



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

Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	10MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	33
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	368 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf877a-i-ml

Email: info@E-XFL.COM

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

7.1 Timer2 Prescaler and Postscaler

The prescaler and postscaler counters are cleared when any of the following occurs:

- a write to the TMR2 register
- a write to the T2CON register
- any device Reset (POR, MCLR Reset, WDT Reset or BOR)

TMR2 is not cleared when T2CON is written.

7.2 Output of TMR2

The output of TMR2 (before the postscaler) is fed to the SSP module, which optionally uses it to generate the shift clock.

TABLE 7-1: REGISTERS ASSOCIATED WITH TIMER2 AS A TIMER/COUNTER

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value POR,	e on: BOR	Value all o Res	e on other sets
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	TMR0IE	INTE	RBIE	TMR0IF	INTF	RBIF	0000	000x	0000	000u
0Ch	PIR1	PSPIF ⁽¹⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000	0000	0000	0000
8Ch	PIE1	PSPIE ⁽¹⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000	0000	0000	0000
11h	TMR2	Timer2 M	Timer2 Module's Register							0000	0000	0000	0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000	0000	-000	0000
92h	PR2	Timer2 P	ïmer2 Period Register							1111	1111	1111	1111

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by the Timer2 module.

Note 1: Bits PSPIE and PSPIF are reserved on 28-pin devices; always maintain these bits clear.

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 o Res	e on other sets
0Bh,8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	TMR0IE	INTE	RBIE	TMR0IF	INTF	RBIF	0000	000x	0000	000u
0Ch	PIR1	PSPIF ⁽¹⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000	0000	0000	0000
0Dh	PIR2	_	_	_	_	_	—	_	CCP2IF		0		0
8Ch	PIE1	PSPIE ⁽¹⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000	0000	0000	0000
8Dh	PIE2	—	—	—	—		—	—	CCP2IE		0		0
87h	TRISC	PORTC D	PORTC Data Direction Register								1111	1111	1111
11h	TMR2	Timer2 M	Fimer2 Module's Register								0000	0000	0000
92h	PR2	Timer2 M	odule's Peri	od Register						1111	1111	1111	1111
12h	T2CON	_	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000	0000	-000	0000
15h	CCPR1L	Capture/C	Compare/PV	VM Registe	r 1 (LSB)					xxxx	xxxx	uuuu	uuuu
16h	CCPR1H	Capture/C	Compare/PV	VM Registe	r 1 (MSB)					xxxx	xxxx	uuuu	uuuu
17h	CCP1CON	_	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	00	0000	00	0000
1Bh	CCPR2L	Capture/C	Capture/Compare/PWM Register 2 (LSB)							xxxx	xxxx	uuuu	uuuu
1Ch	CCPR2H	Capture/C	apture/Compare/PWM Register 2 (MSB)							uuuu	uuuu		
1Dh	CCP2CON		_	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	00	0000	00	0000

TABLE 8-5: REGISTERS ASSOCIATED WITH PWM AND TIMER2

 $\label{eq:logend: Legend: Legend: x = unknown, u = unchanged, - = unimplemented, read as `0'. Shaded cells are not used by PWM and Timer2.$

Note 1: Bits PSPIE and PSPIF are reserved on 28-pin devices; always maintain these bits clear.

NOTES:

9.3.3 ENABLING SPI I/O

To enable the serial port, SSP Enable bit, SSPEN (SSPCON<5>), must be set. To reset or reconfigure SPI mode, clear the SSPEN bit, re-initialize the SSPCON registers and then set the SSPEN bit. This configures the SDI, SDO, SCK and SS pins as serial port pins. For the pins to behave as the serial port function, some must have their data direction bits (in the TRIS register) appropriately programmed. That is:

- SDI is automatically controlled by the SPI module
- SDO must have TRISC<5> bit cleared
- SCK (Master mode) must have TRISC<3> bit cleared
- SCK (Slave mode) must have TRISC<3> bit set
- SS must have TRISC<4> bit set

Any serial port function that is not desired may be overridden by programming the corresponding data direction (TRIS) register to the opposite value.

9.3.4 TYPICAL CONNECTION

Figure 9-2 shows a typical connection between two microcontrollers. The master controller (Processor 1) initiates the data transfer by sending the SCK signal. Data is shifted out of both shift registers on their programmed clock edge and latched on the opposite edge of the clock. Both processors should be programmed to the same Clock Polarity (CKP), then both controllers would send and receive data at the same time. Whether the data is meaningful (or dummy data) depends on the application software. This leads to three scenarios for data transmission:

- Master sends data Slave sends dummy data
- Master sends data Slave sends data
- Master sends dummy data Slave sends data



FIGURE 9-2: SPI MASTER/SLAVE CONNECTION

9.4.7.1 Clock Arbitration

Clock arbitration occurs when the master, during any receive, transmit or Repeated Start/Stop condition, deasserts the SCL pin (SCL allowed to float high). When the SCL pin is allowed to float high, the Baud Rate Generator (BRG) is suspended from counting until the SCL pin is actually sampled high. When the

SCL pin is sampled high, the Baud Rate Generator is reloaded with the contents of SSPADD<6:0> and begins counting. This ensures that the SCL high time will always be at least one BRG rollover count, in the event that the clock is held low by an external device (Figure 9-17).







10.1 USART Baud Rate Generator (BRG)

The BRG supports both the Asynchronous and Synchronous modes of the USART. It is a dedicated 8-bit baud rate generator. The SPBRG register controls the period of a free running 8-bit timer. In Asynchronous mode, bit BRGH (TXSTA<2>) also controls the baud rate. In Synchronous mode, bit BRGH is ignored. Table 10-1 shows the formula for computation of the baud rate for different USART modes which only apply in Master mode (internal clock).

Given the desired baud rate and FOSC, the nearest integer value for the SPBRG register can be calculated using the formula in Table 10-1. From this, the error in baud rate can be determined. It may be advantageous to use the high baud rate (BRGH = 1) even for slower baud clocks. This is because the FOSC/(16 (X + 1)) equation can reduce the baud rate error in some cases.

Writing a new value to the SPBRG register causes the BRG timer to be reset (or cleared). This ensures the BRG does not wait for a timer overflow before outputting the new baud rate.

10.1.1 SAMPLING

The data on the RC7/RX/DT pin is sampled three times by a majority detect circuit to determine if a high or a low level is present at the RX pin.

TABLE 10-1: BAUD RATE FORMULA

SYNC	BRGH = 0 (Low Speed)	BRGH = 1 (High Speed)
0	(Asynchronous) Baud Rate = Fosc/(64 (X + 1))	Baud Rate = Fosc/(16 (X + 1))
1	(Synchronous) Baud Rate = Fosc/(4 (X + 1))	N/A

Legend: X = value in SPBRG (0 to 255)

TABLE 10-2: REGISTERS ASSOCIATED WITH BAUD RATE GENERATOR

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 o Res	e on ther sets
98h	TXSTA	CSRC	TX9	TXEN	SYNC		BRGH	TRMT	TX9D	0000	-010	0000	-010
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000	000x	0000	000x
99h	SPBRG	Baud Ra	aud Rate Generator Register 0000 0					0000	0000	0000			

Legend: x = unknown, - = unimplemented, read as '0'. Shaded cells are not used by the BRG.



FIGURE 10-11: SYNCHRONOUS RECEPTION (MASTER MODE, SREN)

10.4 USART Synchronous Slave Mode

Synchronous Slave mode differs from the Master mode in the fact that the shift clock is supplied externally at the RC6/TX/CK pin (instead of being supplied internally in Master mode). This allows the device to transfer or receive data while in Sleep mode. Slave mode is entered by clearing bit, CSRC (TXSTA<7>).

10.4.1 USART SYNCHRONOUS SLAVE TRANSMIT

The operation of the Synchronous Master and Slave modes is identical, 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:

- a) The first word will immediately transfer to the TSR register and transmit.
- b) The second word will remain in TXREG register.
- c) Flag bit TXIF will not be set.
- d) When the first word has been shifted out of TSR, the TXREG register will transfer the second word to the TSR and flag bit TXIF will now be set.
- e) If enable bit TXIE is set, the interrupt will wake the chip from Sleep and if the global interrupt is enabled, the program will branch to the interrupt vector (0004h).

When setting up a Synchronous Slave Transmission, follow these steps:

- 1. Enable the synchronous slave serial port by setting bits SYNC and SPEN and clearing bit CSRC.
- 2. Clear bits CREN and SREN.
- 3. If interrupts are desired, then set enable bit TXIE.
- 4. If 9-bit transmission is desired, then set bit TX9.
- 5. Enable the transmission by setting enable bit TXEN.
- 6. If 9-bit transmission is selected, the ninth bit should be loaded in bit TX9D.
- 7. Start transmission by loading data to the TXREG register.
- 8. If using interrupts, ensure that GIE and PEIE (bits 7 and 6) of the INTCON register are set.

11.2 Selecting the A/D Conversion Clock

The A/D conversion time per bit is defined as TAD. The A/D conversion requires a minimum 12 TAD per 10-bit conversion. The source of the A/D conversion clock is software selected. The seven possible options for TAD are:

- 2 Tosc
- 4 Tosc
- 8 Tosc
- 16 Tosc
- 32 Tosc
- 64 Tosc
- Internal A/D module RC oscillator (2-6 μs)

For correct A/D conversions, the A/D conversion clock (TaD) must be selected to ensure a minimum TaD time of 1.6 $\mu s.$

Table 11-1 shows the resultant TAD times derived from the device operating frequencies and the A/D clock source selected.

11.3 Configuring Analog Port Pins

The ADCON1 and TRIS registers control the operation of the A/D port pins. The port pins that are desired as analog inputs must have their corresponding TRIS bits set (input). If the TRIS bit is cleared (output), the digital output level (VOH or VOL) will be converted.

The A/D operation is independent of the state of the CHS2:CHS0 bits and the TRIS bits.

- Note 1: When reading the port register, any pin configured as an analog input channel will read as cleared (a low level). Pins configured as digital inputs will convert an analog input. Analog levels on a digitally configured input will not affect the conversion accuracy.
 - 2: Analog levels on any pin that is defined as a digital input (including the AN7:AN0 pins) may cause the input buffer to consume current that is out of the device specifications.

AD Clo	Maximum Davias Francesou			
Operation	ADCS2:ADCS1:ADCS0			
2 Tosc	000	1.25 MHz		
4 Tosc	100	2.5 MHz		
8 Tosc	001	5 MHz		
16 Tosc	101	10 MHz		
32 Tosc	010	20 MHz		
64 Tosc	110	20 MHz		
RC ^(1, 2, 3)	x11	(Note 1)		

TABLE 11-1: TAD vs. MAXIMUM DEVICE OPERATING FREQUENCIES (STANDARD DEVICES (F))

Note 1: The RC source has a typical TAD time of 4 μ s but can vary between 2-6 μ s.

2: When the device frequencies are greater than 1 MHz, the RC A/D conversion clock source is only recommended for Sleep operation.

3: For extended voltage devices (LF), please refer to Section 17.0 "Electrical Characteristics".





12.6 Comparator Interrupts

The comparator interrupt flag is set whenever there is a change in the output value of either comparator. Software will need to maintain information about the status of the output bits, as read from CMCON<7:6>, to determine the actual change that occurred. The CMIF bit (PIR registers) is the Comparator Interrupt Flag. The CMIF bit must be reset by clearing it ('0'). Since it is also possible to write a '1' to this register, a simulated interrupt may be initiated.

The CMIE bit (PIE registers) and the PEIE bit (INTCON register) must be set to enable the interrupt. In addition, the GIE bit must also be set. If any of these bits are clear, the interrupt is not enabled, though the CMIF bit will still be set if an interrupt condition occurs.

Note: If a change in the CMCON register (C1OUT or C2OUT) should occur when a read operation is being executed (start of the Q2 cycle), then the CMIF (PIR registers) interrupt flag may not get set.

The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

- a) Any read or write of CMCON will end the mismatch condition.
- b) Clear flag bit CMIF.

A mismatch condition will continue to set flag bit CMIF. Reading CMCON will end the mismatch condition and allow flag bit CMIF to be cleared.

13.0 COMPARATOR VOLTAGE REFERENCE MODULE

The Comparator Voltage Reference Generator is a 16-tap resistor ladder network that provides a fixed voltage reference when the comparators are in mode '110'. A programmable register controls the function of the reference generator. Register 13-1 lists the bit functions of the CVRCON register.

As shown in Figure 13-1, the resistor ladder is segmented to provide two ranges of CVREF values and has a power-down function to conserve power when the reference is not being used. The comparator reference supply voltage (also referred to as CVRSRC) comes directly from VDD. It should be noted, however, that the voltage at the top of the ladder is CVRSRC – VSAT, where VSAT is the saturation voltage of the power switch transistor. This reference will only be as accurate as the values of CVRSRC and VSAT.

The output of the reference generator may be connected to the RA2/AN2/VREF-/CVREF pin. This can be used as a simple D/A function by the user if a very highimpedance load is used. The primary purpose of this function is to provide a test path for testing the reference generator function.

	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-C				
	CVREN	CVROE	CVRR	_	CVR3	CVR2	CVR1	CVR0				
	bit 7							bi				
bit 7	CVREN : Co 1 = CVREF	omparator V	oltage Refe	rence Enabl	e bit							
	0 = CVREF circuit powered down											
bit 6	CVROE: C	omparator V	REF Output	Enable bit								
	 1 = CVREF voltage level is output on RA2/AN2/VREF-/CVREF pin 0 = CVREF voltage level is disconnected from RA2/AN2/VREF-/CVREF pin 											
bit 5	CVRR: Cor	mparator VR	EF Range S	election bit								

REGISTER 13-1: CVRCON CONTROL REGISTER (ADDRESS 9Dh)

0 = 0.25 CVRSRC to 0.75 CVRSRC, with CVRSRC/32 step size

1 = 0 to 0.75 CVRSRC, with CVRSRC/24 step size

bit 4 Unimplemented: Read as '0'

bit 3-0 **CVR3:CVR0:** Comparator VREF Value Selection bits $0 \le VR3:VR0 \le 15$ <u>When CVRR = 1:</u> CVREF = (VR<3:0>/ 24) • (CVRSRC) <u>When CVRR = 0:</u> CVREF = 1/4 • (CVRSRC) + (VR3:VR0/ 32) • (CVRSRC)

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

14.3 Reset

The PIC16F87XA differentiates between various kinds of Reset:

- Power-on Reset (POR)
- MCLR Reset during normal operation
- MCLR Reset during Sleep
- WDT Reset (during normal operation)
- WDT Wake-up (during Sleep)
- Brown-out Reset (BOR)

Some registers are not affected in any Reset condition. Their status is unknown on POR and unchanged in any other Reset. Most other registers are reset to a "Reset state" on Power-on Reset (POR), on the MCLR and WDT Reset, on MCLR Reset during Sleep and Brownout Reset (BOR). They are not affected by a WDT wake-up which is viewed as the resumption of normal operation. The TO and PD bits are set or cleared differently in different Reset situations as indicated in Table 14-4. These bits are used in software to determine the nature of the Reset. See Table 14-6 for a full description of Reset states of all registers.

A simplified block diagram of the on-chip Reset circuit is shown in Figure 14-4.

FIGURE 14-4: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



CALL	Call Subroutine
Syntax:	[<i>label</i>] CALL k
Operands:	$0 \leq k \leq 2047$
Operation:	(PC)+ 1 \rightarrow TOS, k \rightarrow PC<10:0>, (PCLATH<4:3>) \rightarrow PC<12:11>
Status Affected:	None
Description:	Call Subroutine. First, return address (PC+1) is pushed onto the stack. The eleven-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a two-cycle instruction.

CLRWDT	Clear Watchdog Timer
Syntax:	[label] CLRWDT
Operands:	None
Operation: Status Affected:	$\begin{array}{l} 00h \rightarrow WDT \\ 0 \rightarrow WDT \ prescaler, \\ 1 \rightarrow \overline{TO} \\ 1 \rightarrow \overline{PD} \\ \overline{TO}, \ \overline{PD} \end{array}$
Description:	CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits, \overline{TO} and \overline{PD} , are set.

CLRF	Clear f
Syntax:	[label] CLRF f
Operands:	$0 \leq f \leq 127$
Operation:	$\begin{array}{l} 00h \rightarrow (f) \\ 1 \rightarrow Z \end{array}$
Status Affected:	Z
Description:	The contents of register 'f' are cleared and the Z bit is set.

COMF	Complement f
Syntax:	[<i>label</i>] COMF f,d
Operands:	$\begin{array}{l} 0 \leq f \leq 127 \\ d \in [0,1] \end{array}$
Operation:	$(\overline{f}) \rightarrow (destination)$
Status Affected:	Z
Description:	The contents of register 'f' are complemented. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'.

CLRW	Clear W	DECF	Decrement f [label] DECF f,d		
Syntax:	[label] CLRW	Syntax:			
Operands:	None	Operands:	$\begin{array}{l} 0 \leq f \leq 127 \\ d \in [0,1] \end{array}$		
Operation:	$00h \rightarrow (W)$				
	$1 \rightarrow Z$	Operation:	(f) - 1 \rightarrow (destination)		
Status Affected:	Z	Status Affected:	Z		
Description: W register is cleared. Zero bit (Z) is set.		Description:	Decrement register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.		

16.3 MPLAB C17 and MPLAB C18 C Compilers

The MPLAB C17 and MPLAB C18 Code Development Systems are complete ANSI C compilers for Microchip's PIC17CXXX and PIC18CXXX family of microcontrollers. These compilers provide powerful integration capabilities, superior code optimization and ease of use not found with other compilers.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

16.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK object linker combines relocatable objects created by the MPASM assembler and the MPLAB C17 and MPLAB C18 C compilers. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB object librarian manages the creation and modification of library files of pre-compiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

16.5 MPLAB C30 C Compiler

The MPLAB C30 C compiler is a full-featured, ANSI compliant, optimizing compiler that translates standard ANSI C programs into dsPIC30F assembly language source. The compiler also supports many command-line options and language extensions to take full advantage of the dsPIC30F device hardware capabilities, and afford fine control of the compiler code generator.

MPLAB C30 is distributed with a complete ANSI C standard library. All library functions have been validated and conform to the ANSI C library standard. The library includes functions for string manipulation, dynamic memory allocation, data conversion, time-keeping, and math functions (trigonometric, exponential and hyperbolic). The compiler provides symbolic information for high level source debugging with the MPLAB IDE.

16.6 MPLAB ASM30 Assembler, Linker, and Librarian

MPLAB ASM30 assembler produces relocatable machine code from symbolic assembly language for dsPIC30F devices. MPLAB C30 compiler uses the assembler to produce it's object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire dsPIC30F instruction set
- · Support for fixed-point and floating-point data
- Command line interface
- Rich directive set
- Flexible macro language
- MPLAB IDE compatibility

16.7 MPLAB SIM Software Simulator

The MPLAB SIM software simulator allows code development in a PC hosted environment by simulating the PIC series microcontrollers on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a file, or user defined key press, to any pin. The execution can be performed in Single-Step, Execute Until Break, or Trace mode.

The MPLAB SIM simulator fully supports symbolic debugging using the MPLAB C17 and MPLAB C18 C Compilers, as well as the MPASM assembler. The software simulator offers the flexibility to develop and debug code outside of the laboratory environment, making it an excellent, economical software development tool.

16.8 MPLAB SIM30 Software Simulator

The MPLAB SIM30 software simulator allows code development in a PC hosted environment by simulating the dsPIC30F series microcontrollers on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a file, or user defined key press, to any of the pins.

The MPLAB SIM30 simulator fully supports symbolic debugging using the MPLAB C30 C Compiler and MPLAB ASM30 assembler. The simulator runs in either a Command Line mode for automated tasks, or from MPLAB IDE. This high speed simulator is designed to debug, analyze and optimize time intensive DSP routines.

16.9 MPLAB ICE 2000 High Performance Universal In-Circuit Emulator

The MPLAB ICE 2000 universal in-circuit emulator is intended to provide the product development engineer with a complete microcontroller design tool set for PIC microcontrollers. Software control of the MPLAB ICE 2000 in-circuit emulator is advanced by the MPLAB Integrated Development Environment, which allows editing, building, downloading and source debugging from a single environment.

The MPLAB ICE 2000 is a full-featured emulator system with enhanced trace, trigger and data monitoring features. Interchangeable processor modules allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the MPLAB ICE in-circuit emulator allows expansion to support new PIC microcontrollers.

The MPLAB ICE 2000 in-circuit emulator system has been designed as a real-time emulation system with advanced features that are typically found on more expensive development tools. The PC platform and Microsoft[®] Windows 32-bit operating system were chosen to best make these features available in a simple, unified application.

16.10 MPLAB ICE 4000 High Performance Universal In-Circuit Emulator

The MPLAB ICE 4000 universal in-circuit emulator is intended to provide the product development engineer with a complete microcontroller design tool set for highend PIC microcontrollers. Software control of the MPLAB ICE in-circuit emulator is provided by the MPLAB Integrated Development Environment, which allows editing, building, downloading and source debugging from a single environment.

The MPLAB ICD 4000 is a premium emulator system, providing the features of MPLAB ICE 2000, but with increased emulation memory and high speed performance for dsPIC30F and PIC18XXXX devices. Its advanced emulator features include complex triggering and timing, up to 2 Mb of emulation memory, and the ability to view variables in real-time.

The MPLAB ICE 4000 in-circuit emulator system has been designed as a real-time emulation system with advanced features that are typically found on more expensive development tools. The PC platform and Microsoft Windows 32-bit operating system were chosen to best make these features available in a simple, unified application.

16.11 MPLAB ICD 2 In-Circuit Debugger

Microchip's In-Circuit Debugger, MPLAB ICD 2, is a powerful, low-cost, run-time development tool, connecting to the host PC via an RS-232 or high-speed USB interface. This tool is based on the Flash PIC MCUs and can be used to develop for these and other PIC microcontrollers. The MPLAB ICD 2 utilizes the incircuit debugging capability built into the Flash devices. This feature, along with Microchip's In-Circuit Serial Programming[™] (ICSP[™]) protocol, offers cost effective in-circuit Flash debugging from the graphical user interface of the MPLAB Integrated Development Environment. This enables a designer to develop and debug source code by setting breakpoints, singlestepping and watching variables, CPU status and peripheral registers. Running at full speed enables testing hardware and applications in real-time. MPLAB ICD 2 also serves as a development programmer for selected PIC devices.

16.12 PRO MATE II Universal Device Programmer

The PRO MATE II is a universal, CE compliant device programmer with programmable voltage verification at VDDMIN and VDDMAX for maximum reliability. It features an LCD display for instructions and error messages and a modular detachable socket assembly to support various package types. In Stand-Alone mode, the PRO MATE II device programmer can read, verify, and program PIC devices without a PC connection. It can also set code protection in this mode.

16.13 PICSTART Plus Development Programmer

The PICSTART Plus development programmer is an easy-to-use, low-cost, prototype programmer. It connects to the PC via a COM (RS-232) port. MPLAB Integrated Development Environment software makes using the programmer simple and efficient. The PICSTART Plus development programmer supports most PIC devices up to 40 pins. Larger pin count devices, such as the PIC16C92X and PIC17C76X, may be supported with an adapter socket. The PICSTART Plus development programmer is CE compliant.



TABLE 17-3:	EXTERNAL	CLOCK TIMING	REQUIREMENTS
-------------	----------	---------------------	--------------

Param No.	Symbol	Characteristic	Min	Тур†	Max	Units	Conditions
	Fosc	External CLKI Frequency (Note 1)	DC		1	MHz	XT and RC Osc mode
			DC	—	20	MHz	HS Osc mode
			DC	—	32	kHz	LP Osc mode
		Oscillator Frequency (Note 1)	DC	—	4	MHz	RC Osc mode
			0.1	—	4	MHz	XT Osc mode
			4	—	20	MHz	HS Osc mode
			5	—	200	kHz	LP Osc mode
1	Tosc	External CLKI Period (Note 1)	1000	—		ns	XT and RC Osc mode
			50	—	—	ns	HS Osc mode
			5	—		μs	LP Osc mode
		Oscillator Period (Note 1)	250	—	—	ns	RC Osc mode
			250	—	1	μS	XT Osc mode
			100	—	250	ns	HS Osc mode
			50	—	250	ns	HS Osc mode
			31.25	—		μS	LP Osc mode
2	Тсү	Instruction Cycle Time (Note 1)	200	Тсү	DC	ns	Tcy = 4/Fosc
3	TosL, TosH	External Clock in (OSC1) High or Low Time	100	_	_	ns	XT oscillator
			2.5	—	—	μS	LP oscillator
			15	—	—	ns	HS oscillator
4	TosR, TosF	External Clock in (OSC1) Rise or Fall Time	—	_	25	ns	XT oscillator
			—	—	50	ns	LP oscillator
			—	—	15	ns	HS oscillator

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

Note 1: Instruction cycle period (TCY) equals four 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 OSC1/CLKI pin. When an external clock input is used, the "max." cycle time limit is "DC" (no clock) for all devices.

FIGURE 17-9: CAPTURE/COMPARE/PWM TIMINGS (CCP1 AND CCP2)



TABLE 17-7: CAPTURE/COMPARE/PWM REQUIREMENTS (CCP1 AND CCP2)

Param No.	Symbol	Characteristic			Min	Тур†	Max	Units	Conditions
50*	TCCL	CCP1 and CCP2 Input Low Time	No Prescaler		0.5 TCY + 20	—	—	ns	
			With Prescaler	Standard(F)	10	_	_	ns	
				Extended(LF)	20	_	_	ns	
51*	ТссН	CCP1 and CCP2 No Presca Input High Time With Presc	No Prescaler		0.5 TCY + 20	—	_	ns	
			With Prescaler	Standard(F)	10	—		ns	
				Extended(LF)	20	_	_	ns	
52*	TCCP	CCP1 and CCP2 Input Period			<u>3 Tcy + 40</u> N	—	_	ns	N = prescale value (1, 4 or 16)
53*	TccR	CCP1 and CCP2 Output Rise Time Standard(F) Extended(LF)		Standard(F)	_	10	25	ns	
				Extended(LF)	-	25	50	ns	
54*	TCCF	CCP1 and CCP2 Output Fall Time S		Standard(F)	-	10	25	ns	
				Extended(LF)	_	25	45	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 18-8: AVERAGE Fosc vs. VDD FOR VARIOUS VALUES OF R (RC MODE, C = 100 pF, +25°C)











NOTES: