

Welcome to [E-XFL.COM](https://www.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	4MHz
Connectivity	-
Peripherals	POR, WDT
Number of I/O	3
Program Memory Size	768B (512 x 12)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	24 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	8-VDFN Exposed Pad
Supplier Device Package	8-DFN (2x3)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic10f202t-i-mc

PIC10F200/202/204/206

Pin Diagrams

FIGURE 1: 6-PIN SOT-23

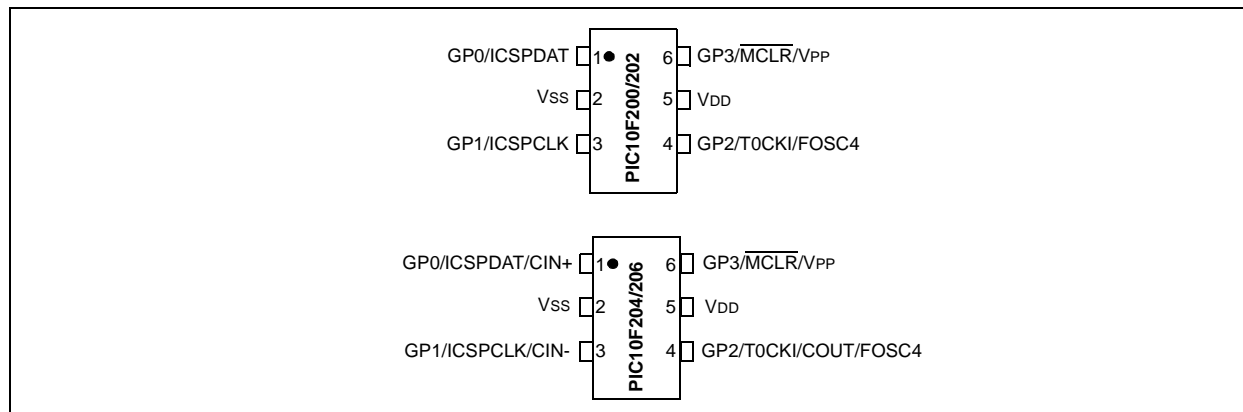


FIGURE 2: 8-PIN PDIP

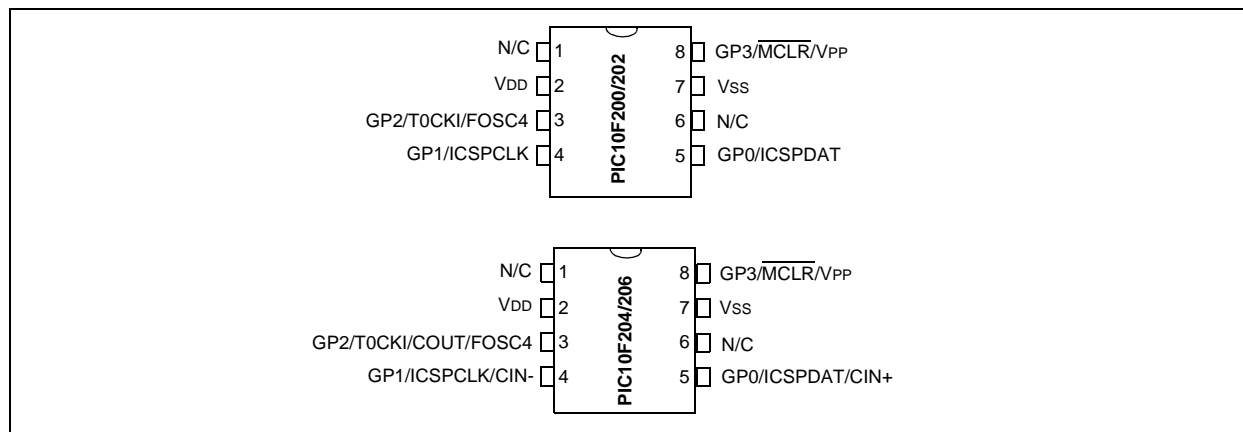


FIGURE 3: 8-PIN DFN

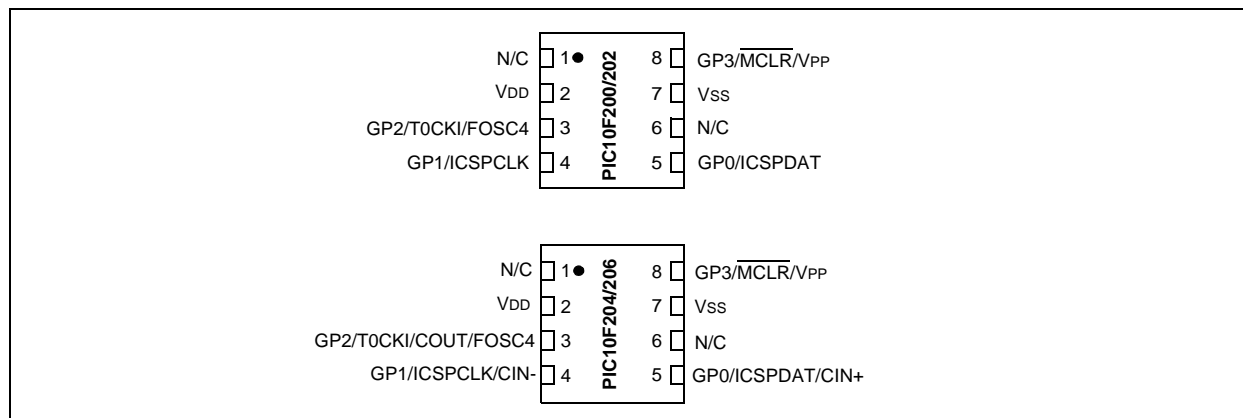


Table of Contents

1.0	General Description.....	4
2.0	PIC10F200/202/204/206 Device Varieties	5
3.0	Architectural Overview	6
4.0	Memory Organization	11
5.0	I/O Port.....	20
6.0	Timer0 Module and TMR0 Register (PIC10F200/202).....	23
7.0	Timer0 Module and TMR0 Register (PIC10F204/206).....	27
8.0	Comparator Module.....	31
9.0	Special Features of the CPU.....	35
10.0	Instruction Set Summary	45
11.0	Development Support.....	53
12.0	Electrical Characteristics	57
13.0	DC and AC Characteristics Graphs and Tables.....	67
14.0	Packaging Information.....	75
	The Microchip Web Site.....	85
	Customer Change Notification Service	85
	Customer Support.....	85
	Product Identification System	86

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at docerrors@microchip.com. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

<http://www.microchip.com>

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000000A is version A of document DS30000000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; <http://www.microchip.com>
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our web site at www.microchip.com to receive the most current information on all of our products.

PIC10F200/202/204/206

3.0 ARCHITECTURAL OVERVIEW

The high performance of the PIC10F200/202/204/206 devices can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC10F200/202/204/206 devices use a Harvard architecture in which program and data are accessed on separate buses. This improves bandwidth over traditional von Neumann architectures where program and data are fetched on the same bus. Separating program and data memory further allows instructions to be sized differently than the 8-bit wide data word. Instruction opcodes are 12 bits wide, making it possible to have all single-word instructions. A 12-bit wide program memory access bus fetches a 12-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions. Consequently, all instructions (33) execute in a single cycle (1 μ s @ 4 MHz) except for program branches.

The table below lists program memory (Flash) and data memory (RAM) for the PIC10F200/202/204/206 devices.

TABLE 3-1: PIC10F2XX MEMORY

Device	Memory	
	Program	Data
PIC10F200	256 x 12	16 x 8
PIC10F202	512 x 12	24 x 8
PIC10F204	256 x 12	16 x 8
PIC10F206	512 x 12	24 x 8

The PIC10F200/202/204/206 devices can directly or indirectly address its register files and data memory. All Special Function Registers (SFR), including the PC, are mapped in the data memory. The PIC10F200/202/204/206 devices have a highly orthogonal (symmetrical) instruction set that makes it possible to carry out any operation, on any register, using any addressing mode. This symmetrical nature and lack of “special optimal situations” make programming with the PIC10F200/202/204/206 devices simple, yet efficient. In addition, the learning curve is reduced significantly.

The PIC10F200/202/204/206 devices contain an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between data in the working register and any register file.

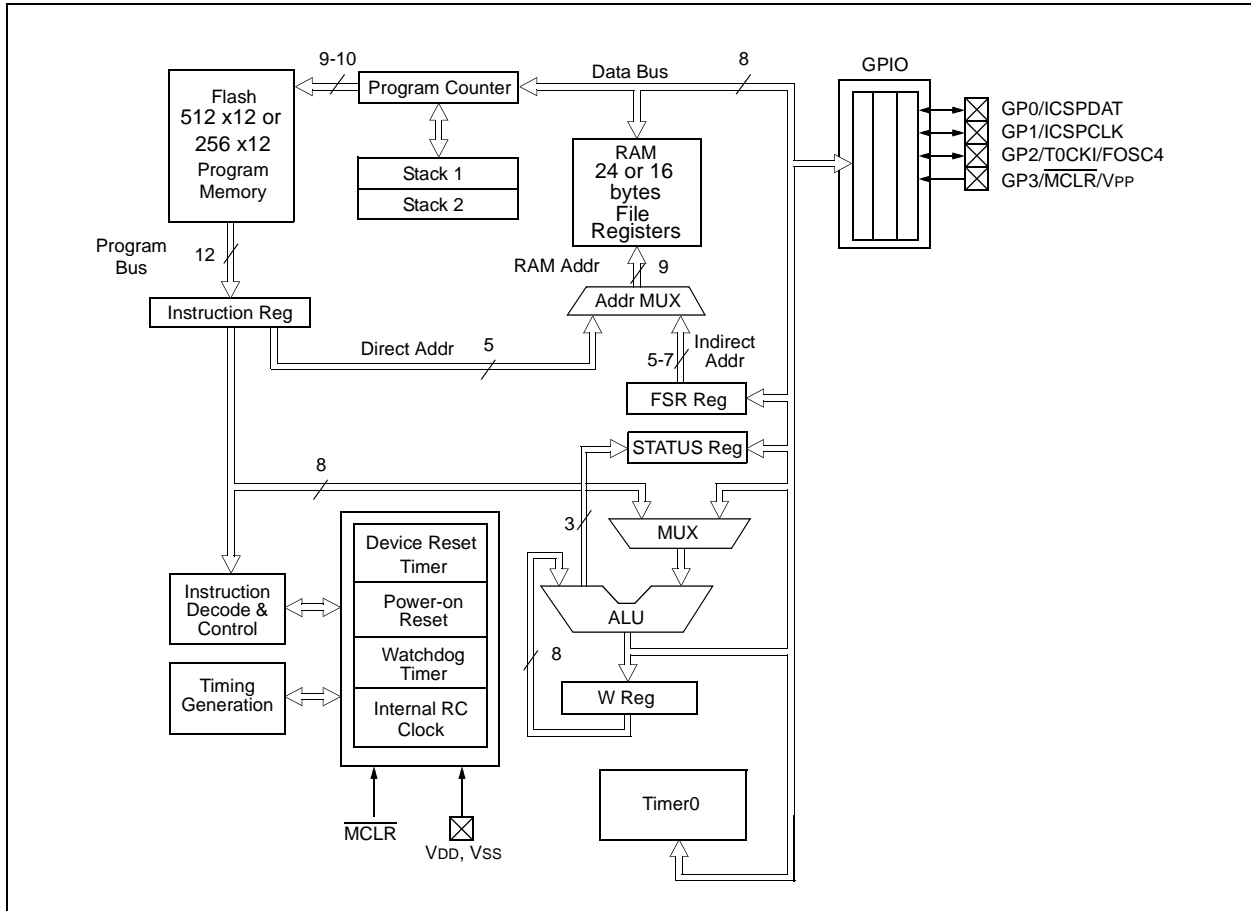
The ALU is 8 bits wide and capable of addition, subtraction, shift and logical operations. Unless otherwise mentioned, arithmetic operations are two's complement in nature. In two-operand instructions, one operand is typically the W (working) register. The other operand is either a file register or an immediate constant. In single operand instructions, the operand is either the W register or a file register.

The W register is an 8-bit working register used for ALU operations. It is not an addressable register.

Depending on the instruction executed, the ALU may affect the values of the Carry (C), Digit Carry (DC) and Zero (Z) bits in the STATUS register. The C and DC bits operate as a borrow and digit borrow out bit, respectively, in subtraction. See the SUBWF and ADDWF instructions for examples.

A simplified block diagram is shown in Figure 3-1 and Figure 3-2, with the corresponding device pins described in Table 3-2.

FIGURE 3-1: PIC10F200/202 BLOCK DIAGRAM



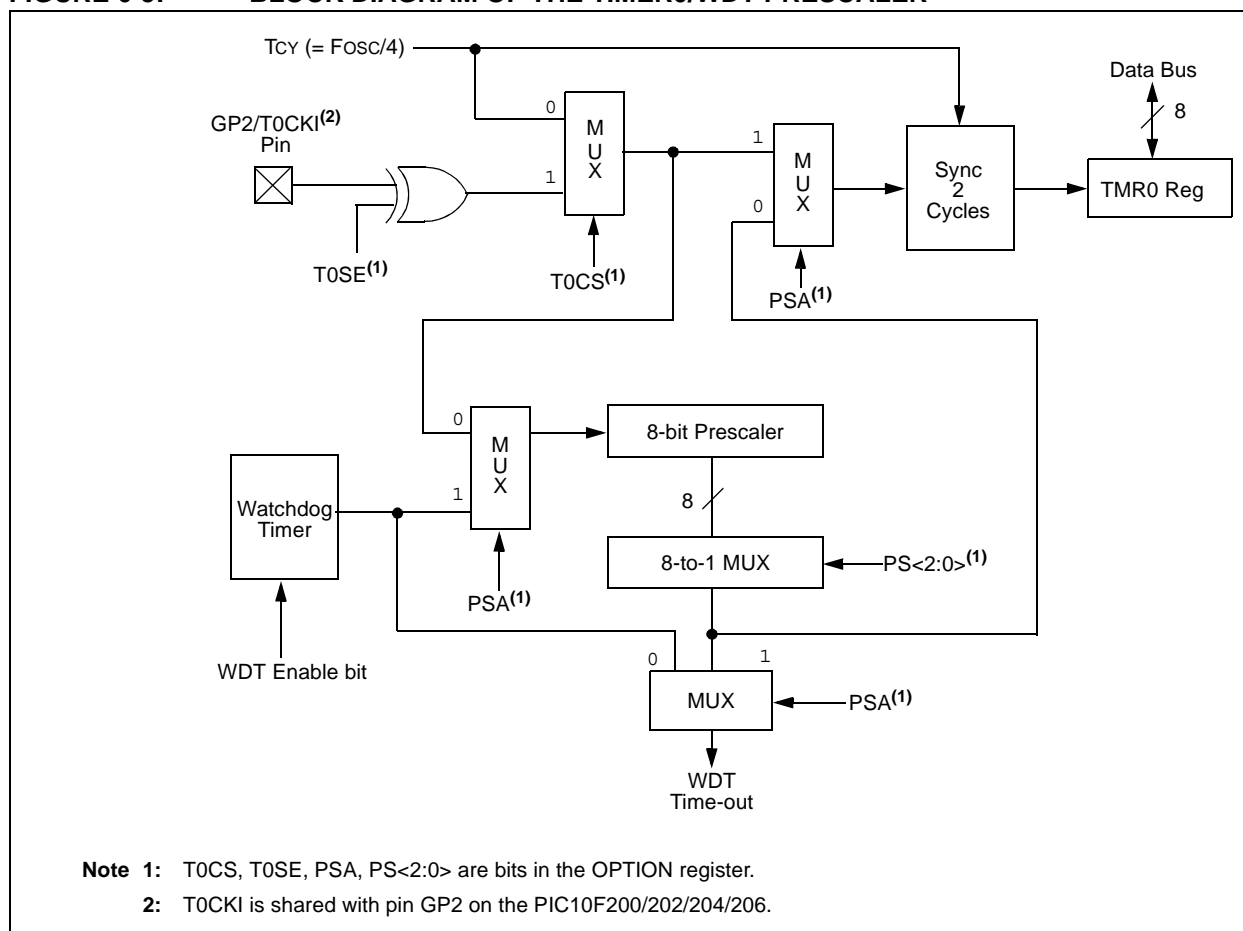
PIC10F200/202/204/206

To change the prescaler from the WDT to the Timer0 module, use the sequence shown in Example 6-2. This sequence must be used even if the WDT is disabled. A CLRWDT instruction should be executed before switching the prescaler.

EXAMPLE 6-2: CHANGING PRESCALER (WDT→TIMER0)

```
CLRWDT      ;Clear WDT and
             ;prescaler
MOVLW  'xxxx0xxx' ;Select TMR0, new
             ;prescale value and
             ;clock source
OPTION
```

FIGURE 6-5: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



7.0 TIMER0 MODULE AND TMR0 REGISTER (PIC10F204/206)

The Timer0 module has the following features:

- 8-bit timer/counter register, TMR0
- Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select:
 - Edge select for external clock
 - External clock from either the T0CKI pin or from the output of the comparator

Figure 7-1 is a simplified block diagram of the Timer0 module.

Timer mode is selected by clearing the T0CS bit (OPTION<5>). In Timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If TMR0 register is written, the increment is inhibited for the following two cycles (Figure 7-2 and Figure 7-3). The user can work around this by writing an adjusted value to the TMR0 register.

There are two types of Counter mode. The first Counter mode uses the T0CKI pin to increment Timer0. It is selected by setting the T0CS bit (OPTION<5>), setting the CMPT0CS bit (CMCON0<4>) and setting the COUTEN bit (CMCON0<6>). In this mode, Timer0 will increment either on every rising or falling edge of pin T0CKI. The T0SE bit (OPTION<4>) determines the source edge. Clearing the T0SE bit selects the rising edge. Restrictions on the external clock input are discussed in detail in **Section 7.1 “Using Timer0 with an External Clock (PIC10F204/206)”**.

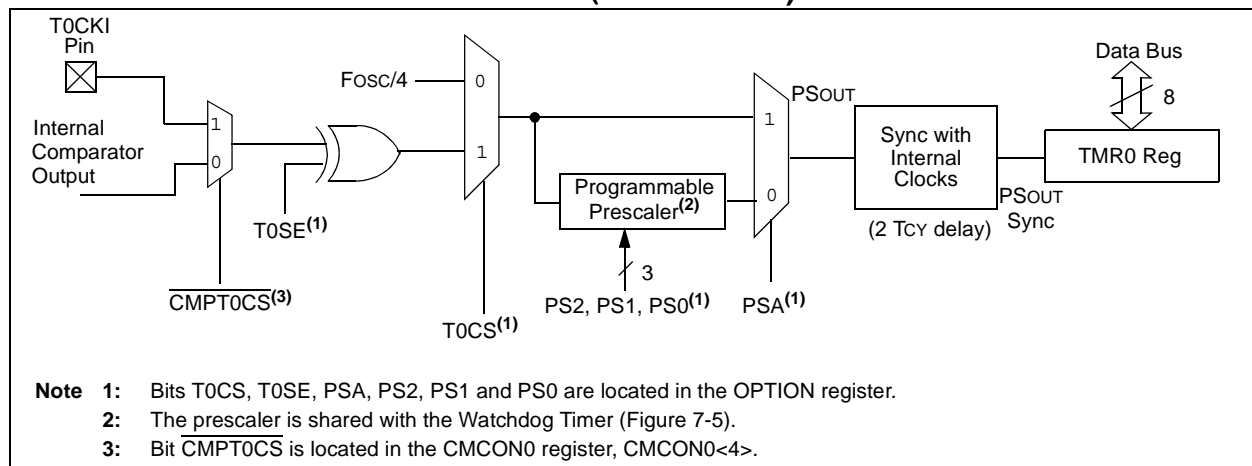
The second Counter mode uses the output of the comparator to increment Timer0. It can be entered in two different ways. The first way is selected by setting the T0CS bit (OPTION<5>) and clearing the CMPT0CS bit (CMCON0<4>); (COUTEN [CMCON0<6>]) does not affect this mode of operation. This enables an internal connection between the comparator and the Timer0.

The second way is selected by setting the T0CS bit (OPTION<5>), setting the CMPT0CS bit (CMCON0<4>) and clearing the COUTEN bit (CMCON0<6>). This allows the output of the comparator onto the T0CKI pin, while keeping the T0CKI input active. Therefore, any comparator change on the COUT pin is fed back into the T0CKI input. The T0SE bit (OPTION<4>) determines the source edge. Clearing the T0SE bit selects the rising edge. Restrictions on the external clock input as discussed in **Section 7.1 “Using Timer0 with an External Clock (PIC10F204/206)”**

The prescaler may be used by either the Timer0 module or the Watchdog Timer, but not both. The prescaler assignment is controlled in software by the control bit, PSA (OPTION<3>). Clearing the PSA bit will assign the prescaler to Timer0. The prescaler is not readable or writable. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4,..., 1:256 are selectable. **Section 7.2 “Prescaler”** details the operation of the prescaler.

A summary of registers associated with the Timer0 module is found in Table 7-1.

FIGURE 7-1: TIMER0 BLOCK DIAGRAM (PIC10F204/206)

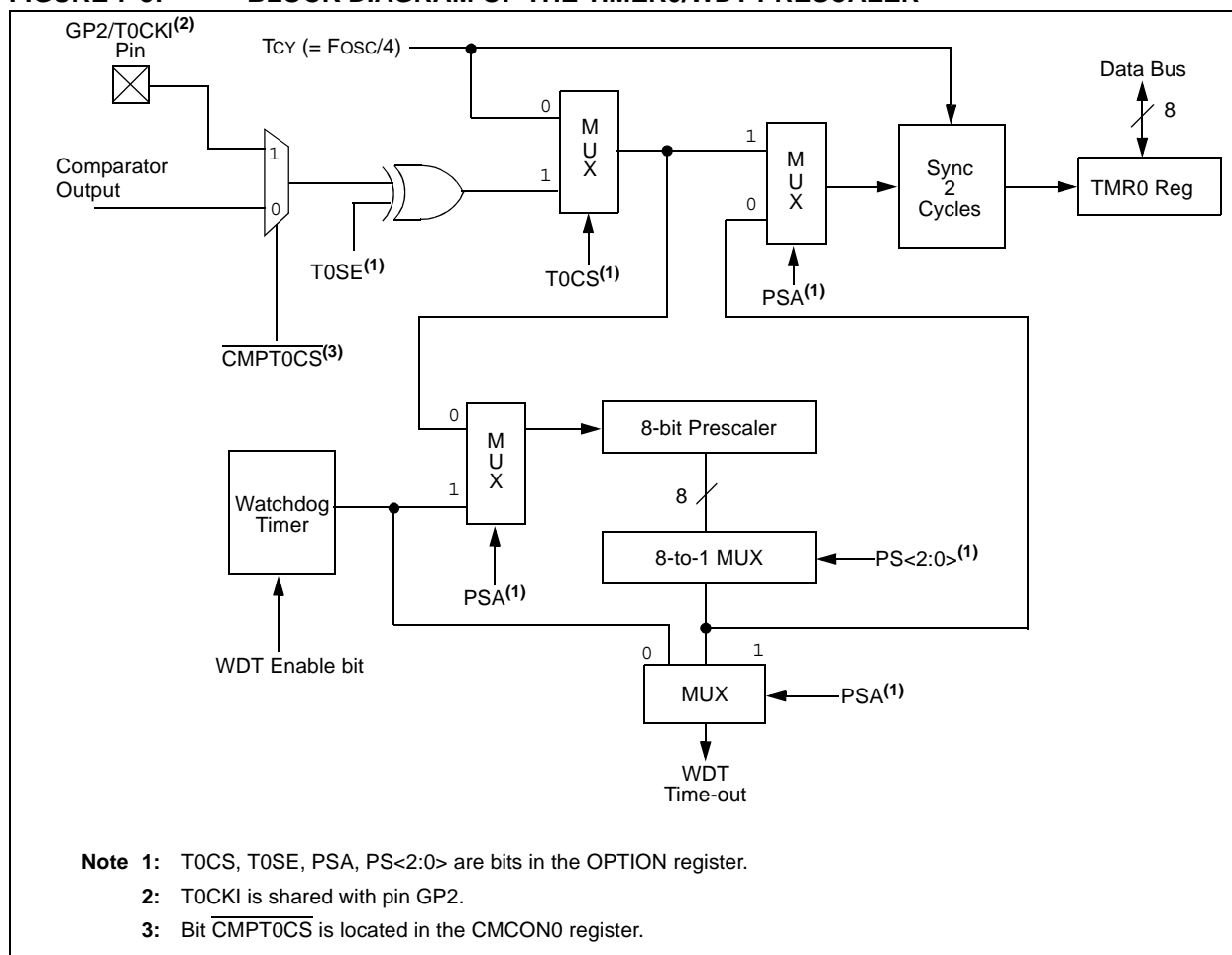


PIC10F200/202/204/206

EXAMPLE 7-2: CHANGING PRESCALER (WDT→TIMER0)

```
CLRWDT      ;Clear WDT and
             ;prescaler
MOVLW  'xxx0xxx' ;Select TMR0, new
             ;prescale value and
             ;clock source
OPTION
```

FIGURE 7-5: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



8.1 Comparator Configuration

The on-board comparator inputs, (GP0/CIN+, GP1/CIN-), as well as the comparator output (GP2/COU^T), are steerable. The CMCON0, OPTION and TRIS registers are used to steer these pins (see Figure 8-1). If the Comparator mode is changed, the comparator output level may not be valid for the specified mode change delay shown in Table 12-1.

Note: The comparator can have an inverted output (see Figure 8-1).

FIGURE 8-1: BLOCK DIAGRAM OF THE COMPARATOR

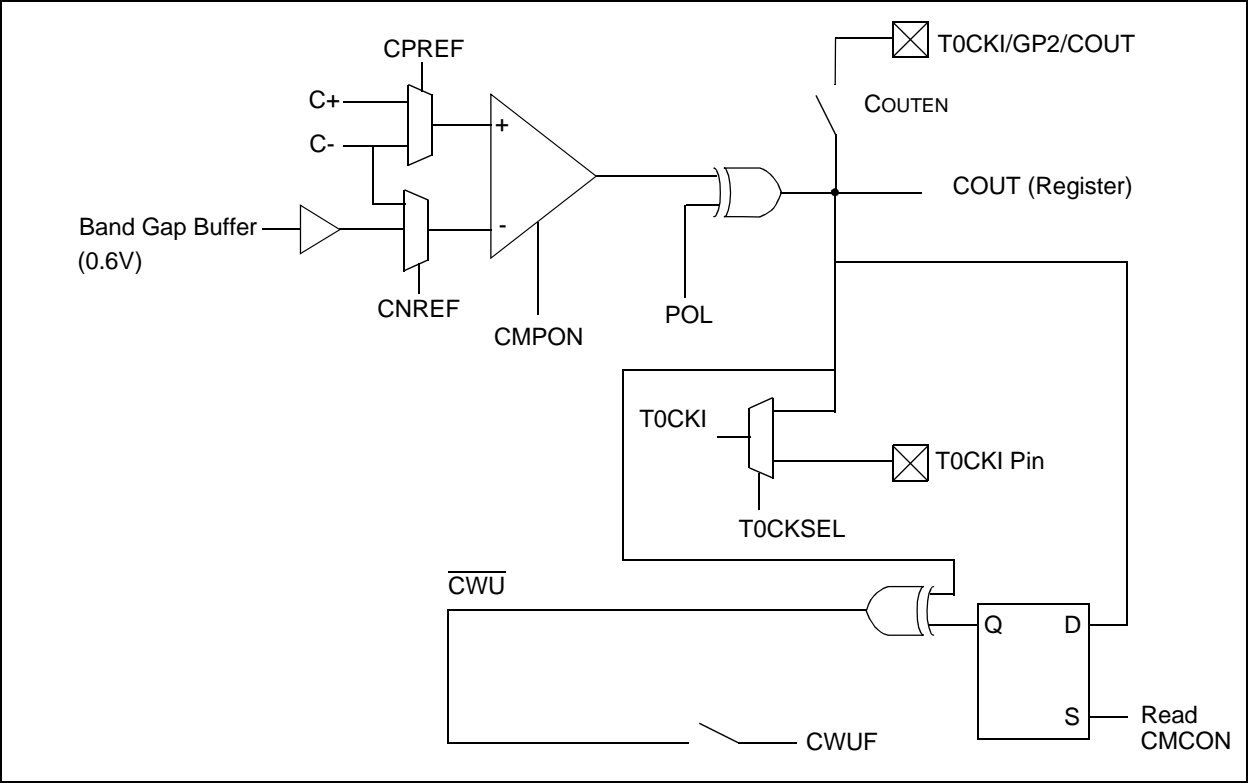


TABLE 8-1: TMR0 CLOCK SOURCE
FUNCTION MUXING

T0CS	CMPT0CS	COUTEN	Source
0	x	x	Internal Instruction Cycle
1	0	0	CMPOUT
1	0	1	CMPOUT
1	1	0	CMPOUT
1	1	1	T0CKI

PIC10F200/202/204/206

FIGURE 8-3: ANALOG INPUT MODE

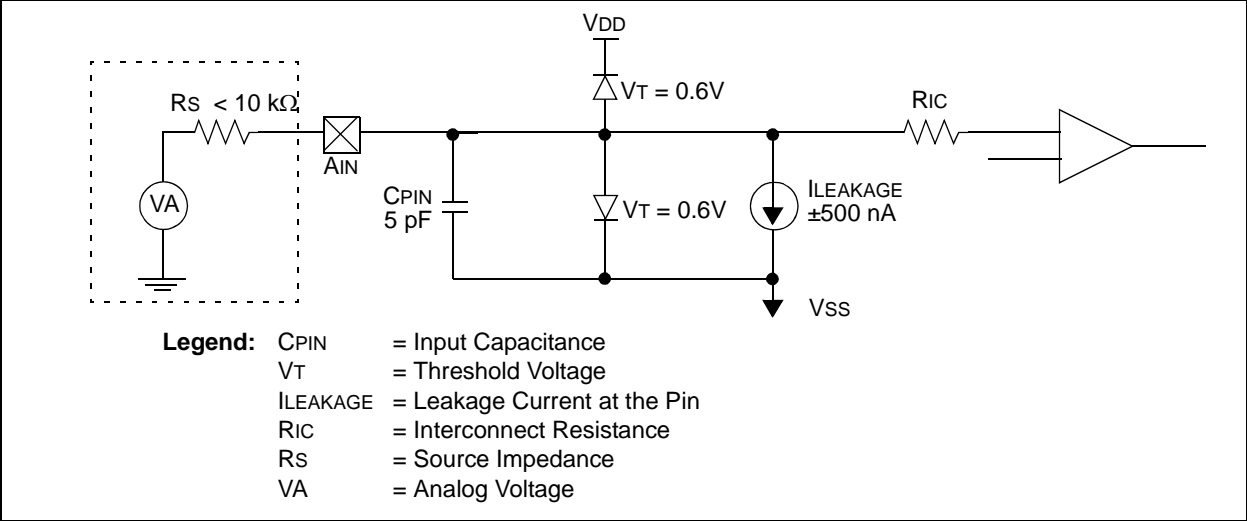


TABLE 8-2: REGISTERS ASSOCIATED WITH COMPARATOR MODULE

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
03h	STATUS	GPWUF	CWUF	—	\overline{TO}	\overline{PD}	Z	DC	C	00-1 1xxx	qq0q quuu
07h	CMCON0	CMPOUT	\overline{COUTEN}	POL	$\overline{CMPT0CS}$	CMPON	CNREF	CPREF	\overline{CWU}	1111 1111	uuuu uuuu
N/A	TRISGPIO	—	—	—	—	I/O Control Register				---- 1111	---- 1111

Legend: x = Unknown, u = Unchanged, — = Unimplemented, read as '0', q = Depends on condition.

PIC10F200/202/204/206

9.2 Oscillator Configurations

9.2.1 OSCILLATOR TYPES

The PIC10F200/202/204/206 devices are offered with Internal Oscillator mode only.

- INTOSC: Internal 4 MHz Oscillator

9.2.2 INTERNAL 4 MHz OSCILLATOR

The internal oscillator provides a 4 MHz (nominal) system clock (see **Section 12.0 “Electrical Characteristics”** for information on variation over voltage and temperature).

In addition, a calibration instruction is programmed into the last address of memory, which contains the calibration value for the internal oscillator. This location is always uncode protected, regardless of the code-protect settings. This value is programmed as a `MOVLW xx` instruction where `xx` is the calibration value and is placed at the Reset vector. This will load the `W` register with the calibration value upon Reset and the PC will then roll over to the users program at address `0x000`. The user then has the option of writing the value to the `OSCCAL` Register (`05h`) or ignoring it.

`OSCCAL`, when written to with the calibration value, will “trim” the internal oscillator to remove process variation from the oscillator frequency.

Note: Erasing the device will also erase the pre-programmed internal calibration value for the internal oscillator. The calibration value must be read prior to erasing the part so it can be reprogrammed correctly later.

9.3 Reset

The device differentiates between various kinds of Reset:

- Power-on Reset (POR)
- $\overline{\text{MCLR}}$ Reset during normal operation
- $\overline{\text{MCLR}}$ Reset during Sleep
- WDT time-out Reset during normal operation
- WDT time-out Reset during Sleep
- Wake-up from Sleep on pin change
- Wake-up from Sleep on comparator change

Some registers are not reset in any way, they are unknown on POR and unchanged in any other Reset. Most other registers are reset to “Reset state” on Power-on Reset (POR), $\overline{\text{MCLR}}$, WDT or Wake-up on pin change Reset during normal operation. They are not affected by a WDT Reset during Sleep or $\overline{\text{MCLR}}$ Reset during Sleep, since these Resets are viewed as resumption of normal operation. The exceptions to this are $\overline{\text{TO}}$, $\overline{\text{PD}}$, `GPWUF` and `CWUF` bits. They are set or cleared differently in different Reset situations. These bits are used in software to determine the nature of Reset. See Table 9-1 for a full description of Reset states of all registers.

TABLE 9-1: RESET CONDITIONS FOR REGISTERS – PIC10F200/202/204/206

Register	Address	Power-on Reset	$\overline{\text{MCLR}}$ Reset, WDT Time-out, Wake-up On Pin Change, Wake on Comparator Change
W	—	q q q q q q q u ⁽¹⁾	q q q q q q q u ⁽¹⁾
INDF	00h	x x x x x x x x	u u u u u u u u
TMR0	01h	x x x x x x x x	u u u u u u u u
PCL	02h	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1
STATUS	03h	0 0 - 1 1 x x x	q 0 0 q q u u u ⁽²⁾
STATUS ⁽³⁾	03h	0 0 - 1 1 x x x	q q 0 q q u u u ⁽²⁾
FSR	04h	1 1 1 x x x x x	1 1 1 u u u u u
OSCCAL	05h	1 1 1 1 1 1 1 0	u u u u u u u u
GPIO	06h	- - - - x x x x	- - - - u u u u
CMCON ⁽³⁾	07h	1 1 1 1 1 1 1 1	u u u u u u u u
OPTION	—	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1
TRISGPIO	—	- - - - 1 1 1 1	- - - - 1 1 1 1

Legend: u = unchanged, x = unknown, – = unimplemented bit, read as ‘0’, q = value depends on condition.

Note 1: Bits <7:2> of W register contain oscillator calibration values due to `MOVLW xx` instruction at top of memory.

2: See Table 9-2 for Reset value for specific conditions.

3: PIC10F204/206 only.

PIC10F200/202/204/206

FIGURE 9-2: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT

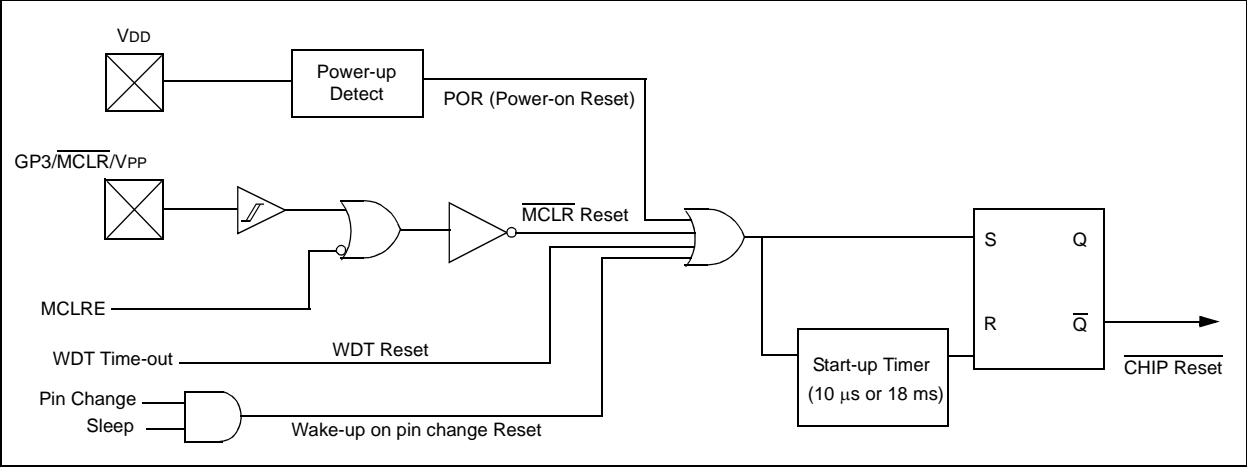


FIGURE 9-3: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ PULLED LOW)

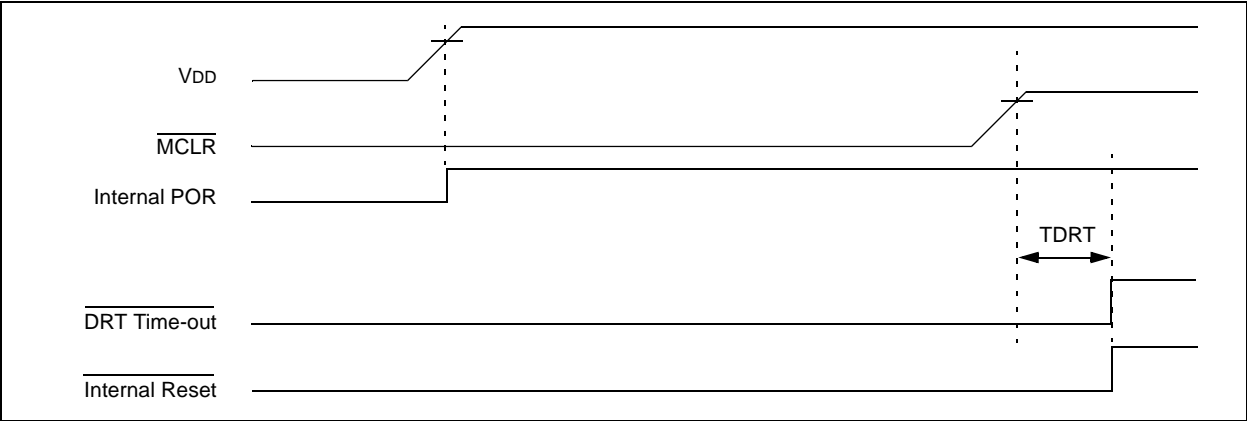


FIGURE 9-4: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ TIED TO VDD): FAST VDD RISE TIME

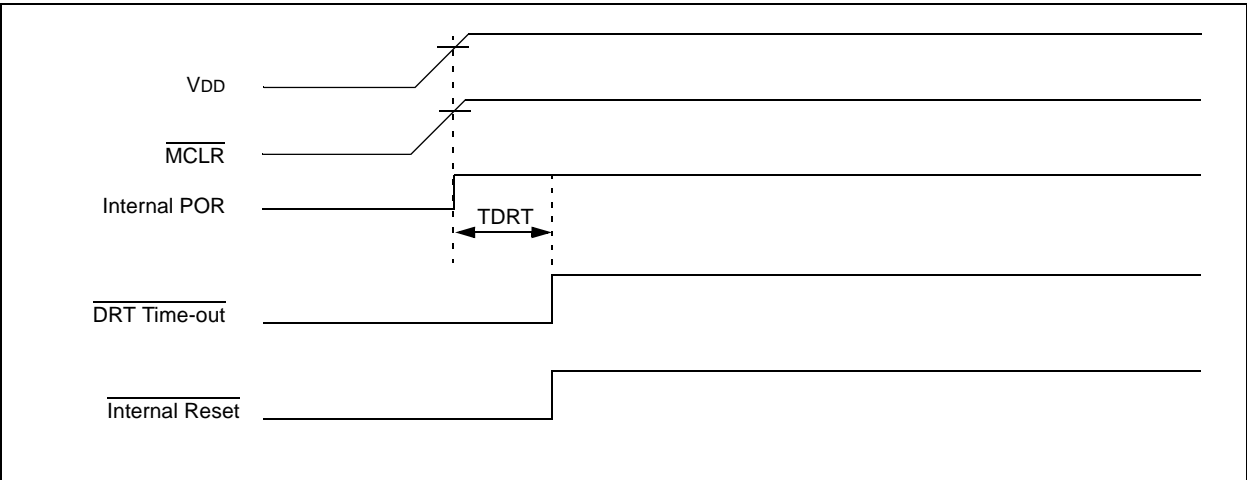
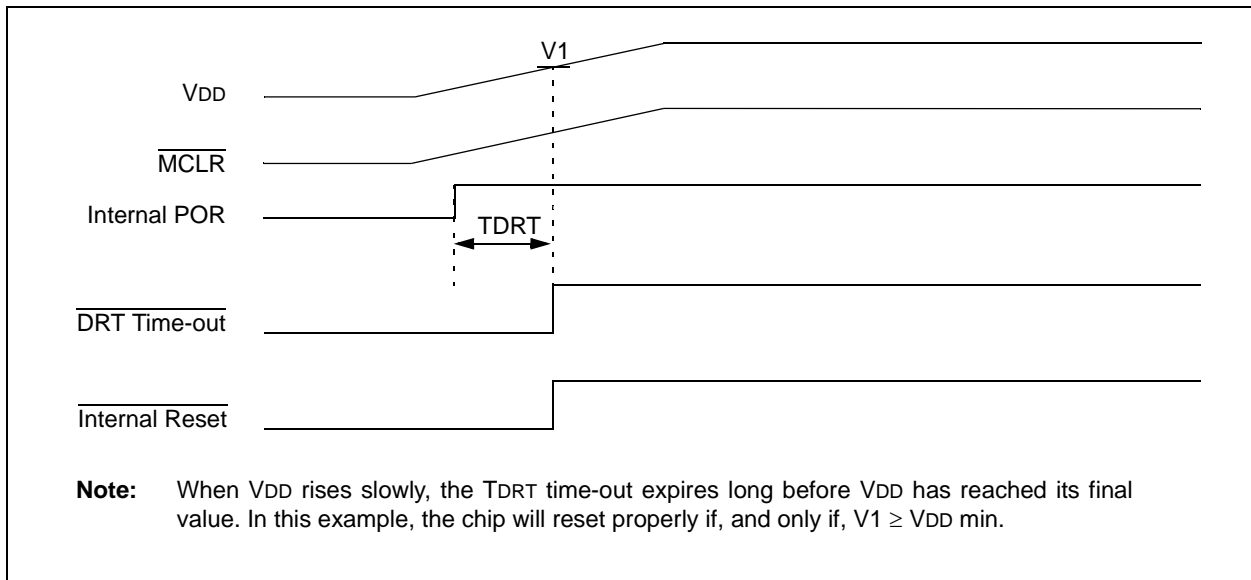


FIGURE 9-5: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ TIED TO V_{DD}): SLOW V_{DD} RISE TIME



10.0 INSTRUCTION SET SUMMARY

The PIC16 instruction set is highly orthogonal and is comprised of three basic categories.

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

Each PIC16 instruction is a 12-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 formats for each of the categories is presented in Figure 10-1, while the various opcode fields are summarized in Table 10-1.

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 '0', the result is placed in the W register. If 'd' is '1', 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 8 or 9-bit constant or literal value.

TABLE 10-1: 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
WDT	Watchdog Timer counter
$\overline{\text{TO}}$	Time-out bit
$\overline{\text{PD}}$	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
<i>italics</i>	User defined term (font is courier)

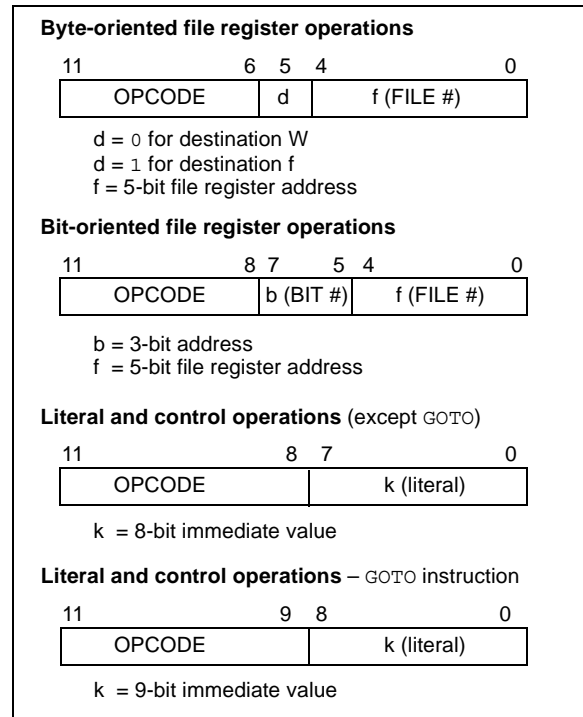
All instructions are executed within a single instruction cycle, unless a conditional test is true or the program counter is changed as a result of an instruction. In this case, the execution takes two instruction cycles. One instruction cycle consists of four oscillator periods. Thus, for an oscillator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

Figure 10-1 shows the three general formats that the instructions can have. All examples in the figure use the following format to represent a hexadecimal number:

0xhhh

where 'h' signifies a hexadecimal digit.

FIGURE 10-1: GENERAL FORMAT FOR INSTRUCTIONS



PIC10F200/202/204/206

BTFSS **Bit Test f, Skip if Set**

Syntax: [*label*] BTFSS f,b

Operands: $0 \leq f \leq 31$
 $0 \leq b < 7$

Operation: skip if (f) = 1

Status Affected: None

Description: If bit 'b' in register 'f' is '1', then the next instruction is skipped.
 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 2-cycle instruction.

CLRW **Clear W**

Syntax: [*label*] CLRW

Operands: None

Operation: 00h → (W);
 1 → Z

Status Affected: Z

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

CALL **Subroutine Call**

Syntax: [*label*] CALL k

Operands: $0 \leq k \leq 255$

Operation: (PC) + 1 → Top-of-Stack;
 k → PC<7:0>;
 (STATUS<6:5>) → PC<10:9>;
 0 → PC<8>

Status Affected: None

Description: Subroutine call. First, return address (PC + 1) is PUSHed onto the stack. The 8-bit immediate address is loaded into PC bits <7:0>. The upper bits PC<10:9> are loaded from STATUS<6:5>, PC<8> is cleared. CALL is a 2-cycle instruction.

CLRWDT **Clear Watchdog Timer**

Syntax: [*label*] CLRWDT

Operands: None

Operation: 00h → WDT;
 0 → WDT prescaler (if assigned);
 1 → \overline{TO} ;
 1 → PD

Status Affected: \overline{TO} , PD

Description: The CLRWDT instruction resets the WDT. It also resets the prescaler, if the prescaler is assigned to the WDT and not Timer0. Status bits \overline{TO} and PD are set.

CLRF **Clear f**

Syntax: [*label*] CLRF f

Operands: $0 \leq f \leq 31$

Operation: 00h → (f);
 1 → Z

Status Affected: Z

Description: The contents of register 'f' are cleared and the Z bit is set.

COMF **Complement f**

Syntax: [*label*] COMF f,d

Operands: $0 \leq f \leq 31$
 d ∈ [0,1]

Operation: (f) → (dest)

Status Affected: Z

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

11.2 MPLAB XC Compilers

The MPLAB XC Compilers are complete ANSI C compilers for all of Microchip's 8, 16, and 32-bit MCU and DSC devices. These compilers provide powerful integration capabilities, superior code optimization and ease of use. MPLAB XC Compilers run on Windows, Linux or MAC OS X.

For easy source level debugging, the compilers provide debug information that is optimized to the MPLAB X IDE.

The free MPLAB XC Compiler editions support all devices and commands, with no time or memory restrictions, and offer sufficient code optimization for most applications.

MPLAB XC Compilers include an assembler, linker and utilities. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. MPLAB XC Compiler uses the assembler to produce its object file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

11.3 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code, and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB X IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multipurpose source files
- Directives that allow complete control over the assembly process

11.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

11.5 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC DSC devices. MPLAB XC Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

12.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings^(†)

Ambient temperature under bias	-40°C to +125°C
Storage temperature	-65°C to +150°C
Voltage on VDD with respect to VSS	0 to +6.5V
Voltage on $\overline{\text{MCLR}}$ with respect to VSS.....	0 to +13.5V
Voltage on all other pins with respect to VSS	-0.3V to (VDD + 0.3V)
Total power dissipation ⁽¹⁾	800 mW
Max. current out of VSS pin	80 mA
Max. current into VDD pin	80 mA
Input clamp current, I _{IK} (V _I < 0 or V _I > VDD).....	±20 mA
Output clamp current, I _{OK} (V _O < 0 or V _O > VDD)	±20 mA
Max. output current sunk by any I/O pin	25 mA
Max. output current sourced by any I/O pin	25 mA
Max. output current sourced by I/O port	75 mA
Max. output current sunk by I/O port	75 mA

Note 1: Power dissipation is calculated as follows: $P_{DIS} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$

[†]NOTICE: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

PIC10F200/202/204/206

TABLE 12-3: CALIBRATED INTERNAL RC FREQUENCIES – PIC10F200/202/204/206

AC CHARACTERISTICS				Standard Operating Conditions (unless otherwise specified) Operating Temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (industrial), $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (extended) Operating Voltage V_{DD} range is described in Section 12.1 “DC Characteristics: PIC10F200/202/204/206 (Industrial)”				
Param. No.	Sym.	Characteristic	Freq. Tolerance	Min.	Typ.†	Max.	Units	Conditions
F10	FOSC	Internal Calibrated INTOSC Frequency ^(1,2)	$\pm 1\%$ $\pm 2\%$ $\pm 5\%$	3.96 3.92 3.80	4.00 4.00 4.00	4.04 4.08 4.20	MHz MHz MHz	$V_{DD}=3.5\text{V}$ @ 25°C $2.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ $0^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (industrial) $2.0\text{V} \leq V_{DD} \leq 5.5\text{V}$ $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (industrial) $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (extended)

* These parameters are characterized but not tested.

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

Note 1: To ensure these oscillator frequency tolerances, V_{DD} and V_{SS} must be capacitively decoupled as close to the device as possible. 0.1 μF and 0.01 μF values in parallel are recommended.

2: Under stable V_{DD} conditions.

FIGURE 12-3: RESET, WATCHDOG TIMER AND DEVICE RESET TIMER TIMING – PIC10F200/202/204/206

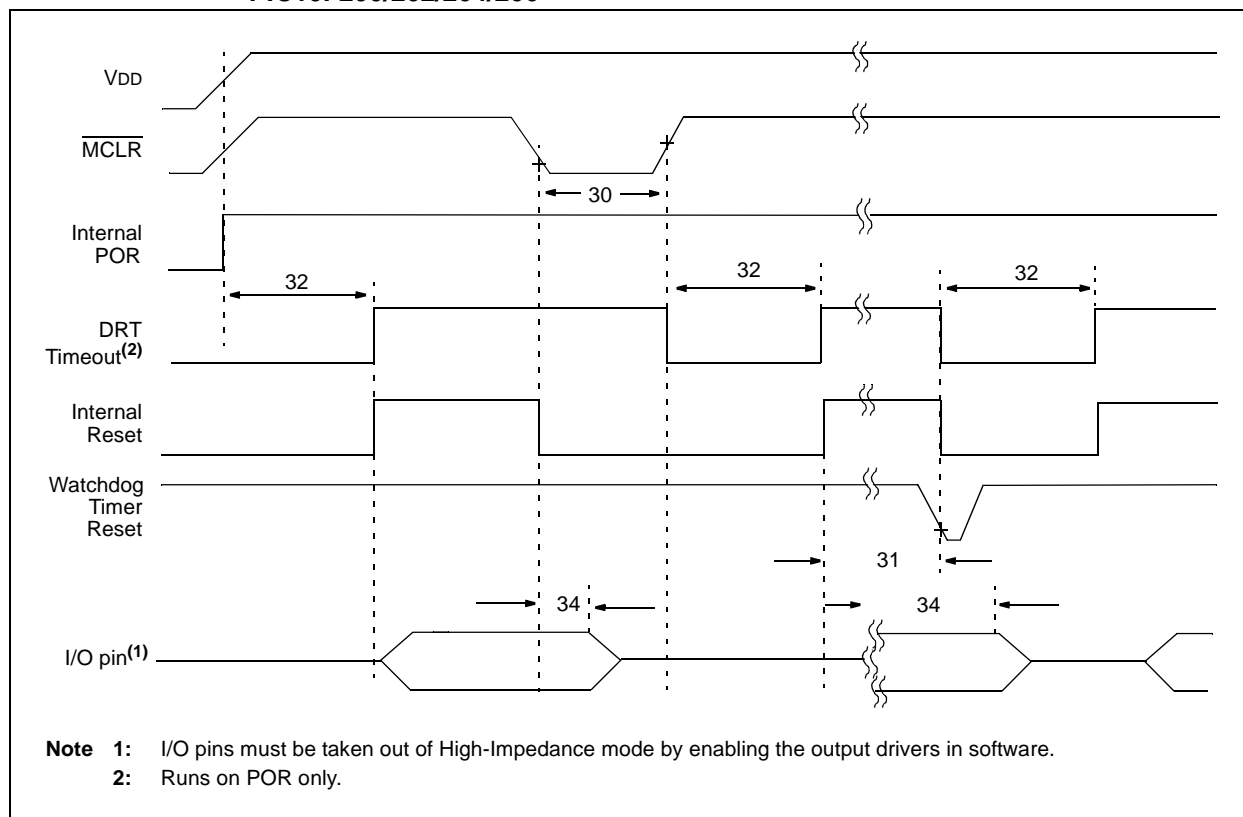


FIGURE 13-2: TYPICAL I_{PD} vs. V_{DD} (SLEEP MODE, ALL PERIPHERALS DISABLED)

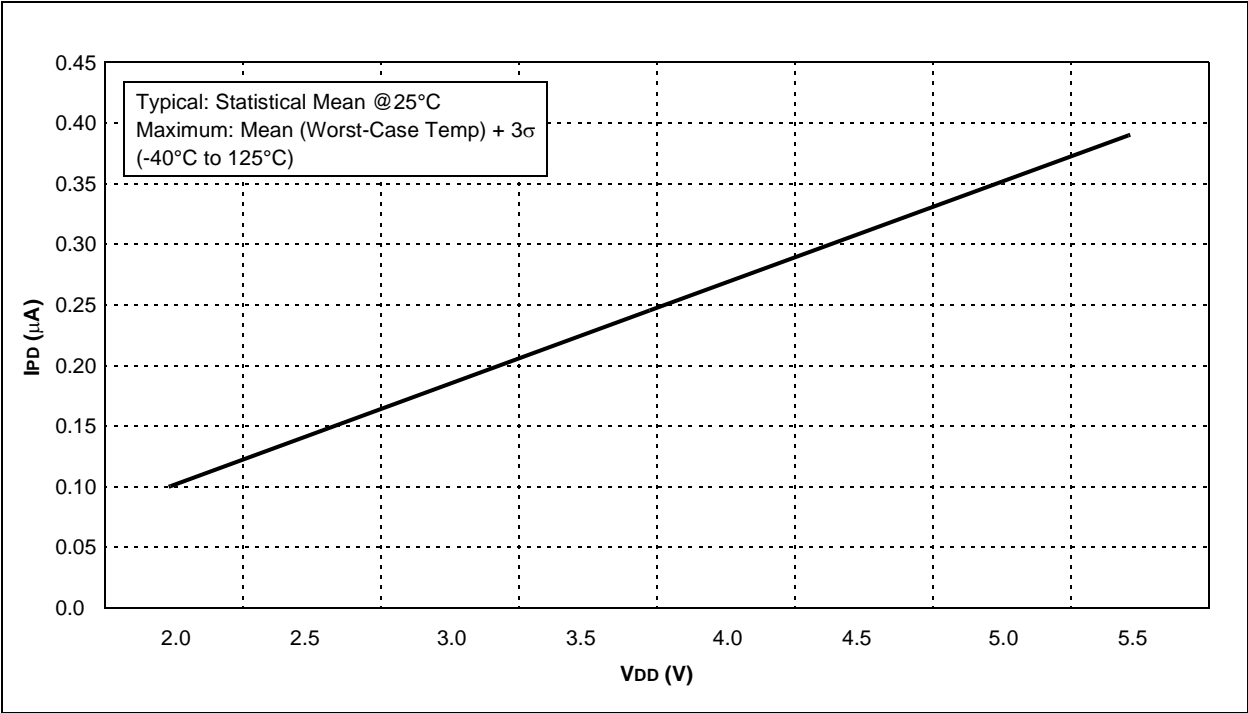


FIGURE 13-3: MAXIMUM I_{PD} vs. V_{DD} (SLEEP MODE, ALL PERIPHERALS DISABLED)

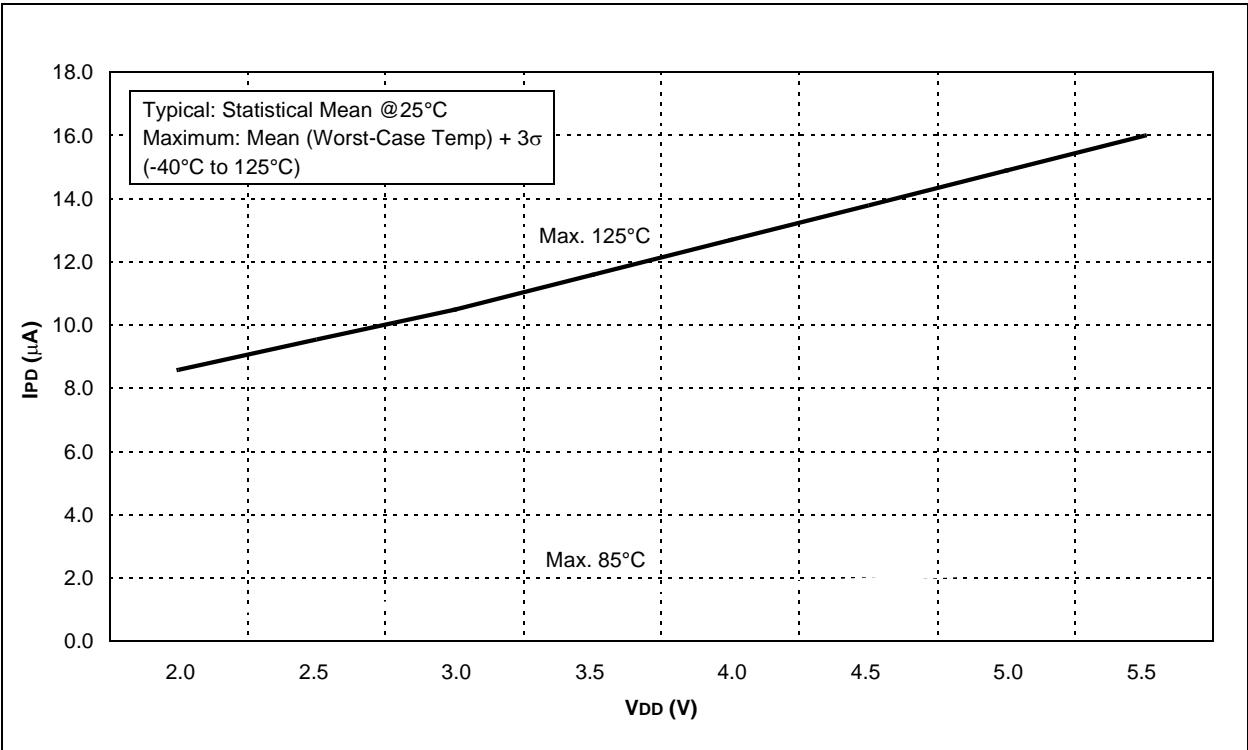


FIGURE 13-4: COMPARATOR I_{PD} vs. V_{DD} (COMPARATOR ENABLED)

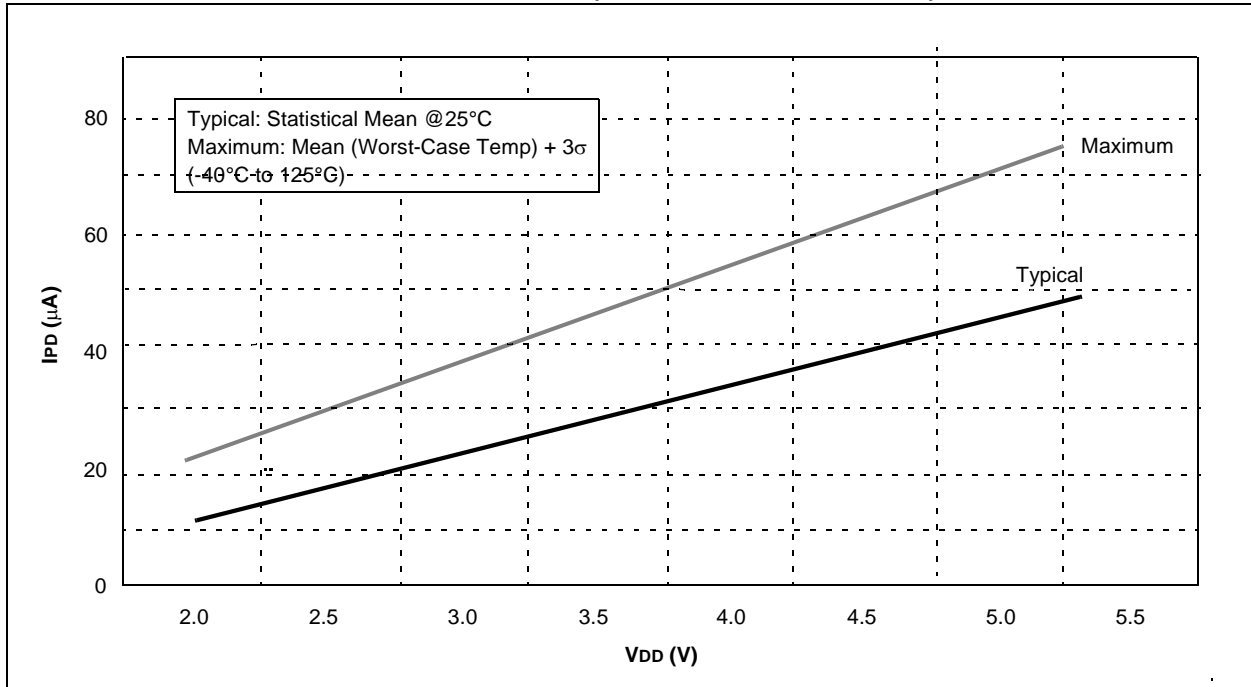
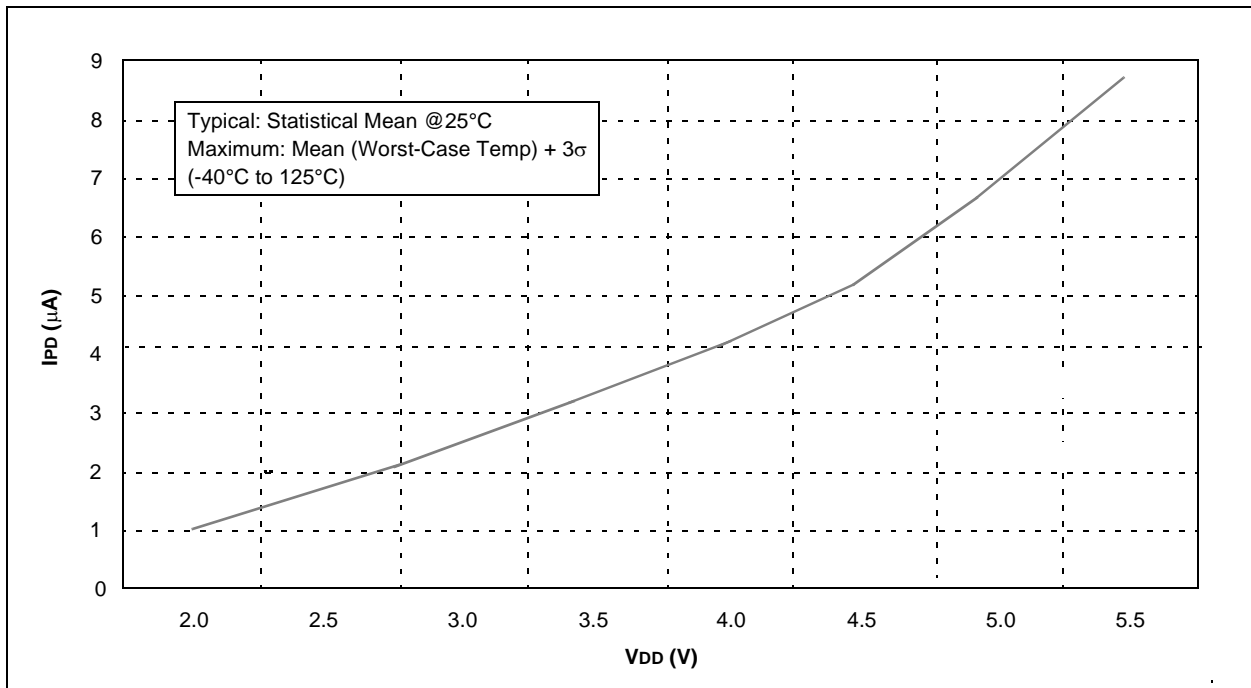


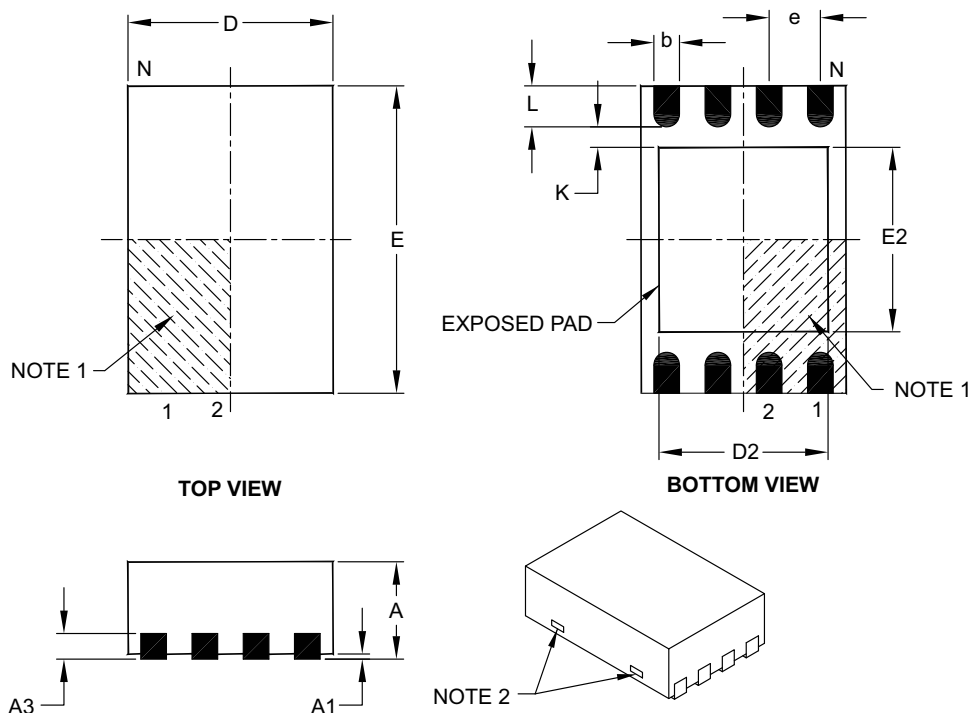
FIGURE 13-5: TYPICAL WDT I_{PD} vs. V_{DD}



PIC10F200/202/204/206

8-Lead Plastic Dual Flat, No Lead Package (MC) – 2x3x0.9 mm Body [DFN]

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



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Pins	N	8		
Pitch	e	0.50 BSC		
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3	0.20 REF		
Overall Length	D	2.00 BSC		
Overall Width	E	3.00 BSC		
Exposed Pad Length	D2	1.30	–	1.55
Exposed Pad Width	E2	1.50	–	1.75
Contact Width	b	0.20	0.25	0.30
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	–	–

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Package may have one or more exposed tie bars at ends.
- Package is saw singulated.
- Dimensioning and tolerancing per ASME Y14.5M.

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

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-123C