



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	4MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	33
Program Memory Size	7KB (4K x 14)
Program Memory Type	ОТР
EEPROM Size	-
RAM Size	192 x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 6V
Data Converters	A/D 5x8b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-QFP
Supplier Device Package	44-MQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lc74a-04i-pq

Email: info@E-XFL.COM

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

PIC16C7X

Pin Diagrams



4.2.2.6 PIE2 REGISTER Applicable Devices 72 73 73 74 74 76 77

This register contains the individual enable bit for the CCP2 peripheral interrupt.

FIGURE 4-14: PIE2 REGISTER (ADDRESS 8Dh)



5.7 Parallel Slave Port Applicable Devices 72 73 73 74 74 76 77

PORTD operates as an 8-bit wide Parallel Slave Port, or microprocessor port when control bit PSPMODE (TRISE<4>) is set. In slave mode it is asynchronously readable and writable by the external world through \overline{RD} control input pin RE0/ \overline{RD} /AN5 and \overline{WR} control input pin RE1/ \overline{WR} /AN6.

It can directly interface to an 8-bit microprocessor data bus. The external microprocessor can read or write the PORTD latch as an 8-bit latch. Setting bit PSPMODE enables port pin RE0/RD/AN5 to be the RD input, RE1/ WR/AN6 to be the WR input and RE2/CS/AN7 to be the CS (chip select) input. For this functionality, the corresponding data direction bits of the TRISE register (TRISE<2:0>) must be configured as inputs (set) and the A/D port configuration bits PCFG2:PCFG0 (ADCON1<2:0>) must be set, which will configure pins RE2:RE0 as digital I/O.

There are actually two 8-bit latches, one for data-out (from the PIC16/17) and one for data input. The user writes 8-bit data to PORTD data latch and reads data from the port pin latch (note that they have the same address). In this mode, the TRISD register is ignored, since the microprocessor is controlling the direction of data flow.

A write to the PSP occurs when both the \overline{CS} and \overline{WR} lines are first detected low. When either the \overline{CS} or \overline{WR} lines become high (level triggered), then the Input Buffer Full status flag bit IBF (TRISE<7>) is set on the Q4 clock cycle, following the next Q2 cycle, to signal the write is complete (Figure 5-12). The interrupt flag bit PSPIF (PIR1<7>) is also set on the same Q4 clock cycle. IBF can only be cleared by reading the PORTD input latch. The input Buffer Overflow status flag bit IBOV (TRISE<5>) is set if a second write to the Parallel Slave Port is attempted when the previous byte has not been read out of the buffer.

A read from the PSP occurs when both the \overline{CS} and \overline{RD} lines are first detected low. The Output Buffer Full status flag bit OBF (TRISE<6>) is cleared immediately (Figure 5-13) indicating that the PORTD latch is waiting to be read by the external bus. When either the \overline{CS} or \overline{RD} pin becomes high (level triggered), the interrupt flag bit PSPIF is set on the Q4 clock cycle, following the next Q2 cycle, indicating that the read is complete. OBF remains low until data is written to PORTD by the user firmware.

When not in Parallel Slave Port mode, the IBF and OBF bits are held clear. However, if flag bit IBOV was previously set, it must be cleared in firmware.

An interrupt is generated and latched into flag bit PSPIF when a read or write operation is completed. PSPIF must be cleared by the user in firmware and the interrupt can be disabled by clearing the interrupt enable bit PSPIE (PIE1<7>).

FIGURE 5-11: PORTD AND PORTE BLOCK DIAGRAM (PARALLEL SLAVE PORT)



10.0 CAPTURE/COMPARE/PWM MODULE(s)

 Applicable Devices

 72
 73
 73A
 74
 74A
 76
 77
 CCP1

 72
 73
 73A
 74
 74A
 76
 77
 CCP2

Each CCP (Capture/Compare/PWM) module contains a 16-bit register which can operate as a 16-bit capture register, as a 16-bit compare register or as a PWM master/slave Duty Cycle register. Both the CCP1 and CCP2 modules are identical in operation, with the exception of the operation of the special event trigger. Table 10-1 and Table 10-2 show the resources and interactions of the CCP module(s). In the following sections, the operation of a CCP module is described with respect to CCP1. CCP2 operates the same as CCP1, except where noted.

CCP1 module:

Capture/Compare/PWM Register1 (CCPR1) is comprised of two 8-bit registers: CCPR1L (low byte) and CCPR1H (high byte). The CCP1CON register controls the operation of CCP1. All are readable and writable.

CCP2 module:

Capture/Compare/PWM Register2 (CCPR2) is comprised of two 8-bit registers: CCPR2L (low byte) and CCPR2H (high byte). The CCP2CON register controls the operation of CCP2. All are readable and writable.

For use of the CCP modules, refer to the Embedded Control Handbook, "Using the CCP Modules" (AN594).

TABLE 10-1: CCP MODE - TIMER RESOURCE

CCP Mode	Timer Resource
Capture	Timer1
Compare	Timer1
PWM	Timer2

TABLE 10-2: INTERACTION OF TWO CCP MODULES

CCPx Mode	CCPy Mode	Interaction
Capture	Capture	Same TMR1 time-base.
Capture	Compare	The compare should be configured for the special event trigger, which clears TMR1.
Compare	Compare	The compare(s) should be configured for the special event trigger, which clears TMR1.
PWM	PWM	The PWMs will have the same frequency, and update rate (TMR2 interrupt).
PWM	Capture	None
PWM	Compare	None

10.1.4 CCP PRESCALER

There are four prescaler settings, specified by bits CCP1M3:CCP1M0. Whenever the CCP module is turned off, or the CCP module is not in capture mode, the prescaler counter is cleared. This means that any reset will clear the prescaler counter.

Switching from one capture prescaler to another may generate an interrupt. Also, the prescaler counter will not be cleared, therefore the first capture may be from a non-zero prescaler. Example 10-1 shows the recommended method for switching between capture prescalers. This example also clears the prescaler counter and will not generate the "false" interrupt.

EXAMPLE 10-1: CHANGING BETWEEN CAPTURE PRESCALERS

CLRF	CCP1CON	;Turn CCP module off
MOVLW	NEW_CAPT_PS	;Load the W reg with
		; the new prescaler
		; mode value and CCP ON
MOVWF	CCP1CON	;Load CCP1CON with this
		; value

10.2 <u>Compare Mode</u>

Applicable Devices

In Compare mode, the 16-bit CCPR1 register value is constantly compared against the TMR1 register pair value. When a match occurs, the RC2/CCP1 pin is:

- Driven High
- Driven Low
- · Remains Unchanged

The action on the pin is based on the value of control bits CCP1M3:CCP1M0 (CCP1CON<3:0>). At the same time, interrupt flag bit CCP1IF is set.

FIGURE 10-3: COMPARE MODE OPERATION BLOCK DIAGRAM



10.2.1 CCP PIN CONFIGURATION

The user must configure the RC2/CCP1 pin as an output by clearing the TRISC<2> bit.

Note:	Clearing the CCP1CON register will force
	the RC2/CCP1 compare output latch to the
	default low level. This is not the data latch.

10.2.2 TIMER1 MODE SELECTION

Timer1 must be running in Timer mode or Synchronized Counter mode if the CCP module is using the compare feature. In Asynchronous Counter mode, the compare operation may not work.

10.2.3 SOFTWARE INTERRUPT MODE

When generate software interrupt is chosen the CCP1 pin is not affected. Only a CCP interrupt is generated (if enabled).

10.2.4 SPECIAL EVENT TRIGGER

In this mode, an internal hardware trigger is generated which may be used to initiate an action.

The special event trigger output of CCP1 resets the TMR1 register pair. This allows the CCPR1 register to effectively be a 16-bit programmable period register for Timer1.

The special trigger output of CCP2 resets the TMR1 register pair, and starts an A/D conversion (if the A/D module is enabled).

For the PIC16C72 only, the special event trigger output of CCP1 resets the TMR1 register pair, and starts an A/D conversion (if the A/D module is enabled).

Note: The special event trigger from the CCP1and CCP2 modules will not set interrupt flag bit TMR1IF (PIR1<0>).

The \overline{SS} pin allows a synchronous slave mode. The SPI must be in slave mode (SSPCON<3:0> = 04h) and the TRISA<5> bit must be set for the synchronous slave mode to be enabled. 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. If the \overline{SS} pin is taken low without resetting SPI mode, the transmission will continue from the point at which it was taken high. External pull-up/ pull-down resistors may be desirable, depending on the application.

Note:	When the SPI is in Slave Mode with \overline{SS} pin
	the SPI module will reset if the \overline{SS} pin is set
	to VDD.
Note:	If the SPI is used in Slave Mode with

CKE = '1', then the SS pin control must be enabled. 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

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 11-11: SPI MODE TIMING, MASTER MODE (PIC16C76/77)

FIGURE 11-12: SPI MODE TIMING (SLAVE MODE WITH CKE = 0) (PIC16C76/77)



TABLE 12-8: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER TRANSMISSION

Address	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
0Ch	PIR1	PSPIF ⁽¹⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	_	FERR	OERR	RX9D	0000 -00x	0000 -00x
19h	TXREG	USART Tra	ansmit Re	0000 0000	0000 0000						
8Ch	PIE1	PSPIE ⁽¹⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate	Generat	or Regis	ter					0000 0000	0000 0000

Legend: x = unknown, - = unimplemented, read as '0'. Shaded cells are not used for Synchronous Master Transmission. Note 1: Bits PSPIE and PSPIF are reserved on the PIC16C73/73A/76, always maintain these bits clear.

FIGURE 12-12: SYNCHRONOUS TRANSMISSION



FIGURE 12-13: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)



12.4 USART Synchronous Slave Mode

Applicable Devices 72 73 73A 74 74A 76 77

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>).

12.4.1 USART SYNCHRONOUS SLAVE TRANSMIT

The operation of the synchronous master and slave modes are 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).

Steps to follow when setting up a Synchronous Slave Transmission:

- 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.

12.4.2 USART SYNCHRONOUS SLAVE RECEPTION

The operation of the synchronous master and slave modes is identical except in the case of the SLEEP mode. Also, bit SREN is a don't care in slave mode.

If receive is enabled, by setting bit CREN, prior to the SLEEP instruction, then a word may be received during SLEEP. On completely receiving the word, the RSR register will transfer the data to the RCREG register and if enable bit RCIE bit is set, the interrupt generated will wake the chip from SLEEP. If the global interrupt is enabled, the program will branch to the interrupt vector (0004h).

Steps to follow when setting up a Synchronous Slave Reception:

- 1. Enable the synchronous master serial port by setting bits SYNC and SPEN and clearing bit CSRC.
- 2. If interrupts are desired, then set enable bit RCIE.
- 3. If 9-bit reception is desired, then set bit RX9.
- 4. To enable reception, set enable bit CREN.
- 5. Flag bit RCIF will be set when reception is complete and an interrupt will be generated, if enable bit RCIE was set.
- 6. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
- 7. Read the 8-bit received data by reading the RCREG register.
- 8. If any error occurred, clear the error by clearing bit CREN.

FIGURE 14-2: CONFIGURATION WORD FOR PIC16C72/73A/74A/76/77

CP1 (CP0	CP1	CP0	CP1	CP0	—	BODEN	CP1	CP0	PWRTE	WDTE	FOSC1	FOSC0	Register: Address	CONFIG 2007h
bit 13-8 5-4:	CP1 11 = 10 = 01 = 00 =	1:CP0 : = Code = Uppe = Uppe = All m	Code e protec er half o er 3/4th nemory	Protect ction of of progr of prog i of prog	ion bits f am mei gram m e protec	(2) mory o emory ted	code prot v code pro	ected otected	I				DITU		
bit 7:	Uni	mplen	nented	: Read	as '1'										
bit 6:	BO 1 = 0 =	DEN: E BOR 6 BOR 0	Brown-o enableo disableo	out Res 1 d	et Enat	ole bit	(1)								
bit 3:	PW 1 = 0 =	rte : F Pwrt Pwrt	Power-u disabl enabl	up Time ed ed	r Enabl	e bit (1)								
bit 2:	WDTE: Watchdog Timer Enable bit 1 = WDT enabled 0 = WDT disabled														
bit 1-0:	FO 11 = 10 = 01 = 00 =	SC1:F (= RC c = HS c = XT o = LP o	OSCO: oscillato oscillato oscillato scillato	Oscillat or or r r	tor Sele	ction	bits								
Note 1: 2:	Ena Ens All c	abling E sure the of the (Brown- e Powe CP1:CF	out Res er-up Tir P0 pairs	et auto ner is e have te	matica nable o be g	ally enabl d anytime iven the	les Pov e Brow same v	ver-up n-out F value to	Timer (P Reset is e o enable	WRT) r enabled the cod	egardle I. le prote	ess of the	value of bit F eme listed.	PWRTE.

14.3 <u>Reset</u> Applicable Devices 72|73|73A|74|74A|76|77

The PIC16CXX differentiates between various kinds of reset:

- Power-on Reset (POR)
- MCLR reset during normal operation
- MCLR reset during SLEEP
- WDT Reset (normal operation)
- Brown-out Reset (BOR) (PIC16C72/73A/74A/76/ 77)

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 $\overline{\text{MCLR}}$ and WDT Reset, on $\overline{\text{MCLR}}$ 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 $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are set or cleared differently in different reset situations as indicated in Table 14-5 and Table 14-6. These bits are used in software to determine the nature of the reset. See Table 14-8 for a full description of reset states of all registers.

A simplified block diagram of the on-chip reset circuit is shown in Figure 14-8.

The PIC16C72/73A/74A/76/77 have a $\overline{\text{MCLR}}$ noise filter in the $\overline{\text{MCLR}}$ reset path. The filter will detect and ignore small pulses.

It should be noted that a WDT Reset does not drive $\overline{\text{MCLR}}$ pin low.







FIGURE 14-11: TIME-OUT SEQUENCE ON POWER-UP (MCLR NOT TIED TO VDD): CASE 2



FIGURE 14-12: TIME-OUT SEQUENCE ON POWER-UP (MCLR TIED TO VDD)



RLF	Rotate L	.eft f thre	ough Ca	rry	RRF	Rotate F	Right f th	rough C	arry
Syntax:	[<i>label</i>] RLF f,d 0 ≤ f ≤ 127				Syntax:	[label]	RRF f	,d	
Operands:	0 ≤ f ≤ 12 d ∈ [0,1]	27			Operands:	$0 \le f \le 12$ $d \in [0,1]$	27		
Operation:	Rotate Left f through Carry $[label]$ RLFf,d $[label]$ RLFf,d $0 \le f \le 127$ $d \in [0,1]$ See description belowOperation:See description belowd:C 00 1101dffffffffffThe contents of register 'f' are rotated one bit to the left through the Carry Flag. If 'd' is 0 the result is placed in the wr register. If 'd is 1 the result is stored back in register 'f'.Operation:See description below1 \mathbf{V} \mathbf{Q} \mathbf{Q} \mathbf{Q} \mathbf{Q} \mathbf{Q} \mathbf{U} <t< td=""><td>elow</td><td></td></t<>			elow					
Status Affected:	С				Status Affected:	С			
Encoding:	00	1101	dfff	ffff	Encoding:	00	1100	dfff	ffff
Description:	The contents of register 'f' are rotated one bit to the left through the Carry Flag. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is stored back in register 'f'.				Description:	The conte one bit to Flag. If 'd' W register back in re	ents of reg the right t is 0 the re r. If 'd' is 1 gister 'f'. $C \rightarrow $	ister 'f' are hrough the esult is pla the result Register f	e rotated e Carry ced in the is placed
Words:	1				Words:	1			
Cycles:	1				Cycles:	1			
Q Cycle Activity:	Q1 Q2 Q3 Q4		Q Cycle Activity:	Q1	Q2	Q3	Q4		
	Decode	Read register 'f'	Process data	Write to destination		Decode	Read register 'f'	Process data	Write to destination
Example	RLF	REG	G1,0		Example	RRF		REG1,0	
	Before Instruction					Before Instruction			
		REG1	= 111	0 0110			REG1	= 111	0 0110
	Aftor Inc	C	= 0			A (1 1	C	= 0	
	Aller IIIS	REG1	- 111	0 0110		After Inst		111	0 0110
		W	= 110	0 1100			W	= 111 = 011	1 0011
		С	= 1				C	= 0	

16.6 <u>PICDEM-1 Low-Cost PIC16/17</u> <u>Demonstration Board</u>

The PICDEM-1 is a simple board which demonstrates the capabilities of several of Microchip's microcontrollers. The microcontrollers supported are: PIC16C5X (PIC16C54 to PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 and PIC17C44. All necessary hardware and software is included to run basic demo programs. The users can program the sample microcontrollers provided with the PICDEM-1 board, on a PRO MATE II or PICSTART-16B programmer, and easily test firmware. The user can also connect the PICDEM-1 board to the PICMASTER emulator and download the firmware to the emulator for testing. Additional prototype area is available for the user to build some additional hardware and connect it to the microcontroller socket(s). Some of the features include an RS-232 interface, a potentiometer for simulated analog input, push-button switches and eight LEDs connected to PORTB.

16.7 <u>PICDEM-2 Low-Cost PIC16CXX</u> Demonstration Board

The PICDEM-2 is a simple demonstration board that supports the PIC16C62, PIC16C64, PIC16C65, PIC16C73 and PIC16C74 microcontrollers. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM-2 board, on a PRO MATE II programmer or PICSTART-16C, and easily test firmware. The PICMASTER emulator may also be used with the PICDEM-2 board to test firmware. Additional prototype area has been provided to the user for adding additional hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push-button switches, a potentiometer for simulated analog input, a Serial EEPROM to demonstrate usage of the I²C bus and separate headers for connection to an LCD module and a keypad.

16.8 <u>PICDEM-3 Low-Cost PIC16CXXX</u> Demonstration Board

The PICDEM-3 is a simple demonstration board that supports the PIC16C923 and PIC16C924 in the PLCC package. It will also support future 44-pin PLCC microcontrollers with a LCD Module. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM-3 board, on a PRO MATE II programmer or PICSTART Plus with an adapter socket, and easily test firmware. The PICMASTER emulator may also be used with the PICDEM-3 board to test firmware. Additional prototype area has been provided to the user for adding hardware and connecting it to the microcontroller socket(s). Some of the features include an RS-232 interface, push-button switches, a potentiometer for simulated analog input, a thermistor and separate headers for connection to an external LCD module and a keypad. Also provided on the PICDEM-3 board is an LCD panel, with 4 commons and 12 segments, that is capable of displaying time, temperature and day of the week. The PICDEM-3 provides an additional RS-232 interface and Windows 3.1 software for showing the demultiplexed LCD signals on a PC. A simple serial interface allows the user to construct a hardware demultiplexer for the LCD signals.

16.9 <u>MPLAB Integrated Development</u> <u>Environment Software</u>

The MPLAB IDE Software brings an ease of software development previously unseen in the 8-bit microcontroller market. MPLAB is a windows based application which contains:

- A full featured editor
- Three operating modes
 - editor
 - emulator
 - simulator
- A project manager
- Customizable tool bar and key mapping
- A status bar with project information

Extensive on-line help

MPLAB allows you to:

- Edit your source files (either assembly or 'C')
- One touch assemble (or compile) and download to PIC16/17 tools (automatically updates all project information)
- Debug using:
- source files
- absolute listing file
- Transfer data dynamically via DDE (soon to be replaced by OLE)
- Run up to four emulators on the same PC

The ability to use MPLAB with Microchip's simulator allows a consistent platform and the ability to easily switch from the low cost simulator to the full featured emulator with minimal retraining due to development tools.

16.10 Assembler (MPASM)

The MPASM Universal Macro Assembler is a PChosted symbolic assembler. It supports all microcontroller series including the PIC12C5XX, PIC14000, PIC16C5X, PIC16CXXX, and PIC17CXX families.

MPASM offers full featured Macro capabilities, conditional assembly, and several source and listing formats. It generates various object code formats to support Microchip's development tools as well as third party programmers.

MPASM allows full symbolic debugging from PICMASTER, Microchip's Universal Emulator System.

	PIC12C5XX	PIC14000	PIC16C5X	PIC16CXXX	PIC16C6X	PIC16C7XX	PIC16C8X	PIC16C9XX	PIC17C4X	PIC17C75X	24CXX 25CXX 93CXX	HCS200 HCS300 HCS301
E ator	>	2	7	7	7	2	2	>	2	Available 3Q97		
ost Ilator	7		7	7	7	7	7					
	2	7	7	7	7	7	7	7	2	7		
	7	7	7	7	7	7	2	>	7	2		
MP noi	7	7	7	7	7	7	7	7	7			
/™ Itor			7	7	7	7	7		7			
ince™ del											7	
w-Cost			7		7	7	2					
ost ev. Kit	7	7	7	7	7	7	7	7	7	7		
=	7	7	7	7	7	7	2	7	7	7	7	7
												7
It											7	
			7	7			7		7			
					7	2						
								7				
Git												7

TABLE 16-1: DEVELOPMENT TOOLS FROM MICROCHIP

Applicable Devices 72 73 73A 74 74A 76 77

TABLE 18-13: A/D CONVERTER CHARACTERISTICS:

PIC16C73/74-04 (Commercial, Industrial) PIC16C73/74-10 (Commercial, Industrial) PIC16C73/74-20 (Commercial, Industrial) PIC16LC73/74-04 (Commercial, Industrial)

Param No.	Sym	Characteristic		Min	Тур†	Мах	Units	Conditions
A01	Nr	Resolution		_	—	8-bits	bit	$\label{eq:VREF} \begin{array}{l} VREF = VDD = 5.12V,\\ VSS \leq VAIN \leq VREF \end{array}$
A02	Eabs	Total Absolute error			_	<±1	LSb	$\label{eq:VREF} \begin{array}{l} VREF = VDD = 5.12V,\\ VSS \leq VAIN \leq VREF \end{array}$
A03	EIL	Integral linearity error		—	—	<±1	LSb	$\label{eq:VREF} \begin{array}{l} VREF = VDD = 5.12V,\\ VSS \leq VAIN \leq VREF \end{array}$
A04	Edl	Differential linearity error		—	_	< ± 1	LSb	$\label{eq:VREF} \begin{array}{l} VREF = VDD = 5.12V,\\ VSS \leq VAIN \leq VREF \end{array}$
A05	Efs	Full scale error		—	_	<±1	LSb	$\label{eq:VREF} \begin{array}{l} VREF = VDD = 5.12V,\\ VSS \leq VAIN \leq VREF \end{array}$
A06	EOFF	Offset error	—	—	<±1	LSb	$\label{eq:VREF} \begin{array}{l} VREF = VDD = 5.12V,\\ VSS \leq VAIN \leq VREF \end{array}$	
A10	_	Monotonicity	—	guaranteed	_	—	$VSS \leq VAIN \leq VREF$	
A20	Vref	Reference voltage	3.0V		Vdd + 0.3	V		
A25	VAIN	Analog input voltage	Vss - 0.3	_	Vref + 0.3	V		
A30	Zain	Recommended impedan analog voltage source	_	—	10.0	kΩ		
A40	IAD A/D conversion current		PIC16 C 73/74	—	180	—	μΑ	Average current consump-
		(VDD)	PIC16 LC 73/74	—	90		μA	tion when A/D is on. (Note 1)
A50	IREF VREF input current (Note 2)			10	_	1000	μA	During VAIN acquisition. Based on differential of VHOLD to VAIN to charge CHOLD, see Section 13.1.
				—	—	10	μA	During A/D Conversion cycle

These parameters are characterized but not tested.

† 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: When A/D is off, it will not consume any current other than minor leakage current. The power-down current spec includes any such leakage from the A/D module.

2: VREF current is from RA3 pin or VDD pin, whichever is selected as reference input.

PIC16C7X

Applicable Devices 72 73 73A 74 74A 76 77





TABLE 19-8: SPI MODE REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Тур†	Max	Units	Conditions
70	TssL2scH, TssL2scL	$\overline{SS}\downarrow$ to SCK \downarrow or SCK \uparrow input	Тсү	_	_	ns	
71	TscH	SCK input high time (slave mode)	Tcy + 20	—	_	ns	
72	TscL	SCK input low time (slave mode)	Tcy + 20	_	_	ns	
73	TdiV2scH, TdiV2scL	Setup time of SDI data input to SCK edge	100	_	_	ns	
74	TscH2diL, TscL2diL	Hold time of SDI data input to SCK edge	100	_	_	ns	
75	TdoR	SDO data output rise time		10	25	ns	
76	TdoF	SDO data output fall time	_	10	25	ns	
77	TssH2doZ	SS↑ to SDO output hi-impedance	10	_	50	ns	
78	TscR	SCK output rise time (master mode)	-	10	25	ns	
79	TscF	SCK output fall time (master mode)		10	25	ns	
80	TscH2doV, TscL2doV	SDO data output valid after SCK edge	_	_	50	ns	

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

Applicable Devices 72 73 73A 74 74A 76 77

20.4 Timing Parameter Symbology

The timing parameter symbols have been created following one of the following formats:

1. TppS2	ppS	3. TCC:ST	(I ² C specifications only)			
2. TppS		4. Ts	(I ² C specifications only)			
Т						
F	Frequency	Т	Time			
Lower	case letters (pp) and their meanings:					
рр						
сс	CCP1	osc	OSC1			
ck	CLKOUT	rd	RD			
CS	CS	rw	RD or WR			
di	SDI	sc	SCK			
do	SDO	SS	SS			
dt	Data in	tO	TOCKI			
io	I/O port	t1	T1CKI			
mc	MCLR	wr	WR			
Upper	case letters and their meanings:					
S						
F	Fall	P	Period			
H	High	R	Rise			
1	Invalid (Hi-impedance)	V	Valid			
L	Low	Z	Hi-impedance			
I ² C only						
AA	output access	High	High			
BUF	Bus free	Low	Low			
Tcc:st	(I ² C specifications only)	_				
CC						
HD	Hold	SU	Setup			
ST						
DAT	DATA input hold	STO	STOP condition			
STA	START condition					
FIGURE	20-1: LOAD CONDITIONS					
			Logd condition 2			
	Load condition 1		Load condition 2			
	VDD/2					
	Pin	P				
	▼ Vss		▼ Vss			
	$R_{1} = 464\Omega$		100			
	$n_{\rm L} = 40432$					
	CL = 50 pF for all pins except OSC2, but including PORTD and PORTE outputs as ports					
	15 pF for OSC2 output					
Note: PORTD and PORTE are not implemented on the PIC16C76.						

Applicable Devices 72 73 73A 74 74A 76 77

21.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES

The graphs and tables provided in this section are for design guidance and are not tested or guaranteed.

In some graphs or tables the data presented are outside specified operating range (i.e., outside specified VDD range). This is for information only and devices are guaranteed to operate properly only within the specified range.

Note: The data presented in this section is a statistical summary of data collected on units from different lots over a period of time and matrix samples. 'Typical' represents the mean of the distribution at, 25°C, while 'max' or 'min' represents (mean $+3\sigma$) and (mean -3σ) respectively where σ is standard deviation.

FIGURE 21-1: TYPICAL IPD vs. VDD (WDT DISABLED, RC MODE)



FIGURE 21-2: MAXIMUM IPD vs. VDD (WDT DISABLED, RC MODE)



22.6 28-Lead Plastic Surface Mount (SSOP - 209 mil Body 5.30 mm) (SS)



Package Group: Plastic SSOP							
	Millimeters			Inches			
Symbol	Min	Max	Notes	Min	Мах	Notes	
α	0°	8°		0°	8°		
A	1.730	1.990		0.068	0.078		
A1	0.050	0.210		0.002	0.008		
В	0.250	0.380		0.010	0.015		
С	0.130	0.220		0.005	0.009		
D	10.070	10.330		0.396	0.407		
E	5.200	5.380		0.205	0.212		
е	0.650	0.650	Reference	0.026	0.026	Reference	
Н	7.650	7.900		0.301	0.311		
L	0.550	0.950		0.022	0.037		
Ν	28	28		28	28		
CP	-	0.102		-	0.004		

LIST OF FIGURES

Figure 3-1:	PIC16C72 Block Diagram 10
Figure 3-2	PIC16C73/73A/76 Block Diagram 11
Figure 3-3	PIC16C74/74A/77 Block Diagram 12
Figure 3-4	Clock/Instruction Cycle 17
Figure 4-1	PIC16C72 Program Memory Map
riguio + 1.	and Stack 19
Figure 4-2	PIC16C73/73A/74/74A Program
rigure + 2.	Memory Man and Stack 10
Figure 4-3	PIC16C76/77 Program Memory
rigure 4-5.	Man and Stack 20
Figure 4-4	PIC16C72 Register File Man 21
Figure 4-5:	PIC16C73/73A/74/74A Register
rigure + 0.	File Man 21
Figure 4-6	PIC16C76/77 Register File Map 22
Figure 4-7:	Status Register (Address 03b
rigure + 7.	83h 103h 183h) 30
Figure 4-8	OPTION Register (Address 81h
rigure + 0.	181h) 31
Figure 4-9	INTCON Register
riguro + 0.	(Address OBb 8Bb 10bb 18bb) 32
Figure 4-10	PIF1 Register PIC16C72
riguie 4 io.	(Address 8Ch) 33
Figure 4-11	PIF1 Register PIC16C73/73A/
rigure + 11.	74/74A/76/77 (Address 8Ch) 34
Figure 4-12	PIR1 Register PIC16C72
rigulo i iz.	(Address 0Ch)
Figure 4-13	PIR1 Register PIC16C73/73A/
rigure + ro.	74/74A/76/77 (Address 0Ch) 36
Figure 4-14	PIF2 Register (Address 8Dh) 37
Figure 4-15	PIR2 Register (Address 0Dh) 38
Figure 4-16:	PCON Register (Address 8Eh) 39
Figure 4-17	Loading of PC In Different
rigulo i ii.	Situations 40
Figure 4-18:	Direct/Indirect Addressing 41
Figure 5-1:	Block Diagram of RA3:RA0
. iguie e ii	and RA5 Pins 43
Figure 5-2:	Block Diagram of RA4/T0CKI Pin43
Figure 5-3:	Block Diagram of RB3:RB0 Pins
Figure 5-4:	Block Diagram of RB7:RB4 Pins
0	(PIC16C73/74)
Figure 5-5:	Block Diagram of
0	RB7:RB4 Pins (PIC16C72/73A/
	74A/76/77)46
Figure 5-6:	
	PORTC Block Diagram
0	PORTC Block Diagram (Peripheral Output Override)
Figure 5-7:	PORTC Block Diagram (Peripheral Output Override)48 PORTD Block Diagram
Figure 5-7:	PORTC Block Diagram (Peripheral Output Override)
Figure 5-7: Figure 5-8:	PORTC Block Diagram (Peripheral Output Override)
Figure 5-7: Figure 5-8:	PORTC Block Diagram (Peripheral Output Override)
Figure 5-7: Figure 5-8: Figure 5-9:	PORTC Block Diagram (Peripheral Output Override)
Figure 5-7: Figure 5-8: Figure 5-9: Figure 5-10:	PORTC Block Diagram (Peripheral Output Override)
Figure 5-7: Figure 5-8: Figure 5-9: Figure 5-10: Figure 5-11:	PORTC Block Diagram (Peripheral Output Override)
Figure 5-7: Figure 5-8: Figure 5-9: Figure 5-10: Figure 5-11:	PORTC Block Diagram (Peripheral Output Override)
Figure 5-7: Figure 5-8: Figure 5-9: Figure 5-10: Figure 5-11: Figure 5-12:	PORTC Block Diagram (Peripheral Output Override)
Figure 5-7: Figure 5-8: Figure 5-9: Figure 5-10: Figure 5-11: Figure 5-12: Figure 5-13:	PORTC Block Diagram (Peripheral Output Override)
Figure 5-7: Figure 5-8: Figure 5-9: Figure 5-10: Figure 5-11: Figure 5-12: Figure 5-13: Figure 7-1:	PORTC Block Diagram (Peripheral Output Override) 48 PORTD Block Diagram (in I/O Port Mode) 50 PORTE Block Diagram (in I/O Port Mode) 51 TRISE Register (Address 89h) 51 Successive I/O Operation 53 PORTD and PORTE Block Diagram (Parallel Slave Port) 54 Parallel Slave Port Write Waveforms 55 Parallel Slave Port Read Waveforms 55 Timer0 Block Diagram
Figure 5-7: Figure 5-8: Figure 5-9: Figure 5-10: Figure 5-11: Figure 5-12: Figure 5-13: Figure 5-13: Figure 7-1: Figure 7-2:	PORTC Block Diagram (Peripheral Output Override) 48 PORTD Block Diagram (in I/O Port Mode) 50 PORTE Block Diagram (in I/O Port Mode) 51 TRISE Register (Address 89h) 51 Successive I/O Operation 53 PORTD and PORTE Block Diagram (Parallel Slave Port) 54 Parallel Slave Port Write Waveforms 55 Parallel Slave Port Read Waveforms 55 Timer0 Block Diagram 59 Timer0 Timing: Internal Clock/No
Figure 5-7: Figure 5-8: Figure 5-9: Figure 5-10: Figure 5-11: Figure 5-12: Figure 5-13: Figure 5-13: Figure 7-1: Figure 7-2:	PORTC Block Diagram (Peripheral Output Override) 48 PORTD Block Diagram (in I/O Port Mode) 50 PORTE Block Diagram (in I/O Port Mode) 51 TRISE Register (Address 89h) 51 Successive I/O Operation 53 PORTD and PORTE Block Diagram (Parallel Slave Port) 54 Parallel Slave Port Write Waveforms 55 Parallel Slave Port Read Waveforms 55 Timer0 Block Diagram 59 Timer0 Timing: Internal Clock/No Prescale
Figure 5-7: Figure 5-8: Figure 5-9: Figure 5-10: Figure 5-11: Figure 5-12: Figure 5-13: Figure 5-13: Figure 7-1: Figure 7-2: Figure 7-3:	PORTC Block Diagram (Peripheral Output Override) 48 PORTD Block Diagram (in I/O Port Mode) 50 PORTE Block Diagram (in I/O Port Mode) (in I/O Port Mode) (in I/O Port Mode) 51 TRISE Register (Address 89h) 51 Successive I/O Operation 53 PORTD and PORTE Block Diagram (Parallel Slave Port) 54 Parallel Slave Port Write Waveforms 55 Timer0 Block Diagram 59 Timer0 Block Diagram 59 Timer0 Timing: Internal Clock/No Prescale 59 Timer0 Timing: Internal
Figure 5-7: Figure 5-8: Figure 5-9: Figure 5-10: Figure 5-11: Figure 5-12: Figure 5-13: Figure 5-13: Figure 7-1: Figure 7-2: Figure 7-3:	PORTC Block Diagram (Peripheral Output Override) 48 PORTD Block Diagram (in I/O Port Mode) 50 PORTE Block Diagram (in I/O Port Mode) (in I/O Port Mode) (in I/O Port Mode) 51 TRISE Register (Address 89h) 51 Successive I/O Operation 53 PORTD and PORTE Block Diagram (Parallel Slave Port) 54 Parallel Slave Port Write Waveforms 55 Timer0 Block Diagram 59 Timer0 Block Diagram 59 Timer0 Timing: Internal Clock/No Prescale 59 Timer0 Timing: Internal Clock/Prescale 1:2
Figure 5-7: Figure 5-8: Figure 5-9: Figure 5-10: Figure 5-11: Figure 5-12: Figure 5-13: Figure 5-13: Figure 7-1: Figure 7-2: Figure 7-3: Figure 7-4:	PORTC Block Diagram (Peripheral Output Override) 48 PORTD Block Diagram (in I/O Port Mode) PORTE Block Diagram (in I/O Port Mode) (in I/O Port Mode) 51 TRISE Register (Address 89h) 51 Successive I/O Operation 53 PORTD and PORTE Block Diagram (Parallel Slave Port) 54 Parallel Slave Port Write Waveforms 55 Parallel Slave Port Read Waveforms 55 Timer0 Block Diagram 59 Timer0 Timing: Internal Clock/No Prescale 59 Timer0 Timing: Internal Clock/Prescale 1:2 60 Timer0 Interrupt Timing
Figure 5-7: Figure 5-8: Figure 5-9: Figure 5-10: Figure 5-11: Figure 5-12: Figure 5-13: Figure 5-13: Figure 7-1: Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5:	PORTC Block Diagram (Peripheral Output Override) 48 PORTD Block Diagram 50 (in I/O Port Mode) 50 PORTE Block Diagram 51 TRISE Register (Address 89h) 51 Successive I/O Operation 53 PORTD and PORTE Block Diagram 54 Parallel Slave Port) 54 Parallel Slave Port Write Waveforms 55 Parallel Slave Port Read Waveforms 59 Timer0 Block Diagram 59 Timer0 Timing: Internal Clock/No 59 Prescale 59 Timer0 Timing: Internal 60 Clock/Prescale 1:2 60 Timer0 Timing with External Clock 61
Figure 5-7: Figure 5-8: Figure 5-9: Figure 5-10: Figure 5-11: Figure 5-12: Figure 5-13: Figure 5-13: Figure 7-1: Figure 7-2: Figure 7-3: Figure 7-4: Figure 7-5: Figure 7-6:	PORTC Block Diagram (Peripheral Output Override) 48 PORTD Block Diagram (in I/O Port Mode) PORTE Block Diagram (in I/O Port Mode) (in I/O Port Mode) 51 TRISE Register (Address 89h) 51 Successive I/O Operation 53 PORTD and PORTE Block Diagram (Parallel Slave Port) 54 Parallel Slave Port Write Waveforms 55 Parallel Slave Port Read Waveforms 59 Timer0 Block Diagram 59 Timer0 Timing: Internal Clock/No Prescale 59 Timer0 Timing: Internal Clock/Prescale 1:2 60 Timer0 Timing with External Clock 61 Block Diagram of the Timer0/WDT

Figure 8-1:	T1CON: Timer1 Control Register (Address 10h)65	5
Figure 8-2:	Timer1 Block Diagram	3
Figure 9-1:	Timer2 Block Diagram 69	à
Figure 9-2	T2CON: Timer2 Control Register	
rigule 5 2.	(Address 12b) 70	h
Figure 10-1	CCP1CON Register (Address 17b)/	,
Figure 10-1.	CCP2CON Register (Address 1711)/	•
F '	CCP2CON Register (Address TDI)	-
Figure 10-2:	Capture Mode Operation	
	Block Diagram	<u>'</u>
Figure 10-3:	Compare Mode Operation	
	Block Diagram73	3
Figure 10-4:	Simplified PWM Block Diagram 74	ł
Figure 10-5:	PWM Output74	ŧ
Figure 11-1:	SSPSTAT: Sync Serial Port Status	
	Register (Address 94h)78	3
Figure 11-2:	SSPCON: Sync Serial Port Control	
-	Register (Address 14h)79)
Figure 11-3:	SSP Block Diagram (SPI Mode) 80)
Figure 11-4:	SPI Master/Slave Connection 81	1
Figure 11-5	SPI Mode Timing Master Mode	
rigulo il ol	or Slave Mode w/o SS Control 82	,
Figure 11-6	SPI Mode Timing Slave Mode with	-
rigule 11-0.	SE Control	,
Figure 11 7	SS COILLOI	-
Figure 11-7:	SSPSTAT: Sync Senar Pont Status	
	Register (Address 94n)(PIC16C76/77) 83	5
Figure 11-8:	SSPCON: Sync Serial Port Control	
	Register (Address 14h)(PIC16C76/77) 84	ł
Figure 11-9:	SSP Block Diagram (SPI Mode)	
	(PIC16C76/77)	5
Figure 11-10:	SPI Master/Slave Connection	
	PIC16C76/77) 86	5
Figure 11-11:	SPI Mode Timing, Master Mode	
0	(PIC16C76/77)	7
Figure 11-12:	SPI Mode Timing (Slave Mode	
	With $CKF = 0$ (PIC16C76/77) 87	7
Figure 11-13	SPI Mode Timing (Slave Mode	
rigule il lo.	With $CKE = 1$ (PIC16C76/77)	2
Figure 11-14	Start and Stop Conditions	, 2
Figure 11 15:	7 bit Address Format	י ר
Figure 11-15.	1 ² C 10 bit Address Format	, ,
Figure 11-16.	Class as a free Ash and a free an	,
Figure 11-17:	Slave-receiver Acknowledge)
Figure 11-18:	Data Transfer Wait State)
Figure 11-19:	Master-transmitter Sequence 91]
Figure 11-20:	Master-receiver Sequence 91	
Figure 11-21:	Combined Format 91	
Figure 11-22:	Multi-master Arbitration	
	(Two Masters)92	2
Figure 11-23:	Clock Synchronization	2
Figure 11-24:	SSP Block Diagram	
0	(I ² C Mode)	3
Figure 11-25	1 ² C Waveforms for Reception	
1 iguro 11 20.	(7-bit Address)	5
Figuro 11 26	I^2C Wayoforms for Transmission	, ,
Figure 11-20.	(7 bit Address)	
Einung 44 07	(7-bit Address))
Figure 11-27:	Operation of the I ⁻ C Module In	
	IDLE_MODE, RCV_MODE or	
	XMIT_MODE 98	3
Figure 12-1:	TXSTA: Transmit Status and	
	Control Register (Address 98h) 99)
Figure 12-2:	RCSTA: Receive Status and	
	Control Register (Address 18h) 100)
Figure 12-3:	RX Pin Sampling Scheme. BRGH = 0	
-	(PIC16C73/73A/74/74A) 104	ŧ
Figure 12-4:	RX Pin Sampling Scheme. BRGH = 1	
5	(PIC16C73/73A/74/74A)	1
	, , , , , , , , , , , , , , , , , , , ,	