



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	20MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	25
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	192 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 11x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf723a-e-so

PIC16(L)F722A/723A

2.5 Indirect Addressing, INDF and FSR Registers

The INDF register is not a physical register. Addressing the INDF register will cause indirect addressing.

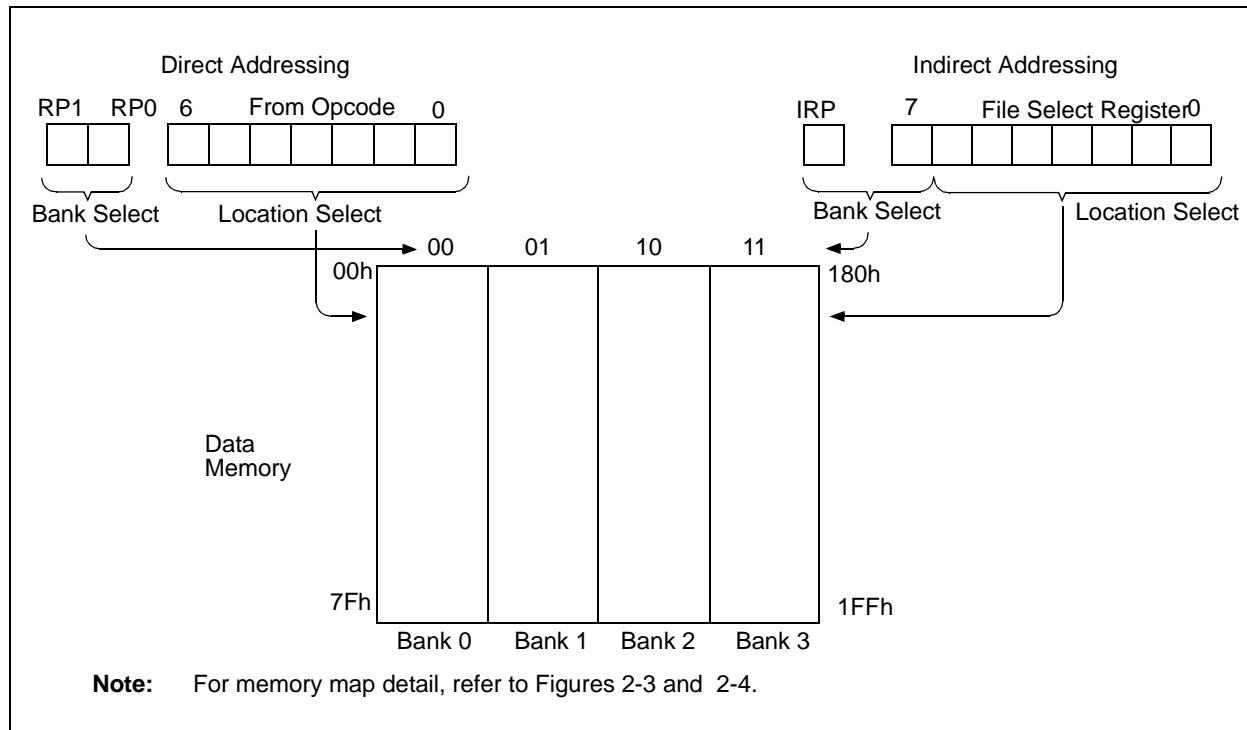
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select Register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no operation (although Status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit of the STATUS register, as shown in Figure 2-6.

A simple program to clear RAM location 020h-02Fh using indirect addressing is shown in Example 2-2.

EXAMPLE 2-2: INDIRECT ADDRESSING

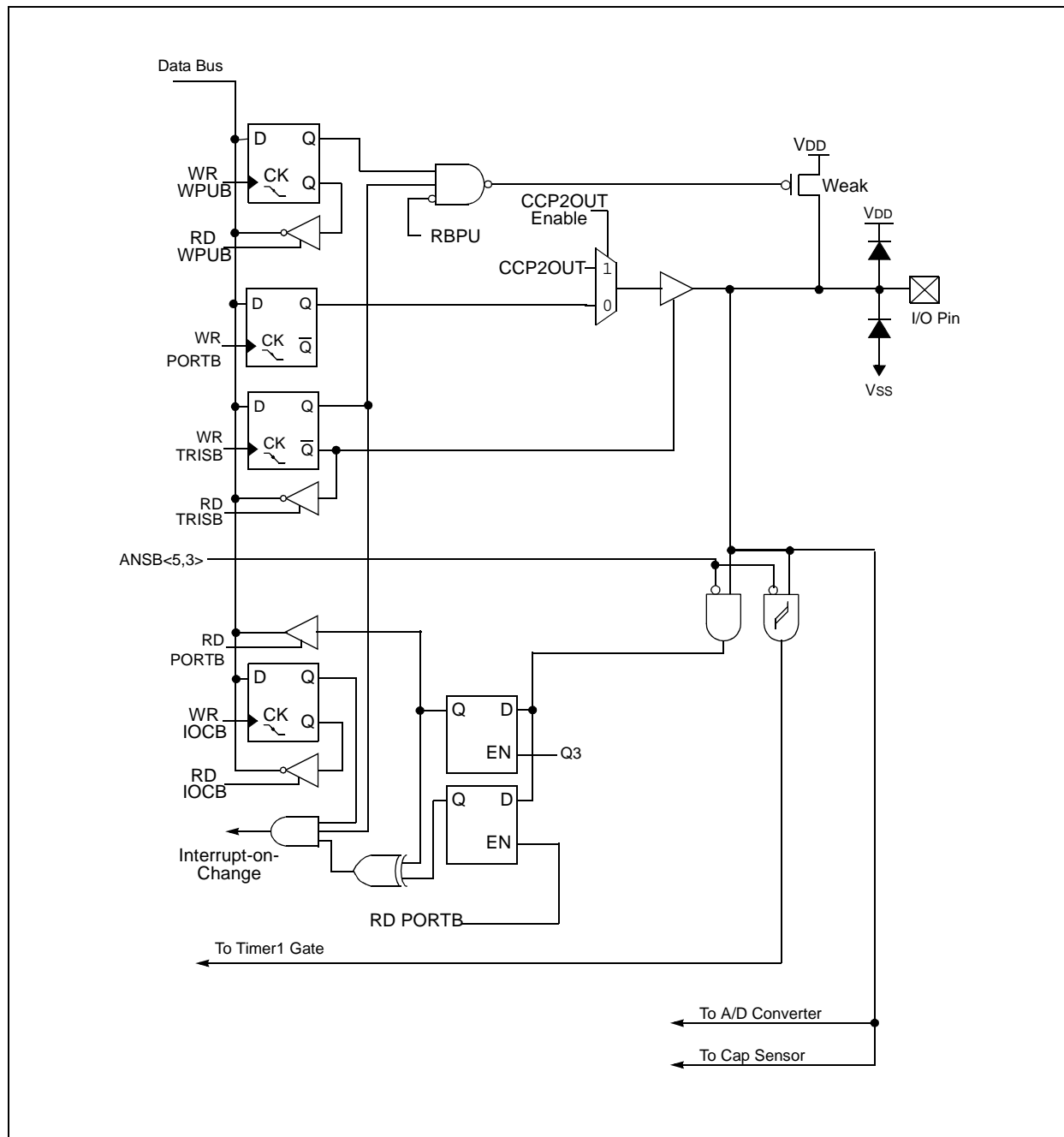
```
MOVLW020h ;initialize pointer
MOVWFFSR ;to RAM
BANKISEL020h
NEXTCLRFINDF ;clear INDF register
INCFFSR ;inc pointer
BTFSSFSR,4 ;all done?
GOTONEXT ;no clear next
CONTINUE ;yes continue
```

FIGURE 2-6: DIRECT/INDIRECT ADDRESSING



PIC16(L)F722A/723A

FIGURE 6-10: BLOCK DIAGRAM OF RB5



7.2 Clock Source Modes

Clock source modes can be classified as external or internal.

- Internal clock source (INTOSC) is contained within the oscillator module and derived from a 500 kHz high precision oscillator. The oscillator module has eight selectable output frequencies, with a maximum internal frequency of 16 MHz.
- External clock modes rely on external circuitry for the clock source. Examples are: oscillator modules (EC mode), quartz crystal resonators or ceramic resonators (LP, XT and HS modes) and Resistor-Capacitor (RC) mode circuits.

The system clock can be selected between external or internal clock sources via the FOSC bits of the Configuration Word 1.

7.3 Internal Clock Modes

The oscillator module has eight output frequencies derived from a 500 kHz high precision oscillator. The IRCF bits of the OSCCON register select the postscaler applied to the clock source dividing the frequency by 1, 2, 4 or 8. Setting the PLEN bit of the Configuration Word 1 locks the internal clock source to 16 MHz before the postscaler is selected by the IRCF bits. The PLEN bit must be set or cleared at the time of programming; therefore, only the upper or low four clock source frequencies are selectable in software.

7.3.1 INTOSC AND INTOSCIO MODES

The INTOSC and INTOSCIO modes configure the internal oscillators as the system clock source when the device is programmed using the oscillator selection or the FOSC<2:0> bits in the CONFIG1 register. See **Section 8.0 “Device Configuration”** for more information.

In INTOSC mode, OSC1/CLKIN is available for general purpose I/O. OSC2/CLKOUT outputs the selected internal oscillator frequency divided by 4. The CLKOUT signal may be used to provide a clock for external circuitry, synchronization, calibration, test or other application requirements.

In INTOSCIO mode, OSC1/CLKIN and OSC2/CLKOUT are available for general purpose I/O.

7.3.2 FREQUENCY SELECT BITS (IRCF)

The output of the 500 kHz INTOSC and 16 MHz INTOSC, with Phase-Locked Loop enabled, connect to a postscaler and multiplexer (see Figure 7-1). The Internal Oscillator Frequency Select bits (IRCF) of the OSCCON register select the frequency output of the internal oscillator. Depending upon the PLEN bit, one of four frequencies of two frequency sets can be selected via software:

If PLEN = 1, frequency selection is as follows:

- 16 MHz
- 8 MHz (Default after Reset)
- 4 MHz
- 2 MHz

If PLEN = 0, frequency selection is as follows:

- 500 kHz
- 250 kHz (Default after Reset)
- 125 kHz
- 62.5 kHz

Note: Following any Reset, the IRCF<1:0> bits of the OSCCON register are set to '10' and the frequency selection is set to 8 MHz or 250 kHz. The user can modify the IRCF bits to select a different frequency.

There is no start-up delay before a new frequency selected in the IRCF bits takes effect. This is because the old and new frequencies are derived from INTOSC via the postscaler and multiplexer.

Start-up delay specifications are located in Table 23-2 in **Section 23.0 “Electrical Specifications”**.

PIC16(L)F722A/723A

REGISTER 9-2: ADCON1: A/D CONTROL REGISTER 1

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
—	ADCS2	ADCS1	ADCS0	—	—	ADREF1	ADREF0
bit 7				bit 0			

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

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

bit 6-4 **ADCS<2:0>:** A/D Conversion Clock Select bits

000 = Fosc/2

001 = Fosc/8

010 = Fosc/32

011 = FRC (clock supplied from a dedicated RC oscillator)

100 = Fosc/4

101 = Fosc/16

110 = Fosc/64

111 = FRC (clock supplied from a dedicated RC oscillator)

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

bit 1-0 **ADREF<1:0>:** Voltage Reference Configuration bits

0x = VREF is connected to VDD

10 = VREF is connected to external VREF (RA3/AN3)

11 = VREF is connected to internal Fixed Voltage Reference

REGISTER 9-3: ADRES: ADC RESULT REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0
bit 7				bit 0			

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 **ADRES<7:0>:** ADC Result Register bits

8-bit conversion result.

PIC16(L)F722A/723A

12.0 TIMER1 MODULE WITH GATE CONTROL

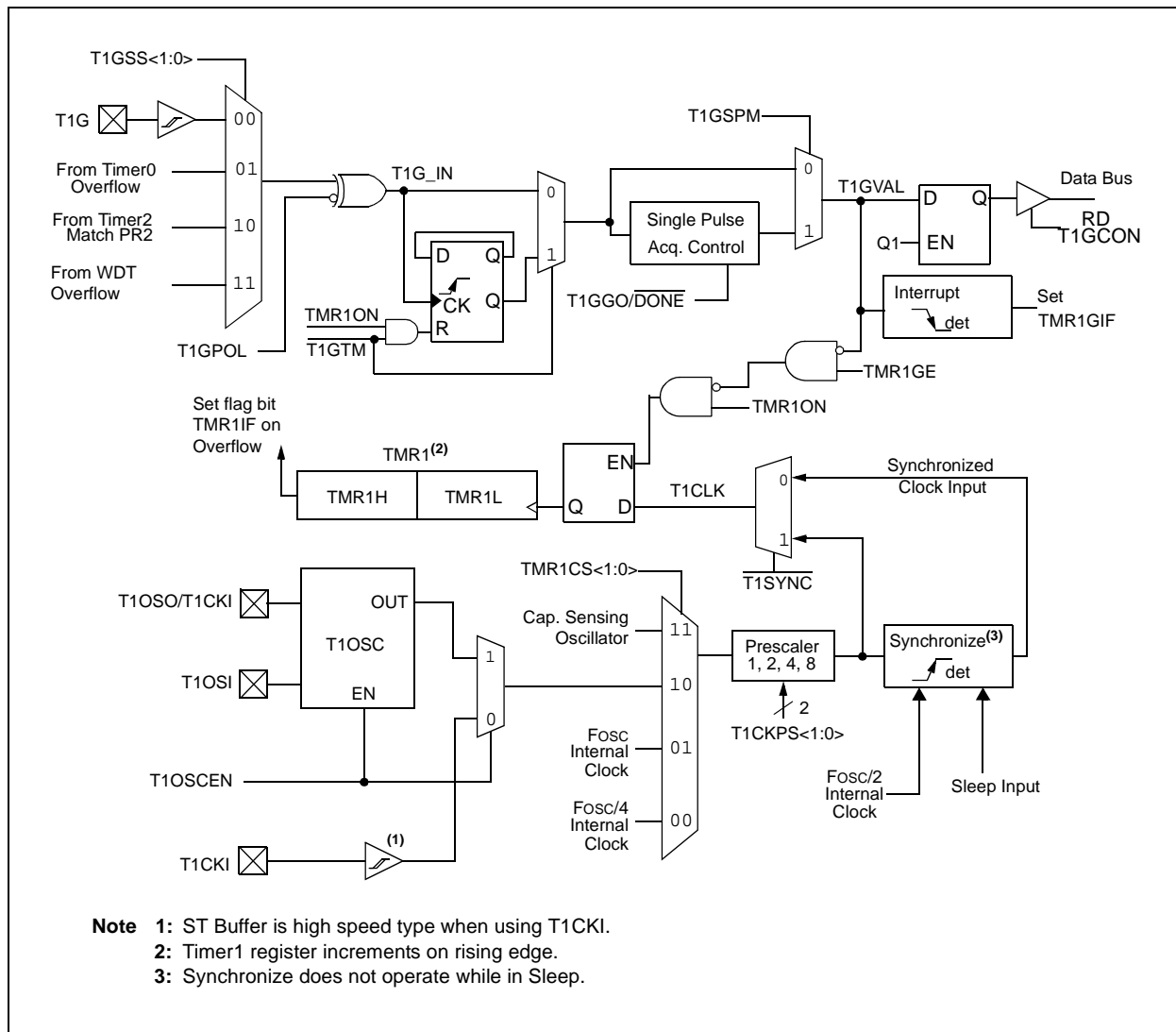
The Timer1 module is a 16-bit timer/counter with the following features:

- 16-bit timer/counter register pair (TMR1H:TMR1L)
- Programmable internal or external clock source
- 3-bit prescaler
- Dedicated LP oscillator circuit
- Synchronous or asynchronous operation
- 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
- Special Event Trigger (with CCP)

- Selectable Gate Source Polarity
- Gate Toggle Mode
- Gate Single-pulse Mode
- Gate Value Status
- Gate Event Interrupt

Figure 12-1 is a block diagram of the Timer1 module.

FIGURE 12-1: TIMER1 BLOCK DIAGRAM



PIC16(L)F722A/723A

13.0 TIMER2 MODULE

The Timer2 module is an 8-bit timer with the following features:

- 8-bit timer register (TMR2)
- 8-bit period register (PR2)
- Interrupt on TMR2 match with PR2
- Software programmable prescaler (1:1, 1:4, 1:16)
- Software programmable postscaler (1:1 to 1:16)

See Figure 13-1 for a block diagram of Timer2.

13.1 Timer2 Operation

The clock input to the Timer2 module is the system instruction clock ($F_{osc}/4$). The clock is fed into the Timer2 prescaler, which has prescale options of 1:1, 1:4 or 1:16. The output of the prescaler is then used to increment the TMR2 register.

The values of TMR2 and PR2 are constantly compared to determine when they match. TMR2 will increment from 00h until it matches the value in PR2. When a match occurs, two things happen:

- TMR2 is reset to 00h on the next increment cycle.
- The Timer2 postscaler is incremented.

The match output of the Timer2/PR2 comparator is then fed into the Timer2 postscaler. The postscaler has postscale options of 1:1 to 1:16 inclusive. The output of the Timer2 postscaler is used to set the TMR2IF interrupt flag bit in the PIR1 register.

The TMR2 and PR2 registers are both fully readable and writable. On any Reset, the TMR2 register is set to 00h and the PR2 register is set to FFh.

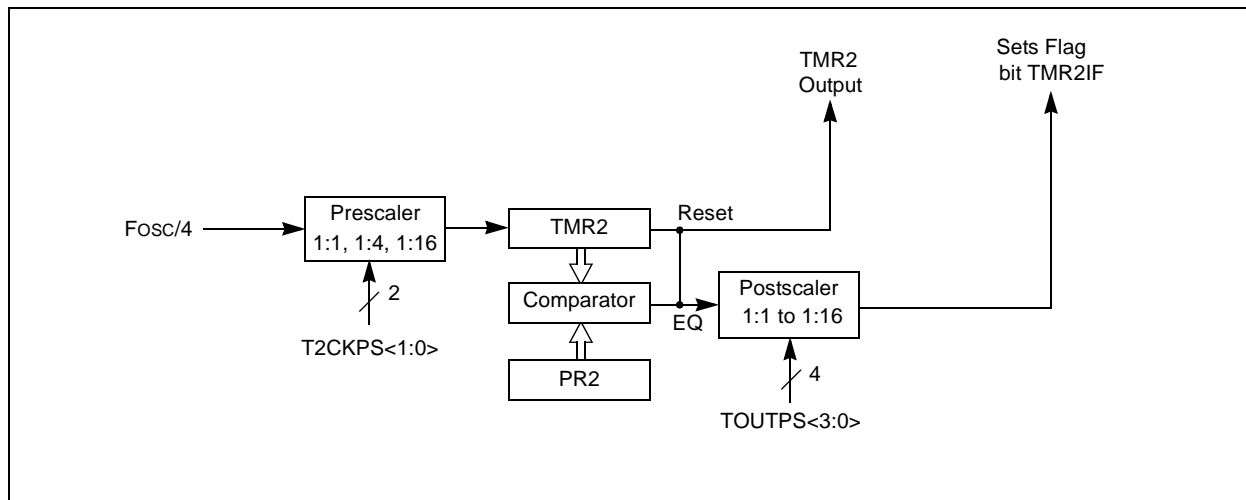
Timer2 is turned on by setting the TMR2ON bit in the T2CON register to a '1'. Timer2 is turned off by clearing the TMR2ON bit to a '0'.

The Timer2 prescaler is controlled by the T2CKPS bits in the T2CON register. The Timer2 postscaler is controlled by the TOUTPS bits in the T2CON register. The prescaler and postscaler counters are cleared when:

- A write to TMR2 occurs.
- A write to T2CON occurs.
- Any device Reset occurs (Power-on Reset, \overline{MCLR} Reset, Watchdog Timer Reset, or Brown-out Reset).

Note: TMR2 is not cleared when T2CON is written.

FIGURE 13-1: TIMER2 BLOCK DIAGRAM



PIC16(L)F722A/723A

15.3 PWM Mode

The PWM mode generates a Pulse-Width Modulated signal on the CCPx pin. The duty cycle, period and resolution are determined by the following registers:

- PR2
- T2CON
- CCPRxL
- CCPxCON

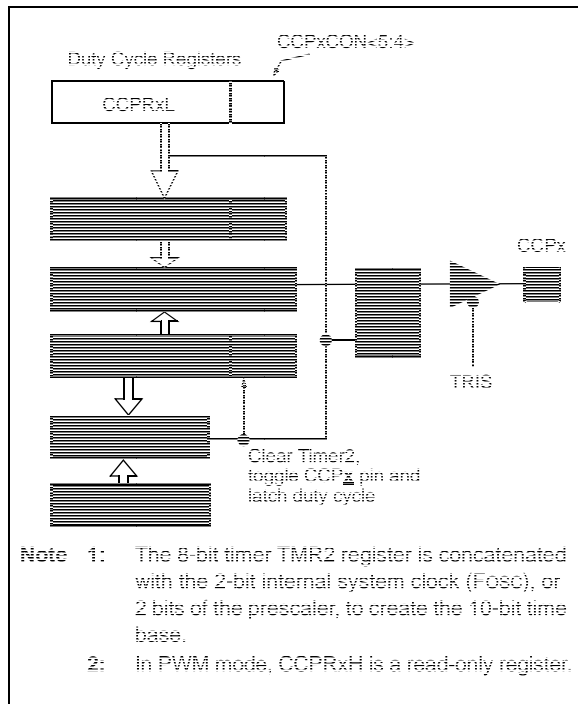
In Pulse-Width Modulation (PWM) mode, the CCP module produces up to a 10-bit resolution PWM output on the CCPx pin.

Figure 15-3 shows a simplified block diagram of PWM operation.

Figure 15-4 shows a typical waveform of the PWM signal.

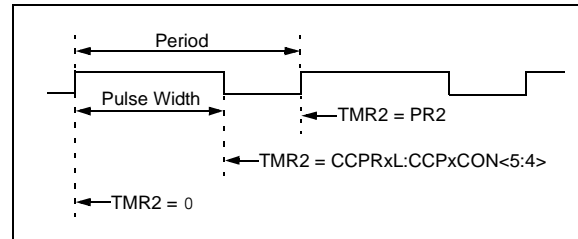
For a step-by-step procedure on how to set up the CCP module for PWM operation, refer to **Section 15.3.8 “Setup for PWM Operation”**.

FIGURE 15-3: SIMPLIFIED PWM BLOCK DIAGRAM



The PWM output (Figure 15-4) has a time base (period) and a time that the output stays high (duty cycle).

FIGURE 15-4: CCP PWM OUTPUT



15.3.1 CCPX PIN CONFIGURATION

In PWM mode, the CCPx pin is multiplexed with the PORT data latch. The user must configure the CCPx pin as an output by clearing the associated TRIS bit.

Either RC1 or RB3 can be selected as the CCP2 pin. Refer to **Section 6.1 “Alternate Pin Function”** for more information.

Note: Clearing the CCPxCON register will relinquish CCPx control of the CCPx pin.

16.1.2.8 Asynchronous Reception Setup:

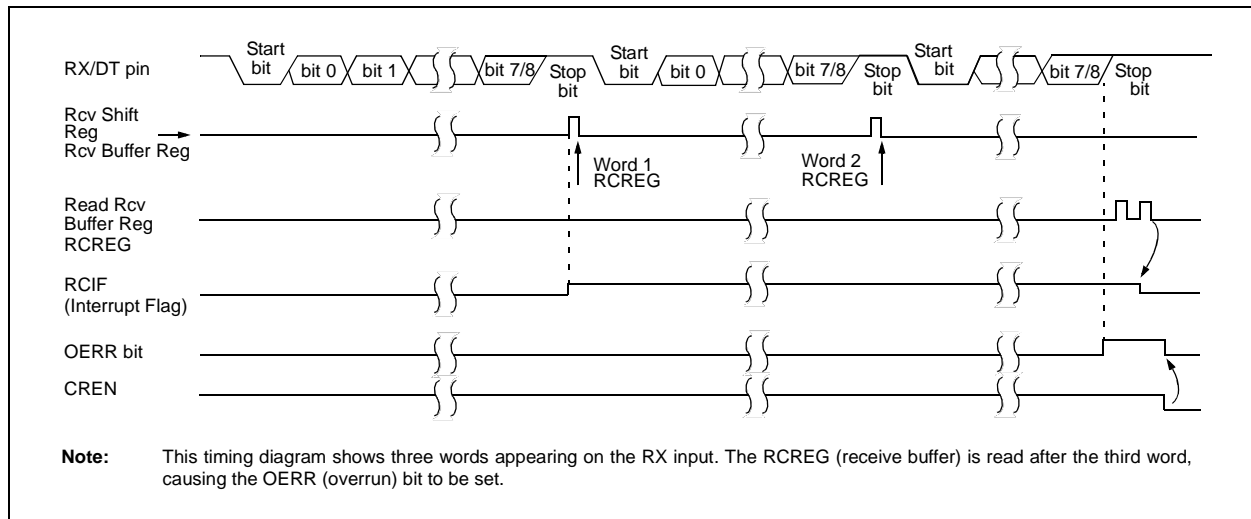
1. Initialize the SPBRG register and the BRGH bit to achieve the desired baud rate (refer to **Section 16.2 “AUSART Baud Rate Generator (BRG)”**).
2. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
3. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
4. If 9-bit reception is desired, set the RX9 bit.
5. Enable reception by setting the CREN bit.
6. The RCIF interrupt flag bit of the PIR1 register will be set when a character is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE bit of the PIE1 register was also set.
7. Read the RCSTA register to get the error flags and, if 9-bit data reception is enabled, the ninth data bit.
8. Get the received eight Least Significant data bits from the receive buffer by reading the RCREG register.
9. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.

16.1.2.9 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 SPBRG register and the BRGH bit to achieve the desired baud rate (refer to **Section 16.2 “AUSART Baud Rate Generator (BRG)”**).
2. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
3. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
4. Enable 9-bit reception by setting the RX9 bit.
5. Enable address detection by setting the ADDEN bit.
6. Enable reception by setting the CREN bit.
7. The RCIF interrupt flag bit of the PIR1 register 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 of the PIE1 register was also set.
8. Read the RCSTA register to get the error flags. The ninth data bit will always be set.
9. Get the received eight Least Significant data bits from the receive buffer by reading the RCREG register. Software determines if this is the device's address.
10. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.
11. If the device has been addressed, clear the ADDEN bit to allow all received data into the receive buffer and generate interrupts.

FIGURE 16-5: ASYNCHRONOUS RECEPTION



PIC16(L)F722A/723A

16.3.1.4 Synchronous Master Reception

Data is received at the RX/DT pin. The RX/DT pin output driver is automatically disabled when the AUSART is configured for synchronous master receive operation.

In Synchronous mode, reception is enabled by setting either the Single Receive Enable bit (SREN of the RCSTA register) or the Continuous Receive Enable bit (CREN of the RCSTA register).

When SREN is set and CREN is clear, only as many clock cycles are generated as there are data bits in a single character. The SREN bit is automatically cleared at the completion of one character. When CREN is set, clocks are continuously generated until CREN is cleared. If CREN is cleared in the middle of a character the CK clock stops immediately and the partial character is discarded. If SREN and CREN are both set, then SREN is cleared at the completion of the first character and CREN takes precedence.

To initiate reception, set either SREN or CREN. Data is sampled at the RX/DT pin on the trailing edge of the TX/CK clock pin and is shifted into the Receive Shift Register (RSR). When a complete character is received into the RSR, the RCIF bit of the PIR1 register is set and the character is automatically transferred to the two character receive FIFO. The Least Significant eight bits of the top character in the receive FIFO are available in RCREG. The RCIF bit remains set as long as there are un-read characters in the receive FIFO.

16.3.1.5 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.

16.3.1.6 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 RCREG is read to access the FIFO. When this happens the OERR bit of the RCSTA 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 RCREG. If the overrun occurred when the CREN bit is set then the error condition is cleared by either clearing the CREN bit of the RCSTA register.

16.3.1.7 Receiving 9-bit Characters

The AUSART supports 9-bit character reception. When the RX9 bit of the RCSTA register is set, the AUSART will shift nine bits into the RSR for each character received. The RX9D bit of the RCSTA 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 RCREG.

Address detection in Synchronous modes is not supported, therefore the ADDEN bit of the RCSTA register must be cleared.

16.3.1.8 Synchronous Master Reception Setup:

1. Initialize the SPBRG register for the appropriate baud rate. Set or clear the BRGH bit, as required, to achieve the desired baud rate.
2. Enable the synchronous master serial port by setting bits SYNC, SPEN and CSRC.
3. Ensure bits CREN and SREN are clear.
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 bit RX9.
6. Verify address detection is disabled by clearing the ADDEN bit of the RCSTA register.
7. Start reception by setting the SREN bit or for continuous reception, set the CREN bit.
8. Interrupt flag bit RCIF of the PIR1 register will be set when reception of a character is complete. An interrupt will be generated if the RCIE interrupt enable bit of the PIE1 register was set.
9. Read the RCSTA 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 RCREG register.
11. If an overrun error occurs, clear the error by either clearing the CREN bit of the RCSTA register or by clearing the SPEN bit, which resets the AUSART.

17.1.2.4 Slave Select Operation

The \overline{SS} pin allows Synchronous Slave mode operation. The SPI must be in Slave mode with \overline{SS} pin control enabled ($SSPM<3:0> = 0100$). The associated TRIS bit for the \overline{SS} pin must be set, making \overline{SS} an input.

In Slave Select mode, when:

- $\overline{SS} = 0$, The device operates as specified in **Section 17.1.2 “Slave Mode”**.
- $\overline{SS} = 1$, The SPI module is held in Reset and the SDO pin will be tri-stated.

Note 1: When the SPI is in Slave mode with \overline{SS} pin control enabled ($SSPM<3:0> = 0100$), the SPI module will reset if the \overline{SS} pin is driven high.

2: If the SPI is used in Slave mode with CKE set, the \overline{SS} pin control must be enabled.

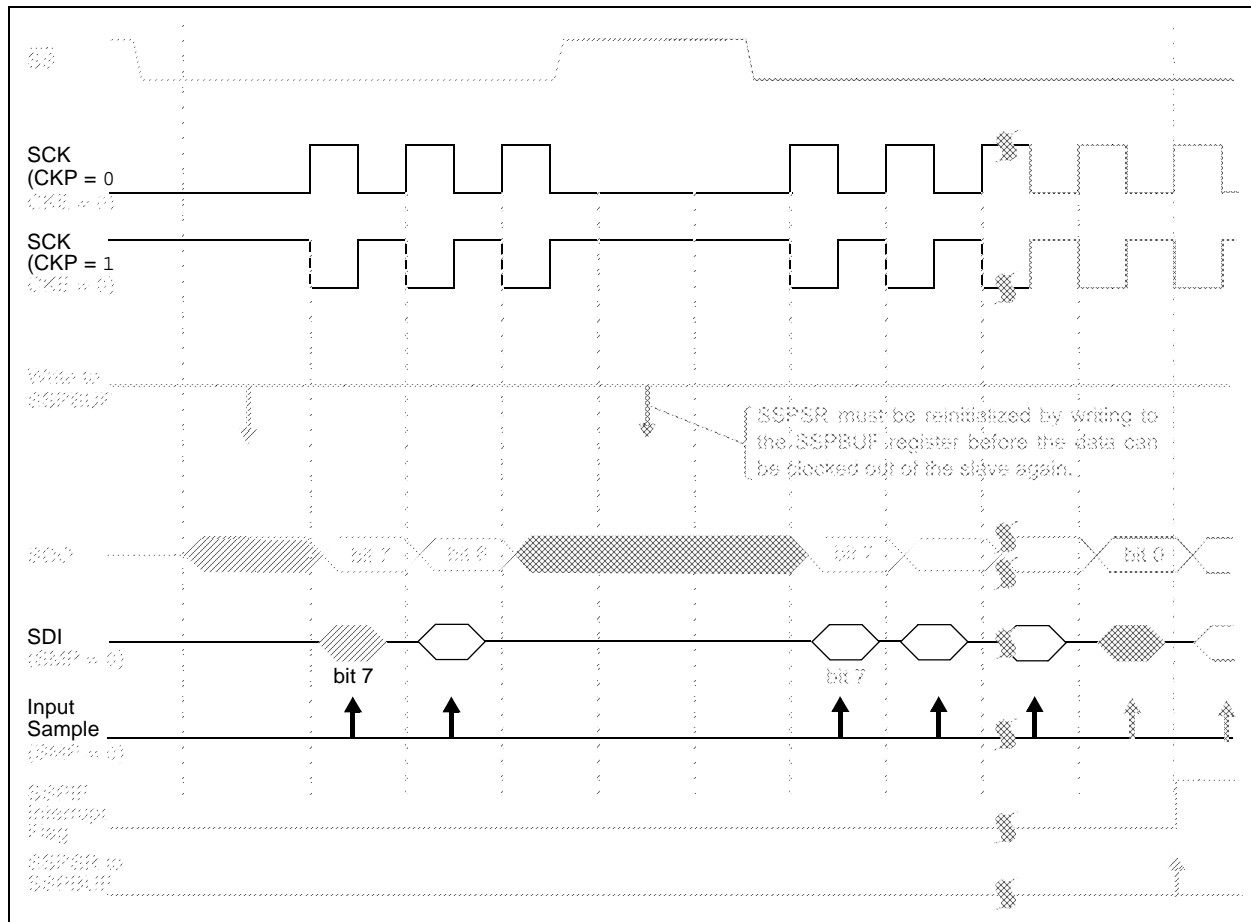
When the SPI module resets, the bit counter is cleared to '0'. This can be done by either forcing the \overline{SS} pin to a high level or clearing the SSPEN bit. Figure 17-6 shows the timing waveform for such a synchronization event.

Note: SSPSR must be reinitialized by writing to the SSPBUF register before the data can be clocked out of the slave again.

17.1.2.5 Sleep in Slave Mode

While in Sleep mode, the slave can transmit/receive data. The SPI Transmit/Receive Shift register operates asynchronously to the device on the externally supplied clock source. This allows the device to be placed in Sleep mode and data to be shifted into the SPI Transmit/Receive Shift register. When all eight bits have been received, the SSP Interrupt Flag bit will be set and if enabled, will wake the device from Sleep.

FIGURE 17-6: SLAVE SELECT SYNCHRONIZATION WAVEFORM



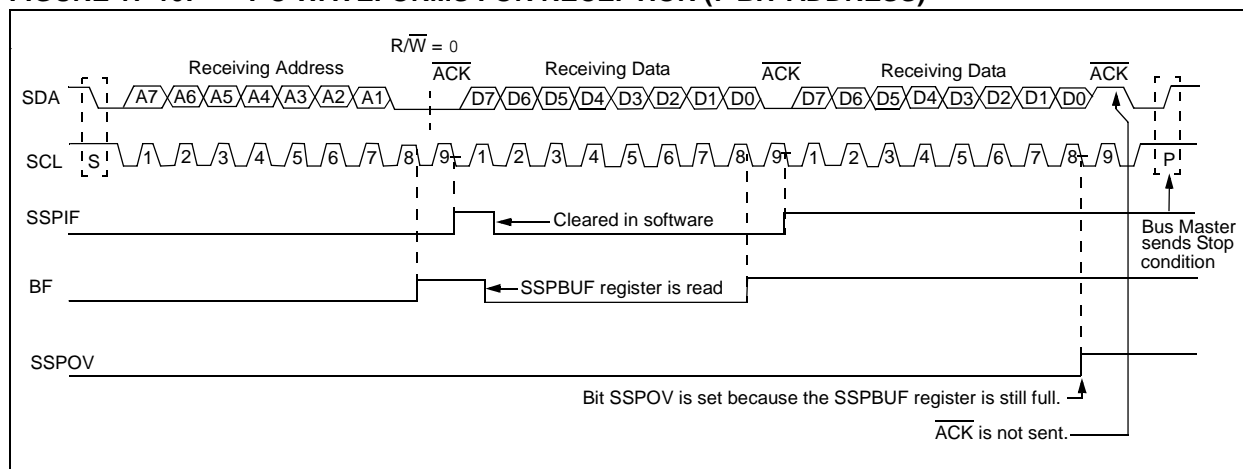
PIC16(L)F722A/723A

17.2.5 RECEPTION

When the R/\overline{W} bit of the received address byte is clear, the master will write data to the slave. If an address match occurs, the received address is loaded into the SSPBUF register. An address byte overflow will occur if that loaded address is not read from the SSPBUF before the next complete byte is received.

An SSP interrupt is generated for each data transfer byte. The BF, R/W and D/A bits of the SSPSTAT register are used to determine the status of the last received byte.

FIGURE 17-10: I²C WAVEFORMS FOR RECEPTION (7-BIT ADDRESS)



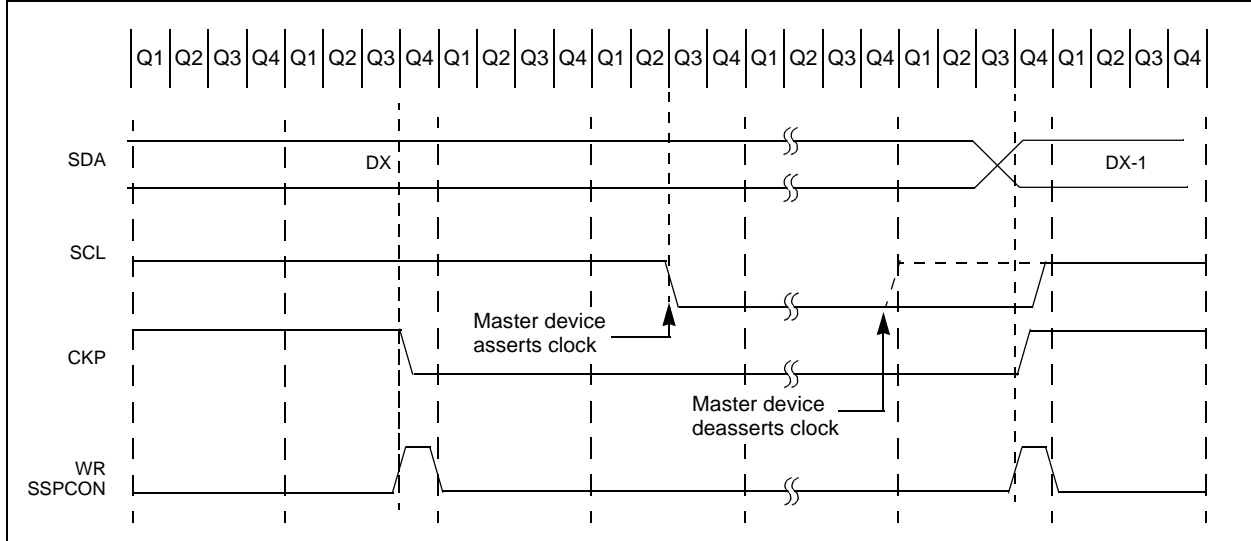
17.2.10 CLOCK SYNCHRONIZATION

When the CKP bit is cleared, the SCL output is held low once it is sampled low. Therefore, the CKP bit will not stretch the SCL line until an external I²C master device has already asserted the SCL line low. The SCL output will remain low until the CKP bit is set and all other devices on the I²C bus have released SCL. This ensures that a write to the CKP bit will not violate the minimum high-time requirement for SCL (Figure 17-14).

17.2.11 SLEEP OPERATION

While in Sleep mode, the I²C module can receive addresses of data, and when an address match or complete byte transfer occurs, wake the processor from Sleep (if SSP interrupt is enabled).

FIGURE 17-14: CLOCK SYNCHRONIZATION TIMING



PIC16(L)F722A/723A

REGISTER 18-1: PMCON1: PROGRAM MEMORY CONTROL 1 REGISTER

R-1	U-0	U-0	U-0	U-0	U-0	U-0	R/S-0
Reserved	—	—	—	—	—	—	RD
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

S = Setable bit, cleared in hardware

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7 **Reserved:** Read as '1'. Maintain this bit set.

bit 6-1 **Unimplemented:** Read as '0'

bit 0 **RD:** Read Control bit

1 = Initiates an program memory read (The RD is cleared in hardware; the RD bit can only be set (not cleared) in software).

0 = Does not initiate a program memory read

REGISTER 18-2: PMDATH: PROGRAM MEMORY DATA HIGH REGISTER

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	PMD13	PMD12	PMD11	PMD10	PMD9	PMD8
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

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

bit 5-0 **PMD<13:8>:** The value of the program memory word pointed to by PMADRH and PMADRL after a Program Memory Read command.

REGISTER 18-3: PMDATL: PROGRAM MEMORY DATA LOW REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
PMD7	PMD6	PMD5	PMD4	PMD3	PMD2	PMD1	PMD0
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 **PMD<7:0>:** The value of the program memory word pointed to by PMADRH and PMADRL after a Program Memory Read command.

RETFIE	Return from Interrupt
Syntax:	[<i>label</i>] RETFIE
Operands:	None
Operation:	TOS → PC, 1 → 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
<u>Example:</u>	<pre>RETFIE</pre> <p>After Interrupt</p> <pre>PC = TOS GIE = 1</pre>

RETLW	Return with literal in W
Syntax:	[<i>label</i>] RETLW k
Operands:	0 ≤ k ≤ 255
Operation:	k → (W); TOS → PC
Status Affected:	None
Description:	The W register is loaded with the 8-bit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a 2-cycle instruction.
Words:	1
Cycles:	2
<u>Example:</u>	<pre>CALL TABLE;W contains table ;offset value • ;W now has table value • • ADDWF PC ;W = offset RETLW k1 ;Begin table RETLW k2 ; • • • RETLW kn ; End of table</pre> <p>Before Instruction W = 0x07</p> <p>After Instruction W = value of k8</p>

RETURN	Return from Subroutine
Syntax:	[<i>label</i>] RETURN
Operands:	None
Operation:	TOS → PC
Status Affected:	None
Description:	Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a 2-cycle instruction.

23.7 AC Characteristics: PIC16F722A/723A-I/E

FIGURE 23-3: CLOCK TIMING

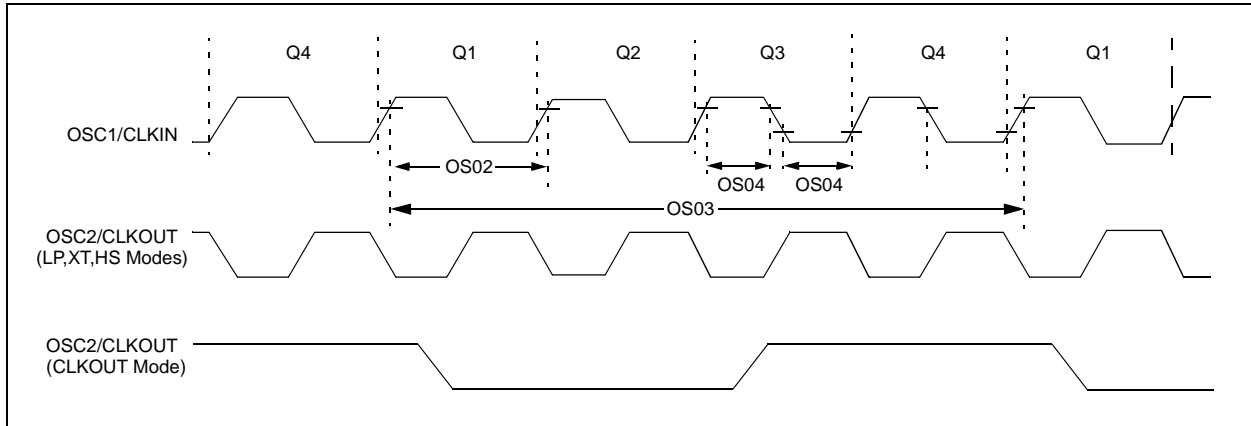
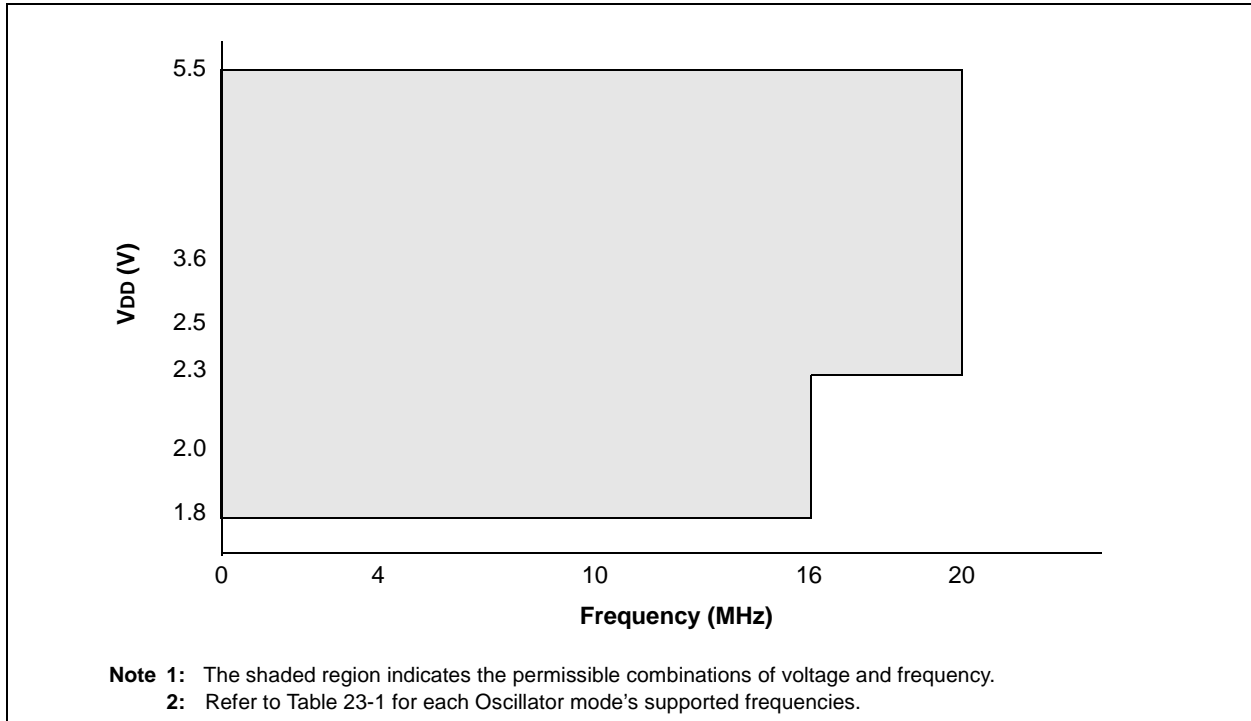


FIGURE 23-4: PIC16F722A/723A VOLTAGE FREQUENCY GRAPH, $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$



PIC16(L)F722A/723A

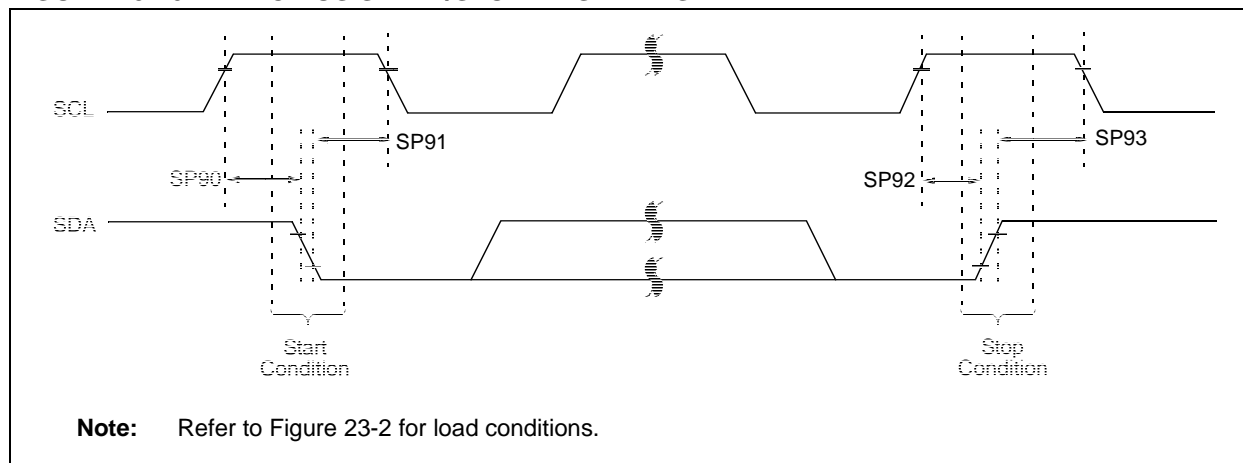
TABLE 23-11: SPI MODE REQUIREMENTS

Param No.	Symbol	Characteristic	Min.	Typ†	Max.	Units	Conditions
SP70*	TssL2sch, TssL2scL	$\overline{SS}\downarrow$ to SCK \downarrow or SCK \uparrow input	Tcy	—	—	ns	
SP71*	Tsch	SCK input high time (Slave mode)	Tcy + 20	—	—	ns	
SP72*	TscL	SCK input low time (Slave mode)	Tcy + 20	—	—	ns	
SP73*	TdIV2sch, TdIV2scL	Setup time of SDI data input to SCK edge	100	—	—	ns	
SP74*	Tsch2diL, TscL2diL	Hold time of SDI data input to SCK edge	100	—	—	ns	
SP75*	TdoR	SDO data output rise time	3.0-5.5V	—	10	25	ns
			1.8-5.5V	—	25	50	ns
SP76*	TdoF	SDO data output fall time	—	10	25	ns	
SP77*	TssH2doZ	$\overline{SS}\uparrow$ to SDO output high-impedance	10	—	50	ns	
SP78*	TscR	SCK output rise time (Master mode)	3.0-5.5V	—	10	25	ns
			1.8-5.5V	—	25	50	ns
SP79*	TscF	SCK output fall time (Master mode)	—	10	25	ns	
SP80*	Tsch2doV, TscL2doV	SDO data output valid after SCK edge	3.0-5.5V	—	—	50	ns
			1.8-5.5V	—	—	145	ns
SP81*	TdoV2sch, TdoV2scL	SDO data output setup to SCK edge	Tcy	—	—	ns	
SP82*	TssL2doV	SDO data output valid after $\overline{SS}\downarrow$ edge	—	—	50	ns	
SP83*	Tsch2ssH, TscL2ssH	$\overline{SS}\uparrow$ after SCK edge	1.5 Tcy + 40	—	—	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.

FIGURE 23-20: I²C BUS START/STOP BITS TIMING



PIC16(L)F722A/723A

TABLE 23-13: I²C BUS DATA REQUIREMENTS

Param. No.	Symbol	Characteristic		Min.	Max.	Units	Conditions
SP100*	THIGH	Clock high time	100 kHz mode	4.0	—	μs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	μs	Device must operate at a minimum of 10 MHz
			SSP Module	1.5T _{CY}	—		
SP101*	TLOW	Clock low time	100 kHz mode	4.7	—	μs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	—	μs	Device must operate at a minimum of 10 MHz
			SSP Module	1.5T _{CY}	—		
SP102*	TR	SDA and SCL rise time	100 kHz mode	—	1000	ns	
			400 kHz mode	20 + 0.1C _B	300	ns	C _B is specified to be from 10-400 pF
SP103*	TF	SDA and SCL fall time	100 kHz mode	—	250	ns	
			400 kHz mode	20 + 0.1C _B	250	ns	C _B is specified to be from 10-400 pF
SP106*	THD:DAT	Data input hold time	100 kHz mode	0	—	ns	
			400 kHz mode	0	0.9	μs	
SP107*	TSU:DAT	Data input setup time	100 kHz mode	250	—	ns	(Note 2)
			400 kHz mode	100	—	ns	
SP109*	TAA	Output valid from clock	100 kHz mode	—	3500	ns	(Note 1)
			400 kHz mode	—	—	ns	
SP110*	TBUF	Bus free time	100 kHz mode	4.7	—	μs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	μs	
SP111	C _B	Bus capacitive loading		—	400	pF	

* These parameters are characterized but not tested.

Note 1: As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of Start or Stop conditions.

2: A Fast mode (400 kHz) I²C bus device can be used in a Standard mode (100 kHz) I²C bus system, but the requirement TSU:DAT ≥ 250 ns must then be met. This will automatically be the case if the device does not stretch the low period of the SCL signal. If such a device does stretch the low period of the SCL signal, it must output the next data bit to the SDA line T_R max. + TSU:DAT = 1000 + 250 = 1250 ns (according to the Standard mode I²C bus specification), before the SCL line is released.

PIC16(L)F722A/723A

FIGURE 24-30: PIC16LF722A/723A TYPICAL BASE IPD vs. VDD

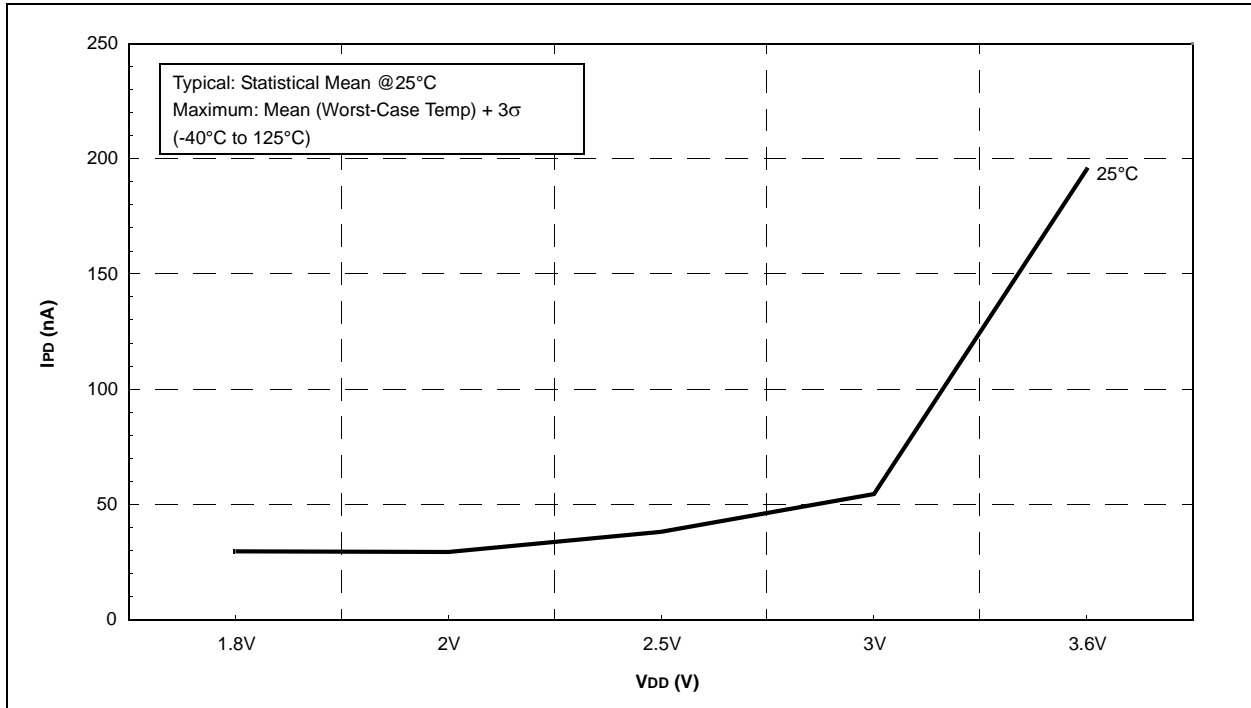
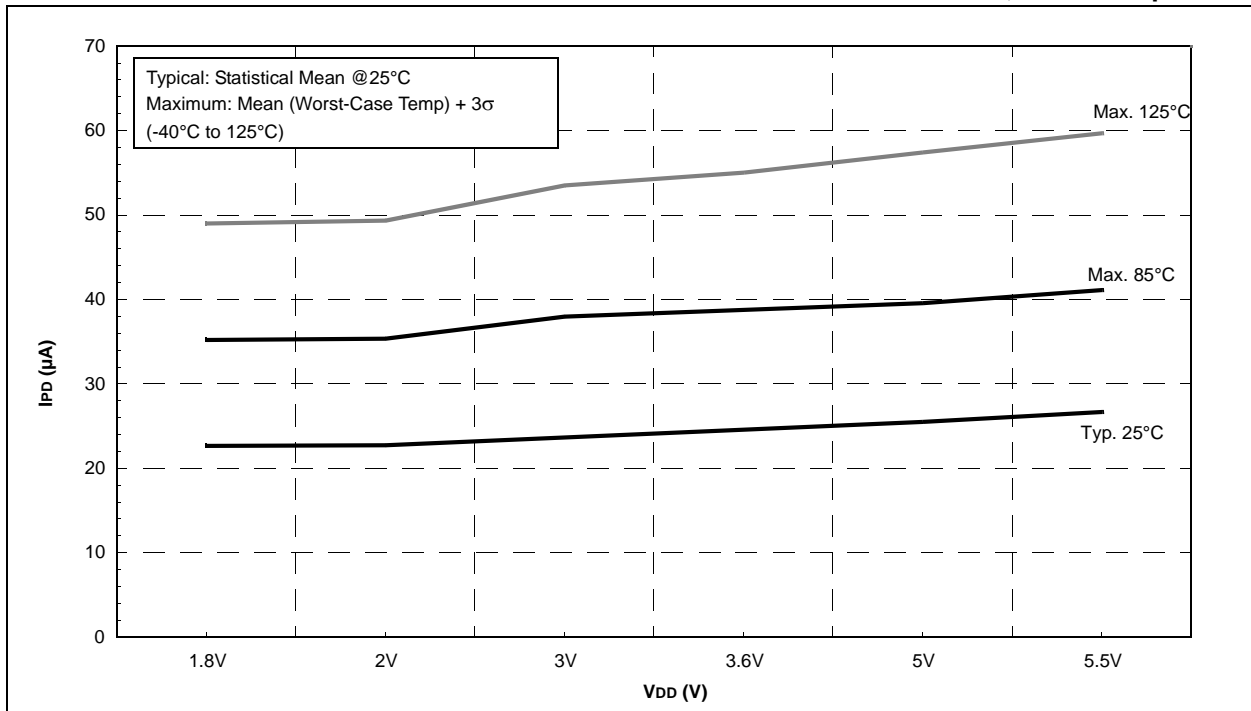


FIGURE 24-31: PIC16F722A/723A FIXED VOLTAGE REFERENCE IPD vs. VDD, VCAP = 0.1μF



PIC16(L)F722A/723A

APPENDIX A: DATA SHEET REVISION HISTORY

Revision A (April 2010)

Original release of this data sheet.

Revision B (January 2012)

Updated the data sheet to new format; Updated Figure 9-1 and Register 9-1; Updated the Packaging Information section; Updated the Product Identification System section; Other minor corrections.

Revision C (03/2016)

Updated Table 2-1, Table 6-1 and Table 6-3; Updated Register 14-2; Other minor corrections.

APPENDIX B: MIGRATING FROM OTHER PIC[®] DEVICES

This discusses some of the issues in migrating from other PIC[®] devices to the PIC16F722A/723A family of devices.

Note: This device has been designed to perform to the parameters of its data sheet. It has been tested to an electrical specification designed to determine its conformance with these parameters. Due to process differences in the manufacture of this device, this device may have different performance characteristics than its earlier version. These differences may cause this device to perform differently in your application than the earlier version of this device.

Note: The user should verify that the device oscillator starts and performs as expected. Adjusting the loading capacitor values and/or the oscillator mode may be required.

B.1 PIC16F77 to PIC16F722A/723A

TABLE B-1: FEATURE COMPARISON

Feature	PIC16F77	PIC16F722A/ 723A
Max. Operating Speed	20 MHz	20 MHz
Max. Program Memory (Words)	8K	4K
Max. SRAM (Bytes)	368	192
A/D Resolution	8-bit	8-bit
Timers (8/16-bit)	2/1	2/1
Oscillator Modes	4	8
Brown-out Reset	Y	Y
Internal Pull ups	RB<7:0>	RB<7:0>
Interrupt-on-change	RB<7:4>	RB<7:0>
Comparator	0	0
USART	Y	Y
Extended WDT	N	N
Software Control Option of WDT/BOR	N	N
INTOSC Frequencies	None	500 kHz - 16 MHz
Clock Switching	N	N