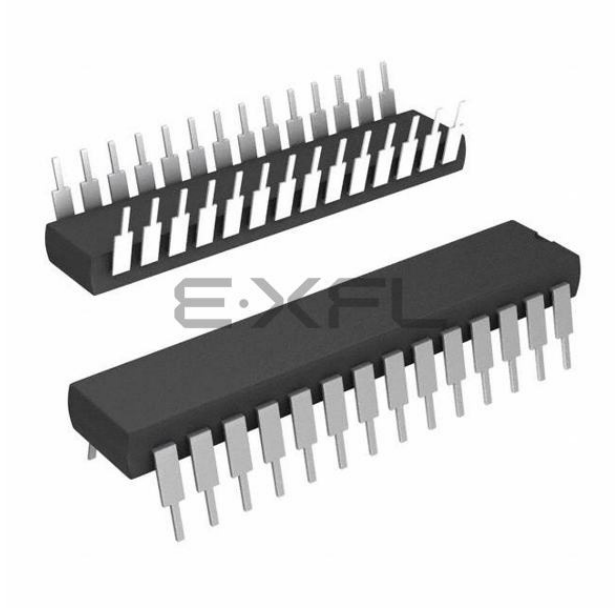


**Welcome to E-XFL.COM**

## 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 | 24 |
| Program Memory Size | 32KB (16K x 16) |
| Program Memory Type | FLASH |
| EEPROM Size | 256 x 8 |
| RAM Size | 1.5K x 8 |
| Voltage - Supply (Vcc/Vdd) | 1.8V ~ 3.6V |
| Data Converters | A/D 19x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 125°C (TA) |
| Mounting Type | Through Hole |
| Package / Case | 28-DIP (0.300", 7.62mm) |
| Supplier Device Package | 28-SPDIP |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic18lf25k22-e-sp |

## 4.2 Register Definitions: Reset Control

**REGISTER 4-1: RCON: RESET CONTROL REGISTER**

| R/W-0/0 | R/W-q/u | U-0 | R/W-1/q | R-1/q | R-1/q | R/W-q/u | R/W-0/q |
|---|---|---|---|---|---|---|---|
| IPEN | SBOREN(1) | — | $\overline{RI}$ | $\overline{TO}$ | $\overline{PD}$ | $\overline{POR}$(2) | $\overline{BOR}$ |
| bit 7 | | | | | | | bit 0 |

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

bit 7 **IPEN:** Interrupt Priority Enable bit
1 = Enable priority levels on interrupts
0 = Disable priority levels on interrupts (PIC16CXXX Compatibility mode)

bit 6 **SBOREN:** BOR Software Enable bit(1)
If BOREN<1:0> = 01:
1 = BOR is enabled
0 = BOR is disabled
If BOREN<1:0> = 00, 10 or 11:
Bit is disabled and read as '0'.

bit 5 **Unimplemented:** Read as '0'

bit 4 $\overline{\textbf{RI}}$**:** RESET Instruction Flag bit
1 = The RESET instruction was not executed (set by firmware or Power-on Reset)
0 = The RESET instruction was executed causing a device Reset (must be set in firmware after a code-executed Reset occurs)

bit 3 $\overline{\textbf{TO}}$**:** Watchdog Time-out Flag bit
1 = Set by power-up, CLRWDT instruction or SLEEP instruction
0 = A WDT time-out occurred

bit 2 $\overline{\textbf{PD}}$**:** Power-down Detection Flag bit
1 = Set by power-up or by the CLRWDT instruction
0 = Set by execution of the SLEEP instruction

bit 1 $\overline{\textbf{POR}}$**:** Power-on Reset Status bit(2)
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0 $\overline{\textbf{BOR}}$**:** Brown-out Reset Status bit(3)
1 = A Brown-out Reset has not occurred (set by firmware only)
0 = A Brown-out Reset occurred (must be set by firmware after a POR or Brown-out Reset occurs)

**Note 1:** When CONFIG2L[2:1] = 01, then the SBOREN Reset state is '1'; otherwise, it is '0'.
**2:** The actual Reset value of $\overline{POR}$ is determined by the type of device Reset. See the notes following this register and **Section 4.7 "Reset State of Registers"** for additional information.
**3:** See Table 4-1.

**Note 1:** Brown-out Reset is indicated when $\overline{BOR}$ is '0' and POR is '1' (assuming that both $\overline{POR}$ and $\overline{BOR}$ were set to '1' by firmware immediately after POR).
**2:** It is recommended that the $\overline{POR}$ bit be set after a Power-on Reset has been detected so that subsequent Power-on Resets may be detected.

**TABLE 5-2: REGISTER FILE SUMMARY FOR PIC18(L)F2X/4XK22 DEVICES (CONTINUED)**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR |
|---|---|---|---|---|---|---|---|---|---|---|
| F3Ah | ANSELC | ANSC7 | ANSC6 | ANSC5 | ANSC4 | ANSC3 | ANSC2 | — | — | 1111 11-- |
| F39h | ANSELB | — | — | ANSB5 | ANSB4 | ANSB3 | ANSB2 | ANSB1 | ANSB0 | --11 1111 |
| F38h | ANSELA | — | — | ANSA5 | — | ANSA3 | ANSA2 | ANSA1 | ANSA0 | --1- 1111 |

**Legend:** x = unknown, u = unchanged, — = unimplemented, q = value depends on condition

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

**2:** PIC18(L)F2XK22 devices only.

**3:** PIC18(L)F23/24K22 and PIC18(L)F43/44K22 devices only.

**4:** PIC18(L)F26K22 and PIC18(L)F46K22 devices only.

## 6.6 Writing to Flash Program Memory

The programming block size is 64 bytes. Word or byte programming is not supported.

Table writes are used internally to load the holding registers needed to program the Flash memory. There are only as many holding registers as there are bytes in a write block (64 bytes).

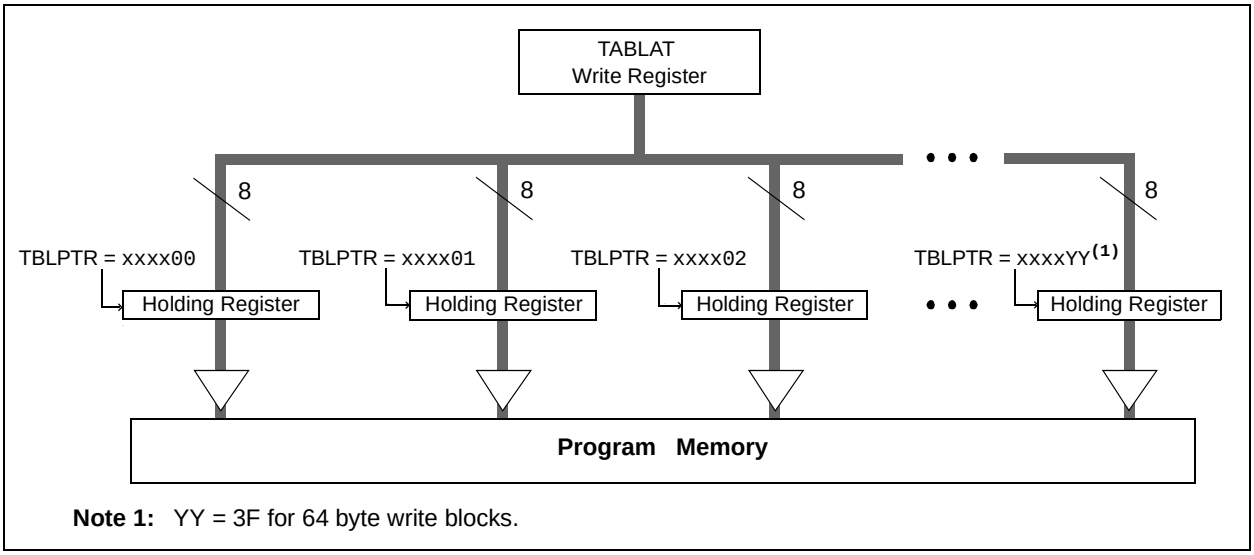
Since the Table Latch (TABLAT) is only a single byte, the TBLWT instruction needs to be executed 64 times for each programming operation. All of the table write operations will essentially be short writes because only the holding registers are written. After all the holding registers have been written, the programming operation of that block of memory is started by configuring the EECON1 register for a program memory write and performing the long write sequence.

The long write is necessary for programming the internal Flash. Instruction execution is halted during a long write cycle. The long write will be terminated by the internal programming timer.

The EEPROM on-chip timer controls the write time. The write/erase voltages are generated by an on-chip charge pump, rated to operate over the voltage range of the device.

| | |
|---|---|
| **Note:** | The default value of the holding registers on device Resets and after write operations is FFh. A write of FFh to a holding register does not modify that byte. This means that individual bytes of program memory may be modified, provided that the change does not attempt to change any bit from a '0' to a '1'. When modifying individual bytes, it is not necessary to load all holding registers before executing a long write operation. |

**FIGURE 6-5: TABLE WRITES TO FLASH PROGRAM MEMORY**




**Note 1:** YY = 3F for 64 byte write blocks.

### 6.6.1 FLASH PROGRAM MEMORY WRITE SEQUENCE

The sequence of events for programming an internal program memory location should be:

1. Read 64 bytes into RAM.
2. Update data values in RAM as necessary.
3. Load Table Pointer register with address being erased.
4. Execute the block erase procedure.
5. Load Table Pointer register with address of first byte being written.
6. Write the 64-byte block into the holding registers with auto-increment.
7. Set the EECON1 register for the write operation:
   - set EEPGD bit to point to program memory;
   - clear the CFGS bit to access program memory;
   - set WREN to enable byte writes.
8. Disable interrupts.
9. Write 55h to EECON2.
10. Write 0AAh to EECON2.
11. Set the WR bit. This will begin the write cycle.
12. The CPU will stall for duration of the write (about 2 ms using internal timer).
13. Re-enable interrupts.
14. Verify the memory (table read).

This procedure will require about 6 ms to update each write block of memory. An example of the required code is given in Example 6-3.

| | |
|---|---|
| **Note:** | Before setting the WR bit, the Table Pointer address needs to be within the intended address range of the bytes in the holding registers. |

**FIGURE 12-5: TIMER1/3/5 GATE TOGGLE MODE**

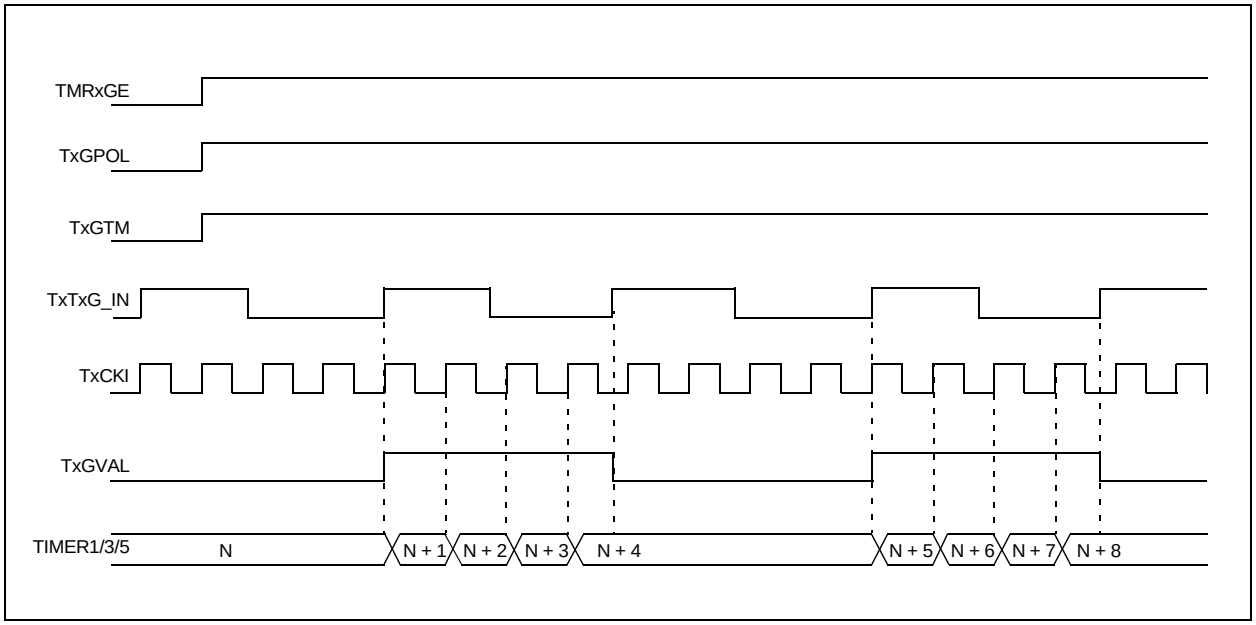



**FIGURE 12-6: TIMER1/3/5 GATE SINGLE-PULSE MODE**

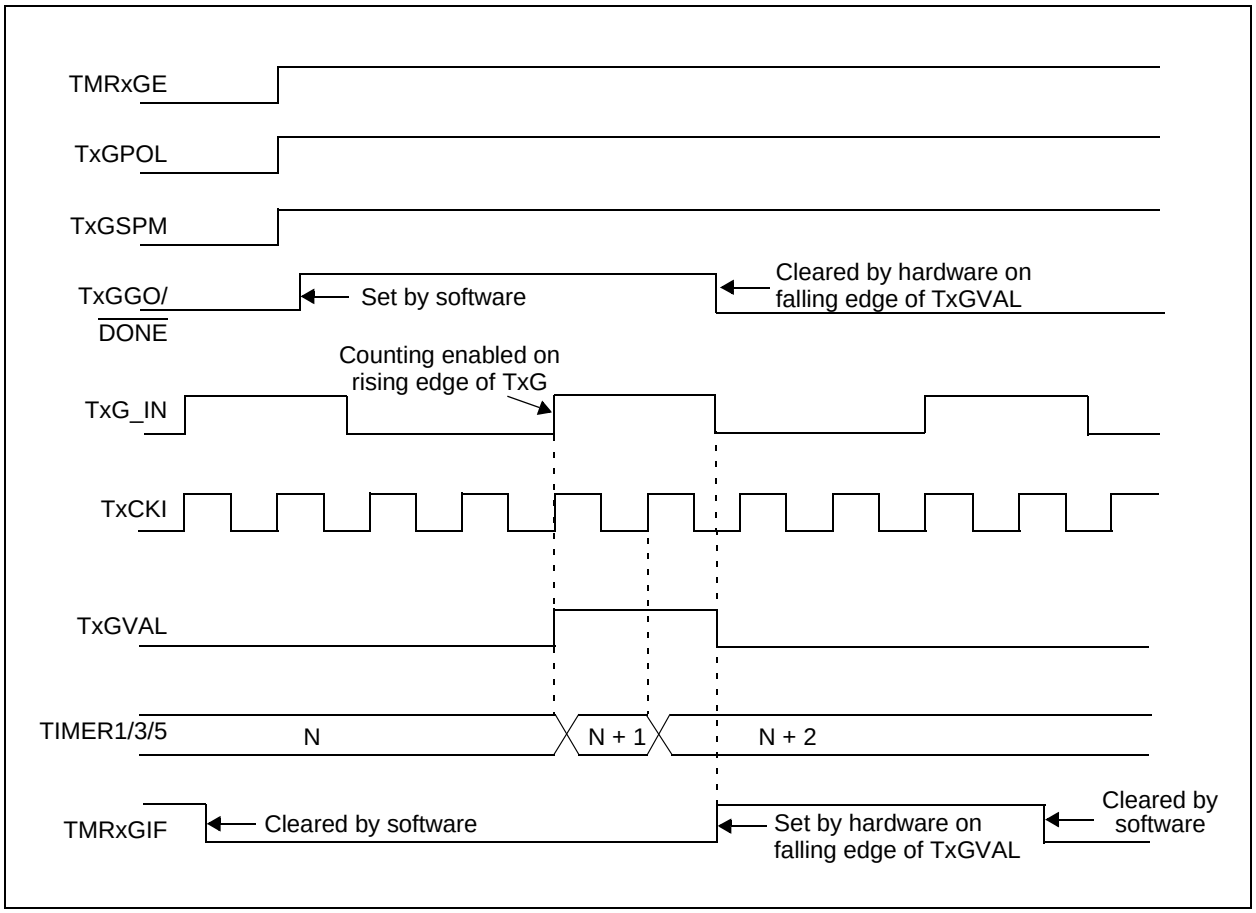


**TABLE 14-5: REGISTERS ASSOCIATED WITH COMPARE (CONTINUED)**

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---|---|---|---|---|---|---|---|---|---|
| IPR2 | OSCFIP | C1IP | C2IP | EEIP | BCL1IP | HLVDIP | TMR3IP | CCP2IP | 122 |
| IPR4 | — | — | — | — | — | CCP5IP | CCP4IP | CCP3IP | 124 |
| PIE1 | — | ADIE | RC1IE | TX1IE | SSP1IE | CCP1IE | TMR2IE | TMR1IE | 117 |
| PIE2 | OSCFIE | C1IE | C2IE | EEIE | BCL1IE | HLVDIE | TMR3IE | CCP2IE | 118 |
| PIE4 | — | — | — | — | — | CCP5IE | CCP4IE | CCP3IE | 120 |
| PIR1 | — | ADIF | RC1IF | TX1IF | SSP1IF | CCP1IF | TMR2IF | TMR1IF | 112 |
| PIR2 | OSCFIF | C1IF | C2IF | EEIF | BCL1IF | HLVDIF | TMR3IF | CCP2IF | 113 |
| PIR4 | — | — | — | — | — | CCP5IF | CCP4IF | CCP3IF | 115 |
| PMD0 | UART2MD | UART1MD | TMR6MD | TMR5MD | TMR4MD | TMR3MD | TMR2MD | TMR1MD | 52 |
| PMD1 | MSSP2MD | MSSP1MD | — | CCP5MD | CCP4MD | CCP3MD | CCP2MD | CCP1MD | 53 |
| T1CON | TMR1CS<1:0> | | T1CKPS<1:0> | | T1SOSCEN | $\overline{\text{T1SYNC}}$ | T1RD16 | TMR1ON | 166 |
| T1GCON | TMR1GE | T1GPOL | T1GTM | T1GSPM | T1GGO/$\overline{\text{DONE}}$ | T1GVAL | T1GSS<1:0> | | 167 |
| T3CON | TMR3CS<1:0> | | T3CKPS<1:0> | | T3SOSCEN | $\overline{\text{T3SYNC}}$ | T3RD16 | TMR3ON | 166 |
| T3GCON | TMR3GE | T3GPOL | T3GTM | T3GSPM | T3GGO/$\overline{\text{DONE}}$ | T3GVAL | T3GSS<1:0> | | 167 |
| T5CON | TMR5CS<1:0> | | T5CKPS<1:0> | | T5SOSCEN | $\overline{\text{T5SYNC}}$ | T5RD16 | TMR5ON | 166 |
| T5GCON | TMR5GE | T5GPOL | T5GTM | T5GSPM | T5GGO/$\overline{\text{DONE}}$ | T5GVAL | T5GSS<1:0> | | 167 |
| TMR1H | Holding Register for the Most Significant Byte of the 16-bit TMR1 Register | | | | | | | | — |
| TMR1L | Least Significant Byte of the 16-bit TMR1 Register | | | | | | | | — |
| TMR3H | Holding Register for the Most Significant Byte of the 16-bit TMR3 Register | | | | | | | | — |
| TMR3L | Least Significant Byte of the 16-bit TMR3 Register | | | | | | | | — |
| TMR5H | Holding Register for the Most Significant Byte of the 16-bit TMR5 Register | | | | | | | | — |
| TMR5L | Least Significant Byte of the 16-bit TMR5 Register | | | | | | | | — |
| TRISA | TRISA7 | TRISA6 | TRISA5 | TRISA4 | TRISA3 | TRISA2 | TRISA1 | TRISA0 | 151 |
| TRISB | TRISB7 | TRISB6 | TRISB5 | TRISB4 | TRISB3 | TRISB2 | TRISB1 | TRISB0 | 151 |
| TRISC | TRISC7 | TRISC6 | TRISC5 | TRISC4 | TRISC3 | TRISC2 | TRISC1 | TRISC0 | 151 |
| TRISD(1) | TRISD7 | TRISD6 | TRISD5 | TRISD4 | TRISD3 | TRISD2 | TRISD1 | TRISD0 | 151 |
| TRISE | WPUE3 | — | — | — | — | TRISE2(1) | TRISE1(1) | TRISE0(1) | 151 |

**Legend:** — = Unimplemented location, read as '0'. Shaded bits are not used by Compare mode.

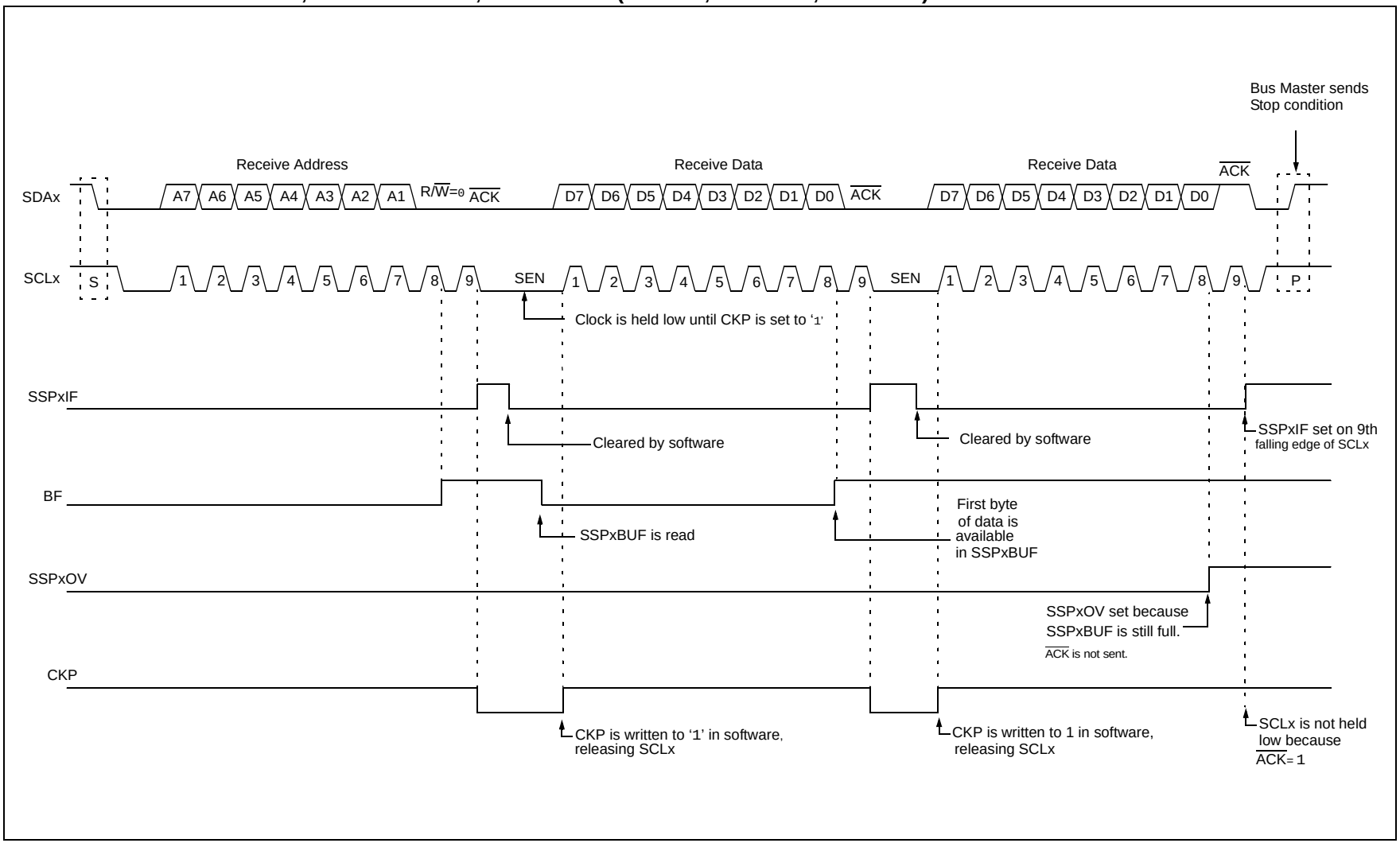
**Note 1:** These registers/bits are available on PIC18(L)F4XK22 devices.

**TABLE 14-6: CONFIGURATION REGISTERS ASSOCIATED WITH COMPARE**

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---|---|---|---|---|---|---|---|---|---|
| CONFIG3H | MCLRE | — | P2BMX | T3CMX | HFOFST | CCP3MX | PBADEN | CCP2MX | 348 |

**Legend:** — = Unimplemented location, read as '0'. Shaded bits are not used by Compare mode.

**FIGURE 15-15: $I^2C$ SLAVE, 7-BIT ADDRESS, RECEPTION (SEN = 1, AHEN = 0, DHEN = 0)**



### 15.5.8 GENERAL CALL ADDRESS SUPPORT

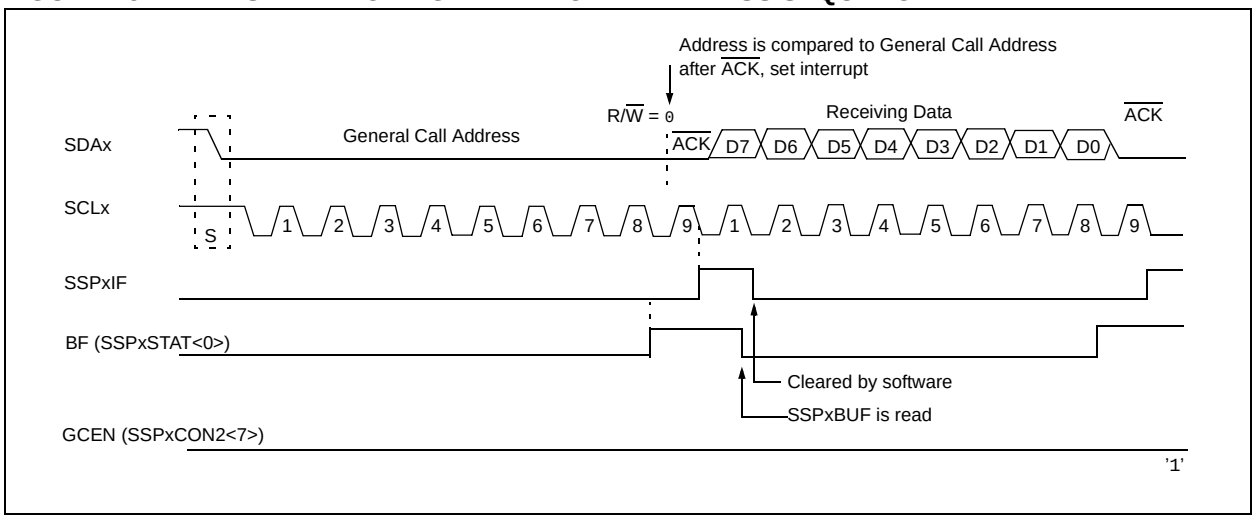
The addressing procedure for the $I^2C$ bus is such that the first byte after the Start condition usually determines which device will be the slave addressed by the master device. The exception is the general call address which can address all devices. When this address is used, all devices should, in theory, respond with an acknowledge.

The general call address is a reserved address in the $I^2C$ protocol, defined as address 0x00. When the GCEN bit of the SSPxCON2 register is set, the slave module will automatically $\overline{ACK}$ the reception of this address regardless of the value stored in SSPxADD. After the slave clocks in an address of all zeros with the R/$\overline{W}$ bit clear, an interrupt is generated and slave software can read SSPxBUF and respond. Figure 15-24 shows a general call reception sequence.

In 10-bit Address mode, the UA bit will not be set on the reception of the general call address. The slave will prepare to receive the second byte as data, just as it would in 7-bit mode.

If the AHEN bit of the SSPxCON3 register is set, just as with any other address reception, the slave hardware will stretch the clock after the 8th falling edge of SCLx. The slave must then set its ACKDT value and release the clock with communication progressing as it would normally.

**FIGURE 15-24: SLAVE MODE GENERAL CALL ADDRESS SEQUENCE**




### 15.5.9 SSPx MASK REGISTER

An SSPx Mask (SSPxMSK) register (Register 15-6) is available in $I^2C$ Slave mode as a mask for the value held in the SSPxSR register during an address comparison operation. A zero ('0') bit in the SSPxMSK register has the effect of making the corresponding bit of the received address a "don't care".

This register is reset to all '1's upon any Reset condition and, therefore, has no effect on standard SSPx operation until written with a mask value.

The SSPx Mask register is active during:

- 7-bit Address mode: address compare of A<7:1>.
- 10-bit Address mode: address compare of A<7:0> only. The SSPx mask has no effect during the reception of the first (high) byte of the address.

**REGISTER 15-5: SSPxCON3: SSPx CONTROL REGISTER 3**

| R-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| ACKTIM | PCIE | SCIE | BOEN | SDAHT | SBCDE | AHEN | DHEN |
| 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 **ACKTIM:** Acknowledge Time Status bit ($I^2C$ mode only)**(3)**

1 = Indicates the $I^2C$ bus is in an Acknowledge sequence, set on 8$^{th}$ falling edge of SCLx clock
0 = Not an Acknowledge sequence, cleared on 9$^{th}$ rising edge of SCLx clock

bit 6 **PCIE**: Stop Condition Interrupt Enable bit ($I^2C$ mode only)

1 = Enable interrupt on detection of Stop condition
0 = Stop detection interrupts are disabled**(2)**

bit 5 **SCIE**: Start Condition Interrupt Enable bit ($I^2C$ mode only)

1 = Enable interrupt on detection of Start or Restart conditions
0 = Start detection interrupts are disabled**(2)**

bit 4 **BOEN:** Buffer Overwrite Enable bit

In SPI Slave mode:**(1)**

1 = SSPxBUF updates every time that a new data byte is shifted in ignoring the BF bit
0 = If new byte is received with BF bit of the SSPxSTAT register already set, SSPxOV bit of the SSPxCON1 register is set, and the buffer is not updated

In $I^2C$ Master mode:

This bit is ignored.

In $I^2C$ Slave mode:

1 = SSPxBUF is updated and $\overline{ACK}$ is generated for a received address/data byte, ignoring the state of the SSPxOV bit only if the BF bit = 0.
0 = SSPxBUF is only updated when SSPxOV is clear

bit 3 **SDAHT:** SDAx Hold Time Selection bit ($I^2C$ mode only)

1 = Minimum of 300 ns hold time on SDAx after the falling edge of SCLx
0 = Minimum of 100 ns hold time on SDAx after the falling edge of SCLx

bit 2 **SBCDE:** Slave Mode Bus Collision Detect Enable bit ($I^2C$ Slave mode only)

If on the rising edge of SCLx, SDAx is sampled low when the module is outputting a high state, the BCLxIF bit of the PIR2 register is set, and bus goes idle

1 = Enable slave bus collision interrupts
0 = Slave bus collision interrupts are disabled

bit 1 **AHEN:** Address Hold Enable bit ($I^2C$ Slave mode only)

1 = Following the 8th falling edge of SCLx for a matching received address byte; CKP bit of the SSPxCON1 register will be cleared and the SCLx will be held low.
0 = Address holding is disabled

**Note 1:** For daisy-chained SPI operation; allows the user to ignore all but the last received byte. SSPxOV is still set when a new byte is received and BF = 1, but hardware continues to write the most recent byte to SSPxBUF.

**2:** This bit has no effect in Slave modes for which Start and Stop condition detection is explicitly listed as enabled.

**3:** The ACKTIM Status bit is active only when the AHEN bit or DHEN bit is set.

### 16.4.4 BREAK CHARACTER SEQUENCE

The EUSART module has the capability of sending the special Break character sequences that are required by the LIN bus standard. A Break character consists of a Start bit, followed by 12 '0' bits and a Stop bit.

To send a Break character, set the SENDB and TXEN bits of the TXSTAx register. The Break character transmission is then initiated by a write to the TXREGx. The value of data written to TXREGx will be ignored and all '0's will be transmitted.

The SENDB bit is automatically reset by hardware after the corresponding Stop bit is sent. This allows the user to preload the transmit FIFO with the next transmit byte following the Break character (typically, the Sync character in the LIN specification).

The TRMT bit of the TXSTAx register indicates when the transmit operation is active or Idle, just as it does during normal transmission. See Figure 16-9 for the timing of the Break character sequence.

#### 16.4.4.1 Break and Sync Transmit Sequence

The following sequence will start a message frame header made up of a Break, followed by an auto-baud Sync byte. This sequence is typical of a LIN bus master.

1. Configure the EUSART for the desired mode.
2. Set the TXEN and SENDB bits to enable the Break sequence.
3. Load the TXREGx with a dummy character to initiate transmission (the value is ignored).
4. Write '55h' to TXREGx to load the Sync character into the transmit FIFO buffer.
5. After the Break has been sent, the SENDB bit is reset by hardware and the Sync character is then transmitted.

When the TXREGx becomes empty, as indicated by the TXxIF, the next data byte can be written to TXREGx.

### 16.4.5 RECEIVING A BREAK CHARACTER

The Enhanced EUSART module can receive a Break character in two ways.

The first method to detect a Break character uses the FERR bit of the RCSTAx register and the Received data as indicated by RCREGx. The Baud Rate Generator is assumed to have been initialized to the expected baud rate.

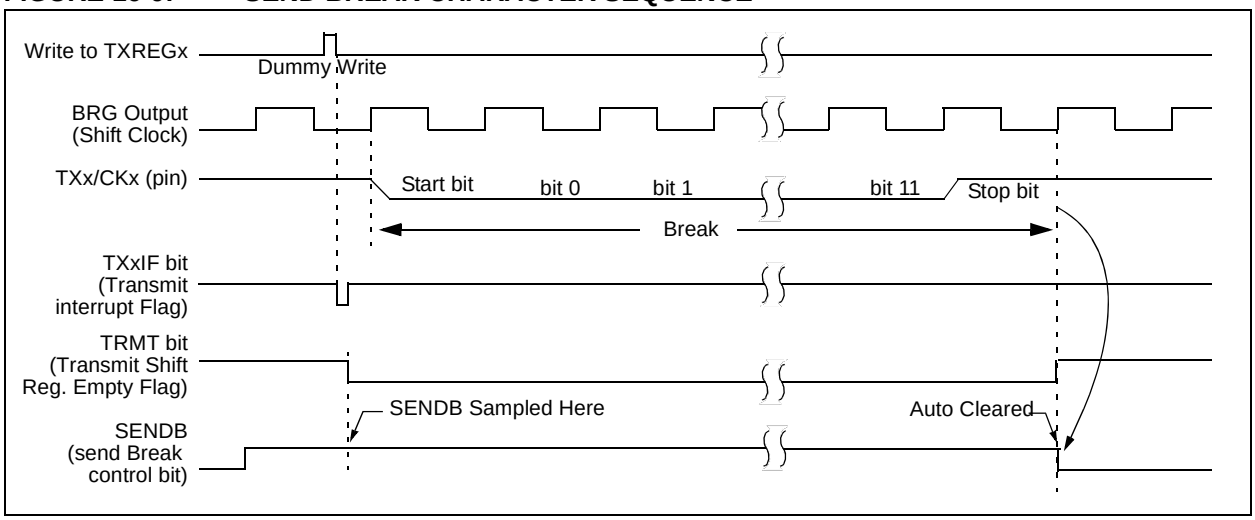
A Break character has been received when;

- RCxIF bit is set
- FERR bit is set
- RCREGx = 00h

The second method uses the Auto-Wake-up feature described in **Section 16.4.3 "Auto-Wake-up on Break"**. By enabling this feature, the EUSART will sample the next two transitions on RXx/DTx, cause an RCxIF interrupt, and receive the next data byte followed by another interrupt.

Note that following a Break character, the user will typically want to enable the Auto-Baud Detect feature. For both methods, the user can set the ABDEN bit of the BAUDCONx register before placing the EUSART in Sleep mode.

**FIGURE 16-9: SEND BREAK CHARACTER SEQUENCE**

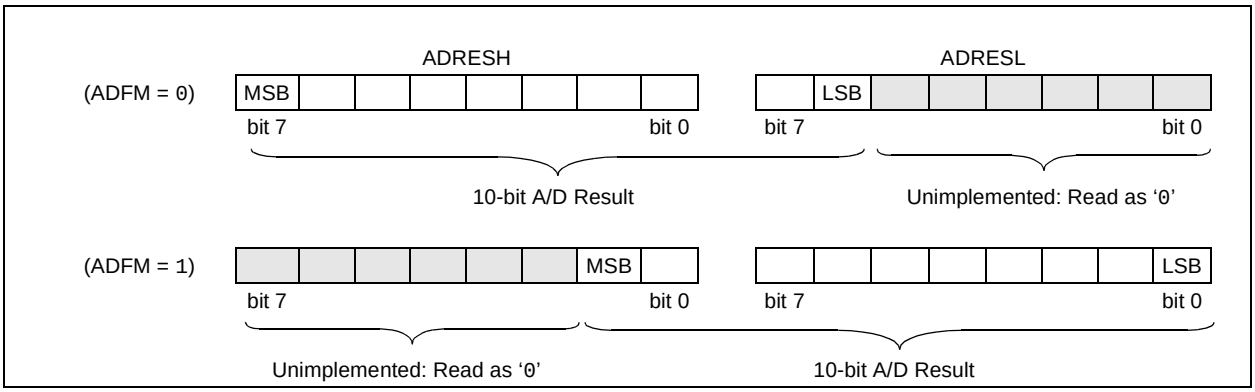


### 17.1.7 RESULT FORMATTING

The 10-bit A/D conversion result can be supplied in two formats, left justified or right justified. The ADFM bit of the ADCON2 register controls the output format.

Figure 17-2 shows the two output formats.

**FIGURE 17-2: 10-BIT A/D CONVERSION RESULT FORMAT**



## 17.3 Register Definitions: ADC Control

| **Note:** | Analog pin control is determined by the ANSELx registers (see Register 10-2) |
|---|---|

**REGISTER 17-1: ADCON0: A/D CONTROL REGISTER 0**

| U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | CHS<4:0> | | | | | GO/$\overline{\text{DONE}}$ | ADON |
| 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 **Unimplemented:** Read as ‘0’

bit 6-2 **CHS<4:0>: Analog Channel Select bits**
00000 = AN0
00001 = AN1
00010 = AN2
00011 = AN3
00100 = AN4
00101 = AN5**(1)**
00110 = AN6**(1)**
00111 = AN7**(1)**
01000 = AN8
01001 = AN9
01010 = AN10
01011 = AN11
01100 = AN12
01101 = AN13
01110 = AN14
01111 = AN15
10000 = AN16
10001 = AN17
10010 = AN18
10011 = AN19
10100 = AN20**(1)**
10101 = AN21**(1)**
10110 = AN22**(1)**
10111 = AN23**(1)**
11000 = AN24**(1)**
11001 = AN25**(1)**
11010 = AN26**(1)**
11011 = AN27**(1)**
11100 = Reserved
11101 = CTMU
11110 = DAC
11111 = FVR BUF2 (1.024V/2.048V/2.096V Volt Fixed Voltage Reference)**(2)**

bit 1 **GO/$\overline{\text{DONE}}$:** A/D Conversion Status bit
1 = A/D conversion cycle in progress. Setting this bit starts an A/D conversion cycle.
This bit is automatically cleared by hardware when the A/D conversion has completed.
0 = A/D conversion completed/not in progress

bit 0 **ADON:** ADC Enable bit
1 = ADC is enabled
0 = ADC is disabled and consumes no operating current

**Note 1:** Available on PIC18(L)F4XK22 devices only.
**2:** Allow greater than 15 μs acquisition time when measuring the Fixed Voltage Reference.

## 17.4 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 17-5. 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), see Figure 17-5. The maximum recommended impedance for analog sources is 3 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 17-1 may be used. This equation assumes that 1/2 LSb error is used (1024 steps for the ADC). The 1/2 LSb error is the maximum error allowed for the ADC to meet its specified resolution.

**EQUATION 17-1: ACQUISITION TIME EXAMPLE**

*Assumptions:* $Temperature = 50°C$ *and external impedance of* $10k\Omega$ $3.0V$ $V_{DD}$

$$T_{ACQ} = \text{Amplifier Settling Time} + \text{Hold Capacitor Charging Time} + \text{Temperature Coefficient}$$
$$= T_{AMP} + T_C + T_{COFF}$$
$$= 5\mu s + T_C + [(\text{Temperature} - 25°C)(0.05\mu s/°C)]$$

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

$$V_{APPLIED}\left(1 - \frac{1}{2047}\right) = V_{CHOLD} \quad ;[1]\ V_{CHOLD}\ \text{charged to within 1/2 lsb}$$

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

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{APPLIED}\left(1 - \frac{1}{2047}\right) \quad ;\text{combining [1] and [2]}$$

*Solving for* $T_C$*:*

$$T_C = -C_{HOLD}(R_{IC} + R_{SS} + R_S)\ ln(1/2047)$$
$$= -13.5pF(1k\Omega + 700\Omega + 10k\Omega)\ ln(0.0004885)$$
$$= 1.20\mu s$$

*Therefore:*

$$T_{ACQ} = 5\mu s + 1.20\mu s + [(50°C - 25°C)(0.05\mu s/°C)]$$
$$= 7.45\mu s$$

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

**2:** The charge holding capacitor (CHOLD) is discharged after each conversion.

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

## 18.5 Operation During Sleep

The comparator, if enabled before entering Sleep mode, remains active during Sleep. The additional current consumed by the comparator is shown separately in **Section 27.0 "Electrical Specifications"**. If the comparator is not used to wake the device, power consumption can be minimized while in Sleep mode by turning off the comparator. Each comparator is turned off by clearing the CxON bit of the CMxCON0 register.

A change to the comparator output can wake-up the device from Sleep. To enable the comparator to wake the device from Sleep, the CxIE bit of the PIE2 register and the PEIE/GIEL bit of the INTCON register must be set. The instruction following the SLEEP instruction always executes following a wake from Sleep. If the GIE/GIEH bit of the INTCON register is also set, the device will then execute the Interrupt Service Routine.

## 18.6 Effects of a Reset

A device Reset forces the CMxCON0 and CM2CON1 registers to their Reset states. This forces both comparators and the voltage references to their Off states.Comparator Control Registers.

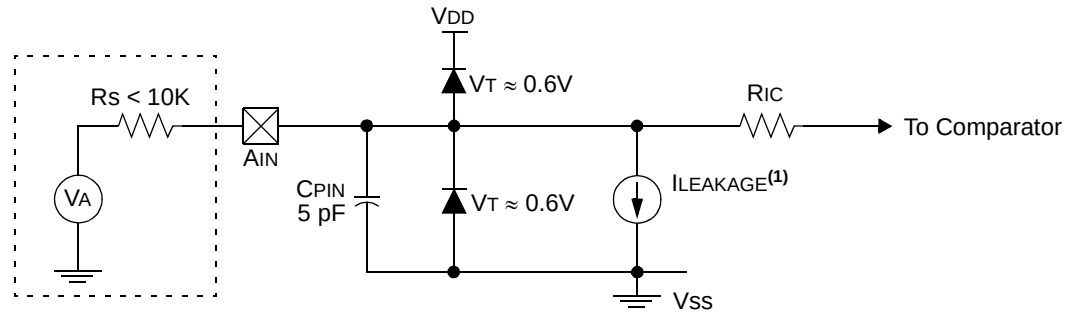
## 18.7 Analog Input Connection Considerations

A simplified circuit for an analog input is shown in Figure 18-5. Since the analog input pins share their connection with a digital input, they have reverse biased ESD protection diodes to $V_{DD}$ and $V_{SS}$. The analog input, therefore, must be between $V_{SS}$ and $V_{DD}$. If the input voltage deviates from this range by more than 0.6V in either direction, one of the diodes is forward biased and a latch-up may occur.

A maximum source impedance of 10 kΩ is recommended for the analog sources. Also, any external component connected to an analog input pin, such as a capacitor or a Zener diode, should have very little leakage current to minimize inaccuracies introduced.

| | |
|---|---|
| **Note 1:** | When reading a PORT register, all pins configured as analog inputs will read as a '0'. Pins configured as digital inputs will convert as an analog input, according to the input specification. |
| **2:** | Analog levels on any pin defined as a digital input, may cause the input buffer to consume more current than is specified. |

**FIGURE 18-5: ANALOG INPUT MODEL**




**Legend:** $C_{PIN}$ = Input Capacitance
$I_{LEAKAGE}$ = Leakage Current at the pin due to various junctions
$R_{IC}$ = Interconnect Resistance
$R_S$ = Source Impedance
$V_A$ = Analog Voltage
$V_T$ = Threshold Voltage

**Note 1:** See **Section 27.0 "Electrical Specifications"**.

**TABLE 20-1: DIVSRCLK FREQUENCY TABLE**

| SRCLK<2:0> | Divider | Fosc = 20 MHz | Fosc = 16 MHz | Fosc = 8 MHz | Fosc = 4 MHz | Fosc = 1 MHz |
|---|---|---|---|---|---|---|
| 111 | 512 | 25.6 μs | 32 μs | 64 μs | 128 μs | 512 μs |
| 110 | 256 | 12.8 μs | 16 μs | 32 μs | 64 μs | 256 μs |
| 101 | 128 | 6.4 μs | 8 μs | 16 μs | 32 μs | 128 μs |
| 100 | 64 | 3.2 μs | 4 μs | 8 μs | 16 μs | 64 μs |
| 011 | 32 | 1.6 μs | 2 μs | 4 μs | 8 μs | 32 μs |
| 010 | 16 | 0.8 μs | 1 μs | 2 μs | 4 μs | 16 μs |
| 001 | 8 | 0.4 μs | 0.5 μs | 1 μs | 2 μs | 8 μs |
| 000 | 4 | 0.2 μs | 0.25 μs | 0.5 μs | 1 μs | 4 μs |

## 27.3 DC Characteristics: RC Run Supply Current, PIC18(L)F2X/4XK22

| PIC18LF2X/4XK22 | | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +125°C | | | | | |
|---|---|---|---|---|---|---|---|
| **PIC18F2X/4XK22** | | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +125°C | | | | | |
| **Param No.** | **Device Characteristics** | **Typ** | **Max** | **Units** | **Conditions** | | |
| D020 | Supply Current (IDD)(1),(2) | 3.6 | 23 | µA | -40°C | VDD = 1.8V | FOSC = 31 kHz (**RC_RUN** mode, LFINTOSC source) |
| | | 3.9 | 25 | µA | +25°C | | |
| | | 3.9 | — | µA | +60°C | | |
| | | 3.9 | 28 | µA | +85°C | | |
| | | 4.0 | 30 | µA | 125°C | | |
| D021 | | 8.1 | 26 | µA | -40°C | VDD = 3.0V | |
| | | 8.4 | 30 | µA | +25°C | | |
| | | 8.6 | — | µA | +60°C | | |
| | | 8.7 | 35 | µA | +85°C | | |
| | | 10.7 | 40 | µA | +125°C | | |
| D022 | | 16 | 35 | µA | -40°C | VDD = 2.3V | FOSC = 31 kHz (**RC_RUN** mode, LFINTOSC source) |
| | | 17 | 35 | µA | +25°C | | |
| | | 18 | 35 | µA | +85°C | | |
| | | 19 | 50 | µA | +125°C | | |
| D023 | | 18 | 50 | µA | -40°C | VDD = 3.0V | |
| | | 20 | 50 | µA | +25°C | | |
| | | 21 | 50 | µA | +85°C | | |
| | | 22 | 60 | µA | +125°C | | |
| D024 | | 19 | 55 | µA | -40°C | VDD = 5.0V | |
| | | 21 | 55 | µA | +25°C | | |
| | | 22 | 55 | µA | +85°C | | |
| | | 23 | 70 | µA | +125°C | | |
| D025 | | 0.14 | 0.25 | mA | -40°C to +125°C | VDD = 1.8V | FOSC = 500 kHz (**RC_RUN** mode, MFINTOSC source) |
| D026 | | 0.17 | 0.30 | mA | -40°C to +125°C | VDD = 3.0V | |
| D027 | | 0.18 | 0.25 | mA | -40°C to +125°C | VDD = 2.3V | FOSC = 500 kHz (**RC_RUN** mode, MFINTOSC source) |
| D028 | | 0.20 | 0.30 | mA | -40°C to +125°C | VDD = 3.0V | |
| D029 | | 0.25 | 0.35 | mA | -40°C to +125°C | VDD = 5.0V | |

**Note 1:** The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption.
Test condition: All Peripheral Module Control bits in PMD0, PMD1 and PMD2 set to '1'.

**2:** The test conditions for all IDD measurements in active operation mode are:
All I/O pins set as outputs driven to Vss;
$\overline{MCLR}$ = VDD;
OSC1 = external square wave, from rail-to-rail (PRI_RUN and PRI_IDLE only).

**TABLE 27-3: FIXED VOLTAGE REFERENCE (FVR) SPECIFICATIONS**

| **Operating Conditions:** -40°C < TA < +125°C (unless otherwise stated) | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym** | **Characteristics** | **Min** | **Typ** | **Max** | **Units** | **Comments** |
| VR01 | VROUT | VR voltage output to ADC | 0.973 | 1.024 | 1.085 | V | 1x output, VDD ≥ 2.5V |
| | | | 1.946 | 2.048 | 2.171 | V | 2x output, VDD ≥ 2.5V |
| | | | 3.891 | 4.096 | 4.342 | V | 4x output, VDD ≥ 4.75V (PIC18F2X/4XK22) |
| VR02 | VROUT | VR voltage output all other modules | 0.942 | 1.024 | 1.096 | V | 1x output, VDD ≥ 2.5V |
| | | | 1.884 | 2.048 | 2.191 | V | 2x output, VDD ≥ 2.5V |
| | | | 3.768 | 4.096 | 4.383 | V | 4x output, VDD ≥ 4.75V (PIC18F2X/4XK22) |
| VR04* | TSTABLE | Settling Time | — | 25 | 100 | µs | 0 to 125°C |

* These parameters are characterized but not tested.

**TABLE 27-4: CHARGE TIME MEASUREMENT UNIT (CTMU) SPECIFICATIONS**

| **Operating Conditions:** 1.8V < VDD < 5.5V, -40°C < TA < +125°C (unless otherwise stated) | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym** | **Characteristics** | **Min** | **Typ(1)** | **Max** | **Units** | **Comments** |
| CT01 | IOUT1 | CTMU Current Source, Base Range | — | 0.55 | — | µA | IRNG<1:0>=01 |
| CT02 | IOUT2 | CTMU Current Source, 10X Range | — | 5.5 | — | µA | IRNG<1:0>=10 |
| CT03 | IOUT3 | CTMU Current Source, 100X Range | — | 55 | — | µA | IRNG<1:0>=11 VDD ≥ 3.0V |

**Note 1:** Nominal value at center point of current trim range (CTMUICON<7:2>=000000).

**FIGURE 27-13: EXAMPLE SPI MASTER MODE TIMING (CKE = 0)**

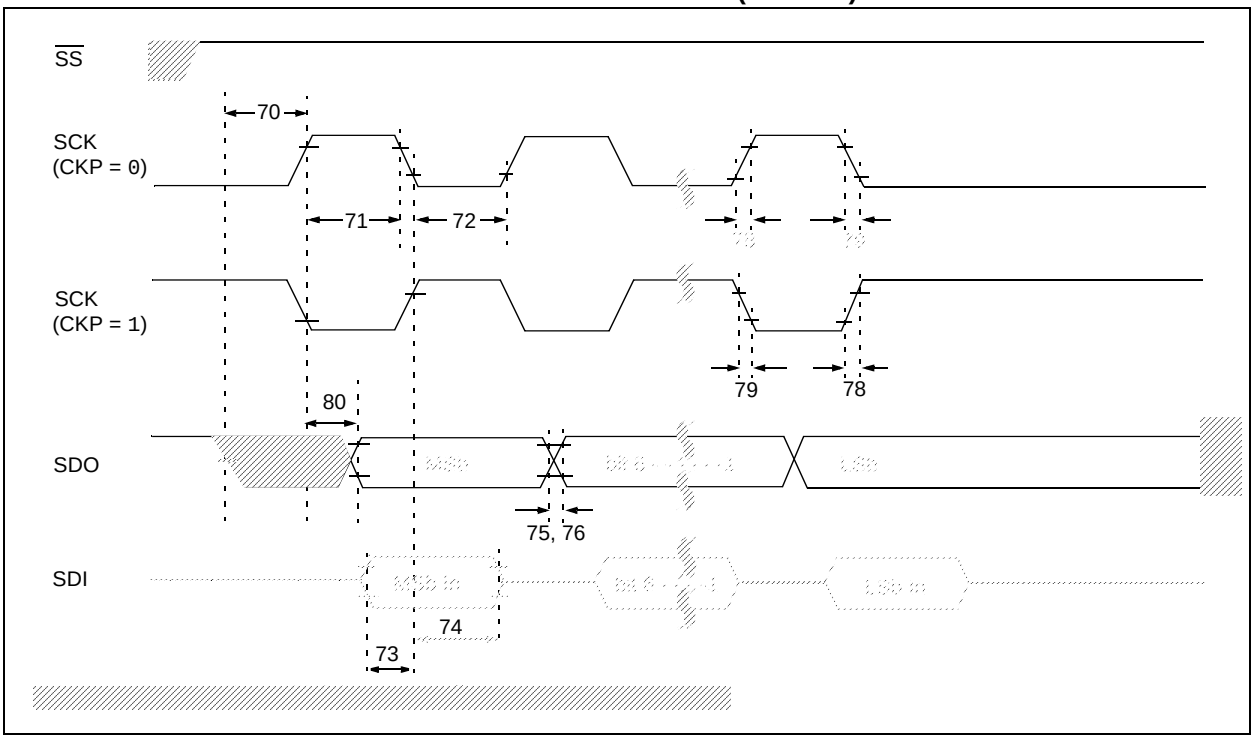
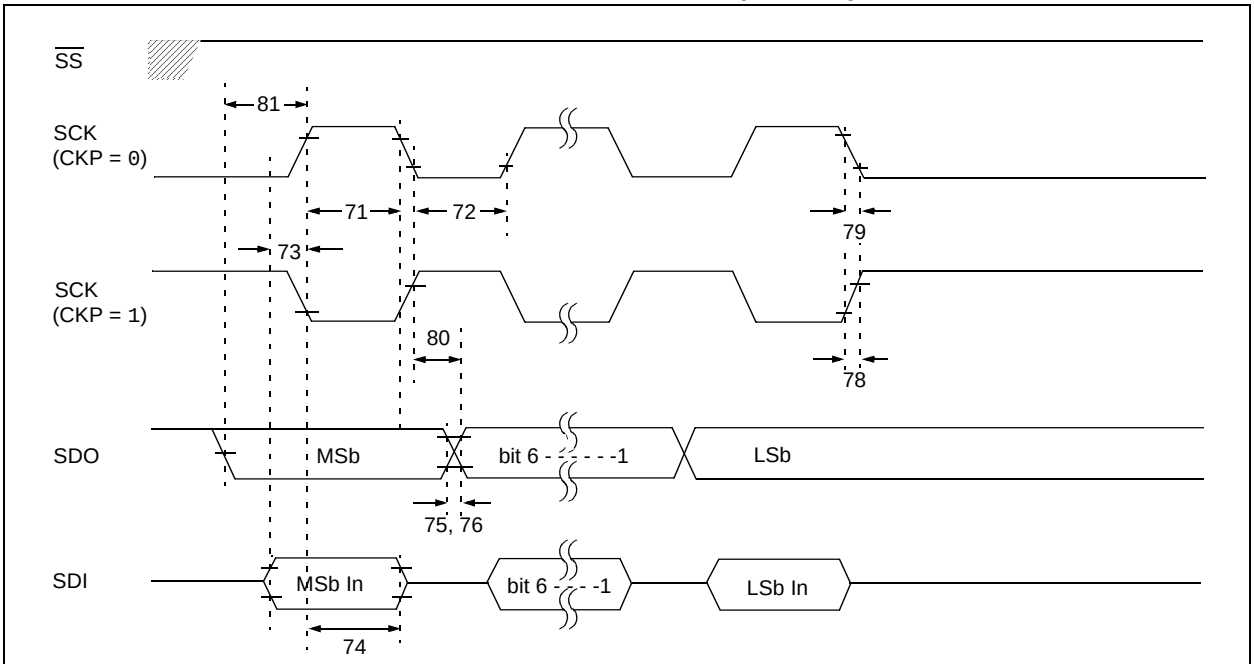



**FIGURE 27-14: EXAMPLE SPI MASTER MODE TIMING (CKE = 1)**




**Note:** Refer to Figure 27-6 for load conditions.

**FIGURE 28-72: PIC18LF2X/4XK22 TYPICAL IDD: SEC_RUN 32.768 kHz**

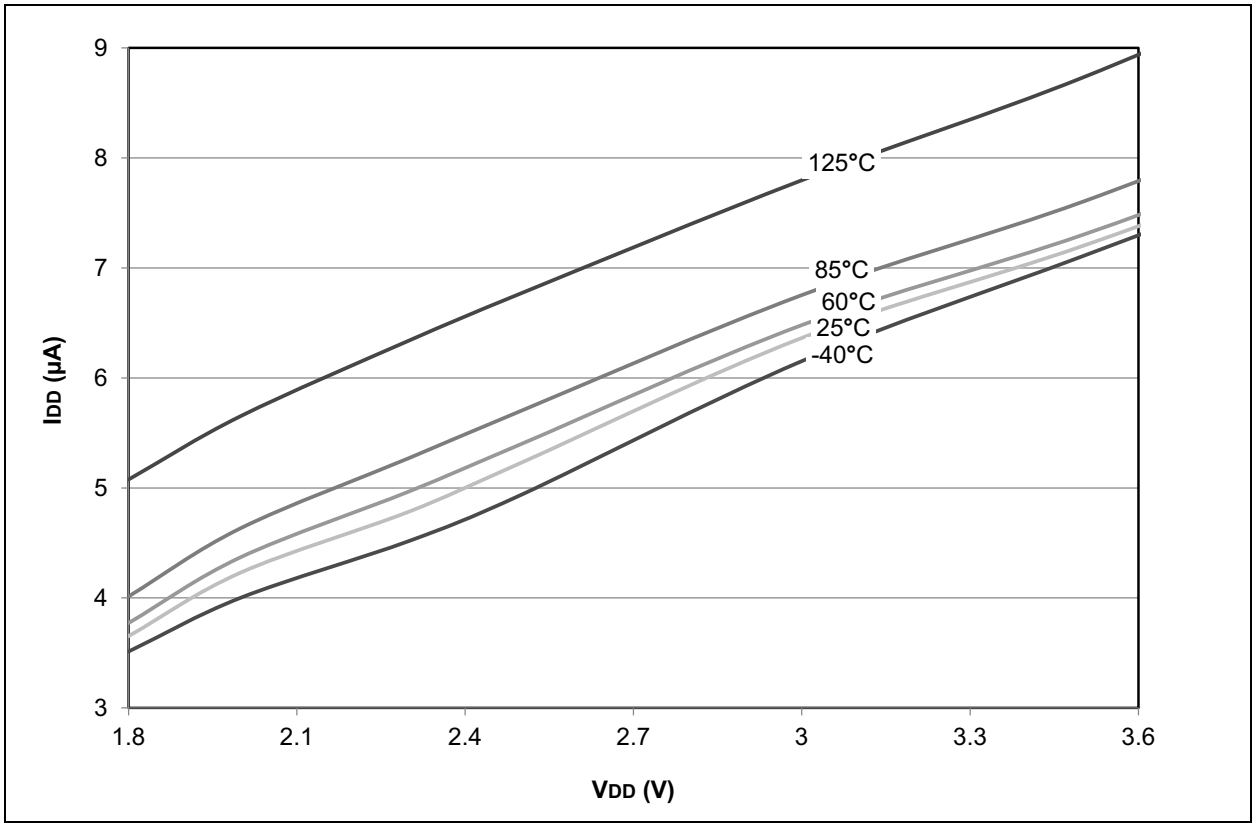



**FIGURE 28-73: PIC18LF2X/4XK22 MAXIMUM IDD: SEC_RUN 32.768 kHz**

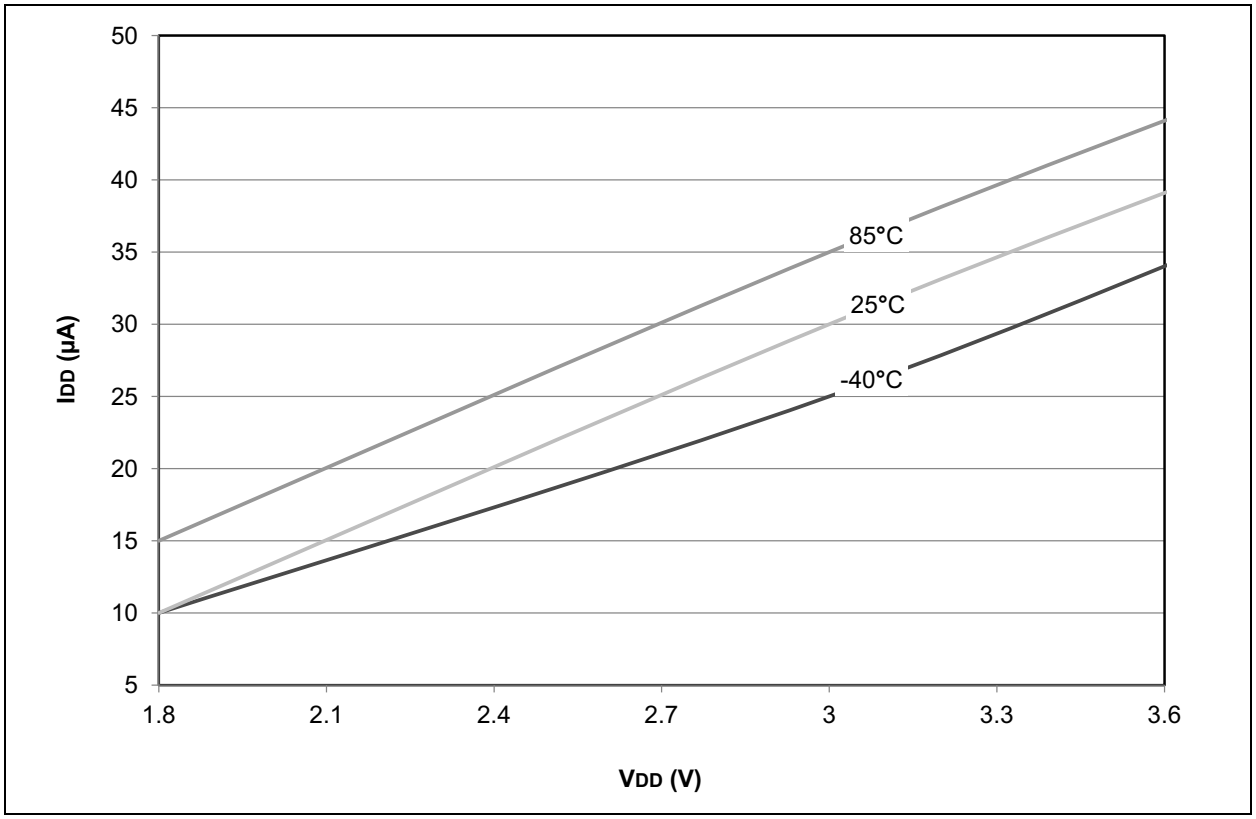


**FIGURE 28-98: PIC18F2X/4XK22 TYPICAL FIXED VOLTAGE REFERENCE 4x OUTPUT**

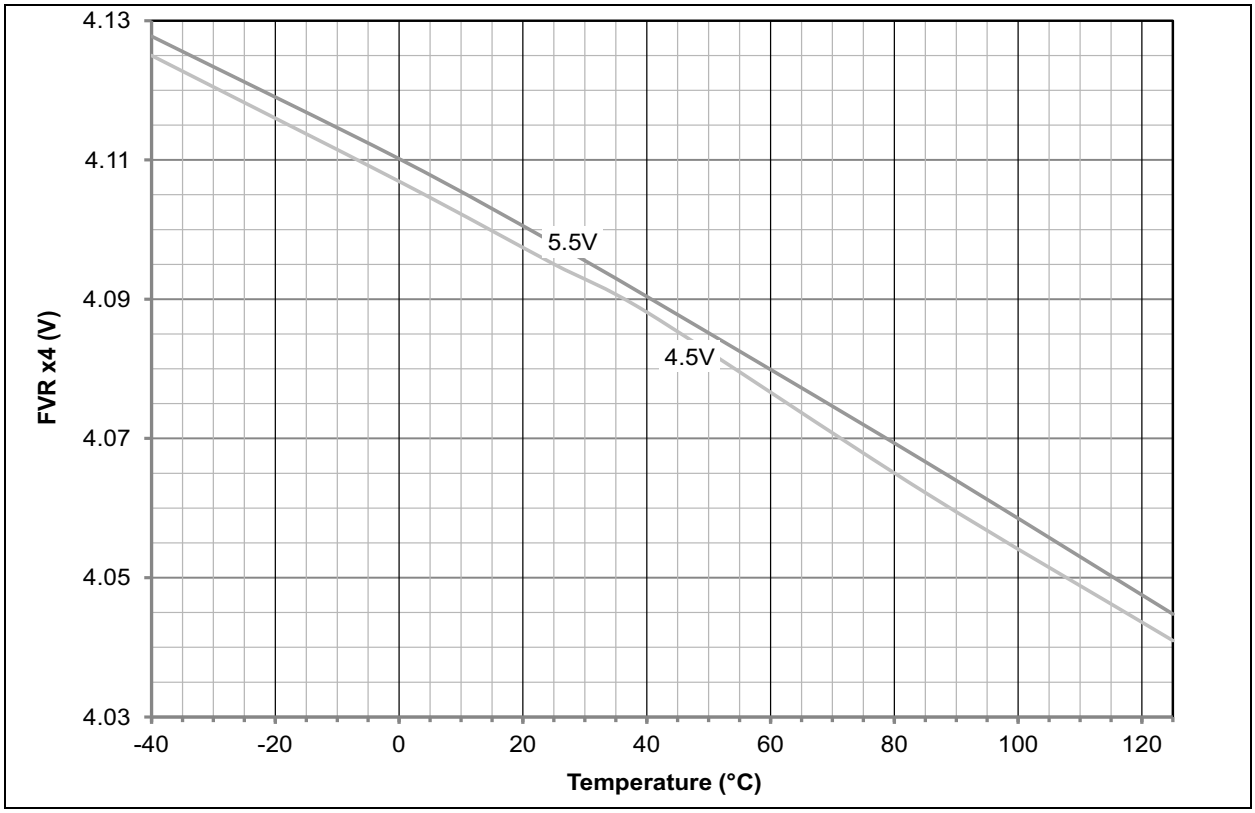
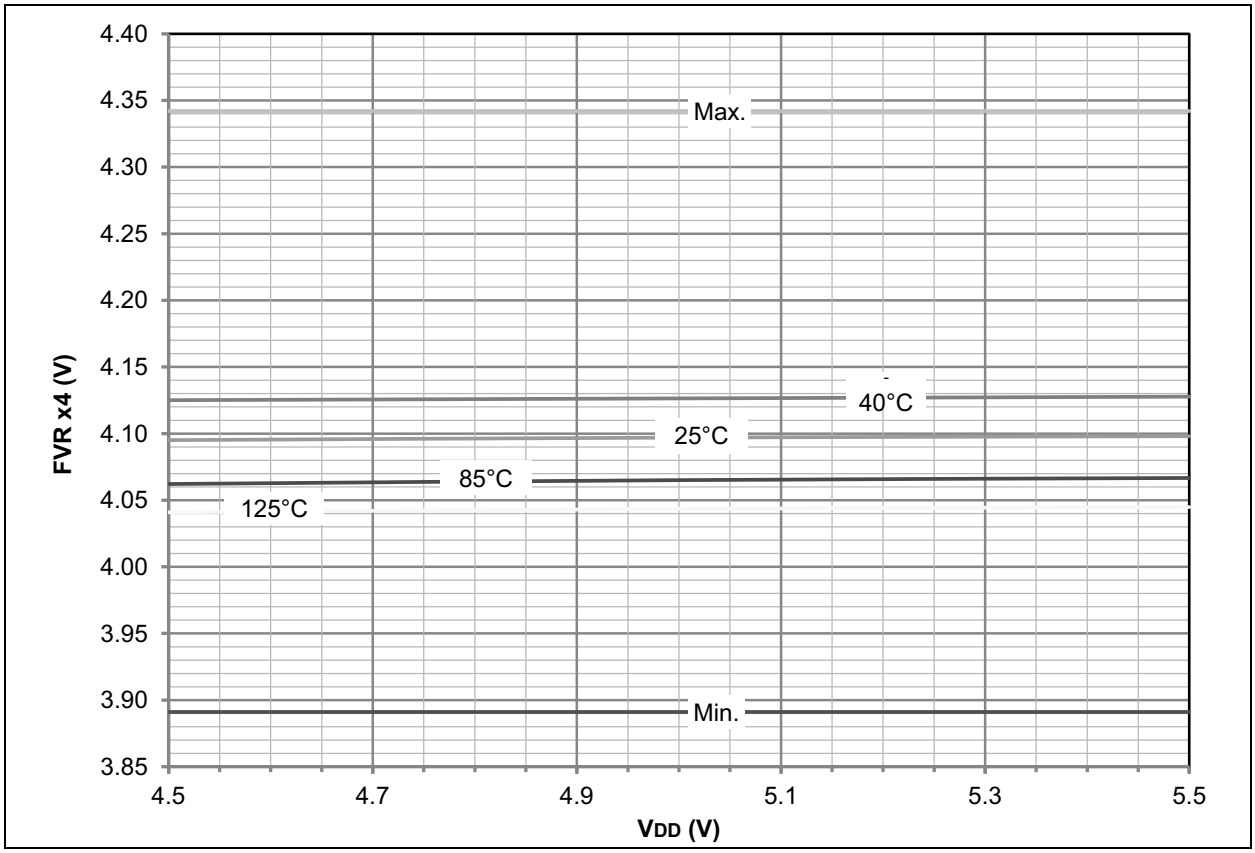
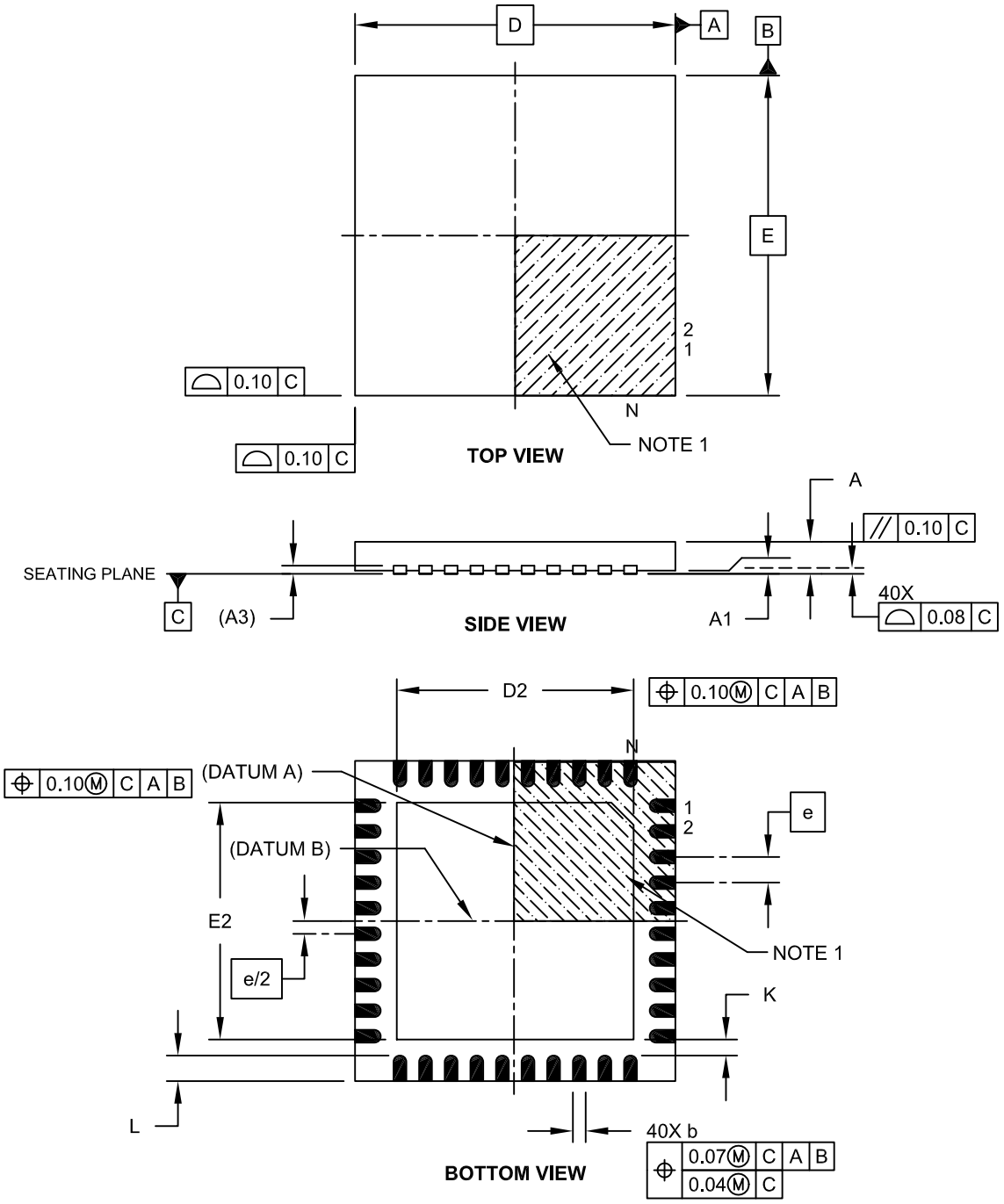



**FIGURE 28-99: PIC18F2X/4XK22 TYPICAL FIXED VOLTAGE REFERENCE 4x OUTPUT**



### 40-Lead Ultra Thin Plastic Quad Flat, No Lead Package (MV) – 5x5x0.5 mm Body [UQFN]

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




Microchip Technology Drawing C04-156A Sheet 1 of 2