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	Obsolete
Core Processor	H8/300H
Core Size	16-Bit
Speed	20MHz
Connectivity	CANbus, SCI, SSU
Peripherals	PWM, WDT
Number of I/O	45
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-20°C ~ 75°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LFQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df36034fzv

User's Manual

16

The revision list can be viewed directly by clicking the title page.

The revision list summarizes the locations of revisions and additions. Details should always be checked by referring to the relevant text.

16 H8/36057Group, H8/36037Group

Hardware Manual

Renesas 16-Bit Single-Chip Microcomputer H8 Family/H8/300H Tiny Series

H8/36057	HD64F36057, HD64F36057G
	HD64336057, HD64336057G
H8/36054	HD64F36054, HD64F36054G
	HD64336054, HD64336054G
H8/36037	HD64F36037, HD64F36037G
	HD64336037, HD64336037G
H8/36036	HD64336036, HD64336036G
H8/36035	HD64336035, HD64336035G
H8/36034	HD64F36034, HD64F36034G
	HD64336034, HD64336034G
H8/36033	HD64336033, HD64336033G
H8/36032	HD64336032, HD64336032G

Renesas Electronics
www.renesas.com

Rev.4.00 2006.05

2.5 Addressing Modes and Effective Address Calculation

The following describes the H8/300H CPU. In this LSI, the upper eight bits are ignored in the generated 24-bit address, so the effective address is 16 bits.

2.5.1 Addressing Modes

The H8/300H CPU supports the eight addressing modes listed in table 2.10. Each instruction uses a subset of these addressing modes. Addressing modes that can be used differ depending on the instruction. For details, refer to appendix A.4, Combinations of Instructions and Addressing Modes.

Arithmetic and logic instructions can use the register direct and immediate modes. Data transfer instructions can use all addressing modes except program-counter relative and memory indirect. Bit-manipulation instructions use register direct, register indirect, or the absolute addressing mode (@aa:8) to specify an operand, and register direct (BSET, BCLR, BNOT, and BTST instructions) or immediate (3-bit) addressing mode to specify a bit number in the operand.

Table 2.10 Addressing Modes

No.	Addressing Mode	Symbol
1	Register direct	Rn
2	Register indirect	@ERn
3	Register indirect with displacement	@(d:16,ERn)/@(d:24,ERn)
4	Register indirect with post-increment Register indirect with pre-decrement	@ERn+ @–ERn
5	Absolute address	@aa:8/@aa:16/@aa:24
6	Immediate	#xx:8/#xx:16/#xx:32
7	Program-counter relative	@(d:8,PC)/@(d:16,PC)
8	Memory indirect	@@aa:8

(1) Register Direct—Rn

The register field of the instruction specifies an 8-, 16-, or 32-bit general register containing the operand. R0H to R7H and R0L to R7L can be specified as 8-bit registers. R0 to R7 and E0 to E7 can be specified as 16-bit registers. ER0 to ER7 can be specified as 32-bit registers.

3.4.3 Interrupt Handling Sequence

Interrupts are controlled by an interrupt controller.

Interrupt operation is described as follows.

1. If an interrupt occurs while the NMI or interrupt enable bit is set to 1, an interrupt request signal is sent to the interrupt controller.
2. When multiple interrupt requests are generated, the interrupt controller requests to the CPU for the interrupt handling with the highest priority at that time according to table 3.1. Other interrupt requests are held pending.
3. The CPU accepts the NMI and address break without depending on the I bit value. Other interrupt requests are accepted, if the I bit is cleared to 0 in CCR; if the I bit is set to 1, the interrupt request is held pending.
4. If the CPU accepts the interrupt after processing of the current instruction is completed, interrupt exception handling will begin. First, both PC and CCR are pushed onto the stack. The state of the stack at this time is shown in figure 3.2. The PC value pushed onto the stack is the address of the first instruction to be executed upon return from interrupt handling.
5. Then, the I bit of CCR is set to 1, masking further interrupts excluding the NMI and address break. Upon return from interrupt handling, the values of I bit and other bits in CCR will be restored and returned to the values prior to the start of interrupt exception handling.
6. Next, the CPU generates the vector address corresponding to the accepted interrupt, and transfers the address to PC as a start address of the interrupt handling-routine. Then a program starts executing from the address indicated in PC.

Figure 3.3 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.

Section 9 I/O Ports

This LSI has forty-five general I/O ports and eight general input-only ports. Port 6 is a large current port, which can drive 20 mA ($@V_{OL}$ = 1.5 V) when a low level signal is output. Any of these ports can become an input port immediately after a reset. They can also be used as I/O pins of the on-chip peripheral modules or external interrupt input pins, and these functions can be switched depending on the register settings. The registers for selecting these functions can be divided into two types: those included in I/O ports and those included in each on-chip peripheral module. General I/O ports are comprised of the port control register for controlling inputs/outputs and the port data register for storing output data and can select inputs/outputs in bit units.

For functions in each port, see appendix B.1, I/O Port Block Diagrams. For the execution of bit-manipulation instructions to the port control register and port data register, see section 2.8.3, Bit Manipulation Instruction.

9.1 Port 1

Port 1 is a general I/O port also functioning as IRQ interrupt input pins, a timer B1 input pin, and a timer V input pin. Figure 9.1 shows its pin configuration.

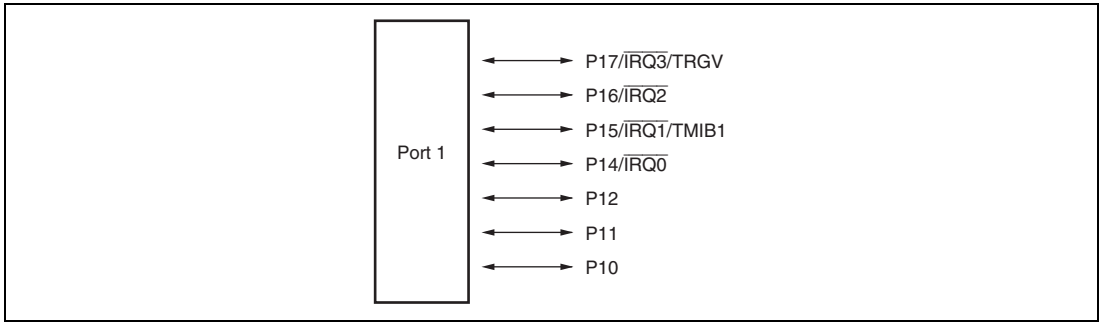


Figure 9.1 Port 1 Pin Configuration

Port 1 has the following registers.

- Port mode register 1 (PMR1)
- Port control register 1 (PCR1)
- Port data register 1 (PDR1)
- Port pull-up control register 1 (PUCR1)

9.4 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 9.4. The register setting of the timer Z has priority for functions of the pins for both uses.

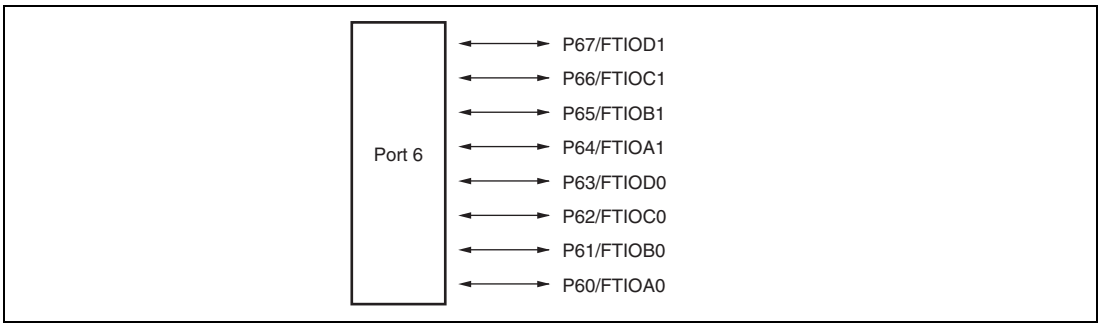


Figure 9.4 Port 6 Pin Configuration

Port 6 has the following registers.

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

9.4.1 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	

- P64/FTIOA1 pin

Register	TOER	TFCR	TIORA1	PCR6	
Bit Name	**EB1**	**CMD1, CMD0**	**IOA2 to IOA0**	**PCR64**	**Pin Function**
Setting Value	1	XX	000 or	0	P64 input/FTIOA1 input pin
			1XX	1	P64 output pin
	0	00	001 or 01X	X	FTIOA1 output pin

[Legend]
X: Don't care.

- P63/FTIOD0 pin

Register	TOER	TFCR	TPMR	TIORC0	PCR6	
Bit Name	**ED0**	**CMD1, CMD0**	**PWMD0**	**IOD2 to IOD0**	**PCR63**	**Pin Function**
Setting Value	1	00	0	000 or 1XX	0	P63 input/FTIOD0 input pin
					1	P63 output pin
	0	00	0	001 or 01X	X	FTIOD0 output pin
			1	XXX		
		Other than 00	X	XXX		

[Legend]
X: Don't care.

	Operating Frequency ϕ (MHz)											
	3.6864			4			4.9152			5		
Bit Rate (bit/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	64	0.70	2	70	0.03	2	86	0.31	2	88	–0.25
150	1	191	0.00	1	207	0.14	1	255	0.00	2	64	0.14
300	1	95	0.00	1	103	0.14	1	127	0.00	1	129	0.14
600	0	191	0.00	0	207	0.14	0	255	0.00	1	64	0.14
1200	0	95	0.00	0	103	0.14	0	127	0.00	0	129	0.14
2400	0	47	0.00	0	51	0.14	0	63	0.00	0	64	0.14
4800	0	23	0.00	0	25	0.14	0	31	0.00	0	32	–1.36
9600	0	11	0.00	0	12	0.14	0	15	0.00	0	15	1.73
19200	0	5	0.00	0	6	–6.99	0	7	0.00	0	7	1.73
31250	—	—	—	0	3	0.00	0	4	–1.70	0	4	0.00
38400	0	2	0.00	0	2	8.51	0	3	0.00	0	3	1.73

[Legend]

—: A setting is available but error occurs.

Bit	Bit Name	Initial Value	R/W	Description
0	RHIM	1	R/W	Reset/Halt Interrupt Mask Enables or disables a reset/halt interrupt request. 0: The reset/halt interrupt request is enabled 1: The reset/halt interrupt request is disabled

- TCIMR1

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 1	—	Reserved These bits are always read as 1.
4	WUPIM	1	R/W	Wakeup Interrupt Mask Enables or disables a wakeup interrupt request. 0: The wakeup interrupt request is enabled 1: The wakeup interrupt request is disabled
3, 2	—	All 1	—	Reserved These bits are always read as 1.
1	OVRIM	1	R/W	Unread Message Interrupt Mask Enables or disables an interrupt request for an unread message. 0: The interrupt request for the unread message is enabled 1: The interrupt request for the unread message is disabled
0	EMPIM	1	R/W	Mailbox Empty Interrupt Mask Enables or disables an interrupt request for mailbox empty. 0: The interrupt request for mailbox empty is enabled 1: The interrupt request for mailbox empty is disabled

Section 17 Subsystem Timer (Subtimer)

The subtimer is a timer for controlling subsystem which has an on-chip oscillator for supplying system clocks in subactive and subsleep modes and an on-chip 8-bit down counter. Since the subtimer has a prescaler that can set the division ratio by software, it can supply a clock with any frequency. This LSI has an on-chip single-channel subtimer.

17.1 Features

- On-chip oscillator
  Oscillation frequency: 64 kHz to 850 kHz
  Temperature characteristic: Source clock ± 10% (typ.)
- Counter: two
  8-bit readable/writable down counter
  8-bit counter for measuring oscillation frequency of the on-chip oscillator
- CPU interrupt source
  Underflow (interrupt interval: 731 µsec to 67.4 msec)
- Subtimer clock supply operating modes:
  Subactive mode
  Subsleep mode
- On-chip oscillator
  The on-chip oscillator supplies three kinds of clocks:
  Subactive or subsleep mode (ϕ_w)
  Subtimer down counter (input clock)
  Watchdog timer (input clock)
- Subtimer prescaler (SBTPS)
  The subtimer prescaler is a divider which controls input clocks to the counter which measures oscillation cycle of the on-chip oscillator and the down counter for the subtimer.

Figure 17.1 shows a block diagram of the subtimer. For details on the signal to the watchdog timer, refer to section 13, Watchdog Timer.

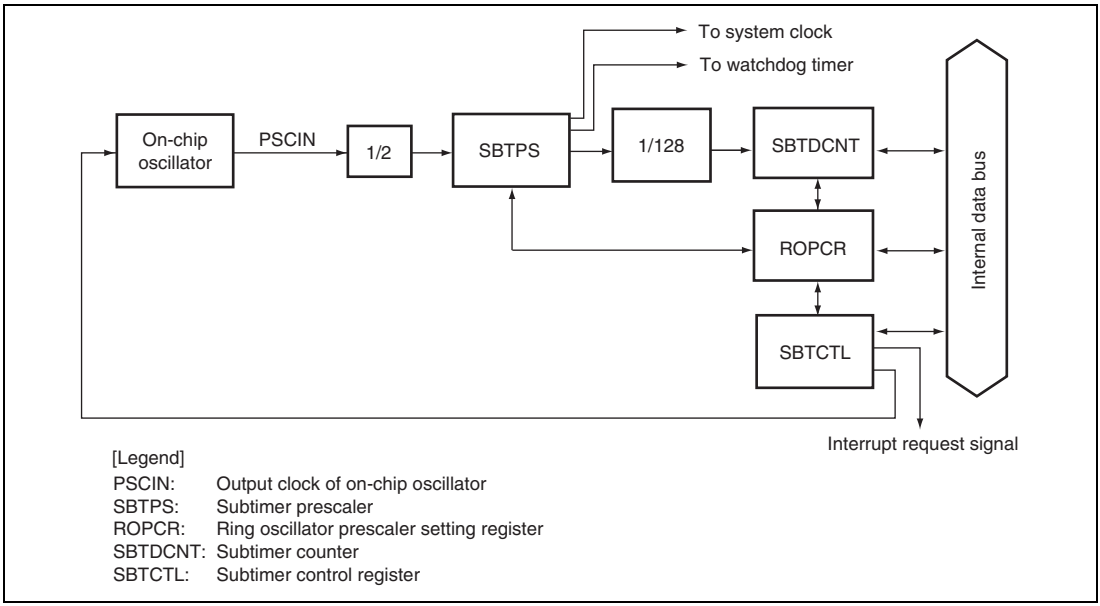


Figure 17.1 Block Diagram of Subtimer

17.4 Count Operation

The subtimer has an 8-bit readable/writable down counter, SBTDCNT. When any value ranging from H′00 to H′FF is written to SBTDCNT and the START bit in SBTCTL is set to 1, the subtimer starts counting down from the configured value in SBTDCNT. When an underflow occurs at H'00, the subtimer requests an interrupt to the CPU. At the end of the exception handling, the subtimer starts counting down again from the configured value written in SBTDCNT. If another value is written in SBTDCNT, the subtimer starts counting down from the rewritten value. Therefore, the underflow cycle can be set in the range from 1 to 256 input clocks according to the configured value in SBTDCNT. Figure 17.4 shows an example of the subtimer operation and figure 17.5 shows the flowchart.

Clocks are supplied to the entire chip by setting the SYSCKS bit in SBTCTL to 1. When the SYSCKS bit is cleared to 0, clock supply to the entire chip is disabled and only the subtimer operates.

(Example) When ϕ is 32 kHz and the underflow cycle is 100 ms:

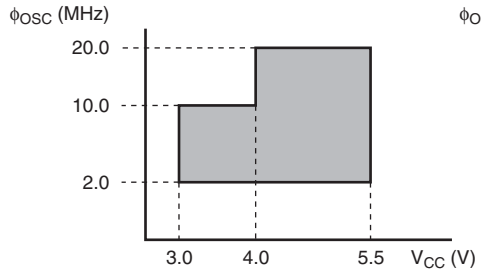
$$\frac{32 \times 10^3}{128} \times 100 \times 10^{-3} = 25$$

Therefore, set 25 (H′19) in SBTDCNT.

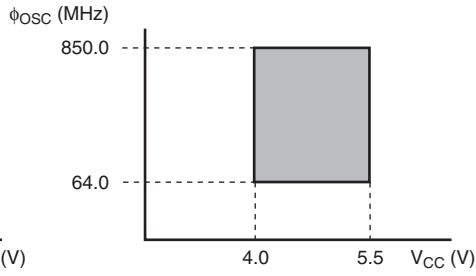
22.2 Electrical Characteristics (F-ZTAT™ Version)

22.2.1 Power Supply Voltage and Operating Ranges

Power Supply Voltage and Oscillation Frequency Range:



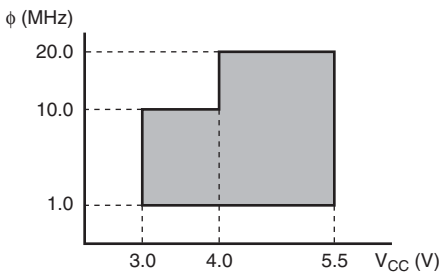
AV_{CC} = 3.3 to 5.5 V
- Active mode
- Sleep mode



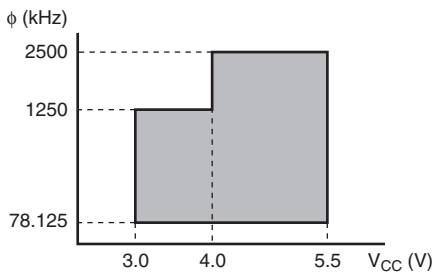
AV_{CC} = 3.3 to 5.5 V
- Active mode
- Sleep mode
- Subactive mode
- Subsleep mode

Note: This frequency range is supplied by the on-chip oscillator for the subtimer and is guaranteed.

Power Supply Voltage and Operating Frequency Range:



AV_{CC} = 3.3 to 5.5 V
- Active mode
- Sleep mode
  (When MA2 in SYSCR2 = 0)



AV_{CC} = 3.3 to 5.5 V
- Active mode
- Sleep mode
  (When MA2 in SYSCR2 = 1)

Item	Symbol	Applicable Pins	Test Condition	Values Min.	Typ.	Max.	Unit	Reference Figure
$\overline{\text{RES}}$ pin low width	t_{REL}	$\overline{\text{RES}}$	At power-on and in modes other than those below	t_{rc}	—	—	ms	Figure 22.2
			In active mode and sleep mode operation	1500	—	—	ns	
Input pin high width	t_{IH}	$\overline{\text{NMI}}$, $\overline{\text{IRQ0}}$ to $\overline{\text{IRQ3}}$, $\overline{\text{WKP0}}$ to $\overline{\text{WKP5}}$, TMCIV, TMRIV, TRGV, $\overline{\text{ADTRG}}$, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1		2	—	—	t_{cyc} t_{subcyc}	Figure 22.3
Input pin low width	t_{IL}	$\overline{\text{NMI}}$, $\overline{\text{IRQ0}}$ to $\overline{\text{IRQ3}}$, $\overline{\text{WKP0}}$ to $\overline{\text{WKP5}}$, TMCIV, TMRIV, TRGV, $\overline{\text{ADTRG}}$, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1		2	—	—	t_{cyc} t_{subcyc}	

Notes: 1. When an external clock is input, the minimum system clock oscillation frequency is 1.0 MHz.

2. Determined by the MA2, MA1, MA0, SA1, and SA0 bits in the system control register 2 (SYSCR2).

Item	Symbol	Applicable Pins	Test Condition	Values Min.	Typ.	Max.	Unit	Notes
Output low voltage	V_{OL}	P10 to P12, P14 to P17, P20 to P24, P50 to P57, P70 to P72, P74 to P76, P85 to P87, P90 to P97	V_{CC} = 4.0 to 5.5 V I_{OL} = 1.6 mA	—	—	0.6	V	
			I_{OL} = 0.4 mA	—	—	0.4		
		P60 to P67	V_{CC} = 4.0 to 5.5 V I_{OL} = 20.0 mA	—	—	1.5	V	
			V_{CC} = 4.0 to 5.5 V I_{OL} = 10.0 mA	—	—	1.0		
			V_{CC} = 4.0 to 5.5 V I_{OL} = 1.6 mA	—	—	0.4		
			I_{OL} = 0.4 mA	—	—	0.4		
Input/ output leakage current	$\lvert I_{IL} \rvert$	OSC1, $\overline{RES}$, $\overline{NMI}$, $\overline{WKP0}$ to $\overline{WKP5}$, $\overline{IRQ0}$ to $\overline{IRQ3}$, $\overline{ADTRG}$, TRGV, TMRIV, TMCIV, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1, RXD, SCK3, RXD_2*[1], SCK3_2*[1], SSCK, $\overline{SCS}$, SSI, SSO, HRXD	V_{IN} = 0.5 V or higher (V_{CC} – 0.5 V)	—	—	1.0	µA	
		P10 to P12, P14 to P17, P20 to P24, P50 to P57, P60 to P67, P70 to P72, P74 to P76, P85 to P87, P90 to P97	V_{IN} = 0.5 V or higher (V_{CC} – 0.5 V)	—	—	1.0	µA	
		PB0 to PB7	V_{IN} = 0.5 V or higher (AV_{CC} – 0.5 V)	—	—	1.0	µA	
Pull-up MOS current	$-I_p$	P10 to P12, P14 to P17, P50 to P55	V_{CC} = 5.0 V, V_{IN} = 0.0 V	50.0	—	300.0	µA	
			V_{CC} = 3.0 V, V_{IN} = 0.0 V	—	60.0	—		Reference value

Table A.1 Instruction Set

1. Data Transfer 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
MOV	MOV.B #xx:8, Rd	B	2									#xx:8 → Rd8	—	—	↕	↕	0	—	2	
	MOV.B Rs, Rd	B		2								Rs8 → Rd8	—	—	↕	↕	0	—	2	
	MOV.B @ERs, Rd	B			2							@ERs → Rd8	—	—	↕	↕	0	—	4	
	MOV.B @(d:16, ERs), Rd	B				4						@(d:16, ERs) → Rd8	—	—	↕	↕	0	—	6	
	MOV.B @(d:24, ERs), Rd	B				8						@(d:24, ERs) → Rd8	—	—	↕	↕	0	—	10	
	MOV.B @ERs+, Rd	B					2					@ERs → Rd8 ERs32+1 → ERs32	—	—	↕	↕	0	—	6	
	MOV.B @aa:8, Rd	B						2				@aa:8 → Rd8	—	—	↕	↕	0	—	4	
	MOV.B @aa:16, Rd	B						4				@aa:16 → Rd8	—	—	↕	↕	0	—	6	
	MOV.B @aa:24, Rd	B						6				@aa:24 → Rd8	—	—	↕	↕	0	—	8	
	MOV.B Rs, @ERd	B			2							Rs8 → @ERd	—	—	↕	↕	0	—	4	
	MOV.B Rs, @(d:16, ERd)	B				4						Rs8 → @(d:16, ERd)	—	—	↕	↕	0	—	6	
	MOV.B Rs, @(d:24, ERd)	B				8						Rs8 → @(d:24, ERd)	—	—	↕	↕	0	—	10	
	MOV.B Rs, @–ERd	B					2					ERd32–1 → ERd32 Rs8 → @ERd	—	—	↕	↕	0	—	6	
	MOV.B Rs, @aa:8	B						2				Rs8 → @aa:8	—	—	↕	↕	0	—	4	
	MOV.B Rs, @aa:16	B						4				Rs8 → @aa:16	—	—	↕	↕	0	—	6	
	MOV.B Rs, @aa:24	B						6				Rs8 → @aa:24	—	—	↕	↕	0	—	8	
	MOV.W #xx:16, Rd	W	4									#xx:16 → Rd16	—	—	↕	↕	0	—	4	
	MOV.W Rs, Rd	W		2								Rs16 → Rd16	—	—	↕	↕	0	—	2	
	MOV.W @ERs, Rd	W			2							@ERs → Rd16	—	—	↕	↕	0	—	4	
	MOV.W @(d:16, ERs), Rd	W				4						@(d:16, ERs) → Rd16	—	—	↕	↕	0	—	6	
	MOV.W @(d:24, ERs), Rd	W				8						@(d:24, ERs) → Rd16	—	—	↕	↕	0	—	10	
	MOV.W @ERs+, Rd	W					2					@ERs → Rd16 ERs32+2 → @ERd32	—	—	↕	↕	0	—	6	
	MOV.W @aa:16, Rd	W						4				@aa:16 → Rd16	—	—	↕	↕	0	—	6	
	MOV.W @aa:24, Rd	W						6				@aa:24 → Rd16	—	—	↕	↕	0	—	8	
	MOV.W Rs, @ERd	W			2							Rs16 → @ERd	—	—	↕	↕	0	—	4	
	MOV.W Rs, @(d:16, ERd)	W				4						Rs16 → @(d:16, ERd)	—	—	↕	↕	0	—	6	
	MOV.W Rs, @(d:24, ERd)	W				8						Rs16 → @(d:24, ERd)	—	—	↕	↕	0	—	10	

A.4 Combinations of Instructions and Addressing Modes

Table A.5 Combinations of Instructions and Addressing Modes

Functions	Instructions	Addressing Mode												
		#xx	Rn	@ERn	@(d:16,ERn)	@(d:24,ERn)	@ERn+/@ERn	@aa:8	@aa:16	@aa:24	@(d:8,PC)	@(d:16,PC)	@@aa:8	|
Data transfer instructions	MOV	BWL	BWL	BWL	BWL	BWL	BWL	B	BWL	BWL	—	—	—	—
	POP, PUSH	—	—	—	—	—	—	—	—	—	—	—	—	WL
	MOVFPE, MOVTPE	—	—	—	—	—	—	—	—	—	—	—	—	—
Arithmetic operations	ADD, CMP	BWL	BWL	—	—	—	—	—	—	—	—	—	—	—
	SUB	WL	BWL	—	—	—	—	—	—	—	—	—	—	—
	ADDX, SUBX	B	B	—	—	—	—	—	—	—	—	—	—	—
	ADDS, SUBS	—	L	—	—	—	—	—	—	—	—	—	—	—
	INC, DEC	—	BWL	—	—	—	—	—	—	—	—	—	—	—
	DAA, DAS	—	B	—	—	—	—	—	—	—	—	—	—	—
	MULXU, MULXS, DIVXU, DIVXS	—	BW	—	—	—	—	—	—	—	—	—	—	—
	NEG	—	BWL	—	—	—	—	—	—	—	—	—	—	—
	EXTU, EXTS	—	WL	—	—	—	—	—	—	—	—	—	—	—
Logical operations	AND, OR, XOR	—	BWL	—	—	—	—	—	—	—	—	—	—	—
	NOT	—	BWL	—	—	—	—	—	—	—	—	—	—	—
Shift operations		—	BWL	—	—	—	—	—	—	—	—	—	—	—
Bit manipulations		—	B	B	—	—	—	B	—	—	—	—	—	—
Branching instructions	BCC, BSR	—	—	—	—	—	—	—	—	—	—	—	—	—
	JMP, JSR	—	—	◯	—	—	—	—	—	—	◯	◯	—	—
	RTS	—	—	—	—	—	—	—	—	◯	—	—	◯	—
System control instructions	TRAPA	—	—	—	—	—	—	—	—	—	—	—	—	◯
	RTE	—	—	—	—	—	—	—	—	—	—	—	—	◯
	SLEEP	—	—	—	—	—	—	—	—	—	—	—	—	◯
	LDC	B	B	W	W	W	W	—	W	W	—	—	—	◯
	STC	—	B	W	W	W	W	—	W	W	—	—	—	—
	ANDC, ORC, XORC	B	—	—	—	—	—	—	—	—	—	—	—	—
	NOP	—	—	—	—	—	—	—	—	—	—	—	—	◯
Block data transfer instructions		—	—	—	—	—	—	—	—	—	—	—	—	BW

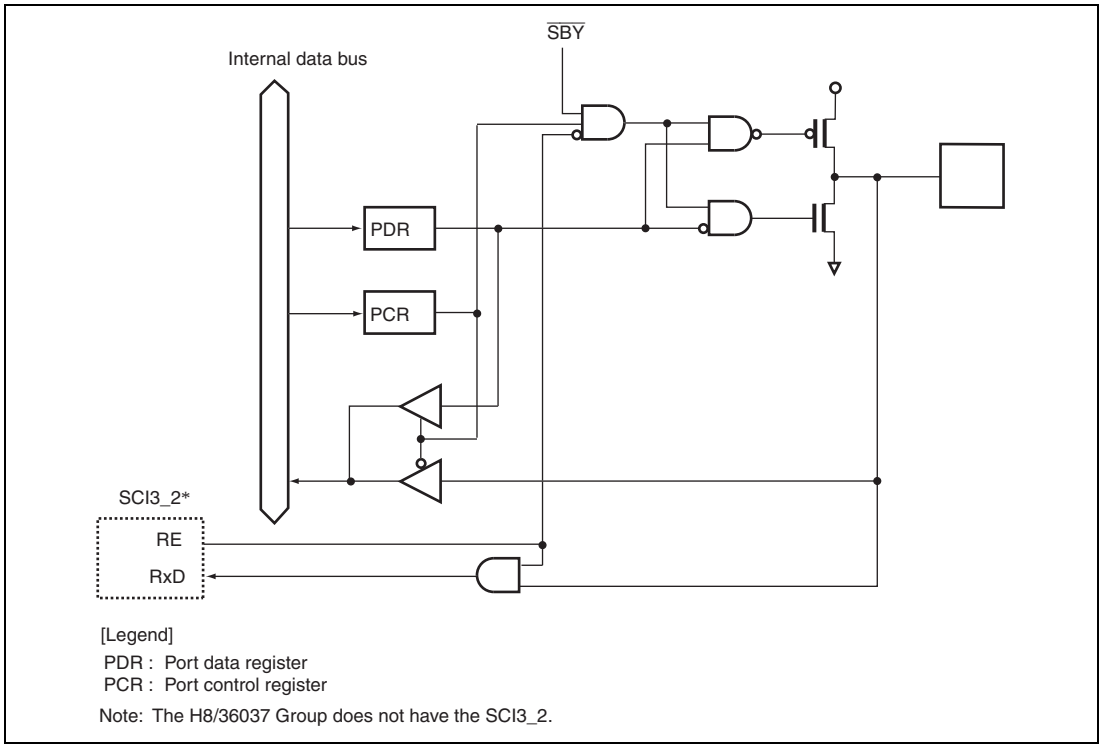


Figure B.17 Port 7 Block Diagram (P71)

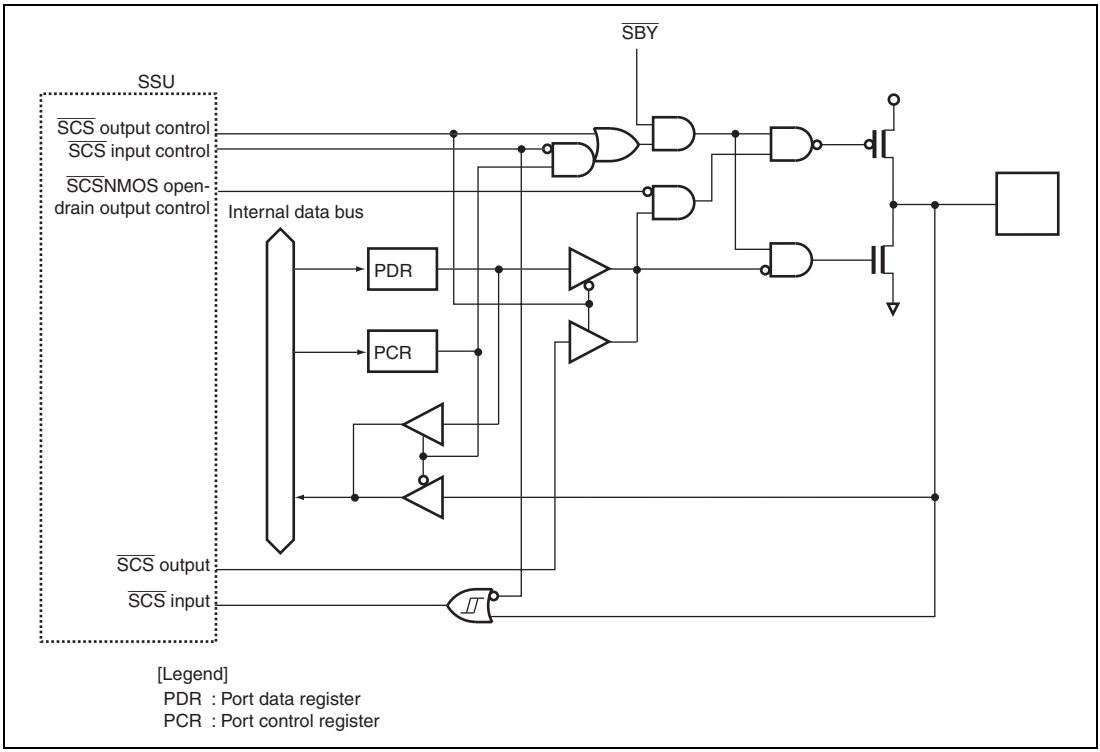


Figure B.26 Port 9 Block Diagram (P90)