Oscillator Module

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. This document applies to all dsPIC33/PIC24 family devices. However, some features in this document will not apply to all 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.

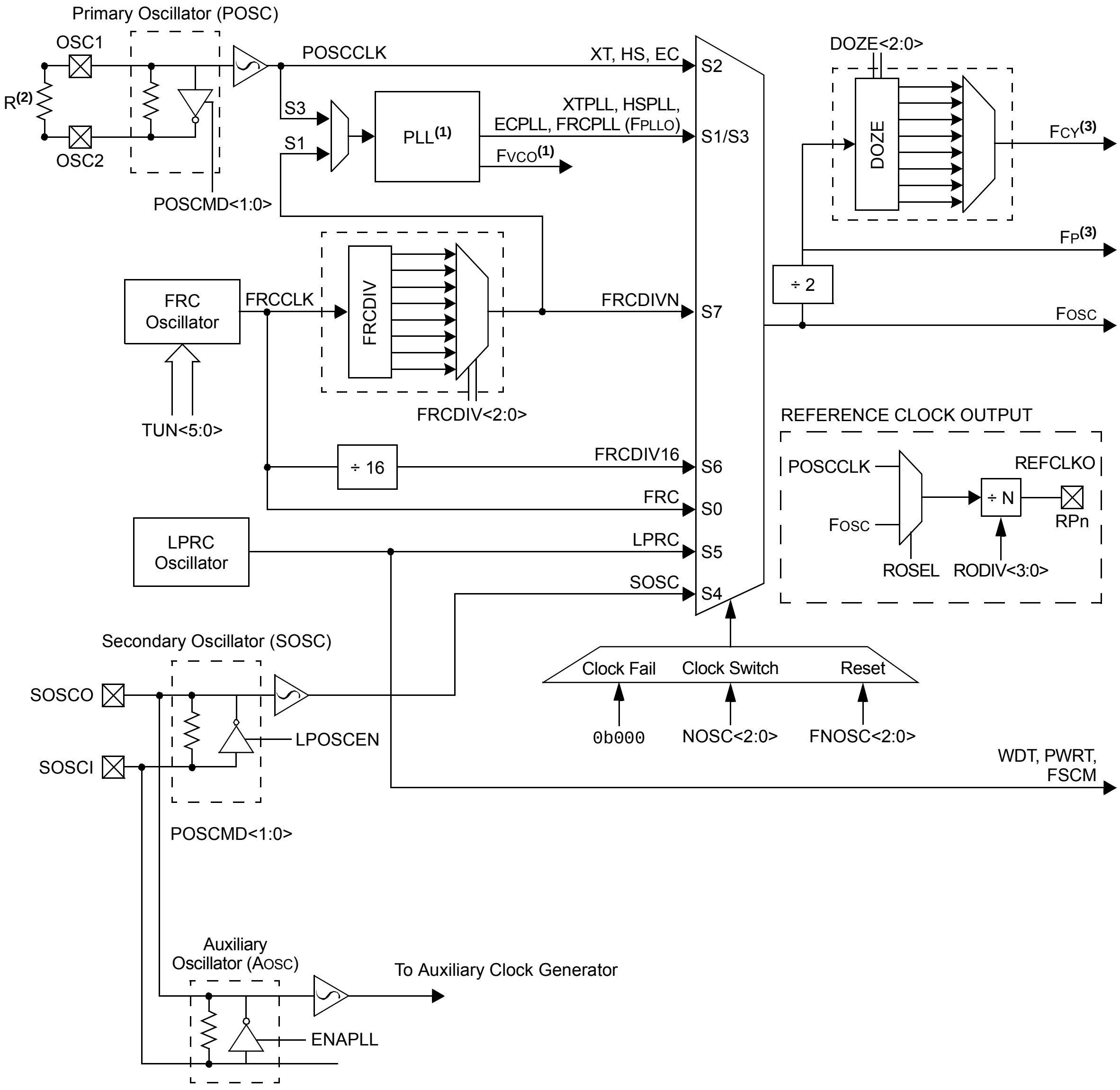
1.0 INTRODUCTION

The dsPIC33/PIC24 family oscillator system includes these characteristics:

- External and internal oscillator sources
- On-chip Phase-Locked Loop (PLL) to boost internal operating frequency on select internal and external oscillator sources
- Auxiliary PLL (APLL) clock generator to boost operating frequency for ADC and PWM
- Auxiliary Oscillator (AOSC) and Auxiliary PLL Clock (ACLK) generator for USB
- Doze mode for system power savings
- Scalable Reference Clock Output (REFCLKO)
- On-the-fly clock switching between various clock sources
- Linear Feedback Shift Register (LFSR) to generate pseudorandom data
- Fail-Safe Clock Monitoring (FSCM) that detects clock failure and permits safe application recovery or shutdown

A block diagram of the dsPIC33/PIC24 family oscillator system is shown in Figure 1-1.

Figure 1-1: Oscillator System Block Diagram

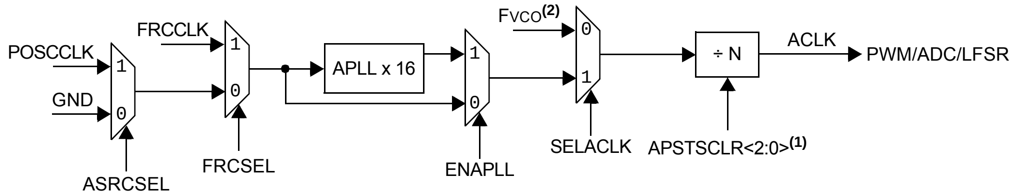


Note 1: See Figure 7-1 for the source of the FVCO signal.

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

3: The term, FP, refers to the clock source for all the peripherals, while FCY (or MIPS) refers to the clock source for the CPU. Throughout this document, FCY and FP are used interchangeably, except in the case of Doze mode. FP and FCY will be different when Doze mode is used in any ratio other than 1:1.

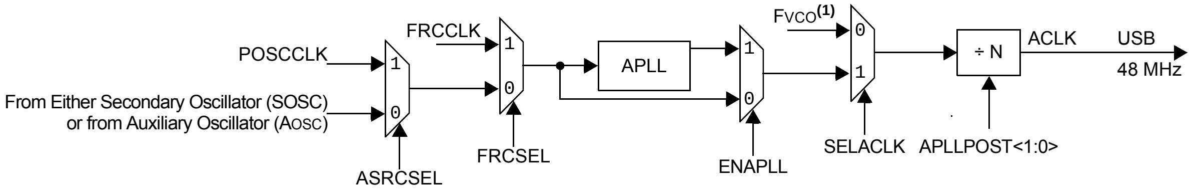
Figure 1-2: Auxiliary Clock Generator for PWM and ADC



Note 1: The Auxiliary Clock postscaler must be configured to divide-by-1 (APSTSCLR<2:0> = `111`) for proper operation of the PWM and ADC modules.

2: See Figure 7-1 for the source of the FVCO signal.

Figure 1-3: Auxiliary Clock Generator for USB



Note 1: See Figure 7-1 for source of FVCO signal.

2.0 CPU CLOCKING

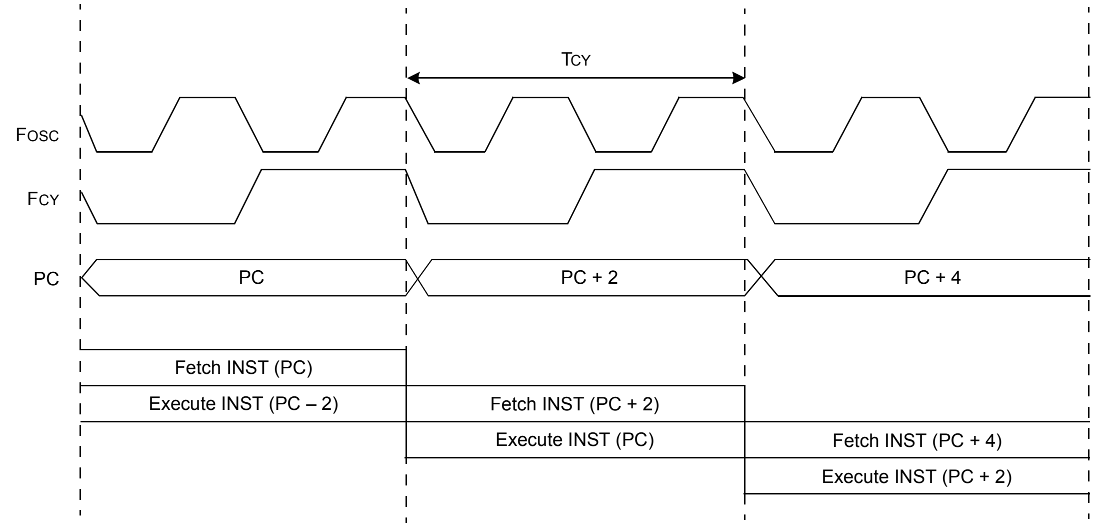
The system clock (FOSC) source can be provided by one of the following options:

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

The system clock source is divided by two to produce the internal instruction cycle clock. In this document, the instruction cycle clock is denoted by FCY. The timing diagram in Figure 2-1 illustrates the relationship between the system clock (FOSC), the instruction cycle clock (FCY) and the Program Counter (PC).

The internal instruction cycle clock (FCY) can be output on the OSC2 I/O pin if the Primary Oscillator mode or the HS mode is not selected as the clock source. For more information, see **Section 5.0 "Primary Oscillator (POSC)"**.

Figure 2-1: Clock and Instruction Cycle Timing



3.0 OSCILLATOR CONFIGURATION REGISTERS

Depending on the device, the Oscillator Configuration registers are implemented in one of two ways:

- Oscillator Configuration registers are located in the program memory space and are not Special Function Registers (SFRs). These registers are mapped into program memory space and are programmed at the time of device programming.
- Can only be programmed indirectly by programming the Flash Configuration Word.

- **FOSCSEL: Oscillator Source Selection Register**

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

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

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

- **FOSC: Oscillator Configuration Register**

  FOSC configures the Primary Oscillator mode, OSC2 pin function, Peripheral Pin Select (PPS), and the Fail-Safe and Clock Switching modes. FOSC contains the following Configuration bits:

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

    If OSCIOFNC is unprogrammed (‘`1`’), the FCY clock is output on the OSC2 pin.

    If OSCIOFNC is programmed (‘`0`’), the OSC2 pin becomes a general purpose I/O pin.

Table 3-1 lists the configuration settings that select the device oscillator source and operating mode at a POR.

Table 3-1: Configuration Bit Values for Clock Selection

Oscillator Source	Oscillator Mode	FNOSC<2:0> Value	POSCMD<1:0> Value	Notes
S0	Fast RC Oscillator (FRC)	000	xx	1
S1	Fast RC Oscillator with PLL (FRCPLL)	001	xx	1
S2	Primary Oscillator (EC)	010	00	1
S2	Primary Oscillator (XT)	010	01	
S2	Primary Oscillator (HS)	010	10	
S3	Primary Oscillator with PLL (ECPLL)	011	00	1
S3	Primary Oscillator with PLL (XTPLL)	011	01	
S3	Primary Oscillator with PLL (HSPLL)	011	10	
S4	Secondary Oscillator (SOSC)	100	xx	1
S5	Low-Power RC Oscillator (LPRC)	101	xx	1
S6	Fast RC Oscillator with ÷ 16 Divider (FRCDIV16)	110	xx	1
S7	Fast RC Oscillator with ÷ N Divider (FRCDIVN)	111	xx	1, 2

Note 1: The OSC2 pin function is determined by the OSCIOFNC Configuration bit.

2: This is the default oscillator mode for an unprogrammed (erased) device.

Register 3-1: FOSCSEL: Oscillator Source Selection Register

U-Z	U-Z	U-Z	U-Z	U-Z	U-Z	U-Z	U-Z
—	—	—	—	—	—	—	—
bit 15							bit 8

R/P	U-Z	U-Z	U-Z	U-Z	R/P	R/P	R/P
IESO	—	—	—	—	FNOSC2	FNOSC1	FNOSC0
bit 7							bit 0

Legend:	Z = Either a ‘1’ or a ‘0’, depending on device		
R = Readable bit	P = Programmable bit	U = Unused bits, Program to Logic ‘1’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Reserved:** Reserved bits must be programmed as ‘`1`’

bit 7 **IESO:** Internal External Start-up Option bit

`1` = Start up device with Internal FRC Oscillator, then automatically switch to the user-selected oscillator source when ready
`0` = Start up device with user-selected oscillator source

bit 6-3 **Reserved:** Reserved bits must be programmed as ‘`1`’

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

`111` = Fast RC Oscillator with Divide-by-N (FRCDIVN)
`110` = Fast RC Oscillator with Divide-by-16 (FRCDIV16)
`101` = Low-Power RC Oscillator (LPRC)
`100` = Secondary Oscillator (SOSC)
`011` = Primary Oscillator with PLL (XTPLL, HSPLL, ECPLL)
`010` = Primary Oscillator (XT, HS, EC)
`001` = Fast RC Oscillator with PLL (FRCPLL)
`000` = Fast RC Oscillator (FRC)

Register 3-2: FOSC: Oscillator Configuration Register

U-Z	U-Z	U-Z	U-Z	U-Z	U-Z	U-Z	R/P
—	—	—	—	—	—	—	PLLKEN
bit 15							bit 8

R/P	R/P	R/P	U-Z	U-Z	R/P	R/P	R/P
FCKSM1	FCKSM0	IOL1WAY	—	—	OSCIOFNC	POSCMD1	POSCMD0
bit 7							bit 0

Legend:	Z = Either a ‘1’ or a ‘0’, depending on device		
R = Readable bit	P = Programmable bit	U = Unused bits, Program to Logic ‘1’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-9 **Reserved:** Reserved bits must be programmed as ‘`1`’

bit 8 **PLLKEN:** PLL Lock Enable bit
`1` = Source for PLL lock signal is the lock detect
`0` = Source for PLL lock signal is the PLL enable signal

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

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

bit 4-3 **Reserved:** Reserved bits must be programmed as ‘`1`’

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

bit 1-0 **POSCMD<1:0>:** Primary Oscillator Mode Selection bits
`11` = Primary Oscillator is disabled
`10` = HS Crystal Oscillator mode (10 MHz to 40 MHz)
`01` = XT Crystal Oscillator mode (3.5 MHz to 10 MHz)
`00` = EC (External Clock) mode (0 MHz to 60 MHz)

4.0 SPECIAL FUNCTION REGISTERS

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

- **OSCCON: Oscillator Control Register(4)**

  This register controls clock switching and provides status information that allows current clock source, PLL lock and clock fail conditions to be monitored.
- **CLKDIV: Clock Divisor Register**

  This register controls the Doze mode and selects the PLL prescaler, PLL postscaler and FRC postscaler.
- **PLLFBD: PLL Feedback Divisor Register**

  This register selects the PLL feedback divisor.
- **OSCTUN: FRC Oscillator Tuning Register**

  This register is used to tune the Internal FRC oscillator frequency in software.
- **REFOCON: Reference Oscillator Control Register**

  This register controls the reference oscillator output.
- **ACLKCON1: Auxiliary Clock Control Register 1(1)**

  This register enables and controls the PLL Auxiliary Oscillator.
- **ACLKCON3: Auxiliary Clock Control Register 3(1)**

  This register controls and provides prescalar and postscalar values for the Auxiliary PLL module.
- **ACLKDIV3: Auxiliary Clock Divisor Control Register 3(1)**

  This register selects the PLL feedback divisor for the Auxiliary PLL module.
- **LFSR: Linear Feedback Shift Register(1)**

  This register provides pseudorandom values.

Register 4-1: OSCCON: Oscillator Control Register(4)

U-0	R-y	R-y	R-y	U-0	R/W-y	R/W-y	R/W-y
—	COSC2(1,2)	COSC1(1,2)	COSC0(1,2)	—	NOSC2	NOSC1	NOSC0
bit 15							bit 8

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

Legend:	U = Unimplemented bit, read as ‘0’		y = Depends on FNOSCx bits (FOSCSEL<2:0>)
R = Readable bit	W = Writable bit	C = Clearable bit	S = Settable bit
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘`0`’

bit 14-12 **COSC<2:0>:** Current Oscillator Selection bits (read-only)(1,2)

`111` = Fast RC Oscillator with Divide-by-N (FRCDIVN)
`110` = Fast RC Oscillator with Divide-by-16 (FRCDIV16)
`101` = Low-Power RC Oscillator (LPRC)
`100` = Secondary Oscillator (SOSC)
`011` = Primary Oscillator with PLL (XTPLL, HSPLL, ECPLL)
`010` = Primary Oscillator (XT, HS, EC)
`001` = Fast RC Oscillator with PLL (FRCPLL)
`000` = Fast RC Oscillator (FRC)

bit 11 **Unimplemented:** Read as ‘`0`’

bit 10-8 **NOSC<2:0>:** New Oscillator Selection bits(3)

`111` = Fast RC Oscillator with Divide by N (FRCDIVN)
`110` = Fast RC Oscillator with Divide by 16 (FRCDIV16)
`101` = Low-Power RC Oscillator (LPRC)
`100` = Secondary Oscillator (SOSC)
`011` = Primary Oscillator with PLL (XTPLL, HSPLL, ECPLL)
`010` = Primary Oscillator (XT, HS, EC)
`001` = Fast RC Oscillator with PLL (FRCPLL)
`000` = Fast RC Oscillator (FRC)

bit 7 **CLKLOCK:** Clock Lock Enable bit

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

bit 6 **IOLOCK:** Peripheral Pin Select (PPS) Lock bit

`1` = Peripheral Pin Select is locked; writes to Peripheral Pin Select registers are not allowed
`0` = Peripheral Pin Select is not locked; writes to Peripheral Pin Select registers are allowed

bit 5 **LOCK:** PLL Lock Status bit (read-only)

`1` = Indicates that PLL is in lock or PLL start-up timer is satisfied
`0` = Indicates that PLL is out of lock, start-up timer is in progress or PLL is disabled

bit 4 **Unimplemented:** Read as ‘`0`’

Note 1: COSC<2:0> are set to FRC value on POR or BOR.

2: COSC<2:0> are loaded with NOSC<2:0> on Reset (not POR or BOR) and at the completion of a successful clock switch.

3: Set to the value specified by the FNOSC<2:0> Configuration bits on any Reset.

4: Writes to this register require an unlock sequence. For more information and examples, see **Section 14.0 “Clock Switching”**.

Register 4-1: OSCCON: Oscillator Control Register[4] (Continued)

bit 3	**CF:** Clock Fail Detect bit (read or cleared by application) `1` = FSCM has detected a clock failure `0` = FSCM has not detected a clock failure
bit 2	**Unimplemented:** Read as ‘`0`’
bit 1	**LPOSCEN:** 32 kHz Secondary (LP) Oscillator Enable bit `1` = Requests oscillator switch to selection specified by the NOSC<2:0> bits `0` = Oscillator switch is complete
bit 0	**OSWEN:** Oscillator Switch Enable bit `1` = Request oscillator switch to selection specified by the NOSC<2:0> bits `0` = Oscillator switch is complete

Note 1: COSC<2:0> are set to FRC value on POR or BOR.

2: COSC<2:0> are loaded with NOSC<2:0> on Reset (not POR or BOR) and at the completion of a successful clock switch.

3: Set to the value specified by the FNOSC<2:0> Configuration bits on any Reset.

4: Writes to this register require an unlock sequence. For more information and examples, see **Section 14.0 “Clock Switching”**.

Register 4-2: CLKDIV: Clock Divisor Register

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
ROI	DOZE2[2,3]	DOZE1[2,3]	DOZE0[2,3]	DOZEN[1]	FRCDIV2	FRCDIV1	FRCDIV0
bit 15							bit 8

R/W-0	R/W-1	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PLLPOST1	PLLPOST0	—	PLLPRE4	PLLPRE3	PLLPRE2	PLLPRE1	PLLPRE0
bit 7							bit 0

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

bit 15 **ROI:** Recover on Interrupt bit

`1` = Interrupts will clear the DOZEN bit and the processor clock, and the peripheral clock ratio is set to 1:1
`0` = Interrupts have no effect on the DOZEN bit

bit 14-12 **DOZE<2:0>:** Processor Clock Reduction Select bits[2,3]

`111` = FCY divided by 128
`110` = FCY divided by 64
`101` = FCY divided by 32
`100` = FCY divided by 16
`011` = FCY divided by 8 (default)
`010` = FCY divided by 4
`001` = FCY divided by 2
`000` = FCY divided by 1

bit 11 **DOZEN:** Doze Mode Enable bit[1]

`1` = DOZE<2:0> field specifies the ratio between the peripheral clocks and the processor clocks
`0` = Processor clock and peripheral clock ratio are forced to 1:1

bit 10-8 **FRCDIV<2:0>:** Internal Fast RC Oscillator Postscaler bits

`111` = FRC divided by 256
`110` = FRC divided by 64
`101` = FRC divided by 32
`100` = FRC divided by 16
`011` = FRC divided by 8
`010` = FRC divided by 4
`001` = FRC divided by 2
`000` = FRC divided by 1 (default)

bit 7-6 **PLLPOST<1:0>:** PLL VCO Output Divider Select bits (also denoted as ‘N2’, PLL postscaler)

`11` = Output divided by 8
`10` = Reserved
`01` = Output divided by 4 (default)
`00` = Output divided by 2

bit 5 **Unimplemented:** Read as ‘`0`’

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

2: The DOZE<2:0> bits can only be written when the DOZEN bit is clear. If DOZEN = `1`, any writes to DOZE<2:0> are ignored.

3: The DOZEN bit cannot be set if DOZE<2:0> = `000`. If DOZE<2:0> = `000`, any attempt by user software to set the DOZEN bit is ignored.

Register 4-2: CLKDIV: Clock Divisor Register (Continued)

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

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

2: The DOZE<2:0> bits can only be written when the DOZEN bit is clear. If DOZEN = `1`, any writes to DOZE<2:0> are ignored.

3: The DOZEN bit cannot be set if DOZE<2:0> = `000`. If DOZE<2:0> = `000`, any attempt by user software to set the DOZEN bit is ignored.

Register 4-3: PLLFBD: PLL Feedback Divisor Register

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	PLLDIV8
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
PLLDIV7	PLLDIV6	PLLDIV5	PLLDIV4	PLLDIV3	PLLDIV2	PLLDIV1	PLLDIV0
bit 7							bit 0

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

bit 15-9	**Unimplemented:** Read as ‘`0`’
bit 8-0	**PLLDIV<8:0>:** PLL Feedback Divisor bits (also denoted as ‘M’, PLL multiplier) `111111111` = 513 • • • `000110000` = 50 (default) • • • `000000010` = 4 `000000001` = 3 `000000000` = 2

Register 4-4: OSCTUN: FRC Oscillator Tuning Register

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

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0
bit 7							bit 0

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

bit 15-6 **Unimplemented:** Read as ‘`0`’

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

`011111` = Center frequency + (See data sheet for tolerance)
`011110` = Center frequency + (See data sheet for tolerance)
•
•
•
`000010` = Center frequency + (See data sheet for tolerance)
`000001` = Center frequency + (See data sheet for tolerance)
`000000` = Center frequency (7.373 MHz nominal)
`111111` = Center frequency - (See data sheet for tolerance)
•
•
•
`100001`= Center frequency – (See data sheet for tolerance)
`100000`= Center frequency – (See data sheet for tolerance)

Register 4-5: REFOCON: Reference Oscillator Control Register

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ROON	—	ROSSLP	ROSEL	RODIV3(1)	RODIV2(1)	RODIV1(1)	RODIV0(1)
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

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

bit 15 **ROON:** Reference Oscillator Output Enable bit

`1` = Reference oscillator output is enabled on REFOCLK pin(2)
`0` = Reference oscillator output is disabled

bit 14 **Unimplemented:** Read as ‘`0`’

bit 13 **ROSSLP:** Reference Oscillator Run in Sleep bit

`1` = Reference oscillator output continues to run in Sleep
`0` = Reference oscillator output is disabled in Sleep

bit 12 **ROSEL:** Reference Oscillator Source Select bit

`1` = Oscillator crystal is used as the reference clock
`0` = System clock is used as the reference clock

bit 11-8 **RODIV<3:0>:** Reference Oscillator Divider bits(1)

`1111` = Reference clock divided by 32,768
`1110` = Reference clock divided by 16,384
`1101` = Reference clock divided by 8,192
`1100` = Reference clock divided by 4,096
`1011` = Reference clock divided by 2,048
`1010` = Reference clock divided by 1,024
`1001` = Reference clock divided by 512
`1000` = Reference clock divided by 256
`0111` = Reference clock divided by 128
`0110` = Reference clock divided by 64
`0101` = Reference clock divided by 32
`0100` = Reference clock divided by 16
`0011` = Reference clock divided by 8
`0010` = Reference clock divided by 4
`0001` = Reference clock divided by 2
`0000` = Reference clock

bit 7-0 **Unimplemented:** Read as ‘`0`’

Note 1: The reference oscillator output must be disabled (ROON = `0`) before writing to these bits.
2: This pin is remappable. See the **“I/O Ports”** chapter in the specific device data sheet for information.

Register 4-6: ACLKCON1: Auxiliary Clock Control Register 1[(1)]

R/W-0	R/W-0	R/W-Z	U-0	U-0	R/W-1	R/W-1	R/W-1
ENAPLL	APLLCK	SELACLK	—	—	APSTSCLR2	APSTSCLR1	APSTSCLR0
bit 15							bit 8

R/W-0	R/W-1	U-0	U-0	U-0	U-0	U-0	U-0
ASRCSEL	FRCSEL	—	—	—	—	—	—
bit 7							bit 0

Legend:	Z = Either a ‘1’ or a ‘0’, depending on device		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **ENAPLL:** Auxiliary PLL Enable bit

`1` = Auxiliary PLL is enabled
`0` = Auxiliary PLL is disabled

bit 14 **APLLCK:** Auxiliary PLL Phase Locked State Status bit (read-only)

`1` = Auxiliary PLL is in lock
`0` = Auxiliary PLL is not in lock

bit 13 **SELACLK:** Select Auxiliary Clock Source for Auxiliary Clock Divider bit

`1` = Auxiliary PLL, FRC or POSC provides the source clock for the Auxiliary Clock divider
`0` = PLL output (FVCO) provides the source clock for the Auxiliary Clock divider

bit 12-11 **Unimplemented:** Read as ‘`0`’

bit 10-8 **APSTSCLR<2:0>:** Auxiliary Clock Output Divider bits

`111` = Divide-by-1 (default)
`110` = Divide-by-2
`101` = Divide-by-4
`100` = Divide-by-8
`011` = Divide-by-16
`010` = Divide-by-32
`001` = Divide-by-64
`000` = Divide-by-256

bit 7 **ASRCSEL:** Select Reference Clock Source for Auxiliary Clock bit

`1` = Primary Oscillator is the clock source
`0` = Reserved

bit 6 **FRCSEL:** Select Reference Clock Source for Auxiliary Clock bit

`1` = Selects FRC clock for clock source
`0` = POSC is the clock source for APLL (determined by the ASRCSEL bit)

bit 5-0 **Unimplemented:** Read as ‘`0`’

Note 1: This register is not available on all devices. Refer to the **“Oscillator Configuration”** chapter in the specific device data sheet for availability.

Register 4-7: ACLKCON3: Auxiliary Clock Control Register 3[1]

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
ENAPLL	—	SELACLK	AOSCMD1	AOSCMD0	ASRCSEL	FRCSEL	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
APLLPOST2	APLLPOST1	APLLPOST0	—	—	APLLPRE2	APLLPRE1	APLLPRE0
bit 7							bit 0

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

bit 15 **ENAPLL:** Enable Auxiliary PLL (APLL) and Select APLL as USB Clock Source bit

`1` = APLL is enabled, the USB clock source is the APLL output
`0` = APLL is disabled, the USB clock source is the input clock to the APLL

bit 14 **Unimplemented:** Read as ‘`0`’

bit 13 **SELACLK:** Select Auxiliary Clock Source for Auxiliary Clock Divider bit

`1` = Auxiliary PLL or oscillators provide the source clock for the Auxiliary Clock divider
`0` = Primary PLL provides the source clock for the Auxiliary Clock divider

bit 12-11 **AOSCMD<1:0>:** Auxiliary Oscillator Mode bits

`11` = EC (External Clock) Oscillator mode select
`10` = XT (Crystal) Oscillator mode select
`01` = HS (High-Speed) Oscillator mode select
`00` = (AOSC) Auxiliary Oscillator is disabled (default)

bit 10 **ASRCSEL:** Select Reference Clock Source for APLL bit

`1` = Primary Oscillator is the clock source for APLL
`0` = Auxiliary Oscillator is the clock source for APLL

bit 9 **FRCSEL:** Select FRC as Reference Clock Source for APLL bit

`1` = FRC is the clock source for APLL
`0` = Auxiliary Oscillator or Primary Oscillator is the clock source for APLL (determined by ASRCSEL bit)

bit 8 **Unimplemented:** Read as ‘`0`’

bit 7-5 **APLLPOST<2:0>:** Select PLL VCO Output Divider bits

`111` = Divided by 2
`110` = Divided by 2
`101` = Divided by 4
`100` = Divided by 8
`011` = Divided by 16
`010` = Divided by 32
`001` = Divided by 64
`000` = Divided by 256 (default)

bit 4-3 **Unimplemented:** Read as ‘`0`’

bit 2-0 **APLLPRE<2:0>:** PLL Phase Detector Input Divider bits

`111` = Divided by 12
`110` = Divided by 10
`101` = Divided by 6
`100` = Divided by 5
`011` = Divided by 4
`010` = Divided by 3
`001` = Divided by 2
`000` = Divided by 1 (default)

Note 1: This register is not available on all devices. Refer to the **“Oscillator Configuration”** chapter in the specific device data sheet for availability.

Register 4-8: ACLKDIV3: Auxiliary Clock Divisor Control Register 3[1]

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

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	APLLDIV2	APLLDIV1	APLLDIV0
bit 7							bit 0

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

bit 15-3 **Unimplemented:** Read as ‘`0`’

bit 2-0 **APLLDIV<2:0>:** PLL Feedback Divisor bits (PLL multiplier ratio)

`111` = 24
`110` = 21
`101` = 20
`100` = 19
`011` = 18
`010` = 17
`001` = 16
`000` = 15 (default)

Note 1: This register is not available on all devices. Refer to the **“Oscillator Configuration”** chapter in the specific device data sheet for availability.

Register 4-9: LFSR: Linear Feedback Shift Register(1)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	LFSR14	LFSR13	LFSR12	LFSR11	LFSR10	LFSR9	LFSR8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LFSR7	LFSR6	LFSR5	LFSR4	LFSR3	LFSR2	LFSR1	LFSR0
bit 7							bit 0

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

bit 15	**Unimplemented:** Read as ‘0’
bit 14-0	**LFSR<14:0>:** Pseudorandom Data bits

Note 1: This register is not available on all devices.

5.0 PRIMARY OSCILLATOR (POSC)

The Primary Oscillator (POSC) is available on the OSC1 and OSC2 pins of the dsPIC33/PIC24 family devices. This connection enables an external crystal (or ceramic resonator) to provide the clock to the device. Optionally, and depending on the device, it can be used with the internal PLL to boost the system frequency (FOSC) up to 140 MHz for 70 MIPS execution. The Primary Oscillator provides three modes of operation:

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

  The XT mode is a Medium Gain, Medium Frequency mode used to work with crystal frequencies of 3.5 MHz to 10 MHz.

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

  The HS mode is a High Gain, High-Frequency mode used to work with crystal frequencies of 10 MHz to 40 MHz.

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

  If the on-chip oscillator is not used, the EC mode allows the internal oscillator to be bypassed. The device clocks are generated from an external source (0 MHz to up to 60 MHz) and input on the OSC1 pin.

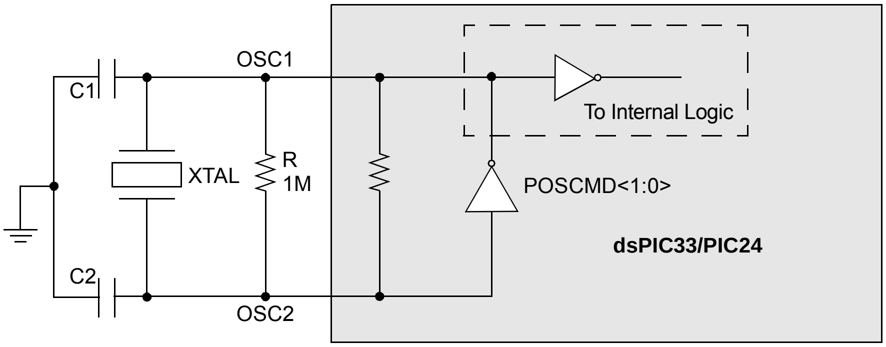
The FNOSC<2:0> Configuration bits in the Oscillator Source Selection register (FOSCSEL<2:0>) specify the Primary Oscillator clock source at Power-on Reset. The POSCMD<1:0> Configuration bits in the Oscillator Configuration register (FOSC<1:0>) specify the Primary Oscillator mode. Table 5-1 provides the options selected by specific bit configurations, which are programmed at the time of device programming.

Table 5-1: Primary Oscillator Clock Source Options

FNOSC<2:0> Value	POSCMD<1:0> Value	Primary Oscillator Source and Mode
`010`	`00`	Primary Oscillator: External Clock Mode (EC)
`010`	`01`	Primary Oscillator: Medium Frequency Mode (XT)
`010`	`10`	Primary Oscillator: High-Frequency Mode (HS)
`011`	`00`	Primary Oscillator with PLL: External Clock Mode (ECPLL)
`011`	`01`	Primary Oscillator with PLL: Medium Frequency Mode (XTPLL)
`011`	`10`	Primary Oscillator with PLL: High-Frequency Mode (HSPLL)

Figure 5-1 is a recommended crystal oscillator circuit diagram for the dsPIC33/PIC24 family devices. Capacitors, C1 and C2, form the Load Capacitance for the crystal. The optimum Load Capacitance (CL) for a given crystal is specified by the crystal manufacturer. Load Capacitance can be calculated as shown in Equation 5-1.

Figure 5-1: Crystal or Ceramic Resonator Operation (XT or HS Oscillator Mode)



Note: For more information, refer to the specific device data sheet.

Equation 5-1: Crystal Load Capacitance

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

Note: Where C_S is the stray capacitance.

Assuming C1 = C2, Equation 5-2 gives the capacitor value (C1, C2) for a given load and stray capacitance.

Equation 5-2: External Capacitor for Crystal

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

For more information on crystal oscillators and their operation, refer to **Section 19.0 "Related Application Notes"**.

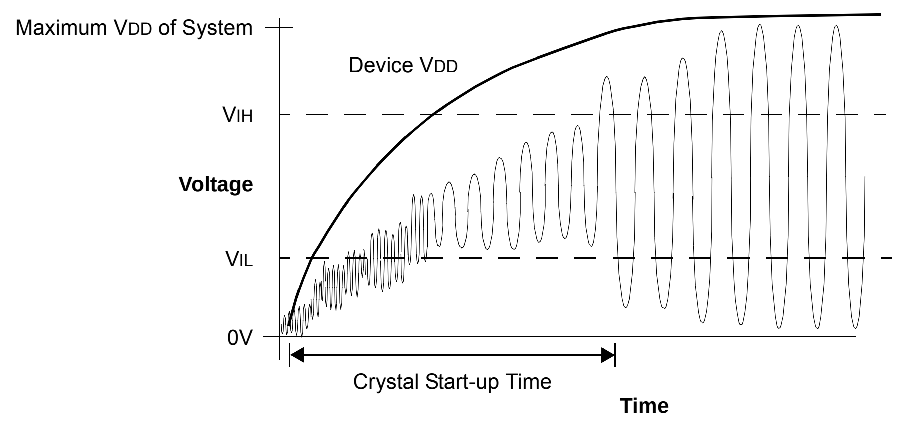
5.1 Oscillator Start-up Time

As the device voltage increases from VSS, the oscillator will start its oscillations. The time required for the oscillator to start oscillating depends on these factors:

- Crystal and resonator frequency
- Capacitor values used (C1 and C2 in Figure 5-1)
- Device VDD rise time
- System temperature
- Series resistor value and type if used
- Oscillator mode selection of device (selects the gain of the internal oscillator inverter)
- Crystal quality
- Oscillator circuit layout
- System noise

Figure 5-2 illustrates a plot of a typical oscillator and resonator start-up.

Figure 5-2: Example Oscillator and Resonator Start-up Characteristics



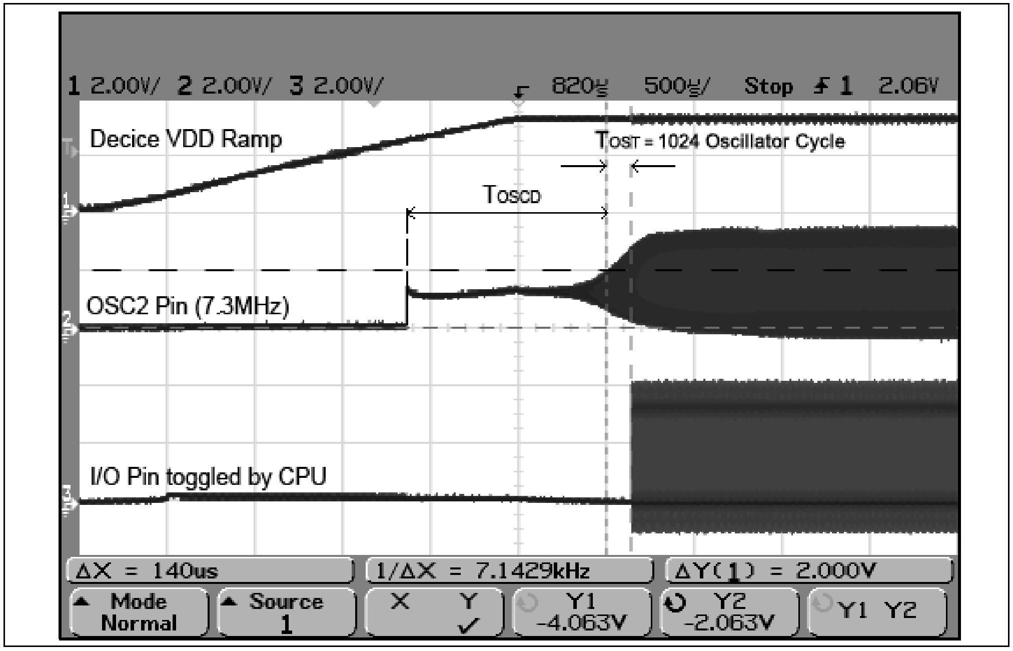
To ensure that a crystal oscillator (or ceramic resonator) has started and stabilized, an Oscillator Start-up Timer (OST) is provided with the Primary Oscillator (POSC) and the Secondary Oscillator (SOSC). The OST is a simple, 10-bit counter that counts 1024 cycles before releasing the oscillator clock to the rest of the system. This time-out period is denoted as 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 restarts (that is, on POR, BOR and wake-up from Sleep mode) when XT or HS mode is selected in the Configuration Words. The TOST timer does not exist when EC mode is selected.

After the Primary Oscillator is enabled, it takes a finite amount of time to start oscillating. This delay is denoted as TOSCD. After TOSCD, the OST timer takes 1024 clock cycles (TOST) to release the clock. The total delay for the clock to be ready is TOSCD + TOST. If the PLL is used, an additional delay is required for the PLL to lock. For more information, see **Section 7.0 "Phase-Locked Loop (PLL)"**.

Primary Oscillator start-up behavior is illustrated in Figure 5-3, where the CPU begins toggling an I/O pin when it starts execution after the TOSCD + TOST interval.

Figure 5-3: Oscillator Start-up Characteristics



5.2 Primary Oscillator Pin Functionality

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

POSCMD<1:0>: Primary Oscillator Mode Selection bits:

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

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

- `1` = OSC2 is the clock output and the instruction cycle (FCY) clock is output on the OCS2 pin (see Figure 5-4)
- `0` = OSC2 is a general purpose digital I/O pin (see Figure 5-5)

The oscillator pin functions are provided in Table 5-2.

Table 5-2: Clock Pin Function Selection

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

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

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

Figure 5-4: OSC2 Pin for Clock Output (in EC Mode)

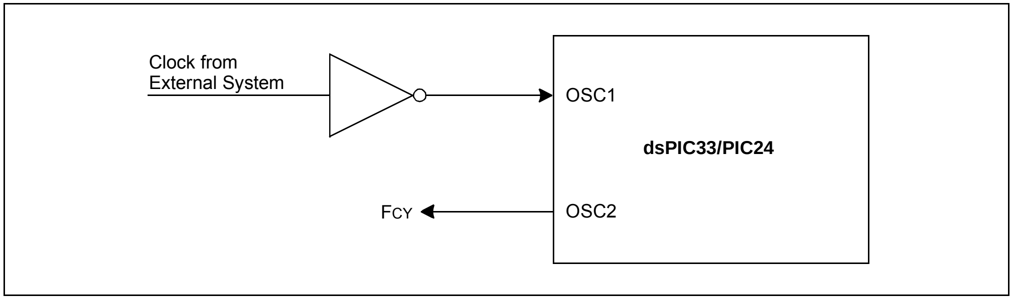
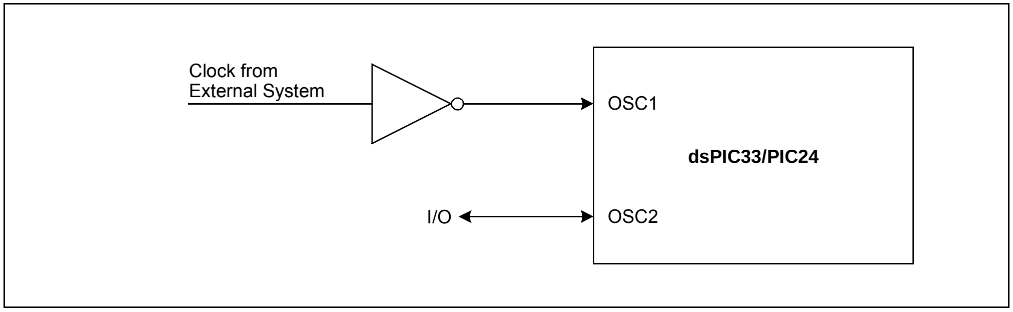


Figure 5-5: OSC2 Pin for Digital I/O (in EC Mode)



6.0 INTERNAL FAST RC (FRC) OSCILLATOR

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

The application software can tune the frequency of the oscillator using the FRC Oscillator Tuning bits (TUN<5:0>) in the FRC Oscillator Tuning register (OSCTUN<5:0>).

Note: Refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for the accuracy of the FRC clock frequency over temperature and voltage variations.

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

The Initial Oscillator Source Selection Configuration bits (FNOSC<2:0>) in the Oscillator Source Selection register (FOSCSEL<2:0>) select the FRC clock source. The FRC clock source options at the time of a Power-on Reset are provided in Table 6-1. The Configuration bits are programmed at the time of device programming.

Table 6-1: FRC Clock Source Options

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

6.1 FRC Postscaler Mode (FRCDIVN)

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

Table 6-2: Internal Fast RC Oscillator Postscaler Settings

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

Optionally, and depending on the device, the FRC postscaler output can be used with the internal PLL to boost the system frequency (FOSC) up to 140 MHz for 70 MIPS instruction cycle execution speed.

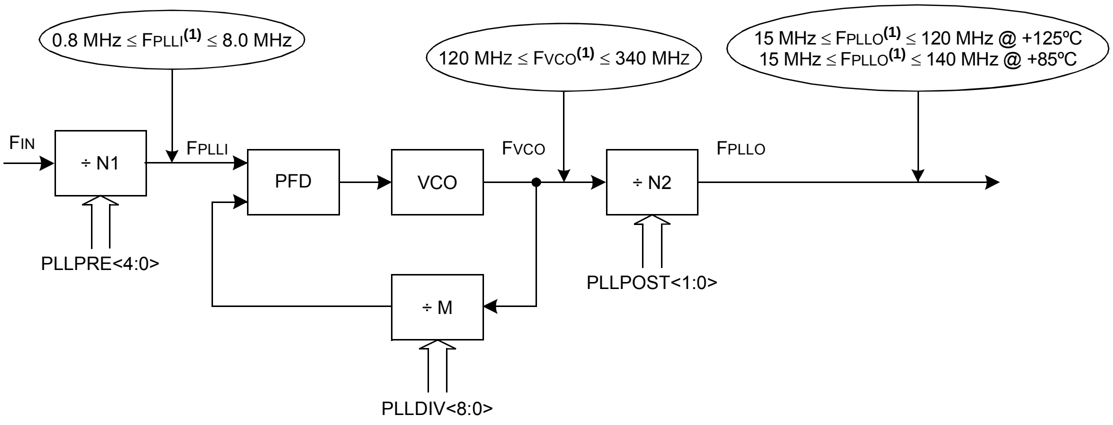
Note: The FRC divider should not be changed dynamically when operating in internal FRC with PLL.

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

7.0 PHASE-LOCKED LOOP (PLL)

The Primary Oscillator and Internal FRC Oscillator sources can optionally use an on-chip PLL to obtain higher operating speeds. Figure 7-1 illustrates a block diagram of the PLL module.

Figure 7-1: dsPIC33/PIC24 Family PLL Block Diagram


0.8 MHz ≤ FPLLI(1) ≤ 8.0 MHz
120 MHz ≤ FVCO(1) ≤ 340 MHz
15 MHz ≤ FPLLO(1) ≤ 120 MHz @ +125ºC
15 MHz ≤ FPLLO(1) ≤ 140 MHz @ +85ºC
FIN
÷ N1
FPLLI
PFD
VCO
FVCO
÷ N2
FPLLO
PLLPRE<4:0>
PLLPOST<1:0>
÷ M
PLLDIV<8:0>

Note 1: This frequency range must be met at all times.

For PLL operation, the Phase Frequency Detector (PFD) input frequency and Voltage Controlled Oscillator (VCO) output frequency must meet the following requirements at all times, no exceptions:

- The PFD Input Frequency (FPLLI) must be in the range of 0.8 MHz to 8.0 MHz
- The VCO Output Frequency (FVCO) must be in the range of 120 MHz to 340 MHz

The PLL Phase Detector Input Divider Select bits (PLLPRE<4:0>) in the Clock Divisor register (CLKDIV<4:0>) specify the input divider ratio (N1), which is used to scale down the input clock (FIN) to meet the PFD input frequency range of 0.8 MHz to 8.0 MHz.

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

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

Equation 7-1 provides the relation between Input Frequency (FIN) and VCO Frequency (FVCO).

Equation 7-1: FVCO Calculation

$$F_{VCO} = F_{IN} \times \left(\frac{M}{N1}\right) = F_{IN} \times \left(\frac{(PLLDIV + 2)}{(PLLPRE + 2)}\right)$$

Equation 7-2 provides the relation between Input Frequency (FIN) and Output Frequency (FPLLO).

Equation 7-2: FPLLO Calculation

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

Where,

$N1 = PLLPRE + 2$
$N2 = 2 \times (PLLPOST + 1)$
$M = PLLDIV + 2$

7.1 Input Clock Limitation at Start-up for PLL Mode

Table 7-1 provides the default values of the PLL Prescaler, PLL Postscaler and PLL Feedback Divisor Configuration bits at Power-on Reset.

Table 7-1: PLL Mode Defaults

Register	Bit Field	Value at POR Reset	PLL Divider Ratio
CLKDIV<4:0>	PLLPRE<4:0>	`00000`	N1 = 2
CLKDIV<7:6>	PLLPOST<1:0>	`01`	N2 = 4
PLLFBD<8:0>	PLLDIV<8:0>	`000110000`	M = 50

Given these Reset values, the following equations provide the relationship between Input Frequency (FIN) and PFD Input Frequency (FPLLI), VCO Frequency (FVCO) and System Clock Frequency (FOSC) at Power-on Reset.

Equation 7-3: FPLLI at Power-on Reset

$$F_{PLLI} = F_{IN}\left(\frac{1}{N1}\right) = 0.5(F_{IN})$$

Equation 7-4: FVCO at Power-on Reset

$$F_{VCO} = F_{IN}\left(\frac{M}{N1}\right) = F_{IN}\left(\frac{50}{2}\right) = 25(F_{IN})$$

Equation 7-5: FPLLO at Power-on Reset

$$F_{PLLO} = F_{IN}\left(\frac{M}{N1 \times N2}\right) = F_{IN}\left(\frac{50}{2} \times 4\right) = 6.25(F_{IN})$$

To use the PLL, and to ensure that the PFD Input Frequency (FPLLI) and the VCO frequency are in the specified frequency range to meet the PLL requirements, follow this process:

1. Power up the device with the Internal FRC or the Primary Oscillator without PLL.
2. Change the PLLDIVx, PLLPREx and PLLPOSTx bit values, based on the input frequency, to meet these PLL requirements:
   - The PFD Input Frequency (FPLLI) must be in the range of 0.8 MHz to 8.0 MHz
   - The VCO Output Frequency (FVCO) must be in the range of 120 MHz to 340 MHz
3. Switch the clock to the PLL mode in software.

Note: Due to the default PLL register setting on Reset, it would violate the OSC specification to power up with the Primary Oscillator with PLL enabled for input clock frequencies greater than 13.6 MHz. In that case, the user would need to power up in a non-PLL mode, configure the PLL registers and then perform a clock switch to a PLL mode.

It is not permitted to directly clock switch from one PLL clock source to a different PLL clock source. The user would need to transition between PLL clock sources with a clock switch to a non-PLL clock source.

7.2 PLL Lock Status

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

T_{LOCK} is applied when the PLL is selected as the clock source at Power-on Reset, or during a clock switching operation. The value of T_{LOCK} is relative to the time at which the clock is available to the PLL input. For example, with the POSC, T_{LOCK} starts after the OST delay. For more information about oscillator start-up delay, see **Section 5.1 "Oscillator Start-up Time"**. Also, refer to the **"Oscillator Configuration"** chapter in the specific device data sheet for more information about typical T_{LOCK} values.

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

Note: The PLL Prescaler (PLLPREx) and PLL Feedback Divisor (PLLDIVx) bits should not be changed when operating in PLL mode. You must clock switch to a non-PLL mode (e.g., Internal FRC) to make the necessary changes and then clock switch back to the PLL mode.

7.2.1 SETUP FOR USING PLL WITH THE PRIMARY OSCILLATOR (POSC)

The following process is used to set up the PLL to operate the device at 60 MIPS with a 10 MHz external crystal:

1. To execute instructions at 60 MHz, ensure that the required system clock frequency is:

   $F_{OSC} = 2 \times F_{CY} = 120$ MHz
2. To set up the PLL and meet the requirements of the PLL, follow these steps:
   a) Select the PLL postscaler to meet the VCO output frequency requirement ($120 \text{ MHz} < F_{VCO} < 340 \text{ MHz}$).
      - Select a PLL postscaler ratio of N2 = 2
      - Ensure that $F_{VCO} = (F_{PLLO} \times N2) = 240$ MHz

   b) Select the PLL prescaler to meet the PFD input frequency requirement ($0.8 \text{ MHz} < F_{PLLI} < 8.0 \text{ MHz}$).
      - Select a PLL prescaler ratio of N1 = 2
      - Ensure that $F_{PLLI} = (F_{IN} \div N1) = 5$ MHz

   c) Select the PLL feedback divisor to generate the required VCO output frequency based on the PFD input frequency.
      - $F_{VCO} = F_{PLLI} \times M$
      - $M = F_{VCO} \div F_{PLLI} = 48$

   d) Configure the FNOSC<2:0> bits (FOSCSEL<2:0>) to select a clock source without the PLL (for example, Internal FRC) at Power-on Reset.

   e) In the main program, change the PLL prescaler, PLL postscaler and PLL feedback divisor values to those just decided in the previous steps, and then perform a clock switch to the PLL mode.

Example 7-1 illustrates code for using the PLL with the Primary Oscillator. (Also, see **Section 14.0 “Clock Switching”** for example code for clock switching.)

Example 7-1: Code Example for Using PLL with the Primary Oscillator (POSC)

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

// Enable Clock Switching and Configure POSC in XT mode

_FOSC(FCKSM_CSECMD & OSCIOFNC_OFF & POSCMD_XT);
int main()
{
// Configure PLL prescaler, PLL postscaler, PLL divisor
PLLFBD=46;                        // M=48
CLKDIVbits.PLLPOST=0;             // N2=2
CLKDIVbits.PLLPRE=0;              // N1=2

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

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

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

7.2.2 SETUP FOR USING PLL WITH 7.37 MHZ INTERNAL FRC

The following process is used to set up the PLL to operate the device at 60 MIPS with a 7.37 MHz Internal FRC.

1. To execute instructions at 60 MHz, ensure that the system clock frequency is:

   FOSC = 2 x FCY = 120 MHz
2. To set up the PLL and meet the requirements of the PLL, follow these steps:
   a) Select the PLL postscaler to meet the VCO output frequency requirement (120 MHz < FVCO < 340 MHz).
      - Select a PLL postscaler ratio of N2 = 2
      - Ensure that FVCO = (FPLLO x N2) = 240 MHz

   b) Select the PLL prescaler to meet the PFD input frequency requirement (0.8 MHz < FPLLI < 8.0 MHz).
      - Select a PLL prescaler ratio of N1 = 2
      - Ensure that FPLLI = (FIN ÷ N1) = 3.68 MHz

   c) Select the PLL feedback divisor to generate the required VCO output frequency based on the PFD input frequency.
      - FVCO = FPLLI x M
      - M = FVCO ÷ FPLLI = 65

   d) Configure the FNOSC<2:0> bits (FOSCSEL<2:0>) to select a clock source without PLL (for example, Internal FRC) at Power-on Reset.

   e) In the main program, change the PLL prescaler, PLL postscaler and PLL feedback divisor to meet the user and PLL requirements, and then perform a clock switch to the PLL mode.

Example 7-2 illustrates code for using PLL with a 7.37 MHz Internal FRC. (See also **Section 14.0 "Clock Switching"** for example code for clock switching.)

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

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

// Enable Clock Switching and Configure Primary Oscillator in XT mode
_FOSC(FCKSM_CSECMD & OSCIOFNC_OFF & POSCMD_NONE);

int main()
{
// Configure PLL prescaler, PLL postscaler, PLL divisor
PLLFBD=63;                        // M=65
CLKDIVbits.PLLPOST=0;             // N2=2
CLKDIVbits.PLLPRE=1;              // N1=3

// Initiate Clock Switch to FRC oscillator with PLL (NOSC=0b001)
__builtin_write_OSCCONH(0x01);
__builtin_write_OSCCONL(OSCCON | 0x01);
// Wait for Clock switch to occur
while (OSCCONbits.COSC!= 0b001);

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

8.0 SECONDARY OSCILLATOR (SOSC)

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

Note 1: The SOSC is sometimes referred to as the Low-Power Secondary Oscillator due to its low-power capabilities. However, this oscillator should not be confused with the Low-Power RC (LPRC) Oscillator.

2: This oscillator is not available on all devices. Refer to the specific device data sheet for more information.

8.1 SOSC for System Clock

The SOSC is enabled as the system clock when:

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

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

8.2 SOSC Start-up Delay

When the SOSC is enabled, it takes a finite amount of time to start oscillating. For more information, refer to **Section 5.1 “Oscillator Start-up Time”**.

8.3 Continuous SOSC Operation

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

There are two reasons to leave the SOSC running:

- Keeping the SOSC always ON allows a fast switch to the 32 kHz system clock for lower power operation, since returning to the faster main oscillator still requires an oscillator start-up time if it is a crystal type source. For more information, refer to **Section 5.1 “Oscillator Start-up Time”**.
- The oscillator should remain on continuously when Timer1 is used as an RTC.

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

9.0 LOW-POWER RC (LPRC) OSCILLATOR

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

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

9.1 LPRC Oscillator for System Clock

The LPRC oscillator is selected as the system clock when:

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

9.2 Enabling the LPRC Oscillator

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

The LPRC oscillator remains enabled under these conditions:

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

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

Note: The LPRC is enabled and running automatically if either WDT or clock fail detect is enabled. The LPRC runs in Sleep mode only if the Watchdog Timer is enabled. Under all other conditions, LPRC is disabled in Sleep mode.

9.3 LPRC Oscillator Start-up Delay

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

10.0 AUXILIARY OSCILLATOR

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

The Auxiliary Oscillator (AOSC) is used by the Universal Serial Bus (USB) module, which needs to operate at a frequency unrelated to the system clock. The Auxiliary Oscillator can use one of the following as its clock source:

- Crystal (XT): Crystal and ceramic resonators in the range of 3.5 MHz to 10 MHz.
- High-Speed Crystal (HS): Crystals in the range of 10 MHz to 40 MHz. The external crystal is connected to the SOSCI and SOSCO pins.
- External Clock (EC): External clock signal up to 60 MHz. The external clock signal is directly applied to the SOSCI pin.

10.1 Enabling the Auxiliary Oscillator

To enable the Auxiliary Oscillator mode, the Enable Auxiliary PLL bit (ENAPLL) must be set in the Auxiliary Clock Control register (ACLKCONx<15>). The Auxiliary Oscillator Mode bits (AOSCMD<1:0>) allow four oscillator mode settings, as listed in Table 10-1.

Table 10-1: Auxiliary Oscillator and External Oscillator Mode Settings

AOSCMD<1:0> Bit Value	Oscillator Mode Setting
`11`	EC (External Clock) Mode Select
`10`	XT (Crystal) Oscillator Mode Select
`01`	HS (High-Speed) Oscillator Mode Select
`00`	Auxiliary Oscillator Disabled (default setting)

Note: By default, the USB module is clocked by the Primary Oscillator with PLL.

10.2 Auxiliary Clock Source

The desired reference clock source for the Auxiliary PLL can be selected by setting the appropriate clock source select bits in the Auxiliary Clock Control Register 1 (ACLKCON1).

Set the Auxiliary Reference Clock Select bit (ASRCSEL) to use the Primary Oscillator as the clock source or clear this bit to use the Auxiliary Oscillator as the clock source.

Set the FRC Select bit (FRCSEL) to use the FRC as the clock source, or clear this bit to use the Auxiliary or Primary Oscillator selected by the ASRCSEL bit as the clock source.

Set the Select Clock Source to Auxiliary Clock Divider bit (SELACLK) in the Auxiliary Clock Control Register 1 (ACLKCON1) to select the Auxiliary PLL or oscillators to provide the clock source for the Auxiliary Clock divider.

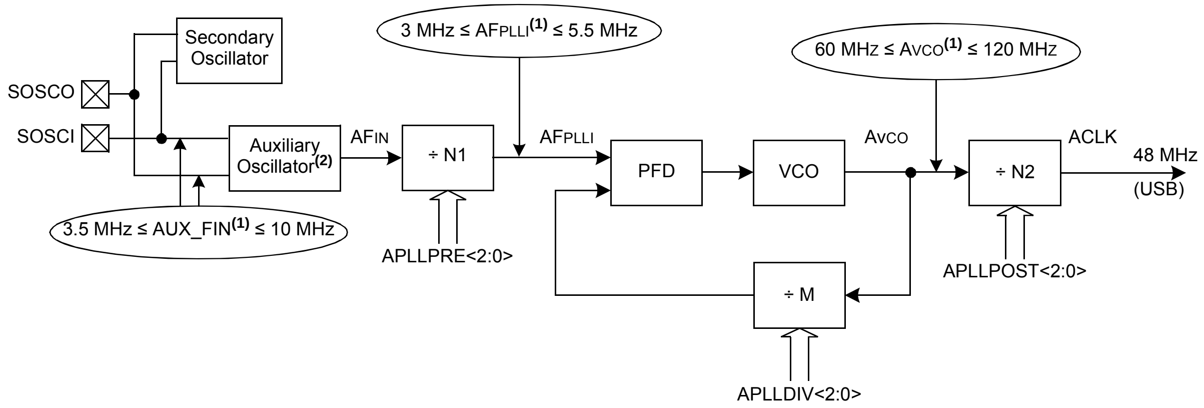
Clearing the SELACLK bit will cause the primary PLL output to act as the clock source to the Auxiliary Clock divider.

11.0 AUXILIARY PHASE-LOCKED LOOP (APLL)

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

The Auxiliary Oscillator uses an on-chip PLL to obtain different Auxiliary Clock speeds. Figure 11-1 shows a block diagram of the APLL module.

Figure 11-1: dsPIC33/PIC24 Family APLL Block Diagram



Note 1: This frequency range must be met at all times.

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

For operation of the APLL, the Auxiliary Phase Frequency Detector (APFD) input frequency and the Auxiliary Voltage Controlled Oscillator (AVCO) output frequency must meet the following requirements:

- The APFD Input Frequency (AFPLLI) must be in the range of 3 MHz to 5.5 MHz
- The AVCO output frequency must be in the range of 60 MHz to 120 MHz

The APLL Phase Detector Input Divider bits (APLLPRE<2:0>) in the Auxiliary Clock Control Register 3 (ACLKCON3<2:0>) specify the input divider ratio (N1), which is used to scale down the Auxiliary PLL Input (AFIN) clock to meet the APFD input frequency range of 3 MHz to 5.5 MHz.

The Auxiliary PLL Feedback Divisor bits (APLLDIV<2:0>) in the Auxiliary Clock Divisor Control Register 3 (ACLKDIV3<2:0>) specify the divider ratio (M), which scales down the AVCO frequency for feedback to the APFD. The AVCO frequency is M times the APFD Input Frequency (AFPLLI).

The APLL VCO Output Divider Select bits (APLLPOST<2:0>) in the Auxiliary Clock Control Register 3 (ACLKCON3<7:5>) specify the divider ratio (N2).

The correct combination of the APLL Phase Detector Input Divider bits (APLLPRE<2:0>), the Auxiliary PLL Feedback Divisor bits (APLLDIV<2:0>) and the APLL VCO Output Divider bits (APLLPOST<2:0>) will provide the 48 MHz Auxiliary Clock (ACLK) frequency needed by the USB module.

Equation 11-1 shows the relationship between the Auxiliary PLL Input (AFIN) clock frequency and the Auxiliary Clock (ACLK) frequency.

Equation 11-1: ACLK Calculation

$$ACLK = AF_{IN} \times \left(\frac{M}{N1 \times N2}\right)$$

$$= AF_{IN} \times \frac{(APLLDIV + 15)}{(APLLPRE + 1)(APLLPOST + 1)}$$

Where,
$N1 = APLLPRE + 1$
$N2 = APLLPOST + 1$
$M = APLLDIV + 15$

Note: When APLLDIV<2:0> = `111`, substitute (APLLDIV + 15) with (APLLDIV + 18) in Equation 11-1.

Equation 11-2 shows the relationship between the Auxiliary PLL Input (AFIN) clock frequency and the AVCO Frequency.

Equation 11-2: Avco Calculation

$$
\begin{aligned}
Avco &= AF_{IN} \times \left(\frac{M}{N}\right) \\
&= AF_{IN} \times \left(\frac{(PLLDIV + 15)}{(APLLPRE + 1)}\right)
\end{aligned}
$$

11.1 APLL Setup

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

11.1.1 SETUP FOR USING APLL WITH AUXILIARY OSCILLATOR WITH AN 8 MHz CRYSTAL

1. Clear the ASRCSEL bit to choose the Auxiliary Oscillator as the clock source for the APLL.
2. Clear the FRCSEL bit to choose the Auxiliary Oscillator at the clock source for the APLL.
3. Set the SELACLK bit to choose the Auxiliary PLL or oscillators to provide the source clock for the Auxiliary Clock divider.
4. Follow these steps to configure the APLL Phase Detector Input Divider bits (APLLPRE<2:0>), the Auxiliary PLL Feedback Divisor bits (APLLDIV<2:0>) and the APLL VCO Output Divider bits (APLLPOST<2:0>) to set up the APLL for a 48 MHz ACLK (used by the USB module) using an 8 MHz Auxiliary Oscillator:
   a) Select the APLL VCO output divider to meet the AVCO output frequency requirement (60 MHz < AVCO < 120 MHz).
      - Select an APLL VCO output divider ratio of N2 = 2
      - Ensure that AVCO = (ACLK x N2) = 96 MHz

   b) Select the APLL phase detector input divider to meet the APFD input frequency requirement (3 MHz < AFPLLI < 5.5 MHz).
      - Select an APLL phase detector input divider ratio of N1 = 2
      - Ensure that AFPLLI = (AFIN ÷ N1) = 4 MHz

   c) Select the Auxiliary PLL feedback divisor to generate the required VCO output frequency based on the PFD input frequency.
      - AVCO = AFPLLI x M
      - M = AVCO ÷ AFPLLI = 24
5. Enable the Auxiliary PLL by setting the ENAPLL bit.

Example 11-1 provides code for using the APLL with the Auxiliary Oscillator.

Example 11-1: Code Example for Using the APLL with the Auxiliary Oscillator

```
// Configure APLL prescaler, APLL postscaler, APLL divisor
ACLKCON3bits.ASRCSEL = 0; // Select Auxiliary Oscillator as the clock source
ACLKCON3bits.FRCSEL = 0;  // Select Auxiliary Oscillator as the clock source
ACLKCON3bits.SELACLK = 1; // Select Auxiliary PLL or oscillators to provide
                          // the source clock for auxiliary clock divider

ACLKDIV3bits.APLLDIV = 0b111;     // M = 24
ACLKCON3bits.APLLPRE = 0b001;     // N1 = 2
ACLKCON3bits.APLLPOST = 0b111;    // N2 = 2
ACLKCON3bits.ENAPLL = 1;          // Enable Auxiliary Clock
```

11.1.2 SETUP FOR USING APLL WITH PRIMARY OSCILLATOR USING AN 8 MHz CRYSTAL

1. Set the ASRCSEL bit to choose the Primary Oscillator as the clock source for the APLL.
2. Clear the FRCSEL bit to choose the Primary Oscillator as the clock source for the APLL.
3. Set the SELACLK bit to choose the Auxiliary PLL or oscillators to provide the clock source for the Auxiliary Clock divider.
4. Follow these steps to configure the APLL Phase Detector Input Divider bits (APLLPRE<2:0>), the Auxiliary PLL Feedback Divisor bits (APLLDIV<2:0>) and the APLL VCO Output Divider bits (APLLPOST<2:0>) to set up the APLL for a 48 MHz ACLK (used by the USB module) using an 8 MHz Auxiliary Oscillator:
   a) Select the APLL VCO output divider to meet the AVCO output frequency requirement (60 MHz < AVCO < 120 MHz).
      - Select an APLL VCO output divider ratio of N2 = 2
      - Ensure that AVCO = (ACLK x N2) = 96 MHz

   b) Select the APLL phase detector input divider to meet the APFD input frequency requirement (3 MHz < AFPLLI < 5.5 MHz).
      - Select an APLL phase detector input divider ratio of N1 = 2
      - Ensure that AVCO = (AFIN ÷ N1) = 4 MHz

   c) Select the Auxiliary PLL feedback divisor to generate the required VCO output frequency based on the PFD input frequency.
      - AVCO = AFPLLI x M
      - M = AVCO ÷ AFPLLI = 24
5. Enable the Auxiliary PLL by setting the ENAPLL bit.

Example 11-2 provides code for using the APLL with the Primary Oscillator.

Example 11-2: Code Example for Using the APLL with the Primary Oscillator

```
// Configure APLL prescaler, APLL postscaler, APLL divisor
ACLKCON3bits.ASRCSEL = 1; // Select Primary Oscillator as the clock source
ACLKCON3bits.FRCSEL = 0;  // Select Primary Oscillator as the clock source
ACLKCON3bits.SELACLK = 1; // Select Auxiliary PLL or oscillators to provide
                          // the source clock for auxiliary clock divider

ACLKDIV3bits.APLLDIV = 0b111;     // M = 24
ACLKCON3bits.APLLPRE = 0b001;     // N1 = 2
ACLKCON3bits.APLLPOST = 0b111;    // N2 = 2
ACLKCON3bits.ENAPLL = 1;          // Enable Auxiliary Clock
```

12.0 AUXILIARY PLL (x16)

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

The Auxiliary PLL can be used to provide a high-speed clock to peripherals, such as the PWM and the ADC. The ACLKCON register selects the reference clock and output dividers for obtaining the necessary Auxiliary Clock for the PWM and ADC modules. The Auxiliary Clock for the PWM and ADC can be either the:

- Internal FRC Oscillator (7.37 MHz nominal)
- Primary Oscillator
- Internal FRC Oscillator with PLL
- Primary Oscillator with PLL
- Auxiliary PLL

12.1 Enabling the Auxiliary PLL

To enable the Auxiliary PLL, the following steps must be performed:

1. Select the reference clock for the Auxiliary PLL by setting the ASRCSEL bit (ACLKCON<7>) for the POSC or by setting the FRCSEL bit (ACLKCON<6>) for the Internal FRC Oscillator.
2. Enable the Auxiliary PLL by setting the ENAPLL bit (ACLKCON<15>).
3. Select the clock source for the Auxiliary Clock output divider by setting the SELACLK bit (ACLKCON<13>).
4. Select the appropriate clock divider by setting the APSTSCLR<2:0> bits (ACLKCON<10:8>).
5. Ensure that the Auxiliary PLL has locked and is ready for operation. This is done by polling the APLLCK bit (ACLKCON<14>).

Example 12-1 illustrates a code example to set up the Auxiliary PLL for 120 MHz, using the Internal FRC Oscillator as a clock reference.

Example 12-1: Enabling the Auxiliary PLL

```
ACLKCONbits.FRCSEL = 1;           /* Internal FRC is clock source for auxiliary PLL */
ACLKCONbits.ENAPLL = 1;           /* APLL is enabled */
ACLKCONbits.SELACLK = 1;          /* Auxiliary PLL provides the source clock for the */
                                  /* clock divider */
ACLKCONbits.APSTSCLR = 7;         /* Auxiliary Clock Output Divider is Divide-by-1 */

while(ACLKCONbits.APLLCK != 1){}; /* Wait for Auxiliary PLL to Lock */

/* With 7.37 MHz FRC input selection, the Auxiliary Clock output will be 16x7.37 MHz = 118 MHz. */
```

> **Note 1:** If the Primary PLL is used as a source for the Auxiliary Clock, then the Primary PLL should be configured to a maximum operation of up to 30 MIPS or less.
>
> **2:** To achieve 1.04 ns PWM resolution, the Auxiliary Clock must use the x16 Auxiliary PLL and be set up for 120 MHz. All other clock sources will have a minimum PWM resolution of 8 ns.
>
> **3:** By using various combinations of clock inputs and PLL settings, it is possible to configure the oscillator out of specification. The user-assigned software must ensure that the Auxiliary Clock is configured to be within the electrical specification range of 112 MHz to 120 MHz.

12.2 Auxiliary Clock Divider

The Auxiliary Clock Output Divider bits (APSTSCLR<2:0>) in the Auxiliary Clock Control register (ACLKCON<10:8>) divide the Auxiliary Clock, which allow a lower frequency to be chosen. These bits allow for eight postscaler settings, from 1:1 to 1:256, as shown in Table 12-1.

The Auxiliary Clock postscaler must be configured to divide-by-1 (APSTSCLR<2:0> = `111`) for proper operation of the PWM module.

Table 12-1: Auxiliary Clock Output Divider Settings

APSTSCLR<2:0> Bit Value	Auxiliary Oscillator Setting
`111`	Divide-by-1 (default setting)
`110`	Divide-by-2
`101`	Divide-by-4
`100`	Divide-by-8
`011`	Divide-by-16
`010`	Divide-by-32
`001`	Divide-by-64
`000`	Divide-by-256

13.0 FAIL-SAFE CLOCK MONITOR (FSCM)

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

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

> **Note:** When the device is in Sleep mode, if the clock fails, the FSCM does not wake-up the device.

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

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

13.1 FSCM Delay

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

The FSCM delay (T_{FSCM}) is applied when the FSCM is enabled and the Primary or Secondary Oscillator is selected as the system clock.

For more information, refer to the *“dsPIC33E/PIC24E Family Reference Manual”*, **“Reset”** (DS70602). For recent documentation, visit the Microchip web site at www.microchip.com.

> **Note:** Refer to the **“Electrical Characteristics”** section of the specific device data sheet for T_{FSCM} values.

13.2 FSCM and WDT

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

14.0 CLOCK SWITCHING

Clock switching can be initiated as a result of a hardware event or a software request. A typical scenario includes:

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

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

dsPIC33/PIC24 family devices feature the Phase-Locked Loop Enable (PLLKEN) bit in the Oscillator Configuration register (FOSC<8>). Setting this bit will cause the device to wait until the PLL locks before switching to the PLL clock source. When this bit is set to ‘`0`’, the device will not wait for the PLL lock and will proceed with the clock switch. The default setting for this bit is ‘`1`’.

With few limitations, applications are free to switch between any of the four clock sources (POSC, SOSC, FRC and LPRC), under software control, at any time. To limit the possible side effects that could result from this flexibility, dsPIC33/PIC24 family devices have a safeguard lock built into the switch process. That is, the OSCCON register is write-protected during clock switching.

14.1 Enabling Clock Switching

The Clock Switching Mode Configuration bits (FCKSM<1:0>) in the Oscillator Configuration register (FOSC<7:6>) must be programmed to enable clock switching and the Fail-Safe Clock Monitor (see Table 14-1).

Table 14-1: Configurable Clock Switching Modes

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

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

14.2 Clock Switch Sequence

The recommended process for a clock switch is as follows:

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

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

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

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

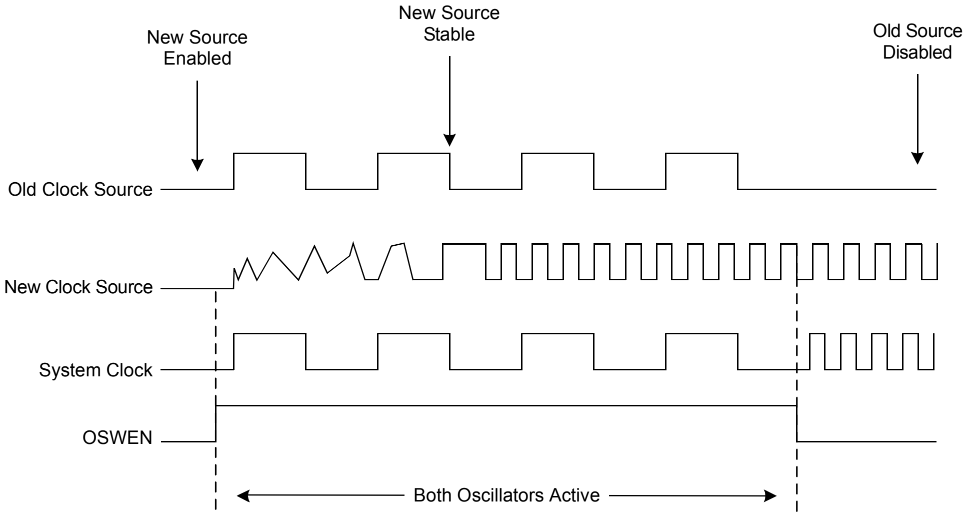
2: Direct clock switching between PLL modes is not possible. For example, clock switching should not occur between the Primary Oscillator with PLL and the Internal FRC oscillator with PLL.

3: Setting the CLKLOCK bit (OSCCON<7>) prevents clock switching when clock switching is enabled and Fail-Safe Clock Monitoring is disabled by Configuration bits, FCKSM<1:0> (FOSC<7:6>) = `01`. The CLKLOCK bit cannot be cleared after it is set by the software; it clears on a Power-on Reset.

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

5: The clock switch will not wait for the PLL lock if the PLLKEN bit in the Oscillator Configuration register (FOSC<8>) is set to ‘`0`’.

Figure 14-1: Clock Transition Timing Diagram



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

The following steps are the recommended code sequence for a clock switch:

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

> **Note:** MPLAB® C30 provides built-in C language functions for unlocking the OSCCON register:
>
> `__builtin_write_OSCCONH(value)`
> `__builtin_write_OSCCONL(OSCCON | value)`
>
> For more information, see the MPLAB IDE Help file.

Example 14-1 illustrates the code sequence for unlocking the OSCCON register and switching from FRC with the PLL clock to the LPRC clock source.

Example 14-1: Code Example for Clock Switching

```
;Place the New Oscillator Selection (NOSC=0b101) in W0
MOV     #0x5,WREG

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

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

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

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

; Enable Clock Switch
BSET    OSCON, #0;     Request Clock Switching by Setting OSWEN bit

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

14.3 Clock Switching Consideration

When you incorporate clock switching into an application, consider these points when designing the code:

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

14.4 Aborting a Clock Switch

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

Typical assembly code for aborting a clock switch is shown in Example 14-2. A clock switch procedure can be aborted at any time. A clock switch that is already in progress can also be aborted by performing a second clock switch.

Example 14-2: Aborting a Clock Switch

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

14.5 Entering Sleep Mode During a Clock Switch

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

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

15.0 TWO-SPEED START-UP

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

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

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

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

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

16.0 REFERENCE CLOCK OUTPUT

The reference clock output provides a clock signal to any remappable pin (RPn). The reference clock can be either the Primary Oscillator or the system clock.

The ROSEL bit (REFOCON<12>) in the Reference Oscillator Control register selects between the external oscillator and the system clock.

Figure 1-1 shows a block diagram for the reference clock. See the REFOCON register (Register 4-5) for the bits associated with the reference clock output. Refer to the specific device data sheet for more information on peripheral remapping.

17.0 LINEAR FEEDBACK SHIFT REGISTER

Some devices incorporate a Linear Feedback Shift Register (LFSR) which is composed of a 15-bit shift register, providing pseudorandom data with $2^{15} - 1$ unique values for use by the user via software. The pseudorandom number generator is incremented on every PWM cycle.

18.0 REGISTER MAPS

Table 18-1 maps the bit functions for the Oscillator Special Function Control registers. Table 18-2 maps the bit functions for the Oscillator Configuration registers.

Table 18-1: Oscillator Special Function Control Registers

File Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
OSCCON	—	COSC2	COSC1	COSC0	—	NOSC2	NOSC1	NOSC0	CLKLOCK	IOLOCK	LOCK	—	CF	—	LPOSCEN	OSWEN	`7700`(1)
CLKDIV	ROI	DOZE2	DOZE1	DOZE0	DOZEN	FRCDIV2	FRCDIV1	FRCDIV0	PLLPOST1	PLLPOST0	—	PLLPRE4	PLLPRE3	PLLPRE2	PLLPRE1	PLLPRE0	`3040`
PLLFBD	—	—	—	—	—	—	—	PLLDIV8	PLLDIV7	PLLDIV6	PLLDIV5	PLLDIV4	PLLDIV3	PLLDIV2	PLLDIV1	PLLDIV0	`0030`
OSCTUN	—	—	—	—	—	—	—	—	—	—	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0	`0000`
REFOCON	ROON	—	ROSSLP	ROSEL	RODIV3	RODIV2	RODIV1	RODIV0	—	—	—	—	—	—	—	—	`0000`
ACLKCON1	ENAPLL	APLLCK	SELACLK	—	—	APSTSCLR2	APSTSCLR1	APSTSCLR0	ASRCSEL	FRCSEL	—	—	—	—	—	—	`2740`
ACLKCON3	ENAPLL	—	SELACLK	AOSCMD1	AOSCMD0	ASRCSEL	FRCSEL	—	APLLPOST2	APLLPOST1	APLLPOST0	—	—	APLLPRE2	APLLPRE1	APLLPRE0	`0000`
ACLKDIV3	—	—	—	—	—	—	—	—	—	—	—	—	—	APLLDIV2	APLLDIV1	APLLDIV0	`0000`
LFSR	—	LFSR14	LFSR13	LFSR12	LFSR11	LFSR10	LFSR9	LFSR8	LFSR7	LFSR6	LFSR5	LFSR4	LFSR3	LFSR2	LFSR1	LFSR0	`0000`

Legend: — = unimplemented, read as ‘`0`’. Reset values are shown in hexadecimal.

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

Table 18-2: Oscillator Configuration Registers

File Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
FOSCSEL	—	—	—	—	—	—	—	—	IESO	—	—	—	—	FNOSC2	FNOSC1	FNOSC0	`0000`
FOSC	—	—	—	—	—	—	—	PLLKEN	FCKSM1	FCKSM0	IOL1WAY	—	—	OSCIOFNC	POSCMD1	POSCMD0	`0000`

Legend: — = unimplemented, read as ‘`0`’. Reset values are shown in hexadecimal.

19.0 RELATED APPLICATION NOTES

This section lists application notes that pertain to this section of the manual. These application notes may not be written specifically for the dsPIC33/PIC24 Product Family, but the concepts are pertinent and could be used with modification and possible limitations. The current application notes related to the Oscillator Module include:

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

Note: Please visit the Microchip web site (www.microchip.com) for additional application notes and code examples for the dsPIC33/PIC24 family of devices.

20.0 REVISION HISTORY

Revision A (August 2013)

This is the initial released revision of this document.

NOTES:

QUALITY MANAGEMENT SYSTEM CERTIFIED BY DNV

═ ISO/TS 16949 ═

Trademarks

♻ Printed on recycled paper.

ISBN: 978-1-62077-367-3

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

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

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

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

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

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

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

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

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

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hangzhou
Tel: 86-571-2819-3187
Fax: 86-571-2819-3189

China - Hong Kong SAR
Tel: 852-2943-5100
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Osaka
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

Japan - Tokyo
Tel: 81-3-6880- 3770
Fax: 81-3-6880-3771

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-213-7828
Fax: 886-7-330-9305

Taiwan - Taipei
Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

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

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

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

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

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

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

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

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

11/29/12