

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.

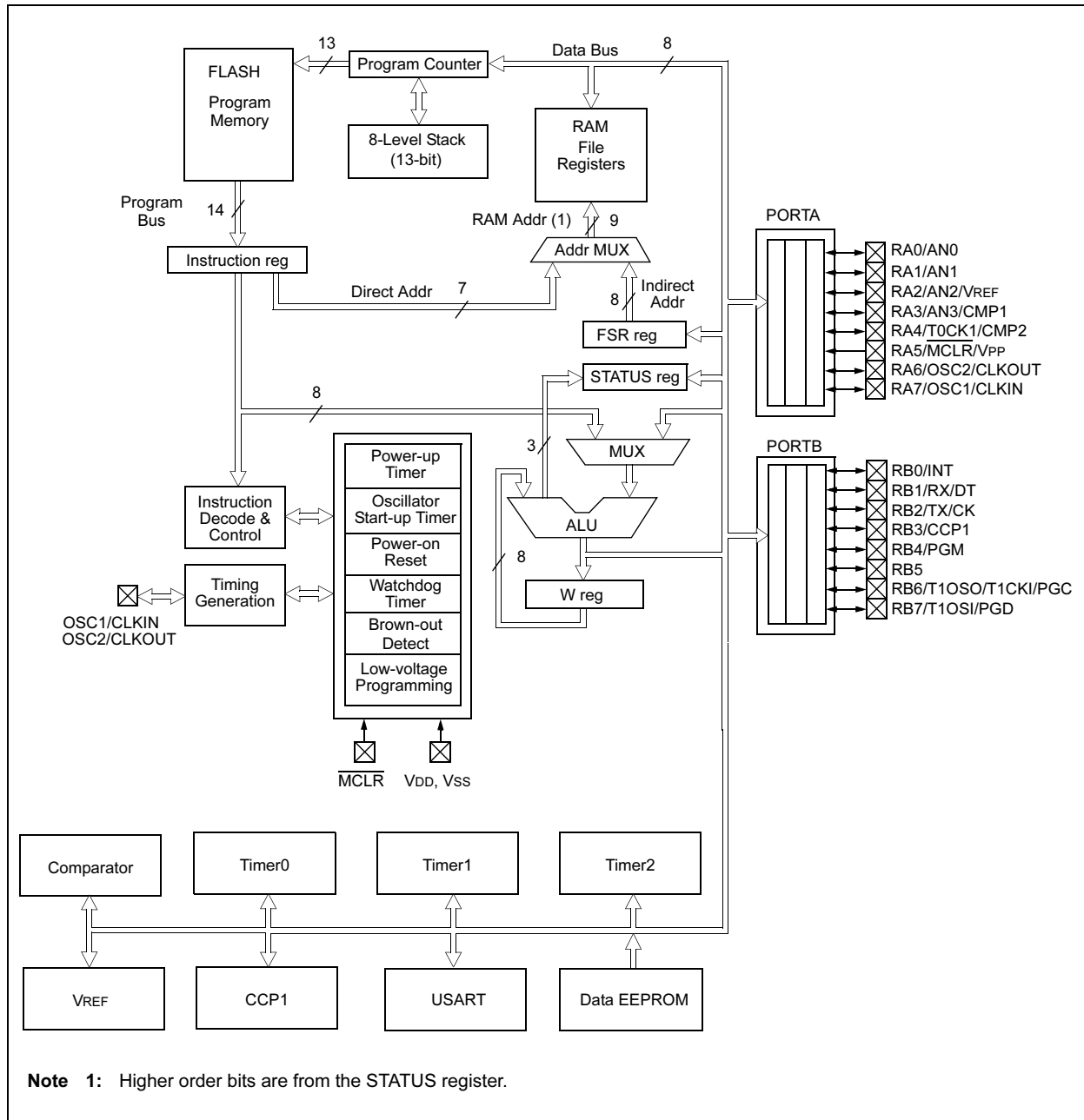
### Applications of "[Embedded - Microcontrollers](#)"

#### Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	20MHz
Connectivity	UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	16
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	224 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/pic16lf628at-i-ml">https://www.e-xfl.com/product-detail/microchip-technology/pic16lf628at-i-ml</a>

# PIC16F62X

**FIGURE 2-1: BLOCK DIAGRAM**



# PIC16F62X

## 3.2.2.4 PIE1 Register

This register contains interrupt enable bits.

### REGISTER 3-4: PIE1 REGISTER (ADDRESS: 8Ch)

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
EEIE	CMIE	RCIE	TXIE	—	CCP1IE	TMR2IE	TMR1IE
bit 7							bit 0

- bit 7 **EEIE:** EE Write Complete Interrupt Enable Bit  
1 = Enables the EE write complete interrupt  
0 = Disables the EE write complete interrupt
- bit 6 **CMIE:** Comparator Interrupt Enable bit  
1 = Enables the comparator interrupt  
0 = Disables the comparator 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 **Unimplemented:** Read as '0'
- 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

#### Legend:

R = Readable bit  
-n = Value at POR

W = Writable bit  
'1' = Bit is set

U = Unimplemented bit, read as '0'  
'0' = Bit is cleared    x = Bit is unknown

# PIC16F62X

## 3.2.2.6 PCON Register

The PCON register contains flag bits to differentiate between a Power-on Reset, an external MCLR Reset, WDT Reset or a Brown-out Detect.

**Note:**  $\overline{\text{BOD}}$  is unknown on Power-on Reset. It must then be set by the user and checked on subsequent RESETS to see if  $\overline{\text{BOD}}$  is cleared, indicating a brown-out has occurred. The BOD STATUS bit is a “don't care” and is not necessarily predictable if the brown-out circuit is disabled (by clearing the BODEN bit in the Configuration word).

### REGISTER 3-6: PCON REGISTER (ADDRESS: 0Ch)

U-0	U-0	U-0	U-0	R/W-1	U-0	R/W-q	R/W-q
—	—	—	—	OSCF	—	$\overline{\text{POR}}$	$\overline{\text{BOD}}$
bit 7							bit 0

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

bit 3 **OSCF:** INTRC/ER oscillator frequency

1 = 4 MHz typical<sup>(1)</sup>

0 = 37 KHz typical

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

bit 1  **$\overline{\text{POR}}$ :** 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  **$\overline{\text{BOD}}$ :** Brown-out Detect 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)

**Note 1:** When in ER Oscillator mode, setting OSCF = 1 will cause the oscillator frequency to change to the frequency specified by the external resistor.

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

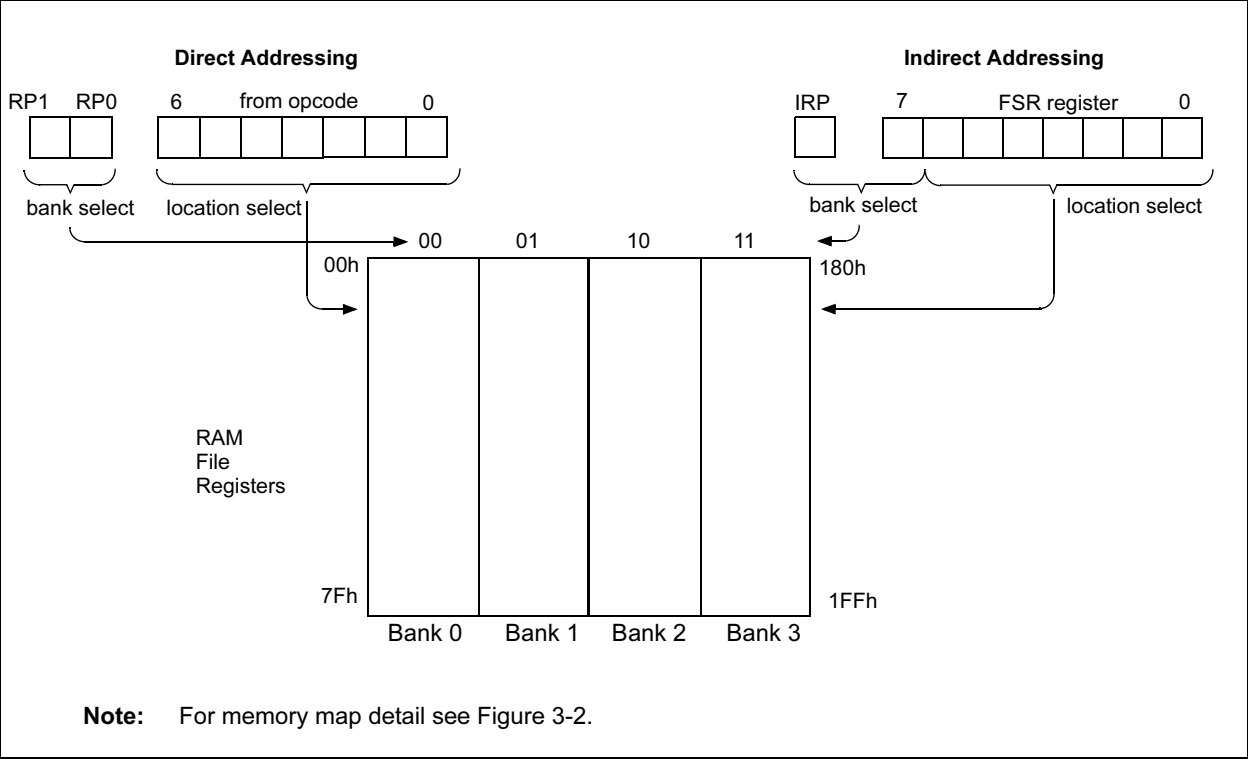
'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

# PIC16F62X

FIGURE 3-4: DIRECT/INDIRECT ADDRESSING PIC16F62X



## 4.0 GENERAL DESCRIPTION

The PIC16F62X are 18-Pin FLASH-based members of the versatile PIC16CXX family of low cost, high performance, CMOS, fully static, 8-bit microcontrollers.

All PICmicro® microcontrollers employ an advanced RISC architecture. The PIC16F62X have enhanced core features, eight-level deep stack, and multiple internal and external interrupt sources. The separate instruction and data buses of the Harvard architecture allow a 14-bit wide instruction word with the separate 8-bit wide data. The two-stage instruction pipeline allows all instructions to execute in a single cycle, except for program branches (which require two cycles). A total of 35 instructions (reduced instruction set) are available. Additionally, a large register set gives some of the architectural innovations used to achieve a very high performance.

PIC16F62X microcontrollers typically achieve a 2:1 code compression and a 4:1 speed improvement over other 8-bit microcontrollers in their class.

PIC16F62X devices have special features to reduce external components, thus reducing system cost, enhancing system reliability and reducing power consumption.

The PIC16F62X has eight oscillator configurations. The single pin ER oscillator provides a low cost solution. The LP oscillator minimizes power consumption, XT is a standard crystal, INTRC is a self-contained internal oscillator. The HS is for High Speed crystals. The EC mode is for an external clock source.

The SLEEP (Power-down) mode offers power savings. The user can wake-up the chip from SLEEP through several external interrupts, internal interrupts, and RESETS.

A highly reliable Watchdog Timer with its own on-chip RC oscillator provides protection against software lock-up.

Table 4-1 shows the features of the PIC16F62X mid-range microcontroller families.

A simplified block diagram of the PIC16F62X is shown in Figure 2.1.

The PIC16F62X series fits in applications ranging from battery chargers to low power remote sensors. The FLASH technology makes customization of application programs (detection levels, pulse generation, timers, etc.) extremely fast and convenient. The small footprint packages make this microcontroller series ideal for all applications with space limitations. Low cost, low power, high performance, ease of use and I/O flexibility make the PIC16F62X very versatile.

## 4.1 Development Support

The PIC16F62X family is supported by a full featured macro assembler, a software simulator, an in-circuit emulator, a low cost development programmer and a full-featured programmer. A Third Party "C" compiler support tool is also available.

**TABLE 4-1: PIC16F62X FAMILY OF DEVICES**

		PIC16F627	PIC16F628	PIC16LF627	PIC16LF628
Clock	Maximum Frequency of Operation (MHz)	20	20	4	4
Memory	FLASH Program Memory (words)	1024	2048	1024	2048
	RAM Data Memory (bytes)	224	224	224	224
	EEPROM Data Memory (bytes)	128	128	128	128
Peripherals	Timer Module(s)	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2
	Comparator(s)	2	2	2	2
	Capture/Compare/PWM modules	1	1	1	1
	Serial Communications	USART	USART	USART	USART
	Internal Voltage Reference	Yes	Yes	Yes	Yes
Features	Interrupt Sources	10	10	10	10
	I/O Pins	16	16	16	16
	Voltage Range (Volts)	3.0-5.5	3.0-5.5	2.0-5.5	2.0-5.5
	Brown-out Detect	Yes	Yes	Yes	Yes
	Packages	18-pin DIP, SOIC, 20-pin SSOP	18-pin DIP, SOIC, 20-pin SSOP	18-pin DIP, SOIC, 20-pin SSOP	18-pin DIP, SOIC, 20-pin SSOP

All PICmicro® Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability. All PIC16F62X Family devices use serial programming with clock pin RB6 and data pin RB7.

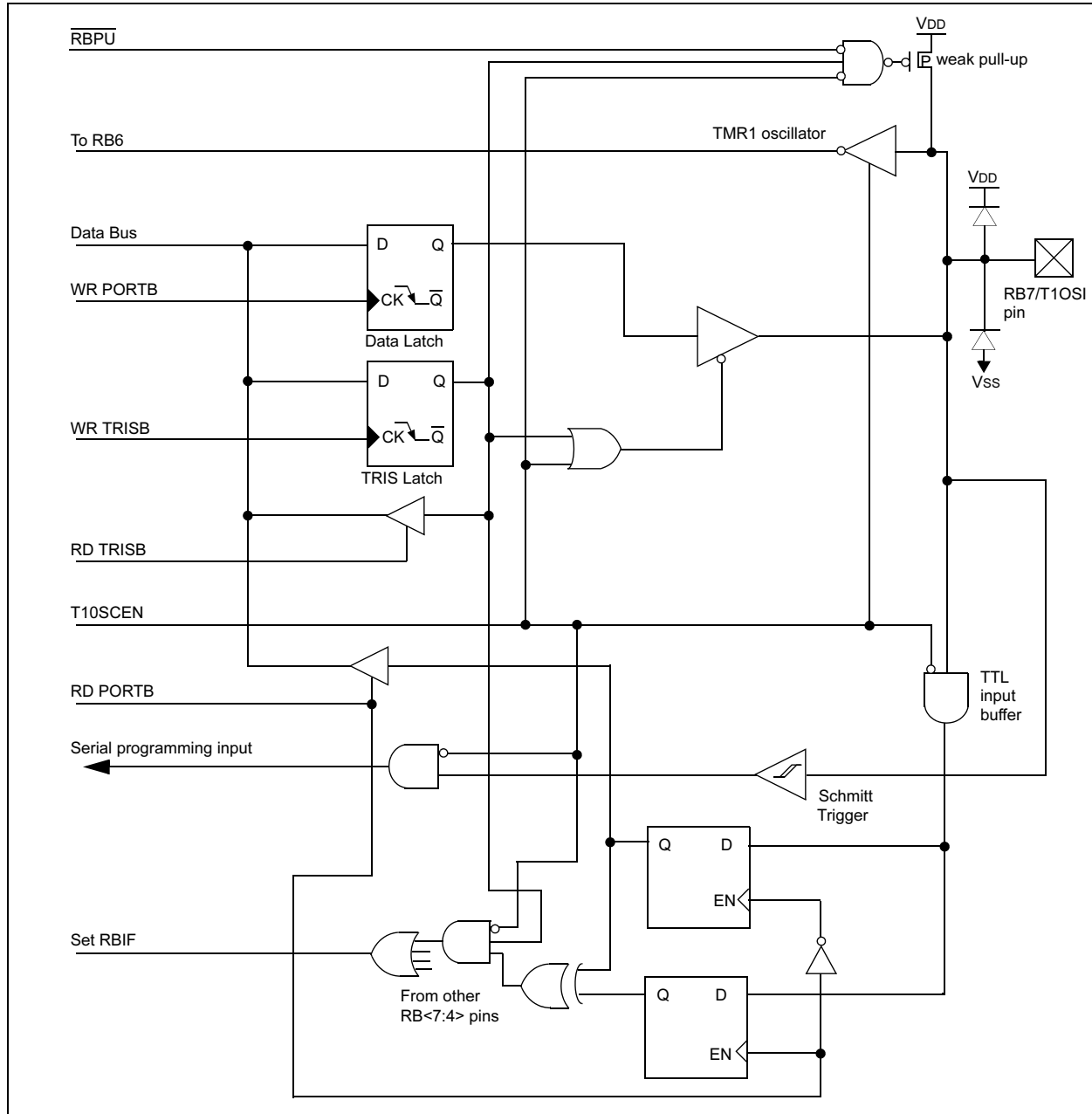


---



# PIC16F62X

**FIGURE 5-15: BLOCK DIAGRAM OF THE RB7/T10SI PIN**



## 9.6 Comparator Interrupts

The Comparator Interrupt flag is set whenever there is a change in the output value of either comparator. Software will need to maintain information about the status of the output bits, as read from CMCON<7:6>, to determine the actual change that has occurred. The CMIF bit, PIR1<6>, is the Comparator Interrupt Flag. The CMIF bit must be RESET by clearing '0'. Since it is also possible to write a '1' to this register, a simulated interrupt may be initiated.

The CMIE bit (PIE1<6>) and the PEIE bit (INTCON<6>) must be set to enable the interrupt. In addition, the GIE bit must also be set. If any of these bits are clear, the interrupt is not enabled, though the CMIF bit will still be set if an interrupt condition occurs.

**Note:** If a change in the CMCON register (C1OUT or C2OUT) should occur when a read operation is being executed (start of the Q2 cycle), then the CMIF (PIR1<6>) interrupt flag may not get set.

The user, in the interrupt service routine, can clear the interrupt in the following manner:

- a) Any write or read of CMCON. This will end the mismatch condition.
- b) Clear flag bit CMIF.

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

## 9.7 Comparator Operation During SLEEP

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

## 9.8 Effects of a RESET

A device RESET forces the CMCON register to its RESET state. This forces the Comparator module to be in the comparator RESET mode, CM2:CM0 = 000. This ensures that all potential inputs are analog inputs. Device current is minimized when analog inputs are present at RESET time. The comparators will be powered-down during the RESET interval.

## 9.9 Analog Input Connection Considerations

A simplified circuit for an analog input is shown in Figure 9-4. Since the analog pins are connected to a digital output, they have reverse biased diodes to VDD and VSS. The analog input therefore, must be between VSS and VDD. If the input voltage deviates from this range by more than 0.6V in either direction, one of the diodes is forward biased and a latchup may occur. A maximum source impedance of 10 kΩ is recommended for the analog sources. Any external component connected to an analog input pin, such as a capacitor or a Zener diode, should have very little leakage current.

# PIC16F62X

FIGURE 9-4: ANALOG INPUT MODE

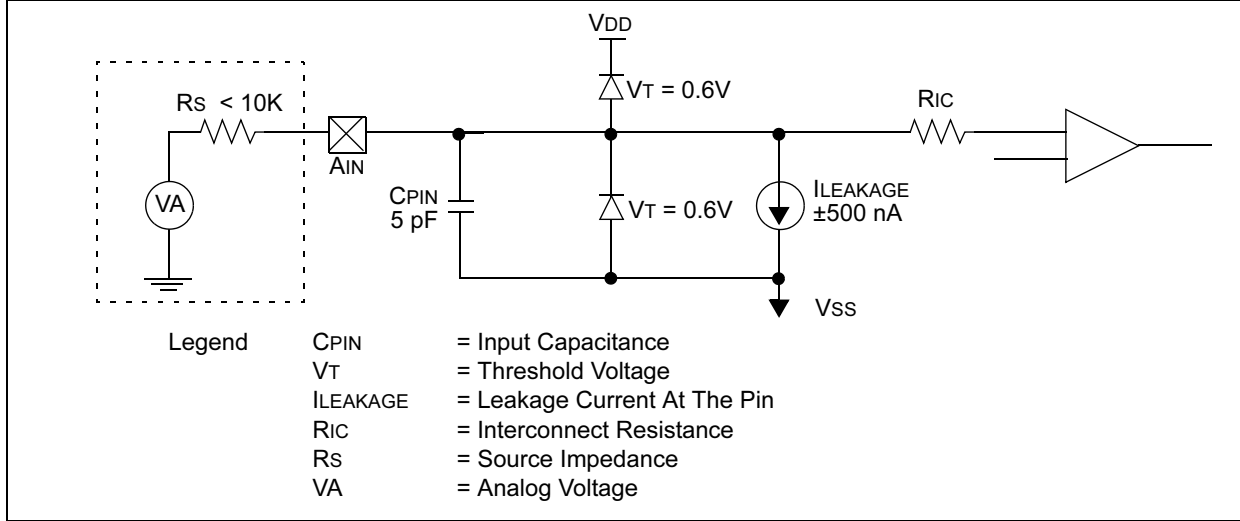


TABLE 9-1: REGISTERS ASSOCIATED WITH COMPARATOR MODULE

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on All Other RESETS
1Fh	CMCON	C2OUT	C1OUT	C2INV	C1NV	CIS	CM2	CM1	CM0	0000 0000	0000 0000
0Bh/8Bh/10Bh/18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	EEIF	CMIF	RCIF	TXIF	—	CCP1IF	TMR2IF	TMR1IF	0000 -000	0000 -000
8Ch	PIE1	EEIE	CMIE	RCIE	TXIE	—	CCP1IE	TMR2IE	TMR1IE	0000 -000	0000 -000
85h	TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111

Legend: x = Unknown, u = Unchanged, - = Unimplemented, read as '0'

## 11.2.2 TIMER1 MODE SELECTION

Timer1 must be running in Timer mode or Synchronized Counter mode if the CCP module is using the compare feature. In Asynchronous Counter mode, the compare operation may not work.

## 11.2.3 SOFTWARE INTERRUPT MODE

When generate software interrupt is chosen, the CCP1 pin is not affected. Only a CCP interrupt is generated (if enabled).

## 11.2.4 SPECIAL EVENT TRIGGER

In this mode, an internal hardware trigger is generated which may be used to initiate an action.

The special event trigger output of CCP1 resets the TMR1 register pair. This allows the CCPR1 register to effectively be a 16-bit programmable period register for Timer1.

**TABLE 11-3: REGISTERS ASSOCIATED WITH CAPTURE, COMPARE, AND TIMER1**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on all other RESETS
0Bh/8Bh/ 10Bh/ 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	EEIF	CMIF	RCIF	TXIF	—	CCP1IF	TMR2IF	TMR1IF	0000 -000	0000 -000
8Ch	PIE1	EEIE	CMIE	RCIE	TXIE	—	CCP1IE	TMR2IE	TMR1IE	0000 -000	0000 -000
87h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	--00 0000	--uu uuuu
15h	CCPR1L	Capture/Compare/PWM register1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM register1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by Capture and Timer1.

# PIC16F62X

**TABLE 12-5: BAUD RATES FOR ASYNCHRONOUS MODE (BRGH = 1)**

BAUD RATE (K)	Fosc = 20 MHz			16 MHz			10 MHz		
	KBAUD	ERROR	SPBRG value (decimal)	KBAUD	ERROR	SPBRG value (decimal)	KBAUD	ERROR	SPBRG value (decimal)
9600	9.615	+0.16%	129	9.615	+0.16%	103	9.615	+0.16%	64
19200	19.230	+0.16%	64	19.230	+0.16%	51	18.939	-1.36%	32
38400	37.878	-1.36%	32	38.461	+0.16%	25	39.062	+1.7%	15
57600	56.818	-1.36%	21	58.823	+2.12%	16	56.818	-1.36%	10
115200	113.636	-1.36%	10	111.111	-3.55%	8	125	+8.51%	4
250000	250	0	4	250	0	3	NA	—	—
625000	625	0	1	NA	—	—	625	0	0
1250000	1250	0	0	NA	—	—	NA	—	—

BAUD RATE (K)	Fosc = 7.16 MHz			5.068 MHz			4 MHz		
	KBAUD	ERROR	SPBRG value (decimal)	KBAUD	ERROR	SPBRG value (decimal)	KBAUD	ERROR	SPBRG value (decimal)
9600	9.520	-0.83%	46	9598.485	0.016%	32	9615.385	0.160%	25
19200	19.454	+1.32%	22	18632.35	-2.956%	16	19230.77	0.160%	12
38400	37.286	-2.90%	11	39593.75	3.109%	7	35714.29	-6.994%	6
57600	55.930	-2.90%	7	52791.67	-8.348%	5	62500	8.507%	3
115200	111.860	-2.90%	3	105583.3	-8.348%	2	125000	8.507%	1
250000	NA	—	—	316750	26.700%	0	250000	0.000%	0
625000	NA	—	—	NA	—	—	NA	—	—
1250000	NA	—	—	NA	—	—	NA	—	—

BAUD RATE (K)	Fosc = 3.579 MHz			1 MHz			32.768 MHz		
	KBAUD	ERROR	SPBRG value (decimal)	KBAUD	ERROR	SPBRG value (decimal)	KBAUD	ERROR	SPBRG value (decimal)
9600	9725.543	1.308%	22	8.928	-6.994%	6	NA	NA	NA
19200	18640.63	-2.913%	11	20833.3	8.507%	2	NA	NA	NA
38400	37281.25	-2.913%	5	31250	-18.620%	1	NA	NA	NA
57600	55921.88	-2.913%	3	62500	+8.507	0	NA	NA	NA
115200	111243.8	-2.913%	1	NA	—	—	NA	NA	NA
250000	223687.5	-10.525%	0	NA	—	—	NA	NA	NA
625000	NA	—	—	NA	—	—	NA	NA	NA
1250000	NA	—	—	NA	—	—	NA	NA	NA

## 12.5.2 USART SYNCHRONOUS SLAVE RECEPTION

The operation of the Synchronous Master and Slave modes is identical except in the case of the SLEEP mode. Also, bit SREN is a don't care in Slave mode.

If receive is enabled, by setting bit CREN, prior to the SLEEP instruction, then a word may be received during SLEEP. On completely receiving the word, the RSR register will transfer the data to the RCREG register and if enable bit RCIE bit is set, the interrupt generated will wake the chip from SLEEP. If the global interrupt is enabled, the program will branch to the interrupt vector (0004h).

Steps to follow when setting up a Synchronous Slave Reception:

1. Enable the synchronous master serial port by

setting bits SYNC and SPEN and clearing bit CSRC.

2. If interrupts are desired, then set enable bit RCIE.
3. If 9-bit reception is desired, then set bit RX9.
4. To enable reception, set enable bit CREN.
5. Flag bit RCIF will be set when reception is complete and an interrupt will be generated, if enable bit RCIE was set.
6. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
7. Read the 8-bit received data by reading the RCREG register.
8. If any error occurred, clear the error by clearing bit CREN.

**TABLE 12-11: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE TRANSMISSION**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on all other RESETS
0Ch	PIR1	EEIF	CMIF	RCIF	TXIF	—	CCP1IF	TMR2IF	TMR1IF	0000 -000	0000 -000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADEN	FERR	OERR	RX9D	0000 -00x	0000 -00x
19h	TXREG	USART Transmit Register								0000 0000	0000 0000
8Ch	PIE1	EEIE	CMIE	RCIE	TXIE	—	CCP1IE	TMR2IE	TMR1IE	0000 -000	0000 -000
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 read as '0'. Shaded cells are not used for Synchronous Slave Transmission.

**TABLE 12-12: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE RECEPTION**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on all other RESETS
0Ch	PIR1	EEIF	CMIF	RCIF	TXIF	—	CCP1IF	TMR2IF	TMR1IF	0000 -000	0000 -000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADEN	FERR	OERR	RX9D	0000 -00x	0000 -00x
1Ah	RCREG	USART Receive Register								0000 0000	0000 0000
8Ch	PIE1	EEIE	CMIE	RCIE	TXIE	—	CCP1IE	TMR2IE	TMR1IE	0000 -000	0000 -000
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 read as '0'. Shaded cells are not used for Synchronous Slave Reception.

## 13.3 READING THE EEPROM DATA MEMORY

To read a data memory location, the user must write the address to the EEADR register 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. EEDATA will hold this value until another read or until it is written to by the user (during a write operation).

### EXAMPLE 13-1: DATA EEPROM READ

```
BSF    STATUS, RP0    ; Bank 1
MOVLW  CONFIG_ADDR    ;
MOVWF  EEADR          ; Address to read
BSF    EECON1, RD     ; EE Read
MOVF   EEDATA, W       ; W = EEDATA
BCF    STATUS, RP0    ; Bank 0
```

## 13.4 WRITING TO THE EEPROM DATA 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 sequence to initiate the write for each byte.

### EXAMPLE 13-2: DATA EEPROM WRITE

Required Sequence	BSF	STATUS, RP0	; Bank 1
	BSF	EECON1, WREN	; Enable write
	BCF	INTCON, GIE	; Disable INTs.
	MOVLW	55h	;
	MOVWF	EECON2	; Write 55h
	MOVLW	AAh	;
	MOVWF	EECON2	; Write AAh
	BSF	EECON1, WR	; Set WR bit
			; begin write
	BSF	INTCON, GIE	; Enable INTs.

The write will not initiate if the above 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. A cycle count is executed during the required sequence. Any number that is not equal to the required cycles to execute the required sequence will cause the data not to be written into the EEPROM.

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. The EEIF bit in the PIR1 registers must be cleared by software.

## 13.5 WRITE VERIFY

Depending on the application, good programming practice may dictate that the value written to the Data EEPROM should be verified (Example 13-3) to the desired value to be written. This should be used in applications where an EEPROM bit will be stressed near the specification limit.

### EXAMPLE 13-3: WRITE VERIFY

```
BSF    STATUS, RP0    ; Bank 1
MOVF   EEDATA, W
BSF    EECON1, RD     ; Read the
                        ; value written
;
; Is the value written (in W reg) and
; read (in EEDATA) the same?
;
SUBWF  EEDATA, W      ;
BCF    STATUS, RP0    ; Bank0
BTFSS  STATUS, Z      ; Is difference 0?
GOTO   WRITE_ERR     ; NO, Write error
:      ; YES, Good write
:      ; Continue program
```

## 13.6 PROTECTION AGAINST SPURIOUS WRITE

There are conditions when the device may not want to write to the data EEPROM memory. To protect against spurious EEPROM writes, various mechanisms have been built in. On power-up, WREN is cleared. Also, the Power-up Timer (72 ms duration) prevents EEPROM write.

The write initiate sequence, and the WREN bit together help prevent an accidental write during brown-out, power glitch, or software malfunction.

## 13.7 DATA EEPROM OPERATION DURING CODE PROTECT

When the device is code protected, the CPU is able to read and write unscrambled data to the Data EEPROM.

## 15.0 INSTRUCTION SET SUMMARY

Each PIC16F62X instruction is a 14-bit word divided into an OPCODE which specifies the instruction type and one or more operands which further specify the operation of the instruction. The PIC16F62X instruction set summary in Table 15-2 lists byte-oriented, bit-oriented, and literal and control operations. Table 15-1 shows the opcode field descriptions.

For byte-oriented instructions, 'f' represents a file register designator and 'd' represents a destination designator. The file register designator specifies which file register is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is zero, the result is placed in the W register. If 'd' is one, the result is placed in the file register specified in the instruction.

For bit-oriented instructions, 'b' represents a bit field designator which selects the number of the bit affected by the operation, while 'f' represents the number of the file in which the bit is located.

For literal and control operations, 'k' represents an eight or eleven bit constant or literal value.

**TABLE 15-1: OPCODE FIELD DESCRIPTIONS**

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1) The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1
label	Label name
TOS	Top of Stack
PC	Program Counter
PCLA	Program Counter High Latch
TH	
GIE	Global Interrupt Enable bit
WDT	Watchdog Timer/Counter
TO	Timeout bit
PD	Power-down bit
dest	Destination either the W register or the specified register file location
[ ]	Options
( )	Contents
→	Assigned to
< >	Register bit field
∈	In the set of
italics	User defined term (font is courier)

The instruction set is highly orthogonal and is grouped into three basic categories:

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal and control** operations

All instructions are executed within one single instruction cycle, unless a conditional test is true or the program counter is changed as a result of an instruction. In this case, the execution takes two instruction cycles with the second cycle executed as a NOP. One instruction cycle consists of four oscillator periods. Thus, for an oscillator frequency of 4 MHz, the normal instruction execution time is 1 μs. If a conditional test is true or the program counter is changed as a result of an instruction, the instruction execution time is 2 μs.

Table 15-2 lists the instructions recognized by the MPASM™ assembler.

Figure 15-1 shows the three general formats that the instructions can have.

**Note 1:** Any unused opcode is reserved. Use of any reserved opcode may cause unexpected operation.

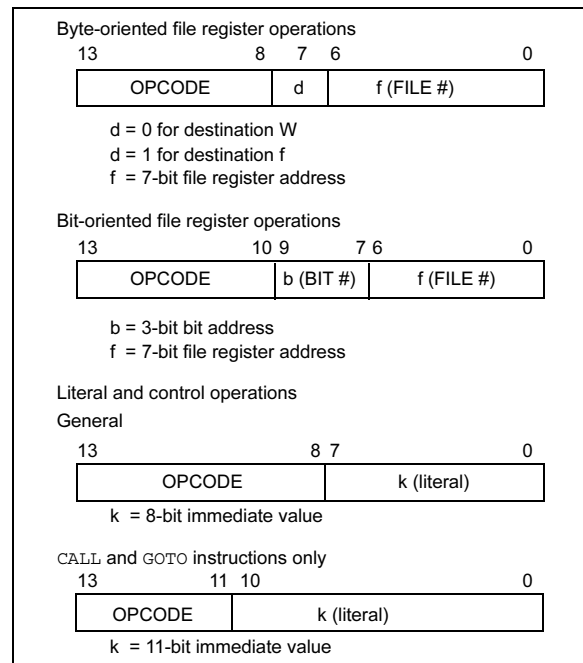
**2:** To maintain upward compatibility with future PICmicro® products, do not use the OPTION and TRIS instructions.

All examples use the following format to represent a hexadecimal number:

0xhh

where h signifies a hexadecimal digit.

**FIGURE 15-1: GENERAL FORMAT FOR INSTRUCTIONS**





## BTFSS Bit Test f, Skip if Set

Syntax:	[ <i>label</i> ] BTFSS <i>f</i> , <i>b</i>		
Operands:	$0 \leq f \leq 127$ $0 \leq b < 7$		
Operation:	skip if ( <i>f</i> < <i>b</i> >) = 1		
Status Affected:	None		
Encoding:	01	11bb	bfff ffff
Description:	<p>If bit 'b' in register 'f' is '1' then the next instruction is skipped.</p> <p>If bit 'b' is '1', then the next instruction fetched during the current instruction execution, is discarded and a NOP is executed instead, making this a two-cycle instruction.</p>		
Words:	1		
Cycles:	1(2)		
Example	<pre> HERE    BTFSS    REG1 FALSE   GOTO     PROCESS_CODE TRUE    •         •         •  Before Instruction       PC = address HERE After Instruction       if FLAG&lt;1&gt; = 0,       PC = address FALSE if FLAG&lt;1&gt; = 1,       PC = address TRUE </pre>		

## CALL Call Subroutine

Syntax:	[ <i>label</i> ] CALL <i>k</i>		
Operands:	$0 \leq k \leq 2047$		
Operation:	(PC)+ 1 → TOS, k → PC<10:0>, (PCLATH<4:3>) → PC<12:11>		
Status Affected:	None		
Encoding:	10	0kkk	kkkk kkkk
Description:	<p>Call Subroutine. First, return address (PC+1) is pushed onto the stack. The eleven bit immediate address is loaded into PC bits &lt;10:0&gt;. The upper bits of the PC are loaded from PCLATH. CALL is a two-cycle instruction.</p>		
Words:	1		
Cycles:	2		
Example	<pre> HERE    CALL    THERE  Before Instruction       PC = Address HERE After Instruction       PC = Address THERE       TOS = Address HERE+1 </pre>		

## CLRF Clear f

Syntax:	[ <i>label</i> ] CLRF <i>f</i>		
Operands:	$0 \leq f \leq 127$		
Operation:	00h → ( <i>f</i> ) 1 → Z		
Status Affected:	Z		
Encoding:	00	0001	1fff ffff
Description:	<p>The contents of register 'f' are cleared and the Z bit is set.</p>		
Words:	1		
Cycles:	1		
Example	<pre> CLRF    REG1  Before Instruction       REG1 = 0x5A After Instruction       REG1 = 0x00       Z    = 1 </pre>		

# PIC16F62X

## CLRW

## Clear W

Syntax: [ *label* ] CLRW

Operands: None

Operation: 00h → (W)  
1 → Z

Status Affected: Z

Encoding: 

00	0001	0000	0011
----	------	------	------

Description: W register is cleared. Zero bit (Z) is set.

Words: 1

Cycles: 1

Example CLRW

Before Instruction  
W = 0x5A

After Instruction  
W = 0x00  
Z = 1

## COMF

## Complement f

Syntax: [ *label* ] COMF f,d

Operands:  $0 \leq f \leq 127$   
 $d \in [0,1]$

Operation:  $(\bar{f}) \rightarrow (\text{dest})$

Status Affected: Z

Encoding: 

00	1001	dfff	ffff
----	------	------	------

Description: The contents of register 'f' are complemented. If 'd' is 0 the result is stored in W. If 'd' is 1 the result is stored back in register 'f'.

Words: 1

Cycles: 1

Example COMF REG1, 0

Before Instruction  
REG1 = 0x13  
After Instruction  
REG1 = 0x13  
W = 0xEC

## CLRWDT

## Clear Watchdog Timer

Syntax: [ *label* ] CLRWDT

Operands: None

Operation: 00h → WDT  
0 → WDT prescaler,  
1 →  $\overline{\text{TO}}$   
1 →  $\overline{\text{PD}}$

Status Affected:  $\overline{\text{TO}}$ ,  $\overline{\text{PD}}$

Encoding: 

00	0000	0110	0100
----	------	------	------

Description: CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. STATUS bits  $\overline{\text{TO}}$  and  $\overline{\text{PD}}$  are set.

Words: 1

Cycles: 1

Example CLRWDT

Before Instruction  
WDT counter = ?

After Instruction  
WDT counter = 0x00  
WDT prescaler = 0  
 $\overline{\text{TO}}$  = 1  
 $\overline{\text{PD}}$  = 1

## DECF

## Decrement f

Syntax: [ *label* ] DECF f,d

Operands:  $0 \leq f \leq 127$   
 $d \in [0,1]$

Operation:  $(f) - 1 \rightarrow (\text{dest})$

Status Affected: Z

Encoding: 

00	0011	dfff	ffff
----	------	------	------

Description: Decrement register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is 1 the result is stored back in register 'f'.

Words: 1

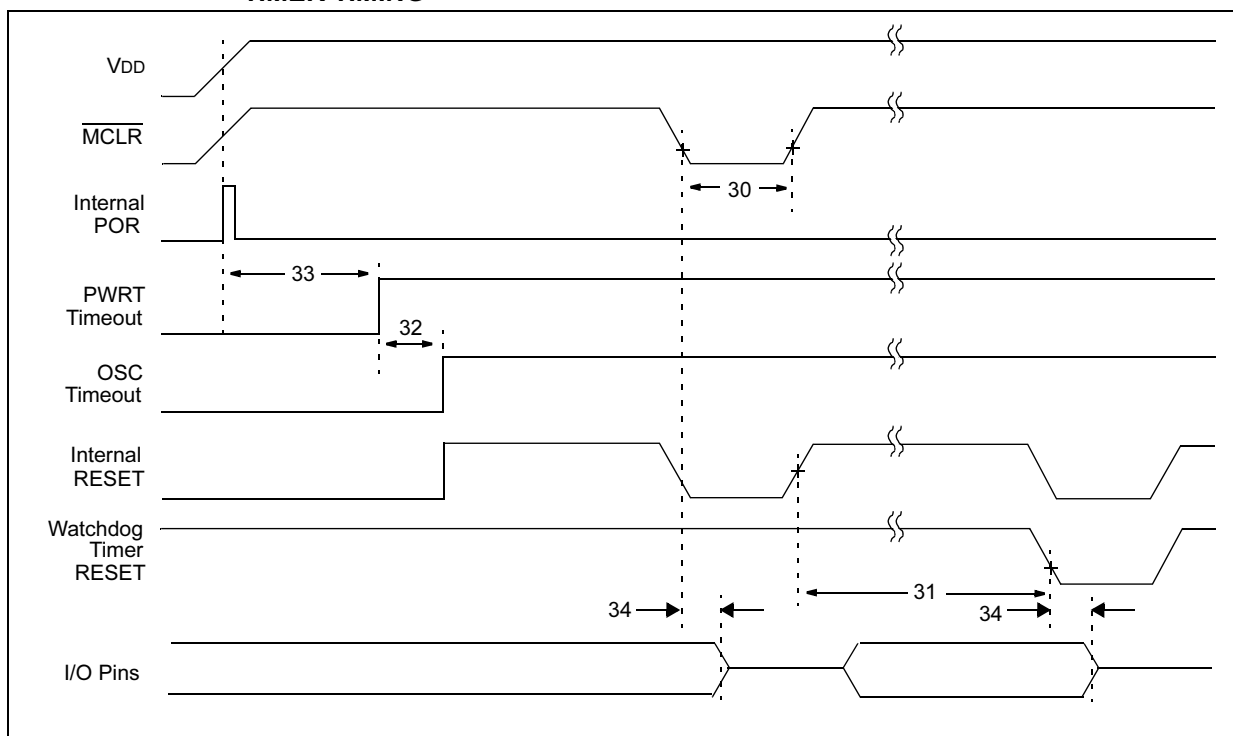
Cycles: 1

Example DECF CNT, 1

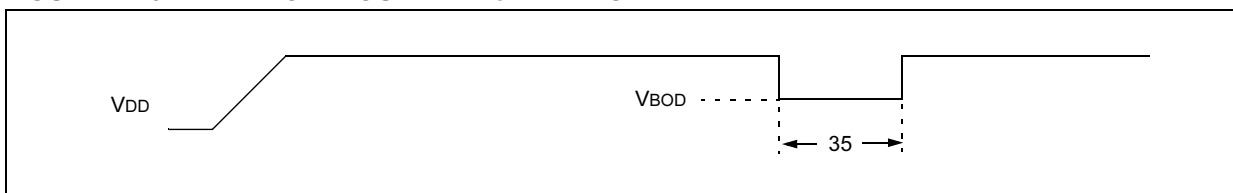
Before Instruction  
CNT = 0x01  
Z = 0  
After Instruction  
CNT = 0x00  
Z = 1

# PIC16F62X

**FIGURE 17-8: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING**



**FIGURE 17-9: BROWN-OUT DETECT TIMING**



**TABLE 17-6: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER REQUIREMENTS**

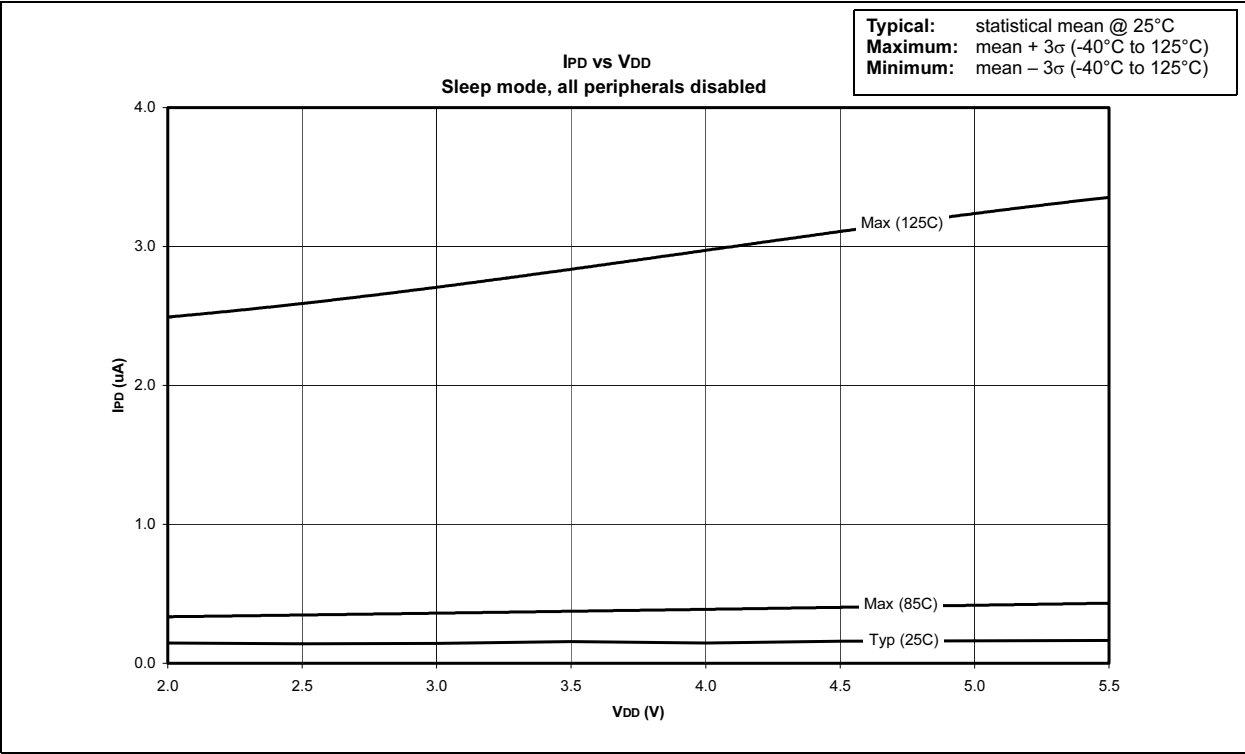
Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30	TmCL	MCLR Pulse Width (low)	2000 TBD	— TBD	— TBD	ns ms	VDD = 5V, -40°C to +85°C Extended temperature
31	Twdt	Watchdog Timer Timeout Period (No Prescaler)	7 TBD	18 TBD	33 TBD	ms ms	VDD = 5V, -40°C to +85°C Extended temperature
32	Tost	Oscillation Start-up Timer Period	—	1024Tosc	—	—	Tosc = OSC1 period
33*	Tpwrt	Power-up Timer Period	28 TBD	72 TBD	132 TBD	ms ms	VDD = 5V, -40°C to +85°C Extended temperature
34	TIOZ	I/O Hi-impedance from MCLR Low or Watchdog Timer Reset	—	—	2.0	μs	
35	TBOD	Brown-out Detect pulse width	100	—	—	μs	VDD ≤ VBOD (D005)

\* These parameters are characterized but not tested.

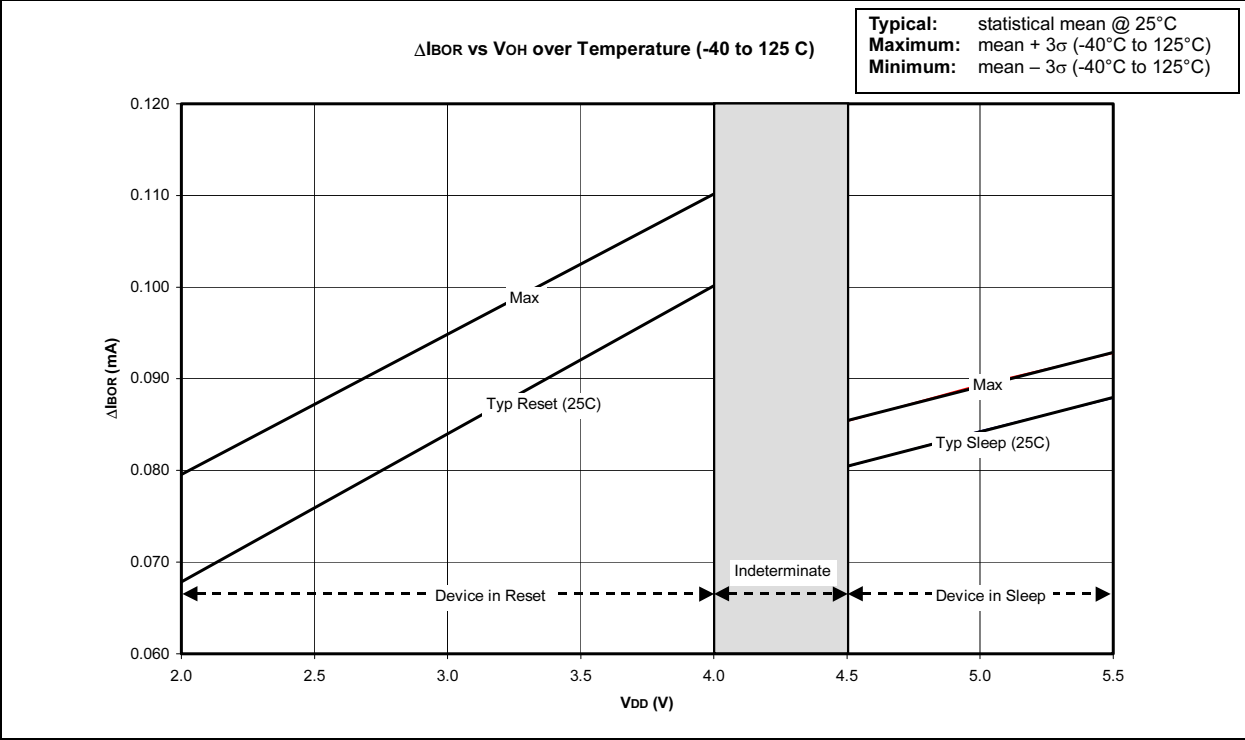
† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**Note:** The graphs and tables provided in this section are for design guidance and are not tested.

**FIGURE 18-10: I<sub>PD</sub> vs V<sub>DD</sub> SLEEP MODE, ALL PERIPHERALS DISABLED**

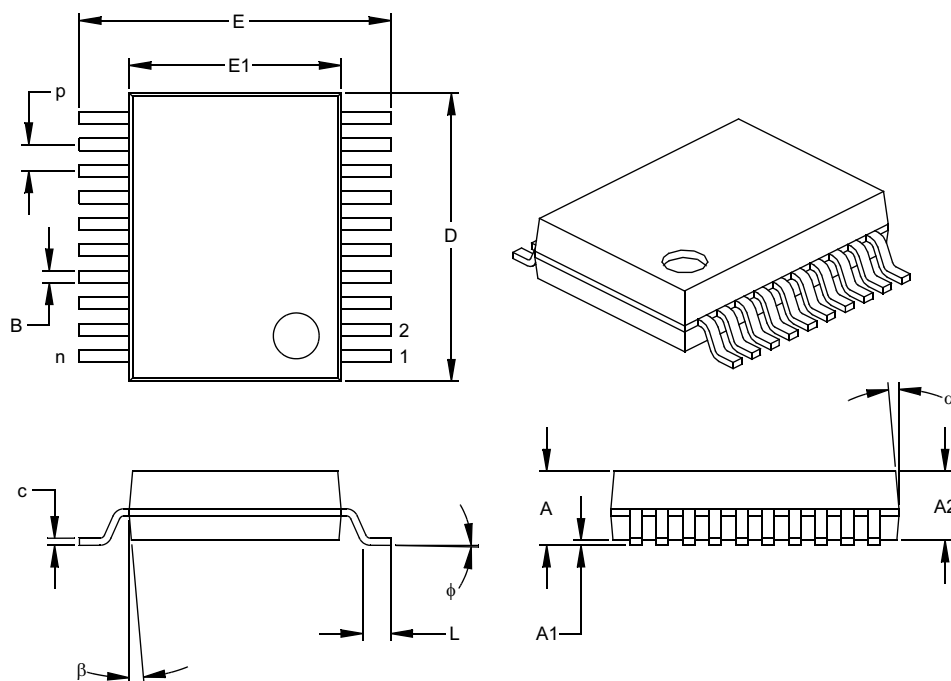


**FIGURE 18-11: ΔI<sub>BOD</sub> vs V<sub>OH</sub> OVER TEMPERATURE (-40 to 125°C)**



# PIC16F62X

## K04-072 20-Lead Plastic Shrink Small Outline (SS) – 5.30 mm



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		20			20	
Pitch	p		.026			0.65	
Overall Height	A	.068	.073	.078	1.73	1.85	1.98
Molded Package Thickness	A2	.064	.068	.072	1.63	1.73	1.83
Standoff §	A1	.002	.006	.010	0.05	0.15	0.25
Overall Width	E	.299	.309	.322	7.59	7.85	8.18
Molded Package Width	E1	.201	.207	.212	5.11	5.25	5.38
Overall Length	D	.278	.284	.289	7.06	7.20	7.34
Foot Length	L	.022	.030	.037	0.56	0.75	0.94
Lead Thickness	c	.004	.007	.010	0.10	0.18	0.25
Foot Angle	φ	0	4	8	0.00	101.60	203.20
Lead Width	B	.010	.013	.015	0.25	0.32	0.38
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-150

Drawing No. C04-072