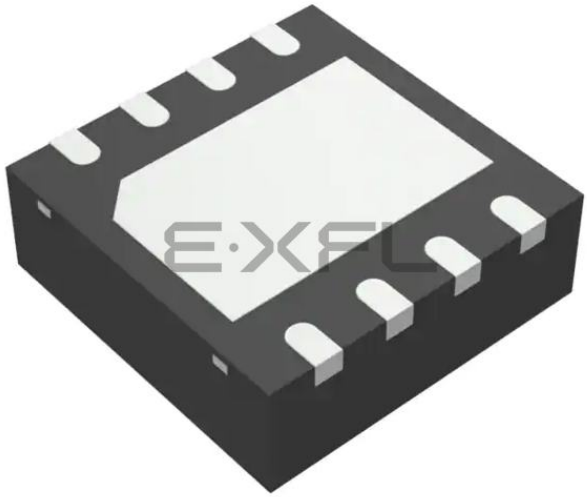


Welcome to E-XFL.COM

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	5
Program Memory Size	1.75KB (1K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	64 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	8-VDFN Exposed Pad
Supplier Device Package	8-DFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic12hv609-e-md

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.

2.0 MEMORY ORGANIZATION

2.1 Program Memory Organization

The PIC12F609/615/617/12HV609/615 has a 13-bit program counter capable of addressing an 8K x 14 program memory space. Only the first 1K x 14 (0000h-03FFh) for the PIC12F609/615/12HV609/615 is physically implemented. For the PIC12F617, the first 2K x 14 (0000h-07FFh) is physically implemented. Accessing a location above these boundaries will cause a wrap-around within the first 1K x 14 space for PIC12F609/615/12HV609/615 devices, and within the first 2K x 14 space for the PIC12F617 device. The Reset vector is at 0000h and the interrupt vector is at 0004h (see Figure 2-1).

FIGURE 2-1: PROGRAM MEMORY MAP AND STACK FOR THE PIC12F609/615/12HV609/615

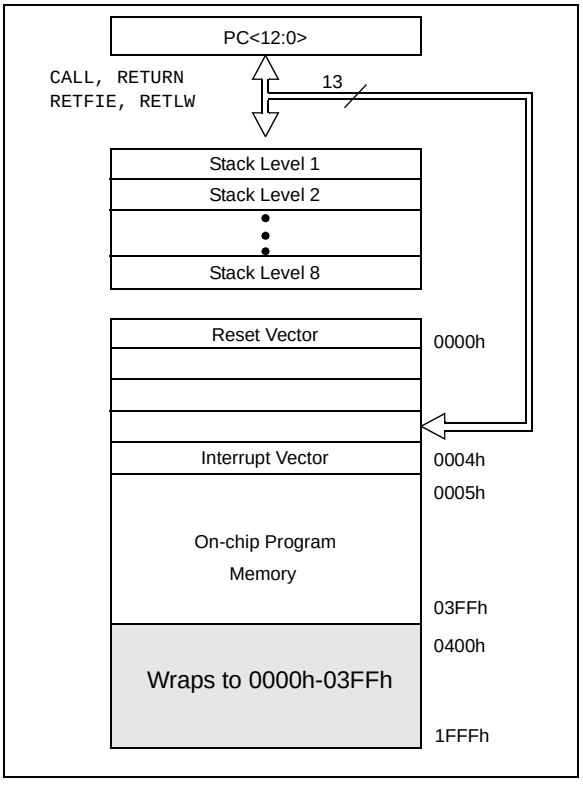
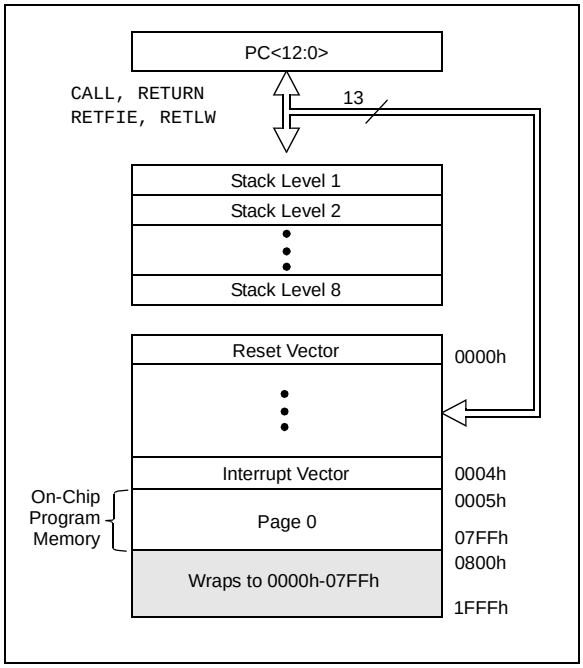


FIGURE 2-2: PROGRAM MEMORY MAP AND STACK FOR THE PIC12F617



2.2 Data Memory Organization

The data memory (see Figure 2-3) 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. Register locations 40h-7Fh in Bank 0 are General Purpose Registers, implemented as static RAM. For the PIC12F617, the register locations 20h-7Fh in Bank 0 and A0h-EFh 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 PIC12F609/615/12HV609/615, and as 128 x 8 in the PIC12F617. 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.

FIGURE 2-3: DATA MEMORY MAP OF THE PIC12F609/HV609

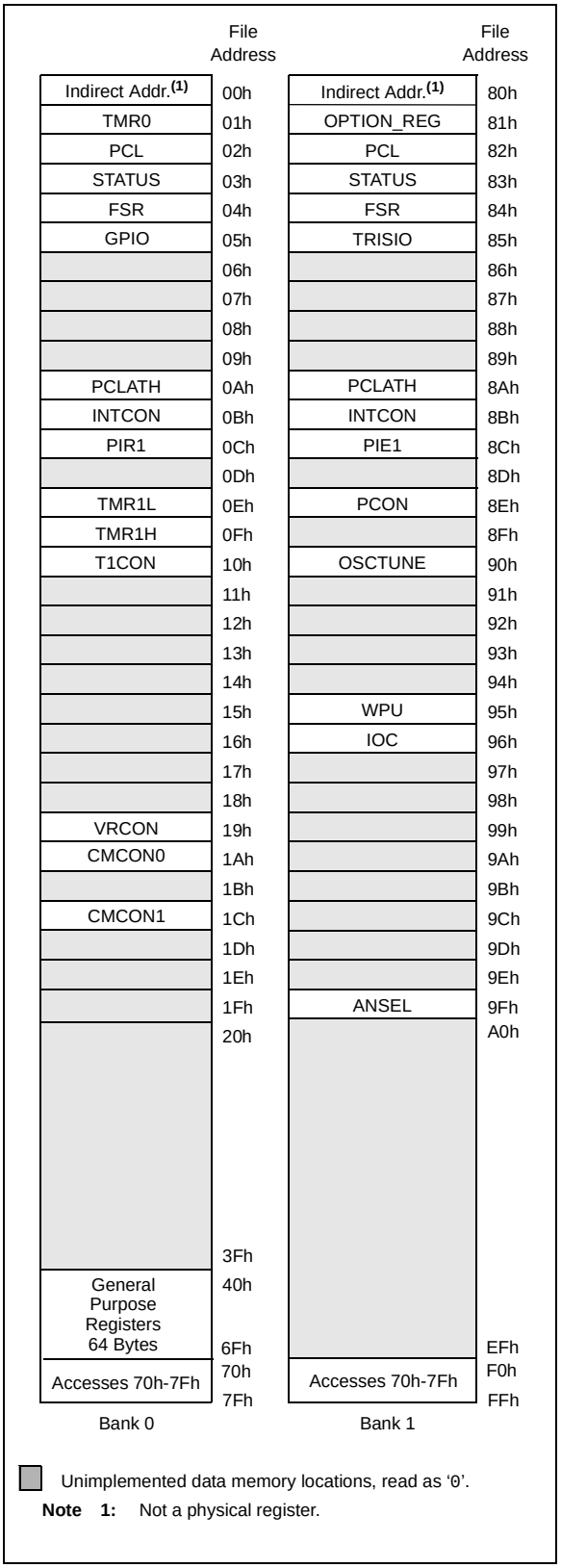


TABLE 2-4: PIC12F615/617/HV615 SPECIAL FUNCTION REGISTERS SUMMARY BANK 1

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Page
Bank 1											
80h	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	25, 116
81h	OPTION_REG	$\overline{GPPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	19, 116
82h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	25, 116
83h	STATUS	IRP(1)	RP1(1)	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	18, 116
84h	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	25, 116
85h	TRISIO	—	—	TRISIO5	TRISIO4	TRISIO3(4)	TRISIO2	TRISIO1	TRISIO0	--11 1111	44, 116
86h	—	Unimplemented								—	—
87h	—	Unimplemented								—	—
88h	—	Unimplemented								—	—
89h	—	Unimplemented								—	—
8Ah	PCLATH	—	—	—	Write Buffer for upper 5 bits of Program Counter					---0 0000	25, 116
8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF(3)	0000 0000	20, 116
8Ch	PIE1	—	ADIE	CCP1IE	—	CMIE	—	TMR2IE	TMR1IE	-00- 0-00	21, 116
8Dh	—	Unimplemented								—	—
8Eh	PCON	—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$	---- --qq	23, 116
8Fh	—	Unimplemented								—	—
90h	OSCTUNE	—	—	—	TUN4	TUN3	TUN2	TUN1	TUN0	---0 0000	41, 116
91h	—	Unimplemented								—	—
92h	PR2	Timer2 Module Period Register								1111 1111	65, 116
93h	APFCON	—	—	—	T1GSEL	—	—	P1BSEL	P1ASEL	---0 --00	21, 116
94h	—	Unimplemented								—	—
95h	WPU(2)	—	—	WPU5	WPU4	—	WPU2	WPU1	WPU0	--11 -111	46, 116
96h	IOC	—	—	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0	--00 0000	46, 116
97h	—	Unimplemented								—	—
98h	PMCON1(7)	—	—	—	—	—	WREN	WR	RD	---- -000	29
99h	PMCON2(7)	Program Memory Control Register 2 (not a physical register).								---- ----	—
9Ah	PMADRL(7)	PMADRL7	PMADRL6	PMADRL5	PMADRL4	PMADRL3	PMADRL2	PMADRL1	PMADRL0	0000 0000	28
9Bh	PMADRH(7)	—	—	—	—	—	PMADRH2	PMADRH1	PMADRH0	---- -000	28
9Ch	PMDATL(7)	PMDATL7	PMDATL6	PMDATL5	PMDATL4	PMDATL3	PMDATL2	PMDATL1	PMDATL0	0000 0000	28
9Dh	PMDATH(7)	—	—	Program Memory Data Register High Byte.						--00 0000	28
9Eh	ADRESL(5, 6)	Least Significant 2 bits of the left shifted result or 8 bits of the right shifted result								xxxx xxxx	85, 117
9Fh	ANSEL	—	ADCS2	ADCS1	ADCS0	ANS3	ANS2	ANS1	ANS0	-000 1111	45, 117

Legend: – = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition, shaded = unimplemented

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

2: GP3 pull-up is enabled when MCLRE is '1' in the Configuration Word register.

3: $\overline{MCLR}$ and WDT Reset does not affect the previous value data latch. The GPIF bit will clear upon Reset but will set again if the mismatch exists.

4: TRISIO3 always reads as '1' since it is an input only pin.

5: Read only register.

6: PIC12F615/617/HV615 only.

7: PIC12F617 only.

9.0 COMPARATOR MODULE

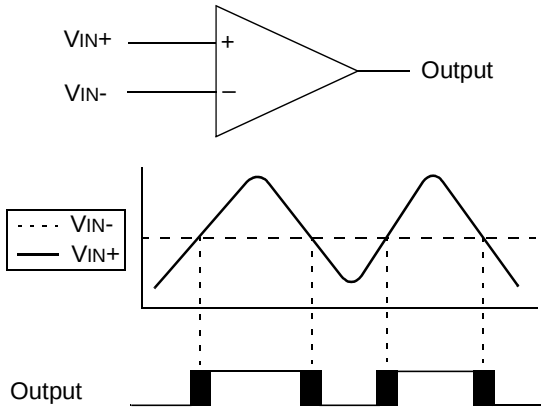
The comparator can be used to interface analog circuits to a digital circuit by comparing two analog voltages and providing a digital indication of their relative magnitudes. The comparator is a very useful mixed signal building block because it provides analog functionality independent of the program execution. The Analog Comparator module includes the following features:

- Programmable input section
- Comparator output is available internally/externally
- Programmable output polarity
- Interrupt-on-change
- Wake-up from Sleep
- PWM shutdown
- Timer1 gate (count enable)
- Output synchronization to Timer1 clock input
- Programmable voltage reference
- User-enable Comparator Hysteresis

9.1 Comparator Overview

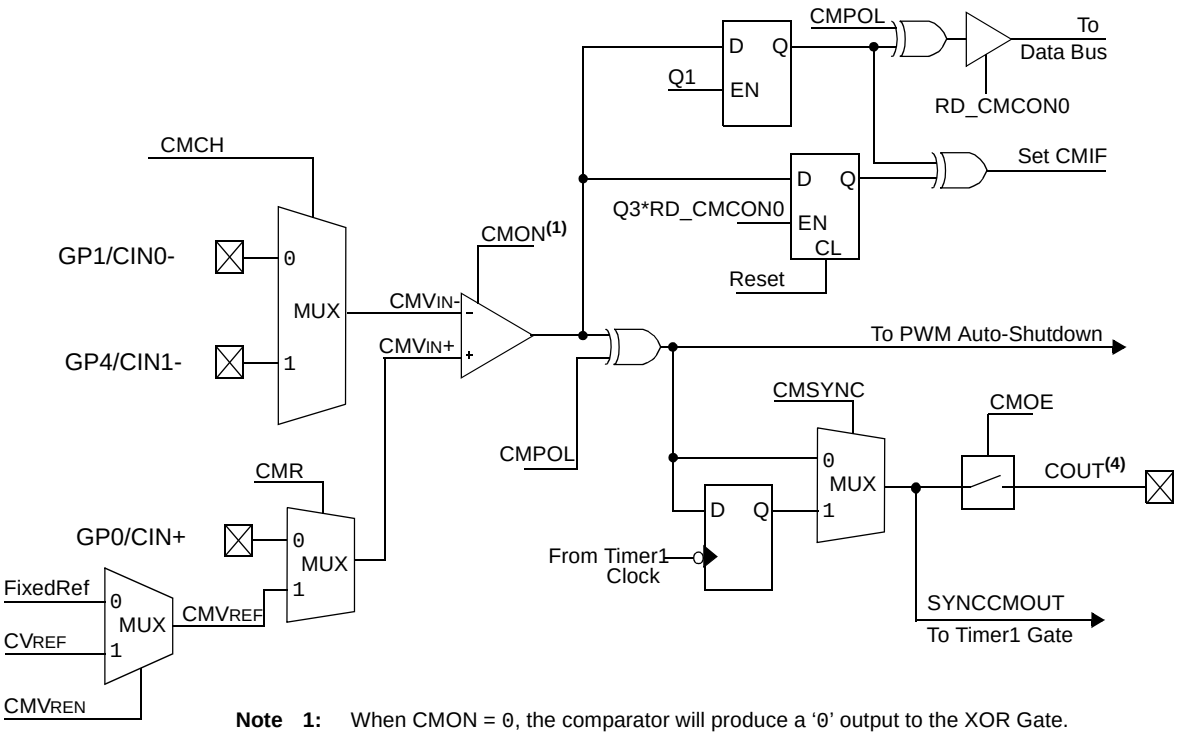
The comparator is shown in Figure 9-1 along with the relationship between the analog input levels and the digital output. When the analog voltage at VIN+ is less than the analog voltage at VIN-, the output of the comparator is a digital low level. When the analog voltage at VIN+ is greater than the analog voltage at VIN-, the output of the comparator is a digital high level.

FIGURE 9-1:SINGLE COMPARATOR



Note: The black areas of the output of the comparator represents the uncertainty due to input offsets and response time.

FIGURE 9-2: COMPARATOR SIMPLIFIED BLOCK DIAGRAM



Note 1: When CMON = 0, the comparator will produce a '0' output to the XOR Gate.
2: Q1 and Q3 are phases of the four-phase system clock (FOSC).
3: Q1 is held high during Sleep mode.
4: Output shown for reference only. See I/O port pin diagram for more details.

10.1 ADC Configuration

When configuring and using the ADC the following functions must be considered:

- Port configuration
- Channel selection
- ADC voltage reference selection
- ADC conversion clock source
- Interrupt control
- Results formatting

10.1.1 PORT CONFIGURATION

The ADC can be used to convert both analog and digital signals. When converting analog signals, the I/O pin should be configured for analog by setting the associated TRIS and ANSEL bits. See the corresponding port section for more information.

Note:	Analog voltages on any pin that is defined as a digital input may cause the input buffer to conduct excess current.

10.1.2 CHANNEL SELECTION

The CHS bits of the ADCON0 register determine which channel is connected to the sample and hold circuit.

When changing channels, a delay is required before starting the next conversion. Refer to **Section 10.2 "ADC Operation"** for more information.

10.1.3 ADC VOLTAGE REFERENCE

The VCFG bit of the ADCON0 register provides control of the positive voltage reference. The positive voltage reference can be either VDD or an external voltage source. The negative voltage reference is always connected to the ground reference.

10.1.4 CONVERSION CLOCK

The source of the conversion clock is software selectable via the ADCS bits of the ANSEL register. There are seven possible clock options:

- FOSC/2
- FOSC/4
- FOSC/8
- FOSC/16
- FOSC/32
- FOSC/64
- FRC (dedicated internal oscillator)

The time to complete one bit conversion is defined as TAD. One full 10-bit conversion requires 11 TAD periods as shown in Figure 10-3.

For correct conversion, the appropriate TAD specification must be met. See A/D conversion requirements in **Section 16.0 "Electrical Specifications"** for more information. Table 10-1 gives examples of appropriate ADC clock selections.

Note:	Unless using the FRC, any changes in the system clock frequency will change the ADC clock frequency, which may adversely affect the ADC result.

11.0 ENHANCED CAPTURE/ COMPARE/PWM (WITH AUTO-SHUTDOWN AND DEAD BAND) MODULE (PIC12F615/617/ HV615 ONLY)

The Enhanced Capture/Compare/PWM module is a peripheral which allows the user to time and control different events. In Capture mode, the peripheral allows the timing of the duration of an event.The Compare mode allows the user to trigger an external event when a predetermined amount of time has expired. The PWM mode can generate a Pulse-Width Modulated signal of varying frequency and duty cycle.

Table 11-1 shows the timer resources required by the ECCP module.

TABLE 11-1: ECCP MODE – TIMER RESOURCES REQUIRED

ECCP Mode	Timer Resource
Capture	Timer1
Compare	Timer1
PWM	Timer2

REGISTER 11-1: CCP1CON: ENHANCED CCP1 CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
P1M	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
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 **P1M:** PWM Output Configuration bits
If CCP1M<3:2> = 00, 01, 10:
x = P1A assigned as Capture/Compare input; P1B assigned as port pins
If CCP1M<3:2> = 11:
0 = Single output; P1A modulated; P1B assigned as port pins
1 = Half-Bridge output; P1A, P1B modulated with dead-band control

bit 6 **Unimplemented:** Read as ‘0’

bit 5-4 **DC1B<1:0>:** PWM Duty Cycle Least Significant bits
Capture mode:
Unused.
Compare mode:
Unused.
PWM mode:
These bits are the two LSbs of the PWM duty cycle. The eight MSbs are found in CCPR1L.

bit 3-0 **CCP1M<3:0>:** ECCP Mode Select bits
0000 =Capture/Compare/PWM off (resets ECCP module)
0001 =Unused (reserved)
0010 =Compare mode, toggle output on match (CCP1IF bit is set)
0011 =Unused (reserved)
0100 =Capture mode, every falling edge
0101 =Capture mode, every rising edge
0110 =Capture mode, every 4th rising edge
0111 =Capture mode, every 16th rising edge
1000 =Compare mode, set output on match (CCP1IF bit is set)
1001 =Compare mode, clear output on match (CCP1IF bit is set)
1010 =Compare mode, generate software interrupt on match (CCP1IF bit is set, CCP1 pin is unaffected)
1011 =Compare mode, trigger special event (CCP1IF bit is set; CCP1 resets TMR1 or TMR2 and starts an A/D conversion, if the ADC module is enabled)
1100 =PWM mode; P1A active-high; P1B active-high
1101 =PWM mode; P1A active-high; P1B active-low
1110 =PWM mode; P1A active-low; P1B active-high
1111 =PWM mode; P1A active-low; P1B active-low

11.3.4 OPERATION IN SLEEP MODE

In Sleep mode, the TMR2 register will not increment and the state of the module will not change. If the CCP1 pin is driving a value, it will continue to drive that value. When the device wakes up, TMR2 will continue from its previous state.

11.3.5 CHANGES IN SYSTEM CLOCK FREQUENCY

The PWM frequency is derived from the system clock frequency. Any changes in the system clock frequency will result in changes to the PWM frequency. See **Section 4.0 "Oscillator Module"** for additional details.

11.3.6 EFFECTS OF RESET

Any Reset will force all ports to Input mode and the CCP registers to their Reset states.

11.3.7 SETUP FOR PWM OPERATION

The following steps should be taken when configuring the CCP module for PWM operation:

1. Disable the PWM pin (CCP1) output drivers by setting the associated TRIS bit.
2. Set the PWM period by loading the PR2 register.
3. Configure the CCP module for the PWM mode by loading the CCP1CON register with the appropriate values.
4. Set the PWM duty cycle by loading the CCPR1L register and DC1B bits of the CCP1CON register.
5. Configure and start Timer2:
   - Clear the TMR2IF interrupt flag bit of the PIR1 register.
   - Set the Timer2 prescale value by loading the T2CKPS bits of the T2CON register.
   - Enable Timer2 by setting the TMR2ON bit of the T2CON register.
6. Enable PWM output after a new PWM cycle has started:
   - Wait until Timer2 overflows (TMR2IF bit of the PIR1 register is set).
   - Enable the CCP1 pin output driver by clearing the associated TRIS bit.

REGISTER 12-1: CONFIG: CONFIGURATION WORD REGISTER (ADDRESS: 2007h) FOR PIC12F609/615/HV609/615 ONLY

U-1	U-1	U-1	U-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	—	BOREN1(1)	BOREN0(1)	IOSCFS	$\overline{CP}$(2)	MCLRE(3)	$\overline{PWRTE}$	WDTE	FOSC2	FOSC1	FOSC0
bit 13													bit 0

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

bit 13-10 **Unimplemented**: Read as ‘1’

bit 9-8 **BOREN<1:0>:** Brown-out Reset Selection bits(1)
11 = BOR enabled
10 = BOR enabled during operation and disabled in Sleep
0x = BOR disabled

bit 7 **IOSCFS:** Internal Oscillator Frequency Select bit
1 = 8 MHz
0 = 4 MHz

bit 6 **$\overline{CP}$:** Code Protection bit(2)
1 = Program memory code protection is disabled
0 = Program memory code protection is enabled

bit 5 **MCLRE:** $\overline{MCLR}$ Pin Function Select bit(3)
1 = $\overline{MCLR}$ pin function is $\overline{MCLR}$
0 = $\overline{MCLR}$ pin function is digital input, $\overline{MCLR}$ internally tied to V_{DD}

bit 4 **$\overline{PWRTE}$:** Power-up Timer Enable bit
1 = PWRT disabled
0 = PWRT enabled

bit 3 **WDTE:** Watchdog Timer Enable bit
1 = WDT enabled
0 = WDT disabled

bit 2-0 **FOSC<2:0>:** Oscillator Selection bits
111 =RC oscillator: CLKOUT function on GP4/OSC2/CLKOUT pin, RC on GP5/OSC1/CLKIN
110 =RCIO oscillator: I/O function on GP4/OSC2/CLKOUT pin, RC on GP5/OSC1/CLKIN
101 =INTOSC oscillator: CLKOUT function on GP4/OSC2/CLKOUT pin, I/O function on GP5/OSC1/CLKIN
100 = INTOSCIO oscillator: I/O function on GP4/OSC2/CLKOUT pin, I/O function on GP5/OSC1/CLKIN
011 =EC: I/O function on GP4/OSC2/CLKOUT pin, CLKIN on GP5/OSC1/CLKIN
010 =HS oscillator: High-speed crystal/resonator on GP4/OSC2/CLKOUT and GP5/OSC1/CLKIN
001 = XT oscillator: Crystal/resonator on GP4/OSC2/CLKOUT and GP5/OSC1/CLKIN
000 = LP oscillator: Low-power crystal on GP4/OSC2/CLKOUT and GP5/OSC1/CLKIN

Note 1: Enabling Brown-out Reset does not automatically enable Power-up Timer.
2: The entire program memory will be erased when the code protection is turned off.
3: When $\overline{MCLR}$ is asserted in INTOSC or RC mode, the internal clock oscillator is disabled.

12.4.2 TIMER0 INTERRUPT

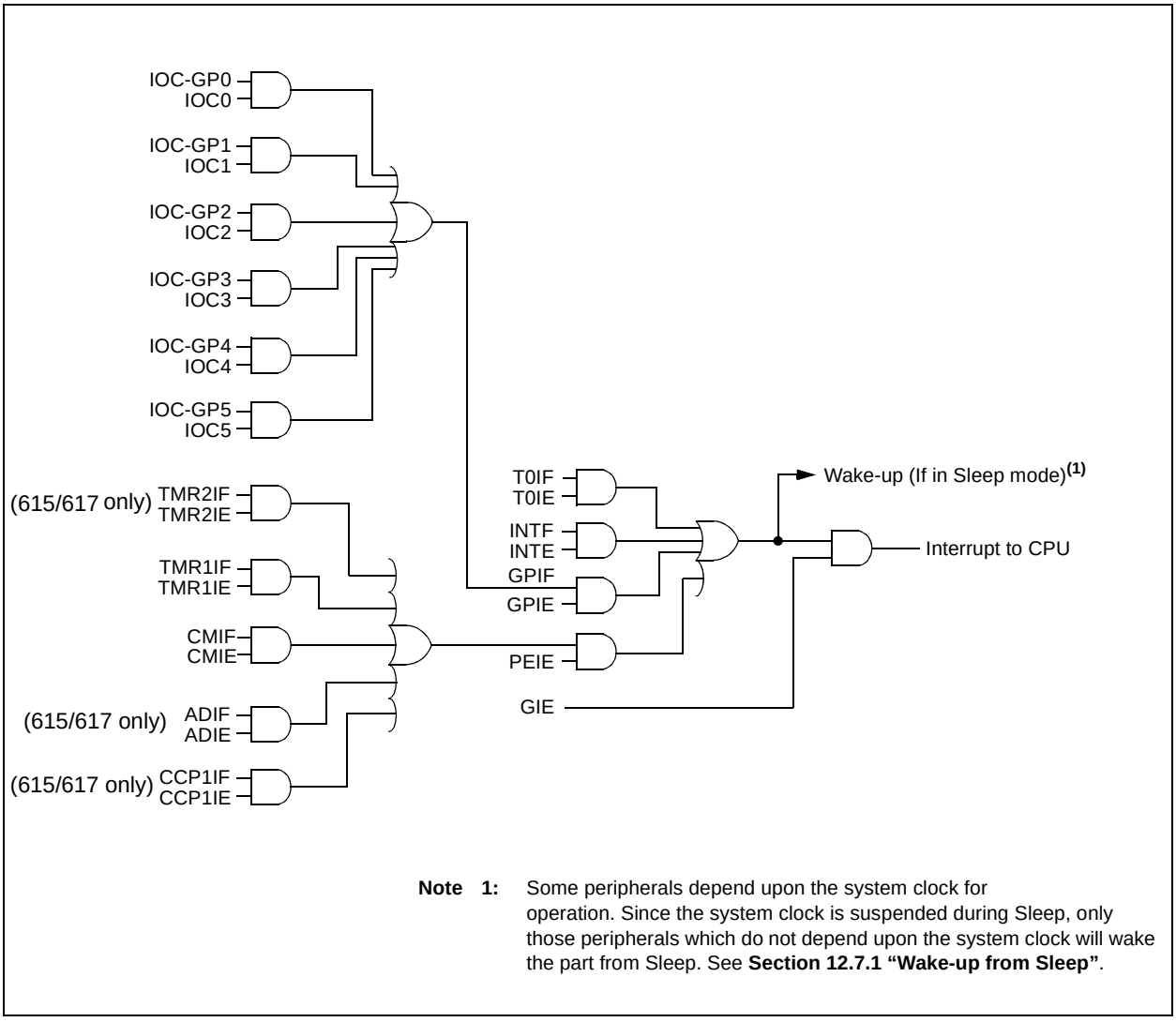
An overflow (FFh → 00h) in the TMR0 register will set the T0IF bit of the INTCON register. The interrupt can be enabled/disabled by setting/clearing T0IE bit of the INTCON register. See **Section 6.0 "Timer0 Module"** for operation of the Timer0 module.

12.4.3 GPIO INTERRUPT-ON-CHANGE

An input change on GPIO sets the GPIF bit of the INTCON register. The interrupt can be enabled/disabled by setting/clearing the GPIE bit of the INTCON register. Plus, individual pins can be configured through the IOC register.

Note:	If a change on the I/O pin should occur when any GPIO operation is being executed, then the GPIF interrupt flag may not get set.

FIGURE 12-7: INTERRUPT LOGIC

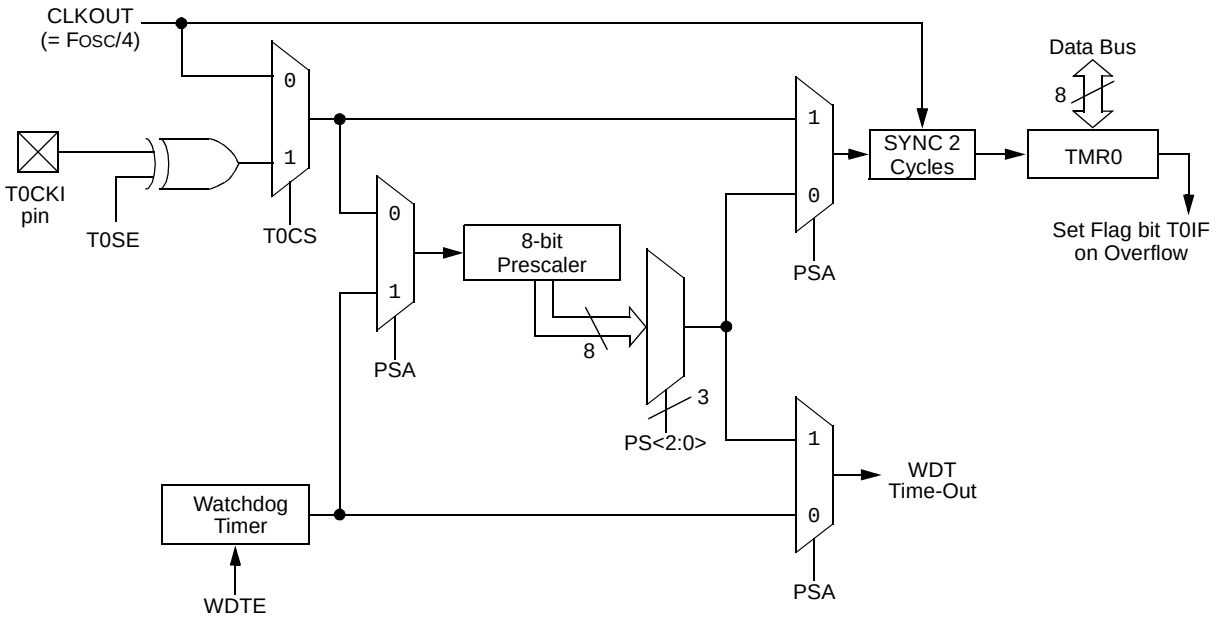


Note 1: Some peripherals depend upon the system clock for operation. Since the system clock is suspended during Sleep, only those peripherals which do not depend upon the system clock will wake the part from Sleep. See **Section 12.7.1 "Wake-up from Sleep"**.

12.6.2 WDT PROGRAMMING CONSIDERATIONS

It should also be taken in account that under worst-case conditions (i.e., VDD = Min., Temperature = Max., Max. WDT prescaler) it may take several seconds before a WDT time out occurs.

FIGURE 12-2: WATCHDOG TIMER BLOCK DIAGRAM



Note 1: T0SE, T0CS, PSA, PS<2:0> are bits in the OPTION register.

TABLE 12-8: WDT STATUS

Conditions	WDT
WDTE = 0	Cleared
CLRWDT Command	
Oscillator Fail Detected	
Exit Sleep + System Clock = T1OSC, EXTRC, INTRC, EXTCLK	
Exit Sleep + System Clock = XT, HS, LP	Cleared until the end of OST

TABLE 12-9: SUMMARY OF REGISTERS ASSOCIATED WITH WATCHDOG TIMER

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
OPTION_REG	$\overline{GPPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
CONFIG	IOSCFS	$\overline{CP}$	MCLRE	$\overline{PWRTE}$	WDTE	FOSC2	FOSC1	FOSC0	—	—

Legend: Shaded cells are not used by the Watchdog Timer.

Note 1: See Register 12-1 for operation of all Configuration Word register bits.

DECFSZ **Decrement f, Skip if 0**

Syntax: [*label*] DECFSZ f,d

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

Operation: (f) - 1 → (destination);
skip if result = 0

Status Affected: None

Description: The contents of register 'f' are decremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.
If the result is '1', the next instruction is executed. If the result is '0', then a NOP is executed instead, making it a two-cycle instruction.

GOTO **Unconditional Branch**

Syntax: [*label*] GOTO k

Operands: $0 \le k \le 2047$

Operation: k → PC<10:0>
PCLATH<4:3> → PC<12:11>

Status Affected: None

Description: GOTO is an unconditional branch. The eleven-bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a two-cycle instruction.

INCF **Increment f**

Syntax: [*label*] INCF f,d

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

Operation: (f) + 1 → (destination)

Status Affected: Z

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

INCFSZ **Increment f, Skip if 0**

Syntax: [*label*] INCFSZ f,d

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

Operation: (f) + 1 → (destination),
skip if result = 0

Status Affected: None

Description: The contents of register 'f' are incremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.
If the result is '1', the next instruction is executed. If the result is '0', a NOP is executed instead, making it a two-cycle instruction.

IORLW **Inclusive OR literal with W**

Syntax: [*label*] IORLW k

Operands: $0 \le k \le 255$

Operation: (W) .OR. k → (W)

Status Affected: Z

Description: The contents of the W register are OR'ed with the eight-bit literal 'k'. The result is placed in the W register.

IORWF **Inclusive OR W with f**

Syntax: [*label*] IORWF f,d

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

Operation: (W) .OR. (f) → (destination)

Status Affected: Z

Description: Inclusive OR the W register with register 'f'. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

RETFIE	Return from Interrupt
Syntax:	[*label*] RETFIE
Operands:	None
Operation:	TOS → PC, 1 → GIE
Status Affected:	None
Description:	Return from Interrupt. Stack is POPed and Top-of-Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two-cycle instruction.
Words:	1
Cycles:	2
Example:	`RETFIE` After Interrupt PC = TOS GIE = 1

RETLW	Return with literal in W
Syntax:	[*label*] RETLW k
Operands:	0 ≤ k ≤ 255
Operation:	k → (W); TOS → PC
Status Affected:	None
Description:	The W register is loaded with the eight-bit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.
Words:	1
Cycles:	2

Example:	`CALL TABLE;W contains`
	`;table offset`
	`;value`
	`GOTO DONE`
	`TABLE`
	•
	•
	`ADDWF PC ;W = offset`
	`RETLW k1 ;Begin table`
	`RETLW k2 ;`
	•
	•
	•
	`RETLW kn ;End of table`
	`DONE`
	Before Instruction
	W = 0x07
	After Instruction
	W = value of k8

RETURN	Return from Subroutine
Syntax:	[*label*] RETURN
Operands:	None
Operation:	TOS → PC
Status Affected:	None
Description:	Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two-cycle instruction.

16.7 DC Characteristics: PIC12HV609/615-E (Extended)

DC CHARACTERISTICS		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C for extended					
Param No.	Device Characteristics	Min	Typ†	Max	Units	Conditions	
						VDD	Note
D020E	**Power-down Base Current (IPD)(2,3)** PIC12HV609/615	—	135	200	µA	2.0	WDT, BOR, Comparator, VREF and T1OSC disabled
		—	210	280	µA	3.0	
		—	260	350	µA	4.5	
D021E		—	135	200	µA	2.0	WDT Current(1)
		—	210	285	µA	3.0	
		—	265	360	µA	4.5	
D022E		—	215	285	µA	3.0	BOR Current(1)
		—	265	360	µA	4.5	
D023E		—	185	280	µA	2.0	Comparator Current(1), single comparator enabled
		—	265	360	µA	3.0	
		—	320	430	µA	4.5	
D024E		—	165	235	µA	2.0	CVREF Current(1) (high range)
		—	255	330	µA	3.0	
		—	330	430	µA	4.5	
D025E*		—	175	245	µA	2.0	CVREF Current(1) (low range)
		—	275	350	µA	3.0	
		—	355	450	µA	4.5	
D026E		—	140	205	µA	2.0	T1OSC Current(1), 32.768 kHz
		—	220	290	µA	3.0	
		—	270	360	µA	4.5	
D027E		—	210	280	µA	3.0	A/D Current(1), no conversion in progress
		—	260	350	µA	4.5	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 4.5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The peripheral current is the sum of the base IDD or IPD and the additional current consumed when this peripheral is enabled. The peripheral Δ current can be determined by subtracting the base IDD or IPD current from this limit. Max values should be used when calculating total current consumption.

2: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD.

3: Shunt regulator is always on and always draws operating current.

FIGURE 17-6: PIC12F609/615/617 IDD INTOSC (4 MHz) vs. VDD

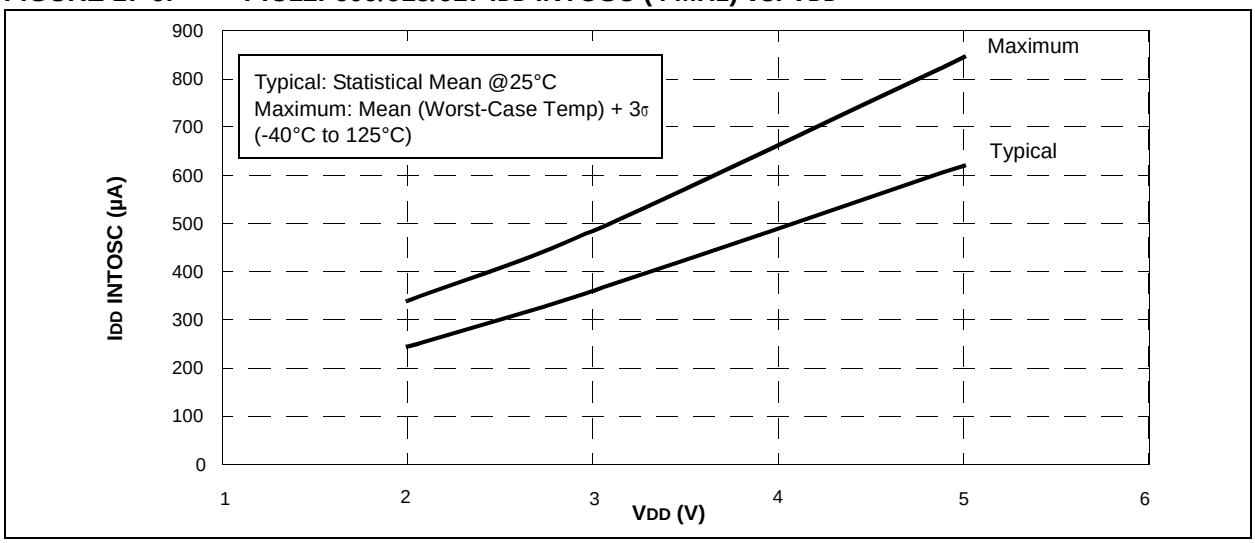


FIGURE 17-7: PIC12F609/615/617 IDD INTOSC (8 MHz) vs. VDD

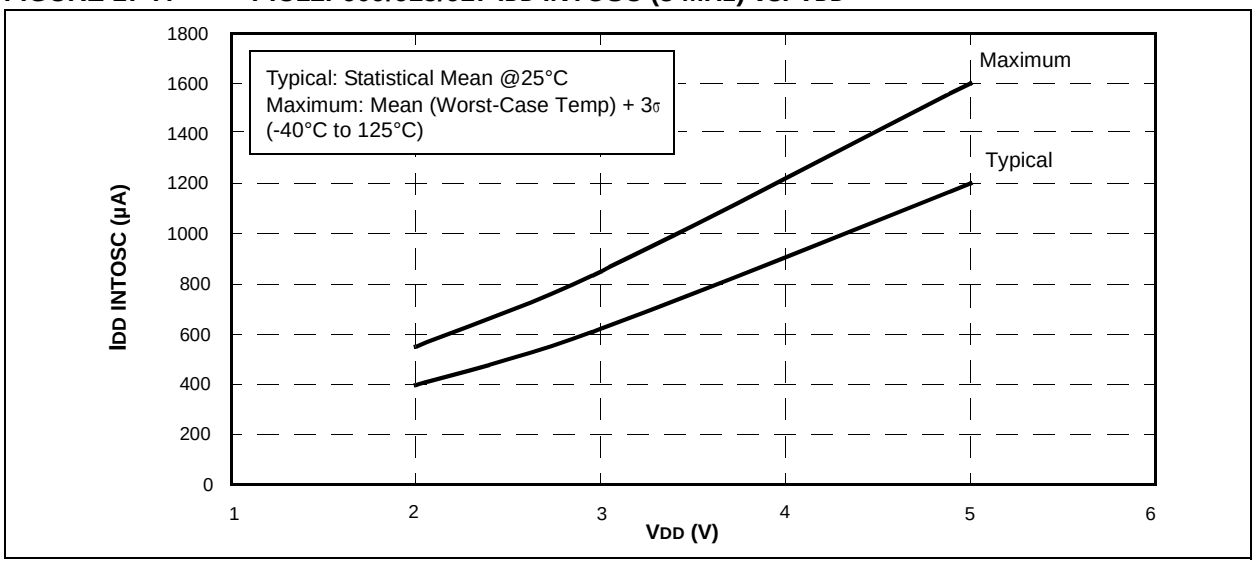


FIGURE 17-12: PIC12F609/615/617 IPD WDT vs. VDD

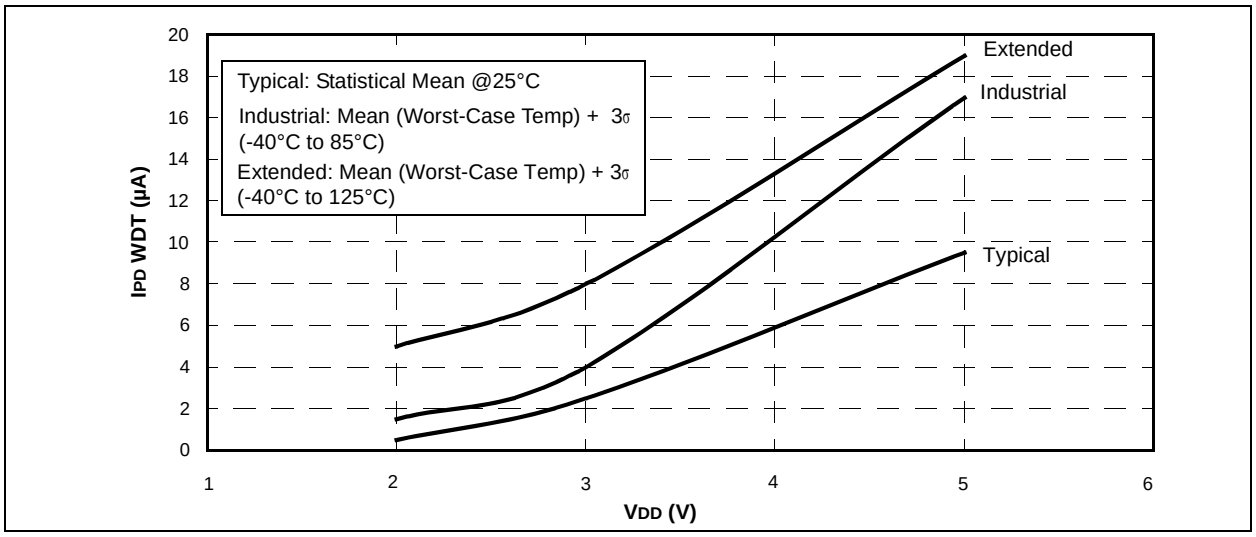


FIGURE 17-13: PIC12F609/615/617 IPD BOR vs. VDD

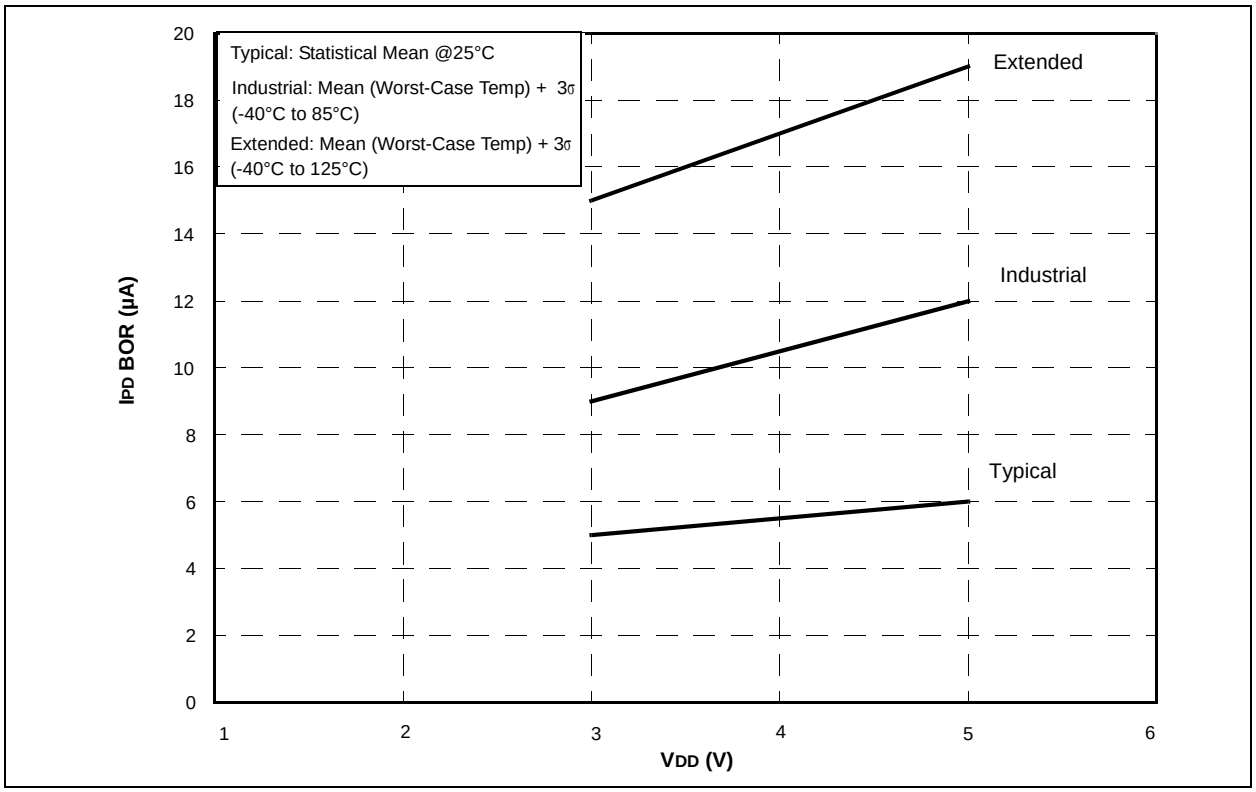


FIGURE 17-18: PIC12HV609/615 IDD LP (32 kHz) vs. VDD

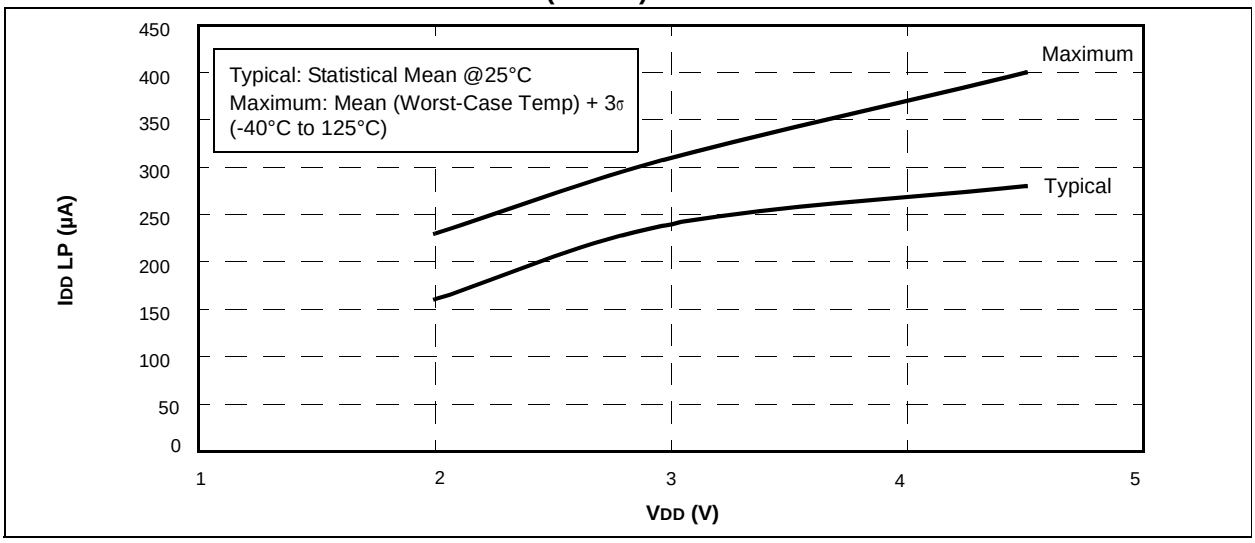


FIGURE 17-19: PIC12HV609/615 IDD EC (1 MHz) vs. VDD

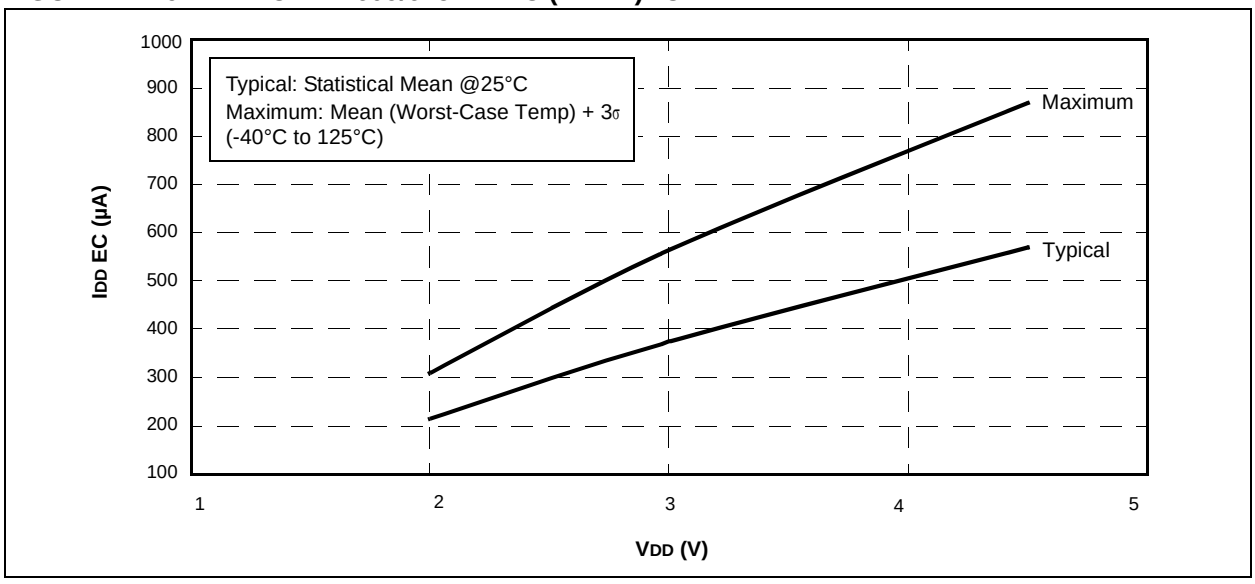


FIGURE 17-20: PIC12HV609/615 IDD EC (4 MHz) vs. VDD

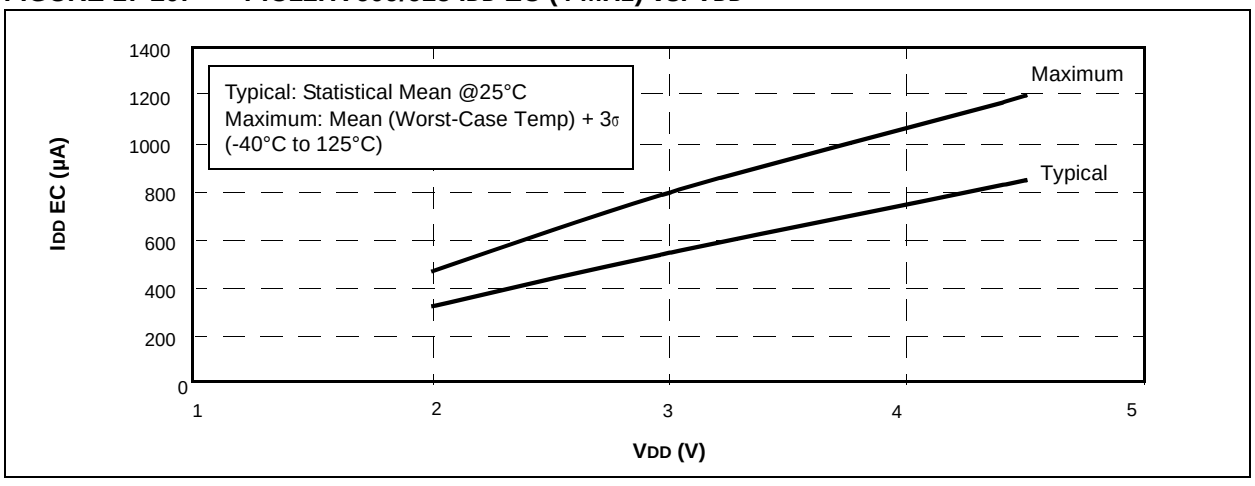


FIGURE 17-35: VOL vs. IOL OVER TEMPERATURE (VDD = 5.0V)

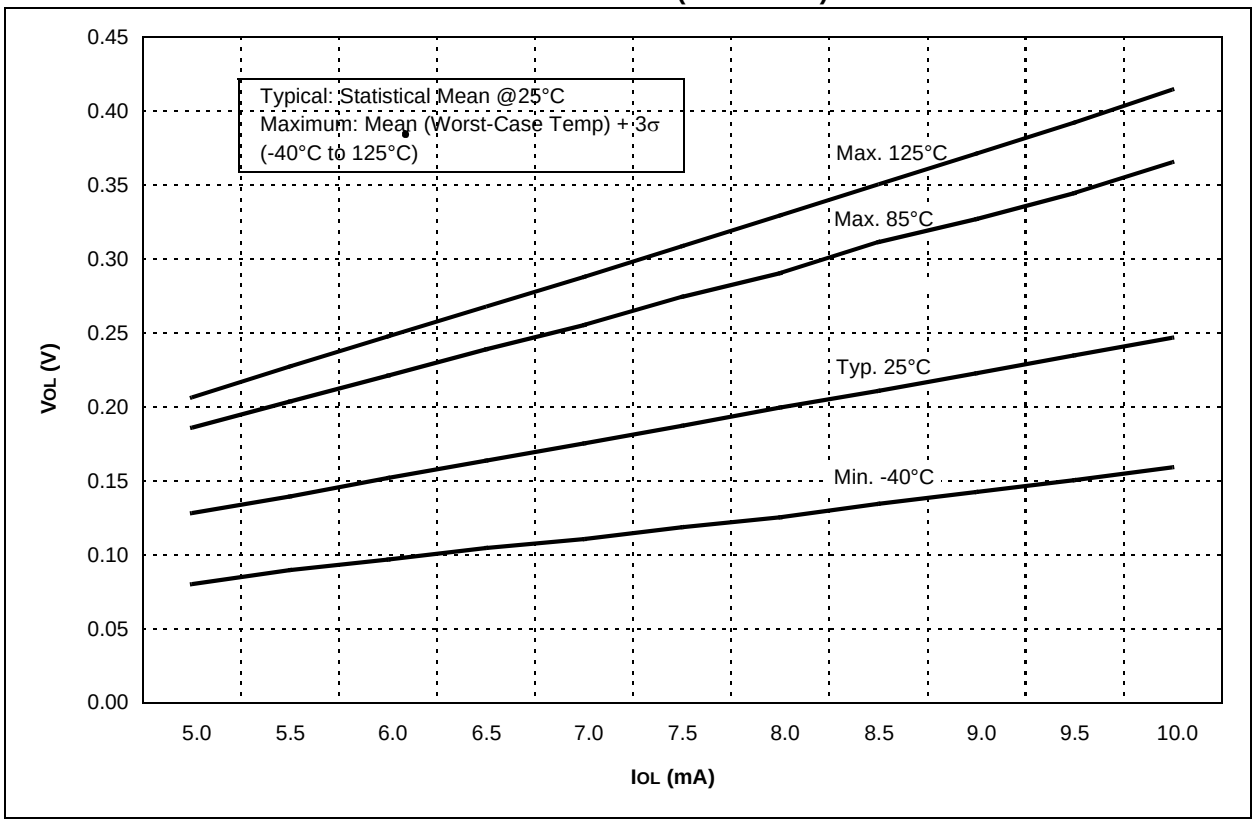
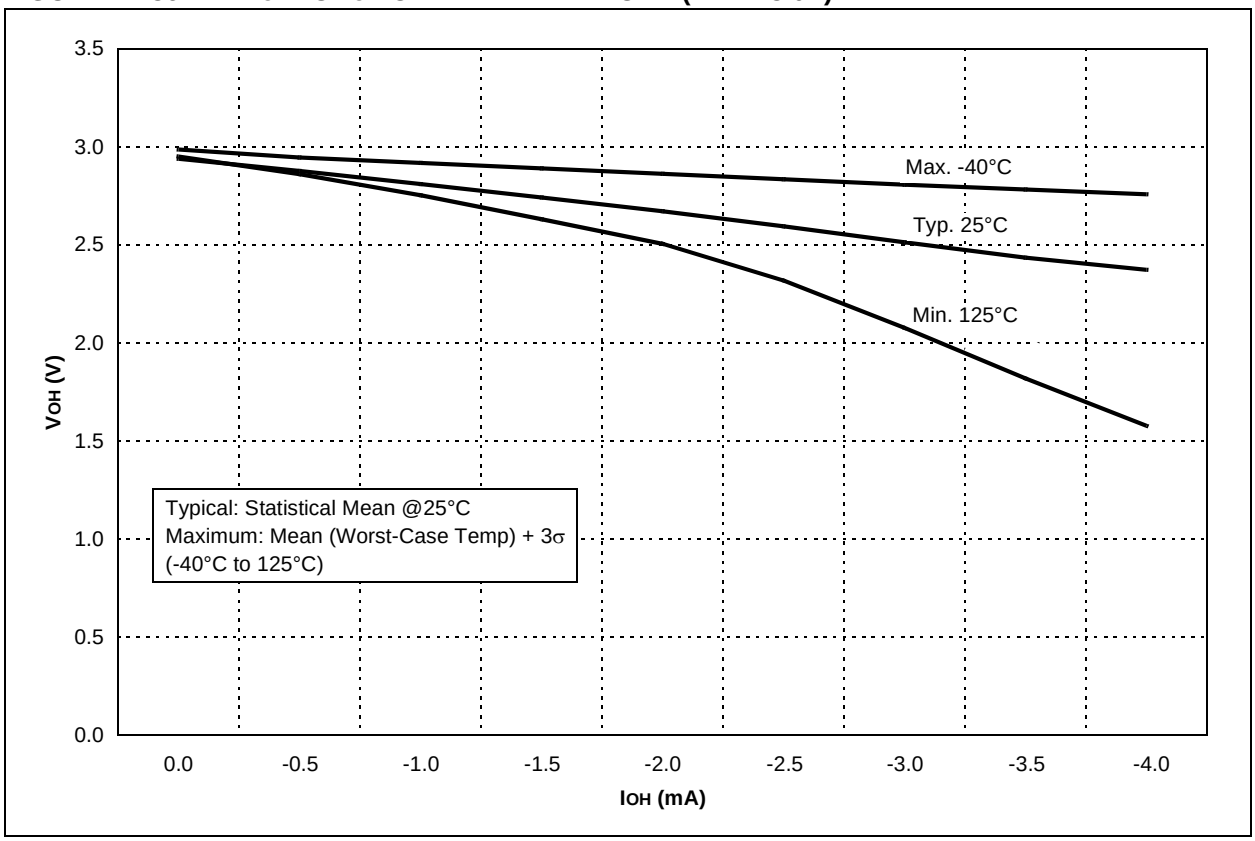
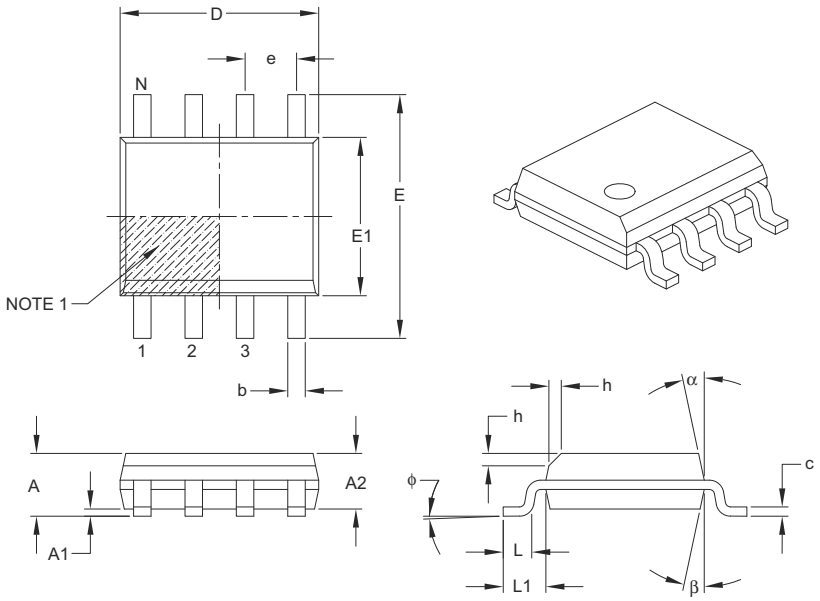


FIGURE 17-36: VOH vs. IOH OVER TEMPERATURE (VDD = 3.0V)



8-Lead Plastic Small Outline (SN) – Narrow, 3.90 mm Body [SOIC]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		8	
Pitch	e		1.27 BSC	
Overall Height	A	–	–	1.75
Molded Package Thickness	A2	1.25	–	–
Standoff §	A1	0.10	–	0.25
Overall Width	E		6.00 BSC	
Molded Package Width	E1		3.90 BSC	
Overall Length	D		4.90 BSC	
Chamfer (optional)	h	0.25	–	0.50
Foot Length	L	0.40	–	1.27
Footprint	L1		1.04 REF	
Foot Angle	φ	0°	–	8°
Lead Thickness	c	0.17	–	0.25
Lead Width	b	0.31	–	0.51
Mold Draft Angle Top	α	5°	–	15°
Mold Draft Angle Bottom	β	5°	–	15°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15 mm per side.
4. 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 C04-057B

Initializing GPIO 43
Saving Status and W Registers in RAM 121
Writing to Flash Program Memory 34
Code Protection 124
Comparator 67
Associated registers 78
Control 69
Gating Timer1 73
Operation During Sleep 71
Overview 67
Response Time 69
Synchronizing COUT w/Timer1 73
Comparator Hysteresis 77
Comparator Voltage Reference (CVREF) 74
Effects of a Reset 71
Comparator Voltage Reference (CVREF)
Response Time 69
Comparator Voltage Reference (CVREF)
Specifications 163
Comparators
C2OUT as T1 Gate 60
Effects of a Reset 71
Specifications 162
Compare Module. *See* Enhanced Capture/Compare/
PWM (ECCP) (PIC12F615/617/HV615 only)
CONFIG Register 108
Configuration Bits 107
CPU Features 107
Customer Change Notification Service 209
Customer Notification Service 209
Customer Support 209

D

Data EEPROM Memory
Associated Registers 35
Data Memory 11
DC and AC Characteristics
Graphs and Tables 171
DC Characteristics
Extended and Industrial 152
Industrial and Extended 145
Development Support 139
Device Overview 7

E

ECCP. *See* Enhanced Capture/Compare/PWM
ECCPAS Register 102
EEDAT Register 28
EEDATH Register 28
Effects of Reset
PWM mode 96
Electrical Specifications 143
Enhanced Capture/Compare/PWM (ECCP)
Enhanced PWM Mode 97
Auto-Restart 103
Auto-shutdown 101
Half-Bridge Application 99
Half-Bridge Application Examples 104
Half-Bridge Mode 99
Output Relationships (Active-High and
Active-Low) 98
Output Relationships Diagram 98
Programmable Dead Band Delay 104
Shoot-through Current 104
Start-up Considerations 100
Specifications 162
Timer Resources 89
Enhanced Capture/Compare/PWM
(PIC12F615/617/HV615 Only) 89
Errata 6

F

Firmware Instructions 129
Flash Program Memory Self Read/Self Write
Control (For PIC12F617 only) 27
Fuses. *See* Configuration Bits

G

General Purpose Register File 12
GPIO 43
Additional Pin Functions 44
ANSEL Register 44
Interrupt-on-Change 44
Weak Pull-Ups 44
Associated registers 52
GP0 47
GP1 47
GP2 48
GP3 49
GP4 50
GP5 51
Pin Descriptions and Diagrams 47
Specifications 158
GPIO Register 43

H

High Temperature Operation 167

I

ID Locations 124
In-Circuit Debugger 125
In-Circuit Serial Programming (ICSP) 125
Indirect Addressing, INDF and FSR registers 25
Instruction Format 129
Instruction Set 129
ADDLW 131
ADDWF 131
ANDLW 131
ANDWF 131
MOVF 134
BCF 131
BSF 131
BTFSC 131
BTFSS 132
CALL 132
CLRF 132
CLRW 132
CLRWDT 132
COMF 132
DECF 132
DECFSZ 133
GOTO 133
INCF 133
INCFSZ 133
IORLW 133
IORWF 133
MOVLW 134
MOVWF 134
NOP 134
RETFIE 135
RETLW 135
RETURN 135