**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 | 256 x 8 |
| RAM Size | 512 x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.3V ~ 5.5V |
| Data Converters | A/D 11x10b; D/A 1x5b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 125°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 14-SOIC (0.154", 3.90mm Width) |
| Supplier Device Package | 14-SOIC |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16f18324-e-sl |

**TABLE 1-2: PIC16(L)F18324 PINOUT DESCRIPTION**

| Name | Function | Input Type | Output Type | Description |
|---|---|---|---|---|
| RA0/ANA0/C1IN0+/DAC1OUT/ ICDDAT/ICSPDAT | RA0 | TTL/ST | CMOS | General purpose I/O. |
| | ANA0 | AN | — | ADC Channel A0 input. |
| | C1IN0+ | AN | — | Comparator C1 positive input. |
| | DAC1OUT | — | AN | Digital-to-Analog Converter output. |
| | ICDDAT | TTL/ST | CMOS | In-Circuit Debug Data I/O. |
| | ICSPDAT | TTL/ST | CMOS | ICSP™ Data I/O. |
| RA1/ANA1/VREF+/C1IN0-/ C2IN0-/DAC1REF+/ ICDCLK/ ICSPCLK | RA1 | TTL/ST | CMOS | General purpose I/O. |
| | ANA1 | AN | — | ADC Channel A1 input. |
| | VREF+ | AN | — | ADC positive voltage reference input. |
| | C1IN0- | AN | — | Comparator C1 negative input. |
| | C2IN0- | AN | — | Comparator C2 negative input. |
| | DAC1REF+ | — | AN | Digital-to-Analog Converter positive reference input. |
| | ICDCLK | TTL/ST | CMOS | In-Circuit Debug Clock I/O. |
| | ICSPCLK | TTL/ST | CMOS | ICSP Clock I/O. |
| RA2/ANA2/VREF-/ DAC1REF-/ T0CKI(1)/ CCP3(1)/CWG1IN(1)/ CWG2IN(1)/INT(1) | RA2 | TTL/ST | CMOS | General purpose I/O. |
| | ANA2 | AN | — | ADC Channel A2 input. |
| | VREF- | AN | — | ADC negative voltage reference input. |
| | DAC1REF- | — | AN | Digital-to-Analog Converter negative reference input. |
| | T0CKI | TTL/ST | — | TMR0 Clock input. |
| | CCP3 | TTL/ST | CMOS | Capture/Compare/PWM 3 input. |
| | CWG1IN | TTL/ST | — | Complementary Waveform Generator 1 input. |
| | CWG2IN | TTL/ST | — | Complementary Waveform Generator 2 input. |
| | INT | TTL/ST | — | External interrupt input. |
| RA3/$\overline{\text{MCLR}}$/VPP | RA3 | TTL/ST | CMOS | General purpose I/O. |
| | $\overline{\text{MCLR}}$ | TTL/ST | — | Master Clear with internal pull-up. |
| | VPP | HV | — | Programming voltage. |
| RA4/ANA4/T1G(1)/ SOSCO/ CLKOUT/OSC2 | RA4 | TTL/ST | CMOS | General purpose I/O. |
| | ANA4 | AN | — | ADC Channel A4 input. |
| | T1G | ST | — | TMR1 gate input. |
| | SOSCO | — | XTAL | Secondary Oscillator connection. |
| | CLKOUT | — | CMOS | FOSC/4 output. |
| | OSC2 | — | XTAL | Crystal/Resonator (LP, XT, HS modes). |
| RA5/ANA5/T1CKI(1)/ SOSCIN/ SOSCI/ CLCIN3(1)/CLKIN/ OSC1 | RA5 | TTL/ST | CMOS | General purpose I/O. |
| | ANA5 | AN | — | ADC Channel A5 input. |
| | T1CKI | TTL/ST | — | TMR1 Clock input. |
| | SOSCIN | TTL/ST | — | Secondary Oscillator input connection. |
| | SOSCI | XTAL | — | Secondary Oscillator connection. |
| | CLCIN3 | TTL/ST | — | Configurable Logic Cell 3 input. |
| | CLKIN | TTL/ST | — | External clock input. |
| | OSC1 | XTAL | — | Crystal/Resonator (LP, XT, HS modes). |

**Legend:** AN = Analog input or output CMOS = CMOS compatible input or output OD = Open-Drain
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels $I^2C$ = Schmitt Trigger input with $I^2C$
HV = High Voltage XTAL = Crystal levels

**Note 1:** Default peripheral input. Input can be moved to any other pin with the PPS input selection registers. See Register 13-1.
**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 13-2.
**3:** These $I^2C$ functions are bidirectional. The output pin selections must be the same as the input pin selections.

## 6.10 Start-up Sequence

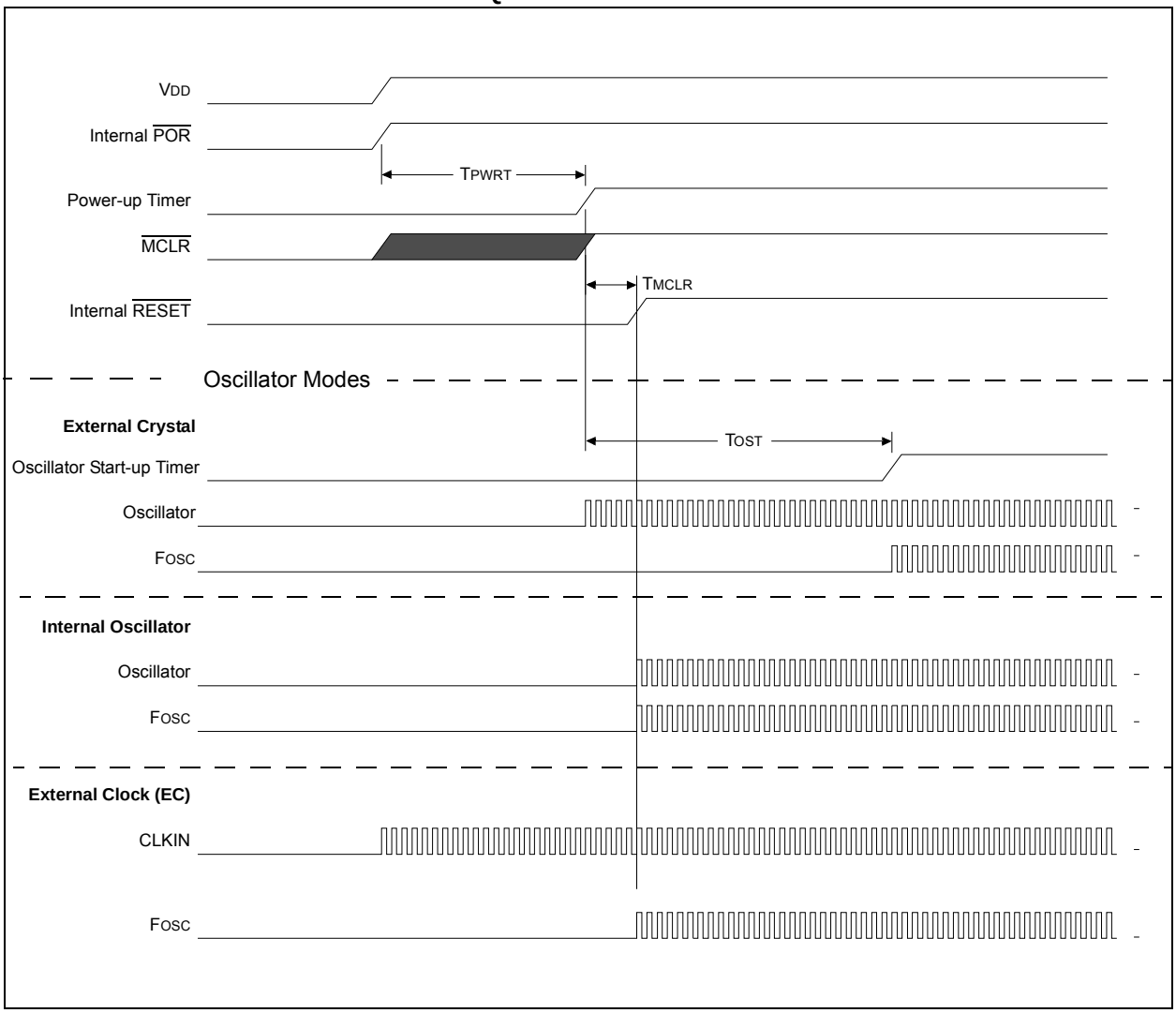
Upon the release of a POR or BOR, the following must occur before the device will begin executing:

1. Power-up Timer runs to completion (if enabled).
2. $\overline{\text{MCLR}}$ must be released (if enabled).
3. Oscillator start-up timer runs to completion (if required for oscillator source).

The total time-out will vary based on oscillator configuration and Power-up Timer Configuration. See **Section 7.0 "Oscillator Module"** for more information.

The Power-up Timer and oscillator start-up timer run independently of $\overline{\text{MCLR}}$ Reset. If $\overline{\text{MCLR}}$ is kept low long enough, the Power-up Timer will expire. Upon bringing $\overline{\text{MCLR}}$ high, the device will begin execution after ten FOSC cycles (see Figure 6-3). This is useful for testing purposes or to synchronize more than one device operating in parallel.

**FIGURE 6-3: RESET START-UP SEQUENCE**



## 7.5 Register Definitions: Oscillator Control

**REGISTER 7-1: OSCCON1: OSCILLATOR CONTROL REGISTER1**

| U-0 | R/W-f/f(1) | R/W-f/f(1) | R/W-f/f(1) | R/W-q/q(4) | R/W-q/q(4) | R/W-q/q(4) | R/W-q/q(4) |
|---|---|---|---|---|---|---|---|
| — | NOSC<2:0>(2,3) | | | NDIV<3:0>(2,3) | | | |
| 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 | f = determined by fuse setting |
| q = Reset value is determined by hardware | | |

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **NOSC<2:0>:** New Oscillator Source Request bits
The setting requests a source oscillator and PLL combination per Table 7-1.
POR value = RSTOSC (Register 5-1).

bit 3-0 **NDIV<3:0>:** New Divider Selection Request bits
The setting determines the new postscaler division ratio per Table 7-1.

**Note 1:** The default value (f/f) is set equal to the RSTOSC Configuration bits.
**2:** If NOSC is written with a reserved value (Table 7-1), the HFINTOSC will be automatically selected as the clock source.
**3:** When CSWEN = 0, this register is read-only and cannot be changed from the POR value.
**4:** When RSTOSC = 110 (HFINTOSC 1 MHz) the NDIV bits will default to ‘0010’ upon Reset; for all other NOSC settings the NVID bits will default to ‘0000’ upon Reset.

**REGISTER 7-2: OSCCON2: OSCILLATOR CONTROL REGISTER 2**

| U-0 | R-q/q(1) | R-q/q(1) | R-q/q(1) | R-q/q1) | R-q/q(1) | R-q/q(1) | R-q/q(1) |
|---|---|---|---|---|---|---|---|
| — | COSC<2:0> | | | CDIV<3:0> | | | |
| bit 7 | | | | | | | bit 0 |

| Legend: | | |
|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as ‘0’ |
| u = Bit is unchanged | x = Bit is unknown | -n/n = Value at POR and BOR/Value at all other Resets |
| ‘1’ = Bit is set | ‘0’ = Bit is cleared | q = Reset value is determined by hardware |

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **COSC<2:0>:** Current Oscillator Source Select bits (read-only)
Indicates the current source oscillator and PLL combination per Table 7-1.

bit 3-0 **CDIV<3:0>:** Current Divider Select bits (read-only)
Indicates the current postscaler division ratio per Table 7-1.

**Note 1:** The Reset value (q/q) will match the NOSC<2:0>/NDIV<3:0> bits.

**REGISTER 8-5: PIE3: PERIPHERAL INTERRUPT ENABLE REGISTER 3**

| 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 |
|---|---|---|---|---|---|---|---|
| OSFIE | CSWIE | TMR3GIE | TMR3IE | CLC4IE | CLC3IE | CLC2IE | CLC1IE |
| 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 **OSFIE**: Oscillator Fail Interrupt Enable bit.
1 = Enables the Oscillator Fail interrupt
0 = Disables the Oscillator Fail interrupt

bit 6 **CSWIE:** Clock Switch Complete Interrupt Enable bit
1 = The clock switch module interrupt is enabled
0 = The clock switch module interrupt is not enabled

bit 5 **TMR3GIE:** Timer3 Gate Interrupt Enable bit
1 = Timer3 Gate interrupt is enabled
0 = Timer3 Gate interrupt is not enabled

bit 4 **TMR3IE:** TMR3 overflow Interrupt Enable bit
1 = TMR3 overflow interrupt is enabled
0 = TMR3 overflow interrupt is not enabled

bit 3 **CLC4IE:** CLC4 Interrupt Flag bit
1 = CLC4 interrupt is enabled
0 = CLC4 interrupt is not enabled

bit 2 **CLC3IE:** CLC3 Interrupt Flag bit
1 = CLC3 interrupt is enabled
0 = CLC3 interrupt is not enabled

bit 1 **CLC2IE:** CLC2 Interrupt Enable bit
1 = CLC2 interrupt enabled
0 = CLC2 interrupt disabled

bit 0 **CLC1IE:** CLC1 Interrupt Enable bit
1 = CLC1 interrupt enabled
0 = CLC1 interrupt disabled

**Note:** Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

## 11.2 Data EEPROM

Data EEPROM consists of 256 bytes of user data memory. The EEPROM provides storage locations for 8-bit user defined data.

EEPROM can be read and/or written through:

- FSR/INDF indirect access (**Section 11.3 “FSR and INDF Access”**)
- NVMREG access (**Section 11.4 “NVMREG Access”**)
- External device programmer

Unlike Program Flash Memory, which must be written to by row, EEPROM can be written to byte by byte.

## 11.3 FSR and INDF Access

The FSR and INDF registers allow indirect access to the Program Flash Memory or EEPROM.

### 11.3.1 FSR READ

With the intended address loaded into an FSR register, a `MOVIW` instruction or read of INDF will read data from the Program Flash Memory or EEPROM. The CPU operation is suspended during the read, and resumes immediately after. Read operations return a single word of memory. When the MSB of the FSR (ex: FSRxH) is set to 0x70, the lower 8-bit address value (in FSRxL) determines the EEPROM location that may be read from (through the INDF register). In other words, the EEPROM address range 0x00-0xFF is mapped into the FSR address space between 0x7000-0x70FF. Writing to the EEPROM cannot be accomplished via the FSR/INDF interface.

### 11.3.2 FSR WRITE

Writing/erasing the NVM through the FSR registers (ex. `MOVWI` instruction) is not supported in the PIC16(L)F18324/18344 devices.

## 12.7 Register Definitions: PORTC

**REGISTER 12-17: PORTC: PORTC REGISTER**

| R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u |
|---|---|---|---|---|---|---|---|
| RC7(1) | RC6(1) | RC5 | RC4 | RC3 | RC2 | RC1 | RC0 |
| 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 **RC<7:6>**: PORTC I/O Value bits(1,2)
1 = Port pin is ≥ VIH
0 = Port pin is ≤ VIL

bit 5-0 **RC<5:0>**: PORTC General Purpose I/O Pin bits(2)
1 = Port pin is ≥ VIH
0 = Port pin is ≤ VIL

**Note 1:** PIC16(L)F18344 only; otherwise read as '0'.
**2:** Writes to PORTC are actually written to corresponding LATC register. Reads from PORTC register is return of actual I/O pin values.

## 18.9 Analog Input Connection Considerations

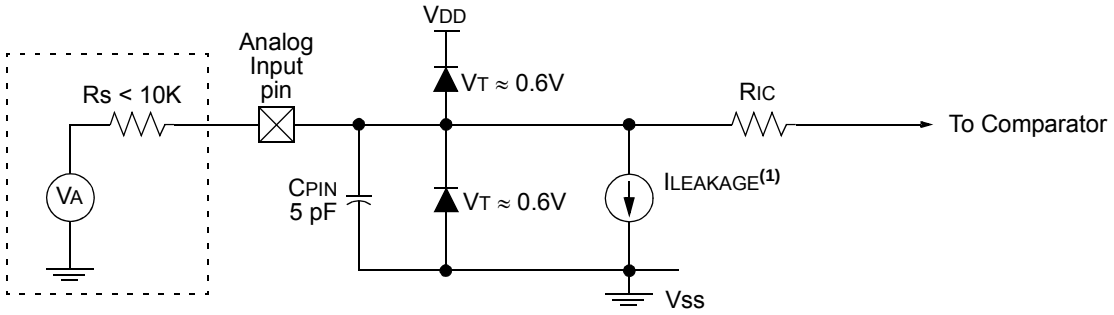
A simplified circuit for an analog input is shown in Figure 18-3. Since the analog input pins share their connection with a digital input, they have reverse biased ESD protection diodes to $V_{DD}$ and $V_{SS}$. The analog input, therefore, must be between $V_{SS}$ and $V_{DD}$. If the input voltage deviates from this range by more than 0.6V in either direction, one of the diodes is forward biased and a latch-up may occur.

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

**Note 1:** When reading a PORT register, all pins configured as analog inputs will read as a '0'. Pins configured as digital inputs will provide an input based on their level as either a TTL or ST input buffer.

**2:** Analog levels on any pin defined as a digital input, may cause the input buffer to consume more current than is specified.

**FIGURE 18-3: ANALOG INPUT MODEL**




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

**Note 1:** See I/O Ports in Table 35-4.

**TABLE 22-3: SUMMARY OF REGISTERS ASSOCIATED WITH ADC**

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---|---|---|---|---|---|---|---|---|---|
| INTCON | GIE | PEIE | — | — | — | — | — | INTEDG | 98 |
| PIE1 | TMR1GIE | ADIE | RCIE | TXIE | SSP1IE | BCL1IE | TMR2IE | TMR1IE | 100 |
| PIR1 | TMR1GIF | ADIF | RCIF | TXIF | SSP1IF | BCL1IF | TMR2IF | TMR1IF | 105 |
| TRISA | — | — | TRISA5 | TRISA4 | —(2) | TRISA2 | TRISA1 | TRISA0 | 140 |
| TRISB(1) | TRISB7 | TRISB6 | TRISB5 | TRISB4 | — | — | — | — | 146 |
| TRISC | TRISC7(1) | TRISC6(1) | TRISC5 | TRISC4 | TRISC3 | TRISC2 | TRISC1 | TRISC0 | 153 |
| ANSELA | — | — | ANSA5 | ANSA4 | — | ANSA2 | ANSA1 | ANSA0 | 141 |
| ANSELB(1) | ANSB7 | ANSB6 | ANSB5 | ANSB4 | — | — | — | — | 147 |
| ANSELC | ANSC7(1) | ANSC6(1) | ANSC5 | ANSC4 | ANSC3 | ANSC2 | ANSC1 | ANSC0 | 154 |
| ADCON0 | CHS<5:0> | | | | | | GO/$\overline{\text{DONE}}$ | ADON | 241 |
| ADCON1 | ADFM | ADCS<2:0> | | | — | ADNREF | ADPREF<1:0> | | 242 |
| ADACT | — | — | — | ADACT<4:0> | | | | | 243 |
| ADRESH | ADRESH<7:0> | | | | | | | | 244 |
| ADRESL | ADRESL<7:0> | | | | | | | | 244 |
| FVRCON | FVREN | FVRRDY | TSEN | TSRNG | CDAFVR<1:0> | | ADFVR<1:0> | | 177 |
| DAC1CON1 | — | — | — | DAC1R<4:0> | | | | | 261 |
| OSCSTAT1 | EXTOR | HFOR | — | LFOR | SOR | ADOR | — | PLLR | 90 |

**Legend:** – = unimplemented read as '0'. Shaded cells are not used for the ADC module.

**Note 1:** PIC16(L)F18344 only.

**2:** Unimplemented, read as '1'.

## 23.5 Interrupts

When the accumulator overflows (NCO_overflow), the NCO1 Interrupt Flag bit, NCO1IF, of the PIR2 register is set. To enable the interrupt event (NCO_interrupt), the following bits must be set:

- N1EN bit of the NCO1CON register
- NCO1IE bit of the PIE2 register
- PEIE bit of the INTCON register
- GIE bit of the INTCON register

The interrupt must be cleared by software by clearing the NCO1IF bit in the Interrupt Service Routine.

## 23.6 Effects of a Reset

All of the NCO1 registers are cleared to zero as the result of a Reset.

## 23.7 Operation in Sleep

The NCO1 module operates independently from the system clock and will continue to run during Sleep, provided that the clock source selected remains active.

The HFINTOSC remains active during Sleep when the NCO1 module is enabled and the HFINTOSC is selected as the clock source, regardless of the system clock source selected.

In other words, if the HFINTOSC is simultaneously selected as the system clock and the NCO1 clock source, when the NCO1 is enabled, the CPU will go idle during Sleep, but the NCO1 will continue to operate and the HFINTOSC will remain active.

This will have a direct effect on the Sleep mode current.

**REGISTER 23-2: NCO1CLK: NCO1 INPUT CLOCK CONTROL REGISTER**

| R/W-0/0 | R/W-0/0 | R/W-0/0 | U-0 | U-0 | U-0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| N1PWS<2:0> | | | — | — | — | N1CKS<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 7-5 **N1PWS<2:0>:** NCO1 Output Pulse-Width Select bits(1, 2)
- 000 = NCO1 output is active for 1 input clock period
- 001 = NCO1 output is active for 2 input clock periods
- 010 = NCO1 output is active for 4 input clock periods
- 011 = NCO1 output is active for 8 input clock periods
- 100 = NCO1 output is active for 16 input clock periods
- 101 = NCO1 output is active for 32 input clock periods
- 110 = NCO1 output is active for 64 input clock periods
- 111 = NCO1 output is active for 128 input clock periods

bit 4-2 **Unimplemented:** Read as ‘0’

bit 1-0 **N1CKS<1:0>:** NCO1 Clock Source Select bits
- 00 = HFINTOSC (16 MHz)
- 01 = FOSC
- 10 = CLC1OUT
- 11 =Reserved

**Note 1:** N1PWS applies only when operating in Pulse Frequency mode.

**2:** If NCO pulse width is greater than NCO overflow period, the NCO1 output does not toggle.

**REGISTER 23-3: NCO1ACCL: NCO1 ACCUMULATOR REGISTER – LOW BYTE**

| 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 |
|---|---|---|---|---|---|---|---|
| NCO1ACC<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 7-0 **NCO1ACC<7:0>:** NCO1 Accumulator, low byte

**REGISTER 23-7: NCO1INCH(1): NCO1 INCREMENT REGISTER – HIGH BYTE**

| 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 |
|---|---|---|---|---|---|---|---|
| NCO1INC<15:8> | | | | | | | |
| bit 7 | | | | | | | bit 0 |

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

bit 7-0 **NCO1INC<15:8>:** NCO1 Increment, high byte

**Note 1:** The logical increment spans NCO1INCU:NCO1INCH:NCO1INCL.

**REGISTER 23-8: NCO1INCU(1): NCO1 INCREMENT REGISTER – UPPER BYTE**

| U-0 | U-0 | U-0 | U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | NCO1INC<19:16> | | | |
| 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 7-4 **Unimplemented:** Read as ‘0’

bit 3-0 **NCO1INC<19:16>:** NCO1 Increment, upper byte

**Note 1:** The logical increment spans NCO1INCU:NCO1INCH:NCO1INCL.

**REGISTER 24-2: DACCON1: VOLTAGE REFERENCE CONTROL REGISTER 1**

| U-0 | U-0 | U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| — | — | — | DAC1R<4:0> | | | | |
| bit 7 | | | | | | | bit 0 |

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

bit 7-5 **Unimplemented:** Read as '0'

bit 4-0 **DAC1R<4:0>:** DAC1 Voltage Output Select bits

$V_{OUT} = (V_{SRC+} - V_{SRC-})*(DAC1R<4:0>/32) + V_{SRC}$

**TABLE 24-1: SUMMARY OF REGISTERS ASSOCIATED WITH THE DAC1 MODULE**

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on page |
|---|---|---|---|---|---|---|---|---|---|
| DACCON0 | DAC1EN | — | DAC1OE | — | DAC1PSS<1:0> | | — | DAC1NSS | 260 |
| DACCON1 | — | — | — | DAC1R<4:0> | | | | | 261 |
| CMxCON1 | CxINTP | CxINTN | CxPCH<2:0> | | | CxNCH<2:0> | | | 188 |
| ADCON0 | CHS<5:0> | | | | | | GO/$\overline{\text{DONE}}$ | ADON | 241 |

**Legend:** — = Unimplemented location, read as '0'. Shaded cells are not used with the DAC module.

## 27.0 TIMER1/3/5 MODULE WITH GATE CONTROL

Timer1/3/5 modules are 16-bit timers/counters, each with the following features:

- 16-bit timer/counter register pair (TMR1H:TMR1L)
- Programmable internal or external clock source
- 2-bit prescaler
- Clock source for optional comparator synchronization
- Multiple Timer1 gate (count enable) sources
- Interrupt on overflow
- Wake-up on overflow (external clock, Asynchronous mode only)
- Time base for the Capture/Compare function with the CCP modules
- Auto-conversion Trigger (with CCP)
- Selectable Gate Source Polarity
- Gate Toggle mode
- Gate Single-pulse mode
- Gate Value Status
- Gate Event Interrupt

Figure 27-1 is a block diagram of the Timer1 module.

| | |
|---|---|
| **Note 1:** | In devices with more than one Timer module, it is very important to pay close attention to the register names used. A number placed after the module acronym is used to distinguish between separate modules. For example, the T1CON and T3CON control the same operational aspects of two completely different Timer modules. |
| **2:** | Throughout this section, generic references to Timer1 module in any of its operating modes may be interpreted as being equally applicable to Timerx module. Register names, module signals, I/O pins and bit names may use the generic designator ‘x’ to indicate the use of a numeral to distinguish a particular module, when required. |

### 27.2.1 TIMER1 (SECONDARY) OSCILLATOR

A dedicated low-power 32.768 kHz oscillator circuit is built-in between pins SOSCI (input) and SOSCO (amplifier output). This internal circuit is designed to be used in conjunction with an external 32.768 kHz crystal.

The oscillator circuit is enabled by setting the T1SOSC bit of the T1CON register. The oscillator will continue to run during Sleep.

| | |
|---|---|
| **Note:** | The oscillator requires a start-up and stabilization time before use. Thus, T1SOSC should be set and a suitable delay observed prior to using Timer1. A suitable delay similar to the OST delay can be implemented in software by clearing the TMR1IF bit then presetting the TMR1H:TMR1L register pair to FC00h. The TMR1IF flag will be set when 1024 clock cycles have elapsed, thereby indicating that the oscillator is running and reasonably stable. |

## 27.3 Timer1 Prescaler

Timer1 has four prescaler options allowing 1, 2, 4 or 8 divisions of the clock input. The T1CKPS bits of the T1CON register control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMR1H or TMR1L.

## 27.4 Timer1 Operation in Asynchronous Mode

If the control bit $\overline{\text{T1SYNC}}$ of the T1CON register is set, the external clock input is not synchronized. The timer increments asynchronously to the internal phase clocks. If the external clock source is selected then the timer will continue to run during Sleep and can generate an interrupt on overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer (see **Section 27.4.1 "Reading and Writing Timer1 in Asynchronous Mode"**).

| | |
|---|---|
| **Note:** | When switching from synchronous to asynchronous operation, it is possible to skip an increment. When switching from asynchronous to synchronous operation, it is possible to produce an additional increment. |

### 27.4.1 READING AND WRITING TIMER1 IN ASYNCHRONOUS MODE

Reading TMR1H or TMR1L while the timer is running from an external asynchronous clock will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself, poses certain problems, since the timer may overflow between the reads.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the TMR1H:TMR1L register pair.

## 27.5 Timer1 Gate

Timer1 can be configured to count freely or the count can be enabled and disabled using Timer1 gate circuitry. This is also referred to as Timer1 Gate Enable.

Timer1 gate can also be driven by multiple selectable sources.

### 27.5.1 TIMER1 GATE ENABLE

The Timer1 Gate Enable mode is enabled by setting the TMR1GE bit of the T1GCON register. The polarity of the Timer1 Gate Enable mode is configured using the T1GPOL bit of the T1GCON register.

When Timer1 Gate Enable mode is enabled, Timer1 will increment on the rising edge of the Timer1 clock source. When Timer1 Gate Enable mode is disabled, no incrementing will occur and Timer1 will hold the current count. See Figure 27-3 for timing details.

**TABLE 27-3: TIMER1 GATE ENABLE SELECTIONS**

| T1CLK | T1GPOL | T1G | Timer1 Operation |
|---|---|---|---|
| ↑ | 0 | 0 | Counts |
| ↑ | 0 | 1 | Holds Count |
| ↑ | 1 | 0 | Holds Count |
| ↑ | 1 | 1 | Counts |

**TABLE 27-5: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER1/3/5**

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---|---|---|---|---|---|---|---|---|---|
| TRISA | — | — | TRISA5 | TRISA4 | —(2) | TRISA2 | TRISA1 | TRISA0 | 140 |
| ANSELA | — | — | ANSA5 | ANSA4 | — | ANSA2 | ANSA1 | ANSA0 | 141 |
| TRISB(1) | TRISB7 | TRISB6 | TRISB5 | TRISB4 | — | — | — | — | 146 |
| ANSELB(1) | ANSB7 | ANSB6 | ANSB5 | ANSB4 | — | — | — | — | 147 |
| TRISC | TRISC7(1) | TRISC6(1) | TRISC5 | TRISC4 | TRISC3 | TRISC2 | TRISC1 | TRISC0 | 153 |
| ANSELC | ANSC7(1) | ANSC6(1) | ANSC5 | ANSC4 | ANSC3 | ANSC2 | ANSC1 | ANSC0 | 154 |
| INTCON | GIE | PEIE | — | — | — | — | — | INTEDG | 98 |
| PIR1 | TMR1GIF | ADIF | RCIF | TXIF | SSP1IF | BCL1IF | TMR2IF | TMR1IF | 105 |
| PIE1 | TMR1GIE | ADIE | RCIE | TXIE | SSP1IE | BCL1IE | TMR2IE | TMR1IE | 100 |
| PIR3 | OSFIF | CSWIF | TMR3GIF | TMR3IF | CLC4IF | CLC3IF | CLC2IF | CLC1IF | 107 |
| PIE3 | OSFIE | CSWIE | TMR3GIE | TMR3IE | CLC4IE | CLC3IE | CLC2IE | CLC1IE | 102 |
| PIR4 | CWG2IF | CWG1IF | TMR5GIF | TMR5IF | CCP4IF | CCP3IF | CCP2IF | CCP1IF | 108 |
| PIE4 | CWG2IE | CWG1IE | TMR5GIE | TMR5IE | CCP4IE | CCP3IE | CCP2IE | CCP1IE | 103 |
| T1CON | TMR1CS<1:0> | | T1CKPS<1:0> | | T1SOSC | $\overline{\text{T1SYNC}}$ | — | TMR1ON | 290 |
| T1GCON | TMR1GE | T1GPOL | T1GTM | T1GSPM | T1GGO/$\overline{\text{DONE}}$ | T1GVAL | T1GSS<1:0> | | 291 |
| TMR1L | TMR1L<7:0> | | | | | | | | 292 |
| TMR1H | TMR1H<7:0> | | | | | | | | 292 |
| T1CKIPPS | — | — | — | T1CKIPPS<4:0> | | | | | 159 |
| T1GPPS | — | — | — | T1GPPS<4:0> | | | | | 159 |
| T3CON | TMR3CS<1:0> | | T3CKPS<1:0> | | T3SOSC | $\overline{\text{T3SYNC}}$ | — | TMR3ON | 290 |
| T3GCON | TMR3GE | T3GPOL | T3GTM | T3GSPM | T3GGO/$\overline{\text{DONE}}$ | T3GVAL | T3GSS<1:0> | | 291 |
| TMR3L | TMR3L<7:0> | | | | | | | | 292 |
| TMR3H | TMR3H<7:0> | | | | | | | | 292 |
| T3CKIPPS | — | — | — | T3CKIPPS<4:0> | | | | | 159 |
| T3GPPS | — | — | — | T3GPPS<4:0> | | | | | 159 |
| T5CON | TMR5CS<1:0> | | T5CKPS<1:0> | | T5SOSC | $\overline{\text{T5SYNC}}$ | — | TMR5ON | 290 |
| T5GCON | TMR5GE | T5GPOL | T5GTM | T5GSPM | T5GGO/$\overline{\text{DONE}}$ | T5GVAL | T5GSS<1:0> | | 291 |
| TMR5L | TMR5L<7:0> | | | | | | | | 292 |
| TMR5H | TMR5H<7:0> | | | | | | | | 292 |
| T5CKIPPS | — | — | — | T5CKIPPS<4:0> | | | | | 159 |
| T5GPPS | — | — | — | T5GPPS<4:0> | | | | | 159 |
| T0CON0 | T0EN | — | T0OUT | T016BIT | T0OUTPS<3:0> | | | | 278 |
| CMxCON0 | CxON | CxOUT | — | CxPOL | — | CxSP | CxHYS | CxSYNC | 187 |
| CCPTMRS | C4TSEL<1:0> | | C3TSEL<1:0> | | C2TSEL<1:0> | | C1TSEL<1:0> | | 309 |
| CCPxCON | CCPxEN | — | CCPxOUT | CCPxFMT | CCPxMODE<3:0> | | | | 306 |
| CLCxSELy | — | — | LCxDyS<5:0> | | | | | | 226 |
| ADACT | — | — | — | ADACT<4:0> | | | | | 243 |

**Legend:** — = unimplemented location, read as '0'. Shaded cells are not used by the Timer1 module.

**Note 1:** PIC16(L)F18344 only.

**2:** Unimplemented, read as '1'.

## 29.0 CAPTURE/COMPARE/PWM MODULES

The Capture/Compare/PWM module is a peripheral that allows the user to time and control different events and to generate Pulse-Width Modulation (PWM) signals. In Capture mode, the peripheral allows the timing of the duration of an event. The Compare mode allows the user to trigger an external event when a predetermined amount of time has expired. The PWM mode can generate Pulse-Width Modulated signals of varying frequency and duty cycle.

This family of devices contains four standard Capture/Compare/PWM modules (CCP1, CCP2, CCP3 and CCP4).

The Capture and Compare functions are identical for all CCP modules.

## 29.1 CCP/PWM Clock Selection

The PIC16(L)F18324/18344 devices allow each individual CCP and PWM module to select the timer source that controls the module. Each module has an independent selection.

As there are up to three 8-bit timers with auto-reload (Timer2, Timer4, and Timer6), PWM mode on the CCP and PWM modules can use any of these timers.

The CCPTMRS register is used to select which timer is used.

| | |
|---|---|
| **Note 1:** | In devices with more than one CCP module, it is very important to pay close attention to the register names used. A number placed after the module acronym is used to distinguish between separate modules. For example, the CCP1CON and CCP2CON control the same operational aspects of two completely different CCP modules. |
| **2:** | Throughout this section, generic references to a CCP module in any of its operating modes may be interpreted as being equally applicable to CCPx module. Register names, module signals, I/O pins, and bit names may use the generic designator ‘x’ to indicate the use of a numeral to distinguish a particular module, when required. |

## 29.2 Capture Mode

Capture mode makes use of either the 16-bit Timer0 or the 16-bit Timer1/3/5 resource. When an event occurs on the capture source, the 16-bit CCPRxH:CCPRxL register pair captures and stores the 16-bit value of the TMR1H:TMR1L register pair, respectively. An event is defined as one of the following and is configured by the CCPxMODE<3:0> bits of the CCPxCON register:

- Every falling edge
- Every rising edge
- Every 4th rising edge
- Every 16th rising edge

When a capture is made, the Interrupt Request Flag bit CCPxIF of the PIR4 register is set. The interrupt flag must be cleared in software. If another capture occurs before the value in the CCPRxH, CCPRxL register pair is read, the old captured value is overwritten by the new captured value.

Figure 29-1 shows a simplified diagram of the capture operation.

### 29.2.1 CAPTURE SOURCES

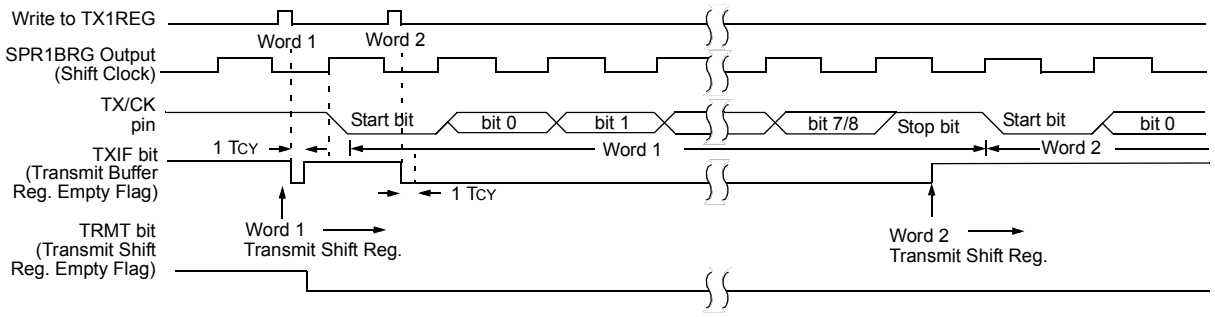
In Capture mode, the CCPx pin should be configured as an input by setting the associated TRIS control bit.

| | |
|---|---|
| **Note:** | If the CCPx pin is configured as an output, a write to the port can cause a Capture condition. |

The capture source is selected by configuring the CCPxCTS<3:0> bits of the CCPxCAP register. The following sources can be selected:

- CCPxPPS input
- C1_output
- C2_output
- NCO_output
- IOC_interrupt
- LC1_output
- LC2_output
- LC3_output
- LC4_output

**FIGURE 31-4: ASYNCHRONOUS TRANSMISSION (BACK-TO-BACK)**




**Note:** This timing diagram shows two consecutive transmissions.

### 31.1.2 EUSART1 ASYNCHRONOUS RECEIVER

The Asynchronous mode is typically used in RS-232 systems. The receiver block diagram is shown in Figure 31-2. The data is received on the RX/DT pin and drives the data recovery block. The data recovery block is actually a high-speed shifter operating at 16 times the baud rate, whereas the serial Receive Shift Register (RSR) operates at the bit rate. When all eight or nine bits of the character have been shifted in, they are immediately transferred to a two character First-In-First-Out (FIFO) memory. The FIFO buffering allows reception of two complete characters and the start of a third character before software must start servicing the EUSART1 receiver. The FIFO and RSR registers are not directly accessible by software. Access to the received data is via the RC1REG register.

#### 31.1.2.1 Enabling the Receiver

The EUSART1 receiver is enabled for asynchronous operation by configuring the following three control bits:

- CREN = 1
- SYNC = 0
- SPEN = 1

All other EUSART1 control bits are assumed to be in their default state.

Setting the CREN bit of the RC1STA register enables the receiver circuitry of the EUSART1. Clearing the SYNC bit of the TX1STA register configures the EUSART1 for asynchronous operation. Setting the SPEN bit of the RC1STA register enables the EUSART1. The programmer must set the corresponding TRIS bit to configure the RX/DT I/O pin as an input.

| **Note:** | If the RX/DT function is on an analog pin, the corresponding ANSEL bit must be cleared for the receiver to function. |
|---|---|

#### 31.1.2.2 Receiving Data

The receiver data recovery circuit initiates character reception on the falling edge of the first bit. The first bit, also known as the Start bit, is always a zero. The data recovery circuit counts one-half bit time to the center of the Start bit and verifies that the bit is still a zero. If it is not a zero then the data recovery circuit aborts character reception, without generating an error, and resumes looking for the falling edge of the Start bit. If the Start bit zero verification succeeds then the data recovery circuit counts a full bit time to the center of the next bit. The bit is then sampled by a majority detect circuit and the resulting '0' or '1' is shifted into the RSR. This repeats until all data bits have been sampled and shifted into the RSR. One final bit time is measured and the level sampled. This is the Stop bit, which is always a '1'. If the data recovery circuit samples a '0' in the Stop bit position then a framing error is set for this character, otherwise the framing error is cleared for this character. See **Section 31.1.2.4 "Receive Framing Error"** for more information on framing errors.

Immediately after all data bits and the Stop bit have been received, the character in the RSR is transferred to the EUSART1 receive FIFO and the RCIF interrupt flag bit of the PIR1 register is set. The top character in the FIFO is transferred out of the FIFO by reading the RC1REG register.

| **Note:** | If the receive FIFO is overrun, no additional characters will be received until the Overrun condition is cleared. See **Section 31.1.2.5 "Receive Overrun Error"** for more information on overrun errors. |
|---|---|

**TABLE 35-7: EXTERNAL CLOCK/OSCILLATOR TIMING SPECIFICATIONS**

| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Param. No.** | **Sym.** | **Characteristic** | **Min.** | **Typ.†** | **Max.** | **Units** | **Conditions** |
| **ECL Oscillator** | | | | | | | |
| OS1 | FECL | Clock Frequency | — | — | 500 | kHz | |
| OS2 | TECL_DC | Clock Duty Cycle | 40 | — | 60 | % | |
| **ECM Oscillator** | | | | | | | |
| OS3 | FECM | Clock Frequency | — | — | 4 | MHz | **Note 4** |
| OS4 | TECM_DC | Clock Duty Cycle | 40 | — | 60 | % | |
| **ECH Oscillator** | | | | | | | |
| OS5 | FECH | Clock Frequency | — | — | 32 | MHz | |
| OS6 | TECH_DC | Clock Duty Cycle | 40 | — | 60 | % | |
| **LP Oscillator** | | | | | | | |
| OS7 | FLP | Clock Frequency | — | — | 100 | kHz | **Note 4** |
| **XT Oscillator** | | | | | | | |
| OS8 | FXT | Clock Frequency | — | — | 4 | MHz | **Note 4** |
| **HS Oscillator** | | | | | | | |
| OS9 | FHS | Clock Frequency | — | — | 20 | MHz | **Note 4** |
| **System Clock** | | | | | | | |
| OS20 | FOSC | System Clock Frequency | — | — | 32 | MHz | **Note 2, Note 3** |
| OS21 | FCY | Instruction Frequency | — | FOSC/4 | — | MHz | |
| OS22 | TCY | Instruction Period | 125 | 1/FCY | — | ns | |

* 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 1:** Instruction cycle period (TCY) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min" values with an external clock applied to OSC1 pin. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

**2:** The system clock frequency (FOSC) is selected by the "main clock switch controls" as described in **Section 7.3 "Clock Switching"**.

**3:** The system clock frequency (FOSC) must meet the voltage requirements defined in the **Section 35.2 "Standard Operating Conditions"**. LP, XT and HS oscillator modes require an appropriate crystal or resonator to be connected to the device.

**4:** For clocking the device with an external square wave, one of the EC mode selections must be used.

## THE MICROCHIP WEBSITE

Microchip provides online support via our website at www.microchip.com. This website is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the website contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

## CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip website at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

## CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

**Technical support is available through the website at: http://www.microchip.com/support**

## PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

| PART NO. | [X](1) | - | X | /XX | XXX |
|---|---|---|---|---|---|
| Device | Tape and Reel Option | | Temperature Range | Package | Pattern |

| | |
|---|---|
| **Device:** | PIC16F18324, PIC16LF18324, PIC16F18344, PIC16LF18344. |
| **Tape and Reel Option:** | Blank = Standard packaging (tube or tray)<br>T = Tape and Reel(1) |
| **Temperature Range:** | I = -40°C to +85°C (Industrial)<br>E = -40°C to +125°C (Extended) |
| **Package:(2)** | JQ = 16-Lead UQFN (4x4)<br>GZ = 20-Lead UQFN (4x4)<br>P = PDIP<br>ST = TSSOP<br>SL = SOIC<br>SO = SOIC<br>SS = SSOP |
| **Pattern:** | QTP, SQTP, Code or Special Requirements (blank otherwise) |

**Examples:**

a) PIC16LF18324- I/P
Industrial temperature
PDIP package

b) PIC16LF18344- E/SO
Extended temperature,
SOIC package

**Note 1:** Tape and Reel identifier only appears in the catalog part number description. This identifier is used for ordering purposes and is not printed on the device package. Check with your Microchip Sales Office for package availability with the Tape and Reel option.

**2:** Small form-factor packaging options may be available. Check www.microchip.com/packaging for small-form factor package availability, or contact your local Sales Office.