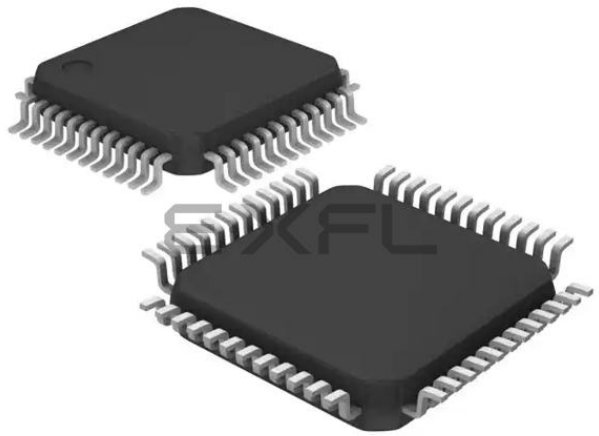

Welcome to E-XFL.COM

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	SH-2
Core Size	32-Bit Single-Core
Speed	50MHz
Connectivity	SCI
Peripherals	POR, PWM, WDT
Number of I/O	23
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-LQFP
Supplier Device Package	48-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df71240ad50fpv

4.4.2 Oscillation Stop Detection Control Register (OSCCR)

OSCCR is an 8-bit readable/writable register that has an oscillation stop detection flag and selects flag status output to an external pin. OSCCR can be accessed only in bytes.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	OSC STOP	-	OSC ERS
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	OSCSTOP	0	R	Oscillation Stop Detection Flag [Setting condition] • When a stop in the clock input is detected during normal operation [Clearing condition] • By a power-on reset input through the $\overline{\text{RES}}$ pin
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	OSCERS	0	R/W	Oscillation Stop Detection Flag Output Select Selects whether to output the oscillation stop detection flag signal through the $\overline{\text{WDTOVF}}$ pin. 0: Outputs only the WDT overflow signal through the $\overline{\text{WDTOVF}}$ pin 1: Outputs the WDT overflow signal and the oscillation stop detection flag signal through the $\overline{\text{WDTOVF}}$ pin

4.6 Oscillator

Clock pulses can be supplied from a connected crystal resonator or an external clock.

4.6.1 Connecting Crystal Resonator

A crystal resonator can be connected as shown in figure 4.2. Use the damping resistance (Rd) listed in table 4.6. Use a crystal resonator that has a resonance frequency of 10 to 12.5 MHz. It is recommended to consult the crystal resonator manufacturer concerning the compatibility of the crystal resonator and the LSI.

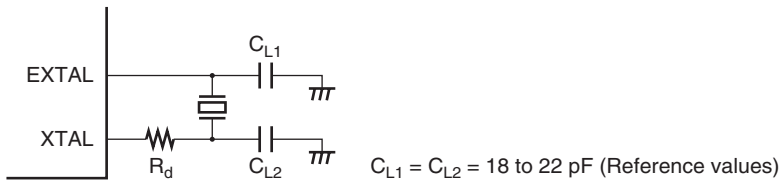

Figure 4.2 Connection of Crystal Resonator (Example)

Table 4.6 Damping Resistance Values (Reference Values)

Frequency (MHz)	10	12.5
Rd (Ω) (Reference Values)	0	0

Figure 4.3 shows an equivalent circuit of the crystal resonator. Use a crystal resonator with the characteristics listed in table 4.7.

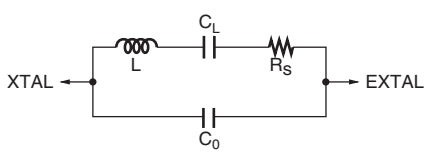

Figure 4.3 Crystal Resonator Equivalent Circuit

5.8 Usage Notes

5.8.1 Value of Stack Pointer (SP)

The SP value must always be a multiple of 4. If it is not, an address error will occur when the stack is accessed during exception handling.

5.8.2 Value of Vector Base Register (VBR)

The VBR value must always be a multiple of 4. If it is not, an address error will occur when the stack is accessed during exception handling.

5.8.3 Address Errors Caused by Stacking for Address Error Exception Handling

When the SP value is not a multiple of 4, an address error will occur when stacking for exception handling (interrupts, etc.) and address error exception handling will start after the first exception handling is ended. Address errors will also occur in the stacking for this address error exception handling. To ensure that address error exception handling does not go into an endless loop, no address errors are accepted at that point. This allows program control to be passed to the handling routine for address error exception and enables error processing.

When an address error occurs during exception handling stacking, the stacking bus cycle (write) is executed. When stacking the SR and PC values, the SP values for both are subtracted by 4, therefore, the SP value is still not a multiple of 4 after the stacking. The address value output during stacking is the SP value whose lower two bits are cleared to 0. So the write data stacked is undefined.

6.6 Interrupt Operation

6.6.1 Interrupt Sequence

The sequence of interrupt operations is explained below. Figure 6.3 is a flowchart of the operations.

1. The interrupt request sources send interrupt request signals to the interrupt controller.
2. The interrupt controller selects the highest priority interrupt from interrupt requests sent, according to the priority levels set in interrupt priority registers A to F and H to M (IPRA to IPRF and IPRH to IPRM). Interrupts that have lower-priority than that of the selected interrupt are ignored*. If interrupts that have the same priority level or interrupts within a same module occur simultaneously, the interrupt with the highest priority is selected according to the default priority shown in table 6.3.
3. The interrupt controller compares the priority level of the selected interrupt request with the interrupt mask bits (I3 to I0) in the status register (SR) of the CPU. If the priority level of the selected request is equal to or less than the level set in bits I3 to I0, the request is ignored. If the priority level of the selected request is higher than the level in bits I3 to I0, the interrupt controller accepts the request and sends an interrupt request signal to the CPU.
4. When the interrupt controller accepts an interrupt, a low level is output from the $\overline{\text{IRQOUT}}$ pin.
5. The CPU detects the interrupt request sent from the interrupt controller in the decode stage of an instruction to be executed. Instead of executing the decoded instruction, the CPU starts interrupt exception handling.
6. SR and PC are saved onto the stack.
7. The priority level of the accepted interrupt is copied to bits (I3 to I0) in SR.
8. When the accepted interrupt is sensed by level or is from an on-chip peripheral module, a high level is output from the $\overline{\text{IRQOUT}}$ pin. When the accepted interrupt is sensed by edge, a high level is output from the $\overline{\text{IRQOUT}}$ pin at the moment when the CPU starts interrupt exception processing instead of instruction execution as noted in 5. above. However, if the interrupt controller accepts an interrupt with a higher priority than the interrupt just to be accepted, the $\overline{\text{IRQOUT}}$ pin holds low level.
9. The CPU reads the start address of the exception handling routine from the exception vector table for the accepted interrupt, branches to that address, and starts executing the program. This branch is not a delayed branch.

4. When data access (address + data) is specified as a break condition:

   When a data value is added to the break conditions, the address of an instruction that is within two instructions of the instruction that matched the break condition is saved in the stack. At which instruction the break occurs cannot be determined accurately.

   When a delay slot instruction matches the condition, the branch destination address is saved in the stack. If the instruction following the instruction that matches the break condition is a branch instruction, the break may occur after the branch instruction or delay slot has finished. In this case, the branch destination address is saved in the stack.

7.3.6 PC Trace

1. Setting PCTE in BRCR to 1 enables PC traces. When branch (branch instruction, and interrupt exception) is generated, the branch source address and branch destination address are stored in BRSR and BRDR, respectively.
2. The values stored in BRSR and BRDR are as given below due to the kind of branch.
   - If a branch occurs due to a branch instruction, the address of the branch instruction is saved in BRSR and the address of the branch destination instruction is saved in BRDR.
   - If a branch occurs due to an interrupt or exception, the value saved in stack due to exception occurrence is saved in BRSR and the start address of the exception handling routine is saved in BRDR.
3. BRSR and BRDR have four pairs of queue structures. The top of queues is read first when the address stored in the PC trace register is read. BRSR and BRDR share the read pointer. Read BRSR and BRDR in order, the queue only shifts after BRDR is read. After switching the PCTE bit (in BRCR) off and on, the values in the queues are invalid.

9.3.6 Timer Status Register (TSR)

The TSR registers are 8-bit readable/writable registers that indicate the status of each channel. The MTU2 has seven TSR registers, two for channel 0 and one each for channels 1 to 5.

- TSR_0, TSR_1, TSR_2, TSR_3, TSR_4

Bit:	7	6	5	4	3	2	1	0
	TCFD	-	TCFU	TCFV	TGFD	TGFC	TGFB	TGFA
Initial value:	1	1	0	0	0	0	0	0
R/W:	R	R	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1

Note: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

Bit	Bit Name	Initial Value	R/W	Description
7	TCFD	1	R	Count Direction Flag Status flag that shows the direction in which TCNT counts in channels 1 to 4. In channel 0, bit 7 is reserved. It is always read as 1 and the write value should always be 1. 0: TCNT counts down 1: TCNT counts up
6	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
5	TCFU	0	R/(W)*1	Underflow Flag Status flag that indicates that TCNT underflow has occurred when channels 1 and 2 are set to phase counting mode. Only 0 can be written, for flag clearing. In channels 0, 3, and 4, bit 5 is reserved. It is always read as 0 and the write value should always be 0. [Setting condition] • When the TCNT value underflows (changes from H'0000 to H'FFFF) [Clearing condition] • When 0 is written to TCFU after reading TCFU = 1*2

Bit	Bit Name	Initial Value	R/W	Description
0	ITB4VE	0*	R/W	TCIV_4 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4BN) with TCIV_4 interrupt skipping operation. 0: Does not link with TCIV_4 interrupt skipping 1: Links with TCIV_4 interrupt skipping

Notes: 1. TADCR must not be accessed in eight bits; it should always be accessed in 16 bits.
2. When interrupt skipping is disabled (the T3AEN and T4VEN bits in the timer interrupt skipping set register (TITCR) are cleared to 0 or the skipping count set bits (3ACOR and 4VCOR) in TITCR are cleared to 0), do not link A/D converter start requests with interrupt skipping operation (clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the timer A/D converter start request control register (TADCR) to 0).
3. If link with interrupt skipping is enabled while interrupt skipping is disabled, A/D converter start requests will not be issued.
* Do not set to 1 when complementary PWM mode is not selected.

Table 9.29 Setting of Transfer Timing by BF1 and BF0 Bits

Bit 7	Bit 6	
BF1	BF0	Description
0	0	Does not transfer data from the cycle set buffer register to the cycle set register.
0	1	Transfers data from the cycle set buffer register to the cycle set register at the crest of the TCNT_4 count.*[1]
1	0	Transfers data from the cycle set buffer register to the cycle set register at the trough of the TCNT_4 count.*[2]
1	1	Transfers data from the cycle set buffer register to the cycle set register at the crest and trough of the TCNT_4 count.*[2]

Notes: 1. Data is transferred from the cycle set buffer register to the cycle set register when the crest of the TCNT_4 count is reached in complementary PWM mode, when compare match occurs between TCNT_3 and TGRA_3 in reset-synchronized PWM mode, or when compare match occurs between TCNT_4 and TGRA_4 in PWM mode 1 or normal operation mode.
2. These settings are prohibited when complementary PWM mode is not selected.

Bit	Bit Name	Initial Value	R/W	Description
2	SYNC2	0	R/W	Timer Synchronous operation 2 to 0
1	SYNC1	0	R/W	These bits are used to select whether operation is independent of or synchronized with other channels.
0	SYNC0	0	R/W	When synchronous operation is selected, the TCNT synchronous presetting of multiple channels, and synchronous clearing by counter clearing on another channel, are possible. To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of bits CCLR0 to CCLR2 in TCR. 0: TCNT_2 to TCNT_0 operates independently (TCNT presetting /clearing is unrelated to other channels) 1: TCNT_2 to TCNT_0 performs synchronous operation TCNT synchronous presetting/synchronous clearing is possible

Table 9.45 shows the TICCR setting and input capture input pins.

Table 9.45 TICCR Setting and Input Capture Input Pins

Target Input Capture	TICCR Setting	Input Capture Input Pins
Input capture from TCNT_1 to TGRA_1	I2AE bit = 0 (initial value)	TIOC1A
	I2AE bit = 1	TIOC1A, TIOC2A
Input capture from TCNT_1 to TGRB_1	I2BE bit = 0 (initial value)	TIOC1B
	I2BE bit = 1	TIOC1B, TIOC2B
Input capture from TCNT_2 to TGRA_2	I1AE bit = 0 (initial value)	TIOC2A
	I1AE bit = 1	TIOC2A, TIOC1A
Input capture from TCNT_2 to TGRB_2	I1BE bit = 0 (initial value)	TIOC2B
	I1BE bit = 1	TIOC2B, TIOC1B

Example of Cascaded Operation Setting Procedure: Figure 9.20 shows an example of the setting procedure for cascaded operation.

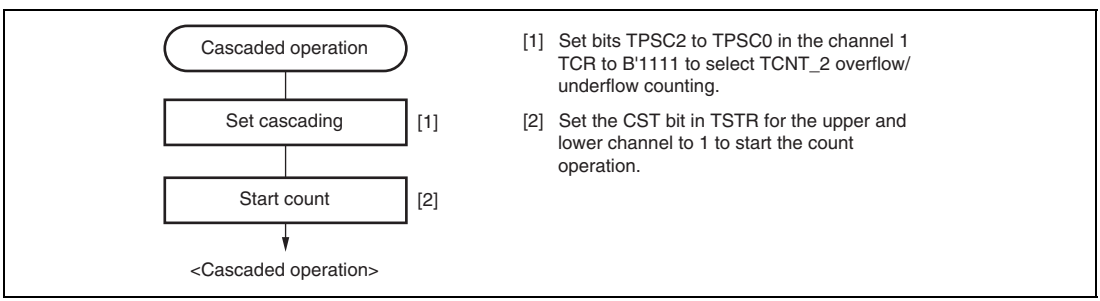

Figure 9.20 Cascaded Operation Setting Procedure

Cascaded Operation Example (a): Figure 9.21 illustrates the operation when TCNT_2 overflow/underflow counting has been set for TCNT_1 and phase counting mode has been designated for channel 2.

TCNT_1 is incremented by TCNT_2 overflow and decremented by TCNT_2 underflow.

13. Counter Clearing by Another Channel

In complementary PWM mode, by setting a mode for synchronization with another channel by means of the timer synchronous register (TSYR), and selecting synchronous clearing with bits CCLR2 to CCLR0 in the timer control register (TCR), it is possible to have TCNT_3, TCNT_4, and TCNTS cleared by another channel.

Figure 9.55 illustrates the operation.

Use of this function enables counter clearing and restarting to be performed by means of an external signal.

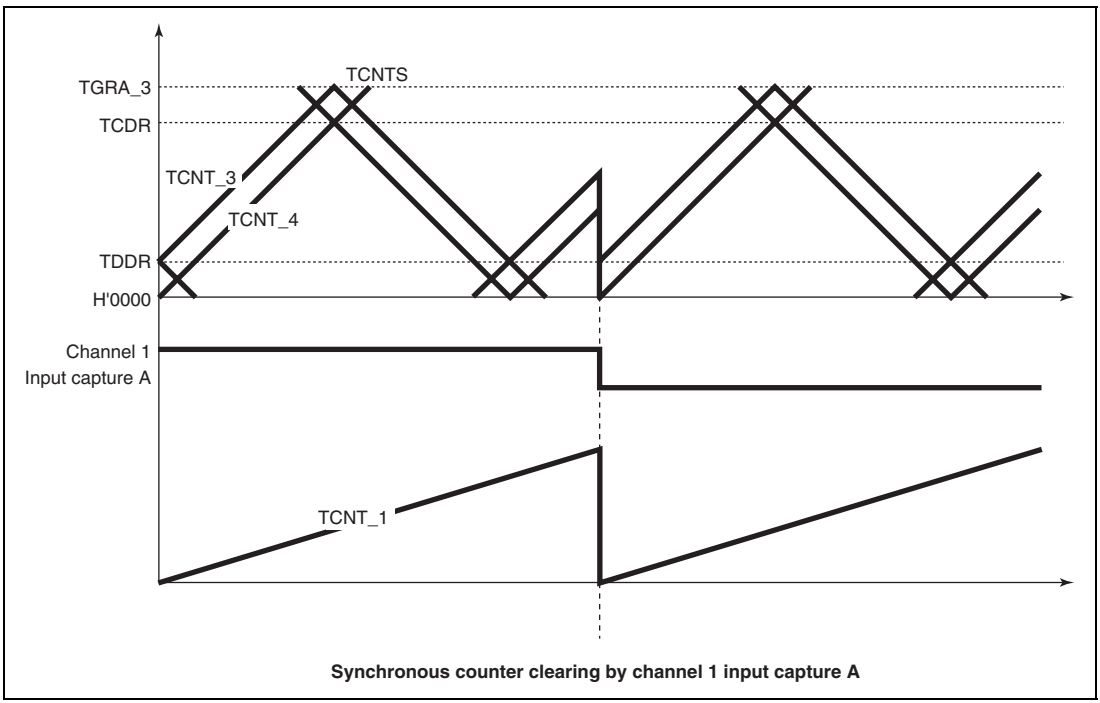

Figure 9.55 Counter Clearing Synchronized with Another Channel

Table 9.57 MTU2 Interrupts

Channel	Name	Interrupt Source	Interrupt Flag	Priority
0	TGIA_0	TGRA_0 input capture/compare match	TGFA_0	High
	TGIB_0	TGRB_0 input capture/compare match	TGFB_0	
	TGIC_0	TGRC_0 input capture/compare match	TGFC_0	
	TGID_0	TGRD_0 input capture/compare match	TGFD_0	
	TCIV_0	TCNT_0 overflow	TCFV_0	
	TGIE_0	TGRE_0 compare match	TGFE_0	
	TGIF_0	TGRF_0 compare match	TGFF_0	
1	TGIA_1	TGRA_1 input capture/compare match	TGFA_1	
	TGIB_1	TGRB_1 input capture/compare match	TGFB_1	
	TCIV_1	TCNT_1 overflow	TCFV_1	
	TCIU_1	TCNT_1 underflow	TCFU_1	
2	TGIA_2	TGRA_2 input capture/compare match	TGFA_2	
	TGIB_2	TGRB_2 input capture/compare match	TGFB_2	
	TCIV_2	TCNT_2 overflow	TCFV_2	
	TCIU_2	TCNT_2 underflow	TCFU_2	
3	TGIA_3	TGRA_3 input capture/compare match	TGFA_3	
	TGIB_3	TGRB_3 input capture/compare match	TGFB_3	
	TGIC_3	TGRC_3 input capture/compare match	TGFC_3	
	TGID_3	TGRD_3 input capture/compare match	TGFD_3	
	TCIV_3	TCNT_3 overflow	TCFV_3	
4	TGIA_4	TGRA_4 input capture/compare match	TGFA_4	
	TGIB_4	TGRB_4 input capture/compare match	TGFB_4	
	TGIC_4	TGRC_4 input capture/compare match	TGFC_4	
	TGID_4	TGRD_4 input capture/compare match	TGFD_4	
	TCIV_4	TCNT_4 overflow/underflow	TCFV_4	
5	TGIU_5	TGRU_5 input capture/compare match	TGFU_5	
	TGIV_5	TGRV_5 input capture/compare match	TGFV_5	
	TGIW_5	TGRW_5 input capture/compare match	TGFW_5	Low

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Reset-Synchronized PWM Mode: Figure 9.135 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in reset-synchronized PWM mode after re-setting.

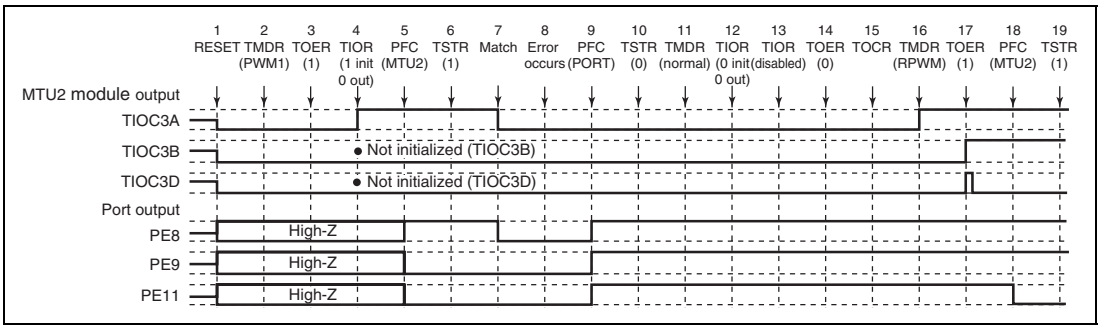

Figure 9.135 Error Occurrence in PWM Mode 1, Recovery in Reset-Synchronized PWM Mode

1 to 14 are the same as in figure 9.134.

15. Select the reset-synchronized PWM output level and cyclic output enabling/disabling with TOCR.
16. Set reset-synchronized PWM.
17. Enable channel 3 and 4 output with TOER.
18. Set MTU2 output with the PFC.
19. Operation is restarted by TSTR.

10.5 Interrupts

The POE issues a request to generate an interrupt when the specified condition is satisfied during input level detection or output level comparison. Table 10.5 shows the interrupt sources and their conditions.

Table 10.5 Interrupt Sources and Conditions

Name	Interrupt Source	Interrupt Flag	Condition
OEI1	Output enable interrupt 1	POE3F, POE1F, POE0F, and OSF1	PIE1 • (POE3F + POE1F + POE0F) + OIE1 • OSF1
OEI3	Output enable interrupt 3	POE8F	PIE3 • POE8F

11.4 Operation

11.4.1 Canceling Software Standbys

The WDT can be used to revoke software standby mode with an NMI interrupt or external interrupt (IRQ). The procedure is described below. (The WDT does not run when resets are used for canceling, so keep the $\overline{\text{RES}}$ pin low until the clock stabilizes.)

1. Before transition to software standby mode, always clear the TME bit in WTCSR to 0. When the TME bit is 1, an erroneous reset or interval timer interrupt may be generated when the count overflows.
2. Set the type of count clock used in the CKS2 to CKS0 bits in WTCSR and the initial values for the counter in the WTCNT counter. These values should ensure that the time till count overflow is longer than the clock oscillation settling time.
3. Transition to software standby mode by executing a SLEEP instruction to stop the clock.
4. The WDT starts counting by detecting a change in the level input to the NMI or IRQ pin.
5. When the WDT count overflows, the CPG starts supplying the clock and the LSI resumes operation. The WOVF flag in WTCSR is not set when this happens.

11.4.2 Using Watchdog Timer Mode

While operating in watchdog timer mode, the WDT generates an internal reset of the type specified by the RSTS bit in WTCSR and asserts a signal through the $\overline{\text{WDTOVF}}$ pin every time the counter overflows.

1. Set the WT/IT bit in WTCSR to 1, set the reset type in the RSTS bit, set the type of count clock in the CKS2 to CKS0 bits, and set the initial value of the counter in the WTCNT counter.
2. Set the TME bit in WTCSR to 1 to start the count in watchdog timer mode.
3. While operating in watchdog timer mode, rewrite the counter periodically to prevent the counter from overflowing.
4. When the counter overflows, the WDT sets the WOVF flag in WTCSR to 1, asserts a signal through the $\overline{\text{WDTOVF}}$ pin for one cycle of the count clock specified by the CKS2 to CKS0 bits, and generates a reset of the type specified by the RSTS bit. The counter then resumes counting.

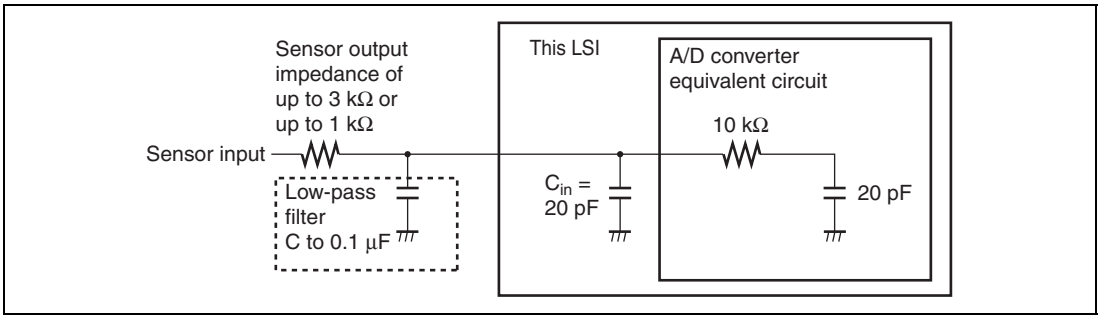

Figure 13.7 Example of Analog Input Circuit

13.7.4 Range of Analog Power Supply and Other Pin Settings

If the conditions below are not met, the reliability of the device may be adversely affected.

- Analog input voltage range

  The voltage applied to analog input pin ANn during A/D conversion should be in the range AVss ≤ VAN ≤ AVref.
- Relationship between AVcc, AVss and Vcc, Vss

  Set Vcc ≤ AVcc ≤ 5.5V, AVss = Vss for the relationship between AVcc, AVss and Vcc, Vss. If the A/D converter is not used, the AVcc and AVss pins must not be left open.

13.7.5 Notes on Board Design

In board design, digital circuitry and analog circuitry should be as mutually isolated as possible, and layout in which digital circuit signal lines and analog circuit signal lines cross or are in close proximity should be avoided as far as possible. Failure to do so may result in incorrect operation of the analog circuitry due to inductance, adversely affecting A/D conversion values. Also, digital circuitry must be isolated from the analog input signals (AN0 to AN7), and analog power supply (AVcc) by the analog ground (AVss). Also, the analog ground (AVss) should be connected at one point to a stable ground (Vss) on the board.

- Port A Control Register L2 (PACRL2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PA7 MD2	PA7 MD1	PA7 MD0	-	PA6 MD2	PA6 MD1	PA6 MD0	-	PA5 MD2	PA5 MD1	PA5 MD0	-	PA4 MD2	PA4 MD1	PA4 MD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PA7MD2	0	R/W	PA7 Mode
13	PA7MD1	0	R/W	Select the function of the PA7/TCLKB/SCK2/TCK pin. When the E10A* is in use ($\overline{\text{ASEMD0}}$ = low), function is fixed to TCK input.
12	PA7MD0	0	R/W	000: PA7 I/O (port) 001: TCLKB input (MTU2) 110: SCK2 I/O (SCI) Other than above: Setting prohibited
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PA6MD2	0	R/W	PA6 Mode
9	PA6MD1	0	R/W	Select the function of the PA6/TCLKA pin.
8	PA6MD0	0	R/W	000: PA6 I/O (port) 001: TCLKA input (MTU2) Other than above: Setting prohibited
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PA5MD2	0	R/W	PA5 Mode
5	PA5MD1	0	R/W	Select the function of the PA5/IRQ3/SCK1 pin.
4	PA5MD0	0	R/W	000: PA5 I/O (port) 001: SCK1 I/O (SCI) 111: IRQ3 input (INTC) Other than above: Setting prohibited

17.2.6 Programming/Erasing Interface

Programming/erasing is executed by downloading the on-chip program to the on-chip RAM and specifying the program address/data and erase block by using the interface registers/parameters.

The procedure program is made by the user in user program mode. The overview of the procedure is as follows. For details, see section 17.5.2, User Program Mode (Only in On-Chip 128-Kbyte and 64-Kbyte ROM Version).

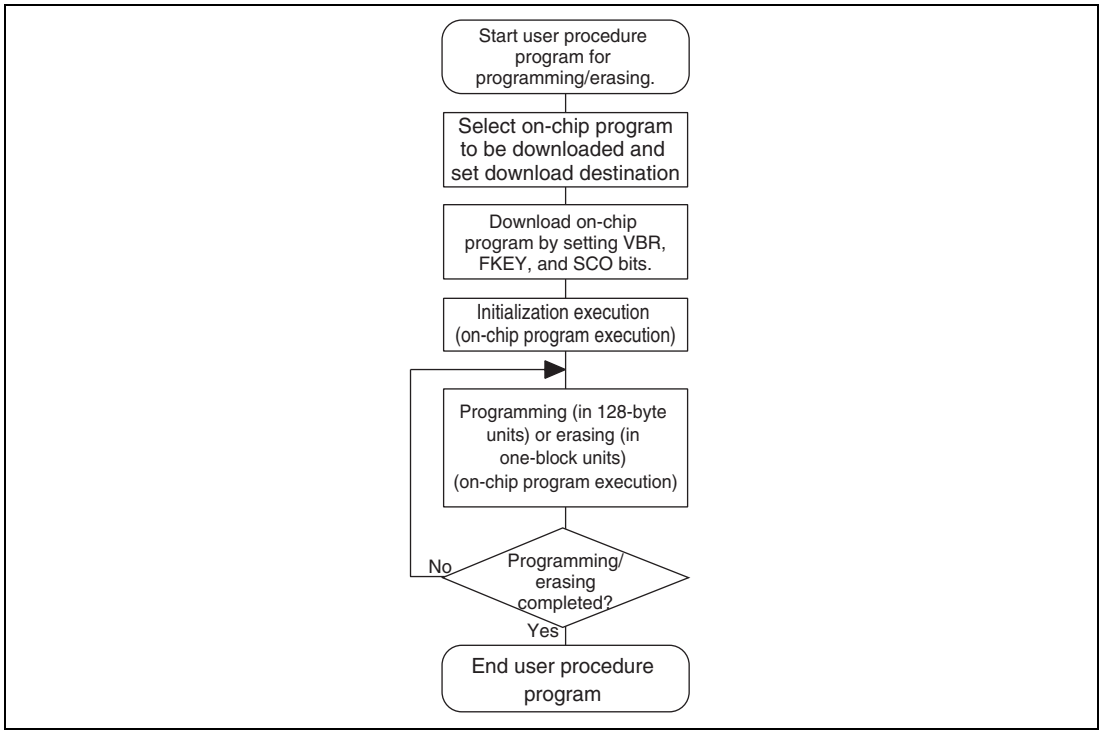

Figure 17.5 Overview of User Procedure Program

(1) Selection of On-Chip Program to be Downloaded and Setting of Download Destination

This LSI has programming/erasing programs and they can be downloaded to the on-chip RAM. The on-chip program to be downloaded is selected by setting the corresponding bits in the programming/erasing interface registers. The download destination can be specified by FTDAR.

table, interrupt processing routine, and user branch program should be transferred to on-chip RAM before programming/erasing of the flash memory starts.

5. The flash memory is not accessible during programming/erasing operations. Therefore, the programming/erasing program must be downloaded to on-chip RAM in advance. Areas for executing each procedure program for initiating programming/erasing, the user program at the user branch destination for programming/erasing, the interrupt vector table, and the interrupt processing routine must be located in on-chip RAM.
6. After programming/erasing, access to flash memory is inhibited until FKEY is cleared.

   A reset state ($\overline{RES} = 0$) for more than at least 100 μs must be taken when the LSI mode is changed to reset on completion of a programming/erasing operation.

   Transitions to the reset state during programming/erasing are inhibited. When the reset signal is accidentally input to the LSI, a longer period in the reset state than usual (100 μs) is needed before the reset signal is released.
7. When the program data storage area indicated by the FMPDR parameter in the programming processing is within the flash memory area, an error will occur. Therefore, temporarily transfer the program data to on-chip RAM to change the address set in FMPDR to an address other than flash memory.

Tables 17.16 and 17.17 show the areas in which the program data can be stored and executed according to the operation type and mode.

Table 17.16 Executable MAT

	Initiated Mode
Operation	**User Program Mode**
Programming	Table 17.17 (1)
Erasing	Table 17.17 (2)

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Output high voltage	TIOC3B, TIOC3D, TIOC4A to TIOC4D	V_{OH}	V_{CC} – 0.8	—	—	V	I_{OH} = –5 mA, V_{CC} = 4.5 V to 5.5 V
	$\overline{WDTOVF}$		V_{CC} – 0.5	—	—	V	I_{OH} = –100 µA
	All other output pins		V_{CC} – 0.5	—	—	V	I_{OH} = –200 µA
			V_{CC} – 1.0	—	—	V	I_{OH} = –1 mA
			V_{CC} – 1.5	—	—	V	I_{OH} = –2 mA (reference values)
Output low voltage	TIOC3B, TIOC3D, TIOC4A to TIOC4D	V_{OL}	—	—	1.0	V	I_{OL} = 15 mA, V_{CC} = 4.5 V to 5.5 V
			—	—	0.6	V	I_{OL} = 10 mA, V_{CC} = 4.5 V to 5.5 V
			—	—	0.44	V	I_{OL} = 8 mA, V_{CC} = 4.5 V to 5.5 V
	All other output pins		—	—	0.4	V	I_{OL} = 1.6 mA
Input capacitance	All input pins	C_{in}	—	—	20	pF	V_{in} = 0 V f = 1 MHz Ta = 25°C
Supply current	Normal operation	I_{CC}	—	52	70	mA	Iϕ = 50 MHz (SH7125, SH7124)
			—	35	50	mA	Iϕ = 50 MHz (SH71251A, SH71241A, SH71250A, SH71240A)
	Sleep		—	33	50	mA	Iϕ = 50 MHz (SH7125, SH7124)
			—	22	30	mA	Iϕ = 50 MHz (SH71251A, SH71241A, SH71250A, SH71240A)
	Software standby		—	—	5	mA	$T_a \leq 50°C$
			—	—	15	mA	$50°C < T_a$
Analog power supply current	During A/D conversion	AI_{CC}	—	3	5	mA	The value per module
	Waiting for A/D conversion		—	—	2	mA	The value per module
	Standby		—	—	15	µA	

[Operating Precautions]

1. When the A/D converter is not used, do not leave the AV_{CC} and AV_{SS} pins open.
2. The supply current was measured when V_{IH} (Min.) = V_{CC} - 0.5 V, V_{IL} (Max.) = 0.5 V, with all output pins unloaded.