Section 7. Oscillator

HIGHLIGHTS

This section of the manual contains the following topics:

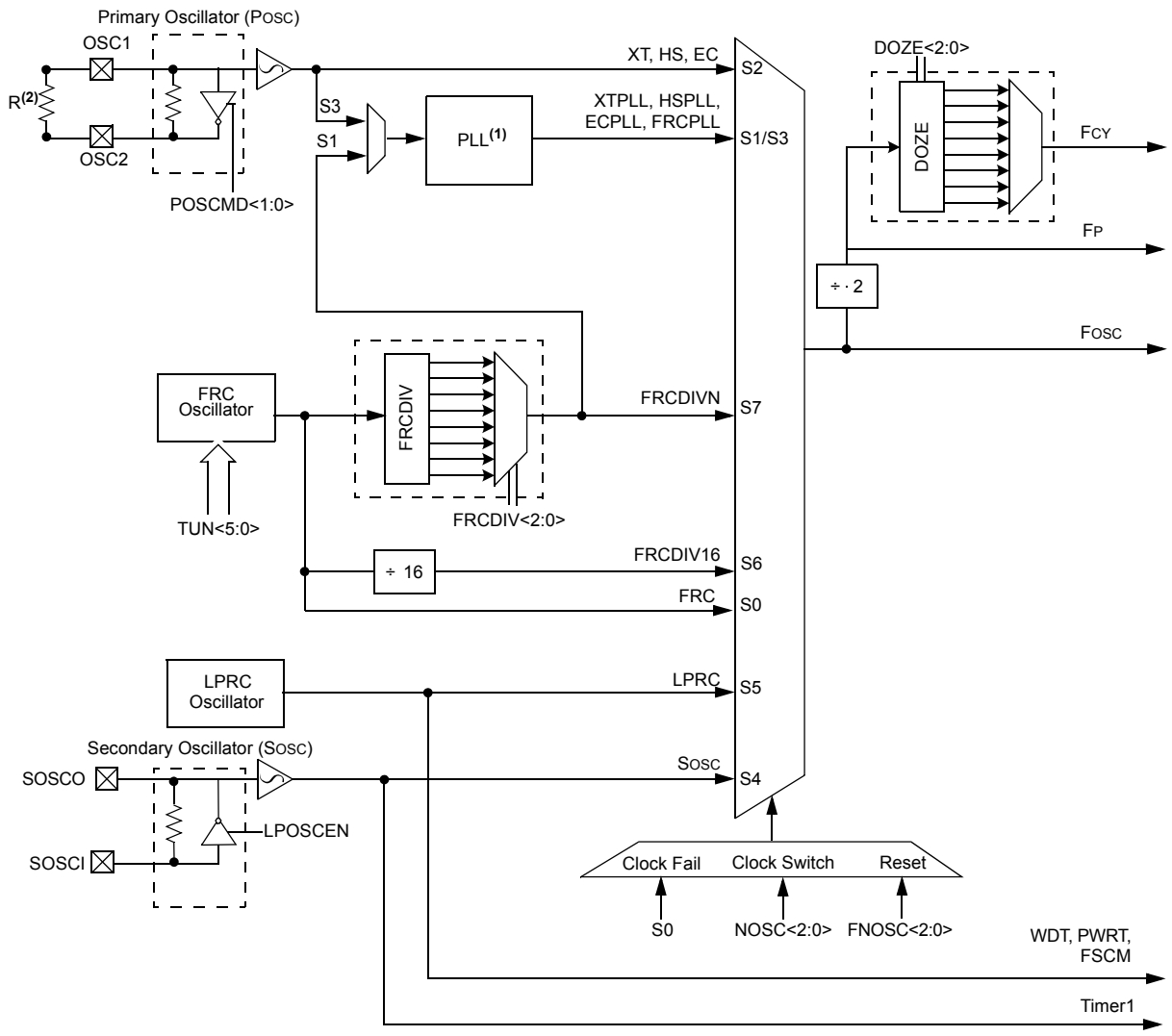
7.1 INTRODUCTION

The dsPIC33F oscillator system includes these characteristics:

- Four external and internal oscillator options
- On-chip Phase-Locked Loop (PLL) to boost the internal operating frequency on selected internal and external oscillator sources
- On-the-fly clock switching between various clock sources
- Doze mode for system power-saving
- Fail-Safe Clock Monitor (FSCM) that detects clock failure and permits safe application recovery or shutdown
- Nonvolatile Configuration bits for clock source selection

A block diagram of the dsPIC33F oscillator system is shown in Figure 7-1.

Figure 7-1: Oscillator System Block Diagram



Note 1: See Figure 7-8 for PLL details.

2: If the oscillator is used with XT or HS modes, an external parallel resistor with the value of 1 MΩ must be connected.

7.2 CPU CLOCKING

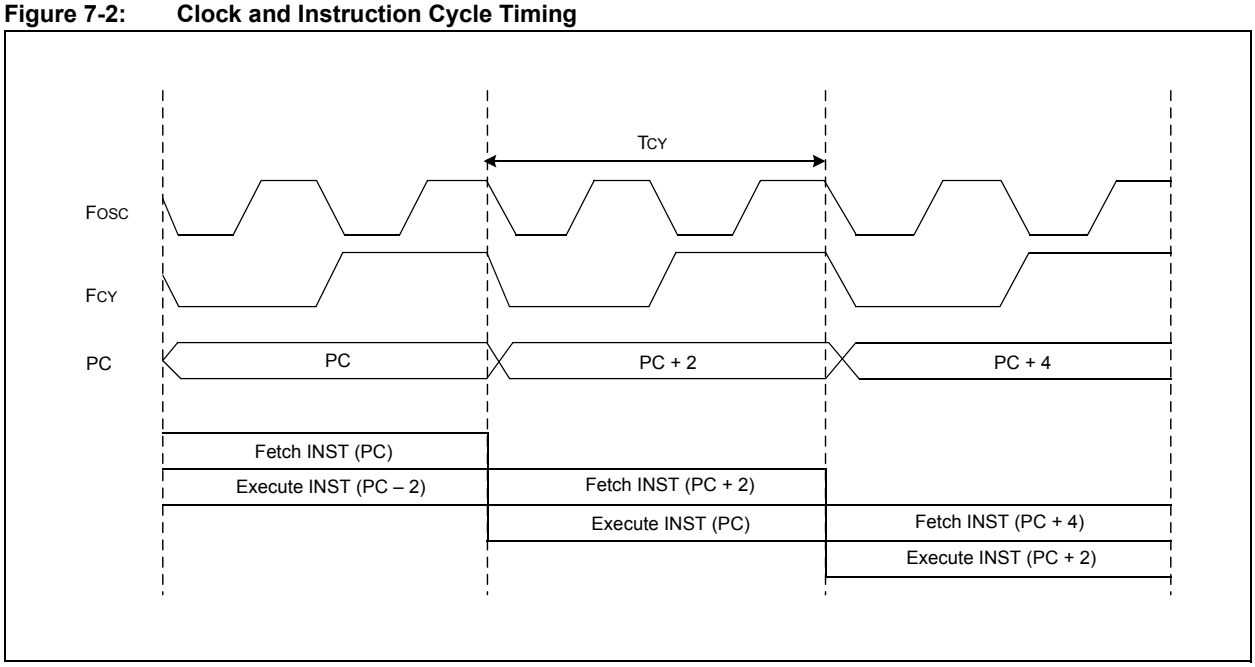
The system clock (FOSC) source can be provided by one of the following options:

- Primary Oscillator (POSC) on the OSC1 and OSC2 pins
- Secondary Oscillator (SOSC) on the SOSCI and SOSCO pins
- Internal Fast RC (FRC) Oscillator with optional clock divider
- Internal Low-Power RC (LPRC) Oscillator
- POSC with PLL
- Internal FRC Oscillator with PLL

The FOSC source is divided by 2 to produce the internal instruction cycle clock. In this document, the instruction cycle clock is denoted by FCY. The timing diagram in Figure 7-2 shows the relationship among FOSC, instruction cycle clock (FCY), and the Program Counter (PC).

FCY can be output on the OSC2 I/O pin if the Primary Oscillator mode or the High Speed Oscillator (HS) mode is not selected as the clock source (see **Section 7.5 "Primary Oscillator (Posc)"**).

Figure 7-2: Clock and Instruction Cycle Timing



7.3 OSCILLATOR CONFIGURATION REGISTERS

Oscillator Configuration registers are located in the program memory space, and are not Special Function Registers (SFRs). These two registers are mapped into program memory space and are programmed at the time of device programming.

- **FOSCSEL: Oscillator Source Selection Register**

  FOSCSEL selects the initial oscillator source and start-up option. FOSCSEL contains the following Configuration bit:

  FNOSC<2:0> Configuration bits in the Oscillator Source Selection (FOSCSEL<2:0>) register determine the clock source that is used at a Power-on Reset (POR). Thereafter, the clock source can be changed between permissible clock sources with clock switching.

  The Internal FRC Oscillator with postscaler (FRCDIVN) is the default (unprogrammed) selection.

- **FOSC: Oscillator Configuration Register**

  FOSC configures the Primary Oscillator mode, OSCO pin function, peripheral pin select, and the fail-safe and clock switching modes. FOSC contains the following Configuration bits:

  - POSCMD (FOSC<1:0>) Configuration bits select the operation mode of the POSC.
  - OSCIOFNC (FOSC<2>) Configuration bit selects the OSC2 pin function, except in HS or Medium-Speed Oscillator (XT) mode.

    If OSCIOFNC is unprogrammed ('1'), the FCY clock is output on the OSC2 pin.
    If OSCIOFNC is programmed ('0'), the OSC2 pin becomes a general purpose I/O pin.

Table 7-1 lists the configuration settings that select the device oscillator source and operating mode at a POR.

Table 7-1: Configuration Bit Values for Clock Selection

Oscillator Source	Oscillator Mode	FNOSC Value	POSCMD Value	Note
S0	Fast RC Oscillator (FRC)	000	xx	**1**
S1	Fast RC Oscillator with PLL (FRCPLL)	001	xx	**1**
S2	Primary Oscillator (EC)	010	00	**1**
S2	Primary Oscillator (XT)	010	01	
S2	Primary Oscillator (HS)	010	10	
S3	Primary Oscillator with PLL (ECPLL)	011	00	**1**
S3	Primary Oscillator with PLL (XTPLL)	011	01	
S3	Primary Oscillator with PLL (HSPLL)	011	10	
S4	Secondary Oscillator (SOSC)	100	xx	**1**
S5	Low-Power RC (LPRC) Oscillator	101	xx	**1**
S6	Fast RC Oscillator with ÷ 16 divider (FRCDIV16)	110	xx	**1**
S7	Fast RC Oscillator with ÷ N divider (FRCDIVN)	111	xx	**1, 2**

Note 1: OSC2 pin function is determined by the OSCIOFNC Configuration bit.
2: Default oscillator mode for an unprogrammed (erased) device.

Register 7-1: FOSCSEL: Oscillator Source Selection Register

U	U	U	U	U	U	U	U
—	—	—	—	—	—	—	—
bit 15							bit 8

R/P	U	U	U	U	R/P	R/P	R/P
IESO	—	—	—	—	FNOSC<2:0>		
bit 7							bit 0

Legend:			
R = Readable bit	P = Programmable bit	U = Unused bits, program to Logic ‘1’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘1’

bit 7 **IESO:** Internal External Start-up Option bit
1 = Start-up device with the Internal FRC Oscillator, then automatically switch to the user-selected oscillator source when ready
0 = Start-up device with user-selected oscillator source

bit 6-3 **Unimplemented:** Read as ‘1’

bit 2-0 **FNOSC<2:0>:** Initial Oscillator Source Selection bits
111 = Fast RC Oscillator with Divide-by-N (FRCDIVN)
110 = Fast RC Oscillator with Divide-by-16 (FRCDIV16)
101 = Low-Power RC (LPRC) Oscillator
100 = Secondary Oscillator (SOSC)
011 = Primary Oscillator with PLL (XTPLL, HSPLL, ECPLL)
010 = Primary Oscillator (XT, HS, EC)
001 = Fast RC Oscillator with PLL (FRCPLL)
000 = Fast RC Oscillator (FRC)

Register 7-2: FOSC: Oscillator Configuration Register

U	U	U	U	U	U	U	U
—	—	—	—	—	—	—	—
bit 15							bit 8

R/P	R/P	R/P	U	U	R/P	R/P	R/P
FCKSM<1:0>		IOL1WAY(1)	—	—	OSCIOFNC	POSCMD<1:0>	
bit 7							bit 0

Legend:			
R = Readable bit	P = Programmable bit	U = Unused bits, program to Logic '1'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as '1'

bit 7-6 **FCKSM<1:0>:** Clock Switching Mode bits
1x = Clock switching is disabled, Fail-Safe Clock Monitor (FSCM) is disabled
01 = Clock switching is enabled, FSCM is disabled
00 = Clock switching is enabled, FSCM is enabled

bit 5 **IOL1WAY:** Peripheral Pin Select Configuration bit(1)
1 = Allow only one reconfiguration
0 = Allow multiple reconfigurations

bit 4-3 **Unimplemented:** Read as '1'

bit 2 **OSCIOFNC:** OSC2 Pin Function bit (except in XT and HS modes)
1 = OSC2 is the clock output and the instruction cycle (FCY) clock is output on OSC2 pin
0 = OSC2 is a general purpose digital I/O pin

bit 1-0 **POSCMD<1:0>:** Primary Oscillator Mode Selection bits
11 = Primary Oscillator is disabled
10 = HS Crystal Oscillator mode
01 = XT Crystal Oscillator mode
00 = EC (External Clock) mode

Note 1: The IOL1WAY bit is not available on all dsPIC33F devices. Refer to the specific device data sheet for more information.

7.4 SPECIAL FUNCTION REGISTERS

The following SFRs provide run-time control and the status of the oscillator system:

- **OSCCON: Oscillator Control Register(2)**

  OSCCON controls clock switching and provides status information that allows current clock source, PLL lock, and clock fail conditions to be monitored.

- **CLKDIV: Clock Divisor Register**

  CLKDIV controls Doze mode and selects a PLL prescaler, a PLL postscaler, and an FRC postscaler.

- **PLLFBD: PLL Feedback Divisor Register**

  PLLFBD selects the PLL feedback divisor.

- **OSCTUN: FRC Oscillator Tuning Register**

  OSCTUN is used to tune the Internal FRC Oscillator frequency in software. It allows the Internal FRC Oscillator frequency to be adjusted over a range of ±12%.

Note:	The oscillator SFRs (OSCCON, CLKDIV, PLLFBD and OSCTUN) are reset only on a POR.

Register 7-3: OSCCON: Oscillator Control Register(2)

U-0	R-y	R-y	R-y	U-0	R/W-y	R/W-y	R/W-y
—	COSC<2:0>			—	NOSC<2:0>		
bit 15							bit 8

R/S-0	R/W-0	R-0	U-0	R/C-0	U-0	R/W-0	R/W-0
CLKLOCK	IOLOCK(1)	LOCK	—	CF	—	LPOSCEN	OSWEN
bit 7							bit 0

Legend:	U = Unimplemented bit, read as '0'	y = Depends on FOSCSEL<FNOSC> bits	
R = Readable bit	W = Writable bit	C = Clearable bit	S = Settable bit
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as '0'

bit 14-12 **COSC<2:0>:** Current Oscillator Selection bits (read-only)
111 = Fast RC Oscillator with Divide-by-N (FRCDIVN)
110 = Fast RC Oscillator with Divide-by-16 (FRCDIV16)
101 = Low-Power RC (LPRC) Oscillator
100 = Secondary Oscillator (SOSC)
011 = Primary Oscillator with PLL (XTPLL, HSPLL, ECPLL)
010 = Primary Oscillator (XT, HS, EC)
001 = Fast RC Oscillator with PLL (FRCPLL)
000 = Fast RC Oscillator (FRC)

bit 11 **Unimplemented:** Read as '0'

bit 10-8 **NOSC<2:0>:** New Oscillator Selection bits
111 = Fast RC Oscillator with Divide-by-N (FRCDIVN)
110 = Fast RC Oscillator with Divide-by-16 (FRCDIV16)
101 = Low-Power RC (LPRC) Oscillator
100 = Secondary Oscillator (SOSC)
011 = Primary Oscillator with PLL (XTPLL, HSPLL, ECPLL)
010 = Primary Oscillator (XT, HS, EC)
001 = Fast RC Oscillator with PLL (FRCPLL)
000 = Fast RC Oscillator (FRC)

bit 7 **CLKLOCK:** Clock Lock Enable bit
If clock switching is enabled and FSCM is disabled, FCKSM<1:0> (FOSC<7:6>) = 01:
1 = Clock switching is disabled, system clock source is locked
0 = Clock switching is enabled, system clock source may be modified by clock switching

bit 6 **IOLOCK:** Peripheral Pin Select (PPS) Lock bit(1)
1 = Peripheral Pin Select is locked. Writes to Peripheral Pin Select registers are not allowed.
0 = Peripheral Pin Select is not locked. Writes to Peripheral Pin Select registers are allowed.

bit 5 **LOCK:** PLL Lock Status bit (read-only)
1 = Indicates that PLL is in lock, or PLL start-up timer is satisfied
0 = Indicates that PLL is out of lock, start-up timer is in progress or PLL is disabled

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

bit 3 **CF:** Clock Fail Detect bit (read or cleared by application)
1 = FSCM has detected clock failure
0 = FSCM has not detected clock failure

Note 1: The IOLOCK bit is not available on all dsPIC33F devices. Refer to the specific device data sheet for more information.

2: Writes to this register require an unlock sequence. For details and examples refer to **7.11 "Clock Switching"**.

Register 7-3: OSCCON: Oscillator Control Register(2)

bit 2 **Unimplemented:** Read as '0'

bit 1 **LPOSCEN:** Secondary Oscillator Enable bit
1 = Enable Secondary Oscillator
0 = Disable Secondary Oscillator

bit 0 **OSWEN:** Oscillator Switch Enable bit
1 = Request oscillator switch to selection specified by the NOSC<2:0> bits
0 = Oscillator switch is complete

Note 1: The IOLOCK bit is not available on all dsPIC33F devices. Refer to the specific device data sheet for more information.

2: Writes to this register require an unlock sequence. For details and examples refer to **7.11 "Clock Switching"**.

Register 7-4: CLKDIV: Clock Divisor Register

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
ROI	DOZE<2:0>			DOZEN(1)	FRCDIV<2:0>		
bit 15							bit 8

R/W-0	R/W-1	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PLLPOST<1:0>		—	PLLPRE<4:0>				
bit 7							bit 0

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

bit 15 **ROI:** Recover On Interrupt bit
1 = Interrupts clear the DOZEN bit and the processor clock, and peripheral clock ratio is set to 1:1
0 = Interrupts have no effect on the DOZEN bit

bit 14-12 **DOZE<2:0>:** Processor Clock Reduction Select bits(2)
111 = FCY divided by 128
110 = FCY divided by 64
101 = FCY divided by 32
100 = FCY divided by 16
011 = FCY divided by 8 (default)
010 = FCY divided by 4
001 = FCY divided by 2
000 = FCY divided by 1

bit 11 **DOZEN:** Doze Mode Enable bit(1,2)
1 = DOZE<2:0> field specifies the ratio between the peripheral clocks and the processor clocks
0 = Processor clock and peripheral clock ratio forced to 1:1

bit 10-8 **FRCDIV<2:0>:** Internal Fast RC Oscillator Postscaler bits
111 = FRC divided by 256
110 = FRC divided by 64
101 = FRC divided by 32
100 = FRC divided by 16
011 = FRC divided by 8
010 = FRC divided by 4
001 = FRC divided by 2
000 = FRC divided by 1 (default)

bit 7-6 **PLLPOST<1:0>:** PLL VCO Output Divider Select bits (also denoted as 'N2', PLL postscaler)
00 = Output divided by 2
01 = Output divided by 4 (default)
10 = Reserved
11 = Output divided by 8

bit 5 **Unimplemented:** Read as '0'

Note 1: This bit is cleared when the ROI bit is set and an interrupt occurs.

2: For more information on Doze mode, refer to **Section 9. "Watchdog Timer and Power-Saving Modes"** (DS70196).

Register 7-4: CLKDIV: Clock Divisor Register (Continued)

bit 4-0 **PLLPRE<4:0>:** PLL Phase Detector Input Divider Select bits (also denoted as 'N1', PLL prescaler)
11111 = Input divided by 33
•
•
•
00001 = Input divided by 3
00000 = Input divided by 2 (default)

Note 1: This bit is cleared when the ROI bit is set and an interrupt occurs.

2: For more information on Doze mode, refer to **Section 9. "Watchdog Timer and Power-Saving Modes"** (DS70196).

Register 7-5: PLLFBD: PLL Feedback Divisor Register

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	PLLDIV<8>
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
PLLDIV<7:0>							
bit 7							bit 0

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

bit 15-9 **Unimplemented:** Read as '0'

bit 8-0 **PLLDIV<8:0>:** PLL Feedback Divisor bits (also denoted as 'M', PLL multiplier)

111111111 = 513
•
•
•
000110000 = 50 (default)
•
•
•
000000010 = 4
000000001 = 3
000000000 = 2

Register 7-6: OSCTUN: FRC Oscillator Tuning Register

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	TUN<5:0>					
bit 7							bit 0

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

bit 15-6 **Unimplemented:** Read as '0'

bit 5-0 **TUN<5:0>:** Fast RC Oscillator Tuning bits

011111 = Center frequency + 11.625% (8.23 MHz)
011110 = Center frequency + 11.25% (8.20 MHz)
•
•
•
000001 = Center frequency + 0.375% (7.40 MHz)
000000 = Center frequency (7.37 MHz nominal)
111111 = Center frequency – 0.375% (7.345 MHz)
•
•
•
100001 = Center frequency – 11.625% (6.52 MHz)
100000 = Center frequency – 12% (6.49 MHz)

7.5 PRIMARY OSCILLATOR (POSC)

The POSC is available on the OSC1 and OSC2 pins of the dsPIC33F device family. This connection enables an external crystal (or ceramic resonator) to provide the clock to the device. Optionally, it can be used with the internal PLL to boost the FOSC to 80 MHz for 40 MIPS execution. The POSC provides the following modes of operation:

- **Medium-Speed Oscillator (XT Mode)**

  XT mode is a medium-gain, medium-frequency mode that is used to work with crystal frequencies of 3-10 MHz.

- **High-Speed Oscillator (HS Mode)**

  HS mode is a high-gain, high-frequency mode that is used to work with crystal frequencies of 10-40 MHz.

- **External Clock Source Operation (EC Mode)**

  If the on-chip oscillator is not used, EC mode allows the internal oscillator to be bypassed. The device clocks are generated from an external source (0.8-64 MHz) and input on the OSC1 pin.

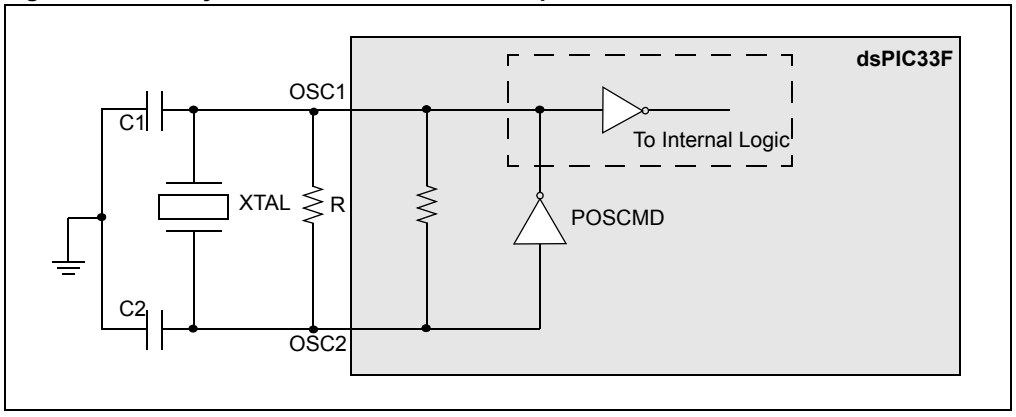
The FNOSC<2:0> Configuration bits in the Oscillator Source Selection register (FOSCSEL<2:0>) specify the POSC clock source at a POR. The POSCMD<1:0> Configuration bits in the Oscillator Configuration register (FOSC<1:0>) specify the Primary Oscillator mode. Table 7-2 shows the options selected by specific bit configurations, which are programmed at the time of device programming.

Table 7-2: Primary Oscillator Clock Source Options

FNOSC Value	POSCMD	Primary Oscillator Source and Mode
010	00	Primary Oscillator: External Clock Mode (EC)
010	01	Primary Oscillator: Medium Frequency Mode (XT)
010	10	Primary Oscillator: High-Frequency Mode (HS)
011	00	Primary Oscillator with PLL: External Clock Mode (ECPLL)
011	01	Primary Oscillator with PLL: Medium-Frequency Mode (XTPLL)
011	10	Primary Oscillator with PLL: High-Frequency Mode (HSPLL)

A diagram of the crystal oscillator circuit that is recommended for the dsPIC33F device is presented in Figure 7-1.

Figure 7-3: Crystal or Ceramic Resonator Operation in XT or HS Oscillator Mode



Capacitors C1 and C2 form the load capacitance for the crystal.

The optimum load capacitance (C_L) for a given crystal is specified by the crystal manufacturer. C_L can be calculated as shown in Equation 7-1.

Equation 7-1: Crystal Load Capacitance

$$C_L = C_S + \frac{C1 \text{ x } C2}{C1 + C2}$$

Where:
C_S is the stray capacitance.

Assuming C1 = C2, Equation 7-2 gives the capacitor value (C1, C2) for a given load and stray capacitance.

Equation 7-2: External Capacitor for Crystal

$$C1 = C2 = 2 \text{ x } (C_L - C_S)$$

For additional information on crystal oscillators and their operation, refer to **7.14 "Related Application Notes"**.

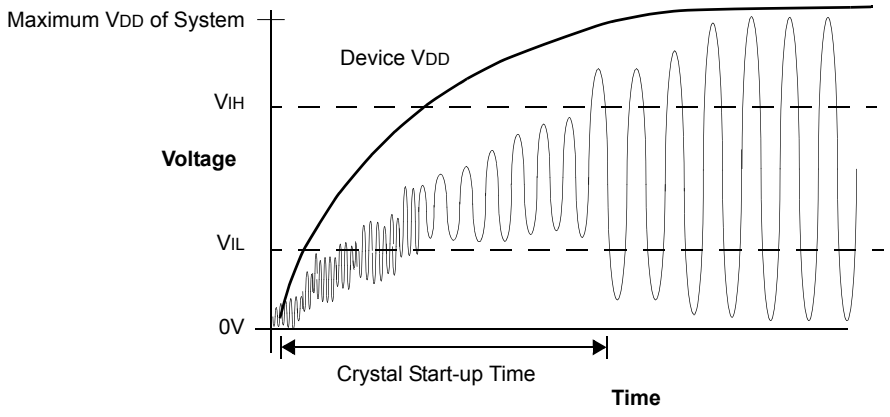
7.5.1 Oscillator Start-up Time

As the device voltage increases from V_{SS}, the oscillator will start its oscillations.The time required for the oscillator to start oscillating depends on many factors. These include:

- Crystal and resonator frequency
- Capacitor values used
- Device V_{DD} rise time
- System temperature
- Series resistor value and type if used
- Oscillator mode selection of device (selects the gain of the internal oscillator inverter)
- Crystal quality
- Oscillator circuit layout
- System noise

Figure 7-4 shows a plot of a typical oscillator and resonator start-up.

Figure 7-4: Example Oscillator and Resonator Start-up Characteristics



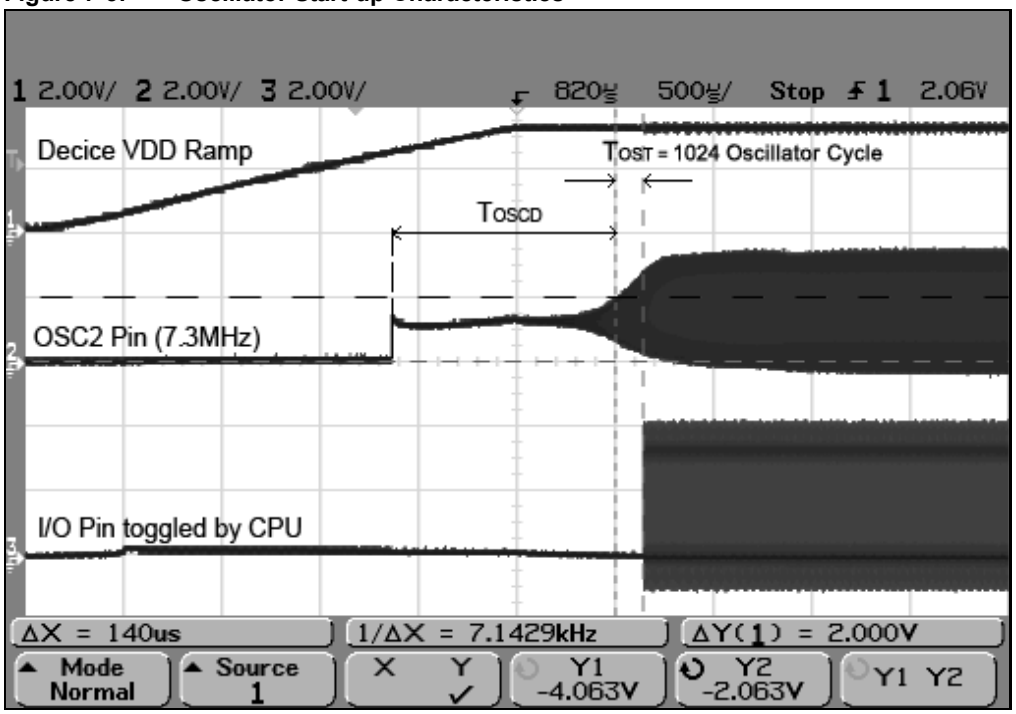
To ensure that a crystal oscillator (or ceramic resonator) has started and stabilized, an Oscillator Start-up Timer (OST) is provided with the POSC and SOSC. The OST is a simple 10-bit counter that counts 1024 cycles before releasing the oscillator clock to the rest of the system. This time-out period is denoted as T_{OST}.

The amplitude of the oscillator signal must reach the VIL and VIH thresholds for the oscillator pins before the OST can begin to count cycles. The TOST interval is required every time the oscillator restarts (e.g., on a POR, BOR and wake-up from Sleep mode).

When the POSC is enabled, it takes a finite amount of time to start oscillating. This delay is denoted as TOSCD. After TOSCD, the OST timer takes 1024 clock cycles (TOST) to release the clock. The total delay for the clock to be ready is TOSCD + TOST. If the PLL is used, an additional delay is required for the PLL to lock, see **7.7 "Phase-Locked Loop"**.

POSC start-up behavior is illustrated in Figure 7-5, indicating where the CPU begins toggling an I/O pin when it starts execution after the TOSCD + TOST interval.

Figure 7-5: Oscillator Start-up Characteristics



7.5.2 POSC Pin Functionality

The POSC pins (OSC1 and OSC2) can be used for other functions when the oscillator is not being used. POSCMD<1:0> Configuration bits in the Oscillator Configuration register (FOSC<1:0>) determine the oscillator pin function. The OSCIOFNC bit (FOSC<2>) determines the OSC2 pin function.

POSCMD: Primary Oscillator Mode Selection bits:

- 11 = Primary Oscillator mode disabled
- 10 = HS Oscillator mode selected
- 01 = XT Oscillator mode selected
- 00 = External Clock mode selected

OSCIOFNC: OSC2 Pin Function bit (except in XT and HS modes):

- 1 = OSC2 is the clock output and the instruction cycle (FCY) clock is output on the OCS2 pin, see Figure 7-6
- 0 = OSC2 is a general purpose digital I/O pin, see Figure 7-7

The oscillator pin functions are provided in Table 7-3.

Table 7-3: Clock Pin Function Selection

Oscillator Source	OSCIOFNC Value	POSCMD<1:0> Value	OSC1(1) Pin Function	OSC2(2) Pin Function
POSC Disabled	1	11	Digital I/O	Clock Output (FCY)
POSC Disabled	0	11	Digital I/O	Digital I/O
HS	x	10	OSC1	OSC2
XT	x	01	OSC1	OSC2
EC	1	00	OSC1	Clock Output (FCY)
EC	0	00	OSC1	Digital I/O

Note 1: OSC1 pin function is determined by the Primary Oscillator Mode Configuration bits (POSCMOD<1:0>).

2: OSC2 pin function is determined by the Primary Oscillator Mode (POSCMOD<1:0>) and the OSC2 Pin Function Configuration bits (OSCIOFNC).

Figure 7-6: OSC2 Pin for Clock Output (in EC Mode)

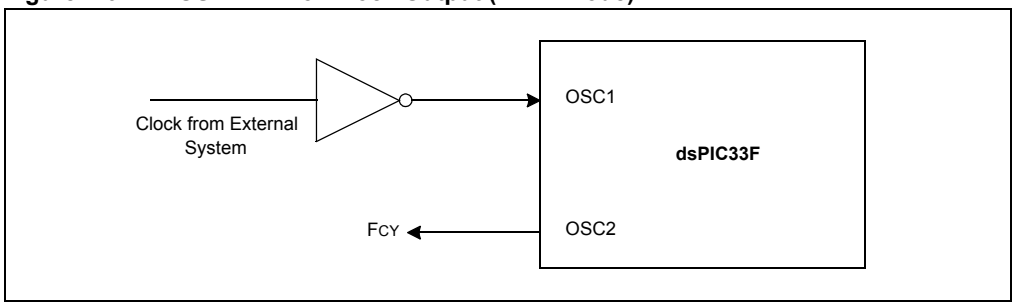
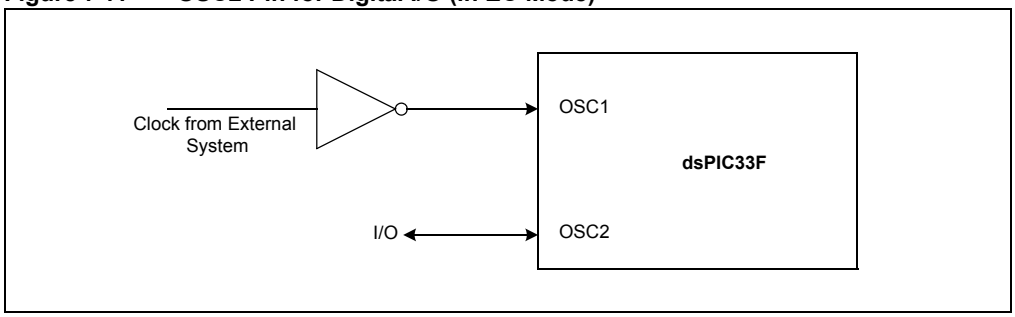


Figure 7-7: OSC2 Pin for Digital I/O (in EC Mode)



7.6 INTERNAL FAST RC (FRC) OSCILLATOR

The Internal FRC Oscillator provides a nominal 7.37 MHz clock without requiring an external crystal or ceramic resonator, which results in system cost savings for applications that do not require a precise clock reference.

The application software can tune the frequency of the oscillator from -12% to +11.625% (30 kHz steps) of the nominal frequency value using the FRC Oscillator Tuning bits (TUN<5:0>) in the FRC Oscillator Tuning register (OSCTUN<5:0>).

Note 1: Refer to the specific device data sheet for the accuracy of the FRC clock frequency over temperature and voltage variations.

2: The FRC Oscillator Tuning bits (TUN<5:0>) should not be changed dynamically when operating in internal FRC with PLL.

To change the FRC Oscillator Tuning bits:

a) Switch the clock to non-PLL mode (e.g., Internal FRC).

b) Make the necessary changes.

c) Switch the clock back to PLL mode.

The Internal FRC Oscillator starts up instantly. Unlike a crystal oscillator, which can take several milliseconds to begin oscillation, the Internal FRC Oscillator starts oscillating immediately.

The Initial Oscillator Source Selection Configuration bits (FNOSC<2:0>) in the Oscillator Source Selection register (FOSCSEL<2:0>) select the FRC clock source. The FRC clock source options at the time of a POR are shown in Table 7-4. The Configuration bits are programmed at the time of device programming.

Table 7-4: FRC Clock Source Options at POR

FNOSC<2:0> Value	Primary Oscillator Source and Mode
000	FRC Oscillator (FRC)
001	FRC Oscillator with PLL (FRCPLL)
110	FRC Oscillator: Postscaler divide by 16 (FRCDIV16)
111	FRC Oscillator: Postscaler divide by N (FRCDIVN)

7.6.1 FRC Postscaler Mode (FRCDIVN)

In FRC Postscaler mode, a variable postscaler divides the FRC clock output and allows a lower frequency to be chosen. The postscaler is controlled by the Internal FRC Oscillator Postscaler bits (FRCDIV<2:0>) in the Clock Divisor register (CLKDIV<10:8>), which allow eight settings, from 1:1-1:256, to be chosen, as shown in Table 7-5.

Table 7-5: Internal FRC Oscillator Postscaler Settings

FRCDIV<2:0> Value	Internal FRC Oscillator Settings
000	FRC divide by 1 (default)
001	FRC divide by 2
010	FRC divide by 4
011	FRC divide by 8
100	FRC divide by 16
101	FRC divide by 32
110	FRC divide by 64
111	FRC divide by 256

Optionally, the FRC postscaler output can be used with the internal PLL to boost FOSC to 80 MHz for 40 MIPS instruction cycle execution speed.

Note: The FRC divider should not be changed dynamically when operating in Internal FRC with a PLL.

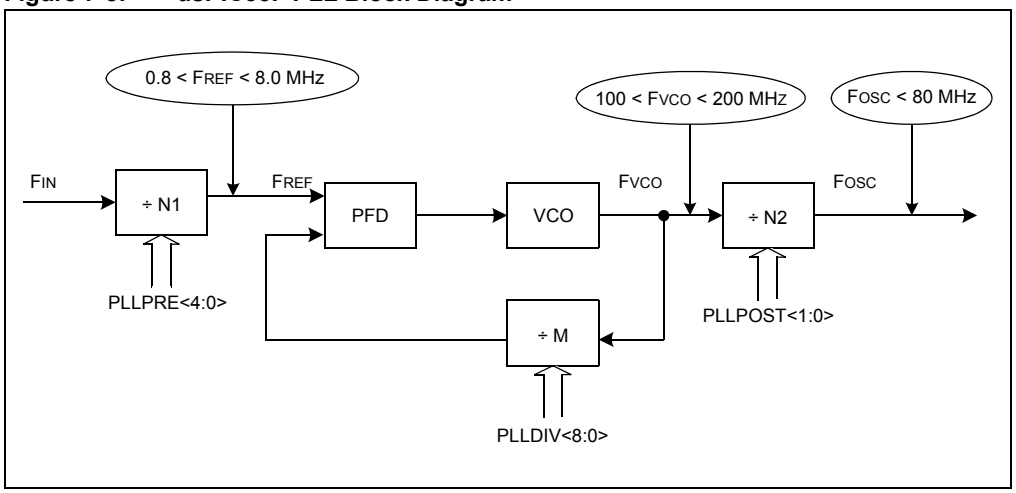
To change the FRC divider:

1. Switch the clock to non-PLL mode (for example, Internal FRC).
2. Make the necessary changes.
3. Switch the clock back to PLL mode.

7.7 PHASE-LOCKED LOOP

The POSC and Internal FRC Oscillator sources can optionally use an on-chip PLL to obtain higher operating speeds. Figure 7-8 shows a block diagram of the PLL module.

Figure 7-8: dsPIC33F PLL Block Diagram



For a proper PLL operation, the Phase Frequency Detector (PFD) input frequency and Voltage Controlled Oscillator (VCO) output frequency must meet the following requirements:

- The PFD input frequency (FREF) must be in the range of 0.8-8.0 MHz
- The VCO output frequency (FVCO) must be in the range of 100-200 MHz

The PLL Phase Detector Input Divider Select bits (PLLPRE<4:0>) in the Clock Divisor register (CLKDIV<4:0>) specify the input divider ratio (N1), which is used to scale down the input frequency (FIN) to meet the PFD input frequency range of 0.8-8 MHz.

The PLL Feedback Divisor bits (PLLDIV<8:0>) in the PLL Feedback Divisor register (PLLFBD<8:0>) specify the divider ratio (M), which scales down FVCO for feedback to the PFD. FVCO is 'M' times FREF.

The PLL VCO Output Divider Select bits (PLLPOST<1:0>) in the Clock Divisor register (CLKDIV<7:6>) specify the divider ratio (N2) to limit FOSC to 80 MHz.

Equation 7-3 shows the relation between FIN and FOSC.

Equation 7-3: Fosc Calculation

$$FOSC = FIN \times \left(\frac{M}{N1 \times N2}\right) = FIN \times \left(\frac{(PLLDIV + 2)}{(PLLPRE + 2) \times 2(PLLPOST + 1)}\right)$$

Where:

$N1$ = PLLPRE + 2

$N2$ = 2 x (PLLPOST + 1)

M = PLLDIV + 2

Equation 7-4 shows the relation between FIN and FVCO.

Equation 7-4: Fvco Calculation

$$FVCO = FIN \times \left(\frac{M}{N1}\right) = FIN \times \left(\frac{(PLLDIV + 2)}{(PLLPRE + 2)}\right)$$

7.7.1 Input Clock Limitation at Start-up for a PLL Mode

Table 7-6 lists the default values of the PLL Prescaler, PLL Postscaler, and PLL Feedback Divisor Configuration bits at a POR.

Table 7-6: PLL Mode Defaults

Register	Bit Field	Value at POR Reset	PLL Divider Ratio
CLKDIV<4:0>	PLLPRE<4:0>	00	N1 = 2
CLKDIV<7:6>	PLLPOST<1:0>	01	N2 = 4
PLLFBD<8:0>	PLLDIV<8:0>	000110000	M = 50

Given these Reset values, the following equations show the relationship between FIN and FREF, FVCO, and FOSC at a POR.

Equation 7-5: FREF at a POR

$$FREF = FIN\left(\frac{1}{N1}\right) = 0.5(FIN)$$

Equation 7-6: Fvco at a POR

$$FVCO = FIN\left(\frac{M}{N1}\right) = FIN\left(\frac{50}{2}\right) = 25(FIN)$$

Equation 7-7: Fosc at a POR

$$FOSC = FIN\left(\frac{M}{N1 \cdot N2}\right) = 6.25(FIN)$$

Given the preceding equations, the FIN to the PLL module must be limited to 4 MHz < FIN < 8 MHz to comply with the FVCO requirement (100M < Fvco < 200M), if the default values of PLLPRE, PLLPOST, and PLLDIV are used.

The POSC can support the following input frequency ranges, which are not within the frequency limit required (4 MHz < FIN< 8 MHz) at a POR.

- POSC in XT mode supports: 3-10 MHz crystal
- POSC in HS mode supports: 10-40 MHz crystal
- POSC in EC mode supports: 0.8-64 MHz input

To use a PLL when the input frequency is not within the 4-8 MHz range, you must follow this process:

1. Power-up the device with the Internal FRC Oscillator, or the POSC, without a PLL.
2. Change PLLDIV, PLLPRE, and PLLPOST bit values, based on the input frequency, to meet these PLL requirements:
   - FREF must be in the range of 0.8-8.0 MHz
   - FVCO must be in the range of 100-200 MHz
3. Switch the clock to a PLL mode in software.

7.7.2 PLL Lock Status

Whenever the PLL input frequency, the PLL prescaler, or the PLL feedback divisor, is changed, the PLL requires a finite amount of time (TLOCK) to synchronize to the new settings.

TLOCK is applied when the PLL is selected as the clock source at a POR, or during a clock switching operation. The value of TLOCK is relative to the time at which the clock is available to the PLL input. For example, with the POSC, TLOCK starts after the OST delay. Refer to **7.5.1 "Oscillator Start-up Time"** for more information about oscillator start-up delay. Refer to the specific device data sheet for information about typical TLOCK values.

The LOCK bit in the Oscillator Control register (OSCCON<5>) is a read-only Status bit that indicates the lock status of the PLL. The LOCK bit is cleared at a POR, and on a clock switch operation, if the PLL is selected as the destination clock source. The LOCK bit remains clear when any clock source that is not using a PLL is selected. After a clock switch event in which a PLL is enabled, it is a good practice to wait for the LOCK bit to be set before executing other code.

Note:	PLLPRE bits and PLLDIV bits should not be changed when operating in the PLL mode. You must clock switch to the non-PLL mode (e.g., Internal FRC) to make the necessary changes, and then clock switch back to the PLL mode.

7.7.2.1 SETUP FOR USING THE PLL WITH THE POSC

The following process is used to set up the PLL to operate the device at 40 MIPS with a 10 MHz external crystal:

1. To execute instructions at 40 MHz, ensure that the required system clock frequency is:

   FOSC = 2 x FCY = 80 MHz
2. Ensure that the default Reset values of PLLPRE, PLLPOST and PLLDIV meet the PLL and user requirements.
3. If the PLL and user requirements are met – directly configure the FNOSC<2:0> bits (FOSCSEL<2:0>) to use the POSC with PLL at a POR.

   If the PLL and user requirements are not met – follow these steps:

   a) Select the PLL postscaler to meet the VCO output frequency requirement (100 < FVCO < 200 MHz).
   b) Select the PLL prescaler to meet the PFD input frequency requirement (0.8 < FREF < 8 MHz).
   c) Select the PLL feedback divisor to generate the required VCO output frequency based on the PFD input frequency.
   d) Configure the FNOSC<2:0> bits (FOSCSEL<2:0>) to select a clock source without the PLL (e.g., Internal FRC) at a POR.
   e) In the main program, change the PLL prescaler, PLL postscaler, and PLL feedback divisor values to the values derived in the previous steps, and then perform a clock switch to the PLL mode.

Example 7-1 illustrates code for using the PLL with the POSC. (See also **Section 7.11 "Clock Switching"** for clock switching example code.)

Example 7-1: Code Example for Using the PLL with the Posc

```
// Select Internal FRC at POR
_FOSCSEL(FNOSC_FRC);

// Enable Clock Switching and Configure Posc in XT mode
_FOSC(FCKSM_CSECMD & OSCIOFNC_OFF & POSCMD_XT);

int main()
{

// Configure PLL prescaler, PLL postscaler, PLL divisor
PLLFBD=30;             // M = 32
CLKDIVbits.PLLPOST = 0;// N2 = 2
CLKDIVbits.PLLPRE = 0; // N1 = 2

// Initiate Clock Switch to Primary Oscillator with PLL (NOSC = 0b011)
__builtin_write_OSCCONH(0x03);
__builtin_write_OSCCONL(0x01);

// Wait for Clock switch to occur
while (OSCCONbits.COSC! = 0b011);

// Wait for PLL to lock
while(OSCCONbits.LOCK! = 1) {};

}
```


7.7.2.2 SETUP FOR USING THE PLL WITH 7.37 MHz INTERNAL FRC

The following process is used to set up the PLL to operate the device at 40 MIPS with a 7.37 MHz Internal FRC.

1. To execute instruction at 40 MHz, ensure that the system clock frequency is:

   FOSC = 2 x FCY = 80 MHz

2. Ensure that the default Reset values of PLLPRE, PLLPOST and PLLDIV meet the PLL and user requirements.
3. If the PLL and user requirements are met – directly configure the FNOSC<2:0> bits (FOSCSEL<2:0>) to use the POSC with PLL at a POR.

   If the PLL and user requirements are not met – follow these steps:

   a) Select the PLL postscaler to meet VCO output frequency requirement (100 < FVCO < 200 MHz).
   b) Select the PLL prescaler to meet PFD input frequency requirement (0.8 < FREF < 8 MHz).
   c) Select the PLL feedback divisor to generate required VCO output frequency based on the PFD input frequency.
   d) Configure the FNOSC<2:0> bits (FOSCSEL<2:0>) to select a clock source without the PLL (e.g., Internal FRC) at a POR.
   e) In the main program, change the PLL prescaler, PLL postscaler and PLL feedback divisor to meet the PLL and user requirements, and then perform a clock switch to the PLL mode.

Example 7-2 illustrates code for using the PLL with a 7.37 MHz Internal FRC. (See also **7.11 "Clock Switching"** for clock switching example code.)

Example 7-2: Code Example for Using the PLL with 7.37 MHz Internal FRC

```
// Select Internal FRC at POR
_FOSCSEL(FNOSC_FRC);

// Enable Clock Switching and Configure
_FOSC(FCKSM_CSECMD & OSCIOFNC_OFF);

int main()
{

// Configure PLL prescaler, PLL postscaler, PLL divisor
PLLFBD = 41;           // M = 43
CLKDIVbits.PLLPOST=0;  // N2 = 2
CLKDIVbits.PLLPRE=0;   // N1 = 2

// Initiate Clock Switch to Internal FRC with PLL (NOSC = 0b001)
__builtin_write_OSCCONH(0x01);
__builtin_write_OSCCONL(0x01);

// Wait for Clock switch to occur
while (OSCCONbits.COSC! = 0b001);

// Wait for PLL to lock
while(OSCCONbits.LOCK! = 1) {};

}
```

7.8 SECONDARY OSCILLATOR (SOSC)

The Secondary Oscillator (SOSC) enables a 32.768 kHz crystal oscillator to be attached to the dsPIC33F device as a secondary crystal clock source for low-power operation. It uses the SOSCI and SOSCO pins. The SOSC can also drive Timer1 for Real-Time Clock (RTC) applications.

Note: The SOSC is sometimes referred to as the Low-Power Secondary Oscillator due to its low-power capabilities. However, this oscillator should not be confused with the LPRC Oscillator.

7.8.1 Secondary Oscillator for System Clock

The SOSC is enabled as the system clock when:

- Initial Oscillator Source Selection Configuration bits (FNOSC<2:0>) in the Oscillator Source Selection register (FOSCSEL<2:0>) are appropriately set to select the SOSC at a POR
- User application initiates a clock switch to the SOSC for low-power operation

When the SOSC is not being used to provide the system clock, or the device enters Sleep mode, the SOSC is disabled to save power.

7.8.2 Secondary Oscillator Start-up Delay

When the SOSC is enabled, it takes a finite amount of time to start oscillating. Refer to **7.5.1 "Oscillator Start-up Time"** for details.

7.8.3 Continuous Secondary Oscillator Operation

Optionally, you can leave the SOSC running continuously. The SOSC is always enabled if the Secondary Oscillator Enable bit (LPOSCEN) is set in the Oscillator Control register (OSCCON<1>).

There are two reasons to leave the SOSC running.

- First, keeping the SOSC always on allows a fast switch to the 32 kHz system clock for lower-power operation, since returning to the faster main oscillator still requires an oscillator start-up time if it is a crystal type source (see **7.5.1 "Oscillator Start-up Time"**).
- Second, the oscillator should remain on continuously when Timer1 is used as an RTC.

Note: In Sleep mode, all clock sources (the POSC, Internal FRC Oscillator, and LPRC Oscillator) are shut down, with the exception of the SOSC. The SOSC can be active in Sleep mode if the Secondary Oscillator Enable bit (LPOSCEN) is set in the Oscillator Control register (OSCCON<1>).

7.9 LOW-POWER RC (LPRC) OSCILLATOR

The Low-Power RC (LPRC) Oscillator provides a nominal clock frequency of 32 kHz. The LPRC Oscillator is the clock source for the Power-up Timer (PWRT), Watchdog Timer (WDT), and Fail-Safe Clock Monitor (FSCM) circuits. It can also be used to provide a low-frequency clock source option for the device in those applications where power consumption is critical and timing accuracy is not required.

Note:	The clock frequency of the LPRC Oscillator will vary, depending on the device voltage and operating temperature. Refer to the **"Electrical Characteristics"** section in the specific device data sheet for details.

7.9.1 LPRC Oscillator for System Clock

The LPRC Oscillator is selected as the system clock in the following conditions:

- the Initial Oscillator Source Selection bits (FNOSC<2:0>) in the Oscillator Source Selection register (FOSCSEL<2:0>) are appropriately set to select LPRC Oscillator at a POR
- the user software initiates a clock switch to the LPRC Oscillator for low-power operation

7.9.2 Enabling the LPRC Oscillator

The LPRC Oscillator is the clock source for the PWRT, WDT, and FSCM. LPRC Oscillator is enabled at a POR when the Power-on Reset Timer Value Select bits (FPWRT) in the POR Configuration Fuse register (FPOR<2:0>) are set.

The LPRC Oscillator remains enabled in the following conditions:

- the FSCM is enabled
- the WDT is enabled
- the LPRC Oscillator is selected as the system clock

If none of these conditions is true, the LPRC Oscillator shuts off after the PWRT expires. The LPRC Oscillator is shut off in Sleep mode.

Note:	The LPRC runs in Sleep mode only if WDT is enabled. Under all other conditions, the LPRC is disabled in Sleep mode.

7.9.3 LPRC Oscillator Start-up Delay

The LPRC Oscillator starts up instantly; unlike a crystal oscillator, which can take several milliseconds to begin oscillation.

7.10 FAIL-SAFE CLOCK MONITOR (FSCM)

The Fail-Safe Clock Monitor (FSCM) allows the device to continue to operate when an oscillator failure occurs. FSCM function is enabled by programming the Clock Switching Mode Configuration bits (FCKSM<1:0>) in the Oscillator Configuration register (FOSC<7:6>) during device programming. When the FSCM is enabled (FCKSM<1:0> = 00), the LPRC Oscillator runs continuously – except during Sleep state.

The FSCM monitors the system clock. If it does not detect a system clock within a specific period of time (2 ms typically, 4 ms maximum), the FSCM generates a clock failure trap and switches the system clock to the Internal FRC Oscillator. At that point, the user application can either attempt to restart the oscillator or execute a controlled shutdown.

Note:	The FSCM does not wake-up the device if the clock fails while the device is in Sleep mode.

The FSCM module takes the following actions when it switches to the Internal FRC Oscillator:

- Current Oscillator Selection bits COSC<2:0> (OSCCON<14:12>) are loaded with '000' (Internal FRC Oscillator)
- Clock Fail Detect bit CF (OSCCON<3>) is set to indicate the clock failure
- Oscillator Switch Enable Control bit OSWEN (OSCCON<0>) is cleared to cancel any pending clock switches

7.10.1 FSCM Delay

The FSCM monitors the system clock for activity after the system clock is ready and the nominal delay (TFSCM) has elapsed.

The FSCM delay is applied when the FSCM is enabled, and the POSC or SOSC is selected as the system clock.

Note:	Refer to the **"Electrical Characteristics"** section of the specific device data sheet for TFSCM values.

For additional information, refer to **Section 8. "Reset"** (DS70192). The most recent documentation can always be found on the Microchip web site, www.microchip.com.

7.10.2 FSCM and WDT

The FSCM and WDT use the LPRC Oscillator as their time base. In the event of a clock failure, the WDT is unaffected and continues to run on the LPRC Oscillator.

7.11 CLOCK SWITCHING

Clock switching can be initiated as a result of a hardware event or a software request. Typical scenarios include:

- Two-speed start-up sequence on a POR, which initially uses the Internal FRC Oscillator for quick start-up, and then automatically switches to the selected clock source when the clock is ready.
- FSCM automatically switches to the Internal FRC Oscillator on a clock failure.
- User application software requests clock switching by setting the OSWEN bit (OSCCON<0>), causing the hardware to switch to the clock source selected by the NOSC<2:0> bits (OSCCON<10:8>) when the clock is ready.

In each of these cases, the clock switch event assures that the proper make-before-break sequence is executed. That is, the new clock source must be ready before the old clock is deactivated, and code must continue to execute as clock switching occurs.

Some dsPIC33F devices feature the Phase-Locked Loop Enable bit (PLLKEN) in the FWDT Fuse Configuration register (FWDT<5>). Setting this bit will cause the device to wait until the PLL locks before switching to the PLL clock source. When this bit is set to '0', the device will not wait for the PLL lock and will proceed with the clock switch. The default setting for this bit is '1'. Refer to **Section 25. "Device Configuration"** (DS70194) of this Family Reference Manual for more information.

With few limitations, applications are free to switch between any of the 4 clock sources (the POSC, SOSC, FRC, and LPRC) that are under software control at any time. To limit the possible side effects that could result from this flexibility, dsPIC33F devices have a safeguard lock built into the switch process. That is, the OSCCON register is write-protected during clock switching.

7.11.1 Enabling Clock Switching

The Clock Switching Mode Configuration bits (FCKSM<1:0>) in the Oscillator Configuration register (FOSC<7:6>) must be programmed to enable clock switching and the FSCM.

Table 7-7: Configurable Clock Switching Modes

FCKSM<1:0> Values	Clock Switching Configuration	FSCM Configuration
1x	Disabled	Disabled
01	Enabled	Disabled
00	Enabled	Enabled

The first bit determines if clock switching is enabled ('0') or disabled ('1'). The second bit determines if the FSCM is enabled ('0') or disabled ('1'). The FSCM can only be enabled if clock switching is also enabled. If clock switching is disabled ('1'), the value of the second bit is irrelevant.

7.11.2 Clock Switch Sequence

The recommended process for a clock switch is as follows:

1. Read the COSC<2:0> bits (OSCCON<14:12>) to determine the current oscillator source (if this information is relevant to the application).
2. Execute the unlock sequence, allowing a write to the high byte of the OSCCON register.
3. Write the appropriate value to the NOSC<2:0> Control bits (OSCCON<10:8>) for the new oscillator source.
4. Execute the unlock sequence, allowing a write to the low byte of the OSCCON register.
5. Set the OSWEN bit (OSCCON<0>) to initiate the oscillator switch.

After the previous steps are completed, the clock switch logic performs the following steps:

1. The clock switching hardware compares the COSC<2:0> Status bits (OSCCON<14:12>) with the new value of the NOSC<2:0> Control bits (OSCCON<10:8>). If they are the same, the clock switch is a redundant operation. In this case, the OSWEN bit (OSCCON<0>) is cleared automatically and the clock switch is aborted.
2. If a valid clock switch has been initiated, the PLL Lock Status bits (OSCCON<5>) and Clock Fail Status bits (OSCCON<3>) are cleared.
3. The new oscillator is turned on by the hardware (if it is not currently running). If a crystal oscillator (the POSC or SOSC) must be turned on, the hardware waits for TOSCD until the crystal starts oscillating and TOST expires. If the new source uses the PLL, the hardware waits until a PLL lock is detected (OSCCON<5> = 1).
4. The hardware waits for the new clock source to stabilize and then performs the clock switch.
5. The hardware clears the OSWEN bit (OSCCON<0>) to indicate a successful clock transition. In addition, the NOSC<2:0> bit values (OSCCON<10:8>) are transferred to the COSC<2:0> Status bits (OSCCON<14:12>).
6. The old clock source is turned off at this time, with the exception of the LPRC (if the WDT or FSCM is enabled) or the SOSC (if the SOSCEN remains set). The timing of the transition between clock sources in shown in Figure 7-9.

Note 1: Clock switching between the XT, HS and EC Primary Oscillator modes is not possible without reprogramming the device.

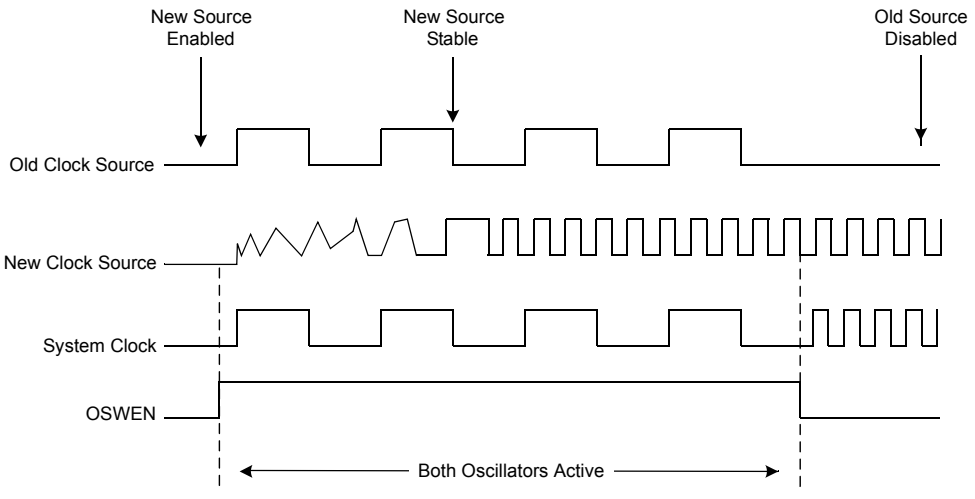
2: Direct clock switching between the PLL modes is not possible. For example, clock switching should not occur between the POSC with PLL and the Internal FRC Oscillator with PLL.

3: Setting the CLKLOCK bit (OSCCON<7>) prevents clock switching when clock switching is enabled and the FSCM is disabled by the Configuration bits FCKSM<1:0> (FOSC<7:6>) = 01. The CLKLOCK bit (OSCCON<7>) cannot be cleared when it has been set by software; it clears on a POR.

4: The processor continues to execute code throughout the clock switching sequence. Timing-sensitive code should not be executed during this time.

5: The clock switch will not wait for the PLL lock if the PLLKEN bit in the FWDT Fuse Configuration register (FWDT<5>) is set to '0'.

Figure 7-9: Clock Transition Timing Diagram



Note: The system clock can be any selected source – POSC, SOSC, FRC or LPRC.

A recommended code sequence for a clock switch includes the following actions:

1. Disable interrupts during the OSCCON register unlock-and-write sequence.
2. Execute the unlock sequence for the OSCCON high byte. In 2 back-to-back instructions:
   - write 0x78 to OSCCON<15:8>
   - write 0x9A to OSCCON<15:8>
3. In the instruction immediately following the unlock sequence, write the new oscillator source to the NOSC<2:0> Control bits (OSCCON<10:8>).
4. Execute the unlock sequence for the OSCCON low byte. In 2 back-to-back instructions:
   - write 0x46 to OSCCON<7:0>
   - write 0x57 to OSCCON<7:0>
5. In the instruction immediately following the unlock sequence, set the OSWEN bit (OSCCON<0>).
6. Continue to execute code that is not clock-sensitive (optional).
7. Check to see if the OSWEN bit (OSCCON<0>) is '0'. If it is, the switch was successful.

Note: MPLAB® C Compiler for dsPIC DSCs provides the following built-in C language functions for unlocking the OSCCON register:

```
__builtin_write_OSCCONL(value)
__builtin_write_OSCCONH(value)
```

See MPLAB C Compiler Help for more information.

Example 7-3 illustrates the code sequence for unlocking the OSCCON register and switching from the FRC with PLL clock source to the LPRC clock source.

Example 7-3: Code Example for Clock Switching

```
;Place the New Oscillator Selection (NOSC=0b101) in W0
MOV #0x15, WREG

;OSCCONH (high byte) Unlock Sequence
MOV #OSCCONH, w1
MOV #0x78, w2
MOV #0x9A, w3
MOV.B w2, [w1] ; Write 0x78
MOV.B w3, [w1] ; Write 0x9A

;Set New Oscillator Selection
MOV.B WREG, OSCCONH

; Place 0x01 in W0 for setting clock switch enabled bit
MOV #0x01, w0

;OSCCONL (low byte) Unlock Sequence
MOV #OSCCONL, w1
MOV #0x46, w2
MOV #0x57, w3
MOV.B w2, [w1] ; Write 0x46
MOV.B w3, [w1] ; Write 0x9A

; Enable Clock Switch
MOV.B w0, [w1] ; Request Clock Switching by Setting OSWEN bit

wait:
    btsc    OSCCONL, #OSWEN
    bra     wait
```

7.11.3 Clock Switching Consideration

When you incorporate clock switching into an application, issues to keep in mind when designing your code include:

- The OSCCON unlock sequence is extremely timing critical. The OSCCON register byte is only writable for 1 instruction cycle following the sequence. Some high-level languages, such as C, may not preserve the timing-sensitive sequence of instructions when compiled. When clock switching is required for an application written in a high-level language, it is best to create the routine in assembler and link it to the application – calling it as a function, when it is required.
- If the destination clock source is a crystal oscillator, the clock switch time will be dominated by the oscillator start-up time.
- If the new clock source does not start, or is not present, clock switching hardware will continue to run from the current clock source. Your software can detect this situation because the OSWEN bit (OSCCON<0>) remains set indefinitely.
- If the new clock source uses a PLL, a clock switch will not occur until lock has been achieved. Your software can detect a loss of a PLL lock because the LOCK bit (OSCCON<5>) is cleared and the OSWEN bit (OSCCON<0>) is set.
- Switching to a low-frequency clock source, such as the secondary oscillator, will result in slow device operation.

7.11.4 Aborting a Clock Switch

If a clock switch does not complete, the clock switch logic can be reset by clearing the OSWEN bit (OSCCON<0>). When OSWEN is cleared, the clock switch process is aborted, OST (if applicable) is stopped and reset, and the PLL (if applicable) is stopped.

Typical assembly code for aborting a clock switch is shown in Example 7-4. A clock switch procedure can be aborted at any time. A clock switch that is already in progress can also be aborted by performing a second clock switch.

Example 7-4: Aborting a Clock Switch

```
MOV     #OSCCON,W1      ; pointer to OSCCON
MOV.b   #0x46,W2        ; first unlock code
MOV.b   #0x57,W3        ; second unlock code
MOV.b   W2, [W1]        ; write first unlock code
MOV.b   W3, [W1]        ; write second unlock code
BCLR    OSCCON,#OSWEN   ; ABORT the switch
```

7.11.5 Entering Sleep Mode During a Clock Switch

If the device enters Sleep mode during a clock switch operation, the clock switch operation is aborted. The processor keeps the old clock selection, and the OSWEN bit is cleared. The `PWRSAV` instruction is then executed normally.

It is particularly useful to perform a clock switch to the Internal FRC Oscillator before entering Sleep mode, as this will ensure fast wake-up from Sleep.

7.12 TWO-SPEED START-UP

The Internal External Start-up Option Configuration bit (IESO) in the Oscillator Source Selection register (FOSCSEL<7>) specifies whether to start the device with a user-selected oscillator source; or to initially start with the Internal FRC Oscillator, and then automatically switch to the user-selected oscillator. If this bit is set to '1', the device will always power-up on the Internal FRC Oscillator, regardless of the other oscillator source settings (FOSCSEL<2:0>). The device, then, automatically switches to the specified oscillator, when it is ready.

Unless FSCM is enabled, the Internal FRC Oscillator is automatically turned off immediately after the clock switch is completed. The Two-Speed Start-up option is a faster way to get the device up and running, and works independently of the state of the Clock Switching Mode bits (FCKSM<1:0>) in Oscillator Configuration register (FOSC<7:6>).

Two-Speed Start-up is particularly useful when an external oscillator is selected by the FNOSC<2:0> Configuration bits (FOSCSEL<2:0>) and a crystal-based oscillator (either a primary or secondary oscillator) has a longer start-up time. As an internal RC oscillator, the FRC clock source is available almost immediately following a POR. With Two-Speed Start-up, the device starts executing code in its default oscillator configuration – FRC. It continues to operate in this mode until the specified external oscillator source becomes stable, at which time it automatically switches to that source.

User code can check which clock source is currently providing the device clocking by checking the status of the COSC<2:0> bits (OSCCON<14:12>) against the NOSC<2:0> bits (OSCCON<10:8>). If these two sets of bits match, the clock switch has been completed successfully and the device is running from the intended clock source.

Note: Two-Speed Start-up is redundant if the selected device clock source is FRC.

7.13 REGISTER MAPS

Table 7-8 maps the bit functions for the Oscillator Special Function Control registers. Table 7-9 maps the bit functions for the Oscillator Configuration registers.

Table 7-8: Oscillator Special Function Control Registers

File Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
OSCCON	—	COSC<2:0>			—	NOSC<2:0>			CLKLOCK	IOLOCK(2)	LOCK	—	CF	—	LPOSCEN	OSWEN	7700(1)
CLKDIV	ROI	DOZE<2:0>			DOZEN	FRCDIV<2:0>			PLLPOST<1:0>		—	PLLPRE<4:0>					3040
PLLFBD	—	—	—	—	—	—	—	PLLDIV<8:0>									0030
OSCTUN	—	—	—	—	—	—	—	—	—	—	TUN<5:0>						0000

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: OSCCON register Reset values are dependent on the FOSCSEL Configuration bits and by type of Reset.

2: The IOLOCK bit is not available on all dsPIC33F devices. Refer to the specific device data sheet for more information.

Table 7-9: Oscillator Configuration Registers

File Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FOSCSEL	—	—	—	—	—	—	—	—	IESO	—	—	—	—	FNOSC<2:0>		
FOSC	—	—	—	—	—	—	—	—	FCKSM<1:0>		IOL1WAY(1)	—	—	OSCIOFNC	POSCMD<1:0>	

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: The IOL1WAY bit is not available on all dsPIC33F devices. Refer to the specific device data sheet for more information.

7.14 RELATED APPLICATION NOTES

This section lists application notes that pertain to this section of the manual. These application notes may not be written specifically for the dsPIC33F Product Family, but the concepts are pertinent and could be used with modification and possible limitations. The current application notes related to the Oscillator module include:

Title	Application Note #
PIC® Microcontroller Oscillator Design Guide	AN588
Low-Power Design using PIC® Microcontrollers	AN606
Crystal Oscillator Basics and Crystal Selection for rfPIC® and PIC® Devices	AN826

Note: Please visit the Microchip web site (www.microchip.com) for additional Application Notes and code examples for the dsPIC33F family of devices.

7.15 REVISION HISTORY

Revision A (January 2007)

This is the initial release of this document.

Revision B (July 2008)

This revision incorporates the following content updates:

- Registers:
  - FOSCSEL: Oscillator Source Selection Register (see Register 7-1): Bit 5 is modified as follows:
    Reserved: Reserved bits must be programmed as '1'
  - OSCTUN: FRC Oscillator Tuning Register (see Register 7-6): Bit 5-0 description is modified as follows:

    TUN<5:0>: FRC Oscillator Tuning bits

    011111 = Center frequency + 11.625% (8.23 MHz)

    011110 = Center frequency + 11.25% (8.20 MHz)
- Tables:
  - Oscillator Special Function Control Registers table (see Table 7-9): Bit 5 register is modified as blank.
- Additional minor corrections such as language and formatting updates have been incorporated throughout the document.

Revision C (December 2008)

This revision incorporates the following content updates:

- Added a new paragraph after the second paragraph that references the use of the Phase-Locked Loop (PLL) Enable (PLLKEN) bit to control clock switching in **Section 7.11 "Clock Switching"**.
- Added Note 5, which references the PLLKEN bit in **Section 7.11.2 "Clock Switch Sequence"**.
- Additional minor corrections such as language and formatting updates have been incorporated throughout the document.

Revision D (August 2009)

This revision incorporates the following content updates:

- Registers:
  - Added Note 2 to Register 7-3
  - Added Note 2 to Register 7-4
- Figures:
  - Updated Figure 7-1
  - Updated Figure 7-1
- Added Note 2 regarding FRC Oscillator Tuning (TUN<5:0>) bits to Internal Fast RC (FRC) Oscillator section (**7.6 "Internal Fast RC (FRC) Oscillator"**)
- Updated code examples (Example 7-1, Example 7-2 and Example 7-3)
- Added register description to Oscillator Configuration Registers section (**7.3 "Oscillator Configuration Registers"**)
- Additional minor corrections such as language and formatting updates have been incorporated throughout the document

NOTES: