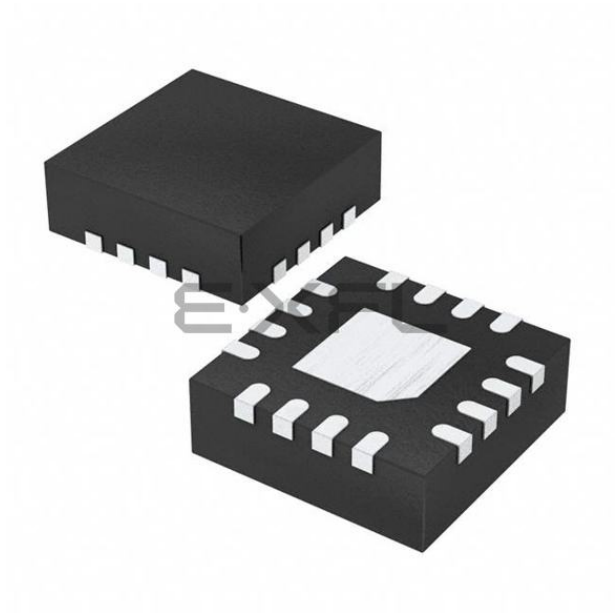


**Welcome to E-XFL.COM**

## 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 | 7KB (4K x 14) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 512 x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.3V ~ 5.5V |
| Data Converters | A/D 8x10b; D/A 1x8b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 125°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 16-VQFN Exposed Pad |
| Supplier Device Package | 16-QFN (4x4) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16f1704-e-ml |

## Pin Allocation Tables

### TABLE 1: 14/16-PIN ALLOCATION TABLE (PIC16(L)F1704)

| I/O(2) | PDIP/SOIC/SSOP | QFN | ADC | Reference | Comparator | Op Amp | DAC | Zero Cross | Timers | CCP | PWM | COG | MSSP | EUSART | CLC | Interrupt | Pull-up | Basic |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RA0 | 13 | 12 | AN0 | VREF- | C1IN+ | — | DAC1OUT1 | — | — | — | — | — | — | — | — | IOC | Y | ICSPDAT |
| RA1 | 12 | 11 | AN1 | VREF+ | C1IN0-<br>C2IN0- | — | — | — | — | — | — | — | — | — | — | IOC | Y | ICSPCLK |
| RA2 | 11 | 10 | AN2 | — | — | — | DAC1OUT2 | ZCD | T0CKI(1) | — | — | COGIN(1) | — | — | — | INT(1)<br>IOC | Y | — |
| RA3 | 4 | 3 | — | — | — | — | — | — | — | — | — | — | — | — | — | IOC | Y | $\overline{MCLR}$<br>VPP |
| RA4 | 3 | 2 | AN3 | — | — | — | — | — | T1G(1)<br>SOSCO | — | — | — | — | — | — | IOC | Y | CLKOUT<br>OSC2 |
| RA5 | 2 | 1 | — | — | — | — | — | — | T1CKI(1)<br>SOSCI | — | — | — | — | — | CLCIN3(1) | IOC | Y | CLKIN<br>OSC1 |
| RC0 | 10 | 9 | AN4 | — | C2IN+ | OPA1IN+ | — | — | — | — | — | — | SCK(1)<br>SCL(3) | — | — | IOC | Y | — |
| RC1 | 9 | 8 | AN5 | — | C1IN1-<br>C2IN1- | OPA1IN- | — | — | — | — | — | — | SDI(1)<br>SDA(3) | — | CLCIN2(1) | IOC | Y | — |
| RC2 | 8 | 7 | AN6 | — | C1IN2-<br>C2IN2- | OPA1OUT | — | — | — | — | — | — | — | — | — | IOC | Y | — |
| RC3 | 7 | 6 | AN7 | — | C1IN3-<br>C2IN3- | OPA2OUT | — | — | — | CCP2(1) | — | — | $\overline{SS}$(1) | — | CLCIN0(1) | IOC | Y | — |
| RC4 | 6 | 5 | — | — | — | OPA2IN- | — | — | — | — | — | — | — | CK(1) | CLCIN1(1) | IOC | Y | — |
| RC5 | 5 | 4 | — | — | — | OPA2IN+ | — | — | — | CCP1(1) | — | — | — | RX(1,3) | — | IOC | Y | — |
| VDD | 1 | 16 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | VDD |
| VSS | 14 | 13 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | VSS |
| OUT(2) | — | — | — | — | C1OUT | — | — | — | — | CPP1 | PWM3OUT | COGA | SDA(3) | CK | CLC1OUT | — | — | — |
| | — | — | — | — | C2OUT | — | — | — | — | CPP2 | PWM4OUT | COGB | SCL(3) | DT(3) | CLC2OUT | — | — | — |
| | — | — | — | — | — | — | — | — | — | — | — | COGC | SDO | TX | CLC3OUT | — | — | — |
| | — | — | — | — | — | — | — | — | — | — | — | COGD | SCK | — | — | — | — | — |

**Note 1:** Default peripheral input. Input can be moved to any other pin with the PPS input selection registers. See Register 12-1.
**2:** All pin outputs default to PORT latch data. Any pin can be selected as a digital peripheral output with the PPS output selection registers. See Register 12-3.
**3:** These peripheral functions are bidirectional. The output pin selections must be the same as the input pin selections.

**TABLE 3-10: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)**

| Addr | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other Resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bank 28** | | | | | | | | | | | |
| E0Ch — E0Eh | — | Unimplemented | | | | | | | | — | — |
| E0Fh | PPSLOCK | — | — | — | — | — | — | — | PPSLOCKED | ---- ---0 | ---- ---0 |
| E10h | INTPPS | — | — | — | INTPPS<4:0> | | | | | ---0 0010 | ---u uuuu |
| E11h | T0CKIPPS | — | — | — | T0CKIPPS<4:0> | | | | | ---0 0010 | ---u uuuu |
| E12h | T1CKIPPS | — | — | — | T1CKIPPS<4:0> | | | | | ---0 0101 | ---u uuuu |
| E13h | T1GPPS | — | — | — | T1GPPS<4:0> | | | | | ---0 0100 | ---u uuuu |
| E14h | CCP1PPS | — | — | — | CCP1PPS<4:0> | | | | | ---1 0101 | ---u uuuu |
| E15h | CCP2PPS | — | — | — | CCP2PPS<4:0> | | | | | ---1 0011 | ---u uuuu |
| E16h | — | Unimplemented | | | | | | | | — | — |
| E17h | COGINPPS | — | — | — | COGINPPS<4:0> | | | | | ---0 0010 | ---u uuuu |
| E18h — E1Fh | — | Unimplemented | | | | | | | | — | — |
| E20h | SSPCLKPPS | — | — | — | SSPCLKPPS<4:0> | | | | | ---1 0000(3) | ---u uuuu |
| | | — | — | — | SSPCLKPPS<4:0> | | | | | ---0 1110(4) | ---u uuuu |
| E21h | SSPDATPPS | — | — | — | SSPDATPPS<4:0> | | | | | ---1 0001(3) | ---u uuuu |
| | | — | — | — | SSPDATPPS<4:0> | | | | | ---0 1100(4) | ---u uuuu |
| E22h | SSPSSPPS | — | — | — | SSPSSPPS<4:0> | | | | | ---1 0011(3) | ---u uuuu |
| | | — | — | — | SSPSSPPS<4:0> | | | | | ---1 0110(4) | ---u uuuu |
| E23h | — | Unimplemented | | | | | | | | — | — |
| E24h | RXPPS | — | — | — | RXPPS<4:0> | | | | | ---1 0101(3) | ---u uuuu |
| | | — | — | — | RXPPS<4:0> | | | | | ---0 1101(4) | ---u uuuu |
| E25h | CKPPS | — | — | — | CKPPS<4:0> | | | | | ---1 0100(3) | ---u uuuu |
| | | — | — | — | CKPPS<4:0> | | | | | ---0 1111(4) | ---u uuuu |
| E26h | — | Unimplemented | | | | | | | | — | — |
| E27h | — | Unimplemented | | | | | | | | — | — |
| E28h | CLCIN0PPS | — | — | — | CLCIN0PPS<4:0> | | | | | ---1 0011 | ---u uuuu |
| E29h | CLCIN1PPS | — | — | — | CLCIN1PPS<4:0> | | | | | ---1 0100 | ---u uuuu |
| E2Ah | CLCIN2PPS | — | — | — | CLCIN2PPS<4:0> | | | | | ---1 0001 | ---u uuuu |
| E2Bh | CLCIN3PPS | — | — | — | CLCIN3PPS<4:0> | | | | | ---0 0101 | ---u uuuu |
| E2Ch to E7Fh | — | Unimplemented | | | | | | | | — | — |

**Legend:** x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations are unimplemented, read as '0'.

**Note 1:** Unimplemented, read as '1'.
**2:** PIC16(L)F1704 only.
**3:** PIC16(L)F1708 only.
**4:** Unimplemented on PIC16LF1704/8.

**TABLE 3-10: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)**

| Addr | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other Resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bank 29** | | | | | | | | | | | |
| E8Ch — E8Fh | — | Unimplemented | | | | | | | | — | — |
| E90h | RA0PPS | — | — | — | RA0PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| E91h | RA1PPS | — | — | — | RA1PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| E92h | RA2PPS | — | — | — | RA2PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| E93h | — | Unimplemented | | | | | | | | — | — |
| E94h | RA4PPS | — | — | — | RA4PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| E95h | RA5PPS | — | — | — | RA5PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| E96h | — | Unimplemented | | | | | | | | — | — |
| E97h | — | Unimplemented | | | | | | | | — | — |
| E98h | — | Unimplemented | | | | | | | | — | — |
| E99h | — | Unimplemented | | | | | | | | — | — |
| E9Ah | — | Unimplemented | | | | | | | | — | — |
| E9Bh | — | Unimplemented | | | | | | | | — | — |
| E9Ch | RB4PPS(3) | — | — | — | RB4PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| E9Dh | RB5PPS(3) | — | — | — | RB5PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| E9Eh | RB6PPS(4) | — | — | — | RB6PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| E9Fh | RB7PPS(3) | — | — | — | RB7PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| EA0h | RC0PPS | — | — | — | RC0PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| EA1h | RC1PPS | — | — | — | RC1PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| EA2h | RC2PPS | — | — | — | RC2PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| EA3h | RC3PPS | — | — | — | RC3PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| EA4h | RC4PPS | — | — | — | RC4PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| EA5h | RC5PPS | — | — | — | RC5PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| EA6h | RC6PPS(4) | — | — | — | RC6PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| EA7h | RC7PPS(4) | — | — | — | RC7PPS<4:0> | | | | | ---0 0000 | ---u uuuu |
| EA8h — EEFh | — | Unimplemented | | | | | | | | — | — |

**Legend:** x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations are unimplemented, read as '0'.

**Note 1:** Unimplemented, read as '1'.
**2:** PIC16(L)F1704 only.
**3:** PIC16(L)F1708 only.
**4:** Unimplemented on PIC16LF1704/8.

**REGISTER 4-1: CONFIG1: CONFIGURATION WORD 1 (CONTINUED)**

bit 2-0 **FOSC<2:0>:** Oscillator Selection bits

111 = ECH: External Clock, High-Power mode (4-20 MHz): device clock supplied to CLKIN pin
110 = ECM: External Clock, Medium-Power mode (0.5-4 MHz): device clock supplied to CLKIN pin
101 = ECL: External Clock, Low-Power mode (0-0.5 MHz): device clock supplied to CLKIN pin
100 = INTOSC oscillator: I/O function on CLKIN pin
011 = EXTRC oscillator: External RC circuit connected to CLKIN pin
010 = HS oscillator: High-speed crystal/resonator connected between OSC1 and OSC2 pins
001 = XT oscillator: Crystal/resonator connected between OSC1 and OSC2 pins
000 = LP oscillator: Low-power crystal connected between OSC1 and OSC2 pins

**Note 1:** The entire Flash program memory will be erased when the code protection is turned off during an erase. When a Bulk Erase Program Memory Command is executed, the entire Program Flash Memory and configuration memory will be erased.

## 4.3 Code Protection

Code protection allows the device to be protected from unauthorized access. Program memory protection is controlled independently. Internal access to the program memory is unaffected by any code protection setting.

### 4.3.1 PROGRAM MEMORY PROTECTION

The entire program memory space is protected from external reads and writes by the $\overline{CP}$ bit in Configuration Words. When $\overline{CP}$ = 0, external reads and writes of program memory are inhibited and a read will return all '0's. The CPU can continue to read program memory, regardless of the protection bit settings. Writing the program memory is dependent upon the write protection setting. See **Section 4.4 "Write Protection"** for more information.

## 4.4 Write Protection

Write protection allows the device to be protected from unintended self-writes. Applications, such as boot loader software, can be protected while allowing other regions of the program memory to be modified.

The WRT<1:0> bits in Configuration Words define the size of the program memory block that is protected.

## 4.5 User ID

Four memory locations (8000h-8003h) are designated as ID locations where the user can store checksum or other code identification numbers. These locations are readable and writable during normal execution. See **Section 10.4 "User ID, Device ID and Configuration Word Access"** for more information on accessing these memory locations. For more information on checksum calculation, see the *"PIC16(L)F170X Memory Programming Specification"* (DS41683).

#### 6.2.2.5 Internal Oscillator Frequency Selection

The system clock speed can be selected via software using the Internal Oscillator Frequency Select bits IRCF<3:0> of the OSCCON register.

The postscaled output of the 16 MHz HFINTOSC, 500 kHz MFINTOSC, and 31 kHz LFINTOSC connect to a multiplexer (see Figure 6-1). The Internal Oscillator Frequency Select bits IRCF<3:0> of the OSCCON register select the frequency output of the internal oscillators. One of the following frequencies can be selected via software:

- 32 MHz (requires 4x PLL)
- 16 MHz
- 8 MHz
- 4 MHz
- 2 MHz
- 1 MHz
- 500 kHz (default after Reset)
- 250 kHz
- 125 kHz
- 62.5 kHz
- 31.25 kHz
- 31 kHz (LFINTOSC)

| **Note:** | Following any Reset, the IRCF<3:0> bits of the OSCCON register are set to '0111' and the frequency selection is set to 500 kHz. The user can modify the IRCF bits to select a different frequency. |
|---|---|

The IRCF<3:0> bits of the OSCCON register allow duplicate selections for some frequencies. These duplicate choices can offer system design trade-offs. Lower power consumption can be obtained when changing oscillator sources for a given frequency. Faster transition times can be obtained between frequency changes that use the same oscillator source.

#### 6.2.2.6 32 MHz Internal Oscillator Frequency Selection

The Internal Oscillator Block can be used with the 4x PLL associated with the External Oscillator Block to produce a 32 MHz internal system clock source. The following settings are required to use the 32 MHz internal clock source:

- The FOSC bits in Configuration Words must be set to use the INTOSC source as the device system clock (FOSC<2:0> = 100).
- The SCS bits in the OSCCON register must be cleared to use the clock determined by FOSC<2:0> in Configuration Words (SCS<1:0> = 00).
- The IRCF bits in the OSCCON register must be set to the 8 MHz HFINTOSC set to use (IRCF<3:0> = 1110).
- The SPLLEN bit in the OSCCON register must be set to enable the 4x PLL, or the PLLEN bit of the Configuration Words must be programmed to a '1'.

| **Note:** | When using the PLLEN bit of the Configuration Words, the 4x PLL cannot be disabled by software and the SPLLEN option will not be available. |
|---|---|

The 4x PLL is not available for use with the internal oscillator when the SCS bits of the OSCCON register are set to '1x'. The SCS bits must be set to '00' to use the 4x PLL with the internal oscillator.

## 7.1 Operation

Interrupts are disabled upon any device Reset. They are enabled by setting the following bits:

- GIE bit of the INTCON register
- Interrupt Enable bit(s) for the specific interrupt event(s)
- PEIE bit of the INTCON register (if the Interrupt Enable bit of the interrupt event is contained in the PIE1 or PIE2 registers)

The INTCON, PIR1 and PIR2 registers record individual interrupts via interrupt flag bits. Interrupt flag bits will be set, regardless of the status of the GIE, PEIE and individual interrupt enable bits.

The following events happen when an interrupt event occurs while the GIE bit is set:

- Current prefetched instruction is flushed
- GIE bit is cleared
- Current Program Counter (PC) is pushed onto the stack
- Critical registers are automatically saved to the shadow registers (See **"Section 7.5 "Automatic Context Saving"**)
- PC is loaded with the interrupt vector 0004h

The firmware within the Interrupt Service Routine (ISR) should determine the source of the interrupt by polling the interrupt flag bits. The interrupt flag bits must be cleared before exiting the ISR to avoid repeated interrupts. Because the GIE bit is cleared, any interrupt that occurs while executing the ISR will be recorded through its interrupt flag, but will not cause the processor to redirect to the interrupt vector.

The `RETFIE` instruction exits the ISR by popping the previous address from the stack, restoring the saved context from the shadow registers and setting the GIE bit.

For additional information on a specific interrupt's operation, refer to its peripheral chapter.

| | |
|---|---|
| **Note 1:** | Individual interrupt flag bits are set, regardless of the state of any other enable bits. |
| **2:** | All interrupts will be ignored while the GIE bit is cleared. Any interrupt occurring while the GIE bit is clear will be serviced when the GIE bit is set again. |

## 7.2 Interrupt Latency

Interrupt latency is defined as the time from when the interrupt event occurs to the time code execution at the interrupt vector begins. The latency for synchronous interrupts is three or four instruction cycles. For asynchronous interrupts, the latency is three to five instruction cycles, depending on when the interrupt occurs. See Figure 7-2 and Figure 7-3 for more details.

**REGISTER 7-7: PIR3: PERIPHERAL INTERRUPT REQUEST REGISTER 3**

| U-0 | U-0 | R/W-0/0 | R/W-0/0 | U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| — | — | COGIF | ZCDIF | — | CLC3IF | CLC2IF | CLC1IF |
| bit 7 | | | | | | | bit 0 |

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

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

bit 5 **COGIF:** COG Auto-Shutdown Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 4 **ZCDIF:** Zero-Cross Detection Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 3 **Unimplemented**: Read as '0'

bit 2 **CLC3IF:** CLC3 Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 1 **CLC2IF:** CLC2 Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 0 **CLC1IF:** CLC1 Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

**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.

### 10.2.3 ERASING FLASH PROGRAM MEMORY

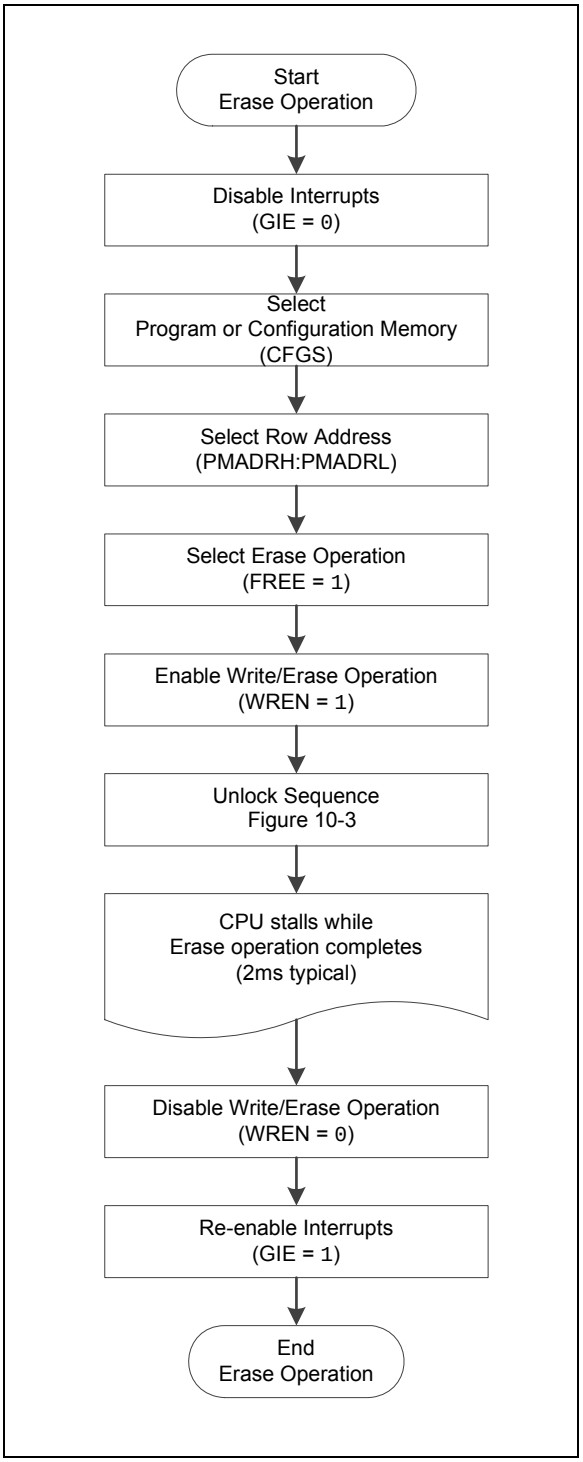
While executing code, program memory can only be erased by rows. To erase a row:

1. Load the PMADRH:PMADRL register pair with any address within the row to be erased.
2. Clear the CFGS bit of the PMCON1 register.
3. Set the FREE and WREN bits of the PMCON1 register.
4. Write 55h, then AAh, to PMCON2 (Flash programming unlock sequence).
5. Set control bit WR of the PMCON1 register to begin the erase operation.

See Example 10-2.

After the "`BSF PMCON1,WR`" instruction, the processor requires two cycles to set up the erase operation. The user must place two `NOP` instructions immediately following the WR bit set instruction. The processor will halt internal operations for the typical 2 ms erase time. This is not Sleep mode as the clocks and peripherals will continue to run. After the erase cycle, the processor will resume operation with the third instruction after the PMCON1 write instruction.

**FIGURE 10-4: FLASH PROGRAM MEMORY ERASE FLOWCHART**



## 10.4 User ID, Device ID and Configuration Word Access

Instead of accessing program memory, the User ID's, Device ID/Revision ID and Configuration Words can be accessed when CFGS = 1 in the PMCON1 register. This is the region that would be pointed to by PC<15> = 1, but not all addresses are accessible. Different access may exist for reads and writes. Refer to Table 10-2.

When read access is initiated on an address outside the parameters listed in Table 10-2, the PMDATH:PMDATL register pair is cleared, reading back '0's.

**TABLE 10-2: USER ID, DEVICE ID AND CONFIGURATION WORD ACCESS (CFGS = 1)**

| Address | Function | Read Access | Write Access |
|---|---|---|---|
| 8000h-8003h | User IDs | Yes | Yes |
| 8005h-8006h | Device ID/Revision ID | Yes | No |
| 8007h-8008h | Configuration Words 1 and 2 | Yes | No |

**EXAMPLE 10-4: CONFIGURATION WORD AND DEVICE ID ACCESS**

```
* This code block will read 1 word of program memory at the memory address:
*   PROG_ADDR_LO (must be 00h-08h) data will be returned in the variables;
*   PROG_DATA_HI, PROG_DATA_LO

    BANKSEL  PMADRL           ; Select correct Bank
    MOVLW    PROG_ADDR_LO     ;
    MOVWF    PMADRL           ; Store LSB of address
    CLRF     PMADRH           ; Clear MSB of address

    BSF      PMCON1,CFGS      ; Select Configuration Space
    BCF      INTCON,GIE       ; Disable interrupts
    BSF      PMCON1,RD        ; Initiate read
    NOP                       ; Executed (See Figure 10-2)
    NOP                       ; Ignored (See Figure 10-2)
    BSF      INTCON,GIE       ; Restore interrupts

    MOVF     PMDATL,W         ; Get LSB of word
    MOVWF    PROG_DATA_LO     ; Store in user location
    MOVF     PMDATH,W         ; Get MSB of word
    MOVWF    PROG_DATA_HI     ; Store in user location
```

## 12.3 Bidirectional Pins

PPS selections for peripherals with bidirectional signals on a single pin must be made so that the PPS input and PPS output select the same pin. Peripherals that have bidirectional signals include:

- EUSART (synchronous operation)
- MSSP ($I^2C$)

| **Note:** | The $I^2C$ default input pins are $I^2C$ and SMBus compatible and are the only pins on the device with this compatibility. |
|---|---|

## 12.4 PPS Lock

The PPS includes a mode in which all input and output selections can be locked to prevent inadvertent changes. PPS selections are locked by setting the PPSLOCKED bit of the PPSLOCK register. Setting and clearing this bit requires a special sequence as an extra precaution against inadvertent changes. Examples of setting and clearing the PPSLOCKED bit are shown in Example 12-1.

**EXAMPLE 12-1: PPS LOCK/UNLOCK SEQUENCE**

```
; suspend interrupts
    bcf     INTCON,GIE
;   BANKSEL PPSLOCK     ; set bank
; required sequence, next 5 instructions
    movlw   0x55
    movwf   PPSLOCK
    movlw   0xAA
    movwf   PPSLOCK
; Set PPSLOCKED bit to disable writes or
; Clear PPSLOCKED bit to enable writes
    bsf     PPSLOCK,PPSLOCKED
; restore interrupts
    bsf     INTCON,GIE
```

## 12.5 PPS Permanent Lock

The PPS can be permanently locked by setting the PPS1WAY Configuration bit. When this bit is set, the PPSLOCKED bit can only be cleared and set one time after a device Reset. This allows for clearing the PPSLOCKED bit so that the input and output selections can be made during initialization. When the PPSLOCKED bit is set after all selections have been made, it will remain set and cannot be cleared until after the next device Reset event.

## 12.6 Operation During Sleep

PPS input and output selections are unaffected by Sleep.

## 12.7 Effects of a Reset

A device Power-On-Reset (POR) clears all PPS input and output selections to their default values. All other Resets leave the selections unchanged. Default input selections are shown in pin allocation Table 1 and Table 2.

## 15.0 TEMPERATURE INDICATOR MODULE

This family of devices is equipped with a temperature circuit designed to measure the operating temperature of the silicon die. The circuit's range of operating temperature falls between -40°C and +85°C. The output is a voltage that is proportional to the device temperature. The output of the temperature indicator is internally connected to the device ADC.

The circuit may be used as a temperature threshold detector or a more accurate temperature indicator, depending on the level of calibration performed. A one-point calibration allows the circuit to indicate a temperature closely surrounding that point. A two-point calibration allows the circuit to sense the entire range of temperature more accurately. Reference Application Note AN1333, "*Use and Calibration of the Internal Temperature Indicator*" (DS01333) for more details regarding the calibration process.

### 15.1 Circuit Operation

Figure 15-1 shows a simplified block diagram of the temperature circuit. The proportional voltage output is achieved by measuring the forward voltage drop across multiple silicon junctions.

Equation 15-1 describes the output characteristics of the temperature indicator.

**EQUATION 15-1: $V_{OUT}$ RANGES**

High Range: $V_{OUT} = V_{DD} - 4V_T$

Low Range: $V_{OUT} = V_{DD} - 2V_T$

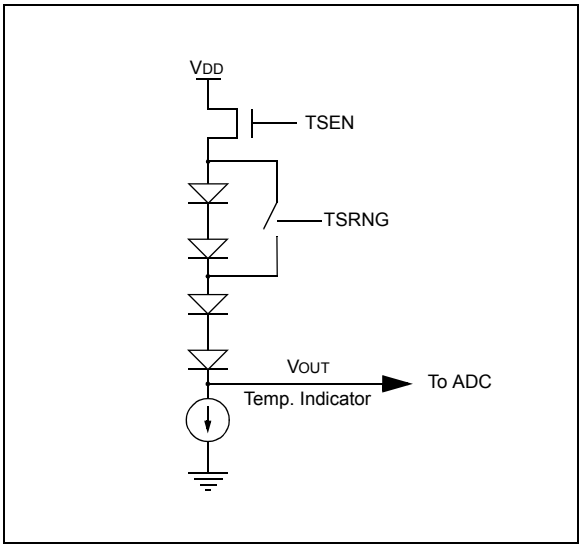
The temperature sense circuit is integrated with the Fixed Voltage Reference (FVR) module. See **Section 14.0 "Fixed Voltage Reference (FVR)"** for more information.

The circuit is enabled by setting the TSEN bit of the FVRCON register. When disabled, the circuit draws no current.

The circuit operates in either high or low range. The high range, selected by setting the TSRNG bit of the FVRCON register, provides a wider output voltage. This provides more resolution over the temperature range, but may be less consistent from part to part. This range requires a higher bias voltage to operate and thus, a higher $V_{DD}$ is needed.

The low range is selected by clearing the TSRNG bit of the FVRCON register. The low range generates a lower voltage drop and thus, a lower bias voltage is needed to operate the circuit. The low range is provided for low voltage operation.

**FIGURE 15-1: TEMPERATURE CIRCUIT DIAGRAM**




### 15.2 Minimum Operating $V_{DD}$

When the temperature circuit is operated in low range, the device may be operated at any operating voltage that is within specifications.

When the temperature circuit is operated in high range, the device operating voltage, $V_{DD}$, must be high enough to ensure that the temperature circuit is correctly biased.

Table 15-1 shows the recommended minimum $V_{DD}$ vs. range setting.

**TABLE 15-1: RECOMMENDED $V_{DD}$ VS. RANGE**

| Min. $V_{DD}$, TSRNG = 1 | Min. $V_{DD}$, TSRNG = 0 |
|---|---|
| 3.6V | 1.8V |

### 15.3 Temperature Output

The output of the circuit is measured using the internal Analog-to-Digital Converter. A channel is reserved for the temperature circuit output. Refer to **Section 20.0 "Analog-to-Digital Converter (ADC) Module"** for detailed information.

## 16.2 Comparator Control

Each comparator has two control registers: CMxCON0 and CMxCON1.

The CMxCON0 register (see Register 16-1) contains Control and Status bits for the following:

- Enable
- Output
- Output polarity
- Zero latency filter
- Speed/Power selection
- Hysteresis enable
- Output synchronization

The CMxCON1 register (see Register 16-2) contains Control bits for the following:

- Interrupt enable
- Interrupt edge polarity
- Positive input channel selection
- Negative input channel selection

### 16.2.1 COMPARATOR ENABLE

Setting the CxON bit of the CMxCON0 register enables the comparator for operation. Clearing the CxON bit disables the comparator resulting in minimum current consumption.

### 16.2.2 COMPARATOR OUTPUT SELECTION

The output of the comparator can be monitored by reading either the CxOUT bit of the CMxCON0 register or the MCxOUT bit of the CMOUT register. In order to make the output available for an external connection, the following conditions must be true:

- Desired pin PPS control
- Corresponding TRIS bit must be cleared
- CxON bit of the CMxCON0 register must be set

**Note 1:** The internal output of the comparator is latched with each instruction cycle. Unless otherwise specified, external outputs are not latched.

### 16.2.3 COMPARATOR OUTPUT POLARITY

Inverting the output of the comparator is functionally equivalent to swapping the comparator inputs. The polarity of the comparator output can be inverted by setting the CxPOL bit of the CMxCON0 register. Clearing the CxPOL bit results in a non-inverted output.

Table 16-2 shows the output state versus input conditions, including polarity control.

**TABLE 16-2: COMPARATOR OUTPUT STATE VS. INPUT CONDITIONS**

| Input Condition | CxPOL | CxOUT |
|---|---|---|
| CxVN > CxVP | 0 | 0 |
| CxVN < CxVP | 0 | 1 |
| CxVN > CxVP | 1 | 1 |
| CxVN < CxVP | 1 | 0 |

### 16.2.4 COMPARATOR SPEED/POWER SELECTION

The trade-off between speed or power can be optimized during program execution with the CxSP control bit. The default state for this bit is '1', which selects the Normal-Speed mode. Device power consumption can be optimized at the cost of slower comparator propagation delay by clearing the CxSP bit to '0'.

## 20.1 ADC Configuration

When configuring and using the ADC the following functions must be considered:

- Port configuration
- Channel selection
- ADC voltage reference selection
- ADC conversion clock source
- Interrupt control
- Result formatting

### 20.1.1 PORT CONFIGURATION

The ADC can be used to convert both analog and digital signals. When converting analog signals, the I/O pin should be configured for analog by setting the associated TRIS and ANSEL bits. Refer to **Section 11.0 "I/O Ports"** for more information.

| | |
|---|---|
| **Note:** | Analog voltages on any pin that is defined as a digital input may cause the input buffer to conduct excess current. |

### 20.1.2 CHANNEL SELECTION

There are up to 17 channel selections available:

- AN<13:8, 4:0> pins (PIC16(L)F1704 only)
- AN<21,13:0> pins (PIC16(L)F1708 only)
- Temperature Indicator
- DAC_output
- FVR_buffer1

The CHS bits of the ADCON0 register (Register 20-1) determine which channel is connected to the sample and hold circuit.

When changing channels, a delay is required before starting the next conversion. Refer to **Section 20.2 "ADC Operation"** for more information.

### 20.1.3 ADC VOLTAGE REFERENCE

The ADPREF bits of the ADCON1 register provides control of the positive voltage reference. The positive voltage reference can be:

- $V_{REF}$+ pin
- $V_{DD}$
- FVR 2.048V
- FVR 4.096V (Not available on LF devices)

The ADNREF bit of the ADCON1 register provides control of the negative voltage reference. The negative voltage reference can be:

- $V_{REF}$- pin
- $V_{SS}$

See **Section 20.0 "Analog-to-Digital Converter (ADC) Module"** for more details on the Fixed Voltage Reference.

### 20.1.4 CONVERSION CLOCK

The source of the conversion clock is software selectable via the ADCS bits of the ADCON1 register. There are seven possible clock options:

- $F_{OSC}$/2
- $F_{OSC}$/4
- $F_{OSC}$/8
- $F_{OSC}$/16
- $F_{OSC}$/32
- $F_{OSC}$/64
- FRC (internal RC oscillator)

The time to complete one bit conversion is defined as $T_{AD}$. One full 10-bit conversion requires 11.5 $T_{AD}$ periods as shown in Figure 20-2.

For correct conversion, the appropriate $T_{AD}$ specification must be met. Refer to Table 32-16: ADC Conversion Requirements for more information. Table 20-1 gives examples of appropriate ADC clock selections.

| | |
|---|---|
| **Note:** | Unless using the FRC, any changes in the system clock frequency will change the ADC clock frequency, which may adversely affect the ADC result. |

**FIGURE 28-15: I²C SLAVE, 7-BIT ADDRESS, RECEPTION (SEN = 1, AHEN = 0, DHEN = 0)**

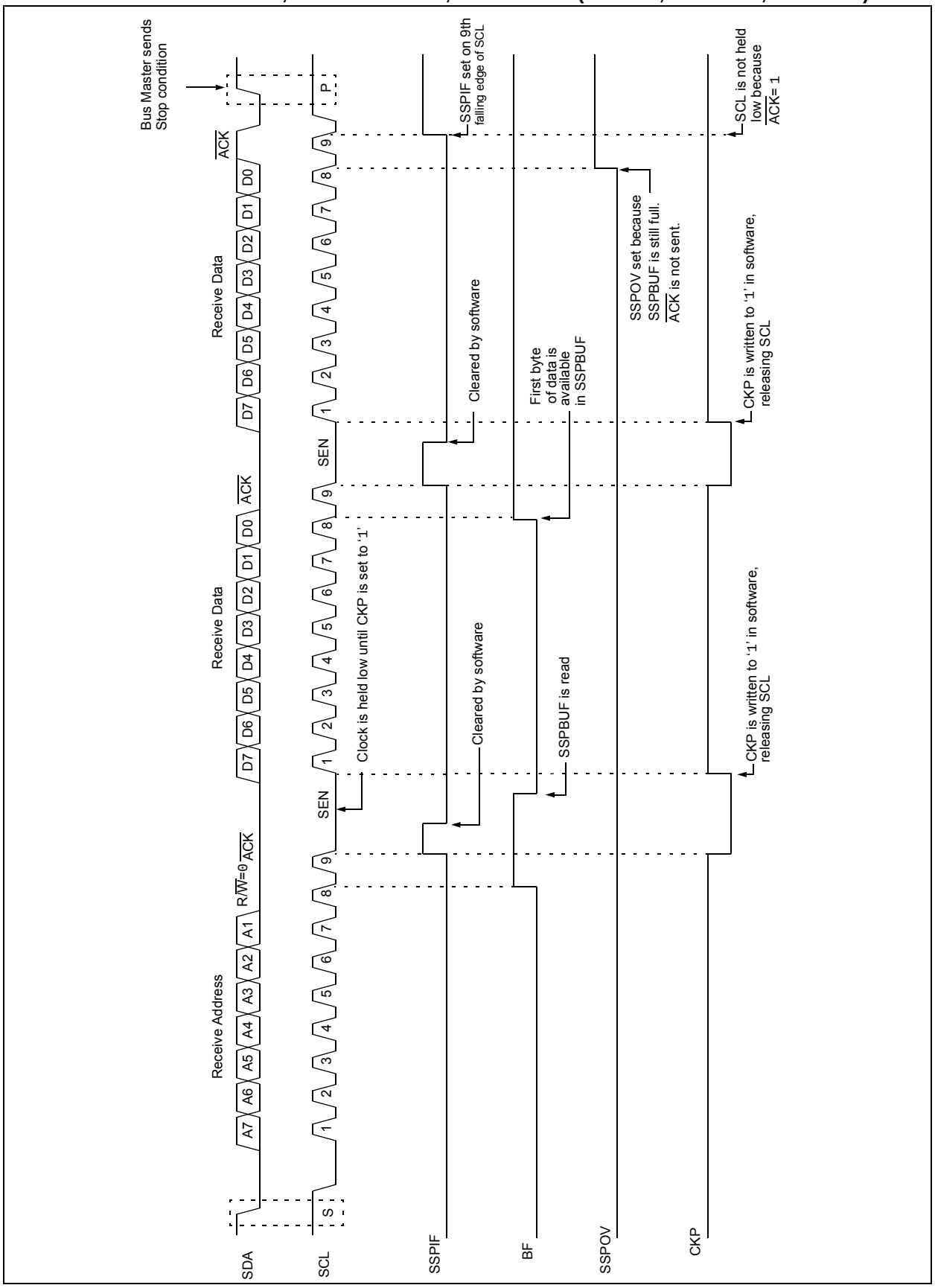
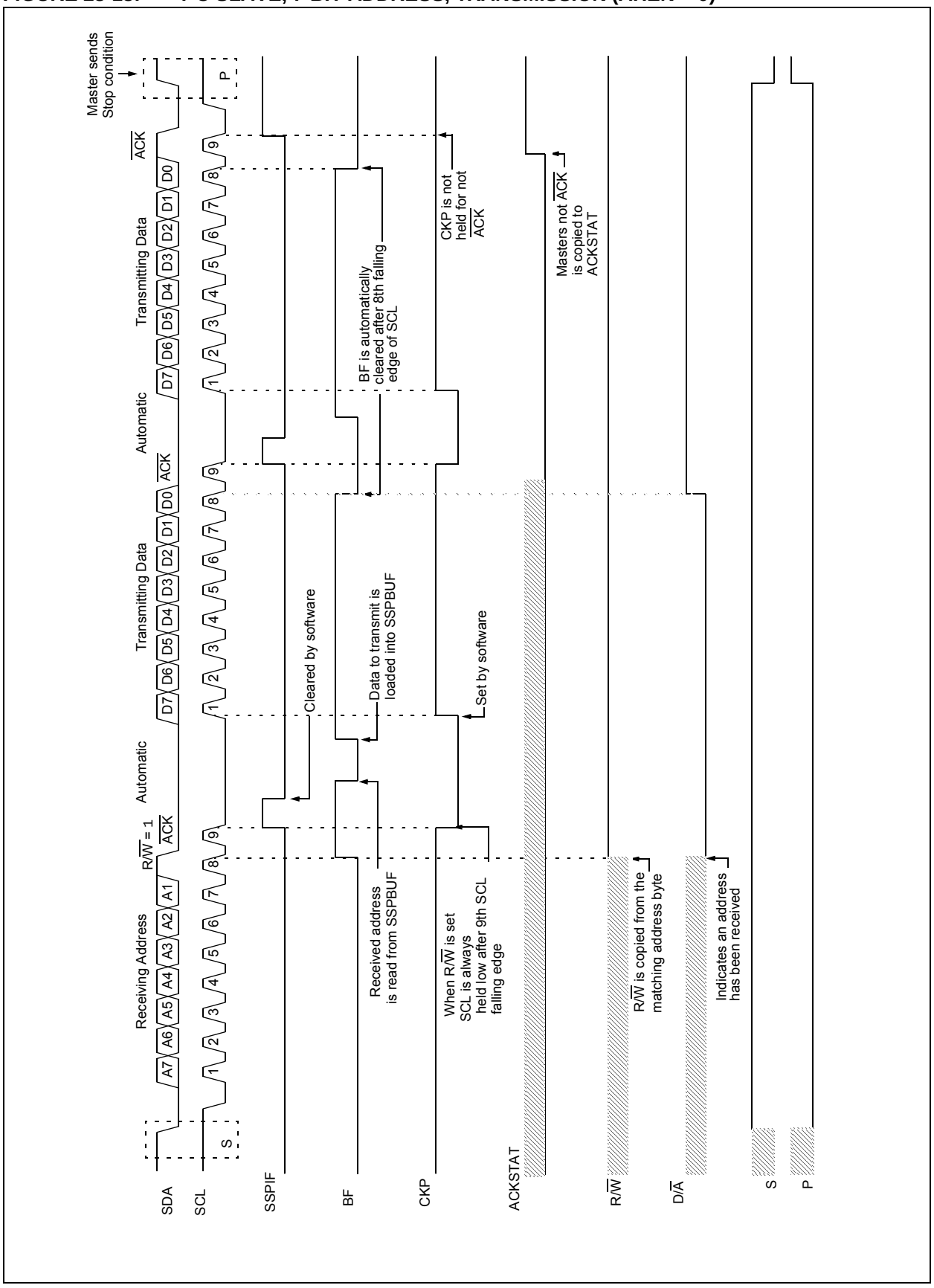


**FIGURE 28-18: I²C SLAVE, 7-BIT ADDRESS, TRANSMISSION (AHEN = 0)**



**REGISTER 28-5: SSP1MSK: SSP 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 |
|---|---|---|---|---|---|---|---|
| MSK<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 **MSK<7:1>:** Mask bits
1 = The received address bit n is compared to SSPADD<n> to detect $I^2C$ address match
0 = The received address bit n is not used to detect $I^2C$ address match

bit 0 **MSK<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 SSPADD<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, the bit is ignored

**REGISTER 28-6: SSP1ADD: MSSP 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 |
|---|---|---|---|---|---|---|---|
| ADD<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 **ADD<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 **ADD<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:**

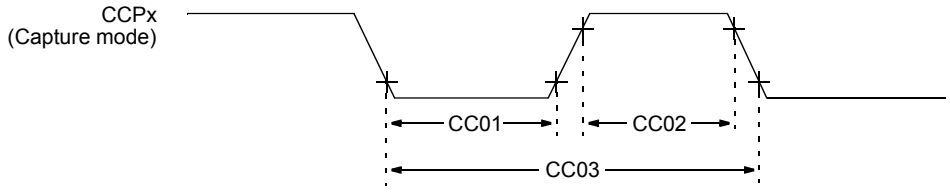
bit 7-0 **ADD<7:0>:** Eight Least Significant bits of 10-bit address

**7-Bit Slave mode:**

bit 7-1 **ADD<7:1>:** 7-bit address

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

**FIGURE 32-11: CAPTURE/COMPARE/PWM TIMINGS (CCP)**




**Note:** Refer to Figure 32-4 for load conditions.

**TABLE 32-13: CAPTURE/COMPARE/PWM REQUIREMENTS (CCP)**

| Standard Operating Conditions (unless otherwise stated) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Param. No.** | **Sym.** | **Characteristic** | | **Min.** | **Typ.†** | **Max.** | **Units** | **Conditions** |
| CC01* | TccL | CCPx Input Low Time | No Prescaler | 0.5 TCY + 20 | — | — | ns | |
| | | | With Prescaler | 20 | — | — | ns | |
| CC02* | TccH | CCPx Input High Time | No Prescaler | 0.5 TCY + 20 | — | — | ns | |
| | | | With Prescaler | 20 | — | — | ns | |
| CC03* | TccP | CCPx Input Period | | $\frac{3\ T_{CY} + 40}{N}$ | — | — | ns | N = prescale value (1, 4 or 16) |

* These parameters are characterized but not tested.

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

**Note:** Unless otherwise noted, VIN = 5V, FOSC = 500 kHz, CIN = 0.1 µF, TA = 25°C.

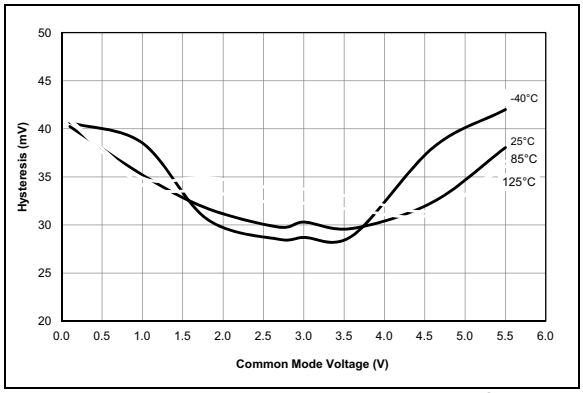



***FIGURE 33-103:*** *Comparator Hysteresis, Normal-Power Mode (CxSP = 1), VDD = 5.5V, Typical Measured Values, PIC16F1704/8 Only.*

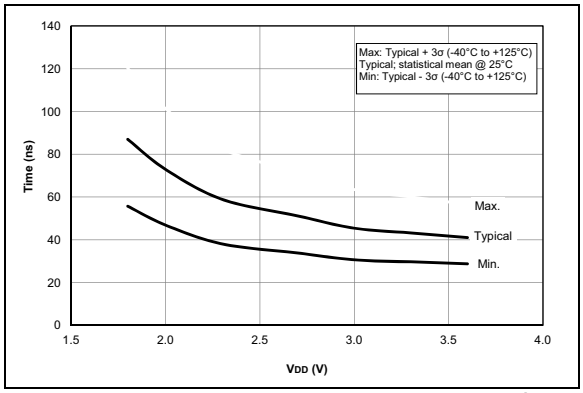



***FIGURE 33-106:*** *Comparator Response Time over Voltage, Normal-Power Mode (CxSP = 1), Typical Measured Values, PIC16LF1704/8 Only.*

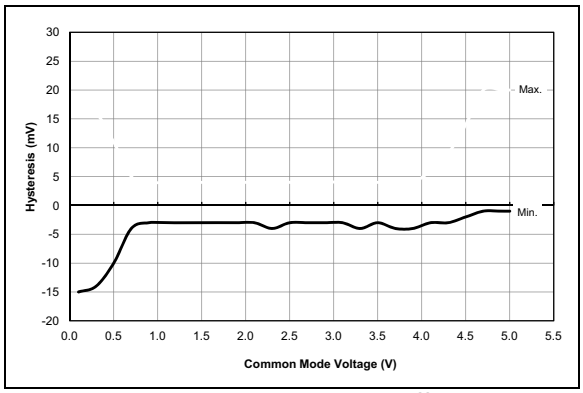



***FIGURE 33-104:*** *Comparator Offset, Normal-Power Mode (CxSP = 1), VDD = 5.0V, Typical Measured Values at 25°C, PIC16F1704/8 Only.*

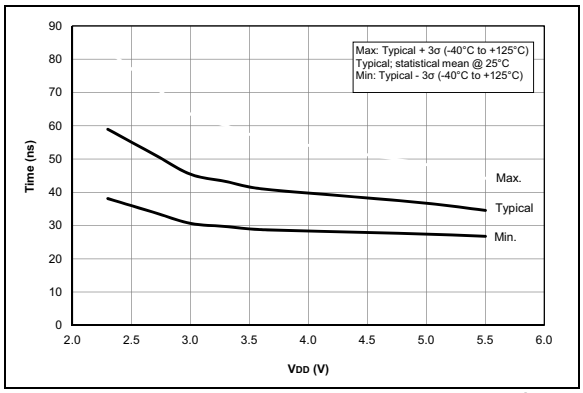



***FIGURE 33-107:*** *Comparator Response Time over Voltage, Normal-Power Mode (CxSP = 1), Typical Measured Values, PIC16F1704/8 Only.*

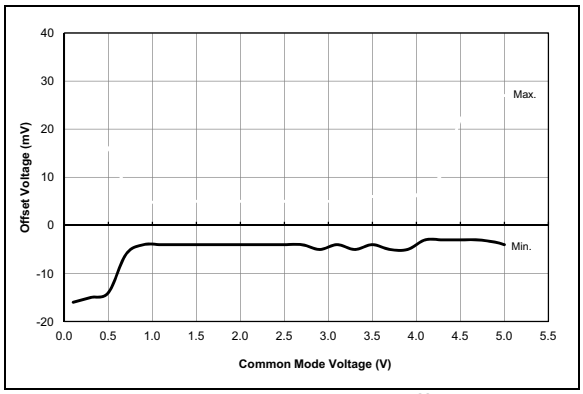



***FIGURE 33-105:*** *Comparator Offset, Normal-Power Mode (CxSP = 1), VDD = 5.5V, Typical Measured Values from -40°C to 125°C, PIC16F1704/8 Only.*

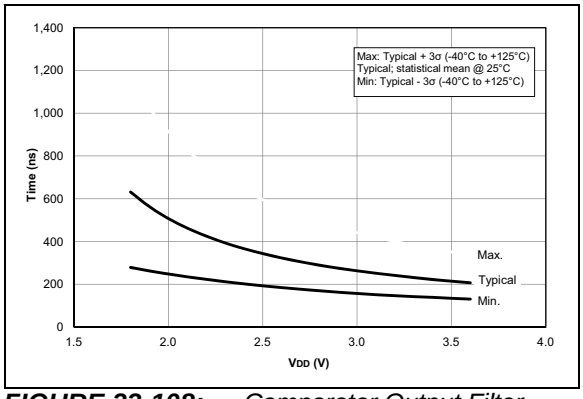
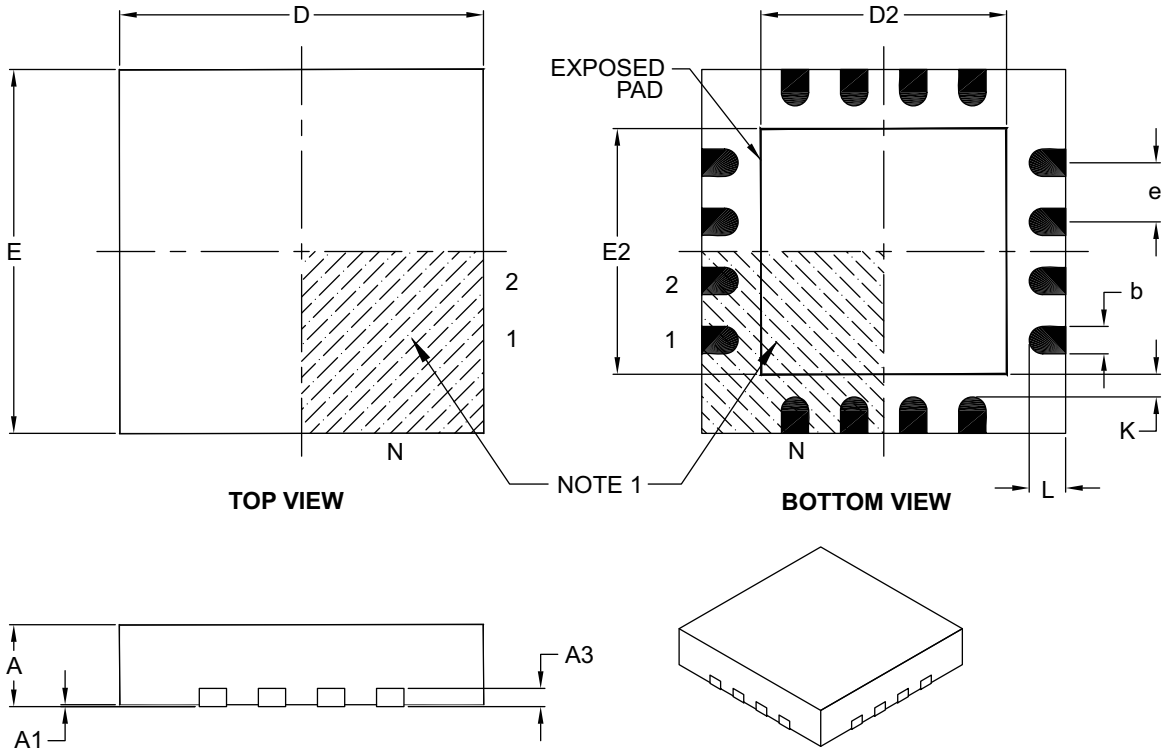



***FIGURE 33-108:*** *Comparator Output Filter Delay Time over Temperature, Normal-Power Mode, Typical Measured Values, PIC16LF1704/8 Only.*

### 16-Lead Plastic Quad Flat, No Lead Package (ML) – 4x4x0.9 mm Body [QFN]

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




| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Pins | N | | 16 | |
| Pitch | e | | 0.65 BSC | |
| Overall Height | A | 0.80 | 0.90 | 1.00 |
| Standoff | A1 | 0.00 | 0.02 | 0.05 |
| Contact Thickness | A3 | | 0.20 REF | |
| Overall Width | E | | 4.00 BSC | |
| Exposed Pad Width | E2 | 2.50 | 2.65 | 2.80 |
| Overall Length | D | | 4.00 BSC | |
| Exposed Pad Length | D2 | 2.50 | 2.65 | 2.80 |
| Contact Width | b | 0.25 | 0.30 | 0.35 |
| Contact Length | L | 0.30 | 0.40 | 0.50 |
| Contact-to-Exposed Pad | K | 0.20 | – | – |

**Notes:**

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-127B