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	I²C, SCI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	47
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 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/df36074gfzv

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins

   Handle unused pins in accord with the directions given under Handling of Unused Pins in the manual.

   — The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

   The state of the product is undefined at the moment when power is supplied.

   — The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.

   In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.

   In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

   Access to reserved addresses is prohibited.

   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.

   — When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.

5. Differences between Products

   Before changing from one product to another, i.e. to one with a different part number, confirm that the change will not lead to problems.

   — The characteristics of MPU/MCU in the same group but having different part numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different part numbers, implement a system-evaluation test for each of the products.

5.6 Notes on System Control

5.6.1 Option Function Select Area Setting Example

The option function select area is allocated in the flash memory, not in the SFRs. Set appropriate values as ROM data by a program. The following shows a setting example.

• To set FFh in the OFS2 register

```
.org 00FFDBH
.byte 0FFh
```

Programming formats vary depending on the compiler. Check the compiler manual.

• To set FFh in the OFS register

```
.org 00FFFCH
.lword reset | (0FF000000h)          ; RESET
```

Programming formats vary depending on the compiler. Check the compiler manual.

9. Clock Generation Circuit

9.1 Overview

The following three circuits are included in the clock generation circuit:
• XIN clock oscillation circuit
• High-speed on-chip oscillator
• Low-speed on-chip oscillator
Table 9.1 lists the Clock Generation Circuit Specifications. Figure 9.1 shows the Clock Generation Circuit Block Diagram. Figure 9.2 shows the Supply of Peripheral Function Clocks. Table 9.2 lists the Clock Generation Circuit Pin Configuration.

Table 9.1 Clock Generation Circuit Specifications

Item	XIN Clock Oscillation Circuit	High-Speed On-Chip Oscillator	Low-Speed On-Chip Oscillator
Clock frequency	0 MHz to 20 MHz (2 MHz to 20 MHz when an oscillator is used)	Approx. 20 MHz	Approx. 125 kHz
Connectable oscillator	• Ceramic resonator • Crystal oscillator	—	—
Oscillator connect pins	XIN, XOUT (1)	—	—
Oscillation start and stop	Usable	Usable	Usable
State after reset	Stopped	Stopped	Oscillates
Others	• An externally generated clock can be input. • A feed-back resistor is included (connected or not connected can be selected).	The system clock can be output from P4_7.	The system clock can be output from P4_7.

Note:
1. When the on-chip oscillator clock instead of the XIN clock oscillation circuit is used as the CPU clock, these pins can be used as P4_6 and P4_7.

12.3 Port 1

Figure 12.1 shows the Port 1 Pin Configuration.

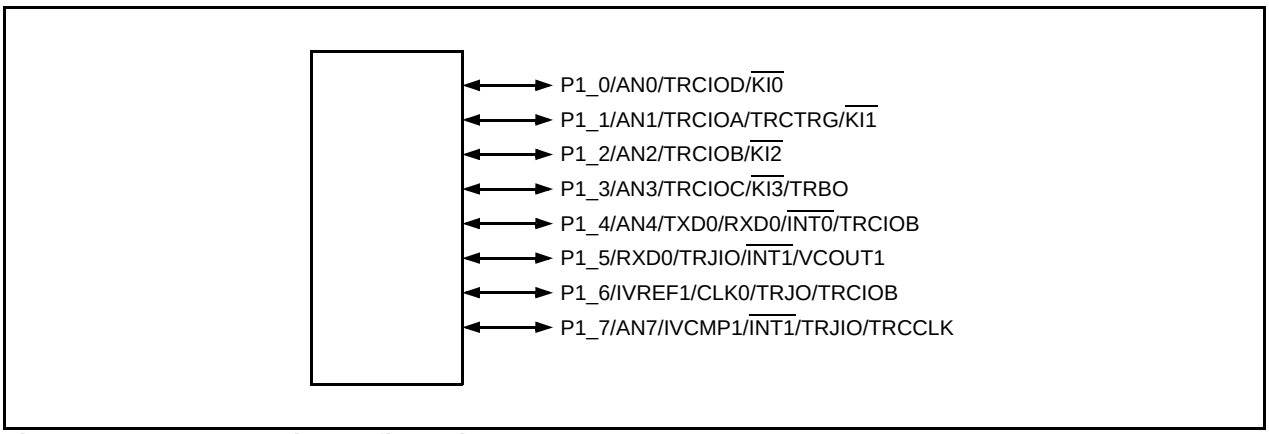


Figure 12.1 Port 1 Pin Configuration

12.4.8 Pin Settings for Port 3

Tables 12.12 to 12.15 list the pin settings for port 3.

Table 12.12 Port P3_3/IVCMP3/TRCCLK/$\overline{\text{INT3}}$

Register	PD3	PMH3		Function
Bit	PD3_3	P33SEL		
		1	0	
Setting value	0	0	0	Input port/IVCMP3
	1	0	0	Output port
	X	0	1	TRCCLK input
	X	1	0	$\overline{\text{INT3}}$ input

X: 0 or 1

Table 12.13 Port P3_4/IVREF3/TRCIOC/$\overline{\text{INT2}}$

Register	PD3	PMH3		Timer RC Setting	Function
Bit	PD3_4	P34SEL			
		1	0		
Setting value	0	0	0	X	Input port/IVREF3
	1	0	0	X	Output port
	X	0	1	See **Table 12.23 TRCIOC Pin Settings**.	TRCIOC input
	X	0	1	See **Table 12.23 TRCIOC Pin Settings**.	TRCIOC output
	X	1	0	X	$\overline{\text{INT2}}$ input

X: 0 or 1

Table 12.14 Port P3_5/TRCIOD/$\overline{\text{KI2}}$/VCOUT3

Register	PD3	PMH3		Timer RC Setting	Function
Bit	PD3_5	P35SEL			
		1	0		
Setting value	0	0	0	X	Input port
	1	0	0	X	Output port
	X	0	1	See **Table 12.24 TRCIOD Pin Settings**.	TRCIOD input
	X	0	1	See **Table 12.24 TRCIOD Pin Settings**.	TRCIOD output
	X	1	0	X	$\overline{\text{KI2}}$ input
	X	1	1	X	VCOUT3 output

X: 0 or 1

Table 12.15 Port P3_7/$\overline{\text{ADTRG}}$/TRJO/TRCIOD

Register	PD3	PMH3		Timer RC Setting	Function
Bit	PD3_7	P37SEL			
		1	0		
Setting value	0	0	0	X	Input port
	1	0	0	X	Output port
	X	0	1	X	$\overline{\text{ADTRG}}$ input
	X	1	0	X	TRJO output
	X	1	1	See **Table 12.24 TRCIOD Pin Settings**.	TRCIOD input
	X	1	1	See **Table 12.24 TRCIOD Pin Settings**.	TRCIOD output

X: 0 or 1

12.8 Pin Settings for Peripheral Function I/O

Tables 12.21 to 12.24 list the pin settings for peripheral function I/O.

Table 12.21 TRCIOA Pin Settings

Register	TRCOER	TRCMR	TRCIOR0			TRCCR2		Function
Bit	EA	PWM2	IOA2	IOA1	IOA0	TCEG1	TCEG0	
Setting value	0	1	0	0	1	X	X	Timer mode waveform output (output compare function)
				1	X			
	0	1	1	X	X	X	X	Timer mode (input capture function)
	1							
	1	0	X	X	X	0	1	PWM2 mode (TRCTRG input)
						1	X	

X: 0 or 1

Table 12.22 TRCIOB Pin Settings

Register	TRCOER	TRCMR		TRCIOR0			Function
Bit	EB	PWM2	PWMB	IOB2	IOB1	IOB0	
Setting value	0	0	X	X	X	X	PWM2 mode waveform output
	0	1	1	X	X	X	PWM mode waveform output
	0	1	0	0	0	1	Timer mode waveform output (output compare function)
					1	X	
	0	1	0	1	X	X	Timer mode (input capture function)
	1						

X: 0 or 1

Table 12.23 TRCIOC Pin Settings

Register	TRCOER	TRCMR		TRCIOR1			Function
Bit	EC	PWM2	PWMC	IOC2	IOC1	IOC0	
Setting value	0	1	1	X	X	X	PWM mode waveform output
	0	1	0	0	0	1	Timer mode waveform output (output compare function)
					1	X	
	0	1	0	1	X	X	Timer mode (input capture function)
	1						

X: 0 or 1

Table 12.24 TRCIOD Pin Settings

Register	TRCOER	TRCMR		TRCIOR1			Function
Bit	ED	PWM2	PWMD	IOD2	IOD1	IOD0	
Setting value	0	1	1	X	X	X	PWM mode waveform output
	0	1	0	0	0	1	Timer mode waveform output (output compare function)
					1	X	
	0	1	0	1	X	X	Timer mode (input capture function)
	1						

X: 0 or 1

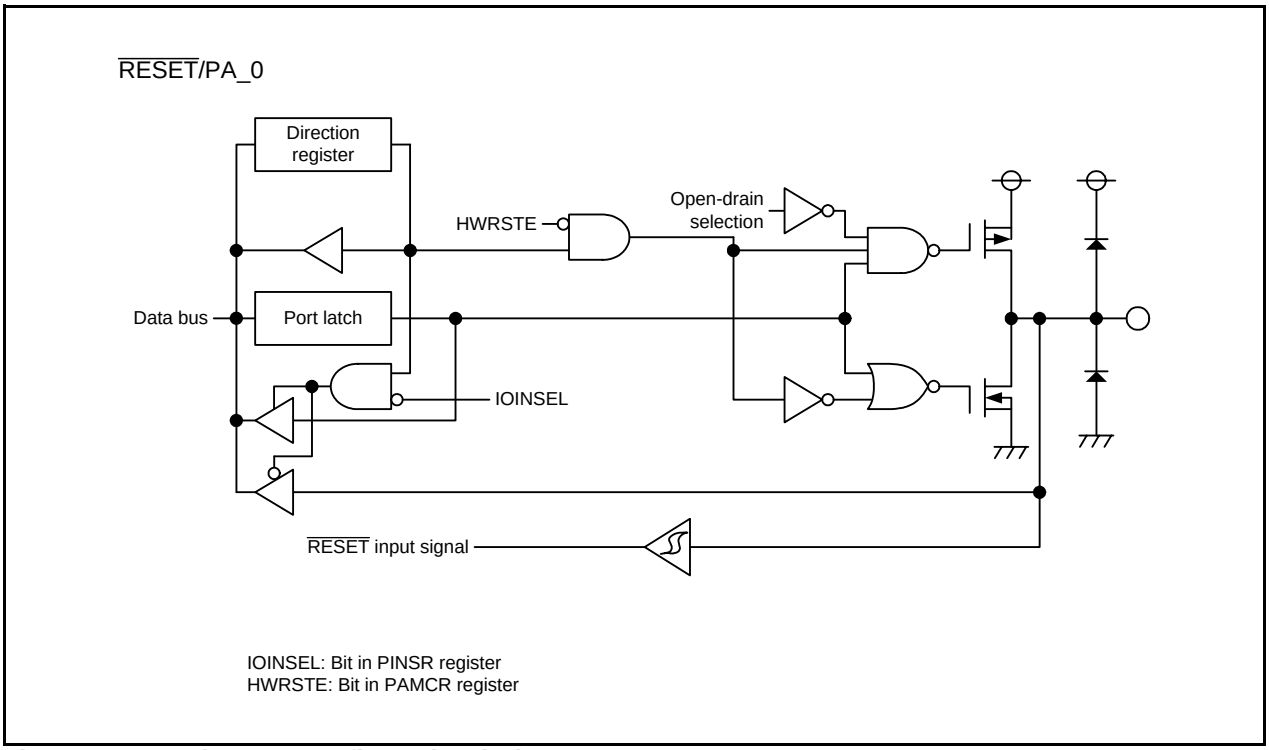


Figure 12.18 I/O Port Configuration (13)

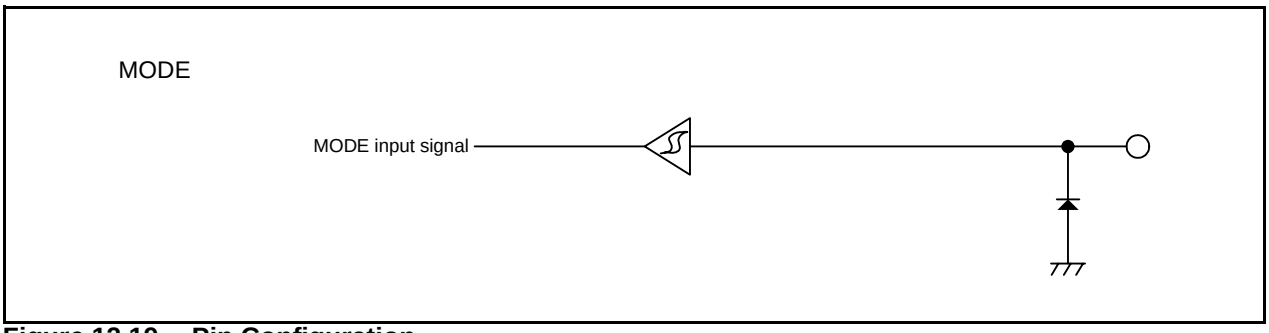


Figure 12.19 Pin Configuration

14.2 I/O Pins

Table 14.2 lists the Timer RB2 Pin Configuration.

Table 14.2 Timer RB2 Pin Configuration

Pin Name	Assigned Pin	I/O	Function
$\overline{\text{INT0}}$	P1_4, P4_5	I	External trigger
TRBO	P1_3, P4_2	O	Continuous pulse or one-shot pulse output

For details on $\overline{\text{INT0}}$, see **11. Interrupts**.

15.2.2 Timer RC General Register A, B, C, and D (TRCGRA, TRCGRB, TRCGRC, and TRCGRD)

Address 000EAh to 000EBh (TRCGRA), 000ECh to 000EDh (TRCGRB), 000EEh to 000EFh (TRCGRC), 000F0h to 000F1h (TRCGRD)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Symbol	—	—	—	—	—	—	—	—
After Reset	1	1	1	1	1	1	1	1

Bit	b15	b14	b13	b12	b11	b10	b9	b8
Symbol	—	—	—	—	—	—	—	—
After Reset	1	1	1	1	1	1	1	1

Table 15.5 Functions of TRCGRj Register when Using Input Capture Function

Register	Setting	Register Function	Input Capture Input Pin
TRCGRA	—	General register. Can be used to read the TRC register value at input capture.	TRCIOA
TRCGRB			TRCIOB
TRCGRC	BUFEA = 0	General register. Can be used to read the TRC register value at input capture.	TRCIOC
TRCGRD	BUFEB = 0		TRCIOD
TRCGRC	BUFEA = 1	Buffer registers. Can be used to hold transferred value from the general register. (Refer to **15.5.5 Buffer Operation Timing**.)	TRCIOA
TRCGRD	BUFEB = 1		TRCIOB

j = A, B, C, or D
BUFEA, BUFEB: Bits in TRCMR register

Table 15.6 Functions of TRCGRj Register when Using Output Compare Function

Register	Setting	Register Function	Output Compare Output Pin
TRCGRA	—	General register. Write a compare value to one of these registers.	TRCIOA
TRCGRB			TRCIOB
TRCGRC	BUFEA = 0	General register. Write a compare value to one of these registers.	TRCIOC
TRCGRD	BUFEB = 0		TRCIOD
TRCGRC	BUFEA = 1	Buffer register. Write the next compare value to one of these registers. (Refer to **15.5.5 Buffer Operation Timing**.)	TRCIOA
TRCGRD	BUFEB = 1		TRCIOB

j = A, B, C, or D
BUFEA, BUFEB: Bits in TRCMR register

Figure 15.8 shows an Example of Buffer Operation during Input Capture. This example applies when the TRCGRA register is set as an input capture register and the TRCGRC register is set as a buffer register for the TRCGRA register. In this example, the TRCCNT register is used for the free-running count operation and both rising and falling edges are selected for the input capture input to the TRCIOA pin. Since buffer operation is set, the value in the TRCCNT register is stored in the TRCGRA register by input capture A and the value that has been stored in the TRCGRA register is transferred to the TRCGRC register at the same time.

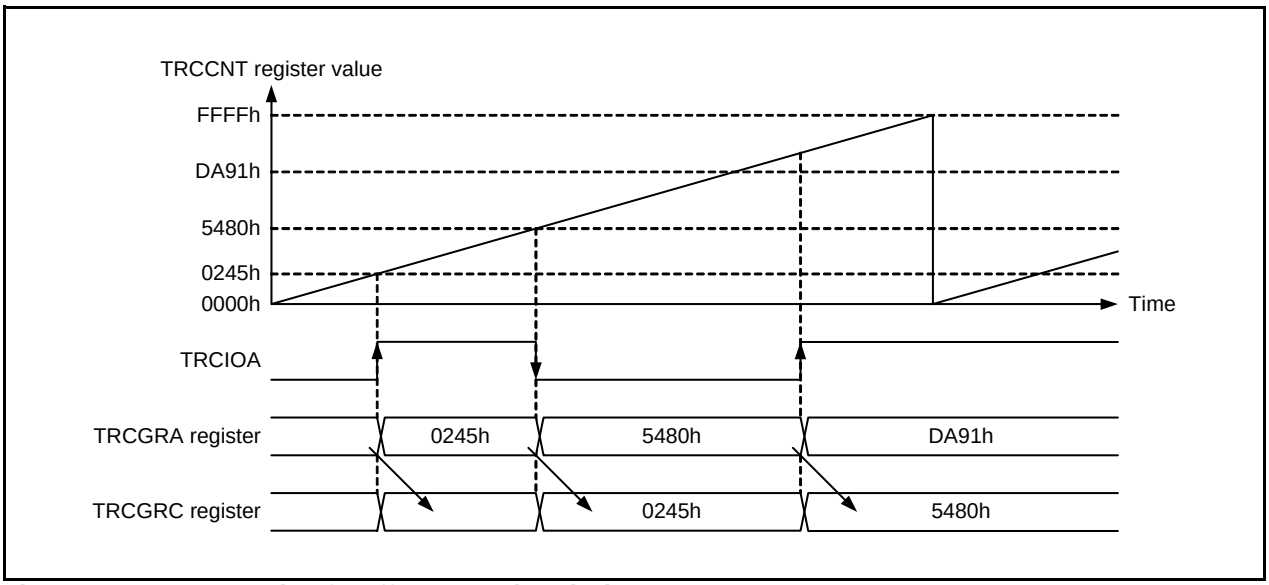


Figure 15.8 Example of Buffer Operation during Input Capture

15.7.5 Input Capture Function

- Set the pulse width of the input capture signal as follows:
  [When the digital filter is not used]
  Three or more cycles of the timer RC operation clock (refer to **Table 15.1 Timer RC Specifications**)
  [When the digital filter is used]
  Five cycles of the digital filter sampling clock + three cycles of the timer RC operating clock, minimum (refer to **Figure 15.19 Digital Filter Circuit Block Diagram**)

- The value of the TRCCNT register is transferred to the TRCGRj register one or two cycles of the timer RC operation clock after the input capture signal is input to the TRCIOj (j = A, B, C, or D) pin (when the digital filter function is not used).

15.7.6 TRCMR Register in PWM2 Mode

When the CSTP bit in the TRCCR2 register is 1 (increment is stopped), do not set the TRCMR register when a compare match occurs between registers TRCCNT and TRCGRA.

15.7.7 MSTCR Register

After stopping the timer RC count, set the MSTTRC bit in the MSTCR register to 1 (standby).

15.7.8 Mode Switching

- When switching the modes during operation, set the CTS bit in the TRCMR register to 0 (count is stopped) before switching.
- After switching the modes, set each flag in the TRCSR register to 0 before operation is started.

15.7.9 Procedure for Setting Registers Associated with Timer RC

Set the registers associated with timer RC following the procedure below:

(1) Set timer RC operating mode (bits PWMB, PWMC, PWMD, and PWM2 in the TRCMR register).
(2) Set the registers other than that set in (1).
(3) Set the port output to be enabled (bits EA to ED in the TRCOER register).

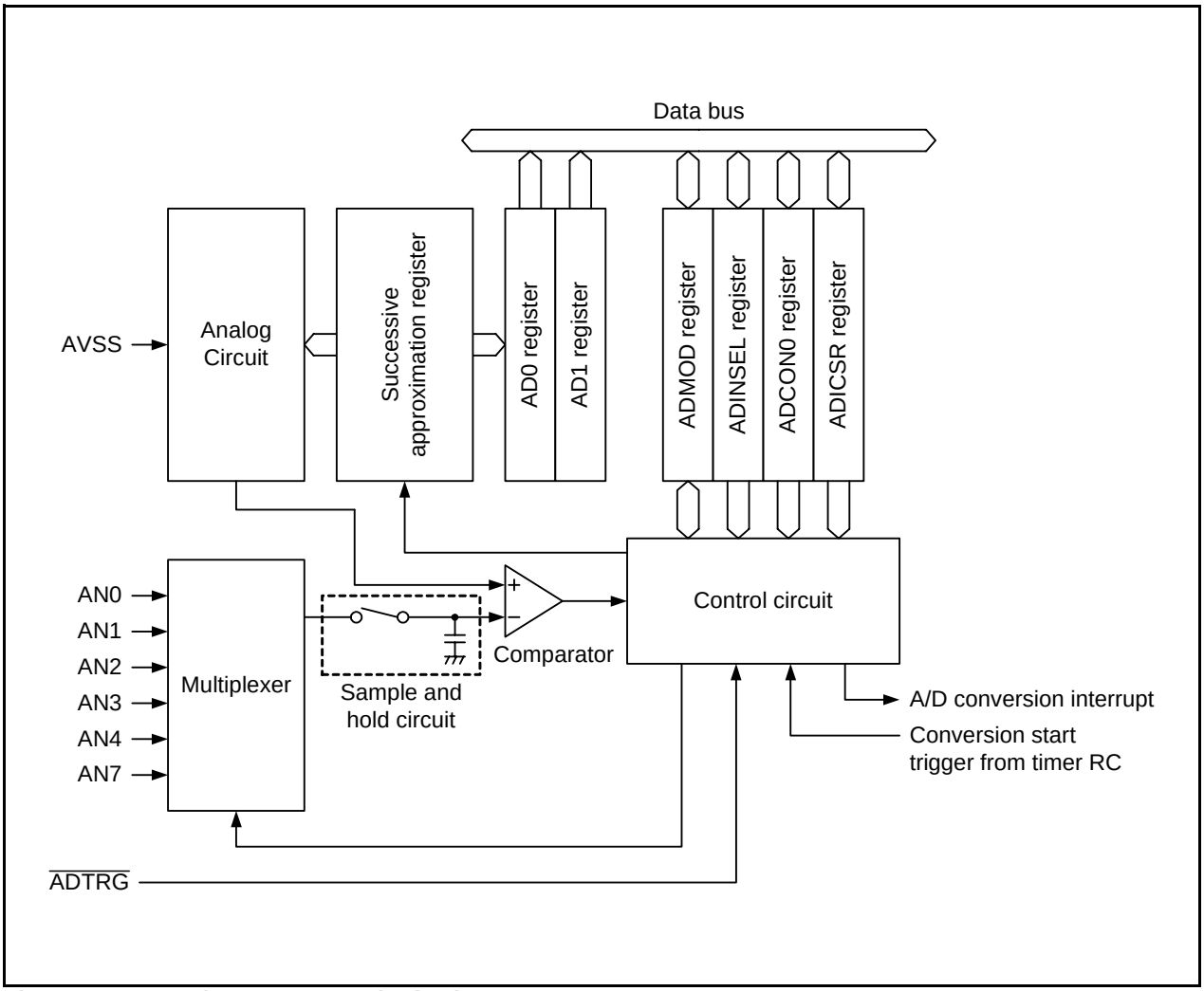


Figure 17.1 A/D Converter Block Diagram

Table 17.2 lists the A/D Converter Pin Configuration.
Pins AVCC and AVSS are used for the power supply to the analog block in the A/D converter.
The six analog input pins are divided into three channel groups.

Table 17.2 A/D Converter Pin Configuration

Pin Name	Assigned Pin	I/O	Function
AVCC	VCC	I	Power supply input for the A/D converter
AVSS	VSS	I	
AN0	P1_0	I	Analog input for channel group 0
AN1	P1_1	I	
AN2	P1_2	I	Analog input for channel group 1
AN3	P1_3	I	
AN4	P1_4	I	Analog input for channel group 2
AN7	P1_7	I	
$\overline{\text{ADTRG}}$	P3_7	I	External trigger input for starting A/D conversion

17.3.5 Repeat Sweep Mode

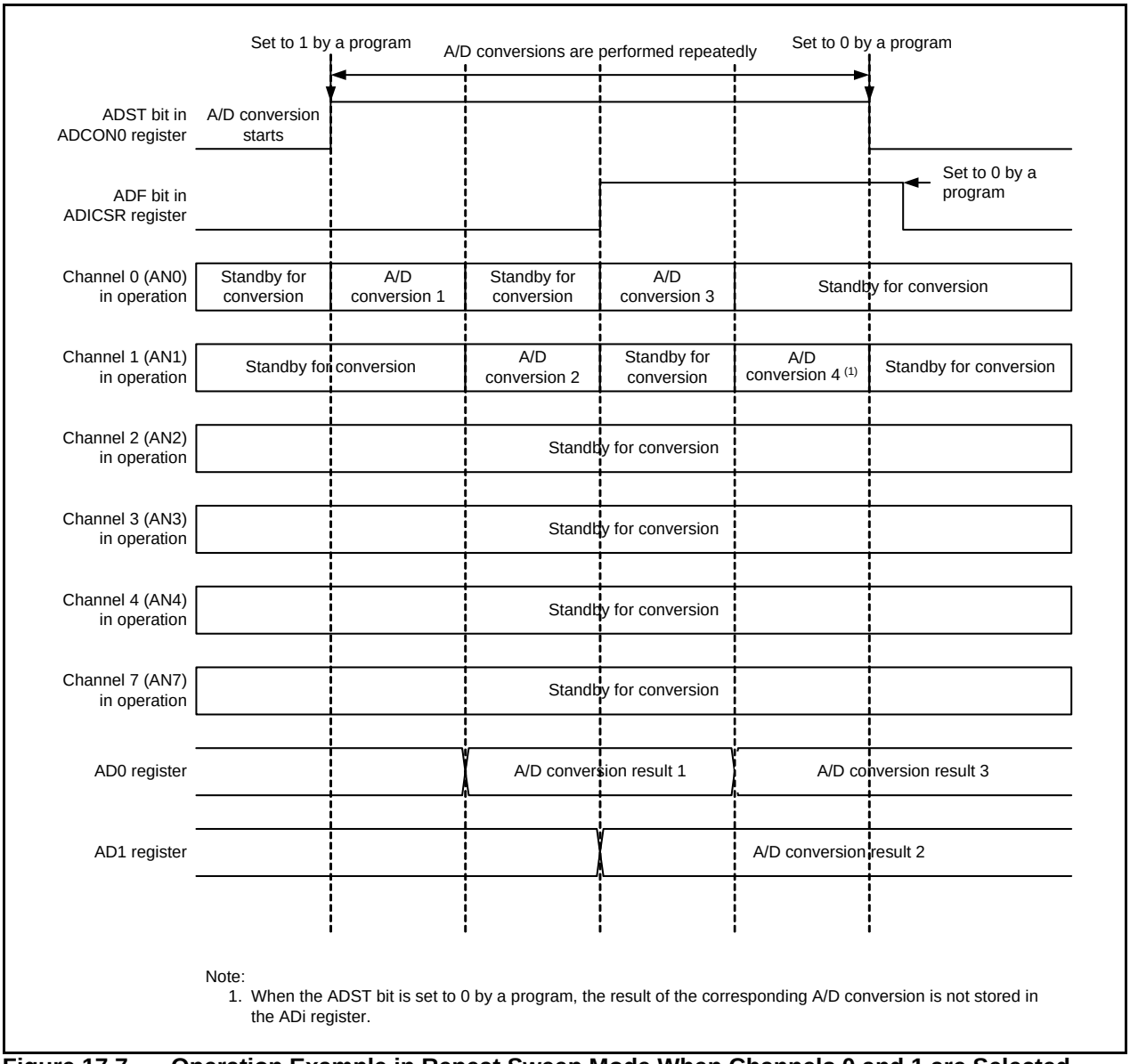
Figure 17.7 shows an Operation Example in Repeat Sweep Mode When Channels 0 and 1 are Selected.
In repeat sweep mode, A/D conversions of the analog inputs are performed for the specified two channels repeatedly as follows:

(1) When the ADST bit in the ADCON0 register is set to 1 (A/D conversion starts) by software trigger, timer RC trigger, or external trigger input, A/D conversion is started from AN0 when channel group 0 is selected, and AN2 when channel group 1 is selected. When channel group 2 is selected, A/D conversion is started from AN4.

(2) When A/D conversion has completed on each channel, the result is transferred to the corresponding ADi register (i = 0 or 1).

(3) When A/D conversion has completed on all the selected channels, the ADF bit in the ADICSR register is set to 1 (interrupt requested).
Writing 0 after reading the value 1 sets the ADF bit to 0 (no interrupt requested).

(4) While the ADST bit is 1 (A/D conversion starts), steps between (2) and (3) are repeated. When the ADST bit is set to 0 (A/D conversion stops), A/D conversion is stopped and the A/D converter enters the standby state. Then, when the ADST bit is set to 1, A/D conversion is restarted from AN0 when channel group 0 is selected, and AN2 when channel group 1 is selected. When channel group 2 is selected, A/D conversion is started from AN4.



Note:
1. When the ADST bit is set to 0 by a program, the result of the corresponding A/D conversion is not stored in the ADi register.

Figure 17.7 Operation Example in Repeat Sweep Mode When Channels 0 and 1 are Selected

19.8.2 CPU Rewrite Mode

19.8.2.1 Prohibited Instructions

The following instructions cannot be used while the program ROM area is being rewritten in EW0 mode because they reference data in the flash memory:
UND, INTO, and BRK

19.8.2.2 Interrupts

Tables 19.12 and 19.13 list the Interrupt Handling during CPU Rewrite Operation.

Table 19.12 Interrupt Handling during CPU Rewrite Operation (EW0 Mode)

Interrupt Type	Data Flash/Program ROM	
	Suspend Enabled (FMR20 = 1)	Suspend Disabled (FMR20 = 0)
Maskable interrupt	When an interrupt request is acknowledged, interrupt handling is executed. (The interrupt vector is allocated in the RAM) The suspend state can be entered by either of the following: (1) When the FMR22 bit is 1 (suspend request enabled by interrupt request), the FMR21 bit is automatically set to 1 (suspend request). The flash memory suspends auto-erase or auto-programming after td(SR-SUS). (2) When the FMR22 bit is 0 (suspend request disabled by interrupt request) and suspend is required, set the FMR21 bit to 1 (suspend request) in the interrupt handling. The flash memory suspends auto-erase or auto-programming after td(SR-SUS). While auto-erase is suspended, auto-programming and reading can be executed for any block other than the blocks being auto-erased. While auto-programming is suspended, any block other than the blocks being auto-programmed can be read. Auto-erase can be restarted by setting the FMR21 bit to 0 (restart).	Interrupt handling is executed with auto-erase or auto-programming executed (The interrupt vector is allocated in the RAM)
Address match	Do not use during auto-erasing or auto-programming.	
UND, INTO, and BRK instructions		
Single-step		
Watchdog timer	When an interrupt request is acknowledged, auto-erase or auto-programming is forcibly stopped immediately and the flash memory is reset. After the specified period, the flash memory is restarted before interrupt handling is started. Since auto-erase or auto-programming is forcibly stopped, the correct values may not be read from the block being auto-erased or the address being auto-programmed. After the flash memory is restarted, execute auto-erase again and verify it complete normally. The watchdog timer does not stop while the command is executing, so interrupt requests may be generated. Initialize the watchdog timer periodically using the erase-suspend function. Since the flash memory control registers are initialized in this case, these registers must be set again. (1)	
Oscillation stop detection		
Voltage monitor 1		

FMR20, FMR21, FMR22: Bits in FMR2 register
Note:
1. Registers FMR0, FMR1, and FMR2 are initialized if a watchdog timer, oscillation stop detection, or voltage monitor 1 interrupt is generated while the flash memory is busy.
When the FMR01 bit in the FMR0 register is 1 (CPU rewrite mode enabled) and the FMSTP bit is 1 (flash memory is stopped), registers FMR0, FMR1, and FMR2 are initialized if a watchdog timer, oscillation stop detection, or voltage monitor 1 interrupt is generated.

Table 19.13 Interrupt Handling during CPU Rewrite Operation (EW1 Mode)

Interrupt Type	Data Flash/Program ROM	
	Suspend Enabled (FMR20 = 1)	Suspend Disabled (FMR20 = 0)
Maskable interrupt	When an interrupt request is acknowledged, the FMR21 bit is automatically set to 1 (suspend request) if the FMR22 bit is 1 (suspend request enabled by interrupt request). The flash memory suspends auto-erase or auto-programming after td(SR-SUS) and interrupt handling is executed. When auto-erase is being suspended, auto-programming and reading can be executed for any block other than the blocks being auto-erased. When auto-programming is being suspended, any block other than the blocks being auto-programmed can be read. After interrupt handling completes, auto-erase or auto-programming can be restarted by setting the FMR21 bit is set to 0 (restart). If the FMR22 bit is set to 0 (suspend request disabled by interrupt request), auto-erasure and auto-programming have priority and interrupt requests are put on standby. Interrupt handling is executed after auto-erase and auto-program complete.	Auto-erase or auto-programming has priority. Interrupt handling is executed after auto-erase or auto-programming.
Address match	Do not use during auto-erasing or auto-programming.	
UND, INTO, and BRK instructions		
Single-step		
Watchdog timer	When an interrupt request is acknowledged, auto-erase or auto-programming is forcibly stopped immediately and the flash memory is reset. After the specified period, the flash memory is restarted before interrupt handling is started. Since auto-erase or auto-programming is forcibly stopped, the correct values may not be read from the block being auto-erased or the address being auto-programmed. After the flash memory is restarted, execute auto-erase again and verify it complete normally. The watchdog timer does not stop while the command is executing, so interrupt requests may be generated. Initialize the watchdog timer periodically using the erase-suspend function. Since the flash memory control registers are initialized in this case, these registers must be set again. (1)	
Oscillation stop detection		
Voltage monitor 1		

FMR20, FMR21, FMR22: Bits in FMR2 register

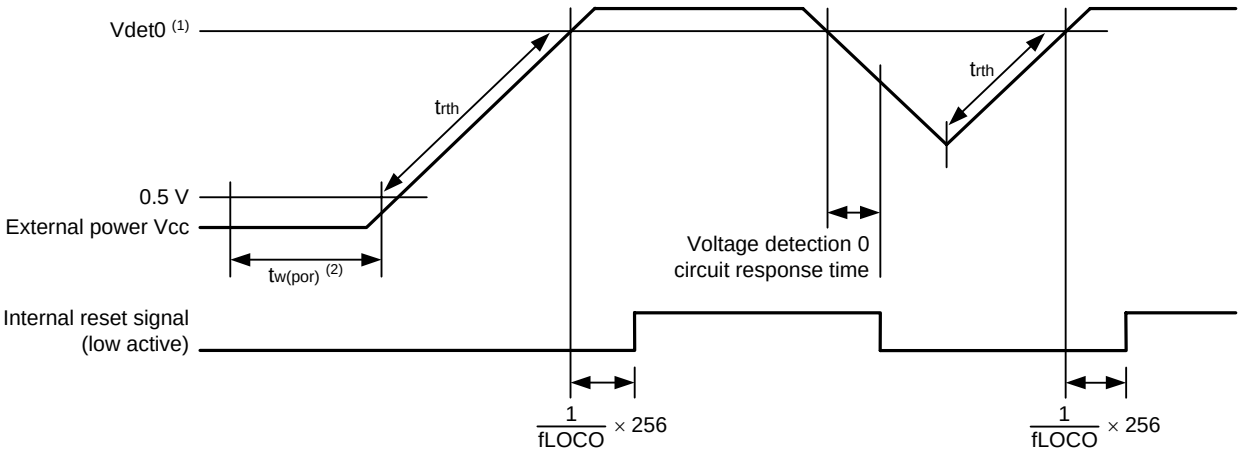
Note:

1. Registers FMR0, FMR1, and FMR2 are initialized if a watchdog timer, oscillation stop detection, or voltage monitor 1 interrupt is generated while the flash memory is busy.
   When the FMR01 bit in the FMR0 register is 1 (CPU rewrite mode enabled) and the FMSTP bit is 1 (flash memory is stopped), registers FMR0, FMR1, and FMR2 are initialized if a watchdog timer, oscillation stop detection, or voltage monitor 1 interrupt is generated.

Table 20.9 Power-On Reset Circuit (2)

Symbol	Parameter	Condition	Standard			Unit
			Min.	Typ.	Max.	
t_{rth}	External power Vcc rise gradient		0	—	50,000	mV/msec

Notes:
1. The measurement condition is Topr = -20 °C to 85 °C (N version)/-40 °C to 85 °C (D version), unless otherwise specified.
2. To use the power-on reset function, enable the voltage monitor 0 reset by setting the LVDAS bit in the OFS register to 0.



Notes:
1. Vdet0 indicates the voltage detection level of the voltage detection 0 circuit. For details, see **7. Voltage Detection Circuit**.
2. $t_{w(por)}$ is required for a power-on reset to be enabled with the external power Vcc held below the valid voltage (0.5 V) to enable a power-on reset. When Vcc decreases with voltage monitor 0 reset disabled and then turns on, maintain $t_{w(por)}$ for 1 ms or more.

Figure 20.3 Power-On Reset Circuit Electrical Characteristics

(8) When the TEDGSEL bit in the TRJIOC register is set to 0 (count on rising edge) and the external signal (TRJIO) is counted in event counter mode, the signal may not be counted correctly depending on the state of the TSTART bit in the TRJCR register (see Figure 21.4).
If the TRJIO pin is set to low before the TSTART bit is set to 1 (count is started) and a valid event is input after the TSTART bit is set to 1, the signal is not counted on the first rising edge of the TRJIO input.
Thus, the number of counted events is obtained as follows:
Number of counted events = {(initial value in the counter – value in the counter on completion of the valid event + 1) + 1}
To avoid this, set the TRJIO pin to low after setting the TSTART bit to 1 (count is started) (see Figure 21.5).

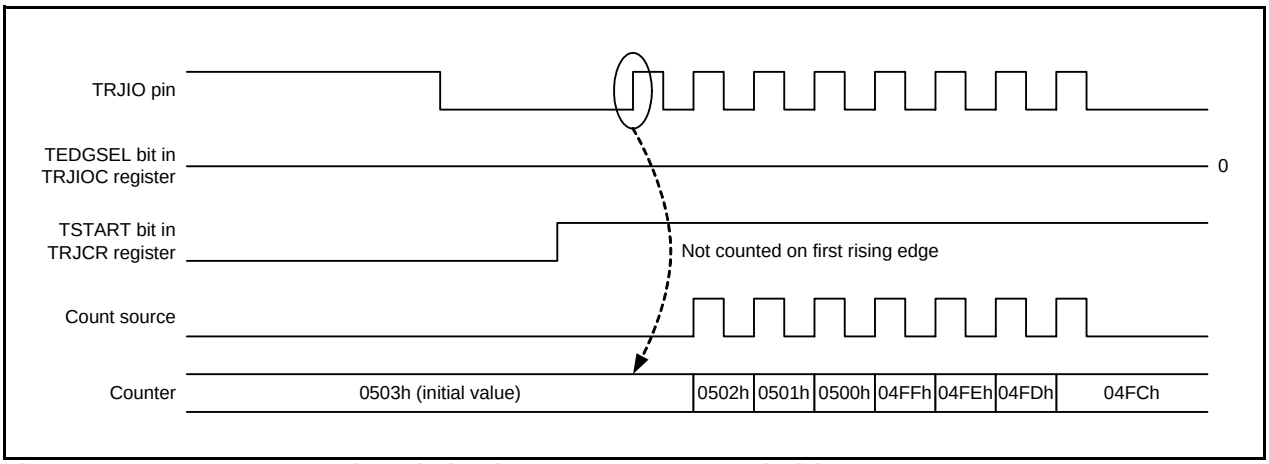


Figure 21.4 TSTART Setting Timing in Event Counter Mode (1)

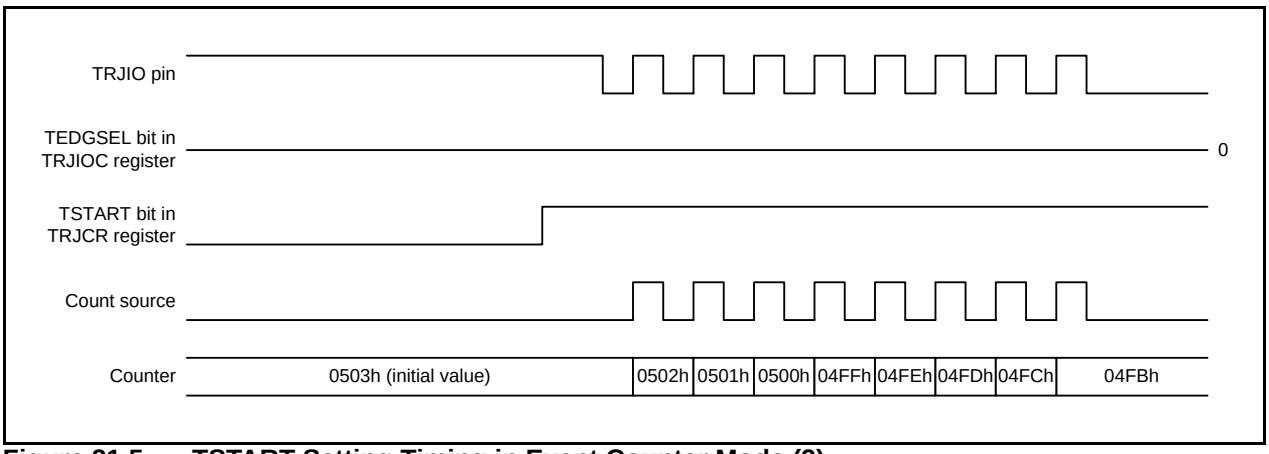


Figure 21.5 TSTART Setting Timing in Event Counter Mode (2)

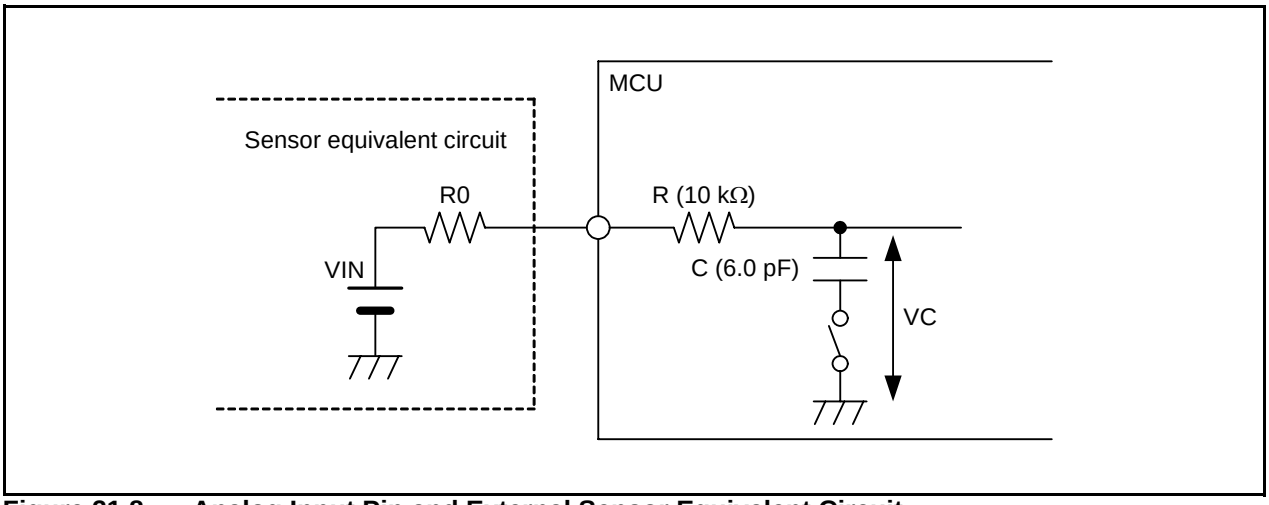


Figure 21.8 Analog Input Pin and External Sensor Equivalent Circuit

21.11.3 Register Setting

- Registers ADMOD and ADINSEL must be written only when A/D conversion is stopped.
- Do not enter stop mode during A/D conversion.
- Do not enter wait mode during A/D conversion while the WCKSTP bit in the CKSTPR register is 1 (system clock is stopped in wait mode).
- Do not set the FMSTP bit in the FMR0 register to 1 (flash memory is stopped) or the FMR27 bit in the FMR2 register to 1 (low-current-consumption read mode enabled) during A/D conversion.
- During A/D conversion, if the ADST bit in the ADCON0 register is set to 0 (A/D conversion stops) by a program to forcibly terminate the conversion, the conversion result from the A/D converter will be undefined and no interrupt will be generated. The value of the ADi register (i = 0 or 1) which is not engaged in A/D conversion may also be undefined.
  If the ADST bit is set to 0 by a program, do not use any of the values of the ADi register.
- When using the A/D converter, it is recommended that the average of the conversion results be taken.

Index