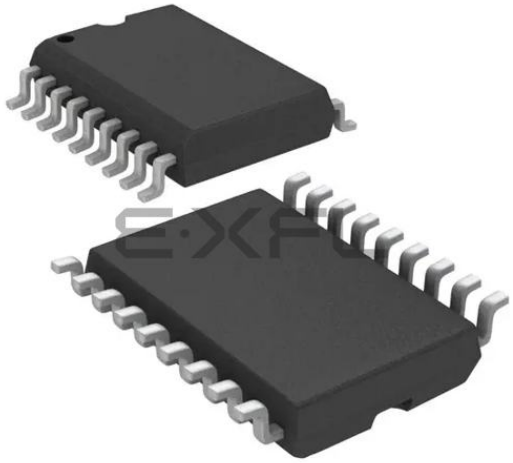


Welcome to E-XFL.COM

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	10MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	16
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	368 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 7x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	18-SOIC (0.295", 7.50mm Width)
Supplier Device Package	18-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf88t-i-so

2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and peripheral modules for controlling the desired operation of the device. These registers are implemented as static RAM. A list of these registers is given in Table 2-1.

The Special Function Registers can be classified into two sets: core (CPU) and peripheral. Those registers associated with the core functions are described in detail in this section. Those related to the operation of the peripheral features are described in detail in the peripheral feature section.

TABLE 2-1: SPECIAL FUNCTION REGISTER SUMMARY

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Details on page
Bank 0											
00h(2)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	26, 135
01h	TMR0	Timer0 Module Register								xxxx xxxx	69
02h(2)	PCL	Program Counter (PC) Least Significant Byte								0000 0000	
03h(2)	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	17
04h(2)	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	135
05h	PORTA	PORTA Data Latch when written; PORTA pins when read (PIC16F87) PORTA Data Latch when written; PORTA pins when read (PIC16F88)								xxxx 0000 xxx0 0000	52
06h	PORTB	PORTB Data Latch when written; PORTB pins when read (PIC16F87) PORTB Data Latch when written; PORTB pins when read (PIC16F88)								xxxx xxxx 00xx xxxx	58
07h	—	Unimplemented								—	—
08h	—	Unimplemented								—	—
09h	—	Unimplemented								—	—
0Ah(1,2)	PCLATH	—	—	—	Write Buffer for the Upper 5 bits of the Program Counter					---0 0000	135
0Bh(2)	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000x	19, 69, 77
0Ch	PIR1	—	ADIF(4)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	21, 77
0Dh	PIR2	OSFIF	CMIF	—	EEIF	—	—	—	—	00-0 ----	23, 34
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	77, 83
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	77, 83
10h	T1CON	—	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	TMR1CS	TMR1ON	-000 0000	72, 83
11h	TMR2	Timer2 Module Register								0000 0000	80, 85
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	80, 85
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	90, 95
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	89, 95
15h	CCPR1L	Capture/Compare/PWM Register 1 (LSB)								xxxx xxxx	83, 85
16h	CCPR1H	Capture/Compare/PWM Register 1 (MSB)								xxxx xxxx	83, 85
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	81, 83
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	98, 99
19h	TXREG	AUSART Transmit Data Register								0000 0000	103
1Ah	RCREG	AUSART Receive Data Register								0000 0000	105
1Bh	—	Unimplemented								—	—
1Ch	—	Unimplemented								—	—
1Dh	—	Unimplemented								—	—
1Eh	ADRESH(4)	A/D Result Register High Byte								xxxx xxxx	120
1Fh	ADCON0(4)	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	—	ADON	0000 00-0	114, 120

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for PC<12:8>, whose contents are transferred to the upper byte of the program counter.

2: These registers can be addressed from any bank.

3: RA5 is an input only; the state of the TRISA5 bit has no effect and will always read '1'.

4: PIC16F88 device only.

2.2.2.4 PIE1 Register

This register contains the individual enable bits for the peripheral interrupts.

Note:	Bit PEIE (INTCON<6>) must be set to enable any peripheral interrupt.

REGISTER 2-4: PIE1: PERIPHERAL INTERRUPT ENABLE REGISTER 1 (ADDRESS 8Ch)

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

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

bit 6 **ADIE:** A/D Converter Interrupt Enable bit(1)
1 = Enabled
0 = Disabled

Note 1: This bit is only implemented on the PIC16F88. The bit will read '0' on the PIC16F87.

bit 5 **RCIE:** AUSART Receive Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 4 **TXIE:** AUSART Transmit Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 3 **SSPIE:** Synchronous Serial Port (SSP) Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 2 **CCP1IE:** CCP1 Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 1 **TMR2IE:** TMR2 to PR2 Match Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 0 **TMR1IE:** TMR1 Overflow Interrupt Enable bit
1 = Enabled
0 = Disabled

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

2.2.2.7 PIR2 Register

The PIR2 register contains the flag bit for the EEPROM write operation interrupt.

Note:	Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit, or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-7: PIR2: PERIPHERAL INTERRUPT REQUEST (FLAG) REGISTER 2 (ADDRESS 0Dh)

R/W-0	R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0
OSFIF	CMIF	—	EEIF	—	—	—	—
bit 7							bit 0

bit 7 **OSFIF:** Oscillator Fail Interrupt Flag bit
1 = System oscillator failed, clock input has changed to INTRC (must be cleared in software)
0 = System clock operating

bit 6 **CMIF:** Comparator Interrupt Flag bit
1 = Comparator input has changed (must be cleared in software)
0 = Comparator input has not changed

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

bit 4 **EEIF:** EEPROM Write Operation Interrupt Flag bit
1 = The write operation completed (must be cleared in software)
0 = The write operation is not complete or has not been started

bit 3-0 **Unimplemented:** Read 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

2.2.2.8 PCON Register

Note:	Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit, or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

The Power Control (PCON) register contains a flag bit to allow differentiation between a Power-on Reset (POR), a Brown-out Reset, an external $\overline{MCLR}$ Reset and WDT Reset.

Note:	$\overline{BOR}$ is unknown on Power-on Reset. It must then be set by the user and checked on subsequent Resets to see if $\overline{BOR}$ is clear, indicating a brown-out has occurred. The $\overline{BOR}$ status bit is a 'don't care' and is not necessarily predictable if the brown-out circuit is disabled (by clearing the BOREN bit in the Configuration Word register).

REGISTER 2-8: PCON: POWER CONTROL REGISTER (ADDRESS 8Eh)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-x
—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$
bit 7							bit 0

bit 7-2 **Unimplemented:** Read as '0'

bit 1 **$\overline{POR}$:** Power-on Reset Status bit
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0 **$\overline{BOR}$:** Brown-out Reset Status bit
1 = No Brown-out Reset occurred
0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

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

3.0 DATA EEPROM AND FLASH PROGRAM MEMORY

The data EEPROM and Flash program memory are readable and writable during normal operation (over the full VDD range). This memory is not directly mapped in the register file space. Instead, it is indirectly addressed through the Special Function Registers. There are six SFRs used to read and write this memory:

- EECON1
- EECON2
- EEDATA
- EEDATH
- EEADR
- EEADRH

This section focuses on reading and writing data EEPROM and Flash program memory during normal operation. Refer to the appropriate device programming specification document for serial programming information.

When interfacing the data memory block, EEDATA holds the 8-bit data for read/write and EEADR holds the address of the EEPROM location being accessed. The PIC16F87/88 devices have 256 bytes of data EEPROM with an address range from 00h to 0FFh. When writing to unimplemented locations, the charge pump will be turned off.

When interfacing the program memory block, the EEDATA and EEDATH registers form a two-byte word that holds the 14-bit data for read/write and the EEADR and EEADRH registers form a two-byte word that holds the 13-bit address of the EEPROM location being accessed. The PIC16F87/88 devices have 4K words of program Flash with an address range from 0000h to 0FFFh. Addresses above the range of the respective device will wraparound to the beginning of program memory.

The EEPROM data memory allows single byte read and write. The Flash program memory allows single-word reads and four-word block writes. Program memory writes must first start with a 32-word block erase, then write in 4-word blocks. A byte write in data EEPROM memory automatically erases the location and writes the new data (erase before write).

The write time is controlled by an on-chip timer. The write/erase voltages are generated by an on-chip charge pump, rated to operate over the voltage range of the device for byte or word operations.

When the device is code-protected, the CPU may continue to read and write the data EEPROM memory. Depending on the settings of the write-protect bits, the device may or may not be able to write certain blocks of the program memory; however, reads of the program memory are allowed. When code-protected, the device programmer can no longer access data or program memory; this does NOT inhibit internal reads or writes.

3.1 EEADR and EEADRH

The EEADRH:EEADR register pair can address up to a maximum of 256 bytes of data EEPROM, or up to a maximum of 8K words of program EEPROM. When selecting a data address value, only the LSB of the address is written to the EEADR register. When selecting a program address value, the MSB of the address is written to the EEADRH register and the LSB is written to the EEADR register.

If the device contains less memory than the full address reach of the address register pair, the Most Significant bits of the registers are not implemented. For example, if the device has 128 bytes of data EEPROM, the Most Significant bit of EEADR is not implemented on access to data EEPROM.

3.2 EECON1 and EECON2 Registers

EECON1 is the control register for memory accesses.

Control bit EEPGD determines if the access will be a program or data memory access. When clear, as it is when reset, any subsequent operations will operate on the data memory. When set, any subsequent operations will operate on the program memory.

Control bits, RD and WR, initiate read and write, respectively. These bits cannot be cleared, only set in software. They are cleared in hardware at completion of the read or write operation. The inability to clear the WR bit in software prevents the accidental, premature termination of a write operation.

The WREN bit, when set, will allow a write or erase operation. On power-up, the WREN bit is clear. The WRERR bit is set when a write (or erase) operation is interrupted by a $\overline{\text{MCLR}}$, or a WDT Time-out Reset during normal operation. In these situations, following Reset, the user can check the WRERR bit and rewrite the location. The data and address will be unchanged in the EEDATA and EEADR registers.

Interrupt flag bit, EEIF in the PIR2 register, is set when the write is complete. It must be cleared in software.

EECON2 is not a physical register. Reading EECON2 will read all '0's. The EECON2 register is used exclusively in the EEPROM write sequence.

3.3 Reading Data EEPROM Memory

To read a data memory location, the user must write the address to the EEADR register, clear the EEPGD control bit (EECON1<7>) and then set control bit RD (EECON1<0>). The data is available in the very next cycle in the EEDATA register; therefore, it can be read in the next instruction (see Example 3-1). EEDATA will hold this value until another read or until it is written to by the user (during a write operation).

The steps to reading the EEPROM data memory are:

1. Write the address to EEADR. Make sure that the address is not larger than the memory size of the device.
2. Clear the EEPGD bit to point to EEPROM data memory.
3. Set the RD bit to start the read operation.
4. Read the data from the EEDATA register.

EXAMPLE 3-1: DATA EEPROM READ

```
BANKSEL EEADR          ; Select Bank of EEADR
MOVF    ADDR, W        ;
MOVWF   EEADR          ; Data Memory Address
                       ; to read
BANKSEL EECON1         ; Select Bank of EECON1
BCF     EECON1, EEPGD  ; Point to Data memory
BSF     EECON1, RD     ; EE Read
BANKSEL EEDATA         ; Select Bank of EEDATA
MOVF    EEDATA, W      ; W = EEDATA
```

3.4 Writing to Data EEPROM Memory

To write an EEPROM data location, the user must first write the address to the EEADR register and the data to the EEDATA register. Then, the user must follow a specific write sequence to initiate the write for each byte.

The write will not initiate if the write sequence is not exactly followed (write 55h to EECON2, write AAh to EECON2, then set WR bit) for each byte. We strongly recommend that interrupts be disabled during this code segment (see Example 3-2).

Additionally, the WREN bit in EECON1 must be set to enable write. This mechanism prevents accidental writes to data EEPROM due to errant (unexpected) code execution (i.e., lost programs). The user should keep the WREN bit clear at all times except when updating EEPROM. The WREN bit is not cleared by hardware

After a write sequence has been initiated, clearing the WREN bit will not affect this write cycle. The WR bit will be inhibited from being set unless the WREN bit is set. At the completion of the write cycle, the WR bit is cleared in hardware and the EE Write Complete Interrupt Flag bit (EEIF) is set. The user can either enable this interrupt or poll this bit. EEIF must be cleared by software.

The steps to write to EEPROM data memory are:

1. If step 10 is not implemented, check the WR bit to see if a write is in progress.
2. Write the address to EEADR. Make sure that the address is not larger than the memory size of the device.
3. Write the 8-bit data value to be programmed in the EEDATA register.
4. Clear the EEPGD bit to point to EEPROM data memory.
5. Set the WREN bit to enable program operations.
6. Disable interrupts (if enabled).
7. Execute the special five instruction sequence:

   Write 55h to EECON2 in two steps (first to W, then to EECON2).

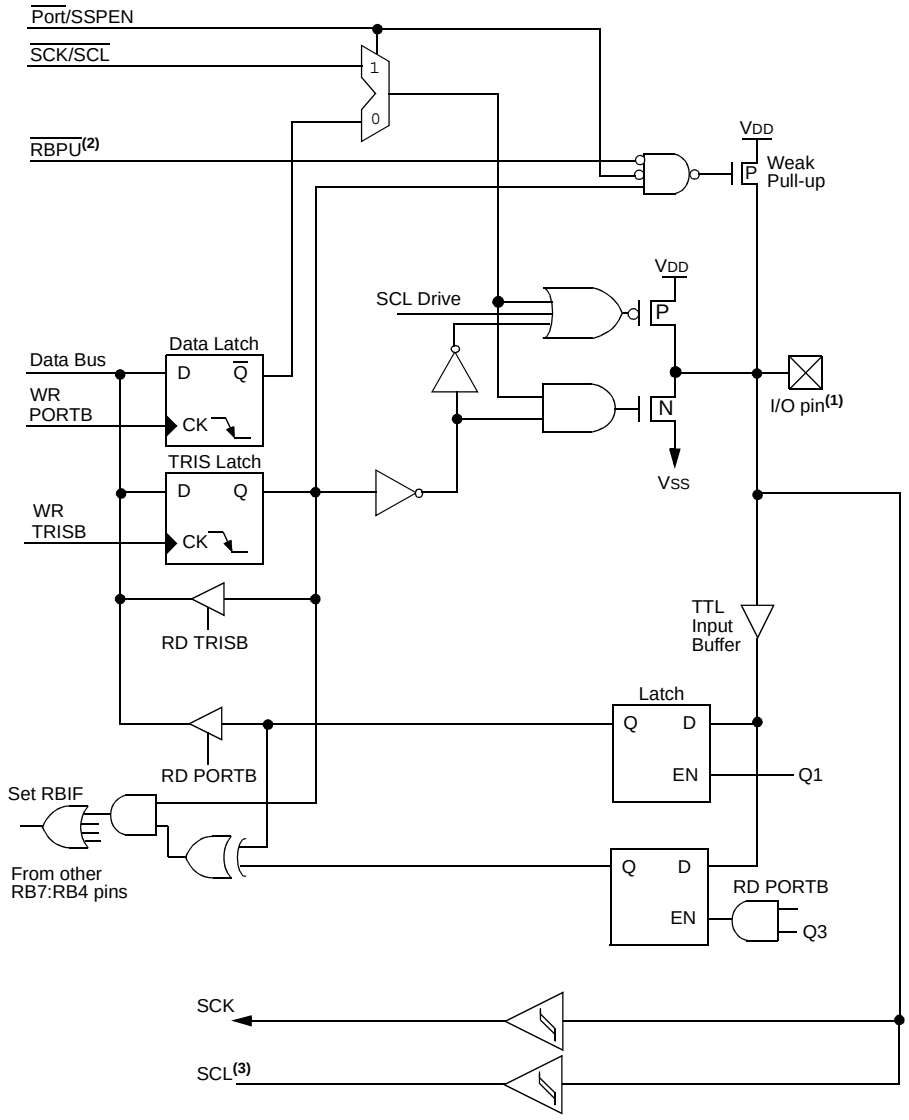
   Write AAh to EECON2 in two steps (first to W, then to EECON2).

   Set the WR bit.
8. Enable interrupts (if using interrupts).
9. Clear the WREN bit to disable program operations.
10. At the completion of the write cycle, the WR bit is cleared and the EEIF interrupt flag bit is set (EEIF must be cleared by firmware). If step 1 is not implemented, then firmware should check for EEIF to be set, or WR to clear, to indicate the end of the program cycle.

EXAMPLE 3-2: DATA EEPROM WRITE

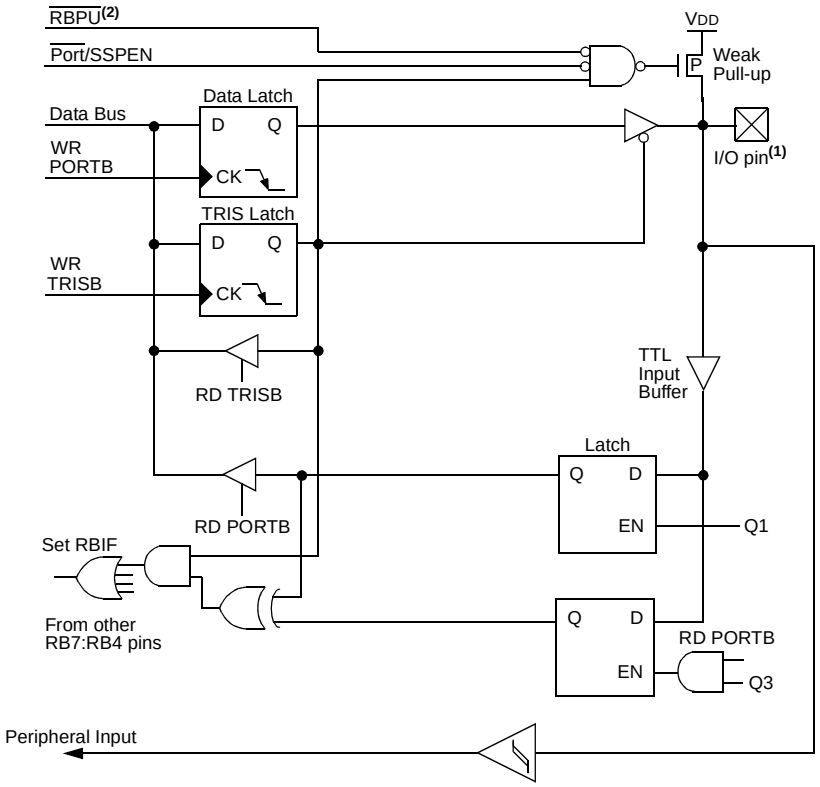
```
        BANKSEL EECON1         ; Select Bank of
                               ; EECON1
        BTFSC   EECON1, WR     ; Wait for write
        GOTO    $-1            ; to complete
        BANKSEL EEADR          ; Select Bank of
                               ; EEADR
        MOVF    ADDR, W        ;
        MOVWF   EEADR          ; Data Memory
                               ; Address to write
        MOVF    VALUE, W       ;
        MOVWF   EEDATA         ; Data Memory Value
                               ; to write
        BANKSEL EECON1         ; Select Bank of
                               ; EECON1
        BCF     EECON1, EEPGD  ; Point to DATA
                               ; memory
        BSF     EECON1, WREN   ; Enable writes

        BCF     INTCON, GIE    ; Disable INTs.
Required
Sequence
        MOVLW   55h            ;
        MOVWF   EECON2         ; Write 55h
        MOVLW   AAh            ;
        MOVWF   EECON2         ; Write AAh
        BSF     EECON1, WR     ; Set WR bit to
                               ; begin write
        BSF     INTCON, GIE    ; Enable INTs.
        BCF     EECON1, WREN   ; Disable writes
```

FIGURE 5-12: BLOCK DIAGRAM OF RB4/SCK/SCL PIN



Note 1: I/O pins have diode protection to VDD and VSS.
2: To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the $\overline{\text{RBPU}}$ bit.
3: The SCL Schmitt conforms to the I^2C™ specification.

FIGURE 5-13: BLOCK DIAGRAM OF RB5/$\overline{SS}$/TX/CK PIN



Note 1: I/O pins have diode protection to VDD and VSS.
2: To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the $\overline{RBPU}$ bit.

REGISTER 10-2: SSPCON: SYNCHRONOUS SERIAL PORT CONTROL REGISTER (ADDRESS 14h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV	SSPEN(1)	CKP	SSPM3	SSPM2	SSPM1	SSPM0
bit 7							bit 0

bit 7 **WCOL:** Write Collision Detect bit

1 = An attempt to write the SSPBUF register failed because the SSP module is busy (must be cleared in software)

0 = No collision

bit 6 **SSPOV:** Receive Overflow Indicator bit

In SPI mode:

1 = A new byte is received while the SSPBUF register is still holding the previous data. In case of overflow, the data in SSPSR is lost. Overflow can only occur in Slave mode. The user must read the SSPBUF, even if only transmitting data, to avoid setting overflow. In Master mode, the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPBUF register.

0 = No overflow

In I^2C mode:

1 = A byte is received while the SSPBUF register is still holding the previous byte. SSPOV is a "don't care" in Transmit mode. SSPOV must be cleared in software in either mode.

0 = No overflow

bit 5 **SSPEN:** Synchronous Serial Port Enable bit(1)

In SPI mode:

1 = Enables serial port and configures SCK, SDO and SDI as serial port pins

0 = Disables serial port and configures these pins as I/O port pins

In I^2C mode:

1 = Enables the serial port and configures the SDA and SCL pins as serial port pins

0 = Disables serial port and configures these pins as I/O port pins

Note 1: In both modes, when enabled, these pins must be properly configured as input or output.

bit 4 **CKP:** Clock Polarity Select bit

In SPI mode:

1 = Transmit happens on falling edge, receive on rising edge. Idle state for clock is a high level.

0 = Transmit happens on rising edge, receive on falling edge. Idle state for clock is a low level.

In I^2C Slave mode:

SCK release control

1 = Enable clock

0 = Holds clock low (clock stretch). (Used to ensure data setup time.)

bit 3-0 **SSPM<3:0>:** Synchronous Serial Port Mode Select bits

0000 = SPI Master mode, clock = OSC/4

0001 = SPI Master mode, clock = OSC/16

0010 = SPI Master mode, clock = OSC/64

0011 = SPI Master mode, clock = TMR2 output/2

0100 = SPI Slave mode, clock = SCK pin. $\overline{SS}$ pin control enabled.

0101 = SPI Slave mode, clock = SCK pin. $\overline{SS}$ pin control disabled. $\overline{SS}$ can be used as I/O pin.

0110 = I^2C Slave mode, 7-bit address

0111 = I^2C Slave mode, 10-bit address

1011 = I^2C Firmware Controlled Master mode (Slave Idle)

1110 = I^2C Slave mode, 7-bit address with Start and Stop bit interrupts enabled

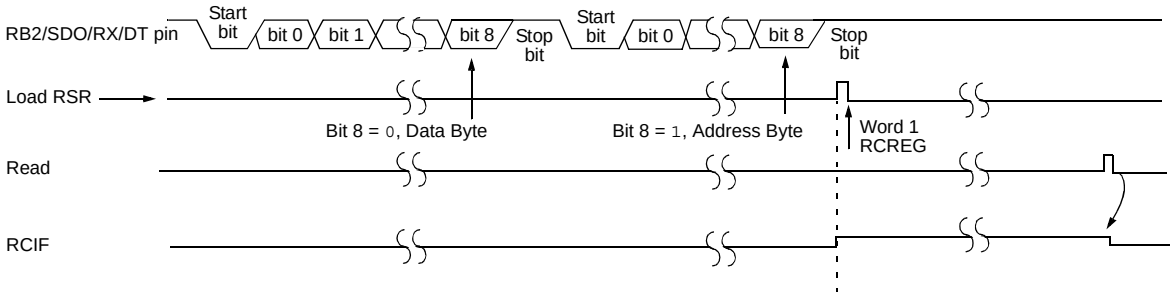
1111 = I^2C Slave mode, 10-bit address with Start and Stop bit interrupts enabled

1000, 1001, 1010, 1100, 1101 = Reserved

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

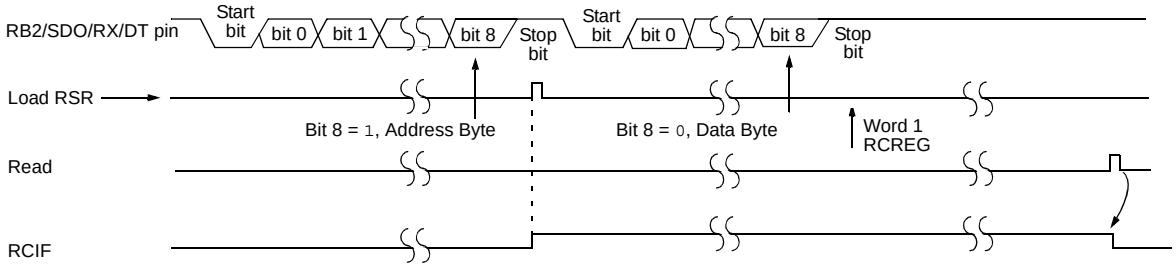
NOTES:

FIGURE 11-7: ASYNCHRONOUS RECEPTION WITH ADDRESS DETECT



Note: This timing diagram shows a data byte followed by an address byte. The data byte is not read into the RCREG (Receive Buffer) because ADDEN = 1.

FIGURE 11-8: ASYNCHRONOUS RECEPTION WITH ADDRESS BYTE FIRST



Note: This timing diagram shows a data byte followed by an address byte. The data byte is not read into the RCREG (Receive Buffer) because ADDEN was not updated and still = 0.

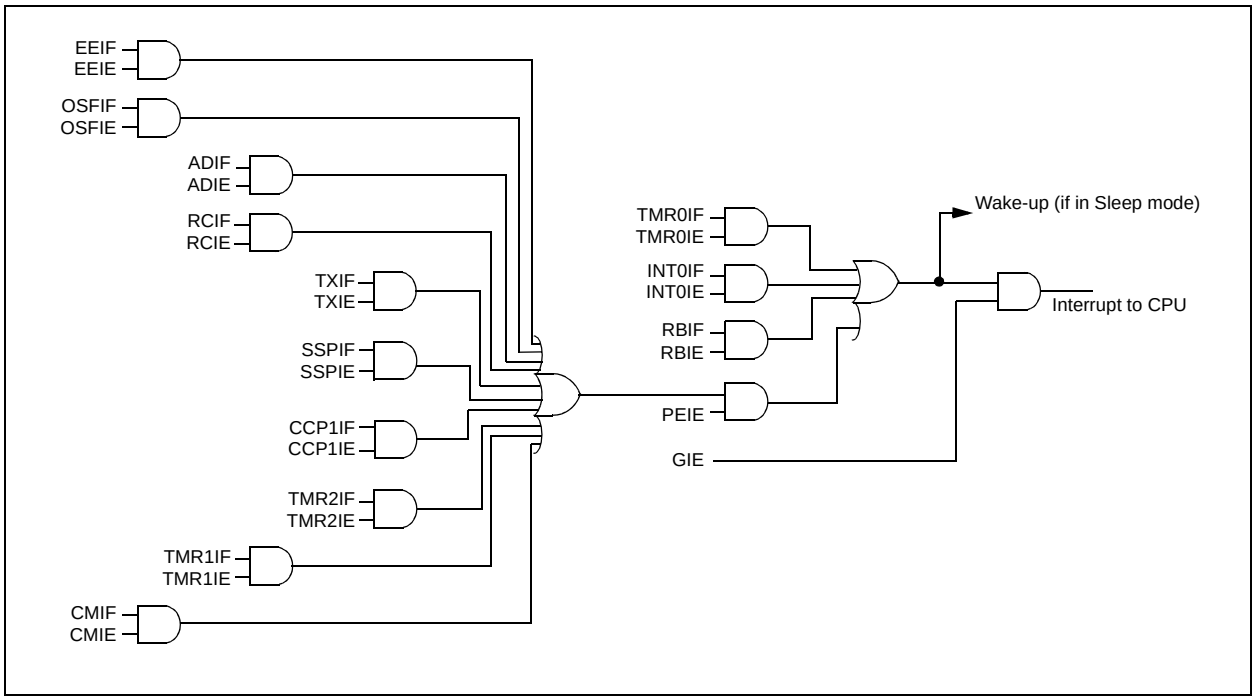
TABLE 11-9: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
0Bh, 8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF(1)	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
1Ah	RCREG	AUSART Receive Data Register								0000 0000	0000 0000
8Ch	PIE1	—	ADIE(1)	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for asynchronous reception.

Note 1: This bit is only implemented on the PIC16F88. The bit will read '0' on the PIC16F87.

FIGURE 15-7: INTERRUPT LOGIC



15.18 Low-Voltage ICSP Programming

The LVP bit of the Configuration Word enables Low-Voltage ICSP Programming. This mode allows the microcontroller to be programmed via ICSP using a VDD source in the operating voltage range. This only means that VPP does not have to be brought to VIHH, but can instead be left at the normal operating voltage. In this mode, the RB3/PGM pin is dedicated to the programming function and ceases to be a general purpose I/O pin.

If Low-Voltage Programming mode is not used, the LVP bit can be programmed to a '0' and RB3/PGM becomes a digital I/O pin. However, the LVP bit may only be programmed when Programming mode is entered with VIHH on MCLR. The LVP bit can only be changed when using high voltage on MCLR.

It should be noted that once the LVP bit is programmed to '0', only the High-Voltage Programming mode is available and only this mode can be used to program the device.

When using Low-Voltage ICSP, the part must be supplied at 4.5V to 5.5V if a bulk erase will be executed. This includes reprogramming of the code-protect bits from an ON state to an OFF state. For all other cases of Low-Voltage ICSP, the part may be programmed at the normal operating voltage. This means calibration values, unique user IDs or user code can be reprogrammed or added.

The following LVP steps assume the LVP bit is set in the Configuration register.

1. Apply VDD to the VDD pin.
2. Drive MCLR low.
3. Apply VDD to the RB3/PGM pin.
4. Apply VDD to the MCLR pin.
5. Follow with the associated programming steps.

Note 1: The High-Voltage Programming mode is always available, regardless of the state of the LVP bit, by applying VIHH to the MCLR pin.

2: While in Low-Voltage ICSP mode (LVP = 1), the RB3 pin can no longer be used as a general purpose I/O pin.

3: When using Low-Voltage ICSP Programming (LVP) and the pull-ups on PORTB are enabled, bit 3 in the TRISB register must be cleared to disable the pull-up on RB3 and ensure the proper operation of the device.

4: RB3 should not be allowed to float if LVP is enabled. An external pull-down device should be used to default the device to normal operating mode. If RB3 floats high, the PIC16F87/88 devices will enter Programming mode.

5: LVP mode is enabled by default on all devices shipped from Microchip. It can be disabled by clearing the LVP bit in the CONFIG1 register.

6: Disabling LVP will provide maximum compatibility to other PIC16CXXX devices.

18.2 DC Characteristics: Power-Down and Supply Current PIC16F87/88 (Industrial, Extended) PIC16LF87/88 (Industrial) (Continued)

PIC16LF87/88 (Industrial)	Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial
PIC16F87/88 (Industrial, Extended)	**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended

Param No.	Device	Typ	Max	Units	Conditions		
	Supply Current (IDD)(2,3)						
	PIC16LF87/88	72	95	µA	-40°C	VDD = 2.0V	FOSC = 1 MHz (RC Oscillator)(3)
		76	90	µA	+25°C		
		76	90	µA	+85°C		
	PIC16LF87/88	138	175	µA	-40°C	VDD = 3.0V	
		136	170	µA	+25°C		
		136	170	µA	+85°C		
	All devices	310	380	µA	-40°C	VDD = 5.0V	
		290	360	µA	+25°C		
		280	360	µA	+85°C		
	Extended devices	330	500	µA	125°C		
	PIC16LF87/88	270	335	µA	-40°C	VDD = 2.0V	FOSC = 4 MHz (RC Oscillator)(3)
		280	330	µA	+25°C		
		285	330	µA	+85°C		
	PIC16LF87/88	460	610	µA	-40°C	VDD = 3.0V	
		450	600	µA	+25°C		
		450	600	µA	+85°C		
	All devices	900	1060	µA	-40°C	VDD = 5.0V	
		890	1050	µA	+25°C		
		890	1050	µA	+85°C		
	Extended devices	.920	1.5	mA	+125°C		

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

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

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

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

OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD;
$\overline{MCLR}$ = VDD; WDT enabled/disabled as specified.

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

18.2 DC Characteristics: Power-Down and Supply Current PIC16F87/88 (Industrial, Extended) PIC16LF87/88 (Industrial) (Continued)

PIC16LF87/88 (Industrial)		**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial					
PIC16F87/88 (Industrial, Extended)		**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended					
Param No.	**Device**	**Typ**	**Max**	**Units**	**Conditions**		
	Supply Current (IDD)(2,3)						
	PIC16LF87/88	8	20	µA	-40°C	VDD = 2.0V	FOSC = 31.25 kHz (**RC_RUN** mode, Internal RC Oscillator)
		7	15	µA	+25°C		
		7	15	µA	+85°C		
	PIC16LF87/88	16	30	µA	-40°C	VDD = 3.0V	
		14	25	µA	+25°C		
		14	25	µA	+85°C		
	All devices	32	40	µA	-40°C	VDD = 5.0V	
		29	35	µA	+25°C		
		29	35	µA	+85°C		
	Extended devices	35	45	µA	+125°C		
	PIC16LF87/88	132	160	µA	-40°C	VDD = 2.0V	FOSC = 1 MHz (**RC_RUN** mode, Internal RC Oscillator)
		126	155	µA	+25°C		
		126	155	µA	+85°C		
	PIC16LF87/88	260	310	µA	-40°C	VDD = 3.0V	
		230	300	µA	+25°C		
		230	300	µA	+85°C		
	All devices	560	690	µA	-40°C	VDD = 5.0V	
		500	650	µA	+25°C		
		500	650	µA	+85°C		
	Extended devices	570	710	µA	+125°C		

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

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

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

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

OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD;
$\overline{\text{MCLR}}$ = VDD; WDT enabled/disabled as specified.

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

TABLE 18-10: I²C™ BUS DATA REQUIREMENTS

Param. No.	Symbol	Characteristic		Min	Max	Units	Conditions
100*	THIGH	Clock High Time	100 kHz mode	4.0	—	µs	
			400 kHz mode	0.6	—	µs	
			SSP Module	1.5 TCY	—		
101*	TLOW	Clock Low Time	100 kHz mode	4.7	—	µs	
			400 kHz mode	1.3	—	µs	
			SSP Module	1.5 TCY	—		
102*	TR	SDA and SCL Rise Time	100 kHz mode	—	1000	ns	
			400 kHz mode	20 + 0.1 CB	300	ns	CB is specified to be from 10-400 pF
103*	TF	SDA and SCL Fall Time	100 kHz mode	—	300	ns	
			400 kHz mode	20 + 0.1 CB	300	ns	CB is specified to be from 10-400 pF
90*	TSU:STA	Start Condition Setup Time	100 kHz mode	4.7	—	µs	Only relevant for Repeated Start condition
			400 kHz mode	0.6	—	µs	
91*	THD:STA	Start Condition Hold Time	100 kHz mode	4.0	—	µs	After this period, the first clock pulse is generated
			400 kHz mode	0.6	—	µs	
106*	THD:DAT	Data Input Hold Time	100 kHz mode	0	—	ns	
			400 kHz mode	0	0.9	µs	
107*	TSU:DAT	Data Input Setup Time	100 kHz mode	250	—	ns	**(Note 2)**
			400 kHz mode	100	—	ns	
92*	TSU:STO	Stop Condition Setup Time	100 kHz mode	4.7	—	µs	
			400 kHz mode	0.6	—	µs	
109*	TAA	Output Valid from Clock	100 kHz mode	—	3500	ns	**(Note 1)**
			400 kHz mode	—	—	ns	
110*	TBUF	Bus Free Time	100 kHz mode	4.7	—	µs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	µs	
	CB	Bus Capacitive Loading		—	400	pF	

* These parameters are characterized but not tested.

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

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

19.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES

Note:	The graphs and tables provided following this note are a statistical summary based on a limited number of samples and are provided for informational purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs or tables, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore, outside the warranted range.

"Typical" represents the mean of the distribution at 25°C. "Maximum" or "minimum" represents (mean + 3σ) or (mean – 3σ) respectively, where σ is a standard deviation, over the whole temperature range.

FIGURE 19-1: TYPICAL IDD vs. FOSC OVER VDD (HS MODE)

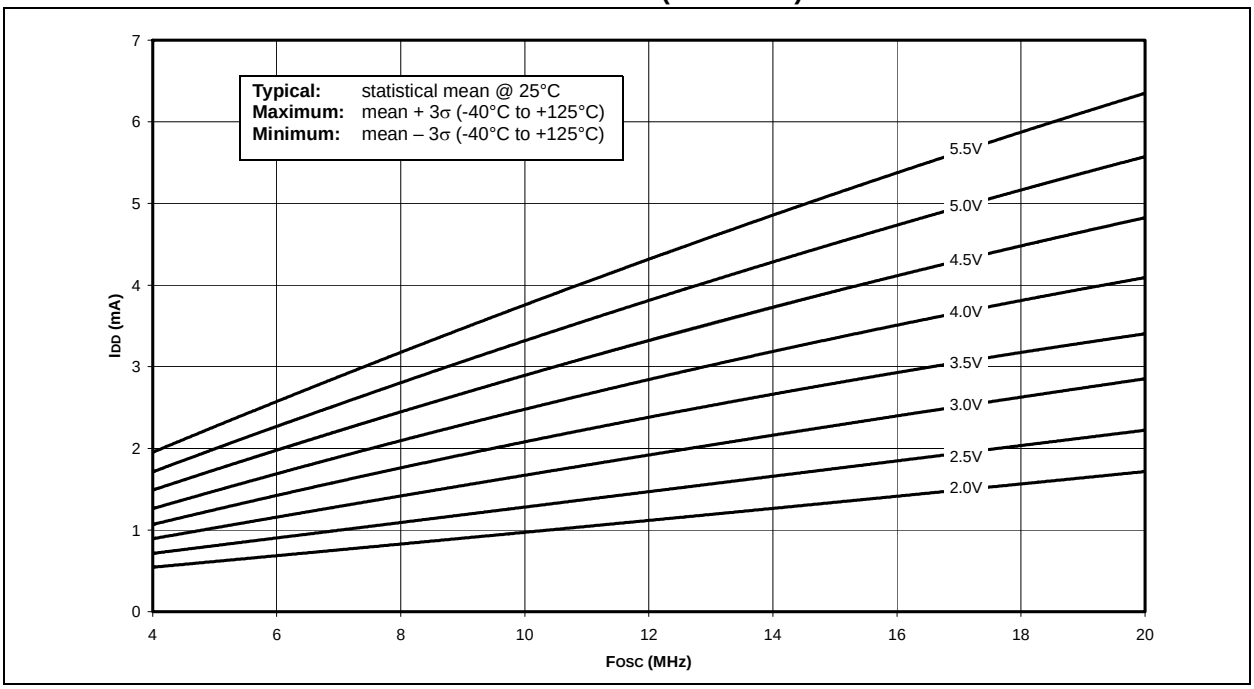


FIGURE 19-2: MAXIMUM IDD vs. FOSC OVER VDD (HS MODE)

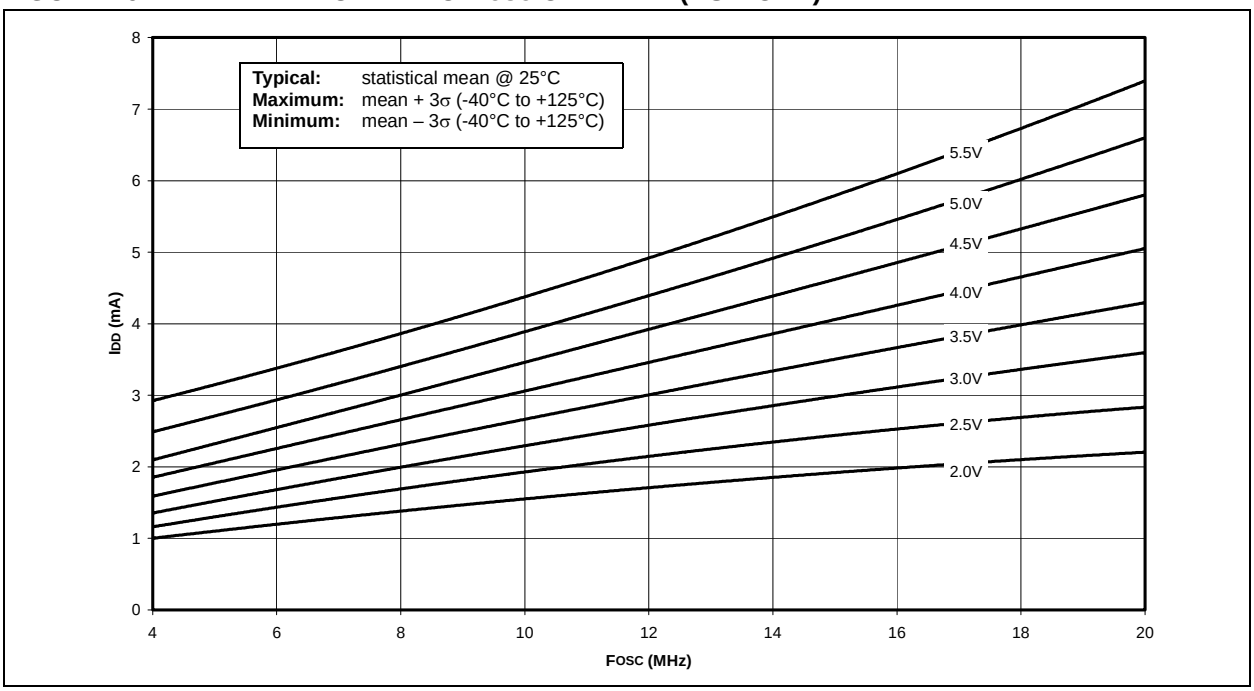


FIGURE 19-7: TYPICAL IDD vs. VDD, -40°C TO +125°C, 1 MHz TO 8 MHz (RC_RUN MODE, ALL PERIPHERALS DISABLED)

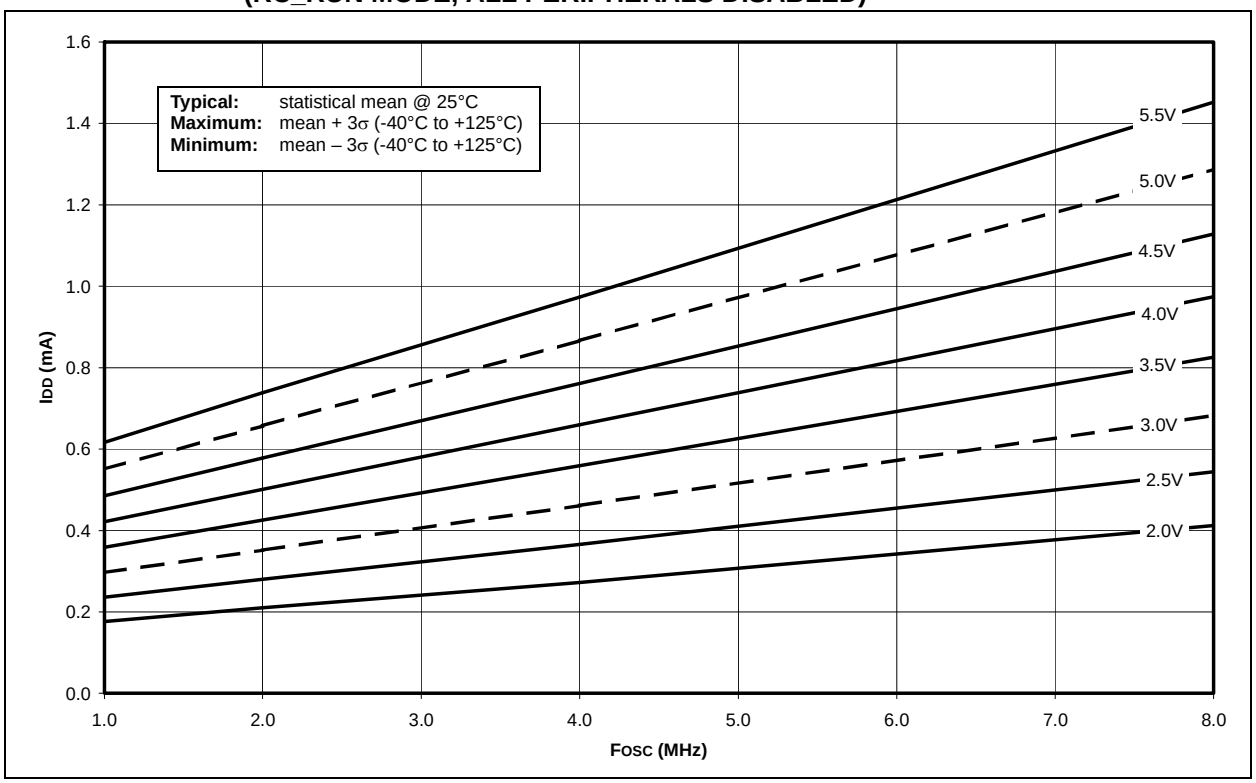
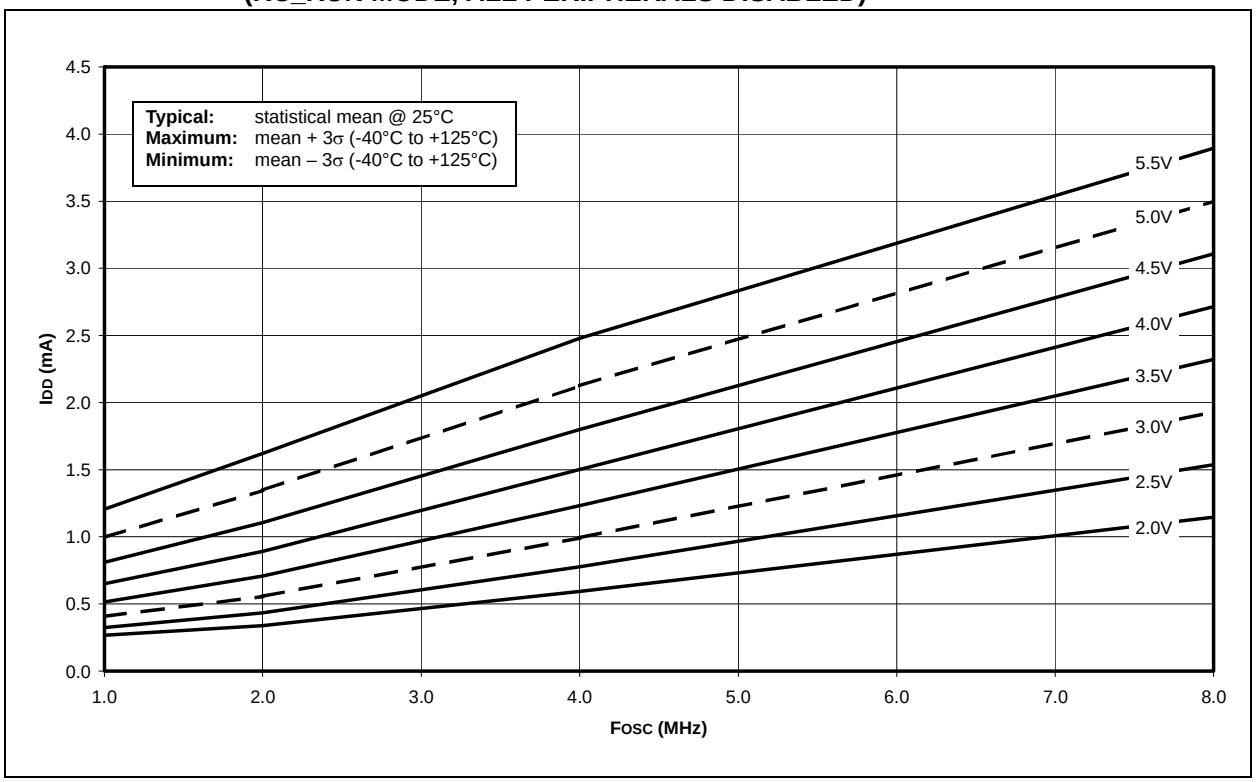
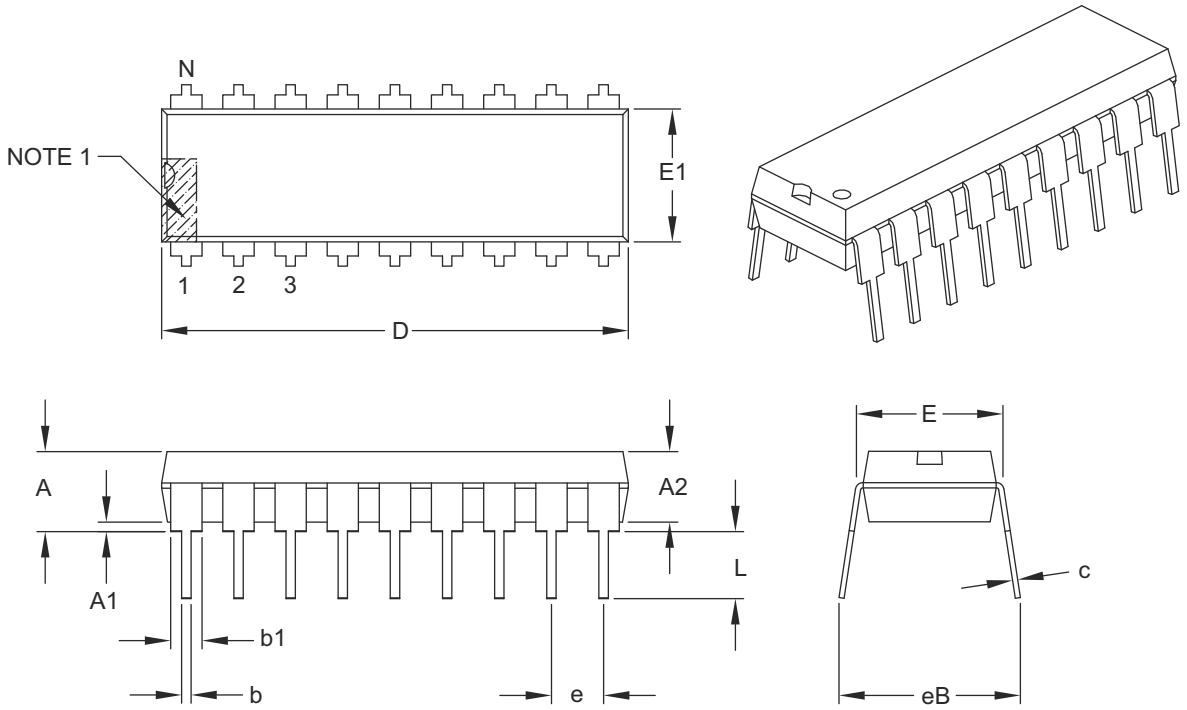


FIGURE 19-8: MAXIMUM IDD vs. VDD, -40°C TO +125°C, 1 MHz TO 8 MHz (RC_RUN MODE, ALL PERIPHERALS DISABLED)



18-Lead Plastic Dual In-Line (P) – 300 mil Body [PDIP]

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



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		18	
Pitch	e		.100 BSC	
Top to Seating Plane	A	–	–	.210
Molded Package Thickness	A2	.115	.130	.195
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.300	.310	.325
Molded Package Width	E1	.240	.250	.280
Overall Length	D	.880	.900	.920
Tip to Seating Plane	L	.115	.130	.150
Lead Thickness	c	.008	.010	.014
Upper Lead Width	b1	.045	.060	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	–	–	.430

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

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

Microchip Technology Drawing C04-007B

INDEX

A

B