (LR8)

Bit	7	6	5	4	3	2	1	0
	—	—	—	BSL4	BSL3	BSL2	BSL1	BSL0
Initial value	—	—	—	0	0	0	0	0
Read/Write	—	—	—	W	W	W	W	W

LR8 is an 8-bit write-only register that specifies the start line of an area made to blink.

Upon reset, LR8 is initialized to H'00.

Bits 7 to 5—Reserved Bits: Bits 7 to 5 are reserved; they should always be cleared to 0.

Bits 4 to 0—Blink Start Line Setting (BSL4 to BSL0): Bits 4 to 0 specify the start line of an area made to blink. Set a value of [blink start line – 1].

The possible settings are 0 to 31 for 1/32 duty, 0 to 15 for 1/16 duty, and 0 to 7 for 1/8 duty. Normal operation is not guaranteed if these ranges are exceeded.

13.2.10 Blink End Line Register (LR9)

Bit	7	6	5	4	3	2	1	0
	—	—	—	BEL4	BEL3	BEL2	BEL1	BEL0
Initial value	—	—	—	0	0	0	0	0
Read/Write	—	—	—	W	W	W	W	W

LR9 is an 8-bit write-only register that specifies the end line of an area made to blink.

Upon reset, LR9 is initialized to H'00.

Bits 7 to 5—Reserved Bits: Bits 7 to 5 are reserved; they should always be cleared to 0.

Bits 4 to 0—Blink End Line Setting (BEL4 to BEL0): Bits 4 to 0 specify the end line of an area made to blink. Set a value of [blink end line - 1].

The possible settings are 0 to 31 for 1/32 duty, 0 to 15 for 1/16 duty, and 0 to 7 for 1/8 duty. Normal operation is not guaranteed if these ranges are exceeded.

13.2.11 Contrast Control Register (LRA)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	CCR3	CCR2	CCR1	CCR0
Initial value	—	—	—	—	0	0	0	0
Read/Write	—	—	—	—	W	W	W	W

LRA is an 8-bit write-only register that specifies the contrast control resistance value.

Upon reset, LRA is initialized to H'00.

Bits 7 to 4—Reserved Bits: Bits 7 to 4 are reserved; they should always be cleared to 0.

Bits 3 to 0—Contrast Control Setting (CCR3 to CCR0): Bits 3 to 0 specify the value of the contrast control resistance between the V_{LCD} and V1 levels. By adjusting the contrast control resistance between the V_{LCD} and V1 levels, it is possible to adjust the contrast of the LCD panel. The contrast control resistance can be set in the range from 0.1R to 1.6R, where R is the LCD bleeder resistance.

Bit 3	Bit 2	Bit 1	Bit 0		
CCR3	**CCR2**	**CCR1**	**CCR0**	**Contrast Control Resistance**	
0	0	0	0	1.6R	(initial value)
			1	1.5R	
		1	0	1.4R	
			1	1.3R	
	1	0	0	1.2R	
			1	1.1R	
		1	0	1.0R	
			1	0.9R	
1	0	0	0	0.8R	
			1	0.7R	
		1	0	0.6R	
			1	0.5R	
	1	0	0	0.4R	
			1	0.3R	
		1	0	0.2R	
			1	0.1R	

13.3 Operation

13.3.1 System Overview

The LCD controller operates at 1/32, 1/16, or 1/8 duty. The display size is a maximum of 40×32 dots (4 rows of 8 columns with a 5×8-dot font). As the LCD controller operates on the subclock to perform display control, the time, etc., can be constantly displayed. As this module includes a built-in 2X or 3X LCD power supply step-up circuit, an LCD system can be configured with just a few external parts (resistors and capacitors). Also, since data in the display RAM is retained even in module standby mode, and step-up operation is not performed, low power consumption can be achieved without affecting the display.

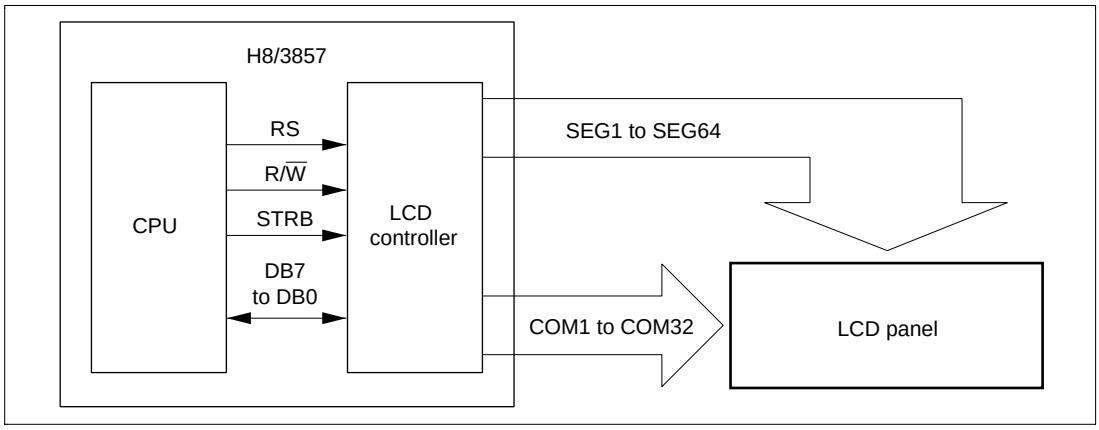


Figure 13.2 System Block Diagram

13.3.2 CPU Interface

The LCD controller's registers are not included in the memory map shown in figure 2.16 (a). They are controlled from the CPU by means of chip-internal LCD pins DB7 to DB0, RS, R/$\overline{W}$, and STRB, via chip-internal I/O ports 9 and A. The pin configuration is shown in table 13.5, and an example of the timing for access to registers in the LCD controller is shown in figure 13.3. For information on port 9 and port A, see the descriptions in section 8, I/O Ports.

Table 13.5 Pin Configuration

Pin Name	Abbrev.	I/O	Function
Data bus pins	DB7 to DB0	I/O	When R/$\overline{W}$ = 0, these pins input data to be written to a register; when R/$\overline{W}$ = 1, they output data read from a register
Register selector pin	RS	Input	When R/S = 0, the index register is selected; when RS = 1, a control register is selected
Read/write select pin	R/$\overline{W}$	Input	When R/$\overline{W}$ = 0, write access is selected; when R/$\overline{W}$ = 1, read access is selected
Strobe pin	STRB	Input	At the fall of STRB, read or write access, as selected by R/$\overline{W}$, is performed on the register selected by RS

Writing to Index Register

When RS and R/$\overline{W}$ are both cleared to 0, data DB7 to DB0 is written to the index register (IR) at the falling edge of STRB. Do not change RS or R/$\overline{W}$ at the fall of STRB.

Reading and Writing to Control Registers

To access a control register, data indicating the number of the register to be accessed must be written to the index register (IR) before making the access. The register number data to be written to IR is shown in table 13.2. As the register number written to IR is retained until IR is written to again, if the same control register is accessed repeatedly, it is not necessary to write to IR each time.

In a write to a control register, when RS has been set to 1 and R/$\overline{W}$ cleared to 0, data DB7 to DB0 is written to the control register specified by the index register (IR) at the falling edge of STRB.

Except for the display data register (LR4), control registers cannot be read. In a read of LR4, when the LR4 register number is written to the index register (IR), and RS and R/$\overline{W}$ are both set to 1, DB7 to DB0 are set to output mode, and the display memory data at the address specified by the address register (LR2) is output from DB7 to DB0 at the rising edge of STRB. If a read is also performed in the next cycle, the data output is held until the next rise of STRB, but if a write is performed in the next cycle, DB7 to DB0 are set to input mode from the point at which R/$\overline{W}$ is cleared to 0, and the output is cleared.

In either case, do not change RS or R/$\overline{W}$ at the fall of STRB.

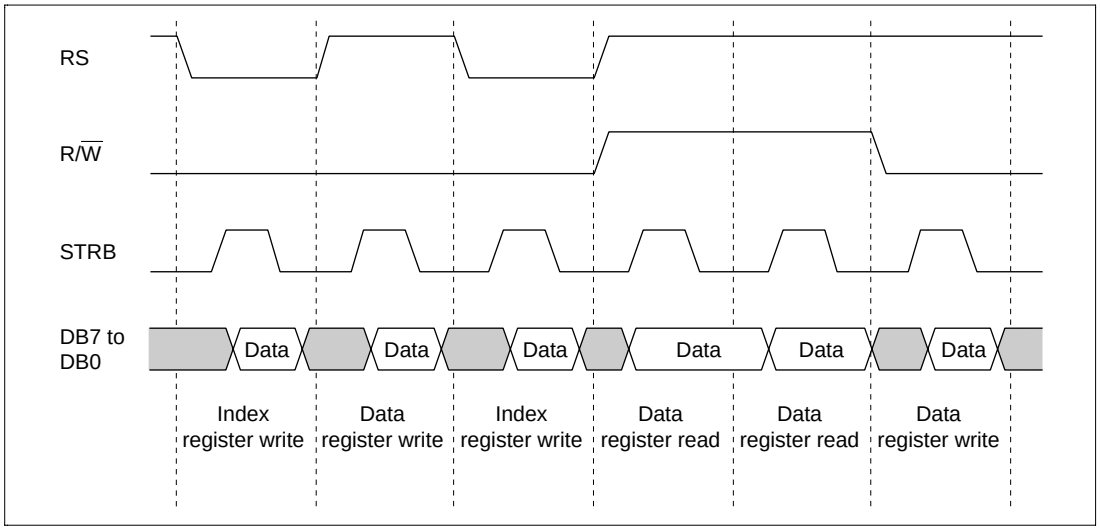


Figure 13.3 Example of Timing Sequence for 8-Bit Data Transfer

Notes on Use of Chip-Internal I/O Ports

For LCD controller interface internal ports 9 and A, port input/output is controlled by means of PCR9 and PCRA in the same way as for ordinary I/O ports, and in output mode, the values set in PDR9 or PDRA are output. Also, LCD controller internal pins RS, R/$\overline{W}$, and STRB are input-only pins, and DB7 to DB0 input/output is controlled by R/$\overline{W}$. Therefore, the following points must be noted.

1. After reset release and standby mode release

   Since the chip's internal I/O ports go to the high-impedance state in a reset and in standby mode, in initialization after reset or standby mode release, H'06 should be set in PDRA, and H'07 in PCRA. This will set port A to output mode. If the PDRA setting were H'00, there would be a possibility of the index register (IR) being written to.

2. Changing register read/write setting

   When an LCD controller register is read (R/$\overline{W}$ = 1), DB7 to DB0 output data from the LCD controller side, and so port 9 must be set to input mode. Therefore, H'00 must be written to PCR9, setting port 9 to input mode, before changing the R/$\overline{W}$ setting from 0 to 1. When writing data to an LCD controller register, first change R/$\overline{W}$ from 1 to 0, then write H'FF to PCR9, setting port 9 to output mode.

   Examples of display data register (LR4) read/write access when read-modify-write is designated are shown below.

[Set index register to display data register]

- Port A set to output mode, RMW set to 1

```
MOV.W    #H'0100,R1
MOV.W    #H'04FF,R0
MOV.B    R1L,@PDRA ..... Clear R/W to 0
MOV.B    R0H,@PDR9
MOV.B    R0L,@PCR9 ..... Output H'04 from port 9
MOV.B    R1H,@PDRA
MOV.B    R1L,@PDRA ..... Write H'04 to index register
```

[Read display data register]

```
MOV.B    R1L,@PCR9 ..... Set port 9 to input mode
MOV.W    #H'0706, R2
MOV.B    R2L,@PDRA ..... Set R/W to 1
MOV.B    R2H,@PDRA
MOV.B    @PDR9, R0H ..... Read PDR9 into general register
MOV.B    R2L,@PDRA
```

[Write to display data register]

```
MOV.W    #H'0504,R3
MOV.B    R3L,@PDRA ..... Clear R/W to 0
NOT.B    R0H        ..... Invert general register data
MOV.B    R0H,@PDR9
MOV.B    R0L,@PCR9 ..... Set port 9 to output mode
MOV.B    R3H,@PDRA
MOV.B    R3L @PDRA ..... Write data to display data register
```

13.3.3 LCD Drive Pin Functions

Common/Segment Output Switching

Among the LCD controller's LCD drive outputs, COM9 to COM32 and SEG64 to SEG41 are switched according to the display duty and display mode.

The display duty is set by control register 1 (LR0) bits DDTY1 and DDTY0, and the display mode by bit SOB.

(1) When SOB = 0 (character display mode)

- 1/8 duty (DDTY1 = 1)
  Common outputs: COM1 to COM8
  Segment outputs: SEG1 to SEG40

Note: COM9/SEG64 to COM16/SEG57 output common signal non-selection waveforms, COM17/SEG56 to COM24/SEG49 output the same waveforms as COM1 to COM8, and COM25/SEG48 to COM32/SEG41 output common signal non-selection waveforms.

- 1/16 duty (DDTY1 = 0, DDTY0 = 1)
  Common outputs: COM1 to COM16
  Segment outputs: SEG1 to SEG40

Note: COM17/SEG56 to COM32/SEG41 output the same waveforms as COM1 to COM16.

- 1/32 duty (DDTY1 = 0, DDTY0 = 0)
  Common outputs: COM1 to COM32
  Segment outputs: SEG1 to SEG40

(2) When SOB = 1 (graphic display mode)

- 1/8 duty (DDTY1 = 1)
  Common outputs: COM1 to COM8
  Segment outputs: SEG1 to SEG64

- 1/16 duty (DDTY1 = 0, DDTY0 = 1)
  Common outputs: COM1 to COM16
  Segment outputs: SEG1 to SEG56

- 1/32 duty (DDTY1 = 0, DDTY0 = 0)
  Common outputs: COM1 to COM32
  Segment outputs: SEG1 to SEG40

Table 13.6 Pin Functions According to Display Mode and Display Duty

	Function					
	SOB = 0 (Character Display Mode)			SOB = 1 (Graphic Display Mode)		
Pin Name	1/8 Duty	1/16 Duty	1/32 Duty	1/8 Duty	1/16 Duty	1/32 Duty
COM1 to COM8	COM1 to COM8	COM1 to COM16	COM1 to COM16	COM1 to COM8	COM1 to COM16	COM1 to COM16
COM9/SEG64 to COM16/SEG57	Common signal non-selection waveform			SEG64 to SEG57		
SEG1 to SEG40	SEG1 to SEG40	SEG1 to SEG40	SEG1 to SEG40	SEG1 to SEG56	SEG1 to SEG56	SEG1 to SEG40
COM32/SEG41 to COM25/SEG48	Common signal non-selection waveform	COM16 to COM1	COM32 to COM17			COM32 to COM17
COM24/SEG49 to COM17/SEG56	COM8 to COM1					

13.3.4 Display Memory Configuration and Display

The LCD controller includes 64 × 32-bit bit-mapped display memory. As the display memory configuration, a 5-bit × 8 or 8-bit × n (n = 5, 7, or 8) X-direction combination can be selected, while the Y-direction configuration is 32 bits. Display data written from the CPU is stored horizontally with the MSB at the left and the LSB at the right, as shown in figure 13.4. On the display, 1 data corresponds to illumination (black), and 0 data to non-illumination (colorless).

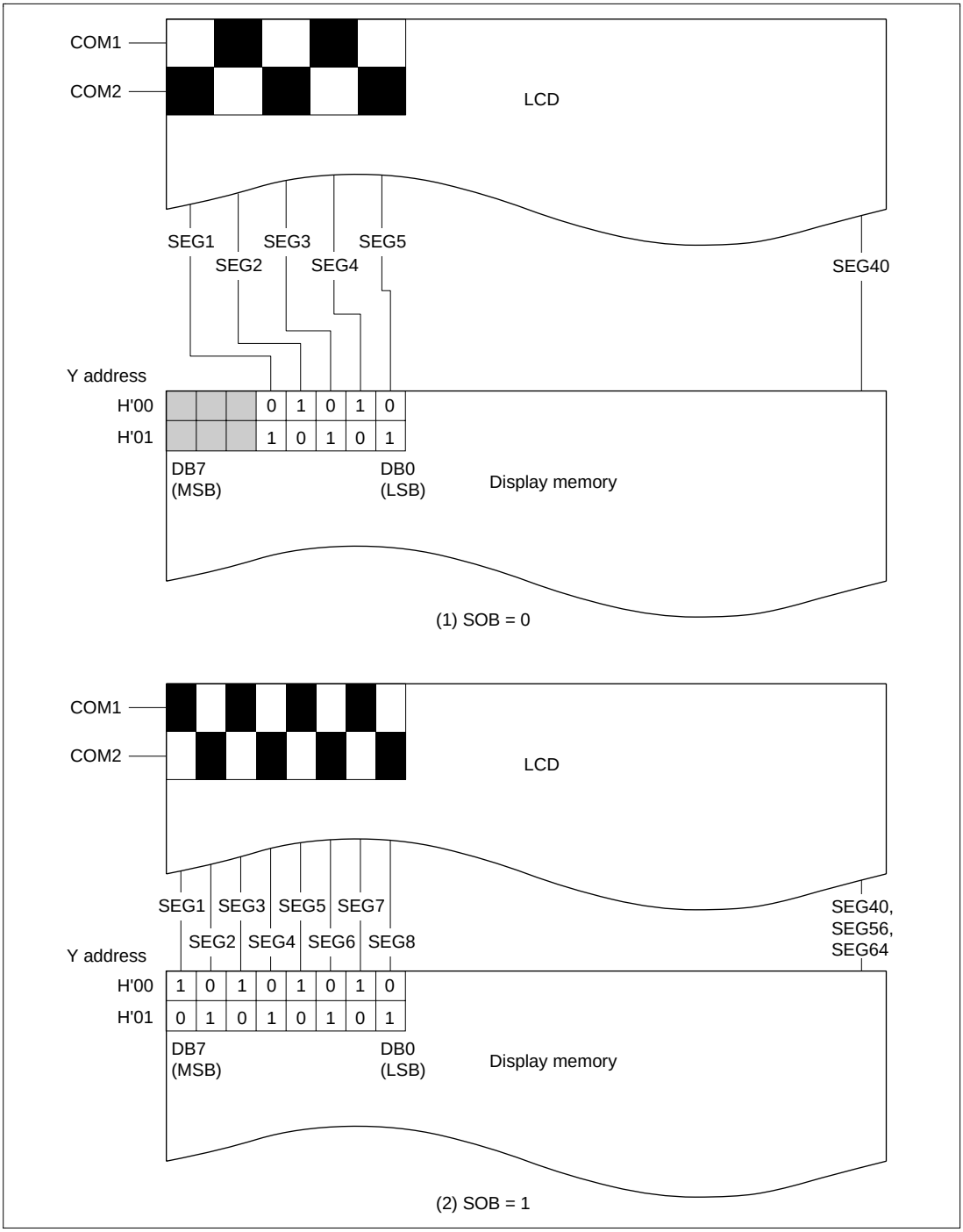


Figure 13.4 Memory Data and Display

13.3.5 Display Data Output

The LCD controller has a character display mode (SOB = 0) in which only 5 bits of each display data byte can be output to perform efficient 5-dot × 8-dot character output, and a graphic display mode (SOB = 1) in which all the bits of a data byte can be output to perform efficient full-dot graphic display. The relationship between the display duty and output pins in each mode is shown in figure 13.5.

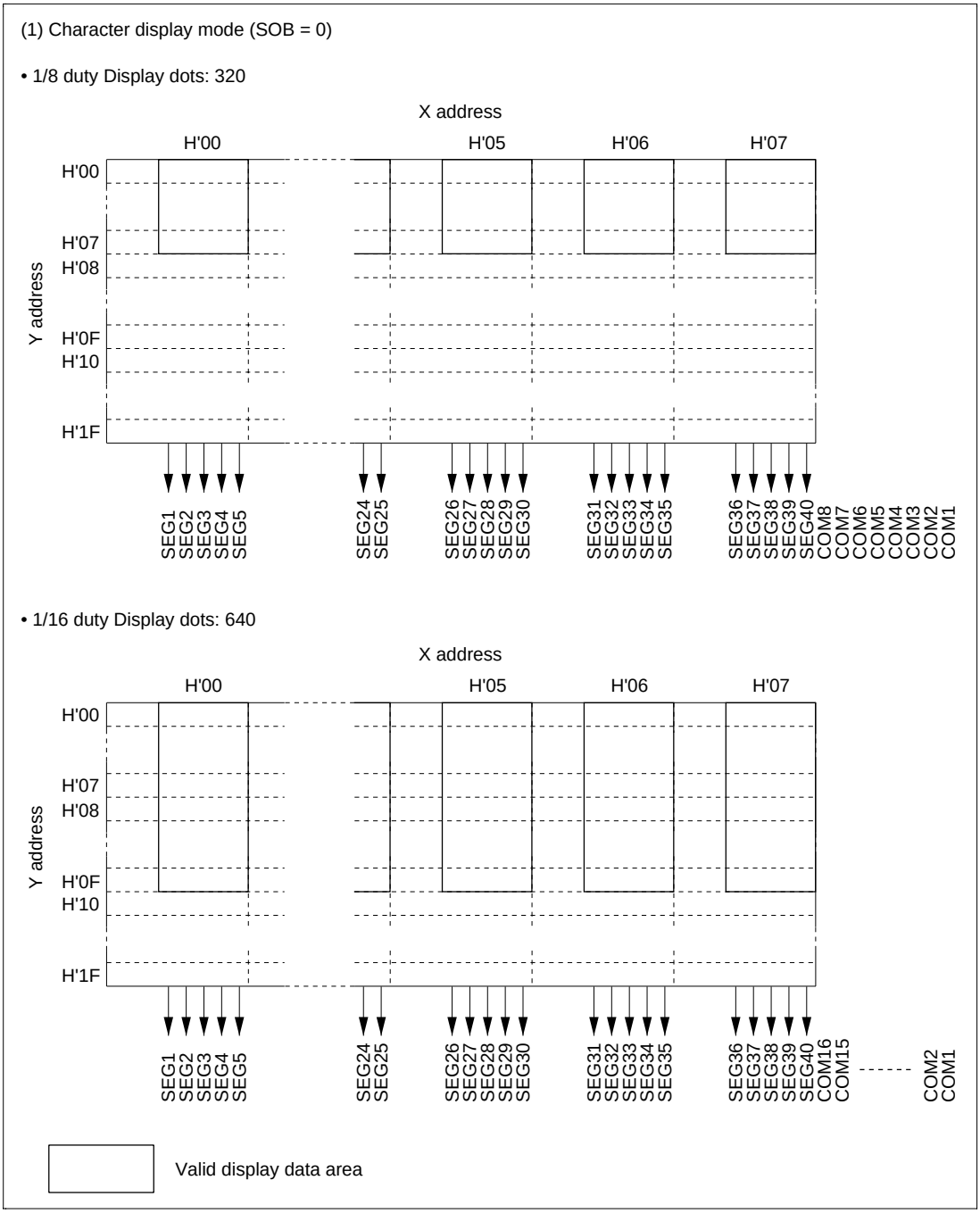


Figure 13.5 Display Duty and Valid Display Data Area (1)

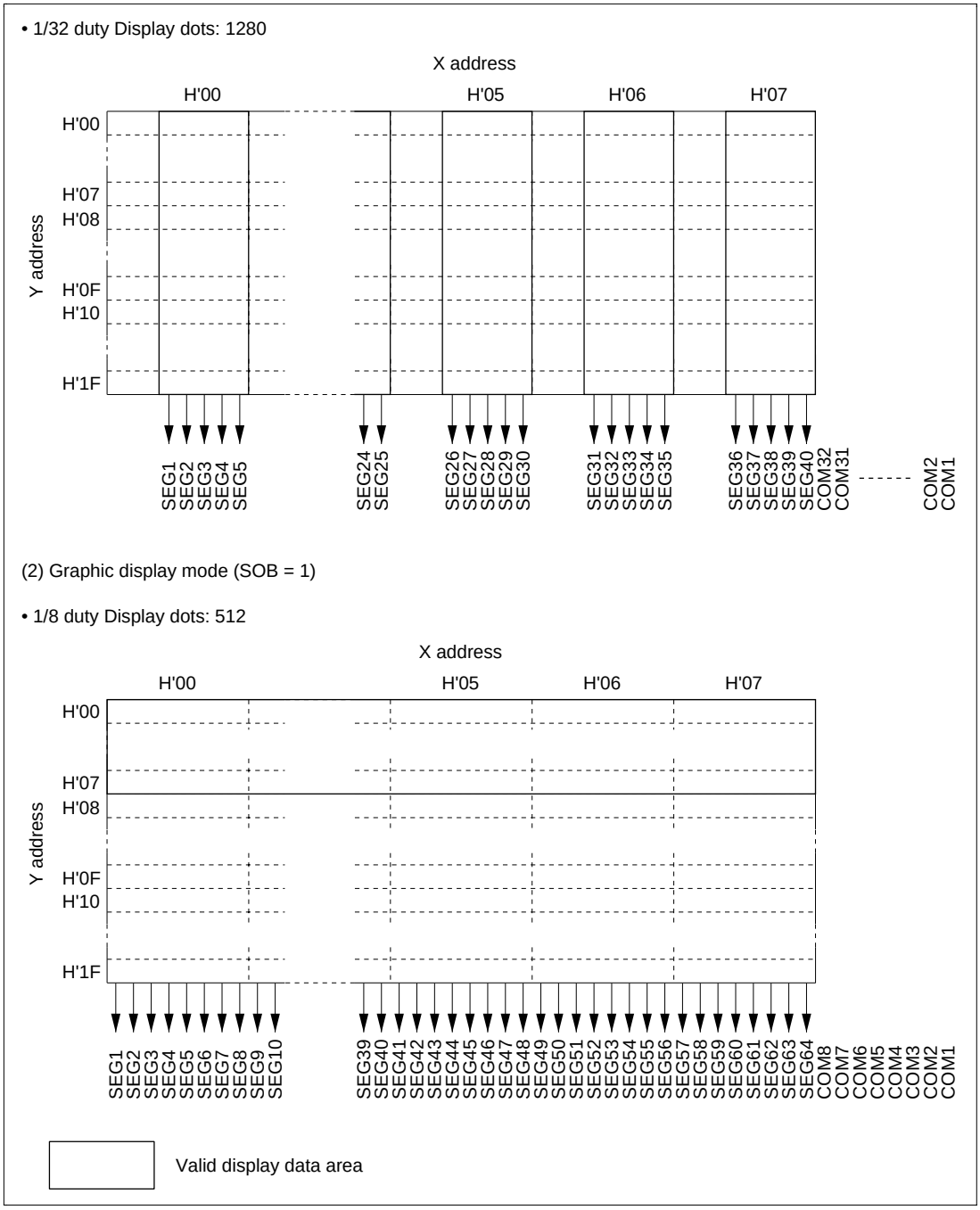


Figure 13.5 Display Duty and Valid Display Data Area (2)

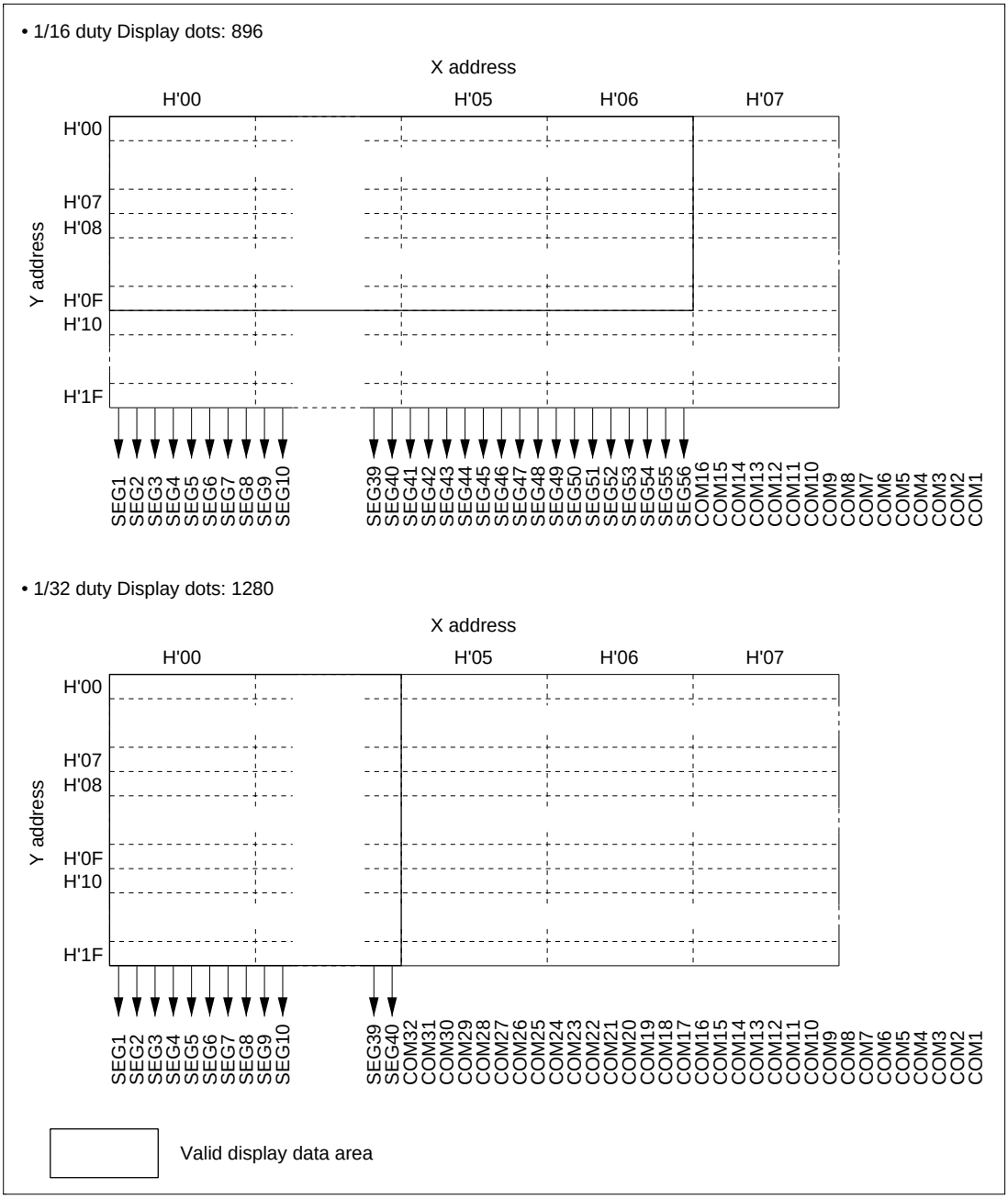


Figure 13.5 Display Duty and Valid Display Data Area (3)

13.3.6 Register and Display Memory Access

Register Access

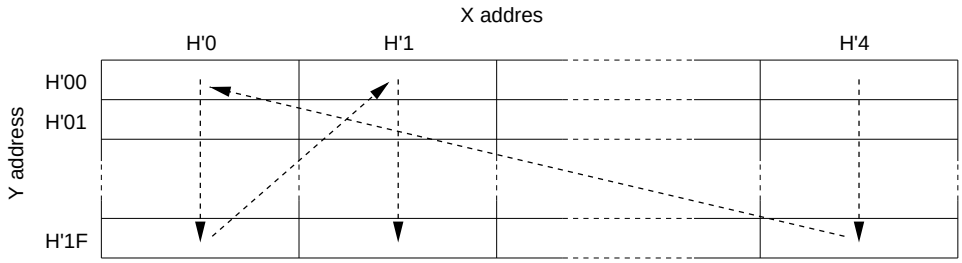
To access a register, RS is first cleared to 0 and the register number of the register to be accessed is set in the index register. Then RS is set to 1, enabling the specified register to be accessed. Some internal registers have nonexistent bits; 0 must be written to these bits. The display data register (LR4) is the only register that can be read.

Display Memory Access

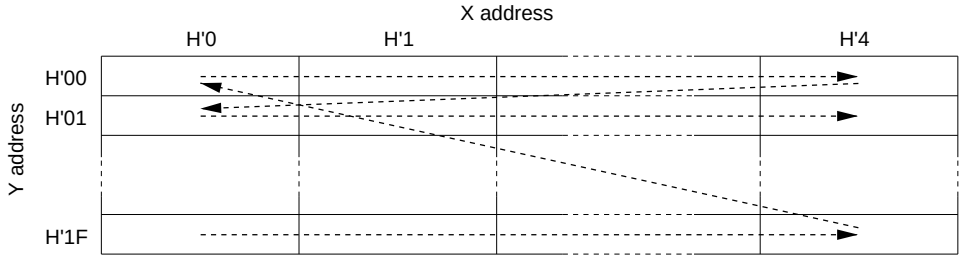
To access the display memory, the address to be accessed is set in the address register (LR2). The memory is then accessed via the display data register (LR4). This access can be performed without awareness of the display-side read. See figure 13.6 for the procedure.

After the respective display data register (LR4) accesses, the X and Y addresses are automatically incremented on the basis of the value set in the INC bit in control register 2 (LR1), and therefore address settings need not be made each time.

With 1/32 duty (DDTY1 = 0, DDTY0 = 0) in graphic display mode (SOB = 1), if INC = 0 the X address remains the same in each read/write access to the display data register (LR4), while the Y address is automatically incremented up to H'1F. After reaching H'1F, the Y address returns to H'00 again, and the X address is simultaneously incremented. If INC = 1, on the other hand, the Y address remains the same in each read/write access to the display data register (LR4), while the X address is automatically incremented up to H'4. After reaching H'4, the X address returns to H'0 again, and the Y address is simultaneously incremented. In this way, consecutive read/write accesses can be made to the entire display memory area.



(1) Priority given to Y-direction data access (INC = 0)



(2) Priority given to X-direction data access (INC = 1)

Notes: 1. Address register (LR2) bits XA2 to XA0 show the X address, and bits YA4 to YA0 show the Y address.
2. X address operation
(1) SOB = 0: Address becomes H'0 after H'7, regardless of the display duty.
(2) SOB = 1: Address becomes H'0 after H'6 when 1/16 duty is set, and after H'7 when 1/8 duty is set.
3. Y address operation
Address becomes H'00 after H'0F when 1/16 duty is set, and after H'07 when 1/8 duty is set.

Figure 13.6 Display Memory Access Methods (SOB = 1, 1/32 Duty)

Reading for Display

The LCD controller's display RAM is of the dual-port type, with accesses from the CPU and reads for LCD display independent of each other. This allows flexible interfacing.

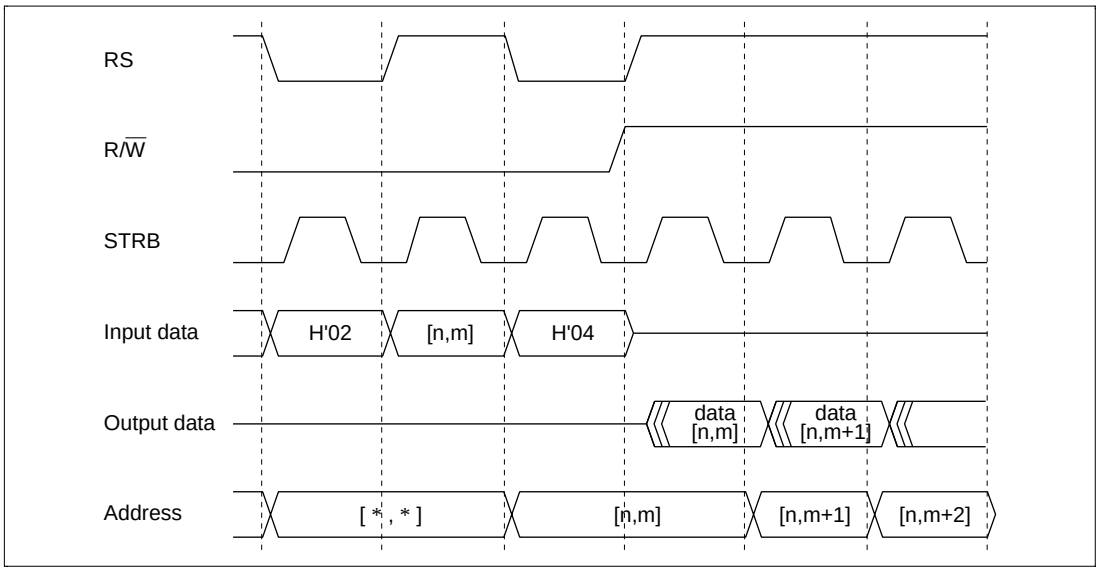


Figure 13.7 Memory Read Procedure

Read-Modify-Write Mode

In the normal state, the X or Y address is incremented after both read and write accesses to the display memory. In read-modify-write mode, the address is incremented only after a write, and remains the same after a read. By using this mode, it is possible to read previously written data, process that data, and then write it back to the same address.

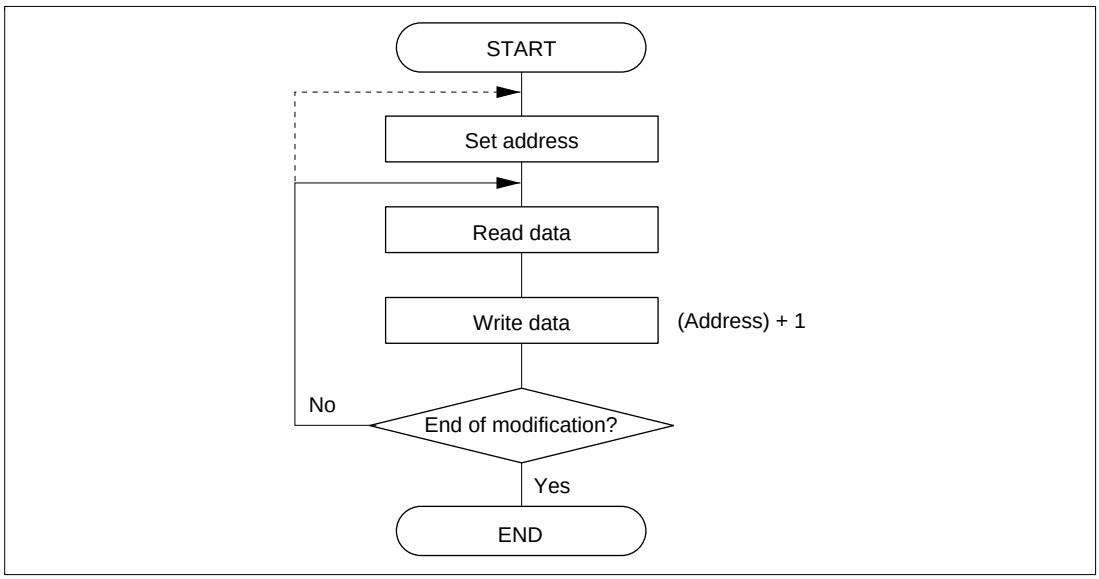


Figure 13.8 Read-Modify-Write Mode Flowchart

13.3.7 Scroll Function

The LCD controller allows vertical scrolling of any number of lines to be performed by specifying the display start line. Figure 13.9 shows the relationship between the display memory and Y address, and the display memory and LCD display after scrolling, for 1/16 duty and 1/8 duty settings. If the display start address is set to H'01, the data at Y address H'00 is displayed in the 16th line. Therefore, when the display is scrolled in order to show the next screen, the data at Y address H'00 must be rewritten with the next display data. This data update should be carried out after the display is scrolled.

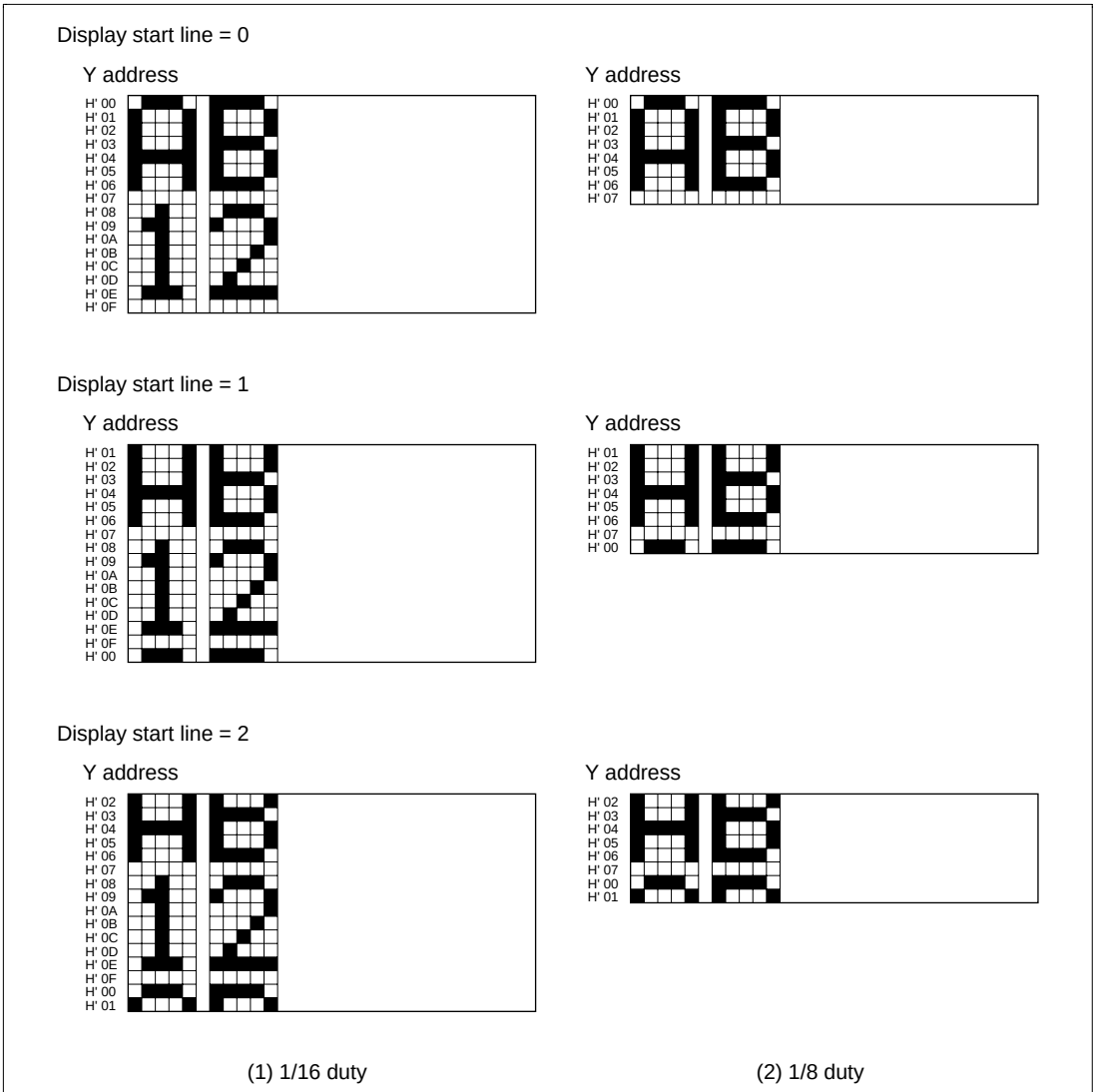


Figure 13.9 Vertical Scrolling

13.3.8 Blink Function

Dot Matrix Display Blinking

The LCD controller can perform blinking display in any area. With an 80 Hz frame frequency, the display goes on and off in a cycle of approximately 1.6 seconds.

To set a blink area, in the horizontal direction the line unit is specified by means of the blink start line register (LR8) and blink end line register (LR9), while in the vertical direction a 5-bit unit (SOB = 0) or 8-bit unit (SOB = 1) is set in the blink register (LR6). When 1 is set in the blink register (LR6), blinking of the corresponding dot is controlled. After making these register settings, blinking is started by setting the BLK bit in control register 2 (LR1) to 1. As the blink area is designated by an absolute specification with respect to the display memory, if the display is scrolled the blink area also shifts accordingly.

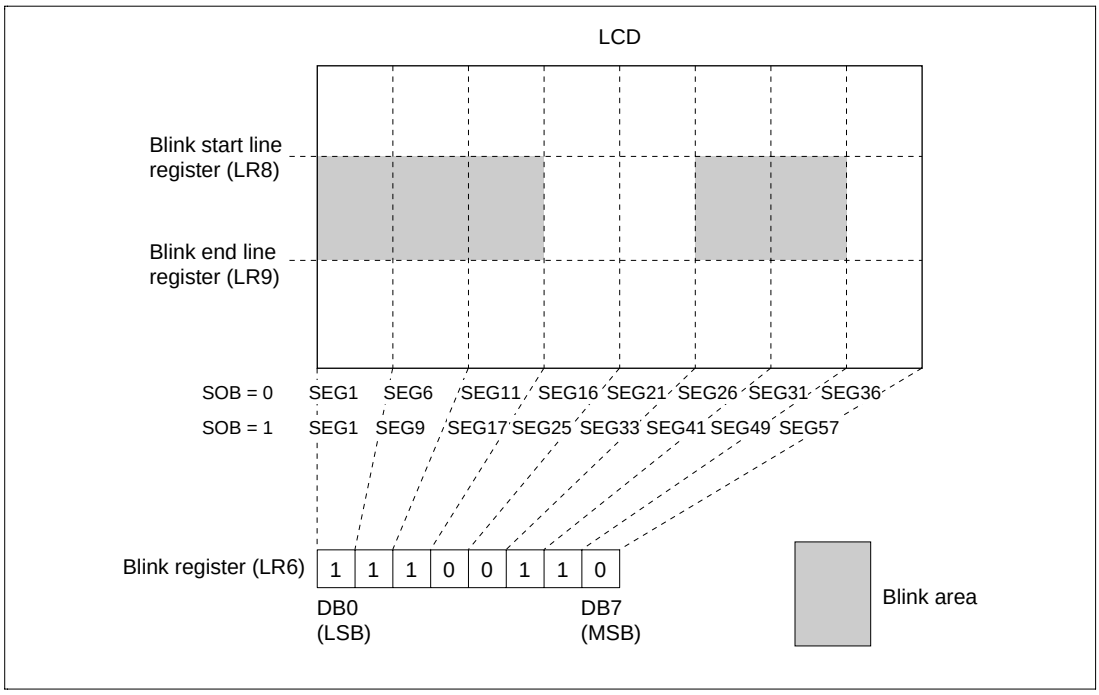


Figure 13.10 Blink Register (LR6) and Blink Locations

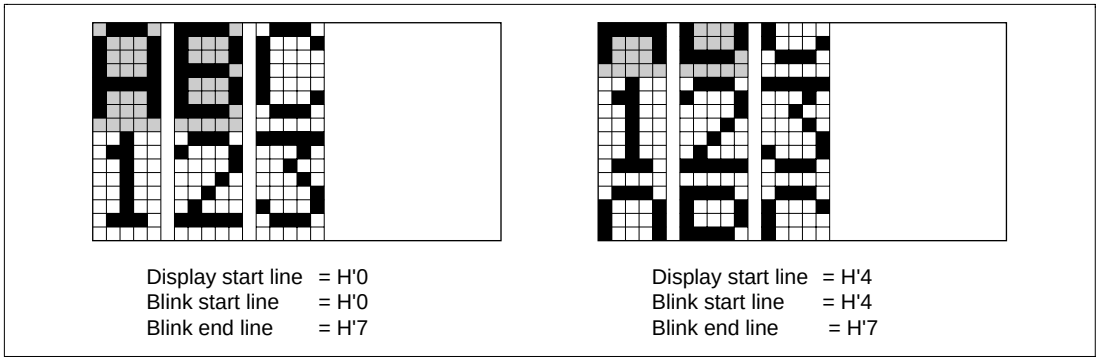


Figure 13.11 Blinking during Display Scrolling (SOB = 0, 1/16 Duty)

13.3.9 Module Standby Mode

The LCD controller has a module standby function that enables low power consumption to be achieved. In module standby mode, the built-in step-up circuit and op-amps are halted, and segment and common outputs go to the V_{SS} (display-off state) level. Display RAM and internal register data is retained, except for the DISP and OPON bits in control register 2 (LR1). The control registers can still be accessed in the module standby state. Figure 13.12 shows the procedures for initiating and clearing module standby mode. The initiation and clearing procedures must be followed exactly in order to protect the display memory contents.

When the CPU is placed in standby mode, set the LSBY bit in control register 1 (LR0) to 1 before executing the standby instruction. After clearing standby mode, follow the module standby clearing procedure to start display.

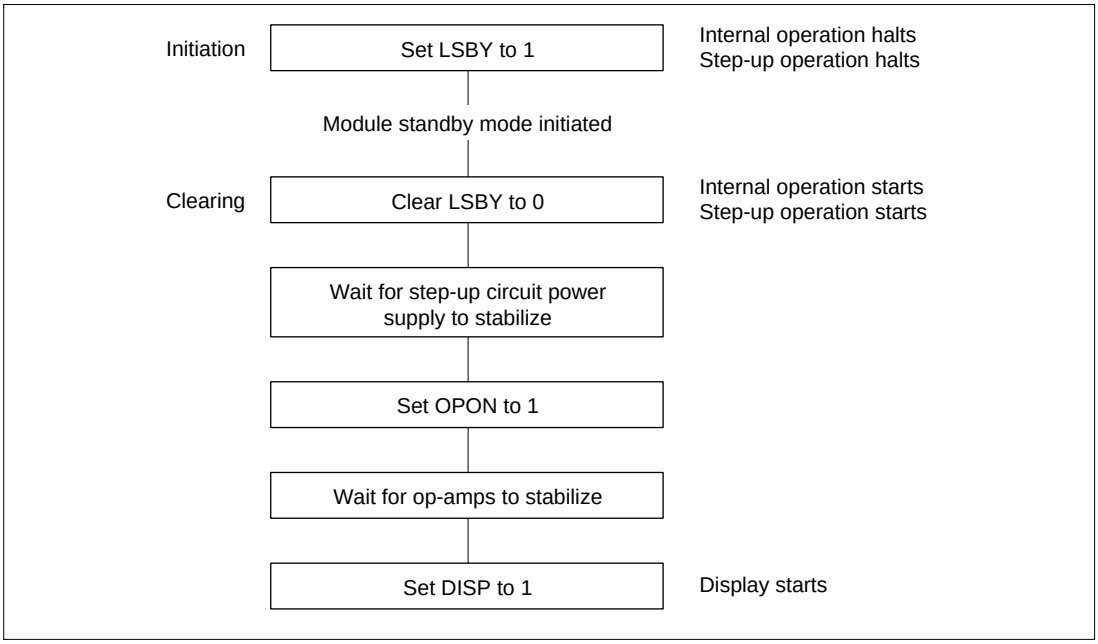


Figure 13.12 Module Standby Mode and Standby Mode Initiation and Clearing Procedures

13.3.10 Power-On and Power-Off Procedures

As the LCD controller incorporates a complete power supply circuit, the procedures shown in figure 13.13 must be followed when powering on and off. Failure to follow these procedures may result in an abnormal display.

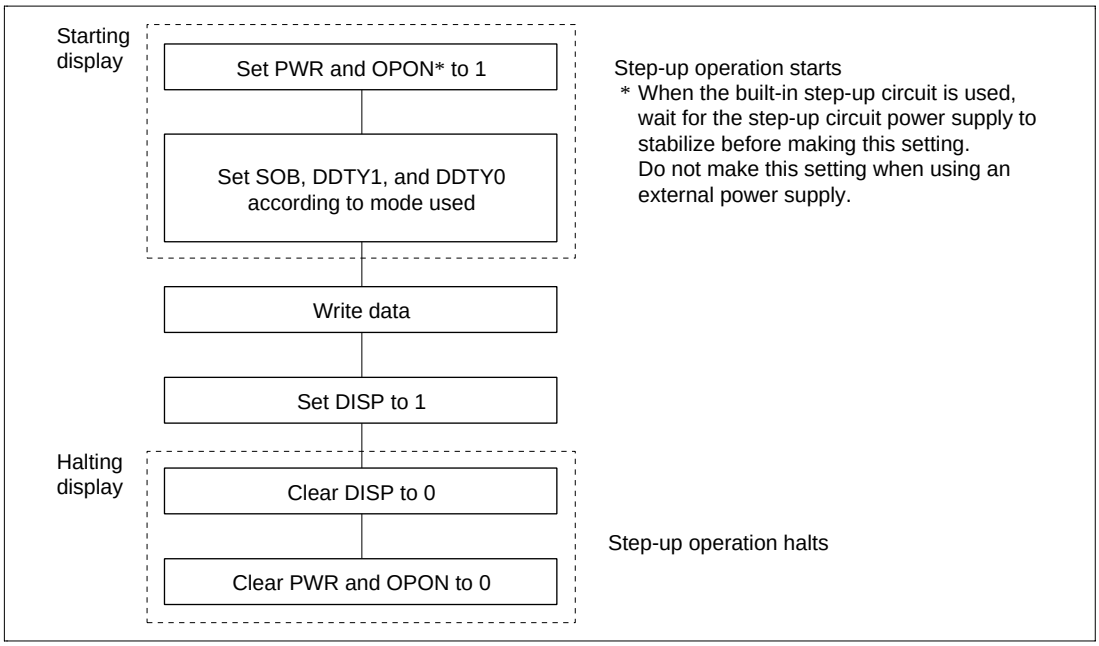


Figure 13.13 Power-On and Power-Off Procedures

13.3.11 Power Supply Circuit

The LCD controller has a built-in 2X or 3X step-up circuit for LCD drive. In standby mode, the power supply circuit is automatically turned off after a maximum of two subclock cycles, and the power consumption of the step-up circuit falls to zero. The power supply circuit can be turned on and off by a command, and an external power supply circuit should be used if the current capacity of the built-in step-up circuit is insufficient.

Step-Up Circuit

By inputting the reference voltage to V_{ci} ($V_{ci} \leq V_{CC}$), connecting a capacitor between V_{SS} and VLOUT, C1+ and C1–, and C2+ and C2–, and setting the PWR bit in control register 1 (LR0) to 1, the potential between V_{ci} and V_{CC} is stepped up by a factor of 2 or 3. See figures 13.17 (1) and (2) for the method of connecting the capacitors. As the subclock is used for voltage step-up, step-up will not be performed unless the subclock is supplied. Since V_{ci} is also used for the step-up circuit power supply, an adequate current must be assured.

Step-up cannot be performed below V_{CC}. Apply a V_{ci} voltage that gives a VLOUT level between V_{CC} and 7.0 V.

If the step-up circuit is not used, connect V_{ci} to V_{CC}.

LCD Drive Level Power Supply

Six power supply levels—V1, V2, V3, V3, V4, V5, and V_{SS}—are necessary for LCD drive. The V1 to V_{SS} power supplies are normally generated by means of resistive division. The power supply circuit includes a voltage follower op-amp for each voltage level generated by resistive division.

When 1/4 bias is used for LCD display, the V3 and V4 pins should be shorted; when 1/5 bias is used, the V3 and V4 pins should be left open. The V34 pin is an internal resistance test pin, and should always be shorted to the V3 pin externally.

Contrast Control

The LCD controller provides for the following two methods of contrast control.

- Using built-in contrast control circuit

  The LCD controller includes a programmable contrast control circuit. The LCD power supply voltage can be adjusted on the basis of a given step-up circuit voltage by making a selection in the contrast control register (LRA).

- By changing step-up circuit reference voltage V_{ci}

  The step-up circuit voltage level can be varied by changing step-up circuit reference voltage V_{ci}.

External Power Supply

- When an external power supply is input to V_{LCD}

  V1 to V5 can be generated by inputting an external power supply to V_{LCD}, and using the built-in op-amps by setting the OPON bit in control register 2 (LR1) to 1. The V_{LCD} input level must be between V_{CC} and 7.0 V.

- When external power supply is input directly to V1 to V5

  A power supply can be applied directly to V1, V2, V3, V4, and V5 from an external source by clearing the PWR bit in control register 1 (LR0) and the OPON bit in control register 2 (LR1) to 0, to halt the built-in step-up circuit and cut the built-in op-amp power supply. The same potential as V1 should be input to V_{LCD}. Apply a voltage not exceeding V_{LCD} to V2 through V5. The input level of V_{LCD} and V1 must be between V_{CC} and 7.0 V.

In either case, inputting a voltage exceeding the maximum rated voltage may adversely affect the reliability of the chip.

13.3.12 LCD Drive Power Supply Voltages

There are six LCD drive power supply voltage values—V1 to V5, and V_{SS}. V1 is the highest voltage, and V_{SS} the lowest. As shown in figure 13.14, the common waveforms are formed from a combination of V1, V2, V5, and V_{SS}, while the segment waveforms are formed from a combination of V1, V3, V4, and V_{SS}. V1 and V_{SS} are shared by both common and segment waveforms, but the intermediate voltages are different.

In figure 13.14, the waveforms of outputs SEG1 to SEG40 differ according to the display data. In this example, LCD panel lines for which COM1 is connected are illuminated, and all other dots are not illuminated.

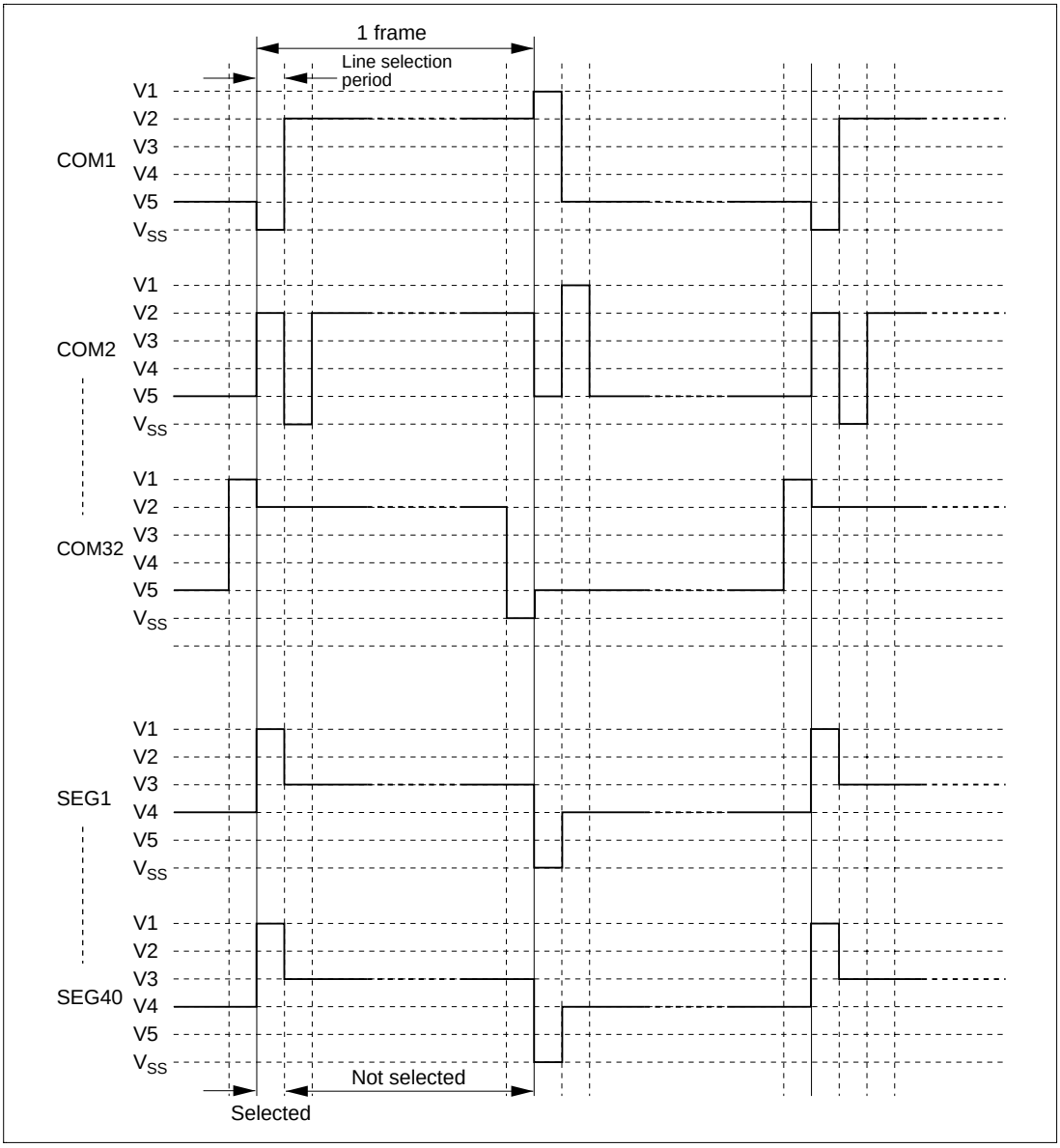


Figure 13.14 LCD Drive Power Supply Waveforms (1/32 Duty)

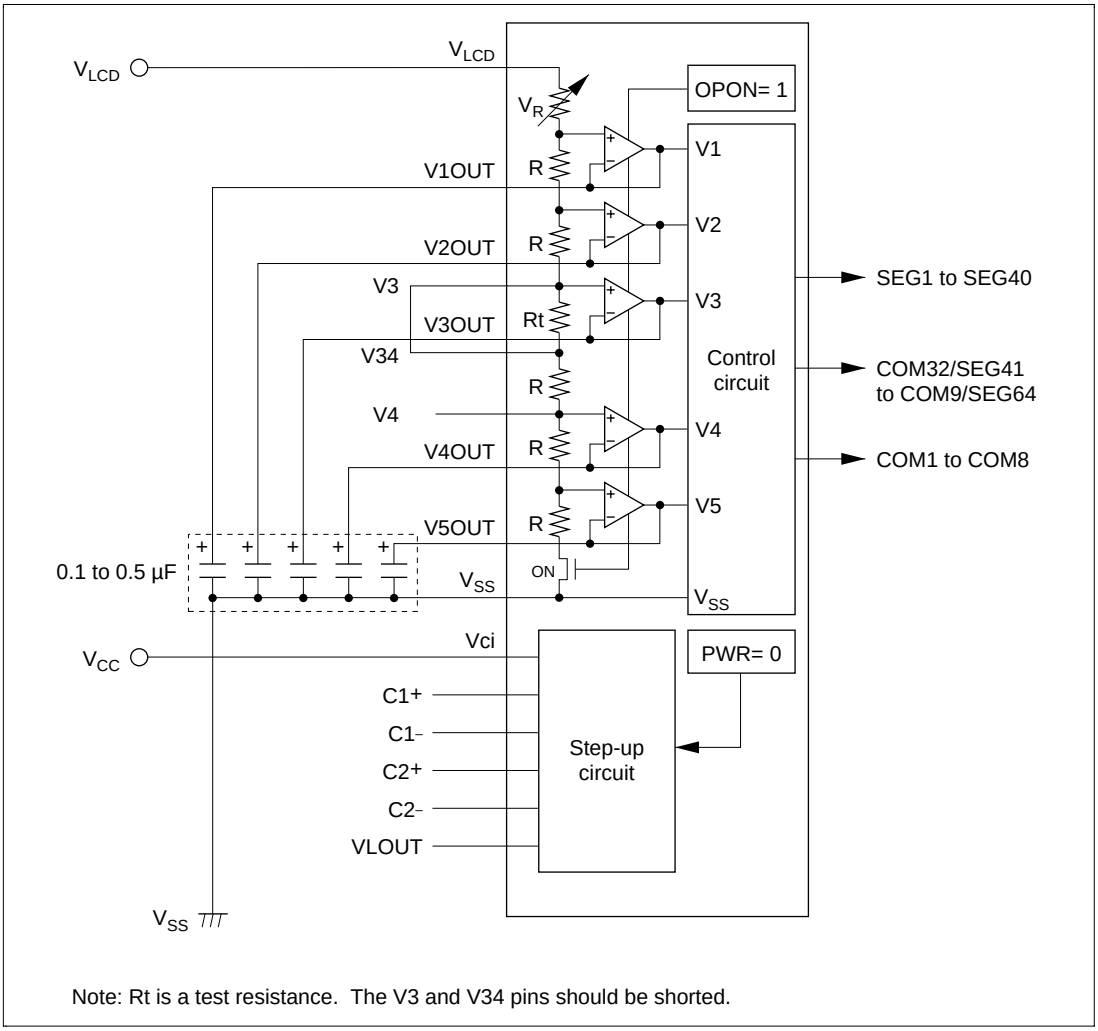
13.3.13 LCD Voltage Generation Circuit

When Using External Power Supply and Built-In Op-Amps

When the built-in step-up circuit is not used, and the LCD drive voltages are supplied directly from an external power supply, connections should be made as shown in figure 13.15. The V_{LCD} input level must be between V_{CC} and 7.0 V.

The LCD controller includes bleeder resistances that generate levels V1 to V5, and voltage follower op-amp circuits. Set the OPON bit in control register 2 (LR1) to 1. Contrast can be controlled by software, using the contrast control register (LRA).

If the capacitance of the LCD panel to be driven is large, capacitors of around 0.1 to 0.5 µF should be inserted between the V1OUT to V5OUT built-in op-amp outputs and V_{SS} to provide stabilization. In order for the op-amps to operate normally, the contrast control register (LRA) should be set so that the potential difference between V_{LCD} and V1, and between V5 and V_{SS}, is at least 0.4 V.



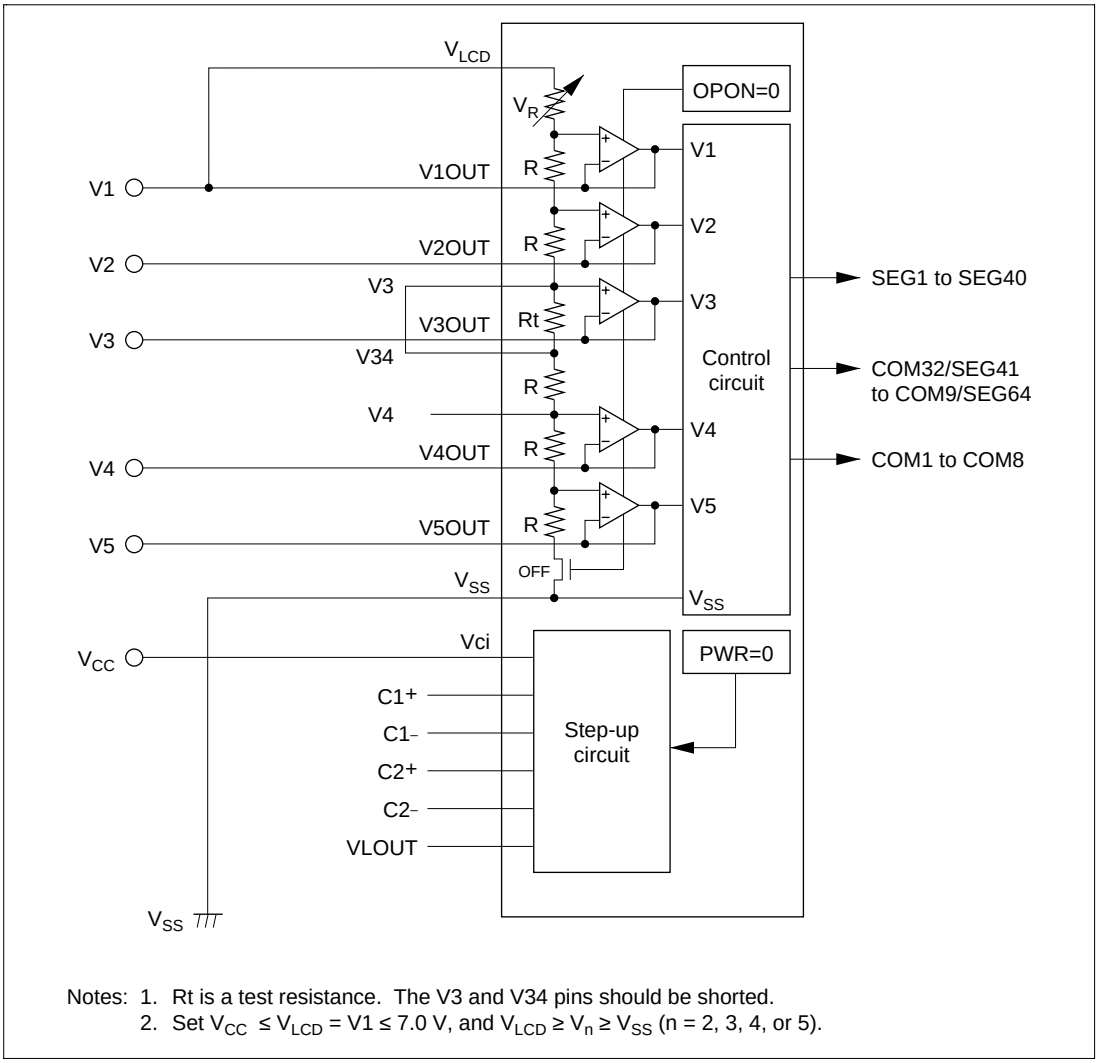
Note: Rt is a test resistance. The V3 and V34 pins should be shorted.

Figure 13.15 Example of Connections when Using Built-In Op-Amps and External LCD Power Supply (1/5 Bias)

When Using External Power Supply but Not Using Built-In Op-Amps

When the built-in step-up circuit and op-amps are not used, and the LCD drive voltages are supplied directly from an external power supply, connections should be made as shown in figure 13.16.

As the built-in step-up circuit and op-amps are not used, the OPON bit in control register 2 (LR1) should be cleared to 0.



Notes: 1. Rt is a test resistance. The V3 and V34 pins should be shorted.
2. Set $V_{CC} \leq V_{LCD} = V1 \leq 7.0$ V, and $V_{LCD} \geq V_n \geq V_{SS}$ (n = 2, 3, 4, or 5).

Figure 13.16 Example of Connections when Not Using Built-In Op-Amps and Using External LCD Power Supply (1/5 Bias)

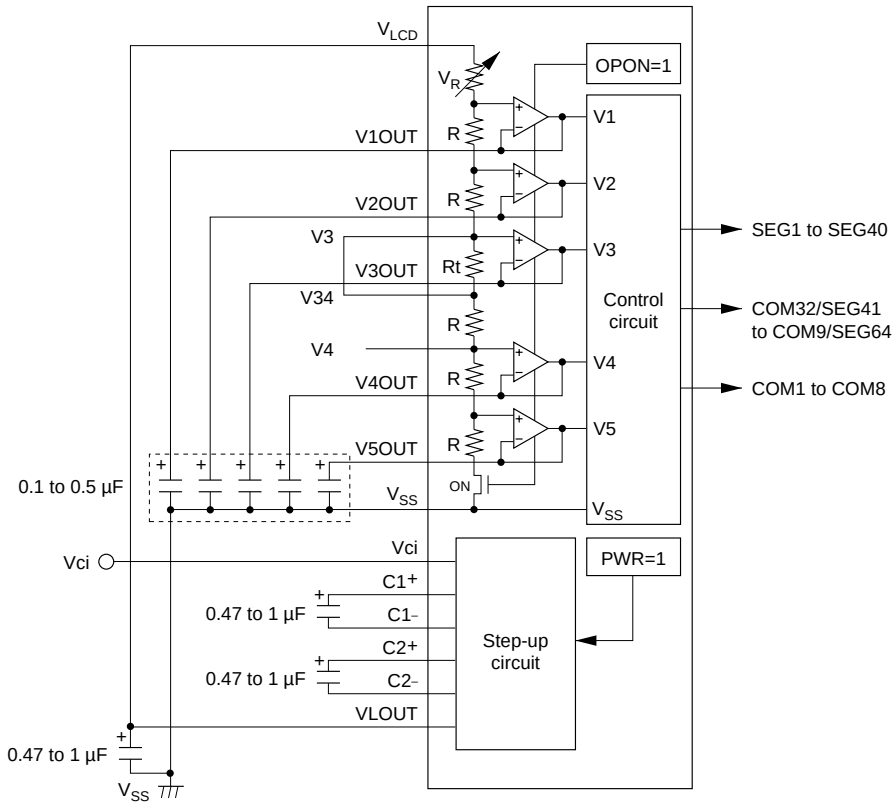
When Using Built-In Step-Up Circuit and Op-Amps

When the built-in step-up circuit is used, connections should be made as shown in figure 13.17 (1) (3X step-up) or figure 13.17 (2) (2X step-up).

The LCD controller includes bleeder resistances that generate levels V1 to V5, and voltage follower op-amp circuits. When the built-in op-amps are used, the OPON bit in control register 2 (LR1) should be set to 1. Contrast can be controlled by software, using the contrast control register (LRA).

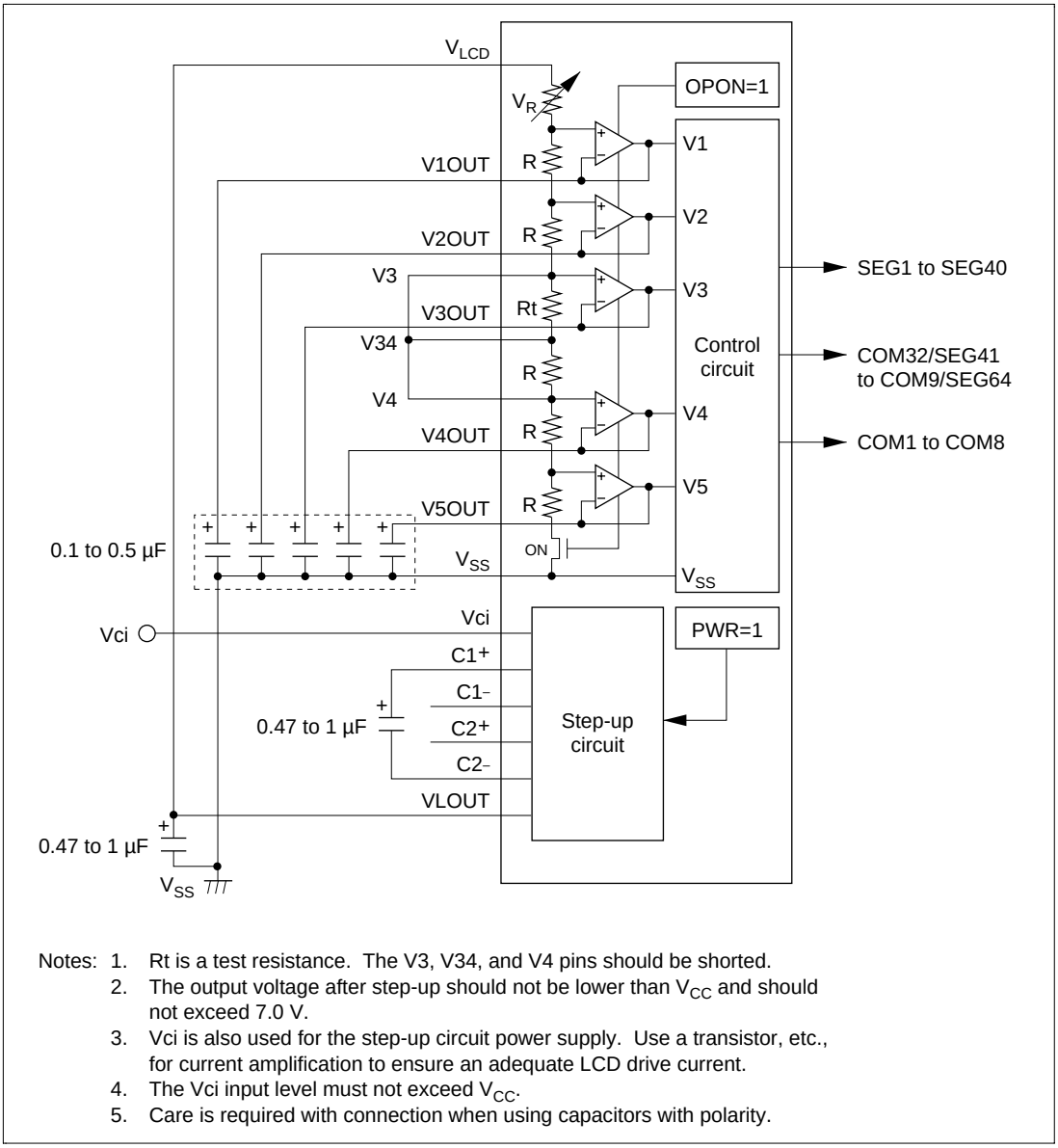
If the capacitance of the LCD panel to be driven is large, capacitors of around 0.1 to 0.5 µF should be inserted between the V1OUT to V5OUT built-in op-amp outputs and V_{SS} to provide stabilization. In order for the op-amps to operate normally, the contrast control register (LRA) should be set so that the potential difference between V_{LCD} and V1, and between V5 and V_{SS}, is at least 0.4 V.

Since the V_{ci} pin is also used for the step-up circuit power supply, ensure that an adequate current can be supplied when carrying out reference voltage adjustment. The V_{ci} input level must not exceed V_{CC}.



Notes: 1. Rt is a test resistance. The V3 and V34 pins should be shorted.
2. The output voltage after step-up should not be lower than V_{CC} and should not exceed 7.0 V. With 3X step-up, in particular, do not input a voltage of 2.3 V or above as the reference voltage (Vci).
3. Vci is also used for the step-up circuit power supply. Use a transistor, etc., for current amplification to ensure an adequate LCD drive current.
4. The Vci input level must not exceed V_{CC}.
5. Care is required with connection when using capacitors with polarity.

Figure 13.17 Example of Connections when Using Built-In Step-Up Circuit (1) (3X Step-Up, 1/5 Bias)



Notes: 1. Rt is a test resistance. The V3, V34, and V4 pins should be shorted.
2. The output voltage after step-up should not be lower than V_{CC} and should not exceed 7.0 V.
3. Vci is also used for the step-up circuit power supply. Use a transistor, etc., for current amplification to ensure an adequate LCD drive current.
4. The Vci input level must not exceed V_{CC}.
5. Care is required with connection when using capacitors with polarity.

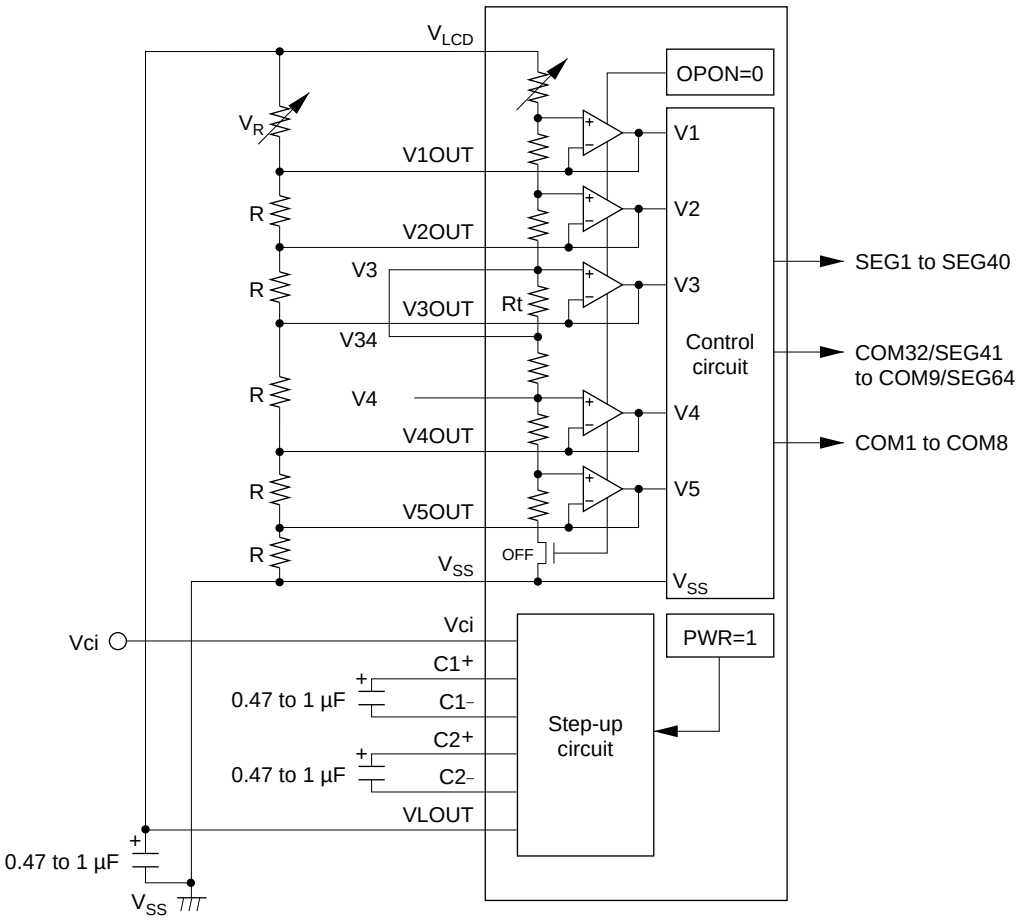
Figure 13.17 Example of Connections when Using Built-In Step-Up Circuit (2) (2X Step-Up, 1/4 Bias)

When Using Built-In Step-Up Circuit and Bleeder Resistances

If the drive capability of the built-in op-amps is insufficient for the size of the LCD panel, the V1 to V5 levels can be supplied from external bleeder resistances. In this case, clear the OPON bit in control register 2 (LR1) to 0 to turn the op-amps off. The built-in contrast control circuit cannot be used, so contrast control must be handled by an external circuit.

A 1/4 or 1/5 bias value can be set, according to the method of connecting the external bleeder resistances. Figure 13.18 shows an example of the connections for 1/5 bias drive.

The 2X or 3X step-up circuit can be used.



Notes:
1. Rt is a test resistance. The V3 and V34 pins should be shorted.
2. A value of around 5 kΩ to 25 kΩ is recommended for external power supply division resistance R.
3. For contrast control, either insert a variable resistance between V_{LCD} and V1, or adjust step-up circuit reference voltage Vci. The built-in contrast control circuit cannot be used.
4. The output voltage after step-up should not be lower than V_{CC} and should not exceed 7.0 V. With 3X step-up, in particular, do not input a voltage of 2.3 V or above as the reference voltage (Vci).
5. Vci is also used for the step-up circuit power supply. Use a transistor, etc., for current amplification to ensure an adequate LCD drive current.
6. The Vci input level must not exceed V_{CC}.
7. Care is required with connection when using capacitors with polarity.
8. Set V1OUT ≤ V_{LCD}.

Figure 13.18 Example of Connections when Using 3X Step-Up Circuit and External Bleeder Resistances (1/5 Bias)

13.3.14 Contrast Control Circuit

Contrast control can be performed by software (electronic control function) by controlling the LCD drive voltage (the potential difference between V_{LCD} and V1) by means of the contrast control register (LRA). Variable resistance value V_R can be adjusted within the range of 0.1 R to 1.6 R, where R is the value of the basic dividing bleeder resistance between V_{LCD} and V1. The contrast control settings by bits CCR3 to CCR0 in the contrast control register (LRA) are shown in table 13.7.

To ensure stable operation of the voltage follower op-amp circuits that output levels V1 to V5, the contrast control register (LRA) should be set so that the potential difference between V_{LCD} and V1, and between V5 and V_{SS}, is at least 0.4 V. The contrast control ranges are shown in table 13.8.

If contrast control cannot be adequately performed by means of on-chip resistance V_R, control can be performed by inserting a resistance between VLOUT and V_{LCD}.

Table 13.7 Contrast Control Settings

Contrast Control Register (LRA)				Variable Resistance	$V1—V_{SS}$ Potential	
CCR3	CCR2	CCR1	CCR0	Value (V_R)	Difference	Display Color
0	0	0	0	1.6R	Small	Light
			1	1.5R		
		1	0	1.4R		
			1	1.3R		
	1	0	0	1.2R		
			1	1.1R		
		1	0	1.0R		
			1	0.9R		
1	0	0	0	0.8R		
			1	0.7R		
		1	0	0.6R		
			1	0.5R		
	1	0	0	0.4R		
			1	0.3R		
		1	0	0.2R		
			1	0.1R	Large	Dark

Table 13.8 Contrast Control Ranges

Bias	LCD Drive Voltage: V_{DR}	Contrast Control Range	
1/5 bias drive	$\frac{5\times R}{5\times R+V_R}\times(V_{LCD}-V_{SS})$	• LCD drive voltage adjustment range:	$0.758\times(V_{LCD}-V_{SS}) \le V_{DR} \le 0.980\times(V_{LCD}-V_{SS})$
		• V5—V_{SS} potential difference limit:	$\frac{R}{5\times R+V_R}\times(V_{LCD}-V_{SS}) \ge 0.4$ [V]
		• V_{LCD}—V1 potential difference limit:	$\frac{V_R}{5\times R+V_R}\times(V_{LCD}-V_{SS}) \ge 0.4$ [V]
1/4 bias drive	$\frac{4\times R}{4\times R+V_R}\times(V_{LCD}-V_{SS})$	• LCD drive voltage adjustment range:	$0.714\times(V_{LCD}-V_{SS}) \le V_{DR} \le 0.976\times(V_{LCD}-V_{SS})$
		• V5—V_{SS} potential difference limit:	$\frac{R}{4\times R+V_R}\times(V_{LCD}-V_{SS}) \ge 0.4$ [V]
		• V_{LCD}—V1 potential difference limit:	$\frac{V_R}{4\times R+V_R}\times(V_{LCD}-V_{SS}) \ge 0.4$ [V]

13.3.15 LCD Drive Bias Selection Circuit

The ideal bias value that gives the best contrast is calculated using the equation shown below. If drive is performed at a bias value lower than the optimum, contrast will deteriorate, but the LCD drive voltage (the potential difference between V1 and V_{SS}) can be kept low. If the LCD drive voltage is inadequate even with a low V_{ci} voltage and use of the 3X step-up circuit, or if the output voltage falls and the LCD display becomes faint as batteries wear out, for instance, the display can be made clearer by decreasing the LCD drive bias.

$$\text{Optimum bias value for 1/N duty drive} = \frac{1}{\sqrt{N+1}}$$

Notes: 1. When using 1/5 bias, leave the V3 and V4 pins open.
2. When using 1/4 bias, short the V3 and V4 pins.
3. The V3 and V34 pins must always be shorted.

Section 14 Dot Matrix LCD Controller (H8/3854 Series)

14.1 Overview

The LCD controller has built-in display RAM, and performs dot matrix LCD display. One bit of display RAM data corresponds to illumination or non-illumination of one dot on the LCD panel, making possible displays with an extremely high degree of freedom.

The LCD controller incorporates all the functions required for LCD display, allowing a dot matrix display of up to 40 × 16 dots.

I/O ports are used for the interface with the CPU, offering excellent software heritability when using a combination of MPU and LCD driver.

This module operates on the subclock, making it ideal for use in small portable devices.

14.1.1 Features

- Built-in bit-mapped display RAM (640 bits)
  Maximum of 640 display bits (selectable from 40 × 16 bits or 40 × 8 bits)
- Choice of 1/8 or 1/16 duty
- Low power consumption enabling extended drive on battery power
  Subclock operation
  Module standby
- Comprehensive display control functions
  Display data read/write, display on/off control, read-modify-write
- CPU interface
  I/O port interface
- Built-in LCD power supply bleeder resistance circuit

14.1.2 Block Diagram

Figure 14.1 shows a block diagram of the LCD controller.

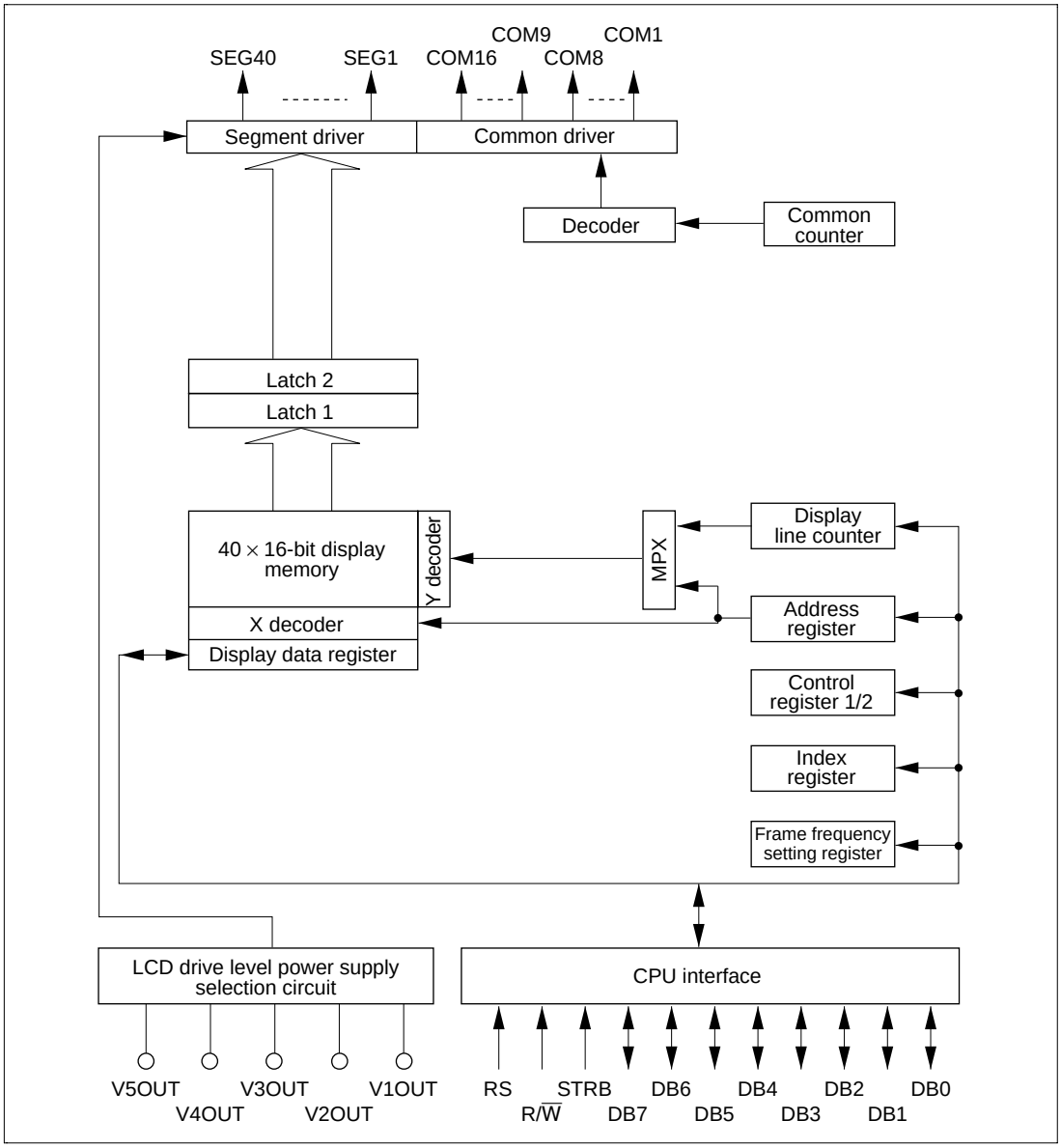


Figure 14.1 Block Diagram of LCD Controller

14.1.3 Pin Configuration

Table 14.1 shows the pins assigned to the LCD controller.

Table 14.1 Pin Configuration

Pin Name	Abbrev.	I/O	Function
Common output pins	COM1 to COM16	Output	LCD common drive pins
Segment output pins	SEG1 to SEG40	Output	LCD segment drive pins
LCD drive power supply level	V1OUT to V5OUT	I/O	LCD drive power supply level input/output pins

14.1.4 Register Configuration

The LCD controller has one index register and five control registers, all of which are accessed via an I/O port interface. Except for the display data register (LR4), these registers cannot be read. The LCD controller register configuration is shown in table 14.2.

Table 14.2 Register Configuration

				Index Register		
Name	**Abbrev.**	**R/W**	**RS**	**IR2**	**IR1**	**IR0**
Index register	IR	W	0	—	—	—
Control register 1	LR0	W	1	0	0	0
Control register 2	LR1	W		0	0	1
Address register	LR2	W		0	1	0
Frame frequency setting register	LR3	W		0	1	1
Display data register	LR4	R/W		1	0	0

14.2 Register Descriptions

14.2.1 Index Register (IR)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	IR2	IR1	IR0
Initial value	—	—	—	—	—	0	0	0
Read/Write	—	—	—	—	—	W	W	W

IR is an 8-bit write-only register that selects one of the LCD controller's five control registers. IR is selected when RS is 0.

Upon reset, IR is initialized to H'00.

Bits 7 to 3—Reserved Bits: Bits 7 to 3 are reserved; they should always be cleared to 0.

Bits 2 to 0—Index Register (IR2 to IR0): Bits 2 to 0 are used to select one of the LCD controller's five control registers. The correspondence between the settings of IR2 to IR0 and the selected registers is shown in table 14.2. Other settings are invalid.

14.2.2 Control Register 1 (LR0)

Bit	7	6	5	4	3	2	1	0
	—	—	LSBY	—	—	—	DDTY1	—
Initial value	—	—	0	—	—	—	0	—
Read/Write	—	—	W	—	—	—	W	—

LR0 is an 8-bit write-only register that performs LCD module standby mode setting and drive duty selection.

Upon reset, LR0 is initialized to H'00.

Bits 7 and 6—Reserved Bits: Bits 7 and 6 are reserved; they should always be cleared to 0.

Bit 5—Module Standby (LSBY): Bit 5 is the module standby setting bit. When LSBY is set to 1, the LCD controller enters standby mode. At this time, bits DISP, LPS1, and LPS0 in LR1 are reset.

Bit 5

LSBY	Description	
0	LCD controller operates normally	(initial value)
1	Power supply to built-in bleeder resistances halts, display is turned off, and LCD controller enters standby mode	

Bits 4 to 2—Reserved Bits: Bits 4 to 2 are reserved; they should always be cleared to 0.

Bit 1—Display Duty Select (DDTY1): Bit 1 selects a display duty of 1/16 or 1/8.

Bit 1

DDTY1	Description	
0	1/16 duty selected	(initial value)
1	1/8 duty selected Y address H'8 to H'F display data is invalid	

Bit 0—Reserved Bit: Bit 0 is reserved; it should always be cleared to 0.

14.2.3 Control Register 2 (LR1)

Bit	7	6	5	4	3	2	1	0
	—	DISP	LPS1	LPS0	RMW	—	INC	—
Initial value	—	0	0	0	0	—	0	—
Read/Write	—	W	W	W	W	—	W	—

LR1 is an 8-bit write-only register that selects operation or halting of LCD display and supply or halting of current to the built-in bleeder resistances, performs read-modify-write mode setting, and selects the address to be incremented in the display memory.

Upon reset, LR1 is initialized to H'00.

Bit 7—Reserved Bit: Bit 7 is reserved; it should always be cleared to 0.

Bit 6—LCD Operation Setting (DISP): Bit 6 selects operation or halting of the LCD display. When the LSBY bit in LR0 is set to 1, DISP is cleared.

Bit 6

DISP	Description	
0	LCD is turned off. All LCD outputs go to the V_{SS} level	(initial value)
1	LCD is turned on	

Bits 5 and 4—LCD Power Supply Setting (LPS1, LPS0): Bits 5 and 4 specify use or non-use of the internal power supply as the LCD drive power supply, and of the built-in LCD power supply bleeder resistances. When LPS1 is set to 1, the internal power supply is connected to the bleeder resistances. When LPS0 is set to 1, a power supply divided by the built-in bleeder resistances is supplied. When the LCD drive power supply level is applied to V1OUT through V5OUT from an external source, LPS1 and LPS0 must be cleared to 0.

When the LSBY bit in LR0 is set, LPS1 and LPS0 are cleared.

Bit 5

LPS1	Description	
0	Power supply to V1OUT is halted	(initial value)
1	Power supply voltage is supplied to V1OUT	

Bit 4

LPS0	Description	
0	Built-in bleeder resistances not used	(initial value)
1	Built-in bleeder resistances used	

Bit 3—Read-Modify-Write Setting (RMW): Bit 3 selects whether display memory X or Y address incrementing is carried out after a write/read access, or only after a write access (read-modify-write mode).

Bit 3

RMW	Description
0	Address is incremented after write/read access to display memory (initial value)
1	Read-modify-write mode is set In this mode, address is incremented only after write access to display memory

Bit 2—Reserved Bit: Bit 2 is reserved; it should always be cleared to 0.

Bit 1—Increment Address Select (INC): Bit 1 selects either the X address or the Y address as the address to be incremented after the display memory access specified by the RMW bit. The selected address is cleared after a display memory access with the maximum value for the valid display data area; in this case the other address is incremented.

Bit 1

INC	Description
0	Incrementing of display memory Y address has priority; X address is incremented after Y address overflow (initial value)
1	Incrementing of display memory X address has priority; Y address is incremented after X address overflow

Bit 0—Reserved Bit: Bit 0 is reserved; it should always be cleared to 0.

14.2.4 Address Register (LR2)

Bit	7	6	5	4	3	2	1	0
	XA2	XA1	XA0	—	YA3	YA2	YA1	YA0
Initial value	0	0	0	—	0	0	0	0
Read/Write	W	W	W	—	W	W	W	W

LR2 is an 8-bit write-only register that sets the display memory X- and Y-direction addresses accessed by the CPU.

Upon reset, LR2 is initialized to H'00.

Bits 7 to 5—X Address Setting (XA2 to XA0): Bits 7 to 5 set the display memory X-direction address. A value from H'0 to H'4 can be set. Do not perform access in the range H'5 to H'7.

When the INC bit in LR1 is set to 1, the address is automatically incremented after the access specified by the RMW bit in LR1, and is cleared after an H'4 access. When INC is 0 and YA3 to YA0 represent the maximum value for the valid display data area, the address is incremented after the access specified by RMW.

Bit 4—Reserved Bit: Bit 4 is reserved; it should always be cleared to 0.

Bits 3 to 0—Y Address Setting (YA3 to YA0): Bits 3 to 0 set the display memory Y-direction address. A value from H'0 to H'F can be set, but display data from H'8 to H'F is invalid with 1/8 duty.

When the INC bit in LR1 is cleared to 0, the address is automatically incremented after the access specified by the RMW bit in LR1, and is cleared after an access with the maximum value for the valid display data area. When INC is 1 and the value in XA2 to XA0 is H'4, the address is incremented after the access specified by RMW.

14.2.5 Frame Frequency Setting Register (LR3)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	FS2	FS1	FS0
Initial value	—	—	—	—	—	0	0	0
Read/Write	—	—	—	—	—	W	W	W

LR3 is an 8-bit write-only register that sets the frame frequency.

Upon reset, LR3 is initialized to H'00.

Bits 7 to 3—Reserved Bits: Bits 7 to 3 are reserved; they should always be cleared to 0.

Bits 2 to 0—Frame Frequency Setting (FS2 to FS0): Bits 2 to 0 control the subclock division ratio and set the LCD frame frequency. The relationship between the LCD frame frequency f_F (Hz), the subclock frequency f_W (Hz), the division ratio r, and the LCD duty 1/N is as follows:

$$f_F = \frac{f_W}{r \times N}$$

Set a division ratio suitable for the characteristics of the LCD panel used. The correspondence between register settings, division ratios, and frame frequencies at each display duty is shown in table 14.3.

Table 14.3 Register Settings, Division Ratios, and Frame Frequencies at Each Display Duty

				Display Duty 1/N			
				1/8		1/16	
				Subclock Frequency f_w (kHz)			
				32.768	38.4	32.768	38.4
FS2	FS1	FS0	Division ratio r	Frame Frequency f_F (Hz)			
0	0	0	2	2048.0	2400.0	1024.0	1200.0
		1	4	1024.0	1200.0	512.0	600.0
	1	0	8	512.0	600.0	256.0	300.0
		1	16	256.0	300.0	128.0	150.0
1	0	0	32	128.0	150.0	64.0	75.0
		1	64	64.0	75.0	32.0	37.5
	1	0	128	32.0	37.5	16.0	18.8
		1	Setting prohibited	Setting prohibited	Setting prohibited	Setting prohibited	Setting prohibited

14.2.6 Display Data Register (LR4)

Bit	7	6	5	4	3	2	1	0
	D7	D6	D5	D4	D3	D2	D1	D0
Initial value	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

LR4 is an 8-bit read/write register used to perform read/write access to the display memory specified by XA2 to XA0 and YA3 to YA0 in LR2.

In a write to display memory, the write is performed directly to the display memory via this register. In a read, the data is temporarily latched into this register before being output to the bus.

After a reset, the display memory and LR4 contents are undefined.

14.3 Operation

14.3.1 System Overview

The LCD controller operates at 1/16 or 1/8 duty. The display size is a maximum of 40×16 dots. As the LCD controller operates on the subclock to perform display control, the time, etc., can be constantly displayed. Also, since data in the display RAM is retained even in module standby mode, low power consumption can be achieved without affecting the display.

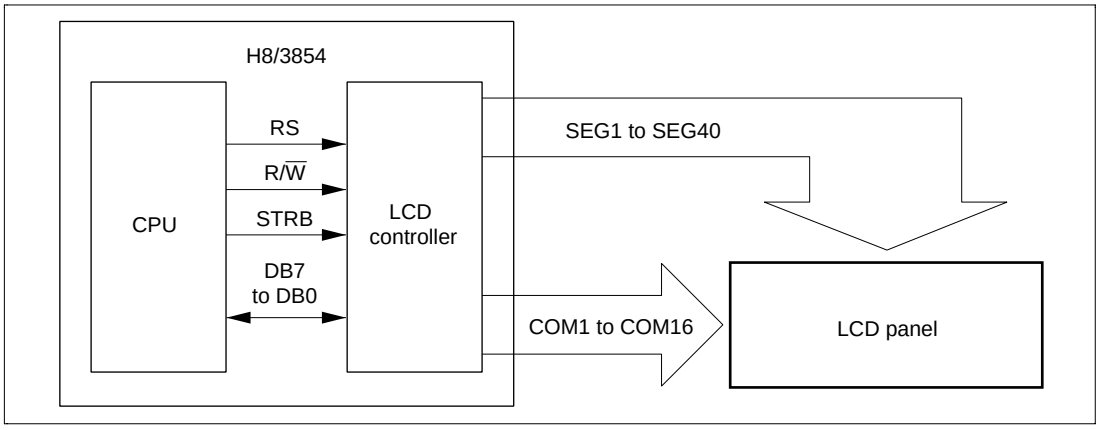


Figure 14.2 System Block Diagram

14.3.2 CPU Interface

The LCD controller's registers are not included in the memory map shown in figure 2.16 (b). They are controlled from the CPU by means of chip-internal LCD pins DB7 to DB0, RS, R/$\overline{W}$, and STRB, via chip-internal I/O ports 9 and A. The pin configuration is shown in table 14.4, and an example of the timing for access to registers in the LCD controller is shown in figure 14.3. For information on port 9 and port A, see the descriptions in section 8, I/O Ports.

Table 14.4 Pin Configuration

Pin Name	Abbrev.	I/O	Function
Data bus pins	DB7 to DB0	I/O	When R/$\overline{W}$ = 0, these pins input data to be written to a register; when R/$\overline{W}$ = 1, they output data read from a register
Register selector pin	RS	Input	When R/S = 0, the index register is selected; when RS = 1, a control register is selected
Read/write select pin	R/$\overline{W}$	Input	When R/$\overline{W}$ = 0, write access is selected; when R/$\overline{W}$ = 1, read access is selected
Strobe pin	STRB	Input	At the fall of STRB, read or write access, as selected by R/$\overline{W}$, is performed on the register selected by RS

Writing to Index Register

When RS and R/$\overline{W}$ are both cleared to 0, data DB7 to DB0 is written to the index register (IR) at the falling edge of STRB. Do not change RS or R/$\overline{W}$ at the fall of STRB.

Reading and Writing to Control Registers

To access a control register, data indicating the number of the register to be accessed must be written to the index register (IR) before making the access. The register number data to be written to IR is shown in table 14.2. As the register number written to IR is retained until IR is written to again, if the same control register is accessed repeatedly, it is not necessary to write to IR each time.

In a write to a control register, when RS has been set to 1 and R/$\overline{W}$ cleared to 0, data DB7 to DB0 is written to the control register specified by the index register (IR) at the falling edge of STRB.

Except for the display data register (LR4), control registers cannot be read. In a read of LR4, when the LR4 register number is written to the index register (IR), and RS and R/$\overline{W}$ are both set to 1, DB7 to DB0 are set to output mode, and the display memory data at the address specified by the address register (LR2) is output from DB7 to DB0 at the falling edge of STRB. If a read is also performed in the next cycle, the data output is held until the next fall of STRB, but if a write is performed in the next cycle, DB7 to DB0 are set to input mode from the point at which R/$\overline{W}$ is cleared to 0, and the output is cleared.

In either case, do not change RS or R/$\overline{W}$ at the fall of STRB.

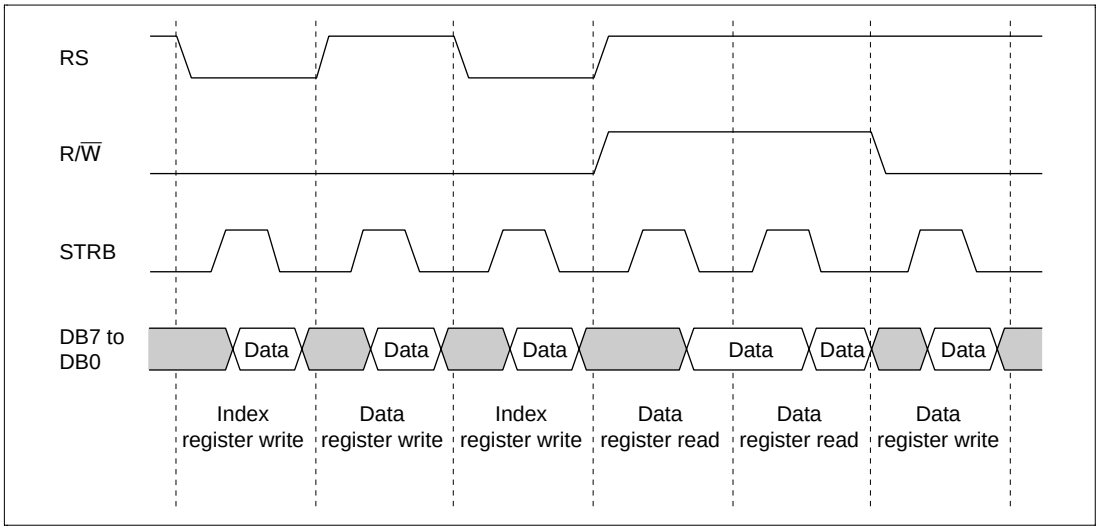


Figure 14.3 Example of Timing Sequence for 8-Bit Data Transfer

Notes on Use of Chip-Internal I/O Ports

For LCD controller interface internal ports 9 and A, port input/output is controlled by means of PCR9 and PCRA in the same way as for ordinary I/O ports, and in output mode, the values set in PDR9 or PDRA are output. Also, LCD controller internal pins RS, R/$\overline{W}$, and STRB are input-only pins, and DB7 to DB0 input/output is controlled by R/$\overline{W}$. Therefore, the following points must be noted.

1. After reset release and standby mode release

   Since the chip's internal I/O ports go to the high-impedance state in a reset and in standby mode, in initialization after reset or standby mode release, H'06 should be set in PDRA, and H'07 in PCRA. This will set port A to output mode. If the PDRA setting were H'00, there would be a possibility of the index register (IR) being written to.

2. Changing register read/write setting

   When an LCD controller register is read (R/$\overline{W}$ = 1), DB7 to DB0 output data from the LCD controller side, and so port 9 must be set to input mode. Therefore, H'00 must be written to PCR9, setting port 9 to input mode, before changing the R/$\overline{W}$ setting from 0 to 1. When writing data to an LCD controller register, first change R/$\overline{W}$ from 1 to 0, then write H'FFF to PCR9, setting port 9 to output mode.

   Examples of display data register (LR4) read/write access when read-modify-write is designated are shown below.

[Set index register to display data register]

- Port A set to output mode, RMW set to 1

```
MOV.W    #H'0100,R1
MOV.W    #H'04FF,R0
MOV.B    R1L,@PDRA ..............  Clear R/W to 0
MOV.B    R0H,@PDR9
MOV.B    R0L,@PCR9 ..............  Output H'04 from port 9
MOV.B    R1H,@PDRA
MOV.B    R1L,@PDRA ..............  Write H'04 to index register
```

[Read display data register]

```
MOV.B    R1L,@PCR9 ..............  Set port 9 to input mode
MOV.W    #H'0706, R2
MOV.B    R2L,@PDRA ..............  Set R/W to 1
MOV.B    R2H,@PDRA
MOV.B    R2L,@PDRA
MOV.B    @PDR9, R0H                Read PDR9 into general register
```

[Write to display data register]

```
MOV.W    #H'0504,R3
MOV.B    R3L,@PDRA ..............  Clear R/W to 0
NOT.B    R0H        ..............  Invert general register data
MOV.B    R0H,@PDR9
MOV.B    R0L,@PCR9 ..............  Set port 9 to output mode
MOV.B    R3H,@PDRA
MOV.B    R3L @PDRA ..............  Write data to display data register
```

14.3.3 LCD Drive Pin Functions

Common/Segment Output

The display duty is set by control register 1 (LR0) bits DDTY1.

- 1/8 duty (DDTY1 = 1)
  Common outputs: COM1 to COM8
  Segment outputs: SEG1 to SEG40

Note: COM9 to COM16 output common signal non-selection waveforms

- 1/16 duty (DDTY1 = 0)
  Common outputs: COM1 to COM16
  Segment outputs: SEG1 to SEG40

Table 14.5 Pin Functions According to Display Duty

	Function	
Pin Name	**1/8 Duty**	**1/16 Duty**
COM1 to COM8	COM1 to COM8	COM1 to COM16
COM9 to COM16	Common signal non-selection waveform	
SEG1 to SEG40	SEG1 to SEG40	SEG1 to SEG40

14.3.4 Display Memory Configuration and Display

The LCD controller includes 40 × 16-bit bit-mapped display memory. As the display memory configuration, an 8-bit × 5 X-direction combination can be selected, while the Y-direction configuration is 16 bits. Display data written from the CPU is stored horizontally with the MSB at the left and the LSB at the right, as shown in figure 14.4. On the display, 1 data corresponds to illumination (black), and 0 data to non-illumination (colorless).

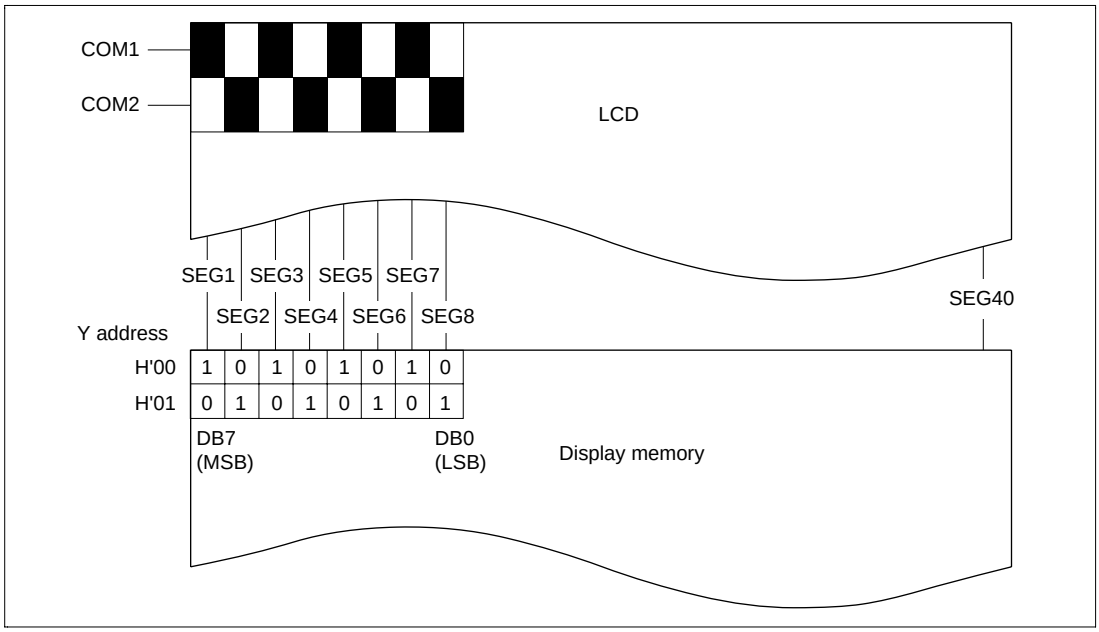


Figure 14.4 Memory Data and Display

14.3.5 Display Data Output

The relationship between the LCD controller display duty and output pins is shown in figure 14.5.

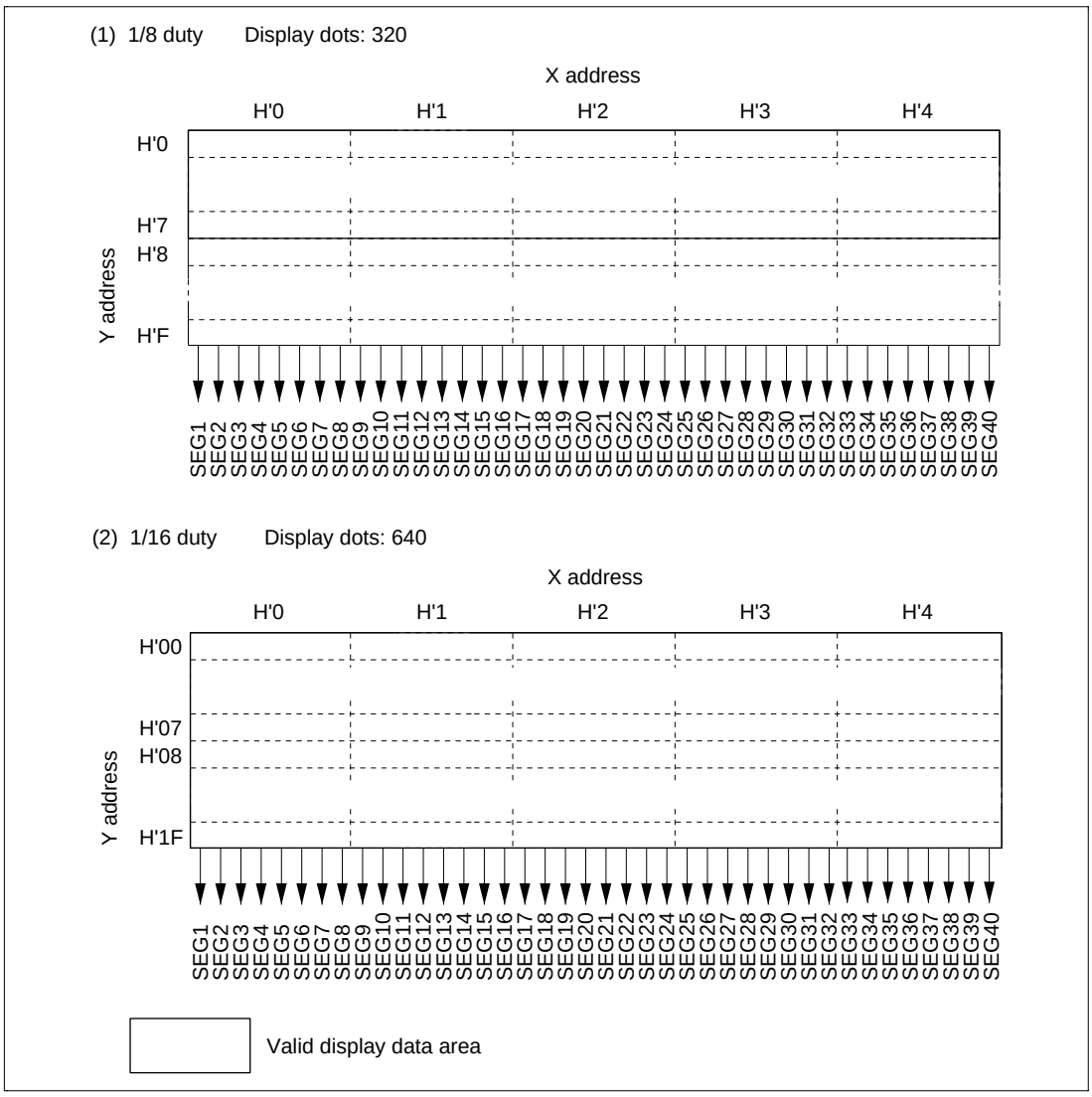


Figure 14.5 Display Duty and Valid Display Data Area

14.3.6 Register and Display Memory Access

Register Access

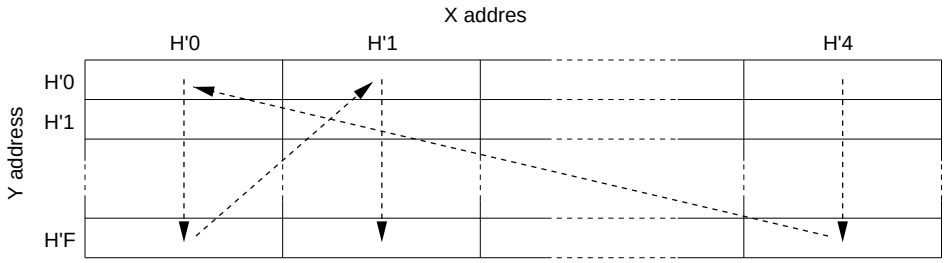
To access a register, RS is first cleared to 0 and the register number of the register to be accessed is set in the index register. Then RS is set to 1, enabling the specified register to be accessed. Some internal registers have nonexistent bits; 0 must be written to these bits. The display data register (LR4) is the only register that can be read.

Display Memory Access

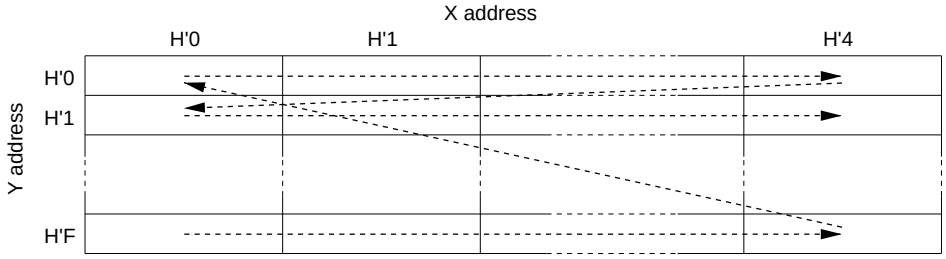
To access the display memory, the address to be accessed is set in the address register (LR2). The memory is then accessed via the display data register (LR4). This access can be performed without awareness of the display-side read. See figure 14.6 for the procedure.

After the respective display data register (LR4) accesses, the X and Y addresses are automatically incremented on the basis of the value set in the INC bit in control register 2 (LR1), and therefore address settings need not be made each time.

With 1/16 duty (DDTY1 = 0), if INC = 0 the X address remains the same in each read/write access to the display data register (LR4), while the Y address is automatically incremented up to H'F. After reaching H'F, the Y address returns to H'0 again, and the X address is simultaneously incremented. If INC = 1, on the other hand, the Y address remains the same in each read/write access to the display data register (LR4), while the X address is automatically incremented up to H'4. After reaching H'4, the X address returns to H'0 again, and the Y address is simultaneously incremented. In this way, consecutive read/write accesses can be made to the entire display memory area.



(1) Priority given to Y-direction data access (INC = 0)



(2) Priority given to X-direction data access (INC = 1)

Notes: 1. Address register (LR2) bits XA2 to XA0 show the X address, and bits YA3 to YA0 show the Y address.
2. X address operation
Address becomes H'0 after H'4, regardless of the duty.
3. Y address operation
Address becomes H'0 after H'0F when 1/16 duty is set, and after H'07 when 1/8 duty is set.

Figure 14.6 Display Memory Access Methods (1/16 Duty)

Reading for Display

Reads for LCD display are performed asynchronously with respect to accesses by the CPU. However, since simultaneous accesses would corrupt data in the RAM, arbitration is carried out within the chip. Basically, accesses by the CPU have priority, and reads for display are performed in the intervals between CPU accesses.

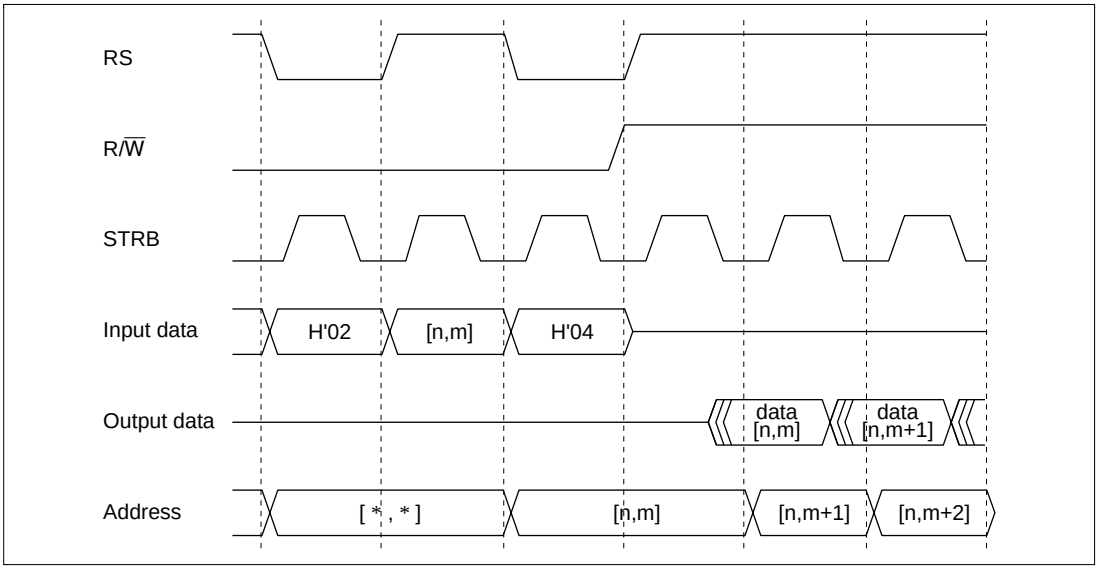


Figure 14.7 Memory Read Procedure

Read-Modify-Write Mode

In the normal state, the X or Y address is incremented after both read and write accesses to the display memory. In read-modify-write mode, the address is incremented only after a write, and remains the same after a read. By using this mode, it is possible to read previously written data, process that data, and then write it back to the same address.

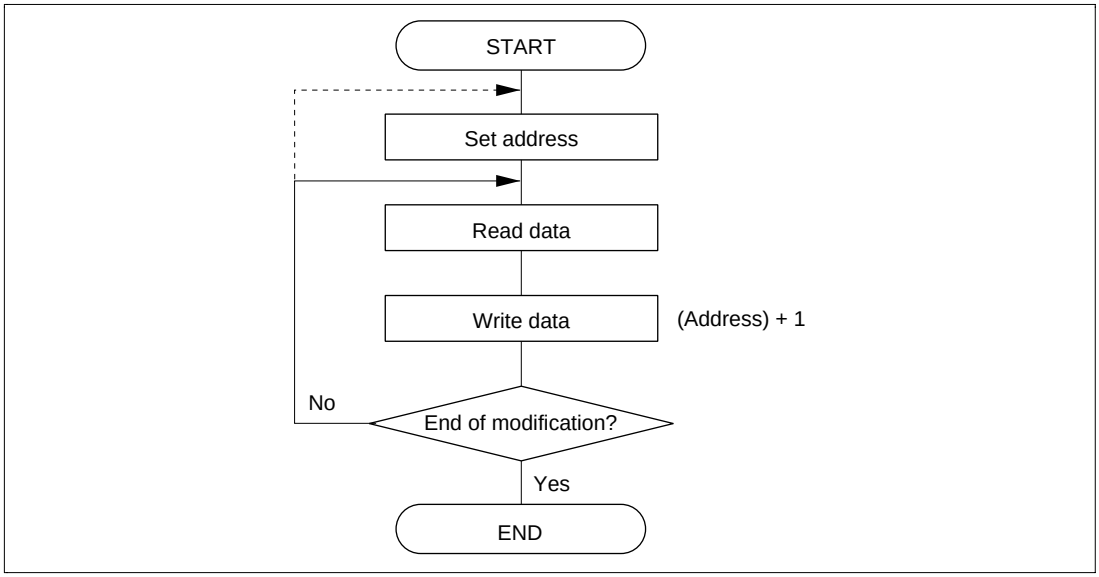


Figure 14.8 Read-Modify-Write Mode Flowchart

14.3.7 Module Standby Mode

The LCD controller has a module standby function that enables low power consumption to be achieved. In module standby mode, the current supply to the built-in bleeder resistances is halted, and segment and common outputs go to the V_{SS} (display-off state) level. Display RAM and internal register data is retained, except for the DISP, LPS1, and LPS0 bits in control register 2 (LR1). The control registers can still be accessed in the module standby state. Figure 14.9 shows the procedures for initiating and clearing module standby mode. The initiation and clearing procedures must be followed exactly in order to protect the display memory contents.

When the CPU is placed in standby mode, set the LSBY bit in control register 1 (LR0) to 1 before executing the standby instruction. After clearing standby mode, follow the module standby clearing procedure to start display.

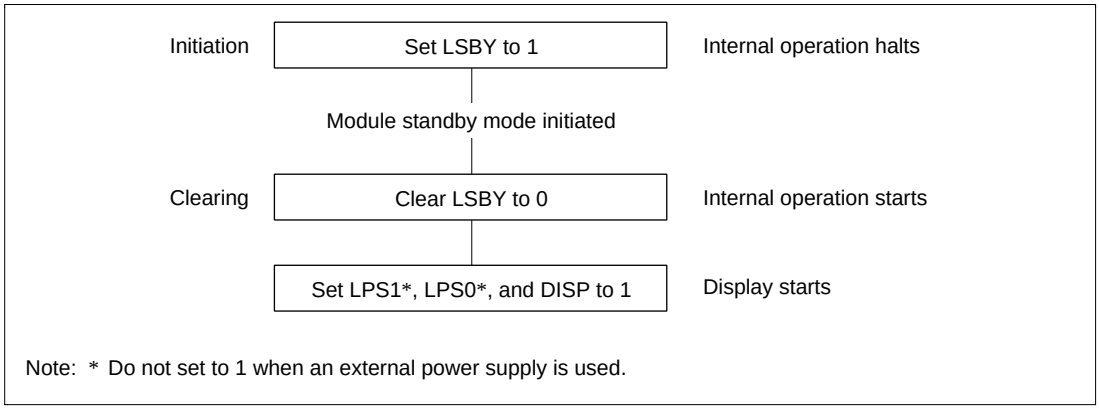
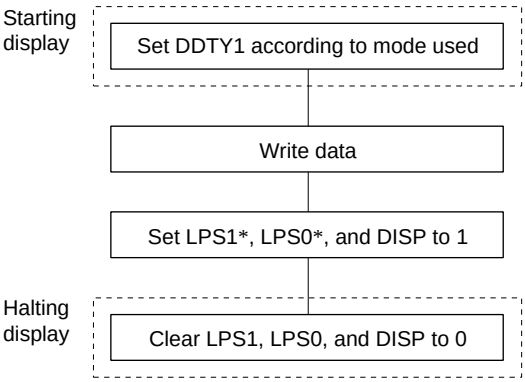


Figure 14.9 Module Standby Mode and Standby Mode Initiation and Clearing Procedures

14.3.8 Power-On and Power-Off Procedures

As the LCD controller incorporates a complete power supply circuit, the procedures shown in figure 14.10 must be followed when powering on and off. Failure to follow these procedures may result in an abnormal display.



Note: * Do not set to 1 when an external power supply is used.

Figure 14.10 Power-On and Power-Off Procedures

14.3.9 Power Supply Circuit

The LCD controller has a built-in bleeder resistance circuit for LCD drive. In standby mode, the voltage circuits are automatically turned off and the power consumption of the power supply circuit falls to zero. The power supply circuit can be turned on and off by a command, and an external power supply circuit should be used if the current capacity of the built-in step-up circuit is insufficient.

LCD Drive Level

Six power supply levels—V1, V2, V3, V3, V4, V5, and V_{SS}—are necessary for LCD drive. The V1 to V_{SS} power supplies are normally generated by means of resistive division.

When 1/4 bias is used for LCD display, the V3OUT and V4OUT pins should be shorted; when 1/5 bias is used, the V3OUT and V4OUT pins should be left open.

External Power Supply

- When external power supply is input directly to pins V1OUT through V5OUT
  A power supply can be applied directly to V1OUT, V2OUT, V3OUT, V4OUT, and V5OUT from an external source by clearing bits LPS0 and LPS1 to 0 in control register 2 (LR1) to halt the power supply to the built-in bleeder resistance circuit. Apply a voltage not exceeding V_{CC} to pins V1OUT through V5OUT.

- When an external power supply is input to pin V1OUT
  V1 to V5 can be generated by inputting an external power supply to V1OUT, and using the built-in bleeder resistances by setting the LPS1 bit to 1 in control register 2 (LR1). Apply a voltage not exceeding V_{CC} to pin V1OUT.

In either case, inputting a voltage exceeding V_{CC} may adversely affect the reliability of the chip.

14.3.10 LCD Drive Power Supply Voltages

There are six LCD drive power supply voltage values—V1 to V5, and V_{SS}. V1 is the highest voltage, and V_{SS} the lowest. As shown in figure 14.11, the common waveforms are formed from a combination of V1, V2, V5, and V_{SS}, while the segment waveforms are formed from a combination of V1, V3, V4, and V_{SS}. V1 and V_{SS} are shared by both common and segment waveforms, but the intermediate voltages are different.

In figure 14.11, the waveforms of outputs SEG1 to SEG40 differ according to the display data. In this example, LCD panel lines for which COM1 is connected are illuminated, and all other dots are not illuminated.

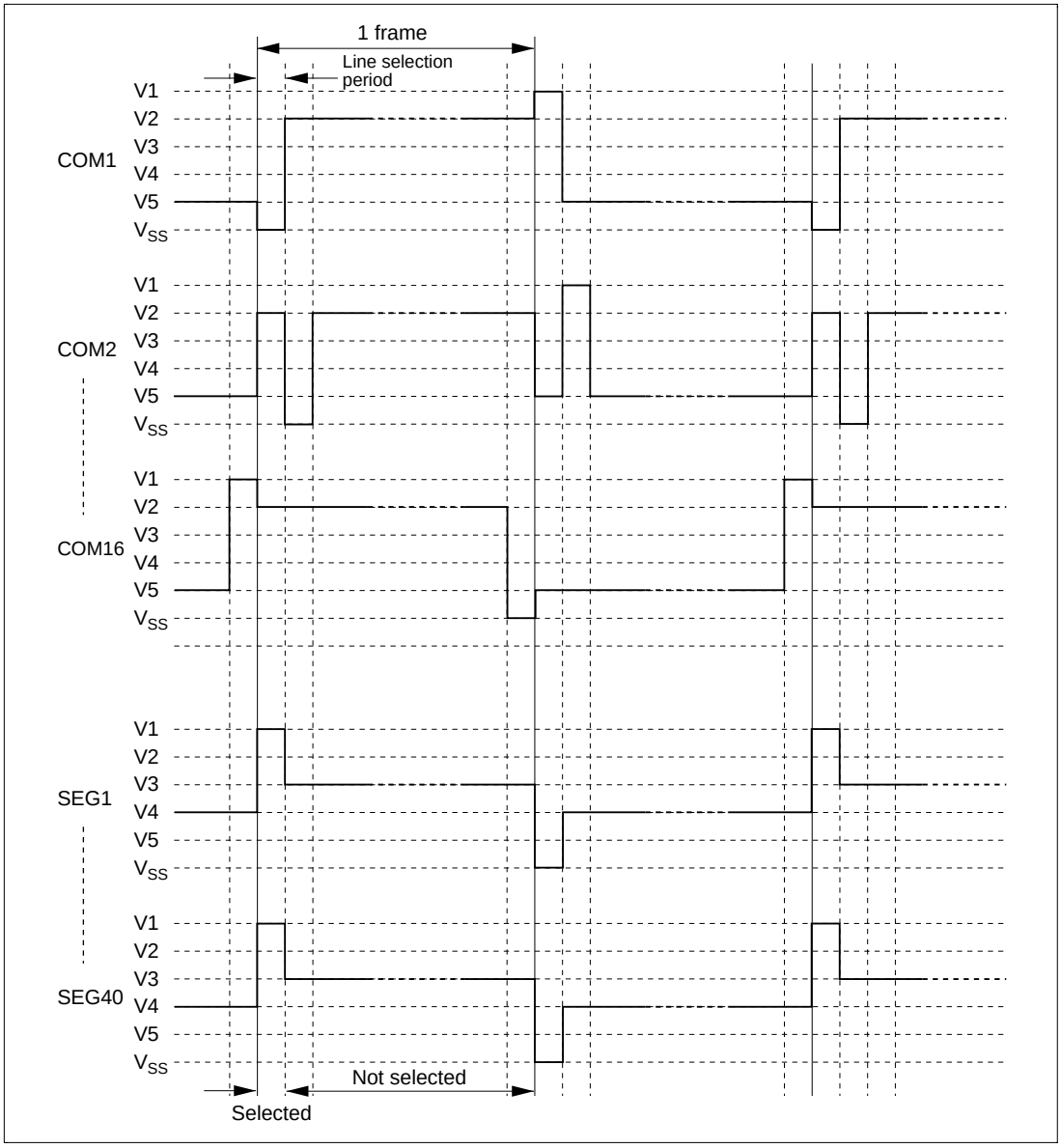
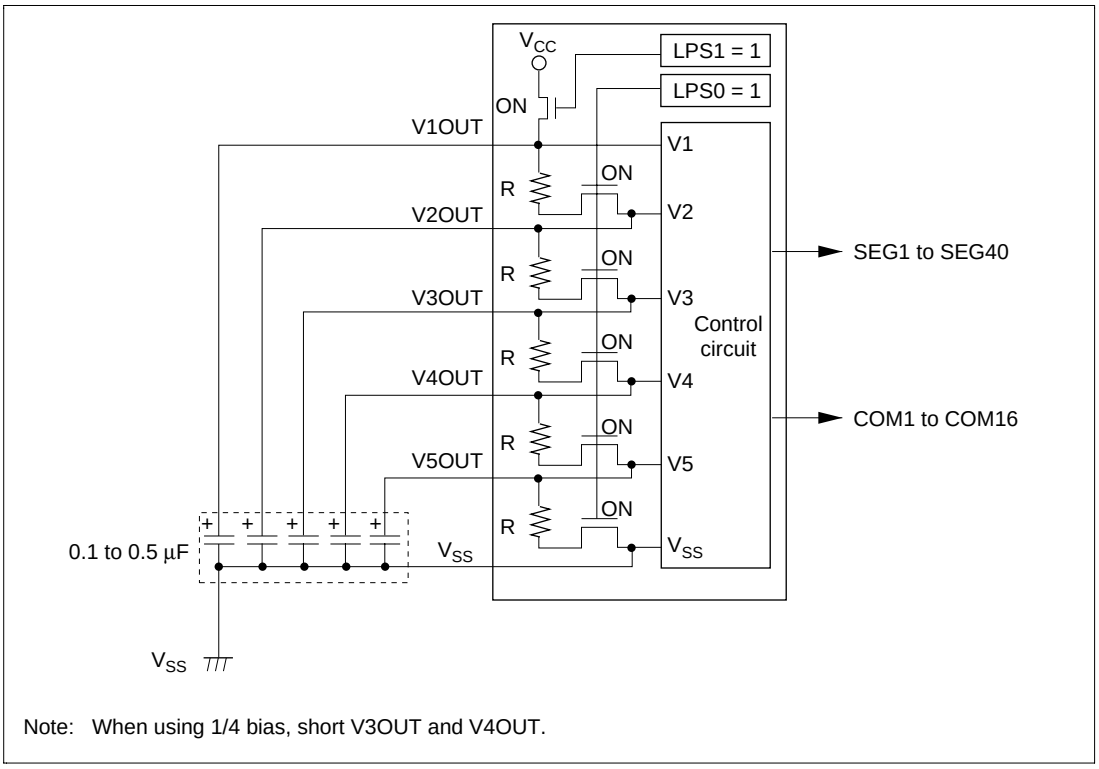


Figure 14.11 LCD Drive Power Supply Waveforms (1/16 Duty)

14.3.11 LCD Voltage Generation Circuit

When Using Internal Power Supply and Built-In Bleeder Resistances

The LCD controller includes bleeder resistances that generate levels V1 to V5. For the LCD drive power supply, drive can be performed using the internal power supply and V_{CC}, or using an external supply. When the internal power supply is used, and the built-in bleeder resistances are employed, bits LPS1 and LPS0 in control register 2 (LR1) should both be set to 1. If the capacitance of the LCD panel to be driven is large, capacitors of around 0.1 to 0.5 µF should be inserted between V1OUT to V5OUT and V_{SS} to provide stabilization.

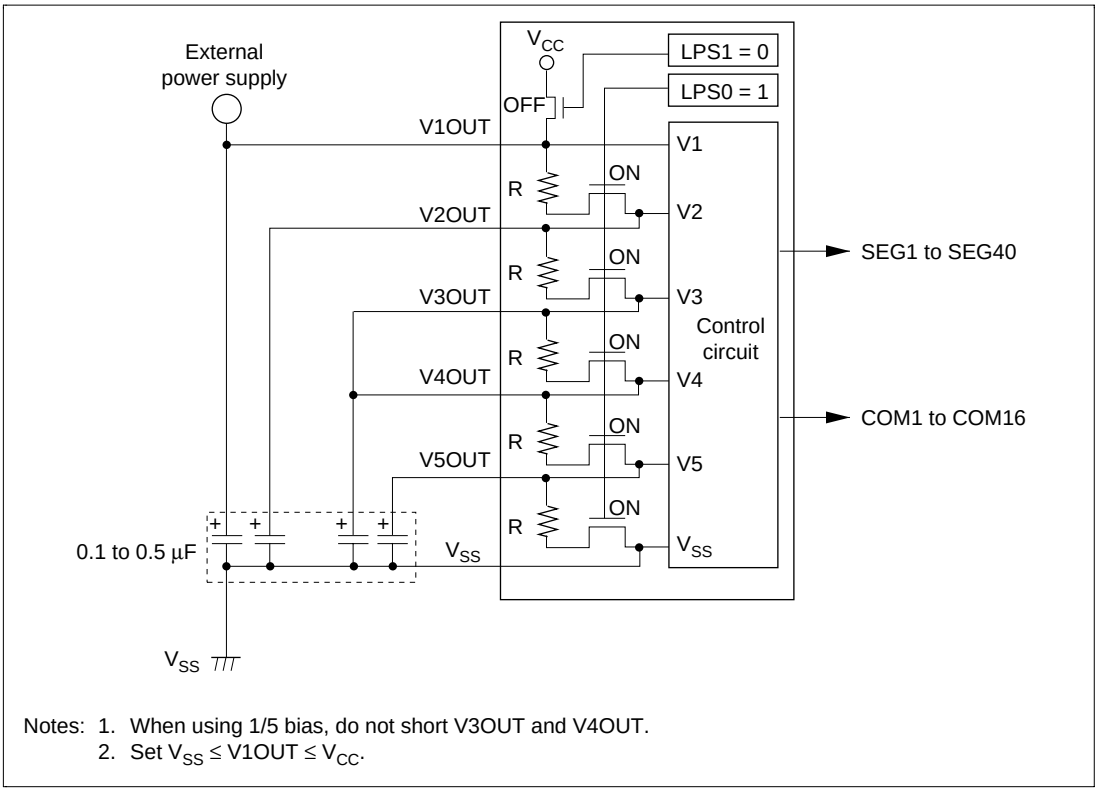


Note: When using 1/4 bias, short V3OUT and V4OUT.

Figure 14.12 When Using Internal Power Supply and Built-In Bleeder Resistances (1/5 Bias)

When Using External Power Supply and Built-In Bleeder Resistances

When an external power supply is supplied from V1OUT and the built-in bleeder resistances are used, clear LPS1 to 0 and set LPS0 to 1 in control register 2 (LR1), and make the connections shown in figure 14.13. The power supply applied to V1OUT must not exceed V_{CC}. If the capacitance of the LCD panel to be driven is large, capacitors of around 0.1 to 0.5 µF should be inserted between V1OUT to V5OUT and V_{SS} to provide stabilization.

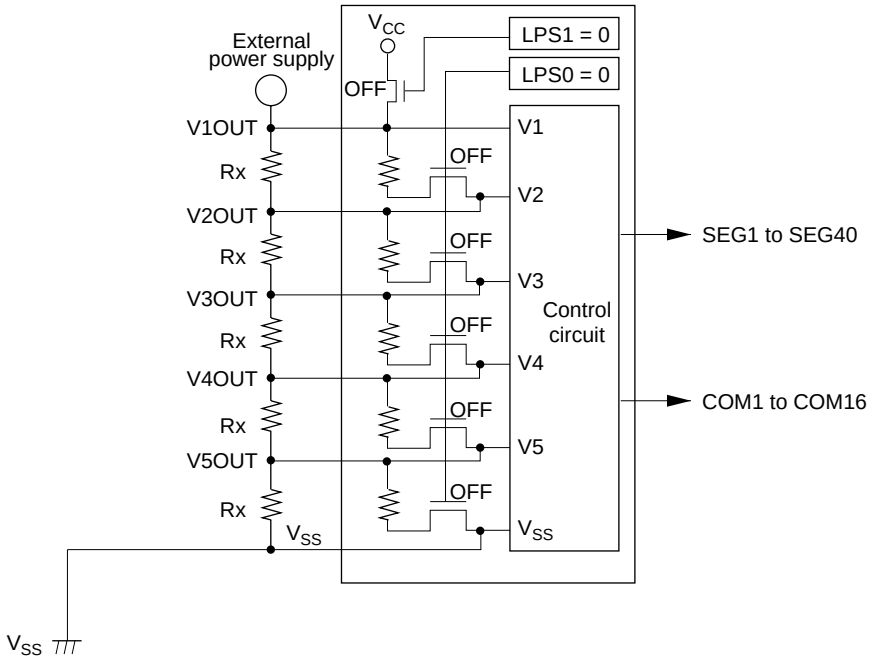


Notes: 1. When using 1/5 bias, do not short V3OUT and V4OUT.
2. Set $V_{SS} \leq V1OUT \leq V_{CC}$.

Figure 14.13 When Using External Power Supply and Built-In Bleeder Circuit (1/4 Bias)

When Using External Power Supply and Bleeder Resistances

If the drive capability of the built-in bleeder resistance is insufficient for the size of the LCD panel, the V1 to V5 levels can be supplied from external bleeder resistances. In this case, clear the LPS1 and LPS0 bits in control register 2 (LR1) to 0, and make the connections shown in figure 14.14.



Notes: 1. When using 1/4 bias, short V3OUT and V4OUT.
2. Set $V_{SS} \leq V_{in} \leq V_{CC}$ (V_{in}: V1OUT to V5OUT).
3. A value of around 5 kΩ to 25 kΩ is recommended for external power supply division resistances Rx.

Figure 14.14 When Using External Power Supply and External Bleeder Circuit (1/5 Bias)

14.3.12 LCD Drive Bias Selection Circuit

The ideal bias value that gives the best contrast is calculated using the equation shown below. If drive is performed at a bias value lower than the optimum, contrast will deteriorate, but the LCD drive voltage (the potential difference between V1 and V_{SS}) can be kept low. If the output voltage falls and the LCD display becomes faint as batteries wear out, for instance, the display can be made clearer by decreasing the LCD drive bias.

$$\text{Optimum bias value for 1/N duty drive} = \frac{1}{\sqrt{N+1}}$$

Notes: 1. When using 1/5 bias, leave the V3OUT and V4OUT pins open.
2. When using 1/4 bias, short the V3OUT and V4OUT pins.

Section 15 Electrical Characteristics (H8/3857 Series)

15.1 H8/3855, H8/3856, and H8/3857 Absolute Maximum Ratings (Standard Specifications)

Table 15.1 shows the absolute maximum ratings.

Table 15.1 Absolute Maximum Ratings

Item		Symbol	Value	Unit	Notes
Power supply voltage		V_{CC}	–0.3 to +7.0	V	
Analog power supply voltage		AV_{CC}	–0.3 to +7.0	V	
LCD power supply voltage		V_{LCD}	–0.3 to +8.0	V	*1
Programming voltage (FWE)		V_{in}	–0.3 to V_{CC}+0.3	V	*2
Input voltage	Except port B and LCD power supply	V_{in}	–0.3 to V_{CC}+0.3	V	
	Port B	AV_{in}	–0.3 to AV_{CC}+0.3	V	
	LCD power supply	V_{in}	–0.3 to V_{LCD}+0.3	V	*3
Operating temperature		T_{opr}	–20 to +75	°C	*4
Storage temperature		T_{stg}	–55 to +125	°C	

Caution: Permanent damage may occur to the chip if maximum ratings are exceeded. Normal operation should be under the conditions specified in Electrical Characteristics. Exceeding these values can result in incorrect operation and reduced reliability.

Notes: 1. A voltage not lower than V_{CC} must be applied as LCD power supply voltage V_{LCD}.

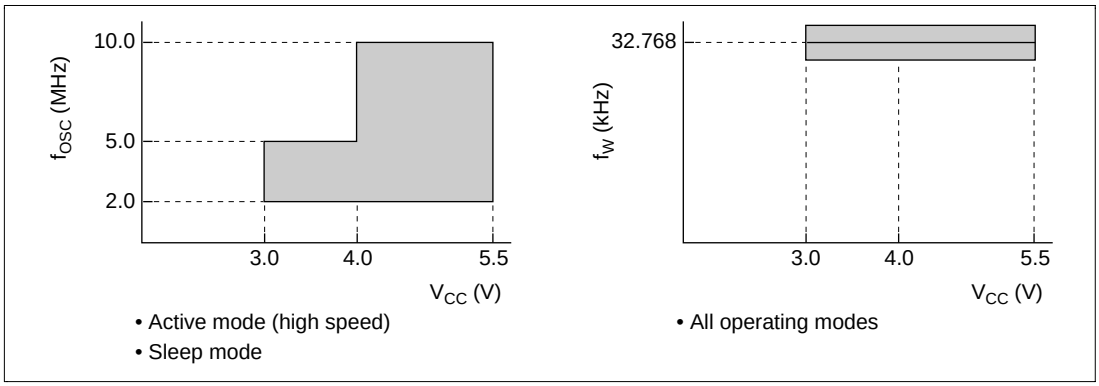
2. 12 V must not be applied to the FWE pin, as this will permanently damage the device.
3. When the built-in op-amps are not used, and the LCD drive voltages are supplied directly from an external source, this applies to V1OUT, V2OUT, V3OUT, V4OUT, and V5OUT.
4. The operating temperature range when programming/erasing flash memory is: T_a = 0°C to +75°C.

15.2 H8/3855, H8/3856, and H8/3857 Electrical Characteristics (Standard Specifications)

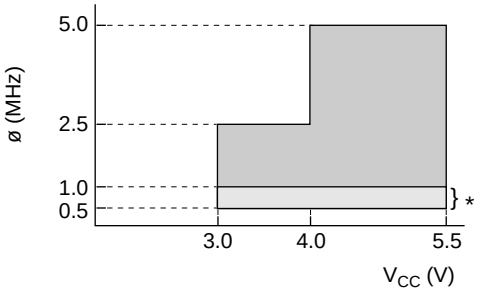
15.2.1 Power Supply Voltage and Operating Range

The power supply voltage and operating range of the H8/3855, H8/3856, and H8/3857 are indicated by the shaded region in the figures below.

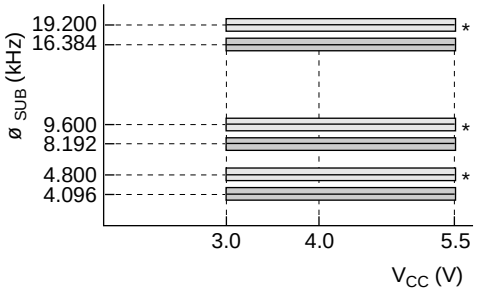
(1) Power Supply Voltage vs. Oscillator Frequency Range



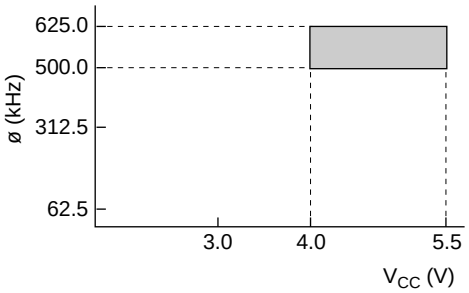
(2) Power Supply Voltage vs. Operating Frequency Range



• Active mode (high speed)
• Sleep mode (except CPU)



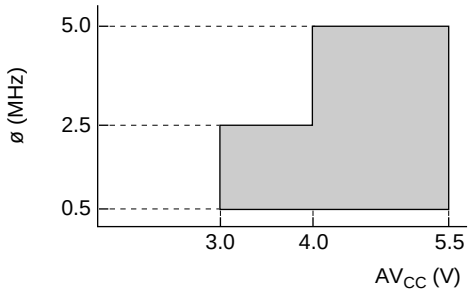
• Subactive mode
• Subsleep mode (except CPU)
• Watch mode (except CPU)



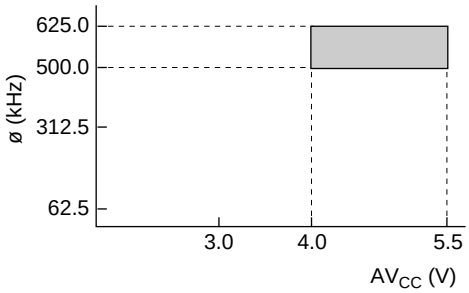
• Active mode (medium speed)

Note : * In case of external clock only

(3) Analog Power Supply Voltage vs. A/D Converter Operating Range



• Active mode (high speed)
• Sleep mode (except CPU)



• Active mode (medium speed)

15.2.2 DC Characteristics

Table 15.2 shows the DC characteristics of the H8/3855, H8/3856, and H8/3857.

Table 15.2 DC Characteristics of H8/3855, H8/3856, and H8/3857 (1)

V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, V_{SS} = AV_{SS} = 0.0 V, T_a = –20°C to +75°C*[4], including subactive mode, unless otherwise specified.

				Values				
Item	**Symbol**	**Applicable Pins**	**Test Conditions**	**Min**	**Typ**	**Max**	**Unit**	**Notes**
Input high voltage	V_{IH}	$\overline{RES}$, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{IRQ_0}$ to $\overline{IRQ_4}$, TMIB, TMIC, TMIF, TEST2, FWE, SCK_1, SCK_3, $\overline{ADTRG}$	V_{CC} = 4.0 V to 5.5 V	0.8V_{CC}	—	V_{CC}+0.3	V	
				0.9V_{CC}	—	V_{CC}+0.3		
		UD, SI_1, RXD	V_{CC} = 4.0 V to 5.5 V	0.7V_{CC}	—	V_{CC}+0.3	V	
				0.8V_{CC}	—	V_{CC}+0.3		
		OSC_1	V_{CC} = 4.0 V to 5.5 V	V_{CC}–0.5	—	V_{CC}+0.3	V	
				V_{CC}–0.3	—	V_{CC}+0.3		
		X1		V_{CC}–0.3	—	V_{CC}+0.3	V	
		$P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$	V_{CC} = 4.0 V to 5.5 V	0.7V_{CC}	—	V_{CC}+0.3	V	
				0.8V_{CC}	—	V_{CC}+0.3		
		PB_0 to PB_7	V_{CC} = 4.0 V to 5.5 V	0.7V_{CC}	—	AV_{CC}+0.3	V	
				0.8V_{CC}	—	AV_{CC}+0.3		
Input low voltage	V_{IL}	$\overline{RES}$, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{IRQ_0}$ to $\overline{IRQ_4}$, TMIB, TMIC, TMIF, TEST2, FWE, SCK_1, SCK_3, $\overline{ADTRG}$	V_{CC} = 4.0 V to 5.5 V	–0.3	—	0.2V_{CC}	V	
				–0.3	—	0.1V_{CC}		
		UD, SI_1, RXD	V_{CC} = 4.0 V to 5.5 V	–0.3	—	0.3V_{CC}	V	
				–0.3	—	0.2V_{CC}		
		OSC_1	V_{CC} = 4.0 V to 5.5 V	–0.3	—	0.5	V	
				–0.3	—	0.3		

Item	Symbol	Applicable Pins	Test Conditions	Values Min	Typ	Max	Unit	Notes
Input low voltage	V_{IL}	X1		–0.3	—	0.3	V	
		$P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$, PB_0 to PB_7	V_{CC} = 4.0 V to 5.5 V	–0.3	—	$0.3V_{CC}$	V	
				–0.3	—	$0.2V_{CC}$		
Output high voltage	V_{OH}	$P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_2$, $P5_0$ to $P5_7$	V_{CC} = 4.0 V to 5.5 V $-I_{OH}$ = 1.0 mA	V_{CC}–1.0	—	—	V	
			V_{CC} = 4.0 V to 5.5 V $-I_{OH}$ = 0.5 mA	V_{CC}–0.5	—	—		
			$-I_{OH}$ = 0.1 mA	V_{CC}–0.5	—	—		
Output low voltage	V_{OL}	$P1_0$ to $P1_7$, $P4_0$ to $P4_2$, $P5_0$ to $P5_7$	V_{CC} = 4.0 V to 5.5 V I_{OL} = 1.6 mA	—	—	0.6	V	
			I_{OL} = 0.4 mA	—	—	0.5		
		$P2_0$ to $P2_7$, $P3_0$ to $P3_7$	V_{CC} = 4.0 V to 5.5 V I_{OL} = 10 mA	—	—	1.5	V	
			V_{CC} = 4.0 V to 5.5 V I_{OL} = 1.6 mA	—	—	0.6		
			I_{OL} = 0.4 mA	—	—	0.5		
Input/output leakage current	$\lvert I_{IL} \rvert$	$\overline{RES}$, TEST2, FWE, OSC_1, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$	V_{in} = 0.5 V to V_{CC} – 0.5 V	—	—	1.0	µA	
		PB_0 to PB_7	V_{in} = 0.5 V to AV_{CC} – 0.5 V	—	—	1.0	µA	
Pull-up MOS current	$-I_p$	$P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P5_0$ to $P5_7$	V_{CC} = 5 V, V_{in} = 0 V	50.0	—	300.0	µA	
			V_{CC} = 3.3 V, V_{in}= 0 V	—	100	—	µA	Reference values
Input capacitance	C_{in}	All input pins except power supply pins	f = 1 MHz, V_{in} = 0 V, T_a = 25°C	—	—	15.0	pF	
Active mode current dissipation	I_{OPE1}	V_{CC}	Active mode (high speed) V_{CC} = 5 V, f_{OSC} = 10 MHz	—	10.0	15.0	mA	*1 *2
	I_{OPE2}	V_{CC}	Active mode (medium speed) V_{CC} = 5 V, f_{OSC} = 10 MHz	—	2.0	3.5	mA	*1 *2

Item	Symbol	Applicable Pins	Test Conditions	Values			Unit	Notes
				Min	Typ	Max		
Sleep mode current dissipation	I_{SLEEP}	V_{CC}	V_{CC} = 5 V, f_{OSC} = 10 MHz	—	4.0	7.0	mA	*1 *2
Subactive mode current dissipation	I_{SUB}	V_{CC}	V_{CC} = 3.3 V, LCD on, (with 2X step-up) 32-kHz crystal oscillator used ($ø_{SUB}$= $ø_W$/2)	—	70	150	µA	*1 *2
			V_{CC} = 3.3 V, LCD on, (with 2X step-up) 32-kHz crystal oscillator used ($ø_{SUB}$= $ø_W$/8)	—	65	—	µA	*1 *2 Reference values
			V_{CC} = 3.3 V, LCD not used, 32-kHz crystal oscillator used ($ø_{SUB}$= $ø_W$/2)	—	20	—	µA	*1 *2 Reference values
Subsleep mode current dissipation	I_{SUBSP}	V_{CC}	V_{CC} = 3.3 V, LCD on, (with 2X step-up) 32-kHz crystal oscillator used ($ø_{SUB}$= $ø_W$/2)	—	65	130	µA	*1 *2
Watch mode current dissipation	I_{WATCH}	V_{CC}	V_{CC} = 3.3 V, LCD on, (with 2X step-up) 32-kHz crystal oscillator used	—	60	90.0	µA	*1 *2
			V_{CC} = 3.3 V, LCD not used, 32-kHz crystal oscillator used	—	7.0	15.0	µA	*1 *2
Standby mode current dissipation	I_{STBY}	V_{CC}	32-kHz crystal oscillator not used	—	—	5.0	µA	*1 *2
Program/ erase current dissipation	I_{FLASH}	V_{CC}	0°C ≤ T_a ≤ 70°C f_{OSC} = 12 MHz	—	16	22	mA	*1 *2 *3
RAM data retaining voltage	V_{RAM}	V_{CC}		2.0	—	—	V	*1 *2

Notes: 1. Pin states during current measurement

Mode	Internal State	Pins	LCD Power Supply	Oscillator Pins
Active mode (high and medium speed)	Operates	V_{CC}	V_{LCD} = 6.0 V	System clock oscillator: Crystal Subclock oscillator: Pin X_1 = V_{CC}
Sleep mode	Only timer operates	V_{CC}	V_{LCD} = 6.0 V	
Subactive mode	Operates	V_{CC}	V_{LCD} = 6.0 V (When LCD is not used, V_{LCD} = V_{CC})	System clock oscillator: Crystal Subclock oscillator: Crystal
Subsleep mode	Only timer operates, CPU stops	V_{CC}	V_{LCD} = 6.0 V	
Watch mode	Only time-base clock operates, CPU stops	V_{CC}	V_{LCD} = 6.0 V (When LCD is not used, V_{LCD} = V_{CC})	
Standby mode	CPU and timers all stop	V_{CC}	V_{LCD} = V_{CC}	System clock oscillator: Crystal Subclock oscillator: Pin X_1 = V_{CC}
Programming/ erasing*[3]	Operates	V_{CC}	V_{LCD} = V_{CC}	System clock oscillator: Crystal Subclock oscillator: Pin X_1 = V_{CC}

2. Excludes current in pull-up MOS transistors and output buffers.
3. Applies to F-ZTAT version only.
4. The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

Table 15.3 DC Characteristics of H8/3855, H8/3856, and H8/3857 (2)

V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, V_{SS} = AV_{SS} = 0.0 V, T_a = –20°C to +75°C*[2], including subactive mode, unless otherwise specified.

Item	Symbol	Applicable Pins	Test Conditions	Values Min	Typ	Max	Unit	Notes
Allowable output low current (per pin)	I_{OL}	Output pins except in ports 2 and 3	V_{CC} = 4.0 V to 5.5 V	—	—	2.0	mA	*[1]
		Ports 2 and 3	V_{CC} = 4.0 V to 5.5 V	—	—	10.0		
		All output pins		—	—	0.5		
Allowable output low current (total)	ΣI_{OL}	Output pins except in ports 2 and 3	V_{CC} = 4.0 V to 5.5 V	—	—	20.0	mA	*[1]
		Ports 2 and 3	V_{CC} = 4.0 V to 5.5 V	—	—	80.0		
		All output pins		—	—	20.0		
Allowable output high current (per pin)	$-I_{OH}$	All output pins	V_{CC} = 4.0 V to 5.5 V	—	—	2.0	mA	*[1]
				—	—	0.2		
Allowable output high current (total)	$\Sigma -I_{OH}$	All output pins	V_{CC} = 4.0 V to 5.5 V	—	—	10.0	mA	*[1]
				—	—	8.0		

Notes: 1. Excludes LCD output pins.
2. The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

15.2.3 AC Characteristics

Table 15.4 shows the control signal timing, and tables 15.5 and 15.6 show the serial interface timing, of the H8/3855, H8/3856, and H8/3857.

Table 15.4 Control Signal Timing of H8/3855, H8/3856, and H8/3857

V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, V_{SS} = AV_{SS} = 0.0 V, T_a = –20°C to +75°C*[3], including subactive mode, unless otherwise specified.

Item	Symbol	Applicable Pins	Test Conditions	Values Min	Typ	Max	Unit	Reference Figure
System clock oscillation frequency	f_{OSC}	OSC1, OSC2	V_{CC} = 4.0 V to 5.5 V	2.0	—	10.0	MHz	
				2.0	—	5.0		
OSC clock ($ø_{OSC}$) cycle time	t_{OSC}	OSC1, OSC2	V_{CC} = 4.0 V to 5.5 V	100.0	—	1000.0	ns	*1
				200.0	—	1000.0		Figure15.1
System clock (ø) cycle time	t_{cyc}			2	—	16	t_{OSC}	*1
				—	—	2000.0	ns	
Subclock oscillation frequency	f_W	X1, X2		—	32.768	—	kHz	
Watch clock ($ø_W$) cycle time	t_W	X1, X2		—	30.5	—	µs	
Subclock ($ø_{SUB}$) cycle time	t_{subcyc}			2	—	8	t_W	*2
Instruction cycle time				2	—	—	t_{cyc} t_{subcyc}	
Oscillation stabilization time (crystal oscillator)	t_{rc}	OSC_1, OSC_2	V_{CC} = 4.0 V to 5.5 V	40.0	—	—	ms	
				60.0	—	—		
Oscillation stabilization time	t_{rc}	X_1, X_2		2	—	—	s	
External clock high width	t_{CPH}	OSC_1	V_{CC} = 4.0 V to 5.5 V	40.0	—	—	ns	Figure 15.1
				80.0	—	—		
External clock low width	t_{CPL}	OSC_1	V_{CC} = 4.0 V to 5.5 V	40.0	—	—	ns	Figure 15.1
				80.0	—	—		
External clock rise time	t_{CPr}	OSC_1	V_{CC} = 4.0 V to 5.5 V	—	—	15.0	ns	Figure 15.1
				—	—	20.0		
External clock fall time	t_{CPf}	OSC_1	V_{CC} = 4.0 V to 5.5 V	—	—	15.0	ns	Figure 15.1
				—	—	20.0		

Item	Symbol	Applicable Pins	Test Conditions	Values Min	Typ	Max	Unit	Reference Figure
External subclock high width	t_{XH}	X_1		0.4/fx	—	—	s	Figure 15.2
External subclock low width	t_{XL}	X_1		0.4/fx	—	—	s	Figure 15.2
External subclock rise time	t_{Xr}	X_1		—	—	100.0	ns	Figure 15.2
External subclock fall time	t_{Xf}	X_1		—	—	100.0	ns	Figure 15.2
$\overline{RES}$ pin low width	t_{REL}	$\overline{RES}$		10	—	—	t_{cyc}	Figure 15.3
Input pin high width	t_{IH}	$\overline{IRQ_0}$ to $\overline{IRQ_4}$, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{ADTRG}$, TMIB, TMIC, TMIF		2	—	—	t_{cyc} t_{subcyc}	Figure 15.4
Input pin low width	t_{IL}	$\overline{IRQ_0}$ to $\overline{IRQ_4}$, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{ADTRG}$, TMIB, TMIC, TMIF		2	—	—	t_{cyc} t_{subcyc}	Figure 15.4
UD pin minimum transition width	t_{UDH} t_{UDL}	UD		4	–	–	t_{cyc} t_{subcyc}	Figure 15.5

Notes: 1. A frequency between 1 MHz and 10 MHz is required when an external clock is input.
2. Selected with bits SA1 and SA0 in system control register 2 (SYSCR2).
3. The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

Table 15.5 Serial Interface (SCI1) Timing of H8/3855, H8/3856, and H8/3857

V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, V_{SS} = AV_{SS} = 0.0 V, T_a = –20°C to +75°C*, unless otherwise specified.

Item	Symbol	Applicable Pins	Test Conditions	Values Min	Typ	Max	Unit	Reference Figure
Input serial clock cycle time	t_{Scyc}	SCK_1		4	—	—	t_{cyc}	Figure 15.6
Input serial clock high width	t_{SCKH}	SCK_1		0.4	—	—	t_{Scyc}	Figure 15.6
Input serial clock low width	t_{SCKL}	SCK_1		0.4	—	—	t_{Scyc}	Figure 15.6
Input serial clock rise time	t_{SCKr}	SCK_1	V_{CC} = 4.0 V to 5.5 V	—	—	60.0	ns	Figure 15.6
				—	—	80.0		
Input serial clock fall time	t_{SCKf}	SCK_1	V_{CC} = 4.0 V to 5.5 V	—	—	60.0	ns	Figure 15.6
				—	—	80.0		
Serial output data delay time	t_{SOD}	SO_1	V_{CC} = 4.0 V to 5.5 V	—	—	200.0	ns	Figure 15.6
				—	—	350.0		
Serial input data setup time	t_{SIS}	SI_1	V_{CC} = 4.0 V to 5.5 V	200.0	—	—	ns	Figure 15.6
				400.0	—	—		
Serial input data hold time	t_{SIH}	SI_1	V_{CC} = 4.0 V to 5.5 V	200.0	—	—	ns	Figure 15.6
				400.0	—	—		

Note: * The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

Table 15.6 Serial Interface (SCI3) Timing of H8/3855, H8/3856, and H8/3857

V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, V_{SS} = AV_{SS} = 0.0 V, T_a = –20°C to +75°C*, unless otherwise specified.

Item		Symbol	Test Conditions	Values			Unit	Reference Figure
				Min	Typ	Max		
Input clock cycle	Asynchronous	t_{Scyc}		4	—	—	t_{cyc}	Figure 15.7
	Synchronous			6	—	—		
Input clock pulse width		t_{SCKW}		0.4	—	0.6	t_{Scyc}	Figure 15.7
Transmit data delay time (synchronous mode)		t_{TXD}	V_{CC} = 4.0 V to 5.5 V	—	—	1	t_{cyc}	Figure 15.8
				—	—	1		
Receive data setup time (synchronous mode)		t_{RXS}	V_{CC} = 4.0 V to 5.5 V	200.0	—	—	ns	Figure 15.8
				400.0	—	—		
Receive data hold time (synchronous mode)		t_{RXH}	V_{CC} = 4.0 V to 5.5 V	200.0	—	—	ns	Figure 15.8
				400.0	—	—		

Note: * The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

15.2.4 A/D Converter Characteristics

Table 15.7 shows the A/D converter characteristics of the H8/3855, H8/3856, and H8/3857.

Table 15.7 A/D Converter Characteristics of H8/3855, H8/3856, and H8/3857

V_{CC} = 3.0 V to 5.5 V, V_{SS} = AV_{SS} = 0.0 V, T_a = –20°C to +75°C*[4], unless otherwise specified.

Item	Symbol	Applicable Pins	Test Conditions	Values Min	Values Typ	Values Max	Unit	Reference Figure
Analog power supply voltage	AV_{CC}	AV_{CC}		3.0	—	5.5	V	*[1]
Analog input voltage	AV_{IN}	AN_0 to AN_7		AV_{SS}–0.3	—	AV_{CC} + 0.3	V	
Analog power supply current	AI_{OPE}	AV_{CC}	AV_{CC} = 5.0 V	—	—	1.5	mA	
	AI_{STOP1}	AV_{CC}	AV_{CC} = 5.0 V	—	300	—	µA	*[2] Reference value
	AI_{STOP2}	AV_{CC}		—	—	5.0	µA	*[3]
Analog input capacitance	C_{AIN}	AN_0 to AN_7		—	—	30.0	pF	
Allowable signal source impedance	R_{AIN}			—	—	5.0	kΩ	
Resolution (data length)				—	—	8	Bit	
Non-linearity error				—	—	±2.0	LSB	
Quantization error				—	—	±0.5	LSB	
Absolute accuracy				—	—	±2.5	LSB	
Conversion time				12.4	—	124	µs	

Notes: 1. Set $AV_{CC} \leq V_{CC}$, and set $AV_{CC} = V_{CC}$ when the A/D converter is not used.
2. AI_{STOP1} is the current in active and sleep modes while the A/D converter is idle.
3. AI_{STOP2} is the current at reset and in standby, watch, subactive, and subsleep modes while the A/D converter is idle.
4. The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

15.2.5 LCD Characteristics

Table 15.8 shows the LCD characteristics, and table 15.9 shows the step-up circuit characteristics, of the H8/3855, H8/3856, and H8/3857.

Table 15.8 LCD Characteristics of H8/3855, H8/3856, and H8/3857

V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, V_{SS} = AV_{SS} = 0.0 V, T_a = –20°C to +75°C*[4], including subactive mode, unless otherwise specified.

				Values				
Item	**Symbol**	**Applicable Pins**	**Test Conditions**	**Min**	**Typ**	**Max**	**Unit**	**Notes**
Common driver on-resistance	R_{COM}	COM1 to COM32	±Id = 0.05 mA, V_{LCD} = 4 V	—	6	20	kΩ	*[1]
Segment driver on-resistance	R_{SEG}	SEG1 to SEG64	±Id = 0.05 mA, V_{LCD} = 4 V	—	6	20	kΩ	*[1]
LCD power supply current	I_{EE}	V_{LCD}	V_{LCD} = 5.5 V, f_x = 32.768 kHz	—	20	40	µA	*[2]
LCD power supply voltage	V_{LCD}	V_{LCD}		V_{CC}	—	7.0	V	*[3]

Notes: 1. Applies to the resistance (R_{COM}) between the V1OUT, V2OUT, V5OUT, and V_{SS} pins and the common signal pins (COM1 to COM32), and the resistance (R_{SEG}) between the V1OUT, V3OUT, V4OUT, and V_{SS} pins and the segment signal pins (SEG1 to SEG64), when Id is flowing in the pins.

2. This is the current when the built-in op-amps are operating and display is halted (all driver outputs are at the V_{SS} level).
3. Specifies the voltage range in which the COM/SEG pin output voltages are within the LCD reference voltage values (V1, V2, V3, V4, V5, and V_{SS}) ±0.15 V in the unloaded state. A voltage not lower than V_{CC} must be applied to V_{LCD}.
4. The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

Table 15.9 Step-Up Circuit Characteristics of H8/3855, H8/3856, and H8/3857

V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, V_{SS} = AV_{SS} = 0.0 V, T_a = –20°C to +75°C*[2], including subactive mode, unless otherwise specified.

Item	Symbol	Applicable Pins	Test Conditions	Values Min	Typ	Max	Unit	Notes
2X step-up output voltage	V_{UP2}	VLOUT	V_{CC} = V_{ci} = 3.0 V, I_O = 0.03 mA, C = 1 µF, X1 = 32 kHz, T_a = 25°C	—	5.96	—	V	Figure 15.9 Reference values
3X step-up output voltage	V_{UP3}	VLOUT	V_{CC} = 3.0 V, V_{ci} = 2.0 V, I_O = 0.03 mA, C = 1 µF, X1 = 32 kHz, T_a = 25°C	—	5.90	—	V	Figure 15.9 Reference values
Step-up circuit reference voltage	V_{ci}	V_{ci}	$V_{ci} \leq V_{CC}$	1.6	—	3.5	V	*[1]

Notes: 1. As $V_{CC} \leq$ VLOUT $\leq$ 7.0 V, with 2X step-up $V_{CC}/2 \leq V_{ci} \leq$ 3.5 V, and with 3X step-up $V_{CC}/3 \leq V_{ci} \leq$ 2.33 V.
A voltage not exceeding V_{CC} should be input to V_{ci}. If this condition is not observed, there is a risk of permanent damage to the device.

2. The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

15.2.6 Flash Memory Characteristics

Table 15.10 shows the flash memory characteristics.

Table 15.10 Flash Memory Characteristics

Conditions: V_{CC} = 4.5 V to 5.5 V, AV_{CC} = 4.5 V to 5.5 V, V_{SS} = AV_{SS} = 0.0 V, T_a = 0°C to +75°C (program/erase operating temperature range)

Item		Symbol	Min	Typ	Max	Unit	Test Conditions
Programming time*1, *2, *4		t_P	—	10	200	ms/32 bytes	
Erase time*1, *3, *5		t_E	—	100	300	ms/block	
Rewrite times		N_{WEC}	—	—	100	Times	
Programming	Wait time after SWE bit setting*1	x	10	—	—	µs	
	Wait time after PSU bit setting*1	y	50	—	—	µs	
	Wait time after P bit setting*1, *4	z	—	—	200	µs	
	Wait time after P bit clearing*1	α	10	—	—	µs	
	Wait time after PSU bit clearing*1	β	10	—	—	µs	
	Wait time after PV bit setting*1	γ	4	—	—	µs	
	Wait time after H'FF dummy write*1	ε	2	—	—	µs	
	Wait time after PV bit clearing*1	η	4	—	—	µs	
	Maximum number of writes*1, *4	N	—	—	1000	Times	
Erasing	Wait time after SWE bit setting*1	x	10	—	—	µs	
	Wait time after ESU bit setting*1	y	200	—	—	µs	
	Wait time after E bit setting*1, *5	z	—	—	5	ms	
	Wait time after E bit clearing*1	α	10	—	—	µs	
	Wait time after ESU bit clearing*1	β	10	—	—	µs	
	Wait time after EV bit setting*1	γ	20	—	—	µs	
	Wait time after H'FF dummy write*1	ε	2	—	—	µs	
	Wait time after EV bit clearing*1	η	5	—	—	µs	
	Maximum number of erases*1, *5	N	—	—	60	Times	

Notes: 1. Follow the program/erase algorithms when making the time settings.

2. Programming time per 32 bytes. (Indicates the total time during which the P bit is set in flash memory control register 1 (FLMCR1). Does not include the program-verify time.)
3. Time to erase one block. (Indicates the time during which the E bit is set in FLMCR1. Does not include the erase-verify time.)
4. Maximum programming time
   (t_P(max) = Wait time after P bit setting (z) × maximum number of writes (N))
5. Maximum erase time
   (t_E(max) = Wait time after E bit setting (z) × maximum number of erases (N))

15.3 Operation Timing

Figures 15.1 to 15.8 show timing diagrams.

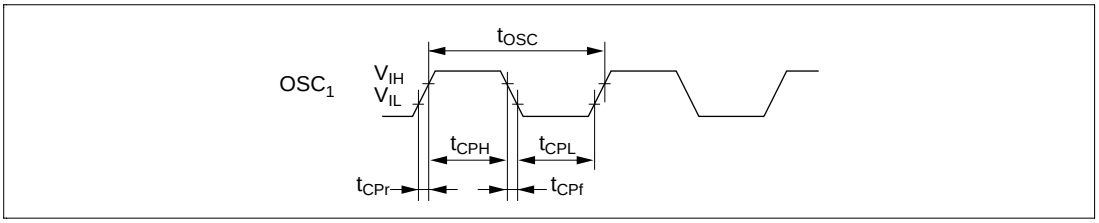


Figure 15.1 System Clock Input Timing

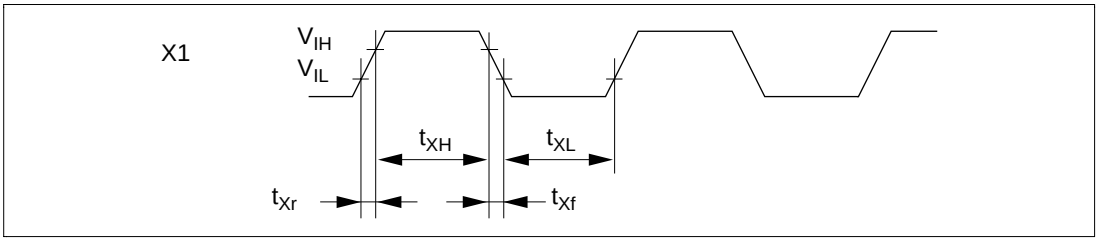


Figure 15.2 Subclock Input Timing

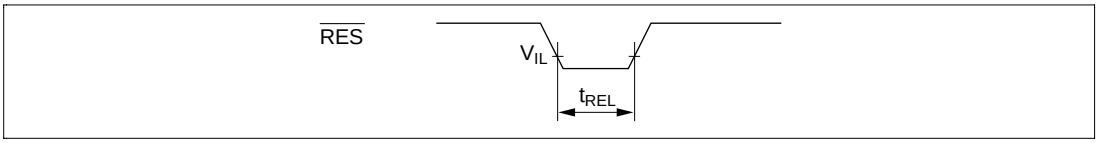


Figure 15.3 $\overline{RES}$ Pin Low Width Timing

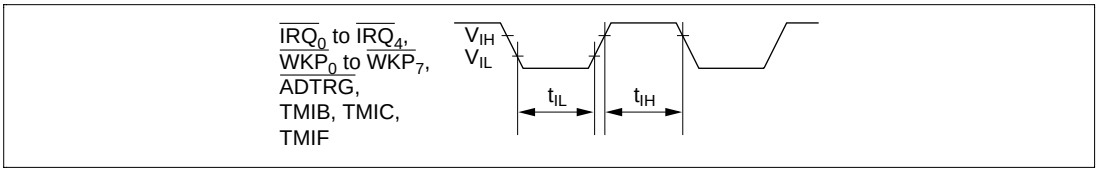


Figure 15.4 Input Timing

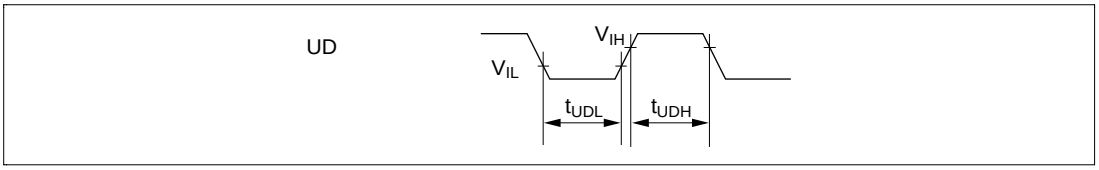
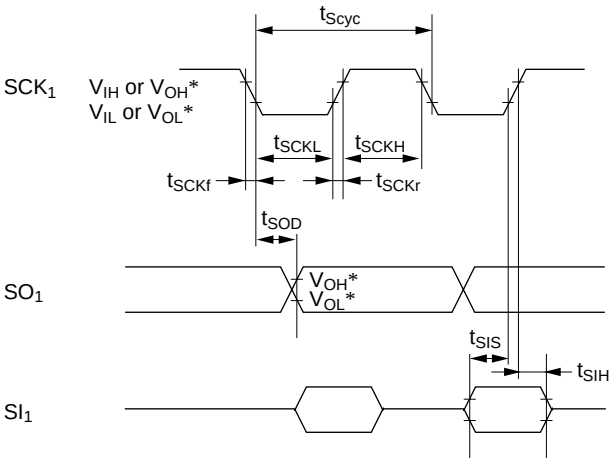


Figure 15.5 UD Pin Minimum Transition Width Timing



Note: * Output timing reference levels

Output high	V_{OH}= 2.0 V
Output low	V_{OL}= 0.8 V

Load conditions are shown in figure 15.10.

Figure 15.6 SCI1 Input/Output Timing

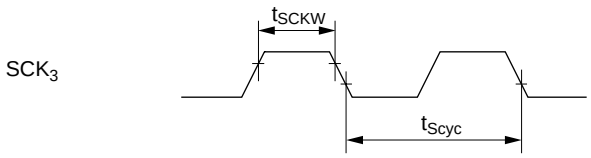
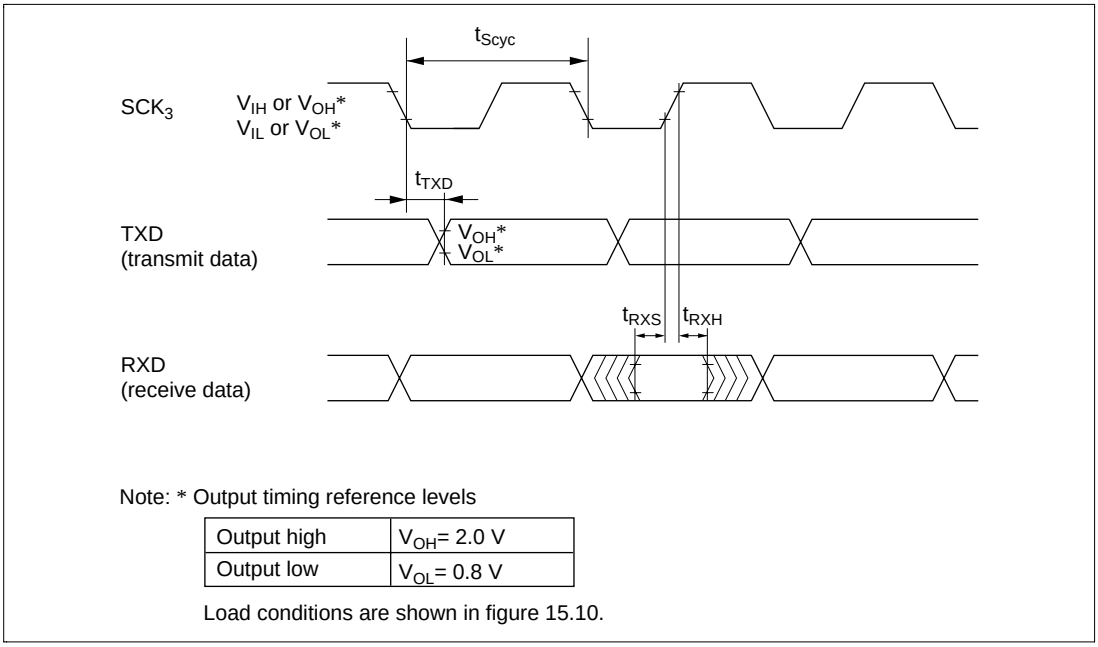


Figure 15.7 SCK3 Input Clock Timing



Note: * Output timing reference levels

Output high	V_{OH}= 2.0 V
Output low	V_{OL}= 0.8 V

Load conditions are shown in figure 15.10.

Figure 15.8 SCI3 Input/Output Timing in Synchronous Mode

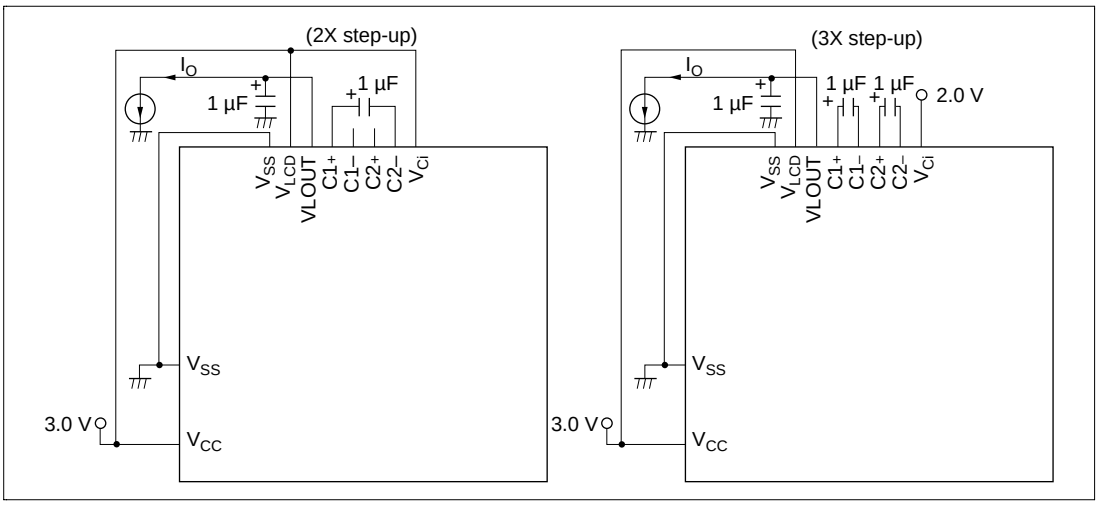


Figure 15.9 Step-Up Circuit Characteristics Test Circuits

15.4 Output Load Circuit

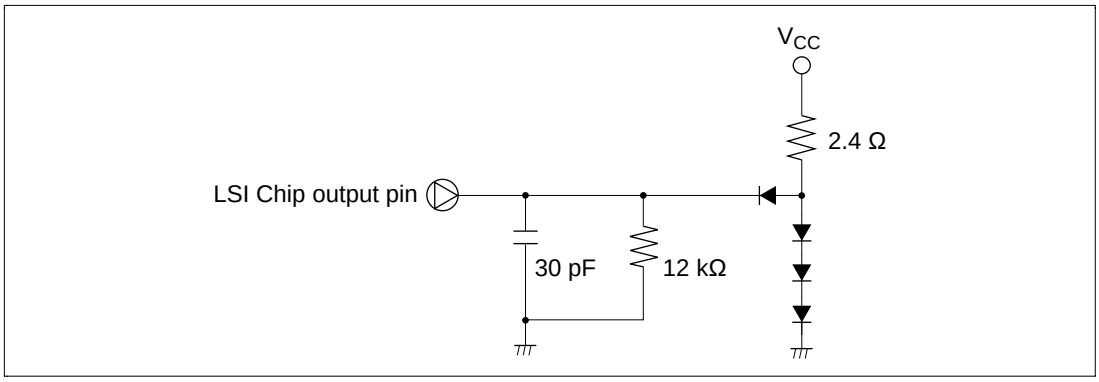


Figure 15.10 Output Load Conditions

15.5 Usage Note

Although both the F-ZTAT and mask ROM versions fully meet the electrical specifications listed in this manual, there may be differences in the actual values of the electrical characteristics, operating margins, noise margins, and so forth, due to differences in the fabrication process, the on-chip ROM, and the layout patterns.

If the F-ZTAT version is used to carry out system evaluation and testing, therefore, when switching to the mask ROM version the same evaluation and testing procedures should also be conducted on the mask ROM version.

Section 16 Electrical Characteristics (H8/3854 Series)

16.1 H8/3852, H8/3853, and H8/3854 Absolute Maximum Ratings (Standard Specifications)

Table 16.1 shows the absolute maximum ratings.

Table 16.1 Absolute Maximum Ratings

Item		Symbol	Value	Unit	Notes
Power supply voltage		V_{CC}	–0.3 to +7.0	V	
Programming voltage (FWE)		V_{in}	–0.3 to V_{CC}+0.3	V	*1
Input voltage	Except LCD power supply	V_{in}	–0.3 to V_{CC}+0.3	V	
	LCD power supply	V_{in}	–0.3 to V_{CC}+0.3	V	*2
Operating temperature		T_{opr}	–20 to +75	°C	*3
Storage temperature		T_{stg}	–55 to +125	°C	

Caution: Permanent damage may occur to the chip if maximum ratings are exceeded. Normal operation should be under the conditions specified in Electrical Characteristics. Exceeding these values can result in incorrect operation and reduced reliability.

Notes: 1. 12 V must not be applied to the FWE pin, as this will permanently damage the device.

2. When the internal power supply and internal bleeder resistances are not used, and the LCD drive voltages are supplied directly from an external source, this applies to V1OUT, V2OUT, V3OUT, V4OUT, and V5OUT.

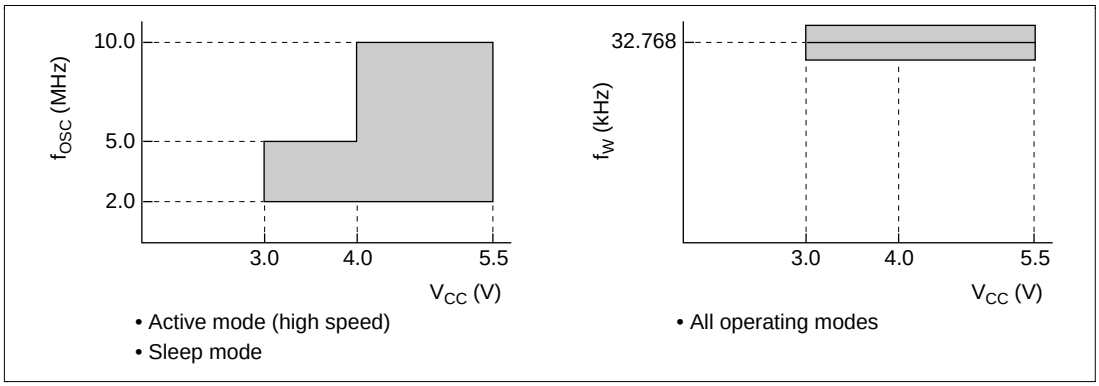
3. The operating temperature range when programming/erasing flash memory is: T_a = 0°C to +75°C.

16.2 H8/3852, H8/3853, and H8/3854 Electrical Characteristics (Standard Specifications)

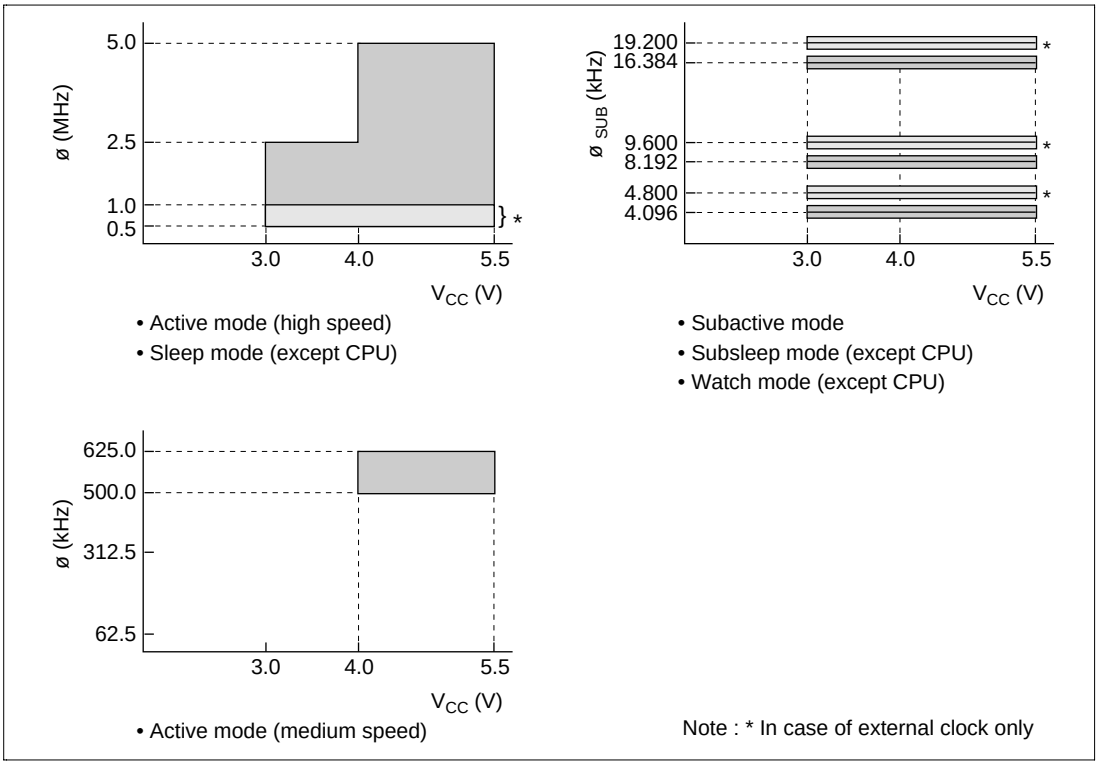
16.2.1 Power Supply Voltage and Operating Range

The power supply voltage and operating range of the H8/3852, H8/3853, and H8/3854 are indicated by the shaded region in the figures below.

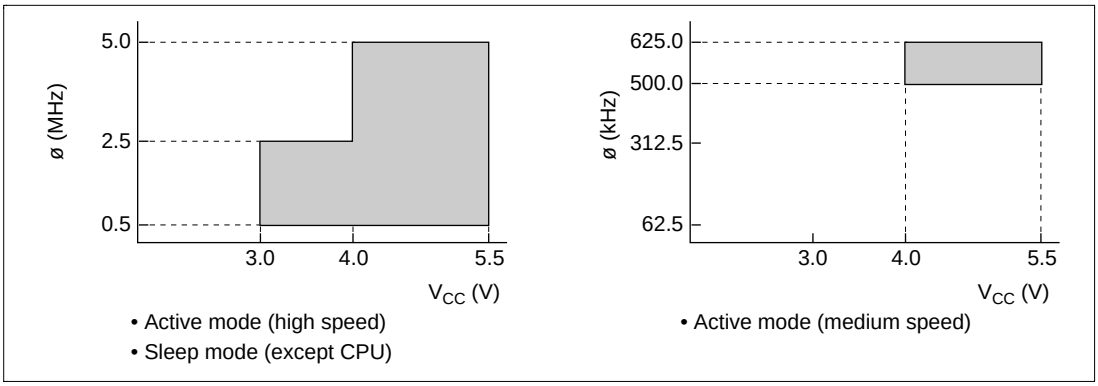
(1) Power Supply Voltage vs. Oscillator Frequency Range



(2) Power Supply Voltage vs. Operating Frequency Range



(3) Power Supply Voltage vs. A/D Converter Operating Range



16.2.2 DC Characteristics

Table 16.2 shows the DC characteristics of the H8/3852, H8/3853, and H8/3854.

Table 16.2 DC Characteristics of H8/3852, H8/3853, and H8/3854 (1)

V_{CC} = 3.0 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20°C to +75°C*[4], including subactive mode, unless otherwise specified.

				Values				
Item	**Symbol**	**Applicable Pins**	**Test Conditions**	**Min**	**Typ**	**Max**	**Unit**	**Notes**
Input high voltage	V_{IH}	$\overline{RES}$, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{IRQ_0}$, $\overline{IRQ_1}$, $\overline{IRQ_3}$, $\overline{IRQ_4}$, TMIB, TMIF, TEST2, FWE, SCK_3, $\overline{ADTRG}$	V_{CC} = 4.0 V to 5.5 V	0.8V_{CC}	—	V_{CC}+0.3	V	
				0.9V_{CC}	—	V_{CC}+0.3		
		RXD	V_{CC} = 4.0 V to 5.5 V	0.7V_{CC}	—	V_{CC}+0.3	V	
				0.8V_{CC}	—	V_{CC}+0.3		
		OSC_1	V_{CC} = 4.0 V to 5.5 V	V_{CC}–0.5	—	V_{CC}+0.3	V	
				V_{CC}–0.3	—	V_{CC}+0.3		
		X1		V_{CC}–0.3	—	V_{CC}+0.3	V	
		$P1_0$ to $P1_2$, $P1_5$, $P1_7$, $P2_0$ to $P2_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$, PB_4 to PB_7	V_{CC} = 4.0 V to 5.5 V	0.7V_{CC}	—	V_{CC}+0.3	V	
				0.8V_{CC}	—	V_{CC}+0.3		
Input low voltage	V_{IL}	$\overline{RES}$, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{IRQ_0}$, $\overline{IRQ_1}$, $\overline{IRQ_3}$, $\overline{IRQ_4}$, TMIB, TMIF, TEST2, FWE, SCK_3, $\overline{ADTRG}$	V_{CC} = 4.0 V to 5.5 V	–0.3	—	0.2V_{CC}	V	
				–0.3	—	0.1V_{CC}		
		RXD	V_{CC} = 4.0 V to 5.5 V	–0.3	—	0.3V_{CC}	V	
				–0.3	—	0.2V_{CC}		
		OSC_1	V_{CC} = 4.0 V to 5.5 V	–0.3	—	0.5	V	
				–0.3	—	0.3		

Item	Symbol	Applicable Pins	Test Conditions	Values Min	Typ	Max	Unit	Notes
Input low voltage	V_{IL}	X1		–0.3	—	0.3	V	
		$P1_0$ to $P1_2$, $P1_5$, $P1_7$, $P2_0$ to $P2_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$, PB_4 to PB_7	V_{CC} = 4.0 V to 5.5 V	–0.3	—	$0.3V_{CC}$	V	
				–0.3	—	$0.2V_{CC}$		
Output high voltage	V_{OH}	$P1_0$ to $P1_2$, $P1_5$, $P1_7$, $P2_0$ to $P2_7$, $P4_0$ to $P4_2$, $P5_0$ to $P5_7$	V_{CC} = 4.0 V to 5.5 V $-I_{OH}$ = 1.0 mA	V_{CC}–1.0	—	—	V	
			V_{CC} = 4.0 V to 5.5 V $-I_{OH}$ = 0.5 mA	V_{CC}–0.5	—	—		
			$-I_{OH}$ = 0.1 mA	V_{CC}–0.5	—	—		
Output low voltage	V_{OL}	$P1_0$ to $P1_2$, $P1_5$, $P1_7$, $P4_0$ to $P4_2$, $P5_0$ to $P5_7$	V_{CC} = 4.0 V to 5.5 V I_{OL}= 1.6 mA	—	—	0.6	V	
			I_{OL}= 0.4 mA	—	—	0.5		
		$P2_0$ to $P2_7$	V_{CC} = 4.0 V to 5.5 V I_{OL}= 10 mA	—	—	1.5		
			V_{CC} = 4.0 V to 5.5 V I_{OL}= 1.6 mA	—	—	0.6		
			I_{OL}= 0.4 mA	—	—	0.5		
Input/output leakage current	$\lvert I_{IL} \rvert$	$\overline{RES}$, TEST2, FWE, OSC_1, $P1_0$ to $P1_2$, $P1_5$, $P1_7$, $P2_0$ to $P2_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$, PB_4 to PB_7	V_{in} = 0.5 V to V_{CC} – 0.5 V	—	—	1.0	µA	
Pull-up MOS current	$-I_p$	$P1_0$ to $P1_2$, $P1_5$, $P1_7$, $P5_0$ to $P5_7$	V_{CC} = 5 V, V_{in}= 0 V	50.0	—	300.0	µA	
			V_{CC} = 3.3 V, V_{in} = 0 V	—	100	—	µA	Reference values
Input capacitance	C_{in}	All input pins except power supply pins	f = 1 MHz, V_{in} = 0 V, T_a = 25°C	—	—	15.0	pF	

Item	Symbol	Applicable Pins	Test Conditions	Values			Unit	Notes
				Min	Typ	Max		
Active mode current dissipation	I_{OPE1}	V_{CC}	Active mode (high speed) V_{CC} = 5 V, f_{OSC} = 10 MHz A/D not used	—	10.0	15.0	mA	*1 *2
	I_{OPE3}	V_{CC}	Active mode (high speed) V_{CC} = 5 V, f_{OSC} = 10 MHz A/D operating	—	—	16.5	mA	*1 *2
	I_{OPE2}	V_{CC}	Active mode (medium speed) V_{CC} = 5 V, f_{OSC} = 10 MHz A/D not used	—	2.0	3.5	mA	*1 *2
Sleep mode current dissipation	I_{SLEEP}	V_{CC}	V_{CC} = 5 V, f_{OSC} = 10 MHz A/D not used	—	4.3	7.0	mA	*1 *2
Subactive mode current dissipation	I_{SUB}	V_{CC}	V_{CC} = 5.0 V, LCD on, 32-kHz crystal oscillator used ($ø_{SUB}$= $ø_W$/2)	—	80	160	µA	*1 *2 *5
			V_{CC} = 5.0 V, LCD on, 32-kHz crystal oscillator used ($ø_{SUB}$= $ø_W$/8)	—	70	—	µA	*1 *2 *5 Reference values
			V_{CC} = 3.3 V, LCD not used, 32-kHz crystal oscillator used ($ø_{SUB}$= $ø_W$/2)	—	20	—	µA	*1 *2 Reference values
Subsleep mode current dissipation	I_{SUBSP}	V_{CC}	V_{CC} = 5.0 V, LCD on, 32-kHz crystal oscillator used ($ø_{SUB}$= $ø_W$/2)	—	50	100	µA	*1 *2 *5
Watch mode current dissipation	I_{WATCH}	V_{CC}	V_{CC} = 5.0 V, LCD on, 32-kHz crystal oscillator used	—	40	80	µA	*1 *2 *5
			V_{CC} = 3.3 V, LCD not used, 32-kHz crystal oscillator used	—	7.0	15.0	µA	*1 *2
Standby mode current dissipation	I_{STBY}	V_{CC}	32-kHz crystal oscillator not used	—	—	5.0	µA	*1 *2

Item	Symbol	Applicable Pins	Test Conditions	Values Min	Typ	Max	Unit	Notes
Program/ erase current dissipation	I_{FLASH}	V_{CC}	0°C ≤ T_a ≤ 70°C f_{OSC} = 12 MHz	—	16	22	mA	*1 *2 *3
RAM data retaining voltage	V_{RAM}	V_{CC}		2.0	—	—	V	*1 *2

Notes: 1. Pin states during current measurement

Mode	Internal State	Pins	Oscillator Pins
Active mode (high and medium speed)	Operates	V_{CC}	System clock oscillator: Crystal Subclock oscillator: Pin X_1 = V_{CC}
Sleep mode	Only timer operates	V_{CC}	
Subactive mode	Operates	V_{CC}	System clock oscillator: Crystal Subclock oscillator: Crystal
Subsleep mode	Only timer operates, CPU stops	V_{CC}	
Watch mode	Only time-base clock operates, CPU stops	V_{CC}	
Standby mode	CPU and timers all stop	V_{CC}	System clock oscillator: Crystal Subclock oscillator: Pin X_1 = V_{CC}
Programming/ erasing*3	Operates	V_{CC}	System clock oscillator: Crystal Subclock oscillator: Pin X_1 = V_{CC}

2. Excludes current in pull-up MOS transistors and output buffers.
3. Applies to F-ZTAT version only.
4. The guaranteed temperature as an electrical characteristic for shipped products is 75°C.
5. When power is supplied to the built-in bleeder resistances from V_{CC} (LPS0 = LPS1 = 1 in LR2).

Table 16.3 DC Characteristics of H8/3852, H8/3853, and H8/3854 (2)

V_{CC} = 3.0 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20°C to +75°C*[2], including subactive mode, unless otherwise specified.

				Values				
Item	**Symbol**	**Applicable Pins**	**Test Conditions**	**Min**	**Typ**	**Max**	**Unit**	**Notes**
Allowable output low current (per pin)	I_{OL}	Output pins except in port 2	V_{CC} = 4.0 V to 5.5 V	—	—	2.0	mA	*[1]
		Port 2	V_{CC} = 4.0 V to 5.5 V	—	—	10.0		
		All output pins		—	—	0.5		
Allowable output low current (total)	ΣI_{OL}	Output pins except in port 2	V_{CC} = 4.0 V to 5.5 V	—	—	20.0	mA	*[1]
		Port 2	V_{CC} = 4.0 V to 5.5 V	—	—	80.0		
		All output pins		—	—	20.0		
Allowable output high current (per pin)	$-I_{OH}$	All output pins	V_{CC} = 4.0 V to 5.5 V	—	—	2.0	mA	*[1]
				—	—	0.2		
Allowable output high current (total)	$\Sigma -I_{OH}$	All output pins	V_{CC} = 4.0 V to 5.5 V	—	—	10.0	mA	*[1]
				—	—	8.0		

Notes: 1. Excludes LCD output pins.
2. The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

16.2.3 AC Characteristics

Table 16.4 shows the control signal timing, and tables 16.5 and 16.6 show the serial interface timing, of the H8/3852, H8/3853, and H8/3854.

Table 16.4 Control Signal Timing of H8/3852, H8/3853, and H8/3854

V_{CC} = 3.0 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20°C to +75°C*[3], including subactive mode, unless otherwise specified.

Item	Symbol	Applicable Pins	Test Conditions	Values Min	Typ	Max	Unit	Reference Figure
System clock oscillation frequency	f_{OSC}	OSC1, OSC2	V_{CC} = 4.0 V to 5.5 V	2.0	—	10.0	MHz	
				2.0	—	5.0		
OSC clock ($ø_{OSC}$) cycle time	t_{OSC}	OSC1, OSC2	V_{CC} = 4.0 V to 5.5 V	100.0	—	1000.0	ns	*1
				200.0	—	1000.0		Figure16.1
System clock (ø) cycle time	t_{cyc}			2	—	16	t_{OSC}	*1
				—	—	2000.0	ns	
Subclock oscillation frequency	f_W	X1, X2		—	32.768	—	kHz	
Watch clock ($ø_W$) cycle time	t_W	X1, X2		—	30.5	—	µs	
Subclock ($ø_{SUB}$) cycle time	t_{subcyc}			2	—	8	t_W	*2
Instruction cycle time				2	—	—	t_{cyc} t_{subcyc}	
Oscillation stabilization time (crystal oscillator)	t_{rc}	OSC_1, OSC_2	V_{CC} = 4.0 V to 5.5 V	40.0	—	—	ms	
				60.0	—	—		
Oscillation stabilization time	t_{rc}	X_1, X_2		2	—	—	s	
External clock high width	t_{CPH}	OSC_1	V_{CC} = 4.0 V to 5.5 V	40.0	—	—	ns	Figure 16.1
				80.0	—	—		
External clock low width	t_{CPL}	OSC_1	V_{CC} = 4.0 V to 5.5 V	40.0	—	—	ns	Figure 16.1
				80.0	—	—		
External clock rise time	t_{CPr}	OSC_1	V_{CC} = 4.0 V to 5.5 V	—	—	15.0	ns	Figure 16.1
				—	—	20.0		
External clock fall time	t_{CPf}	OSC_1	V_{CC} = 4.0 V to 5.5 V	—	—	15.0	ns	Figure 16.1
				—	—	20.0		

Item	Symbol	Applicable Pins	Test Conditions	Values Min	Typ	Max	Unit	Reference Figure
External subclock high width	t_{XH}	X_1		0.4/fx	—	—	s	Figure 16.2
External subclock low width	t_{XL}	X_1		0.4/fx	—	—	s	Figure 16.2
External subclock rise time	t_{Xr}	X_1		—	—	100.0	ns	Figure 16.2
External subclock fall time	t_{Xf}	X_1		—	—	100.0	ns	Figure 16.2
$\overline{RES}$ pin low width	t_{REL}	$\overline{RES}$		10	—	—	t_{cyc}	Figure 16.3
Input pin high width	t_{IH}	$\overline{IRQ_0}$, $\overline{IRQ_1}$, $\overline{IRQ_3}$, $\overline{IRQ_4}$, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{ADTRG}$, TMIB, TMIF		2	—	—	t_{cyc} t_{subcyc}	Figure 16.4
Input pin low width	t_{IL}	$\overline{IRQ_0}$, $\overline{IRQ_1}$, $\overline{IRQ_3}$, $\overline{IRQ_4}$, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{ADTRG}$, TMIB, TMIF		2	—	—	t_{cyc} t_{subcyc}	Figure 16.4

Notes: 1. A frequency between 1 MHz and 10 MHz is required when an external clock is input.
2. Selected with bits SA1 and SA0 in system control register 2 (SYSCR2).
3. The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

Table 16.5 Serial Interface (SCI3) Timing of H8/3852, H8/3853, and H8/3854

V_{CC} = 3.0 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20°C to +75°C*, unless otherwise specified.

Item		Symbol	Test Conditions	Values Min	Typ	Max	Unit	Reference Figure
Input clock cycle	Asynchronous	t_{Scyc}		4	—	—	t_{cyc}	Figure 16.5
	Synchronous			6	—	—		
Input clock pulse width		t_{SCKW}		0.4	—	0.6	t_{Scyc}	Figure 16.5
Transmit data delay time (synchronous mode)		t_{TXD}	V_{CC} = 4.0 V to 5.5 V	—	—	1	t_{cyc}	Figure 16.6
				—	—	1		
Receive data setup time (synchronous mode)		t_{RXS}	V_{CC} = 4.0 V to 5.5 V	200.0	—	—	ns	Figure 16.6
				400.0	—	—		
Receive data hold time (synchronous mode)		t_{RXH}	V_{CC} = 4.0 V to 5.5 V	200.0	—	—	ns	Figure 16.6
				400.0	—	—		

Note: * The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

16.2.4 A/D Converter Characteristics

Table 16.6 shows the A/D converter characteristics of the H8/3852, H8/3853, and H8/3854.

Table 16.6 A/D Converter Characteristics of H8/3852, H8/3853, and H8/3854

V_{CC} = 3.0 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20°C to +75°C*, unless otherwise specified.

Item	Symbol	Applicable Pins	Test Conditions	Values			Unit	Reference Figure
				Min	Typ	Max		
Analog input voltage	AV_{IN}	AN_4 to AN_7		V_{SS} – 0.3	—	V_{CC} + 0.3	V	
Analog input capacitance	C_{AIN}	AN_4 to AN_7		—	—	30.0	pF	
Allowable signal source impedance	R_{AIN}			—	—	5.0	kΩ	
Resolution (data length)				—	—	8	Bit	
Non-linearity error				—	—	±2.0	LSB	
Quantization error				—	—	±0.5	LSB	
Absolute accuracy				—	—	±2.5	LSB	
Conversion time				12.4	—	124	µs	

Note: * The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

16.2.5 LCD Characteristics

Table 16.7 shows the LCD characteristics of the H8/3852, H8/3853, and H8/3854.

Table 16.7 LCD Characteristics of H8/3852, H8/3853, and H8/3854

V_{CC} = 3.0 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20°C to +75°C*[2], including subactive mode, unless otherwise specified.

				Values				
Item	**Symbol**	**Applicable Pins**	**Test Conditions**	**Min**	**Typ**	**Max**	**Unit**	**Notes**
Common driver on-resistance	R_{COM}	COM1 to COM16	±Id = 0.05 mA, V_{CC} = 4 V	—	6	20	kΩ	*[1]
Segment driver on-resistance	R_{SEG}	SEG1 to SEG40	±Id = 0.05 mA, V_{CC} = 4 V	—	6	20	kΩ	*[1]
LCD power supply bleeder resistance	R_{LCD}		V_{CC} = 5.0 V, f_x = 32.768 kHz	200	400	700	kΩ	

Notes: 1. Applies to the resistance (R_{COM}) between the V1OUT, V2OUT, V5OUT, and V_{SS} pins and the common signal pins (COM1 to COM16), and the resistance (R_{SEG}) between the V1OUT, V3OUT, V4OUT, and V_{SS} pins and the segment signal pins (SEG1 to SEG40), when Id is flowing in the pins.

The voltage applied to V1OUT through V5OUT must not exceed V_{CC}.

2. The guaranteed temperature as an electrical characteristic for shipped products is 75°C.

16.2.6 Flash Memory Characteristics

Table 16.8 shows the flash memory characteristics.

Table 16.8 Flash Memory Characteristics

Conditions: V_{CC} = 4.5 V to 5.5 V, V_{SS} = 0.0 V, T_a = 0°C to +75°C (program/erase operating temperature range)

Item		Symbol	Min	Typ	Max	Unit	Test Conditions
Programming time*1, *2, *4		t_P	—	10	200	ms/32 bytes	
Erase time*1, *3, *5		t_E	—	100	300	ms/block	
Rewrite times		N_{WEC}	—	—	100	Times	
Programming	Wait time after SWE bit setting*1	x	10	—	—	µs	
	Wait time after PSU bit setting*1	y	50	—	—	µs	
	Wait time after P bit setting*1, *4	z	—	—	200	µs	
	Wait time after P bit clearing*1	α	10	—	—	µs	
	Wait time after PSU bit clearing*1	β	10	—	—	µs	
	Wait time after PV bit setting*1	γ	4	—	—	µs	
	Wait time after H'FF dummy write*1	ε	2	—	—	µs	
	Wait time after PV bit clearing*1	η	4	—	—	µs	
	Maximum number of writes*1, *4	N	—	—	1000	Times	
Erasing	Wait time after SWE bit setting*1	x	10	—	—	µs	
	Wait time after ESU bit setting*1	y	200	—	—	µs	
	Wait time after E bit setting*1, *5	z	—	—	5	ms	
	Wait time after E bit clearing*1	α	10	—	—	µs	
	Wait time after ESU bit clearing*1	β	10	—	—	µs	
	Wait time after EV bit setting*1	γ	20	—	—	µs	
	Wait time after H'FF dummy write*1	ε	2	—	—	µs	
	Wait time after EV bit clearing*1	η	5	—	—	µs	
	Maximum number of erases*1, *5	N	—	—	60	Times	

Notes:
1. Follow the program/erase algorithms when making the time settings.
2. Programming time per 32 bytes. (Indicates the total time during which the P bit is set in flash memory control register 1 (FLMCR1). Does not include the program-verify time.)
3. Time to erase one block. (Indicates the time during which the E bit is set in FLMCR1. Does not include the erase-verify time.)
4. Maximum programming time
   (t_P(max) = Wait time after P bit setting (z) × maximum number of writes (N))
5. Maximum erase time
   (t_E(max) = Wait time after E bit setting (z) × maximum number of erases (N))

16.3 Operation Timing

Figures 16.1 to 16.6 show timing diagrams.

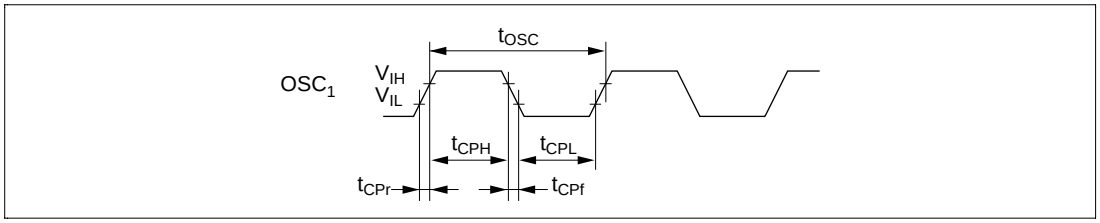


Figure 16.1 System Clock Input Timing

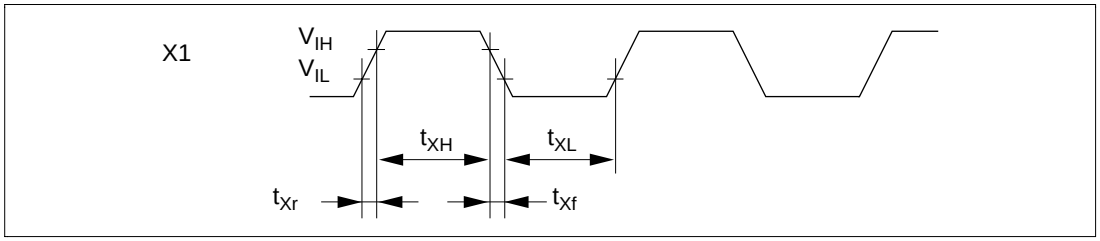


Figure 16.2 Subclock Input Timing

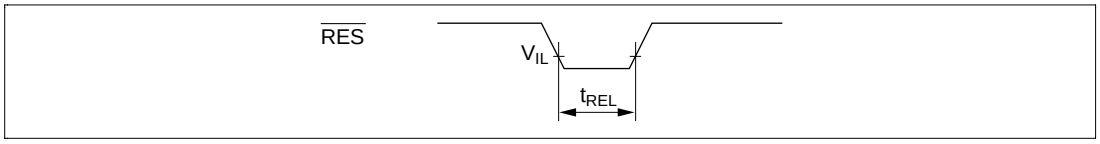


Figure 16.3 $\overline{RES}$ Pin Low Width Timing

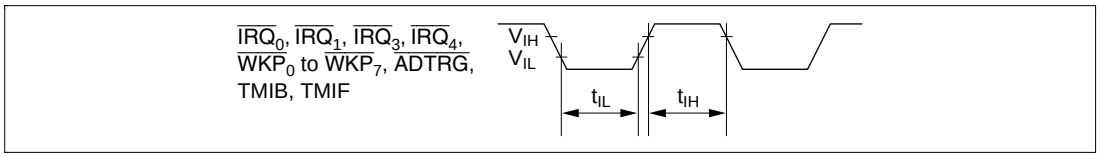


Figure 16.4 Input Timing

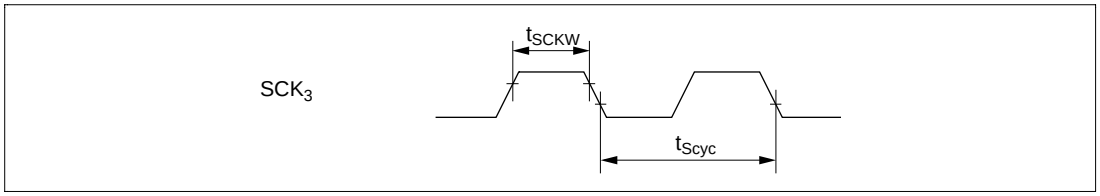
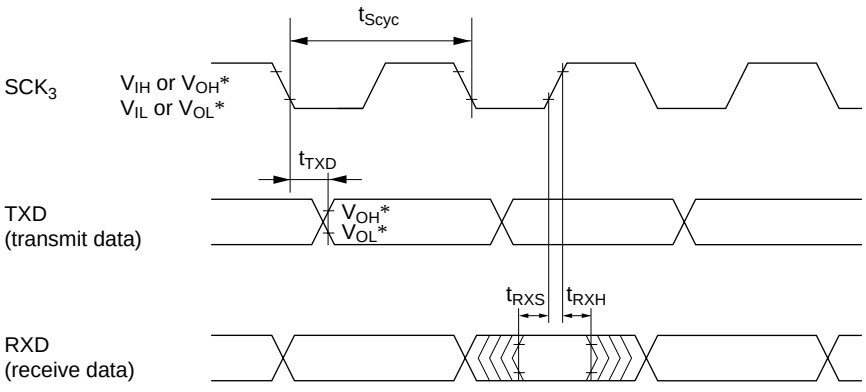


Figure 16.5 SCK3 Input Clock Timing



Note: * Output timing reference levels

Output high	V_{OH}= 2.0 V
Output low	V_{OL}= 0.8 V

Load conditions are shown in figure 16.7.

Figure 16.6 SCK3 Input/Output Timing in Synchronous Mode

16.4 Output Load Circuit

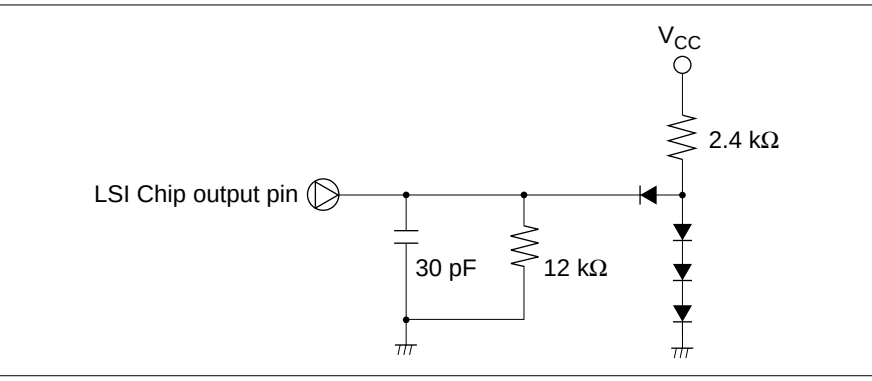


Figure 16.7 Output Load Conditions

16.5 Usage Note

Although both the F-ZTAT and mask ROM versions fully meet the electrical specifications listed in this manual, there may be differences in the actual values of the electrical characteristics, operating margins, noise margins, and so forth, due to differences in the fabrication process, the on-chip ROM, and the layout patterns.

If the F-ZTAT version is used to carry out system evaluation and testing, therefore, when switching to the mask ROM version the same evaluation and testing procedures should also be conducted on the mask ROM version.

Appendix A CPU Instruction Set

A.1 Instructions

Operation Notation

Rd8/16	General register (destination) (8 or 16 bits)
Rs8/16	General register (source) (8 or 16 bits)
Rn8/16	General register (8 or 16 bits)
CCR	Condition code register
N	N (negative) flag in CCR
Z	Z (zero) flag in CCR
V	V (overflow) flag in CCR
C	C (carry) flag in CCR
PC	Program counter
SP	Stack pointer
#xx: 3/8/16	Immediate data (3, 8, or 16 bits)
d: 8/16	Displacement (8 or 16 bits)
@aa: 8/16	Absolute address (8 or 16 bits)
+	Addition
–	Subtraction
×	Multiplication
÷	Division
∧	Logical AND
∨	Logical OR
⊕	Exclusive logical OR
→	Move
¯	Logical complement

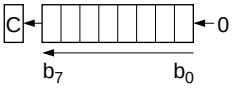
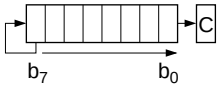
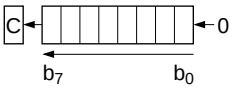
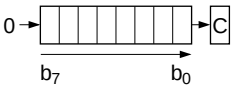
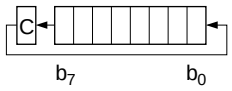
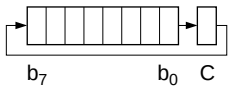
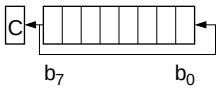
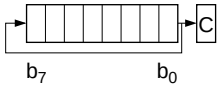
Condition Code Notation

Symbol	
↕	Modified according to the instruction result
*	Not fixed (value not guaranteed)
0	Always cleared to 0
—	Not affected by the instruction execution result

Table A.1 Instruction Set

Mnemonic	Operand Size	Operation	Addressing Mode/ Instruction Length (Bytes)									Condition Code						No. of States
			#xx: 8/16	Rn	@Rn	@(d:16, Rn)	@–Rn/@Rn+	@aa: 8/16	@(d:8, PC)	@@aa	Implied	I	H	N	Z	V	C	
MOV.B #xx:8, Rd	B	#xx:8 → Rd8	2									—	—	↕	↕	0	—	2
MOV.B Rs, Rd	B	Rs8 → Rd8		2								—	—	↕	↕	0	—	2
MOV.B @Rs, Rd	B	@Rs16 → Rd8			2							—	—	↕	↕	0	—	4
MOV.B @(d:16, Rs), Rd	B	@(d:16, Rs16)→ Rd8				4						—	—	↕	↕	0	—	6
MOV.B @Rs+, Rd	B	@Rs16 → Rd8 Rs16+1 → Rs16					2					—	—	↕	↕	0	—	6
MOV.B @aa:8, Rd	B	@aa:8 → Rd8						2				—	—	↕	↕	0	—	4
MOV.B @aa:16, Rd	B	@aa:16 → Rd8						4				—	—	↕	↕	0	—	6
MOV.B Rs, @Rd	B	Rs8 → @Rd16			2							—	—	↕	↕	0	—	4
MOV.B Rs, @(d:16, Rd)	B	Rs8 → @(d:16, Rd16)				4						—	—	↕	↕	0	—	6
MOV.B Rs, @–Rd	B	Rd16–1 → Rd16 Rs8 → @Rd16					2					—	—	↕	↕	0	—	6
MOV.B Rs, @aa:8	B	Rs8 → @aa:8						2				—	—	↕	↕	0	—	4
MOV.B Rs, @aa:16	B	Rs8 → @aa:16						4				—	—	↕	↕	0	—	6
MOV.W #xx:16, Rd	W	#xx:16 → Rd	4									—	—	↕	↕	0	—	4
MOV.W Rs, Rd	W	Rs16 → Rd16		2								—	—	↕	↕	0	—	2
MOV.W @Rs, Rd	W	@Rs16 → Rd16			2							—	—	↕	↕	0	—	4
MOV.W @(d:16, Rs), Rd	W	@(d:16, Rs16) → Rd16				4						—	—	↕	↕	0	—	6
MOV.W @Rs+, Rd	W	@Rs16 → Rd16 Rs16+2 → Rs16					2					—	—	↕	↕	0	—	6
MOV.W @aa:16, Rd	W	@aa:16 → Rd16						4				—	—	↕	↕	0	—	6
MOV.W Rs, @Rd	W	Rs16 → @Rd16			2							—	—	↕	↕	0	—	4
MOV.W Rs, @(d:16, Rd)	W	Rs16 → @(d:16, Rd16)				4						—	—	↕	↕	0	—	6
MOV.W Rs, @–Rd	W	Rd16–2 → Rd16 Rs16 → @Rd16					2					—	—	↕	↕	0	—	6
MOV.W Rs, @aa:16	W	Rs16 → @aa:16						4				—	—	↕	↕	0	—	6
POP Rd	W	@SP → Rd16 SP+2 → SP					2					—	—	↕	↕	0	—	6

Mnemonic	Operand Size	Operation	Addressing Mode/ Instruction Length (Bytes)									Condition Code						No. of States
			#xx: 8/16	Rn	@Rn	@(d:16, Rn)	@–Rn/@Rn+	@aa: 8/16	@(d:8, PC)	@@aa	Implied	I	H	N	Z	V	C	
PUSH Rs	W	SP–2 → SP Rs16 → @SP					2					—	—	↕	↕	0	—	6
ADD.B #xx:8, Rd	B	Rd8+#xx:8 → Rd8	2									—	↕	↕	↕	↕	↕	2
ADD.B Rs, Rd	B	Rd8+Rs8 → Rd8		2								—	↕	↕	↕	↕	↕	2
ADD.W Rs, Rd	W	Rd16+Rs16 → Rd16		2								—	(1)	↕	↕	↕	↕	2
ADDX.B #xx:8, Rd	B	Rd8+#xx:8 +C → Rd8	2									—	↕	↕	(2)	↕	↕	2
ADDX.B Rs, Rd	B	Rd8+Rs8 +C → Rd8		2								—	↕	↕	(2)	↕	↕	2
ADDS.W #1, Rd	W	Rd16+1 → Rd16		2								—	—	—	—	—	—	2
ADDS.W #2, Rd	W	Rd16+2 → Rd16		2								—	—	—	—	—	—	2
INC.B Rd	B	Rd8+1 → Rd8		2								—	—	↕	↕	↕	—	2
DAA.B Rd	B	Rd8 decimal adjust → Rd8		2								—	*	↕	↕	*	(3)	2
SUB.B Rs, Rd	B	Rd8–Rs8 → Rd8		2								—	↕	↕	↕	↕	↕	2
SUB.W Rs, Rd	W	Rd16–Rs16 → Rd16		2								—	(1)	↕	↕	↕	↕	2
SUBX.B #xx:8, Rd	B	Rd8–#xx:8 –C → Rd8	2									—	↕	↕	(2)	↕	↕	2
SUBX.B Rs, Rd	B	Rd8–Rs8 –C → Rd8		2								—	↕	↕	(2)	↕	↕	2
SUBS.W #1, Rd	W	Rd16–1 → Rd16		2								—	—	—	—	—	—	2
SUBS.W #2, Rd	W	Rd16–2 → Rd16		2								—	—	—	—	—	—	2
DEC.B Rd	B	Rd8–1 → Rd8		2								—	—	↕	↕	↕	—	2
DAS.B Rd	B	Rd8 decimal adjust → Rd8		2								—	*	↕	↕	*	—	2
NEG.B Rd	B	0–Rd → Rd		2								—	↕	↕	↕	↕	↕	2
CMP.B #xx:8, Rd	B	Rd8–#xx:8	2									—	↕	↕	↕	↕	↕	2
CMP.B Rs, Rd	B	Rd8–Rs8		2								—	↕	↕	↕	↕	↕	2
CMP.W Rs, Rd	W	Rd16–Rs16		2								—	(1)	↕	↕	↕	↕	2
MULXU.B Rs, Rd	B	Rd8 × Rs8 → Rd16		2								—	—	—	—	—	—	14

Mnemonic	Operand Size	Operation	Addressing Mode/ Instruction Length (Bytes)									Condition Code						No. of States
			#xx: 8/16	Rn	@Rn	@(d:16, Rn)	@–Rn/@Rn+	@aa: 8/16	@(d:8, PC)	@@aa	Implied	I	H	N	Z	V	C	
DIVXU.B Rs, Rd	B	Rd16÷Rs8 → Rd16 (RdH: remainder, RdL: quotient)		2								—	—	(5)	(6)	—	—	14
AND.B #xx:8, Rd	B	Rd8∧#xx:8 → Rd8	2									—	—	↕	↕	0	—	2
AND.B Rs, Rd	B	Rd8∧Rs8 → Rd8		2								—	—	↕	↕	0	—	2
OR.B #xx:8, Rd	B	Rd8∨#xx:8 → Rd8	2									—	—	↕	↕	0	—	2
OR.B Rs, Rd	B	Rd8∨Rs8 → Rd8		2								—	—	↕	↕	0	—	2
XOR.B #xx:8, Rd	B	Rd8⊕#xx:8 → Rd8	2									—	—	↕	↕	0	—	2
XOR.B Rs, Rd	B	Rd8⊕Rs8 → Rd8		2								—	—	↕	↕	0	—	2
NOT.B Rd	B	$\overline{Rd}$ → Rd		2								—	—	↕	↕	0	—	2
SHAL.B Rd	B			2								—	—	↕	↕	↕	↕	2
SHAR.B Rd	B			2								—	—	↕	↕	0	↕	2
SHLL.B Rd	B			2								—	—	↕	↕	0	↕	2
SHLR.B Rd	B			2								—	—	0	↕	0	↕	2
ROTXL.B Rd	B			2								—	—	↕	↕	0	↕	2
ROTXR.B Rd	B			2								—	—	↕	↕	0	↕	2
ROTL.B Rd	B			2								—	—	↕	↕	0	↕	2
ROTR.B Rd	B			2								—	—	↕	↕	0	↕	2

Mnemonic	Operand Size	Operation	Addressing Mode/ Instruction Length (Bytes)									Condition Code						No. of States
			#xx: 8/16	Rn	@Rn	@(d:16, Rn)	@–Rn/@Rn+	@aa: 8/16	@(d:8, PC)	@@aa	Implied	I	H	N	Z	V	C	
BSET #xx:3, Rd	B	(#xx:3 of Rd8) ← 1		2								—	—	—	—	—	—	2
BSET #xx:3, @Rd	B	(#xx:3 of @Rd16) ← 1			4							—	—	—	—	—	—	8
BSET #xx:3, @aa:8	B	(#xx:3 of @aa:8) ← 1						4				—	—	—	—	—	—	8
BSET Rn, Rd	B	(Rn8 of Rd8) ← 1		2								—	—	—	—	—	—	2
BSET Rn, @Rd	B	(Rn8 of @Rd16) ← 1			4							—	—	—	—	—	—	8
BSET Rn, @aa:8	B	(Rn8 of @aa:8) ← 1						4				—	—	—	—	—	—	8
BCLR #xx:3, Rd	B	(#xx:3 of Rd8) ← 0		2								—	—	—	—	—	—	2
BCLR #xx:3, @Rd	B	(#xx:3 of @Rd16) ← 0			4							—	—	—	—	—	—	8
BCLR #xx:3, @aa:8	B	(#xx:3 of @aa:8) ← 0						4				—	—	—	—	—	—	8
BCLR Rn, Rd	B	(Rn8 of Rd8) ← 0		2								—	—	—	—	—	—	2
BCLR Rn, @Rd	B	(Rn8 of @Rd16) ← 0			4							—	—	—	—	—	—	8
BCLR Rn, @aa:8	B	(Rn8 of @aa:8) ← 0						4				—	—	—	—	—	—	8
BNOT #xx:3, Rd	B	(#xx:3 of Rd8) ← ($\overline{\text{\#xx:3 of Rd8}}$)		2								—	—	—	—	—	—	2
BNOT #xx:3, @Rd	B	(#xx:3 of @Rd16) ← ($\overline{\text{\#xx:3 of @Rd16}}$)			4							—	—	—	—	—	—	8
BNOT #xx:3, @aa:8	B	(#xx:3 of @aa:8) ← ($\overline{\text{\#xx:3 of @aa:8}}$)						4				—	—	—	—	—	—	8
BNOT Rn, Rd	B	(Rn8 of Rd8) ← ($\overline{\text{Rn8 of Rd8}}$)		2								—	—	—	—	—	—	2
BNOT Rn, @Rd	B	(Rn8 of @Rd16) ← ($\overline{\text{Rn8 of @Rd16}}$)			4							—	—	—	—	—	—	8
BNOT Rn, @aa:8	B	(Rn8 of @aa:8) ← ($\overline{\text{Rn8 of @aa:8}}$)						4				—	—	—	—	—	—	8
BTST #xx:3, Rd	B	($\overline{\text{\#xx:3 of Rd8}}$) → Z		2								—	—	—	↕	—	—	2
BTST #xx:3, @Rd	B	($\overline{\text{\#xx:3 of @Rd16}}$) → Z			4							—	—	—	↕	—	—	6
BTST #xx:3, @aa:8	B	($\overline{\text{\#xx:3 of @aa:8}}$) → Z						4				—	—	—	↕	—	—	6
BTST Rn, Rd	B	($\overline{\text{Rn8 of Rd8}}$) → Z		2								—	—	—	↕	—	—	2
BTST Rn, @Rd	B	($\overline{\text{Rn8 of @Rd16}}$) → Z			4							—	—	—	↕	—	—	6
BTST Rn, @aa:8	B	($\overline{\text{Rn8 of @aa:8}}$) → Z						4				—	—	—	↕	—	—	6

Mnemonic	Operand Size	Operation	Addressing Mode/ Instruction Length (Bytes)									Condition Code						No. of States
			#xx: 8/16	Rn	@Rn	@(d:16, Rn)	@–Rn/@Rn+	@aa: 8/16	@(d:8, PC)	@@aa	Implied	I	H	N	Z	V	C	
BLD #xx:3, Rd	B	(#xx:3 of Rd8) → C		2								—	—	—	—	—	↕	2
BLD #xx:3, @Rd	B	(#xx:3 of @Rd16) → C			4							—	—	—	—	—	↕	6
BLD #xx:3, @aa:8	B	(#xx:3 of @aa:8) → C						4				—	—	—	—	—	↕	6
BILD #xx:3, Rd	B	($\overline{\text{#xx:3 of Rd8}}$) → C		2								—	—	—	—	—	↕	2
BILD #xx:3, @Rd	B	($\overline{\text{#xx:3 of @Rd16}}$) → C			4							—	—	—	—	—	↕	6
BILD #xx:3, @aa:8	B	($\overline{\text{#xx:3 of @aa:8}}$) → C						4				—	—	—	—	—	↕	6
BST #xx:3, Rd	B	C → (#xx:3 of Rd8)		2								—	—	—	—	—	—	2
BST #xx:3, @Rd	B	C → (#xx:3 of @Rd16)			4							—	—	—	—	—	—	8
BST #xx:3, @aa:8	B	C → (#xx:3 of @aa:8)						4				—	—	—	—	—	—	8
BIST #xx:3, Rd	B	$\overline{\text{C}}$ → (#xx:3 of Rd8)		2								—	—	—	—	—	—	2
BIST #xx:3, @Rd	B	$\overline{\text{C}}$ → (#xx:3 of @Rd16)			4							—	—	—	—	—	—	8
BIST #xx:3, @aa:8	B	$\overline{\text{C}}$ → (#xx:3 of @aa:8)						4				—	—	—	—	—	—	8
BAND #xx:3, Rd	B	C∧(#xx:3 of Rd8) → C		2								—	—	—	—	—	↕	2
BAND #xx:3, @Rd	B	C∧(#xx:3 of @Rd16) → C			4							—	—	—	—	—	↕	6
BAND #xx:3, @aa:8	B	C∧(#xx:3 of @aa:8) → C						4				—	—	—	—	—	↕	6
BIAND #xx:3, Rd	B	C∧($\overline{\text{#xx:3 of Rd8}}$) → C		2								—	—	—	—	—	↕	2
BIAND #xx:3, @Rd	B	C∧($\overline{\text{#xx:3 of @Rd16}}$) → C			4							—	—	—	—	—	↕	6
BIAND #xx:3, @aa:8	B	C∧($\overline{\text{#xx:3 of @aa:8}}$) → C						4				—	—	—	—	—	↕	6
BOR #xx:3, Rd	B	C∨(#xx:3 of Rd8) → C		2								—	—	—	—	—	↕	2
BOR #xx:3, @Rd	B	C∨(#xx:3 of @Rd16) → C			4							—	—	—	—	—	↕	6
BOR #xx:3, @aa:8	B	C∨(#xx:3 of @aa:8) → C						4				—	—	—	—	—	↕	6
BIOR #xx:3, Rd	B	C∨($\overline{\text{#xx:3 of Rd8}}$) → C		2								—	—	—	—	—	↕	2
BIOR #xx:3, @Rd	B	C∨($\overline{\text{#xx:3 of @Rd16}}$) → C			4							—	—	—	—	—	↕	6
BIOR #xx:3, @aa:8	B	C∨($\overline{\text{#xx:3 of @aa:8}}$) → C						4				—	—	—	—	—	↕	6
BXOR #xx:3, Rd	B	C⊕(#xx:3 of Rd8) → C		2								—	—	—	—	—	↕	2
BXOR #xx:3, @Rd	B	C⊕(#xx:3 of @Rd16) → C			4							—	—	—	—	—	↕	6
BXOR #xx:3, @aa:8	B	C⊕(#xx:3 of @aa:8) → C						4				—	—	—	—	—	↕	6
BIXOR #xx:3, Rd	B	C⊕($\overline{\text{#xx:3 of Rd8}}$) → C		2								—	—	—	—	—	↕	2

Mnemonic	Operand Size	Operation	Branching Condition	Addressing Mode/ Instruction Length (Bytes)									Condition Code						No. of States
				#xx: 8/16	Rn	@Rn	@(d:16, Rn)	@–Rn/@Rn+	@aa: 8/16	@(d:8, PC)	@@aa	Implied	I	H	N	Z	V	C	
BIXOR #xx:3, @Rd	B	C⊕(#xx:3 of @Rd16) → C				4							—	—	—	—	—	↕	6
BIXOR #xx:3, @aa:8	B	C⊕(#xx:3 of @aa:8) → C							4				—	—	—	—	—	↕	6
BRA d:8 (BT d:8)	—	PC ← PC+d:8								2			—	—	—	—	—	—	4
BRN d:8 (BF d:8)	—	PC ← PC+2								2			—	—	—	—	—	—	4
BHI d:8	—	If	C ∨ Z = 0							2			—	—	—	—	—	—	4
BLS d:8	—	condition	C ∨ Z = 1							2			—	—	—	—	—	—	4
BCC d:8 (BHS d:8)	—	is true	C = 0							2			—	—	—	—	—	—	4
BCS d:8 (BLO d:8)	—	then	C = 1							2			—	—	—	—	—	—	4
BNE d:8	—	PC ←	Z = 0							2			—	—	—	—	—	—	4
BEQ d:8	—	PC+d:8	Z = 1							2			—	—	—	—	—	—	4
BVC d:8	—	else next;	V = 0							2			—	—	—	—	—	—	4
BVS d:8	—		V = 1							2			—	—	—	—	—	—	4
BPL d:8	—		N = 0							2			—	—	—	—	—	—	4
BMI d:8	—		N = 1							2			—	—	—	—	—	—	4
BGE d:8	—		N⊕V = 0							2			—	—	—	—	—	—	4
BLT d:8	—		N⊕V = 1							2			—	—	—	—	—	—	4
BGT d:8	—		Z ∨ (N⊕V) = 0							2			—	—	—	—	—	—	4
BLE d:8	—		Z ∨ (N⊕V) = 1							2			—	—	—	—	—	—	4
JMP @Rn	—	PC ← Rn16				2							—	—	—	—	—	—	4
JMP @aa:16	—	PC ← aa:16							4				—	—	—	—	—	—	6
JMP @@aa:8	—	PC ← @aa:8									2		—	—	—	—	—	—	8
BSR d:8	—	SP–2 → SP PC → @SP PC ← PC+d:8								2			—	—	—	—	—	—	6
JSR @Rn	—	SP–2 → SP PC → @SP PC ← Rn16				2							—	—	—	—	—	—	6
JSR @aa:16	—	SP–2 → SP PC → @SP PC ← aa:16							4				—	—	—	—	—	—	8

Mnemonic	Operand Size	Operation	Addressing Mode/ Instruction Length (Bytes)									Condition Code						No. of States
			#xx: 8/16	Rn	@Rn	@(d:16, Rn)	@–Rn/@Rn+	@aa: 8/16	@(d:8, PC)	@@aa	Implied	I	H	N	Z	V	C	
JSR @@aa:8		SP–2 → SP PC → @SP PC ← @aa:8								2		—	—	—	—	—	—	8
RTS	—	PC ← @SP SP+2 → SP									2	—	—	—	—	—	—	8
RTE	—	CCR ← @SP SP+2 → SP PC ← @SP SP+2 → SP									2	↕	↕	↕	↕	↕	↕	10
SLEEP	—	Transit to power-down state									2	—	—	—	—	—	—	2
LDC #xx:8, CCR	B	#xx:8 → CCR	2									↕	↕	↕	↕	↕	↕	2
LDC Rs, CCR	B	Rs8 → CCR		2								↕	↕	↕	↕	↕	↕	2
STC CCR, Rd	B	CCR → Rd8		2								—	—	—	—	—	—	2
ANDC #xx:8, CCR	B	CCR∧#xx:8 → CCR	2									↕	↕	↕	↕	↕	↕	2
ORC #xx:8, CCR	B	CCR∨#xx:8 → CCR	2									↕	↕	↕	↕	↕	↕	2
XORC #xx:8, CCR	B	CCR⊕#xx:8 → CCR	2									↕	↕	↕	↕	↕	↕	2
NOP	—	PC ← PC+2									2	—	—	—	—	—	—	2
EEPMOV	—	if R4L≠0 Repeat @R5 → @R6 R5+1 → R5 R6+1 → R6 R4L–1 → R4L Until R4L=0 else next;									4	—	—	—	—	—	—	(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 cleared to 0.

(3) Set to 1 if decimal adjustment produces a carry; otherwise retains value prior to arithmetic operation.

(4) The number of states required for execution is 4n + 9 (n = value of R4L).

(5) Set to 1 if the divisor is negative; otherwise cleared to 0.

(6) Set to 1 if the divisor is zero; otherwise cleared to 0.

A.2 Operation Code Map

Table A.2 is an operation code map. It shows the operation codes contained in the first byte of the instruction code (bits 15 to 8 of the first instruction word).

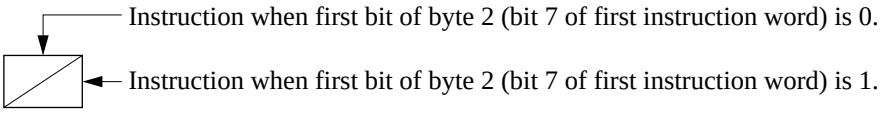
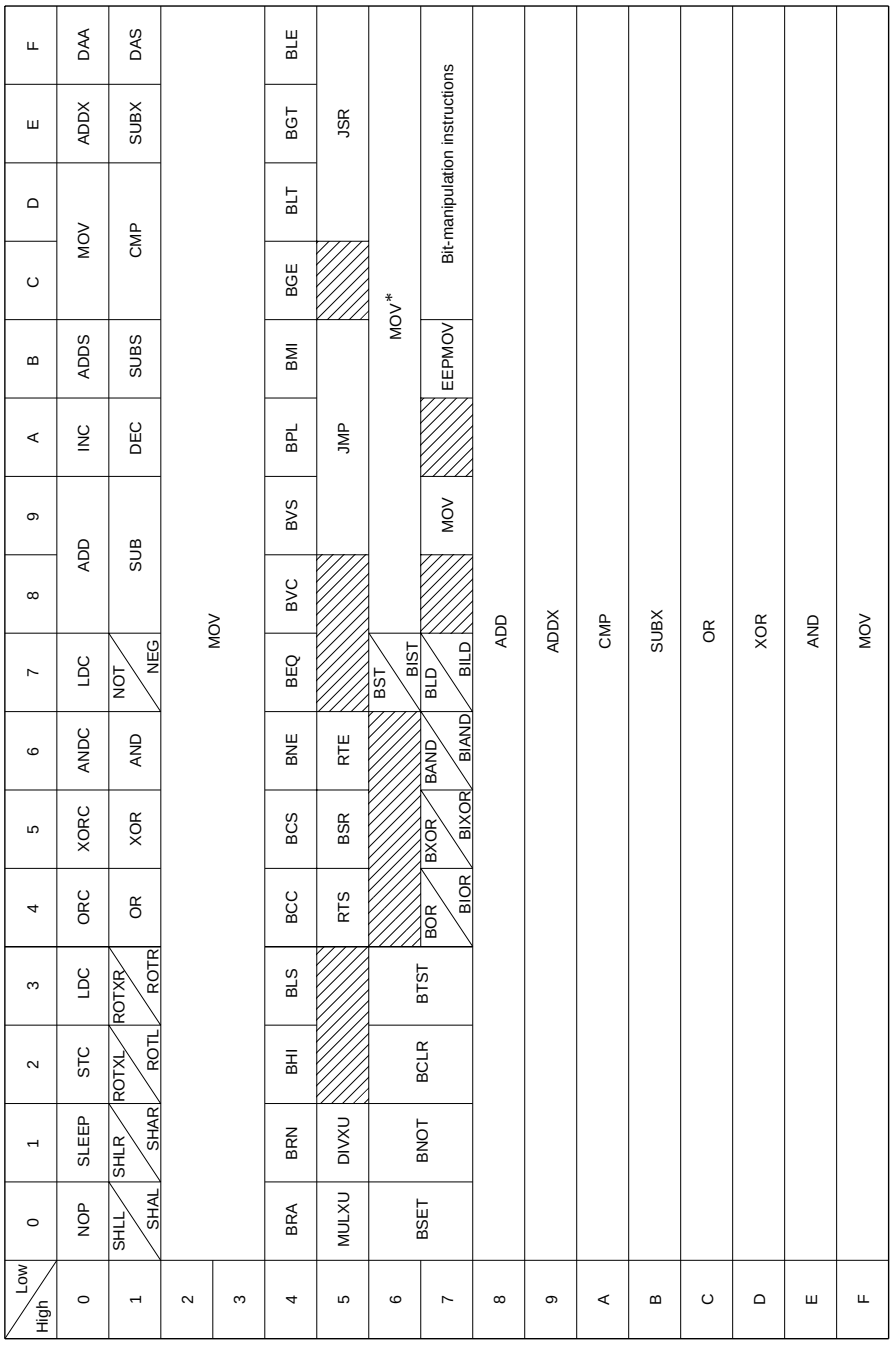


Table A.2 Operation Code Map



Note: * The PUSH and POP instructions are identical in machine language to MOV instructions.

A.3 Number of Execution States

The tables here can be used to calculate the number of states required for instruction execution. Table A-4 indicates the number of states required for each cycle (instruction fetch, data read/write, etc.) in instruction execution, and table A-3 indicates the number of cycles of each type occurring in each instruction. The total number of states required for execution of an instruction can be calculated from these two tables as follows:

$$\text{Execution states} = I \times S_I + J \times S_J + K \times S_K + L \times S_L + M \times S_M + N \times S_N$$

Examples: When instruction is fetched from on-chip ROM, and an on-chip RAM is accessed.

BSET #0, @FF00

From table A.4:
I = L = 2, J = K = M = N= 0

From table A.3:
$S_I = 2$, $S_L = 2$

Number of states required for execution = $2 \times 2 + 2 \times 2 = 8$

When instruction is fetched from on-chip ROM, branch address is read from on-chip ROM, and on-chip RAM is used for stack area.

JSR @@ 30

From table A.4:
I = 2, J = K = 1, L = M = N = 0

From table A.3:
$S_I = S_J = S_K = 2$

Number of states required for execution = $2 \times 2 + 1 \times 2 + 1 \times 2 = 8$

Table A.3 Number of Cycles in Each Instruction

Execution Status (Instruction Cycle)		Access Location	
		On-Chip Memory	On-Chip Peripheral Module
Instruction fetch	S_I	2	—
Branch address read	S_J		
Stack operation	S_K		
Byte data access	S_L		2 or 3*
Word data access	S_M		—
Internal operation	S_N	1	

Note: * Depends on which on-chip module is accessed. See 2.9.1, Notes on Data Access for details.

Table A.4 Number of Cycles in Each Instruction

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
ADD	ADD.B #xx:8, Rd	1					
	ADD.B Rs, Rd	1					
	ADD.W Rs, Rd	1					
ADDS	ADDS.W #1, Rd	1					
	ADDS.W #2, Rd	1					
ADDX	ADDX.B #xx:8, Rd	1					
	ADDX.B Rs, Rd	1					
AND	AND.B #xx:8, Rd	1					
	AND.B Rs, Rd	1					
ANDC	ANDC #xx:8, CCR	1					
BAND	BAND #xx:3, Rd	1					
	BAND #xx:3, @Rd	2			1		
	BAND #xx:3, @aa:8	2			1		
Bcc	BRA d:8 (BT d:8)	2					
	BRN d:8 (BF d:8)	2					
	BHI d:8	2					
	BLS d:8	2					
	BCC d:8 (BHS d:8)	2					
	BCS d:8 (BLO d:8)	2					
	BNE d:8	2					
	BEQ d:8	2					
	BVC d:8	2					
	BVS d:8	2					
	BPL d:8	2					
	BMI d:8	2					
	BGE d:8	2					
	BLT d:8	2					
	BGT d:8	2					
	BLE d:8	2					
BCLR	BCLR #xx:3, Rd	1					
	BCLR #xx:3, @Rd	2			2		
	BCLR #xx:3, @aa:8	2			2		
	BCLR Rn, Rd	1					

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
BCLR	BCLR Rn, @Rd	2			2		
	BCLR Rn, @aa:8	2			2		
BIAND	BIAND #xx:3, Rd	1					
	BIAND #xx:3, @Rd	2			1		
	BIAND #xx:3, @aa:8	2			1		
BILD	BILD #xx:3, Rd	1					
	BILD #xx:3, @Rd	2			1		
	BILD #xx:3, @aa:8	2			1		
BIOR	BIOR #xx:3, Rd	1					
	BIOR #xx:3, @Rd	2			1		
	BIOR #xx:3, @aa:8	2			1		
BIST	BIST #xx:3, Rd	1					
	BIST #xx:3, @Rd	2			2		
	BIST #xx:3, @aa:8	2			2		
BIXOR	BIXOR #xx:3, Rd	1					
	BIXOR #xx:3, @Rd	2			1		
	BIXOR #xx:3, @aa:8	2			1		
BLD	BLD #xx:3, Rd	1					
	BLD #xx:3, @Rd	2			1		
	BLD #xx:3, @aa:8	2			1		
BNOT	BNOT #xx:3, Rd	1					
	BNOT #xx:3, @Rd	2			2		
	BNOT #xx:3, @aa:8	2			2		
	BNOT Rn, Rd	1					
	BNOT Rn, @Rd	2			2		
	BNOT Rn, @aa:8	2			2		
BOR	BOR #xx:3, Rd	1					
	BOR #xx:3, @Rd	2			1		
	BOR #xx:3, @aa:8	2			1		
BSET	BSET #xx:3, Rd	1					
	BSET #xx:3, @Rd	2			2		
	BSET #xx:3, @aa:8	2			2		
	BSET Rn, Rd	1					
	BSET Rn, @Rd	2			2		

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
BSET	BSET Rn, @aa:8	2			2		
BSR	BSR d:8	2		1			
BST	BST #xx:3, Rd	1					
	BST #xx:3, @Rd	2			2		
	BST #xx:3, @aa:8	2			2		
BTST	BTST #xx:3, Rd	1					
	BTST #xx:3, @Rd	2			1		
	BTST #xx:3, @aa:8	2			1		
	BTST Rn, Rd	1					
	BTST Rn, @Rd	2			1		
	BTST Rn, @aa:8	2			1		
BXOR	BXOR #xx:3, Rd	1					
	BXOR #xx:3, @Rd	2			1		
	BXOR #xx:3, @aa:8	2			1		
CMP	CMP. B #xx:8, Rd	1					
	CMP. B Rs, Rd	1					
	CMP.W Rs, Rd	1					
DAA	DAA.B Rd	1					
DAS	DAS.B Rd	1					
DEC	DEC.B Rd	1					
DIVXU	DIVXU.B Rs, Rd	1					12
EEPMOV	EEPMOV	2			2n+2*		1
INC	INC.B Rd	1					
JMP	JMP @Rn	2					
	JMP @aa:16	2					2
	JMP @@aa:8	2	1				2
JSR	JSR @Rn	2		1			
	JSR @aa:16	2		1			2
	JSR @@aa:8	2	1	1			
LDC	LDC #xx:8, CCR	1					
	LDC Rs, CCR	1					
MOV	MOV.B #xx:8, Rd	1					
	MOV.B Rs, Rd	1					

Note: n: Initial value in R4L. The source and destination operands are accessed n + 1 times each.

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
MOV	MOV.B @Rs, Rd	1			1		
	MOV.B @(d:16, Rs), Rd	2			1		
	MOV.B @Rs+, Rd	1			1		2
	MOV.B @aa:8, Rd	1			1		
	MOV.B @aa:16, Rd	2			1		
	MOV.B Rs, @Rd	1			1		
	MOV.B Rs, @(d:16, Rd)	2			1		
	MOV.B Rs, @–Rd	1			1		2
	MOV.B Rs, @aa:8	1			1		
	MOV.B Rs, @aa:16	2			1		
	MOV.W #xx:16, Rd	2					
	MOV.W Rs, Rd	1					
	MOV.W @Rs, Rd	1				1	
	MOV.W @(d:16, Rs), Rd	2				1	
	MOV.W @Rs+, Rd	1				1	2
	MOV.W @aa:16, Rd	2				1	
	MOV.W Rs, @Rd	1				1	
	MOV.W Rs, @(d:16, Rd)	2				1	
	MOV.W Rs, @–Rd	1				1	2
	MOV.W Rs, @aa:16	2				1	
MULXU	MULXU.B Rs, Rd	1					12
NEG	NEG.B Rd	1					
NOP	NOP	1					
NOT	NOT.B Rd	1					
OR	OR.B #xx:8, Rd	1					
	OR.B Rs, Rd	1					
ORC	ORC #xx:8, CCR	1					
ROTL	ROTL.B Rd	1					
ROTR	ROTR.B Rd	1					

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
ROTXL	ROTXL.B Rd	1					
ROTXR	ROTXR.B Rd	1					
RTE	RTE	2		2			2
RTS	RTS	2		1			2
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			2
PUSH	PUSH Rs	1		1			2
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					

Appendix B Internal I/O Registers

B.1 Register Addresses

B.1.1 H8/3857 Series Addresses

Address (low)	Register Name	Bit Names								Module Name
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'80	FLMCR1*1	FWE	SWE	—	—	EV	PV	E	P	Flash memory
H'81	FLMCR2*1	FLER	—	—	—	—	—	ESU	PSU	
H'82										
H'83	EBR*1	—	EB6	EB5	EB4	EB3	EB2	EB1	EB0	
H'84										
H'85										
H'86										
H'87										
H'88										
H'89	MDCR*1	—	—	—	—	—	—	TSDS2	TSDS1	
H'8A										
H'8B										
H'8C										
H'8D										
H'8E										
H'8F	SYSCR3*1	—	—	—	—	FLSHE	—	—	—	
H'90	TCSRW*2	B6WI	TCWE	B4WI	TCSRWE	B2WI	WDON	B0WI	WRST	Watchdog timer
H'91	TCW*2	TCW7	TCW6	TCW5	TCW4	TCW3	TCW2	TCW1	TCW0	
H'92	TMW*2	—	—	—	—	—	CKS2	CKS1	CKS0	
H'93										
H'94										
H'95										
H'96										
H'97										
H'98										
H'99										
H'9A										
H'9B										

Address (low)	Register Name	Bit Names								Module Name
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'9C										
H'9D										
H'9E										
H'9F										
H'A0	SCR1	SNC1	SNC0	—	—	CKS3	CKS2	CKS1	CKS0	SCI1
H'A1	SCSR1	—	SOL	ORER	—	—	—	—	STF	
H'A2	SDRU	SDRU7	SDRU6	SDRU5	SDRU4	SDRU3	SDRU2	SDRU1	SDRU0	
H'A3	SDRL	SDRL7	SDRL6	SDRL5	SDRL4	SDRL3	SDRL2	SDRL1	SDRL0	
H'A4										
H'A5										
H'A6										
H'A7										
H'A8	SMR	COM	CHR	PE	PM	STOP	MP	CKS1	CKS0	SCI3
H'A9	BRR	BRR7	BRR6	BRR5	BRR4	BRR3	BRR2	BRR1	BRR0	
H'AA	SCR3	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
H'AB	TDR	TDR7	TDR6	TDR5	TDR4	TDR3	TDR2	TDR1	TDR0	
H'AC	SSR	TDRE	RDRF	OER	FER	PER	TEND	MPBR	MPBT	
H'AD	RDR	RDR7	RDR6	RDR5	RDR4	RDR3	RDR2	RDR1	RDR0	
H'AE										
H'AF										
H'B0	TMA	TMA7	TMA6	TMA5	—	TMA3	TMA2	TMA1	TMA0	Timer A
H'B1	TCA	TCA7	TCA6	TCA5	TCA4	TCA3	TCA2	TCA1	TCA0	
H'B2	TMB	TMB7	—	—	—	—	TMB2	TMB1	TMB0	Timer B
H'B3	TCB/TLB	TCB7/ TLB7	TCB6/ TLB6	TCB5/ TLB5	TCB4/ TLB4	TCB3/ TLB3	TCB2/ TLB2	TCB1/ TLB1	TCB0/ TLB0	
H'B4	TMC	TMC7	TMC6	TMC5	—	—	TMC2	TMC1	TMC0	Timer C
H'B5	TCC/TLC	TCC7/ TLC7	TCC6/ TLC6	TCC5/ TLC5	TCC4/ TLC4	TCC3/ TLC3	TCC2/ TLC2	TCC1/ TLC1	TCC0/ TLC0	
H'B6	TCRF	TOLH	CKSH2	CKSH1	CKSH0	TOLL	CKSL2	CKSL1	CKSL0	Timer F
H'B7	TCSRF	OVFH	CMFH	OVIEH	CCLRH	OVFL	CMFL	OVIEL	CCLRL	
H'B8	TCFH	TCFH7	TCFH6	TCFH5	TCFH4	TCFH3	TCFH2	TCFH1	TCFH0	

Address (low)	Register Name	Bit Names								Module Name
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'B9	TCFL	TCFL7	TCFL6	TCFL5	TCFL4	TCFL3	TCFL2	TCFL1	TCFL0	Timer F
H'BA	OCRFH	OCRFH7	OCRFH6	OCRFH5	OCRFH4	OCRFH3	OCRFH2	OCRFH1	OCRFH0	
H'BB	OCRFL	OCRFL7	OCRFL6	OCRFL5	OCRFL4	OCRFL3	OCRFL2	OCRFL1	OCRFL0	
H'BC										
H'BD										
H'BE										
H'BF										
H'C0										
H'C1										
H'C2										
H'C3										
H'C4	AMR	CKS	TRGE	—	—	CH3	CH2	CH1	CH0	A/D converter
H'C5	ADRR	ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0	
H'C6	ADSR	ADSF	—	—	—	—	—	—	—	
H'C7										
H'C8	PMR1	IRQ3	IRQ2	IRQ1	PWM	—	TMOFH	TMOFL	TMOW	I/O ports
H'C9	PMR2	—	—	—	—	IRQ0	POF1	UD	IRQ4	
H'CA	PMR3	—	—	—	—	—	SO1	SI1	SCK1	
H'CB	PMR4	NMOD7	NMOD6	NMOD5	NMOD4	NMOD3	NMOD2	NMOD1	NMOD0	
H'CC	PMR5	WKP7	WKP6	WKP5	WKP4	WKP3	WKP2	WKP1	WKP0	
H'CD										
H'CE										
H'CF										
H'D0	PWCR	—	—	—	—	—	—	—	PWCR0	14-bit PWM
H'D1	PWDRU	—	—	PWDRU5	PWDRU4	PWDRU3	PWDRU2	PWDRU1	PWDRU0	
H'D2	PWDRL	PWDRL7	PWDRL6	PWDRL5	PWDRL4	PWDRL3	PWDRL2	PWDRL1	PWDRL0	
H'D3										
H'D4	PDR1	$P1_7$	$P1_6$	$P1_5$	$P1_4$	$P1_3$	$P1_2$	$P1_1$	$P1_0$	I/O ports
H'D5	PDR2	$P2_7$	$P2_6$	$P2_5$	$P2_4$	$P2_3$	$P2_2$	$P2_1$	$P2_0$	
H'D6	PDR3	$P3_7$	$P3_6$	$P3_5$	$P3_4$	$P3_3$	$P3_2$	$P3_1$	$P3_0$	
H'D7	PDR4	—	—	—	—	$P4_3$	$P4_2$	$P4_1$	$P4_0$	
H'D8	PDR5	$P5_7$	$P5_6$	$P5_5$	$P5_4$	$P5_3$	$P5_2$	$P5_1$	$P5_0$	
H'D9										

Address (low)	Register Name	Bit Names								Module Name
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'DA										I/O ports
H'DB										
H'DC	PDR9	$P9_7$	$P9_6$	$P9_5$	$P9_4$	$P9_3$	$P9_2$	$P9_1$	$P9_0$	
H'DD	PDRA	—	—	—	—	PA_3	PA_2	PA_1	PA_0	
H'DE	PDRB	PB_7	PB_6	PB_5	PB_4	PB_3	PB_2	PB_1	PB_0	
H'DF										
H'E0	PUCR1	$PUCR1_7$	$PUCR1_6$	$PUCR1_5$	$PUCR1_4$	$PUCR1_3$	$PUCR1_2$	$PUCR1_1$	$PUCR1_0$	
H'E1	PUCR3	$PUCR3_7$	$PUCR3_6$	$PUCR3_5$	$PUCR3_4$	$PUCR3_3$	$PUCR3_2$	$PUCR3_1$	$PUCR3_0$	
H'E2	PUCR5	$PUCR5_7$	$PUCR5_6$	$PUCR5_5$	$PUCR5_4$	$PUCR5_3$	$PUCR5_2$	$PUCR5_1$	$PUCR5_0$	
H'E3										
H'E4	PCR1	$PCR1_7$	$PCR1_6$	$PCR1_5$	$PCR1_4$	$PCR1_3$	$PCR1_2$	$PCR1_1$	$PCR1_0$	
H'E5	PCR2	$PCR2_7$	$PCR2_6$	$PCR2_5$	$PCR2_4$	$PCR2_3$	$PCR2_2$	$PCR2_1$	$PCR2_0$	
H'E6	PCR3	$PCR3_7$	$PCR3_6$	$PCR3_5$	$PCR3_4$	$PCR3_3$	$PCR3_2$	$PCR3_1$	$PCR3_0$	
H'E7	PCR4	—	—	—	—	—	$PCR4_2$	$PCR4_1$	$PCR4_0$	
H'E8	PCR5	$PCR5_7$	$PCR5_6$	$PCR5_5$	$PCR5_4$	$PCR5_3$	$PCR5_2$	$PCR5_1$	$PCR5_0$	
H'E9										
H'EA										
H'EB										
H'EC	PCR9	$PCR9_7$	$PCR9_6$	$PCR9_5$	$PCR9_4$	$PCR9_3$	$PCR9_2$	$PCR9_1$	$PCR9_0$	
H'ED	PCRA	—	—	—	—	$PCRA_3$	$PCRA_2$	$PCRA_1$	$PCRA_0$	
H'EE										
H'EF										
H'F0	SYSCR1	SSBY	STS2	STS1	STS0	LSON	—	—	—	System control
H'F1	SYSCR2	—	—	—	NESEL	DTON	MSON	SA1	SA0	
H'F2	IEGR	—	—	—	IEG4	IEG3	IEG2	IEG1	IEG0	
H'F3	IENR1	IENTA	IENS1	IENWP	IEN4	IEN3	IEN2	IEN1	IEN0	
H'F4	IENR2	IENDT	IENAD	—	—	IENTFH	IENTFL	IENTC	IENTB	
H'F5										
H'F6	IRR1	IRRTA	IRRS1	—	IRRI4	IRRI3	IRRI2	IRRI1	IRRI0	
H'F7	IRR2	IRRDT	IRRAD	—	—	IRRTFH	IRRTFL	IRRTC	IRRTB	
H'F8										
H'F9	IWPR	IWPF7	IWPF6	IWPF5	IWPF4	IWPF3	IWPF2	IWPF1	IWPF0	

Address (low)	Register Name	Bit Names								Module Name
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'FA										
H'FB										
H'FC										
H'FD										
H'FE										
H'FF										

Notation:

SCI1: Serial communication interface 1

SCI3: Serial communication interface 3

Notes: 1. Applies to the F-ZTAT version. In the mask ROM version, a read access to the address of a register other than MDCR will always return 0, a read access to the MDCR address will return an undefined value, and writes are invalid.

2. Applies to the F-ZTAT version. In the mask ROM version, read accesses to the corresponding addresses will always return 1, and writes are invalid.

B.1.2 H8/3854 Series Addresses

Address (low)	Register Name	Bit Names								Module Name
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'80	FLMCR1*1	FWE	SWE	—	—	EV	PV	E	P	Flash memory
H'81	FLMCR2*1	FLER	—	—	—	—	—	ESU	PSU	
H'82										
H'83	EBR*1	—	EB6	EB5	EB4	EB3	EB2	EB1	EB0	
H'84										
H'85										
H'86										
H'87										
H'88										
H'89	MDCR*1	—	—	—	—	—	—	TSDS2	TSDS1	
H'8A										
H'8B										
H'8C										
H'8D										
H'8E										
H'8F	SYSCR3*1	—	—	—	—	FLSHE	—	—	—	
H'90	TCSRW*2	B6WI	TCWE	B4WI	TCSRWE	B2WI	WDON	B0WI	WRST	Watchdog timer
H'91	TCW*2	TCW7	TCW6	TCW5	TCW4	TCW3	TCW2	TCW1	TCW0	
H'92	TMW*2	—	—	—	—	—	CKS2	CKS1	CKS0	
H'93										
H'94										
H'95										
H'96										
H'97										
H'98										
H'99										
H'9A										
H'9B										
H'9C										
H'9D										
H'9E										
H'9F										

Address (low)	Register Name	Bit Names Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module Name
H'A0										
H'A1										
H'A2										
H'A3										
H'A4										
H'A5										
H'A6										
H'A7										
H'A8	SMR	COM	CHR	PE	PM	STOP	MP	CKS1	CKS0	SCI3
H'A9	BRR	BRR7	BRR6	BRR5	BRR4	BRR3	BRR2	BRR1	BRR0	
H'AA	SCR3	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
H'AB	TDR	TDR7	TDR6	TDR5	TDR4	TDR3	TDR2	TDR1	TDR0	
H'AC	SSR	TDRE	RDRF	OER	FER	PER	TEND	MPBR	MPBT	
H'AD	RDR	RDR7	RDR6	RDR5	RDR4	RDR3	RDR2	RDR1	RDR0	
H'AE										
H'AF										
H'B0	TMA	TMA7	TMA6	TMA5	—	TMA3	TMA2	TMA1	TMA0	Timer A
H'B1	TCA	TCA7	TCA6	TCA5	TCA4	TCA3	TCA2	TCA1	TCA0	
H'B2	TMB	TMB7	—	—	—	—	TMB2	TMB1	TMB0	Timer B
H'B3	TCB/TLB	TCB7/ TLB7	TCB6/ TLB6	TCB5/ TLB5	TCB4/ TLB4	TCB3/ TLB3	TCB2/ TLB2	TCB1/ TLB1	TCB0/ TLB0	
H'B4										
H'B5										
H'B6	TCRF	TOLH	CKSH2	CKSH1	CKSH0	TOLL	CKSL2	CKSL1	CKSL0	Timer F
H'B7	TCSRF	OVFH	CMFH	OVIEH	CCLRH	OVFL	CMFL	OVIEL	CCLRL	
H'B8	TCFH	TCFH7	TCFH6	TCFH5	TCFH4	TCFH3	TCFH2	TCFH1	TCFH0	
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										
H'BD										
H'BE										
H'BF										
H'C0										
H'C1										

Address (low)	Register Name	Bit Names								Module Name
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'C2										
H'C3										
H'C4	AMR	CKS	TRGE	—	—	CH3	CH2	CH1	CH0	A/D converter
H'C5	ADRR	ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0	
H'C6	ADSR	ADSF	—	—	—	—	—	—	—	
H'C7										
H'C8	PMR1	IRQ3	—	IRQ1	—	—	TMOFH	TMOFL	TMOW	I/O ports
H'C9	PMR2	—	—	—	—	IRQ0	—	—	IRQ4	
H'CA										
H'CB	PMR4	NMOD7	NMOD6	NMOD5	NMOD4	NMOD3	NMOD2	NMOD1	NMOD0	
H'CC	PMR5	WKP7	WKP6	WKP5	WKP4	WKP3	WKP2	WKP1	WKP0	
H'CD										
H'CE										
H'CF										
H'D0										
H'D1										
H'D2										
H'D3										
H'D4	PDR1	$P1_7$	—	$P1_5$	—	—	$P1_2$	$P1_1$	$P1_0$	I/O ports
H'D5	PDR2	$P2_7$	$P2_6$	$P2_5$	$P2_4$	$P2_3$	$P2_2$	$P2_1$	$P2_0$	
H'D6										
H'D7	PDR4	—	—	—	—	$P4_3$	$P4_2$	$P4_1$	$P4_0$	
H'D8	PDR5	$P5_7$	$P5_6$	$P5_5$	$P5_4$	$P5_3$	$P5_2$	$P5_1$	$P5_0$	
H'D9										
H'DA										
H'DB										
H'DC	PDR9	$P9_7$	$P9_6$	$P9_5$	$P9_4$	$P9_3$	$P9_2$	$P9_1$	$P9_0$	
H'DD	PDRA	—	—	—	—	PA_3	PA_2	PA_1	PA_0	
H'DE	PDRB	PB_7	PB_6	PB_5	PB_4	—	—	—	—	
H'DF										
H'E0	PUCR1	$PUCR1_7$	—	$PUCR1_5$	—	—	$PUCR1_2$	$PUCR1_1$	$PUCR1_0$	
H'E1										
H'E2	PUCR5	$PUCR5_7$	$PUCR5_6$	$PUCR5_5$	$PUCR5_4$	$PUCR5_3$	$PUCR5_2$	$PUCR5_1$	$PUCR5_0$	
H'E3										

Address (low)	Register Name	Bit Names								Module Name
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'E4	PCR1	$PCR1_7$	—	$PCR1_5$	—	—	$PCR1_2$	$PCR1_1$	$PCR1_0$	I/O ports
H'E5	PCR2	$PCR2_7$	$PCR2_6$	$PCR2_5$	$PCR2_4$	$PCR2_3$	$PCR2_2$	$PCR2_1$	$PCR2_0$	
H'E6										
H'E7	PCR4	—	—	—	—	—	$PCR4_2$	$PCR4_1$	$PCR4_0$	
H'E8	PCR5	$PCR5_7$	$PCR5_6$	$PCR5_5$	$PCR5_4$	$PCR5_3$	$PCR5_2$	$PCR5_1$	$PCR5_0$	
H'E9										
H'EA										
H'EB										
H'EC	PCR9	$PCR9_7$	$PCR9_6$	$PCR9_5$	$PCR9_4$	$PCR9_3$	$PCR9_2$	$PCR9_1$	$PCR9_0$	
H'ED	PCRA	—	—	—	—	$PCRA_3$	$PCRA_2$	$PCRA_1$	$PCRA_0$	
H'EE										
H'EF										
H'F0	SYSCR1	SSBY	STS2	STS1	STS0	LSON	—	—	—	System control
H'F1	SYSCR2	—	—	—	NESEL	DTON	MSON	SA1	SA0	
H'F2	IEGR	—	—	—	IEG4	IEG3	—	IEG1	IEG0	
H'F3	IENR1	IENTA	—	IENWP	IEN4	IEN3	—	IEN1	IEN0	
H'F4	IENR2	IENDT	IENAD	—	—	IENTFH	IENTFL	—	IENTB	
H'F5										
H'F6	IRR1	IRRTA	—	—	IRRI4	IRRI3	—	IRRI1	IRRI0	
H'F7	IRR2	IRRDT	IRRAD	—	—	IRRTFH	IRRTFL	—	IRRTB	
H'F8										
H'F9	IWPR	IWPF7	IWPF6	IWPF5	IWPF4	IWPF3	IWPF2	IWPF1	IWPF0	
H'FA										
H'FB										
H'FC										
H'FD										
H'FE										
H'FF										

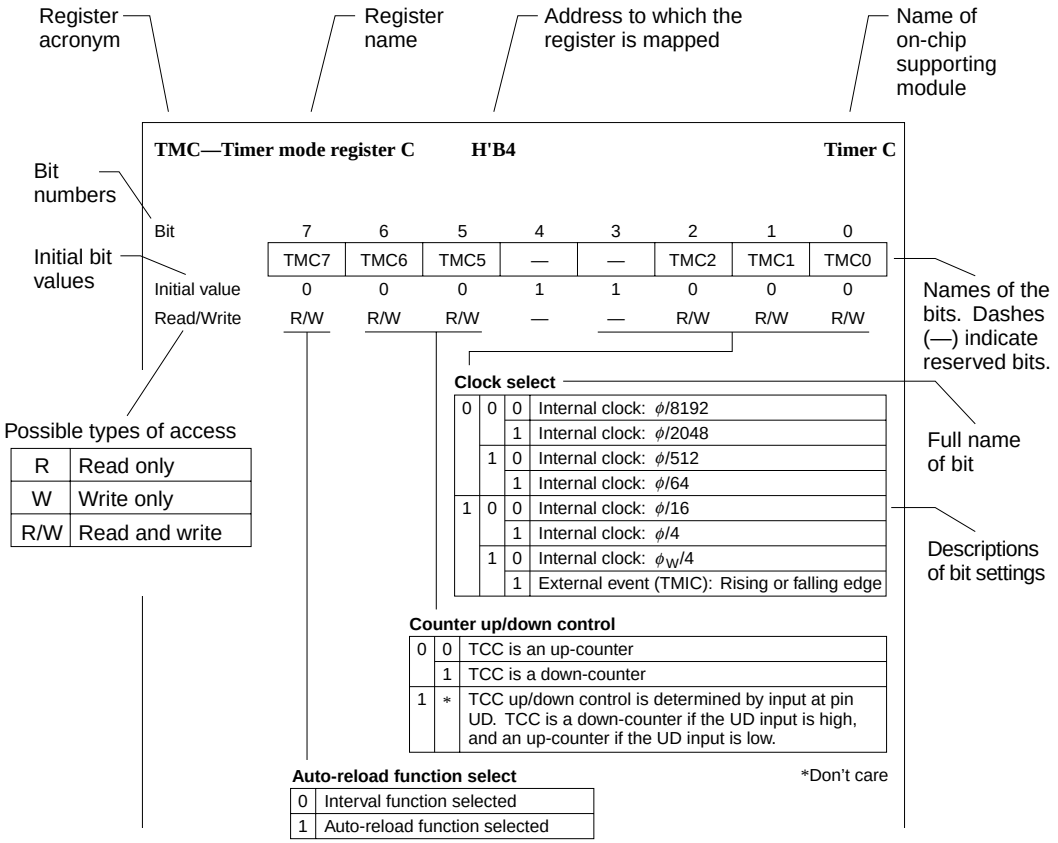
Notation:

SCI3: Serial communication interface 3

Notes: 1. Applies to the F-ZTAT version. In the mask ROM version, a read access to the address of a register other than MDCR will always return 0, a read access to the MDCR address will return an undefined value, and writes are invalid.

2. Applies to the F-ZTAT version. In the mask ROM version, read accesses to the corresponding addresses will always return 1, and writes are invalid.

B.2 Register Descriptions



Bit	:	7	6	5	4	3	2	1	0
		FWE	SWE	—	—	EV	PV	E	P
Initial value	:	—*	0	0	0	0	0	0	0
R/W	:	R	R/W	—	—	R/W	R/W	R/W	R/W

Program

0	Program mode cleared
1	Transition to program mode [Setting condition] When FWE = 1, SWE = 1, and PSU = 1

Erase

0	Erase mode cleared
1	Transition to erase mode [Setting condition] When FWE = 1, SWE = 1, and ESU = 1

Program-verify

0	Program-verify mode cleared
1	Transition to program-verify mode [Setting condition] When FWE = 1 and SWE = 1

Erase -verify

0	Erase -verify mode cleared
1	Transition to erase-verify mode [Setting condition] When FWE = 1 and SWE = 1

Software write enable

0	Programmin/ disabled
1	Programming enabled [Setting condition] When FWE = 1

Flash write enable

0	When a low level is input to the FWE pin (hardware-protected state)
1	When a high level is input to the FWE pin

Note: * Determined by the state of the FWE pin.

FLMCR2—Flash memory control register 2 **H'81** **Flash memory (On-chip flash memory version only)**

Bit	:	7	6	5	4	3	2	1	0
		FLER	—	—	—	—	—	ESU	PSU
Initial value	:	0	0	0	0	0	0	0	0
R/W	:	R	—	—	—	—	—	R/W	R/W

Program setup

0	Program setup cleared
1	Program setup [Setting condition] When FWE = 1 and SWE = 1

Erase setup

0	Erase setup cleared
1	Erase setup [Setting condition] When FWE = 1 and SWE = 1

Flash memory error

0	Flash memory is operating normally Flash memory program/erase protection (error protection) is disabled [Clearing condition] Reset or hardware standby mode
1	An error occurred during flash memory programming/erasing Flash memory program/erase protection (error protection) is enabled [Setting condition] See section 6.6.3, Error Protection

Bit	:	7	6	5	4	3	2	1	0
		—	EB6	EB5	EB4	EB3	EB2	EB1	EB0
Initial value	:	0	0	0	0	0	0	0	0
R/W	:	—	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Flash memory erase blocks

Block (Size)	Addresses
EB0 (1 kbyte)	H'0000 to H'03FF
EB1 (1 kbyte)	H'0400 to H'07FF
EB2 (1 kbyte)	H'0800 to H'0BFF
EB3 (1 kbyte)	H'0C00 to H'0FFF
EB4 (28 kbytes)	H'1000 to H'7FFF
EB5 (16 kbytes)	H'8000 to H'BFFF
EB6 (12 kbytes)	H'C000 to H'EDFF

MDCR—Mode control register **H'89** **Flash memory (On-chip flash memory version only)**

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSDS2	TSDS1
Initial value	0	0	0	0	0	0	—*	—*
R/W	—	—	—	—	—	—	R	R

Test pin monitor bits

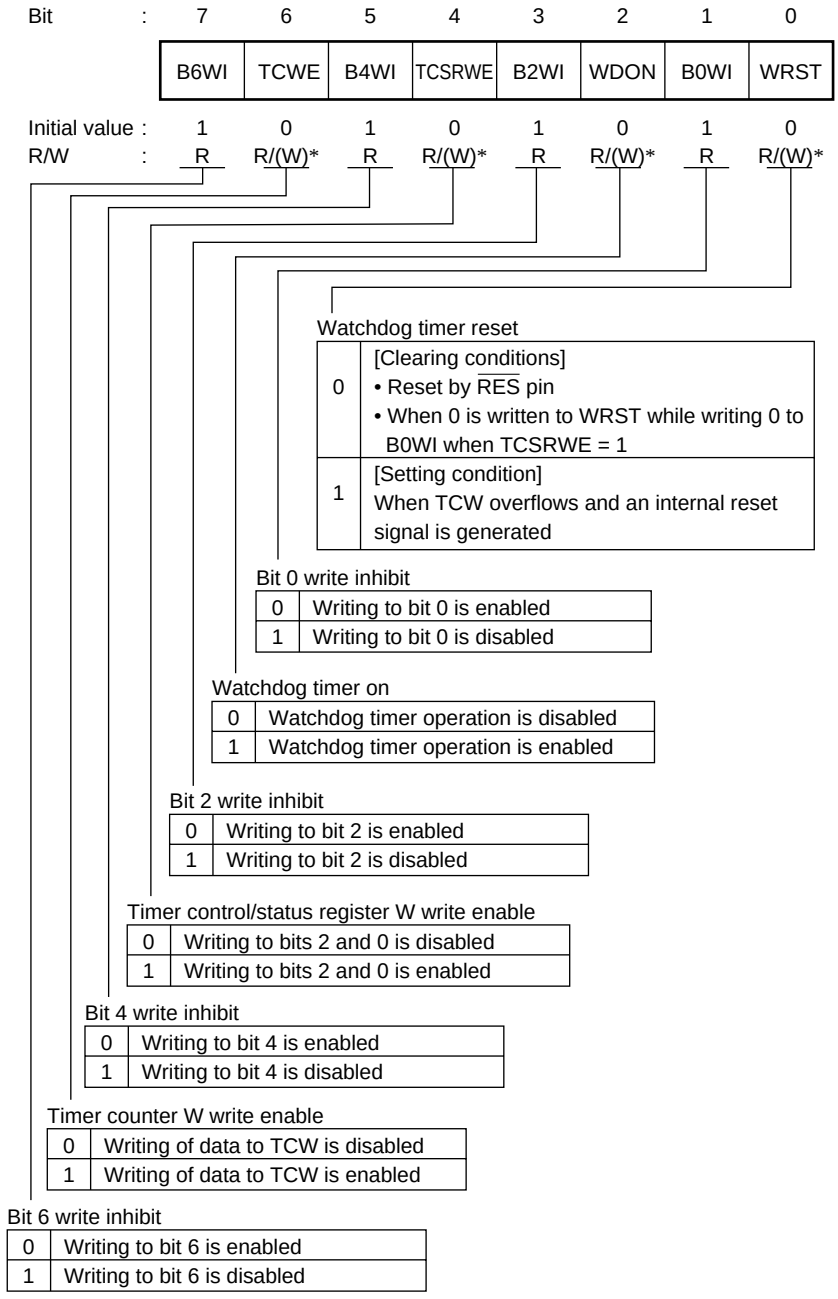
Note: * Determined by the TEST and TEST2 pins.

SYSCR3—System control register 3 **H'8F** **Flash memory (On-chip flash memory version only)**

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	FLSHE	—	—	—
Initial value	0	0	0	0	0	0	0	0
R/W	—	—	—	—	R/W	—	—	—

Flash memory control register enable

0	Flash memory control registers are unselected
1	Flash memory control registers are selected



Note: * Can be written to only when the write condition is satisfied.

TCW—Timer counter W **H'91** **Watchdog timer (On-chip flash memory version only)**

Bit	:	7	6	5	4	3	2	1	0
		TCW7	TCW6	TCW5	TCW4	TCW3	TCW2	TCW1	TCW0
Initial value	:	0	0	0	0	0	0	0	0
R/W	:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Count value

TMW—Timer mode register W **H'92** **Watchdog timer (On-chip flash memory version only)**

Bit	:	7	6	5	4	3	2	1	0
		—	—	—	—	—	CKS2	CKS1	CKS0
Initial value	:	1	1	1	1	1	1	1	1
R/W	:	—	—	—	—	—	R/W	R/W	R/W

Clock select

Bit 2	Bit 1	Bit 0	Description
CKS2	CKS1	CKS0	
0	0	0	Internal clock: ø/64
0	0	1	Internal clock: ø/128
0	1	0	Internal clock: ø/256
0	1	1	Internal clock: ø/512
1	0	0	Internal clock: ø/1024
1	0	1	Internal clock: ø/2048
1	1	0	Internal clock: ø/4096
1	1	1	Internal clock: ø/8192 (initial value)

Note: TMW is an 8-bit read/write register that selects the input clock. Upon reset, TMW is initialized to H'FF.

Bit	7	6	5	4	3	2	1	0
	SNC1	SNC0	—	—	CKS3	CKS2	CKS1	CKS0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Clock Select (CKS2 to CKS0)

Bit 2	Bit 1	Bit 0	Prescaler Division	Serial Clock Cycle: Synchronous ϕ = 5 MHz	Serial Clock Cycle: Synchronous ϕ = 2.5 MHz
CKS2	CKS1	CKS0			
0	0	0	ϕ/1024	204.8 µs	409.6 µs
		1	ϕ/256	51.2 µs	102.4 µs
	1	0	ϕ/64	12.8 µs	25.6 µs
		1	ϕ/32	6.4 µs	12.8 µs
1	0	0	ϕ/16	3.2 µs	6.4 µs
		1	ϕ/8	1.6 µs	3.2 µs
	1	0	ϕ/4	0.8 µs	1.6 µs
		1	ϕ/2	—	0.8 µs

Clock source select

0	Clock source is prescaler S, and pin SCK_1 is output pin
1	Clock source is external clock, and pin SCK_1 is input pin

Operation mode select

0	0	8-bit synchronous transfer mode
	1	16-bit synchronous transfer mode
1	0	Continuous clock output mode
	1	Reserved

Bit	7	6	5	4	3	2	1	0
	—	SOL	ORER	—	—	—	—	STF
Initial value	1	0	0	0	0	0	0	0
Read/Write	—	R/W	R/(W)*	—	—	—	—	R/W

Start flag

0	Read	Indicates that transfer is stopped
	Write	Invalid
1	Read	Indicates transfer in progress
	Write	Starts a transfer operation

Overrun error flag

0	[Clearing condition] After reading 1, cleared by writing 0
1	[Setting condition] Set if a clock pulse is input after transfer is complete, when an external clock is used

Extended data bit

0	Read	SO_1 pin output level is low
	Write	SO_1 pin output level changes to low
1	Read	SO_1 pin output level is high
	Write	SO_1 pin output level changes to high

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

SDRU—Serial data register U **H'A2** **SCI1**
(H8/3857 Series only)

Bit	7	6	5	4	3	2	1	0
	SDRU7	SDRU6	SDRU5	SDRU4	SDRU3	SDRU2	SDRU1	SDRU0
Initial value	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

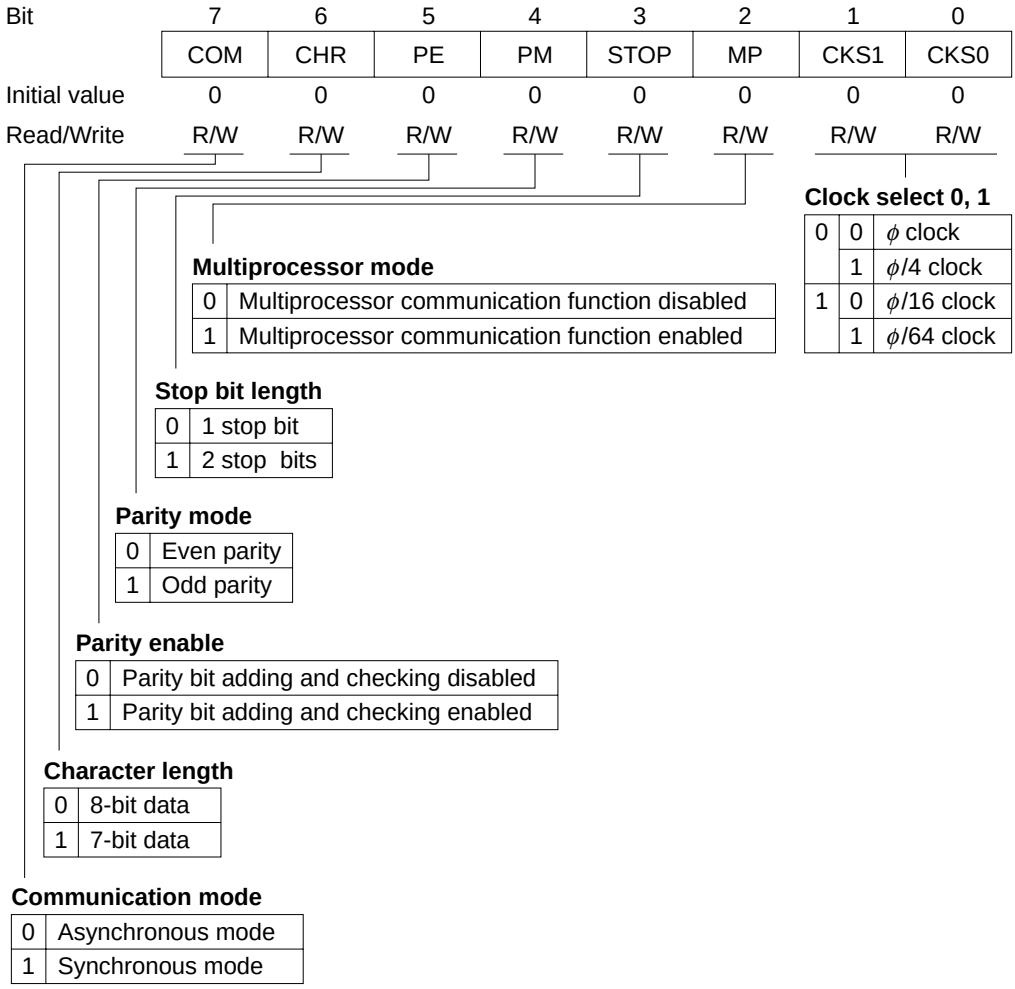
Used to set transmit data and store receive data
8-bit transfer mode: Not used
16-bit transfer mode: Upper 8 bits of data

SDRL—Serial data register L **H'A3** **SCI1**
(H8/3857 Series only)

Bit	7	6	5	4	3	2	1	0
	SDRL7	SDRL6	SDRL5	SDRL4	SDRL3	SDRL2	SDRL1	SDRL0
Initial value	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Used to set transmit data and store receive data
8-bit transfer mode: 8-bit data
16-bit transfer mode: Lower 8 bits of data

SMR—Serial mode register **H'A8** **SCI3**



BRR—Bit rate register **H'A9** **SCI3**

Bit	7	6	5	4	3	2	1	0
	BRR7	BRR6	BRR5	BRR4	BRR3	BRR2	BRR1	BRR0
Initial value	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	7	6	5	4	3	2	1	0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Clock enable

Bit 1	Bit 0	Description		
CKE1	CKE0	Communication Mode	Clock Source	SCK_3 Pin Function
0	0	Asynchronous	Internal clock	I/O port
		Synchronous	Internal clock	Serial clock output
	1	Asynchronous	Internal clock	Clock output
		Synchronous	Reserved (Do not set this combination)	Reserved (Do not set this combination)
1	0	Asynchronous	External clock	Clock input
		Synchronous	External clock	Serial clock input
	1	Asynchronous	Reserved (Do not set this combination)	Reserved (Do not set this combination)
		Synchronous	Reserved (Do not set this combination)	Reserved (Do not set this combination)

Transmit end interrupt enable

0	Transmit end interrupt (TEI) disabled
1	Transmit end interrupt (TEI) enabled

Multiprocessor interrupt enable

0	Multiprocessor interrupt request disabled (ordinary receive operation) [Clearing condition] Multiprocessor bit receives a data value of 1
1	Multiprocessor interrupt request enabled Until a multiprocessor bit value of 1 is received, the receive data full interrupt (RXI) and receive error interrupt (ERI) are disabled, and serial status register (SSR) flags RDRF, FER, and OER are not set.

Receive enable

0	Receive operation disabled (RXD is a general I/O port)
1	Receive operation enabled (RXD is the receive data pin)

Transmit enable

0	Transmit operation disabled (TXD is a general I/O port)
1	Transmit operation enabled (TXD is the transmit data pin)

Receive interrupt enable

0	Receive data full interrupt request (RXI) and receive error interrupt request (ERI) disabled
1	Receive data full interrupt request (RXI) and receive error interrupt request (ERI) enabled

Transmit interrupt enable

0	Transmit data empty interrupt request (TXI) disabled
1	Transmit data empty interrupt request (TXI) enabled

TDR—Transmit data register **H'AB** **SCI3**

Bit	7	6	5	4	3	2	1	0
	TDR7	TDR6	TDR5	TDR4	TDR3	TDR2	TDR1	TDR0
Initial value	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Data to be transferred to TSR

Bit	7	6	5	4	3	2	1	0
	TDRE	RDRF	OER	FER	PER	TEND	MPBR	MPBT
Initial value	1	0	0	0	0	1	0	0
Read/Write	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R/W

Multiprocessor bit receive

0	Indicates reception of data in which the multiprocessor bit is 0
1	Indicates reception of data in which the multiprocessor bit is 1

Multiprocessor bit transmit

0	The multiprocessor bit in transmit data is 0
1	The multiprocessor bit in transmit data is 1

Transmit end

0	Indicates that transmission is in progress [Clearing conditions] After reading TDRE = 1, cleared by writing 0 to TDRE. When data is written to TDR by an instruction.
1	Indicates that a transmission has ended [Setting conditions] When bit TE in serial control register 3 (SCR3) is 0. If TDRE is set to 1 when the last bit of a transmitted character is sent.

Parity error

0	Indicates that data receiving is in progress or has been completed [Clearing conditions] After reading PER = 1, cleared by writing 0
1	Indicates that a parity error occurred in data receiving [Setting conditions] When the sum of 1s in received data plus the parity bit does not match the parity mode bit (PM) setting in the serial mode register (SMR)

Framing error

0	Indicates that data receiving is in progress or has been completed [Clearing conditions] After reading FER = 1, cleared by writing 0
1	Indicates that a framing error occurred in data receiving [Setting conditions] The stop bit at the end of receive data is checked and found to be 0

Overrun error

0	Indicates that data receiving is in progress or has been completed [Clearing conditions] After reading OER = 1, cleared by writing 0
1	Indicates that an overrun error occurred in data receiving [Setting conditions] When reception of the next serial data is completed while RDRF is set to 1

Receive data register full

0	Indicates there is no receive data in RDR [Clearing conditions] After reading RDRF = 1, cleared by writing 0. When data is read from RDR by an instruction.
1	Indicates that there is receive data in RDR [Setting conditions] When receiving ends normally, with receive data transferred from RSR to RDR

Transmit data register empty

0	Indicates that transmit data written to TDR has not been transferred to TSR [Clearing conditions] After reading TDRE = 1, cleared by writing 0. When data is written to TDR by an instruction.
1	Indicates that no transmit data has been written to TDR, or the transmit data written to TDR has been transferred to TSR [Setting conditions] When bit TE in serial control register 3 (SCR3) is 0. When data is transferred from TDR to TSR.

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

RDR—Receive data register **H'AD** **SCI3**

Bit	7	6	5	4	3	2	1	0
	RDR7	RDR6	RDR5	RDR4	RDR3	RDR2	RDR1	RDR0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R	R

TMA—Timer mode register A **H'B0** **Timer A**

Bit	7	6	5	4	3	2	1	0
	TMA7	TMA6	TMA5	—	TMA3	TMA2	TMA1	TMA0
Initial value	0	0	0	1	0	0	0	0
Read/Write	R/W	R/W	R/W	—	R/W	R/W	R/W	R/W

Clock output select

0	0	0	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

Internal clock select

TMA3	TMA2	TMA1	TMA0	Prescaler and Divider Ratio or Overflow Period		Function
0	0	0	0	PSS	ϕ/8192	Interval timer
			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	Time base
			1	PSW	0.5 s	
		1	0	PSW	0.25 s	
			1	PSW	0.03125 s	
	1	0	0	PSW and TCA are reset		
			1			
		1	0			
			1			

TCA—Timer counter A **H'B1** **Timer A**

Bit	7	6	5	4	3	2	1	0
	TCA7	TCA6	TCA5	TCA4	TCA3	TCA2	TCA1	TCA0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R	R

Count value

TMB—Timer mode register B **H'B2** **Timer B**

Bit	7	6	5	4	3	2	1	0
	TMB7	—	—	—	—	TMB2	TMB1	TMB0
Initial value	0	1	1	1	1	0	0	0
Read/Write	R/W	—	—	—	—	R/W	R/W	R/W

Auto-reload function select

0	Interval timer function selected
1	Auto-reload function selected

Clock select

0	0	0	Internal clock: ϕ/8192
		1	Internal clock: ϕ/2048
	1	0	Internal clock: ϕ/512
		1	Internal clock: ϕ/256
1	0	0	Internal clock: ϕ/64
		1	Internal clock: ϕ/16
	1	0	Internal clock: ϕ/4
		1	External event (TMIB): Rising or falling edge

TCB—Timer counter B **H'B3** **Timer B**

Bit	7	6	5	4	3	2	1	0
	TCB7	TCB6	TCB5	TCB4	TCB3	TCB2	TCB1	TCB0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R	R

Count value

TLB—Timer load register B **H'B3** **Timer B**

Bit	7	6	5	4	3	2	1	0
	TLB7	TLB6	TLB5	TLB4	TLB3	TLB2	TLB1	TLB0
Initial value	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W

Reload value

TMC—Timer mode register C **H'B4** **Timer C**
(H8/3857 Series only)

Bit	7	6	5	4	3	2	1	0
	TMC7	TMC6	TMC5	—	—	TMC2	TMC1	TMC0
Initial value	0	0	0	1	1	0	0	0
Read/Write	R/W	R/W	R/W	—	—	R/W	R/W	R/W

Auto-reload function select

0	Interval timer function selected
1	Auto-reload function selected

Clock select

0	0	0	Internal clock: ϕ/8192
		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: ϕ_W/4
		1	External event (TMIC): Rising or falling edge

Counter up/down control

0	0	TCC is an up-counter
	1	TCC is a down-counter
1	*	TCC up/down operation is hardware-controlled by input at the UD pin. TCC is a down-counter if the UD input is high, and an up-counter if the UD input is low.

Note: * Don't care

TCC—Timer counter C **H'B5** **Timer C**
(H8/3857 Series only)

Bit	7	6	5	4	3	2	1	0
	TCC7	TCC6	TCC5	TCC4	TCC3	TCC2	TCC1	TCC0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R	R

Count value

TLC—Timer load register C **H'B5** **Timer C**
(H8/3857 Series only)

Bit	7	6	5	4	3	2	1	0
	TLC7	TLC6	TLC5	TLC4	TLC3	TLC2	TLC1	TLC0
Initial value	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W

Reload value

TCRF—Timer control register F **H'B6** **Timer F**

Bit	7	6	5	4	3	2	1	0
	TOLH	CKSH2	CKSH1	CKSH0	TOLL	CKSL2	CKSL1	CKSL0
Initial value	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W

Toggle output level H

0	Low level
1	High level

Clock select L

0	*	*	External event (TMIF): Rising or falling edge
1	0	0	Internal clock: ϕ/32
		1	Internal clock: ϕ/16
	1	0	Internal clock: ϕ/4
		1	Internal clock: ϕ/2

Toggle output level L

0	Low level
1	High level

Clock select H

0	*	*	16-bit mode selected. TCFL overflow signals are counted.
1	0	0	Internal clock: ϕ/32
		1	Internal clock: ϕ/16
	1	0	Internal clock: ϕ/4
		1	Internal clock: ϕ/2

Note: * Don't care

Bit	7	6	5	4	3	2	1	0
	OVFH	CMFH	OVIEH	CCLRH	OVFL	CMFL	OVIEL	CCLRL
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/(W)*	R/(W)*	R/W	R/W	R/(W)*	R/(W)*	R/W	R/W

Timer overflow interrupt enable L

0	TCFL overflow interrupt disabled
1	TCFL overflow interrupt enabled

Compare match flag L

0	[Clearing condition] After reading CMFL = 1, cleared by writing 0 to CMFL
1	[Setting condition] 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] When the value of TCFL goes from H'FF to H'00

Counter clear H

0	16-bit mode: TCF clearing by compare match disabled 8-bit mode: TCFH clearing by compare match disabled
1	16-bit mode: TCF clearing by compare match enabled 8-bit mode: TCFH clearing by compare match enabled

Timer overflow interrupt enable H

0	TCFH overflow interrupt disabled
1	TCFH overflow interrupt enabled

Counter clear L

0	TCFL clearing by compare match disabled
1	TCFL clearing by compare match enabled

Compare match flag H

0	[Clearing condition] After reading CMFH = 1, cleared by writing 0 to CMFH
1	[Setting condition] 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] When the value of TCFH goes from H'FF to H'00

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

TCFH—8-bit timer counter FH **H'B8** **Timer F**

Bit	7	6	5	4	3	2	1	0
	TCFH7	TCFH6	TCFH5	TCFH4	TCFH3	TCFH2	TCFH1	TCFH0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Count value

TCFL—8-bit timer counter FL **H'B9** **Timer F**

Bit	7	6	5	4	3	2	1	0
	TCFL7	TCFL6	TCFL5	TCFL4	TCFL3	TCFL2	TCFL1	TCFL0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Count value

OCRFH—Output compare register FH **H'BA** **Timer F**

Bit	7	6	5	4	3	2	1	0
	OCRFH7	OCRFH6	OCRFH5	OCRFH4	OCRFH3	OCRFH2	OCRFH1	OCRFH0
Initial value	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

OCRFL—Output compare register FL **H'BB** **Timer F**

Bit	7	6	5	4	3	2	1	0
	OCRFL7	OCRFL6	OCRFL5	OCRFL4	OCRFL3	OCRFL2	OCRFL1	OCRFL0
Initial value	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	7	6	5	4	3	2	1	0
	CKS	TRGE	—	—	CH3	CH2	CH1	CH0
Initial value	0	0	1	1	0	0	0	0
Read/Write	R/W	R/W	—	—	R/W	R/W	R/W	R/W

Channel select

Bit 3	Bit 2	Bit 1	Bit 0	
CH3	CH2	CH1	CH0	Analog input channel
0	0	*	*	No channel selected
	1	0	0	AN_0 *1
			1	AN_1 *1
		1	0	AN_2 *1
			1	AN_3 *1
1	0	0	0	AN_4
			1	AN_5
		1	0	AN_6
			1	AN_7
1	1	*	*	Reserved

External trigger select

0	Disables start of A/D conversion by external trigger
1	Enables start of A/D conversion by rising or falling edge of external trigger at pin $\overline{\text{ADTRG}}$

Clock select

Bit 7		Conversion Time	
CKS	Conversion Period	ϕ = 2 MHz	ϕ = 5 MHz
0	62/ϕ	31 µs	12.4 µs
1	31/ϕ	15.5 µs	—*2

Notes: * Don't care

1. AN_0 to AN_3 can be selected in the H8/3857 Series only. They must not be selected in the H8/3854 Series.
2. Operation is not guaranteed if the conversion time is less than 12.4 µs. Set bit 7 for a value of at least 12.4 µs.

ADRR—A/D result register **H'C5** **A/D converter**

Bit	7	6	5	4	3	2	1	0
	ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0
Initial value	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
Read/Write	R	R	R	R	R	R	R	R

A/D conversion result

ADSR—A/D start register **H'C6** **A/D converter**

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

A/D status flag

0	Read	Indicates the completion of A/D conversion
	Write	Stops A/D conversion
1	Read	Indicates A/D conversion in progress
	Write	Starts A/D conversion

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

$P1_0$/TMOW pin function switch

0	Functions as $P1_0$ I/O pin
1	Functions as TMOW output pin

$P1_1$/TMOFL pin function switch

0	Functions as $P1_1$ I/O pin
1	Functions as TMOFL output pin

$P1_2$/TMOFH pin function switch

0	Functions as $P1_2$ I/O pin
1	Functions as TMOFH output pin

$P1_4$/PWM pin function switch

0	Functions as $P1_4$ I/O pin
1	Functions as PWM output pin

$P1_5/\overline{IRQ}_1$/TMIB pin function switch

0	Functions as $P1_5$ I/O pin
1	Functions as $\overline{IRQ}_1$/TMIB input pin

$P1_6/\overline{IRQ}_2$/TMIC pin function switch

0	Functions as $P1_6$ I/O pin
1	Functions as $\overline{IRQ}_2$/TMIC input pin

$P1_7/\overline{IRQ}_3$/TMIF pin function switch

0	Functions as $P1_7$ I/O pin
1	Functions as $\overline{IRQ}_3$/TMIF input pin

Note: * IRQ2 and PWM are functions of the H8/3857 Series only.
In the H8/3854 Series these bits are reserved, and must always be cleared to 0.

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

$P2_0$/$\overline{IRQ_4}$/$\overline{ADTRG}$ pin function switch

0	Functions as $P2_0$ I/O pin
1	Functions as $\overline{IRQ_4}$/$\overline{ADTRG}$ input pin

$P2_1$/UD pin function switch

0	Functions as $P2_1$ I/O pin
1	Functions as UD input pin

$P3_2$/SO_1 pin PMOS control

0	CMOS output
1	NMOS open-drain output

$P4_3$/$\overline{IRQ_0}$ pin function switch

0	Functions as $P4_3$ I/O pin
1	Functions as $\overline{IRQ_0}$ input pin

Note: * POF1 and UD are functions of the H8/3857 Series only.
In the H8/3854 Series these bits are reserved, and must always be cleared to 0.

PMR3—Port mode register 3 **H'CA** **I/O ports**
(H8/3857 Series only)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	SO1	SI1	SCK1
Initial value	0	0	0	0	0	0	0	0
Read/Write	—	—	—	—	—	R/W	R/W	R/W

$P3_0/SCK_1$ pin function switch

0	Functions as $P3_0$ I/O pin
1	Functions as SCK_1 I/O pin

$P3_1/SI_1$ pin function switch

0	Functions as $P3_1$ I/O pin
1	Functions as SI_1 input pin

$P3_2/SO_1$ pin function switch

0	Functions as $P3_2$ I/O pin
1	Functions as SO_1 output pin

PMR4—Port mode register 4 **H'CB** **I/O ports**

Bit	7	6	5	4	3	2	1	0
	NMOD7	NMOD6	NMOD5	NMOD4	NMOD3	NMOD2	NMOD1	NMOD0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

0	$P2_n$ has CMOS output
1	$P2_n$ has NMOS open-drain output

PMR5—Port mode register 5 **H'CC** **I/O ports**

Bit	7	6	5	4	3	2	1	0
	WKP_7	WKP_6	WKP_5	WKP_4	WKP_3	WKP_2	WKP_1	WKP_0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

$P5_n/\overline{WKP_n}$ pin function switch

0	Functions as $P5_n$ I/O pin
1	Functions as $\overline{WKP_n}$ input pin

PWCR—PWM control register **H'D0** **14-bit PWM (H8/3857 Series only)**

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PWCR0
Initial value	1	1	1	1	1	1	1	0
Read/Write	—	—	—	—	—	—	—	W

Clock select

0	The input clock is $\phi/2$ ($t\phi^* = 2/\phi$). The conversion period is 16,384/ϕ, with a minimum modulation width of 1/ϕ
1	The input clock is $\phi/4$ ($t\phi^* = 4/\phi$). The conversion period is 32,768/ϕ, with a minimum modulation width of 2/ϕ

Note: *tø: Period of PWM input clock

PWDRU—PWM data register U **H'D1** **14-bit PWM (H8/3857 Series only)**

Bit	7	6	5	4	3	2	1	0
	—	—	PWDRU5	PWDRU4	PWDRU3	PWDRU2	PWDUR1	PWDRU0
Initial value	1	1	0	0	0	0	0	0
Read/Write	—	—	W	W	W	W	W	W

Upper 6 bits of data for generating PWM waveform

PWDRL—PWM data register L **H'D2** **14-bit PWM (H8/3857 Series only)**

Bit	7	6	5	4	3	2	1	0
	PWDRL7	PWDRL6	PWDRL5	PWDRL4	PWDRL3	PWDRL2	PWDRL1	PWDRL0
Initial value	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W

Lower 8 bits of data for generating PWM waveform

PDR1—Port data register 1 **H'D4** **I/O ports**

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

Note: * $P1_6$, $P1_4$, and $P1_3$ are functions of the H8/3857 Series only.
In the H8/3854 Series these bits are reserved, and must always be set to 1.

PDR2—Port data register 2 **H'D5** **I/O ports**

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

PDR3—Port data register 3 **H'D6** **I/O ports (H8/3857 Series only)**

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

PDR4—Port data register 4 **H'D7** **I/O ports**

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

PDR5—Port data register 5 **H'D8** **I/O ports**

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

PDR9—Port data register 9 **H'DC** **I/O ports**

Bit	7	6	5	4	3	2	1	0
	$P9_7$	$P9_6$	$P9_5$	$P9_4$	$P9_3$	$P9_2$	$P9_1$	$P9_0$
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PDRA—Port data register A **H'DD** **I/O ports**

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	PA_3	PA_2	PA_1	PA_0
Initial value	1	1	1	1	0	0	0	0
Read/Write	—	—	—	—	R/W	R/W	R/W	R/W

PDRB—Port data register B **H'DE** **I/O ports**

Bit	7	6	5	4	3	2	1	0
	PB_7	PB_6	PB_5	PB_4	PB_3*	PB_2*	PB_1*	PB_0*
Read/Write	R	R	R	R	R	R	R	R

Note: * PB_3 to PB_0 are functions of the H8/3857 Series only.
In the H8/3854 Series these bits are reserved.

PUCR1—Port pull-up control register 1 **H'E0** **I/O ports**

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

Note: * $PUCR1_6$, $PUCR1_4$, and $PUCR1_3$ are functions of the H8/3857 Series only.
In the H8/3854 Series these bits are reserved, and must always be cleared to 0.

PUCR3—Port pull-up control register 3 **H'E1** **I/O ports**
(H8/3857 Series only)

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

PUCR5—Port pull-up control register 5 **H'E2** **I/O ports**

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

PCR1—Port control register 1 **H'E4** **I/O ports**

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

Port 1 input/output select

0	Input pin
1	Output pin

Note: * $PCR1_6$, $PCR1_4$, and $PCR1_3$ are functions of the H8/3857 Series only.
In the H8/3854 Series these bits are reserved, and must always be cleared to 0.

PCR2—Port control register 2 **H'E5** **I/O ports**

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

Port 2 input/output select

0	Input pin
1	Output pin

PCR3—Port control register 3 **H'E6** **I/O ports (H8/3857 Series only)**

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

Port 3 input/output select

0	Input pin
1	Output pin

PCR4—Port control register 4 **H'E7** **I/O ports**

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

Port 4 input/output select

0	Input pin
1	Output pin

PCR5—Port control register 5 **H'E8** **I/O ports**

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

Port 5 input/output select

0	Input pin
1	Output pin

PCR9—Port control register 9 **H'EC** **I/O ports**

Bit	7	6	5	4	3	2	1	0
	$PCR9_7$	$PCR9_6$	$PCR9_5$	$PCR9_4$	$PCR9_3$	$PCR9_2$	$PCR9_1$	$PCR9_0$
Initial value	0	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W	W

Port 9 input/output select

0	Input pin
1	Output pin

PCRA—Port control register A **H'ED** **I/O ports**

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	$PCRA_3$	$PCRA_2$	$PCRA_1$	$PCRA_0$
Initial value	1	1	1	1	0	0	0	0
Read/Write	—	—	—	—	W	W	W	W

Port A input/output select

0	Input pin
1	Output pin

SYSCR1—System control register 1 **H'F0** **System control**

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

Low speed on flag

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

Standby timer select 2 to 0

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

Software standby

0	When a SLEEP instruction is executed in active mode, a transition is made to sleep mode. When a SLEEP instruction is executed in subactive mode, a transition is made to subsleep mode.
1	When a SLEEP instruction is executed in active mode, a transition is made to standby mode or watch mode. When a SLEEP instruction is executed in subactive mode, a transition is made to watch mode.

Note: * Don’t care

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

Medium speed on flag

0	Operates in active (high-speed) mode
1	Operates in active (medium-speed) mode

Subactive mode clock select

0	0	$\phi_W/8$
	1	$\phi_W/4$
1	*	$\phi_W/2$

Direct transfer on flag

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

Noise elimination sampling frequency select

0	Sampling rate is $\phi_{OSC}/16$
1	Sampling rate is $\phi_{OSC}/4$

Note: * Don't care

IEGR—IRQ edge select register **H'F2** **System control**

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

$\overline{IRQ_0}$ edge select

0	Falling edge of $\overline{IRQ_0}$ pin input is detected
1	Rising edge of $\overline{IRQ_0}$ pin input is detected

$\overline{IRQ_1}$ edge select

0	Falling edge of $\overline{IRQ_1}$/TMIB pin input is detected
1	Rising edge of $\overline{IRQ_1}$/TMIB pin input is detected

$\overline{IRQ_2}$ edge select

0	Falling edge of $\overline{IRQ_2}$/TMIC pin input is detected
1	Rising edge of $\overline{IRQ_2}$/TMIC pin input is detected

$\overline{IRQ_3}$ edge select

0	Falling edge of $\overline{IRQ_3}$/TMIF pin input is detected
1	Rising edge of $\overline{IRQ_3}$/TMIF pin input is detected

$\overline{IRQ_4}$ edge select

0	Falling edge of $\overline{IRQ_4}$/$\overline{ADTRG}$ pin input is detected
1	Rising edge of $\overline{IRQ_4}$/$\overline{ADTRG}$ pin input is detected

Note: * IEG2 is a function of the H8/3857 Series only.
In the H8/3854 Series this bit is reserved, and must always be cleared to 0.

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

$\overline{IRQ}_4$ to $\overline{IRQ}_0$ interrupt enable

0	Disables interrupt request $\overline{IRQ}_n$
1	Enables interrupt request $\overline{IRQ}_n$

(n = 4 to 0)

Wakeup interrupt enable

0	Disables interrupt requests from $\overline{WKP}_7$ to $\overline{WKP}_0$
1	Enables interrupt requests from $\overline{WKP}_7$ to $\overline{WKP}_0$

SCI1 interrupt enable

0	Disables SCI1 interrupts
1	Enables SCI1 interrupts

Timer A interrupt enable

0	Disables timer A interrupts
1	Enables timer A interrupts

Note: * IENS1 and IEN2 are functions of the H8/3857 Series only.
In the H8/3854 Series these bits are reserved, and must always be cleared to 0.

Bit	7	6	5	4	3	2	1	0
	IENDT	IENAD	—	—	IENTFH	IENTFL	IENTC*	IENTB
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Timer B interrupt enable

0	Disables timer B interrupts
1	Enables timer B interrupts

Timer C interrupt enable

0	Disables timer C interrupts
1	Enables timer C interrupts

Timer FL interrupt enable

0	Disables timer FL interrupts
1	Enables timer FL interrupts

Timer FH interrupt enable

0	Disables timer FH interrupts
1	Enables timer FH interrupts

A/D converter interrupt enable

0	Disables A/D converter interrupt requests
1	Enables A/D converter interrupt requests

Direct transfer interrupt enable

0	Disables direct transfer interrupt requests
1	Enables direct transfer interrupt requests

Note: * IENTC is a function of the H8/3857 Series only.
In the H8/3854 Series this bit is reserved, and must always be cleared to 0.

IRR1—Interrupt request register 1 **H'F6** **System control**

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

IRQ_4 to IRQ_0 interrupt request flag

0	[Clearing condition] When IRRI4 = 1, it is cleared by writing 0 When 0 is written to IRRI4 when IRRI4 = 1 The same also applies to IRRI3—IRRI0
1	[Setting condition] When pin $\overline{IRQ_4}$ is set to interrupt input and the designated signal edge is detected When pin $\overline{IRQ_4}$ is set to interrupt input and the designated edge is input at this pin The same also applies to IRRI3—IRRI0

SCI1 interrupt request flag

0	[Clearing condition] When IRRS1 = 1, it is cleared by writing 0
1	[Setting condition] When an SCI1 transfer is completed

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 overflows from H'FF to H'00

Notes: 1. IRRS1 and IRRI2 are functions of the H8/3857 Series only.
In the H8/3854 Series these bits are reserved, and are always 0.
2. Only a write of 0 for flag clearing is possible.

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

Timer B interrupt request flag

0	[Clearing condition]	When IRRTB = 1, it is cleared by writing 0
1	[Setting condition]	When the timer B counter overflows from H'FF to H'00

Timer C interrupt request flag

0	[Clearing condition]	When IRRTC = 1, it is cleared by writing 0
1	[Setting condition]	When the timer C counter overflows from H'FF to H'00 or underflows from H'00 to H'FF

Timer FL interrupt request flag

0	[Clearing condition]	When IRRTFL = 1, it is cleared by writing 0
1	[Setting condition]	When counter FL matches output compare register FL in 8-bit mode

Timer FH interrupt request flag

0	[Clearing condition]	When IRRTFH = 1, it is cleared by writing 0
1	[Setting condition]	When counter FH matches output compare register FH in 8-bit mode, or when 16-bit counter F (TCFL, TCFH) matches 16-bit output compare register F (OCRFL, OCRFH) in 16-bit mode

A/D converter interrupt request flag

0	[Clearing condition]	When IRRAD = 1, it is cleared by writing 0
1	[Setting condition]	When A/D conversion is completed and ADSF is reset

Direct transfer interrupt request flag

0	[Clearing condition]	When IRRDT = 1, it is cleared by writing 0
1	[Setting condition]	A SLEEP instruction is executed when DTON = 1 and a direct transfer is made

Notes: 1. IRRTC is a function of the H8/3857 Series only.
In the H8/3854 Series this bit is reserved, and is always 0.
2. Only a write of 0 for flag clearing is possible.

IWPR—Wakeup interrupt request register **H'F9** **System control**

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

Wakeup interrupt request flag

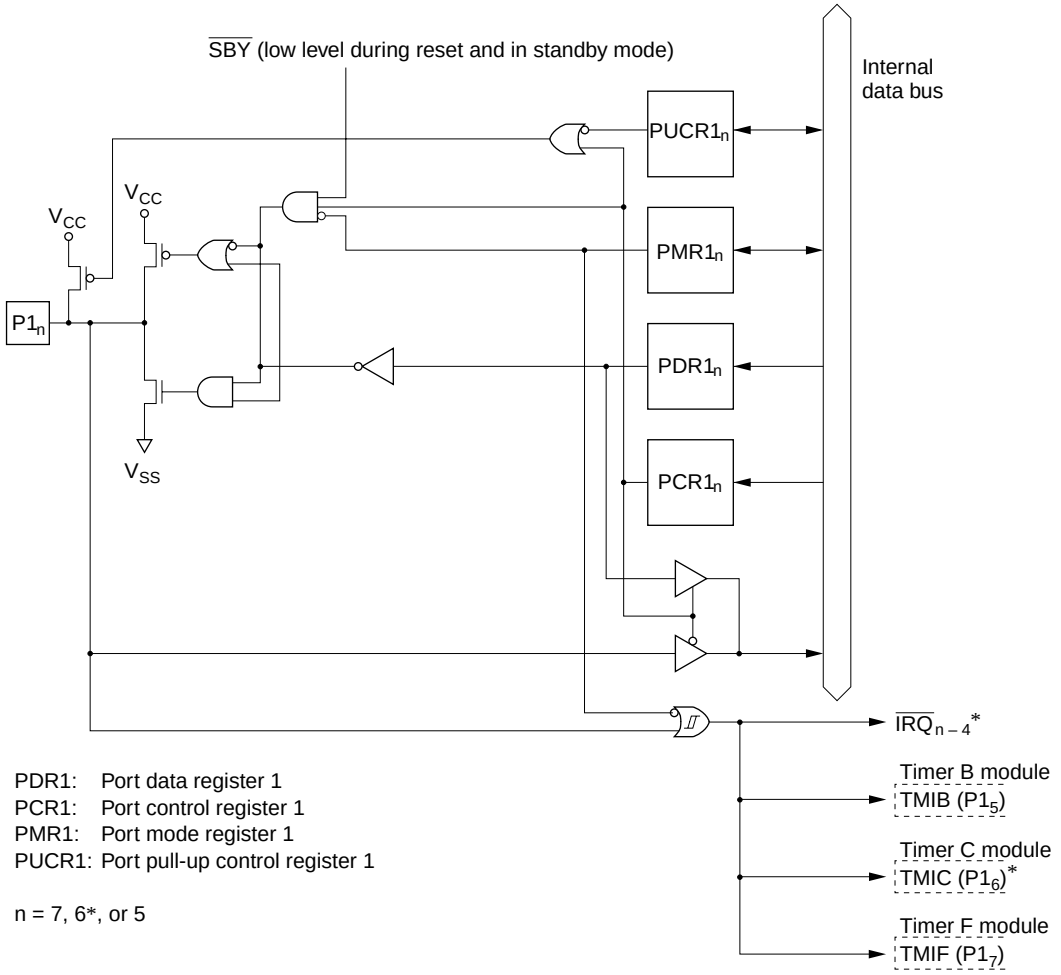
0	[Clearing condition] When IWPFn = 1, it is cleared by writing 0
1	[Setting condition] When pin $\overline{WKP}_n$ is set to interrupt input and a falling signal edge is detected

(n = 7 to 0)

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

Appendix C I/O Port Block Diagrams

C.1 Block Diagram of Port 1



Note: * $P1_6$, the timer C module (TMIC), and n = 6 ($PDR1_6$, $PCR1_6$, $PMR1_6$, $PUCR1_6$, $\overline{IRQ}_2$) are functions of the H8/3857 Series only, and are not provided in the H8/3854 Series.

Figure C.1 (a) Port 1 Block Diagram
(Pins $P1_7$ to $P1_5$: H8/3857 Series, Pins $P1_7$, $P1_5$: H8/3854 Series)

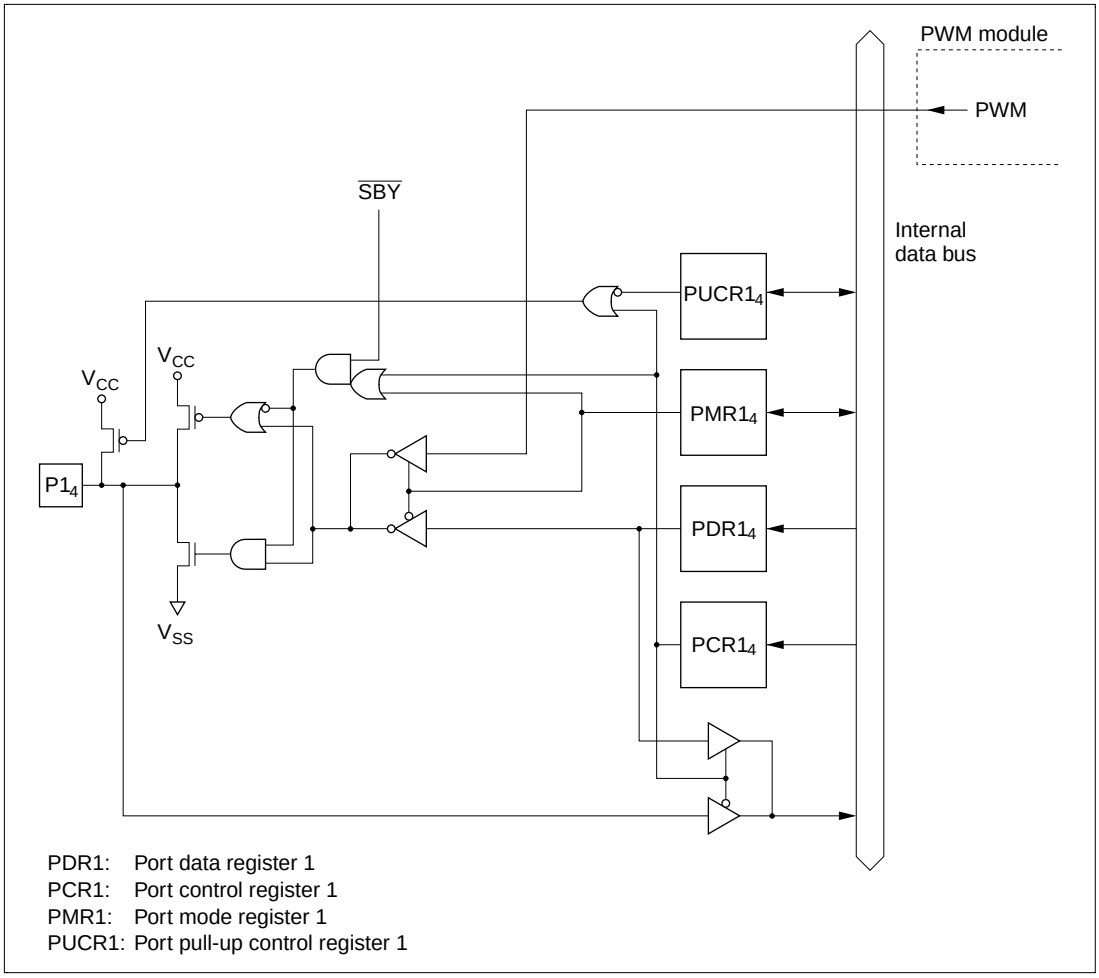


Figure C.1 (b) Port 1 Block Diagram (Pin $P1_4$: Function of H8/3857 Series Only)

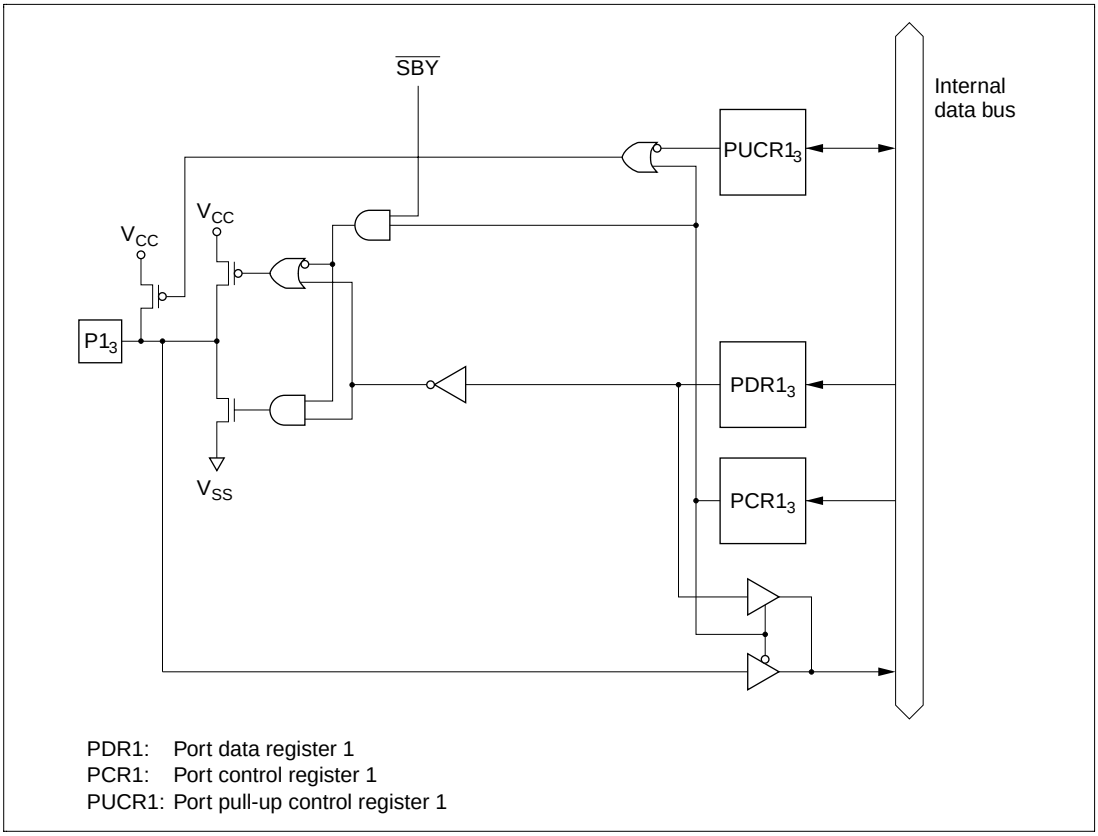


Figure C.1 (c) Port 1 Block Diagram (Pin $P1_3$: Function of H8/3857 Series Only)

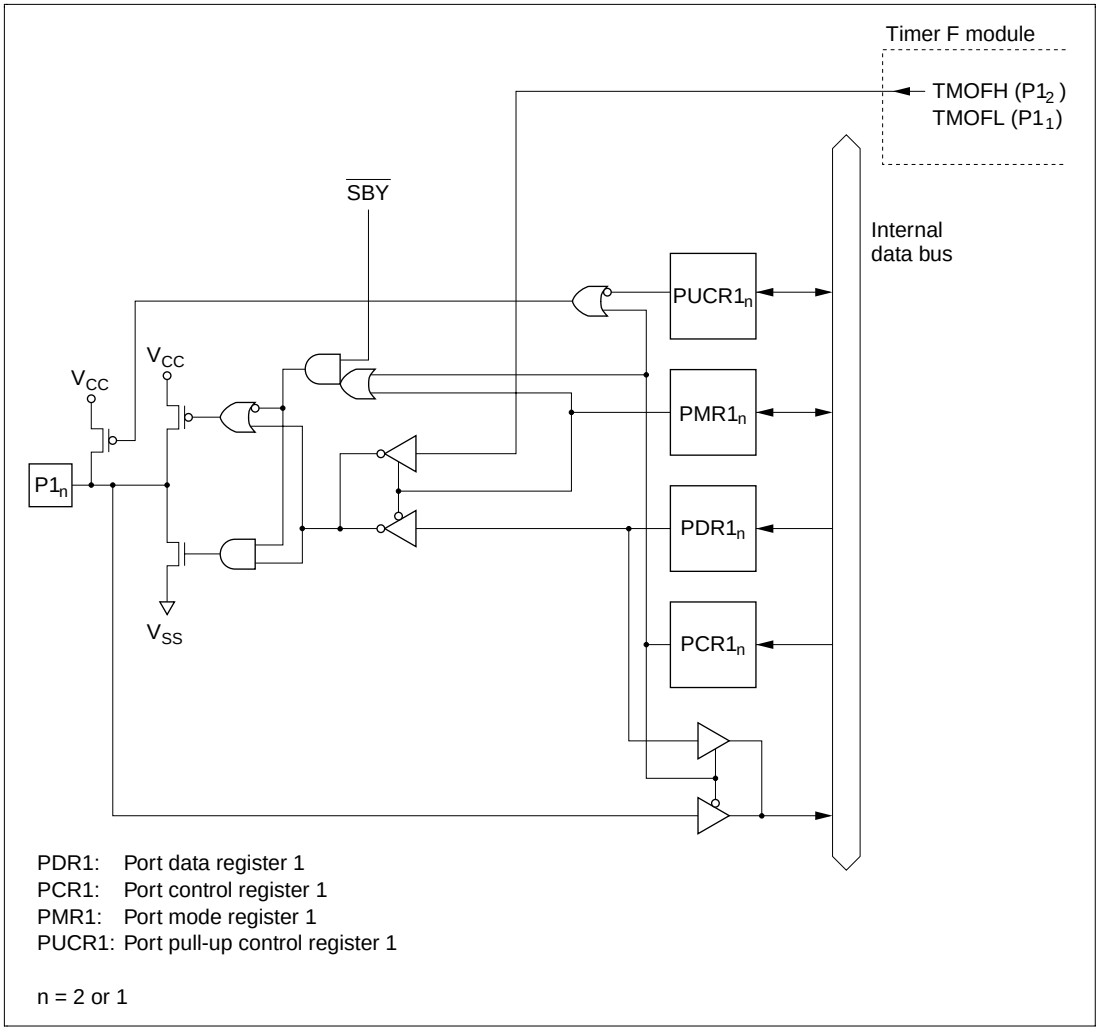


Figure C.1 (d) Port 1 Block Diagram (Pins $P1_2$ and $P1_1$)

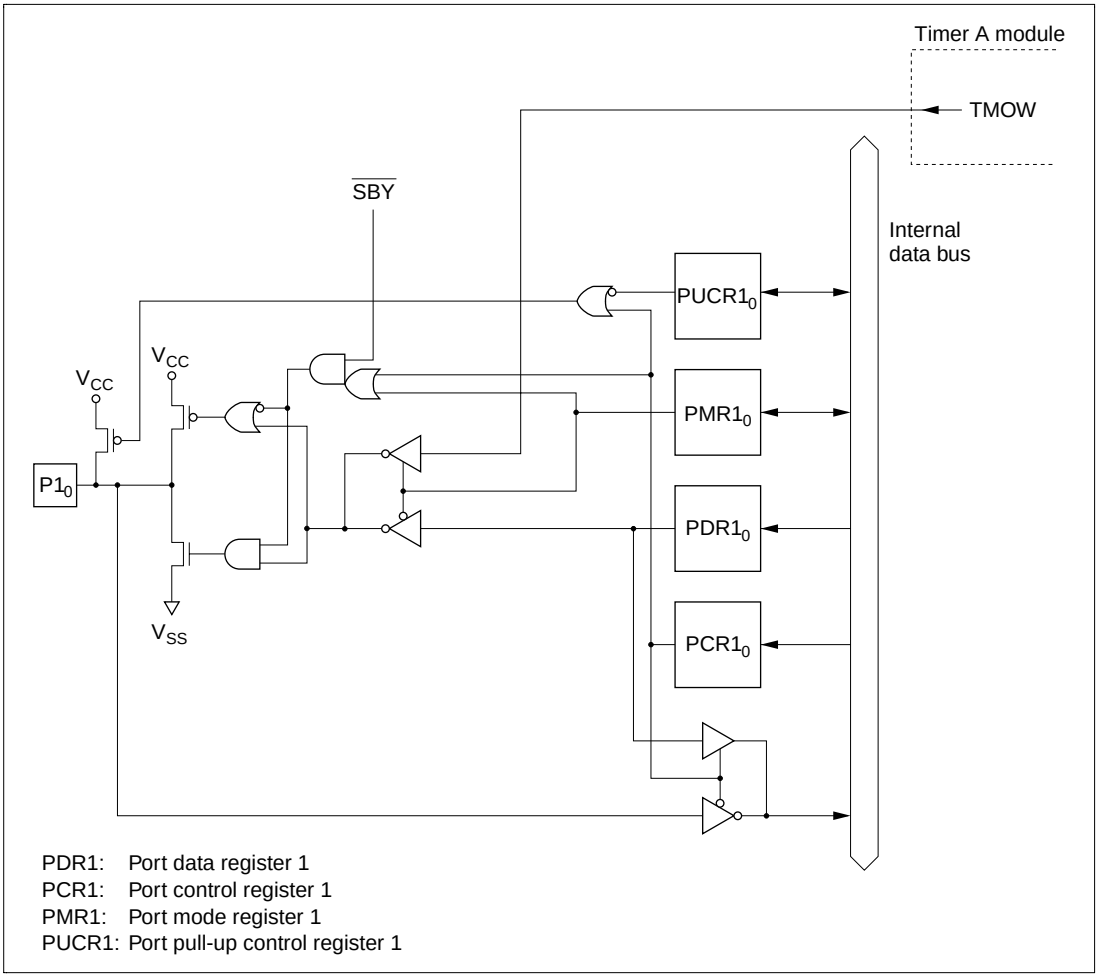


Figure C.1 (e) Port 1 Block Diagram (Pin $P1_0$)

C.2 Block Diagram of Port 2

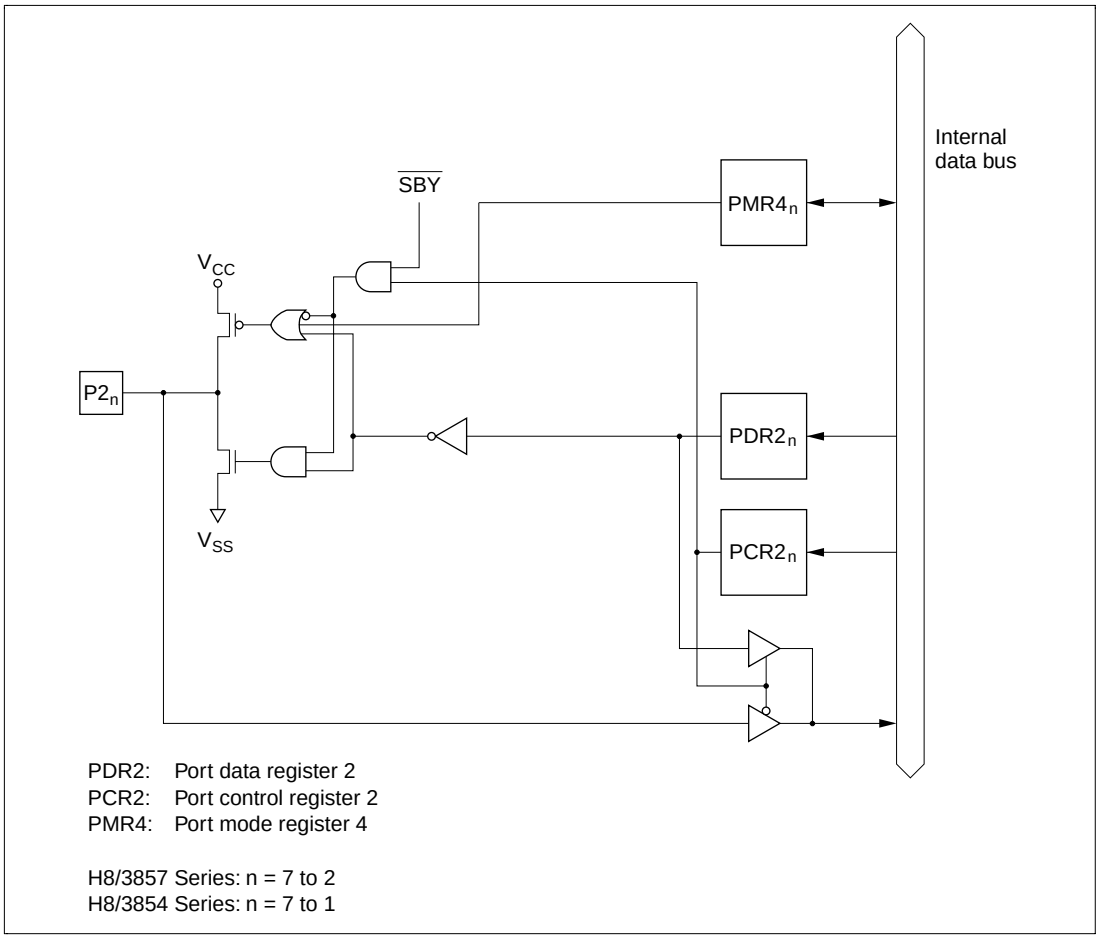


Figure C.2 (a) Port 2 Block Diagram
(Pins $P2_7$ to $P2_2$: H8/3857 Series; Pins $P2_7$ to $P2_1$: H8/3854 Series)

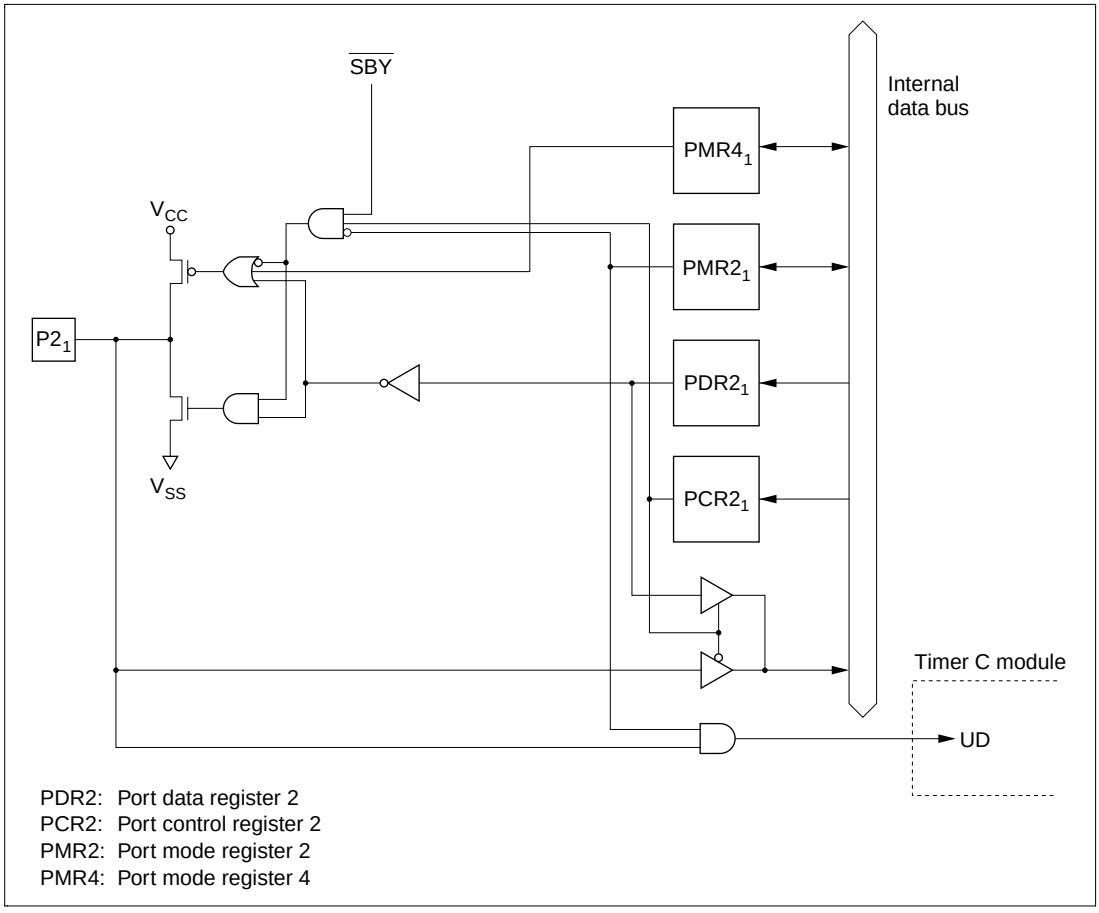


Figure C.2 (b) Port 2 Block Diagram (Pin $P2_1$: H8/3857 Series)

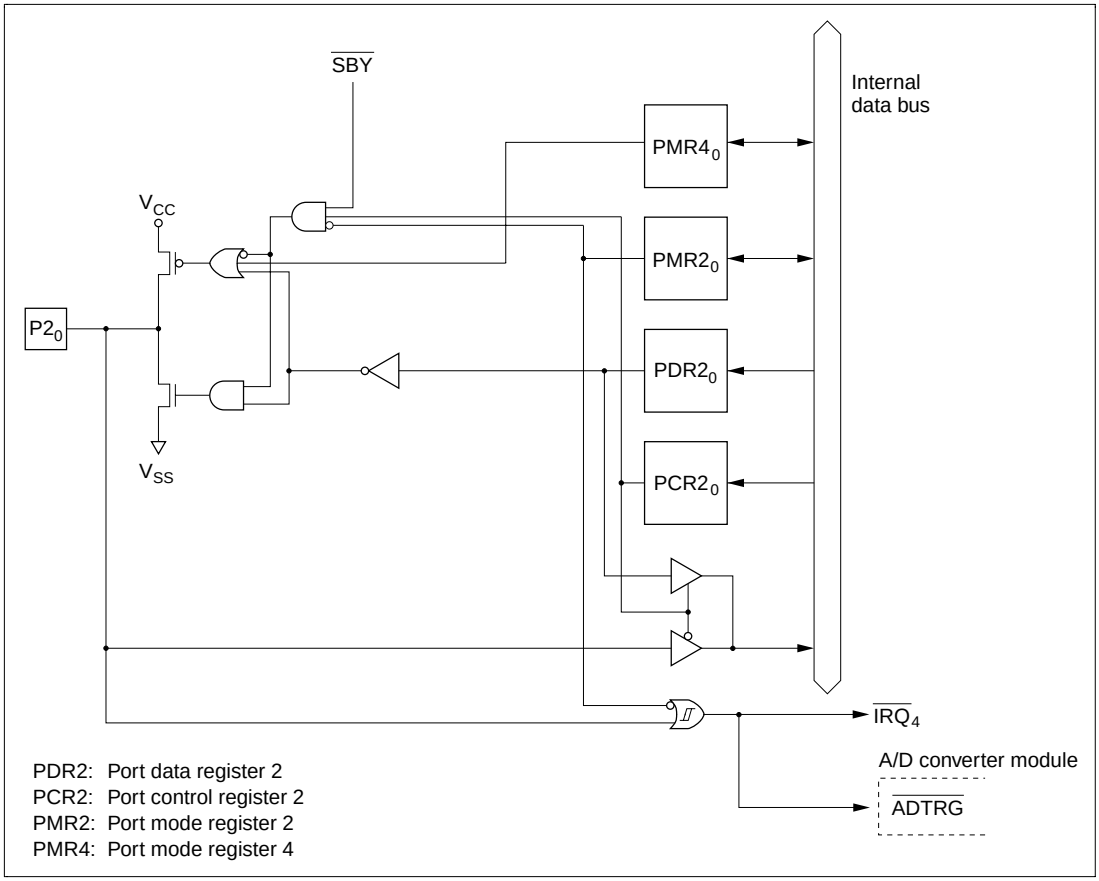


Figure C.2 (c) Port 2 Block Diagram (Pin $P2_0$)

C.3 Block Diagram of Port 3 (H8/3857 Series Only)

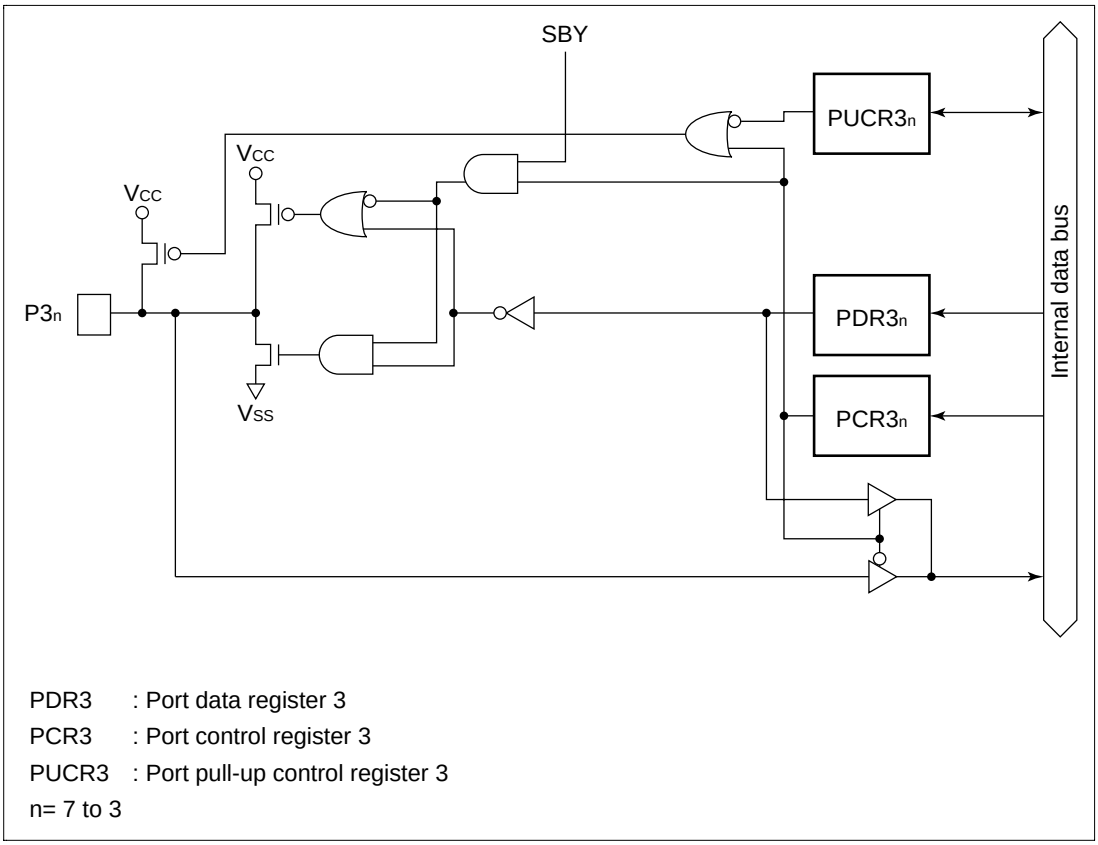


Figure C.3 (a) Port 3 Block Diagram (Pins $P3_7$ to $P3_3$: Functions of H8/3857 Series Only)

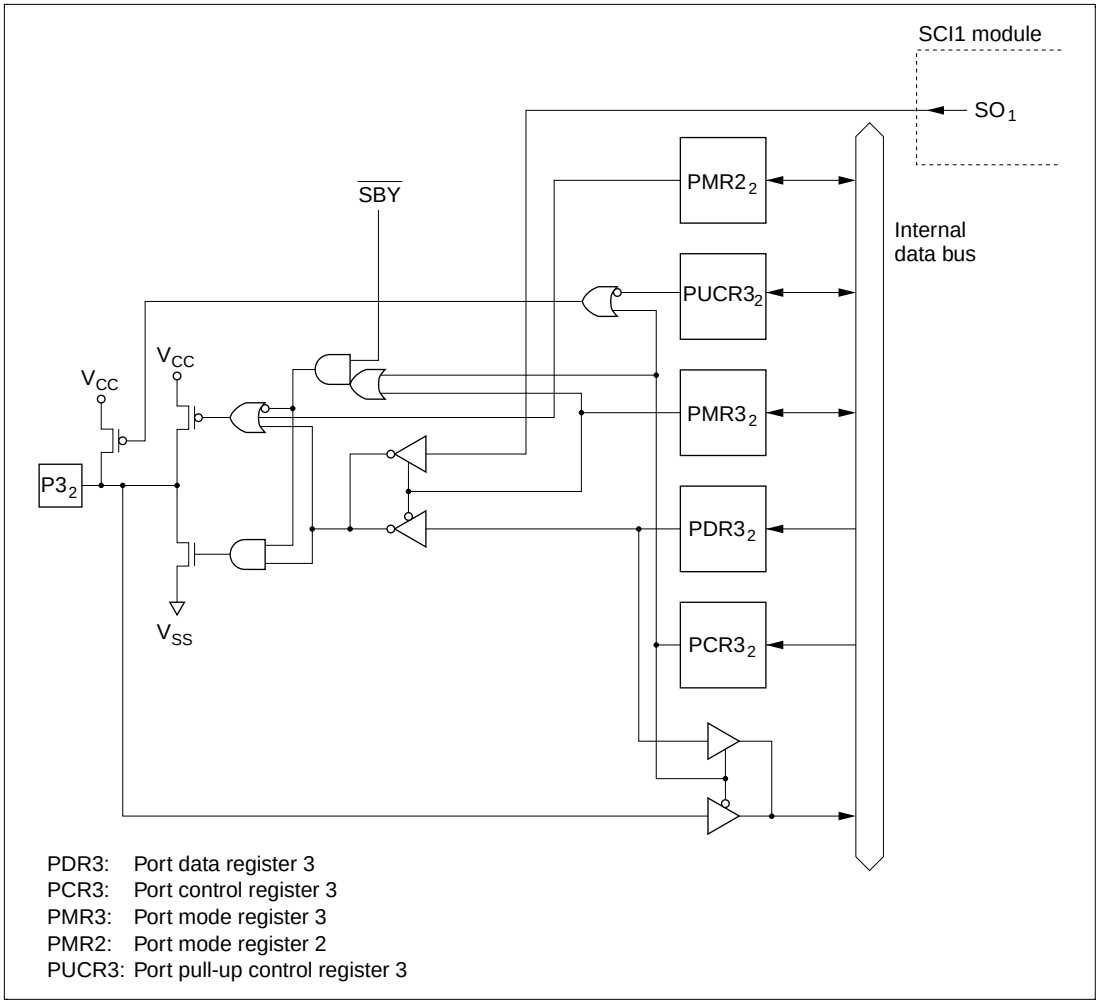


Figure C.3 (b) Port 3 Block Diagram (Pin $P3_2$: Function of H8/3857 Series Only)

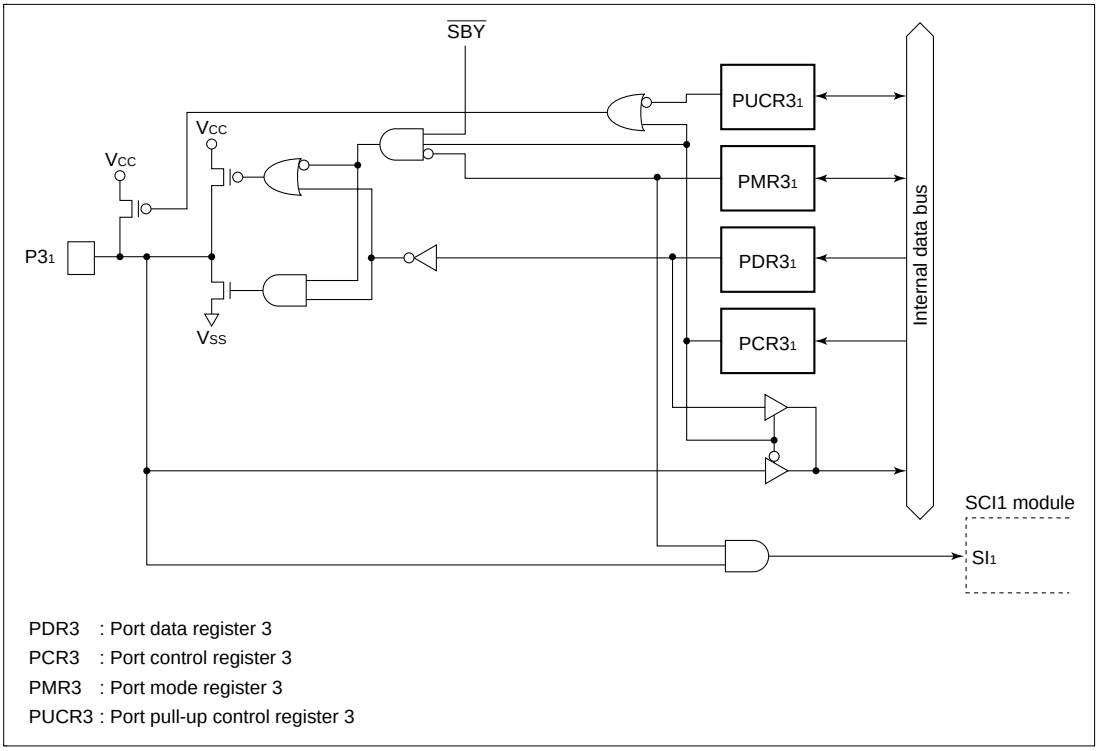


Figure C.3 (c) Port 3 Block Diagram (Pin $P3_1$: Function of H8/3857 Series Only)

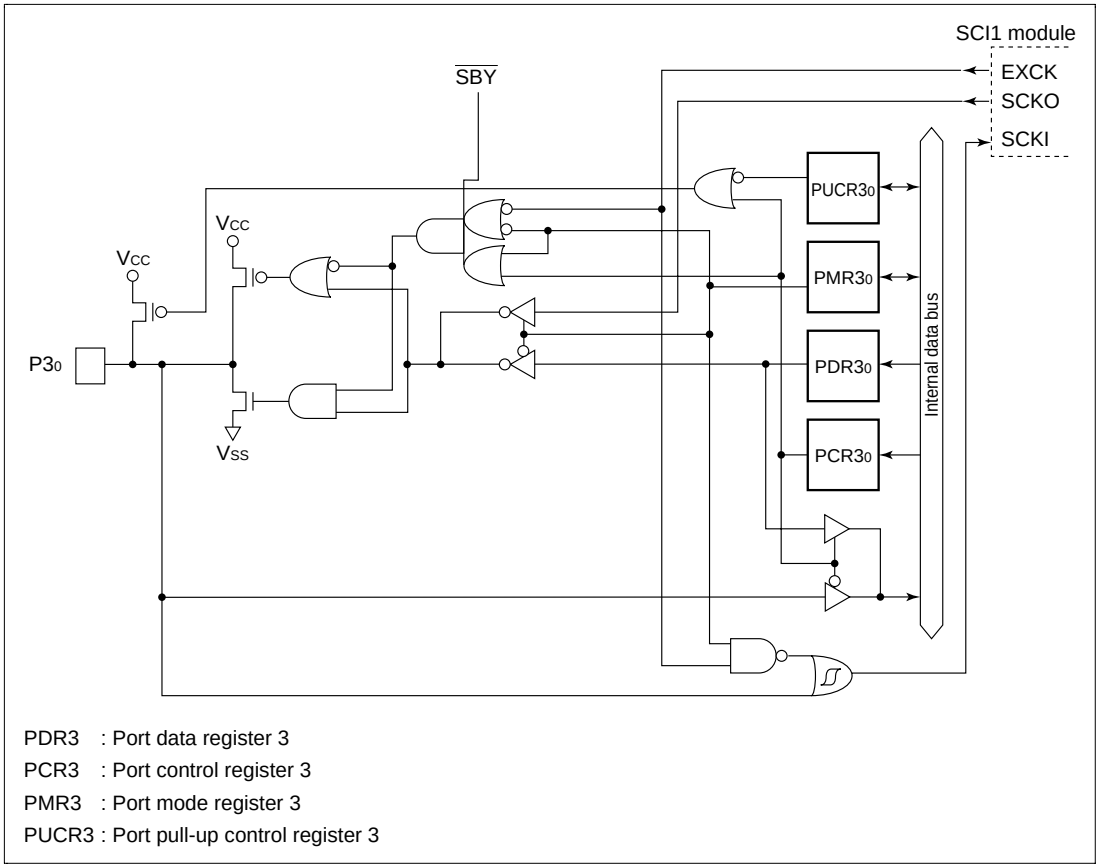


Figure C.3 (d) Port 3 Block Diagram (Pin $P3_0$: Function of H8/3857 Series Only)

C.4 Block Diagram of Port 4

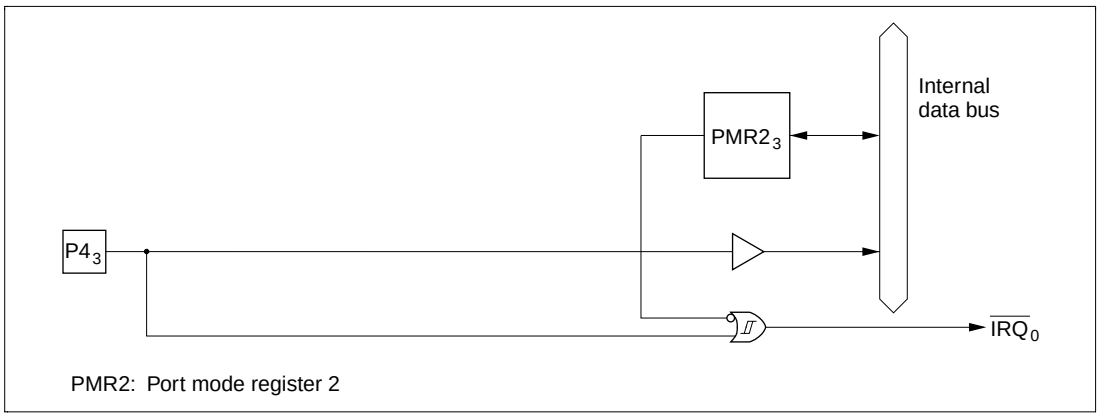


Figure C.4 (a) Port 4 Block Diagram (Pin $P4_3$)

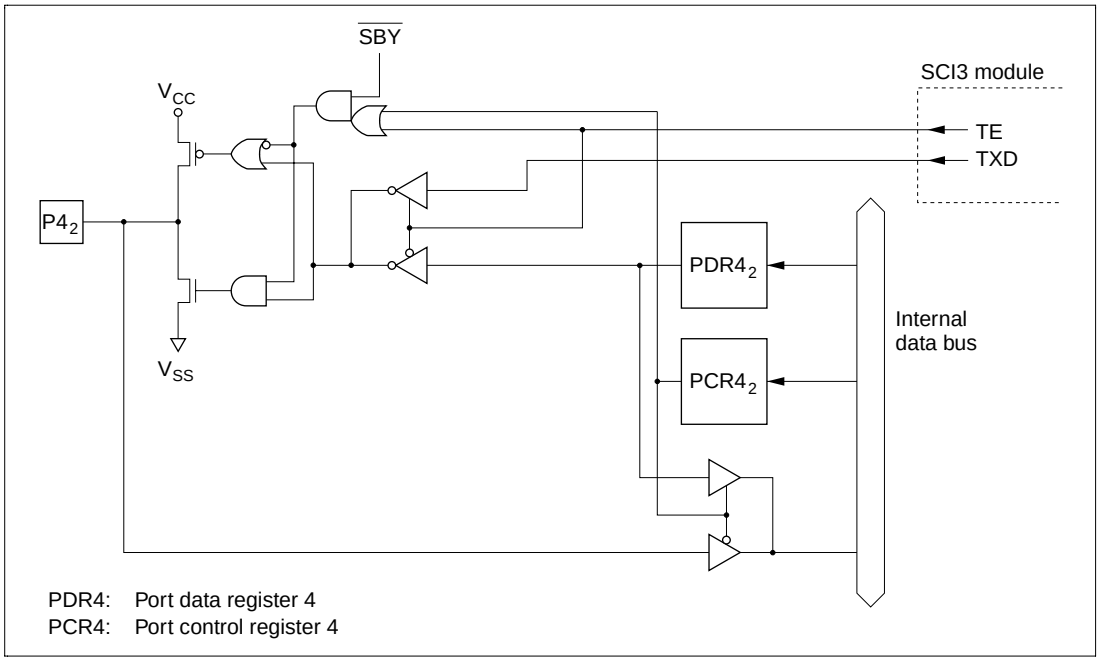


Figure C.4 (b) Port 4 Block Diagram (Pin $P4_2$)

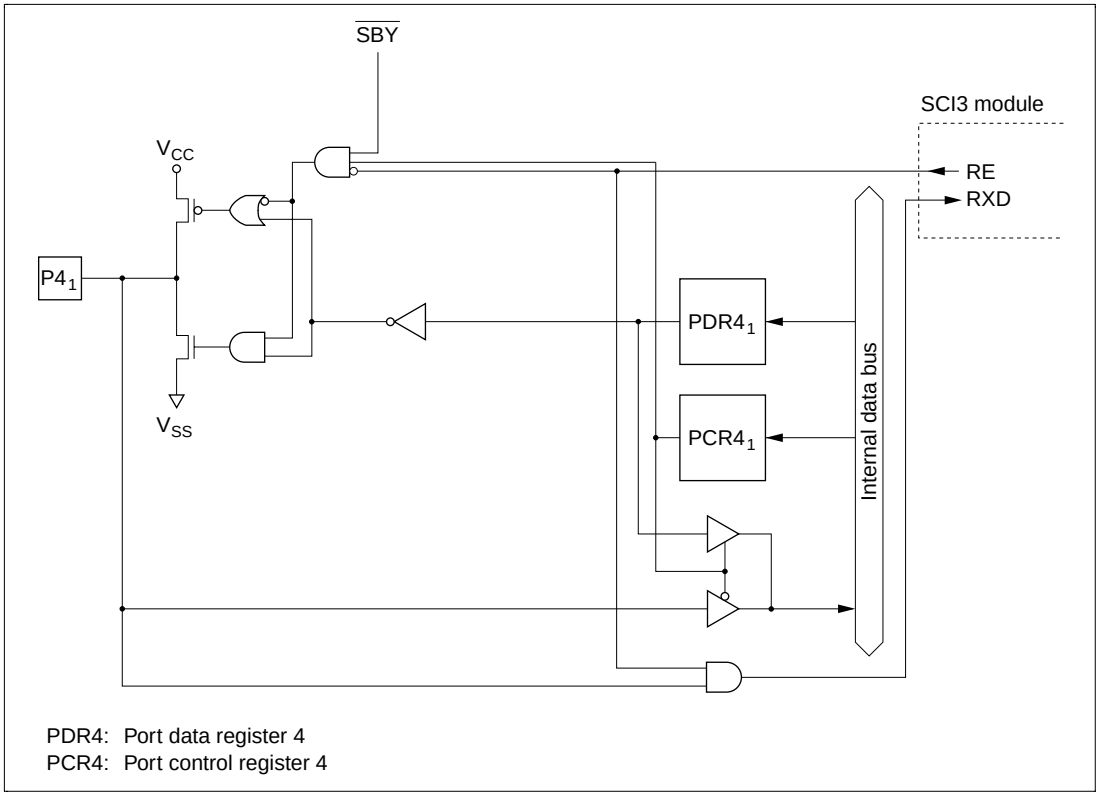


Figure C.4 (c) Port 4 Block Diagram (Pin $P4_1$)

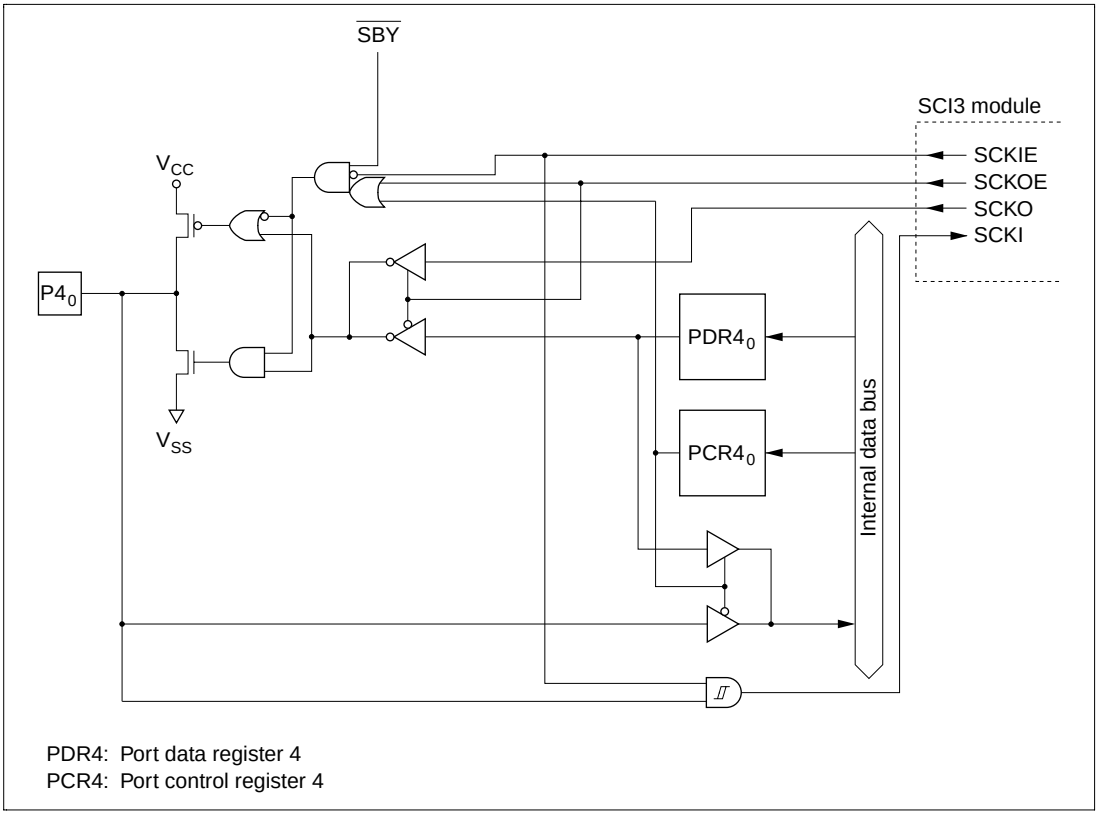


Figure C.4 (d) Port 4 Block Diagram (Pin $P4_0$)

C.5 Block Diagram of Port 5

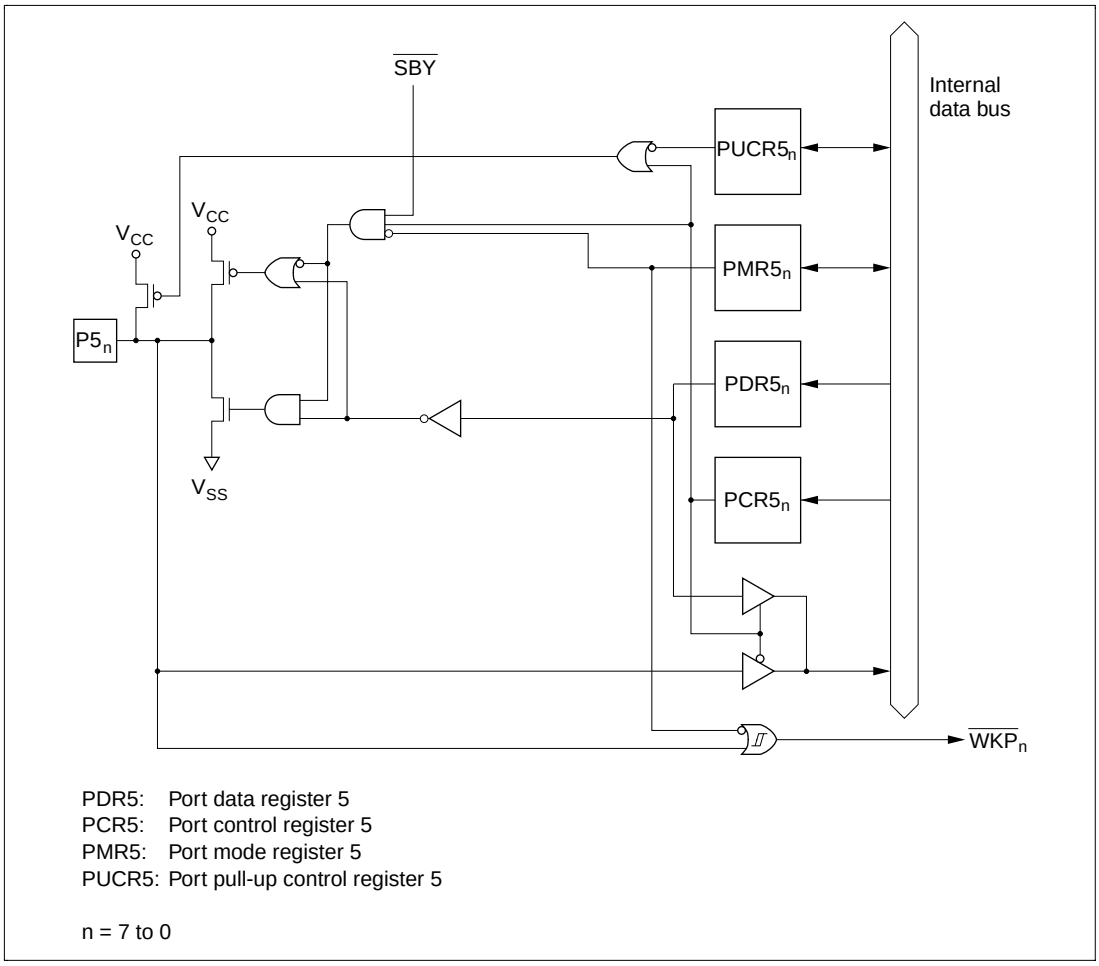


Figure C.5 Port 5 Block Diagram

C.6 Block Diagram of Port 9

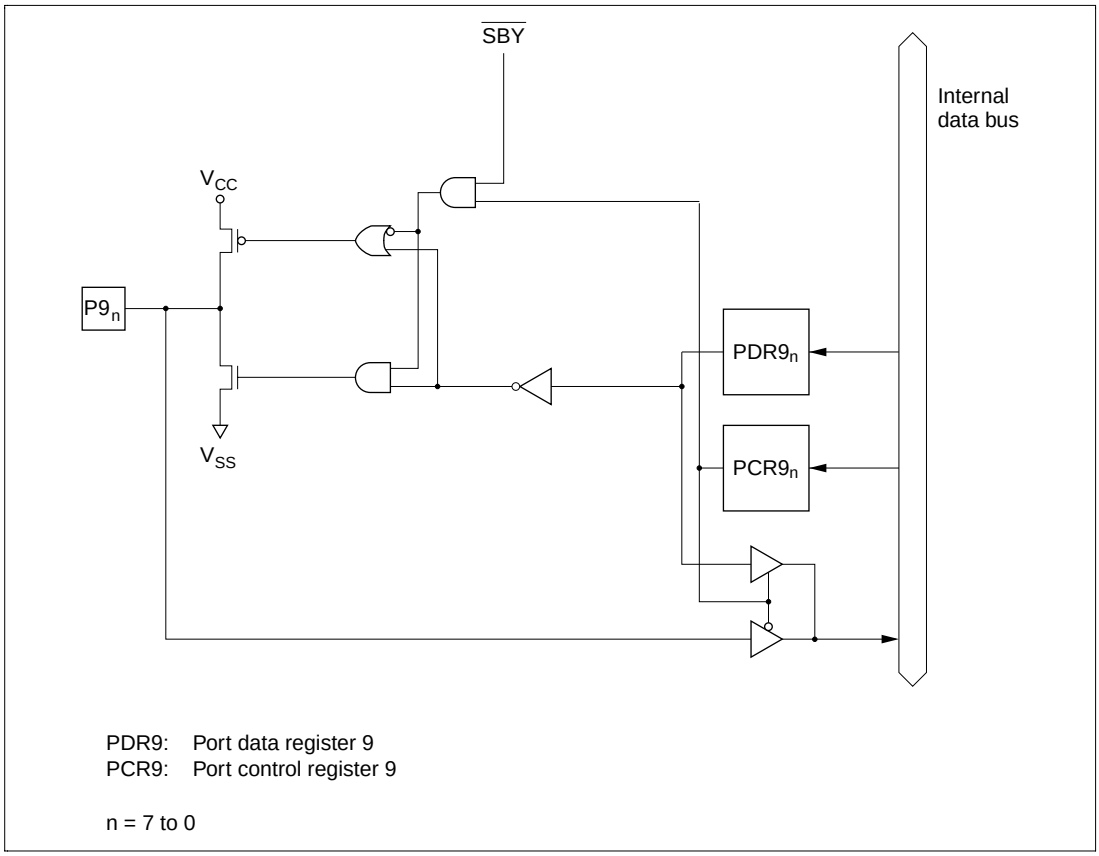


Figure C.6 Port 9 Block Diagram

C.7 Block Diagram of Port A

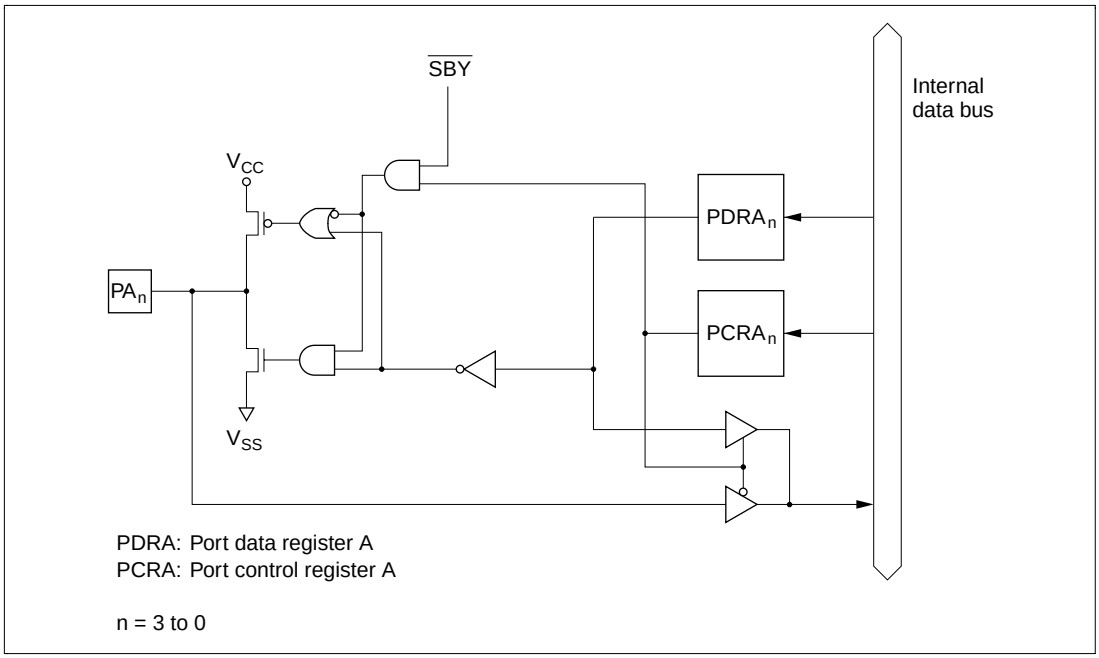


Figure C.7 Port A Block Diagram

C.8 Block Diagram of Port B

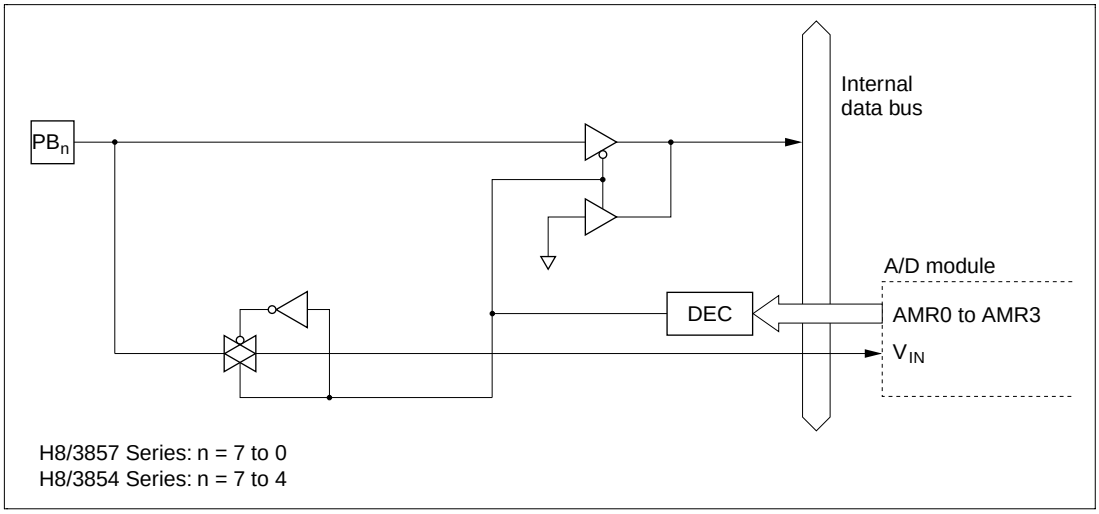


Figure C.8 Port B Block Diagram
(Pins PB_7 to PB_0: H8/3857 Series, Pins PB_7 to PB_4: H8/3854 Series)

Appendix D Port States in the Different Processing States

Table D.1 Port States Overview

Port	Reset	Sleep	Subsleep	Standby	Watch	Subactive	Active
$P1_7$ to $P1_0$*[1]	High impedance	Retained	Retained	High impedance*[2]	Retained	Functions	Functions
$P2_7$ to $P2_0$	High impedance	Retained	Retained	High impedance	Retained	Functions	Functions
$P3_7$ to $P3_0$*[1]	High impedance	Retained	Retained	High impedance*[2]	Retained	Functions	Functions
$P4_3$ to $P4_0$	High impedance	Retained	Retained	High impedance	Retained	Functions	Functions
$P5_7$ to $P5_0$	High impedance	Retained	Retained	High impedance*[2]	Retained	Functions	Functions
$P9_7$ to $P9_0$	High impedance	Retained	Retained	High impedance	Retained	Functions	Functions
PA_3 to PA_0	High impedance	Retained	Retained	High impedance	Retained	Functions	Functions
PB_7 to PB_0*[1]	High impedance	High impedance	High impedance	High impedance	High impedance	High impedance	High impedance

Notes: 1. $P1_6$, $P1_4$, $P1_3$, $P3_7$ to $P3_0$, and PB_3 to PB_0 are functions of the H8/3857 Series only, and are not provided in the H8/3854 Series.

2. High level output when MOS pull-up is in on state.

Appendix E List of Product Codes

Table E.1 H8/3857 Series Product Code Lineup

Product Type			Product Code	Mask Code	Package (Hitachi Package Code)
H8/3857F	F-ZTAT versions	Standard models	HD64F3857FQ	HD64F3857FQ	144-pin QFP (FP-144H)
			HD64F3857TG	HD64F3857TG	144-pin TQFP (TFP-144)
			HCD64F3857	—	Die
H8/3857	Mask ROM versions	Standard models	HD6433857FQ	HD6433857(***)FQ	144-pin QFP (FP-144H)
			HD6433857TG	HD6433857(***)TG	144-pin TQFP (TFP-144)
			HCD6433857	—	Die
H8/3856	Mask ROM versions	Standard models	HD6433856FQ	HD6433856(***)FQ	144-pin QFP (FP-144H)
			HD6433856TG	HD6433856(***)TG	144-pin TQFP (TFP-144)
			HCD6433856	—	Die
H8/3855	Mask ROM versions	Standard models	HD6433855FQ	HD6433855(***)FQ	144-pin QFP (FP-144H)
			HD6433855TG	HD6433855(***)TG	144-pin TQFP (TFP-144)
			HCD6433855	—	Die

Table E.2 H8/3854 Series Product Code Lineup

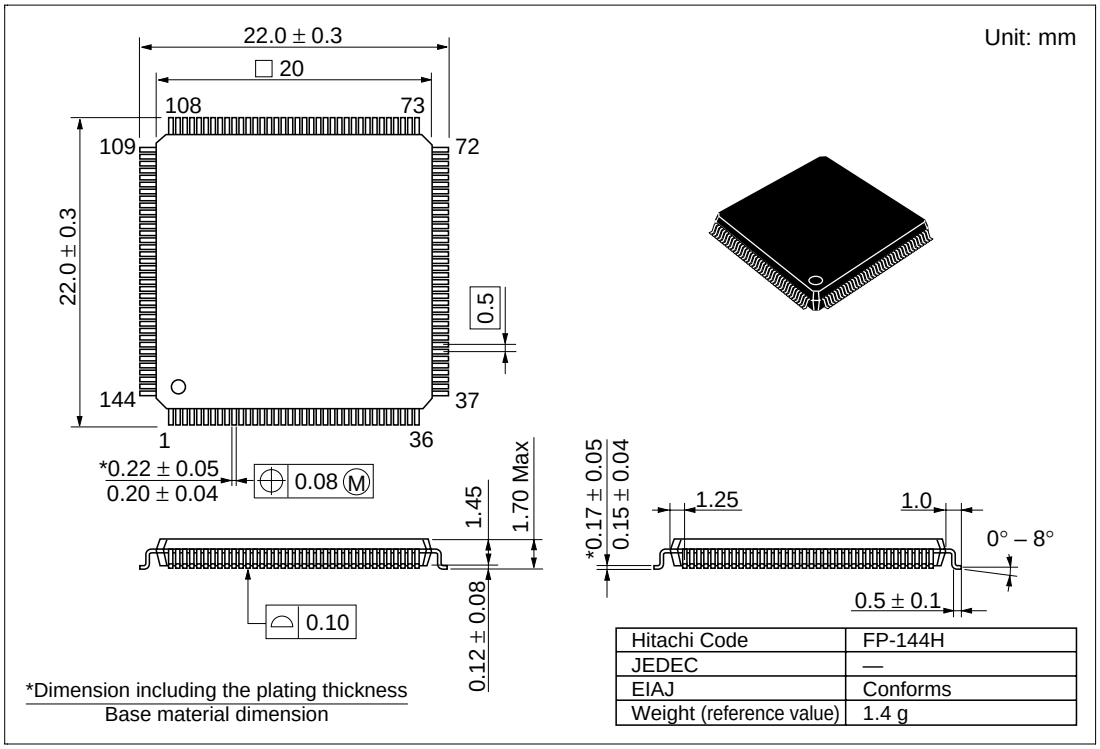
Product Type			Product Code	Mask Code	Package (Hitachi Package Code)
H8/3854F	F-ZTAT versions	Standard models	HD64F3854H	HD64F3854H	100-pin QFP (FP-100B)
			HD64F3854W	HD64F3854W	100-pin TQFP (TFP-100G)
			HCD64F3854	—	Die
H8/3854	Mask ROM versions*	Standard models	HD6433854H	HD6433854(***)H	100-pin QFP (FP-100B)
			HD6433854W	HD6433854(***)W	100-pin TQFP (TFP-100G)
			HCD6433854	—	Die
H8/3853	Mask ROM versions*	Standard models	HD6433853H	HD6433853(***)H	100-pin QFP (FP-100B)
			HD6433853W	HD6433853(***)W	100-pin TQFP (TFP-100G)
			HCD6433853	—	Die
H8/3852	Mask ROM versions*	Standard models	HD6433852H	HD6433852(***)H	100-pin QFP (FP-100B)
			HD6433852W	HD6433852(***)W	100-pin TQFP (TFP-100G)
			HCD6433852	—	Die

Note: For mask ROM versions, (***) is the ROM code.

* Under development

Appendix F Package Dimensions

Dimensional drawings of H8/3857 Series packages FP-144H and TFP-144 are shown in figures F.1 and F.2. Dimensional drawings of H8/3854 Series packages FP-100B and TFP-100G are shown in figures F.3 and F.4.



Hitachi Code	FP-144H
JEDEC	—
EIAJ	Conforms
Weight (reference value)	1.4 g

Figure F.1 FP-144H Package Dimensions

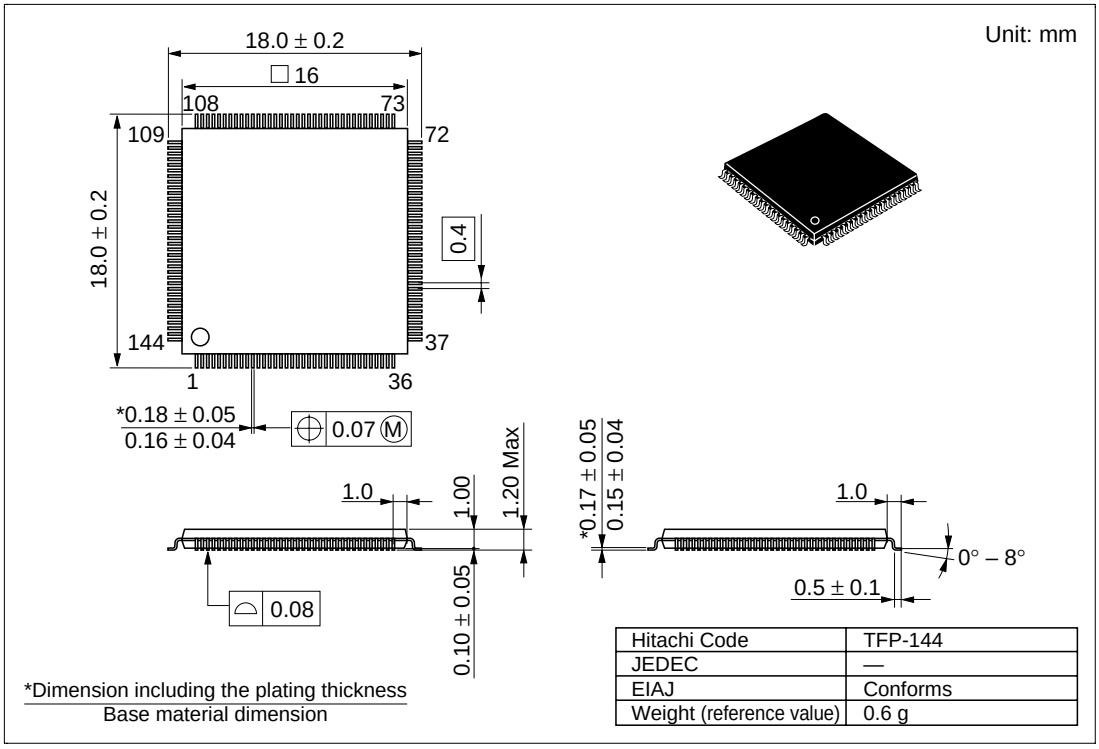


Figure F.2 TFP-144 Package Dimensions

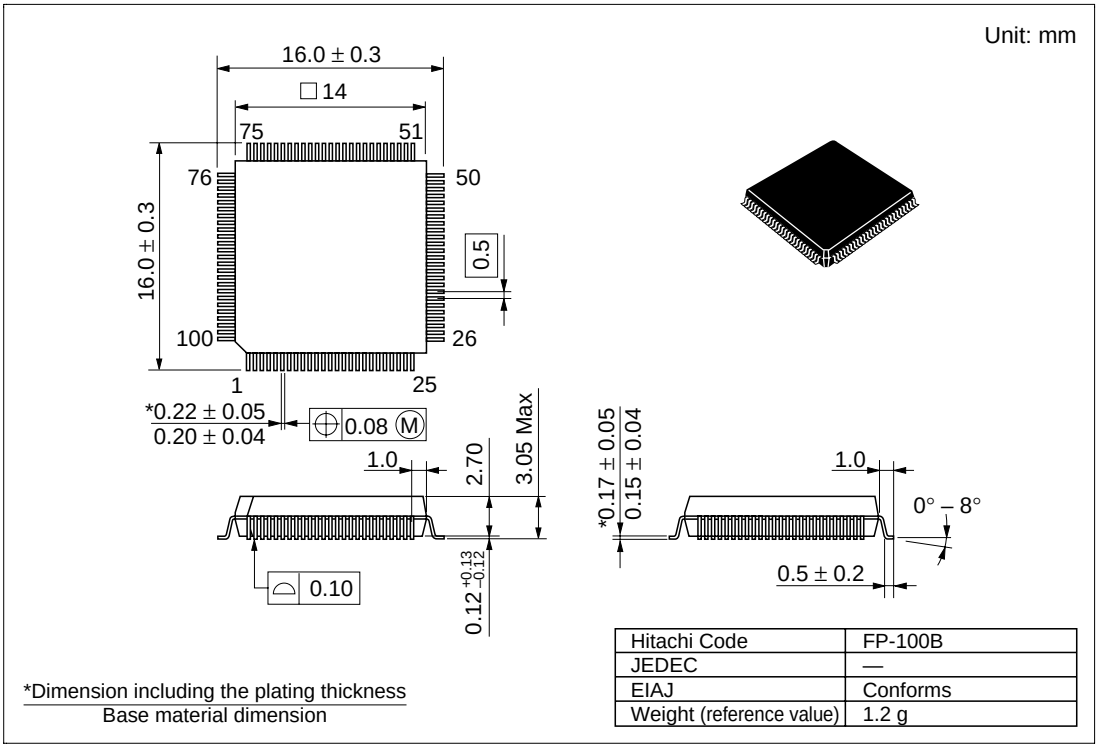
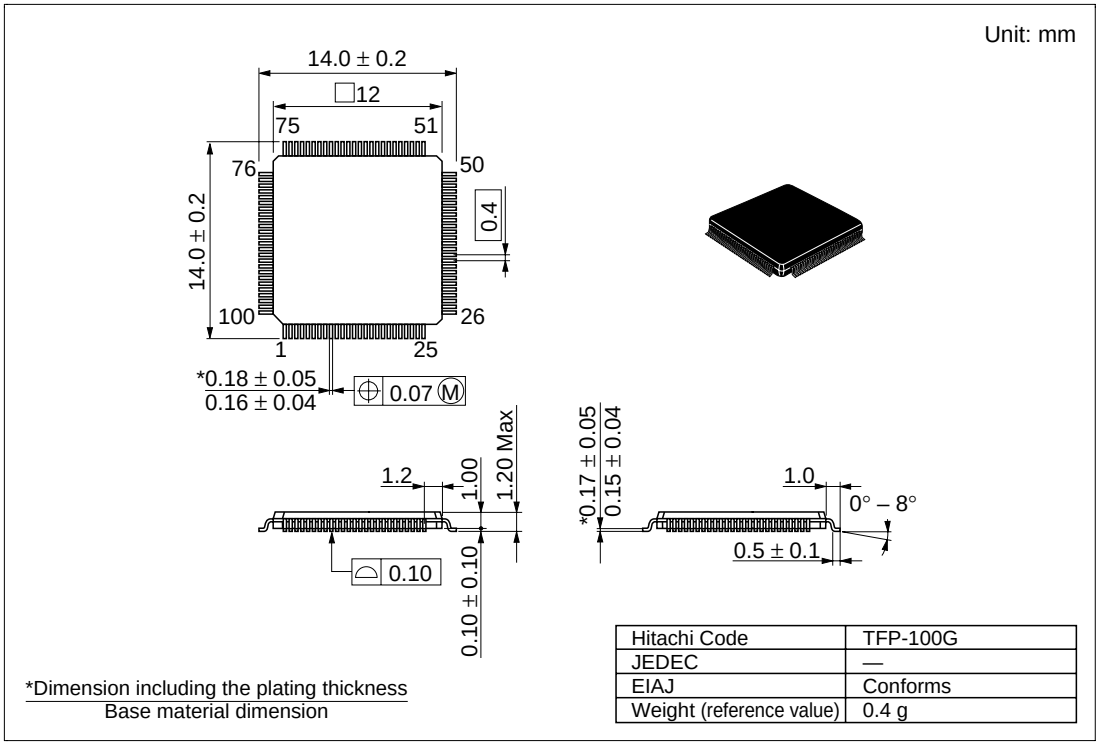


Figure F.3 FP-100B Package Dimensions



Hitachi Code	TFP-100G
JEDEC	—
EIAJ	Conforms
Weight (reference value)	0.4 g

Figure F.4 TFP-100G Package Dimensions

H8/3857 Series, H8/3857 F-ZTAT™
H8/3854 Series, H8/3854 F-ZTAT™
Hardware Manual

Publication Date: 1st Edition, March 1999
2nd Edition, February 2000
Published by: Electronic Devices Sales & Marketing Group
Semiconductor & Integrated Circuits
Hitachi, Ltd.
Edited by: Technical Documentation Group
Hitachi Kodaira Semiconductor Co., Ltd.