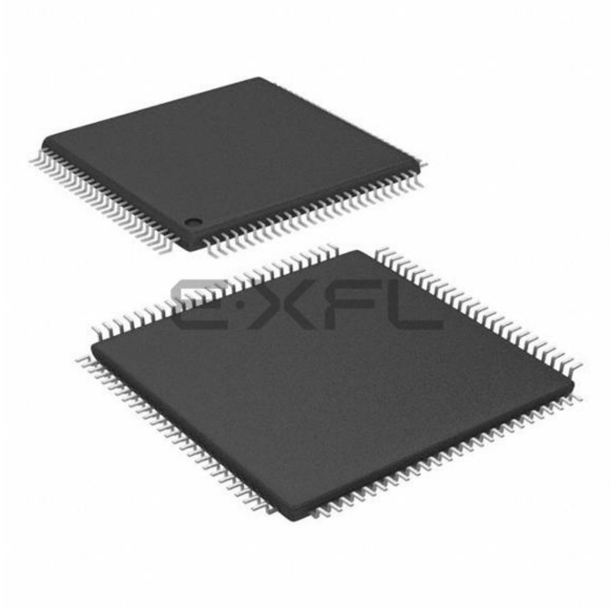
Microchip Technology - PIC18F96J60-I/PF Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	41.667MHz
Connectivity	EBI/EMI, Ethernet, I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	70
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	3808 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TQFP
Supplier Device Package	100-TQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f96j60-i-pf

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE**.** Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949:2002 ==

Trademarks

Printed on recycled paper.

ISBN: 978-1-61341-069-1

Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

TABLE 1-6: PIC18F96J60/96J65/97J60 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number	Pin Type	Buffer Type	Description
	TQFP			
				PORTD is a bidirectional I/O port.
RD0/AD0/PSP0	92			
RD0		I/O	ST	Digital I/O.
AD0		I/O	TTL	External Memory Address/Data 0.
PSP0		I/O	TTL	Parallel Slave Port data.
RD1/AD1/PSP1	91			
RD1		I/O	ST	Digital I/O.
AD1		I/O	TTL	External Memory Address/Data 1.
PSP1		I/O	TTL	Parallel Slave Port data.
RD2/AD2/PSP2	90			
RD2		I/O	ST	Digital I/O.
AD2		I/O	TTL	External Memory Address/Data 2.
PSP2		I/O	TTL	Parallel Slave Port data.
RD3/AD3/PSP3	89			
RD3		I/O	ST	Digital I/O.
AD3		I/O	TTL	External Memory Address/Data 3.
PSP3		I/O	TTL	Parallel Slave Port data.
RD4/AD4/PSP4/SDO2	88			
RD4		I/O	ST	Digital I/O.
AD4		I/O	TTL	External Memory Address/Data 4.
PSP4		I/O	TTL	Parallel Slave Port data.
SDO2		O	—	SPI data out.
RD5/AD5/PSP5/ SDI2/SDA2	87			
RD5		I/O	ST	Digital I/O.
AD5		I/O	TTL	External Memory Address/Data 5.
PSP5		I/O	TTL	Parallel Slave Port data.
SDI2		I	ST	SPI data in.
SDA2		I/O	ST	I^{2}C™ data I/O.
RD6/AD6/PSP6/ SCK2/SCL2	84			
RD6		I/O	ST	Digital I/O.
AD6		I/O	TTL	External Memory Address/Data 6.
PSP6		I/O	TTL	Parallel Slave Port data.
SCK2		I/O	ST	Synchronous serial clock input/output for SPI mode.
SCL2		I/O	ST	Synchronous serial clock input/output for I^{2}C™ mode.
RD7/AD7/PSP7/$\overline{\text{SS2}}$	83			
RD7		I/O	ST	Digital I/O.
AD7		I/O	TTL	External Memory Address/Data 7.
PSP7		I/O	TTL	Parallel Slave Port data.
$\overline{\text{SS2}}$		I	TTL	SPI slave select input.

Legend:	TTL	= TTL compatible input	CMOS	=	CMOS compatible input or output
	ST	= Schmitt Trigger input with CMOS levels	Analog	=	Analog input
	I	= Input	O	=	Output
	P	= Power	OD	=	Open-Drain (no P diode to VDD)

Note 1: Alternate assignment for ECCP2/P2A when CCP2MX Configuration bit is cleared (Extended Microcontroller mode).

2: Default assignment for ECCP2/P2A for all devices in all operating modes (CCP2MX Configuration bit is set).

3: Default assignments for P1B/P1C/P3B/P3C (ECCPMX Configuration bit is set).

4: Alternate assignment for ECCP2/P2A when CCP2MX Configuration bit is cleared (Microcontroller mode).

5: Alternate assignments for P1B/P1C/P3B/P3C (ECCPMX Configuration bit is cleared).

NOTES:

REGISTER 11-1: PSPCON: PARALLEL SLAVE PORT CONTROL REGISTER

R-0	R-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
IBF	OBF	IBOV	PSPMODE	—	—	—	—
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 **IBF:** Input Buffer Full Status bit

1 = A word has been received and is waiting to be read by the CPU
0 = No word has been received

bit 6 **OBF:** Output Buffer Full Status bit

1 = The output buffer still holds a previously written word
0 = The output buffer has been read

bit 5 **IBOV:** Input Buffer Overflow Detect bit

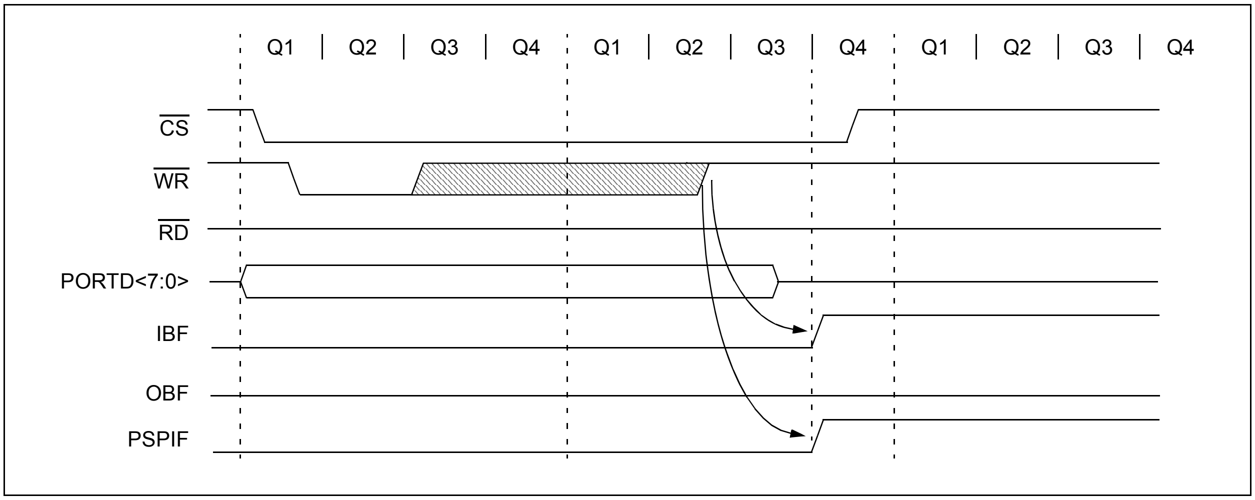
1 = A write occurred when a previously input word has not been read (must be cleared in software)
0 = No overflow occurred

bit 4 **PSPMODE:** Parallel Slave Port Mode Select bit

1 = Parallel Slave Port mode
0 = General Purpose I/O mode

bit 3-0 **Unimplemented:** Read as ‘0’

FIGURE 11-3: PARALLEL SLAVE PORT WRITE WAVEFORMS



15.0 TIMER3 MODULE

The Timer3 timer/counter module incorporates these features:

- Software selectable operation as a 16-bit timer or counter
- Readable and writable 8-bit registers (TMR3H and TMR3L)
- Selectable clock source (internal or external) with device clock or Timer1 oscillator internal options
- Interrupt on overflow
- Module Reset on CCPx/ECCPx Special Event Trigger

A simplified block diagram of the Timer3 module is shown in Figure 15-1. A block diagram of the module's operation in Read/Write mode is shown in Figure 15-2.

The Timer3 module is controlled through the T3CON register (Register 15-1). It also selects the clock source options for the CCPx and ECCPx modules; see **Section 17.1.1 "CCPx/ECCPx Modules and Timer Resources"** for more information.

REGISTER 15-1: T3CON: TIMER3 CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RD16	T3CCP2	T3CKPS1	T3CKPS0	T3CCP1	$\overline{\text{T3SYNC}}$	TMR3CS	TMR3ON
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 Timer3 in one 16-bit operation
0 = Enables register read/write of Timer3 in two 8-bit operations

bit 6,3 **T3CCP<2:1>:** Timer3 and Timer1 to CCPx/ECCPx Enable bits

11 = Timer3 and Timer4 are the clock sources for all CCPx/ECCPx modules
10 = Timer3 and Timer4 are the clock sources for ECCP3, CCP4 and CCP5; Timer1 and Timer2 are the clock sources for ECCP1 and ECCP2
01 = Timer3 and Timer4 are the clock sources for ECCP2, ECCP3, CCP4 and CCP5; Timer1 and Timer2 are the clock sources for ECCP1
00 = Timer1 and Timer2 are the clock sources for all CCPx/ECCPx modules

bit 5-4 **T3CKPS<1:0>:** Timer3 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 2 **$\overline{\text{T3SYNC}}$:** Timer3 External Clock Input Synchronization Select bit
(not usable if the device clock comes from Timer1/Timer3)

When TMR3CS = 1:
1 = Do not synchronize external clock input
0 = Synchronize external clock input

When TMR3CS = 0:
This bit is ignored. Timer3 uses the internal clock when TMR3CS = 0.

bit 1 **TMR3CS:** Timer3 Clock Source Select bit

1 = External clock input from Timer1 oscillator or T13CKI (on the rising edge after the first falling edge)
0 = Internal clock (FOSC/4)

bit 0 **TMR3ON:** Timer3 On bit

1 = Enables Timer3
0 = Stops Timer3

15.2 Timer3 16-Bit Read/Write Mode

Timer3 can be configured for 16-bit reads and writes (see Figure 15-2). When the RD16 control bit (T3CON<7>) is set, the address for TMR3H is mapped to a buffer register for the high byte of Timer3. A read from TMR3L will load the contents of the high byte of Timer3 into the Timer3 High Byte Buffer register. This provides the user with the ability to accurately read all 16 bits of Timer1 without having to determine whether a read of the high byte, followed by a read of the low byte, has become invalid due to a rollover between reads.

A write to the high byte of Timer3 must also take place through the TMR3H Buffer register. The Timer3 high byte is updated with the contents of TMR3H when a write occurs to TMR3L. This allows a user to write all 16 bits to both the high and low bytes of Timer3 at once.

The high byte of Timer3 is not directly readable or writable in this mode. All reads and writes must take place through the Timer3 High Byte Buffer register.

Writes to TMR3H do not clear the Timer3 prescaler. The prescaler is only cleared on writes to TMR3L.

15.3 Using the Timer1 Oscillator as the Timer3 Clock Source

The Timer1 internal oscillator may be used as the clock source for Timer3. The Timer1 oscillator is enabled by setting the T1OSCEN (T1CON<3>) bit. To use it as the Timer3 clock source, the TMR3CS bit must also be set. As previously noted, this also configures Timer3 to increment on every rising edge of the oscillator source.

The Timer1 oscillator is described in **Section 13.0 "Timer1 Module"**.

15.4 Timer3 Interrupt

The TMR3 register pair (TMR3H:TMR3L) increments from 0000h to FFFFh, and overflows to 0000h. The Timer3 interrupt, if enabled, is generated on overflow and is latched in interrupt flag bit, TMR3IF (PIR2<1>). This interrupt can be enabled or disabled by setting or clearing the Timer3 Interrupt Enable bit, TMR3IE (PIE2<1>).

15.5 Resetting Timer3 Using the ECCPx Special Event Trigger

If ECCP1 or ECCP2 is configured to use Timer3 and to generate a Special Event Trigger in Compare mode (CCPxM<3:0> = `1011`), this signal will reset Timer3. The trigger from ECCP2 will also start an A/D conversion if the A/D module is enabled (see **Section 18.2.1 "Special Event Trigger"** for more information).

The module must be configured as either a timer or synchronous counter to take advantage of this feature. When used this way, the CCPRxH:CCPRxL register pair effectively becomes a Period register for Timer3.

If Timer3 is running in Asynchronous Counter mode, the Reset operation may not work.

In the event that a write to Timer3 coincides with a Special Event Trigger from an ECCPx module, the write will take precedence.

Note: The Special Event Triggers from the ECCPx module will not set the TMR3IF interrupt flag bit (PIR2<1>).

TABLE 15-1: REGISTERS ASSOCIATED WITH TIMER3 AS A TIMER/COUNTER

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on Page:
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	69
PIR2	OSCFIF	CMIF	ETHIF	r	BCL1IF	—	TMR3IF	CCP2IF	71
PIE2	OSCFIE	CMIE	ETHIE	r	BCL1IE	—	TMR3IE	CCP2IE	71
IPR2	OSCFIP	CMIP	ETHIP	r	BCL1IP	—	TMR3IP	CCP2IP	71
TMR3L	Timer3 Register Low Byte								70
TMR3H	Timer3 Register High Byte								70
T1CON	RD16	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	70
T3CON	RD16	T3CCP2	T3CKPS1	T3CKPS0	T3CCP1	$\overline{\text{T3SYNC}}$	TMR3CS	TMR3ON	71

Legend: — = unimplemented, read as '`0`', r = reserved. Shaded cells are not used by the Timer3 module.

TABLE 18-3: PIN CONFIGURATIONS FOR ECCP3

ECCP Mode	CCP3CON Configuration	RD1 or RG0(1)	RE4	RE3	RD2 or RG3(1)	RH5(2)	RH4(2)
64-Pin Devices; 80-Pin Devices, ECCPMX = 1; 100-Pin Devices, ECCPMX = 1, Microcontroller mode:							
Compatible CCP	00xx 11xx	ECCP3	RE4	RE3	RD2/RG3	RH5/AN13	RH4/AN12
Dual PWM	10xx 11xx	P3A	P3B	RE3	RD2/RG3	RH5/AN13	RH4/AN12
Quad PWM	x1xx 11xx	P3A	P3B	P3C	P3D	RH5/AN13	RH4/AN12
80-Pin Devices, ECCPMX = 0; 100-Pin Devices, ECCPMX = 0, All Program Memory modes:							
Compatible CCP	00xx 11xx	ECCP3	RE6/AD14	RE5/AD13	RD2/RG3	RH5/AN13	RH4/AN12
Dual PWM	10xx 11xx	P3A	RE6/AD14	RE5/AD13	RD2/RG3	P3B	RH4/AN12
Quad PWM(3)	x1xx 11xx	P3A	RE6/AD14	RE5/AD13	P3D	P3B	P3C
100-Pin Devices, ECCPMX = 1, Extended Microcontroller with 12-Bit Address Width:							
Compatible CCP	00xx 11xx	ECCP3	RE4/AD12	RE3/AD11	RD2/RG3	RH5/AN13	RH4/AN12
Dual PWM	10xx 11xx	P3A	P3B	RE3/AD11	RD2/RG3	RH5/AN13	RH4/AN12
100-Pin Devices, ECCPMX = 1, Extended Microcontroller mode with 16-Bit or 20-Bit Address Width:							
Compatible CCP	00xx 11xx	ECCP3	RE6/AD14	RE5/AD13	RD2/RG3	RH5/AN13	RH4/AN12

Legend: x = Don't care. Shaded cells indicate pin assignments not used by ECCP3 in a given mode.

Note 1: ECCP3/P3A and CCP4/P3D are multiplexed with RD1 and RD2 in 64-pin devices, and RG0 and RG3 in 80-pin and 100-pin devices.

2: These pin options are not available in 64-pin devices.

3: With ECCP3 in Quad PWM mode, the CCP4 pin's output is overridden by P3D; otherwise, CCP4 is fully operational.

18.4.7.1 Auto-Shutdown and Automatic Restart

The auto-shutdown feature can be configured to allow automatic restarts of the module following a shutdown event. This is enabled by setting the P1RSEN bit of the ECCP1DEL register (ECCP1DEL<7>).

In Shutdown mode with P1RSEN = `1` (Figure 18-10), the ECCP1ASE bit will remain set for as long as the cause of the shutdown continues. When the shutdown condition clears, the ECCP1ASE bit is cleared. If P1RSEN = `0` (Figure 18-11), once a shutdown condition occurs, the ECCP1ASE bit will remain set until it is cleared by firmware. Once ECCP1ASE is cleared, the Enhanced PWM will resume at the beginning of the next PWM period.

Note: Writing to the ECCP1ASE bit is disabled while a shutdown condition is active.

Independent of the P1RSEN bit setting, if the auto-shutdown source is one of the comparators, the shutdown condition is a level. The ECCP1ASE bit cannot be cleared as long as the cause of the shutdown persists.

The Auto-Shutdown mode can be forced by writing a ‘`1`’ to the ECCP1ASE bit.

18.4.8 START-UP CONSIDERATIONS

When the ECCP1 module is used in the PWM mode, the application hardware must use the proper external pull-up and/or pull-down resistors on the PWM output pins. When the microcontroller is released from Reset, all of the I/O pins are in the high-impedance state. The external circuits must keep the power switch devices in the OFF state until the microcontroller drives the I/O pins with the proper signal levels, or activates the PWM output(s).

The CCP1M<1:0> bits (CCP1CON<1:0>) allow the user to choose whether the PWM output signals are active-high or active-low for each pair of PWM output pins (P1A/P1C and P1B/P1D). The PWM output polarities must be selected before the PWM pins are configured as outputs. Changing the polarity configuration while the PWM pins are configured as outputs is not recommended since it may result in damage to the application circuits.

The P1A, P1B, P1C and P1D output latches may not be in the proper states when the PWM module is initialized. Enabling the PWM pins for output at the same time as the ECCP1 module may cause damage to the application circuit. The ECCP1 module must be enabled in the proper Output mode and complete a full PWM cycle before configuring the PWM pins as outputs. The completion of a full PWM cycle is indicated by the TMR2IF bit being set as the second PWM period begins.

FIGURE 18-10: PWM AUTO-SHUTDOWN (P1RSEN = 1, AUTO-RESTART ENABLED)

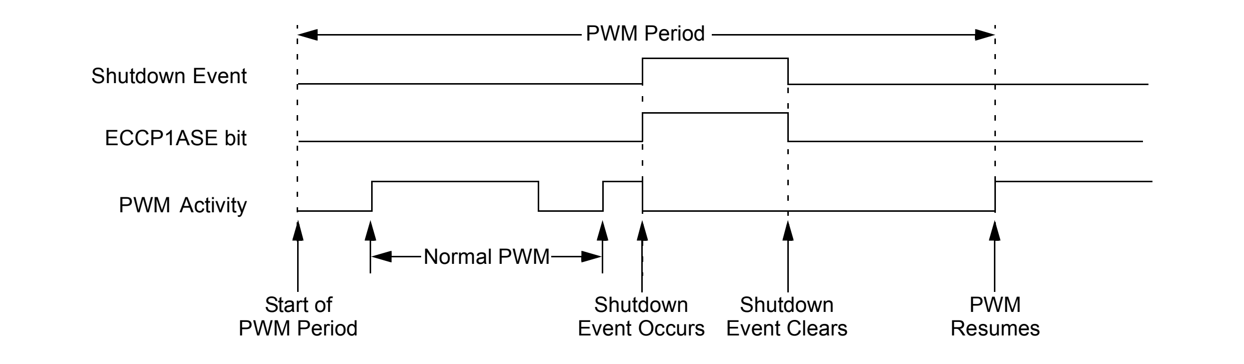
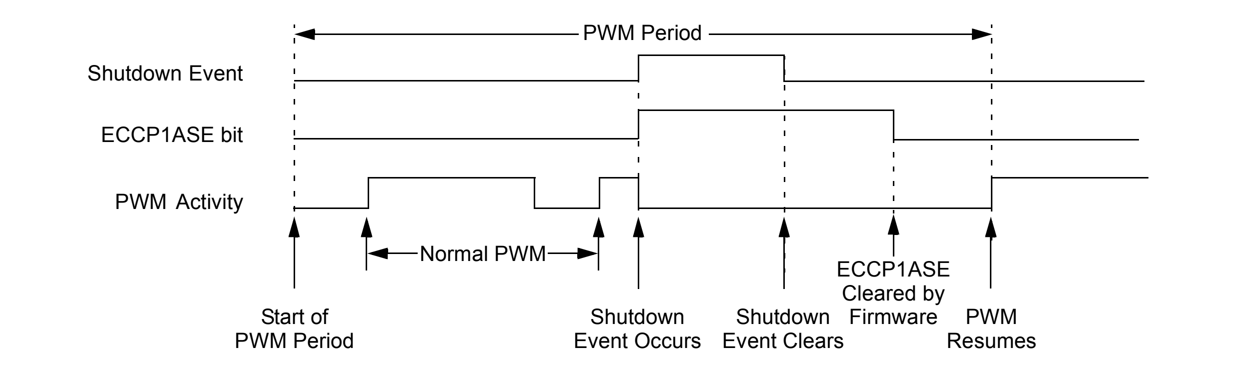


FIGURE 18-11: PWM AUTO-SHUTDOWN (P1RSEN = `0`, AUTO-RESTART DISABLED)



REGISTER 19-15: EIR: ETHERNET INTERRUPT REQUEST (FLAG) REGISTER

U-0	R-0	R/C-0	R-0	R/C-0	U-0	R/C-0	R/C-0
—	PKTIF	DMAIF	LINKIF	TXIF	—	TXERIF	RXERIF
bit 7							bit 0

Legend:			
R = Readable bit	C = Clearable 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 **PKTIF:** Receive Packet Pending Interrupt Flag bit
1 = Receive buffer contains one or more unprocessed packets; cleared only when EPKTCNT is decremented to 0 by setting PKTDEC (ECON2<6>)
0 = Receive buffer is empty

bit 5 **DMAIF:** DMA Interrupt Flag bit
1 = DMA copy or checksum calculation has completed
0 = No DMA interrupt is pending

bit 4 **LINKIF:** Link Change Interrupt Flag bit
1 = PHY reports that the link status has changed; read PHIR register to clear
0 = Link status has not changed

bit 3 **TXIF:** Transmit Interrupt Flag bit
1 = Transmit request has ended
0 = No transmit interrupt is pending

bit 2 **Unimplemented:** Read as ‘0’

bit 1 **TXERIF:** Transmit Error Interrupt Flag bit
1 = A transmit error has occurred
0 = No transmit error has occurred

bit 0 **RXERIF:** Receive Error Interrupt Flag bit
1 = A packet was aborted because there is insufficient buffer space, or a buffer overrun has occurred
0 = No receive error interrupt is pending

REGISTER 19-18: MABBIPG: MAC BACK-TO-BACK INTER-PACKET GAP REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	BBIPG6	BBIPG5	BBIPG4	BBIPG3	BBIPG2	BBIPG1	BBIPG0
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 6-0 **BBIPG<6:0>:** Back-to-Back Inter-Packet Gap Delay Time bits

When FULDPX (MACON3<0>) = 1:
Nibble time offset delay between the end of one transmission and the beginning of the next in a back-to-back sequence. The register value should be programmed to the desired period in nibble times minus 3. The recommended setting is 15h which represents the minimum IEEE specified Inter-Packet Gap (IPG) of 9.6 μs.

When FULDPX (MACON3<0>) = 0:
Nibble time offset delay between the end of one transmission and the beginning of the next in a back-to-back sequence. The register value should be programmed to the desired period in nibble times minus 6. The recommended setting is 12h which represents the minimum IEEE specified Inter-Packet Gap (IPG) of 9.6 μs.

19.4.6 PHY INITIALIZATION SETTINGS

Depending on the application, bits in three of the PHY module’s registers may also require configuration.

The PDPXMD bit (PHCON1<8>) controls the PHY half/full-duplex configuration. The application must program the bit properly, along with the FULDPX bit (MACON3<0>).

The HDLDIS bit (PHCON2<8>) disables automatic loopback of data. For proper operation, always set both HDLDIS and RXAPDIS (PHCON2<4>).

The PHY register, PHLCON (Register 19-13), controls the outputs of LEDA and LEDB. If an application requires a LED configuration other than the default, alter this register to match the new requirements. The settings for LED operation are discussed in **Section 19.1.2 “LED Configuration”**.

19.4.7 DISABLING THE ETHERNET MODULE

There may be circumstances during which the Ethernet module is not needed for prolonged periods. For example, in situations where the application only needs to transmit or receive Ethernet packets on the occurrence of a particular event. In these cases, the module can be selectively powered down.

To selectively disable the module:

1. Turn off packet reception by clearing the RXEN bit.
2. Wait for any in-progress packets to finish being received by polling the RXBUSY bit (ESTAT<2>). This bit should be clear before proceeding.
3. Wait for any current transmissions to end by confirming that the TXRTS bit (ECON1<3>) is clear.
4. Clear the ETHEN bit. This removes power and clock sources from the module, and makes the PHY registers inaccessible. The PHYRDY bit is also cleared automatically.

REGISTER 19-20: ERXFCON: ETHERNET RECEIVE FILTER CONTROL REGISTER

R/W-1	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1
UCEN	ANDOR	CRCEN	PMEN	MPEN	HTEN	MCEN	BCEN
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 **UCEN:** Unicast Filter Enable bit
When ANDOR = 1:
1 = Packets not having a destination address matching the local MAC address will be discarded
0 = Filter is disabled
When ANDOR = 0:
1 = Packets with a destination address matching the local MAC address will be accepted
0 = Filter is disabled

bit 6 **ANDOR:** AND/OR Filter Select bit
1 = AND: Packets will be rejected unless all enabled filters accept the packet
0 = OR: Packets will be accepted unless all enabled filters reject the packet

bit 5 **CRCEN:** Post-Filter CRC Check Enable bit
1 = All packets with an invalid CRC will be discarded
0 = The CRC validity will be ignored

bit 4 **PMEN:** Pattern Match Filter Enable bit
When ANDOR = 1:
1 = Packets must meet the Pattern Match criteria or they will be discarded
0 = Filter is disabled
When ANDOR = 0:
1 = Packets which meet the Pattern Match criteria will be accepted
0 = Filter is disabled

bit 3 **MPEN:** Magic Packet Filter Enable bit
When ANDOR = 1:
1 = Packets must be Magic Packets for the local MAC address or they will be discarded
0 = Filter is disabled
When ANDOR = 0:
1 = Magic Packets for the local MAC address will be accepted
0 = Filter is disabled

bit 2 **HTEN:** Hash Table Filter Enable bit
When ANDOR = 1:
1 = Packets must meet the Hash Table criteria or they will be discarded
0 = Filter is disabled
When ANDOR = 0:
1 = Packets which meet the Hash Table criteria will be accepted
0 = Filter is disabled

bit 1 **MCEN:** Multicast Filter Enable bit
When ANDOR = 1:
1 = The LSb of the first byte of the packet's destination address must be set or it will be discarded
0 = Filter is disabled
When ANDOR = 0:
1 = Packets which have the LSb of the first byte in the destination address set will be accepted
0 = Filter is disabled

bit 0 **BCEN:** Broadcast Filter Enable bit
When ANDOR = 1:
1 = Packets must have a destination address of FF-FF-FF-FF-FF-FF or they will be discarded
0 = Filter is disabled
When ANDOR = 0:
1 = Packets which have a destination address of FF-FF-FF-FF-FF-FF will be accepted
0 = Filter is disabled

20.4.10 I^{2}C MASTER MODE TRANSMISSION

Transmission of a data byte, a 7-bit address or the other half of a 10-bit address is accomplished by simply writing a value to the SSPxBUF register. This action will set the Buffer Full flag bit, BF, and allow the Baud Rate Generator to begin counting and start the next transmission. Each bit of address/data will be shifted out onto the SDAx pin after the falling edge of SCLx is asserted (see data hold time specification Parameter 106). SCLx is held low for one Baud Rate Generator rollover count (TBRG). Data should be valid before SCLx is released high (see data setup time specification Parameter 107). When the SCLx pin is released high, it is held that way for TBRG. The data on the SDAx pin must remain stable for that duration and some hold time after the next falling edge of SCLx. After the eighth bit is shifted out (the falling edge of the eighth clock), the BF flag is cleared and the master releases SDAx. This allows the slave device being addressed to respond with an $\overline{\text{ACK}}$ bit during the ninth bit time if an address match occurred, or if data was received properly. The status of $\overline{\text{ACK}}$ is written into the ACKDT bit on the falling edge of the ninth clock. If the master receives an Acknowledge, the Acknowledge Status bit, ACKSTAT, is cleared; if not, the bit is set. After the ninth clock, the SSPxIF bit is set and the master clock (Baud Rate Generator) is suspended until the next data byte is loaded into the SSPxBUF, leaving SCLx low and SDAx unchanged (Figure 20-23).

After the write to the SSPxBUF, each bit of the address will be shifted out on the falling edge of SCLx until all seven address bits and the R/$\overline{\text{W}}$ bit are completed. On the falling edge of the eighth clock, the master will deassert the SDAx pin, allowing the slave to respond with an Acknowledge. On the falling edge of the ninth clock, the master will sample the SDAx pin to see if the address was recognized by a slave. The status of the $\overline{\text{ACK}}$ bit is loaded into the ACKSTAT status bit (SSPxCON2<6>). Following the falling edge of the ninth clock transmission of the address, the SSPxIF is set, the BF flag is cleared and the Baud Rate Generator is turned off until another write to the SSPxBUF takes place, holding SCLx low and allowing SDAx to float.

20.4.10.1 BF Status Flag

In Transmit mode, the BF bit (SSPxSTAT<0>) is set when the CPU writes to SSPxBUF, and is cleared when all 8 bits are shifted out.

20.4.10.2 WCOL Status Flag

If the user writes to the SSPxBUF when a transmit is already in progress (i.e., SSPxSR is still shifting out a data byte), the WCOL is set and the contents of the buffer are unchanged (the write doesn't occur) after 2 TCY after the SSPxBUF write. If SSPxBUF is rewritten within 2 TCY, the WCOL bit is set and SSPxBUF is updated. This may result in a corrupted transfer.

The user should verify that the WCOL is clear after each write to SSPxBUF to ensure the transfer is correct. In all cases, WCOL must be cleared in software.

20.4.10.3 ACKSTAT Status Flag

In Transmit mode, the ACKSTAT bit (SSPxCON2<6>) is cleared when the slave has sent an Acknowledge ($\overline{\text{ACK}}$ = 0) and is set when the slave does not Acknowledge ($\overline{\text{ACK}}$ = 1). A slave sends an Acknowledge when it has recognized its address (including a general call), or when the slave has properly received its data.

20.4.11 I^{2}C MASTER MODE RECEPTION

Master mode reception is enabled by programming the Receive Enable bit, RCEN (SSPxCON2<3>).

Note: The MSSP module must be in an Idle state before the RCEN bit is set or the RCEN bit will be disregarded.

The Baud Rate Generator begins counting and on each rollover. The state of the SCLx pin changes (high-to-low/low-to-high) and data is shifted into the SSPxSR. After the falling edge of the eighth clock, the receive enable flag is automatically cleared, the contents of the SSPxSR are loaded into the SSPxBUF, the BF flag bit is set, the SSPxIF flag bit is set and the Baud Rate Generator is suspended from counting, holding SCLx low. The MSSP is now in Idle state awaiting the next command. When the buffer is read by the CPU, the BF flag bit is automatically cleared. The user can then send an Acknowledge bit at the end of reception by setting the Acknowledge Sequence Enable bit, ACKEN (SSPxCON2<4>).

20.4.11.1 BF Status Flag

In receive operation, the BF bit is set when an address or data byte is loaded into SSPxBUF from SSPxSR. It is cleared when the SSPxBUF register is read.

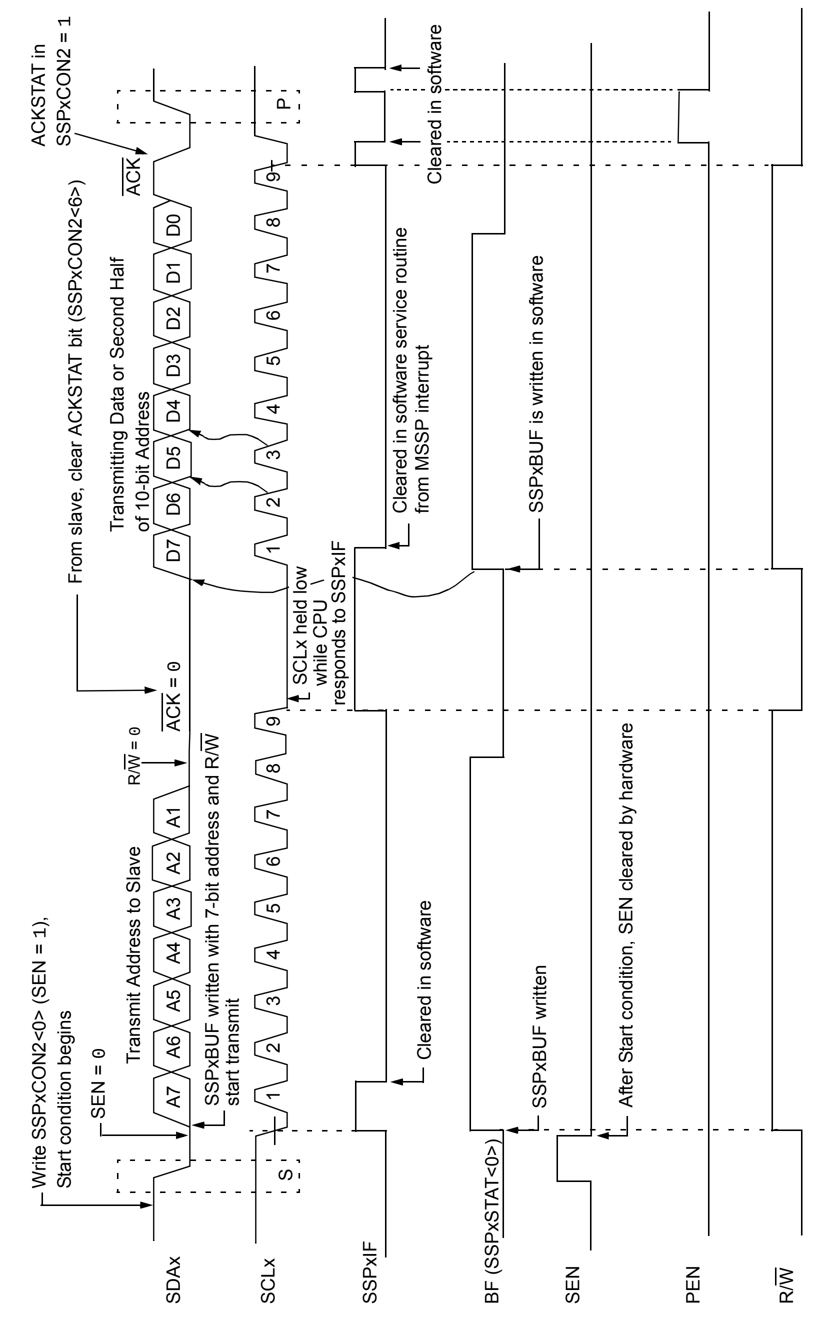
20.4.11.2 SSPOV Status Flag

In receive operation, the SSPOV bit is set when 8 bits are received into the SSPxSR and the BF flag bit is already set from a previous reception.

20.4.11.3 WCOL Status Flag

If the user writes the SSPxBUF when a receive is already in progress (i.e., SSPxSR is still shifting in a data byte), the WCOL bit is set and the contents of the buffer are unchanged (the write doesn't occur).

FIGURE 20-23: I²C™ MASTER MODE WAVEFORM (TRANSMISSION, 7 OR 10-BIT ADDRESS)



REGISTER 25-6: CONFIG3H: CONFIGURATION REGISTER 3 HIGH (BYTE ADDRESS 300005h)

U-0	U-0	U-0	U-0	U-0	R/WO-1	R/WO-1	R/WO-1
—[1]	—[1]	—[1]	—[1]	—	ETHLED	ECCPMX[2]	CCP2MX[2]
bit 7							bit 0

Legend:			
R = Readable bit	WO = Write-Once bit	U = Unimplemented bit, read as ‘0’	
-n = Value when device is unprogrammed		‘1’ = Bit is set	‘0’ = Bit is cleared

bit 7-3 **Unimplemented:** Read as ‘0’

bit 2 **ETHLED:** Ethernet LED Enable bit

1 = RA0/RA1 are multiplexed with LEDA/LEDB when the Ethernet module is enabled and function as I/O when the Ethernet is disabled
0 = RA0/RA1 function as I/O regardless of Ethernet module status

bit 1 **ECCPMX:** ECCP MUX bit[2]

1 = ECCP1 outputs (P1B/P1C) are multiplexed with RE6 and RE5;
ECCP3 outputs (P3B/P3C) are multiplexed with RE4 and RE3
0 = ECCP1 outputs (P1B/P1C) are multiplexed with RH7 and RH6;
ECCP3 outputs (P3B/P3C) are multiplexed with RH5 and RH4

bit 0 **CCP2MX:** ECCP2 MUX bit[2]

1 = ECCP2/P2A is multiplexed with RC1
0 = ECCP2/P2A is multiplexed with RE7 in Microcontroller mode (80-pin and 100-pin devices) or with RB3 in Extended Microcontroller mode (100-pin devices only)

Note 1: The value of these bits in program memory should always be ‘1’. This ensures that the location is executed as a NOP if it is accidentally executed.

2: Implemented in 80-pin and 100-pin devices only.

IORLW — Inclusive OR Literal with W

Syntax: IORLW k

Operands: $0 \leq k \leq 255$

Operation: (W) .OR. k → W

Status Affected: N, Z

Encoding:

`0000`	`1001`	`kkkk`	`kkkk`

Description: The contents of W are ORed with the eight-bit literal 'k'. The result is placed in W.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	Write to W

Example: `IORLW 35h`

Before Instruction
W = 9Ah
After Instruction
W = BFh

IORWF — Inclusive OR W with f

Syntax: IORWF f {,d {,a}}

Operands: $0 \leq f \leq 255$
$d \in [0,1]$
$a \in [0,1]$

Operation: (W) .OR. (f) → dest

Status Affected: N, Z

Encoding:

`0001`	`00da`	`ffff`	`ffff`

Description: Inclusive OR W with register 'f'. If 'd' is '`0`', the result is placed in W. If 'd' is '`1`', the result is placed 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 $f \leq 95$ (5Fh). See **Section 26.2.3 "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 to destination

Example: `IORWF RESULT, 0, 1`

Before Instruction
RESULT = 13h
W = 91h
After Instruction
RESULT = 13h
W = 93h

MOVSS — Move Indexed to Indexed

Syntax: MOVSS [z_s], [z_d]

Operands: $0 \le z_s \le 127$
$0 \le z_d \le 127$

Operation: ((FSR2) + z_s) → ((FSR2) + z_d)

Status Affected: None

Encoding:				
1st word (source)	`1110`	`1011`	`1zzz`	$\texttt{zzzz}_s$
2nd word (dest.)	`1111`	`xxxx`	`xzzz`	$\texttt{zzzz}_d$

Description: The contents of the source register are moved to the destination register. The addresses of the source and destination registers are determined by adding the 7-bit literal offsets 'z_s' or 'z_d', respectively, to the value of FSR2. Both registers can be located anywhere in the 4096-byte data memory space (000h to FFFh).

The `MOVSS` instruction cannot use the PCL, TOSU, TOSH or TOSL as the destination register.

If the resultant source address points to an indirect addressing register, the value returned will be 00h. If the resultant destination address points to an indirect addressing register, the instruction will execute as a `NOP`.

Words: 2

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Determine source addr	Determine source addr	Read source reg
Decode	Determine dest addr	Determine dest addr	Write to dest reg

Example: `MOVSS [05h], [06h]`

Before Instruction

FSR2	=	80h
Contents of 85h	=	33h
Contents of 86h	=	11h

After Instruction

FSR2	=	80h
Contents of 85h	=	33h
Contents of 86h	=	33h

PUSHL — Store Literal at FSR2, Decrement FSR2

Syntax: PUSHL k

Operands: $0 \le k \le 255$

Operation: k → (FSR2),
FSR2 – 1 → FSR2

Status Affected: None

Encoding:				
	`1110`	`1010`	`kkkk`	`kkkk`

Description: The 8-bit literal 'k' is written to the data memory address specified by FSR2. FSR2 is decremented by 1 after the operation.

This instruction allows users to push values onto a software stack.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read 'k'	Process data	Write to destination

Example: `PUSHL 08h`

Before Instruction

FSR2H:FSR2L	=	01ECh
Memory (01ECh)	=	00h

After Instruction

FSR2H:FSR2L	=	01EBh
Memory (01ECh)	=	08h

ADDWF — ADD W to Indexed (Indexed Literal Offset mode)

Syntax: ADDWF [k] {,d}

Operands: $0 \le k \le 95$
$d \in [0,1]$

Operation: (W) + ((FSR2) + k) → dest

Status Affected: N, OV, C, DC, Z

Encoding:

0010	01d0	kkkk	kkkk

Description: The contents of W are added to the contents of the register indicated by FSR2, offset by the value ‘k’.

If ‘d’ is ‘0’, the result is stored in W. If ‘d’ is ‘1’, the result is stored back in register ‘f’ (default).

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read ‘k’	Process Data	Write to destination

Example: `ADDWF [OFST] ,0`

Before Instruction
W = 17h
OFST = 2Ch
FSR2 = 0A00h
Contents of 0A2Ch = 20h

After Instruction
W = 37h
Contents of 0A2Ch = 20h

BSF — Bit Set Indexed (Indexed Literal Offset mode)

Syntax: BSF [k], b

Operands: $0 \le f \le 95$
$0 \le b \le 7$

Operation: 1 → ((FSR2) + k)<b>

Status Affected: None

Encoding:

1000	bbb0	kkkk	kkkk

Description: Bit ‘b’ of the register indicated by FSR2, offset by the value ‘k’, is set.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example: `BSF [FLAG_OFST], 7`

Before Instruction
FLAG_OFST = 0Ah
FSR2 = 0A00h
Contents of 0A0Ah = 55h

After Instruction
Contents of 0A0Ah = D5h

SETF — Set Indexed (Indexed Literal Offset mode)

Syntax: SETF [k]

Operands: $0 \le k \le 95$

Operation: FFh → ((FSR2) + k)

Status Affected: None

Encoding:

0110	1000	kkkk	kkkk

Description: The contents of the register indicated by FSR2, offset by ‘k’, are set to FFh.

Words: 1

Cycles: 1

Q Cycle Activity:

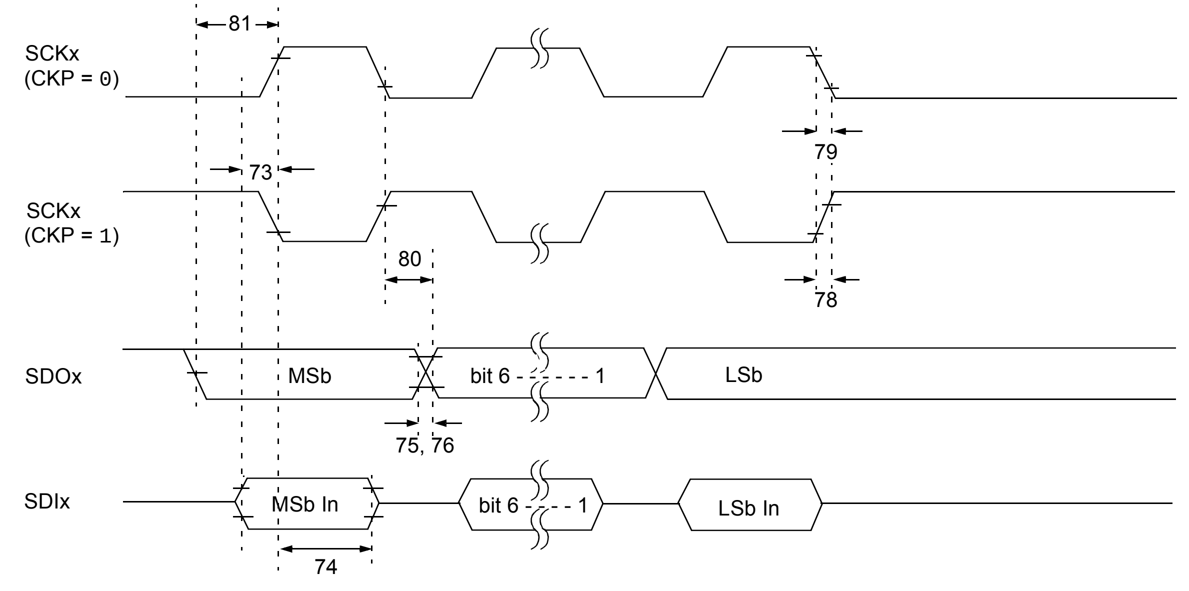
Q1	Q2	Q3	Q4
Decode	Read ‘k’	Process Data	Write register

Example: `SETF [OFST]`

Before Instruction
OFST = 2Ch
FSR2 = 0A00h
Contents of 0A2Ch = 00h

After Instruction
Contents of 0A2Ch = FFh

FIGURE 28-12: EXAMPLE SPI MASTER MODE TIMING (CKE = 1)



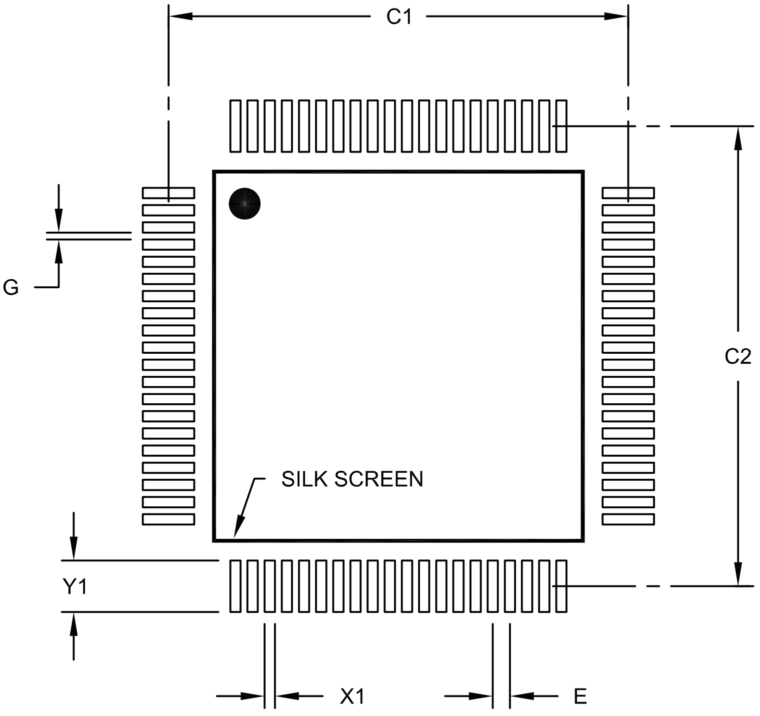
Note: Refer to Figure 28-3 for load conditions.

TABLE 28-17: EXAMPLE SPI MODE REQUIREMENTS (MASTER MODE, CKE = 1)

Param. No.	Symbol	Characteristic	Min	Max	Units	Conditions
73	TDIV2SCH, TDIV2SCL	Setup Time of SDIx Data Input to SCKx Edge	100	—	ns	
74	TSCH2DIL, TSCL2DIL	Hold Time of SDIx Data Input to SCKx Edge	100	—	ns	
75	TDOR	SDOx Data Output Rise Time	—	25	ns	
76	TDOF	SDOx Data Output Fall Time	—	25	ns	
78	TSCR	SCKx Output Rise Time	—	25	ns	
79	TSCF	SCKx Output Fall Time	—	25	ns	
80	TSCH2DOV, TSCL2DOV	SDOx Data Output Valid after SCKx Edge	—	50	ns	
81	TDOV2SCH, TDOV2SCL	SDOx Data Output Setup to SCKx Edge	TCY	—	ns	

80-Lead Plastic Thin Quad Flatpack (PT) – 12x12x1 mm Body, 2.00 mm [TQFP]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Contact Pad Spacing	C1		13.40	
Contact Pad Spacing	C2		13.40	
Contact Pad Width (X80)	X1			0.30
Contact Pad Length (X80)	Y1			1.50
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2092A