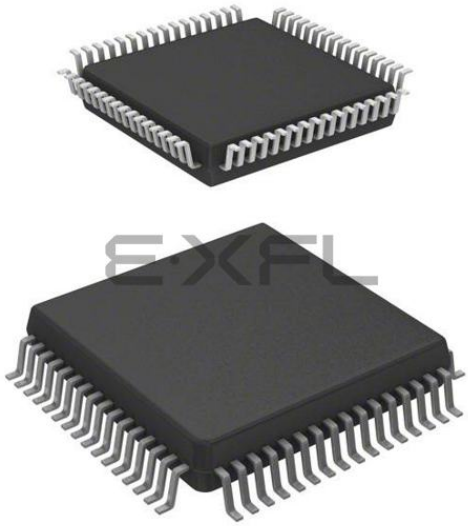


Welcome to E-XFL.COM

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	16MHz
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)	3V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-BQFP
Supplier Device Package	64-QFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/df36074lhwv

1. Overview

1.1 Features

The R8C/M11A Group and R8C/M12A Group of single-chip microcontrollers (MCUs) incorporate the R8C CPU core, which provides sophisticated instructions for a high level of efficiency. With 1 Mbyte of address space, the CPU core is capable of executing instructions at high speed. In addition, it features a multiplier for high-speed arithmetic processing.
Power consumption is low, and the supported operating modes allow additional power control. These MCUs are designed to maximize EMI/EMS performance.
Integration of many peripheral functions on the same chip, including multifunction timer and serial interface, reduces the number of system components.
The R8C/M11A Group and R8C/M12A Group include data flash (1 KB × 2 blocks).

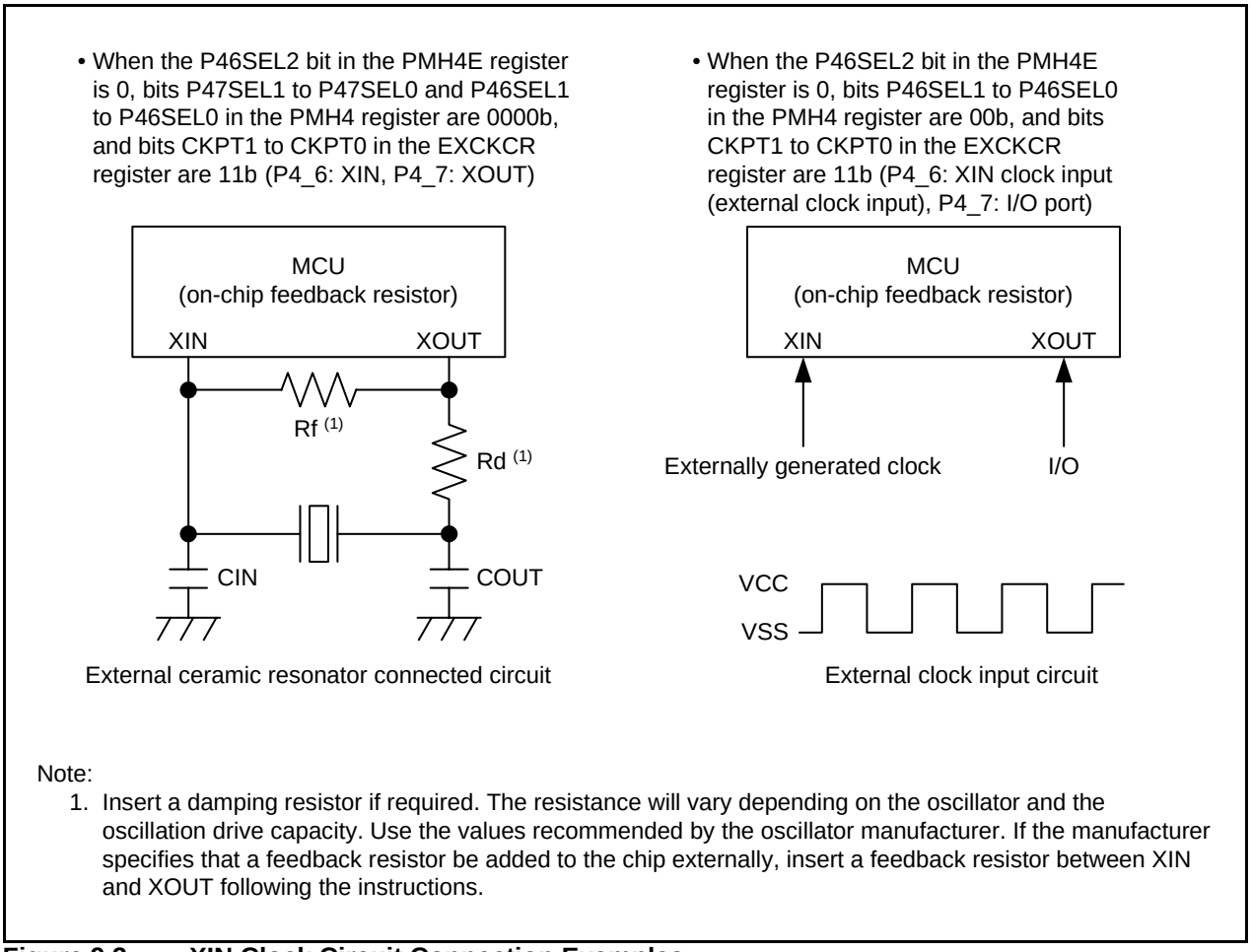
1.1.1 Applications

Home appliances, office equipment, audio equipment, consumer products, etc.

9.3 Clock Oscillation Circuit

9.3.1 XIN Clock Oscillation Circuit

The XIN clock is supplied by the XIN clock oscillation circuit. This clock is used as the clock source for the CPU clock and the peripheral function clock. The XIN clock oscillation circuit is configured by connecting an oscillator between pins XIN and XOUT. The XIN clock oscillation circuit includes an on-chip feedback resistor, which is disconnected from the oscillation circuit in stop mode to reduce power consumption. To input an externally generated clock to the XIN pin, set the P46SEL2 bit in the PMH4E register to 0, bits P46SEL1 to P46SEL0 in the PMH4 register to 00b (I/O port or XIN input), and bits CKPT1 to CKPT0 in the EXCKCR register to 01b (XIN clock input).
Figure 9.3 shows the XIN Clock Circuit Connection Examples.
The XIN clock is stopped during and after a reset.
The XIN clock starts oscillating when the P46SEL2 bit in the PMH4E register is set to 0, bits P47SEL1 to P47SEL0 and P46SEL1 to P46SEL0 in the PMH4 register are set to 0000b, and bits CKPT1 to CKPT0 in the EXCKCR register are set to 11b (P4_6: XIN, P4_7: XOUT). After the XIN clock oscillation stabilizes, when the HSCKSEL bit in the SCKCR register is set to 0 (XIN clock) and the SCKSEL bit in the CKSTPR register is set to 1 (fHSCK), the XIN clock is selected to be used as the clock source for the CPU clock and the peripheral function clock.
When the high-speed on-chip oscillator or the low-speed on-chip oscillator is used as the system base clock, the XIN clock oscillation is stopped by setting bits CKPT1 to CKPT0 in the EXCKCR register to 00b. This reduces power consumption.
The XIN clock is stopped in stop mode. When inputting an externally generated clock to the XIN clock, do not use stop mode. See **10. Power Control** for details.



Note:
1. Insert a damping resistor if required. The resistance will vary depending on the oscillator and the oscillation drive capacity. Use the values recommended by the oscillator manufacturer. If the manufacturer specifies that a feedback resistor be added to the chip externally, insert a feedback resistor between XIN and XOUT following the instructions.

Figure 9.3 XIN Clock Circuit Connection Examples

The system base clock when returning from stop mode by a peripheral interrupt is the clock set by the STOPRS bit in the CKRSCR register. At this time, bits PHISSEL0 to PHISSEL2 in the SCKCR register and the SCKSEL bit in the CKSTPR register are automatically changed according to bits PHISRS and STOPRS.
When an interrupt is generated, oscillation is started, and a period until the clock supply (oscillation stabilization time) is generated automatically. If the system base clock when returning is the XIN clock, set pins P4_6 and P4_7 to XIN oscillation by a program before entering stop mode.
Depending on the clock to be used, set appropriate values for oscillation stabilization time using bits CKST0 to CKST3 in the CKRSCR register. It is unnecessary to generate a wait time by a program.

FMR0 Register FMSTP bit	Internal Power Stabilization Time (T0)	Time until Flash Memory Activation (T1)	Clock Stabilization Time (T2)	Time until CPU Clock Supply (T3)	Time for Interrupt Sequence (T4)	Remarks
0 (flash memory operates)	100 μs (max.)	60 μs (max.)	Set by bits CKST0 to CKST3 in the CKRSCR register	CPU clock period × 2 cycles	CPU clock period × 20 cycles	The total on the left amounts to the time from stop mode to execution of an interrupt routine.
1 (flash memory is stopped)	100 μs (max.)	— (Flash memory is not activated)				

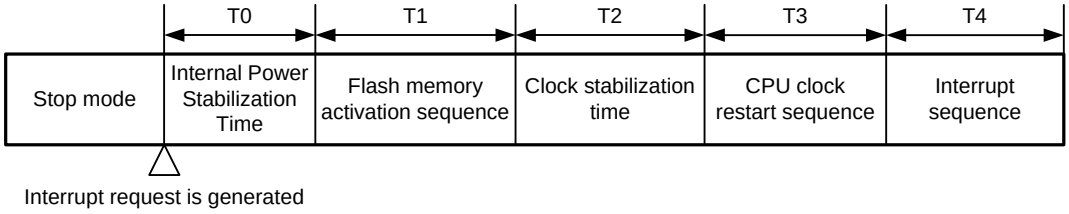


Figure 10.4 Sequence from Stop Mode to Interrupt Routine Execution

11.2.9 External Interrupt Flag Register (IRR3)

Address 00053h

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Symbol	—	—	IRKI	—	IRI3	IRI2	IRI1	IRI0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	IRI0	$\overline{\text{INT0}}$ interrupt request flag	0: No interrupt requested 1: Interrupt requested	R/W
b1	IRI1	$\overline{\text{INT1}}$ interrupt request flag		R/W
b2	IRI2	$\overline{\text{INT2}}$ interrupt request flag		R/W
b3	IRI3	$\overline{\text{INT3}}$ interrupt request flag		R/W
b4	—	Reserved	Set to 0.	R/W
b5	IRKI	Key input interrupt request flag	0: No interrupt requested 1: Interrupt requested	R/W
b6	—	Nothing is assigned. The write value must be 0. The read value is 0.		—
b7	—			

IRI0 Bit ($\overline{\text{INT0}}$ interrupt request flag)

Writing 0 after reading the value 1 sets the IRI0 bit to 0. This bit is also automatically set to 0 when the corresponding interrupt ($\overline{\text{INT0}}$) is acknowledged.

IRI1 Bit ($\overline{\text{INT1}}$ interrupt request flag)

Writing 0 after reading the value 1 sets the IRI1 bit to 0. This bit is also automatically set to 0 when the corresponding interrupt ($\overline{\text{INT1}}$) is acknowledged.

IRI2 Bit ($\overline{\text{INT2}}$ interrupt request flag)

Writing 0 after reading the value 1 sets the IRI2 bit to 0. This bit is also automatically set to 0 when the corresponding interrupt ($\overline{\text{INT2}}$) is acknowledged.

IRI3 Bit ($\overline{\text{INT3}}$ interrupt request flag)

Writing 0 after reading the value 1 sets the IRI3 bit to 0. This bit is also automatically set to 0 when the corresponding interrupt ($\overline{\text{INT3}}$) is acknowledged.

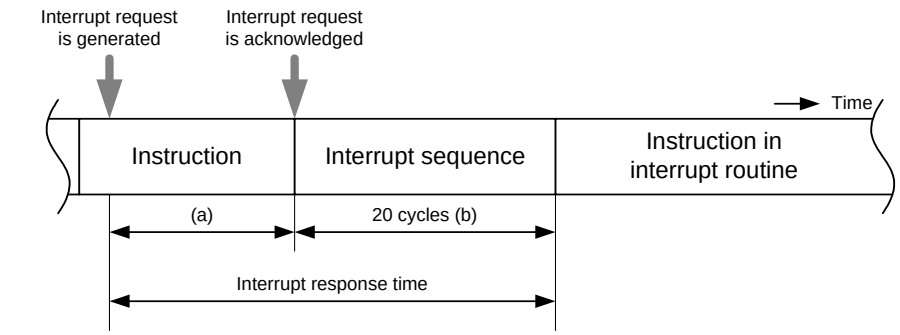
IRKI Bit (Key input interrupt request flag)

Writing 0 after reading the value 1 sets the IRKI bit to 0. This bit is also automatically set to 0 when the corresponding interrupt (key input) is acknowledged.

The interrupt priority level register must be rewritten only while no interrupt requests corresponding to that register are generated. See **11.9.7 Changing Interrupt Priority Levels and Flag Registers**.

11.4.5 Interrupt Response Time

Figure 11.4 shows the Interrupt Response Time. The interrupt response time is the period from when an interrupt request is generated until the first instruction in the interrupt routine is executed. This time consists of two periods: the first period ranges from when an interrupt request is generated until the currently executing instruction is completed ((a) in Figure 11.4) and the second from when an interrupt request is acknowledged until the interrupt sequence is executed (20 cycles (b)).



(a) The period from when an interrupt request is generated until the currently executing instruction is completed. The length of this time varies with the instruction being executed. The DIVX instruction requires the longest time, 30 cycles (no wait states, and the divisor is a register).
(b) 21 cycles for address match and single-step interrupts.

Figure 11.4 Interrupt Response Time

11.4.6 IPL Change When Interrupt Request is Acknowledged

When a maskable interrupt request is acknowledged, the interrupt priority level of the acknowledged interrupt is set in the IPL.

For a software interrupt or special interrupt request, the level listed in Table 11.10 is set in the IPL.

Table 11.10 IPL Value When Software Interrupt or Special Interrupt is Acknowledged

Interrupt Source without Interrupt Priority Level	Value Set in IPL
Watchdog timer, oscillation stop detection, voltage monitor 1	7
Software, address match, single-step	Not changed

11.9.5 $\overline{\text{INTi}}$ Input Filter (i = 0 to 3) When Returning from Wait Mode or Stop Mode to Standard Mode

When a transition is made to wait mode or stop mode with the WCKSTP bit in the CKSTPR register set to 1 (system clock stopped in wait mode) while in use of the $\overline{\text{INTi}}$ input filter, the $\overline{\text{INTi}}$ interrupt cannot be used to return to standard operating mode.

When the $\overline{\text{INTi}}$ interrupt is used to return, set the WCKSTP bit to 1 and bits INTiF1 to INTiF0 in the INTF0 register to 00b (no filter) before a transition is made to wait mode or stop mode. When the $\overline{\text{INTi}}$ input filter is used again, select the sampling clock with bits INTiF0 to INTiF1 to enable the INTiEN bit in the INTEN register.

Figure 11.14 shows the Register Setting Procedure When $\overline{\text{INTi}}$ Input Filter (i = 0 to 3) is Used.

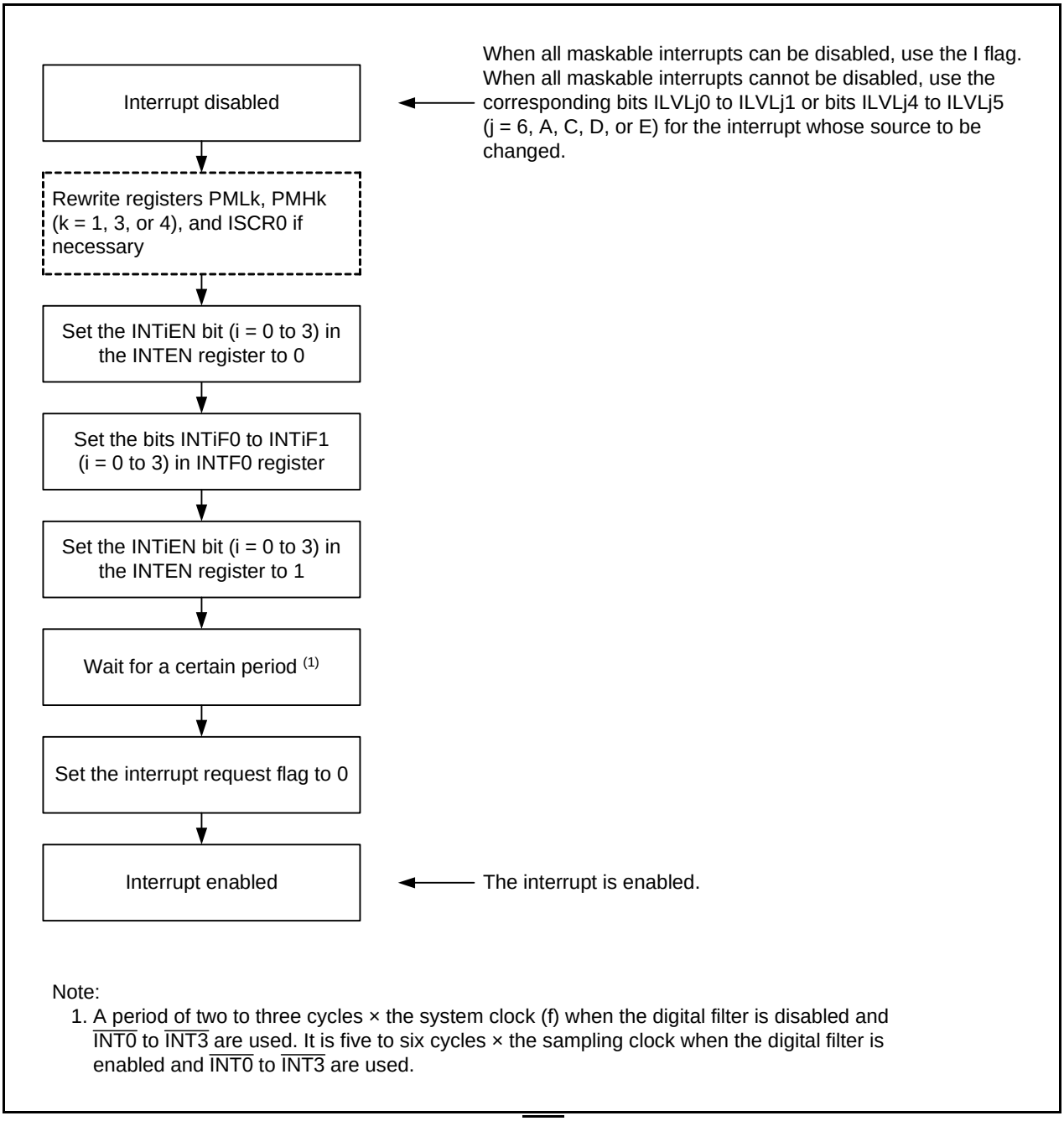


Figure 11.14 Register Setting Procedure When $\overline{\text{INTi}}$ Input Filter (i = 0 to 3) is Used

13.3.4 Timer RJ Mode Register (TRJMR)

Address 000DCh

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Symbol	TCKCUT	TCK2	TCK1	TCK0	TEDGPL	TMOD2	TMOD1	TMOD0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	TMOD0	Timer RJ operating mode select bits	b2 b1 b0 0 0 0: Timer mode 0 0 1: Pulse output mode 0 1 0: Event counter mode 0 1 1: Pulse width measurement mode 1 0 0: Pulse period measurement mode Other than the above: Do not set.	R/W
b1	TMOD1			R/W
b2	TMOD2			R/W
b3	TEDGPL	TRJIO edge polarity select bit	0: One-way edge 1: Two-way edge	R/W
b4	TCK0	Timer RJ count source select bits (1, 2)	b6 b5 b4 0 0 0: f1 0 0 1: f8 0 1 0: fHOCO 0 1 1: f2 Other than the above: Do not set.	R/W
b5	TCK1			R/W
b6	TCK2			R/W
b7	TCKCUT	Timer RJ count source cutoff bit (2)	0: Count source is supplied 1: Count source is cut off	R/W

Notes:
1. When event counter mode is selected, the external input (TRJIO) is selected as the count source regardless of the setting of bits TCK0 to TCK2.
2. Do not switch or cut off the count source during count operation. When switching or cutting off the count source, set the TSTART bit in the TRJCR register to 0 (count is stopped) and the TCSTF bit to 0 (count is stopped) to stop the timer count.

Select the operating mode when the count is stopped (the TSTART bit is 0 and the TCSTF bit is 0).
When a value is written to the TRJMR register, the toggle flip-flop is initialized.

13.3.5 Timer RJ Event Select Register (TRJISR)

Address 000DDh

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Symbol	—	—	—	—	—	RCCPSEL2	RCCPSEL1	RCCPSEL0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	RCCPSEL0	Timer RC output signal select bits	b1 b0 0 0: TRCIOD 0 1: TRCIOC 1 0: TRCIOB 1 1: TRCIOA	R/W
b1	RCCPSEL1			R/W
b2	RCCPSEL2	Timer RC output signal inversion bit	0: Low-level period of timer RC output signal is counted 1: High-level period of timer RC output signal is counted	R/W
b3	—	Nothing is assigned. The write value must be 0. The read value is 0.		—
b4	—			
b5	—			
b6	—			
b7	—			

13.4.4 Event Counter Mode

In this mode, the counter is decremented by an external pulse signal input to the TRJIO pin.
Various periods for counting events can be set by bits TIOGT0 to TIOGT1 in the TRJIOC register and the TRJISR register. In addition, the filter function for the TRJIO input can be specified by bits TIPF0 to TIPF1 in the TRJIOC register.
Also, the output from the TRJO pin can be toggled even in event counter mode.
When event counter mode is used, see **13.5 Notes on Timer RJ2 (3) (8) (9)**.
Figure 13.5 shows an Operation Example in Event Counter Mode.

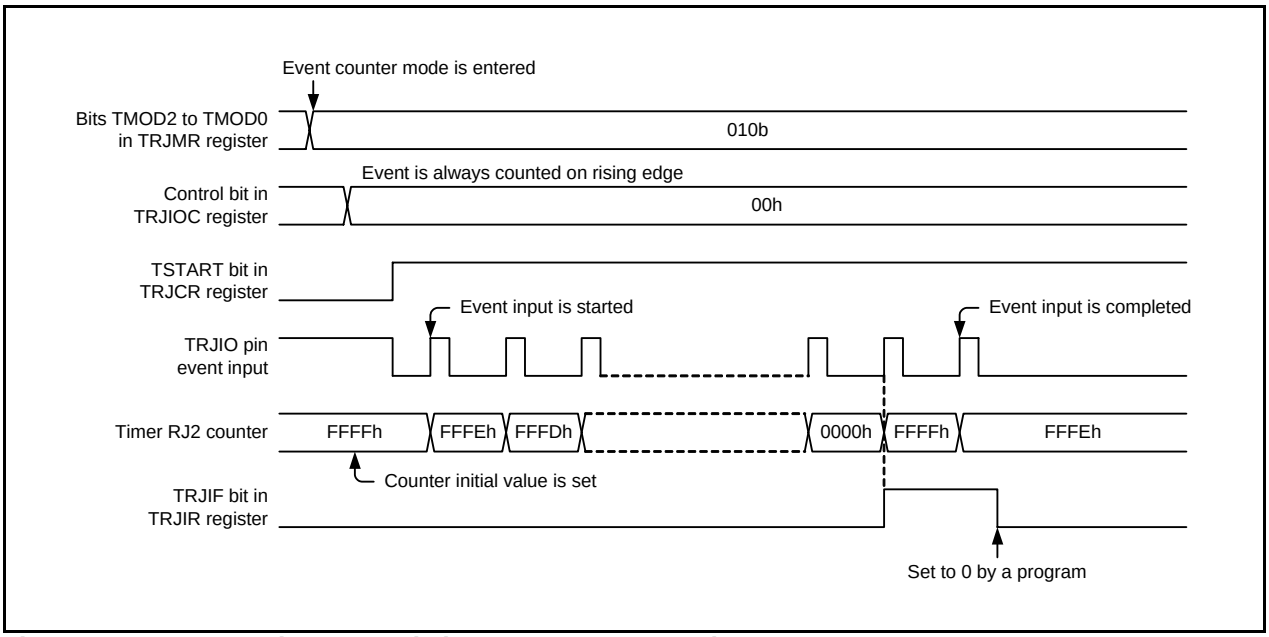
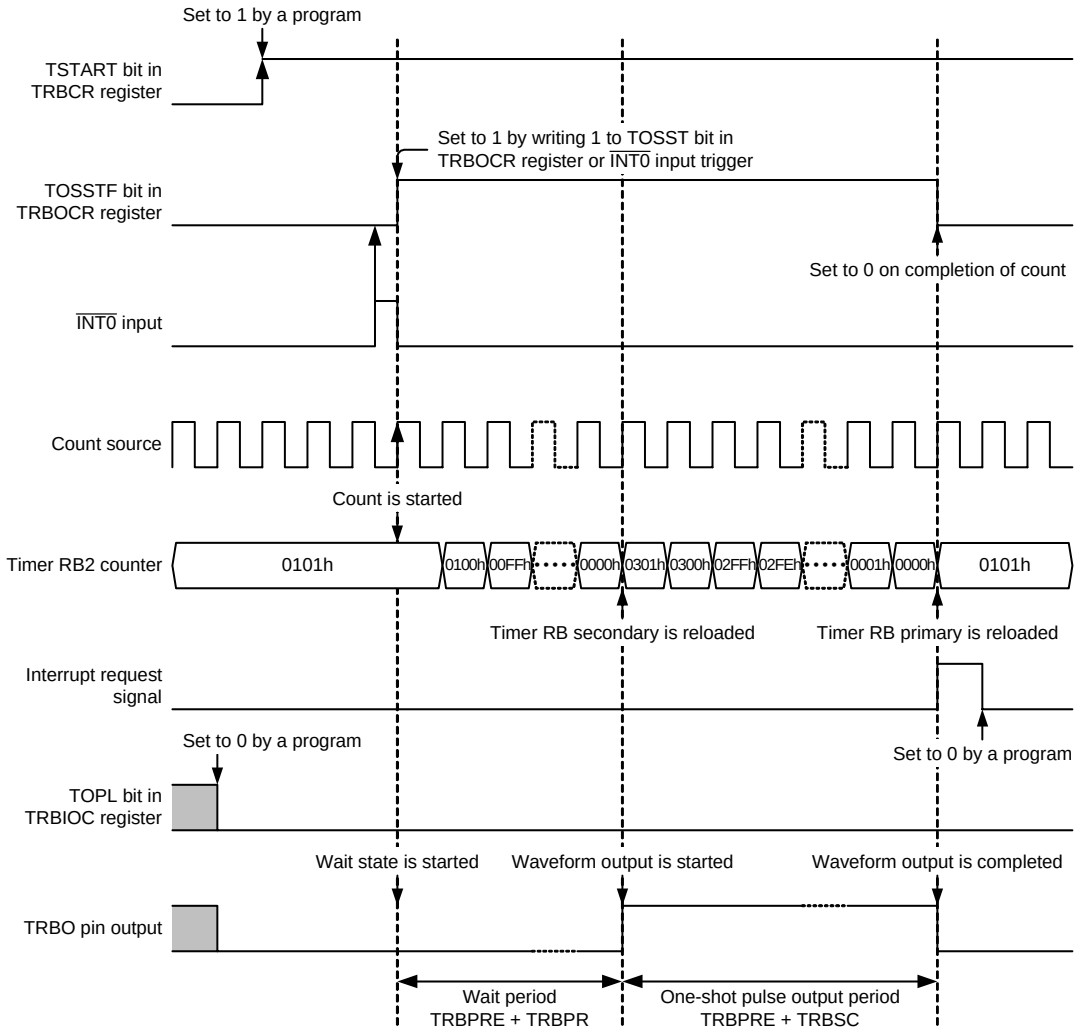


Figure 13.5 Operation Example in Event Counter Mode



The above diagram applies under the following conditions:
- TRBPRE register = 01h, TRBPR register = 01h, TRBSC register = 03h
- TOPL bit = 0, TOCNT bit = 0 (waveform output),
  INOSTG bit = 1 (one-shot trigger to INT0 pin enabled), INOSEG bit = 1 (rising edge) in TRBIOC register
- TCNT16 bit in TRBMR register = 1 (16-bit timer)

Figure 14.8 Example of 16-Bit Timer Operation in Programmable Wait One-Shot Generation Mode

15.3 Operation

Table 15.10 lists the Timer RC Operating Modes.

Table 15.10 Timer RC Operating Modes

Item	Description
Timer mode	Timer mode is used by setting the PWM2 bit to 0 and bits PWMB to PWMD to 0 in the TRCMR register. In this case, the output compare function or input capture function is used by setting bits IOA0 to IOA2 and IOB0 to IOB2 in the TRCIOR0 register and bits IOC0 to IOC2 and IOD0 to IOD2 in the TRCIOR1 register.
PWM mode	PWM mode is used by setting the PWM2 bit to 0 and bits PWMB to PWMD to 1 in the TRCMR register.
PWM2 mode	PWM2 mode is used by setting the PWM2 bit in the TRCMR register to 1.

Tables 15.11 to 15.14 list the settings of pins TRCIOA to TRCIOD. For the assignments of pins TRCIOA to TRCIOD, see **12. I/O Ports**.

Table 15.11 TRCIOA Pin Settings

Register	TRCOER	TRCMR	TRCIOR0			Function
Bit	EA	PWM2	IOA2	IOA1	IOA0	
Setting value	0	1	0	0	1	Timer mode waveform output (output compare function)
				1	X	
	X	1	1	X	X	Timer mode (input capture function)
	Other than the above					I/O port

X: 0 or 1

Table 15.12 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	
	X	1	0	1	X	X	Timer mode (input capture function)
	Other than the above						I/O port

X: 0 or 1

Table 15.13 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	
	X	1	0	1	X	X	Timer mode (input capture function)
	PWM2 = 1 and other than the above						I/O port

X: 0 or 1

Table 15.14 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	
	X	1	0	1	X	X	Timer mode (input capture function)
	PWM2 = 1 and other than the above						I/O port

X: 0 or 1

15.3.2 PWM Mode

In PWM mode, when the TRCGRA register is set as the period register and registers TRCGRB, TRCGRC, and TRCGRD are set as duty registers, a PWM waveform is output from pins TRCIOB, TRCIOC, and TRCIOD individually. A PWM waveform with up to three phases can be output. In this mode, the general register automatically functions as an output compare register. The settings of bits IOB2, IOC2, and IOD2 are invalid. The initial output level of the corresponding pin is set according to the values in bits TOA to TOD in the TRCCR1 register and bits POLB to POLD in the TRCCR2 register.
Table 15.15 lists the Initial Output Levels of TRCIOB Pin.

Table 15.15 Initial Output Levels of TRCIOB Pin

TOB Bit in TRCCR1 Register	POLB Bit in TRCCR2 Register	Initial Output Level
0	0	1
	1	0
1	0	0
	1	1

The output level is determined by bits POLB to POLD in the TRCCR2 register. When the POLB bit is 0 (output level is active low), the TRCIOB output pin is set to low at compare match B and high at compare match A. When the POLB bit is 1 (output level is active high), the TRCIOB output pin is set to high at compare match B and low at compare match A.
The setting values of bits PWMD to PWMB in TRCMR take precedence over those in registers TRCIOR0 and TRCIOR1. When the values set in the period and duty registers are the same, the output value remains unchanged even if a compare match occurs.

In PWM2 mode, the TRCTRG input is used to output a pulse with an arbitrary delay time and width from the TRCIOB pin.
Set bits TCEG1 to TCEG0 in the TRCCR2 register to 10b (falling edge) to set the falling edge for the TRCTRG input. Set the CSTP bit in the TRCCR2 register to 0 (increment is continued) to continue incrementing when compare match A with the TRCGRA register occurs. Set the BUFEB bit in the TRCMR register to 1 (TRCGRD register is used as a buffer register for TRCGRB register) to set the TRCGRD register as the buffer register. Set the TOB bit in the TRCCR1 register to 0 (output value 0) or 1 (output value 1) to set the initial level of the output level to 0 or 1. Next, set the CCLR bit in the TRCCR1 register to 1 (TRCCNT counter is cleared by input capture/compare match A) to clear the TRCCNT register by compare match A.
Figure 15.14 shows an Operation Example in PWM2 Mode When TRCTRG Input is Enabled. Figure 15.15 shows an Operation Example in PWM2 Mode When TRCTRG Input is Disabled. These examples apply when the PWM2 bit in the TRCMR register is set to 0 (PWM2 mode) and a waveform is output from the TRCIOB pin.
In PWM2 mode, when the TOB bit in the TRCCR1 register is 0 (output value 0), the TRCTRG input edge is disabled while a high level is output from the TRCIOB pin. Likewise, when the TOB bit is 1 (output value 1), the TRCTRG input edge is disabled while a low level is output from the TRCIOB pin. In addition, transfer from registers TRCGRD to TRCGRB is performed when a compare match with the TRCGRA register or TRCTRG input occurs. However, if the TRCTRG input is disabled depending on the level of the TRCIOB pin, transfer from registers TRCGRD to TRCGRB is not performed.

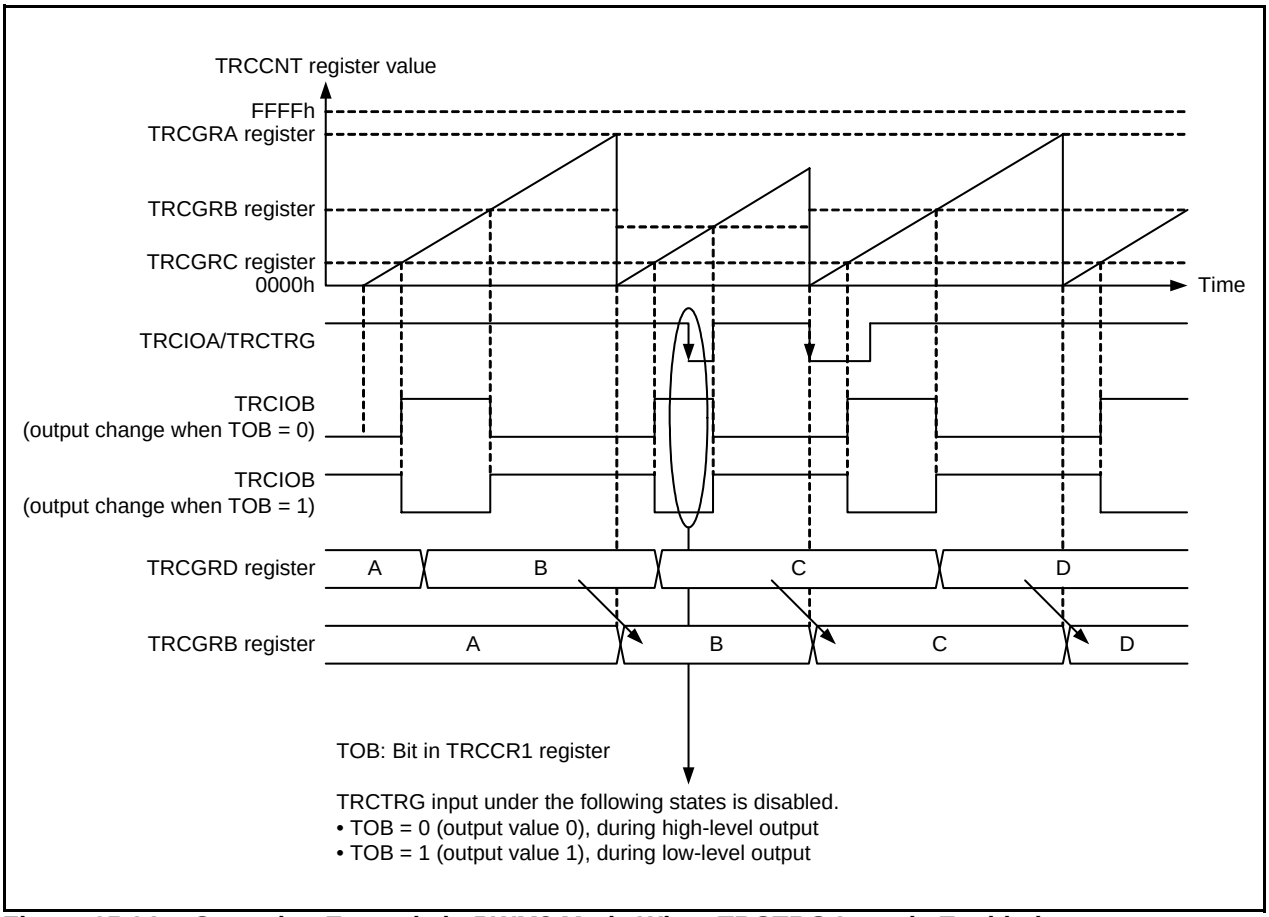
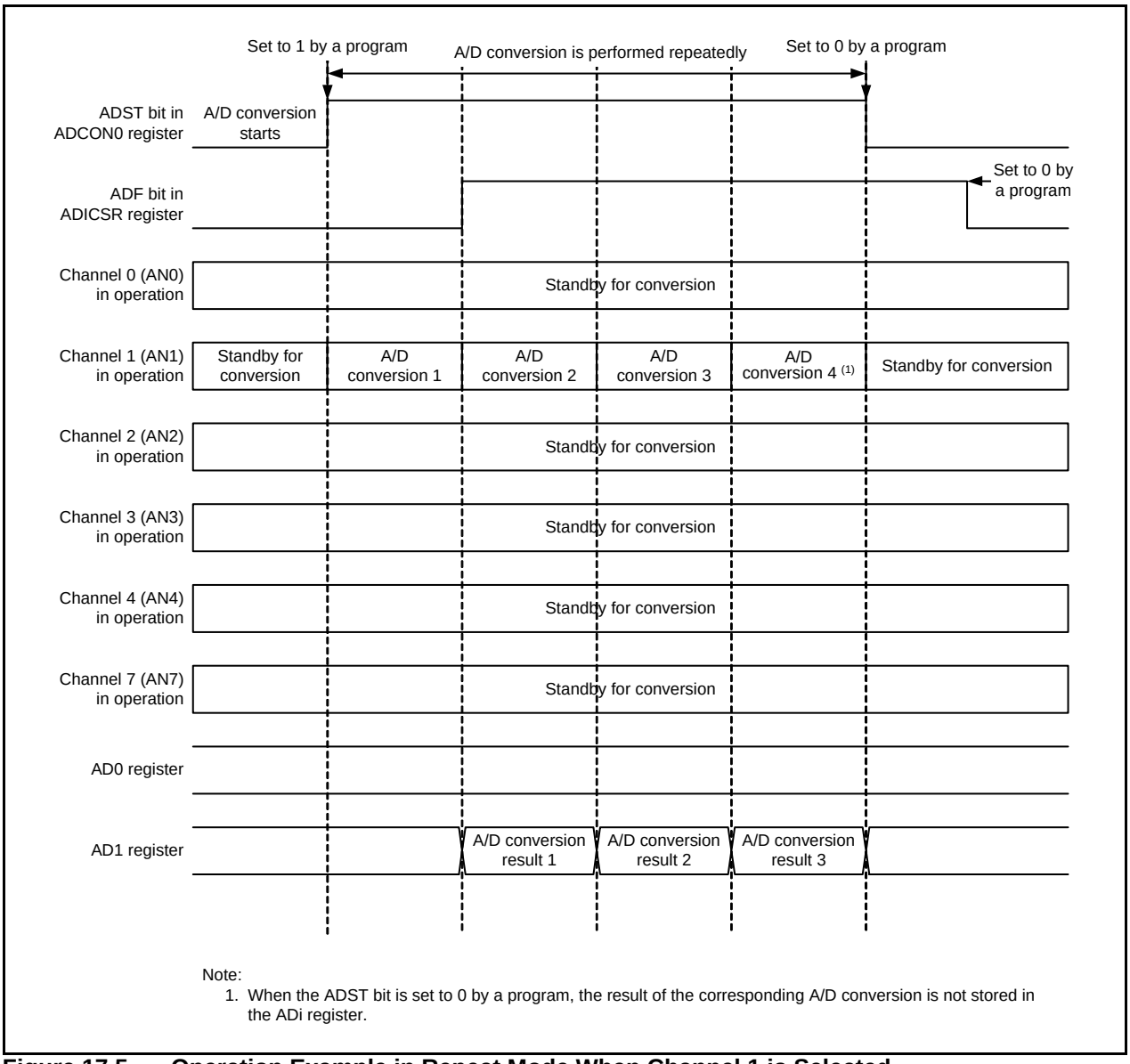


Figure 15.14 Operation Example in PWM2 Mode When TRCTRG Input is Enabled

17.3.3 Repeat Mode

Figure 17.5 shows an Operation Example in Repeat Mode When Channel 1 is Selected.
In repeat mode, A/D conversions of an analog input are performed for the specified single channel 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 on the selected channel.
(2) When A/D conversion completes, the result is transferred to the ADi register (i = 0 or 1) corresponding to the channel.
(3) When A/D conversion completes, 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 on the selected channel.



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 17.5 Operation Example in Repeat Mode When Channel 1 is Selected

18. Comparator B

Comparator B consists of two independent comparators, B1 and B3, which compare an analog input voltage with a reference input voltage.

18.1 Overview

The comparison result between the reference input voltage and the analog input voltage can be read by software. Table 18.1 lists the Comparator B Specifications. Figure 18.1 shows the Comparator B Block Diagram. Table 18.2 lists the Comparator B Pin Configuration.

Table 18.1 Comparator B Specifications

Item	Specification	
Input voltage	Reference input	Input from the reference pin (IVREFi)
	Analog input	Input voltage from the analog pin (IVCMPi)
Comparison result	The result can be read from the WCBiOUT bit in the WCMPR register or monitored with the VCOUTi pin.	
Interrupt request generation timing	When the comparison result changes.	
Digital filter function	• The digital filter can be enabled or disabled. • The sampling frequency can be selected (f1, f8, or f32).	

i = 1 or 3

18.2 Registers

Table 18.3 lists the Comparator B Register Configuration.

Table 18.3 Comparator B Register Configuration

Register Name	Symbol	After Reset	Address	Access Size
Comparator B Control Register	WCMPR	00h	00180h	8
Comparator B1 Interrupt Control Register	WCB1INTR	00h	00181h	8
Comparator B3 Interrupt Control Register	WCB3INTR	00h	00182h	8

18.2.1 Comparator B Control Register (WCMPR)

Address 00180h

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Symbol	WCB3OUT	—	—	WCB3M0	WCB1OUT	—	—	WCB1M0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b0	WCB1M0	Comparator B1 operation enable bit	0: Operation disabled 1: Operation enabled	R/W
b1	—	Reserved	Set to 0.	R/W
b2	—			
b3	WCB1OUT	Comparator B1 monitor flag	0: IVCMP1 < IVREF1 or comparator B1 disabled 1: IVCMP1 > IVREF1	R
b4	WCB3M0	Comparator B3 operation enable bit	0: Operation disabled 1: Operation enabled	R/W
b5	—	Reserved	Set to 0.	R/W
b6	—			
b7	WCB3OUT	Comparator B3 monitor flag	0: IVCMP3 < IVREF3 or comparator B3 disabled 1: IVCMP3 > IVREF3	R

Table 20.13 DC Characteristics (1) [4.0 V ≤ Vcc ≤ 5.5 V]

Symbol	Parameter		Condition		Standard			Unit
					Min.	Typ.	Max.	
V_{OH}	Output high voltage	P1_2, P1_3, P1_4, P1_5, P3_3, P3_4, P3_5, P3_7 (2)	When drive capacity is high	I_{OH} = -20 mA	Vcc - 2.0	—	Vcc	V
			When drive capacity is low	I_{OH} = -5 mA	Vcc - 2.0	—	Vcc	V
		P1_0, P1_1, P1_6, P1_7, P4_2, P4_5, P4_6, P4_7, PA_0		I_{OH} = -5 mA	Vcc - 2.0	—	Vcc	V
V_{OL}	Output low voltage	P1_2, P1_3, P1_4, P1_5, P3_3, P3_4, P3_5, P3_7 (2)	When drive capacity is high	I_{OL} = 20 mA	—	—	2.0	V
			When drive capacity is low	I_{OL} = 5 mA	—	—	2.0	V
		P1_0, P1_1, P1_6, P1_7, P4_2, P4_5, P4_6, P4_7, PA_0		I_{OL} = 5 mA	—	—	2.0	V
V_{T+}-V_{T-}	Hysteresis	$\overline{INT0}$, $\overline{INT1}$, $\overline{INT2}$, $\overline{INT3}$, $\overline{KI0}$, $\overline{KI1}$, $\overline{KI2}$, $\overline{KI3}$, TRJIO, TRCIOA, TRCIOB, TRCIOC, TRCIOD, RXD0, CLK0	Vcc = 5 V		0.1	1.2	—	V
		$\overline{RESET}$	Vcc = 5 V		0.1	1.2	—	V
I_{IH}	Input high current		V_I = 5 V, Vcc = 5.0 V		—	—	5.0	µA
I_{IL}	Input low current		V_I = 0 V, Vcc = 5.0 V		—	—	-5.0	µA
R_{PULLUP}	Pull-up resistance		V_I = 0 V, Vcc = 5.0 V		25	50	100	kΩ
R_{fXIN}	Feedback resistance	XIN			—	2.2	—	MΩ
V_{RAM}	RAM hold voltage		In stop mode		1.8	—	—	V

Notes:
1. 4.0 V ≤ Vcc ≤ 5.5 V and Topr = -20 °C to 85 °C (N version)/-40 °C to 85 °C (D version), f(XIN) = 20 MHz, unless otherwise specified.
2. High drive capacity can also be used while the peripheral output function is used.

21.8 Notes on Timer RB2

- Timer RB2 stops counting after a reset. Start the count after setting the value in the timer and prescaler.
- In the 8-bit timer with 8-bit prescaler, even if the prescaler and timer are read in 16-bit units, they are actually read sequentially byte by byte in the MCU. This may cause the value in the timer to be updated during reading of these two registers.
  In the 16-bit timer, access the TRBPRE register first and then the TRBPR register. Read the TRBPRE register first to read the count value in the lower byte. The count value in the higher byte will be retained. Next, read the TRBPR register to read the retained value in the higher byte. The timer value is not updated during reading of these two registers.
- In programmable one-shot and programmable wait one-shot generation modes, when the TOSSP bit in the TRBOCR register is set to 1 and the one-shot is stopped, the timer reloads the reload register value and is stopped. The timer count value must be read before the timer is stopped.
- After 1 (count is started) is written to the TSTART bit in the TRBCR register while the count is stopped, the TCSTF bit in the TRBCR register remains 0 (count is stopped) for two to three cycles of the count source. Do not access the registers associated with timer RB2 (1) other than the TCSTF bit until this bit is set to 1 (count is in progress). The count is started on the first active edge of the counter source after the TCSTF bit is set to 1.
  After 0 (count is stopped) is written to the TSTART bit during count operation, the TCSTF bit remains 1 for two to three cycles of the count source. When the TCSTF bit is set to 0, the count is stopped. Do not access the registers associated with timer RB2 (1) other than the TCSTF bit until this bit is set to 0.
  Note:
  1. Registers associated with timer RB2:
     TRBCR, TRBOCR, TRBIOC, TRBMR, TRBPRE, TRBPR, and TRBSC
- In timer mode, do not set both the TRBPRE and TRBPR registers to 00h at the same time.
- When the TSTART bit in the TRBCR register is 0 (count is stopped), change the values of registers TRBPRE, TRBPR, and TRBSC, then wait for at least two cycles of the system clock (f) before setting the TSTART bit in the TRBCR register to 1 (count is started).
- When the TSTART bit in the TRBCR register is 1 (count is started) or the TCSTF bit is 1 (count is in progress), do not change the values in registers TRBIOC and TRBMR, and the TRBIE bit in the TRBIR register.
- Make sure the TCSTF bit in the TRBCR register is 1 (count is in progress) before writing 1 (one-shot count is started) to the TOSST bit in the TRBOCR register. When the TCSTF bit is 0 (count is stopped), writing 1 (one-shot count is started) to the TOSST bit is invalid.
- When writing to registers TRBPRE, TRBPR, and TRBSC during count operation (the TSTART bit is 1 or the TCSTF bit is 1), note the following points:
  - When writing to the TRBPRE register successively, allow at least three cycles of the count source for each write interval.
  - When writing to the TRBPR register successively, allow at least three cycles of the count source for each write interval.
  - When writing to the TRBSC register successively, allow at least three cycles of the count source for each write interval.
- When the TRBPR register is rewritten in programmable waveform generation mode, do not write to the TRBPRE, TRBPR, or TRBSC register during the secondary output period as described below after rewriting.
  - 8-bit timer with 8-bit prescaler:
    Two cycles of the prescaler underflow before the secondary output period ends.
  - 16-bit timer:
    Two cycles of the count source clock before the secondary output period ends.
- When the underflow signal from timer RJ2 is used as the count source for timer RB2, set timer RJ2 to timer mode, pulse output mode, or event counter mode.

21.10 Notes on Serial Interface (UART0)

Regardless of clock synchronous I/O mode or clock asynchronous I/O mode, read the U0RB register in 16-bit units.
When the U0RBH register is read, bits FER and PER in the U0RB register are set to 0 (no framing error, no parity error). Also, the RI bit in the U0C1 register is set to 0 (the U0RB register empty).
To check receive errors, use the data read from the U0RB register.

• Program example to read the receive buffer register

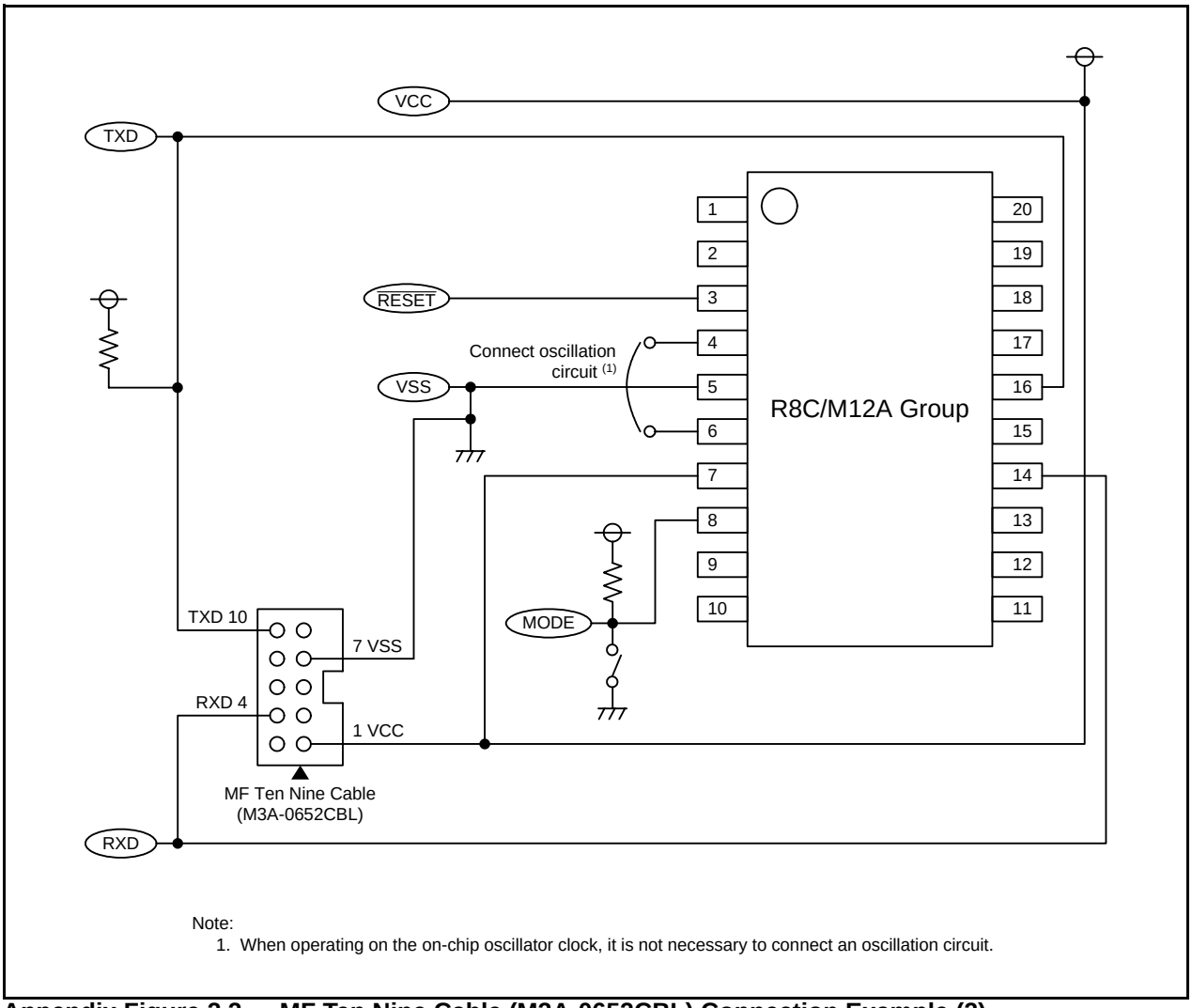
```
MOV.W    0086H, R0        ; Read the U0RB register
```

When the transfer data is 9 bits long in clock asynchronous I/O mode, write to the U0TB register in the order U0TBH first and then U0TBL in 8-bit units.

• Program example to write to the transmit buffer register

```
MOV.B    #XXH, 0083H      ; Write to the U0TBH register
MOV.B    #XXH, 0082H      ; Write to the U0TBL register
```

Do not set the MSTUART bit in the MSTCR register to 1 (standby) during communication. When setting the module to the standby state, confirm whether communication has completed. After communication has completed, set bits TE and RE in the U0C1 register to 0 (communication disabled) before setting the module to the standby state. After the module standby state is cleared, the initial settings for communication must be set again.



Note:
1. When operating on the on-chip oscillator clock, it is not necessary to connect an oscillation circuit.

Appendix Figure 2.2 MF Ten Nine Cable (M3A-0652CBL) Connection Example (2)