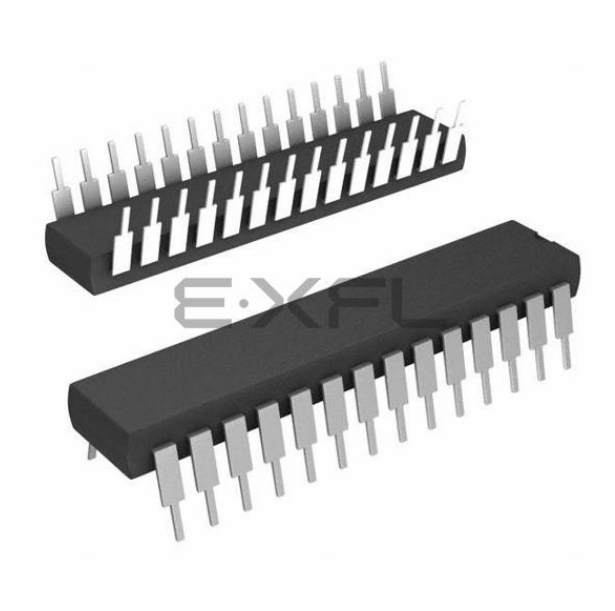


Welcome to E-XFL.COM

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	40MHz
Connectivity	CANbus, I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	22
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)	4.2V ~ 5.5V
Data Converters	A/D 5x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.300", 7.62mm)
Supplier Device Package	28-SPDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f248-i-sp

2.6 Oscillator Switching Feature

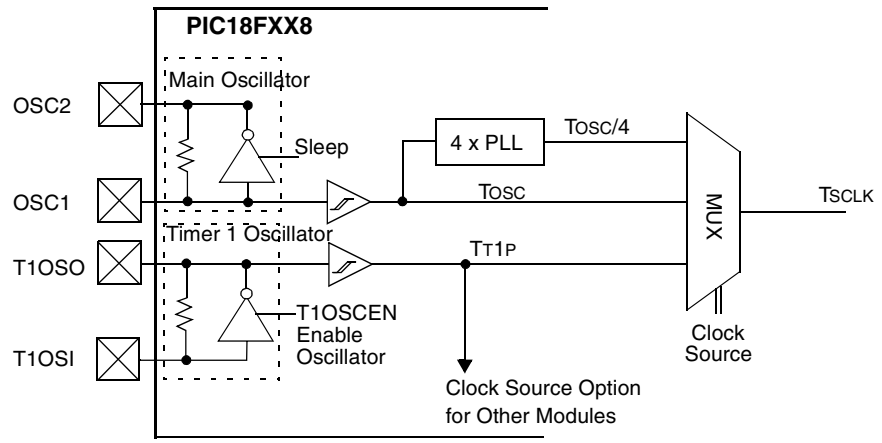
The PIC18FXX8 devices include a feature that allows the system clock source to be switched from the main oscillator to an alternate low-frequency clock source. For the PIC18FXX8 devices, this alternate clock source is the Timer1 oscillator. If a low-frequency crystal (32 kHz, for example) has been attached to the Timer1 oscillator pins and the Timer1 oscillator has been enabled, the device can switch to a Low-Power Execution mode. Figure 2-6 shows a block diagram of the system clock sources. The clock switching feature is enabled by programming the Oscillator Switching Enable ($\overline{\text{OSCSEN}}$) bit in Configuration register, CONFIG1H, to a '0'. Clock switching is disabled in an erased device. See **Section 12.2 "Timer1 Oscillator"** for further details of the Timer1 oscillator and **Section 24.1 "Configuration Bits"** for Configuration register details.

2.6.1 SYSTEM CLOCK SWITCH BIT

The system clock source switching is performed under software control. The system clock switch bit, SCS (OSCCON register), controls the clock switching. When the SCS bit is '0', the system clock source comes from the main oscillator selected by the FOSC2:FOSC0 configuration bits. When the SCS bit is set, the system clock source comes from the Timer1 oscillator. The SCS bit is cleared on all forms of Reset.

Note:	The Timer1 oscillator must be enabled to switch the system clock source. The Timer1 oscillator is enabled by setting the T1OSCEN bit in the Timer1 Control register (T1CON). If the Timer1 oscillator is not enabled, any write to the SCS bit will be ignored (SCS bit forced cleared) and the main oscillator continues to be the system clock source.

FIGURE 2-6: DEVICE CLOCK SOURCES



Note: I/O pins have diode protection to VDD and VSS.

REGISTER 2-1: OSCCON: OSCILLATOR CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-1
—	—	—	—	—	—	—	SCS
bit 7							bit 0

bit 7-1 **Unimplemented:** Read as '0'

bit 0 **SCS:** System Clock Switch bit

When $\overline{\text{OSCSEN}}$ configuration bit = 0 and T1OSCEN bit is set:

1 = Switch to Timer1 oscillator/clock pin

0 = Use primary oscillator/clock input pin

When $\overline{\text{OSCSEN}}$ is clear or T1OSCEN is clear:

Bit is forced clear.

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

NOTES:

FIGURE 3-3: TIME-OUT SEQUENCE ON POWER-UP (MCLR TIED TO VDD)

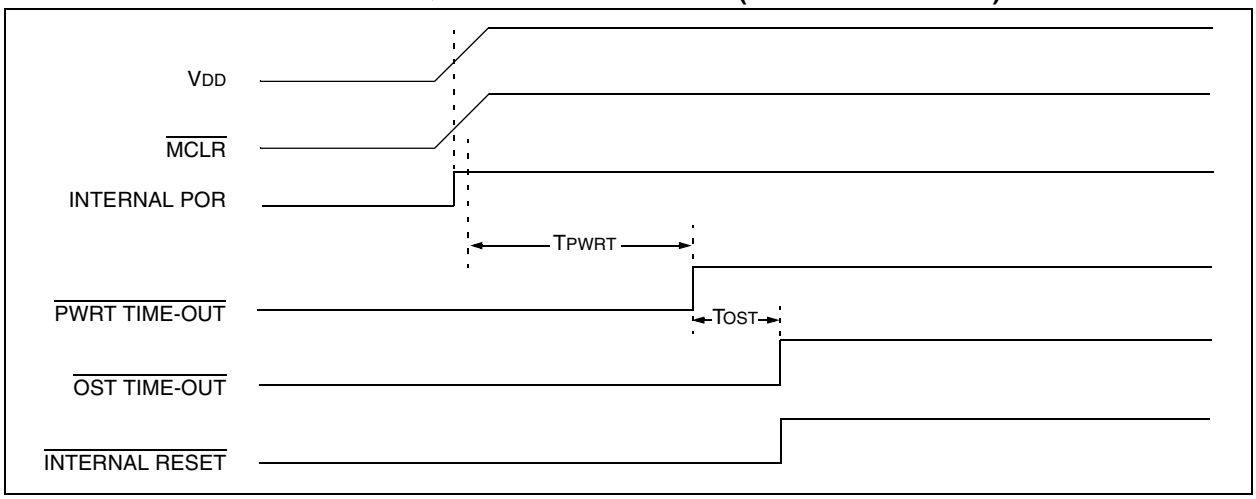


FIGURE 3-4: TIME-OUT SEQUENCE ON POWER-UP (MCLR NOT TIED TO VDD): CASE 1

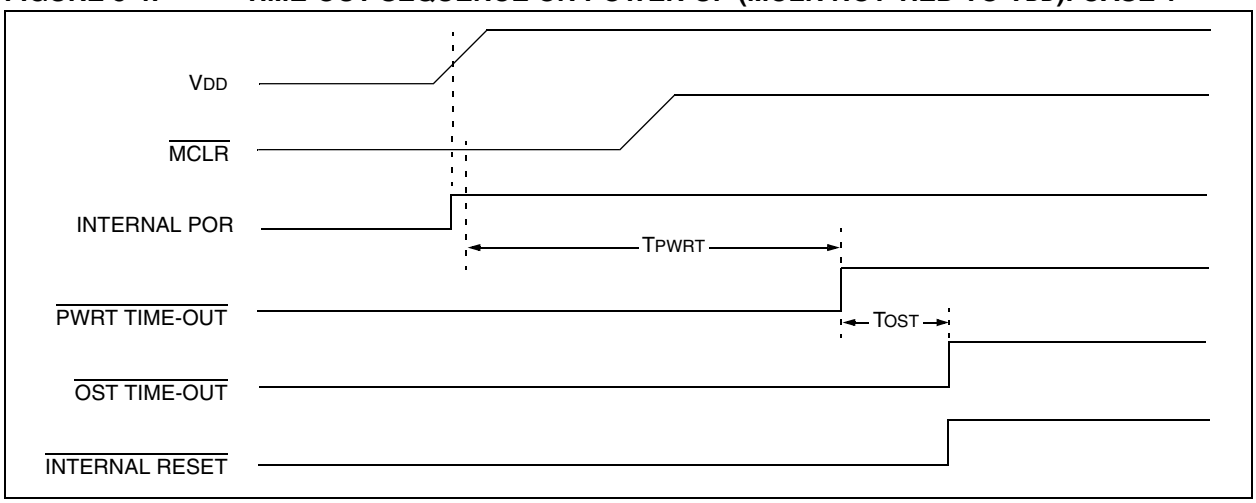


FIGURE 3-5: TIME-OUT SEQUENCE ON POWER-UP (MCLR NOT TIED TO VDD): CASE 2

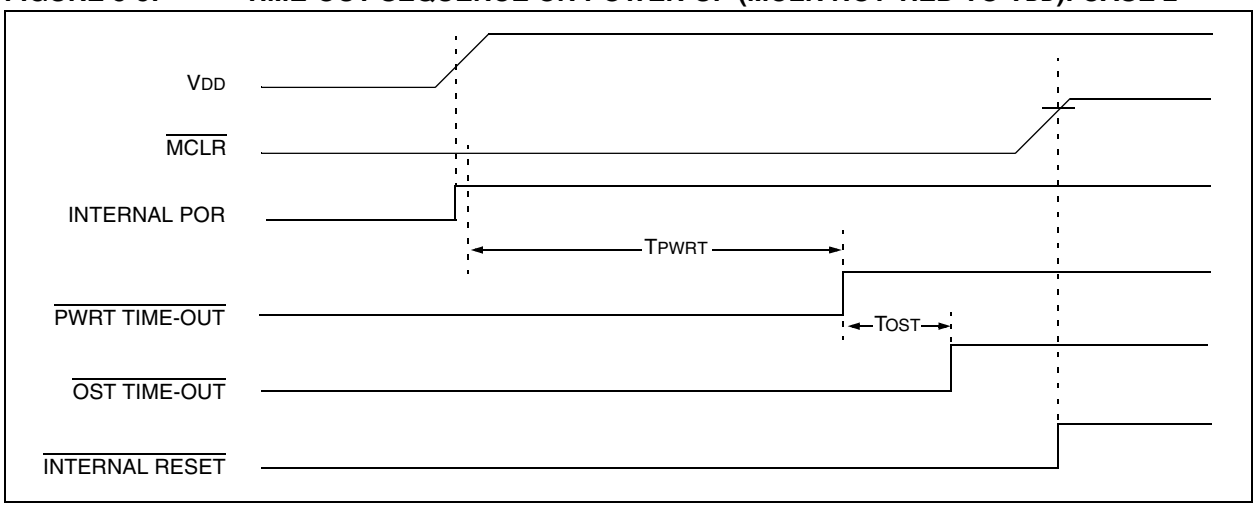


TABLE 3-3: INITIALIZATION CONDITIONS FOR ALL REGISTERS

Register	Applicable Devices		Power-on Reset, Brown-out Reset	MCLR Reset WDT Reset RESET Instruction Stack Resets	Wake-up via WDT or Interrupt
TOSU	PIC18F2X8	PIC18F4X8	---0 0000	---0 0000	---0 uuuu(3)
TOSH	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu(3)
TOSL	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu(3)
STKPTR	PIC18F2X8	PIC18F4X8	00-0 0000	uu-0 0000	uu-u uuuu(3)
PCLATU	PIC18F2X8	PIC18F4X8	---0 0000	---0 0000	---u uuuu
PCLATH	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu
PCL	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	PC + 2(2)
TBLPTRU	PIC18F2X8	PIC18F4X8	--00 0000	--00 0000	--uu uuuu
TBLPTRH	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu
TBLPTRL	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu
TABLAT	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu
PRODH	PIC18F2X8	PIC18F4X8	xxxx xxxx	uuuu uuuu	uuuu uuuu
PRODL	PIC18F2X8	PIC18F4X8	xxxx xxxx	uuuu uuuu	uuuu uuuu
INTCON	PIC18F2X8	PIC18F4X8	0000 000x	0000 000u	uuuu uuuu(1)
INTCON2	PIC18F2X8	PIC18F4X8	111- -1-1	111- -1-1	uuu- -u-u(1)
INTCON3	PIC18F2X8	PIC18F4X8	11-0 0-00	11-0 0-00	uu-u u-uu(1)
INDF0	PIC18F2X8	PIC18F4X8	N/A	N/A	N/A
POSTINC0	PIC18F2X8	PIC18F4X8	N/A	N/A	N/A
POSTDEC0	PIC18F2X8	PIC18F4X8	N/A	N/A	N/A
PREINC0	PIC18F2X8	PIC18F4X8	N/A	N/A	N/A
PLUSW0	PIC18F2X8	PIC18F4X8	N/A	N/A	N/A
FSR0H	PIC18F2X8	PIC18F4X8	---- xxxx	---- uuuu	---- uuuu
FSR0L	PIC18F2X8	PIC18F4X8	xxxx xxxx	uuuu uuuu	uuuu uuuu
WREG	PIC18F2X8	PIC18F4X8	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF1	PIC18F2X8	PIC18F4X8	N/A	N/A	N/A
POSTINC1	PIC18F2X8	PIC18F4X8	N/A	N/A	N/A
POSTDEC1	PIC18F2X8	PIC18F4X8	N/A	N/A	N/A
PREINC1	PIC18F2X8	PIC18F4X8	N/A	N/A	N/A
PLUSW1	PIC18F2X8	PIC18F4X8	N/A	N/A	N/A

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition. Shaded cells indicate conditions do not apply for the designated device.

Note 1: One or more bits in the INTCONx or PIRx registers will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the PC is loaded with the interrupt vector (0008h or 0018h).

3: When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the TOSU, TOSH and TOSL are updated with the current value of the PC. The STKPTR is modified to point to the next location in the hardware stack.

4: See Table 3-2 for Reset value for specific condition.

5: Bit 6 of PORTA, LATA and TRISA are enabled in ECIO and RCIO Oscillator modes only. In all other oscillator modes, they are disabled and read '0'.

6: Values for CANSTAT also apply to its other instances (CANSTATRO1 through CANSTATRO4).

TABLE 3-3: INITIALIZATION CONDITIONS FOR ALL REGISTERS (CONTINUED)

Register	Applicable Devices		Power-on Reset, Brown-out Reset	$\overline{\text{MCLR}}$ Reset WDT Reset RESET Instruction Stack Resets	Wake-up via WDT or Interrupt
ADCON1	PIC18F2X8	PIC18F4X8	00-- 0000	00-- 0000	uu-- uuuu
CCPR1H	PIC18F2X8	PIC18F4X8	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR1L	PIC18F2X8	PIC18F4X8	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP1CON	PIC18F2X8	PIC18F4X8	--00 0000	--00 0000	--uu uuuu
ECCPR1H	PIC18F2X8	PIC18F4X8	xxxx xxxx	uuuu uuuu	uuuu uuuu
ECCPR1L	PIC18F2X8	PIC18F4X8	xxxx xxxx	uuuu uuuu	uuuu uuuu
ECCP1CON	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	0000 0000
ECCP1DEL	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	0000 0000
ECCPAS	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	0000 0000
CVRCON	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu
CMCON	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu
TMR3H	PIC18F2X8	PIC18F4X8	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR3L	PIC18F2X8	PIC18F4X8	xxxx xxxx	uuuu uuuu	uuuu uuuu
T3CON	PIC18F2X8	PIC18F4X8	0000 0000	uuuu uuuu	uuuu uuuu
SPBRG	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu
RCREG	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu
TXREG	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu
TXSTA	PIC18F2X8	PIC18F4X8	0000 -010	0000 -010	uuuu -uuu
RCSTA	PIC18F2X8	PIC18F4X8	0000 000x	0000 000u	uuuu uuuu
EEADR	PIC18F2X8	PIC18F4X8	xxxx xxxx	uuuu uuuu	uuuu uuuu
EEDATA	PIC18F2X8	PIC18F4X8	xxxx xxxx	uuuu uuuu	uuuu uuuu
EECON2	PIC18F2X8	PIC18F4X8	xxxx xxxx	uuuu uuuu	uuuu uuuu
EECON1	PIC18F2X8	PIC18F4X8	xx-0 x000	uu-0 u000	uu-0 u000
IPR3	PIC18F2X8	PIC18F4X8	1111 1111	1111 1111	uuuu uuuu
PIR3	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu
PIE3	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu
IPR2	PIC18F2X8	PIC18F4X8	-1-1 1111	-1-1 1111	-u-u uuuu
PIR2	PIC18F2X8	PIC18F4X8	-0-0 0000	-0-0 0000	-u-u uuuu(1)
PIE2	PIC18F2X8	PIC18F4X8	-0-0 0000	-0-0 0000	-u-u uuuu
IPR1	PIC18F2X8	PIC18F4X8	1111 1111	1111 1111	uuuu uuuu
PIR1	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu(1)
PIE1	PIC18F2X8	PIC18F4X8	0000 0000	0000 0000	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition. Shaded cells indicate conditions do not apply for the designated device.

Note 1: One or more bits in the INTCONx or PIRx registers will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the PC is loaded with the interrupt vector (0008h or 0018h).

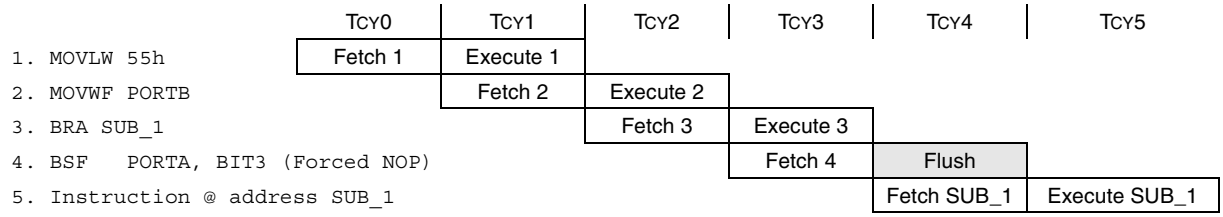
3: When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the TOSU, TOSH and TOSL are updated with the current value of the PC. The STKPTR is modified to point to the next location in the hardware stack.

4: See Table 3-2 for Reset value for specific condition.

5: Bit 6 of PORTA, LATA and TRISA are enabled in ECIO and RCIO Oscillator modes only. In all other oscillator modes, they are disabled and read '0'.

6: Values for CANSTAT also apply to its other instances (CANSTATRO1 through CANSTATRO4).

EXAMPLE 4-2: INSTRUCTION PIPELINE FLOW



Note: All instructions are single cycle, except for any program branches. These take two cycles, since the fetch instruction is "flushed" from the pipeline while the new instruction is being fetched and then executed.

EXAMPLE 4-3: INSTRUCTIONS IN PROGRAM MEMORY

Instruction	Opcode	Memory	Address
—			000007h
MOVLW 055h	0E55h	55h	000008h
		0Eh	000009h
GOTO 000006h	0EF03h, 0F000h	03h	00000Ah
		0EFh	00000Bh
		00h	00000Ch
		0F0h	00000Dh
MOVFF 123h, 456h	0C123h, 0F456h	23h	00000Eh
		0C1h	00000Fh
		56h	000010h
		0F4h	000011h
—			000012h

8.3 PIE Registers

The Peripheral Interrupt Enable (PIE) registers contain the individual enable bits for the peripheral interrupts (Register 8-7 through Register 8-9). Due to the number of peripheral interrupt sources, there are three Peripheral Interrupt Enable registers (PIE1, PIE2, PIE3). When IPEN is clear, the PEIE bit must be set to enable any of these peripheral interrupts.

REGISTER 8-7: PIE1: PERIPHERAL INTERRUPT ENABLE REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
bit 7							bit 0

bit 7 **PSPIE:** Parallel Slave Port Read/Write Interrupt Enable bit(1)
1 = Enables the PSP read/write interrupt
0 = Disables the PSP read/write interrupt

bit 6 **ADIE**: A/D Converter Interrupt Enable bit
1 = Enables the A/D interrupt
0 = Disables the A/D interrupt

bit 5 **RCIE**: USART Receive Interrupt Enable bit
1 = Enables the USART receive interrupt
0 = Disables the USART receive interrupt

bit 4 **TXIE**: USART Transmit Interrupt Enable bit
1 = Enables the USART transmit interrupt
0 = Disables the USART transmit interrupt

bit 3 **SSPIE**: Master Synchronous Serial Port Interrupt Enable bit
1 = Enables the MSSP interrupt
0 = Disables the MSSP interrupt

bit 2 **CCP1IE**: CCP1 Interrupt Enable bit
1 = Enables the CCP1 interrupt
0 = Disables the CCP1 interrupt

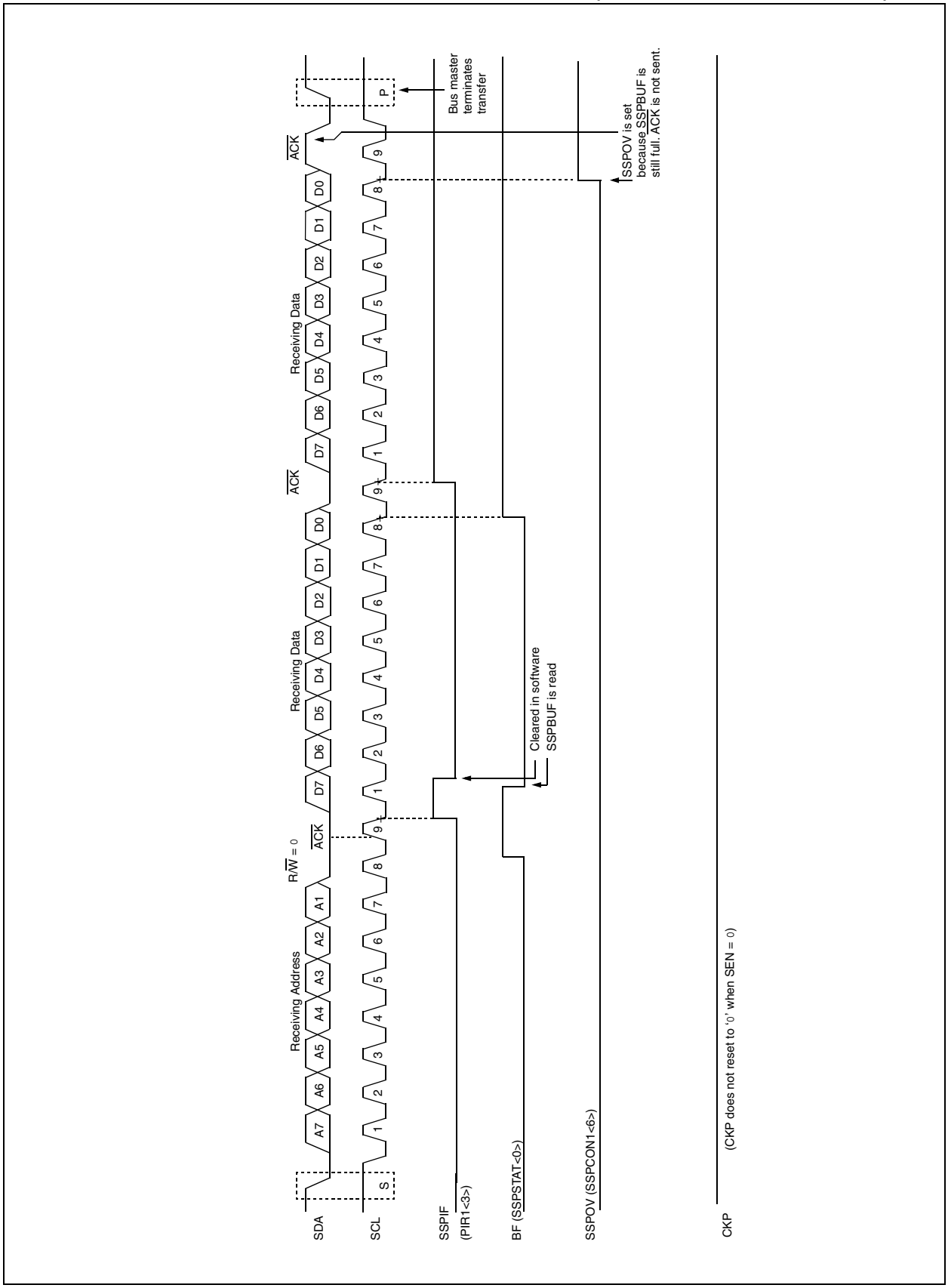
bit 1 **TMR2IE**: TMR2 to PR2 Match Interrupt Enable bit
1 = Enables the TMR2 to PR2 match interrupt
0 = Disables the TMR2 to PR2 match interrupt

bit 0 **TMR1IE**: TMR1 Overflow Interrupt Enable bit
1 = Enables the TMR1 overflow interrupt
0 = Disables the TMR1 overflow interrupt

Note 1: This bit is only available on PIC18F4X8 devices. For PIC18F2X8 devices, this bit is unimplemented and reads as '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

FIGURE 17-8: I²C™ SLAVE MODE TIMING WITH SEN = 0 (RECEPTION, 7-BIT ADDRESS)



REGISTER 19-3: COMSTAT: COMMUNICATION STATUS REGISTER

R/C-0	R/C-0	R-0	R-0	R-0	R-0	R-0	R-0
RXB0OVFL	RXB1OVFL	TXBO	TXBP	RXBP	TXWARN	RXWARN	EWARN
bit 7							bit 0

bit 7 **RXB0OVFL:** Receive Buffer 0 Overflow bit
1 = Receive Buffer 0 overflowed
0 = Receive Buffer 0 has not overflowed

bit 6 **RXB1OVFL:** Receive Buffer 1 Overflow bit
1 = Receive Buffer 1 overflowed
0 = Receive Buffer 1 has not overflowed

bit 5 **TXBO:** Transmitter Bus-Off bit
1 = Transmit Error Counter > 255
0 = Transmit Error Counter ≤ 255

bit 4 **TXBP:** Transmitter Bus Passive bit
1 = Transmission Error Counter > 127
0 = Transmission Error Counter ≤ 127

bit 3 **RXBP:** Receiver Bus Passive bit
1 = Receive Error Counter > 127
0 = Receive Error Counter ≤ 127

bit 2 **TXWARN:** Transmitter Warning bit
1 = 127 ≥ Transmit Error Counter > 95
0 = Transmit Error Counter ≤ 95

bit 1 **RXWARN:** Receiver Warning bit
1 = 127 ≥ Receive Error Counter > 95
0 = Receive Error Counter ≤ 95

bit 0 **EWARN:** Error Warning bit
This bit is a flag of the RXWARN and TXWARN bits.
1 = The RXWARN or the TXWARN bits are set
0 = Neither the RXWARN or the TXWARN bits are set

Legend:			
R = Readable bit	W = Writable 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

19.2.3.1 Message Acceptance Filters and Masks

This subsection describes the message acceptance filters and masks for the CAN receive buffers.

REGISTER 19-21: RXFnSIDH: RECEIVE ACCEPTANCE FILTER n STANDARD IDENTIFIER FILTER, HIGH BYTE REGISTERS

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 7							bit 0

bit 7-0 **SID10:SID3:** Standard Identifier Filter bits if EXIDEN = 0 or
Extended Identifier Filter bits EID28:EID21 if EXIDEN = 1

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

REGISTER 19-22: RXFnSIDL: RECEIVE ACCEPTANCE FILTER n STANDARD IDENTIFIER FILTER, LOW BYTE REGISTERS

R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
SID2	SID1	SID0	—	EXIDEN	—	EID17	EID16
bit 7							bit 0

bit 7-5 **SID2:SID0:** Standard Identifier Filter bits if EXIDEN = 0 or
Extended Identifier Filter bits EID20:EID18 if EXIDEN = 1

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

bit 3 **EXIDEN:** Extended Identifier Filter Enable bit
1 = Filter will only accept extended ID messages
0 = Filter will only accept standard ID messages

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

bit 1-0 **EID17:EID16:** Extended Identifier Filter bits

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

REGISTER 19-30: BRGCON2: BAUD RATE CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SEG2PHTS	SAM	SEG1PH2	SEG1PH1	SEG1PH0	PRSEG2	PRSEG1	PRSEG0
bit 7							bit 0

bit 7 **SEG2PHTS:** Phase Segment 2 Time Select bit
1 = Freely programmable
0 = Maximum of PHEG1 or Information Processing Time (IPT), whichever is greater

bit 6 **SAM:** Sample of the CAN bus Line bit
1 = Bus line is sampled three times prior to the sample point
0 = Bus line is sampled once at the sample point

bit 5-3 **SEG1PH2:SEG1PH0:** Phase Segment 1 bits
111 = Phase Segment 1 Time = 8 x TQ
110 = Phase Segment 1 Time = 7 x TQ
101 = Phase Segment 1 Time = 6 x TQ
100 = Phase Segment 1 Time = 5 x TQ
011 = Phase Segment 1 Time = 4 x TQ
010 = Phase Segment 1 Time = 3 x TQ
001 = Phase Segment 1 Time = 2 x TQ
000 = Phase Segment 1 Time = 1 x TQ

bit 2-0 **PRSEG2:PRSEG0:** Propagation Time Select bits
111 = Propagation Time = 8 x TQ
110 = Propagation Time = 7 x TQ
101 = Propagation Time = 6 x TQ
100 = Propagation Time = 5 x TQ
011 = Propagation Time = 4 x TQ
010 = Propagation Time = 3 x TQ
001 = Propagation Time = 2 x TQ
000 = Propagation Time = 1 x TQ

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

Note: This register is accessible in Configuration mode only.

TABLE 19-3: VALUES FOR ICODE<2:0>

ICOD <2:0>	Interrupt	Boolean Expression
000	None	$\overline{ERR}$•WAK•$\overline{TX0}$•$\overline{TX1}$•$\overline{TX2}$•$\overline{RX0}$•$\overline{RX1}$
001	Error	ERR
010	TXB2	$\overline{ERR}$•$\overline{TX0}$•$\overline{TX1}$•TX2
011	TXB1	$\overline{ERR}$•$\overline{TX0}$•TX1
100	TXB0	$\overline{ERR}$•TX0
101	RXB1	$\overline{ERR}$•$\overline{TX0}$•$\overline{TX1}$•$\overline{TX2}$•$\overline{RX0}$•RX1
110	RXB0	$\overline{ERR}$•$\overline{TX0}$•$\overline{TX1}$•$\overline{TX2}$•RX0
111	Wake on Interrupt	$\overline{ERR}$•$\overline{TX0}$•$\overline{TX1}$•$\overline{TX2}$•$\overline{RX0}$•$\overline{RX1}$•WAK

Key:
ERR = ERRIF * ERRIE RX0 = RXB0IF * RXB0IE
TX0 = TXB0IF * TXB0IE RX1 = RXB1IF * RXB1IE
TX1 = TXB1IF * TXB1IE WAK = WAKIF * WAKIE
TX2 = TXB2IF * TXB2IE

19.13.4 MESSAGE ERROR INTERRUPT

When an error occurs during transmission or reception of a message, the message error flag IRXIF will be set and if the IRXIE bit is set, an interrupt will be generated. This is intended to be used to facilitate baud rate determination when used in conjunction with Listen Only mode.

19.13.5 BUS ACTIVITY WAKE-UP INTERRUPT

When the PIC18FXX8 is in Sleep mode and the bus activity wake-up interrupt is enabled, an interrupt will be generated and the WAKIF bit will be set when activity is detected on the CAN bus. This interrupt causes the PIC18FXX8 to exit Sleep mode. The interrupt is reset by the MCU, clearing the WAKIF bit.

19.13.6 ERROR INTERRUPT

When the error interrupt is enabled, an interrupt is generated if an overflow condition occurs or if the error state of transmitter or receiver has changed. The error flags in COMSTAT will indicate one of the following conditions.

19.13.6.1 Receiver Overflow

An overflow condition occurs when the MAB has assembled a valid received message (the message meets the criteria of the acceptance filters) and the receive buffer associated with the filter is not available for loading of a new message. The associated COMSTAT.RXnOVFL bit will be set to indicate the overflow condition. This bit must be cleared by the MCU.

19.13.6.2 Receiver Warning

The receive error counter has reached the MCU warning limit of 96.

19.13.6.3 Transmitter Warning

The transmit error counter has reached the MCU warning limit of 96.

19.13.6.4 Receiver Bus Passive

The receive error counter has exceeded the error-passive limit of 127 and the device has gone to error-passive state.

19.13.6.5 Transmitter Bus Passive

The transmit error counter has exceeded the error-passive limit of 127 and the device has gone to error-passive state.

19.13.6.6 Bus-Off

The transmit error counter has exceeded 255 and the device has gone to bus-off state.

19.13.7 INTERRUPT ACKNOWLEDGE

Interrupts are directly associated with one or more status flags in the PIR register. Interrupts are pending as long as one of the flags is set. Once an interrupt flag is set by the device, the flag cannot be reset by the microcontroller until the interrupt condition is removed.

REGISTER 24-7: CONFIG6L: CONFIGURATION REGISTER 6 LOW (BYTE ADDRESS 30000Ah)

U-0	U-0	U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	—	WRT3(1)	WRT2(1)	WRT1	WRT0
bit 7							bit 0

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

bit 3 **WRT3:** Write Protection bit(1)
1 = Block 3 (006000-007FFFh) not write-protected
0 = Block 3 (006000-007FFFh) write-protected

bit 2 **WRT2:** Write Protection bit(1)
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 (000200-001FFFh) not write-protected
0 = Block 0 (000200-001FFFh) write-protected

Note 1: Unimplemented in PIC18FX48 devices; maintain this bit set.

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as '0'
-n = Value when device is unprogrammed		u = Unchanged from programmed state

REGISTER 24-8: CONFIG6H: CONFIGURATION REGISTER 6 HIGH (BYTE ADDRESS 30000Bh)

R/P-1	R/P-1	R-1	U-0	U-0	U-0	U-0	U-0
WRTD	WRTB	WRTC	—	—	—	—	—
bit 7							bit 0

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 (000000-0001FFh) not write-protected
0 = Boot Block (000000-0001FFh) write-protected

bit 5 **WRTC:** Configuration Register Write Protection bit
1 = Configuration registers (300000-3000FFh) not write-protected
0 = Configuration registers (300000-3000FFh) write-protected

Note: This bit is read-only and cannot be changed in user mode.

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

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as '0'
-n = Value when device is unprogrammed		u = Unchanged from programmed state

RLNCF **Rotate Left f (no carry)**

Syntax: [*label*] RLNCF f [,d [,a]]

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

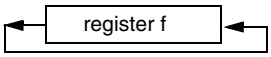
Operation: (f<n>) → dest<n + 1>,
(f<7>) → dest<0>

Status Affected: N, Z

Encoding:

0100	01da	ffff	ffff

Description: The contents of register 'f' are rotated one bit to the left. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is stored back in register 'f' (default). If 'a' is '0', the Access Bank will be selected, overriding the BSR value. If 'a' is '1', then the bank will be selected as per the BSR value (default).



Words: 1

Cycles: 1

Q Cycle Activity:

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

Example: RLNCF REG

Before Instruction
REG = 1010 1011
After Instruction
REG = 0101 0111

RRCF **Rotate Right f through Carry**

Syntax: [*label*] RRCF f [,d [,a]]

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

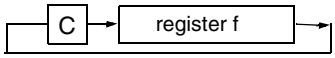
Operation: (f<n>) → dest<n – 1>,
(f<0>) → C,
(C) → dest<7>

Status Affected: C, N, Z

Encoding:

0011	00da	ffff	ffff

Description: The contents of register 'f' are rotated one bit to the right through the Carry flag. 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 will be selected, overriding the BSR value. If 'a' is '1', then the bank will be selected as per the BSR value (default).



Words: 1

Cycles: 1

Q Cycle Activity:

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

Example: RRCF REG, W

Before Instruction
REG = 1110 0110
C = 0
After Instruction
REG = 1110 0110
W = 0111 0011
C = 0

TBLRD **Table Read**

Syntax: [*label*] TBLRD (*; *+; *-; +*)

Operands: None

Operation: if TBLRD *,
(Prog Mem (TBLPTR)) → TABLAT;
TBLPTR – No Change;
if TBLRD *+,
(Prog Mem (TBLPTR)) → TABLAT;
(TBLPTR) + 1 → TBLPTR;
if TBLRD *-,
(Prog Mem (TBLPTR)) → TABLAT;
(TBLPTR) – 1 → TBLPTR;
if TBLRD +*,
(TBLPTR) + 1 → TBLPTR;
(Prog Mem (TBLPTR)) → TABLAT

Status Affected: None

Encoding:

0000	0000	0000	10nn nn=0 * =1 *+ =2 *- =3 +*

Description: This instruction is used to read the contents of Program Memory (P.M.). To address the program memory, a pointer called Table Pointer (TBLPTR) is used.
The TBLPTR (a 21-bit pointer) points to each byte in the program memory. TBLPTR has a 2-Mbyte address range.

TBLPTR[0] = 0: Least Significant Byte of Program Memory Word

TBLPTR[0] = 1: Most Significant Byte of Program Memory Word

The TBLRD instruction can modify the value of TBLPTR as follows:
- no change
- post-increment
- post-decrement
- pre-increment

Words: 1

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	No operation	No operation
No operation	No operation (Read Program Memory)	No operation	No operation (Write TABLAT)

Example 1:
```
TBLRD  *+ ;
```
Before Instruction
TABLAT = 0x55
TBLPTR = 0x00A356
MEMORY(0x00A356) = 0x34
After Instruction
TABLAT = 0x34
TBLPTR = 0x00A357

Example 2:
```
TBLRD  +* ;
```
Before Instruction
TABLAT = 0xAA
TBLPTR = 0x01A357
MEMORY(0x01A357) = 0x12
MEMORY(0x01A358) = 0x34
After Instruction
TABLAT = 0x34
TBLPTR = 0x01A358

27.1 DC Characteristics (Continued)

PIC18LFXX8 (Industrial)			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial				
PIC18FXX8 (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.	**Symbol**	**Characteristic/ Device**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
	IDD	**Supply Current(2,3,4)**					
D010		PIC18LFXX8					XT oscillator configuration
			—	.7	2	mA	VDD = 2.0V, +25°C, FOSC = 4 MHz
			—	.7	2	mA	VDD = 2.0V, -40°C to +85°C, FOSC = 4 MHz
			—	1.7	4	mA	VDD = 4.2V, -40°C to +85°C, FOSC = 4 MHz
							RC oscillator configuration
			—	1	2.5	mA	VDD = 2.0V, +25°C, FOSC = 4 MHz
			—	1	2.5	mA	VDD = 2.0V, -40°C to +85°C, FOSC = 4 MHz
			—	2.5	5	mA	VDD = 4.2V, -40°C to +85°C, FOSC = 4 MHz
							RCIO oscillator configuration
			—	.7	2.5	mA	VDD = 2.0V, +25°C, FOSC = 4 MHz
			—	.7	2.5	mA	VDD = 2.0V, -40°C to +85°C, FOSC = 4 MHz
			—	1.8	4	mA	VDD = 4.2V, -40°C to +85°C, FOSC = 4 MHz
D010		PIC18FXX8					XT oscillator configuration
			—	1.7	4	mA	VDD = 4.2V, +25°C, FOSC = 4 MHz
			—	1.7	4	mA	VDD = 4.2V, -40°C to +85°C, FOSC = 4 MHz
			—	1.7	4	mA	VDD = 4.2V, -40°C to +125°C, FOSC = 4 MHz
							RC oscillator configuration
			—	2.5	5	mA	VDD = 4.2V, +25°C, FOSC = 4 MHz
			—	2.5	5	mA	VDD = 4.2V, -40°C to +85°C, FOSC = 4 MHz
			—	2.5	6	mA	VDD = 4.2V, -40°C to +125°C, FOSC = 4 MHz
							RCIO oscillator configuration
			—	1.8	4	mA	VDD = 4.2V, +25°C, FOSC = 4 MHz
			—	1.8	5	mA	VDD = 4.2V, -40°C to +85°C, FOSC = 4 MHz
			—	1.8	5	mA	VDD = 4.2V, -40°C to +125°C, FOSC = 4 MHz
D010A		PIC18LFXX8					LP oscillator, FOSC = 32 kHz, WDT disabled
			—	18	40	µA	VDD = 2.0V, -40°C to +85°C
D010A		PIC18FXX8					LP oscillator, FOSC = 32 kHz, WDT disabled
			—	60	150	µA	VDD = 4.2V, -40°C to +85°C
			—	60	180	µA	VDD = 4.2V, -40°C to +125°C

Legend: Rows are shaded for improved readability.

Note 1: This is the limit to which VDD can be lowered in Sleep mode, or during a device Reset, without losing RAM data.

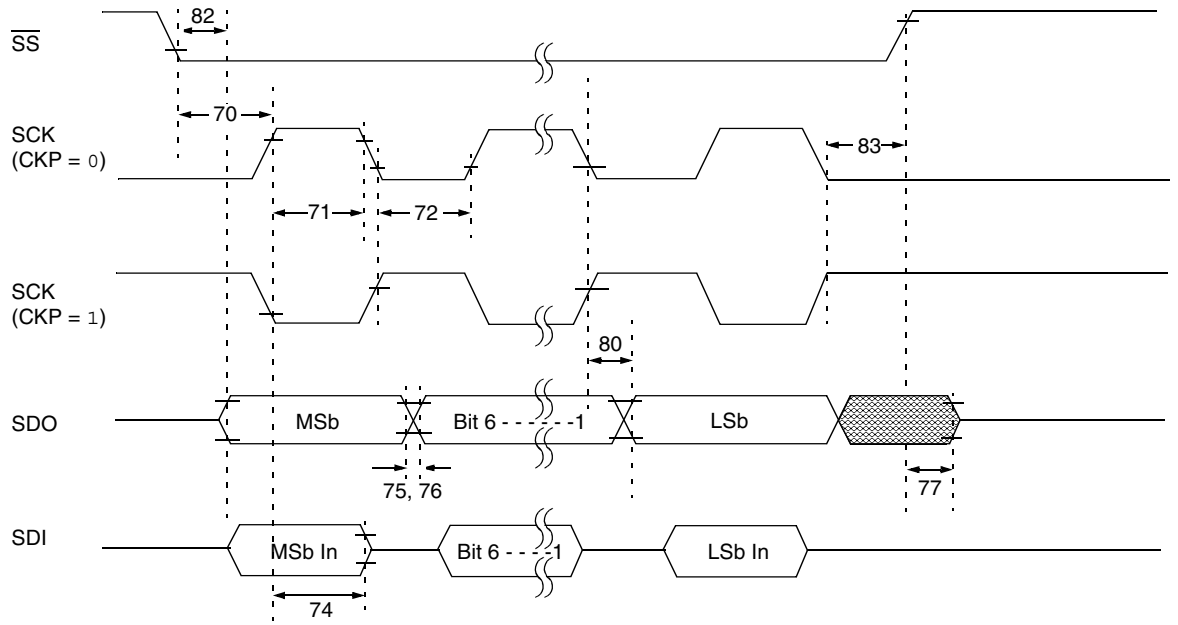
2: The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption.
The test conditions for all IDD measurements in active operation mode are:
OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD
$\overline{MCLR}$ = VDD; WDT enabled/disabled as specified.

3: 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 and VSS and all features that add delta current disabled (such as WDT, Timer1 Oscillator, BOR, ...).

4: For RC oscillator configuration, current through REXT is not included. The current through the resistor can be estimated by the formula Ir = VDD/2 REXT (mA) with REXT in kOhm.

5: The LVD and BOR modules share a large portion of circuitry. The ΔIBOR and ΔILVD currents are not additive. Once one of these modules is enabled, the other may also be enabled without further penalty.

FIGURE 27-16: EXAMPLE SPI™ SLAVE MODE TIMING (CKE = 1)



Note: Refer to Figure 27-5 for load conditions.

TABLE 27-16: EXAMPLE SPI™ SLAVE MODE REQUIREMENTS (CKE = 1)

Param No.	Symbol	Characteristic		Min	Max	Units	Conditions
70	TssL2scH, TssL2scL	$\overline{SS}$ ↓ to SCK ↓ or SCK ↑ Input		TCY	—	ns	
71	TscH	SCK Input High Time (Slave mode)	Continuous	1.25 TCY + 30	—	ns	
71A			Single Byte	40	—	ns	**(Note 1)**
72	TscL	SCK Input Low Time (Slave mode)	Continuous	1.25 TCY + 30	—	ns	
72A			Single Byte	40	—	ns	**(Note 1)**
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		100	—	ns	
75	TdoR	SDO Data Output Rise Time	PIC18**F**XX8	—	25	ns	
			PIC18**LF**XX8	—	45	ns	
76	TdoF	SDO Data Output Fall Time		—	25	ns	
77	TssH2doZ	$\overline{SS}$ ↑ to SDO Output High-Impedance		10	50	ns	
78	TscR	SCK Output Rise Time (Master mode)	PIC18**F**XX8	—	25	ns	
			PIC18**LF**XX8	—	45	ns	
79	TscF	SCK Output Fall Time (Master mode)		—	25	ns	
80	TscH2doV, TscL2doV	SDO Data Output Valid after SCK Edge	PIC18**F**XX8	—	50	ns	
			PIC18**LF**XX8	—	100	ns	
82	TssL2doV	SDO Data Output Valid after $\overline{SS}$ ↓ Edge	PIC18**F**XX8	—	50	ns	
			PIC18**LF**XX8	—	100	ns	
83	TscH2ssH, TscL2ssH	$\overline{SS}$ ↑ after SCK Edge		1.5 TCY + 40	—	ns	

Note 1: Requires the use of parameter #73A.

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

FIGURE 28-9: TYPICAL IDD vs. FOSC OVER VDD (EC MODE)

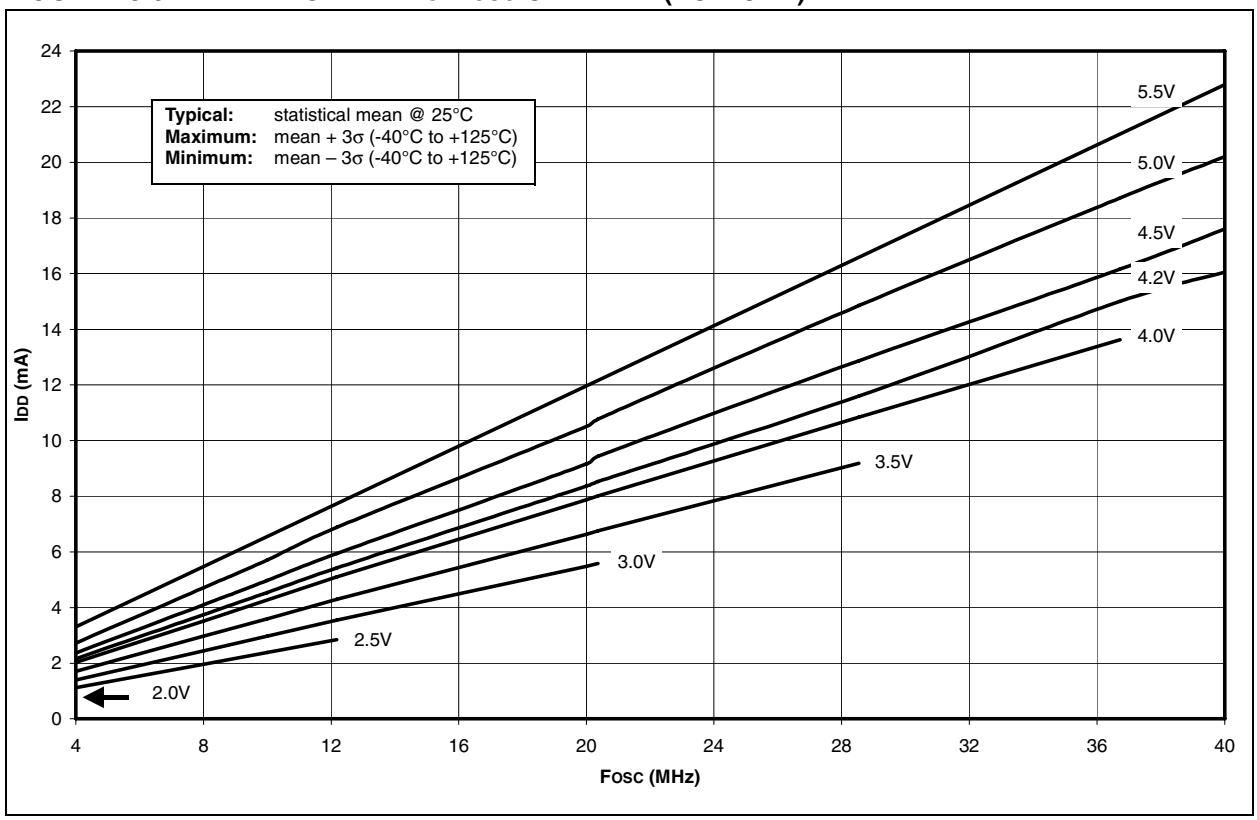
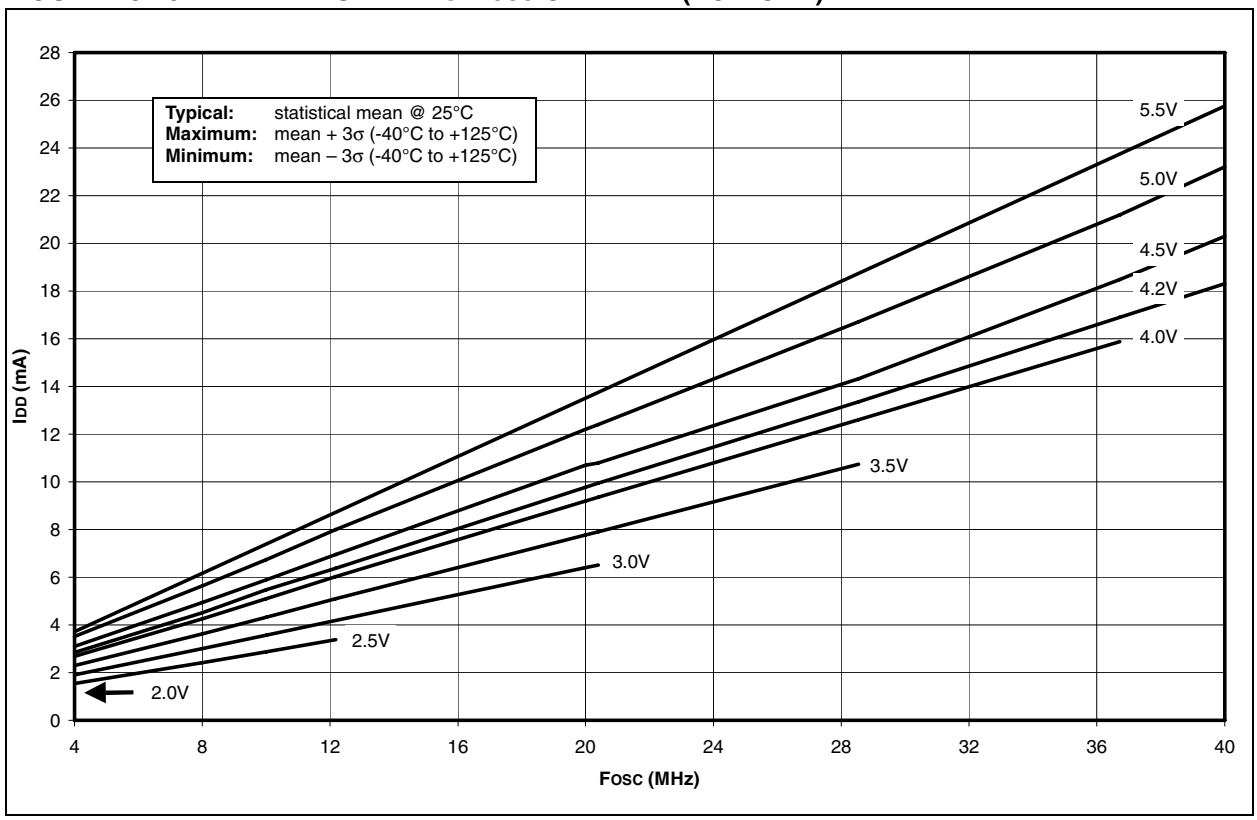
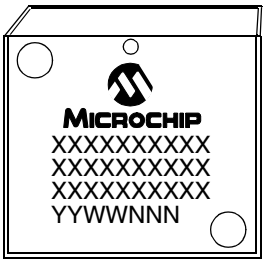


FIGURE 28-10: MAXIMUM IDD vs. FOSC OVER VDD (EC MODE)

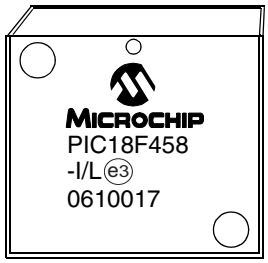


29.1 Package Marking Information (Continued)

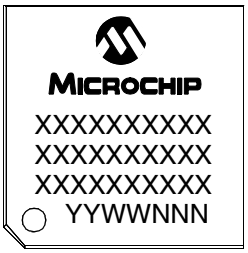
44-Lead PLCC



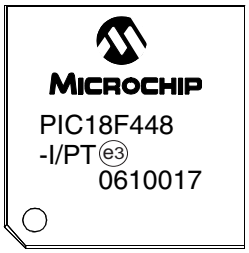
Example



44-Lead TQFP



Example



C