

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	13
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 4x8b
Oscillator Type	External
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	18-SOIC (0.295", 7.50mm Width)
Supplier Device Package	18-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f716-e-so

PIC16F716

2.2.2.2 OPTION Register

The OPTION register is a readable and writable register, which contains various control bits to configure the TMR0 prescaler/WDT postscaler (single assignable register known also as the prescaler), the External INT Interrupt, TMR0 and the weak pull-ups on PORTB.

Note: To achieve a 1:1 prescaler assignment for the Timer0 register, assign the prescaler to the Watchdog Timer.

REGISTER 2-2: 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
RBPU	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 **RBPU:** PORTB Pull-up Enable bit
 1 = PORTB pull-ups are disabled
 0 = PORTB pull-ups are enabled by individual PORT latch values
- bit 6 **INTEDG:** Interrupt Edge Select bit
 1 = Interrupt on rising edge of RB0/INT pin
 0 = Interrupt on falling edge of RB0/INT pin
- bit 5 **T0CS:** Timer0 Clock Source Select bit
 1 = Transition on RA4/T0CKI pin
 0 = Internal instruction cycle clock (Fosc/4)
- bit 4 **T0SE:** Timer0 Source Edge Select bit
 1 = Increment on high-to-low transition on RA4/T0CKI pin
 0 = Increment on low-to-high transition on RA4/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	Timer0 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

PIC16F716

2.2.2.4 PIE1 Register

This register contains the individual enable bits for the peripheral interrupts.

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

REGISTER 2-4: PIE1: PERIPHERAL INTERRUPT ENABLE REGISTER 1

U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE
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 **Unimplemented:** Read as '0'
- bit 6 **ADIE:** A/D Converter (ADC) Interrupt Enable bit
1 = Enables the ADC interrupt
0 = Disables the ADC interrupt
- bit 5-3 **Unimplemented:** Read as '0'
- bit 2 **CCP1IE:** CCP1 Interrupt Enable bit
1 = Enables the CCP1 interrupt
0 = Disables the CCP1 interrupt
- bit 1 **TMR2IE:** Timer2 to PR2 Match Interrupt Enable bit
1 = Enables the Timer2 to PR2 match interrupt
0 = Disables the Timer2 to PR2 match interrupt
- bit 0 **TMR1IE:** Timer1 Overflow Interrupt Enable bit
1 = Enables the Timer1 overflow interrupt
0 = Disables the Timer1 overflow interrupt

2.2.2.5 PIR1 Register

This register contains the individual flag bits for the peripheral interrupts.

Note: Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-5: PIR1: PERIPHERAL INTERRUPT REQUEST REGISTER 1

U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF
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 **Unimplemented:** Read as '0'
- bit 6 **ADIF:** A/D Interrupt Flag bit
 1 = A/D conversion complete
 0 = A/D conversion has not completed or has not been started
- bit 5-3 **Unimplemented:** Read as '0'
- bit 2 **CCP1IF:** CCP1 Interrupt Flag bit
 Capture Mode
 1 = A TMR1 register capture occurred (must be cleared in software)
 0 = No TMR1 register capture occurred
 Compare Mode
 1 = A TMR1 register compare match occurred (must be cleared in software)
 0 = No TMR1 register compare match occurred
 PWM Mode
 Unused in this mode
- bit 1 **TMR2IF:** Timer2 to PR2 Match Interrupt Flag bit
 1 = Timer2 to PR2 match occurred (must be cleared in software)
 0 = Timer2 to PR2 match has not occurred
- bit 0 **TMR1IF:** Timer1 Overflow Interrupt Flag bit
 1 = Timer1 register overflowed (must be cleared in software)
 0 = Timer1 has not overflowed

PIC16F716

FIGURE 3-4: BLOCK DIAGRAM OF RB1/T1OSO/T1CKI PIN

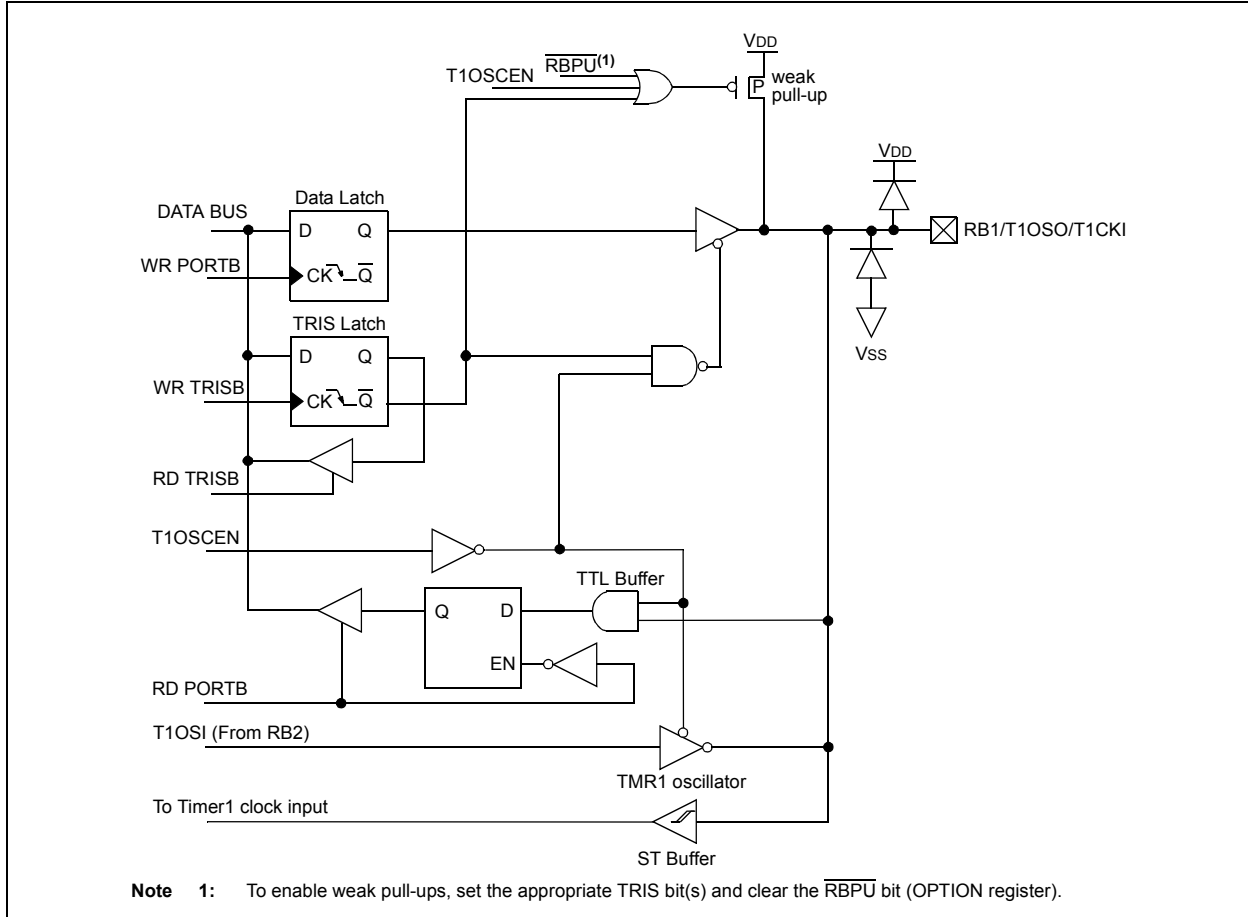


FIGURE 3-5: BLOCK DIAGRAM OF RB2/T1OSI PIN

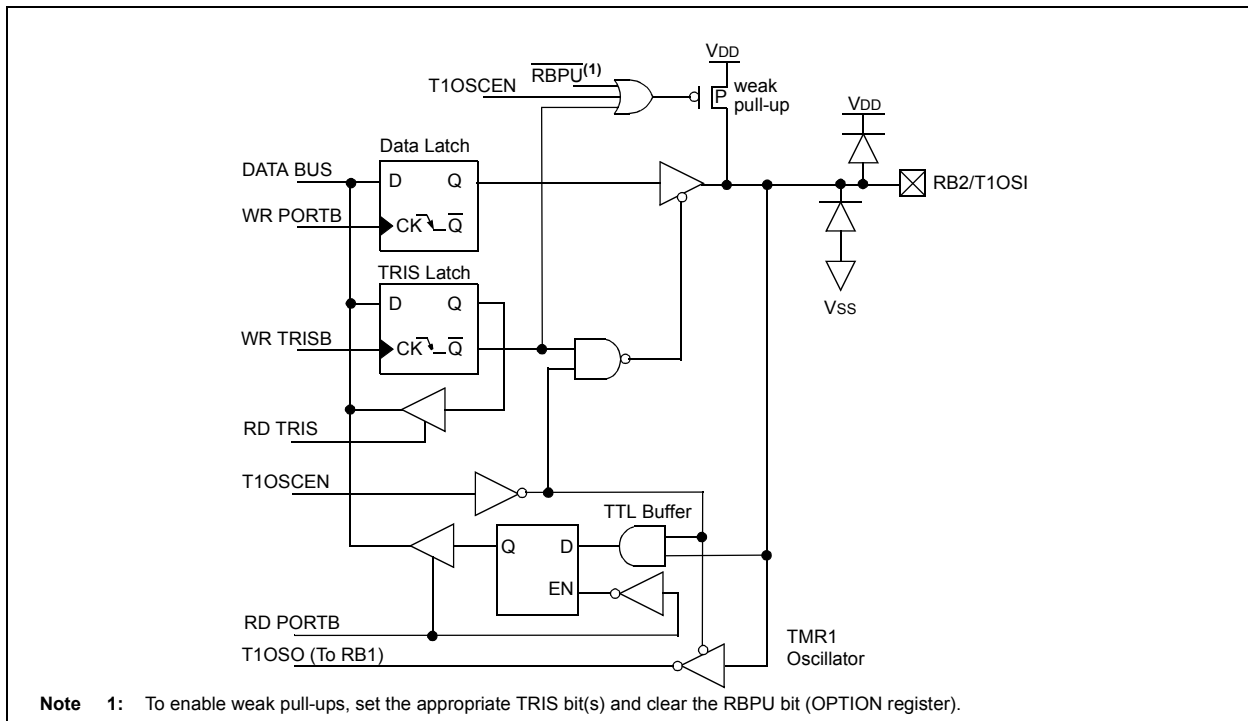


TABLE 5-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER1

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
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	-0-- -000
PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- -000	-0-- -000
TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	--00 0000	--uu uuuu

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

TABLE 8-2: REGISTERS ASSOCIATED WITH CAPTURE

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
CCPR1L	Capture/Compare/PWM Register 1 (LSB)								xxxx xxxx	xxxx xxxx
CCPR1H	Capture/Compare/PWM Register 1 (MSB)								xxxx xxxx	xxxx xxxx
CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	0000 0000
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	-0-- -000
PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- -000	-0-- -000
PR2	Timer2 Period Register								1111 1111	1111 1111
TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	xxxx xxxx
TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	xxxx xxxx
TMR2	Timer2 module's register								0000 0000	0000 0000
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111

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

PIC16F716

8.2 Compare Mode

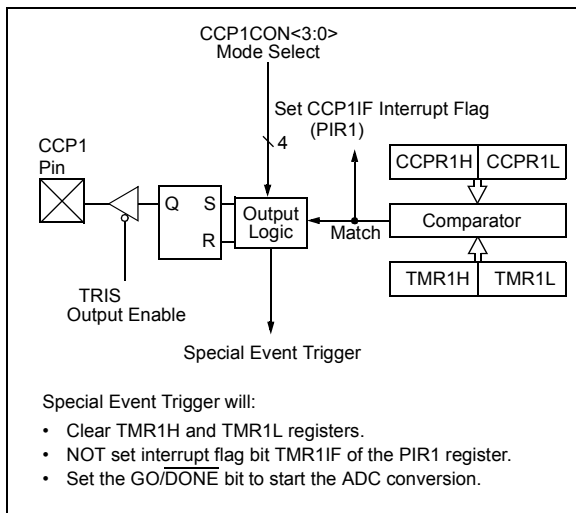
In Compare mode, the 16-bit CCPR1 register value is constantly compared against the TMR1 register pair value. When a match occurs, the CCP1 module may:

- Toggle the CCP1 output.
- Set the CCP1 output.
- Clear the CCP1 output.
- Generate a Special Event Trigger.
- Generate a Software Interrupt.

The action on the pin is based on the value of the CCP1M<3:0> control bits of the CCP1CON register.

All Compare modes can generate an interrupt.

FIGURE 8-2: COMPARE MODE OPERATION BLOCK DIAGRAM



8.2.1 CCP1 PIN CONFIGURATION

The user must configure the CCP1 pin as an output by clearing the associated TRIS bit.

Note: Clearing the CCP1CON register will force the CCP1 compare output latch to the default low level. This is not the PORT I/O data latch.

8.2.2 TIMER1 MODE SELECTION

In Compare mode, Timer1 must be running in either Timer mode or Synchronized Counter mode. The compare operation may not work in Asynchronous Counter mode.

8.2.3 SOFTWARE INTERRUPT MODE

When Generate Software Interrupt mode is chosen (CCP1M<3:0> = 1010), the CCP1 module does not assert control of the CCP1 pin (see the CCP1CON register).

8.2.4 SPECIAL EVENT TRIGGER

When Special Event Trigger mode is chosen (CCP1M<3:0> = 1011), the CCP1 module does the following:

- Resets Timer1
- Starts an ADC conversion if ADC is enabled

The CCP1 module does not assert control of the CCP1 pin in this mode (see the CCP1CON register).

The Special Event Trigger output of the CCP occurs immediately upon a match between the TMR1H, TMR1L register pair and the CCPR1H, CCPR1L register pair. The TMR1H, TMR1L register pair is not reset until the next rising edge of the Timer1 clock. This allows the CCPR1H, CCPR1L register pair to effectively provide a 16-bit programmable period register for Timer1.

Note 1: The Special Event Trigger from the CCP module does not set interrupt flag bit TMRxIF of the PIR1 register.

- 2: Removing the match condition by changing the contents of the CCPR1H and CCPR1L register pair, between the clock edge that generates the Special Event Trigger and the clock edge that generates the Timer1 Reset, will preclude the Reset from occurring.

8.3.1 PWM PERIOD

The PWM period is specified by the PR2 register of Timer2. The PWM period can be calculated using the formula of Equation 8-1.

EQUATION 8-1: PWM PERIOD

$$PWM\ Period = [(PR2) + 1] \cdot 4 \cdot TOSC \cdot (TMR2\ Prescale\ Value)$$

When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The CCP1 pin is set. (Exception: If the PWM duty cycle = 0%, the pin will not be set.)
- The PWM duty cycle is latched from CCPR1L into CCPR1H.

Note: The Timer2 postscaler (see **Section 6.0 “Timer2 Module”**) is not used in the determination of the PWM frequency.

8.3.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing a 10-bit value to multiple registers: CCPR1L register and DC1B<1:0> bits of the CCP1CON register. The CCPR1L contains the eight MSBs and the DC1B<1:0> bits of the CCP1CON register contain the two LSBs. CCPR1L and DC1B<1:0> bits of the CCP1CON register can be written to at any time. The duty cycle value is not latched into CCPR1H until after the period completes (i.e., a match between PR2 and TMR2 registers occurs). While using the PWM, the CCPR1H register is read-only.

Equation 8-2 is used to calculate the PWM pulse width.

Equation 8-3 is used to calculate the PWM duty cycle ratio.

EQUATION 8-2: PULSE WIDTH

$$Pulse\ Width = (CCPR1L:CCP1CON<5:4>) \cdot TOSC \cdot (TMR2\ Prescale\ Value)$$

EQUATION 8-3: DUTY CYCLE RATIO

$$Duty\ Cycle\ Ratio = \frac{(CCPR1L:CCP1CON<5:4>)}{4(PR2 + 1)}$$

The CCPR1H register and a 2-bit internal latch are used to double buffer the PWM duty cycle. This double buffering is essential for glitchless PWM operation.

The 8-bit timer TMR2 register is concatenated with either the 2-bit internal system clock (FOSC), or 2 bits of the prescaler, to create the 10-bit time base. The system clock is used if the Timer2 prescaler is set to 1:1.

When the 10-bit time base matches the CCPR1H and 2-bit latch, then the CCP1 pin is cleared (see Figure 8-3).

9.0 SPECIAL FEATURES OF THE CPU

The PIC16F716 device has a host of features intended to maximize system reliability, minimize cost through elimination of external components, provide power-saving operating modes and offer code protection. These are:

- OSC Selection
- Reset
 - Power-on Reset (POR)
 - Power-up Timer (PWRT)
 - Oscillator Start-up Timer (OST)
 - Brown-out Reset (BOR)
- Interrupts
- Watchdog Timer (WDT)
- Sleep
- Code protection
- ID locations
- In-Circuit Serial Programming™ (ICSP™)

The PIC16F716 device has a Watchdog Timer, which can be shut off only through Configuration bits. It runs off its own RC oscillator for added reliability. There are two timers that offer necessary delays on power-up. One is the Oscillator Start-up Timer (OST), intended to keep the chip in Reset until the crystal oscillator is stable. The other is the Power-up Timer (PWRT), which provides a fixed delay on power-up only and is designed to keep the part in Reset while the power supply stabilizes. With these two timers on-chip, most applications need no external Reset circuitry.

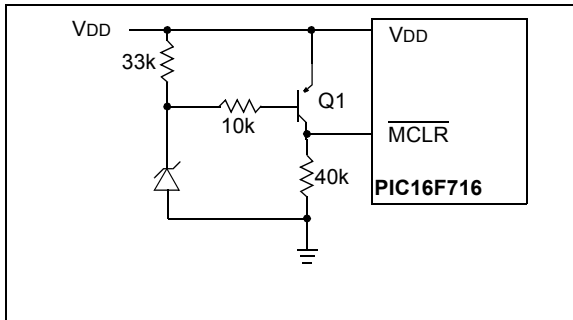
Sleep mode is designed to offer a very low-current Power-Down mode. The user can wake-up from Sleep through external Reset, Watchdog Timer Wake-up, or through an interrupt. Several oscillator options are also made available to allow the part to fit the application. The RC oscillator option saves system cost, while the LP crystal option saves power. A set of Configuration bits are used to select various options.

9.1 Configuration Bits

The Configuration bits can be programmed (read as '0') or left unprogrammed (read as '1') to select various device configurations. These bits are mapped in program memory location 2007h.

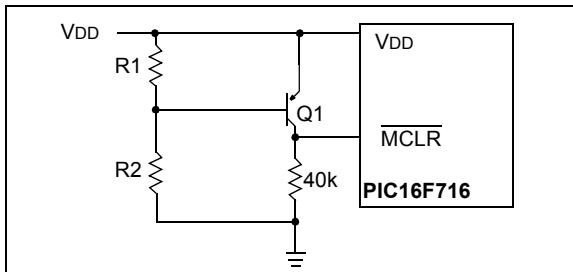
The user will note that address 2007h is beyond the user program memory space. In fact, it belongs to the special configuration memory space (2000h-3FFFh), which can be accessed only during programming.

FIGURE 9-7: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 1



- Note 1:** This circuit will activate Reset when VDD goes below (Vz + 0.7V) where Vz = Zener voltage.
- Note 2:** Internal Brown-out Reset circuitry should be disabled when using this circuit.

FIGURE 9-8: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 2

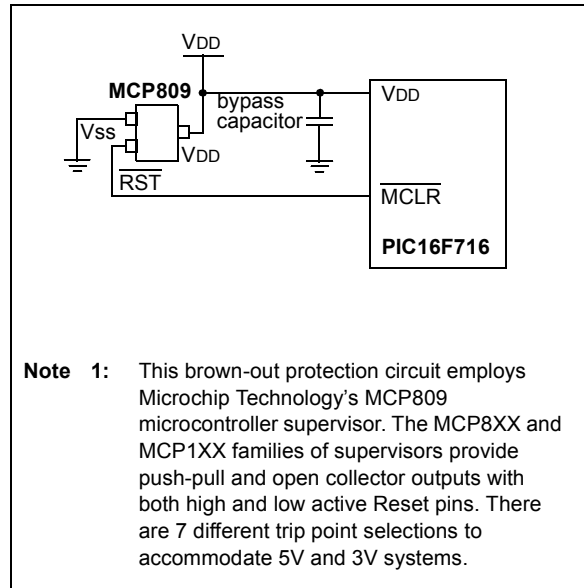


- Note 1:** This brown-out circuit is less expensive, albeit less accurate. Transistor Q1 turns off when VDD is below a certain level such that:

$$V_{DD} \times \frac{R1}{R1 + R2} = 0.7 \text{ V}$$

- Note 2:** Internal Brown-out Reset should be disabled when using this circuit.
- Note 3:** Resistors should be adjusted for the characteristics of the transistor.

FIGURE 9-9: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 3



- Note 1:** This brown-out protection circuit employs Microchip Technology's MCP809 microcontroller supervisor. The MCP8XX and MCP1XX families of supervisors provide push-pull and open collector outputs with both high and low active Reset pins. There are 7 different trip point selections to accommodate 5V and 3V systems.

FIGURE 9-10: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ TIED TO V_{DD})

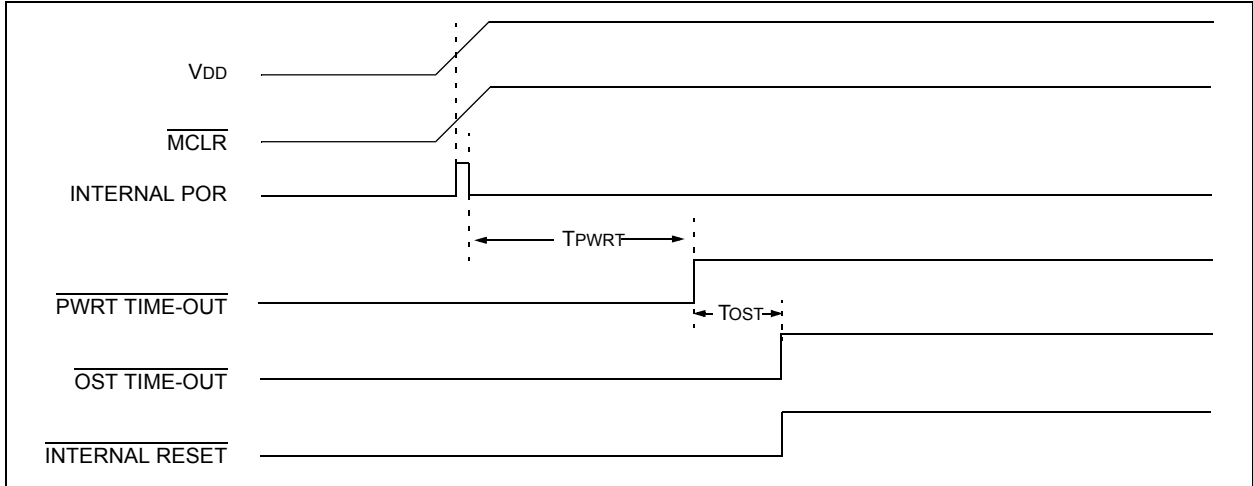


FIGURE 9-11: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ NOT TIED TO V_{DD}): CASE 1

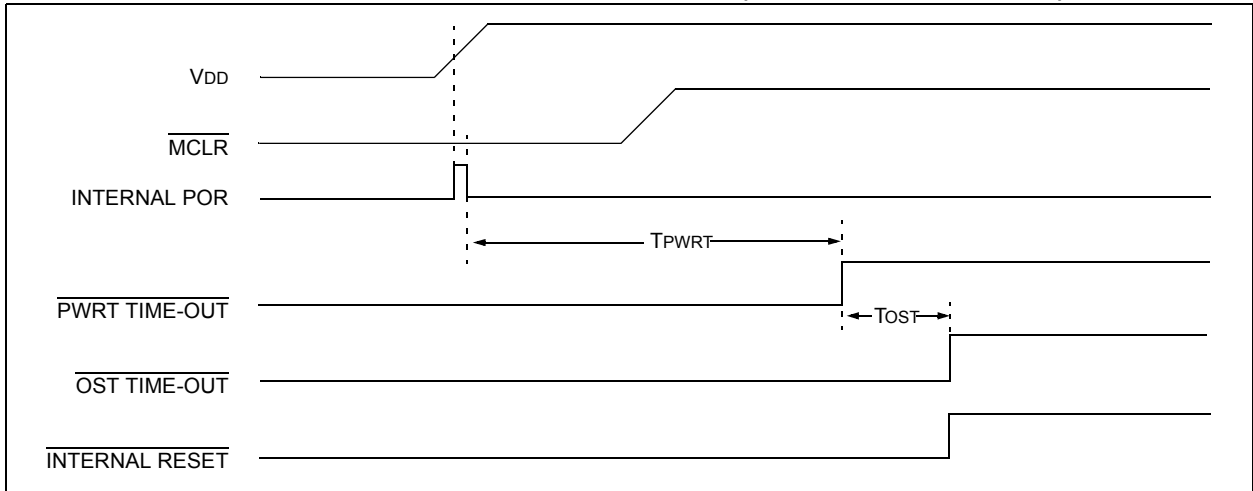
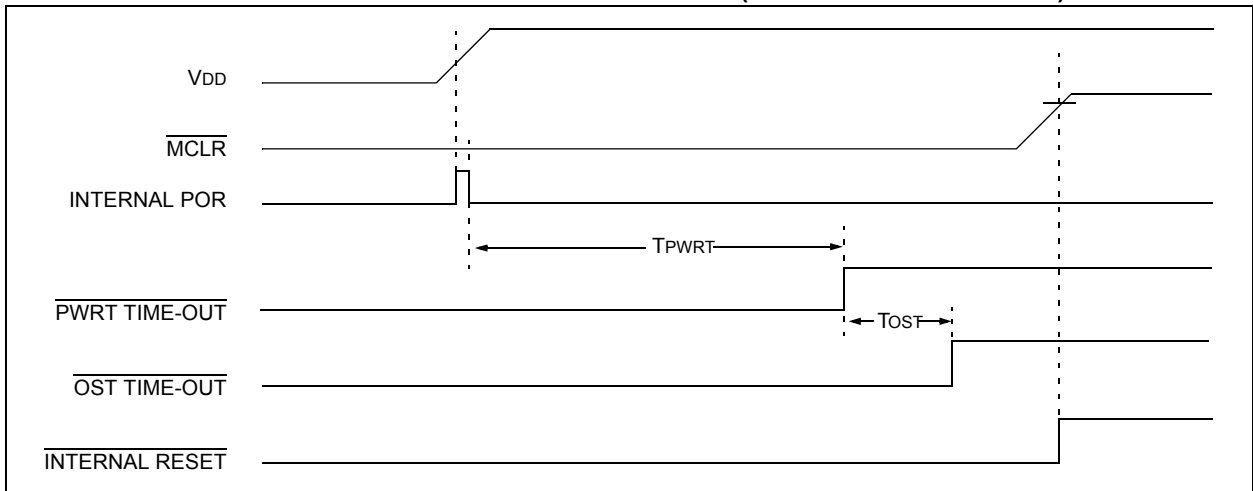


FIGURE 9-12: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ NOT TIED TO V_{DD}): CASE 2



PIC16F716

BTFS Bit Test f, Skip if Set

Syntax: [*label*] BTFS f,b
Operands: $0 \leq f \leq 127$
 $0 \leq b < 7$
Operation: skip if (f) = 1
Status Affected: None
Description: If bit 'b' in register 'f' is '0', the next instruction is executed.
If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead, making this a two-cycle instruction.

CALL Call Subroutine

Syntax: [*label*] CALL k
Operands: $0 \leq k \leq 2047$
Operation: (PC)+ 1 → TOS,
k → PC<10:0>,
(PCLATH<4:3>) → PC<12:11>
Status Affected: None
Description: Call Subroutine. First, return address (PC + 1) is pushed onto the stack. The eleven-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a two-cycle instruction.

CLRF Clear f

Syntax: [*label*] CLRF f
Operands: $0 \leq f \leq 127$
Operation: 00h → (f)
1 → Z
Status Affected: Z
Description: The contents of register 'f' are cleared and the Z bit is set.

CLRW Clear W

Syntax: [*label*] CLRW
Operands: None
Operation: 00h → (W)
1 → Z
Status Affected: Z
Description: W register is cleared. Zero bit (Z) is set.

CLRWD Clear Watchdog Timer

Syntax: [*label*] CLRWD
Operands: None
Operation: 00h → WDT
0 → WDT prescaler,
1 → \overline{TO}
1 → \overline{PD}
Status Affected: \overline{TO} , \overline{PD}
Description: CLRWD instruction resets the Watchdog Timer. It also resets the prescaler of the WDT.
Status bits \overline{TO} and \overline{PD} are set.

COMF Complement f

Syntax: [*label*] COMF f,d
Operands: $0 \leq f \leq 127$
d ∈ [0,1]
Operation: (\bar{f}) → (destination)
Status Affected: Z
Description: The contents of register 'f' are complemented. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'.

DECF Decrement f

Syntax: [*label*] DECF f,d
Operands: $0 \leq f \leq 127$
d ∈ [0,1]
Operation: (f) - 1 → (destination)
Status Affected: Z
Description: Decrement register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

PIC16F716

MOVF	Move f
Syntax:	[<i>label</i>] MOVF f,d
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	(f) → (dest)
Status Affected:	Z
Description:	The contents of register f is moved to a destination dependent upon the status of d. If d = 0, destination is W register. If d = 1, the destination is file register f itself. d = 1 is useful to test a file register since status flag Z is affected.
Words:	1
Cycles:	1
Example:	MOVF FSR, 0 After Instruction W = value in FSR register Z = 1

MOVLW	Move literal to W
Syntax:	[<i>label</i>] MOVLW k
Operands:	$0 \leq k \leq 255$
Operation:	k → (W)
Status Affected:	None
Description:	The eight-bit literal 'k' is loaded into W register. The "don't cares" will assemble as '0's.
Words:	1
Cycles:	1
Example:	MOVLW 0x5A After Instruction W = 0x5A

MOVWF	Move W to f
Syntax:	[<i>label</i>] MOVWF f
Operands:	$0 \leq f \leq 127$
Operation:	(W) → (f)
Status Affected:	None
Description:	Move data from W register to register 'f'.
Words:	1
Cycles:	1
Example:	MOVWF OPTION F Before Instruction OPTION = 0xFF W = 0x4F After Instruction OPTION = 0x4F W = 0x4F

NOP	No Operation
Syntax:	[<i>label</i>] NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Description:	No operation.
Words:	1
Cycles:	1
Example:	NOP

PIC16F716

11.2 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for all PIC 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 IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multi-purpose source files
- Directives that allow complete control over the assembly process

11.3 MPLAB C18 and MPLAB C30 C Compilers

The MPLAB C18 and MPLAB C30 Code Development Systems are complete ANSI C compilers for Microchip's PIC18 and PIC24 families of microcontrollers and the dsPIC30 and dsPIC33 family of digital signal controllers. These compilers provide powerful integration capabilities, superior code optimization and ease of use not found with other compilers.

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

11.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler and the MPLAB C18 C Compiler. 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/librarian 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 ASM30 Assembler, Linker and Librarian

MPLAB ASM30 Assembler produces relocatable machine code from symbolic assembly language for dsPIC30F devices. MPLAB C30 C 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 dsPIC30F instruction set
- Support for fixed-point and floating-point data
- Command line interface
- Rich directive set
- Flexible macro language
- MPLAB IDE compatibility

11.6 MPLAB SIM Software Simulator

The MPLAB SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC® DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB SIM Software Simulator fully supports symbolic debugging using the MPLAB C18 and MPLAB C30 C Compilers, and the MPASM and MPLAB ASM30 Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

FIGURE 12-6: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING⁽¹⁾

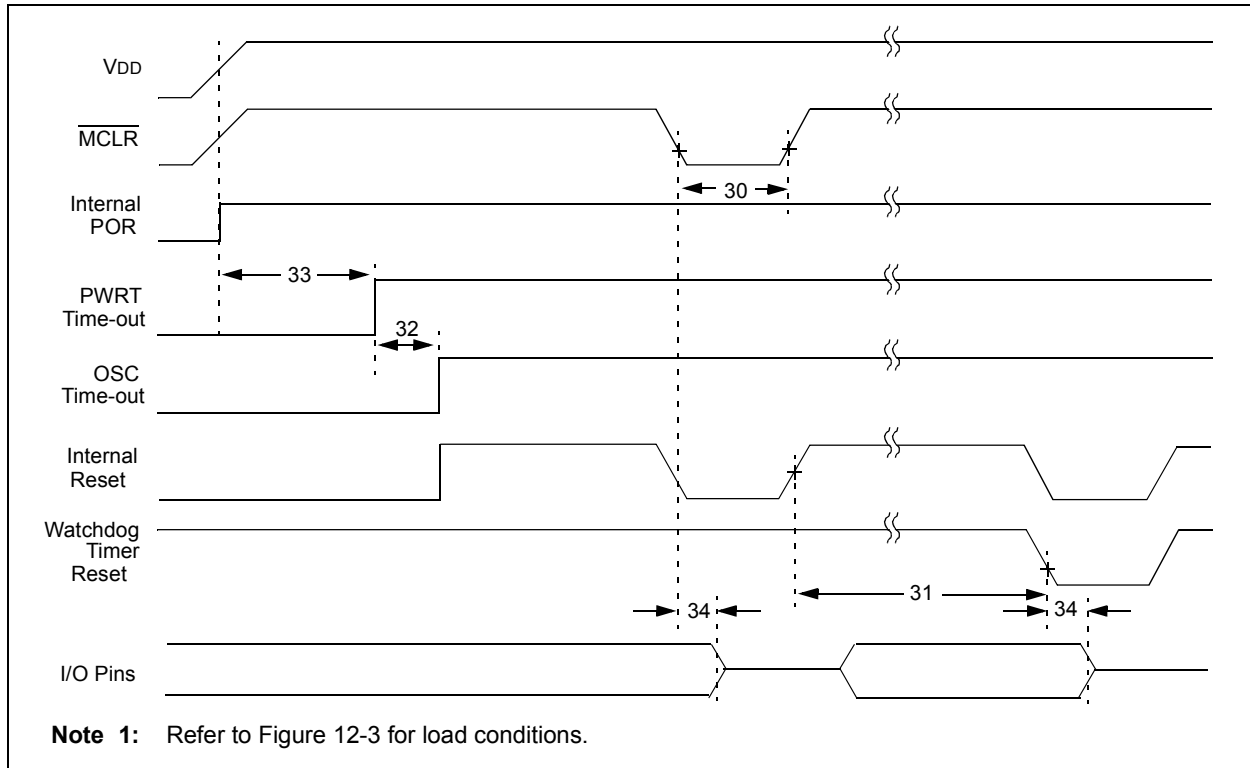


FIGURE 12-7: BROWN-OUT RESET TIMING

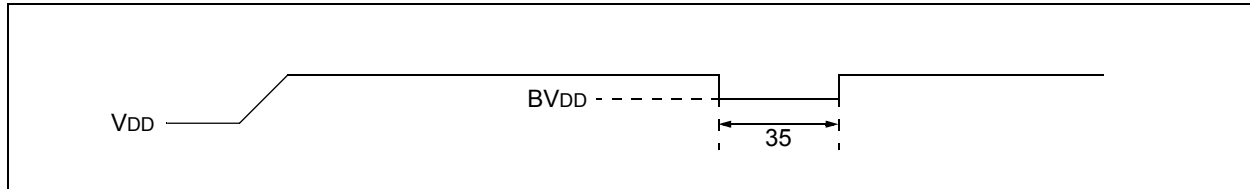


TABLE 12-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER, AND BROWN-OUT RESET REQUIREMENTS

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30	TMCLR	MCLR Pulse Width (low)	2	—	—	μs	VDD = 5V, -40°C to +125°C
31*	TWDT	Watchdog Timer Time-out Period (No Prescaler)	7	18	33	ms	VDD = 5V, -40°C to +85°C
			TBD	TBD	TBD	ms	VDD = 5V, +85°C to +125°C
32	TOST	Oscillation Start-up Timer Period	—	1024 TOSC	—	—	TOSC = OSC1 period
33*	TPWRT	Power-up Timer Period	28	72	132	ms	VDD = 5V, -40°C to +85°C
			TBD	TBD	TBD	ms	VDD = 5V, +85°C to +125°C
34	TIOZ	I/O high-impedance from MCLR Low or WDT Reset	—	—	2.1	μs	
35	TBOR	Brown-out Reset Pulse Width	100	—	—	μs	VDD ≤ BVDD (D005)

* These parameters are characterized but not tested.

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

FIGURE 12-9: CAPTURE/COMPARE/PWM TIMINGS⁽¹⁾

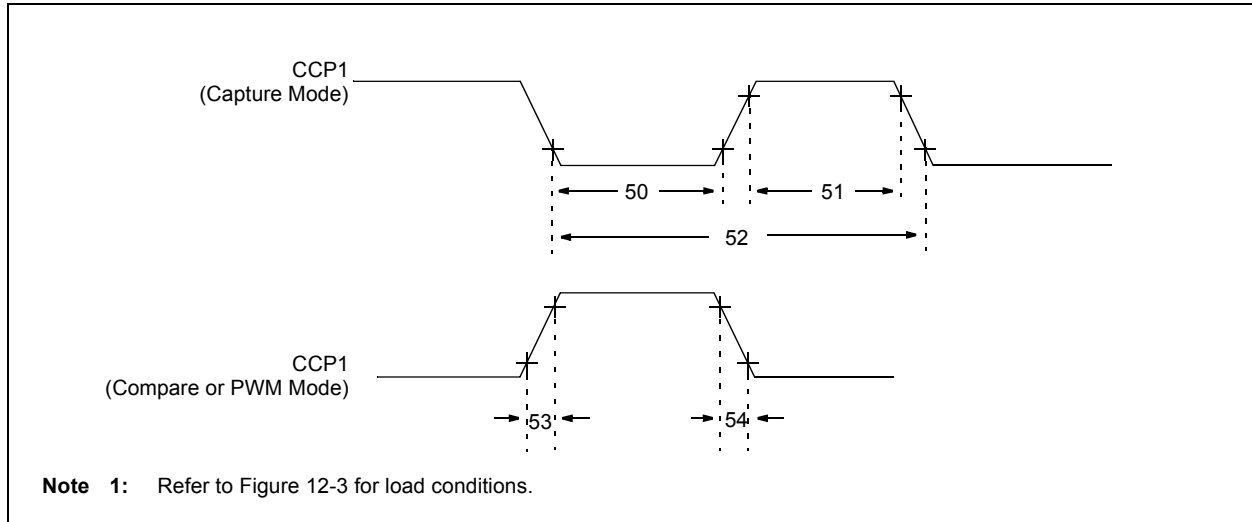


TABLE 12-6: CAPTURE/COMPARE/PWM REQUIREMENTS

Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
50*	TccL	CCP1 input low time	No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
			With Prescaler Standard	10	—	—	ns	
51*	TccH	CCP1 input high time	No Prescaler	$0.5T_{CY} + 20$	—	—	ns	
			With Prescaler Standard	10	—	—	ns	
52*	TccP	CCP1 input period		$\frac{3T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1,4, or 16)
53*	TccR	CCP1 output rise time	Standard	—	10	40	ns	
53A*			Extended	—	—	80	ns	
54*	TccF	CCP1 output fall time	Standard	—	10	40	ns	
54A*			Extended	—	—	80	ns	

* These parameters are characterized but not tested.

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

PIC16F716

FIGURE 13-14: MAXIMUM WDT IPD vs. VDD OVER TEMPERATURE

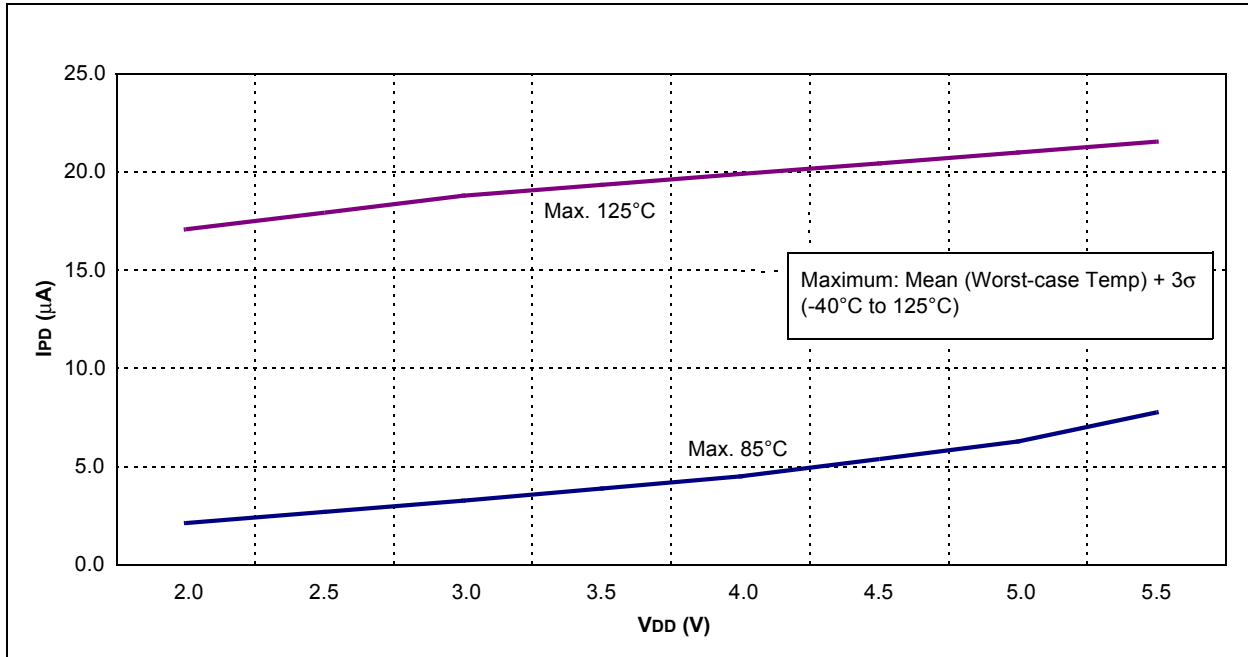
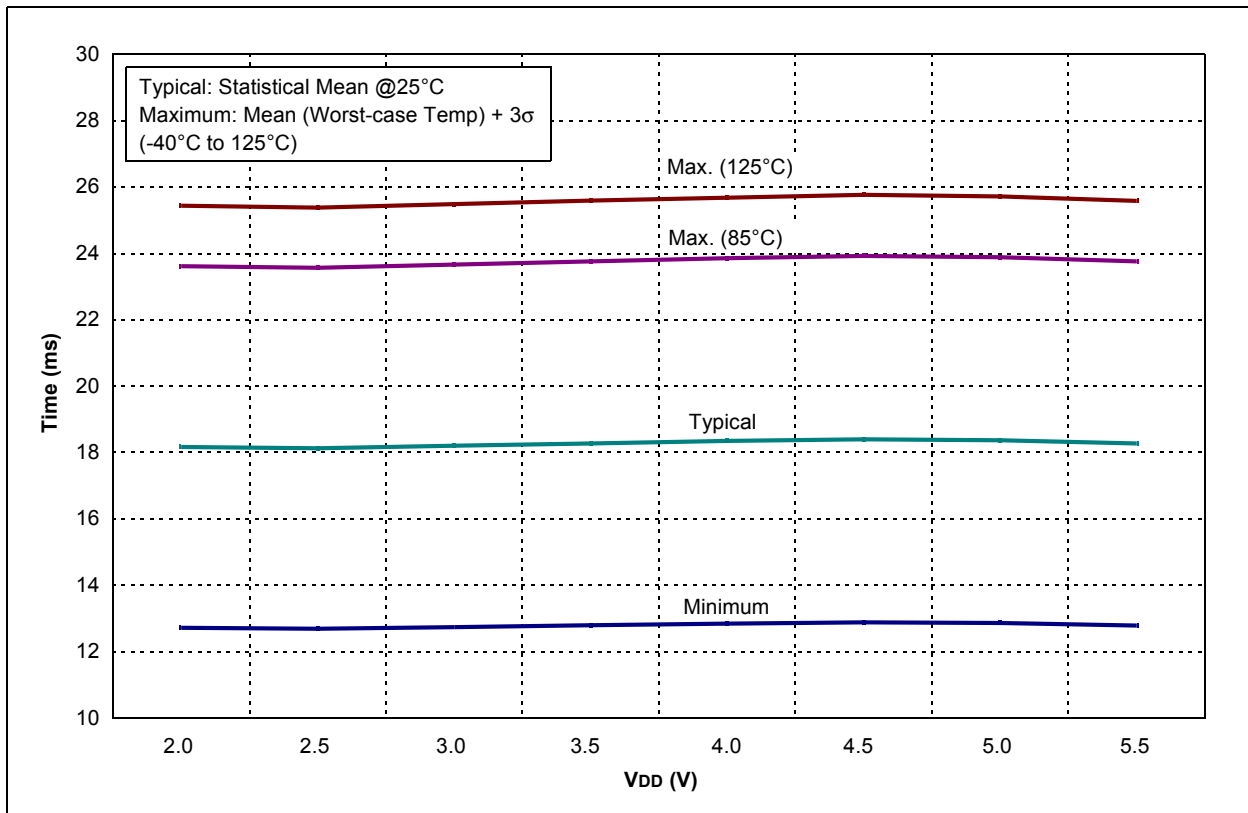


FIGURE 13-15: WDT PERIOD vs. VDD OVER TEMPERATURE



PIC16F716

FIGURE 13-17: VOL vs. IoL OVER TEMPERATURE (VDD = 3.0V)

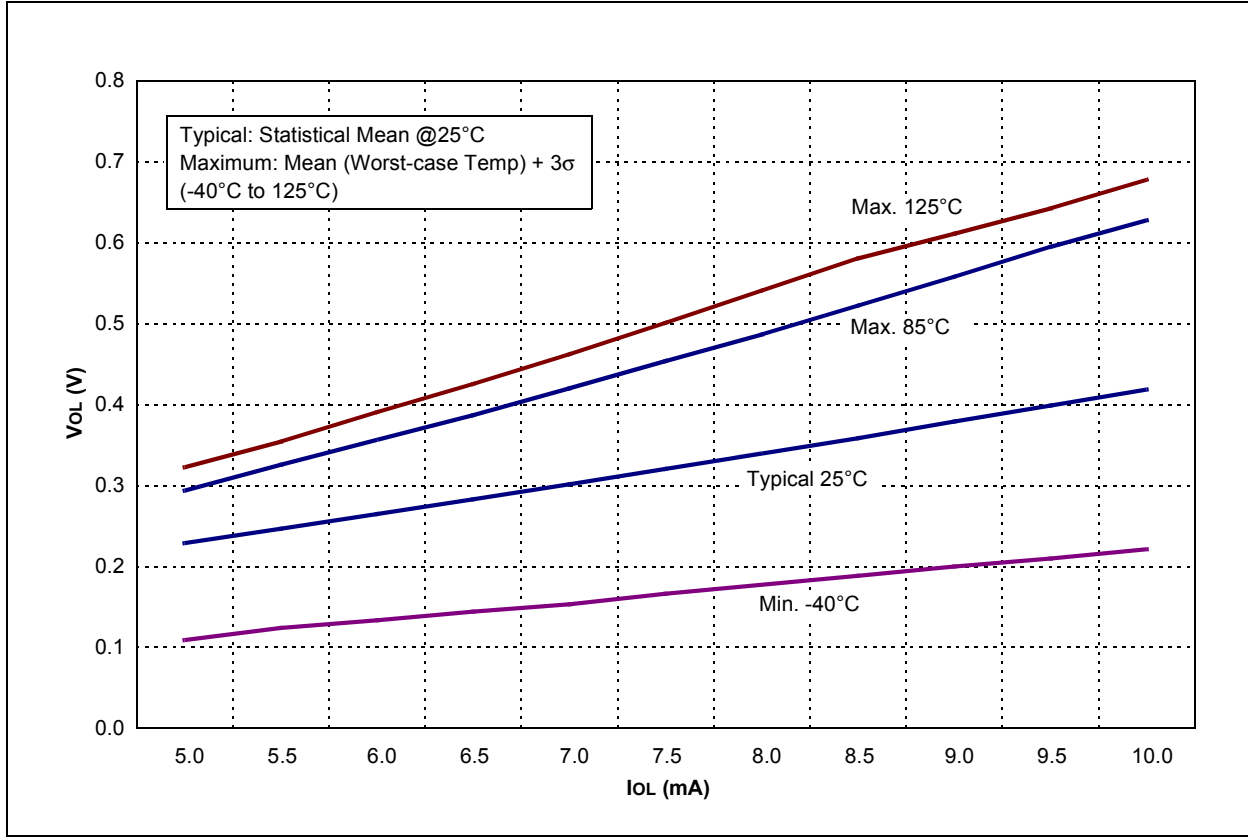
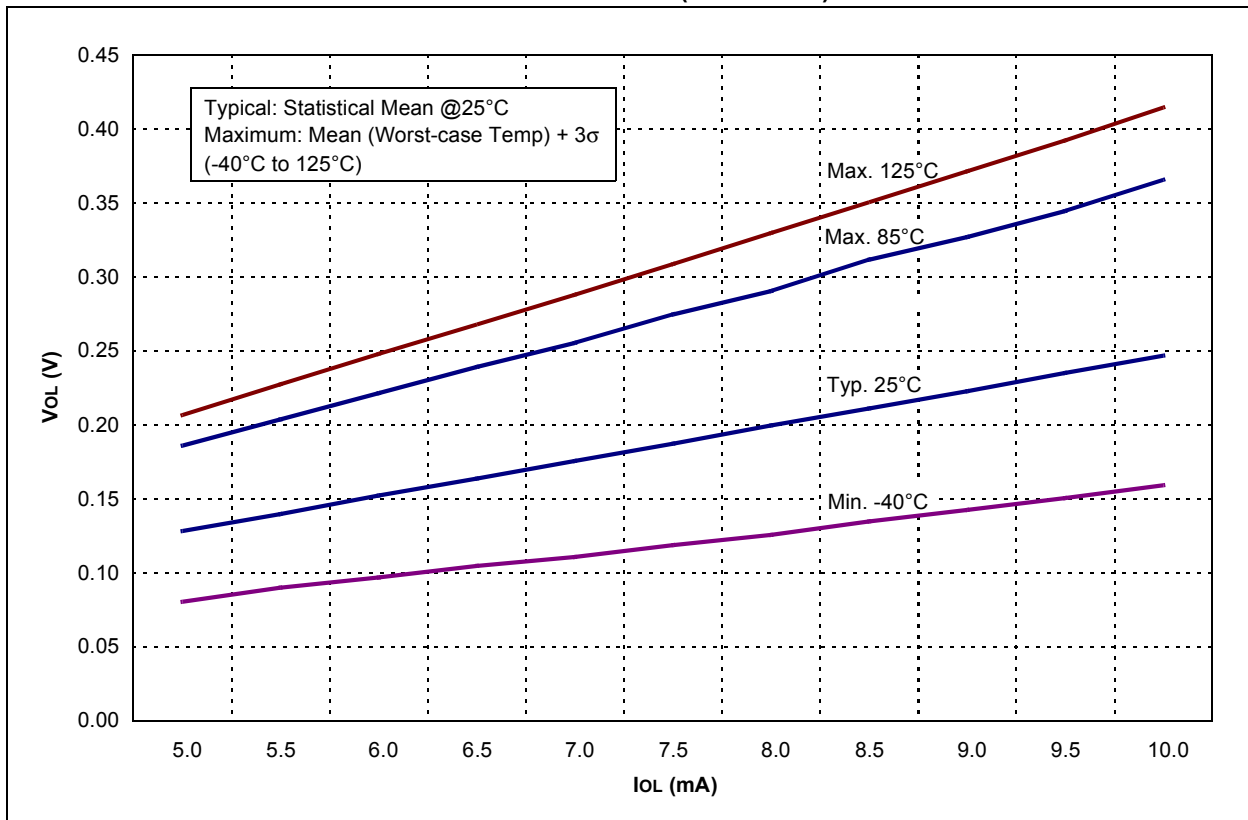


FIGURE 13-18: VOL vs. IoL OVER TEMPERATURE (VDD = 5.0V)



PIC16F716

READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this document.

To: Technical Publications Manager
RE: Reader Response
Total Pages Sent _____

From: Name _____
Company _____
Address _____
City / State / ZIP / Country _____
Telephone: (_____) _____ - _____ FAX: (_____) _____ - _____

Application (optional):

Would you like a reply? ___Y ___N

Device: PIC16F716

Literature Number: DS41206B

Questions:

1. What are the best features of this document?

2. How does this document meet your hardware and software development needs?

3. Do you find the organization of this document easy to follow? If not, why?

4. What additions to the document do you think would enhance the structure and subject?

5. What deletions from the document could be made without affecting the overall usefulness?

6. Is there any incorrect or misleading information (what and where)?

7. How would you improve this document?
