
Section 29. Oscillator

HIGHLIGHTS

This section of the manual contains the following topics:

29.1	Introduction	29-2
29.2	CPU Clocking.....	29-4
29.3	Oscillator Configuration Registers	29-5
29.4	Special Function Registers	29-8
29.5	Primary Oscillator.....	29-13
29.6	Internal Fast RC (FRC) Oscillator	29-17
29.7	Phase Lock Loop (PLL)	29-21
29.8	Low-Power RC (LPRC) Oscillator.....	29-22
29.9	Fail-Safe Clock Monitor.....	29-23
29.10	Clock Switching.....	29-24
29.11	Differences Between Industrial and Extended Temperature Range Devices	29-28
29.12	Register Maps.....	29-29
29.13	Related Application Notes.....	29-30
29.14	Revision History	29-31

29.1 Introduction

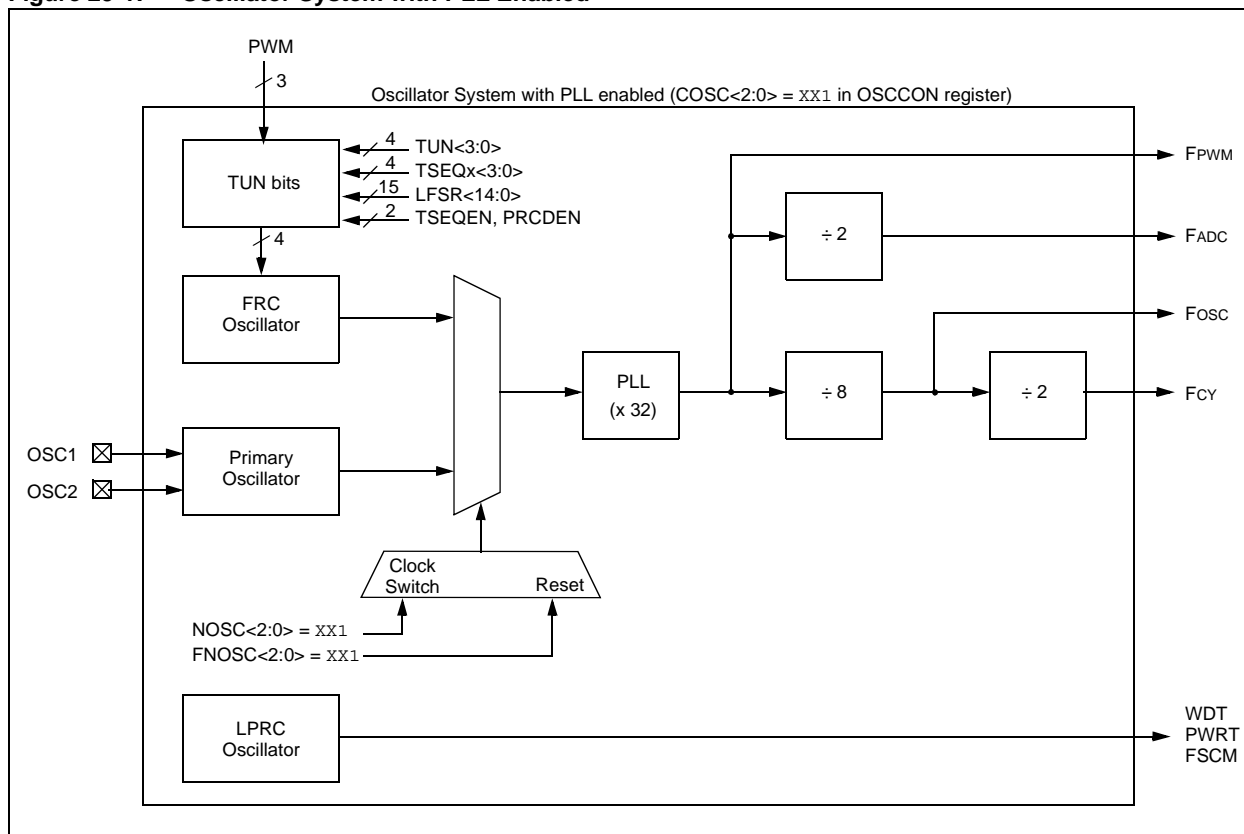
This section describes the operation of the oscillator system for dsPIC30F SMPS devices.

The dsPIC30F SMPS oscillator system includes the following characteristics:

- External and internal oscillators
- On-chip Phase Lock Loop (PLL) to boost internal operating frequency on internal and external oscillator sources
- On-the-fly clock switching between various clock sources
- Fail-Safe Clock Monitor (FSCM) that detects clock failure and permits safe application recovery or shutdown
- Non-volatile configuration bits
- Operates in both industrial and extended temperature ranges, with some exceptions noted in **Section 29.11 “Differences Between Industrial and Extended Temperature Range Devices”**

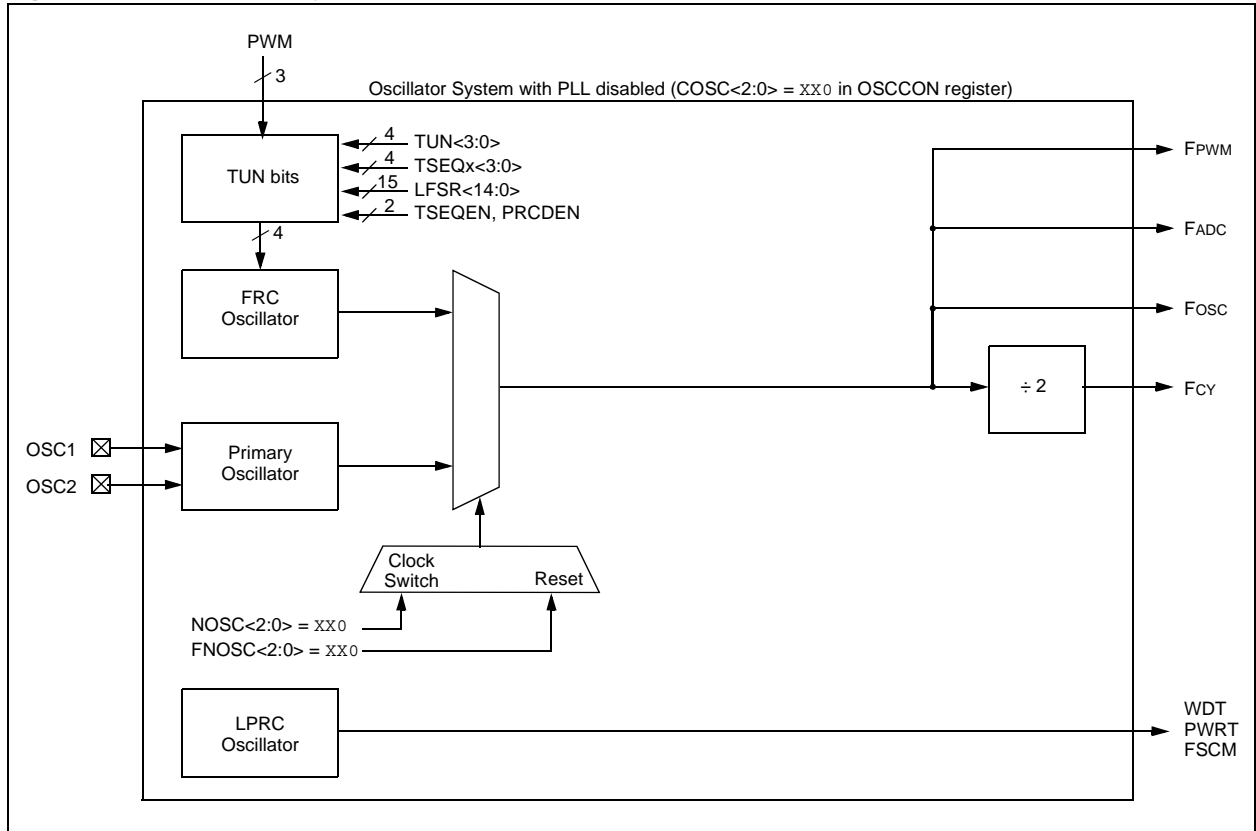
Block diagrams of the dsPIC30F SMPS oscillator system with the PLL enabled and with the PLL disabled are shown in Figure 29-1 and Figure 29-2 respectively.

Figure 29-1: Oscillator System with PLL Enabled



As an example, if the FRC oscillator is selected with a frequency of 14.55 MHz, input to the PLL will be 15 MHz (`TUN<3:0> = 0x7`, see **Section 29.6.2 “FRC Tuning and Dithering”**). The output of the PLL drives the PWM module (FPWM). The PLL output signal is divided by 2 to produce the ADC clock (FADC). Additional dividers are used to produce the instruction clock (Fcy) of 30 MHz/30 MIPS ($\div 8$ and $\div 2$) and the system clock (Fosc) of 60 MHz ($\div 8$).

Figure 29-2: Oscillator System with PLL Disabled



29.2 CPU Clocking

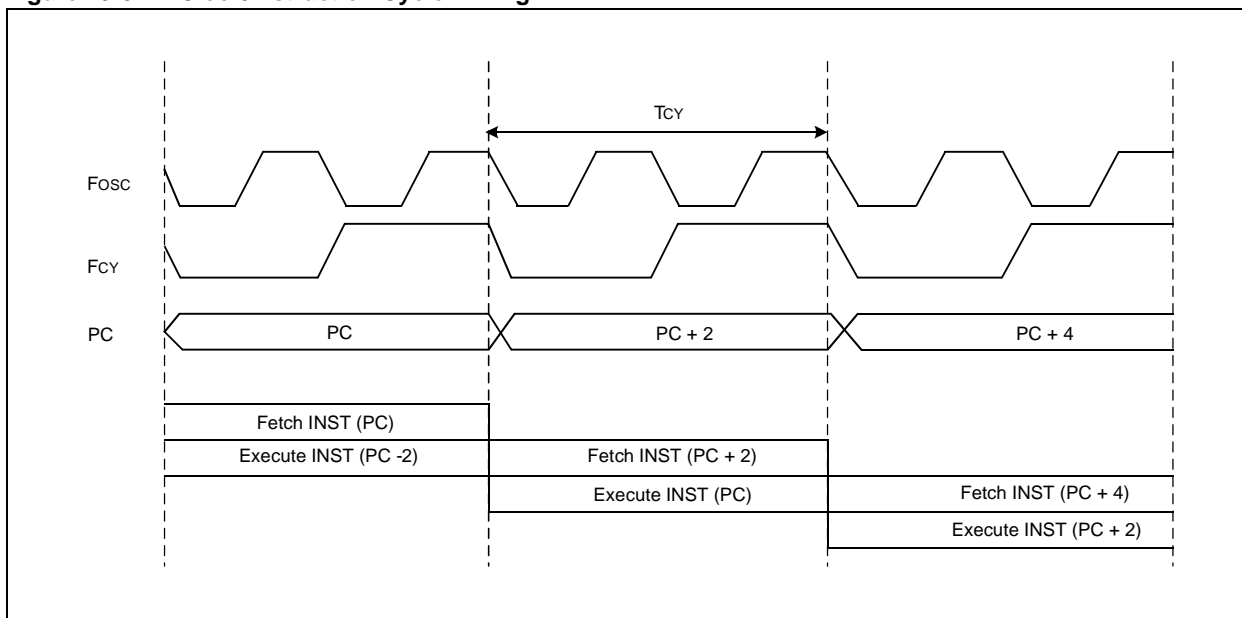
The system clock (F_{osc}) source can be provided by one of the following options:

- Primary Oscillator (Posc) on the OSC1 and OSC2 pins
- Internal Fast RC Oscillator (FRC)
- Primary Oscillator with PLL
- Internal Fast RC Oscillator with PLL

The system clock source is divided by two to produce the internal instruction cycle clock. In this document, the instruction cycle clock is denoted by F_{CY} . The timing diagram in Figure 29-3 shows the relationship between the system clock (F_{osc}), the instruction cycle clock (F_{CY}) and the Program Counter (PC).

F_{CY} can be output on the OSC2 I/O pin if HS mode is not selected as the clock source (see **Section 29.5 “Primary Oscillator”**).

Figure 29-3: Clock/Instruction Cycle Timing



29.3 Oscillator Configuration Registers

Table 29-1 lists the configuration settings that select the device oscillator source and operating mode at Power-on Reset (POR). The configuration bits are contained in the following registers:

FOSCSEL: Oscillator Source Selection Register

FOSC: Oscillator Configuration Register

The configuration registers are located in program memory space. These are not Special Function Registers. They are mapped into program memory space and programmed at the time of device programming.

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

The POSCMD<1:0> configuration bits in the Oscillator Configuration (FOSC) register select the operation mode of the primary oscillator.

The OSCIOFNC configuration bit in the FOSC register selects the OSC2 pin function, except in HS mode. When OSCIOFNC is unprogrammed ('1'), FCY clock is output on the OSC2 pin. When OSCIOFNC is programmed ('0'), the OSC2 pin becomes a general purpose I/O pin.

Table 29-1: Configuration Bit Values for Clock Selection

Oscillator Mode	FNOSC<2:0>	POSCMD<1:0>
Fast RC Oscillator (FRC)	000	xx
Fast RC Oscillator with PLL (FRCPLL)	001	xx
Primary Oscillator (EC)	010	00
Primary Oscillator (HS)	010	10
Primary Oscillator with PLL (ECPLL)	011	00
Primary Oscillator with PLL (HSPLL)	011	10

dsPIC30F Family Reference Manual

Register 29-1: FOSCSEL: Oscillator Source Selection Register

Upper Byte:							
U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 23						bit 16	

Middle Byte:							
U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

Lower Byte:								
U-0	U-0	U-0	U-0	U-0	R-P	R-P	R-P	
—	—	—	—	—	FNOSC<2:0>			
bit 7						bit 0		

bit 23-3 **Unimplemented:** Read as '0'

bit 2-0 **FNOSC<2:0>:** Initial Oscillator Source Selection bits

- 000 = Fast RC Oscillator (FRC)
- 001 = Fast RC Oscillator with PLL (FRCPLL)
- 010 = Primary Oscillator (HS, EC)
- 011 = Primary Oscillator with PLL (HSPLL, ECPLL)
- 100 = Reserved
- 101 = Reserved
- 110 = Reserved
- 111 = Reserved

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

Register 29-2: FOSC: Oscillator Configuration Register

Upper Byte:							
U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 23				bit 16			

Middle Byte:							
U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15				bit 8			

Lower Byte:								
R-P	R-P	R-P	U-0	U-0	R-P	R-P	R-P	
FCKSM<1:0>		FRANGE	—	—	OSCIOFNC	POSCMD<1:0>		
bit 7			bit 0					

bit 23-8 **Unimplemented:** Read as '0'

bit 7-6 **FCKSM<1:0>:** Clock Switching and Monitor Selection Configuration bits

1x = Clock switching is disabled, fail-safe clock monitor is disabled

01 = Clock switching is enabled, fail-safe clock monitor is disabled

00 = Clock switching is enabled, fail-safe clock monitor is enabled

bit 5 **FRANGE:** Frequency Range Select for FRC and PLL bit

Acts like a “Gear Shift” feature that enables the dsPIC30F SMPS device to operate at reduced MIPS at a reduced supply voltage (3.3V)

FRANGE Bit Value	Temperature Rating	FRC Frequency (Nominal)
1 = High Range	Industrial	14.55 MHz
	Extended	9.7 MHz
0 = Low Range	Industrial	9.7 MHz
	Extended	6.4 MHz

bit 4-3 **Unimplemented:** Read as '0'

bit 2 **OSCIOFNC:** OSC2 Pin I/O Enable bit

1 = CLKO output signal active on the OSC2 pin

0 = CLKO output disabled

bit 1-0 **POSCMD<1:0>:** Primary Oscillator Mode

11 = Primary Oscillator disabled

10 = HS Oscillator mode selected

01 = Reserved

00 = EC External Clock mode selected

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

29.4 Special Function Registers

The following Special Function Registers provide run-time control and status of the oscillator system:

- **OSCCON: Oscillator Control Register**

The Oscillator Control (OSCCON) register controls clock switching and provides status information that allows current clock source, PLL lock and clock fail conditions to be monitored.

- **OSCTUN: FRC Oscillator Tuning Register**

The FRC Oscillator Tuning (OSCTUN) register is used to tune the internal FRC oscillator frequency in software. See **Section 29.6.2 “FRC Tuning and Dithering”** for more details.

- **OSCTUN2: FRC Oscillator Tuning Register 2**

The FRC Oscillator Tuning (OSCTUN2) register 2 is a continuation of the OSCTUN register and provides four additional 4-bit tune values for tuning the internal FRC oscillator frequency in software. See **Section 29.6.2 “FRC Tuning and Dithering”** for more details.

- **LFSR: Linear Feedback Shift Register**

The Linear Feedback Shift (LFSR) register provides a seed number for a pseudo-random number generation algorithm. See **Section 29.6.2 “FRC Tuning and Dithering”** for more details.

<p>Note: The Oscillator Special Function Registers (OSCCON, OSCTUN and OSCTUN2) are reset only on Power-on Reset (POR).</p>
--

Register 29-3: OSCCON: Oscillator Control Register

Upper Byte:							
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

Lower Byte:							
R/W-0	U-0	R-0, HS, HC	R/W-0	R/C-0, HS, HC	R/W-0	U-0	R/W-0, HC
CLKLOCK	—	LOCK	PRCDEN	CF	TSEQEN	—	OSWEN
bit 7							bit 0

- bit 15 **Unimplemented:** Read as '0'
- bit 14-12 **COSC<2:0>:** Current Oscillator Selection bits (read-only)
 - 000 = Fast RC Oscillator (FRC)
 - 001 = Fast RC Oscillator with PLL Module (FRCPLL)
 - 010 = Primary Oscillator (HS, EC)
 - 011 = Primary Oscillator with PLL Module (HSPLL, ECPLL)
 - 100 = Reserved
 - 101 = Reserved
 - 110 = Reserved
 - 111 = Reserved
 - Note 1:** These bits are loaded with the FNOSC<2:0> value on POR or BOR.
 - 2:** These bits are set to '000' if the FSCM detects a failure and switches to FRC.
 - 3:** These bits are loaded with the NOSC<2:0> value at the completion of a successful clock switch.
- bit 11 **Unimplemented:** Read as '0'
- bit 10-8 **NOSC<2:0>:** New Oscillator Selection bits
 - 000 = Fast RC Oscillator (FRC)
 - 001 = Fast RC Oscillator with PLL Module (FRCPLL)
 - 010 = Primary Oscillator (HS, EC)
 - 011 = Primary Oscillator with PLL Module (HSPLL, ECPLL)
 - 100 = Reserved
 - 101 = Reserved
 - 110 = Reserved
 - 111 = Reserved
- bit 7 **CLKLOCK:** Clock Lock Enable bit
 - 1 = If (FCKSM1 = 1), then clock and PLL configurations are locked
 - If (FCKSM1 = 0), then clock and PLL configurations may be modified
 - 0 = Clock and PLL selection are not locked, configurations may be modified
 - Note:** Once set, this bit can only be cleared by a Reset.
- bit 6 **Unimplemented:** Read as '0'
- bit 5 **LOCK:** PLL Lock Status bit (read-only)
 - 1 = Indicates that PLL is in lock
 - 0 = Indicates that PLL is out of lock (or disabled)
- bit 4 **PRCDEN:** Pseudo-Random Clock Dither Enable bit
 - 1 = Pseudo-random clock dither is enabled
 - 0 = Pseudo-random clock dither is disabled
- bit 3 **CF:** Clock Fail Detect bit (read/clear by application)
 - 1 = FSCM has detected clock failure
 - 0 = FSCM has not detected clock failure
- bit 2 **TSEQEN:** FRC Tune Sequencer Enable bit
 - 1 = The TUN<3:0>, TSEQ1<3:0>, ..., TSEQ7<3:0> bits in the OSCTUN and OSCTUN2 registers sequentially tune the FRC oscillator
 - 0 = The TUN<3:0> bits in the OSCTUN register tune the FRC oscillator

dsPIC30F Family Reference Manual

Register 29-3: OSCCON: Oscillator Control Register (Continued)

bit 1 **Unimplemented:** Read as '0'

bit 0 **OSWEN:** Oscillator Switch Enable bit

1 = Request oscillator switch to selection specified by NOSC<2:0> bits

0 = Oscillator switch is complete

Note: This register is write-protected and must be unlocked before it can be written. See **Section 29.6.2 “FRC Tuning and Dithering”** and **Section 29.10.2 “Clock Switch Sequence”** for more details.

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

Register 29-4: OSCTUN: FRC Oscillator Tuning Register

Upper Byte:							
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TSEQ3<3:0>				TSEQ2<3:0>			
bit 15				bit 8			

Lower Byte:							
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TSEQ1<3:0>				TUN<3:0>			
bit 7				bit 0			

- bit 15-12 **TSEQ3<3:0>**: Tune Sequence Value #3 bits
- bit 11-8 **TSEQ2<3:0>**: Tune Sequence Value #2 bits
- bit 7-4 **TSEQ1<3:0>**: Tune Sequence Value #1 bits
- bit 3-0 **TUN<3:0>**: FRC Oscillator Tuning bits
 - 0111 = Maximum frequency
 - 0110 =
 - 0101 =
 - 0100 =
 - 0011 =
 - 0010 =
 - 0001 =
 - 0000 = Center frequency, oscillator is running at calibrated frequency
 - 1111 =
 - 1110 =
 - 1101 =
 - 1100 =
 - 1011 =
 - 1010 =
 - 1001 =
 - 1000 = Minimum frequency

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

dsPIC30F Family Reference Manual

Register 29-5: OSCTUN2: FRC Oscillator Tuning Register 2

Upper Byte:							
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TSEQ7<3:0>				TSEQ6<3:0>			
bit 15				bit 8			

Lower Byte:							
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TSEQ5<3:0>				TSEQ4<3:0>			
bit 7				bit 0			

bit 15-12 **TSEQ7<3:0>**: Tune Sequence Value #7 bits

bit 11-8 **TSEQ6<3:0>**: Tune Sequence Value #6 bits

bit 7-4 **TSEQ5<3:0>**: Tune Sequence Value #5 bits

bit 3-0 **TSEQ4<3:0>**: Tune Sequence Value #4 bits

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

Register 29-6: LFSR: Linear Feedback Shift Register

Upper Byte:							
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	LFSR<14:8>						
bit 15				bit 8			

Lower Byte:							
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LFSR<7:0>							
bit 7				bit 0			

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

bit 14-8 **LFSR<14:0>**: Most Significant 7 bits of the pseudo-random FRC trim value bits

bit 7-0 **LFSR<7:0>**: Least Significant 8 bits of the pseudo-random FRC trim value bits

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

29.5 PRIMARY OSCILLATOR

The Primary Oscillator is available on the OSC1 and OSC2 pins of the dsPIC30F SMPS device family. This connection enables an external crystal (or ceramic resonator) to provide the clock to the device. It can optionally be used with the internal PLL to boost the system frequency (FOSC) to 60 MHz for 30 MIPS execution. The primary oscillator provides two modes of operation.

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

The HS mode is a high-gain, high-frequency mode used to work with crystal frequencies of 6 to 15 MHz.

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

If the on-chip oscillator is not used, the EC mode allows the internal oscillator to be bypassed. The device clocks are generated from an external source (6 to 15 MHz) and input on the OSC1 pin.

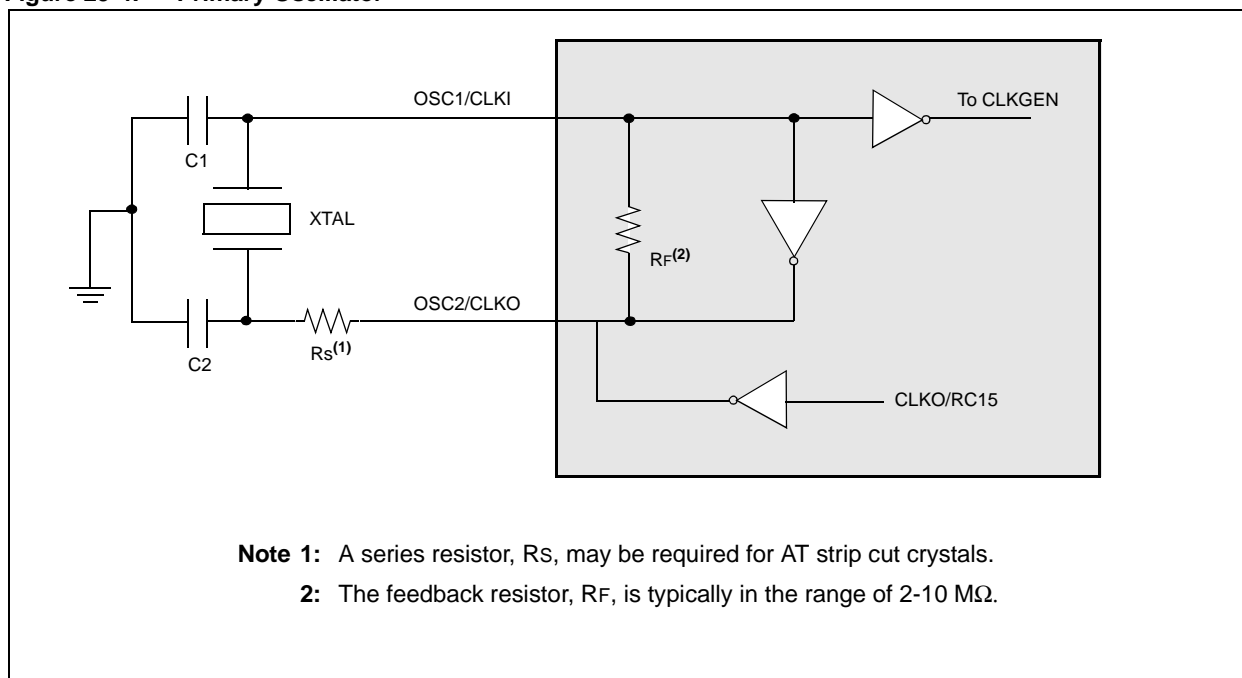
The FNOSC<2:0> configuration bits in the Oscillator Source Selection (FOSCSEL) register specify the primary oscillator clock source at Power-on Reset. The POSCMD<1:0> configuration bits in the Oscillator Configuration (FOSC) register specify the primary oscillator mode. Table 29-2 shows the options selected by specific bit configurations, which are programmed at the time of device programming.

Table 29-2: Primary Oscillator Clock Source Options

FNOSC Value	POSCMD	Primary Oscillator Source/Mode
010	00	Primary Oscillator: External Clock Mode (EC)
010	10	Primary Oscillator: High Frequency Mode (HS)
011	00	Primary Oscillator with PLL: External Clock Mode (ECPLL)
011	10	Primary Oscillator with PLL: High Frequency Mode (HSPLL)

Figure 29-4 provides a circuit diagram for a recommended crystal oscillator configuration for dsPIC30F SMPS devices. Capacitors C1 and C2 form the load capacitance for the crystal. The optimum load capacitance (CL) for a given crystal is specified by the crystal manufacturer. Load capacitance can be calculated as shown in Equation 29-1.

Figure 29-4: Primary Oscillator



Equation 29-1: Crystal Load Capacitance

$$C_L = C_S + \frac{C_1 \times C_2}{C_1 + C_2}$$

C_S is the stray capacitance.

Assuming $C_1 = C_2$, Equation 29-2 gives the capacitor value (C_1 , C_2) for a given load and stray capacitance.

Equation 29-2: External Capacitor for Crystal

$$C_1 = C_2 = 2 \times (C_L - C_S)$$

For additional information on crystal oscillators and their operation, refer to **Section 29.13 “Related Application Notes”**.

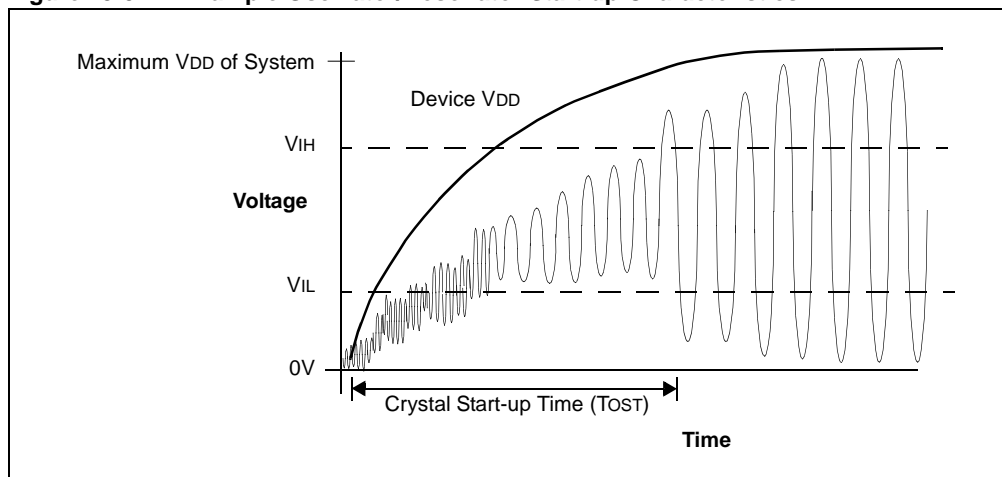
29.5.1 Oscillator Start-up Time

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, including the following:

- Crystal/resonator frequency
- Capacitor values used (C_1 and C_2 in Figure 29-4)
- Device V_{DD} rise time
- System temperature
- Series resistor value and type, if used (R_S in Figure 29-4)
- Oscillator mode selection of device (selects the gain of the internal oscillator inverter)
- Crystal quality
- Oscillator circuit layout
- System noise

Figure 29-5 shows a plot of a typical oscillator/resonator start-up.

Figure 29-5: Example Oscillator/Resonator Start-up Characteristics



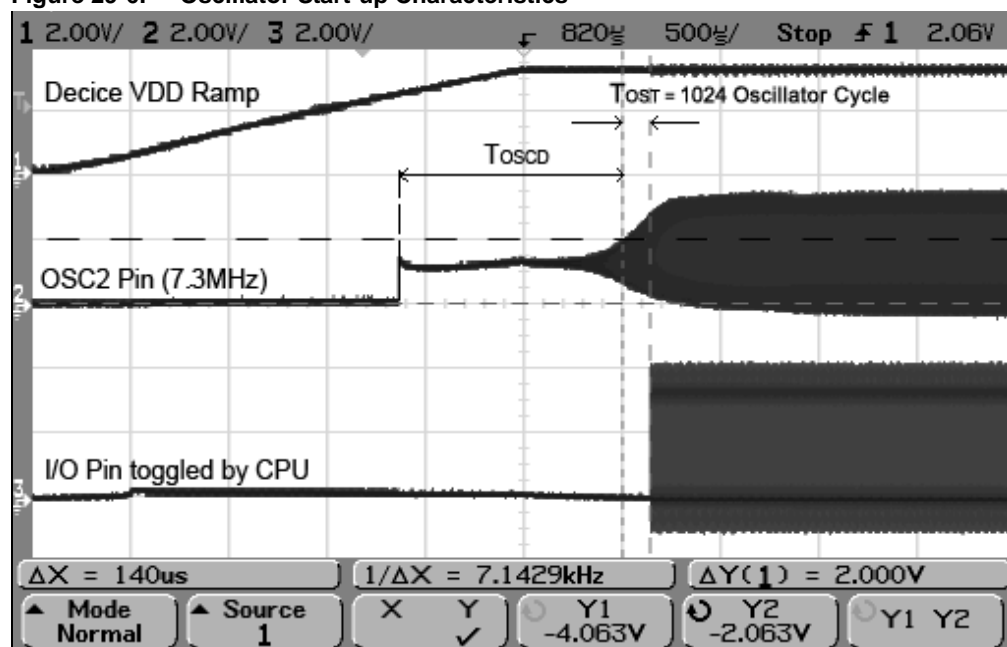
To ensure that a crystal oscillator (or ceramic resonator) has started and stabilized, an Oscillator Start-up Timer (OST) is provided with the Primary Oscillator (POSC). The OST is a simple 10-bit counter that counts 1024 cycles before releasing the oscillator clock to the rest of the system. This time-out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

The amplitude of the oscillator signal must reach the V_{IL} and V_{IH} thresholds for the oscillator pins before the OST can begin to count cycles. The T_{OST} interval is required every time the oscillator restarts (i.e., on POR, BOR and Wake-up from Sleep mode).

Once the Primary Oscillator is enabled, it takes a finite amount of time to start oscillating. This delay is denoted as T_{OSCD} . After T_{OSCD} , the OST timer takes 1024 clock cycles (T_{OST}) to release the clock. The total delay for the clock to be ready is $T_{OSCD} + T_{OST}$. If the PLL is used, an additional delay is required for the PLL to lock (see **Section 29.7 “Phase Lock Loop (PLL)”**).

Primary Oscillator start-up behavior is illustrated in Figure 29-6, where the CPU starts toggling an I/O pin when it starts execution after the $T_{OSCD} + T_{OST}$ interval.

Figure 29-6: Oscillator Start-up Characteristics



29.5.2 Primary Oscillator Pin Functionality

The Primary Oscillator pins (OSC1/OSC2) can be used for other functions when the oscillator is not being used. OSC1 pin function is determined by the $POSCMD<1:0>$ configuration bits in the Oscillator Configuration (FOSC<1:0>) register. OSC2 pin function is determined by both the $POSCMD<1:0>$ and $OSCIOfNC$ configuration bits in the Oscillator Configuration (FOSC) register.

POSCMD: Primary Oscillator Mode Selection bits

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

OSCIOfNC: OSC2 Pin Function bit (except in HS mode)

- 1 = OSC2 is the clock output, and instruction cycle (F_{CY}) clock is output on OSC2 pin (see Figure 29-7)
- 0 = OSC2 is a general purpose digital I/O pin (see Figure 29-8)

The oscillator pin functions are given in Table 29-3.

dsPIC30F Family Reference Manual

Table 29-3: Clock PIN Function Selection

Oscillator Source	OSCIOFNC Value	POSCMD<1:0> Value	OSC1 ⁽¹⁾ Pin Function	OSC2 ⁽²⁾ Pin Function
Primary OSC Disabled	1	11	Digital I/O	Clock Output (Fcy)
Primary OSC Disabled	0	11	Digital I/O	Digital I/O
HS	X	10	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 (POSCMOD<1:0>) configuration bits.
Note 2: OSC2 pin function is determined by the Primary Oscillator Mode (POSCMOD<1:0>) and OSC2 Pin Function (OSCIOFNC) configuration bits.

Figure 29-7: OSC2 Pin Configured for Clock Output (in EC Mode)

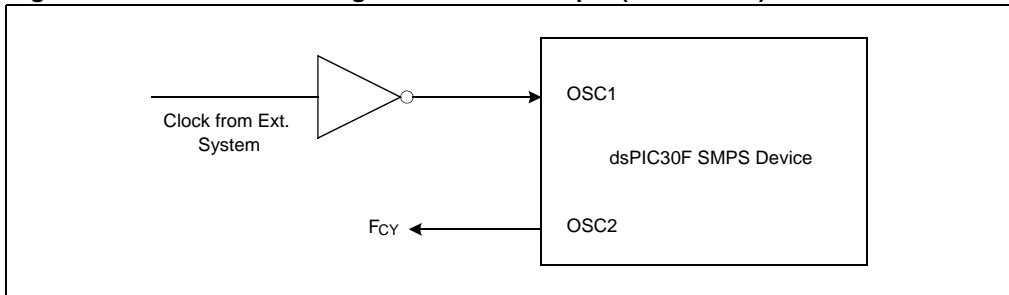
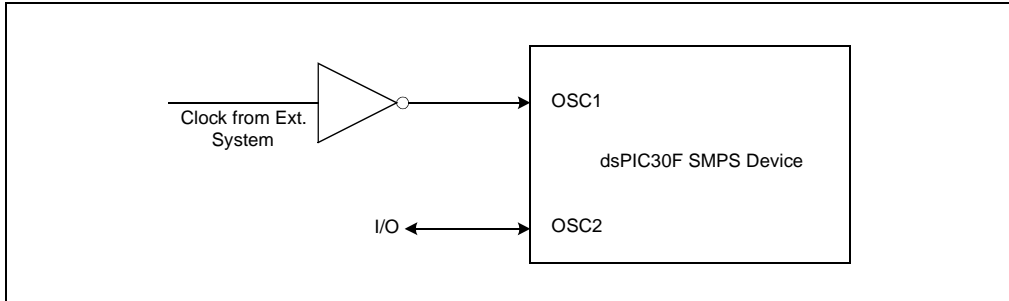


Figure 29-8: OSC2 Pin Configured for Digital I/O (in EC Mode)



29.6 Internal Fast RC (FRC) Oscillator

The Internal Fast RC (FRC) Oscillator provides a nominal 9.7 or 14.55 MHz ($\pm 2\%$ accuracy) clock without requiring an external crystal or ceramic resonator. The FRC oscillator option is intended to be accurate enough to provide the clock frequency necessary to maintain baud rate tolerance for serial data transmissions.

Note: Nominal FRC frequency differs between industrial and extended temperature dsPIC30F SMPS devices. See **Section 29.11 “Differences Between Industrial and Extended Temperature Range Devices”**.

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

The Initial Oscillator Source Selection (FNOSC<2:0>) configuration bits in the Oscillator Source Selection (FOSCSEL<2:0>) register select the FRC clock source. FRC Clock source options available at the time of Power-on Reset are shown in Table 29-4 below. The configuration bits are programmed at the time of device programming.

Table 29-4: FRC Clock Source Options

FNOSC<2:0> Value	Primary Oscillator Source/Mode
000	FRC Oscillator (FRC)
001	FRC Oscillator with PLL (FRCPLL)

29.6.1 Frequency Range Selection

The FRC module has a “Gear Shift” control signal that selects “Low Range” (9.7 MHz) or “High Range” (14.55 MHz) frequency of operation. This feature enables a dsPIC30F SMPS device to operate at 3.3V or 5.0V at 20 or 30 MIPS and remain within system specifications.

Setting the Frequency Range Select bit (FRANGE) in the Oscillator Configuration (FOSC) register to '1' configures the FRC for 14.55 MHz operation when a 5.0V supply voltage is used.

Setting FRANGE = 1 when a 3.3V supply voltage is used is invalid operation and must be avoided.

Setting FRANGE = 0 configures FRC for 9.7 MHz operation for both 3.3V and 5.0V supply voltages.

29.6.2 FRC Tuning and Dithering

The application software can tune the frequency of the Internal Fast RC (FRC) from +3% to -3% of the nominal frequency value using the TUN<3:0> or TSEQx<3:0> bits in the FRC Oscillator Tuning (OSCTUN) and FRC Oscillator Tuning 2 (OSCTUN2) registers. The nominal or tuned frequency of the FRC Oscillator is expected to remain within $\pm 2\%$ of the tuned value over the temperature and voltage variations of a particular device. Table 29-5 illustrates how the tuning bits affect the FRC frequency for 14.55 MHz configuration.

Table 29-5: Effect of Tuning Bits for FRC Frequency of 14.55 MHz

Tuning Bits (TUN<3:0> or TSEQx<3:0>)	FRC Output ⁽¹⁾
0111	14.55 MHz + 3% = 15 MHz
0110	14.55 MHz + 2.57%
0101	14.55 MHz + 2.14%
0100	14.55 MHz + 1.71%
0011	14.55 MHz + 1.29%
0010	14.55 MHz + 0.86%
0001	14.55 MHz + 0.43%
0000	14.55 MHz (Center frequency, FRC running at calibrated frequency)
1111	14.55 MHz – 0.375%
1110	14.55 MHz – 0.75%
1101	14.55 MHz – 1.125%
1100	14.55 MHz – 1.5%
1011	14.55 MHz – 1.875%
1010	14.55 MHz – 2.25%
1001	14.55 MHz – 2.625%
1000	14.55 MHz – 3% = 14.1 MHz

Note 1: Nominal FRC frequency differs between industrial and extended temperature dsPIC30F SMPS devices. See **Section 29.11 “Differences Between Industrial and Extended Temperature Range Devices”** for more information.

To minimize EMI emission, FRC tuning functionality can be used to dither (change) the FRC frequency with random or user defined patterns. The FRC Tune Sequencer Enable (TSEQEN) bit and Pseudo-Random Clock Dither Enable (PRCDEN) bit in the Oscillator Control (OSCCON) register control whether the FRC will dither (and if so, the type of dither) or tune to a single frequency. Table 29-6 summarizes this functionality.

Table 29-6: FRC Tuning Functionality

PRCDEN	TSEQEN	FRC Behavior
0	0	FRC tuning is based on the 4-bit number specified by the TUN<3:0> bits (OSCTUN<3:0>), as shown in Table 29-5.
0	1	<p>Every 8th PWM cycle, the oscillator system picks one 4-bit number from the following sequence of the tuning bits groups:</p> <ul style="list-style-type: none"> • TUN<3:0> • TSEQ1<3:0> • TSEQ2<3:0> • TSEQ3<3:0> • TSEQ4<3:0> • TSEQ5<3:0> • TSEQ6<3:0> • TSEQ7<3:0> <p>Once the 4-bit number is picked, the FRC is retuned to the new frequency based on this value, as shown in Table 29-5. For example, the value of TUN<3:0> is initially used to tune the FRC frequency. After 16 μs (if the PWM period is set to 2 μs), the FRC is retuned to the new frequency based on the value specified by TSEQ1<3:0>.</p>
1	X	<p>The oscillator system generates a 4-bit number (0x0-0xF) based on a pseudo-random number generation algorithm. It then uses this value to tune the FRC as shown in Table 29-5. The pseudo-random number is regenerated (and the FRC is retuned) every 8th PWM cycle. For example, if the user has specified a PWM period of 2 μs, the FRC is retuned to a new frequency every 16 μs.</p> <p>Optionally, the user application can use the Linear Feedback Shift register (LFSR) to provide a seed number for the pseudo-random number generation algorithm.</p>

To change the PRCDEN and TSEQEN bits in the OSCCON register, the user application must first perform an unlock sequence on the OSCCON register low byte by executing the following instruction sequence in back-to-back instructions:

1. Write '0x46' to OSCCON<7:0>
2. Write '0x57' to OSCCON<7:0>
3. In the immediately following instruction, write new values into the PRCDEN and TSEQEN bits.

The MPLAB C30 compiler simplifies this unlock sequence by providing built-in C language primitives for that purpose. Example 29-1 sets the dsPIC30F SMPS oscillator tuning bits and enables sequential tuning/dithering. It uses the MPLAB C30 compiler's built-in function to unlock the OSCCON register.

dsPIC30F Family Reference Manual

Example 29-1: Sequential Dithering

```
#include "p30f2020.h"

/* Configuration Bit Settings */
_FOSCSEL(FRC_PLL)
_FOSC(CSW_FSCM_OFF & FRC_HI_RANGE & OSC2_CLKO)
_FPOR(PWRT_128)
_FGS(CODE_PROT_OFF)
_FBS(BSS_NO_FLASH)

int main()
{
    //~~~~~ FRC Dither Configuration ~~~~~
    __builtin_write_OSCCONL(4);           // Built-in function to unlock OSCCON register
                                           // Writing 0x0004 sets the TSEQEN bit
                                           // to enable FRC tune sequencer

    // FRC will tune to one of the following every 8 PWM cycles
    OSCTUNbits.TUN = 8;                   // Minimum frequency
    OSCTUNbits.TSEQ1 = 10;
    OSCTUNbits.TSEQ2 = 12;
    OSCTUNbits.TSEQ3 = 14;
    OSCTUN2bits.TSEQ4 = 1;
    OSCTUN2bits.TSEQ5 = 3;
    OSCTUN2bits.TSEQ6 = 5;
    OSCTUN2bits.TSEQ7 = 7;               // Maximum frequency

    //~~~~~End FRC Dither Configuration ~~~~~

    //~~~~~ PWM Configuration ~~~~~
    PTPER = 9520;                         // PWM period of 10 usec

    PDC1 = 4761;                          // PWM1 pulse width of 5 usec
    PDC2 = 2380;                          // PWM2 pulse width of 2.5 usec

    DTR1 = 64;                            // 67.2 nsec dead time
    DTR2 = 64;                            // 67.2 nsec dead time

    ALTDTR1 = 64;                         // 67.2 nsec dead time
    ALTDTR2 = 64;                         // 67.2 nsec dead time

    IOCON1bits.PENH = 1;                  // PWM1 outputs controlled by PWM
    IOCON1bits.PENL = 1;                  // module

    IOCON2bits.PENH = 1;                  // PWM2 outputs controlled by PWM
    IOCON2bits.PENL = 1;                  // module

    PTCONbits.PTEN = 1;                  // Turn ON PWM module
    //~~~~~ End PWM Configuration ~~~~~

    while(1);                            // Infinite loop
}

```

29.7 Phase Lock Loop (PLL)

The Primary Oscillator and Internal FRC Oscillator sources can optionally use an on-chip PLL to achieve higher operating speeds. The PLL does not require any special configuration and is preconfigured to multiply the input frequency by 32.

The PLL requires a finite amount of time (TLOCK) to synchronize. TLOCK is applied when the PLL is selected as the clock source at Power-on Reset, or during a clock-switch operation. The value of TLOCK is relative to the time at which the clock is available to the PLL input. For example, with the Primary Oscillator, TLOCK starts after the OST delay. Refer to **Section 29.5.1 “Oscillator Start-up Time”** for detailed information.

The PLL Lock Status (LOCK) bit in the Oscillator Control (OSCCON<5>) register is a read-only status bit that indicates the lock status of the PLL. When the PLL is selected as the destination clock source, the LOCK bit is cleared at Power-on Reset and on a clock-switch operation. It remains clear when any clock source not using the PLL is selected. It is good practice to wait for the LOCK bit to be set before executing other code after a clock-switch event in which the PLL is enabled. Example 29-2 shows a code example.

Example 29-2: Wait for PLL to Lock

```
while(OSCCONbits.LOCK!=1) {};
```

29.8 Low-Power RC (LPRC) Oscillator

The Low-Power RC (LPRC) oscillator provides a nominal clock frequency of 32 kHz. The LPRC is the clock source for the Power-up Timer (PWRT), Watch Dog Timer (WDT) and Fail-Safe Clock Monitor (FSCM) circuits.

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

29.8.1 Enabling the LPRC Oscillator

The LPRC oscillator is the clock source for the PWRT, WDT, and FSCM.

The LPRC oscillator is enabled at Power-on Reset by the Power-on Reset Timer Value Select (FPWRT) bits in the POR Configuration Fuse (FPOR<2:0>) register.

The LPRC oscillator remains enabled under the following conditions:

- The FSCM is enabled
- The WDT is enabled

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.

29.8.2 LPRC Oscillator Start-up Delay

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

29.9 FAIL-SAFE CLOCK MONITOR

The Fail-Safe Clock Monitor (FSCM) allows the device to continue to operate in the event of an oscillator failure. The FSCM function is enabled by programming the Clock Switching Mode (FCKSM<1:0> configuration bits in the Oscillator Configuration (FOSC<7:6>) register at the time of device programming. When the FSCM is enabled (FCKSM = 00), the LPRC internal oscillator will run at all times (except during Sleep).

The FSCM monitors the system clock. If it does not detect a system clock within a specific period of time (typically 2 ms), it generates a clock failure trap and switches the system clock to the FRC oscillator. The user application then has the option to either attempt to restart the oscillator or execute a controlled shutdown.

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

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

Note: For more information about the oscillator failure trap, please refer to “**Section 28. Interrupts**”, which is available from the Microchip Web site.

29.9.1 FSCM Delay

On a Power-on Reset or Wake-up From Sleep mode, a nominal delay (TFSCM) can be inserted before the FSCM begins to monitor the system clock source. The FSCM delay provides time for the oscillator and/or PLL to stabilize when the Power-up Timer (PWRT) is not used. The FSCM delay is generated after the internal System Reset signal (SYSRST) has been released. For FSCM delay timing information, refer to **Section 8. “Reset”**, which is available from the Microchip Web site.

The FSCM delay is applied when the FSCM is enabled and the Primary or Secondary Oscillator is selected as the system clock.

Note: Please refer to the “**Electrical Characteristics**” section of the specific device data sheet for TFSCM values.

29.9.2 FSCM and Slow Oscillator Start-up

If the chosen device oscillator has a slow start-up time coming out of a Power-on Reset or Sleep mode, it is possible for the FSCM delay to expire before the oscillator has started. If this condition occurs, the FSCM initiates a clock failure trap. As this happens, the OSCCON<14:12> bits are loaded with the FRC oscillator selection, which effectively shuts off the original oscillator that was trying to start. The user application can detect this situation and initiate a clock switch back to the desired oscillator in the Trap Service Routine.

29.9.3 FSCM and WDT

The FSCM and the WDT both 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.

29.10 CLOCK SWITCHING

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

- Two-speed start-up sequence upon Power-on Reset, 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.
- Fail-Safe Clock Monitor automatically switches to Internal FRC Oscillator on a clock failure.
- User application software requests clock switching by setting the OSCCON<OSWEN> bit, causing the hardware to switch to the clock source selected by the OSCCON<NOSC> bits when the clock is ready.

In each of these cases, the clock switch event assures that 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.

With few limitations, applications are free to switch between any of the four clock sources (POSC, SOSC, FRC and LPRC) under software control at any time. To limit the possible side effects that could result from this flexibility, dsPIC30F SMPS devices have a safeguard lock built into the switch process. That is, the OSCCON register is write-protected during clock switching.

29.10.1 Enabling Clock Switching

The Clock Switching Mode (FCKSM<1:0>) configuration bits in the Oscillator Configuration (FOSC<7:6>) register must be programmed to enable clock switching and configure the Fail-Safe Clock Monitor (see Table 29-7).

Table 29-7: 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.

29.10.2 Clock Switch Sequence

The recommended process for a clock switch is as follows:

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

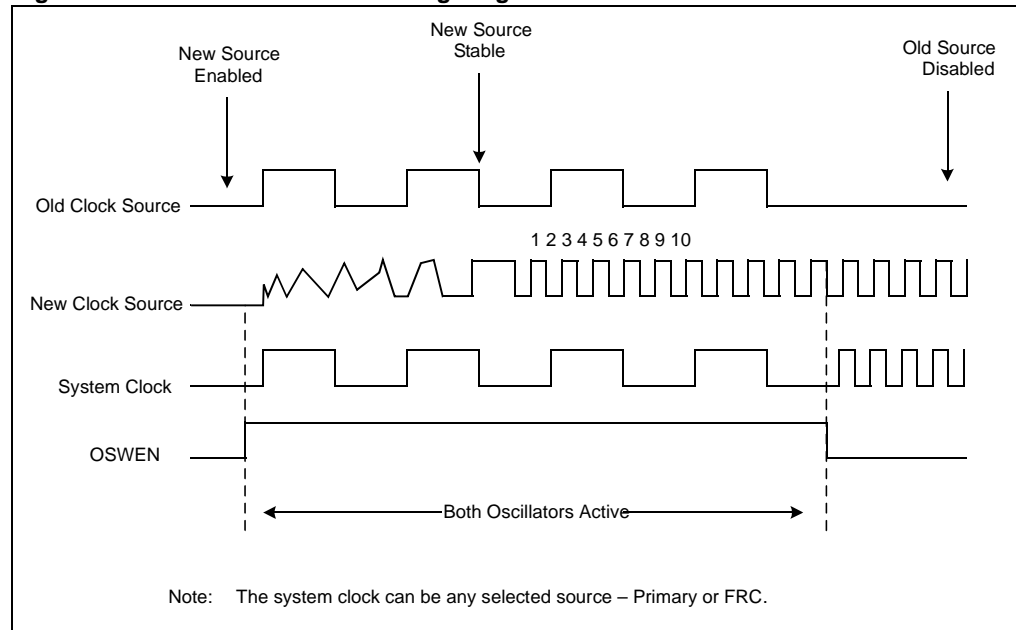
After the above steps are completed, the clock switch logic performs the following:

1. The clock switching hardware compares the OSCCON<COSC> status bits with the new value of the OSCCON<NOSC> control bits. If they are the same, the clock switch is a redundant operation. In this case, the OSCCON<OSWEN> bit is cleared automatically and the clock switch is aborted.
2. If a valid clock switch has been initiated, the PLL Lock (OSCCON<LOCK>) and Clock Fail (OSCCON<CF>) status bits are cleared.
3. The new oscillator is turned on by the hardware (if it is not currently running). If a crystal oscillator (Primary or Secondary) must be turned on, the hardware waits until the OST expires. If the new source uses the PLL, the hardware waits until a PLL lock is detected (OSCCON<LOCK> = 1).

4. The hardware waits for the new clock source to stabilize and then performs the clock switch.
5. The hardware clears the OSCCON<OSWEN> bit to indicate a successful clock transition. In addition, the OSCCON<NOSC> bit values are transferred to the OSCCON<COSC> status bits.
6. The old clock source is turned off at this time, with the exception of the LPRC (if the WDT or FSCM are enabled). The timing of the transition between clock sources is shown in Figure 29-9.

- Note 1:** Clock switching between HS and EC primary oscillator modes is not possible without reprogramming the device.
- 2:** Direct clock switching between PLL modes is not possible. For example, clock switching should not occur between the Primary Oscillator with PLL and Internal FRC oscillator with PLL.
- 3:** Setting the OSCCON<CLKLOCK> bit prevents clock switching when clock switching is enabled and fail-safe clock monitoring is disabled by configuration bits (FOSC<FCKSM> = 01). The OSCCON<CLKLOCK> bit cannot be cleared once it is set by the software. It clears on Power-on Reset.
- 4:** The processor continues to execute code throughout the clock switching sequence. Timing sensitive code should not be executed during this time.
- 5:** The application should not attempt to switch to a clock frequency lower than 100 kHz when the Fail-Safe Clock Monitor is enabled. If clock switching is performed, the device may generate an oscillator fail trap and switch to the FRC.

Figure 29-9: Clock Transition Timing Diagram



dsPIC30F Family Reference Manual

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

1. Disable interrupts during the OSCCON register unlock and write sequence.
2. Execute the unlock sequence for the OSCCON high byte.
In two, back-to-back instructions do the following:
 - Write '78h' to OSCCON<15:8>
 - Write '9Ah' to OSCCON<15:8>
3. In the instruction immediately following the unlock sequence, write the new oscillator source to the OSCCON<NOSC> control bits.
4. Execute the unlock sequence for the OSCCON low byte.
In two, back-to-back instructions do the following:
 - Write '46h' to OSCCON<7:0>
 - Write '57h' to OSCCON<7:0>
5. In the instruction immediately following the unlock sequence, set the OSCCON<OSWEN> bit.
6. Continue to execute code that is not clock-sensitive (optional).
7. Check to see if OSCCON<OSWEN> is '0'. If it is, the switch was successful.

Note: As stated earlier, MPLAB[®] C30 provides built-in C language functions for unlocking the OSCCON register:
__ builtin_write_OSCCONL (value)
__ builtin_write_OSCCONH (value)
See the MPLAB Help for more information.

Example 29-3 illustrates the code sequence for unlocking the OSCCON register and switching from the FRC with PLL clock to the Primary Oscillator HS clock source.

Example 29-3: Code Example for Clock Switching

```
;Place the New Oscillator Selection (NOSC=0b010) in W0
MOV #0x12,w0

;OSCCONH (high byte) Unlock Sequence
MOV #OSCCONH, w1
MOV #0x78, w2
MOV #0x9A, w3
MOV.B w2, [w1] ; Write 78h
MOV.B w3, [w1] ; Write 9Ah

;Set New Oscillator Selection
MOV.B w0, [w1]

; 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 46h
MOV.B w3, [w1] ; Write 57h

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

29.10.3 Clock Switching Considerations

When you incorporate clock switching into an application, keep the following issues in mind while designing your code:

- The OSCCON unlock sequence is extremely timing critical. The OSCCON register byte is only writable for one 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, the clock switching hardware will wait indefinitely for the new clock source. Your software can detect this situation because the OSCCON<OSWEN> bit remains set indefinitely.
- If the new clock source uses the PLL, a clock switch will not occur until lock has been achieved. Your software can detect a loss of PLL lock because the OSCCON<LOCK> bit is cleared and the OSCCON<OSWEN> bit is set.

29.10.4 Aborting a Clock Switch

If a clock switch does not complete, the clock switch logic can be reset by clearing the OSCCON<OSWEN> bit. When OSWEN is cleared, the clock switch process is aborted, the Oscillator Start Timer (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 29-4. 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 29-4: 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
    
```

29.10.5 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 PWR-SAV instruction is then executed normally.

29.11 Differences Between Industrial and Extended Temperature Range Devices

Industrial temperature range dsPIC30F SMPS devices are designed for operation up to 30 MIPS. Extended temperature range devices are designed for up to 20 MIPS operation. This difference influences some of the oscillator module functionality as shown in Table 29-8.

Table 29-8: Temperature Range Influences on Oscillator Functionality

Functionality	Industrial Temperature Range dsPIC30F SMPS Devices	Extended Temperature Range dsPIC30F SMPS Devices
Maximum FCY	30 MHz	20 MHz
Maximum FOSC	60 MHz	40 MHz
Maximum FPWM	480 MHz	320 MHz
Maximum FADC	240 MHz	160 MHz
High Range FRC frequency for 5.0V supply voltage	14.55 MHz	9.7 MHz
Low Range FRC frequency for 5.0V supply voltage	9.7 MHz	6.4 MHz

29.12 Register Maps

Table 29-9 provides a map of the registers related to the oscillator system. Table 29-10 lists the configuration registers related to the oscillator system.

Table 29-9: Oscillator Control Registers

File Name	Addr	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	0742	—	COSC<2:0>			—	NOSC<2:0>			CLKLOCK	—	LOCK	PRCDEN	CF	TSEQEN	—	OSWEN	7700 ⁽¹⁾
OSCTUN	0748	TSEQ3<3:0>			TSEQ2<3:0>			TSEQ1<3:0>			TUN<3:0>					0000		
OSCTUN2	0746	TSEQ7<3:0>			TSEQ6<3:0>			TSEQ5<3:0>			TSEQ4<3:0>					0000		
LFSR	0748	—	LFSR<14:0>															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 type of Reset.

Table 29-10: Oscillator Configuration Registers

File Name	Addr	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
FOSCSEL	F80006	—	—	—	—	—	—	—	—	—	—	—	—	—	FNOSC<2:0>			
FOSC	F80008	—	—	—	—	—	—	—	—	FCKSM<1:0>		FRANGE	—	—	OSCI0FNC	POSCMD<1:0>		

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

29.13 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 dsPIC30F SMPS 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 the following:

Title	Application Note #
<i>"PICmicro[®] Microcontroller Oscillator Design Guide"</i>	AN588
<i>"Low Power Design using PICmicro[®] Microcontrollers"</i>	AN606
<i>"Crystal Oscillator Basics and Crystal Selection for rfPIC[®] and PICmicro[®] Devices"</i>	AN826

Note: Please visit the Microchip web site (www.microchip.com) for additional Application Notes and code examples for the dsPIC30F SMPS Family of devices.

29.14 Revision History

Revision A

This is the initial release of this document. (January 2007)

NOTES: