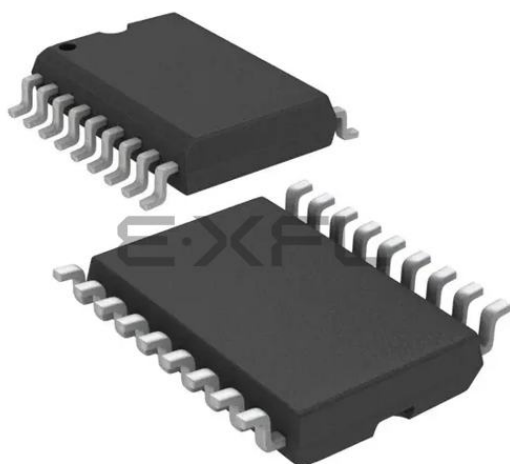




Welcome to [E-XFL.COM](https://www.e-xfl.com)

What is "[Embedded - Microcontrollers](#)"?

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

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

Details

Product Status	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	20MHz
Connectivity	I ² C, SPI
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	15
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 6x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°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/pic16lc717t-i-so

File Address		File Address		File Address		File Address	
Indirect addr.(*)	00h	Indirect addr.(*)	80h	Indirect addr.(*)	100h	Indirect addr.(*)	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h		105h		185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
	07h		87h		107h		187h
	08h		88h		108h		188h
	09h		89h		109h		189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	PMDATL	10Ch	PMCON1	18Ch
PIR2	0Dh	PIE2	8Dh	PMADRL	10Dh		18Dh
TMR1L	0Eh	PCON	8Eh	PMDATH	10Eh		18Eh
TMR1H	0Fh		8Fh	PMADRH	10Fh		18Fh
T1CON	10h		90h		110h		190h
TMR2	11h	SSPCON2	91h		111h		191h
T2CON	12h	PR2	92h		112h		192h
SSPBUF	13h	SSPAD	93h		113h		193h
SSPCON	14h	SSPSTAT	94h		114h		194h
CCPR1L	15h	WPUB	95h		115h		195h
CCPR1H	16h	IOCB	96h		116h		196h
CCP1CON	17h	P1DEL	97h		117h		197h
	18h		98h		118h		198h
	19h		99h		119h		199h
	1Ah		9Ah		11Ah		19Ah
	1Bh	REFCON	9Bh		11Bh		19Bh
	1Ch	LVDCON	9Ch		11Ch		19Ch
	1Dh	ANSEL	9Dh		11Dh		19Dh
ADRESH	1Eh	ADRESL	9Eh		11Eh		19Eh
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh
General Purpose Register 96 Bytes	20h	General Purpose Register 80 Bytes	A0h	General Purpose Register 80 Bytes	120h		1A0h
			EFh		16Fh		1EFh
		accesses 70h-7Fh	F0h	accesses 70h - 7Fh	170h	accesses 70h - 7Fh	1F0h
			FFh		17Fh		1FFh
Bank 0	7Fh	Bank 1		Bank 2		Bank 3	

Unimplemented data memory locations, read as '0'.

* Not a physical register.

PIC16C717/770/771

TABLE 2-1: PIC16C717/770/771 SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Details on Page:
Bank 2											
100h ⁽³⁾	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	23
101h	TMR0	Timer0 module's register								xxxx xxxx	45
102h ⁽³⁾	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	22
103h ⁽³⁾	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	14
104h ⁽³⁾	FSR	Indirect data memory address pointer								xxxx xxxx	23
105h	—	Unimplemented								—	—
106h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xx11	33
107h	—	Unimplemented								—	—
108h	—	Unimplemented								—	—
109h	—	Unimplemented								—	—
10Ah ^(1,3)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter				---0 0000	22	
10Bh ⁽³⁾	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	16
10Ch	PMDATL	Program memory read data low								xxxx xxxx	
10Dh	PMADRL	Program memory read address low								xxxx xxxx	
10Eh	PMDATH	—	—	Program memory read data high				--xx xxxx			
10Fh	PMADRH	—	—	—	—	Program memory read address high			---- xxxx		
110h-11Fh	—	Unimplemented								—	—
Bank 3											
180h ⁽³⁾	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	23
181h	OPTION_REG	$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	15
182h ⁽³⁾	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	22
183h ⁽³⁾	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	14
184h ⁽³⁾	FSR	Indirect data memory address pointer								xxxx xxxx	23
185h	—	Unimplemented								—	—
186h	TRISB	PORTB Data Direction Register								1111 1111	33
187h	—	Unimplemented								—	—
188h	—	Unimplemented								—	—
189h	—	Unimplemented								—	—
18Ah ^(1,3)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter				---0 0000	22	
18Bh ⁽³⁾	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	16
18Ch	PMCON1	Reserved	—	—	—	—	—	—	RD	1--- ---0	
18Dh-18Fh	—	Unimplemented								—	—

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0'.

Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.

2: Other (non Power-up) Resets include external RESET through MCLR and Watchdog Timer Reset.

3: These registers can be addressed from any bank.

2.2.2.6 PIE2 REGISTER

This register contains the individual enable bits for the SSP bus collision and low voltage detect interrupts.

REGISTER 2-6: PERIPHERAL INTERRUPT ENABLE REGISTER 2 (PIE2: 8Dh)

R/W-0	U-0	U-0	U-0	R/W-0	U-0	U-0	U-0
LVDIE	—	—	—	BCLIE	—	—	—
bit 7				bit 0			

bit 7 **LVDIE:** Low Voltage Detect Interrupt Enable bit

1 = LVD Interrupt is enabled
0 = LVD Interrupt is disabled

bit 6-4 **Unimplemented:** Read as '0'

bit 3 **BCLIE:** Bus Collision Interrupt Enable bit

1 = Bus Collision interrupt is enabled
0 = Bus Collision interrupt is disabled

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

Legend:

R = Readable bit
- n = Value at POR

W = Writable bit
'1' = Bit is set

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

3.3 PORTB and the TRISB Register

PORTB is an 8-bit wide bi-directional port. The corresponding data direction register is TRISB. Setting a TRISB bit (=1) will make the corresponding PORTB pin an input (i.e., put the corresponding output driver in a Hi-impedance mode). Clearing a TRISB bit (=0) will make the corresponding PORTB pin an output (i.e., put the contents of the output latch on the selected pin).

EXAMPLE 3-2: Initializing PORTB

```
BCF    STATUS, RP0 ;
CLRF   PORTB       ; Initialize PORTB by
                   ; clearing output
                   ; data latches
BSF    STATUS, RP0 ; Select Bank 1
MOVLW  0xCF         ; Value used to
                   ; initialize data
                   ; direction
MOVWF  TRISB        ; Set RB<3:0> as inputs
                   ; RB<5:4> as outputs
                   ; RB<7:6> as inputs
MOVLW  0x30         ; Set RB<1:0> as analog
                   ; inputs
MOVWF  ANSEL        ;
BCF    STATUS, RP0 ; Return to Bank 0
```

Each of the PORTB pins has an internal pull-up, which can be individually enabled from the WPUB register. A single global enable bit can turn on/off the enabled pull-ups. Clearing the RBPU bit, (OPTION_REG<7>), enables the weak pull-up resistors. The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset.

Each of the PORTB pins, if configured as input, also has an interrupt-on-change feature, which can be individually selected from the IOCB register. The RBIE bit in the INTCON register functions as a global enable bit to turn on/off the interrupt-on-change feature. The selected inputs are compared to the old value latched on the last read of PORTB. The "mismatch" outputs are OR'ed together to generate the RB Port Change Interrupt with flag bit RBIF (INTCON<0>).

This interrupt can wake the device from SLEEP. The user, in the interrupt service routine, can clear the interrupt in the following manner:

- a) Any read or write of PORTB. This will end the mismatch condition.
- a) Clear flag bit RBIF.

A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition and allow flag bit RBIF to be cleared.

The interrupt-on-change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt-on-change feature. Polling of PORTB is not recommended while using the interrupt-on-change feature.

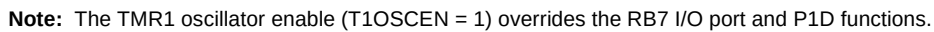


FIGURE 8-7: EXAMPLE OF HALF-BRIDGE OUTPUT MODE APPLICATIONS

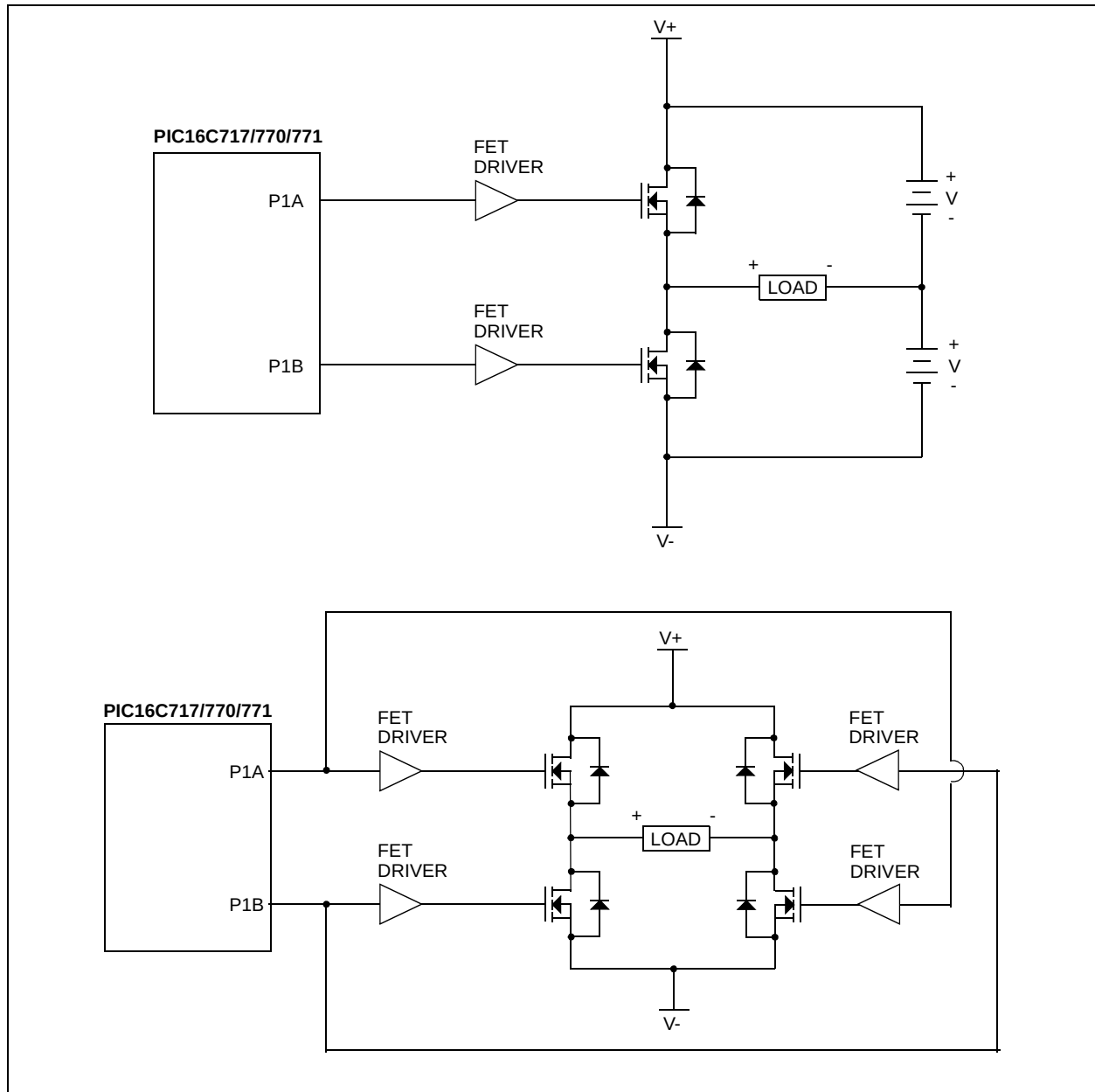


FIGURE 8-9: EXAMPLE OF FULL-BRIDGE APPLICATION

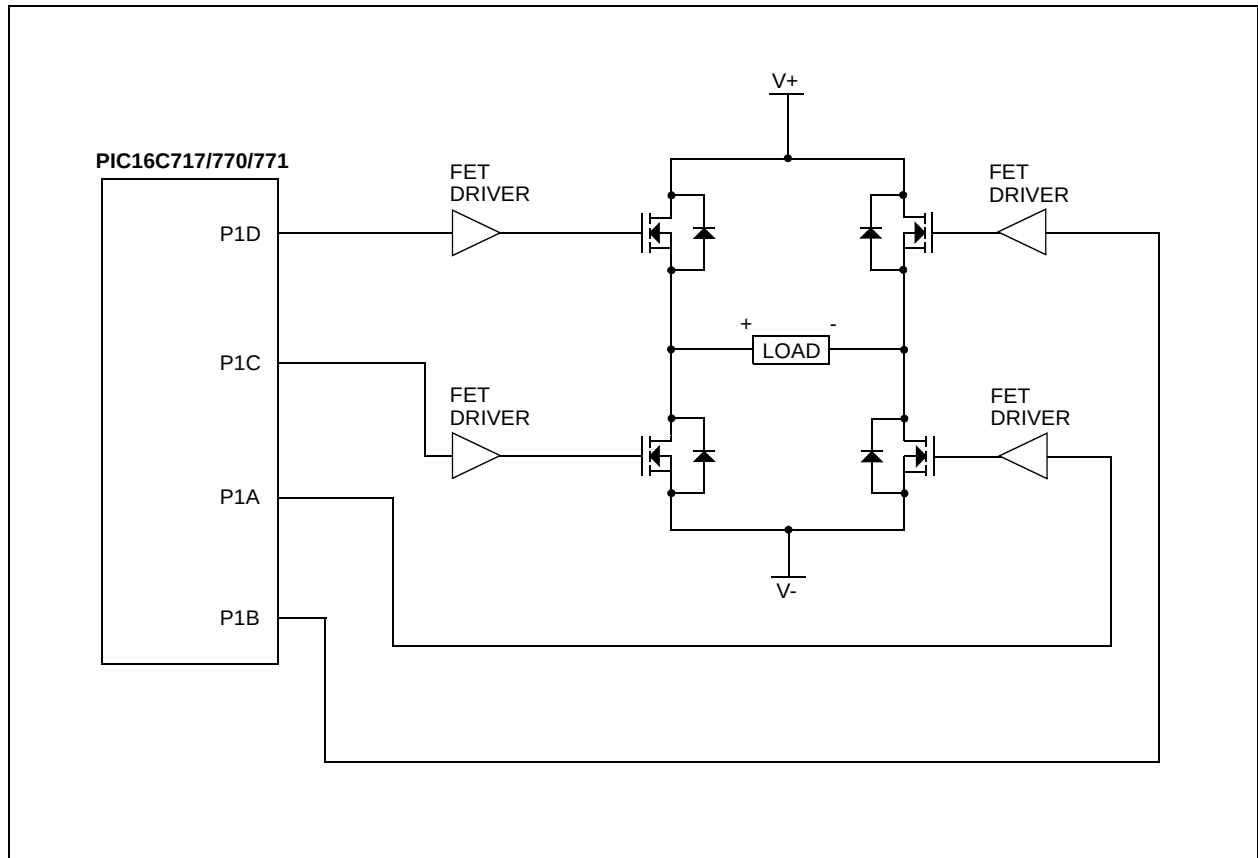
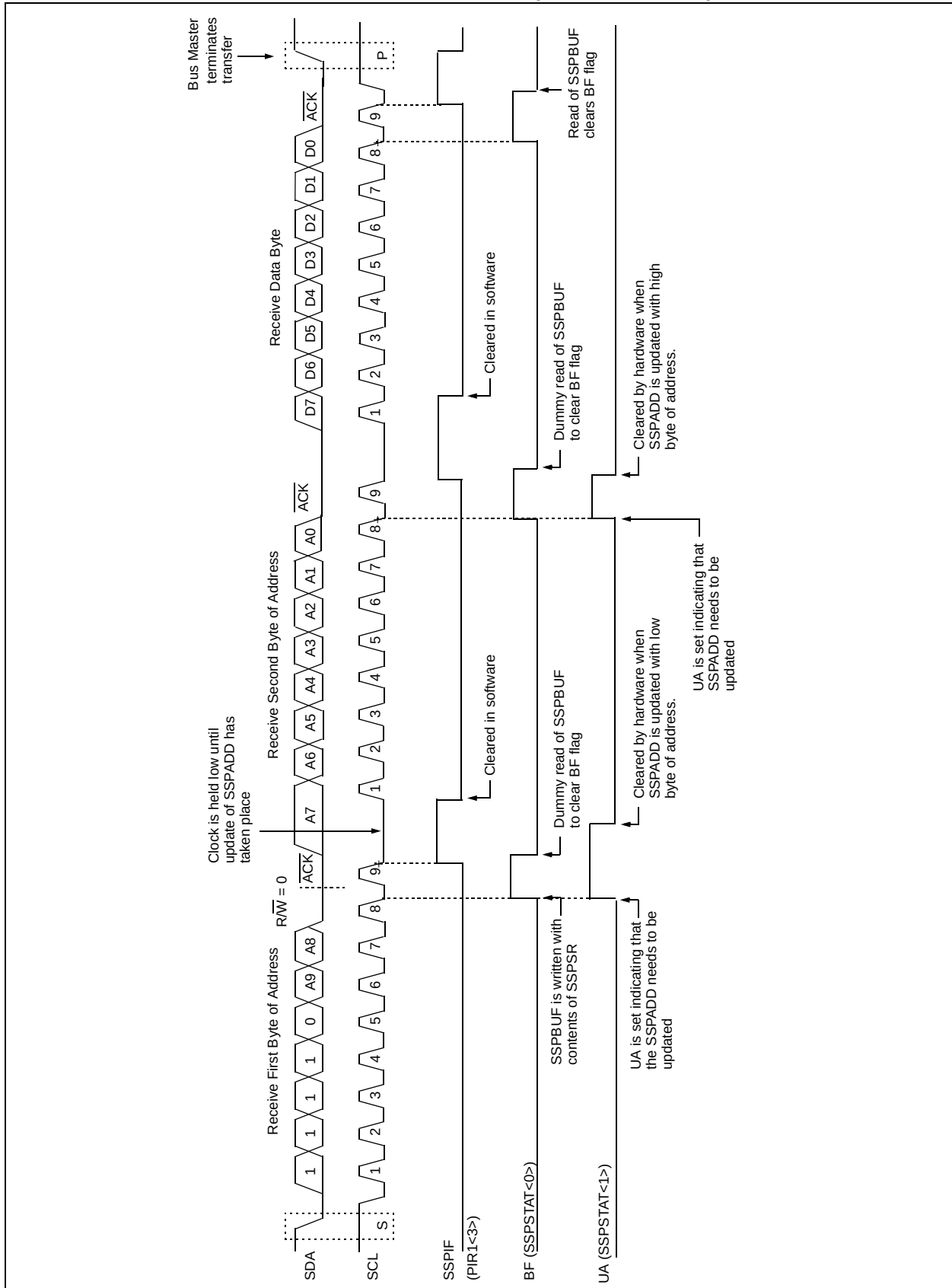


FIGURE 9-9: I²C SLAVE MODE FOR RECEPTION (10-BIT ADDRESS)



9.2.7 MULTI-MASTER OPERATION

In Multi-Master mode, the interrupt generation on the detection of the START and STOP conditions allows the determination of when the bus is free. The STOP (P) and START (S) bits are cleared from a RESET or when the MSSP module is disabled. Control of the I²C bus may be taken when bit P (SSPSTAT<4>) is set, or the bus is idle with both the S and P bits clear. When the bus is busy, enabling the SSP Interrupt will generate the interrupt when the STOP condition occurs.

In multi-master operation, the SDA line must be monitored for arbitration to see if the signal level is the expected output level. This check is performed in hardware, with the result placed in the BCLIF bit.

The states where arbitration can be lost are:

- Address Transfer
- Data Transfer
- A START Condition
- A Repeated START Condition
- An Acknowledge Condition

Refer to Application Note AN578, "Use of the SSP Module in the I²C™ Multi-Master Environment."

9.2.8 I²C MASTER OPERATION

Master mode is enabled by setting and clearing the appropriate SSPM bits in SSPCON and by setting the SSPEN bit. Once Master mode is enabled, the user has six options.

1. Assert a START condition on SDA and SCL.
2. Assert a Repeated START condition on SDA and SCL.
3. Write to the SSPBUF register initiating transmission of data/address.
4. Generate a STOP condition on SDA and SCL.
5. Configure the I²C port to receive data.
6. Generate an Acknowledge condition at the end of a received byte of data.

The master device generates all serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a Repeated START condition. Since the Repeated START condition is also the beginning of the next serial transfer, the I²C bus will not be released.

Note: The MSSP Module, when configured in I²C Master mode, does not allow queueing of events. For instance, the user is not allowed to initiate a START condition and immediately write the SSPBUF register to initiate transmission before the START condition is complete. In this case, the SSPBUF will not be written to, and the WCOL bit will be set, indicating that a write to the SSPBUF did not occur.

9.2.9 BAUD RATE GENERATOR

The baud rate generator used for SPI mode operation is used in the I²C Master mode to set the SCL clock frequency. Standard SCL clock frequencies are 100 kHz, 400 kHz, and 1 MHz. One of these frequencies can be achieved by setting the SSPADD register to the appropriate number for the selected Fosc frequency. One half of the SCL period is equal to $[(SSPADD+1) \bullet 2]/F_{osc}$.

The baud rate generator reload value is contained in the lower seven bits of the SSPADD register (Figure 9-14). When the BRG is loaded with this value, the BRG counts down to 0 and stops until another reload occurs. The BRG count is decremented twice per instruction cycle (Tcy) on the Q2 and Q4 clock.

In I²C Master mode, the BRG is reloaded automatically provided that the SCL line is sampled high. For example, if Clock Arbitration is taking place, the BRG reload will be suppressed until the SCL line is released by the slave allowing the pin to float high (Figure 9-15).

FIGURE 9-14: BAUD RATE GENERATOR BLOCK DIAGRAM

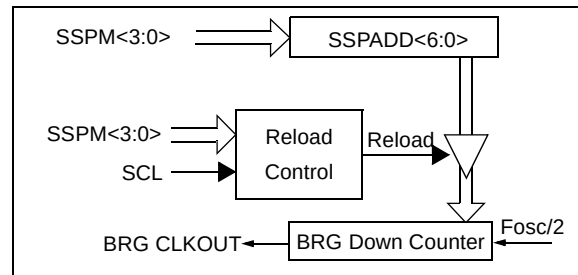
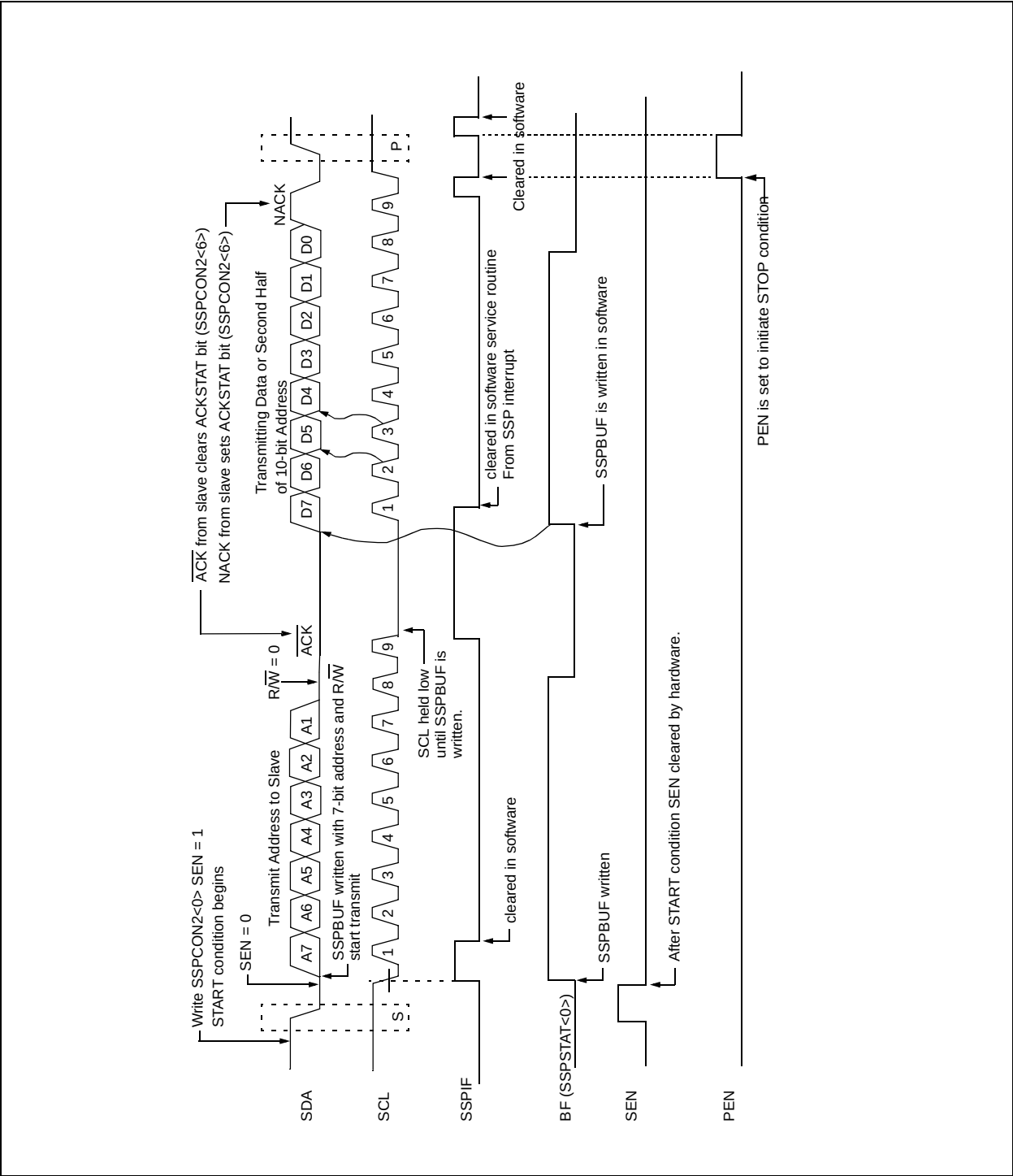


FIGURE 9-18: I²C MASTER MODE WAVEFORMS FOR TRANSMISSION (7 OR 10-BIT ADDRESS)



12.13 Power-down Mode (SLEEP)

Power-down mode is entered by executing a `SLEEP` instruction.

If enabled, the Watchdog Timer will be cleared but keeps running, the `PD` bit (`STATUS<3>`) is cleared, the `TO` (`STATUS<4>`) bit is set, and the oscillator driver is turned off. The I/O ports maintain the status they had, before the `SLEEP` instruction was executed (driving high, low, or hi-impedance).

For lowest current consumption in this mode, place all I/O pins at either V_{DD} , or V_{SS} , ensure no external circuitry is drawing current from the I/O pin, power-down the A/D, disable external clocks. Pull all I/O pins, that are hi-impedance inputs, high or low externally to avoid switching currents caused by floating inputs. The `T0CKI` input should also be at V_{DD} or V_{SS} for lowest current consumption. The contribution from on-chip pull-ups on `PORTB` should be considered.

12.13.1 WAKE-UP FROM SLEEP

The device can wake-up from `SLEEP` through one of the following events:

1. External `RESET` input on `MCLR` pin.
2. Watchdog Timer Wake-up (if `WDT` was enabled).
3. Interrupt from `INT` pin, `RB` port change, or some Peripheral Interrupts.

External `MCLR` Reset will cause a device `RESET`. All other events are considered a continuation of program execution and cause a "wake-up". The `TO` and `PD` bits in the `STATUS` register can be used to determine the cause of device `RESET`. The `PD` bit, which is set on power-up, is cleared when `SLEEP` is invoked. The `TO` bit is cleared if a `WDT` time-out occurred (and caused wake-up).

The following peripheral interrupts can wake the device from `SLEEP`:

1. `TMR1` interrupt. `Timer1` must be operating as an asynchronous counter.
2. `CCP` Capture mode interrupt.
3. Special event trigger (`Timer1` in Asynchronous mode using an external clock).
4. `SSP` (`START/STOP`) bit detect interrupt.
5. `SSP` transmit or receive in Slave mode (`SPI/I2C`).
6. A/D conversion (when A/D clock source is `RC`).
7. Low Voltage detect.

Other peripherals cannot generate interrupts since during `SLEEP`, no on-chip clocks are present.

When the `SLEEP` instruction is being executed, the next instruction (`PC + 1`) is pre-fetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set (enabled). Wake-up is regardless of the state of the `GIE` bit. If the `GIE` bit is

clear (disabled), the device continues execution at the instruction after the `SLEEP` instruction. If the `GIE` bit is set (enabled), the device executes the instruction after the `SLEEP` instruction and then branches to the interrupt address (`0004h`). In cases where the execution of the instruction following `SLEEP` is not desirable, the user should have a `NOP` after the `SLEEP` instruction.

12.13.2 WAKE-UP USING INTERRUPTS

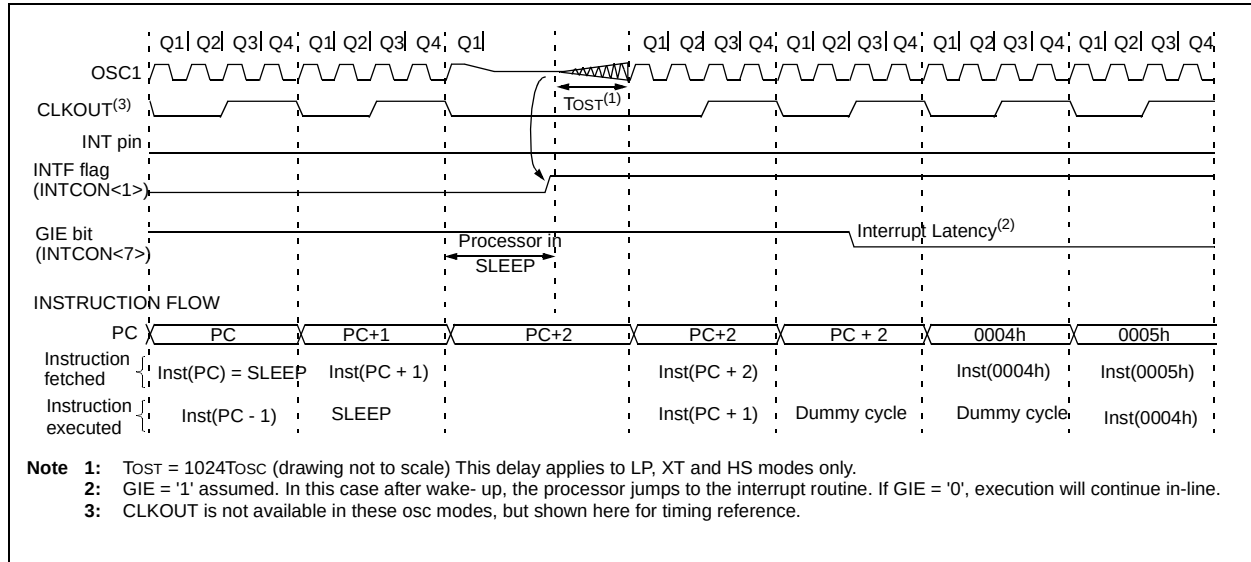
When global interrupts are disabled (`GIE` cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of a `SLEEP` instruction, the `SLEEP` instruction will complete as a `NOP`. Therefore, the `WDT` and `WDT` postscaler will not be cleared, the `TO` bit will not be set and `PD` bits will not be cleared.
- If the interrupt occurs **during or after** the execution of a `SLEEP` instruction, the device will immediately wake-up from `SLEEP`. The `SLEEP` instruction will be completely executed before the wake-up. Therefore, the `WDT` and `WDT` postscaler will be cleared, the `TO` bit will be set and the `PD` bit will be cleared.

Even if the flag bits were checked before executing a `SLEEP` instruction, it may be possible for flag bits to become set before the `SLEEP` instruction completes. To determine whether a `SLEEP` instruction executed, test the `PD` bit. If the `PD` bit is set, the `SLEEP` instruction was executed as a `NOP`.

If a peripheral can wake the device from `SLEEP`, then to ensure that the `WDT` is cleared, a `CLRWDT` instruction should be executed before a `SLEEP` instruction.

FIGURE 12-12: WAKE-UP FROM SLEEP THROUGH INTERRUPT



12.14 Program Verification/Code Protection

If the code protection bit(s) have not been programmed, the on-chip program memory can be read out for verification purposes.

Note: Microchip does not recommend code protecting windowed devices. Code protected devices are not reprogrammable.

12.15 ID Locations

Four memory locations (2000h - 2003h) are designated as ID locations where the user can store checksum or other code-identification numbers. These locations are not accessible during normal execution but are readable and writable during program/verify. It is recommended that only the 4 Least Significant bits of the ID location are used.

12.16 In-Circuit Serial Programming (ICSP™)

PIC16CXXX microcontrollers can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data, and three other lines for power, ground, and the programming voltage. This allows customers to manufacture boards with unprogrammed devices, and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

For complete details of serial programming, please refer to the In-Circuit Serial Programming (ICSP™) Guide, (DS30277).

14.8 MPLAB ICD In-Circuit Debugger

Microchip's In-Circuit Debugger, MPLAB ICD, is a powerful, low cost, run-time development tool. This tool is based on the FLASH PIC MCUs and can be used to develop for this and other PIC microcontrollers. The MPLAB ICD utilizes the in-circuit debugging capability built into the FLASH devices. This feature, along with Microchip's In-Circuit Serial Programming™ protocol, offers cost-effective in-circuit FLASH debugging from the graphical user interface of the MPLAB Integrated Development Environment. This enables a designer to develop and debug source code by watching variables, single-stepping and setting break points. Running at full speed enables testing hardware in real-time.

14.9 PRO MATE II Universal Device Programmer

The PRO MATE II universal device programmer is a full-featured programmer, capable of operating in Stand-alone mode, as well as PC-hosted mode. The PRO MATE II device programmer is CE compliant.

The PRO MATE II device programmer has programmable VDD and VPP supplies, which allow it to verify programmed memory at VDD min and VDD max for maximum reliability. It has an LCD display for instructions and error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In Stand-alone mode, the PRO MATE II device programmer can read, verify, or program PIC devices. It can also set code protection in this mode.

14.10 PICSTART Plus Entry Level Development Programmer

The PICSTART Plus development programmer is an easy-to-use, low cost, prototype programmer. It connects to the PC via a COM (RS-232) port. MPLAB Integrated Development Environment software makes using the programmer simple and efficient.

The PICSTART Plus development programmer supports all PIC devices with up to 40 pins. Larger pin count devices, such as the PIC16C92X and PIC17C76X, may be supported with an adapter socket. The PICSTART Plus development programmer is CE compliant.

14.11 PICDEM 1 Low Cost PIC MCU Demonstration Board

The PICDEM 1 demonstration board is a simple board which demonstrates the capabilities of several of Microchip's microcontrollers. The microcontrollers supported are: PIC16C5X (PIC16C54 to PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 and PIC17C44. All necessary hardware and software is included to run basic demo programs. The user can program the sample microcontrollers provided with the PICDEM 1 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The user can also connect the PICDEM 1 demonstration board to the MPLAB ICE in-circuit emulator and download the firmware to the emulator for testing. A prototype area is available for the user to build some additional hardware and connect it to the microcontroller socket(s). Some of the features include an RS-232 interface, a potentiometer for simulated analog input, push button switches and eight LEDs connected to PORTB.

14.12 PICDEM 2 Low Cost PIC16CXX Demonstration Board

The PICDEM 2 demonstration board is a simple demonstration board that supports the PIC16C62, PIC16C64, PIC16C65, PIC16C73 and PIC16C74 microcontrollers. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM 2 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The MPLAB ICE in-circuit emulator may also be used with the PICDEM 2 demonstration board to test firmware. A prototype area has been provided to the user for adding additional hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push button switches, a potentiometer for simulated analog input, a serial EEPROM to demonstrate usage of the I²C™ bus and separate headers for connection to an LCD module and a keypad.

15.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings †

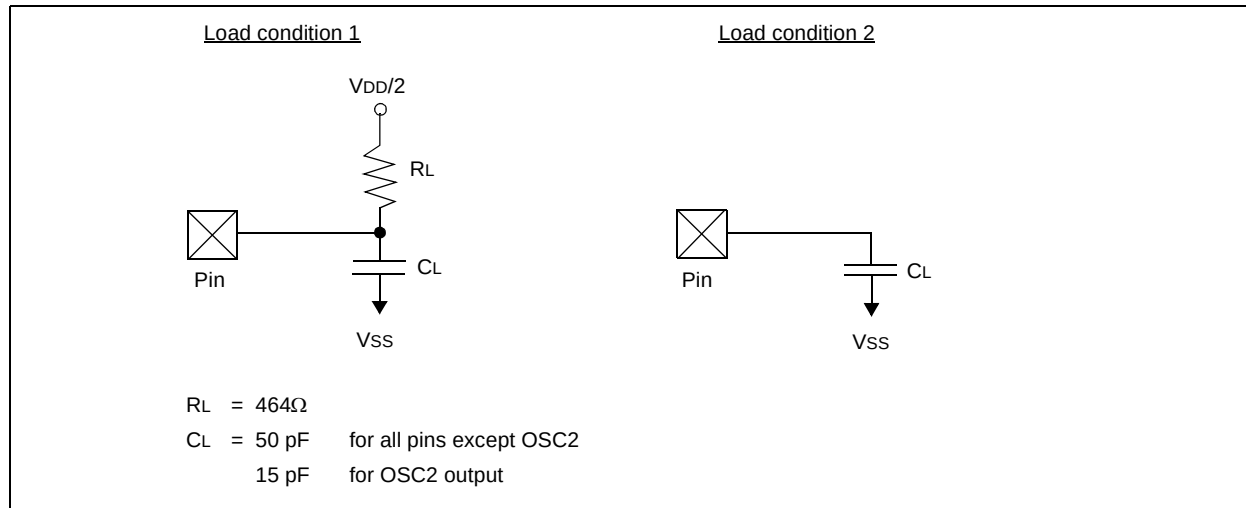
Ambient temperature under bias	-55 to +125°C
Storage temperature	-65°C to +150°C
Voltage on any pin with respect to Vss (except VDD, $\overline{\text{MCLR}}$ and RA4)	-0.3V to (VDD + 0.3V)
Voltage on VDD with respect to Vss	-0.3 to +7.5V
Maximum voltage between AVDD and VDD pins.....	± 0.3V
Maximum voltage between AVss and Vss pins	± 0.3V
Voltage on $\overline{\text{MCLR}}$ with respect to Vss.....	-0.3V to +8.5V
Voltage on RA4 with respect to Vss.....	-0.3V to +10.5V
Total power dissipation (Note 1)	1.0W
Maximum current out of Vss pin	300 mA
Maximum current into VDD pin	250 mA
Input clamp current, I _{IK} (V _I < 0 or V _I > VDD).....	± 20 mA
Output clamp current, I _{OK} (V _O < 0 or V _O > VDD)	± 20 mA
Maximum output current sunk by any I/O pin.....	25 mA
Maximum output current sourced by any I/O pin	25 mA
Maximum current sunk by PORTA and PORTB (combined).....	200 mA
Maximum current sourced by PORTA and PORTB (combined)	200 mA

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

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

PIC16C717/770/771

FIGURE 15-4: LOAD CONDITIONS



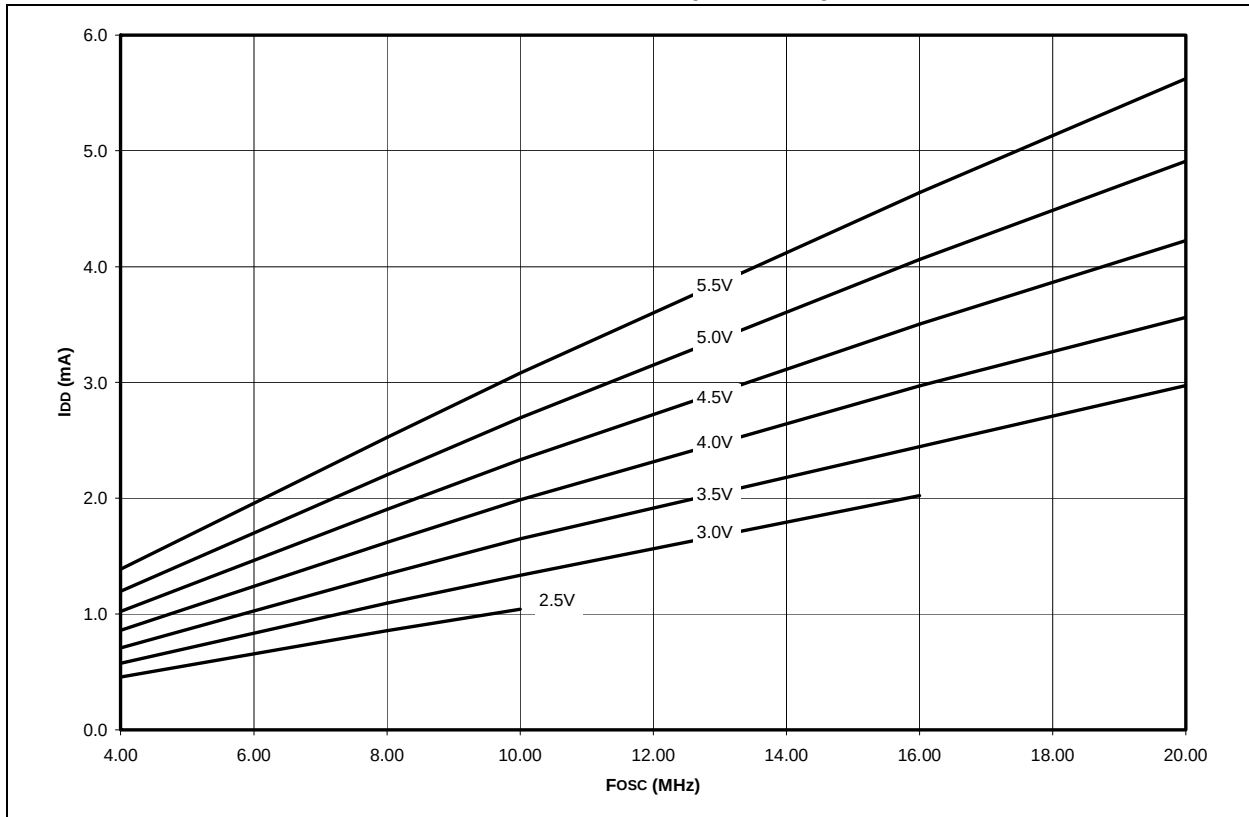
16.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES

The graphs and tables provided following this note are a statistical summary based on a limited number of samples and are provided for informational purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs or tables, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore outside the warranted range.

“Typical” represents the mean of the distribution at 25°C. “Maximum” or “minimum” represents (mean + 3σ) or (mean - 3σ) respectively, where σ is a standard deviation, over the whole temperature range.

The Fosc I_{DD} was determined using an external sinusoidal clock source with a peak amplitude ranging from V_{SS} to V_{DD}.

FIGURE 16-1: MAXIMUM I_{DD} VS. Fosc OVER V_{DD} (HS MODE)



PIC16C717/770/771

FIGURE 16-10: TYPICAL I_{DD} VS. F_{osc} OVER V_{DD} (ER MODE)

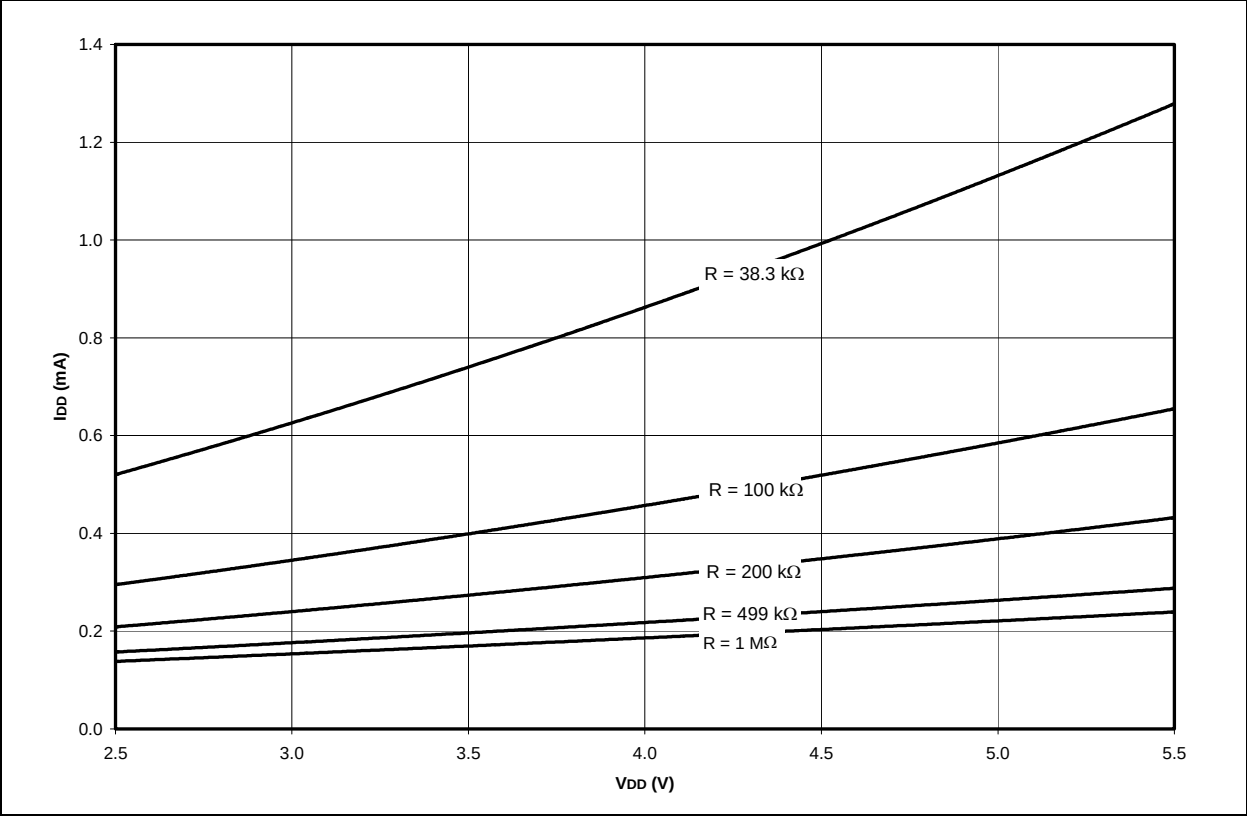


FIGURE 16-11: TYPICAL F_{osc} VS. V_{DD} (ER MODE)

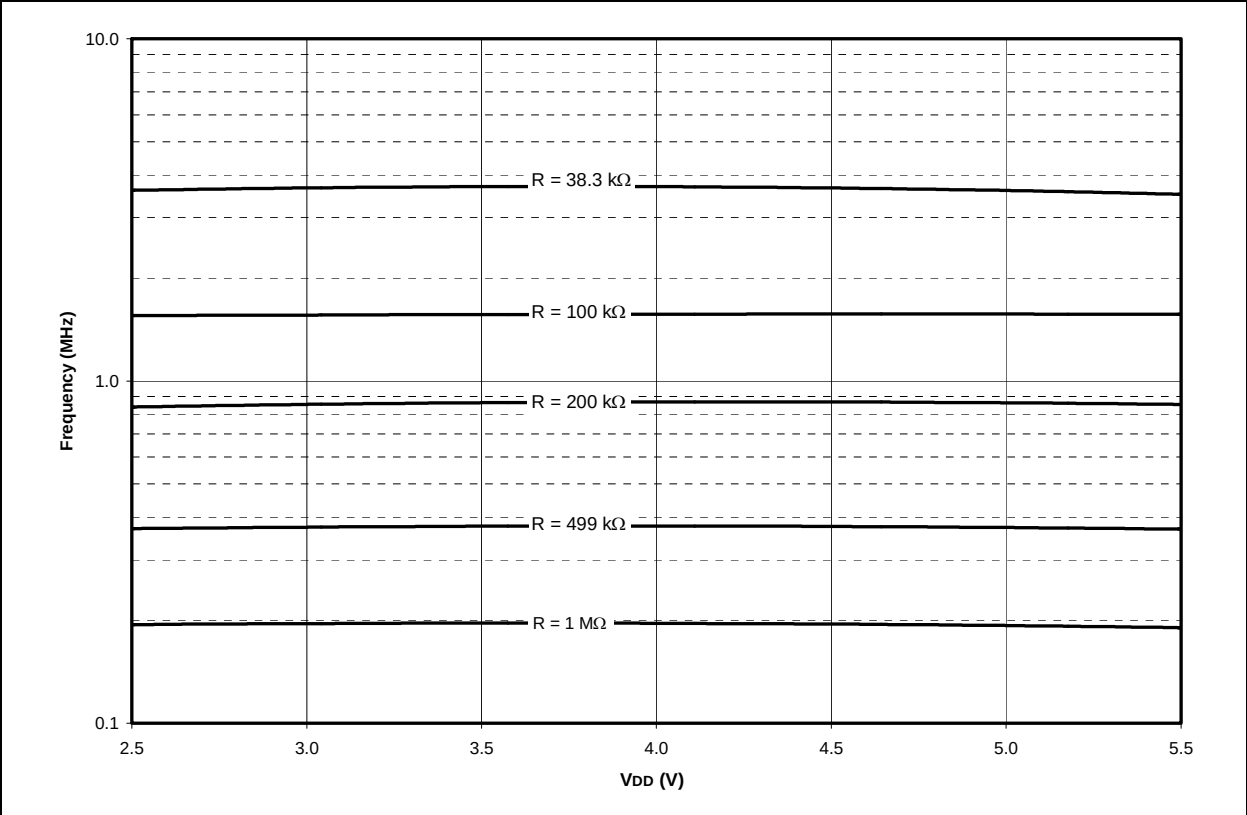


FIGURE 16-19: TYPICAL AND MAXIMUM ΔI_{WDT} VS. V_{DD} (-40°C TO +125°C)

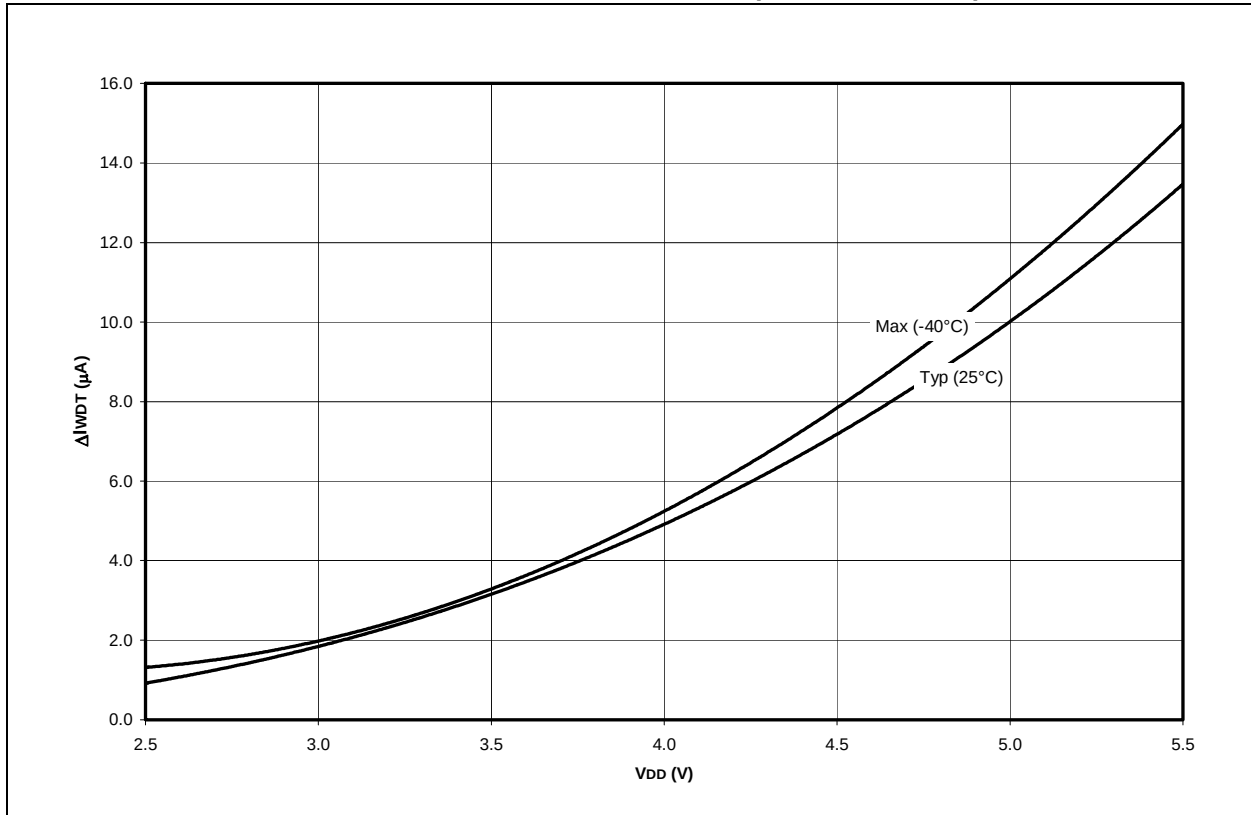


FIGURE 16-26: TYPICAL AND MAXIMUM ΔI_{BOR} VS. V_{DD} (-40°C TO $+125^{\circ}\text{C}$) ($V_{BOR} = 4.5\text{V}$)

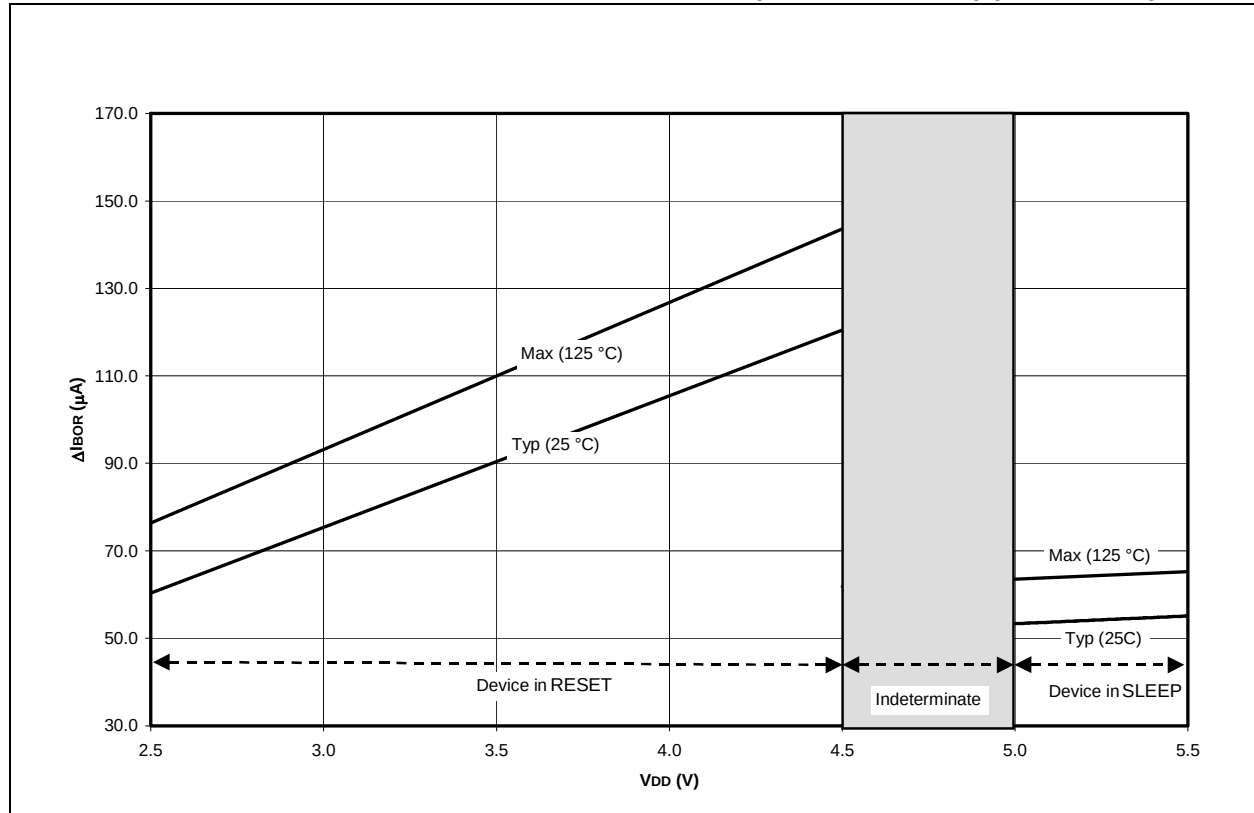
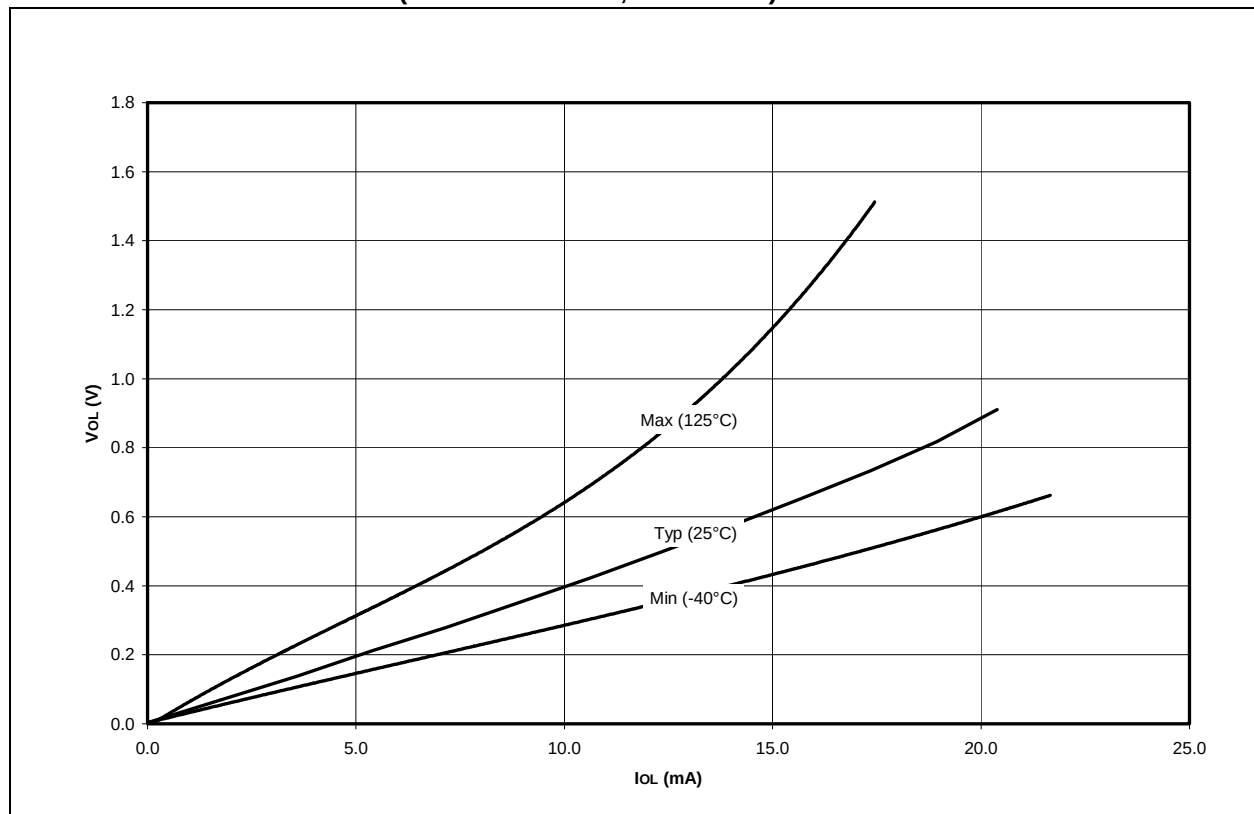
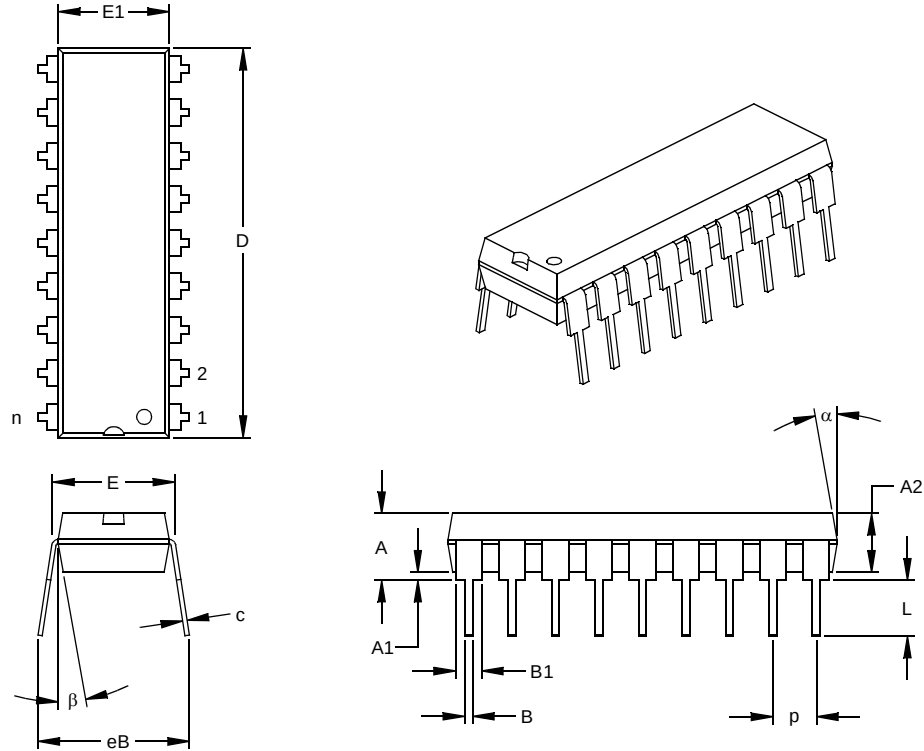


FIGURE 16-27: V_{OL} VS. I_{OL} (-40°C TO $+125^{\circ}\text{C}$, $V_{DD} = 3.0\text{V}$)



17.2 18-Lead Plastic Dual In-line (P) – 300 mil (PDIP)

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



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		18			18	
Pitch	p		.100			2.54	
Top to Seating Plane	A	.140	.155	.170	3.56	3.94	4.32
Molded Package Thickness	A2	.115	.130	.145	2.92	3.30	3.68
Base to Seating Plane	A1	.015			0.38		
Shoulder to Shoulder Width	E	.300	.313	.325	7.62	7.94	8.26
Molded Package Width	E1	.240	.250	.260	6.10	6.35	6.60
Overall Length	D	.890	.898	.905	22.61	22.80	22.99
Tip to Seating Plane	L	.125	.130	.135	3.18	3.30	3.43
Lead Thickness	c	.008	.012	.015	0.20	0.29	0.38
Upper Lead Width	B1	.045	.058	.070	1.14	1.46	1.78
Lower Lead Width	B	.014	.018	.022	0.36	0.46	0.56
Overall Row Spacing	§ eB	.310	.370	.430	7.87	9.40	10.92
Mold Draft Angle Top	α	5	10	15	5	10	15
Mold Draft Angle Bottom	β	5	10	15	5	10	15

* Controlling Parameter
§ Significant Characteristic

Notes:
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.
JEDEC Equivalent: MS-001
Drawing No. C04-007