**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 | 18 |
| Program Memory Size | 28KB (16K x 14) |
| Program Memory Type | FLASH |
| EEPROM Size | 256 x 8 |
| RAM Size | 2K x 8 |
| Voltage - Supply (Vcc/Vdd) | 1.8V ~ 3.6V |
| Data Converters | A/D 17x10b; D/A 1x5b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°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/pic16lf18346t-i-so |

**TABLE 4-4: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-31 (CONTINUED)**

| Address | Name | PIC16(L)F18326 | PIC16(L)F18346 | 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 5** | | | | | | | | | | | | | |
| **CPU CORE REGISTERS; see Table 4-2 for specifics** | | | | | | | | | | | | | |
| 28Ch | ODCONA | | | — | — | ODCA5 | ODCA4 | — | ODCA2 | ODCA1 | ODCA0 | --00 -000 | --00 -000 |
| 28Dh | ODCONB | X | — | Unimplemented | | | | | | | | — | — |
| | | — | X | ODCB7 | ODCB6 | ODCB5 | ODCB4 | — | — | — | — | 0000 ---- | 0000 ---- |
| 28Eh | ODCONC | X | — | — | — | ODCC5 | ODCC4 | ODCC3 | ODCC2 | ODCC1 | ODCC0 | --00 0000 | --00 0000 |
| | | — | X | ODCC7 | ODCC6 | ODCC5 | ODCC4 | ODCC3 | ODCC2 | ODCC1 | ODCC0 | 0000 0000 | 0000 0000 |
| 28Fh | — | — | | Unimplemented | | | | | | | | — | — |
| 290h | — | — | | Unimplemented | | | | | | | | — | — |
| 291h | CCPR1L | | | CCPR1<7:0> | | | | | | | | xxxx xxxx | xxxx xxxx |
| 292h | CCPR1H | | | CCPR1<15:8> | | | | | | | | xxxx xxxx | xxxx xxxx |
| 293h | CCP1CON | | | CCP1EN | — | CCP1OUT | CCP1FMT | CCP1MODE<3:0> | | | | 0-x0 0000 | 0-x0 0000 |
| 294h | CCP1CAP | | | — | | — | — | CCP1CTS<3:0> | | | | ---- 0000 | ---- xxxx |
| 295h | CCPR2L | | | CCPR2<7:0> | | | | | | | | xxxx xxxx | xxxx xxxx |
| 296h | CCPR2H | | | CCPR2<15:8> | | | | | | | | xxxx xxxx | xxxx xxxx |
| 297h | CCP2CON | | | CCP2EN | — | CCP2OUT | CCP2FMT | CCP2MODE<3:0> | | | | 0-x0 0000 | 0-x0 0000 |
| 298h | CCP2CAP | | | — | — | — | — | CCP2CTS<3:0> | | | | ---- 0000 | ---- xxxx |
| 299h | — | — | | Unimplemented | | | | | | | | — | — |
| 29Ah | — | — | | Unimplemented | | | | | | | | — | — |
| 29Bh | — | — | | Unimplemented | | | | | | | | — | — |
| 29Ch | — | — | | Unimplemented | | | | | | | | — | — |
| 29Dh | — | — | | Unimplemented | | | | | | | | — | — |
| 29Eh | — | — | | Unimplemented | | | | | | | | — | — |
| 29Fh | CCPTMRS | | | C4TSEL<1:0> | | C3TSEL<1:0> | | C2TSEL<1:0> | | C1TSEL<1:0> | | 0101 0101 | 0101 0101 |

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

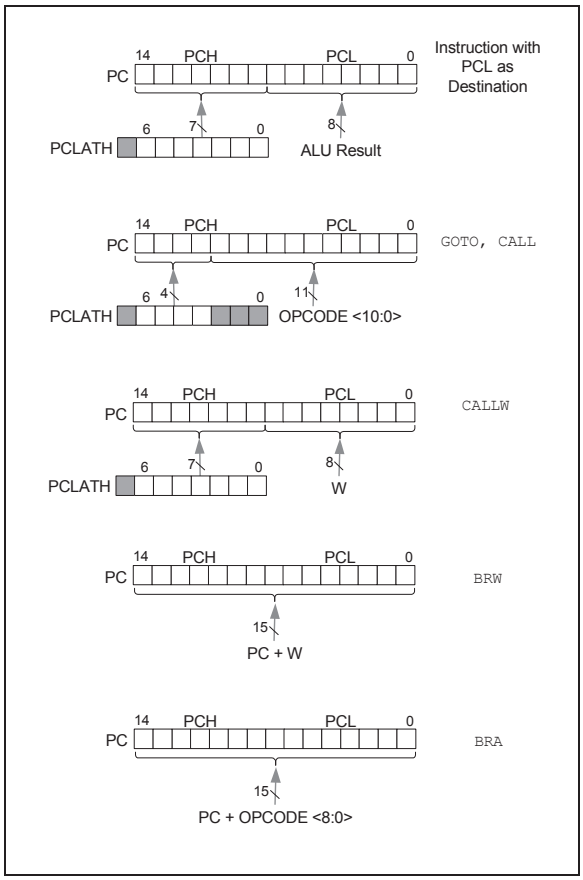
**Note 1:** Only on PIC16F18326/18346.

**2:** Register accessible from both User and ICD Debugger.

## 4.3 PCL and PCLATH

The Program Counter (PC) is 15 bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<14:8>) is not directly readable or writable and comes from PCLATH. On any Reset, the PC is cleared. Figure 4-3 shows the five situations for the loading of the PC.

**FIGURE 4-3: LOADING OF PC IN DIFFERENT SITUATIONS**




### 4.3.1 MODIFYING PCL

Executing any instruction with the PCL register as the destination simultaneously causes the Program Counter PC<14:8> bits (PCH) to be replaced by the contents of the PCLATH register. This allows the entire contents of the program counter to be changed by writing the desired upper seven bits to the PCLATH register. When the lower eight bits are written to the PCL register, all 15 bits of the program counter will change to the values contained in the PCLATH register and those being written to the PCL register.

### 4.3.2 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL). When performing a table read using a computed GOTO method, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block). Refer to Application Note AN556, *Implementing a Table Read* (DS00556).

### 4.3.3 COMPUTED FUNCTION CALLS

A computed function CALL allows programs to maintain tables of functions and provide another way to execute state machines or look-up tables. When performing a table read using a computed function CALL, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block).

If using the CALL instruction, the PCH<2:0> and PCL registers are loaded with the operand of the CALL instruction. PCH<6:3> is loaded with PCLATH<6:3>.

The CALLW instruction enables computed calls by combining PCLATH and $\overline{W}$ to form the destination address. A computed CALLW is accomplished by loading the $\overline{W}$ register with the desired address and executing CALLW. The PCL register is loaded with the value of $\overline{W}$ and PCH is loaded with PCLATH.

### 4.3.4 BRANCHING

The branching instructions add an offset to the PC. This allows relocatable code and code that crosses page boundaries. There are two forms of branching, BRW and BRA. The PC will have incremented to fetch the next instruction in both cases. When using either branching instruction, a PCL memory boundary may be crossed.

If using BRW, load the $\overline{W}$ register with the desired unsigned address and execute BRW. The entire PC will be loaded with the address PC + 1 + $\overline{W}$.

If using BRA, the entire PC will be loaded with PC + 1 + the signed value of the operand of the BRA instruction.

## 5.7 Register Definitions: Device and Revision

**REGISTER 5-5: DEVID: DEVICE ID REGISTER**

| R | R | R | R | R | R |
|---|---|---|---|---|---|
| DEV<13:8> | | | | | |
| bit 13 | | | | | bit 8 |

| R | R | R | R | R | R | R | R |
|---|---|---|---|---|---|---|---|
| DEV<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | |
|---|---|
| R = Readable bit | |
| '1' = Bit is set | '0' = Bit is cleared |

bit 13-0 **DEV<13:0>:** Device ID bits

| Device | DEVID<13:0> Values |
|---|---|
| PIC16F18326 | 11 0000 1010 0100 (30A4) |
| PIC16LF18326 | 11 0000 1010 0110 (30A6) |
| PIC16F18346 | 11 0000 1010 0101 (30A5) |
| PIC16LF18346 | 11 0000 1010 0111 (30A7) |

**REGISTER 5-6: REVID: REVISION ID REGISTER**

| R-1 | R-0 | R | R | R | R |
|---|---|---|---|---|---|
| REV<13:8> | | | | | |
| bit 13 | | | | | bit 8 |

| R | R | R | R | R | R | R | R |
|---|---|---|---|---|---|---|---|
| REV<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 0 |

| **Legend:** | |
|---|---|
| R = Readable bit | |
| '1' = Bit is set | '0' = Bit is cleared |

bit 13-0 **REV<13:0>:** Revision ID bits

**Note:** The upper two bits of the Revision ID Register will always read '10'.

**REGISTER 12-5: WPUA: WEAK PULL-UP PORTA REGISTER**

| U-0 | U-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 |
|---|---|---|---|---|---|---|---|
| — | — | WPUA5 | WPUA4 | WPUA3(1) | WPUA2 | WPUA1 | WPUA0 |
| 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 5-0 **WPUA<5:0>**: Weak Pull-up Register bits(2)
1 = Pull-up enabled
0 = Pull-up disabled

**Note 1:** If MCLRE = 1, the weak pull-up in RA3 is always enabled; bit WPUA3 is not affected.
**2:** The weak pull-up device is automatically disabled if the pin is configured as an output.

**REGISTER 12-6: ODCONA: PORTA OPEN-DRAIN CONTROL REGISTER**

| 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 |
|---|---|---|---|---|---|---|---|
| — | — | ODCA5 | ODCA4 | — | ODCA2 | ODCA1 | ODCA0 |
| 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 5-4 **ODCA<5:4>:** PORTA Open-Drain Enable bits
For RA<5:4> pins, respectively
1 = Port pin operates as open-drain drive (sink current only)
0 = Port pin operates as standard push-pull drive (source and sink current)

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

bit 2-0 **ODCA<2:0>:** PORTA Open-Drain Enable bits
For RA<2:0> pins, respectively
1 = Port pin operates as open-drain drive (sink current only)
0 = Port pin operates as standard push-pull drive (source and sink current)

## 13.3 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. This requires configuring both the appropriate xxxPPS input and RxyPPS output registers. For example, if the SCL1 line is routed to pin RC0, the SSP1SCLPPS input register would be set to '10000' (routes to RC0) and the RC0PPS output register would be set to '11000' (routes the SCL1 internal connection to RC0). Peripherals that have bidirectional signals are:

- EUSART1 (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 PIC16(L)F18326 with this compatibility. For the PIC16(L)F18346, in addition to the default pins as described above, RC0, RC1, RC4, and RC5 are also $I^2C$ and SMBus compatible. Clock and data signals can be routed to any pin, however pins without $I^2C$ compatibility will operate at standard TTL/ST logic levels as selected by the INVLV register. |

## 13.4 PPSLOCKED Bit

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 13-1.

**EXAMPLE 13-1: 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
```

## 13.5 PPS1WAY Bit

The PPS can be locked by setting the PPS1WAY bit of Configuration Word 2.

When the PPS1WAY bit is set, the PPSLOCKED bit of the PPSLOCK register can be cleared and set only one time after a device Reset. Once the PPS registers are configured, user software sets the PPSLOCKED bit, preventing any further writes to the PPS registers. the PPS registers can be read at any time, regardless of the PPS1WAY or PPSLOCKED settings.

When the PPS1WAY bit is clear, the PPSLOCKED bit of the PPSLOCK register can be cleared and set multiple times during code execution, but requires the PPS lock/unlock sequence to be performed each time modifications to the PPS registers are made.

## 13.6 Operation During Sleep

PPS input and output selections are unaffected by Sleep.

## 13.7 Effects of a Reset

A device Power-On-Reset (POR) clears all PPS input and output selections to their default values, and clears the PPSLOCKED bit of the PPSLOCK register. All other Resets leave the selections unchanged. Default input selections are shown in pin allocation Table 2 and Table 3.

**REGISTER 18-2: CMxCON1: COMPARATOR Cx CONTROL REGISTER 1**

| R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| CxINTP | CxINTN | CxPCH<2:0> | | | CxNCH<2:0> | | |
| 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 **CxINTP:** Comparator Interrupt on Positive Going Edge Enable bits
1 = The CxIF interrupt flag will be set upon a positive going edge of the CxOUT bit
0 = No interrupt flag will be set on a positive going edge of the CxOUT bit

bit 6 **CxINTN:** Comparator Interrupt on Negative Going Edge Enable bits
1 = The CxIF interrupt flag will be set upon a negative going edge of the CxOUT bit
0 = No interrupt flag will be set on a negative going edge of the CxOUT bit

bit 5-3 **CxPCH<2:0>:** Comparator Positive Input Channel Select bits
111 = CxVP connects to Vss
110 = CxVP connects to FVR Buffer 2
101 = CxVP connects to DAC output
100 = CxVP unconnected
011 = CxVP unconnected
010 = CxVP unconnected
001 = CxVN unconnected
000 = CxVP connects to CxIN0+ pin

bit 2-0 **CxNCH<2:0>:** Comparator Negative Input Channel Select bits
111 = CxVN connects to Vss
110 = CxVN connects to FVR Buffer 2
101 = CxVN unconnected
100 = CxVN unconnected
011 = CxVN connects to CxIN3- pin
010 = CxVN connects to CxIN2- pin
001 = CxVN connects to CxIN1- pin
000 = CxVN connects to CxIN0- pin

**REGISTER 20-4: CWGxDAT: CWGx DATA INPUT SELECTION REGISTER**

| U-0 | U-0 | U-0 | U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | DAT<3:0> | | | |
| bit 7 | | | | | | | bit 0 |

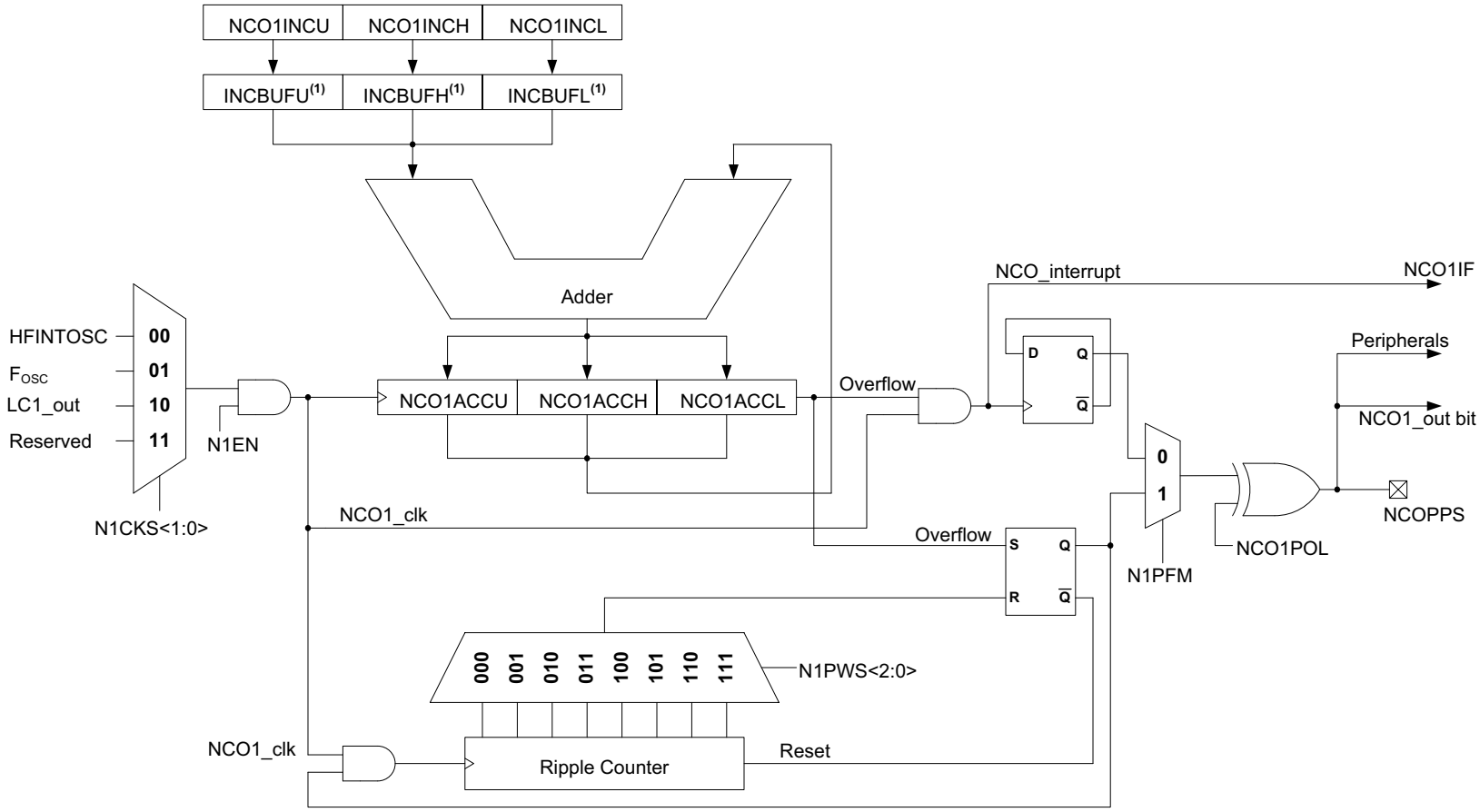
| **Legend:** | | |
|---|---|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| u = Bit is unchanged | x = Bit is unknown | -n/n = Value at POR and BOR/Value at all other Resets |
| '1' = Bit is set | '0' = Bit is cleared | q = Value depends on condition |

bit 7-4 **Unimplemented:** Read as '0'

bit 3-0 **DAT<3:0>:** CWG Data Input Selection bits

| DAT | Data Source |
|---|---|
| 0000 | CWGxPPS |
| 0001 | C1OUT |
| 0010 | C2OUT |
| 0011 | CCP1 |
| 0100 | CCP2 |
| 0101 | CCP3 |
| 0110 | CCP4 |
| 0111 | PWM5 |
| 1000 | PWM6 |
| 1001 | NCO1 |
| 1010 | CLC1 |
| 1011 | CLC2 |
| 1100 | CLC3 |
| 1101 | CLC4 |
| 1110 | Reserved |
| 1111 | Reserved |

**FIGURE 23-1: NUMERICALLY CONTROLLED OSCILLATOR MODULE SIMPLIFIED BLOCK DIAGRAM**




**Note 1:** The increment registers are double-buffered to allow for value changes to be made without first disabling the NCO1 module. They are shown for reference only and are not user accessible.

## 25.0 DATA SIGNAL MODULATOR (DSM) MODULE

The Data Signal Modulator (DSM) is a peripheral which allows the user to mix a data stream, also known as a modulator signal, with a carrier signal to produce a modulated output.

Both the carrier and the modulator signals are supplied to the DSM module either internally from the output of a peripheral, or externally through an input pin.

The modulated output signal is generated by performing a logical “AND” operation of both the carrier and modulator signals and then provided to the MDOUT pin.

The carrier signal is comprised of two distinct and separate signals. A carrier high (CARH) signal and a carrier low (CARL) signal. During the time in which the modulator (MOD) signal is in a logic high state, the DSM mixes the carrier high signal with the modulator signal. When the modulator signal is in a logic low state, the DSM mixes the carrier low signal with the modulator signal.

Using this method, the DSM can generate the following types of key modulation schemes:

- Frequency-Shift Keying (FSK)
- Phase-Shift Keying (PSK)
- On-Off Keying (OOK)

Additionally, the following features are provided within the DSM module:

- Carrier Synchronization
- Carrier Source Polarity Select
- Carrier Source Pin Disable
- Programmable Modulator Data
- Modulator Source Pin Disable
- Modulated Output Polarity Select
- Slew Rate Control

Figure 25-1 shows a simplified block diagram of the Data Signal Modulator peripheral.

**REGISTER 27-2: TxGCON(1): TIMERx GATE CONTROL REGISTER**

| R/W-0/u | R/W-0/u | R/W-0/u | R/W-0/u | R/W/HC-0/u | R-x/x | R/W-0/u | R/W-0/u |
|---|---|---|---|---|---|---|---|
| TMRxGE | TxGPOL | TxGTM | TxGSPM | TxGGO/$\overline{\text{DONE}}$ | TxGVAL | TxGSS<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 | HC = Bit is cleared by hardware |

bit 7 **TMRxGE:** Timer1 Gate Enable bit

If TMRxON = 0:
This bit is ignored
If TMRxON = 1:
1 = Timerx counting is controlled by the Timer1 gate function
0 = Timerx is always counting

bit 6 **TxGPOL:** Timerx Gate Polarity bit

1 = Timerx gate is active-high (Timerx counts when gate is high)
0 = Timerx gate is active-low (Timerx counts when gate is low)

bit 5 **TxGTM:** Timerx Gate Toggle Mode bit

1 = Timerx Gate Toggle mode is enabled
0 = Timerx Gate Toggle mode is disabled and toggle flip-flop is cleared
Timerx gate flip-flop toggles on every rising edge.

bit 4 **TxGSPM:** Timerx Gate Single-Pulse Mode bit

1 = Timerx Gate Single-Pulse mode is enabled and is controlling Timerx gate
0 = Timerx Gate Single-Pulse mode is disabled

bit 3 **TxGGO/$\overline{\text{DONE}}$:** Timerx Gate Single-Pulse Acquisition Status bit

1 = Timerx gate single-pulse acquisition is ready, waiting for an edge
0 = Timerx gate single-pulse acquisition has completed or has not been started
This bit is automatically cleared when TxGSPM is cleared

bit 2 **TxGVAL:** Timerx Gate Value Status bit

Indicates the current state of the Timerx gate, latched at Q1, provided to TMRxH:TMRxL
Unaffected by Timerx Gate Enable (TMRxGE)

bit 1-0 **TxGSS<1:0>:** Timerx Gate Source Select bits

11 = Comparator 2 optionally synchronized output
10 = Comparator 1 optionally synchronized output
01 = Timer0 overflow output
00 = Timerx gate pin

**Note 1:** 'x' refers to either '1', '3' or '5' for the respective Timer1/3/5 registers.

### 30.6.8 ACKNOWLEDGE SEQUENCE TIMING

An Acknowledge sequence is enabled by setting the Acknowledge Sequence Enable bit, ACKEN bit of the SSPxCON2 register. When this bit is set, the SCL pin is pulled low and the contents of the Acknowledge data bit are presented on the SDA pin. If the user wishes to generate an Acknowledge, then the ACKDT bit should be cleared. If not, the user should set the ACKDT bit before starting an Acknowledge sequence. The Baud Rate Generator then counts for one rollover period (TBRG) and the SCL pin is deasserted (pulled high). When the SCL pin is sampled high (clock arbitration), the Baud Rate Generator counts for TBRG. The SCL pin is then pulled low. Following this, the ACKEN bit is automatically cleared, the Baud Rate Generator is turned off and the MSSPx module then goes into Idle mode (Figure 30-30).

#### 30.6.8.1 WCOL Status Flag

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

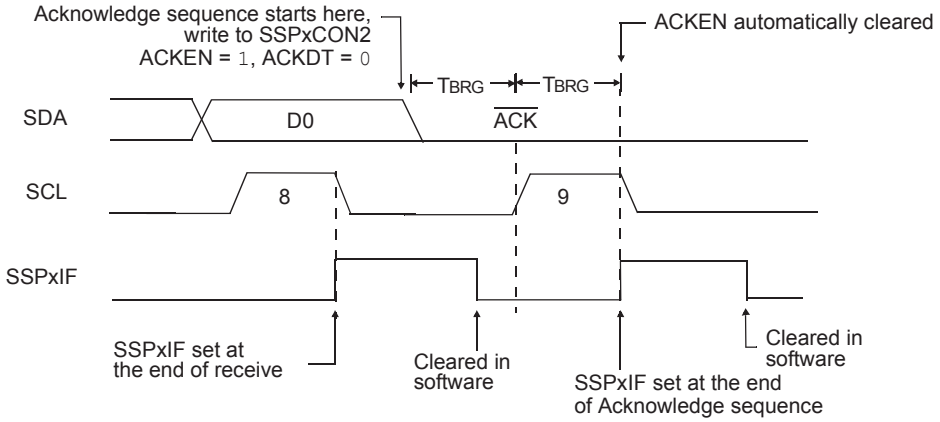
### 30.6.9 STOP CONDITION TIMING

A Stop bit is asserted on the SDA pin at the end of a receive/transmit by setting the Stop Sequence Enable bit, PEN bit of the SSPxCON2 register. At the end of a receive/transmit, the SCL line is held low after the falling edge of the ninth clock. When the PEN bit is set, the master will assert the SDA line low. When the SDA line is sampled low, the Baud Rate Generator is reloaded and counts down to '0'. When the Baud Rate Generator times out, the SCL pin will be brought high and one TBRG (Baud Rate Generator rollover count) later, the SDA pin will be deasserted. When the SDA pin is sampled high while SCL is high, the P bit of the SSPxSTAT register is set. A TBRG later, the PEN bit is cleared and the SSPxIF bit is set (Figure 30-31).

#### 30.6.9.1 WCOL Status Flag

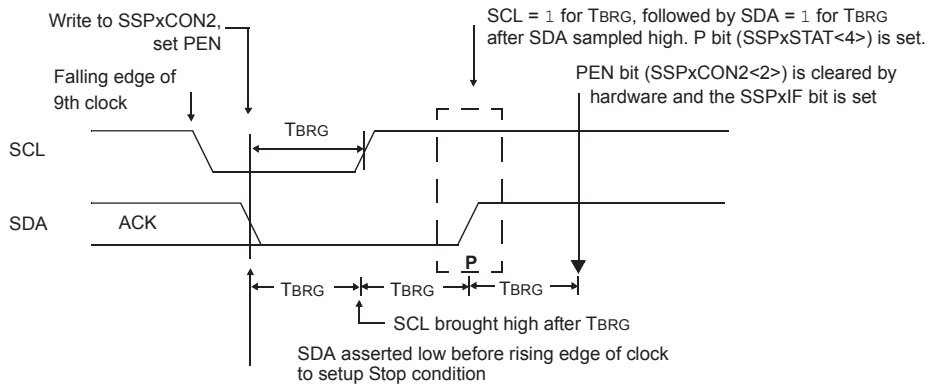
If the user writes the SSPxBUF when a Stop sequence is in progress, then the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

**FIGURE 30-30: ACKNOWLEDGE SEQUENCE WAVEFORM**




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

**FIGURE 30-31: STOP CONDITION RECEIVE OR TRANSMIT MODE**




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

**REGISTER 30-3: SSPxCON2: SSPx CONTROL REGISTER 2 ($I^2C$ MODE ONLY)(1)**

| R/W-0/0 | R/HS/HC-0 | R/W-0/0 | R/S/HC-0/0 | R/S/HC-0/0 | R/S/HC-0/0 | R/S/HC-0/0 | R/S/HC-0/0 |
|---|---|---|---|---|---|---|---|
| GCEN | ACKSTAT | ACKDT | ACKEN | RCEN | PEN | RSEN | SEN |
| 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 | HC = Cleared by hardware S = User set |

bit 7 **GCEN:** General Call Enable bit (in $I^2C$ Slave mode only)
1 = Enable interrupt when a general call address (0x00 or 00h) is received in the SSPSR
0 = General call address disabled

bit 6 **ACKSTAT:** Acknowledge Status bit (in $I^2C$ mode only)
1 = Acknowledge was not received
0 = Acknowledge was received

bit 5 **ACKDT:** Acknowledge Data bit (in $I^2C$ mode only)
In Receive mode:
Value transmitted when the user initiates an Acknowledge sequence at the end of a receive
1 = Not Acknowledge
0 = Acknowledge

bit 4 **ACKEN:** Acknowledge Sequence Enable bit (in $I^2C$ Master mode only)
In Master Receive mode:
1 = Initiate Acknowledge sequence on SDA and SCL pins, and transmit ACKDT data bit. Automatically cleared by hardware.
0 = Acknowledge sequence idle

bit 3 **RCEN:** Receive Enable bit (in $I^2C$ Master mode only)
1 = Enables Receive mode for $I^2C$
0 = Receive idle

bit 2 **PEN:** Stop Condition Enable bit (in $I^2C$ Master mode only)
1 = Initiate Stop condition on SDA and SCL pins. Automatically cleared by hardware.
0 = Stop condition Idle

bit 1 **RSEN:** Repeated Start Condition Enable bit (in $I^2C$ Master mode only)
1 = Initiate Repeated Start condition on SDA and SCL pins. Automatically cleared by hardware.
0 = Repeated Start condition Idle

bit 0 **SEN:** Start Condition Enable/Stretch Enable bit
In Master mode:
1 = Initiate Start condition on SDA and SCL pins. Automatically cleared by hardware.
0 = Start condition Idle
In Slave mode:
1 = Clock stretching is enabled for both slave transmit and slave receive (stretch enabled)
0 = Clock stretching is disabled

**Note 1:** For bits ACKEN, RCEN, PEN, RSEN, SEN: If the $I^2C$ module is not in the idle state, these bits may not be set (no spooling) and the SPPxBUF may not be written.

**REGISTER 30-5: SSPxMSK: 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 |
|---|---|---|---|---|---|---|---|
| SSPxMSK<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 **SSPxMSK<7:1>:** Mask bits
1 = The received address bit n is compared to SPPxADD<n> to detect $I^2C$ address match
0 = The received address bit n is not used to detect $I^2C$ address match

bit 0 **SSPxMSK<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 SPPxADD<0> to detect $I^2C$ address match
0 = The received address bit 0 is not used to detect $I^2C$ address match
$I^2C$ Slave mode, 7-bit address:
MSK0 bit is ignored.

**REGISTER 30-6: SSPxADD: 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 |
|---|---|---|---|---|---|---|---|
| SSPxADD<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 **SSPxADD<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 **SSPxADD<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 **SSPxADD<7:0>:** Eight Least Significant bits of 10-bit address

**7-bit Slave mode:**

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

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

## 31.1 EUSART1 Asynchronous Mode

The EUSART1 transmits and receives data using the standard non-return-to-zero (NRZ) format. NRZ is implemented with two levels: a VOH Mark state which represents a '1' data bit, and a VOL Space state which represents a '0' data bit. NRZ refers to the fact that consecutively transmitted data bits of the same value stay at the output level of that bit without returning to a neutral level between each bit transmission. An NRZ transmission port idles in the Mark state. Each character transmission consists of one Start bit followed by eight or nine data bits and is always terminated by one or more Stop bits. The Start bit is always a space and the Stop bits are always marks. The most common data format is eight bits. Each transmitted bit persists for a period of 1/(Baud Rate). An on-chip dedicated 8-bit/16-bit Baud Rate Generator is used to derive standard baud rate frequencies from the system oscillator. See Table 31-3 for examples of baud rate configurations.

The EUSART1 transmits and receives the LSb first. The EUSART1's transmitter and receiver are functionally independent, but share the same data format and baud rate. Parity is not supported by the hardware, but can be implemented in software and stored as the ninth data bit.

### 31.1.1 EUSART1 ASYNCHRONOUS TRANSMITTER

The EUSART1 transmitter block diagram is shown in Figure 31-1. The heart of the transmitter is the serial Transmit Shift Register (TSR), which is not directly accessible by software. The TSR obtains its data from the transmit buffer, which is the TX1REG register.

#### 31.1.1.1 Enabling the Transmitter

The EUSART1 transmitter is enabled for asynchronous operations by configuring the following three control bits:

- TXEN = 1
- SYNC = 0
- SPEN = 1

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

Setting the TXEN bit of the TX1STA register enables the transmitter 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 and automatically configures the TX/CK I/O pin as an output. If the TX/CK pin is shared with an analog peripheral, the analog I/O function must be disabled by clearing the corresponding ANSEL bit.

| **Note:** | The TXIF Transmitter Interrupt flag is set when the TXEN enable bit is set. |
|---|---|

#### 31.1.1.2 Transmitting Data

A transmission is initiated by writing a character to the TX1REG register. If this is the first character, or the previous character has been completely flushed from the TSR, the data in the TX1REG is immediately transferred to the TSR register. If the TSR still contains all or part of a previous character, the new character data is held in the TX1REG until the Stop bit of the previous character has been transmitted. The pending character in the TX1REG is then transferred to the TSR in one TCY immediately following the Stop bit transmission. The transmission of the Start bit, data bits and Stop bit sequence commences immediately following the transfer of the data to the TSR from the TX1REG.

#### 31.1.1.3 Transmit Data Polarity

The polarity of the transmit data can be controlled with the SCKP bit of the BAUD1CON register. The default state of this bit is '0' which selects high true transmit idle and data bits. Setting the SCKP bit to '1' will invert the transmit data resulting in low true idle and data bits. The SCKP bit controls transmit data polarity in Asynchronous mode only. In Synchronous mode, the SCKP bit has a different function. See **Section 31.4.1.2 "Clock Polarity"**.

#### 31.1.1.4 Transmit Interrupt Flag

The TXIF interrupt flag bit of the PIR1 register is set whenever the EUSART1 transmitter is enabled and no character is being held for transmission in the TX1REG. In other words, the TXIF bit is only clear when the TSR is busy with a character and a new character has been queued for transmission in the TX1REG. The TXIF flag bit is not cleared immediately upon writing TX1REG. TXIF becomes valid in the second instruction cycle following the write execution. Polling TXIF immediately following the TX1REG write will return invalid results. The TXIF bit is read-only, it cannot be set or cleared by software.

The TXIF interrupt can be enabled by setting the TXIE interrupt enable bit of the PIE1 register. However, the TXIF flag bit will be set whenever the TX1REG is empty, regardless of the state of TXIE enable bit.

To use interrupts when transmitting data, set the TXIE bit only when there is more data to send. Clear the TXIE interrupt enable bit upon writing the last character of the transmission to the TX1REG.

#### 31.4.1.6 Slave Clock

Synchronous data transfers use a separate clock line, which is synchronous with the data. A device configured as a slave receives the clock on the TX/CK line. The TX/CK pin output driver is automatically disabled when the device is configured for synchronous slave transmit or receive operation. Serial data bits change on the leading edge to ensure they are valid at the trailing edge of each clock. One data bit is transferred for each clock cycle. Only as many clock cycles should be received as there are data bits.

| **Note:** | If the device is configured as a slave and the TX/CK function is on an analog pin, the corresponding ANSEL bit must be cleared. |
|---|---|

#### 31.4.1.7 Receive Overrun Error

The receive FIFO buffer can hold two characters. An overrun error will be generated if a third character, in its entirety, is received before RC1REG is read to access the FIFO. When this happens the OERR bit of the RC1STA register is set. Previous data in the FIFO will not be overwritten. The two characters in the FIFO buffer can be read, however, no additional characters will be received until the error is cleared. The OERR bit can only be cleared by clearing the Overrun condition. If the overrun error occurred when the SREN bit is set and CREN is clear then the error is cleared by reading RC1REG. If the overrun occurred when the CREN bit is set then the Error condition is cleared by either clearing the CREN bit of the RC1STA register or by clearing the SPEN bit which resets the EUSART1.

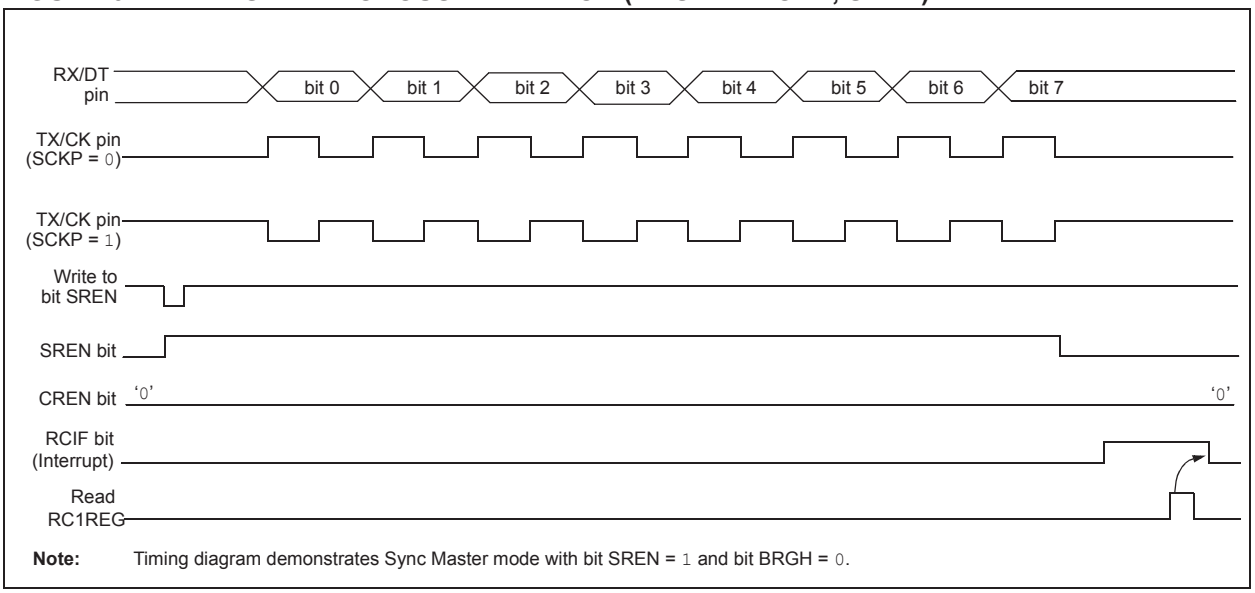
#### 31.4.1.8 Receiving 9-bit Characters

The EUSART1 supports 9-bit character reception. When the RX9 bit of the RC1STA register is set the EUSART1 will shift nine bits into the RSR for each character received. The RX9D bit of the RC1STA register is the ninth, and Most Significant, data bit of the top unread character in the receive FIFO. When reading 9-bit data from the receive FIFO buffer, the RX9D data bit must be read before reading the eight Least Significant bits from the RC1REG.

#### 31.4.1.9 Synchronous Master Reception Setup

1. Initialize the SP1BRGH, SP1BRGL register pair for the appropriate baud rate. Set or clear the BRGH and BRG16 bits, as required, to achieve the desired baud rate.
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the synchronous master serial port by setting bits SYNC, SPEN and CSRC.
4. Ensure bits CREN and SREN are clear.
5. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
6. If 9-bit reception is desired, set bit RX9.
7. Start reception by setting the SREN bit or for continuous reception, set the CREN bit.
8. Interrupt flag bit RCIF will be set when reception of a character is complete. An interrupt will be generated if the enable bit RCIE was set.
9. Read the RC1STA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
10. Read the 8-bit received data by reading the RC1REG register.
11. If an overrun error occurs, clear the error by either clearing the CREN bit of the RC1STA register or by clearing the SPEN bit which resets the EUSART1.

**FIGURE 31-12: SYNCHRONOUS RECEPTION (MASTER MODE, SREN)**




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

**REGISTER 32-1: CLKRCON: REFERENCE CLOCK CONTROL REGISTER**

| R/W-0/0 | U-0 | U-0 | R/W-1/1 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| CLKREN | — | — | CLKRDC<1:0> | | CLKRDIV<2:0> | | |
| 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 **CLKREN:** Reference Clock Module Enable bit
1 = Reference clock module enabled
0 = Reference clock module is disabled

bit 6-5 **Unimplemented:** Read as '0'

bit 4-3 **CLKRDC<1:0>:** Reference Clock Duty Cycle bits (1)
11 = Clock outputs duty cycle of 75%
10 = Clock outputs duty cycle of 50%
01 = Clock outputs duty cycle of 25%
00 = Clock outputs duty cycle of 0%

bit 2-0 **CLKRDIV<2:0>:** Reference Clock Divider bits
111 = Fosc divided by 128
110 = Fosc divided by 64
101 = Fosc divided by 32
100 = Fosc divided by 16
011 = Fosc divided by 8
010 = Fosc divided by 4
001 = Fosc divided by 2
000 = Fosc

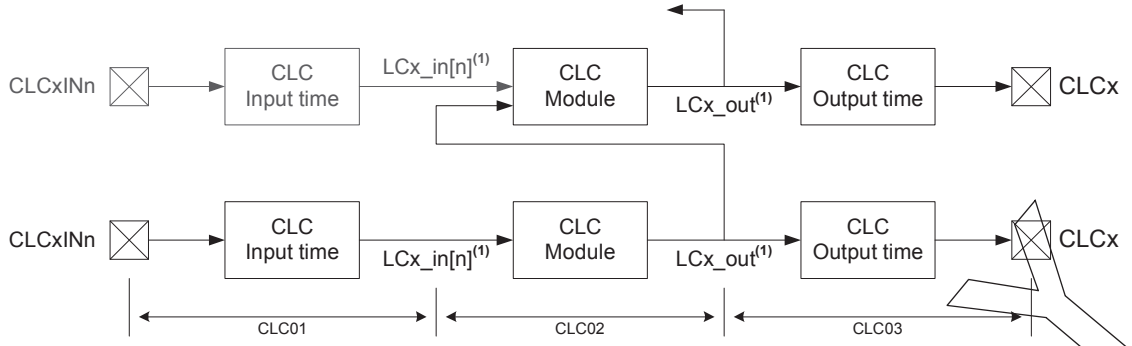
**Note 1:** Bits are valid for Reference Clock divider values of two or larger, the base clock cannot be further divided.

**TABLE 32-1: SUMMARY OF REGISTERS ASSOCIATED WITH CLOCK REFERENCE OUTPUT**

| 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 | 143 |
| TRISB(1) | TRISB7 | TRISB6 | TRISB5 | TRISB4 | — | — | — | — | 149 |
| TRISC | TRISC7(1) | TRISC6(1) | TRISC5 | TRISC4 | TRISC3 | TRISC2 | TRISC1 | TRISC0 | 155 |
| CLKRCON | CLKREN | — | — | CLKRDC<1:0> | | CLKRDIV<2:0> | | | 227 |
| CLCxSELy | — | — | LCxDyS<5:0> | | | | | | 229 |
| MDCARH | — | MDCHPOL | MDCHSYNC | — | MDCH<3:0> | | | | 273 |
| MDCARL | — | MDCLPOL | MDCLSYNC | — | MDCL<3:0> | | | | 274 |

**Legend:** — = unimplemented, read as '0'. Shaded cells are not used by the CLKR module.
**Note 1:** PIC16(L)F18346 only.
**2:** Unimplemented, read as '1'.

**FIGURE 35-14: CLC PROPAGATION TIMING**




**Note 1:** See Figure 21-1, "CLCx Simplified Block Diagram" to identify specific CLC signals.

**TABLE 35-19: CONFIGURABLE LOGIC CELL (CLC) CHARACTERISTICS**

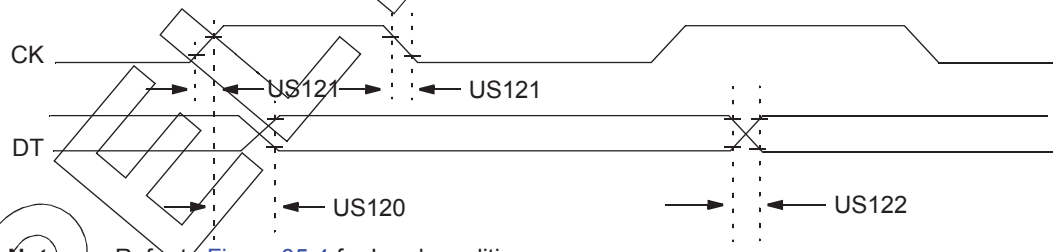
| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Param. No.** | **Sym.** | **Characteristic** | **Min.** | **Typ.†** | **Max.** | **Units** | **Conditions** |
| CLC01* | TCLCIN | CLC input time | — | 7 | OS17 | ns | **(Note 1)** |
| CLC02* | TCLC | CLC module input to output propagation time | — | 24 | — | ns | VDD = 1.8V |
| | | | — | 12 | — | ns | VDD > 3.6V |
| CLC03* | TCLCOUT | CLC output time Rise Time | — | OS18 | — | — | **(Note 1)** |
| | | Fall Time | — | OS19 | — | — | **(Note 1)** |
| CLC04* | FCLCMAX | CLC maximum switching frequency | — | 32 | FOSC | MHz | |

* 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:** See Table 35-10 for OS17, OS18 and OS19 rise and fall times.

**FIGURE 35-15: EUSART SYNCHRONOUS TRANSMISSION (MASTER/SLAVE) TIMING**




**Note:** Refer to Figure 35-4 for Load conditions.

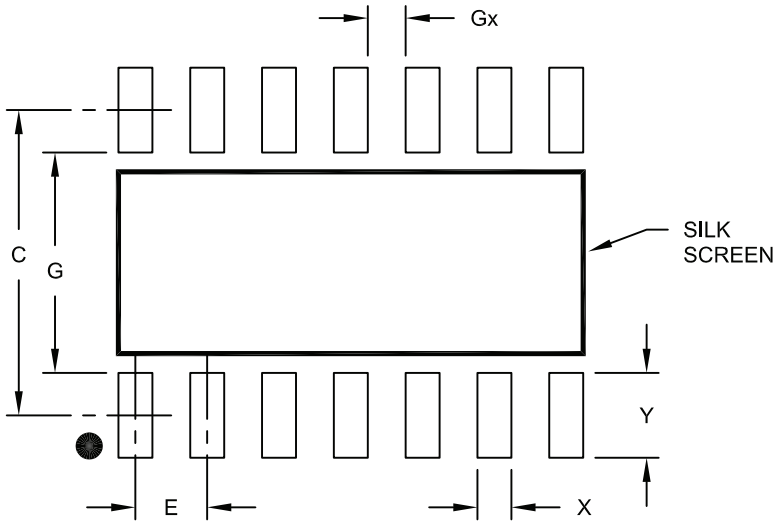
## 36.0 DC AND AC CHARACTERISTICS GRAPHS AND CHARTS

Charts and graphs are not available at this time.

14-Lead Plastic Small Outline (SL) - Narrow, 3.90 mm Body [SOIC]

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




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | 1.27 BSC | | |
| Contact Pad Spacing | C | | 5.40 | |
| Contact Pad Width | X | | | 0.60 |
| Contact Pad Length | Y | | | 1.50 |
| Distance Between Pads | Gx | 0.67 | | |
| Distance Between Pads | G | 3.90 | | |

Notes:

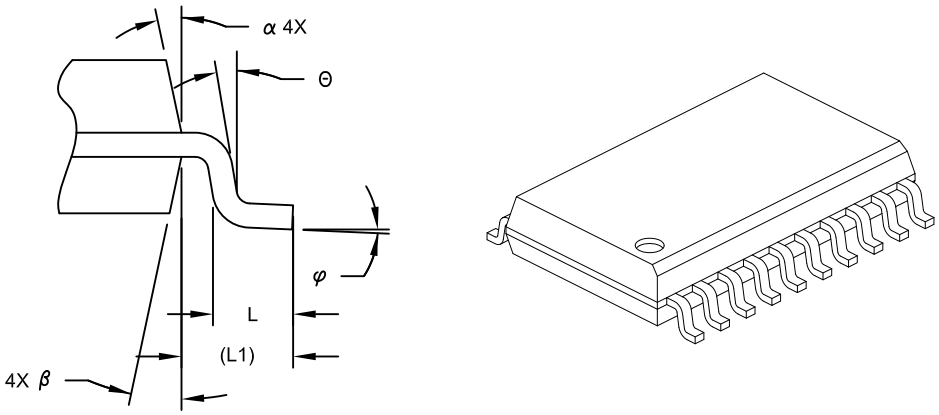
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2065A

### 20-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

**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 | | 20 | |
| Pitch | e | | 1.27 BSC | |
| Overall Height | A | - | - | 2.65 |
| Molded Package Thickness | A2 | 2.05 | - | - |
| Standoff § | A1 | 0.10 | - | 0.30 |
| Overall Width | E | | 10.30 BSC | |
| Molded Package Width | E1 | | 7.50 BSC | |
| Overall Length | D | | 12.80 BSC | |
| Chamfer (Optional) | h | 0.25 | - | 0.75 |
| Foot Length | L | 0.40 | - | 1.27 |
| Footprint | L1 | | 1.40 REF | |
| Lead Angle | Θ | 0° | - | - |
| Foot Angle | φ | 0° | - | 8° |
| Lead Thickness | c | 0.20 | - | 0.33 |
| Lead Width | b | 0.31 | - | 0.51 |
| Mold Draft Angle Top | α | 5° | - | 15° |
| Mold Draft Angle Bottom | β | 5° | - | 15° |

**Notes:**

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic
3. Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
4. 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.
5. Datums A & B to be determined at Datum H.

Microchip Technology Drawing No. C04-094C Sheet 2 of 2