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	M16C/60
Core Size	16-Bit
Speed	20MHz
Connectivity	CANbus, I²C, IEBus, SIO, UART/USART
Peripherals	DMA, POR, PWM, Voltage Detect, WDT
Number of I/O	55
Program Memory Size	96KB (96K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°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/m30291fahp-u3a

Figure 7.11 shows the state transition from normal operation mode to stop mode and wait mode. **Figure 7.12** shows the state transition in normal operation mode.

Table 7.7 shows a state transition matrix describing allowed transition and setting. The vertical line shows current state and horizontal line shows state after transition.

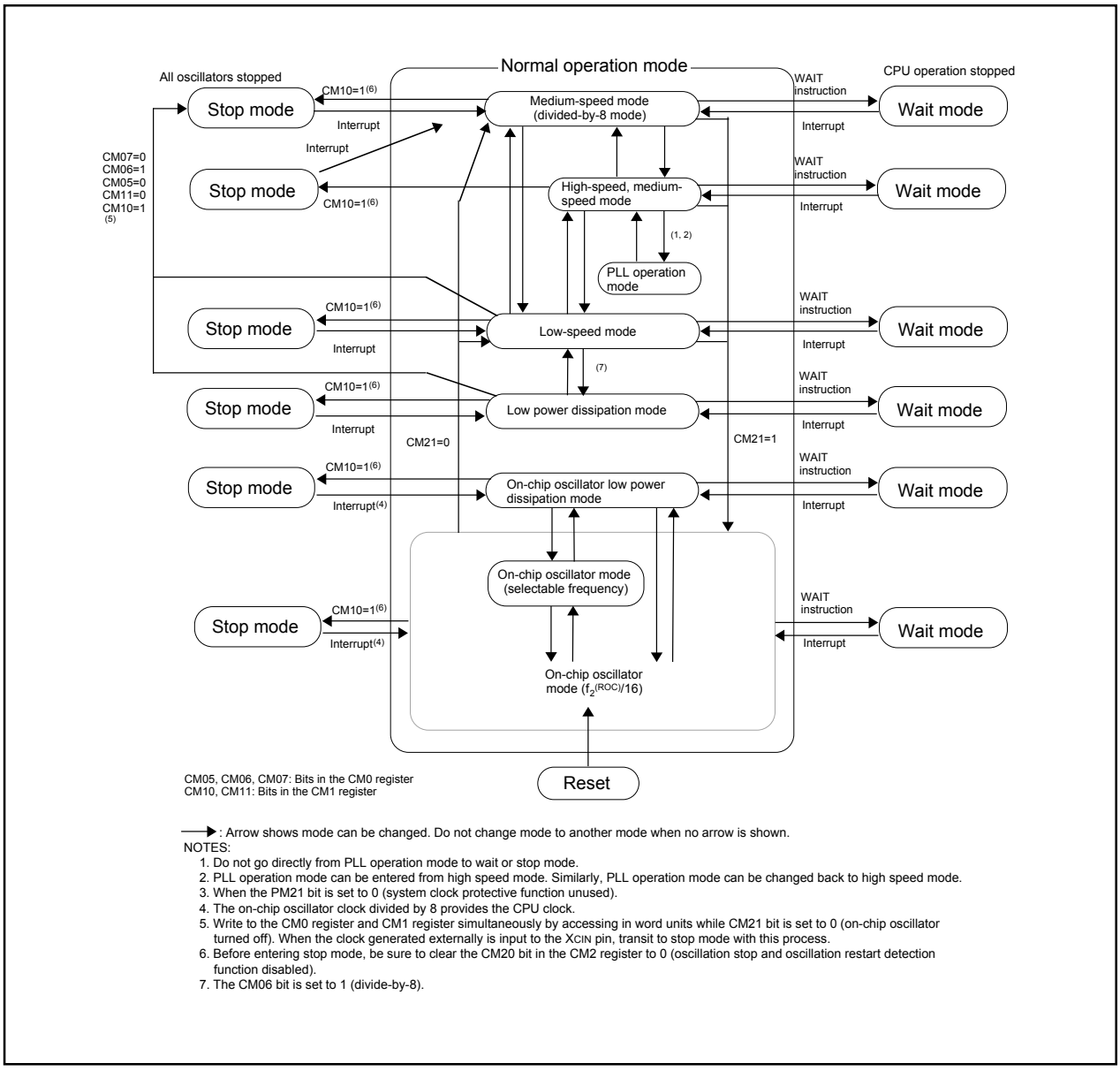


Figure 7.11 State Transition to Stop Mode and Wait Mode

9.4 Interrupt Sequence

An interrupt sequence (the device behavior from the instant an interrupt is accepted to the instant the interrupt routine is executed) is described here.

If an interrupt occurs during execution of an instruction, the processor determines its priority when the execution of the instruction is completed, and transfers control to the interrupt sequence from the next cycle. If an interrupt occurs during execution of either the SMOVB, SMOVF, SSTR or RMPA instruction, the processor temporarily suspends the instruction being executed, and transfers control to the interrupt sequence.

The CPU behavior during the interrupt sequence is described below. **Figure 9.5** shows time required for executing the interrupt sequence.

(1) The CPU gets interrupt information (interrupt number and interrupt request priority level) by reading the address 00000_{16}. Then it clears the IR bit for the corresponding interrupt to 0 (interrupt not requested).

(2) The FLG register immediately before entering the interrupt sequence is saved to the CPU's internal temporary register(Note).

(3) The I, D and U flags in the FLG register become as follows:
The I flag is cleared to 0 (interrupts disabled).
The D flag is cleared to 0 (single-step interrupt disabled).
The U flag is cleared to 0 (ISP selected).
However, the U flag does not change state if an INT instruction for software interrupt Nos. 32 to 63 is executed.

(4) The CPU's internal temporary register(1) is saved to the stack.

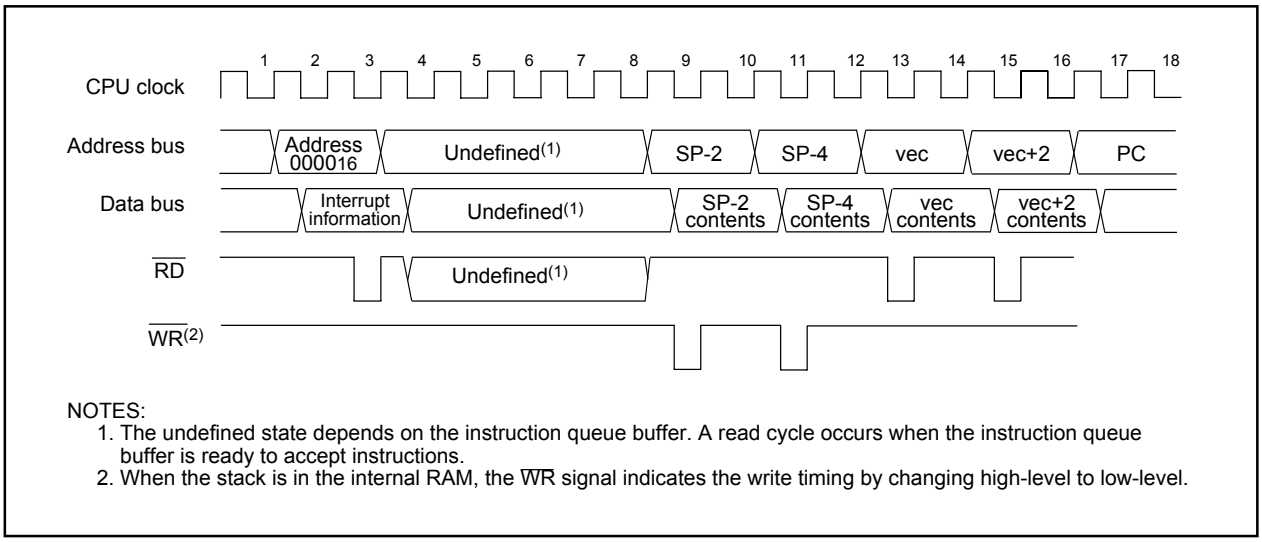
(5) The PC is saved to the stack.

(6) The interrupt priority level of the accepted interrupt is set in the IPL.

(7) The start address of the relevant interrupt routine set in the interrupt vector is stored in the PC.

After the interrupt sequence is completed, the processor resumes executing instructions from the start address of the interrupt routine.

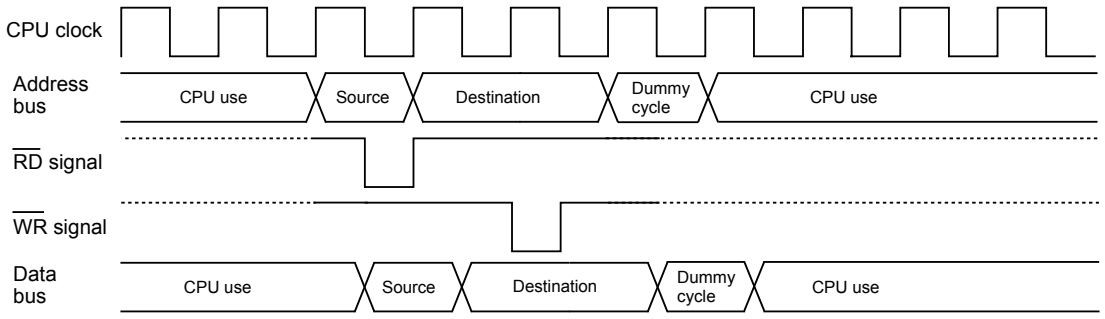
NOTE:
1. This register cannot be used by user.



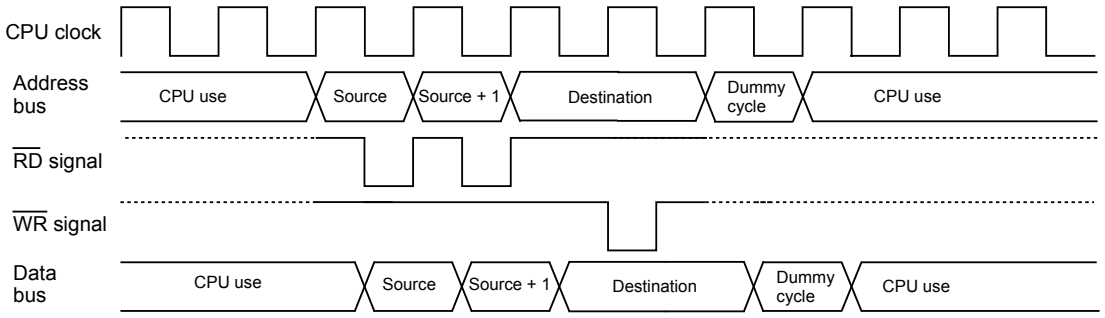
NOTES:
1. The undefined state depends on the instruction queue buffer. A read cycle occurs when the instruction queue buffer is ready to accept instructions.
2. When the stack is in the internal RAM, the $\overline{WR}$ signal indicates the write timing by changing high-level to low-level.

Figure 9.5 Time Required for Executing Interrupt Sequence

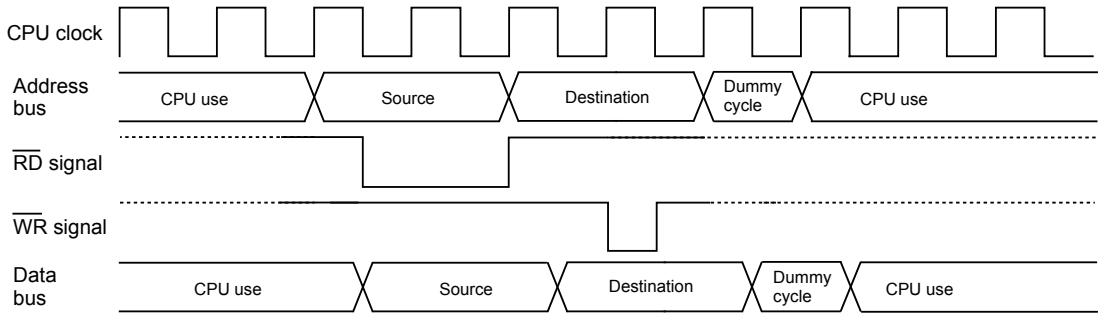
(1) When the transfer unit is 8 or 16 bits and the source of transfer is an even address



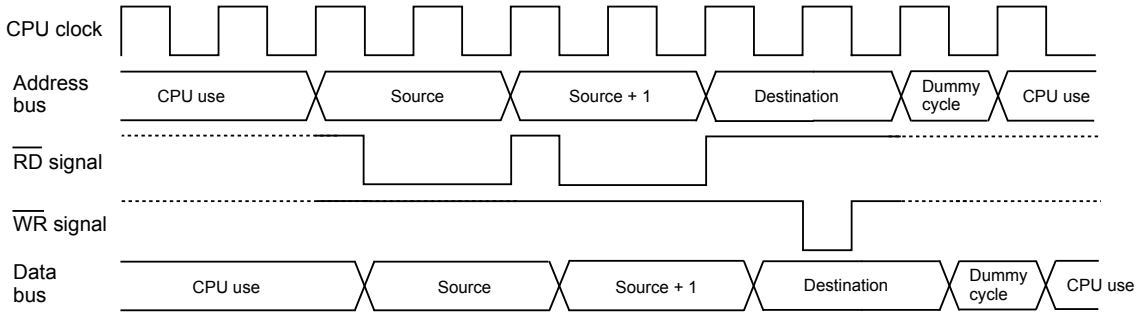
(2) When the transfer unit is 16 bits and the source address of transfer is an odd address.



(3) When the source read cycle under condition (1) has one wait state inserted



(4) When the source read cycle under condition (2) has one wait state inserted



NOTE:
1. The same timing changes occur with the respective conditions at the destination as at the source.

Figure 11.5 Transfer Cycles for Source Read

11.3 DMA Enable

When a data transfer starts after setting the DMAE bit in the DMiCON register (i = 0, 1) to 1 (enabled), the DMAC operates as follows:

(a) Reload the forward address pointer with the SARi register value when the DSD bit in DMiCON register is 1 (forward) or the DARi register value when the DAD bit in the DMiCON register is 1 (forward).

(b) Reload the DMAi transfer counter with the DMAi transfer counter reload register value.

If the DMAE bit is set to 1 again while it remains set, the DMAC performs the above operation. However, if a DMA request may occur simultaneously when the DMAE bit is being written, follow the steps below.

(1) Write 1 to bits DMAE and DMAS in DMiCON register simultaneously.

(2) Make sure that the DMAi is in an initial state as described above (a) and (b) by program.

If the DMAi is not in an initial state, the above steps should be repeated.

11.4 DMA Request

The DMAC can generate a DMA request as triggered by the cause of request that is selected with the DMS bit and bits DSEL3 to DSEL0 in the DMiSL register (i = 0, 1) on either channel. **Table 11.4** shows the timing at which the DMAS bit changes state.

Whenever a DMA request is generated, the DMAS bit is set to 1 (DMA requested) regardless of whether or not the DMAE bit is set. If the DMAE bit was set to 1 (enabled) when this occurred, the DMAS bit is set to 0 (DMA not requested) immediately before a data transfer starts. This bit cannot be set to 1 by program (it can only be set to 0).

The DMAS bit may be set to 1 when the DMS or the DSEL3 to DSEL0 bits change state. Therefore, always be sure to set the DMAS bit to 0 after changing the DMS or the DSEL3 to DSEL0 bits.

Because if the DMAE bit is set to 1, a data transfer starts immediately after a DMA request is generated, the DMAS bit in almost all cases is 0 when read by program. Read the DMAE bit to determine whether the DMAC is enabled.

Table 11.4 Timing at Which the DMAS Bit Changes State

DMA Factor	DMAS Bit in the DMiCON Register	
	Timing at which the bit is set to 1	Timing at which the bit is set to 0
Software trigger	When the DSR bit in the DMiSL register is set to 1	• Immediately before a data transfer starts • When set by writing 0 by program
Peripheral function	When the interrupt control register for the peripheral function that is selected by bits DSEL3 to DSEL0 and the DMS bit in the DMiSL register has its IR bit set to 1	

Three-phase PWM Control Register 1 (1)

b7	b6	b5	b4	b3	b2	b1	b0
0							

Symbol	Address	After Reset
INVC1	034916	0016

Bit Symbol	Bit Name	Function	RW
INV10	Timer A1, A2, A4 start trigger signal select bit	0: Timer B2 underflow 1: Timer B2 underflow and write to the TB2 register (2)	RW
INV11	Timer A1-1, A2-1, A4-1 control bit (3)	0: Three-phase mode 0 (4) 1: Three-phase mode 1	RW
INV12	Dead time timer count source select bit	0: f1 or f2 1: f1 divided by 2 or f2 divided by 2	RW
INV13	Carrier wave detect flag (5)	0: Timer Reload control signal is set to 0 1: Timer Reload control signal is set to 1	RO
INV14	Output polarity control bit	0 : Output waveform “L” active 1 : Output waveform “H” active	RW
INV15	Dead time invalid bit	0: Dead time timer enabled 1: Dead time timer disabled	RW
INV16	Dead time timer trigger select bit	0: Falling edge of timer A4, A1 or A2 one-shot pulse 1: Rising edge of three-phase output shift register (U, V or W phase) output (6)	RW
$\overline{\text{(b7)}}$	Reserved bit	Set to 0	RW

NOTES:

1. Write to this register after setting the PRC1 bit in the PRCR register to 1 (write enable). Note also that this register can only be rewritten when timers A1, A2, A4 and B2 are idle.
2. A start trigger is generated by writing to the TB2 register only while timer B2 stops.
3. The effects of the INV11 bit are described in the table below.

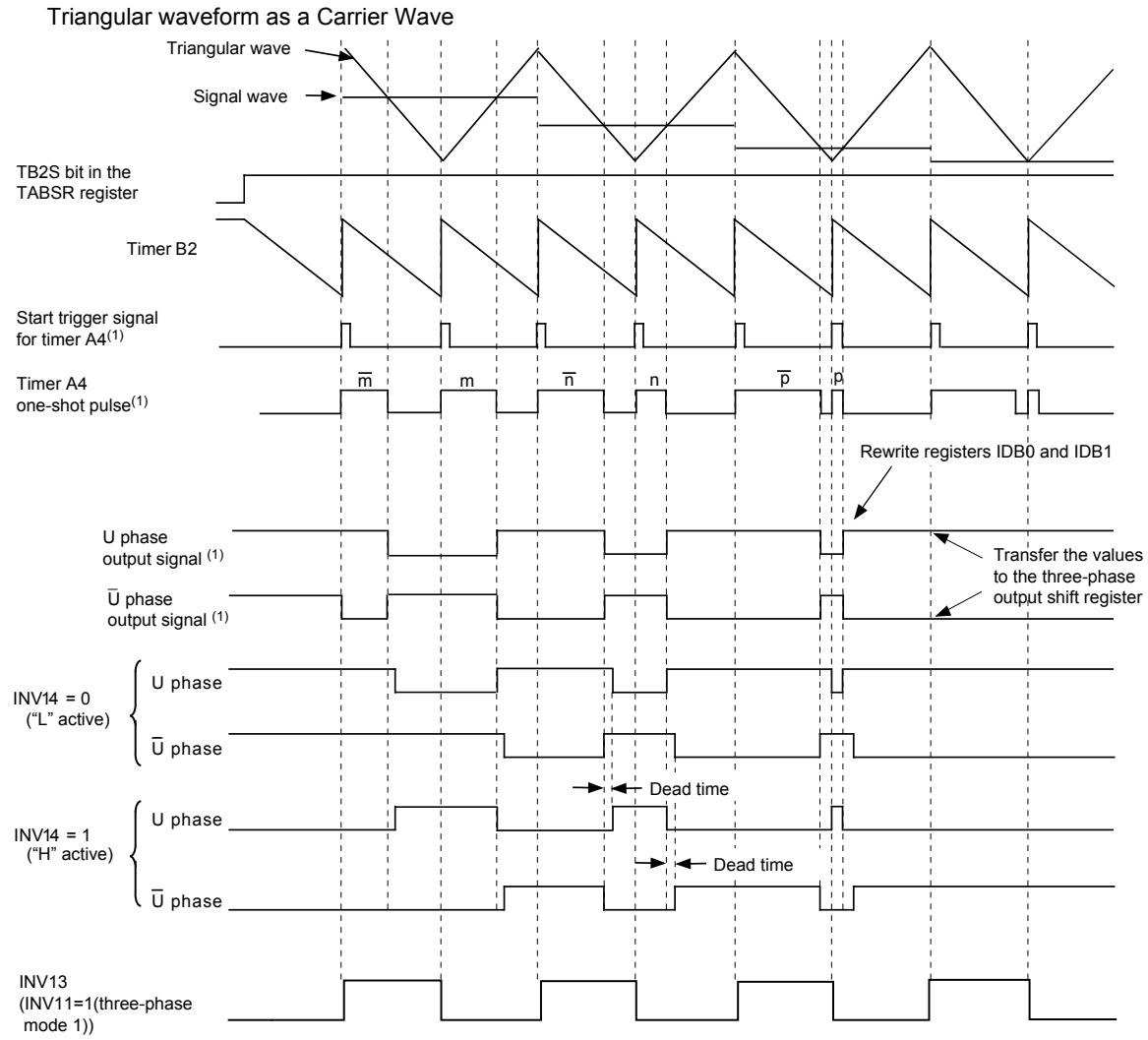
Item	INV11=0	INV11=1
Mode	Three-phase mode 0	Three-phase mode 1
TA11, TA21, TA41 registers	Not Used	Used
INV00 bit, INV01 bit	Has no effect. ICTB2 counted every time timer B2 underflows regardless of whether bits INV00 and INV01 are set	Effect
INV13 bit	Has no effect	Effective when INV11 bit is 1 and INV06 bit is 0

4. If the INV06 bit is 1 (sawtooth wave modulation mode), set this bit to 0 (three-phase mode 0). Also, if the INV11 bit is 0, set the PWCON bit to 0 (timer B2 reloaded by a timer B2 underflow).
5. The INV13 bit is effective only when the INV06 bit is set to 0 (triangular wave modulation mode) and the INV11 bit is set to 1 (three-phase mode 1).
6. If all of the following conditions hold true, set the INV16 bit to 1 (dead time timer triggered by the rising edge of three-phase output shift register output)
   - The INV15 bit is 0 (dead time timer enabled)
   - When the INV03 bit is set to 1 (three-phase motor control timer output enabled), the Dij bit and DiBj bit (i:U, V, or W, j: 0 to 1) have always different values (the positive-phase and negative-phase always output different levels during the period other than dead time).

Conversely, if either one of the above conditions holds false, set the INV16 bit to 0 (dead time timer triggered by the falling edge of one-shot pulse).

Figure 12.27 INVC1 Register

The three-phase motor control timer function is enabled by setting the INV02 bit in the INVC0 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 12.33** shows the example of triangular modulation waveform, and **Figure 12.34** shows the example of sawtooth modulation waveform.



NOTE:
1. Internal signals. See **Figure 12.25**.

The above applies under the following conditions:
INVC0 = 00XX11XX2 (X varies depending on each system) and INVC1 = 010XXXX02.
Examples of PWM output change are:

(1)When INV11 = 1 (three-phase mode 1)
- INV01 = 0 and ICTB2 = 216 (the timer B2 interrupt is generated every two times the timer B2 underflows), or INV01 = 1, INV00 = 1, and ICTB2=116 (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.

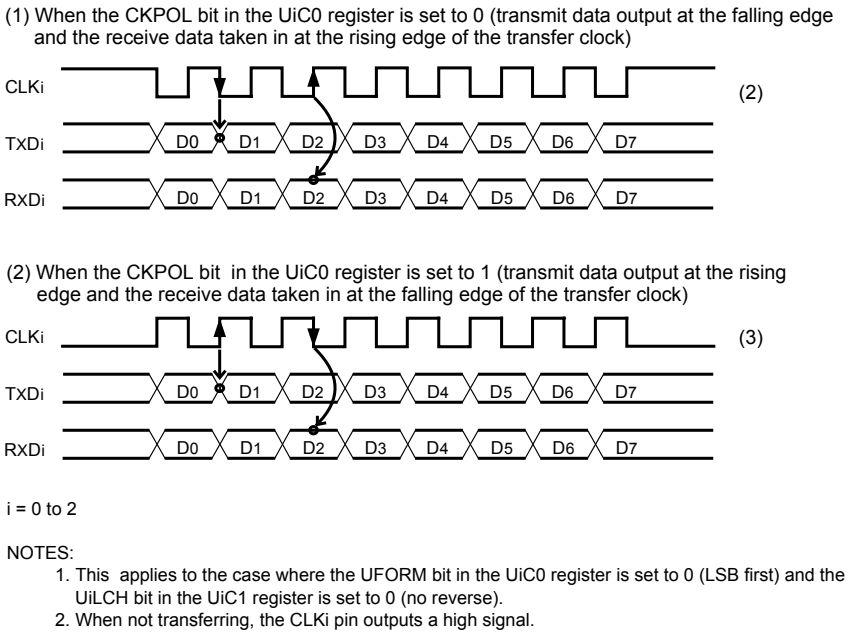
(2)When INV11 = 0 (three-phase mode 0)
- INV01 = 0, ICTB2 = 116 (the timer B2 interrupt is generated whenever timer B2 underflows)
- Default value of the timer: TA4 = $\overline{m}$. The TA4 register is changed whenever the timer B2 interrupt is generated.
  First time: TA4 = m. Second tim:, TA4 = $\overline{n}$.
  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 12.33 Triangular Wave Modulation Operation

14.1.1.2 CLK Polarity Select Function

Use the CKPOL bit in the UiC0 register (i=0 to 2) to select the transfer clock polarity. **Figure 14.11** shows the polarity of the transfer clock.



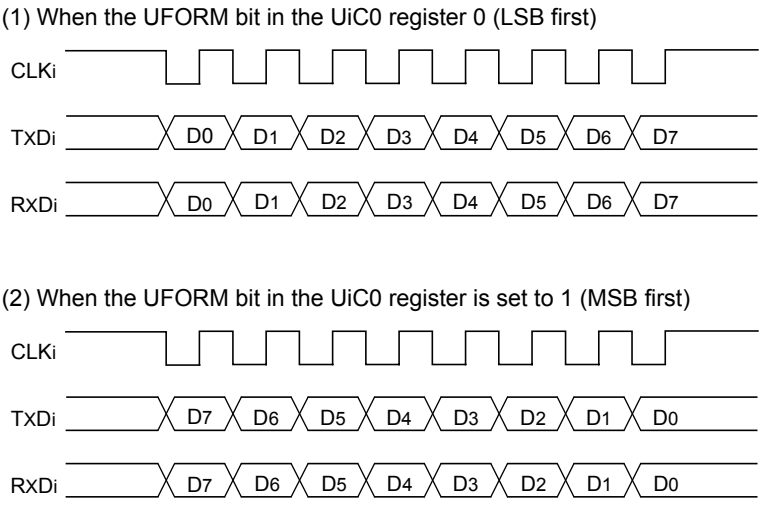
i = 0 to 2

NOTES:
1. This applies to the case where the UFORM bit in the UiC0 register is set to 0 (LSB first) and the UiLCH bit in the UiC1 register is set to 0 (no reverse).
2. When not transferring, the CLKi pin outputs a high signal.

Figure 14.11 Polarity of transfer clock

14.1.1.3 LSB First/MSB First Select Function

Use the UFORM bit in the UiC0 register (i=0 to 2) to select the transfer format. **Figure 14.12** shows the transfer format.



i = 0 to 2

NOTE:
1. This applies to the case where the CKPOL bit in the UiC0 register is set to 0 (transmit data output at the falling edge and the receive data taken in at the rising edge of the transfer clock) and the UiLCH bit in the UiC1 register 0 (no reverse).

Figure 14.12 Transfer format

14.1.2 Clock Asynchronous Serial I/O (UART) Mode

The UART mode allows transmitting and receiving data after setting the desired bit rate and transfer data format. **Table 14.5** lists the specifications of the UART mode.

Table 14.5 UART Mode Specifications

Item	Specification
Transfer data format	• Character bit (transfer data): Selectable from 7, 8 or 9 bits • Start bit: 1 bit • Parity bit: Selectable from odd, even, or none • Stop bit: Selectable from 1 or 2 bits
Transfer clock	• The CKDIR bit in the UiMR(i=0 to 2) register is set to 0 (internal clock) : fj/ (16(n+1)) fj = f_{1SIO}, f_{2SIO}, f_{8SIO}, f_{32SIO}. n: Setting value of UiBRG register 00_{16} to FF_{16} • CKDIR bit is set to 1 (external clock) : f_{EXT}/16(n+1) f_{EXT}: Input from CLKi pin. n :Setting value of UiBRG register 00_{16} to FF_{16}
Transmission, reception control	• Selectable from $\overline{CTS}$ function, $\overline{RTS}$ function or $\overline{CTS}/\overline{RTS}$ function disable
Transmission start condition	• Before transmission can start, the following requirements must be met – The TE bit in the UiC1 register is set to 1 (transmission enabled) – The TI bit in the UiC1 register is set to 0 (data present in UiTB register) – If $\overline{CTS}$ function is selected, input on the $\overline{CTS}$i pin is set to “L”
Reception start condition	• Before reception can start, the following requirements must be met" – The RE bit in the UiC1 register is set to 1 (reception enabled) – Start bit detection
Interrupt request generation timing	• For transmission, one of the following conditions can be selected – The UiIRS bit (2) is set to 0 (transmit buffer empty): when transferring data from the UiTB register to the UARTi transmit register (at start of transmission) – The UiIRS bit is set to1 (transfer completed): when the serial I/O finished sending data from the UARTi transmit register • For reception When transferring data from the UARTi receive register to the UiRB register (at completion of reception)
Error detection	• Overrun error (1) This error occurs if the serial I/O started receiving the next data before reading the UiRB register and received the bit one before the last stop bit in the the next data • Framing error This error occurs when the number of stop bits set is not detected • Parity error This error occurs when if parity is enabled, the number of 1 in parity and character bits does not match the number of 1 set • Error sum flag This flag is set to 1 when any of the overrun, framing, and parity errors is encountered
Select function	• LSB first, MSB first selection Whether to start sending/receiving data beginning with bit 0 or beginning with bit 7 can be selected • Serial data logic switch (UART2) This function reverses the logic of the transmit/receive data. The start and stop bits are not reversed. • TxD, RxD I/O polarity switch (UART2) This function reverses the polarities of hte TxD pin output and RxD pin input. The logic levels of all I/O data is reversed. • Separate $\overline{CTS}/\overline{RTS}$ pins (UART0) $\overline{CTS_0}$ and $\overline{RTS_0}$ are input/output from separate pins • UART1 pin remapping selection The UART1 pin can be selected from the $P6_7$ to $P6_4$ or $P7_3$ to $P7_0$

NOTES:

1. If an overrun error occurs, bits 8 to 0 in the UiRB register are undefined. The IR bit in the SiRIC register remains unchange.
2. Bits U0IRS and U1IRS respectively are the UCON register bits 0 and 1; the U2IRS bit is the U2C1 register bit 4.

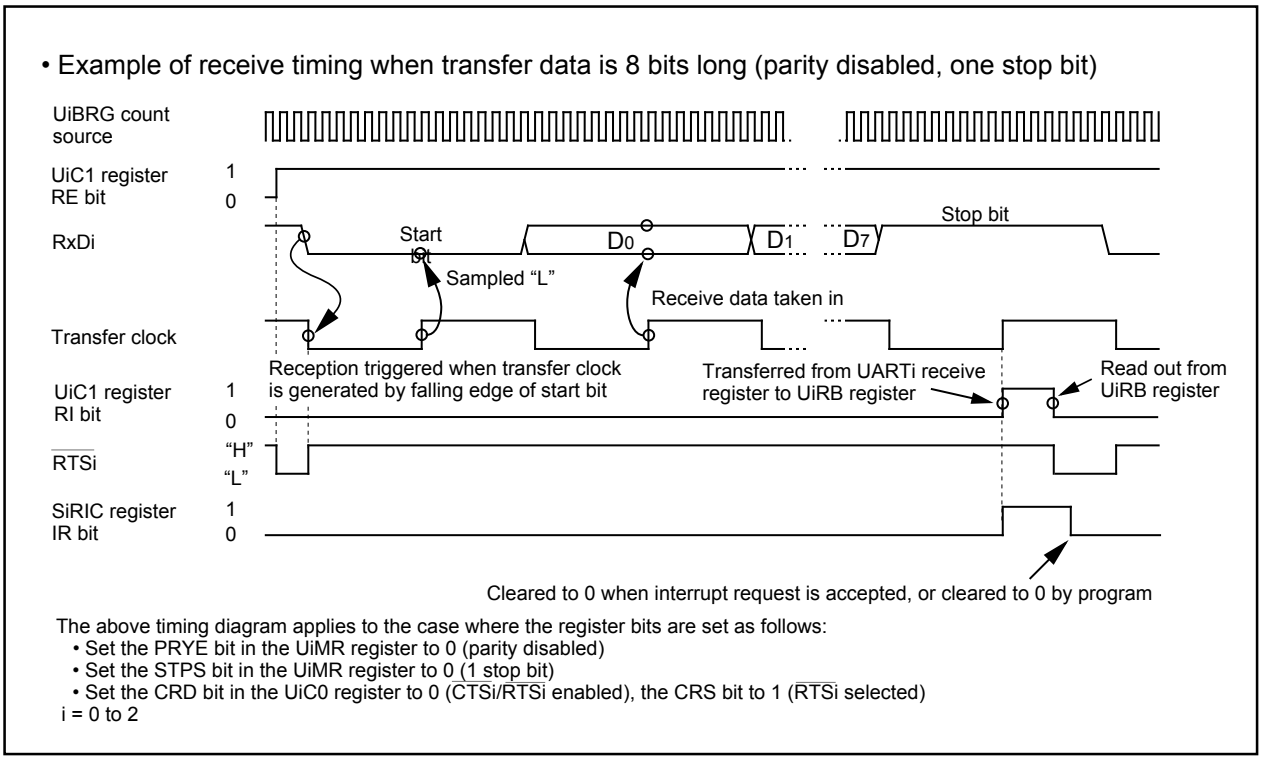


Figure 14.17 Receive Operation

14.1.2.1 Bit Rates

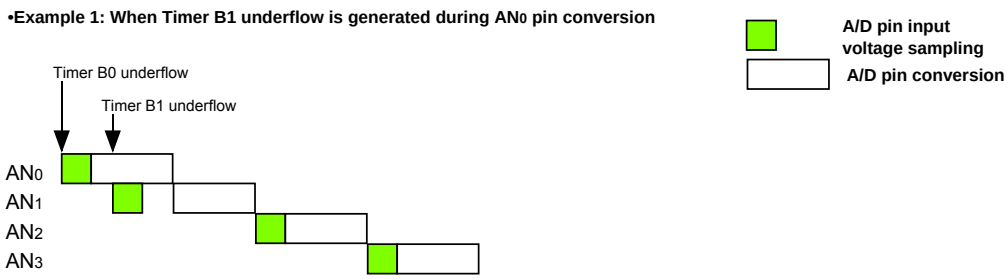
In UART mode, the frequency set by the UiBRG register (i=0 to 2) divided by 16 become the bit rates. **Table 14.9** lists example of bit rate and settings.

Table 14.9 Example of Bit Rates and Settings

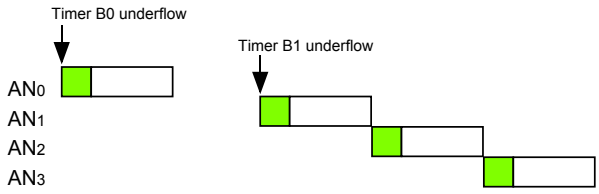
Bit Rate (bps)	Count Source of BRG	Peripheral Function Clock : 16MHz		Peripheral Function Clock : 20MHz	
		Set Value of BRG : n	Actual Time (bps)	Set Value of BRG : n	Actual Time (bps)
1200	f8	103(67h)	1202	129(81h)	1202
2400	f8	51(33h)	2404	64(40h)	2404
4800	f8	25(19h)	4808	32(20h)	4735
9600	f1	103(67h)	9615	129(81h)	9615
14400	f1	68(44h)	14493	86(56h)	14368
19200	f1	51(33h)	19231	64(40h)	19231
28800	f1	34(22h)	28571	42(2Ah)	29070
31250	f1	31(1Fh)	31250	39(27h)	31250
38400	f1	25(19h)	38462	32(20h)	37879
51200	f1	19(13h)	50000	24(18h)	50000

•Example when selecting AN_0 to AN_3 to A/D sweep pins (SCAN1 to SCAN0 = 01_2)

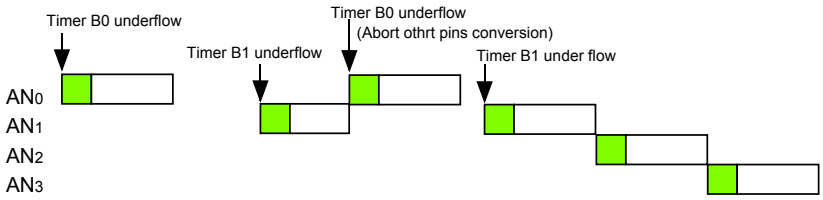
•Example 1: When Timer B1 underflow is generated during AN_0 pin conversion



•Example 2: When Timer B1 underflow is generated after AN_0 pin conversion



•Example 3: When Timer B0 underflow is generated during A/D conversion of any pins except AN_0 pin



•Example 4: When Timer B0 underflow is generated again before Timer B1 underflow is generated after Timer B0 underflow generation

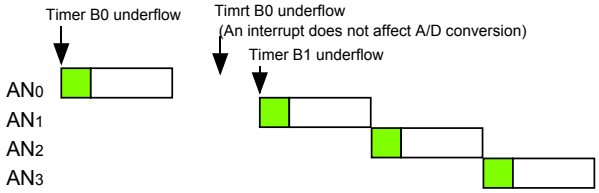


Figure 15.19 Operation Example in Delayed Trigger Mode 0

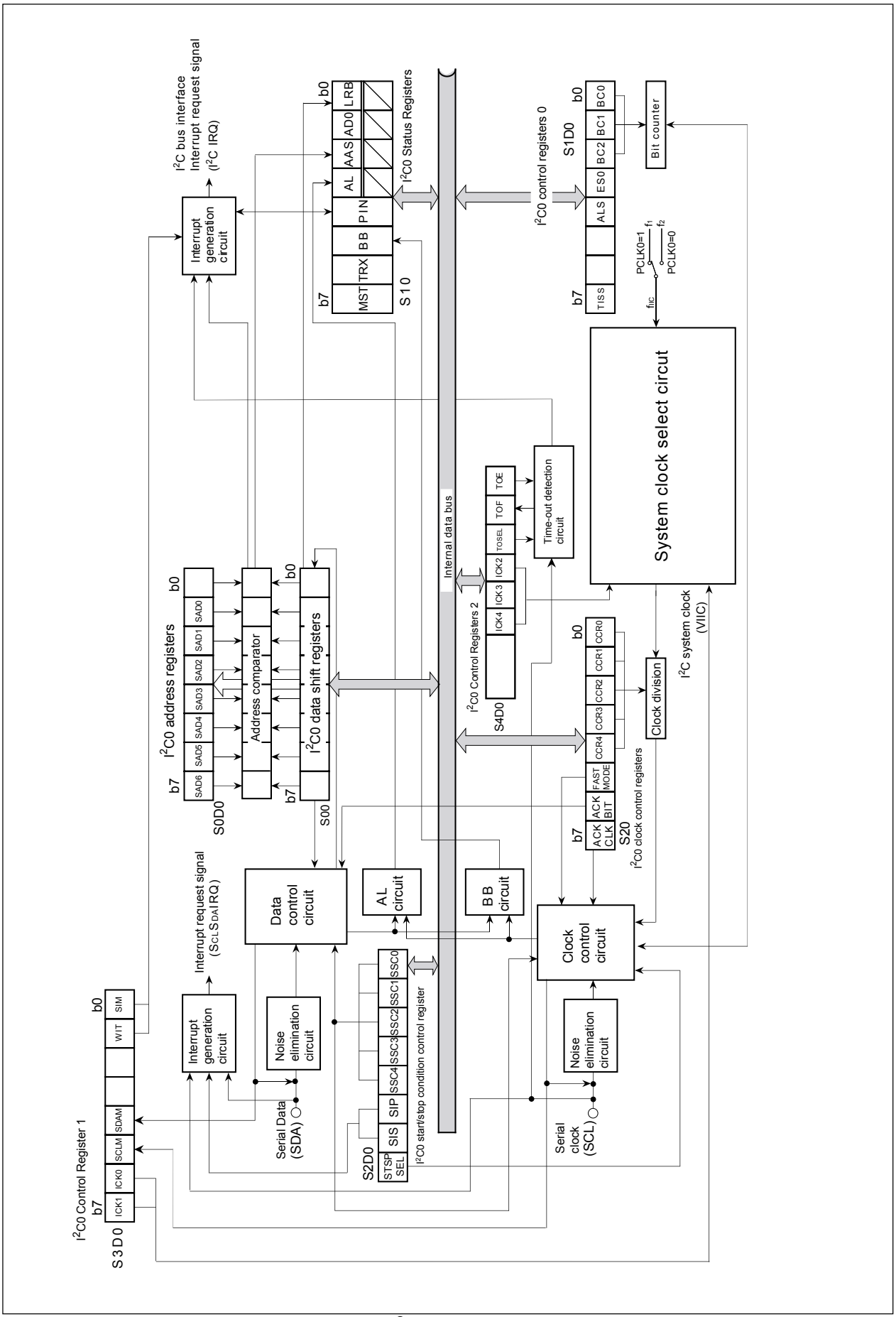


Figure 16.1 Block diagram of multi-master I2C bus interface

16.4.5 Bit 7: I2C bus Interface Pin Input Level Select Bit (TISS)

The TISS bit selects the input level of the SCL and SDA pins for the multi-master I2C bus interface. When the TISS bit is set to 1, the P20 and P21 become the SMBus input level.

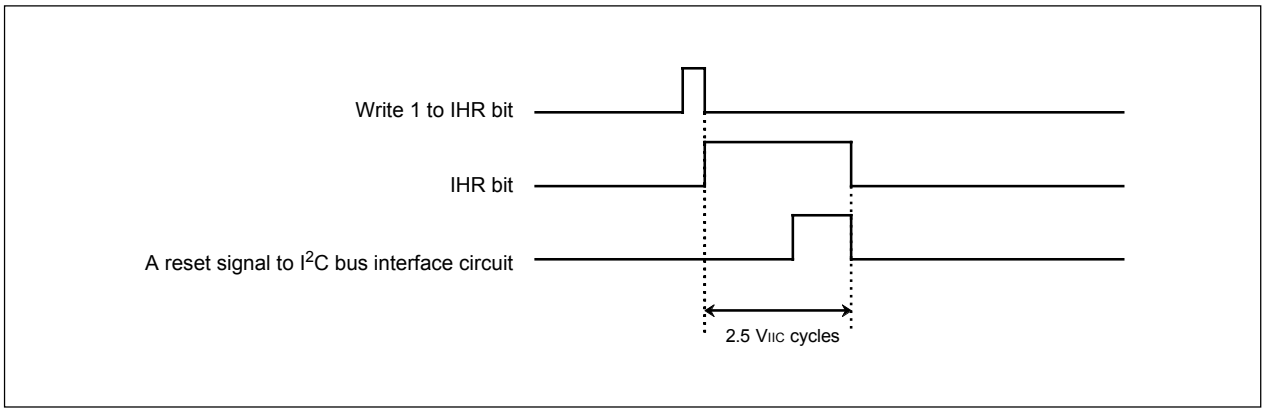


Figure 16.10 The timing of reset to the I2C bus interface circuit

17.3 Configuration of the CAN Module System Clock

The M16C/29 Group has a CAN module system clock select circuit.

Configuration of the CAN module system clock can be done through manipulating the CCLKR register and the BRP bit in the C0CONR register.

For the CCLKR register, refer to **7. Clock Generation Circuit**.

Figure 17.19 shows a block diagram of the clock generation circuit of the CAN module system.

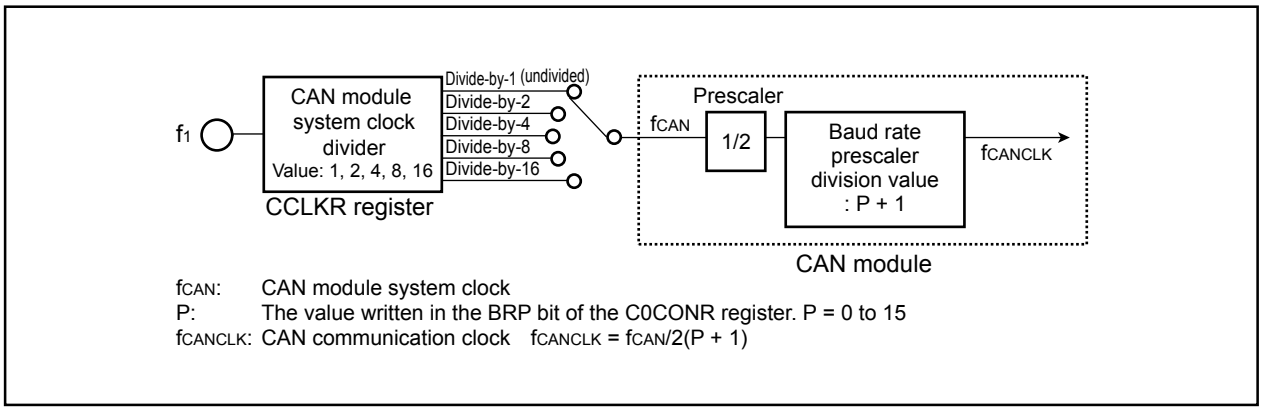


Figure 17.19 Block Diagram of CAN Module System Clock Generation Circuit

17.3.1 Bit Timing Configuration

The bit time consists of the following four segments:

- Synchronization segment (SS)
  This serves for monitoring a falling edge for synchronization.
- Propagation time segment (PTS)
  This segment absorbs physical delay on the CAN network which amounts to double the total sum of delay on the CAN bus, the input comparator delay, and the output driver delay.
- Phase buffer segment 1 (PBS1)
  This serves for compensating the phase error. When the falling edge of the bit falls later than expected, the segment can become longer by the maximum of the value defined in SJW.
- Phase buffer segment 2 (PBS2)
  This segment has the same function as the phase buffer segment 1. When the falling edge of the bit falls earlier than expected, the segment can become shorter by the maximum of the value defined in SJW.

Figure 17.20 shows the bit timing.

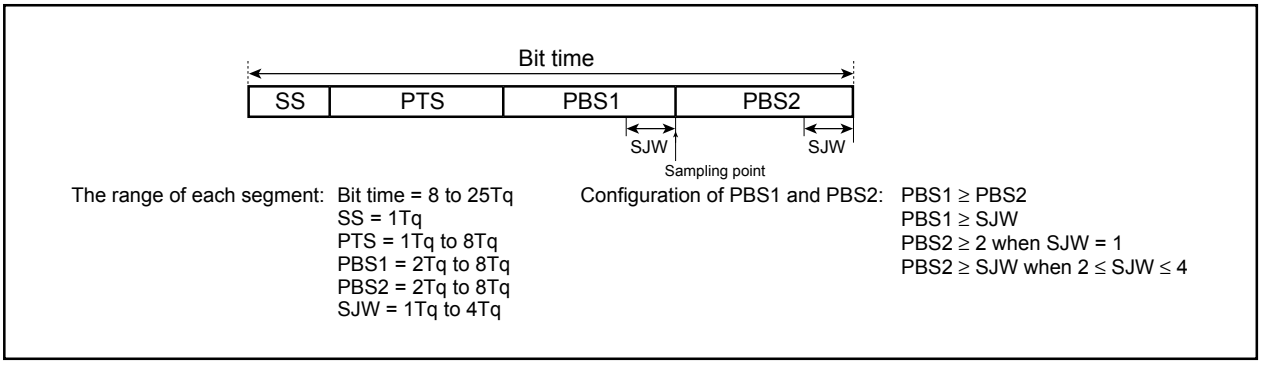


Figure 17.20 Bit Timing

17.10.2 Transmission

Figure 17.26 shows the timing of the transmit sequence.

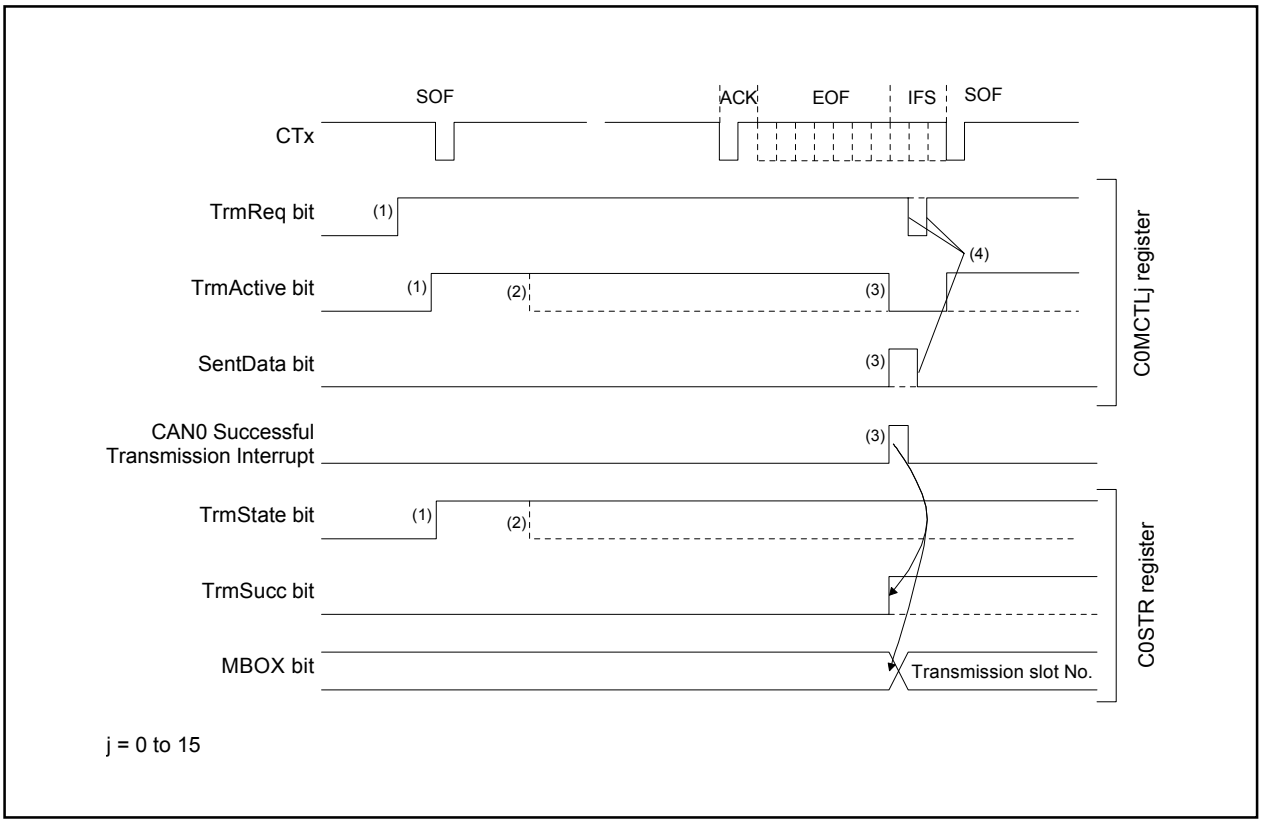


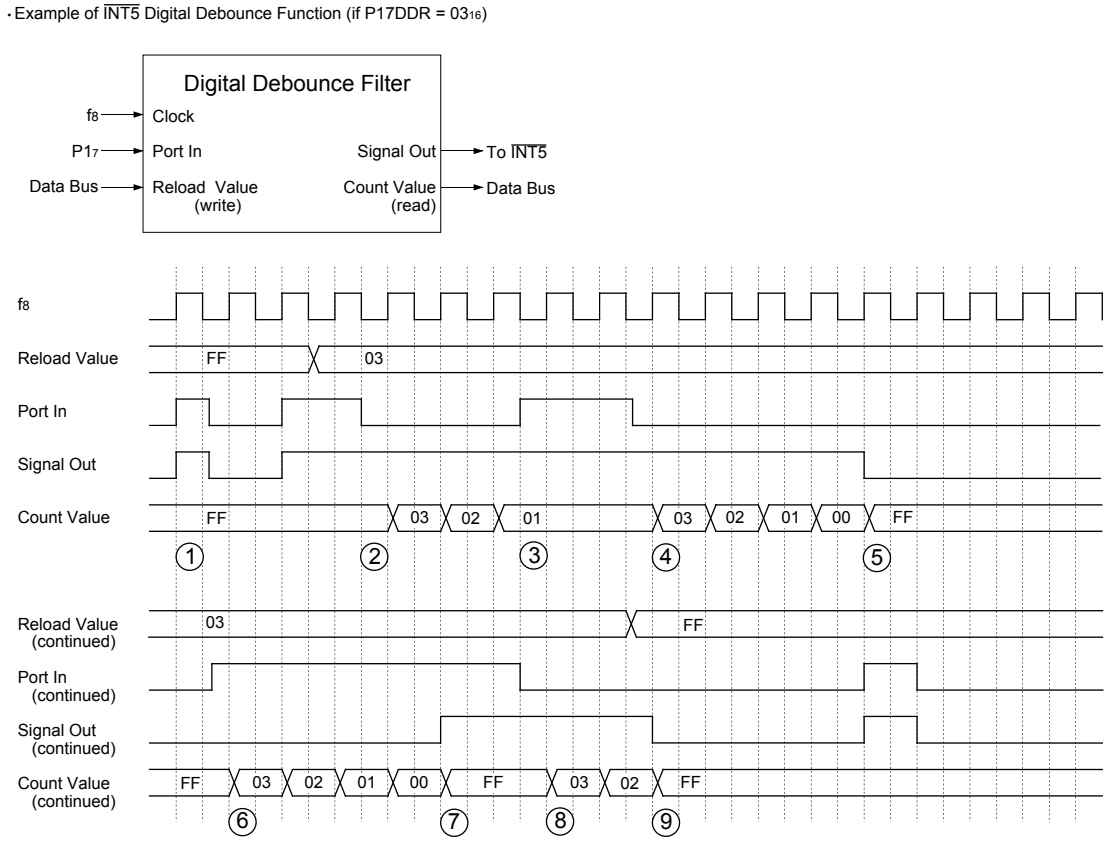
Figure 17.26 Timing of Transmit Sequence

(1) If the TrmReq bit in the C0MCTLj register (j = 0 to 15) is set to 1 (Transmission slot) in the bus idle state, the TrmActive bit in the C0MCTLj register and the TrmState bit in the C0STR register are set to 1 (Transmitting/Transmitter), and CAN module starts the transmission.

(2) If the arbitration is lost after the CAN module starts the transmission, the TrmActive and TrmState bits are set to 0.

(3) If the transmission has been successful without lost in arbitration, the SentData bit in the C0MCTLj register is set to 1 (Transmission is successfully completed) and TrmActive bit in the C0MCTLj register is set to 0 (Waiting for bus idle or completion of arbitration). And when the interrupt enable bits in the C0ICR register = 1 (Interrupt enabled), CAN0 successful transmission interrupt request is generated and the MBOX (the slot number which transmitted the message) and TrmSucc bit in the C0STR register are changed.

(4) When starting the next transmission, set bits SentData and TrmReq to 0. And set the TrmReq bit to 1 after checking that bits SentData and TrmReq are set to 0.



1. (Condition after reset). P17DDR=FF16. Pin input signal will be output directly.
2. Set the P17DDR register to 0316. The P17DDR register starts decrement along the f8 as a counter source, if the pin input level (e.g.,"L") and the signal output level (e.g.,"H") are not matched.
3. The P17DDR register will stops counting when the pin input level and the signal output level are matched (e.g., both levels are "H") while counting.
4. If the pin input level (e.g.,"L") and the signal output level (e.g.,"H") are not matched the P17DDR register will start decrement again after the setting value is reloaded.
5. When the P17DDR register is underflow, it stops counting and the signal output will output the same as pin input level (e.g."L").
6. If the pin input level (e.g.,"H") and the signal output level (e.g., "L") are not matched again, the P17DDR register will start decrement again after the setting value is reloaded.
7. When the P17DDR register is underflow, it stops counting and the signal output will output the same as pin input level (e.g."H").
8. If the pin input level (e.g.,"H") and the signal output level (e.g., "L") are not matched again, the P17DDR register will start decrement again after the setting value is reloaded.
9. Set the P17DDR register to FF16. The P17DDR register starts counting after the setting value is reloaded. Pin input signal will be output directly.

Figure 19.12 Functioning of Digital Debounce Filter

20.10 Parallel I/O Mode

In parallel input/output mode, the user ROM can be rewritten by a parallel programmer supporting the M16C/29 group. Contact your parallel programmer manufacturer for more information on the parallel programmer. Refer to the user's manual included with your parallel programmer for instructions.

20.10.1 ROM Code Protect Function

The ROM code protect function prevents the flash memory from being read or rewritten. (Refer to **20.3 Functions To Prevent Flash Memory from Rewriting**).

V_{CC} = 3V

Table 21.25 Electrical Characteristics (2) (Note 1)

Symbol	Parameter	Measurement Condition			Standard			Unit
					Min.	Typ.	Max.	
I_{CC}	Power Supply Current (V_{CC} = 2.7 to 3.6V)	Output pins are left open and other pins are connected to V_{SS}	Mask ROM	f(BCLK) = 10 MHz, main clock, no division		8	13	mA
				On-chip oscillation, $f_{2(ROC)}$ selected, f(BCLK) = 1 MHz		1		mA
			Flash memory	f(BCLK) = 10 MHz, main clock, no division		8	13	mA
			Flash memory program	f(BCLK) = 10 MHz, Vcc = 3.0 V		11		mA
			Flash memory erase	f(BCLK) = 10 MHz, Vcc= 3.0 V		11		mA
			Mask ROM	f(BCLK) = 32 kHz, In low-power consumption mode, Program running on ROM(3)		20		µA
				On-chip oscillation, $f_{2(ROC)}$ selected, f(BCLK) = 1 MHz, In wait mode		25		µA
			Flash memory	f(BCLK) = 32 kHz, In low-power consumption mode, Program running on RAM(3)		20		µA
				f(BCLK) = 32 kHz, In low-power consumption mode, Program running on flash memory(3)		450		µA
				On-chip oscillation, $f_{2(ROC)}$ selected, f(BCLK) = 1 MHz, In wait mode(4)		45		µA
			Mask ROM, Flash memory	f(BCLK) = 32 kHz, In wait mode(2), Oscillation capacity high		6.6		µA
				f(BCLK) = 32 kHz, In wait mode(2), Oscillation capacity low		2.2		µA
				While clock stops, Topr = 25° C		0.7	3	µA
Idet4	Low voltage detection dissipation current(4)					0.6	4	µA
Idet3	Reset level detection dissipation current(4)					1.0	5	µA

NOTES:

1. Referenced to V_{CC} = 2.7 to 3.6 V, V_{SS} = 0 V at Topr = -20 to 85 ° C / -40 to 85 ° C, f(BCLK) = 10 MHz unless otherwise specified.
2. With one timer operates, using f_{C32}.
3. This indicates the memory in which the program to be executed exists.
4. Idet is dissipation current when the following bit is set to 1 (detection circuit enabled).
   Idet4: the VC27 bit of the VCR2 register
   Idet3: the VC26 bit in the VCR2 register

22.8.2 UART Mode

22.8.2.1 Special Mode 1 (I^2C bus Mode)

When generating start, stop and restart conditions, set the STSPSEL bit in the U2SMR4 register to 0 and wait for more than half cycle of the transfer clock before setting each condition generate bit (STAREQ, RSTAREQ and STPREQ) from 0 to 1.

22.8.2.2 Special Mode 2

If a low-level signal is applied to the $\overline{SD}$ pin when the IVPCR1 bit in the TB2SC register is set to 1 (three-phase output forcible cutoff by input on $\overline{SD}$ pin enabled), the RTS_2 and CLK_2 pins go to a high-impedance state.

22.8.2.3 Special Mode 4 (SIM Mode)

A transmit interrupt request is generated by setting the U2C1 register U2IRS bit to 1 (transmission complete) and U2ERE bit to 1 (error signal output) after reset. Therefore, when using SIM mode, be sure to clear the IR bit to 0 (no interrupt request) after setting these bits.

22.8.3 SI/O3, SI/O4

The SOUTi default value which is set to the SOUTi pin by the SMi7 bit approximately 10ns may be output when changing the SMi3 bit from 0 (I/O port) to 1 (SOUTi output and CLKfunction) while the SMi2 bit in the SiC (i=3 and 4) to 0 (SOUTi output) and the SMi6 bit is set to 1 (internal clock). And then the SOUTi pin is held high-impedance.

If the level which is output from the SOUTi pin is a problem when changing the SMi3 bit from 0 to 1, set the default value of the SOUTi pin by the SMi7 bit.

Rev.	Date	Description	
		Page	Summary
		360	Table 21.1 is partly revised.
		368	Section “21.4.2 EW1 Mode” is partly revised.
0.80	Sep/03/Y04	2,3	Table 1.2.1 and Table 1.2.2 are partly revised.
		6,7	Table 1.4.1 to Table 1.4.3 are partly revised.
		7	Figure 1.4.1 is partly revised.
		8,9	Figure 1.5.1 and Figure 1.5.2 are partly revised.
		21	Figure 4.7 is partly revised.
		24	Figure 4.10 is partly revised.
		26	Section “5.1.2 Hardware Reset 2” is partly revised.
		29 to 34	Section “5.5 Voltage Detection Circuit” is revised.
		80	Section “10.2 Cold start / Warm start” is added.
		322	Table 20.2 is partly revised.
		323	Table 20.3 is partly revised.
		325	Table 20.6 and Table 20.7 are partly revised.
		327	Table 20.9 is partly revised.
		331	Title of Table 20.23 is partly revised.
		335	Table 20.25 is partly revised.
		339	Title of Table 20.39 is partly revised.
		343	Table 20.41 is partly revised.
		344	Table 20.42 is partly revised.
		346	“Low Voltage Detection Circuit Electrical Characteristics” is deleted.
			Talbe 20.45 is partly revised.
		348	Table 20.47 is partly revised.
		352	Title of Table 20.61 is partly revised.
		356	Talbe 20.63 is partly revised.
		360	Title of Table 20.77 is partly revised.
		398	64P6Q-A package is revised.
1.00	Nov/01/Y04	All pages	Words standardized (on-chip oscillator, A/D)
		2, 3	Table1.2.1 and Table 1.2.2 are partly revised.
		8, 9	Table 1.4.4 to 1.4.6 and figure1.4.2 to 1.4.6 are added.
		28	“5.1.2 Hardware Reset 2” is partly revised.
		29	“5.4 Oscillation Stop Detection Reset” is partly revised.
		38	Table 7.1 is partly revised.
		41	Note 6 in Figure 7.3 is partly revised. b7 to b4 bit in Figure 7.4 is revised.
		42	Figure 7.5 is partly revised.
		43	“PCLKR register” in Figure 7.6 is partly revised.
		50	“7.6.1 Normal Operation Mode” is partly revised.
		51	Note 1 in Table 7.6.1.1 is partly revised.
		57	“7.8 Oscillation Stop and Re-oscillation Detect Function” is partly revised.