



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.

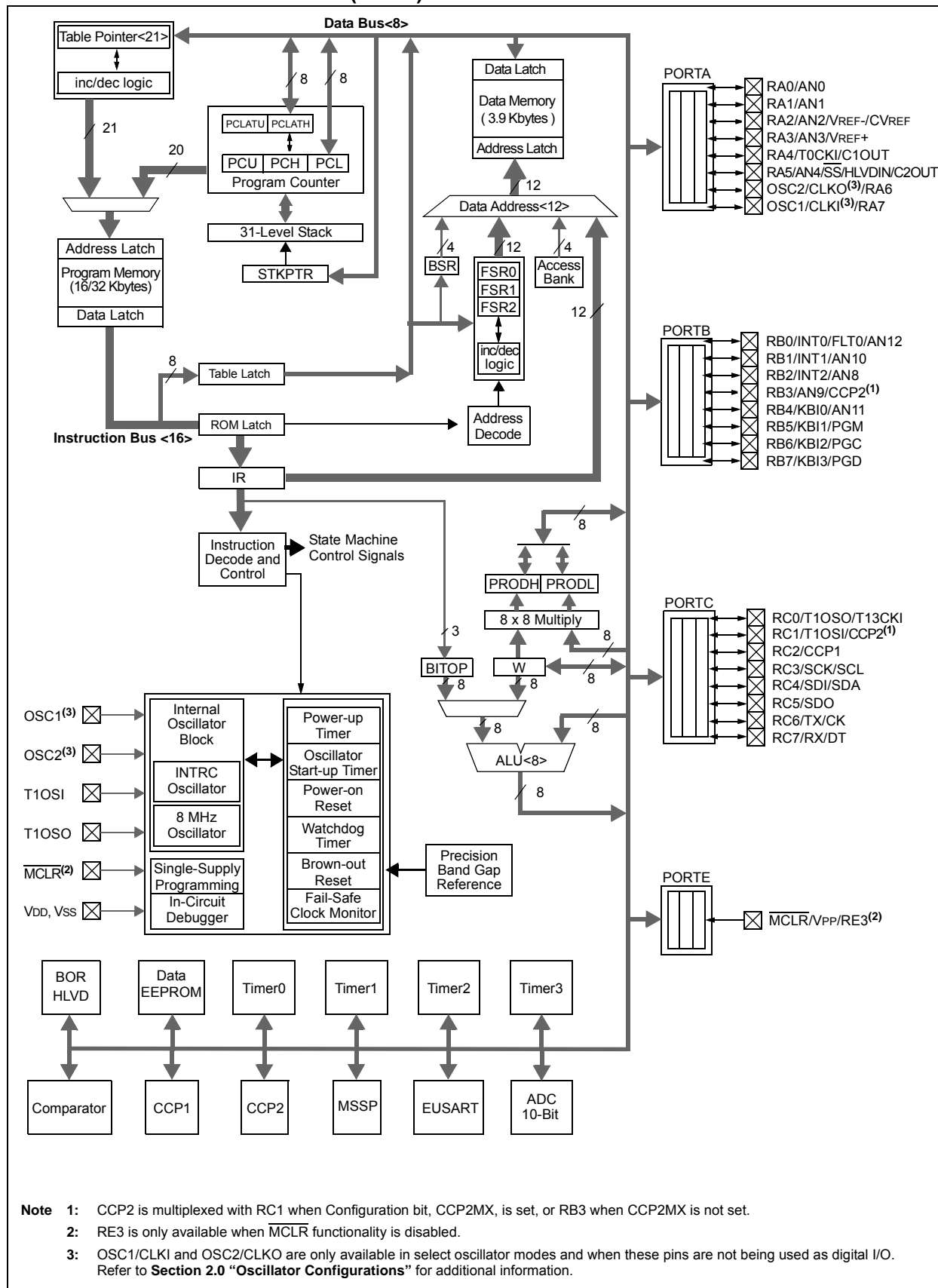
### Applications of "[Embedded - Microcontrollers](#)"

#### Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	40MHz
Connectivity	I <sup>2</sup> C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, HLVD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	32KB (16K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1.5K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 13x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/pic18lf4520-i-ml">https://www.e-xfl.com/product-detail/microchip-technology/pic18lf4520-i-ml</a>

# PIC18F2420/2520/4420/4520

**FIGURE 1-1: PIC18F2420/2520 (28-PIN) BLOCK DIAGRAM**



# PIC18F2420/2520/4420/4520

**TABLE 1-2: PIC18F2420/2520 PINOUT I/O DESCRIPTIONS**

Pin Name	Pin Number		Pin Type	Buffer Type	Description
	SPDIP, SOIC	QFN			
MCLR/VPP/RE3 MCLR  VPP RE3	1	26	I  P I	ST  ST	Master Clear (input) or programming voltage (input). Master Clear (Reset) input. This pin is an active-low Reset to the device. Programming voltage input. Digital input.
OSC1/CLKI/RA7 OSC1  CLKI   RA7	9	6	I  I  I/O	ST  CMOS  TTL	Oscillator crystal or external clock input. Oscillator crystal input or external clock source input. ST buffer when configured in RC mode; CMOS otherwise. External clock source input. Always associated with pin function, OSC1. (See related OSC1/CLKI, OSC2/CLKO pins.) General purpose I/O pin.
OSC2/CLKO/RA6 OSC2  CLKO  RA6	10	7	O  O I/O	—  — TTL	Oscillator crystal or clock output. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. In RC mode, OSC2 pin outputs CLKO which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate. General purpose I/O pin.

**Legend:** TTL = TTL compatible input      CMOS = CMOS compatible input or output  
ST = Schmitt Trigger input with CMOS levels      I = Input  
O = Output      P = Power

**Note 1:** Default assignment for CCP2 when Configuration bit, CCP2MX, is set.  
**Note 2:** Alternate assignment for CCP2 when Configuration bit, CCP2MX, is cleared.

# PIC18F2420/2520/4420/4520

**TABLE 1-2: PIC18F2420/2520 PINOUT I/O DESCRIPTIONS (CONTINUED)**

Pin Name	Pin Number		Pin Type	Buffer Type	Description
	SPDIP, SOIC	QFN			
RB0/INT0/FLT0/AN12	21	18			PORTB is a bidirectional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs.
RB0			I/O	TTL	Digital I/O.
INT0			I	ST	External interrupt 0.
FLT0			I	ST	PWM Fault input for CCP1.
AN12			I	Analog	Analog input 12.
RB1/INT1/AN10	22	19			
RB1			I/O	TTL	Digital I/O.
INT1			I	ST	External interrupt 1.
AN10			I	Analog	Analog input 10.
RB2/INT2/AN8	23	20			
RB2			I/O	TTL	Digital I/O.
INT2			I	ST	External interrupt 2.
AN8			I	Analog	Analog input 8.
RB3/AN9/CCP2	24	21			
RB3			I/O	TTL	Digital I/O.
AN9			I	Analog	Analog input 9.
CCP2 <sup>(1)</sup>			I/O	ST	Capture 2 input/Compare 2 output/PWM2 output.
RB4/KBI0/AN11	25	22			
RB4			I/O	TTL	Digital I/O.
KBI0			I	TTL	Interrupt-on-change pin.
AN11			I	Analog	Analog input 11.
RB5/KBI1/PGM	26	23			
RB5			I/O	TTL	Digital I/O.
KBI1			I	TTL	Interrupt-on-change pin.
PGM			I/O	ST	Low-Voltage ICSP™ Programming enable pin.
RB6/KBI2/PGC	27	24			
RB6			I/O	TTL	Digital I/O.
KBI2			I	TTL	Interrupt-on-change pin.
PGC			I/O	ST	In-Circuit Debugger and ICSP programming clock pin.
RB7/KBI3/PGD	28	25			
RB7			I/O	TTL	Digital I/O.
KBI3			I	TTL	Interrupt-on-change pin.
PGD			I/O	ST	In-Circuit Debugger and ICSP programming data pin.

**Legend:** TTL = TTL compatible input

ST = Schmitt Trigger input with CMOS levels

O = Output

CMOS = CMOS compatible input or output

I = Input

P = Power

**Note 1:** Default assignment for CCP2 when Configuration bit, CCP2MX, is set.

**Note 2:** Alternate assignment for CCP2 when Configuration bit, CCP2MX, is cleared.

# PIC18F2420/2520/4420/4520

---

NOTES:

# PIC18F2420/2520/4420/4520

## REGISTER 9-7:    **PIE2: PERIPHERAL INTERRUPT ENABLE REGISTER 2**

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
OSCFIE	CMIE	—	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE
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       **OSCFIE:** Oscillator Fail Interrupt Enable bit

1 = Enabled

0 = Disabled

bit 6       **CMIE:** Comparator Interrupt Enable bit

1 = Enabled

0 = Disabled

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

bit 4       **EEIE:** Data EEPROM/Flash Write Operation Interrupt Enable bit

1 = Enabled

0 = Disabled

bit 3       **BCLIE:** Bus Collision Interrupt Enable bit

1 = Enabled

0 = Disabled

bit 2       **HLVDIE:** High/Low-Voltage Detect Interrupt Enable bit

1 = Enabled

0 = Disabled

bit 1       **TMR3IE:** TMR3 Overflow Interrupt Enable bit

1 = Enabled

0 = Disabled

bit 0       **CCP2IE:** CCP2 Interrupt Enable bit

1 = Enabled

0 = Disabled

## 12.0 TIMER1 MODULE

The Timer1 timer/counter module incorporates these features:

- Software selectable operation as a 16-bit timer or counter
- Readable and writable 8-bit registers (TMR1H and TMR1L)
- Selectable clock source (internal or external) with device clock or Timer1 oscillator internal options
- Interrupt-on-overflow
- Reset on CCP Special Event Trigger
- Device clock status flag (T1RUN)

A simplified block diagram of the Timer1 module is shown in Figure 12-1. A block diagram of the module's operation in Read/Write mode is shown in Figure 12-2.

The module incorporates its own low-power oscillator to provide an additional clocking option. The Timer1 oscillator can also be used as a low-power clock source for the microcontroller in power-managed operation.

Timer1 can also be used to provide Real-Time Clock (RTC) functionality to applications with only a minimal addition of external components and code overhead.

Timer1 is controlled through the T1CON Control register (Register 12-1). It also contains the Timer1 Oscillator Enable bit (T1OSCEN). Timer1 can be enabled or disabled by setting or clearing control bit, TMR1ON (T1CON<0>).

### REGISTER 12-1: T1CON: TIMER1 CONTROL REGISTER

R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RD16	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON
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 **RD16:** 16-Bit Read/Write Mode Enable bit  
 1 = Enables register read/write of Timer1 in one 16-bit operation  
 0 = Enables register read/write of Timer1 in two 8-bit operations
- bit 6 **T1RUN:** Timer1 System Clock Status bit  
 1 = Device clock is derived from Timer1 oscillator  
 0 = Device clock is derived from another source
- bit 5-4 **T1CKPS<1:0>:** Timer1 Input Clock Prescale Select bits  
 11 = 1:8 Prescale value  
 10 = 1:4 Prescale value  
 01 = 1:2 Prescale value  
 00 = 1:1 Prescale value
- bit 3 **T1OSCEN:** Timer1 Oscillator Enable bit  
 1 = Timer1 oscillator is enabled  
 0 = Timer1 oscillator is shut off  
 The oscillator inverter and feedback resistor are turned off to eliminate power drain.
- bit 2 **T1SYNC:** Timer1 External Clock Input Synchronization Select bit  
When TMR1CS = 1:  
 1 = Do not synchronize external clock input  
 0 = Synchronize external clock input  
When TMR1CS = 0:  
 This bit is ignored. Timer1 uses the internal clock when TMR1CS = 0.
- bit 1 **TMR1CS:** Timer1 Clock Source Select bit  
 1 = External clock from pin RC0/T1OSO/T13CKI (on the rising edge)  
 0 = Internal clock (FOSC/4)
- bit 0 **TMR1ON:** Timer1 On bit  
 1 = Enables Timer1  
 0 = Stops Timer1

# PIC18F2420/2520/4420/4520

## 15.3 Compare Mode

In Compare mode, the 16-bit CCPRx register value is constantly compared against either the TMR1 or TMR3 register pair value. When a match occurs, the CCPx pin can be:

- driven high
- driven low
- toggled (high-to-low or low-to-high)
- remain unchanged (that is, reflects the state of the I/O latch)

The action on the pin is based on the value of the mode select bits (CCPxM<3:0>). At the same time, the interrupt flag bit, CCPxIF, is set.

### 15.3.1 CCP PIN CONFIGURATION

The user must configure the CCPx pin as an output by clearing the appropriate TRIS bit.

**Note:** Clearing the CCP2CON register will force the RB3 or RC1 compare output latch (depending on device configuration) to the default low level. This is not the PORTB or PORTC I/O data latch.

### 15.3.2 TIMER1/TIMER3 MODE SELECTION

Timer1 and/or Timer3 must be running in Timer mode or Synchronized Counter mode if the CCP module is using the compare feature. In Asynchronous Counter mode, the compare operation may not work.

### 15.3.3 SOFTWARE INTERRUPT MODE

When the Generate Software Interrupt mode is chosen (CCPxM<3:0> = 1010), the corresponding CCPx pin is not affected. A CCP interrupt is generated when the CCPxIF interrupt flag is set while the CCPxIE bit is set.

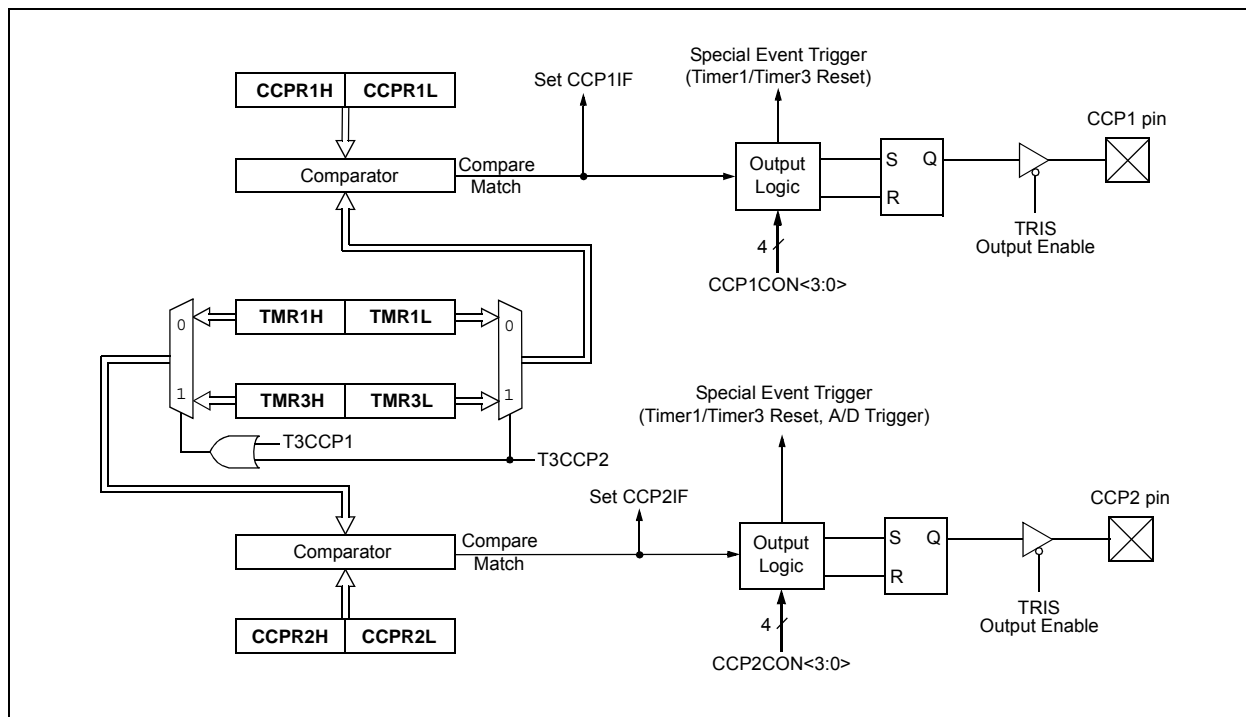
### 15.3.4 SPECIAL EVENT TRIGGER

Both CCP modules are equipped with a Special Event Trigger. This is an internal hardware signal generated in Compare mode to trigger actions by other modules. The Special Event Trigger is enabled by selecting the Compare Special Event Trigger mode (CCPxM<3:0> = 1011).

For either CCP module, the Special Event Trigger resets the Timer register pair for whichever timer resource is currently assigned as the module's time base. This allows the CCPRx registers to serve as a programmable Period register for either timer.

The Special Event Trigger for CCP2 can also start an A/D conversion. In order to do this, the A/D Converter must already be enabled.

FIGURE 15-2: COMPARE MODE OPERATION BLOCK DIAGRAM





## 16.4 Enhanced PWM Mode

The Enhanced PWM mode provides additional PWM output options for a broader range of control applications. The module is a backward compatible version of the standard CCP module and offers up to four outputs, designated P1A through P1D. Users are also able to select the polarity of the signal (either active-high or active-low). The module's output mode and polarity are configured by setting the P1M<1:0> and CCP1M<3:0> bits of the CCP1CON register.

Figure 16-1 shows a simplified block diagram of PWM operation. All control registers are double-buffered and are loaded at the beginning of a new PWM cycle (the period boundary when Timer2 resets) in order to prevent glitches on any of the outputs. The exception is the PWM Dead-Band Delay register, PWM1CON, which is loaded at either the duty cycle boundary or the period boundary (whichever comes first). Because of the buffering, the module waits until the assigned timer resets instead of starting immediately. This means that Enhanced PWM waveforms do not exactly match the standard PWM waveforms, but are instead offset by one full instruction cycle (4 TOSC).

As before, the user must manually configure the appropriate TRIS bits for output.

### 16.4.1 PWM PERIOD

The PWM period is specified by writing to the PR2 register. The PWM period can be calculated using the following equation.

#### EQUATION 16-1:

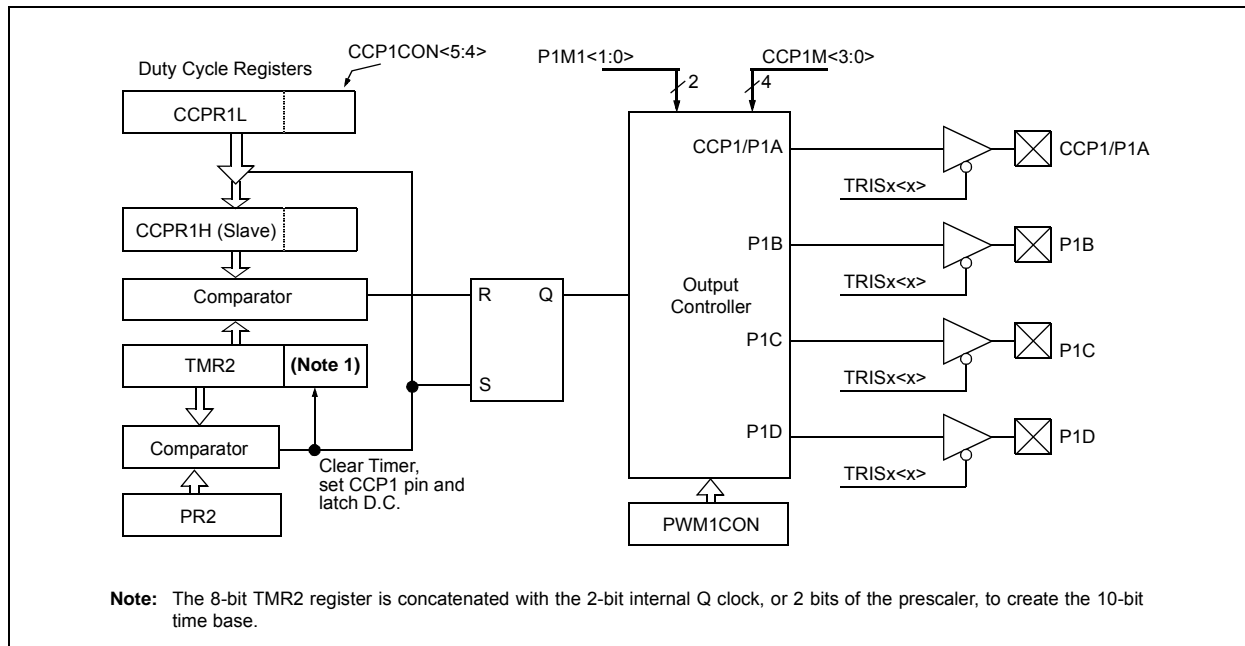
$$\text{PWM Period} = [(PR2) + 1] \cdot 4 \cdot T_{OSC} \cdot (\text{TMR2 Prescale Value})$$

PWM frequency is defined as  $1/[\text{PWM period}]$ . When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The CCP1 pin is set (if PWM duty cycle = 0%, the CCP1 pin will not be set)
- The PWM duty cycle is copied from CCPR1L into CCPR1H

**Note:** The Timer2 postscaler (see **Section 13.0 “Timer2 Module”**) is not used in the determination of the PWM frequency. The postscaler could be used to have a servo update rate at a different frequency than the PWM output.

**FIGURE 16-1: SIMPLIFIED BLOCK DIAGRAM OF THE ENHANCED PWM MODULE**



## 17.4.6 MASTER MODE

Master mode is enabled by setting and clearing the appropriate SSPM bits in SSPCON1 and by setting the SSPEN bit. In Master mode, the SCL and SDA lines are manipulated by the MSSP hardware.

Master mode of operation is supported by interrupt generation on the detection of the Start and Stop conditions. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSP module is disabled. Control of the I<sup>2</sup>C bus may be taken when the P bit is set, or the bus is Idle, with both the S and P bits clear.

In Firmware Controlled Master mode, user code conducts all I<sup>2</sup>C bus operations based on Start and Stop bit conditions.

Once Master mode is enabled, the user has six options.

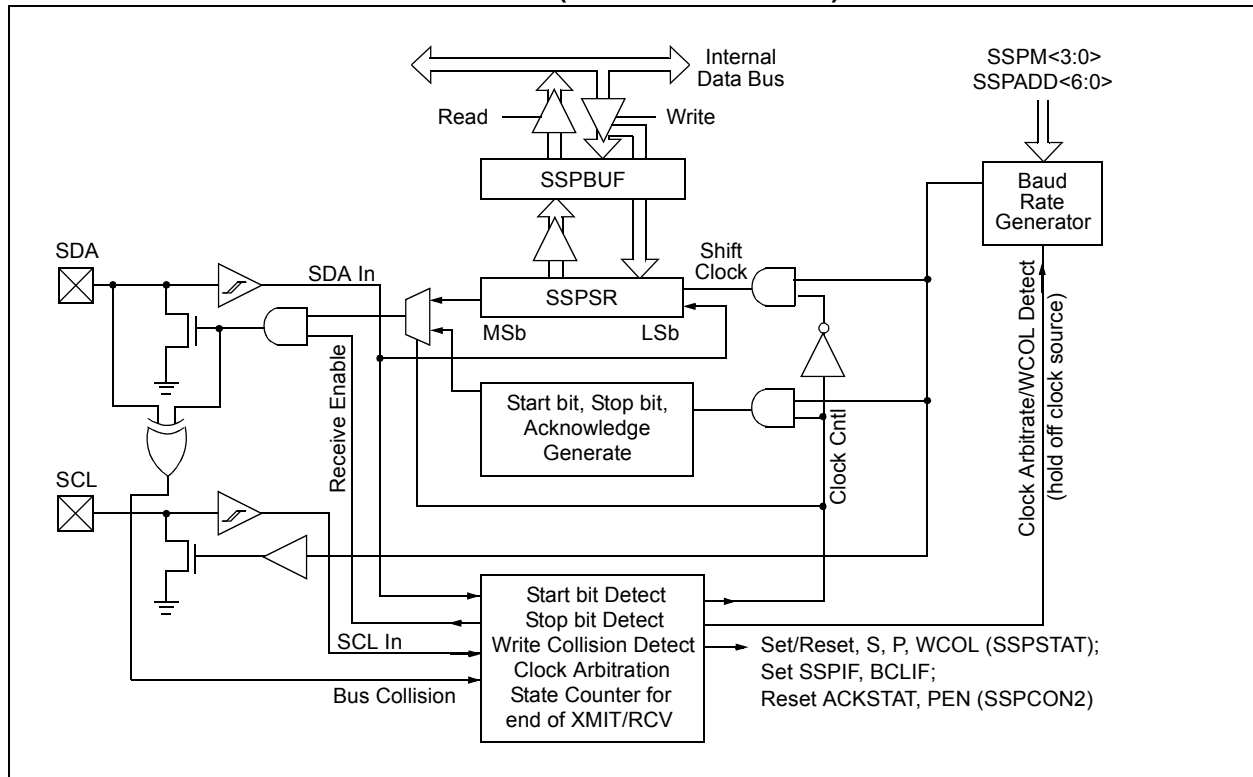
1. Assert a Start condition on SDA and SCL.
2. Assert a Repeated Start condition on SDA and SCL.
3. Write to the SSPBUF register initiating transmission of data/address.
4. Configure the I<sup>2</sup>C port to receive data.
5. Generate an Acknowledge condition at the end of a received byte of data.
6. Generate a Stop condition on SDA and SCL.

**Note:** The MSSP module, when configured in I<sup>2</sup>C Master mode, does not allow queueing of events. For instance, the user is not allowed to initiate a Start condition and immediately write the SSPBUF register to initiate transmission before the Start condition is complete. In this case, the SSPBUF will not be written to and the WCOL bit will be set, indicating that a write to the SSPBUF did not occur.

The following events will cause the MSSP Interrupt Flag bit, SSPIF, to be set (MSSP interrupt, if enabled):

- Start condition
- Stop condition
- Data transfer byte transmitted/received
- Acknowledge transmit
- Repeated Start

**FIGURE 17-16: MSSP BLOCK DIAGRAM (I<sup>2</sup>C™ MASTER MODE)**



# PIC18F2420/2520/4420/4520

## EXAMPLE 18-1: CALCULATING BAUD RATE ERROR

For a device with FOSC of 16 MHz, desired baud rate of 9600, Asynchronous mode, 8-bit BRG:

Desired Baud Rate =  $F_{OSC}/(64 ([SPBRGH:SPBRG] + 1))$

Solving for SPBRGH:SPBRG:

X =  $((F_{OSC}/\text{Desired Baud Rate})/64) - 1$

=  $((16000000/9600)/64) - 1$

=  $[25.042] = 25$

Calculated Baud Rate =  $16000000/(64 (25 + 1))$

= 9615

Error =  $(\text{Calculated Baud Rate} - \text{Desired Baud Rate})/\text{Desired Baud Rate}$

=  $(9615 - 9600)/9600 = 0.16\%$

TABLE 18-2: REGISTERS ASSOCIATED WITH BAUD RATE GENERATOR

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	51
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	51
BAUDCON	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN	51
SPBRGH	EUSART Baud Rate Generator Register High Byte								51
SPBRG	EUSART Baud Rate Generator Register Low Byte								51

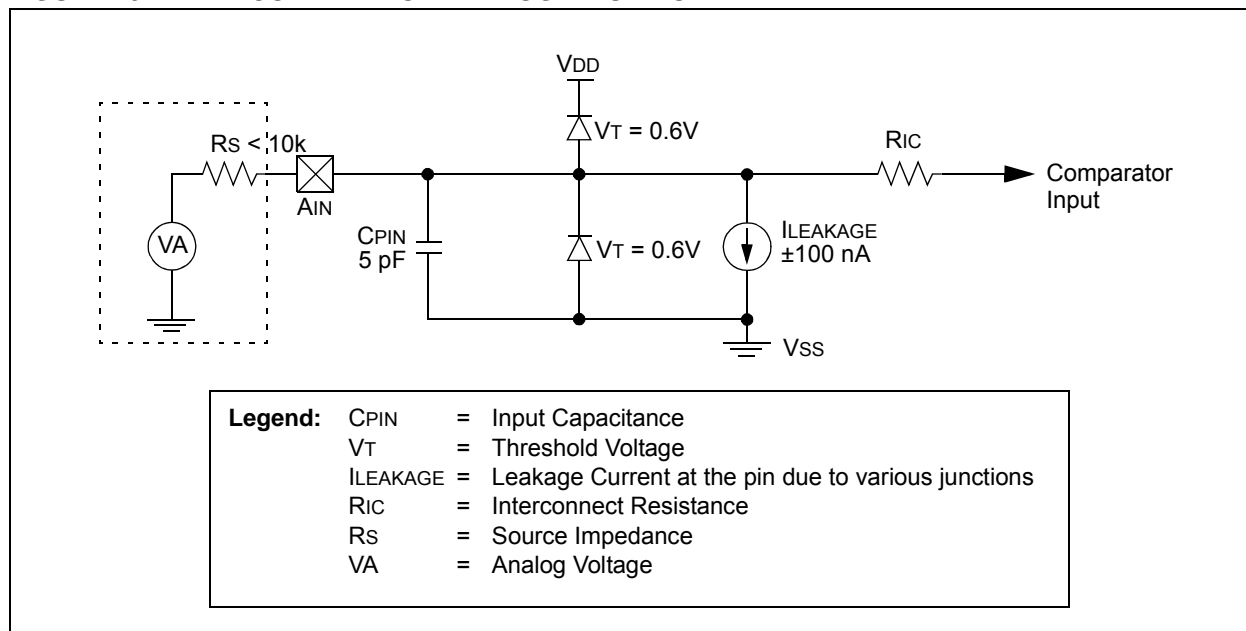
**Legend:** — = unimplemented, read as '0'. Shaded cells are not used by the BRG.

## 20.9 Analog Input Connection Considerations

A simplified circuit for an analog input is shown in Figure 20-4. Since the analog pins are connected to a digital output, they have reverse biased diodes to VDD and VSS. The analog input, therefore, must be between VSS and VDD. If the input voltage deviates from this

range by more than 0.6V in either direction, one of the diodes is forward biased and a latch-up condition may occur. A maximum source impedance of 10 kΩ is recommended for the analog sources. Any external component connected to an analog input pin, such as a capacitor or a Zener diode, should have very little leakage current.

**FIGURE 20-4: COMPARATOR ANALOG INPUT MODEL**



**TABLE 20-1: REGISTERS ASSOCIATED WITH COMPARATOR MODULE**

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	51
CVRCON	CVREN	CVROE	CVRR	CVRSS	CVR3	CVR2	CVR1	CVR0	51
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	52
PIR2	OSCFIF	CMIF	—	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF	52
PIE2	OSCFIE	CMIE	—	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE	52
IPR2	OSCFIP	CMIP	—	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP	52
PORTA	RA7 <sup>(1)</sup>	RA6 <sup>(1)</sup>	RA5	RA4	RA3	RA2	RA1	RA0	52
LATA	LATA7 <sup>(1)</sup>	LATA6 <sup>(1)</sup>	PORTA Data Latch Register (Read and Write to Data Latch)						52
TRISA	TRISA7 <sup>(1)</sup>	TRISA6 <sup>(1)</sup>	PORTA Data Direction Register						52

**Legend:** — = unimplemented, read as '0'. Shaded cells are unused by the comparator module.

**Note 1:** PORTA<7:6> and their direction and latch bits are individually configured as port pins based on various primary oscillator modes. When disabled, these bits are read as '0'.

# PIC18F2420/2520/4420/4520

## REGISTER 23-8: CONFIG6L: CONFIGURATION REGISTER 6 LOW (BYTE ADDRESS 30000Ah)

U-0	U-0	U-0	U-0	R/C-1	R/C-1	R/C-1	R/C-1
—	—	—	—	WRT3 <sup>(1)</sup>	WRT2 <sup>(1)</sup>	WRT1	WRT0
bit 7				bit 0			

### Legend:

R = Readable bit

C = Clearable bit

U = Unimplemented bit, read as '0'

-n = Value when device is unprogrammed

u = Unchanged from programmed state

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

bit 3 **WRT3:** Write Protection bit<sup>(1)</sup>

1 = Block 3 (006000-007FFFh) not write-protected

0 = Block 3 (006000-007FFFh) write-protected

bit 2 **WRT2:** Write Protection bit<sup>(1)</sup>

1 = Block 2 (004000-005FFFh) not write-protected

0 = Block 2 (004000-005FFFh) write-protected

bit 1 **WRT1:** Write Protection bit

1 = Block 1 (002000-003FFFh) not write-protected

0 = Block 1 (002000-003FFFh) write-protected

bit 0 **WRT0:** Write Protection bit

1 = Block 0 (000800-001FFFh) not write-protected

0 = Block 0 (000800-001FFFh) write-protected

**Note 1:** Unimplemented in PIC18F2420/4420 devices; maintain this bit set.

## REGISTER 23-9: CONFIG6H: CONFIGURATION REGISTER 6 HIGH (BYTE ADDRESS 30000Bh)

R/C-1	R/C-1	R/C-1	U-0	U-0	U-0	U-0	U-0
WRTD	WRTB	WRTC <sup>(1)</sup>	—	—	—	—	—
bit 7				bit 0			

### Legend:

R = Readable bit

C = Clearable bit

U = Unimplemented bit, read as '0'

-n = Value when device is unprogrammed

u = Unchanged from programmed state

bit 7 **WRTD:** Data EEPROM Write Protection bit

1 = Data EEPROM not write-protected

0 = Data EEPROM write-protected

bit 6 **WRTB:** Boot Block Write Protection bit

1 = Boot block (000000-0007FFFh) not write-protected

0 = Boot block (000000-0007FFFh) write-protected

bit 5 **WRTC:** Configuration Register Write Protection bit<sup>(1)</sup>

1 = Configuration registers (300000-3000FFFh) not write-protected

0 = Configuration registers (300000-3000FFFh) write-protected

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

**Note 1:** This bit is read-only in normal execution mode; it can be written only in Program mode.

# PIC18F2420/2520/4420/4520

**TABLE 24-2: PIC18FXXXX INSTRUCTION SET (CONTINUED)**

Mnemonic, Operands	Description	Cycles	16-Bit Instruction Word				Status Affected	Notes	
			MSb		LSb				
LITERAL OPERATIONS									
ADDLW k	Add Literal and WREG	1	0000	1111	kkkk	kkkk	C, DC, Z, OV, N		
ANDLW k	AND Literal with WREG	1	0000	1011	kkkk	kkkk	Z, N		
IORLW k	Inclusive OR Literal with WREG	1	0000	1001	kkkk	kkkk	Z, N		
LFSR f, k	Move Literal (12-bit)2nd word to FSR(f) 1st word	2	1110	1110	00ff	kkkk	None		
			1111	0000	kkkk	kkkk			
MOVLB k	Move Literal to BSR<3:0>	1	0000	0001	0000	kkkk	None		
MOVLW k	Move Literal to WREG	1	0000	1110	kkkk	kkkk	None		
MULLW k	Multiply Literal with WREG	1	0000	1101	kkkk	kkkk	None		
RETLW k	Return with Literal in WREG	2	0000	1100	kkkk	kkkk	None		
SUBLW k	Subtract WREG from Literal	1	0000	1000	kkkk	kkkk	C, DC, Z, OV, N		
XORLW k	Exclusive OR Literal with WREG	1	0000	1010	kkkk	kkkk	Z, N		
DATA MEMORY ↔ PROGRAM MEMORY OPERATIONS									
TBLRD*	Table Read	2	0000	0000	0000	1000	None		
TBLRD*+	Table Read with Post-Increment		0000	0000	0000	1001	None		
TBLRD*-	Table Read with Post-Decrement		0000	0000	0000	1010	None		
TBLRD+*	Table Read with Pre-Increment		0000	0000	0000	1011	None		
TBLWT*	Table Write	2	0000	0000	0000	1100	None		
TBLWT*+	Table Write with Post-Increment		0000	0000	0000	1101	None		
TBLWT*-	Table Write with Post-Decrement		0000	0000	0000	1110	None		
TBLWT+*	Table Write with Pre-Increment		0000	0000	0000	1111	None		

- Note 1:** When a PORT register is modified as a function of itself (e.g., `MOVF PORTB, 1, 0`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.
- 2:** If this instruction is executed on the TMR0 register (and where applicable, 'd' = 1), the prescaler will be cleared if assigned.
- 3:** If the Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.
- 4:** Some instructions are two-word instructions. The second word of these instructions will be executed as a NOP unless the first word of the instruction retrieves the information embedded in these 16 bits. This ensures that all program memory locations have a valid instruction.

# PIC18F2420/2520/4420/4520

## ANDWF

## AND W with f

Syntax:	ANDWF f {,d {,a}}			
Operands:	$0 \leq f \leq 255$ $d \in [0,1]$ $a \in [0,1]$			
Operation:	(W) .AND. (f) $\rightarrow$ dest			
Status Affected:	N, Z			
Encoding:	0001	01da	ffff	ffff
Description:	<p>The contents of W are ANDed with register 'f'. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f' (default).</p> <p>If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank (default).</p> <p>If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever <math>f \leq 95</math> (5Fh). See <b>Section 24.2.3 “Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode”</b> for details.</p>			
Words:	1			
Cycles:	1			
Q Cycle Activity:				

**Example:** ANDWF REG, 0, 0

Before Instruction

W = 17h  
REG = C2h

After Instruction

W = 02h  
REG = C2h

## BC

## Branch if Carry

Syntax:	BC    n				
Operands:	$-128 \leq n \leq 127$				
Operation:	if Carry bit is '1', (PC) + 2 + 2n → PC				
Status Affected:	None				
Encoding:	<table border="1"><tr><td>1110</td><td>0010</td><td>nnnn</td><td>nnnn</td></tr></table>	1110	0010	nnnn	nnnn
1110	0010	nnnn	nnnn		
Description:	<p>If the Carry bit is '1', then the program will branch.</p> <p>The 2's complement number '2n' is added to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is then a two-cycle instruction.</p>				
Words:	1				
Cycles:	1(2)				
Q Cycle Activity:					
If Jump:					

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

If No Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	No operation

**Example:** HERE BC 5

Before Instruction

PC = address (HERE)

After Instruction

If Carry = 1;  
PC = address (HERE + 12)  
If Carry = 0;  
PC = address (HERE + 2)

# PIC18F2420/2520/4420/4520

## BTG

## Bit Toggle f

Syntax: BTG f, b {,a}

Operands:  $0 \leq f \leq 255$

$0 \leq b < 7$

$a \in [0,1]$

Operation:  $\overline{(f \ll b)} \rightarrow f \ll b$

Status Affected: None

Encoding: 

0111	bbba	ffff	ffff
------	------	------	------

Description: Bit 'b' in data memory location 'f' is inverted.  
If 'a' is '0', the Access Bank is selected.  
If 'a' is '1', the BSR is used to select the GPR bank (default).  
If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever  $f \leq 95$  (5Fh). See **Section 24.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write register 'f'

**Example:** BTG PORTC, 4, 0

Before Instruction:

PORTC = 0111 0101 [75h]

After Instruction:

PORTC = 0110 0101 [65h]

## BOV

## Branch if Overflow

Syntax: BOV n

Operands:  $-128 \leq n \leq 127$

Operation: if Overflow bit is '1',  
 $(PC) + 2 + 2n \rightarrow PC$

Status Affected: None

Encoding: 

1110	0100	nnnn	nnnn
------	------	------	------

Description: If the Overflow bit is '1', then the program will branch.  
The 2's complement number, '2n', is added to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be  $PC + 2 + 2n$ . This instruction is then a two-cycle instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:

If Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

If No Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	No operation

**Example:** HERE BOV Jump

Before Instruction

PC = address (HERE)

After Instruction

If Overflow = 1;

PC = address (Jump)

If Overflow = 0;

PC = address (HERE + 2)



# PIC18F2420/2520/4420/4520

COMF		Complement f							
Syntax:	COMF    f {,d {,a}}								
Operands:	$0 \leq f \leq 255$ $d \in [0,1]$ $a \in [0,1]$								
Operation:	$(\bar{f}) \rightarrow \text{dest}$								
Status Affected:	N, Z								
Encoding:	<table border="1"><tr><td>0001</td><td>11da</td><td>ffff</td><td>ffff</td></tr></table>					0001	11da	ffff	ffff
0001	11da	ffff	ffff						
Description:	<p>The contents of register 'f' are complemented. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f' (default). If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank (default). If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever <math>f \leq 95</math> (5Fh). See <b>Section 24.2.3 “Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode”</b> for details.</p>								
Words:	1								
Cycles:	1								
Q Cycle Activity:									

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination

**Example:** COMF REG, 0, 0

Before Instruction  
 REG = 13h  
 After Instruction  
 REG = 13h  
 W = ECh

CPFSEQ		Compare f with W, Skip if f = W							
Syntax:	CPFSEQ f {,a}								
Operands:	$0 \leq f \leq 255$ $a \in [0,1]$								
Operation:	$(f) - (W)$ , skip if $(f) = (W)$ (unsigned comparison)								
Status Affected:	None								
Encoding:	<table border="1"><tr><td>0110</td><td>001a</td><td>ffff</td><td>ffff</td></tr></table>					0110	001a	ffff	ffff
0110	001a	ffff	ffff						
Description:	<p>Compares the contents of data memory location 'f' to the contents of W by performing an unsigned subtraction. If 'f' = W, then the fetched instruction is discarded and a NOP is executed instead, making this a two-cycle instruction.</p> <p>If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank (default).</p> <p>If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever <math>f \leq 95</math> (5Fh). See <b>Section 24.2.3 “Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode”</b> for details.</p>								
Words:	1								
Cycles:	1(2)								

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	No operation

If skip:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation

If skip and followed by 2-word instruction:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation
No operation	No operation	No operation	No operation

**Example:** HERE CPFSEQ REG, 0  
 NEQUAL :  
 EQUAL :

Before Instruction

PC Address = HERE  
 W = ?  
 REG = ?

After Instruction

If REG = W;  
 PC = Address (EQUAL)  
 If REG  $\neq$  W;  
 PC = Address (NEQUAL)

# PIC18F2420/2520/4420/4520

DECFSZ		Decrement f, Skip if 0							
Syntax:	DECFSZ f {,d {,a}}								
Operands:	$0 \leq f \leq 255$ $d \in [0,1]$ $a \in [0,1]$								
Operation:	$(f) - 1 \rightarrow \text{dest}$ , skip if result = 0								
Status Affected:	None								
Encoding:	<table border="1"><tr><td>0010</td><td>11da</td><td>ffff</td><td>ffff</td></tr></table>					0010	11da	ffff	ffff
0010	11da	ffff	ffff						
Description:	<p>The contents of register 'f' are decremented. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed back in register 'f' (default). If the result is '0', the next instruction, which is already fetched, is discarded and a NOP is executed instead, making it a two-cycle instruction.</p> <p>If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank (default).</p> <p>If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever <math>f \leq 95</math> (5Fh). See <b>Section 24.2.3 “Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode”</b> for details.</p>								
Words:	1								
Cycles:	1(2)								
	<b>Note:</b> 3 cycles if skip and followed by a 2-word instruction.								

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination

If skip:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation

If skip and followed by 2-word instruction:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation
No operation	No operation	No operation	No operation

**Example:**

```

HERE      DECFSZ  CNT, 1, 1
          GOTO    LOOP
          CONTINUE
  
```

Before Instruction

PC = Address (HERE)

After Instruction

CNT = CNT - 1

If CNT = 0;

PC = Address (CONTINUE)

If CNT  $\neq$  0;

PC = Address (HERE + 2)

DCFSNZ	Decrement f, Skip if Not 0				
Syntax:	DCFSNZ f {,d {,a}}				
Operands:	$0 \leq f \leq 255$ $d \in [0,1]$ $a \in [0,1]$				
Operation:	$(f) - 1 \rightarrow \text{dest}$ , skip if result $\neq 0$				
Status Affected:	None				
Encoding:	<table><tr><td>0100</td><td>11da</td><td>ffff</td><td>ffff</td></tr></table>	0100	11da	ffff	ffff
0100	11da	ffff	ffff		
Description:	<p>The contents of register 'f' are decremented. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed back in register 'f' (default). If the result is not '0', the next instruction, which is already fetched, is discarded and a NOP is executed instead, making it a two-cycle instruction.</p> <p>If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank (default).</p> <p>If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever <math>f \leq 95</math> (5Fh). See <b>Section 24.2.3 “Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode”</b> for details.</p>				
Words:	1				
Cycles:	1(2) <b>Note:</b> 3 cycles if skip and followed by a 2-word instruction.				

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination

If skip:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation

If skip and followed by 2-word instruction:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation
No operation	No operation	No operation	No operation

**Example:**

```

HERE      DCFSNZ  TEMP, 1, 0
ZERO      :
NZERO     :
  
```

Before Instruction

TEMP = ?

After Instruction

TEMP = TEMP - 1,

If TEMP = 0;

PC = Address (ZERO)

If TEMP  $\neq$  0;

PC = Address (NZERO)

# PIC18F2420/2520/4420/4520

## 26.2 DC Characteristics: Power-Down and Supply Current PIC18F2420/2520/4420/4520 (Industrial) PIC18LF2420/2520/4420/4520 (Industrial) (Continued)

PIC18LF2420/2520/4420/4520 (Industrial)		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial					
PIC18F2420/2520/4420/4520 (Industrial, Extended)		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended					
Param No.	Device	Typ	Max	Units	Conditions		
	Supply Current (IDD) <sup>(2)</sup>						
	PIC18LF2X2X/4X20	0.8	1.1	mA	-40°C	VDD = 2.0V	Fosc = 4 MHz (RC_RUN mode, INTOSC source)
		0.8	1.1	mA	+25°C		
		0.8	1.1	mA	+85°C		
	PIC18LF2X2X/4X20	1.3	1.7	mA	-40°C	VDD = 3.0V	
		1.3	1.7	mA	+25°C		
		1.3	1.7	mA	+85°C		
	All devices	2.5	3.5	mA	-40°C	VDD = 5.0V	
		2.5	3.5	mA	+25°C		
		2.5	3.5	mA	+85°C		
	Extended devices only	2.5	3.5	mA	+125°C		
	PIC18LF2X2X/4X20	2.9	5	μA	-40°C	VDD = 2.0V	Fosc = 31 kHz (RC_IDLE mode, INTRC source)
		3.1	5	μA	+25°C		
		3.6	9.5	μA	+85°C		
	PIC18LF2X2X/4X20	4.5	8	μA	-40°C	VDD = 3.0V	
		4.8	8	μA	+25°C		
		5.8	15	μA	+85°C		
	All devices	9.2	16	μA	-40°C	VDD = 5.0V	
		9.8	16	μA	+25°C		
		11.0	35	μA	+85°C		
Extended devices only	21	160	μA	+125°C			

**Legend:** Shading of rows is to assist in readability of the table.

**Note 1:** The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD or VSS and all features that add delta current disabled (such as WDT, Timer1 Oscillator, BOR, etc.).

**2:** The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD or VSS;

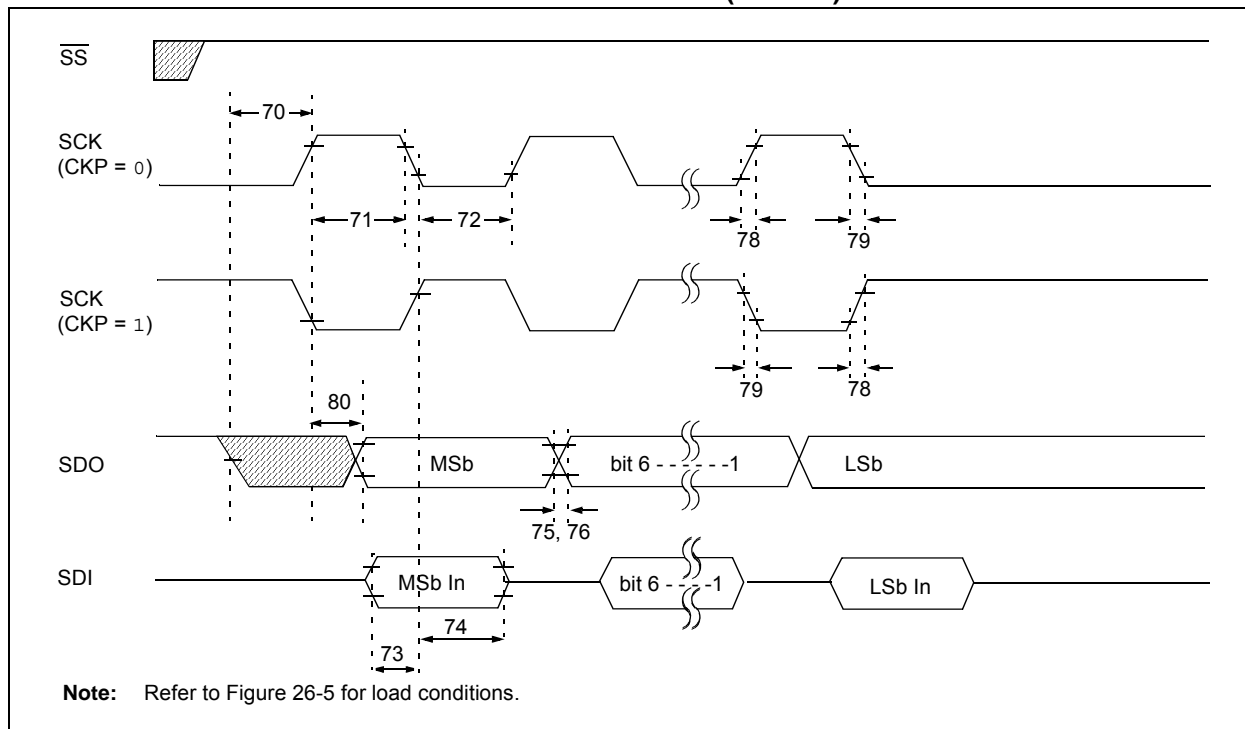
MCLR = VDD; WDT enabled/disabled as specified.

**3:** When operation below -10°C is expected, use T1OSC High-Power mode, where LPT1OSC (CONFIG3H<2>) = 0. When operation will always be above -10°C, then the low-power Timer1 oscillator may be selected.

**4:** BOR and HLVD enable internal band gap reference. With both modules enabled, current consumption will be less than the sum of both specifications.

# PIC18F2420/2520/4420/4520

**FIGURE 26-13: EXAMPLE SPI MASTER MODE TIMING (CKE = 0)**



**TABLE 26-14: EXAMPLE SPI MODE REQUIREMENTS (MASTER MODE, CKE = 0)**

Param No.	Symbol	Characteristic	Min	Max	Units	Conditions
70	TssL2sch, TssL2scL	$\overline{SS} \downarrow$ to SCK $\downarrow$ or SCK $\uparrow$ Input	Tcy	—	ns	
73	TdiV2sch, TdiV2scL	Setup Time of SDI Data Input to SCK Edge	20	—	ns	
73A	Tb2b	Last Clock Edge of Byte 1 to the 1st Clock Edge of Byte 2	1.5 Tcy + 40	—	ns	(Note 2)
74	Tsch2diL, TscL2diL	Hold Time of SDI Data Input to SCK Edge	40	—	ns	
75	TdoR	SDO Data Output Rise Time	PIC18FXXXX —	25	ns	VDD = 2.0V
76	TdoF	SDO Data Output Fall Time	PIC18LFXXXX —	45	ns	
78	TscR	SCK Output Rise Time (Master mode)	PIC18FXXXX —	25	ns	VDD = 2.0V
79	TscF	SCK Output Fall Time (Master mode)	PIC18LFXXXX —	45	ns	
80	Tsch2doV, TscL2doV	SDO Data Output Valid after SCK Edge	PIC18FXXXX —	50	ns	VDD = 2.0V
			PIC18LFXXXX —	100	ns	

**Note 1:** Requires the use of Parameter #73A.

**2:** Only if Parameter #71A and #72A are used.

# PIC18F2420/2520/4420/4520

## C

C Compilers	
MPLAB C18	318
MPLAB C30	318
CALL	282
CALLW	311
Capture (CCP Module)	141
Associated Registers	143
CCP Pin Configuration	141
CCPRxH:CCPRxL Registers	141
Prescaler	141
Software Interrupt	141
Timer1/Timer3 Mode Selection	141
Capture (ECCP Module)	148
Capture/Compare/PWM (CCP)	139
Capture Mode. See Capture.	
CCP Mode and Timer Resources	140
CCPRxH Register	140
CCPRxL Register	140
Compare Mode. See Compare.	
Interaction of Two CCP Modules	140
Module Configuration	140
Clock Sources	28
Selecting the 31 kHz Source	29
Selection Using OSCCON Register	29
CLRF	283
CLRWDWT	283
Code Examples	
16 x 16 Signed Multiply Routine	90
16 x 16 Unsigned Multiply Routine	90
8 x 8 Signed Multiply Routine	89
8 x 8 Unsigned Multiply Routine	89
Changing Between Capture Prescalers	141
Computed GOTO Using an Offset Value	56
Data EEPROM Read	85
Data EEPROM Refresh Routine	86
Data EEPROM Write	85
Erasing a Flash Program Memory Row	78
Fast Register Stack	56
How to Clear RAM (Bank 1) Using Indirect Addressing	68
Implementing a Real-Time Clock Using a Timer1 Interrupt Service	131
Initializing PORTA	105
Initializing PORTB	108
Initializing PORTC	111
Initializing PORTD	114
Initializing PORTE	117
Loading the SSPBUF (SSPSR) Register	164
Reading a Flash Program Memory Word	77
Saving STATUS, WREG and BSR Registers in RAM	103
Writing to Flash Program Memory	80–81
Code Protection	249
COMF	284
Comparator	233
Analog Input Connection Considerations	237
Associated Registers	237
Configuration	234
Effects of a Reset	236
Interrupts	236
Operation	235
Operation During Sleep	236
Outputs	235
Reference	235

External Signal	235
Internal Signal	235
Response Time	235
Comparator Specifications	338
Comparator Voltage Reference	239
Accuracy and Error	240
Associated Registers	241
Configuring	239
Connection Considerations	240
Effects of a Reset	240
Operation During Sleep	240
Compare (CCP Module)	142
Associated Registers	143
CCPRx Register	142
Pin Configuration	142
Software Interrupt	142
Special Event Trigger	137, 142, 232
Timer1/Timer3 Mode Selection	142
Compare (ECCP Module)	148
Special Event Trigger	148
Computed GOTO	56
Configuration Bits	249
Configuration Register Protection	266
Context Saving During Interrupts	103
CPFSEQ	284
CPFSGT	285
CPFSLT	285
Crystal Oscillator/Ceramic Resonator	23
Customer Change Notification Service	407
Customer Notification Service	407
Customer Support	407

## D

Data Addressing Modes	68
Comparing Addressing Modes with the Extended Instruction Set Enabled	71
Direct	68
Indexed Literal Offset	70
Instructions Affected	70
Indirect	68
Inherent and Literal	68
Data EEPROM	
Code Protection	266
Data EEPROM Memory	83
Associated Registers	87
EEADR Register	83
EECON1 and EECON2 Registers	83
Operation During Code-Protect	86
Protection Against Spurious Write	86
Reading	85
Using	86
Write Verify	85
Writing	85
Data Memory	59
Access Bank	62
and the Extended Instruction Set	70
Bank Select Register (BSR)	59
General Purpose Registers	62
Map for PIC18F2420/4420	60
Map for PIC18F2520/4520	61
Special Function Registers	63
DAW	286
DC and AC Characteristics	
Graphs and Tables	361