**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 | LINbus, 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 12x10b; D/A 1x5b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 125°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 20-SOIC (0.295", 7.50mm Width) |
| Supplier Device Package | 20-SOIC |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16f1578-e-so |

**TABLE 4: 20-PIN ALLOCATION TABLE (PIC16(L)F1578/9)**

| I/O | 20-Pin PDIP/SOIC/SSOP | 20-Pin UQFN | ADC | Reference | Comparator | Timers | PWM | EUSART | CWG | Interrupt | Pull-up | Basic |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RA0 | 19 | 16 | AN0 | DAC1OUT1 | C1IN+ | — | — | — | — | IOC | Y | ICSPDAT |
| RA1 | 18 | 15 | AN1 | VREF+ | C1IN0-/C2IN0- | — | — | — | — | IOC | Y | ICSPCLK |
| RA2 | 17 | 14 | AN2 | — | — | T0CKI(1) | — | — | CWG1IN(1) | INT(1)/IOC | Y | — |
| RA3 | 4 | 1 | — | — | — | — | — | — | — | IOC | Y | $\overline{\text{MCLR}}$/VPP |
| RA4 | 3 | 20 | AN3 | — | — | T1G(1) | — | — | — | IOC | Y | CLKOUT |
| RA5 | 2 | 19 | — | — | — | T1CKI(1) | — | — | — | IOC | Y | CLKIN |
| RB4 | 13 | 10 | AN10 | — | — | — | — | — | — | IOC | Y | — |
| RB5 | 12 | 9 | AN11 | — | — | — | — | RX(1,3) | — | IOC | Y | — |
| RB6 | 11 | 8 | — | — | — | — | — | — | — | IOC | Y | — |
| RB7 | 10 | 7 | — | — | — | — | — | CK(1) | — | IOC | Y | — |
| RC0 | 16 | 13 | AN4 | — | C2IN+ | — | — | — | — | IOC | Y | — |
| RC1 | 15 | 12 | AN5 | — | C1IN1-/C2IN1- | — | — | — | — | IOC | Y | — |
| RC2 | 14 | 11 | AN6 | — | C1IN2-/C2IN2- | — | — | — | — | IOC | Y | — |
| RC3 | 7 | 4 | AN7 | — | C1IN3-/C2IN3- | — | — | — | — | IOC | Y | — |
| RC4 | 6 | 3 | ADCACT(1) | — | — | — | — | — | — | IOC | Y | — |
| RC5 | 5 | 2 | — | — | — | — | — | — | — | IOC | Y | — |
| RC6 | 8 | 5 | AN8 | — | — | — | — | — | — | IOC | Y | — |
| RC7 | 9 | 6 | AN9 | — | — | — | — | — | — | IOC | Y | — |
| VDD | 1 | 18 | — | — | — | — | — | — | — | — | — | VDD |
| Vss | 20 | 17 | — | — | — | — | — | — | — | — | — | Vss |
| OUT(2) | — | — | — | — | C1OUT | — | PWM1OUT | DT(3) | CWG1A | — | — | — |
| | — | — | — | — | C2OUT | — | PWM2OUT | CK | CWG1B | — | — | — |
| | — | — | — | — | — | — | PWM3OUT | TX | — | — | — | — |
| | — | — | — | — | — | — | PWM4OUT | — | — | — | — | — |

**Note 1:** Default peripheral input. Input can be moved to any other pin with the PPS Input Selection registers.
**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.
**3:** These peripheral functions are bidirectional. The output pin selections must be the same as the input pin selections.

**TABLE 3-15: SPECIAL FUNCTION REGISTER SUMMARY**

| Address | 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 0** | | | | | | | | | | | |
| 00Ch | PORTA | — | — | RA5 | RA4 | RA3 | RA2 | RA1 | RA0 | --xx xxxx | --xx xxxx |
| 00Dh | PORTB(1) | RB7 | RB6 | RB5 | RB4 | — | — | — | — | xxxx ---- | xxxx ---- |
| 00Eh | PORTC | RC7(1) | RC6(1) | RC5 | RC4 | RC3 | RC2 | RC1 | RC0 | xxxx xxxx | xxxx xxxx |
| 00Fh | — | Unimplemented | | | | | | | | — | — |
| 010h | — | Unimplemented | | | | | | | | — | — |
| 011h | PIR1 | TMR1GIF | ADIF | RCIF | TXIF | — | — | TMR2IF | TMR1IF | 0000 --00 | 0000 --00 |
| 012h | PIR2 | — | C2IF | C1IF | — | — | — | — | — | -00- ---- | -00- ---- |
| 013h | PIR3 | PWM4IF | PWM3IF | PWM2IF | PWM1IF | — | — | — | — | 0000 ---- | 0000 ---- |
| 014h | — | | | | | | | | | — | — |
| 015h | TMR0 | Holding Register for the 8-bit Timer0 Count | | | | | | | | xxxx xxxx | uuuu uuuu |
| 016h | TMR1L | Holding Register for the Least Significant Byte of the 16-bit TMR1 Count | | | | | | | | xxxx xxxx | uuuu uuuu |
| 017h | TMR1H | Holding Register for the Most Significant Byte of the 16-bit TMR1 Count | | | | | | | | xxxx xxxx | uuuu uuuu |
| 018h | T1CON | TMR1CS<1:0> | | T1CKPS<1:0> | | — | $\overline{\text{T1SYNC}}$ | — | TMR1ON | 0000 -0-0 | uuuu -u-u |
| 019h | T1GCON | TMR1GE | T1GPOL | T1GTM | T1GSPM | $\overline{\text{T1GGO}}$/ $\overline{\text{DONE}}$ | T1GVAL | T1GSS<1:0> | | 0000 0x00 | uuuu uxuu |
| 01Ah | TMR2 | Timer2 Module Register | | | | | | | | 0000 0000 | 0000 0000 |
| 01Bh | PR2 | Timer2 Period Register | | | | | | | | 1111 1111 | 1111 1111 |
| 01Ch | T2CON | — | T2OUTPS<3:0> | | | | TMR2ON | T2CKPS<1:0> | | -000 0000 | -000 0000 |
| 01Dh | — | Unimplemented | | | | | | | | — | — |
| 01Eh | — | Unimplemented | | | | | | | | — | — |
| 01Fh | — | Unimplemented | | | | | | | | — | — |

**Legend:** x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, r = reserved. Shaded locations are unimplemented, read as '0'.
**Note 1:** PIC16(L)F1578/9 only.
**2:** PIC16F1574/5/8/9 only.
**3:** Unimplemented, read as '1'.

**TABLE 3-15: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)**

| Address | 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 7** | | | | | | | | | | | |
| 38Ch | INLVLA | — | — | INLVLA5 | INLVLA4 | INLVLA3 | INLVLA2 | INLVLA1 | INLVLA0 | --11 1111 | --11 1111 |
| 38Dh | INLVLB(1) | INLVLB7 | INLVLB6 | INLVLB5 | INLVLB4 | — | — | — | — | 1111 ---- | 1111 ---- |
| 38Eh | INLVLC | INLVLC7(1) | INLVLC6(1) | INLVLC5 | INLVLC4 | INLVLC3 | INLVLC2 | INLVLC1 | INLVLC0 | 1111 1111 | 1111 1111 |
| 38Fh to 390h | — | Unimplemented | | | | | | | | — | — |
| 391h | IOCAP | — | — | IOCAP5 | IOCAP4 | IOCAP3 | IOCAP2 | IOCAP1 | IOCAP0 | --00 0000 | --00 0000 |
| 392h | IOCAN | — | — | IOCAN5 | IOCAN4 | IOCAN3 | IOCAN2 | IOCAN1 | IOCAN0 | --00 0000 | --00 0000 |
| 393h | IOCAF | — | — | IOCAF5 | IOCAF4 | IOCAF3 | IOCAF2 | IOCAF1 | IOCAF0 | --00 0000 | --00 0000 |
| 394h | IOCBP(1) | IOCBP7 | IOCBP6 | IOCBP5 | IOCBP4 | — | — | — | — | 0000 ---- | --00 ---- |
| 395h | IOCBN(1) | IOCBN7 | IOCBN6 | IOCBN5 | IOCBN4 | — | — | — | — | 0000 ---- | --00 ---- |
| 396h | IOCBF(1) | IOCBF7 | IOCBF6 | IOCBF5 | IOCBF4 | — | — | — | — | 0000 ---- | --00 ---- |
| 397h | IOCCP | IOCCP7(1) | IOCCP6(1) | IOCCP5 | IOCCP4 | IOCCP3 | IOCCP2 | IOCCP1 | IOCCP0 | 0000 0000 | 0000 0000 |
| 398h | IOCCN | IOCCN7(1) | IOCCN6(1) | IOCCN5 | IOCCN4 | IOCCN3 | IOCCN2 | IOCCN1 | IOCCN0 | 0000 0000 | 0000 0000 |
| 399h | IOCCF | IOCCF7(1) | IOCCF6(1) | IOCCF5 | IOCCF4 | IOCCF3 | IOCCF2 | IOCCF1 | IOCCF0 | 0000 0000 | 0000 0000 |
| 39Ah to 39Fh | — | Unimplemented | | | | | | | | — | — |
| **Bank 8** | | | | | | | | | | | |
| 40Ch to 41Fh | — | Unimplemented | | | | | | | | — | — |
| **Bank 9** | | | | | | | | | | | |
| 48Ch to 49Fh | — | Unimplemented | | | | | | | | — | — |

**Legend:** x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, r = reserved. Shaded locations are unimplemented, read as '0'.
**Note 1:** PIC16(L)F1578/9 only.
**2:** PIC16F1574/5/8/9 only.
**3:** Unimplemented, read as '1'.

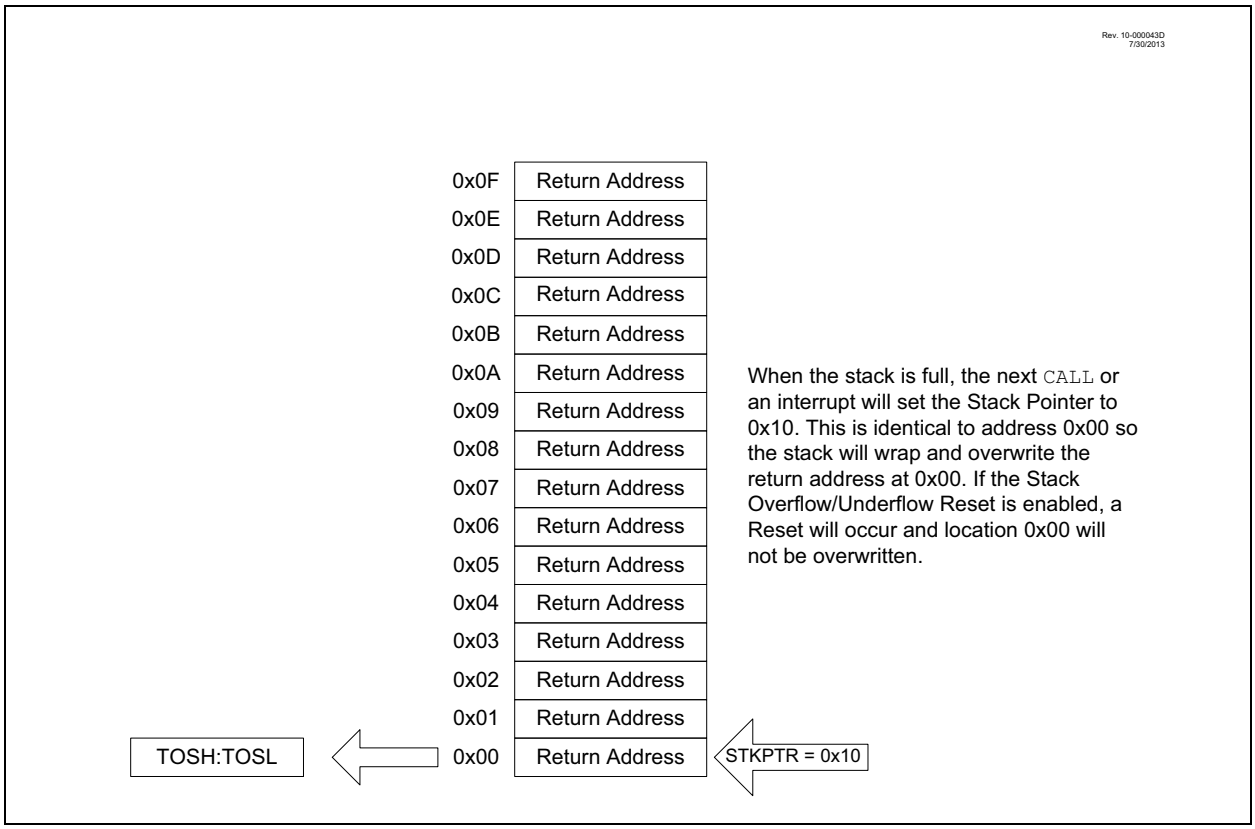
**TABLE 3-15: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)**

| Address | 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 27** | | | | | | | | | | | |
| D8Ch | — | Unimplemented | | | | | | | | — | — |
| D8Dh | — | Unimplemented | | | | | | | | — | — |
| D8Eh | PWMEN | — | — | — | — | PWM4EN_A | PWM3EN_A | PWM2EN_A | PWM1EN_A | ---- 0000 | ---- 0000 |
| D8Fh | PWMLD | — | — | — | — | PWM4LDA_A | PWM3LDA_A | PWM2LDA_A | PWM1LDA_A | ---- 0000 | ---- 0000 |
| D90h | PWMOUT | — | — | — | — | PWM4OUT_A | PWM3OUT_A | PWM2OUT_A | PWM1OUT_A | ---- 0000 | ---- 0000 |
| D91h | PWM1PHL | PH<7:0> | | | | | | | | xxxx xxxx | uuuu uuuu |
| D92h | PWM1PHH | PH<15:8> | | | | | | | | xxxx xxxx | uuuu uuuu |
| D93h | PWM1DCL | DC<7:0> | | | | | | | | xxxx xxxx | uuuu uuuu |
| D94h | PWM1DCH | DC<15:8> | | | | | | | | xxxx xxxx | uuuu uuuu |
| D95h | PWM1PRL | PR<7:0> | | | | | | | | xxxx xxxx | uuuu uuuu |
| D96h | PWM1PRH | PR<15:8> | | | | | | | | xxxx xxxx | uuuu uuuu |
| D97h | PWM1OFL | OF<7:0> | | | | | | | | xxxx xxxx | uuuu uuuu |
| D98h | PWM1OFH | OF<15:8> | | | | | | | | xxxx xxxx | uuuu uuuu |
| D99h | PWM1TMRL | TMR<7:0> | | | | | | | | xxxx xxxx | uuuu uuuu |
| D9Ah | PWM1TMRH | TMR<15:8> | | | | | | | | xxxx xxxx | uuuu uuuu |
| D9Bh | PWM1CON | EN | — | OUT | POL | MODE<1:0> | | — | — | 0-00 00-- | 0-00 00-- |
| D9Ch | PWM1INTE | — | — | — | — | OFIE | PHIE | DCIE | PRIE | ---- 000 | ---- 000 |
| D9Dh | PWM1INTF | — | — | — | — | OFIF | PHIF | DCIF | PRIF | ---- 000 | ---- 000 |
| D9Eh | PWM1CLKCON | — | PS<2:0> | | | — | — | CS<1:0> | | -000 -000 | -000 --00 |
| D9Fh | PWM1LDCON | LDA | LDT | — | — | — | — | LDS<1:0> | | 00-- -000 | 00-- --00 |
| DA0h | PWM1OFCON | — | OFM<1:0> | | OFO | — | — | OFS<1:0> | | -000 -000 | -000 --00 |
| DA1h | PWM2PHL | PH<7:0> | | | | | | | | xxxx xxxx | uuuu uuuu |
| DA2h | PWM2PHH | PH<15:8> | | | | | | | | xxxx xxxx | uuuu uuuu |
| DA3h | PWM2DCL | DC<7:0> | | | | | | | | xxxx xxxx | uuuu uuuu |
| DA4h | PWM2DCH | DC<15:8> | | | | | | | | xxxx xxxx | uuuu uuuu |
| DA5h | PWM2PRL | PR<7:0> | | | | | | | | xxxx xxxx | uuuu uuuu |
| DA6h | PWM2PRH | PR<15:8> | | | | | | | | xxxx xxxx | uuuu uuuu |
| DA7h | PWM2OFL | OF<7:0> | | | | | | | | xxxx xxxx | uuuu uuuu |
| DA8h | PWM2OFH | OF<15:8> | | | | | | | | xxxx xxxx | uuuu uuuu |
| DA9h | PWM2TMRL | TMR<7:0> | | | | | | | | xxxx xxxx | uuuu uuuu |
| DAAh | PWM2TMRH | TMR<15:8> | | | | | | | | xxxx xxxx | uuuu uuuu |

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

**Note 1:** PIC16(L)F1578/9 only.
**2:** PIC16F1574/5/8/9 only.
**3:** Unimplemented, read as '1'.

**FIGURE 3-8: ACCESSING THE STACK EXAMPLE 4**




### 3.5.2 OVERFLOW/UNDERFLOW RESET

If the STVREN bit in Configuration Words is programmed to '1', the device will be reset if the stack is PUSHed beyond the sixteenth level or POPed beyond the first level, setting the appropriate bits (STKOVF or STKUNF, respectively) in the PCON register.

## 3.6 Indirect Addressing

The INDFn registers are not physical registers. Any instruction that accesses an INDFn register actually accesses the register at the address specified by the File Select Registers (FSR). If the FSRn address specifies one of the two INDFn registers, the read will return '0' and the write will not occur (though Status bits may be affected). The FSRn register value is created by the pair FSRnH and FSRnL.

The FSR registers form a 16-bit address that allows an addressing space with 65536 locations. These locations are divided into three memory regions:

- Traditional Data Memory
- Linear Data Memory
- Program Flash Memory

**REGISTER 5-2: OSCSTAT: OSCILLATOR STATUS REGISTER**

| U-0 | R-0/q | R-q/q | R-0/q | R-0/q | R-q/q | R-0/q | R-0/q |
|---|---|---|---|---|---|---|---|
| — | PLLR | OSTS | HFIOFR | HFIOFL | MFIOFR | LFIOFR | HFIOFS |
| 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 | q = Conditional |

bit 7 **Unimplemented:** Read as ‘0’

bit 6 **PLLR** 4x PLL Ready bit
1 = 4x PLL is ready
0 = 4x PLL is not ready

bit 5 **OSTS:** Oscillator Start-up Timer Status bit
1 = Running from the clock defined by the FOSC<1:0> bits of the Configuration Words
0 = Running from an internal oscillator (FOSC<1:0> = 00)

bit 4 **HFIOFR:** High-Frequency Internal Oscillator Ready bit
1 = HFINTOSC is ready
0 = HFINTOSC is not ready

bit 3 **HFIOFL:** High-Frequency Internal Oscillator Locked bit
1 = HFINTOSC is at least 2% accurate
0 = HFINTOSC is not 2% accurate

bit 2 **MFIOFR:** Medium-Frequency Internal Oscillator Ready bit
1 = MFINTOSC is ready
0 = MFINTOSC is not ready

bit 1 **LFIOFR:** Low-Frequency Internal Oscillator Ready bit
1 = LFINTOSC is ready
0 = LFINTOSC is not ready

bit 0 **HFIOFS:** High-Frequency Internal Oscillator Stable bit
1 = HFINTOSC is at least 0.5% accurate
0 = HFINTOSC is not 0.5% accurate

**REGISTER 7-5: PIR1: PERIPHERAL INTERRUPT REQUEST REGISTER 1**

| R/W-0/0 | R/W-0/0 | R-0/0 | R-0/0 | U-0 | U-0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| TMR1GIF | ADIF | RCIF | TXIF | — | — | TMR2IF | TMR1IF |
| 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 **TMR1GIF:** Timer1 Gate Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 6 **ADIF:** ADC Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 5 **RCIF:** USART Receive Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 4 **TXIF:** USART Transmit Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 3-2 **Unimplemented:** Read as '0'

bit 1 **TMR2IF:** Timer2 to PR2 Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 0 **TMR1IF:** Timer1 Overflow 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 Interrupt Enable bit, GIE of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt. The USART RCIF and TXIF bits are read-only.

## 21.0 TIMER2 MODULE

The Timer2 module incorporates the following features:

- 8-bit Timer and Period registers (TMR2 and PR2, respectively)
- Readable and writable (both registers)
- Software programmable prescaler (1:1, 1:4, 1:16, and 1:64)
- Software programmable postscaler (1:1 to 1:16)
- Interrupt on TMR2 match with PR2

See Figure 21-1 for a block diagram of Timer2.

**FIGURE 21-1: TIMER2 BLOCK DIAGRAM**

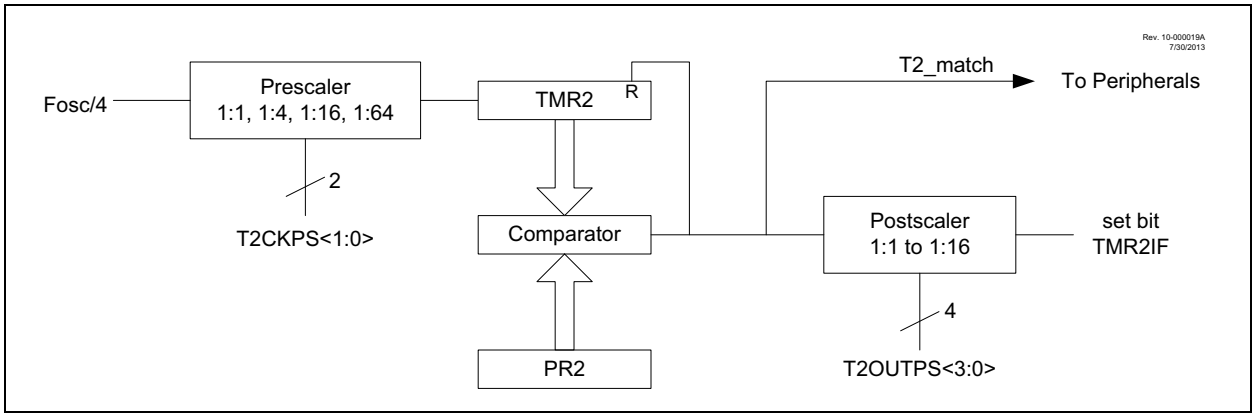
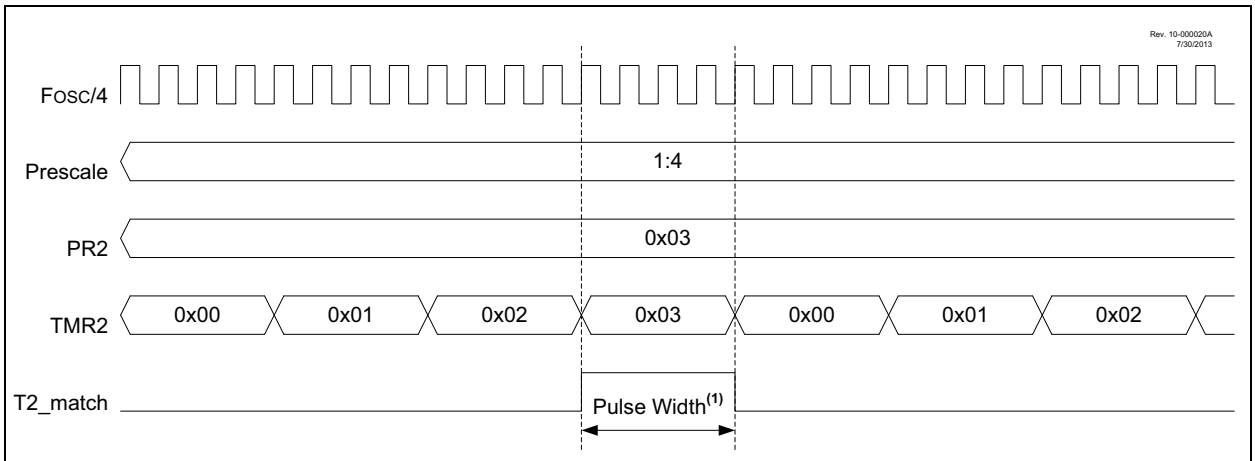
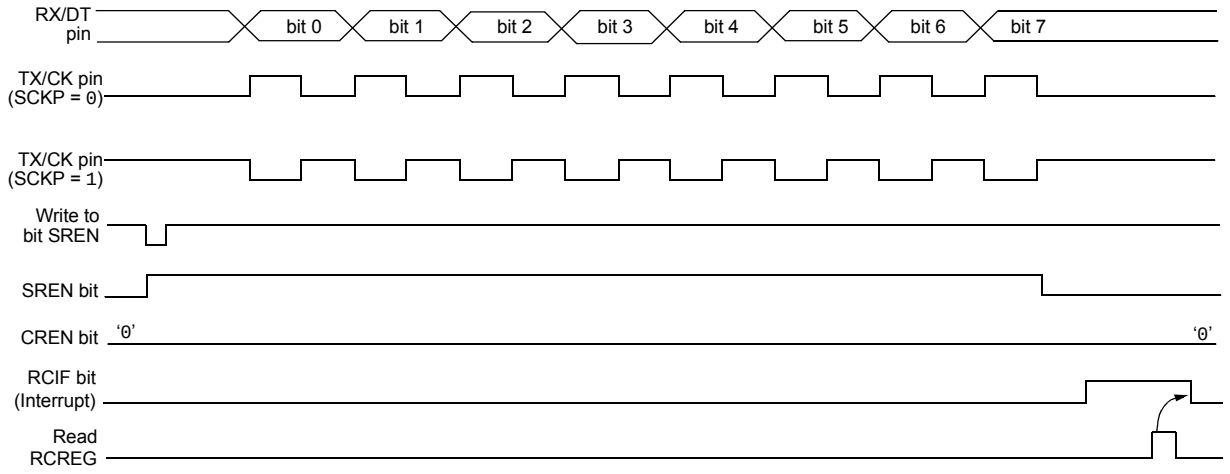



**FIGURE 21-2: TIMER2 TIMING DIAGRAM**




**Note 1:** The Pulse Width of T2_match is equal to the scaled input of TMR2.

**FIGURE 22-12: SYNCHRONOUS RECEPTION (MASTER MODE, SREN)**




**Note:** Timing diagram demonstrates Sync Master mode with bit SREN = 1 and bit BRGH = 0.

**TABLE 22-8: SUMMARY OF REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER RECEPTION**

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---|---|---|---|---|---|---|---|---|---|
| BAUDCON | ABDOVF | RCIDL | — | SCKP | BRG16 | — | WUE | ABDEN | 204 |
| INTCON | GIE | PEIE | TMR0IE | INTE | IOCIE | TMR0IF | INTF | IOCIF | 86 |
| PIE1 | TMR1GIE | ADIE | RCIE | TXIE | — | — | TMR2IE | TMR1IE | 87 |
| PIR1 | TMR1GIF | ADIF | RCIF | TXIF | — | — | TMR2IF | TMR1IF | 90 |
| RCREG | EUSART Receive Data Register | | | | | | | | 197* |
| RCSTA | SPEN | RX9 | SREN | CREN | ADDEN | FERR | OERR | RX9D | 203 |
| SPBRGL | BRG<7:0> | | | | | | | | 205* |
| SPBRGH | BRG<15:8> | | | | | | | | 205* |
| TXSTA | CSRC | TX9 | TXEN | SYNC | SENDB | BRGH | TRMT | TX9D | 202 |

**Legend:** — = unimplemented location, read as '0'. Shaded cells are not used for synchronous master reception.

* Page provides register information.

**REGISTER 23-4: PWMxCLKCON: PWM CLOCK CONTROL REGISTER**

| U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | U-0 | U-0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| — | PS<2:0> | | | — | — | CS<1:0> | |
| 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 **Unimplemented:** Read as '0'

bit 6-4 **PS<2:0>:** Clock Source Prescaler Select bits
111 = Divide clock source by 128
110 = Divide clock source by 64
101 = Divide clock source by 32
100 = Divide clock source by 16
011 = Divide clock source by 8
010 = Divide clock source by 4
001 = Divide clock source by 2
000 = No Prescaler

bit 3-2 **Unimplemented:** Read as '0'

bit 1-0 **CS<1:0>:** Clock Source Select bits
11 = Reserved
10 = LFINTOSC (continues to operate during Sleep)
01 = HFINTOSC (continues to operate during Sleep)
00 = FOSC

## 26.0 INSTRUCTION SET SUMMARY

Each instruction is a 14-bit word containing the operation code (opcode) and all required operands. The opcodes are broken into three broad categories.

- Byte Oriented
- Bit Oriented
- Literal and Control

The literal and control category contains the most varied instruction word format.

Table 26-3 lists the instructions recognized by the MPASM™ assembler.

All instructions are executed within a single instruction cycle, with the following exceptions, which may take two or three cycles:

- Subroutine takes two cycles (CALL, CALLW)
- Returns from interrupts or subroutines take two cycles (RETURN, RETLW, RETFIE)
- Program branching takes two cycles (GOTO, BRA, BRW, BTFSS, BTFSC, DECFSZ, INCSFZ)
- One additional instruction cycle will be used when any instruction references an indirect file register and the file select register is pointing to program memory.

One instruction cycle consists of 4 oscillator cycles; for an oscillator frequency of 4 MHz, this gives a nominal instruction execution rate of 1 MHz.

All instruction examples use the format '0xhh' to represent a hexadecimal number, where 'h' signifies a hexadecimal digit.

### 26.1 Read-Modify-Write Operations

Any instruction that specifies a file register as part of the instruction performs a Read-Modify-Write (R-M-W) operation. The register is read, the data is modified, and the result is stored according to either the instruction, or the destination designator 'd'. A read operation is performed on a register even if the instruction writes to that register.

**TABLE 26-1: OPCODE FIELD DESCRIPTIONS**

| Field | Description |
|---|---|
| f | Register file address (0x00 to 0x7F) |
| W | Working register (accumulator) |
| b | Bit address within an 8-bit file register |
| k | Literal field, constant data or label |
| x | Don't care location (= 0 or 1).<br>The assembler will generate code with x = 0.<br>It is the recommended form of use for compatibility with all Microchip software tools. |
| d | Destination select; d = 0: store result in W,<br>d = 1: store result in file register f.<br>Default is d = 1. |
| n | FSR or INDF number. (0-1) |
| mm | Pre-post increment-decrement mode selection |

**TABLE 26-2: ABBREVIATION DESCRIPTIONS**

| Field | Description |
|---|---|
| PC | Program Counter |
| $\overline{TO}$ | Time-Out bit |
| C | Carry bit |
| DC | Digit Carry bit |
| Z | Zero bit |
| $\overline{PD}$ | Power-Down bit |

**TABLE 26-3: ENHANCED MID-RANGE INSTRUCTION SET**

| Mnemonic, Operands | | Description | Cycles | 14-Bit Opcode MSb | | | LSb | Status Affected | Notes |
|---|---|---|---|---|---|---|---|---|---|
| **BYTE-ORIENTED FILE REGISTER OPERATIONS** | | | | | | | | | |
| ADDWF | f, d | Add W and f | 1 | 00 | 0111 | dfff | ffff | C, DC, Z | 2 |
| ADDWFC | f, d | Add with Carry W and f | 1 | 11 | 1101 | dfff | ffff | C, DC, Z | 2 |
| ANDWF | f, d | AND W with f | 1 | 00 | 0101 | dfff | ffff | Z | 2 |
| ASRF | f, d | Arithmetic Right Shift | 1 | 11 | 0111 | dfff | ffff | C, Z | 2 |
| LSLF | f, d | Logical Left Shift | 1 | 11 | 0101 | dfff | ffff | C, Z | 2 |
| LSRF | f, d | Logical Right Shift | 1 | 11 | 0110 | dfff | ffff | C, Z | 2 |
| CLRF | f | Clear f | 1 | 00 | 0001 | lfff | ffff | Z | 2 |
| CLRW | – | Clear W | 1 | 00 | 0001 | 0000 | 00xx | Z | |
| COMF | f, d | Complement f | 1 | 00 | 1001 | dfff | ffff | Z | 2 |
| DECF | f, d | Decrement f | 1 | 00 | 0011 | dfff | ffff | Z | 2 |
| INCF | f, d | Increment f | 1 | 00 | 1010 | dfff | ffff | Z | 2 |
| IORWF | f, d | Inclusive OR W with f | 1 | 00 | 0100 | dfff | ffff | Z | 2 |
| MOVF | f, d | Move f | 1 | 00 | 1000 | dfff | ffff | Z | 2 |
| MOVWF | f | Move W to f | 1 | 00 | 0000 | 1fff | ffff | | 2 |
| RLF | f, d | Rotate Left f through Carry | 1 | 00 | 1101 | dfff | ffff | C | 2 |
| RRF | f, d | Rotate Right f through Carry | 1 | 00 | 1100 | dfff | ffff | C | 2 |
| SUBWF | f, d | Subtract W from f | 1 | 00 | 0010 | dfff | ffff | C, DC, Z | 2 |
| SUBWFB | f, d | Subtract with Borrow W from f | 1 | 11 | 1011 | dfff | ffff | C, DC, Z | 2 |
| SWAPF | f, d | Swap nibbles in f | 1 | 00 | 1110 | dfff | ffff | | 2 |
| XORWF | f, d | Exclusive OR W with f | 1 | 00 | 0110 | dfff | ffff | Z | 2 |
| **BYTE ORIENTED SKIP OPERATIONS** | | | | | | | | | |
| DECFSZ | f, d | Decrement f, Skip if 0 | 1(2) | 00 | 1011 | dfff | ffff | | 1, 2 |
| INCFSZ | f, d | Increment f, Skip if 0 | 1(2) | 00 | 1111 | dfff | ffff | | 1, 2 |
| **BIT-ORIENTED FILE REGISTER OPERATIONS** | | | | | | | | | |
| BCF | f, b | Bit Clear f | 1 | 01 | 00bb | bfff | ffff | | 2 |
| BSF | f, b | Bit Set f | 1 | 01 | 01bb | bfff | ffff | | 2 |
| **BIT-ORIENTED SKIP OPERATIONS** | | | | | | | | | |
| BTFSC | f, b | Bit Test f, Skip if Clear | 1 (2) | 01 | 10bb | bfff | ffff | | 1, 2 |
| BTFSS | f, b | Bit Test f, Skip if Set | 1 (2) | 01 | 11bb | bfff | ffff | | 1, 2 |
| **LITERAL OPERATIONS** | | | | | | | | | |
| ADDLW | k | Add literal and W | 1 | 11 | 1110 | kkkk | kkkk | C, DC, Z | |
| ANDLW | k | AND literal with W | 1 | 11 | 1001 | kkkk | kkkk | Z | |
| IORLW | k | Inclusive OR literal with W | 1 | 11 | 1000 | kkkk | kkkk | Z | |
| MOVLB | k | Move literal to BSR | 1 | 00 | 0000 | 001k | kkkk | | |
| MOVLP | k | Move literal to PCLATH | 1 | 11 | 0001 | 1kkk | kkkk | | |
| MOVLW | k | Move literal to W | 1 | 11 | 0000 | kkkk | kkkk | | |
| SUBLW | k | Subtract W from literal | 1 | 11 | 1100 | kkkk | kkkk | C, DC, Z | |
| XORLW | k | Exclusive OR literal with W | 1 | 11 | 1010 | kkkk | kkkk | Z | |

**Note 1:** If the Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

**2:** If this instruction addresses an INDF register and the MSb of the corresponding FSR is set, this instruction will require one additional instruction cycle.

## 27.2 Standard Operating Conditions

The standard operating conditions for any device are defined as:

Operating Voltage: $V_{DDMIN} \leq V_{DD} \leq V_{DDMAX}$
Operating Temperature: $T_{A\_MIN} \leq T_A \leq T_{A\_MAX}$

**$V_{DD}$ — Operating Supply Voltage(1)**

PIC16LF1574/5/8/9

- $V_{DDMIN}$ (Fosc ≤ 16 MHz) ..... +1.8V
- $V_{DDMIN}$ (Fosc ≤ 32 MHz) ..... +2.5V
- $V_{DDMAX}$ ..... +3.6V

PIC16F1574/5/8/9

- $V_{DDMIN}$ (Fosc ≤ 16 MHz) ..... +2.3V
- $V_{DDMIN}$ (Fosc ≤ 32 MHz) ..... +2.5V
- $V_{DDMAX}$ ..... +5.5V

**$T_A$ — Operating Ambient Temperature Range**

Industrial Temperature

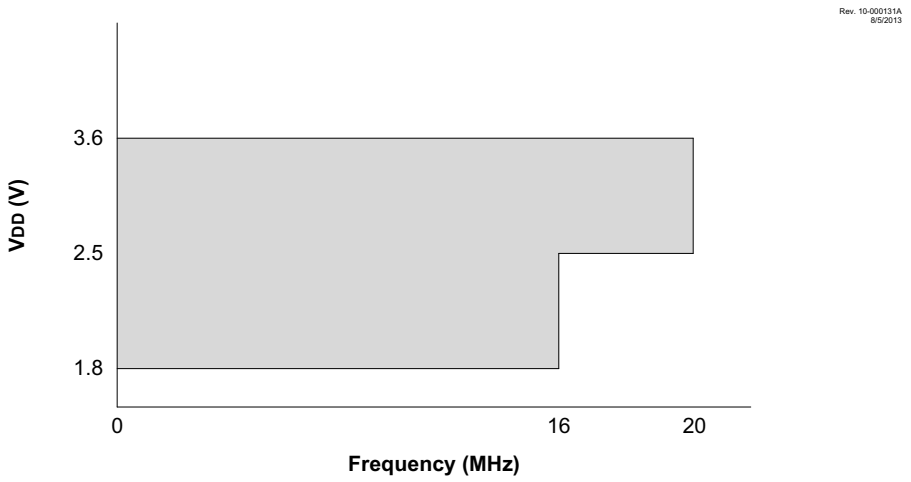
- $T_{A\_MIN}$ ..... -40°C
- $T_{A\_MAX}$ ..... +85°C

Extended Temperature

- $T_{A\_MIN}$ ..... -40°C
- $T_{A\_MAX}$ ..... +125°C

**Note 1:** See Parameter D001, DS Characteristics: Supply Voltage.

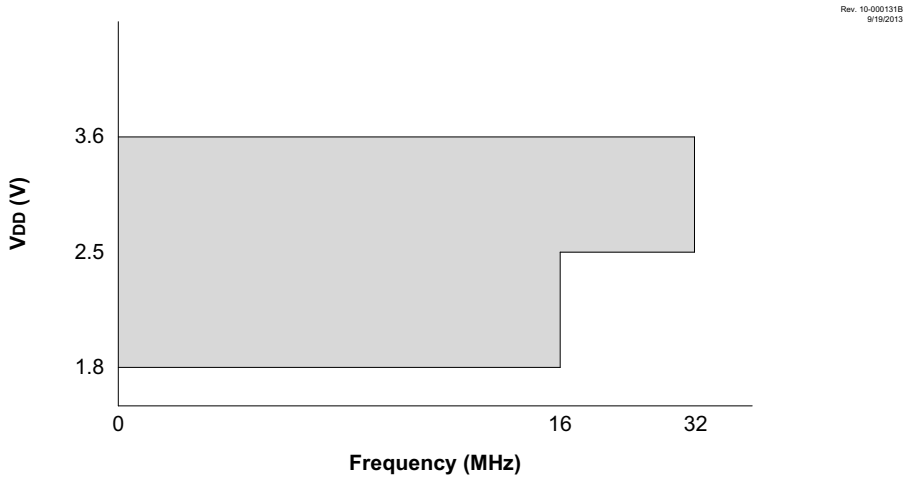
**FIGURE 27-1: VOLTAGE FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C, PIC16F1574/5/8/9 ONLY**




**Note 1: T**he shaded region indicates the permissible combinations of voltage and frequency.

**2: R**efer to Table 27-7 for each Oscillator mode's supported frequencies.

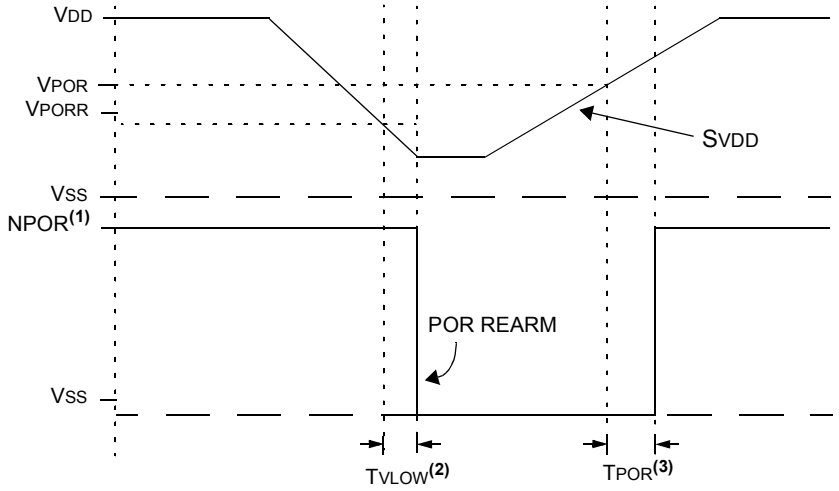
**FIGURE 27-2: VOLTAGE FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C, PIC16LF1574/5/8/9 ONLY**




**Note 1: T**he shaded region indicates the permissible combinations of voltage and frequency.

**2: R**efer to Table 27-7 for each Oscillator mode's supported frequencies.

**FIGURE 27-3: POR AND POR REARM WITH SLOW RISING VDD**




**Note 1:** When NPOR is low, the device is held in Reset.
**2:** TPOR 1 μs typical.
**3:** TVLOW 2.7 μs typical.

## 27.4 AC Characteristics

Timing Parameter Symbology has been created with one of the following formats:

1. TppS2ppS
2. TppS

| T | | | |
|---|---|---|---|
| F | Frequency | T | Time |

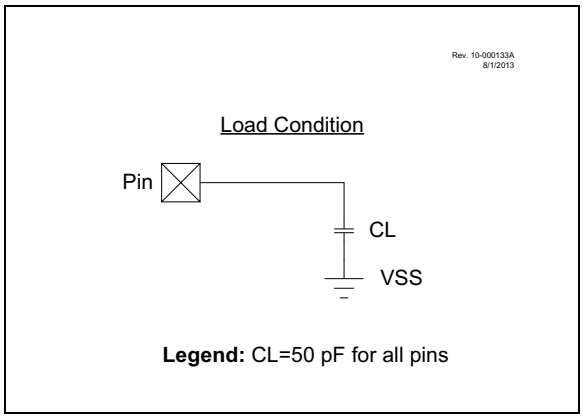
Lowercase letters (pp) and their meanings:

| pp | | | |
|---|---|---|---|
| cc | CCP1 | osc | CLKIN |
| ck | CLKOUT | rd | $\overline{RD}$ |
| cs | $\overline{CS}$ | rw | $\overline{RD}$ or $\overline{WR}$ |
| di | SDIx | sc | SCKx |
| do | SDO | ss | $\overline{SS}$ |
| dt | Data in | t0 | T0CKI |
| io | I/O PORT | t1 | T1CKI |
| mc | $\overline{MCLR}$ | wr | $\overline{WR}$ |

Uppercase letters and their meanings:

| S | | | |
|---|---|---|---|
| F | Fall | P | Period |
| H | High | R | Rise |
| I | Invalid (High-impedance) | V | Valid |
| L | Low | Z | High-impedance |

**FIGURE 27-4: LOAD CONDITIONS**




**Legend:** CL=50 pF for all pins

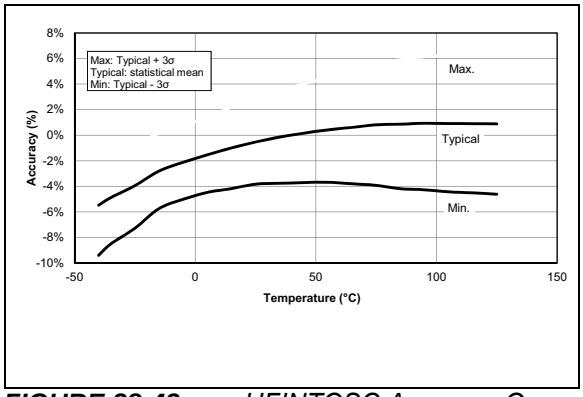



***FIGURE 28-43:*** *HFINTOSC Accuracy Over Temperature, VDD = 1.8V, LF Devices Only.*

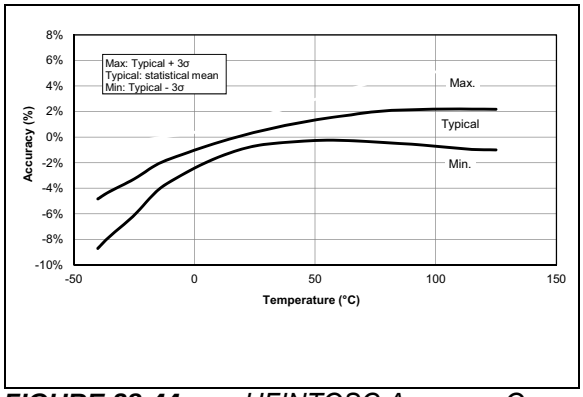



***FIGURE 28-44:*** *HFINTOSC Accuracy Over Temperature, 2.3V ≤ VDD ≤ 5.5V.*

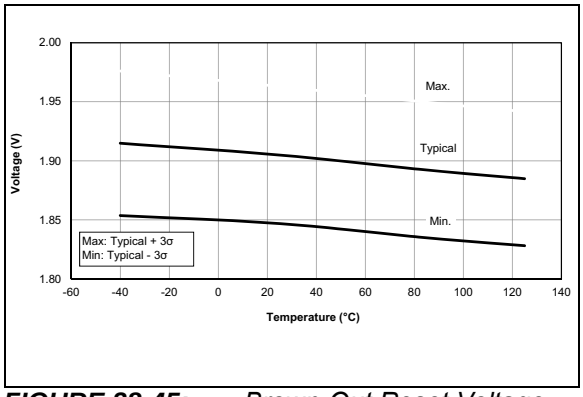



***FIGURE 28-45:*** *Brown-Out Reset Voltage, BORV = 1, PIC16LF1574/5/8/9 Only.*

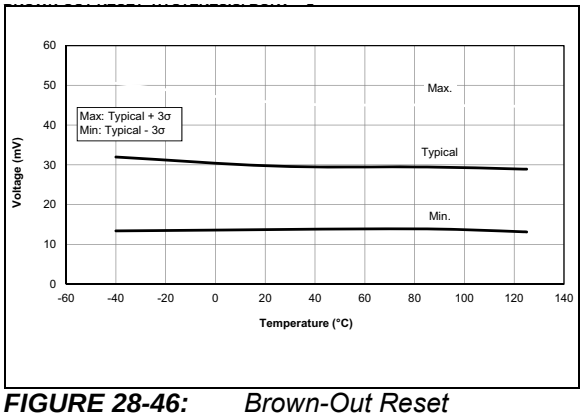



***FIGURE 28-46:*** *Brown-Out Reset Hysteresis, BORV = 1, PIC16LF1574/5/8/9 Only.*

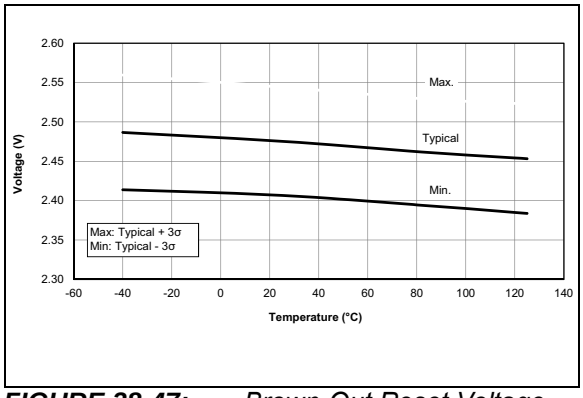



***FIGURE 28-47:*** *Brown-Out Reset Voltage, BORV = 1, PIC16F1574/5/8/9 Only.*

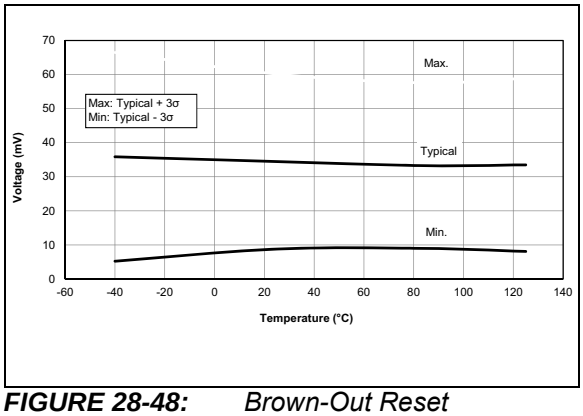



***FIGURE 28-48:*** *Brown-Out Reset Hysteresis, BORV = 1, PIC16F1574/5/8/9 Only.*

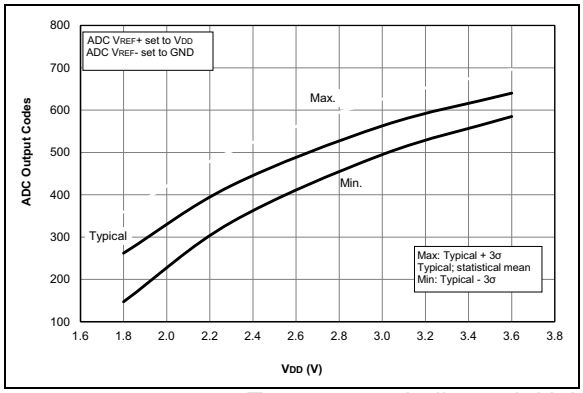



***FIGURE 28-67:*** *Temperature Indicator Initial Offset, Low Range, Temp = 20°C, LF Devices Only.*

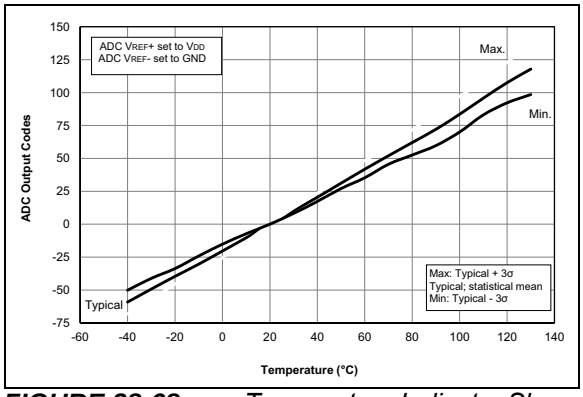



***FIGURE 28-68:*** *Temperature Indicator Slope Normalized TO 20°C, High Range, VDD = 5.5V, F Devices Only.*

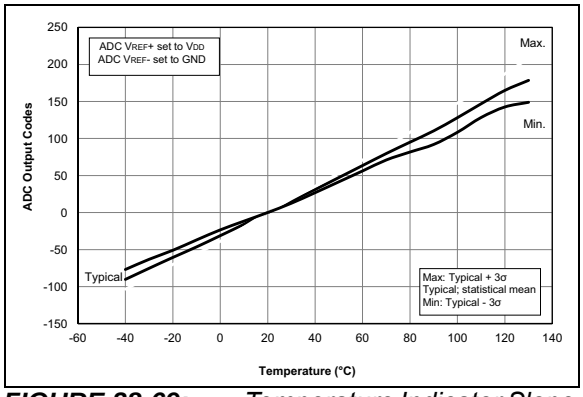



***FIGURE 28-69:*** *Temperature Indicator Slope Normalized TO 20°C, High Range, VDD = 3.6V, F Devices Only.*

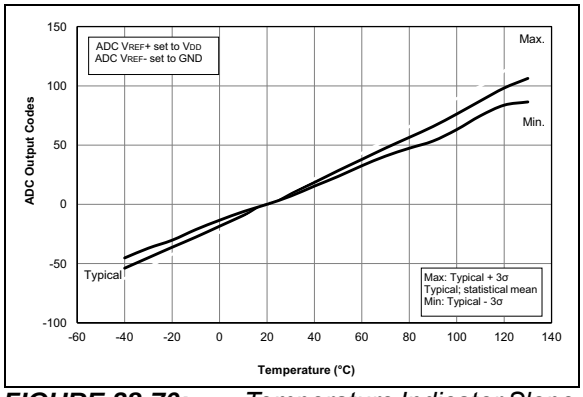



***FIGURE 28-70:*** *Temperature Indicator Slope Normalized TO 20°C, Low Range, VDD = 3.0V, F Devices Only.*

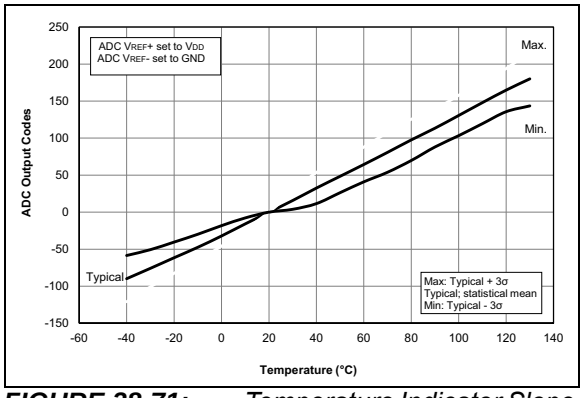



***FIGURE 28-71:*** *Temperature Indicator Slope Normalized TO 20°C, Low Range, VDD = 1.8V, LF Devices Only.*

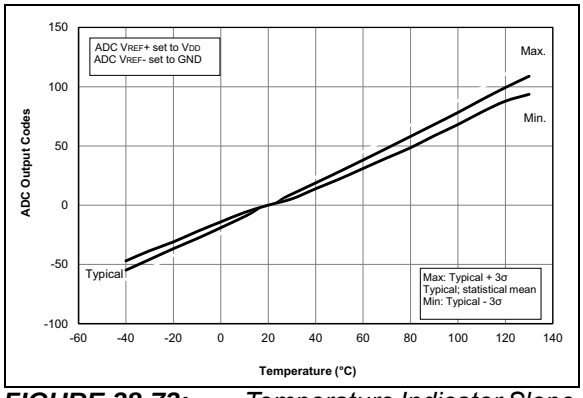



***FIGURE 28-72:*** *Temperature Indicator Slope Normalized TO 20°C, Low Range, VDD = 3.0V, LF Devices Only.*

## 29.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers (MCU) and dsPIC® digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® X IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB XC Compiler
  - MPASM™ Assembler
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB X SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers/Programmers
  - MPLAB ICD 3
  - PICkit™ 3
- Device Programmers
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits and Starter Kits
- Third-party development tools

### 29.1 MPLAB X Integrated Development Environment Software

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows®, Linux and Mac OS® X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- Call graph window

Project-Based Workspaces:

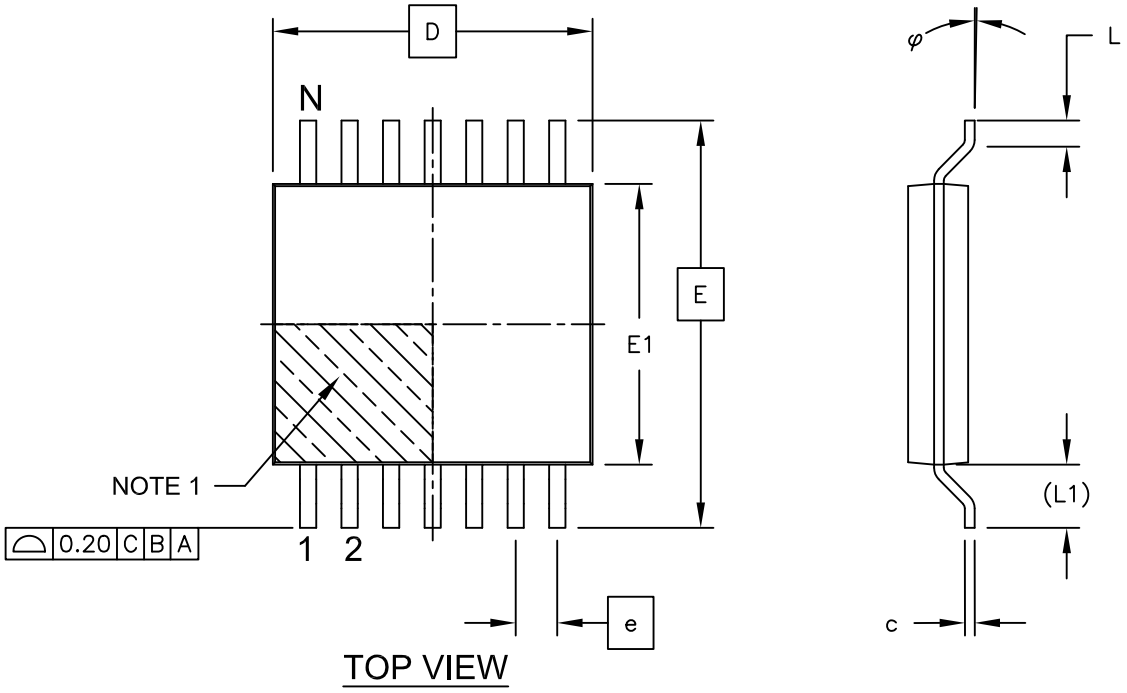
- Multiple projects
- Multiple tools
- Multiple configurations
- Simultaneous debugging sessions

File History and Bug Tracking:

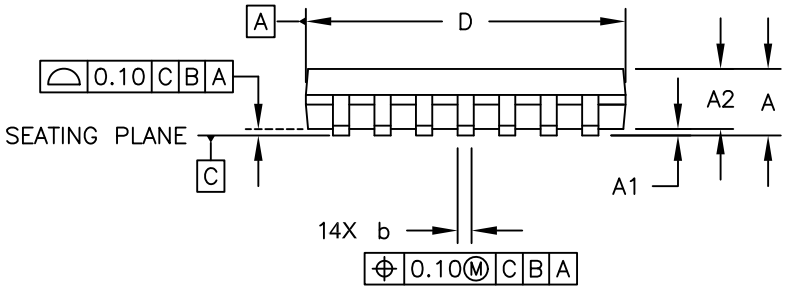
- Local file history feature
- Built-in support for Bugzilla issue tracker

### 14-Lead Plastic Thin Shrink Small Outline (ST) - 4.4 mm Body [TSSOP]

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




TOP VIEW




SIDE VIEW

Microchip Technology Drawing C04-087C Sheet 1 of 2