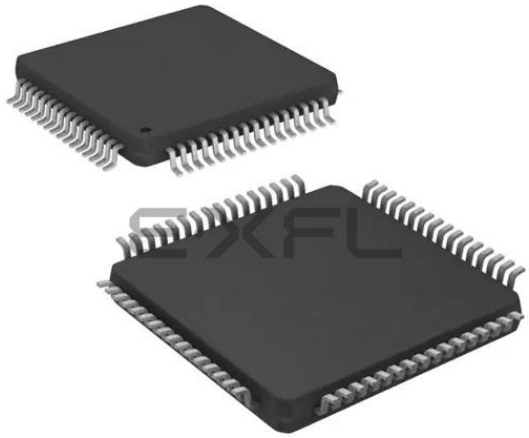


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	33MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	50
Program Memory Size	32KB (16K x 16)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	902 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 12x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic17c756a-33i-pt

TABLE 3-1: PINOUT DESCRIPTIONS (CONTINUED)

Name	PIC17C75X			PIC17C76X		I/O/P Type	Buffer Type	Description
	DIP No.	PLCC No.	TQFP No.	PLCC No.	QFP No.			
								PORTC is a bi-directional I/O Port.
RC0/AD0	2	3	58	3	72	I/O	TTL	This is also the least significant byte (LSB) of the 16-bit wide system bus in Microprocessor mode or Extended Microcontroller mode. In multiplexed system bus configuration, these pins are address output as well as data input or output.
RC1/AD1	63	67	55	83	69	I/O	TTL	
RC2/AD2	62	66	54	82	68	I/O	TTL	
RC3/AD3	61	65	53	81	67	I/O	TTL	
RC4/AD4	60	64	52	80	66	I/O	TTL	
RC5/AD5	58	63	51	79	65	I/O	TTL	
RC6/AD6	58	62	50	78	64	I/O	TTL	
RC7/AD7	57	61	49	77	63	I/O	TTL	
								PORTD is a bi-directional I/O Port.
RD0/AD8	10	11	2	15	4	I/O	TTL	This is also the most significant byte (MSB) of the 16-bit system bus in Microprocessor mode or Extended Microcontroller mode. In multiplexed system bus configuration, these pins are address output as well as data input or output.
RD1/AD9	9	10	1	14	3	I/O	TTL	
RD2/AD10	8	9	64	9	78	I/O	TTL	
RD3/AD11	7	8	63	8	77	I/O	TTL	
RD4/AD12	6	7	62	7	76	I/O	TTL	
RD5/AD13	5	6	61	6	75	I/O	TTL	
RD6/AD14	4	5	60	5	74	I/O	TTL	
RD7/AD15	3	4	59	4	73	I/O	TTL	
								PORTE is a bi-directional I/O Port.
RE0/ALE	11	12	3	16	5	I/O	TTL	In Microprocessor mode or Extended Microcontroller mode, RE0 is the Address Latch Enable (ALE) output. Address should be latched on the falling edge of ALE output.
RE1/$\overline{OE}$	12	13	4	17	6	I/O	TTL	In Microprocessor or Extended Microcontroller mode, RE1 is the Output Enable ($\overline{OE}$) control output (active low).
RE2/$\overline{WR}$	13	14	5	18	7	I/O	TTL	In Microprocessor or Extended Microcontroller mode, RE2 is the Write Enable ($\overline{WR}$) control output (active low).
RE3/CAP4	14	15	6	19	8	I/O	ST	RE3 can also be the Capture4 input pin.
								PORTF is a bi-directional I/O Port.
RF0/AN4	26	28	18	36	24	I/O	ST	RF0 can also be analog input 4.
RF1/AN5	25	27	17	35	23	I/O	ST	RF1 can also be analog input 5.
RF2/AN6	24	26	16	30	18	I/O	ST	RF2 can also be analog input 6.
RF3/AN7	23	25	15	29	17	I/O	ST	RF3 can also be analog input 7.
RF4/AN8	22	24	14	28	16	I/O	ST	RF4 can also be analog input 8.
RF5/AN9	21	23	13	27	15	I/O	ST	RF5 can also be analog input 9.
RF6/AN10	20	22	12	26	14	I/O	ST	RF6 can also be analog input 10.
RF7/AN11	19	21	11	25	13	I/O	ST	RF7 can also be analog input 11.

Legend: I = Input only; O = Output only; I/O = Input/Output;
P = Power; — = Not Used; TTL = TTL input; ST = Schmitt Trigger input

Note 1: The output is only available by the peripheral operation.

2: Open drain input/output pin. Pin forced to input upon any device RESET.

NOTES:

6.2 Peripheral Interrupt Enable Register1 (PIE1) and Register2 (PIE2)

These registers contains the individual enable bits for the peripheral interrupts.

REGISTER 6-2: PIE1 REGISTER (ADDRESS: 17h, BANK 1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RBIE	TMR3IE	TMR2IE	TMR1IE	CA2IE	CA1IE	TX1IE	RC1IE
bit 7							bit 0

bit 7 **RBIE**: PORTB Interrupt-on-Change Enable bit
1 = Enable PORTB interrupt-on-change
0 = Disable PORTB interrupt-on-change

bit 6 **TMR3IE**: TMR3 Interrupt Enable bit
1 = Enable TMR3 interrupt
0 = Disable TMR3 interrupt

bit 5 **TMR2IE**: TMR2 Interrupt Enable bit
1 = Enable TMR2 interrupt
0 = Disable TMR2 interrupt

bit 4 **TMR1IE**: TMR1 Interrupt Enable bit
1 = Enable TMR1 interrupt
0 = Disable TMR1 interrupt

bit 3 **CA2IE**: Capture2 Interrupt Enable bit
1 = Enable Capture2 interrupt
0 = Disable Capture2 interrupt

bit 2 **CA1IE**: Capture1 Interrupt Enable bit
1 = Enable Capture1 interrupt
0 = Disable Capture1 interrupt

bit 1 **TX1IE**: USART1 Transmit Interrupt Enable bit
1 = Enable USART1 Transmit buffer empty interrupt
0 = Disable USART1 Transmit buffer empty interrupt

bit 0 **RC1IE**: USART1 Receive Interrupt Enable bit
1 = Enable USART1 Receive buffer full interrupt
0 = Disable USART1 Receive buffer full interrupt

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
- n = Value at POR Reset	’1’ = Bit is set	’0’ = Bit is cleared	x = Bit is unknown

REGISTER 6-5: PIR2 REGISTER (ADDRESS: 10h, BANK 4)

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R-1	R-0
SSPIF	BCLIF	ADIF	—	CA4IF	CA3IF	TX2IF	RC2IF
bit 7							bit 0

bit 7 **SSPIF**: Synchronous Serial Port (SSP) Interrupt Flag bit
1 = The SSP interrupt condition has occurred and must be cleared in software before returning from the Interrupt Service Routine. The conditions that will set this bit are:

SPI:
A transmission/reception has taken place.

I^2C Slave/Master:
A transmission/reception has taken place.

I^2C Master:
The initiated START condition was completed by the SSP module.
The initiated STOP condition was completed by the SSP module.
The initiated Restart condition was completed by the SSP module.
The initiated Acknowledge condition was completed by the SSP module.
A START condition occurred while the SSP module was idle (Multi-master system).
A STOP condition occurred while the SSP module was idle (Multi-master system).

0 = An SSP interrupt condition has NOT occurred

bit 6 **BCLIF**: Bus Collision Interrupt Flag bit
1 = A bus collision has occurred in the SSP, when configured for I^2C Master mode
0 = No bus collision has occurred

bit 5 **ADIF**: A/D Module Interrupt Flag bit
1 = An A/D conversion is complete
0 = An A/D conversion is not complete

bit 4 **Unimplemented**: Read as '0'

bit 3 **CA4IF**: Capture4 Interrupt Flag bit
1 = Capture event occurred on RE3/CAP4 pin
0 = Capture event did not occur on RE3/CAP4 pin

bit 2 **CA3IF**: Capture3 Interrupt Flag bit
1 = Capture event occurred on RG4/CAP3 pin
0 = Capture event did not occur on RG4/CAP3 pin

bit 1 **TX2IF**:USART2 Transmit Interrupt Flag bit (state controlled by hardware)
1 = USART2 Transmit buffer is empty
0 = USART2 Transmit buffer is full

bit 0 **RC2IF**: USART2 Receive Interrupt Flag bit (state controlled by hardware)
1 = USART2 Receive buffer is full
0 = USART2 Receive buffer is empty

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
- n = Value at POR Reset	’1’ = Bit is set	’0’ = Bit is cleared	x = Bit is unknown

TABLE 7-3: SPECIAL FUNCTION REGISTERS (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	$\overline{\text{MCLR}}$, WDT
Bank 2											
10h	TMR1	Timer1's Register								xxxx xxxx	uuuu uuuu
11h	TMR2	Timer2's Register								xxxx xxxx	uuuu uuuu
12h	TMR3L	Timer3's Register; Low Byte								xxxx xxxx	uuuu uuuu
13h	TMR3H	Timer3's Register; High Byte								xxxx xxxx	uuuu uuuu
14h	PR1	Timer1's Period Register								xxxx xxxx	uuuu uuuu
15h	PR2	Timer2's Period Register								xxxx xxxx	uuuu uuuu
16h	PR3L/CA1L	Timer3's Period Register - Low Byte/Capture1 Register; Low Byte								xxxx xxxx	uuuu uuuu
17h	PR3H/CA1H	Timer3's Period Register - High Byte/Capture1 Register; High Byte								xxxx xxxx	uuuu uuuu
Bank 3											
10h	PW1DCL	DC1	DC0	—	—	—	—	—	—	xx-- ----	uu-- ----
11h	PW2DCL	DC1	DC0	TM2PW2	—	—	—	—	—	xx0- ----	uu0- ----
12h	PW1DCH	DC9	DC8	DC7	DC6	DC5	DC4	DC3	DC2	xxxx xxxx	uuuu uuuu
13h	PW2DCH	DC9	DC8	DC7	DC6	DC5	DC4	DC3	DC2	xxxx xxxx	uuuu uuuu
14h	CA2L	Capture2 Low Byte								xxxx xxxx	uuuu uuuu
15h	CA2H	Capture2 High Byte								xxxx xxxx	uuuu uuuu
16h	TCON1	CA2ED1	CA2ED0	CA1ED1	CA1ED0	T16	TMR3CS	TMR2CS	TMR1CS	0000 0000	0000 0000
17h	TCON2	CA2OVF	CA1OVF	PWM2ON	PWM1ON	CA1/$\overline{\text{PR3}}$	TMR3ON	TMR2ON	TMR1ON	0000 0000	0000 0000
Bank 4											
10h	PIR2	SSPIF	BCLIF	ADIF	—	CA4IF	CA3IF	TX2IF	RC2IF	000- 0010	000- 0010
11h	PIE2	SSPIE	BCLIE	ADIE	—	CA4IE	CA3IE	TX2IE	RC2IE	000- 0000	000- 0000
12h	Unimplemented	—	—	—	—	—	—	—	—	---- ----	---- ----
13h	RCSTA2	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00u
14h	RCREG2	Serial Port Receive Register for USART2								xxxx xxxx	uuuu uuuu
15h	TXSTA2	CSRC	TX9	TXEN	SYNC	—	—	TRMT	TX9D	0000 --1x	0000 --1u
16h	TXREG2	Serial Port Transmit Register for USART2								xxxx xxxx	uuuu uuuu
17h	SPBRG2	Baud Rate Generator for USART2								0000 0000	0000 0000
Bank 5:											
10h	DDRF	Data Direction Register for PORTF								1111 1111	1111 1111
11h	PORTF(4)	RF7/ AN11	RF6/ AN10	RF5/ AN9	RF4/ AN8	RF3/ AN7	RF2/ AN6	RF1/ AN5	RF0/ AN4	0000 0000	0000 0000
12h	DDRG	Data Direction Register for PORTG								1111 1111	1111 1111
13h	PORTG(4)	RG7/ TX2/CK2	RG6/ RX2/DT2	RG5/ PWM3	RG4/ CAP3	RG3/ AN0	RG2/ AN1	RG1/ AN2	RG0/ AN3	xxxx 0000	uuuu 0000
14h	ADCON0	CHS3	CHS2	CHS1	CHS0	—	GO/$\overline{\text{DONE}}$	—	ADON	0000 -0-0	0000 -0-0
15h	ADCON1	ADCS1	ADCS0	ADFM	—	PCFG3	PCFG2	PCFG1	PCFG0	000- 0000	000- 0000
16h	ADRESL	A/D Result Register Low Byte								xxxx xxxx	uuuu uuuu
17h	ADRESH	A/D Result Register High Byte								xxxx xxxx	uuuu uuuu

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

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for PC<15:8> whose contents are updated from, or transferred to, the upper byte of the program counter.

2: The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ status bits in CPUSTA are not affected by a $\overline{\text{MCLR}}$ Reset.

3: Bank 8 and associated registers are only implemented on the PIC17C76X devices.

4: This is the value that will be in the port output latch.

5: When the device is configured for Microprocessor or Extended Microcontroller mode, the operation of this port does not rely on these registers.

6: On any device RESET, these pins are configured as inputs.

8.2 Table Writes to External Memory

Table writes to external memory are always two-cycle instructions. The second cycle writes the data to the external memory location. The sequence of events for an external memory write are the same for an internal write.

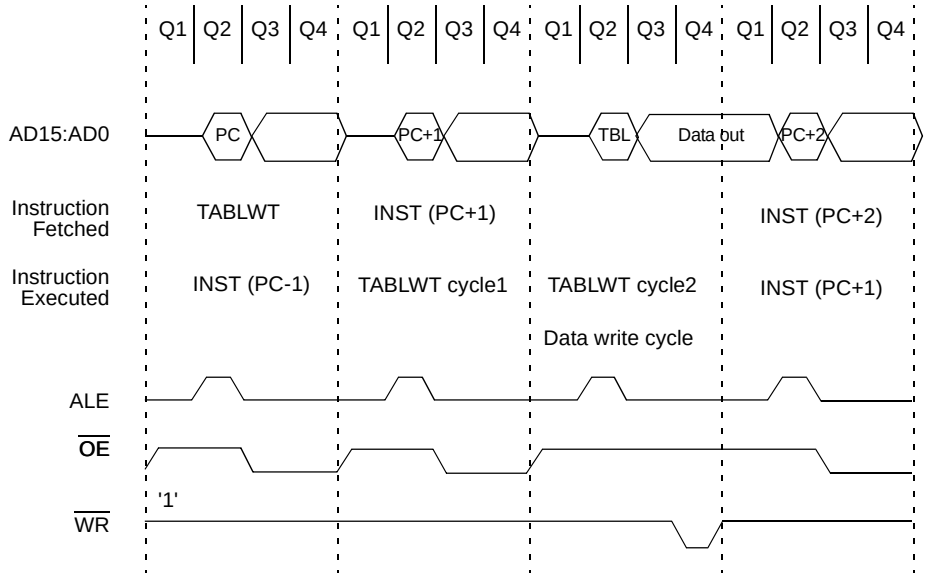
8.2.1 TABLE WRITE CODE

The "i" operand of the TABLWT instruction can specify that the value in the 16-bit TBLPTR register is automatically incremented (for the next write). In Example 8-1, the TBLPTR register is not automatically incremented.

EXAMPLE 8-1: TABLE WRITE

```
CLRWDT                  ; Clear WDT
MOVLW   HIGH (TBL_ADDR) ; Load the Table
MOVWF   TBLPTRH         ;   address
MOVLW   LOW (TBL_ADDR)  ;
MOVWF   TBLPTRL         ;
MOVLW   HIGH (DATA)     ; Load HI byte
TLWT    1, WREG         ;   in TABLATH
MOVLW   LOW (DATA)      ; Load LO byte
TABLWT  0,0,WREG        ;   in TABLATL
                        ;   and write to
                        ;   program memory
                        ;   (Ext. SRAM)
```

FIGURE 8-5: TABLWT WRITE TIMING (EXTERNAL MEMORY)



Note: If external write and GLINTD = '1' and Enable bit = '1', then when '1' → Flag bit, do table write. The highest pending interrupt is cleared.

NOTES:

REGISTER 13-2: TCON2 REGISTER (ADDRESS: 17h, BANK 3)

R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CA2OVF	CA1OVF	PWM2ON	PWM1ON	CA1/$\overline{PR3}$	TMR3ON	TMR2ON	TMR1ON
bit 7							bit 0

bit 7 **CA2OVF**: Capture2 Overflow Status bit
This bit indicates that the capture value had not been read from the capture register pair (CA2H:CA2L) before the next capture event occurred. The capture register retains the oldest unread capture value (last capture before overflow). Subsequent capture events will not update the capture register with the TMR3 value until the capture register has been read (both bytes).
1 = Overflow occurred on Capture2 register
0 = No overflow occurred on Capture2 register

bit 6 **CA1OVF**: Capture1 Overflow Status bit
This bit indicates that the capture value had not been read from the capture register pair (PR3H/CA1H:PR3L/CA1L), before the next capture event occurred. The capture register retains the oldest unread capture value (last capture before overflow). Subsequent capture events will not update the capture register with the TMR3 value until the capture register has been read (both bytes).
1 = Overflow occurred on Capture1 register
0 = No overflow occurred on Capture1 register

bit 5 **PWM2ON**: PWM2 On bit
1 = PWM2 is enabled
(The RB3/PWM2 pin ignores the state of the DDRB<3> bit.)
0 = PWM2 is disabled
(The RB3/PWM2 pin uses the state of the DDRB<3> bit for data direction.)

bit 4 **PWM1ON**: PWM1 On bit
1 = PWM1 is enabled
(The RB2/PWM1 pin ignores the state of the DDRB<2> bit.)
0 = PWM1 is disabled
(The RB2/PWM1 pin uses the state of the DDRB<2> bit for data direction.)

bit 3 **CA1/$\overline{PR3}$**: CA1/$\overline{PR3}$ Register Mode Select bit
1 = Enables Capture1
(PR3H/CA1H:PR3L/CA1L is the Capture1 register. Timer3 runs without a period register.)
0 = Enables the Period register
(PR3H/CA1H:PR3L/CA1L is the Period register for Timer3.)

bit 2 **TMR3ON**: Timer3 On bit
1 = Starts Timer3
0 = Stops Timer3

bit 1 **TMR2ON**: Timer2 On bit
This bit controls the incrementing of the TMR2 register. When TMR2:TMR1 form the 16-bit timer (T16 is set), TMR2ON must be set. This allows the MSB of the timer to increment.
1 = Starts Timer2 (must be enabled if the T16 bit (TCON1<3>) is set)
0 = Stops Timer2

bit 0 **TMR1ON**: Timer1 On bit

When T16 is set (in 16-bit Timer mode):
1 = Starts 16-bit TMR2:TMR1
0 = Stops 16-bit TMR2:TMR1

When T16 is clear (in 8-bit Timer mode:
1 = Starts 8-bit Timer1
0 = Stops 8-bit Timer1

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
- n = Value at POR Reset	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

13.2.3 READING THE CAPTURE REGISTERS

The Capture overflow status flag bits are double buffered. The master bit is set if one captured word is already residing in the Capture register and another "event" has occurred on the CAPx pin. The new event will not transfer the TMR3 value to the capture register, protecting the previous unread capture value. When the user reads both the high and the low bytes (in any order) of the Capture register, the master overflow bit is transferred to the slave overflow bit (CAxOVF) and then the master bit is reset. The user can then read TCONx to determine the value of CAxOVF.

An example of an instruction sequence to read capture registers and capture overflow flag bits is shown in Example 13-1. Depending on the capture source, different registers will need to be read.

EXAMPLE 13-1: SEQUENCE TO READ CAPTURE REGISTERS

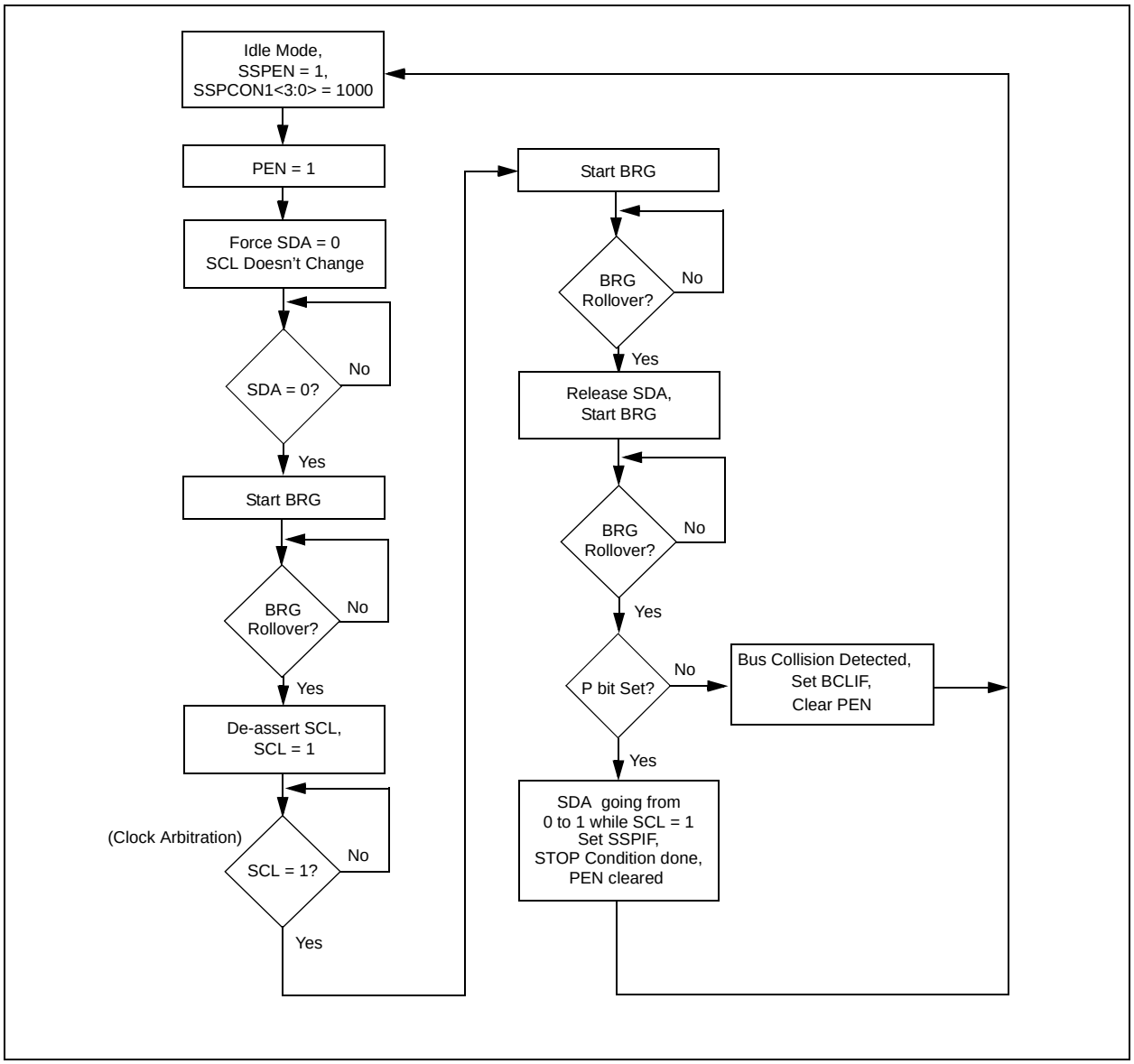
```
MOVLB 3                  ; Select Bank 3
MOVPF CA2L, LO_BYTE      ; Read Capture2 low byte, store in LO_BYTE
MOVPF CA2H, HI_BYTE      ; Read Capture2 high byte, store in HI_BYTE
MOVPF TCON2, STAT_VAL    ; Read TCON2 into file STAT_VAL
```

TABLE 13-6: REGISTERS ASSOCIATED WITH CAPTURE

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	$\overline{MCLR}$, WDT
16h, Bank 3	TCON1	CA2ED1	CA2ED0	CA1ED1	CA1ED0	T16	TMR3CS	TMR2CS	TMR1CS	0000 0000	0000 0000
17h, Bank 3	TCON2	CA2OVF	CA1OVF	PWM2ON	PWM1ON	CA1/$\overline{PR3}$	TMR3ON	TMR2ON	TMR1ON	0000 0000	0000 0000
16h, Bank 7	TCON3	—	CA4OVF	CA3OVF	CA4ED1	CA4ED0	CA3ED1	CA3ED0	PWM3ON	-000 0000	-000 0000
12h, Bank 2	TMR3L	Holding Register for the Low Byte of the 16-bit TMR3 Register								xxxx xxxx	uuuu uuuu
13h, Bank 2	TMR3H	Holding Register for the High Byte of the 16-bit TMR3 Register								xxxx xxxx	uuuu uuuu
16h, Bank 1	PIR1	RBIF	TMR3IF	TMR2IF	TMR1IF	CA2IF	CA1IF	TX1IF	RC1IF	x000 0010	u000 0010
17h, Bank 1	PIE1	RBIE	TMR3IE	TMR2IE	TMR1IE	CA2IE	CA1IE	TX1IE	RC1IE	0000 0000	0000 0000
10h, Bank 4	PIR2	SSPIF	BCLIF	ADIF	—	CA4IF	CA3IF	TX2IF	RC2IF	000- 0010	000- 0010
11h, Bank 4	PIE2	SSPIE	BCLIE	ADIE	—	CA4IE	CA3IE	TX2IE	RC2IE	000- 0000	000- 0000
07h, Unbanked	INTSTA	PEIF	T0CKIF	T0IF	INTF	PEIE	T0CKIE	T0IE	INTE	0000 0000	0000 0000
06h, Unbanked	CPUSTA	—	—	STKAV	GLINTD	$\overline{TO}$	$\overline{PD}$	$\overline{POR}$	$\overline{BOR}$	--11 11qq	--11 qquu
16h, Bank 2	PR3L/CA1L	Timer3 Period Register, Low Byte/Capture1 Register, Low Byte								xxxx xxxx	uuuu uuuu
17h, Bank 2	PR3H/CA1H	Timer3 Period Register, High Byte/Capture1 Register, High Byte								xxxx xxxx	uuuu uuuu
14h, Bank 3	CA2L	Capture2 Low Byte								xxxx xxxx	uuuu uuuu
15h, Bank 3	CA2H	Capture2 High Byte								xxxx xxxx	uuuu uuuu
12h, Bank 7	CA3L	Capture3 Low Byte								xxxx xxxx	uuuu uuuu
13h, Bank 7	CA3H	Capture3 High Byte								xxxx xxxx	uuuu uuuu
14h, Bank 7	CA4L	Capture4 Low Byte								xxxx xxxx	uuuu uuuu
15h, Bank 7	CA4H	Capture4 High Byte								xxxx xxxx	uuuu uuuu

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0', q = value depends on condition.
Shaded cells are not used by Capture.

FIGURE 15-32: STOP CONDITION FLOW CHART



15.2.18.1 Bus Collision During a START Condition

During a START condition, a bus collision occurs if:

a) SDA or SCL are sampled low at the beginning of the START condition (Figure 15-35).

b) SCL is sampled low before SDA is asserted low (Figure 15-36).

During a START condition, both the SDA and the SCL pins are monitored.

If:

the SDA pin is already low
or the SCL pin is already low,

then:

the START condition is aborted,
and the BCLIF flag is set,
and the SSP module is reset to its IDLE state (Figure 15-35).

The START condition begins with the SDA and SCL pins de-asserted. When the SDA pin is sampled high, the baud rate generator is loaded from SSPADD<6:0> and counts down to '0'. If the SCL pin is sampled low while SDA is high, a bus collision occurs, because it is assumed that another master is attempting to drive a data '1' during the START condition.

If the SDA pin is sampled low during this count, the BRG is reset and the SDA line is asserted early (Figure 15-37). If, however, a '1' is sampled on the SDA pin, the SDA pin is asserted low at the end of the BRG count. The baud rate generator is then reloaded and counts down to 0 and during this time, if the SCL pin is sampled as '0', a bus collision does not occur. At the end of the BRG count, the SCL pin is asserted low.

Note: The reason that bus collision is not a factor during a START condition is that no two bus masters can assert a START condition at the exact same time. Therefore, one master will always assert SDA before the other. This condition does not cause a bus collision because the two masters must be allowed to arbitrate the first address following the START condition and if the address is the same, arbitration must be allowed to continue into the data portion, Repeated Start, or Stop conditions.

FIGURE 15-35: BUS COLLISION DURING START CONDITION (SDA ONLY)

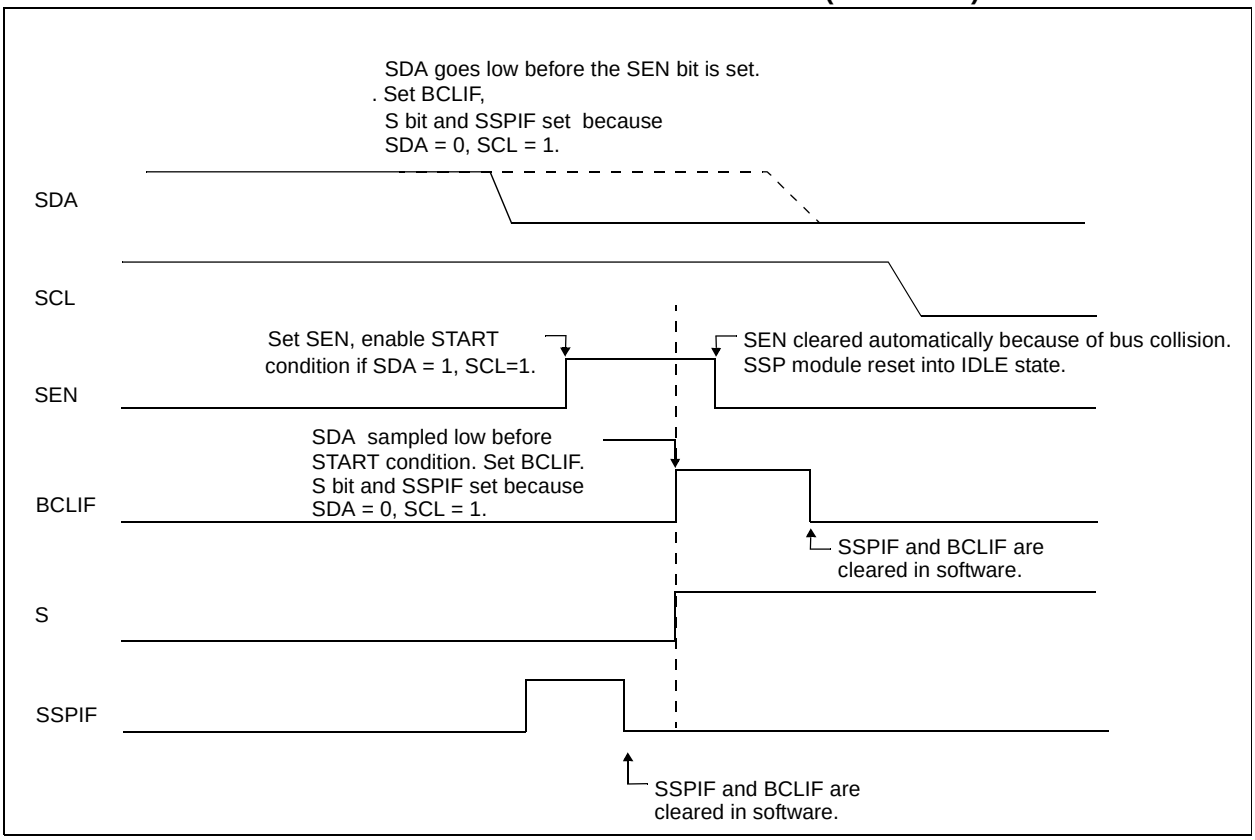
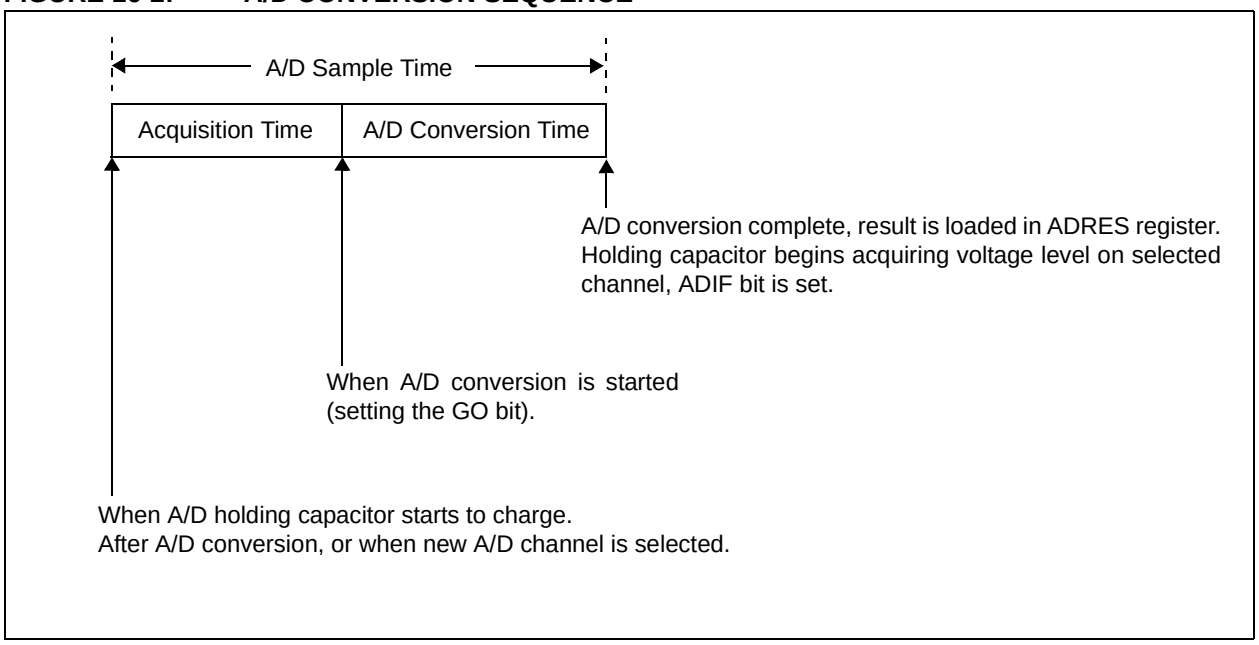


Figure 16-2 shows the conversion sequence and the terms that are used. Acquisition time is the time that the A/D module's holding capacitor is connected to the external voltage level. Then, there is the conversion time of 12 TAD, which is started when the GO bit is set. The sum of these two times is the sampling time. There is a minimum acquisition time to ensure that the holding capacitor is charged to a level that will give the desired accuracy for the A/D conversion.

FIGURE 16-2: A/D CONVERSION SEQUENCE



17.3 Watchdog Timer (WDT)

The Watchdog Timer's function is to recover from software malfunction, or to reset the device while in SLEEP mode. The WDT uses an internal free running on-chip RC oscillator for its clock source. This does not require any external components. This RC oscillator is separate from the RC oscillator of the OSC1/CLKIN pin. That means that the WDT will run even if the clock on the OSC1/CLKIN and OSC2/CLKOUT pins has been stopped, for example, by execution of a SLEEP instruction. During normal operation, a WDT time-out generates a device RESET. The WDT can be permanently disabled by programming the configuration bits WDTPS1:WDTPS0 as '00' (Section 17.1).

Under normal operation, the WDT must be cleared on a regular interval. This time must be less than the minimum WDT overflow time. Not clearing the WDT in this time frame will cause the WDT to overflow and reset the device.

17.3.1 WDT PERIOD

The WDT has a nominal time-out period of 12 ms (with postscaler = 1). The time-out periods vary with temperature, VDD and process variations from part to part (see DC specs). If longer time-out periods are desired, configuration bits should be used to enable the WDT with a greater prescale. Thus, typical time-out periods up to 3.0 seconds can be realized.

The CLRWDT and SLEEP instructions clear the WDT and its postscale setting and prevent it from timing out, thus generating a device RESET condition.

The $\overline{TO}$ bit in the CPUSTA register will be cleared upon a WDT time-out.

17.3.2 CLEARING THE WDT AND POSTSCALER

The WDT and postscaler are cleared when:

- The device is in the RESET state
- A SLEEP instruction is executed
- A CLRWDT instruction is executed
- Wake-up from SLEEP by an interrupt

The WDT counter/postscaler will start counting on the first edge after the device exits the RESET state.

17.3.3 WDT PROGRAMMING CONSIDERATIONS

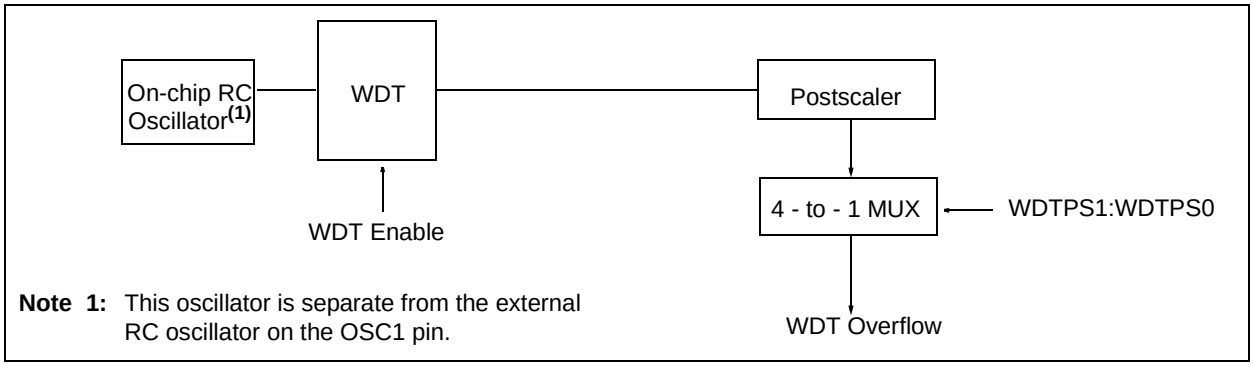
It should also be taken in account that under worst case conditions (VDD = Min., Temperature = Max., Max. WDT postscaler), it may take several seconds before a WDT time-out occurs.

The WDT and postscaler become the Power-up Timer whenever the PWRT is invoked.

17.3.4 WDT AS NORMAL TIMER

When the WDT is selected as a normal timer, the clock source is the device clock. Neither the WDT nor the postscaler are directly readable or writable. The overflow time is 65536 TOSC cycles. On overflow, the $\overline{TO}$ bit is cleared (device is not RESET). The CLRWDT instruction can be used to set the $\overline{TO}$ bit. This allows the WDT to be a simple overflow timer. The simple timer does not increment when in SLEEP.

FIGURE 17-1: WATCHDOG TIMER BLOCK DIAGRAM



Note 1: This oscillator is separate from the external RC oscillator on the OSC1 pin.

TABLE 17-2: REGISTERS/BITS ASSOCIATED WITH THE WATCHDOG TIMER

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	$\overline{MCLR}$, WDT
—	Config	See Figure 17-1 for location of WDTPSx bits in Configuration Word.								(Note 1)	(Note 1)
06h, Unbanked	CPUSTA	—	—	STKAV	GLINTD	$\overline{TO}$	$\overline{PD}$	$\overline{POR}$	$\overline{BOR}$	--11 11qq	--11 qquu

Legend: - = unimplemented, read as '0', q = value depends on condition. Shaded cells are not used by the WDT.
Note 1: This value will be as the device was programmed, or if unprogrammed, will read as all '1's.

MOVPF **Move p to f**

Syntax: [*label*] MOVPF p,f

Operands: $0 \le f \le 255$
$0 \le p \le 31$

Operation: (p) → (f)

Status Affected: Z

Encoding:

010p	pppp	ffff	ffff

Description: Move data from data memory location 'p' to data memory location 'f'. Location 'f' can be anywhere in the 256 byte data space (00h to FFh), while 'p' can be 00h to 1Fh.

Either 'p' or 'f' can be WREG (a useful, special situation).

MOVPF is particularly useful for transferring a peripheral register (e.g. the timer or an I/O port) to a data memory location. Both 'f' and 'p' can be indirectly addressed.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'p'	Process Data	Write register 'f'

Example:
```
MOVPF   REG1, REG2
```
Before Instruction
REG1 = 0x11
REG2 = 0x33

After Instruction
REG1 = 0x11
REG2 = 0x11

MOVWF **Move WREG to f**

Syntax: [*label*] MOVWF f

Operands: $0 \le f \le 255$

Operation: (WREG) → (f)

Status Affected: None

Encoding:

0000	0001	ffff	ffff

Description: Move data from WREG to register 'f'. Location 'f' can be anywhere in the 256 byte data space.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write register 'f'

Example:
```
MOVWF   REG
```
Before Instruction
WREG = 0x4F
REG = 0xFF

After Instruction
WREG = 0x4F
REG = 0x4F

19.8 MPLAB ICD In-Circuit Debugger

Microchip's In-Circuit Debugger, MPLAB ICD, is a powerful, low cost, run-time development tool. This tool is based on the FLASH PIC16F87X and can be used to develop for this and other PIC microcontrollers from the PIC16CXXX family. The MPLAB ICD utilizes the in-circuit debugging capability built into the PIC16F87X. This feature, along with Microchip's In-Circuit Serial Programming™ 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 watching variables, single-stepping and setting break points. Running at full speed enables testing hardware in real-time.

19.9 PRO MATE II Universal Device Programmer

The PRO MATE II universal device programmer is a full-featured programmer, capable of operating in stand-alone mode, as well as PC-hosted mode. The PRO MATE II device programmer is CE compliant.

The PRO MATE II device programmer has programmable V_{DD} and V_{PP} supplies, which allow it to verify programmed memory at V_{DD} min and V_{DD} max for maximum reliability. It has an LCD display for instructions and 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 device programmer can read, verify, or program PIC MCU devices. It can also set code protection in this mode.

19.10 PICSTART Plus Entry Level 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 all PIC devices with 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.

19.11 PICDEM 1 Low Cost PIC MCU Demonstration Board

The PICDEM 1 demonstration board 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 user can program the sample microcontrollers provided with the PICDEM 1 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The user can also connect the PICDEM 1 demonstration board to the MPLAB ICE in-circuit emulator and download the firmware to the emulator for testing. A 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.

19.12 PICDEM 2 Low Cost PIC16CXX Demonstration Board

The PICDEM 2 demonstration board 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 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The MPLAB ICE in-circuit emulator may also be used with the PICDEM 2 demonstration board to test firmware. A 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 I^2C™ bus and separate headers for connection to an LCD module and a keypad.

FIGURE 20-1: PIC17C7XX-33 VOLTAGE-FREQUENCY GRAPH

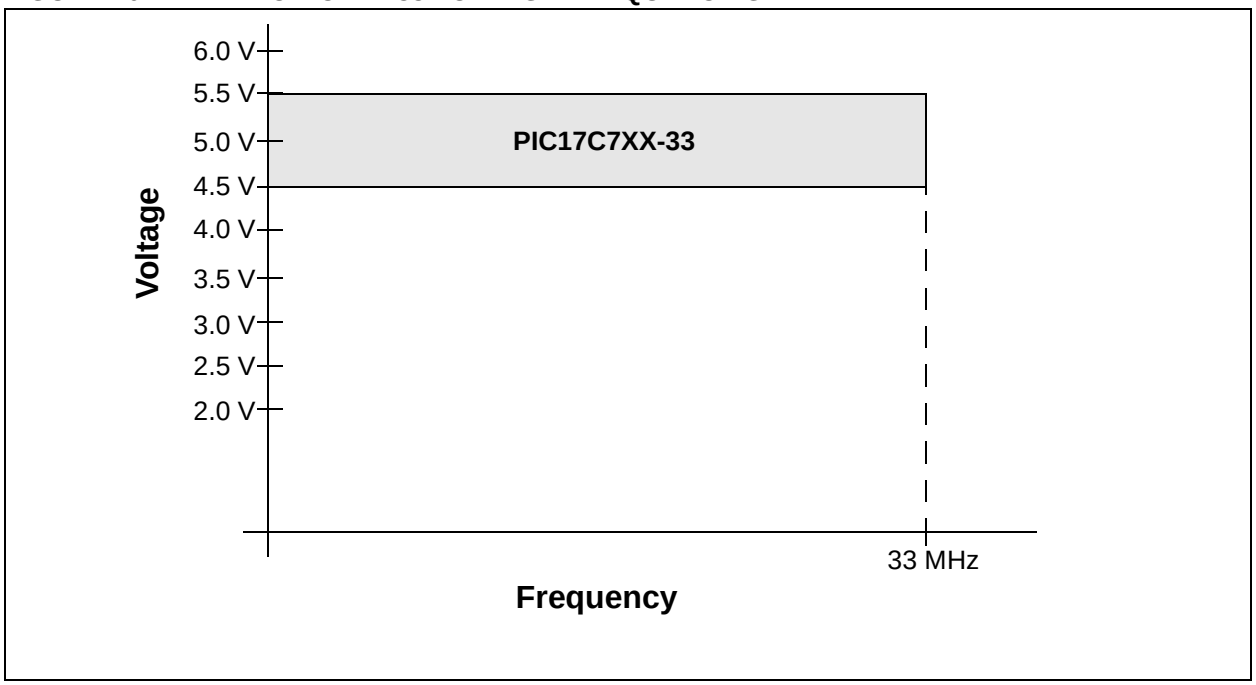
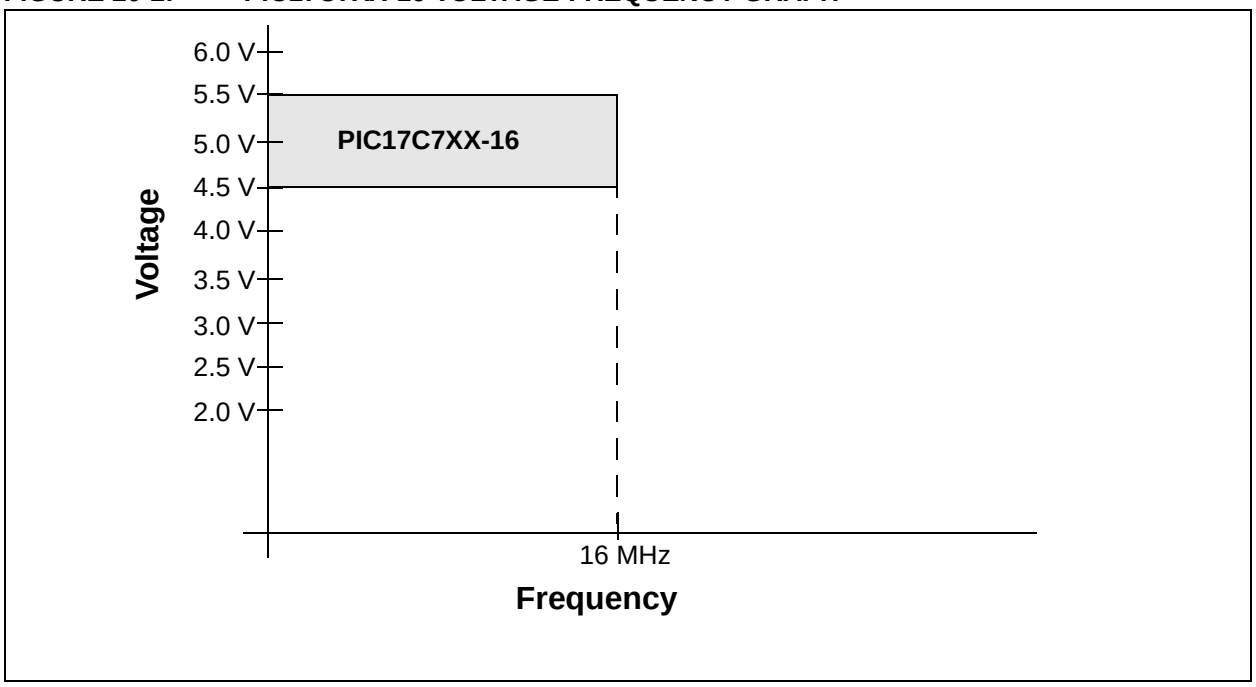


FIGURE 20-2: PIC17C7XX-16 VOLTAGE-FREQUENCY GRAPH



20.2 DC Characteristics: PIC17C7XX-16 (Commercial, Industrial, Extended) PIC17C7XX-33 (Commercial, Industrial, Extended) PIC17LC7XX-08 (Commercial, Industrial)

DC CHARACTERISTICS			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +125°C for extended -40°C ≤ TA ≤ +85°C for industrial 0°C ≤ TA ≤ +70°C for commercial Operating voltage VDD range as described in Section 20.1				
Param. No.	**Sym**	**Characteristic**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
		Input Low Voltage					
	VIL	I/O ports					
D030		with TTL buffer **(Note 6)**	Vss	–	0.8	V	4.5V ≤ VDD ≤ 5.5V
			Vss	–	0.2VDD	V	3.0V ≤ VDD ≤ 4.5V
D031		with Schmitt Trigger buffer					
		RA2, RA3	Vss	–	0.3VDD	V	I^2C compliant
		All others	Vss	–	0.2VDD	V	
D032		$\overline{MCLR}$, OSC1 (in EC and RC mode)	Vss	–	0.2VDD	V	**(Note 1)**
D033		OSC1 (in XT, and LF mode)	–	0.5VDD	–	V	
		Input High Voltage					
	VIH	I/O ports					
D040		with TTL buffer **(Note 6)**	2.0	–	VDD	V	4.5V ≤ VDD ≤ 5.5V
			1 + 0.2VDD	–	VDD	V	3.0V ≤ VDD ≤ 4.5V
D041		with Schmitt Trigger buffer					
		RA2, RA3	0.7VDD	–	VDD	V	I^2C compliant
		All others	0.8VDD	–	VDD	V	
D042		$\overline{MCLR}$	0.8VDD	–	VDD	V	**(Note 1)**
D043		OSC1 (XT, and LF mode)	–	0.5VDD	–	V	
D050	VHYS	**Hysteresis of Schmitt Trigger Inputs**	0.15VDD	–	–	V	

† Data in "Typ" column is at 5V, 25°C unless otherwise stated.

Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC17CXXX devices be driven with external clock in RC mode.

2: The leakage current on the $\overline{MCLR}$ pin is strongly dependent on the applied voltage level. Higher leakage current may be measured at different input voltages.

3: Negative current is defined as current sourced by the pin.

4: These specifications are for the programming of the on-chip program memory EPROM through the use of the table write instructions. The complete programming specifications can be found in: PIC17C7XX Programming Specifications (Literature number DS TBD).

5: The $\overline{MCLR}$/VPP pin may be kept in this range at times other than programming, but is not recommended.

6: For TTL buffers, the better of the two specifications may be used.

20.4 Timing Diagrams and Specifications

FIGURE 20-6: EXTERNAL CLOCK TIMING

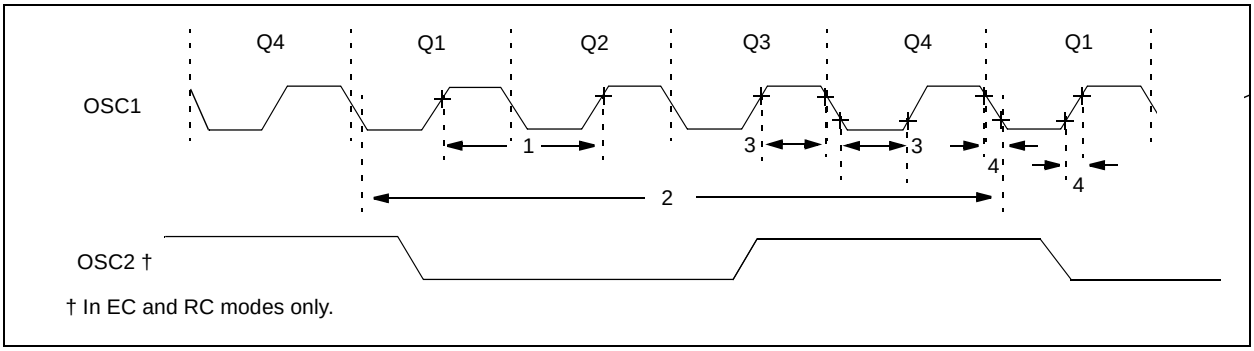


TABLE 20-1: EXTERNAL CLOCK TIMING REQUIREMENTS

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
	FOSC	**External CLKIN Frequency (Note 1)**	DC	—	8	MHz	EC osc mode - 08 devices (8 MHz devices)
			DC	—	16	MHz	- 16 devices (16 MHz devices)
			DC	—	33	MHz	- 33 devices (33 MHz devices)
		Oscillator Frequency (Note 1)	DC	—	4	MHz	RC osc mode
			2	—	8	MHz	XT osc mode - 08 devices (8 MHz devices)
			2	—	16	MHz	- 16 devices (16 MHz devices)
			2	—	33	MHz	- 33 devices (33 MHz devices)
			DC	—	2	MHz	LF osc mode
1	TOSC	**External CLKIN Period (Note 1)**	125	—	—	ns	EC osc mode - 08 devices (8 MHz devices)
			62.5	—	—	ns	- 16 devices (16 MHz devices)
			30.3	—	—	ns	- 33 devices (33 MHz devices)
		Oscillator Period (Note 1)	250	—	—	ns	RC osc mode
			125	—	1,000	ns	XT osc mode - 08 devices (8 MHz devices)
			62.5	—	1,000	ns	- 16 devices (16 MHz devices)
			30.3	—	1,000	ns	- 33 devices (33 MHz devices)
			500	—	—	ns	LF osc mode
2	TCY	**Instruction Cycle Time (Note 1)**	121.2	4/FOSC	DC	ns	
3	TOSL, TOSH	**Clock in (OSC1) High or Low Time**	10	—	—	ns	EC oscillator
4	TOSR, TOSF	**Clock in (OSC1) Rise or Fall Time**	—	—	5	ns	EC oscillator

† Data in "Typ" column is at 5V, 25°C unless otherwise stated.

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/CLKIN pin. When an external clock input is used, the "max." cycle time limit is "DC" (no clock) for all devices.

FIGURE 21-17: TYPICAL, MINIMUM AND MAXIMUM V_{OH} vs. I_{OH} (V_{DD} = 5V, -40°C TO +125°C)

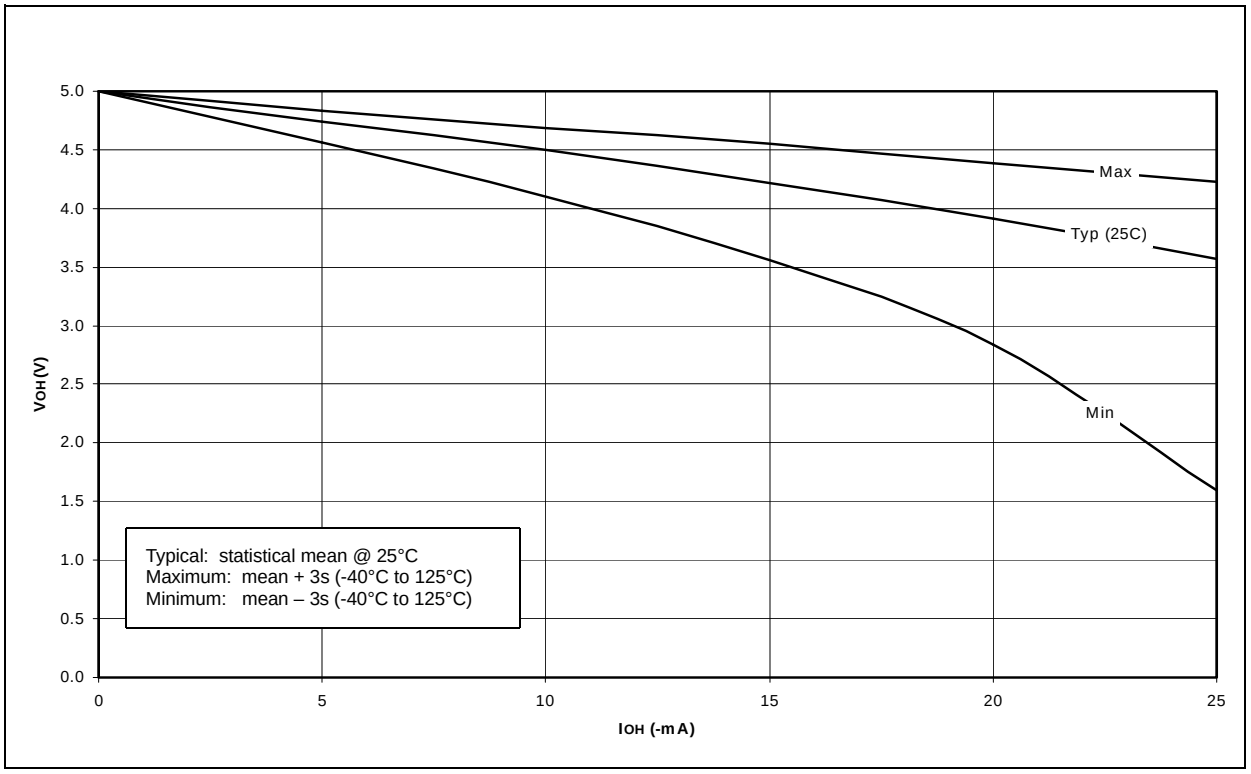


FIGURE 21-18: TYPICAL, MINIMUM AND MAXIMUM V_{OL} vs. I_{OL} (V_{DD} = 5V, -40°C TO +125°C)