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	H8S/2600
Core Size	16-Bit
Speed	28MHz
Connectivity	I²C, IrDA, SCI, SmartCard
Peripherals	DMA, POR, PWM, WDT
Number of I/O	73
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 16x10b; D/A 4x8b
Oscillator Type	Internal
Operating Temperature	-20°C ~ 75°C (TA)
Mounting Type	Surface Mount
Package / Case	128-BFQFP
Supplier Device Package	128-QFP (14x20)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df2633rf28v

H8S/2633 Group Manuals:

Document Title	Document No.
H8S/2633 Group Hardware Manual	This manual
H8S/2600 Series, H8S/2000 Series Software Manual	REJ09B0139

Users Manuals for Development Tools:

Document Title	Document No.
H8S, H8/300 Series C/C++ Compiler, Assembler, Optimized Linkage Editor User's Manual	REJ10J2039
H8S, H8/300 Series Simulator/Debugger User's Manual	REJ10B0211
High-performance Embedded Workshop User's Manual	REJ10J2169

Application Notes:

Document Title	Document No.
H8S Family Technical Q & A	REJ05B0397

Type	Symbol	I/O	Name and Function
16-bit timer-pulse unit (TPU)	TCLKD to TCLKA	Input	Clock input D to A: These pins input an external clock.
	TIOCA0, TIOCB0, TIOCC0, TIOCD0	I/O	Input capture/ output compare match A0 to D0: The TGR0A to TGR0D input capture input or output compare output, or PWM output pins.
	TIOCA1, TIOCB1	I/O	Input capture/ output compare match A1 and B1: The TGR1A and TGR1B input capture input or output compare output, or PWM output pins.
	TIOCA2, TIOCB2	I/O	Input capture/ output compare match A2 and B2: The TGR2A and TGR2B input capture input or output compare output, or PWM output pins.
	TIOCA3, TIOCB3, TIOCC3, TIOCD3	I/O	Input capture/ output compare match A3 to D3: The TGR3A to TGR3D input capture input or output compare output, or PWM output pins.
	TIOCA4, TIOCB4	I/O	Input capture/output compare match A4 and B4: The TGR4A and TGR4B input capture input or output compare output, or PWM output pins.
	TIOCA5, TIOCB5	I/O	Input capture/output compare match A5 and B5: The TGR5A and TGR5B input capture input or output compare output, or PWM output pins.
Programmable pulse generator (PPG)	PO15 to PO8	Output	Pulse output 15 to 8: Pulse output pins.
8-bit timer	TMO0 to TMO3	Output	Compare match output: The compare match output pins.
	TMCI01, TMCI23	Input	Counter external clock input: Input pins for the external clock input to the counter.
	TMRI01, TMRI23	Input	Counter external reset input: The counter reset input pins.
14-bit PWM timer (PWMX)	PWM0 to PWM3	Output	PWMX timer output: PWM D/A pulse output pins.
Watchdog timer (WDT)	$\overline{\text{WDTOVF}}$	Output	Watchdog timer overflows: The counter overflows signal output pin in watchdog timer mode.
	BUZZ	Output	BUZZ output: Output pins for the pulse divided by the watchdog timer.

2.3 Address Space

Figure 2.6 shows a memory map of the H8S/2600 CPU. The H8S/2600 CPU provides linear access to a maximum 64-kbyte address space in normal mode, and a maximum 16-Mbyte (architecturally 4-Gbyte) address space in advanced mode.

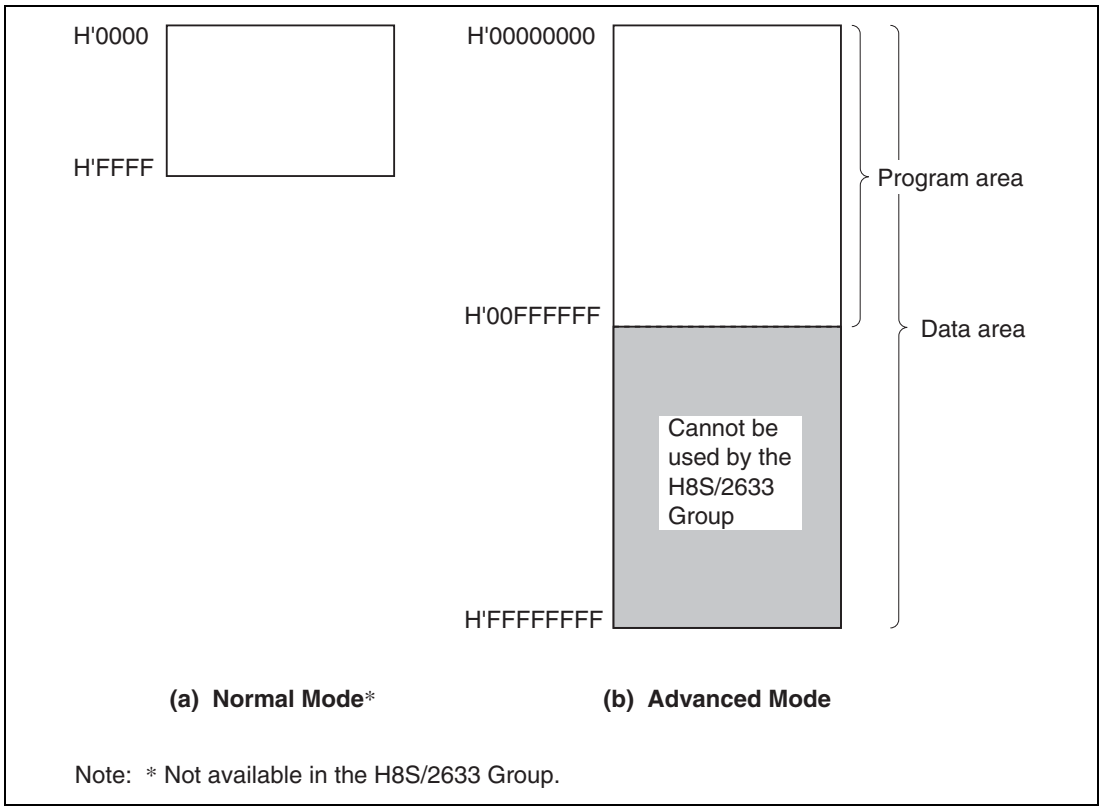


Figure 2.6 Memory Map

5.6.2 Block Diagram

Figure 5.9 shows a block diagram of the DTC and DMAC interrupt controller.

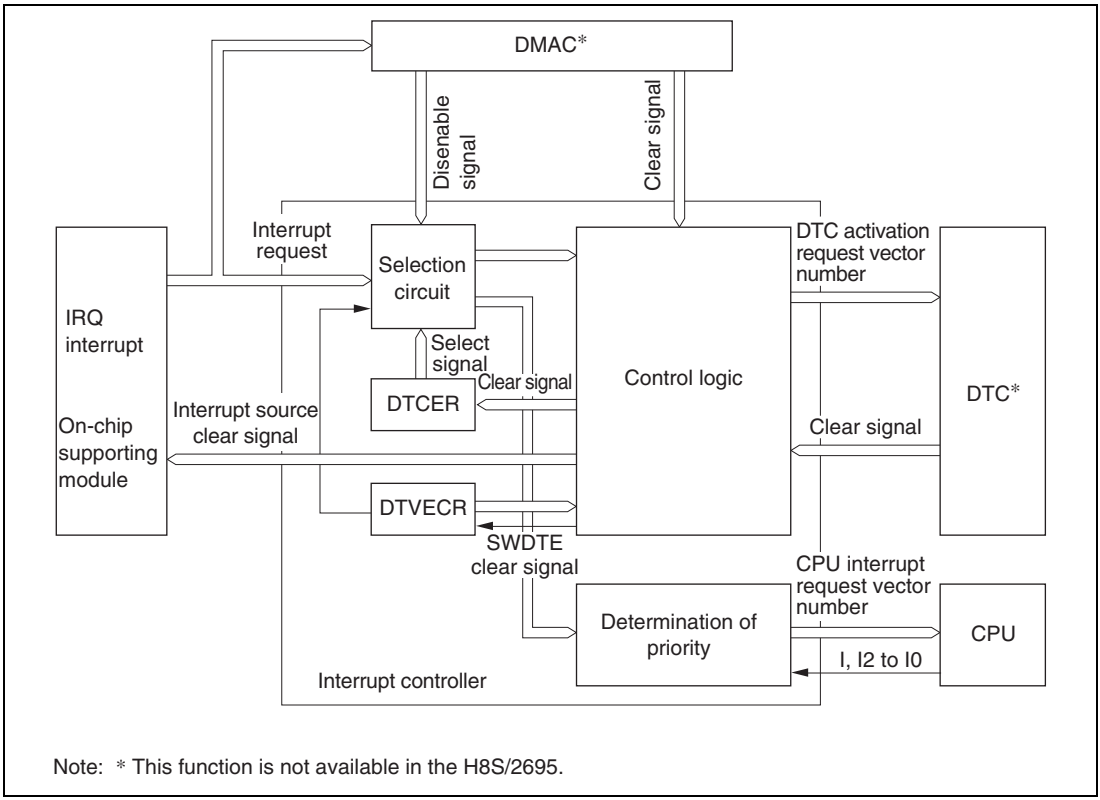


Figure 5.9 Interrupt Control for DTC* and DMAC*

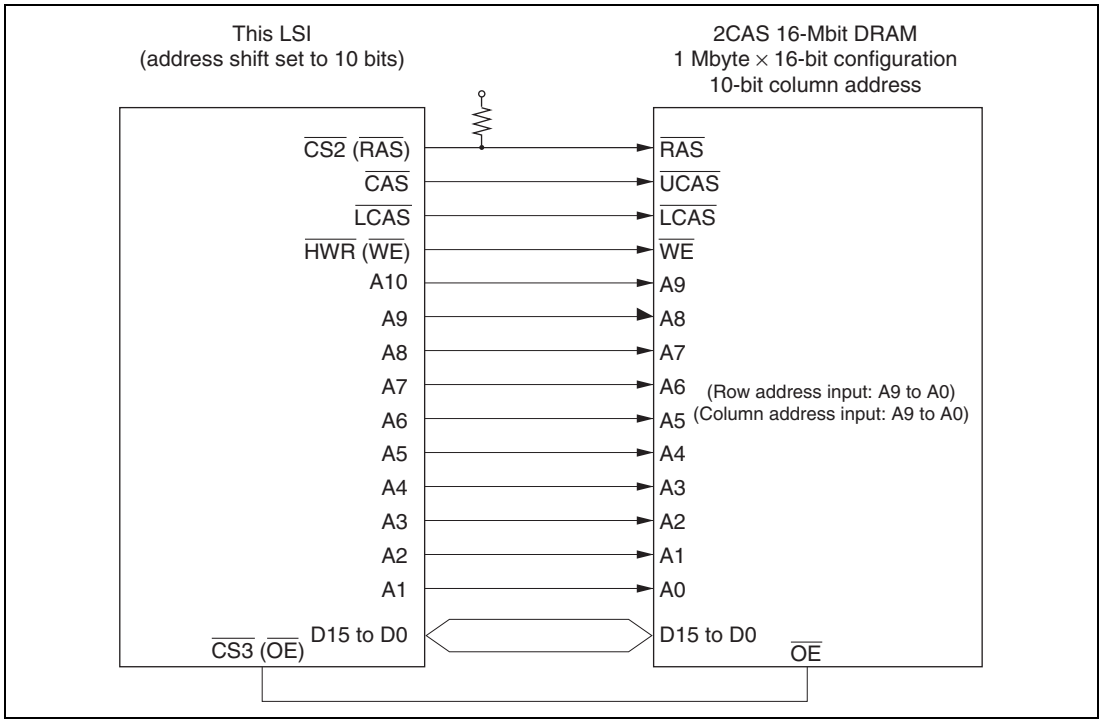


Figure 7.21 Example Connection of EDO Page Mode DRAM (OES=1)

7.5.10 Burst Operation

In addition to full DRAM access (normal DRAM access), in which the row address is output each time the data in DRAM is accessed, there is also a high-speed page mode that allows high-speed access (burst access). In this method, if the same row address is accessed successively, the row address is output once and then only the column address is changed. Burst access is selected by setting the BE bit of the MCR to 1.

10B.9.3 Pin Functions for Each Mode

(1) Modes 4 and 5

In modes 4 and 5, port C pins function as address outputs automatically.

Figure 10B.9 shows the port C pin functions.

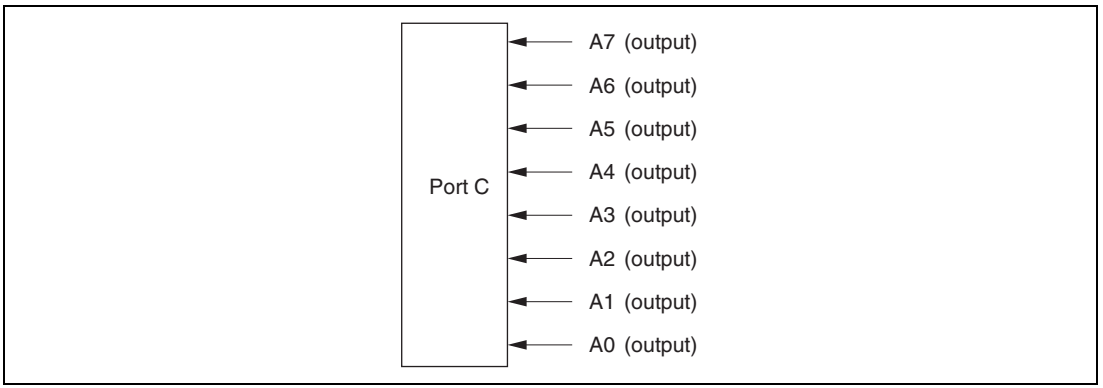


Figure 10B.9 Port C Pin Functions (Modes 4 and 5)

(2) Mode 6

In mode 6, port C pints function as address outputs or input ports and I/O can be specified in bit units. When each bit in PCDDR is set to 1, the corresponding pin functions as an address output and when the bit cleared to 0, the pin functions as an input port.

Figure 10B.10 shows the port C pin functions.

Pin **Selection Method and Pin Functions**

PF3/LWR/ADTRG/ IRQ3

The pin function is switched as shown below according to the operating mode, the bus mode, A/D converter bits TRGS1 and TRGS0, and bit PF3DDR.

Operating mode	Modes 4 to 6			Mode 7	
Bus mode	16-bit bus mode	8-bit bus mode		—	
PF3DDR	—	0	1	0	1
Pin function	LWR output pin	PF3 input pin	PF3 output pin	PF3 input pin	PF3 output pin
		ADTRG input pin*1			
		IRQ3 input pin*2			

Notes: 1. ADTRG input when TRGS0 = TRGS1 = 1.
2. When used as an external interrupt input pin, do not use as an I/O pin for another function.

PF2/ WAIT/ BREQO

The pin function is switched as shown below according to the combination of the operating mode and bits BREQOE, WAITE, ABW5 to ABW2, and PF2DDR.

Operating Mode	Modes 4 to 6				Mode 7	
BREQOE	0			1	—	
WAITE	0		1	—	—	
PF2DDR	0	1	—	—	0	1
Pin function	PF2 input pin	PF2 output pin	WAIT input pin	BREQO output pin	PF2 input pin	PF2 output pin

PF1/BACK

The pin function is switched as shown below according to the combination of the operating mode and bits BRLE and PF1DDR.

Operating Mode	Modes 4 to 6			Mode 7	
BRLE	0		1	—	
PF1DDR	0	1	—	0	1
Pin function	PF1 input pin	PF1 output pin	BACK output pin	PF1 input pin	PF1 output pin

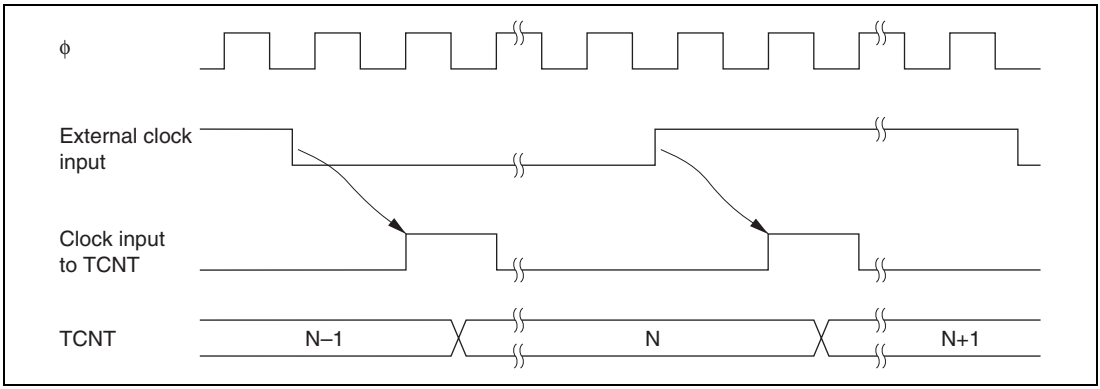


Figure 13.3 Count Timing for External Clock Input

13.3.2 Compare Match Timing

Setting of Compare Match Flags A and B (CMFA, CMFB): The CMFA and CMFB flags in TCSR are set to 1 by a compare match signal generated when the TCOR and TCNT values match. The compare match signal is generated at the last state in which the match is true, just before the timer counter is updated.

Therefore, when TCOR and TCNT match, the compare match signal is not generated until the next incrementation clock input. Figure 13.4 shows this timing.

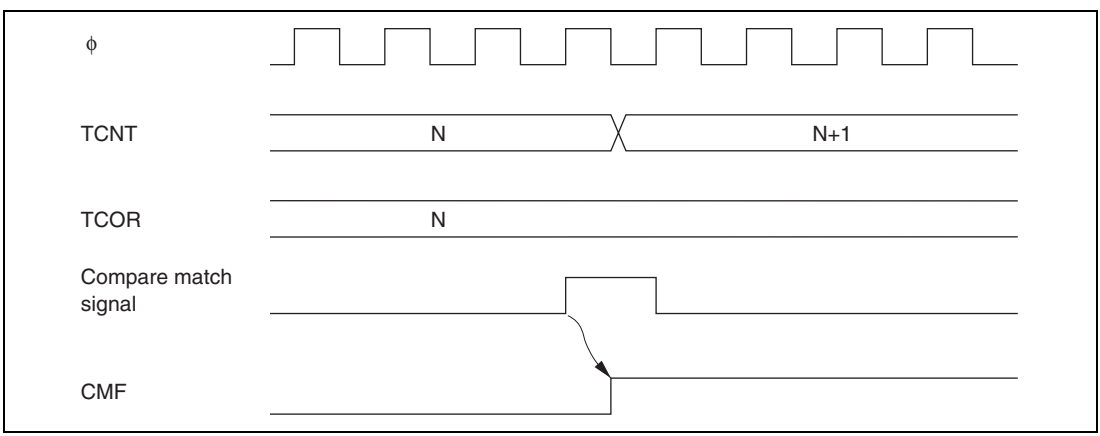
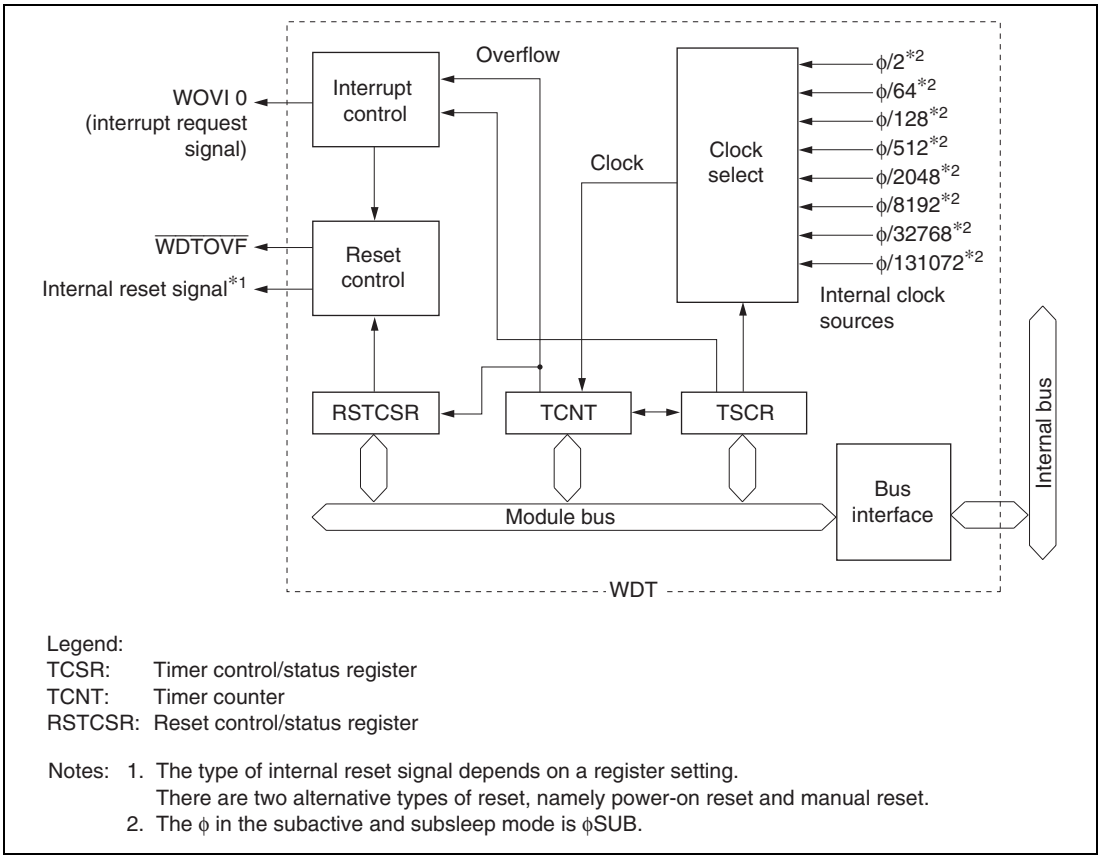


Figure 13.4 Timing of CMF Setting

15.1.2 Block Diagram

Figure 15.1 (a) and 15.1 (b) show a block diagram of the WDT.



Legend:
TCSR: Timer control/status register
TCNT: Timer counter
RSTCSR: Reset control/status register

Notes: 1. The type of internal reset signal depends on a register setting.
There are two alternative types of reset, namely power-on reset and manual reset.
2. The φ in the subactive and subsleep mode is φSUB.

Figure 15.1 (a) Block Diagram of WDT0

18.3.8 Operation Using the DTC*

The I²C bus format provides for selection of the slave device and transfer direction by means of the slave address and the R/$\overline{W}$ bit, confirmation of reception with the acknowledge bit, indication of the last frame, and so on. Therefore, continuous data transfer using the DTC* must be carried out in conjunction with CPU processing by means of interrupts.

Table 18.5 shows some examples of processing using the DTC*. These examples assume that the number of transfer data bytes is known in slave mode.

Note: * The DTC function is not available in the H8S/2695.

Table 18.5 Examples of Operation Using the DTC*

Item	Master Transmit Mode	Master Receive Mode	Slave Transmit Mode	Slave Receive Mode
Slave address + R/$\overline{W}$ bit transmission/ reception	Transmission by DTC* (ICDR write)	Transmission by CPU (ICDR write)	Reception by CPU (ICDR read)	Reception by CPU (ICDR read)
Dummy data read	—	Processing by CPU (ICDR read)	—	—
Actual data transmission/ reception	Transmission by DTC* (ICDR write)	Reception by DTC* (ICDR read)	Transmission by DTC* (ICDR write)	Reception by DTC* (ICDR read)
Dummy data (H'FF) write	—	—	Processing by DTC* (ICDR write)	—
Last frame processing	Not necessary	Reception by CPU (ICDR read)	Not necessary	Reception by CPU (ICDR read)
Transfer request processing after last frame processing	1st time: Clearing by CPU 2nd time: End condition issuance by CPU	Not necessary	Automatic clearing on detection of end condition during transmission of dummy data (H'FF)	Not necessary
Setting of number of DTC* transfer data frames	Transmission: Actual data count + 1 (+1 equivalent to slave address + R/$\overline{W}$ bits)	Reception: Actual data count	Transmission: Actual data count + 1 (+1 equivalent to dummy data (H'FF))	Reception: Actual data count

Note: * The DTC function is not available in the H8S/2695.

23B.6 Bus Master Clock Selection Circuit

The bus master clock selection circuit selects the system clock (ϕ) or one of the medium-speed clocks (ϕ/2, ϕ/4, ϕ/8, ϕ/16, and ϕ/32) to be supplied to the bus master, according to the settings of the SCK2 to SCK0 bits in SCKCR.

23B.7 Subclock Oscillator (This function is not available in the H8S/2695)

(1) Connecting 32.768kHz Quartz Resonator

To supply a clock to the subclock oscillator, connect a 32.768kHz quartz resonator, as shown in figure 23B.8. See section 23B.3.1, Notes on Board Design for notes on connecting crystal resonators.

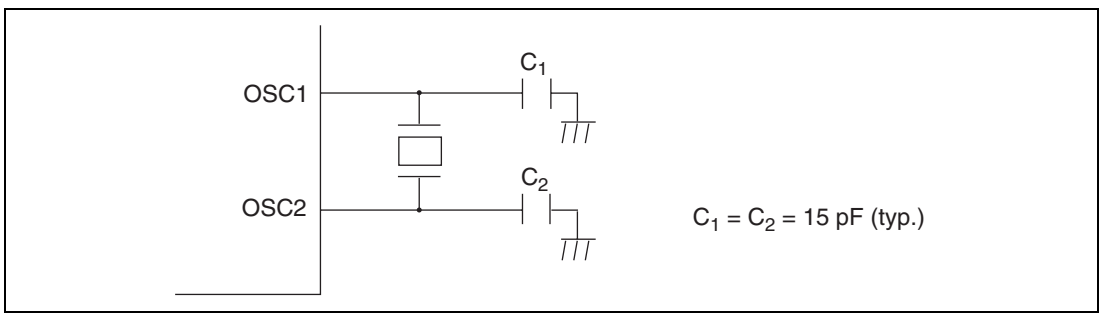


Figure 23B.8 Example Connection of 32.768kHz Crystal Resonator

Figure 23B.9 shows the equivalence circuit for a 32.768kHz resonator.

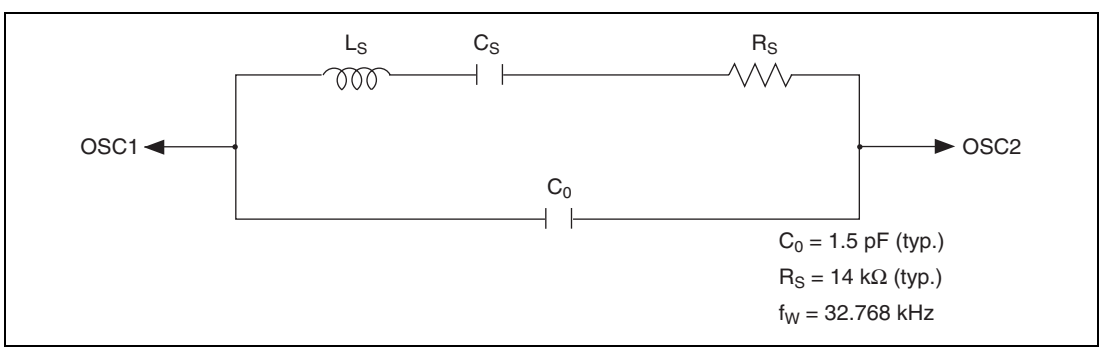


Figure 23B.9 Equivalence Circuit for 32.768kHz Resonator

Item		Symbol	Min	Typ	Max	Unit	Test Conditions
Input leakage current	$\overline{RES}$, FWE	$\lvert I_{in} \rvert$	—	—	1.0	µA	V_{in} = 0.5 V to PV_{CC} – 0.5 V
	$\overline{STBY}$, NMI, MD2 to MD0		—	—	1.0	µA	
	Ports 4, 9		—	—	1.0	µA	V_{in} = 0.5 V to AV_{CC} – 0.5 V
Three-state leakage current (off state)	Ports 1, 3, 7, A to G	$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V_{in} = 0.5 V to PV_{CC} – 0.5 V
MOS input pull-up current	Ports A to E	$-I_P$	50	—	300	µA	V_{in} = 0 V
Input capacitance	$\overline{RES}$	C_{in}	—	—	30	pF	V_{in} = 0 V
	NMI		—	—	30	pF	f = 1 MHz
	All input pins except $\overline{RES}$ and NMI		—	—	15	pF	T_a = 25°C
Current dissipation*2	Normal operation	I_{CC}*4	—	70	84	mA	f = 28 MHz
	Sleep mode		—	55	77	mA	f = 28 MHz
	All modules stopped		—	40	—	mA	f = 28 MHz (reference values)
	Medium-speed mode (φ/32)		—	54	—	mA	f = 28 MHz (reference values)
	Subactive mode		—	120	200	µA	Using 32.768 kHz crystal resonator
	Subsleep mode		—	70	150	µA	Using 32.768 kHz crystal resonator
	Watch mode		—	20	50	µA	Using 32.768 kHz crystal resonator
	Standby mode		—	0.1	5.0	µA	$T_a \leq 50°C$
			—	—	20		50°C < T_a

Mnemonic	Operand Size	Addressing Mode/ 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	Advanced
BTST	BTST #xx:3,@aa:32	B					8			¬ (#xx:3 of @aa:32)→Z	—	—	—	↕	—	—	5
	BTST Rn,Rd	B	2							¬ (Rn8 of Rd8)→Z	—	—	—	↕	—	—	1
	BTST Rn,@ERd	B		4						¬ (Rn8 of @ERd)→Z	—	—	—	↕	—	—	3
	BTST Rn,@aa:8	B					4			¬ (Rn8 of @aa:8)→Z	—	—	—	↕	—	—	3
	BTST Rn,@aa:16	B					6			¬ (Rn8 of @aa:16)→Z	—	—	—	↕	—	—	4
	BTST Rn,@aa:32	B					8			¬ (Rn8 of @aa:32)→Z	—	—	—	↕	—	—	5
BLD	BLD #xx:3,Rd	B	2							(#xx:3 of Rd8)→C	—	—	—	—	—	↕	1
	BLD #xx:3,@ERd	B		4						(#xx:3 of @ERd)→C	—	—	—	—	—	↕	3
	BLD #xx:3,@aa:8	B					4			(#xx:3 of @aa:8)→C	—	—	—	—	—	↕	3
	BLD #xx:3,@aa:16	B					6			(#xx:3 of @aa:16)→C	—	—	—	—	—	↕	4
	BLD #xx:3,@aa:32	B					8			(#xx:3 of @aa:32)→C	—	—	—	—	—	↕	5
BILD	BILD #xx:3,Rd	B	2							¬ (#xx:3 of Rd8)→C	—	—	—	—	—	↕	1
	BILD #xx:3,@ERd	B		4						¬ (#xx:3 of @ERd)→C	—	—	—	—	—	↕	3
	BILD #xx:3,@aa:8	B					4			¬ (#xx:3 of @aa:8)→C	—	—	—	—	—	↕	3
	BILD #xx:3,@aa:16	B					6			¬ (#xx:3 of @aa:16)→C	—	—	—	—	—	↕	4
	BILD #xx:3,@aa:32	B					8			¬ (#xx:3 of @aa:32)→C	—	—	—	—	—	↕	5
BST	BST #xx:3,Rd	B	2							C→(#xx:3 of Rd8)	—	—	—	—	—	—	1
	BST #xx:3,@ERd	B		4						C→(#xx:3 of @ERd)	—	—	—	—	—	—	4
	BST #xx:3,@aa:8	B					4			C→(#xx:3 of @aa:8)	—	—	—	—	—	—	4

Mnemonic	Operand Size	Addressing Mode/ 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	Advanced
BOR	BOR #xx:3,@aa:8	B					4			C∨(#xx:3 of @aa:8)→C	—	—	—	—	—	↕	3
	BOR #xx:3,@aa:16	B					6			C∨(#xx:3 of @aa:16)→C	—	—	—	—	—	↕	4
	BOR #xx:3,@aa:32	B					8			C∨(#xx:3 of @aa:32)→C	—	—	—	—	—	↕	5
BIOR	BIOR #xx:3,Rd	B	2							C∨[¬ (#xx:3 of Rd8)]→C	—	—	—	—	—	↕	1
	BIOR #xx:3,@ERd	B		4						C∨[¬ (#xx:3 of @ERd)]→C	—	—	—	—	—	↕	3
	BIOR #xx:3,@aa:8	B					4			C∨[¬ (#xx:3 of @aa:8)]→C	—	—	—	—	—	↕	3
	BIOR #xx:3,@aa:16	B					6			C∨[¬ (#xx:3 of @aa:16)]→C	—	—	—	—	—	↕	4
	BIOR #xx:3,@aa:32	B					8			C∨[¬ (#xx:3 of @aa:32)]→C	—	—	—	—	—	↕	5
BXOR	BXOR #xx:3,Rd	B	2							C⊕(#xx:3 of Rd8)→C	—	—	—	—	—	↕	1
	BXOR #xx:3,@ERd	B		4						C⊕(#xx:3 of @ERd)→C	—	—	—	—	—	↕	3
	BXOR #xx:3,@aa:8	B					4			C⊕(#xx:3 of @aa:8)→C	—	—	—	—	—	↕	3
	BXOR #xx:3,@aa:16	B					6			C⊕(#xx:3 of @aa:16)→C	—	—	—	—	—	↕	4
	BXOR #xx:3,@aa:32	B					8			C⊕(#xx:3 of @aa:32)→C	—	—	—	—	—	↕	5
BIXOR	BIXOR #xx:3,Rd	B	2							C⊕[¬ (#xx:3 of Rd8)]→C	—	—	—	—	—	↕	1
	BIXOR #xx:3,@ERd	B		4						C⊕[¬ (#xx:3 of @ERd)]→C	—	—	—	—	—	↕	3
	BIXOR #xx:3,@aa:8	B					4			C⊕[¬ (#xx:3 of @aa:8)]→C	—	—	—	—	—	↕	3
	BIXOR #xx:3,@aa:16	B					6			C⊕[¬ (#xx:3 of @aa:16)]→C	—	—	—	—	—	↕	4
	BIXOR #xx:3,@aa:32	B					8			C⊕[¬ (#xx:3 of @aa:32)]→C	—	—	—	—	—	↕	5

Instruction	Mnemonic	Size	Instruction Format									
			1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte	9th byte	10th byte
XORC	XORC #xx:8,CCR	B	0 5	IMM								
	XORC #xx:8,EXR	B	0 1	4 1	0 5	IMM						

Notes: 1. Bit 7 of the 4th byte of the MOV.L ERs, @(d:32,ERd) instruction can be either 1 or 0.
2. When using the TAS instruction, use register ER0, ER1, ER4, or ER5.
3. Only register ER0 to ER6 should be used when using the STM/LDM instruction.

Legend:

IMM:	Immediate data (2, 3, 8, 16, or 32 bits)
abs:	Absolute address (8, 16, 24, or 32 bits)
disp:	Displacement (8, 16, or 32 bits)
rs, rd, rn:	Register field (4 bits specifying an 8-bit or 16-bit register. The symbols rs, rd, and rn correspond to operand symbols Rs, Rd,and Rn.)
ers, erd, ern, erm:	Register field (3 bits specifying an address register or 32-bit register. The symbols ers, erd, ern, and erm correspond to operand symbols ERs, ERd, ERn, and ERm.)

The register fields specify general registers as follows.

Address Register 32-Bit Register		16-Bit Register		8-Bit Register	
Register Field	General Register	Register Field	General Register	Register Field	General Register
000	ER0	0000	R0	0000	R0H
001	ER1	0001	R1	0001	R1H
•	•	•	•	•	•
•	•	•	•	•	•
•	•	•	•	•	•
111	ER7	0111	R7	0111	R7H
		1000	E0	1000	R0L
		1001	E1	1001	R1L
		•	•	•	•
		•	•	•	•
		•	•	•	•
		1111	E7	1111	R7L

Instruction	H	N	Z	V	C	Definition
CLRMAC	—	—	—	—	—	
CMP	↕	↕	↕	↕	↕	H = Sm–4 · $\overline{Dm–4}$ + $\overline{Dm–4}$ · Rm–4 + Sm–4 · Rm–4 N = Rm Z = $\overline{Rm}$ · $\overline{Rm–1}$ · ····· · $\overline{R0}$ V = $\overline{Sm}$ · Dm · $\overline{Rm}$ + Sm · $\overline{Dm}$ · Rm C = Sm · $\overline{Dm}$ + $\overline{Dm}$ · Rm + Sm · Rm
DAA	*	↕	↕	*	↕	N = Rm Z = $\overline{Rm}$ · $\overline{Rm–1}$ · ····· · $\overline{R0}$ C: decimal arithmetic carry
DAS	*	↕	↕	*	↕	N = Rm Z = $\overline{Rm}$ · $\overline{Rm–1}$ · ····· · $\overline{R0}$ C: decimal arithmetic borrow
DEC	—	↕	↕	↕	—	N = Rm Z = $\overline{Rm}$ · $\overline{Rm–1}$ · ····· · $\overline{R0}$ V = Dm · $\overline{Rm}$
DIVXS	—	↕	↕	—	—	N = Sm · $\overline{Dm}$ + $\overline{Sm}$ · Dm Z = $\overline{Sm}$ · $\overline{Sm–1}$ · ····· · $\overline{S0}$
DIVXU	—	↕	↕	—	—	N = Sm Z = $\overline{Sm}$ · $\overline{Sm–1}$ · ····· · $\overline{S0}$
EEPMOV	—	—	—	—	—	
EXTS	—	↕	↕	0	—	N = Rm Z = $\overline{Rm}$ · $\overline{Rm–1}$ · ····· · $\overline{R0}$
EXTU	—	0	↕	0	—	Z = $\overline{Rm}$ · $\overline{Rm–1}$ · ····· · $\overline{R0}$
INC	—	↕	↕	↕	—	N = Rm Z = $\overline{Rm}$ · $\overline{Rm–1}$ · ····· · $\overline{R0}$ V = $\overline{Dm}$ · Rm
JMP	—	—	—	—	—	
JSR	—	—	—	—	—	
LDC	↕	↕	↕	↕	↕	Stores the corresponding bits of the result. No flags change when the operand is EXR.
LDM*2	—	—	—	—	—	
LDMAC	—	—	—	—	—	

Address	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module Name	Data Bus Width (bits)
H'FFA2 (write)	TCSR1/ TCNT1	OVF	WT/$\overline{IT}$	TME	PSS	RST/ $\overline{NMI}$	CKS2	CKS1	CKS0	WDT1	16
H'FFA3 (read)	TCNT1										
H'FFA4	DADR0									D/A0, D/A1	8
H'FFA5	DADR1										
H'FFA6	DACR01	DAOE1	DAOE0	DAE	—	—	—	—	—		
H'FFA8	FLMCR1	FWE	SWE1	ESU1	PSU1	EV1	PV1	E1	P1	FLASH	8
H'FFA9	FLMCR2	FLER	—	—	—	—	—	—	—		
H'FFAA	EBR1	EB7	EB6	EB5	EB4	EB3	EB2	EB1	EB0		
H'FFAB	EBR2	—	—	—	—	EB11	EB10	EB9	EB8		
H'FFAC	FLPWCR	PDWND	—	—	—	—	—	—	—		
H'FFB0	PORT1	P17	P16	P15	P14	P13	P12	P11	P10	Port	8
H'FFB2	PORT3	P37	P36	P35	P34	P33	P32	P31	P30		
H'FFB3	PORT4	P47	P46	P45	P44	P43	P42	P41	P40		
H'FFB6	PORT7	P77	P76	P75	P74	P73	P72	P71	P70		
H'FFB8	PORT9	P97	P96	P95	P94	P93	P92	P91	P90		
H'FFB9	PORTA	—	—	—	—	PA3	PA2	PA1	PA0		
H'FFBA	PORTB	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0		
H'FFBB	PORTC	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0		
H'FFBC	PORTD	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0		
H'FFBD	PORTE	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0		
H'FFBE	PORTF	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0		
H'FFBF	PORTG	—	—	—	PG4	PG3	PG2	PG1	PG0		

Note: Undefined and reserved addresses are for use in future functional expansion or have test registers, etc., assigned to them. These registers must not be accessed, since operation in the event of such access, and subsequent operation, cannot be guaranteed.

TMDR0—Timer Mode Register 0 **H'FF11** **TPU0**
TMDR3—Timer Mode Register 3 **H'FE81** **TPU3**

Channel 0: TMDR0

Channel 3: TMDR3

Bit	:	7	6	5	4	3	2	1	0
		—	—	BFB	BFA	MD3	MD2	MD1	MD0
Initial value	:	1	1	0	0	0	0	0	0
R/W	:	—	—	R/W	R/W	R/W	R/W	R/W	R/W

Buffer operation B

0	Normal TGRB operation.
1	Buffer operation of TGRB and TGRD.

Buffer operation A

0	Normal TGRA operation.
1	Buffer operation of TGRA and TGRC.

Modes 3 to 0

MD3*1	MD2*2	MD1	MD0	
0	0	0	0	Normal operation
			1	Reserved
		1	0	PWM mode 1
			1	PWM mode 2
	1	0	0	Phase calculation mode 1
			1	Phase calculation mode 2
		1	0	Phase calculation mode 3
			1	Phase calculation mode 4
1	*	*	*	—

* : Don't care

Notes: 1. MD3 is a reserved bit. Only write 0 to this bit.
2. Phase calculation mode cannot be set for channels 0 and 3. Only write 0 to MD2.

MAR1AH—Memory Address Register 1AH **H'FEF0** **DMAC**
MAR1AL—Memory Address Register 1AL **H'FEF2** **DMAC**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MAR1AH	—	—	—	—	—	—	—	—								
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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MAR1AL																
Initial value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

In short address mode: Specifies transfer destination/transfer source address
In full address mode: Specifies transfer destination

*: Undefined

ETCR1A—Transfer Count Register 1A **H'FEF6** **DMAC**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETCR1A																
Initial value	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Sequential mode, Idle mode, Normal mode: Transfer counter

Repeat mode: Holds number of transfers; Transfer counter

Block transfer mode: Holds block size; Block size counter

*: Undefined

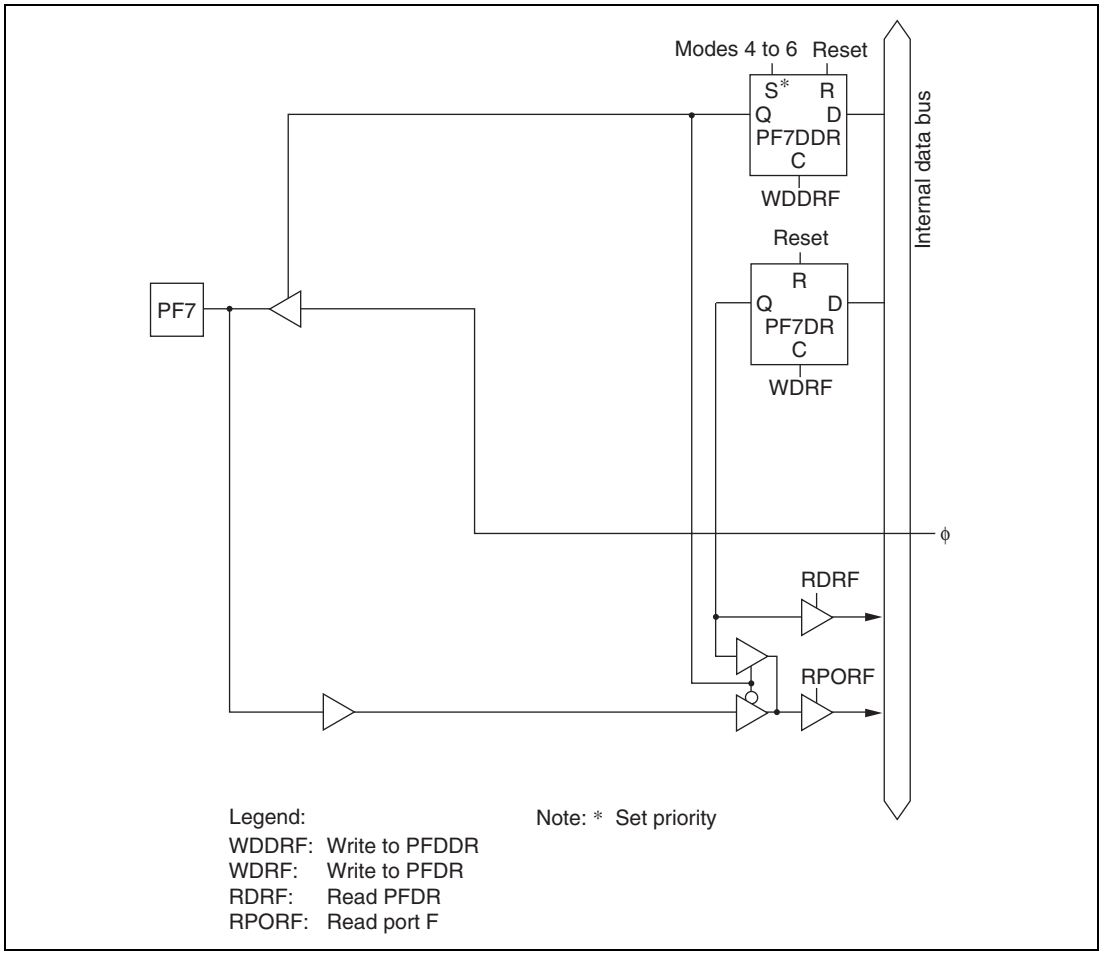


Figure C.23 (h) Port F Block Diagram (Pin PF7)