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	Not For New Designs
Core Processor	H8S/2000
Core Size	16-Bit
Speed	33MHz
Connectivity	I²C, IrDA, LPC, SCI, SmartCard
Peripherals	POR, PWM, WDT
Number of I/O	106
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	40K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x10b; D/A 2x8b
Oscillator Type	External
Operating Temperature	-20°C ~ 75°C (TA)
Mounting Type	Surface Mount
Package / Case	144-TQFP
Supplier Device Package	144-TQFP (16x16)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df2166vt33wv

(2) 8-Bit, 3-State Data Access Space: Figure 6.15 shows the bus timing for an 8-bit, 3-state access space. When an 8-bit access space is accessed, the upper half (AD15 to AD8) of the data bus is used. Wait states can be inserted.

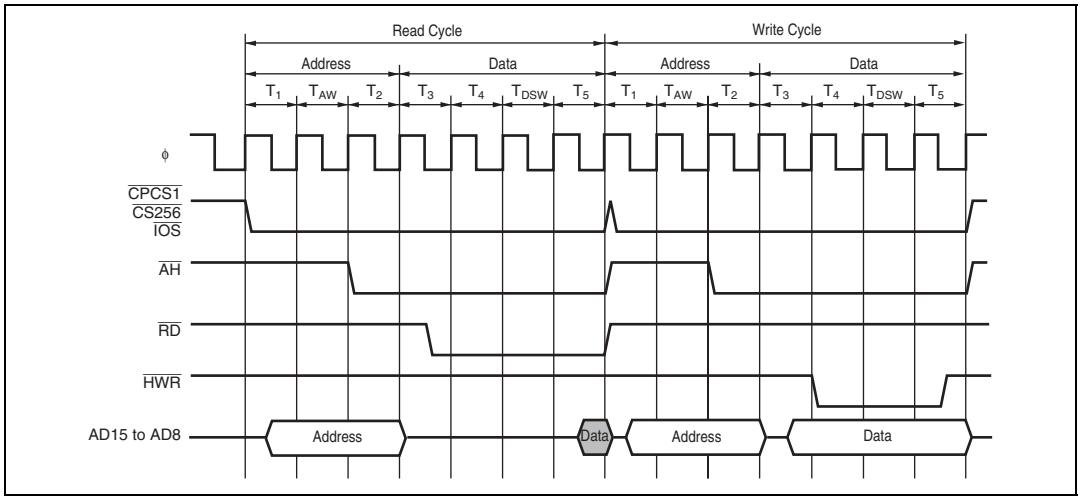


Figure 6.15 Bus Timing for 8-Bit, 3-State Access Space

(3) 16-Bit, 2-State Data Access Space: Figures 6.16 to 6.21 show bus timings for a 16-bit, 2-state access space. When a 16-bit access space is accessed, the upper half (AD15 to AD8) of the data bus is used for even addresses, and the lower half (AD7 to AD0) for odd addresses. Wait states cannot be inserted.

7.6.2 Repeat Mode

In repeat mode, one activation source transfers one byte or one word of data. Table 7.6 lists the register functions in repeat mode. From 1 to 256 transfers can be specified. Once the specified number of transfers has been completed, the initial states of the transfer counter and the address register that is specified as the repeat area is restored, and transfer is repeated. In repeat mode, the transfer counter value does not reach H'00, and therefore CPU interrupts cannot be requested when the DISEL bit in MRB is cleared to 0.

Table 7.6 Register Functions in Repeat Mode

Name	Abbreviation	Function
DTC source address register	SAR	Transfer source address
DTC destination address register	DAR	Transfer destination address
DTC transfer count register AH	CRAH	Holds number of transfers
DTC transfer count register AL	CRAL	Transfer Count
DTC transfer count register B	CRB	Not used

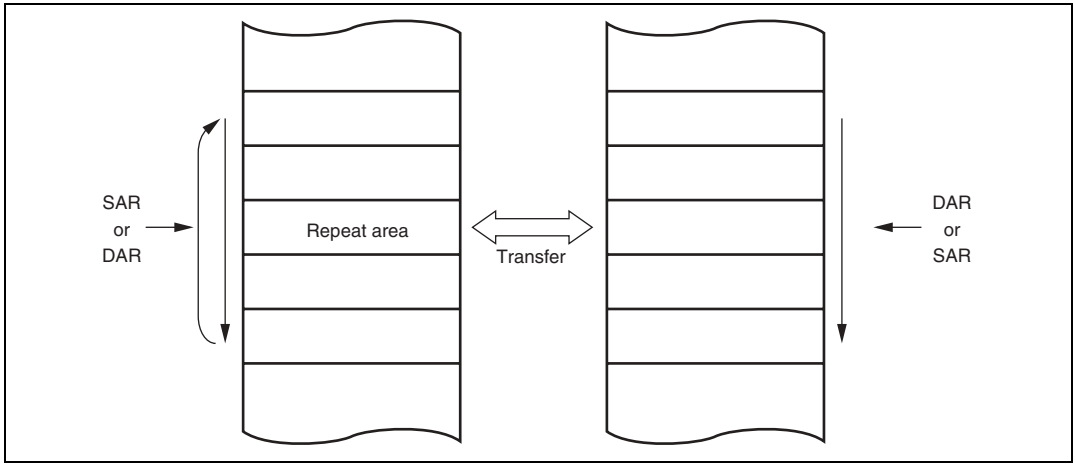


Figure 7.6 Memory Mapping in Repeat Mode

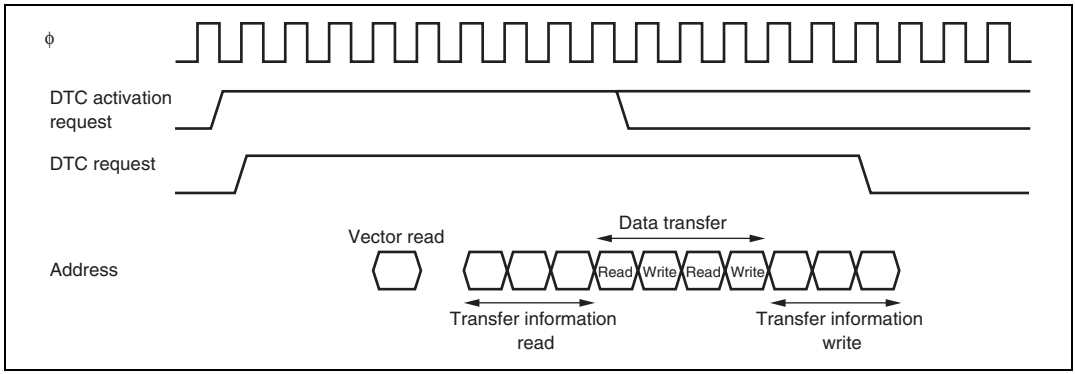


Figure 7.10 DTC Operation Timing (Example of Block Transfer Mode, with Block Size of 2)

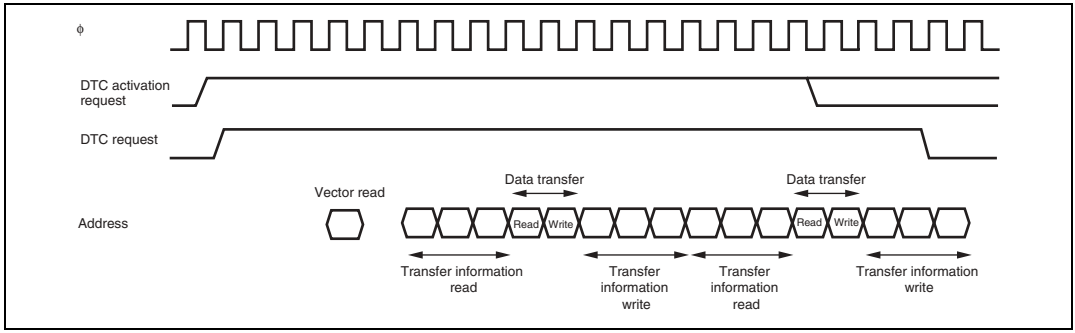


Figure 7.11 DTC Operation Timing (Example of Chain Transfer)

7.6.7 Number of DTC Execution States

Table 7.8 lists the execution status for a single DTC data transfer, and table 7.9 shows the number of states required for each execution status.

Table 7.8 DTC Execution Status

Mode	Vector Read I	Register Information Read/Write J	Data Read K	Data Write L	Internal Operations M
Normal	1	6	1	1	3
Repeat	1	6	1	1	3
Block transfer	1	6	N	N	3

[Legend]

N: Block size (initial setting of CRAH and CRAL)

8.6 Port 6

Port 6 is an 8-bit I/O port. Port 6 pins also function as the FRT input/output pin, keyboard input pin, and noise cancel input pin. Port 6 functions change according to the operating mode. The port can be used as the extended data bus (lower eight bits). Port 6 has the following registers.

- Port 6 data direction register (P6DDR)
- Port 6 data register (P6DR)
- Port 6 pull-up MOS control register (KMPCR6)
- System control register 2 (SYSCR2)
- Noise canceler enable register (P6NCE)
- Noise canceler decision control register (P6NCMC)
- Noise cancel cycle setting register (P6NCCS)

8.6.1 Port 6 Data Direction Register (P6DDR)

The individual bits of P6DDR specify input or output for the pins of port 6.

Bit	Bit Name	Initial Value	R/W	Description
7	P67DDR	0	W	Normal extended mode (16-bit data bus): The port functions as the data bus regardless of the values in these bits. Other mode: If port 6 pins are specified for use as the general I/O port, the corresponding port 6 pins are output ports when the P6DDR bits are set to 1, and input ports when cleared to 0.
6	P66DDR	0	W	
5	P65DDR	0	W	
4	P64DDR	0	W	
3	P63DDR	0	W	
2	P62DDR	0	W	
1	P61DDR	0	W	
0	P60DDR	0	W	

8.6.2 Port 6 Data Register (P6DR)

P6DR stores output data for the port 6 pins.

Bit	Bit Name	Initial Value	R/W	Description
7	P67DR	0	R/W	Normal extended mode (16-bit data bus):
6	P66DR	0	R/W	If a port 6 read is performed while the P6DDR bits are set to 1, the P6DR values are read. If a port 6 read is performed while the P6DDR bits are cleared to 0, 1 is read.
5	P65DR	0	R/W	
4	P64DR	0	R/W	
3	P63DR	0	R/W	Other mode:
2	P62DR	0	R/W	P6DR stores output data for the port 6 pins that are used as the general output port.
1	P61DR	0	R/W	
0	P60DR	0	R/W	If a port 6 read is performed while the P6DDR bits are set to 1, the P6DR values are read. If a port 6 read is performed while the P6DDR bits are cleared to 0, the pin states are read.

8.6.3 Port 6 Pull-Up MOS Control Register (KMPCR6)

KMPCR6 controls the port 6 built-in input pull-up MOSs. This register is accessible when SYSCR KINWUE is 1. See section 3.2.2, System Control Register (SYSCR).

Bit	Bit Name	Initial Value	R/W	Description
7	KM7PCR	0	R/W	Normal extended mode (16-bit data bus):
6	KM6PCR	0	R/W	Operation is not affected.
5	KM5PCR	0	R/W	Other mode:
4	KM4PCR	0	R/W	When the pins are in input state, the corresponding input pull-up MOS is turned on when a KMPCR6 bit is set to 1.
3	KM3PCR	0	R/W	
2	KM2PCR	0	R/W	
1	KM1PCR	0	R/W	
0	KM0PCR	0	R/W	

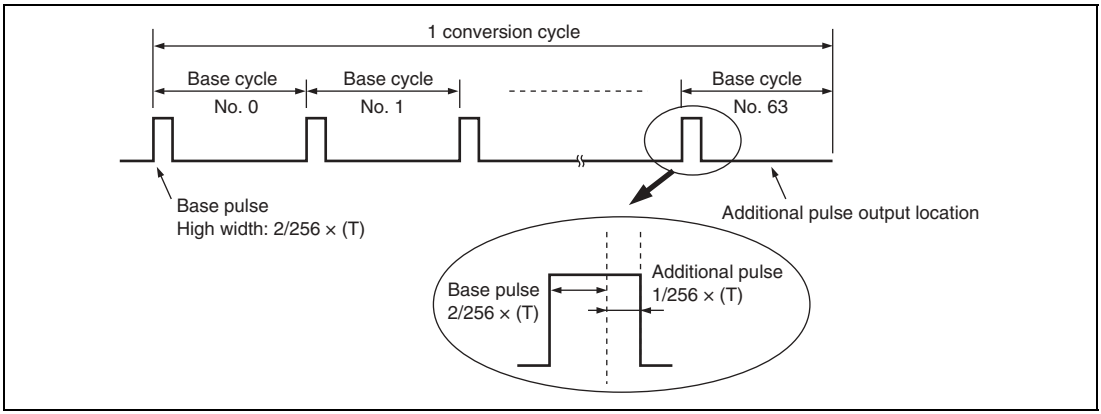


Figure 10.6 Output Waveform when DADR = H'0207 (OS = 1)

However, when CFS = 0 (base cycle = resolution (T) × 64), the duty cycle of the base pulse is determined by the upper six bits and the locations of the additional pulses by the subsequent eight bits with a method similar to as above.

11.5 Operation Timing

11.5.1 FRC Increment Timing

Figure 11.3 shows the FRC increment timing with an internal clock source. Figure 11.4 shows the increment timing with an external clock source. The pulse width of the external clock signal must be at least 1.5 system clocks (ϕ). The counter will not increment correctly if the pulse width is shorter than 1.5 system clocks (ϕ).

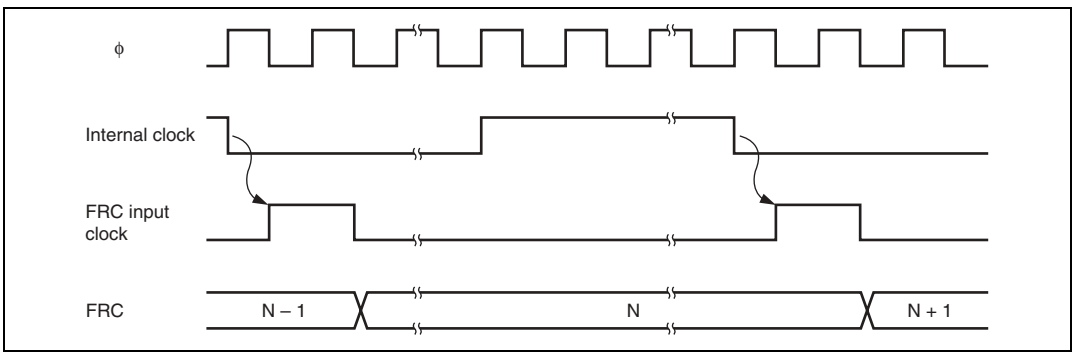


Figure 11.3 Increment Timing with Internal Clock Source

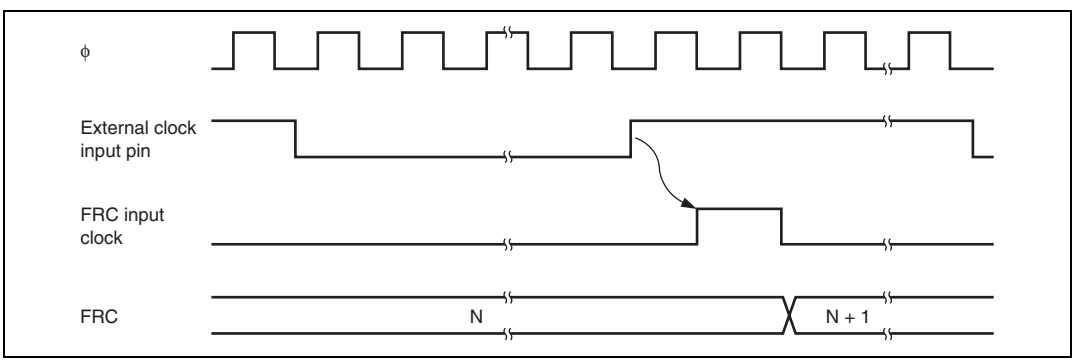


Figure 11.4 Increment Timing with External Clock Source

11.7.2 Conflict between FRC Write and Increment

If an FRC increment pulse is generated during the state after an FRC write cycle, the write takes priority and FRC is not incremented. Figure 11.18 shows the timing for this type of conflict.

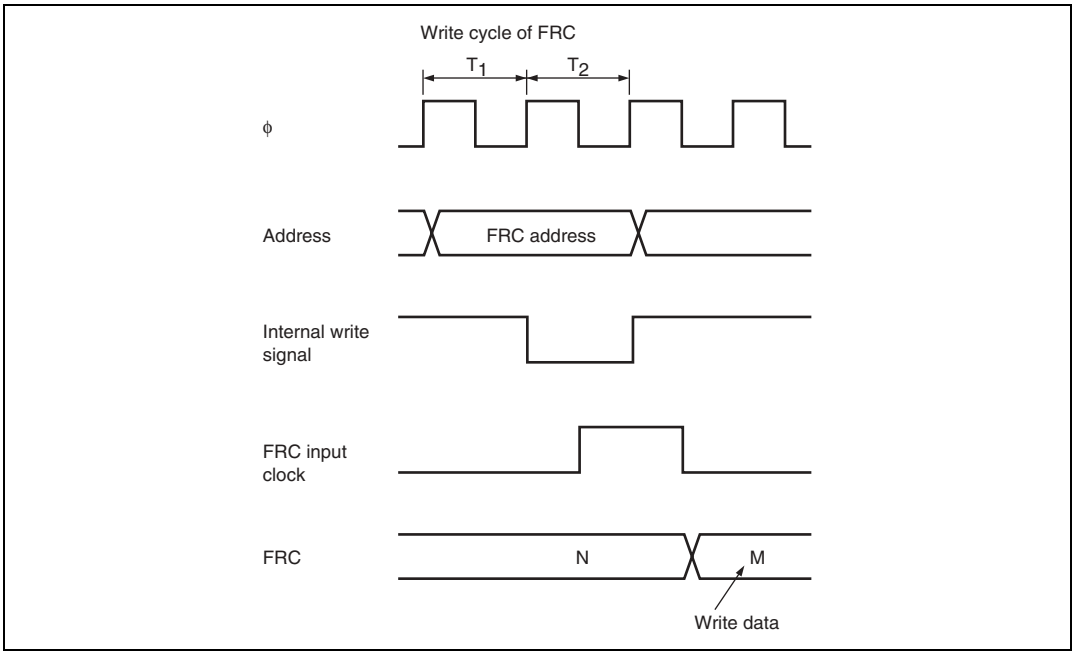


Figure 11.18 Conflict between FRC Write and Increment

ACS 4	ACS 2	ACS 1	ACS 0	Average Transfer Rate	System Clock (ϕ)	Operating Clock
1	0	1	1	230.392 kbps	20 MHz	Transfer rate × 16
1	1	0	0	115.196 kbps	24 MHz	Transfer rate × 16
1	1	0	1	230.392 kbps	24 MHz	Transfer rate × 16
1	1	1	0	460.784 kbps	24 MHz	Transfer rate × 16
1	1	1	1	720 kbps	24 MHz	Transfer rate × 8

14.6.4 Serial Data Reception (Clock Synchronous Mode)

Figure 14.21 shows an example of SCI operation for reception in clock synchronous mode. In serial reception, the SCI operates as described below.

1. The SCI performs internal initialization in synchronization with a synchronization clock input or output, starts receiving data, and stores the receive data in RSR.
2. If an overrun error (when reception of the next data is completed while the RDRF flag is still set to 1) occurs, the ORER bit in SSR is set to 1. If the RIE bit in SCR is set to 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR. The RDRF flag remains to be set to 1.
3. If reception finishes successfully, the RDRF bit in SSR is set to 1, and receive data is transferred to RDR. If the RIE bit in SCR is set to 1 at this time, an RXI interrupt request is generated. Because the RXI interrupt routine reads the receive data transferred to RDR before reception of the next receive data has finished, continuous reception can be enabled.

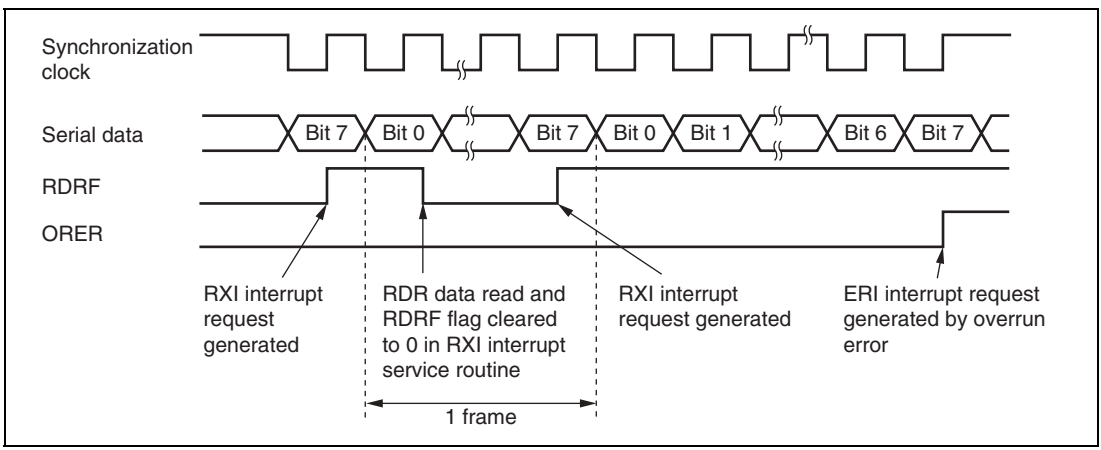


Figure 14.21 Example of SCI Receive Operation in Clock Synchronous Mode

Reception cannot be resumed while a receive error flag is set to 1. Accordingly, clear the ORER, FER, PER, and RDRF bits to 0 before resuming reception. Figure 14.22 shows a sample flowchart for serial data reception.

Table 15.4 Flags and Transfer States (Master Mode)

MST	TRS	BBSY	ESTP	STOP	IRTR	AASX	AL	AAS	ADZ	ACKB	ICDRF	ICDRE	State
1	1	0	0	0	0	0↓	0	0↓	0↓	0	—	0	Idle state (flag clearing required)
1	1	1↑	0	0	1↑	0	0	0	0	0	—	1↑	Start condition detected
1	—	1	0	0	—	0	0	0	0	—	—	—	Wait state
1	1	1	0	0	—	0	0	0	0	1↑	—	—	Transmission end (ACKE=1 and ACKB=1)
1	1	1	0	0	1↑	0	0	0	0	0	—	1↑	Transmission end with ICDRE=0
1	1	1	0	0	—	0	0	0	0	0	—	0↓	ICDR write with the above state
1	1	1	0	0	—	0	0	0	0	0	—	1	Transmission end with ICDRE=1
1	1	1	0	0	—	0	0	0	0	0	—	0↓	ICDR write with the above state or after start condition detected
1	1	1	0	0	1↑	0	0	0	0	0	—	1↑	Automatic data transfer from ICDRT to ICDRS with the above state
1	0	1	0	0	1↑	0	0	0	0	—	1↑	—	Reception end with ICDRF=0
1	0	1	0	0	—	0	0	0	0	—	0↓	—	ICDR read with the above state
1	0	1	0	0	—	0	0	0	0	—	1	—	Reception end with ICDRF=1
1	0	1	0	0	—	0	0	0	0	—	0↓	—	ICDR read with the above state
1	0	1	0	0	1↑	0	0	0	0	—	1↑	—	Automatic data transfer from ICDRS to ICDRR with the above state
0↓	0↓	1	0	0	—	0	1↑	0	0	—	—	—	Arbitration lost
1	—	0↓	0	0	—	0	0	0	0	—	—	0↓	Stop condition detected

[Legend]

0: 0-state retained 1: 1-state retained —: Previous state retained

0↓: Cleared to 0 1↑: Set to 1

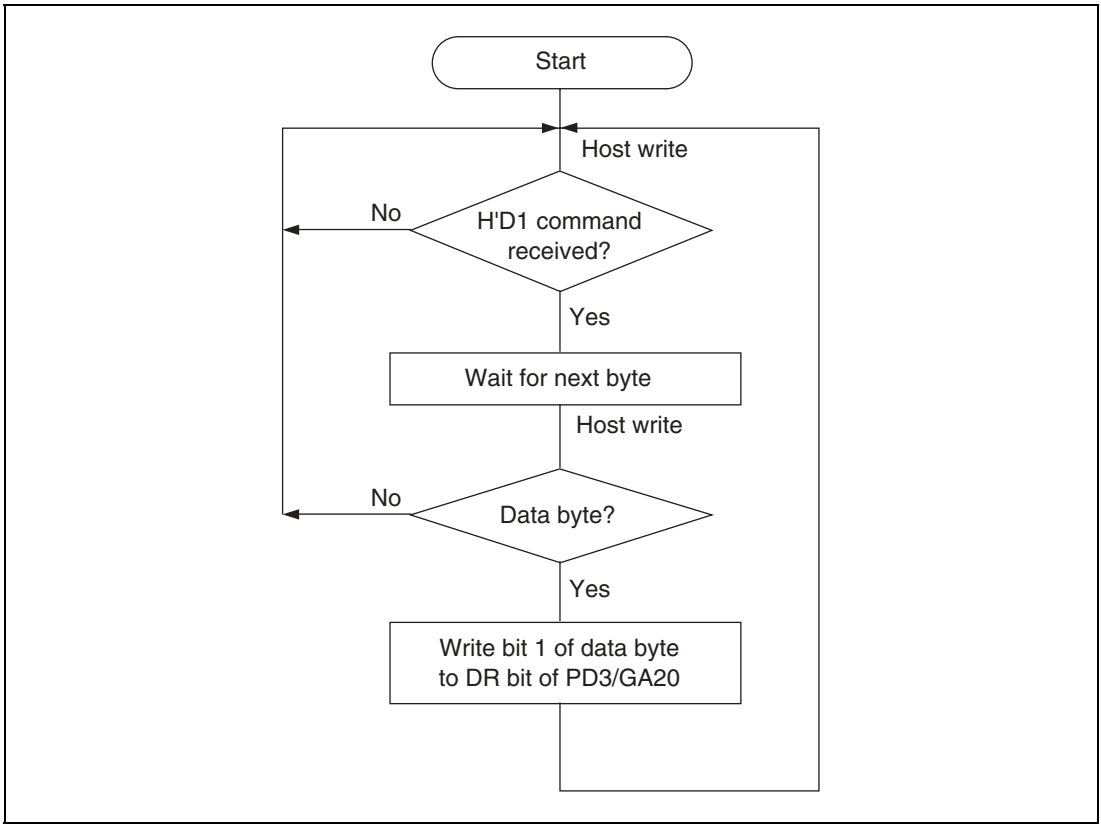
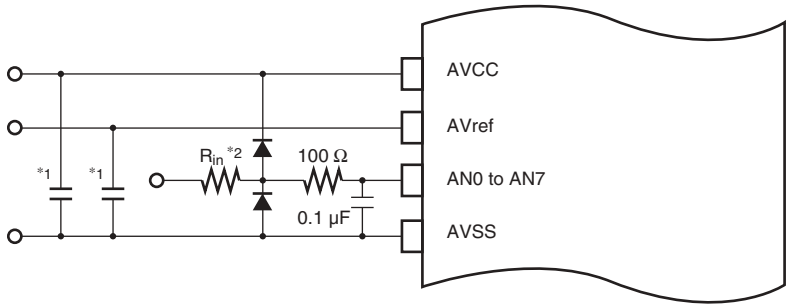
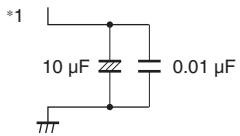


Figure 16.8 GA20 Output

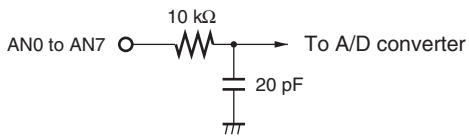


Notes: Values are reference values.



*2 R_{in}: Input impedance

Figure 18.7 Example of Analog Input Protection Circuit



Note: Values are reference values.

Figure 18.8 Analog Input Pin Equivalent Circuit

20.4.2 User Program Mode

The user MAT can be programmed/erased in user program mode. (The user boot MAT cannot be programmed/erased.)

Programming/erasing is executed by downloading the program in the microcomputer.

The overview flow is shown in figure 20.9.

High voltage is applied to internal flash memory during the programming/erasing processing. Therefore, transition to reset or hardware standby must not be executed. Doing so may damage or destroy flash memory. If reset is executed accidentally, reset must be released after the reset input period of 100 µs which is longer than normal.

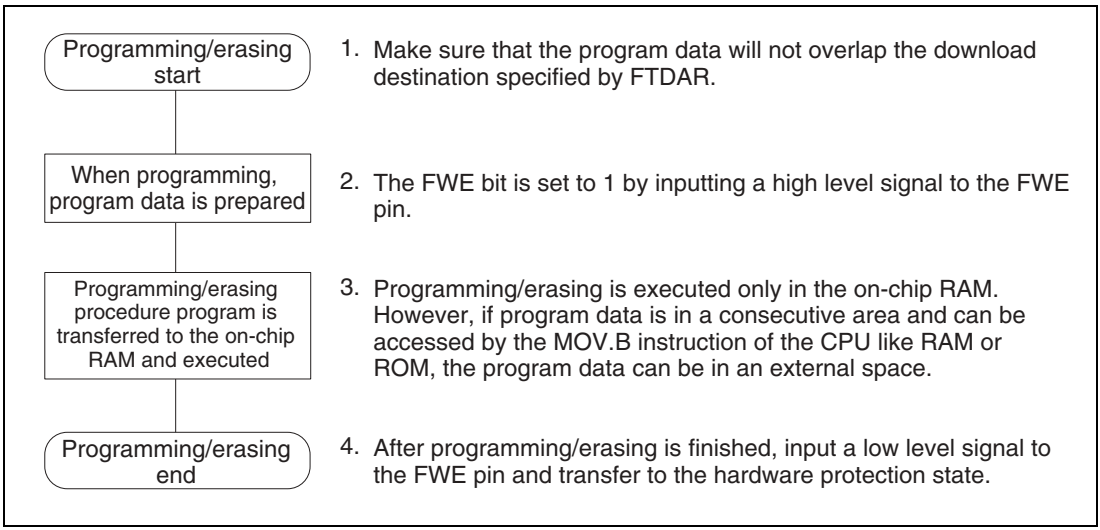


Figure 20.9 Programming/Erasing Overview Flow

	Storable/Executable Area				Selected MAT	
Item	On-chip RAM	User Boot MAT	External Space (Expanded Mode)	User MAT	User Boot MAT	Embedded Program Storage Area
Operation for Settings of Program Parameter	○	×	○	○		
Execution of Programming	○	×	×	○		
Determination of Program Result	○	×	○	○		
Operation for Program Error	○	×*2	○	○		
Operation for FKEY Clear	○	×	○	○		
Switching MATs by FMATS	○	×	×		○	

Notes: 1. Transferring the data to the on-chip RAM enables this area to be used.
2. Switching FMATS by a program in the on-chip RAM enables this area to be used.

This response consists of 4 bytes of data.

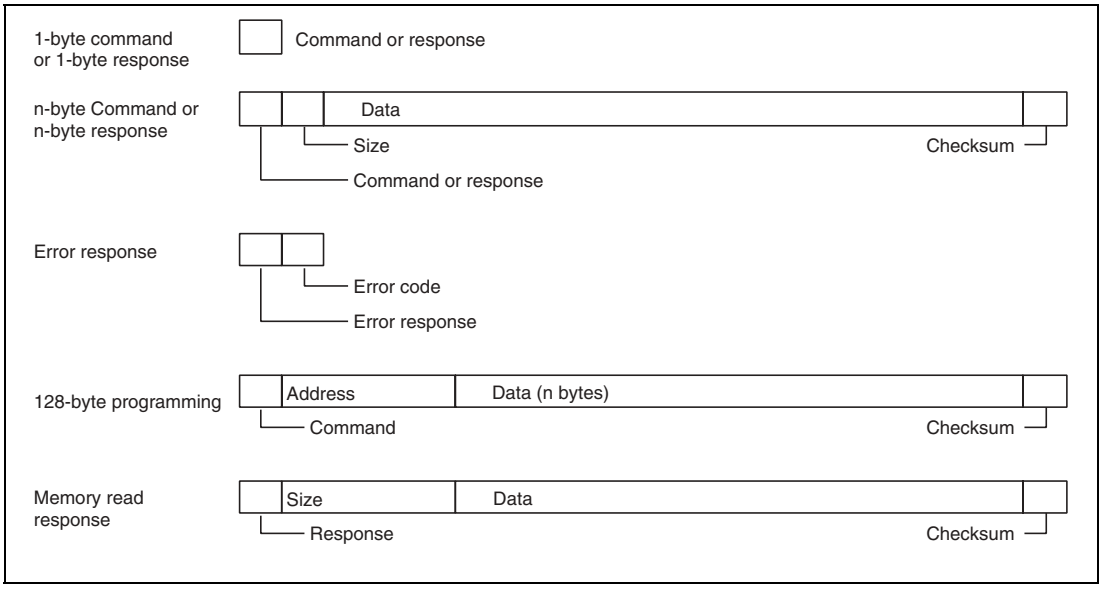


Figure 20.21 Communication Protocol Format

- Command (1 byte): Commands including inquiries, selection, programming, erasing, and checking
- Response (1 byte): Response to an inquiry
- Size (1 byte): The amount of data for transmission excluding the command, amount of data, and checksum
- Checksum (1 byte): The checksum is calculated so that the total of all values from the command byte to the SUM byte becomes H′00.
- Data (n bytes): Detailed data of a command or response
- Error response (1 byte): Error response to a command
- Error code (1 byte): Type of the error
- Address (4 bytes): Address for programming
- Data (n bytes): Data to be programmed (the size is indicated in the response to the programming unit inquiry.)
- Size (4 bytes): 4-byte response to a memory read

(k) New Bit-Rate Selection

The boot program will set a new bit rate and return the new bit rate.

This selection should be sent after sending the clock mode selection command.

Command	H'3F	Size	Bit rate	Input frequency
	Number of multiplication ratios	Multiplication ratio 1	Multiplication ratio 2	
	SUM			

- Command, H'3F, (1 byte): Selection of new bit rate
- Size (1 byte): The number of bytes that represents the bit rate, input frequency, number of multiplication ratios, and multiplication ratio
- Bit rate (2 bytes): New bit rate
  One hundredth of the value (e.g. when the value is 19200 bps, it will be 192, which is H′00C0.)
- Input frequency (2 bytes): Frequency of the clock input to the boot program
  This is valid to the hundredths place and represents the value in MHz multiplied by 100. (E.g. when the value is 20.00 MHz, it will be 2000, which is H'07D0.)
- Number of multiplication ratios (1 byte): The number of multiplication ratios to which the device can be set.
- Multiplication ratio 1 (1 byte) : The value of multiplication or division ratios for the main operating frequency
  Multiplication ratio (1 byte): The value of the multiplication ratio (e.g. when the clock frequency is multiplied by four, the multiplication ratio will be H'04.)
  Division ratio: The inverse of the division ratio, as a negative number (e.g. when the clock frequency is divided by two, the value of division ratio will be H'FE. H'FE = D'-2)
- Multiplication ratio 2 (1 byte): The value of multiplication or division ratios for the peripheral frequency
  Multiplication ratio (1 byte): The value of the multiplication ratio (e.g. when the clock frequency is multiplied by four, the multiplication ratio will be H'04.)
  (Division ratio: The inverse of the division ratio, as a negative number (E.g. when the clock is divided by two, the value of division ratio will be H'FE. H'FE = D'-2)
- SUM (1 byte): Checksum

Response	H'06

- Response, H'06, (1 byte): Response to selection of a new bit rate
  When it is possible to set the bit rate, the response will be ACK.

Error Response	H'BF	ERROR

- Error response, H'BF, (1 byte): Error response to selection of new bit rate

22.7 Clock Select Circuit

The clock select circuit selects the system clock that is used in this LSI.

A clock generated by the oscillator, to which the EXTAL and XTAL pins are input, and multiplied by the PLL circuit is selected as a system clock when returning from high-speed mode, medium-speed mode, sleep mode, the reset state, or standby mode.

In subactive mode, subsleep mode, or watch mode, a subclock input from the EXCL pin is selected as a system clock when the EXCLE bit in LPWRCR is 1. At this time, modules such as the CPU, TMR_0, TMR_1, WDT_0, WDT_1, ports, and interrupt controller and their functions operate on the ϕSUB. The count clock and sampling clock for each timer are divided ϕSUB clocks.

22.8 Usage Notes

22.8.1 Note on Resonator

Since all kinds of characteristics of the resonator are closely related to the board design by the user, use the example of resonator connection in this document for only reference; be sure to use an resonator that has been sufficiently evaluated by the user. Consult with the resonator manufacturer about the resonator circuit ratings which vary depending on the stray capacitances of the resonator and installation circuit. Make sure the voltage applied to the oscillation pins do not exceed the maximum rating.

22.8.2 Notes on Board Design

When using a crystal resonator, the crystal resonator and its load capacitors should be placed as close as possible to the EXTAL and XTAL pins. Other signal lines should be routed away from the oscillation circuit to prevent inductive interference with the correct oscillation as shown in figure 22.5.

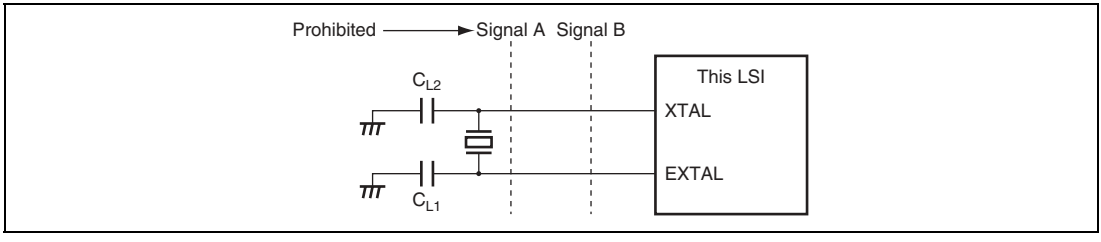


Figure 22.5 Note on Board Design of Oscillation Circuit Section

23.7 Watch Mode

The CPU makes a transition to watch mode when the SLEEP instruction is executed in high-speed mode or subactive mode with the SSBY bit in SBYCR set to 1, the DTON bit in LPWRCR cleared to 0, and the PSS bit in TCSR (WDT_1) set to 1.

In watch mode, the CPU is stopped and peripheral modules other than WDT_1 are also stopped. The contents of the CPU's internal registers, several on-chip peripheral module registers, and on-chip RAM data are retained and the I/O ports retain their values before transition as long as the prescribed voltage is supplied.

Watch mode is exited by an interrupt (WOVI1, NMI, IRQ0 to IRQ15, KIN0 to KIN15, or WUE8 to WUE15), $\overline{\text{RES}}$ pin input, or $\overline{\text{STBY}}$ pin input.

When an interrupt occurs, watch mode is exited and a transition is made to high-speed mode or medium-speed mode when the LSON bit in LPWRCR cleared to 0 or to subactive mode when the LSON bit is set to 1. When a transition is made to high-speed mode, a stable clock is supplied to the entire LSI and interrupt exception handling starts after the time set in the STS2 to STS0 bits in SBYCR has elapsed.

In the case of an IRQ0 to IRQ15 interrupt, watch mode is not exited if the corresponding enable bit has been cleared to 0 or the interrupt is masked by the CPU. In the case of a KIN0 to KIN15 or WUE8 to WUE15 interrupt, watch mode is not exited if input is disabled or the interrupt is masked by the CPU. In the case of an interrupt from the on-chip peripheral modules, watch mode is not exited if the interrupt enable register has been set to disable the reception of that interrupt or the interrupt is masked by the CPU.

When the $\overline{\text{RES}}$ pin is driven low, system clock oscillation starts. Simultaneously with the start of system clock oscillation, the system clock is supplied to the entire LSI. Note that the $\overline{\text{RES}}$ pin must be held low until clock oscillation is settled. If the $\overline{\text{RES}}$ pin is driven high after the clock oscillation settling time has passed, the CPU begins reset exception handling.

If the $\overline{\text{STBY}}$ pin is driven low, the LSI enters hardware standby mode.

23.9 Subactive Mode

The CPU makes a transition to subactive mode when the SLEEP instruction is executed in high-speed mode with the SSBY bit in SBYCR set to 1, the DTON bit and LSON bit in LPWRCR set to 1, and the PSS bit in TCSR (WDT_1) set to 1. When an interrupt occurs in watch mode, and if the LSON bit in LPWRCR is 1, a direct transition is made to subactive mode. Similarly, if an interrupt occurs in subsleep mode, a transition is made to subactive mode.

In subactive mode, the CPU operates at a low speed based on the subclock and sequentially executes programs. Peripheral modules other than TMR_0, TMR_1, WDT_0, and WDT_1 are also stopped.

When operating the CPU in subactive mode, the SCK2 to SCK0 bits in SBYCR must be cleared to 0.

Subactive mode is exited by the SLEEP instruction, $\overline{\text{RES}}$ pin input, or $\overline{\text{STBY}}$ pin input.

When the SLEEP instruction is executed with the SSBY bit in SBYCR set to 1, the DTON bit in LPWRCR cleared to 0, and the PSS bit in TCSR (WDT_1) set to 1, the CPU exits subactive mode and a transition is made to watch mode. When the SLEEP instruction is executed with the SSBY bit in SBYCR cleared to 0, the LSON bit in LPWRCR set to 1, and the PSS bit in TCSR (WDT_1) set to 1, a transition is made to subsleep mode. When the SLEEP instruction is executed with the SSBY bit in SBYCR set to 1, the DTON bit and LSON bit in LPWRCR set to 10, and the PSS bit in TCSR (WDT_1) set to 1, a direct transition is made to high-speed mode.

For details of direct transitions, see section 23.11, Direct Transitions.

When the $\overline{\text{RES}}$ pin is driven low, system clock oscillation starts. Simultaneously with the start of system clock oscillation, the system clock is supplied to the entire LSI. Note that the $\overline{\text{RES}}$ pin must be held low until the clock oscillation is settled. If the $\overline{\text{RES}}$ pin is driven high after the clock oscillation settling time has passed, the CPU begins reset exception handling.

If the $\overline{\text{STBY}}$ pin is driven low, the LSI enters hardware standby mode.