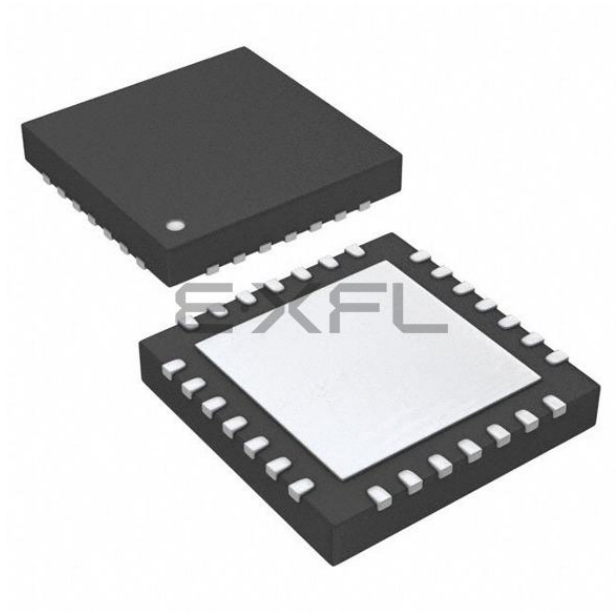


**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Active |
| Core Processor | PIC |
| Core Size | 8-Bit |
| Speed | 20MHz |
| Connectivity | - |
| Peripherals | Brown-out Detect/Reset, LCD, POR, PWM, WDT |
| Number of I/O | 25 |
| Program Memory Size | 7KB (4K x 14) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 256 x 8 |
| Voltage - Supply (Vcc/Vdd) | 1.8V ~ 3.6V |
| Data Converters | A/D 11x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 125°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 28-VQFN Exposed Pad |
| Supplier Device Package | 28-QFN (6x6) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1903-e-ml |

## TABLE 3-3: PIC16LF1902/3 MEMORY MAP

| | BANK 0 | | BANK 1 | | BANK 2 | | BANK 3 | | BANK 4 | | BANK 5 | | BANK 6 | | BANK 7 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | Core Registers (Table 3-2) | 080h | Core Registers (Table 3-2) | 100h | Core Registers (Table 3-2) | 180h | Core Registers (Table 3-2) | 200h | Core Registers (Table 3-2) | 280h | Core Registers (Table 3-2) | 300h | Core Registers (Table 3-2) | 380h | Core Registers (Table 3-2) |
| 00Bh | | 08Bh | | 10Bh | | 18Bh | | 20Bh | | 28Bh | | 30Bh | | 38Bh | |
| 00Ch | PORTA | 08Ch | TRISA | 10Ch | LATA | 18Ch | ANSELA | 20Ch | — | 28Ch | — | 30Ch | — | 38Ch | — |
| 00Dh | PORTB | 08Dh | TRISB | 10Dh | LATB | 18Dh | ANSELB | 20Dh | WPUB | 28Dh | — | 30Dh | — | 38Dh | — |
| 00Eh | PORTC | 08Eh | TRISC | 10Eh | LATC | 18Eh | — | 20Eh | — | 28Eh | — | 30Eh | — | 38Eh | — |
| 00Fh | — | 08Fh | — | 10Fh | — | 18Fh | — | 20Fh | — | 28Fh | — | 30Fh | — | 38Fh | — |
| 010h | PORTE | 090h | — | 110h | — | 190h | — | 210h | WPUE | 290h | — | 310h | — | 390h | — |
| 011h | PIR1 | 091h | PIE1 | 111h | — | 191h | PMADRL | 211h | — | 291h | — | 311h | — | 391h | — |
| 012h | PIR2 | 092h | PIE2 | 112h | — | 192h | PMADRH | 212h | — | 292h | — | 312h | — | 392h | — |
| 013h | — | 093h | — | 113h | — | 193h | PMDATL | 213h | — | 293h | — | 313h | — | 393h | — |
| 014h | — | 094h | — | 114h | — | 194h | PMDATH | 214h | — | 294h | — | 314h | — | 394h | IOCBP |
| 015h | TMR0 | 095h | OPTION_REG | 115h | — | 195h | PMCON1 | 215h | — | 295h | — | 315h | — | 395h | IOCBN |
| 016h | TMR1L | 096h | PCON | 116h | BORCON | 196h | PMCON2 | 216h | — | 296h | — | 316h | — | 396h | IOCBF |
| 017h | TMR1H | 097h | WDTCON | 117h | FVRCON | 197h | — | 217h | — | 297h | — | 317h | — | 397h | — |
| 018h | T1CON | 098h | — | 118h | — | 198h | — | 218h | — | 298h | — | 318h | — | 398h | — |
| 019h | T1GCON | 099h | OSCCON | 119h | — | 199h | — | 219h | — | 299h | — | 319h | — | 399h | — |
| 01Ah | — | 09Ah | OSCSTAT | 11Ah | — | 19Ah | — | 21Ah | — | 29Ah | — | 31Ah | — | 39Ah | — |
| 01Bh | — | 09Bh | ADRESL | 11Bh | — | 19Bh | — | 21Bh | — | 29Bh | — | 31Bh | — | 39Bh | — |
| 01Ch | — | 09Ch | ADRESH | 11Ch | — | 19Ch | — | 21Ch | — | 29Ch | — | 31Ch | — | 39Ch | — |
| 01Dh | — | 09Dh | ADCON0 | 11Dh | — | 19Dh | — | 21Dh | — | 29Dh | — | 31Dh | — | 39Dh | — |
| 01Eh | — | 09Eh | ADCON1 | 11Eh | — | 19Eh | — | 21Eh | — | 29Eh | — | 31Eh | — | 39Eh | — |
| 01Fh | — | 09Fh | — | 11Fh | — | 19Fh | — | 21Fh | — | 29Fh | — | 31Fh | — | 39Fh | — |
| 020h | General Purpose Register 96 Bytes | 0A0h | General Purpose Register 32 Bytes | 120h | General Purpose Register 80 Bytes(1) | 1A0h | Unimplemented Read as '0' | 220h | Unimplemented Read as '0' | 2A0h | Unimplemented Read as '0' | 320h | Unimplemented Read as '0' | 3A0h | Unimplemented Read as '0' |
| 06Fh | | 0EFh | General Purpose Register 48 Bytes(1) | 16Fh | | 1EFh | | 26Fh | | 2EFh | | 36Fh | | 3EFh | |
| 070h | Accesses 70h – 7Fh | 0F0h | Accesses 70h – 7Fh | 170h | Accesses 70h – 7Fh | 1F0h | Accesses 70h – 7Fh | 270h | Accesses 70h – 7Fh | 2F0h | Accesses 70h – 7Fh | 370h | Accesses 70h – 7Fh | 3F0h | Accesses 70h – 7Fh |
| 07Fh | | 0FFh | | 17Fh | | 1FFh | | 27Fh | | 2FFh | | 37Fh | | 3FFh | |

**Legend:** = Unimplemented data memory locations, read as '0'.

**Note 1:** PIC16LF1903 only.

## 5.0 RESETS

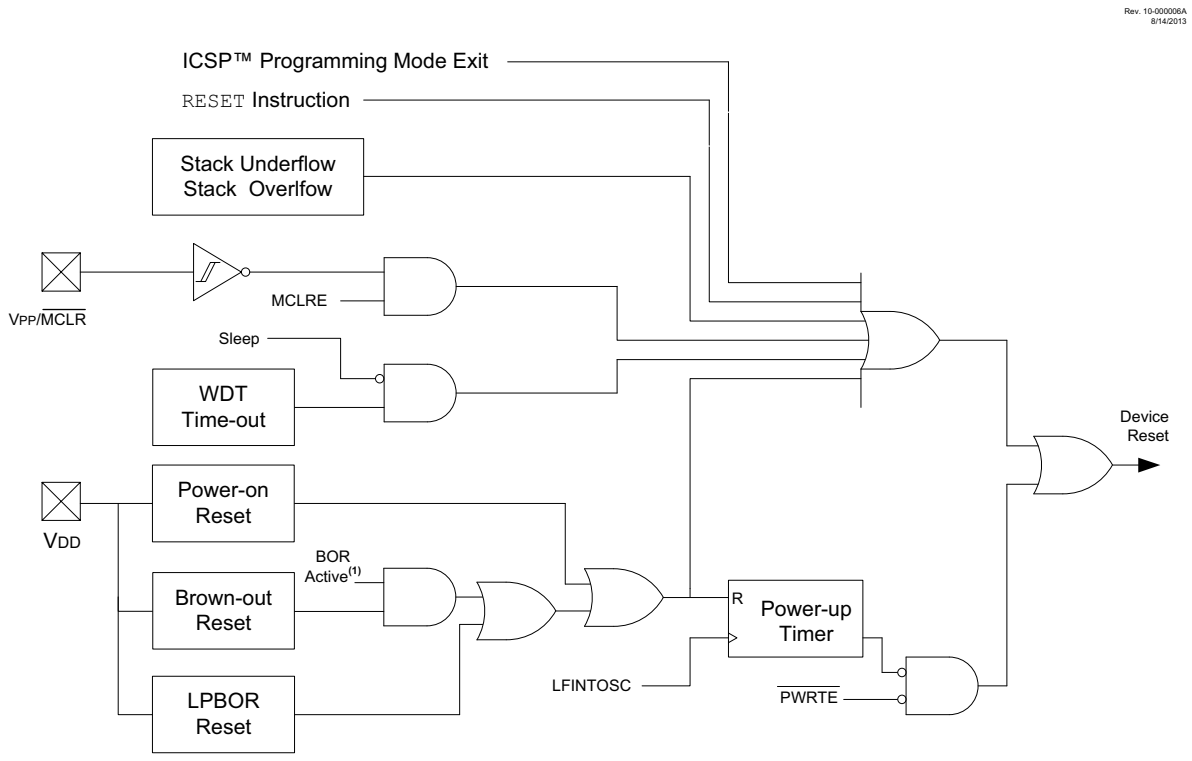
There are multiple ways to reset this device:

- Power-on Reset (POR)
- Brown-out Reset (BOR)
- Low-Power Brown-out Reset (LPBOR)
- $\overline{\text{MCLR}}$ Reset
- WDT Reset
- RESET instruction
- Stack Overflow
- Stack Underflow
- Programming mode exit

To allow VDD to stabilize, an optional power-up timer can be enabled to extend the Reset time after a BOR or POR event.

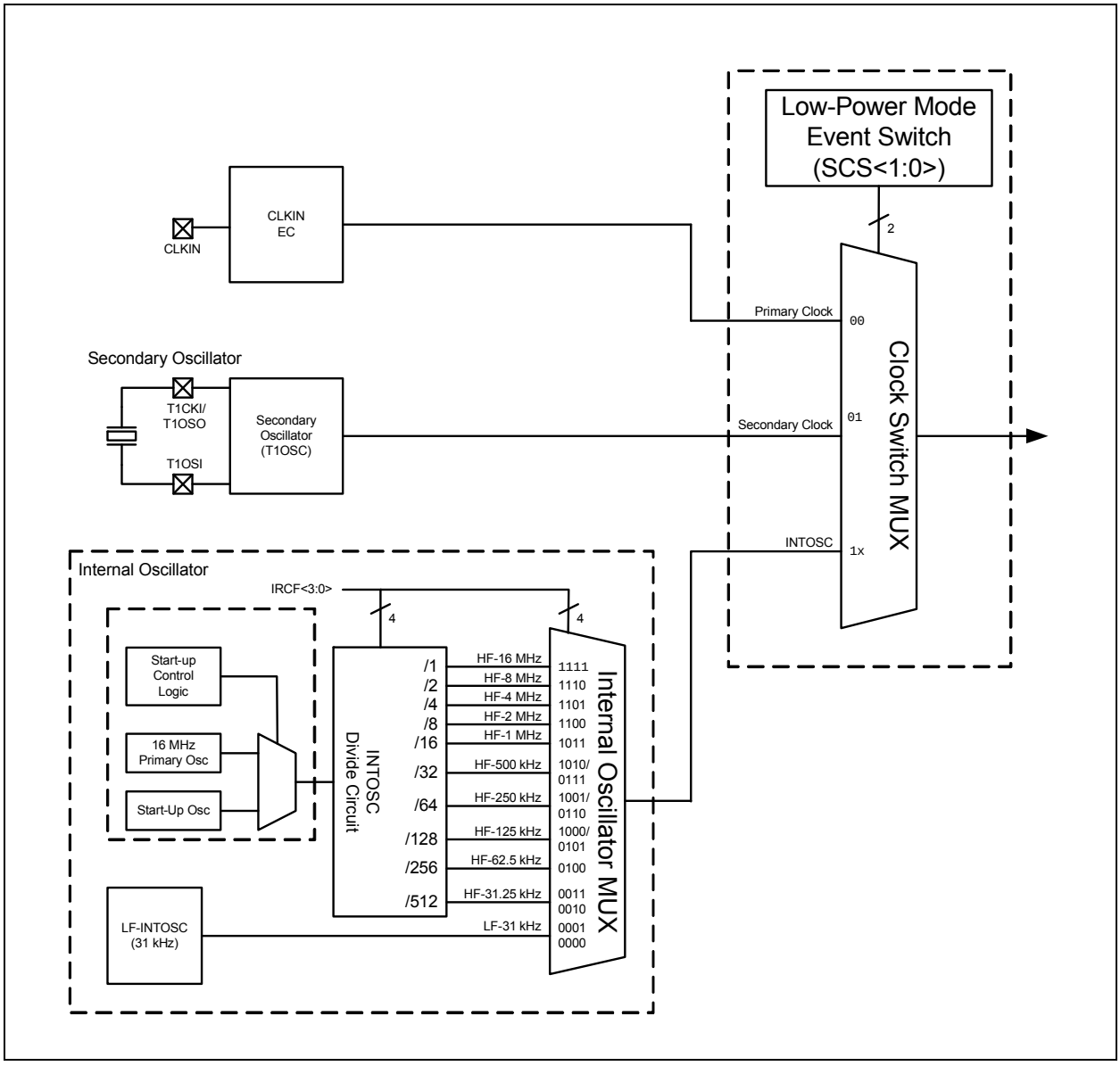
A simplified block diagram of the On-Chip Reset Circuit is shown in Figure 5-1.

**FIGURE 5-1: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT**




**Note 1:** See Table 5-1 for BOR active conditions.

**FIGURE 6-1: SIMPLIFIED PIC® MCU CLOCK SOURCE BLOCK DIAGRAM**



## 9.1 Independent Clock Source

The WDT derives its time base from the 31 kHz LFINTOSC internal oscillator. Time intervals in this chapter are based on a nominal interval of 1 ms. See **Section 21.0 "Electrical Specifications"** for the LFINTOSC tolerances.

## 9.2 WDT Operating Modes

The Watchdog Timer module has four operating modes controlled by the WDTE<1:0> bits in Configuration Word 1. See Table 9-1.

### 9.2.1 WDT IS ALWAYS ON

When the WDTE bits of Configuration Word 1 are set to '11', the WDT is always on.

WDT protection is active during Sleep.

### 9.2.2 WDT IS OFF IN SLEEP

When the WDTE bits of Configuration Word 1 are set to '10', the WDT is on, except in Sleep.

WDT protection is not active during Sleep.

### 9.2.3 WDT CONTROLLED BY SOFTWARE

When the WDTE bits of Configuration Word 1 are set to '01', the WDT is controlled by the SWDTEN bit of the WDTCON register.

WDT protection is unchanged by Sleep. See Table 9-1 for more details.

**TABLE 9-1: WDT OPERATING MODES**

| WDTE<1:0> | SWDTEN | Device Mode | WDT Mode |
|---|---|---|---|
| 11 | X | X | Active |
| 10 | X | Awake | Active |
| | | Sleep | Disabled |
| 01 | 1 | X | Active |
| | 0 | | Disabled |
| 00 | X | X | Disabled |

## 9.3 Time-Out Period

The WDTPS bits of the WDTCON register set the time-out period from 1 ms to 256 seconds (nominal). After a Reset, the default time-out period is two seconds.

## 9.4 Clearing the WDT

The WDT is cleared when any of the following conditions occur:

- Any Reset
- CLRWDT instruction is executed
- Device enters Sleep
- Device wakes up from Sleep
- Oscillator fail event
- WDT is disabled

See Table 9-2 for more information.

## 9.5 Operation During Sleep

When the device enters Sleep, the WDT is cleared. If the WDT is enabled during Sleep, the WDT resumes counting.

When the device exits Sleep, the WDT is cleared again.

When a WDT time-out occurs while the device is in Sleep, no Reset is generated. Instead, the device wakes up and resumes operation. The $\overline{TO}$ and $\overline{PD}$ bits in the STATUS register are changed to indicate the event. See **Section 3.0 "Memory Organization"** and STATUS register (**Register 3-1)** for more information.

**TABLE 9-2: WDT CLEARING CONDITIONS**

| Conditions | WDT |
|---|---|
| WDTE<1:0> = 00 | Cleared |
| WDTE<1:0> = 01 and SWDTEN = 0 | |
| WDTE<1:0> = 10 and enter Sleep | |
| CLRWDT Command | |
| Oscillator Fail Detected | |
| Exit Sleep + System Clock = T1OSC, EXTRC, INTOSC, EXTCLK | |
| Change INTOSC divider (IRCF bits) | Unaffected |

### 10.2.2 FLASH MEMORY UNLOCK SEQUENCE

The unlock sequence is a mechanism that protects the Flash Program Memory from unintended self-write programming or erasing. The sequence must be executed and completed without interruption to successfully complete any of the following operations:

- Row Erase
- Load program memory write latches
- Write of program memory write latches to program memory
- Write of program memory write latches to User IDs

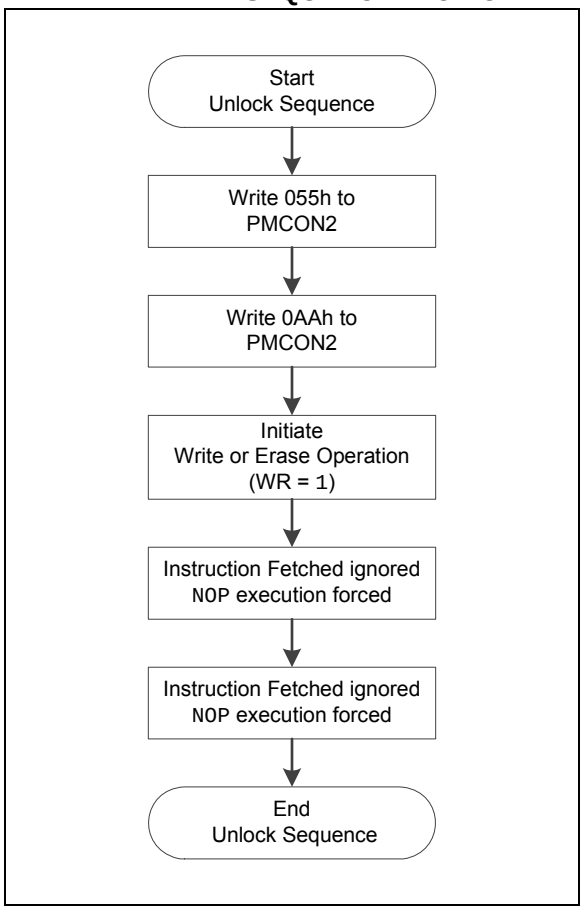
The unlock sequence consists of the following steps:

1. Write 55h to PMCON2
2. Write AAh to PMCON2
3. Set the WR bit in PMCON1
4. NOP instruction
5. NOP instruction

Once the WR bit is set, the processor will always force two NOP instructions. When an Erase Row or Program Row operation is being performed, the processor will stall internal operations (typical 2 ms), until the operation is complete and then resume with the next instruction. When the operation is loading the program memory write latches, the processor will always force the two NOP instructions and continue uninterrupted with the next instruction.

Since the unlock sequence must not be interrupted, global interrupts should be disabled prior to the unlock sequence and re-enabled after the unlock sequence is completed.

**FIGURE 10-3: FLASH PROGRAM MEMORY UNLOCK SEQUENCE FLOWCHART**



### 10.2.4 WRITING TO FLASH PROGRAM MEMORY

Program memory is programmed using the following steps:

1. Load the address in PMADRH:PMADRL of the row to be programmed.
2. Load each write latch with data.
3. Initiate a programming operation.
4. Repeat steps 1 through 3 until all data is written.

Before writing to program memory, the word(s) to be written must be erased or previously unwritten. Program memory can only be erased one row at a time. No automatic erase occurs upon the initiation of the write.

Program memory can be written one or more words at a time. The maximum number of words written at one time is equal to the number of write latches. See Figure 10-5 (row writes to program memory with 32 write latches) for more details.

The write latches are aligned to the Flash row address boundary defined by the upper ten bits of PMADRH:PMADRL, (PMADRH<6:0>:PMADRL<7:5>) with the lower five bits of PMADRL, (PMADRL<4:0>) determining the write latch being loaded. Write operations do not cross these boundaries. At the completion of a program memory write operation, the data in the write latches is reset to contain 0x3FFF.

The following steps should be completed to load the write latches and program a row of program memory. These steps are divided into two parts. First, each write latch is loaded with data from the PMDATH:PMDATL using the unlock sequence with LWLO = 1. When the last word to be loaded into the write latch is ready, the LWLO bit is cleared and the unlock sequence executed. This initiates the programming operation, writing all the latches into Flash Program Memory.

| **Note:** | The special unlock sequence is required to load a write latch with data or initiate a Flash programming operation. If the unlock sequence is interrupted, writing to the latches or program memory will not be initiated. |
|---|---|

1. Set the WREN bit of the PMCON1 register.
2. Clear the CFGS bit of the PMCON1 register.
3. Set the LWLO bit of the PMCON1 register. When the LWLO bit of the PMCON1 register is '1', the write sequence will only load the write latches and will not initiate the write to Flash Program Memory.
4. Load the PMADRH:PMADRL register pair with the address of the location to be written.
5. Load the PMDATH:PMDATL register pair with the program memory data to be written.
6. Execute the unlock sequence (**Section 10.2.2 "Flash Memory Unlock Sequence"**). The write latch is now loaded.
7. Increment the PMADRH:PMADRL register pair to point to the next location.
8. Repeat steps 5 through 7 until all but the last write latch has been loaded.
9. Clear the LWLO bit of the PMCON1 register. When the LWLO bit of the PMCON1 register is '0', the write sequence will initiate the write to Flash Program Memory.
10. Load the PMDATH:PMDATL register pair with the program memory data to be written.
11. Execute the unlock sequence (**Section 10.2.2 "Flash Memory Unlock Sequence"**). The entire program memory latch content is now written to Flash Program Memory.

| **Note:** | The program memory write latches are reset to the blank state (0x3FFF) at the completion of every write or erase operation. As a result, it is not necessary to load all the program memory write latches. Unloaded latches will remain in the blank state. |
|---|---|

An example of the complete write sequence is shown in Example 10-3. The initial address is loaded into the PMADRH:PMADRL register pair; the data is loaded using indirect addressing.

**REGISTER 11-8: ANSELB: PORTB ANALOG SELECT REGISTER**

| U-0 | U-0 | R/W-1/1 | R/W-1/1 | R/W-1/1 | R/W-1/1 | R/W-1/1 | R/W-1/1 |
|---|---|---|---|---|---|---|---|
| — | — | ANSB5 | ANSB4 | ANSB3 | ANSB2 | ANSB1 | ANSB0 |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | |
|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| u = Bit is unchanged | x = Bit is unknown | -n/n = Value at POR and BOR/Value at all other Resets |
| '1' = Bit is set | '0' = Bit is cleared | |

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

bit 5-0 **ANSB<5:0>**: Analog Select between Analog or Digital Function on pins RB<5:0>, respectively
0 = Digital I/O. Pin is assigned to port or digital special function.
1 = Analog input. Pin is assigned as analog input(1). Digital input buffer disabled.

**Note 1:** When setting a pin to an analog input, the corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

**REGISTER 11-9: WPUB: WEAK PULL-UP PORTB REGISTER**

| R/W-1/1 | R/W-1/1 | R/W-1/1 | R/W-1/1 | R/W-1/1 | R/W-1/1 | R/W-1/1 | R/W-1/1 |
|---|---|---|---|---|---|---|---|
| WPUB7 | WPUB6 | WPUB5 | WPUB4 | WPUB3 | WPUB2 | WPUB1 | WPUB0 |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | |
|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| u = Bit is unchanged | x = Bit is unknown | -n/n = Value at POR and BOR/Value at all other Resets |
| '1' = Bit is set | '0' = Bit is cleared | |

bit 7-0 **WPUB<7:0>**: Weak Pull-up Register bits
1 = Pull-up enabled
0 = Pull-up disabled

**Note 1:** Global $\overline{\text{WPUEN}}$ bit of the OPTION_REG register must be cleared for individual pull-ups to be enabled.
**2:** The weak pull-up device is automatically disabled if the pin is in configured as an output.

**TABLE 11-6: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB**

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---|---|---|---|---|---|---|---|---|---|
| ANSELB | — | — | ANSB5 | ANSB4 | ANSB3 | ANSB2 | ANSB1 | ANSB0 | 94 |
| LATB | LATB7 | LATB6 | LATB5 | LATB4 | LATB3 | LATB2 | LATB1 | LATB0 | 93 |
| PORTB | RB7 | RB6 | RB5 | RB4 | RB3 | RB2 | RB1 | RB0 | 93 |
| TRISB | TRISB7 | TRISB6 | TRISB5 | TRISB4 | TRISB3 | TRISB2 | TRISB1 | TRISB0 | 93 |
| WPUB | WPUB7 | WPUB6 | WPUB5 | WPUB4 | WPUB3 | WPUB2 | WPUB1 | WPUB0 | 94 |

**Legend:** x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTB.

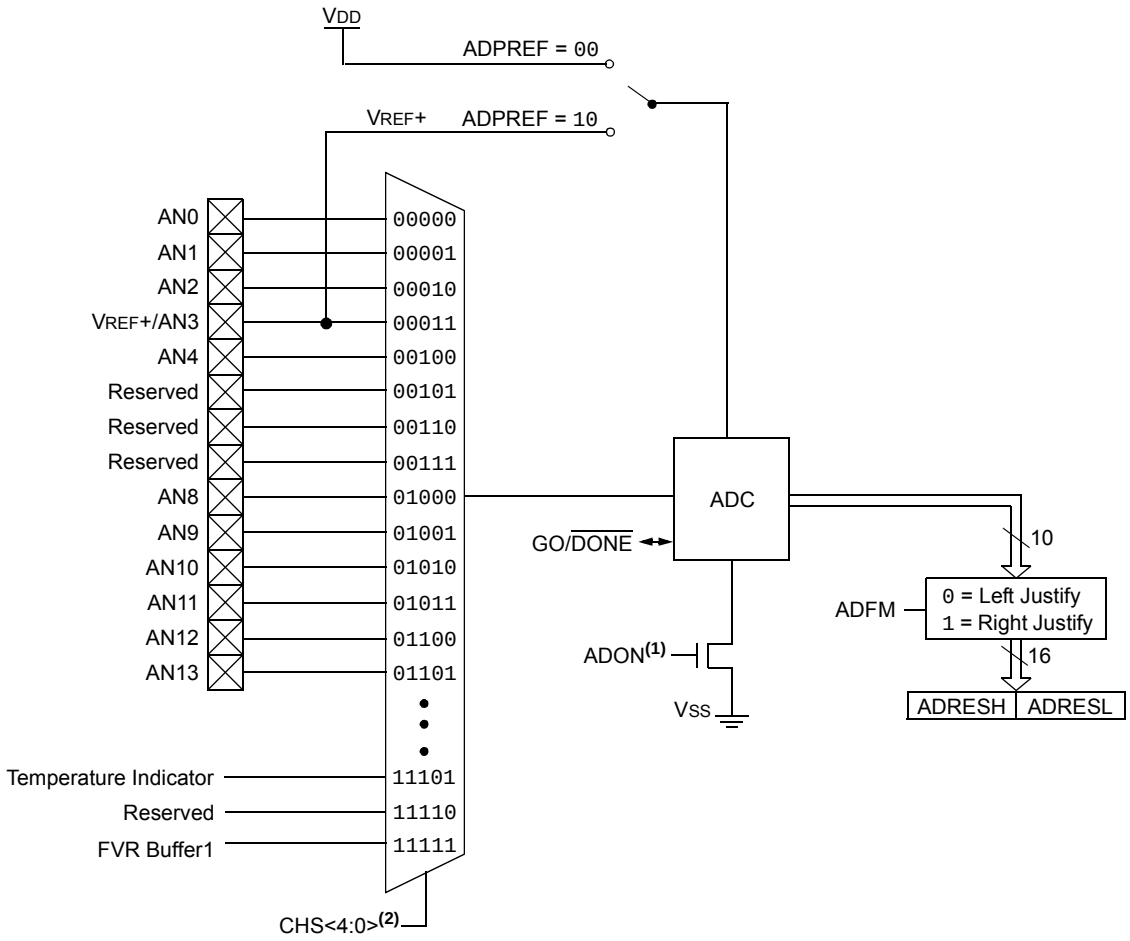
## 15.0 ANALOG-TO-DIGITAL CONVERTER (ADC) MODULE

The Analog-to-Digital Converter (ADC) allows conversion of an analog input signal to a 10-bit binary representation of that signal. This device uses analog inputs, which are multiplexed into a single sample and hold circuit. The output of the sample and hold is connected to the input of the converter. The converter generates a 10-bit binary result via successive approximation and stores the conversion result into the ADC result registers (ADRESH:ADRESL register pair). Figure 15-1 shows the block diagram of the ADC.

The ADC voltage reference is software selectable to be either internally generated or externally supplied.

The ADC can generate an interrupt upon completion of a conversion. This interrupt can be used to wake-up the device from Sleep.

**FIGURE 15-1: ADC BLOCK DIAGRAM**




**Note 1:** When ADON = 0, all multiplexer inputs are disconnected.

**2:** See ADCON0 register (Example 15-1) for detailed analog channel selection per device.

### 15.3 A/D Acquisition Requirements

For the ADC to meet its specified accuracy, the charge holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The Analog Input model is shown in Figure 15-4. The source impedance (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD), refer to Figure 15-4. **The maximum recommended impedance for analog sources is 10 kΩ.** As the source impedance is decreased, the acquisition time may be decreased. After the analog input channel is selected (or changed), an A/D acquisition must be done before the conversion can be started. To calculate the minimum acquisition time, Equation 15-1 may be used. This equation assumes that 1/2 LSb error is used (1,024 steps for the ADC). The 1/2 LSb error is the maximum error allowed for the ADC to meet its specified resolution.

**EQUATION 15-1: ACQUISITION TIME EXAMPLE**

*Assumptions:* $Temperature = 50°C$ *and external impedance of* $10k\Omega$ *5.0V* $V_{DD}$

$$T_{ACQ} = \text{Amplifier Settling Time} + \text{Hold Capacitor Charging Time} + \text{Temperature Coefficient}$$
$$= T_{AMP} + T_C + T_{COFF}$$
$$= 2\mu s + T_C + [(Temperature - 25°C)(0.05\mu s/°C)]$$

*The value for* $T_C$ *can be approximated with the following equations:*

$$V_{APPLIED}\left(1 - \frac{1}{(2^{n+1}) - 1}\right) = V_{CHOLD}$$ ;[1] $V_{CHOLD}$ *charged to within 1/2 lsb*

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{CHOLD}$$ ;[2] $V_{CHOLD}$ *charge response to* $V_{APPLIED}$

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{APPLIED}\left(1 - \frac{1}{(2^{n+1}) - 1}\right)$$ ;combining [1] and [2]

*Note: Where n = number of bits of the ADC.*

*Solving for* $T_C$*:*

$$T_C = -C_{HOLD}(R_{IC} + R_{SS} + R_S)\ ln(1/2047)$$
$$= -12.5pF(1k\Omega + 7k\Omega + 10k\Omega)\ ln(0.0004885)$$
$$= 1.715\mu s$$

*Therefore:*

$$T_{ACQ} = 2\mu s + 1.715\mu s + [(50°C - 25°C)(0.05\mu s/°C)]$$
$$= 4.96\mu s$$

**Note 1:** The reference voltage (VREF) has no effect on the equation, since it cancels itself out.

**2:** The charge holding capacitor (CHOLD) is not discharged after each conversion.

**3:** The maximum recommended impedance for analog sources is 10 kΩ. This is required to meet the pin leakage specification.

### 16.1.3 SOFTWARE PROGRAMMABLE PRESCALER

A software programmable prescaler is available for exclusive use with Timer0. The prescaler is enabled by clearing the PSA bit of the OPTION_REG register.

| | |
|---|---|
| **Note:** | The Watchdog Timer (WDT) uses its own independent prescaler. |

There are eight prescaler options for the Timer0 module ranging from 1:2 to 1:256. The prescale values are selectable via the PS<2:0> bits of the OPTION_REG register. In order to have a 1:1 prescaler value for the Timer0 module, the prescaler must be disabled by setting the PSA bit of the OPTION_REG register.

The prescaler is not readable or writable. All instructions writing to the TMR0 register will clear the prescaler.

### 16.1.4 TIMER0 INTERRUPT

Timer0 will generate an interrupt when the TMR0 register overflows from FFh to 00h. The TMR0IF interrupt flag bit of the INTCON register is set every time the TMR0 register overflows, regardless of whether or not the Timer0 interrupt is enabled. The TMR0IF bit can only be cleared in software. The Timer0 interrupt enable is the TMR0IE bit of the INTCON register.

| | |
|---|---|
| **Note:** | The Timer0 interrupt cannot wake the processor from Sleep since the timer is frozen during Sleep. |

### 16.1.5 8-BIT COUNTER MODE SYNCHRONIZATION

When in 8-Bit Counter mode, the incrementing edge on the T0CKI pin must be synchronized to the instruction clock. Synchronization can be accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the instruction clock. The high and low periods of the external clocking source must meet the timing requirements as shown in **Section 21.0 "Electrical Specifications"**.

### 16.1.6 OPERATION DURING SLEEP

Timer0 cannot operate while the processor is in Sleep mode. The contents of the TMR0 register will remain unchanged while the processor is in Sleep mode.

#### 17.6.2.1 T1G Pin Gate Operation

The T1G pin is one source for Timer1 gate control. It can be used to supply an external source to the Timer1 gate circuitry.

#### 17.6.2.2 Timer0 Overflow Gate Operation

When Timer0 increments from FFh to 00h, a low-to-high pulse will automatically be generated and internally supplied to the Timer1 gate circuitry.

### 17.6.3 TIMER1 GATE TOGGLE MODE

When Timer1 Gate Toggle mode is enabled, it is possible to measure the full-cycle length of a Timer1 gate signal, as opposed to the duration of a single level pulse.

The Timer1 gate source is routed through a flip-flop that changes state on every incrementing edge of the signal. See Figure 17-4 for timing details.

Timer1 Gate Toggle mode is enabled by setting the T1GTM bit of the T1GCON register. When the T1GTM bit is cleared, the flip-flop is cleared and held clear. This is necessary in order to control which edge is measured.

| **Note:** | Enabling Toggle mode at the same time as changing the gate polarity may result in indeterminate operation. |
|---|---|

### 17.6.4 TIMER1 GATE SINGLE-PULSE MODE

When Timer1 Gate Single-Pulse mode is enabled, it is possible to capture a single pulse gate event. Timer1 Gate Single-Pulse mode is first enabled by setting the T1GSPM bit in the T1GCON register. Next, the T1GGO/$\overline{\text{DONE}}$ bit in the T1GCON register must be set. The Timer1 will be fully enabled on the next incrementing edge. On the next trailing edge of the pulse, the T1GGO/$\overline{\text{DONE}}$ bit will automatically be cleared. No other gate events will be allowed to increment Timer1 until the T1GGO/$\overline{\text{DONE}}$ bit is once again set in software. See Figure 17-5 for timing details.

If the Single Pulse Gate mode is disabled by clearing the T1GSPM bit in the T1GCON register, the T1GGO/$\overline{\text{DONE}}$ bit should also be cleared.

Enabling the Toggle mode and the Single-Pulse mode simultaneously will permit both sections to work together. This allows the cycle times on the Timer1 gate source to be measured. See Figure 17-6 for timing details.

### 17.6.5 TIMER1 GATE VALUE STATUS

When Timer1 gate value status is utilized, it is possible to read the most current level of the gate control value. The value is stored in the T1GVAL bit in the T1GCON register. The T1GVAL bit is valid even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

### 17.6.6 TIMER1 GATE EVENT INTERRUPT

When Timer1 Gate Event Interrupt is enabled, it is possible to generate an interrupt upon the completion of a gate event. When the falling edge of T1GVAL occurs, the TMR1GIF flag bit in the PIR1 register will be set. If the TMR1GIE bit in the PIE1 register is set, then an interrupt will be recognized.

The TMR1GIF flag bit operates even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

**REGISTER 18-4: LCDCST: LCD CONTRAST CONTROL REGISTER**

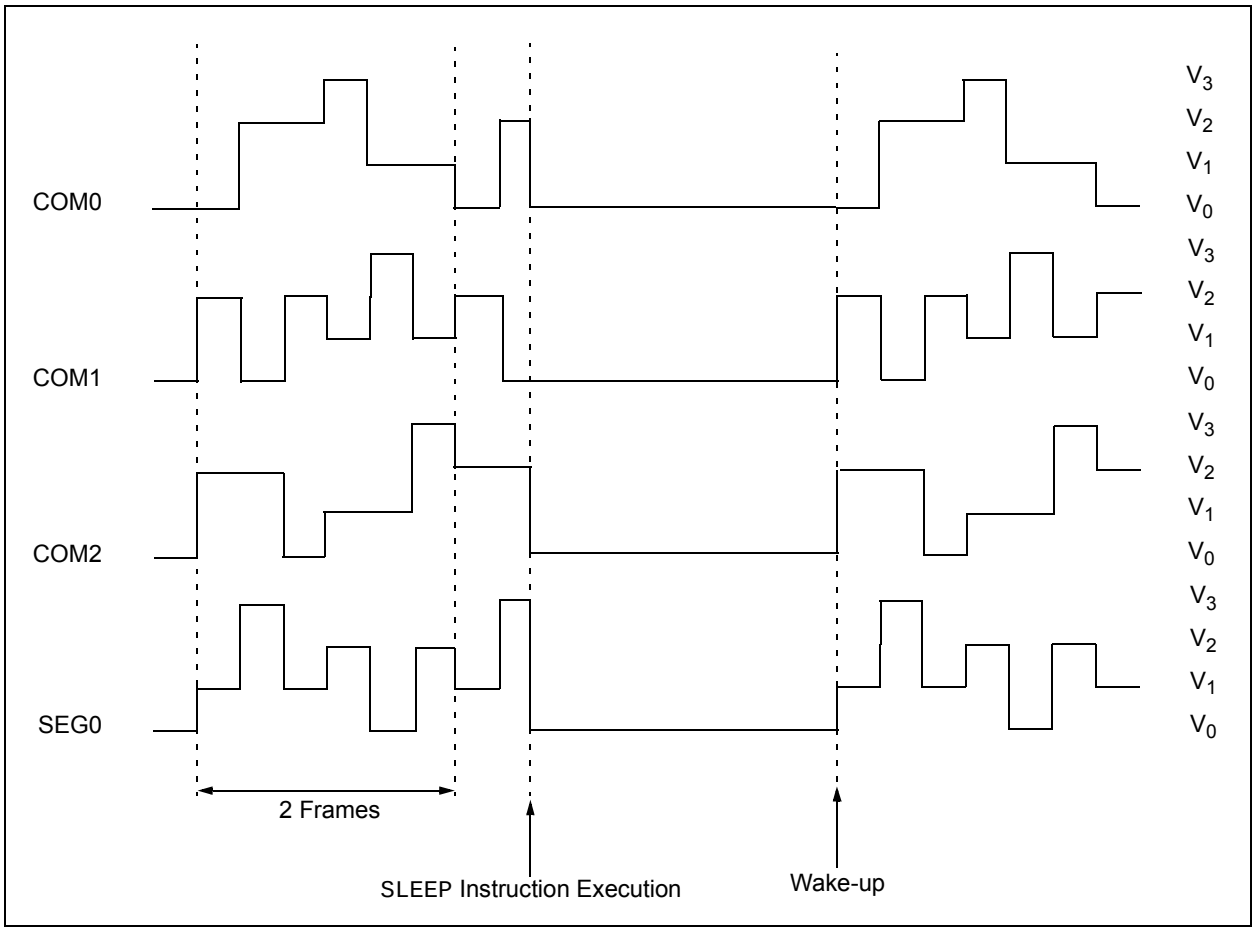
| U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | LCDCST<2:0> | | |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | | |
|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ |
| u = Bit is unchanged | x = Bit is unknown | -n/n = Value at POR and BOR/Value at all other Resets |
| ‘1’ = Bit is set | ‘0’ = Bit is cleared | C = Only clearable bit |

bit 7-3 **Unimplemented:** Read as ‘0’

bit 2-0 **LCDCST<2:0>:** LCD Contrast Control bits
Selects the resistance of the LCD contrast control resistor ladder
Bit Value = Resistor ladder
000 = Minimum Resistance (Maximum contrast). Resistor ladder is shorted.
001 = Resistor ladder is at 1/7th of maximum resistance
010 = Resistor ladder is at 2/7th of maximum resistance
011 = Resistor ladder is at 3/7th of maximum resistance
100 = Resistor ladder is at 4/7th of maximum resistance
101 = Resistor ladder is at 5/7th of maximum resistance
110 = Resistor ladder is at 6/7th of maximum resistance
111 = Resistor ladder is at maximum resistance (Minimum contrast).

**FIGURE 18-20: SLEEP ENTRY/EXIT WHEN SLPEN = 1**



## 20.0 INSTRUCTION SET SUMMARY

Each PIC16 instruction is a 14-bit word containing the operation code (opcode) and all required operands. The opcodes are broken into three broad categories.

- Byte Oriented
- Bit Oriented
- Literal and Control

The literal and control category contains the most varied instruction word format.

Table 20-3 lists the instructions recognized by the MPASM™ assembler.

All instructions are executed within a single instruction cycle, with the following exceptions, which may take two or three cycles:

- Subroutine takes two cycles (CALL, CALLW)
- Returns from interrupts or subroutines take two cycles (RETURN, RETLW, RETFIE)
- Program branching takes two cycles (GOTO, BRA, BRW, BTFSS, BTFSC, DECFSZ, INCSFZ)
- One additional instruction cycle will be used when any instruction references an indirect file register and the file select register is pointing to program memory.

One instruction cycle consists of four oscillator cycles; for an oscillator frequency of 4 MHz, this gives a nominal instruction execution rate of 1 MHz.

All instruction examples use the format '0xhh' to represent a hexadecimal number, where 'h' signifies a hexadecimal digit.

### 20.1 Read-Modify-Write Operations

Any instruction that specifies a file register as part of the instruction performs a Read-Modify-Write (R-M-W) operation. The register is read, the data is modified, and the result is stored according to either the instruction, or the destination designator 'd'. A read operation is performed on a register even if the instruction writes to that register.

**TABLE 20-1: OPCODE FIELD DESCRIPTIONS**

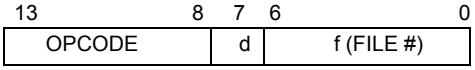
| Field | Description |
|---|---|
| f | Register file address (0x00 to 0x7F) |
| W | Working register (accumulator) |
| b | Bit address within an 8-bit file register |
| k | Literal field, constant data or label |
| x | Don't care location (= 0 or 1).<br>The assembler will generate code with x = 0.<br>It is the recommended form of use for compatibility with all Microchip software tools. |
| d | Destination select; d = 0: store result in W,<br>d = 1: store result in file register f.<br>Default is d = 1. |
| n | FSR or INDF number. (0-1) |
| mm | Pre-post increment-decrement mode selection |

**TABLE 20-2: ABBREVIATION DESCRIPTIONS**

| Field | Description |
|---|---|
| PC | Program Counter |
| $\overline{TO}$ | Time-out bit |
| C | Carry bit |
| DC | Digit carry bit |
| Z | Zero bit |
| $\overline{PD}$ | Power-down bit |

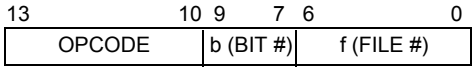
**FIGURE 20-1: GENERAL FORMAT FOR INSTRUCTIONS**

**Byte-oriented file register operations**




d = 0 for destination W
d = 1 for destination f
f = 7-bit file register address

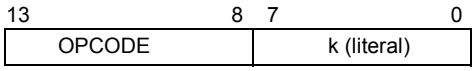
**Bit-oriented file register operations**




b = 3-bit bit address
f = 7-bit file register address

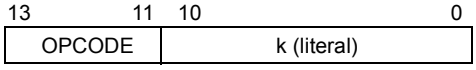
**Literal and control operations**

General



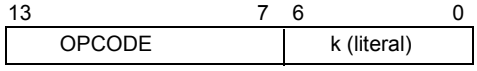

k = 8-bit immediate value

CALL and GOTO instructions only



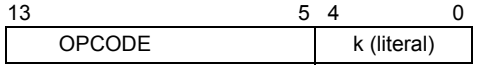

k = 11-bit immediate value

MOVLP instruction only



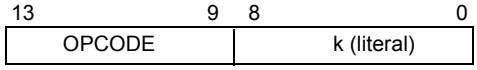

k = 7-bit immediate value

MOVLB instruction only



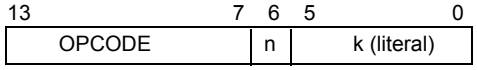

k = 5-bit immediate value

BRA instruction only



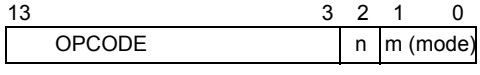

k = 9-bit immediate value

FSR Offset instructions



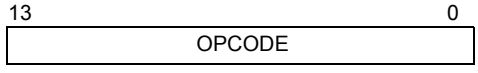

n = appropriate FSR
k = 6-bit immediate value

FSR Increment instructions




n = appropriate FSR
m = 2-bit mode value

OPCODE only



**RRF** **Rotate Right f through Carry**

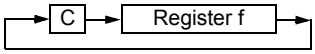
Syntax: [ *label* ] RRF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: See description below

Status Affected: C

Description: The contents of register 'f' are rotated one bit to the right through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.




**SLEEP** **Enter Sleep mode**

Syntax: [ *label* ] SLEEP

Operands: None

Operation: 00h → WDT,
0 → WDT prescaler,
1 → $\overline{TO}$,
0 → $\overline{PD}$

Status Affected: $\overline{TO}$, $\overline{PD}$

Description: The power-down Status bit, $\overline{PD}$ is cleared. Time-out Status bit, $\overline{TO}$ is set. Watchdog Timer and its prescaler are cleared.
The processor is put into Sleep mode with the oscillator stopped.

**SUBLW** **Subtract W from literal**

Syntax: [ *label* ] SUBLW k

Operands: $0 \le k \le 255$

Operation: k - (W) → (W)

Status Affected: C, DC, Z

Description: The W register is subtracted (2's complement method) from the 8-bit literal 'k'. The result is placed in the W register.

| | |
|---|---|
| C = 0 | W > k |
| C = 1 | W ≤ k |
| DC = 0 | W<3:0> > k<3:0> |
| DC = 1 | W<3:0> ≤ k<3:0> |

**SUBWF** **Subtract W from f**

Syntax: [ *label* ] SUBWF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) - (W) → (destination)

Status Affected: C, DC, Z

Description: Subtract (2's complement method) W register from register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f.

| | |
|---|---|
| C = 0 | W > f |
| C = 1 | W ≤ f |
| DC = 0 | W<3:0> > f<3:0> |
| DC = 1 | W<3:0> ≤ f<3:0> |

**SUBWFB** **Subtract W from f with Borrow**

Syntax: SUBWFB f {,d}

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) – (W) – ($\overline{B}$) → dest

Status Affected: C, DC, Z

Description: Subtract W and the BORROW flag (CARRY) from register 'f' (2's complement method). If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'.

### TABLE 21-7: CLOCK OSCILLATOR TIMING REQUIREMENTS

| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Param. No.** | **Sym.** | **Characteristic** | **Min.** | **Typ†** | **Max.** | **Units** | **Conditions** |
| OS01 | FOSC | External CLKIN Frequency(1) | DC | — | 0.5 | MHz | External Clock (ECL) |
| | | | DC | — | 4 | MHz | External Clock (ECM) |
| | | | DC | — | 20 | MHz | External Clock (ECH) |
| OS02 | TOSC | External CLKIN Period(1) | 50 | — | ∞ | ns | External Clock (EC) |
| OS03 | TCY | Instruction Cycle Time(1) | 200 | TCY | DC | ns | TCY = 4/FOSC |

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**Note 1:** Instruction cycle period (TCY) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min" values with an external clock applied to CLKIN pin. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

### TABLE 21-8: OSCILLATOR PARAMETERS

| Standard Operating Conditions (unless otherwise stated) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Param. No.** | **Sym.** | **Characteristic** | **Freq. Tolerance** | **Min.** | **Typ†** | **Max.** | **Units** | **Conditions** |
| OS08 | HFOSC | Internal Calibrated HFINTOSC Frequency(2) | ±8% | — | 16 | — | MHz | 0°C ≤ TA ≤ +85°C |
| | | | ±6.5% | — | 16 | — | MHz | VDD = 3.0V at +25°C |
| OS10A* | TIOSC ST | HFINTOSC 16 MHz | — | — | 5 | 15 | µs | VDD = 2.0V, -40°C to +85°C |
| | | Oscillator Wake-up from Sleep Start-up Time | — | — | 5 | 15 | µs | VDD = 3.0V, -40°C to +85°C |

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**Note 1:** Instruction cycle period (TCY) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min" values with an external clock applied to the OSC1 pin. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

**2:** To ensure these oscillator frequency tolerances, VDD and VSS must be capacitively decoupled as close to the device as possible. 0.1 µF and 0.01 µF values in parallel are recommended.

**FIGURE 21-5: CLKOUT AND I/O TIMING**

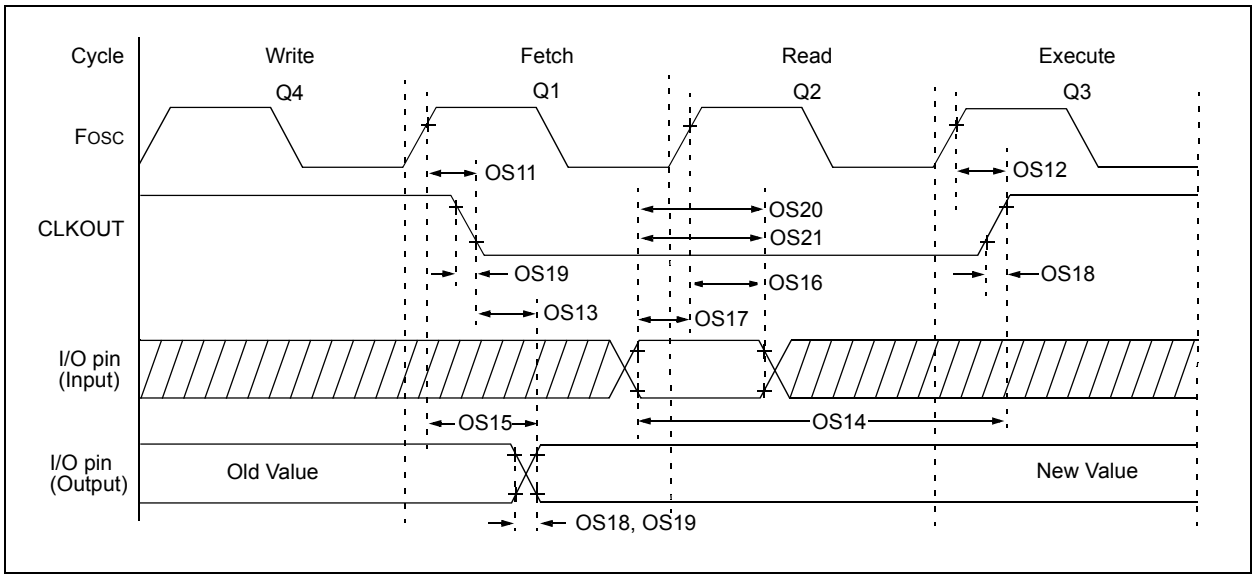



**TABLE 21-9: CLKOUT AND I/O TIMING PARAMETERS**

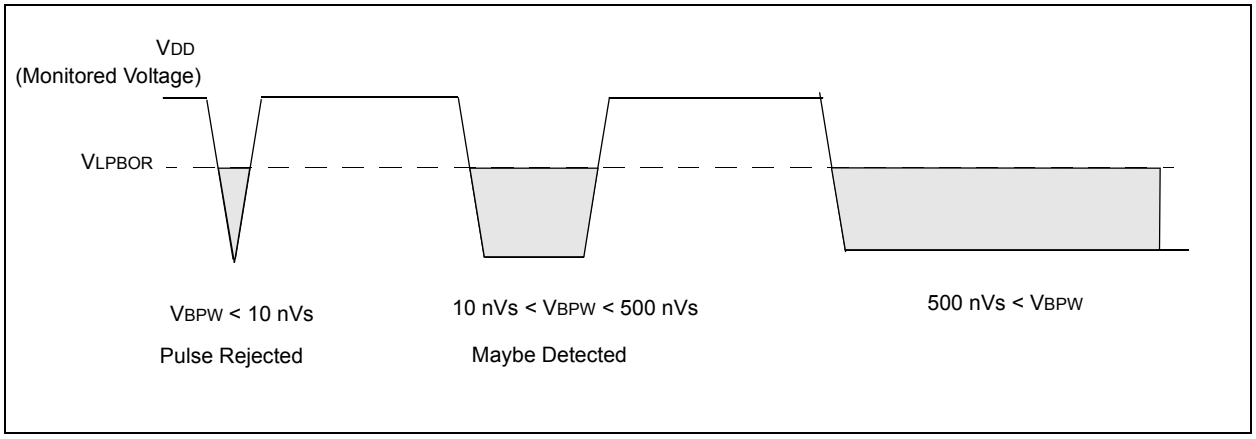
| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym.** | **Characteristic** | **Min.** | **Typ†** | **Max.** | **Units** | **Conditions** |
| OS11 | TosH2ckL | FOSC↑ to CLKOUT↓ (1) | — | — | 70 | ns | VDD = 3.0V |
| OS12 | TosH2ckH | FOSC↑ to CLKOUT↑ (1) | — | — | 72 | ns | VDD = 3.0V |
| OS13 | TckL2ioV | CLKOUT↓ to Port out valid(1) | — | — | 20 | ns | |
| OS14 | TioV2ckH | Port input valid before CLKOUT↑(1) | TOSC + 200 ns | — | — | ns | |
| OS15 | TosH2ioV | FOSC↑ (Q1 cycle) to Port out valid | — | 50 | 70* | ns | VDD = 3.0V |
| OS16 | TosH2ioI | FOSC↑ (Q2 cycle) to Port input invalid (I/O in hold time) | 50 | — | — | ns | VDD = 3.0V |
| OS17 | TioV2osH | Port input valid to FOSC↑ (Q2 cycle) (I/O in setup time) | 20 | — | — | ns | |
| OS18 | TioR | Port output rise time | —<br>— | 40<br>15 | 72<br>32 | ns | VDD = 3.0V<br>VDD = 2.0V |
| OS19 | TioF | Port output fall time | —<br>— | 28<br>15 | 55<br>30 | ns | VDD = 2.0V<br>VDD = 3.0V |
| OS20* | Tinp | INT pin input high or low time | 25 | — | — | ns | |
| OS21* | Tioc | Interrupt-on-change new input level time | 25 | — | — | ns | |

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated.

**Note 1:** Measurements are taken in EC mode where CLKOUT output is 4 x TOSC.

**FIGURE 21-8: MINIMUM PULSE WIDTH FOR LPBOR DETECTION**



## APPENDIX A: DATA SHEET REVISION HISTORY

### Revision A (01/2011)

Original release.

### Revision B (04/2011)

Revised Sections: Flexible Oscillator Structure; Low-Power Features; Electrical Specifications; Changed ULPBOR to LPBOR.

### Revision C (07/2014)

Updated Example 3-2, Register 4-2 and Table 5-1; Updated section 6, Oscillator Module, and section 9, Watchdog Timer; Updated table 10-3 and Figure 18-7; Removed Figure 19-1; Updated section 21, Electrical Specifications; Other minor corrections.

### Revision D (04/2015)

Updated Equation 15-1; Figures 5-1, 18-7, 21-2, and 21-8; Register 4-2; Sections 4.1, 18.4.5, and 21.0; and Table 5-1, 9-2, 10-4, 21-1, and 21-10.

### Revision E (01/2016)

Added ‘Memory’ section; Updated ‘PIC16LF1902/3 Family Types’ table; Other minor corrections.

### Revision F (05/2016)

Updated Figure 21-2. Removed parameter 34A in Table 21-10. Removed Table 18-7, LCD Worksheet. Other minor corrections.