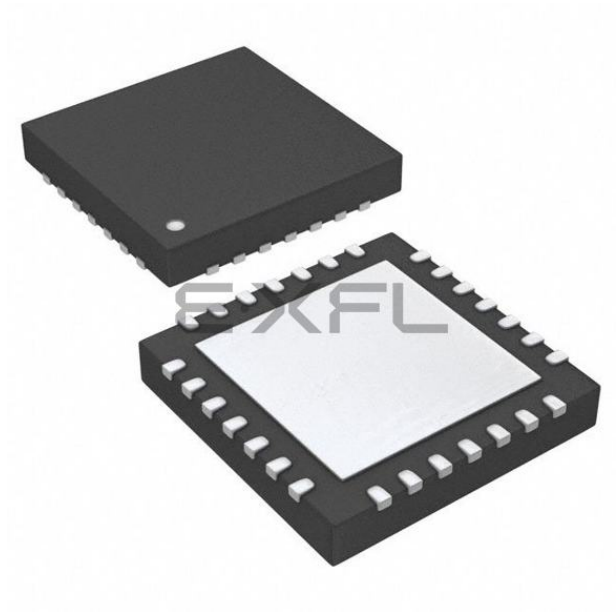


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	16-Bit
Speed	32MHz
Connectivity	I²C, PMP, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	21
Program Memory Size	32KB (11K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 10x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24fj32ga002-i-ml

TABLE 4-22: CLOCK CONTROL REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
RCON	0740	TRAPR	IOPUWR	—	—	—	—	CM	PMSLP	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	(Note 1)
OSCCON	0742	—	COSC2	COSC1	COSC0	—	NOSC2	NOSC1	NOSC0	CLKLOCK	IOLOCK	LOCK	—	CF	—	SOSCEN	OSWEN	(Note 2)
CLKDIV	0744	ROI	DOZE2	DOZE1	DOZE0	DOZEN	RCDIV2	RCDIV1	RCDIV0	—	—	—	—	—	—	—	—	3140
OSCTUN	0748	—	—	—	—	—	—	—	—	—	—	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.
Note 1: RCON register Reset values are dependent on the type of Reset.
2: OSCCON register Reset values are dependent on configuration fuses and by the type of Reset.

TABLE 4-23: NVM REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
NVMCON	0760	WR	WREN	WRERR	—	—	—	—	—	—	ERASE	—	—	NVMOP3	NVMOP2	NVMOP1	NVMOP0	0000(1)
NVMKEY	0766	—	—	—	—	—	—	—	—	NVMKEY<7:0>								0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.
Note 1: Reset value shown is for a POR only. The value on other Reset states is dependent on the state of the memory write or erase operations at the time of Reset.

TABLE 4-24: PMD REGISTER MAP

File Name	Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	—	—	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—	—	ADC1MD	0000
PMD2	0772	—	—	—	IC5MD	IC4MD	IC3MD	IC2MD	IC1MD	—	—	—	OC5MD	OC4MD	OC3MD	OC2MD	OC1MD	0000
PMD3	0774	—	—	—	—	—	CMPMD	RTCCMD	PMPMD	CRCPMD	—	—	—	—	—	I2C2MD	—	0000

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

EXAMPLE 5-2: LOADING THE WRITE BUFFERS

```
; Set up NVMCON for row programming operations
       MOV      #0x4001, W0                    ;
       MOV      W0, NVMCON                     ; Initialize NVMCON
; Set up a pointer to the first program memory location to be written
; program memory selected, and writes enabled
       MOV      #0x0000, W0                    ;
       MOV      W0, TBLPAG                     ; Initialize PM Page Boundary SFR
       MOV      #0x6000, W0                    ; An example program memory address
; Perform the TBLWT instructions to write the latches
; 0th_program_word
       MOV      #LOW_WORD_0, W2                ;
       MOV      #HIGH_BYTE_0, W3               ;
       TBLWTL   W2, [W0]                       ; Write PM low word into program latch
       TBLWTH   W3, [W0++]                     ; Write PM high byte into program latch
; 1st_program_word
       MOV      #LOW_WORD_1, W2                ;
       MOV      #HIGH_BYTE_1, W3               ;
       TBLWTL   W2, [W0]                       ; Write PM low word into program latch
       TBLWTH   W3, [W0++]                     ; Write PM high byte into program latch
; 2nd_program_word
       MOV      #LOW_WORD_2, W2                ;
       MOV      #HIGH_BYTE_2, W3               ;
       TBLWTL   W2, [W0]                       ; Write PM low word into program latch
       TBLWTH   W3, [W0++]                     ; Write PM high byte into program latch
       •
       •
       •
; 63rd_program_word
       MOV      #LOW_WORD_31, W2               ;
       MOV      #HIGH_BYTE_31, W3              ;
       TBLWTL   W2, [W0]                       ; Write PM low word into program latch
       TBLWTH   W3, [W0]                       ; Write PM high byte into program latch
```

EXAMPLE 5-3: INITIATING A PROGRAMMING SEQUENCE

```
       DISI   #5                       ; Block all interrupts with priority <7
                                       ; for next 5 instructions
       MOV    #0x55, W0
       MOV    W0, NVMKEY               ; Write the 55 key
       MOV    #0xAA, W1                ;
       MOV    W1, NVMKEY               ; Write the AA key
       BSET   NVMCON, #WR              ; Start the erase sequence
       NOP                             ; 2 NOPs required after setting WR
       NOP                             ;
       BTSC   NVMCON, #15              ; Wait for the sequence to be completed
       BRA    $-2                      ;
```

REGISTER 7-29: IPC16: INTERRUPT PRIORITY CONTROL REGISTER 16

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	CRCIP2	CRCIP1	CRCIP0	—	U2ERIP2	U2ERIP1	U2ERIP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	U1ERIP2	U1ERIP1	U1ERIP0	—	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-12 **CRCIP<2:0>:** CRC Generator Error Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **U2ERIP<2:0>:** UART2 Error Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **U1ERIP<2:0>:** UART1 Error Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 3-0 **Unimplemented:** Read as ‘0’

7.4 Interrupt Setup Procedures

7.4.1 INITIALIZATION

To configure an interrupt source:

1. Set the NSTDIS Control bit (INTCON1<15>) if nested interrupts are not desired.
2. Select the user-assigned priority level for the interrupt source by writing the control bits in the appropriate IPCx register. The priority level will depend on the specific application and type of interrupt source. If multiple priority levels are not desired, the IPCx register control bits for all enabled interrupt sources may be programmed to the same non-zero value.

Note: At a device Reset, the IPCx registers are initialized, such that all user interrupt sources are assigned to Priority Level 4.

3. Clear the interrupt flag status bit associated with the peripheral in the associated IFSx register.
4. Enable the interrupt source by setting the interrupt enable control bit associated with the source in the appropriate IECx register.

7.4.2 INTERRUPT SERVICE ROUTINE

The method that is used to declare an ISR and initialize the IVT with the correct vector address will depend on the programming language (i.e., 'C' or assembler) and the language development toolsuite that is used to develop the application. In general, the user must clear the interrupt flag in the appropriate IFSx register for the source of the interrupt that the ISR handles. Otherwise, the ISR will be re-entered immediately after exiting the routine. If the ISR is coded in assembly language, it must be terminated using a `RETFIE` instruction to unstack the saved PC value, SRL value and old CPU priority level.

7.4.3 TRAP SERVICE ROUTINE

A Trap Service Routine (TSR) is coded like an ISR, except that the appropriate trap status flag in the INTCON1 register must be cleared to avoid re-entry into the TSR.

7.4.4 INTERRUPT DISABLE

All user interrupts can be disabled using the following procedure:

1. Push the current SR value onto the software stack using the `PUSH` instruction.
2. Force the CPU to Priority Level 7 by inclusive ORing the value, OEh, with SRL.

To enable user interrupts, the `POP` instruction may be used to restore the previous SR value.

Note that only user interrupts with a priority level of 7 or less can be disabled. Trap sources (Levels 8-15) cannot be disabled.

The `DISI` instruction provides a convenient way to disable interrupts of Priority Levels 1-6 for a fixed period of time. Level 7 interrupt sources are not disabled by the `DISI` instruction.

REGISTER 10-9: RPINR18: PERIPHERAL PIN SELECT INPUT REGISTER 18

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	U1CTSR4	U1CTSR3	U1CTSR2	U1CTSR1	U1CTSR0
bit 15							bit 8

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	U1RXR4	U1RXR3	U1RXR2	U1RXR1	U1RXR0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as ‘0’

bit 12-8 **U1CTSR<4:0>:** Assign UART1 Clear-to-Send ($\overline{\text{U1CTS}}$) to the Corresponding RPn Pin bits

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **U1RXR<4:0>:** Assign UART1 Receive (U1RX) to the Corresponding RPn Pin bits

REGISTER 10-10: RPINR19: PERIPHERAL PIN SELECT INPUT REGISTER 19

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	U2CTSR4	U2CTSR3	U2CTSR2	U2CTSR1	U2CTSR0
bit 15							bit 8

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	U2RXR4	U2RXR3	U2RXR2	U2RXR1	U2RXR0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as ‘0’

bit 12-8 **U2CTSR<4:0>:** Assign UART2 Clear-to-Send ($\overline{\text{U2CTS}}$) to the Corresponding RPn Pin bits

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **U2RXR<4:0>:** Assign UART2 Receive (U2RX) to the Corresponding RPn Pin bits

12.0 TIMER2/3 AND TIMER4/5

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the *"PIC24F Family Reference Manual"*, **"Timers"** (DS39704).

The Timer2/3 and Timer4/5 modules are 32-bit timers, which can also be configured as four independent, 16-bit timers with selectable operating modes.

As a 32-bit timer, Timer2/3 and Timer4/5 operate in three modes:

- Two independent, 16-bit timers (Timer2 and Timer3) with all 16-bit operating modes (except Asynchronous Counter mode)
- Single 32-bit timer
- Single 32-bit synchronous counter

They also support these features:

- Timer gate operation
- Selectable prescaler settings
- Timer operation during Idle and Sleep modes
- Interrupt on a 32-Bit Period register match
- A/D Event Trigger (Timer2/3 only)

Individually, all four of the 16-bit timers can function as synchronous timers or counters. They also offer the features listed above, except for the A/D Event Trigger; this is implemented only with Timer3. The operating modes and enabled features are determined by setting the appropriate bit(s) in the T2CON, T3CON, T4CON and T5CON registers. T2CON and T4CON are shown in generic form in Register 12-1; T3CON and T5CON are shown in generic form in Register 12-2.

For 32-bit timer/counter operation, Timer2 and Timer4 are the least significant word; Timer3 and Timer4 are the most significant word of the 32-bit timers.

Note: For 32-bit operation, T3CON and T5CON control bits are ignored. Only T2CON and T4CON control bits are used for setup and control. Timer2 and Timer4 clock and gate inputs are utilized for the 32-bit timer modules, but an interrupt is generated with the Timer3 or Timer5 interrupt flags.

To configure Timer2/3 or Timer4/5 for 32-bit operation:

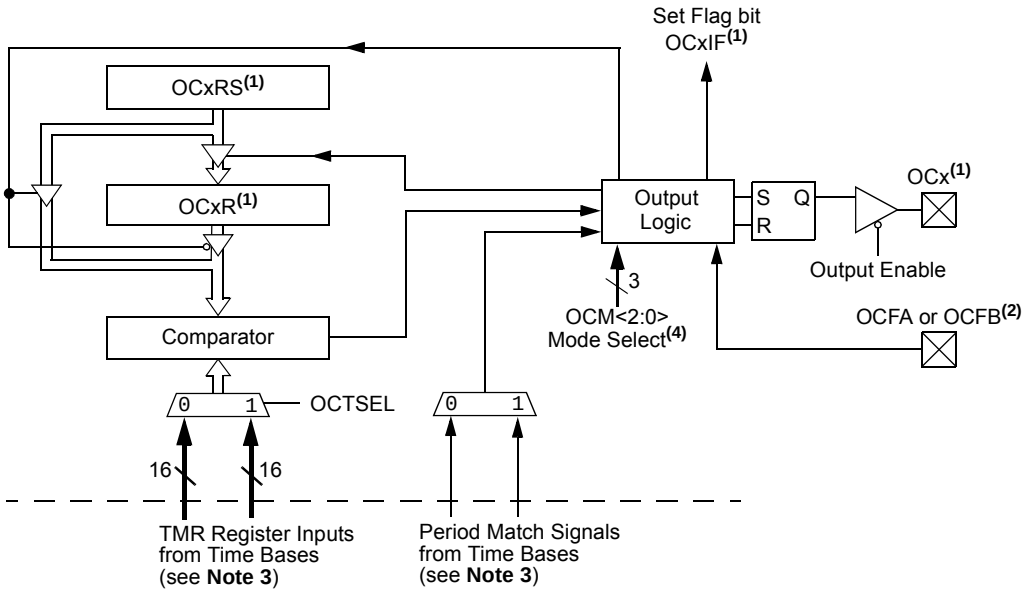
1. Set the T32 bit (T2CON<3> or T4CON<3> = 1).
2. Select the prescaler ratio for Timer2 or Timer4 using the TCKPS<1:0> bits.
3. Set the Clock and Gating modes using the TCS and TGATE bits. If TCS is set to the external clock, RPINRx (TxCK) must be configured to an available RPn pin. See **Section 10.4 "Peripheral Pin Select (PPS)"** for more information.
4. Load the timer period value. PR3 (or PR5) will contain the most significant word of the value while PR2 (or PR4) contains the least significant word.
5. If interrupts are required, set the Timer3/5 Interrupt Enable bit, T3IE or T5IE; use the priority bits, T3IP<2:0> or T5IP<2:0>, to set the interrupt priority. Note that while Timer2 or Timer4 controls the timer, the interrupt appears as a Timer3 or Timer5 interrupt.
6. Set the TON bit (= 1).

The timer value, at any point, is stored in the register pair, TMR3:TMR2 (or TMR5:TMR4). TMR3 (TMR5) always contains the most significant word of the count, while TMR2 (TMR4) contains the least significant word.

To configure any of the timers for individual 16-bit operation:

1. Clear the T32 bit corresponding to that timer (T2CON<3> for Timer2 and Timer3 or T4CON<3> for Timer4 and Timer5).
2. Select the timer prescaler ratio using the TCKPS<1:0> bits.
3. Set the Clock and Gating modes using the TCS and TGATE bits. See **Section 10.4 "Peripheral Pin Select (PPS)"** for more information.
4. Load the timer period value into the PRx register.
5. If interrupts are required, set the Timerx Interrupt Enable bit, TxIE; use the priority bits, TxIP<2:0>, to set the interrupt priority.
6. Set the TON bit (TxCON<15> = 1).

FIGURE 14-1: OUTPUT COMPARE x MODULE BLOCK DIAGRAM



Note 1: Where 'x' is shown, reference is made to the registers associated with the respective Output Compare Channels 1 through 5.

2: The OCFA pin controls the OC1-OC4 channels. The OCFB pin controls the OC5 channel.

3: Each output compare channel can use one of two selectable time bases. Refer to the device data sheet for the time bases associated with the module.

4: This peripheral's inputs and outputs must be assigned to an available RPn pin before use. Please see **Section 10.4 "Peripheral Pin Select (PPS)"** for more information.

REGISTER 15-2: SPIxCON1: SPIx CONTROL REGISTER 1

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	DISSCK(1)	DISSDO(2)	MODE16	SMP	CKE(3)
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SSEN(4)	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as '0'

bit 12 **DISSCK:** Disables SCKx Pin bit (SPI Master modes only)(1)
- 1 = Internal SPI clock is disabled; pin functions as I/O
- 0 = Internal SPI clock is enabled

bit 11 **DISSDO:** Disables SDOx Pin bit(2)
- 1 = SDOx pin is not used by the module; pin functions as I/O
- 0 = SDOx pin is controlled by the module

bit 10 **MODE16:** Word/Byte Communication Select bit
- 1 = Communication is word-wide (16 bits)
- 0 = Communication is byte-wide (8 bits)

bit 9 **SMP:** SPIx Data Input Sample Phase bit

Master mode:
- 1 = Input data is sampled at end of data output time
- 0 = Input data is sampled at middle of data output time

Slave mode:
SMP must be cleared when SPIx is used in Slave mode.

bit 8 **CKE:** SPIx Clock Edge Select bit(3)
- 1 = Serial output data changes on transition from active clock state to Idle clock state (see bit 6)
- 0 = Serial output data changes on transition from Idle clock state to active clock state (see bit 6)

bit 7 **SSEN:** Slave Select Enable bit (Slave mode)(4)
- 1 = $\overline{SSx}$ pin is used for Slave mode
- 0 = $\overline{SSx}$ pin is not used by the module; pin is controlled by port function

bit 6 **CKP:** Clock Polarity Select bit
- 1 = Idle state for the clock is a high level; active state is a low level
- 0 = Idle state for the clock is a low level; active state is a high level

bit 5 **MSTEN:** Master Mode Enable bit
- 1 = Master mode
- 0 = Slave mode

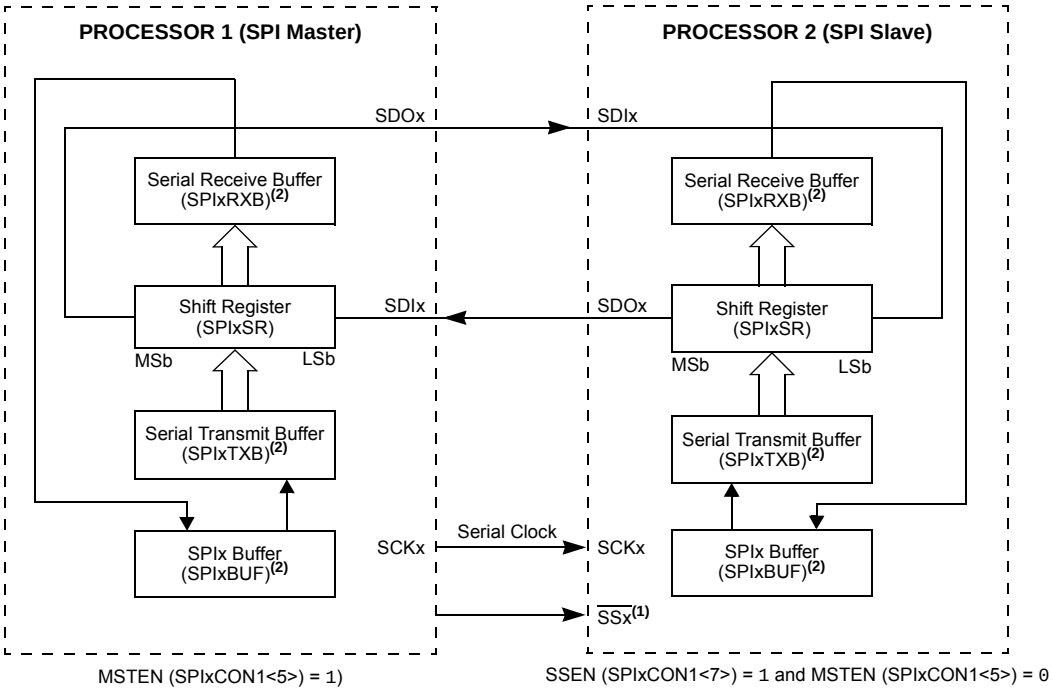
Note 1: If DISSCK = 0, SCKx must be configured to an available RPn pin. See **Section 10.4 "Peripheral Pin Select (PPS)"** for more information.

2: If DISSDO = 0, SDOx must be configured to an available RPn pin. See **Section 10.4 "Peripheral Pin Select (PPS)"** for more information.

3: The CKE bit is not used in the Framed SPI modes. The user should program this bit to '0' for the Framed SPI modes (FRMEN = 1).

4: If SSEN = 1, $\overline{SSx}$ must be configured to an available RPn pin. See **Section 10.4 "Peripheral Pin Select (PPS)"** for more information.

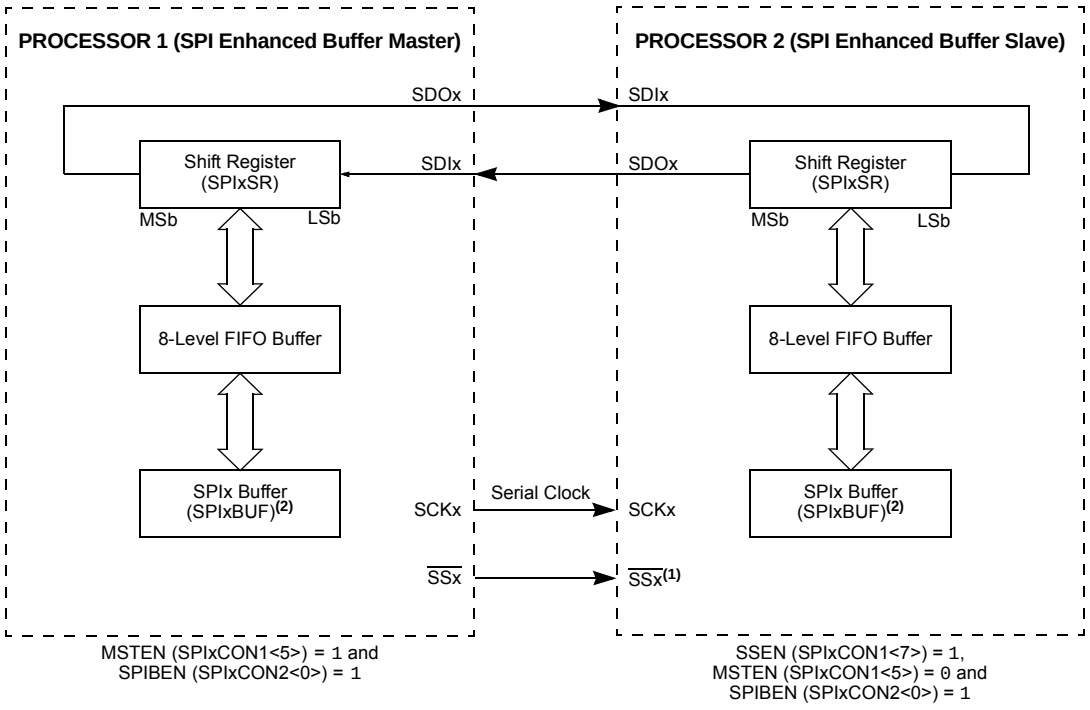
FIGURE 15-3: SPIx MASTER/SLAVE CONNECTION (STANDARD MODE)



Note 1: Using the $\overline{SSx}$ pin in Slave mode of operation is optional.

2: User must write transmit data to read received data from SPIxBUF. The SPIxTXB and SPIxRXB registers are memory mapped to SPIxBUF.

FIGURE 15-4: SPIx MASTER/SLAVE CONNECTION (ENHANCED BUFFER MODES)



Note 1: Using the $\overline{SSx}$ pin in Slave mode of operation is optional.

2: User must write transmit data to read received data from SPIxBUF. The SPIxTXB and SPIxRXB registers are memory mapped to SPIxBUF.

REGISTER 16-2: I2CxSTAT: I2Cx STATUS REGISTER (CONTINUED)

bit 5 **D/$\overline{A}$:** Data/$\overline{Address}$ bit (when operating as I^2C slave)
1 = Indicates that the last byte received was data
0 = Indicates that the last byte received was a device address
Hardware is clear at a device address match. Hardware is set by a write to I2CxTRN or by reception of a slave byte.

bit 4 **P:** Stop bit
1 = Indicates that a Stop bit has been detected last
0 = Stop bit was not detected last
Hardware is set or clear when Start, Repeated Start or Stop is detected.

bit 3 **S:** Start bit
1 = Indicates that a Start (or Repeated Start) bit has been detected last
0 = Start bit was not detected last
Hardware is set or clear when Start, Repeated Start or Stop is detected.

bit 2 **R/$\overline{W}$:** Read/$\overline{Write}$ Information bit (when operating as I^2C slave)
1 = Read – Indicates data transfer is output from slave
0 = Write – Indicates data transfer is input to slave
Hardware is set or clear after reception of an I^2C device address byte.

bit 1 **RBF:** Receive Buffer Full Status bit
1 = Receive is complete, I2CxRCV is full
0 = Receive is not complete, I2CxRCV is empty
Hardware is set when I2CxRCV is written with received byte. Hardware is clear when software reads I2CxRCV.

bit 0 **TBF:** Transmit Buffer Full Status bit
1 = Transmit is in progress, I2CxTRN is full
0 = Transmit is complete, I2CxTRN is empty
Hardware is set when software writes I2CxTRN. Hardware is clear at completion of data transmission.

Note 1: In both Master and Slave modes, the ACKSTAT bit is only updated when transmitting data resulting in the reception of an ACK or NACK from another device. Do not check the state of ACKSTAT when receiving data, either as a slave or a master. Reading ACKSTAT after receiving address or data bytes returns an invalid result.

REGISTER 16-3: I2CxMSK: I2Cx SLAVE MODE ADDRESS MASK REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	AMSK9	AMSK8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
AMSK7	AMSK6	AMSK5	AMSK4	AMSK3	AMSK2	AMSK1	AMSK0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-10 **Unimplemented:** Read as ‘0’

bit 9-0 **AMSK<9:0>:** Mask for Address Bit x Select bits
1 = Enables masking for bit x of incoming message address; bit match is not required in this position
0 = Disables masking for bit x; bit match is required in this position

REGISTER 17-1: UxMODE: UARTx MODE REGISTER (CONTINUED)

bit 4 **RXINV:** Receive Polarity Inversion bit
1 = UxRX Idle state is '0'
0 = UxRX Idle state is '1'

bit 3 **BRGH:** High Baud Rate Enable bit
1 = BRG generates 4 clocks per bit period (4x baud clock, High-Speed mode)
0 = BRG generates 16 clocks per bit period (16x baud clock, Standard mode)

bit 2-1 **PDSEL<1:0>:** Parity and Data Selection bits
11 = 9-bit data, no parity
10 = 8-bit data, odd parity
01 = 8-bit data, even parity
00 = 8-bit data, no parity

bit 0 **STSEL:** Stop Bit Selection bit
1 = Two Stop bits
0 = One Stop bit

Note 1: If UARTEN = 1, the peripheral inputs and outputs must be configured to an available RPn pin. See **Section 10.4 "Peripheral Pin Select (PPS)"** for more information.

2: This feature is only available for the 16x BRG mode (BRGH = 0).

3: Bit availability depends on pin availability.

REGISTER 17-3: UxTXREG: UARTx TRANSMIT REGISTER

U-x	U-x	U-x	U-x	U-x	U-x	U-x	W-x
—	—	—	—	—	—	—	UTX8
bit 15							bit 8

W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
UTX7	UTX6	UTX5	UTX4	UTX3	UTX2	UTX1	UTX0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-9 **Unimplemented:** Read as '0'

bit 8 **UTX8:** UARTx Data of the Transmitted Character bit (in 9-bit mode)

bit 7-0 **UTX<7:0>:** UARTx Data of the Transmitted Character bits

REGISTER 17-4: UxRXREG: UARTx RECEIVE REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-0
—	—	—	—	—	—	—	URX8
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
URX7	URX6	URX5	URX4	URX3	URX2	URX1	URX0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-9 **Unimplemented:** Read as '0'

bit 8 **URX8:** UARTx Data of the Received Character bit (in 9-bit mode)

bit 7-0 **URX<7:0>:** UARTx Data of the Received Character bits

REGISTER 18-1: PMCON: PARALLEL PORT CONTROL REGISTER (CONTINUED)

bit 2 **BEP:** Byte Enable Polarity bit
1 = Byte enable is active-high (PMBE)
0 = Byte enable is active-low ($\overline{PMBE}$)

bit 1 **WRSP:** Write Strobe Polarity bit
For Slave Modes and Master Mode 2 (PMMODE<9:8> = 00, 01, 10):
1 = Write strobe is active-high (PMWR)
0 = Write strobe is active-low ($\overline{PMWR}$)
For Master Mode 1 (PMMODE<9:8> = 11):
1 = Enable strobe is active-high (PMENB)
0 = Enable strobe is active-low ($\overline{PMENB}$)

bit 0 **RDSP:** Read Strobe Polarity bit
For Slave Modes and Master Mode 2 (PMMODE<9:8> = 00, 01, 10):
1 = Read strobe is active-high (PMRD)
0 = Read strobe is active-low ($\overline{PMRD}$)
For Master Mode 1 (PMMODE<9:8> = 11):
1 = Read/write strobe is active-high (PMRD/$\overline{PMWR}$)
0 = Read/write strobe is active-low ($\overline{PMRD}$/$\overline{PMWR}$)

Note 1: PMA<10:2> bits are not available on 28-pin devices.
2: These bits have no effect when their corresponding pins are used as address lines.

REGISTER 18-3: PMADDR: PARALLEL PORT ADDRESS REGISTER

U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	CS1	—	—	—	ADDR10(1)	ADDR9(1)	ADDR8(1)
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADDR7(1)	ADDR6(1)	ADDR5(1)	ADDR4(1)	ADDR3(1)	ADDR2(1)	ADDR1(1)	ADDR0(1)
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as '0'

bit 14 **CS1:** Chip Select 1 bit
1 = Chip Select 1 is active
0 = Chip Select 1 is inactive

bit 13-11 **Unimplemented:** Read as '0'

bit 10-0 **ADDR<10:0>:** Parallel Port Destination Address bits(1)

Note 1: PMA<10:2> bits are not available on 28-pin devices.

REGISTER 18-4: PMAEN: PARALLEL PORT ENABLE REGISTER

U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	PTEN14	—	—	—	PTEN10(1)	PTEN9(1)	PTEN8(1)
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTEN7(1)	PTEN6(1)	PTEN5(1)	PTEN4(1)	PTEN3(1)	PTEN2(1)	PTEN1	PTEN0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as '0'

bit 14 **PTEN14:** PMCS1 Strobe Enable bit
1 = PMCS1 pin functions as chip select
0 = PMCS1 pin functions as port I/O

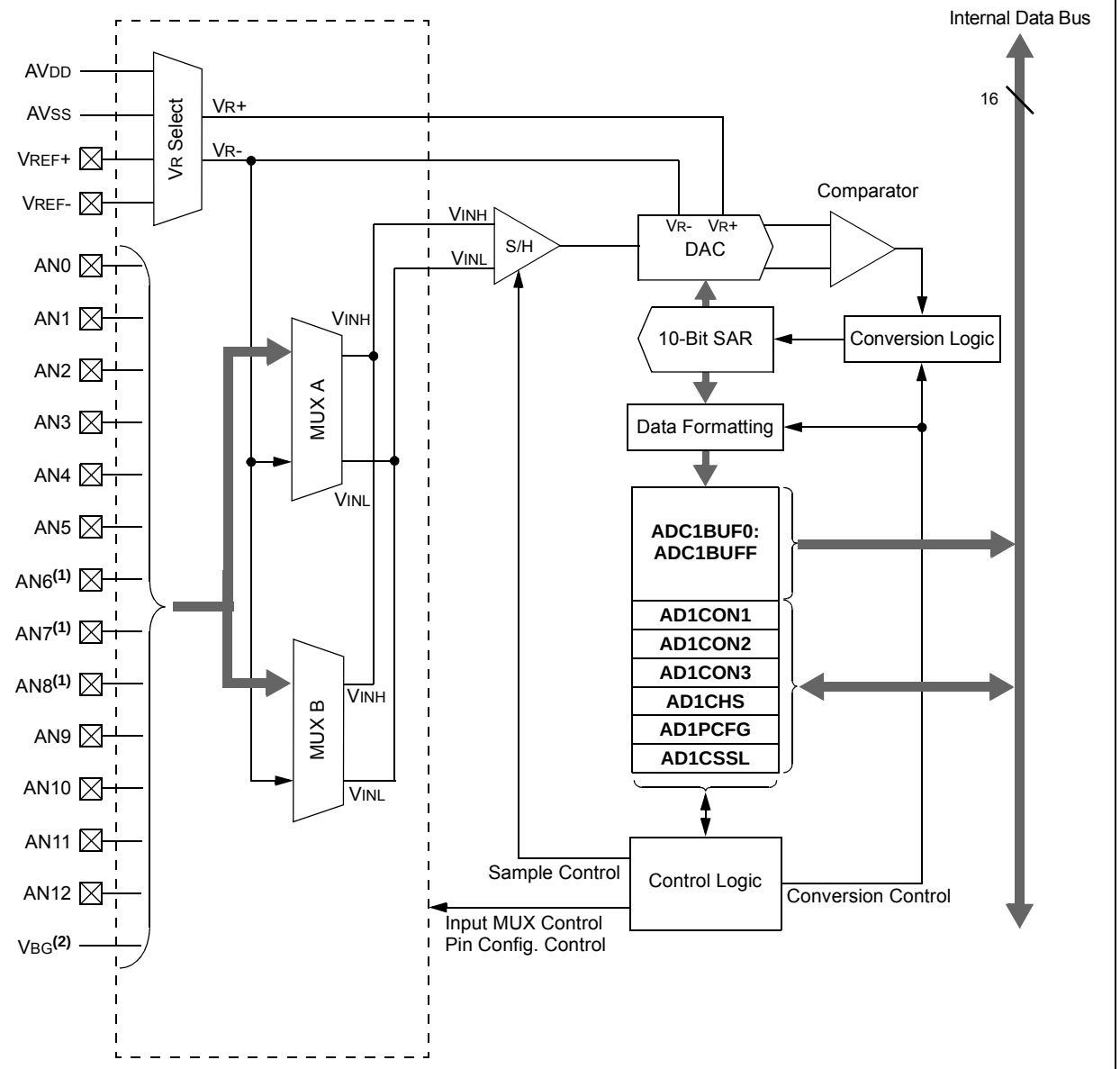
bit 13-11 **Unimplemented:** Read as '0'

bit 10-2 **PTEN<10:2>:** PMP Address Port Enable bits(1)
1 = PMA<10:2> function as PMP address lines
0 = PMA<10:2> function as port I/O

bit 1-0 **PTEN<1:0>:** PMALH/PMALL Strobe Enable bits
1 = PMA1 and PMA0 function as either PMA<1:0> or PMALH and PMALL
0 = PMA1 and PMA0 pads functions as port I/O

Note 1: PMA<10:2> bits are not available on 28-pin devices.

FIGURE 21-1: 10-BIT HIGH-SPEED A/D CONVERTER BLOCK DIAGRAM



Note 1: Analog Channels, AN6 through AN8, are available on 44-pin devices only.

2: Band Gap Voltage (V_{BG}) reference is internally connected to Analog Channel AN15, which does not appear on any pin.

25.6 MPLAB X SIM Software Simulator

The MPLAB X SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB X SIM Software Simulator fully supports symbolic debugging using the MPLAB XC Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

25.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to three meters) interconnection cables.

25.8 MPLAB ICD 3 In-Circuit Debugger System

The MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost-effective, high-speed hardware debugger/programmer for Microchip Flash DSC and MCU devices. It debugs and programs PIC Flash microcontrollers and dsPIC DSCs with the powerful, yet easy-to-use graphical user interface of the MPLAB IDE.

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

25.9 PICkit 3 In-Circuit Debugger/ Programmer

The MPLAB PICkit 3 allows debugging and programming of PIC and dsPIC Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB IDE. The MPLAB PICkit 3 is connected to the design engineer's PC using a full-speed USB interface and can be connected to the target via a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the Reset line to implement in-circuit debugging and In-Circuit Serial Programming™ (ICSP™).

25.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at V_{DDMIN} and V_{DDMAX} for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.

TABLE 27-6: DC CHARACTERISTICS: POWER-DOWN CURRENT (I_{PD})

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended			
Parameter No.	**Typical(1)**	**Max**	**Units**	**Conditions**		
Power-Down Current (I_{PD}): PMD Bits are Set, PMSLP Bit is '0'(2)						
DC60	0.1	1	µA	-40°C	2.0V(3)	Base Power-Down Current(5)
DC60a	0.15	1	µA	+25°C		
DC60m	2.2	7.4	µA	+60°C		
DC60b	3.7	12	µA	+85°C		
DC60j	15	50	µA	+125°C		
DC60c	0.2	1	µA	-40°C	2.5V(3)	
DC60d	0.25	1	µA	+25°C		
DC60n	2.6	15	µA	+60°C		
DC60e	4.2	25	µA	+85°C		
DC60k	16	100	µA	+125°C		
DC60f	3.3	9	µA	-40°C	3.3V(4)	
DC60g	3.5	10	µA	+25°C		
DC60o	6.7	22	µA	+60°C		
DC60h	9	30	µA	+85°C		
DC60l	36	120	µA	+125°C		
DC61	1.75	3	µA	-40°C	2.0V(3)	Watchdog Timer Current: ΔI_{WDT}(5)
DC61a	1.75	3	µA	+25°C		
DC61m	1.75	3	µA	+60°C		
DC61b	1.75	3	µA	+85°C		
DC61j	3.5	6	µA	+125°C		
DC61c	2.4	4	µA	-40°C	2.5V(3)	
DC61d	2.4	4	µA	+25°C		
DC61n	2.4	4	µA	+60°C		
DC61e	2.4	4	µA	+85°C		
DC61k	4.8	8	µA	+125°C		
DC61f	2.8	5	µA	-40°C	3.3V(4)	
DC61g	2.8	5	µA	+25°C		
DC61o	2.8	5	µA	+60°C		
DC61h	2.8	5	µA	+85°C		
DC61l	5.6	10	µA	+125°C		

Note 1: Data in the Typical column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

2: Base I_{PD} is measured with all peripherals and clocks shut down. All I/Os are configured as inputs and pulled high. WDT, etc., are all switched off.

3: On-chip voltage regulator is disabled (DISVREG tied to V_{DD}).

4: On-chip voltage regulator is enabled (DISVREG tied to V_{SS}). Low-Voltage Detect (LVD) and Brown-out Detect (BOD) are enabled.

5: The Δ current is the additional current consumed when the module is enabled. This current should be added to the base I_{PD} current.

FIGURE 27-4: EXTERNAL CLOCK TIMING

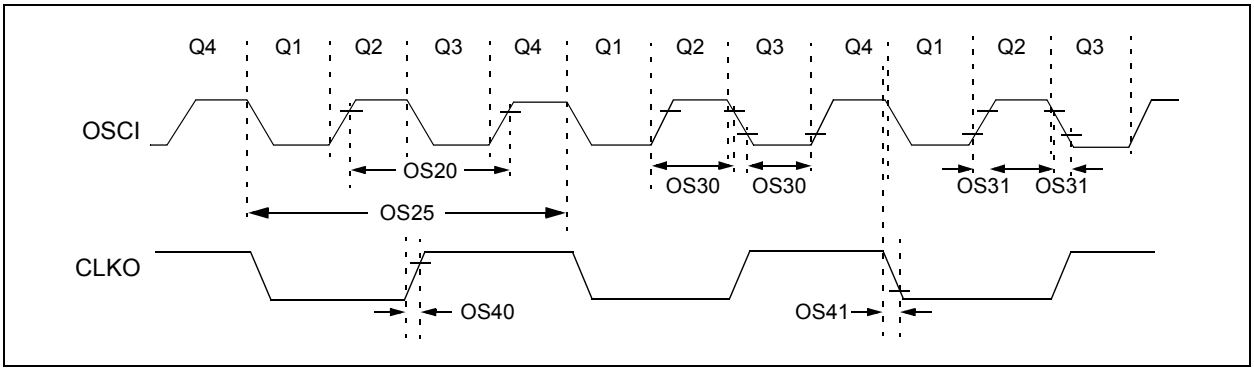


TABLE 27-15: EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0 to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Sym**	**Characteristic**	**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**
OS10	FOSC	External CLKI Frequency (External clocks allowed only in EC mode)	DC	—	32	MHz	EC, -40°C ≤ TA ≤ +85°C
			4	—	8	MHz	ECPLL, -40°C ≤ TA ≤ +85°C
			DC	—	24	MHz	EC, -40°C ≤ TA ≤ +125°C
			4	—	6	MHz	ECPLL, -40°C ≤ TA ≤ +125°C
		Oscillator Frequency	3	—	10	MHz	XT
			3	—	8	MHz	XTPLL, -40°C ≤ TA ≤ +85°C
			10	—	32	MHz	HS, -40°C ≤ TA ≤ +85°C
			31	—	33	kHz	SOSC
			3	—	6	MHz	XTPLL, -40°C ≤ TA ≤ +125°C
			10	—	24	MHz	HS, -40°C ≤ TA ≤ +125°C
OS20	TOSC	TOSC = 1/FOSC	—	—	—	—	See Parameter OS10 for FOSC value
OS25	TCY	Instruction Cycle Time(2)	62.5	—	DC	ns	
OS30	TosL, TosH	External Clock In (OSCI) High or Low Time	0.45 x TOSC	—	—	ns	EC
OS31	TosR, TosF	External Clock In (OSCI) Rise or Fall Time	—	—	20	ns	EC
OS40	TckR	CLKO Rise Time(3)	—	6	10	ns	
OS41	TckF	CLKO Fall Time(3)	—	6	10	ns	

Note 1: Data in "Typ" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

2: Instruction cycle period (TCY) equals two times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type, under standard operating conditions, with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "Min." values with an external clock applied to the OSCI/CLKI pin. When an external clock input is used, the "Max." cycle time limit is "DC" (no clock) for all devices.

3: Measurements are taken in EC mode. The CLKO signal is measured on the OSCO pin. CLKO is low for the Q1-Q2 period (1/2 TCY) and high for the Q3-Q4 period (1/2 TCY).

TABLE 27-16: PLL CLOCK TIMING SPECIFICATIONS (V_{DD} = 2.0V TO 3.6V)

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Sym**	**Characteristic(1)**	**Min**	**Typ(2)**	**Max**	**Units**	**Conditions**
OS50	FPLLI	PLL Input Frequency Range	3	—	8	MHz	ECPLL, HSPLL, XTPLL modes, -40°C ≤ TA ≤ +85°C
			3	—	6	MHz	ECPLL, HSPLL, XTPLL modes, -40°C ≤ TA ≤ +125°C
OS51	FSYS	PLL Output Frequency Range	8	—	32	MHz	-40°C ≤ TA ≤ +85°C
			8	—	24	MHz	-40°C ≤ TA ≤ +125°C
OS52	TLOCK	PLL Start-up Time (Lock Time)	—	—	2	ms	
OS53	DCLK	CLKO Stability (Jitter)	-2	1	2	%	Measured over 100 ms period

Note 1: These parameters are characterized but not tested in manufacturing.

2: Data in "Typ" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

TABLE 27-17: INTERNAL RC OSCILLATOR SPECIFICATIONS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Sym**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
	TFRC	FRC Start-up Time	—	15	—	μs	
	TLPRC	LPRC Start-up Time	—	40	—	μs	

TABLE 27-18: AC CHARACTERISTICS: INTERNAL RC ACCURACY

AC CHARACTERISTICS		Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended					
Param No.	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**	
F20	Internal FRC @ 8 MHz(1)	-2	—	2	%	+25°C	3.0V ≤ VDD ≤ 3.6V
		-5	—	5	%	-40°C ≤ TA ≤ +85°C	
		-7	—	7	%	+125°C	
F21	LPRC @ 31 kHz(2)	-15	—	15	%	+25°C	3.0V ≤ VDD ≤ 3.6V
		-15	—	15	%	-40°C ≤ TA ≤ +85°C	
		-30	—	30	%	+125°C	

Note 1: Frequency calibrated at +25°C and 3.3V. OSCTUN bits can be used to compensate for temperature drift.

2: Change of LPRC frequency as VDD changes.

READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this document.

TO: Technical Publications Manager
RE: Reader Response

Total Pages Sent _____

From: Name _____

Company _____

Address _____

City / State / ZIP / Country _____

Telephone: (_____) _____ - _____ FAX: (_____) _____ - _____

Application (optional):

Would you like a reply? ___ Y ___ N

Device: PIC24FJ64GA004 Family Literature Number: DS39881E

Questions:

1. What are the best features of this document?

2. How does this document meet your hardware and software development needs?

3. Do you find the organization of this document easy to follow? If not, why?

4. What additions to the document do you think would enhance the structure and subject?

5. What deletions from the document could be made without affecting the overall usefulness?

6. Is there any incorrect or misleading information (what and where)?

7. How would you improve this document?