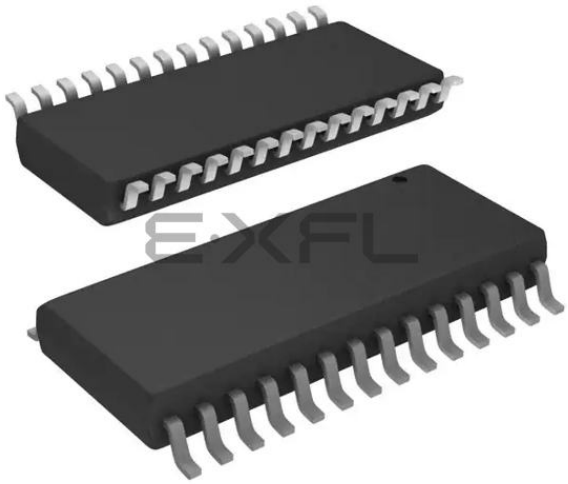


**Welcome to E-XFL.COM**

## 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 | 32MHz |
| Connectivity | I²C, LINbus, SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, POR, PWM, WDT |
| Number of I/O | 25 |
| Program Memory Size | 28KB (16K x 14) |
| Program Memory Type | FLASH |
| EEPROM Size | 256 x 8 |
| RAM Size | 2K x 8 |
| Voltage - Supply (Vcc/Vdd) | 1.8V ~ 3.6V |
| Data Converters | A/D 24x10b; D/A 1x5b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 28-SOIC (0.295", 7.50mm Width) |
| Supplier Device Package | 28-SOIC |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18856t-i-so |

## 5.11 Determining the Cause of a Reset

Upon any Reset, multiple bits in the STATUS and PCON register are updated to indicate the cause of the Reset. Table 5-3 and Table 5-4 show the Reset conditions of these registers.

**TABLE 5-3: RESET STATUS BITS AND THEIR SIGNIFICANCE**

| STKOVF | STKUNF | $\overline{\text{RWDT}}$ | $\overline{\text{RMCLR}}$ | $\overline{\text{RI}}$ | $\overline{\text{POR}}$ | $\overline{\text{BOR}}$ | $\overline{\text{TO}}$ | $\overline{\text{PD}}$ | Condition |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 | 0 | x | 1 | 1 | Power-on Reset |
| 0 | 0 | 1 | 1 | 1 | 0 | x | 0 | x | Illegal, $\overline{\text{TO}}$ is set on $\overline{\text{POR}}$ |
| 0 | 0 | 1 | 1 | 1 | 0 | x | x | 0 | Illegal, $\overline{\text{PD}}$ is set on $\overline{\text{POR}}$ |
| 0 | 0 | u | 1 | 1 | u | 0 | 1 | 1 | Brown-out Reset |
| u | u | 0 | u | u | u | u | 0 | u | WDT Reset |
| u | u | u | u | u | u | u | 0 | 0 | WDT Wake-up from Sleep |
| u | u | u | u | u | u | u | 1 | 0 | Interrupt Wake-up from Sleep |
| u | u | u | 0 | u | u | u | u | u | $\overline{\text{MCLR}}$ Reset during normal operation |
| u | u | u | 0 | u | u | u | 1 | 0 | $\overline{\text{MCLR}}$ Reset during Sleep |
| u | u | u | u | 0 | u | u | u | u | RESET Instruction Executed |
| 1 | u | u | u | u | u | u | u | u | Stack Overflow Reset (STVREN = 1) |
| u | 1 | u | u | u | u | u | u | u | Stack Underflow Reset (STVREN = 1) |

**TABLE 5-4: RESET CONDITION FOR SPECIAL REGISTERS**

| Condition | Program Counter | STATUS Register | PCON0 Register |
|---|---|---|---|
| Power-on Reset | 0000h | ---1 1000 | 00-- 110x |
| $\overline{\text{MCLR}}$ Reset during normal operation | 0000h | ---u uuuu | uu-- 0uuu |
| $\overline{\text{MCLR}}$ Reset during Sleep | 0000h | ---1 0uuu | uu-- 0uuu |
| WDT Reset | 0000h | ---0 uuuu | uu-0 uuuu |
| WDT Wake-up from Sleep | PC + 1 | ---0 0uuu | uu-u uuuu |
| WDT Window Violation | 0000h | ---0 uuuu | uu00 uuuu |
| Brown-out Reset | 0000h | ---1 1000 | 00-1 11u0 |
| Interrupt Wake-up from Sleep | PC + 1(1) | ---1 0uuu | uu-u uuuu |
| RESET Instruction Executed | 0000h | ---u uuuu | uu-u u0uu |
| Stack Overflow Reset (STVREN = 1) | 0000h | ---u uuuu | 1u-u uuuu |
| Stack Underflow Reset (STVREN = 1) | 0000h | ---u uuuu | u1-u uuuu |

**Legend:** u = unchanged, x = unknown, - = unimplemented bit, reads as '0'.

**Note 1:** When the wake-up is due to an interrupt and Global Enable bit (GIE) is set, the return address is pushed on the stack and PC is loaded with the interrupt vector (0004h) after execution of PC + 1.

**REGISTER 7-12: PIR1: PERIPHERAL INTERRUPT REQUEST REGISTER 1**

| R/W/HS-0/0 | R/W/HS-0/0 | U-0 | U-0 | U-0 | U-0 | R/W/HS-0/0 | R/W/HS-0/0 |
|---|---|---|---|---|---|---|---|
| OSFIF | CSWIF | — | — | — | — | ADTIF | ADIF |
| 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 | HS = Hardware set |

bit 7 **OSFIF**: Oscillator Fail-Safe Interrupt Flag bit
1 = Oscillator fail-safe interrupt has occurred (must be cleared in software)
0 = No oscillator fail-safe interrupt

bit 6 **CSWIF:** Clock Switch Complete Interrupt Flag bit
1 = The clock switch module indicates an interrupt condition (must be cleared in software)
0 = The clock switch does not indicate an interrupt condition

bit 5-2 **Unimplemented:** Read as '0'

bit 1 **ADTIF**: Analog-to-Digital Converter (ADC) Threshold Compare Interrupt Flag bit
1 = An A/D measurement was beyond the configured threshold (must be cleared in software)
0 = A/D measurements have been within the configured threshold

bit 0 **ADIF**: Analog-to-Digital Converter (ADC) Interrupt Flag bit
1 = An A/D conversion or complex operation has completed (must be cleared in software)
0 = An A/D conversion or complex operation is not complete

**Note:** Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

## 8.3 Register Definitions: Voltage Regulator and DOZE Control

**REGISTER 8-1: VREGCON: VOLTAGE REGULATOR CONTROL REGISTER (1)**

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0/0 | R/W-1/1 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | VREGPM | Reserved |
| 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 7-2 **Unimplemented:** Read as '0'

bit 1 **VREGPM**: Voltage Regulator Power Mode Selection bit
- 1 = Low-Power Sleep mode enabled in Sleep(2)
  Draws lowest current in Sleep, slower wake-up
- 0 = Normal Power mode enabled in Sleep(2)
  Draws higher current in Sleep, faster wake-up

bit 0 **Reserved:** Read as '1'. Maintain this bit set.

**Note 1:** PIC16F18855/75 only.

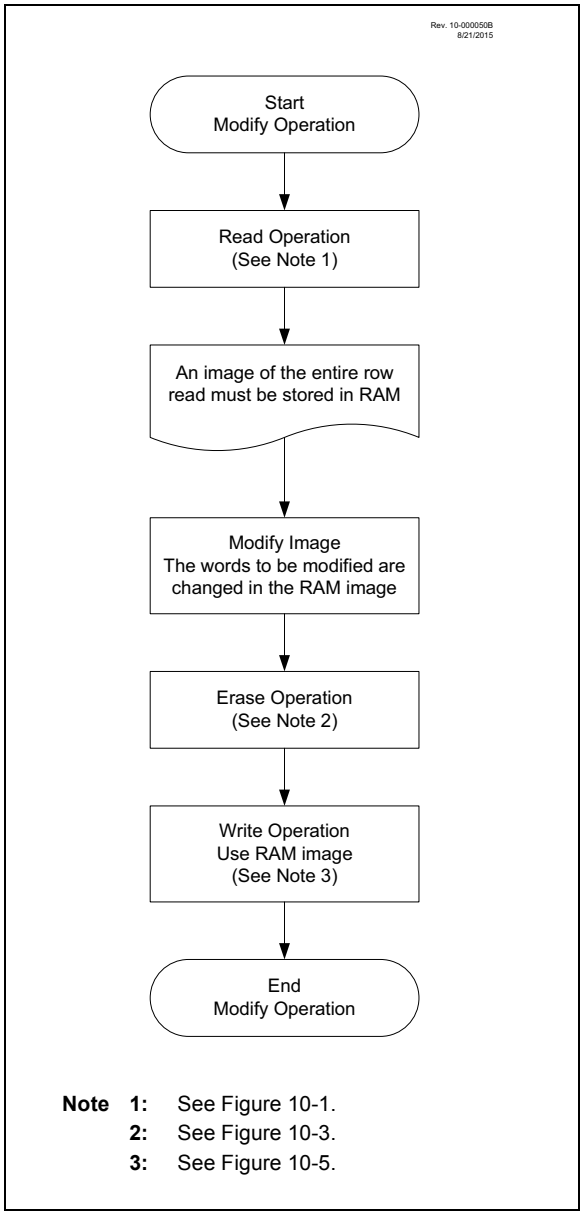
**2:** See **Section 37.0 "Electrical Specifications"**.

### 10.4.6 MODIFYING FLASH PROGRAM MEMORY

When modifying existing data in a program memory row, and data within that row must be preserved, it must first be read and saved in a RAM image. Program memory is modified using the following steps:

1. Load the starting address of the row to be modified.
2. Read the existing data from the row into a RAM image.
3. Modify the RAM image to contain the new data to be written into program memory.
4. Load the starting address of the row to be rewritten.
5. Erase the program memory row.
6. Load the write latches with data from the RAM image.
7. Initiate a programming operation.

**FIGURE 10-6: FLASH PROGRAM MEMORY MODIFY FLOWCHART**




**Note 1:** See Figure 10-1.
**2:** See Figure 10-3.
**3:** See Figure 10-5.

## 12.11 Register Definitions: PORTD

**REGISTER 12-32: PORTD: PORTD REGISTER**

| R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u |
|---|---|---|---|---|---|---|---|
| RD7 | RD6 | RD5 | RD4 | RD3 | RD2 | RD1 | RD0 |
| 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 **RD<7:0>**: PORTD I/O Value bits(1)
1 = Port pin is ≥ VIH
0 = Port pin is ≤ VIL

**Note 1:** Writes to PORTD are actually written to corresponding LATD register. Reads from PORTD register is return of actual I/O pin values.

**REGISTER 12-33: TRISD: PORTD TRI-STATE REGISTER**

| R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u |
|---|---|---|---|---|---|---|---|
| TRISD7 | TRISD6 | TRISD5 | TRISD4 | TRISD3 | TRISD2 | TRISD1 | TRISD0 |
| 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 **TRISD<7:0>**: TRISD I/O Value bits
1 = Port pin is ≥ VIH
0 = Port pin is ≤ VIL

## 16.0 FIXED VOLTAGE REFERENCE (FVR)

The Fixed Voltage Reference, or FVR, is a stable voltage reference, independent of VDD, with 1.024V, 2.048V or 4.096V selectable output levels. The output of the FVR can be configured to supply a reference voltage to the following:

- ADC input channel
- ADC positive reference
- Comparator positive input
- Digital-to-Analog Converter (DAC)

The FVR can be enabled by setting the FVREN bit of the FVRCON register.

| **Note:** | Fixed Voltage Reference output cannot exceed VDD. |
|---|---|

### 16.1 Independent Gain Amplifiers

The output of the FVR, which is connected to the ADC, comparators, and DAC, is routed through two independent programmable gain amplifiers. Each amplifier can be programmed for a gain of 1x, 2x or 4x, to produce the three possible voltage levels.

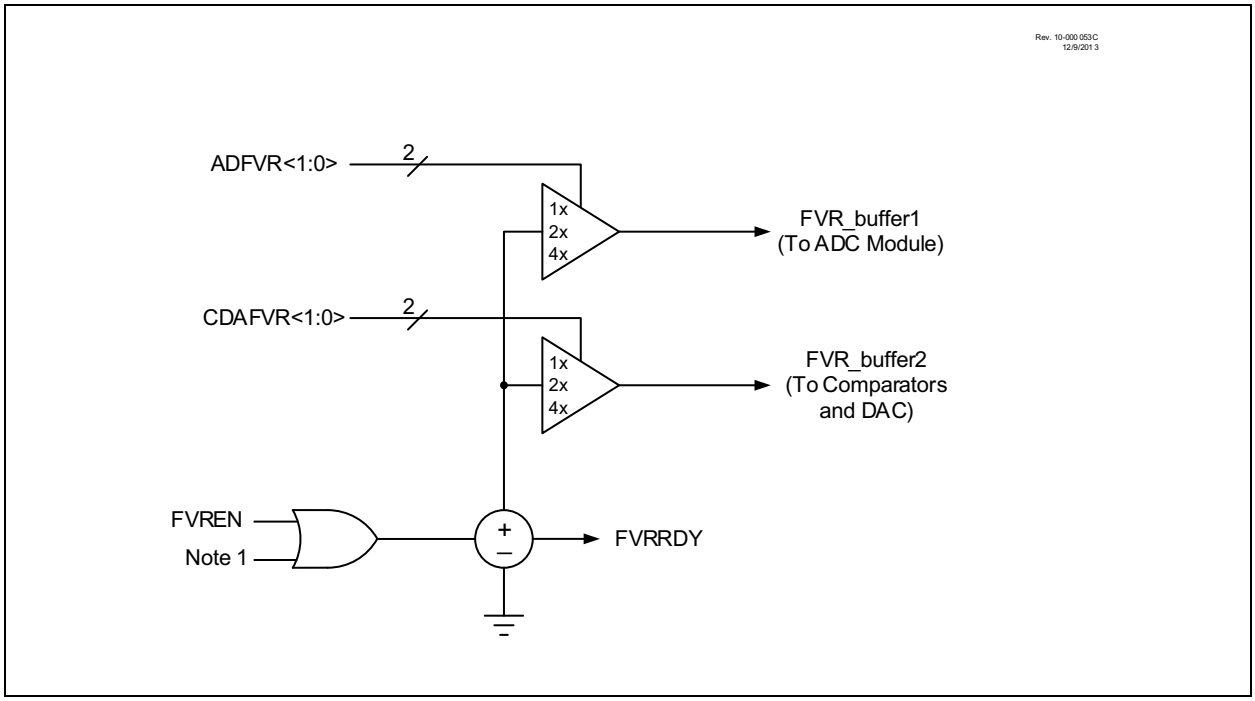
The ADFVR<1:0> bits of the FVRCON register are used to enable and configure the gain amplifier settings for the reference supplied to the ADC module. Reference **Section 23.0 "Analog-to-Digital Converter With Computation (ADC2) Module"** for additional information.

The CDAFVR<1:0> bits of the FVRCON register are used to enable and configure the gain amplifier settings for the reference supplied to the DAC and comparator module. Reference **Section 25.0 "5-Bit Digital-to-Analog Converter (DAC1) Module"** and **Section 18.0 "Comparator Module"** for additional information.

### 16.2 FVR Stabilization Period

When the Fixed Voltage Reference module is enabled, it requires time for the reference and amplifier circuits to stabilize. Once the circuits stabilize and are ready for use, the FVRRDY bit of the FVRCON register will be set.

**FIGURE 16-1: VOLTAGE REFERENCE BLOCK DIAGRAM**



## 22.0 CONFIGURABLE LOGIC CELL (CLC)

The Configurable Logic Cell (CLCx) module provides programmable logic that operates outside the speed limitations of software execution. The logic cell takes up to 32 input signals and, through the use of configurable gates, reduces the 32 inputs to four logic lines that drive one of eight selectable single-output logic functions.

Input sources are a combination of the following:

- I/O pins
- Internal clocks
- Peripherals
- Register bits

The output can be directed internally to peripherals and to an output pin.

The CLC modules available are shown in Table 22-1.

**TABLE 22-1: AVAILABLE CLC MODULES**

| Device | CLC1 | CLC2 | CLC3 | CLC4 |
|---|---|---|---|---|
| PIC16(L)F18856/76 | ● | ● | ● | ● |

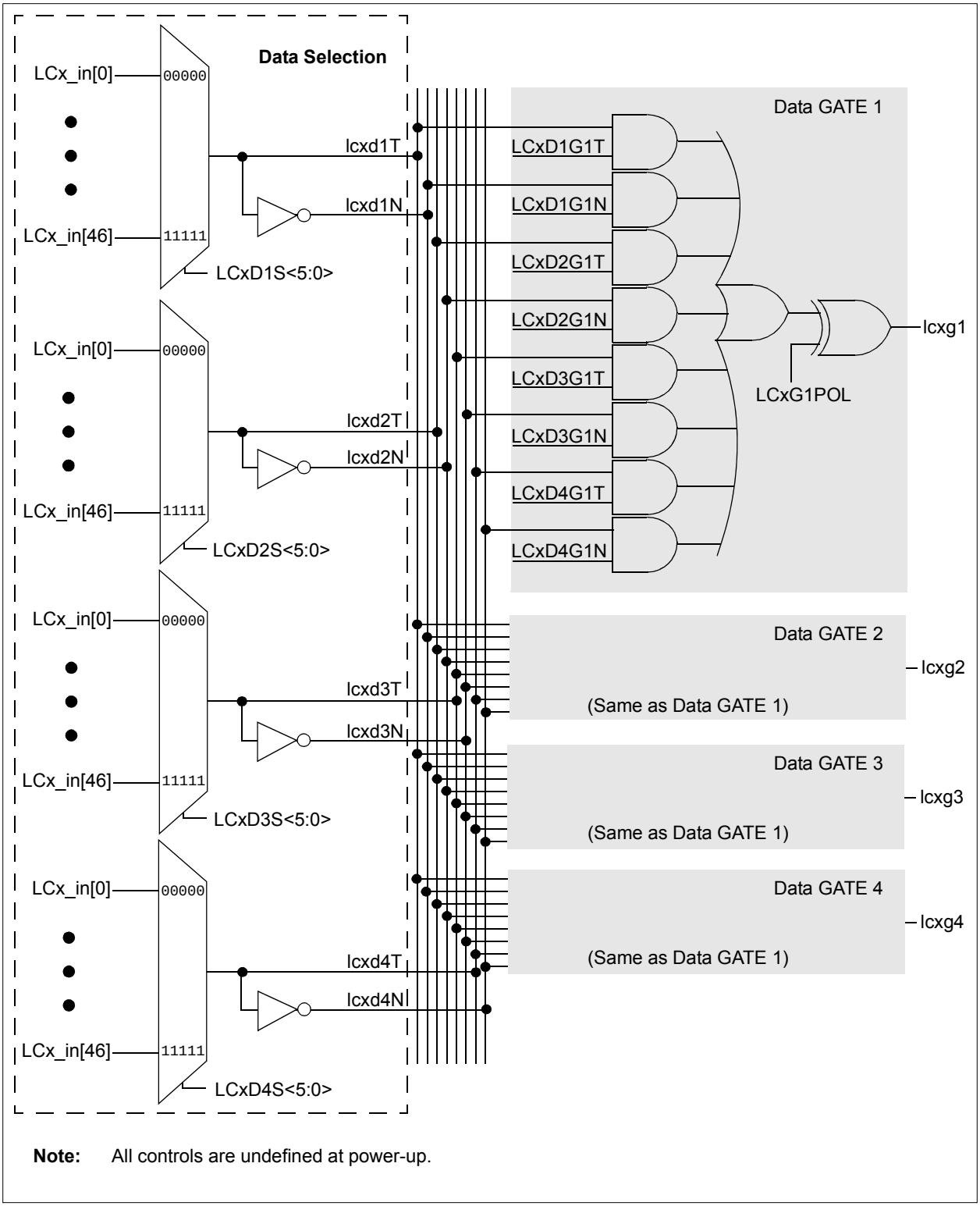
**Note:** The CLC1, CLC2, CLC3 and CLC4 are four separate module instances of the same CLC module design. Throughout this section, the lower case 'x' in register and bit names is a generic reference to the CLC number (which should be substituted with 1, 2, 3, or 4 during code development). For example, the control register is generically described in this chapter as CLCxCON, but the actual device registers are CLC1CON, CLC2CON, CLC3CON and CLC4CON. Similarly, the LCxEN bit represents the LC1EN, LC2EN, LC3EN and LC4EN bits.

Refer to Figure 22-1 for a simplified diagram showing signal flow through the CLCx.

Possible configurations include:

- Combinatorial Logic
  - AND
  - NAND
  - AND-OR
  - AND-OR-INVERT
  - OR-XOR
  - OR-XNOR
- Latches
  - S-R
  - Clocked D with Set and Reset
  - Transparent D with Set and Reset
  - Clocked J-K with Reset

**FIGURE 22-2: INPUT DATA SELECTION AND GATING**




**Note:** All controls are undefined at power-up.

**TABLE 23-1: ADC CLOCK PERIOD (TAD) VS. DEVICE OPERATING FREQUENCIES**

| ADC Clock Period (TAD) | | Device Frequency (FOSC) | | | | | |
|---|---|---|---|---|---|---|---|
| ADC Clock Source | ADCCS<5:0> | 32 MHz | 20 MHz | 16 MHz | 8 MHz | 4 MHz | 1 MHz |
| FOSC/2 | 000000 | 62.5ns(2) | 100 ns(2) | 125 ns(2) | 250 ns(2) | 500 ns(2) | 2.0 μs |
| FOSC/4 | 000001 | 125 ns(2) | 200 ns(2) | 250 ns(2) | 500 ns(2) | 1.0 μs | 4.0 μs |
| FOSC/6 | 000010 | 187.5 ns(2) | 300 ns(2) | 375 ns(2) | 750 ns(2) | 1.5 μs | 6.0 μs |
| FOSC/8 | 000011 | 250 μs(2) | 400 ns(2) | 500 μs(2) | 1.0 μs | 2.0 μs | 8.0 μs(3) |
| ... | ... | ... | ... | ... | ... | ... | ... |
| FOSC/16 | 000111 | 500 ns(2) | 800 ns(2) | 1.0 μs | 2.0 μs | 4.0 μs | 16.0 μs(2) |
| ... | ... | ... | ... | ... | ... | ... | ... |
| FOSC/128 | 111111 | 4.0 μs | 6.4 μs | 8.0 μs | 16.0 μs(3) | 32.0 μs(2) | 128.0 μs(2) |
| FRC | ADCS(ADCON0 <4>)=1 | 1.0-6.0 μs(1) | 1.0-6.0 μs(1) | 1.0-6.0 μs(1) | 1.0-6.0 μs(1) | 1.0-6.0 μs(1) | 1.0-6.0 μs(1) |

**Legend:** Shaded cells are outside of recommended range.

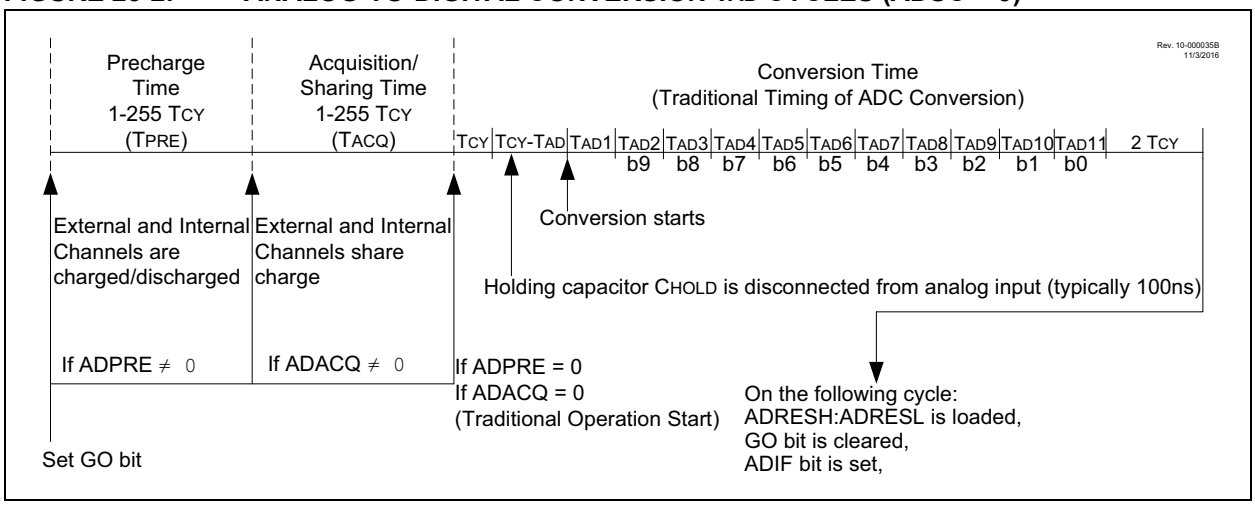
**Note 1:** See TAD parameter for FRC source typical TAD value.

**2:** These values violate the required TAD time.

**3:** Outside the recommended TAD time.

**4:** The ADC clock period (TAD) and total ADC conversion time can be minimized when the ADC clock is derived from the system clock FOSC. However, the FRC oscillator source must be used when conversions are to be performed with the device in Sleep mode.

**FIGURE 23-2: ANALOG-TO-DIGITAL CONVERSION TAD CYCLES (ADSC = 0)**



**REGISTER 27-2: T0CON1: TIMER0 CONTROL REGISTER 1**

| R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| T0CS<2:0> | | | T0ASYNC | T0CKPS<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 Resets |
| ‘1’ = Bit is set | ‘0’ = Bit is cleared | |

bit 7-5 **T0CS<2:0>:** Timer0 Clock Source select bits
111 = Reserved
110 = LC1_out
101 = SOSC
100 = LFINTOSC
011 = HFINTOSC
010 = FOSC/4
001 = T0CKIPPS (Inverted)
000 = T0CKIPPS (True)

bit 4 **T0ASYNC:** TMR0 Input Asynchronization Enable bit
1 = The input to the TMR0 counter is not synchronized to system clocks
0 = The input to the TMR0 counter is synchronized to FOSC/4

bit 3-0 **T0CKPS<3:0>:** Prescaler Rate Select bit
1111 = 1:32768
1110 = 1:16384
1101 = 1:8192
1100 = 1:4096
1011 = 1:2048
1010 = 1:1024
1001 = 1:512
1000 = 1:256
0111 = 1:128
0110 = 1:64
0101 = 1:32
0100 = 1:16
0011 = 1:8
0010 = 1:4
0001 = 1:2
0000 = 1:1

### 31.2.6 SPI OPERATION IN SLEEP MODE

In SPI Master mode, module clocks may be operating at a different speed than when in Full-Power mode; in the case of the Sleep mode, all clocks are halted.

Special care must be taken by the user when the MSSP clock is much faster than the system clock.

In Slave mode, when MSSP interrupts are enabled, after the master completes sending data, an MSSP interrupt will wake the controller from Sleep.

If an exit from Sleep mode is not desired, MSSP interrupts should be disabled.

In SPI Master mode, when the Sleep mode is selected, all module clocks are halted and the transmission/reception will remain in that state until the device wakes. After the device returns to Run mode, the module will resume transmitting and receiving data.

In SPI Slave mode, the SPI Transmit/Receive Shift register operates asynchronously to the device. This allows the device to be placed in Sleep mode and data to be shifted into the SPI Transmit/Receive Shift register. When all eight bits have been received, the MSSP interrupt flag bit will be set and if enabled, will wake the device.

## 31.3 I²C MODE OVERVIEW

The Inter-Integrated Circuit ($I^2C$) bus is a multi-master serial data communication bus. Devices communicate in a master/slave environment where the master devices initiate the communication. A slave device is controlled through addressing.

The $I^2C$ bus specifies two signal connections:

- Serial Clock (SCL)
- Serial Data (SDA)

Figure 31-11 shows the block diagram of the MSSP module when operating in $I^2C$ mode.

Both the SCL and SDA connections are bidirectional open-drain lines, each requiring pull-up resistors for the supply voltage. Pulling the line to ground is considered a logical zero and letting the line float is considered a logical one.

Figure 31-11 shows a typical connection between two processors configured as master and slave devices.

The $I^2C$ bus can operate with one or more master devices and one or more slave devices.

There are four potential modes of operation for a given device:

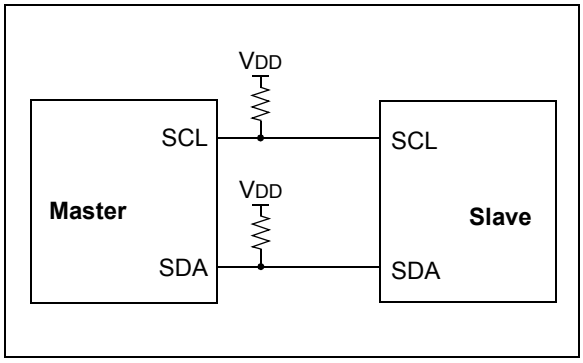
- Master Transmit mode
  (master is transmitting data to a slave)
- Master Receive mode
  (master is receiving data from a slave)
- Slave Transmit mode
  (slave is transmitting data to a master)
- Slave Receive mode
  (slave is receiving data from the master)

To begin communication, a master device starts out in Master Transmit mode. The master device sends out a Start bit followed by the address byte of the slave it intends to communicate with. This is followed by a single Read/Write bit, which determines whether the master intends to transmit to or receive data from the slave device.

If the requested slave exists on the bus, it will respond with an Acknowledge bit, otherwise known as an $\overline{ACK}$. The master then continues in either Transmit mode or Receive mode and the slave continues in the complement, either in Receive mode or Transmit mode, respectively.

A Start bit is indicated by a high-to-low transition of the SDA line while the SCL line is held high. Address and data bytes are sent out, Most Significant bit (MSb) first. The Read/Write bit is sent out as a logical one when the master intends to read data from the slave, and is sent out as a logical zero when it intends to write data to the slave.

**FIGURE 31-11: I²C MASTER/ SLAVE CONNECTION**




The Acknowledge bit ($\overline{ACK}$) is an active-low signal, which holds the SDA line low to indicate to the transmitter that the slave device has received the transmitted data and is ready to receive more.

The transition of a data bit is always performed while the SCL line is held low. Transitions that occur while the SCL line is held high are used to indicate Start and Stop bits.

**REGISTER 31-5: SSPxMSK: SSPx MASK 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 |
|---|---|---|---|---|---|---|---|
| SSPMSK<7:0> | | | | | | | |
| 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 | |

bit 7-1 **SSPMSK<7:1>:** Mask bits
1 = The received address bit n is compared to SSPxADD<n> to detect $I^2C$ address match
0 = The received address bit n is not used to detect $I^2C$ address match

bit 0 **SSPMSK<0>:** Mask bit for $I^2C$ Slave mode, 10-bit Address
$I^2C$ Slave mode, 10-bit address (SSPM<3:0> = 0111 or 1111):
1 = The received address bit 0 is compared to SSPxADD<0> to detect $I^2C$ address match
0 = The received address bit 0 is not used to detect $I^2C$ address match
$I^2C$ Slave mode, 7-bit address:
MSK0 bit is ignored.

**REGISTER 31-6: SSPxADD: MSSPx ADDRESS AND BAUD RATE REGISTER ($I^2C$ MODE)**

| R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| SSPADD<7:0> | | | | | | | |
| 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 | |

**Master mode:**

bit 7-0 **SSPADD<7:0>:** Baud Rate Clock Divider bits
SCL pin clock period = ((ADD<7:0> + 1) *4)/FOSC

**10-Bit Slave mode – Most Significant Address Byte:**

bit 7-3 **Not used:** Unused for Most Significant Address Byte. Bit state of this register is a "don't care". Bit pattern sent by master is fixed by $I^2C$ specification and must be equal to '11110'. However, those bits are compared by hardware and are not affected by the value in this register.

bit 2-1 **SSPADD<2:1>:** Two Most Significant bits of 10-bit Address

bit 0 **Not used:** Unused in this mode. Bit state is a "don't care".

**10-Bit Slave mode – Least Significant Address Byte:**

bit 7-0 **SSPADD<7:0>:** Eight Least Significant bits of 10-bit Address

**7-Bit Slave mode:**

bit 7-1 **SSPADD<7:1>:** 7-bit Address

bit 0 **Not used:** Unused in this mode. Bit state is a "don't care".

**FIGURE 32-10: WINDOWED MEASURE MODE REPEAT ACQUISITION TIMING DIAGRAM**

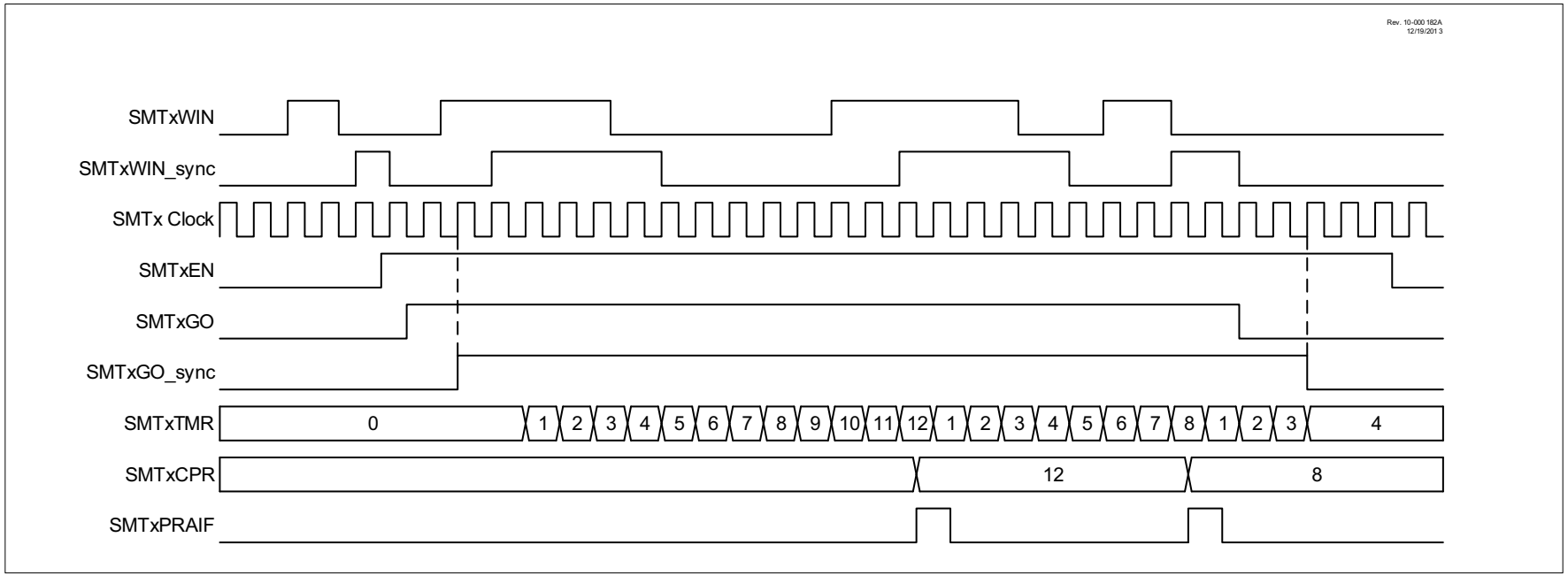
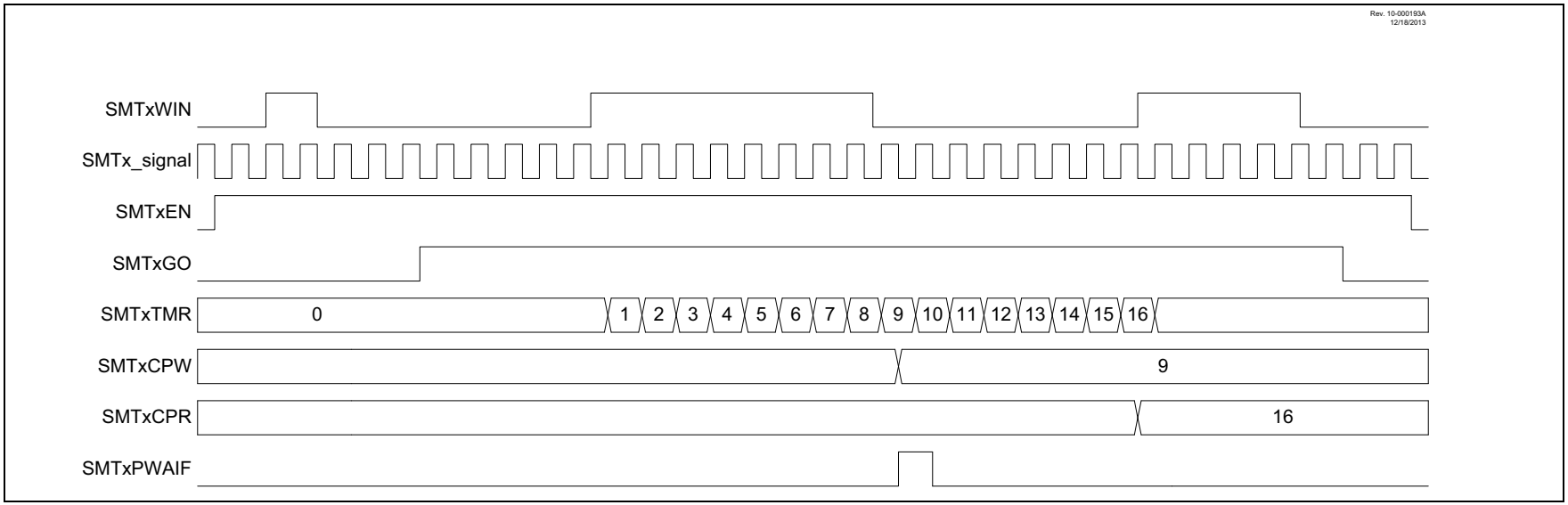


**FIGURE 32-22: WINDOWED COUNTER MODE SINGLE ACQUISITION TIMING DIAGRAM**



**BCF** **Bit Clear f**

| | |
|---|---|
| Syntax: | [ *label* ] BCF f,b |
| Operands: | 0 ≤ f ≤ 127<br>0 ≤ b ≤ 7 |
| Operation: | 0 → (f<b>) |
| Status Affected: | None |
| Description: | Bit 'b' in register 'f' is cleared. |

**BRA** **Relative Branch**

| | |
|---|---|
| Syntax: | [ *label* ] BRA label<br>[ *label* ] BRA $+k |
| Operands: | -256 ≤ label - PC + 1 ≤ 255<br>-256 ≤ k ≤ 255 |
| Operation: | (PC) + 1 + k → PC |
| Status Affected: | None |
| Description: | Add the signed 9-bit literal 'k' to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 1 + k. This instruction is a 2-cycle instruction. This branch has a limited range. |

**BRW** **Relative Branch with W**

| | |
|---|---|
| Syntax: | [ *label* ] BRW |
| Operands: | None |
| Operation: | (PC) + (W) → PC |
| Status Affected: | None |
| Description: | Add the contents of W (unsigned) to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 1 + (W). This instruction is a 2-cycle instruction. |

**BSF** **Bit Set f**

| | |
|---|---|
| Syntax: | [ *label* ] BSF f,b |
| Operands: | 0 ≤ f ≤ 127<br>0 ≤ b ≤ 7 |
| Operation: | 1 → (f<b>) |
| Status Affected: | None |
| Description: | Bit 'b' in register 'f' is set. |

**BTFSC** **Bit Test f, Skip if Clear**

| | |
|---|---|
| Syntax: | [ *label* ] BTFSC f,b |
| Operands: | 0 ≤ f ≤ 127<br>0 ≤ b ≤ 7 |
| Operation: | skip if (f<b>) = 0 |
| Status Affected: | None |
| Description: | If bit 'b' in register 'f' is '1', the next instruction is executed.<br>If bit 'b', in register 'f', is '0', the next instruction is discarded, and a NOP is executed instead, making this a 2-cycle instruction. |

**BTFSS** **Bit Test f, Skip if Set**

| | |
|---|---|
| Syntax: | [ *label* ] BTFSS f,b |
| Operands: | 0 ≤ f ≤ 127<br>0 ≤ b < 7 |
| Operation: | skip if (f<b>) = 1 |
| Status Affected: | None |
| Description: | If bit 'b' in register 'f' is '0', the next instruction is executed.<br>If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead, making this a 2-cycle instruction. |

**CALL** **Call Subroutine**

| | |
|---|---|
| Syntax: | [ *label* ] CALL k |
| Operands: | $0 \le k \le 2047$ |
| Operation: | (PC)+ 1→ TOS,<br>k → PC<10:0>,<br>(PCLATH<6:3>) → PC<14:11> |
| Status Affected: | None |
| Description: | Call Subroutine. First, return address (PC + 1) is pushed onto the stack. The 11-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a 2-cycle instruction. |

**CALLW** **Subroutine Call With W**

| | |
|---|---|
| Syntax: | [ *label* ] CALLW |
| Operands: | None |
| Operation: | (PC) +1 → TOS,<br>(W) → PC<7:0>,<br>(PCLATH<6:0>) → PC<14:8> |
| Status Affected: | None |
| Description: | Subroutine call with W. First, the return address (PC + 1) is pushed onto the return stack. Then, the contents of W is loaded into PC<7:0>, and the contents of PCLATH into PC<14:8>. CALLW is a 2-cycle instruction. |

**CLRF** **Clear f**

| | |
|---|---|
| Syntax: | [ *label* ] CLRF f |
| Operands: | $0 \le f \le 127$ |
| Operation: | 00h → (f)<br>1 → Z |
| Status Affected: | Z |
| Description: | The contents of register ‘f’ are cleared and the Z bit is set. |

**CLRW** **Clear W**

| | |
|---|---|
| Syntax: | [ *label* ] CLRW |
| Operands: | None |
| Operation: | 00h → (W)<br>1 → Z |
| Status Affected: | Z |
| Description: | W register is cleared. Zero bit (Z) is set. |

**CLRWDT** **Clear Watchdog Timer**

| | |
|---|---|
| Syntax: | [ *label* ] CLRWDT |
| Operands: | None |
| Operation: | 00h → WDT<br>0 → WDT prescaler,<br>1 → $\overline{TO}$<br>1 → $\overline{PD}$ |
| Status Affected: | $\overline{TO}$, $\overline{PD}$ |
| Description: | CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set. |

**COMF** **Complement f**

| | |
|---|---|
| Syntax: | [ *label* ] COMF f,d |
| Operands: | $0 \le f \le 127$<br>$d \in [0,1]$ |
| Operation: | ($\bar{f}$) → (destination) |
| Status Affected: | Z |
| Description: | The contents of register ‘f’ are complemented. If ‘d’ is ‘0’, the result is stored in W. If ‘d’ is ‘1’, the result is stored back in register ‘f’. |

**DECF** **Decrement f**

| | |
|---|---|
| Syntax: | [ *label* ] DECF f,d |
| Operands: | $0 \le f \le 127$<br>$d \in [0,1]$ |
| Operation: | (f) - 1 → (destination) |
| Status Affected: | Z |
| Description: | Decrement 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’. |

**TABLE 37-3: POWER-DOWN CURRENT ($I_{PD}$)(1,2)**

| PIC16LF18856/76 | | | | Standard Operating Conditions (unless otherwise stated) | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| PIC16F18856/76 | | | | Standard Operating Conditions (unless otherwise stated)<br>VREGPM = 1 | | | | | |
| Param. No. | Symbol | Device Characteristics | Min. | Typ.† | Max. +85°C | Max. +125°C | Units | Conditions | |
| | | | | | | | | $V_{DD}$ | Note |
| D200 | $I_{PD}$ | $I_{PD}$ Base | — | 0.05 | 2 | 9 | µA | 3.0V | |
| D200<br>D200A | $I_{PD}$ | $I_{PD}$ Base | — | 0.4 | 4 | 12 | µA | 3.0V | |
| | | | — | 10 | 15 | 20 | µA | 3.0V | VREGPM = 0 |
| D201 | $I_{PD\_WDT}$ | Low-Frequency Internal Oscillator/WDT | — | 0.4 | — | — | µA | 3.0V | |
| D201 | $I_{PD\_WDT}$ | Low-Frequency Internal Oscillator/WDT | — | 0.6 | 5 | 13 | µA | 3.0V | |
| D202 | $I_{PD\_SOSC}$ | Secondary Oscillator (SOSC) | — | 0.6 | 5 | 13 | µA | 3.0V | |
| D202 | $I_{PD\_SOSC}$ | Secondary Oscillator (SOSC) | — | 0.8 | 8.5 | 15 | µA | 3.0V | |
| D203 | $I_{PD\_FVR}$ | FVR | — | 8 | 18 | — | µA | 3.0V | |
| D203 | $I_{PD\_FVR}$ | FVR | — | 10 | 20 | — | µA | 3.0V | |
| D204 | $I_{PD\_BOR}$ | Brown-out Reset (BOR) | — | 9 | 14 | 18 | µA | 3.0V | |
| D204 | $I_{PD\_BOR}$ | Brown-out Reset (BOR) | — | 14 | 19 | 21 | µA | 3.0V | |
| D205 | $I_{PD\_LPBOR}$ | Low-Power Brown-out Reset (LPBOR) | — | 0.5 | — | — | µA | 3.0V | |
| D205 | $I_{PD\_LPBOR}$ | Low-Power Brown-out Reset (LPBOR) | — | 0.7 | — | — | µA | 3.0V | |
| D206 | $I_{PD\_ADCA}$ | ADC - Active | — | 250 | — | — | µA | 3.0V | ADC is converting (4) |
| D206 | $I_{PD\_ADCA}$ | ADC - Active | — | 280 | — | — | µA | 3.0V | ADC is converting (4) |
| D207 | $I_{PD\_CMP}$ | Comparator | — | 25 | 38 | 40 | µA | 3.0V | |
| D207 | $I_{PD\_CMP}$ | Comparator | — | 28 | 40 | 50 | µA | 3.0V | |

† 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:** The peripheral current is the sum of the base $I_{DD}$ and the additional current consumed when this peripheral is enabled. The peripheral Δ current can be determined by subtracting the base $I_{DD}$ or $I_{PD}$ current from this limit. Max. values should be used when calculating total current consumption.

**2:** The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode with all I/O pins in high-impedance state and tied to $V_{SS}$.

**3:** All peripheral currents listed are on a per-peripheral basis if more than one instance of a peripheral is available.

**4:** ADC clock source is FRC.

**Note:** Unless otherwise noted, VIN = 5V, FOSC = 300 kHz, CIN = 0.1 µF, TA = 25°C.

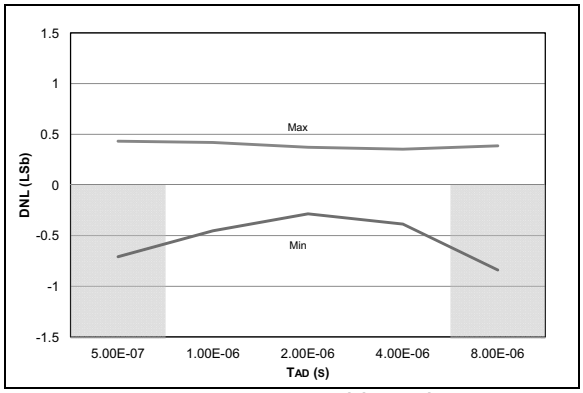



***FIGURE 38-37:*** *ADC 10-bit Mode, Single-Ended DNL, VDD = 3.0V, VREF = 3.0V.*

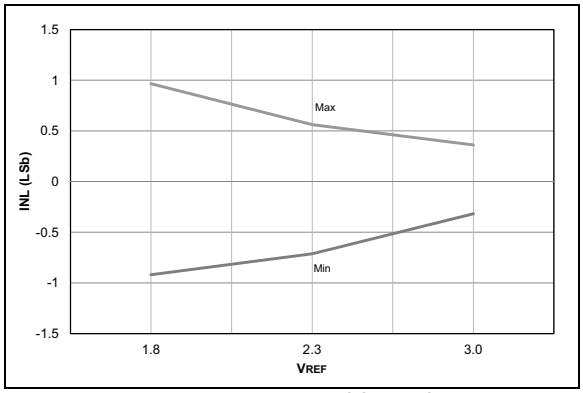



***FIGURE 38-40:*** *ADC 10-bit Mode, Single-Ended INL, VDD = 3.0V, TAD = 1 µS.*

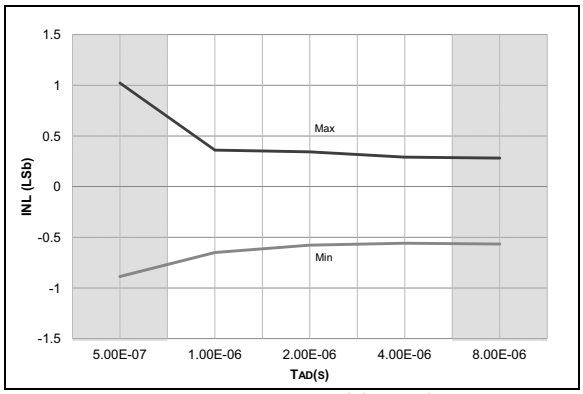



***FIGURE 38-38:*** *ADC 10-bit Mode, Single-Ended INL, VDD = 3.0V, VREF = 3.0V.*

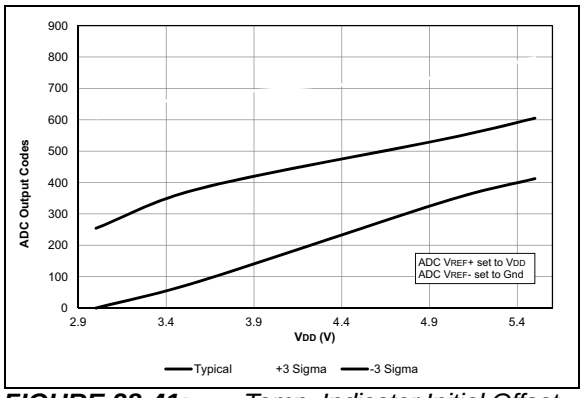



***FIGURE 38-41:*** *Temp. Indicator Initial Offset, High Range, Temp. = 20°C, PIC16F18856/76 Only.*

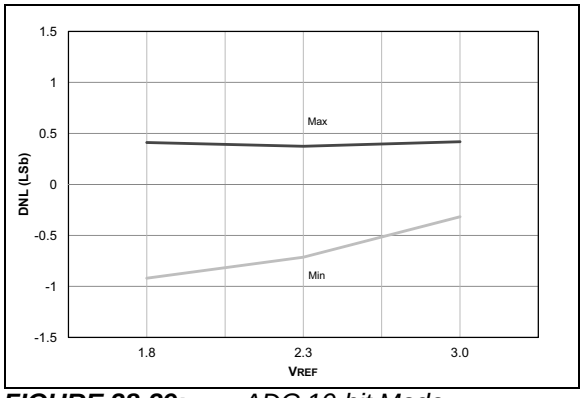



***FIGURE 38-39:*** *ADC 10-bit Mode, Single-Ended DNL, VDD = 3.0V, TAD = 1 µS.*

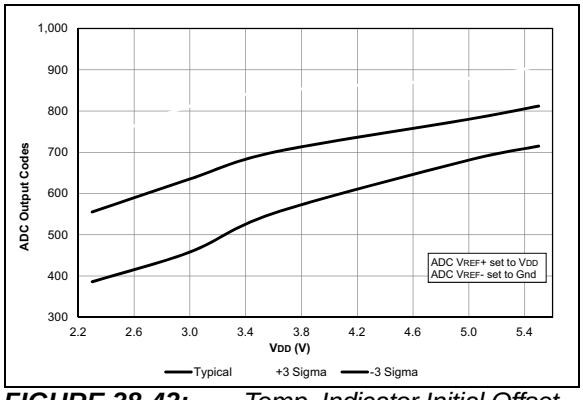



***FIGURE 38-42:*** *Temp. Indicator Initial Offset, Low Range, Temp. = 20°C, PIC16F18856/76 Only.*

**Note:** Unless otherwise noted, VIN = 5V, FOSC = 300 kHz, CIN = 0.1 µF, TA = 25°C.

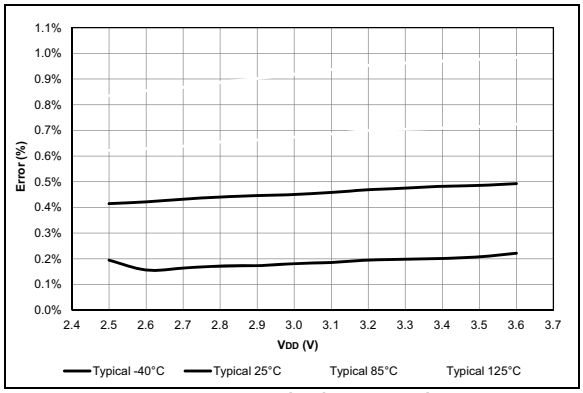



***FIGURE 38-73:*** *Typical FVR Voltage 1x, PIC16LF18856/76 Only.*

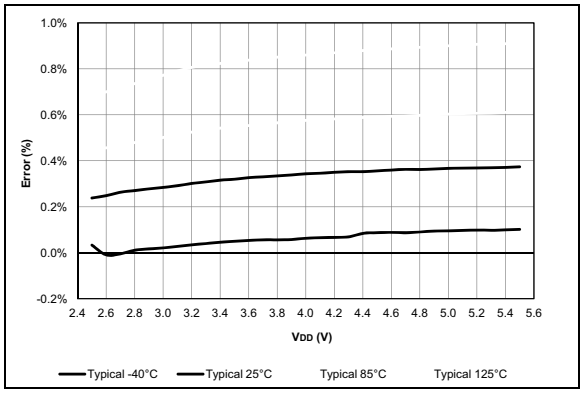



***FIGURE 38-76:*** *FVR Voltage Error 2x, PIC16F18856/76 Only.*

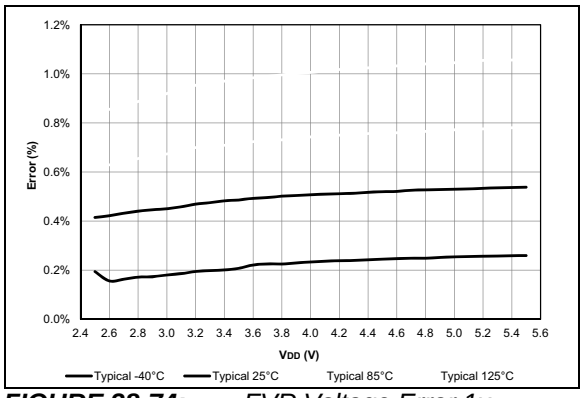



***FIGURE 38-74:*** *FVR Voltage Error 1x, PIC16F18856/76 Only.*

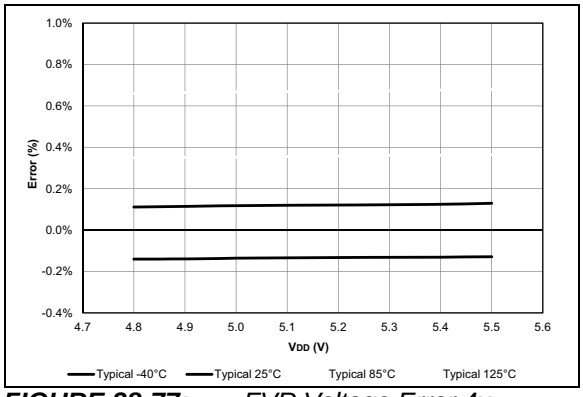



***FIGURE 38-77:*** *FVR Voltage Error 4x, PIC16F18856/76 Only.*

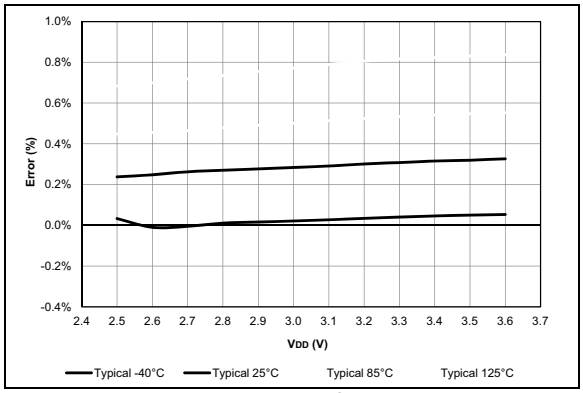



***FIGURE 38-75:*** *FVR Voltage Error 2x, PIC16LF18856/76 Only.*

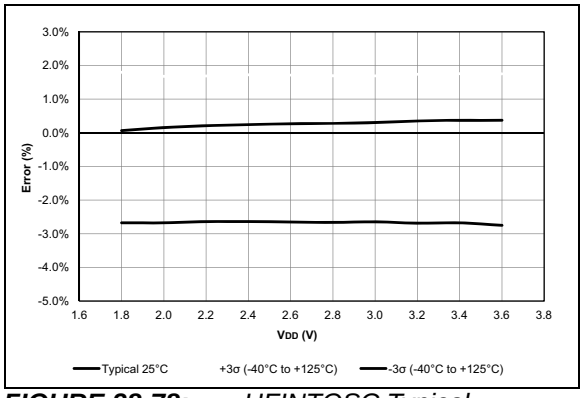
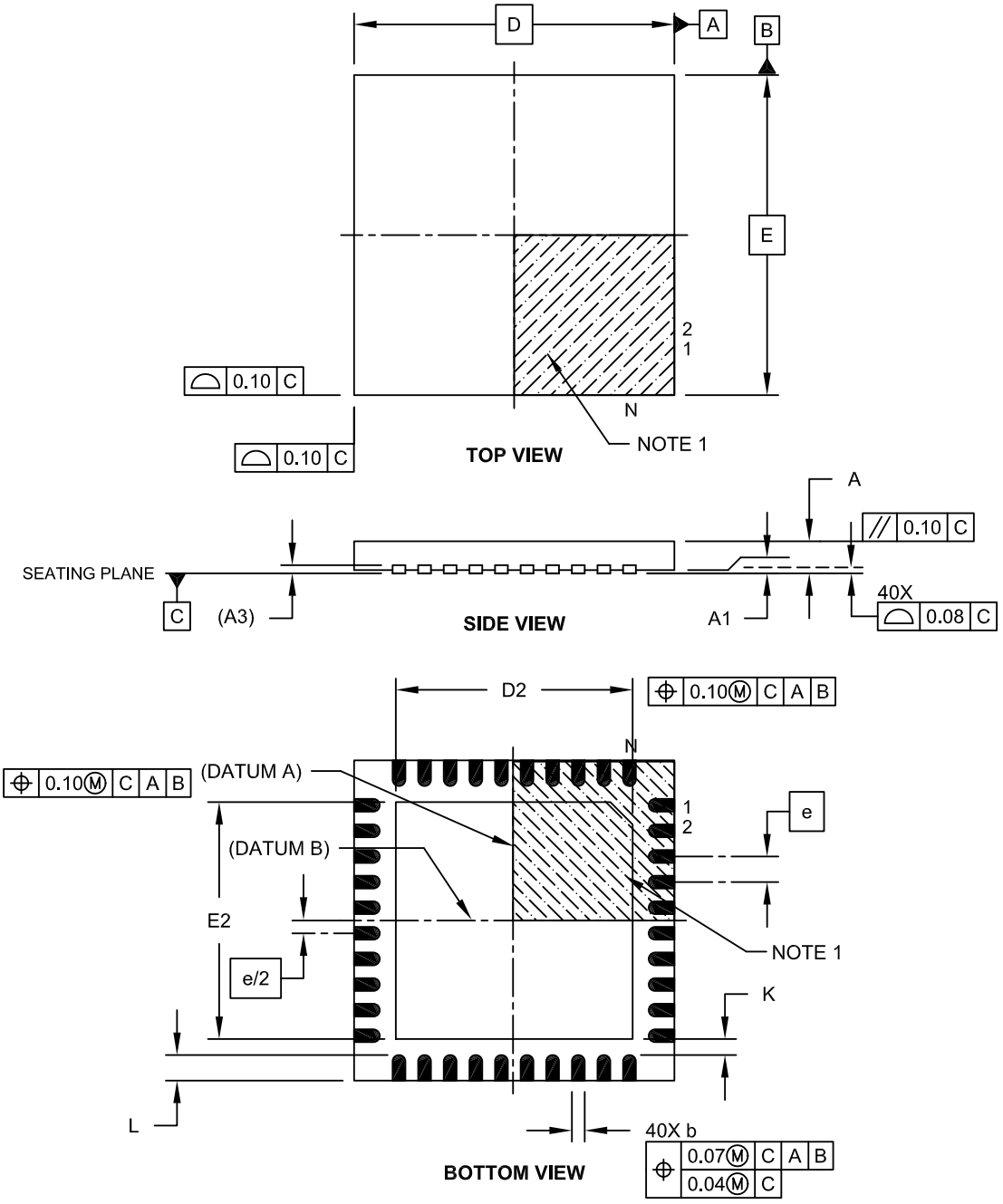



***FIGURE 38-78:*** *HFINTOSC Typical Frequency Error, PIC16LF18856/76 Only.*

### 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