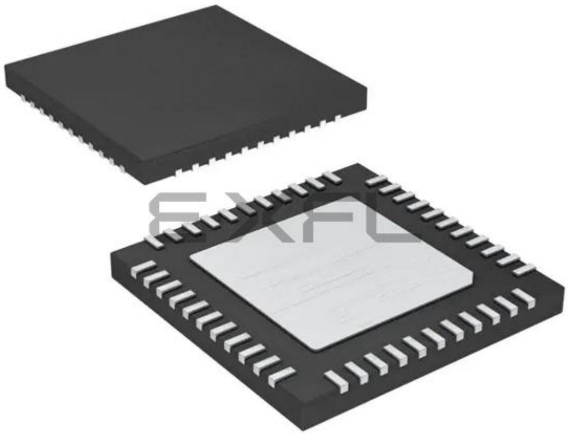


Welcome to E-XFL.COM

## 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 | 48MHz |
| Connectivity | I²C, SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, HLVD, POR, PWM, WDT |
| Number of I/O | 35 |
| Program Memory Size | 16KB (8K x 16) |
| Program Memory Type | FLASH |
| EEPROM Size | 256 x 8 |
| RAM Size | 768 x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.3V ~ 5.5V |
| Data Converters | A/D 30x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 125°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 44-VQFN Exposed Pad |
| Supplier Device Package | 44-QFN (8x8) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic18f44k22-e-ml |

**TABLE 1-3: PIC18(L)F4XK22 PINOUT I/O DESCRIPTIONS (CONTINUED)**

| Pin Number | | | | Pin Name | Pin Type | Buffer Type | Description |
|---|---|---|---|---|---|---|---|
| PDIP | TQFP | QFN | UQFN | | | | |
| 18 | 37 | 37 | 33 | RC3/SCK1/SCL1/AN15 | | | |
| | | | | RC3 | I/O | ST | Digital I/O. |
| | | | | SCK1 | I/O | ST | Synchronous serial clock input/output for SPI mode (MSSP). |
| | | | | SCL1 | I/O | ST | Synchronous serial clock input/output for I$^2$C mode (MSSP). |
| | | | | AN15 | I | Analog | Analog input 15. |
| 23 | 42 | 42 | 38 | RC4/SDI1/SDA1/AN16 | | | |
| | | | | RC4 | I/O | ST | Digital I/O. |
| | | | | SDI1 | I | ST | SPI data in (MSSP). |
| | | | | SDA1 | I/O | ST | I$^2$C data I/O (MSSP). |
| | | | | AN16 | I | Analog | Analog input 16. |
| 24 | 43 | 43 | 39 | RC5/SDO1/AN17 | | | |
| | | | | RC5 | I/O | ST | Digital I/O. |
| | | | | SDO1 | O | — | SPI data out (MSSP). |
| | | | | AN17 | I | Analog | Analog input 17. |
| 25 | 44 | 44 | 40 | RC6/TX1/CK1/AN18 | | | |
| | | | | RC6 | I/O | ST | Digital I/O. |
| | | | | TX1 | O | — | EUSART asynchronous transmit. |
| | | | | CK1 | I/O | ST | EUSART synchronous clock (see related RXx/DTx). |
| | | | | AN18 | I | Analog | Analog input 18. |
| 26 | 1 | 1 | 1 | RC7/RX1/DT1/AN19 | | | |
| | | | | RC7 | I/O | ST | Digital I/O. |
| | | | | RX1 | I | ST | EUSART asynchronous receive. |
| | | | | DT1 | I/O | ST | EUSART synchronous data (see related TXx/CKx). |
| | | | | AN19 | I | Analog | Analog input 19. |
| 19 | 38 | 38 | 34 | RD0/SCK2/SCL2/AN20 | | | |
| | | | | RD0 | I/O | ST | Digital I/O. |
| | | | | SCK2 | I/O | ST | Synchronous serial clock input/output for SPI mode (MSSP). |
| | | | | SCL2 | I/O | ST | Synchronous serial clock input/output for I$^2$C mode (MSSP). |
| | | | | AN20 | I | Analog | Analog input 20. |
| 20 | 39 | 39 | 35 | RD1/CCP4/SDI2/SDA2/AN21 | | | |
| | | | | RD1 | I/O | ST | Digital I/O. |
| | | | | CCP4 | I/O | ST | Capture 4 input/Compare 4 output/PWM 4 output. |
| | | | | SDI2 | I | ST | SPI data in (MSSP). |
| | | | | SDA2 | I/O | ST | I$^2$C data I/O (MSSP). |
| | | | | AN21 | I | Analog | Analog input 21. |

**Legend:** TTL = TTL compatible input CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; I = Input; O = Output; P = Power.

**Note 1:** Default pin assignment for P2B, T3CKI, CCP3/P3A and CCP2/P2A when Configuration bits PB2MX, T3CMX, CCP3MX and CCP2MX are set.

**2:** Alternate pin assignment for P2B, T3CKI, CCP3/P3A and CCP2/P2A when Configuration bits PB2MX, T3CMX, CCP3MX and CCP2MX are clear.

## 2.2 Oscillator Control

The OSCCON, OSCCON2 and OSCTUNE registers (Register 2-1 to Register 2-3) control several aspects of the device clock's operation, both in full-power operation and in power-managed modes.

- Main System Clock Selection (SCS)
- Primary Oscillator Circuit Shutdown (PRISD)
- Secondary Oscillator Enable (SOSCGO)
- Primary Clock Frequency 4x multiplier (PLLEN)
- Internal Frequency selection bits (IRCF, INTSRC)
- Clock Status bits (OSTS, HFIOFS, MFIOFS, LFIOFS. SOSCRUN, PLLRDY)
- Power management selection (IDLEN)

### 2.2.1 MAIN SYSTEM CLOCK SELECTION

The System Clock Select bits, SCS<1:0>, select the main clock source. The available clock sources are

- Primary clock defined by the FOSC<3:0> bits of CONFIG1H. The primary clock can be the primary oscillator, an external clock, or the internal oscillator block.
- Secondary clock (secondary oscillator)
- Internal oscillator block (HFINTOSC, MFINTOSC and LFINTOSC).

The clock source changes immediately after one or more of the bits is written to, following a brief clock transition interval. The SCS bits are cleared to select the primary clock on all forms of Reset.

### 2.2.2 INTERNAL FREQUENCY SELECTION

The Internal Oscillator Frequency Select bits (IRCF<2:0>) select the frequency output of the internal oscillator block. The choices are the LFINTOSC source (31.25 kHz), the MFINTOSC source (31.25 kHz, 250 kHz or 500 kHz) and the HFINTOSC source (16 MHz) or one of the frequencies derived from the HFINTOSC postscaler (31.25 kHz to 8 MHz). If the internal oscillator block is supplying the main clock, changing the states of these bits will have an immediate change on the internal oscillator's output. On device Resets, the output frequency of the internal oscillator is set to the default frequency of 1 MHz.

### 2.2.3 LOW FREQUENCY SELECTION

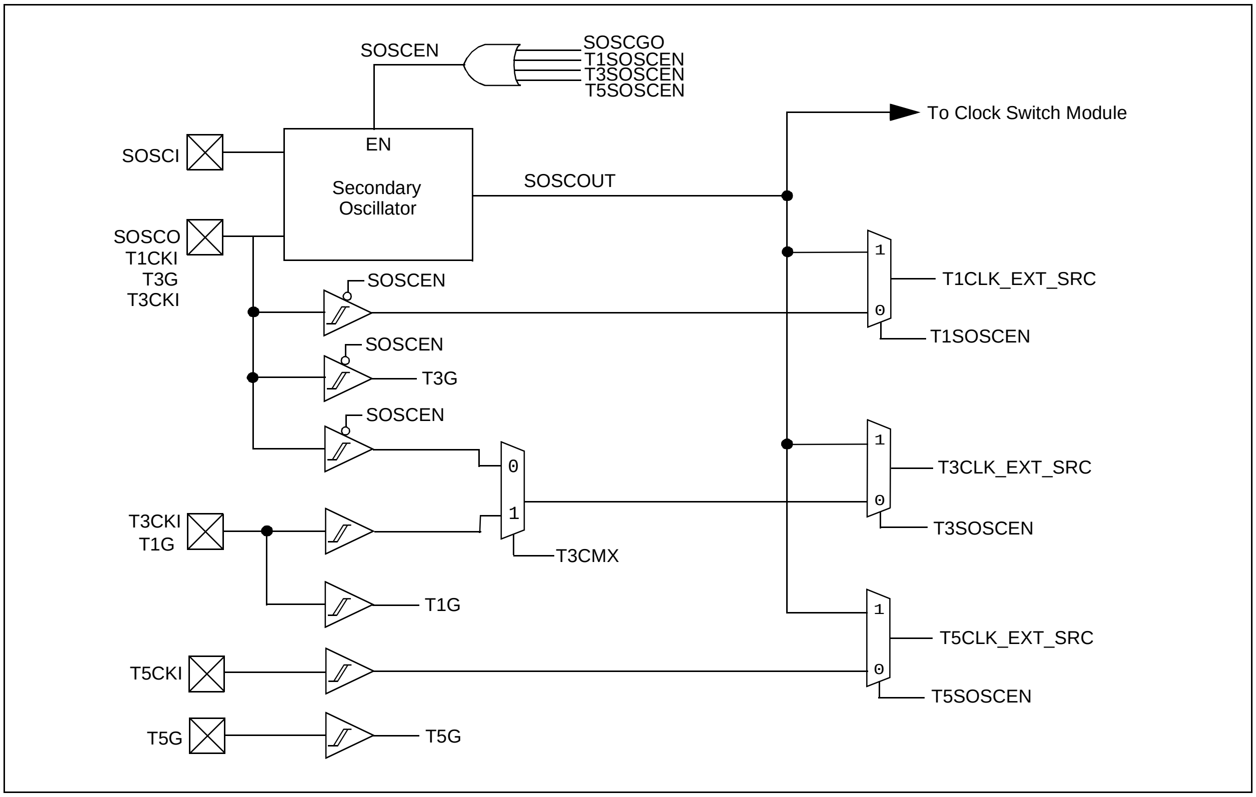
When a nominal output frequency of 31.25 kHz is selected (IRCF<2:0> = `000`), users may choose which internal oscillator acts as the source. This is done with the INTSRC bit of the OSCTUNE register and MFIOSEL bit of the OSCCON2 register. See Figure 2-2 and Register 2-1 for specific 31.25 kHz selection. This option allows users to select a 31.25 kHz clock (MFINTOSC or HFINTOSC) that can be tuned using the TUN<5:0> bits in OSCTUNE register, while maintaining power savings with a very low clock speed. LFINTOSC always remains the clock source for features such as the Watchdog Timer and the Fail-Safe Clock Monitor, regardless of the setting of INTSRC and MFIOSEL bits

This option allows users to select the tunable and more precise HFINTOSC as a clock source, while maintaining power savings with a very low clock speed.

### 2.2.4 POWER MANAGEMENT

The IDLEN bit of the OSCCON register determines whether the device goes into Sleep mode or one of the Idle modes when the `SLEEP` instruction is executed.

**FIGURE 2-4: SECONDARY OSCILLATOR AND EXTERNAL CLOCK INPUTS**



## 2.4 Clock Source Modes

Clock Source modes can be classified as external or internal.

- External Clock modes rely on external circuitry for the clock source. Examples are: Clock modules (EC mode), quartz crystal resonators or ceramic resonators (LP, XT and HS modes) and Resistor-Capacitor (RC mode) circuits.
- Internal clock sources are contained internally within the Oscillator block. The Oscillator block has three internal oscillators: the 16 MHz High-Frequency Internal Oscillator (HFINTOSC), 500 kHz Medium-Frequency Internal Oscillator (MFINTOSC) and the 31.25 kHz Low-Frequency Internal Oscillator (LFINTOSC).

The system clock can be selected between external or internal clock sources via the System Clock Select (SCS<1:0>) bits of the OSCCON register. See **Section 2.11 "Clock Switching"** for additional information.

## 2.5 External Clock Modes

### 2.5.1 OSCILLATOR START-UP TIMER (OST)

When the oscillator module is configured for LP, XT or HS modes, the Oscillator Start-up Timer (OST) counts 1024 oscillations from OSC1. This occurs following a Power-on Reset (POR) and when the Power-up Timer (PWRT) has expired (if configured), or a wake-up from Sleep. During this time, the program counter does not increment and program execution is suspended. The OST ensures that the oscillator circuit, using a quartz crystal resonator or ceramic resonator, has started and is providing a stable system clock to the oscillator module. When switching between clock sources, a delay is required to allow the new clock to stabilize. These oscillator delays are shown in Table 2-2.

In order to minimize latency between external oscillator start-up and code execution, the Two-Speed Clock Start-up mode can be selected (see **Section 2.12 "Two-Speed Clock Start-up Mode"**).

**TABLE 2-2: OSCILLATOR DELAY EXAMPLES**

| Switch From | Switch To | Frequency | Oscillator Delay |
|---|---|---|---|
| Sleep/POR/BOR | LFINTOSC<br>MFINTOSC<br>HFINTOSC | 31.25 kHz<br>31.25 kHz to 500 kHz<br>31.25 kHz to 16 MHz | Oscillator Start-up Delay ($T_{IOSC\_ST}$) |
| Sleep/POR/BOR | EC, RC | DC – 64 MHz | 2 instruction cycles |
| LFINTOSC (31.25 kHz) | EC, RC | DC – 64 MHz | 1 cycle of each |
| Sleep/POR/BOR | LP, XT, HS | 32 kHz to 40 MHz | 1024 Clock Cycles (OST) |
| Sleep/POR/BOR | 4xPLL | 32 MHz to 64 MHz | 1024 Clock Cycles (OST) + 2 ms |
| LFINTOSC (31.25 kHz) | LFINTOSC<br>HFINTOSC | 31.25 kHz to 16 MHz | 1 $\mu$s (approx.) |

### 2.5.2 EC MODE

The External Clock (EC) mode allows an externally generated logic level as the system clock source. When operating in this mode, an external clock source is connected to the OSC1 input and the OSC2 is available for general purpose I/O. Figure 2-5 shows the pin connections for EC mode.

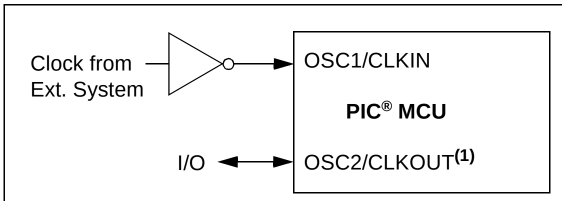
The External Clock (EC) offers different power modes, Low Power (ECLP), Medium Power (ECMP) and High Power (ECHP), selectable by the FOSC<3:0> bits. Each mode is best suited for a certain range of frequencies. The ranges are:

- ECLP – below 500 kHz
- ECMP – between 500 kHz and 16 MHz
- ECHP – above 16 MHz

The Oscillator Start-up Timer (OST) is disabled when EC mode is selected. Therefore, there is no delay in operation after a Power-on Reset (POR) or wake-up from Sleep.

Because the PIC® MCU design is fully static, stopping the external clock input will have the effect of halting the device while leaving all data intact. Upon restarting the external clock, the device will resume operation as if no time had elapsed.

**FIGURE 2-5: EXTERNAL CLOCK (EC) MODE OPERATION**




**Note 1:** Alternate pin functions are listed in **Section 1.0 "Device Overview"**.

### 2.11.3 CLOCK SWITCH TIMING

When switching between one oscillator and another, the new oscillator may not be operating which saves power (see Figure 2-9). If this is the case, there is a delay after the SCS<1:0> bits of the OSCCON register are modified before the frequency change takes place. The OSTS and IOFS bits of the OSCCON register will reflect the current active status of the external and HFINTOSC oscillators. The timing of a frequency selection is as follows:

1. SCS<1:0> bits of the OSCCON register are modified.
2. The old clock continues to operate until the new clock is ready.
3. Clock switch circuitry waits for two consecutive rising edges of the old clock after the new clock ready signal goes true.
4. The system clock is held low starting at the next falling edge of the old clock.
5. Clock switch circuitry waits for an additional two rising edges of the new clock.
6. On the next falling edge of the new clock the low hold on the system clock is released and new clock is switched in as the system clock.
7. Clock switch is complete.

See Figure 2-1 for more details.

If the HFINTOSC is the source of both the old and new frequency, there is no start-up delay before the new frequency is active. This is because the old and new frequencies are derived from the HFINTOSC via the postscaler and multiplexer.

Start-up delay specifications are located in **Section 27.0 "Electrical Specifications"**, under AC Specifications (Oscillator Module).

## 2.12 Two-Speed Clock Start-up Mode

Two-Speed Start-up mode provides additional power savings by minimizing the latency between external oscillator start-up and code execution. In applications that make heavy use of the Sleep mode, Two-Speed Start-up will remove the external oscillator start-up time from the time spent awake and can reduce the overall power consumption of the device.

This mode allows the application to wake-up from Sleep, perform a few instructions using the HFINTOSC as the clock source and go back to Sleep without waiting for the primary oscillator to become stable.

| **Note:** | Executing a `SLEEP` instruction will abort the oscillator start-up time and will cause the OSTS bit of the OSCCON register to remain clear. |
|---|---|

When the oscillator module is configured for LP, XT or HS modes, the Oscillator Start-up Timer (OST) is enabled (see **Section 2.5.1 "Oscillator Start-up Timer (OST)"**). The OST will suspend program execution until 1024 oscillations are counted. Two-Speed Start-up mode minimizes the delay in code execution by operating from the internal oscillator as the OST is counting. When the OST count reaches 1024 and the OSTS bit of the OSCCON register is set, program execution switches to the external oscillator.

### 2.12.1 TWO-SPEED START-UP MODE CONFIGURATION

Two-Speed Start-up mode is enabled when all of the following settings are configured as noted:

- Two-Speed Start-up mode is enabled when the IESO of the CONFIG1H Configuration register is set.
- SCS<1:0> (of the OSCCON register) = `00`.
- FOSC<2:0> bits of the CONFIG1H Configuration register are configured for LP, XT or HS mode.

Two-Speed Start-up mode becomes active after:

- Power-on Reset (POR) and, if enabled, after Power-up Timer (PWRT) has expired, or
- Wake-up from Sleep.

## 3.5 Exiting Idle and Sleep Modes

An exit from Sleep mode or any of the Idle modes is triggered by any one of the following:

- an interrupt
- a Reset
- a Watchdog Time-out

This section discusses the triggers that cause exits from power-managed modes. The clocking subsystem actions are discussed in each of the power-managed modes (see **Section 3.2 "Run Modes", Section 3.3 "Sleep Mode"** and **Section 3.4 "Idle Modes"**).

### 3.5.1 EXIT BY INTERRUPT

Any of the available interrupt sources can cause the device to exit from an Idle mode or the Sleep mode to a Run mode. To enable this functionality, an interrupt source must be enabled by setting its enable bit in one of the INTCON or PIE registers. The exit sequence is initiated when the corresponding interrupt flag bit is set.

The instruction immediately following the `SLEEP` instruction is executed on all exits by interrupt from Idle or Sleep modes. Code execution then branches to the interrupt vector if the GIE/GIEH bit of the INTCON register is set, otherwise code execution continues without branching (see **Section 9.0 "Interrupts"**).

A fixed delay of interval $T_{CSD}$ following the wake event is required when leaving Sleep and Idle modes. This delay is required for the CPU to prepare for execution. Instruction execution resumes on the first clock cycle following this delay.

### 3.5.2 EXIT BY WDT TIME-OUT

A WDT time-out will cause different actions depending on which power-managed mode the device is in when the time-out occurs.

If the device is not executing code (all Idle modes and Sleep mode), the time-out will result in an exit from the power-managed mode (see **Section 3.2 "Run Modes"** and **Section 3.3 "Sleep Mode"**). If the device is executing code (all Run modes), the time-out will result in a WDT Reset (see **Section 24.3 "Watchdog Timer (WDT)"**).

The WDT timer and postscaler are cleared by any one of the following:

- executing a `SLEEP` instruction
- executing a `CLRWDT` instruction
- the loss of the currently selected clock source when the Fail-Safe Clock Monitor is enabled
- modifying the IRCF bits in the OSCCON register when the internal oscillator block is the device clock source

### 3.5.3 EXIT BY RESET

Exiting Sleep and Idle modes by Reset causes code execution to restart at address '0'. See **Section 4.0 "Reset"** for more details.

The exit delay time from Reset to the start of code execution depends on both the clock sources before and after the wake-up and the type of oscillator.

### 3.5.4 EXIT WITHOUT AN OSCILLATOR START-UP DELAY

Certain exits from power-managed modes do not invoke the OST at all. There are two cases:

- PRI_IDLE mode, where the primary clock source is not stopped and
- the primary clock source is not any of the LP, XT, HS or HSPLL modes.

In these instances, the primary clock source either does not require an oscillator start-up delay since it is already running (PRI_IDLE), or normally does not require an oscillator start-up delay (RC, EC, INTOSC, and INTOSCIO modes). However, a fixed delay of interval $T_{CSD}$ following the wake event is still required when leaving Sleep and Idle modes to allow the CPU to prepare for execution. Instruction execution resumes on the first clock cycle following this delay.

**REGISTER 9-11: PIE3: PERIPHERAL INTERRUPT ENABLE (FLAG) REGISTER 3**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| SSP2IE | BCL2IE | RC2IE | TX2IE | CTMUIE | TMR5GIE | TMR3GIE | TMR1GIE |
| 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 7 **SSP2IE:** Master Synchronous Serial Port 2 Interrupt Enable bit
1 = Enables the MSSP2 interrupt
0 = Disables the MSSP2 interrupt

bit 6 **BCL2IE:** Bus Collision Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 5 **RC2IE:** EUSART2 Receive Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 4 **TX2IE:** EUSART2 Transmit Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 3 **CTMUIE:** CTMU Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 2 **TMR5GIE:** TMR5 Gate Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 1 **TMR3GIE:** TMR3 Gate Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 0 **TMR1GIE:** TMR1 Gate Interrupt Enable bit
1 = Enabled
0 = Disabled

### REGISTER 10-7: ANSELE – PORTE ANALOG SELECT REGISTER

| U-0 | U-0 | U-0 | U-0 | U-0 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | ANSE2(1) | ANSE1(1) | ANSE0(1) |
| 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 7-3 **Unimplemented:** Read as ‘0’

bit 2-0 **ANSE<2:0>:** RE<2:0> Analog Select bit(1)
1 = Digital input buffer disabled
0 = Digital input buffer enabled

**Note 1:** Available on PIC18(L)F4XK22 devices only.

### REGISTER 10-8: TRISx: PORTx TRI-STATE REGISTER(1)

| R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| TRISx7 | TRISx6 | TRISx5 | TRISx4 | TRISx3 | TRISx2 | TRISx1 | TRISx0 |
| 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 7-0 **TRISx<7:0>:** PORTx Tri-State Control bit
1 = PORTx pin configured as an input (tri-stated)
0 = PORTx pin configured as an output

**Note 1:** Register description for TRISA, TRISB, TRISC and TRISD.

### REGISTER 10-9: TRISE: PORTE TRI-STATE REGISTER

| R/W-1 | U-0 | U-0 | U-0 | U-0 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| WPUE3 | — | — | — | — | TRISE2(1) | TRISE1(1) | TRISE0(1) |
| 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 7 **WPUE3:** Weak Pull-up Register bits
1 = Pull-up enabled on PORT pin
0 = Pull-up disabled on PORT pin

bit 6-3 **Unimplemented:** Read as ‘0’

bit 2-0 **TRISE<7:0>:** PORTE Tri-State Control bit(1)
1 = PORTE pin configured as an input (tri-stated)
0 = PORTE pin configured as an output

**Note 1:** Available on PIC18(L)F4XK22 devices only.

### 14.4.6 PWM STEERING MODE

In Single Output mode, PWM steering allows any of the PWM pins to be the modulated signal. Additionally, the same PWM signal can be simultaneously available on multiple pins.

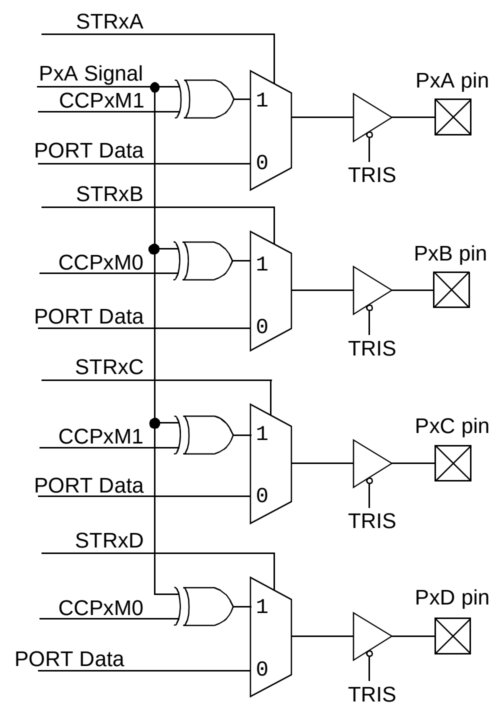
Once the Single Output mode is selected (CCPxM<3:2> = 11 and PxM<1:0> = 00 of the CCPxCON register), the user firmware can bring out the same PWM signal to one, two, three or four output pins by setting the appropriate Steering Enable bits (STRxA, STRxB, STRxC and/or STRxD) of the PSTRxCON register, as shown in Table 14-13.

| | |
|---|---|
| **Note:** | The associated TRIS bits must be set to output ('0') to enable the pin output driver in order to see the PWM signal on the pin. |

While the PWM Steering mode is active, CCPxM<1:0> bits of the CCPxCON register select the PWM output polarity for the PxD, PxC, PxB and PxA pins.

The PWM auto-shutdown operation also applies to PWM Steering mode as described in **Section 14.4.3 "Enhanced PWM Auto-shutdown Mode"**. An auto-shutdown event will only affect pins that have PWM outputs enabled.

**FIGURE 14-18: SIMPLIFIED STEERING BLOCK DIAGRAM**




**Note 1:** Port outputs are configured as shown when the CCPxCON register bits PxM<1:0> = 00 and CCPxM<3:2> = 11.

**2:** Single PWM output requires setting at least one of the STRx bits.

#### 14.4.6.1 Steering Synchronization

The STRxSYNC bit of the PSTRxCON register gives the user two selections of when the steering event will happen. When the STRxSYNC bit is '0', the steering event will happen at the end of the instruction that writes to the PSTRxCON register. In this case, the output signal at the PxA, PxB, PxC and PxD pins may be an incomplete PWM waveform. This operation is useful when the user firmware needs to immediately remove a PWM signal from the pin.

When the STRxSYNC bit is '1', the effective steering update will happen at the beginning of the next PWM period. In this case, steering on/off the PWM output will always produce a complete PWM waveform.

Figures 14-19 and 14-20 illustrate the timing diagrams of the PWM steering depending on the STRxSYNC setting.

**REGISTER 14-2: CCPxCON: ENHANCED CCPx CONTROL REGISTER**

| R/x-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| PxM<1:0> | | DCxB<1:0> | | CCPxM<3:0> | | | |
| 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 Reset |
| ‘1’ = Bit is set | ‘0’ = Bit is cleared | |

bit 7-6 **PxM<1:0>**: Enhanced PWM Output Configuration bits

If CCPxM<3:2> = `00, 01, 10`: (Capture/Compare modes)

`xx` = PxA assigned as Capture/Compare input; PxB, PxC, PxD assigned as port pins

Half-Bridge ECCP Modules[1]:

If CCPxM<3:2> = `11`: (PWM modes)

`0x` = Single output; PxA modulated; PxB assigned as port pin

`1x` = Half-Bridge output; PxA, PxB modulated with dead-band control

Full-Bridge ECCP Modules[1]:

If CCPxM<3:2> = `11`: (PWM modes)

`00` = Single output; PxA modulated; PxB, PxC, PxD assigned as port pins

`01` = Full-Bridge output forward; PxD modulated; PxA active; PxB, PxC inactive

`10` = Half-Bridge output; PxA, PxB modulated with dead-band control; PxC, PxD assigned as port pins

`11` = Full-Bridge output reverse; PxB modulated; PxC active; PxA, PxD inactive

bit 5-4 **DCxB<1:0>:** PWM Duty Cycle Least Significant bits

Capture mode:
Unused

Compare mode:
Unused

PWM mode:
These bits are the two LSbs of the PWM duty cycle. The eight MSbs are found in CCPRxL.

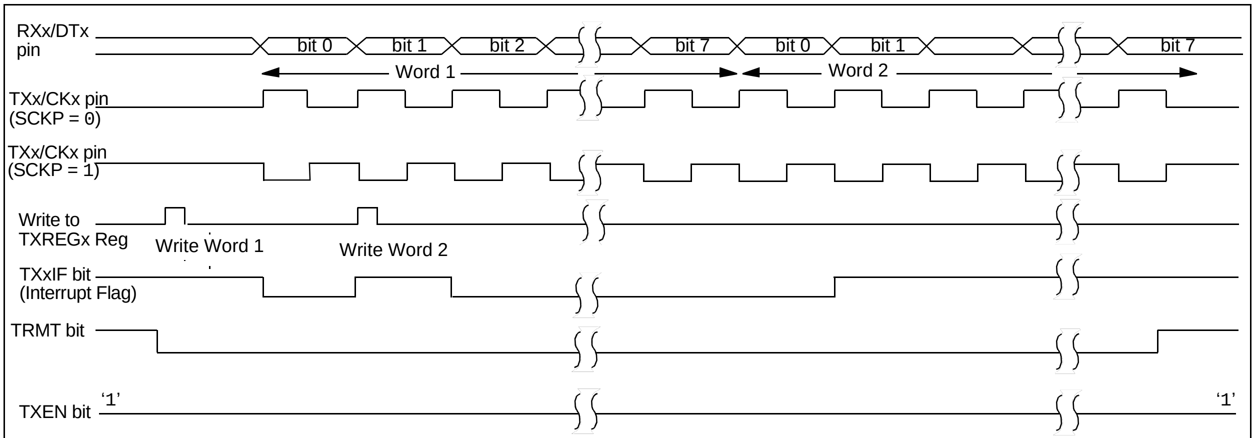
**Note 1:** See Table 14-1 to determine full-bridge and half-bridge ECCPs for the device being used.

#### 16.5.1.5 Synchronous Master Transmission Setup:

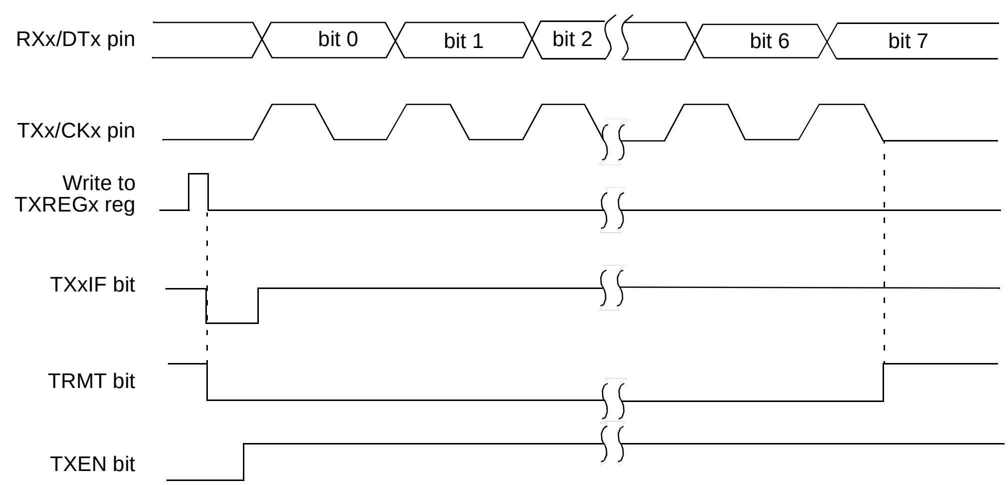
1. Initialize the SPBRGHx, SPBRGx register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 16.4 “EUSART Baud Rate Generator (BRG)”**).
2. Set the RXx/DTx and TXx/CKx TRIS controls to ‘1’.
3. Enable the synchronous master serial port by setting bits SYNC, SPEN and CSRC. Set the TRIS bits corresponding to the RXx/DTx and TXx/CKx I/O pins.
4. Disable Receive mode by clearing bits SREN and CREN.
5. Enable Transmit mode by setting the TXEN bit.
6. If 9-bit transmission is desired, set the TX9 bit.
7. If interrupts are desired, set the TXxIE, GIE/GIEH and PEIE/GIEL interrupt enable bits.
8. If 9-bit transmission is selected, the ninth bit should be loaded in the TX9D bit.
9. Start transmission by loading data to the TXREGx register.

**FIGURE 16-10: SYNCHRONOUS TRANSMISSION**




**Note:** Sync Master mode, SPBRGx = 0, continuous transmission of two 8-bit words.

**FIGURE 16-11: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)**



### 17.2.2 COMPLETION OF A CONVERSION

When the conversion is complete, the ADC module will:

- Clear the GO/$\overline{\text{DONE}}$ bit
- Set the ADIF flag bit
- Update the ADRESH:ADRESL registers with new conversion result

### 17.2.3 DISCHARGE

The discharge phase is used to initialize the value of the capacitor array. The array is discharged after every sample. This feature helps to optimize the unity-gain amplifier, as the circuit always needs to charge the capacitor array, rather than charge/discharge based on previous measure values.

### 17.2.4 TERMINATING A CONVERSION

If a conversion must be terminated before completion, the GO/$\overline{\text{DONE}}$ bit can be cleared by software. The ADRESH:ADRESL registers will not be updated with the partially complete Analog-to-Digital conversion sample. Instead, the ADRESH:ADRESL register pair will retain the value of the previous conversion.

| | |
|---|---|
| **Note:** | A device Reset forces all registers to their Reset state. Thus, the ADC module is turned off and any pending conversion is terminated. |

### 17.2.5 DELAY BETWEEN CONVERSIONS

After the A/D conversion is completed or aborted, a 2 $T_{AD}$ wait is required before the next acquisition can be started. After this wait, the currently selected channel is reconnected to the charge holding capacitor commencing the next acquisition.

### 17.2.6 ADC OPERATION IN POWER-MANAGED MODES

The selection of the automatic acquisition time and A/D conversion clock is determined in part by the clock source and frequency while in a power-managed mode.

If the A/D is expected to operate while the device is in a power-managed mode, the ACQT<2:0> and ADCS<2:0> bits in ADCON2 should be updated in accordance with the clock source to be used in that mode. After entering the mode, an A/D acquisition or conversion may be started. Once started, the device should continue to be clocked by the same clock source until the conversion has been completed.

If desired, the device may be placed into the corresponding Idle mode during the conversion. If the device clock frequency is less than 1 MHz, the A/D FRC clock source should be selected.

### 17.2.7 ADC OPERATION DURING SLEEP

The ADC module can operate during Sleep. This requires the ADC clock source to be set to the FRC option. When the FRC clock source is selected, the ADC waits one additional instruction before starting the conversion. This allows the `SLEEP` instruction to be executed, which can reduce system noise during the conversion. If the ADC interrupt is enabled, the device will wake-up from Sleep when the conversion completes. If the ADC interrupt is disabled, the ADC module is turned off after the conversion completes, although the ADON bit remains set.

When the ADC clock source is something other than FRC, a `SLEEP` instruction causes the present conversion to be aborted and the ADC module is turned off, although the ADON bit remains set.

### 17.2.8 SPECIAL EVENT TRIGGER

Two Special Event Triggers are available to start an A/D conversion: CTMU and CCP5. The Special Event Trigger source is selected using the TRIGSEL bit in ADCON1.

When TRIGSEL = `0`, the CCP5 module is selected as the Special Event Trigger source. To enable the Special Event Trigger in the CCP module, set CCP5M<3:0> = `1011`, in the CCP5CON register.

When TRIGSEL = `1`, the CTMU module is selected. The CTMU module requires that the CTTRIG bit in CTMUCONH is set to enable the Special Event Trigger.

In addition to TRIGSEL bit, the following steps are required to start an A/D conversion:

- The A/D module must be enabled (ADON = `1`)
- The appropriate analog input channel selected
- The minimum acquisition period set one of these ways:
  - Timing provided by the user
  - Selection made of an appropriate $T_{ACQ}$ time

With these conditions met, the trigger sets the GO/$\overline{\text{DONE}}$ bit and the A/D acquisition starts.

If the A/D module is not enabled (ADON = `0`), the module ignores the Special Event Trigger.

### 17.2.9 PERIPHERAL MODULE DISABLE

When a peripheral module is not used or inactive, the module can be disabled by setting the Module Disable bit in the PMD registers. This will reduce power consumption to an absolute minimum. Setting the PMD bits holds the module in Reset and disconnects the module's clock source. The Module Disable bit for the ADC module is ADCMD in the PMD2 Register. See **Section 3.0 "Power-Managed Modes"** for more information.

## 25.0 INSTRUCTION SET SUMMARY

PIC18(L)F2X/4XK22 devices incorporate the standard set of 75 PIC18 core instructions, as well as an extended set of eight new instructions, for the optimization of code that is recursive or that utilizes a software stack. The extended set is discussed later in this section.

### 25.1 Standard Instruction Set

The standard PIC18 instruction set adds many enhancements to the previous PIC® MCU instruction sets, while maintaining an easy migration from these PIC® MCU instruction sets. Most instructions are a single program memory word (16 bits), but there are four instructions that require two program memory locations.

Each single-word instruction is a 16-bit word divided into an opcode, which specifies the instruction type and one or more operands, which further specify the operation of the instruction.

The instruction set is highly orthogonal and is grouped into four basic categories:

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal** operations
- **Control** operations

The PIC18 instruction set summary in Table 25-2 lists **byte-oriented**, **bit-oriented**, **literal** and **control** operations. Table 25-1 shows the opcode field descriptions.

Most **byte-oriented** instructions have three operands:

1. The file register (specified by ‘f’)
2. The destination of the result (specified by ‘d’)
3. The accessed memory (specified by ‘a’)

The file register designator ‘f’ specifies which file register is to be used by the instruction. The destination designator ‘d’ specifies where the result of the operation is to be placed. If ‘d’ is zero, the result is placed in the WREG register. If ‘d’ is one, the result is placed in the file register specified in the instruction.

All **bit-oriented** instructions have three operands:

1. The file register (specified by ‘f’)
2. The bit in the file register (specified by ‘b’)
3. The accessed memory (specified by ‘a’)

The bit field designator ‘b’ selects the number of the bit affected by the operation, while the file register designator ‘f’ represents the number of the file in which the bit is located.

The **literal** instructions may use some of the following operands:

- A literal value to be loaded into a file register (specified by ‘k’)
- The desired FSR register to load the literal value into (specified by ‘f’)
- No operand required (specified by ‘—’)

The **control** instructions may use some of the following operands:

- A program memory address (specified by ‘n’)
- The mode of the `CALL` or `RETURN` instructions (specified by ‘s’)
- The mode of the table read and table write instructions (specified by ‘m’)
- No operand required (specified by ‘—’)

All instructions are a single word, except for four double-word instructions. These instructions were made double-word to contain the required information in 32 bits. In the second word, the four MSbs are ‘1’s. If this second word is executed as an instruction (by itself), it will execute as a `NOP`.

All single-word instructions are executed in a single instruction cycle, unless a conditional test is true or the program counter is changed as a result of the instruction. In these cases, the execution takes two instruction cycles, with the additional instruction cycle(s) executed as a `NOP`.

The double-word instructions execute in two instruction cycles.

One instruction cycle consists of four oscillator periods. Thus, for an oscillator frequency of 4 MHz, the normal instruction execution time is 1 μs. If a conditional test is true, or the program counter is changed as a result of an instruction, the instruction execution time is 2 μs. Two-word branch instructions (if true) would take 3 μs.

Figure 25-1 shows the general formats that the instructions can have. All examples use the convention ‘nnh’ to represent a hexadecimal number.

The Instruction Set Summary, shown in Table 25-2, lists the standard instructions recognized by the Microchip Assembler (MPASM™).

**Section 25.1.1 “Standard Instruction Set”** provides a description of each instruction.

**FIGURE 27-15: EXAMPLE SPI SLAVE MODE TIMING (CKE = 0)**

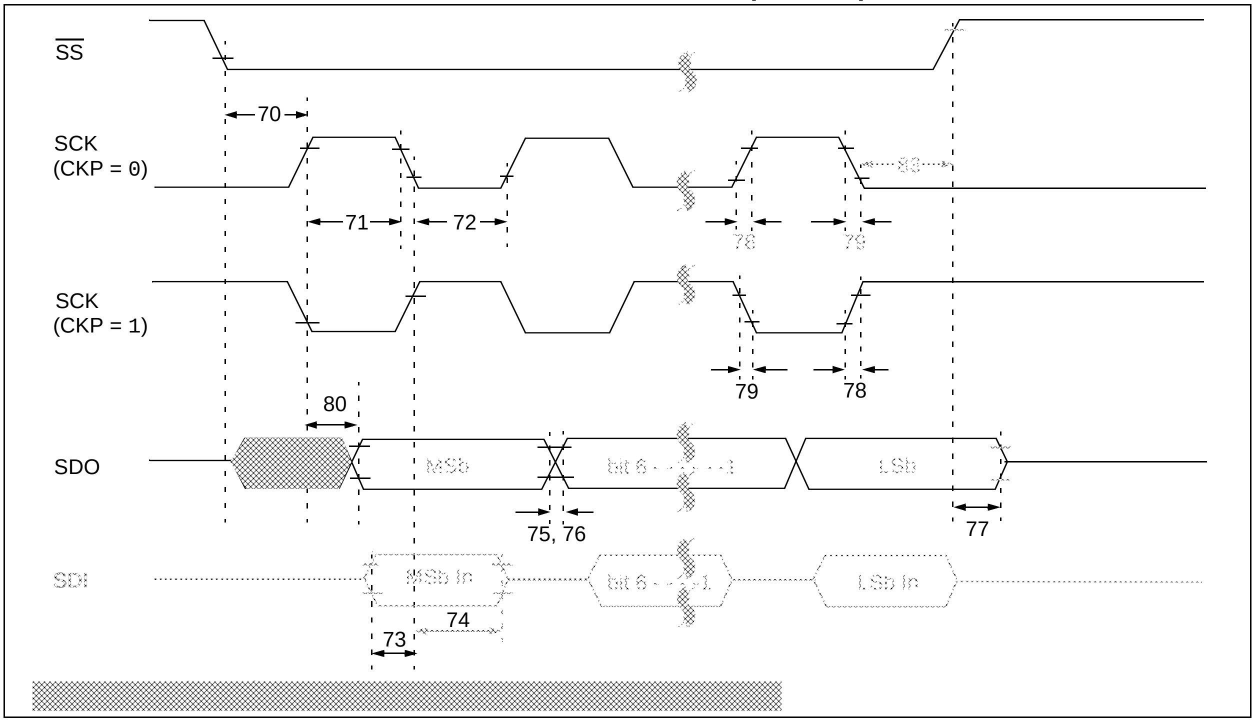



**FIGURE 27-16: EXAMPLE SPI SLAVE MODE TIMING (CKE = 1)**

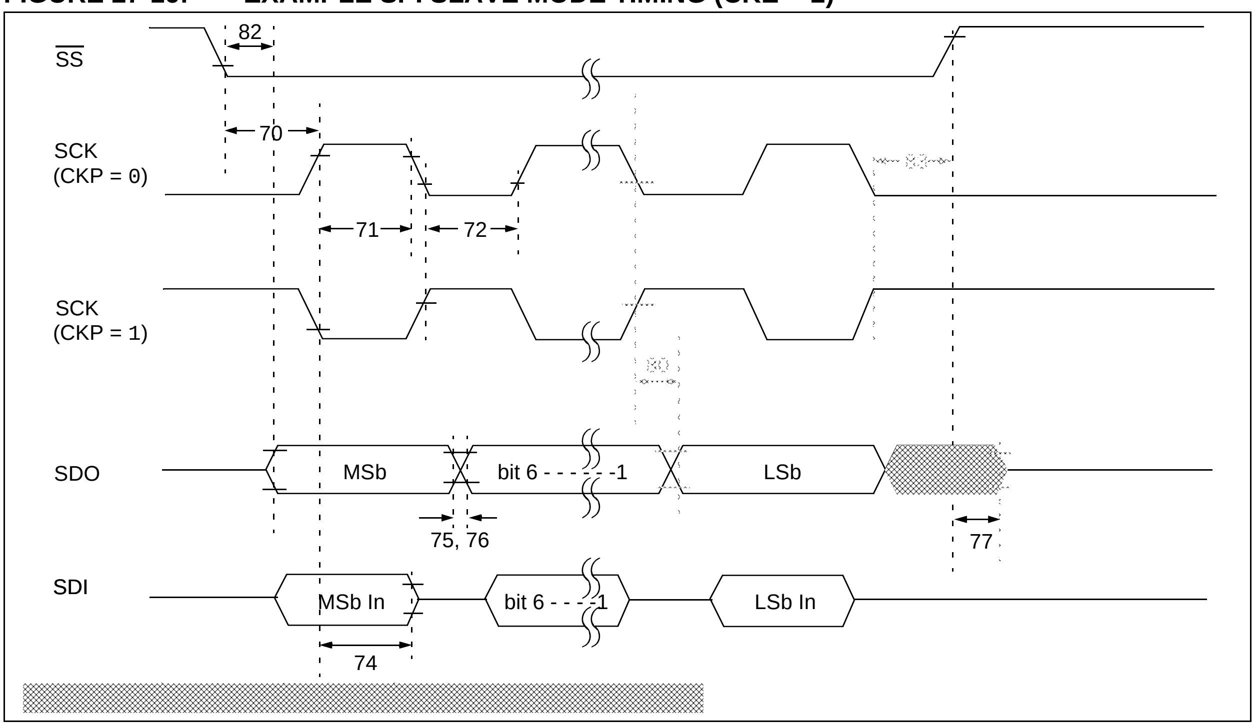


**FIGURE 28-34: PIC18LF2X/4XK22 TYPICAL IDD: RC_IDLE LF-INTOSC 31 kHz**

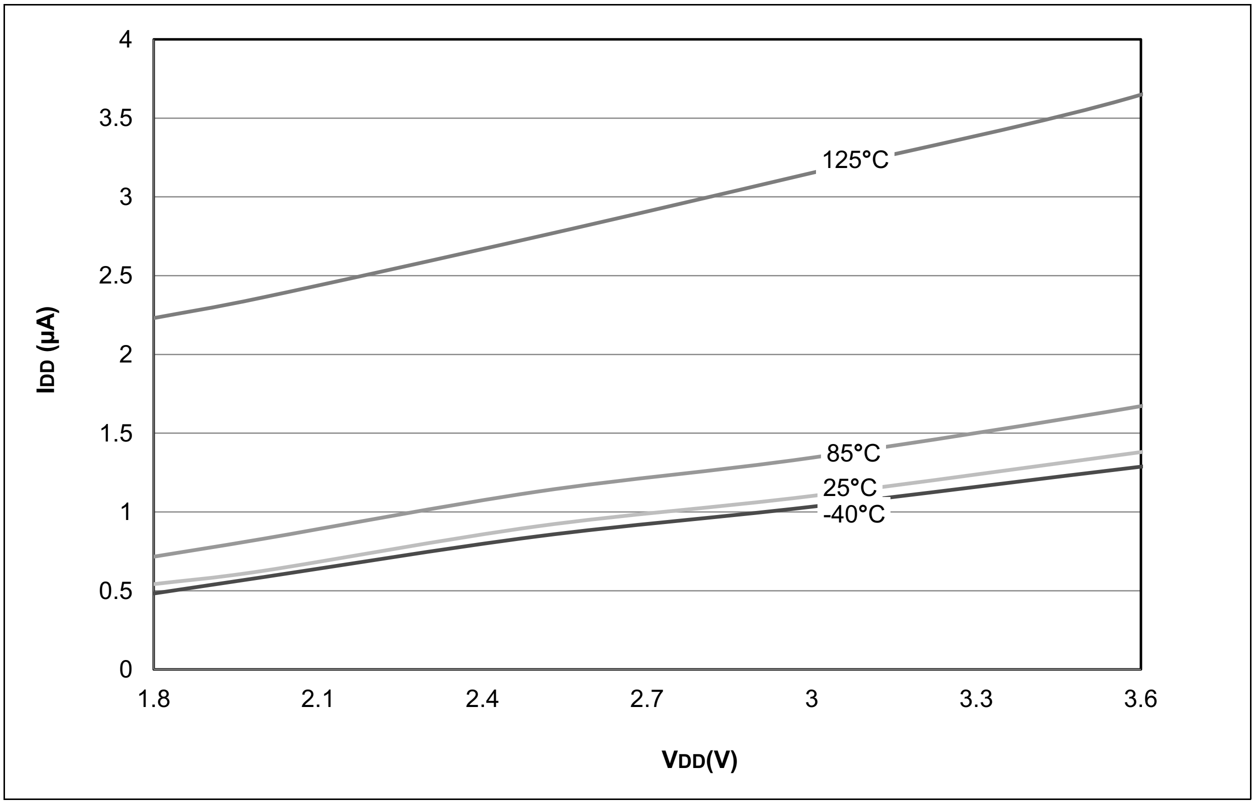



**FIGURE 28-35: PIC18LF2X/4XK22 MAXIMUM IDD: RC_IDLE LF-INTOSC 31 kHz**

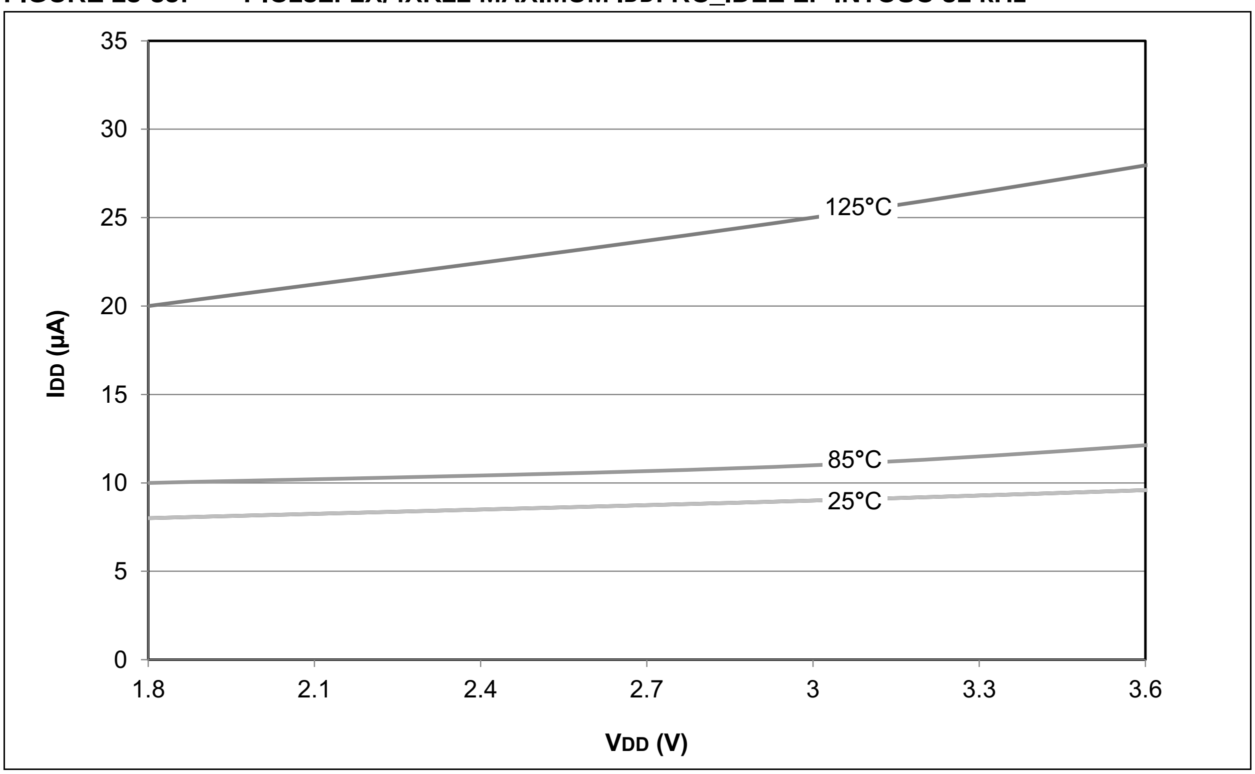


**FIGURE 28-58: PIC18F2X/4XK22 TYPICAL IDD: PRI_RUN EC with PLL**

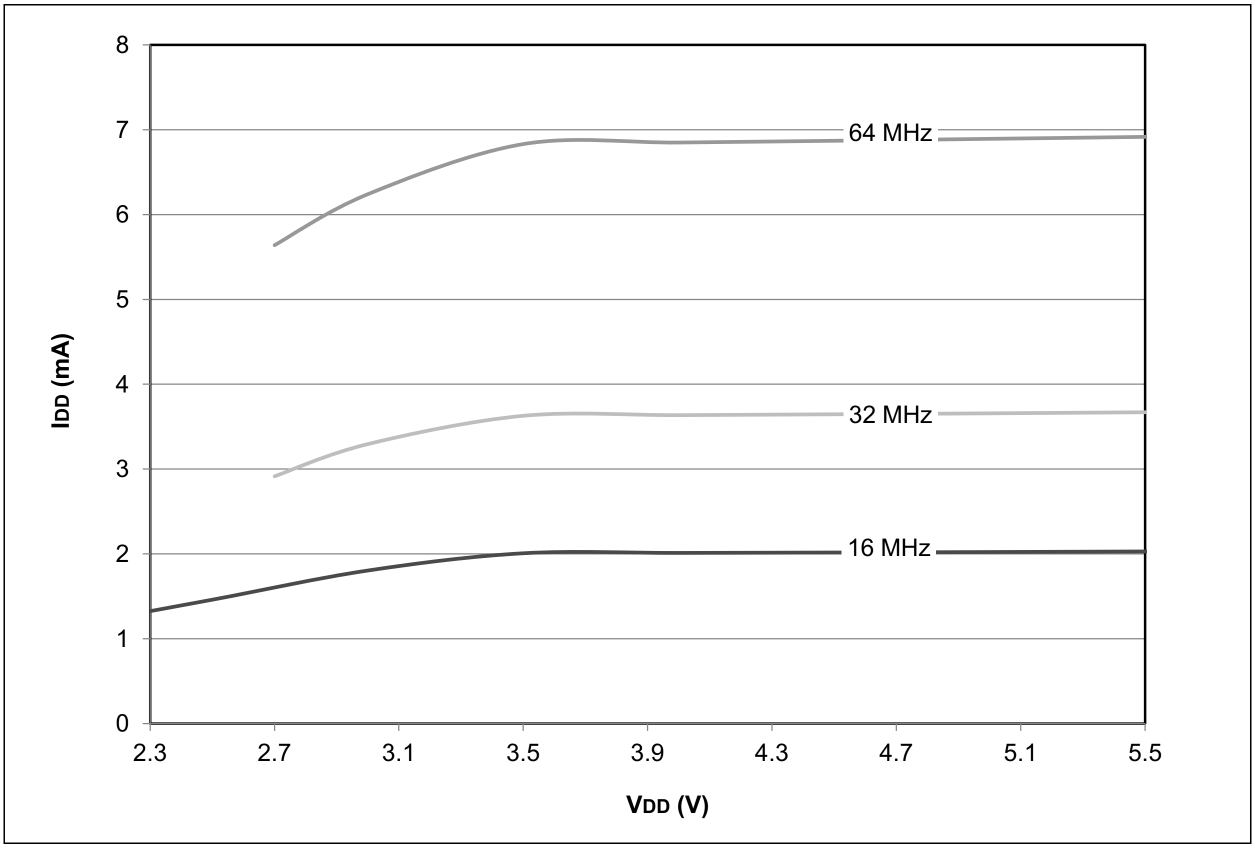



**FIGURE 28-59: PIC18F2X/4XK22 MAXIMUM IDD: PRI_RUN EC with PLL**

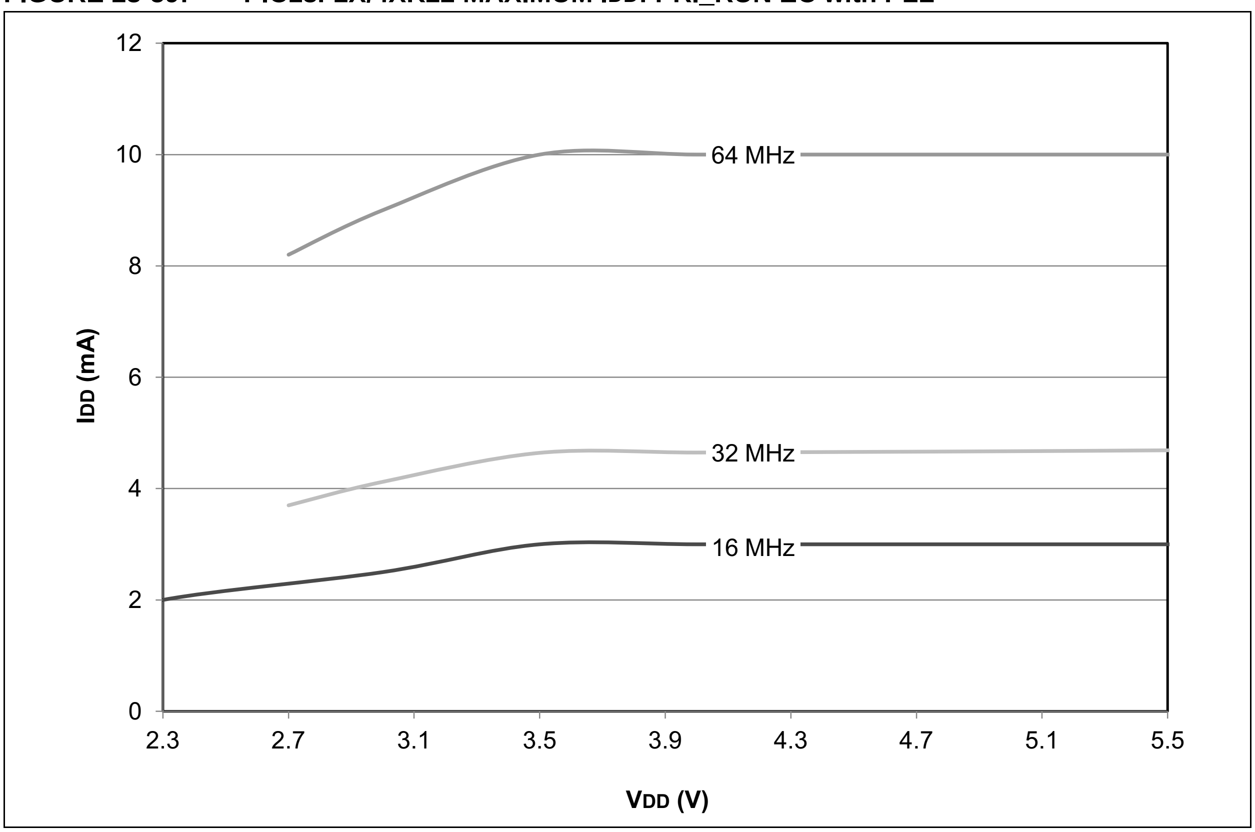


**FIGURE 28-76: PIC18LF2X/4XK22 TYPICAL IDD: SEC_IDLE 32.768 kHz**

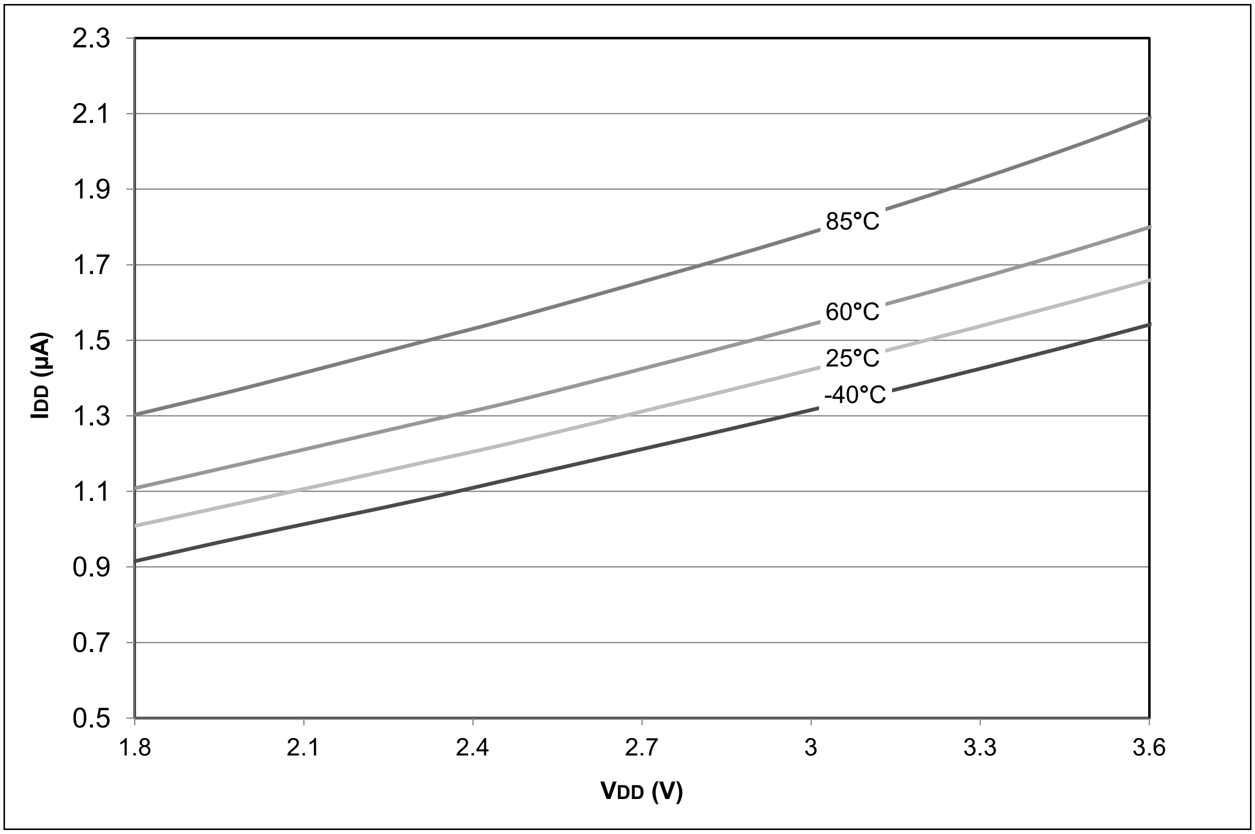



**FIGURE 28-77: PIC18LF2X/4XK22 MAXIMUM IDD: SEC_IDLE 32.768 kHz**

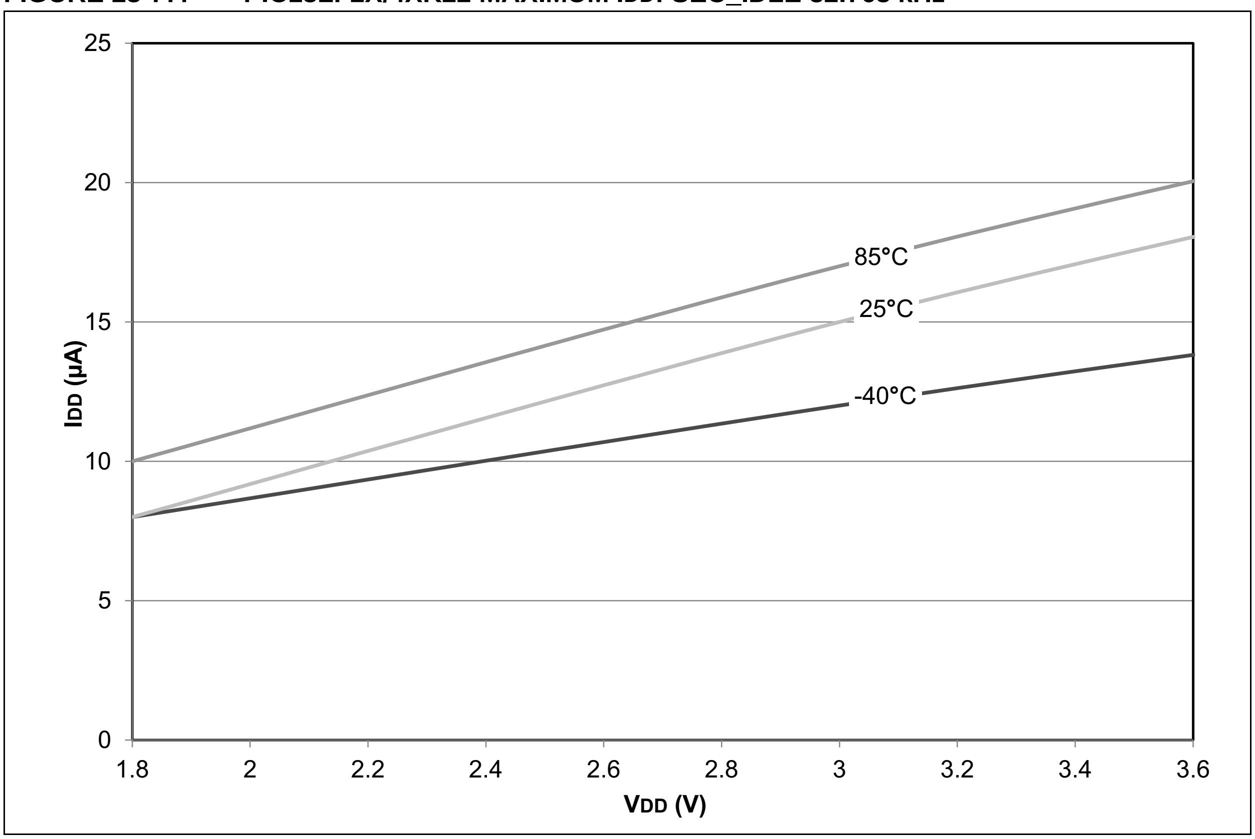


FIGURE 28-98: PIC18F2X/4XK22 TYPICAL FIXED VOLTAGE REFERENCE 4x OUTPUT

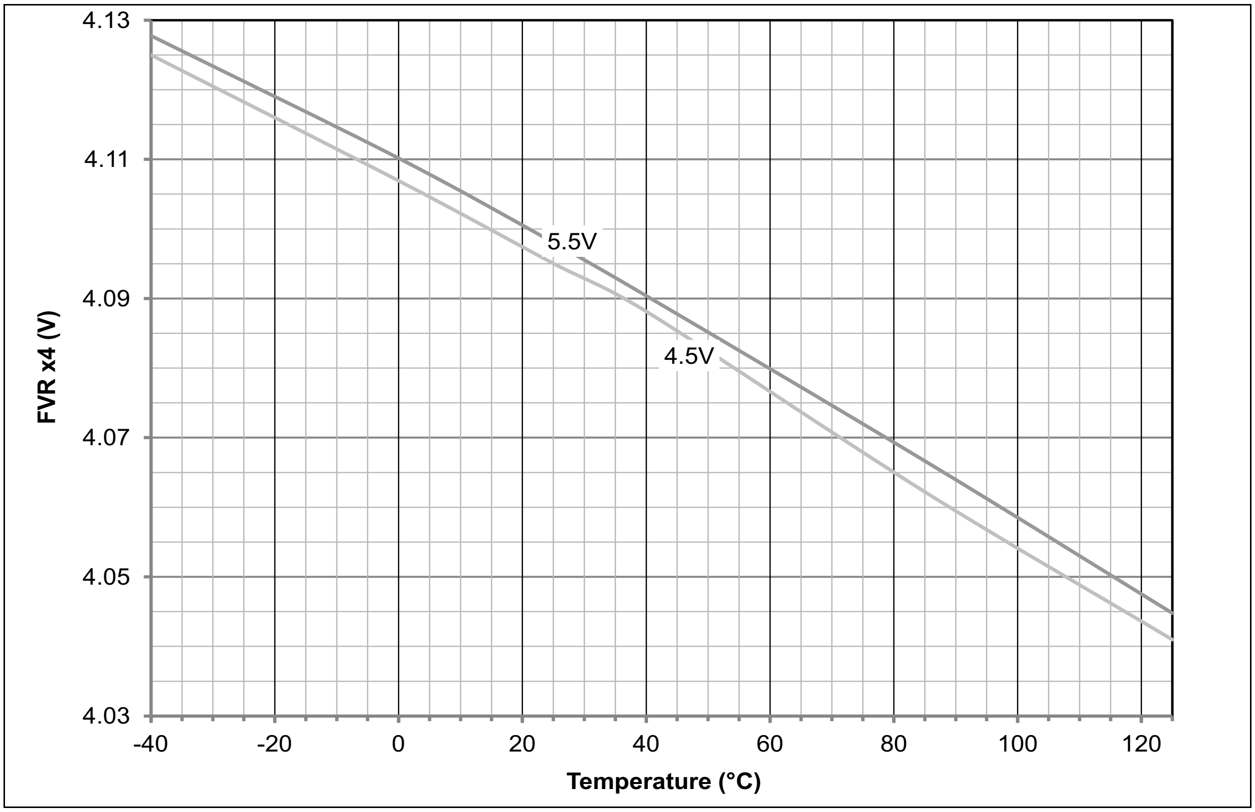
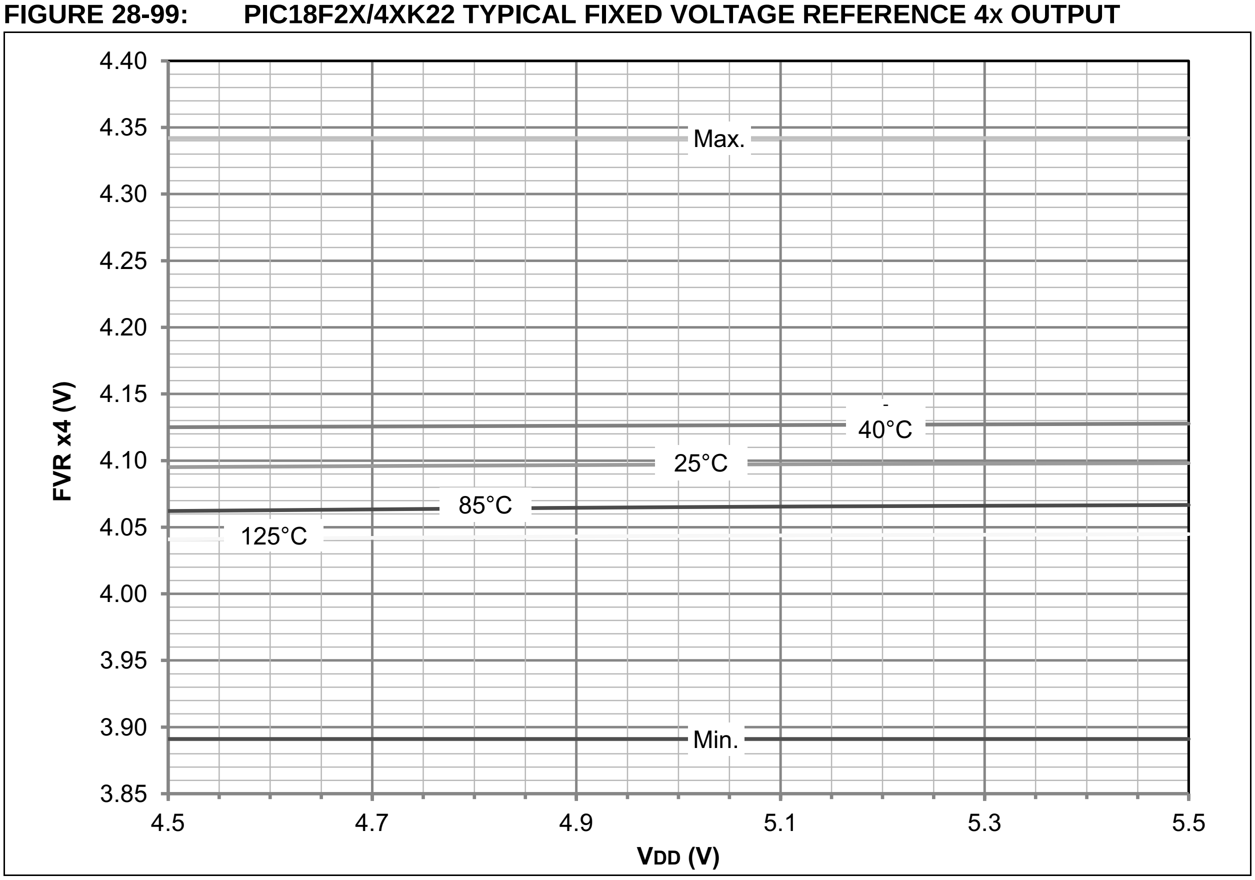
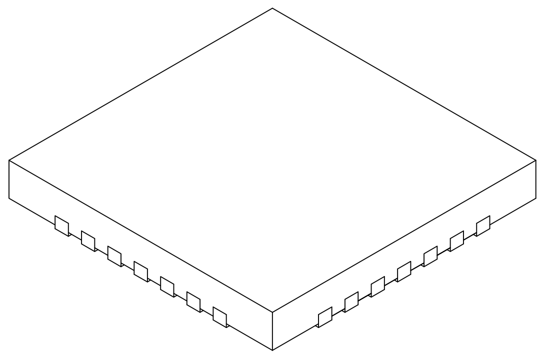



FIGURE 28-99: PIC18F2X/4XK22 TYPICAL FIXED VOLTAGE REFERENCE 4x OUTPUT



## 28-Lead Plastic Ultra Thin Quad Flat, No Lead Package (MV) – 4x4x0.5 mm Body [UQFN]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Pins | N | | 28 | |
| Pitch | e | | 0.40 BSC | |
| Overall Height | A | 0.45 | 0.50 | 0.55 |
| Standoff | A1 | 0.00 | 0.02 | 0.05 |
| Contact Thickness | A3 | | 0.127 REF | |
| Overall Width | E | | 4.00 BSC | |
| Exposed Pad Width | E2 | 2.55 | 2.65 | 2.75 |
| Overall Length | D | | 4.00 BSC | |
| Exposed Pad Length | D2 | 2.55 | 2.65 | 2.75 |
| Contact Width | b | 0.15 | 0.20 | 0.25 |
| Contact Length | L | 0.30 | 0.40 | 0.50 |
| Contact-to-Exposed Pad | K | 0.20 | - | - |

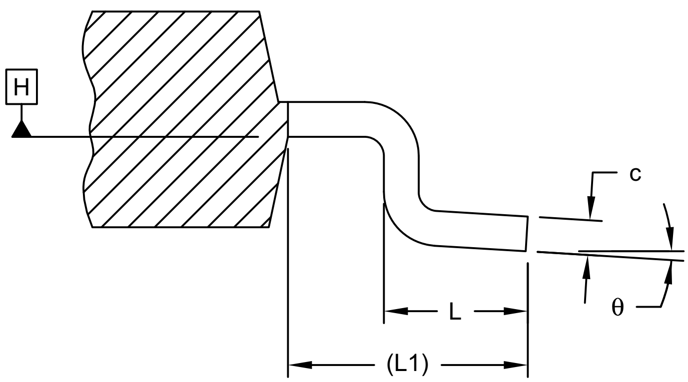
**Notes:**

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

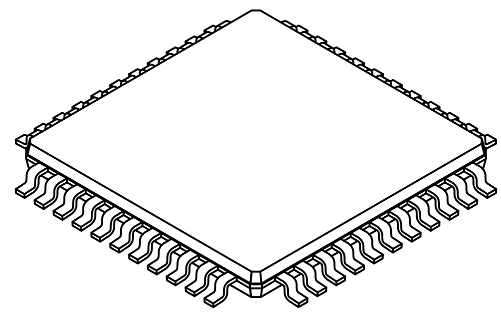
Microchip Technology Drawing C04-152A Sheet 2 of 2

## 44-Lead Plastic Thin Quad Flatpack (PT) - 10x10x1.0 mm Body [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




SECTION A-A



| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Leads | N | | 44 | |
| Lead Pitch | e | | 0.80 BSC | |
| Overall Height | A | - | - | 1.20 |
| Standoff | A1 | 0.05 | - | 0.15 |
| Molded Package Thickness | A2 | 0.95 | 1.00 | 1.05 |
| Overall Width | E | | 12.00 BSC | |
| Molded Package Width | E1 | | 10.00 BSC | |
| Overall Length | D | | 12.00 BSC | |
| Molded Package Length | D1 | | 10.00 BSC | |
| Lead Width | b | 0.30 | 0.37 | 0.45 |
| Lead Thickness | c | 0.09 | - | 0.20 |
| Lead Length | L | 0.45 | 0.60 | 0.75 |
| Footprint | L1 | | 1.00 REF | |
| Foot Angle | θ | 0° | 3.5° | 7° |

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Exact shape of each corner is optional.
3. Dimensioning and tolerancing per ASME Y14.5M

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-076C Sheet 2 of 2