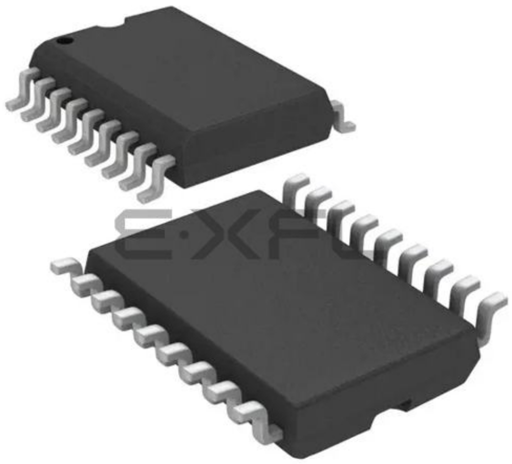


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

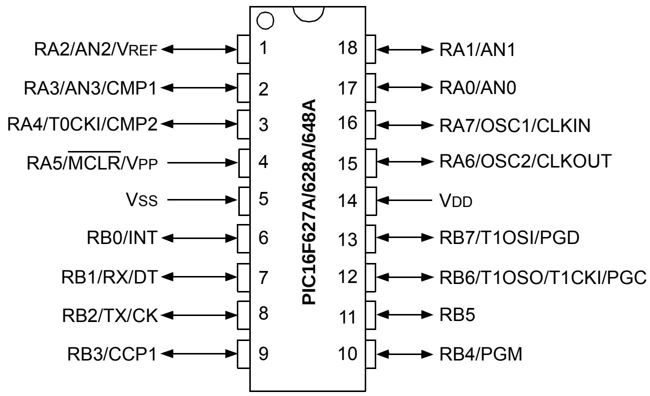
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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"

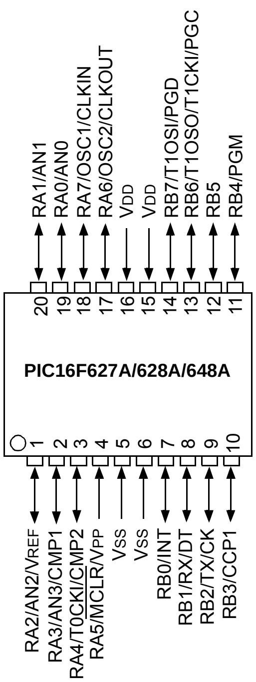
Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	20MHz
Connectivity	UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	16
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	18-SOIC (0.295", 7.50mm Width)
Supplier Device Package	18-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf648at-i-so

Pin Diagrams

PDIP, SOIC



SSOP



28-Pin QFN

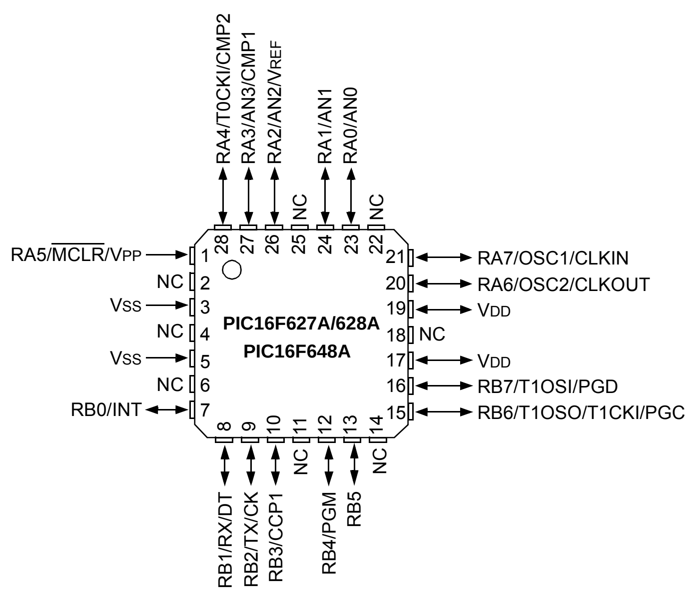


Table of Contents

TO OUR VALUED CUSTOMERS

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

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@microchip.com** or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. We welcome your feedback.

Most Current Data Sheet

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

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

Errata

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

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

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

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

Customer Notification System

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

NOTES:

FIGURE 4-3: DATA MEMORY MAP OF THE PIC16F648A

File Address	Bank 0	File Address	Bank 1	File Address	Bank 2	File Address	Bank 3
00h	Indirect addr.(1)	80h	Indirect addr.(1)	100h	Indirect addr.(1)	180h	Indirect addr.(1)
01h	TMR0	81h	OPTION	101h	TMR0	181h	OPTION
02h	PCL	82h	PCL	102h	PCL	182h	PCL
03h	STATUS	83h	STATUS	103h	STATUS	183h	STATUS
04h	FSR	84h	FSR	104h	FSR	184h	FSR
05h	PORTA	85h	TRISA	105h		185h	
06h	PORTB	86h	TRISB	106h	PORTB	186h	TRISB
07h		87h		107h		187h	
08h		88h		108h		188h	
09h		89h		109h		189h	
0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah	PCLATH
0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh	INTCON
0Ch	PIR1	8Ch	PIE1	10Ch		18Ch	
0Dh		8Dh		10Dh		18Dh	
0Eh	TMR1L	8Eh	PCON	10Eh		18Eh	
0Fh	TMR1H	8Fh		10Fh		18Fh	
10h	T1CON	90h					
11h	TMR2	91h					
12h	T2CON	92h	PR2				
13h		93h					
14h		94h					
15h	CCPR1L	95h					
16h	CCPR1H	96h					
17h	CCP1CON	97h					
18h	RCSTA	98h	TXSTA				
19h	TXREG	99h	SPBRG				
1Ah	RCREG	9Ah	EEDATA				
1Bh		9Bh	EEADR				
1Ch		9Ch	EECON1				
1Dh		9Dh	EECON2(1)				
1Eh		9Eh					
1Fh	CMCON	9Fh	VRCON	11Fh			
20h–6Fh	General Purpose Register 80 Bytes	A0h–EFh	General Purpose Register 80 Bytes	120h–16Fh	General Purpose Register 80 Bytes	1EFh	
70h–7Fh	16 Bytes	F0h–FFh	accesses 70h-7Fh	170h–17Fh	accesses 70h-7Fh	1F0h–1FFh	accesses 70h-7Fh

Shaded (blank) cells: Unimplemented data memory locations, read as ‘0’.

Note 1: Not a physical register.

4.2.2.2 OPTION Register

The Option register is a readable and writable register, which contains various control bits to configure the TMR0/WDT prescaler, the external RB0/INT interrupt, TMR0 and the weak pull-ups on PORTB.

Note:	To achieve a 1:1 prescaler assignment for TMR0, assign the prescaler to the WDT (PSA = 1). See **Section 6.3.1 “Switching Prescaler Assignment”**.

REGISTER 4-2: OPTION_REG – OPTION REGISTER (ADDRESS: 81h, 181h)

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

bit 7 **$\overline{\text{RBPU}}$**: 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**: TMR0 Clock Source Select bit
1 = Transition on RA4/T0CKI/CMP2 pin
0 = Internal instruction cycle clock (CLKOUT)

bit 4 **T0SE**: TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on RA4/T0CKI/CMP2 pin
0 = Increment on low-to-high transition on RA4/T0CKI/CMP2 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

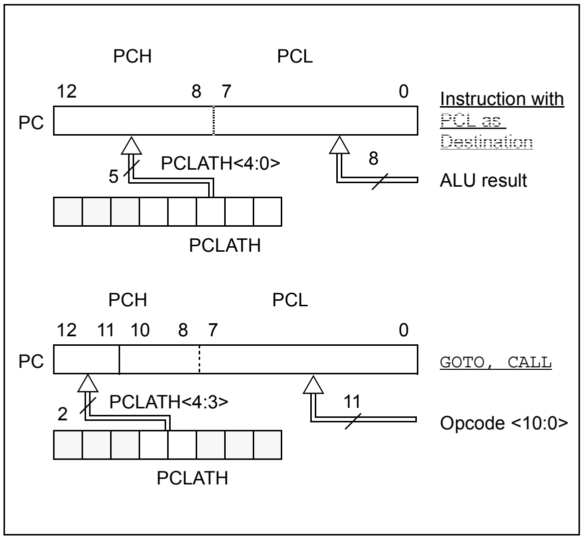
Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

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

4.3 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 4-4 shows the two situations for loading the PC. The upper example in Figure 4-4 shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in Figure 4-4 shows how the PC is loaded during a `CALL` or `GOTO` instruction (PCLATH<4:3> → PCH).

FIGURE 4-4: LOADING OF PC IN DIFFERENT SITUATIONS



4.3.1 COMPUTED `GOTO`

A computed `GOTO` is accomplished by adding an offset to the program counter (`ADDWF PCL`). When doing a table read using a computed `GOTO` method, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block). Refer to the Application Note AN556 "*Implementing a Table Read*" (DS00556).

4.3.2 STACK

The PIC16F627A/628A/648A family has an 8-level deep x 13-bit wide hardware stack (Figure 4-1). 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.

4.4 Indirect Addressing, INDF and FSR Registers

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

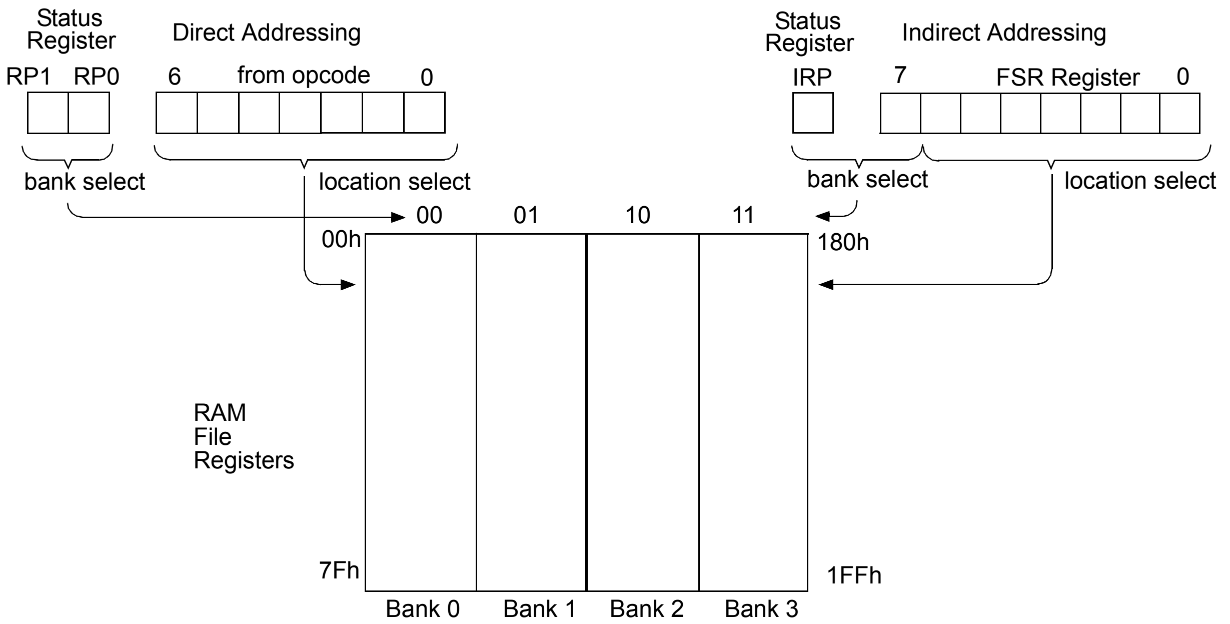
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select Register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no-operation (although Status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in Figure 4-5.

A simple program to clear RAM location 20h-2Fh using indirect addressing is shown in Example 4-1.

EXAMPLE 4-1: INDIRECT ADDRESSING

```
        MOVLW   0x20    ;initialize pointer
        MOVWF   FSR     ;to RAM
NEXT    CLRF    INDF    ;clear INDF register
        INCF    FSR     ;inc pointer
        BTFSS   FSR,4   ;all done?
        GOTO    NEXT    ;no clear next
                        ;yes continue
```

FIGURE 4-5: DIRECT/INDIRECT ADDRESSING PIC16F627A/628A/648A



Note: For memory map detail see Figure 4-3, Figure 4-2 and Figure 4-1.

FIGURE 5-2: BLOCK DIAGRAM OF RA2/AN2/VREF PIN

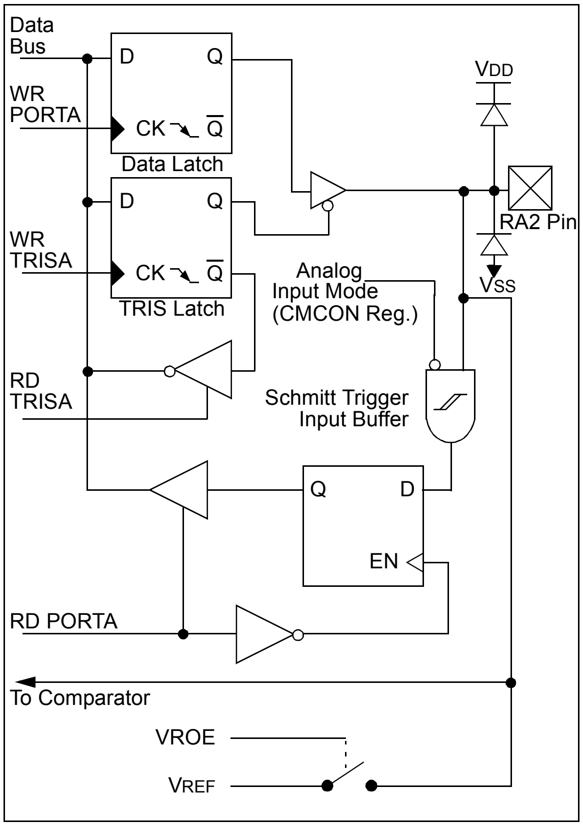


FIGURE 5-3: BLOCK DIAGRAM OF THE RA3/AN3/CMP1 PIN

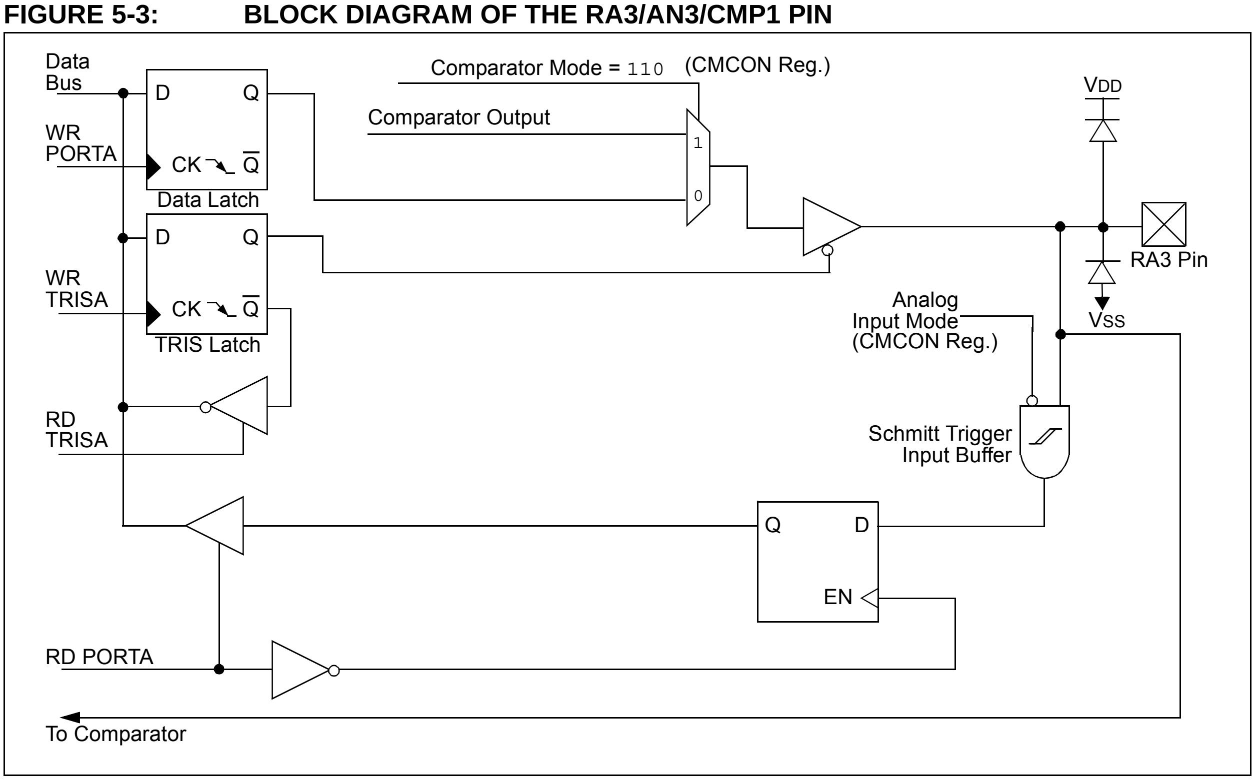
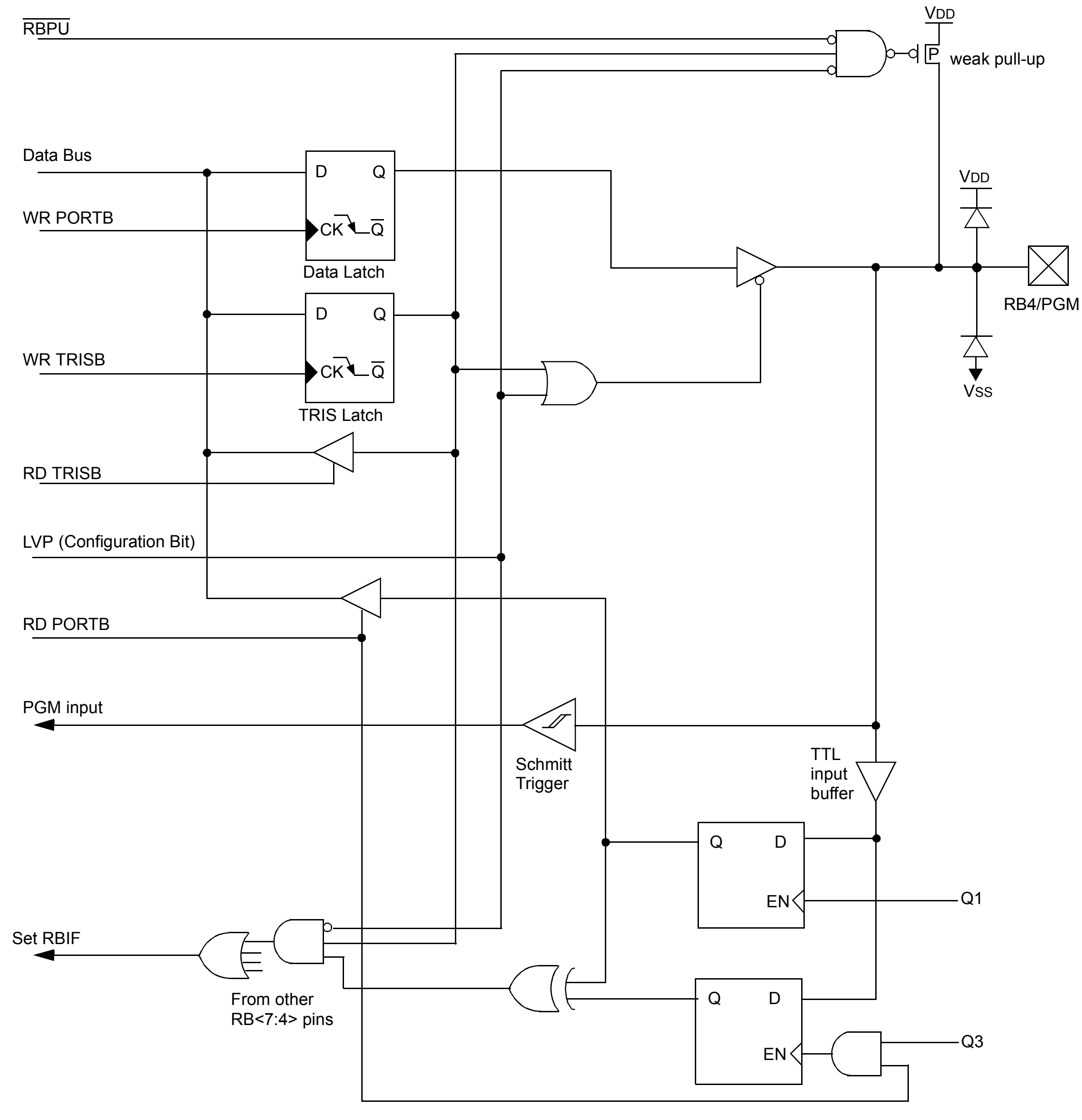


FIGURE 5-12: BLOCK DIAGRAM OF RB4/PGM PIN



Note: The low-voltage programming disables the interrupt-on-change and the weak pull-ups on RB4.

10.6 Comparator Interrupts

The comparator interrupt flag is set whenever there is a change in the output value of either comparator. Software will need to maintain information about the status of the output bits, as read from CMCON<7:6>, to determine the actual change that has occurred. The CMIF bit, PIR1<6>, is the comparator interrupt flag. The CMIF bit must be reset by clearing '0'. Since it is also possible to write a '1' to this register, a simulated interrupt may be initiated.

The CMIE bit (PIE1<6>) and the PEIE bit (INTCON<6>) must be set to enable the interrupt. In addition, the GIE bit must also be set. If any of these bits are clear, the interrupt is not enabled, though the CMIF bit will still be set if an interrupt condition occurs.

Note:	If a change in the CMCON register (C1OUT or C2OUT) should occur when a read operation is being executed (start of the Q2 cycle), then the CMIF (PIR1<6>) interrupt flag may not get set.

The user, in the interrupt service routine, can clear the interrupt in the following manner:

a) Any write or read of CMCON. This will end the mismatch condition.

b) Clear flag bit CMIF.

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

10.7 Comparator Operation During Sleep

When a comparator is active and the device is placed in Sleep mode, the comparator remains active and the interrupt is functional if enabled. This interrupt will wake-up the device from Sleep mode when enabled. While the comparator is powered-up, higher Sleep currents than shown in the power-down current specification will occur. Each comparator that is operational will consume additional current as shown in the comparator specifications. To minimize power consumption while in Sleep mode, turn off the comparators, CM<2:0> = `111`, before entering Sleep. If the device wakes up from Sleep, the contents of the CMCON register are not affected.

10.8 Effects of a Reset

A device Reset forces the CMCON register to its Reset state. This forces the Comparator module to be in the comparator Reset mode, CM<2:0> = `000`. This ensures that all potential inputs are analog inputs. Device current is minimized when analog inputs are present at Reset time. The comparators will be powered-down during the Reset interval.

10.9 Analog Input Connection Considerations

A simplified circuit for an analog input is shown in Figure 10-4. Since the analog pins are connected to a digital output, they have reverse biased diodes to V_{DD} and V_{SS}. The analog input therefore, must be between V_{SS} and V_{DD}. If the input voltage deviates from this range by more than 0.6V in either direction, one of the diodes is forward biased and a latch-up may occur. A maximum source impedance of 10 kΩ is recommended for the analog sources. Any external component connected to an analog input pin, such as a capacitor or a Zener diode, should have very little leakage current.

REGISTER 14-1: CONFIG – CONFIGURATION WORD REGISTER

CP	—	—	—	—	CPD	LVP	BOREN	MCLRE	FOSC2	PWRTE	WDTE	F0SC1	F0SC0
bit 13													bit 0

bit 13: **CP:** Flash Program Memory Code Protection bit[2]
(PIC16F648A)
1 = Code protection off
0 = 0000h to 0FFFh code-protected
(PIC16F628A)
1 = Code protection off
0 = 0000h to 07FFh code-protected
(PIC16F627A)
1 = Code protection off
0 = 0000h to 03FFh code-protected

bit 12-9: **Unimplemented**: Read as '0'

bit 8: **CPD:** Data Code Protection bit[3]
1 = Data memory code protection off
0 = Data memory code-protected

bit 7: **LVP**: Low-Voltage Programming Enable bit
1 = RB4/PGM pin has PGM function, low-voltage programming enabled
0 = RB4/PGM is digital I/O, HV on MCLR must be used for programming

bit 6: **BOREN**: Brown-out Reset Enable bit [1]
1 = BOR Reset enabled
0 = BOR Reset disabled

bit 5: **MCLRE**: RA5/MCLR/VPP Pin Function Select bit
1 = RA5/MCLR/VPP pin function is MCLR
0 = RA5/MCLR/VPP pin function is digital Input, MCLR internally tied to VDD

bit 3: **PWRTE**: Power-up Timer Enable bit [1]
1 = PWRT disabled
0 = PWRT enabled

bit 2: **WDTE**: Watchdog Timer Enable bit
1 = WDT enabled
0 = WDT disabled

bit 4, 1-0: **FOSC<2:0>**: Oscillator Selection bits[4]
111 = RC oscillator: CLKOUT function on RA6/OSC2/CLKOUT pin, Resistor and Capacitor on RA7/OSC1/CLKIN
110 = RC oscillator: I/O function on RA6/OSC2/CLKOUT pin, Resistor and Capacitor on RA7/OSC1/CLKIN
101 = INTOSC oscillator: CLKOUT function on RA6/OSC2/CLKOUT pin, I/O function on RA7/OSC1/CLKIN
100 = INTOSC oscillator: I/O function on RA6/OSC2/CLKOUT pin, I/O function on RA7/OSC1/CLKIN
011 = EC: I/O function on RA6/OSC2/CLKOUT pin, CLKIN on RA7/OSC1/CLKIN
010 = HS oscillator: High-speed crystal/resonator on RA6/OSC2/CLKOUT and RA7/OSC1/CLKIN
001 = XT oscillator: Crystal/resonator on RA6/OSC2/CLKOUT and RA7/OSC1/CLKIN
000 = LP oscillator: Low-power crystal on RA6/OSC2/CLKOUT and RA7/OSC1/CLKIN

Note 1: Enabling Brown-out Reset does not automatically enable the Power-up Timer (PWRT) the way it does on the PIC16F627/628 devices.
2: The code protection scheme has changed from the code protection scheme used on the PIC16F627/628 devices. The entire Flash program memory needs to be bulk erased to set the CP bit, turning the code protection off. See "*PIC16F627A/628A/648A EEPROM Memory Programming Specification*" (DS41196) for details.
3: The entire data EEPROM needs to be bulk erased to set the CPD bit, turning the code protection off. See "*PIC16F627A/628A/648A EEPROM Memory Programming Specification*" (DS41196) for details.
4: When MCLR is asserted in INTOSC mode, the internal clock oscillator is disabled.

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

CLRW — **Clear W**

Syntax:	[*label*] CLRW
Operands:	None
Operation:	00h → (W) 1 → Z
Status Affected:	Z
Encoding:	00 | 0001 | 0000 | 0011
Description:	W register is cleared. Zero bit (Z) is set.
Words:	1
Cycles:	1
Example	`CLRW` Before Instruction W = 0x5A After Instruction W = 0x00 Z = 1

CLRWDT — **Clear Watchdog Timer**

Syntax:	[*label*] CLRWDT
Operands:	None
Operation:	00h → WDT 0 → WDT prescaler, 1 → $\overline{TO}$ 1 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Encoding:	00 | 0000 | 0110 | 0100
Description:	`CLRWDT` instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.
Words:	1
Cycles:	1
Example	`CLRWDT` Before Instruction WDT counter = ? After Instruction WDT counter = 0x00 WDT prescaler = 0 $\overline{TO}$ = 1 $\overline{PD}$ = 1

COMF — **Complement f**

Syntax:	[*label*] COMF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	$(\bar{f}) \rightarrow$ (dest)
Status Affected:	Z
Encoding:	00 | 1001 | dfff | ffff
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’.
Words:	1
Cycles:	1
Example	`COMF REG1, 0` Before Instruction REG1 = 0x13 After Instruction REG1 = 0x13 W = 0xEC

DECF — **Decrement f**

Syntax:	[*label*] DECF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) - 1 → (dest)
Status Affected:	Z
Encoding:	00 | 0011 | dfff | ffff
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’.
Words:	1
Cycles:	1
Example	`DECF CNT, 1` Before Instruction CNT = 0x01 Z = 0 After Instruction CNT = 0x00 Z = 1

RRF — Rotate Right f through Carry

Syntax:	[*label*] RRF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	See description below
Status Affected:	C
Encoding:	`00` | `1100` | `dfff` | `ffff`
Description:	The contents of register 'f' are rotated one bit to the right through the Carry Flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

Words:	1
Cycles:	1
Example	`RRF REG1, 0`

Before Instruction
- REG1 = `1110 0110`
- C = `0`

After Instruction
- REG1 = `1110 0110`
- W = `0111 0011`
- C = `0`

SLEEP

Syntax:	[*label*] SLEEP
Operands:	None
Operation:	00h → WDT, 0 → WDT prescaler, 1 → $\overline{TO}$, 0 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Encoding:	`00` | `0000` | `0110` | `0011`
Description:	The power-down Status bit, $\overline{PD}$ is cleared. Time out Status bit, $\overline{TO}$ is set. Watchdog Timer and its prescaler are cleared. The processor is put into Sleep mode with the oscillator stopped. See **Section 14.8 "Power-Down Mode (Sleep)"** for more details.
Words:	1
Cycles:	1
Example:	`SLEEP`

SUBLW — Subtract W from Literal

Syntax:	[*label*] SUBLW k
Operands:	$0 \le k \le 255$
Operation:	k - (W) → (W)
Status Affected:	C, DC, Z
Encoding:	`11` | `110x` | `kkkk` | `kkkk`
Description:	The W register is subtracted (2's complement method) from the eight-bit literal 'k'. The result is placed in the W register.
Words:	1
Cycles:	1
Example 1:	`SUBLW 0x02`

Example 1:

Before Instruction
- W = `1`
- C = ?

After Instruction
- W = `1`
- C = `1`; result is positive

Example 2:

Before Instruction
- W = 2
- C = ?

After Instruction
- W = `0`
- C = `1`; result is zero

Example 3:

Before Instruction
- W = 3
- C = ?

After Instruction
- W = 0xFF
- C = `0`; result is negative

16.7 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 C Compilers, and the MPASM and MPLAB 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.

16.8 MPLAB REAL ICE In-Circuit Emulator System

MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs PIC® Flash MCUs and dsPIC® Flash DSCs with the easy-to-use, powerful graphical user interface of the MPLAB Integrated Development Environment (IDE), included with each kit.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB IDE. In upcoming releases of MPLAB IDE, new devices will be supported, and new features will be added. MPLAB REAL ICE offers significant advantages over competitive emulators including low-cost, full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, a ruggedized probe interface and long (up to three meters) interconnection cables.

16.9 MPLAB ICD 3 In-Circuit Debugger System

MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost effective high-speed hardware debugger/programmer for Microchip Flash Digital Signal Controller (DSC) and microcontroller (MCU) devices. It debugs and programs PIC® Flash microcontrollers and dsPIC® DSCs with the powerful, yet easy-to-use graphical user interface of MPLAB Integrated Development Environment (IDE).

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

16.10 PICkit 3 In-Circuit Debugger/ Programmer and PICkit 3 Debug Express

The MPLAB PICkit 3 allows debugging and programming of PIC® and dsPIC® Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB Integrated Development Environment (IDE). The MPLAB PICkit 3 is connected to the design engineer's PC using a full speed USB interface and can be connected to the target via an Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the reset line to implement in-circuit debugging and In-Circuit Serial Programming™.

The PICkit 3 Debug Express include the PICkit 3, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

FIGURE 18-2: TYPICAL BASELINE IPD vs. VDD (85°C)

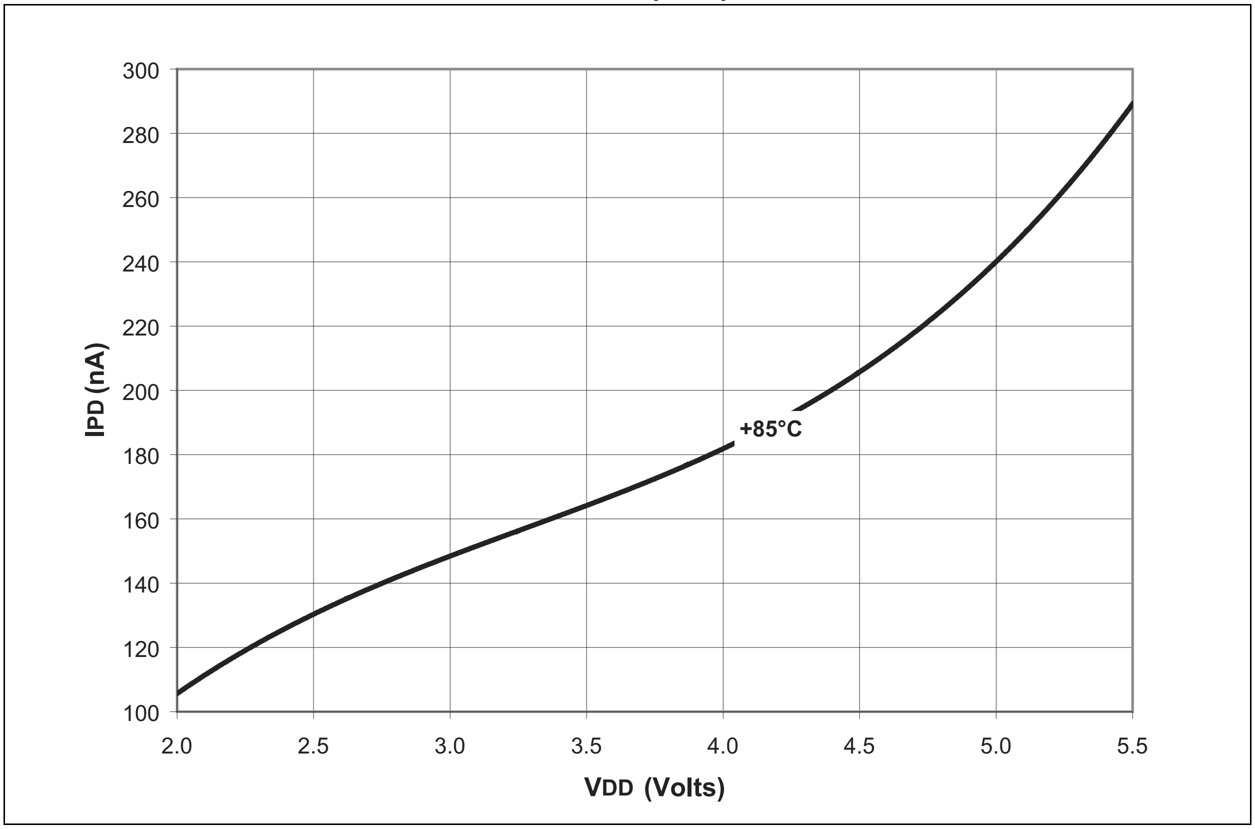


FIGURE 18-3: TYPICAL BASELINE CURRENT IPD vs. VDD (125°C)

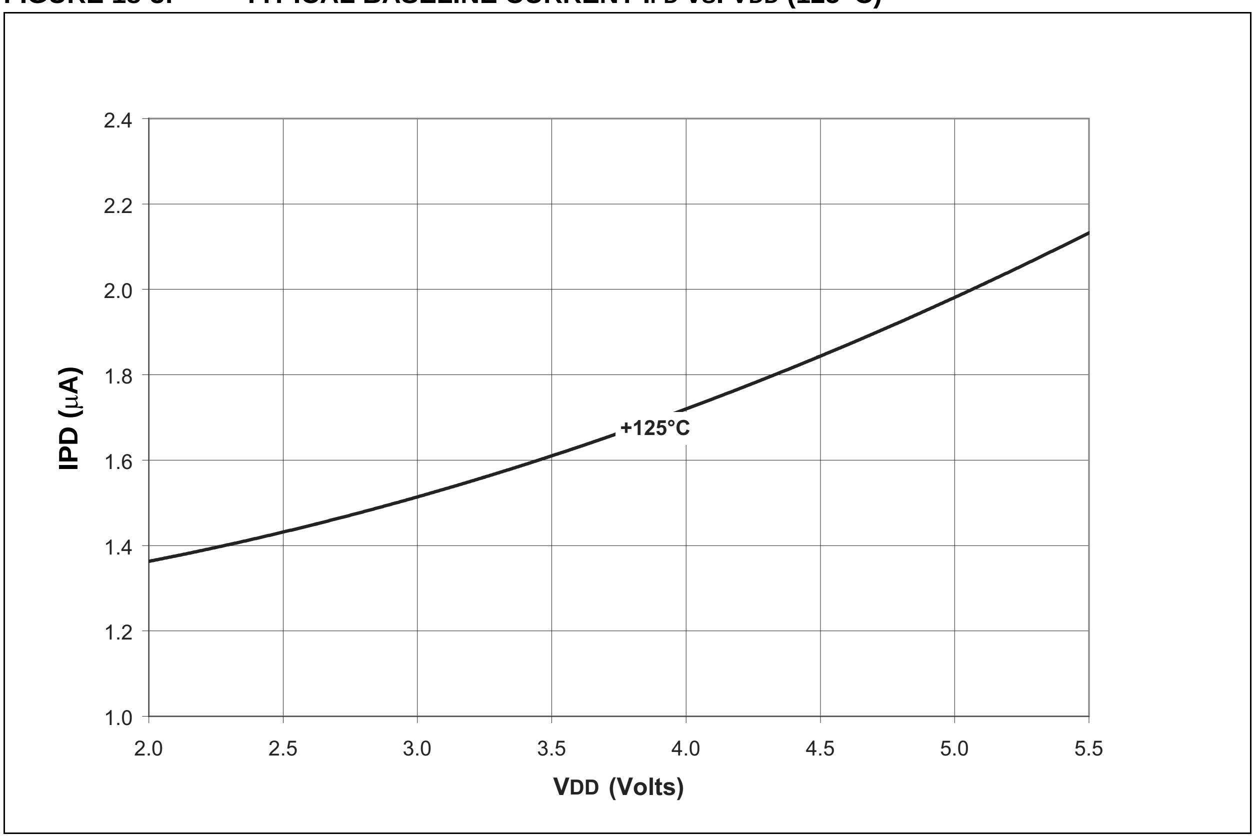


FIGURE 18-12: TYPICAL INTERNAL OSCILLATOR DEVIATION vs. VDD AT 25°C – 4 MHz MODE

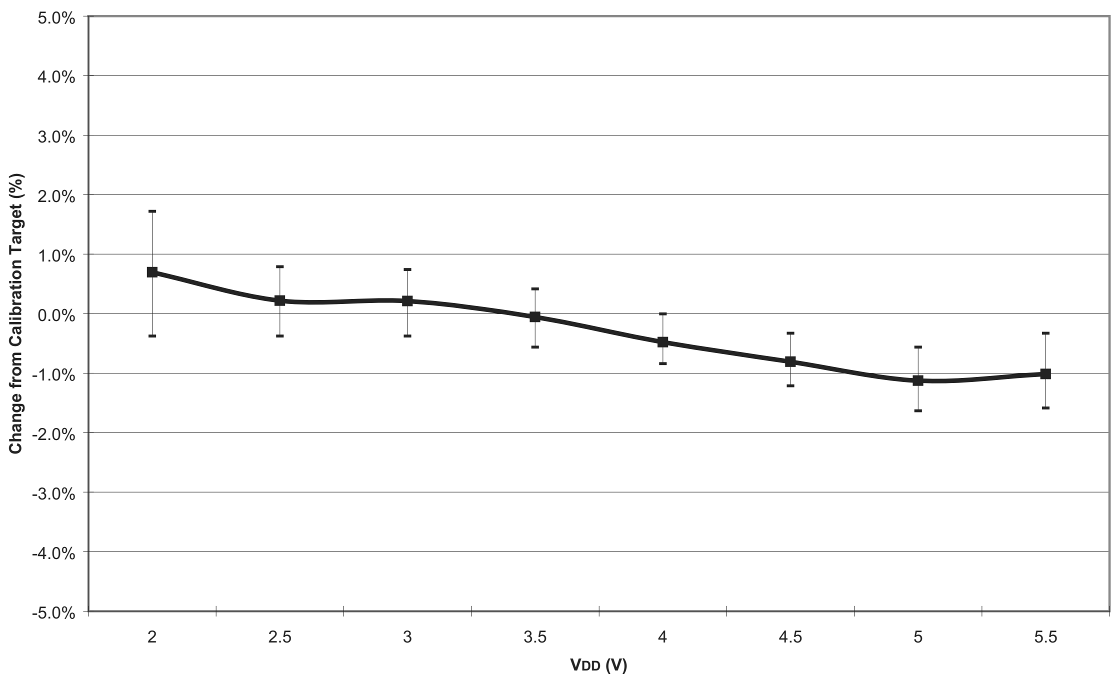


FIGURE 18-13: TYPICAL INTERNAL OSCILLATOR FREQUENCY vs. VDD TEMPERATURE = -40°C TO 85°C

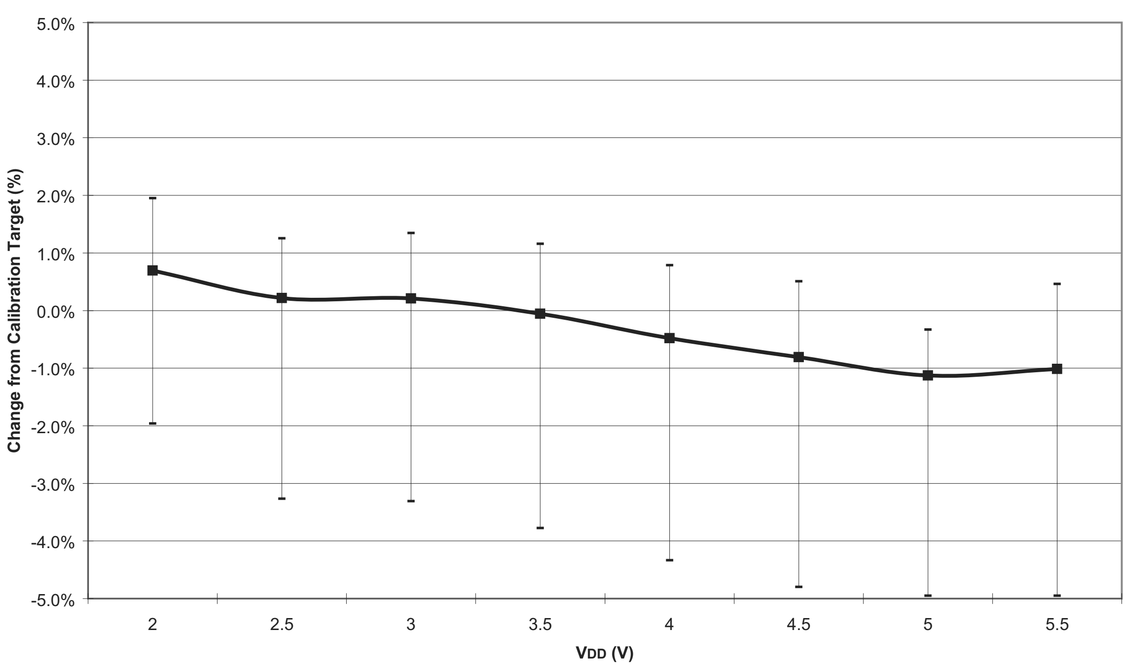
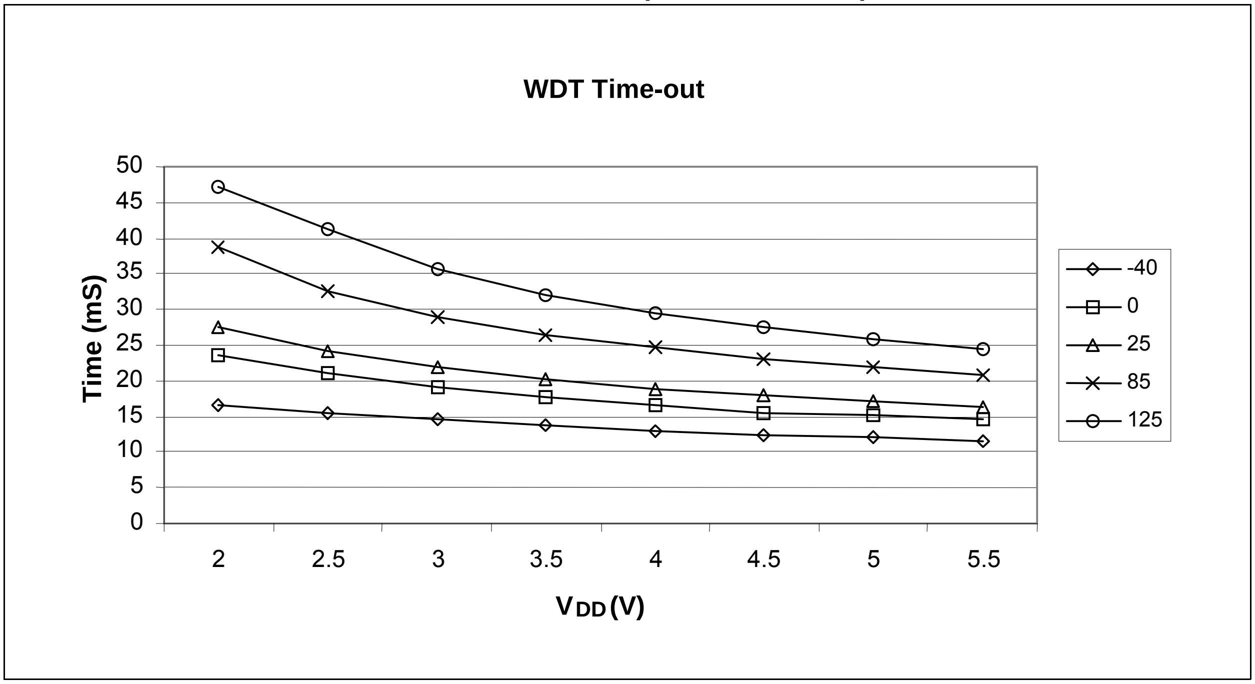
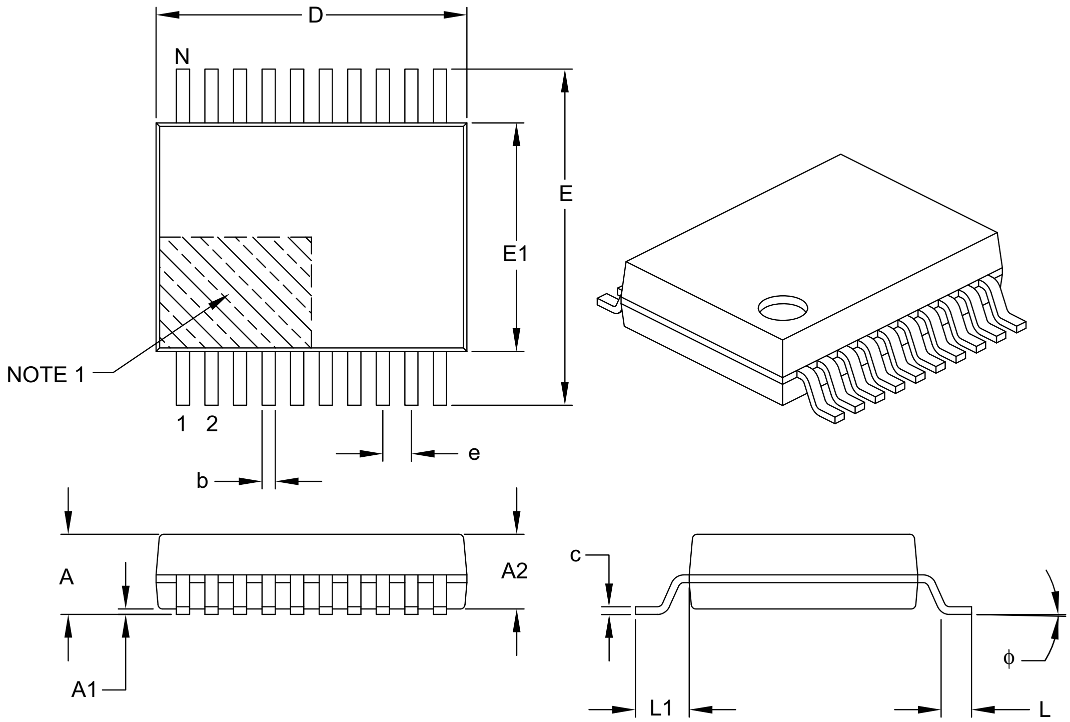


FIGURE 18-20: TYPICAL WDT PERIOD vs. VDD (-40°C TO +125°C)



20-Lead Plastic Shrink Small Outline (SS) – 5.30 mm Body [SSOP]

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		20	
Pitch	e		0.65 BSC	
Overall Height	A	–	–	2.00
Molded Package Thickness	A2	1.65	1.75	1.85
Standoff	A1	0.05	–	–
Overall Width	E	7.40	7.80	8.20
Molded Package Width	E1	5.00	5.30	5.60
Overall Length	D	6.90	7.20	7.50
Foot Length	L	0.55	0.75	0.95
Footprint	L1		1.25 REF	
Lead Thickness	c	0.09	–	0.25
Foot Angle	φ	0°	4°	8°
Lead Width	b	0.22	–	0.38

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.20 mm per side.
3. 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 C04-072B

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

PART NO.	X	/XX	XXX
Device	Temperature Range	Package	Pattern

Device:	PIC16F627A/628A/648A:Standard V_{DD} range 3.0V to 5.5V PIC16F627A/628A/648AT:V_{DD} range 3.0V to 5.5V (Tape and Reel) PIC16LF627A/628A/648A:V_{DD} range 2.0V to 5.5V PIC16LF627A/628A/648AT:V_{DD} range 2.0V to 5.5V (Tape and Reel)
Temperature Range:	I = -40°C to +85°C E = -40°C to+125°C
Package:	P = PDIP SO = SOIC (Gull Wing, 7.50 mm body) SS = SSOP (5.30 mm ML = QFN (28 Lead)

Examples:

a) PIC16F627A - E/P 301 = Extended Temp., PDIP package, 20 MHz, normal V_{DD} limits, QTP pattern #301.

b) PIC16LF627A - I/SO = Industrial Temp., SOIC package, 20 MHz, extended V_{DD} limits.