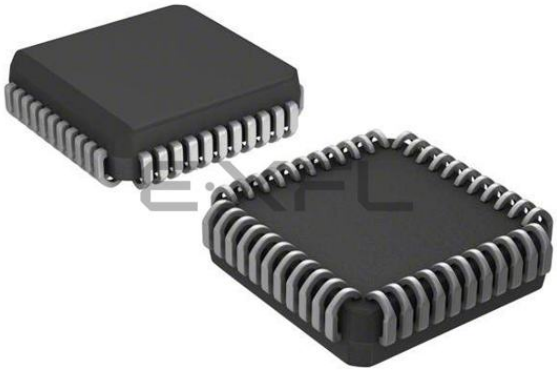


Welcome to E-XFL.COM

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	20MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	33
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	368 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	44-PLCC (16.59x16.59)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f877-20i-l

2.0 MEMORY ORGANIZATION

There are three memory blocks in each of the PIC16F87X MCUs. The Program Memory and Data Memory have separate buses so that concurrent access can occur and is detailed in this section. The EEPROM data memory block is detailed in Section 4.0.

Additional information on device memory may be found in the PIC® MCU Mid-Range Reference Manual, (DS33023).

2.1 Program Memory Organization

The PIC16F87X devices have a 13-bit program counter capable of addressing an 8K x 14 program memory space. The PIC16F877/876 devices have 8K x 14 words of FLASH program memory, and the PIC16F873/874 devices have 4K x 14. Accessing a location above the physically implemented address will cause a wraparound.

The RESET vector is at 0000h and the interrupt vector is at 0004h.

FIGURE 2-1: PIC16F877/876 PROGRAM MEMORY MAP AND STACK

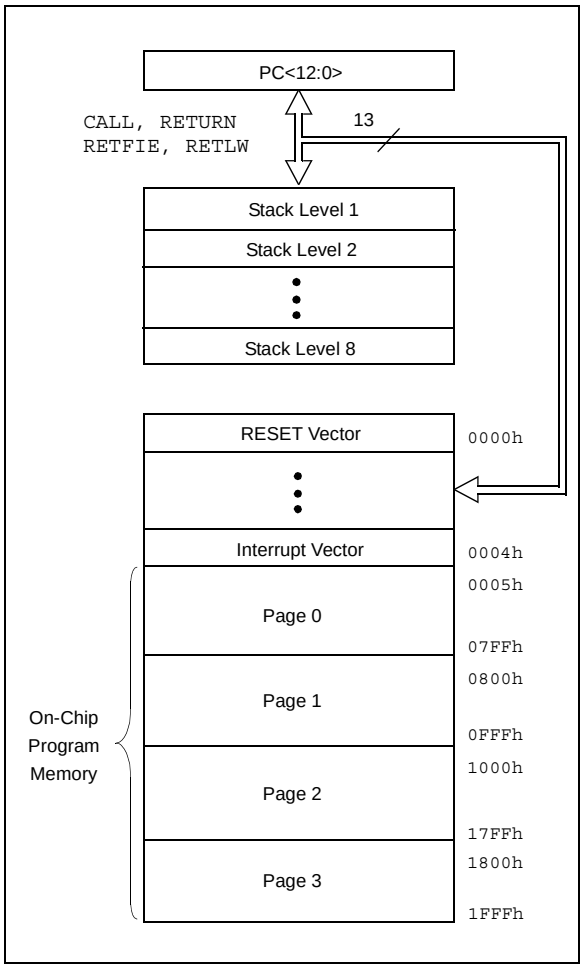
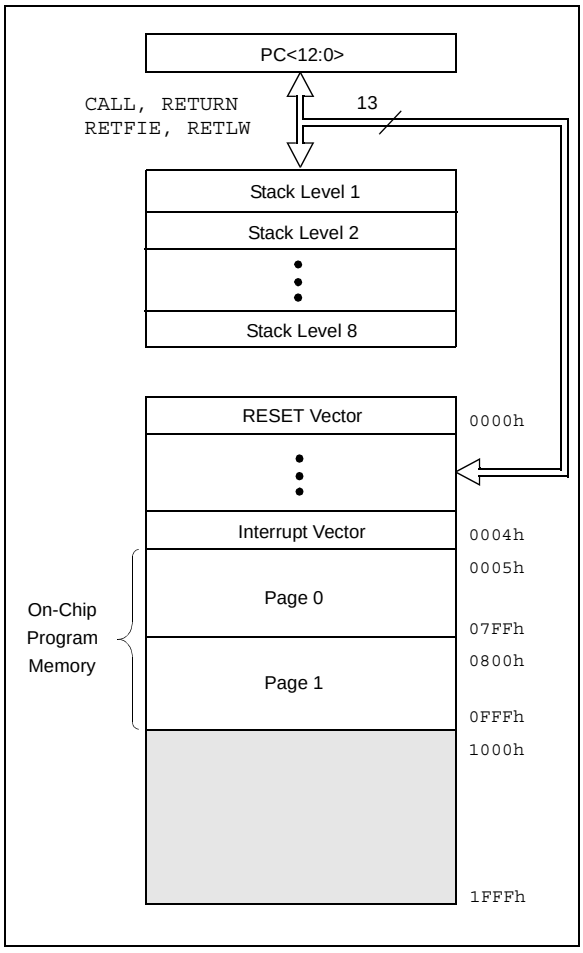


FIGURE 2-2: PIC16F874/873 PROGRAM MEMORY MAP AND STACK



2.2.2.8 PCON Register

The Power Control (PCON) Register contains flag bits to allow differentiation between a Power-on Reset (POR), a Brown-out Reset (BOR), a Watchdog Reset (WDT), and an external MCLR Reset.

Note:	BOR is unknown on POR. It must be set by the user and checked on subsequent RESETS to see if BOR is clear, indicating a brown-out has occurred. The BOR status bit is a “don’t care” and is not predictable if the brown-out circuit is disabled (by clearing the BODEN bit in the configuration word).

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

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-1
—	—	—	—	—	—	POR	BOR
bit 7							bit 0

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

bit 1 **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 **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

FIGURE 3-10: PARALLEL SLAVE PORT WRITE WAVEFORMS

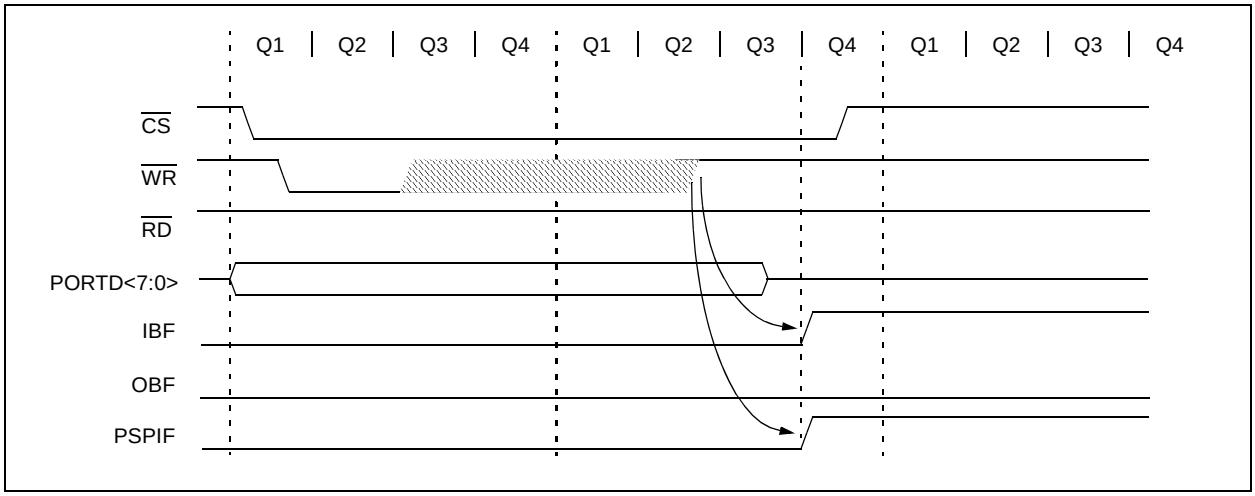


FIGURE 3-11: PARALLEL SLAVE PORT READ WAVEFORMS

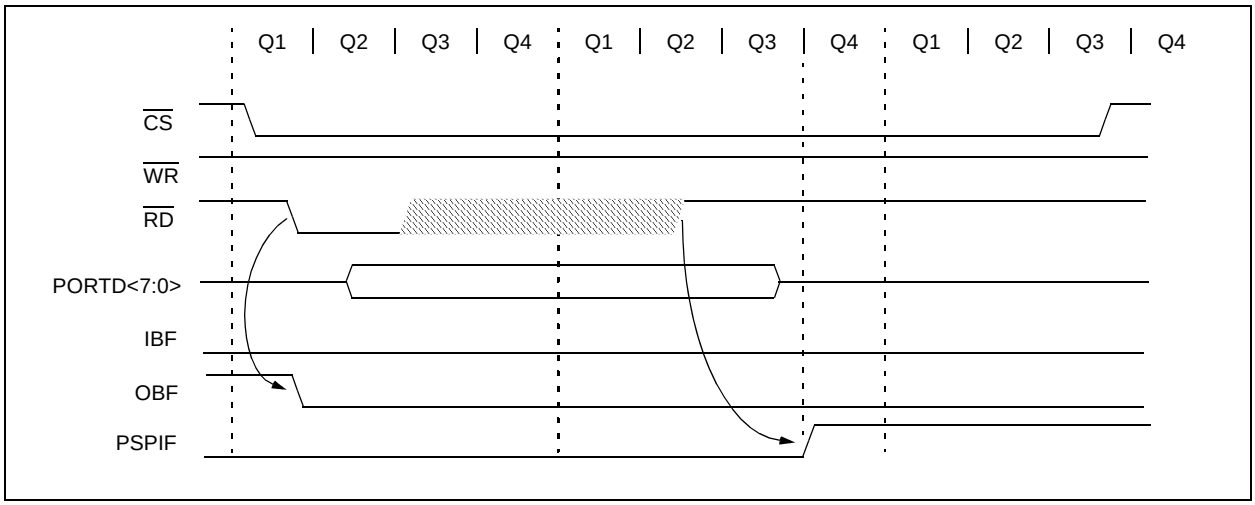


TABLE 3-11: REGISTERS ASSOCIATED WITH PARALLEL SLAVE PORT

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
08h	PORTD	Port Data Latch when written: Port pins when read								xxxx xxxx	uuuu uuuu
09h	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu
89h	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction Bits			0000 -111	0000 -111
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
9Fh	ADCON1	ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0	--0- 0000	--0- 0000

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by the Parallel Slave Port.
Note 1: Bits PSPIE and PSPIF are reserved on the PIC16F873/876; always maintain these bits clear.

4.4 Reading the FLASH Program Memory

Reading FLASH program memory is much like that of EEPROM data memory, only two NOP instructions must be inserted after the RD bit is set. These two instruction cycles that the NOP instructions execute, will be used by the microcontroller to read the data out of program memory and insert the value into the EEDATH:EEDATA registers. Data will be available following the second NOP instruction. EEDATH and EEDATA will hold their value until another read operation is initiated, or until they are written by firmware.

The steps to reading the FLASH program memory are:

1. Write the address to EEADRH:EEADR. Make sure that the address is not larger than the memory size of the PIC16F87X device.
2. Set the EEPGD bit to point to FLASH program memory.
3. Set the RD bit to start the read operation.
4. Execute two NOP instructions to allow the microcontroller to read out of program memory.
5. Read the data from the EEDATH:EEDATA registers.

EXAMPLE 4-3: FLASH PROGRAM READ

```
BSF     STATUS, RP1    ;
BCF     STATUS, RP0    ;Bank 2
MOVF    ADDRL, W       ;Write the
MOVWF   EEADR          ;address bytes
MOVF    ADDRH,W        ;for the desired
MOVWF   EEADRH         ;address to read
BSF     STATUS, RP0    ;Bank 3
BSF     EECON1, EEPGD  ;Point to Program memory
BSF     EECON1, RD     ;Start read operation
NOP                    ;Required two NOPs
NOP                    ;
BCF     STATUS, RP0    ;Bank 2
MOVF    EEDATA, W      ;DATAL = EEDATA
MOVWF   DATAL          ;
MOVF    EEDATH,W       ;DATAH = EEDATH
MOVWF   DATAH          ;
```

4.5 Writing to the FLASH Program Memory

Writing to FLASH program memory is unique, in that the microcontroller does not execute instructions while programming is taking place. The oscillator continues to run and all peripherals continue to operate and queue interrupts, if enabled. Once the write operation completes (specification D133), the processor begins executing code from where it left off. The other important difference when writing to FLASH program memory, is that the WRT configuration bit, when clear, prevents any writes to program memory (see Table 4-1).

Just like EEPROM data memory, there are many steps in writing to the FLASH program memory. Both address and data values must be written to the SFRs. The EEPGD bit must be set, and the WREN bit must be set to enable writes. The WREN bit should be kept clear at all times, except when writing to the FLASH Program memory. The WR bit can only be set if the WREN bit was set in a previous operation, i.e., they both cannot be set in the same operation. The WREN bit should then be cleared by firmware after the write. Clearing the WREN bit before the write actually completes will not terminate the write in progress.

Writes to program memory must also be prefaced with a special sequence of instructions that prevent inadvertent write operations. This is a sequence of five instructions that must be executed without interruption for each byte written. These instructions must then be followed by two NOP instructions to allow the microcontroller to setup for the write operation. Once the write is complete, the execution of instructions starts with the instruction after the second NOP.

The steps to write to program memory are:

1. Write the address to EEADRH:EEADR. Make sure that the address is not larger than the memory size of the PIC16F87X device.
2. Write the 14-bit data value to be programmed in the EEDATH:EEDATA registers.
3. Set the EEPGD bit to point to FLASH program memory.
4. Set the WREN bit to enable program operations.
5. Disable interrupts (if enabled).
6. 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
7. Execute two NOP instructions to allow the microcontroller to setup for write operation.
8. Enable interrupts (if using interrupts).
9. Clear the WREN bit to disable program operations.

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.) Since the microcontroller does not execute instructions during the write cycle, the firmware does not necessarily have to check either EEIF, or WR, to determine if the write had finished.

EXAMPLE 4-4: FLASH PROGRAM WRITE

```
BSF     STATUS, RP1   ;
BCF     STATUS, RP0   ;Bank 2
MOVF    ADDRL, W      ;Write address
MOVWF   EEADR         ;of desired
MOVF    ADDRH, W      ;program memory
MOVWF   EEADRH        ;location
MOVF    VALUEL, W     ;Write value to
MOVWF   EEDATA        ;program at
MOVF    VALUEH, W     ;desired memory
MOVWF   EEDATH        ;location
BSF     STATUS, RP0   ;Bank 3
BSF     EECON1, EEPGD ;Point to Program memory
BSF     EECON1, WREN  ;Enable writes
                      ;Only disable interrupts
BCF     INTCON, GIE   ;if already enabled,
                      ;otherwise discard
MOVLW   0x55          ;Write 55h to
MOVWF   EECON2        ;EECON2
MOVLW   0xAA          ;Write AAh to
MOVWF   EECON2        ;EECON2
BSF     EECON1, WR    ;Start write operation
NOP                   ;Two NOPs to allow micro
NOP                   ;to setup for write
                      ;Only enable interrupts
BSF     INTCON, GIE   ;if using interrupts,
                      ;otherwise discard

BCF     EECON1, WREN  ;Disable writes
```

4.6 Write Verify

The PIC16F87X devices do not automatically verify the value written during a write operation. Depending on the application, good programming practice may dictate that the value written to memory be verified against the original value. This should be used in applications where excessive writes can stress bits near the specified endurance limits.

4.7 Protection Against Spurious Writes

There are conditions when the device may not want to write to the EEPROM data memory or FLASH program memory. To protect against these spurious write conditions, various mechanisms have been built into the PIC16F87X devices. On power-up, the WREN bit is cleared and the Power-up Timer (if enabled) prevents writes.

The write initiate sequence, and the WREN bit together, help prevent any accidental writes during brown-out, power glitches, or firmware malfunction.

4.8 Operation While Code Protected

The PIC16F87X devices have two code protect mechanisms, one bit for EEPROM data memory and two bits for FLASH program memory. Data can be read and written to the EEPROM data memory, regardless of the state of the code protection bit, CPD. When code protection is enabled and CPD cleared, external access via ICSP is disabled, regardless of the state of the program memory code protect bits. This prevents the contents of EEPROM data memory from being read out of the device.

The state of the program memory code protect bits, CP0 and CP1, do not affect the execution of instructions out of program memory. The PIC16F87X devices can always read the values in program memory, regardless of the state of the code protect bits. However, the state of the code protect bits and the WRT bit will have different effects on writing to program memory. Table 4-1 shows the effect of the code protect bits and the WRT bit on program memory.

Once code protection has been enabled for either EEPROM data memory or FLASH program memory, only a full erase of the entire device will disable code protection.

7.1 Timer2 Prescaler and Postscaler

The prescaler and postscaler counters are cleared when any of the following occurs:

- a write to the TMR2 register
- a write to the T2CON register
- any device RESET (POR, $\overline{\text{MCLR}}$ Reset, WDT Reset, or BOR)

TMR2 is not cleared when T2CON is written.

7.2 Output of TMR2

The output of TMR2 (before the postscaler) is fed to the SSP module, which optionally uses it to generate shift clock.

TABLE 7-1: REGISTERS ASSOCIATED WITH TIMER2 AS A TIMER/COUNTER

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	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
11h	TMR2	Timer2 Module's Register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
92h	PR2	Timer2 Period Register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by the Timer2 module.
Note 1: Bits PSPIE and PSPIF are reserved on the PIC16F873/876; always maintain these bits clear.

9.0 MASTER SYNCHRONOUS SERIAL PORT (MSSP) MODULE

The Master Synchronous Serial Port (MSSP) module is a serial interface, useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, shift registers, display drivers, A/D converters, etc. The MSSP module can operate in one of two modes:

- Serial Peripheral Interface (SPI)
- Inter-Integrated Circuit (I^2C)

Figure 9-1 shows a block diagram for the SPI mode, while Figure 9-5 and Figure 9-9 show the block diagrams for the two different I^2C modes of operation.

The Application Note AN734, “Using the PIC® MCU SSP for Slave I^2C™ Communication” describes the slave operation of the MSSP module on the PIC16F87X devices. AN735, “Using the PIC® MCU MSSP Module for I^2C™ Communications” describes the master operation of the MSSP module on the PIC16F87X devices.

REGISTER 9-1: SSPSTAT: SYNC SERIAL PORT STATUS REGISTER (ADDRESS: 94h)

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF
bit 7							bit 0

bit 7 **SMP:** Sample bit
SPI Master mode:
1 = Input data sampled at end of data output time
0 = Input data sampled at middle of data output time
SPI Slave mode:
SMP must be cleared when SPI is used in slave mode
In I^2C Master or Slave mode:
1 = Slew rate control disabled for standard speed mode (100 kHz and 1 MHz)
0 = Slew rate control enabled for high speed mode (400 kHz)

bit 6 **CKE**: SPI Clock Edge Select (Figure 9-2, Figure 9-3 and Figure 9-4)
SPI mode:
For CKP = 0
1 = Data transmitted on rising edge of SCK
0 = Data transmitted on falling edge of SCK
For CKP = 1
1 = Data transmitted on falling edge of SCK
0 = Data transmitted on rising edge of SCK
In I^2C Master or Slave mode:
1 = Input levels conform to SMBus spec
0 = Input levels conform to I^2C specs

bit 5 **D/A**: Data/$\overline{\text{Address}}$ bit (I^2C mode only)
1 = Indicates that the last byte received or transmitted was data
0 = Indicates that the last byte received or transmitted was address

bit 4 **P**: STOP bit
(I^2C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared.)
1 = Indicates that a STOP bit has been detected last (this bit is '0' on RESET)
0 = STOP bit was not detected last

bit 3 **S**: START bit
(I^2C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared.)
1 = Indicates that a START bit has been detected last (this bit is '0' on RESET)
0 = START bit was not detected last

bit 2 **R/W**: Read/Write bit Information (I^2C mode only)
This bit holds the R/W bit information following the last address match. This bit is only valid from the address match to the next START bit, STOP bit or not ACK bit.
In I^2C Slave mode:
1 = Read
0 = Write
In I^2C Master mode:
1 = Transmit is in progress
0 = Transmit is not in progress
Logical OR of this bit with SEN, RSEN, PEN, RCEN, or ACKEN will indicate if the MSSP is in IDLE mode.

bit 1 **UA**: Update Address (10-bit I^2C mode only)
1 = Indicates that the user needs to update the address in the SSPADD register
0 = Address does not need to be updated

bit **BF**: Buffer Full Status bit
Receive (SPI and I^2C modes):
1 = Receive complete, SSPBUF is full
0 = Receive not complete, SSPBUF is empty
Transmit (I^2C mode only):
1 = Data transmit in progress (does not include the ACK and STOP bits), SSPBUF is full
0 = Data transmit complete (does not include the ACK and STOP bits), SSPBUF is empty

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

9.2.18 MULTI -MASTER COMMUNICATION, BUS COLLISION, AND BUS ARBITRATION

Multi-Master mode support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA, by letting SDA float high and another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin = '0', a bus collision has taken place. The master will set the Bus Collision Interrupt Flag, BCLIF and reset the I^2C port to its IDLE state (Figure 9-19).

If a transmit was in progress when the bus collision occurred, the transmission is halted, the BF flag is cleared, the SDA and SCL lines are de-asserted, and the SSPBUF can be written to. When the user services the bus collision Interrupt Service Routine, and if the I^2C bus is free, the user can resume communication by asserting a START condition.

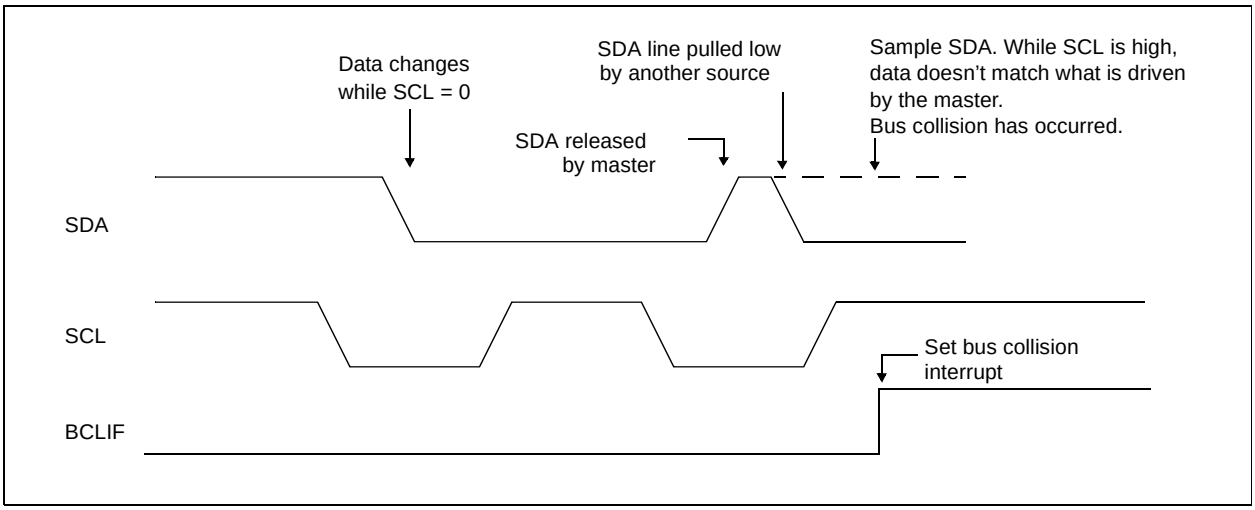
If a START, Repeated START, STOP, or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are de-asserted, and the respective control bits in the SSPCON2 register are cleared. When the user services the bus collision Interrupt Service Routine, and if the I^2C bus is free, the user can resume communication by asserting a START condition.

The master will continue to monitor the SDA and SCL pins and if a STOP condition occurs, the SSPIF bit will be set.

A write to the SSPBUF will start the transmission of data at the first data bit, regardless of where the transmitter left off when the bus collision occurred.

In Multi-Master mode, the interrupt generation on the detection of START and STOP conditions allows the determination of when the bus is free. Control of the I^2C bus can be taken when the P bit is set in the SSPSTAT register, or the bus is idle and the S and P bits are cleared.

FIGURE 9-19: BUS COLLISION TIMING FOR TRANSMIT AND ACKNOWLEDGE



9.3 Connection Considerations for I²C Bus

For standard-mode I²C bus devices, the values of resistors R_p and R_s in Figure 9-27 depend on the following parameters:

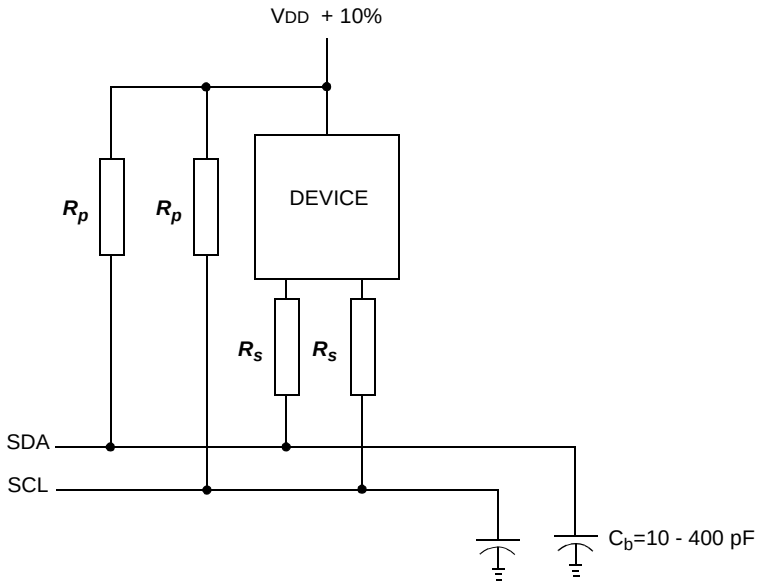
- Supply voltage
- Bus capacitance
- Number of connected devices (input current + leakage current)

The supply voltage limits the minimum value of resistor R_p, due to the specified minimum sink current of 3 mA at V_{OL} max = 0.4V, for the specified output stages. For example, with a supply voltage of V_{DD} = 5V±10% and V_{OL} max = 0.4V at 3 mA, R_p min = (5.5-0.4)/0.003 = 1.7 kΩ. V_{DD} as a function of R_p is shown in Figure 9-27. The desired noise margin of 0.1V_{DD} for the low level limits the maximum value of R_s. Series resistors are optional and used to improve ESD susceptibility.

The bus capacitance is the total capacitance of wire, connections, and pins. This capacitance limits the maximum value of R_p due to the specified rise time (Figure 9-27).

The SMP bit is the slew rate control enabled bit. This bit is in the SSPSTAT register, and controls the slew rate of the I/O pins when in I²C mode (master or slave).

FIGURE 9-27: SAMPLE DEVICE CONFIGURATION FOR I²C BUS



Note: I²C devices with input levels related to V_{DD} must have one common supply line to which the pull-up resistor is also connected.

12.3 RESET

The PIC16F87X differentiates between various kinds of RESET:

- Power-on Reset (POR)
- $\overline{MCLR}$ Reset during normal operation
- $\overline{MCLR}$ Reset during SLEEP
- WDT Reset (during normal operation)
- WDT Wake-up (during SLEEP)
- Brown-out Reset (BOR)

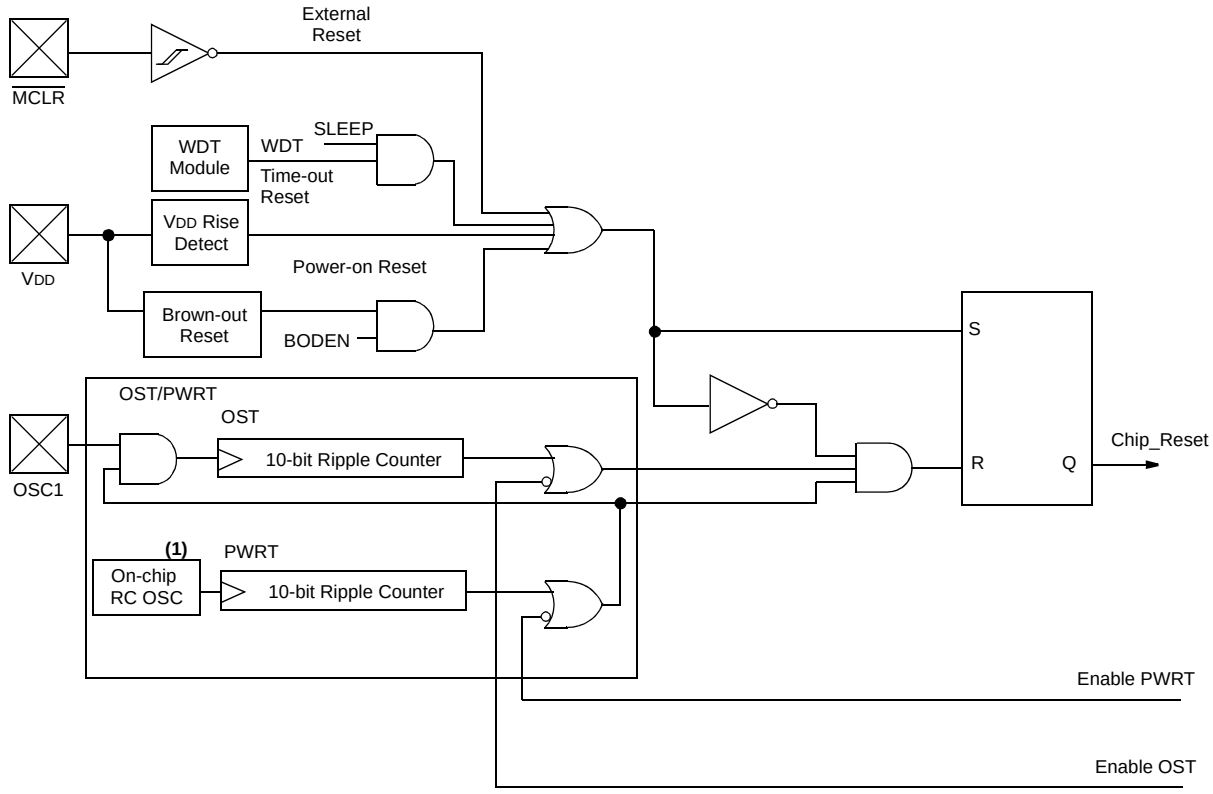
Some registers are not affected in any RESET condition. Their status is unknown on POR and unchanged in any other RESET. Most other registers are reset to a "RESET state" on Power-on Reset (POR), on the $\overline{MCLR}$ and WDT Reset, on $\overline{MCLR}$ Reset during SLEEP, and Brown-out Reset (BOR). They are not affected by a WDT Wake-up, which is viewed as the resumption of normal operation. The $\overline{TO}$ and $\overline{PD}$ bits are set or cleared differently in different RESET situations as indicated in Table 12-4. These bits are used in software to determine the nature of the RESET. See Table 12-6 for a full description of RESET states of all registers.

A simplified block diagram of the On-Chip Reset Circuit is shown in Figure 12-4.

These devices have a $\overline{MCLR}$ noise filter in the $\overline{MCLR}$ Reset path. The filter will detect and ignore small pulses.

It should be noted that a WDT Reset does not drive $\overline{MCLR}$ pin low.

FIGURE 12-4: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



Note 1: This is a separate oscillator from the RC oscillator of the CLKIN pin.

TABLE 12-6: INITIALIZATION CONDITIONS FOR ALL REGISTERS

Register	Devices				Power-on Reset, Brown-out Reset	$\overline{\text{MCLR}}$ Resets, WDT Reset	Wake-up via WDT or Interrupt
W	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	873	874	876	877	N/A	N/A	N/A
TMR0	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	873	874	876	877	0000h	0000h	PC + 1(2)
STATUS	873	874	876	877	0001 1xxx	000q quuu(3)	uuuq quuu(3)
FSR	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA	873	874	876	877	--0x 0000	--0u 0000	--uu uuuu
PORTB	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTC	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTD	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTE	873	874	876	877	---- -xxx	---- -uuu	---- -uuu
PCLATH	873	874	876	877	---0 0000	---0 0000	---u uuuu
INTCON	873	874	876	877	0000 000x	0000 000u	uuuu uuuu(1)
PIR1	873	874	876	877	r000 0000	r000 0000	ruuu uuuu(1)
	873	874	876	877	0000 0000	0000 0000	uuuu uuuu(1)
PIR2	873	874	876	877	-r-0 0--0	-r-0 0--0	-r-u u--u(1)
TMR1L	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	873	874	876	877	--00 0000	--uu uuuu	--uu uuuu
TMR2	873	874	876	877	0000 0000	0000 0000	uuuu uuuu
T2CON	873	874	876	877	-000 0000	-000 0000	-uuu uuuu
SSPBUF	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
SSPCON	873	874	876	877	0000 0000	0000 0000	uuuu uuuu
CCPR1L	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR1H	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP1CON	873	874	876	877	--00 0000	--00 0000	--uu uuuu
RCSTA	873	874	876	877	0000 000x	0000 000x	uuuu uuuu
TXREG	873	874	876	877	0000 0000	0000 0000	uuuu uuuu
RCREG	873	874	876	877	0000 0000	0000 0000	uuuu uuuu
CCPR2L	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR2H	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP2CON	873	874	876	877	0000 0000	0000 0000	uuuu uuuu
ADRESH	873	874	876	877	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0	873	874	876	877	0000 00-0	0000 00-0	uuuu uu-u
OPTION_REG	873	874	876	877	1111 1111	1111 1111	uuuu uuuu
TRISA	873	874	876	877	--11 1111	--11 1111	--uu uuuu
TRISB	873	874	876	877	1111 1111	1111 1111	uuuu uuuu
TRISC	873	874	876	877	1111 1111	1111 1111	uuuu uuuu
TRISD	873	874	876	877	1111 1111	1111 1111	uuuu uuuu
TRISE	873	874	876	877	0000 -111	0000 -111	uuuu -uuu
PIE1	873	874	876	877	r000 0000	r000 0000	ruuu uuuu
	873	874	876	877	0000 0000	0000 0000	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition, r = reserved, maintain clear

Note 1: One or more bits in INTCON, PIR1 and/or PIR2 will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

3: See Table 12-5 for RESET value for specific condition.

TABLE 14-1: DEVELOPMENT TOOLS FROM MICROCHIP

		PIC12CXXX	PIC14000	PIC16C5X	PIC16C6X	PIC16CXXX	PIC16F62X	PIC16C7X	PIC16C7XX	PIC16C8X	PIC16F8XX	PIC16C9XX	PIC17C4X	PIC17C7XX	PIC18CXX2	24CXX/ 25CXX/ 93CXX	HCSXXX	MCRFXXX	MCP2510
Software Tools	MPLAB® Integrated Development Environment	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
	MPLAB® C17 C Compiler												✓	✓					
	MPLAB® C18 C Compiler														✓				
	MPASM™ Assembler/ MPLINK™ Object Linker	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Emulators	MPLAB® ICE In-Circuit Emulator	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓				
	ICEPIC™ In-Circuit Emulator	✓		✓	✓	✓		✓	✓	✓		✓							
Debugger	MPLAB® ICD In-Circuit Debugger				✓*			✓*			✓								
Programmers	PICSTART® Plus Entry Level Development Programmer	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓				
	PRO MATE® II Universal Device Programmer	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Demo Boards and Eval Kits	PICDEM™ 1 Demonstration Board			✓		✓		✓†		✓			✓						
	PICDEM™ 2 Demonstration Board				✓†			✓†							✓				
	PICDEM™ 3 Demonstration Board											✓							
	PICDEM™ 14A Demonstration Board		✓																
	PICDEM™ 17 Demonstration Board													✓					
	KEELOQ® Evaluation Kit																✓		
	KEELOQ® Transponder Kit																✓		
	microID™ Programmer's Kit																	✓	
	125 kHz microID™ Developer's Kit																	✓	
	125 kHz Anticollision microID™ Developer's Kit																	✓	
	13.56 MHz Anticollision microID™ Developer's Kit																	✓	
	MCP2510 CAN Developer's Kit																		✓

* Contact the Microchip Technology Inc. web site at www.microchip.com for information on how to use the MPLAB® ICD In-Circuit Debugger (DV164001) with PIC16C62, 63, 64, 65, 72, 73, 74, 76, 77.
** Contact Microchip Technology Inc. for availability date.
† Development tool is available on select devices.

FIGURE 15-1: PIC16F87X-20 VOLTAGE-FREQUENCY GRAPH (COMMERCIAL AND INDUSTRIAL TEMPERATURE RANGES ONLY)

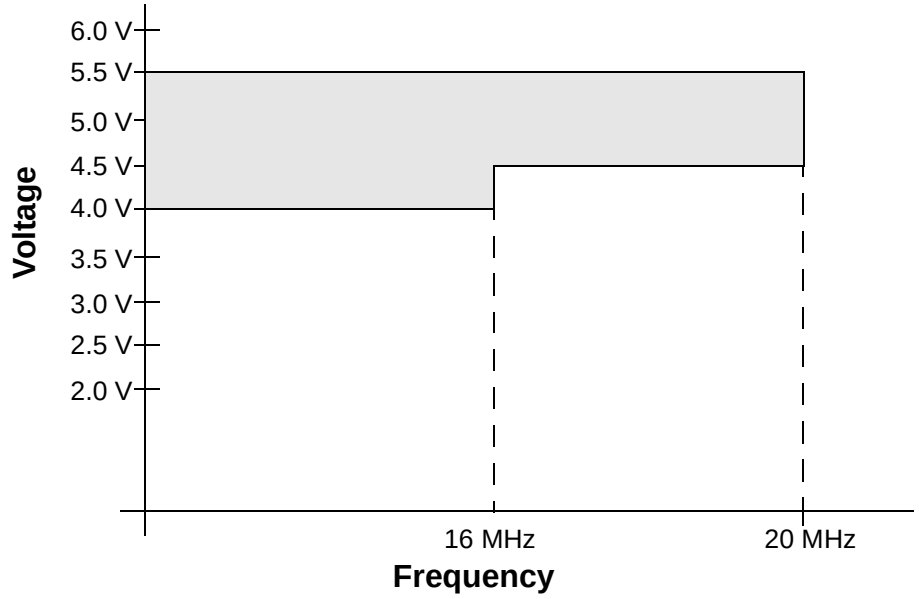
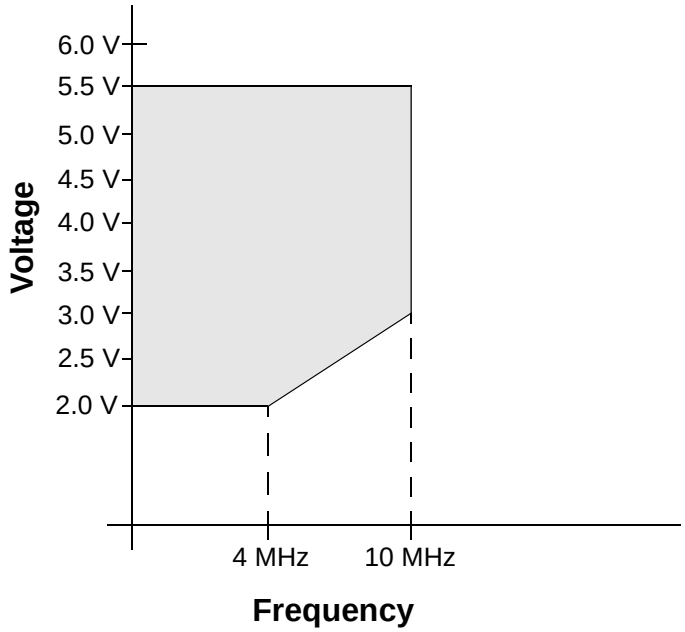


FIGURE 15-2: PIC16LF87X-04 VOLTAGE-FREQUENCY GRAPH (COMMERCIAL AND INDUSTRIAL TEMPERATURE RANGES ONLY)



FMAX = (6.0 MHz/V) (VDDAPPMIN - 2.0 V) + 4 MHz

Note 1: VDDAPPMIN is the minimum voltage of the PIC® device in the application.

2: FMAX has a maximum frequency of 10MHz.

FIGURE 15-3: PIC16F87X-04 VOLTAGE-FREQUENCY GRAPH (ALL TEMPERATURE RANGES)

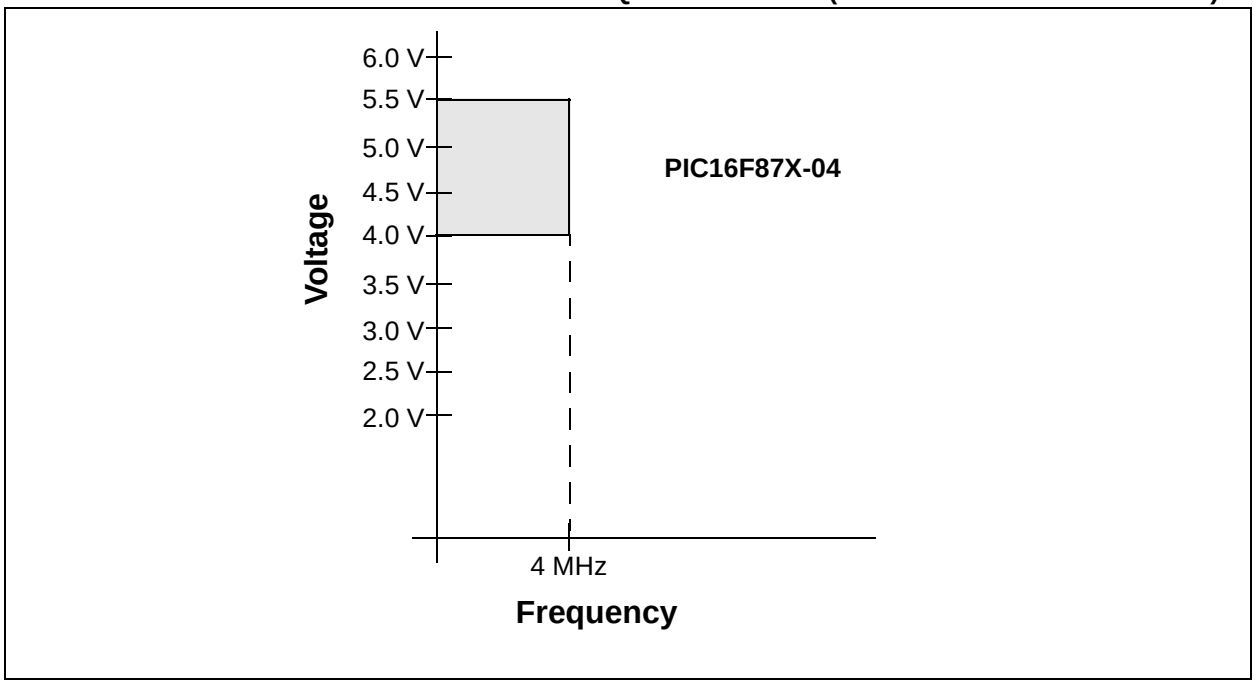
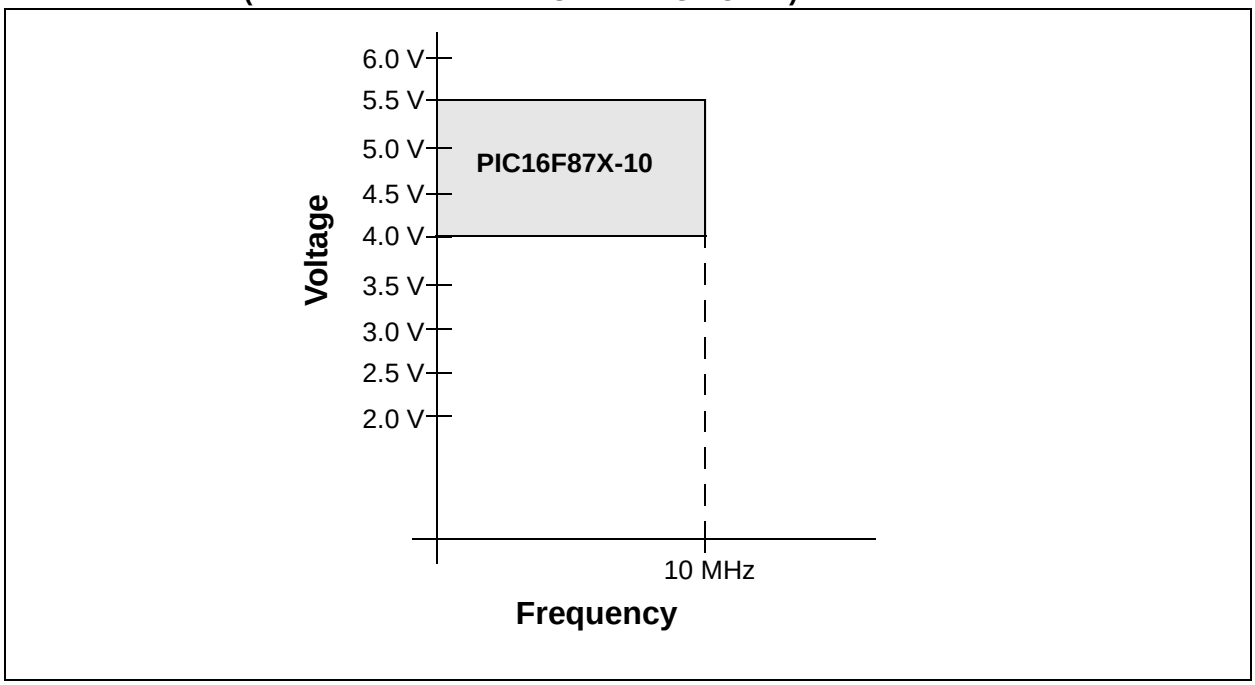


FIGURE 15-4: PIC16F87X-10 VOLTAGE-FREQUENCY GRAPH (EXTENDED TEMPERATURE RANGE ONLY)



15.1 DC Characteristics: PIC16F873/874/876/877-04 (Commercial, Industrial) PIC16F873/874/876/877-20 (Commercial, Industrial) PIC16LF873/874/876/877-04 (Commercial, Industrial) (Continued)

PIC16LF873/874/876/877-04 (Commercial, Industrial)			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial 0°C ≤ TA ≤ +70°C for commercial				
PIC16F873/874/876/877-04 PIC16F873/874/876/877-20 (Commercial, Industrial)			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial 0°C ≤ TA ≤ +70°C for commercial				
Param No.	**Symbol**	**Characteristic/ Device**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
	IPD	**Power-down Current(3,5)**					
D020		16LF87X	—	7.5	30	µA	VDD = 3.0V, WDT enabled, -40°C to +85°C
D020		16F87X	—	10.5	42	µA	VDD = 4.0V, WDT enabled, -40°C to +85°C
D021		16LF87X	—	0.9	5	µA	VDD = 3.0V, WDT enabled, 0°C to +70°C
D021		16F87X	—	1.5	16	µA	VDD = 4.0V, WDT enabled, -40°C to +85°C
D021A		16LF87X		0.9	5	µA	VDD = 3.0V, WDT enabled, -40°C to +85°C
D021A		16F87X		1.5	19	µA	VDD = 4.0V, WDT enabled, -40°C to +85°C
D023	ΔIBOR	**Brown-out Reset Current(6)**	—	85	200	µA	BOR enabled, VDD = 5.0V

Legend: Rows with standard voltage device data only are shaded for improved readability.

† Data in "Typ" column is at 5V, 25°C, unless otherwise stated. These parameters are for design guidance only, and are not tested.

Note 1: This is the limit to which VDD can be lowered 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, 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 hi-impedance state and tied to VDD and VSS.

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

5: Timer1 oscillator (when enabled) adds approximately 20 µA to the specification. This value is from characterization and is for design guidance only. This is not tested.

6: The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

7: When BOR is enabled, the device will operate correctly until the VBOR voltage trip point is reached.

15.3 DC Characteristics: PIC16F873/874/876/877-04 (Extended) PIC16F873/874/876/877-10 (Extended) (Continued)

PIC16F873/874/876/877-04 PIC16F873/874/876/877-20 (Extended)			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C				
Param No.	**Symbol**	**Characteristic/ Device**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
	IDD	**Supply Current(2,5)**					
D010			—	1.6	4	mA	RC osc configurations FOSC = 4 MHz, VDD = 5.5V
D013			—	7	15	mA	HS osc configuration, FOSC = 10 MHz, VDD = 5.5V
D015	ΔIBOR	**Brown-out Reset Current(6)**	—	85	200	μA	BOR enabled, VDD = 5.0V
	IPD	**Power-down Current(3,5)**					
D020A				10.5	60	μA	VDD = 4.0V, WDT enabled
D021B				1.5	30	μA	VDD = 4.0V, WDT disabled
D023	ΔIBOR	**Brown-out Reset Current(6)**	—	85	200	μA	BOR enabled, VDD = 5.0V

† Data is “Typ” column is at 5V, 25°C, unless otherwise stated. These parameters are for design guidance only, and are not tested.

Note 1: This is the limit to which VDD can be lowered 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, 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 hi-impedance state and tied to VDD and VSS.

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

5: Timer1 oscillator (when enabled) adds approximately 20 μA to the specification. This value is from characterization and is for design guidance only. This is not tested.

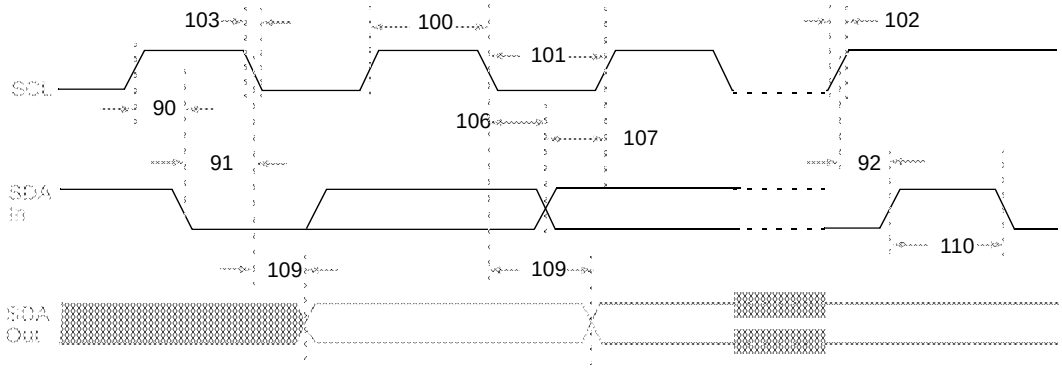
6: The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

7: When BOR is enabled, the device will operate correctly until the VBOR voltage trip point is reached.

TABLE 15-8: I²C BUS START/STOP BITS REQUIREMENTS

Parameter No.	Symbol	Characteristic		Min	Typ	Max	Units	Conditions
90	Tsu:sta	START condition Setup time	100 kHz mode	4700	—	—	ns	Only relevant for Repeated START condition
			400 kHz mode	600	—	—		
91	Thd:sta	START condition Hold time	100 kHz mode	4000	—	—	ns	After this period, the first clock pulse is generated
			400 kHz mode	600	—	—		
92	Tsu:sto	STOP condition Setup time	100 kHz mode	4700	—	—	ns	
			400 kHz mode	600	—	—		
93	Thd:sto	STOP condition Hold time	100 kHz mode	4000	—	—	ns	
			400 kHz mode	600	—	—		

FIGURE 15-18: I²C BUS DATA TIMING



Note: Refer to Figure 15-5 for load conditions.

FIGURE 16-9: AVERAGE FOSC vs. VDD FOR VARIOUS VALUES OF R (RC MODE, C = 300 pF, 25°C)

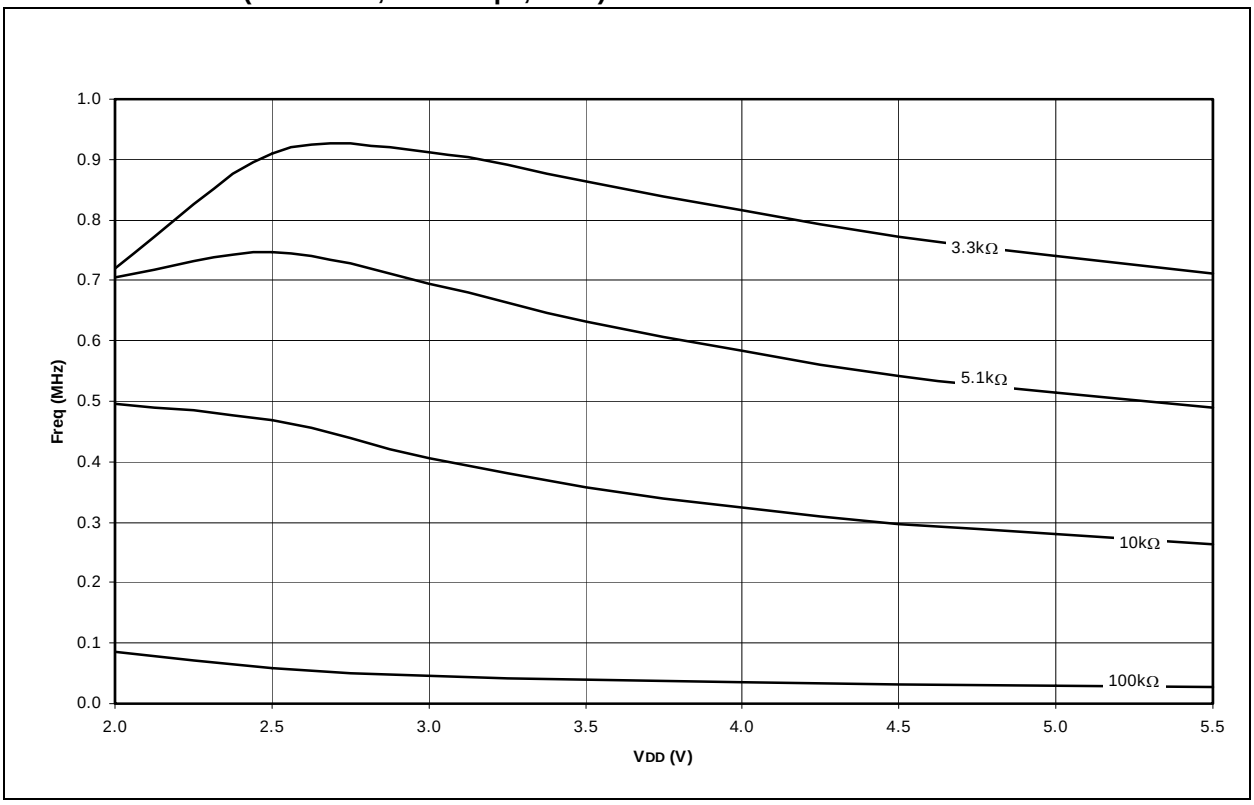
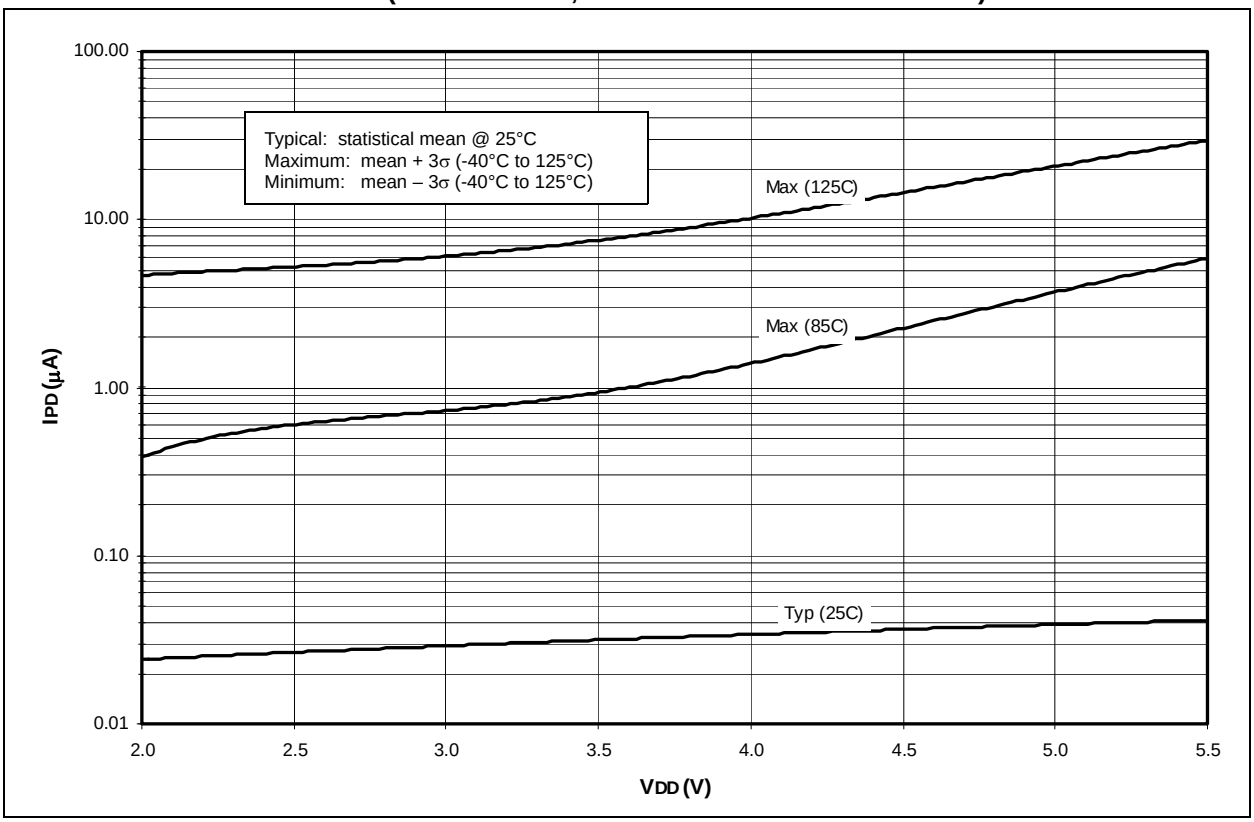
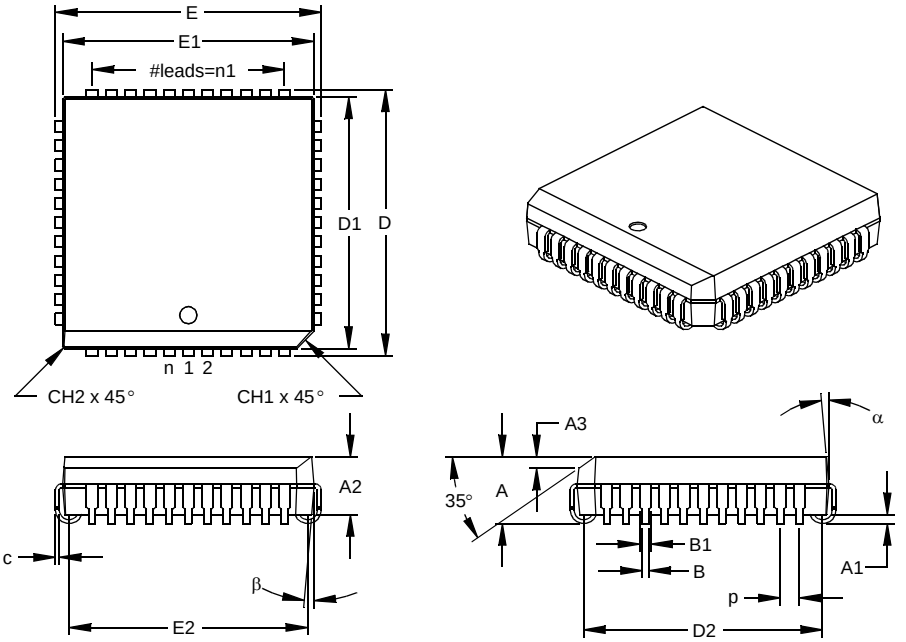


FIGURE 16-10: IPD vs. VDD (SLEEP MODE, ALL PERIPHERALS DISABLED)



44-Lead Plastic Leaded Chip Carrier (L) – Square (PLCC)

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



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		44			44	
Pitch	p		.050			1.27	
Pins per Side	n1		11			11	
Overall Height	A	.165	.173	.180	4.19	4.39	4.57
Molded Package Thickness	A2	.145	.153	.160	3.68	3.87	4.06
Standoff §	A1	.020	.028	.035	0.51	0.71	0.89
Side 1 Chamfer Height	A3	.024	.029	.034	0.61	0.74	0.86
Corner Chamfer 1	CH1	.040	.045	.050	1.02	1.14	1.27
Corner Chamfer (others)	CH2	.000	.005	.010	0.00	0.13	0.25
Overall Width	E	.685	.690	.695	17.40	17.53	17.65
Overall Length	D	.685	.690	.695	17.40	17.53	17.65
Molded Package Width	E1	.650	.653	.656	16.51	16.59	16.66
Molded Package Length	D1	.650	.653	.656	16.51	16.59	16.66
Footprint Width	E2	.590	.620	.630	14.99	15.75	16.00
Footprint Length	D2	.590	.620	.630	14.99	15.75	16.00
Lead Thickness	c	.008	.011	.013	0.20	0.27	0.33
Upper Lead Width	B1	.026	.029	.032	0.66	0.74	0.81
Lower Lead Width	B	.013	.020	.021	0.33	0.51	0.53
Mold Draft Angle Top	α	0	5	10	0	5	10
Mold Draft Angle Bottom	β	0	5	10	0	5	10

* Controlling Parameter
§ Significant Characteristic
Notes:
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010” (0.254mm) per side.
JEDEC Equivalent: MO-047
Drawing No. C04-048