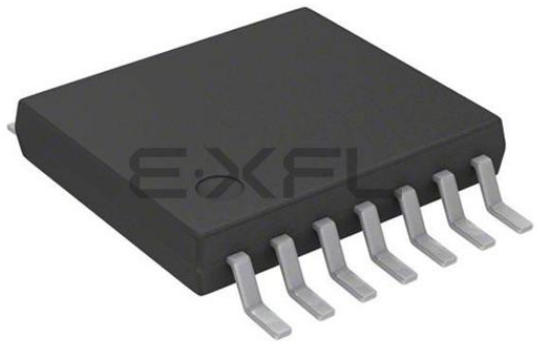


Welcome to E-XFL.COM

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	-
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	11
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	14-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	14-TSSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f616-e-st

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Trademarks

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

2.2 Data Memory Organization

The data memory (see Figure 2-4) is partitioned into two banks, which contain the General Purpose Registers (GPR) and the Special Function Registers (SFR). The Special Function Registers are located in the first 32 locations of each bank. PIC16F610/16HV610 Register locations 40h-7Fh in Bank 0 are General Purpose Registers, implemented as static RAM. PIC16F616/16HV616 Register locations 20h-7Fh in Bank 0 and A0h-BFh in Bank 1 are General Purpose Registers, implemented as static RAM. Register locations F0h-FFh in Bank 1 point to addresses 70h-7Fh in Bank 0. All other RAM is unimplemented and returns '0' when read. The RP0 bit of the STATUS register is the bank select bit.

RP0

0 → Bank 0 is selected

1 → Bank 1 is selected

Note:	The IRP and RP1 bits of the STATUS register are reserved and should always be maintained as '0's.

2.2.1 GENERAL PURPOSE REGISTER FILE

The register file is organized as 64 x 8 in the PIC16F610/16HV610 and 128 x 8 in the PIC16F616/16HV616. Each register is accessed, either directly or indirectly, through the File Select Register (FSR) (see **Section 2.4 "Indirect Addressing, INDF and FSR Registers"**).

2.2.2 SPECIAL FUNCTION REGISTERS

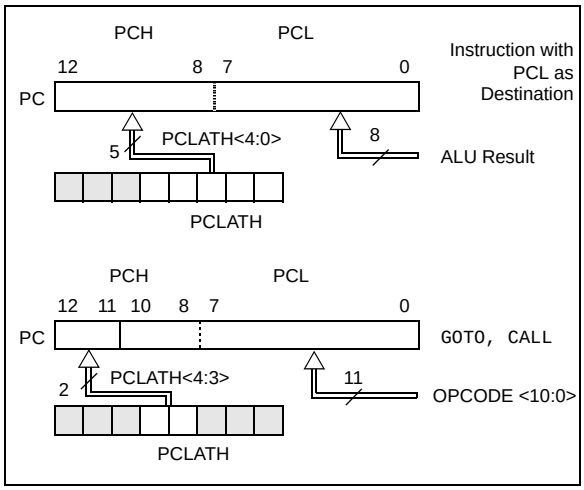
The Special Function Registers are registers used by the CPU and peripheral functions for controlling the desired operation of the device (see Table 2-1). These registers are static RAM.

The special registers can be classified into two sets: core and peripheral. The Special Function Registers associated with the "core" are described in this section. Those related to the operation of the peripheral features are described in the section of that peripheral feature.

2.3 PCL and PCLATH

The Program Counter (PC) is 13 bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<12:8>) is not directly readable or writable and comes from PCLATH. On any Reset, the PC is cleared. Figure 2-5 shows the two situations for the loading of the PC. The upper example in Figure 2-5 shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in Figure 2-5 shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

FIGURE 2-5: LOADING OF PC IN DIFFERENT SITUATIONS



2.3.1 MODIFYING PCL

Executing any instruction with the PCL register as the destination simultaneously causes the Program Counter PC<12:8> bits (PCH) to be replaced by the contents of the PCLATH register. This allows the entire contents of the program counter to be changed by writing the desired upper 5 bits to the PCLATH register. When the lower 8 bits are written to the PCL register, all 13 bits of the program counter will change to the values contained in the PCLATH register and those being written to the PCL register.

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL). Care should be exercised when jumping into a look-up table or program branch table (computed GOTO) by modifying the PCL register. Assuming that PCLATH is set to the table start address, if the table length is greater than 255 instructions or if the lower 8 bits of the memory address rolls over from 0xFF to 0x00 in the middle of the table, then PCLATH must be incremented for each address rollover that occurs between the table beginning and the target location within the table.

For more information refer to Application Note AN556, "*Implementing a Table Read*" (DS00556).

2.3.2 STACK

The PIC16F610/616/16HV610/616 Family has an 8-level x 13-bit wide hardware stack (see Figure 2-1). The stack space is not part of either program or data space and the Stack Pointer is not readable or writable. The PC is PUSHed onto the stack when a CALL instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or POP operation.

The stack operates as a circular buffer. This means that after the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

Note 1:	There are no Status bits to indicate stack overflow or stack underflow conditions.
2:	There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, RETURN, RETLW and RETFIE instructions or the vectoring to an interrupt address.

2.4 Indirect Addressing, INDF and FSR Registers

The INDF register is not a physical register. Addressing the INDF register will cause indirect addressing.

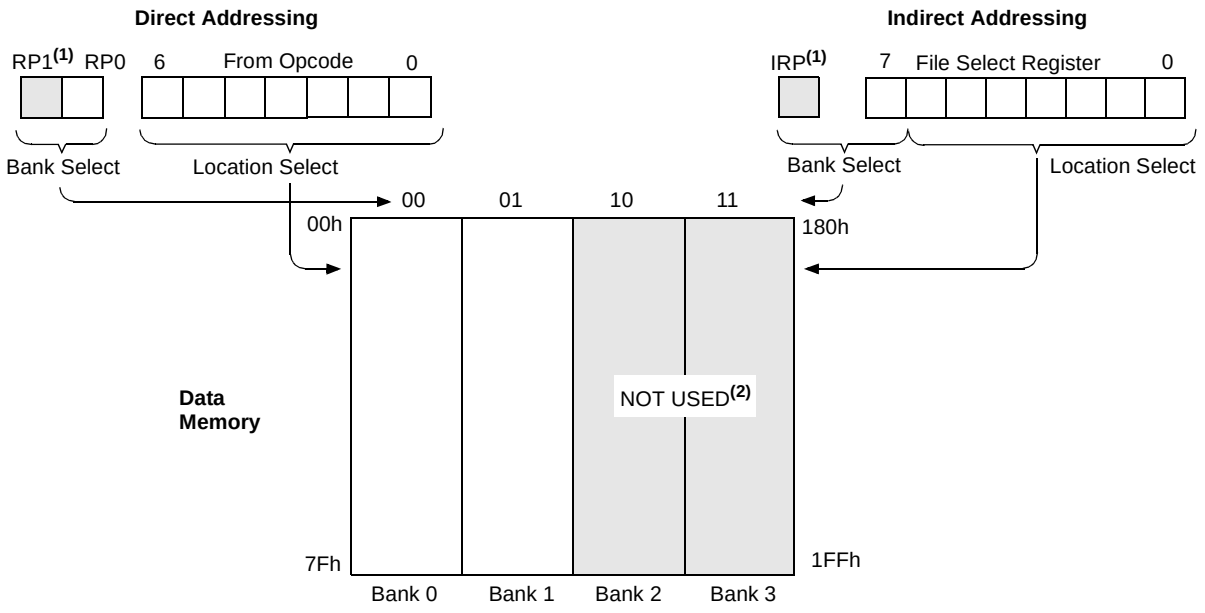
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select Register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no operation (although Status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR and the IRP bit of the STATUS register, as shown in Figure 2-7.

A simple program to clear RAM location 40h-4Fh using indirect addressing is shown in Example 2-1.

EXAMPLE 2-1: INDIRECT ADDRESSING

```
        MOVLW   0x40    ;initialize pointer
        MOVWF   FSR     ;to RAM
NEXT    CLRF    INDF    ;clear INDF register
        INCF    FSR, F  ;inc pointer
        BTFSS   FSR,4   ;all done?
        GOTO    NEXT    ;no clear next
CONTINUE                ;yes continue
```

FIGURE 2-6: DIRECT/INDIRECT ADDRESSING PIC16F610/16HV610



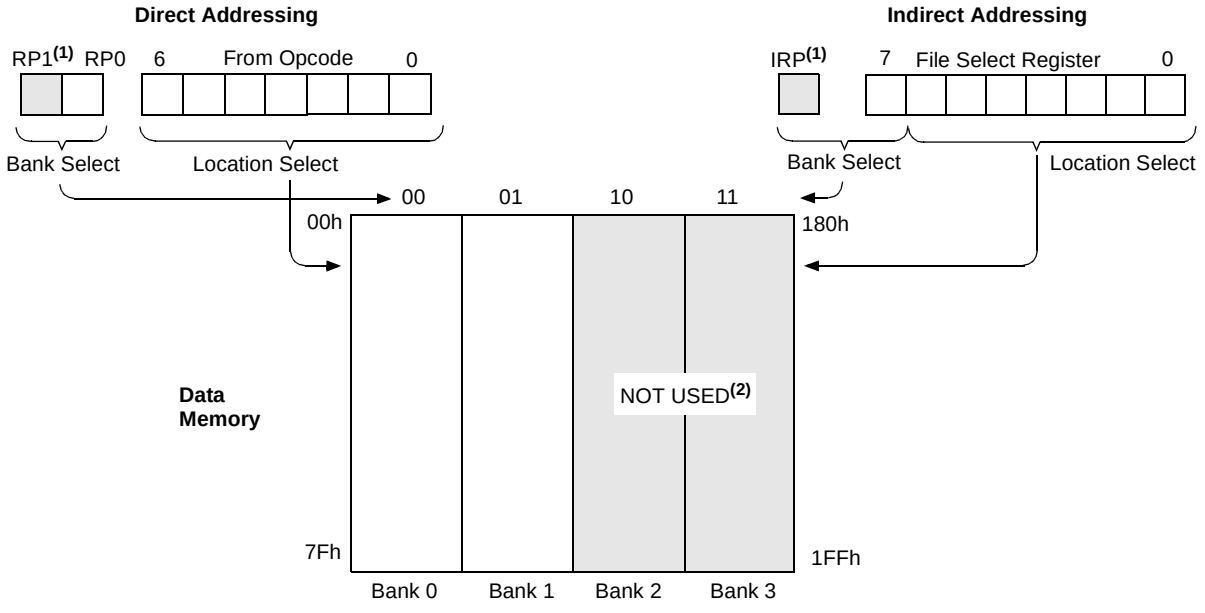
For memory map detail, see Figure 2-3.

Unimplemented data memory locations, read as '0'.

Note 1: The RP1 and IRP bits are reserved; always maintain these bits clear.

2: Accesses in Bank 2 and Bank 3 are mirrored back into Bank 0 and Bank 1, respectively.

FIGURE 2-7: DIRECT/INDIRECT ADDRESSING PIC16F616/16HV616



For memory map detail, see Figure 2-4.

Unimplemented data memory locations, read as '0'.

Note 1: The RP1 and IRP bits are reserved; always maintain these bits clear.

2: Accesses in Bank 2 and Bank 3 are mirrored back into Bank 0 and Bank 1, respectively.

NOTES:

REGISTER 5-1: OPTION_REG: OPTION REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RAPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **$\overline{\text{RAPU}}$:** PORTA Pull-up Enable bit
1 = PORTA pull-ups are disabled
0 = PORTA pull-ups are enabled by individual PORT latch values

bit 6 **INTEDG:** Interrupt Edge Select bit
1 = Interrupt on rising edge of INT pin
0 = Interrupt on falling edge of INT pin

bit 5 **T0CS:** TMR0 Clock Source Select bit
1 = Transition on T0CKI pin
0 = Internal instruction cycle clock (FOSC/4)

bit 4 **T0SE:** TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on T0CKI pin
0 = Increment on low-to-high transition on T0CKI pin

bit 3 **PSA:** Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the Timer0 module

bit 2-0 **PS<2:0>:** Prescaler Rate Select bits

BIT VALUE	TMR0 RATE	WDT RATE
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

TABLE 5-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER0

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
TMR0	Timer0 Modules Register								xxxx xxxx	uuuu uuuu
INTCON	GIE	PEIE	T0IE	INTE	RAIE	T0IF	INTF	RAIF	0000 0000	0000 0000
OPTION_REG	$\overline{\text{RAPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111

Legend: – = Unimplemented locations, read as '0', u = unchanged, x = unknown. Shaded cells are not used by the Timer0 module.

NOTES:

6.0 TIMER1 MODULE WITH GATE CONTROL

The Timer1 module is a 16-bit timer/counter with the following features:

- 16-bit timer/counter register pair (TMR1H:TMR1L)
- Programmable internal or external clock source
- 3-bit prescaler
- Optional LP oscillator
- Synchronous or asynchronous operation
- Timer1 gate (count enable) via comparator or $\overline{T1G}$ pin
- Interrupt on overflow
- Wake-up on overflow (external clock, Asynchronous mode only)
- Time base for the Capture/Compare function
- Special Event Trigger (with ECCP)
- Comparator output synchronization to Timer1 clock

Figure 6-1 is a block diagram of the Timer1 module.

6.1 Timer1 Operation

The Timer1 module is a 16-bit incrementing counter which is accessed through the TMR1H:TMR1L register pair. Writes to TMR1H or TMR1L directly update the counter.

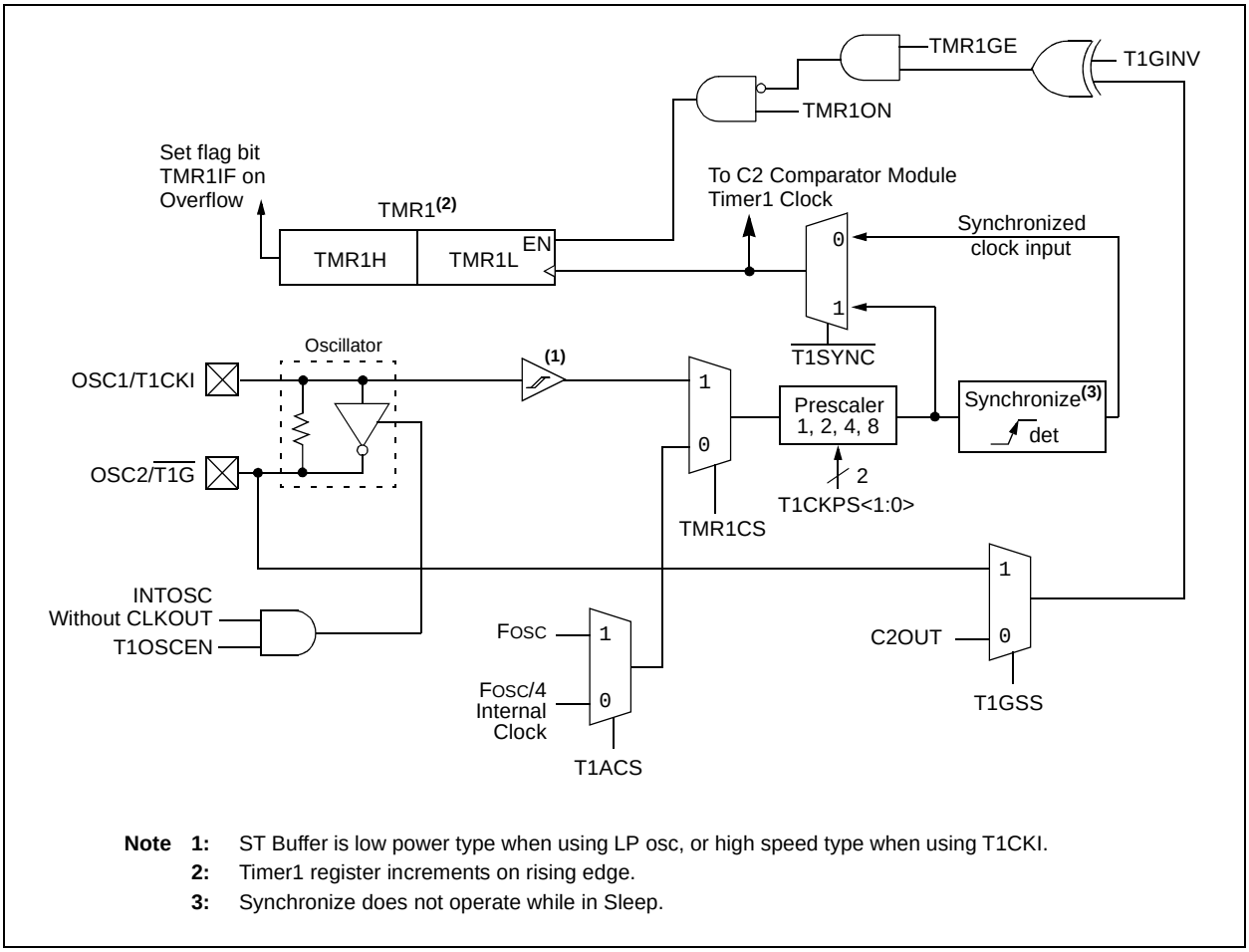
When used with an internal clock source, the module is a timer. When used with an external clock source, the module can be used as either a timer or counter.

6.2 Clock Source Selection

The TMR1CS bit of the T1CON register is used to select the clock source. When TMR1CS = 0, the clock source is FOSC/4. When TMR1CS = 1, the clock source is supplied externally.

Clock Source	TMR1CS	T1ACS
FOSC/4	0	0
FOSC	0	1
T1CKI pin	1	x

FIGURE 6-1: TIMER1 BLOCK DIAGRAM



Note 1: ST Buffer is low power type when using LP osc, or high speed type when using T1CKI.
2: Timer1 register increments on rising edge.
3: Synchronize does not operate while in Sleep.

8.4 Comparator Interrupt Operation

The comparator interrupt flag can be set whenever there is a change in the output value of the comparator. Changes are recognized by means of a mismatch circuit which consists of two latches and an exclusive-or gate (see Figure 8-2 and Figure 8-3). One latch is updated with the comparator output level when the CMxCON0 register is read. This latch retains the value until the next read of the CMxCON0 register or the occurrence of a Reset. The other latch of the mismatch circuit is updated on every Q1 system clock. A mismatch condition will occur when a comparator output change is clocked through the second latch on the Q1 clock cycle. At this point the two mismatch latches have opposite output levels which is detected by the exclusive-or gate and fed to the interrupt circuitry. The mismatch condition persists until either the CMxCON0 register is read or the comparator output returns to the previous state.

Note 1: A write operation to the CMxCON0 register will also clear the mismatch condition because all writes include a read operation at the beginning of the write cycle.

2: Comparator interrupts will operate correctly regardless of the state of CxOE.

The comparator interrupt is set by the mismatch edge and not the mismatch level. This means that the interrupt flag can be reset without the additional step of reading or writing the CMxCON0 register to clear the mismatch registers. When the mismatch registers are cleared, an interrupt will occur upon the comparator's return to the previous state, otherwise no interrupt will be generated.

Software will need to maintain information about the status of the comparator output, as read from the CMxCON0 register, or CM2CON1 register, to determine the actual change that has occurred.

The CxIF bit of the PIR1 register is the comparator interrupt flag. This bit must be reset in software by clearing it to '0'. Since it is also possible to write a '1' to this register, an interrupt can be generated.

The CxIE bit of the PIE1 register and the PEIE and GIE bits of the INTCON register must all be set to enable comparator interrupts. If any of these bits are cleared, the interrupt is not enabled, although the CxIF bit of the PIR1 register will still be set if an interrupt condition occurs.

FIGURE 8-4: COMPARATOR INTERRUPT TIMING W/O CMxCON0 READ

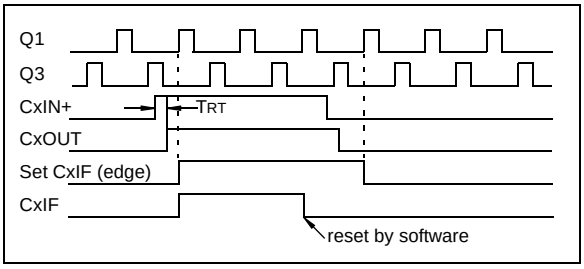
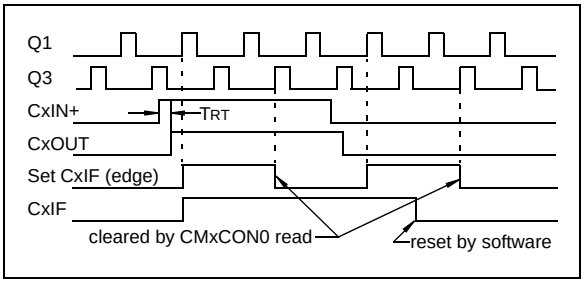


FIGURE 8-5: COMPARATOR INTERRUPT TIMING WITH CMxCON0 READ



Note 1: If a change in the CMxCON0 register (CxOUT) should occur when a read operation is being executed (start of the Q2 cycle), then the CxIF of the PIR1 register interrupt flag may not get set.

2: When either comparator is first enabled, bias circuitry in the comparator module may cause an invalid output from the comparator until the bias circuitry is stable. Allow about 1 μs for bias settling then clear the mismatch condition and interrupt flags before enabling comparator interrupts.

8.11 Comparator Voltage Reference

The comparator voltage reference module provides an internally generated voltage reference for the comparators. The following features are available:

- Independent from Comparator operation
- Two 16-level voltage ranges
- Output clamped to VSS
- Ratiometric with VDD
- Fixed Reference (0.6V)

The VRCON register (Register 8-6) controls the voltage reference module shown in Figure 8-9.

8.11.1 INDEPENDENT OPERATION

The comparator voltage reference is independent of the comparator configuration. Setting the FVREN bit of the VRCON register will enable the voltage reference.

8.11.2 OUTPUT VOLTAGE SELECTION

The CVREF voltage reference has 2 ranges with 16 voltage levels in each range. Range selection is controlled by the VRR bit of the VRCON register. The 16 levels are set with the VR<3:0> bits of the VRCON register.

The CVREF output voltage is determined by the following equations:

EQUATION 8-1: CVREF OUTPUT VOLTAGE

$$\begin{aligned} &VRR = 1 \text{ (low range):} \\ &\quad CVREF = (VR<3:0>/24) \times VDD \\ &VRR = 0 \text{ (high range):} \\ &\quad CVREF = (VDD/4) + (VR<3:0> \times VDD/32) \end{aligned}$$

The full range of VSS to VDD cannot be realized due to the construction of the module. See Figure 8-9.

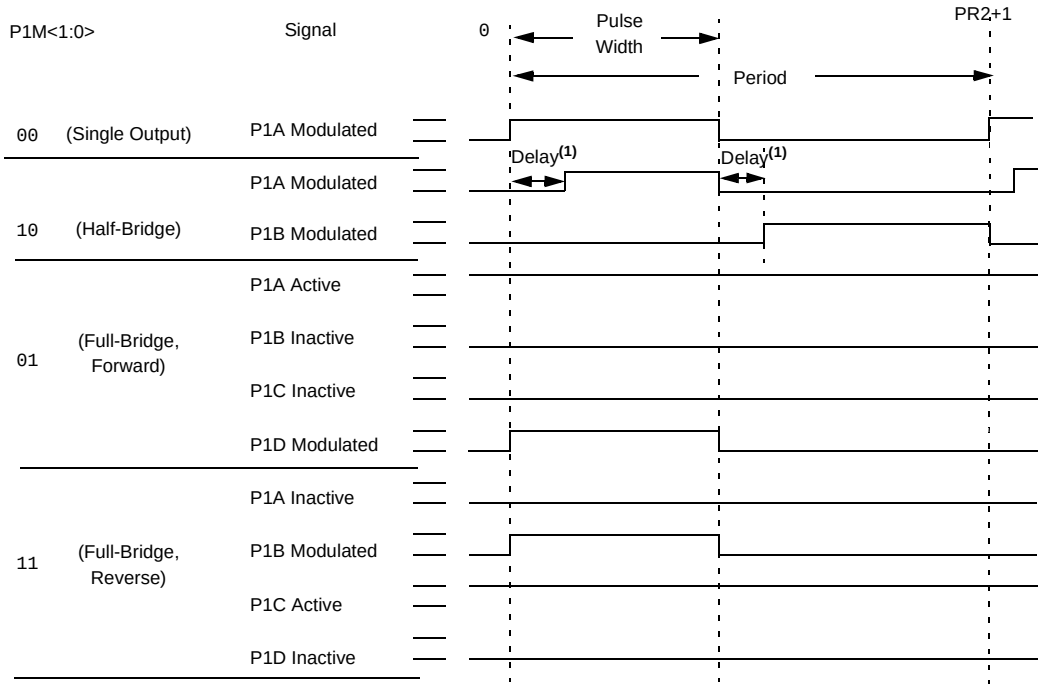
8.11.3 OUTPUT CLAMPED TO VSS

The fixed voltage reference output voltage can be set to Vss with no power consumption by clearing the FVREN bit of the VRCON register (FVREN = 0). This allows the comparator to detect a zero-crossing while not consuming additional module current.

8.11.4 OUTPUT RATIOMETRIC TO VDD

The comparator voltage reference is VDD derived and therefore, the CVREF output changes with fluctuations in VDD. The tested absolute accuracy of the Comparator Voltage Reference can be found in **Section 15.0 "Electrical Specifications"**.

FIGURE 10-6: EXAMPLE PWM (ENHANCED MODE) OUTPUT RELATIONSHIPS (ACTIVE-HIGH STATE)



Relationships:

- Period = 4 * Tosc * (PR2 + 1) * (TMR2 Prescale Value)
- Pulse Width = Tosc * (CCPR1L<7:0>:CCP1CON<5:4>) * (TMR2 Prescale Value)
- Delay = 4 * Tosc * (PWM1CON<6:0>)

Note 1: Dead-band delay is programmed using the PWM1CON register (**Section 10.4.6 "Programmable Dead-Band Delay mode"**).

14.11 PICkit 2 Development Programmer/Debugger and PICkit 2 Debug Express

The PICkit™ 2 Development Programmer/Debugger is a low-cost development tool with an easy to use interface for programming and debugging Microchip's Flash families of microcontrollers. The full featured Windows® programming interface supports baseline (PIC10F, PIC12F5xx, PIC16F5xx), midrange (PIC12F6xx, PIC16F), PIC18F, PIC24, dsPIC30, dsPIC33, and PIC32 families of 8-bit, 16-bit, and 32-bit microcontrollers, and many Microchip Serial EEPROM products. With Microchip's powerful MPLAB Integrated Development Environment (IDE) the PICkit™ 2 enables in-circuit debugging on most PIC® microcontrollers. In-Circuit-Debugging runs, halts and single steps the program while the PIC microcontroller is embedded in the application. When halted at a breakpoint, the file registers can be examined and modified.

The PICkit 2 Debug Express include the PICkit 2, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

14.12 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at V_{DDMIN} and V_{DDMAX} for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages and a modular, detachable socket assembly to support various package types. The ICSP™ cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices and incorporates an MMC card for file storage and data applications.

14.13 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

15.10 Thermal Considerations

Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C					
Param No.	**Sym**	**Characteristic**	**Typ**	**Units**	**Conditions**
TH01	θJA	Thermal Resistance Junction to Ambient	70*	C/W	14-pin PDIP package
			85.0*	C/W	14-pin SOIC package
			100*	C/W	14-pin TSSOP package
			37*	C/W	16-pin QFN 4x4mm package
TH02	θJC	Thermal Resistance Junction to Case	32.5*	C/W	14-pin PDIP package
			31.0*	C/W	14-pin SOIC package
			31.7*	C/W	14-pin TSSOP package
			2.6*	C/W	16-pin QFN 4x4mm package
TH03	TDIE	Die Temperature	150*	C	
TH04	PD	Power Dissipation	—	W	PD = PINTERNAL + PI/O
TH05	PINTERNAL	Internal Power Dissipation	—	W	PINTERNAL = IDD x VDD **(NOTE 1)**
TH06	PI/O	I/O Power Dissipation	—	W	PI/O = Σ (IOL * VOL) + Σ (IOH * (VDD - VOH))
TH07	PDER	Derated Power	—	W	PDER = PDMAX (TDIE - TA)/θJA **(NOTE 2)**

* These parameters are characterized but not tested.

Note 1: IDD is current to run the chip alone without driving any load on the output pins.

2: TA = Ambient Temperature.

TABLE 15-14: DC CHARACTERISTICS FOR IDD SPECIFICATIONS FOR PIC16F616 – H (High Temp.)

Param No.	Device Characteristics	Units	Min	Typ	Max	Condition	
						VDD	Note
D010	Supply Current (IDD)	μA	—	13	58	2.0	IDD LP OSC (32 kHz)
			—	19	67	3.0	
			—	32	92	5.0	
D011		μA	—	135	316	2.0	IDD XT OSC (1 MHz)
			—	185	400	3.0	
			—	300	537	5.0	
D012		μA	—	240	495	2.0	IDD XT OSC (4 MHz)
			—	360	680	3.0	
		mA	—	0.660	1.20	5.0	
D013		μA	—	75	158	2.0	IDD EC OSC (1 MHz)
			—	155	338	3.0	
			—	345	792	5.0	
D014		μA	—	185	357	2.0	IDD EC OSC (4 MHz)
			—	325	625	3.0	
		mA	—	0.665	1.30	5.0	
D016		μA	—	245	476	2.0	IDD INTOSC (4 MHz)
			—	360	672	3.0	
			—	620	1.10	5.0	
D017		μA	—	395	757	2.0	IDD INTOSC (8 MHz)
		mA	—	0.620	1.20	3.0	
			—	1.20	2.20	5.0	
D018		μA	—	175	332	2.0	IDD EXTRC (4 MHz)
			—	285	518	3.0	
			—	530	972	5.0	
D019		mA	—	2.20	4.10	4.5	IDD HS OSC (20 MHz)
			—	2.80	4.80	5.0	

TABLE 15-15: DC CHARACTERISTICS FOR I_{PD} SPECIFICATIONS FOR PIC16F616 – H (High Temp.)

Param No.	Device Characteristics	Units	Min	Typ	Max	Condition	
						V_{DD}	Note
D020E	Power Down I_{PD}	µA	—	0.05	12	2.0	I_{PD} Base
			—	0.15	13	3.0	
			—	0.35	14	5.0	
D021E		µA	—	0.5	20	2.0	WDT Current
			—	2.5	25	3.0	
			—	9.5	36	5.0	
D022E		µA	—	5.0	28	3.0	BOR Current
			—	6.0	36	5.0	
D023E		µA	—	105	195	2.0	I_{PD} Current (Both Comparators Enabled)
			—	110	210	3.0	
			—	116	220	5.0	
		µA	—	50	105	2.0	I_{PD} Current (One Comparator Enabled)
			—	55	110	3.0	
			—	60	125	5.0	
D024E		µA	—	30	58	2.0	I_{PD} (C_{VREF}, High Range)
			—	45	85	3.0	
			—	75	142	5.0	
D025E		µA	—	39	76	2.0	I_{PD} (C_{VREF}, Low Range)
			—	59	114	3.0	
			—	98	190	5.0	
D026E		µA	—	5.5	30	2.0	I_{PD} (T1 OSC, 32 kHz)
			—	7.0	35	3.0	
			—	8.5	45	5.0	
D027E		µA	—	0.2	12	3.0	I_{PD} (A2D on, not converting)
			—	0.3	15	5.0	

TABLE 15-16: WATCHDOG TIMER SPECIFICATIONS FOR PIC16F616 – H (High Temp.)

Param No.	Sym	Characteristic	Units	Min	Typ	Max	Conditions
31	T_{WDT}	Watchdog Timer Time-out Period (No Prescaler)	ms	6	20	70	150°C Temperature

TABLE 15-17: LEAKAGE CURRENT SPECIFICATIONS FOR PIC16F616 – H (High Temp.)

Param No.	Sym	Characteristic	Units	Min	Typ	Max	Conditions
D061	I_{IL}	Input Leakage Current(1) (GP3/RA3/$\overline{MCLR}$)	µA	—	±0.5	±5.0	$V_{SS} \le V_{PIN} \le V_{DD}$
D062	I_{IL}	Input Leakage Current(2) (GP3/RA3/$\overline{MCLR}$)	µA	50	250	400	V_{DD} = 5.0V

Note 1: This specification applies when GP3/RA3/$\overline{MCLR}$ is configured as an input with the pull-up disabled. The leakage current for the GP3/RA3/$\overline{MCLR}$ pin is higher than for the standard I/O port pins.

2: This specification applies when GP3/RA3/$\overline{MCLR}$ is configured as the $\overline{MCLR}$ Reset pin function with the weak pull-up enabled.

FIGURE 16-8: PIC16F610/616 IDD EXTRC (4 MHz) vs. VDD

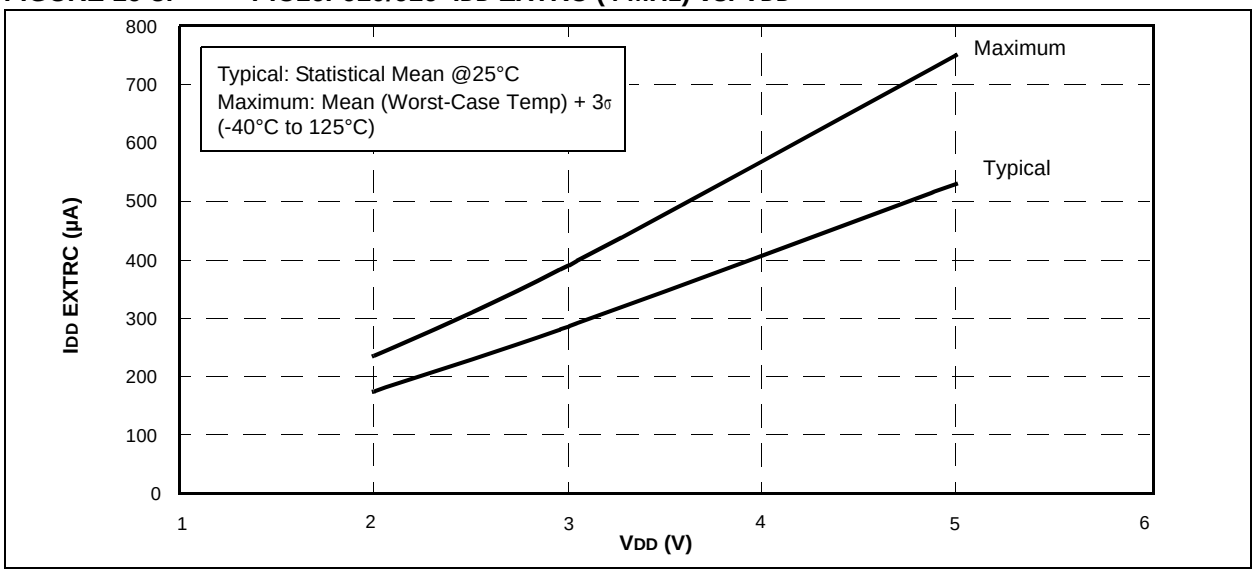


FIGURE 16-9: PIC16F610/616 IDD HS (20 MHz) vs. VDD

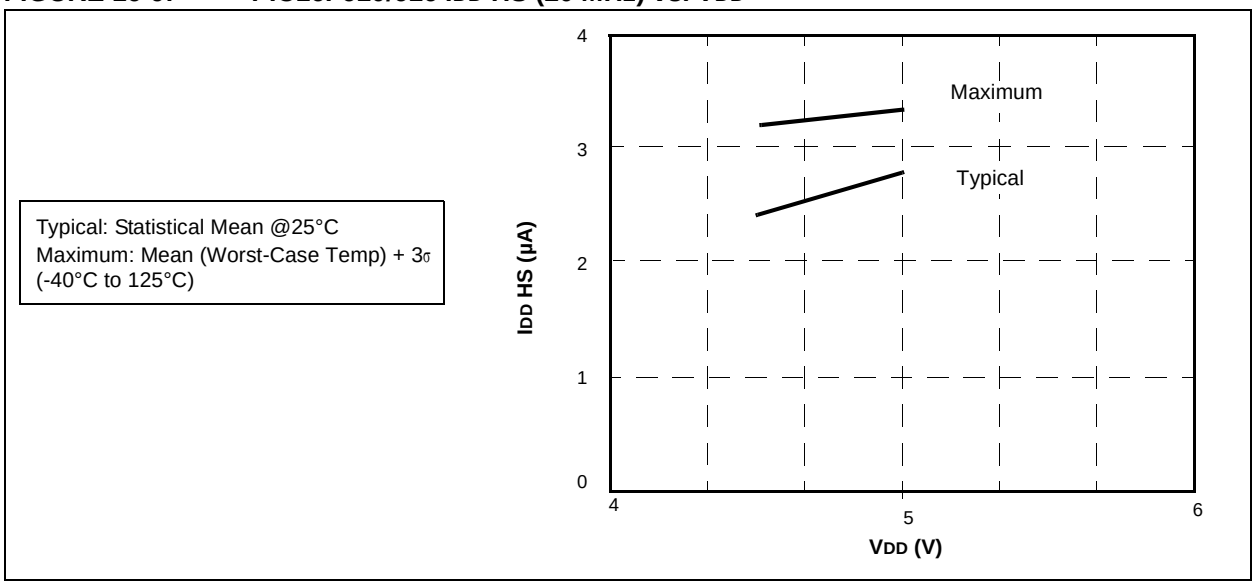


FIGURE 16-29: PIC16HV610/616 IPD COMPARATOR (BOTH ON) vs. VDD

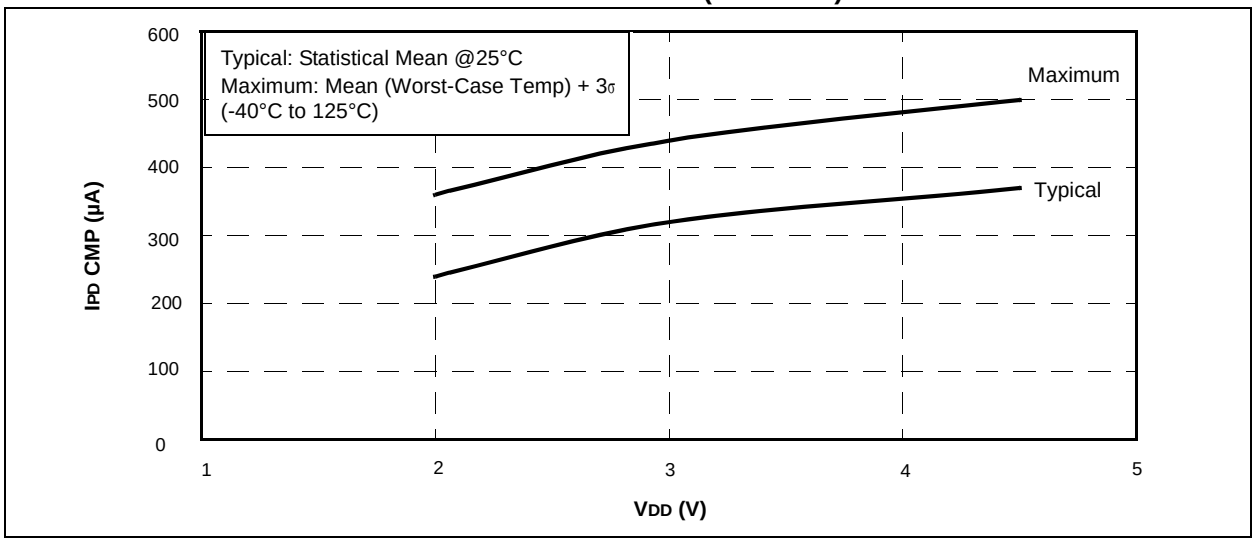


FIGURE 16-30: PIC16HV610/616 IPD WDT vs. VDD

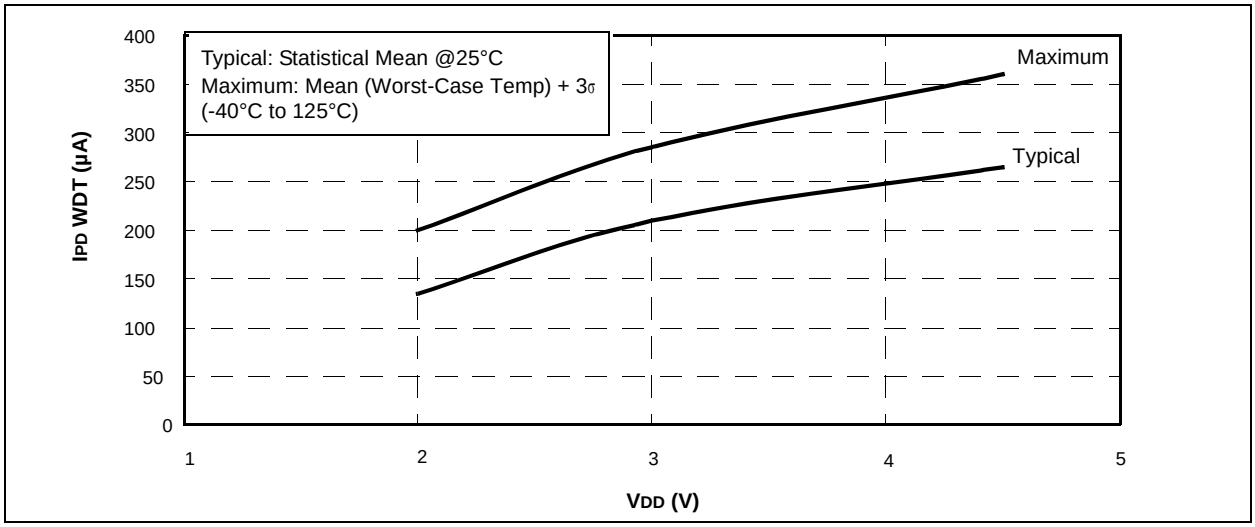


FIGURE 16-31: PIC16HV610/616 IPD BOR vs. VDD

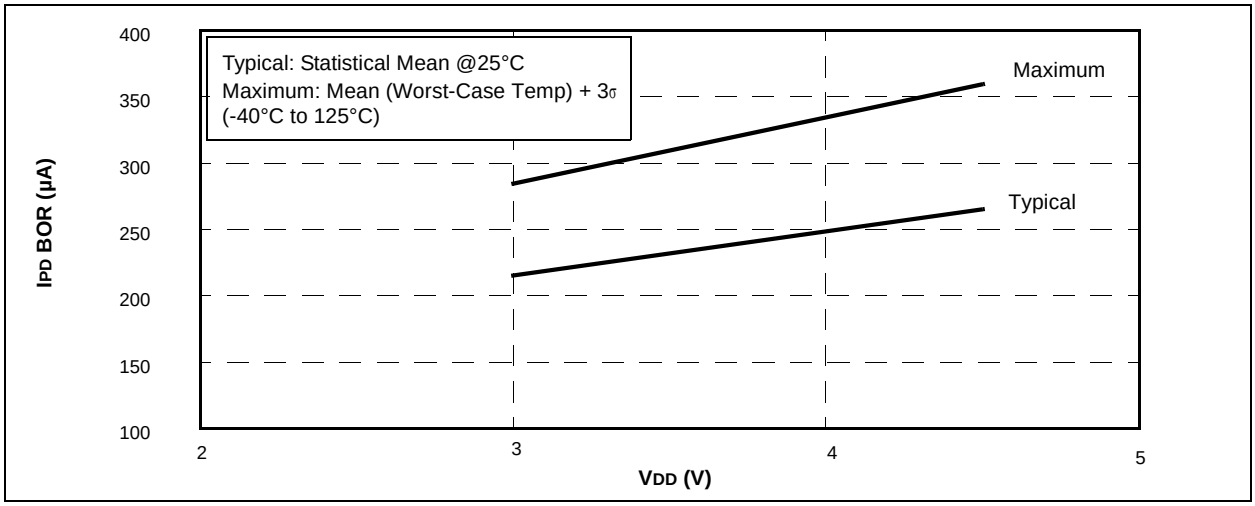


FIGURE 16-47: TYPICAL HFINTOSC FREQUENCY CHANGE vs. VDD (125°C)

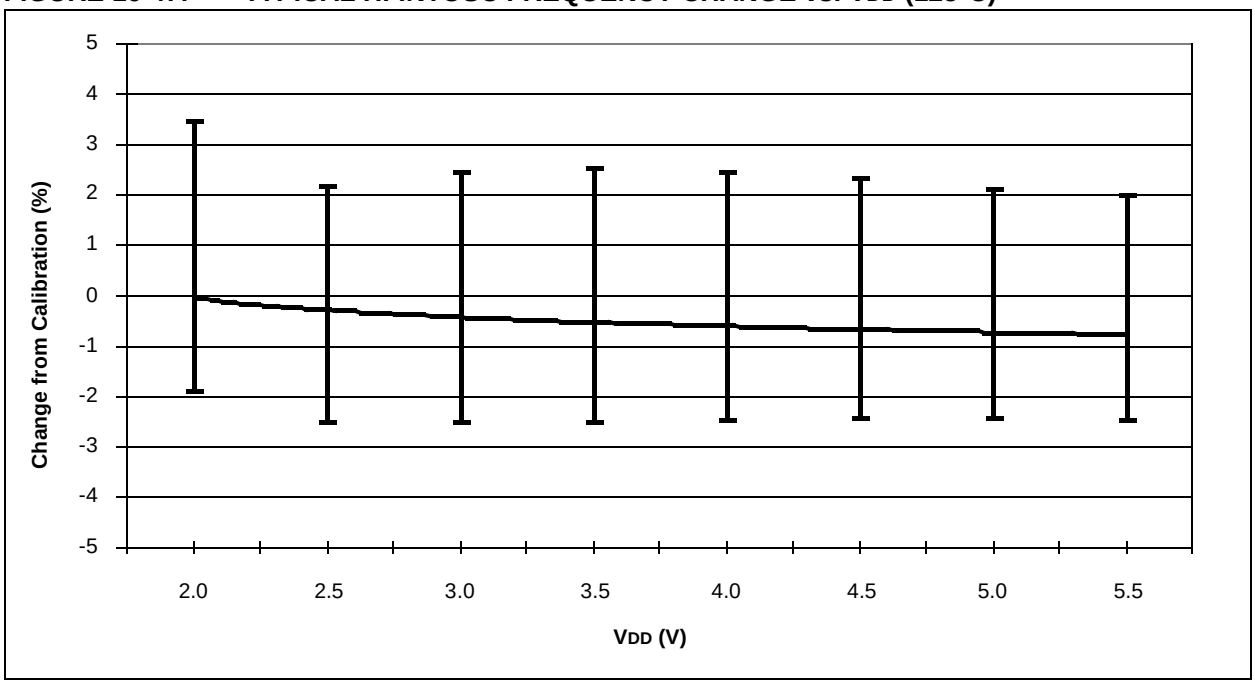
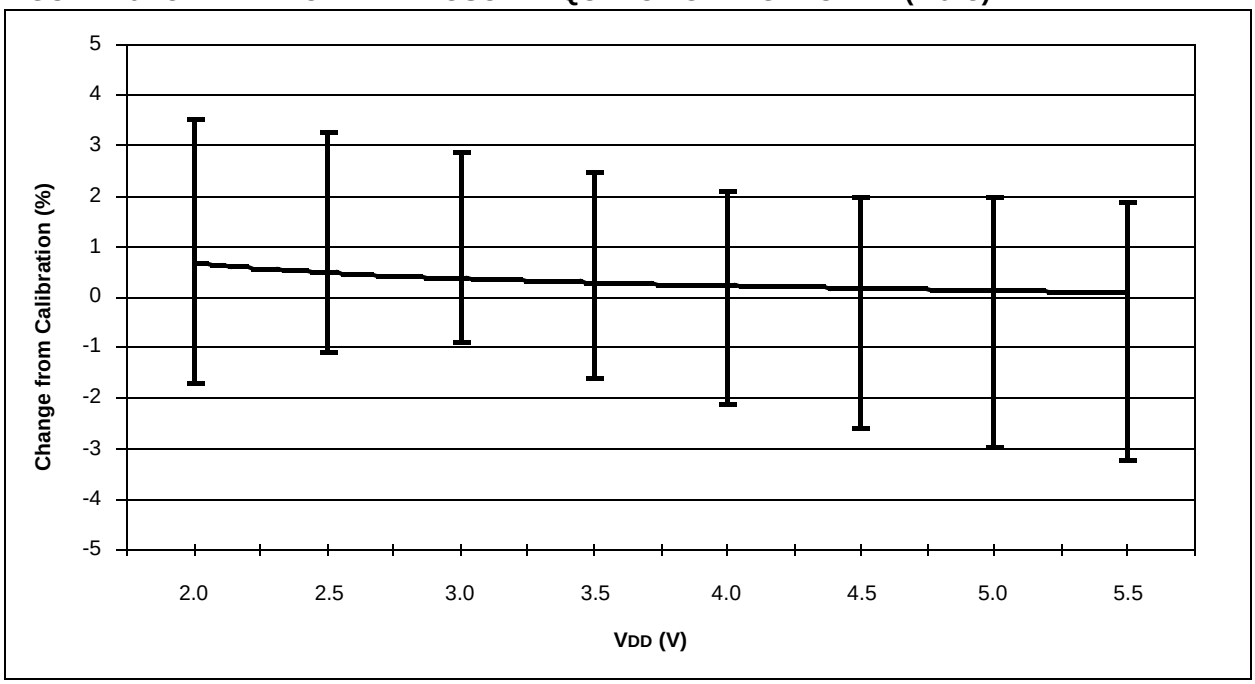


FIGURE 16-48: TYPICAL HFINTOSC FREQUENCY CHANGE vs. VDD (-40°C)



APPENDIX B: MIGRATING FROM OTHER PIC® DEVICES

This discusses some of the issues in migrating from other PIC® devices to the PIC16F6XX Family of devices.

B.1 PIC16F676 to PIC16F610/616/16HV610/616

TABLE B-1: FEATURE COMPARISON

Feature	PIC16F676	PIC16F610/16HV610	PIC16F616/16HV616
Max Operating Speed	20 MHz	20 MHz	20 MHz
Max Program Memory (Words)	1024	1024	2048
SRAM (bytes)	64	64	128
A/D Resolution	10-bit	None	10-bit
Timers (8/16-bit)	1/1	1/1	2/1
Oscillator Modes	8	8	8
Brown-out Reset	Y	Y	Y
Internal Pull-ups	RA0/1/2/4/5	RA0/1/2/4/5, $\overline{\text{MCLR}}$	RA0/1/2/4/5, $\overline{\text{MCLR}}$
Interrupt-on-change	RA0/1/2/3/4/5	RA0/1/2/3/4/5	RA0/1/2/3/4/5
Comparator	1	2	2
ECCP	N	N	Y
INTOSC Frequencies	4 MHz	4/8 MHz	4/8 MHz
Internal Shunt Regulator	N	Y (PIC16HV610)	Y (PIC16HV616)

Note: This device has been designed to perform to the parameters of its data sheet. It has been tested to an electrical specification designed to determine its conformance with these parameters. Due to process differences in the manufacture of this device, this device may have different performance characteristics than its earlier version. These differences may cause this device to perform differently in your application than the earlier version of this device.

INDEX