Renesas Electronics America Inc - M30260F3AGP#U5A Datasheet

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	M16C/60
Core Size	16-Bit
Speed	20MHz
Connectivity	I²C, IEBus, SIO, UART/USART
Peripherals	DMA, PWM, Voltage Detect, WDT
Number of I/O	39
Program Memory Size	24KB (24K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 12x10b
Oscillator Type	Internal
Operating Temperature	-20°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-LQFP
Supplier Device Package	48-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/m30260f3agp-u5a

Table 1.2. Performance outline of M16C/26A group (M16C/26A, M16C/26B) (42-pin package)

	Item	Performance
CPU	Basic instructions	91 instructions
	Minimun instruction execution time	41.7 ns (f(BCLK) = 24 MHz (3), VCC = 4.2 to 5.5 V (M16C/26B) 50 ns (f(BCLK) = 20 MHz, Vcc = 3.0 to 5.5 V) (M16C/26A, M16C/26B) 100 ns (f(BCLK) = 10 MHz, Vcc = 2.7 to 5.5 V) (M16C/26A, M16C/26B)
	Operation mode	Single-chip mode
	Address space	1M byte
	Memory capacity	ROM/RAM: See **1.4 Product Information**
Peripheral function	Port	33 I/O pins
	Multifunction timer	Timer A: 16 bits x 5 channels, Timer B: 16 bits x 3 channels Three-phase motor control timer
	Serial I/O	1 channel (UART, clock synchronous serial I/O) 1 channel (UART, clock synchronous, I^2C bus, or IEBus(1))
	A/D converter	10 bit A/D converter: 1 circuit, 10 channels
	DMAC	2 channels
	CRC calcuration circuit	1 circuits (CRC-CCITT and CRC-16) with MSB/LSB selectable
	Watchdog timer	15 bits x 1 channel (with prescaler)
	Interrupt	18 internal and 8 external sources, 4 software sources, Interrupt priority level: 7
	Clock generation circuit	4 circuits Main clock(*), Sub-clock(*) On-chip oscillator, PLL frequency synthesizer (*)Equipped with a built-in feedback resister.
	Oscillation stop detection	Main clock oscillation stop, re-oscillation detection function
	Voltage detection circuit	On-chip
Electrical Characteristics	Supply voltage	VCC = 4.2 to 5.5 V (f(BCLK) = 24 MHz)(3) (M16C/26B) VCC = 3.0 to 5.5 V (f(BCLK) = 20 MHz) (M16C/26A, M16C/26B) VCC = 2.7 to 5.5 V (f(BCLK) = 10 MHz)
	Power Consumption	20 mA (Vcc = 5 V, f(BCLK) = 24 MHz) (M16C/26B) 16 mA (Vcc = 5 V, f(BCLK) = 20 MHz) 25 μA (f(XCIN) = 32 KHz on RAM) 3 μA (Vcc = 3 V, f(XCIN) = 32 KHz, in wait mode) 0.7 μA (Vcc = 3 V, in stop mode)
Flash memory	Programming/erasure voltage	2.7 to 5.5 V
	Programming/erasure endurance	100 times (all area) or 1,000 times (block 0 to 3) / 10,000 times (block A, block B)(2)
Operating Ambient Temperature		-20 to 85°C / -40 to 85°C (2)
Package		42-pin plastic molded SSOP

NOTES:
1. IEBus is a trademark of NEC Electronics Corporation.
2. See **Tables 1.7 and 1.8 Product Code** for the program and erase endurance, and operating ambient temperature.
3. The PLL frequency synthesizer is used to run the M16C/26B at f(BCLK) = 24 MHz.

Table 4.6 SFR Information(6)(1)

Address	Register	Symbol	After Reset
03C0₁₆	A/D register 0	AD0	XXXXXXXX₂
03C1₁₆			XXXXXXXX₂
03C2₁₆	A/D register 1	AD1	XXXXXXXX2
03C3₁₆			XXXXXXXX₂
03C4₁₆	A/D register 2	AD2	XXXXXXXX₂
03C5₁₆			XXXXXXXX₂
03C6₁₆	A/D register 3	AD3	XXXXXXXX₂
03C7₁₆			XXXXXXXX₂
03C8₁₆	A/D register 4	AD4	XXXXXXXX₂
03C9₁₆			XXXXXXXX₂
03CA₁₆	A/D register 5	AD5	XXXXXXXX₂
03CB₁₆			XXXXXXXX₂
03CC₁₆	A/D register 6	AD6	XXXXXXXX₂
03CD₁₆			XXXXXXXX₂
03CE₁₆	A/D register 7	AD7	XXXXXXXX₂
03CF₁₆			XXXXXXXX₂
03D0₁₆			
03D1₁₆			
03D2₁₆	A/D trigger control register	ADTRGCON	00₁₆
03D3₁₆	A/D status register 0	ADSTAT0	00000X00₂
03D4₁₆	A/D control register 2	ADCON2	00₁₆
03D5₁₆			
03D6₁₆	A/D control register 0	ADCON0	00000XXX₂
03D7₁₆	A/D control register 1	ADCON1	00₁₆
03D8₁₆			
03D9₁₆			
03DA₁₆			
03DB₁₆			
03DC₁₆			
03DD₁₆			
03DE₁₆			
03DF₁₆			
03E0₁₆			
03E1₁₆	Port P1 register	P1	XX₁₆
03E2₁₆			
03E3₁₆	Port P1 direction register	PD1	00₁₆
03E4₁₆			
03E5₁₆			
03E6₁₆			
03E7₁₆			
03E8₁₆			
03E9₁₆			
03EA₁₆			
03EB₁₆			
03EC₁₆	Port P6 register	P6	XX₁₆
03ED₁₆	Port P7 register	P7	XX₁₆
03EE₁₆	Port P6 direction register	PD6	00₁₆
03EF₁₆	Port P7 direction register	PD7	00₁₆
03F0₁₆	Port P8 register	P8	XX₁₆
03F1₁₆	Port P9 register	P9	XXXXXXXX₂
03F2₁₆	Port P8 direction register	PD8	00₁₆
03F3₁₆	Port P9 direction register	PD9	XXXX0000₂
03F4₁₆	Port P10 register	P10	XX₁₆
03F5₁₆			
03F6₁₆	Port P10 direction register	PD10	00₁₆
03F7₁₆			
03F8₁₆			
03F9₁₆			
03FA₁₆			
03FB₁₆			
03FC₁₆	Pull-up control register 0	PUR0	00₁₆
03FD₁₆	Pull-up control register 1	PUR1	00₁₆
03FE₁₆	Pull-up control register 2	PUR2	00₁₆
03FF₁₆	Port control register	PCR	00₁₆

NOTE:
1. Blank spaces are reserved. No access is allowed.

X: Undefined

Table 7.6.1. Allowed Transition and Setting

		State after transition							
		High-speed mode, middle-speed mode	Low-speed mode[2]	Low power dissipation mode	PLL operation mode[2]	On-chip oscillator mode	On-chip oscillator low power dissipation mode	Stop mode	Wait mode
Current state	High-speed mode, middle-speed mode	8	(9)[7]	--	(13)[3]	(15)	--	(16)[1]	(17)
	Low-speed mode[2]	(8)		(11)[1, 6]	--	(8)	--	(16)[1]	(17)
	Low power dissipation mode	--	(10)		--	--	--	(16)[1]	(17)
	PLL operation mode[2]	(12)[3]	--	--		--	--	--	--
	On-chip oscillator mode	(14)[4]	(9)[7]	--	--	8	(11)[1]	(16)[1]	(17)
	On-chip oscillator low power dissipation mode	--	--	--	--	(10)	8	(16)[1]	(17)
	Stop mode	(18)[5]	(18)	(18)	--	(18)[5]	(18)[5]		--
	Wait mode	(18)	(18)	(18)	--	(18)	(18)	--	

--: Cannot transit

NOTES:
1. Avoid making a transition when the CM20 bit is set to 1 (oscillation stop, re-oscillation detection function enabled). Set the CM20 bit to 0 (oscillation stop, re-oscillation detection function disabled) before transiting.
2. On-chip oscillator clock oscillates and stops in low-speed mode. In this mode, the on-chip oscillator can be used as peripheral function clock. Sub clock oscillates and stops in PLL operation mode. In this mode, sub clock can be used as a clock for the timers A and B.
3. PLL operation mode can only be entered from and changed to high-speed mode.
4. Set the CM06 bit to 1 (division by 8 mode) before transiting from on-chip oscillator mode to high- or middle-speed mode.
5. When exiting stop mode, the CM06 bit is set to 1 (division by 8 mode).
6. If the CM05 bit is set to 1 (main clock stop), then the CM06 bit is set to 1 (division by 8 mode).
7. A transition can be made only when sub clock is oscillating.
8. State transitions within the same mode (divide-by-n values changed or subclock oscillation turned on or off) are shown in the table below.

		Sub clock oscillating					Sub clock turned off				
		No division	Divided by 2	Divided by 4	Divided by 8	Divided by 16	No division	Divided by 2	Divided by 4	Divided by 8	Divided by 16
Sub clock oscillating	No division		(4)	(5)	(7)	(6)	(1)	--	--	--	--
	Divided by 2	(3)		(5)	(7)	(6)	--	(1)	--	--	--
	Divided by 4	(3)	(4)		(7)	(6)	--	--	(1)	--	--
	Divided by 8	(3)	(4)	(5)		(6)	--	--	--	(1)	--
	Divided by 16	(3)	(4)	(5)	(7)		--	--	--	--	(1)
Sub clock turned off	No division	(2)	--	--	--	--		(4)	(5)	(7)	(6)
	Divided by 2	--	(2)	--	--	--	(3)		(5)	(7)	(6)
	Divided by 4	--	--	(2)	--	--	(3)	(4)		(7)	(6)
	Divided by 8	--	--	--	(2)	--	(3)	(4)	(5)		(6)
	Divided by 16	--	--	--	--	(2)	(3)	(4)	(5)	(7)	

--: Cannot transit

9. ( ) : setting method. Refer to following table.

	Setting	Operation
(1)	CM04 = 0	Sub clock turned off
(2)	CM04 = 1	Sub clock oscillating
(3)	CM06 = 0, CM17 = 0 , CM16 = 0	CPU clock no division mode
(4)	CM06 = 0, CM17 = 0 , CM16 = 1	CPU clock division by 2 mode
(5)	CM06 = 0, CM17 = 1 , CM16 = 0	CPU clock division by 4 mode
(6)	CM06 = 0, CM17 = 1 , CM16 = 1	CPU clock division by 16 mode
(7)	CM06 = 1	CPU clock division by 8 mode
(8)	CM07 = 0	Main clock, PLL clock, or on-chip oscillator clock selected
(9)	CM07 = 1	Sub clock selected
(10)	CM05 = 0	Main clock oscillating
(11)	CM05 = 1	Main clock turned off
(12)	PLC07 = 0, CM11 = 0	Main clock selected
(13)	PLC07 = 1, CM11 = 1	PLL clock selected
(14)	CM21 = 0	Main clock or PLL clock selected
(15)	CM21 = 1	On-chip oscillator clock selected
(16)	CM10 = 1	Transition to stop mode
(17)	wait instruction	Transition to wait mode
(18)	Hardware interrupt	Exit stop mode or wait mode

CM04, CM05, CM06, CM07 : Bits in the CM0 register
CM10, CM11, CM16, CM17 : Bits in the CM1 register
CM20, CM21 : Bits in the CM2 register
PLC07 : Bit in the PLC0 register

9.2 Interrupts and Interrupt Vector

One interrupt vector consists of 4 bytes. Set the start address of each interrupt routine in the respective interrupt vectors. When an interrupt request is accepted, the CPU branches to the address set in the corresponding interrupt vector. Figure 9.2.1 shows the interrupt vector.

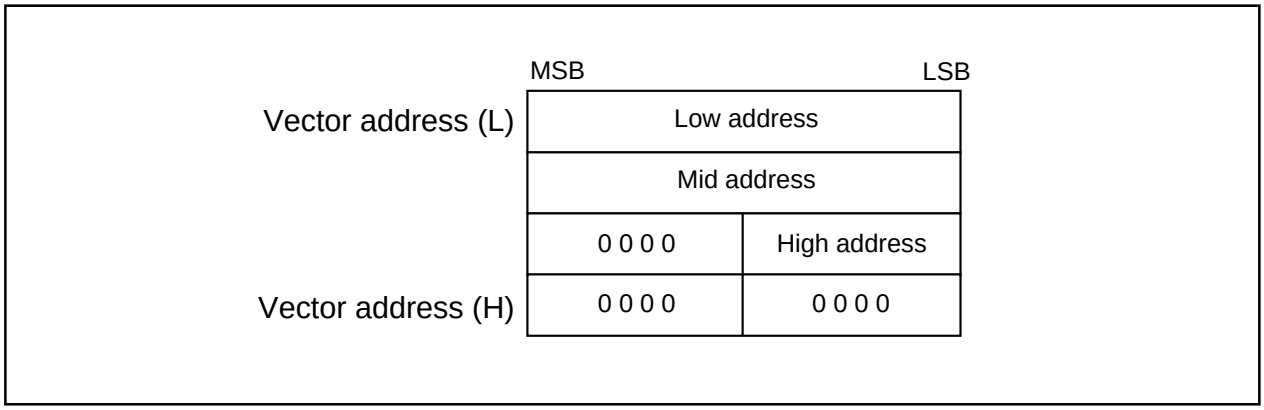


Figure 9.2.1. Interrupt Vector

9.2.1 Fixed Vector Tables

The fixed vector tables are allocated to the addresses from FFFDC16 to FFFFF16. Table 9.2.1.1 lists the fixed vector tables. In the flash memory version of microcomputer, the vector addresses (H) of fixed vectors are used by the ID code check function. For details, refer to the section **17.3 Flash Memory Rewrite Disabling Function**.

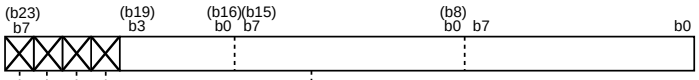
Table 9.2.1.1. Fixed Vector Tables

Interrupt source	Vector table addresses Address (L) to address (H)	Remarks	Reference
Undefined instruction	FFFDC16 to FFFDF16	Interrupt on UND instruction	M16C/60, M16C/20 serise software maual
Overflow	FFFE016 to FFFE316	Interrupt on INTO instruction	
BRK instruction	FFFE416 to FFFE716	If the contents of address FFFE716 is FF16, program execution starts from the address shown by the vector in the relocatable vector table.	
Address match	FFFE816 to FFFEB16		Address match interrupt
Single step (1)	FFFEC16 to FFFEF16		
Watchdog timer Oscillation stop and re-oscillation detection Voltage down detection	FFFF016 to FFFF316		Watchdog timer Clock generating circuit Voltage detection circuit
$\overline{\text{DBC}}$ (1)	FFFF416 to FFFF716		
$\overline{\text{NMI}}$	FFFF816 to FFFFB16		$\overline{\text{NMI}}$ interrupt
Reset (2)	FFFFC16 to FFFFF16		Reset

NOTES:

1. Do not normally use this interrupt because it is provided exclusively for use by development tools.
2. The b3 to b0 in address 0FFFFF16 are reserve bits. Set these bits to “11112”.

DMAi source pointer (i = 0, 1) (1)

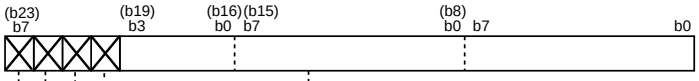


Symbol	Address	After reset
SAR0	0022_{16} to 0020_{16}	Indeterminate
SAR1	0032_{16} to 0030_{16}	Indeterminate

Function	Setting range	RW
Set the source address of transfer	00000_{16} to FFFFF$_{16}$	RW
Nothing is assigned. When write, set “0”. When read, these contents are “0”.		—

NOTE:
1. If the DSD bit in the DMiCON register is “0” (fixed), this register can only be written to when the DMAE bit in the DMiCON register is “0” (DMA disabled). If the DSD bit is set to “1” (forward direction), this register can be written to at any time. If the DSD bit is set to “1” and the DMAE bit is “1” (DMA enabled), the DMAi forward address pointer can be read from this register. Otherwise, the value written to it can be read.

DMAi destination pointer (i = 0, 1)(1)

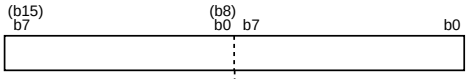


Symbol	Address	After reset
DAR0	0026_{16} to 0024_{16}	Indeterminate
DAR1	0036_{16} to 0034_{16}	Indeterminate

Function	Setting range	RW
Set the destination address of transfer	00000_{16} to FFFFF$_{16}$	RW
Nothing is assigned. When write, set “0”. When read, these contents are “0”.		—

NOTE:
1. If the DAD bit in the DMiCON register is “0” (fixed), this register can only be written to when the DMAE bit in the DMiCON register is “0”(DMA disabled).If the DAD bit is set to “1” (forward direction), this register can be written to at any time. If the DAD bit is set to “1” and the DMAE bit is “1” (DMA enabled), the DMAi forward address pointer can be read from this register. Otherwise, the value written to it can be read.

DMAi transfer counter (i = 0, 1)



Symbol	Address	After reset
TCR0	0029_{16}, 0028_{16}	Indeterminate
TCR1	0039_{16}, 0038_{16}	Indeterminate

Function	Setting range	RW
Set the transfer count minus 1. The written value is stored in the DMAi transfer counter reload register, and when the DMAE bit in the DMiCON register is set to “1” (DMA enabled) or the DMAi transfer counter underflows when the DMASL bit in the DMiCON register is “1” (repeat transfer), the value of the DMAi transfer counter reload register is transferred to the DMAi transfer counter. When read, the DMAi transfer counter is read.	0000_{16} to FFFF$_{16}$	RW

Figure 11.4 SAR0 and SAR1, DAR0 and DAR1, TCR0 and TCR1 Registers

12.1.3. One-shot Timer Mode

In one-shot timer mode, the timer is activated only once by one trigger. (See Table 12.1.3.1.) When the trigger occurs, the timer starts up and continues operating for a given period. Figure 12.1.3.1 shows the TAiMR register in one-shot timer mode.

Table 12.1.3.1. Specifications in One-shot Timer Mode

Item	Specification
Count source	f_1, f_2, f_8, f_{32}, f_{C32}
Count operation	• Down-count • When the counter reaches 0000_{16}, it stops counting after reloading a new value • If a trigger occurs when counting, the timer reloads a new count and restarts counting
Divide ratio	1/n n : set value of TAi register 0000_{16} to $FFFF_{16}$ However, the counter does not work if the divide-by-n value is set to 0000_{16}.
Count start condition	TAiS bit in the TABSR register is set to “1” (start counting) and one of the following triggers occurs. • External trigger input from the TAi_{IN} pin • Timer B2 overflow or underflow, timer Aj (j=i-1, except j=4 if i=0) overflow or underflow, timer Ak (k=i+1, except k=0 if i=4) overflow or underflow • The TAiOS bit in the ONSF register is set to “1” (= timer starts)
Count stop condition	• When the counter is reloaded after reaching “0000_{16}” • TAiS bit is set to “0” (= stop counting)
Interrupt request generation timing	When the counter reaches “0000_{16}”
TAi_{IN} pin function	I/O port or trigger input
TAi_{OUT} pin function	I/O port or pulse output
Read from timer	An indeterminate value is read by reading TAi register
Write to timer	• When not counting and until the 1st count source is input after counting start Value written to TAi register is written to both reload register and counter • When counting (after 1st count source input) Value written to TAi register is written to only reload register (Transferred to counter when reloaded next)
Select function	• Pulse output function The timer outputs a low when not counting and a high when counting.

12.2.3 Pulse Period and Pulse Width Measurement Mode

In pulse period and pulse width measurement mode, the timer measures pulse period or pulse width of an external signal (see Table 12.2.3.1). Figure 12.2.3.1 shows TBiMR register in pulse period and pulse width measurement mode. Figure 12.2.3.2 shows the operation timing when measuring a pulse period. Figure 12.2.3.3 shows the operation timing when measuring a pulse width.

Table 12.2.3.1 Specifications in Pulse Period and Pulse Width Measurement Mode

Item	Specification
Count source	f_1, f_2, f_8, f_{32}, f_{C32}
Count operation	• Up-count • Counter value is transferred to reload register at an effective edge of measurement pulse. The counter value is set to "0000_{16}" to continue counting.
Count start condition	Set TBiS (i=0 to 2) bit(3) to "1" (= start counting)
Count stop condition	Set TBiS bit to "0" (= stop counting)
Interrupt request generation timing	• When an effective edge of measurement pulse is input(1) • Timer overflow. When an overflow occurs, MR3 bit in the TBiMR register is set to "1" (overflowed) simultaneously. MR3 bit is cleared to "0" (no overflow) by writing to TBiMR register at the next count timing or later after MR3 bit was set to "1". At this time, make sure TBiS bit is set to "1" (start counting).
TBi_{IN} pin function	Measurement pulse input
Read from timer	Contents of the reload register (measurement result) can be read by reading TBi register(2)
Write to timer	Value written to TBi register is written to neither reload register nor counter

NOTES:
1. Interrupt request is not generated when the first effective edge is input after the timer started counting.
2. Value read from TBi register is indeterminate until the second valid edge is input after the timer starts counting.
3. The TB0S to TB2S bits are assigned to the bit 5 to bit 7 in the TABSR register.

Timer Bi mode register (i=0 to 2)

b7	b6	b5	b4	b3	b2	b1	b0
						1	0

Symbol: TB0MR to TB2MR — Address: $039B_{16}$ to $039D_{16}$ — After reset: $00XX0000_2$

Bit symbol	Bit name	Function	RW
TMOD0	Operation mode select bit	b1 b0 1 0 : Pulse period / pulse width measurement mode	RW
TMOD1			RW
MR0	Measurement mode select bit	b3 b2 0 0 : Pulse period measurement (Measurement between a falling edge and the next falling edge of measured pulse) 0 1 : Pulse period measurement (Measurement between a rising edge and the next rising edge of measured pulse) 1 0 : Pulse width measurement (Measurement between a falling edge and the next rising edge of measured pulse and between a rising edge and the next falling edge) 1 1 : Must not be set.	RW
MR1			RW
MR2	TB0MR register Must be set to "0" in pulse period and pulse width measurement mode		RW
	TB1MR, TB2MR registers Nothing is assigned. When write, set to "0". When read, its content turns out to be indeterminate.		—
MR3	Timer Bi overflow flag (1)	0 : Timer did not overflow 1 : Timer has overflowed	RO
TCK0	Count source select bit	b7 b6 0 0 : f_1 or f_2 0 1 : f_8 1 0 : f_{32} 1 1 : f_{C32}	RW
TCK1			RW

NOTE:
1. This flag is indeterminate after reset. When the TBiS bit is set to "1" (start counting), the MR3 bit is cleared to "0" (no overflow) by writing to the TBiMR register at the next count timing or later after the MR3 bit was set to "1" (overflowed). The MR3 bit cannot be set to "1" in a program. The TB0S to TB2S bits are assigned to the bit 5 to bit 7 in the TABSR register.

Figure 12.2.3.1 TBiMR Register in Pulse Period and Pulse Width Measurement Mode

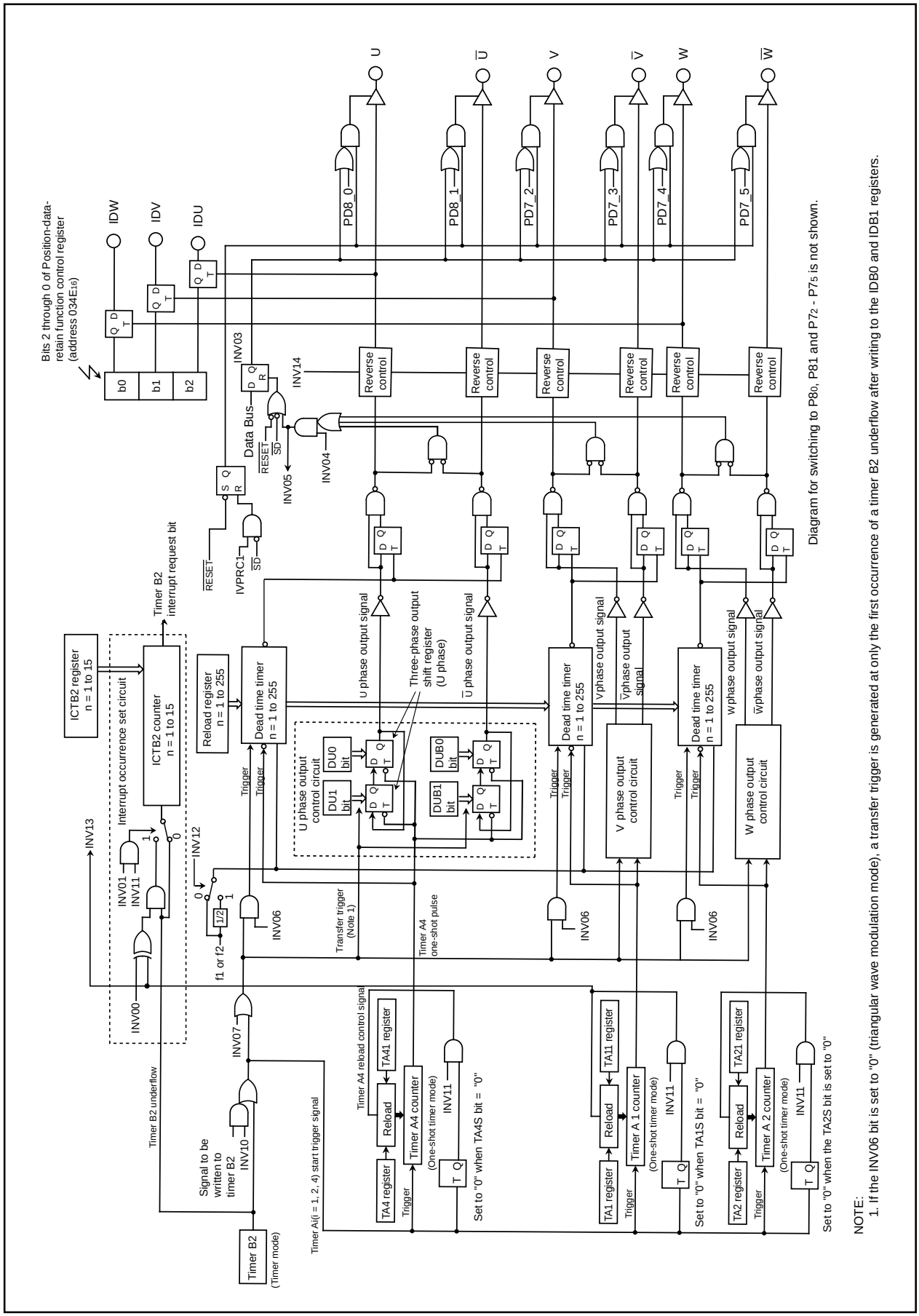
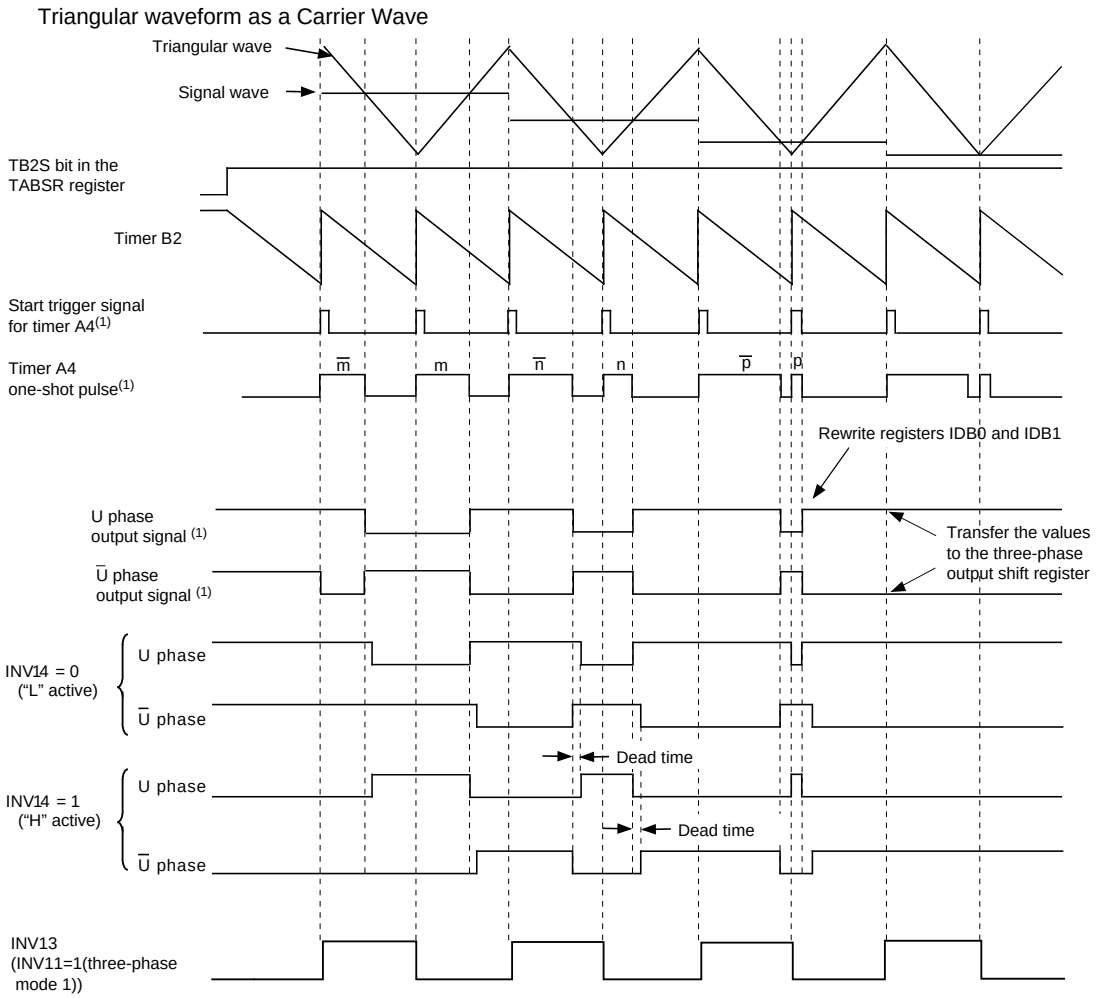


Figure 12.3.1. Three-phase Motor Control Timer Functions Block Diagram

The three-phase motor control timer function is enabled by setting the INV02 bit in the VC0 register to “1”. When this function is on, timer B2 is used to control the carrier wave, and timers A4, A1 and A2 are used to control three-phase PWM outputs (U, $\overline{U}$, V, $\overline{V}$, W and $\overline{W}$). The dead time is controlled by a dedicated dead-time timer. Figure 12.3.9 shows the example of triangular modulation waveform, and Figure 12.3.10 shows the example of sawtooth modulation waveform.



NOTE:
1. Internal signals. See **Figure 12.3.1**.

The above applies under the following conditions:
INVC0 = 00XX11XX$_2$ (X varies depending on each system) and INVC1 = 010XXXX0$_2$.
Examples of PWM output change are:

(1)When INV11 = 1 (three-phase mode 1)
- INV01 = 0 and ICTB2 = 2$_{16}$ (the timer B2 interrupt is generated every two times the timer B2 underflows), or INV01 = 1, INV00 = 1, and ICTB2=1$_{16}$ (the timer B2 interrupt is generated at the falling edge of the timer A1 reload control signal.)
- Default value of the timer: TA41 = $\overline{m}$, TA4 = m. Registers TA4 and TA41 are changed whenever the timer B2 interrupt is generated. First time, TA41 = $\overline{n}$, TA4 = n. Second time, TA41 = $\overline{p}$, TA4 = p.
- Default values of registers IDB0 and IDB1: DU0 = 1, DUB0 = 0, DU1 = 0, DUB1 = 1. They are changed to DU0 = 1, DUB0 = 0, DU1= 1 and DUB1 = 0 when the third timer B2 interrupt is generated.

(2)When INV11 = 0 (three-phase mode 0)
- INV01 = 0, ICTB2 = 1$_{16}$ (the timer B2 interrupt is generated whenever timer B2 underflows)
- Default value of the timer: TA4 = $\overline{m}$. The TA4 register is changed whenever the timer B2 interrupt is generated. First time: TA4 = m. Second tim:, TA4 = $\overline{n}$. Third time: TA4 = n. Fourth time: TA4 = $\overline{p}$. Fifth time: TA4 = p.
- Default values of registers IDB0 and IDB1: DU0 = 1, DUB0 = 0, DU1 = 0, DUB1 = 1. They are changed to DU0 = 1, DUB0 = 0, DU1 = 1, and DUB1 = 0 when the sixth timer B2 interrupt is generated.

The value written to registers TA4 and TA41 becomes effective at the rising edge of the timer A1 reload control signal.

Figure 12.3.9. Triangular Wave Modulation Operation

Table 13.1.3.4. I²C bus Mode Functions

Function	Clock synchronous serial I/O mode (SMD2 to SMD0 = 001₂, IICM = 0)	I²C bus mode (SMD2 to SMD0 = 010₂, IICM = 1)			
		IICM2 = 0 (NACK/ACK interrupt)		IICM2 = 1 (UART transmit/ receive interrupt)	
		CKPH = 0 (No clock delay)	CKPH = 1 (Clock delay)	CKPH = 0 (No clock delay)	CKPH = 1 (Clock delay)
Factor of interrupt number 10 (1) (Refer to **Fig.13.1.3.2**.)	————	Start condition detection or stop condition detection (Refer to **Figure 13.1.3.2.1. STSPSEL Bit Function**)			
Factor of interrupt number 15 (1) (Refer to **Fig.13.1.3.2**.)	UART2 transmission Transmission started or completed (selected by U2IRS)	No acknowledgment detection (NACK) Rising edge of SCL₂ 9th bit		UART2 transmission Rising edge of SCL₂ 9th bit	UART2 transmission Falling edge of SCL₂ next to the 9th bit
Factor of interrupt number 16 (1) 1(Refer to **Fig.13.1.3.2**.)	UART2 reception When 8th bit received CKPOL = 0 (rising edge) CKPOL = 1 (falling edge)	Acknowledgment detection (ACK) Rising edge of SCL₂ 9th bit		UART2 transmission Falling edge of SCL₂ 9th bit	
Timing for transferring data from the UART reception shift register to the U2RB register	CKPOL = 0 (rising edge) CKPOL = 1 (falling edge)	Rising edge of SCL₂ 9th bit		Falling edge of SCL₂ 9th bit	Falling and rising edges of SCL₂ 9th bit
UART2 transmission output delay	Not delayed	Delayed			
Functions of P7₀ pin	TxD2 output	SDA₂ input/output			
Functions of P7₁ pin	RxD2 input	SCL₂ input/output			
Functions of P7₂ pin	CLK2 input or output selected	———— (Cannot be used in I²C mode)			
Noise filter width	15ns	200ns			
Read RxD2 and SCL₂ pin levels	Possible when the corresponding port direction bit = 0	Always possible no matter how the corresponding port direction bit is set			
Initial value of TxD2 and SDA₂ outputs	CKPOL = 0 (H) CKPOL = 1 (L)	The value set in the port register before setting I²C bus mode (2)			
Initial and end values of SCL₂	————	H	L	H	L
DMA1 factor (Refer to **Fig. 14.1.3.2.**)	UART2 reception	Acknowledgment detection (ACK)		UART2 reception Falling edge of SCL₂ 9th bit	
Store received data	1st to 8th bits are stored in U2RB register bit 0 to bit 7	1st to 8th bits are stored in U2RB register bit 7 to bit 0		1st to 7th bits are stored in U2RB register bit 6 to bit 0, with 8th bit stored in U2RB register bit 8	1st to 8th bits are stored in U2RB register bit 7 to bit 0 (3)
Read received data	U2RB register status is read directly as is				Read U2RB register Bit 6 to bit 0 as bit 7 to bit 1, and bit 8 as bit 0 (4)

NOTES:

1. If the source or cause of any interrupt is changed, the IR bit in the interrupt control register for the changed interrupt may inadvertently be set to 1 (interrupt requested). (Refer to “Notes on interrupts” in Usage Notes) If one of the bits shown below is changed, the interrupt source, the interrupt timing, etc. change. Therefore, always be sure to clear the IR bit to 0 (interrupt not requested) after changing those bits. SMD2 to SMD0 bits in the U2MR register, IICM bit in the U2SMR register, IICM2 bit in the U2SMR2 register, CKPH bit in the U2SMR3 register
2. Set the initial value of SDA₂ output while the SMD2 to SMD0 bits in the U2MR register is set to ‘000₂’ (serial I/O disabled).
3. Second data transfer to U2RB register (Rising edge of SCL₂ 9th bit)
4. First data transfer to U2RB register (Falling edge of SCL₂ 9th bit)

13.1.4 Special Mode 2 (UART2)

Multiple slaves can be serially communicated from one master. Transfer clock polarity and phase are selectable. Table 13.1.4.1 lists the specifications of Special Mode 2. Table 13.1.4.2 lists the registers used in Special Mode 2 and the register values set. Figure 13.1.4.1 shows communication control example for Special Mode 2.

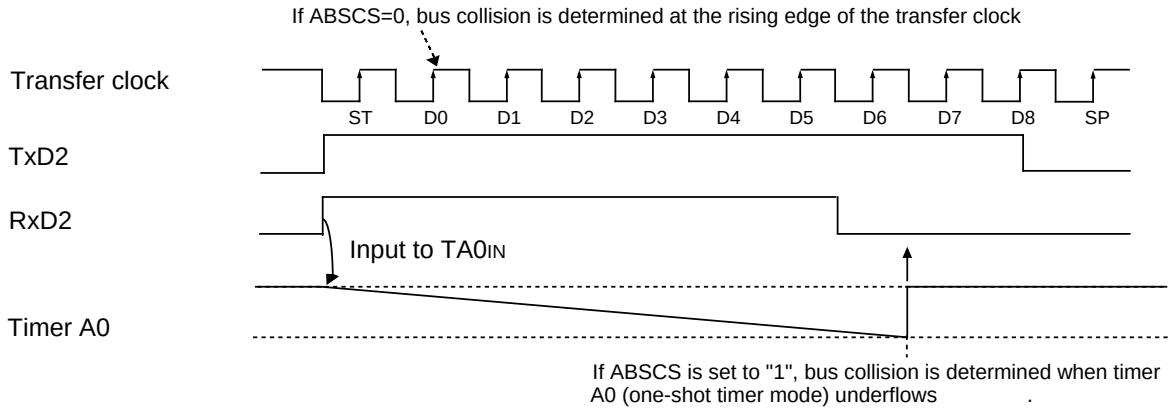
Table 13.1.4.1. Special Mode 2 Specifications

Item	Specification
Transfer data format	• Transfer data length: 8 bits
Transfer clock	• Master mode The CKDIR bit in the U2MR register is set to "0" (internal clock) : fj/ (2(n+1)) fj = f_{1SIO}, f_{2SIO}, f_{8SIO}, f_{32SIO}. n: Setting value of U2BRG register 00_{16} to FF_{16} • Slave mode The CKDIR bit is set to "1" (external clock selected) : Input from CLK2 pin
Transmit/receive control	Controlled by input/output ports
Transmission start condition	• Before transmission can start, the following requirements must be met (1) – The TE bit in the U2C1 register is set to "1" (transmission enabled) – The TI bit in the U2C1 register is set to "0" (data present in U2TB register)
Reception start condition	• Before reception can start, the following requirements must be met (1) – The RE bit in the U2C1 register is set to "1" (reception enabled) – The TE bit in the U2C1 register is set to "1" (transmission enabled) – The TI bit in the U2C1 register is set to "0" (data present in the U2TB register)
Interrupt request generation timing	• While transmitting, one of the following conditions can be selected – The U2IRS bit in the U2C1 register is set to "0" (transmit buffer empty): when transferring data from the U2TB register to the UART2 transmit register (at start of transmission) – The U2IRS bit is set to "1" (transfer completed): when the serial I/O finished sending data from the UART2 transmit register • While receiving When transferring data from the UART2 receive register to the U2RB register (at completion of reception)
Error detection	• Overrun error (2) This error occurs if the serial I/O started receiving the next data before reading the U2RB register and received the 7th bit of the next data
Select function	• Clock phase setting Selectable from four combinations of transfer clock polarities and phases

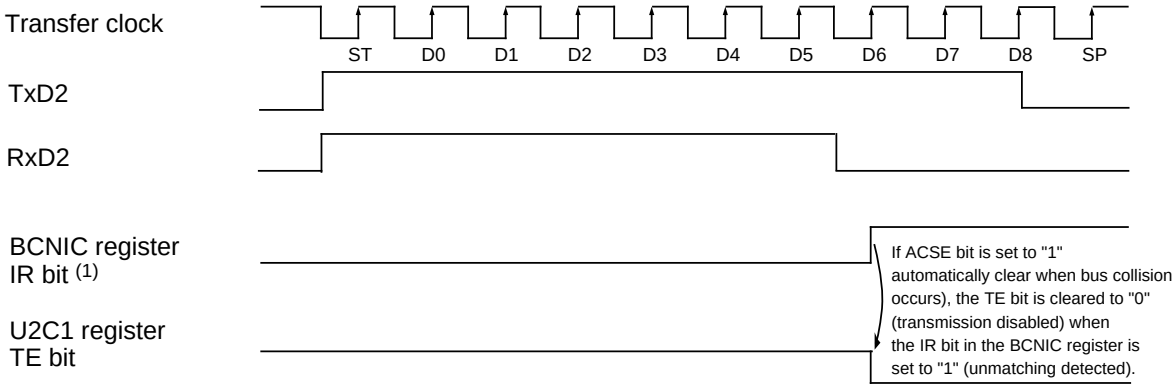
NOTES:

1. When an external clock is selected, the conditions must be met while if the CKPOL bit in the U2C0 register "0" (transmit data output at the falling edge and the receive data taken in at the rising edge of the transfer clock), the external clock is in the high state; if the CKPOL bit in the U2C0 register "1" (transmit data output at the rising edge and the receive data taken in at the falling edge of the transfer clock), the external clock is in the low state.
2. If an overrun error occurs, bits 8 to 0 in UiRB register are undefined. The IR bit in the SiRIC register remains unchanged.

(1) The ABSCS bit in the U2SMR register (bus collision detect sampling clock select)

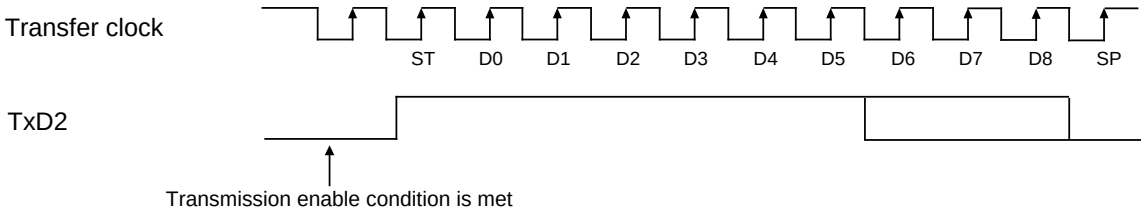


(2) The ACSE bit in the U2SMR register (auto clear of transmit enable bit)

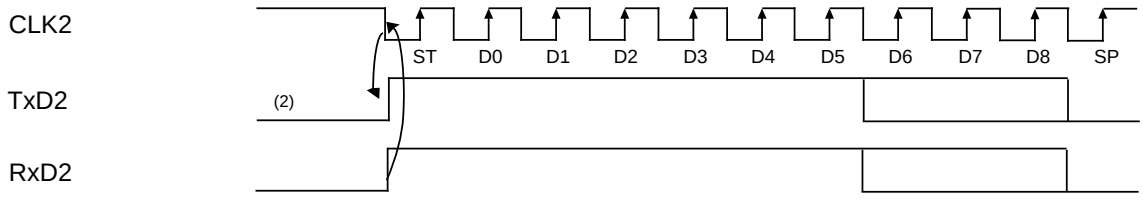


(3) The SSS bit in the U2SMR register (Transmit start condition select)

If SSS bit is set to "0", the serial I/O starts sending data one transfer clock cycle after the transmission enable condition is met.



If SSS bit = 1, the serial I/O starts sending data at the rising edge (1) of RxD2



NOTES:
1. The falling edge of RxD2 when the IOPOL is set to "0"; the rising edge of RxD2 when the IOPOL is set to "1".
2. The transmit condition must be met before the falling edge (Note 1) of RxD.

This diagram applies to the case where the IOPOL is set to "1" (reversed)

Figure 13.1.5.1. Bus Collision Detect Function-Related Bits

14.1.5 Repeat Sweep Mode 1

In repeat sweep mode 1, analog voltages applied to the all selected pins are converted to a digital code, with mainly used in the selected pins. Table 14.1.5.1 shows the repeat sweep mode 1 specifications. Figure 14.1.5.1 shows the operation example in repeat sweep mode 1. Figure 14.1.5.2 shows the ADCON0 to ADCON2 registers in repeat sweep mode 1.

Table 14.1.5.1 Repeat Sweep Mode 1 Specifications

Item	Specification
Function	The SCAN1 to SCAN0 bits in the ADCON1 register and the ADGSEL1 to ADGSEL0 bits in the ADCON2 register mainly select pins. Analog voltage applied to the all selected pins is repeatedly converted to a digital code Example : When selecting AN_0 Analog voltage is converted to a digital code in the following order $AN_0 \rightarrow AN_1 \rightarrow AN_0 \rightarrow AN_2 \rightarrow AN_0 \rightarrow AN_3$, and so on.
A/D Conversion Start Condition	• When the TRG bit in the ADCON0 register is “0” (software trigger) Set the ADST bit in the ADCON0 register to “1” (A/D conversion started) • When the TRG bit in the ADCON0 register is “1” (hardware trigger) The $\overline{ADTRG}$ pin input changes state from “H” to “L” after setting the ADST bit to “1” (A/D conversion started)
A/D Conversion Stop Condition	Set the ADST bit to “0” (A/D conversion halted)
Interrupt Request Generation Timing	None generated
Analog Input Pins Mainly Used in A/D Conversions	Select from AN_0 (1 pins), AN_0 to AN_1 (2 pins), AN_0 to AN_2 (3 pins), AN_0 to AN_3 (4 pins) (1)
Readout of A/D Conversion Result	Readout one of the AD0 to AD7 registers that corresponds to the selected pin

NOTE:

1. AN_{30} to AN_{32} can be used in the same way as AN_0 to AN_7. However, all input pins need to belong to the same group.

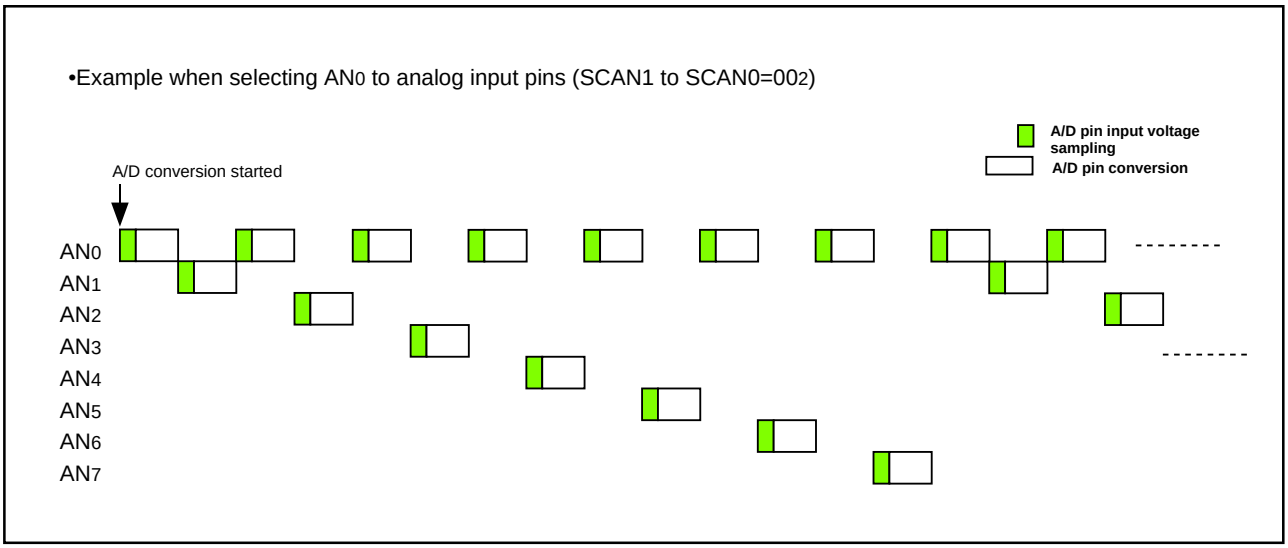
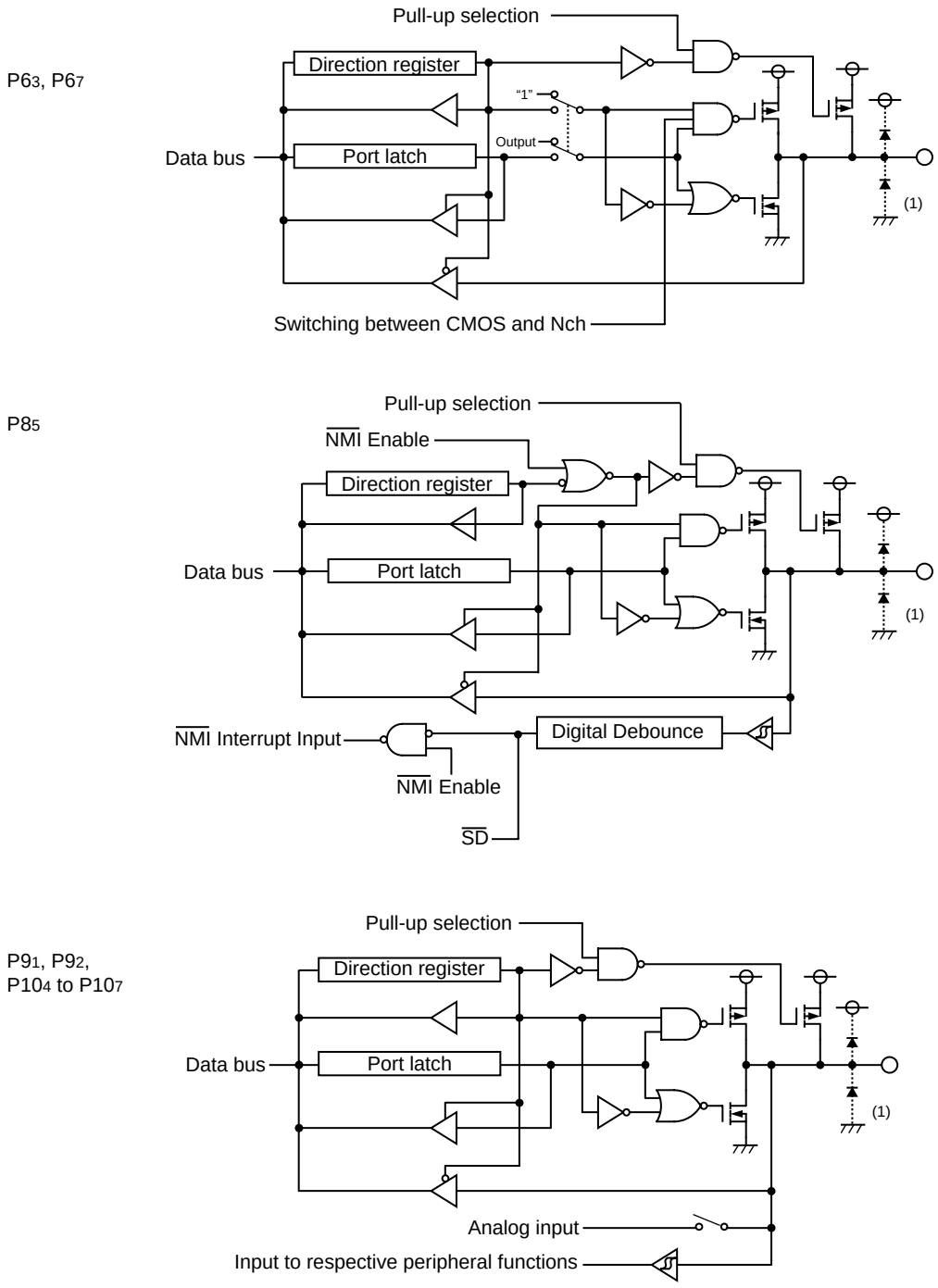


Figure 14.1.5.1 Operation Example in Repeat Sweep Mode 1



NOTE:
1. ⋯◀⋯ symbolizes a parasitic diode.
Make sure the input voltage on each port will not exceed Vcc.

Figure 16.3. I/O Ports (3)

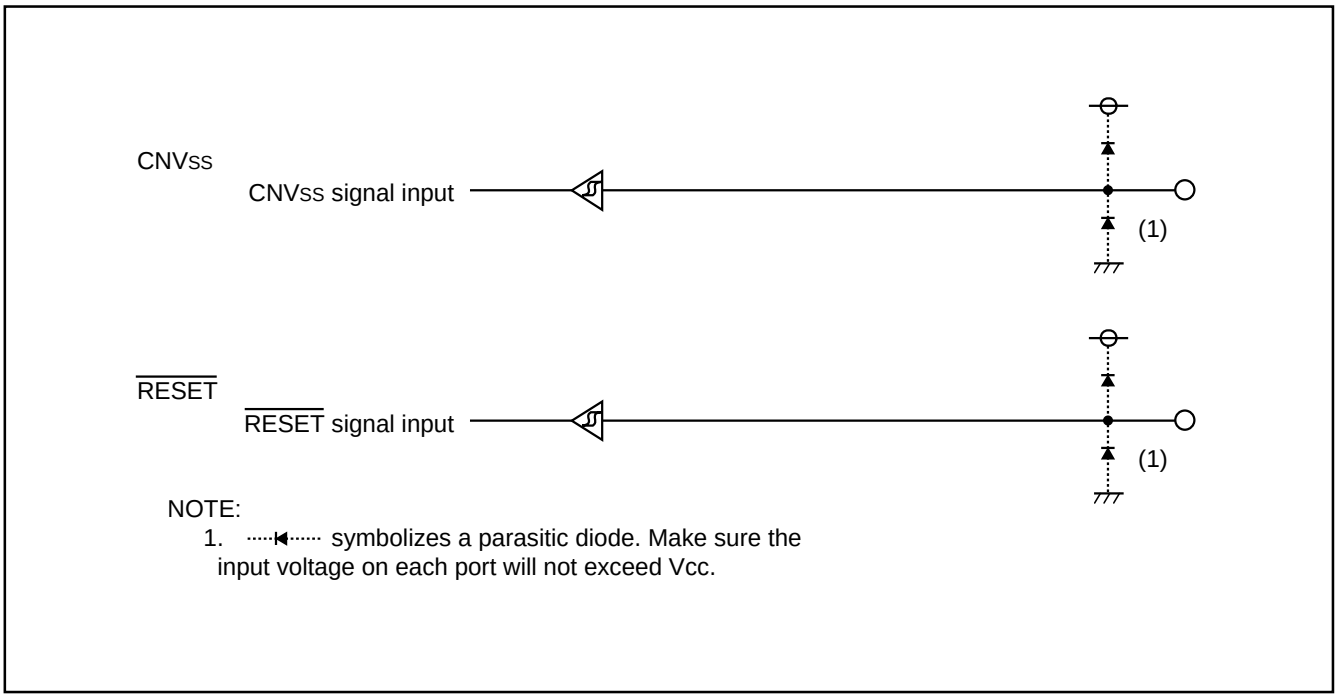


Figure 16.5. I/O Pins

17.2 Memory Map

The flash memory contains the user ROM area and the boot ROM area (reserved area). Figures 17.2.1 to 17.2.3 show the flash memory block diagram. The user ROM area has space to store the microcomputer operation program in single-chip mode and a separate 2-Kbyte space as the block A and B.

The user ROM area is divided into several blocks. The user ROM area can be rewritten in CPU rewrite, standard serial input/output, and parallel input/output modes. However, if block 0 and 1 are rewritten in CPU rewrite mode, setting the FMR02 bit in the FMR0 register to “1” (block 0, 1 rewrite enabled) and the FMR16 bit in the FMR1 register to “1”(blocks 0 to 3 rewrite enabled) enable rewriting. Also, if blocks 2 to 3 are rewritten in CPU rewrite mode, setting the FMR16 bit in the FMR1 register to “1” (blocks 0 to 3 rewrite enabled) enables writing. Setting the PM10 bit in the PM1 register to “1”(data area access enabled) for block A and B enables to use.

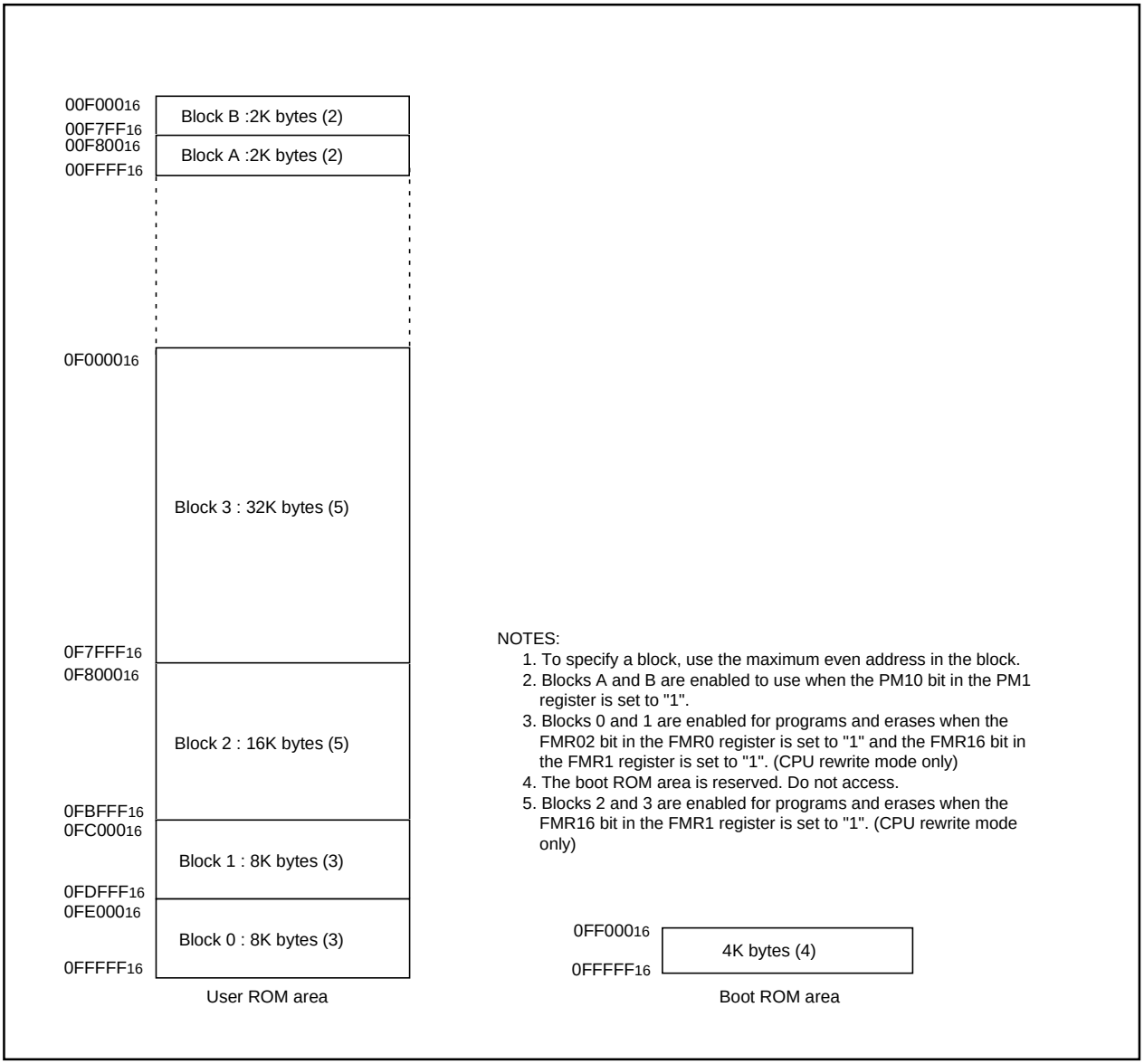
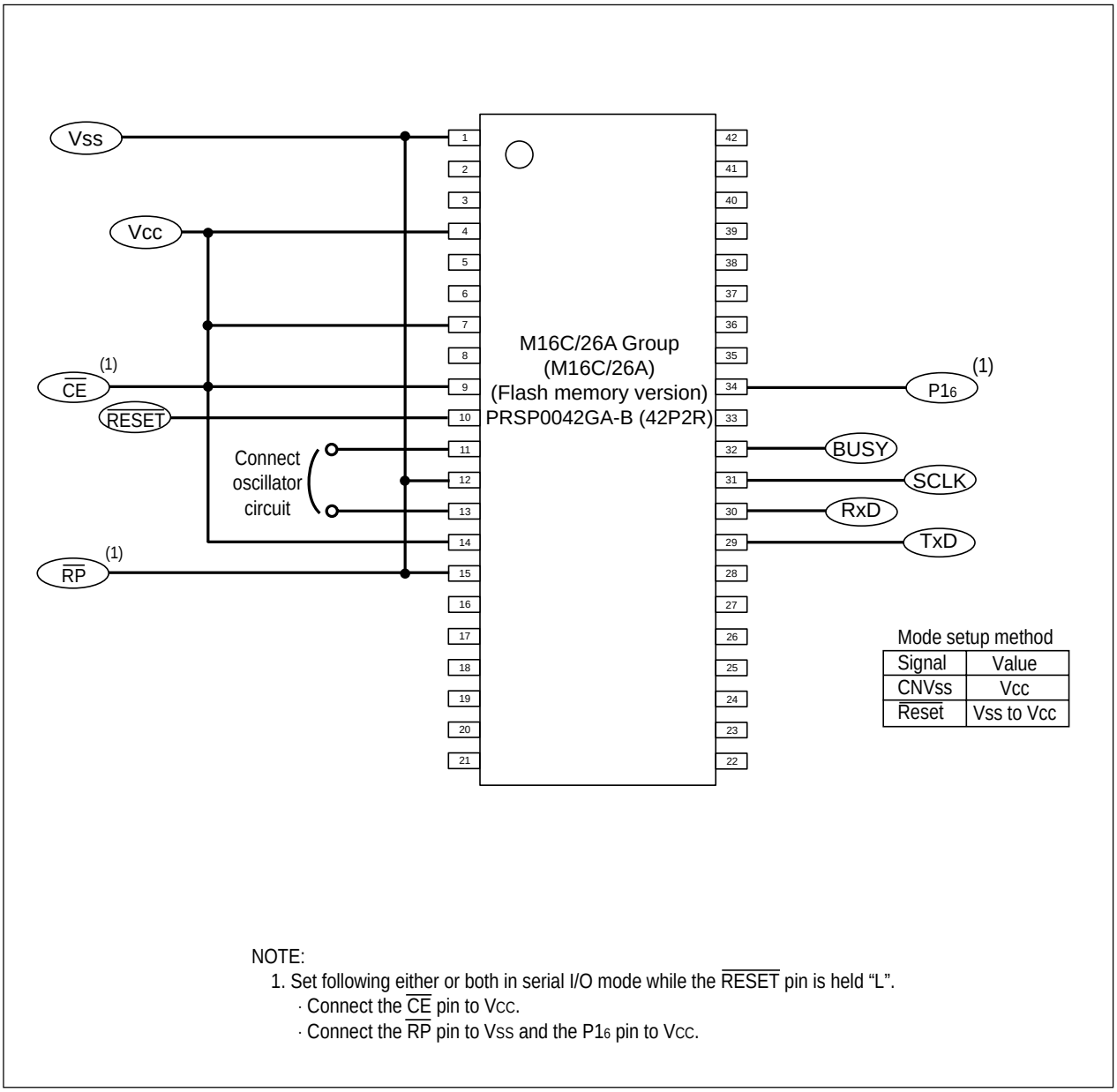


Figure 17.2.1. Flash Memory Block Diagram (ROM capacity 64K byte)



NOTE:
1. Set following either or both in serial I/O mode while the RESET pin is held "L".
   · Connect the CE pin to Vcc.
   · Connect the RP pin to Vss and the P16 pin to Vcc.

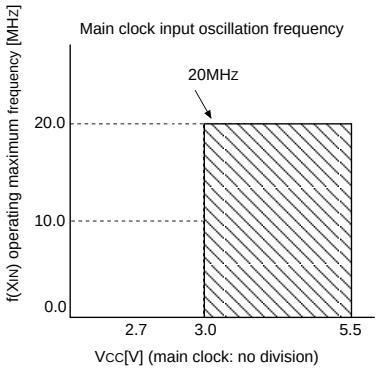
Figure 17.9.1. Pin Connections for Serial I/O Mode (1)

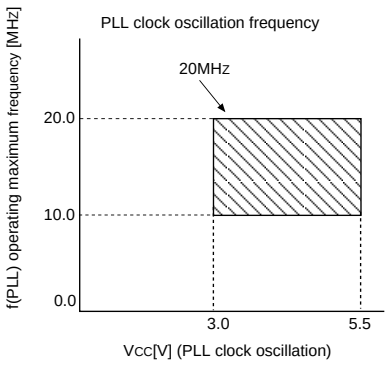
Table 18.39. Recommended Operating Conditions (1)

Symbol	Parameter		Standard			Unit
			Min.	Typ.	Max.	
V_{CC}	Supply Voltage		3.0		5.5	V
AV_{CC}	Analog Supply Voltage			V_{CC}		V
V_{SS}	Supply Voltage			0		V
AV_{SS}	Analog Supply Voltage			0		V
V_{IH}	Input High ("H") Voltage	$P1_5$ to $P1_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_3$, $P10_0$ to $P10_7$	0.7 V_{CC}		V_{CC}	V
		XIN, $\overline{RESET}$, CNVSS	0.8 V_{CC}		V_{CC}	V
V_{IL}	Input Low ("L") Voltage	$P1_5$ to $P1_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_3$, $P10_0$ to $P10_7$	0		0.3 V_{CC}	V
		XIN, $\overline{RESET}$, CNVSS	0		0.2V_{CC}	V
$I_{OH(peak)}$	Peak Output High ("H") Current	$P1_5$ to $P1_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_3$, $P10_0$ to $P10_7$			-10.0	mA
$I_{OH(avg)}$	Average Output High ("H") Current	$P1_5$ to $P1_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_3$, $P10_0$ to $P10_7$			-5.0	mA
$I_{OL(peak)}$	Peak Output Low ("L") Current	$P1_5$ to $P1_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_3$, $P10_0$ to $P10_7$			10.0	mA
$I_{OL(avg)}$	Average Output Low ("L") Current	$P1_5$ to $P1_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_3$, $P10_0$ to $P10_7$			5.0	mA
f(X_{IN})	Main Clock Input Oscillation Frequency(4)		0		20	MHz
f(X_{CIN})	Sub Clock Oscillation Frequency			32.768	50	kHz
f_1(ROC)	On-chip Oscillator Frequency 1		0.5	1	2	MHz
f2(ROC)	On-chip Oscillator Frequency 2		1	2	4	MHz
f3(ROC)	On-chip Oscillator Frequency 3		8	16	26	MHz
f(PLL)	PLL Clock Oscillation Frequency(4)		10		20	MHz
f(BCLK)	CPU Operation Clock Frequency		0		20	MHz
t_{SU}(PLL)	Wait Time to Stabilize PLL Frequency Synthesizer	V_{CC} = 5.0 V			20	ms
		V_{CC} = 3.0 V			50	ms

NOTES:

1. Referenced to V_{CC} = 3.0 to 5.5 V at Topr = -40 to 85 ° C unless otherwise specified.
2. The mean output current is the mean value within 100 ms.
3. The total $I_{OL(peak)}$ for all ports must be 80 mA or less. The total $I_{OH(peak)}$ for all ports must be -80 mA or less.
4. Relationship among main clock oscillation frequency, PLL clock oscillation frequency and supply voltage.





VCC = 5V

Timing Requirements

(VCC = 5V, VSS = 0V, at Topr = – 40 to 85°C unless otherwise specified)

Table 18.46. External Clock Input (XIN input)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t_c	External clock input cycle time	50		ns
$t_{w(H)}$	External clock input HIGH pulse width	20		ns
$t_{w(L)}$	External clock input LOW pulse width	20		ns
t_r	External clock rise time		9	ns
t_f	External clock fall time		9	ns

19.3 Power Control

1. When exiting stop mode by hardware reset, the device will startup using the on-chip oscillator.

2. Set the MR0 bit in the TAiMR register(i=0 to 4) to “0”(pulse is not output) to use the timer A to exit stop mode.

3. When entering wait mode, insert a JMP.B instruction before a WAIT instruction. Do not excute any instructions which can generate a write to RAM between the JMP.B and WAIT instructions. Disable the DMA transfers, if a DMA transfer may occur between the JMP.B and WAIT instructions. After the WAIT instruction, insert at least 4 NOP instructions. When entering wait mode, the instruction queue reads ahead the instructions following WAIT, and depending on timing, some of these may execute before the microcomputer enters wait mode.

Program example when entering wait mode

```
Program Example:    JMP.B   L1   ; Insert JMP.B instruction before WAIT instruction
                L1:
                    FSET    I    ;
                    WAIT         ; Enter wait mode
                    NOP          ; More than 4 NOP instructions
                    NOP
                    NOP
                    NOP
```

4. When entering stop mode, insert a JMP.B instruction immediately after executing an instruction which sets the CM10 bit in the CM1 register to “1”, and then insert at least 4 NOP instructions. When entering stop mode, the instruction queue reads ahead the instructions following the instruction which sets the CM10 bit to “1” (all clock stops), and, some of these may execute before the microcomputer enters stop mode or before the interrupt routine for returning from stop mode.

Program example when entering stop mode

```
Program Example:    FSET    I
                    BSET    CM10    ; Enter stop mode
                    JMP.B   L1      ; Insert JMP.B instruction
                L1:
                    NOP             ; More than 4 NOP instructions
                    NOP
                    NOP
                    NOP
```