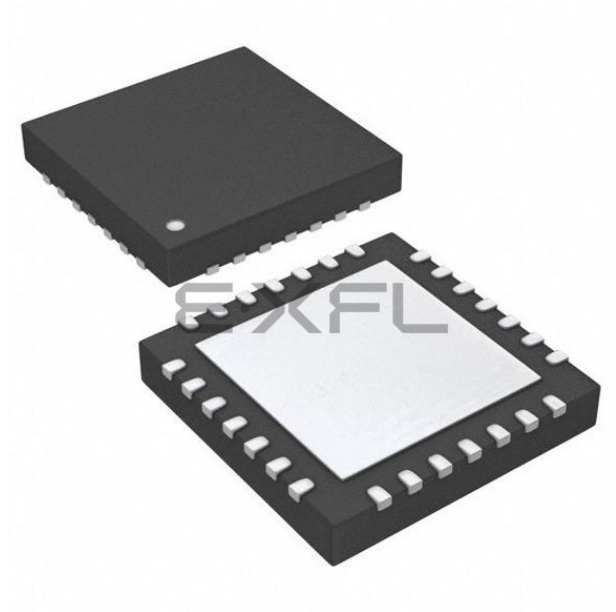


Welcome to E-XFL.COM

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	8-Bit
Speed	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PSMC, PWM, WDT
Number of I/O	24
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 11x12b; D/A 1x8b
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/pic16lf1782t-i-ml

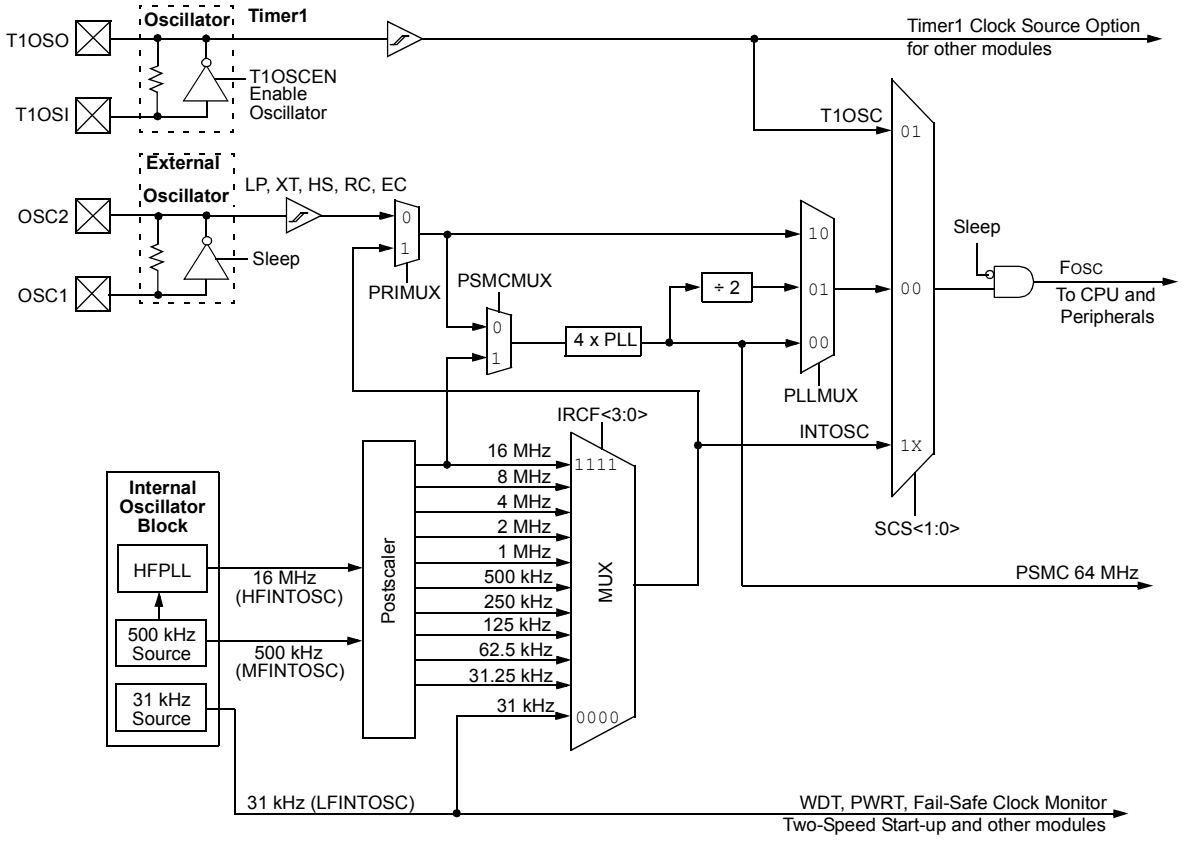
TABLE 3-8: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Bank 16 (Continued)											
831h	PSMC2CON	PSMC2EN	PSMC2LD	PSMC2DBFE	PSMC2DBRE	P2MODE<3:0>				0000 0000	0000 0000
832h	PSMC2MDL	P2MDLEN	P2MDLPOL	P2MDLBIT	—	P2MSRC<3:0>				000- 0000	000- 0000
833h	PSMC2SYNC	—	—	—	—	—	—	P2SYNC<1:0>		---- --00	---- --00
834h	PSMC2CLK	—	—	P2CPRE<1:0>		—	—	P2CSRC<1:0>		--00 --00	--00 --00
835h	PSMC2OEN	—	—	—	—	—	—	P2OEB	P2OEA	---- --00	---- --00
836h	PSMC2POL	—	P2INPOL	—	—	—	—	P2POLB	P2POLA	-0-- --00	-0-- --00
837h	PSMC2BLNK	—	—	P2FEBM<1:0>		—	—	P2REBM<1:0>		--00 --00	--00 --00
838h	PSMC2REBS	P2REBIN	—	—	—	P2REBSC3	P2REBSC2	P2REBSC1	—	0--- 000-	0--- 000-
839h	PSMC2FEBS	P2FEBIN	—	—	—	P2FEBSC3	P2FEBSC2	P2FEBSC1	—	0--- 000-	0--- 000-
83Ah	PSMC2PHS	P2PHSIN	—	—	—	P2PHSC3	P2PHSC2	P2PHSC1	P2PHST	0--- 0000	0--- 0000
83Bh	PSMC2DCS	P2DCSIN	—	—	—	P2DCSC3	P2DCSC2	P2DCSC1	P2DCST	0--- 0000	0--- 0000
83Ch	PSMC2PRS	P2PRSIN	—	—	—	P2PRSC3	P2PRSC2	P2PRSC1	P2PRST	0--- 0000	0--- 0000
83Dh	PSMC2ASDC	P2ASE	P2ASDEN	P2ARSEN	—	—	—	—	P2ASDOV	000- ---0	000- ---0
83Eh	PSMC2ASDL	—	—	P2ASDLF	P2ASDLE	P2ASDLD	P2ASDLC	P2ASDLB	P2ASDLA	--00 0000	--00 0000
83Fh	PSMC2ASDS	P2ASDSIN	—	—	—	P2ASDSC3	P2ASDSC2	P2ASDSC1	—	0--- 000-	0--- 000-
840h	PSMC2INT	P2TOVIE	P2TPHIE	P2TDCIE	P2TPRIE	P2TOVIF	P2TPHIF	P2TDCIF	P2TPRIF	0000 0000	0000 0000
841h	PSMC2PHL	Phase Low Count								0000 0000	0000 0000
842h	PSMC2PHH	Phase High Count								0000 0000	0000 0000
843h	PSMC2DCL	Duty Cycle Low Count								0000 0000	0000 0000
844h	PSMC2DCH	Duty Cycle High Count								0000 0000	0000 0000
845h	PSMC2PRL	Period Low Count								0000 0000	0000 0000
846h	PSMC2PRH	Period High Count								0000 0000	0000 0000
847h	PSMC2TMRL	Time base Low Counter								0000 0001	0000 0001
848h	PSMC2TMRH	Time base High Counter								0000 0000	0000 0000
849h	PSMC2DBR	rising Edge Dead-band Counter								0000 0000	0000 0000
84Ah	PSMC2DBF	Falling Edge Dead-band Counter								0000 0000	0000 0000
84Bh	PSMC2BLKR	rising Edge Blanking Counter								0000 0000	0000 0000
84Ch	PSMC2BLKF	Falling Edge Blanking Counter								0000 0000	0000 0000
84Dh	PSMC2FFA	—	—	—	—	Fractional Frequency Adjust Register				---- 0000	---- 0000
84Eh	PSMC2STR0	—	—	—	—	—	—	P2STRB	P2STRA	---- --01	---- --01
84Fh	PSMC2STR1	P2SYNC	—	—	—	—	—	P2LSMEN	P2HSMEN	0--- --00	0--- --00
850h — 86Fh	—	Unimplemented								—	—
Bank 17-30											
x0Ch or x8Ch to x1Fh or x9Fh	—	Unimplemented								—	—

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, read as '0', r = reserved.
Shaded locations are unimplemented, read as '0'.

Note 1: These registers can be addressed from any bank.
2: Unimplemented, read as '1'.
3: PIC16F1782/3 only.

FIGURE 6-1: SIMPLIFIED PIC® MCU CLOCK SOURCE BLOCK DIAGRAM



SCS	FOSC<2:0>	PLLEN or SPLLEN	PRIMUX	PSMCMUX	PLLMUX
=00	=100	0	1	1	10
		1	1	1	01
	≠100	0	0	1	10
		1(1)	0	0	00
≠00	XXX	X	X	1	XX

Note 1: This selection should not be made when the PSMC is using the 64 MHz clock option.

6.2.1.4 4x PLL

The oscillator module contains a 4x PLL that can be used with both external and internal clock sources to provide a system clock source. The input frequency for the 4x PLL must fall within specifications. See the PLL Clock Timing Specifications in **Section 30.0 "Electrical Specifications"**.

The 4x PLL may be enabled for use by one of two methods:

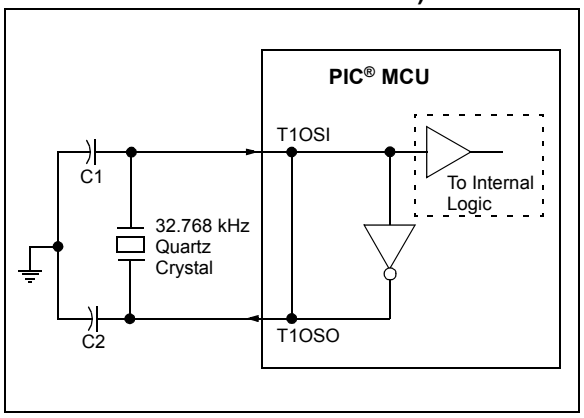
1. Program the PLLEN bit in Configuration Words to a '1'.
2. Write the SPLLEN bit in the OSCCON register to a '1'. If the PLLEN bit in Configuration Words is programmed to a '1', then the value of SPLLEN is ignored.

6.2.1.5 TIMER1 Oscillator

The Timer1 oscillator is a separate crystal oscillator that is associated with the Timer1 peripheral. It is optimized for timekeeping operations with a 32.768 kHz crystal connected between the T1OSO and T1OSI device pins.

The Timer1 oscillator can be used as an alternate system clock source and can be selected during run-time using clock switching. Refer to **Section 6.3 "Clock Switching"** for more information.

FIGURE 6-5: QUARTZ CRYSTAL OPERATION (TIMER1 OSCILLATOR)



Note 1: Quartz crystal characteristics vary according to type, package and manufacturer. The user should consult the manufacturer data sheets for specifications and recommended application.

2: Always verify oscillator performance over the VDD and temperature range that is expected for the application.

3: For oscillator design assistance, reference the following Microchip Applications Notes:

- AN826, "*Crystal Oscillator Basics and Crystal Selection for rfPIC® and PIC® Devices*" (DS00826)
- AN849, "*Basic PIC® Oscillator Design*" (DS00849)
- AN943, "*Practical PIC® Oscillator Analysis and Design*" (DS00943)
- AN949, "*Making Your Oscillator Work*" (DS00949)
- TB097, "*Interfacing a Micro Crystal MS1V-T1K 32.768 kHz Tuning Fork Crystal to a PIC16F690/SS*" (DS91097)
- AN1288, "*Design Practices for Low-Power External Oscillators*" (DS01288)

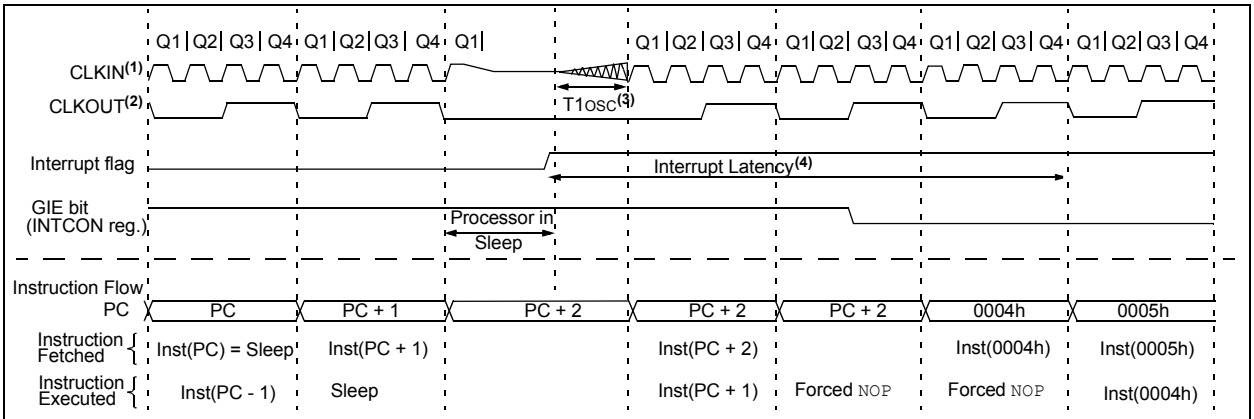
9.1.1 WAKE-UP USING INTERRUPTS

When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of a SLEEP instruction
  - SLEEP instruction will execute as a NOP.
  - WDT and WDT prescaler will not be cleared
  - $\overline{TO}$ bit of the STATUS register will not be set
  - $\overline{PD}$ bit of the STATUS register will not be cleared.
- If the interrupt occurs **during or after** the execution of a SLEEP instruction
  - SLEEP instruction will be completely executed
  - Device will immediately wake-up from Sleep
  - WDT and WDT prescaler will be cleared
  - $\overline{TO}$ bit of the STATUS register will be set
  - $\overline{PD}$ bit of the STATUS register will be cleared.

Even if the flag bits were checked before executing a SLEEP instruction, it may be possible for flag bits to become set before the SLEEP instruction completes. To determine whether a SLEEP instruction executed, test the $\overline{PD}$ bit. If the $\overline{PD}$ bit is set, the SLEEP instruction was executed as a NOP.

FIGURE 9-1: WAKE-UP FROM SLEEP THROUGH INTERRUPT



Note 1: External clock. High, Medium, Low mode assumed.
2: CLKOUT is shown here for timing reference.
3: T1OSC; See **Section 30.0 "Electrical Specifications"**.
4: GIE = 1 assumed. In this case after wake-up, the processor calls the ISR at 0004h. If GIE = 0, execution will continue in-line.

9.3 Register Definitions: Voltage Regulator Control

REGISTER 9-1: VREGCON: VOLTAGE REGULATOR CONTROL REGISTER(1)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-1/1
—	—	—	—	—	—	VREGPM	Reserved
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-2 **Unimplemented:** Read as '0'

bit 1 **VREGPM:** Voltage Regulator Power Mode Selection bit
- 1 = Low-Power Sleep mode enabled in Sleep(2)
  Draws lowest current in Sleep, slower wake-up
- 0 = Normal-Power mode enabled in Sleep(2)
  Draws higher current in Sleep, faster wake-up

bit 0 **Reserved:** Read as '1'. Maintain this bit set.

Note 1: "F" devices only.
2: See **Section 30.0 "Electrical Specifications"**.

TABLE 9-1: SUMMARY OF REGISTERS ASSOCIATED WITH POWER-DOWN MODE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	RAIF	79
IOCBF	IOCBF7	IOCBF6	IOCBF5	IOCBF4	IOCBF3	IOCBF2	IOCBF1	IOCBF0	134
IOCBN	IOCBN7	IOCBN6	IOCBN5	IOCBN4	IOCBN3	IOCBN2	IOCBN1	IOCBN0	133
IOCBP	IOCBP7	IOCBP6	IOCBP5	IOCBP4	IOCBP3	IOCBP2	IOCBP1	IOCBP0	133
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	80
PIE2	OSEIE	C2IE	C1IE	EEIE	BCL1IE	—	C3IE	CCP2IE	81
—	Unimplemented								—
PIE4	—	—	PSMC2TIE	PSMC1TIE	—	—	PSMC2SIE	PSMC1SIE	82
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	80
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	—	C3IF	CCP2IF	84
—	Unimplemented								—
PIR4	—	—	PSMC2TIF	PSMC1TIF	—	—	PSMC2SIF	PSMC1SIF	85
STATUS	—	—	—	$\overline{TO}$	$\overline{PD}$	Z	DC	C	18
VREGCON	—	—	—	—	—	—	VREGPM	Reserved	90
WDTCON	—	—	WDTPS<4:0>					SWDTEN	94

Legend: — = unimplemented location, read as '0'. Shaded cells are not used in Power-Down mode.

REGISTER 17-4: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 0

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
AD<11:4>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **AD<11:4>**: ADC Result Register bits
Upper 8 bits of 12-bit conversion result

REGISTER 17-5: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 0

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
AD<3:0>				—	—	—	ADSIGN
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-4 **AD<3:0>**: ADC Result Register bits
Lower 4 bits of 12-bit conversion result

bit 3-1 **Extended LSb bits**: These are cleared to zero by DC conversion.

bit 0 **ADSIGN**: ADC Result Sign bit

TABLE 19-1: SUMMARY OF REGISTERS ASSOCIATED WITH THE DAC MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>		137
DACCON0	DACEN	—	DACOE1	DACOE2	DACPSS<1:0>		—	DACNSS	161
DACCON1	DACR<7:0>								161

Legend: — = Unimplemented location, read as '0'. Shaded cells are not used with the DAC module.

21.2 Register Definitions: Option Register

REGISTER 21-1: OPTION_REG: OPTION REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
$\overline{\text{WPUEN}}$	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>		
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **$\overline{\text{WPUEN}}$:** Weak Pull-Up Enable bit
1 = All weak pull-ups are disabled (except $\overline{\text{MCLR}}$, if it is enabled)
0 = Weak pull-ups are enabled by individual WPUx latch values

bit 6 **INTEDG:** Interrupt Edge Select bit
1 = Interrupt on rising edge of INT pin
0 = Interrupt on falling edge of INT pin

bit 5 **TMR0CS:** Timer0 Clock Source Select bit
1 = Transition on T0CKI pin
0 = Internal instruction cycle clock (FOSC/4)

bit 4 **TMR0SE:** Timer0 Source Edge Select bit
1 = Increment on high-to-low transition on T0CKI pin
0 = Increment on low-to-high transition on T0CKI pin

bit 3 **PSA:** Prescaler Assignment bit
1 = Prescaler is not assigned to the Timer0 module
0 = Prescaler is assigned to the Timer0 module

bit 2-0 **PS<2:0>:** Prescaler Rate Select bits

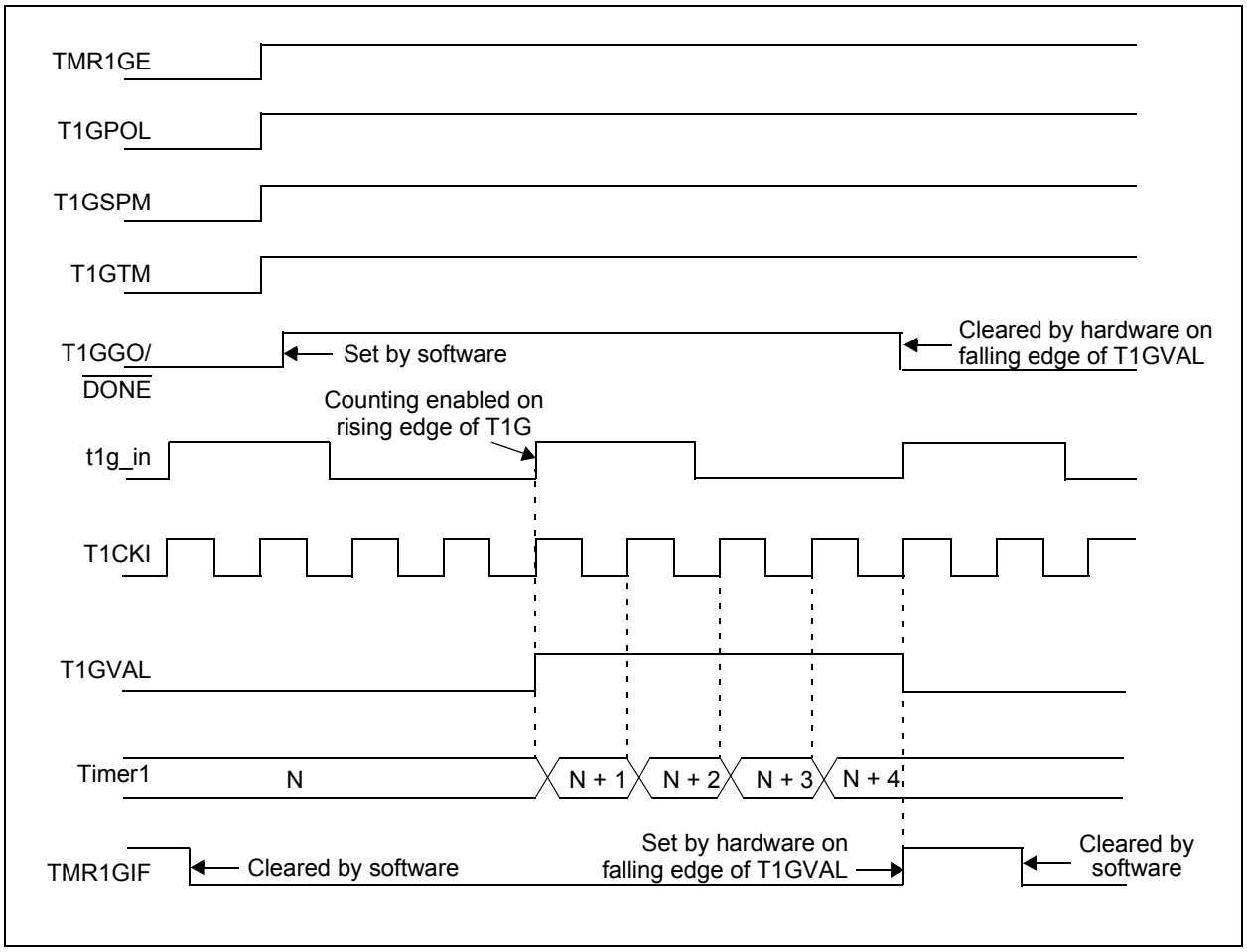
Bit Value	Timer0 Rate
000	1 : 2
001	1 : 4
010	1 : 8
011	1 : 16
100	1 : 32
101	1 : 64
110	1 : 128
111	1 : 256

TABLE 21-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER0

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	79
OPTION_REG	$\overline{\text{WPUEN}}$	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			174
TMR0	Timer0 Module Register								172*
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	114

Legend: — = Unimplemented location, read as '0'. Shaded cells are not used by the Timer0 module.
* Page provides register information.

FIGURE 22-6: TIMER1 GATE SINGLE-PULSE AND TOGGLE COMBINED MODE



23.1 Timer2 Operation

The clock input to the Timer2 modules is the system instruction clock (FOSC/4).

A 4-bit counter/prescaler on the clock input allows direct input, divide-by-4 and divide-by-16 prescale options. These options are selected by the prescaler control bits, T2CKPS<1:0> of the T2CON register. The value of TMR2 is compared to that of the Period register, PR2, on each clock cycle. When the two values match, the comparator generates a match signal as the timer output. This signal also resets the value of TMR2 to 00h on the next cycle and drives the output counter/postscaler (see **Section 23.2 "Timer2 Interrupt"**).

The TMR2 and PR2 registers are both directly readable and writable. The TMR2 register is cleared on any device Reset, whereas the PR2 register initializes to FFh. Both the prescaler and postscaler counters are cleared on the following events:

- a write to the TMR2 register
- a write to the T2CON register
- Power-on Reset (POR)
- Brown-out Reset (BOR)
- $\overline{\text{MCLR}}$ Reset
- Watchdog Timer (WDT) Reset
- Stack Overflow Reset
- Stack Underflow Reset
- RESET Instruction

Note:	TMR2 is not cleared when T2CON is written.

23.2 Timer2 Interrupt

Timer2 can also generate an optional device interrupt. The Timer2 output signal (TMR2-to-PR2 match) provides the input for the 4-bit counter/postscaler. This counter generates the TMR2 match interrupt flag which is latched in TMR2IF of the PIR1 register. The interrupt is enabled by setting the TMR2 Match Interrupt Enable bit, TMR2IE, of the PIE1 register.

A range of 16 postscale options (from 1:1 through 1:16 inclusive) can be selected with the postscaler control bits, T2OUTPS<3:0>, of the T2CON register.

23.3 Timer2 Output

The unscaled output of TMR2 is available primarily to the CCP modules, where it is used as a time base for operations in PWM mode.

Timer2 can be optionally used as the shift clock source for the MSSP module operating in SPI mode. Additional information is provided in **Section 26.0 "Master Synchronous Serial Port (MSSP) Module"**

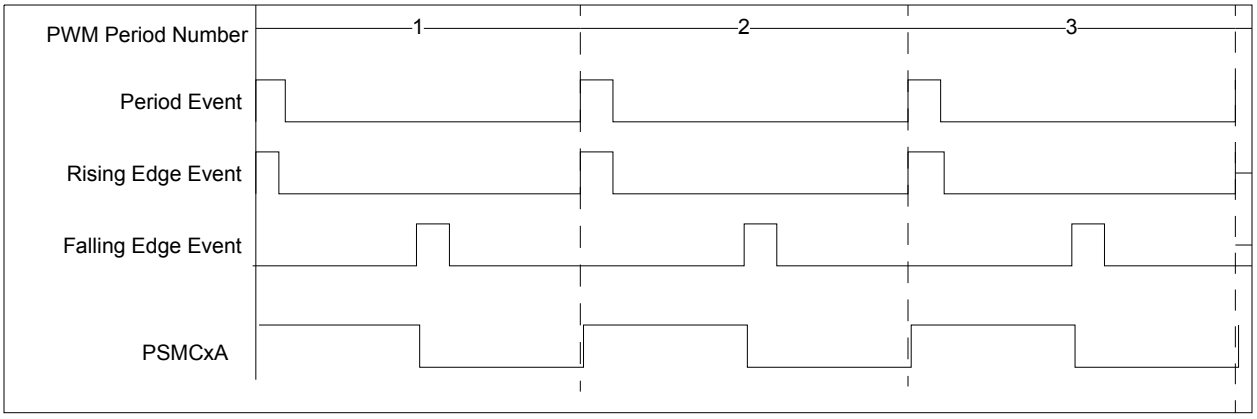
23.4 Timer2 Operation During Sleep

The Timer2 timers cannot be operated while the processor is in Sleep mode. The contents of the TMR2 and PR2 registers will remain unchanged while the processor is in Sleep mode.

EXAMPLE 24-1: SINGLE-PHASE SETUP

```
; Single-phase PWM PSMC setup
; Fully synchronous operation
; Period = 10 us
; Duty cycle = 50%
    BANKSEL PSMC1CON
    MOVLW   0x02        ; set period
    MOVWF   PSMC1PRH
    MOVLW   0x7F
    MOVWF   PSMC1PRL
    MOVLW   0x01        ; set duty cycle
    MOVWF   PSMC1DCH
    MOVLW   0x3F
    MOVWF   PSMC1DCL
    CLRF    PSMC1PHH    ; no phase offset
    CLRF    PSMC1PHL
    MOVLW   0x01        ; PSMC clock=64 MHz
    MOVWF   PSMC1CLK
; output on A, normal polarity
    BSF     PSMC1STR0,P1STRA
    BCF     PSMC1POL, P1POLA
    BSF     PSMC1OEN, P1OEA
; set time base as source for all events
    BSF     PSMC1PRS, P1PRST
    BSF     PSMC1PHS, P1PHST
    BSF     PSMC1DCS, P1DCST
; enable PSMC in Single-Phase Mode
; this also loads steering and time buffers
    MOVLW   B'11000000'
    MOVWF   PSMC1CON
    BANKSEL TRISC
    BCF     TRISC, 0    ; enable pin driver
```

FIGURE 24-4: SINGLE PWM WAVEFORM – PSMCXSTR0 = 01H



24.3.7 PULSE-SKIPPING PWM

The pulse-skipping PWM is used to generate a series of fixed-length pulses that can be triggered at each period event. A rising edge event will be generated when any enabled asynchronous rising edge input is active when the period event occurs, otherwise no event will be generated.

The rising edge event occurs based upon the value in the PSMCxPH register pair.

The falling edge event always occurs according to the enabled event inputs without qualification between any two inputs.

24.3.7.1 Mode Features

- No dead-band control available
- No steering control available
- PWM is output to only one pin:
  - PSMCxA

24.3.7.2 Waveform Generation

Rising Edge Event

If any enabled asynchronous rising edge event = 1 when there is a period event, then upon the next synchronous rising edge event:

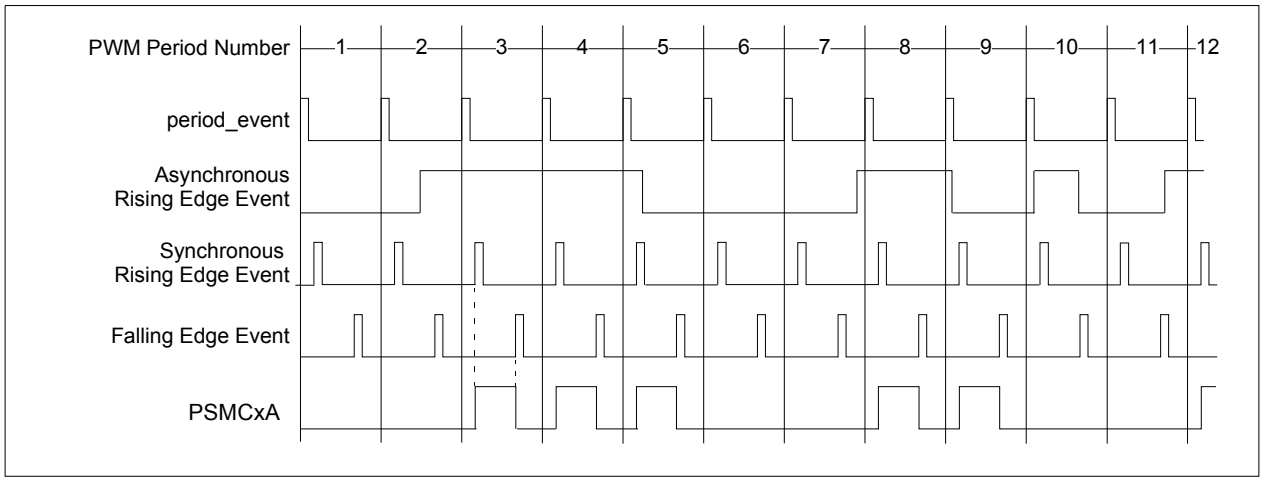
- PSMCxA is set active

Falling Edge Event

- PSMCxA is set inactive

Note:	To use this mode, an external source must be used for the determination of whether or not to generate the set pulse. If the phase time base is used, it will either always generate a pulse or never generate a pulse based on the PSMCxPH value.

FIGURE 24-10: PULSE-SKIPPING PWM WAVEFORM



REGISTER 24-15: PSMCxASDL: PSMC AUTO-SHUTDOWN OUTPUT LEVEL REGISTER

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	PxASDLF(1)	PxASDLE(1)	PxASDLD(1)	PxASDLC(1)	PxASDLB	PxASDLA
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-6 **Unimplemented:** Read as '0'

bit 5 **PxASDLF:** PSMCx Output F Auto-Shutdown Pin Level bit(1)
- 1 = When auto-shutdown is asserted, pin PSMCxF will drive logic '1'
- 0 = When auto-shutdown is asserted, pin PSMCxF will drive logic '0'

bit 4 **PxASDLE:** PSMCx Output E Auto-Shutdown Pin Level bit(1)
- 1 = When auto-shutdown is asserted, pin PSMCxE will drive logic '1'
- 0 = When auto-shutdown is asserted, pin PSMCxE will drive logic '0'

bit 3 **PxASDLD:** PSMCx Output D Auto-Shutdown Pin Level bit(1)
- 1 = When auto-shutdown is asserted, pin PSMCxD will drive logic '1'
- 0 = When auto-shutdown is asserted, pin PSMCxD will drive logic '0'

bit 2 **PxASDLC:** PSMCx Output C Auto-Shutdown Pin Level bit(1)
- 1 = When auto-shutdown is asserted, pin PSMCxC will drive logic '1'
- 0 = When auto-shutdown is asserted, pin PSMCxC will drive logic '0'

bit 1 **PxASDLB:** PSMCx Output B Auto-Shutdown Pin Level bit
- 1 = When auto-shutdown is asserted, pin PSMCxB will drive logic '1'
- 0 = When auto-shutdown is asserted, pin PSMCxB will drive logic '0'

bit 0 **PxASDLA:** PSMCx Output A Auto-Shutdown Pin Level bit
- 1 = When auto-shutdown is asserted, pin PSMCxA will drive logic '1'
- 0 = When auto-shutdown is asserted, pin PSMCxA will drive logic '0'

Note 1: These bits are not implemented on PSMC2.

27.5.2 SYNCHRONOUS SLAVE MODE

The following bits are used to configure the EUSART for synchronous slave operation:

- SYNC = 1
- CSRC = 0
- SREN = 0 (for transmit); SREN = 1 (for receive)
- CREN = 0 (for transmit); CREN = 1 (for receive)
- SPEN = 1

Setting the SYNC bit of the TXSTA register configures the device for synchronous operation. Clearing the CSRC bit of the TXSTA register configures the device as a slave. Clearing the SREN and CREN bits of the RCSTA register ensures that the device is in the Transmit mode, otherwise the device will be configured to receive. Setting the SPEN bit of the RCSTA register enables the EUSART.

27.5.2.1 EUSART Synchronous Slave Transmit

The operation of the Synchronous Master and Slave modes are identical (see **Section 27.5.1.3 "Synchronous Master Transmission"**), except in the case of the Sleep mode.

If two words are written to the TXREG and then the SLEEP instruction is executed, the following will occur:

1. The first character will immediately transfer to the TSR register and transmit.
2. The second word will remain in TXREG register.
3. The TXIF bit will not be set.
4. After the first character has been shifted out of TSR, the TXREG register will transfer the second character to the TSR and the TXIF bit will now be set.
5. If the PEIE and TXIE bits are set, the interrupt will wake the device from Sleep and execute the next instruction. If the GIE bit is also set, the program will call the Interrupt Service Routine.

27.5.2.2 Synchronous Slave Transmission Set-up:

1. Set the SYNC and SPEN bits and clear the CSRC bit.
2. Clear the ANSEL bit for the CK pin (if applicable).
3. Clear the CREN and SREN bits.
4. If interrupts are desired, set the TXIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
5. If 9-bit transmission is desired, set the TX9 bit.
6. Enable transmission by setting the TXEN bit.
7. If 9-bit transmission is selected, insert the Most Significant bit into the TX9D bit.
8. Start transmission by writing the Least Significant 8 bits to the TXREG register.

TABLE 27-9: SUMMARY OF REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE TRANSMISSION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON	C2OUTSEL	CC1PSEL	SDOSEL	SCKSEL	SDISEL	TXSEL	RXSEL	CCP2SEL	111
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	322
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	79
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	80
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	83
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	321
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	125
TXREG	EUSART Transmit Data Register								312*
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	320

Legend: — = unimplemented location, read as '0'. Shaded cells are not used for synchronous slave transmission.

* Page provides register information.

TABLE 29-3: INSTRUCTION SET

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb			LSb	Status Affected	Notes
BYTE-ORIENTED FILE REGISTER OPERATIONS									
ADDWF	f, d	Add W and f	1	00	0111	dfff	ffff	C, DC, Z	2
ADDWFC	f, d	Add with Carry W and f	1	11	1101	dfff	ffff	C, DC, Z	2
ANDWF	f, d	AND W with f	1	00	0101	dfff	ffff	Z	2
ASRF	f, d	Arithmetic Right Shift	1	11	0111	dfff	ffff	C, Z	2
LSLF	f, d	Logical Left Shift	1	11	0101	dfff	ffff	C, Z	2
LSRF	f, d	Logical Right Shift	1	11	0110	dfff	ffff	C, Z	2
CLRF	f	Clear f	1	00	0001	lfff	ffff	Z	2
CLRW	–	Clear W	1	00	0001	0000	00xx	Z	
COMF	f, d	Complement f	1	00	1001	dfff	ffff	Z	2
DECF	f, d	Decrement f	1	00	0011	dfff	ffff	Z	2
INCF	f, d	Increment f	1	00	1010	dfff	ffff	Z	2
IORWF	f, d	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	2
MOVF	f, d	Move f	1	00	1000	dfff	ffff	Z	2
MOVWF	f	Move W to f	1	00	0000	1fff	ffff		2
RLF	f, d	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	2
RRF	f, d	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	2
SUBWF	f, d	Subtract W from f	1	00	0010	dfff	ffff	C, DC, Z	2
SUBWFB	f, d	Subtract with Borrow W from f	1	11	1011	dfff	ffff	C, DC, Z	2
SWAPF	f, d	Swap nibbles in f	1	00	1110	dfff	ffff		2
XORWF	f, d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	2
BYTE ORIENTED SKIP OPERATIONS									
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1, 2
INCFSZ	f, d	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1, 2
BIT-ORIENTED FILE REGISTER OPERATIONS									
BCF	f, b	Bit Clear f	1	01	00bb	bfff	ffff		2
BSF	f, b	Bit Set f	1	01	01bb	bfff	ffff		2
BIT-ORIENTED SKIP OPERATIONS									
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01	10bb	bfff	ffff		1, 2
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01	11bb	bfff	ffff		1, 2
LITERAL OPERATIONS									
ADDLW	k	Add literal and W	1	11	1110	kkkk	kkkk	C, DC, Z	
ANDLW	k	AND literal with W	1	11	1001	kkkk	kkkk	Z	
IORLW	k	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
MOVLB	k	Move literal to BSR	1	00	0000	001k	kkkk		
MOVLP	k	Move literal to PCLATH	1	11	0001	1kkk	kkkk		
MOVLW	k	Move literal to W	1	11	0000	kkkk	kkkk		
SUBLW	k	Subtract W from literal	1	11	1100	kkkk	kkkk	C, DC, Z	
XORLW	k	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Z	

Note 1: If the Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

2: If this instruction addresses an INDF register and the MSb of the corresponding FSR is set, this instruction will require one additional instruction cycle.

TABLE 30-2: SUPPLY VOLTAGE (I_{DD})(1,2)

PIC16LF1782/3			Standard Operating Conditions (unless otherwise stated)				
PIC16F1782/3							
Param No.	Device Characteristics	Min.	Typ†	Max.	Units	Conditions	
						V_{DD}	Note
D009	LDO Regulator	—	75	—	µA	—	High-Power mode, normal operation
		—	15	—	µA	—	Sleep VREGCON<1> = 0
		—	0.3	—	µA	—	Sleep VREGCON<1> = 1
D010		—	8	20	µA	1.8	F_{OSC} = 32 kHz LP Oscillator mode **(Note 4)**, -40°C ≤ T_A ≤ +85°C
		—	12	24	µA	3.0	
D010		—	18	63	µA	2.3	F_{OSC} = 32 kHz LP Oscillator mode **(Note 4, 5)**, -40°C ≤ T_A ≤ +85°C
		—	20	74	µA	3.0	
		—	22	79	µA	5.0	
D012		—	160	650	µA	1.8	F_{OSC} = 4 MHz XT Oscillator mode
		—	320	1000	µA	3.0	
D012		—	260	700	µA	2.3	F_{OSC} = 4 MHz XT Oscillator mode **(Note 5)**
		—	330	1100	µA	3.0	
		—	380	1300	µA	5.0	

Note 1: The test conditions for all I_{DD} measurements in active operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to V_{DD}; $\overline{MCLR}$ = V_{DD}; WDT disabled.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption.

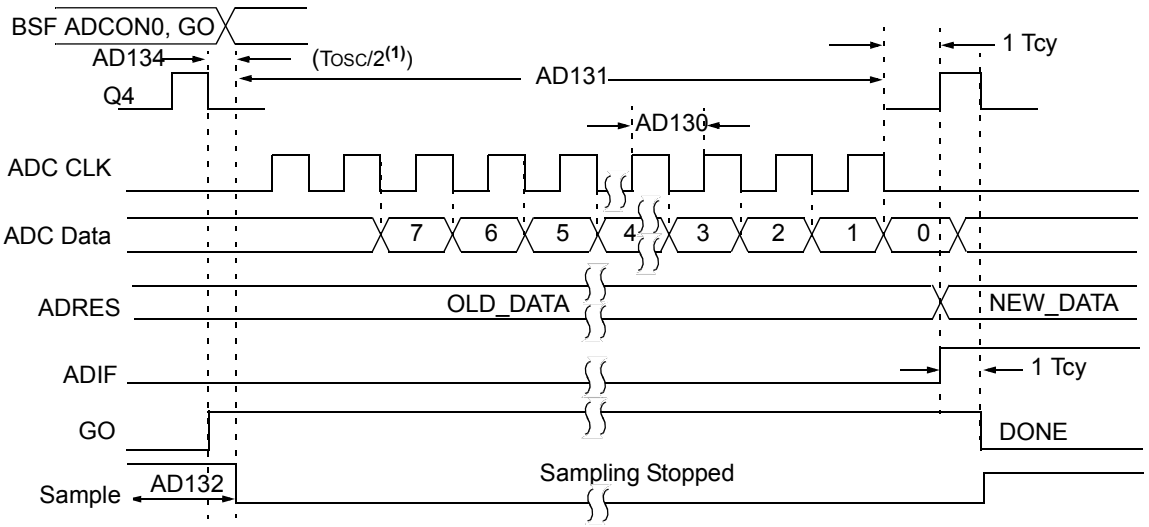
3: For RC oscillator configurations, current through R_{EXT} is not included. The current through the resistor can be extended by the formula $I_R = V_{DD}/2R_{EXT}$ (mA) with R_{EXT} in kΩ.

4: FVR and BOR are disabled.

5: 0.1 µF capacitor on V_{CAP}.

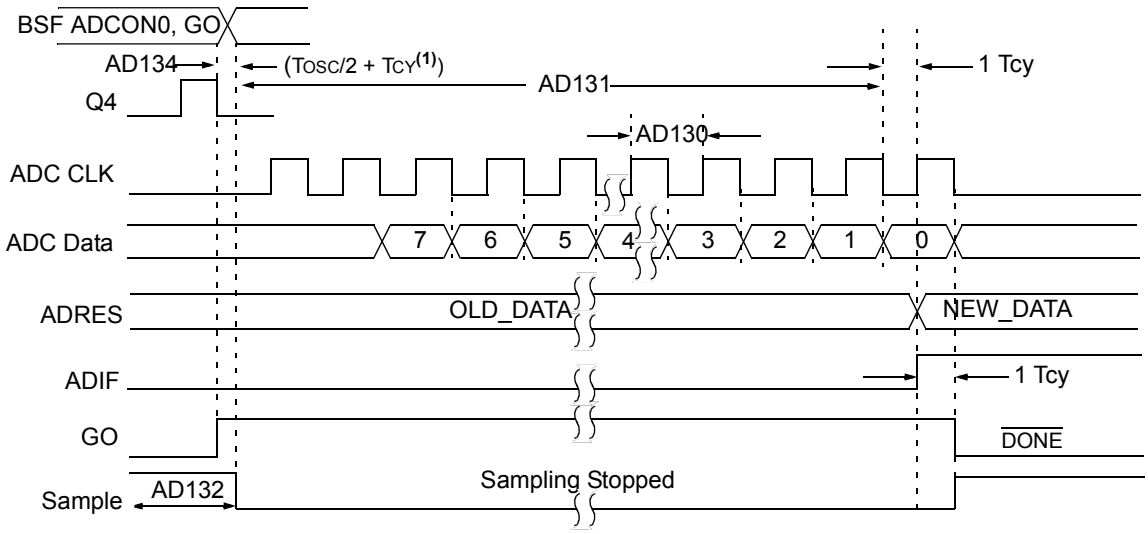
6: 8 MHz crystal oscillator with 4x PLL enabled.

FIGURE 30-12: ADC CONVERSION TIMING (NORMAL MODE)



Note 1: If the ADC clock source is selected as RC, a time of TCY is added before the ADC clock starts. This allows the SLEEP instruction to be executed.

FIGURE 30-13: ADC CONVERSION TIMING (SLEEP MODE)



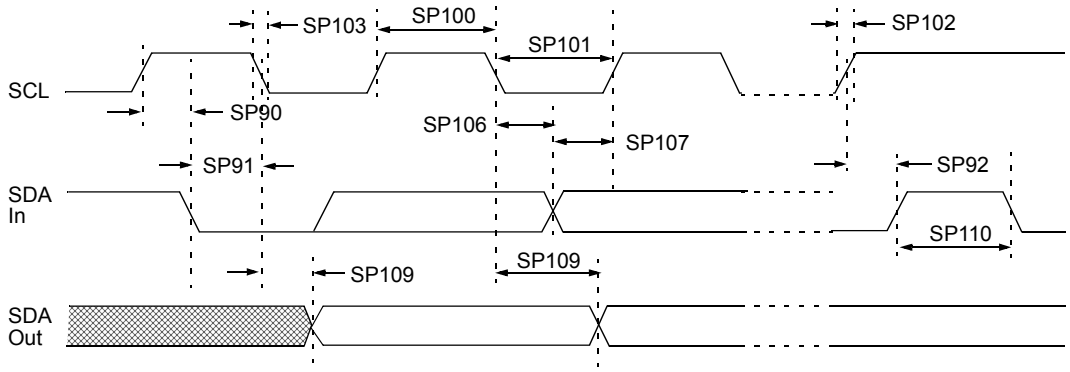
Note 1: If the ADC clock source is selected as RC, a time of TCY is added before the ADC clock starts. This allows the SLEEP instruction to be executed.

TABLE 30-21: I²C™ BUS START/STOP BITS REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)								
Param No.	**Symbol**	**Characteristic**		**Min.**	**Typ**	**Max.**	**Units**	**Conditions**
SP90*	TSU:STA	Start condition Setup time	100 kHz mode	4700	—	—	ns	Only relevant for Repeated Start condition
			400 kHz mode	600	—	—		
SP91*	THD:STA	Start condition Hold time	100 kHz mode	4000	—	—	ns	After this period, the first clock pulse is generated
			400 kHz mode	600	—	—		
SP92*	TSU:STO	Stop condition Setup time	100 kHz mode	4700	—	—	ns	
			400 kHz mode	600	—	—		
SP93	THD:STO	Stop condition Hold time	100 kHz mode	4000	—	—	ns	
			400 kHz mode	600	—	—		

* These parameters are characterized but not tested.

FIGURE 30-21: I²C™ BUS DATA TIMING



Note: Refer to Figure 30-4 for load conditions.

32.11 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

32.12 Third-Party Development Tools

Microchip also offers a great collection of tools from third-party vendors. These tools are carefully selected to offer good value and unique functionality.

- Device Programmers and Gang Programmers from companies, such as SoftLog and CCS
- Software Tools from companies, such as Gimpel and Trace Systems
- Protocol Analyzers from companies, such as Saleae and Total Phase
- Demonstration Boards from companies, such as MikroElektronika, Digilent® and Olimex
- Embedded Ethernet Solutions from companies, such as EZ Web Lynx, WIZnet and IPLogika®

NOTES: