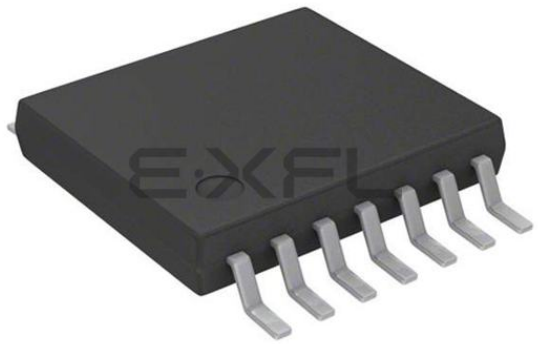


**Welcome to E-XFL.COM**

## 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 | 12 |
| 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) | 2.3V ~ 5.5V |
| Data Converters | A/D 11x12b; D/A 1x5b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 14-TSSOP (0.173", 4.40mm Width) |
| Supplier Device Package | 14-TSSOP |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16f18426-i-st |

## 3.2 16-Level Stack with Overflow and Underflow

These devices have a hardware stack memory 15 bits wide and 16 words deep. A Stack Overflow or Underflow will set the appropriate bit (STKOVF or STKUNF) in the PCON0 register, and if enabled, will cause a software Reset.

**Related Links**

Stack

PCON0

## 3.3 File Select Registers

There are two 16-bit File Select Registers (FSR). FSRs can access all file registers and program memory, which allows one Data Pointer for all memory. When an FSR points to program memory, there is one additional instruction cycle in instructions using INDF to allow the data to be fetched. General purpose memory can also be addressed linearly, providing the ability to access contiguous data larger than 80 bytes.

**Related Links**

Indirect Addressing

## 3.4 Instruction Set

There are 50 instructions for the enhanced mid-range CPU to support the features of the CPU.

**Related Links**

Instruction Set Summary

bit of the PIR0 register will be set when a valid edge appears on the INT pin. If the GIE and INTE bits are also set, the processor will redirect program execution to the interrupt vector.

## 10.5 Automatic Context Saving

Upon entering an interrupt, the return PC address is saved on the stack. Additionally, the following registers are automatically saved in the shadow registers:

- W register
- STATUS register (except for $\overline{TO}$ and $\overline{PD}$)
- BSR register
- FSR registers
- PCLATH register

Upon exiting the Interrupt Service Routine, these registers are automatically restored. Any modifications to these registers during the ISR will be lost. If modifications to any of these registers are desired, the corresponding shadow register should be modified and the value will be restored when exiting the ISR. The shadow registers are available in Bank 63 and are readable and writable. Depending on the user's application, other registers may also need to be saved.

**Related Links**

Register Definitions: Shadow Registers

### 10.7.15 PIR4

**Name:** PIR4
**Address:** 0x710

Peripheral Interrupt Request (Flag) Register 4

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | TMR6IF | TMR5IF | TMR4IF | TMR3IF | TMR2IF | TMR1IF |
| Access | | | R/W/HS | R/W/HS | R/W/HS | R/W/HS | R/W/HS | R/W/HS |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 5 – TMR6IF** TMR6 to PR6 Match Interrupt Flag bit

| Value | Description |
|---|---|
| 1 | The TMR6 postscaler overflowed, or in 1:1 mode, a TMR6 to PR6 match occurred (must be cleared in software) |
| 0 | No TMR6 event has occurred |

**Bit 4 – TMR5IF** TMR5 Overflow Interrupt Flag bit

| Value | Description |
|---|---|
| 1 | TMR5 register overflowed (must be cleared in software) |
| 0 | TMR5 register did not overflow |

**Bit 3 – TMR4IF** TMR4 to PR4 Match Interrupt Flag bit

| Value | Description |
|---|---|
| 1 | The TMR4 postscaler overflowed, or in 1:1 mode, a TMR4 to PR4 match occurred (must be cleared in software) |
| 0 | No TMR4 event has occurred |

**Bit 2 – TMR3IF** TMR3 Overflow Interrupt Flag bit

| Value | Description |
|---|---|
| 1 | TMR3 register overflowed (must be cleared in software) |
| 0 | TMR3 register did not overflow |

**Bit 1 – TMR2IF** TMR2 to PR2 Match Interrupt Flag bit

| Value | Description |
|---|---|
| 1 | The TMR2 postscaler overflowed, or in 1:1 mode, a TMR2 to PR2 match occurred (must be cleared in software) |
| 0 | No TMR2 event has occurred |

**Bit 0 – TMR1IF** TMR1 Overflow Interrupt Flag bit

| Value | Description |
|---|---|
| 1 | TMR1 register overflowed (must be cleared in software) |
| 0 | TMR1 register did not overflow |

The Low-Power Sleep mode is beneficial for applications that stay in Sleep mode for long periods of time. The Normal mode is beneficial for applications that need to wake from Sleep quickly and frequently.

#### 11.2.3.2 Peripheral Usage in Sleep

Some peripherals that can operate in Sleep mode will not operate properly with the Low-Power Sleep mode selected. The Low-Power Sleep mode is intended for use with these peripherals:

- Brown-out Reset (BOR)
- Windowed Watchdog Timer (WWDT)
- External interrupt pin/Interrupt-On-Change pins
- Timer1 (with external clock source)

It is the responsibility of the end user to determine what is acceptable for their application when setting the VREGPM settings in order to ensure operation in Sleep.

**Important:** The LF devices do not have a configurable Low-Power Sleep mode. LFs are unregulated devices and are always in the lowest power state when in Sleep, with no wake-up time penalty. These devices have a lower maximum $V_{DD}$ and I/O voltage than the F devices.

## 11.3 Idle Mode

When the Idle Enable (IDLEN) bit is clear (IDLEN = 0), the SLEEP instruction will put the device into full Sleep mode. When IDLEN is set (IDLEN = 1), the SLEEP instruction will put the device into IDLE mode. In IDLE mode, the CPU and memory operations are halted, but the peripheral clocks continue to run. This mode is similar to DOZE mode, except that in IDLE both the CPU and program memory are shut off.

**Important:** Peripherals using $F_{OSC}$ will continue running while in Idle (but not in Sleep). Peripherals using HFINTOSC:LFINTOSC will continue running in both Idle and Sleep.

**Important:** If $\overline{\text{CLKOUTEN}}$ is enabled ($\overline{\text{CLKOUTEN}}$ = 0, Configuration Word 1), the output will continue operating while in Idle.

### 11.3.1 Idle and Interrupts

IDLE mode ends when an interrupt occurs (even if GIE = 0), but IDLEN is not changed. The device can re-enter IDLE by executing the SLEEP instruction.

If Recover-On-Interrupt is enabled (ROI = 1), the interrupt that brings the device out of Idle also restores full-speed CPU execution when doze is also enabled.

### 11.3.2 Idle and WWDT

When in Idle, the WWDT Reset is blocked and will instead wake the device. The WWDT wake-up is not an interrupt, therefore ROI does not apply.

- Write of Program Flash Memory write latches to program memory
- Write of Program Flash Memory write latches to User IDs
- Write to EEPROM

The unlock sequence consists of the following steps and must be completed in order:

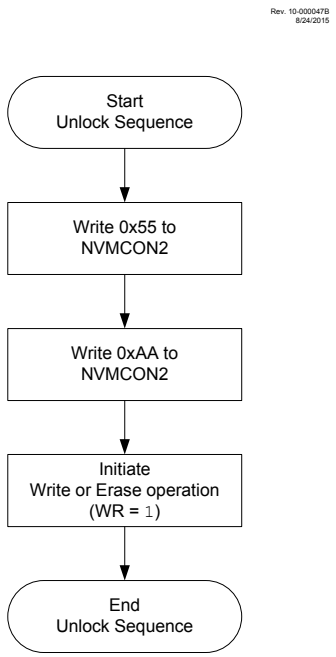
- Write 55h to NVMCON2
- Write AAh to NMVCON2
- Set the WR bit of NVMCON1

Once the WR bit is set, the processor will stall internal operations until the operation is complete and then resume with the next instruction.

**Important:** The two `NOP` instructions after setting the WR bit that were required in previous devices are not required for PIC16(L)F184XX devices. See figure below.

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

**Figure 13-2. NVM Unlock Sequence Flowchart**




**NVM Unlock Sequence**

```
BCF      INTCON, GIE       ; Recommended so sequence is not interrupted
BANKSEL  NVMCON1           ;
BSF      NVMCON1, WREN     ; Enable write/erase
MOVLW    55h               ; Load 55h
MOVWF    NVMCON2           ; Step 1: Load 55h into NVMCON2
MOVLW    AAh               ; Step 2: Load W with AAh
MOVWF    NVMCON2           ; Step 3: Load AAH into NVMCON2
```

```
BANKSEL     NVMADRL
MOVF        ADDRL,W
MOVWF       NVMADRL           ; Load lower 8 bits of erase address boundary
MOVF        ADDRH,W
MOVWF       NVMADRH           ; Load upper 6 bits of erase address boundary
BCF         NVMCON1,NVMREGS   ; Choose PFM memory area
BSF         NVMCON1,FREE      ; Specify an erase operation
BSF         NVMCON1,WREN      ; Enable writes
BCF         INTCON,GIE        ; Disable interrupts during unlock sequence

; ---------------------REQUIRED UNLOCK SEQUENCE:--------------------

MOVLW       55h               ; Load 55h to get ready for unlock sequence
MOVWF       NVMCON2           ; First step is to load 55h into NVMCON2
MOVLW       AAh               ; Second step is to load AAh into W
MOVWF       NVMCON2           ; Third step is to load AAh into NVMCON2
BSF         NVMCON1,WR        ; Final step is to set WR bit

; ---------------------------------------------------------------------

BSF         INTCON,GIE        ; Re-enable interrupts, erase is complete
BCF         NVMCON1,WREN      ; Disable writes
```

**Table 13-1. NVM Organization and Access Information**

| Master Values | | | | NVMREG Access | | | FSR Access | |
|---|---|---|---|---|---|---|---|---|
| Memory Function | Memory Type | Program Counter (PC) | ICSP Address | NVMREGS bit (NVMCON1) | NVMADR<14:0> | Allowed Operations | FSR Address | FSR Programming Access |
| RESET VECTOR | Program Flash Memory | 0000h | 0000h | 0 | 0000h | READ/ WRITE | 8000h | READ ONLY |
| USER MEMORY | | 0001h | 0001h | 0 | 0001h | | 8001h | |
| | | 0003h | 0003h | | 0003h | | 8003h | |
| INT VECTOR | | 0004h | 0004h | 0 | 0004h | | 8004h | |
| USER MEMORY | | 0005h | 0005h | 0 | 0005h | | 8005h | |
| | | 7FFFh(1) | 7FFFh(1) | | 7FFFh(1) | | FFFFh | |
| USER ID | Program Flash Memory | NO PC ACCESS | 8000h | 1 | 0000h | READ | NO ACCESS | |
| | | | 8003h | | 0003h | | | |
| Reserved | — | | — | — | 0004h | — | | |
| REV ID | HC | | 8005h | 1 | 0005h | READ | | |
| DEVICE ID | | | 8006h | 1 | 0006h | | | |
| CONFIG1 | FUSE | | 8007h | 1 | 0007h | READ/ WRITE | | |
| CONFIG2 | | | 8008h | 1 | 0008h | | | |
| CONFIG3 | | | 8009h | 1 | 0009h | | | |
| CONFIG4 | | | 800Ah | 1 | 000Ah | | | |
| CONFIG5 | | | 800Bh | 1 | 000Bh | | | |
| DIA and DCI | PFM | | 8100h | 1 | 0100h | READ | | |
| | | | 82FFh | 1 | 02FFh | | | |
| USER MEMORY | EEPROM | | F000h | 1 | 7000h | READ/ WRITE | 7000h | READ ONLY |
| | | | F0FFh | | 70FFh | | 70FFh | |

9. Clear the LWLO bit of the NVMCON1 register. When the LWLO bit of the NVMCON1 register is '0', the write sequence will initiate the write to Flash program memory.
10. Load the NVMDATH:NVMDATL register pair with the program memory data to be written.
11. Execute the unlock sequence. The entire program memory latch content is now written to Flash program memory.

**Important:** The program memory write latches are reset to the blank state (0x7FFF) at the completion of every write or erase operation. As a result, it is not necessary to load all the program memory write latches. Unloaded latches will remain in the blank state.

An example of the complete write sequence is shown in Writing to Program Flash Memory. The initial address is loaded into the NVMADRH:NVMADRL register pair; the data is loaded using indirect addressing.

**Figure 13-4. NVMREG Writes to Program Flash Memory With 32 Write Latches**

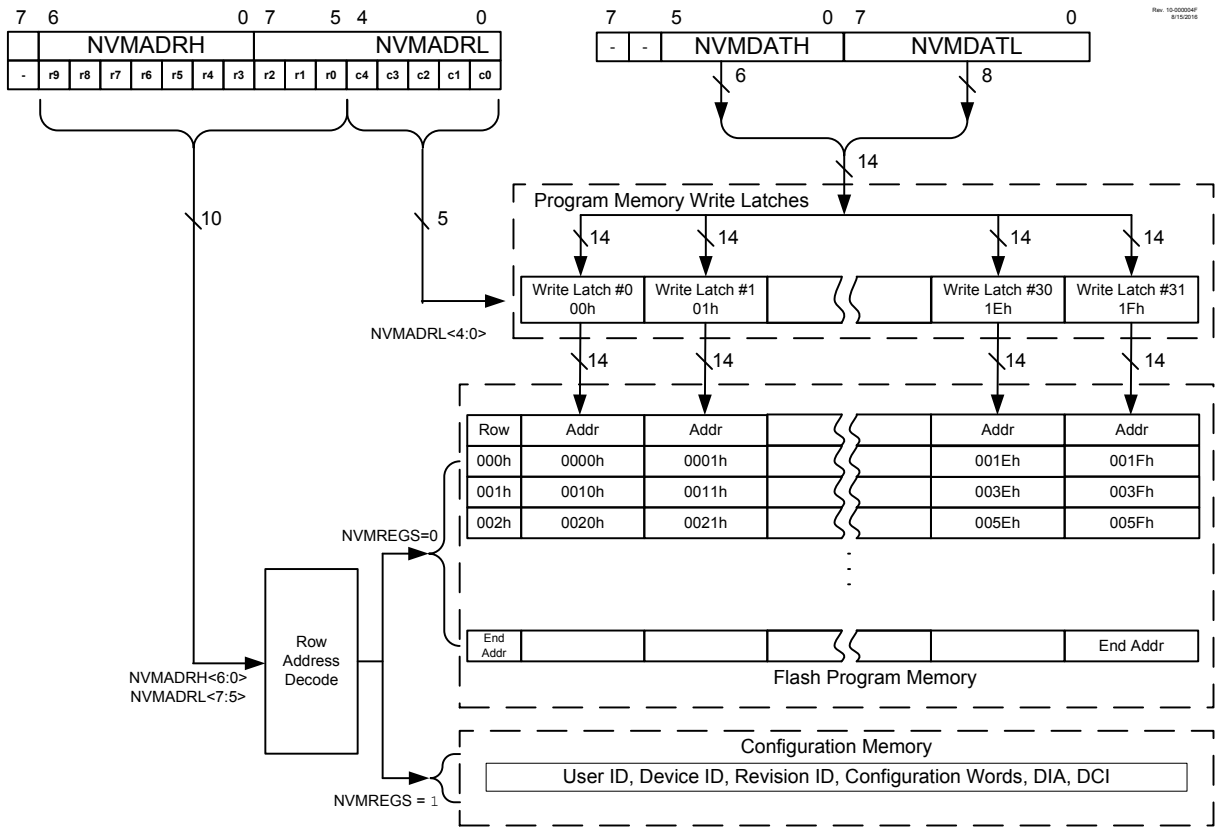
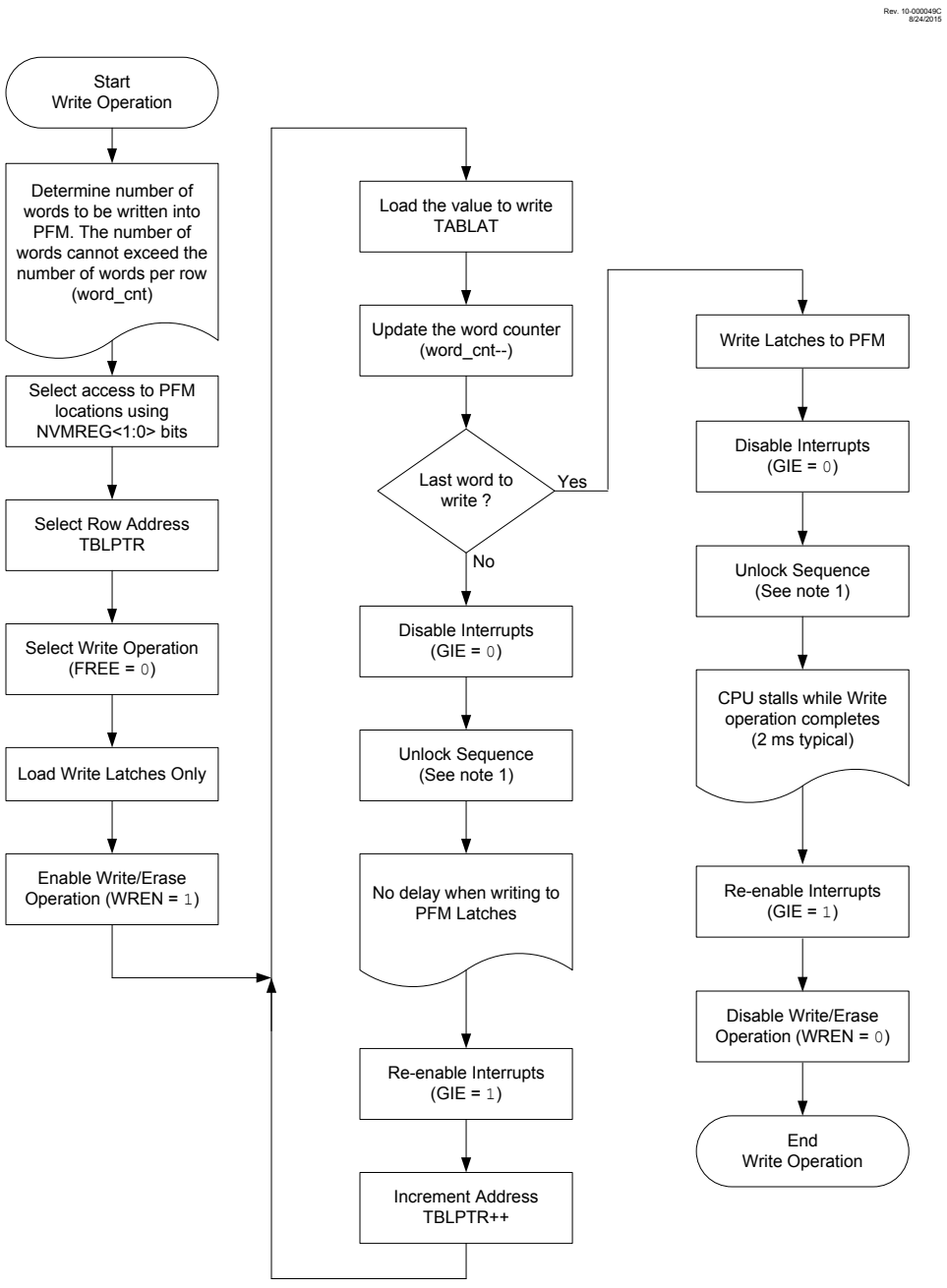


**Figure 13-5. Program Flash Memory Flowchart**




**Note:**

1. See NVM Unlock Sequence Flowchart

### Writing to Program Flash Memory

```
   ; This write routine assumes the following:
   ; 1. 64 bytes of data are loaded, starting at the address in DATA_ADDR
   ; 2. Each word of data to be written is made up of two adjacent bytes in
DATA_ADDR,
   ; stored in little endian format
   ; 3. A valid starting address (the least significant bits = 00000) is
```

# 14. I/O Ports

## 14.1 PORT Availability

**Table 14-1. PORT Availability Per Device**

| PORTs | PORT Description | PIC16(L)F18426 | PIC16(L)F18446 |
|---|---|---|---|
| PORTA | 6-bit wide, bidirectional port. | • | • |
| PORTB | 4-bit wide, bidirectional port. | | • |
| PORTC | 6/8-bit wide, bidirectional port. | • | • |

## 14.2 I/O Ports Description

Each port has standard registers for its operation. These registers are:

- PORTx registers (reads the levels on the pins of the device)
- LATx registers (output latch)
- TRISx registers (data direction)
- ANSELx registers (analog select)
- WPUx registers (weak pull-up)
- INLVLx (input level control)
- SLRCONx registers (slew rate control)
- ODCONx registers (open-drain control)

Most port pins share functions with device peripherals, both analog and digital. In general, when a peripheral is enabled on a port pin, that pin cannot be used as a general purpose output; however, the pin can still be read.

The Data Latch (LATx registers) is useful for read-modify-write operations on the value that the I/O pins are driving.

A write operation to the LATx register has the same effect as a write to the corresponding PORTx register. A read of the LATx register reads of the values held in the I/O PORT latches, while a read of the PORTx register reads the actual I/O pin value.

Ports that support analog inputs have an associated ANSELx register. When an ANSELx bit is set, the digital input buffer associated with that bit is disabled.

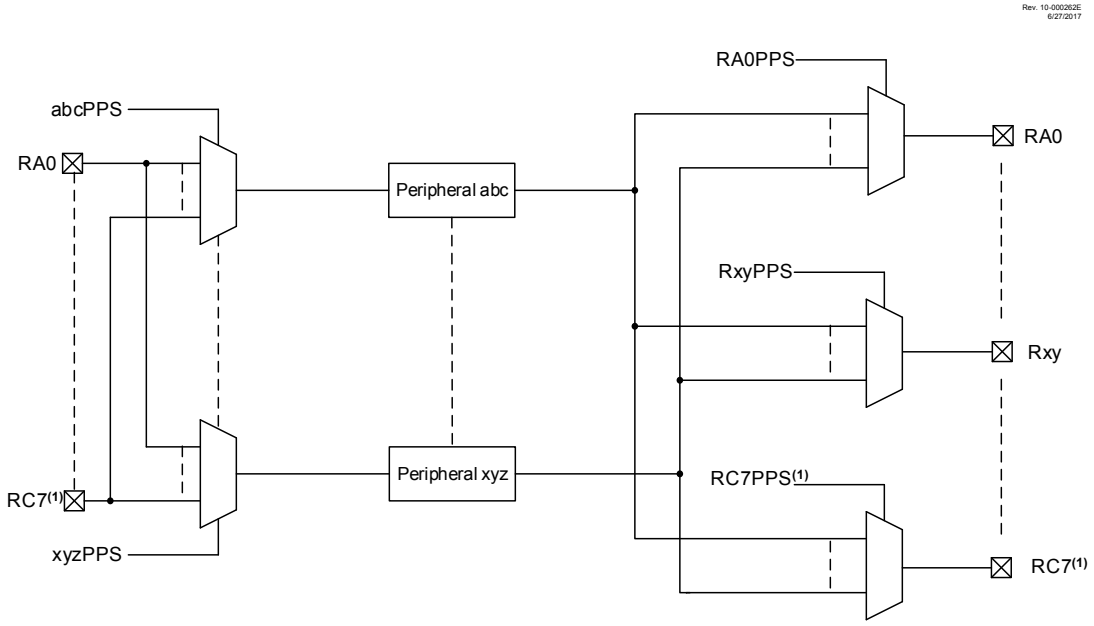
Disabling the input buffer prevents analog signal levels on the pin between a logic high and low from causing excessive current in the logic input circuitry. A simplified model of a generic I/O port, without the interfaces to other peripherals, is shown in the following figure:

# 15. (PPS) Peripheral Pin Select Module

The Peripheral Pin Select (PPS) module connects peripheral inputs and outputs to the device I/O pins. Only digital signals are included in the selections.

All analog inputs and outputs remain fixed to their assigned pins. Input and output selections are independent as shown in the figure below.

**Figure 15-1. Simplified PPS Block Diagram**




**Note:**

1. Not present on 14-pin devices.

## 15.1 PPS Inputs

Each peripheral has an xxxPPS register with which the input pin to the peripheral is selected. Not all ports are available for input as shown in the following table.

Multiple peripherals can operate from the same source simultaneously. Port reads always return the pin level regardless of peripheral PPS selection. If a pin also has analog functions associated, the ANSEL bit for that pin must be cleared to enable the digital input buffer.

**Important:** The notation “xxx” in the generic register name is a place holder for the peripheral identifier. For example, xxx = INT for the INTPPS register.

### 22.9.2 NCOxCLK

**Name:** NCOxCLK
**Address:** 0x0593

NCO Input Clock Control Register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PWS[2:0] | | | | CKS[3:0] | | | |
| Access | R/W | R/W | R/W | | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | | 0 | 0 | 0 | 0 |

**Bits 7:5 – PWS[2:0]** NCO Output Pulse Width Select bits(1)

| Value | Description |
|---|---|
| 111 | NCO output is active for 128 input clock periods |
| 110 | NCO output is active for 64 input clock periods |
| 101 | NCO output is active for 32 input clock periods |
| 100 | NCO output is active for 16 input clock periods |
| 011 | NCO output is active for 8 input clock periods |
| 010 | NCO output is active for 4 input clock periods |
| 001 | NCO output is active for 2 input clock periods |
| 000 | NCO output is active for 1 input clock periods |

**Bits 3:0 – CKS[3:0]** NCO Clock Source Select bits
CKS values are available in the NCO Clock Sources table.

**Note:**

1. PWS applies only when operating in Pulse Frequency mode.

### 23.15.1 CMxCON0

**Name:** CMxCON0
**Address:** 0x990,0x994

Comparator x Control Register 0

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EN | OUT | | POL | | | HYS | SYNC |
| Access | R/W | RO | | R/W | | | R/W | R/W |
| Reset | 0 | 0 | | 0 | | | 0 | 0 |

**Bit 7 – EN** Comparator Enable bit

| Value | Description |
|---|---|
| 1 | Comparator is enabled |
| 0 | Comparator is disabled and consumes no active power |

**Bit 6 – OUT** Comparator Output bit

| Value | Condition | Description |
|---|---|---|
| 1 | If POL = 0 (non-inverted polarity): | CxVP > CxVN |
| 0 | If POL = 0 (non-inverted polarity): | CxVP < CxVN |
| 1 | If POL = 1 (inverted polarity): | CxVP < CxVN |
| 0 | If POL = 1 (inverted polarity): | CxVP > CxVN |

**Bit 4 – POL** Comparator Output Polarity Select bit

| Value | Description |
|---|---|
| 1 | Comparator output is inverted |
| 0 | Comparator output is not inverted |

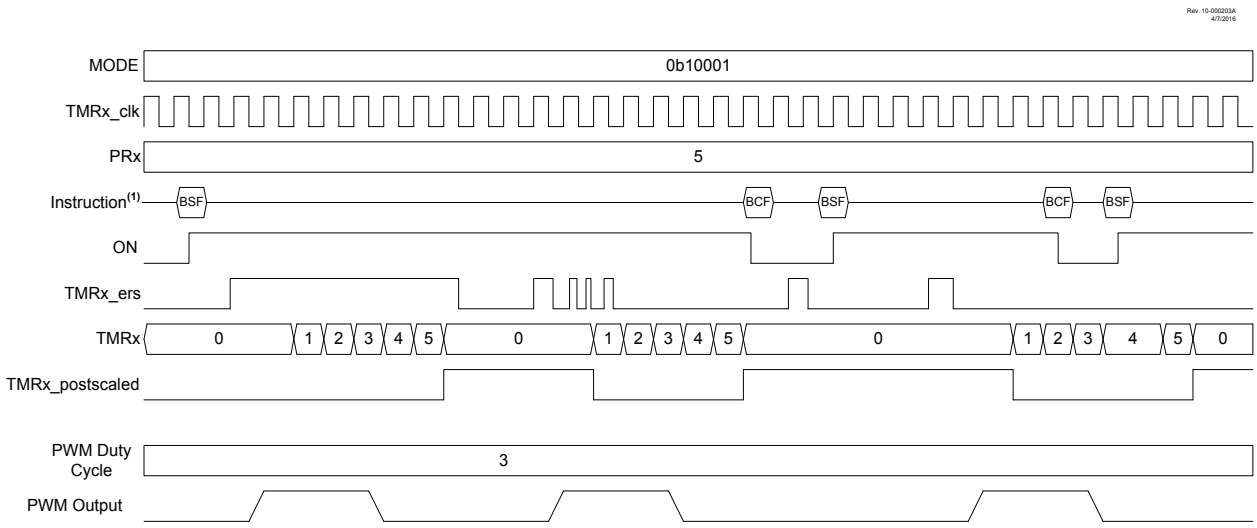
**Bit 1 – HYS** Comparator Hysteresis Enable bit

| Value | Description |
|---|---|
| 1 | Comparator hysteresis enabled |
| 0 | Comparator hysteresis disabled |

**Bit 0 – SYNC** Comparator Output Synchronous Mode bit
Output updated on the falling edge of prescaled Timer1 clock.

| Value | Description |
|---|---|
| 1 | Comparator output to Timer1 and I/O pin is synchronous to changes on the prescaled Timer1 clock. |
| 0 | Comparator output to Timer1 and I/O pin is asynchronous |

**Figure 27-11. Rising Edge-Triggered Monostable Mode Timing Diagram (MODE = 10001)**




**Note:**

1. BSF and BCF represent Bit-Set File and Bit-Clear File instructions executed by the CPU to set or clear the ON bit of TxCON. CPU execution is asynchronous to the timer clock input.

**Related Links**

PWM Overview

(PWM) Pulse-Width Modulation

### 27.6.10 Level-Triggered Hardware Limit One-Shot Modes

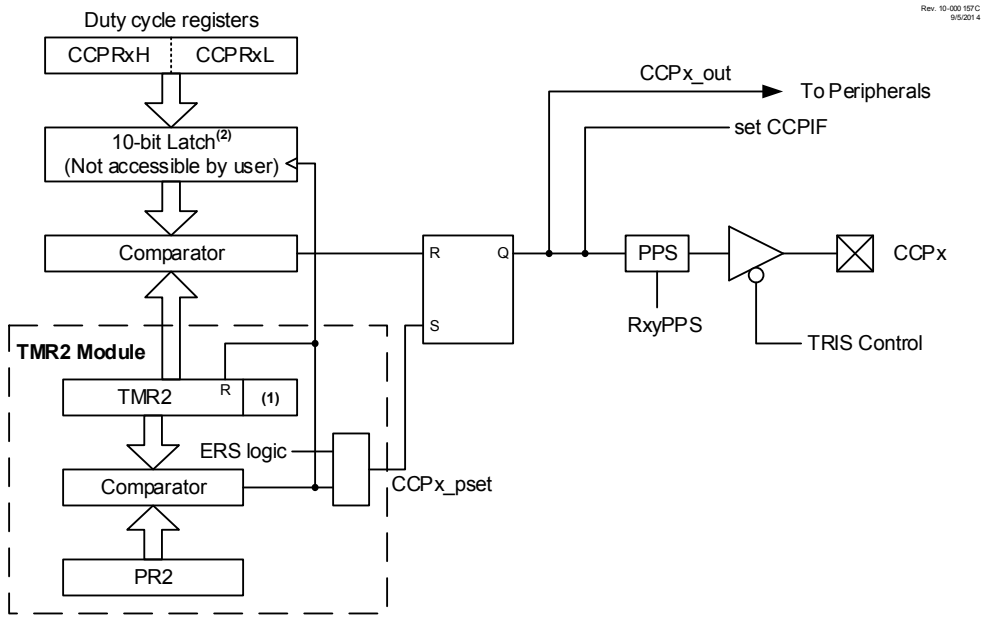
The Level-Triggered Hardware Limit One-Shot modes hold the timer in Reset on an external Reset level and start counting when both the ON bit is set and the external signal is not at the Reset level. If one of either the external signal is not in Reset or the ON bit is set, then the other signal being set/made active will start the timer. Reset levels are selected as follows:

- Low Reset level (MODE<4:0> = 10110)
- High Reset level (MODE<4:0> = 10111)

When the timer count matches the PRx period count, the timer is reset and the ON bit is cleared. When the ON bit is cleared by either a PRx match or by software control, the timer will stay in Reset until both the ON bit is set and the external signal is not at the Reset level.

When Level-Triggered Hardware Limit One-Shot modes are used in conjunction with the CCP PWM operation, the PWM drive goes active with either the external signal edge or the setting of the ON bit, whichever of the two starts the timer.

**Figure 29-4. Simplified PWM Block Diagram**




**Note:**

1. 8-bit timer is concatenated with two bits generated by $F_{OSC}$ or two bits of the internal prescaler to create 10-bit time-base.
2. The alignment of the 10 bits from the CCPRx register is determined by the CCPxFMT bit.

**Important:** The corresponding TRIS bit must be cleared to enable the PWM output on the CCPx pin.

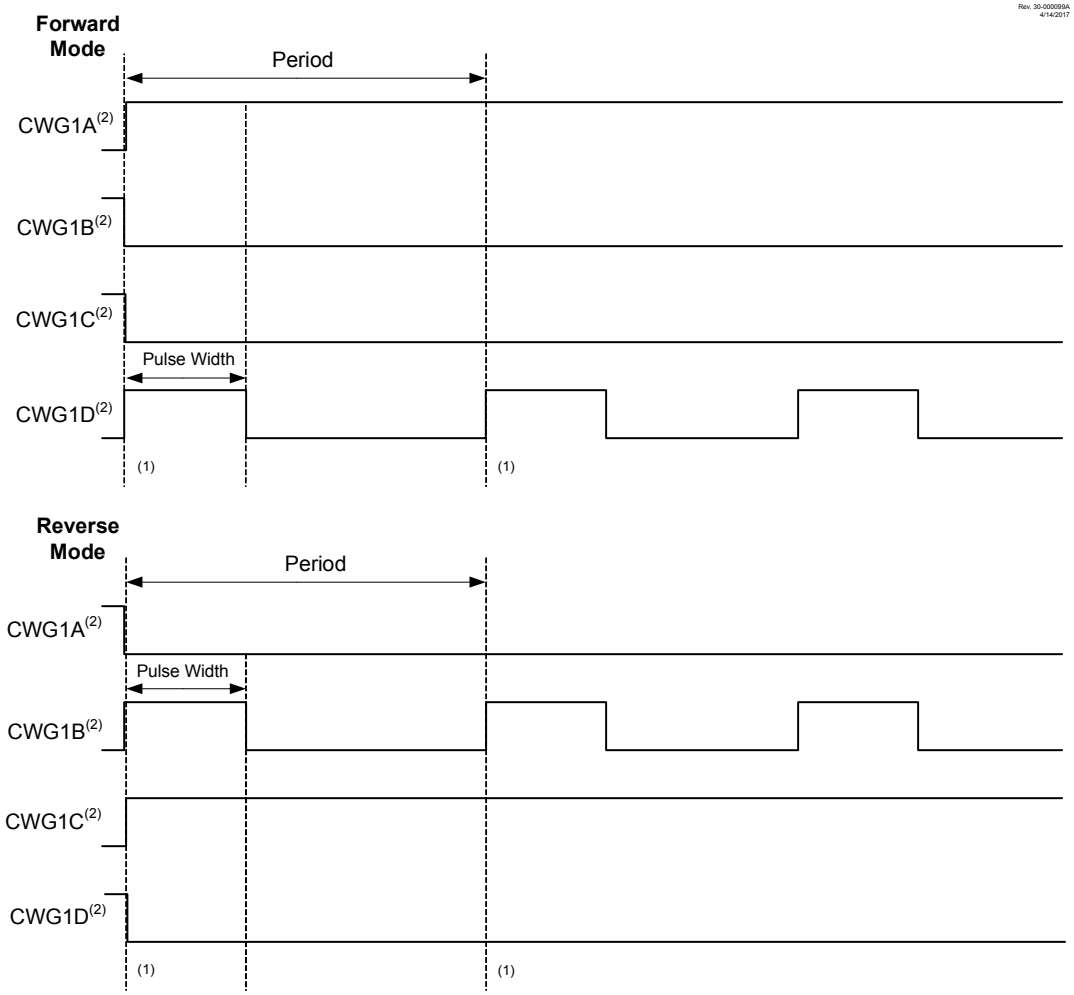
### 29.4.2 Setup for PWM Operation

The following steps should be taken when configuring the CCP module for standard PWM operation:

1. Use the desired output pin RxyPPS control to select CCPx as the source and disable the CCPx pin output driver by setting the associated TRIS bit.
2. Load the T2PR register with the PWM period value.
3. Configure the CCP module for the PWM mode by loading the CCPxCON register with the appropriate values.
4. Load the CCPRx register with the PWM duty cycle value and configure the FMT bit to set the proper register alignment.
5. Configure and start Timer2:
   - Clear the TMR2IF interrupt flag bit of the PIRx register. See Note below.
   - Select the timer clock source to be as $F_{OSC}/4$ using the TxCLKCON register. This is required for correct operation of the PWM module.
   - Configure the T2CKPS bits of the T2CON register with the Timer prescale value.
   - Enable the Timer by setting the T2ON bit.

In Full-Bridge mode, the dead-band period is used when there is a switch from forward to reverse or vice-versa. This dead-band control is described in Dead-Band Control, with additional details in Rising Edge and Reverse Dead Band and Falling Edge and Forward Dead Band. Steering modes are not used with either of the Full-Bridge modes. The mode selection may be toggled between forward and reverse toggling the MODE<0> bit of the CWGxCON0 while keeping MODE<2:1> static, without disabling the CWG module.

**Figure 31-7. Example of Full-Bridge Output**




**Note:**

1. A rising CWG data input creates a rising event on the modulated output.
2. Output signals shown as active-high; all POLy bits are clear.

#### 31.2.3.1 Direction Change in Full-Bridge Mode

In Full-Bridge mode, changing MODE controls the forward/reverse direction. Direction changes occur on the next rising edge of the modulated input.

A direction change is initiated in software by changing the MODE bits. The sequence is illustrated in Figure 31-8.

- The associated active output CWGxA and the inactive output CWGxC are switched to drive in the opposite direction.

### 31.15.4 CWGxISM

**Name:** CWGxISM
**Address:** 0x60D,0x617

CWGx Input Selection Register

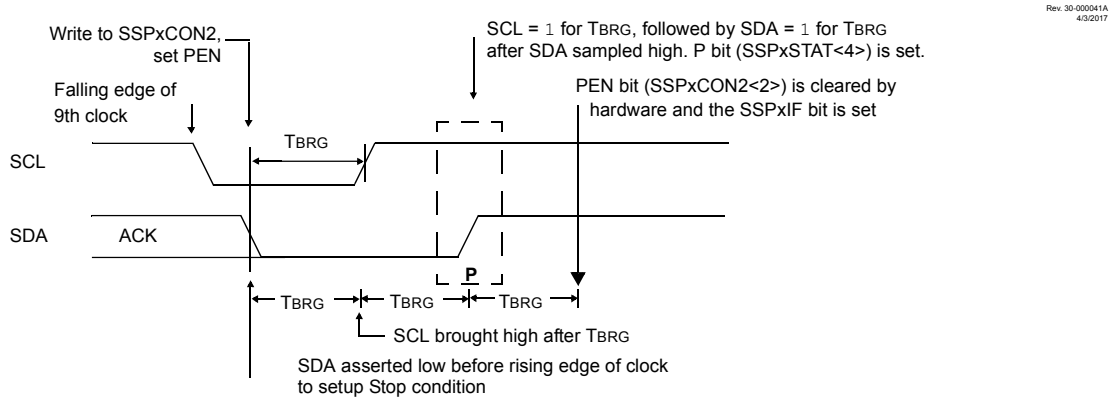
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ISM[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 3:0 – ISM[3:0]** CWG Data Input Source Select bits

**Table 31-4. CWG Data Input Sources**

| ISM | Data Source |
|---|---|
| 1111 | Reserved |
| 1110 | CLC4_out |
| 1101 | CLC3_out |
| 1100 | CLC2_out |
| 1011 | CLC1_out |
| 1010 | DSM1_out |
| 1001 | C2_out |
| 1000 | C1_out |
| 0111 | NCO1_out |
| 0110 | PWM7_out |
| 0101 | PWM6_out |
| 0100 | CCP4_out |
| 0011 | CCP3_out |
| 0010 | CCP2_out |
| 0001 | CCP1_out |
| 0000 | Pin selected by CWGxINPPS |

**Figure 35-31. Stop Condition in Receive or Transmit Mode**




**Note:** TBRG = one Baud Rate Generator period.

### 35.6.9.1 Write Collision on Stop

If the user writes the SSPxBUF when a Stop sequence is in progress, then the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

### 35.6.10 Sleep Operation

While in Sleep mode, the $I^2C$ slave module can receive addresses or data and when an address match or complete byte transfer occurs, wake the processor from Sleep (if the MSSP interrupt is enabled).

### 35.6.11 Effects of a Reset

A Reset disables the MSSP module and terminates the current transfer.

### 35.6.12 Multi-Master Mode

In Multi-Master mode, the interrupt generation on the detection of the Start and Stop conditions allows the determination of when the bus is free. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSP module is disabled. Control of the $I^2C$ bus may be taken when the P bit is set, or the bus is Idle, with both the S and P bits clear. When the bus is busy, enabling the SSP interrupt will generate the interrupt when the Stop condition occurs.

In multi-master operation, the SDA line must be monitored for arbitration to see if the signal level is the expected output level. This check is performed by hardware with the result placed in the BCLxIF bit.

The states where arbitration can be lost are:

- Address Transfer
- Data Transfer
- A Start Condition
- A Repeated Start Condition
- An Acknowledge Condition

### 35.6.13 Multi -Master Communication, Bus Collision and Bus Arbitration

Multi-Master mode support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA, by letting SDA float high and another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin is '0', then a bus collision has taken place. The master will set the Bus Collision Interrupt Flag, BCLxIF and reset the $I^2C$ port to its Idle state (Figure 35-32).

## 36.5 Register Summary - EUSART

| Offset | Name | Bit Pos. | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x0119 | RC1REG | 7:0 | RCREG[7:0] | | | | | | | |
| 0x011A | TX1REG | 7:0 | TXREG[7:0] | | | | | | | |
| 0x011B | SP1BRG | 7:0 | SPBRGL[7:0] | | | | | | | |
| | | 15:8 | SPBRGH[7:0] | | | | | | | |
| 0x011D | RC1STA | 7:0 | SPEN | RX9 | SREN | CREN | ADDEN | FERR | OERR | RX9D |
| 0x011E | TX1STA | 7:0 | CSRC | TX9 | TXEN | SYNC | SENDB | BRGH | TRMT | TX9D |
| 0x011F | BAUD1CON | 7:0 | ABDOVF | RCIDL | | SCKP | BRG16 | | WUE | ABDEN |

## 36.6 Register Definitions: EUSART Control

## 37.2 Register Summary - SMT Control

| Offset | Name | Bit Pos. | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x048C | SMT1TMR | 7:0 | TMRL[7:0] | | | | | | | |
| | | 15:8 | TMRH[7:0] | | | | | | | |
| | | 23:16 | TMRU[7:0] | | | | | | | |
| 0x048F | SMT1CPR | 7:0 | CPRL[7:0] | | | | | | | |
| | | 15:8 | CPRH[7:0] | | | | | | | |
| | | 23:16 | CPRU[7:0] | | | | | | | |
| 0x0492 | SMT1CPW | 7:0 | CPWL[7:0] | | | | | | | |
| | | 15:8 | CPWH[7:0] | | | | | | | |
| | | 23:16 | CPWU[7:0] | | | | | | | |
| 0x0495 | SMT1PR | 7:0 | PRL[7:0] | | | | | | | |
| | | 15:8 | PRH[7:0] | | | | | | | |
| | | 23:16 | PRU[7:0] | | | | | | | |
| 0x0498 | SMT1CON0 | 7:0 | EN | | STP | WPOL | SPOL | CPOL | PS[1:0] | |
| 0x0499 | SMT1CON1 | 7:0 | GO | REPEAT | | | MODE[3:0] | | | |
| 0x049A | SMT1STAT | 7:0 | CPRUP | CPWUP | | RST | | TS | WS | AS |
| 0x049B | SMT1CLK | 7:0 | | | | | | CSEL[2:0] | | |
| 0x049C | SMT1SIG | 7:0 | | | | SSEL[4:0] | | | | |
| 0x049D | SMT1WIN | 7:0 | | | | WSEL[4:0] | | | | |

## 37.3 Register Definitions: SMT Control

### 42.4.6 Temperature Indicator Requirements

**Table 42-12.**

| Standard Operating Conditions (unless otherwise stated) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Param. No.** | **Sym.** | **Characteristic** | | **Min.** | | **Max.** | **Units** | **Conditions** |
| TS01 | $T_{ACQMIN}$ | Minimum ADC Acquisition Time Delay | | — | 25 | — | µs | |
| TS02 | Mv | Voltage Sensitivity | High Range | — | -3.684 | — | mV/°C | TSRNG = 1 |
| | | | Low Range | — | -3.456 | — | mV/°C | TSRNG = 0 |

* - These parameters are characterized but not tested.

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

### 42.4.7 Analog-To-Digital Converter (ADC) Accuracy Specifications(1,2)

**Table 42-13.**

| Standard Operating Conditions (unless otherwise stated)<br>$V_{DD}$ = 3.0V, $T_A$ = 25°C, $T_{AD}$ = 1µs | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym.** | **Characteristic** | **Min.** | **Typ. †** | **Max.** | **Units** | **Conditions** |
| AD01 | $N_R$ | Resolution | — | — | 12 | bit | |
| AD02 | $E_{IL}$ | Integral Error | — | ±1.0 | ±0.2 | LSb | $ADC_{REF}$+=3.0V, $ADC_{REF}$- = 0V |
| AD03 | $E_{DL}$ | Differential Error | — | ±1.0 | ±1.0 | LSb | $ADC_{REF}$+=3.0V, $ADC_{REF}$- = 0V |
| AD04 | $E_{OFF}$ | Offset Error | — | 0.5 | 6.5 | LSb | $ADC_{REF}$+=3.0V, $ADC_{REF}$- = 0V |
| AD05 | $E_{GN}$ | Gain Error | — | ±0.2 | ±6.0 | LSb | $ADC_{REF}$+=3.0V, $ADC_{REF}$- = 0V |
| AD06 | $V_{ADREF}$ | ADC Reference Voltage ($AD_{REF}$+ - $AD_{REF}$-) | 1.8 | — | $V_{DD}$ | V | |
| AD07 | $V_{AIN}$ | Full-Scale Range | $AD_{REF}$- | — | $AD_{REF}$+ | V | |
| AD08 | $Z_{AIN}$ | Recommended Impedance of Analog Voltage Source | — | 10 | — | kΩ | |
| AD09 | $R_{VREF}$ | ADC Voltage Reference Ladder Impedance | — | 50 | — | kΩ | |

* - These parameters are characterized but not tested.