



Welcome to [E-XFL.COM](https://www.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	20MHz
Connectivity	I ² C, SCI
Peripherals	PWM, WDT
Number of I/O	59
Program Memory Size	96KB (96K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-BQFP
Supplier Device Package	80-QFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df36049hwv

Section 8	RAM	109
Section 9	I/O Ports.....	111
9.1	Port 1.....	111
9.1.1	Port Mode Register 1 (PMR1)	112
9.1.2	Port Control Register 1 (PCR1)	113
9.1.3	Port Data Register 1 (PDR1)	113
9.1.4	Port Pull-Up Control Register 1 (PUCR1).....	114
9.1.5	Pin Functions	114
9.2	Port 2.....	116
9.2.1	Port Control Register 2 (PCR2)	117
9.2.2	Port Data Register 2 (PDR2)	117
9.2.3	Port Mode Register 3 (PMR3).....	118
9.2.4	Pin Functions	118
9.3	Port 3.....	120
9.3.1	Port Control Register 3 (PCR3)	120
9.3.2	Port Data Register 3 (PDR3)	121
9.3.3	Pin Functions	121
9.4	Port 5.....	123
9.4.1	Port Mode Register 5 (PMR5).....	124
9.4.2	Port Control Register 5 (PCR5)	125
9.4.3	Port Data Register 5 (PDR5)	125
9.4.4	Port Pull-Up Control Register 5 (PUCR5).....	126
9.4.5	Pin Functions	126
9.5	Port 6.....	129
9.5.1	Port Control Register 6 (PCR6)	129
9.5.2	Port Data Register 6 (PDR6)	130
9.5.3	Pin Functions	130
9.6	Port 7.....	134
9.6.1	Port Control Register 7 (PCR7)	134
9.6.2	Port Data Register 7 (PDR7)	135
9.6.3	Pin Functions	135
9.7	Port 8.....	137
9.7.1	Port Control Register 8 (PCR8)	138
9.7.2	Port Data Register 8 (PDR8)	138
9.7.3	Pin Functions	139
9.8	Port 9.....	141
9.8.1	Port Control Register 9 (PCR9)	142
9.8.2	Port Data Register 9 (PDR9)	142
9.8.3	Pin Functions	143

Table 17.5	Examples of BRR Settings for Various Bit Rates (Clocked Synchronous Mode) (1).....	316
Table 17.5	Examples of BRR Settings for Various Bit Rates (Clocked Synchronous Mode) (2).....	317
Table 17.6	SSR Status Flags and Receive Data Handling	323
Table 17.7	SCI3 Interrupt Requests.....	338
Section 18 I²C Bus Interface 2 (IIC2)		
Table 18.1	Pin Configuration.....	343
Table 18.2	Transfer Rate	346
Table 18.3	Interrupt Requests	373
Table 18.4	Time for Monitoring SCL.....	374
Section 19 A/D Converter		
Table 19.1	Pin Configuration.....	379
Table 19.2	Analog Input Channels and Corresponding ADDR Registers	380
Table 19.3	A/D Conversion Time (Single Mode).....	386
Section 20 Power-On Reset and Low-Voltage Detection Circuits (Optional)		
Table 20.1	LVDCR Settings and Select Functions	394
Section 23 Electrical Characteristic		
Table 23.1	Absolute Maximum Ratings	423
Table 23.2	DC Characteristics (1).....	426
Table 23.2	DC Characteristics (2).....	432
Table 23.3	AC Characteristics	433
Table 23.4	I ² C Bus Interface Timing.....	435
Table 23.5	Serial Communication Interface (SCI) Timing.....	436
Table 23.6	A/D Converter Characteristics	437
Table 23.7	Watchdog Timer Characteristics.....	438
Table 23.8	Flash Memory Characteristics	439
Table 23.9	Power-Supply-Voltage Detection Circuit Characteristics.....	441
Table 23.10	Power-On Reset Circuit Characteristics.....	442
Table 23.11	DC Characteristics (1).....	445
Table 23.11	DC Characteristics (2).....	451
Table 23.12	AC Characteristics	452
Table 23.13	I ² C Bus Interface Timing	454
Table 23.14	Serial Communication Interface (SCI) Timing.....	455
Table 23.15	A/D Converter Characteristics	456
Table 23.16	Watchdog Timer Characteristics.....	457
Table 23.17	Power-Supply-Voltage Detection Circuit Characteristics.....	458
Table 23.18	Power-On Reset Circuit Characteristics.....	459

1.2 Internal Block Diagram

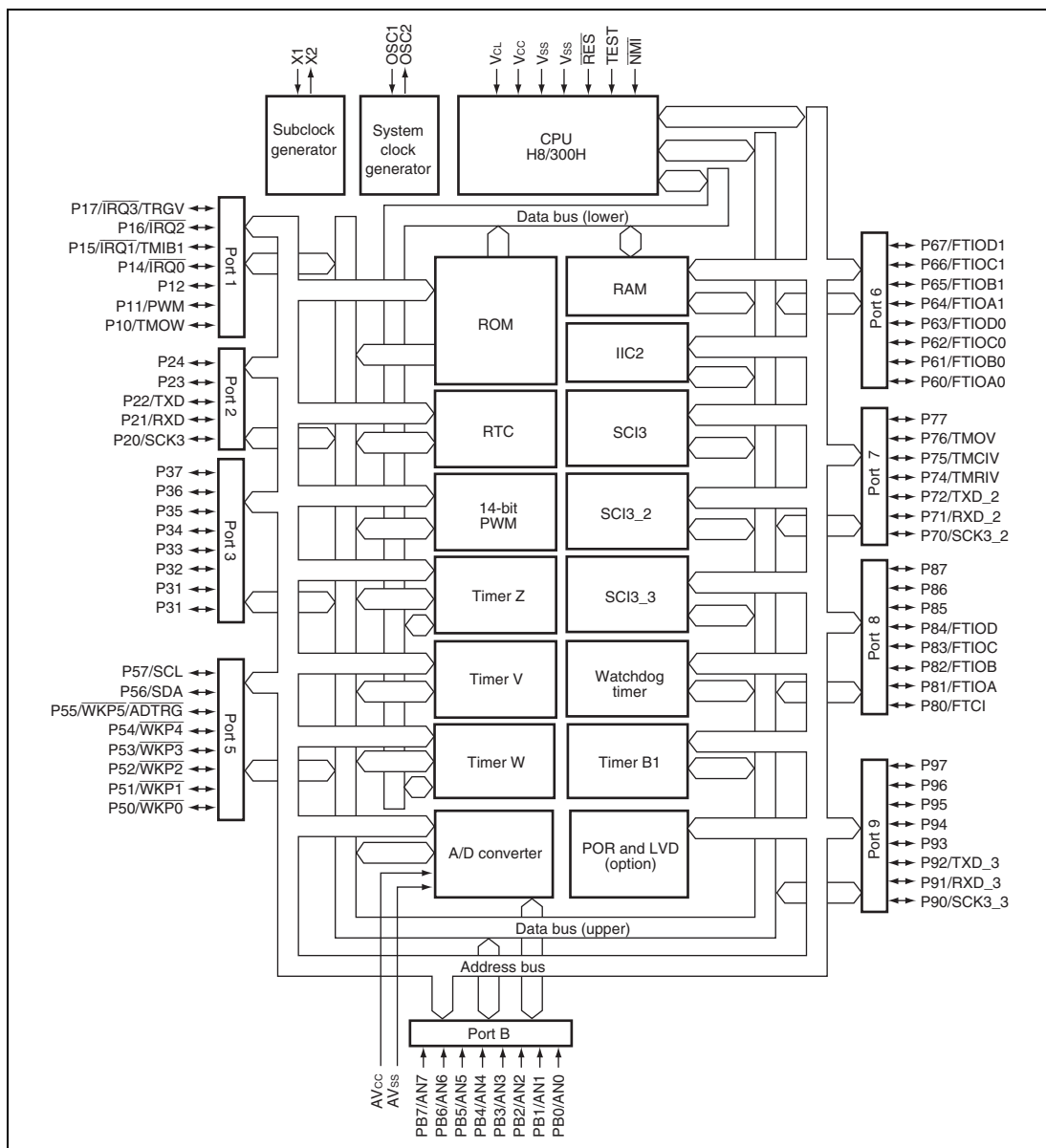


Figure 1.1 Internal Block Diagram

2.3.2 Memory Data Formats

Figure 2.6 shows the data formats in memory. The H8/300H CPU can access word data and longword data in memory, however word or longword data must begin at an even address. If an attempt is made to access word or longword data at an odd address, an address error does not occur, however the least significant bit of the address is regarded as 0, so access begins the preceding address. This also applies to instruction fetches.

When ER7 (SP) is used as an address register to access the stack area, the operand size should be word or longword.

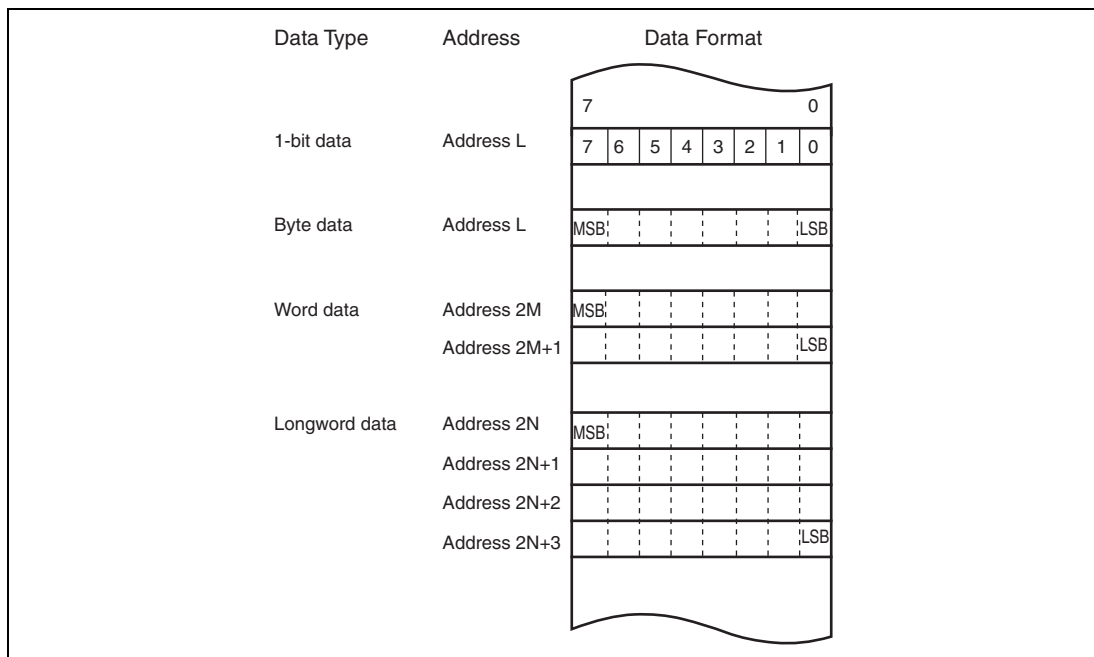


Figure 2.6 Memory Data Formats

Section 3 Exception Handling

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

- Reset

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

- Trap Instruction

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

- Interrupts

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

Bit	Bit Name	Initial Value	R/W	Description
1	SA1	0	R/W	Subactive Mode Clock Select 1 and 0
0	SA0	0	R/W	These bits select the operating clock frequency in subactive and subsleep modes. The operating clock frequency changes to the set frequency after the SLEEP instruction is executed. 00: $\phi_W/8$ 01: $\phi_W/4$ 1x: $\phi_W/2$

[Legend] x: Don't care.

6.2 Mode Transitions and States of LSI

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

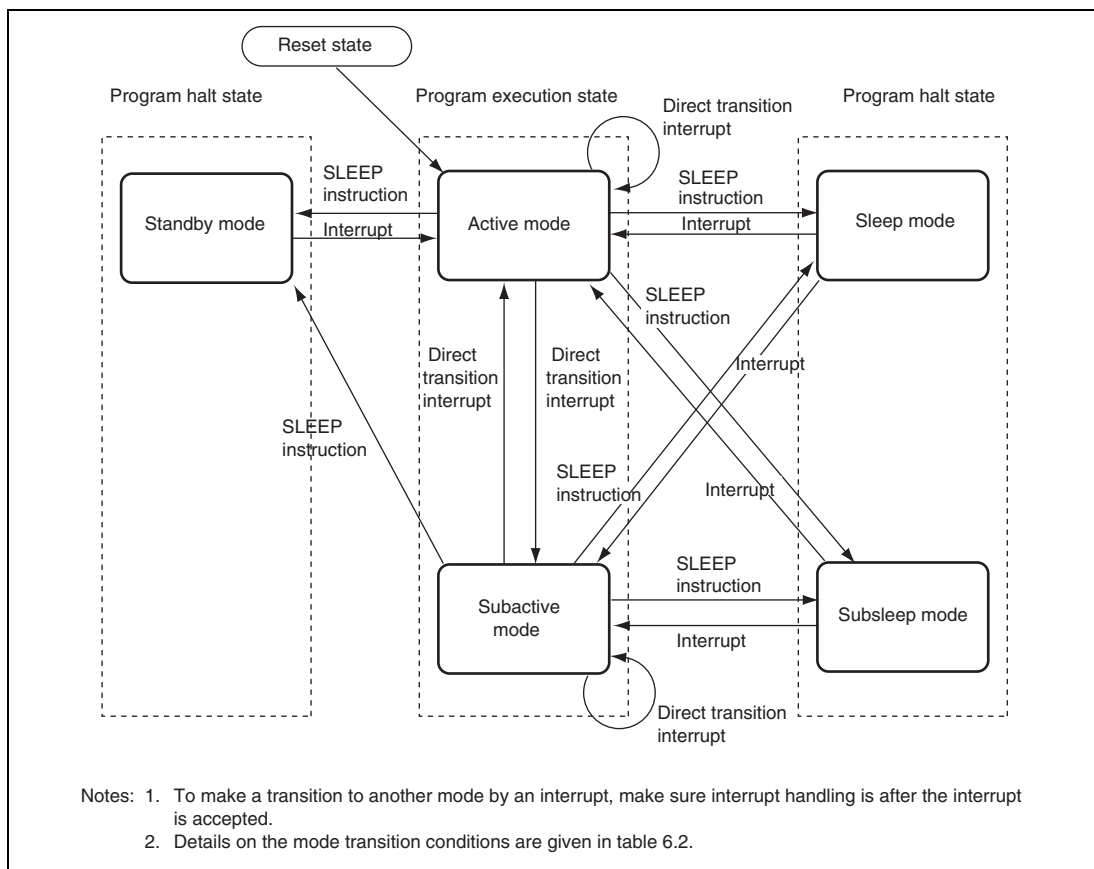


Figure 6.1 Mode Transition Diagram

7.4 Flash Memory Programming/Erasing

A software method using the CPU is employed to program and erase flash memory in the on-board programming modes. Depending on the FLMCR1 setting, the flash memory operates in one of the following four modes: Program mode, program-verify mode, erase mode, and erase-verify mode. The programming control program in boot mode and the user program/erase control program in user program mode use these operating modes in combination to perform programming/erasing. Flash memory programming and erasing should be performed in accordance with the descriptions in section 7.4.1, Program/Program-Verify and section 7.4.2, Erase/Erase-Verify, respectively.

7.4.1 Program/Program-Verify

When writing data or programs to the flash memory, the program/program-verify flowchart shown in figure 7.3 should be followed. Performing programming operations according to this flowchart will enable data or programs to be written to the flash memory without subjecting the chip to voltage stress or sacrificing program data reliability.

1. Programming must be done to an empty address. Do not reprogram an address to which programming has already been performed.
2. Programming should be carried out 128 bytes at a time. A 128-byte data transfer must be performed even if writing fewer than 128 bytes. In this case, H'FF data must be written to the extra addresses.
3. Prepare the following data storage areas in RAM: A 128-byte programming data area, a 128-byte reprogramming data area, and a 128-byte additional-programming data area. Perform reprogramming data computation according to table 7.4, and additional programming data computation according to table 7.5.
4. Consecutively transfer 128 bytes of data in byte units from the reprogramming data area or additional-programming data area to the flash memory. The program address and 128-byte data are latched in the flash memory. The lower 8 bits of the start address in the flash memory destination area must be H'00 or H'80.
5. The time during which the P bit is set to 1 is the programming time. Table 7.6 shows the allowable programming times.
6. The watchdog timer (WDT) is set to prevent overprogramming due to program runaway, etc. An overflow cycle of approximately 6.6 ms is allowed.
7. For a dummy write to a verify address, write 1-byte data H'FF to an address whose lower 2 bits are B'00. Verify data can be read in words or in longwords from the address to which a dummy write was performed.

9.5 Port 6

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

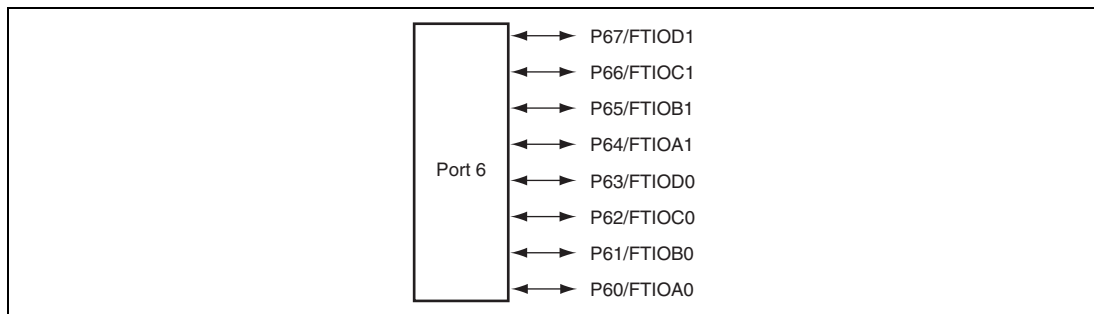


Figure 9.5 Port 6 Pin Configuration

Port 6 has the following registers.

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

9.5.1 Port Control Register 6 (PCR6)

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

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

13.5.5 Buffer Operation Timing

Figures 13.19 and 13.20 show the buffer operation timing.

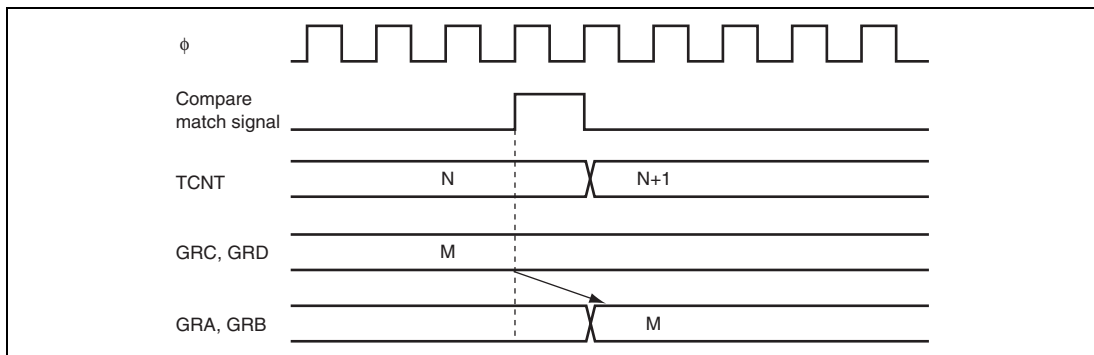


Figure 13.19 Buffer Operation Timing (Compare Match)

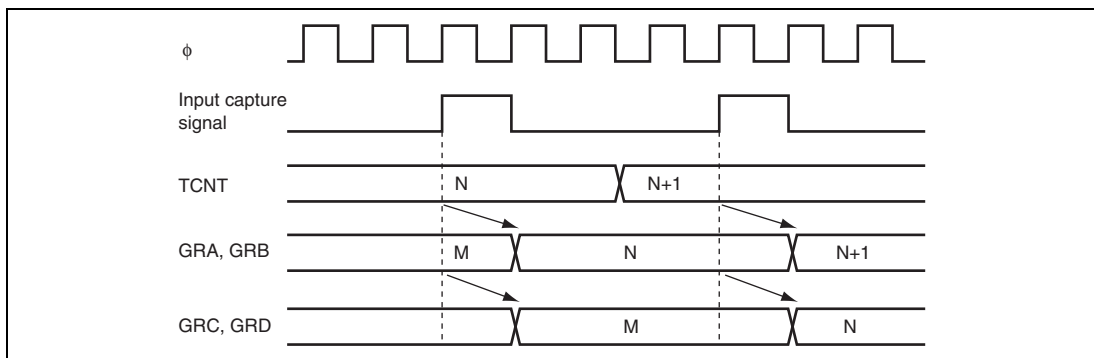


Figure 13.20 Buffer Operation Timing (Input Capture)

14.3.3 Timer PWM Mode Register (TPMR)

TPMR sets the pin to enter PWM mode.

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

B. External clock operation

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

Figure 14.11 illustrates the detection timing of the rising and falling edges.

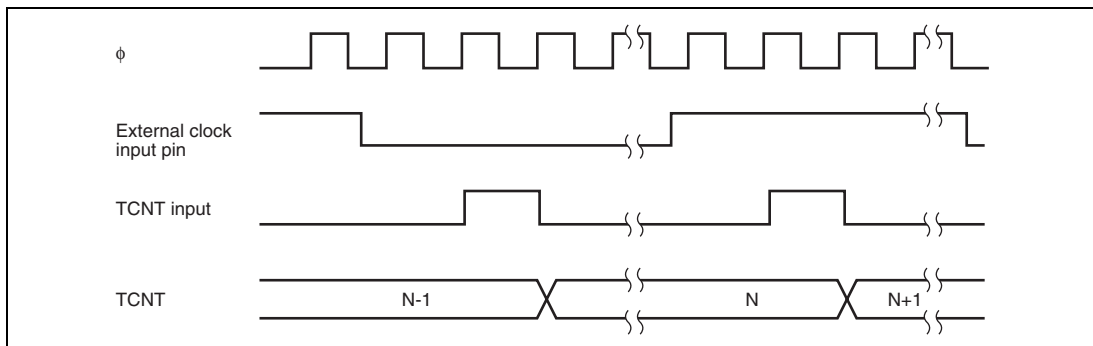


Figure 14.11 Count Timing at External Clock Operation (Both Edges Detected)

16.3.2 PWM Data Registers U, L (PWDRU, PWDRL)

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

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

PWDRU and PWDRL are initialized to H'C000.

16.4 Operation

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

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

One conversion period consists of 64 pulses, as shown in figure 16.2. The total high-level width during this period (T_H) corresponds to the data in PWDRU and PWDRL. This relation can be expressed as follows:

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

where $t\phi$ is the period of PWM clock input: $2/\phi$ (bit PWCR0 = 0) or $4/\phi$ (bit PWCR0 = 1).

If the data value in PWDRU and PWDRL is from H'FFC0 to H'FFFF, the PWM output stays high. When the data value is H'C000, T_H is calculated as follows:

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

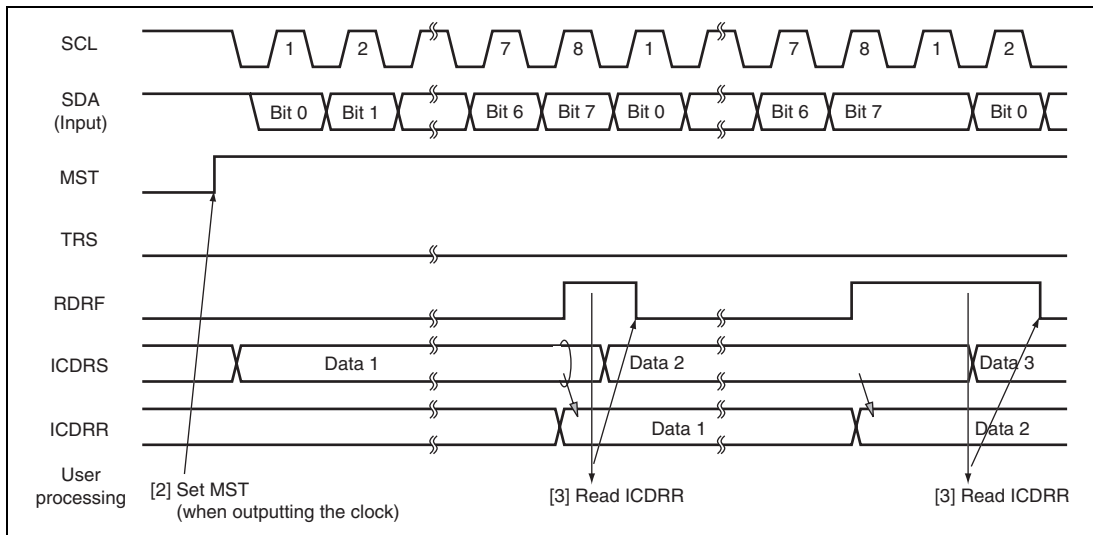


Figure 18.15 Receive Mode Operation Timing

18.4.7 Noise Canceller

The logic levels at the SCL and SDA pins are routed through noise cancellers before being latched internally. Figure 18.16 shows a block diagram of the noise canceller circuit.

The noise canceller consists of two cascaded latches and a match detector. The SCL (or SDA) input signal is sampled on the system clock, but is not passed forward to the next circuit unless the outputs of both latches agree. If they do not agree, the previous value is held.

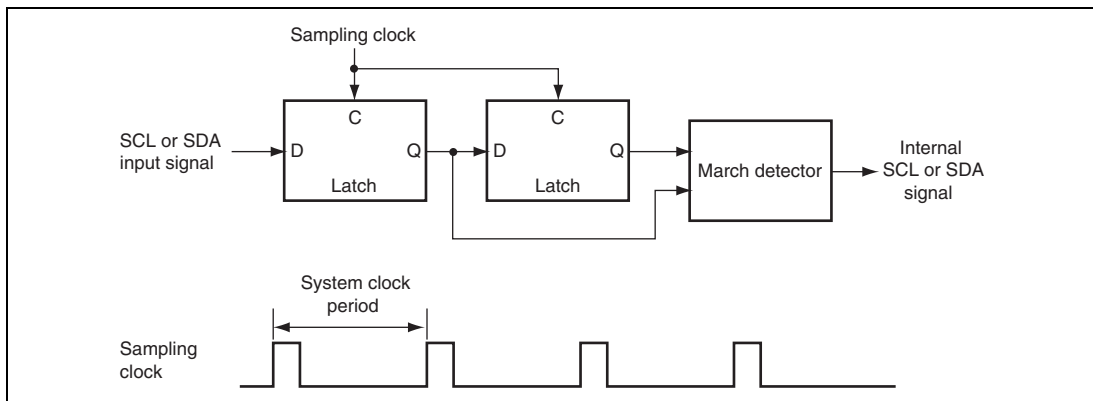


Figure 18.16 Block Diagram of Noise Canceller

19.6 Usage Notes

19.6.1 Permissible Signal Source Impedance

This LSI's analog input is designed such that conversion accuracy is guaranteed for an input signal for which the signal source impedance is $5\text{ k}\Omega$ or less. This specification is provided to enable the A/D converter's sample-and-hold circuit input capacitance to be charged within the sampling time; if the sensor output impedance exceeds $5\text{ k}\Omega$, charging may be insufficient and it may not be possible to guarantee A/D conversion accuracy. However, for A/D conversion in single mode with a large capacitance provided externally, the input load will essentially comprise only the internal input resistance of $10\text{ k}\Omega$, and the signal source impedance is ignored. However, as a low-pass filter effect is obtained in this case, it may not be possible to follow an analog signal with a large differential coefficient (e.g., $5\text{ mV}/\mu\text{s}$ or greater) (see figure 19.5). When converting a high-speed analog signal or converting in scan mode, a low-impedance buffer should be inserted.

19.6.2 Influences on Absolute Accuracy

Adding capacitance results in coupling with GND, and therefore noise in GND may adversely affect absolute accuracy. Be sure to make the connection to an electrically stable GND.

Care is also required to ensure that filter circuits do not interfere with digital signals or act as antennas on the mounting board.

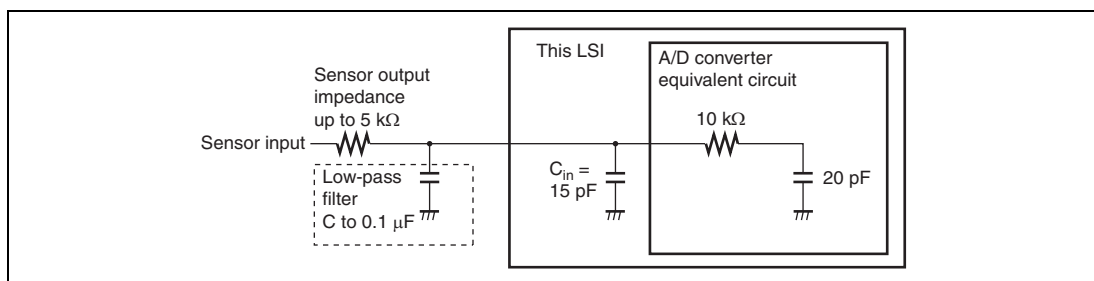


Figure 19.5 Analog Input Circuit Example

Table 23.4 I²C Bus Interface Timing

$V_{CC} = 3.0$ to 5.5 V, $V_{SS} = 0.0$ V, $T_a = -20$ to $+75^{\circ}\text{C}$, unless otherwise indicated.

Item	Symbol	Test Condition	Values			Unit	Reference Figure
			Min.	Typ.	Max.		
SCL input cycle time	t_{SCL}		$12t_{cyc} + 600$	—	—	ns	Figure 23.4
SCL input high width	t_{SCLH}		$3t_{cyc} + 300$	—	—	ns	
SCL input low width	t_{SCLL}		$5t_{cyc} + 300$	—	—	ns	
SCL and SDA input fall time	t_{Sf}		—	—	300	ns	
SCL and SDA input spike pulse removal time	t_{SP}		—	—	$1t_{cyc}$	ns	
SDA input bus-free time	t_{BUF}		$5t_{cyc}$	—	—	ns	
Start condition input hold time	t_{STAH}		$3t_{cyc}$	—	—	ns	
Retransmission start condition input setup time	t_{STAS}		$3t_{cyc}$	—	—	ns	
Setup time for stop condition input	t_{STOS}		$3t_{cyc}$	—	—	ns	
Data-input setup time	t_{SDAS}		$1t_{cyc} + 20$	—	—	ns	
Data-input hold time	t_{SDAH}		0	—	—	ns	
Capacitive load of SCL and SDA	C_b		0	—	400	pF	
SCL and SDA output fall time	t_{Sf}	$V_{CC} = 4.0$ to 5.5 V	—	—	250	ns	
			—	—	300		

Instruction	Mnemonic	Instruction	Branch	Stack	Byte Data	Word Data	Internal
		Fetch	Addr. Read	Operation	Access	Access	Operation
		I	J	K	L	M	N
INC	INC.B Rd	1					
	INC.W #1/2, Rd	1					
	INC.L #1/2, ERd	1					
JMP	JMP @ERn	2					
	JMP @aa:24	2					2
	JMP @ @aa:8	2	1				2
JSR	JSR @ERn	2		1			
	JSR @aa:24	2		1			2
	JSR @ @aa:8	2	1	1			
LDC	LDC #xx:8, CCR	1					
	LDC Rs, CCR	1					
	LDC@ERs, CCR	2				1	
	LDC@(d:16, ERs), CCR	3				1	
	LDC@(d:24,ERs), CCR	5				1	
	LDC@ERs+, CCR	2				1	2
	LDC@aa:16, CCR	3				1	
	LDC@aa:24, CCR	4				1	
MOV	MOV.B #xx:8, Rd	1					
	MOV.B Rs, Rd	1					
	MOV.B @ERs, Rd	1			1		
	MOV.B @(d:16, ERs), Rd	2			1		
	MOV.B @(d:24, ERs), Rd	4			1		
	MOV.B @ERs+, Rd	1			1		2
	MOV.B @aa:8, Rd	1			1		
	MOV.B @aa:16, Rd	2			1		
	MOV.B @aa:24, Rd	3			1		
	MOV.B Rs, @ERd	1			1		
	MOV.B Rs, @(d:16, ERd)	2			1		
	MOV.B Rs, @(d:24, ERd)	4			1		
	MOV.B Rs, @-ERd	1			1		2
	MOV.B Rs, @aa:8	1			1		

Instruction	Mnemonic	Instruction	Branch	Stack	Byte Data	Word Data	Internal
		Fetch	Addr. Read	Operation	Access	Access	Operation
		I	J	K	L	M	N
MOV	MOV.B Rs, @aa:16	2			1		
	MOV.B Rs, @aa:24	3			1		
	MOV.W #xx:16, Rd	2					
	MOV.W Rs, Rd	1					
	MOV.W @ERs, Rd	1				1	
	MOV.W @(d:16,ERs), Rd	2				1	
	MOV.W @(d:24,ERs), Rd	4				1	
	MOV.W @ERs+, Rd	1				1	2
	MOV.W @aa:16, Rd	2				1	
	MOV.W @aa:24, Rd	3				1	
	MOV.W Rs, @ERd	1				1	
	MOV.W Rs, @(d:16,ERd)	2				1	
	MOV.W Rs, @(d:24,ERd)	4				1	
MOV	MOV.W Rs, @-ERd	1				1	2
	MOV.W Rs, @aa:16	2				1	
	MOV.W Rs, @aa:24	3				1	
	MOV.L #xx:32, ERd	3					
	MOV.L ERs, ERd	1					
	MOV.L @ERs, ERd	2				2	
	MOV.L @(d:16,ERs), ERd	3				2	
	MOV.L @(d:24,ERs), ERd	5				2	
	MOV.L @ERs+, ERd	2				2	2
	MOV.L @aa:16, ERd	3				2	
	MOV.L @aa:24, ERd	4				2	
	MOV.L ERs, @ERd	2				2	
	MOV.L ERs, @(d:16,ERd)	3				2	
	MOV.L ERs, @(d:24,ERd)	5				2	
	MOV.L ERs, @-ERd	2				2	2
	MOV.L ERs, @aa:16	3				2	
	MOV.L ERs, @aa:24	4				2	
MOVFP	MOVFP @aa:16, Rd* ²	2			1		
MOVTPE	MOVTPE Rs, @aa:16* ²	2			1		

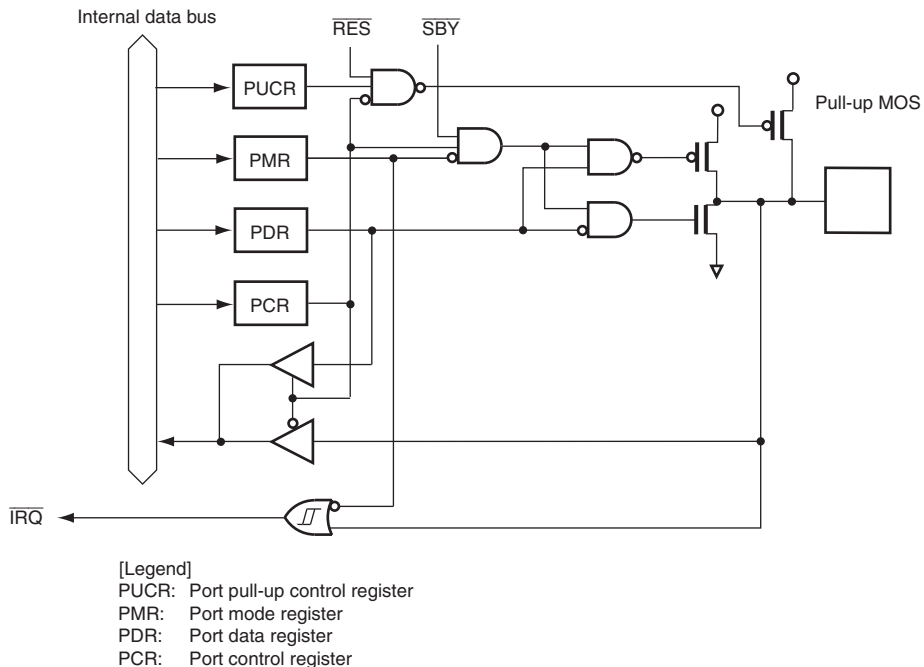


Figure B.2 Port 1 Block Diagram (P14, P16)

