PIC16F527

20-Pin, 8-Bit Flash Microcontroller

Processor Features:

- Interrupt Capability
- PIC16F527 Operating Speed:
  - DC – 20 MHz Crystal oscillator
  - DC – 200 ns Instruction cycle
- Flash Program Memory:
  - 1024 x 12 user execution memory
  - 64 x 8 self-writable data memory
  - 10K minimum erase/write cycles
- General Purpose Registers (SRAM):
  - 68 x 8 for PIC16F527
- Only 36 Single-Word Instructions to Learn:
  - Added `RETURN` and `RETFIE` instructions
  - Added `MOVLB` instruction
- All Instructions are Single-Cycle except for Program Branches which are Two-Cycle
- Four-Level Deep Hardware Stack
- Direct, Indirect and Relative Addressing modes for Data and Instructions

Peripheral Features:

- Device Features:
  - 1 Input-only pin
  - 17 I/Os
  - Individual direction control
  - High-current source/sink
- 8-Bit Real-Time Clock/Counter (TMR0) with 8-Bit Programmable Prescaler
- In-Circuit Serial Programming™ (ICSP™) via Two External Pin Connections
- Analog Comparator (CMP):
  - Two analog comparators
  - Absolute and programmable references
- Analog-to-Digital Converter (ADC):
  - 8-bit resolution
  - 8 external input channels
  - 1 internal channel to convert comparator
  - 0.6V reference input
- Operational Amplifiers (op amps):
  - 2 operational amplifiers
  - Fully-accessible visibility

Microcontroller Features:

- Brown-out Reset (BOR)
- Power-on Reset (POR)
- Device Reset Timer (DRT)
- Watchdog Timer (WDT) with a Dedicated RC Oscillator
- Programmable Code Protection (CP)
- Power-Saving Sleep mode with Wake-up on Change Feature
- Selectable Oscillator Options:
  - INTOSC: Precision 4 or 8 MHz internal oscillator
  - EXTRC: Low-cost external RC oscillator
  - LP: Power-saving, low-frequency crystal
  - XT: Standard crystal/resonator
  - HS: High-speed crystal/resonator
  - EC: High-speed external clock
- Variety of Packaging Options:
  - 20-Lead PDIP, SOIC, SSOP, QFN

CMOS Technology:

- Low-Power, High-Speed CMOS Flash Technology
- Fully-Static Design
- Wide Operating Voltage and Temperature Range:
  - Industrial: 2.0V to 5.5V
  - Extended: 2.0V to 5.5V
- Operating Current:
  - 170 uA @ 2V, 4 MHz, typical – 15 uA @ 2V, 32 kHz, typical
- Standby Current:
  - 100 nA @ 2V, typical

Device	Program Memory	Data Memory		I/O	Comparators	Timers 8-bit	8-bit A/D Channels	Op Amps
	Flash (words)	SRAM (bytes)	Flash (bytes)					
PIC16F527	1024	68	64	18	2	1	8	2

FIGURE 1: 20-PIN PDIP, SOIC, SSOP DIAGRAM FOR PIC16F527

20-pin PDIP, SSOP, SOIC

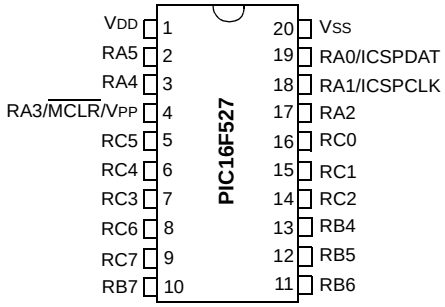


FIGURE 2: 20-PIN QFN DIAGRAM FOR PIC16F527

20-pin QFN

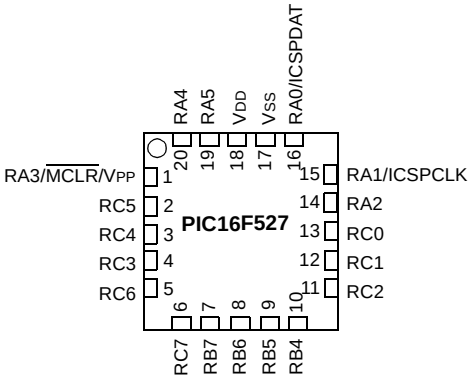


TABLE 1: 20-PIN ALLOCATION TABLE

I/O	20-Pin PDIP/SOIC/SSOP	20-Pin QFN	Analog	Oscillator	Comparator	Reference	Timers	Op Amp	Clock Reference	ICSP™	Basic	Pull-up	Interrupt-on-Change
RA0	19	16	AN0	—	C1IN+	—	—	—	—	ICSPDAT	—	Y	Y
RA1	18	15	AN1	—	C1IN-	CV_{REF}	—	—	—	ICSPCLK	—	Y	Y
RA2	17	14	AN2	—	C1OUT	—	T0CKI	—	—	—	—	—	—
RA3	4	1	—	—	—	—	—	—	—	—	$\overline{MCLR}$ V_{PP}	Y	Y
RA4	3	20	AN3	OSC2	—	—	—	—	CLKOUT	—	—	Y	Y
RA5	2	19	—	OSC1	—	—	—	—	CLKIN	—	—	—	—
RB4	13	10	—	—	—	—	—	OP2-	—	—	—	—	—
RB5	12	9	—	—	—	—	—	OP2+	—	—	—	—	—
RB6	11	8	—	—	—	—	—	—	—	—	—	—	—
RB7	10	7	—	—	—	—	—	—	—	—	—	—	—
RC0	16	13	AN4	—	C2IN+	—	—	—	—	—	—	—	—
RC1	15	12	AN5	—	C2IN-	—	—	—	—	—	—	—	—
RC2	14	11	AN6	—	—	—	—	OP2	—	—	—	—	—
RC3	7	4	AN7	—	—	—	—	OP1	—	—	—	—	—
RC4	6	3	—	—	C2OUT	—	—	—	—	—	—	—	—
RC5	5	2	—	—	—	—	—	—	—	—	—	—	—
RC6	8	5	—	—	—	—	—	OP1-	—	—	—	—	—
RC7	9	6	—	—	—	—	—	OP1+	—	—	—	—	—
V_{DD}	1	18	—	—	—	—	—	—	—	—	—	—	—
V_{SS}	20	17	—	—	—	—	—	—	—	—	—	—	—

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.

1.0 GENERAL DESCRIPTION

The PIC16F527 device from Microchip Technology is a low-cost, high-performance, 8-bit, fully-static, Flash-based CMOS microcontroller. It employs a RISC architecture with only 36 single-word/single-cycle instructions. All instructions are single cycle except for program branches, which take two cycles. The PIC16F527 device delivers performance an order of magnitude higher than its competitors in the same price category. The 12-bit wide instructions are highly symmetrical, resulting in a typical 2:1 code compression over other 8-bit microcontrollers in its class. The easy-to-use and easy to remember instruction set reduces development time significantly.

The PIC16F527 product is equipped with special features that reduce system cost and power requirements. The Power-on Reset (POR) and Device Reset Timer (DRT) eliminate the need for external Reset circuitry. There are several oscillator configurations to choose from, including INTRC Internal Oscillator mode and the power-saving LP (Low-Power) Oscillator mode. Power-Saving Sleep mode, Watchdog Timer and code protection features improve system cost, power and reliability.

The PIC16F527 device is available in the cost-effective Flash programmable version, which is suitable for production in any volume. The customer can take full advantage of Microchip's price leadership in Flash programmable microcontrollers, while benefiting from the Flash programmable flexibility.

The PIC16F527 product is supported by a full-featured macro assembler, a software simulator, an in-circuit emulator, a 'C' compiler, a low-cost development programmer and a full-featured programmer. All the tools are supported on IBM® PC and compatible machines.

1.1 Applications

The PIC16F527 device fits in applications ranging from personal care appliances and security systems to low-power remote transmitters/receivers. The Flash technology makes customizing application programs (transmitter codes, appliance settings, receiver frequencies, etc.) extremely fast and convenient. The small footprint packages, for through hole or surface mounting, make these microcontrollers perfect for applications with space limitations. Low cost, low power, high performance, ease of use and I/O flexibility make the PIC16F527 device very versatile, even in areas where no microcontroller use has been considered before (e.g., timer functions, logic and PLDs in larger systems and coprocessor applications).

TABLE 1-1: FEATURES AND MEMORY OF PIC16F527

		PIC16F527
Clock	Maximum Frequency of Operation (MHz)	20
Memory	Flash Program Memory	1024
	SRAM Data Memory (bytes)	68
	Flash Data Memory (bytes)	64
Peripherals	Timer Module(s)	TMR0
	Wake-up from Sleep on Pin Change	Yes
Features	I/O Pins	17
	Input Pins	1
	Internal Pull-ups	Yes
	In-Circuit Serial Programming™	Yes
	Number of Instructions	36
	Packages	20-pin PDIP, SOIC, SSOP, QFN

The PIC16F527 device has Power-on Reset, selectable Watchdog Timer, selectable code-protect, high I/O current capability and precision internal oscillator.
The PIC16F527 device uses serial programming with the ICSPDAT data pin and the ICSPCLK clock pin.

NOTES:

2.0 PIC16F527 DEVICE VARIETIES

A variety of packaging options are available. Depending on application and production requirements, the proper device option can be selected using the information in this section. When placing orders, please use the PIC16F527 Product Identification System at the back of this data sheet to specify the correct part number.

2.1 Quick Turn Programming (QTP) Devices

Microchip offers a QTP programming service for factory production orders. This service is made available for users who choose not to program medium-to-high quantity units and whose code patterns have stabilized. The devices are identical to the Flash devices but with all Flash locations and fuse options already programmed by the factory. Certain code and prototype verification procedures do apply before production shipments are available. Please contact your local Microchip Technology sales office for more details.

2.2 Serialized Quick Turn Programming^{SM} (SQTP^{SM}) Devices

Microchip offers a unique programming service, where a few user-defined locations in each device are programmed with different serial numbers. The serial numbers may be random, pseudo-random or sequential.

Serial programming allows each device to have a unique number, which can serve as an entry code, password or ID number.

NOTES:

3.0 ARCHITECTURAL OVERVIEW

The high performance of the PIC16F527 device can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC16F527 device uses a Harvard architecture in which program and data are accessed on separate buses. This improves bandwidth over traditional von Neumann architectures where program and data are fetched on the same bus. Separating program and data memory further allows instructions to be sized differently than the 8-bit wide data word. Instruction opcodes are 12 bits wide, making it possible to have all single-word instructions. A 12-bit wide program memory access bus fetches a 12-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions. Consequently, all instructions execute in a single cycle (200 ns @ 20 MHz, 1 μs @ 4 MHz) except for program branches.

Table 3-1 below lists memory supported by the PIC16F527 device.

TABLE 3-1: PIC16F527 MEMORY

Device	Program Memory	Data Memory	
	Flash (words)	SRAM (bytes)	Flash (bytes)
PIC16F527	1024	68	64

The PIC16F527 device can directly or indirectly address its register files and data memory. All Special Function Registers (SFR), including the PC, are mapped in the data memory. The PIC16F527 device has a highly orthogonal (symmetrical) instruction set that makes it possible to carry out any operation, on any register, using any Addressing mode. This symmetrical nature and lack of “special optimal situations” make programming with the PIC16F527 device simple, yet efficient. In addition, the learning curve is reduced significantly.

The PIC16F527 device contains an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between data in the working register and any register file.

The ALU is eight bits wide and capable of addition, subtraction, shift and logical operations. Unless otherwise mentioned, arithmetic operations are two’s complement in nature. In two-operand instructions, one operand is typically the W (working) register. The other operand is either a file register or an immediate constant. In single operand instructions, the operand is either the W register or a file register.

The W register is an 8-bit working register used for ALU operations. It is not an addressable register.

Depending on the instruction executed, the ALU may affect the values of the Carry (C), Digit Carry (DC) and Zero (Z) bits in the STATUS register. The C and DC bits operate as a borrow and digit borrow out bit, respectively, in subtraction. See the SUBWF and ADDWF instructions for examples.

A simplified block diagram is shown in Figure 3-2, with the corresponding device pins described in Table 3-2.

FIGURE 3-1: PIC16F527 BLOCK DIAGRAM

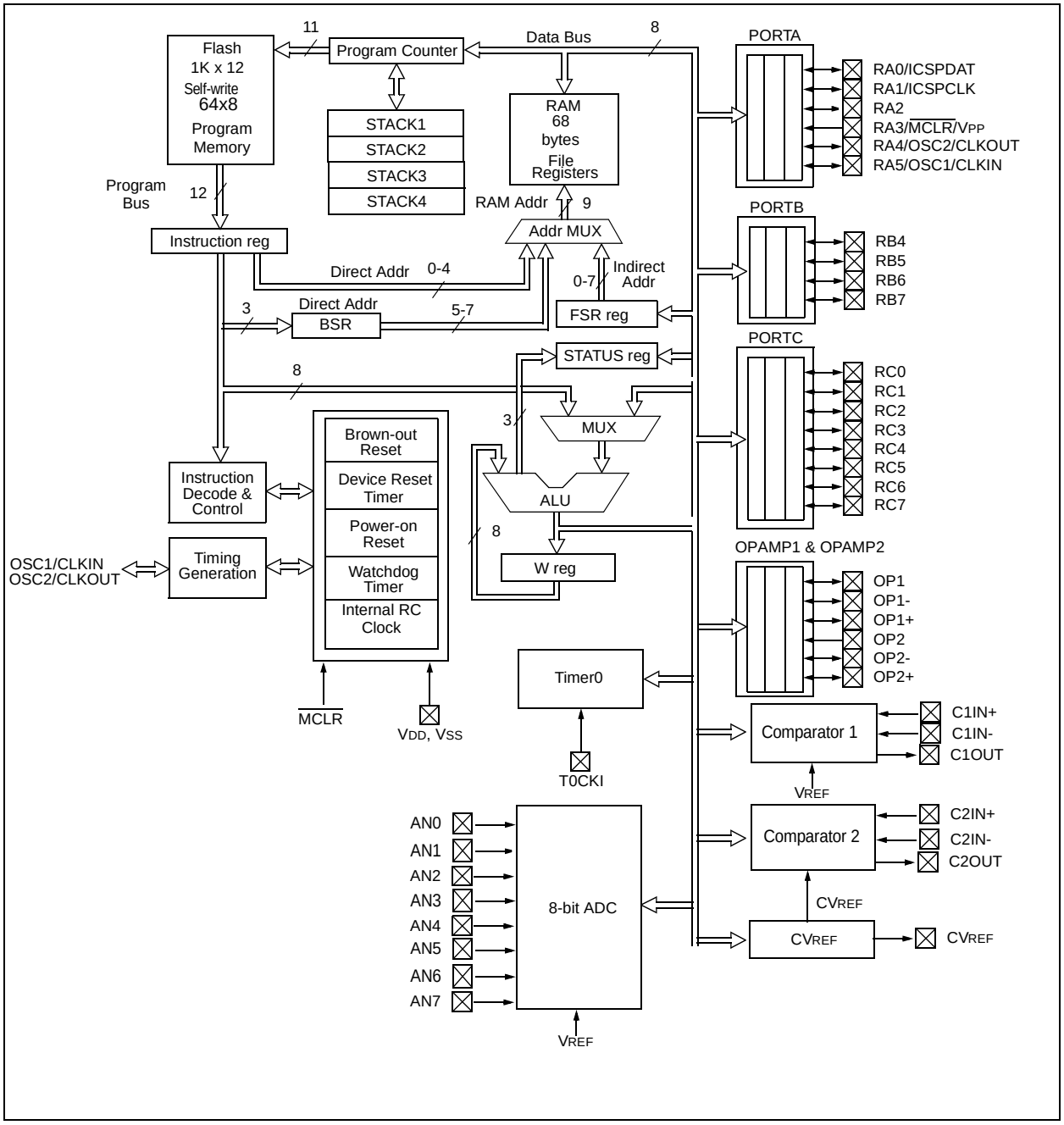


TABLE 3-2: PIC16F527 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/AN0/C1IN+/ICSPDAT	RA0	TTL	CMOS	Bidirectional I/O pin. It can be software programmed for internal weak pull-up and wake-up from Sleep on pin change.
	ICSPDAT	ST	CMOS	ICSP™ mode Schmitt Trigger.
	C1IN+	AN	—	Comparator 1 input.
	AN0	AN	—	ADC channel input.
RA1/AN1/C1IN-/CVREF/ ICSPCLK	RA1	TTL	CMOS	Bidirectional I/O pin. It can be software programmed for internal weak pull-up and wake-up from Sleep on pin change.
	ICSPCLK	ST	—	ICSP™ mode Schmitt Trigger.
	C1IN-	AN	—	Comparator 1 input.
	CVREF	—	AN	Programmable Voltage Reference output.
	AN1	AN	—	ADC channel input.
RA2/AN2/C1OUT/T0CKI	RA2	TTL	CMOS	Bidirectional I/O port.
	C1OUT	—	CMOS	Comparator 1 output.
	AN2	AN	—	ADC channel input.
	T0CKI	ST	—	Timer0 Schmitt Trigger input pin.
RA3/$\overline{\text{MCLR}}$/VPP	RA3	TTL	—	Standard TTL input with weak pull-up.
	$\overline{\text{MCLR}}$	ST	—	Master Clear (Reset). When configured as $\overline{\text{MCLR}}$, this pin is an active-low Reset to the device. Voltage on $\overline{\text{MCLR}}$/VPP must not exceed VDD during normal device operation or the device will enter Programming mode. Weak pull-up is always on if configured as $\overline{\text{MCLR}}$.
	VPP	HV	—	Test mode high-voltage pin.
RA4/AN3/OSC2/CLKOUT	RA4	TTL	CMOS	Bidirectional I/O pin. It can be software programmed for internal weak pull-up and wake-up from Sleep on pin change.
	OSC2	—	XTAL	Oscillator crystal output. Connections to crystal or resonator in Crystal Oscillator mode (XT, HS and LP modes only, PORTB in other modes).
	CLKOUT	—	CMOS	EXTRC/INTRC CLKOUT pin (FOSC/4).
	AN3	AN	—	ADC channel input.
RA5/OSC1/CLKIN	RA5	TTL	CMOS	Bidirectional I/O port.
	OSC1	XTAL	—	XTAL oscillator input pin.
	CLKIN	ST	—	EXTRC Schmitt Trigger input.
RB4/OP2-	RB4	TTL	CMOS	Bidirectional I/O port.
	OP2-	AN	—	Op amp 2 inverting input.
RB5/OP2+	RB5	TTL	CMOS	Bidirectional I/O port.
	OP2+	AN	—	Op amp 2 non-inverting input.
RB6	RB6	TTL	CMOS	Bidirectional I/O port.
RB7	RB7	TTL	CMOS	Bidirectional I/O port.
RC0/AN4/C2IN+	RC0	ST	CMOS	Bidirectional I/O port.
	AN4	AN	—	ADC channel input.
	C2IN+	AN	—	Comparator 2 input.

Legend: I = Input, O = Output, I/O = Input/Output, P = Power, — = Not Used, TTL = TTL input, ST = Schmitt Trigger input, HV = High Voltage, AN = Analog Voltage

TABLE 3-2: PIC16F527 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RC1/AN5/C2IN-	RC1	ST	CMOS	Bidirectional I/O port.
	AN5	AN	—	ADC channel input.
	C2IN-	AN	—	Comparator 2 input.
RC2/AN6/OP2	RC2	ST	CMOS	Bidirectional I/O port.
	AN6	AN	—	ADC channel input.
	OP2	—	AN	Op amp 2 output.
RC3/AN7/OP1	RC3	ST	CMOS	Bidirectional I/O port.
	AN7	AN	—	ADC channel input.
	OP1	—	AN	Op amp 1 output.
RC4/C2OUT	RC4	ST	CMOS	Bidirectional I/O port.
	C2OUT	—	CMOS	Comparator 2 output.
RC5	RC5	ST	CMOS	Bidirectional I/O port.
RC6/OP1-	RC6	ST	CMOS	Bidirectional I/O port.
	OP1-	AN	—	Op amp 1 inverting input.
RC7/OP1+	RC7	ST	CMOS	Bidirectional I/O port.
	OP1+	AN	—	Op amp 1 non-inverting input.
V_{DD}	V_{DD}	—	P	Positive supply for logic and I/O pins.
V_{SS}	V_{SS}	—	P	Ground reference for logic and I/O pins.

Legend: I = Input, O = Output, I/O = Input/Output, P = Power, — = Not Used, TTL = TTL input, ST = Schmitt Trigger input, HV = High Voltage, AN = Analog Voltage

3.1 Clocking Scheme/Instruction Cycle

The clock input (OSC1/CLKIN pin) is internally divided by four to generate four non-overlapping quadrature clocks, namely Q1, Q2, Q3 and Q4. Internally, the PC is incremented every Q1 and the instruction is fetched from program memory and latched into the instruction register in Q4. It is decoded and executed during the following Q1 through Q4. The clocks and instruction execution flow is shown in Figure 3-2 and Example 3-1.

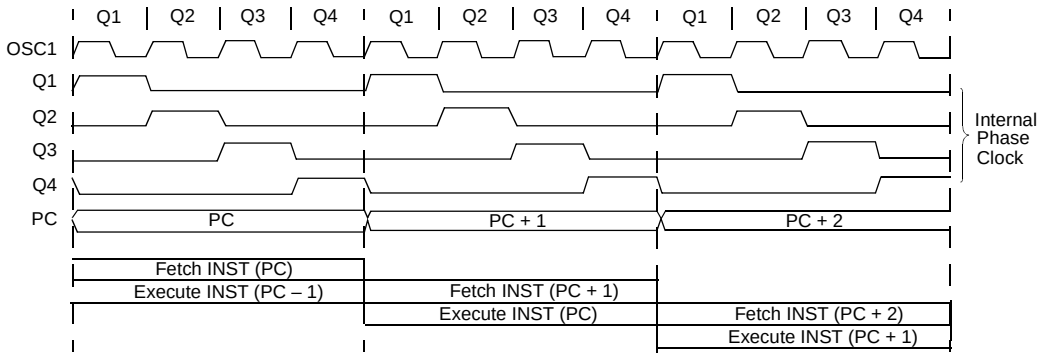
3.2 Instruction Flow/Pipelining

An instruction cycle consists of four Q cycles (Q1, Q2, Q3 and Q4). The instruction fetch and execute are pipelined such that fetch takes one instruction cycle, while decode and execute take another instruction cycle. However, due to the pipelining, each instruction effectively executes in one cycle. If an instruction causes the PC to change (e.g., `GOTO` or an interrupt), then two cycles are required to complete the instruction (Example 3-1).

A fetch cycle begins with the PC incrementing in Q1.

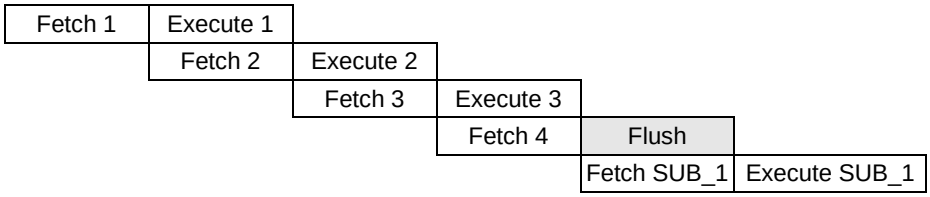
In the execution cycle, the fetched instruction is latched into the Instruction Register (IR) in cycle Q1. This instruction is then decoded and executed during the Q2, Q3 and Q4 cycles. Data memory is read during Q2 (operand read) and written during Q4 (destination write).

FIGURE 3-2: CLOCK/INSTRUCTION CYCLE



EXAMPLE 3-1: INSTRUCTION PIPELINE FLOW

```
1. MOVLW 03H
2. MOVWF PORTB
3. CALL SUB_1
4. BSF PORTB, BIT1
```



All instructions are single cycle, except for any program branches. These take two cycles, since the fetch instruction is “flushed” from the pipeline, while the new instruction is being fetched and then executed.

NOTES:

4.0 MEMORY ORGANIZATION

The PIC16F527 memories are organized into program memory and data memory (SRAM).The self-writable portion of the program memory called self-writable Flash data memory is located at addresses 400h-43Fh. All program mode commands that work on the normal Flash memory, work on the Flash data memory. This includes bulk erase, row/column/cycling toggles, Load and Read data commands (Refer to **Section 5.0 "Flash Data Memory Control"** for more details). For devices with more than 512 bytes of program memory, a paging scheme is used. Program memory pages are accessed using one STATUS register bit. For the PIC16F527, with data memory register files of more than 32 registers, a banking scheme is used. Data memory banks are accessed using the File Select Register (FSR).

4.1 Program Memory Organization for PIC16F527

The PIC16F527 device has an 11-bit Program Counter (PC) capable of addressing a 2K x 12 program memory space. Program memory is partitioned into user memory, data memory and configuration memory spaces.

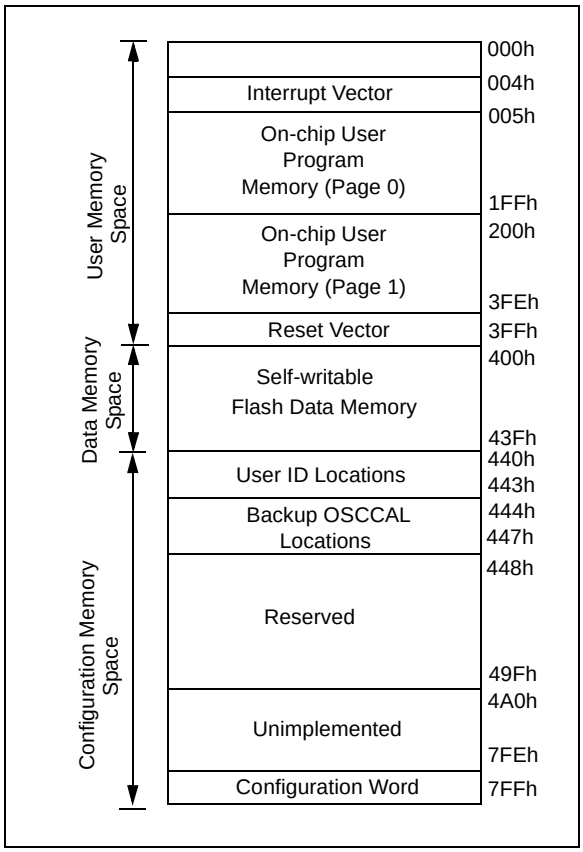
The user memory space is the on-chip user program memory. As shown in Figure 4-1, it extends from 0x000 to 0x3FF and partitions into pages, including an Interrupt vector at address 0x004 and a Reset vector at address 0x3FF.

The data memory space is the self-writable Flash data memory block and is located at addresses PC = 400h-43Fh. All program mode commands that work on the normal Flash memory, work on the Flash data memory block. This includes bulk erase, Load and Read data commands.

The configuration memory space extends from 0x440 to 0x7FF. Locations from 0x448 through 0x49F are reserved. The user ID locations extend from 0x440 through 0x443. The Backup OSCCAL locations extend from 0x444 through 0x447. The Configuration Word is physically located at 0x7FF.

Refer to "*PIC16F527 Memory Programming Specification*" (DS41640) for more details.

FIGURE 4-1: MEMORY MAP



4.2 Data Memory (SRAM and FSRs)

Data memory is composed of registers or bytes of SRAM. Therefore, data memory for a device is specified by its register file. The register file is divided into two functional groups: Special Function Registers (SFR) and General Purpose Registers (GPR).

The Special Function Registers are registers used by the CPU and peripheral functions for controlling desired operations of the PIC16F527. See **Section 4.3 "STATUS Register"** for details.

The PIC16F527 register file is composed of 16 Special Function Registers and 67 General Purpose Registers.

4.2.1 GENERAL PURPOSE REGISTER FILE

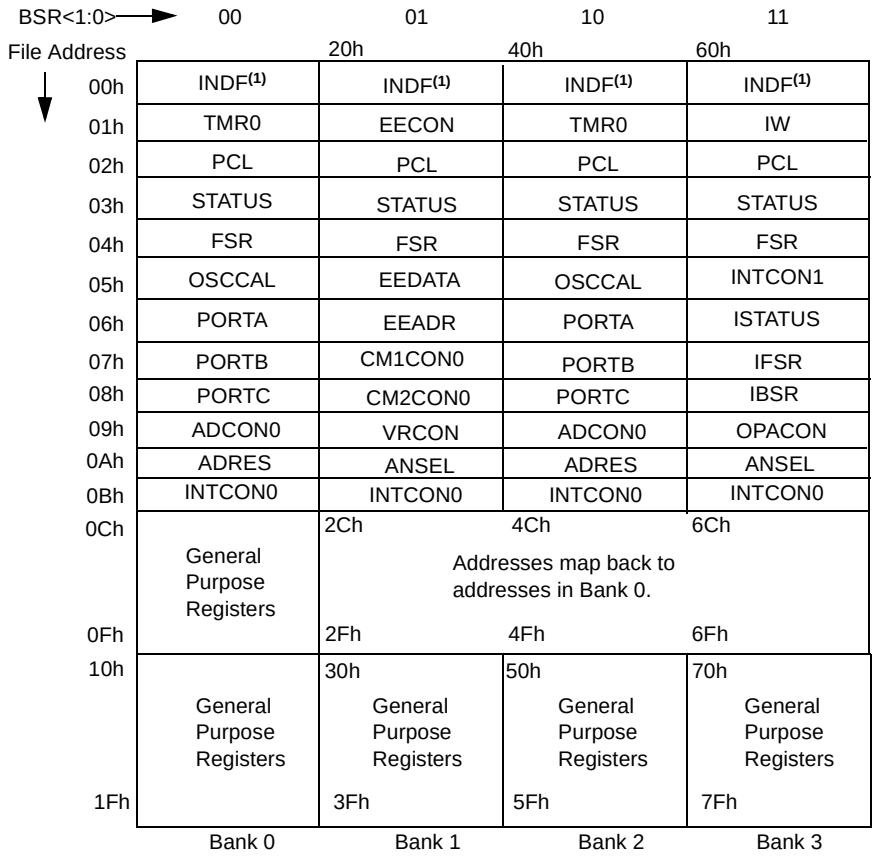
The General Purpose Register file is accessed directly or indirectly. See **Section 4.8 "Direct and Indirect Addressing"**.

4.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers (SFRs) are registers used by the CPU and peripheral functions to control the operation of the device (**Section 4.3 "STATUS Register"**).

The Special Function Registers can be classified into two sets. The Special Function Registers associated with the "core" functions are described in this section. Those related to the operation of the peripheral features are described in the section for each peripheral feature.

FIGURE 4-2: PIC16F527 REGISTER FILE MAP



Note 1: Not a physical register. See **Section 4.8 "Direct and Indirect Addressing"**.

TABLE 4-1: SPECIAL FUNCTION REGISTER SUMMARY

Address	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
Bank 0											
N/A	W(2)	Working Register (W)								xxxx xxxx	xxxx xxxx
N/A	TRIS	I/O Control Registers (TRISA, TRISB, TRISC)								1111 1111	1111 1111
N/A	OPTION	Contains control bits to configure Timer0 and Timer0/WDT prescaler								1111 1111	1111 1111
N/A	BSR(2)	—	—	—	—	—	BSR2	BSR1	BSR0	---- -000	---- -0uu
00h	INDF	Uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu
01h	TMR0	Timer0 module Register								xxxx xxxx	uuuu uuuu
02h	PCL(1)	Low-order eight bits of PC								1111 1111	1111 1111
03h	STATUS(2)	Reserved	Reserved	PA0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	-001 1xxx	-00q qqqq
04h	FSR(2)	—	Indirect data memory address pointer							0xxx xxxx	0uuu uuuu
05h	OSCCAL	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	1111 111-	uuuu uuu-
06h	PORTA	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--uu uuuu
07h	PORTB	RB7	RB6	RB5	RB4	—	—	—	—	xxxx ----	uuuu ----
08h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
09h	ADCON0	ADCS1	ADCS0	CHS3	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	ADON	1111 1100	1111 1100
0Ah	ADRES	ADC Conversion Result								xxxx xxxx	uuuu uuuu
0Bh	INTCON0	ADIF	CWIF	T0IF	RAIF	—	—	—	GIE	0000 ---0	0000 ---0
Bank 1											
N/A	W(2)	Working Register (W)								xxxx xxxx	xxxx xxxx
N/A	TRIS	I/O Control Registers (TRISA, TRISB, TRISC)								1111 1111	1111 1111
N/A	OPTION	Contains control bits to configure Timer0 and Timer0/WDT prescaler								1111 1111	1111 1111
N/A	BSR(2)	—	—	—	—	—	BSR2	BSR1	BSR0	---- -000	---- -0uu
20h	INDF	Uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu
21h	EECON	—	—	—	FREE	WRERR	WREN	WR	RD	---0 0000	---0 0000
22h	PCL(1)	Low-order eight bits of PC								1111 1111	1111 1111
23h	STATUS(2)	Reserved	Reserved	PA0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	-001 1xxx	-00q qqqq
24h	FSR(2)	—	Indirect data memory address pointer							0xxx xxxx	0uuu uuuu
25h	EEDATA	Self Read/Write Data								xxxx xxxx	uuuu uuuu
26h	EEADR	—	—	Self Read/Write Address						--xx xxxx	--uu uuuu
27h	CM1CON0	C1OUT	$\overline{C1OUTEN}$	C1POL	$\overline{C1T0CS}$	C1ON	C1NREF	C1PREF	$\overline{C1WU}$	1111 1111	quuu uuuu
28h	CM2CON0	C2OUT	$\overline{C2OUTEN}$	C2POL	C2PREF2	C2ON	C2NREF	C2PREF1	$\overline{C2WU}$	1111 1111	quuu uuuu
29h	VRCON	VREN	VROE	VRR	—	VR3	VR2	VR1	VR0	001- 1111	uuu- uuuu
2Ah	ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
2Bh	INTCON0	ADIF	CWIF	T0IF	RAIF	—	—	—	GIE	0000 ---0	0000 ---0

Legend: x = unknown, u = unchanged, – = unimplemented, read as '0' (if applicable), q = value depends on condition.
Shaded cells = unimplemented or unused

Note 1: The upper byte of the Program Counter is not directly accessible. See **Section 4.6 "Program Counter"** for an explanation of how to access these bits.

2: Registers are implemented as two physical registers. When executing from within an ISR, a secondary register is used at the same logical location. Both registers are persistent. See **Section 8.11 "Interrupts"**.

3: These registers show the contents of the registers in the other context: ISR or main line code. See **Section 8.11 "Interrupts"**.

TABLE 4-1: 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	Value on all other Resets
Bank 2											
N/A	W(2)	Working Register (W)								xxxx xxxx	xxxx xxxx
N/A	TRIS	I/O Control Registers (TRISA, TRISB, TRISC)								1111 1111	1111 1111
N/A	OPTION	Contains control bits to configure Timer0 and Timer0/WDT prescaler								1111 1111	1111 1111
N/A	BSR(2)	—	—	—	—	—	BSR2	BSR1	BSR0	---- -000	---- -0uu
40h	INDF	Uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu
41h	TMR0	Timer0 module Register								xxxx xxxx	uuuu uuuu
42h	PCL(1)	Low-order eight bits of PC								1111 1111	1111 1111
43h	STATUS(2)	Reserved	Reserved	PA0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	-001 1xxx	-00q qqqq
44h	FSR(2)	—	Indirect data memory address pointer							0xxx xxxx	0uuu uuuu
45h	OSCCAL	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	1111 111-	uuuu uuu-
46h	PORTA	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--uu uuuu
47h	PORTB	RB7	RB6	RB5	RB4	—	—	—	—	xxxx ----	uuuu ----
48h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
49h	ADCON0	ADCS1	ADCS0	CHS3	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	ADON	1111 1100	1111 1100
4Ah	ADRES	ADC Conversion Result								xxxx xxxx	uuuu uuuu
4Bh	INTCON0	ADIF	CWIF	T0IF	RAIF	—	—	—	GIE	0000 ---0	0000 ---0
Bank 3											
N/A	W(2)	Working Register (W)								xxxx xxxx	xxxx xxxx
N/A	TRIS	I/O Control Registers (TRISA, TRISB, TRISC)								1111 1111	1111 1111
N/A	OPTION	Contains control bits to configure Timer0 and Timer0/WDT prescaler								1111 1111	1111 1111
N/A	BSR(2)	—	—	—	—	—	BSR2	BSR1	BSR0	---- -000	---- -0uu
60h	INDF	Uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu
61h	IW(3)	Interrupt Working Register. (Addressed also as W register when within ISR)								xxxx xxxx	xxxx xxxx
62h	PCL(1)	Low-order eight bits of PC								1111 1111	1111 1111
63h	STATUS(2)	Reserved	Reserved	PA0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	-001 1xxx	-00q qqqq
64h	FSR(2)	—	Indirect data memory address pointer							0xxx xxxx	0uuu uuuu
65h	INTCON1	ADIE	CWIE	T0IE	RAIE	—	—	—	WUR	0000 ---0	0000 ---0
66h	ISTATUS(3)	Reserved	Reserved	PA0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	-xxx xxxx	-00q qqqq
67h	IFSR(3)	—	Indirect data memory address pointer							0xxx xxxx	0uuu uuuu
68h	IBSR(3)	—	—	—	—	—	BSR2	BSR1	BSR0	---- -0xx	---- -0uu
69h	OPACON	—	—	—	—	—	—	OPA2ON	OPA1ON	---- --00	---- --00
6Bh	INTCON0	ADIF	CWIF	T0IF	RAIF	—	—	—	GIE	0000 ---0	0000 ---0

Legend: x = unknown, u = unchanged, – = unimplemented, read as '0' (if applicable), q = value depends on condition.
Shaded cells = unimplemented or unused

Note 1: The upper byte of the Program Counter is not directly accessible. See **Section 4.6 "Program Counter"** for an explanation of how to access these bits.

2: Registers are implemented as two physical registers. When executing from within an ISR, a secondary register is used at the same logical location. Both registers are persistent. See **Section 8.11 "Interrupts"**.

3: These registers show the contents of the registers in the other context: ISR or main line code. See **Section 8.11 "Interrupts"**.

4.3 STATUS Register

This register contains the arithmetic status of the ALU, the Reset status and the page preselect bit.

The STATUS register can be the destination for any instruction, as with any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the $\overline{TO}$ and $\overline{PD}$ bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, `CLRF STATUS,` will clear the upper three bits and set the Z bit. This leaves the STATUS register as `000u u1uu` (where u = unchanged).

Therefore, it is recommended that only `BCF`, `BSF` and `MOVWF` instructions be used to alter the STATUS register. These instructions do not affect the Z, DC or C bits from the STATUS register. For other instructions which do affect Status bits, see **Section 13.0 "Instruction Set Summary"**.

REGISTER 4-1: STATUS: STATUS REGISTER

R-0	R-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
Reserved	Reserved	PA0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
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 7-6 **Reserved**: Read as '0'

bit 5 **PA0**: Program Page Preselect bit
1 = Page 1 (000h-1FFh)
0 = Page 0 (200h-3FFh)

bit 4 **$\overline{TO}$**: Time-Out bit
1 = After power-up, `CLRWDT` instruction, or `SLEEP` instruction
0 = A WDT time-out occurred

bit 3 **$\overline{PD}$**: Power-Down bit
1 = After power-up or by the `CLRWDT` instruction
0 = By execution of the `SLEEP` instruction

bit 2 **Z**: Zero bit
1 = The result of an arithmetic or logic operation is zero
0 = The result of an arithmetic or logic operation is not zero

bit 1 **DC**: Digit carry/$\overline{borrow}$ bit (for `ADDWF` and `SUBWF` instructions)
`ADDWF:`
1 = A carry from the 4th low-order bit of the result occurred
0 = A carry from the 4th low-order bit of the result did not occur
`SUBWF:`
1 = A borrow from the 4th low-order bit of the result did not occur
0 = A borrow from the 4th low-order bit of the result occurred

bit 0 **C**: Carry/$\overline{borrow}$ bit (for `ADDWF`, `SUBWF` and `RRF`, `RLF` instructions)
`ADDWF: SUBWF: RRF` or `RLF:`
1 = A carry occurred 1 = A borrow did not occur; Load bit with LSb or MSb, respectively
0 = A carry did not occur 0 = A borrow occurred

4.4 OPTION Register

The OPTION register is a 8-bit wide, write-only register, which contains various control bits to configure the Timer0/WDT prescaler and Timer0.

By executing the OPTION instruction, the contents of the W register will be transferred to the OPTION register. A Reset sets the OPTION <7:0> bits.

Note:	If TRIS bit is set to '0', the wake-up on change and pull-up functions are disabled for that pin (i.e., note that TRIS overrides Option control of $\overline{\text{RAPU}}$ and $\overline{\text{RAWU}}$).

REGISTER 4-2: OPTION: OPTION REGISTER

W-1	W-1	W-1	W-1	W-1	W-1	W-1	W-1
$\overline{\text{RAWU}}$(2)	$\overline{\text{RAPU}}$	T0CS(1)	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

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

bit 7 **$\overline{\text{RAWU}}$:** Enable PORTA Interrupt Flag on Pin Change bit(2)
1 = Disabled
0 = Enabled

bit 6 **$\overline{\text{RAPU}}$:** Enable PORTA Weak Pull-Ups bit
1 = Disabled
0 = Enabled

bit 5 **T0CS:** Timer0 Clock Source Select bit(1)
1 = Transition on T0CKI pin
0 = Internal instruction cycle clock (CLKOUT)

bit 4 **T0SE:** Timer0 Source Edge Select bit
1 = Increment on high-to-low transition on T0CKI pin
0 = Increment on low-to-high transition on T0CKI pin

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

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

Note 1: If the T0CS bit is set to '1', it will override the TRIS function on the T0CKI pin.
2: The $\overline{\text{RAWU}}$ bit of the OPTION register must be set to enable the RAIF function in the INTCON0 register.

4.5 OSCCAL Register

The Oscillator Calibration (OSCCAL) register is used to calibrate the 8 MHz internal oscillator macro. It contains seven bits of calibration that uses a two's complement scheme for controlling the oscillator speed. See Register 4-3 for details.

REGISTER 4-3: OSCCAL: OSCILLATOR CALIBRATION REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	U-0
CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—
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 7-1 **CAL<6:0>**: Oscillator Calibration bits

0111111 = Maximum frequency
•
•
•
0000001
0000000 = Center frequency
1111111
•
•
•
1000000 = Minimum frequency

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

4.6 Program Counter

As a program instruction is executed, the Program Counter (PC) will contain the address of the next program instruction to be executed. The PC value is increased by one every instruction cycle, unless an instruction changes the PC.

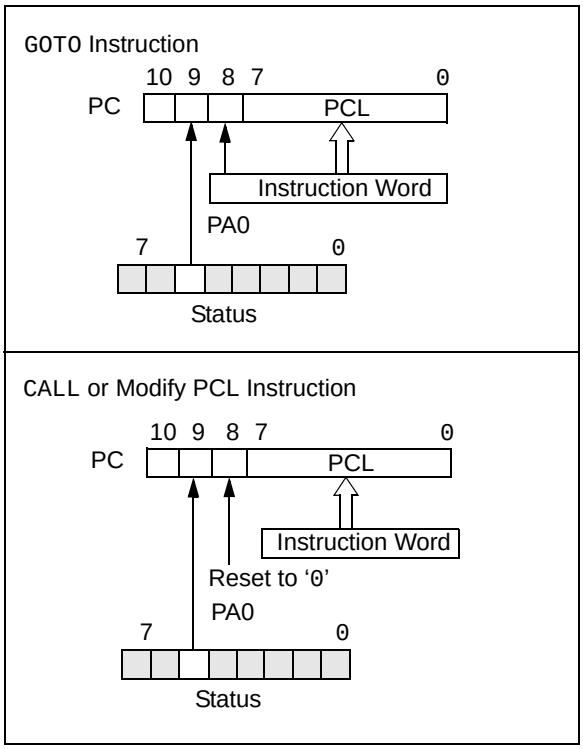
For a GOTO instruction, bits <8:0> of the PC are provided by the GOTO instruction word. The Program Counter (PCL) is mapped to PC<7:0>. Bit 5 of the STATUS register provides page information to bit 9 of the PC (Figure 4-3).

For a CALL instruction, or any instruction where the PCL is the destination, bits <7:0> of the PC again are provided by the instruction word. However, PC<8> does not come from the instruction word, but is always cleared (Figure 4-3).

Instructions where the PCL is the destination, or modify PCL instructions, include MOVWF PCL, ADDWF PCL and BSF PCL,5.

Note:	Because bit 8 of the PC is cleared in the CALL instruction or any modify PCL instruction, all subroutine calls or computed jumps are limited to the first 256 locations of any program memory page (512 words long).

FIGURE 4-3: LOADING OF PC BRANCH INSTRUCTIONS



4.6.1 EFFECTS OF RESET

The PC is set upon a Reset, which means that the PC addresses the last location in the last page (i.e., the oscillator calibration instruction). After executing MOVLW XX, the PC will roll over to location 00h and begin executing user code.

The STATUS register page preselect bits are cleared upon a Reset, which means that page 0 is pre-selected.

Therefore, upon a Reset, a GOTO instruction will automatically cause the program to jump to page 0 until the value of the page bits is altered.

4.7 Stack

The PIC16F527 device has a 4-deep, 12-bit wide hardware PUSH/POP stack.

A CALL instruction or an interrupt will PUSH the current PC value, incremented by one, into Stack Level 1. If there was a previous value in the Stack 1 location, it will be pushed into the Stack 2 location. This process will be continued throughout the remaining stack locations populated with values. If more than four sequential CALLs are executed, only the most recent four return addresses are stored.

A RETLW, RETURN or RETFIE instruction will POP the contents of Stack Level 1 into the PC. If there was a previous value in the Stack 2 location, it will be copied into the Stack Level 1 location. This process will be continued throughout the remaining stack locations populated with values. If more than four sequential RETLWs are executed, the stack will be filled with the address previously stored in Stack Level 4. Note that the W register will be loaded with the literal value specified in the instruction. This is particularly useful for the implementation of data look-up tables within the program memory.

Note 1:	There are no Status bits to indicate Stack Overflows or Stack Underflow conditions.
2:	There are no instruction mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, RETURN, RETFIE and RETLW instructions.

4.8 Direct and Indirect Addressing

4.8.1 DIRECT DATA ADDRESSING: BSR REGISTER

Traditional data memory addressing is performed in the Direct Addressing mode. In Direct Addressing, the Bank Select Register bits BSR<1:0>, in the new BSR register, are used to select the data memory bank. The address location within that bank comes directly from the opcode being executed.

BSR<1:0> are the bank select bits and are used to select the bank to be addressed (00 = Bank 0, 01 = Bank 1, 10 = Bank 2, 11 = Bank 3).

A new instruction supports the addition of the BSR register, called the MOVLB instruction. See **Section 13.0 "Instruction Set Summary"** for more information.

4.8.2 INDIRECT DATA ADDRESSING: INDF AND FSR REGISTERS

The INDF Register is not a physical register. Addressing INDF actually addresses the register whose address is contained in the FSR Register (FSR is a *pointer*). This is indirect addressing.

Reading INDF itself indirectly (FSR = 0) will produce 00h. Writing to the INDF Register indirectly results in a no-operation (although Status bits may be affected).

The FSR is an 8-bit wide register. It is used in conjunction with the INDF Register to indirectly address the data memory area.

The FSR<6:0> bits are used to select data memory addresses 00h to 1Fh.

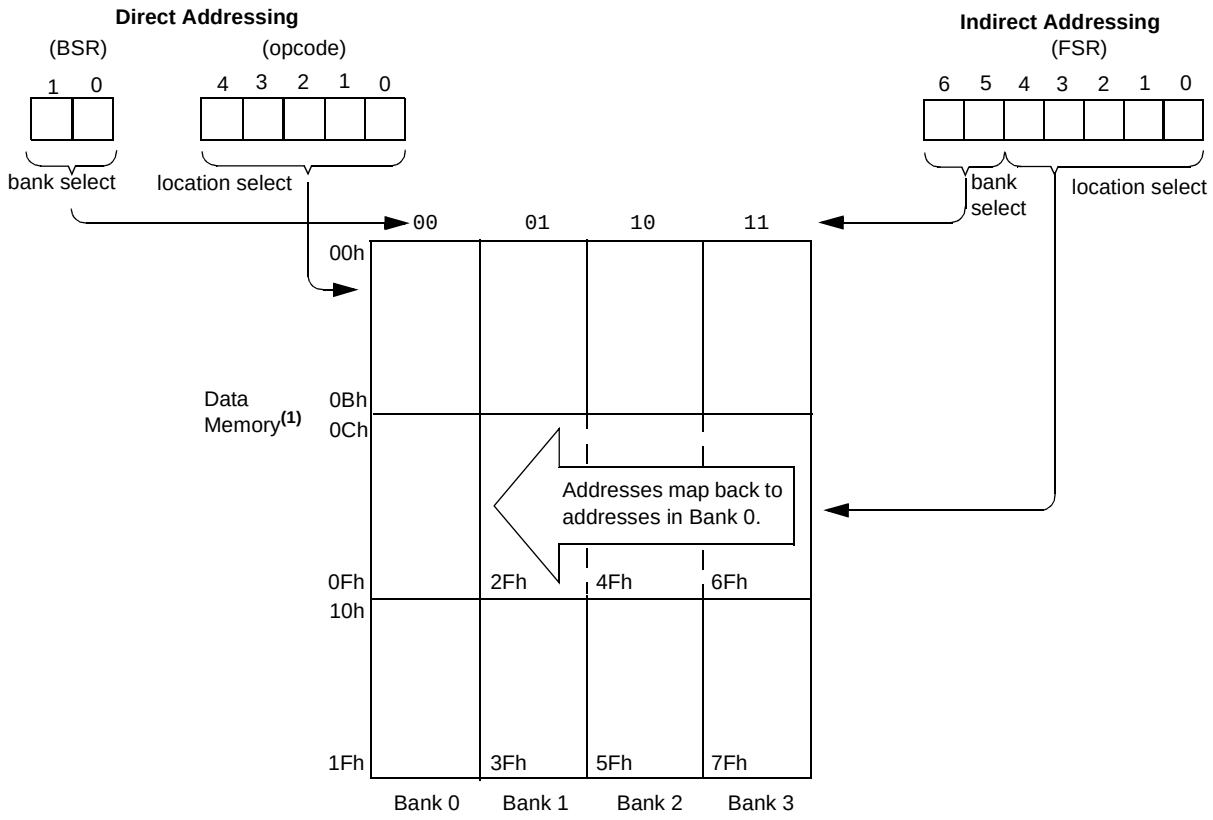
FSR<7> is unimplemented and read as '0'.

A simple program to clear RAM locations 10h-1Fh using indirect addressing is shown in Example 4-1.

EXAMPLE 4-1: HOW TO CLEAR RAM USING INDIRECT ADDRESSING

```
        MOVLW   0x10    ;initialize pointer
        MOVWF   FSR     ;to RAM
NEXT    CLRF    INDF    ;clear INDF
                        ;register
        INCF    FSR,F   ;inc pointer
        BTFSC   FSR,4   ;all done?
        GOTO    NEXT    ;NO, clear next
CONTINUE
            :           ;YES, continue
            :
```

FIGURE 4-4: DIRECT/INDIRECT ADDRESSING



Note 1: For register map detail see **Section 4.3 "STATUS Register"**.

5.0 FLASH DATA MEMORY CONTROL

The Flash data memory is readable and writable during normal operation (full VDD range). This memory is not directly mapped in the register file space. Instead, it is indirectly addressed through the Special Function Registers (SFRs).

5.1 Reading Flash Data Memory

To read a Flash data memory location the user must:

- Write the EEADR register
- Set the RD bit of the EECON register

The value written to the EEADR register determines which Flash data memory location is read. Setting the RD bit of the EECON register initiates the read. Data from the Flash data memory read is available in the EEDATA register immediately. The EEDATA register will hold this value until another read is initiated or it is modified by a write operation. Program execution is suspended while the read cycle is in progress. Execution will continue with the instruction following the one that sets the WR bit. See Example 1 for sample code.

EXAMPLE 1: READING FROM FLASH DATA MEMORY

```
BANKSEL EEADR            ;
MOVF DATA_EE_ADDR, W     ;
MOVWF EEADR              ;Data Memory
                         ;Address to read
BANKSEL EECON1           ;

BSF EECON, RD            ;EE Read
MOVF EEDATA, W           ;W = EEDATA
```

Note: Only a BSF command will work to enable the Flash data memory read documented in Example 1. No other sequence of commands will work, no exceptions.

5.2 Writing and Erasing Flash Data Memory

Flash data memory is erased one row at a time and written one byte at a time. The 64-byte array is made up of eight rows. A row contains eight sequential bytes. Row boundaries exist every eight bytes.

Generally, the procedure to write a byte of data to Flash data memory is:

1. Identify the row containing the address where the byte will be written.
2. If there is other information in that row that must be saved, copy those bytes from Flash data memory to RAM.
3. Perform a row erase of the row of interest.
4. Write the new byte of data and any saved bytes back to the appropriate addresses in Flash data memory.

To prevent accidental corruption of the Flash data memory, an unlock sequence is required to initiate a write or erase cycle. This sequence requires that the bit set instructions used to configure the EECON register happen exactly as shown in Example 2 and Example 3, depending on the operation requested.

5.2.1 ERASING FLASH DATA MEMORY

A row must be manually erased before writing new data. The following sequence must be performed for a single row erase.

1. Load EEADR with an address in the row to be erased.
2. Set the FREE bit to enable the erase.
3. Set the WREN bit to enable write access to the array.
4. Set the WR bit to initiate the erase cycle.

If the WREN bit is not set in the instruction cycle after the FREE bit is set, the FREE bit will be cleared in hardware.

If the WR bit is not set in the instruction cycle after the WREN bit is set, the WREN bit will be cleared in hardware.

Sample code that follows this procedure is included in Example 2.

Program execution is suspended while the erase cycle is in progress. Execution will continue with the instruction following the one that sets the WR bit.

EXAMPLE 2: ERASING A FLASH DATA MEMORY ROW

```
BANKSEL   EEADR
MOVLW     EE_ADR_ERASE   ; LOAD ADDRESS OF ROW TO
                         ; ERASE
MOVWF     EEADR          ;
BSF       EECON,FREE     ; SELECT ERASE
BSF       EECON,WREN     ; ENABLE WRITES
BSF       EECON,WR       ; INITITATE ERASE
```

Note 1: The FREE bit may be set by any command normally used by the core. However, the WREN and WR bits can only be set using a series of BSF commands, as documented in Example 1. No other sequence of commands will work, no exceptions.

2: Bits <5:3> of the EEADR register indicate which row is to be erased.

5.2.2 WRITING TO FLASH DATA MEMORY

Once a cell is erased, new data can be written. Program execution is suspended during the write cycle. The following sequence must be performed for a single byte write.

1. Load EEADR with the address.
2. Load EEDATA with the data to write.
3. Set the WREN bit to enable write access to the array.
4. Set the WR bit to initiate the erase cycle.

If the WR bit is not set in the instruction cycle after the WREN bit is set, the WREN bit will be cleared in hardware.

Sample code that follows this procedure is included in Example 3.

EXAMPLE 3: WRITING A FLASH DATA MEMORY ROW

```
BANKSEL   EEADR
MOVLW     EE_ADR_WRITE     ; LOAD ADDRESS
MOVWF     EEADR            ;
MOVLW     EE_DATA_TO_WRITE ; LOAD DATA
MOVWF     EEDATA           ; INTO EEDATA REGISTER
BSF       EECON,WREN       ; ENABLE WRITES
BSF       EECON,WR         ; INITITATE ERASE
```

Note 1: Only a series of BSF commands will work to enable the memory write sequence documented in Example 2. No other sequence of commands will work, no exceptions.

2: For reads, erases and writes to the Flash data memory, there is no need to insert a NOP into the user code as is done on mid-range devices. The instruction immediately following the "BSF EECON,WR/RD" will be fetched and executed properly.

5.3 Write Verify

Depending on the application, good programming practice may dictate that data written to the Flash data memory be verified. Example 4 is an example of a write verify.

EXAMPLE 4: WRITE VERIFY OF FLASH DATA MEMORY

```
MOVF      EEDATA, W     ;EEDATA has not changed
                        ;from previous write
BSF       EECON, RD     ;Read the value written
XORWF     EEDATA, W     ;
BTFSS     STATUS, Z     ;Is data the same
GOTO      WRITE_ERR     ;No, handle error
                        ;Yes, continue
```


5.4 Register Definitions — Memory Control

REGISTER 5-1: EEDATA: FLASH DATA REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EEDATA7	EEDATA6	EEDATA5	EEDATA4	EEDATA3	EEDATA2	EEDATA1	EEDATA0
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 7-0 **EEDATA<7:0>**: Eight bits of data to be read from/written to data Flash

REGISTER 5-2: EEADR: FLASH ADDRESS REGISTER

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	EEADR5	EEADR4	EEADR3	EEADR2	EEADR1	EEADR0
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 7-6 **Unimplemented:** Read as ‘0’.

bit 5-0 **EEADR<5:0>**: Six bits of data to be read from/written to data Flash

REGISTER 5-3: EECON: FLASH CONTROL REGISTER

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	FREE	WRERR	WREN	WR	RD
bit 7							bit 0

Legend:			
S = Bit can only be set			
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 7-5 **Unimplemented:** Read as ‘0’.

bit 4 **FREE:** Flash Data Memory Row Erase Enable Bit
- 1 = Program memory row being pointed to by EEADR will be erased on the next write cycle. No write will be performed. This bit is cleared at the completion of the erase operation.
- 0 = Perform write only

bit 3 **WRERR:** Write Error Flag bit
- 1 = A write operation terminated prematurely (by device Reset)
- 0 = Write operation completed successfully

bit 2 **WREN:** Write Enable bit
- 1 = Allows write cycle to Flash data memory
- 0 = Inhibits write cycle to Flash data memory

bit 1 **WR:** Write Control bit
- 1 = Initiate a erase or write cycle
- 0 = Write/Erase cycle is complete

bit 0 **RD:** Read Control bit
- 1 = Initiate a read of Flash data memory
- 0 = Do not read Flash data memory

5.5 Code Protection

Code protection does not prevent the CPU from performing read or write operations on the Flash data memory. Refer to the code protection chapter for more information.

6.0 I/O PORT

As with any other register, the I/O register(s) can be written and read under program control. However, read instructions (e.g., `MOVF PORTB,W`) always read the I/O pins independent of the pin's Input/Output modes. On Reset, all I/O ports are defined as input (inputs are at high-impedance) since the I/O control registers are all set.

6.1 PORTA

PORTA is a 6-bit I/O register. Only the low-order six bits are used (RA<5:0>). Bits 7 and 6 are unimplemented and read as '0's. Please note that RA3 is an input-only pin. The Configuration Word can set several I/Os to alternate functions. When acting as alternate functions, the pins will read as '0' during a port read. Pins RA0, RA1, RA3 and RA4 can be configured with weak pull-ups and also for wake-up on change. The wake-up on change and weak pull-up functions are not pin selectable. If RA3/$\overline{\text{MCLR}}$ is configured as $\overline{\text{MCLR}}$, weak pull-up is always on and wake-up on change for this pin is not enabled.

6.2 PORTB

PORTB is a 4-bit I/O register. Only the high-order four bits are used (RB<7:4>). Bits 0 through 3 are unimplemented and read as '0's.

6.3 PORTC

PORTC is a 8-bit I/O register.

6.4 TRIS Register

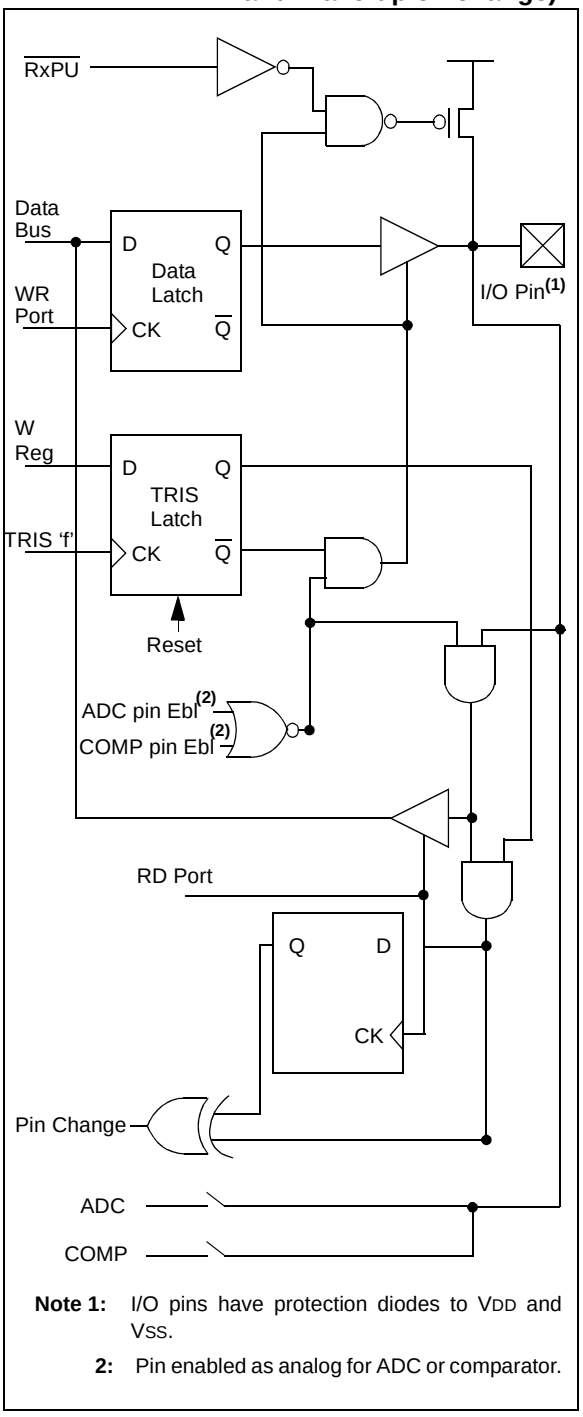
The Output Driver Control register is loaded with the contents of the W register by executing the `TRIS` instruction. A '1' from a TRIS register bit puts the corresponding output driver in a High-Impedance mode. A '0' puts the contents of the output data latch on the selected pins, enabling the output buffer. The exceptions are RA3, which is input-only and the T0CKI pin, which may be controlled by the OPTION register (see Register 4-2).

TRIS registers are "write-only". Active bits in these registers are set (output drivers disabled) upon Reset.

6.5 I/O Interfacing

The equivalent circuit for an I/O port pin is shown in Figure 6-1. All port pins, except the MCLR pin which is input-only, may be used for both input and output operations. For input operations, these ports are non-latching. Any input must be present until read by an input instruction (e.g., `MOVF PORTB, W`). The outputs are latched and remain unchanged until the output latch is rewritten. To use a port pin as output, the corresponding direction control bit in TRIS must be cleared (= `0`). For use as an input, the corresponding TRIS bit must be set. Any I/O pin (except MCLR) can be programmed individually as input or output.

FIGURE 6-1: BLOCK DIAGRAM OF I/O PIN (Example shown of RA2 with Weak Pull-up and Wake-up on change)



Note 1: I/O pins have protection diodes to VDD and VSS.

2: Pin enabled as analog for ADC or comparator.

6.6 Register Definitions — PORT Control

REGISTER 6-1: PORTA: PORTA REGISTER

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	RA5	RA4	RA3	RA2	RA1	RA0
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 7-6 **Unimplemented**: Read as '0'

bit 5-0 **RA<5:0>**: PORTA I/O Pin bits
1 = Port pin is >VIH min.
0 = Port pin is <VIL max.

TABLE 6-1: PORTA PINS ORDER OF PRECEDENCE

Priority	RA5	RA4	RA3	RA2	RA1	RA0
1	OSC1	OSC2	RA3/$\overline{\text{MCLR}}$	AN2	CVREF	AN0
2	CLKIN	CLKOUT	—	C1OUT	AN1	C1IN+
3	TRISA5	AN3	—	T0CKI	C1IN-	TRISA0
4	—	TRISA5	—	TRISA2	TRISA1	—

TABLE 6-2: WEAK PULL-UP ENABLED PINS

Device	RA0 Weak Pull-up	RA1 Weak Pull-up	RA3 Weak Pull-up(1)	RA4 Weak Pull-up
PIC16F527	Yes	Yes	Yes	Yes

Note 1: When MCLREN = 1, the weak pull-up on $\overline{\text{MCLR}}$ is always enabled.

REGISTER 6-2: PORTB: PORTB REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	U-0	U-0	U-0	U-0
RB7	RB6	RB5	RB4	—	—	—	—
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 7-4 **RB<7:4>**: PORTB I/O Pin bits
1 = Port pin is >VIH min.
0 = Port pin is <VIL max.

bit 3-0 **Unimplemented**: Read as '0'

TABLE 6-3: PORTB PINS ORDER OF PRECEDENCE

Priority	RB7	RB6	RB5	RB4
1	TRISB7	TRISB6	OP2+	OP2-
2	—	—	TRISB5	TRISB4

REGISTER 6-3: PORTC: PORTC REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
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 7-0 **RC<7:0>**: PORTC I/O Pin bits
1 = Port pin is >VIH min.
0 = Port pin is <VIL max.

TABLE 6-4: PORTC PINS ORDER OF PRECEDENCE

Priority	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
1	OP1+	OP1-	TRISC5	C2OUT	OP1	OP2	C2IN-	C2IN+
2	TRISC7	TRISC6	—	TRISC4	AN7	AN6	AN5	AN4
3	—	—	—	—	TRISC3	TRISC2	TRISC1	TRISC0

REGISTER 6-4: ANSEL 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
ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0
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 7-0 **ANS<7:0>:** ADC Analog Input Pin Select(1), (2)
0 = Analog function on selected ANx pin is disabled
1 = ANx configured as an analog input

Note 1: When the ANSx bits are set, the channels selected will automatically be forced into Analog mode, regardless of the pin function previously defined. The only exception to this is the comparator, where the analog input to the comparator and the ADC will be active at the same time. It is the user's responsibility to ensure that the ADC loading on the comparator input does not affect their application.
2: The ANS<7:0> bits are active regardless of the condition of ADON.

TABLE 6-5: REGISTERS ASSOCIATED WITH THE I/O PORTS

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-On Reset	Value on MCLR and WDT Reset
N/A	TRIS(1)	I/O Control Registers (TRISA, TRISB, TRISC)(1)								1111 1111	1111 1111
06h	PORTA	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--uu uuuu
07h	PORTB	RB7	RB6	RB5	RB4	—	—	—	—	xxxx ----	uuuu ----
27h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu

Legend: x = unknown, u = unchanged, — = unimplemented, read as '0', Shaded cells = unimplemented, read as '0'
Note 1: TRISA3 is read-only '1', and cannot be set as output.

6.7 I/O Programming Considerations

6.7.1 BIDIRECTIONAL I/O PORTS

Some instructions operate internally as read followed by write operations. The BCF and BSF instructions, for example, read the entire port into the CPU, execute the bit operation and rewrite the result. Caution must be used when these instructions are applied to a port where one or more pins are used as input/outputs. For example, a BSF operation on bit 5 of PORTB will cause all eight bits of PORTB to be read into the CPU, bit 5 to be set and the PORTB value to be written to the output latches. If another bit of PORTB is used as a bidirectional I/O pin (say bit 0) and it is defined as an input at this time, the input signal present on the pin itself would be read into the CPU and rewritten to the data latch of this particular pin, overwriting the previous content. As long as the pin stays in the Input mode, no problem occurs. However, if bit 0 is switched into Output mode later on, the content of the data latch may now be unknown.

Example 6-1 shows the effect of two sequential Read-Modify-Write instructions (e.g., BCF, BSF, etc.) on an I/O port.

A pin actively outputting a high or a low should not be driven from external devices at the same time in order to change the level on this pin ("wired OR", "wired AND"). The resulting high output currents may damage the chip.

EXAMPLE 6-1: READ-MODIFY-WRITE INSTRUCTIONS ON AN I/O PORT(e.g. PIC16F527)

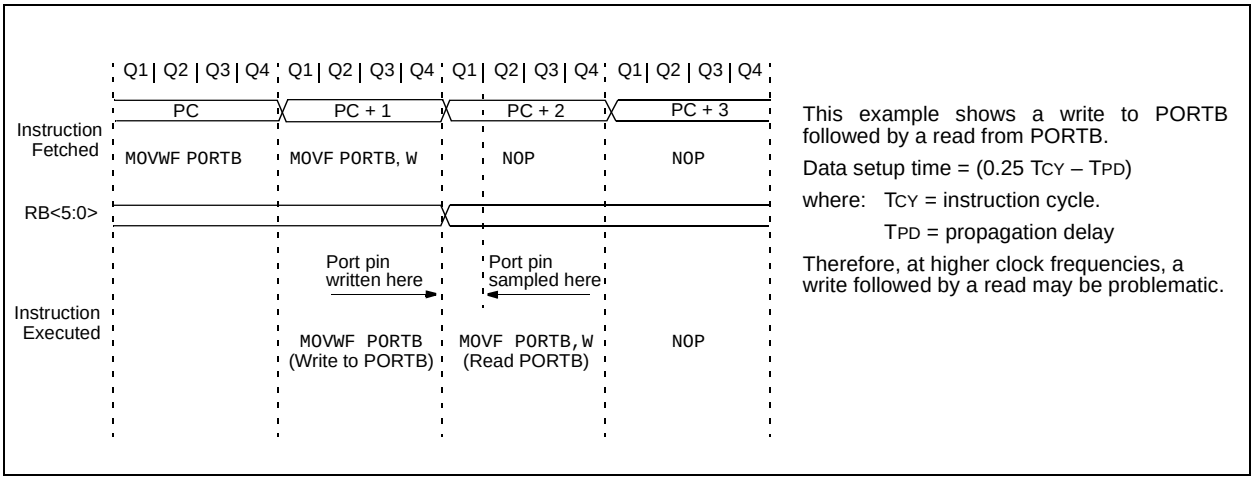
```
;Initial PORTB Settings
;PORTB<5:3> Inputs
;PORTB<2:0> Outputs
;
;   PORTB latch PORTB pins
;   ----------------------
  BCF       PORTB, 5    ;--01 -ppp--11 pppp
  BCF       PORTB, 4    ;--10 -ppp--11 pppp
  MOVLW     007h        ;
  TRIS      PORTB       ;--10 -ppp--11 pppp
;
```

Note 1: The user may have expected the pin values to be '--00 pppp'. The 2nd BCF caused RB5 to be latched as the pin value (High).

6.7.2 SUCCESSIVE OPERATIONS ON I/O PORTS

The actual write to an I/O port happens at the end of an instruction cycle, whereas for reading, the data must be valid at the beginning of the instruction cycle (Figure 6-2). Therefore, care must be exercised if a write followed by a read operation is carried out on the same I/O port. The sequence of instructions should allow the pin voltage to stabilize (load dependent) before the next instruction causes that file to be read into the CPU. Otherwise, the previous state of that pin may be read into the CPU rather than the new state. When in doubt, it is better to separate these instructions with a NOP or another instruction not accessing this I/O port.

FIGURE 6-2: SUCCESSIVE I/O OPERATION



This example shows a write to PORTB followed by a read from PORTB.

Data setup time = (0.25 T_{CY} – T_{PD})

where: T_{CY} = instruction cycle.

T_{PD} = propagation delay

Therefore, at higher clock frequencies, a write followed by a read may be problematic.

NOTES:

7.0 TIMER0 MODULE AND TMR0 REGISTER

The Timer0 module has the following features:

- 8-bit timer/counter register, TMR0
- Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select:
  - Edge select for external clock

Figure 7-1 is a simplified block diagram of the Timer0 module.

Timer mode is selected by clearing the T0CS bit of the OPTION register. In Timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If TMR0 register is written, the increment is inhibited for the following two cycles (Figure 7-2 and Figure 7-3). The user can work around this by writing an adjusted value to the TMR0 register.

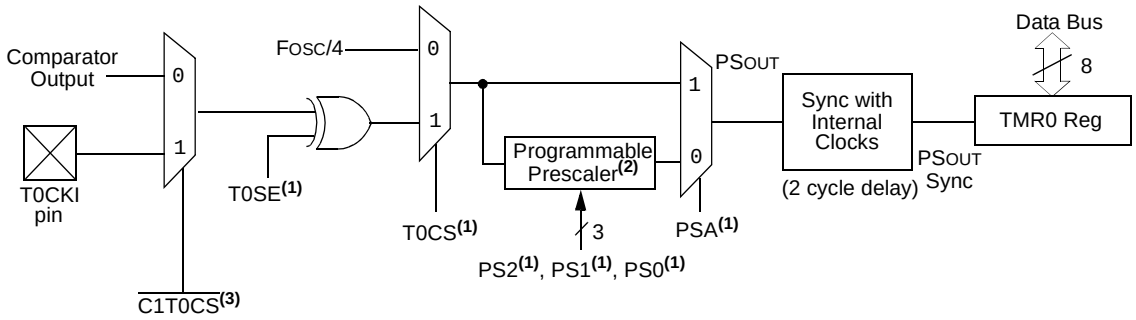
There are two types of Counter mode. The first Counter mode uses the T0CKI pin to increment Timer0. It is selected by setting the T0CS bit of the OPTION register, setting the C1T0CS bit of the CM1CON0 register and setting the C1OUTEN bit of the CM1CON0 register. In this mode, Timer0 will increment either on every rising or falling edge of pin T0CKI. The T0SE bit of the OPTION register determines the source edge. Clearing the T0SE bit selects the rising edge. Restrictions on the external clock input are discussed in detail in **Section 7.1 "Using Timer0 with an External Clock"**.

The second Counter mode uses the output of the comparator to increment Timer0. It can be entered in by setting the T0CS bit of the OPTION register, and clearing the C1T0CS bit of the CM1CON0 register (C1OUTEN [CM1CON0<6>] does not affect this mode of operation). This enables an internal connection between the comparator and the Timer0.

The prescaler may be used by either the Timer0 module or the Watchdog Timer, but not both. The prescaler assignment is controlled in software by the control bit, PSA of the OPTION register. Clearing the PSA bit will assign the prescaler to Timer0. The prescaler is not readable or writable. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4,..., 1:256 are selectable. **Section 7.2 "Prescaler"** details the operation of the prescaler.

A summary of registers associated with the Timer0 module is found in Table 7-1.

FIGURE 7-1: TIMER0 BLOCK DIAGRAM



Note 1: Bits T0CS, T0SE, PSA, PS2, PS1 and PS0 are located in the OPTION register.
2: The prescaler is shared with the Watchdog Timer.
3: The C1T0CS bit is in the CM1CON0 register.

FIGURE 7-2: TIMER0 TIMING: INTERNAL CLOCK/NO PRESCALE

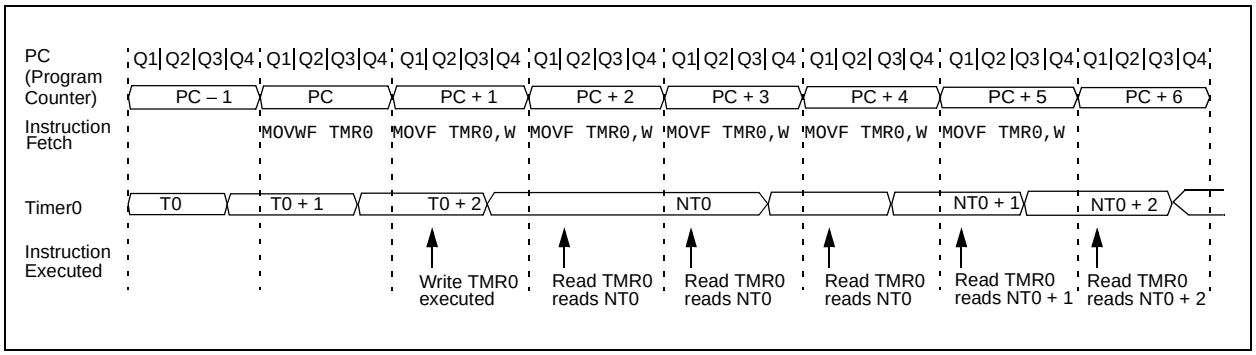


FIGURE 7-3: TIMER0 TIMING: INTERNAL CLOCK/PRESCALE 1:2

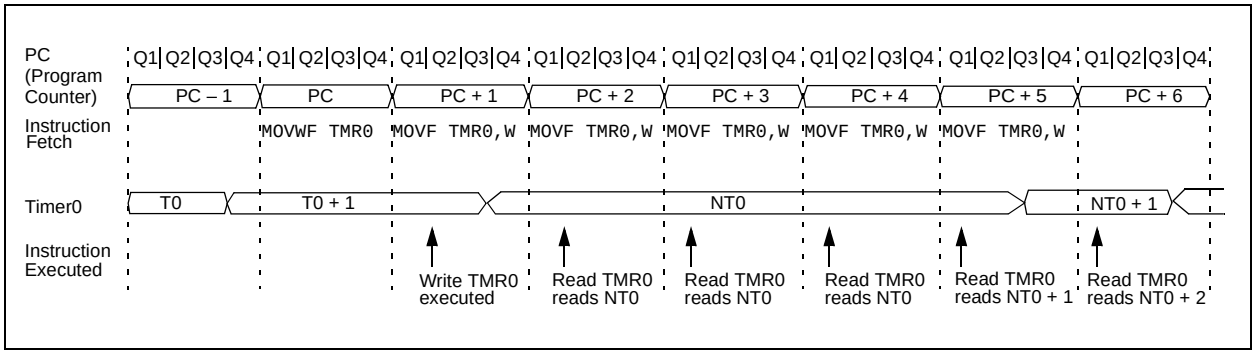


TABLE 7-1: REGISTERS ASSOCIATED WITH TIMER0

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
TMR0	Timer0 module Register								—
CM1CON0	C1OUT	$\overline{\text{C1OUTEN}}$	C1POL	$\overline{\text{C1T0CS}}$	C1ON	C1NREF	C1PREF	$\overline{\text{C1WU}}$	66
CM2CON0	C2OUT	$\overline{\text{C2OUTEN}}$	C2POL	C2PREF2	C2ON	C2NREF	C2PREF1	$\overline{\text{C2WU}}$	67
OPTION	$\overline{\text{RAWU}}$	$\overline{\text{RAPU}}$	T0CS	T0SE	PSA	PS2	PS1	PS0	20
TRIS(1)	I/O Control Registers (TRISA, TRISB, TRISC)								—

Legend: Shaded cells are not used by Timer0. – = unimplemented, x = unknown, u = unchanged.
Note 1: The TRIS of the T0CKI pin is overridden when T0CS = 1.

7.1 Using Timer0 with an External Clock

When an external clock input is used for Timer0, it must meet certain requirements. The external clock requirement is due to internal phase clock (TOSC) synchronization. Also, there is a delay in the actual incrementing of Timer0 after synchronization.

7.1.1 EXTERNAL CLOCK SYNCHRONIZATION

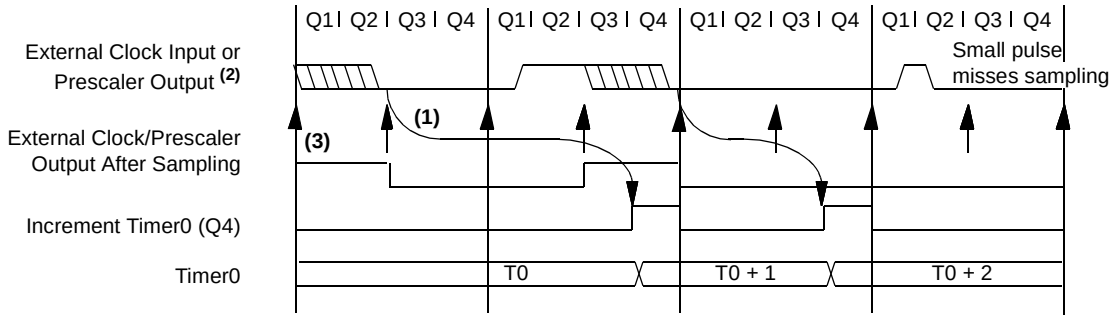
When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI with the internal phase clocks is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks (Figure 7-4). Therefore, it is necessary for T0CKI to be high for at least 2 TOSC (and a small RC delay of 2 Tt0H) and low for at least 2 TOSC (and a small RC delay of 2 Tt0H). Refer to the electrical specification of the desired device.

When a prescaler is used, the external clock input is divided by the asynchronous ripple counter-type prescaler, so that the prescaler output is symmetrical. For the external clock to meet the sampling requirement, the ripple counter must be taken into account. Therefore, it is necessary for T0CKI to have a period of at least 4 TOSC (and a small RC delay of 4 Tt0H) divided by the prescaler value. The only requirement on T0CKI high and low time is that they do not violate the minimum pulse width requirement of Tt0H. Refer to parameters 40, 41 and 42 in the electrical specification of the desired device.

7.1.2 TIMER0 INCREMENT DELAY

Since the prescaler output is synchronized with the internal clocks, there is a small delay from the time the external clock edge occurs to the time the Timer0 module is actually incremented. Figure 7-4 shows the delay from the external clock edge to the timer incrementing.

FIGURE 7-4: TIMER0 TIMING WITH EXTERNAL CLOCK



Note 1: Delay from clock input change to Timer0 increment is 3 TOSC to 7 TOSC. (Duration of Q = TOSC). Therefore, the error in measuring the interval between two edges on Timer0 input = ±4 TOSC max.

2: External clock if no prescaler selected; prescaler output otherwise.

3: The arrows indicate the points in time where sampling occurs.

7.2 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module or as a postscaler for the Watchdog Timer (WDT), respectively (see **Section 8.7 "Watchdog Timer (WDT)"**). For simplicity, this counter is being referred to as "prescaler" throughout this data sheet.

Note:	The prescaler may be used by either the Timer0 module or the WDT, but not both. Thus, a prescaler assignment for the Timer0 module means that there is no prescaler for the WDT and vice versa.

The PSA and PS<2:0> bits of the OPTION register determine prescaler assignment and prescale ratio.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g., `CLRF TMR0`, `MOVWF TMR0`, etc.) will clear the prescaler. When assigned to WDT, a `CLRWDT` instruction will clear the prescaler along with the WDT. The prescaler is neither readable nor writable. On a Reset, the prescaler contains all '0's.

7.2.1 SWITCHING PRESCALER ASSIGNMENT

The prescaler assignment is fully under software control (i.e., it can be changed "on-the-fly" during program execution). To avoid an unintended device Reset, the following instruction sequence (Example 7-1) must be executed when changing the prescaler assignment from Timer0 to the WDT.

EXAMPLE 7-1: CHANGING PRESCALER (TIMER0 → WDT)

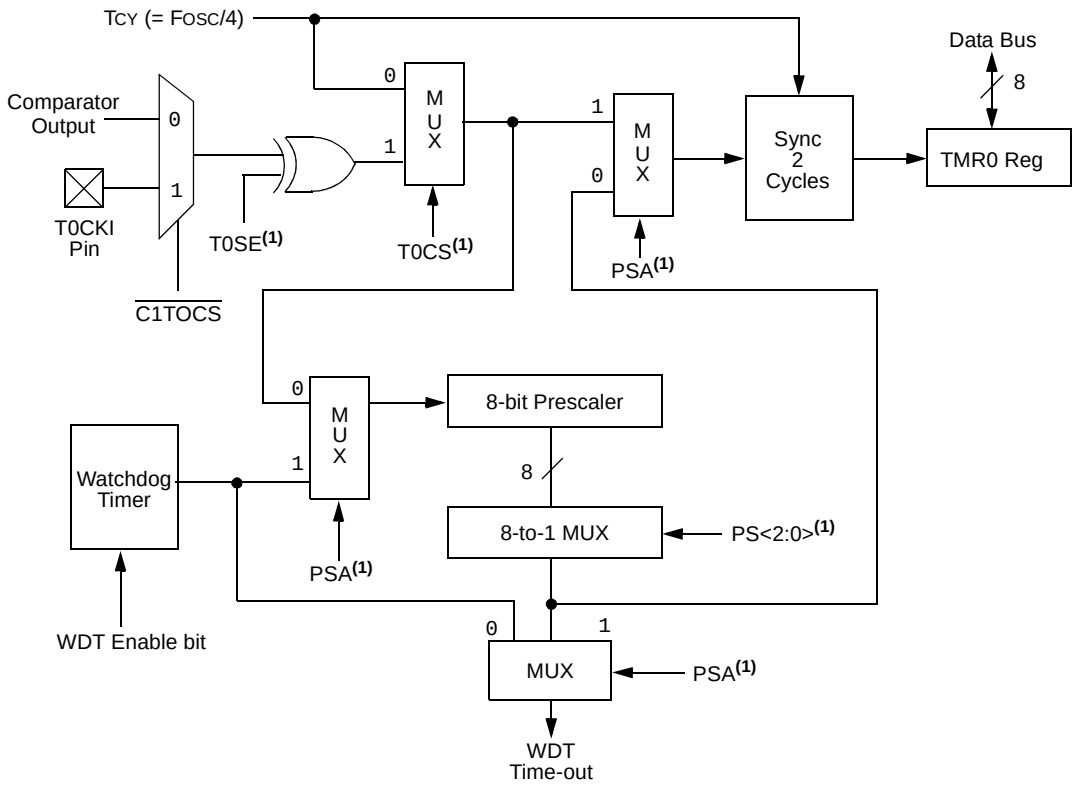
```
CLRWDT              ;Clear WDT
CLRF   TMR0         ;Clear TMR0 & Prescaler
MOVLW  b'00xx1111'
CLRWDT              ;PS<2:0> are 000 or 001
MOVLW  b'00xx1xxx'  ;Set Postscaler to
OPTION              ;desired WDT rate
```

To change the prescaler from the WDT to the Timer0 module, use the sequence shown in Example 7-2. This sequence must be used even if the WDT is disabled. A `CLRWDT` instruction should be executed before switching the prescaler.

EXAMPLE 7-2: CHANGING PRESCALER (WDT → TIMER0)

```
CLRWDT              ;Clear WDT and
                    ;prescaler
MOVLW  b'xxxx0xxx'  ;Select TMR0, new
                    ;prescale value and
                    ;clock source
OPTION
```


FIGURE 7-5: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



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

NOTES:

8.0 SPECIAL FEATURES OF THE CPU

What sets a microcontroller apart from other processors are special circuits that deal with the needs of real-time applications. The PIC16F527 microcontrollers have a host of such features intended to maximize system reliability, minimize cost through elimination of external components, provide power-saving operating modes and offer code protection. These features are:

- Oscillator Selection
- Reset:
  - Power-on Reset (POR)
  - Brown-out Reset (BOR)
  - Device Reset Timer (DRT)
  - Wake-up from Sleep on Pin Change
- Watchdog Timer (WDT)
- Sleep
- Code Protection
- ID Locations
- In-Circuit Serial Programming™
- Clock Out

The device has a Watchdog Timer, which can be shut off only through Configuration bit WDTE. The Watchdog Timer runs off of its own RC oscillator for added reliability.

There is also a Device Reset Timer (DRT), intended to keep the chip in Reset until the crystal oscillator is stable. The DRT can be enabled with the DRTEN Configuration bit. For the HS, XT or LP oscillator options, the 18 ms (nominal) delay is always provided by the Device Reset Timer and the DRTEN bit is ignored. When using the EC clock, INTRC or EXTRC oscillator options, there is a standard delay of 10 us on power-up, which can be extended to 18 ms with the use of the DRT timer. With the DRT timer on-chip, most applications require no additional external Reset circuitry.

The Sleep mode is designed to offer a very low current Power-Down mode. The user can wake-up from Sleep through a change on input pin or through a Watchdog Timer time-out. Several oscillator options are also made available to allow the part to fit the application, including an internal 4/8 MHz oscillator. The EXTRC oscillator option saves system cost while the LP crystal option saves power. A set of Configuration bits are used to select various options.

8.1 Configuration Bits

The PIC16F527 Configuration Words consist of 12 bits, although some bits may be unimplemented and read as '1'. Configuration bits can be programmed to select various device configurations. As an example, three bits are for the selection of the oscillator type; one bit is the Watchdog Timer enable bit, one bit is the $\overline{\text{MCLR}}$ enable bit and one bit is for code protection (Register 8-1).

8.2 Register Definitions — Configuration Word

REGISTER 8-1: CONFIG: CONFIGURATION WORD REGISTER

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
—	—	DRTEN	BOREN	$\overline{\text{CPSW}}$	IOSCFS	MCLRE	$\overline{\text{CP}}$	WDTE	FOSC2	FOSC1	FOSC0
bit 11											bit 0

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as '1'
'0' = Bit is cleared	'1' = Bit is set	-n = Value when blank or after Bulk Erase

bit 11-10 **Unimplemented:** Read as '1'

bit 9 **DRTEN:** Device Reset Timer Enable bit
- 1 = DRT Enabled (18 ms)
- 0 = DRT Disabled

bit 8 **BOREN:** Brown-out Reset Enable bit
- 1 = BOR Enabled
- 0 = BOR Disabled

bit 7 **$\overline{\text{CPSW}}$:** Code Protection bit – Self Writable Memory
- 1 = Code protection off
- 0 = Code protection on

bit 6 **IOSCFS:** Internal Oscillator Frequency Select bit
- 1 = 8 MHz INTOSC speed
- 0 = 4 MHz INTOSC speed

bit 5 **MCLRE:** Master Clear Enable bit
- 1 = RA3/$\overline{\text{MCLR}}$ pin functions as $\overline{\text{MCLR}}$
- 0 = RA3/$\overline{\text{MCLR}}$ pin functions as RA3, $\overline{\text{MCLR}}$ tied internally to V_{DD}

bit 4 **$\overline{\text{CP}}$:** Code Protection bit – User Program Memory
- 1 = Code protection off
- 0 = Code protection on

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

bit 2-0 **FOSC<2:0>:** Oscillator Selection bits
- 000 = LP oscillator and automatic 18 ms DRT (DRTEN fuse ignored)
- 001 = XT oscillator and automatic 18 ms DRT (DRTEN fuse ignored)
- 010 = HS oscillator and automatic 18 ms DRT (DRTEN fuse ignored)
- 011 = EC oscillator with RA4 function on RA4/OSC2/CLKOUT and 10 us start-up time(2,3)
- 100 = INTRC with RA4 function on RA4/OSC2/CLKOUT and 10 us start-up time(2,3)
- 101 = INTRC with CLKOUT function on RA4/OSC2/CLKOUT and 10 us start-up time(2,3)
- 110 = EXTRC with RA4 function on RA4/OSC2/CLKOUT and 10 us start-up time(2,3)
- 111 = EXTRC with CLKOUT function on RA4/OSC2/CLKOUT and 10 us start-up time(2,3)

Note 1: Refer to the "*PIC16F527 Memory Programming Specification*", DS41640 to determine how to access the Configuration Word.

2: DRT length and start-up time are functions of the Clock mode selection. It is the responsibility of the application designer to ensure the use of either will result in acceptable operation. Refer to **Section 15.0 "Electrical Characteristics"** for V_{DD} rise time and stability requirements for this mode of operation.

3: The optional DRTEN fuse can be used to extend the start-up time to 18 ms.

8.3 Oscillator Configurations

8.3.1 OSCILLATOR TYPES

The PIC16F527 device can be operated in up to six different oscillator modes. The user can program up to three Configuration bits (FOSC<2:0>). To select one of these modes:

- LP: Low-Power Crystal
- XT: Crystal/Resonator
- HS: High-Speed Crystal/Resonator
- INTRC: Internal 4/8 MHz Oscillator
- EXTRC: External Resistor/Capacitor
- EC: External High-Speed Clock Input

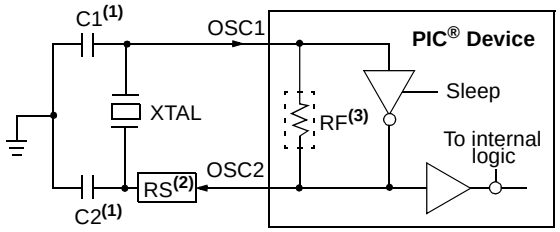
8.3.2 CRYSTAL OSCILLATOR/CERAMIC RESONATORS

In HS, XT or LP modes, a crystal or ceramic resonator is connected to the OSC1/CLKIN and OSC2/CLKOUT pins to establish oscillation (Figure 8-1). The PIC16F527 oscillator designs require the use of a parallel cut crystal. Use of a series cut crystal may give a frequency out of the crystal manufacturers specifications. When in HS, XT or LP modes, the device can have an external clock source drive the OSC1/CLKIN pin (Figure 8-2). In this mode, the output drive levels on the OSC2 pin are very weak. If the part is used in this fashion, then this pin should be left open and unloaded. Also when using this mode, the external clock should observe the frequency limits for the Clock mode chosen (HS, XT or LP).

Note 1: This device has been designed to perform to the parameters of its data sheet. It has been tested to an electrical specification designed to determine its conformance with these parameters. Due to process differences in the manufacture of this device, this device may have different performance characteristics than its earlier version. These differences may cause this device to perform differently in your application than the earlier version of this device.

2: The user should verify that the device oscillator starts and performs as expected. Adjusting the loading capacitor values and/or the Oscillator mode may be required.

FIGURE 8-1: CRYSTAL OPERATION (OR CERAMIC RESONATOR) (HS, XT OR LP OSC CONFIGURATION)

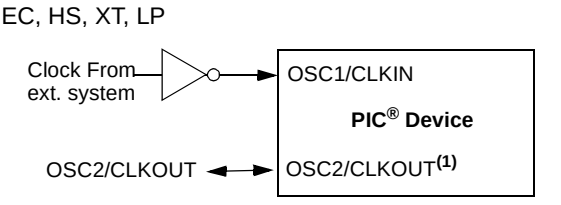


Note 1: See Capacitor Selection tables for recommended values of C1 and C2.

2: A series resistor (RS) may be required for AT strip cut crystals.

3: RF approx. value = 10 MΩ.

FIGURE 8-2: EXTERNAL CLOCK INPUT OPERATION (HS, XT, LP OR EC OSC CONFIGURATION)



Note 1: Available in EC mode only.

TABLE 8-1: CAPACITOR SELECTION FOR CERAMIC RESONATORS

Osc Type	Resonator Freq.	Cap. Range C1	Cap. Range C2
XT	4.0 MHz	30 pF	30 pF
HS	16 MHz	10-47 pF	10-47 pF

Note 1: These values are for design guidance only. Since each resonator has its own characteristics, the user should consult the resonator manufacturer for appropriate values of external components.

TABLE 8-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR(2)

Osc Type	Resonator Freq.	Cap. Range C1	Cap. Range C2
LP	32 kHz(1)	15 pF	15 pF
XT	200 kHz 1 MHz 4 MHz	47-68 pF 15 pF 15 pF	47-68 pF 15 pF 15 pF
HS	20 MHz	15-47 pF	15-47 pF

Note 1: For V_{DD} > 4.5V, C1 = C2 ≈ 30 pF is recommended.

2: These values are for design guidance only. Rs may be required to avoid overdriving crystals with low drive level specification. Since each crystal has its own characteristics, the user should consult the crystal manufacturer for appropriate values of external components.

8.3.3 EXTERNAL CRYSTAL OSCILLATOR CIRCUIT

Either a prepackaged oscillator or a simple oscillator circuit with TTL gates can be used as an external crystal oscillator circuit. Prepackaged oscillators provide a wide operating range and better stability. A well-designed crystal oscillator will provide good performance with TTL gates. Two types of crystal oscillator circuits can be used: one with parallel resonance, or one with series resonance.

Figure 8-3 shows implementation of a parallel resonant oscillator circuit. The circuit is designed to use the fundamental frequency of the crystal. The 74AS04 inverter performs the 180-degree phase shift that a parallel oscillator requires. The 4.7 kΩ resistor provides the negative feedback for stability. The 10 kΩ potentiometers bias the 74AS04 in the linear region. This circuit could be used for external oscillator designs.

FIGURE 8-3: EXTERNAL PARALLEL RESONANT CRYSTAL OSCILLATOR CIRCUIT

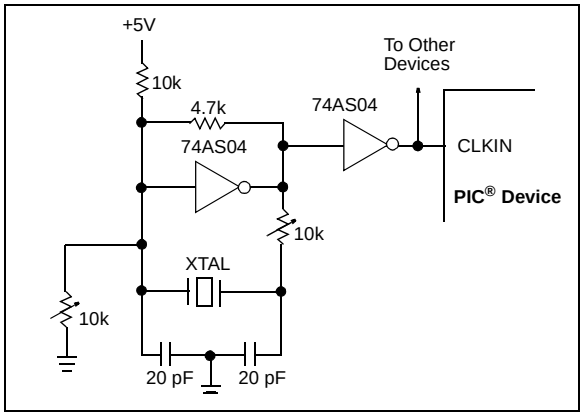
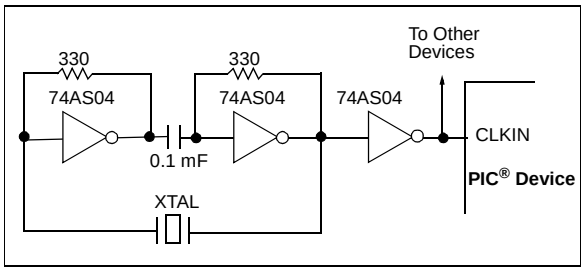


Figure 8-4 shows a series resonant oscillator circuit. This circuit is also designed to use the fundamental frequency of the crystal. The inverter performs a 180-degree phase shift in a series resonant oscillator circuit. The 330 Ω resistors provide the negative feedback to bias the inverters in their linear region.

FIGURE 8-4: EXTERNAL SERIES RESONANT CRYSTAL OSCILLATOR CIRCUIT



8.3.4 EXTERNAL RC OSCILLATOR

For timing insensitive applications, the RC device option offers additional cost savings. The RC oscillator frequency is a function of the supply voltage, the resistor (REXT) and capacitor (CEXT) values, and the operating temperature. In addition to this, the oscillator frequency will vary from unit-to-unit due to normal process parameter variation. Furthermore, the difference in lead frame capacitance between package types will also affect the oscillation frequency, especially for low CEXT values. The user also needs to take into account variation due to tolerance of external R and C components used.

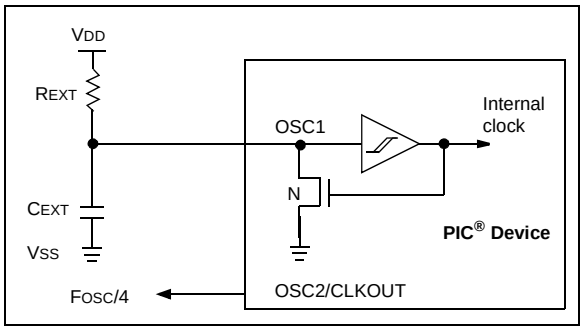
Figure 8-5 shows how the R/C combination is connected to the PIC16F527 device. For REXT values below 3.0 kΩ, the oscillator operation may become unstable, or stop completely. For very high REXT values (e.g., 1 MΩ), the oscillator becomes sensitive to noise, humidity and leakage. Thus, we recommend keeping REXT between 5.0 kΩ and 100 kΩ.

Although the oscillator will operate with no external capacitor (CEXT = 0 pF), we recommend using values above 20 pF for noise and stability reasons. With no external capacitance or with values below 20 pF, the oscillation frequency can vary dramatically due to changes in external capacitances, such as PCB trace capacitance or package lead frame capacitance.

Section 15.0 "Electrical Characteristics" shows RC frequency variation from part-to-part due to normal process variation. The variation is larger for larger values of R (since leakage current variation will affect RC frequency more for large R) and for smaller values of C (since variation of input capacitance will affect RC frequency more).

Also, see the Electrical Specifications section for variation of oscillator frequency due to VDD for given REXT/CEXT values, as well as frequency variation due to operating temperature for given R, C and VDD values.

FIGURE 8-5: EXTERNAL RC OSCILLATOR MODE



8.3.5 INTERNAL 4/8 MHz RC OSCILLATOR

The internal RC oscillator provides a fixed 4/8 MHz (nominal) system clock at VDD = 5V and 25°C, (see **Section 15.0 "Electrical Characteristics"** for information on variation over voltage and temperature).

In addition, a calibration instruction is programmed into the last address of memory, which contains the calibration value for the internal RC oscillator. This location is always non-code protected, regardless of the code-protect settings. This value is programmed as a `MOVLW XX` instruction where XX is the calibration value, and is placed at the Reset vector. This will load the W register with the calibration value upon Reset and the PC will then roll over to the users program at address 0x000. The user then has the option of writing the value to the OSCCAL Register or ignoring it.

OSCCAL, when written to with the calibration value, will "trim" the internal oscillator to remove process variation from the oscillator frequency.

Note:	Erasing the device will also erase the pre-programmed internal calibration value for the internal oscillator. The calibration value must be read prior to erasing the part so it can be reprogrammed correctly later.

For the PIC16F527 device, only bits <7:1> of OSCCAL are used for calibration. See Register 4-3 for more information.

Note:	The bit 0 of the OSCCAL register is unimplemented and should be written as '0' when modifying OSCCAL for compatibility with future devices.

8.4 Reset

The device differentiates between various kinds of Reset:

- Power-on Reset (POR)
- Brown-out Reset (BOR)
- $\overline{MCLR}$ Reset during normal operation
- $\overline{MCLR}$ Reset during Sleep
- WDT Time-out Reset during normal operation
- WDT Time-out Reset during Sleep
- Wake-up from Sleep on pin change

Some registers are not reset in any way, they are unknown on POR/BOR and unchanged in any other Reset. Most other registers are reset to “Reset state” on Power-on Reset (POR)/Brown-out Reset (BOR), $\overline{MCLR}$, WDT or Wake-up on pin change Reset during normal operation. They are not affected by a WDT Reset during Sleep or $\overline{MCLR}$ Reset during Sleep, since these Resets are viewed as resumption of normal operation. The exceptions to this are the $\overline{TO}$ and $\overline{PD}$ bits. They are set or cleared differently in different Reset situations. These bits are used in software to determine the nature of Reset. See Table 4-1 for a full description of Reset states of all registers.

TABLE 8-3: RESET CONDITION FOR SPECIAL REGISTERS

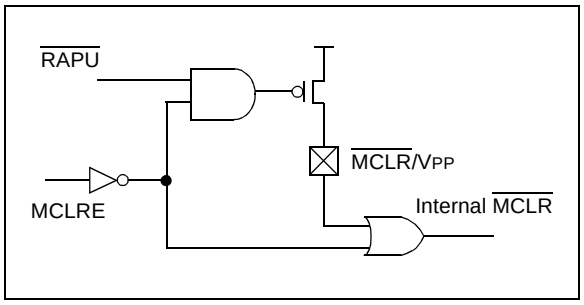
	STATUS Addr: 03h
Power-on Reset (POR) or Brown-out Reset (BOR)	0001 1xxx
$\overline{MCLR}$ Reset during normal operation	000u uuuu
$\overline{MCLR}$ Reset during Sleep	0001 0uuu
WDT Reset during Sleep	0000 0uuu
WDT Reset normal operation	0000 uuuu
Wake-up from Sleep on pin change	1001 0uuu
Wake-up from Sleep on comparator change	0101 0uuu

Legend: u = unchanged, x = unknown, – = unimplemented bit, read as ‘0’.

8.4.1 MCLR ENABLE

This Configuration bit, when set to a '1', enables the external MCLR Reset function. When cleared to '0', the MCLR function is tied to the internal VDD and the pin is assigned to be an input-only pin function. See Figure 8-6.

FIGURE 8-6: MCLR SELECT



8.5 Power-on Reset (POR)

The PIC16F527 device incorporates an on-chip Power-on Reset (POR) circuitry, which provides an internal chip Reset for most power-up situations.

The on-chip POR circuit holds the chip in Reset until VDD has reached a high enough level for proper operation. To take advantage of the internal POR, program the MCLR/VPP pin as MCLR and tie through a resistor to VDD, or program the pin as an input pin. An internal weak pull-up resistor is implemented using a transistor (refer to Table 15-7 for the pull-up resistor ranges). This will eliminate external RC components usually needed to create a Power-on Reset. A maximum rise time for VDD is specified. See **Section 15.0 "Electrical Characteristics"** for details.

When the device starts normal operation (exit the Reset condition), device operating parameters (voltage, frequency, temperature,...) must be met to ensure operation. If these conditions are not met, the device must be held in Reset until the operating parameters are met.

A simplified block diagram of the on-chip Power-on Reset circuit is shown in Figure 8-7.

The Power-on Reset circuit and the Device Reset Timer (see **Section 8.6 "Device Reset Timer (DRT)"**) circuit are closely related. On power-up, the Reset latch is set and the DRT is reset. The DRT timer begins counting once it detects MCLR to be high. After the time-out period, it will reset the Reset latch and thus end the on-chip Reset signal.

A power-up example where MCLR is held low is shown in Figure 8-8. VDD is allowed to rise and stabilize before bringing MCLR high. The chip will actually come out of Reset TDRT msec after MCLR goes high.

In Figure 8-9, the on-chip Power-on Reset feature is being used (MCLR and VDD are tied together or the pin is programmed to be an input pin). The VDD is stable before the start-up timer times out and there is no problem in getting a proper Reset. However, Figure 8-10 depicts a problem situation where VDD rises too slowly. The time between when the DRT senses that MCLR is high and when MCLR and VDD actually reach their full value, is too long. In this situation, when the start-up timer times out, VDD has not reached the VDD (min) value and the chip may not function correctly. For such situations, we recommend that external RC circuits be used to achieve longer POR delay times (Figure 8-9).

Note:	When the device starts normal operation (exit the Reset condition), device operating parameters (voltage, frequency, temperature, etc.) must be met to ensure operation. If these conditions are not met, the device must be held in Reset until the operating conditions are met.

For additional information, refer to Application Notes AN522, *"Power-Up Considerations"* (DS00522) and AN607, *"Power-up Trouble Shooting"* (DS00607).

FIGURE 8-7: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT

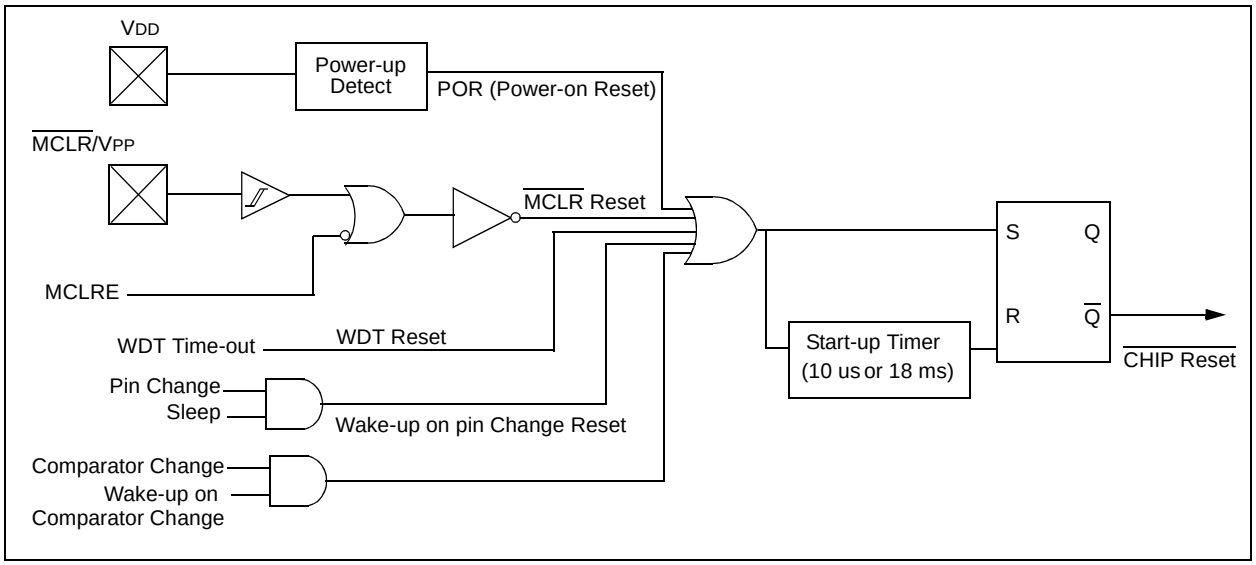


FIGURE 8-8: TIME-OUT SEQUENCE ON POWER-UP (MCLR PULLED LOW)

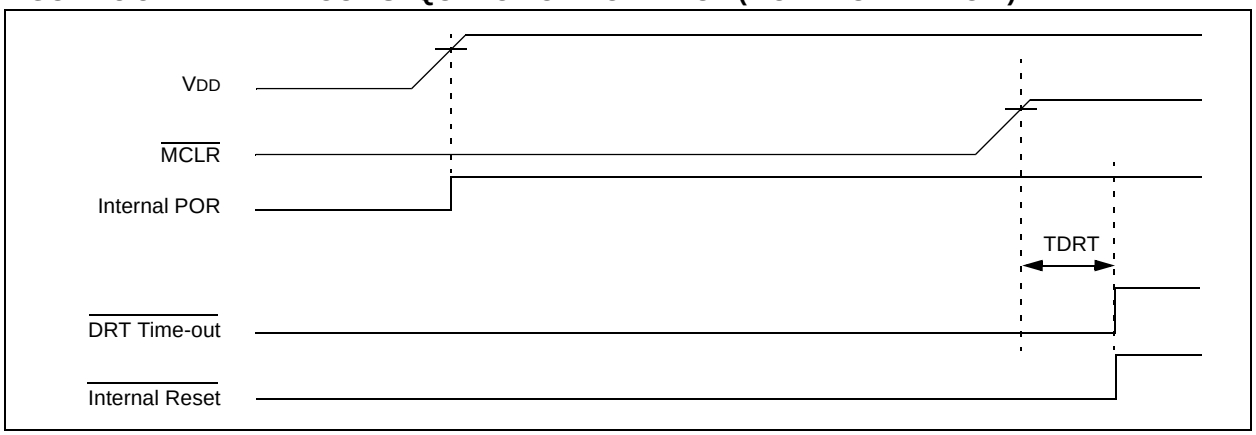


FIGURE 8-9: TIME-OUT SEQUENCE ON POWER-UP (MCLR TIED TO VDD): FAST VDD RISE TIME

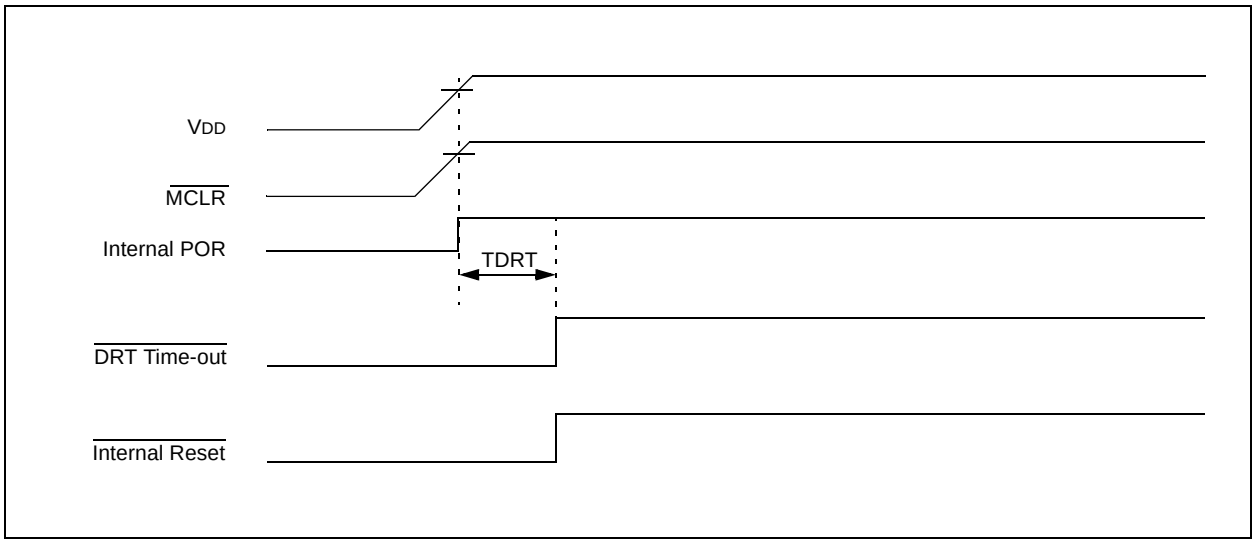
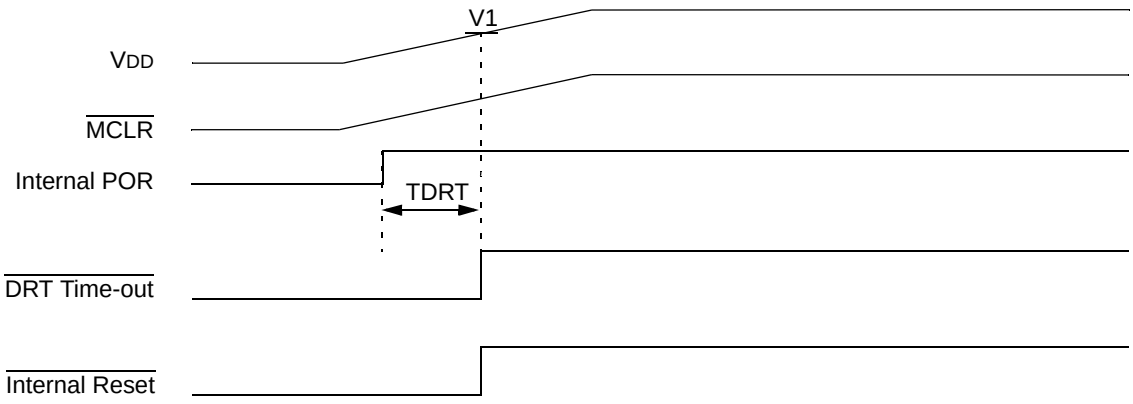


FIGURE 8-10: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ TIED TO V_{DD}): SLOW V_{DD} RISE TIME



Note: When V_{DD} rises slowly, the T_{DRT} time-out expires long before V_{DD} has reached its final value. In this example, the chip will reset properly if, and only if, V1 ≥ V_{DD} min.

8.6 Device Reset Timer (DRT)

On the PIC16F527 device, the DRT runs any time the device is powered up. DRT runs from Reset and varies based on oscillator selection and Reset type (see Table 8-4).

The DRT operates on an internal RC oscillator. The processor is kept in Reset as long as the DRT is active. The DRT delay allows VDD to rise above VDD min. and for the oscillator to stabilize.

Oscillator circuits based on crystals or ceramic resonators require a certain time after power-up to establish a stable oscillation. The on-chip DRT keeps the device in a Reset condition after MCLR has reached a logic high (VIH MCLR) level. Programming MCLR/VPP as MCLR and using an external RC network connected to the MCLR input is not required in most cases. This allows savings in cost-sensitive and/or space restricted applications, as well as allowing the use of that pin as a general purpose input.

The Device Reset Time delays will vary from chip-to-chip due to VDD, temperature and process variation. See AC parameters for details.

The DRT will also be triggered upon a Watchdog Timer time-out from Sleep. This is particularly important for applications using the WDT to wake from Sleep mode automatically.

Reset sources are POR, MCLR, WDT time-out and wake-up on pin or comparator change. See **Section 8.10.2 "Wake-up from Sleep", Notes 1, 2** and **3**.

8.7 Watchdog Timer (WDT)

The Watchdog Timer (WDT) is a free running on-chip RC oscillator, which does not require any external components. This RC oscillator is separate from the external RC oscillator of the OSC1/CLKIN pin and the internal 4/8 MHz oscillator. This means that the WDT will run even if the main processor clock has been stopped, for example, by execution of a SLEEP instruction. During normal operation or Sleep, a WDT Reset or wake-up Reset, generates a device Reset.

The TO bit of the STATUS register will be cleared upon a Watchdog Timer Reset.

The WDT can be permanently disabled by programming the configuration WDTE as a '0' (see **Section 8.1 "Configuration Bits"**). Refer to the PIC16F527 Programming Specifications to determine how to access the Configuration Word.

TABLE 8-4: TYPICAL DRT PERIODS

Oscillator Configuration	POR Reset	Subsequent Resets
HS, XT, LP	18 ms	18 ms
EC	10 us	10 µs
INTOSC, EXTRC	10 us	10 µs

8.7.1 WDT PERIOD

The WDT has a nominal time-out period of 18 ms, (with no prescaler). If a longer time-out period is desired, a prescaler with a division ratio of up to 1:128 can be assigned to the WDT (under software control) by writing to the OPTION register. Thus, a time-out period of a nominal 2.3 seconds can be realized. These periods vary with temperature, VDD and part-to-part process variations (see DC specs).

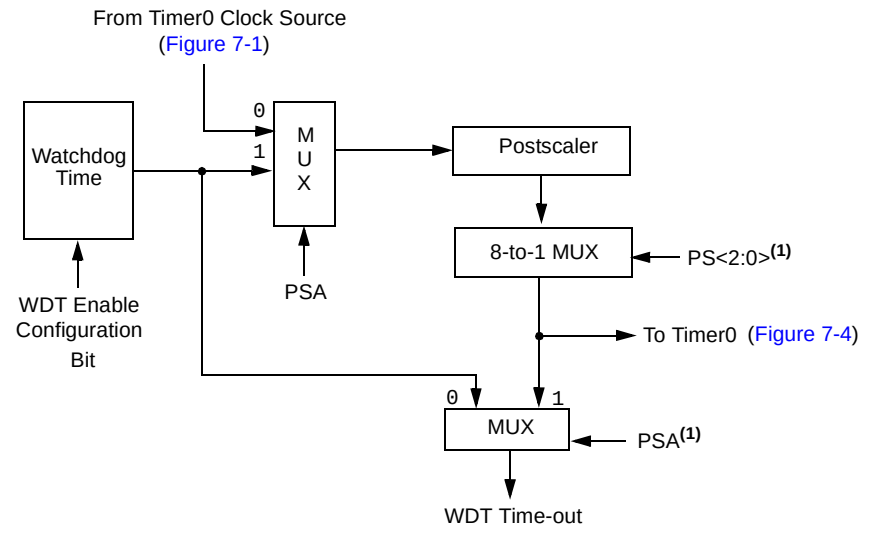
Under worst-case conditions (VDD = Min., Temperature = Max., max. WDT prescaler), it may take several seconds before a WDT time-out occurs.

8.7.2 WDT PROGRAMMING CONSIDERATIONS

The CLRWDT instruction clears the WDT and the postscaler, if assigned to the WDT, and prevents it from timing out and generating a device Reset.

The SLEEP instruction resets the WDT and the postscaler, if assigned to the WDT. This gives the maximum Sleep time before a WDT wake-up Reset.

FIGURE 8-11: WATCHDOG TIMER BLOCK DIAGRAM



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

TABLE 8-5: REGISTERS ASSOCIATED WITH THE WATCHDOG TIMER

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
OPTION	$\overline{\text{RAWU}}$	$\overline{\text{RAPU}}$	T0SC	T0SE	PSA	PS2	PS1	PS0	20

Legend: Shaded boxes = Not used by Watchdog Timer.

8.8 Time-out Sequence ($\overline{\text{TO}}$) and Power-down ($\overline{\text{PD}}$) Reset Status

The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits in the STATUS register can be tested to determine if a Reset condition has been caused by a power-up condition, a $\overline{\text{MCLR}}$ or Watchdog Timer (WDT) Reset.

TABLE 8-6: $\overline{\text{TO}}$/$\overline{\text{PD}}$ STATUS AFTER RESET

$\overline{\text{TO}}$	$\overline{\text{PD}}$	Reset Caused By
0	0	WDT wake-up from Sleep
0	u	WDT time-out (not from Sleep)
1	0	$\overline{\text{MCLR}}$ wake-up from Sleep
1	1	Power-up or Brown-out Reset
u	u	$\overline{\text{MCLR}}$ not during Sleep

Legend: u = unchanged

Note 1: The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits maintain their status (u) until a Reset occurs. A low pulse on the $\overline{\text{MCLR}}$ input does not change the $\overline{\text{TO}}$ and $\overline{\text{PD}}$ Status bits.

8.9 Brown-out Reset (BOR)

A brown-out is a condition where device power (V_{DD}) dips below its minimum value, but not to zero, and then recovers. The device should be reset in the event of a brown-out. The Brown-out Reset feature is enabled by the BOREN Configuration bit.

If V_{DD} falls below V_{BOR} for greater than parameter (T_{BOR}) (see Figure 8-12), the brown-out situation will reset the device. This will occur regardless of V_{DD} slew rate. A Reset is not insured to occur if V_{DD} falls below V_{BOR} for less than parameter (T_{BOR}).

On any Reset (Power-on, Brown-out Reset, Watchdog Timer, etc.), the chip will remain in Reset until V_{DD} rises above V_{BOR} (see Figure 8-12). If enabled, the Device Reset Timer will now be invoked, and will keep the chip in Reset an additional 18 ms.

Note:	The Device Reset Timer is enabled by the DRTEN bit in the Configuration Word register.

If V_{DD} drops below V_{BOR} while the Device Reset Timer is running, the chip will go back into a Brown-out Reset and the Device Reset Timer will be re-initialized. Once V_{DD} rises above V_{BOR}, the Device Reset Timer will execute a 18 ms Reset.

FIGURE 8-12: BROWN-OUT RESET TIMING AND CHARACTERISTICS

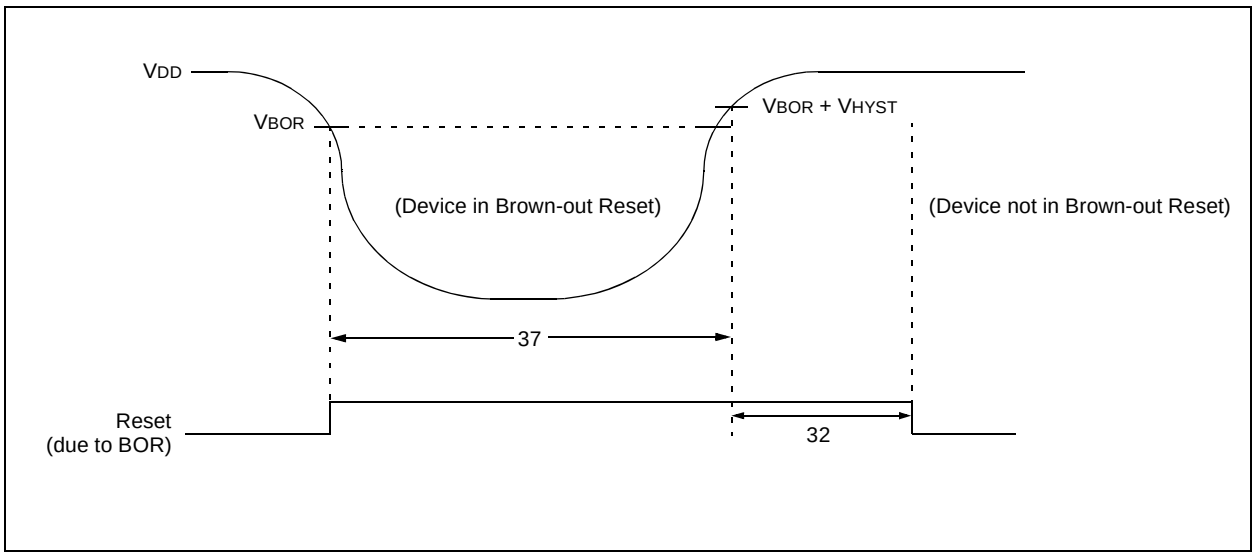
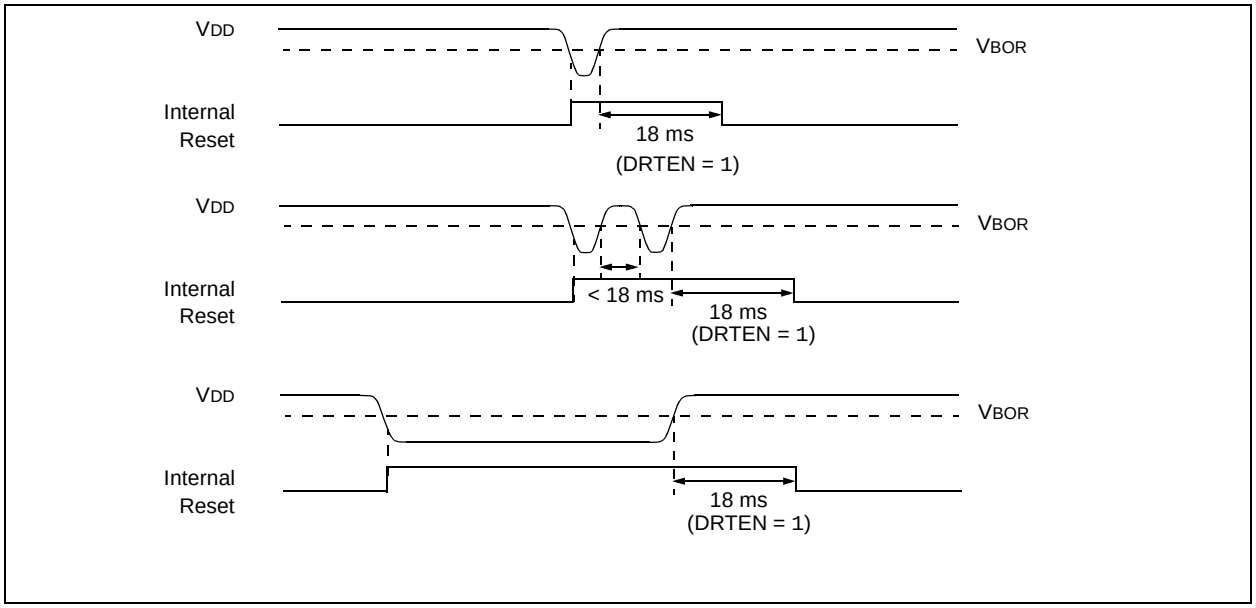


FIGURE 8-13: BROWN-OUT SITUATIONS



8.10 Power-down Mode (Sleep)

A device may be powered down (Sleep) and later powered up (wake-up from Sleep).

8.10.1 SLEEP

The Power-Down mode is entered by executing a SLEEP instruction.

If enabled, the Watchdog Timer will be cleared but keeps running, the $\overline{TO}$ bit of the STATUS register is set, the $\overline{PD}$ bit of the STATUS register is cleared and the oscillator driver is turned off. The I/O ports maintain the status they had before the SLEEP instruction was executed (driving high, driving low or high-impedance).

Note:	A Reset generated by a WDT time-out does not drive the $\overline{MCLR}$ pin low.

For lowest current consumption while powered down, the T0CKI input should be at V_{DD} or V_{SS} and the $\overline{MCLR}/V_{PP}$ pin must be at a logic high level if $\overline{MCLR}$ is enabled.

8.10.2 WAKE-UP FROM SLEEP

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

1. An external Reset input on RB3/$\overline{MCLR}/V_{PP}$ pin, when configured as $\overline{MCLR}$.
2. A Watchdog Timer Time-out Reset (if WDT was enabled).
3. From an interrupt source, see **Section 8.11 "Interrupts"** for more information.

On waking from Sleep, the processor will continue to execute the instruction immediately following the SLEEP instruction. If the WUR bit is also set, upon waking from Sleep, the device will reset. If the GIE bit is also set, upon waking from Sleep, the processor will branch to the interrupt vector. Please see **Section 8.11 "Interrupts"** for more information.

The $\overline{TO}$ and $\overline{PD}$ bits can be used to determine the cause of the device Reset. The $\overline{TO}$ bit is cleared if a WDT time-out occurred and subsequently caused a wake-up. The $\overline{PD}$ bit, which is set on power-up, is cleared when SLEEP is invoked.

.

Note:	**Caution:** Right before entering Sleep, read the input pins. When in Sleep, wake-up occurs when the values at the pins change from the state they were in at the last reading. If a wake-up on change occurs and the pins are not read before re-entering Sleep, a wake-up will occur immediately even if no pins change while in Sleep mode.

The WDT is cleared when the device wakes from Sleep, regardless of the wake-up source.

Note:	**Caution:** Right before entering Sleep, read the comparator Configuration register(s) CM1CON0 and CM2CON0. When in Sleep, wake-up occurs when the comparator output bit C1OUT and C2OUT change from the state they were in at the last reading. If a wake-up on comparator change occurs and the pins are not read before re-entering Sleep, a wake-up will occur immediately, even if no pins change while in Sleep mode.

8.11 Interrupts

The interrupt feature allows certain events to preempt normal program flow. Firmware is used to determine the source of the interrupt and act accordingly. Some interrupts can be configured to wake the MCU from Sleep mode.

These following interrupt sources are available on the PIC16F527 device:

- Timer0 Overflow
- ADC Completion
- Comparator Output Change
- Interrupt-on-change pin

Refer to the corresponding chapters for details.

8.12 Operation

Interrupts are disabled upon any device Reset. They are enabled by setting the following bits:

- GIE bit of the INTCON register
- Interrupt Enable bit(s) for the specific interrupt event(s)

The enable bits for specific interrupts can be found in the INTCON1 register. An interrupt is recorded for a specific interrupt via flag bits found in the INTCON0 register.

The ADC Conversion flag and the Timer0 Overflow flags will be set regardless of the status of the GIE and individual interrupt enable bits.

The Comparator and Interrupt-on-change flags must be enabled for use. One or both of the comparator outputs can be enabled to affect the interrupt flag by setting the C1WU bit in the CM1CON0 register and the C2WU bit in the CM2CON0 register. The Interrupt-on-change flag is enabled by setting the RAWU bit in the OPTION register.

The following events happen when an interrupt event occurs while the GIE bit is set:

- Current prefetched instruction is flushed
- GIE bit is cleared
- Current Program Counter (PC) is pushed onto the stack
- Several registers are automatically switched to a secondary set of registers to store critical data. (See **Section 8.13 "Automatic Context Switching"**)
- PC is loaded with the interrupt vector 0004h

The firmware within the Interrupt Service Routine (ISR) should determine the source of the interrupt by polling the interrupt flag bits. The interrupt flag bits must be cleared before exiting the ISR to avoid repeated interrupts. Because the GIE bit is cleared, any interrupt that occurs while executing the ISR will be recorded through its interrupt flag, but will not cause the processor to redirect to the interrupt vector.

8.13 Automatic Context Switching

While the device is executing from the ISR, a secondary set of W, STATUS, FSR and BSR registers are used by the CPU. These registers are still addressed at the same location, but hold persistent, independent values for use inside the ISR. This allows the contents of the primary set of registers to be unaffected by interrupts in the main line execution. The contents of the secondary set of context registers are visible in the SFR map as the IW, ISTATUS, IFSR and IBSR registers. When executing code from within the ISR, these registers will read back the main line context, and vice versa.

The RETFIE instruction exits the ISR by popping the previous address from the stack, switching back to the original set of critical registers and setting the GIE bit.

For additional information on a specific interrupt's operation, refer to its peripheral chapter.

Note 1:	Individual interrupt flag bits may be set, regardless of the state of any other enable bits.
2:	All interrupts will be ignored while the GIE bit is cleared. Any interrupt occurring while the GIE bit is clear will be serviced when the GIE bit is set again.

8.14 Interrupts during Sleep

Any of the interrupt sources can be used to wake from Sleep. To wake from Sleep, the peripheral must be operating without the system clock. The interrupt source must have the appropriate Interrupt Enable bit(s) set prior to entering Sleep.

On waking from Sleep, if the GIE bit is also set, the processor will branch to the interrupt vector. Otherwise, the processor will continue executing instructions after the SLEEP instruction. The instruction directly after the SLEEP instruction will always be executed before branching to the ISR. Refer to the **Section 8.10 "Power-down Mode (Sleep)"** for more details.

TABLE 8-7: INTERRUPT PRIORITIES

	In Sleep	GIE	WUR
Vector or Wake-up and Vector	X	1	0
Wake-up Reset	1	X	1
Wake-up Inline	1	0	0
Watchdog Wake-up Inline	1	X	0
Watchdog Wake-up Reset	1	X	1

8.15 Register Definitions — Interrupt Control

REGISTER 8-2: INTCON0 REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0
ADIF	CWIF	T0IF	RAIF	—	—	—	GIE
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 **ADIF:** A/D Converter Interrupt Flag bit
- 1 = A/D conversion complete (must be cleared by software)
- 0 = A/D conversion has not completed or has not been started

bit 6 **CWIF:** Comparator 1 or 2 Interrupt Flag bit
- 1 = Comparator interrupt-on-change has occurred(1)
- 0 = No change in Comparator 1 or 2 output

bit 5 **T0IF:** Timer0 Overflow Interrupt Flag bit
- 1 = TMR0 register has overflowed (must be cleared by software)
- 0 = TMR0 register did not overflow

bit 4 **RAIF:** Port A Interrupt-on-change Flag bit
- 1 = Wake-up or interrupt has occurred (cleared in software)(2)
- 0 = Wake-up or interrupt has not occurred

bit 3-1 **Unimplemented:** Read as '0'

bit 0 **GIE:** Global Interrupt Enable bit
- 1 = Interrupt sets PC to address 0x004 (Vector to ISR)
- 0 = Interrupt causes wake-up and inline code execution

Note 1: This bit only functions when the $\overline{\text{C1WU}}$ or $\overline{\text{C2WU}}$ bits are set (see Register 10-1 and Register 10-2).
2: The $\overline{\text{RAWU}}$ bit of the OPTION register must be set to enable this function (see Register 4-2).

REGISTER 8-3: INTCON1 REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0
ADIE	CWIE	T0IE	RAIE	—	—	—	WUR
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 **ADIE:** A/D Converter (ADC) Interrupt Enable bit
- 1 = Enables the ADC interrupt
- 0 = Disables the ADC interrupt

bit 6 **CWIE:** Comparator 1 and 2 Interrupt Enable bit
- 1 = Enables the Comparator 1 and 2 Interrupt
- 0 = Disables the Comparator 1 and 2 Interrupt

bit 5 **T0IE:** Timer0 Overflow Interrupt Enable bit
- 1 = Enables the Timer0 interrupt
- 0 = Disables the Timer0 interrupt

bit 4 **RAIE:** Port A on Pin Change Interrupt Enable bit
- 1 = Interrupt-on-change pin enabled
- 0 = Interrupt-on-change pin disabled

bit 3-1 **Unimplemented:** Read as '0'

bit 0 **WUR:** Wake-up Reset Enable bit
- 1 = Interrupt source causes device Reset on wake-up
- 0 = Interrupt source wakes up device from Sleep (Vector to ISR or inline execution)

8.16 Program Verification/Code Protection

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

The first 64 locations and the last location (OSCCAL) can be read, regardless of the code protection bit setting.

8.17 ID Locations

Four memory locations 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.

Use only the lower four bits of the ID locations and always program the upper eight bits as '0's.

8.18 In-Circuit Serial Programming™

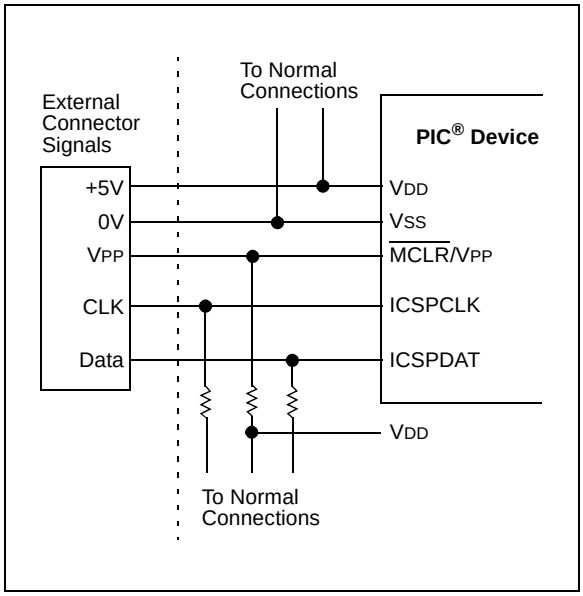
The PIC16F527 microcontroller can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data, and three other lines for power, ground and the programming voltage. This allows customers to manufacture boards with unprogrammed devices and then program the microcontroller just before shipping the product. This also allows the most recent firmware, or a custom firmware, to be programmed.

The devices are placed into a Program/Verify mode by holding the ICSPCLK and ICSPDAT pins low while raising the $\overline{MCLR}$ (V_{PP}) pin from V_{IL} to V_{IHH} (see programming specification). ICSPCLK becomes the programming clock and ICSPDAT becomes the programming data. Both ICSPCLK and ICSPDAT are Schmitt Trigger inputs in this mode.

After Reset, a 6-bit command is then supplied to the device. Depending on the command, 14 bits of program data are then supplied to or from the device, depending if the command was a load or a read. For complete details of serial programming, please refer to the PIC16F527 Programming Specifications.

A typical In-Circuit Serial Programming connection is shown in Figure 8-14.

FIGURE 8-14: TYPICAL IN-CIRCUIT SERIAL PROGRAMMING CONNECTION



NOTES:

9.0 ANALOG-TO-DIGITAL (A/D) CONVERTER

The A/D Converter allows conversion of an analog signal into an 8-bit digital signal.

9.1 Clock Divisors

The ADC has four clock source settings ADCS<1:0>. There are three divisor values 16, 8 and 4. The fourth setting is INTOSC with a divisor of four. These settings will allow a proper conversion when using an external oscillator at speeds from 20 MHz to 350 kHz. Using an external oscillator at a frequency below 350 kHz requires the ADC oscillator setting to be INTOSC/4 (ADCS<1:0> = 11) for valid ADC results.

The ADC requires 13 TAD periods to complete a conversion. The divisor values do not affect the number of TAD periods required to perform a conversion. The divisor values determine the length of the TAD period.

When the ADCS<1:0> bits are changed while an ADC conversion is in process, the new ADC clock source will not be selected until the next conversion is started. This clock source selection will be lost when the device enters Sleep.

Note:	The ADC clock is derived from the instruction clock. The ADCS divisors are then applied to create the ADC clock

9.1.1 VOLTAGE REFERENCE

There is no external voltage reference for the ADC. The ADC reference voltage will always be VDD.

9.1.2 ANALOG MODE SELECTION

The ANS<7:0> bits are used to configure pins for analog input. Upon any Reset, ANS<7:0> defaults to 11. This configures pins AN0, AN1 and AN2 as analog inputs. Pins configured as analog inputs are not available for digital output. Users should not change the ANS bits while a conversion is in process. ANS bits are active regardless of the condition of ADON.

9.1.3 ADC CHANNEL SELECTION

The CHS bits are used to select the analog channel to be sampled by the ADC. The CHS<3:0> bits can be changed at any time without adversely effecting a conversion. To acquire an analog signal the CHS<3:0> selection must match one of the pin(s) selected by the ANS<7:0> bits. When the ADC is on (ADON = 1) and a channel is selected that is also being used by the comparator, then both the comparator and the ADC will see the analog voltage on the pin.

Note:	It is the users responsibility to ensure that use of the ADC and comparator simultaneously on the same pin, does not adversely affect the signal being monitored or adversely effect device operation.

When the CHS<3:0> bits are changed during an ADC conversion, the new channel will not be selected until the current conversion is completed. This allows the current conversion to complete with valid results. All channel selection information will be lost when the device enters Sleep.

TABLE 9-1: CHANNEL SELECT (ADCS) BITS AFTER AN EVENT

Event	ADCS<1:0>
$\overline{\text{MCLR}}$	11
Conversion completed	CS<1:0>
Conversion terminated	CS<1:0>
Power-on	11
Wake from Sleep	11

9.1.4 THE GO/$\overline{\text{DONE}}$ BIT

The GO/$\overline{\text{DONE}}$ bit is used to determine the status of a conversion, to start a conversion and to manually halt a conversion in process. Setting the GO/$\overline{\text{DONE}}$ bit starts a conversion. When the conversion is complete, the ADC module clears the GO/$\overline{\text{DONE}}$ bit and sets the ADIF bit in the INTCON register.

A conversion can be terminated by manually clearing the GO/$\overline{\text{DONE}}$ bit while a conversion is in process. Manual termination of a conversion may result in a partially converted result in ADRES.

The GO/$\overline{\text{DONE}}$ bit is cleared when the device enters Sleep, stopping the current conversion. The ADC does not have a dedicated oscillator, it runs off of the instruction clock. Therefore, no conversion can occur in Sleep.

The GO/$\overline{\text{DONE}}$ bit cannot be set when ADON is clear.

9.1.5 SLEEP

This ADC does not have a dedicated ADC clock, and therefore, no conversion in Sleep is possible. If a conversion is underway and a Sleep command is executed, the GO/$\overline{\text{DONE}}$ and ADON bit will be cleared. This will stop any conversion in process and power-down the ADC module to conserve power. Due to the nature of the conversion process, the ADRES may contain a partial conversion. At least one bit must have been converted prior to Sleep to have partial conversion data in ADRES. The ADCS and CHS bits are reset to their default condition; ANS<7:0> = 1s and CHS<3:0> = 1s.

- For accurate conversions, TAD must meet the following:
- 500 ns < TAD < 50 µs
- TAD = 1/(FOSC/divisor)

Shaded areas indicate TAD out of range for accurate conversions. If analog input is desired at these frequencies, use INTOSC/8 for the ADC clock source.

TABLE 9-2: TAD FOR ADCS SETTINGS WITH VARIOUS OSCILLATORS

Source	ADCS <1:0>	Divisor	20 MHz	16 MHz	8 MHz	4 MHz	1 MHz	500 kHz	350 kHz	200 kHz	100 kHz	32 kHz
INTOSC	11	4	—	—	.5 µs	1 µs	—	—	—	—	—	—
FOSC	10	4	.2 µs	.25 µs	.5 µs	1 µs	4 µs	8 µs	11 µs	20 µs	40 µs	125 µs
FOSC	01	8	.4 µs	.5 µs	1 µs	2 µs	8 µs	16 µs	23 µs	40 µs	80 µs	250 µs
FOSC	00	16	.8 µs	1 µs	2 µs	4 µs	16 µs	32 µs	46 µs	80 µs	160 µs	500 µs

TABLE 9-3: EFFECTS OF SLEEP ON ADCON0

	ANS<7:0>	ADCS1	ADCS0	CHS<3:0>	GO/$\overline{\text{DONE}}$	ADON
Entering Sleep	Unchanged	1	1	1	0	0
Wake or Reset	1	1	1	1	0	0

9.1.6 ANALOG CONVERSION RESULT REGISTER

The ADRES register contains the results of the last conversion. These results are present during the sampling period of the next analog conversion process. After the sampling period is over, ADRES is cleared (= 0). A 'leading one' is then right shifted into the ADRES to serve as an internal conversion complete bit. As each bit weight, starting with the MSB, is converted, the leading one is shifted right and the converted bit is stuffed into ADRES. After a total of nine right shifts of the 'leading one' have taken place, the conversion is complete; the 'leading one' has been shifted out and the GO/$\overline{\text{DONE}}$ bit is cleared.

If the GO/$\overline{\text{DONE}}$ bit is cleared in software during a conversion, the conversion stops and the ADIF bit will not be set to a '1'. The data in ADRES is the partial conversion result. This data is valid for the bit weights that have been converted. The position of the 'leading one' determines the number of bits that have been converted. The bits that were not converted before the GO/$\overline{\text{DONE}}$ was cleared are unrecoverable.

REGISTER 9-1: ADCON0: A/D CONTROL REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0
ADCS1	ADCS0	CHS3	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON
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 7-6 **ADCS<1:0>:** ADC Conversion Clock Select bits
- 00 = FOSC/16
- 01 = FOSC/8
- 10 = FOSC/4
- 11 = INTOSC/4

bit 5-2 **CHS<3:0>:** ADC Channel Select Bits(1)
- 0000 = Channel 0 (RA0/AN0)
- 0001 = Channel 1 (RA1/AN1)
- 0010 = Channel 2 (RA2/AN2)
- 0011 = Channel 3 (RA4/AN3)
- 0100 = Channel 4 (RC0/AN4)
- 0101 = Channel 5 (RC1/AN5)
- 0110 = Channel 6 (RC2/AN6)
- 0111 = Channel 7 (RC3/AN7)
- 1xxx = Reserved
- 1111 = 0.6V reference from INTOSC

bit 1 **GO/$\overline{\text{DONE}}$:** ADC Conversion Status Bit(2)
- 1 = ADC conversion in progress. Setting this bit starts an ADC conversion cycle. This bit is automatically cleared by hardware when the ADC is done converting.
- 0 = ADC conversion completed/not in progress. Manually clearing this bit while a conversion is in process terminates the current conversion.

bit 0 **ADON:** ADC Enable bit
- 1 = ADC module is operating
- 0 = ADC module is shut-off and consumes no power

Note 1: CHS<3:0> bits default to 1 after any Reset.
2: If the ADON bit is clear, the GO/$\overline{\text{DONE}}$ bit cannot be set.

REGISTER 9-2: ADRES: A/D CONVERSION RESULTS REGISTER

R/W-X	R/W-X	R/W-X	R/W-X	R/W-X	R/W-X	R/W-X	R/W-X
ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0
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 7-0 **ADRES<7:0>**: ADC Result Register bits

EXAMPLE 9-1: PERFORMING AN ANALOG-TO-DIGITAL CONVERSION

```
;Sample code operates out of BANK0
        MOVLW 0xF1     ;configure A/D
        MOVWF ADCON0
        BSF ADCON0, 1  ;start conversion
loop0   BTFSC ADCON0, 1;wait for 'DONE'
        GOTO loop0
        MOVF ADRES, W  ;read result
        MOVWF result0  ;save result

        BSF ADCON0, 2  ;setup for read of
                       ;channel 1
        BSF ADCON0, 1  ;start conversion
loop1   BTFSC ADCON0, 1;wait for 'DONE'
        GOTO loop1
        MOVF ADRES, W  ;read result
        MOVWF result1  ;save result

        BSF ADCON0, 3  ;setup for read of
        BCF ADCON0, 2  ;channel 2
        BSF ADCON0, 1  ;start conversion
loop2   BTFSC ADCON0, 1;wait for 'DONE'
        GOTO loop2
        MOVF ADRES, W  ;read result
        MOVWF result2  ;save result
```

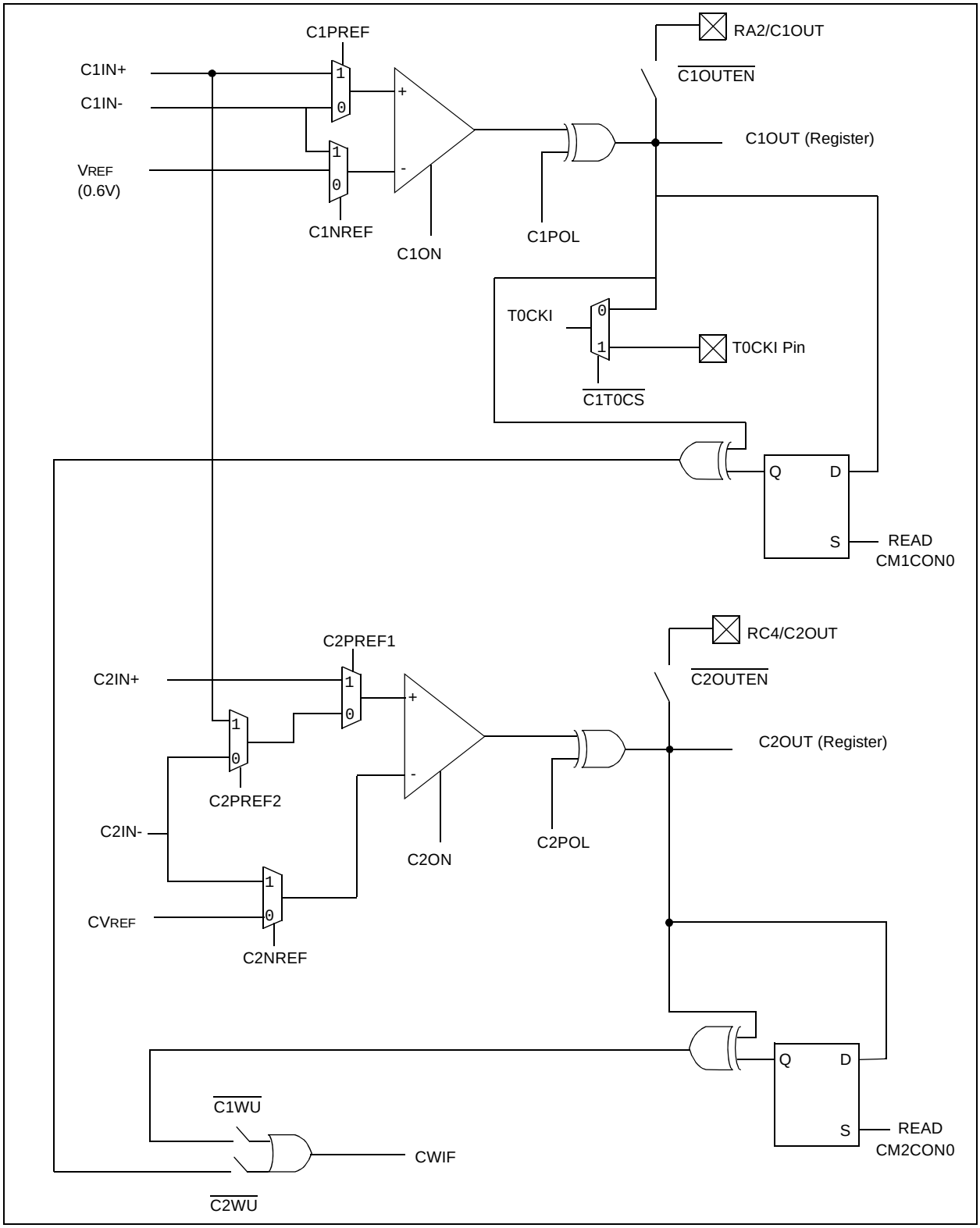
EXAMPLE 9-2: CHANNEL SELECTION CHANGE DURING CONVERSION

```
        MOVLW 0xF1     ;configure A/D
        MOVWF ADCON0
        BSF ADCON0, 1  ;start conversion
        BSF ADCON0, 2  ;setup for read of
                       ;channel 1
loop0   BTFSC ADCON0, 1;wait for 'DONE'
        GOTO loop0
        MOVF ADRES, W  ;read result
        MOVWF result0  ;save result

        BSF ADCON0, 1  ;start conversion
        BSF ADCON0, 3  ;setup for read of
        BCF ADCON0, 2  ;channel 2
loop1   BTFSC ADCON0, 1;wait for 'DONE'
        GOTO loop1
        MOVF ADRES, W  ;read result
        MOVWF result1  ;save result

        BSF ADCON0, 1  ;start conversion
loop2   BTFSC ADCON0, 1;wait for 'DONE'
        GOTO loop2
        MOVF ADRES, W  ;read result
        MOVWF result2  ;save result
        CLRF ADCON0    ;optional: returns
        ;pins to Digital mode and turns off
        ;the ADC module
```


10.0 COMPARATOR(S)

This device contains two comparators and a comparator voltage reference.

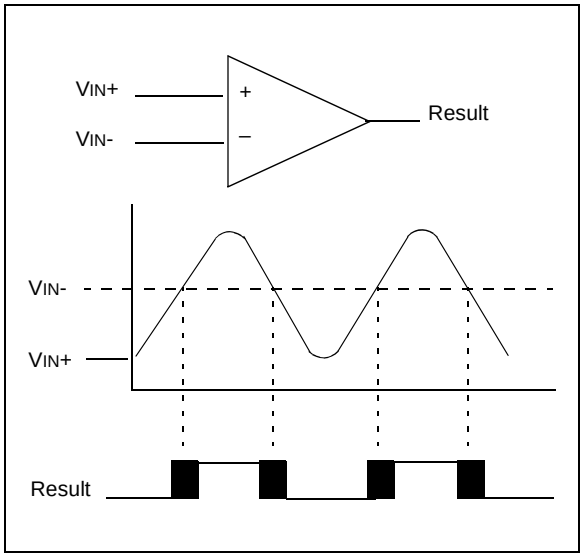
FIGURE 10-1: COMPARATORS BLOCK DIAGRAM



10.1 Comparator Operation

A single comparator is shown in Figure 10-2 along with the relationship between the analog input levels and the digital output. When the analog input at VIN+ is less than the analog input VIN-, the output of the comparator is a digital low level. The shaded area of the output of the comparator in Figure 10-2 represent the uncertainty due to input offsets and response time. See Table 15-2 for Common Mode Voltage.

FIGURE 10-2: SINGLE COMPARATOR



10.2 Comparator Reference

An internal reference signal may be used depending on the comparator operating mode. The analog signal that is present at VIN- is compared to the signal at VIN+, and the digital output of the comparator is adjusted accordingly (Figure 10-2). Please see **Section 11.0 "Comparator Voltage Reference Module"** for internal reference specifications.

10.3 Comparator Response Time

Response time is the minimum time after selecting a new reference voltage or input source before the comparator output is to have a valid level. If the comparator inputs are changed, a delay must be used to allow the comparator to settle to its new state. Please see Table 15-6 for comparator response time specifications.

10.4 Comparator Output

The comparator output is read through the CxOUT bit in the CM1CON0 or CM2CON0 register. This bit is read-only. The comparator output may also be used externally, see **Section 10.1 "Comparator Operation"**.

Note:	Analog levels on any pin that is defined as a digital input may cause the input buffer to consume more current than is specified.

10.5 Comparator Wake-up Flag

The Comparator Wake-up Flag bit, CWIF, in the INTCON0 register, is set whenever all of the following conditions are met:

- $\overline{\text{C1WU}}$ = 0 (CM1CON0<0>) or $\overline{\text{C2WU}}$ = 0 (CM2CON0<0>)
- CM1CON0 or CM2CON0 has been read to latch the last known state of the C1OUT and C2OUT bit (`MOVF CM1CON0, W`)
- The output of a comparator has changed state

The wake-up flag may be cleared in software or by another device Reset.

10.6 Comparator Operation During Sleep

When the comparator is enabled it is active. To minimize power consumption while in Sleep mode, turn off the comparator before entering Sleep.

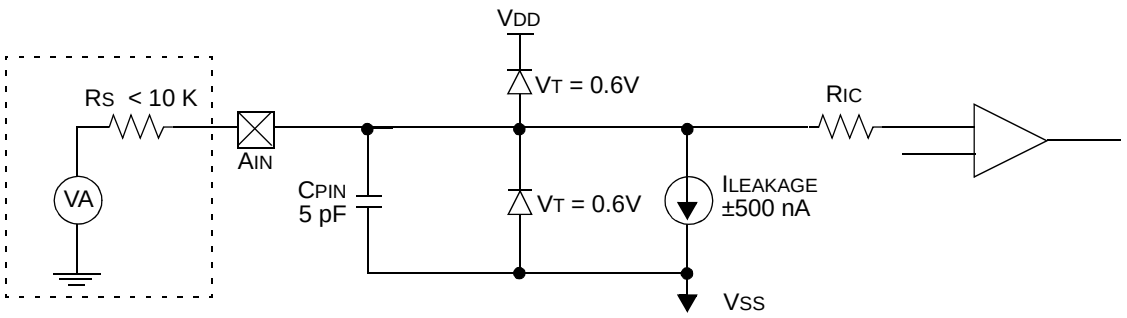
10.7 Effects of Reset

A Power-on Reset (POR) forces the CMxCON0 register to its Reset state. This forces the Comparator input pins to analog Reset mode. Device current is minimized when analog inputs are present at Reset time.

10.8 Analog Input Connection Considerations

A simplified circuit for an analog input is shown in Figure 10-3. Since the analog pins are connected to a digital output, they have reverse biased diodes to VDD and VSS. The analog input, therefore, must be between VSS and VDD. 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.

FIGURE 10-3: ANALOG INPUT MODE



Legend:

CPIN	=	Input Capacitance
VT	=	Threshold Voltage
ILEAKAGE	=	Leakage Current at the Pin
RIC	=	Interconnect Resistance
RS	=	Source Impedance
VA	=	Analog Voltage

10.9 Register Definitions — Comparator Control

REGISTER 10-1: CM1CON0: COMPARATOR C1 CONTROL REGISTER

R-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
C1OUT	$\overline{\text{C1OUTEN}}$	C1POL	$\overline{\text{C1T0CS}}$	C1ON	C1NREF	C1PREF	$\overline{\text{C1WU}}$
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 **C1OUT:** Comparator Output bit
1 = $V_{IN+} > V_{IN-}$
0 = $V_{IN+} < V_{IN-}$

bit 6 **$\overline{\text{C1OUTEN}}$:** Comparator Output Enable bit(1), (2)
1 = Output of comparator is NOT placed on the C1OUT pin
0 = Output of comparator is placed in the C1OUT pin

bit 5 **C1POL:** Comparator Output Polarity bit(2)
1 = Output of comparator is not inverted
0 = Output of comparator is inverted

bit 4 **$\overline{\text{C1T0CS}}$:** Comparator TMR0 Clock Source bit(2)
1 = TMR0 clock source selected by T0CS control bit
0 = Comparator output used as TMR0 clock source

bit 3 **C1ON:** Comparator Enable bit
1 = Comparator is on
0 = Comparator is off

bit 2 **C1NREF:** Comparator Negative Reference Select bit(2)
1 = C1IN- pin
0 = 0.6V V_{REF}

bit 1 **C1PREF:** Comparator Positive Reference Select bit(2)
1 = C1IN+ pin
0 = C1IN- pin

bit 0 **$\overline{\text{C1WU}}$:** Comparator Wake-up On Change Enable bit(2)
1 = Wake-up On Comparator Change is disabled
0 = Wake-up On Comparator Change is enabled

Note 1: Overrides TRIS control of RA2.

2: When comparator is turned on, these control bits assert themselves. Otherwise, the other registers have precedence.

3: The $\overline{\text{C1WU}}$ bit must be set to enable the CWIF function. See the INTCON0 register (Register 8-2) for more information.

REGISTER 10-2: CM2CON0: COMPARATOR C2 CONTROL REGISTER

R-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
C2OUT	$\overline{\text{C2OUTEN}}$	C2POL	C2PREF2	C2ON	C2NREF	C2PREF1	$\overline{\text{C2WU}}$
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 **C2OUT:** Comparator Output bit
1 = VIN+ > VIN-
0 = VIN+ < VIN-

bit 6 **$\overline{\text{C2OUTEN}}$:** Comparator Output Enable bit(1), (2)
1 = Output of comparator is NOT placed on the C2OUT pin
0 = Output of comparator is placed in the C2OUT pin

bit 5 **C2POL:** Comparator Output Polarity bit(2)
1 = Output of comparator not inverted
0 = Output of comparator inverted

bit 4 **C2PREF2:** Comparator Positive Reference Select bit(2)
1 = C1IN+ pin
0 = C2IN- pin

bit 3 **C2ON:** Comparator Enable bit
1 = Comparator is on
0 = Comparator is off

bit 2 **C2NREF:** Comparator Negative Reference Select bit(2)
1 = C2IN- pin
0 = CVREF

bit 1 **C2PREF1:** Comparator Positive Reference Select bit(2)
1 = C2IN+ pin
0 = C2PREF2 controls analog input selection

bit 0 **$\overline{\text{C2WU}}$:** Comparator Wake-up on Change Enable bit(2)
1 = Wake-up on Comparator change is disabled
0 = Wake-up on Comparator change is enabled.

Note 1: Overrides TRIS control of RC4.

2: When comparator is turned on, these control bits assert themselves. Otherwise, the other registers have precedence.

3: The $\overline{\text{C2WU}}$ bit must be set to enable the CWIF function. See the INTCON0 register (Register 8-2) for more information.

TABLE 10-1: REGISTERS ASSOCIATED WITH COMPARATOR MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
STATUS	—	—	PA0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	19
CM1CON0	C1OUT	$\overline{\text{C1OUTEN}}$	C1POL	$\overline{\text{C1T0CS}}$	C1ON	C1NREF	C1PREF	$\overline{\text{C1WU}}$	66
CM2CON0	C2OUT	$\overline{\text{C2OUTEN}}$	C2POL	C2PREF2	C2ON	C2NREF	C2PREF1	$\overline{\text{C2WU}}$	67
TRIS	I/O Control Register (TRISA, TRISB, TRISC)								—

Legend: x = Unknown, u = Unchanged, – = Unimplemented, read as '0', q = Depends on condition.

NOTES:

11.0 COMPARATOR VOLTAGE REFERENCE MODULE

The Comparator Voltage Reference module also allows the selection of an internally generated voltage reference for one of the C2 comparator inputs. The VRCON register (Register 11-1) controls the voltage reference module shown in Figure 11-1.

11.1 Configuring The Voltage Reference

The voltage reference can output 32 voltage levels; 16 in a high range and 16 in a low range.

Equation 11-1 determines the output voltages:

EQUATION 11-1:

VRR = 1 (low range):

$$CV_{REF} = (VR<3:0>/24) \times V_{DD}$$

VRR = 0 (high range):

$$CV_{REF} = (V_{DD}/4) + (VR<3:0> \times V_{DD}/32)$$

11.2 Voltage Reference Accuracy/Error

The full range of VSS to VDD cannot be realized due to construction of the module. The transistors on the top and bottom of the resistor ladder network (Figure 11-1) keep CVREF from approaching VSS or VDD. The exception is when the module is disabled by clearing the VREN bit of the VRCON register. When disabled, the reference voltage is VSS when VR<3:0> is '0000' and the VRR bit of the VRCON register is set. This allows the comparator to detect a zero-crossing and not consume the CVREF module current.

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

REGISTER 11-1: VRCON: VOLTAGE REFERENCE CONTROL REGISTER

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
VREN	VROE	VRR	—	VR3	VR2	VR1	VR0
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 **VREN:** CVREF Enable bit
1 = CVREF is powered on
0 = CVREF is powered down, no current is drawn

bit 6 **VROE:** CVREF Output Enable bit(1)
1 = CVREF output is enabled
0 = CVREF output is disabled

bit 5 **VRR:** CVREF Range Selection bit
1 = Low range
0 = High range

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

bit 3-0 **VR<3:0>** CVREF Value Selection bits
When VRR = 1: CVREF= (VR<3:0>/24)*VDD
When VRR = 0: CVREF= VDD/4+(VR<3:0>/32)*VDD

Note 1: When this bit is set, the TRIS for the CVREF pin is overridden and the analog voltage is placed on the CVREF pin.

FIGURE 11-1: COMPARATOR VOLTAGE REFERENCE BLOCK DIAGRAM

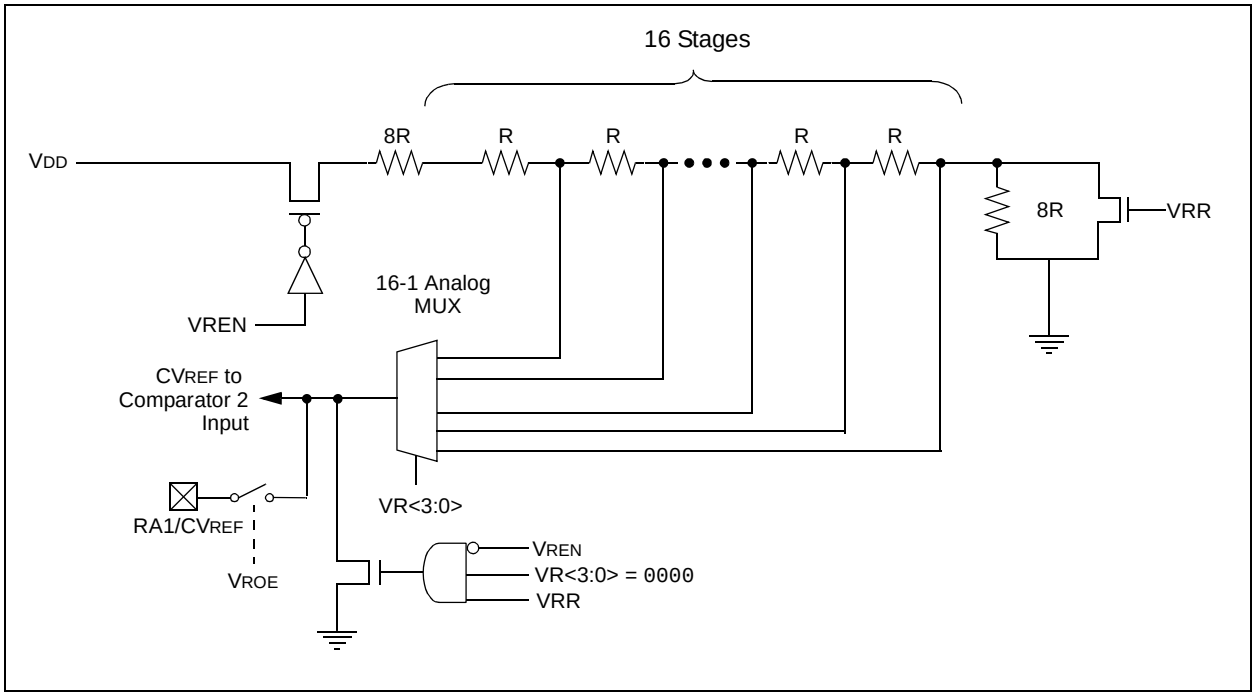


TABLE 11-1: REGISTERS ASSOCIATED WITH COMPARATOR VOLTAGE REFERENCE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
VRCON	VREN	VROE	VRR	—	VR3	VR2	VR1	VR0	69
CM1CON0	C1OUT	$\overline{\text{C1OUTEN}}$	C1POL	$\overline{\text{C1T0CS}}$	C1ON	C1NREF	C1PREF	$\overline{\text{C1WU}}$	66
CM2CON0	C2OUT	$\overline{\text{C2OUTEN}}$	C2POL	$\overline{\text{C2PREF2}}$	C2ON	C2NREF	C2PREF1	$\overline{\text{C2WU}}$	67

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

12.0 OPERATIONAL AMPLIFIER (OPA) MODULE

The OPA module has the following features:

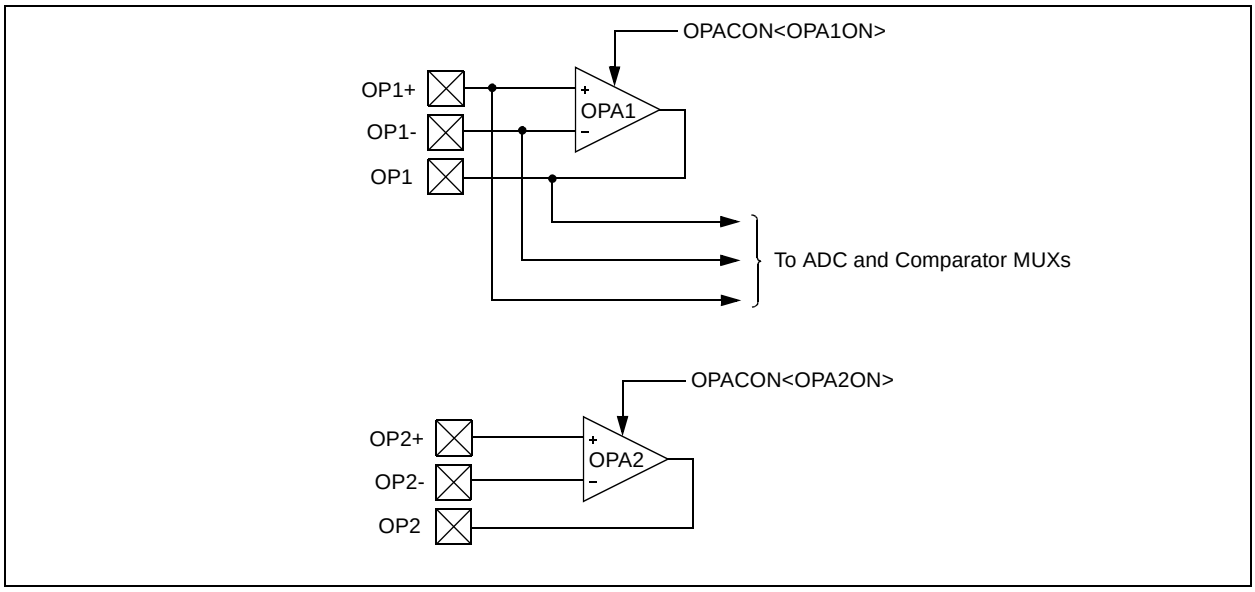
- Two independent Operational Amplifiers
- External connections to all ports
- 3 MHz Gain Bandwidth Product (GBWP)

12.1 OPACON Register

The OPA module is enabled by setting the OPAxON bit of the OPACON Register. When enabled, OPAxON forces the output driver of OP1 for OPA1, and OP2 for OPA2, into tri-state to prevent contention between the driver and the OPA output.

Note:	When OPA1 or OPA2 is enabled, the OP1 pin or OP2 pin, respectively, is driven by the op amp output, not by the PORTC driver. Refer to Table 15-4 for the electrical specifications for the op amp output drive capability.

FIGURE 12-1: OPA MODULE BLOCK DIAGRAM



REGISTER 12-1: OPACON: OP AMP CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	OPA2ON	OPA1ON
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 7-2 **Unimplemented:** Read as '0'

bit 1 **OPA2ON:** Op Amp Enable bit
1 = Op amp 2 is enabled
0 = Op amp 2 is disabled

bit 0 **OPA1ON:** Op Amp Enable bit
1 = Op amp 1 is enabled
0 = Op amp 1 is disabled

12.2 Effects of a Reset

A device Reset forces all registers to their Reset state. This disables both op amps.

12.3 OPA Module Performance

Common AC and DC performance specifications for the OPA module:

- Common Mode Voltage Range
- Leakage Current
- Input Offset Voltage
- Open Loop Gain
- Gain Bandwidth Product (GBWP)

Common mode voltage range is the specified voltage range for the OPA+ and OPA- inputs, for which the OPA module will perform to within its specifications. The OPA module is designed to operate with input voltages between 0 and V_{DD}-1.5V. Behavior for common mode voltages greater than V_{DD}-1.5V, or below 0V, are beyond the normal operating range.

Leakage current is a measure of the small source or sink currents on the OPA+ and OPA- inputs. To minimize the effect of leakage currents, the effective impedances connected to the OPA+ and OPA- inputs should be kept as small as possible and equal.

Input offset voltage is a measure of the voltage difference between the OPA+ and OPA- inputs in a closed loop circuit with the OPA in its linear region. The offset voltage will appear as a DC offset in the output equal to the input offset voltage, multiplied by the gain of the circuit. The input offset voltage is also affected by the common mode voltage.

Open loop gain is the ratio of the output voltage to the differential input voltage, (OPA+) - (OPA-). The gain is greatest at DC and falls off with frequency.

Gain Bandwidth Product or GBWP is the frequency at which the open loop gain falls off to 0 dB.

12.4 Effects of Sleep

When enabled, the op amps continue to operate and consume current while the processor is in Sleep mode.

TABLE 12-1: REGISTERS ASSOCIATED WITH THE OPA MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	32
OPACON	—	—	—	—	—	—	OPA2ON	OPA1ON	72
TRIS	I/O Control Registers (TRISA, TRISB, TRISC)								—

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

13.0 INSTRUCTION SET SUMMARY

The PIC16 instruction set is highly orthogonal and is comprised of three basic categories.

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal and control** operations

Each PIC16 instruction is a 12-bit word divided into an **opcode**, which specifies the instruction type, and one or more **operands** which further specify the operation of the instruction. The formats for each of the categories is presented in Figure 13-1, while the various opcode fields are summarized in Table 13-1.

For **byte-oriented** instructions, 'f' represents a file register designator and 'd' represents a destination designator. The file register designator specifies which file register is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed in the file register specified in the instruction.

For **bit-oriented** instructions, 'b' represents a bit field designator which selects the number of the bit affected by the operation, while 'f' represents the number of the file in which the bit is located.

For **literal and control** operations, 'k' represents an 8 or 9-bit constant or literal value.

TABLE 13-1: OPCODE FIELD DESCRIPTIONS

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1) The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0 (store result in W) d = 1 (store result in file register 'f') Default is d = 1
label	Label name
TOS	Top-of-Stack
PC	Program Counter
WDT	Watchdog Timer counter
$\overline{TO}$	Time-out bit
$\overline{PD}$	Power-down bit
dest	Destination, either the W register or the specified register file location
[ ]	Options
( )	Contents
Æ	Assigned to
< >	Register bit field
Œ	In the set of
italics	User defined term (font is courier)

All instructions are executed within a single instruction cycle, unless a conditional test is true or the program counter is changed as a result of an instruction. In this case, the execution takes two instruction cycles. One instruction cycle consists of four oscillator periods. Thus, for an oscillator frequency of 4 MHz, the normal instruction execution time is 1 μs. If a conditional test is true or the program counter is changed as a result of an instruction, the instruction execution time is 2 μs.

Figure 13-1 shows the three general formats that the instructions can have. All examples in the figure use the following format to represent a hexadecimal number:

0xhhh

where 'h' signifies a hexadecimal digit.

FIGURE 13-1: GENERAL FORMAT FOR INSTRUCTIONS

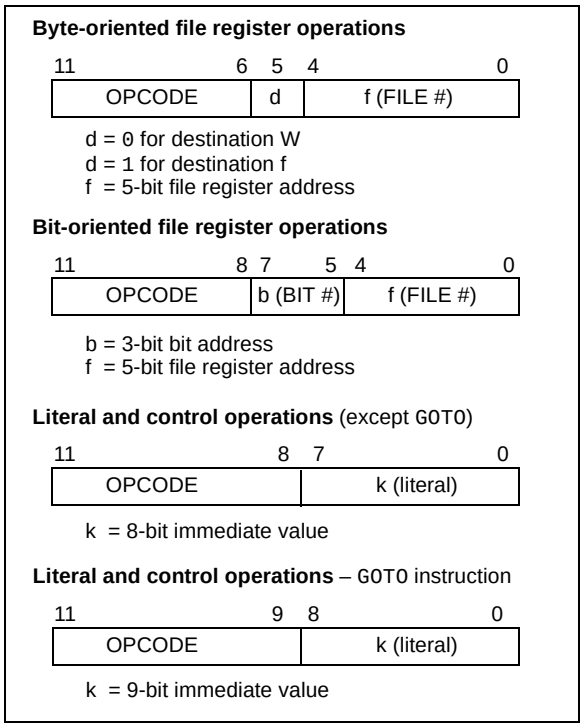


TABLE 13-2: INSTRUCTION SET SUMMARY

Mnemonic, Operands		Description	Cycles	12-Bit Opcode			Status Affected	Notes
				MSb		LSb		
ADDWF	f, d	Add W and f	1	0001	11df	ffff	C, DC, Z	**1, 2, 4**
ANDWF	f, d	AND W with f	1	0001	01df	ffff	Z	**2, 4**
CLRF	f	Clear f	1	0000	011f	ffff	Z	**4**
CLRW	—	Clear W	1	0000	0100	0000	Z	
COMF	f, d	Complement f	1	0010	01df	ffff	Z	
DECF	f, d	Decrement f	1	0000	11df	ffff	Z	**2, 4**
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	0010	11df	ffff	None	**2, 4**
INCF	f, d	Increment f	1	0010	10df	ffff	Z	**2, 4**
INCFSZ	f, d	Increment f, Skip if 0	1(2)	0011	11df	ffff	None	**2, 4**
IORWF	f, d	Inclusive OR W with f	1	0001	00df	ffff	Z	**2, 4**
MOVF	f, d	Move f	1	0010	00df	ffff	Z	**2, 4**
MOVWF	f	Move W to f	1	0000	001f	ffff	None	**1, 4**
NOP	—	No Operation	1	0000	0000	0000	None	
RLF	f, d	Rotate left f through Carry	1	0011	01df	ffff	C	**2, 4**
RRF	f, d	Rotate right f through Carry	1	0011	00df	ffff	C	**2, 4**
SUBWF	f, d	Subtract W from f	1	0000	10df	ffff	C, DC, Z	**1, 2, 4**
SWAPF	f, d	Swap f	1	0011	10df	ffff	None	**2, 4**
XORWF	f, d	Exclusive OR W with f	1	0001	10df	ffff	Z	**2, 4**
BIT-ORIENTED FILE REGISTER OPERATIONS								
BCF	f, b	Bit Clear f	1	0100	bbbf	ffff	None	**2, 4**
BSF	f, b	Bit Set f	1	0101	bbbf	ffff	None	**2, 4**
BTFSC	f, b	Bit Test f, Skip if Clear	1(2)	0110	bbbf	ffff	None	
BTFSS	f, b	Bit Test f, Skip if Set	1(2)	0111	bbbf	ffff	None	
LITERAL AND CONTROL OPERATIONS								
ANDLW	k	AND literal with W	1	1110	kkkk	kkkk	Z	
CALL	k	Call Subroutine	2	1001	kkkk	kkkk	None	**1**
CLRWDT	—	Clear Watchdog Timer	1	0000	0000	0100	$\overline{TO}$, $\overline{PD}$	
GOTO	k	Unconditional branch	2	101k	kkkk	kkkk	None	
IORLW	k	Inclusive OR literal with W	1	1101	kkkk	kkkk	Z	
MOVLB	k	Move Literal to BSR Register	1	0000	0001	0kkk	None	
MOVLW	k	Move literal to W	1	1100	kkkk	kkkk	None	
OPTION	—	Load OPTION register	1	0000	0000	0010	None	
RETFIE	—	Return from Interrupt	2	0000	0001	1111	None	
RETLW	k	Return, place literal in W	2	1000	kkkk	kkkk	None	**3**
RETURN	—	Return, maintain W	2	0000	0001	1110	None	
SLEEP	—	Go into Standby mode	1	0000	0000	0011	$\overline{TO}$, $\overline{PD}$	
TRIS	f	Load TRIS register	1	0000	0000	0fff	None	
XORLW	k	Exclusive OR literal to W	1	1111	kkkk	kkkk	Z	

Note 1: The 9th bit of the program counter will be forced to a '0' by any instruction that writes to the PC except for GOTO. See **Section 4.6 "Program Counter"**.

2: When an I/O register is modified as a function of itself (e.g. MOVF PORTB, 1), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

3: The instruction TRIS f, where f = 6, causes the contents of the W register to be written to the tri-state latches of PORTA. A '1' forces the pin to a high-impedance state and disables the output buffers.

4: If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared (if assigned to TMR0).

ADDWF	Add W and f
Syntax:	[*label*] ADDWF f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	(W) + (f) → (dest)
Status Affected:	C, DC, Z
Description:	Add the contents of the W register and 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'.

ANDLW	AND literal with W
Syntax:	[*label*] ANDLW k
Operands:	$0 \le k \le 255$
Operation:	(W).AND. (k) → (W)
Status Affected:	Z
Description:	The contents of the W register are AND'ed with the eight-bit literal 'k'. The result is placed in the W register.

ANDWF	AND W with f
Syntax:	[*label*] ANDWF f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	(W) .AND. (f) → (dest)
Status Affected:	Z
Description:	The contents of the W register are AND'ed with 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'.

BCF	Bit Clear f
Syntax:	[*label*] BCF f,b
Operands:	$0 \le f \le 31$ $0 \le b \le 7$
Operation:	0 → (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is cleared.

BSF	Bit Set f
Syntax:	[*label*] BSF f,b
Operands:	$0 \le f \le 31$ $0 \le b \le 7$
Operation:	1 → (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is set.

BTFSC	Bit Test f, Skip if Clear
Syntax:	[*label*] BTFSC f,b
Operands:	$0 \le f \le 31$ $0 \le b \le 7$
Operation:	skip if (f<b>) = 0
Status Affected:	None
Description:	If bit 'b' in register 'f' is '0', then the next instruction is skipped. If bit 'b' is '0', then the next instruction fetched during the current instruction execution is discarded, and a NOP is executed instead, making this a two-cycle instruction.

BTFSS	Bit Test f, Skip if Set
Syntax:	[*label*] BTFSS f,b
Operands:	$0 \le f \le 31$ $0 \le b < 7$
Operation:	skip if (f<b>) = 1
Status Affected:	None
Description:	If bit 'b' in register 'f' is '1', then the next instruction is skipped. If bit 'b' is '1', then the next instruction fetched during the current instruction execution, is discarded and a NOP is executed instead, making this a two-cycle instruction.

CALL	Subroutine Call
Syntax:	[*label*] CALL k
Operands:	$0 \le k \le 255$
Operation:	(PC) + 1→ Top-of-Stack; k → PC<7:0>; (STATUS<6:5>) → PC<10:9>; 0 → PC<8>
Status Affected:	None
Description:	Subroutine call. First, return address (PC + 1) is PUSHed onto the stack. The eight-bit immediate address is loaded into PC bits <7:0>. The upper bits PC<10:9> are loaded from STATUS<6:5>, PC<8> is cleared. CALL is a two-cycle instruction.

CLRF	Clear f
Syntax:	[*label*] CLRF f
Operands:	$0 \le f \le 31$
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:	The W register is cleared. Zero bit (Z) is set.

CLRWDT	Clear Watchdog Timer
Syntax:	[*label*] CLRWDT
Operands:	None
Operation:	00h → WDT; 0 → WDT prescaler (if assigned); 1 → $\overline{TO}$; 1 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	The CLRWDT instruction resets the WDT. It also resets the prescaler, if the prescaler is assigned to the WDT and not Timer0. Status bits $\overline{TO}$ and $\overline{PD}$ are set.

COMF	Complement f
Syntax:	[*label*] COMF f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	$(\bar{f}) \rightarrow$ (dest)
Status Affected:	Z
Description:	The contents of register 'f' are complemented. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

DECF	Decrement f
Syntax:	[*label*] DECF f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	(f) – 1 → (dest)
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’.

DECFSZ	Decrement f, Skip if 0
Syntax:	[*label*] DECFSZ f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	(f) – 1 → d; 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 ‘0’, the next instruction, which is already fetched, is discarded and a NOP is executed instead making it a two-cycle instruction.

GOTO	Unconditional Branch
Syntax:	[*label*] GOTO k
Operands:	$0 \le k \le 511$
Operation:	k → PC<8:0>; STATUS<6:5> → PC<10:9>
Status Affected:	None
Description:	GOTO is an unconditional branch. The 9-bit immediate value is loaded into PC bits <8:0>. The upper bits of PC are loaded from STATUS<6:5>. GOTO is a two-cycle instruction.

INCF	Increment f
Syntax:	[*label*] INCF f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	(f) + 1 → (dest)
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:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	(f) + 1 → (dest), 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 ‘0’, then the next instruction, which is already fetched, is discarded and 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:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	(W).OR. (f) → (dest)
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'.

MOVF	Move f
Syntax:	[*label*] MOVF f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	(f) → (dest)
Status Affected:	Z
Description:	The contents of register 'f' are moved to destination 'd'. If 'd' is '0', destination is the W register. If 'd' is '1', the destination is file register 'f'. 'd' = 1 is useful as a test of a file register, since status flag Z is affected.

MOVLB	Move Literal to BSR
Syntax:	[*label*] MOVLB k
Operands:	$0 \le k \le 7$
Operation:	k → (BSR)
Status Affected:	None
Description:	The three-bit literal 'k' is loaded into the BSR register.

MOVLW	Move Literal to W
Syntax:	[*label*] MOVLW k
Operands:	$0 \le k \le 255$
Operation:	k → (W)
Status Affected:	None
Description:	The eight-bit literal 'k' is loaded into the W register. The "don't cares" will assembled as '0's.

MOVWF	Move W to f
Syntax:	[*label*] MOVWF f
Operands:	$0 \le f \le 31$
Operation:	(W) → (f)
Status Affected:	None
Description:	Move data from the W register to register 'f'.

NOP	No Operation
Syntax:	[*label*] NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Description:	No operation.

OPTION	Load OPTION Register
Syntax:	[*label*] OPTION
Operands:	None
Operation:	(W) → OPTION
Status Affected:	None
Description:	The content of the W register is loaded into the OPTION register.

RETFIE	Return From Interrupt
Syntax:	[*label*] RETFIE
Operands:	None
Operation:	TOS → PC 1 → GIE
Status Affected:	None
Description:	The program counter is loaded from the top of the stack (the return address). GIE bit of INTCON0 is set. This is a two-cycle instruction.

RETLW **Return with Literal in W**

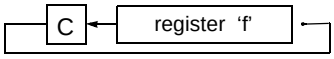
Syntax:	[*label*] RETLW k
Operands:	$0 \le k \le 255$
Operation:	$k \rightarrow (W)$; $TOS \rightarrow PC$
Status Affected:	None
Description:	The W register is loaded with the eight-bit literal ‘k’. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.

RETURN **Return**

Syntax:	[*label*] RETURN
Operands:	None
Operation:	$TOS \rightarrow PC$
Status Affected:	None
Description:	The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.

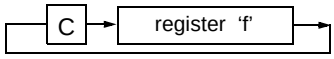
RLF **Rotate Left f through Carry**

Syntax:	[*label*] RLF f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	See description below
Status Affected:	C
Description:	The contents of register ‘f’ are rotated one bit to the left through the Carry flag. If ‘d’ is ‘0’, the result is placed in the W register. If ‘d’ is ‘1’, the result is stored back in register ‘f’.



RRF **Rotate Right f through Carry**

Syntax:	[*label*] RRF f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	See description below
Status Affected:	C
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’.



SLEEP **Enter SLEEP Mode**

Syntax:	[*label*] SLEEP
Operands:	None
Operation:	$00h \rightarrow WDT$; $0 \rightarrow \overline{WDT}$ prescaler; $1 \rightarrow \overline{TO}$; $0 \rightarrow \overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$, RBWUF
Description:	Time-out Status bit ($\overline{TO}$) is set. The Power-down Status bit ($\overline{PD}$) is cleared. RBWUF is unaffected. The WDT and its prescaler are cleared. The processor is put into Sleep mode with the oscillator stopped. See **Section 8.10 “Power-down Mode (Sleep)”** on Sleep for more details.

SUBWF **Subtract W from f**

Syntax:	[*label*] SUBWF f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	$(f) - (W) \rightarrow (dest)$
Status Affected:	C, DC, Z
Description:	Subtract (2’s complement method) the W register from 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’.

SWAPF	Swap Nibbles in f
Syntax:	[*label*] SWAPF f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	(f<3:0>) → (dest<7:4>); (f<7:4>) → (dest<3:0>)
Status Affected:	None
Description:	The upper and lower nibbles of register ‘f’ are exchanged. If ‘d’ is ‘0’, the result is placed in W register. If ‘d’ is ‘1’, the result is placed in register ‘f’.

TRIS	Load TRIS Register
Syntax:	[*label*] TRIS f
Operands:	f = 6
Operation:	(W) → TRIS register f
Status Affected:	None
Description:	TRIS register ‘f’ (f = 6, 7 or 8) is loaded with the contents of the W register

XORLW	Exclusive OR literal with W
Syntax:	[*label*] XORLW k
Operands:	$0 \le k \le 255$
Operation:	(W) .XOR. k → (W)
Status Affected:	Z
Description:	The contents of the W register are XOR’ed with the eight-bit literal ‘k’. The result is placed in the W register.

XORWF	Exclusive OR W with f
Syntax:	[*label*] XORWF f,d
Operands:	$0 \le f \le 31$ $d \in [0,1]$
Operation:	(W) .XOR. (f) → (dest)
Status Affected:	Z
Description:	Exclusive OR the contents of the W register with 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’.

14.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers and dsPIC® digital signal controllers are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB C Compiler for Various Device Families
  - HI-TECH C® for Various Device Families
  - MPASM™ Assembler
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers
  - MPLAB ICD 3
  - PICkit™ 3 Debug Express
- Device Programmers
  - PICkit™ 2 Programmer
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits, and Starter Kits

14.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8/16/32-bit microcontroller market. The MPLAB IDE is a Windows® operating system-based application that contains:

- A single graphical interface to all debugging tools
  - Simulator
  - Programmer (sold separately)
  - In-Circuit Emulator (sold separately)
  - In-Circuit Debugger (sold separately)
- A full-featured editor with color-coded context
- A multiple project manager
- Customizable data windows with direct edit of contents
- High-level source code debugging
- Mouse over variable inspection
- Drag and drop variables from source to watch windows
- Extensive on-line help
- Integration of select third party tools, such as IAR C Compilers

The MPLAB IDE allows you to:

- Edit your source files (either C or assembly)
- One-touch compile or assemble, and download to emulator and simulator tools (automatically updates all project information)
- Debug using:
  - Source files (C or assembly)
  - Mixed C and assembly
  - Machine code

MPLAB IDE supports multiple debugging tools in a single development paradigm, from the cost-effective simulators, through low-cost in-circuit debuggers, to full-featured emulators. This eliminates the learning curve when upgrading to tools with increased flexibility and power.

14.2 MPLAB C Compilers for Various Device Families

The MPLAB C Compiler code development systems are complete ANSI C compilers for Microchip's PIC18, PIC24 and PIC32 families of microcontrollers and the dsPIC30 and dsPIC33 families of digital signal controllers. These compilers provide powerful integration capabilities, superior code optimization and ease of use.

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

14.3 HI-TECH C for Various Device Families

The HI-TECH C Compiler code development systems are complete ANSI C compilers for Microchip's PIC family of microcontrollers and the dsPIC family of digital signal controllers. These compilers provide powerful integration capabilities, omniscient code generation and ease of use.

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

The compilers include a macro assembler, linker, preprocessor, and one-step driver, and can run on multiple platforms.

14.4 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 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

14.5 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/library 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

14.6 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC devices. MPLAB 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 device instruction set
- Support for fixed-point and floating-point data
- Command line interface
- Rich directive set
- Flexible macro language
- MPLAB IDE compatibility

14.7 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.

14.8 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.

14.9 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.

14.10 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.

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

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

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

14.12 MPLAB PM3 Device Programmer

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

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

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

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

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

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

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

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

15.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings(†)

Ambient temperature under bias ..... -40°C to +125°C
Storage temperature ..... -65°C to +150°C
Voltage on V_{DD} with respect to V_{SS} ..... 0 to +6.5V
Voltage on $\overline{MCLR}$ with respect to V_{SS} ..... 0 to +13.5V
Voltage on all other pins with respect to V_{SS} ..... -0.3V to (V_{DD} + 0.3V)
Total power dissipation(1) ..... 700 mW
Max. current out of V_{SS} pin ..... 200 mA
Max. current into V_{DD} pin ..... 150 mA
Input clamp current, I_{IK} (V_I < 0 or V_I > V_{DD}) ..... ±20 mA
Output clamp current, I_{OK} (V_O < 0 or V_O > V_{DD}) ..... ±20 mA
Max. output current sunk by any I/O pin ..... 25 mA
Max. output current sourced by any I/O pin ..... 25 mA
Max. output current sourced by I/O port ..... 75 mA
Max. output current sunk by I/O port ..... 75 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.

FIGURE 15-1: PIC16F527 VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C

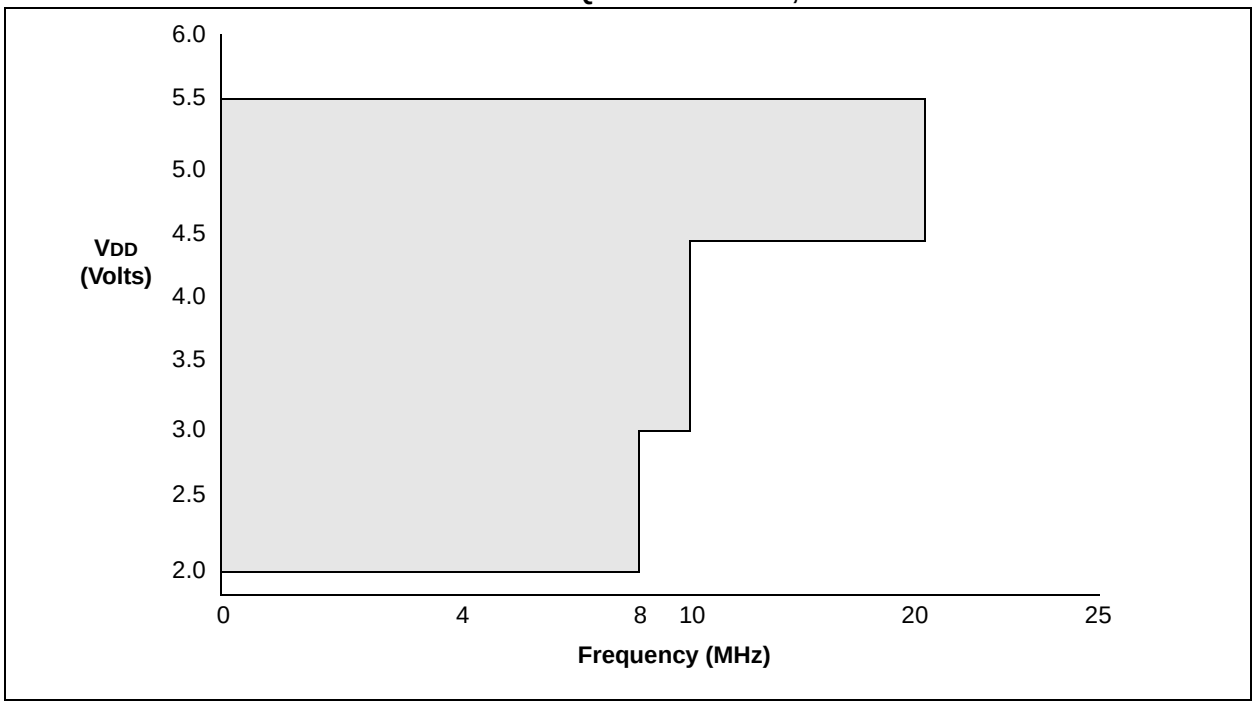
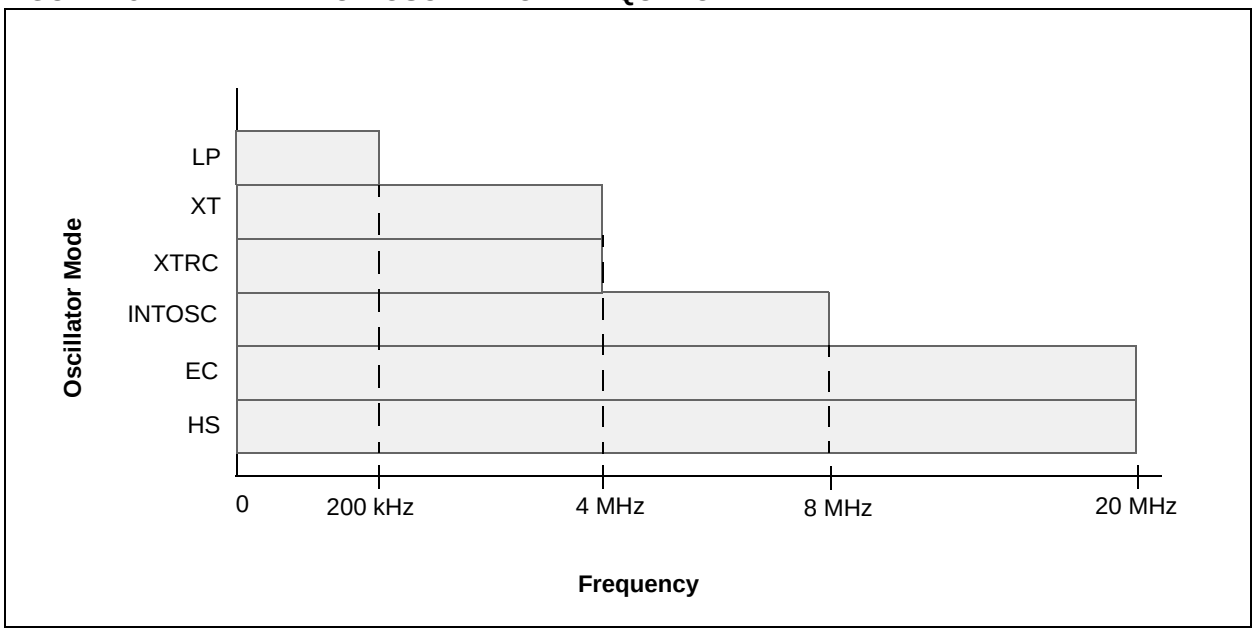


FIGURE 15-2: MAXIMUM OSCILLATOR FREQUENCY TABLE



15.1 DC Characteristics: PIC16F527 (Industrial)

DC Characteristics			Standard Operating Conditions (unless otherwise specified) Operating Temperature -40°C ≤ TA ≤ +85°C (industrial)				
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ.(1)**	**Max.**	**Units**	**Conditions**
D001	VDD	**Supply Voltage**	2.0	—	5.5	V	See **Figure 15-1**
D002	VDR	**RAM Data Retention Voltage(2)**	—	1.5*	—	V	Device in Sleep mode
D003	VPOR	**VDD Start Voltage** to ensure Power-on Reset	—	Vss	—	V	See **Section 8.5 "Power-on Reset (POR)"** for details
D004	SVDD	**VDD Rise Rate** to ensure Power-on Reset	0.05*	—	—	V/ms	See **Section 8.5 "Power-on Reset (POR)"** for details
D005	IDDP	Supply Current During Prog/Erase	—	250*	—	µA	
D010	IDD	**Supply Current(3, 4, 6)**	— —	175 400	— —	µA µA	FOSC = 4 MHz, VDD = 2.0V FOSC = 4 MHz, VDD = 5.0V
			— —	250 0.75	— —	µA mA	FOSC = 8 MHz, VDD = 2.0V FOSC = 8 MHz, VDD = 5.0V
			—	1.4	—	mA	FOSC = 20 MHz, VDD = 5.0V
			— —	11 38	— —	µA µA	FOSC = 32 kHz, VDD = 2.0V FOSC = 32 kHz, VDD = 5.0V
D020	IPD	**Power-down Current(5)**	— —	0.1 0.35	— —	µA µA	VDD = 2.0V VDD = 5.0V
D022	IWDT	**WDT Current(5)**	— —	1.0 7.0	— —	µA µA	VDD = 2.0V VDD = 5.0V
D023	ICMP	**Comparator Current(5)**	— —	15 60	— —	µA µA	VDD = 2.0V (per comparator) VDD = 5.0V (per comparator)
D022	ICVREF	**CVREF Current(5)**	— —	30 75	— —	µA µA	VDD = 2.0V (high range) VDD = 5.0V (high range)
D023	IFVR	**Internal 0.6V Fixed Voltage Reference Current(5)**	— —	100 175	— —	µA µA	VDD = 2.0V (reference and 1 comparator enabled) VDD = 5.0V (reference and 1 comparator enabled)
D024	ΔIAD1*	**A/D Conversion Current**	—	120	—	µA	2.0V, conversion in progress
			—	200	—	µA	5.0V, conversion in progress
D025	ΔIAD2	**A/D Conversion Current**	—	0.20	—	µA	2.0V, no conversion in progress
			—	0.36	—	µA	5.0V, no conversion in progress
D026	ΔIBOR	BOR Current	—	5	—	µA	3.0V
			—	6	—	µA	5.0V

* These parameters are characterized but not tested.

Note 1: Data in the Typical ("Typ") column is based on characterization results at 25°C. This data is for design guidance only and is not tested.

2: This is the limit to which VDD can be lowered in Sleep mode without losing RAM data.

3: The supply current is mainly a function of the operating voltage and frequency. Other factors such as bus loading, oscillator type, bus rate, internal code execution pattern and temperature also have an impact on the current consumption.

4: The test conditions for all IDD measurements in Active Operation mode are:
OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VSS, T0CKI = VDD, $\overline{MCLR}$ = VDD; WDT enabled/disabled as specified.

5: For standby current measurements, the conditions are the same as IDD, except that the device is in Sleep mode. If a module current is listed, the current is for that specific module enabled and the device in Sleep.

6: For EXTRC mode, does not include current through REXT. The current through the resistor can be estimated by the formula:
I = VDD/2REXT (mA) with REXT in kΩ.

15.2 DC Characteristics: PIC16F527 (Extended)

DC Characteristics			Standard Operating Conditions (unless otherwise specified) Operating Temperature -40°C ≤ TA ≤ +125°C (extended)				
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ.(1)**	**Max.**	**Units**	**Conditions**
D001	VDD	**Supply Voltage**	2.0	—	5.5	V	See **Figure 15-1**
D002	VDR	**RAM Data Retention Voltage(2)**	—	1.5*	—	V	Device in Sleep mode
D003	VPOR	**VDD Start Voltage** to ensure Power-on Reset	—	Vss	—	V	See **Section 8.5 "Power-on Reset (POR)"** for details
D004	SVDD	**VDD Rise Rate** to ensure Power-on Reset	0.05*	—	—	V/ms	See **Section 8.5 "Power-on Reset (POR)"** for details
D005	IDDP	Supply Current During Prog/Erase	—	250*	—	µA	
D010	IDD	**Supply Current(3,4,6)**	— —	175 400	— —	µA µA	FOSC = 4 MHz, VDD = 2.0V FOSC = 4 MHz, VDD = 5.0V
			— —	250 0.75	— —	µA mA	FOSC = 8 MHz, VDD = 2.0V FOSC = 8 MHz, VDD = 5.0V
			—	1.4	—	mA	FOSC = 20 MHz, VDD = 5.0V
			— —	11 38	— —	µA µA	FOSC = 32 kHz, VDD = 2.0V FOSC = 32 kHz, VDD = 5.0V
D020	IPD	**Power-down Current(5)**	— —	0.1 0.35	— —	µA µA	VDD = 2.0V VDD = 5.0V
D022	IWDT	**WDT Current(5)**	— —	1.0 7.0	— —	µA µA	VDD = 2.0V VDD = 5.0V
D023	ICMP	**Comparator Current(5)**	— —	15 60	— —	µA µA	VDD = 2.0V (per comparator) VDD = 5.0V (per comparator)
D022	ICVREF	**CVREF Current(5)**	— —	30 75	— —	µA µA	VDD = 2.0V (high range) VDD = 5.0V (high range)
D023	IFVR	**Internal 0.6V Fixed Voltage Reference Current(5)**	— —	100 175	— —	µA µA	VDD = 2.0V (reference and 1 comparator enabled) VDD = 5.0V (reference and 1 comparator enabled)
D024	ΔIAD1*	**A/D Conversion Current**	—	120	—	µA	2.0V, conversion in progress
			—	200	—	µA	5.0V, conversion in progress
D025	ΔIAD2	**A/D Conversion Current**	—	0.20	—	µA	2.0V, no conversion in progress
			—	0.36	—	µA	5.0V, no conversion in progress
D026	ΔIBOR	BOR Current	—	5	—	µA	3.0V
			—	6	—	µA	5.0V

* These parameters are characterized but not tested.

Note 1: Data in the Typical ("Typ") column is based on characterization results at 25°C. This data is for design guidance only and is not tested.

2: This is the limit to which VDD can be lowered in Sleep mode without losing RAM data.

3: The supply current is mainly a function of the operating voltage and frequency. Other factors such as bus loading, oscillator type, bus rate, internal code execution pattern and temperature also have an impact on the current consumption.

4: The test conditions for all IDD measurements in Active Operation mode are:
OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VSS, T0CKI = VDD, $\overline{MCLR}$ = VDD; WDT enabled/disabled as specified.

5: For standby current measurements, the conditions are the same as IDD, except that the device is in Sleep mode. If a module current is listed, the current is for that specific module enabled and the device in Sleep.

6: For EXTRC mode, does not include current through REXT. The current through the resistor can be estimated by the formula:
I = VDD/2REXT (mA) with REXT in kΩ.

TABLE 15-1: DC CHARACTERISTICS: PIC16F527 (Industrial, Extended)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise specified) Operating temperature -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended) Operating voltage VDD range as described in DC spec.				
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ.†**	**Max.**	**Units**	**Conditions**
	VIL	**Input Low Voltage**					
		I/O ports					
D030		with TTL buffer	Vss	—	0.8	V	For all 4.5 ≤ VDD ≤ 5.5V
D030A			Vss	—	0.15 VDD	V	Otherwise
D031		with Schmitt Trigger buffer	Vss	—	0.15 VDD	V	
D032		$\overline{MCLR}$, T0CKI	Vss	—	0.15 VDD	V	
D033		OSC1 (EXTRC mode), EC(1)	Vss	—	0.15 VDD	V	
D033		OSC1 (HS mode)	Vss	—	0.3 VDD	V	
D033		OSC1 (XT and LP modes)	Vss	—	0.3	V	
	VIH	**Input High Voltage**					
		I/O ports		—			
D040		with TTL buffer	2.0	—	VDD	V	4.5 ≤ VDD ≤ 5.5V
D040A			0.25 VDD + 0.8V	—	VDD	V	Otherwise
D041		with Schmitt Trigger buffer	0.85 VDD	—	VDD	V	For entire VDD range
D042		$\overline{MCLR}$, T0CKI	0.85 VDD	—	VDD	V	
D042A		OSC1 (EXTRC mode), EC(1)	0.85 VDD	—	VDD	V	
D042A		OSC1 (HS mode)	0.7 VDD	—	VDD	V	
D043		OSC1 (XT and LP modes)	1.6	—	VDD	V	
D070	IPUR	**PORTB Weak pull-up current(4)**	50	250	400	µA	VDD = 5V, VPIN = VSS
	IIL	**Input Leakage Current(2,5)**					
D060		I/O ports	—	—	±1	µA	Vss ≤ VPIN ≤ VDD, Pin at high-impedance
D061		RB3/$\overline{MCLR}$(3)	—	±0.7	±5	µA	Vss ≤ VPIN ≤ VDD
D063		OSC1	—	—	±5	µA	Vss ≤ VPIN ≤ VDD, XT, HS and LP osc configuration
	VOL	**Output Low Voltage**					
D080		I/O ports/CLKOUT	—	—	0.6	V	IOL = 8.5 mA, VDD = 4.5V, -40°C to +85°C
D080A			—	—	0.6	V	IOL = 7.0 mA, VDD = 4.5V, -40°C to +125°C
	VOH	**Output High Voltage**					
D090		I/O ports/CLKOUT	VDD – 0.7	—	—	V	IOH = -3.0 mA, VDD = 4.5V, -40°C to +85°C
D090A			VDD – 0.7	—	—	V	IOH = -2.5 mA, VDD = 4.5V, -40°C to +125°C
		Capacitive Loading Specs on Output Pins					
D100	COSC2	OSC2 pin	—	—	15	pF	In XT, HS and LP modes when external clock is used to drive OSC1.
D101	CIO	All I/O pins and OSC2	—	—	50	pF	
		Flash Data Memory					
D120	ED	Byte endurance	100K	1M	—	E/W	-40°C ≤ TA ≤ +85°C
D120A	ED	Byte endurance	10K	100K	—	E/W	+85°C ≤ TA ≤ +125°C
D121	VDRW	VDD for read/write	VMIN	—	5.5	V	

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

Note 1: In EXTRC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16F527 be driven with external clock in RC mode.

2: Negative current is defined as coming out of the pin.

3: This spec. applies to RB3/$\overline{MCLR}$ configured as RB3 with pull-up disabled.

4: This spec. applies to all weak pull-up devices, including the weak pull-up found on RB3/$\overline{MCLR}$. The current value listed will be the same whether or not the pin is configured as RB3 with pull-up enabled or as $\overline{MCLR}$.

5: The leakage current on the $\overline{MCLR}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage may be measured at different input voltages.

TABLE 15-2: COMPARATOR SPECIFICATIONS

Comparator Specifications	Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C to 125°C					
Characteristics	**Sym.**	**Min.**	**Typ.**	**Max.**	**Units**	**Comments**
Internal Voltage Reference	VIVRF	0.50	0.60	0.70	V	
Input offset voltage	VOS	—	± 5.0	—	mV	
Input common mode voltage*	VCM	0	—	VDD – 1.5	V	
CMRR*	CMRR	55	—	—	db	
Response Time(1)*	TRT	—	150	—	ns	
Comparator Mode Change to Output Valid*	TMC2COV	—	—	10	µs	

* These parameters are characterized but not tested.

Note 1: Response time measured with one comparator input at (VDD – 1.5)/2 while the other input transitions from VSS to VDD – 1.5V.

TABLE 15-3: COMPARATOR VOLTAGE REFERENCE (VREF) SPECIFICATIONS

Sym.	Characteristics	Min.	Typ.	Max.	Units	Comments
CVRES	Resolution	— —	VDD/24* VDD/32	— —	LSb LSb	Low Range (VRR = 1) High Range (VRR = 0)
	Absolute Accuracy(2)	— —	— —	±1/2* ±1/2*	LSb LSb	Low Range (VRR = 1) High Range (VRR = 0)
	Unit Resistor Value (R)	— —	2K*	—	Ω	
	Settling Time(1)	—	—	10*	µs	

* These parameters are characterized but not tested.

Note 1: Settling time measured while VRR = 1 and VR<3:0> transitions from 0000 to 1111.

2: Do not use reference externally when VDD < 2.7V. Under this condition, reference should only be used with comparator Voltage Common mode observed.

TABLE 15-4: OPERATIONAL AMPLIFIER (OPA) MODULE DC SPECIFICATIONS

OPA DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) V_{CM} = 0V, V_{OUT} = V_{DD}/2, V_{DD} = 5.0V, V_{SS} = 0V, C_L = 50 pF, R_L = 100k Operating temperature -40°C ≤ T_A ≤ +125°C				
Param No.	**Sym.**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Comments**
OPA01	V_{OS}	**Input Offset Voltage**	—	±5	—	mV	
		Input current and impedance					
OPA02*	I_B	Input bias current	—	±2*	—	nA	
OPA03*	I_{OS}	Input offset bias current	—	±1*	—	pA	
		Common Mode					
OPA04*	V_{CM}	Common mode input range	V_{SS}	—	V_{DD} – 1.5	V	V_{DD} = 5.0V
OPA05*	CMR	Common mode rejection	65	70	—	dB	V_{CM} = V_{DD}/2, Freq. = DC
		Open Loop Gain					
OPA06A*	A_{OL}	DC Open loop gain	—	90	—	dB	No load
OPA06B*	A_{OL}	DC Open loop gain	—	60	—	dB	Standard load
		Output					
OPA07*	Vout	Output voltage swing	V_{SS}+100	—	V_{DD} – 100	mV	To V_{DD}/2 (20 kΩ connected to V_{DD}, 20 kΩ + 20 pF to Vss)
OPA08*	Isc	Output short circuit current	—	25	28	mA	
		Power Supply					
OPA10*	PSR	Power supply rejection	80	—	—	dB	

* These parameters are characterized but not tested.

TABLE 15-5: OPERATIONAL AMPLIFIER (OPA) MODULE AC SPECIFICATIONS

OPA AC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) V_{CM} = 0V, V_{OUT} = V_{DD}/2, V_{DD} = 5.0V, V_{SS} = 0V, C_L = 50 pF, R_L = 100k Operating temperature -40°C ≤ T_A ≤ +125°C				
Param No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Comments**
OPA11*	GBWP	Gain bandwidth product	—	3	—	MHz	
OPA12*	T_{ON}	Turn on time	—	10	15	μs	
OPA13*	ΘM	Phase margin	—	60	—	deg	
OPA14*	SR	Slew rate	2	—	—	V/μs	

* These parameters are characterized but not tested.

TABLE 15-6: A/D CONVERTER CHARACTERISTICS

A/D Converter Specifications			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C				
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ.†**	**Max.**	**Units**	**Conditions**
A01	NR	Resolution	—	—	8	bit	
A03	EINL	Integral Error	—	—	±1.5	LSb	VDD = 5.0V
A04	EDNL	Differential Error	—	—	-1< EDNL ≤1.7	LSb	No missing codes to eight bits VDD = 5.0V
A06	EOFF	Offset Error	—	—	±1.5	LSb	VDD = 5.0V
A07	EGN	Gain Error	-0.7	—	+2.2	LSb	VDD = 5.0V
A10	—	Monotonicity	—	guaranteed(1)	—	—	VSS ≤ VAIN ≤ VDD
A25	VAIN	Analog Input Voltage	VSS	—	VDD	V	
A30	ZAIN	Recommended Impedance of Analog Voltage Source	—	—	10	KΩ	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The A/D conversion result never decreases with an increase in the input voltage and has no missing codes.

TABLE 15-7: PULL-UP RESISTOR RANGES

VDD (Volts)	Temperature (°C)	Min.	Typ.	Max.	Units
RA0/RA1/RA4					
2.0	-40	73K	105K	186K	Ω
	25	73K	113K	187K	Ω
	85	82K	123K	190K	Ω
	125	86K	132k	190K	Ω
5.5	-40	15K	21K	33K	Ω
	25	15K	22K	34K	Ω
	85	19K	26k	35K	Ω
	125	23K	29K	35K	Ω
RA3					
2.0	-40	63K	81K	96K	Ω
	25	77K	93K	116K	Ω
	85	82K	96k	116K	Ω
	125	86K	100K	119K	Ω
5.5	-40	16K	20k	22K	Ω
	25	16K	21K	23K	Ω
	85	24K	25k	28K	Ω
	125	26K	27K	29K	Ω

15.3 Timing Parameter Symbology and Load Conditions

The timing parameter symbols have been created following one of the following formats:

1. TppS2ppS
2. TppS

T			
F	Frequency	T	Time

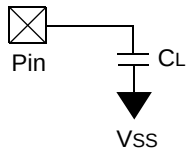
Lowercase subscripts (pp) and their meanings:

pp			
2	to	mc	$\overline{MCLR}$
ck	CLKOUT	osc	Oscillator
cy	Cycle time	os	OSC1
drt	Device Reset Timer	t0	T0CKI
io	I/O port	wdt	Watchdog Timer

Uppercase letters and their meanings:

S			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (high-impedance)	V	Valid
L	Low	Z	High-impedance

FIGURE 15-3: LOAD CONDITIONS



Legend:

C_L = 50 pF for all pins except OSC2

15 pF for OSC2 in XT, HS or LP modes when external clock is used to drive OSC1

FIGURE 15-4: EXTERNAL CLOCK TIMING

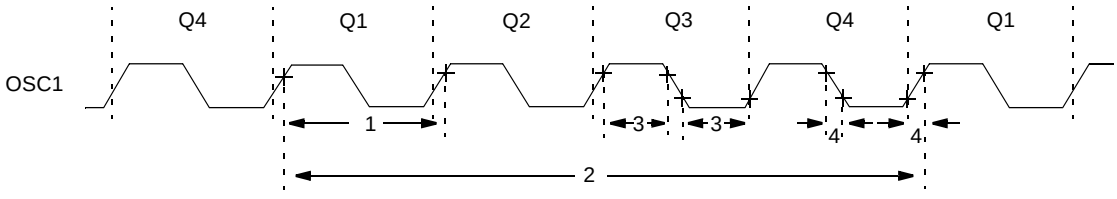


TABLE 15-8: EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions (unless otherwise specified) Operating Temperature -40°C ≤ TA ≤ +85°C (industrial), -40°C ≤ TA ≤ +125°C (extended) Operating Voltage VDD range is described in **Section 15.1 "DC Characteristics: PIC16F527 (Industrial)"**				
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ.(1)**	**Max.**	**Units**	**Conditions**
1A	FOSC	External CLKIN Frequency(2)	DC	—	4	MHz	XT Oscillator mode
			DC	—	20	MHz	HS/EC Oscillator mode
			DC	—	200	kHz	LP Oscillator mode
		Oscillator Frequency(2)	—	—	4	MHz	EXTRC Oscillator mode
			0.1	—	4	MHz	XT Oscillator mode
			4	—	20	MHz	HS/EC Oscillator mode
			—	—	200	kHz	LP Oscillator mode
1	TOSC	External CLKIN Period(2)	250	—	—	ns	XT Oscillator mode
			50	—	—	ns	HS/EC Oscillator mode
			5	—	—	µs	LP Oscillator mode
		Oscillator Period(2)	250	—	—	ns	EXTRC Oscillator mode
			250	—	10,000	ns	XT Oscillator mode
			50	—	250	ns	HS/EC Oscillator mode
			5	—	—	µs	LP Oscillator mode
2	TCY	Instruction Cycle Time	200	4/FOSC	—	ns	
3	TosL, TosH	Clock in (OSC1) Low or High Time	50*	—	—	ns	XT Oscillator
			2*	—	—	µs	LP Oscillator
			10*	—	—	ns	HS/EC Oscillator
4	TosR, TosF	Clock in (OSC1) Rise or Fall Time	—	—	25*	ns	XT Oscillator
			—	—	50*	ns	LP Oscillator
			—	—	15*	ns	HS/EC Oscillator

* These parameters are characterized but not tested.

Note 1: Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

2: All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

TABLE 15-9: CALIBRATED INTERNAL RC FREQUENCIES

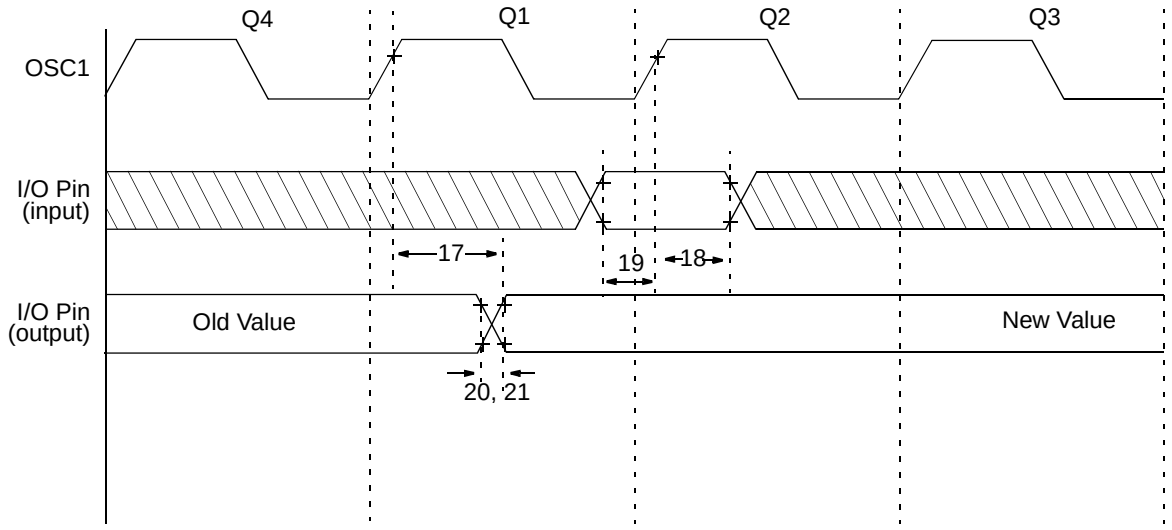
AC CHARACTERISTICS			Standard Operating Conditions (unless otherwise specified) Operating Temperature -40°C ≤ TA ≤ +85°C (industrial), -40°C ≤ TA ≤ +125°C (extended) Operating Voltage VDD range is described in Section 15.1 "DC Characteristics: PIC16F527 (Industrial)"					
Param No.	**Sym.**	**Characteristic**	**Freq. Tolerance**	**Min.**	**Typ.†**	**Max.**	**Units**	**Conditions**
F10	FOSC	Internal Calibrated INTOSC Frequency(1)	± 1%	7.92	8.00	8.08	MHz	3.5V, +25°C
			± 2%	7.84	8.00	8.16	MHz	2.5V ≤ VDD ≤ 5.5V 0°C ≤ TA ≤ +85°C
			± 5%	7.60	8.00	8.40	MHz	2.0V ≤ VDD ≤ 5.5V -40°C ≤ TA ≤ +85°C (Ind.) -40°C ≤ TA ≤ +125°C (Ext.)

* These parameters are characterized but not tested.

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

Note 1: To ensure these oscillator frequency tolerances, VDD and VSS must be capacitively decoupled as close to the device as possible. 0.1 uF and 0.01 uF values in parallel are recommended.

FIGURE 15-5: I/O TIMING



Note: All tests must be done with specified capacitive loads (see data sheet) 50 pF on I/O pins and CLKOUT.

TABLE 15-10: TIMING REQUIREMENTS

AC CHARACTERISTICS		**Standard Operating Conditions (unless otherwise specified)** Operating Temperature -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended) Operating Voltage VDD range is described in **Section 15.1 "DC Characteristics: PIC16F527 (Industrial)"**				
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ.(1)**	**Max.**	**Units**
17	TOSH2IOV	OSC1↑ (Q1 cycle) to Port Out Valid(2), (3)	—	—	100*	ns
18	TOSH2IOI	OSC1↑ (Q2 cycle) to Port Input Invalid (I/O in hold time)(2)	50	—	—	ns
19	TIOV2OSH	Port Input Valid to OSC1↑ (I/O in setup time)	20	—	—	ns
20	TIOR	Port Output Rise Time(3)	—	10	50**	ns
21	TIOF	Port Output Fall Time(3)	—	10	58**	ns

* These parameters are characterized but not tested.

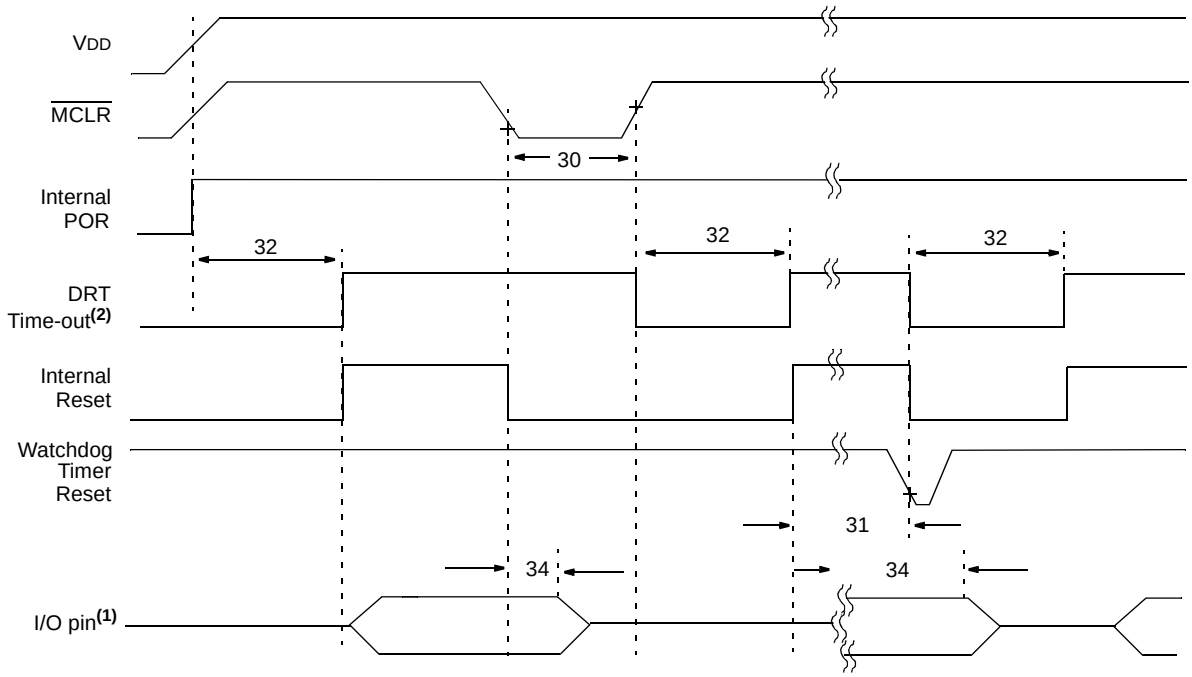
** These parameters are design targets and are not tested.

Note 1: Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

2: Measurements are taken in EXTRC mode.

3: See Figure 15-3 for loading conditions.

FIGURE 15-6: RESET, WATCHDOG TIMER AND DEVICE RESET TIMER TIMING



Note 1: I/O pins must be taken out of High-Impedance mode by enabling the output drivers in software.

2: Runs in MCLR or WDT Reset only in XT, LP and HS modes.

TABLE 15-11: RESET, WATCHDOG TIMER AND DEVICE RESET TIMER

AC CHARACTERISTICS			Standard Operating Conditions (unless otherwise specified) Operating Temperature -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended) Operating Voltage VDD range is described in Section 15.1 "DC Characteristics: PIC16F527 (Industrial)"				
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ.(1)**	**Max.**	**Units**	**Conditions**
30	TMCL	$\overline{\text{MCLR}}$ Pulse Width (low)	2000*	—	—	ns	VDD = 5.0V
31	TWDT	Watchdog Timer Time-out Period (no prescaler)	9* 9*	18* 18*	30* 40*	ms ms	VDD = 5.0V (Industrial) VDD = 5.0V (Extended)
32	TDRT	Device Reset Timer Period					
		Standard	9* 9*	18* 18*	30* 40*	ms ms	VDD = 5.0V (Industrial) VDD = 5.0V (Extended)
34	TIOZ	I/O High-impedance from $\overline{\text{MCLR}}$ low	—	—	2000*	ns	

* These parameters are characterized but not tested.

Note 1: Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 15-7: TIMER0 CLOCK TIMINGS

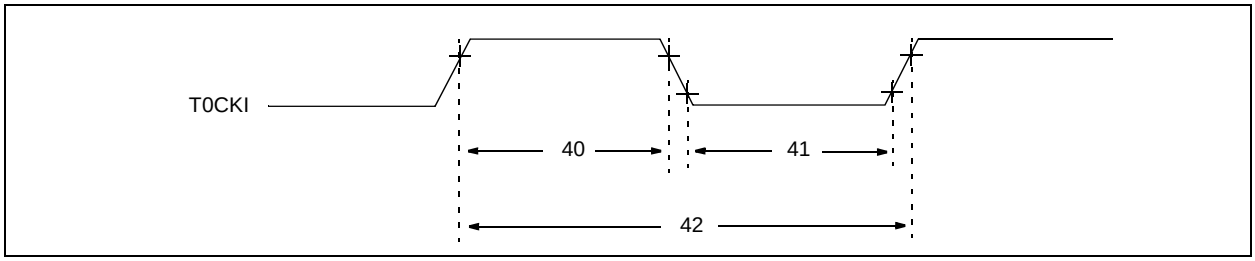


TABLE 15-12: TIMER0 CLOCK REQUIREMENT

AC CHARACTERISTICS				Standard Operating Conditions (unless otherwise specified) Operating Temperature -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended) Operating Voltage VDD range is described in Section 15.1 "DC Characteristics: PIC16F527 (Industrial)"				
Param No.	**Sym.**	**Characteristic**		**Min.**	**Typ.(1)**	**Max.**	**Units**	**Conditions**
40	Tt0H	T0CKI High Pulse Width	No Prescaler	0.5 TCY + 20*	—	—	ns	
			With Prescaler	10*	—	—	ns	
41	Tt0L	T0CKI Low Pulse Width	No Prescaler	0.5 TCY + 20*	—	—	ns	
			With Prescaler	10*	—	—	ns	
42	Tt0P	T0CKI Period		20 or TCY + 40* N	—	—	ns	Whichever is greater. N = Prescale Value (1, 2, 4,..., 256)

* These parameters are characterized but not tested.

Note 1: Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

TABLE 15-13: FLASH DATA MEMORY WRITE/ERASE TIME

AC CHARACTERISTICS			Standard Operating Conditions (unless otherwise specified) Operating Temperature -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended) Operating Voltage VDD range is described in Section 15.1 "DC Characteristics: PIC16F527 (Industrial)"				
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ.(1)**	**Max.**	**Units**	**Conditions**
43	TDW	Flash Data Memory Write Cycle Time	2	3.5	5	ms	
44	TDE	Flash Data Memory Erase Cycle Time	2	3.5	5	ms	

* These parameters are characterized but not tested.

Note 1: Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

16.0 DC AND AC CHARACTERISTICS GRAPHS AND CHARTS

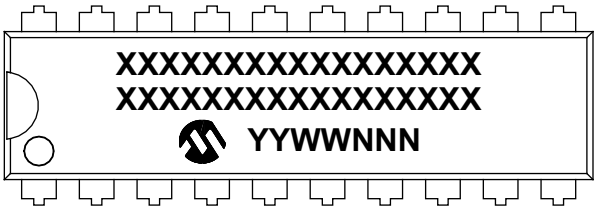
Graphs and tables are not available at this time.

NOTES:

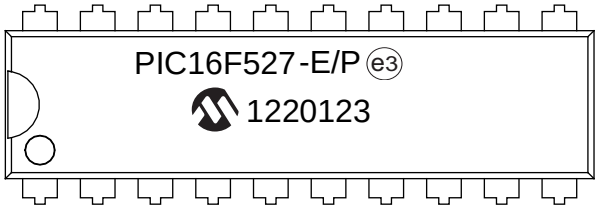
17.0 PACKAGING INFORMATION

17.1 Package Marking Information

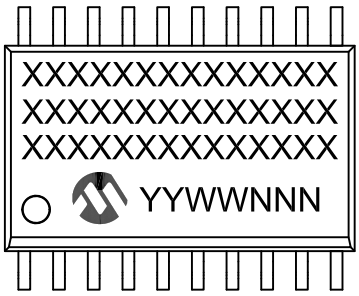
20-Lead PDIP (300 mil)



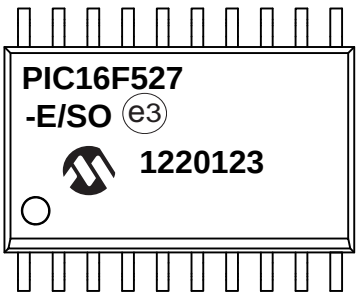
Example



20-Lead SOIC (7.50 mm)



Example



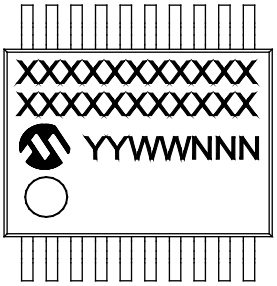
Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week ‘01’)
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

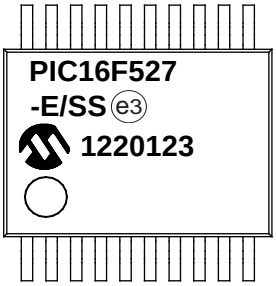
***** Standard PICmicro® device marking consists of Microchip part number, year code, week code and traceability code. For PICmicro device marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

Package Marking Information (Continued)

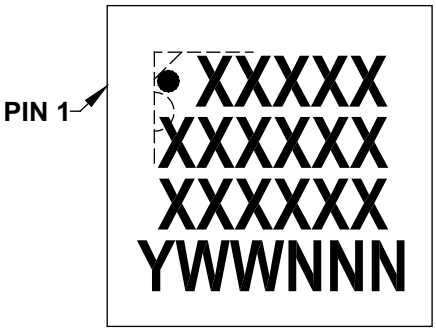
20-Lead SSOP (5.30 mm)



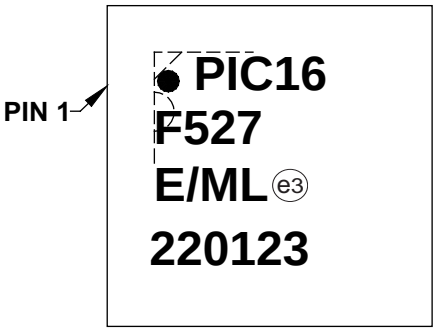
Example



20-Lead QFN (4x4x0.9 mm)



Example



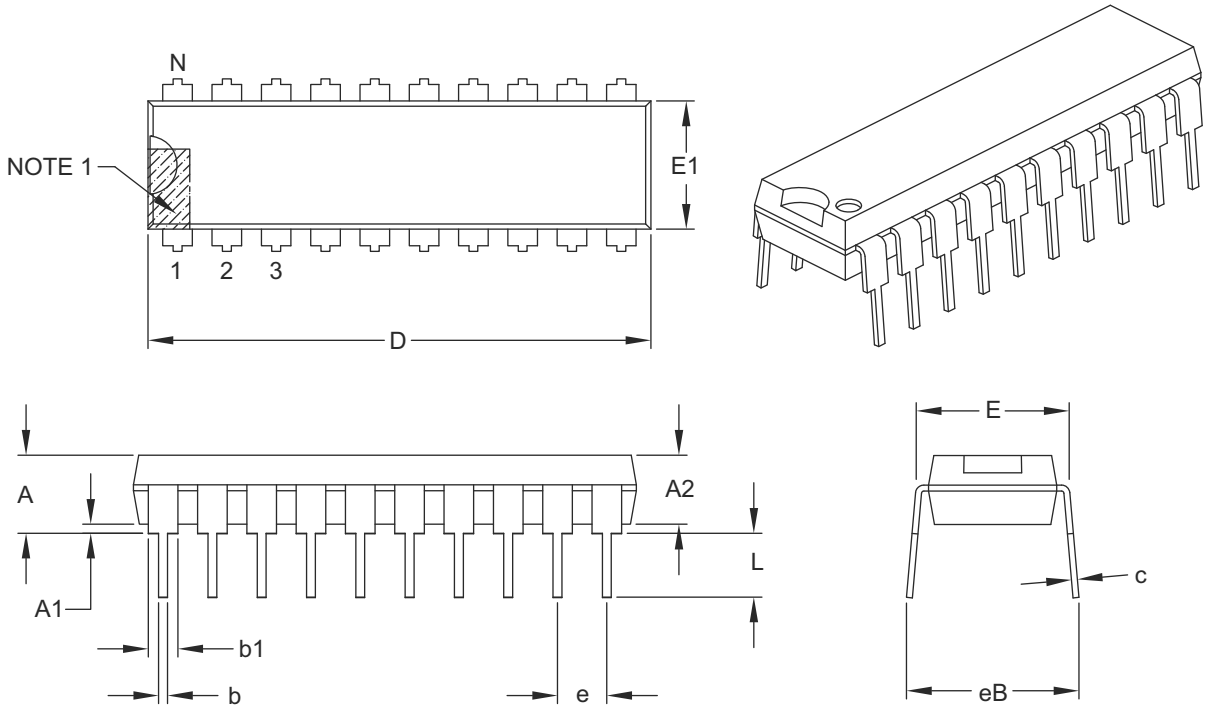
Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week ‘01’)
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

***** Standard PICmicro® device marking consists of Microchip part number, year code, week code and traceability code. For PICmicro device marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

20-Lead Plastic Dual In-Line (P) – 300 mil Body [PDIP]

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



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		20	
Pitch	e		.100 BSC	
Top to Seating Plane	A	–	–	.210
Molded Package Thickness	A2	.115	.130	.195
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.300	.310	.325
Molded Package Width	E1	.240	.250	.280
Overall Length	D	.980	1.030	1.060
Tip to Seating Plane	L	.115	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.045	.060	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	–	–	.430

Notes:

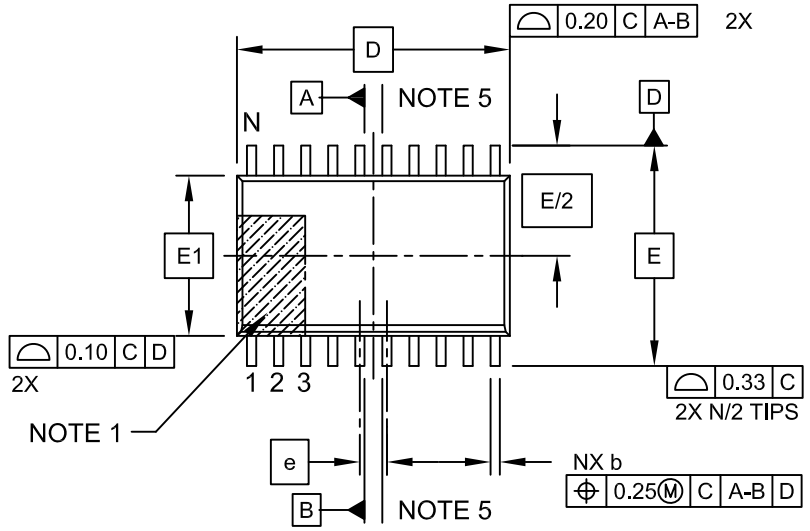
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

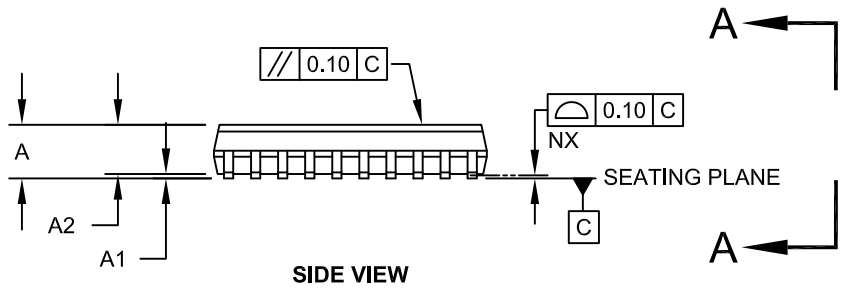
Microchip Technology Drawing C04-019B

20-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

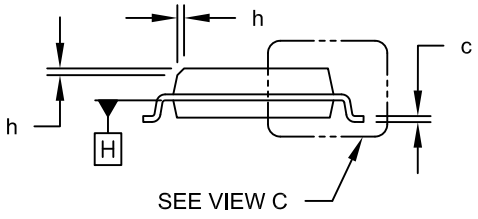
Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



TOP VIEW



SIDE VIEW

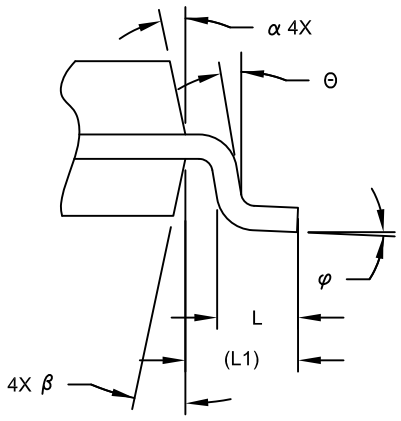


VIEW A-A

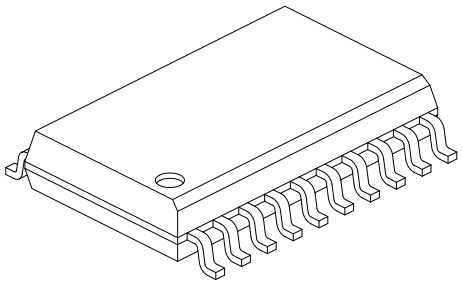
Microchip Technology Drawing C04-094C Sheet 1 of 2

20-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

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



VIEW C



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		20	
Pitch	e		1.27 BSC	
Overall Height	A	-	-	2.65
Molded Package Thickness	A2	2.05	-	-
Standoff §	A1	0.10	-	0.30
Overall Width	E		10.30 BSC	
Molded Package Width	E1		7.50 BSC	
Overall Length	D		12.80 BSC	
Chamfer (Optional)	h	0.25	-	0.75
Foot Length	L	0.40	-	1.27
Footprint	L1		1.40 REF	
Lead Angle	Θ	0°	-	-
Foot Angle	φ	0°	-	8°
Lead Thickness	c	0.20	-	0.33
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. Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M

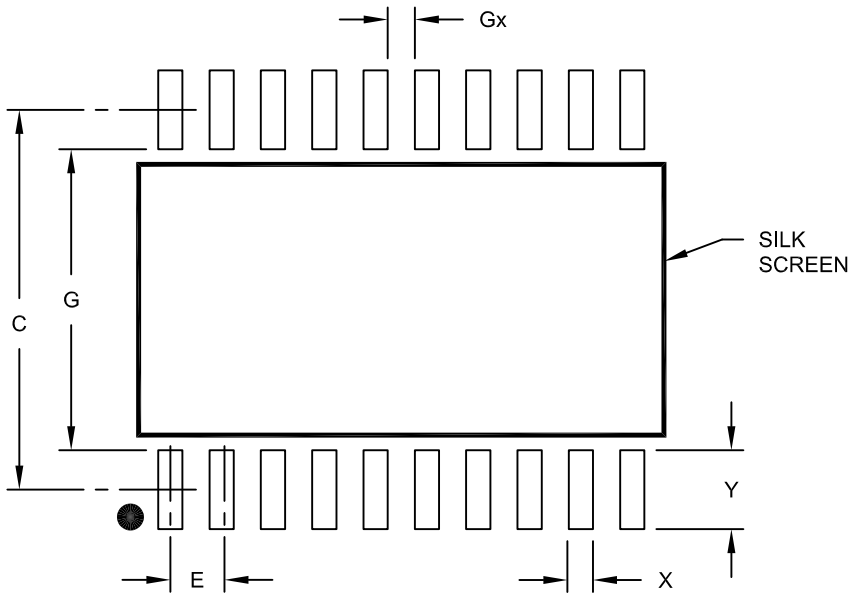
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.
5. Datums A & B to be determined at Datum H.

Microchip Technology Drawing No. C04-094C Sheet 2 of 2

20-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		1.27 BSC	
Contact Pad Spacing	C		9.40	
Contact Pad Width (X20)	X			0.60
Contact Pad Length (X20)	Y			1.95
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	7.45		

Notes:

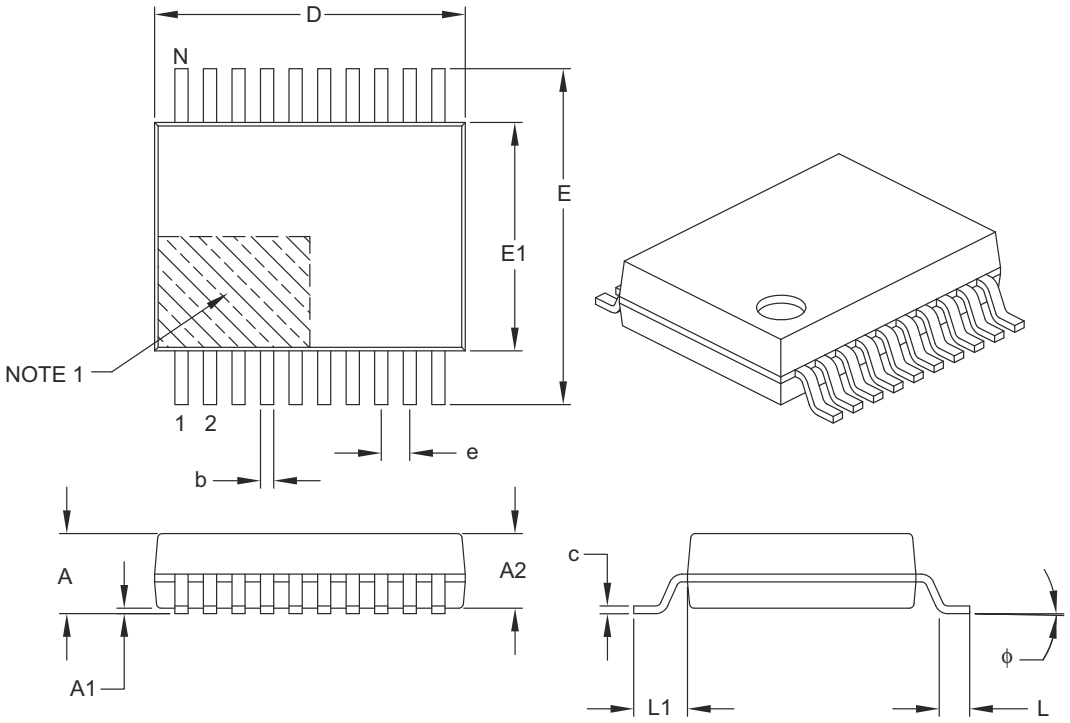
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2094A

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.

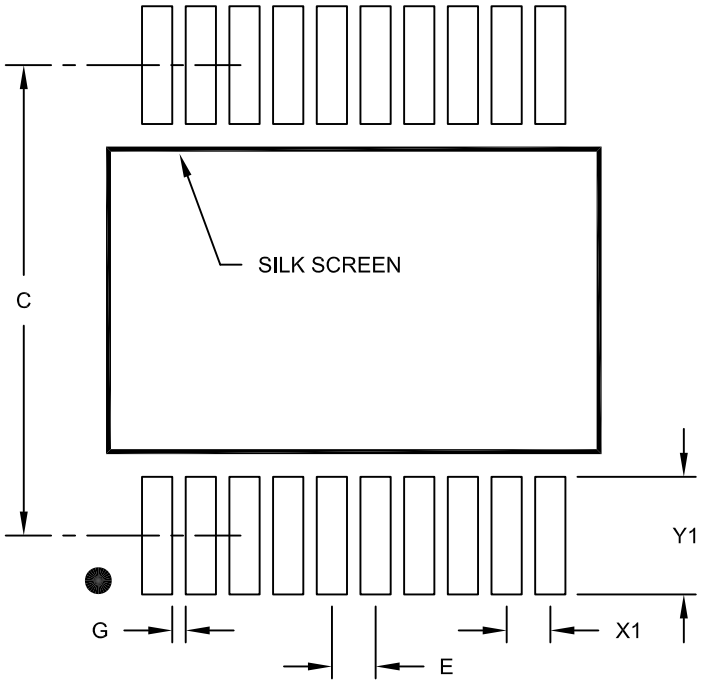
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-072B

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.65 BSC	
Contact Pad Spacing	C		7.20	
Contact Pad Width (X20)	X1			0.45
Contact Pad Length (X20)	Y1			1.75
Distance Between Pads	G	0.20		

Notes:

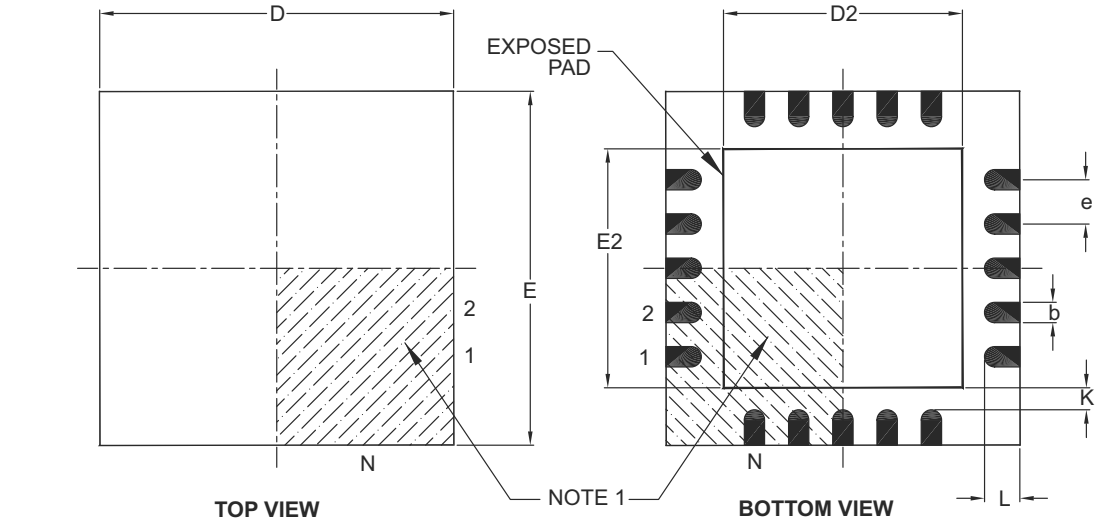
1. Dimensioning and tolerancing per ASME Y14.5M

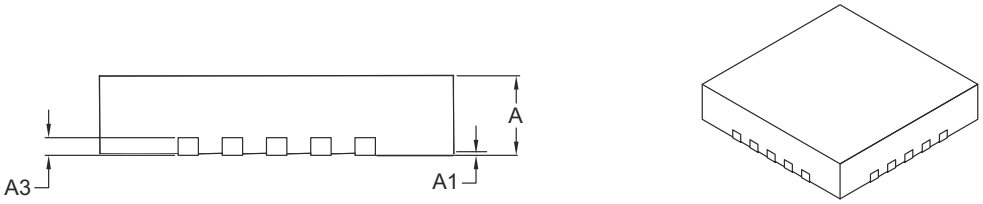
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2072A

20-Lead Plastic Quad Flat, No Lead Package (ML) – 4x4x0.9 mm Body [QFN]

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		0.50 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3		0.20 REF	
Overall Width	E		4.00 BSC	
Exposed Pad Width	E2	2.60	2.70	2.80
Overall Length	D		4.00 BSC	
Exposed Pad Length	D2	2.60	2.70	2.80
Contact Width	b	0.18	0.25	0.30
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	–	–

Notes:

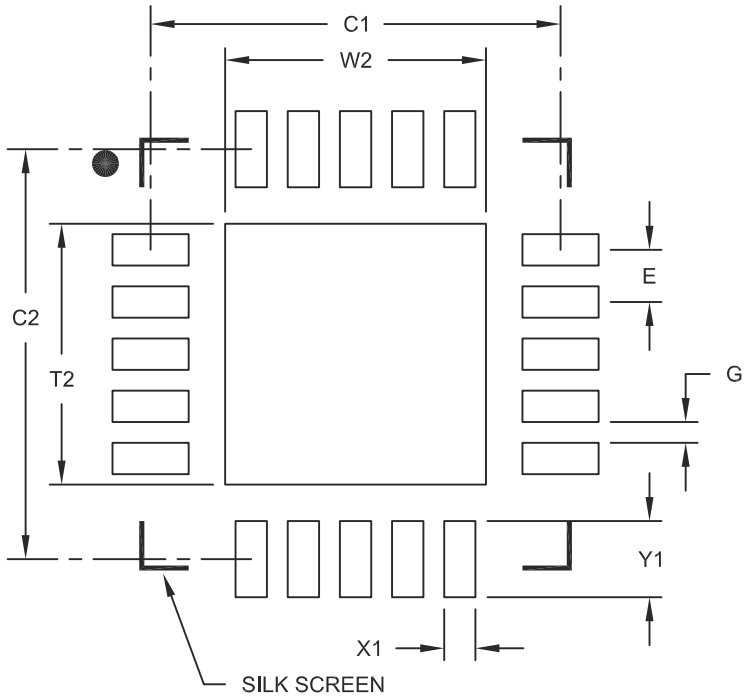
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-126B

20-Lead Plastic Quad Flat, No Lead Package (ML) - 4x4 mm Body [QFN]
With 0.40 mm Contact Length

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.50 BSC	
Optional Center Pad Width	W2			2.50
Optional Center Pad Length	T2			2.50
Contact Pad Spacing	C1		3.93	
Contact Pad Spacing	C2		3.93	
Contact Pad Width	X1			0.30
Contact Pad Length	Y1			0.73
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2126A

APPENDIX A: DATA SHEET REVISION HISTORY

Revision A (09/2012)

Initial release of this document.

NOTES:

INDEX

T

W

Z

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://microchip.com/support

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 Total Pages Sent _____

RE: Reader Response

From: Name _____

Company _____

Address _____

City / State / ZIP / Country _____

Telephone: (_____) _____ - _____ FAX: (_____) _____ - _____

Application (optional):

Would you like a reply? ___ Y ___ N

Device: PIC16F527 Literature Number: DS41652A

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?

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:	PIC16F527	
Temperature Range:	I =	-40°C to +85°C (Industrial)
	E =	-40°C to +125°C (Extended)
Package:	P =	Plastic (PDIP)(2)
	SO =	20L Small Outline, 7.50 mm (SOIC)(1,2)
	SS =	Shrink Small Outline (SSOP)(1,2)
	ML =	20-Lead 4x4 (QFN)(1,2)
Pattern:	Special Requirements	

Examples:

a) PIC16F527-E/P 301 = Extended Temp., PDIP package, QTP pattern #301

b) PIC16F527-I/SO = Industrial Temp., SOIC package

c) PIC16F527T-E/SS = Extended Temp., SSOP package, Tape and Reel

d) PIC16F527T-I/ML = Industrial Temp., QFN Package, Tape and Reel

Note 1: T = in tape and reel SOIC, SSOP and QFN packages only

2: Pb-free.

NOTES:

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

ISBN: 978-1-62076-527-2

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949

Microchip received ISO/TS-16949:2009 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.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario, Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hangzhou
Tel: 86-571-2819-3187
Fax: 86-571-2819-3189

China - Hong Kong SAR
Tel: 852-2401-1200
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Osaka
Tel: 81-66-152-7160
Fax: 81-66-152-9310

Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-536-4818
Fax: 886-7-330-9305

Taiwan - Taipei
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

11/29/11