Section 42. Oscillators with Enhanced PLL

HIGHLIGHTS

This section of the manual contains the following major topics:

Note:	This family reference manual section is meant to serve as a complement to device data sheets. Depending on the device variant, this manual section may not apply to all PIC32 devices. Please consult the note at the beginning of the **"Oscillator Configuration"** chapter in the current device data sheet to check whether this document supports the device you are using. Device data sheets and family reference manual sections are available for download from the Microchip Worldwide Web site at: http://www.microchip.com

42.1 INTRODUCTION

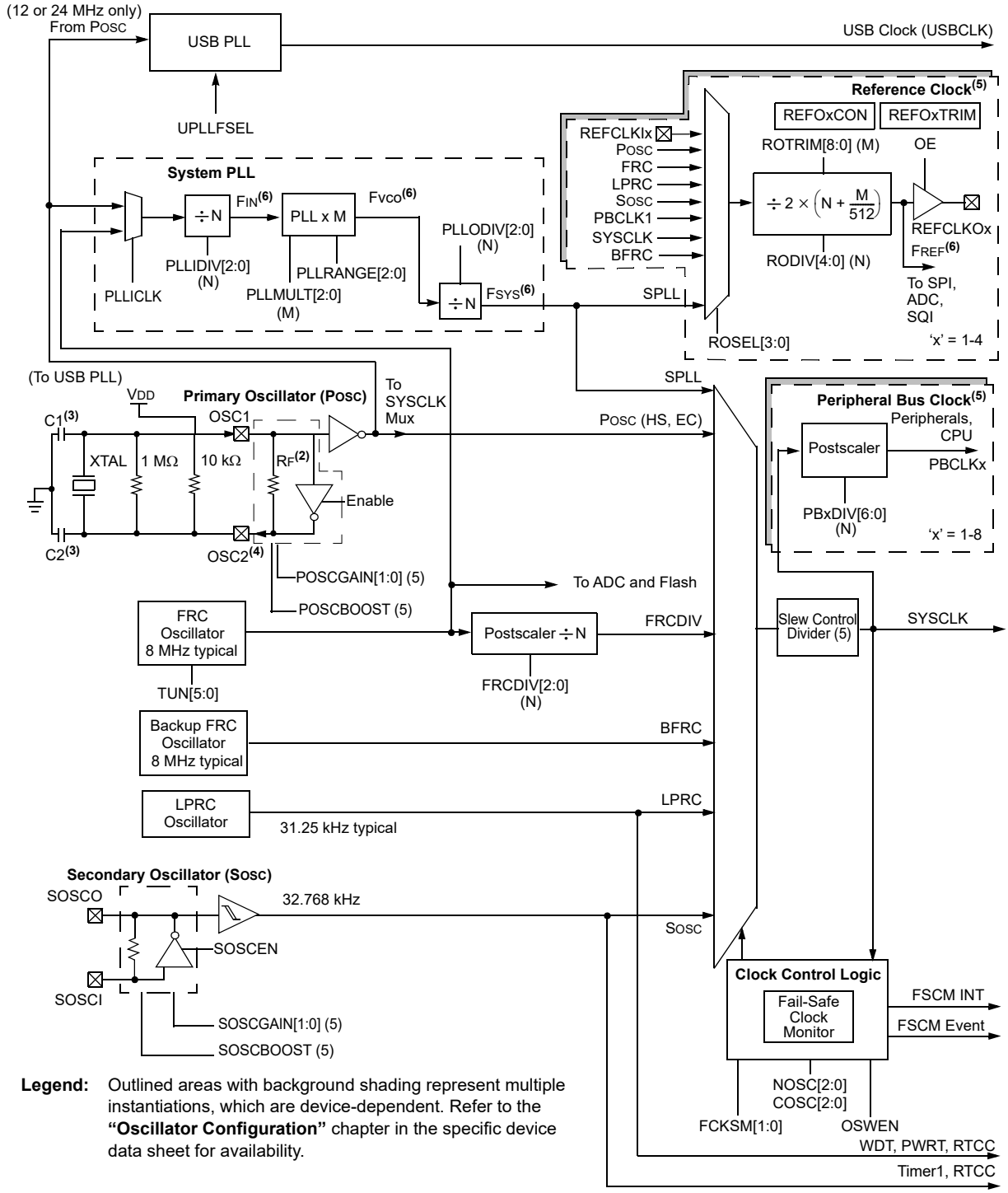
This section describes the PIC32 oscillator system and its operation. The PIC32 oscillator system has the following modules and features:

- Five external and internal oscillator options as clock sources
- On-chip Phase-Locked Loop (PLL) with a user-selectable input divider and multiplier, as well as an output divider, to boost operating frequency on select internal and external oscillator sources
- On-chip user-selectable divisor postscaler on select oscillator sources
- Software-controllable switching between various clock sources
- Flexible reference clock output
- Multiple clock branches for peripherals, which provides better performance flexibility
- A Fail-Safe Clock Monitor (FSCM) that detects clock failure and permits safe application recovery or shutdown
- A dedicated Backup Fast RC oscillator (BFRC), which provides an 8 MHz clock when the FSCM detects a clock failure to support Class B operation (this feature is not available on all PIC32 devices; refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability)

A block diagram of the PIC32 oscillator system is shown in Figure 42-1.

Figure 42-1: PIC32 Family Oscillator System Block Diagram

Note: This diagram depicts a general representation of the Oscillator module on a PIC32 device; therefore, depending on the device, not all features may be available. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet to determine which features are available on your device.



Legend: Outlined areas with background shading represent multiple instantiations, which are device-dependent. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

Note 1: A series resistor, RS, may be required for AT strip cut crystals or to eliminate clipping. Alternatively, to increase oscillator circuit gain, add a parallel resistor, RP, with a value of 1 MΩ.

2: The internal feedback resistor, RF, is typically in the range of 2 to 10 MΩ.

3: See **42.7.3.1 "Determining the Best Values for Oscillator Components"**.

4: PBCLK1 divided by 2 is available on the OSC2 pin in certain clock modes.

5: This feature is not available on all PIC32 devices. Refer to the **"Oscillator Configuration"** Chapter in the specific device data sheet for more information.

6: Refer to the **"Electrical Characteristics"** chapter in the specific device data sheet for FSYS frequency limitations.

42.2 CONTROL REGISTERS

The Oscillator module consists of the following Special Function Registers (SFRs):

- **OSCCON: Oscillator Control Register**

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

- **OSCTUN: FRC Tuning Register**

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

- **SPLLCON: System PLL Control Register**(2)

  This register is used to control the system clock PLL. It allows the input frequency to be used to generate a higher system frequency.

- **SPLLCON: System PLL Control Register (PIC32MZ W1 only)**(3)

  This register is used to control the system clock PLL. It allows the input frequency to be used to generate a higher system frequency.

- **UPLLCON: UPLL Control Register**(2)

  This register is used to control the USB clock PLL.

- **BTPLLCON: BTPLL Control Register**(2)

  This register is used to control the Ethernet-Wi-Fi clock PLL.

- **EWPLLCON: EWPLL Control Register**(2)

  This register is used to control the Bluetooth clock PLL.

- **REFOCON/REFOxCON: Reference Oscillator Control Register**

  This register controls the reference oscillator output.

- **REFOTRIM/REFOxTRIM: Reference Oscillator Trim Register**

  This register fine tunes the reference oscillator(s).

- **PBxDIV: Peripheral Bus 'x' Clock Divisor Control Register ('x' = 1-8)**

  This register is used to control the amount by which the system clock is divided to drive the individual peripheral clocks.

- **SLEWCON: Oscillator Slew Control Register**

  This register (see **Note 2**) is used to control how quickly the oscillator changes from one frequency to another. When used, the frequency change can be stepped across dividers to make the change more stable.

- **CLKSTAT: Oscillator Clock Status Register**

  This register (see **Note 2**) provides a status on the availability of various clock sources within the Oscillator module and obtains some of the status bits that were in the OSCCON register.

- **CLKSTAT: Oscillator Clock Status Register (PIC32MZ W1 Only)**(3)

  This register provides a status on the availability of various clock sources within the Oscillator module and obtains some of the status bits that were in the OSCCON register.

Note 1: Device Configuration Word registers are also available to provide additional configuration settings that are related to the Oscillator module. Refer to the **"Special Features"** chapter in the specific device data sheet for detailed information on these registers.

2: This register is not available for all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

3: This register is available only in the PIC32MZ W1 Family. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for more details.

42.2.1 Special Function Register Summary

Table 42-1 provides a brief summary of the related Oscillator module registers. The corresponding registers appear after the summary, which include detailed descriptions.

Table 42-1: Oscillators SFR Summary

Name	Bit Range	Bit 31/15	Bit 30/14	Bit 29/13	Bit 28/12	Bit 27/11	Bit 26/10	Bit 25/9	Bit 24/8	Bit 23/7	Bit 22/6	Bit 21/5	Bit 20/4	Bit 19/3	Bit 118/2	Bit 17/1	Bit 16/0
OSCCON	31:16	—	—	—	—	—	FRCDIV[2:0]			DRMEN	SOSCRDY(1)	SLP2SPD(1)	—	—	—	—	—
	15:0	—	COSC[2:0]			—	NOSC[2:0]			CLKLOCK	ULOCK(1)	SLOCK(1)	SLPEN	CF	UFRCEN(1)	SOSCEN	OSWEN
OSCTUN	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	15:0	—	—	—	—	—	—	—	—	—	—	TUN[5:0](1)					
SPLLCON	31:16	—	—	—	—	—	PLLODIV[2:0]			—	PLLMULT[6:0]						
	15:0	—	—	—	—	—	PLLIDIV[2:0]			PLLICLK	—	—	—	—	PLLRANGE[2:0]		
SPLLCON(2)	31:16	SPLL_BYP	SPLLICLK	—	—	SPLLREFDIV[5:0]						SPLLFBDIV[9:0]					
	15:0	SPLLFBDIV[9:0]				SPLLRST	SPLL-FLOCK	SPLLPOSTDIV1[5:0]						SPLLP-WDN	SPLLBSWSEL[2:0]		
UPLLCON(1)	31:16	UPLL_BYP	—	—	—	UPLLREFDIV[5:0]						UPLLFBDIV[9:0]					
	15:0	UPLLFBDIV[9:0]				UPLLRST	UPLL-FLOCK	UPLLPOSTDIV1[5:0]						UPLLP-WDN	UPLLBSWSEL[2:0]		
BTPLLCON(1)	31:16	BTPLL_BYP	BTPLLICLK	—	BTCLK-OUTEN	BTPLLREFDIV[5:0]						BTPLLFBDIV[9:0]					
	15:0	BTPLLFBDIV[9:0]				BTPLLRST	BTPLL-FLOCK	BTPLLPOSTDIV1[5:0]						BTPLL-PWDN	BTPLLBSWSEL[2:0]		
EWPLLCON(1)	31:16	EWPLL_BYP	EWPLLI-CLK	—	ETHCLK OUTEN	EWPLLREFDIV[5:0]						EWPLLFBDIV[9:0]					
	15:0	EWPLLFBDIV[9:0]				EWPLLRST	EWPLL-FLOCK	EWPLLPOSTDIV1[5:0]						EWPLL-PWDN	EWPLLBSWSEL[2:0]		
REFOCON/ REFOxCON	31:16	—	RODIV[14:0]														
	15:0	ON	—	SIDL	OE	RSLP	—	DIVSWEN	ACTIVE	—	—	—	—	ROSEL[3:0]			
REFOTRIM/ REFOxTRIM	31:16	ROTRIM[8:0]									—	—	—	—	—	—	—
	15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
PBxDIV	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	15:0	ON	—	—	—	PBDIVRDY	—	—	—	—	PBDIV[6:0]						
SLEWCON(1)	31:16	—	—	—	—	SLW_DELAY[3:0](1)				—	—	—	—	SYSDIV[3:0]			
	15:0	—	—	—	—	—	SLWDIV[2:0]			—	—	—	—	—	UPEN	DNEN	BUSY

Note 1: These bits or registers are not available in all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

2: This register is available only in thePIC32MZ W1 Family. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for more details.

Table 42-1: Oscillators SFR Summary

Name	Bit Range	Bit 31/15	Bit 30/14	Bit 29/13	Bit 28/12	Bit 27/11	Bit 26/10	Bit 25/9	Bit 24/8	Bit 23/7	Bit 22/6	Bit 21/5	Bit 20/4	Bit 19/3	Bit 118/2	Bit 17/1	Bit 16/0
CLKSTAT(1)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	15:0	—	—	—	—	—	—	—	—	SPLLRDY	—	LPRCRDY	SOSCRDY	—	POSCRDY	—	FRCRDY
CLKSTAT(2)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	15:0	—	—	—	—	SYSCLKRDY	PB1-CLKRDY	SPLLAL-TRDY	WIFI-CLKRDY	ETHPLL-RDY	BTPLLRDY	LPRC RDY	SOSC RDY	UPLL-RDY	POSC RDY	SPLL DIVRDY	FRCRDY

Note 1: These bits or registers are not available in all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

2: This register is available only in thePIC32MZ W1 Family. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for more details.

Register 42-1: OSCCON: Oscillator Control Register

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-1
	—	—	—	—	—	FRCDIV[2:0]		
23:16	R/W-0	R-0	R-0	U-0	U-0	U-0	U-0	U-0
	DRMEN(1)	SOSCRDY(1)	SLP2SPD(1)	—	—	—	—	—
15:8	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]		
7:0	R/W-0	R-0	R-0	R/W-0	R/W-0, HS	U-0	R/W-y	R/W-y
	CLKLOCK	ULOCK(1)	SLOCK(1)	SLPEN	CF(2)	UFRCEN(1)	SOSCEN	OSWEN(3)

Legend:	y = Value set from Configuration bits on POR		HS = Set by Hardware
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 31-27 **Unimplemented:** Read as '0'

bit 26-24 **FRCDIV[2:0]:** Internal Fast RC (FRC) Oscillator Clock Divider 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 (POR default setting)

bit 23 **DRMEN:** Dream Mode Enable bit(1)
- 1 = Enable Dream mode. DMA transfers that occur from an interrupt are completed without waking up the CPU from Sleep mode.
- 0 = Disable Dream mode. All interrupts from operating peripherals will wake-up the CPU.

bit 22 **SOSCRDY:** Secondary Oscillator (SOSC) Ready Indicator bit(1)
- 1 = Indicates that the SOSC is running and is stable
- 0 = SOSC is still warming up or is turned off

bit 21 **SLP2SPD:** Sleep Two-Speed Start-up Control bit(1)
- 1 = Use FRC as SYSCLK until selected clock is ready
- 0 = Use the selected clock directly when it is available

bit 20-15 **Unimplemented:** Read as '0'

Note 1: This bit or feature is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

2: Writing a '1' to this bit will cause a FSCM clock switchover.

3: The reset value of this bit depends on the value of the IESO bit (DEVCFG1[7]). If the IESO bit is set to '1', the OSWEN bit will reset to a '1' to trigger the change from the FRC to the options chosen in the DEVCFG1 and DEVCFG2 registers.

Note: Writes to this register require an unlock sequence. Refer to **42.3.7.2 "Oscillator Switching Sequence"** for details.

Register 42-1: OSCCON: Oscillator Control Register (Continued)

bit 14-12 **COSC[2:0]:** Current Oscillator Selection bits
111 = Fast RC Oscillator (FRC) divided by FRCDIV[2:0] setting
110 = Back-up Fast RC Oscillator (BFRC) switched by hardware FSCM only; not user-selectable(1)
101 = Low-Power RC (LPRC) Oscillator
100 = SOSC
011 = Reserved; do not use
010 = Primary Oscillator (POSC) HS and EC
001 = System PLL (SPLL)
000 = FRC Oscillator divided by FRCDIV[2:0] setting

The definitions and availability of these bits is device-dependent. For details, please refer to the **"Oscillator Configuration"** chapter in the specific device data sheet. In addition, upon Reset, these bits are set to the value of the FNOSC Configuration bits (DEVCFG1[2:0]).

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

bit 10-8 **NOSC[2:0]:** New Oscillator Selection bits
111 = FRC divided by FRCDIV[2:0] setting
110 = Reserved; do not use
101 = LPRC
100 = SOSC
011 = Reserved; do not use
010 = POSC HS and EC
001 = SPLL
000 = FRC divided by FRCDIV[2:0] setting

The definitions and availability of these bits is device-dependent. For details, please refer to the **"Oscillator Configuration"** chapter in the specific device data sheet. In addition, upon Reset, these bits are set to the value of the FNOSC[2:0] Configuration bits (DEVCFG1[2:0]).

bit 7 **CLKLOCK:** Clock Selection Lock Enable bit
1 = Clock and PLL configurations are locked
0 = Clock and PLL configurations can be modified

Once CLKLOCK is set, it can only be cleared by a device Reset. When active, this bit prevents writes to the NOSC[2:0] and OSWEN bits.

bit 6 **ULOCK:** USB PLL Lock Status bit(1)
1 = Indicates that the USB PLL module is in lock or the USB PLL module start-up timer is satisfied
0 = Indicates that the USB PLL module is out of lock, or the USB PLL module start-up timer is in progress, or the USB PLL is disabled

bit 5 **SLOCK:** System PLL Lock Status bit(1)
1 = System PLL module is in lock or module start-up timer is satisfied
0 = System PLL module is out of lock, start-up timer is running or system PLL is disabled

bit 4 **SLPEN:** Sleep Mode Enable bit
1 = Device will enter Sleep mode when a WAIT instruction is executed
0 = Device will enter Idle mode when a WAIT instruction is executed

bit 3 **CF:** Clock Fail Detect bit(2)
1 = FSCM has detected a clock failure
0 = No clock failure has been detected

Note 1: This bit or feature is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

2: Writing a '1' to this bit will cause a FSCM clock switchover.

3: The reset value of this bit depends on the value of the IESO bit (DEVCFG1[7]). If the IESO bit is set to '1', the OSWEN bit will reset to a '1' to trigger the change from the FRC to the options chosen in the DEVCFG1 and DEVCFG2 registers.

Note: Writes to this register require an unlock sequence. Refer to **42.3.7.2 "Oscillator Switching Sequence"** for details.

Register 42-1: OSCCON: Oscillator Control Register (Continued)

bit 2 **UFRCEN:** USB FRC Clock Enable bit(1)
1 = Enable FRC as the clock source for the USB clock source
0 = Use the primary oscillator or UPLL as the USB clock source

bit 1 **SOSCEN:** Secondary Oscillator (SOSC) Enable bit
1 = Enable SOSC
0 = Disable SOSC
The POR default is set by the FSOCSEN bit (DEVCFG1[5]).

bit 0 **OSWEN:** Oscillator Switch Enable bit(3)
1 = Initiate an oscillator switch to selection specified by NOSC[2:0] bits
0 = Oscillator switch is complete

Note 1: This bit or feature is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

2: Writing a '1' to this bit will cause a FSCM clock switchover.

3: The reset value of this bit depends on the value of the IESO bit (DEVCFG1[7]). If the IESO bit is set to '1', the OSWEN bit will reset to a '1' to trigger the change from the FRC to the options chosen in the DEVCFG1 and DEVCFG2 registers.

Note: Writes to this register require an unlock sequence. Refer to **42.3.7.2 "Oscillator Switching Sequence"** for details.

Register 42-2: OSCTUN: FRC Tuning Register

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	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](1)					

Legend:	y = Value set from Configuration bits on POR		
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 31-6 **Unimplemented:** Read as '0'

bit 5-0 **TUN[5:0]:** FRC Oscillator Tuning bits(1)

100000 = Center frequency -12.5%
100001 =
•
•
•
111111 =
000000 = Center frequency. Oscillator runs at calibrated frequency (8 MHz)
000001 =
•
•
•
011110 =
011111 = Center frequency +12.5%

Note 1: OSCTUN functionality has been provided to assist customers with compensating for temperature effects on the FRC frequency over a wide range of temperatures. The tuning step size is an approximation, and is neither characterized nor tested.

Note: Writes to this register require an unlock sequence. Refer to **42.3.7.2 "Oscillator Switching Sequence"** for details.

Register 42-3: SPLLCON: System PLL Control Register

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-y	R/W-y	R/W-y
	—	—	—	—	—	PLLODIV[2:0]		
23:16	U-0	R/W-y	R/W-y	R/W-y	R/W-y	R/W-y	R/W-y	R/W-y
	—	PLLMULT[6:0]						
15:8	U-0	U-0	U-0	U-0	U-0	R/W-y	R/W-y	R/W-y
	—					PLLIDIV[2:0]		
7:0	R/W-y	U-0	U-0	U-0	U-0	R/W-y	R/W-y	R/W-y
	PLLICLK	—	—	—	—	PLLRANGE[2:0]		

Legend:	y = Value set from Configuration bits on POR		
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 31-27 **Unimplemented:** Read as '0'

bit 26-24 **PLLODIV[2:0]:** System PLL (SPLL) Output Clock Divider bits

Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for descriptions and availability of these bits.

The default setting is specified by the FPLLODIV[2:0] Configuration bits in the DEVCFG2 register. Refer to the **"Special Features"** chapter in the specific device data sheet for more information.

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

bit 22-16 **PLLMULT[6:0]:** System Clock PLL Multiplier bits

Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for descriptions and availability of these bits.

The default setting is specified by the FPLLMUL[6:0] Configuration bits in the DEVCFG2 register. Refer to the **"Special Features"** chapter in the specific device data sheet for information. For additional bit values (if available), refer to the **"Oscillator Configuration"** chapter in the specific device data sheet.

bit 15-11 **Unimplemented:** Read as '0'

bit 10-8 **PLLIDIV[2:0]:** SPLL Input Clock Divider bits

Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for descriptions and availability of these bits.

The default setting is specified by the FPLLIDIV[2:0] Configuration bits in the DEVCFG2 register. Refer to the **"Special Features"** chapter in the specific device data sheet for details.

bit 7 **PLLICLK:** SPLL Input Clock Source bit

1 = FRC is selected as the input to the SPLL
0 = POSC is selected as the input to the SPLL

The POR default is specified by the FPLLICLK Configuration bit in the DEVCFG2 register. Refer to the **"Special Features"** chapter in the specific device data sheet for details.

bit 6-3 **Unimplemented:** Read as '0'

bit 2-0 **PLLRANGE[2:0]:** SPLL Frequency Range Selection bits

Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for descriptions and availability of these bits.

The POR default is specified by the FPLLRANGE[2:0] Configuration bits in the DEVCFG2 registers. Refer to the **"Special Features"** chapter in the specific device data sheet for details.

Note 1: Writes to this register require an unlock sequence. Refer to **42.3.7.2 "Oscillator Switching Sequence"** for details.

2: In addition, writes to this register are not allowed if the SPLL is selected as the clock source (COSC[2:0] = 001).

Register 42-4: SPLLCON: System PLL Control Register (PIC32MZ W1 only)

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W/L-1	R/W/L-1	U-0	U-0	R/W-0	R/W/L-0	R/W-0	R/W-0
	SPLL_BYP	SPLLICLK	—	—	SPLLREFDIV[5:2]			
23:16	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0
	SPLLREFDIV[1:0]		SPLLFBDIV[9:4]					
15:8	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-1	R/W/L-0	R/W/L-0	R/W/L-0
	SPLLFBDIV[3:0]				SPLLRST	SPLLFLOCK	SPLLPOSTDIV1[5-4]	
7:0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-1	R/W/L-0	R/W/L-0	R/W/L-0
	SPLLPOSTDIV1[3:0]				SPLLPWDN	SPLLBSWSEL[2:0]		

Legend:	L = Value set from Configuration bits on POR		
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 31 **SPLL_BYP:** SPLL Bypass bit; when this bit is set, the input clock REF bypasses PLL to PLLOUTx.

Note: Recommendation is to setup SPLL first before setting up other PLLs especially when using SYS-PLL for generating the main system clock. The Reset value must be 1, because the PLL needs to be set up in SW for generating the desired frequency.

bit 30 **SPLLICLK:** Source Input Clock Selection bit:

0 = POSC is the SPLL input clock source
1 = FRC is the SPLL input clock source

Note: The Reset value must be 1, because the POSC is not available upon Reset.

bit 29-28 **Unimplemented:** Read as '0'

bit 27-22 **SPLLREFDIV[5:0]:** Reference Frequency Divide bit,1 ≤ SPLLDIVR ≤ 63, value of 0 is unused.

bit 21-12 **SPLLFBDIV[9:0]:** PLL Feedback Divider bit, 16 ≤ SPLLFBDIV ≤ 1023, values of 0 to 15 are unused.

bit 11 **SPLLRST:** System PLL Reset bit

1 = Assert the Reset to the SPLL
0 = De-assert the Reset to the SPLL

bit 10 **SPLLFLOCK:** System PLL Force Lock bit

1 = Force the SPLL lock signal to be asserted
0 = Do not force the SPLL lock signal to be asserted

bit 9-4 **SPLLPOSTDIV1[5:0]:** First Post Divide Value bit. 1 ≤ SPLLPOSTDIV1 ≤ 63, value of 0 is unused.

bit 3 **SPLLPWDN:** PLL Power Down Register bit

1 = PLL is powered down
0 = PLL is active

bit 2-0 **SPLLBSWSEL[2:0]:** PLL Bandwidth Select bit

Use the frequency range that matches the PLL closed loop bandwidth based on the reference frequency divided by REFDIV to be set to allow the PLL loop filter to work with the post-reference divider frequency.

Note 1: The system unlock sequence must be done before this register can be written.

2: This register is available only in the PIC32MZ W1 Family. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for more details.

Register 42-5: UPLLCON: UPLL Control Register

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W/L-1	U-0	U-0	U-0	R/W-0	R/W/L-0	R/W-0	R/W-0
	UPLL_BYP	—	—	—	UPLLREFDIV[5:2]			
23:16	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0
	UPLLREFDIV[1:0]		UPLLFBDIV[9:4]					
15:8	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-1	R/W/L-0	R/W/L-0	R/W/L-0
	UPLLFBDIV[3:0]				UPLLRST	UPLLFLOCK	UPLLPOSTDIV1[5-4]	
7:0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-1	R/W/L-0	R/W/L-0	R/W/L-0
	UPLLPOSTDIV1[3:0]				USPLLPWDN	UPLLBSWSEL[2:0]		

Legend:	L = Value set from the Configuration bits on POR		
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 31 **UPLL_BYP:** UPLL Bypass; when this bit is set, the input clock REF bypasses PLL to PLLOUTx.

Note: Recommendation is to set up SPLL first before setting up other PLLs, especially when using SYS-PLL for generating the main system clock. The Reset value must be 1, because the PLL needs to be set up in SW for generating the desired frequency.

bit 30-28 **Unimplemented:** Read as '0'

bit 29-28 **Unimplemented:** Read as '0'

bit 27-22 **UPLLREFDIV[5:0]:** Reference Frequency Divide, 1 ≤ UPLLDIVR ≤ 63, value of 0 is unused.

bit 21-12 **UPLLFBDIV[9:0]:** PLL Feedback Divider, 16 ≤ UPLLFBDIV ≤ 1023, values of 0 to 15 are unused.

bit 11 **UPLLRST:** USB PLL Reset

1 = Assert the reset to the UPLL
0 = De-assert the reset to the UPLL

bit 10 **UPLLFLOCK:** USB PLL Force Lock

1 = Force the UPLL lock signal to be asserted
0 = Do not force the UPLL lock signal to be asserted

bit 9-4 **UPLLPOSTDIV1[5:0]:** First Post Divide Value, 1 ≤ UPLLPOSTDIV1 ≤ 63, value of 0 is unused.

bit 3 **UPLLPWDN:** PLL Power Down Register bit

1 = PLL is powered down
0 = PLL is active

bit 2-0 **UPLLBSWSEL[2:0]:** PLL Bandwidth Select

Use the frequency range that matches the PLL closed loop bandwidth based on the reference frequency divided by REFDIV to be set to allow the PLL loop filter to work with the post-reference divider frequency.

Note 1: The system unlock sequence must be done before this register can be written.

2: This register is not available for all devices. Refer to the "Oscillator Configuration" chapter in the specific device data sheet for availability.

Register 42-6: BTPLLCON: BTPLL Control Register

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W/L-1	R/W/L-1	U-0	R/W/L-0	R/W-0	R/W/L-0	R/W-0	R/W-0
	BTPLL_BYP	BTPLLICLK	—	BTCLKOUTEN	BTPLLREFDIV[5:2]			
23:16	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0
	BTPLLREFDIV[1:0]		BTPLLFBDIV[9:4]					
15:8	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-1	R/W/L-0	R/W/L-0	R/W/L-0
	BTPLLFBDIV[3:0]				BTPLLRST	BTPLLFLOCK	BTPLLPOSTDIV1[5-4]	
7:0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-1	R/W/L-0	R/W/L-0	R/W/L-0
	BTPLLPOSTDIV1[3:0]				BTPLLPWDN	BTPLLBSWSEL[2:0]		

Legend:	L = Value set from the Configuration bits on POR		
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 31 **BTPLL_BYP:** BTPLL Bypass; when this bit is set, the input clock REF bypasses PLL to PLLOUTx.

Note: Recommendation is to set up the SPLL first before setting up other PLLs, especially when using SYSPLL for generating the main system clock. The Reset value must be 1, because the PLL needs to be set up in SW for generating the desired frequency.

bit 30 **BTPLLICLK:** Source Input Clock Selection bit:

0 = POSC is the BTPLL input clock source
1 = FRC is the BTPLL input clock source

Note: The Reset value must be 1, because the POSC is not available upon Reset.

bit 29 **Unimplemented:** Read as '0'

bit 28 **BTCLKOUTEN:** BT Clock Out pin Enable bit

1 = BT_CLK_OUT Pin is enabled
0 = BT_CLK_OUT Pin is disabled

bit 27-22 **BTPLLREFDIV[5:0]:** Reference Frequency Divide, 1 ≤ BTPLLDIVR ≤ 63, value of 0 is unused.

bit 21-12 **BTPLLFBDIV[9:0]:** PLL Feedback Divider, 16 ≤ BTPLLFBDIV ≤ 1023, values of 0 to 15 are unused.

bit 11 **BTPLLRST:** BT PLL Reset

1 = Assert the reset to the BTPLL
0 = De-assert the reset to the BTPLL

bit 10 **BTPLLFLOCK:** BT PLL Force Lock

1 = Force the BTPLL lock signal to be asserted
0 = Do not force the BTPLL lock signal to be asserted

bit 9-4 **BTPLLPOSTDIV1[5:0]:** First Post Divide Value, 1 ≤ BTPLLPOSTDIV1 ≤ 63, value of 0 is unused.

bit 3 **BTPLLPWDN:** PLL Power Down Register bit

1 = PLL is powered down
0 = PLL is active

bit 2-0 **BTPLLBSWSEL[2:0]:** PLL Bandwidth Select

Use the frequency range that matches the PLL closed loop bandwidth based on the reference frequency divided by REFDIV to be set to allow the PLL loop filter to work with the post-reference divider frequency.

Note 1: The system unlock sequence must be done before this register can be written.

2: This register is not available for all devices. Refer to the "Oscillator Configuration" chapter in the specific device data sheet for availability.

Register 42-7: EWPLLCON: EWPLL Control Register

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W/L-1	R/W/L-1	U-0	R/W/L-0	R/W-0	R/W/L-0	R/W-0	R/W-0
	EWPLL_BYP	EWPLLICLK	—	ETHCLKOUTEN	EWPLLREFDIV[5:2]			
23:16	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0
	EWPLLREFDIV[1:0]		EWPLLFBDIV[9:4]					
15:8	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-1	R/W/L-0	R/W/L-0	R/W/L-0
	EWPLLFBDIV[3:0]				EWPLLRST	EWPLLFLOCK	EWPLLPOSTDIV1[5-4]	
7:0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-0	R/W/L-1	R/W/L-0	R/W/L-0	R/W/L-0
	EWPLLPOSTDIV1[3:0]				EWPLLPWDN	EWPLLBSWSEL[2:0]		

Legend:	L = Value set from the Configuration bits on POR		
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 31 **EWPLL_BYP:** EWPLL Bypass; when this bit is set, the input clock REF bypasses PLL to PLLOUTx.

Note: Recommendation is to set up the SPLL first before setting up other PLLs, especially when using SYSPLL for generating the main system clock. The Reset value must be 1, because the PLL needs to be set up in SW for generating the desired frequency.

bit 30 **EWPLLICLK:** Source Input Clock Selection bit:
0 = POSC is the EWPLL input clock source
1 = FRC is the EWPLL input clock source

Note: The Reset value must be 1, because the POSC is not available upon Reset. For Wi-Fi operation, this bit must be set to 0. For Ethernet, when providing reference clock to external PHY, this bit must be set to 0.

bit 29 **Unimplemented:** Read as '0'

bit 28 **ETHCLKOUTEN:** Ethernet Clock Out pin Enable bit
1 = ETH_CLK_OUT pin is enabled
0 = ETH_CLK_OUT pin is disabled

bit 27-22 **EWPLLREFDIV[5:0]:** Reference Frequency Divide, 1 ≤ EWPLLDIVR ≤ 63, value of 0 is unused.

bit 21-12 **EWPLLFBDIV[9:0]:** PLL Feedback Divider, 16 ≤ EWPLLFBDIV ≤ 1023, values of 0 to 15 are unused.

bit 11 **EWPLLRST:** EW PLL Reset
1 = Assert the reset to the EWPLL
0 = De-assert the reset to the EWPLL

bit 10 **EWPLLFLOCK:** EW PLL Force Lock
1 = Force the EWPLL lock signal to be asserted
0 = Do not force the EWPLL lock signal to be asserted

bit 9-4 **EWPLLPOSTDIV1[5:0]:** First Post Divide Value, 1 ≤ EWPLLPOSTDIV1 ≤ 63, value of 0 is unused.

bit 3 **EWPLLPWDN:** PLL Power Down Register bit
1 = PLL is powered down
0 = PLL is active

bit 2-0 **EWPLLBSWSEL[2:0]:** PLL Bandwidth Select

Use the frequency range that matches the PLL closed loop bandwidth based on the reference frequency divided by REFDIV to be set to allow the PLL loop filter to work with the post-reference divider frequency.

Note 1: The system unlock sequence must be done before this register can be written.

2: This register is not available for all devices. Refer to the "Oscillator Configuration" chapter in the specific device data sheet for availability.

Register 42-8: REFOCON/REFOxCON: Reference Oscillator Control Register

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	RODIV[14:8](1,3)						
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RODIV[7:0](1,3)							
15:8	R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0, HC	R-0, HS, HC
	ON(5)	—	SIDL	OE	RSLP(2)	—	DIVSWEN	ACTIVE
7:0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	ROSEL[3:0](1,3)			

Legend:	HC = Hardware Clearable	HS = Hardware Settable	
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 31 **Unimplemented:** Read as '0'

bit 30-16 **RODIV[14:0]** Reference Clock Divider bits(1,3)

Output clock frequency is the source clock frequency divided by 2 * [RODIV + (ROTRIM ÷ 512)]. This value selects the reference clock divider bits. See Figure 42-1 for more information.

111111111111111 = Output clock is source clock frequency divided by 65,534
111111111111110 = Output clock is source clock frequency divided by 65,532
•
•
•
000000000000010 = Output clock is source clock frequency divided by 4
000000000000001 = Output clock is source clock frequency divided by 2
000000000000000 = Output clock is same frequency as source clock (no divider)

bit 15 **ON:** Output Enable bit(5)

1 = Reference Oscillator Module enabled
0 = Reference Oscillator Module disabled

bit 14 **Unimplemented:** Read as '0'

bit 13 **SIDL:** Peripheral Stop in Idle Mode bit

1 = Discontinue module operation when device enters Idle mode
0 = Continue module operation in Idle mode

bit 12 **OE:** Reference Clock Output Enable bit

1 = Reference clock is driven out on REFCLKOx pin
0 = Reference clock is not driven out on REFCLKOx pin

bit 11 **RSLP:** Reference Oscillator Module Run in Sleep bit(2)

1 = Reference Oscillator Module output continues to run in Sleep
0 = Reference Oscillator Module output is disabled in Sleep

Note 1: The ROSEL[3:0] bits and the RODIV[14:0] bits should not be written while the ACTIVE bit is '1', as undefined behavior may result.

2: This bit is ignored when the ROSEL[3:0] bits = 0000 or 0001.

3: While the ON bit (REFOCON[15]) is '1', writes to these bits do not take effect until the DIVSWEN bit is set to '1'.

4: This bit selection is only available on devices with a BFRC oscillator. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

5: Do not write to this register when the ON bit is not equal to the ACTIVE bit.

Note: This register is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

Register 42-8: REFOCON/REFOxCON: Reference Oscillator Control Register (Continued)

bit 10 **Unimplemented:** Read as '0'

bit 9 **DIVSWEN:** Divider Switch Enable bit
1 = Divider switch is in progress
0 = Divider switch is complete

bit 8 **ACTIVE:** Reference Clock Request Status bit
1 = Reference clock request is active
0 = Reference clock request is not active

bit 7-4 **Unimplemented:** Read as '0'

bit 3-0 **ROSEL[3:0]:** Reference Clock Source Select bits(1,3)
1111 = Reserved; do not use
•
•
•
1001 = Reserved in devices without a BFRC oscillator; do not use
= BFRC(4)
1000 = REFCLKI
0111 = System PLL
0110 = Reserved; do not use
0101 = SOSC
0100 = LPRC
0011 = FRC
0010 = POSC
0001 = PBCLK/PBCLK1
0000 = SYSCLK

Note 1: The ROSEL[3:0] bits and the RODIV[14:0] bits should not be written while the ACTIVE bit is '1', as undefined behavior may result.

2: This bit is ignored when the ROSEL[3:0] bits = 0000 or 0001.

3: While the ON bit (REFOCON[15]) is '1', writes to these bits do not take effect until the DIVSWEN bit is set to '1'.

4: This bit selection is only available on devices with a BFRC oscillator. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

5: Do not write to this register when the ON bit is not equal to the ACTIVE bit.

Note:	This register is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

Register 42-9: REFOTRIM/REFOxTRIM: Reference Oscillator Trim Register

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ROTRIM[8:1]							
23:16	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	ROTRIM[0]	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

Legend:	y = Value set from Configuration bits on POR		
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 31-23 **ROTRIM[8:0]:** Reference Oscillator Trim bits

111111111 = 511/512 divisor added to RODIV value
111111110 = 510/512 divisor added to RODIV value
•
•
•
100000000 = 256/512 divisor added to RODIV value
•
•
•
000000010 = 2/512 divisor added to RODIV value
000000001 = 1/512 divisor added to RODIV value
000000000 = 0/512 divisor added to RODIV value

bit 22-0 **Unimplemented:** Read as '0'

Note 1: While the ON bit (REFOCON[15]) is '1', writes to this register do not take effect until the DIVSWEN bit is set to '1'.

2: Do not write to this register when the ON bit (REFOxCON[15]) is not equal to the ACTIVE bit (REFOxCON[8]).

3: Specified values in this register do not take effect if RODIV[14:0] (REFOxCON[30:16]) = 0.

4: This register is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

Register 42-10: PBxDIV: Peripheral Bus 'x' Clock Divisor Control Register ('x' = 1-8)

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-1	U-0	U-0	U-0	R-1	U-0	U-0	U-0
	ON(1)	—	—	—	PBDIVRDY	—	—	—
7:0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
	—	PBDIV[6:0]						

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 31-16 **Unimplemented:** Read as '0'

bit 15 **ON:** Peripheral Bus 'x' Output Clock Enable bit(1)
1 = Output clock is enabled
0 = Output clock is disabled

bit 14-12 **Unimplemented:** Read as '0'

bit 11 **PBDIVRDY:** Peripheral Bus 'x' Clock Divisor Ready bit
1 = Clock divisor logic is not switching divisors and the PBxDIV[6:0] bits may be written
0 = Clock divisor logic is currently switching values and the PBxDIV[6:0] bits cannot be written

bit 10-7 **Unimplemented:** Read as '0'

bit 6-0 **PBDIV[6:0]:** Peripheral Bus 'x' Clock Divisor Control bits
1111111 = PBCLKx is SYSCLK divided by 128
1111110 = PBCLKx is SYSCLK divided by 127
•
•
•
0000011 = PBCLKx is SYSCLK divided by 4
0000010 = PBCLKx is SYSCLK divided by 3
0000001 = PBCLKx is SYSCLK divided by 2 (default value for x ≠ 7)
0000000 = PBCLKx is SYSCLK divided by 1 (default value for x = 7)

For devices with only one PBCLK, the POR default is specified by the FPDIV[6:0] Configuration bits in the DEVCFG1 register. For devices with multiple PBCLKs, the POR default is device-dependent and is not configurable. Refer to the **"Oscillator Configuration"** and **"Special Features"** chapters in the specific device data sheet for details.

Note 1: The clock for peripheral bus 1 cannot be turned off. Therefore, the ON bit in the PB1DIV register cannot be written as a '0'.

Note 1: Writes to this register require an unlock sequence. Refer to **42.3.7.2 "Oscillator Switching Sequence"** for details.

2: This register is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

Register 42-11: SLEWCON: Oscillator Slew Control Register

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-cfg	R/W-cfg	R/W-cfg
	—	—	—	—	SLWDLY[3:0](1)			
23:16	U-0	U-0	U-0	U-0		R/W-0	R/W-0	R/W-0
	—	—	—	—	SYSDIV[3:0]			
15:8	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	SLWDIV[2:0]		
7:0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R-0, HS, HC
	—	—	—	—	—	UPEN	DNEN	BUSY

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 31-28 **Unimplemented:** Read as '0'

bit 24-27 **SLWDLY[3:0]:** Number of clocks generated at each slew step for a clock switch(1)

0000 = 1 clock is generated at each slew step
0001 = 2 clocks are generated at each slew step

1111 = 16 clocks are generated at each slew step

bit 20-23 **Unimplemented:** Read as '0'

bit 19-16 **SYSDIV[3:0]:** System Clock Divide Control bits

1111 = SYSCLK is divided by 16
1110 = SYSCLK is divided by 15
•
•
•
0010 = SYSCLK is divided by 3
0001 = SYSCLK is divided by 2
0000 = SYSCLK is not divided

bit 15-11 **Unimplemented:** Read as '0'

bit 10-8 **SLWDIV[2:0]:** Slew Divisor Steps Control bits

These bits control the maximum division steps used when slewing during a frequency change.
111 = Steps are divide by 128, 64, 32, 16, 8, 4, 2, and then no divisor
110 = Steps are divide by 64, 32, 16, 8, 4, 2, and then no divisor
101 = Steps are divide by 32, 16, 8, 4, 2, and then no divisor
100 = Steps are divide by 16, 8, 4, 2, and then no divisor
011 = Steps are divide by 8, 4, 2, and then no divisor
010 = Steps are divide by 4, 2, and then no divisor
001 = Steps are divide by 2, and then no divisor
000 = No divisor is used during slewing

bit 7-3 **Unimplemented:** Read as '0'

bit 2 **UPEN:** Upward Slew Enable bit

1 = Slewing enabled for switching to a higher frequency
0 = Slewing disabled for switching to a higher frequency

bit 1 **DNEN:** Downward Slew Enable bit

1 = Slewing enabled for switching to a lower frequency
0 = Slewing disabled for switching to a lower frequency

Note 1: This bit or register is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet to determine availability.

Register 42-11: SLEWCON: Oscillator Slew Control Register

bit 0 **BUSY:** Clock Switching Slewing Active Status bit
1 = Clock frequency is being actively slewed to the new frequency
0 = Clock switch has reached its final value

Note 1: This bit or register is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet to determine availability.

Register 42-12: CLKSTAT: Oscillator Clock Status Register

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	R-0	U-0	R-0	R-0	U-0	R-0	R-0	R-0
	SPLLRDY	—	LPRCRDY	SOSCRDY	—	POSCRDY	DIVSPLLRDY	FRCRDY

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 31-8 **Unimplemented:** Read as '0'

bit 7 **SPLLRDY:** System PLL Ready Status bit
- 1 = System PLL is running and locked
- 0 = System PLL is not running

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

bit 5 **LPRCRDY**: Low-Power RC (LPRC) Oscillator Ready Status bit
- 1 = LPRC is stable and ready
- 0 = LPRC is disabled or not operating

bit 4 **SOSCRDY:** Secondary Oscillator (SOSC) Ready Status bit
- 1 = SOSC is stable and ready
- 0 = SOSC is disabled or not operating

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

bit 2 **POSCRDY:** Primary Oscillator (POSC) Ready Status bit
- 1 = POSC is stable and ready
- 0 = POSC is disabled or not operating

bit 1 **DIVSPLLRDY:** Divided System PLL Ready Status bit
- 1 = Divided System PLL is ready
- 0 = Divided System PLL is not ready

bit 0 **FRCRDY:** Fast RC (FRC) Oscillator Ready Status bit
- 1 = FRC is stable and ready
- 0 = FRC is disabled for not operating

Note 1: This register or bit is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet to determine availability.

Register 42-13: CLKSTAT: Oscillator Clock Status Register (PIC32MZ W1 Only)

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	R/HS/HC-0	R/HS/HC-0	R/HS/HC-0	R/HS/HC-0
	—	—	—	—	SYSCLKRDY	PB1CLKRDY	SPLLALTRDY	WIFICLKRDY
7:0	R/HS/HC-0	R/HS/HC-0	R/HS/HC-0	R/HS/HC-0	R/HS/HC-0	R/HS/HC-0	R/HS/HC-0	R/HS/HC-0
	ETHPLLRDY	BTPLLRDY	LPRCRDY	SOSCRDY	UPLLRDY	POSCRDY	SPLLRDY	FRCRDY

Legend:	HC = Hardware Set	HC = Hardware Cleared	
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 31-12 **Unimplemented:** Read as '0'

bit 11 **SYSCLKRDY:** System Clock Ready Status bit
- 0 = SYSCLK is not stable and not ready
- 1 = SYSCLK is stable and ready

bit 10 **PB1CLKRDY:** PB1 Clock Ready Status bit
- 0 = PB1CLK is not stable and not ready
- 1 = PB1CLK is stable and ready

bit 9 **SPLLALTRDY:** System PLL Ready Status bit
- 0 = SPLL alternate output is not stable and not ready
- 1 = SPLL alternate output is stable and ready

bit 8 **WIFICLKRDY:** Wi-Fi Clock Ready Status bit
- 0 = WIFICLK is not stable and not ready
- 1 = WIFICLK is stable and ready

bit 7 **ETHPLLRDY:** ETHPLL Ready Status bit
- 0 = ETHPLL is not stable and not ready
- 1 = ETHPLL is stable and ready

bit 6 **BTPLLRDY:** Bluetooth PLL Ready Status bit
- 0 = BTPLL is not stable and not ready
- 1 = BTPLL is stable and ready

bit 5 **LPRCRDY:** LPRC Ready Status bit
- 0 = LPRC is not stable and not ready
- 1 = LPRC is stable and ready

bit 4 **SOSCRDY:** SOSC Ready Status bit
- 0 = SOSC is not stable and not ready
- 1 = SOSC is stable and ready

bit 3 **UPLLRDY:** USB PLL Ready Status bit
- 0 = UPLL is not stable and not ready
- 1 = UPLL is stable and ready

Note: This register is available only in the PIC32MZ W1 Family. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for more details.

Register 42-13: CLKSTAT: Oscillator Clock Status Register (PIC32MZ W1 Only) (Continued)

bit 2 **POSCRDY:** Primary Oscillator Ready Status bit
0 = POSC is not stable and not ready
1 = POSC is stable and ready

bit 1 **SPLLRDY:** System PLL Ready Status bit
0 = SPLL Primary output is not stable and not ready
1 = SPLL Primary output is stable and ready

bit 0 **FRCRDY:** FRC Ready Status bit
0 = FRC is not stable and not ready
1 = FRC is stable and ready

Note: This register is available only in the PIC32MZ W1 Family. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for more details.

42.3 OPERATION: CLOCK GENERATION AND CLOCK SOURCES

The PIC32 family of devices has multiple internal clocks that are derived from internal or external clock sources. Some of these clock sources have Phase-Locked Loops (PLLs), a programmable output divider, and/or an input divider, to scale the input frequency to suit the application. The clock source can be changed on-the-fly by software. The oscillator control register is locked by hardware, it must be unlocked by a series of writes before software can perform a clock switch.

There are three main clocks in a PIC32 device:

- System Clock (SYSCLK)
- One or more Peripheral Bus Clocks (PBCLKx)
- USB Clock (USBCLK)
- BT Clock(1)
- Ethernet_Wi-Fi Clock(1)

The PIC32 clocks are derived from one of the following sources:

- Primary Oscillator (POSC) on the OSC1 and OSC2 pins
- Secondary Oscillator (SOSC) on the SOSCI and SOSCO pins
- Internal Fast RC (FRC) Oscillator
- Internal Backup FRC (BFRC) Oscillator
- Internal Low-Power RC (LPRC) Oscillator

Note:	The BFRC Oscillator is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

Each of the clock sources has unique configurable options, such as a PLL, an input divider and/or output divider, which are detailed in their respective sections.

Note:	Clock sources for peripherals that use external clocks, such as the Real-Time Clock and Calendar (RTC) and Timer1, are covered in their respective family reference manual sections. Refer to **Section 14. "Timers"** (DS60001105) and **Section 29. "Real-Time Clock and Calendar"** (DS60001125) of the *"PIC32 Family Reference Manual"* for further details.

42.3.1 System Clock (SYSCLK) Generation

The SYSCLK is derived from one of the following four clock sources:

- POSC
- SOSC
- Internal FRC Oscillator
- LPRC Oscillator

Some of the clock sources have specific clock multipliers and/or divider options.

No clock scaling is applied, other than the user-specified values.

The SYSCLK source is selected by the device configuration and can be changed by software during operation. The ability to switch clock sources during operation allows the application to reduce power consumption by reducing the clock speed.

For devices with a SLEWCON register, the SYSCLK may be further divided before going into the system. Utilizing the SYSDIV[3:0] bits, SYSCLK may be divided up to a maximum of 16. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet to determine availability of the SLEWCON register.

For a list of SYSCLK sources and how they can be combined with the System PLL to produce the SYSCLK, refer to Table 42-2.

1. This clock is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet to determine availability.

Table 42-2: Clock Selection Configuration Bit Values

Oscillator Mode	Oscillator Source	PLLICLK	POSCMOD[1:0] (see Note 4)	FNOSC[2:0] (see Note 4)	See Note
FRC Oscillator with Postscaler (FRCDIV)	Internal	x	xx	111	**1, 2**
LPRC Oscillator	Internal	x	xx	101	**1**
SOSC (Timer1/RTCC)	Secondary	x	xx	100	**1**
POSC in HS mode with PLL Module (HSPLL)	Primary	0	10	001	**3**
POSC in EC mode with PLL Module (ECPLL)	Primary	0	00	001	**3**
POSC in HS mode	Primary	x	10	010	—
POSC in EC mode	Primary	x	00	010	—
FRC Oscillator with PLL Module (FRCPLL)	Internal	1	xx	001	**1**
FRC Oscillator with Postscaler (FRCDIV)	Internal	x	xx	000	**1**

Note 1: OSC2 pin function, as PBCLK1 out or digital I/O, is determined by the OSCIOFNC Configuration bit (DEVCFG1[10]). When the pin is not required by the oscillator mode, it may be configured for one of these options.

2: Default oscillator mode for an unprogrammed (erased) device.

3: When using the PLL modes, the input divider must be chosen such that the resulting frequency applied to the PLL is in the range that is specified in the **"Electrical Characteristics"** chapter in the specific device data sheet.

4: See the **"Special Features"** chapter in the specific device data sheet and **Section 32. "Configuration"** (DS60001124) of the *"PIC32 Family Reference Manual"* for information on the DEVCFG1 and DEVCFG2 Configuration registers.

42.3.1.1 PRIMARY OSCILLATOR (POSC)

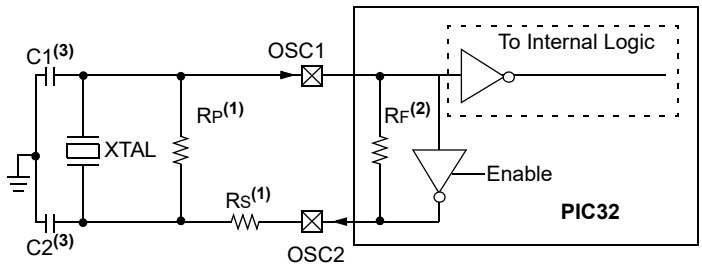
The Primary Oscillator (POSC) has four operating modes, as summarized in Figure 42-2, Table 42-3, Figure 42-4 show various configurations of the POSC.

Table 42-3: Primary Oscillator Operating Modes

Oscillator Mode	Description
HS	High-speed crystal
EC	External clock input
HSPLL	Crystal, PLL enabled
ECPLL	External clock input, PLL enabled

Note: The clock applied to the CPU, after applicable prescalers, postscalers, and PLL multipliers, must not exceed the maximum allowable processor frequency. Refer to the **"Electrical Characteristics"** chapter in the specific device data sheet for more information.

Figure 42-2: Crystal or Ceramic Resonator Operation (HS or HSPLL Oscillator Mode)



Note 1: A series resistor, Rs, may be required for AT strip cut crystals or to eliminate clipping. Alternatively, to increase oscillator circuit gain, add a parallel resistor, RP, with a value of 1 MΩ.

2: The internal feedback resistor, RF, is typically in the range of 2 MΩ to 10 MΩ.

3: See **42.7.3.1 "Determining the Best Values for Oscillator Components"**.

The POSC is connected to the OSC1 and OSC2 pins in this device family. The POSC can be configured for an external clock input, or an external crystal or resonator.

The HS and HSPLL modes are external crystal or resonator controller oscillator modes. OSC2 provides crystal/resonator feedback in HS Oscillator mode and is not available for use as an input or output in this mode. The HSPLL mode has a Phase-Locked Loop (PLL) with a user-selectable input divider and multiplier, and an output divider, to provide a wide range of output frequencies. The oscillator circuit will consume more current when the PLL is enabled.

The External Clock modes, EC and ECPLL, allow the SYSCLK to be derived from an external clock source. The EC/ECPLL modes configure the OSC1 pin as a high-impedance input that can be driven by a CMOS driver. The external clock can be used to drive the SYSCLK directly (EC) or the ECPLL module with prescaler and postscaler can be used to change the input clock frequency (ECPLL). The External Clock mode also disables the internal feedback buffer allowing the OSC2 pin to be used for other functions. In the External Clock mode, the OSC2 pin can be used as an additional device I/O pin (see Figure 42-3) or a PBCLK output pin (see Figure 42-4).

Note: When using the PLL modes, the input divider must be chosen such that the resulting frequency applied to the PLL is in the range that is specified in the **"Electrical Characteristics"** chapter in the specific device data sheet.

Figure 42-3: External Clock Input Operation with No Clock-Out (EC, ECPLL Mode)

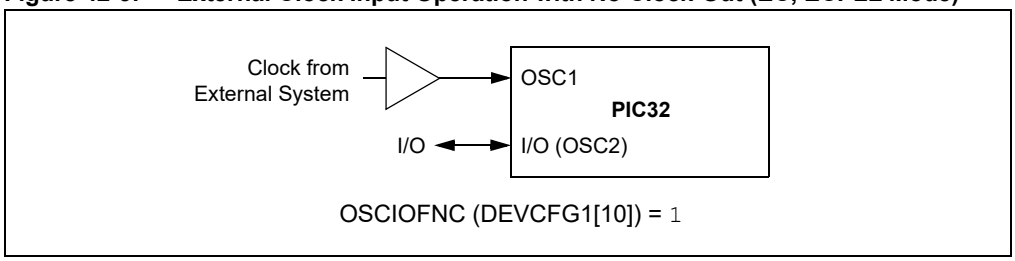
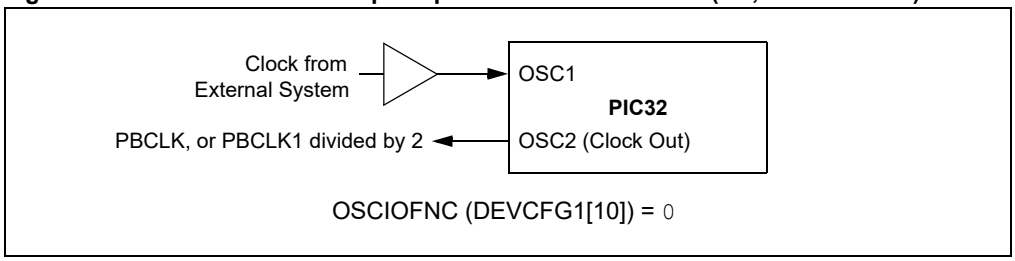


Figure 42-4: External Clock Input Operation with Clock-Out (EC, ECPLL Mode)



42.3.1.1.1 Primary Oscillator (POSC) Configuration

To configure the POSC, perform the following steps:

1. Select the POSC as the default oscillator in the device Configuration register DEVCFG1 by setting FNOSC[2:0] = 010 (without PLL) or to '001' (with PLL). If using the PLL, set the FPLLICLK bit in the DEVCFG2 register so that the SYSCLK is driven by the PLL and the POSC, instead of the PLL and FRC.
2. Select the desired mode, HS or EC, using the POSCMOD[1:0] bits in DEVCFG1.
3. If the PLL is to be used:
   a) Select the appropriate Configuration bits for the PLL input divider to scale the input frequency using the FPLLIDIV[2:0] in DEVCFG2. The input frequency must be in a range appropriate for the device. Refer to the **"Electrical Characteristics"** chapter in the specific device data sheet for details.
   b) Select the range of the frequency going into the multiplier using the FPLLRNG[2:0] bits (DEVCFG2[6:4]).
   c) Select the desired PLL multiplier ratio using FPLLMUL[2:0] in DEVCFG2.
   d) Select the desired PLL output divider using FPLLODIV (DEVCFG1[29:27]) to provide the desired SYSCLK frequency.
4. The values from the DEVCFGx locations are copied to the OSCCON and SPLLCON registers upon a device Reset. In addition, adjustments may be made during runtime by modifying these registers.

Note 1: Refer to the **"Special Features"** chapter in the specific device data sheet and **Section 32. "Configuration"** (DS60001124) of the *"PIC32 Family Reference Manual"* for information on the DEVCFG1 and DEVCFG2 Configuration registers.

2: An unlock sequence is required before a write to the OSCCON register can occur. Refer to **42.3.7.2 "Oscillator Switching Sequence"** for more information.

42.3.1.1.2 Oscillator Start-up Timer (OST)

In order to ensure that a crystal oscillator (or ceramic resonator) has started and stabilized, an Oscillator Start-up Timer (OST) is provided. The OST is a simple 10-bit counter that counts 1024 TOSC cycles before releasing the oscillator clock to the rest of the system. This time-out period is designated as TOST. 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 has to restart (i.e., on a Power-on Reset (POR), Brown-out Reset (BOR) or a wake-up from Sleep mode). The OST is applied to the HS mode for the POSC, and the SOSC (see **42.3.1.2 "Secondary Oscillator (SOSC)"**).

Note:	The oscillator start-up timer is disabled when POSC is configured for EC mode or ECPLL mode.

42.3.1.1.3 Primary Oscillator Start-up from Sleep Mode

To ensure reliable wake-up from Sleep, care must be taken to design the Primary Oscillator circuit. This is because the load capacitors have both partially charged to some quiescent value and phase differential at wake-up is minimal. Therefore, more time is required to achieve stable oscillation. Also, low voltage, high temperatures, and lower frequency clock modes also impose limitations on loop gain, which in turn, affects start-up.

Each of the following factors increases the start-up time:

- Low-frequency design (with a Low Gain Clock mode)
- Quiet environment (such as a battery operated device)
- Operating in a shielded box (away from the noisy RF area)
- Low voltage
- High temperature
- Wake-up from Sleep mode

42.3.1.1.4 Primary Oscillator Pin Functionality

The Primary Oscillator pins (OSCI/OSCO) can be used for other functions when the oscillator is not being used.

The POSCMD Configuration bits in the Oscillator Configuration (FOSC[1:0]) register determine the oscillator pin function.

The OSCIOFNC bit (DEVCFG1[10]) determines the OSC2 pin function. Refer to the **"Special Features"** chapter in the specific device data sheet for OSCIOFNC functionality.

42.3.1.2 SECONDARY OSCILLATOR (SOSC)

The Secondary Oscillator (SOSC) is designed specifically for low-power operation with an external 32.768 kHz crystal. The oscillator is located on the SOSCO and SOSCI device pins and serves as a secondary crystal clock source for low-power operation. It can also drive Timer1 and/or the Real-Time Clock and Calendar (RTCC) module for Real-Time Clock (RTC) applications.

42.3.1.2.1 Enabling the SOSC

The SOSC is hardware enabled by the FSOSCEN Configuration bit (DEVCFG1[5]). Refer to the **"Special Features"** chapter of the specific device data sheet and **Section 32. "Configuration"** (DS60001124) in the *"PIC32 Family Reference Manual"* for information on the DEVCFG1 Configuration register. The software can control the SOSC by modifying the SOSCEN bit (OSCCON[1]). Setting SOSCEN enables the oscillator; the SOSCO and SOSCI pins are controlled by the oscillator and cannot be used for port I/O or other functions.

Note:	An unlock sequence is required before a write to OSCCON can occur. Refer to **42.3.7.2 "Oscillator Switching Sequence"** for more information.

The SOSC requires a warm-up period before it can be used as a clock source. When the oscillator is enabled, a warm-up counter increments to 1024. When the counter expires the SOSCRDY bit (OSCCON[22]) is set to '1'. Refer to **42.3.1.1.2 "Oscillator Start-up Timer (OST)"**.

42.3.1.2.2 SOSC Continuous Operation

The SOSC is always enabled when SOSCEN bit (OSCCON[1]) is set. Leaving the oscillator running at all times allows a fast switch to the 32 kHz SYSCLK for lower power operation. Returning to the faster main oscillator will still require an oscillator start-up time if it is a crystal type source and/or uses the PLL (see **42.3.1.1.2 "Oscillator Start-up Timer (OST)"**).

In addition, the oscillator will need to remain running at all times for Real-Time Clock applications and may be required for Timer1. Refer to **Section 14. "Timers"** (DS60001105) and **Section 29. "Real-Time Clock and Calendar"** (DS60001125) of the *"PIC32 Family Reference Manual"* for further details.

Example 42-1: Enabling the Sosc

```
SYSKEY = 0x0;              // Ensure OSCCON is locked
SYSKEY = 0xAA996655;       // Write Key1 to SYSKEY
SYSKEY = 0x556699AA;       // Write Key2 to SYSKEY
                           // OSCCON is now unlocked
                           // Make the desired change
OSCCON |= 2;               // Enable Secondary Oscillator
                           // Relock the SYSKEY
SYSKEY = 0x0;              // Write any value other than Key1 or Key2
                           // OSCCON is relocked
```

42.3.1.3 SOSC EXTERNAL CLOCKING

The SOSCI pin can be driven by an external 32.768 kHz clock source instead of using a crystal. The SOSCEN bit must still be enabled; however, the SOSCO pin will be usable as an I/O pin.

42.3.1.4 INTERNAL FAST RC (FRC) OSCILLATOR

The FRC Oscillator is a fast (8 MHz nominal), user-trimmable, internal RC oscillator with a user-selectable input divider, PLL multiplier, and output divider. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for more information about the FRC Oscillator.

42.3.1.4.1 FRC Postscaler Mode (FRCDIV)

Users are not limited to the nominal 8 MHz FRC output if they want to use the fast internal oscillator as a clock source. An additional FRC mode, FRCDIV, implements a selectable output divider that allows the choice of a lower clock frequency from seven different options, plus the direct 8 MHz output. The output divider is configured using the FRCDIV[2:0] bits (OSCCON[26:24]). Assuming a nominal 8 MHz output, available lower frequency options range from 4 MHz (divide-by-2) to 31 kHz (divide-by-256). The range of frequencies allows users the ability to save power at any time in an application by simply changing the FRCDIV[2:0] bits.

42.3.1.4.2 FRC Oscillator with PLL Mode (FRCPLL)

The output of the FRC may also be combined with a user-selectable PLL multiplier and output divider to produce a SYSCLK across a wide range of frequencies. The FRC PLL mode is selected whenever the COSC[2:0] bits (OSCCON[14:12]) are '001'. The desired PLL multiplier and output divider values can be chosen to provide the desired device frequency.

42.3.1.4.3 Oscillator Tune Register (OSCTUN)

The FRC Oscillator Tuning register, OSCTUN, allows the user to fine tune the FRC Oscillator over a range of approximately ±12% (typical). Each bit increment or decrement changes the factory calibrated frequency of the FRC Oscillator by a fixed amount.

Note:	An unlock sequence is required before a write to OSCTUN register can occur. Refer to **42.3.7.2 "Oscillator Switching Sequence"** for more information.

42.3.1.5 INTERNAL BACKUP FAST RC (BFRC) OSCILLATOR

A dedicated Backup Fast RC oscillator (BFRC) is available on certain devices, which provides an 8 MHz clock when the FSCM detects a clock failure to support Class B operation. It is important to know that this oscillator is not user-selectable as the primary SYSCLK. The BFRC is only intended as a backup clock, and therefore, it is not factory calibrated to the accuracy of the FRC Oscillator. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

42.3.1.6 INTERNAL LOW-POWER RC (LPRC) OSCILLATOR

The LPRC Oscillator is separate from the FRC. It oscillates at a nominal frequency of 31.25 kHz. The LPRC Oscillator is the clock source for the Power-up Timer (PWRT), Watchdog Timer (WDT), Fail-Safe Clock Monitor (FSCM) and Phase-Locked Loop (PLL) reference circuits. It may 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.

42.3.1.6.1 Enabling the LPRC Oscillator

Since it serves as the PWRT clock source, the LPRC Oscillator is enabled at a POR whenever the on-board voltage regulator is enabled. After the PWRT expires, the LPRC Oscillator will remain ON if any one of the following is true:

- The Fail-Safe Clock Monitor is enabled
- The WDT is enabled
- The LPRC Oscillator is selected as the SYSCLK (COSC[2:0] = `100`)

If none of the above is true, the LPRC will shut off after the PWRT expires.

42.3.2 PLL Clock Generator

42.3.2.1 SYSTEM CLOCK PHASE-LOCKED LOOP (PLL)

The system clock PLL provides a user-configurable input divider and multiplier, and output divider, which can be used with the HS and EC POSC modes and with the Internal FRC Oscillator mode to create a variety of clock frequencies from a single clock source.

The input divider, multiplier, and output divider control initial value bits are contained in the DEVCFG2 Device Configuration register. The multiplier and output divider bits are also contained in the OSCCON register. As part of a device Reset, values from the device configuration register DEVCFG2 are copied to the OSCCON register. This allows the user to preset the input divider to provide the appropriate input frequency to the PLL and set an initial PLL multiplier when programming the device. At runtime, the multiplier and output divider can be changed by software to scale the clock frequency to suit the application.

To configure the PLL, the following steps are required:

1. Calculate the PLL input divider, range, multiplier, and output divider values.
2. Set the PLL input divider, input clock, range, and the initial multiplier and output divider values in the DEVCFG2 register when programming the device.
3. At runtime, all four settings can be changed in the SPLLCON register to suit the application.

Note: Refer to the **"Special Features"** chapter in the specific device data sheet and **Section 32. "Configuration"** (DS60001124) of the *"PIC32 Family Reference Manual"* for information on the DEVCFG2 Configuration register.

Example 42-2 shows a sample configuration setup for running the system at 200 MHz.

Example 42-2: Setup Example for 200 MHz Operation

```
#include <xc.h>

#pragma config POSCMOD = EC               // External Clock Mode
#pragma config FNOSC = SPLL               // System Clock is through System PLL
#pragma config FPLLICLK = PLL_POSC        // Input to PLL is from POSC
#pragma config FPLLIDIV = DIV_3           // Divide input by 3
#pragma config FPLLRNG = RANGE_5_10_MHz   // Input to PLL will be in 5-10 MHz
                                          // range after division
#pragma config FPLLMUL = MUL_50           // Multiply frequency by 50
#pragma config FPLLODIV = DIV_2           // Divide VCO output by 2
```

Combinations of the PLL input divider, multiplier, and output divider provide a combined multiplier of approximately 0.004 to 128 times the input frequency. For reliable operation, the output of the PLL module must not exceed the maximum clock frequency of the device. The PLL input divider value should be chosen to limit the input frequency to the PLL to the range that is appropriate for the device.

When the FRC is selected as the input to the PLL, the input divider is forced to the divide-by-1 setting, regardless of the setting of the FPLLIDIV or PLLIDIV bits.

42.3.2.2 PLL LOCK STATUS

Due to the time required for the PLL to provide a stable output, the SLOCK Status bit (OSCCON[5]) is provided. When the clock input to the PLL is changed, this bit is driven low ('0'). After the PLL has achieved a lock or the PLL start-up timer has expired, the bit is set. The bit will be set upon the expiration of the timer even if the PLL has not achieved a lock.

The PLL Lock Status indicates the lock status of the PLL. It is set automatically after a typical time delay for the PLL to achieve lock, also designated as TLOCK. If the PLL does not stabilize during start-up, SLOCK may not reflect the status of the PLL lock, nor does it detect when the PLL loses lock during normal operation. The SLOCK bit is cleared at a POR and on clock switches when the PLL is selected as a destination clock source. It remains clear when any clock source not using the PLL is selected. Refer to the **"Electrical Characteristics"** chapter in the specific device data sheet for further information on the PLL lock interval.

42.3.3 Peripheral Bus Clocks (PBCLKx) Generation

Certain PIC32 devices include more than one PBCLK, allowing peripherals to run at different bus speeds, depending on the application. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for details as to which devices have multiple PBCLKs, and the peripherals that are available on each peripheral bus.

42.3.3.1 PBCLKx SPEED CONTROL

Each PBCLK is derived from the SYSCLK divided by the PBCLKx Divisor bits PBDIV[6:0] (PBxDIVx[6:0]). The PBDIV[6:0] bits allow postscalers of 1:1 to 1:128.

The peripheral bus frequency can be changed on the fly by writing a new value to the PBDIV[6:0] bits in the PBxDIV register. A state machine is used to control the changing of the PB frequency. This state machine requires up to 60 CPU clocks to perform a switch and be ready to receive a new PBxDIV value. If a new value is written to the PBDIV[6:0] bits before the state machine has completed the operation, the new value will be ignored and the PBDIV[6:0] bits will reflect the previous value.

The PBDIVRDY bit (PBxDIV[11]) indicates whether a divisor switch is in progress during which time the PBDIV[6:0] bits should not be written. Rewriting the current value to the PBDIV[6:0] bits is ignored and has no effect.

Note:	When the PBDIV divisor is set to a ratio of 1:1, the SYSCLK and PBCLK are equivalent in frequency. The PBCLK frequency is never greater than the processor clock frequency. The effect of changing the PBCLK frequency on individual peripherals should be taken into account when selecting or changing the PBDIV value. Performing back-to-back operations on PBCLK peripheral registers when the PB divisor is not set at 1:1 will cause the CPU to stall for a number of cycles. This stall occurs to prevent an operation from occurring before the previous one has completed. The length of the stall is determined by the ratio of the CPU and PBCLK and synchronizing time between the two busses. Changing the PBCLK frequency has no effect on the operation of peripherals on the SYSCLK.

42.3.3.2 PBCLKx LIMITATIONS

Not all Peripheral Bus Clocks can operate at the same speed as the SYSCLK. Ensure that the PBCLKx does not exceed the maximum speed rating for that peripheral bus. Refer to the **"Electrical Characteristics"** chapter in the specific device data sheet to determine the maximum speed for each peripheral bus in your device.

42.3.3.3 PBCLKx ON/OFF CONTROL

In addition to speed, each Peripheral Bus Clock can also be turned ON and OFF as desired. This capability permits unused portions of the PIC32 device to be turned off to reduce active power consumption. Control of the PBCLKx is accomplished by setting or clearing the ON bit (PBxDIV[15]).

Note:	Some of the PB clocks control certain system functions that must not be disabled by removing the clock. Therefore, those clocks cannot be disabled and do not have an ON bit to control. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability of the PBCLKx ON/OFF control.

42.3.4 USB Clock (USBCLK) Generation

Note:	This feature is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

For PIC32 devices that support High-Speed USB, the 480 MHz USBCLK is generated within the USB PHY. It is not under software control; however, it is configurable through the DEVCFG2 register.

42.3.4.1 USB PLL LOCK STATUS

The ULOCK bit (OSCCON[6]) is a read-only Status bit that indicates the lock status of the USB PLL. It is automatically set after the typical time delay for the PLL to achieve lock. If the PLL does not stabilize properly during start-up, ULOCK may not reflect the actual status of PLL lock, nor does it detect when the PLL loses lock during normal operation.

42.3.4.2 CLOCK REQUIREMENTS(1)

For the HS USB PLL to operate at the correct 480 MHz frequency, the Primary Oscillator (POSC) must be used, and the crystal or clock source must be either 12 MHz or 24 MHz.

The USB PHY knows the speed value being provided through the UPLLFSEL bit (DEVCFG2[30]).

1. Some devices may have full speed USB with a different clock requirement. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for availability.

Refer to the **"Special Features"** chapter in the specific device data sheet and **Section 32. "Configuration"** (DS60001124) of the *"PIC32 Family Reference Manual"* for information on the DEVCFG2 Configuration register.

42.3.5 Two-Speed Start-up

Two-Speed Start-up mode can be used to reduce the device start-up latency when using all external crystal POSC modes including PLL. Two-Speed Start-up uses the FRC clock as the SYSCLK source until the Primary Oscillator (POSC) has stabilized. After the user selected oscillator has stabilized, the clock source will switch to POSC. This allows the CPU to begin running code, at a lower speed, while the oscillator is stabilizing. When the POSC has met the start-up criteria, an automatic clock switch occurs to switch to POSC. This mode is enabled by the IESO Configuration bit (DEVCFG1[7]).

Refer to the **"Special Features"** chapter in the specific device data sheet and **Section 32. "Configuration"** (DS60001124) of the *"PIC32 Family Reference Manual"* for information on the DEVCFG1 Configuration register. Two-Speed Start-up operates after a POR or on exit from Sleep. Software can determine the oscillator source currently in use by reading the COSC[2:0] bits (OSCCON[14:12]).

Note:	The Watchdog Timer (WDT), if enabled, will continue to count at the same rate regardless of the SYSCLK frequency. Care must be taken to service the WDT during Two-Speed Start-up, taking into account the change in SYSCLK.

42.3.6 Fail-Safe Clock Monitor (FSCM) Operation

The Fail-Safe Clock Monitor (FSCM) is designed to allow continued device operation if the current oscillator fails. The FSCM automatically switches the SYSCLK to an internal FRC oscillator if a failure is detected on the original clock source. The switch to an internal FRC oscillator allows continued device operation and the ability to retry the POSC or to execute code appropriate for a clock failure.

The FSCM mode is controlled by the FCKSM[1:0] bits in the DEVCFG1 register. Any of the POSC modes can be used with FSCM. Refer to the **"Special Features"** chapter in the specific device data sheet and **Section 32. "Configuration"** (DS60001124) of the *"PIC32 Family Reference Manual"* for information on the DEVCFG1 Configuration register.

When a clock failure is detected by the FSCM, a Non-Maskable Interrupt (NMI) is generated. The Interrupt Service Routine (ISR) attached to the NMI can read the CF bit in the RMNICON register to detect that the NMI was generated by the FSCM. Refer to the **"Resets"** chapter in the specific device data sheet for details.

Note:	All devices have at least one internal Fast RC oscillator, which is referred to as the FRC; however, depending on the device, another internal FRC oscillator, the Backup FRC (BRFC), is available that is dedicated to this fail-safe clock function

The FSCM module takes the following actions when a clock failure is detected:

1. The COSC[2:0] bits (OSCCON[14:12]) are loaded with '000' if no BFRC oscillator is present; otherwise '110' if a BFRC oscillator is present.
2. The CF bit (OSCCON[3]) is set to indicate the clock failure.
3. The OSWEN control bit (OSCCON[0]) is cleared to cancel any pending clock switches.

To enable the FSCM, set FCKSM[1:0] = 1 (DEVCFG1[15:14]) and set the desired SYSCLK configuration, as previously described in **42.3.1 "System Clock (SYSCLK) Generation"**.

42.3.6.1 FSCM DELAY

On a POR, BOR, or wake from a Sleep mode event, a nominal delay (TFSCM) may be inserted before the FSCM begins to monitor the SYSCLK source. The purpose of the FSCM delay is to provide time for the oscillator and/or PLL to stabilize when the Power-up Timer (PWRT) is not utilized. The FSCM delay will be generated after the internal System Reset signal, SYSRST, has been released. Refer to **Section 7. "Resets"** (DS60001118) of the *"PIC32 Family Reference Manual"* for FSCM delay timing information.

The TFSCM interval is applied whenever the FSCM is enabled and the HS, HSPLL, or SOSC modes are selected as the SYSCLK.

Note:	Please refer to the **"Electrical Characteristics"** chapter in the specific device data sheet for TFSCM specification values.

42.3.6.2 FSCM AND SLOW OSCILLATOR START-UP

If the chosen device oscillator has a slow start-up time coming out of POR, BOR or Sleep mode, it is possible that the FSCM delay will expire before the oscillator has started. In this case, the FSCM will initiate a clock failure trap. As this happens, the COSC[2:0] bits (OSCCON[14:12]) are loaded with the BFRC Oscillator selection. This will effectively shut off the original oscillator that was trying to start. Software can detect a clock failure using the NMI ISR or by polling the CF bit (OSCCON[3]).

42.3.6.3 FSCM AND WDT

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

42.3.6.4 SOFTWARE TRIGGER OF FSCM

Triggering a FSCM switchover sequence through software is useful for the purpose of testing how the PIC32 application handles this event without having to physically remove a clock source.

To trigger a FSCM event in software, execute the following procedure:

1. Unlock the OSCCON register for writing using the unlock sequence, which is described in **42.3.7.2 "Oscillator Switching Sequence"**.
2. Write a '1' to the CF bit (OSCCON[3]).

The NMI ISR will then be entered, once the switchover has completed.

It is also possible to trigger the NMI ISR without actually switching the clock to the BFRC. This is done by writing a '1' to the CF bit in the RNMICON register. Refer to **Section 6. "Resets"** (DS60001112) in the *"PIC32 Family Reference Manual"* for information.

42.3.6.5 CLEARING A FSCM EVENT CONDITION

The NMI handler procedure (`_nmi_handler`), which is available through the MPLAB® XC32 C Compiler, can be used to attempt a restart of the main oscillator. To return to the point of execution where the FSCM event occurred, use an `ERET` instruction.

42.3.7 Clock Switching Operation

With few limitations, applications are free to switch between any of the four clock sources (POSC, SOSC, FRC, and LPRC) under software control and at any time. To limit the possible side effects that could result from this flexibility, PIC32 devices have a safeguard lock built into the switch process.

Note 1:	Primary Oscillator mode has two different submodes (HS, and EC) which are determined by the POSCMOD Configuration bits in DEVCFG1. While an application can switch to and from Primary Oscillator mode in software, it cannot switch between the different primary submodes without reprogramming the device. Refer to the **"Special Features"** chapter in the specific device data sheet and **Section 32. "Configuration"** (DS60001124) of the *"PIC32 Family Reference Manual"* for information on the DEVCFG1 Configuration register.
2:	The user application should not change the PLL multiplier, prescaler, or postscaler values when running from the affected PLL source. To perform any of these clock switching functions, the clock switch should be performed in two steps. The clock source should first be switched to a non-PLL source, such as FRC, and then switched to the desired source. This requirement only applies to PLL-based clock sources.

42.3.7.1 ENABLING CLOCK SWITCHING

To enable clock switching, the FCKSM[0] Configuration bit (DEVCFG1[14]) must be programmed to '1'.

The NOSC[2:0] Control bits (OSCCON[10:8]) do not control the clock selection when clock switching is disabled. However, the COSC[2:0] bits (OSCCON[14:12]) will reflect the clock source selected by the FNOSC[2:0] Configuration bits.

The OSWEN Control bit (OSCCON[0]) has no effect when clock switching is disabled. It is held at '0' at all times.

42.3.7.2 OSCILLATOR SWITCHING SEQUENCE

At a minimum, performing a clock switch requires the following sequence:

1. If desired, read the COSC[2:0] bits (OSCCON[14:12]) to determine the current oscillator source.
2. Perform the unlock sequence to allow a write to the OSCCON register. The unlock sequence has critical timing requirements and should be performed with interrupts and DMA disabled.
3. Write the appropriate value to the NOSC[2:0] control bits (OSCCON[10:8]) for the new oscillator source.
4. Set the OSWEN bit (OSCCON[0]) to initiate the oscillator switch.
5. Optionally, perform the lock sequence to lock the OSCCON register. The lock sequence must be performed separately from any other operation.

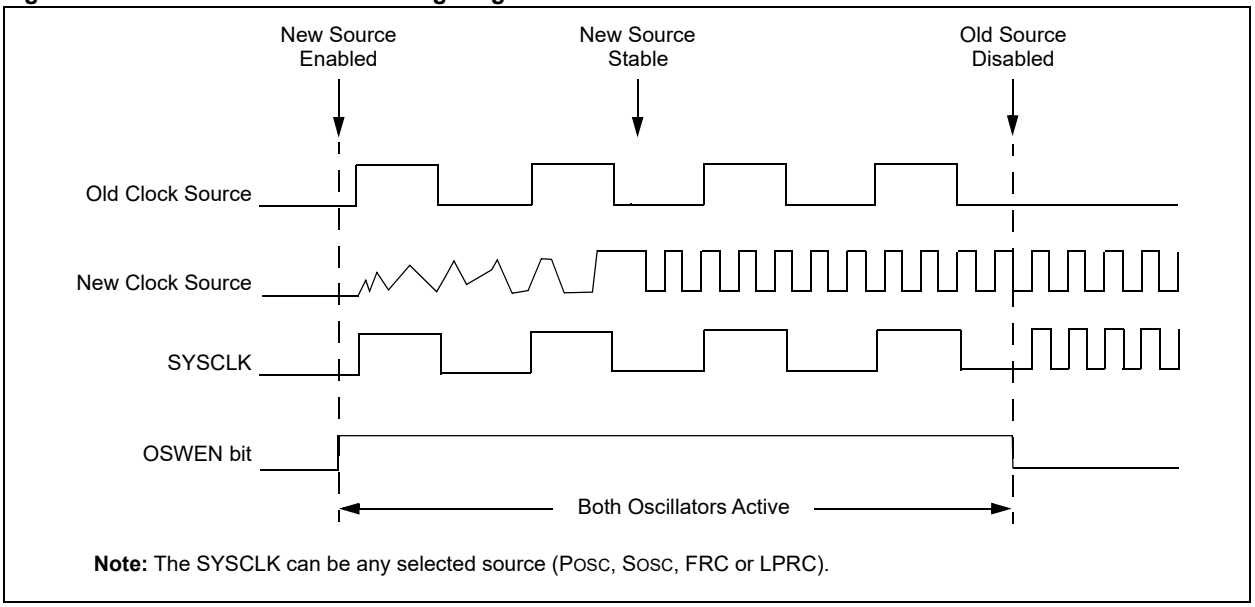
After the basic sequence is completed, the SYSCLK hardware responds automatically as follows:

1. The clock switching hardware compares the COSC[2:0] Status bits with the new value of the NOSC[2:0] control bits. If they are the same, then the clock switch is a redundant operation. In this case, the OSWEN bit is cleared automatically and the clock switch is aborted.
2. The new oscillator is turned on by the hardware if it is not currently running. If a crystal oscillator must be turned on, the hardware will wait until the Oscillator Start-up timer (OST) expires. If the new source is using the PLL, then the hardware waits until a PLL lock is detected (SLOCK = 1).
3. The hardware clears the OSWEN bit to indicate a successful clock transition. In addition, the NOSC[2:0] bit values are transferred to the COSC[2:0] Status bits.
4. The old clock source is turned off at this time if the clock is not being used by any modules.

The transition timing between the clock sources is shown in Figure 42-5.

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

Figure 42-5: Clock Transition Timing Diagram



Note: The SYSCLK can be any selected source (POSC, SOSC, FRC or LPRC).

The following is a recommended code sequence for a clock switch:

1. Disable interrupts and DMA prior to the system unlock sequence.
2. Execute the system unlock sequence by writing the Key values of 0xAA996655 and 0x556699AA to the SYSKEY register in two back-to-back Assembly or 'C' instructions.
3. Write the new oscillator source value to the NOSC[2:0] control bits.
4. Set the OSWEN bit in the OSCCON register to initiate the clock switch.
5. Write a non-key value (such as, 0x33333333) to the SYSKEY register to perform a lock. Continue to execute code that is not clock-sensitive (optional).
6. Check if the OSWEN bit is '0'. If it is, the switch was successful. Loop until the bit is '0'.
7. Re-enable interrupts and DMA.

Note: There are no timing requirements for the steps other than the initial back-to-back writing of the Key values to perform the unlock sequence.

The unlock sequence unlocks all registers that are secured by the lock function. It is recommended that the amount of time the system is unlocked is kept to a minimum. For example code for unlocking the OSCCON register, refer to Example 42-1.

42.3.7.3 CLOCK SWITCHING CONSIDERATIONS

When incorporating clock switching into an application, consider the following issues when designing their code:

- The SYSLOCK unlock sequence is timing critical. The two key values must be written back-to-back with no in-between peripheral register access. Prevent unintended peripheral register accesses by disabling all interrupts and DMA transfers.
- The system will not relock automatically. Perform the relock sequence as soon as possible after the clock switch
- The unlock sequence unlocks other registers such as the those related to Real-Time Clock control
- If the destination clock source is a crystal oscillator, the clock switch time is dictated by the oscillator start-up time
- If the new clock source does not start, or is not present, the OSWEN bit remains set
- A clock switch to a different frequency affects the clocks to peripherals. Peripherals may require reconfiguration to continue operation at the same rate as they did before the clock switch occurred
- If the new clock source uses the PLL, a clock switch does not occur until lock has been achieved
- If the WDT is used, care must be taken to ensure it can be serviced in a timely manner at the new clock rate

Note 1: When the Fail-Safe Clock Monitor is enabled, the application should not attempt to switch to a clock that has a frequency lower than 100 kHz. Clock switching in these instances may generate a false oscillator fail event and result in a switch to the FRC oscillator or the BFRC oscillator.

2: The user application should not change the PLL multiplier, prescaler, or postscaler values when running from the affected PLL source. To perform either of the above clock switching functions, the clock switch should be performed in two steps. The clock source should first be switched to a non-PLL source, such as FRC; and then, switched to the desired source. This requirement only applies to PLL-based clock sources.

42.3.7.4 ENTERING SLEEP MODE DURING A CLOCK SWITCH

If, during a clock switch operation, the device enters Sleep mode, the clock switch operation is not aborted. If the clock switch does not complete before the device enters Sleep mode, the device will perform the switch when it exits Sleep, and then the code after the `WAIT` instruction executes normally

42.3.7.5 CLOCK SWITCH SLEWING

Note:	This feature is not available on all devices. Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet to determine availability.

Clock slewing can be used to reduce power spikes when the Oscillator changes frequencies from one speed to another. With clock slewing, the rate at which the frequency changes can be controlled, both when changing from faster to slower speeds, and from slower to faster speeds.

This feature only operates during clock switches initiated by software, or when the device wakes from Sleep mode. Hardware clock switches are handled automatically, and do not use the settings from the SLEWCON register, except in the case of Two-Speed Start-up from FRC to NOSC.

Setting up the clock slewing is done through the SLEWCON register (Register 42-11). The SYSDIV[3:0] bits control the final output of the SYSCLK after slewing is complete. The SLWDIV[2:0] bits control the number of steps taken when a clock change takes place. The UPEN bit enables clock slewing when changing to a faster clock, and the DNEN bit enables clock slewing when changing to a slower clock. The BUSY bit is a status bit, indicating that a clock slew is active.

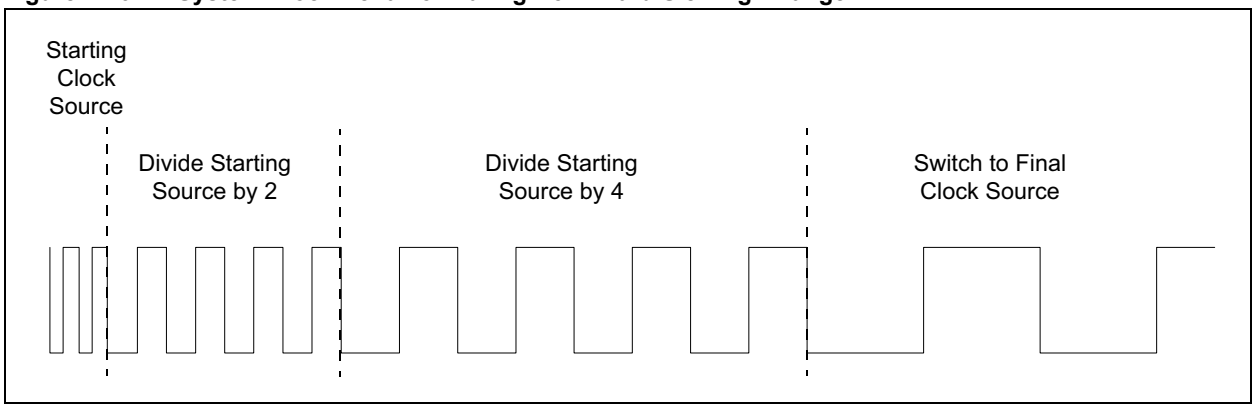
42.3.7.5.1 Clock Slewing to a Slower Clock Speed

When a clock slew to a slower clock speed takes place, these steps occur within the Clock module:

1. The BUSY bit is set to '1'.
2. The current SYSCLK source is initially divided by 2.
3. Wait for the new clock source to be ready.
4. After four clocks at this frequency, the clock divider is set to decrease frequency by 2x.
5. Step 4 is repeated until the clock divider reaches SYSDIV * SLWDIV.
6. The clock source is switched to the slower clock source, and the clock divider is set to SYSDIV.
7. The BUSY bit is cleared.
8. The clock divider remains at the SYSDIV value until the next clock change or reset event.

Figure 42-6 illustrates the system clock behavior during a downward slewing change using a two-step slew (i.e., SLWDIV[3:0] = 010).

Figure 42-6: System Clock Behavior During Downward Slewing Change



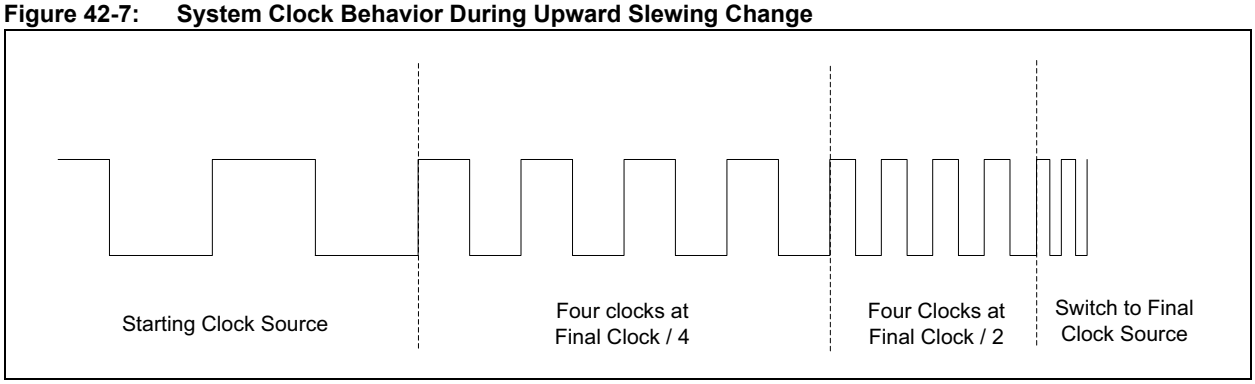
42.3.7.5.2 Clock Slewing to a Faster Clock Speed

When a clock slew to a faster clock speed takes place, these steps occur within the Clock module:

1. The BUSY bit is set to '1'.
2. Wait for the new clock source to be ready.
3. Switch to the new clock source, with the clock divider set to SYSDIV * SLWDIV.
4. After four clocks at this frequency, the clock divider is set to increase frequency by 2x.
5. Step 4 is repeated until the clock divider equals the SYSDIV value.
6. The BUSY bit is cleared.
7. The clock divider remains at the new value of SYSDIV until the next clock change or reset event.

Figure 42-7 illustrates the system clock behavior during an upward slewing change using a two-step slew (i.e., SLWDIV[3:0] = 010).

Figure 42-7: System Clock Behavior During Upward Slewing Change



42.3.8 Real-Time Clock Oscillator

To provide accurate timekeeping, the Real-Time Clock and Calendar (RTCC) requires a precise time base. To achieve this, the SOSC is used as the time base for the RTCC. The SOSC uses an external 32.768 kHz crystal connected to the SOSCI and SOSCO pins.

42.3.8.1 SOSC CONTROL

The SOSC can be used by modules other than the RTCC; therefore, the SOSC is controlled by a combination of software and hardware. Setting the SOSCEN bit (OSCCON[1]) to '1' enables the SOSC. The SOSC is disabled when it is not being used by the CPU module and the SOSCEN bit is '0'. If the SOSC is being used as SYSCLK, such as after a clock switch, it cannot be disabled by writing a '0' to the SOSCEN bit. If the SOSC is enabled by the SOSCEN bit, it will continue to operate when the device is in Sleep. To prevent inadvertent clock changes, the OSCCON register is locked. It must be unlocked prior to software enabling or disabling the SOSC.

Note: If the RTCC is to be used when the CPU clock source is to be switched between SOSC and another clock source, the SOSCEN bit should be set to '1' in software. Failure to set the bit will cause the SOSC to be disabled when the CPU is switched to another clock source.

Due to the start-up time for an external crystal, the user should wait for stable SOCSC oscillator output before enabling the RTCC. Refer to the device data sheet for the external crystal for information on the crystal's start-up time. Once the clock is stable, 256 cycles (approximately 8 ms) must pass before the RTCC module is turned on. The actual time required will depend on the crystal in use and the application.

There are numerous system and peripheral registers that are protected from inadvertent writes by the SYSREG lock. Performing a lock or unlock affects access to all registers protected by SYSREG including the OSCCON register.

42.3.8.2 LPRC CONTROL

On certain PIC32 devices, the RTCC module can also be driven by the LPRC oscillator. If this is done, the timing of the RTCC module will not be accurate for date/time purposes.

42.3.9 Timer1 External Oscillator

Timers can be clocked using an external signal or the TxCK pins (where 'x' is the number of the Timer module). In addition, the Timer1 module has the ability to use the SOSC as a clock source to increment Timer1. The SOSC is designed to use an external 32.768 kHz crystal connected to the SOSCI and SOSCO pins.

42.3.9.1 SOSC CONTROL

The SOSC can be used by modules other than Timer1, therefore, the SOSC is controlled by a combination of software and hardware. Setting the SOSCEN bit (OSCCON[1]) to '1' enables the SOSC. The SOSC is disabled when it is not being used by the CPU module and the SOSCEN bit is '0'. If the SOSC is being used as SYSCLK, such as after a clock switch, it cannot be disabled by writing a '0' to the SOSCEN bit. If the SOSC is enabled by the SOSCEN bit, it will continue to operate when the device is in Sleep. To prevent inadvertent clock changes the OSCCON register is locked. It must be unlocked prior to software enabling or disabling the SOSC.

Note: If the Timer1 module is to be used when the CPU clock source is to be switched between SOSC and another clock source, the SOSCEN bit should be set to '1' in software.

Due to the start-up time for an external crystal the user should wait for stable SOSC output before attempting to use Timer1 for accurate measurements. Refer to the data sheet for the external crystal for information on the crystal's start-up time. Once the clock is stable, 256 cycles (approximately 8 ms) must pass before the Timer1 module is turned on. The actual time required will depend on the crystal in use and the application.

There are numerous system and peripheral registers that are protected from inadvertent writes by the SYSREG lock. Performing a lock or unlock affects access to all registers protected by SYSREG including the OSCCON register.

42.3.10 Reference Clock Output

The reference clock output provides a clock signal on the REFCLKOx pin. The reference clock can be selected from various clock sources.

Depending on the PIC32 device, these sources may include one of the following:

- External REFCLKI pin
- Internal FRC oscillator
- Internal LPRC oscillator
- Sosc
- PBCLK1
- SYSCLK
- BFRC
- BTPLL Clock
- EWPLL Wi-Fi Clock
- EWPLL Ethernet Clock
- USB PLL

The ROSEL[3:0] bits (REFOxCON[3:0]) select between these sources.

After the clock source has been selected, it may be further divided by using the RODIV[14:0]) bits (REFOxCON[30:16]) and the ROTRIM[8:0] bits (REFOxTRIM[31:23]). The formula for determining the final frequency output is shown in Equation 42-1.

Equation 42-1: Calculating Final Frequency Output

$$F_{REFOUT} = \frac{F_{REFIN}}{2 \cdot \left(N + \frac{M}{512}\right)}$$

Where:
F_{REFOUT} = Output Frequency
F_{REFIN} = Input Frequency
N = RODIV[14:0]
M = ROTRIM[8:0]
When $N = 0$, the initial clock is the same as the input clock

For example, for an input frequency of 100 MHz, an N of 5 and an M of 256, the resulting frequency would be:

$$F_{REFOUT} = \frac{100MHz}{2 \cdot \left(5 + \frac{256}{512}\right)} \cong 9.091MHz$$

Refer to Figure 42-1 for a block diagram of the reference clock. See the REFOxCON register (Register 42-8) for the bits associated with the reference clock output.

Note:	This feature is not available on all devices. See the **"Oscillators"** chapter in the specific device data sheet for availability.

42.3.10.1 REFERENCE CLOCK OPERATION

The fractional part of the REFCLK divider, the ROTRIM bit in the REFOxTRIM register (Register 42-9), is actually an averaging circuit, meaning that it does not actually produce a fractional frequency of the input source clock. Instead, the circuit steals REFCLK input cycles so that the resulting average clock output frequency over a period of time averages out to the expected frequency. How often the trim circuit steals REFCLK input cycles is dependent on the fractional remainder of the desired REFCLK trim bits, (i.e. ((2 * ROTRIM[8:0]) / 512)).

Referring to Figure 42-8, how often the cycle steals occur increases from a minimum of 0 for a X.0 fractional trim remainder to a maximum as the fractional trim clock remainder approaches the midpoint X.5, and then begins to decrease again as the remainder approaches an integer value. In summary, the frequency of the cycle steals decreases as the trim fractional remainder approaches an integer value and increases the further away from an integer value with X.5 being the worst case. The cycle steals are symmetrical, meaning it is the same for X.2 as X.8 or X.4 as X.6, and so on.

The following guidelines will assist in controlling the amount of jitter when using REFCLK.

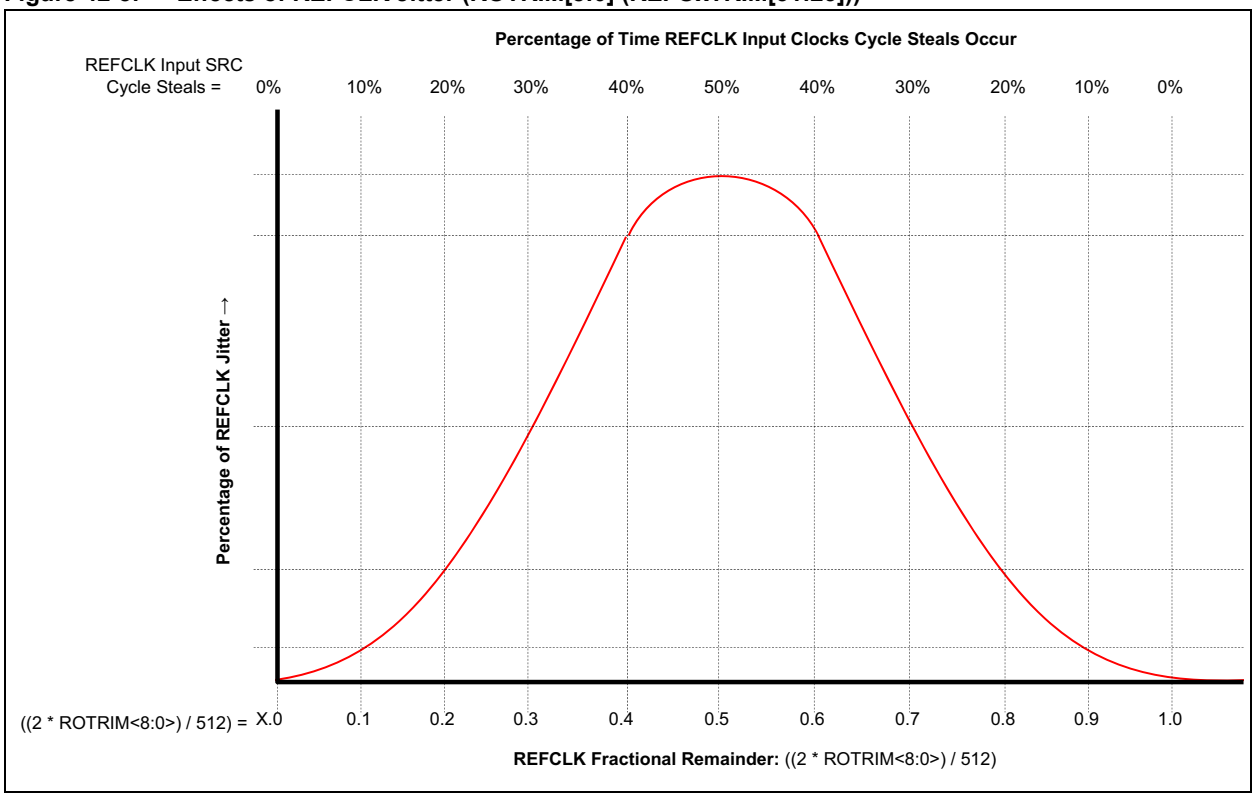
1. As the value of the integer portion “N” of the divider (RODIV[14:0] bits (REFOxCON[30:16])) increases for a given trim value (ROTRIM[8:0] bits (REFOxTRIM[31:23])), the REFCLK jitter percentage will decrease. This is due to the fact that as the REFCLK output clock period becomes larger relative to the REFCLK input source period where the cycle steals occurs, it becomes a smaller percentage. Therefore, the jitter percentage is less relative to the REFCLK output.
2. Conversely, as the frequency of the REFCLK input clock source, (i.e., ROSEL[3:0] bits (REFOxCON[3:0]), increases for a given (RODIV[14:0]), the REFCLK jitter percentage will also decrease due to the same effect as stated in item 1.
3. REFCLKO = ROSEL[3:0] Clock source / (2 (N + (M / 512)). Fractional divider component, Trim = (2 * M / 512) = ((2 * ROTRIM[8:0]) / 512).

   Therefore, for values of:

   - ROTRIM[8:0] bits = 0x000, no REFCLKO jitter, which equates to an even integer: ((2 * 0) / 512) = 0.0.
   - ROTRIM[8:0] bits = 0x100, no REFCLKO jitter, which equates to an even integer: ((2 * 256) / 512) = 1.0.
   - ROTRIM[8:0] bits = 0x1FF, almost no REFCLKO jitter, which equates to X.996, and is as close to an integer as possible without actually being an integer.

   For all other values of ROTRIM[8:0], the decimal remainder relative to 0.5, (i.e., max jitter), will cause decreasing amounts of jitter
4. REFCLKO jitter, (i.e., I^2S MCLK), is not a function of either the I^2S word size, 16 bits or 24 bits, and/or the sample rate, (i.e., audio quality). It is purely a function of the fractional ((2 * ROTRIM[8:0]) / 512 Trim decimal remainder.
5. The absolute maximum frequency deviation is always (1 / REFCLK input clock source freq), but how frequently the deviations occur per unit time, is a function of the trim decimal remainder, ((2 * ROTRIM[8:0]) / 512). See Figure 42-8 for an example.
6. Using a PIC32 crystal value or external user REFCLKI clock input that is a multiple of the desired REFCLKO as the input clock to the REFCLK circuit will eliminate almost all the jitter except the system PLL jitter, which is minimal if using a PLL clock derivative.
7. If ((2 * ROTRIM[8:0]) / 512) = X.5, every other REFCLK input clock source will be stolen, (i.e., cycle steals). Conversely, if X.1, 10 percent of the time a REFCLK input clock will be stolen. The more frequent the input source clock cycle steals, the higher the observed jitter percentage. Using the recommendation described in item 5, the user can minimize or eliminate REFCLK jitter.

Figure 42-8: Effects of REFCLK Jitter (ROTRIM[8:0] (REFOxTRIM[31:23]))



42.4 INTERRUPTS

The only interrupt generated by the Oscillator module is a Non-maskable Interrupt (NMI), which is detectable through a special Clock Fail Detect (CF) bit in the RNMICON register. This will be triggered when the Oscillator module detects that the COSC source has stopped clocking.

42.4.1 FSCM Non-Maskable Interrupt

The ISR attached to the NMI (i.e., `_nmi_handler`) can read the CF bit (RNMICON[1]) to detect that the FSCM has generated the NMI. Refer to the **"Resets"** chapter in the specific device data sheet for information on the RNMICON register and to determine whether the CF bit is available on your device.

42.5 OPERATION IN POWER-SAVING MODES

42.5.1 Oscillator Operation in Sleep Mode

Clock sources are disabled in Sleep unless they are being used by a peripheral. The following sections outline the behavior of each of the clock sources in Sleep mode.

42.5.1.1 PRIMARY OSCILLATOR IN SLEEP MODE

The POSC is always disabled in Sleep. Start-up delays apply when exiting Sleep.

42.5.1.2 SECONDARY OSCILLATOR IN SLEEP MODE

The SOSC is disabled in Sleep unless the SOSCEN bit is set or it is in use by an enabled module that operates in Sleep. Start-up delays apply when exiting Sleep if the SOSC is not already running.

42.5.1.3 FAST RC OSCILLATOR IN SLEEP MODE

The FRC oscillator is disabled in Sleep.

42.5.1.4 LOW-POWER OSCILLATOR IN SLEEP MODE

The LPRC Oscillator is disabled in Sleep if the WDT is disabled.

42.5.2 Oscillator Operation in Dream Mode

With Dream mode enabled, DMA interrupts with a priority higher than the priority of the CPU causes the device to wake up and change to Run mode. After completion of the activating data transfer, code execution will continue with the next instruction after the instruction that placed the device into Sleep mode. Therefore, prior to the DMA interrupt, the device is in Sleep Mode and oscillator behavior is the same as in Sleep Mode (see **42.5.1.1 "Primary Oscillator in Sleep Mode"** for details). After the DMA interrupt the device exits Sleep Mode and enters Run mode.

If the DMA interrupt is lower in priority than the priority of the CPU, the data transfer will cause the device to wake up into Idle mode. Memory transfers are then made without waking the CPU and the device will return to Sleep mode. Therefore, prior to the DMA interrupt, the device is in Sleep Mode and oscillator behavior is the same as in Sleep Mode (see **42.5.1.1 "Primary Oscillator in Sleep Mode"** for details). After the DMA interrupt, the device enters Idle mode. When the DMA transfer is complete the device re-enters Sleep Mode.

42.5.3 Oscillator Operation in Idle Mode

Clock sources are not disabled in Idle mode. Start-up delays do not apply when exiting Idle mode.

42.5.4 Oscillator Operation in Debug Mode

The Oscillator module continues to operate while the device is in Debug mode.

42.6 EFFECTS OF VARIOUS RESETS

On all forms of device Reset, OSCCON is set to the default value, and the COSC[2:0], PLLIDIV[2:0], PLLMULT[6:0], PLLODIV[2:0], PLLICLK, and PLLRANGE[2:0] bit values are forced to the values defined in the DEVCFG1 and DEVCFG2 registers. The oscillator source is transferred to the source as defined in the DEVCFG1 register. Oscillator start-up delays will apply.

Note:	Refer to the **"Special Features"** chapter in the specific device data sheet and **Section 32. "Configuration"** (DS60001124) of the *"PIC32 Family Reference Manual"* for information on the DEVCFG1 and DEVCFG2 Configuration registers.

42.7 CLOCKING GUIDELINES

42.7.1 Crystal Oscillators and Ceramic Resonators

In HS mode, a crystal or ceramic resonator is connected to the OSC1 and OSC2 pins to establish oscillation. The PIC32 oscillator design requires the use of a parallel cut crystal. Using a series cut crystal may give a frequency out of the crystal manufacturer's specifications.

In general, users should select the oscillator option with the lowest possible gain that still meets their specifications. This will result in lower dynamic currents (I_{DD}). The frequency range of each oscillator mode is the recommended frequency cut-off, but the selection of a different gain mode is acceptable as long as a thorough validation is performed (voltage, temperature and component variations, such as resistor, capacitor and internal oscillator circuitry).

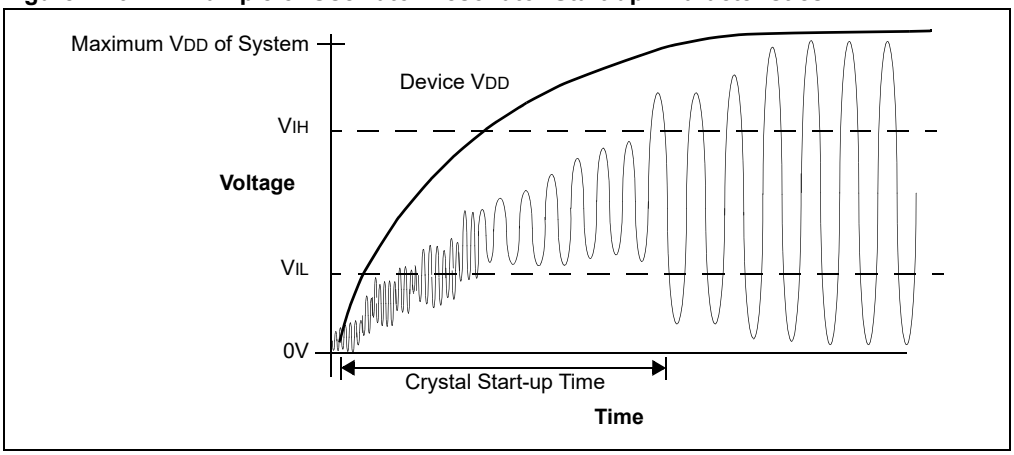
42.7.2 Oscillator/Resonator Start-up

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

The course of a typical crystal or resonator start-up is shown in Figure 42-9. Notice that the time to achieve stable oscillation is not instantaneous. Refer to the **"Electrical Characteristics"** chapter in the specific device data sheet for further information regarding frequency range for each crystal mode.

Figure 42-9: Example of Oscillator/Resonator Start-up Characteristics



42.7.3 Tuning the Oscillator Circuit

Since Microchip devices have wide operating ranges (frequency, voltage and temperature; depending on the part and version ordered) and external components (crystals, capacitors, etc.) of varying quality and manufacture, validation of operation needs to be performed to ensure that the component selection will comply with the requirements of the application. There are many factors that go into the selection and arrangement of these external components. Depending on the application, these may include one of the following:

- Amplifier gain
- Desired frequency
- Resonant frequency of the crystal
- Temperature of operation
- Supply voltage range
- Start-up time
- Stability
- Crystal life
- Power consumption
- Simplification of the circuit
- Use of standard components
- Component count

42.7.3.1 DETERMINING THE BEST VALUES FOR OSCILLATOR COMPONENTS

The best method for selecting components is to apply a little knowledge and a lot of trial measurement and testing. Crystals are usually selected by their parallel resonant frequency only; however, other parameters may be important to your design, such as temperature or frequency tolerance. The Microchip application note, AN588 *"PIC® Microcontroller Oscillator Design Guide"* (DS00000588), is an excellent reference from which to learn more about crystal operation and ordering information.

The PIC32 internal oscillator circuit is a parallel oscillator circuit which requires that a parallel resonant crystal be selected. The load capacitance is usually specified in the 22 pF to 33 pF range. The crystal will oscillate closest to the desired frequency with a load capacitance in this range. It may be necessary to alter these values, as described later, in order to achieve other benefits.

C1 and C2 should be initially selected based on the load capacitance, as suggested by the crystal manufacturer, and the tables supplied in the device data sheet. The values given in the device data sheet can only be used as a starting point since the crystal manufacturer, supply voltage, PCB layout and other factors already mentioned may cause your circuit to differ from those used in the factory characterization process.

Ideally, the capacitance is chosen so that it will oscillate at the highest temperature and the lowest VDD that the circuit will be expected to perform under. High-temperature and low VDD both have a limiting effect on the loop gain, such that if the circuit functions at these extremes, the designer can be more assured of proper operation at other temperatures and supply voltage combinations. The output sine wave should not be clipped in the highest gain environment (highest VDD and lowest temperature) and the sine output amplitude should be large enough in the lowest gain environment (lowest VDD and highest temperature) to cover the logic input requirements of the clock as listed in the specific device data sheet.

A method for improving start-up is to use a value of C2 that is greater than the value of C1. This causes a greater phase shift across the crystal at power-up which speeds oscillator start-up. Besides loading the crystal for proper frequency response, these capacitors can have the effect of lowering loop gain if their value is increased. C2 can be selected to affect the overall gain of the circuit. A higher C2 can lower the gain if the crystal is being overdriven (also, see discussion on Rs). Capacitance values that are too high can store and dump too much current through the crystal, so C1 and C2 should not become excessively large. Measuring the wattage through a crystal is difficult, but if you do not stray too far from the suggested values you should not have to be concerned with this.

A series resistor, Rs, is added to the circuit if, after all other external components are selected to satisfaction, the crystal is still being overdriven. This can be determined by looking at the OSC2 pin, which is the driven pin, with an oscilloscope. Connecting the probe to the OSC1 pin will load the pin too much and negatively affect performance. Remember that a scope probe adds its own capacitance to the circuit, so this may have to be accounted for in your design (i.e., if the circuit worked best with a C2 of 22 pF and the scope probe was 10 pF, a 33 pF capacitor may actually be called for). The output signal should not be clipping or flattened. Overdriving the crystal can also lead to the circuit jumping to a higher harmonic level, or even, crystal damage.

The OSC2 signal should be a clean sine wave that easily spans the input minimum and maximum of the clock input pin. An easy way to set this is to again test the circuit at the minimum temperature and maximum V_{DD} that the design will be expected to perform in, then look at the output. This should be the maximum amplitude of the clock output. If there is clipping, or the sine wave is distorted near V_{DD} and V_{SS}, increasing load capacitors may cause too much current to flow through the crystal or push the value too far from the manufacturer's load specification. To adjust the crystal current, add a trimmer potentiometer between the crystal inverter output pin and C2 and adjust it until the sine wave is clean. The crystal will experience the highest drive currents at the low temperature and high V_{DD} extremes.

The trimmer potentiometer should be adjusted at these limits to prevent overdriving. A series resistor, Rs, of the closest standard value can now be inserted in place of the trimmer. If Rs is too high, perhaps more than 20 kΩ, the input will be too isolated from the output, making the clock more susceptible to noise. If you find a value this high is needed to prevent overdriving the crystal, try increasing C2 to compensate or changing the Oscillator Operating mode. Try to get a combination where Rs is around 10 kΩ or less and load capacitance is not too far from the manufacturer's specification.

42.8 RELATED APPLICATION NOTES

This section lists application notes that are related to this section of the manual. These application notes may not be written specifically for the PIC32 device family, but the concepts are pertinent and could be used with modification and possible limitations. The current application notes related to the Oscillators with Enhanced PLL module are:

Title	Application Note #
Crystal Oscillator Basics and Crystal Selection for rfPIC® and PIC® MCU Devices	AN826
Basic PIC® Microcontroller Oscillator Design	AN849
Practical PIC® Microcontroller Oscillator Analysis and Design	AN943
Making Your Oscillator Work	AN949

Note: Please visit the Microchip web site (www.microchip.com) for additional application notes and code examples for the PIC32 family of devices.

42.9 REVISION HISTORY

Revision A (November 2013)

This is the initial released version of this document.

Revision B (June 2015)

Revision includes the following updates:

- The PIC32 Family Oscillator System Block Diagram was updated (see Figure 42-1)
- The SLEWCON and CLKSTAT registers were added (see Register 42-11 and Register 42-12)
- The SLP2SPD bit was added to the OSCCON register (see Register 42-1)
- **42.3.2.1 "System Clock Phase-Locked Loop (PLL)"** was updated
- **42.3.7.5 "Clock Switch Slewing"** was added
- **42.3.10.1 "Reference Clock Operation"** was added
- **42.4 "Interrupts"** was updated
- In addition, minor updates to text and formatting were incorporated throughout the document

Revision C (September 2020)

Revision includes the following:

- Added the following registers in **42.2 "Control Registers"** and Table 42-1:
  - Register 42-5, Register 42-6, Register 42-7, Register 42-4 and Register 42-13
- Updated the following:
  - Register 42-1 with UFRCEN information in bit 2
  - Register 42-11 with SLW_DELAY [3:0] in bit 24-27
  - **42.3 "Operation: Clock Generation and Clock Sources"** with BT Clock and Ethernet-WiFi Clock
  - **42.3.10 "Reference Clock Output"** with BTPLL Clock, EWPLL WiFi Clock, EWPLL Ethernet Clock and USB PLL
- Added non-availability related notes in the following:
  - Table 42-1
  - Register 42-1, Register 42-3 and Register 42-12
  - **42.3.3.3 "PBCLKx On/Off Control"** and **42.3.4.2 "Clock Requirements()"**

NOTES:

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods being used in attempts to breach the code protection features of the Microchip devices. We believe that these methods require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Attempts to breach these code protection features, most likely, cannot be accomplished without violating Microchip's intellectual property rights.
- Microchip is willing to work with any customer who is concerned about the integrity of its code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is "unbreakable." Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

ISBN: 978-1-5224-6563-8

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4485-5910
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-72400

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7288-4388

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820