



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	25
Program Memory Size	16KB (8K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	768 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 10x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/pic18lf2420-i-so">https://www.e-xfl.com/product-detail/microchip-technology/pic18lf2420-i-so</a>

## TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at [docerrors@microchip.com](mailto:docerrors@microchip.com) or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. We welcome your feedback.

### Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

<http://www.microchip.com>

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

### Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; <http://www.microchip.com>
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include literature number) you are using.

### Customer Notification System

Register on our web site at [www.microchip.com](http://www.microchip.com) to receive the most current information on all of our products.

## REGISTER 2-1: OSCTUNE: OSCILLATOR TUNING REGISTER

**Legend:**

- |         |  |
|---------|--|
| bit 7   | <b>INTSRC:</b> Internal Oscillator Low-Frequency Source Select bit<br>1 = 31.25 kHz device clock derived from 8 MHz INTOSC source (divide-by-256 enabled)<br>0 = 31 kHz device clock derived directly from INTRC internal oscillator   |
| bit 6   | <b>Pllen:</b> Frequency Multiplier PLL for INTOSC Enable bit <sup>(1)</sup><br>1 = PLL enabled for INTOSC (4 MHz and 8 MHz only)<br>0 = PLL disabled   |
| bit 5   | <b>Unimplemented:</b> Read as ‘0’  |
| bit 4-0 | <b>TUN&lt;4:0&gt;:</b> Frequency Tuning bits<br><br>011111 = Maximum frequency<br><div style="text-align: center;">•                 •</div> <div style="text-align: center;">•                 •</div><br>000001<br>000000 = Center frequency. Oscillator module is running at the calibrated frequency.<br>111111<br><div style="text-align: center;">•                 •</div> <div style="text-align: center;">•                 •</div><br>100000 = Minimum frequency |

#### 2.6.5.1 Compensating with the EUSART

An adjustment may be required when the EUSART begins to generate framing errors or receives data with errors while in Asynchronous mode. Framing errors indicate that the device clock frequency is too high. To adjust for this, decrement the value in OSCTUNE to reduce the clock frequency. On the other hand, errors in data may suggest that the clock speed is too low. To compensate, increment OSCTUNE to increase the clock frequency.

This technique compares device clock speed to some reference clock. Two timers may be used; one timer is clocked by the peripheral clock, while the other is clocked by a fixed reference source, such as the Timer1 oscillator.

Both timers are cleared, but the timer clocked by the reference generates interrupts. When an interrupt occurs, the internally clocked timer is read and both timers are cleared. If the internally clocked timer value is greater than expected, then the internal oscillator block is running too fast. To adjust for this, decrement the OSCTUNE register.

### 2.6.5.3 Compensating with the CCP Module in Capture Mode

If the measured time is much greater than the calculated time, the internal oscillator block is running too fast; to compensate, decrement the OSC\_TUNE register. If the measured time is much less than the calculated time, the internal oscillator block is running too slow; to compensate, increment the OSC\_TUNE register.

## 3.4.3 RC\_IDLE MODE

In RC\_IDLE mode, the CPU is disabled but the peripherals continue to be clocked from the internal oscillator block using the INTOSC multiplexer. This mode allows for controllable power conservation during Idle periods.

From RC\_RUN, this mode is entered by setting the IDLEN bit and executing a *SLEEP* instruction. If the device is in another Run mode, first set IDLEN, then set the SCS1 bit and execute *SLEEP*. Although its value is ignored, it is recommended that SCS0 also be cleared; this is to maintain software compatibility with future devices. The INTOSC multiplexer may be used to select a higher clock frequency by modifying the IRCF bits before executing the *SLEEP* instruction. When the clock source is switched to the INTOSC multiplexer, the primary oscillator is shut down and the OSTS bit is cleared.

If the IRCF bits are set to any non-zero value, or the INTSRC bit is set, the INTOSC output is enabled. The IOFS bit becomes set, after the INTOSC output becomes stable, after an interval of TIOBST (parameter 39, Table 26-10). Clocks to the peripherals continue while the INTOSC source stabilizes. If the IRCF bits were previously at a non-zero value, or INTSRC was set before the *SLEEP* instruction was executed and the INTOSC source was already stable, the IOFS bit will remain set. If the IRCF bits and INTSRC are all clear, the INTOSC output will not be enabled, the IOFS bit will remain clear and there will be no indication of the current clock source.

When a wake event occurs, the peripherals continue to be clocked from the INTOSC multiplexer. After a delay of TcSD following the wake event, the CPU begins executing code being clocked by the INTOSC multiplexer. The IDLEN and SCS bits are not affected by the wake-up. The INTRC source will continue to run if either the WDT or the Fail-Safe Clock Monitor is enabled.

## 3.5 Exiting Idle and Sleep Modes

An exit from Sleep mode or any of the Idle modes is triggered by an interrupt, a Reset or a WDT time-out. This section discusses the triggers that cause exits from power-managed modes. The clocking subsystem actions are discussed in each of the power-managed modes (see **Section 3.2 “Run Modes”**, **Section 3.3 “Sleep Mode”** and **Section 3.4 “Idle Modes”**).

### 3.5.1 EXIT BY INTERRUPT

Any of the available interrupt sources can cause the device to exit from an Idle mode or the Sleep mode to a Run mode. To enable this functionality, an interrupt source must be enabled by setting its enable bit in one of the INTCON or PIE registers. The exit sequence is initiated when the corresponding interrupt flag bit is set.

On all exits from Idle or Sleep modes by interrupt, code execution branches to the interrupt vector if the GIE/GIEH bit (INTCON<7>) is set. Otherwise, code execution continues or resumes without branching (see **Section 9.0 “Interrupts”**).

A fixed delay of interval TcSD following the wake event is required when leaving Sleep and Idle modes. This delay is required for the CPU to prepare for execution. Instruction execution resumes on the first clock cycle following this delay.

### 3.5.2 EXIT BY WDT TIME-OUT

A WDT time-out will cause different actions depending on which power-managed mode the device is in when the time-out occurs.

If the device is not executing code (all Idle modes and Sleep mode), the time-out will result in an exit from the power-managed mode (see **Section 3.2 “Run Modes”** and **Section 3.3 “Sleep Mode”**). If the device is executing code (all Run modes), the time-out will result in a WDT Reset (see **Section 23.2 “Watchdog Timer (WDT)”**).

The WDT timer and postscaler are cleared by executing a *SLEEP* or *CLRWDT* instruction, the loss of a currently selected clock source (if the Fail-Safe Clock Monitor is enabled) and modifying the IRCF bits in the OSCCON register if the internal oscillator block is the device clock source.

### 3.5.3 EXIT BY RESET

Normally, the device is held in Reset by the Oscillator Start-up Timer (OST) until the primary clock becomes ready. At that time, the OSTS bit is set and the device begins executing code. If the internal oscillator block is the new clock source, the IOFS bit is set instead.

The exit delay time from Reset to the start of code execution depends on both the clock sources before and after the wake-up and the type of oscillator if the new clock source is the primary clock. Exit delays are summarized in Table 3-2.

Code execution can begin before the primary clock becomes ready. If either the Two-Speed Start-up (see **Section 23.3 “Two-Speed Start-up”**) or Fail-Safe Clock Monitor (see **Section 23.4 “Fail-Safe Clock Monitor”**) is enabled, the device may begin execution as soon as the Reset source has cleared. Execution is clocked by the INTOSC multiplexer driven by the internal oscillator block. Execution is clocked by the internal oscillator block until either the primary clock becomes ready or a power-managed mode is entered before the primary clock becomes ready; the primary clock is then shut down.

# PIC18F2420/2520/4420/4520

## EXAMPLE 6-3: WRITING TO FLASH PROGRAM MEMORY (CONTINUED)

PROGRAM_MEMORY		BSF	EECON1, EEPGD	; point to Flash program memory
		BCF	EECON1, CFGS	; access Flash program memory
		BSF	EECON1, WREN	; enable write to memory
		BCF	INTCON, GIE	; disable interrupts
Required Sequence		MOVLW	55h	
		MOVWF	EECON2	; write 55h
		MOVLW	0AAh	
		MOVWF	EECON2	; write 0AAh
		BSF	EECON1, WR	; start program (CPU stall)
		BSF	INTCON, GIE	; re-enable interrupts
		BCF	EECON1, WREN	; disable write to memory

### 6.5.2 WRITE VERIFY

Depending on the application, good programming practice may dictate that the value written to the memory should be verified against the original value. This should be used in applications where excessive writes can stress bits near the specification limit.

### 6.5.3 UNEXPECTED TERMINATION OF WRITE OPERATION

If a write is terminated by an unplanned event, such as loss of power or an unexpected Reset, the memory location just programmed should be verified and reprogrammed if needed. If the write operation is interrupted by a MCLR Reset or a WDT Time-out Reset during normal operation, the user can check the WRERR bit and rewrite the location(s) as needed.

### 6.5.4 PROTECTION AGAINST SPURIOUS WRITES

To protect against spurious writes to Flash program memory, the write initiate sequence must also be followed. See **Section 23.0 “Special Features of the CPU”** for more detail.

## 6.6 Flash Program Operation During Code Protection

See **Section 23.5 “Program Verification and Code Protection”** for details on code protection of Flash program memory.

**TABLE 6-2: REGISTERS ASSOCIATED WITH PROGRAM FLASH MEMORY**

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
TBLPTRU	—	—	bit 21	Program Memory Table Pointer Upper Byte (TBLPTR<20:16>)					49
TBLPTRH	Program Memory Table Pointer High Byte (TBLPTR<15:8>)								49
TBLPTRL	Program Memory Table Pointer Low Byte (TBLPTR<7:0>)								49
TABLAT	Program Memory Table Latch								49
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	49
EECON2	EEPROM Control Register 2 (not a physical register)								51
EECON1	EEPGD	CFGS	—	FREE	WRERR	WREN	WR	RD	51
IPR2	OSCFIP	CMIP	—	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP	52
PIR2	OSCFIF	CMIF	—	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF	52
PIE2	OSCFIE	CMIE	—	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE	52

**Legend:** — = unimplemented, read as ‘0’. Shaded cells are not used during Flash/EEPROM access.

# PIC18F2420/2520/4420/4520

## 8.0 8 x 8 HARDWARE MULTIPLIER

### 8.1 Introduction

All PIC18 devices include an 8 x 8 hardware multiplier as part of the ALU. The multiplier performs an unsigned operation and yields a 16-bit result that is stored in the product register pair, PRODH:PRODL. The multiplier's operation does not affect any flags in the STATUS register.

Making multiplication a hardware operation allows it to be completed in a single instruction cycle. This has the advantages of higher computational throughput and reduced code size for multiplication algorithms and allows the PIC18 devices to be used in many applications previously reserved for digital signal processors. A comparison of various hardware and software multiply operations, along with the savings in memory and execution time, is shown in Table 8-1.

### 8.2 Operation

Example 8-1 shows the instruction sequence for an 8 x 8 unsigned multiplication. Only one instruction is required when one of the arguments is already loaded in the WREG register.

Example 8-2 shows the sequence to do an 8 x 8 signed multiplication. To account for the sign bits of the arguments, each argument's Most Significant bit (MSb) is tested and the appropriate subtractions are done.

#### EXAMPLE 8-1: 8 x 8 UNSIGNED MULTIPLY ROUTINE

```
MOVF    ARG1, W    ;  
MULWF   ARG2        ; ARG1 * ARG2 ->  
                        ; PRODH:PRODL
```

#### EXAMPLE 8-2: 8 x 8 SIGNED MULTIPLY ROUTINE

```
MOVF    ARG1, W  
MULWF   ARG2        ; ARG1 * ARG2 ->  
                        ; PRODH:PRODL  
  
BTFSC   ARG2, SB    ; Test Sign Bit  
SUBWF   PRODH, F    ; PRODH = PRODH  
                        ; - ARG1  
  
MOVF    ARG2, W  
BTFSC   ARG1, SB    ; Test Sign Bit  
SUBWF   PRODH, F    ; PRODH = PRODH  
                        ; - ARG2
```

TABLE 8-1: PERFORMANCE COMPARISON FOR VARIOUS MULTIPLY OPERATIONS

Routine	Multiply Method	Program Memory (Words)	Cycles (Max)	Time		
				@ 40 MHz	@ 10 MHz	@ 4 MHz
8 x 8 unsigned	Without hardware multiply	13	69	6.9 µs	27.6 µs	69 µs
	Hardware multiply	1	1	100 ns	400 ns	1 µs
8 x 8 signed	Without hardware multiply	33	91	9.1 µs	36.4 µs	91 µs
	Hardware multiply	6	6	600 ns	2.4 µs	6 µs
16 x 16 unsigned	Without hardware multiply	21	242	24.2 µs	96.8 µs	242 µs
	Hardware multiply	28	28	2.8 µs	11.2 µs	28 µs
16 x 16 signed	Without hardware multiply	52	254	25.4 µs	102.6 µs	254 µs
	Hardware multiply	35	40	4.0 µs	16.0 µs	40 µs

# PIC18F2420/2520/4420/4520

## 10.2 PORTB, TRISB and LATB Registers

PORTB is an 8-bit wide, bidirectional port. The corresponding Data Direction register is TRISB. Setting a TRISB bit (= 1) will make the corresponding PORTB pin an input (i.e., put the corresponding output driver in a high-impedance mode). Clearing a TRISB bit (= 0) will make the corresponding PORTB pin an output (i.e., put the contents of the output latch on the selected pin).

The Data Latch register (LATB) is also memory mapped. Read-modify-write operations on the LATB register read and write the latched output value for PORTB.

### EXAMPLE 10-2: INITIALIZING PORTB

```
CLRF    PORTB    ; Initialize PORTB by
                  ; clearing output
                  ; data latches
CLRF    LATB     ; Alternate method
                  ; to clear output
                  ; data latches
MOVLW   0Fh      ; Set RB<4:0> as
MOVWF   ADCON1   ; digital I/O pins
                  ; (required if config bit
                  ; PBDEN is set)
MOVLW   0CFh     ; Value used to
                  ; initialize data
                  ; direction
MOVWF   TRISB    ; Set RB<3:0> as inputs
                  ; RB<5:4> as outputs
                  ; RB<7:6> as inputs
```

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by clearing bit,  $\overline{\text{RBPU}}$  (INTCON2<7>). The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset.

**Note:** On a Power-on Reset, RB<4:0> are configured as analog inputs by default and read as '0'; RB<7:5> are configured as digital inputs.

By programming the Configuration bit, PBDEN, RB<4:0> will alternatively be configured as digital inputs on POR.

Four of the PORTB pins (RB<7:4>) have an interrupt-on-change feature. Only pins configured as inputs can cause this interrupt to occur (i.e., any RB<7:4> pin configured as an output is excluded from the interrupt-on-change comparison). The input pins (of RB<7:4>) are compared with the old value latched on the last read of PORTB. The "mismatch" outputs of RB<7:4> are ORed together to generate the RB Port Change Interrupt with Flag bit, RBIF (INTCON<0>).

This interrupt can wake the device from the Sleep mode, or any of the Idle modes. The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

- Any read or write of PORTB (except with the MOVFF (ANY), PORTB instruction).
- Clear flag bit, RBIF.

A mismatch condition will continue to set flag bit, RBIF. Reading PORTB will end the mismatch condition and allow flag bit, RBIF, to be cleared.

The interrupt-on-change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt-on-change feature. Polling of PORTB is not recommended while using the interrupt-on-change feature.

RB3 can be configured by the Configuration bit, CCP2MX, as the alternate peripheral pin for the CCP2 module (CCP2MX = 0).

# PIC18F2420/2520/4420/4520

**TABLE 10-9: PORTE I/O SUMMARY**

Pin	Function	TRIS Setting	I/O	I/O Type	Description
RE0/ $\overline{\text{RD}}$ /AN5	RE0	0	O	DIG	LATE<0> data output; not affected by analog input.
		1	I	ST	PORTE<0> data input; disabled when analog input enabled.
	$\overline{\text{RD}}$	1	I	TTL	PSP read enable input (PSP enabled).
	AN5	1	I	ANA	A/D input channel 5; default input configuration on POR.
RE1/ $\overline{\text{WR}}$ /AN6	RE1	0	O	DIG	LATE<1> data output; not affected by analog input.
		1	I	ST	PORTE<1> data input; disabled when analog input enabled.
	$\overline{\text{WR}}$	1	I	TTL	PSP write enable input (PSP enabled).
	AN6	1	I	ANA	A/D input channel 6; default input configuration on POR.
RE2/ $\overline{\text{CS}}$ /AN7	RE2	0	O	DIG	LATE<2> data output; not affected by analog input.
		1	I	ST	PORTE<2> data input; disabled when analog input enabled.
	$\overline{\text{CS}}$	1	I	TTL	PSP write enable input (PSP enabled).
	AN7	1	I	ANA	A/D input channel 7; default input configuration on POR.
$\overline{\text{MCLR}}$ / $\overline{\text{VPP}}$ /RE3 <sup>(1)</sup>	$\overline{\text{MCLR}}$	—	I	ST	External Master Clear input; enabled when MCLRE Configuration bit is set.
	$\overline{\text{VPP}}$	—	I	ANA	High-voltage detection; used for ICSP™ mode entry detection. Always available regardless of pin mode.
	RE3	— <sup>(2)</sup>	I	ST	PORTE<3> data input; enabled when MCLRE Configuration bit is clear.

**Legend:** DIG = Digital level output; TTL = TTL input buffer; ST = Schmitt Trigger input buffer; ANA = Analog level input/output; x = Don't care (TRIS bit does not affect port direction or is overridden for this option).

**Note 1:** RE3 is available on both 28-pin and 40/44-pin devices. All other PORTE pins are only implemented on 40/44-pin devices.

**2:** RE3 does not have a corresponding TRIS bit to control data direction.

**TABLE 10-10: SUMMARY OF REGISTERS ASSOCIATED WITH PORTE**

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
PORTE	—	—	—	—	RE3 <sup>(1,2)</sup>	RE2	RE1	RE0	52
LATE <sup>(2)</sup>	—	—	—	—	—	LATE Data Latch Register			52
TRISE	IBF	OBF	IBOV	PSPMODE	—	TRISE2	TRISE1	TRISE0	52
ADCON1	—	—	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0	51

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

**Note 1:** Implemented only when Master Clear functionality is disabled (MCLRE Configuration bit = 0).

**2:** RE3 is the only PORTE bit implemented on both 28-pin and 40/44-pin devices. All other bits are implemented only when PORTE is implemented (i.e., 40/44-pin devices).



# PIC18F2420/2520/4420/4520

## 12.1 Timer1 Operation

Timer1 can operate in one of these modes:

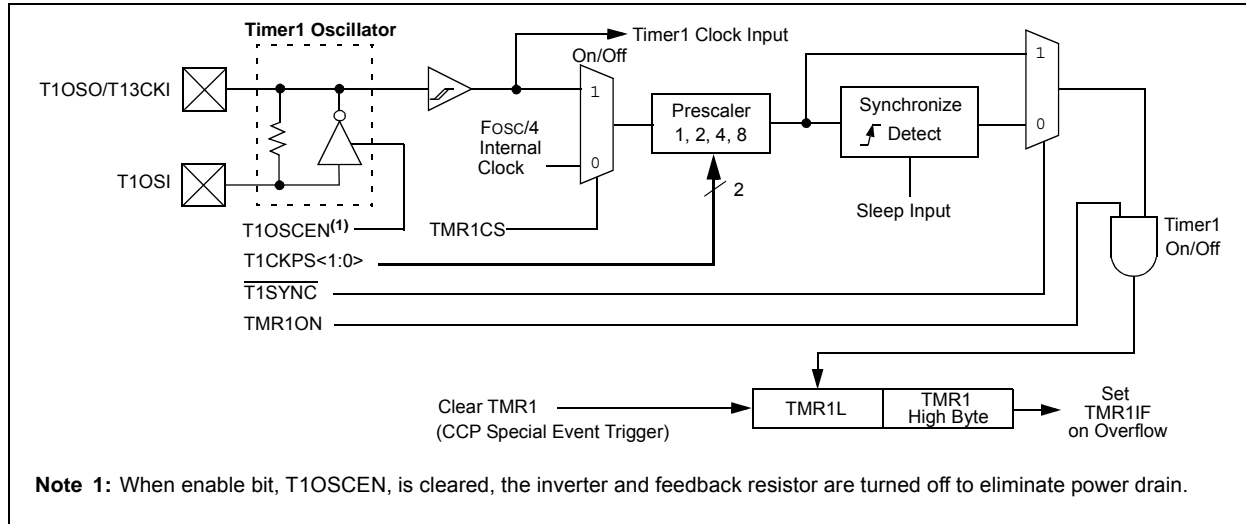
- Timer
- Synchronous Counter
- Asynchronous Counter

The operating mode is determined by the clock select bit, TMR1CS (T1CON<1>). When TMR1CS is cleared (= 0), Timer1 increments on every internal instruction

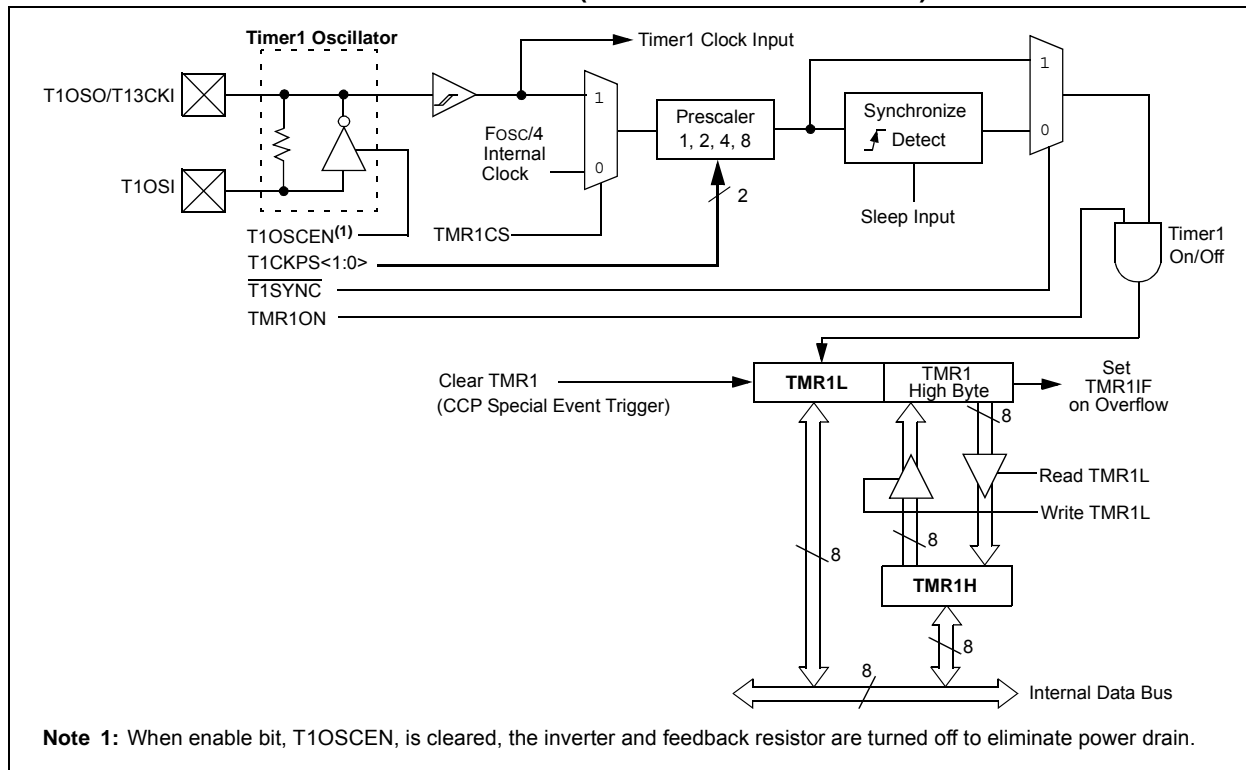
cycle ( $F_{osc}/4$ ). When the bit is set, Timer1 increments on every rising edge of the Timer1 external clock input or the Timer1 oscillator, if enabled.

When Timer1 is enabled, the RC1/T1OSI and RC0/T1OSO/T13CKI pins become inputs. This means the values of TRISC<1:0> are ignored and the pins are read as '0'.

**FIGURE 12-1: TIMER1 BLOCK DIAGRAM**



**FIGURE 12-2: TIMER1 BLOCK DIAGRAM (16-BIT READ/WRITE MODE)**



## 12.7 Considerations in Asynchronous Counter Mode

Following a Timer1 interrupt and an update to the TMR1 registers, the Timer1 module uses a falling edge on its clock source to trigger the next register update on the rising edge. If the update is completed after the clock input has fallen, the next rising edge will not be counted.

If the application can reliably update TMR1 before the timer input goes low, no additional action is needed. Otherwise, an adjusted update can be performed following a later Timer1 increment. This can be done by

monitoring TMR1L within the interrupt routine until it increments, and then updating the TMR1H:TMR1L register pair while the clock is low, or one-half of the period of the clock source. Assuming that Timer1 is being used as a Real-Time Clock, the clock source is a 32.768 kHz crystal oscillator; in this case, one half period of the clock is 15.25  $\mu$ s.

The Real-Time Clock application code in Example 12-1 shows a typical ISR for Timer1, as well as the optional code required if the update cannot be done reliably within the required interval.

### EXAMPLE 12-1: IMPLEMENTING A REAL-TIME CLOCK USING A TIMER1 INTERRUPT SERVICE

```

RTCinit
    MOVLW    80h           ; Preload TMR1 register pair
    MOVWF    TMR1H         ; for 1 second overflow
    CLRF     TMR1L
    MOVLW    b'00001111'   ; Configure for external clock,
    MOVWF    T1CON         ; Asynchronous operation, external oscillator
    CLRF     secs          ; Initialize timekeeping registers
    CLRF     mins
    MOVLW    .12
    MOVWF    hours
    BSF      PIE1, TMR1IE   ; Enable Timer1 interrupt
    RETURN

RTCisr
    ; Start ISR here
    ; Insert the next 4 lines of code when TMR1
    ; can not be reliably updated before clock pulse goes low
    BTFSC    TMR1L,0       ; wait for TMR1L<0> to become clear
    BRA      $-2           ; (may already be clear)
    BTFSS    TMR1L,0       ; wait for TMR1L<0> to become set
    BRA      $-2           ; TMR1 has just incremented

    ; If TMR1 update can be completed before clock pulse goes low
    BSF      TMR1H, 7       ; Preload for 1 sec overflow
    BCF      PIR1, TMR1IF   ; Clear interrupt flag
    INCF     secs, F        ; Increment seconds
    MOVLW    .59            ; 60 seconds elapsed?
    CPFSGT   secs
    RETURN    ; No, done
    CLRF     secs          ; Clear seconds
    INCF     mins, F        ; Increment minutes
    MOVLW    .59            ; 60 minutes elapsed?
    CPFSGT   mins
    RETURN    ; No, done
    CLRF     mins          ; clear minutes
    INCF     hours, F       ; Increment hours
    MOVLW    .23            ; 24 hours elapsed?
    CPFSGT   hours
    RETURN    ; No, done
    CLRF     hours         ; Reset hours
    RETURN    ; Done
    
```

# PIC18F2420/2520/4420/4520

In addition to the expanded range of modes available through the CCP1CON register and ECCP1AS register, the ECCP module has an additional register associated with Enhanced PWM operation and auto-shutdown features. It is:

- PWM1CON (PWM Dead-Band Delay)

## 16.1 ECCP Outputs and Configuration

The Enhanced CCP module may have up to four PWM outputs, depending on the selected operating mode. These outputs, designated P1A through P1D, are multiplexed with I/O pins on PORTC and PORTD. The outputs that are active depend on the CCP operating mode selected. The pin assignments are summarized in Table 16-1.

To configure the I/O pins as PWM outputs, the proper PWM mode must be selected by setting the P1M<1:0> and CCP1M<3:0> bits. The appropriate TRISC and TRISD direction bits for the port pins must also be set as outputs.

### 16.1.1 ECCP MODULES AND TIMER RESOURCES

Like the standard CCP modules, the ECCP module can utilize Timers 1, 2 or 3, depending on the mode selected. Timer1 and Timer3 are available for modules in Capture or Compare modes, while Timer2 is available for modules in PWM mode. Interactions between the standard and Enhanced CCP modules are identical to those described for standard CCP modules. Additional details on timer resources are provided in **Section 15.1.1 “CCP Modules and Timer Resources”**.

## 16.2 Capture and Compare Modes

Except for the operation of the Special Event Trigger discussed below, the Capture and Compare modes of the ECCP module are identical in operation to that of CCP2. These are discussed in detail in **Section 15.2 “Capture Mode”** and **Section 15.3 “Compare Mode”**. No changes are required when moving between 28-pin and 40/44-pin devices.

### 16.2.1 SPECIAL EVENT TRIGGER

The Special Event Trigger output of ECCP resets the TMR1 or TMR3 register pair, depending on which timer resource is currently selected. This allows the CCPR1 register to effectively be a 16-Bit Programmable Period register for Timer1 or Timer3.

## 16.3 Standard PWM Mode

When configured in Single Output mode, the ECCP module functions identically to the standard CCP module in PWM mode, as described in **Section 15.4 “PWM Mode”**. This is also sometimes referred to as “Compatible CCP” mode, as in Table 16-1.

**Note:** When setting up single output PWM operations, users are free to use either of the processes described in **Section 15.4.4 “Setup for PWM Operation”** or **Section 16.4.9 “Setup for PWM Operation”**. The latter is more generic and will work for either single or multi-output PWM.

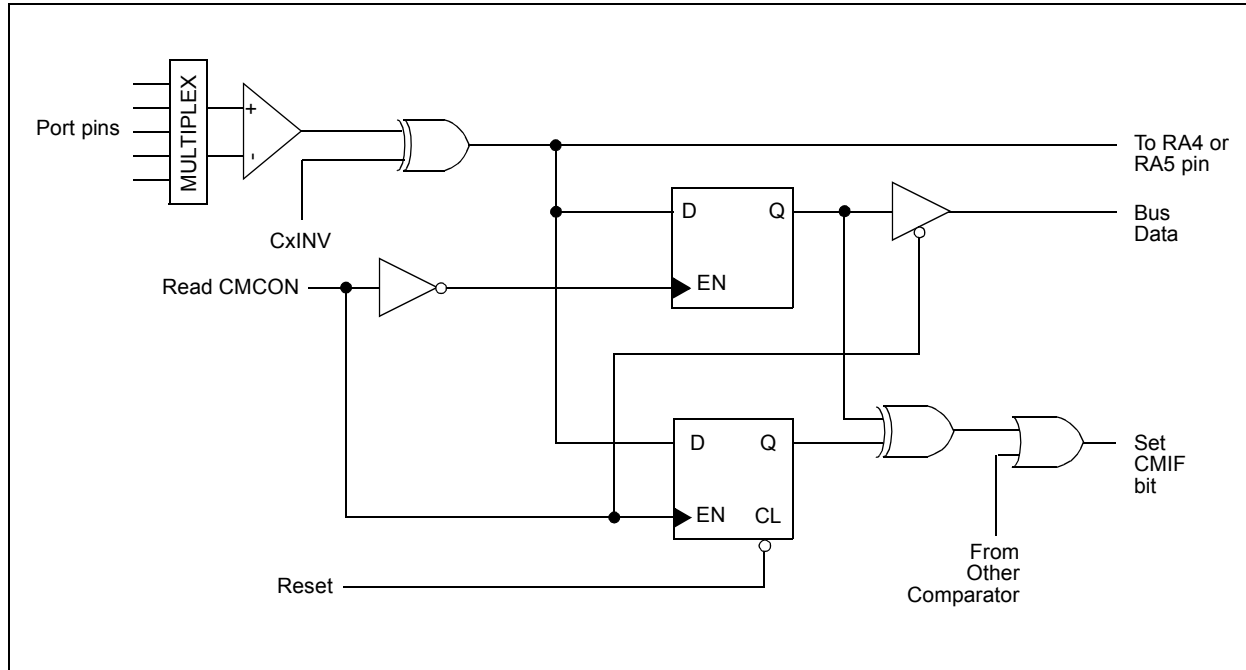
**TABLE 16-1: PIN ASSIGNMENTS FOR VARIOUS ECCP MODES**

ECCP Mode	CCP1CON Configuration	RC2	RD5	RD6	RD7
All 40/44-Pin Devices:					
Compatible CCP	00xx 11xx	CCP1	RD5/PSP5	RD6/PSP6	RD7/PSP7
Dual PWM	10xx 11xx	P1A	P1B	RD6/PSP6	RD7/PSP7
Quad PWM	x1xx 11xx	P1A	P1B	P1C	P1D

**Legend:** x = Don't care. Shaded cells indicate pin assignments not used by ECCP in a given mode.

# PIC18F2420/2520/4420/4520

FIGURE 20-3: COMPARATOR OUTPUT BLOCK DIAGRAM



## 20.6 Comparator Interrupts

The comparator interrupt flag is set whenever there is a change in the output value of either comparator. Software will need to maintain information about the status of the output bits, as read from CMCON<7:6>, to determine the actual change that occurred. The CMIF bit (PIR2<6>) is the Comparator Interrupt Flag. The CMIF bit must be reset by clearing it. Since it is also possible to write a '1' to this register, a simulated interrupt may be initiated.

Both the CMIE bit (PIE2<6>) and the PEIE bit (INTCON<6>) must be set to enable the interrupt. In addition, the GIE bit (INTCON<7>) must also be set. If any of these bits are clear, the interrupt is not enabled, though the CMIF bit will still be set if an interrupt condition occurs.

**Note:** If a change in the CMCON register (C1OUT or C2OUT) should occur when a read operation is being executed (start of the Q2 cycle), then the CMIF (PIR2<6>) interrupt flag may not get set.

The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

- Any read or write of CMCON will end the mismatch condition.
- Clear flag bit, CMIF.

A mismatch condition will continue to set flag bit, CMIF. Reading CMCON will end the mismatch condition and allow flag bit, CMIF, to be cleared.

## 20.7 Comparator Operation During Sleep

When a comparator is active and the device is placed in Sleep mode, the comparator remains active and the interrupt is functional if enabled. This interrupt will wake-up the device from Sleep mode when enabled. Each operational comparator will consume additional current, as shown in the comparator specifications. To minimize power consumption while in Sleep mode, turn off the comparators (CM<2:0> = 111) before entering Sleep. If the device wakes up from Sleep, the contents of the CMCON register are not affected.

## 20.8 Effects of a Reset

A device Reset forces the CMCON register to its Reset state, causing the comparator modules to be turned off (CM<2:0> = 111). However, the input pins (RA0 through RA3) are configured as analog inputs by default on device Reset. The I/O configuration for these pins is determined by the setting of the PCFG<3:0> bits (ADCON1<3:0>). Therefore, device current is minimized when analog inputs are present at Reset time.

# PIC18F2420/2520/4420/4520

## 23.5.1 PROGRAM MEMORY CODE PROTECTION

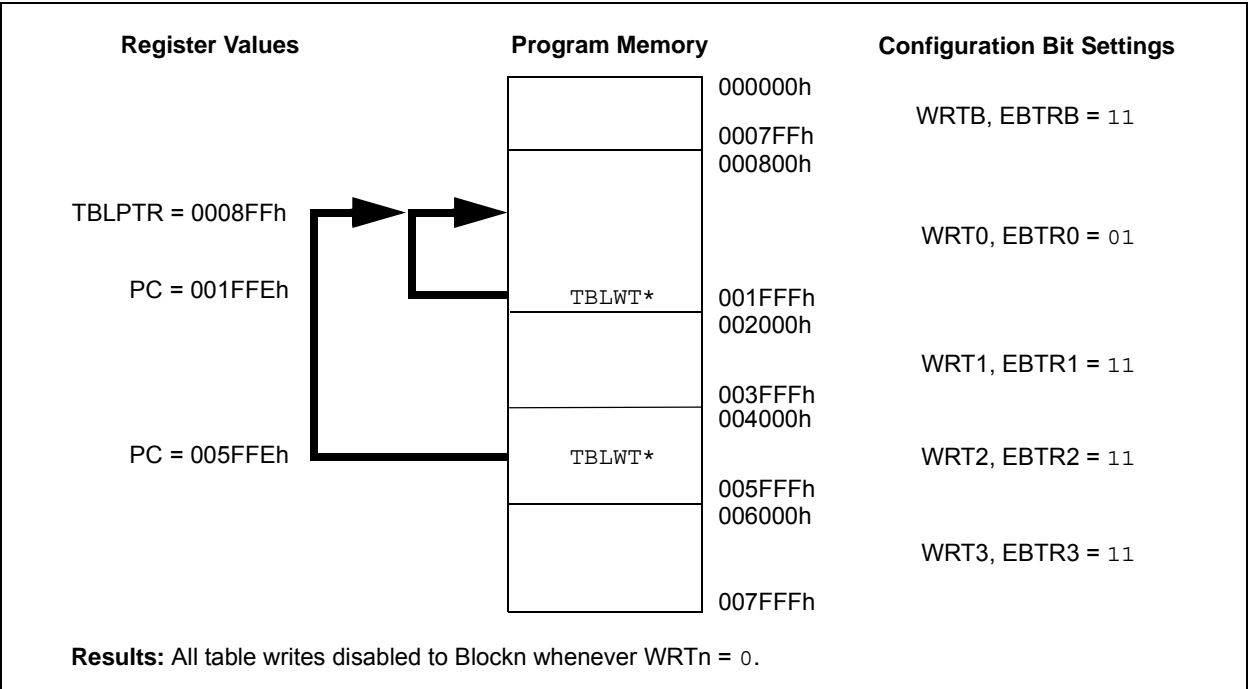
The program memory may be read to, or written from, any location using the table read and table write instructions. The Device ID may be read with table reads. The Configuration registers may be read and written with the table read and table write instructions.

In normal execution mode, the CPn bits have no direct effect. CPn bits inhibit external reads and writes. A block of user memory may be protected from table writes if the WRTn Configuration bit is '0'. The EBTRn bits control table reads. For a block of user memory with the EBTRn bit set to '0', a table read instruction that executes from within that block is allowed to read. A table read

instruction that executes from a location outside of that block is not allowed to read and will result in reading '0's. Figures 23-6 through 23-8 illustrate table write and table read protection.

**Note:** Code protection bits may only be written to a '0' from a '1' state. It is not possible to write a '1' to a bit in the '0' state. Code protection bits are only set to '1' by a full chip erase or block erase function. The full chip erase and block erase functions can only be initiated via ICSP or an external programmer.

FIGURE 23-6: TABLE WRITE (WRTn) DISALLOWED



# PIC18F2420/2520/4420/4520

**TABLE 24-1: OPCODE FIELD DESCRIPTIONS**

Field	Description
a	RAM access bit a = 0: RAM location in Access RAM (BSR register is ignored) a = 1: RAM bank is specified by BSR register
bbb	Bit address within an 8-bit file register (0 to 7).
BSR	Bank Select Register. Used to select the current RAM bank.
C, DC, Z, OV, N	ALU Status bits: <b>C</b> arry, <b>D</b> igit <b>C</b> arry, <b>Z</b> ero, <b>O</b> verflow, <b>N</b> egative.
d	Destination select bit d = 0: store result in WREG d = 1: store result in file register f
dest	Destination: either the WREG register or the specified register file location.
f	8-bit Register file address (00h to FFh) or 2-bit FSR designator (0h to 3h).
f <sub>s</sub>	12-bit Register file address (000h to FFFh). This is the source address.
f <sub>d</sub>	12-bit Register file address (000h to FFFh). This is the destination address.
GIE	Global Interrupt Enable bit.
k	Literal field, constant data or label (may be either an 8-bit, 12-bit or a 20-bit value).
label	Label name.
mm	The mode of the TBLPTR register for the table read and table write instructions. Only used with table read and table write instructions:
*	No change to register (such as TBLPTR with table reads and writes)
*+	Post-Increment register (such as TBLPTR with table reads and writes)
*-	Post-Decrement register (such as TBLPTR with table reads and writes)
+*	Pre-Increment register (such as TBLPTR with table reads and writes)
n	The relative address (2's complement number) for relative branch instructions or the direct address for Call/Branch and Return instructions.
PC	Program Counter.
PCL	Program Counter Low Byte.
PCH	Program Counter High Byte.
PCLATH	Program Counter High Byte Latch.
PCLATU	Program Counter Upper Byte Latch.
PD	Power-down bit.
PRODH	Product of Multiply High Byte.
PRODL	Product of Multiply Low Byte.
s	Fast Call/Return mode select bit s = 0: do not update into/from shadow registers s = 1: certain registers loaded into/from shadow registers (Fast mode)
TBLPTR	21-bit Table Pointer (points to a Program Memory location).
TABLAT	8-bit Table Latch.
TO	Time-out bit.
TOS	Top-of-Stack.
u	Unused or unchanged.
WDT	Watchdog Timer.
WREG	Working register (accumulator).
x	Don't care ('0' or '1'). The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
z <sub>s</sub>	7-bit offset value for indirect addressing of register files (source).
z <sub>d</sub>	7-bit offset value for indirect addressing of register files (destination).
{ }	Optional argument.
[text]	Indicates an indexed address.
(text)	The contents of text.
[expr] <n>	Specifies bit n of the register indicated by the pointer expr.
→	Assigned to.
< >	Register bit field.
∈	In the set of.
italics	User-defined term (font is Courier New).

# PIC18F2420/2520/4420/4520

**TABLE 24-2: PIC18FXXX INSTRUCTION SET**

Mnemonic, Operands	Description	Cycles	16-Bit Instruction Word				Status Affected	Notes
			MSb		LSb			
BYTE-ORIENTED OPERATIONS								
ADDWF f, d, a	Add WREG and f	1	0010	01da	ffff	ffff	C, DC, Z, OV, N	1, 2
ADDWFC f, d, a	Add WREG and Carry bit to f	1	0010	00da	ffff	ffff	C, DC, Z, OV, N	1, 2
ANDWF f, d, a	AND WREG with f	1	0001	01da	ffff	ffff	Z, N	1, 2
CLRF f, a	Clear f	1	0110	101a	ffff	ffff	Z	2
COMF f, d, a	Complement f	1	0001	11da	ffff	ffff	Z, N	1, 2
CPFSEQ f, a	Compare f with WREG, Skip =	1 (2 or 3)	0110	001a	ffff	ffff	None	4
CPFSGT f, a	Compare f with WREG, Skip >	1 (2 or 3)	0110	010a	ffff	ffff	None	4
CPFSLT f, a	Compare f with WREG, Skip <	1 (2 or 3)	0110	000a	ffff	ffff	None	1, 2
DECf f, d, a	Decrement f	1	0000	01da	ffff	ffff	C, DC, Z, OV, N	1, 2, 3, 4
DECFSZ f, d, a	Decrement f, Skip if 0	1 (2 or 3)	0010	11da	ffff	ffff	None	1, 2, 3, 4
DCFSNZ f, d, a	Decrement f, Skip if Not 0	1 (2 or 3)	0100	11da	ffff	ffff	None	1, 2
INCF f, d, a	Increment f	1	0010	10da	ffff	ffff	C, DC, Z, OV, N	1, 2, 3, 4
INCFSZ f, d, a	Increment f, Skip if 0	1 (2 or 3)	0011	11da	ffff	ffff	None	4
INFSNZ f, d, a	Increment f, Skip if Not 0	1 (2 or 3)	0100	10da	ffff	ffff	None	1, 2
IORWF f, d, a	Inclusive OR WREG with f	1	0001	00da	ffff	ffff	Z, N	1, 2
MOVF f, d, a	Move f	1	0101	00da	ffff	ffff	Z, N	1
MOVFF f <sub>s</sub> , f <sub>d</sub>	Move f <sub>s</sub> (source) to f <sub>d</sub> (destination)	2	1100	ffff	ffff	ffff	None	
	1st word		1111	ffff	ffff	ffff		
MOVWF f, a	Move WREG to f	1	0110	111a	ffff	ffff	None	
MULWF f, a	Multiply WREG with f	1	0000	001a	ffff	ffff	None	1, 2
NEGf f, a	Negate f	1	0110	110a	ffff	ffff	C, DC, Z, OV, N	
RLCF f, d, a	Rotate Left f through Carry	1	0011	01da	ffff	ffff	C, Z, N	1, 2
RLNCF f, d, a	Rotate Left f (No Carry)	1	0100	01da	ffff	ffff	Z, N	
RRCf f, d, a	Rotate Right f through Carry	1	0011	00da	ffff	ffff	C, Z, N	
RRNCF f, d, a	Rotate Right f (No Carry)	1	0100	00da	ffff	ffff	Z, N	
SETf f, a	Set f	1	0110	100a	ffff	ffff	None	1, 2
SUBFWB f, d, a	Subtract f from WREG with Borrow	1	0101	01da	ffff	ffff	C, DC, Z, OV, N	
SUBWF f, d, a	Subtract WREG from f	1	0101	11da	ffff	ffff	C, DC, Z, OV, N	1, 2
SUBWFB f, d, a	Subtract WREG from f with Borrow	1	0101	10da	ffff	ffff	C, DC, Z, OV, N	
SWAPf f, d, a	Swap Nibbles in f	1	0011	10da	ffff	ffff	None	4
TSTFSZ f, a	Test f, Skip if 0	1 (2 or 3)	0110	011a	ffff	ffff	None	1, 2
XORWF f, d, a	Exclusive OR WREG with f	1	0001	10da	ffff	ffff	Z, N	

- 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

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><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)



## 26.0 ELECTRICAL CHARACTERISTICS

### Absolute Maximum Ratings<sup>(†)</sup>

Ambient temperature under bias .....	-40°C to +125°C
Storage temperature .....	-65°C to +150°C
Voltage on any pin with respect to V <sub>SS</sub> (except V <sub>DD</sub> and $\overline{\text{MCLR}}$ ) .....	-0.3V to (V <sub>DD</sub> + 0.3V)
Voltage on V <sub>DD</sub> with respect to V <sub>SS</sub> .....	-0.3V to +7.5V
Voltage on $\overline{\text{MCLR}}$ with respect to V <sub>SS</sub> ( <b>Note 2</b> ) .....	0V to +13.25V
Total power dissipation ( <b>Note 1</b> ) .....	1.0W
Maximum current out of V <sub>SS</sub> pin .....	300 mA
Maximum current into V <sub>DD</sub> pin .....	250 mA
Input clamp current, I <sub>IK</sub> (V <sub>I</sub> < 0 or V <sub>I</sub> > V <sub>DD</sub> ) .....	±20 mA
Output clamp current, I <sub>OK</sub> (V <sub>O</sub> < 0 or V <sub>O</sub> > V <sub>DD</sub> ) .....	±20 mA
Maximum output current sunk by any I/O pin .....	25 mA
Maximum output current sourced by any I/O pin .....	25 mA
Maximum current sunk by all ports .....	200 mA
Maximum current sourced by all ports .....	200 mA

**Note 1:** Power dissipation is calculated as follows:

$$P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$$

- 2:** Voltage spikes below V<sub>SS</sub> at the  $\overline{\text{MCLR}}$ /V<sub>PP</sub>/RE3 pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100Ω should be used when applying a “low” level to the  $\overline{\text{MCLR}}$ /V<sub>PP</sub>/RE3 pin, rather than pulling this pin directly to V<sub>SS</sub>.

† **NOTICE:** Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

# PIC18F2420/2520/4420/4520

## 26.3 DC Characteristics: PIC18F2420/2520/4420/4520 (Industrial) PIC18LF2420/2520/4420/4520 (Industrial)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial			
Param No.	Symbol	Characteristic	Min	Max	Units	Conditions
D030 D030A D031 D031A D031B D032 D033 D033A D033B D034	$V_{IL}$	<b>Input Low Voltage</b> I/O Ports: with TTL Buffer with Schmitt Trigger Buffer RC3 and RC4 $\overline{\text{MCLR}}$ OSC1 OSC1 OSC1 T13CKI	$V_{SS}$ — $V_{SS}$ $V_{SS}$ $V_{SS}$ $V_{SS}$ $V_{SS}$ $V_{SS}$ $V_{SS}$ $V_{SS}$	$0.15 V_{DD}$ 0.8 $0.2 V_{DD}$ $0.3 V_{DD}$ 0.8 $0.2 V_{DD}$ $0.3 V_{DD}$ $0.2 V_{DD}$ 0.3 0.3	V V V V V V V V V V	$V_{DD} < 4.5\text{V}$ $4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ $I^2\text{C}^{\text{TM}}$ enabled SMBus enabled HS, HSPLL modes RC, EC modes <sup>(1)</sup> XT, LP modes
D040 D040A D041 D041A D041B D042 D043 D043A D043B D043C D044	$V_{IH}$	<b>Input High Voltage</b> I/O Ports: with TTL Buffer with Schmitt Trigger Buffer RC3 and RC4 $\overline{\text{MCLR}}$ OSC1 OSC1 OSC1 OSC1 T13CKI	$0.25 V_{DD} + 0.8\text{V}$ 2.0 $0.8 V_{DD}$ $0.7 V_{DD}$ 2.1 $0.8 V_{DD}$ $0.7 V_{DD}$ $0.8 V_{DD}$ $0.9 V_{DD}$ 1.6 1.6	$V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$	V V V V V V V V V V V	$V_{DD} < 4.5\text{V}$ $4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ $I^2\text{C}$ enabled SMBus enabled HS, HSPLL modes EC mode RC mode <sup>(1)</sup> XT, LP modes
D060  D061 D063	$I_{IL}$	<b>Input Leakage Current<sup>(2,3)</sup></b> I/O Ports $\overline{\text{MCLR}}$ OSC1	— — —	$\pm 200$ $\pm 50$ $\pm 1$ $\pm 1$	nA nA $\mu\text{A}$ $\mu\text{A}$	$V_{DD} < 5.5\text{V}$ , $V_{SS} \leq V_{PIN} \leq V_{DD}$ , Pin at high-impedance $V_{DD} < 3\text{V}$ , $V_{SS} \leq V_{PIN} \leq V_{DD}$ , Pin at high-impedance $V_{SS} \leq V_{PIN} \leq V_{DD}$ $V_{SS} \leq V_{PIN} \leq V_{DD}$
D070	IPU IPURB	<b>Weak Pull-up Current</b> PORTB Weak Pull-up Current	50	400	$\mu\text{A}$	$V_{DD} = 5\text{V}$ , $V_{PIN} = V_{SS}$

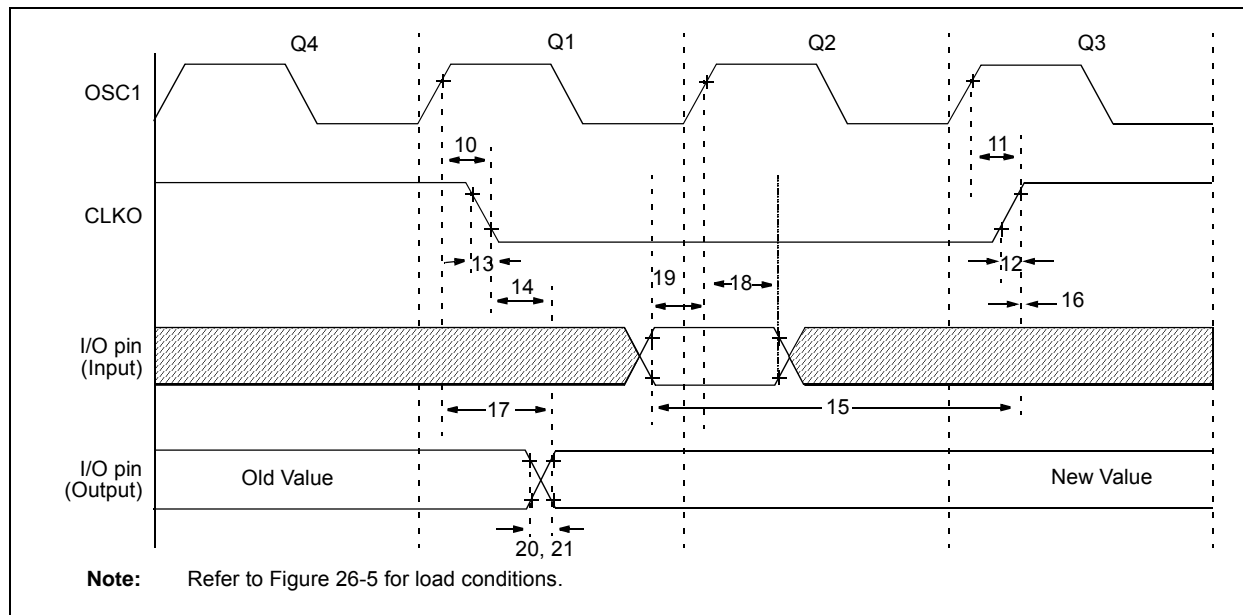
**Note 1:** In RC oscillator configuration, the OSC1/CLKI pin is a Schmitt Trigger input. It is not recommended that the PIC<sup>®</sup> device be driven with an external clock while in RC mode.

**2:** The leakage current on the  $\overline{\text{MCLR}}$  pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

**3:** Negative current is defined as current sourced by the pin.

# PIC18F2420/2520/4420/4520

**FIGURE 26-7: CLKO AND I/O TIMING**



**TABLE 26-9: CLKO AND I/O TIMING REQUIREMENTS**

Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions
10	TosH2ckL	OSC1 ↑ to CLKO ↓	—	75	200	ns	(Note 1)
11	TosH2ckH	OSC1 ↑ to CLKO ↑	—	75	200	ns	(Note 1)
12	TckR	CLKO Rise Time	—	35	100	ns	(Note 1)
13	TckF	CLKO Fall Time	—	35	100	ns	(Note 1)
14	TckL2ioV	CLKO ↓ to Port Out Valid	—	—	0.5 T <sub>CY</sub> + 20	ns	(Note 1)
15	TioV2ckH	Port In Valid before CLKO ↑	0.25 T <sub>CY</sub> + 25	—	—	ns	(Note 1)
16	TckH2ioI	Port In Hold after CLKO ↑	0	—	—	ns	(Note 1)
17	TosH2ioV	OSC1 ↑ (Q1 cycle) to Port Out Valid	—	50	150	ns	
18	TosH2ioI	OSC1 ↑ (Q2 cycle) to Port Input Invalid (I/O in hold time)	PIC18FXXXX	100	—	—	ns
18A			PIC18LFXXXX	200	—	—	ns V <sub>DD</sub> = 2.0V
19	TioV2osH	Port Input Valid to OSC1 ↑ (I/O in setup time)	0	—	—	ns	
20	TioR	Port Output Rise Time	PIC18FXXXX	—	10	25	ns
20A			PIC18LFXXXX	—	—	60	ns V <sub>DD</sub> = 2.0V
21	TioF	Port Output Fall Time	PIC18FXXXX	—	10	25	ns
21A			PIC18LFXXXX	—	—	60	ns V <sub>DD</sub> = 2.0V
22†	TINP	INTx pin High or Low Time	T <sub>CY</sub>	—	—	ns	
23†	TRBP	RB<7:4> Change INTx High or Low Time	T <sub>CY</sub>	—	—	ns	

† These parameters are asynchronous events not related to any internal clock edges.

**Note 1:** Measurements are taken in RC mode, where CLKO output is 4 x T<sub>OSC</sub>.

# PIC18F2420/2520/4420/4520

FIGURE 27-11: TYPICAL  $I_{DD}$  ACROSS  $V_{DD}$  (RC\_RUN MODE, +25°C)

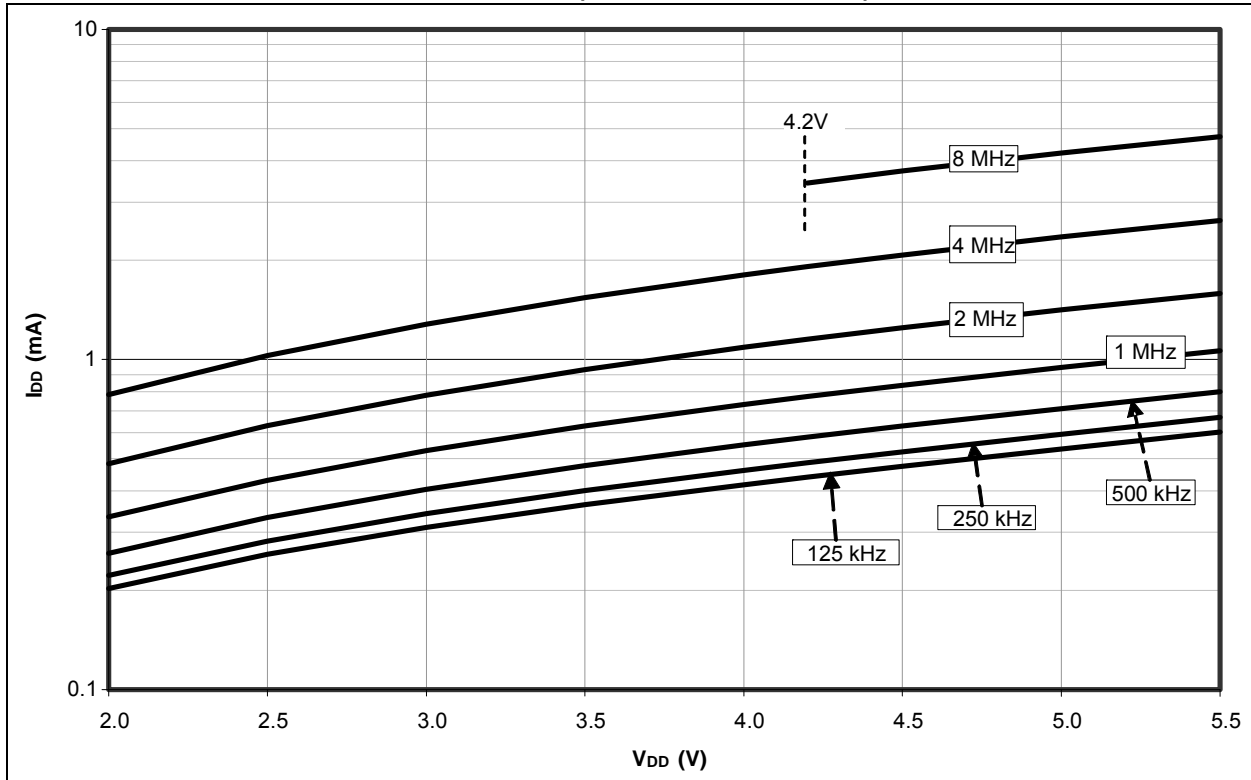
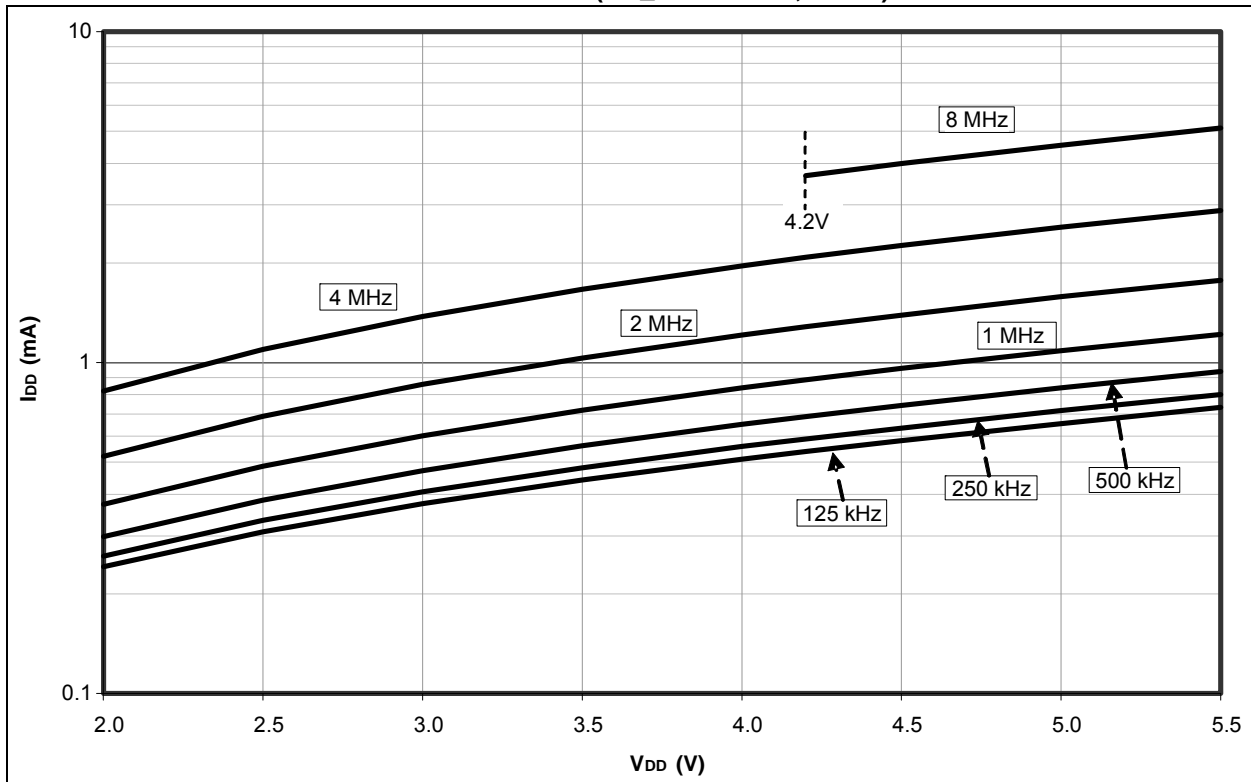


FIGURE 27-12: MAXIMUM  $I_{DD}$  ACROSS  $V_{DD}$  (RC\_RUN MODE, +85°C)



## THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at [www.microchip.com](http://www.microchip.com). This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

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

## CUSTOMER CHANGE NOTIFICATION SERVICE

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

To register, access the Microchip web site at [www.microchip.com](http://www.microchip.com), click on Customer Change Notification and follow the registration instructions.

## CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support
- Development Systems Information Line

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

**Technical support is available through the web site at: <http://support.microchip.com>**