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	20MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	36
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	368 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 5.5V
Data Converters	A/D 14x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.600", 15.24mm)
Supplier Device Package	40-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f747-i-p

Pin Diagrams (Continued)

PDIP (40-pin)
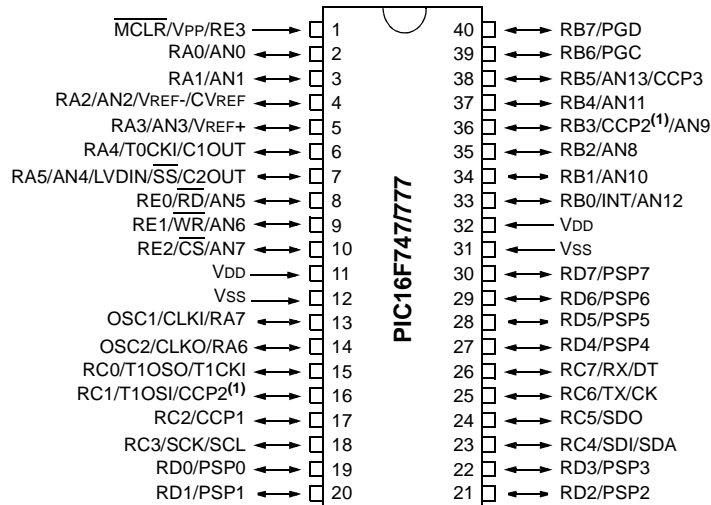
TQFP (44-pin)
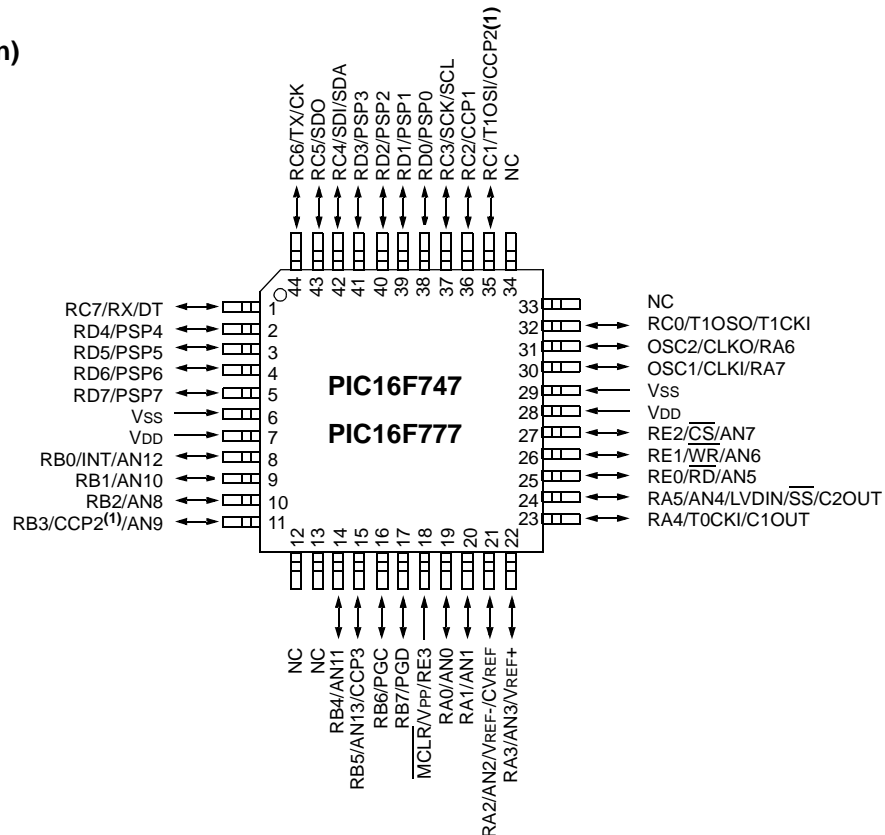
Note 1: Pin location of CCP2 is determined by the CCPMX bit in Configuration Word Register 1.

PIC16F7X7

TABLE 1-2: PIC16F737 AND PIC16F767 PINOUT DESCRIPTION

Pin Name	PDIP SOIC SSOP Pin #	QFN Pin #	I/O/P Type	Buffer Type	Description
OSC1/CLKI/RA7	9	6		ST/CMOS[3]	Oscillator crystal or external clock input.
OSC1			I		Oscillator crystal input or external clock source input. ST buffer when configured in RC mode; otherwise CMOS.
CLKI			I		External clock source input. Always associated with pin function OSC1 (see OSC1/CLKI, OSC2/CLKO pins).
RA7			I/O	ST	Digital I/O.
OSC2/CLKO/RA6	10	7		—	Oscillator crystal or clock output.
OSC2			O		Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode.
CLKO			O		In RC mode, OSC2 pin outputs CLKO which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
RA6			I/O	ST	Digital I/O.
$\overline{\text{MCLR}}$/VPP/RE3	1	26		ST	Master Clear (input) or programming voltage (output).
$\overline{\text{MCLR}}$			I		Master Clear (Reset) input. This pin is an active-low Reset to the device.
VPP			P		Programming voltage input.
RE3			I	ST	Digital input only pin.
					PORTA is a bidirectional I/O port.
RA0/AN0	2	27		TTL	
RA0			I/O		Digital I/O.
AN0			I		Analog input 0.
RA1/AN1	3	28		TTL	
RA1			I/O		Digital I/O.
AN1			I		Analog input 1.
RA2/AN2/VREF-/CVREF	4	1		TTL	
RA2			I/O		Digital I/O.
AN2			I		Analog input 2.
VREF-			I		A/D reference voltage input (low).
CVREF			0		Comparator voltage reference output.
RA3/AN3/VREF+	5	2		TTL	
RA3			I/O		Digital I/O.
AN3			I		Analog input 3.
VREF+			I		A/D reference voltage input (high).
RA4/T0CKI/C1OUT	6	3		ST	
RA4			I/O		Digital I/O – Open-drain when configured as output.
T0CKI			I		Timer0 external clock input.
C1OUT			O		Comparator 1 output bit.
RA5/AN4/LVDIN/$\overline{\text{SS}}$/C2OUT	7	4		TTL	
RA5			I/O		Digital I/O.
AN4			I		Analog input 4.
LVDIN			I/O		Low-Voltage Detect input.
$\overline{\text{SS}}$			I		SPI slave select input.
C2OUT			O		Comparator 2 output bit.

Legend: I = input   O = output   I/O = input/output   P = power
        — = Not used   TTL = TTL input   ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.
4: Pin location of CCP2 is determined by the CCPMX bit in Configuration Word Register 1.

TABLE 1-3: **PIC16F747 AND PIC16F777 PINOUT DESCRIPTION (CONTINUED)**

Pin Name	PDIP Pin #	QFN Pin #	TQFP Pin #	I/O/P Type	Buffer Type	Description
						PORTB is a bidirectional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs.
RB0/INT/AN12	33	9	8		TTL/ST[1]	
RB0				I/O		Digital I/O.
INT				I		External interrupt.
AN12				I		Analog input channel 12.
RB1/AN10	34	10	9		TTL	
RB1				I/O		Digital I/O.
AN10				I		Analog input channel 10.
RB2/AN8	35	11	10		TTL	
RB2				I/O		Digital I/O.
AN8				I		Analog input channel 8.
RB3/CCP2/AN9	36	12	11		TTL	
RB3				I/O		Digital I/O.
CCP2[5]				I/O		CCP2 capture input, compare output, PWM output.
AN9				I		Analog input channel 9.
RB4/AN11	37	14	14		TTL	
RB4				I/O		Digital I/O.
AN11				I		Analog input channel 11
RB5/AN13/CCP3	38	15	15		TTL	
RB5				I/O		Digital I/O.
AN13				I		Analog input channel 13.
CCP3				I		CCP3 capture input, compare output, PWM output.
RB6/PGC	39	16	16		TTL/ST[2]	
RB6				I/O		Digital I/O.
PGC				I/O		In-Circuit Debugger and ICSP™ programming clock.
RB7/PGD	40	17	17		TTL/ST[2]	
RB7				I/O		Digital I/O.
PGD				I/O		In-Circuit Debugger and ICSP programming data.

Legend: I = input   O = output   I/O = input/output   P = power
— = Not used   TTL = TTL input   ST = Schmitt Trigger input

Note **1:** This buffer is a Schmitt Trigger input when configured as an external interrupt.
     2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
     3: This buffer is a Schmitt Trigger input when configured as a general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).
     4: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.
     5: Pin location of CCP2 is determined by the CCPMX bit in Configuration Word Register 1.

TABLE 2-1: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Details on page
Bank 2											
100h[(4)]	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	30, 180
101h	TMR0	Timer0 Module Register								xxxx xxxx	76, 180
102h[(4)]	PCL	Program Counter (PC) Least Significant Byte								0000 0000	29, 180
103h[(4)]	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	21, 180
104h[(4)]	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	30, 180
105h	WDTCON	—	—	—	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN	---0 1000	187
106h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	64, 180
107h	—	Unimplemented								—	—
108h	—	Unimplemented								—	—
109h	LVDCON	—	—	IRVST	LVDEN	LVDL3	LVDL2	LVDL1	LVDL0	--00 0101	176
10Ah[(1,4)]	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	23, 180
10Bh[(4)]	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000x	25, 180
10Ch	PMDATA	EEPROM Data Register Low Byte								xxxx xxxx	32, 181
10Dh	PMADR	EEPROM Address Register Low Byte								xxxx xxxx	32, 181
10Eh	PMDATH	—	—	EEPROM Data Register High Byte						--xx xxxx	32, 181
10Fh	PMADRH	—	—	—	—	EEPROM Address Register High Byte				---- xxxx	32, 181
Bank 3											
180h[(4)]	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	30, 180
181h	OPTION_REG	$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	22, 180
182h[(4)]	PCL	Program Counter (PC) Least Significant Byte								0000 0000	29, 180
183h[(4)]	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	21, 180
184h[(4)]	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	30, 180
185h	—	Unimplemented								—	—
186h	TRISB	PORTB Data Direction Register								1111 1111	64, 181
187h	—	Unimplemented								—	—
188h	—	Unimplemented								—	—
189h	—	Unimplemented								—	—
18Ah[(1,4)]	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	23, 180
18Bh[(4)]	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000x	25, 180
18Ch	PMCON1	r[(6)]	—	—	—	—	—	—	RD	1--- ---0	32, 181
18Dh	—	Reserved, maintain clear								—	—
18Eh	—	Reserved, maintain clear								—	—
18Fh	—	Reserved, maintain clear								—	—

Legend: x = unknown, u = unchanged, q = value depends on condition, — = unimplemented, read as '0', r = reserved.
Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> bits, whose contents are transferred to the upper byte of the program counter during branches (CALL or GOTO).
2: Other (non Power-up) Resets include external Reset through $\overline{MCLR}$ and Watchdog Timer Reset.
3: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.
4: These registers can be addressed from any bank.
5: PORTD, PORTE, TRISD and TRISE are not physically implemented on the 28-pin devices (except for RE3), read as '0'.
6: This bit always reads as a '1'.
7: OSCCON<OSTS> bit resets to '0' with dual-speed start-up and LP, HS or HS-PLL selected as the oscillator.
8: RE3 is an input only. The state of the TRISE3 bit has no effect and will always read '1'.

2.2.2.1 Status Register

The Status register contains the arithmetic status of the ALU, the Reset status and the bank select bits for data memory.

The Status register can be the destination for any instruction, as with any other register. If the Status register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the TO and PD bits are not writable, therefore, the result of an instruction with the Status register as destination may be different than intended.

For example, CLRF STATUS, will clear the upper three bits and set the Z bit. This leaves the Status register as 000u u1uu (where u = unchanged).

It is recommended, therefore, that only BCF, BSF, SWAPF and MOVWF instructions are used to alter the Status register because these instructions do not affect the Z, C or DC bits from the Status register. For other instructions not affecting any Status bits, see **Section 16.0 "Instruction Set Summary"**.

> **Note 1:** The C and DC bits operate as a borrow and digit borrow bit, respectively, in subtraction. See the SUBLW and SUBWF instructions for examples.

REGISTER 2-1:   STATUS: ARITHMETIC STATUS REGISTER (ADDRESS 03h, 83h, 103h, 183h)

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C

bit 7   bit 0

bit 7   **IRP:** Register Bank Select bit (used for indirect addressing)
1 = Bank 2, 3 (100h-1FFh)
0 = Bank 0, 1 (00h-FFh)

bit 6-5   **RP1:RP0**: Register Bank Select bits (used for direct addressing)
11 = Bank 3 (180h-1FFh)
10 = Bank 2 (100h-17Fh)
01 = Bank 1 (80h-FFh)
00 = Bank 0 (00h-7Fh)
Each bank is 128 bytes.

bit 4   **$\overline{\text{TO}}$**: Time-out bit
1 = After power-up, CLRWDT instruction or SLEEP instruction
0 = A WDT time-out occurred

bit 3   **$\overline{\text{PD}}$**: Power-Down bit
1 = After power-up or by the CLRWDT instruction
0 = By execution of the SLEEP instruction

bit 2   **Z**: Zero bit
1 = The result of an arithmetic or logic operation is zero
0 = The result of an arithmetic or logic operation is not zero

bit 1   **DC**: Digit Carry/$\overline{\text{borrow}}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)
1 = A carry-out from the 4th low-order bit of the result occurred
0 = No carry-out from the 4th low-order bit of the result

bit 0   **C:** Carry/$\overline{\text{borrow}}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)
1 = A carry-out from the Most Significant bit of the result occurred
0 = No carry-out from the Most Significant bit of the result occurred

> **Note:** For $\overline{\text{borrow}}$, the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (RRF, RLF) instructions, this bit is loaded with either the high or low-order bit of the source register.

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

4.6 Clock Sources and Oscillator Switching

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

Essentially, there are three clock sources for these devices:

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

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

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

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

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

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

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

4.6.1 OSCCON REGISTER

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

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

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

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

4.6.2 CLOCK SWITCHING

Clock switching will occur for the following reasons:

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

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

4.7.3 SEC_RUN/RC_RUN TO PRIMARY CLOCK SOURCE

When switching from a SEC_RUN or RC_RUN mode back to the primary system clock, following a change of SCS<1:0> to '00', the sequence of events that take place will depend upon the value of the FOSC bits in the Configuration register. If the primary clock source is configured as a crystal (HS, XT or LP), then the transition will take place after 1024 clock cycles. This is necessary because the crystal oscillator has been powered down until the time of the transition. In order to provide the system with a reliable clock when the changeover has occurred, the clock will not be released to the changeover circuit until the 1024 counts have expired.

During the oscillator start-up time, the system clock comes from the current system clock. Instruction execution and/or peripheral operation continues using the currently selected oscillator as the CPU clock source, until the necessary clock count has expired, to ensure that the primary system clock is stable.

To know when the OST has expired, the OSTS bit should be monitored. OSTS = 1 indicates that the Oscillator Start-up Timer has timed out and the system clock comes from the primary clock source.

Following the oscillator start-up time, the internal Q clocks are held in the Q1 state until eight falling edge clocks are counted from the primary system clock. The clock input to the Q clocks is then released and operation resumes with the primary system clock determined by the FOSC bits (see Figure 4-10).

When in SEC_RUN mode, the act of clearing the T1OSCEN bit in the T1CON register will cause SCS<0> to be cleared, which causes the SCS<1:0> bits to revert to '00' or '10' depending on what SCS<1> is. Although the T1OSCEN bit was cleared, T1OSC will be enabled and instruction execution will continue until the OST time-out for the main system clock is complete. At that time, the system clock will switch from the T1OSC to the primary clock or the INTRC. Following this, the Timer1 oscillator will be shut-down.

> **Note:** If the primary system clock is either RC or EC, an internal delay timer (5-10 μs) will suspend operation after exiting Secondary Clock mode to allow the CPU to become ready for code execution.

4.7.3.1 Returning to Primary Clock Source Sequence

Changing back to the primary oscillator from SEC_RUN or RC_RUN can be accomplished by either changing SCS<1:0> to '00' or clearing the T1OSCEN bit in the T1CON register (if T1OSC was the secondary clock).

The sequence of events that follows is the same for both modes:

1. If the primary system clock is configured as EC, RC or INTRC, then the OST time-out is skipped. Skip to step 3.
2. If the primary system clock is configured as an external oscillator (HS, XT, LP), then the OST will be active, waiting for 1024 clocks of the primary system clock.
3. On the following Q1, the device holds the system clock in Q1.
4. The device stays in Q1 while eight falling edges of the primary system clock are counted.
5. Once the eight counts transpire, the device begins to run from the primary oscillator.
6. If the secondary clock was INTRC and the primary clock is not INTRC, the INTRC will be shut-down to save current, providing that the INTRC is not being used for any other function, such as WDT or Fail-Safe Clock Monitoring.
7. If the secondary clock was T1OSC, the T1OSC will continue to run if T1OSCEN is still set; otherwise, the Timer1 oscillator will be shut-down.

FIGURE 5-13: BLOCK DIAGRAM OF RB5/AN13/CCP3 PIN

Note 1: To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the RBPU bit.
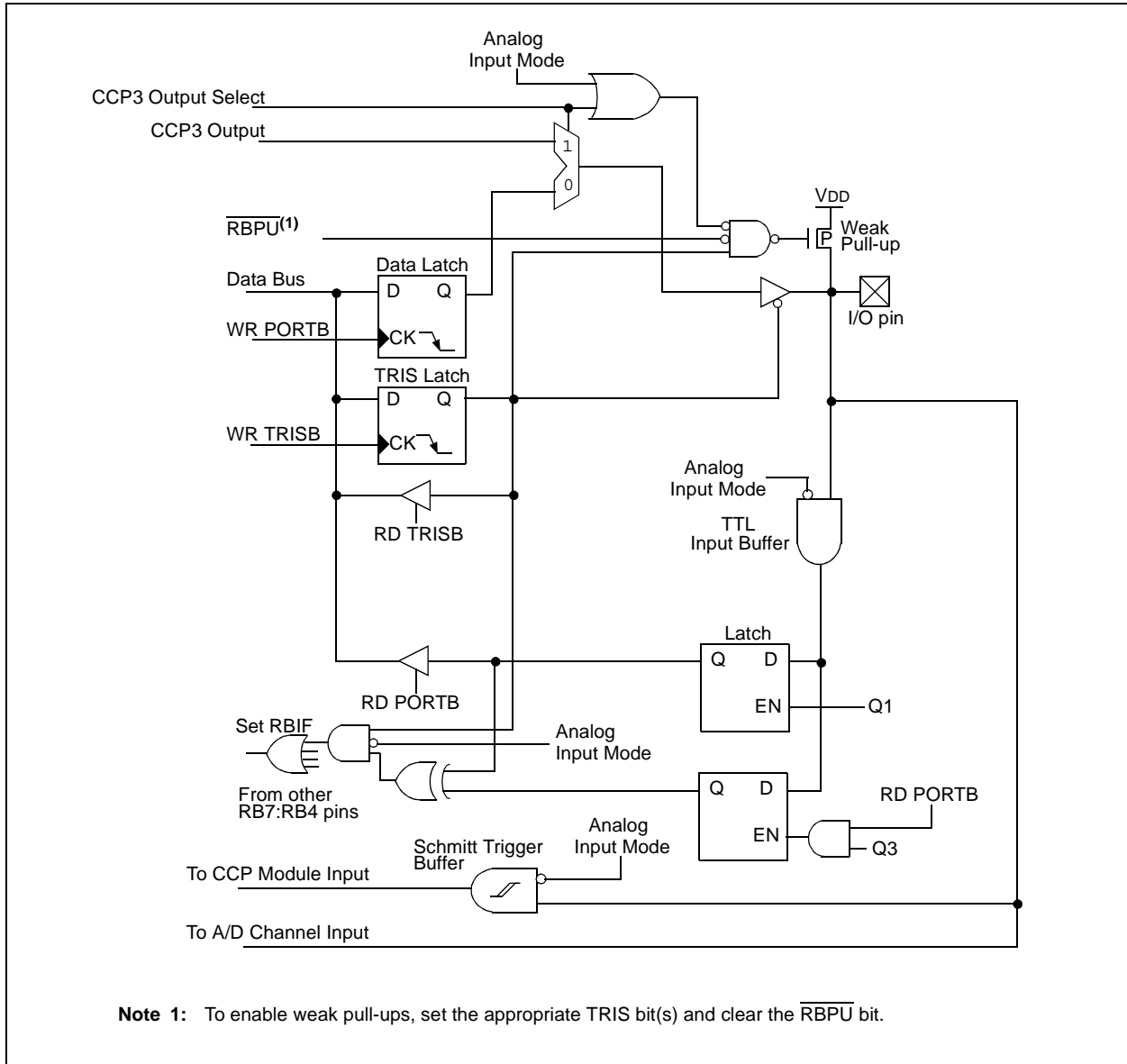
PIC16F7X7

7.5 Timer1 Operation in Asynchronous Counter Mode

If control bit, $\overline{\text{T1SYNC}}$ (T1CON<2>), is set, the external clock input is not synchronized. The timer continues to increment asynchronous to the internal phase clocks. The timer will continue to run during Sleep and can generate an interrupt on overflow that will wake-up the processor. However, special precautions in software are needed to read/write the timer (**Section 7.5.1 "Reading and Writing Timer1 in Asynchronous Counter Mode"**).

In Asynchronous Counter mode, Timer1 cannot be used as a time base for capture or compare operations.

7.5.1 READING AND WRITING TIMER1 IN ASYNCHRONOUS COUNTER MODE

Reading TMR1H or TMR1L while the timer is running from an external asynchronous clock will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself, poses certain problems, since the timer may overflow between the reads.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the Timer registers while the register is incrementing. This may produce an unpredictable value in the Timer register.

Reading the 16-bit value requires some care. The example codes provided in Example 7-1 and Example 7-2 demonstrate how to write to and read Timer1 while it is running in Asynchronous mode.

EXAMPLE 7-1: WRITING A 16-BIT FREE RUNNING TIMER

```
; All interrupts are disabled
CLRF       TMR1L           ; Clear Low byte, Ensures no rollover into TMR1H
MOVLW      HI_BYTE         ; Value to load into TMR1H
MOVWF      TMR1H, F        ; Write High byte
MOVLW      LO_BYTE         ; Value to load into TMR1L
MOVWF      TMR1H, F        ; Write Low byte
; Re-enable the Interrupt (if required)
CONTINUE                   ; Continue with your code
```

EXAMPLE 7-2: READING A 16-BIT FREE RUNNING TIMER

```
; All interrupts are disabled
MOVF       TMR1H, W        ; Read high byte
MOVWF      TMPH
MOVF       TMR1L, W        ; Read low byte
MOVWF      TMPL
MOVF       TMR1H, W        ; Read high byte
SUBWF      TMPH, W         ; Sub 1st read with 2nd read
BTFSC      STATUS, Z       ; Is result = 0
GOTO       CONTINUE        ; Good 16-bit read
; TMR1L may have rolled over between the read of the high and low bytes.
; Reading the high and low bytes now will read a good value.
MOVF       TMR1H, W        ; Read high byte
MOVWF      TMPH
MOVF       TMR1L, W        ; Read low byte
MOVWF      TMPL            ; Re-enable the Interrupt (if required)
CONTINUE                   ; Continue with your code
```

FIGURE 10-14: I²C™ SLAVE MODE TIMING WITH SEN = 1 (RECEPTION, 10-BIT ADDRESS)
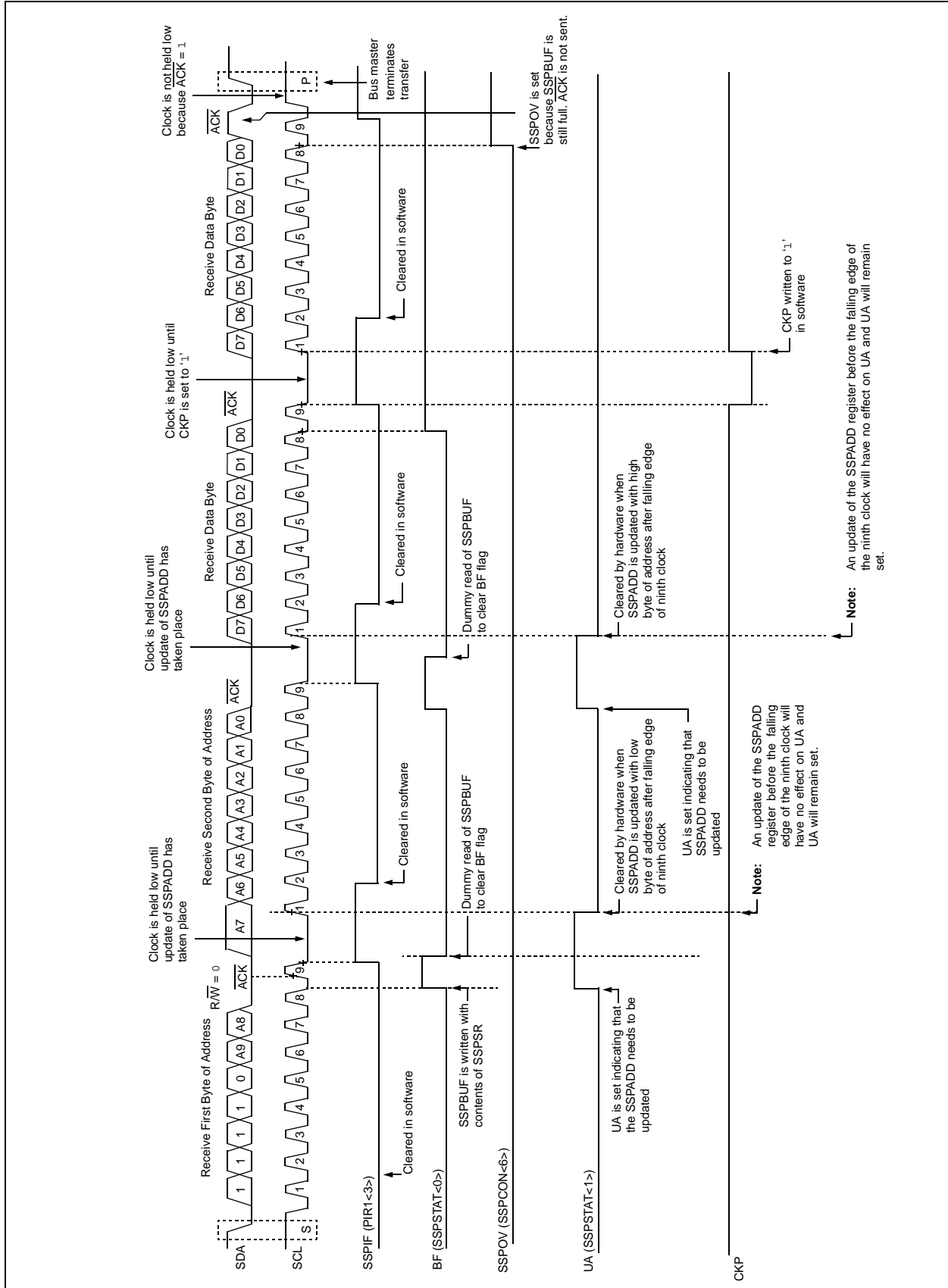
PIC16F7X7

10.4.9   I²C MASTER MODE REPEATED START CONDITION TIMING

A Repeated Start condition occurs when the RSEN bit (SSPCON2<1>) is programmed high and the I²C logic module is in the Idle state. When the RSEN bit is set, the SCL pin is asserted low. When the SCL pin is sampled low, the Baud Rate Generator is loaded with the contents of SSPADD<5:0> and begins counting. The SDA pin is released (brought high) for one Baud Rate Generator count (TBRG). When the Baud Rate Generator times out, if SDA is sampled high, the SCL pin will be deasserted (brought high). When SCL is sampled high, the Baud Rate Generator is reloaded with the contents of SSPADD<6:0> and begins counting. SDA and SCL must be sampled high for one TBRG. This action is then followed by assertion of the SDA pin (SDA = 0) for one TBRG while SCL is high. Following this, the RSEN bit (SSPCON2<1>) will be automatically cleared and the Baud Rate Generator will not be reloaded, leaving the SDA pin held low. As soon as a Start condition is detected on the SDA and SCL pins, the S bit (SSPSTAT<3>) will be set. The SSPIF bit will not be set until the Baud Rate Generator has timed out.

> **Note 1:** If RSEN is programmed while any other event is in progress, it will not take effect.
>
> **2:** A bus collision during the Repeated Start condition occurs if:
> - SDA is sampled low when SCL goes from low-to-high.
> - SCL goes low before SDA is asserted low. This may indicate that another master is attempting to transmit a data '1'.

Immediately following the SSPIF bit getting set, the user may write the SSPBUF with the 7-bit address in 7-bit mode or the default first address in 10-bit mode. After the first eight bits are transmitted and an ACK is received, the user may then transmit an additional eight bits of address (10-bit mode) or eight bits of data (7-bit mode).

10.4.9.1   WCOL Status Flag

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

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

FIGURE 10-20:   REPEATED START CONDITION WAVEFORM
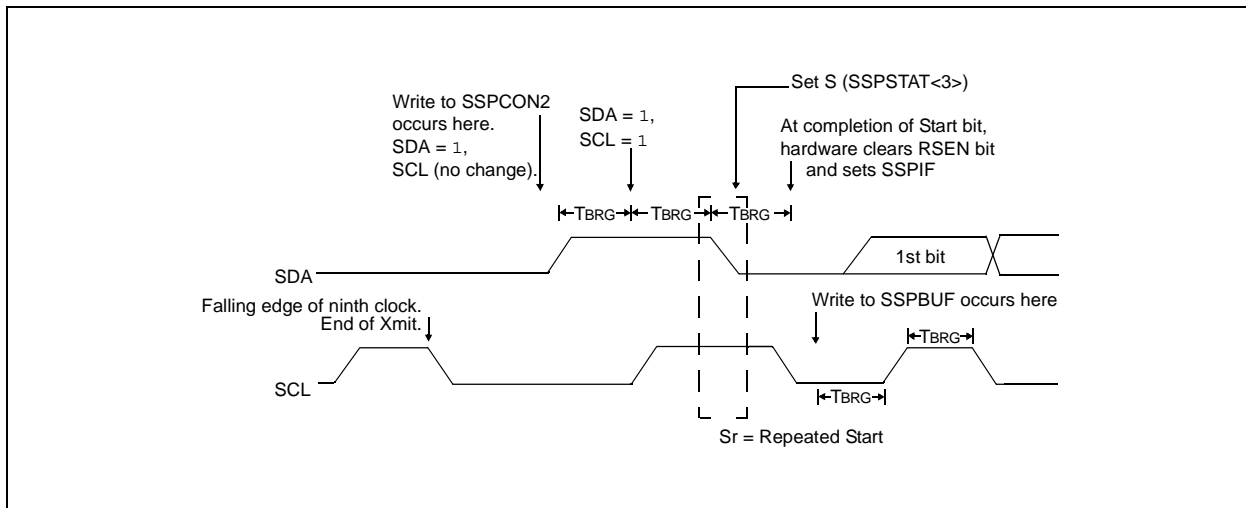
10.4.10 I²C MASTER MODE TRANSMISSION

Transmission of a data byte, a 7-bit address or the other half of a 10-bit address, is accomplished by simply writing a value to the SSPBUF register. This action will set the Buffer Full flag bit, BF and allow the Baud Rate Generator to begin counting and start the next transmission. Each bit of address/data will be shifted out onto the SDA pin after the falling edge of SCL is asserted (see data hold time specification parameter #106). SCL is held low for one Baud Rate Generator rollover count (TBRG). Data should be valid before SCL is released high (see data setup time specification parameter #107). When the SCL pin is released high, it is held that way for TBRG. The data on the SDA pin must remain stable for that duration and some hold time after the next falling edge of SCL. After the eighth bit is shifted out (the falling edge of the eighth clock), the BF flag is cleared and the master releases SDA. This allows the slave device being addressed to respond with an $\overline{ACK}$ bit, during the ninth bit time, if an address match occurred or if data was received properly. The status of $\overline{ACK}$ is written into the ACKDT bit on the falling edge of the ninth clock. If the master receives an Acknowledge, the Acknowledge Status bit, ACKSTAT, is cleared. If not, the bit is set. After the ninth clock, the SSPIF bit is set and the master clock (Baud Rate Generator) is suspended until the next data byte is loaded into the SSPBUF, leaving SCL low and SDA unchanged (Figure 10-21).

After the write to the SSPBUF, each bit of address will be shifted out on the falling edge of SCL until all seven address bits and the R/$\overline{W}$ bit are completed. On the falling edge of the eighth clock, the master will deassert the SDA pin, allowing the slave to respond with an Acknowledge. On the falling edge of the ninth clock, the master will sample the SDA pin to see if the address was recognized by a slave. The status of the ACK bit is loaded into the ACKSTAT status bit (SSPCON2<6>). Following the falling edge of the ninth clock transmission of the address, the SSPIF is set, The BF flag Is cleared and the Baud Rate Generator is turned off until another write to the SSPBUF takes place, holding SCL low and allowing SDA to float.

10.4.10.1 BF Status Flag

In Transmit mode, the BF bit (SSPSTAT<0>) is set when the CPU writes to SSPBUF and is cleared when all 8 bits are shifted out.

10.4.10.2 WCOL Status Flag

If the user writes the SSPBUF when a transmit is already in progress (i.e., SSPSR is still shifting out a data byte), the WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

WCOL must be cleared in software.

10.4.10.3 ACKSTAT Status Flag

In Transmit mode, the ACKSTAT bit (SSPCON2<6>) is cleared when the slave has sent an Acknowledge ($\overline{ACK}$ = 0) and is set when the slave does not Acknowledge ($\overline{ACK}$ = 1). A slave sends an Acknowledge when it has recognized its address (including a general call) or when the slave has properly received its data.

10.4.11 I²C MASTER MODE RECEPTION

Master mode reception is enabled by programming the Receive Enable bit, RCEN (SSPCON2<3>).

> **Note:** The MSSP module must be in an Idle state before the RCEN bit is set or the RCEN bit will be disregarded.

The Baud Rate Generator begins counting and on each rollover, the state of the SCL pin changes (high-to-low/low-to-high) and data is shifted into the SSPSR. After the falling edge of the eighth clock, the receive enable flag is automatically cleared, the contents of the SSPSR are loaded into the SSPBUF, the BF flag bit is set, the SSPIF flag bit is set and the Baud Rate Generator is suspended from counting, holding SCL low. The MSSP is now in Idle state, awaiting the next command. When the buffer is read by the CPU, the BF flag bit is automatically cleared. The user can then send an Acknowledge bit at the end of reception by setting the Acknowledge Sequence Enable bit, ACKEN (SSPCON2<4>).

10.4.11.1 BF Status Flag

In receive operation, the BF bit is set when an address or data byte is loaded into SSPBUF from SSPSR. It is cleared when the SSPBUF register is read.

10.4.11.2 SSPOV Status Flag

In receive operation, the SSPOV bit is set when 8 bits are received into the SSPSR and the BF flag bit is already set from a previous reception.

10.4.11.3 WCOL Status Flag

If the user writes the SSPBUF when a receive is already in progress (i.e., SSPSR is still shifting in a data byte), the WCOL bit is set and the contents of the buffer are unchanged (the write doesn't occur).

PIC16F7X7

11.3.2 AUSART SYNCHRONOUS MASTER RECEPTION

Once Synchronous mode is selected, reception is enabled by setting either enable bit, SREN (RCSTA<5>) or enable bit, CREN (RCSTA<4>). Data is sampled on the RC7/RX/DT pin on the falling edge of the clock. If enable bit SREN is set, then only a single word is received. If enable bit CREN is set, the reception is continuous until CREN is cleared. If both bits are set, CREN takes precedence. After clocking the last bit, the received data in the Receive Shift Register (RSR) is transferred to the RCREG register (if it is empty). When the transfer is complete, interrupt flag bit, RCIF (PIR1<5>), is set. The actual interrupt can be enabled/disabled by setting/clearing enable bit, RCIE (PIE1<5>). Flag bit RCIF is a read-only bit which is reset by the hardware. In this case, it is reset when the RCREG register has been read and is empty. The RCREG is a double-buffered register (i.e., it is a two-deep FIFO). It is possible for two bytes of data to be received and transferred to the RCREG FIFO and a third byte to begin shifting into the RSR register. On the clocking of the last bit of the third byte, if the RCREG register is still full, then Overrun Error bit, OERR (RCSTA<1>), is set. The word in the RSR will be lost. The RCREG register can be read twice to retrieve the two bytes in the FIFO. Bit OERR has to be cleared in software (by clearing bit CREN). If bit OERR is set, transfers from the RSR to the RCREG are inhibited, so it is essential to clear bit OERR if it is set. The ninth receive bit is buffered the same way as the receive

data. Reading the RCREG register will load bit RX9D with a new value; therefore, it is essential for the user to read the RCSTA register before reading RCREG in order not to lose the old RX9D information.

When setting up a Synchronous Master Reception:

1. Initialize the SPBRG register for the appropriate baud rate (see **Section 11.1 "AUSART Baud Rate Generator (BRG)"**).
2. Enable the synchronous master serial port by setting bits SYNC, SPEN and CSRC.
3. Ensure bits CREN and SREN are clear.
4. If interrupts are desired, then set enable bit RCIE.
5. If 9-bit reception is desired, then set bit RX9.
6. If a single reception is required, set bit SREN. For continuous reception, set bit CREN.
7. Interrupt flag bit RCIF will be set when reception is complete and an interrupt will be generated if enable bit RCIE was set.
8. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
9. Read the 8-bit received data by reading the RCREG register.
10. If any error occurred, clear the error by clearing bit CREN.
11. If using interrupts, ensure that GIE and PEIE (bits 7 and 6) of the INTCON register are set.

TABLE 11-11: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER RECEPTION

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, 10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
1Ah	RCREG	AUSART Receive Register								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 Generator Register								0000 0000	0000 0000

Legend: x = unknown, — = unimplemented, read as '0'. Shaded cells are not used for synchronous master reception.
Note 1: Bits PSPIE and PSPIF are reserved on 28-pin devices; always maintain these bits clear.

PIC16F7X7
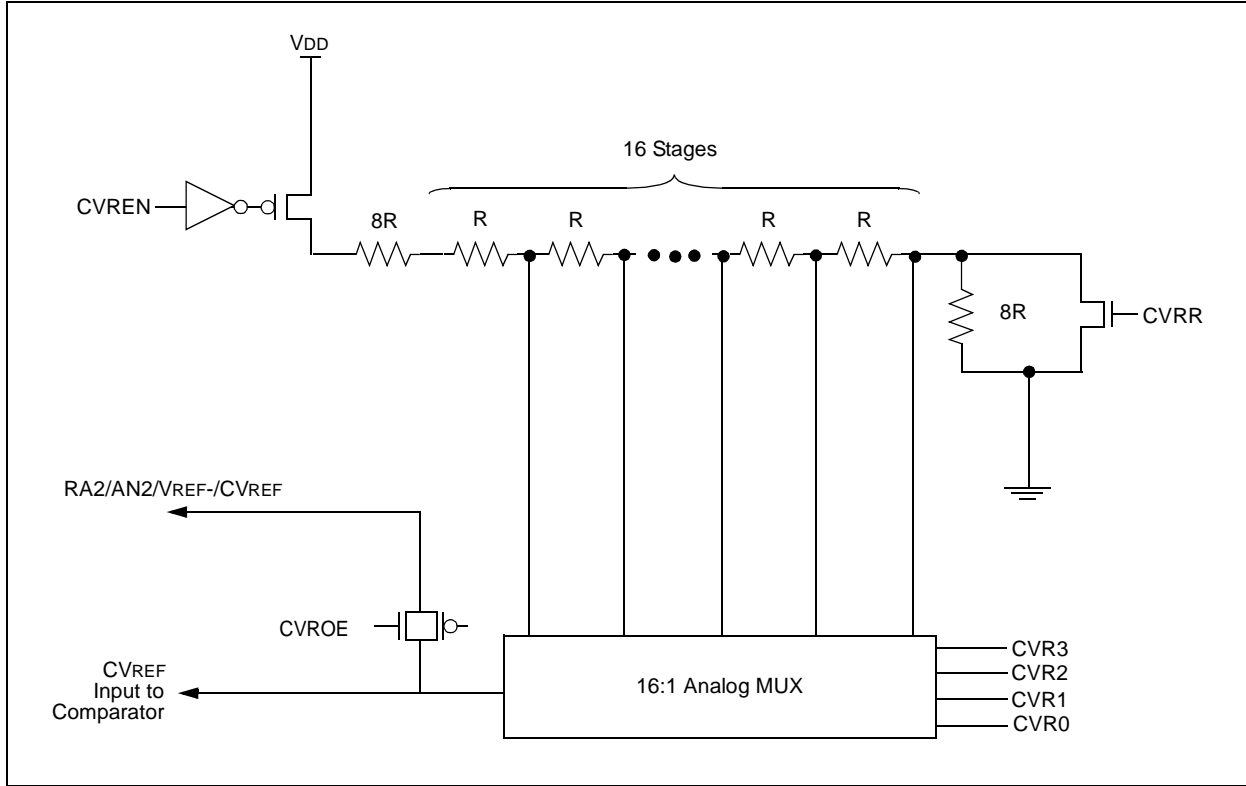
FIGURE 14-1: COMPARATOR VOLTAGE REFERENCE BLOCK DIAGRAM

TABLE 14-1: REGISTERS ASSOCIATED WITH COMPARATOR VOLTAGE REFERENCE

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on all other Resets
9Dh	CVRCON	CVREN	CVROE	CVRR	—	CVR3	CVR2	CVR1	CVR0	000- 0000	000- 0000
9Ch	CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0111	0000 0111

Legend: x = unknown, u = unchanged, — = unimplemented, read as '0'.
Shaded cells are not used with the comparator voltage reference.

PIC16F7X7

15.9   Control Register

The Low-Voltage Detect Control register controls the operation of the Low-Voltage Detect circuitry.

REGISTER 15-3:   **LVDCON: LOW-VOLTAGE DETECT CONTROL REGISTER (ADDRESS 109h)**

U-0	U-0	R-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-1
—	—	IRVST	LVDEN	LVDL3	LVDL2	LVDL1	LVDL0

bit 7   bit 0

bit 7-6   **Unimplemented:** Read as '0'

bit 5   **IRVST:** Internal Reference Voltage Stable Flag bit

1 = Indicates that the Low-Voltage Detect logic will generate the interrupt flag at the specified voltage range

0 = Indicates that the Low-Voltage Detect logic will not generate the interrupt flag at the specified voltage range and the LVD interrupt should not be enabled

bit 4   **LVDEN:** Low-Voltage Detect Power Enable bit

1 = Enables LVD, powers up LVD circuit
0 = Disables LVD, powers down LVD circuit

bit 3-0   **LVDL3:LVDL0:** Voltage Detection Limit bits

1111 = External analog input is used (input comes from the LVDIN pin)
1110 = Maximum setting
.
.
.
0001 = Minimum setting

Note: See Table 18-3 in **Section 18.0 "Electrical Characteristics"** for the specifications.

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

17.7   MPLAB SIM Software Simulator

The MPLAB SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC® DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB SIM Software Simulator fully supports symbolic debugging using the MPLAB C Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

17.8   MPLAB REAL ICE In-Circuit Emulator System

MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs PIC® Flash MCUs and dsPIC® Flash DSCs with the easy-to-use, powerful graphical user interface of the MPLAB Integrated Development Environment (IDE), included with each kit.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB IDE. In upcoming releases of MPLAB IDE, new devices will be supported, and new features will be added. MPLAB REAL ICE offers significant advantages over competitive emulators including low-cost, full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, a ruggedized probe interface and long (up to three meters) interconnection cables.

17.9   MPLAB ICD 3 In-Circuit Debugger System

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

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

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

The MPLAB PICkit 3 allows debugging and programming of PIC® and dsPIC® Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB Integrated Development Environment (IDE). The MPLAB PICkit 3 is connected to the design engineer's PC using a full speed USB interface and can be connected to the target via an Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the reset line to implement in-circuit debugging and In-Circuit Serial Programming™.

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

PIC16F7X7

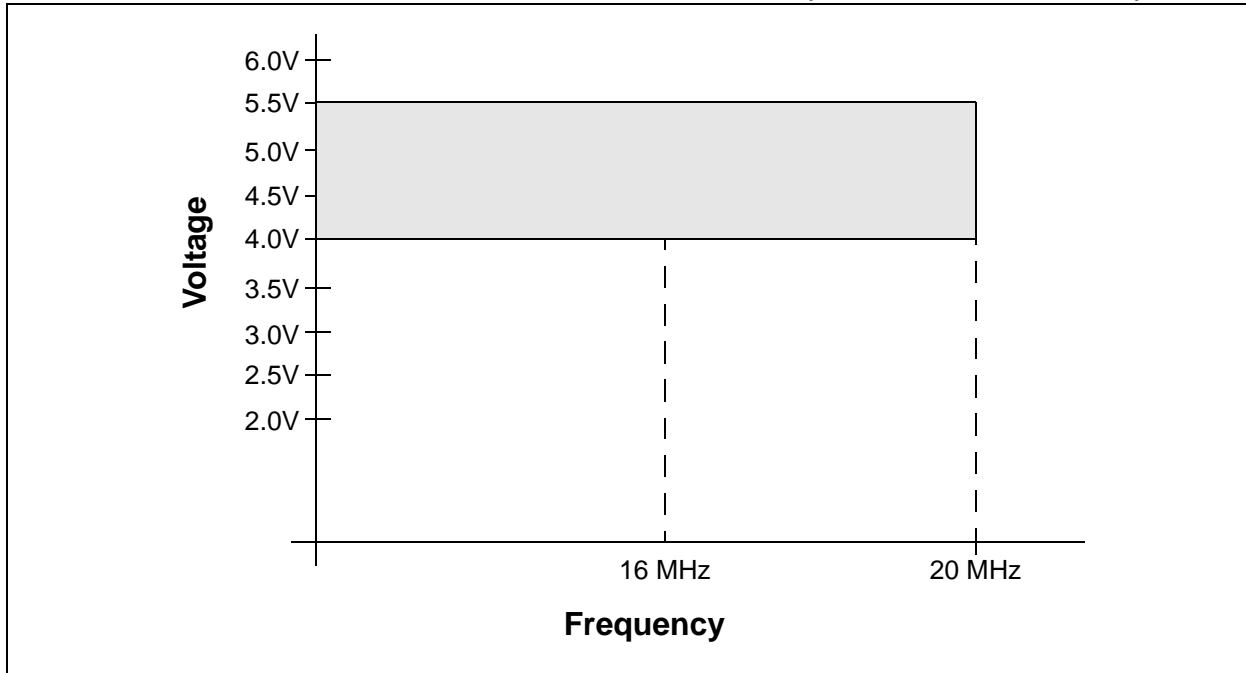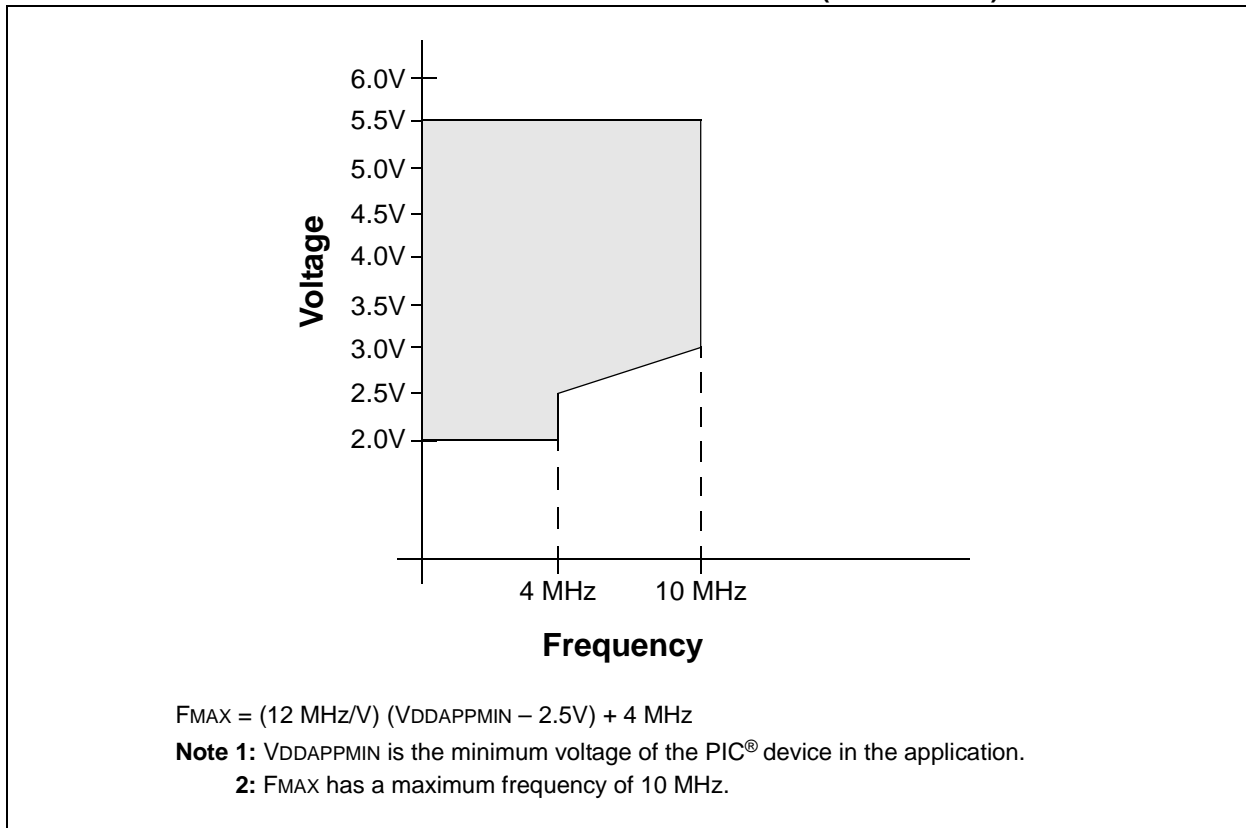**FIGURE 18-1:** **PIC16F7X7 VOLTAGE-FREQUENCY GRAPH (INDUSTRIAL, EXTENDED)**

FIGURE 18-2: **PIC16LF7X7 VOLTAGE-FREQUENCY GRAPH (INDUSTRIAL)**

$F_{MAX} = (12 \text{ MHz/V}) (V_{DDAPPMIN} - 2.5\text{V}) + 4 \text{ MHz}$

Note 1: $V_{DDAPPMIN}$ is the minimum voltage of the PIC® device in the application.

    2: F_{MAX} has a maximum frequency of 10 MHz.

18.2   DC Characteristics: Power-Down and Supply Current
PIC16F737/747/767/777 (Industrial, Extended)
PIC16LF737/747/767/777 (Industrial) (Continued)

PIC16LF737/747/767/777 (Industrial)	Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial				
PIC16F737/747/767/777 (Industrial, Extended)	**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				

Param No.	Device	Typ	Max	Units	Conditions	
	Module Differential Currents (ΔIWDT, ΔIBOR, ΔILVD, ΔIOSCB, ΔIAD)					
D022 (ΔIWDT)	**Watchdog Timer**	1.5	3.8	μA	-40°C	VDD = 2.0V
		2.2	3.8	μA	+25°C	
		2.7	4.0	μA	+85°C	
		2.3	4.6	μA	-40°C	VDD = 3.0V
		2.7	4.6	μA	+25°C	
		3.1	4.8	μA	+85°C	
		3.0	10.0	μA	-40°C	VDD = 5.0V
		3.3	10.0	μA	+25°C	
		3.9	13.0	μA	+85°C	
	Extended devices	5.0	21.0	μA	+125°C	
D022A (ΔIBOR)	**Brown-out Reset**	17	35	μA	-40°C to +85°C	VDD = 3.0V
		47	45	μA	-40°C to +85°C	VDD = 5.0V
		0	0	μA	-40°C to +85°C	VDD = 2.0V VDD = 3.0V VDD = 5.0V BOREN:BORSEN = `10` in Sleep mode
	Extended devices	48	50	μA	-40°C to +125°C	VDD = 5.0V
D022B (ΔILVD)	**Low-Voltage Detect**	14	25	μA	-40°C to +85°C	VDD = 2.0V
		18	35	μA	-40°C to +85°C	VDD = 3.0V
		21	45	μA	-40°C to +85°C	VDD = 5.0V
	Extended devices	24	50	μA	-40°C to +125°C	VDD = 5.0V

Legend: Shading of rows is to assist in readability of the table.

Note 1:   The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD or VSS and all features that add delta current disabled (such as WDT, Timer1 Oscillator, BOR, etc.).

2:   The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:
> OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD;
> MCLR = VDD; WDT enabled/disabled as specified.

3:   For RC oscillator configurations, current through REXT is not included. The current through the resistor can be estimated by the formula Ir = VDD/2REXT (mA) with REXT in kΩ.

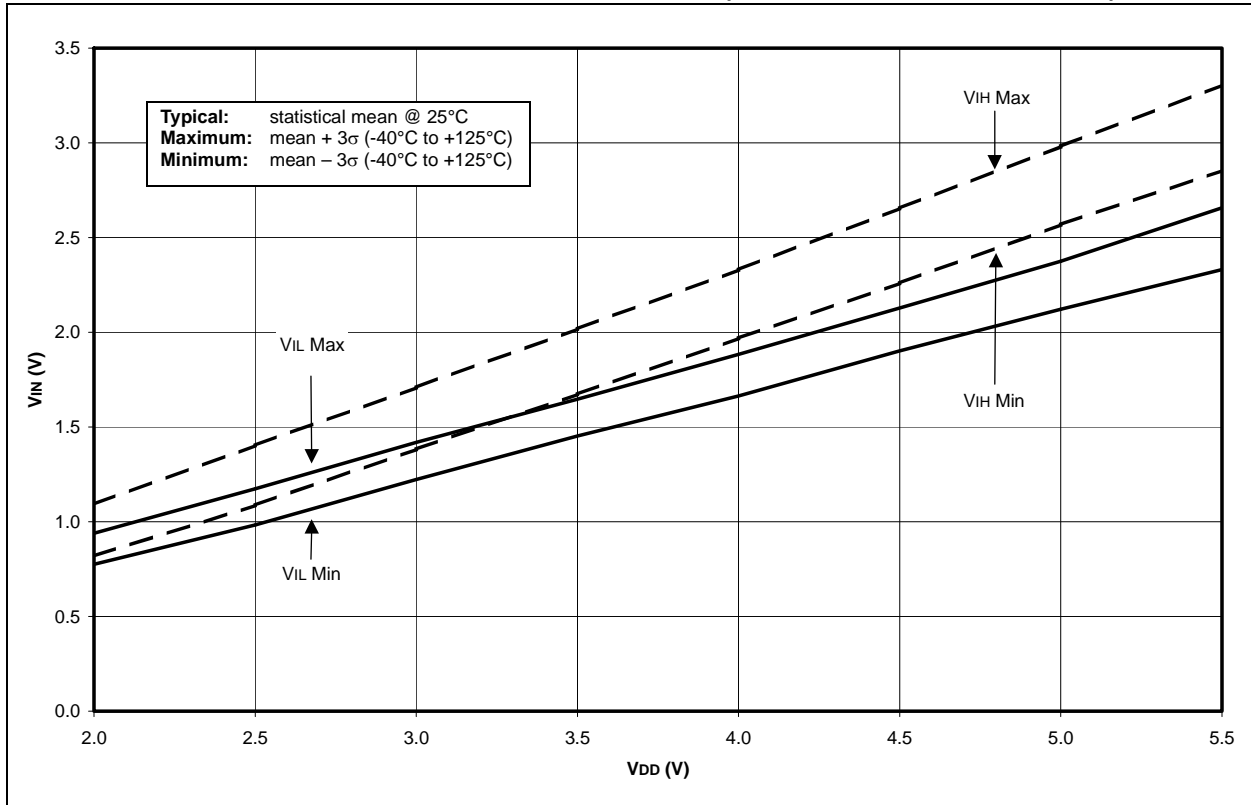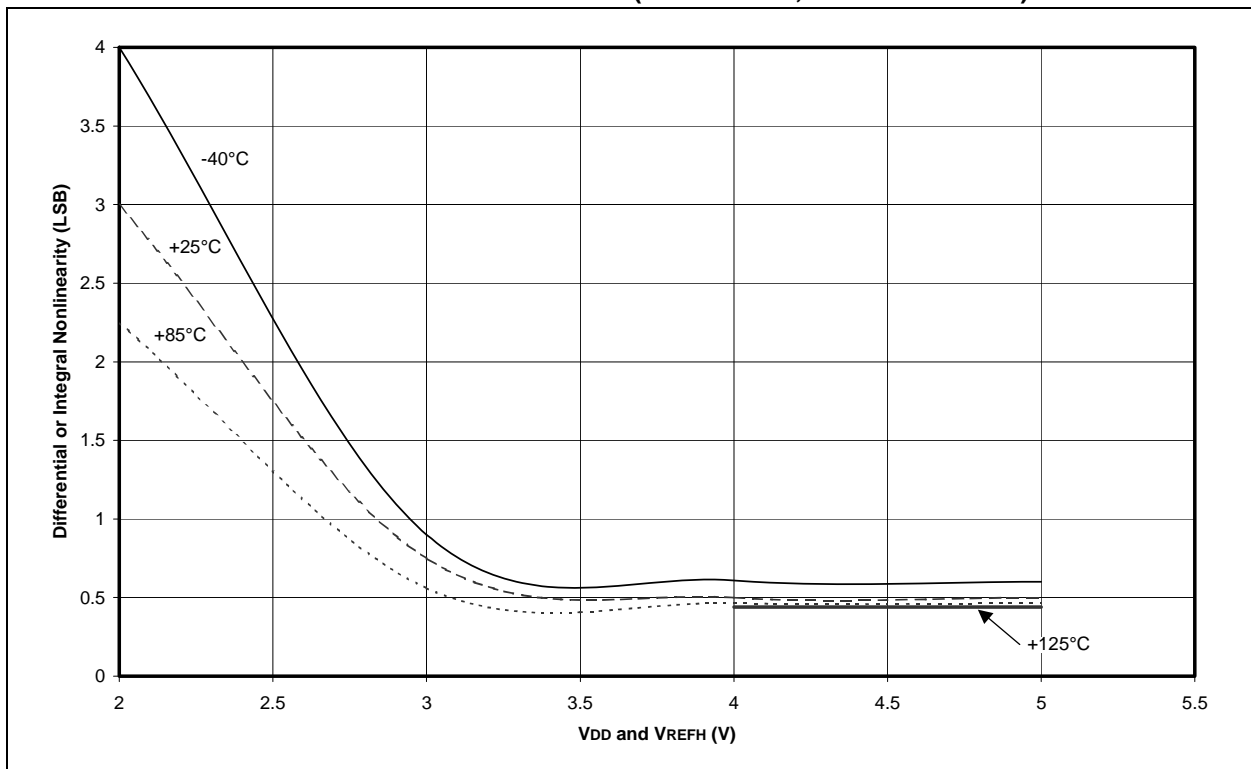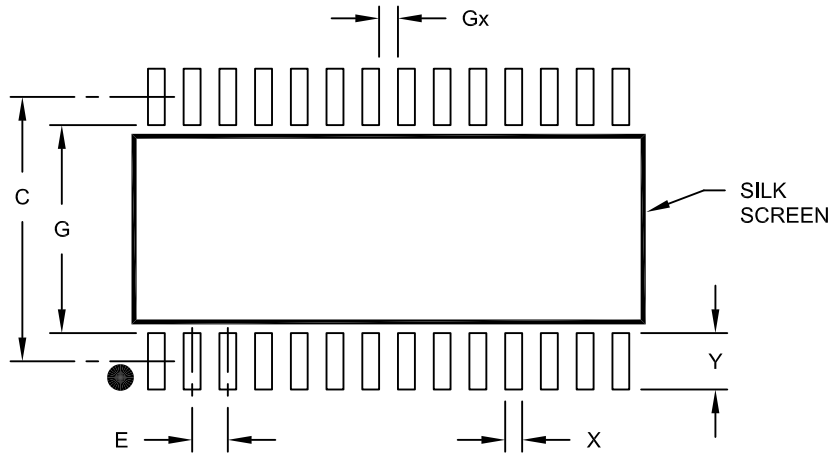**FIGURE 19-25:** MINIMUM AND MAXIMUM VIN vs. VDD (I²C™ INPUT, -40°C TO +125°C)

FIGURE 19-26: A/D NONLINEARITY vs. VREFH (VDD = VREFH, -40°C TO +125°C)

PIC16F7X7

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

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

RECOMMENDED LAND PATTERN

	Units		MILLIMETERS	
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		1.27 BSC	
Contact Pad Spacing	C		9.40	
Contact Pad Width (X28)	X			0.60
Contact Pad Length (X28)	Y			2.00
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	7.40		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2052A