

Welcome to [E-XFL.COM](https://www.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	I <sup>2</sup> C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	18
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1K 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-UFQFN Exposed Pad
Supplier Device Package	20-UQFN (4x4)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18345t-i-gz">https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18345t-i-gz</a>

## 5.3 Code Protection

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

### 5.3.1 PROGRAM MEMORY PROTECTION

The entire program memory space is protected from external reads and writes by the  $\overline{CP}$  bit in Configuration Words. When  $\overline{CP} = 0$ , external reads and writes of program memory are inhibited and a read will return all '0's. The CPU can continue to read program memory, regardless of the protection bit settings. Self-write writing the program memory is dependent upon the write protection setting. See **Section 5.4 “Write Protection”** for more information.

### 5.3.2 DATA MEMORY PROTECTION

The entire data EEPROM is protected from external reads and writes by the  $\overline{CPD}$  bit in the Configuration Words. When  $\overline{CPD} = 0$ , external reads and writes of EEPROM memory are inhibited and a read will return all '0's. The CPU can continue to read and write EEPROM memory, regardless of the protection bit settings.

## 5.4 Write Protection

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

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

## 5.5 User ID

Four memory locations (8000h-8003h) are designated as ID locations where the user can store checksum or other code identification numbers. These locations are readable and writable during normal execution. See **Section 11.4.7 “NVMREG EEPROM, User ID, Device ID and Configuration Word Access”** for more information on accessing these memory locations. For more information on checksum calculation, see the “PIC16(L)F183XX Memory Programming Specification” (DS40001738).

## 5.6 Device ID and Revision ID

The 14-bit device ID word is located at 8006h and the 14-bit revision ID is located at 8005h. These locations are read-only and cannot be erased or modified. See **Section 11.4.7 “NVMREG EEPROM, User ID, Device ID and Configuration Word Access”** for more information on accessing these memory locations.

Development tools, such as device programmers and debuggers, may be used to read the Device ID and Revision ID.

# PIC16(L)F18325/18345

## 6.2.1 BOR IS ALWAYS ON

When the BOREN bits of Configuration Words are programmed to '11', the BOR is always on. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is active during Sleep. The BOR does not delay wake-up from Sleep.

## 6.2.2 BOR IS OFF IN SLEEP

When the BOREN bits of Configuration Words are programmed to '10', the BOR is on, except in Sleep. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is not active during Sleep, but device wake-up will be delayed until the BOR can determine that VDD is higher than the BOR threshold. The device wake-up will be delayed until the BOR is ready.

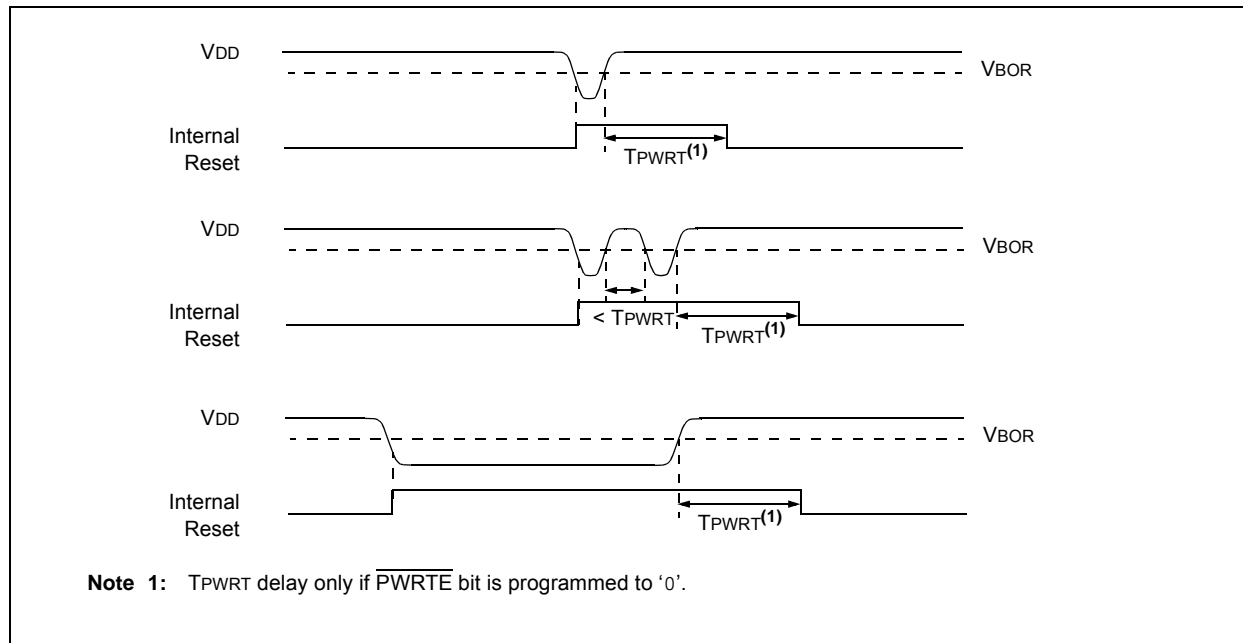
## 6.2.3 BOR CONTROLLED BY SOFTWARE

When the BOREN bits of Configuration Words are programmed to '01', the BOR is controlled by the SBOREN bit of the BORCON register. The device wake from Sleep is not delayed by the BOR Ready condition or the VDD level only when the SBOREN bit is cleared in software and the device is starting up from a non POR/BOR Reset event.

BOR protection begins as soon as the BOR circuit is ready. The status of the BOR circuit is reflected in the BORRDY bit of the BORCON register.

BOR protection is unchanged by Sleep.

**FIGURE 6-2: BROWN-OUT SITUATIONS**



## 6.2.4 BOR ALWAYS OFF

When the BOREN bits of Configuration Word 2 are programmed to '00', the BOR is always disable. In the configuration, setting the SWBOREN bit will have no affect on BOR operation.

## 11.4.3 NVMREG WRITE TO EEPROM

Writing to the EEPROM is accomplished by the following steps:

1. Set the NVMREGS and WREN bits of the NVMCON1 register.
2. Write the desired address (address +7000h) into the NVMADRH:NVMADRL register pair (Table 11-2).
3. Perform the unlock sequence as described in **Section 11.4.2 “NVM Unlock Sequence”**.

A single EEPROM byte is written with NVMDATA. The operation includes an implicit erase cycle for that byte (it is not necessary to set the FREE bit), and requires many instruction cycles to finish. CPU execution continues in parallel and, when complete, WR is cleared by hardware, NVMIF is set, and an interrupt will occur if NVMIE is also set. Software must poll the WR bit to determine when writing is complete, or wait for the interrupt to occur. WREN will remain unchanged.

Once the EEPROM write operation begins, clearing the WR bit will have no effect; the operation will run to completion.

## 11.4.4 NVMREG ERASE OF PROGRAM FLASH MEMORY

Program Flash memory can only be erased one row at a time. No automatic erase occurs upon the initiation of the write to program Flash memory.

To erase a program Flash memory row:

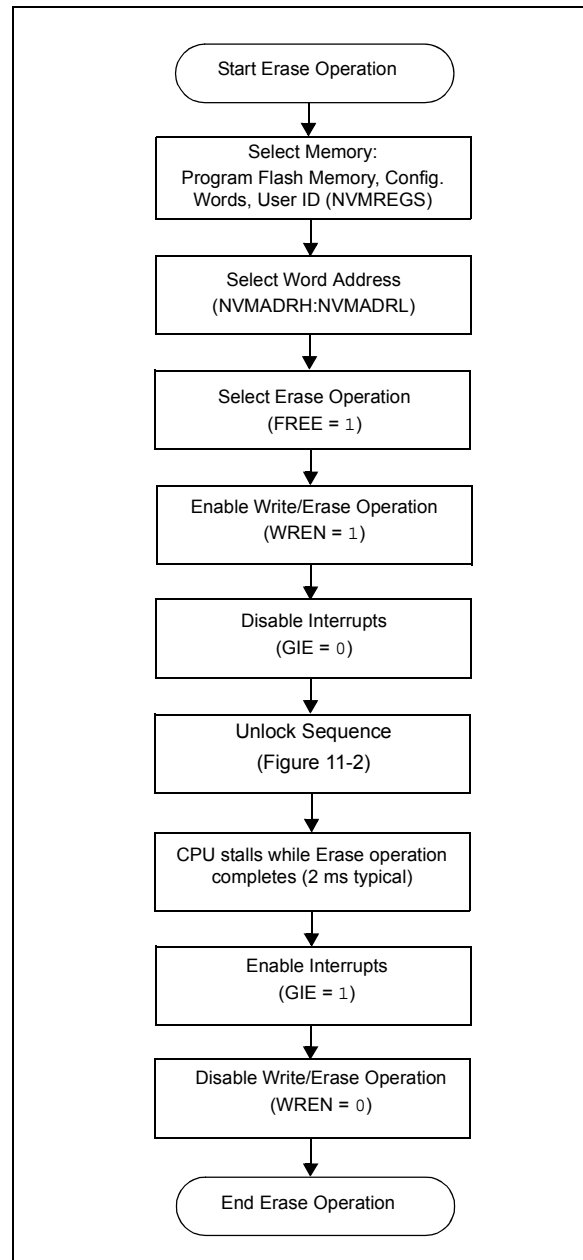
1. Clear the NVMREGS bit of the NVMCON1 register to erase program Flash memory locations, or set the NVMREGS bit to erase user ID locations.
2. Write the desired address into the NVMADRH:NVMADRL register pair (Table 11-2).
3. Set the FREE and WREN bits of the NVMCON1 register.
4. Perform the unlock sequence as described in **Section 11.4.2 “NVM Unlock Sequence”**.

If the program Flash memory address is write-protected, the WR bit will be cleared and the erase operation will not take place.

While erasing program Flash memory, CPU operation is suspended, and resumes when the operation is complete. Upon completion, the NVMIF is set, and an interrupt will occur if the NVMIE bit is also set.

Write latch data is not affected by erase operations, and WREN will remain unchanged.

**FIGURE 11-3: NVM ERASE FLOWCHART**



# PIC16(L)F18325/18345

---

## 11.4.7 NVMREG EEPROM, USER ID, DEVICE ID AND CONFIGURATION WORD ACCESS

Instead of accessing program Flash memory, the EEPROM, the user ID's, Device ID/Revision ID and Configuration Words can be accessed when NVMREGS = 1 in the NVMCON1 register. This is the region that would be pointed to by PC<15> = 1, but not all addresses are accessible. Different access may exist for reads and writes. Refer to Table 11-3.

When read access is initiated on an address outside the parameters listed in Table 11-3, the NVMDATH: NVMDATL register pair is cleared, reading back '0's.

**TABLE 11-3: EEPROM, USER ID, DEV/REV ID AND CONFIGURATION WORD ACCESS (NVMREGS = 1)**

Address	Function	Read Access	Write Access
8000h-8003h	User IDs	Yes	Yes
8005h-8006h	Device ID/Revision ID	Yes	No
8007h-800Ah	Configuration Words 1-4	Yes	No
F000h-F0FFh	EEPROM	Yes	Yes

# PIC16(L)F18325/18345

## 11.4.9 WRERR BIT

The WRERR bit can be used to determine if a write error occurred.

WRERR will be set if one of the following conditions occurs:

- If WR is set while the NVMADRH:NMVADRL points to a write-protected address
- A Reset occurs while a self-write operation was in progress
- An unlock sequence was interrupted

The WRERR bit is normally set by hardware, but can be set by the user for test purposes. Once set, WRERR must be cleared in software.

**TABLE 11-4: ACTIONS FOR PROGRAM FLASH MEMORY WHEN WR = 1**

Free	LWLO	Actions for Program Flash Memory when WR = 1	Comments
0	0	Write the write latch data to program Flash memory row. See <b>Section 11.4.4 “NVMREG Erase of Program Flash Memory”</b>	<ul style="list-style-type: none"><li>• If WP is enabled, WR is cleared and WRERR is set</li><li>• Write latches are reset to 3FFh</li><li>• NVMDATH:NVMDATL is ignored</li></ul>
0	1	Copy NVMDATH:NVMDATL to the write latch corresponding to NVMADR LSBs. See <b>Section 11.4.4 “NVMREG Erase of Program Flash Memory”</b>	<ul style="list-style-type: none"><li>• Write protection is ignored</li><li>• No memory access occurs</li></ul>
1	x	Erase the 32-word row of NVMADRH:NMVADRL location. See <b>Section 11.4.3 “NVMREG Write to EEPROM”</b>	<ul style="list-style-type: none"><li>• If WP is enabled, WR is cleared and WRERR is set</li><li>• All 32 words are erased</li><li>• NVMDATH:NVMDATL is ignored</li></ul>

# PIC16(L)F18325/18345

## REGISTER 12-7: SLRCONA: PORTA SLEW RATE CONTROL REGISTER

U-0	U-0	R/W-1/1	R/W-1/1	U-0	R/W-1/1	R/W-1/1	R/W-1/1
—	—	SLRA5	SLRA4	—	SLRA2	SLRA1	SLRA0
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                      **SLRA<5:4>:** PORTA Slew Rate Enable bits  
                                 For RA<5:4> pins, respectively  
                                 1 = Port pin slew rate is limited  
                                 0 = Port pin slews at maximum rate  
 bit 3                        **Unimplemented:** Read as '0'  
 bit 2-0                      **SLRA<2:0>:** PORTA Slew Rate Enable bits  
                                 For RA<2:0> pins, respectively  
                                 1 = Port pin slew rate is limited  
                                 0 = Port pin slews at maximum rate

## REGISTER 12-8: INLVLA: PORTA INPUT LEVEL CONTROL REGISTER

U-0	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0
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                      **INLVLA<5:0>:** PORTA Input Level Select bits  
                                 For RA<5:0> pins, respectively  
                                 1 = ST input used for PORT reads and interrupt-on-change  
                                 0 = TTL input used for PORT reads and interrupt-on-change

# PIC16(L)F18325/18345

## REGISTER 12-18: TRISC: PORTC TRI-STATE 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
TRISC7 <sup>(1)</sup>	TRISC6 <sup>(1)</sup>	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
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 **TRISC<7:6>**: PORTC Tri-State Control bits<sup>(1)</sup>  
1 = PORTC pin configured as an input (tri-stated)  
0 = PORTC pin configured as an output

bit 5-0 **TRISC<5:0>**: PORTC Tri-State Control bits  
1 = PORTC pin configured as an input (tri-stated)  
0 = PORTC pin configured as an output

**Note 1:** PIC16(L)F18345 only; otherwise read as '0'.

## REGISTER 12-19: LATC: PORTC DATA LATCH 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
LATC7 <sup>(1)</sup>	LATC6 <sup>(1)</sup>	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0
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 **LATC<7:6>**: PORTC Output Latch Value bits<sup>(1)</sup>

bit 5-0 **LATC<5:0>**: PORTC Output Latch Value bits

**Note 1:** PIC16(L)F18345 only; otherwise read as '0'.



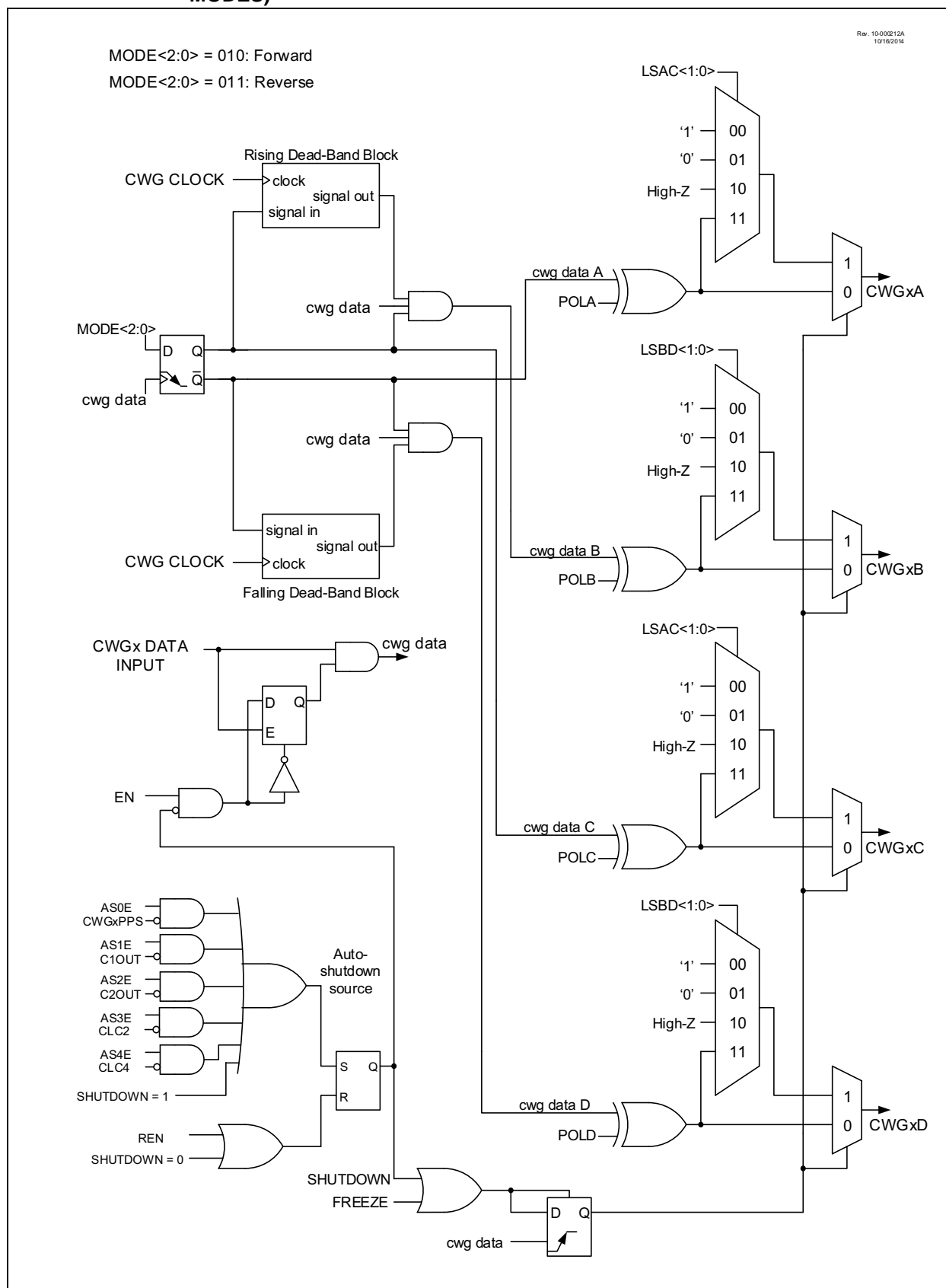
**TABLE 16-1: SUMMARY OF REGISTERS ASSOCIATED WITH FIXED VOLTAGE REFERENCE**

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>		180
ADCON0	CHS<5:0>						GO/DONE	ADON	244
ADCON1	ADFM	ADCS<2:0>			—	ADNREF	ADPREF<1:0>		245
CMxCON1	CxINTP	CxINTN	CxPCH<2:0>			CxNCH<2:0>			191
DAC1CON0	DAC1EN	—	DAC1OE	—	DAC1PPS<1:0>		—	DAC1NSS	263

**Legend:** Shaded cells are not used with the Fixed Voltage Reference.

# PIC16(L)F18325/18345

**FIGURE 20-11: SIMPLIFIED CWG BLOCK DIAGRAM (FORWARD AND REVERSE FULL-BRIDGE MODES)**



# PIC16(L)F18325/18345

## REGISTER 21-3: CLCxSEL0: GENERIC CLCx DATA 0 SELECT REGISTER

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	LCxD1S<5:0>					
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 7-6 **Unimplemented:** Read as '0'

bit 5-0 **LCxD1S<5:0>**: CLCx Data1 Input Selection bits  
See Table 21-1.

## REGISTER 21-4: CLCxSEL1: GENERIC CLCx DATA 1 SELECT REGISTER

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	LCxD2S<5:0>					
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 7-6 **Unimplemented:** Read as '0'

bit 5-0 **LCxD2S<5:0>**: CLCx Data 2 Input Selection bits  
See Table 21-1.

## REGISTER 21-5: CLCxSEL2: GENERIC CLCx DATA 2 SELECT REGISTER

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	LCxD3S<5:0>					
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 7-6 **Unimplemented:** Read as '0'

bit 5-0 **LCxD3S<5:0>**: CLCx Data 3 Input Selection bits  
See Table 21-1.

## 24.0 5-BIT DIGITAL-TO-ANALOG CONVERTER (DAC1) MODULE

The Digital-to-Analog Converter supplies a variable voltage reference, ratiometric with the input source, with 32 selectable output levels.

The input of the DAC can be connected to:

- External VREF pins
- VDD supply voltage
- FVR (Fixed Voltage Reference)

The output of the DAC can be configured to supply a reference voltage to the following:

- Comparator positive input
- ADC input channel
- DAC1OUT pin

The Digital-to-Analog Converter (DAC) is enabled by setting the DAC1EN bit of the DAC1CON0 register.

## 24.1 Output Voltage Selection

The DAC has 32 voltage level ranges. The 32 levels are set with the DAC1R<4:0> bits of the DAC1CON1 register.

The DAC output voltage is determined by Equation 24-1:

### EQUATION 24-1: DAC OUTPUT VOLTAGE

$$V_{OUT} = \left( (V_{SOURCE+} - V_{SOURCE-}) \times \frac{DAC1R\langle 4:0 \rangle}{2^5} \right) + (V_{SOURCE-})$$

$$V_{SOURCE+} = V_{DD} \text{ or } V_{REF+} \text{ or } FVR$$

$$V_{SOURCE-} = V_{SS} \text{ or } V_{REF-}$$

## 24.2 Ratiometric Output Level

The DAC output value is derived using a resistor ladder with each end of the ladder tied to a positive and negative voltage reference input source. If the voltage of either input source fluctuates, a similar fluctuation will result in the DAC output value.

The value of the individual resistors within the ladder can be found in Table 35-15.

## 24.3 DAC Voltage Reference Output

The DAC voltage can be output to the DAC1OUT pin by setting the DAC1OE bit of the DAC1CON0 register. Selecting the DAC reference voltage for output on the DAC1OUT pin automatically overrides the digital output buffer and digital input threshold detector functions, it disables the weak pull-up and the constant-current drive function of that pin. Reading the DAC1OUT pin when it has been configured for DAC reference voltage output will always return a '0'.

Due to the limited current drive capability, a buffer must be used on the DAC voltage reference output for external connections to the DAC1OUT pin. Figure 24-2 shows an example buffering technique.

# PIC16(L)F18325/18345

## REGISTER 25-2: MDSRC: MODULATION SOURCE CONTROL REGISTER

U-0	U-0	U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	—	—	MDMS<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

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

bit 3-0      **MDMS<3:0>** Modulation Source Selection bits

1111 = CLC4 output

1110 = CLC3 output

1101 = CLC2 output

1100 = CLC1 output

1011 = NCO1 output

1010 = EUSART1 TX output

1001 = MSSP2 SDO2 output

1000 = MSSP1 SDO1 output

0111 = C2 (Comparator 2) output

0110 = C1 (Comparator 1) output

0101 = PWM6 output

0100 = PWM5 output

0011 = CCP2 output (PWM Output mode only)

0010 = CCP1 output (PWM Output mode only)

0001 = MDMINPPS

0000 = MDBIT bit of MDCON register is modulation source

# PIC16(L)F18325/18345

**TABLE 28-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER2/4/6**

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	—	—	—	—	—	INTEDG	101
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	BCL1IF	TMR2IF	TMR1IF	108
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	BCL1IE	TMR2IE	TMR1IE	103
PIR2	TMR6IF	C2IF	C1IF	NVMIF	SSP2IF	BCL2IF	TMR4IF	NCO1IF	109
PIE2	TMR6IE	C2IE	C1IE	NVMIE	SSP2IE	BCL2IE	TMR4IE	NCO1IE	104
T2CON	—	T2OUTPS<3:0>				TMR2ON	T2CKPS<1:0>		298
TMR2	TMR2<7:0>								299
PR2	PR2<7:0>								299
T4CON	—	T4OUTPS<3:0>				TMR4ON	T4CKPS<1:0>		298
TMR4	TMR4<7:0>								299
PR4	PR4<7:0>								299
T6CON	—	T6OUTPS<3:0>				TMR6ON	T6CKPS<1:0>		298
TMR6	TMR6<7:0>								299
PR6	PR6<7:0>								299
ADACT	—	—	—	ADACT<4:0>					246
PWMTMRS	—	—	—	—	P6TSEL<1:0>		P5TSEL<1:0>		197
CLCxSELY	—	—	LCxDyS<5:0>						229

**Legend:** — = unimplemented location, read as '0'. Shaded cells are not used for Timer2/4/6 module.

## 30.2.4 SPI SLAVE MODE

In Slave mode, the data is transmitted and received as external clock pulses appear on SCK. When the last bit is latched, the SSPxIF interrupt flag bit is set.

Before enabling the module in SPI Slave mode, the clock line must match the proper Idle state. The clock line can be observed by reading the SCK pin. The Idle state is determined by the CKP bit of the SSPxCON1 register.

While in Slave mode, the external clock is supplied by the external clock source on the SCK pin. This external clock must meet the minimum high and low times as specified in the electrical specifications.

While in Sleep mode, the slave can transmit/receive data. The shift register is clocked from the SCK pin input and when a byte is received, the device will generate an interrupt. If enabled, the device will wake-up from Sleep.

### 30.2.4.1 Daisy-Chain Configuration

The SPI bus can sometimes be connected in a daisy-chain configuration. The first slave output is connected to the second slave input, the second slave output is connected to the third slave input, and so on. The final slave output is connected to the master input. Each slave sends out, during a second group of clock pulses, an exact copy of what was received during the first group of clock pulses. The whole chain acts as one large communication shift register. The daisy-chain feature only requires a single Slave Select line from the master device.

Figure 30-7 shows the block diagram of a typical daisy-chain connection when operating in SPI mode.

In a daisy-chain configuration, only the most recent byte on the bus is required by the slave. Setting the BOEN bit of the SSPxCON3 register will enable writes to the SSPxBUF register, even if the previous byte has not been read. This allows the software to ignore data that may not apply to it.

## 30.2.5 SLAVE SELECT SYNCHRONIZATION

The Slave Select can also be used to synchronize communication. The Slave Select line is held high until the master device is ready to communicate. When the Slave Select line is pulled low, the slave knows that a new transmission is starting.

If the slave fails to receive the communication properly, it will be reset at the end of the transmission, when the Slave Select line returns to a high state. The slave is then ready to receive a new transmission when the Slave Select line is pulled low again. If the Slave Select line is not used, there is a risk that the slave will eventually become out of sync with the master. If the slave misses a bit, it will always be one bit off in future transmissions. Use of the Slave Select line allows the slave and master to align themselves at the beginning of each transmission.

The  $\overline{SS}$  pin allows a Synchronous Slave mode. The SPI must be in Slave mode with  $\overline{SS}$  pin control enabled (SSPxCON1<3:0> = 0100).

When the  $\overline{SS}$  pin is low, transmission and reception are enabled and the SDO pin is driven.

When the  $\overline{SS}$  pin goes high, the SDO pin is no longer driven, even if in the middle of a transmitted byte and becomes a floating output. External pull-up/pull-down resistors may be desirable depending on the application.

- Note 1:** When the SPI is in Slave mode with  $\overline{SS}$  pin control enabled (SSPxCON1<3:0> = 0100), the SPI module will reset if the  $\overline{SS}$  pin is set to VDD.

**2:** When the SPI is used in Slave mode with CKE set; the user must enable  $\overline{SS}$  pin control.

**3:** While operated in SPI Slave mode the SMP bit of the SSPxSTAT register must remain clear.

When the SPI module resets, the bit counter is forced to '0'. This can be done by either forcing the  $\overline{SS}$  pin to a high level or clearing the SSPEN bit.

# PIC16(L)F18325/18345

## 31.1.2.8 Asynchronous Reception Setup

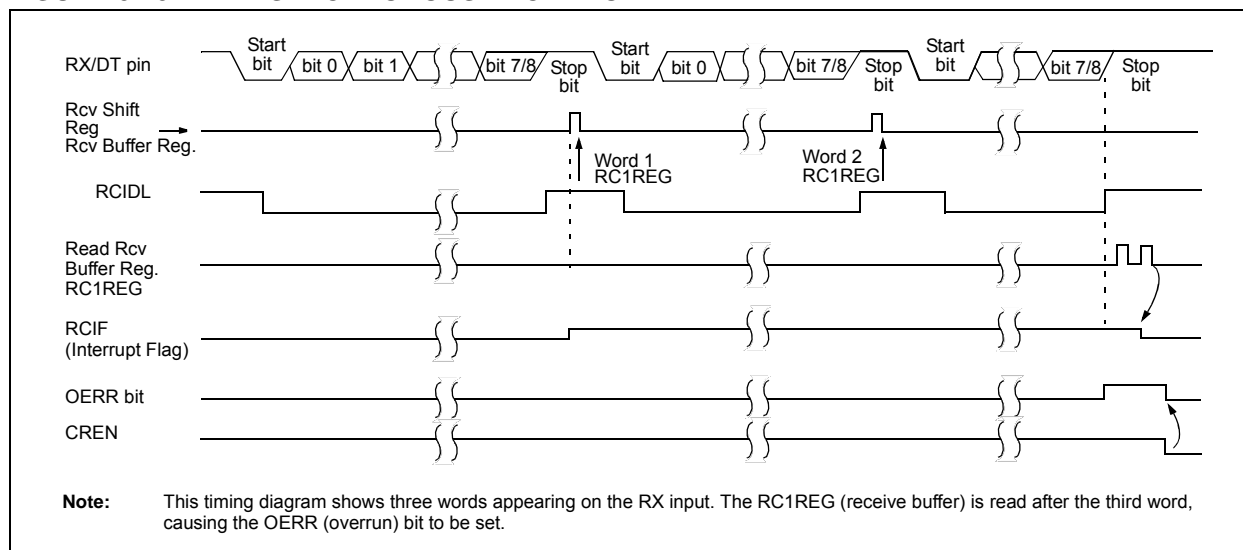
1. Initialize the SP1BRGH, SP1BRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 31.3 “EUSART1 Baud Rate Generator (BRG)”**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
5. If 9-bit reception is desired, set the RX9 bit.
6. Enable reception by setting the CREN bit.
7. The RCIF interrupt flag bit will be set when a character is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
8. Read the RC1STA register to get the error flags and, if 9-bit data reception is enabled, the ninth data bit.
9. Get the received eight Least Significant data bits from the receive buffer by reading the RC1REG register.
10. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.

## 31.1.2.9 9-bit Address Detection Mode Setup

This mode would typically be used in RS-485 systems. To set up an Asynchronous Reception with Address Detect Enable:

1. Initialize the SP1BRGH, SP1BRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 31.3 “EUSART1 Baud Rate Generator (BRG)”**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
5. Enable 9-bit reception by setting the RX9 bit.
6. Enable address detection by setting the ADDEN bit.
7. Enable reception by setting the CREN bit.
8. The RCIF interrupt flag bit will be set when a character with the ninth bit set is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
9. Read the RC1STA register to get the error flags. The ninth data bit will always be set.
10. Get the received eight Least Significant data bits from the receive buffer by reading the RC1REG register. Software determines if this is the device's address.
11. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.
12. If the device has been addressed, clear the ADDEN bit to allow all received data into the receive buffer and generate interrupts.

**FIGURE 31-5: ASYNCHRONOUS RECEPTION**





## 31.5 EUSART1 Operation During Sleep

The EUSART1 will remain active during Sleep only in the Synchronous Slave mode. All other modes require the system clock and therefore cannot generate the necessary signals to run the Transmit or Receive Shift registers during Sleep.

Synchronous Slave mode uses an externally generated clock to run the Transmit and Receive Shift registers.

### 31.5.1 SYNCHRONOUS RECEIVE DURING SLEEP

To receive during Sleep, all the following conditions must be met before entering Sleep mode:

- RC1STA and TX1STA Control registers must be configured for Synchronous Slave Reception (see **Section 31.4.2.4 “Synchronous Slave Reception Set-up”**).
- If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
- The RCIF interrupt flag must be cleared by reading RC1REG to unload any pending characters in the receive buffer.

Upon entering Sleep mode, the device will be ready to accept data and clocks on the RX/DT and TX/CK pins, respectively. When the data word has been completely clocked in by the external device, the RCIF interrupt flag bit of the PIR1 register will be set. Thereby, waking the processor from Sleep.

Upon waking from Sleep, the instruction following the SLEEP instruction will be executed. If the Global Interrupt Enable (GIE) bit of the INTCON register is also set, then the Interrupt Service Routine at address 004h will be called.

### 31.5.2 SYNCHRONOUS TRANSMIT DURING SLEEP

To transmit during Sleep, all the following conditions must be met before entering Sleep mode:

- The RC1STA and TX1STA Control registers must be configured for synchronous slave transmission (see **Section 31.4.2.2 “Synchronous Slave Transmission Set-up”**).
- The TXIF interrupt flag must be cleared by writing the output data to the TX1REG, thereby filling the TSR and transmit buffer.
- If interrupts are desired, set the TXIE bit of the PIE1 register and the PEIE bit of the INTCON register.
- Interrupt enable bits TXIE of the PIE1 register and PEIE of the INTCON register must set.

Upon entering Sleep mode, the device will be ready to accept clocks on TX/CK pin and transmit data on the RX/DT pin. When the data word in the TSR has been completely clocked out by the external device, the pending byte in the TX1REG will transfer to the TSR and the TXIF flag will be set. Thereby, waking the processor from Sleep. At this point, the TX1REG is available to accept another character for transmission, which will clear the TXIF flag.

Upon waking from Sleep, the instruction following the SLEEP instruction will be executed. If the Global Interrupt Enable (GIE) bit is also set then the Interrupt Service Routine at address 0004h will be called.

# PIC16(L)F18325/18345

## REGISTER 31-4: RC1REG<sup>(1)</sup>: RECEIVE DATA REGISTER

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
RC1REG<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 **RC1REG<7:0>**: Lower eight bits of the received data; read-only; see also RX9D (Register 31-2)

**Note 1:** RC1REG (including the ninth bit) is double buffered, and data is available while new data is being received.

## REGISTER 31-5: TX1REG<sup>(1)</sup>: TRANSMIT DATA REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TX1REG<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 **TX1REG<7:0>**: Lower eight bits of the received data; read-only; see also RX9D (Register 31-1)

**Note 1:** TX1REG (including the ninth bit) is double buffered, and can be written when previous data has started shifting.

## REGISTER 31-6: SP1BRGL<sup>(1)</sup>: BAUD RATE GENERATOR REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SP1BRG<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 **SP1BRG<7:0>**: Lower eight bits of the Baud Rate Generator

**Note 1:** Writing to SP1BRG resets the BRG counter.

# PIC16(L)F18325/18345

## MOVWI Move W to INDFn

Syntax: [ *label* ] MOVWI ++FSRn  
[ *label* ] MOVWI --FSRn  
[ *label* ] MOVWI FSRn++  
[ *label* ] MOVWI FSRn--  
[ *label* ] MOVWI k[FSRn]

Operands:  $n \in [0,1]$   
 $mm \in [00,01, 10, 11]$   
 $-32 \leq k \leq 31$

Operation:  $W \rightarrow \text{INDFn}$   
Effective address is determined by

- FSR + 1 (preincrement)
- FSR - 1 (predecrement)
- FSR + k (relative offset)

After the Move, the FSR value will be either:

- FSR + 1 (all increments)
- FSR - 1 (all decrements)

Unchanged

Status Affected: None

Mode	Syntax	mm
Preincrement	++FSRn	00
Predecrement	--FSRn	01
Postincrement	FSRn++	10
Postdecrement	FSRn--	11

Description: This instruction is used to move data between W and one of the indirect registers (INDFn). Before/after this move, the pointer (FSRn) is updated by pre/post incrementing/decrementing it.

**Note:** The INDFn registers are not physical registers. Any instruction that accesses an INDFn register actually accesses the register at the address specified by the FSRn.

FSRn is limited to the range 0000h-FFFFh. Incrementing/decrementing it beyond these bounds will cause it to wrap-around.

The increment/decrement operation on FSRn WILL NOT affect any Status bits.

## NOP No Operation

Syntax: [ *label* ] NOP

Operands: None

Operation: No operation

Status Affected: None

Description: No operation.

Words: 1

Cycles: 1

Example: NOP

## RESET Software Reset

Syntax: [ *label* ] RESET

Operands: None

Operation: Execute a device Reset. Resets the RI flag of the PCON register.

Status Affected: None

Description: This instruction provides a way to execute a hardware Reset by software.

## RETFIE Return from Interrupt

Syntax: [ *label* ] RETFIE k

Operands: None

Operation:  $\text{TOS} \rightarrow \text{PC}$ ,  
 $1 \rightarrow \text{GIE}$

Status Affected: None

Description: Return from Interrupt. Stack is POPed and Top-of-Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a 2-cycle instruction.

Words: 1

Cycles: 2

Example: RETFIE

After Interrupt

PC	=	TOS
GIE	=	1



---

---

## Worldwide Sales and Service

---

---

### AMERICAS

#### Corporate Office

2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 480-792-7200  
Fax: 480-792-7277  
Technical Support:  
<http://www.microchip.com/support>  
Web Address:  
[www.microchip.com](http://www.microchip.com)

#### Atlanta

Duluth, GA  
Tel: 678-957-9614  
Fax: 678-957-1455

#### Austin, TX

Tel: 512-257-3370

#### Boston

Westborough, MA  
Tel: 774-760-0087  
Fax: 774-760-0088

#### Chicago

Itasca, IL  
Tel: 630-285-0071  
Fax: 630-285-0075

#### Cleveland

Independence, OH  
Tel: 216-447-0464  
Fax: 216-447-0643

#### Dallas

Addison, TX  
Tel: 972-818-7423  
Fax: 972-818-2924

#### Detroit

Novi, MI  
Tel: 248-848-4000

#### Houston, TX

Tel: 281-894-5983

#### Indianapolis

Noblesville, IN  
Tel: 317-773-8323  
Fax: 317-773-5453

#### Los Angeles

Mission Viejo, CA  
Tel: 949-462-9523  
Fax: 949-462-9608

#### New York, NY

Tel: 631-435-6000

#### San Jose, CA

Tel: 408-735-9110

#### Canada - Toronto

Tel: 905-695-1980  
Fax: 905-695-2078

### ASIA/PACIFIC

#### Asia Pacific Office

Suites 3707-14, 37th Floor  
Tower 6, The Gateway  
Harbour City, Kowloon

#### Hong Kong

Tel: 852-2943-5100  
Fax: 852-2401-3431

#### Australia - Sydney

Tel: 61-2-9868-6733  
Fax: 61-2-9868-6755

#### China - Beijing

Tel: 86-10-8569-7000  
Fax: 86-10-8528-2104

#### China - Chengdu

Tel: 86-28-8665-5511  
Fax: 86-28-8665-7889

#### China - Chongqing

Tel: 86-23-8980-9588  
Fax: 86-23-8980-9500

#### China - Dongguan

Tel: 86-769-8702-9880

#### China - Guangzhou

Tel: 86-20-8755-8029

#### China - Hangzhou

Tel: 86-571-8792-8115  
Fax: 86-571-8792-8116

#### China - Hong Kong SAR

Tel: 852-2943-5100  
Fax: 852-2401-3431

#### China - Nanjing

Tel: 86-25-8473-2460  
Fax: 86-25-8473-2470

#### China - Qingdao

Tel: 86-532-8502-7355  
Fax: 86-532-8502-7205

#### China - Shanghai

Tel: 86-21-5407-5533  
Fax: 86-21-5407-5066

#### China - Shenyang

Tel: 86-24-2334-2829  
Fax: 86-24-2334-2393

#### China - Shenzhen

Tel: 86-755-8864-2200  
Fax: 86-755-8203-1760

#### China - Wuhan

Tel: 86-27-5980-5300  
Fax: 86-27-5980-5118

#### China - Xian

Tel: 86-29-8833-7252  
Fax: 86-29-8833-7256

### ASIA/PACIFIC

#### China - Xiamen

Tel: 86-592-2388138  
Fax: 86-592-2388130

#### China - Zhuhai

Tel: 86-756-3210040  
Fax: 86-756-3210049

#### India - Bangalore

Tel: 91-80-3090-4444  
Fax: 91-80-3090-4123

#### India - New Delhi

Tel: 91-11-4160-8631  
Fax: 91-11-4160-8632

#### India - Pune

Tel: 91-20-3019-1500

#### Japan - Osaka

Tel: 81-6-6152-7160  
Fax: 81-6-6152-9310

#### Japan - Tokyo

Tel: 81-3-6880-3770  
Fax: 81-3-6880-3771

#### Korea - Daegu

Tel: 82-53-744-4301  
Fax: 82-53-744-4302

#### Korea - Seoul

Tel: 82-2-554-7200  
Fax: 82-2-558-5932 or  
82-2-558-5934

#### Malaysia - Kuala Lumpur

Tel: 60-3-6201-9857  
Fax: 60-3-6201-9859

#### Malaysia - Penang

Tel: 60-4-227-8870  
Fax: 60-4-227-4068

#### Philippines - Manila

Tel: 63-2-634-9065  
Fax: 63-2-634-9069

#### Singapore

Tel: 65-6334-8870  
Fax: 65-6334-8850

#### Taiwan - Hsin Chu

Tel: 886-3-5778-366  
Fax: 886-3-5770-955

#### Taiwan - Kaohsiung

Tel: 886-7-213-7828

#### Taiwan - Taipei

Tel: 886-2-2508-8600  
Fax: 886-2-2508-0102

#### Thailand - Bangkok

Tel: 66-2-694-1351  
Fax: 66-2-694-1350

### EUROPE

#### Austria - Wels

Tel: 43-7242-2244-39  
Fax: 43-7242-2244-393

#### Denmark - Copenhagen

Tel: 45-4450-2828  
Fax: 45-4485-2829

#### France - Paris

Tel: 33-1-69-53-63-20  
Fax: 33-1-69-30-90-79

#### Germany - Dusseldorf

Tel: 49-2129-3766400

#### Germany - Karlsruhe

Tel: 49-721-625370

#### Germany - Munich

Tel: 49-89-627-144-0  
Fax: 49-89-627-144-44

#### Italy - Milan

Tel: 39-0331-742611  
Fax: 39-0331-466781

#### Italy - Venice

Tel: 39-049-7625286

#### Netherlands - Drunen

Tel: 31-416-690399  
Fax: 31-416-690340

#### Poland - Warsaw

Tel: 48-22-3325737

#### Spain - Madrid

Tel: 34-91-708-08-90  
Fax: 34-91-708-08-91

#### Sweden - Stockholm

Tel: 46-8-5090-4654

#### UK - Wokingham

Tel: 44-118-921-5800  
Fax: 44-118-921-5820

06/23/16