What is "Embedded - Microcontrollers"?

Welcome to E-XFL.COM





Applications of "<u>Embedded -</u> <u>Microcontrollers</u>"

#### Details

Product Status	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	10MHz
Connectivity	I <sup>2</sup> C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	36
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	·
RAM Size	368 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 14x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8×8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf747t-i-ml

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

### 4.6 Clock Sources and Oscillator Switching

The PIC16F7X7 devices include a feature that allows the system clock source to be switched from the main oscillator to an alternate low-frequency clock source. PIC16F7X7 devices offer three alternate clock sources. When enabled, these give additional options for switching to the various power-managed operating modes.

Essentially, there are three clock sources for these devices:

- Primary oscillators
- Secondary oscillators
- Internal oscillator block (INTRC)

The **primary oscillators** include the External Crystal and Resonator modes, the External RC modes, the External Clock mode and the internal oscillator block. The particular mode is defined on POR by the contents of Configuration Word 1. The details of these modes are covered earlier in this chapter.

The **secondary oscillators** are those external sources not connected to the OSC1 or OSC2 pins. These sources may continue to operate even after the controller is placed in a power-managed mode.

PIC16F7X7 devices offer the Timer1 oscillator as a secondary oscillator. This oscillator continues to run when a SLEEP instruction is executed and is often the time base for functions, such as a real-time clock.

Most often, a 32.768 kHz watch crystal is connected between the RC0/T1OSO/T1CKI and RC1/T1OSI/CCP2 pins. Like the LP mode oscillator circuit, loading capacitors are also connected from each pin to ground. The Timer1 oscillator is discussed in greater detail in Section 7.6 "Timer1 Oscillator".

In addition to being a primary clock source, the **internal oscillator block** is available as a power-managed mode clock source. The 31.25 kHz INTRC source is also used as the clock source for several special features, such as the WDT, Fail-Safe Clock Monitor, Power-up Timer and Two-Speed Start-up.

The clock sources for the PIC16F7X7 devices are shown in Figure 4-6. See **Section 7.0** "**Timer1 Module**" for further details of the Timer1 oscillator. See **Section 15.1** "**Configuration Bits**" for Configuration register details.

# 4.6.1 OSCCON REGISTER

The OSCCON register (Register 4-2) controls several aspects of the system clock's operation, both in full power operation and in power-managed modes.

The system clock select bits, SCS1:SCS0, select the clock source that is used when the device is operating in power-managed modes. When the bits are cleared (SCS<1:0> = 0.0), the system clock source comes from

the main oscillator that is selected by the FOSC2:FOSC0 configuration bits in Configuration Register 1. When the bits are set in any other manner, the system clock source is provided by the Timer1 oscillator (SCS1:SCS0 = 01) or from the internal oscillator block (SCS1:SCS0 = 10). After a Reset, SCS<1:0> are always set to '00'.

The internal oscillator select bits, IRCF2:IRCF0, select the frequency output of the internal oscillator block that is used to drive the system clock. The choices are the INTRC source (31.25 kHz), the INTOSC source (8 MHz) or one of the six frequencies derived from the INTOSC postscaler (125 kHz to 4 MHz). Changing the configuration of these bits has an immediate change on the multiplexor's frequency output.

The OSTS and IOFS bits indicate the status of the primary oscillator and INTOSC source; these bits are set when their respective oscillators are stable. In particular, OSTS indicates that the Oscillator Start-up Timer has timed out.

### 4.6.2 CLOCK SWITCHING

Clock switching will occur for the following reasons:

- The FCMEN (CONFIG2<0>) bit is set, the device is running from the primary oscillator and the primary oscillator fails. The clock source will be the internal RC oscillator.
- The FCMEN bit is set, the device is running from the Timer1 oscillator (T1OSC) and T1OSC fails. The clock source will be the internal RC oscillator.
- Following a wake-up due to a Reset or a POR, when the device is configured for Two-Speed Start-up mode, switching will occur between the INTRC and the system clock defined by the FOSC<2:0> bits.
- A wake-up from Sleep occurs due to interrupt or WDT wake-up and Two-Speed Start-up is enabled. If the primary clock is XT, HS or LP, the clock will switch between the INTRC and the primary system clock after 1024 clocks and 8 clocks of the primary oscillator. This is conditional upon the SCS bits being set equal to '00'.
- SCS bits are modified from their original value.
- IRCF bits are modified from their original value.

Note: Because the SCS bits are cleared on any Reset, no clock switching will occur on a Reset unless the Two-Speed Start-up is enabled and the primary clock is XT, HS or LP. The device will wait for the primary clock to become stable before execution begins (Two-Speed Start-up disabled).



#### 7.9 Resetting Timer1 Register Pair (TMR1H, TMR1L)

TMR1H and TMR1L registers are not reset to 00h on a POR, or any other Reset, except by the CCP1 special event triggers.

T1CON register is reset to 00h on a Power-on Reset or a Brown-out Reset, which shuts off the timer and leaves a 1:1 prescale. In all other Resets, the register is unaffected.

### 7.10 Timer1 Prescaler

The prescaler counter is cleared on writes to the TMR1H or TMR1L registers.

### 7.11 Using Timer1 as a Real-Time Clock

Adding an external LP oscillator to Timer1 (such as the one described in **Section 7.6** "**Timer1 Oscillator**") gives users the option to include RTC functionality in their applications. This is accomplished with an inexpensive watch crystal to provide an accurate time base and several lines of application code to calculate the time. When operating in Sleep mode and using a battery or supercapacitor as a power source, it can completely eliminate the need for a separate RTC device and battery backup.

The application code routine, RTCisr, shown in Example 7-3, demonstrates a simple method to increment a counter at one-second intervals using an Interrupt Service Routine. Incrementing the TMR1 register pair to overflow, triggers the interrupt and calls the routine which increments the seconds counter by one; additional counters for minutes and hours are incremented as the previous counter overflows.

Since the register pair is 16 bits wide, counting up to overflow the register directly from a 32.768 kHz clock would take 2 seconds. To force the overflow at the required one-second intervals, it is necessary to preload it. The simplest method is to set the MSb of TMR1H with a BSF instruction. Note that the TMR1L register is never preloaded or altered; doing so may introduce cumulative error over many cycles.

For this method to be accurate, Timer1 must operate in Asynchronous mode and the Timer1 overflow interrupt must be enabled (PIE1<0> = 1) as shown in the routine, RTCinit. The Timer1 oscillator must also be enabled and running at all times.

RTCinit	BANKSEL	TMR1H	
	MOVLW	0x80	; Preload TMR1 register pair
	MOVWF	TMR1H	; for 1 second overflow
	CLRF	TMR1L	
	MOVLW	b'00001111'	; Configure for external clock,
	MOVWF	T1CON	; Asynchronous operation, external oscillator
	CLRF	secs	; Initialize timekeeping registers
	CLRF	mins	
	MOVLW	.12	
	MOVWF	hours	
	BANKSEL	PIE1	
	BSF	PIE1, TMR1IE	; Enable Timer1 interrupt
	RETURN		
RTCisr	BANKSEL	TMR1H	
	BSF	TMR1H, 7	; Preload for 1 sec overflow
	BCF	PIR1, TMR1IF	; Clear interrupt flag
	INCF	secs, F	; Increment seconds
	MOVF	secs, w	
	SUBLW	.60	
	BTFSS	STATUS, Z	; 60 seconds elapsed?
	RETURN		; No, done
	CLRF	seconds	; Clear seconds
	INCF	mins, f	; Increment minutes
	MOVF	mins, w	
	SUBLW	.60	
	BTFSS	STATUS, Z	; 60 seconds elapsed?
	RETURN		; No, done
	CLRF	mins	; Clear minutes
	INCF	hours, f	; Increment hours
	MOVF	hours, w	
	SUBLW	.24	
	BTFSS	STATUS, Z	; 24 hours elapsed?
	RETURN		; No, done
	CLRF	hours	; Clear hours
	RETURN		; Done

EXAMPLE 7-3: IMPLEMENTING A REAL-TIME CLOCK USING A TIMER1 INTERRUPT SERVICE

|--|

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value POR,	e on BOR	Valu all c Res	e on other sets
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IF	INTOIF	RBIF	0000	000x	0000	000u
0Ch	PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000	0000	0000	0000
8Ch	PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000	0000	0000	0000
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register						xxxx	xxxx	uuuu	uuuu		
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register							xxxx	xxxx	uuuu	uuuu	
10h	T1CON		T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR10N	-000	0000	-uuu	uuuu

Legend: x = unknown, u = unchanged, — = unimplemented, read as '0'. Shaded cells are not used by the Timer1 module.

**Note 1:** Bits PSPIE and PSPIF are reserved on the PIC16F737/767 devices; always maintain these bits clear.

REGISTER 9-1:	CCPxCON	: CCPx C	ONTROL F	REGISTER	(ADDRES	S 17h, 1D	h, <b>97h)</b>		
	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
	—	—	CCPxX	CCPxY	CCPxM3	CCPxM2	CCPxM1	CCPxM0	
	bit 7							bit 0	
bit 7-6	Unimplem	ented: Rea	<b>d as</b> '0'						
bit 5-4	CCPxX:CC	PxY: PWM	Least Signi	ficant bits					
	<u>Capture mo</u> Unused.	ode:							
	<u>Compare m</u> Unused.	node:							
	<u>PWM mode:</u> These bits are the two LSbs of the PWM duty cycle. The eight MSbs are found in CCPRxL.								
bit 3-0	CCPxM3:CCPxM0: CCPx Mode Select bits								
	0000 = Capture/Compare/PWM disabled (resets CCPx module)								
	0100 <b>= Ca</b> p	oture mode,	every falling	g edge					
	0101 <b>= Ca</b> p	oture mode,	every rising	, edge					
	0110 = Cap	oture mode,	every 4th ri	sing edge					
	1000 - Cor	nnare mode	every rour	on match ((	CPvIE bit is	s set)			
	1001 = Compare mode, set output on match (CCPxIF bit is set)								
	1010 = Compare mode, generate software interrupt on match (CCPxIF bit is set, CCPx pin is unaffected)								
	1011 = Compare mode, trigger special event (CCPxIF bit is set, CCPx pin is unaffected); CCP1 clears Timer1; CCP2 clears Timer1 and starts an A/D conversion (if A/D module is enabled)								
	11xx = PW	M mode							
	Legend:								
	R = Reada	ble bit	W = V	Vritable bit	U = Unii	mplemented	l bit, read as	·'O'	

'1' = Bit is set

'0' = Bit is cleared

-n = Value at POR

x = Bit is unknown

#### 10.3.6 SLAVE MODE

In Slave mode, the data is transmitted and received as the external clock pulses appear on SCK. When the last bit is latched, the SSPIF interrupt flag bit is set.

While in Slave mode, the external clock is supplied by the external clock source on the SCK pin. This external clock must meet the minimum high and low times, as specified in the electrical specifications.

While in Sleep mode, the slave can transmit/receive data. When a byte is received, the device will wake-up from Sleep.

Before enabling the module in SPI Slave mode, the clock line must match the proper Idle state. The clock line can be observed by reading the SCK pin. The Idle state is determined by the CKP bit (SSPCON1<4>).

#### 10.3.7 SLAVE SELECT SYNCHRONIZATION

The  $\overline{SS}$  pin allows a Synchronous Slave mode. The SPI must be in Slave mode with  $\overline{SS}$  pin control enabled (SSPCON<3:0> = 4h). The pin must not be driven low for the  $\overline{SS}$  pin to function as an input. The data latch

must be high. When the  $\overline{SS}$  pin is low, transmission and reception are enabled and the SDO pin is driven. When the  $\overline{SS}$  pin goes high, the SDO pin is no longer driven, even if in the middle of a transmitted byte and becomes a floating output. External pull-up/pull-down resistors may be desirable, depending on the application.

- Note 1: When the SPI is in Slave mode with  $\overline{SS}$  pin control enabled (SSPCON<3:0> = 0100), the SPI module will reset if the  $\overline{SS}$  pin is set to VDD.
  - 2: If the SPI is used in Slave mode with CKE set, then the SS pin control must be enabled.

When the SPI module resets, the bit counter is forced to '0'. This can be done by either forcing the SS pin to a high level or clearing the SSPEN bit.

To emulate two-wire communication, the SDO pin can be connected to the SDI pin. When the SPI needs to operate as a receiver, the SDO pin can be configured as an input. This disables transmissions from the SDO. The SDI can always be left as an input (SDI function) since it cannot create a bus conflict.

#### FIGURE 10-4: SLAVE SYNCHRONIZATION WAVEFORM



#### 10.4.2 OPERATION

The MSSP module functions are enabled by setting MSSP enable bit, SSPEN (SSPCON<5>).

The SSPCON register allows control of the  $l^2C$  operation. Four mode selection bits (SSPCON<3:0>) allow one of the following  $l^2C$  modes to be selected:

- I<sup>2</sup>C Master mode, clock = Oscillator/4 (SSPADD + 1)
- I<sup>2</sup>C Slave mode (7-bit address)
- I<sup>2</sup>C Slave mode (10-bit address)
- I<sup>2</sup>C Slave mode (7-bit address), with Start and Stop bit interrupts enabled
- I<sup>2</sup>C Slave mode (10-bit address), with Start and Stop bit interrupts enabled
- I<sup>2</sup>C Firmware Controlled Master mode, slave is Idle

Selection of any  $I^2C$  mode, with the SSPEN bit set, forces the SCL and SDA pins to be open-drain, provided these pins are programmed to inputs by setting the appropriate TRISC bits. To ensure proper operation of the module, pull-up resistors must be provided externally to the SCL and SDA pins.

#### 10.4.3 SLAVE MODE

In Slave mode, the SCL and SDA pins must be configured as inputs (TRISC<4:3> set). The MSSP module will override the input state with the output data when required (slave-transmitter).

To ensure proper communication of the  $l^2C$  Slave mode, the TRIS bits (TRISx [SDA, SCL]) corresponding to the  $l^2C$  pins must be set to '1'. If any TRIS bits (TRISx<7:0>) of the port containing the  $l^2C$  pins (PORTx [SDA, SCL]) are changed in software, during  $l^2C$  communication using a Read-Modify-Write instruction (BSF, BCF), then the  $l^2C$  mode may stop functioning properly and  $l^2C$  communication may suspend. Do not change any of the TRISx bits (TRIS bits of the port containing the  $l^2C$  pins) using the instruction BSF or BCF during  $l^2C$  communication. If it is absolutely necessary to change the TRISx bits during communication, the following method can be used:

```
MOVF
TRISC, W
; Example for a 40-pin part such as the PIC16F877A

IORLW
0x18
; Ensures <4:3> bits are '11'

ANDLW
B'1111001'
; Sets <2:1> as output, but will not alter other bits

...
...
...

MOVWF
TRISC
TRISC
```

The  $I^2C$  Slave mode hardware will always generate an interrupt on an address match. Through the mode select bits, the user can also choose to interrupt on Start and Stop bits.

When an address is matched, or the data transfer after an address match is received, the hardware automatically will generate the Acknowledge (ACK) pulse and load the SSPBUF register with the received value currently in the SSPSR register.

Any combination of the following conditions will cause the MSSP module not to give this ACK pulse:

- The Buffer Full bit, BF (SSPSTAT<0>), was set before the transfer was received.
- The overflow bit, SSPOV (SSPCON<6>), was set before the transfer was received.

In this case, the SSPSR register value is not loaded into the SSPBUF, but bit SSPIF (PIR1<3>) is set. The BF bit is cleared by reading the SSPBUF register, while bit SSPOV is cleared through software.

The SCL clock input must have a minimum high and low for proper operation. The high and low times of the  $I^2C$  specification, as well as the requirement of the MSSP module, are shown in timing parameter #100 and parameter #101.

#### 10.4.3.1 Addressing

Once the MSSP module has been enabled, it waits for a Start condition to occur. Following the Start condition, the 8 bits are shifted into the SSPSR register. All incoming bits are sampled with the rising edge of the clock (SCL) line. The value of register SSPSR<7:1> is compared to the value of the SSPADD register. The address is compared on the falling edge of the eighth clock (SCL) pulse. If the addresses match and the BF and SSPOV bits are clear, the following events occur:

- 1. The SSPSR register value is loaded into the SSPBUF register.
- 2. The Buffer Full bit, BF, is set.
- 3. An ACK pulse is generated.
- 4. MSSP Interrupt Flag bit, SSPIF (PIR1<3>), is set (interrupt is generated if enabled) on the falling edge of the ninth SCL pulse.

#### 10.4.8 I<sup>2</sup>C MASTER MODE START CONDITION TIMING

To initiate a Start condition, the user sets the Start Condition Enable bit, SEN (SSPCON2<0>). If the SDA and SCL pins are sampled high, the Baud Rate Generator is reloaded with the contents of SSPADD<6:0> and starts its count. If SCL and SDA are both sampled high when the Baud Rate Generator times out (TBRG), the SDA pin is driven low. The action of the SDA being driven low while SCL is high is the Start condition and causes the S bit (SSPSTAT<3>) to be set. Following this, the Baud Rate Generator is reloaded with the contents of SSPADD<6:0> and resumes its count. When the Baud Rate Generator times out (TBRG), the SEN bit (SSPCON2<0>) will be automatically cleared by hardware, the Baud Rate Generator is suspended, leaving the SDA line held low and the Start condition is complete.

**Note:** If at the beginning of the Start condition, the SDA and SCL pins are already sampled low, or if during the Start condition, the SCL line is sampled low before the SDA line is driven low, a bus collision occurs, the Bus Collision Interrupt Flag, BCLIF, is set, the Start condition is aborted and the I<sup>2</sup>C module is reset into its Idle state.

# FIGURE 10-19: FIRST START BIT TIMING



If the user writes the SSPBUF when a Start sequence is in progress, the WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

Note: Because queueing of events is not allowed, writing to the lower 5 bits of SSPCON2 is disabled until the Start condition is complete.



#### 11.2.2 AUSART ASYNCHRONOUS RECEIVER

The receiver block diagram is shown in Figure 11-4. The data is received on the RC7/RX/DT pin and drives the data recovery block. The data recovery block is actually a high-speed shifter, operating at x16 times the baud rate; whereas, the main receive serial shifter operates at the bit rate or at Fosc.

Once Asynchronous mode is selected, reception is enabled by setting bit, CREN (RCSTA<4>).

The heart of the receiver is the Receive (Serial) Shift Register (RSR). After sampling the Stop bit, the received data in the RSR is transferred to the RCREG register (if it is empty). If the transfer is complete, flag bit, RCIF (PIR1<5>), is set. The actual interrupt can be enabled/disabled by setting/clearing enable bit, RCIE (PIE1<5>). Flag bit RCIF is a read-only bit which is cleared by the hardware. It is cleared when the RCREG register has been read and is empty. The RCREG is a double-buffered register (i.e., it is a two-deep FIFO). It is possible for two bytes of data to be received and transferred to the RCREG FIFO and a third byte to begin shifting to the RSR register. On the detection of the Stop bit of the third byte, if the RCREG register is still full, the Overrun Error bit, OERR (RCSTA<1>), will be set. The word in the RSR will be lost. The RCREG register can be read twice to retrieve the two bytes in the FIFO. Overrun bit, OERR, has to be cleared in software. This is done by resetting the receive logic (CREN is cleared and then set). If bit OERR is set, transfers from the RSR register to the RCREG register are inhibited and no further data will be received. It is, therefore, essential to clear error bit OERR if it is set. Framing Error bit, FERR (RCSTA<2>), is set if a Stop bit is detected as clear. Bit FERR and the 9th receive bit are buffered the same way as the receive data. Reading the RCREG will load bits RX9D and FERR with new values; therefore, it is essential for the user to read the RCSTA register before reading the RCREG register in order not to lose the old FERR and RX9D information.











#### FIGURE 11-11: SYNCHRONOUS RECEPTION (MASTER MODE, SREN)

### 15.10.1 REFERENCE VOLTAGE SET POINT

The internal reference voltage of the LVD module may be used by other internal circuitry (the Programmable Brown-out Reset). If these circuits are disabled (lower current consumption), the reference voltage circuit requires a time to become stable before a low-voltage condition can be reliably detected. This time is invariant of system clock speed. This start-up time is specified in electrical specification parameter #36. The low-voltage interrupt flag will not be enabled until a stable reference voltage is reached. Refer to the waveform in Figure 15-6.

#### 15.10.2 CURRENT CONSUMPTION

When the module is enabled, the LVD comparator and voltage divider are enabled and will consume static current. The voltage divider can be tapped from multiple places in the resistor array. Total current consumption, when enabled, is specified in electrical specification parameter #D022B.

# 15.11 Operation During Sleep

When enabled, the LVD circuitry continues to operate during Sleep. If the device voltage crosses the trip point, the LVDIF bit will be set and the device will wakeup from Sleep. Device execution will continue from the interrupt vector address if interrupts have been globally enabled.

### 15.12 Effects of a Reset

A device Reset forces all registers to their Reset state. This forces the LVD module to be turned off.

Note: If the LVD is enabled and the BOR module is not enabled, the band gap will require a start-up time of no more than 50 μs before the band gap reference is stable. Before enabling the LVD interrupt, the user should ensure that the band gap reference voltage is stable by monitoring the IRVST bit in the LVDCON register. The LVD could cause erroneous interrupts before the band gap is stable.

#### 15.13 Time-out Sequence

On power-up, the time-out sequence is as follows: the PWRT delay starts (if enabled) when a POR occurs; then, OST starts counting 1024 oscillator cycles when PWRT ends (LP, XT, HS); when the OST ends, the device comes out of Reset.

If MCLR is kept low long enough, all delays will expire. Bringing MCLR high will begin execution immediately. This is useful for testing purposes or to synchronize more than one PIC16F7X7 device operating in parallel.

Table 15-3 shows the Reset conditions for the Status, PCON and PC registers, while Table 15-4 shows the Reset conditions for all the registers.

### 15.14 Power Control/Status Register (PCON)

The Power Control/Status register, PCON, has two bits to indicate the type of Reset that last occurred.

Bit 0 is Brown-out Reset status bit, BOR. Bit BOR is unknown on a Power-on Reset. It must then be set by the user and checked on subsequent Resets to see if bit BOR cleared, indicating a Brown-out Reset occurred. When the Brown-out Reset is disabled, the state of the BOR bit is unpredictable.

Bit 1 is Power-on Reset Status bit, POR. It is cleared on a Power-on Reset and unaffected otherwise. The user must set this bit following a Power-on Reset.

MOVF	Move f
Syntax:	[label] MOVF f,d
Operands:	$\begin{array}{l} 0\leq f\leq 127\\ d\in [0,1] \end{array}$
Operation:	(f) $\rightarrow$ (destination)
Status Affected:	Z
Description:	The contents of register 'f' are moved to a destination dependant upon the status of 'd'. If $d = 0$ , the destination is W register. If d = 1, the destination is file register 'f' itself. $d = 1$ is useful to test a file register since status flag Z is affected.

NOP	No Operation
Syntax:	[label] NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Description:	No operation.

MOVLW	Move Literal to W					
Syntax:	[ <i>label</i> ] MOVLW k					
Operands:	$0 \le k \le 255$					
Operation:	$k \rightarrow (W)$					
Status Affected:	None					
Description:	The eight-bit literal 'k' is loaded into W register. The don't cares will assemble as '0's.					

RETFIE	Return from Interrupt				
Syntax:	[label] RETFIE				
Operands:	None				
Operation:	$TOS \rightarrow PC, \\ 1 \rightarrow GIE$				
Status Affected:	None				

MOVWF	Move W to f					
Syntax:	[ <i>label</i> ] MOVWF f					
Operands:	$0 \leq f \leq 127$					
Operation:	$(W) \to (f)$					
Status Affected:	None					
Description:	Move data from W register to register 'f'.					

RETLW	Return with Literal in W				
Syntax:	[ <i>label</i> ] RETLW k				
Operands:	$0 \leq k \leq 255$				
Operation:	$k \rightarrow (W);$ TOS $\rightarrow PC$				
Status Affected:	None				
Description:	The W register is loaded with the eight-bit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.				

# 17.7 MPLAB SIM Software Simulator

The MPLAB SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC<sup>®</sup> 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 SIM Software Simulator fully supports symbolic debugging using the MPLAB C 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.

### 17.8 MPLAB REAL ICE In-Circuit Emulator System

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 PIC<sup>®</sup> Flash MCUs and dsPIC<sup>®</sup> Flash DSCs with the easy-to-use, powerful graphical user interface of the MPLAB Integrated Development Environment (IDE), included with each kit.

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 incircuit debugger systems (RJ11) 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 IDE. In upcoming releases of MPLAB IDE, new devices will be supported, and new features will be added. MPLAB REAL ICE offers significant advantages over competitive emulators including low-cost, full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, a ruggedized probe interface and long (up to three meters) interconnection cables.

#### 17.9 MPLAB ICD 3 In-Circuit Debugger System

MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost effective high-speed hardware debugger/programmer for Microchip Flash Digital Signal Controller (DSC) and microcontroller (MCU) devices. It debugs and programs PIC<sup>®</sup> Flash microcontrollers and dsPIC<sup>®</sup> DSCs with the powerful, yet easyto-use graphical user interface of MPLAB Integrated Development Environment (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.

# 17.10 PICkit 3 In-Circuit Debugger/ Programmer and PICkit 3 Debug Express

The MPLAB PICkit 3 allows debugging and programming of PIC<sup>®</sup> and dsPIC<sup>®</sup> Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB Integrated Development Environment (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 an 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<sup>™</sup>.

The PICkit 3 Debug Express include the PICkit 3, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.







#### TABLE 18-1: COMPARATOR SPECIFICATIONS

<b>Operating Conditions:</b> 3.0V < VDD < 5.5V, -40°C < TA < +85°C (unless otherwise stated).									
Param No.	Sym	Characteristics	Min	Тур	Мах	Units	Comments		
D300	VIOFF	Input Offset Voltage	_	± 5.0	± 10	mV			
D301	VICM	Input Common Mode Voltage*	0	_	Vdd - 1.5	V			
D302	CMRR	Common Mode Rejection Ratio*	55	_	—	dB			
300 300A	Tresp	Response Time <sup>(1)*</sup>	_	150	400 600	ns ns	PIC16F7X7 PIC16LF7X7		
301	TMC2OV	Comparator Mode Change to Output Valid*	—	—	10	μS			
	These n	arameters are characterized but not	tested	•	•		•		

These parameters are characterized but not tested.

Note 1: Response time measured with one comparator input at (VDD - 1.5)/2, while the other input transitions from Vss to VDD.

### TABLE 18-2: VOLTAGE REFERENCE SPECIFICATIONS

<b>Operating Conditions:</b> $3.0V < VDD < 5.5V$ , $-40^{\circ}C < TA < +85^{\circ}C$ (unless otherwise stated).								
Param No.	Sym	Characteristics	Min	Тур	Мах	Units	Comments	
D310	VRES	Resolution	Vdd/24		Vdd/32	LSb		
D311	VRAA	Absolute Accuracy	_		1/4 1/2	LSb LSb	Low Range (CVRR = 1) High Range (CVRR = 0)	
D312	VRur	Unit Resistor Value (R)*	—	2k	—	Ω		
310	TSET	Settling Time <sup>(1)*</sup>	_		10	μS		

\* These parameters are characterized but not tested.

**Note 1:** Settling time measured while CVRR = 1 and CVR<3:0> transition from '0000' to '1111'.



#### TABLE 18-9: PARALLEL SLAVE PORT REQUIREMENTS (PIC16F747/777 DEVICES ONLY)

Param No.	Symbol	Characteristic		Min	Тур†	Max	Units	Conditions
62	TdtV2wrH	Data In Valid before $\overline{WR} \uparrow$ or $\overline{CS} \uparrow$ (setup time)		20 25	_		ns ns	Extended range only
63*	TwrH2dtI	$\overline{WR}$ $\uparrow$ or $\overline{CS}$ $\uparrow$ to Data In Invalid	PIC16F7X7	20	—	_	ns	
		(hold time)	PIC16LF7X7	35	—		ns	
64	TrdL2dtV	$\overline{RD} \downarrow and \ \overline{CS} \downarrow to \ Data \ Out \ Valid$		—	—	80	ns	Extended range only
						90	ns	Extended range only
65	TrdH2dtI	$\overline{RD}$ $\uparrow$ or $\overline{CS} \downarrow$ to Data Out Invalid		10	—	30	ns	
* These parameters are characterized but not tested								

These parameters are characterized but not tested.

t Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.





#### FIGURE 19-12: AVERAGE FOSC vs. VDD FOR VARIOUS VALUES OF R (RC MODE, C = 100 pF, +25°C)



DS30498D-page 240

### 28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging





Units		MILLIMETERS				
Dimension L		MIN	NOM	MAX		
Number of Pins	И	28				
Pitch	е	1.27 BSC				
Overall Height	A	2.6				
Molded Package Thickness	A2	2.05		-		
Standoff §	A1	0.10	-	0.30		
Overall Width	E	10.30 BSC				
Molded Package Width	E1	7.50 BSC				
Overall Length	D	17.90 BSC				
Chamfer (Optional)	h	0.25	-	0.75		
Foot Length	L	0.40	-	1.27		
Footprint	L1	1.40 REF				
Lead Angle	O	0°	-	-		
Foot Angle	φ	0°	-	8°		
Lead Thickness	С	0.18	-	0.33		
Lead Width	b	0.31	-	0.51		
Mold Draft Angle Top	α	5°	-	15°		
Mold Draft Angle Bottom	β	5°	-	15°		

#### Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. § Significant Characteristic
- Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M BSC: Basic Dimension. Theoretically exact value shown without tolerances. REF: Reference Dimension, usually without tolerance, for information purposes only.
- 5. Datums A & B to be determined at Datum H.

Microchip Technology Drawing C04-052C Sheet 2 of 2

28-Lead Plastic Shrink Small Outline (SS) - 5.30 mm Body [SSOP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



	MILLIMETERS			
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Contact Pad Spacing	С		7.20	
Contact Pad Width (X28)	X1			0.45
Contact Pad Length (X28)	Y1			1.75
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2073A

NOTES: