Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	H8/300H
Core Size	16-Bit
Speed	10MHz
Connectivity	I²C, IrDA, SCI
Peripherals	LCD, POR, PWM, WDT
Number of I/O	55
Program Memory Size	52KB (52K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-20°C ~ 75°C (TA)
Mounting Type	Surface Mount
Package / Case	80-TQFP
Supplier Device Package	80-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df38076rw10v

Table 1.2 Pad Coordinate of HCD64F38076R

		Coordinate	
Pad No.	**Pad Name**	**X (µm)**	**Y (µm)**
1	P13/TIOCB1/TCLKB	–2223	1797
2	P14/TIOCA2/TCLKC	–2223	1615
3	P15/TIOCB2	–2223	1434
4	P16/SCK4	-2223	1295
5	P30/SCK32/TMOW	–2223	1150
6	P31/RXD32/SDA	–2223	941
7	P32/TXD32/SCL	–2223	732
8	P36/SI4	–2223	523
9	P37/SO4	–2223	314
10	X1	–2223	105
11	X2	–2223	–105
12	AVss	–2223	–314
13	Vss	–2223	–418
14	OSC2	–2223	–523
15	OSC1	–2223	–732
16	TEST/$\overline{\text{ADTRG}}$	–2223	–941
17	$\overline{\text{RES}}$	–2223	–1150
18	$\overline{\text{NMI}}$	–2223	–1360
19	P40/SCK31/TMIF	–2223	–1569
20	P41/RXD31/IrRXD/TMOFL	–2223	–1778
21	P42/TXD31/IrTXD/TMOFH	–2223	–1987
22	Vcc	–1987	–2223
23	C1	–1775	–2223
24	C2	–1569	–2223
25	V1	–1360	–2223
26	V2	–1150	–2223
27	V3	–941	–2223
28	PA0/COM1	–732	–2223
29	PA1/COM2	–523	–2223

Table 2.8 System Control Instructions

Instruction	Size*	Function
RTE	—	Returns from an exception-handling routine.
SLEEP	—	Causes a transition to a power-down state.
LDC	B/W	(EAs) → CCR Moves the source operand contents to the CCR. The CCR size is one byte, but in transfer from memory, data is read by word access.
STC	B/W	CCR → (EAd) Transfers the CCR contents to a destination location. The condition code register size is one byte, but in transfer to memory, data is written by word access.
ANDC	B	CCR ∧ #IMM → CCR Logically ANDs the CCR with immediate data.
ORC	B	CCR ∨ #IMM → CCR Logically ORs the CCR with immediate data.
XORC	B	CCR ⊕ #IMM → CCR Logically XORs the CCR with immediate data.
NOP	—	PC + 2 → PC Only increments the program counter.

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

Section 4 Interrupt Controller

4.1 Features

This LSI controls interrupts by the interrupt controller. The interrupt controller has the following features.

- Mask levels settable with IPR

  An interrupt priority register (IPR) is provided for setting interrupt mask levels. Three mask levels can be set for each module for all interrupts except NMI and address break.
- Interrupts can be enabled or disabled in three levels by the INTM1 and INTM0 bits in the interrupt mask register (INTM).
- Fourteen external interrupts

  NMI is the highest-priority interrupt, and is accepted at all times. Rising or falling edge sensing can be selected for NMI. Rising or falling edge sensing can be selected for IRQ0, IRQ1, IRQ3, IRQ4, and WKP0 to WKP7. Rising, falling, or both edge sensing can be selected for IRQAEC.

A block diagram of the interrupt controller is shown in figure 4.1.

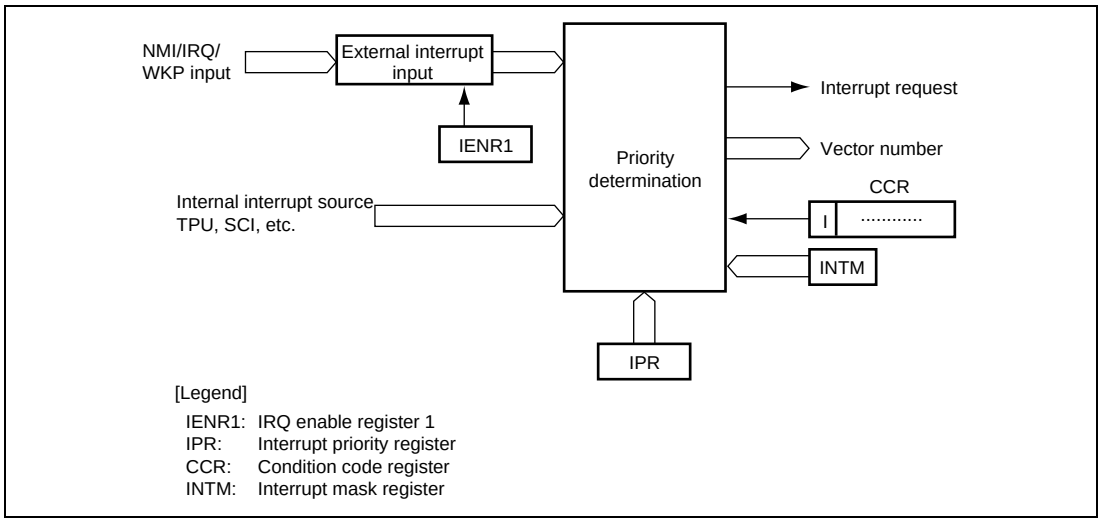


Figure 4.1 Block Diagram of Interrupt Controller

4.4.2 Internal Interrupts

Internal interrupts generated from the on-chip peripheral modules have the following features:

- For each on-chip peripheral module, there are flags that indicate the interrupt request status, and enable bits that select enabling or disabling of these interrupts. Internal interrupts can be controlled independently. If an enable bit is set to 1, an interrupt request is sent to the interrupt controller.
- The interrupt mask level can be set by IPR.

4.5 Interrupt Exception Handling Vector Table

Table 4.2 shows interrupt exception handling sources, vector addresses, and interrupt priorities. The lower the vector number, the higher the priority. The priority within a module is fixed. Mask levels for interrupts other than NMI and address break can be modified by IPR.

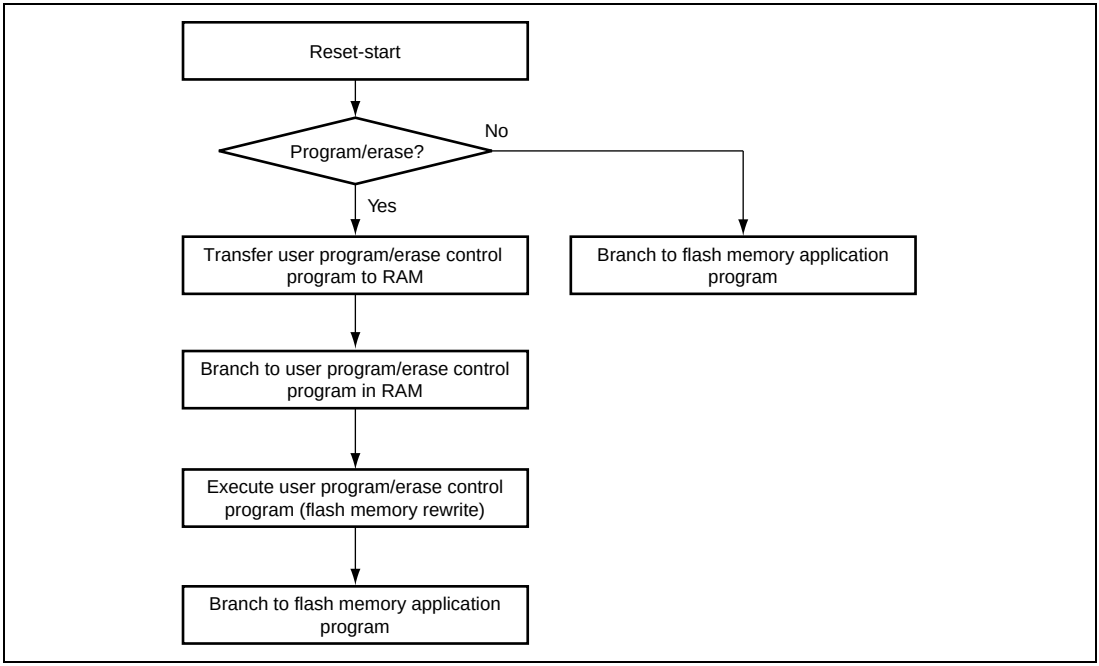


Figure 7.2 Programming/Erasing Flowchart Example in User Program Mode

9.1.1 Port Data Register 1 (PDR1)

PDR1 is a register that stores data of port 1.

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	—	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. Bit 7 is reserved. This bit is always read as 1 and cannot be modified.
6	P16	0	R/W	
5	P15	0	R/W	
4	P14	0	R/W	
3	P13	0	R/W	
2	P12	0	R/W	
1	P11	0	R/W	
0	P10	0	R/W	

9.1.2 Port Control Register 1 (PCR1)

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

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	—	Setting a PCR1 bit to 1 makes the corresponding pin (P16 to P10) an output pin, while clearing the bit to 0 makes the pin an input pin. The settings in PCR1 and in PDR1 are valid when the corresponding pin is designated as a general I/O pin. PCR1 is a write-only register. These bits are always read as 1. Bit 7 is reserved. This bit cannot be modified.
6	PCR16	0	W	
5	PCR15	0	W	
4	PCR14	0	W	
3	PCR13	0	W	
2	PCR12	0	W	
1	PCR11	0	W	
0	PCR10	0	W	

9.7.1 Port Data Register 8 (PDR8)

PDR8 is a register that stores data of port 8.

Bit	Bit Name	Initial Value	R/W	Description
7	P87	0	R/W	If port 8 is read while PCR8 bits are set to 1, the values stored in PDR8 are read, regardless of the actual pin states. If port 8 is read while PCR8 bits are cleared to 0, the pin states are read.
6	P86	0	R/W	
5	P85	0	R/W	
4	P84	0	R/W	
3	P83	0	R/W	
2	P82	0	R/W	
1	P81	0	R/W	
0	P80	0	R/W	

9.7.2 Port Control Register 8 (PCR8)

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

Bit	Bit Name	Initial Value	R/W	Description
7	PCR87	0	W	Setting a PCR8 bit to 1 makes the corresponding pin (P87 to P80) an output pin, while clearing the bit to 0 makes the pin an input pin. The settings in PCR8 and in PDR8 are valid when the corresponding pin is designated as a general I/O pin.
6	PCR86	0	W	
5	PCR85	0	W	
4	PCR84	0	W	
3	PCR83	0	W	PCR8 is a write-only register. These bits are always read as 1.
2	PCR82	0	W	
1	PCR81	0	W	
0	PCR80	0	W	

12.4 Interface to CPU

12.4.1 16-Bit Registers

TCNT and TGR are 16-bit registers. As the data bus to the CPU is 16 bits wide, these registers cannot be read or written to in 8-bit units; 16-bit access must always be used.

An example of 16-bit register access operation is shown in figure 12.2.

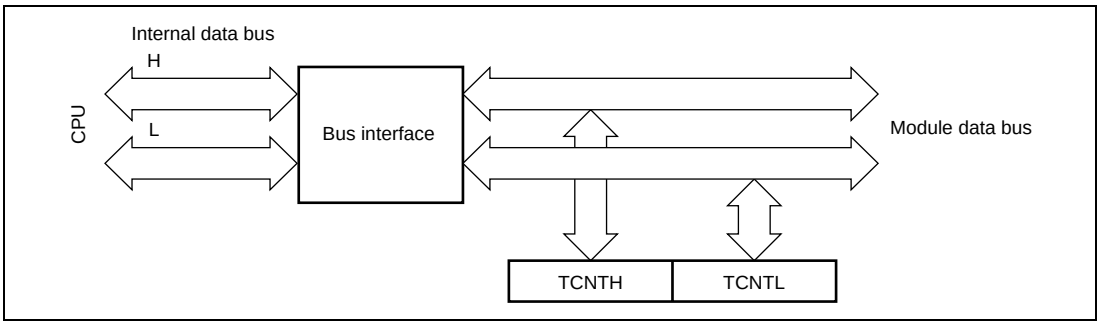


Figure 12.2 16-Bit Register Access Operation [CPU ↔ TCNT (16 Bits)]

12.4.2 8-Bit Registers

Registers other than TCNT and TGR are 8-bit. They can also be read and written to in 8-bit units.

Examples of 8-bit register access operation are shown in figures 12.3 and 12.4.

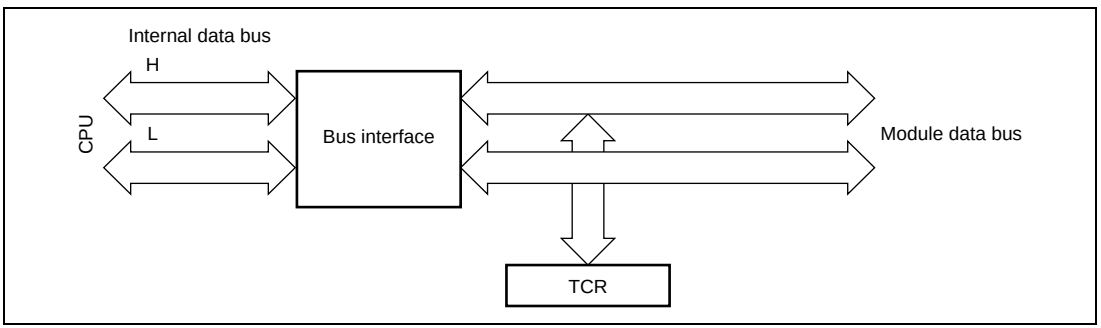


Figure 12.3 8-Bit Register Access Operation [CPU ↔ TCR (Upper 8 Bits)]

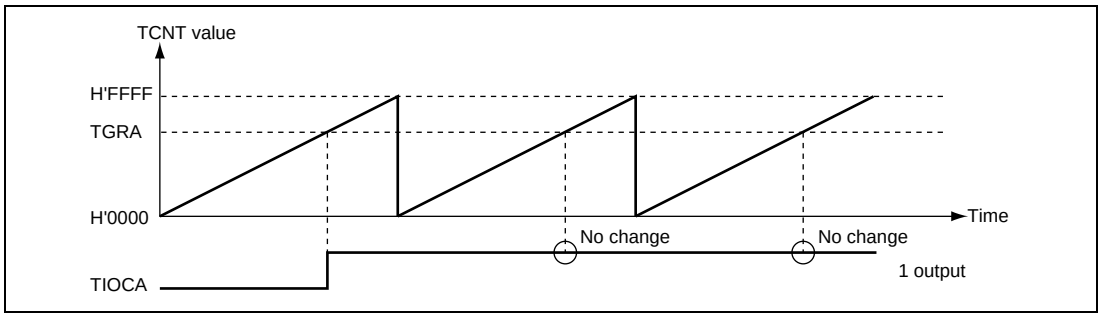


Figure 12.9 Example of 1 Output Operation

Figure 12.10 shows an example of toggle output.

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

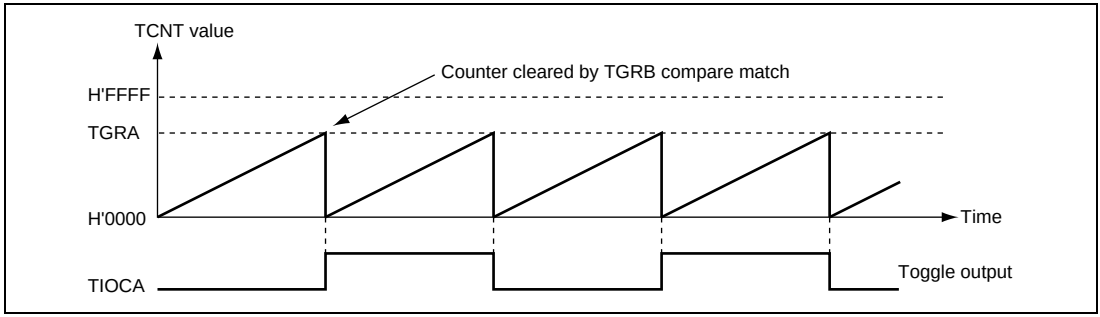


Figure 12.10 Example of Toggle Output Operation

(3) Input Capture Function

The TCNT value can be transferred to TGR on detection of the TIOC pin input edge.

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

(a) Example of Input Capture Operation Setting Procedure

Figure 12.11 shows an example of the setting procedure for input capture operation.

14.3.2 Interval Timer Mode

Figure 14.3 shows the operation in interval timer mode. To use the WDT as an interval timer, set the WT/$\overline{IT}$ bit in TCSRWD2 to 1.

When the WDT is used as an interval timer, an interval timer interrupt request is generated each time the TCNT overflows. Therefore, an interval timer interrupt can be generated at intervals.

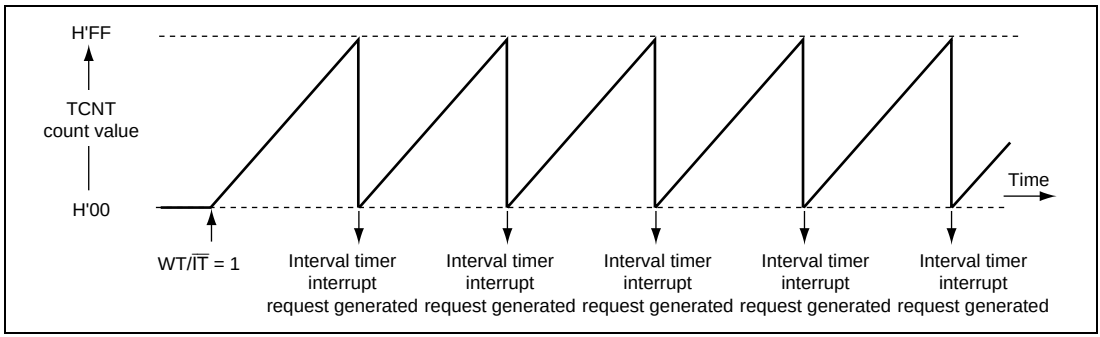


Figure 14.3 Interval Timer Mode Operation

14.3.3 Timing of Overflow Flag (OVF) Setting

Figure 14.4 shows the timing of the OVF flag setting. The OVF flag in TCSRWD2 is set to 1 if TCNT overflows. At the same time, a reset signal is output in watchdog timer mode and an interval timer interrupt is generated in interval timer mode.

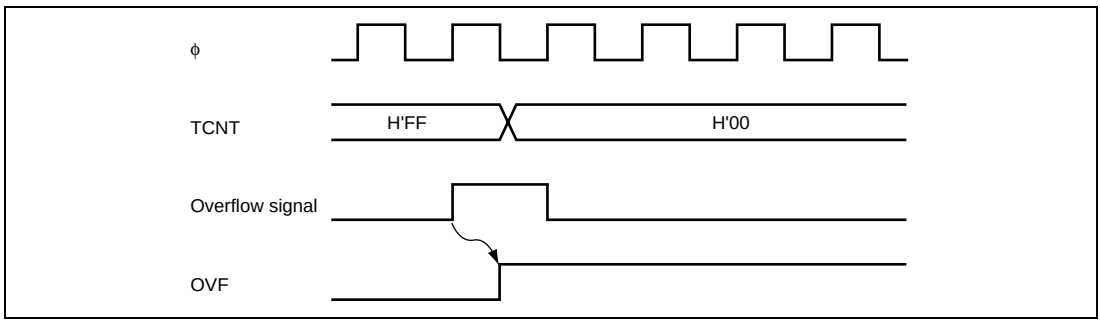
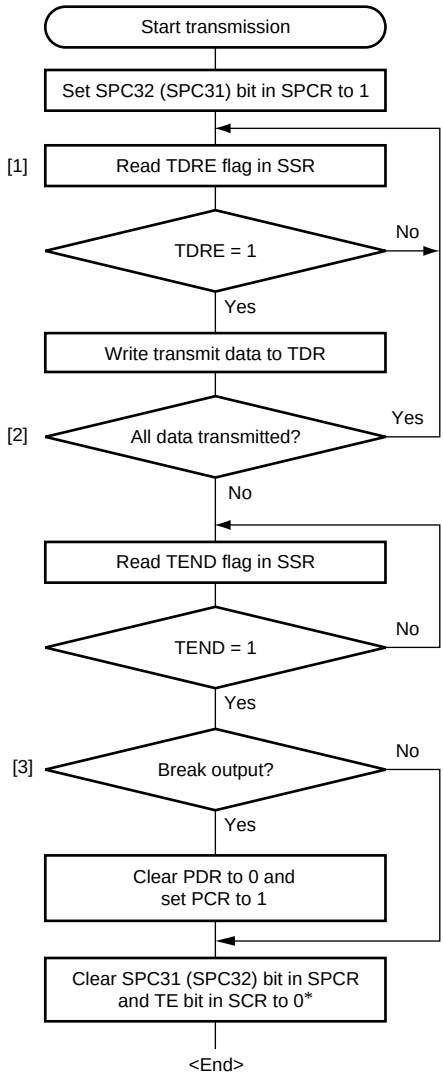


Figure 14.4 Timing of OVF Flag Setting

Table 15.3 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (3)

	4.9152MHz			5MHz			6MHz			6.144MHz		
Bit Rate (bit/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	86	0.31	2	88	–0.25	2	106	–0.44	2	108	0.08
150	3	15	0.00	2	64	0.16	2	77	0.16	3	19	0.00
200	3	11	0.00	2	48	–0.35	2	58	–0.69	3	14	0.00
250	2	37	1.05	2	38	0.16	2	46	–0.27	3	11	0.00
300	3	7	0.00	2	32	–1.36	2	38	0.16	3	9	0.00
600	3	3	0.00	0	255	1.73	3	4	–2.34	3	4	0.00
1200	3	1	0.00	0	129	0.16	0	155	0.16	2	9	0.00
2400	3	0	0.00	0	64	0.16	0	77	0.16	2	4	0.00
4800	2	1	0.00	0	32	–1.36	0	38	0.16	0	39	0.00
9600	2	0	0.00	2	0	1.73	0	19	–2.34	0	19	0.00
19200	0	7	0.00	0	7	1.73	0	9	–2.34	0	9	0.00
31250	0	4	–1.70	0	4	0.00	0	5	0.00	0	5	2.4
38400	0	3	0.00	0	3	1.73	0	4	–2.34	0	4	0.00



[1] Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR. When data is written to TDR, the TDRE flag is automaticaly cleared to 0. (After the TE bit is set to 1, one frame of 1 is output, then transmission is possible.)

[2] To continue serial transmission, read 1 from the TDRE flag to confirm that writing is possible, then write data to TDR. When data is written to TDR, the TDRE flag is automaticaly cleared to 0.

[3] To output a break in serial transmission, after setting PCR to 1 and PDR to 0, clear the SPC31 (SPC32) bit in SPCR and the TE bit in SCR to 0.

Figure 15.6 Sample Serial Transmission Flowchart (Asynchronous Mode)

Table 15.14 Transmit/Receive Interrupts

Interrupt	Flags	Interrupt Request Conditions	Notes
RXI31 (RXI32)	RDRF RIE	When serial reception is performed normally and receive data is transferred from RSR to RDR, bit RDRF is set to 1, and if bit RIE is set to 1 at this time, an RXI31 (RXI32) is enabled and an interrupt is requested. (See figure 15.17 (a).)	The RXI31 (RXI32) interrupt routine reads the receive data transferred to RDR and clears bit RDRF to 0. Continuous reception can be performed by repeating the above operations until reception of the next RSR data is completed.
TXI31 (TXI32)	TDRE TIE	When TSR is found to be empty (on completion of the previous transmission) and the transmit data placed in TDR is transferred to TSR, bit TDRE is set to 1. If bit TIE is set to 1 at this time, a TXI31 (TXI32) is enabled and an interrupt is requested. (See figure 15.17 (b).)	The TXI31 (TXI32) interrupt routine writes the next transmit data to TDR and clears bit TDRE to 0. Continuous transmission can be performed by repeating the above operations until the data transferred to TSR has been transmitted.
TEI31 (TEI32)	TEND TEIE	When the last bit of the character in TSR is transmitted, if bit TDRE is set to 1, bit TEND is set to 1. If bit TEIE is set to 1 at this time, a TEI31 (TEI32) is enabled and an interrupt is requested. (See figure 15.17 (c).)	A TEI31 (TEI32) indicates that the next transmit data has not been written to TDR when the last bit of the transmit character in TSR is transmitted.

Table 19.3 Segment Driver Selection

				Function of Pins SEG32 to SEG1							
Bit 3: SGS3	Bit 2: SGS2	Bit 1: SGS1	Bit 0: SGS0	SEG32 to SEG29	SEG28 to SEG25	SEG24 to SEG21	SEG20 to SEG17	SEG16 to SEG13	SEG12 to SEG9	SEG8 to SEG5	SEG4 to SEG1
0	0	0	0	Port	Port	Port	Port	Port	Port	Port	Port
			1	Port	Port	Port	Port	Port	Port	Port	SEG
		1	0	Port	Port	Port	Port	Port	Port	SEG	SEG
			1	Port	Port	Port	Port	Port	SEG	SEG	SEG
	1	0	0	Port	Port	Port	Port	SEG	SEG	SEG	SEG
			1	Port	Port	Port	SEG	SEG	SEG	SEG	SEG
		1	0	Port	Port	SEG	SEG	SEG	SEG	SEG	SEG
			1	Port	SEG	SEG	SEG	SEG	SEG	SEG	SEG
1	0	0	0	SEG	SEG	SEG	SEG	SEG	SEG	SEG	SEG
			1	SEG	SEG	SEG	SEG	SEG	SEG	SEG	Port
		1	0	SEG	SEG	SEG	SEG	SEG	SEG	Port	Port
			1	SEG	SEG	SEG	SEG	SEG	Port	Port	Port
	1	0	0	SEG	SEG	SEG	SEG	Port	Port	Port	Port
			1	SEG	SEG	SEG	Port	Port	Port	Port	Port
		1	0	SEG	SEG	Port	Port	Port	Port	Port	Port
			1	SEG	Port	Port	Port	Port	Port	Port	Port

19.3.2 LCD Control Register (LCR)

LCR controls LCD drive power supply and display data, and selects the frame frequency.

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	—	Reserved This bit is always read as 1 and cannot be modified.
6	PSW	0	R/W	LCD Drive Power Supply Control Can be used to turn off the LCD drive power supply when LCD display is not required in power-down mode, or when an external power supply is used. When the ACT bit is cleared to 0 or in standby mode, the LCD drive power supply is turned off regardless of the setting of this bit. 0: LCD drive power supply is turned off 1: LCD drive power supply is turned on

Bit	Bit Name	Initial Value	R/W	Description
5	MST	0	R/W	Master/Slave Select
4	TRS	0	R/W	Transmit/Receive Select In master mode with the I2C bus format, when arbitration is lost, MST and TRS are both reset by hardware, causing a transition to slave receive mode. Modification of the TRS bit should be made between transfer frames. After data receive has been started in slave receive mode, when the first seven bits of the receive data agree with the slave address that is set to SAR and the eighth bit is 1, TRS is automatically set to 1. If an overrun error occurs in master mode with the clock synchronous serial format, MST is cleared to 0 and slave receive mode is entered. Operating modes are described below according to MST and TRS combination. When clocked synchronous serial format is selected and MST is 1, clock is output. 00: Slave receive mode 01: Slave transmit mode 10: Master receive mode 11: Master transmit mode
3	CKS3	0	R/W	Transfer Clock Select 3 to 0 In master mode, set these bits according to the necessary transfer rate (see table 20.2, Transfer Rate). In slave mode, these bits are used to secure the data setup time in transmission mode. When CKS3 = 0, the data setup time is 10 tcyc and when CKS3 = 1, the data setup time is 20 tcyc.
2	CKS2	0	R/W	
1	CKS1	0	R/W	
0	CKS0	0	R/W	

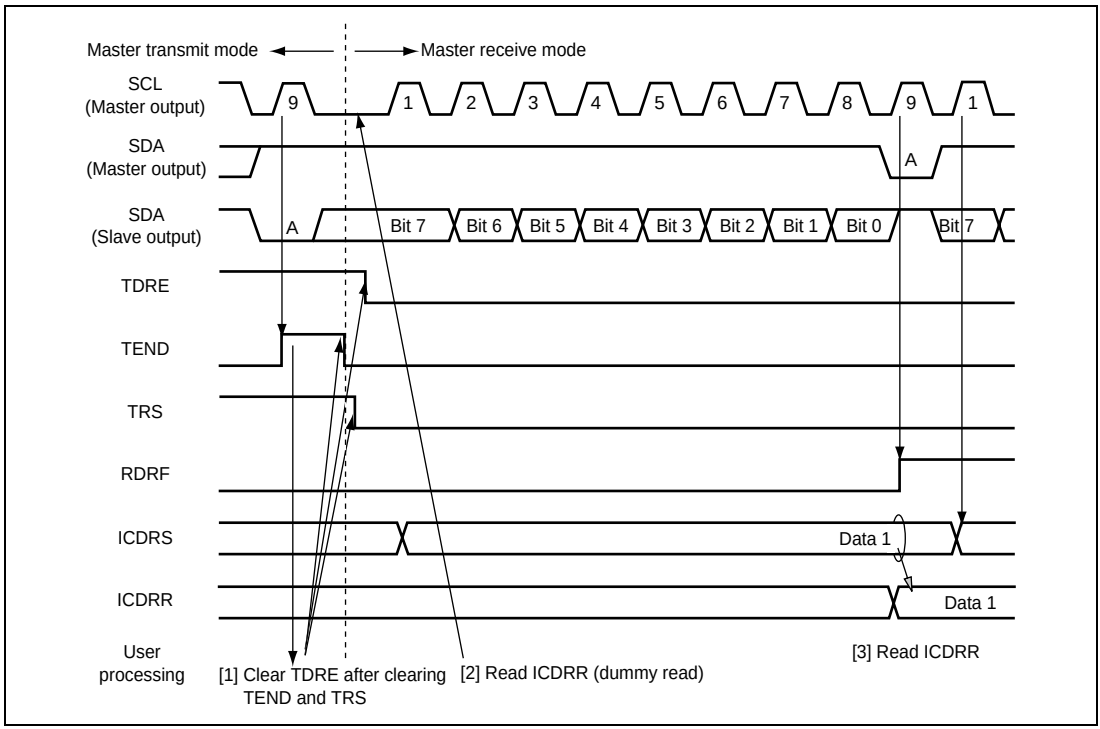


Figure 20.7 Master Receive Mode Operation Timing (1)

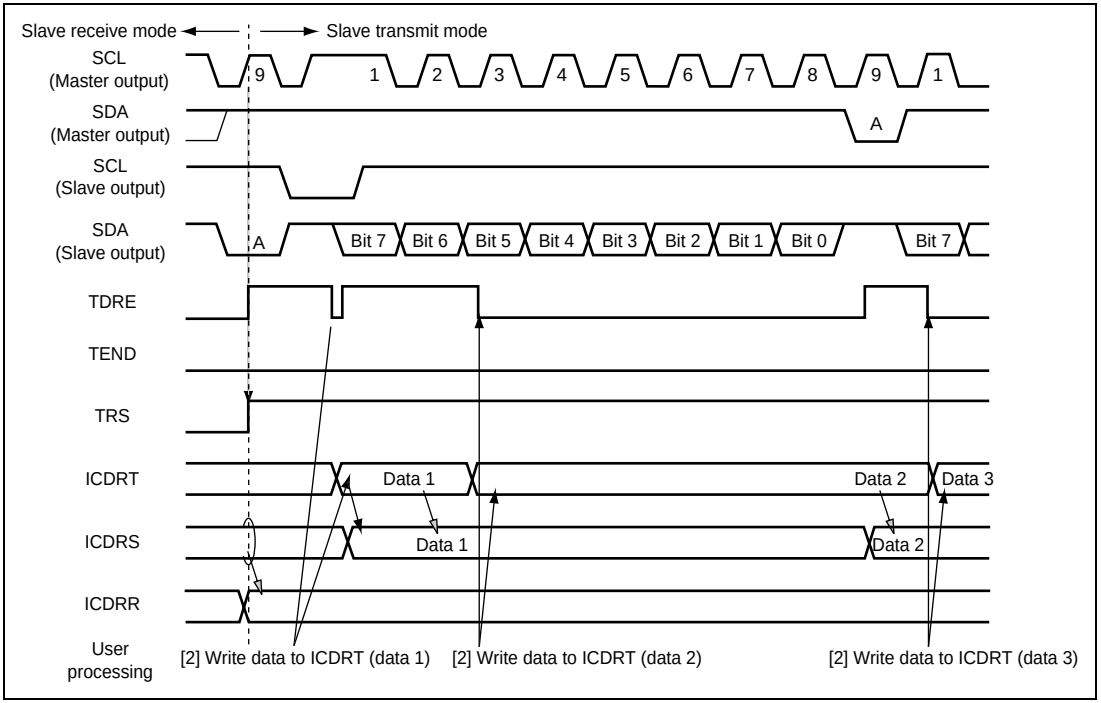


Figure 20.9 Slave Transmit Mode Operation Timing (1)

5. Bit-Manipulation Instructions

Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	|		I	H	N	Z	V	C	Normal	Advanced
BSET	BSET #xx:3, Rd	B		2								(#xx:3 of Rd8) ← 1	—	—	—	—	—	—	2	
	BSET #xx:3, @ERd	B			4							(#xx:3 of @ERd) ← 1	—	—	—	—	—	—	8	
	BSET #xx:3, @aa:8	B						4				(#xx:3 of @aa:8) ← 1	—	—	—	—	—	—	8	
	BSET Rn, Rd	B		2								(Rn8 of Rd8) ← 1	—	—	—	—	—	—	2	
	BSET Rn, @ERd	B			4							(Rn8 of @ERd) ← 1	—	—	—	—	—	—	8	
	BSET Rn, @aa:8	B						4				(Rn8 of @aa:8) ← 1	—	—	—	—	—	—	8	
BCLR	BCLR #xx:3, Rd	B		2								(#xx:3 of Rd8) ← 0	—	—	—	—	—	—	2	
	BCLR #xx:3, @ERd	B			4							(#xx:3 of @ERd) ← 0	—	—	—	—	—	—	8	
	BCLR #xx:3, @aa:8	B						4				(#xx:3 of @aa:8) ← 0	—	—	—	—	—	—	8	
	BCLR Rn, Rd	B		2								(Rn8 of Rd8) ← 0	—	—	—	—	—	—	2	
	BCLR Rn, @ERd	B			4							(Rn8 of @ERd) ← 0	—	—	—	—	—	—	8	
	BCLR Rn, @aa:8	B						4				(Rn8 of @aa:8) ← 0	—	—	—	—	—	—	8	
BNOT	BNOT #xx:3, Rd	B		2								(#xx:3 of Rd8) ← ¬ (#xx:3 of Rd8)	—	—	—	—	—	—	2	
	BNOT #xx:3, @ERd	B			4							(#xx:3 of @ERd) ← ¬ (#xx:3 of @ERd)	—	—	—	—	—	—	8	
	BNOT #xx:3, @aa:8	B						4				(#xx:3 of @aa:8) ← ¬ (#xx:3 of @aa:8)	—	—	—	—	—	—	8	
	BNOT Rn, Rd	B		2								(Rn8 of Rd8) ← ¬ (Rn8 of Rd8)	—	—	—	—	—	—	2	
	BNOT Rn, @ERd	B			4							(Rn8 of @ERd) ← ¬ (Rn8 of @ERd)	—	—	—	—	—	—	8	
	BNOT Rn, @aa:8	B						4				(Rn8 of @aa:8) ← ¬ (Rn8 of @aa:8)	—	—	—	—	—	—	8	
BTST	BTST #xx:3, Rd	B		2								¬ (#xx:3 of Rd8) → Z	—	—	—	↕	—	—	2	
	BTST #xx:3, @ERd	B			4							¬ (#xx:3 of @ERd) → Z	—	—	—	↕	—	—	6	
	BTST #xx:3, @aa:8	B						4				¬ (#xx:3 of @aa:8) → Z	—	—	—	↕	—	—	6	
	BTST Rn, Rd	B		2								¬ (Rn8 of @Rd8) → Z	—	—	—	↕	—	—	2	
	BTST Rn, @ERd	B			4							¬ (Rn8 of @ERd) → Z	—	—	—	↕	—	—	6	
	BTST Rn, @aa:8	B						4				¬ (Rn8 of @aa:8) → Z	—	—	—	↕	—	—	6	
BLD	BLD #xx:3, Rd	B		2								(#xx:3 of Rd8) → C	—	—	—	—	—	↕	2	