



Welcome to [E-XFL.COM](https://www.e-xfl.com)

What is "[Embedded - Microcontrollers](#)"?

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

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	10MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	22
Program Memory Size	7KB (4K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	192 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 6V
Data Converters	A/D 5x8b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16c73a-10i-so

PIC16C7X

TABLE 1-1: PIC16C7XX FAMILY OF DEVICES

		PIC16C710	PIC16C71	PIC16C711	PIC16C715	PIC16C72	PIC16CR72 ⁽¹⁾
Clock	Maximum Frequency of Operation (MHz)	20	20	20	20	20	20
	EPROM Program Memory (x14 words)	512	1K	1K	2K	2K	—
Memory	ROM Program Memory (14K words)	—	—	—	—	—	2K
	Data Memory (bytes)	36	36	68	128	128	128
Peripherals	Timer Module(s)	TMR0	TMR0	TMR0	TMR0	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2
	Capture/Compare/PWM Module(s)	—	—	—	—	1	1
Features	Serial Port(s) (SPI/I ² C, USART)	—	—	—	—	SPI/I ² C	SPI/I ² C
	Parallel Slave Port	—	—	—	—	—	—
	A/D Converter (8-bit) Channels	4	4	4	4	5	5
	Interrupt Sources	4	4	4	4	8	8
	I/O Pins	13	13	13	13	22	22
	Voltage Range (Volts)	3.0-6.0	3.0-6.0	3.0-6.0	3.0-5.5	2.5-6.0	3.0-5.5
	In-Circuit Serial Programming	Yes	Yes	Yes	Yes	Yes	Yes
	Brown-out Reset	Yes	—	Yes	Yes	Yes	Yes
	Packages	18-pin DIP, SOIC, 20-pin SSOP	18-pin DIP, SOIC	18-pin DIP, SOIC, 20-pin SSOP	18-pin DIP, SOIC, 20-pin SSOP	28-pin SDIP, SOIC, SSOP	28-pin SDIP, SOIC, SSOP

		PIC16C73A	PIC16C74A	PIC16C76	PIC16C77
Clock	Maximum Frequency of Operation (MHz)	20	20	20	20
	EPROM Program Memory (x14 words)	4K	4K	8K	8K
Memory	Data Memory (bytes)	192	192	368	368
	Timer Module(s)	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2
Peripherals	Capture/Compare/PWM Module(s)	2	2	2	2
	Serial Port(s) (SPI/I ² C, USART)	SPI/I ² C, USART	SPI/I ² C, USART	SPI/I ² C, USART	SPI/I ² C, USART
Features	Parallel Slave Port	—	Yes	—	Yes
	A/D Converter (8-bit) Channels	5	8	5	8
	Interrupt Sources	11	12	11	12
	I/O Pins	22	33	22	33
	Voltage Range (Volts)	2.5-6.0	2.5-6.0	2.5-6.0	2.5-6.0
	In-Circuit Serial Programming	Yes	Yes	Yes	Yes
	Brown-out Reset	Yes	Yes	Yes	Yes
	Packages	28-pin SDIP, SOIC	40-pin DIP; 44-pin PLCC, MQFP, TQFP	28-pin SDIP, SOIC	40-pin DIP; 44-pin PLCC, MQFP, TQFP

All PIC16/17 Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability. All PIC16C7XX Family devices use serial programming with clock pin RB6 and data pin RB7.

Note 1: Please contact your local Microchip sales office for availability of these devices.

PIC16C7X

NOTES:

3.0 ARCHITECTURAL OVERVIEW

The high performance of the PIC16CXX family can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC16CXX uses a Harvard architecture, in which, program and data are accessed from separate memories using separate buses. This improves bandwidth over traditional von Neumann architecture in which program and data are fetched from the same memory using the same bus. Separating program and data buses further allows instructions to be sized differently than the 8-bit wide data word. Instruction opcodes are 14-bits wide making it possible to have all single word instructions. A 14-bit wide program memory access bus fetches a 14-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions (Example 3-1). Consequently, all instructions (35) execute in a single cycle (200 ns @ 20 MHz) except for program branches.

The table below lists program memory (EPROM) and data memory (RAM) for each PIC16C7X device.

Device	Program Memory	Data Memory
PIC16C72	2K x 14	128 x 8
PIC16C73	4K x 14	192 x 8
PIC16C73A	4K x 14	192 x 8
PIC16C74	4K x 14	192 x 8
PIC16C74A	4K x 14	192 x 8
PIC16C76	8K x 14	368 x 8
PIC16C77	8K x 14	386 x 8

The PIC16CXX can directly or indirectly address its register files or data memory. All special function registers, including the program counter, are mapped in the data memory. The PIC16CXX has an orthogonal (symmetrical) instruction set that makes it possible to carry out any operation on any register using any addressing mode. This symmetrical nature and lack of 'special optimal situations' make programming with the PIC16CXX simple yet efficient. In addition, the learning curve is reduced significantly.

PIC16CXX devices contain an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between the data in the working register and any register file.

The ALU is 8-bits wide and capable of addition, subtraction, shift and logical operations. Unless otherwise mentioned, arithmetic operations are two's complement in nature. In two-operand instructions, typically one operand is the working register (W register). The other operand is a file register or an immediate constant. In single operand instructions, the operand is either the W register or a file register.

The W register is an 8-bit working register used for ALU operations. It is not an addressable register.

Depending on the instruction executed, the ALU may affect the values of the Carry (C), Digit Carry (DC), and Zero (Z) bits in the STATUS register. The C and DC bits operate as a borrow bit and a digit borrow out bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

PIC16C7X

FIGURE 3-1: PIC16C72 BLOCK DIAGRAM

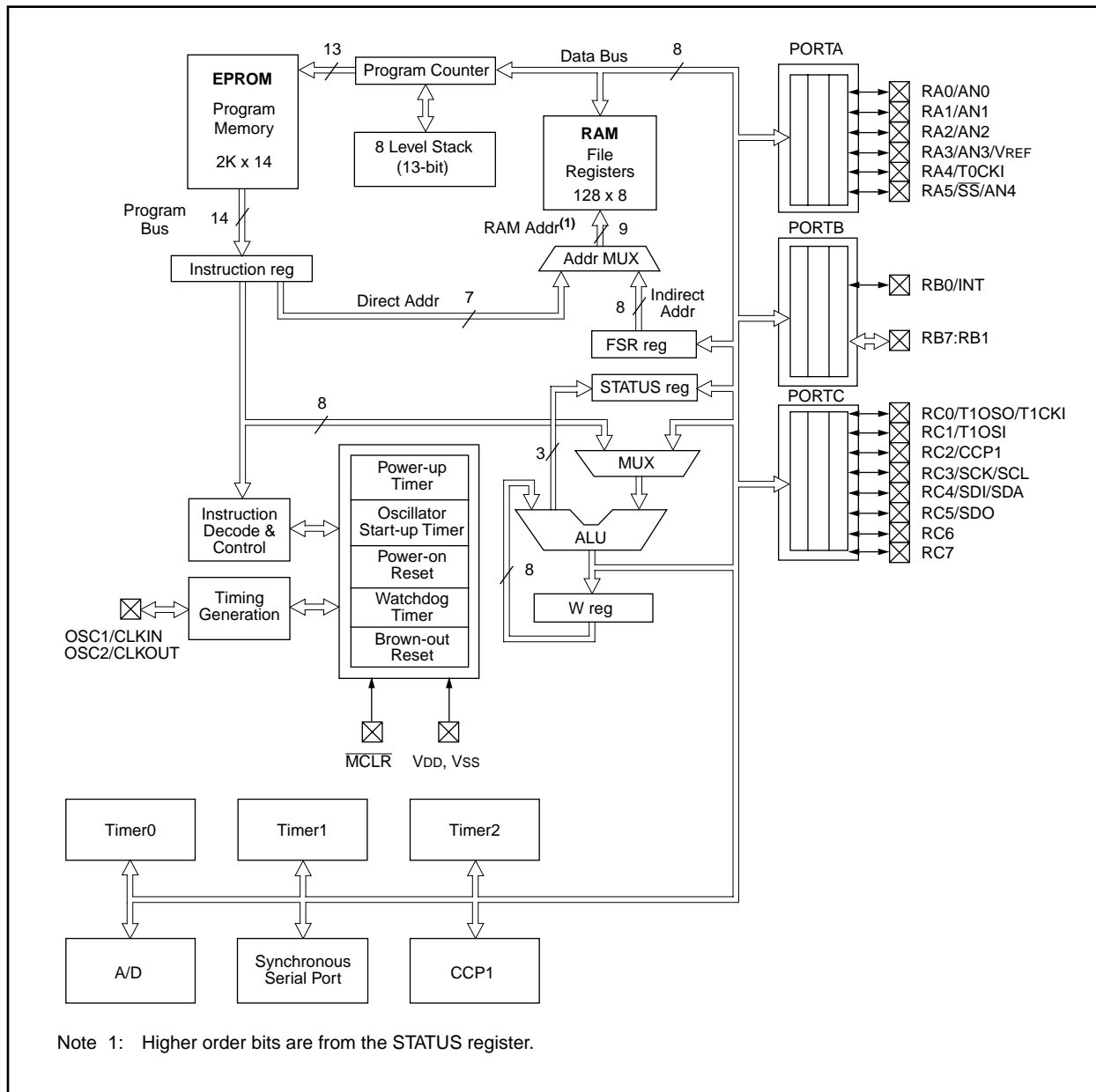



FIGURE 4-4: PIC16C72 REGISTER FILE MAP


File Address			File Address
00h	INDF ⁽¹⁾	INDF ⁽¹⁾	80h
01h	TMR0	OPTION	81h
02h	PCL	PCL	82h
03h	STATUS	STATUS	83h
04h	FSR	FSR	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h	PORTC	TRISC	87h
08h			88h
09h			89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch	PIR1	PIE1	8Ch
0Dh			8Dh
0Eh	TMR1L	PCON	8Eh
0Fh	TMR1H		8Fh
10h	T1CON		90h
11h	TMR2		91h
12h	T2CON	PR2	92h
13h	SSPBUF	SSPADDD	93h
14h	SSPCON	SSPSTAT	94h
15h	CCPR1L		95h
16h	CCPR1H		96h
17h	CCP1CON		97h
18h			98h
19h			99h
1Ah			9Ah
1Bh			9Bh
1Ch			9Ch
1Dh			9Dh
1Eh	ADRES		9Eh
1Fh	ADCON0	ADCON1	9Fh
20h	General Purpose Register	General Purpose Register	A0h
			BFh
			C0h
7Fh	Bank 0	Bank 1	FFh

 Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.

FIGURE 4-5: PIC16C73/73A/74/74A REGISTER FILE MAP

File Address			File Address
00h	INDF ⁽¹⁾	INDF ⁽¹⁾	80h
01h	TMR0	OPTION	81h
02h	PCL	PCL	82h
03h	STATUS	STATUS	83h
04h	FSR	FSR	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h	PORTC	TRISC	87h
08h	PORTD ⁽²⁾	TRISD ⁽²⁾	88h
09h	PORTE ⁽²⁾	TRISE ⁽²⁾	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch	PIR1	PIE1	8Ch
0Dh	PIR2	PIE2	8Dh
0Eh	TMR1L	PCON	8Eh
0Fh	TMR1H		8Fh
10h	T1CON		90h
11h	TMR2		91h
12h	T2CON	PR2	92h
13h	SSPBUF	SSPADD	93h
14h	SSPCON	SSPSTAT	94h
15h	CCPR1L		95h
16h	CCPR1H		96h
17h	CCP1CON		97h
18h	RCSTA	TXSTA	98h
19h	TXREG	SPBRG	99h
1Ah	RCREG		9Ah
1Bh	CCPR2L		9Bh
1Ch	CCPR2H		9Ch
1Dh	CCP2CON		9Dh
1Eh	ADRES		9Eh
1Fh	ADCON0	ADCON1	9Fh
20h			A0h
	General Purpose Register	General Purpose Register	
7Fh			FFh
	Bank 0	Bank 1	


 Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.
 Note 2: These registers are not physically implemented on the PIC16C73/73A, read as '0'.

PIC16C7X

FIGURE 4-6: PIC16C76/77 REGISTER FILE MAP

								File Address
Indirect addr. ^(*)	00h	Indirect addr. ^(*)	80h	Indirect addr. ^(*)	100h	Indirect addr. ^(*)	180h	
TMR0	01h	OPTION	81h	TMR0	101h	OPTION	181h	
PCL	02h	PCL	82h	PCL	102h	PCL	182h	
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h	
FSR	04h	FSR	84h	FSR	104h	FSR	184h	
PORTA	05h	TRISA	85h		105h		185h	
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h	
PORTC	07h	TRISC	87h		107h		187h	
PORTD ⁽¹⁾	08h	TRISD ⁽¹⁾	88h		108h		188h	
PORTE ⁽¹⁾	09h	TRISE ⁽¹⁾	89h		109h		189h	
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah	
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh	
PIR1	0Ch	PIE1	8Ch		10Ch		18Ch	
PIR2	0Dh	PIE2	8Dh		10Dh		18Dh	
TMR1L	0Eh	PCON	8Eh		10Eh		18Eh	
TMR1H	0Fh		8Fh		10Fh		18Fh	
T1CON	10h		90h	General Purpose Register 16 Bytes	110h	General Purpose Register 16 Bytes	190h	
TMR2	11h		91h		111h		191h	
T2CON	12h	PR2	92h		112h		192h	
SSPBUF	13h	SSPADD	93h		113h		193h	
SSPCON	14h	SSPSTAT	94h		114h		194h	
CCPR1L	15h		95h		115h		195h	
CCPR1H	16h		96h		116h		196h	
CCP1CON	17h		97h		117h		197h	
RCSTA	18h	TXSTA	98h		118h		198h	
TXREG	19h	SPBRG	99h		119h		199h	
RCREG	1Ah		9Ah		11Ah		19Ah	
CCPR2L	1Bh		9Bh		11Bh		19Bh	
CCPR2H	1Ch		9Ch		11Ch		19Ch	
CCP2CON	1Dh		9Dh		11Dh		19Dh	
ADRES	1Eh		9Eh		11Eh		19Eh	
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh	
General Purpose Register 96 Bytes	20h	General Purpose Register 80 Bytes	A0h	General Purpose Register 80 Bytes	120h	General Purpose Register 80 Bytes	1A0h	
			EFh		16Fh		1EFh	
			F0h		170h		1F0h	
		accesses 70h-7Fh	FFh	accesses 70h-7Fh	17Fh	accesses 70h - 7Fh	1FFh	
Bank 0	7Fh	Bank 1		Bank 2		Bank 3		

 Unimplemented data memory locations, read as '0'.
^{*} Not a physical register.
 Note 1: PORTD, PORTE, TRISD, and TRISE are unimplemented on the PIC16C76, read as '0'.

Note: The upper 16 bytes of data memory in banks 1, 2, and 3 are mapped in Bank 0. This may require relocation of data memory usage in the user application code if upgrading to the PIC16C76/77.

PIC16C7X

FIGURE 7-3: TMR0 TIMING: INTERNAL CLOCK/PRESCALE 1:2

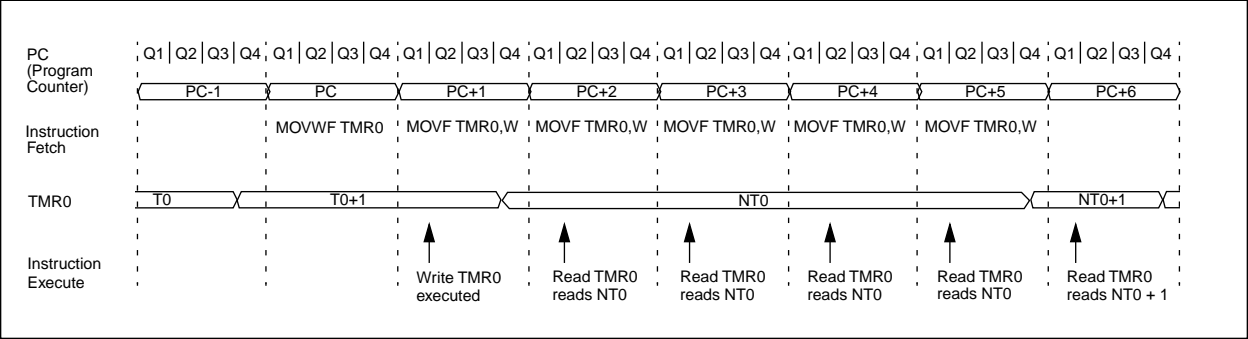
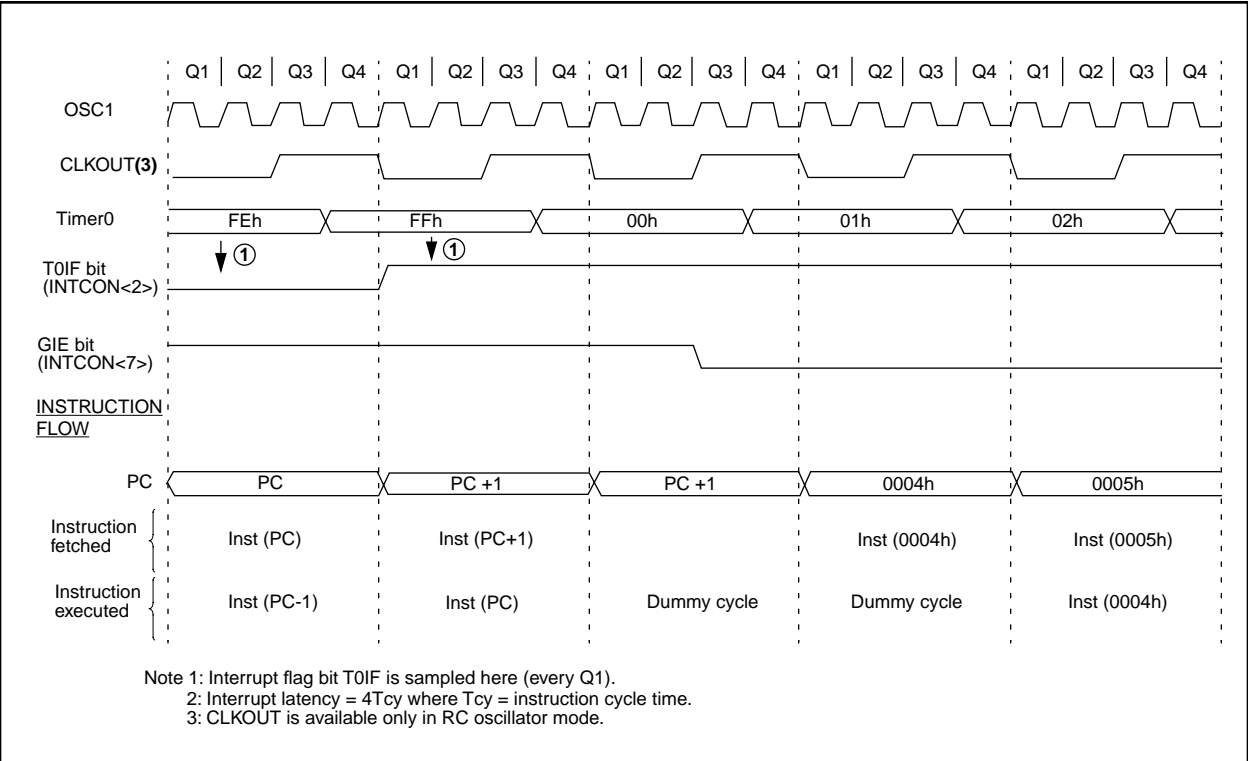


FIGURE 7-4: TMR0 INTERRUPT TIMING



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 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.

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 11-5: SPI MODE TIMING, MASTER MODE OR SLAVE MODE W/O \overline{SS} CONTROL

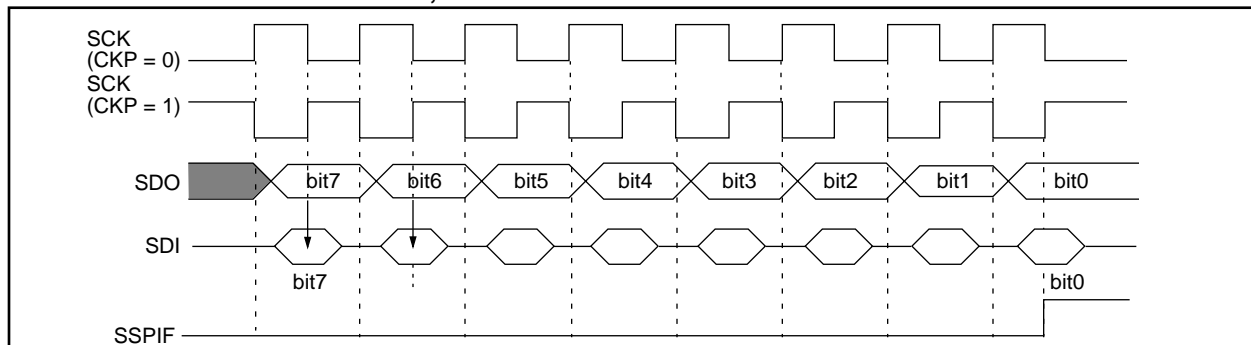


FIGURE 11-6: SPI MODE TIMING, SLAVE MODE WITH \overline{SS} CONTROL

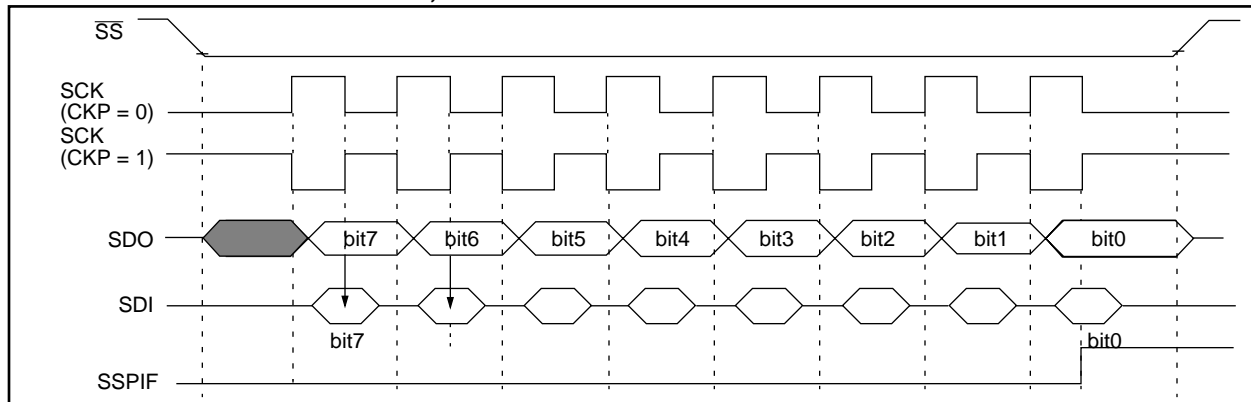


TABLE 11-1: REGISTERS ASSOCIATED WITH SPI OPERATION

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
0Bh,8Bh	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF ^(1,2)	ADIF	RCIF ⁽²⁾	TXIF ⁽²⁾	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE ^(1,2)	ADIE	RCIE ⁽²⁾	TXIE ⁽²⁾	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
94h	SSPSTAT	—	—	D/A	P	S	R/W	UA	BF	--00 0000	--00 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by the SSP in SPI mode.

Note 1: Bits PSPIE and PSPIF are reserved on the PIC16C73/73A, always maintain these bits clear.

2: The PIC16C72 does not have a Parallel Slave Port or USART, these bits are unimplemented, read as '0'.

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

- SDI must have TRISC<4> set
- SDO must have TRISC<5> cleared
- SCK (Master mode) must have TRISC<3> cleared
- SCK (Slave mode) must have TRISC<3> set
- \overline{SS} must have TRISA<5> set

Any serial port function that is not desired may be overridden by programming the corresponding data direction (TRIS) register to the opposite value. An example would be in master mode where you are only sending data (to a display driver), then both SDI and \overline{SS} could be used as general purpose outputs by clearing their corresponding TRIS register bits.

Figure 11-10 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 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 firmware. 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

The master can initiate the data transfer at any time because it controls the SCK. The master determines when the slave (Processor 2) is to broadcast data by the firmware protocol.

In master mode the data is transmitted/received as soon as the SSPBUF register is written to. If the SPI is only going to receive, the SCK output could be disabled (programmed as an input). The SSPSR register will continue to shift in the signal present on the SDI pin at the programmed clock rate. As each byte is received, it will be loaded into the SSPBUF register as if a normal received byte (interrupts and status bits appropriately set). This could be useful in receiver applications as a “line activity monitor” 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 interrupt flag bit SSPIF (PIR1<3>) is set.

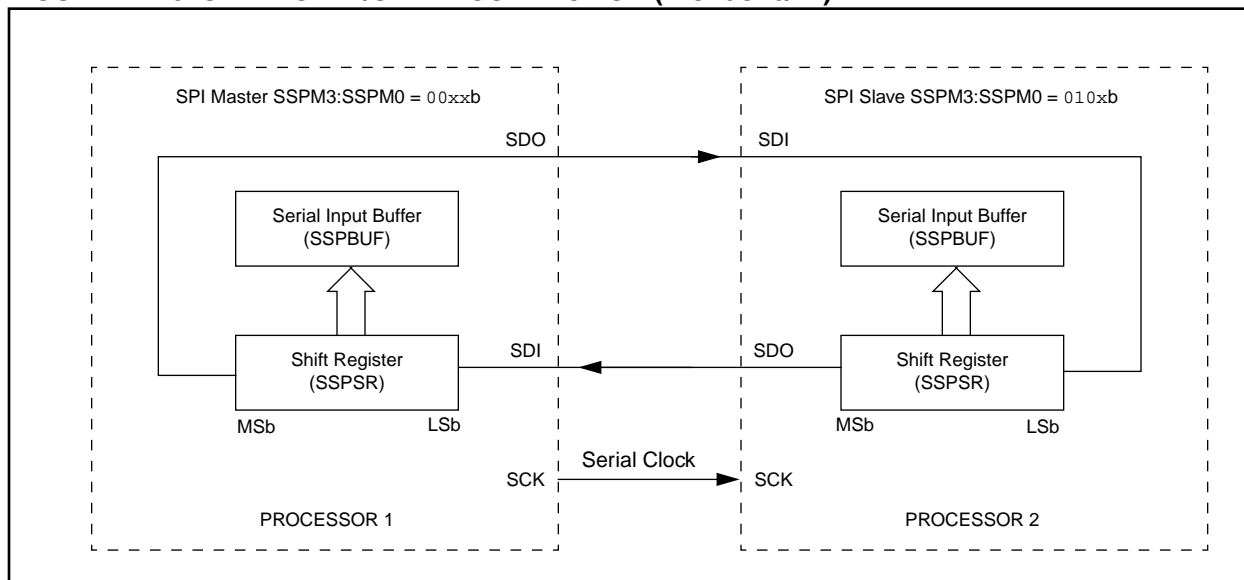
The clock polarity is selected by appropriately programming bit CKP (SSPCON<4>). This then would give waveforms for SPI communication as shown in Figure 11-11, Figure 11-12, and Figure 11-13 where the MSB is transmitted first. In master mode, the SPI clock rate (bit rate) is user programmable to be one of the following:

- $F_{osc}/4$ (or T_{cy})
- $F_{osc}/16$ (or $4 \cdot T_{cy}$)
- $F_{osc}/64$ (or $16 \cdot T_{cy}$)
- Timer2 output/2

This allows a maximum bit clock frequency (at 20 MHz) of 5 MHz. When in slave mode the external clock must meet the minimum high and low times.

In sleep mode, the slave can transmit and receive data and wake the device from sleep.

FIGURE 11-10: SPI MASTER/SLAVE CONNECTION (PIC16C76/77)



PIC16C7X

12.1.1 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. If bit BRGH (TXSTA<2>) is clear (i.e., at the low baud rates), the sampling is done on the seventh, eighth and ninth falling edges of a x16 clock (Figure 12-3). If bit BRGH is

set (i.e., at the high baud rates), the sampling is done on the 3 clock edges preceding the second rising edge after the first falling edge of a x4 clock (Figure 12-4 and Figure 12-5).

FIGURE 12-3: RX PIN SAMPLING SCHEME. BRGH = 0 (PIC16C73/73A/74/74A)

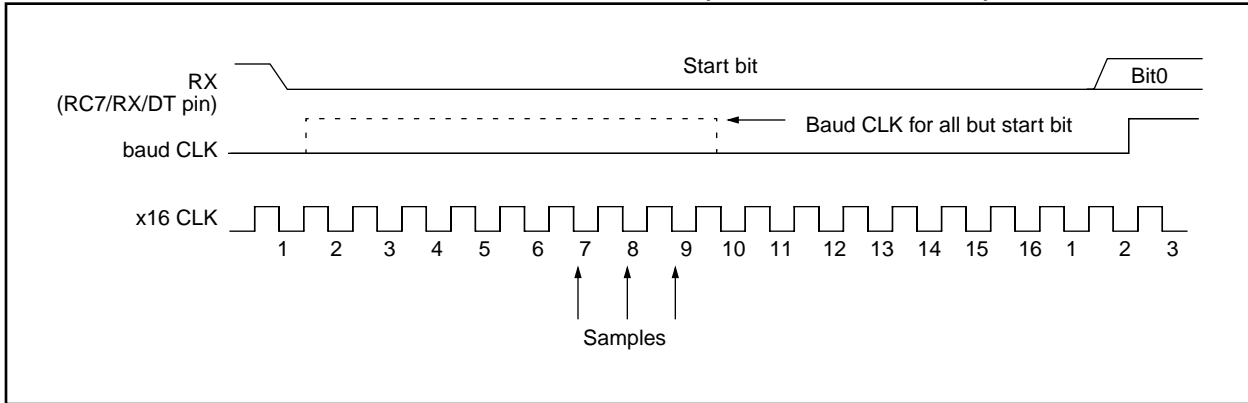


FIGURE 12-4: RX PIN SAMPLING SCHEME, BRGH = 1 (PIC16C73/73A/74/74A)

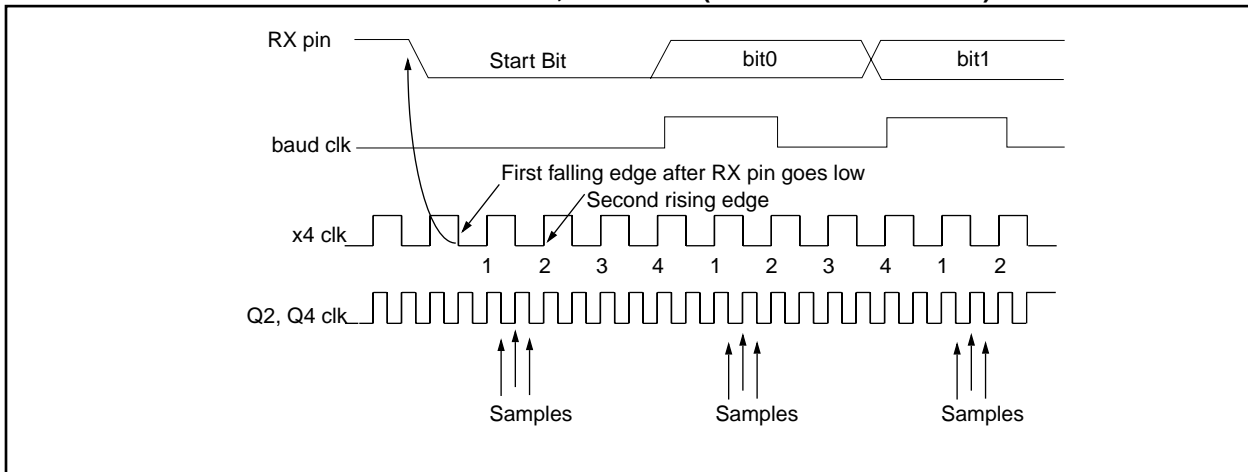
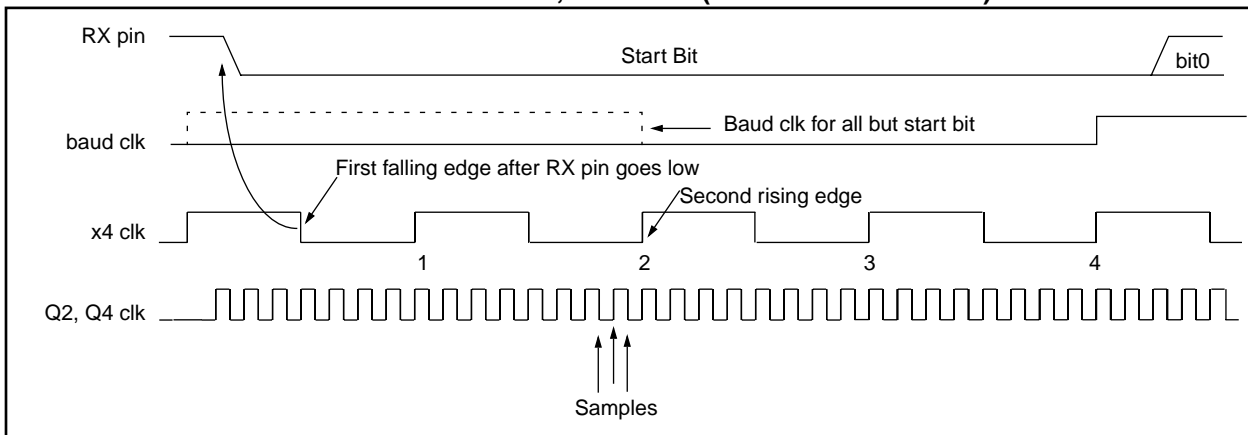


FIGURE 12-5: RX PIN SAMPLING SCHEME, BRGH = 1 (PIC16C73/73A/74/74A)



COMF		Complement f						
Syntax:	[<i>label</i>] COMF f,d							
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$							
Operation:	$(\bar{f}) \rightarrow (\text{destination})$							
Status Affected:	Z							
Encoding:	<table><tr><td>00</td><td>1001</td><td>dfff</td><td>ffff</td></tr></table>				00	1001	dfff	ffff
00	1001	dfff	ffff					
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'.							
Words:	1							
Cycles:	1							
Q Cycle Activity:	Q1	Q2	Q3	Q4				
	Decode	Read register 'f'	Process data	Write to destination				

Example

```

COMF    REG1, 0

Before Instruction
    REG1 = 0x13
After Instruction
    REG1 = 0x13
    W    = 0xEC
  
```

DECf		Decrement f						
Syntax:	[<i>label</i>] DECf f,d							
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$							
Operation:	$(f) - 1 \rightarrow (\text{destination})$							
Status Affected:	Z							
Encoding:	00		0011	dfff	ffff			
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'.							
Words:	1							
Cycles:	1							
Q Cycle Activity:	Q1		Q2		Q3		Q4	
	Decode		Read register 'f'		Process data		Write to destination	

Example

```

DECf    CNT, 1

Before Instruction
    CNT = 0x01
    Z   = 0
After Instruction
    CNT = 0x00
    Z   = 1
  
```

DECFSZ		Decrement f, Skip if 0										
Syntax:	[<i>label</i>] DECFSZ f,d											
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$											
Operation:	$(f) - 1 \rightarrow (\text{destination});$ skip if result = 0											
Status Affected:	None											
Encoding:	<table border="1"><tr><td>00</td><td>1011</td><td>dfff</td><td>ffff</td></tr></table>				00	1011	dfff	ffff				
00	1011	dfff	ffff									
Description:	<p>The contents of register 'f' are decremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.</p> <p>If the result is 1, the next instruction, is executed. If the result is 0, then a NOP is executed instead making it a 2Tcy instruction.</p>											
Words:	1											
Cycles:	1(2)											
Q Cycle Activity:	<table border="1"><tr><th>Q1</th><th>Q2</th><th>Q3</th><th>Q4</th></tr><tr><td>Decode</td><td>Read register 'f'</td><td>Process data</td><td>Write to destination</td></tr></table>				Q1	Q2	Q3	Q4	Decode	Read register 'f'	Process data	Write to destination
Q1	Q2	Q3	Q4									
Decode	Read register 'f'	Process data	Write to destination									
If Skip:	(2nd Cycle)											
	<table border="1"><tr><th>Q1</th><th>Q2</th><th>Q3</th><th>Q4</th></tr><tr><td>No-Operation</td><td>No-Operation</td><td>No-Operation</td><td>No-Operation</td></tr></table>				Q1	Q2	Q3	Q4	No-Operation	No-Operation	No-Operation	No-Operation
Q1	Q2	Q3	Q4									
No-Operation	No-Operation	No-Operation	No-Operation									

Example

```

HERE    DECFSZ    CNT, 1
        GOTO     LOOP
CONTINUE
        .
        .
        .

Before Instruction
    PC = address HERE
After Instruction
    CNT = CNT - 1
    if CNT = 0,
    PC = address CONTINUE
    if CNT ≠ 0,
    PC = address HERE+1
  
```

16.0 DEVELOPMENT SUPPORT

16.1 Development Tools

The PIC16/17 microcontrollers are supported with a full range of hardware and software development tools:

- PICMASTER/PICMASTER CE Real-Time In-Circuit Emulator
- ICEPIC Low-Cost PIC16C5X and PIC16CXXX In-Circuit Emulator
- PRO MATE® II Universal Programmer
- PICSTART® Plus Entry-Level Prototype Programmer
- PICDEM-1 Low-Cost Demonstration Board
- PICDEM-2 Low-Cost Demonstration Board
- PICDEM-3 Low-Cost Demonstration Board
- MPASM Assembler
- MPLAB-SIM Software Simulator
- MPLAB-C (C Compiler)
- Fuzzy logic development system (*fuzzyTECH*®-MP)

16.2 PICMASTER: High Performance Universal In-Circuit Emulator with MPLAB IDE

The PICMASTER Universal In-Circuit Emulator is intended to provide the product development engineer with a complete microcontroller design tool set for all microcontrollers in the PIC12C5XX, PIC14C000, PIC16C5X, PIC16CXXX and PIC17CXX families. PICMASTER is supplied with the MPLAB™ Integrated Development Environment (IDE), which allows editing, “make” and download, and source debugging from a single environment.

Interchangeable target probes allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the PICMASTER allows expansion to support all new Microchip microcontrollers.

The PICMASTER Emulator System has been designed as a real-time emulation system with advanced features that are generally found on more expensive development tools. The PC compatible 386 (and higher) machine platform and Microsoft Windows® 3.x environment were chosen to best make these features available to you, the end user.

A CE compliant version of PICMASTER is available for European Union (EU) countries.

16.3 ICEPIC: Low-cost PIC16CXXX In-Circuit Emulator

ICEPIC is a low-cost in-circuit emulator solution for the Microchip PIC16C5X and PIC16CXXX families of 8-bit OTP microcontrollers.

ICEPIC is designed to operate on PC-compatible machines ranging from 286-AT® through Pentium™ based machines under Windows 3.x environment. ICEPIC features real time, non-intrusive emulation.

16.4 PRO MATE II: Universal Programmer

The PRO MATE II Universal Programmer is a full-featured programmer capable of operating in stand-alone mode as well as PC-hosted mode.

The PRO MATE II has programmable VDD and VPP supplies which allows it to verify programmed memory at VDD min and VDD max for maximum reliability. It has an LCD display for displaying error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In stand-alone mode the PRO MATE II can read, verify or program PIC16C5X, PIC16CXXX, PIC17CXX and PIC14000 devices. It can also set configuration and code-protect bits in this mode.

16.5 PICSTART Plus Entry Level Development System

The PICSTART programmer is an easy-to-use, low-cost prototype programmer. It connects to the PC via one of the COM (RS-232) ports. MPLAB Integrated Development Environment software makes using the programmer simple and efficient. PICSTART Plus is not recommended for production programming.

PICSTART Plus supports all PIC12C5XX, PIC14000, PIC16C5X, PIC16CXXX and PIC17CXX devices with up to 40 pins. Larger pin count devices such as the PIC16C923 and PIC16C924 may be supported with an adapter socket.

PIC16C7X

Applicable Devices 72 73 73A 74 74A 76 77

18.3 DC Characteristics: PIC16C73/74-04 (Commercial, Industrial) PIC16C73/74-10 (Commercial, Industrial) PIC16C73/74-20 (Commercial, Industrial) PIC16LC73/74-04 (Commercial, Industrial)

Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial and $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial Operating voltage V_{DD} range as described in DC spec Section 18.1 and Section 18.2.							
Param No.	Characteristic	Sym	Min	Typ †	Max	Units	Conditions
D030 D030A D031 D032 D033	Input Low Voltage I/O ports with TTL buffer with Schmitt Trigger buffer MCLR, OSC1 (in RC mode) OSC1 (in XT, HS and LP)	V _{IL}	V _{SS} V _{SS} V _{SS} V _{SS} V _{SS}	- - - - -	0.15V _{DD} 0.8V 0.2V _{DD} 0.2V _{DD} 0.3V _{DD}	V V V V V	For entire V _{DD} range 4.5V ≤ V _{DD} ≤ 5.5V Note1
D040 D040A D041 D042 D042A D043	Input High Voltage I/O ports with TTL buffer with Schmitt Trigger buffer MCLR OSC1 (XT, HS and LP) OSC1 (in RC mode)	V _{IH}	2.0 0.25V _{DD} + 0.8V 0.8V _{DD} 0.8V _{DD} 0.7V _{DD} 0.9V _{DD}	- - - - - -	V _{DD} V _{DD} V _{DD} V _{DD} V _{DD} V _{DD}	V V V V V V	4.5V ≤ V _{DD} ≤ 5.5V For entire V _{DD} range For entire V _{DD} range Note1
D070	PORTB weak pull-up current	I _{PURB}	50	250	400	μA	V _{DD} = 5V, V _{PIN} = V _{SS}
D060 D061 D063	Input Leakage Current (Notes 2, 3) I/O ports MCLR, RA4/T0CKI OSC1	I _{IL}	-	-	±1 ±5 ±5	μA μA μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , Pin at hi-impedance V _{SS} ≤ V _{PIN} ≤ V _{DD} V _{SS} ≤ V _{PIN} ≤ V _{DD} , XT, HS and LP osc configuration
D080 D083	Output Low Voltage I/O ports OSC2/CLKOUT (RC osc config)	V _{OL}	-	-	0.6 0.6	V V	I _{OL} = 8.5 mA, V _{DD} = 4.5V, -40°C to +85°C I _{OL} = 1.6 mA, V _{DD} = 4.5V, -40°C to +85°C
D090 D092	Output High Voltage I/O ports (Note 3) OSC2/CLKOUT (RC osc config)	V _{OH}	V _{DD} - 0.7 V _{DD} - 0.7	-	-	V V	I _{OH} = -3.0 mA, V _{DD} = 4.5V, -40°C to +85°C I _{OH} = -1.3 mA, V _{DD} = 4.5V, -40°C to +85°C
D150*	Open-Drain High Voltage	V _{OD}	-	-	14	V	RA4 pin

* 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: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C7X be driven with external clock in RC mode.
- 2: The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.
- 3: Negative current is defined as current sourced by the pin.

PIC16C7X

Applicable Devices 72 73 73A 74 74A 76 77

19.3 DC Characteristics: **PIC16C73A/74A-04 (Commercial, Industrial, Extended)**
PIC16C73A/74A-10 (Commercial, Industrial, Extended)
PIC16C73A/74A-20 (Commercial, Industrial, Extended)
PIC16LC73A/74A-04 (Commercial, Industrial)

Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended, $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial and $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial Operating voltage VDD range as described in DC spec Section 19.1 and Section 19.2.							
Param No.	Characteristic	Sym	Min	Typ †	Max	Units	Conditions
D030 D030A D031 D032 D033	Input Low Voltage I/O ports with TTL buffer with Schmitt Trigger buffer MCLR, OSC1 (in RC mode) OSC1 (in XT, HS and LP)	V _{IL}	V _{SS} V _{SS} V _{SS} V _{SS} V _{SS}	- - - - -	0.15V _{DD} 0.8V 0.2V _{DD} 0.2V _{DD} 0.3V _{DD}	V V V V V	For entire VDD range 4.5V ≤ VDD ≤ 5.5V Note1
D040 D040A D041 D042 D042A D043	Input High Voltage I/O ports with TTL buffer with Schmitt Trigger buffer MCLR OSC1 (XT, HS and LP) OSC1 (in RC mode)	V _{IH}	2.0 0.25V _{DD} + 0.8V 0.8V _{DD} 0.8V _{DD} 0.7V _{DD} 0.9V _{DD}	- - - - - - -	V _{DD} V _{DD} V _{DD} V _{DD} V _{DD} V _{DD}	V V V V V V	4.5V ≤ VDD ≤ 5.5V For entire VDD range For entire VDD range Note1
D070	PORTB weak pull-up current	IPURB	50	250	400	μA	VDD = 5V, VPIN = VSS
D060 D061 D063	Input Leakage Current (Notes 2, 3) I/O ports MCLR, RA4/T0CKI OSC1	I _{IL}	- - -	- - -	±1 ±5 ±5	μA μA μA	VSS ≤ VPIN ≤ VDD, Pin at hi-impedance VSS ≤ VPIN ≤ VDD VSS ≤ VPIN ≤ VDD, XT, HS and LP osc configuration
D080 D080A D083 D083A	Output Low Voltage I/O ports OSC2/CLKOUT (RC osc config)	V _{OL}	- - - -	- - - -	0.6 0.6 0.6 0.6	V V V V	IOL = 8.5 mA, VDD = 4.5V, -40°C to +85°C IOL = 7.0 mA, VDD = 4.5V, -40°C to +125°C IOL = 1.6 mA, VDD = 4.5V, -40°C to +85°C IOL = 1.2 mA, VDD = 4.5V, -40°C to +125°C

* 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: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C7X be driven with external clock in RC mode.
- 2: The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.
- 3: Negative current is defined as current sourced by the pin.

FIGURE 20-3: CLKOUT AND I/O TIMING

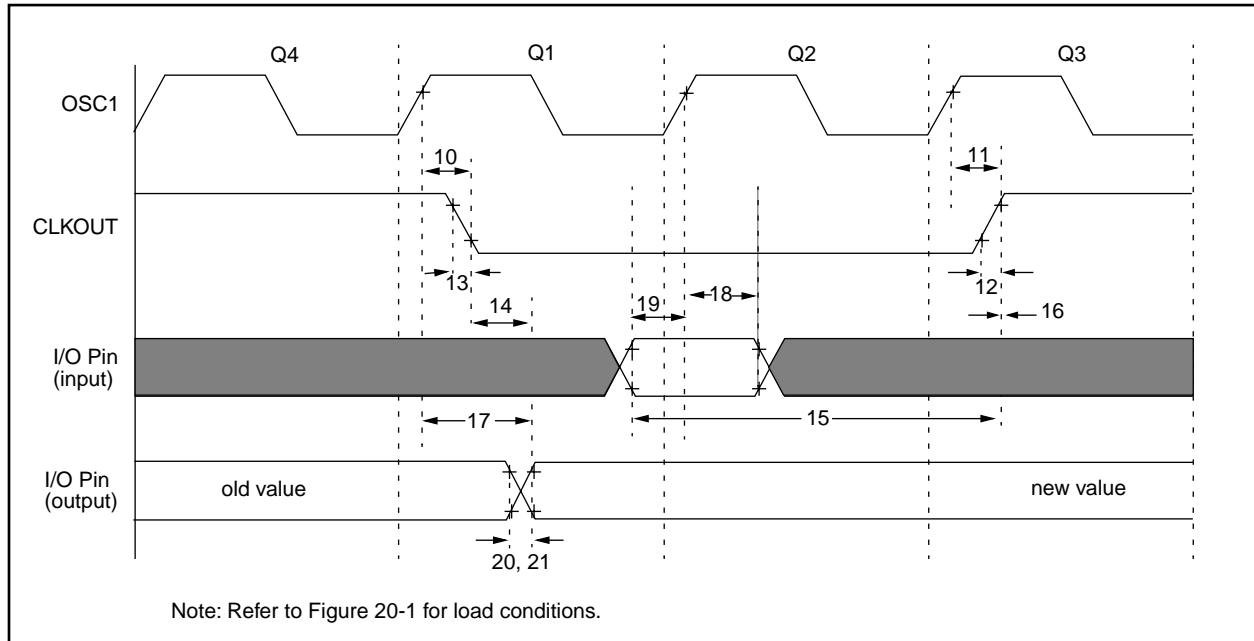


TABLE 20-3: CLKOUT AND I/O TIMING REQUIREMENTS

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
10*	TosH2ckL	OSC1↑ to CLKOUT↓	—	75	200	ns	Note 1
11*	TosH2ckH	OSC1↑ to CLKOUT↑	—	75	200	ns	Note 1
12*	TckR	CLKOUT rise time	—	35	100	ns	Note 1
13*	TckF	CLKOUT fall time	—	35	100	ns	Note 1
14*	TckL2ioV	CLKOUT ↓ to Port out valid	—	—	0.5T _{CY} + 20	ns	Note 1
15*	TioV2ckH	Port in valid before CLKOUT ↑	Tosc + 200	—	—	ns	Note 1
16*	TckH2iol	Port in hold after CLKOUT ↑	0	—	—	ns	Note 1
17*	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid	—	50	150	ns	
18*	TosH2iol	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	PIC16C76/77	100	—	—	ns
			PIC16LC76/77	200	—	—	ns
19*	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)	0	—	—	ns	
20*	TioR	Port output rise time	PIC16C76/77	—	10	40	ns
			PIC16LC76/77	—	—	80	ns
21*	TioF	Port output fall time	PIC16C76/77	—	10	40	ns
			PIC16LC76/77	—	—	80	ns
22††*	Tinp	INT pin high or low time	T _{CY}	—	—	ns	
23††*	Trbp	RB7:RB4 change INT high or low time	T _{CY}	—	—	ns	

* 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.

†† These parameters are asynchronous events not related to any internal clock edges.

Note 1: Measurements are taken in RC Mode where CLKOUT output is 4 x T_{OSC}.

PIC16C7X

Applicable Devices 72 73 73A 74 74A 76 77

TABLE 20-13: A/D CONVERTER CHARACTERISTICS:

PIC16C76/77-04 (Commercial, Industrial, Extended)
 PIC16C76/77-10 (Commercial, Industrial, Extended)
 PIC16C76/77-20 (Commercial, Industrial, Extended)
 PIC16LC76/77-04 (Commercial, Industrial)

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
A01	NR	Resolution	—	—	8-bits	bit	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A02	EABS	Total Absolute error	—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A03	EIL	Integral linearity error	—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A04	EDL	Differential linearity error	—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A05	EFS	Full scale error	—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A06	EOFF	Offset error	—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A10	—	Monotonicity	—	guaranteed	—	—	VSS ≤ VAIN ≤ 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 impedance of analog voltage source	—	—	10.0	kΩ	
A40	IAD	A/D conversion current (VDD)	PIC16C76/77	—	180	—	Average current consumption when A/D is on. (Note 1)
			PIC16LC76/77	—	90	—	
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.

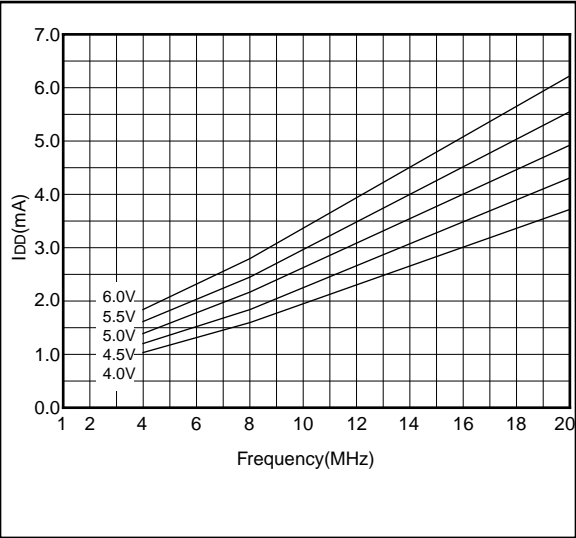
Note 1: When A/D is off, it will not consume any current other than minor leakage current. The power-down current spec includes any such leakage from the A/D module.

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

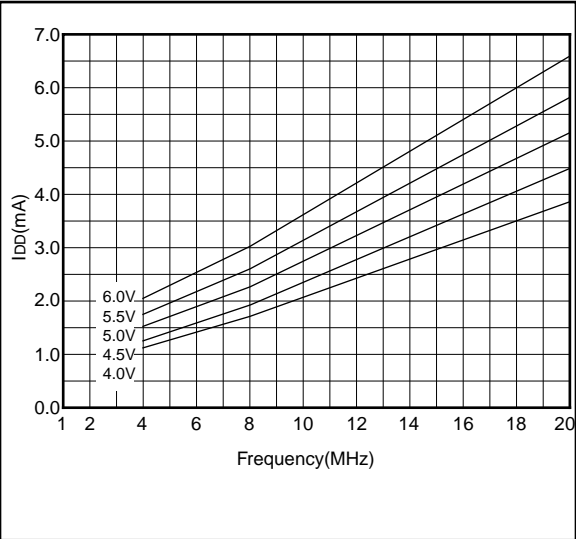
PIC16C7X

Applicable Devices	72	73	73A	74	74A	76	77
--------------------	----	----	-----	----	-----	----	----

**FIGURE 21-29: TYPICAL I_{DD} vs. FREQUENCY
(HS MODE, 25°C)**



**FIGURE 21-30: MAXIMUM I_{DD} vs.
FREQUENCY
(HS MODE, -40°C TO 85°C)**



Data based on matrix samples. See first page of this section for details.

APPENDIX A:

The following are the list of modifications over the PIC16C5X microcontroller family:

1. Instruction word length is increased to 14-bits. This allows larger page sizes both in program memory (2K now as opposed to 512 before) and register file (128 bytes now versus 32 bytes before).
2. A PC high latch register (PCLATH) is added to handle program memory paging. Bits PA2, PA1, PA0 are removed from STATUS register.
3. Data memory paging is redefined slightly. STATUS register is modified.
4. Four new instructions have been added: RETURN, RETFIE, ADDLW, and SUBLW. Two instructions TRIS and OPTION are being phased out although they are kept for compatibility with PIC16C5X.
5. OPTION and TRIS registers are made addressable.
6. Interrupt capability is added. Interrupt vector is at 0004h.
7. Stack size is increased to 8 deep.
8. Reset vector is changed to 0000h.
9. Reset of all registers is revisited. Five different reset (and wake-up) types are recognized. Registers are reset differently.
10. Wake up from SLEEP through interrupt is added.
11. Two separate timers, Oscillator Start-up Timer (OST) and Power-up Timer (PWRT) are included for more reliable power-up. These timers are invoked selectively to avoid unnecessary delays on power-up and wake-up.
12. PORTB has weak pull-ups and interrupt on change feature.
13. T0CKI pin is also a port pin (RA4) now.
14. FSR is made a full eight bit register.
15. "In-circuit serial programming" is made possible. The user can program PIC16CXX devices using only five pins: VDD, VSS, $\overline{\text{MCLR}}$ /VPP, RB6 (clock) and RB7 (data in/out).
16. PCON status register is added with a Power-on Reset status bit (POR).
17. Code protection scheme is enhanced such that portions of the program memory can be protected, while the remainder is unprotected.
18. Brown-out protection circuitry has been added. Controlled by configuration word bit BODEN. Brown-out reset ensures the device is placed in a reset condition if VDD dips below a fixed set-point.

APPENDIX B: COMPATIBILITY

To convert code written for PIC16C5X to PIC16CXX, the user should take the following steps:

1. Remove any program memory page select operations (PA2, PA1, PA0 bits) for CALL, GOTO.
2. Revisit any computed jump operations (write to PC or add to PC, etc.) to make sure page bits are set properly under the new scheme.
3. Eliminate any data memory page switching. Redefine data variables to reallocate them.
4. Verify all writes to STATUS, OPTION, and FSR registers since these have changed.
5. Change reset vector to 0000h.

APPENDIX E: PIC16/17 MICROCONTROLLERS

E.1 PIC12CXXX Family of Devices

		PIC12C508	PIC12C509	PIC12C671	PIC12C672
Clock	Maximum Frequency of Operation (MHz)	4	4	4	4
Memory	EPROM Program Memory	512 x 12	1024 x 12	1024 x 14	2048 x 14
	Data Memory (bytes)	25	41	128	128
Peripherals	Timer Module(s)	TMR0	TMR0	TMR0	TMR0
	A/D Converter (8-bit) Channels	—	—	4	4
Features	Wake-up from SLEEP on pin change	Yes	Yes	Yes	Yes
	I/O Pins	5	5	5	5
	Input Pins	1	1	1	1
	Internal Pull-ups	Yes	Yes	Yes	Yes
	Voltage Range (Volts)	2.5-5.5	2.5-5.5	2.5-5.5	2.5-5.5
	In-Circuit Serial Programming	Yes	Yes	Yes	Yes
	Number of Instructions	33	33	35	35
	Packages	8-pin DIP, SOIC	8-pin DIP, SOIC	8-pin DIP, SOIC	8-pin DIP, SOIC

All PIC12C5XX devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability.
All PIC12C5XX devices use serial programming with data pin GP1 and clock pin GP0.

E.2 PIC14C000 Family of Devices

		PIC14C000
Clock	Maximum Frequency of Operation (MHz)	20
Memory	EPROM Program Memory (x14 words)	4K
	Data Memory (bytes)	192
	Timer Module(s)	TMR0 ADTMR
Peripherals	Serial Port(s) (SPI/I ² C, USART)	I ² C with SMBus Support
Features	Slope A/D Converter Channels	8 External; 6 Internal
	Interrupt Sources	11
	I/O Pins	22
	Voltage Range (Volts)	2.7-6.0
	In-Circuit Serial Programming	Yes
	Additional On-chip Features	Internal 4MHz Oscillator, Bandgap Reference, Temperature Sensor, Calibration Factors, Low Voltage Detector, SLEEP, HIBERNATE, Comparators with Programmable References (2)
	Packages	28-pin DIP (.300 mil), SOIC, SSOP

PIC16C7X

Figure 20-9:	SPI Master Mode Timing (CKE = 0).....	232
Figure 20-10:	SPI Master Mode Timing (CKE = 1).....	232
Figure 20-11:	SPI Slave Mode Timing (CKE = 0).....	233
Figure 20-12:	SPI Slave Mode Timing (CKE = 1).....	233
Figure 20-13:	I ² C Bus Start/Stop Bits Timing.....	235
Figure 20-14:	I ² C Bus Data Timing.....	236
Figure 20-15:	USART Synchronous Transmission (Master/Slave) Timing.....	237
Figure 20-16:	USART Synchronous Receive (Master/Slave) Timing.....	237
Figure 20-17:	A/D Conversion Timing.....	239
Figure 21-1:	Typical IPD vs. VDD (WDT Disabled, RC Mode).....	241
Figure 21-2:	Maximum IPD vs. VDD (WDT Disabled, RC Mode).....	241
Figure 21-3:	Typical IPD vs. VDD @ 25°C (WDT Enabled, RC Mode).....	242
Figure 21-4:	Maximum IPD vs. VDD (WDT Enabled, RC Mode).....	242
Figure 21-5:	Typical RC Oscillator Frequency vs. VDD.....	242
Figure 21-6:	Typical RC Oscillator Frequency vs. VDD.....	242
Figure 21-7:	Typical RC Oscillator Frequency vs. VDD.....	242
Figure 21-8:	Typical IPD vs. VDD Brown-out Detect Enabled (RC Mode).....	243
Figure 21-9:	Maximum IPD vs. VDD Brown-out Detect Enabled (85°C to -40°C, RC Mode).....	243
Figure 21-10:	Typical IPD vs. Timer1 Enabled (32 kHz, RC0/RC1= 33 pF/33 pF, RC Mode).....	243
Figure 21-11:	Maximum IPD vs. Timer1 Enabled (32 kHz, RC0/RC1 = 33 pF/33 pF, 85°C to -40°C, RC Mode).....	243
Figure 21-12:	Typical IDD vs. Frequency (RC Mode @ 22 pF, 25°C).....	244
Figure 21-13:	Maximum IDD vs. Frequency (RC Mode @ 22 pF, -40°C to 85°C).....	244
Figure 21-14:	Typical IDD vs. Frequency (RC Mode @ 100 pF, 25°C).....	245
Figure 21-15:	Maximum IDD vs. Frequency (RC Mode @ 100 pF, -40°C to 85°C).....	245
Figure 21-16:	Typical IDD vs. Frequency (RC Mode @ 300 pF, 25°C).....	246
Figure 21-17:	Maximum IDD vs. Frequency (RC Mode @ 300 pF, -40°C to 85°C).....	246
Figure 21-18:	Typical IDD vs. Capacitance @ 500 kHz (RC Mode).....	247
Figure 21-19:	Transconductance(gm) of HS Oscillator vs. VDD.....	247
Figure 21-20:	Transconductance(gm) of LP Oscillator vs. VDD.....	247
Figure 21-21:	Transconductance(gm) of XT Oscillator vs. VDD.....	247
Figure 21-22:	Typical XTAL Startup Time vs. VDD (LP Mode, 25°C).....	248
Figure 21-23:	Typical XTAL Startup Time vs. VDD (HS Mode, 25°C).....	248
Figure 21-24:	Typical XTAL Startup Time vs. VDD (XT Mode, 25°C).....	248
Figure 21-25:	Typical Idd vs. Frequency (LP Mode, 25°C).....	249
Figure 21-26:	Maximum IDD vs. Frequency (LP Mode, 85°C to -40°C).....	249
Figure 21-27:	Typical IDD vs. Frequency (XT Mode, 25°C).....	249
Figure 21-28:	Maximum IDD vs. Frequency (XT Mode, -40°C to 85°C).....	249
Figure 21-29:	Typical IDD vs. Frequency (HS Mode, 25°C).....	250
Figure 21-30:	Maximum IDD vs. Frequency (HS Mode, -40°C to 85°C).....	250