



**PIC12F629/675**  
**Data Sheet**

8-Pin FLASH-Based 8-Bit  
CMOS Microcontrollers

---

**Note the following details of the code protection feature on PICmicro® MCUs.**

- The PICmicro family meets the specifications contained in the Microchip Data Sheet.
- Microchip believes that its family of PICmicro microcontrollers is one of the most secure products 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 PICmicro microcontroller in a manner outside the operating specifications contained in the data sheet. The person doing so may be 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 product.

If you have any further questions about this matter, please contact the local sales office nearest to you.

---

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.

#### Trademarks

The Microchip name and logo, the Microchip logo, FilterLab, KEELOQ, microID, MPLAB, PIC, PICmicro, PICMASTER, PICSTART, PRO MATE, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

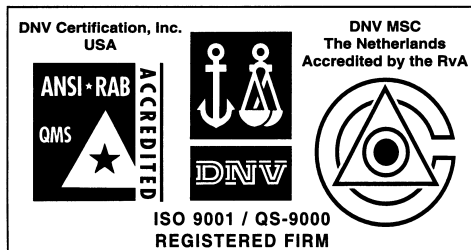
dsPIC, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, microPort, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, MXDEV, PICC, PICDEM, PICDEM.net, rfPIC, Select Mode and Total Endurance are trademarks of Microchip Technology Incorporated in the U.S.A.

Serialized Quick Turn Programming (SQTP) is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2002, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.



*Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company's quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001 certified.*

## 8-Pin FLASH-Based 8-Bit CMOS Microcontroller

### Devices included in this Data Sheet:

- PIC12F629
- PIC12F675

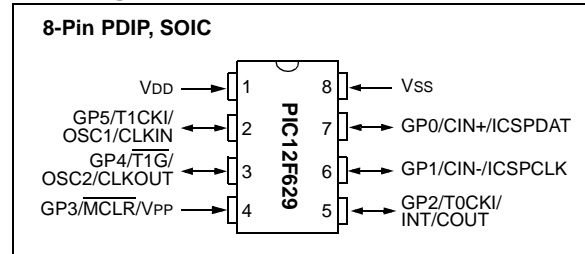
### High Performance RISC CPU:

- Only 35 instructions to learn
- All single cycle instructions (200 ns), except for program branches which are two-cycle
- Operating speed:
  - DC - 20 MHz oscillator/clock input
  - DC - 200 ns instruction cycle
- Memory
  - 1024 x 14 words of FLASH Program Memory
  - 64 x 8 bytes of Data Memory (SRAM)
  - 128 x 8 bytes of EEPROM data memory
- Interrupt capability
- 16 special function hardware registers
- 8-level deep hardware stack
- Direct, Indirect, and Relative Addressing modes

### Peripheral Features:

- 6 I/O pins with individual direction control
- High current sink/source for direct LED drive
- Analog comparator module with:
  - One analog comparator
  - Programmable on-chip comparator voltage reference (CVREF) module
  - Programmable input multiplexing from device inputs
  - Comparator output is externally accessible
- Analog-to-Digital Converter module (PIC12F675):
  - 10-bit resolution
  - Programmable 4-channel input
  - Voltage reference input
- Timer0: 8-bit timer/counter with 8-bit programmable prescaler
- Enhanced Timer1:
  - 16-bit timer/counter with prescaler
  - External Gate Input mode
  - Option to use OSC1 and OSC2 in LP mode as Timer1 oscillator, if INTRC Oscillator mode selected
- 64 bytes of general purpose RAM

### Pin Diagram



### Special Microcontroller Features:

- Low power Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Low power Brown-out Detect (BOD)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Multiplexed  $\overline{\text{MCLR}}$  pin
- Interrupt-on-pin change
- Individual programmable weak pull-ups
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
  - RC: External RC oscillator
  - INTOSC: 4 MHz internal oscillator
  - EC: External Clock input
  - XT: Standard crystal/resonator
  - HS: High speed crystal/resonator
  - LP: Power saving, low frequency crystal
- In-Circuit Serial Programming™ (ICSP™) via two pins
- Four user programmable ID locations

### CMOS Technology:

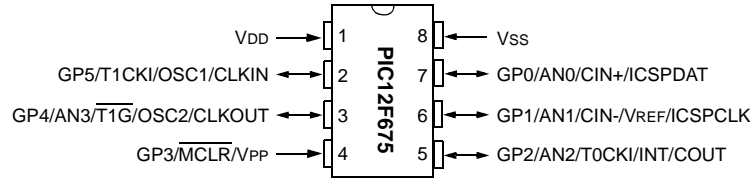
- Low power, high speed CMOS FLASH technology
- Fully static design
- Wide operating voltage range
  - PIC12F629/675 - 2.0V to 5.5V
- Industrial and Extended temperature range
- Low power consumption
  - < 1.0 mA @ 5.5V, 4.0 MHz
  - 20  $\mu\text{A}$  typical @ 2.0V, 32 kHz
  - < 1.0  $\mu\text{A}$  typical standby current @ 2.0V

\* 8-bit, 8-pin devices protected by Microchip's Low Pin Count Patent: U.S. Patent No. 5,847,450. Additional U.S. and foreign patents and applications may be issued or pending.

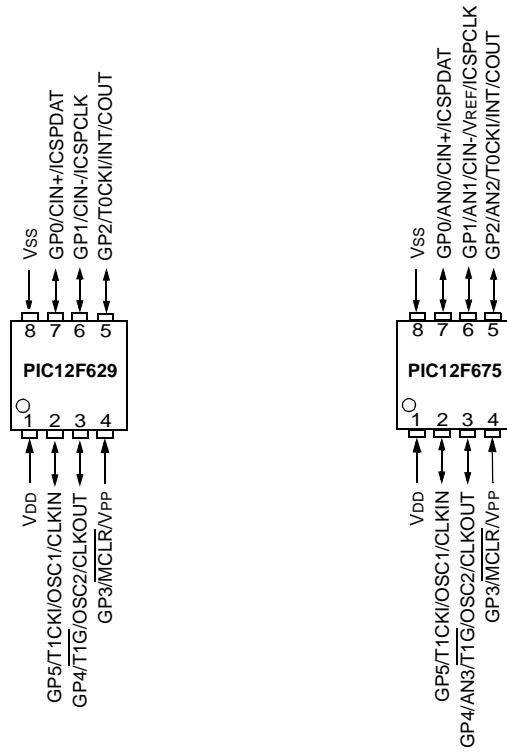
# PIC12F629/675

## Pin Diagrams

### 8-pin PDIP, SOIC



### 8-pin MLF-S



## Table of Contents

1.0	Device Overview .....	5
2.0	Memory Organization .....	7
3.0	GPIO Port .....	19
4.0	Timer0 Module .....	25
5.0	Timer1 Module with Gate Control.....	28
6.0	Comparator Module.....	33
7.0	Analog-to-Digital Converter (A/D) Module (PIC12F675 only).....	39
8.0	Data EEPROM Memory .....	47
9.0	Special Features of the CPU.....	51
10.0	Instruction Set Summary .....	69
11.0	Development Support.....	77
12.0	Electrical Specifications.....	83
13.0	Packaging Information.....	101
Appendix A:	Data Sheet Revision History .....	107
Appendix B:	Device Differences.....	107
Appendix C:	Device Migrations .....	108
Appendix D:	Migrating from other PICmicro Devices .....	108
Appendix E:	Development Tool Version Requirements .....	109
Index .....		111
On-Line Support .....		115
Reader Response .....		116
Product Identification System .....		117

## 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@mail.microchip.com](mailto:docerrors@mail.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)
- The Microchip Corporate Literature Center; U.S. FAX: (480) 792-7277

When contacting a sales office or the literature center, 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/cn](http://www.microchip.com/cn) to receive the most current information on all of our products.

# PIC12F629/675

---

NOTES:

# PIC12F629/675

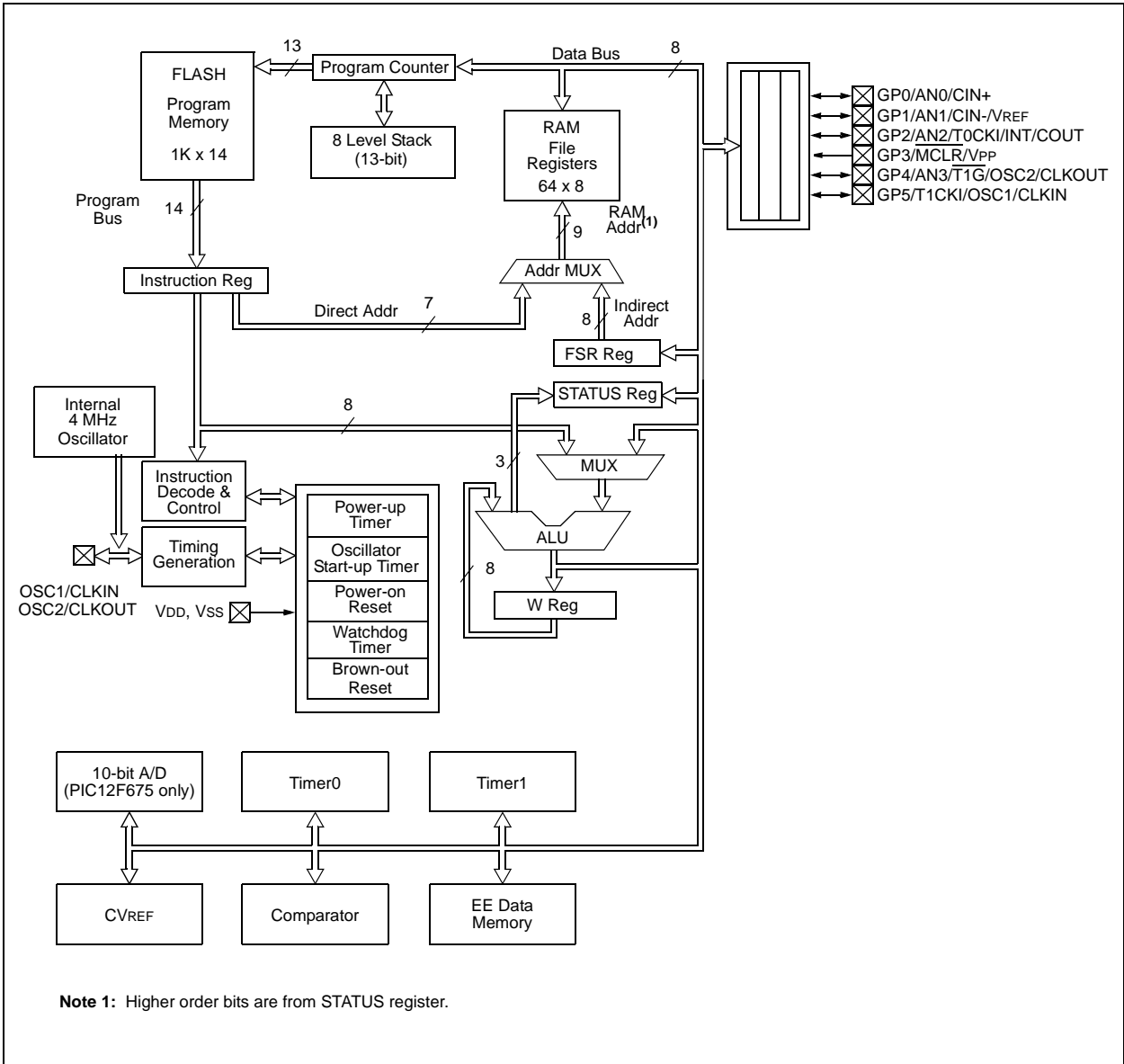
## 1.0 DEVICE OVERVIEW

This document contains device specific information for the PIC12F629/675. Additional information may be found in the PICmicro™ Mid-Range Reference Manual (DS33023), which may be obtained from your local Microchip Sales Representative or downloaded from the Microchip web site. The Reference Manual should be considered a complementary document to this Data

Sheet, and is highly recommended reading for a better understanding of the device architecture and operation of the peripheral modules.

The PIC12F629 and PIC12F675 devices are covered by this Data Sheet. They are identical, except the PIC12F675 has a 10-bit A/D converter. They come in 8-pin PDIP, SOIC, and MLF-S packages. Figure 1-1 shows a block diagram of the PIC12F629/675 devices. Table 1-1 shows the Pinout Description.

**FIGURE 1-1: PIC12F629/675 BLOCK DIAGRAM**



# PIC12F629/675

**TABLE 1-1: PIC12F629/675 PINOUT DESCRIPTION**

Name	Function	Input Type	Output Type	Description
GP0/AN0/CIN+/ICSPDAT	GP0	TTL	CMOS	Bi-directional I/O w/ programmable pull-up and interrupt-on-change
	AN0	AN		A/D Channel 0 input (PIC12F675 only)
	CIN+	AN		Comparator input
	ICSPDAT	TTL	CMOS	Serial programming I/O
GP1/AN1/CIN-/VREF/ICSPCLK	GP1	TTL	CMOS	Bi-directional I/O w/ programmable pull-up and interrupt-on-change
	AN1	AN		A/D Channel 1 input (PIC12F675 only)
	CIN-	AN		Comparator input
	VREF	AN		External voltage reference (PIC12F675 only)
ICSPCLK	ST		Serial programming clock	
GP2/AN2/T0CKI/INT/COU $\bar{T}$	GP2	ST	CMOS	Bi-directional I/O w/ programmable pull-up and interrupt-on-change
	AN2	AN		A/D Channel 2 input (PIC12F675 only)
	T0CKI	ST		TMR0 clock input
	INT	ST		External interrupt
COU $\bar{T}$		CMOS	Comparator output	
GP3/ $\overline{MCLR}$ /VPP	GP3	TTL		Input port w/ interrupt-on-change
	$\overline{MCLR}$	ST		Master Clear
	VPP	HV		Programming voltage
GP4/AN3/ $\overline{T1G}$ /OSC2/CLKOUT	GP4	TTL	CMOS	Bi-directional I/O w/ programmable pull-up and interrupt-on-change
	AN3	AN		A/D Channel 3 input (PIC12F675 only)
	$\overline{T1G}$	ST		TMR1 gate
	OSC2		XTAL	Crystal/resonator
CLKOUT		CMOS	Fosc/4 output	
GP5/T1CKI/OSC1/CLKIN	GP5	TTL	CMOS	Bi-directional I/O w/ programmable pull-up and interrupt-on-change
	T1CKI	ST		TMR1 clock
	OSC1	XTAL		Crystal/resonator
	CLKIN	ST		External clock input/RC oscillator connection
VSS	VSS	Power		Ground reference
VDD	VDD	Power		Positive supply

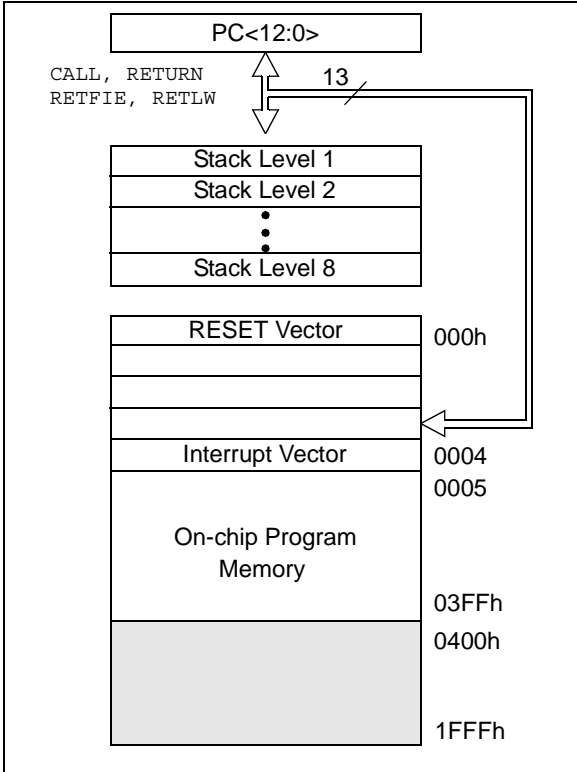


## 2.0 MEMORY ORGANIZATION

### 2.1 Program Memory Organization

The PIC12F629/675 devices have a 13-bit program counter capable of addressing an 8K x 14 program memory space. Only the first 1K x 14 (0000h - 03FFh) for the PIC12F629/675 devices are physically implemented. Accessing a location above these boundaries will cause a wrap around within the first 1K x 14 space. The RESET vector is at 0000h and the interrupt vector is at 0004h (see Figure 2-1).

**FIGURE 2-1: PROGRAM MEMORY MAP AND STACK FOR THE PIC12F629/675**



### 2.2 Data Memory Organization

The data memory (see Figure 2-2) is partitioned into two banks, which contain the General Purpose registers and the Special Function registers. The Special Function registers are located in the first 32 locations of each bank. Register locations 20h-5Fh are General Purpose registers, implemented as static RAM and are mapped across both banks. All other RAM is unimplemented and returns '0' when read. RP0 (STATUS<5>) is the bank select bit.

- RP0 = 0 Bank 0 is selected
- RP0 = 1 Bank 1 is selected

**Note:** The IRP and RP1 bits STATUS<7:6> are reserved and should always be maintained as '0's.

#### 2.2.1 GENERAL PURPOSE REGISTER FILE

The register file is organized as 64 x 8 in the PIC12F629/675 devices. Each register is accessed, either directly or indirectly, through the File Select Register FSR (see Section 2.4).

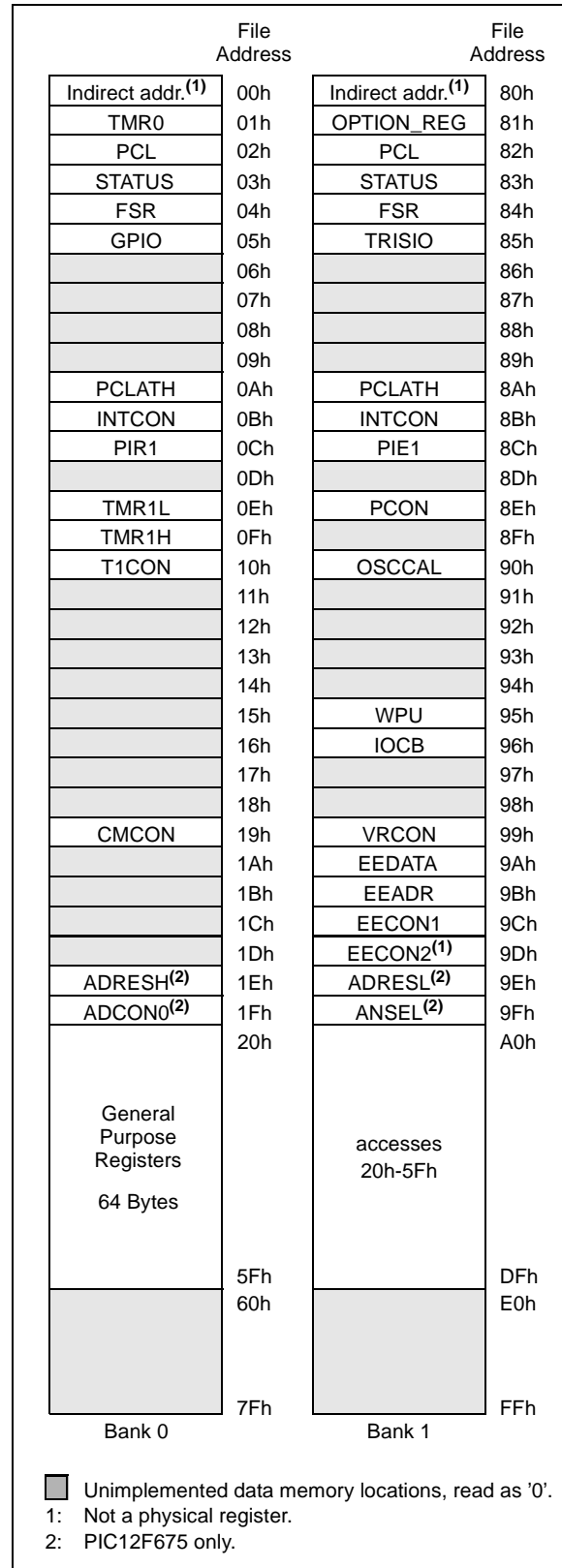
# PIC12F629/675

## 2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function registers are registers used by the CPU and peripheral functions for controlling the desired operation of the device (see Table 2-1). These registers are static RAM.

The special registers can be classified into two sets: core and peripheral. The Special Function registers associated with the “core” are described in this section. Those related to the operation of the peripheral features are described in the section of that peripheral feature.

**FIGURE 2-2: DATA MEMORY MAP OF THE PIC12F629/675**



**TABLE 2-1: SPECIAL FUNCTION REGISTERS SUMMARY**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR Reset	Page
<b>Bank 0</b>											
00h	INDF <sup>(1)</sup>	Addressing this Location uses Contents of FSR to Address Data Memory								0000 0000	18,59
01h	TMR0	Timer0 Module's Register								xxxx xxxx	25
02h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	17
03h	STATUS	IRP <sup>(2)</sup>	RP1 <sup>(2)</sup>	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	11
04h	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	18
05h	GPIO	—	—	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	--xx xxxx	19
06h	—	Unimplemented								—	—
07h	—	Unimplemented								—	—
08h	—	Unimplemented								—	—
09h	—	Unimplemented								—	—
0Ah	PCLATH	—	—	—	Write Buffer for Upper 5 bits of Program Counter				---	0000	17
0Bh	INTCON	GIE	PEIE	TOIE	INTE	GPIE	TOIF	INTF	GPIF	0000 0000	13
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	15
0Dh	—	Unimplemented								—	—
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit Timer1								xxxx xxxx	28
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit Timer1								xxxx xxxx	28
10h	T1CON	—	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	TMR1CS	TMR1ON	-000 0000	30
11h	—	Unimplemented								—	—
12h	—	Unimplemented								—	—
13h	—	Unimplemented								—	—
14h	—	Unimplemented								—	—
15h	—	Unimplemented								—	—
16h	—	Unimplemented								—	—
17h	—	Unimplemented								—	—
18h	—	Unimplemented								—	—
19h	CMCON	—	COU $\overline{T}$	—	CINV	CIS	CM2	CM1	CM0	-0-0 0000	33
1Ah	—	Unimplemented								—	—
1Bh	—	Unimplemented								—	—
1Ch	—	Unimplemented								—	—
1Dh	—	Unimplemented								—	—
1Eh	ADRESH <sup>(3)</sup>	Most Significant 8 bits of the Left Shifted A/D Result or 2 bits of the Right Shifted Result								xxxx xxxx	40
1Fh	ADCON0 <sup>(3)</sup>	ADFM	VCFG	—	—	CHS1	CHS0	GO/ $\overline{DONE}$	ADON	00-- 0000	41,59

Legend: — = unimplemented locations read as '0', u = unchanged, x = unknown,  $\alpha$  = value depends on condition, shaded = unimplemented

- Note 1:** This is not a physical register.  
**Note 2:** These bits are reserved and should always be maintained as '0'.  
**Note 3:** PIC12F675 only.

# PIC12F629/675

**TABLE 2-1: SPECIAL FUNCTION REGISTERS SUMMARY (CONTINUED)**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR Reset	Page	
<b>Bank 1</b>												
80h	INDF <sup>(1)</sup>	Addressing this Location uses Contents of FSR to Address Data Memory								0000 0000	18,59	
81h	OPTION_REG	GPPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	12,26	
82h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	17	
83h	STATUS	IRP <sup>(2)</sup>	RP1 <sup>(2)</sup>	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	11	
84h	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	18	
85h	TRISIO	—	—	TRIS5	TRIS4	TRIS3	TRIS2	TRIS1	TRIS0	--11 1111	19	
86h	—	Unimplemented								—	—	
87h	—	Unimplemented								—	—	
88h	—	Unimplemented								—	—	
89h	—	Unimplemented								—	—	
8Ah	PCLATH	—	—	—	Write Buffer for Upper 5 bits of Program Counter				---	0 0000	17	
8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	13	
8Ch	PIE1	EEIE	ADIE	—	—	CMIE	—	—	TMR1IE	00-- 0--0	14	
8Dh	—	Unimplemented								—	—	
8Eh	PCON	—	—	—	—	—	—	POR	BOD	---- --0x	16	
8Fh	—	Unimplemented								—	—	
90h	OSCCAL	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	—	1000 00--	16	
91h	—	Unimplemented								—	—	
92h	—	Unimplemented								—	—	
93h	—	Unimplemented								—	—	
94h	—	Unimplemented								—	—	
95h	WPU	—	—	WPU5	WPU4	—	WPU2	WPU1	WPU0	--11 1111	19	
96h	IOCB	—	—	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0	--00 0000	20	
97h	—	Unimplemented								—	—	
98h	—	Unimplemented								—	—	
99h	VRCON	VREN	—	VRR	—	VR3	VR2	VR1	VR0	0-0- 0000	38	
9Ah	EEDATA	Data EEPROM Data Register								0000 0000	47	
9Bh	EEADR	—	Data EEPROM Address Register								-000 0000	47
9Ch	EECON1	—	—	—	—	WRERR	WREN	WR	RD	---- x000	48	
9Dh	EECON2 <sup>(1)</sup>	EEPROM Control Register 2								---- ----	48	
9Eh	ADRESL <sup>(3)</sup>	Least Significant 2 bits of the Left Shifted A/D Result of 8 bits or the Right Shifted Result								xxxx xxxx	40	
9Fh	ANSEL <sup>(3)</sup>	—	ADCS2	ADCS1	ADCS0	ANS3	ANS2	ANS1	ANS0	-000 1111	42,59	

Legend: — = unimplemented locations read as '0', u = unchanged, x = unknown,  $\bar{c}$  = value depends on condition, shaded = unimplemented

- Note 1:** This is not a physical register.  
**Note 2:** These bits are reserved and should always be maintained as '0'.  
**Note 3:** PIC12F675 only.

## 2.2.2.1 STATUS Register

The STATUS register, shown in Register 2-1, contains:

- the arithmetic status of the ALU
- the RESET status
- the bank select bits for data memory (SRAM)

The STATUS register can be the destination for any instruction, like 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).

It is recommended, therefore, that only `BCF`, `BSF`, `SWAPF` and `MOVWF` instructions are used to alter the STATUS register, because these instructions do not affect any STATUS bits. For other instructions not affecting any STATUS bits, see the "Instruction Set Summary".

**Note 1:** Bits IRP and RP1 (STATUS<7:6>) are not used by the PIC12F629/675 and should be maintained as clear. Use of these bits is not recommended, since this may affect upward compatibility with future products.

**2:** The C and DC bits operate as a Borrow and Digit Borrow out bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

### REGISTER 2-1: STATUS — STATUS REGISTER (ADDRESS: 03h OR 83h)

Reserved	Reserved	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x	
IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	
bit 7								bit 0

- bit 7 **IRP:** This bit is reserved and should be maintained as '0'
- bit 6 **RP1:** This bit is reserved and should be maintained as '0'
- bit 5 **RP0:** Register Bank Select bit (used for direct addressing)  
 0 = Bank 0 (00h - 7Fh)  
 1 = Bank 1 (80h - FFh)
- 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/borrow bit (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions)  
 For borrow, the polarity is reversed.  
 1 = A carry-out from the 4th low order bit of the result occurred  
 0 = No carry-out from the 4th low order bit of the result
- bit 0 **C:** Carry/borrow bit (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions)  
 1 = A carry-out from the Most Significant bit of the result occurred  
 0 = No carry-out from the Most Significant bit of the result occurred

**Note:** For borrow the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (`RRF`, `RLF`) instructions, this bit is loaded with either the high or low order bit of the source register

**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

# PIC12F629/675

## 2.2.2.2 OPTION Register

The OPTION register is a readable and writable register, which contains various control bits to configure:

- TMR0/WDT prescaler
- External GP2/INT interrupt
- TMR0
- Weak pull-ups on GPIO

**Note:** To achieve a 1:1 prescaler assignment for TMR0, assign the prescaler to the WDT by setting PSA bit to '1' (OPTION<3>). See Section 4.4.

### REGISTER 2-2: OPTION\_REG — OPTION REGISTER (ADDRESS: 81h)

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

- bit 7  **$\overline{\text{GPPU}}$** : GPIO Pull-up Enable bit  
 1 = GPIO pull-ups are disabled  
 0 = GPIO pull-ups are enabled by individual port latch values
- bit 6 **INTEDG**: Interrupt Edge Select bit  
 1 = Interrupt on rising edge of GP2/INT pin  
 0 = Interrupt on falling edge of GP2/INT pin
- bit 5 **T0CS**: TMR0 Clock Source Select bit  
 1 = Transition on GP2/T0CKI pin  
 0 = Internal instruction cycle clock (CLKOUT)
- bit 4 **T0SE**: TMR0 Source Edge Select bit  
 1 = Increment on high-to-low transition on GP2/T0CKI pin  
 0 = Increment on low-to-high transition on GP2/T0CKI pin
- bit 3 **PSA**: Prescaler Assignment bit  
 1 = Prescaler is assigned to the WDT  
 0 = Prescaler is assigned to the TIMER0 module
- bit 2-0 **PS2:PS0**: Prescaler Rate Select bits

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

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

## 2.2.2.3 INTCON Register

The INTCON register is a readable and writable register, which contains the various enable and flag bits for TMR0 register overflow, GPIO port change and external GP2/INT pin interrupts.

**Note:** Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

### REGISTER 2-3: INTCON — INTERRUPT CONTROL REGISTER (ADDRESS: 0Bh OR 8Bh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
GIE	PEIE	TOIE	INTE	GPIE	T0IF	INTF	GPIF	
bit 7								bit 0

- bit 7      **GIE:** Global Interrupt Enable bit  
1 = Enables all unmasked interrupts  
0 = Disables all interrupts
- bit 6      **PEIE:** Peripheral Interrupt Enable bit  
1 = Enables all unmasked peripheral interrupts  
0 = Disables all peripheral interrupts
- bit 5      **TOIE:** TMR0 Overflow Interrupt Enable bit  
1 = Enables the TMR0 interrupt  
0 = Disables the TMR0 interrupt
- bit 4      **INTE:** GP2/INT External Interrupt Enable bit  
1 = Enables the GP2/INT external interrupt  
0 = Disables the GP2/INT external interrupt
- bit 3      **GPIE:** Port Change Interrupt Enable bit  
1 = Enables the GPIO port change interrupt  
0 = Disables the GPIO port change interrupt
- bit 2      **T0IF:** TMR0 Overflow Interrupt Flag bit<sup>(1)</sup>  
1 = TMR0 register has overflowed (must be cleared in software)  
0 = TMR0 register did not overflow
- bit 1      **INTF:** GP2/INT External Interrupt Flag bit  
1 = The GP2/INT external interrupt occurred (must be cleared in software)  
0 = The GP2/INT external interrupt did not occur
- bit 0      **GPIF:** Port Change Interrupt Flag bit  
1 = When at least one of the GP5:GP0 pins changed state (must be cleared in software)  
0 = None of the GP5:GP0 pins have changed state

**Note 1:** T0IF bit is set when TIMER0 rolls over. TIMER0 is unchanged on RESET and should be initialized before clearing T0IF bit.

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

# PIC12F629/675

## 2.2.2.4 PIE1 Register

The PIE1 register contains the interrupt enable bits, as shown in Register 2-4.

**Note:** Bit PEIE (INTCON<6>) must be set to enable any peripheral interrupt.

### REGISTER 2-4: PIE1 — PERIPHERAL INTERRUPT ENABLE REGISTER 1 (ADDRESS: 8Ch)

R/W-0	R/W-0	U-0	U-0	R/W-0	U-0	U-0	R/W-0
EEIE	ADIE	—	—	CMIE	—	—	TMR1IE
bit 7							bit 0

- bit 7 **EEIE:** EE Write Complete Interrupt Enable bit  
1 = Enables the EE write complete interrupt  
0 = Disables the EE write complete interrupt
- bit 6 **ADIE:** A/D Converter Interrupt Enable bit (PIC12F675 only)  
1 = Enables the A/D converter interrupt  
0 = Disables the A/D converter interrupt
- bit 5-4 **Unimplemented:** Read as '0'
- bit 3 **CMIE:** Comparator Interrupt Enable bit  
1 = Enables the comparator interrupt  
0 = Disables the comparator interrupt
- bit 2-1 **Unimplemented:** Read as '0'
- bit 0 **TMR1IE:** TMR1 Overflow Interrupt Enable bit  
1 = Enables the TMR1 overflow interrupt  
0 = Disables the TMR1 overflow interrupt

**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



## 2.2.2.5 PIR1 Register

The PIR1 register contains the interrupt flag bits, as shown in Register 2-5.

**Note:** Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

### REGISTER 2-5: PIR1 — PERIPHERAL INTERRUPT REGISTER 1 (ADDRESS: 0Ch)

R/W-0	R/W-0	U-0	U-0	R/W-0	U-0	U-0	R/W-0
EEIF	ADIF	—	—	CMIF	—	—	TMR1IF
bit 7				bit 0			

- bit 7     **EEIF:** EEPROM Write Operation Interrupt Flag bit  
           1 = The write operation completed (must be cleared in software)  
           0 = The write operation has not completed or has not been started
- bit 6     **ADIF:** A/D Converter Interrupt Flag bit (PIC12F675 only)  
           1 = The A/D conversion is complete (must be cleared in software)  
           0 = The A/D conversion is not complete
- bit 5-4   **Unimplemented:** Read as '0'
- bit 3     **CMIF:** Comparator Interrupt Flag bit  
           1 = Comparator input has changed (must be cleared in software)  
           0 = Comparator input has not changed
- bit 2-1   **Unimplemented:** Read as '0'
- bit 0     **TMR1IF:** TMR1 Overflow Interrupt Flag bit  
           1 = TMR1 register overflowed (must be cleared in software)  
           0 = TMR1 register did not overflow

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

# PIC12F629/675

## 2.2.2.6 PCON Register

The Power Control (PCON) register contains flag bits to differentiate between a:

- Power-on Reset (POR)
- Brown-out Reset (BOR)
- Watchdog Timer Reset (WDT)
- External MCLR Reset

The PCON Register bits are shown in Register 2-6.

### REGISTER 2-6: PCON — POWER CONTROL REGISTER (ADDRESS: 8Eh)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-x
—	—	—	—	—	—	POR	BOD
bit 7						bit 0	

bit 7-2 **Unimplemented:** Read as '0'

bit 1 **POR:** Power-on Reset Status bit

1 = No Power-on Reset occurred

0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0 **BOD:** Brown-out Detect Status bit

1 = No Brown-out Reset occurred

0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

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

## 2.2.2.7 OSCCAL Register

The Oscillator Calibration register (OSCCAL) is used to calibrate the internal 4 MHz oscillator. It contains 6 bits to adjust the frequency up or down to achieve 4 MHz.

The OSCCAL register bits are shown in Register 2-7.

### REGISTER 2-7: OSCCAL — OSCILLATOR CALIBRATION REGISTER (ADDRESS: 90h)

R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	—
bit 7						bit 0	

bit 7-2 **CAL5:CAL0:** 6-bit Signed Oscillator Calibration bits

111111 = Maximum frequency

100000 = Center frequency

000000 = Minimum frequency

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

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

- n = Value at POR

'1' = Bit is set

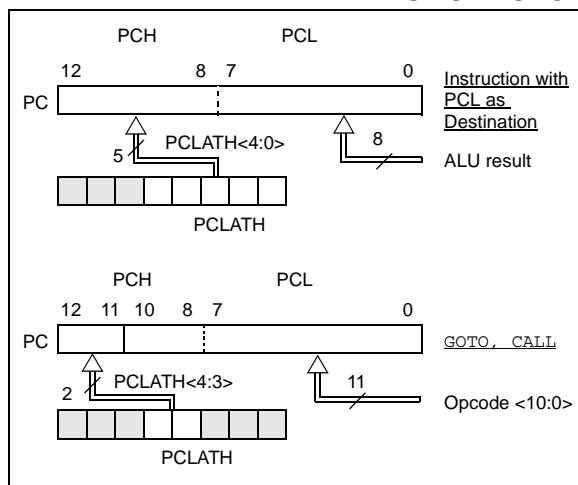
'0' = Bit is cleared

x = Bit is unknown

## 2.3 PCL and PCLATH

The program counter (PC) is 13-bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<12:8>) is not directly readable or writable and comes from PCLATH. On any RESET, the PC is cleared. Figure 2-3 shows the two situations for the loading of the PC. The upper example in Figure 2-3 shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in Figure 2-3 shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

**FIGURE 2-3: LOADING OF PC IN DIFFERENT SITUATIONS**



### 2.3.1 COMPUTED GOTO

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

### 2.3.2 STACK

The PIC12F629/675 family has an 8 level deep x 13-bit wide hardware stack (see Figure 2-1). The stack space is not part of either program or data space and the stack pointer is not readable or writable. The PC is PUSHed onto the stack when a CALL instruction is executed, or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or POP operation.

The stack operates as a circular buffer. This means that after the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

**Note 1:** There are no STATUS bits to indicate stack overflow or stack underflow conditions.

**2:** There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, RETURN, RETLW and RETFIE instructions, or the vectoring to an interrupt address.

# PIC12F629/675

## 2.4 Indirect Addressing, INDF and FSR Registers

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

Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no operation (although STATUS bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in Figure 2-4.

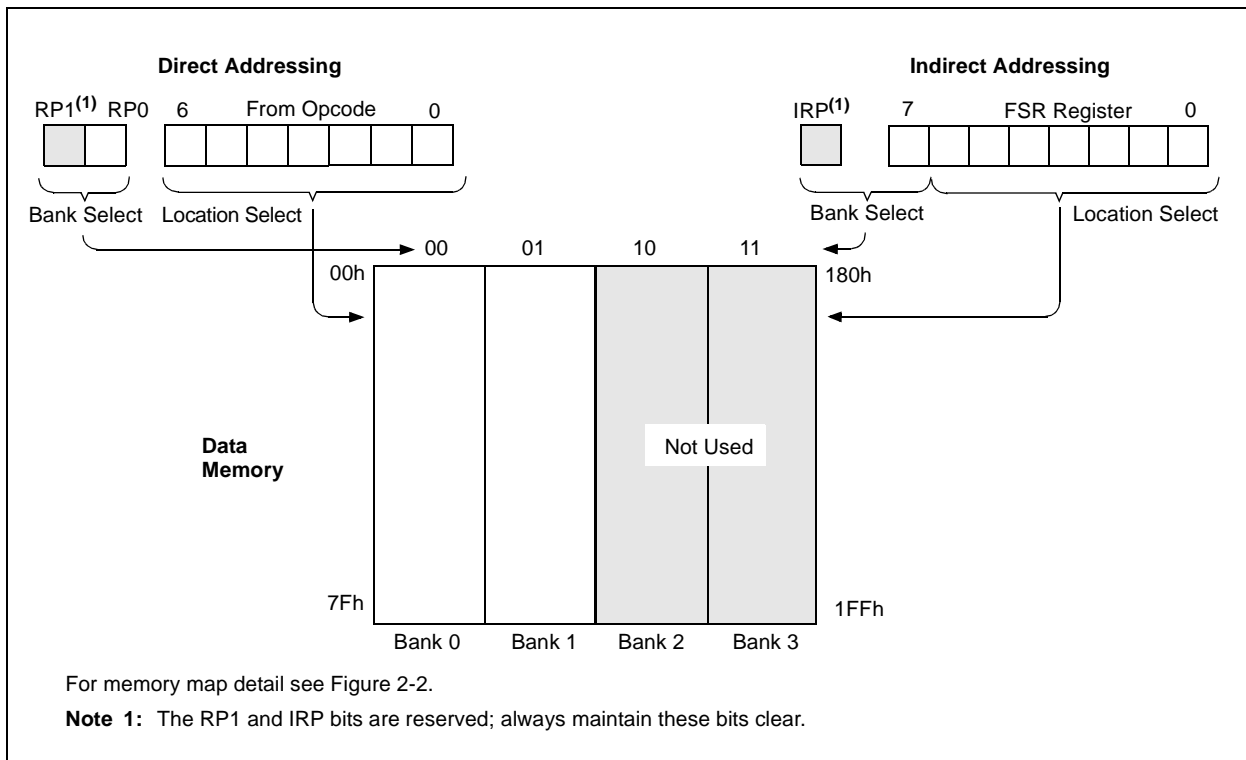
A simple program to clear RAM location 20h-2Fh using indirect addressing is shown in Example 2-1.

### EXAMPLE 2-1: INDIRECT ADDRESSING

```

movlw 0x20 ;initialize pointer
movwf FSR ;to RAM
NEXT   clrf INDF ;clear INDF register
       incf FSR ;inc pointer
       btfss FSR,4 ;all done?
       goto NEXT ;no clear next
CONTINUE ;yes continue
    
```

FIGURE 2-4: DIRECT/INDIRECT ADDRESSING PIC12F629/675



## 3.0 GPIO PORT

There are as many as six general purpose I/O pins available. Depending on which peripherals are enabled, some or all of the pins may not be available as general purpose I/O. In general, when a peripheral is enabled, the associated pin may not be used as a general purpose I/O pin.

**Note:** Additional information on I/O ports may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023)

### 3.1 GPIO and the TRISIO Registers

GPIO is an 6-bit wide, bi-directional port. The corresponding data direction register is TRISIO. Setting a TRISIO bit (= 1) will make the corresponding GPIO pin an input (i.e., put the corresponding output driver in a Hi-Impedance mode). Clearing a TRISIO bit (= 0) will make the corresponding GPIO pin an output (i.e., put the contents of the output latch on the selected pin). The exception is GP3, which is input only and its TRIS bit will always read as '1'. Example 3-1 shows how to initialize GPIO.

Reading the GPIO register reads the status of the pins, whereas writing to it will write to the port latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified, and then written to the port data latch. GP3 reads '0' when MCLREN = 1.

The TRISIO register controls the direction of the GP pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISIO register are maintained set when using them as analog inputs.

#### EXAMPLE 3-1: INITIALIZING GPIO

```
bcf    STATUS,RP0    ;Bank 0
clrf   GPIO          ;Init GPIO
movlw  07h           ;Set GP<2:0> to
movwf  CMCON         ;digital IO
bsf    STATUS,RP0    ;Bank 1
movlw  0Ch           ;Set GP<3:2> as inputs
movwf  TRISIO        ;and set GP<5:4,1:0>
                           ;as outputs
bcf    STATUS,RP0    ;Bank 0
```

### 3.2 Additional Pin Functions

Every GPIO pin on the PIC12F629/675 has an interrupt-on-change option and every GPIO pin, except GP3, has a weak pull-up option. The next two sections describe these functions.

#### 3.2.1 WEAK PULL-UP

Each of the GPIO pins, except GP3, has an individually configurable weak internal pull-up. Control bits WPUx enable or disable each pull-up. Refer to Register 3-1. Each weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset by the GPPU bit (OPTION<7>).

#### REGISTER 3-1: WPU — WEAK PULL-UP REGISTER (ADDRESS: 95h)

U-0	U-0	R/W-1	R/W-1	U-0	R/W-1	R/W-1	R/W-1
—	—	WPU5	WPU4	—	WPU2	WPU1	WPU0
bit 7				bit 0			

- bit 7-6     **Unimplemented:** Read as '0'
- bit 5-4     **WPU<5:4>:** Weak Pull-up Register bit  
1 = Pull-up enabled  
0 = Pull-up disabled
- bit 3       **Unimplemented:** Read as '0'
- bit 2-0     **WPU<2:0>:** Weak Pull-up Register bit  
1 = Pull-up enabled  
0 = Pull-up disabled

- Note 1:** Global  $\overline{\text{GPPU}}$  must be enabled for individual pull-ups to be enabled.  
**Note 2:** The weak pull-up device is automatically disabled if the pin is in Output mode (TRISIO = 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

# PIC12F629/675

## 3.2.2 INTERRUPT-ON-CHANGE

Each of the GPIO pins is individually configurable as an interrupt-on-change pin. Control bits IOCBx enable or disable the interrupt function for each pin. Refer to Register 3-2. The interrupt-on-change is disabled on a Power-on Reset.

For enabled interrupt-on-change pins, the values are compared with the old value latched on the last read of GPIO. The 'mismatch' outputs of the last read are OR'd together to set, or clear, the GP Port Change Interrupt flag bit (GPIF) in the INTCON register.

This interrupt can wake the device from SLEEP. The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

- a) Any read or write of GPIO. This will end the mismatch condition.
- b) Clear the flag bit GPIF.

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

**Note:** If a change on the I/O pin should occur when the read operation is being executed (start of the Q2 cycle), then the GPIF interrupt flag may not get set.

### REGISTER 3-2: IOCB — INTERRUPT-ON-CHANGE GPIO REGISTER (ADDRESS: 96h)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
—	—	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0	
bit 7								bit 0

- bit 7-6     **Unimplemented:** Read as '0'
- bit 5-0     **IOCB<5:0>:** Interrupt-on-Change GPIO Control bit
  - 1 = Interrupt-on-change enabled
  - 0 = Interrupt-on-change disabled

**Note 1:** Global interrupt enables (GIE and GPIE) must be enabled for individual interrupts to be recognized.

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

## 3.3 Pin Descriptions and Diagrams

Each GPIO pin is multiplexed with other functions. The pins and their combined functions are briefly described here. For specific information about individual functions such as the comparator or the A/D, refer to the appropriate section in this Data Sheet.

### 3.3.1 GP0/AN0/CIN+

Figure 3-1 shows the diagram for this pin. The GP0 pin is configurable to function as one of the following:

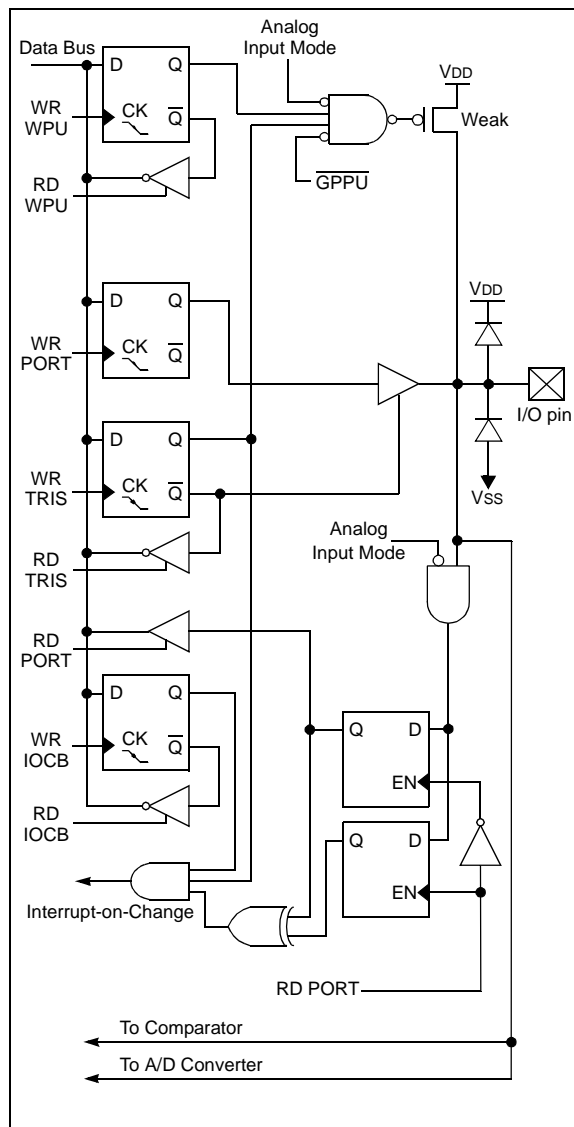
- a general purpose I/O
- an analog input for the A/D (PIC12F675 only)
- an analog input to the comparator

### 3.3.2 GP1/AN1/CIN-/VREF

Figure 3-1 shows the diagram for this pin. The GP1 pin is configurable to function as one of the following:

- as a general purpose I/O
- an analog input for the A/D (PIC12F675 only)
- an analog input to the comparator
- a voltage reference input for the A/D (PIC12F675 only)

**FIGURE 3-1: BLOCK DIAGRAM OF GP0 AND GP1 PINS**



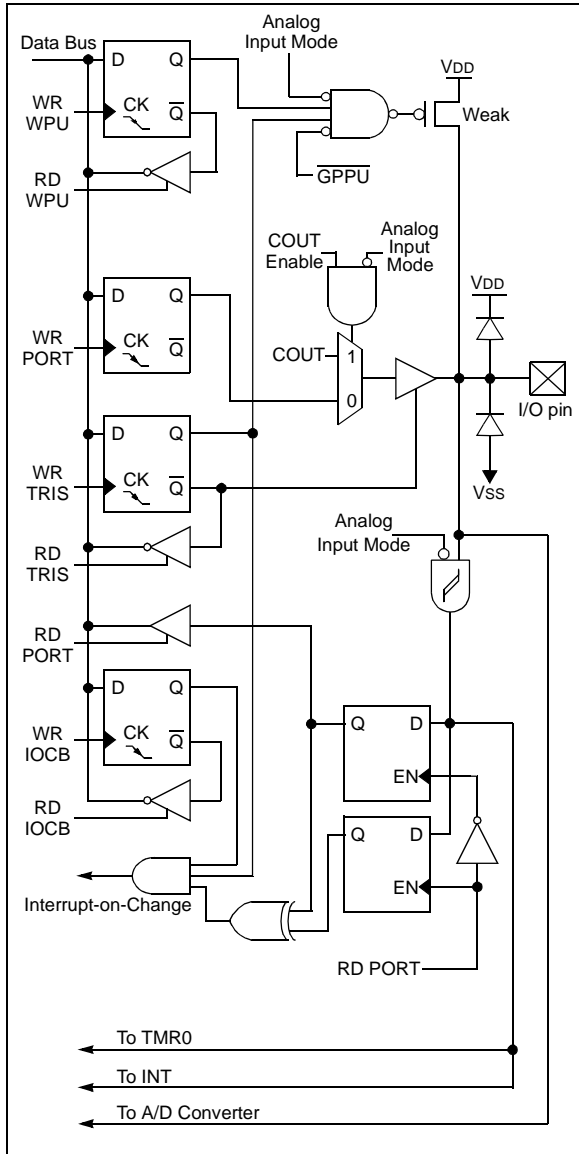
# PIC12F629/675

## 3.3.3 GP2/AN2/T0CKI/INT/COU

Figure 3-2 shows the diagram for this pin. The GP2 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the A/D (PIC12F675 only)
- a digital output from the comparator
- the clock input for TMR0
- an external edge triggered interrupt

**FIGURE 3-2: BLOCK DIAGRAM OF GP2**

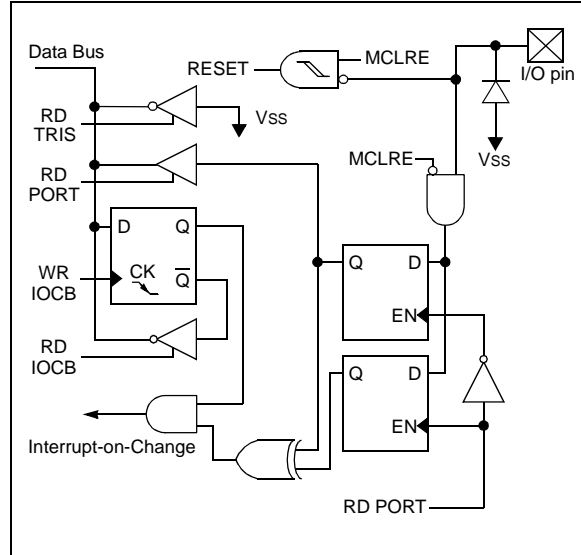


## 3.3.4 GP3/MCLR/VPP

Figure 3-3 shows the diagram for this pin. The GP3 pin is configurable to function as one of the following:

- a general purpose input
- as Master Clear Reset

**FIGURE 3-3: BLOCK DIAGRAM OF GP3**



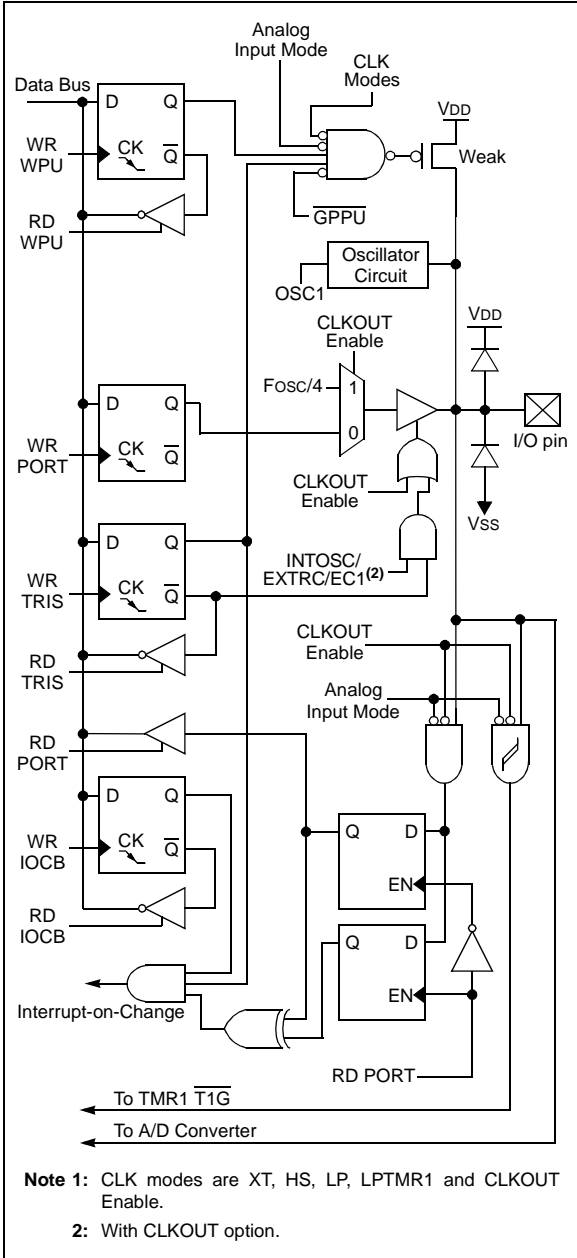


### 3.3.5 GP4/AN3/T1G/OSC2/CLKOUT

Figure 3-4 shows the diagram for this pin. The GP4 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the A/D (PIC12F675 only)
- a TMR1 gate input
- a crystal/resonator connection
- a clock output

**FIGURE 3-4: BLOCK DIAGRAM OF GP4**

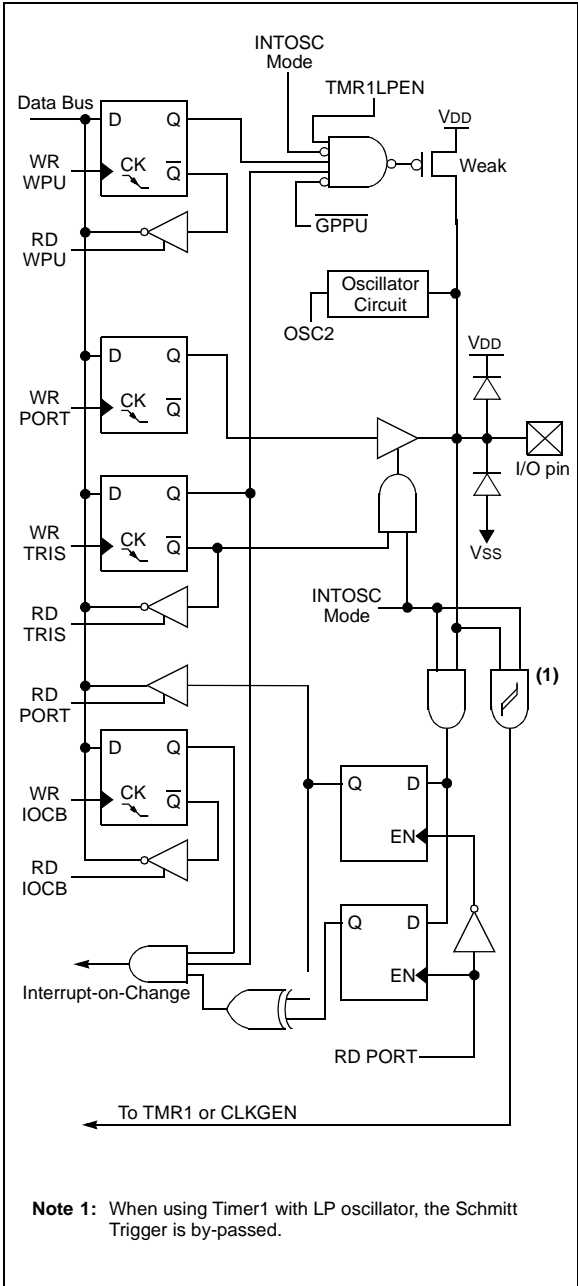


### 3.3.6 GP5/T1CKI/OSC1/CLKIN

Figure 3-5 shows the diagram for this pin. The GP5 pin is configurable to function as one of the following:

- a general purpose I/O
- a TMR1 clock input
- a crystal/resonator connection
- a clock input

**FIGURE 3-5: BLOCK DIAGRAM OF GP5**



# PIC12F629/675

**TABLE 3-1: SUMMARY OF REGISTERS ASSOCIATED WITH GPIO**

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
05h	GPIO	—	—	GP5	GP4	GP3	GP2	GP1	GP0	--xx xxxx	--uu uuuu
0Bh/8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	0000 000u
19h	CMCON	—	COUT	—	CINV	CIS	CM2	CM1	CM0	-0-0 0000	-0-0 0000
81h	OPTION_REG	$\overline{\text{GPPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
85h	TRISIO	—	—	TRIS5	TRIS4	TRIS3	TRIS2	TRIS1	TRIS0	--11 1111	--11 1111
95h	WPU	—	—	WPU5	WPU4	—	WPU2	WPU1	WPU0	--11 -111	--11 -111
96h	IOCB	—	—	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0	--00 0000	--00 0000
9Fh	ANSEL	—	ADCS2	ADCS1	ADCS0	ANS3	ANS2	ANS1	ANS0	-000 1111	-000 1111

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

## 4.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:

- 8-bit timer/counter
- Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select
- Interrupt on overflow from FFh to 00h
- Edge select for external clock

Figure 4-1 is a block diagram of the Timer0 module and the prescaler shared with the WDT.

**Note:** Additional information on the Timer0 module is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

### 4.1 Timer0 Operation

Timer mode is selected by clearing the T0CS bit (OPTION\_REG<5>). In Timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If TMR0 is written, the increment is inhibited for the following two instruction cycles. The user can work around this by writing an adjusted value to the TMR0 register.

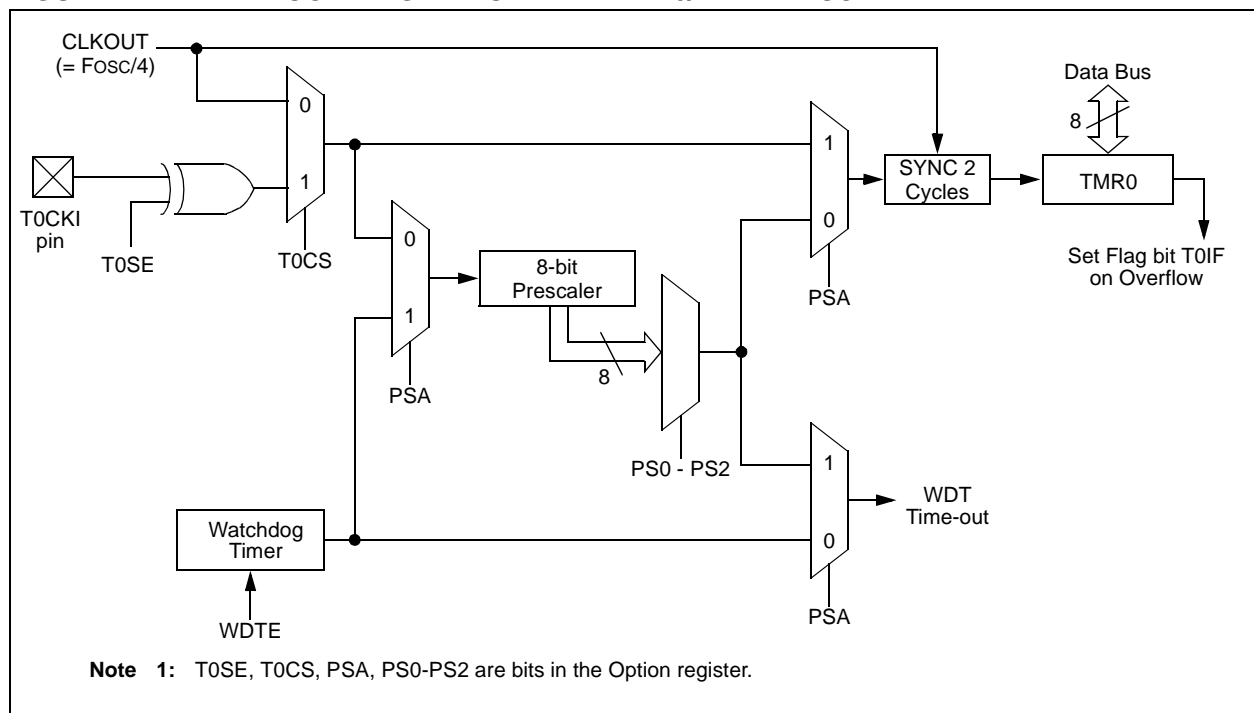
Counter mode is selected by setting the T0CS bit (OPTION\_REG<5>). In this mode, the Timer0 module will increment either on every rising or falling edge of pin GP2/T0CKI. The incrementing edge is determined by the source edge (T0SE) control bit (OPTION\_REG<4>). Clearing the T0SE bit selects the rising edge.

**Note:** Counter mode has specific external clock requirements. Additional information on these requirements is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

### 4.2 Timer0 Interrupt

A Timer0 interrupt is generated when the TMR0 register timer/counter overflows from FFh to 00h. This overflow sets the T0IF bit. The interrupt can be masked by clearing the T0IE bit (INTCON<5>). The T0IF bit (INTCON<2>) must be cleared in software by the Timer0 module Interrupt Service Routine before re-enabling this interrupt. The Timer0 interrupt cannot wake the processor from SLEEP since the timer is shut-off during SLEEP.

**FIGURE 4-1: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER**



# PIC12F629/675

## 4.3 Using Timer0 with an External Clock

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 accom-

plished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks. Therefore, it is necessary for T0CKI to be high for at least 2Tosc (and a small RC delay of 20 ns) and low for at least 2Tosc (and a small RC delay of 20 ns). Refer to the electrical specification of the desired device.

### REGISTER 4-1: OPTION\_REG — OPTION REGISTER (ADDRESS: 81h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
GPPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0

bit 7

bit 0

- bit 7 **GPPU:** GPIO Pull-up Enable bit  
1 = GPIO pull-ups are disabled  
0 = GPIO pull-ups are enabled by individual port latch values
- bit 6 **INTEDG:** Interrupt Edge Select bit  
1 = Interrupt on rising edge of GP2/INT pin  
0 = Interrupt on falling edge of GP2/INT pin
- bit 5 **T0CS:** TMR0 Clock Source Select bit  
1 = Transition on GP2/T0CKI pin  
0 = Internal instruction cycle clock (CLKOUT)
- bit 4 **T0SE:** TMR0 Source Edge Select bit  
1 = Increment on high-to-low transition on GP2/T0CKI pin  
0 = Increment on low-to-high transition on GP2/T0CKI pin
- bit 3 **PSA:** Prescaler Assignment bit  
1 = Prescaler is assigned to the WDT  
0 = Prescaler is assigned to the TIMER0 module
- bit 2-0 **PS2:PS0:** Prescaler Rate Select bits

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

**Legend:**

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

## 4.4 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module, or as a postscaler for the Watchdog Timer. For simplicity, this counter will be referred to as “prescaler” throughout this Data Sheet. The prescaler assignment is controlled in software by the control bit PSA (OPTION\_REG<3>). Clearing the PSA bit will assign the prescaler to Timer0. Prescale values are selectable via the PS2:PS0 bits (OPTION\_REG<2:0>).

The prescaler is not readable or writable. When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g., CLRF 1, MOVWF 1, BSF 1, x...etc.) will clear the prescaler. When assigned to WDT, a CLRWDT instruction will clear the prescaler along with the Watchdog Timer.

### 4.4.1 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 4-1) must be executed when changing the prescaler assignment from Timer0 to WDT.

### EXAMPLE 4-1: CHANGING PRESCALER (TIMER0→WDT)

```

bcf    STATUS,RP0    ;Bank 0
clrwdt                ;Clear WDT
clrf   TMR0          ;Clear TMR0 and
                        ; prescaler
bsf    STATUS,RP0    ;Bank 1

movlw  b'00101111'   ;Required if desired
movwf  OPTION_REG    ; PS2:PS0 is
clrwdt                ; 000 or 001
                        ;
movlw  b'00101xxx'   ;Set postscaler to
movwf  OPTION_REG    ; desired WDT rate
bcf    STATUS,RP0    ;Bank 0
    
```

To change prescaler from the WDT to the TMR0 module, use the sequence shown in Example 4-2. This precaution must be taken even if the WDT is disabled.

### EXAMPLE 4-2: CHANGING PRESCALER (WDT→TIMER0)

```

clrwdt                ;Clear WDT and
                        ; postscaler
bsf    STATUS,RP0    ;Bank 1

movlw  b'xxxx0xxx'   ;Select TMR0,
                        ; prescale, and
                        ; clock source
movwf  OPTION_REG    ;
bcf    STATUS,RP0    ;Bank 0
    
```

**TABLE 4-1: REGISTERS ASSOCIATED WITH TIMER0**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on all other RESETS
01h	TMR0	Timer0 Module Register								xxxx xxxx	uuuu uuuu
0Bh/8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	0000 000u
81h	OPTION_REG	$\overline{\text{GPPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
85h	TRISIO	—	—	TRIS5	TRIS4	TRIS3	TRIS2	TRIS1	TRIS0	--11 1111	--11 1111

Legend: — = Unimplemented locations, read as '0', u = unchanged, x = unknown.  
Shaded cells are not used by the Timer0 module.

# PIC12F629/675

## 5.0 TIMER1 MODULE WITH GATE CONTROL

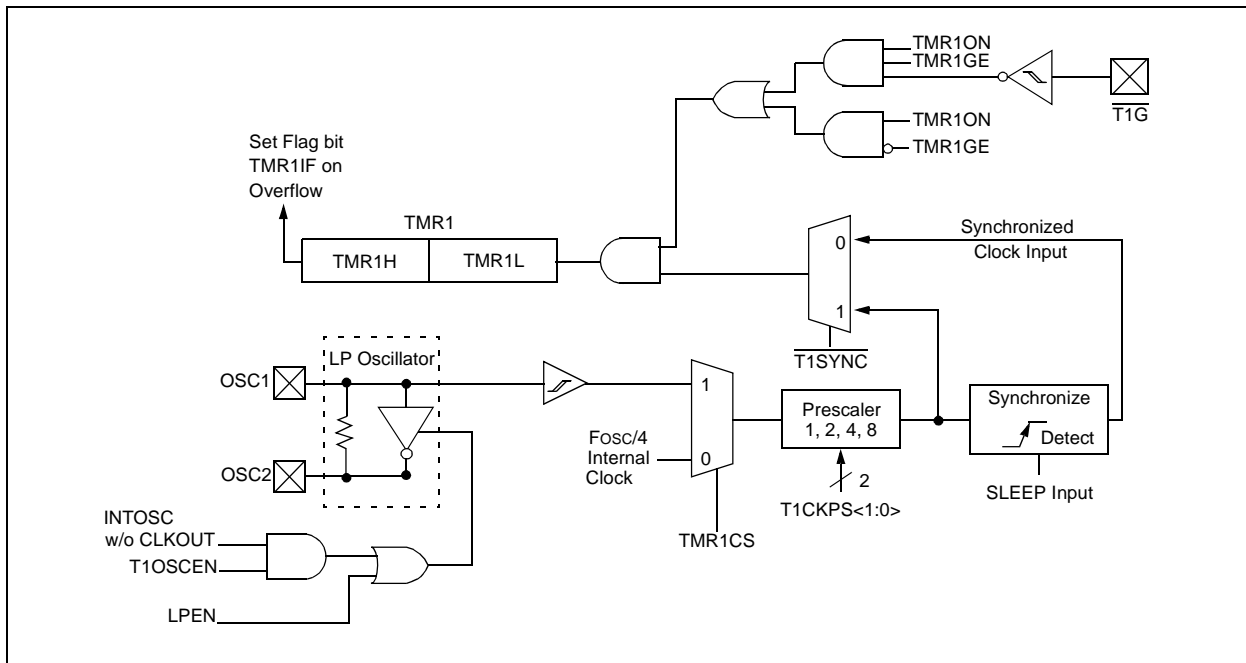
The PIC12F629/675 devices have a 16-bit timer. Figure 5-1 shows the basic block diagram of the Timer1 module. Timer1 has the following features:

- 16-bit timer/counter (TMR1H:TMR1L)
- Readable and writable
- Internal or external clock selection
- Synchronous or asynchronous operation
- Interrupt on overflow from FFFFh to 0000h
- Wake-up upon overflow (Asynchronous mode)
- Optional external enable input ( $\overline{T1G}$ )
- Optional LP oscillator

The Timer1 Control register (T1CON), shown in Register 5-1, is used to enable/disable Timer1 and select the various features of the Timer1 module.

**Note:** Additional information on timer modules is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

**FIGURE 5-1: TIMER1 BLOCK DIAGRAM**



## 5.1 Timer1 Modes of Operation

Timer1 can operate in one of three modes:

- 16-bit timer with prescaler
- 16-bit synchronous counter
- 16-bit asynchronous counter

In Timer mode, Timer1 is incremented on every instruction cycle. In Counter mode, Timer1 is incremented on the rising edge of the external clock input T1CKI. In addition, the Counter mode clock can be synchronized to the microcontroller system clock or run asynchronously.

In Counter and Timer modules, the counter/timer clock can be gated by the  $\overline{T1G}$  input.

If an external clock oscillator is needed (and the microcontroller is using the INTRC w/o CLKOUT), Timer1 can use the LP oscillator as a clock source.

**Note:** In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge.

## 5.2 Timer1 Interrupt

The Timer1 register pair (TMR1H:TMR1L) increments to FFFFh and rolls over to 0000h. When Timer1 rolls over, the Timer1 interrupt flag bit (PIR1<0>) is set. To enable the interrupt on rollover, you must set these bits:

- Timer1 interrupt Enable bit (PIE1<0>)
- PEIE bit (INTCON<6>)
- GIE bit (INTCON<7>).

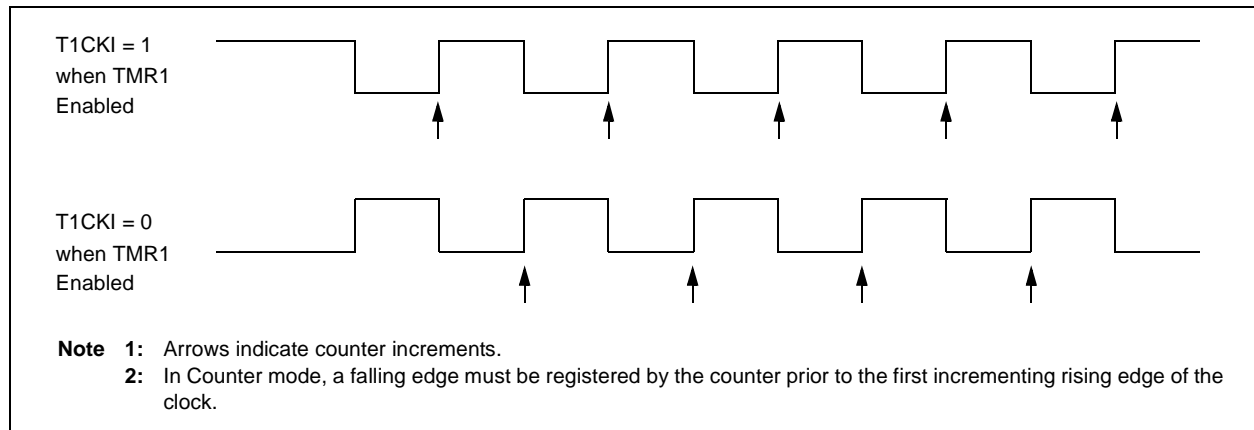
The interrupt is cleared by clearing the TMR1IF in the Interrupt Service Routine.

**Note:** The TMR1H:TMR1L register pair and the TMR1IF bit should be cleared before enabling interrupts.

## 5.3 Timer1 Prescaler

Timer1 has four prescaler options allowing 1, 2, 4, or 8 divisions of the clock input. The T1CKPS bits (T1CON<5:4>) control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMR1H or TMR1L.

**FIGURE 5-2: TIMER1 INCREMENTING EDGE**



# PIC12F629/675

## REGISTER 5-1: T1CON — TIMER1 CONTROL REGISTER (ADDRESS: 10h)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	TMR1CS	TMR1ON
bit 7							bit 0

- bit 7 **Unimplemented:** Read as '0'
- bit 6 **TMR1GE:** Timer1 Gate Enable bit  
If TMR1ON = 0:  
 This bit is ignored  
If TMR1ON = 1:  
 1 = Timer1 is on if  $\overline{T1G}$  pin is low  
 0 = Timer1 is on
- bit 5-4 **T1CKPS1:T1CKPS0:** Timer1 Input Clock Prescale Select bits  
 11 = 1:8 Prescale Value  
 10 = 1:4 Prescale Value  
 01 = 1:2 Prescale Value  
 00 = 1:1 Prescale Value
- bit 3 **T1OSCEN:** LP Oscillator Enable Control bit  
If INTOSC without CLKOUT oscillator is active:  
 1 = LP oscillator is enabled for Timer1 clock  
 0 = LP oscillator is off  
Else:  
 This bit is ignored
- bit 2 **T1SYNC:** Timer1 External Clock Input Synchronization Control bit  
TMR1CS = 1:  
 1 = Do not synchronize external clock input  
 0 = Synchronize external clock input  
TMR1CS = 0:  
 This bit is ignored. Timer1 uses the internal clock.
- bit 1 **TMR1CS:** Timer1 Clock Source Select bit  
 1 = External clock from T1OSO/T1CKI pin (on the rising edge)  
 0 = Internal clock ( $F_{osc}/4$ )
- bit 0 **TMR1ON:** Timer1 On bit  
 1 = Enables Timer1  
 0 = Stops Timer1

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



## 5.4 Timer1 Operation in Asynchronous Counter Mode

If control bit  $\overline{T1SYNC}$  (T1CON<2>) is set, the external clock input is not synchronized. The timer continues to increment asynchronous to the internal phase clocks. The timer will continue to run during SLEEP and can generate an interrupt on overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer (Section 5.4.1).

### 5.4.1 READING AND WRITING TIMER1 IN ASYNCHRONOUS COUNTER MODE

Reading TMR1H or TMR1L, while the timer is running from an external asynchronous clock, will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself, poses certain problems, since the timer may overflow between the reads.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the timer register.

Reading the 16-bit value requires some care. Examples 12-2 and 12-3 in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023) show how to read and write Timer1 when it is running in Asynchronous mode.

## 5.5 Timer1 Oscillator

A crystal oscillator circuit is built-in between pins OSC1 (input) and OSC2 (amplifier output). It is enabled by setting control bit T1OSCEN (T1CON<3>). The oscillator is a low power oscillator rated up to 200 kHz. It will continue to run during SLEEP. It is primarily intended for a 32 kHz crystal. Table 5-1 shows the capacitor selection for the Timer1 oscillator.

The Timer1 oscillator is shared with the system LP oscillator. Thus, Timer1 can use this mode only when the system clock is derived from the internal oscillator. As with the system LP oscillator, the user must provide a software time delay to ensure proper oscillator start-up.

**Note:** The oscillator requires a start-up and stabilization time before use. Thus, T1OSCEN should be set and a suitable delay observed prior to enabling Timer1.

**TABLE 5-1: CAPACITOR SELECTION FOR THE TIMER1 OSCILLATOR**

Osc Type	Freq	C1	C2
LP	32 kHz	33 pF	33 pF
	100 kHz	15 pF	15 pF
	200 kHz	15 pF	15 pF

**These values are for design guidance only.**

**Note 1:** Higher capacitance increases the stability of oscillator but also increases the start-up time.

**2:** Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.

## 5.6 Timer1 Operation During SLEEP

Timer1 can only operate during SLEEP when setup in Asynchronous Counter mode. In this mode, an external crystal or clock source can be used to increment the counter. To setup the timer to wake the device:

- Timer1 must be on (T1CON<0>)
- TMR1IE bit (PIE1<0>) must be set
- PEIE bit (INTCON<6>) must be set

The device will wake-up on an overflow. If the GIE bit (INTCON<7>) is set, the device will wake-up and jump to the Interrupt Service Routine on an overflow.

**TABLE 5-2: REGISTERS ASSOCIATED WITH TIMER1 AS A TIMER/COUNTER**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on all other RESETS
0Bh/8Bh	INTCON	GIE	PEIE	TOIE	INTE	GPIE	TOIF	INTF	GPIF	0000 0000	0000 000u
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	00-- 0--0
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register							xxxx xxxx	uuuu uuuu	
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register							xxxx xxxx	uuuu uuuu	
10h	T1CON	—	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	TMR1CS	TMR1ON	-000 0000	-uuu uuuu
8Ch	PIE1	EEIE	ADIE	—	—	CMIE	—	—	TMR1IE	00-- 0--0	00-- 0--0

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

# PIC12F629/675

---

NOTES:

## 6.0 COMPARATOR MODULE

The PIC12F629/675 devices have one analog comparator. The inputs to the comparator are multiplexed with the GP0 and GP1 pins. There is an on-chip Compara-

tor Voltage Reference that can also be applied to an input of the comparator. In addition, GP2 can be configured as the comparator output. The Comparator Control Register (CMCON), shown in Register 6-1, contains the bits to control the comparator.

### REGISTER 6-1: CMCON — COMPARATOR CONTROL REGISTER (ADDRESS: 19h)

U-0	R-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	COUT	—	CINV	CIS	CM2	CM1	CM0
bit 7							bit 0

- bit 7      **Unimplemented:** Read as '0'
- bit 6      **COUT:** Comparator Output bit  
When CINV = 0:  
 1 =  $V_{IN+} > V_{IN-}$   
 0 =  $V_{IN+} < V_{IN-}$   
When CINV = 1:  
 0 =  $V_{IN+} > V_{IN-}$   
 1 =  $V_{IN+} < V_{IN-}$
- bit 5      **Unimplemented:** Read as '0'
- bit 4      **CINV:** Comparator Output Inversion bit  
 1 = Output inverted  
 0 = Output not inverted
- bit 3      **CIS:** Comparator Input Switch bit  
When CM2:CM0 = 110 or 101:  
 1 =  $V_{IN-}$  connects to CIN+  
 0 =  $V_{IN-}$  connects to CIN-
- bit 2-0    **CM2:CM0:** Comparator Mode bits  
 Figure 6-2 shows the Comparator modes and CM2:CM0 bit settings

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

# PIC12F629/675

## 6.1 Comparator Operation

A single comparator is shown in Figure 6-1, along with the relationship between the analog input levels and the digital output. When the analog input at  $V_{IN+}$  is less than the analog input  $V_{IN-}$ , the output of the comparator is a digital low level. When the analog input at  $V_{IN+}$  is greater than the analog input  $V_{IN-}$ , the output of the comparator is a digital high level. The shaded areas of the output of the comparator in Figure 6-1 represent the uncertainty due to input offsets and response time.

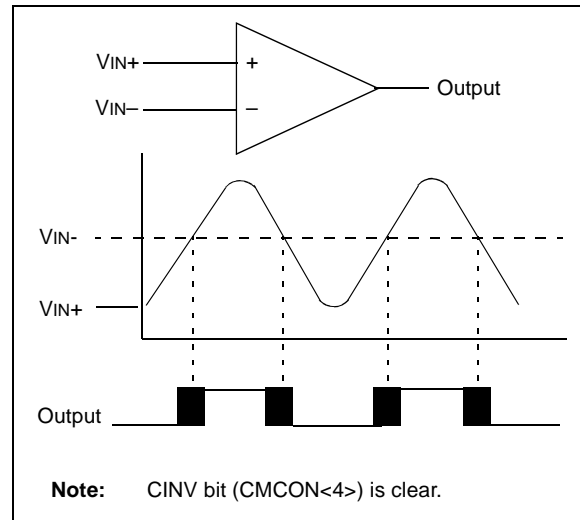
**Note:** To use  $AN<3:0>$  as analog inputs, the appropriate bits must be programmed in the ANSEL register.

The polarity of the comparator output can be inverted by setting the CINV bit ( $CMCON<4>$ ). Clearing CINV results in a non-inverted output. A complete table showing the output state versus input conditions and the polarity bit is shown in Table 6-1.

**TABLE 6-1: OUTPUT STATE VS. INPUT CONDITIONS**

Input Conditions	CINV	COU
$V_{IN-} > V_{IN+}$	0	0
$V_{IN-} < V_{IN+}$	0	1
$V_{IN-} > V_{IN+}$	1	1
$V_{IN-} < V_{IN+}$	1	0

**FIGURE 6-1: SINGLE COMPARATOR**



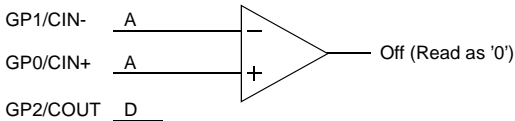
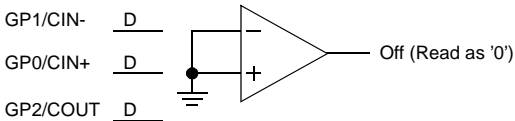
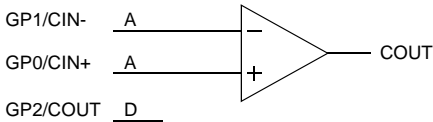
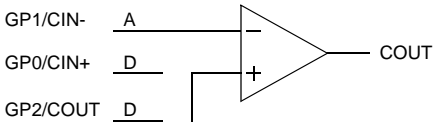
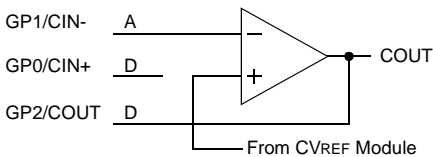
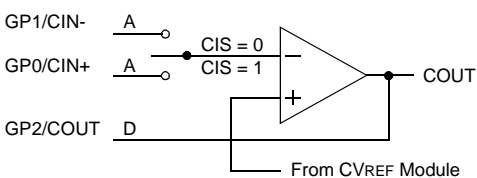
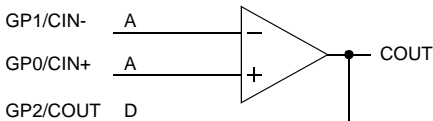
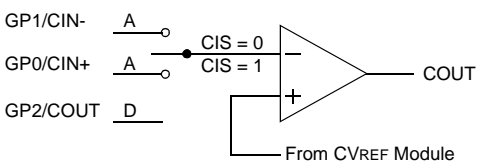
## 6.2 Comparator Configuration

There are eight modes of operation for the comparator. The CMCON register, shown in Register 6-1, is used to select the mode. Figure 6-2 shows the eight possible modes. The TRISIO register controls the data direction of the comparator pins for each mode. If the compara-

tor mode is changed, the comparator output level may not be valid for a specified period of time. Refer to the specifications in Section 12.0.

**Note:** Comparator interrupts should be disabled during a comparator mode change. Otherwise, a false interrupt may occur.

**FIGURE 6-2: COMPARATOR I/O OPERATING MODES**

<p>Comparator Reset (POR Default Value - low power) CM2:CM0 = 000</p>  <p>GP1/CIN- <u>A</u> GP0/CIN+ <u>A</u> GP2/COU <u>D</u></p>	<p>Comparator Off (Lowest power) CM2:CM0 = 111</p>  <p>GP1/CIN- <u>D</u> GP0/CIN+ <u>D</u> GP2/COU <u>D</u></p>
<p>Comparator without Output CM2:CM0 = 010</p>  <p>GP1/CIN- <u>A</u> GP0/CIN+ <u>A</u> GP2/COU <u>D</u></p>	<p>Comparator w/o Output and with Internal Reference CM2:CM0 = 100</p>  <p>GP1/CIN- <u>A</u> GP0/CIN+ <u>D</u> GP2/COU <u>D</u></p> <p style="text-align: right;">From CVREF Module</p>
<p>Comparator with Output and Internal Reference CM2:CM0 = 011</p>  <p>GP1/CIN- <u>A</u> GP0/CIN+ <u>D</u> GP2/COU <u>D</u></p> <p style="text-align: right;">From CVREF Module</p>	<p>Multiplexed Input with Internal Reference and Output CM2:CM0 = 101</p>  <p>GP1/CIN- <u>A</u> GP0/CIN+ <u>A</u> GP2/COU <u>D</u></p> <p style="text-align: right;">From CVREF Module</p>
<p>Comparator with Output CM2:CM0 = 001</p>  <p>GP1/CIN- <u>A</u> GP0/CIN+ <u>A</u> GP2/COU <u>D</u></p>	<p>Multiplexed Input with Internal Reference CM2:CM0 = 110</p>  <p>GP1/CIN- <u>A</u> GP0/CIN+ <u>A</u> GP2/COU <u>D</u></p> <p style="text-align: right;">From CVREF Module</p>
<p>A = Analog Input, ports always reads '0' D = Digital Input CIS = Comparator Input Switch (CMCON&lt;3&gt;)</p>	

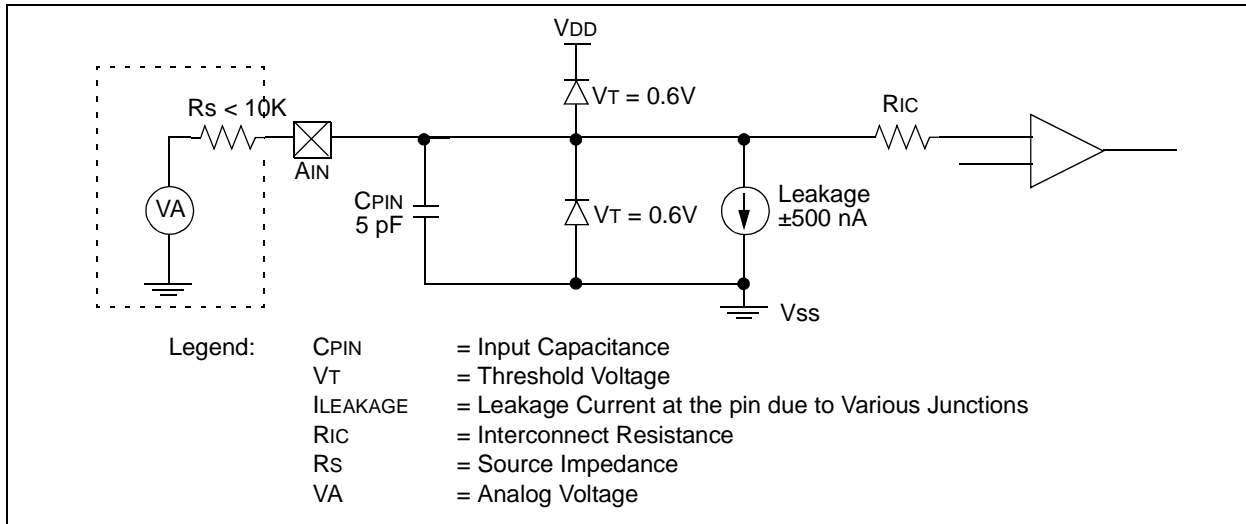
# PIC12F629/675

## 6.3 Analog Input Connection Considerations

A simplified circuit for an analog input is shown in Figure 6-3. 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 latchup 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 6-3: ANALOG INPUT MODE**



## 6.4 Comparator Output

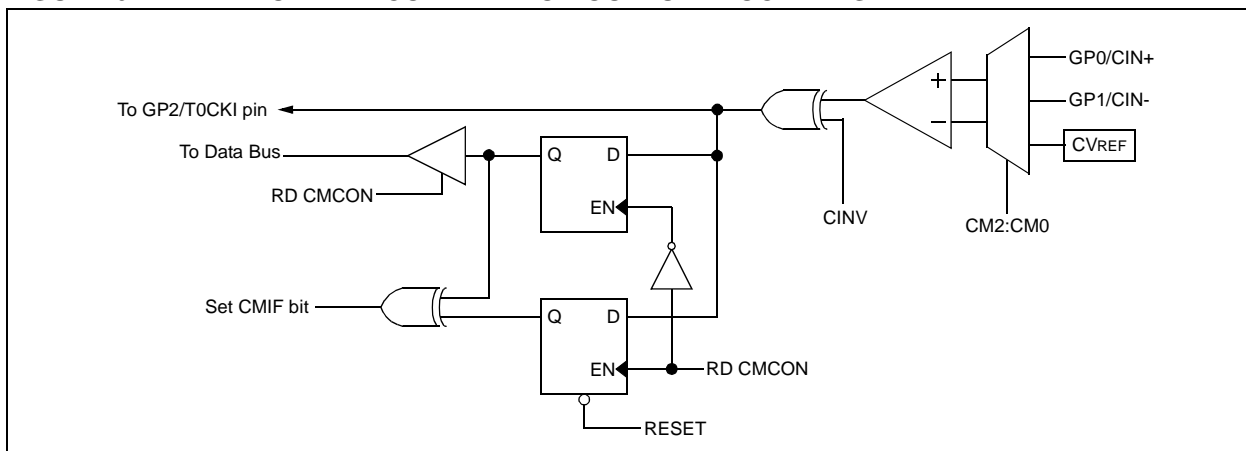
The comparator output, COUT, is read through the CMCON register. This bit is read only. The comparator output may also be directly output to the GP2 pin in three of the eight possible modes, as shown in Figure 6-2. When in one of these modes, the output on GP2 is asynchronous to the internal clock. Figure 6-4 shows the comparator output block diagram.

The TRISIO<2> bit functions as an output enable/disable for the GP2 pin while the comparator is in an output mode.

**Note 1:** When reading the GPIO register, all pins configured as analog inputs will read as a '0'. Pins configured as digital inputs will convert an analog input according to the TTL input specification.

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

**FIGURE 6-4: MODIFIED COMPARATOR OUTPUT BLOCK DIAGRAM**



## 6.5 Comparator Reference

The comparator module also allows the selection of an internally generated voltage reference for one of the comparator inputs. The internal reference signal is used for four of the eight Comparator modes. The VRCON register, Register 6-2, controls the voltage reference module shown in Figure 6-5.

### 6.5.1 CONFIGURING THE VOLTAGE REFERENCE

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

The following equations determine the output voltages:

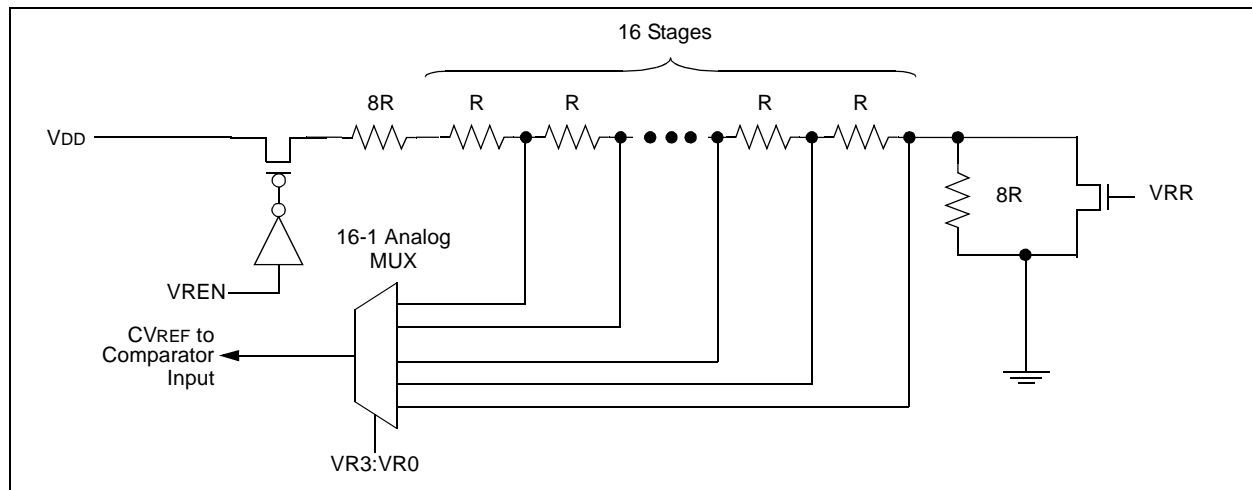
$$VRR = 1 \text{ (low range): } CVREF = (VR3:VR0 / 24) \times VDD$$

$$VRR = 0 \text{ (high range): } CVREF = (VDD / 4) + (VR3:VR0 \times VDD / 32)$$

### 6.5.2 VOLTAGE REFERENCE ACCURACY/ERROR

The full range of VSS to VDD cannot be realized due to the construction of the module. The transistors on the top and bottom of the resistor ladder network (Figure 6-5) keep CVREF from approaching VSS or VDD. 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 12.0.

**FIGURE 6-5: COMPARATOR VOLTAGE REFERENCE BLOCK DIAGRAM**



## 6.6 Comparator Response Time

Response time is the minimum time, after selecting a new reference voltage or input source, before the comparator output is ensured to have a valid level. If the internal reference is changed, the maximum delay of the internal voltage reference must be considered when using the comparator outputs. Otherwise, the maximum delay of the comparators should be used (Table 12-4).

## 6.7 Operation During SLEEP

Both the comparator and voltage reference, if enabled before entering SLEEP mode, remain active during SLEEP. This results in higher SLEEP currents than shown in the power-down specifications. The additional current consumed by the comparator and the voltage reference is shown separately in the specifications. To minimize power consumption while in SLEEP mode, turn off the comparator, CM2:CM0 = 111, and voltage reference, VRCON<7> = 0.

While the comparator is enabled during SLEEP, an interrupt will wake-up the device. If the device wakes up from SLEEP, the contents of the CMCON and VRCON registers are not affected.

## 6.8 Effects of a RESET

A device RESET forces the CMCON and VRCON registers to their RESET states. This forces the comparator module to be in the Comparator Reset mode, CM2:CM0 = 000 and the voltage reference to its off state. Thus, all potential inputs are analog inputs with the comparator and voltage reference disabled to consume the smallest current possible.

# PIC12F629/675

**REGISTER 6-2: VRCON — VOLTAGE REFERENCE CONTROL REGISTER (ADDRESS: 99h)**

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
VREN	—	VRR	—	VR3	VR2	VR1	VR0
bit 7				bit 0			

- bit 7 **VREN:** CVREF Enable bit  
1 = CVREF circuit powered on  
0 = CVREF circuit powered down, no IDD drain
- bit 6 **Unimplemented:** Read as '0'
- bit 5 **VRR:** CVREF Range Selection bit  
1 = Low range  
0 = High range
- bit 4 **Unimplemented:** Read as '0'
- bit 3-0 **VR3:VR0:** CVREF value selection  $0 \leq VR [3:0] \leq 15$   
When VRR = 1:  $CVREF = (VR3:VR0 / 24) * VDD$   
When VRR = 0:  $CVREF = VDD/4 + (VR3:VR0 / 32) * VDD$

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

## 6.9 Comparator Interrupts

The comparator interrupt flag is set whenever there is a change in the output value of the comparator. Software will need to maintain information about the status of the output bits, as read from CMCON<6>, to determine the actual change that has occurred. The CMIF bit, PIR1<3>, is the comparator interrupt flag. This bit must be reset in software by clearing it to '0'. Since it is also possible to write a '1' to this register, a simulated interrupt may be initiated.

The CMIE bit (PIE1<3>) and the PEIE bit (INTCON<6>) must be set to enable the interrupt. In addition, the GIE bit must also be set. If any of these bits are cleared, the interrupt is not enabled, though the CMIF bit will still be set if an interrupt condition occurs.

The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

- a) Any read or write of CMCON. This will end the mismatch condition.
- b) Clear flag bit CMIF.

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

**Note:** If a change in the CMCON register (COUT) should occur when a read operation is being executed (start of the Q2 cycle), then the CMIF (PIR1<3>) interrupt flag may not get set.

**TABLE 6-2: REGISTERS ASSOCIATED WITH COMPARATOR MODULE**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on all other RESETS
0Bh/8Bh	INTCON	GIE	PEIE	TOIE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	0000 000u
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	00-- 0--0
19h	CMCON	—	COUT	—	CINV	CIS	CM2	CM1	CM0	-0-0 0000	-0-0 0000
8Ch	PIE1	EEIE	ADIE	—	—	CMIE	—	—	TMR1IE	00-- 0--0	00-- 0--0
85h	TRISIO	—	—	TRIS5	TRIS4	TRIS3	TRIS2	TRIS1	TRIS0	--11 1111	--11 1111
99h	VRCON	VREN	—	VRR	—	VR3	VR2	VR1	VR0	0-0- 0000	0-0- 0000

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

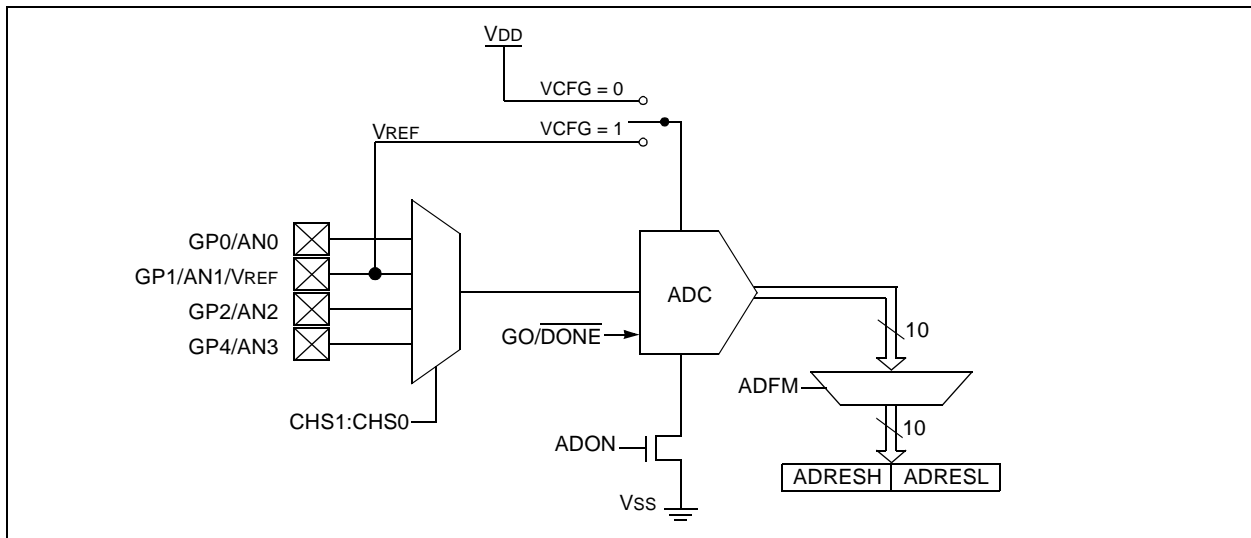


## 7.0 ANALOG-TO-DIGITAL CONVERTER (A/D) MODULE (PIC12F675 ONLY)

The analog-to-digital converter (A/D) allows conversion of an analog input signal to a 10-bit binary representation of that signal. The PIC12F675 has four analog inputs, multiplexed into one sample and hold circuit.

The output of the sample and hold is connected to the input of the converter. The converter generates a binary result via successive approximation and stores the result in a 10-bit register. The voltage reference used in the conversion is software selectable to either VDD or a voltage applied by the VREF pin. Figure 7-1 shows the block diagram of the A/D on the PIC12F675.

**FIGURE 7-1: A/D BLOCK DIAGRAM**



### 7.1 A/D Configuration and Operation

There are two registers available to control the functionality of the A/D module:

1. ADCON0 (Register 7-1)
2. ANSEL (Register 7-2)

#### 7.1.1 ANALOG PORT PINS

The ANS3:ANS0 bits (ANSEL<3:0>) and the TRISIO bits control the operation of the A/D port pins. Set the corresponding TRISIO bits to set the pin output driver to its high impedance state. Likewise, set the corresponding ANS bit to disable the digital input buffer.

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

#### 7.1.2 CHANNEL SELECTION

There are four analog channels on the PIC12F675, AN0 through AN3. The CHS1:CHS0 bits (ADCON0<3:2>) control which channel is connected to the sample and hold circuit.

#### 7.1.3 VOLTAGE REFERENCE

There are two options for the voltage reference to the A/D converter: either VDD is used, or an analog voltage applied to VREF is used. The VCFG bit (ADCON0<6>) controls the voltage reference selection. If VCFG is set, then the voltage on the VREF pin is the reference; otherwise, VDD is the reference.

#### 7.1.4 CONVERSION CLOCK

The A/D conversion cycle requires 11 TAD. The source of the conversion clock is software selectable via the ADCS bits (ANSEL<6:4>). There are seven possible clock options:

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

For correct conversion, the A/D conversion clock (1/TAD) must be selected to ensure a minimum TAD of 1.6  $\mu$ s. Table 7-1 shows a few TAD calculations for selected frequencies.

# PIC12F629/675

**TABLE 7-1: TAD vs. DEVICE OPERATING FREQUENCIES**

A/D Clock Source (TAD)		Device Frequency			
Operation	ADCS2:ADCS0	20 MHz	5 MHz	4 MHz	1.25 MHz
2 TOSC	000	100 ns <sup>(2)</sup>	400 ns <sup>(2)</sup>	500 ns <sup>(2)</sup>	1.6 μs
4 TOSC	100	200 ns <sup>(2)</sup>	800 ns <sup>(2)</sup>	1.0 μs <sup>(2)</sup>	3.2 μs
8 TOSC	001	400 ns <sup>(2)</sup>	1.6 μs	2.0 μs	6.4 μs
16 TOSC	101	800 ns <sup>(2)</sup>	3.2 μs	4.0 μs	12.8 μs <sup>(3)</sup>
32 TOSC	010	1.6 μs	6.4 μs	8.0 μs <sup>(3)</sup>	25.6 μs <sup>(3)</sup>
64 TOSC	110	3.2 μs	12.8 μs <sup>(3)</sup>	16.0 μs <sup>(3)</sup>	51.2 μs <sup>(3)</sup>
A/D RC	x11	2 - 6 μs <sup>(1,4)</sup>	2 - 6 μs <sup>(1,4)</sup>	2 - 6 μs <sup>(1,4)</sup>	2 - 6 μs <sup>(1,4)</sup>

Legend: Shaded cells are outside of recommended range.

**Note 1:** The A/D RC source has a typical TAD time of 4 μs for VDD > 3.0V.

**2:** These values violate the minimum required TAD time.

**3:** For faster conversion times, the selection of another clock source is recommended.

**4:** When the device frequency is greater than 1 MHz, the A/D RC clock source is only recommended if the conversion will be performed during SLEEP.

## 7.1.5 STARTING A CONVERSION

The A/D conversion is initiated by setting the GO/DONE bit (ADCON0<1>). When the conversion is complete, the A/D module:

- Clears the GO/DONE bit
- Sets the ADIF flag (PIR1<6>)
- Generates an interrupt (if enabled).

If the conversion must be aborted, the GO/DONE bit can be cleared in software. The ADRESH:ADRESL registers will not be updated with the partially complete A/D conversion sample. Instead, the ADRESH:ADRESL registers will retain the value of the

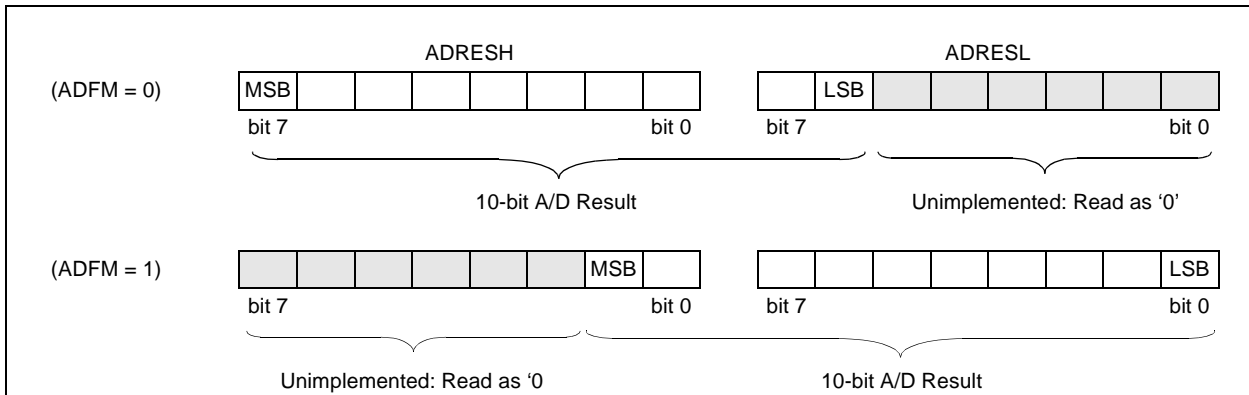
previous conversion. After an aborted conversion, a 2 TAD delay is required before another acquisition can be initiated. Following the delay, an input acquisition is automatically started on the selected channel.

**Note:** The GO/DONE bit should not be set in the same instruction that turns on the A/D.

## 7.1.6 CONVERSION OUTPUT

The A/D conversion can be supplied in two formats: left or right shifted. The ADFM bit (ADCON0<7>) controls the output format. Figure 7-2 shows the output formats.

**FIGURE 7-2: 10-BIT A/D RESULT FORMAT**



## REGISTER 7-1: ADCON0 — A/D CONTROL REGISTER (ADDRESS: 1Fh)

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	VCFG	—	—	CHS1	CHS0	GO/DONE	ADON

bit 7

bit 0

- bit 7      **ADFM:** A/D Result Formed Select bit  
 1 = Right justified  
 0 = Left justified
- bit 6      **VCFG:** Voltage Reference bit  
 1 = VREF pin  
 0 = VDD
- bit 5-4    **Unimplemented:** Read as zero
- bit 3-2    **CHS1:CHS0:** Analog Channel Select bits  
 00 = Channel 00 (AN0)  
 01 = Channel 01 (AN1)  
 10 = Channel 02 (AN2)  
 11 = Channel 03 (AN3)
- bit 1      **GO/DONE:** A/D Conversion Status bit  
 1 = A/D conversion cycle in progress. Setting this bit starts an A/D conversion cycle.  
           This bit is automatically cleared by hardware when the A/D conversion has completed.  
 0 = A/D conversion completed/not in progress
- bit 0      **ADON:** A/D Conversion Status bit  
 1 = A/D converter module is operating  
 0 = A/D converter is shut-off and consumes no operating current

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

# PIC12F629/675

## REGISTER 7-2: ANSEL — ANALOG SELECT REGISTER (ADDRESS: 9Fh)

U-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-1	R/W-1
—	ADCS2	ADCS1	ADCS0	ANS3	ANS2	ANS1	ANS0
bit 7							bit 0

bit 7 **Unimplemented:** Read as '0'.

bit 6-4 **ADCS<2:0>:** A/D Conversion Clock Select bits

000 = Fosc/2

001 = Fosc/8

010 = Fosc/32

x11 = FRC (clock derived from a dedicated internal oscillator = 500 kHz max)

100 = Fosc/4

101 = Fosc/16

110 = Fosc/64

bit 3-0 **ANS3:ANS0:** Analog Select bits

(Between analog or digital function on pins AN<3:0>, respectively.)

0 = Digital I/O; pin is assigned to port or special function

1 = Analog input; pin is assigned as analog input<sup>(1)</sup>

**Note 1:** Setting a pin to an analog input automatically disables the digital input circuitry, weak pull-ups, and interrupt-on-change. The corresponding TRISIO bit must be set to Input mode in order to allow external control of the voltage on the pin.

**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

## 7.2 A/D Acquisition Requirements

### 7.2.1 RECOMMENDED SOURCE IMPEDANCE

The maximum recommended impedance for analog sources is 2.5 kΩ. This value is calculated based on the maximum leakage current of the input pin. The leakage current is 100 nA max., and the analog input voltage cannot be varied by more than 1/4 LSB or 250 μV due to leakage. This places a requirement on the input impedance of 250 μV/100 nA = 2.5 kΩ.

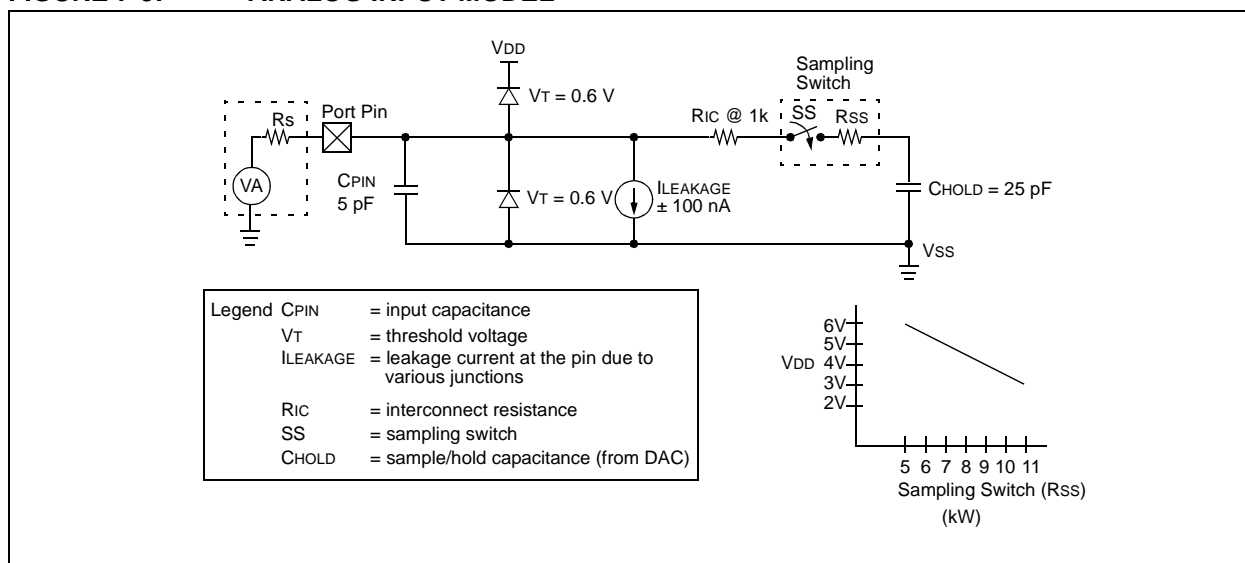
### 7.2.2 SAMPLING TIME CALCULATION

For the A/D converter to meet its specified accuracy, the charge holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The analog input model is shown in Figure 7-3. The source impedance (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD). **The maximum recommended impedance for analog sources is 2.5 kΩ.** After the analog input channel is selected (changed), this sampling must be done before the conversion can be started.

To calculate the minimum sampling time, Equation 7-1 may be used. This equation assumes that 1/4 LSB error is used (4096 steps for the A/D). The 1/4 LSB error is the maximum error allowed for the A/D to meet its specified resolution.

The CHOLD is assumed to be 25 pF for the 10-bit A/D.

FIGURE 7-3: ANALOG INPUT MODEL



# PIC12F629/675

## EQUATION 7-1: A/D SAMPLING TIME

$$V_{HOLD} = \left( V_{REF} - \frac{V_{REF}}{4096} \right) = (V_{REF}) \cdot \left( 1 - e^{\left( \frac{-T_C}{C_{HOLD}} (R_{IC} + R_{SS} + R_S) \right)} \right) V_{REF} \left( 1 - \frac{1}{4096} \right) = V_{REF} \cdot \left( 1 - e^{\left( \frac{-T_C}{C_{HOLD}} (R_{IC} + R_{SS} + R_S) \right)} \right)$$

$$T_C = -C_{HOLD} (1k\Omega + R_{SS} + R_S) \ln \left( \frac{1}{4096} \right)$$

Example 7-1 shows the calculation of the minimum time required to charge CHOLD. This calculation is based on the following system assumptions:

CHOLD = 25 pF

RS = 2.5 kΩ

1/4 LSB error

VDD = 5V → RSS = 10 kΩ (worst case)

Temp (system max.) = 50°C

### EXAMPLE 7-1: CALCULATING THE MINIMUM REQUIRED SAMPLE TIME

TACQ	=	Amplifier Settling Time + Holding Capacitor Charging Time + Temperature Offset †
TACQ	=	5 μs + TC + [(Temp - 25°C)(0.05 ms/°C)] †
TC	=	Holding Capacitor Charging Time
TC	=	(CHOLD) (RIC + RSS + RS) ln (1/4096)
TC	=	-25 pF (1 kΩ + 10 kΩ + 2.5 kΩ) ln (1/4096)
TC	=	-25 pF (13.5 kΩ) ln (1/4096)
TC	=	-0.338 (-9.704) μs
TC	=	3.3 μs
TACQ	=	5 μs + 3.3 μs + [(50°C - 25°C)(0.05 μs / °C)]
TACQ	=	8.3 μs + 1.25 μs
TAC	=	9.55 μs
†		The temperature coefficient is only required for temperatures > 25°C.

**Note 1:** The reference voltage (VREF) has no effect on the equation, since it cancels itself out.

**2:** The charge holding capacitor (CHOLD) is not discharged after each conversion.

**3:** The maximum recommended impedance for analog sources is 2.5 kΩ. This is required to meet the pin leakage specification.

**4:** After a conversion has completed, you must wait 2 TAD time before sampling can begin again. During this time, the holding capacitor is not connected to the selected A/D input channel.

## 7.3 A/D Operation During SLEEP

The A/D converter module can operate during SLEEP. This requires the A/D clock source to be set to the internal RC oscillator. When the RC clock source is selected, the A/D waits one instruction before starting the conversion. This allows the SLEEP instruction to be executed, thus eliminating much of the switching noise from the conversion. When the conversion is complete, the  $\overline{\text{GO/DONE}}$  bit is cleared, and the result is loaded into the ADRESH:ADRESL registers. If the A/D interrupt is enabled, the device awakens from SLEEP. If the A/D interrupt is not enabled, the A/D module is turned off, although the ADON bit remains set.

When the A/D clock source is something other than RC, a SLEEP instruction causes the present conversion to be aborted, and the A/D module is turned off. The ADON bit remains set.

## 7.4 Effects of RESET

A device RESET forces all registers to their RESET state. Thus the A/D module is turned off and any pending conversion is aborted. The ADRESH:ADRESL registers are unchanged.

**TABLE 7-2: SUMMARY OF A/D REGISTERS**

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
05h	GPIO	—	—	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	--xx xxxxx	--uu uuuu
0Bh, 8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	0000 000u
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	00-- 0--0
1Eh	ADRESH	Most Significant 8 bits of the Left Shifted A/D result or 2 bits of the Right Shifted Result								xxxx xxxxx	uuuu uuuu
1Fh	ADCON0	ADFM	VCFG	—	—	CHS1	CHS0	GO	ADON	00-- 0000	00-- 0000
85h	TRISIO	—	—	TRIS5	TRIS4	TRIS3	TRIS2	TRIS1	TRIS0	--11 1111	--11 1111
8Ch	PIE1	EEIE	ADIE	—	—	CMIE	—	—	TMR1IE	00-- 0--0	00-- 0--0
9Eh	ADRESL	Least Significant 2 bits of the Left Shifted A/D Result or 8 bits of the Right Shifted Result								xxxx xxxxx	uuuu uuuu
9Fh	ANSEL	—	ADCS2	ADCS1	ADCS0	ANS3	ANS2	ANS1	ANS0	-000 1111	-000 1111

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used for A/D converter module.

# PIC12F629/675

---

NOTES:



## 8.0 DATA EEPROM MEMORY

The EEPROM 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. There are four SFRs used to read and write this memory:

- EECON1
- EECON2 (not a physically implemented register)
- EEDATA
- EEADR

EEDATA holds the 8-bit data for read/write, and EEADR holds the address of the EEPROM location being accessed. PIC12F629/675 devices have 128 bytes of data EEPROM with an address range from 0h to 7Fh.

The EEPROM data memory allows byte read and write. A byte write automatically erases the location and writes the new data (erase before write). The EEPROM data memory is rated for high erase/write cycles. The write time is controlled by an on-chip timer. The write time will vary with voltage and temperature as well as from chip to chip. Please refer to AC Specifications for exact limits.

When the data memory is code protected, the CPU may continue to read and write the data EEPROM memory. The device programmer can no longer access this memory.

Additional information on the Data EEPROM is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

### REGISTER 8-1: EEDAT — EEPROM DATA REGISTER (ADDRESS: 9Ah)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EEDAT7	EEDAT6	EEDAT5	EEDAT4	EEDAT3	EEDAT2	EEDAT1	EEDAT0
bit 7							bit 0

bit 7-0 **EEDATn:** Byte value to write to or read from Data EEPROM

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

### REGISTER 8-2: EEADR — EEPROM ADDRESS REGISTER (ADDRESS: 9Bh)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	EADR6	EADR5	EADR4	EADR3	EADR2	EADR1	EADR0
bit 7							bit 0

bit 7 **Unimplemented:** Should be set to '0'

bit 6-0 **EEADR:** Specifies one of 128 locations for EEPROM Read/Write Operation

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

# PIC12F629/675

## 8.1 EEADR

The EEADR register can address up to a maximum of 128 bytes of data EEPROM. Only seven of the eight bits in the register (EEADR<6:0>) are required. The MSb (bit 7) is ignored.

The upper bit should always be '0' to remain upward compatible with devices that have more data EEPROM memory.

## 8.2 EECON1 AND EECON2 REGISTERS

EECON1 is the control register with four low order bits physically implemented. The upper four bits are non-implemented and read as '0's.

Control bits RD and WR initiate read and write, respectively. These bits cannot be cleared, only set, in software. They are cleared in hardware at completion

of the read or write operation. The inability to clear the WR bit in software prevents the accidental, premature termination of a write operation.

The WREN bit, when set, will allow a write operation. On power-up, the WREN bit is clear. The WRERR bit is set when a write operation is interrupted by a  $\overline{\text{MCLR}}$  Reset, or a WDT Time-out Reset during normal operation. In these situations, following RESET, the user can check the WRERR bit, clear it, and rewrite the location. The data and address will be cleared, therefore, the EEDATA and EEADR registers will need to be re-initialized.

Interrupt flag bit EEIF in the PIR1 register is set when write is complete. This bit must be cleared in software.

EECON2 is not a physical register. Reading EECON2 will read all '0's. The EECON2 register is used exclusively in the Data EEPROM write sequence.

### REGISTER 8-3: EECON1 — EEPROM CONTROL REGISTER (ADDRESS: 9Ch)

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

bit 7-4 **Unimplemented:** Read as '0'

bit 3 **WRERR:** EEPROM Error Flag bit  
 1 = A write operation is prematurely terminated (any  $\overline{\text{MCLR}}$  Reset, any WDT Reset during normal operation or BOD detect)  
 0 = The write operation completed

bit 2 **WREN:** EEPROM Write Enable bit  
 1 = Allows write cycles  
 0 = Inhibits write to the data EEPROM

bit 1 **WR:** Write Control bit  
 1 = Initiates a write cycle (The bit is cleared by hardware once write is complete. The WR bit can only be set, not cleared, in software.)  
 0 = Write cycle to the data EEPROM is complete

bit 0 **RD:** Read Control bit  
 1 = Initiates an EEPROM read (Read takes one cycle. RD is cleared in hardware. The RD bit can only be set, not cleared, in software.)  
 0 = Does not initiate an EEPROM read

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

## 8.3 READING THE EEPROM DATA MEMORY

To read a data memory location, the user must write the address to the EEADR register and then set control bit RD (EECON1<0>), as shown in Example 8-1. The data is available, in the very next cycle, in the EEDATA register. Therefore, it can be read in the next instruction. EEDATA holds this value until another read, or until it is written to by the user (during a write operation).

### EXAMPLE 8-1: DATA EEPROM READ

```

bsf    STATUS,RP0    ;Bank 1
movlw  CONFIG_ADDR  ;
movwf  EEADR         ;Address to read
bsf    EECON1,RD    ;EE Read
movf   EEDATA,W     ;Move data to W
    
```

## 8.4 WRITING TO THE EEPROM DATA MEMORY

To write an EEPROM data location, the user must first write the address to the EEADR register and the data to the EEDATA register. Then the user must follow a specific sequence to initiate the write for each byte, as shown in Example 8-2.

### EXAMPLE 8-2: DATA EEPROM WRITE

```

bsf    STATUS,RP0    ;Bank 1
bsf    EECON1,WREN   ;Enable write
bcf    INTCON,GIE    ;Disable INTs
movlw  55h           ;Unlock write
movwf  EECON2        ;
movlw  AAh           ;
movwf  EECON2        ;
bsf    EECON1,WR     ;Start the write
bsf    INTCON,GIE    ;Enable INTS
    
```

The write will not initiate if the above sequence is not exactly followed (write 55h to EECON2, write AAh to EECON2, then set WR bit) for each byte. We strongly recommend that interrupts be disabled during this code segment. A cycle count is executed during the required sequence. Any number that is not equal to the required cycles to execute the required sequence will prevent the data from being written into the EEPROM.

Additionally, the WREN bit in EECON1 must be set to enable write. This mechanism prevents accidental writes to data EEPROM due to errant (unexpected) code execution (i.e., lost programs). The user should keep the WREN bit clear at all times, except when updating EEPROM. The WREN bit is not cleared by hardware.

After a write sequence has been initiated, clearing the WREN bit will not affect this write cycle. The WR bit will be inhibited from being set unless the WREN bit is set.

At the completion of the write cycle, the WR bit is cleared in hardware and the EE Write Complete Interrupt Flag bit (EEIF) is set. The user can either enable this interrupt or poll this bit. The EEIF bit (PIR<7>) register must be cleared by software.

## 8.5 WRITE VERIFY

Depending on the application, good programming practice may dictate that the value written to the Data EEPROM should be verified (see Example 8-3) to the desired value to be written.

### EXAMPLE 8-3: WRITE VERIFY

```

bcf    STATUS,RP0    ;Bank 0
:      ;Any code
bsf    STATUS,RP0    ;Bank 1 READ
movf   EEDATA,W     ;EEDATA not changed
:      ;from previous write

bsf    EECON1,RD    ;YES, Read the
:      ;value written

xorwf  EEDATA,W     ;
btfsz  STATUS,Z     ;Is data the same
goto   WRITE_ERR    ;No, handle error
:      ;Yes, continue
    
```

### 8.5.1 MAXIMIZING ENDURANCE

For applications that will exceed 10% of the minimum specified cell endurance (parameters D120, D120A, D130, and D130A), every location should be refreshed within intervals not exceeding 1/10 of this specified cell endurance. Please refer to AN790 (DS00790) for more details.

## 8.6 PROTECTION AGAINST SPURIOUS WRITE

There are conditions when the device may not want to write to the data EEPROM memory. To protect against spurious EEPROM writes, various mechanisms have been built in. On power-up, WREN is cleared. Also, the Power-up Timer (72 ms duration) prevents EEPROM write.

The write initiate sequence and the WREN bit together help prevent an accidental write during:

- brown-out
- power glitch
- software malfunction

# PIC12F629/675

## 8.7 DATA EEPROM OPERATION DURING CODE PROTECT

Data memory can be code protected by programming the CPD bit to '0'.

When the data memory is code protected, the CPU is able to read and write data to the Data EEPROM. It is recommended to code protect the program memory when code protecting data memory. This prevents anyone from programming zeroes over the existing code (which will execute as NOPs) to reach an added routine, programmed in unused program memory, which outputs the contents of data memory. Programming unused locations to '0' will also help prevent data memory code protection from becoming breached.

**TABLE 8-1: REGISTERS/BITS ASSOCIATED WITH DATA EEPROM**

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 all other RESETS	
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	00-- 0--0	
9Ah	EEDATA	EEPROM Data Register								0000 0000	0000 0000	
9Bh	EEADR	—	EEPROM Address Register								-000 0000	-000 0000
9Ch	EECON1	—	—	—	—	WRERR	WREN	WR	RD	---- x000	---- q000	
9Dh	EECON2 <sup>(1)</sup>	EEPROM Control Register 2								---- ----	---- ----	

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

Shaded cells are not used by Data EEPROM module.

**Note 1:** EECON2 is not a physical register.

## 9.0 SPECIAL FEATURES OF THE CPU

Certain special circuits that deal with the needs of real time applications are what sets a microcontroller apart from other processors. The PIC12F629/675 family has 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)
  - Power-up Timer (PWRT)
  - Oscillator Start-Up Timer (OST)
  - Brown-out Reset (BOR)
- Interrupts
- Watchdog Timer (WDT)
- SLEEP
- Code protection
- ID Locations
- In-Circuit Serial Programming

The PIC12F629/675 has a Watchdog Timer that is controlled by configuration bits. It runs off its own RC oscillator for added reliability. There are two timers that offer necessary delays on power-up. One is the Oscillator Start-up Timer (OST), intended to keep the chip in RESET until the crystal oscillator is stable. The other is the Power-up Timer (PWRT), which provides a fixed delay of 72 ms (nominal) on power-up only, designed to keep the part in RESET while the power supply stabilizes. There is also circuitry to reset the device if a brown-out occurs, which can provide at least a 72 ms RESET. With these three functions on-chip, most applications need no 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:

- External RESET
- Watchdog Timer wake-up
- An interrupt

Several oscillator options are also made available to allow the part to fit the application. The RC oscillator option saves system cost while the LP crystal option saves power. A set of configuration bits are used to select various options (see Register 9-1).

# PIC12F629/675

## 9.1 Configuration Bits

The configuration bits can be programmed (read as '0'), or left unprogrammed (read as '1') to select various device configurations, as shown in Register 9-1. These bits are mapped in program memory location 2007h.

**Note:** Address 2007h is beyond the user program memory space. It belongs to the special configuration memory space (2000h - 3FFFh), which can be accessed only during programming. See PIC12F629/675 Programming Specification for more information.

**REGISTER 9-1: CONFIG — CONFIGURATION WORD (ADDRESS: 2007h)**

R/P-1	R/P-1	U-0	U-0	U-0	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
BG1	BG0	—	—	—	$\overline{\text{CPD}}$	$\overline{\text{CP}}$	BODEN	MCLRE	$\overline{\text{PWRT}}\overline{\text{E}}$	WDTE	F0SC2	F0SC1	F0SC0	
bit 13													bit 0	

- bit 13-12 **BG1:BG0:** Bandgap Calibration bits<sup>(1)</sup>  
00 = Lowest bandgap voltage  
11 = Highest bandgap voltage
- bit 11-9 **Unimplemented:** Read as '0'
- bit 8  **$\overline{\text{CPD}}$ :** Data Code Protection bit<sup>(2)</sup>  
1 = Data memory code protection is disabled  
0 = Data memory code protection is enabled
- bit 7  **$\overline{\text{CP}}$ :** Code Protection bit<sup>(3)</sup>  
1 = Program Memory code protection is disabled  
0 = Program Memory code protection is enabled
- bit 6 **BODEN:** Brown-out Detect Enable bit<sup>(4)</sup>  
1 = BOD enabled  
0 = BOD disabled
- bit 5 **MCLRE:** GP3/ $\overline{\text{MCLR}}$  pin function select<sup>(5)</sup>  
1 = GP3/ $\overline{\text{MCLR}}$  pin function is MCLR  
0 = GP3/ $\overline{\text{MCLR}}$  pin function is digital I/O,  $\overline{\text{MCLR}}$  internally tied to VDD
- bit 4  **$\overline{\text{PWRT}}\overline{\text{E}}$ :** Power-up Timer Enable bit  
1 = PWRT disabled  
0 = PWRT enabled
- bit 3 **WDTE:** Watchdog Timer Enable bit  
1 = WDT enabled  
0 = WDT disabled
- bit 2-0 **F0SC2:F0SC0:** Oscillator Selection bits  
111 = RC oscillator: CLKOUT function on GP4/OSC2/CLKOUT pin, RC on GP5/OSC1/CLKIN  
110 = RC oscillator: I/O function on GP4/OSC2/CLKOUT pin, RC on GP5/OSC1/CLKIN  
101 = INTOSC oscillator: CLKOUT function on GP4/OSC2/CLKOUT pin, I/O function on GP5/OSC1/CLKIN  
100 = INTOSC oscillator: I/O function on GP4/OSC2/CLKOUT pin, I/O function on GP5/OSC1/CLKIN  
011 = EC: I/O function on GP4/OSC2/CLKOUT pin, CLKIN on GP5/OSC1/CLKIN  
010 = HS oscillator: High speed crystal/resonator on GP4/OSC2/CLKOUT and RA7/OSC1/CLKIN  
001 = XT oscillator: Crystal/resonator on GP4/OSC2/CLKOUT and GP5/OSC1/CLKIN  
000 = LP oscillator: Low power crystal on GP4/OSC2/CLKOUT and GP5/OSC1/CLKIN

- Note 1:** The Bandgap Calibration bits are factory programmed and must be read and saved prior to erasing the device.
- Note 2:** The entire data EEPROM will be erased when the code protection is turned off.
- Note 3:** The entire program EEPROM will be erased, including OSCCAL value, when the code protection is turned off.
- Note 4:** Enabling Brown-out Reset does not automatically enable Power-Up Timer.
- Note 5:** When MCLR is asserted in INTOSC or RC mode, the internal clock oscillator is disabled.

Legend:

P = Programmed using ICSP		
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

## 9.2 Oscillator Configurations

### 9.2.1 OSCILLATOR TYPES

The PIC12F629/675 can be operated in eight different oscillator option modes. The user can program three configuration bits (FOSC2 through FOSC0) to select one of these eight modes:

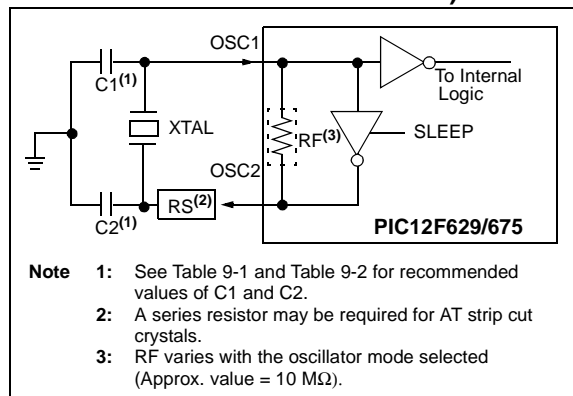
- LP Low Power Crystal
- XT Crystal/Resonator
- HS High Speed Crystal/Resonator
- RC External Resistor/Capacitor (2 modes)
- INTOSC Internal Oscillator (2 modes)
- EC External Clock In

**Note:** Additional information on oscillator configurations is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

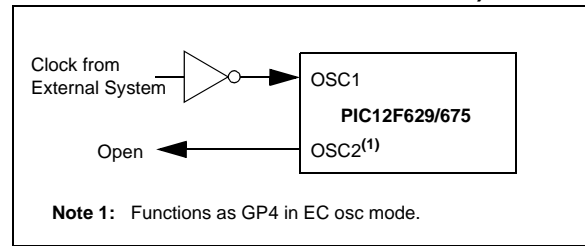
### 9.2.2 CRYSTAL OSCILLATOR / CERAMIC RESONATORS

In XT, LP or HS modes a crystal or ceramic resonator is connected to the OSC1 and OSC2 pins to establish oscillation (see Figure 9-1). The PIC12F629/675 oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may yield a frequency outside of the crystal manufacturers specifications. When in XT, LP or HS modes, the device can have an external clock source to drive the OSC1 pin (see Figure 9-2).

**FIGURE 9-1: CRYSTAL OPERATION (OR CERAMIC RESONATOR) (HS, XT OR LP OSC CONFIGURATION)**



**FIGURE 9-2: EXTERNAL CLOCK INPUT OPERATION (HS, XT, EC, OR LP OSC CONFIGURATION)**



**TABLE 9-1: CAPACITOR SELECTION FOR CERAMIC RESONATORS**

Ranges Characterized:			
Mode	Freq	OSC1(C1)	OSC2(C2)
XT	455 kHz	68 - 100 pF	68 - 100 pF
	2.0 MHz	15 - 68 pF	15 - 68 pF
	4.0 MHz	15 - 68 pF	15 - 68 pF
HS	8.0 MHz	10 - 68 pF	10 - 68 pF
	16.0 MHz	10 - 22 pF	10 - 22 pF

**Note 1:** Higher capacitance increases the stability of the oscillator but also increases the start-up time. 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 9-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR**

Mode	Freq	OSC1(C1)	OSC2(C2)
LP	32 kHz	68 - 100 pF	68 - 100 pF
	200 kHz	15 - 30 pF	15 - 30 pF
XT	100 kHz	68 - 150 pF	150 - 200 pF
	2 MHz	15 - 30 pF	15 - 30 pF
	4 MHz	15 - 30 pF	15 - 30 pF
HS	8 MHz	15 - 30 pF	15 - 30 pF
	10 MHz	15 - 30 pF	15 - 30 pF
	20 MHz	15 - 30 pF	15 - 30 pF

**Note 1:** Higher capacitance increases the stability of the oscillator but also increases the start-up time. These values are for design guidance only. Rs may be required in HS mode as well as XT mode to avoid over-driving 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.

# PIC12F629/675

## 9.2.3 EXTERNAL CLOCK IN

For applications where a clock is already available elsewhere, users may directly drive the PIC12F629/675 provided that this external clock source meets the AC/DC timing requirements listed in Section 12.0. Figure 9-2 below shows how an external clock circuit should be configured.

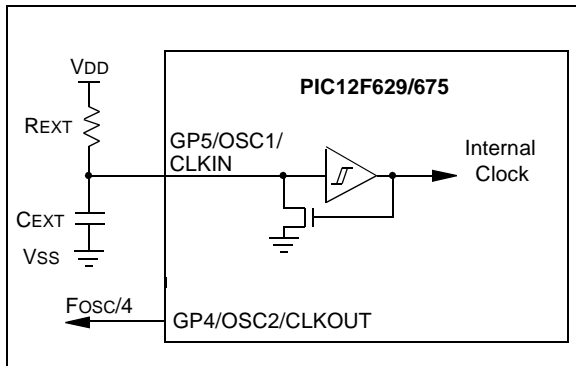
## 9.2.4 RC OSCILLATOR

For applications where precise timing is not a requirement, the RC oscillator option is available. The operation and functionality of the RC oscillator is dependent upon a number of variables. The RC oscillator frequency is a function of:

- Supply voltage
- Resistor (R<sub>EXT</sub>) and capacitor (C<sub>EXT</sub>) values
- Operating temperature.

The oscillator frequency will vary from unit to unit due to normal process parameter variation. The difference in lead frame capacitance between package types will also affect the oscillation frequency, especially for low C<sub>EXT</sub> values. The user also needs to account for the tolerance of the external R and C components. Figure 9-3 shows how the R/C combination is connected.

**FIGURE 9-3: RC OSCILLATOR MODE**



## 9.2.5 INTERNAL 4 MHz OSCILLATOR

When calibrated, the internal oscillator provides a fixed 4 MHz (nominal) system clock. See Electrical Specifications, Section 12.0, for information on variation over voltage and temperature.

### 9.2.5.1 Calibrating the Internal Oscillator

A calibration instruction is programmed into the last location of program memory. This instruction is a RETLW *XX*, where the literal is the calibration value. The literal is placed in the OSCCAL register to set the calibration of the internal oscillator. Example 9-1 demonstrates how to calibrate the internal oscillator.

**Note:** Erasing the device will also erase the pre-programmed internal calibration value for the internal oscillator. The calibration value must be saved prior to erasing part.

### EXAMPLE 9-1: CALIBRATING THE INTERNAL OSCILLATOR

```
bsf    STATUS, RP0    ;Bank 1
call   3FFh           ;Get the cal value
movwf  OSCCAL         ;Calibrate
bcf    STATUS, RP0    ;Bank 0
```

## 9.2.6 CLKOUT

The PIC12F629/675 devices can be configured to provide a clock out signal in the INTOSC and RC oscillator modes. When configured, the oscillator frequency divided by four (F<sub>OSC</sub>/4) is output on the GP4/OSC2/CLKOUT pin. F<sub>OSC</sub>/4 can be used for test purposes or to synchronize other logic.



## 9.3 RESET

The PIC12F629/675 differentiates between various kinds of RESET:

- Power-on Reset (POR)
- $\overline{\text{MCLR}}$  Reset during normal operation
- $\overline{\text{MCLR}}$  Reset during SLEEP
- WDT Reset (normal operation)
- Brown-out Detect (BOD)

Some registers are not affected in any RESET condition; their status is unknown on POR and unchanged in any other RESET. Most other registers are reset to a "RESET state" on:

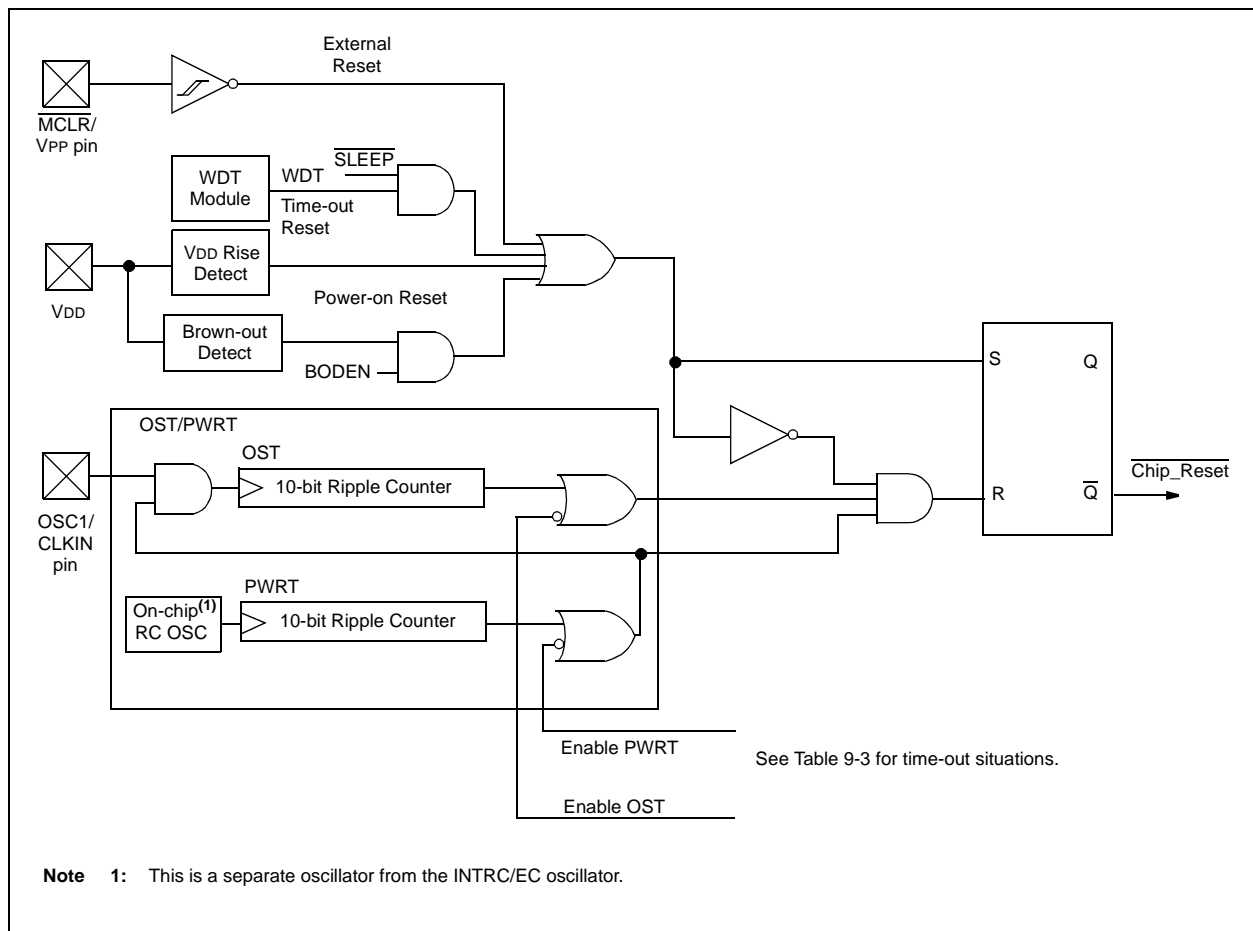
- Power-on Reset
- $\overline{\text{MCLR}}$  Reset
- WDT Reset
- $\overline{\text{MCLR}}$  Reset during SLEEP
- Brown-out Detect (BOD) Reset

They are not affected by a WDT wake-up, since this is viewed as the resumption of normal operation.  $\overline{\text{TO}}$  and  $\overline{\text{PD}}$  bits are set or cleared differently in different RESET situations as indicated in Table 9-4. These bits are used in software to determine the nature of the RESET. See Table 9-7 for a full description of RESET states of all registers.

A simplified block diagram of the On-Chip Reset Circuit is shown in Figure 9-4.

The  $\overline{\text{MCLR}}$  Reset path has a noise filter to detect and ignore small pulses. See Table 12-4 in Electrical Specifications Section for pulse width specification.

**FIGURE 9-4: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT**



# PIC12F629/675

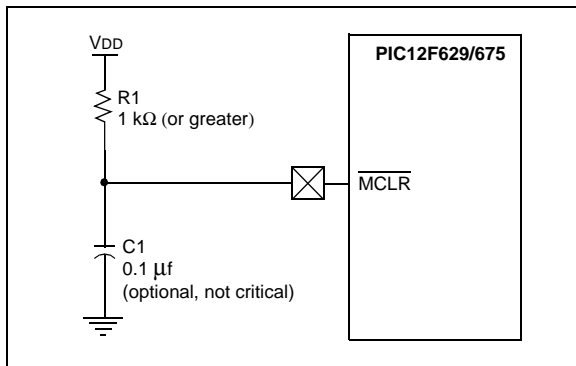
## 9.3.1 $\overline{\text{MCLR}}$

PIC12F629/675 devices have a noise filter in the  $\overline{\text{MCLR}}$  Reset path. The filter will detect and ignore small pulses.

It should be noted that a WDT Reset does not drive  $\overline{\text{MCLR}}$  pin low.

The behavior of the ESD protection on the  $\overline{\text{MCLR}}$  pin has been altered from previous devices of this family. Voltages applied to the pin that exceed its specification can result in both  $\overline{\text{MCLR}}$  Resets and excessive current beyond the device specification during the ESD event. For this reason, Microchip recommends that the  $\overline{\text{MCLR}}$  pin no longer be tied directly to VDD. The use of an RC network, as shown in Figure 9-5, is suggested.

**FIGURE 9-5: RECOMMENDED  $\overline{\text{MCLR}}$  CIRCUIT**



## 9.3.2 POWER-ON RESET (POR)

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 POR, simply tie the  $\overline{\text{MCLR}}$  pin through a resistor to VDD. This will eliminate external RC components usually needed to create Power-on Reset. A maximum rise time for VDD is required. See Electrical Specifications for details.

**Note:** The POR circuit does not produce an internal RESET when VDD declines.

When the device starts normal operation (exits the RESET condition), device operating parameters (i.e., 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 Note AN607 "Power-up Trouble Shooting".

## 9.3.3 POWER-UP TIMER (PWRT)

The Power-up Timer provides a fixed 72 ms (nominal) time-out on power-up only, from POR or Brown-out Reset. The Power-up Timer operates on an internal RC oscillator. The chip is kept in RESET as long as PWRT is active. The PWRT delay allows the VDD to rise to an acceptable level. A configuration bit,  $\overline{\text{PWRTEN}}$  can disable (if set) or enable (if cleared or programmed) the Power-up Timer. The Power-up Timer should always be enabled when Brown-out Reset is enabled.

The Power-Up Time delay will vary from chip to chip and due to:

- VDD variation
- Temperature variation
- Process variation.

See DC parameters for details.

## 9.3.4 OSCILLATOR START-UP TIMER (OST)

The Oscillator Start-Up Timer (OST) provides a 1024 oscillator cycle (from OSC1 input) delay after the PWRT delay is over. This ensures that the crystal oscillator or resonator has started and stabilized.

The OST time-out is invoked only for XT, LP and HS modes and only on Power-on Reset or wake-up from SLEEP.

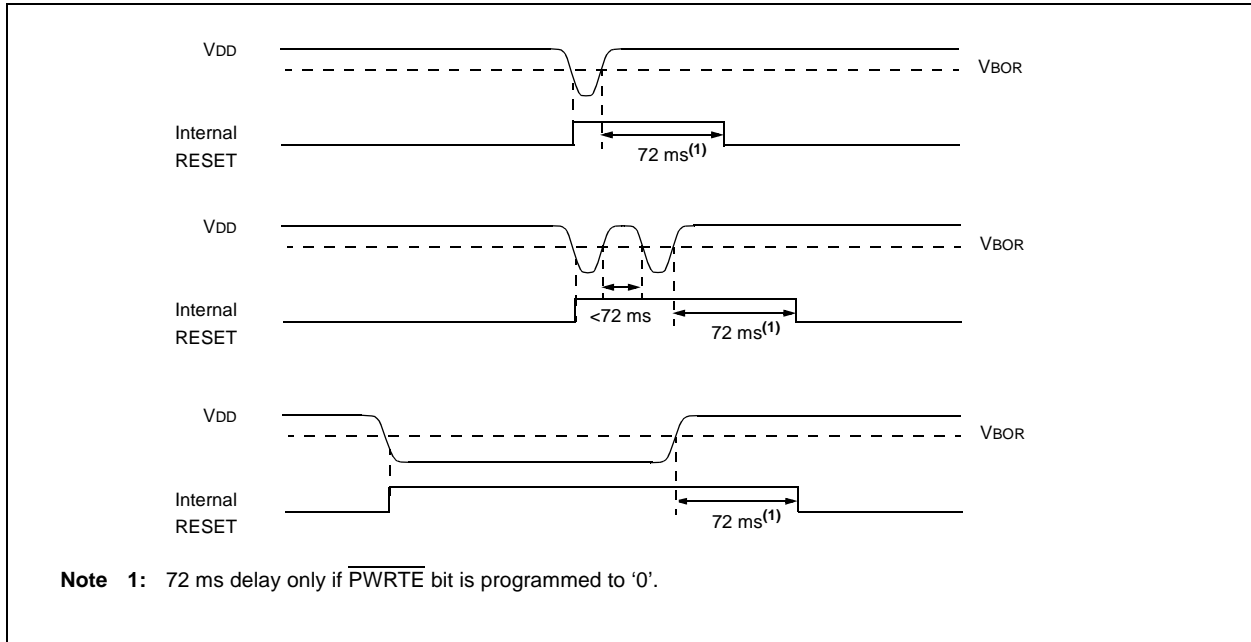
## 9.3.5 BROWN-OUT DETECT (BOD)

The PIC12F629/675 members have on-chip Brown-out Detect circuitry. A configuration bit, BODEN, can disable (if clear/programmed) or enable (if set) the Brown-out Detect circuitry. If VDD falls below VBOR for greater than parameter (TBOR) in Table 12-4 (see Section 12.0). The brown-out situation will reset the chip. A RESET is not guaranteed to occur if VDD falls below VBOR for less than parameter (TBOR).

On any RESET (Power-on, Brown-out, Watchdog, etc.), the chip will remain in RESET until VDD rises above BVDD (see Figure 9-6). The Power-up Timer will now be invoked, if enabled, and will keep the chip in RESET an additional 72 ms.

If VDD drops below BVDD while the Power-up Timer is running, the chip will go back into a Brown-out Reset and the Power-up Timer will be re-initialized. Once VDD rises above BVDD, the Power-Up Timer will execute a 72 ms RESET. The Power-up Timer should always be enabled when Brown-out Detect is enabled. Figure 9-6 shows typical Brown-out situations.

**FIGURE 9-6: BROWN-OUT SITUATIONS**



### 9.3.6 TIME-OUT SEQUENCE

On power-up, the time-out sequence is as follows: first, PWRT time-out is invoked after POR has expired. Then, OST is activated. The total time-out will vary based on oscillator configuration and  $\overline{\text{PWRTE}}$  bit status. For example, in EC mode with  $\overline{\text{PWRTE}}$  bit erased (PWRT disabled), there will be no time-out at all. Figure 9-7, Figure 9-8 and Figure 9-9 depict time-out sequences.

Since the time-outs occur from the POR pulse, if  $\overline{\text{MCLR}}$  is kept low long enough, the time-outs will expire. Then bringing  $\overline{\text{MCLR}}$  high will begin execution immediately (see Figure 9-8). This is useful for testing purposes or to synchronize more than one PIC12F629/675 device operating in parallel.

Table 9-6 shows the RESET conditions for some special registers, while Table 9-7 shows the RESET conditions for all the registers.

### 9.3.7 POWER CONTROL (PCON) STATUS REGISTER

The power control/status register, PCON (address 8Eh) has two bits.

Bit0 is  $\overline{\text{BOD}}$  (Brown-out).  $\overline{\text{BOD}}$  is unknown on Power-on Reset. It must then be set by the user and checked on subsequent RESETS to see if  $\overline{\text{BOD}} = 0$ , indicating that a brown-out has occurred. The  $\overline{\text{BOD}}$  status bit is a don't care and is not necessarily predictable if the brown-out circuit is disabled (by setting  $\overline{\text{BODEN}}$  bit = 0 in the Configuration word).

Bit1 is  $\overline{\text{POR}}$  (Power-on Reset). It is a '0' on Power-on Reset and unaffected otherwise. The user must write a '1' to this bit following a Power-on Reset. On a subsequent RESET, if  $\overline{\text{POR}}$  is '0', it will indicate that a Power-on Reset must have occurred (i.e., VDD may have gone too low).

# PIC12F629/675

**TABLE 9-3: TIME-OUT IN VARIOUS SITUATIONS**

Oscillator Configuration	Power-up		Brown-out Reset		Wake-up from SLEEP
	$\overline{\text{PWRTE}} = 0$	$\overline{\text{PWRTE}} = 1$	$\overline{\text{PWRTE}} = 0$	$\overline{\text{PWRTE}} = 1$	
XT, HS, LP	TPWRT + 1024•TOSC	1024•TOSC	TPWRT + 1024•TOSC	1024•TOSC	1024•TOSC
RC, EC, INTOSC	TPWRT	—	TPWRT	—	—

**TABLE 9-4: STATUS/PCON BITS AND THEIR SIGNIFICANCE**

$\overline{\text{POR}}$	$\overline{\text{BOD}}$	$\overline{\text{TO}}$	$\overline{\text{PD}}$	
0	u	1	1	Power-on Reset
1	0	1	1	Brown-out Detect
u	u	0	u	WDT Reset
u	u	0	0	WDT Wake-up
u	u	u	u	$\overline{\text{MCLR}}$ Reset during normal operation
u	u	1	0	$\overline{\text{MCLR}}$ Reset during SLEEP

Legend: u = unchanged, x = unknown

**TABLE 9-5: SUMMARY OF REGISTERS ASSOCIATED WITH BROWN-OUT**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR Reset	Value on all other RESETS <sup>(1)</sup>
03h	STATUS	IRP	RP1	RPO	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0001 1xxx	000q quuu
8Eh	PCON	—	—	—	—	—	—	$\overline{\text{POR}}$	$\overline{\text{BOD}}$	---- --0x	---- --uq

**Note 1:** Other (non Power-up) Resets include  $\overline{\text{MCLR}}$  Reset, Brown-out Detect and Watchdog Timer Reset during normal operation.

**TABLE 9-6: INITIALIZATION CONDITION FOR SPECIAL REGISTERS**

Condition	Program Counter	STATUS Register	PCON Register
Power-on Reset	000h	0001 1xxx	---- --0x
$\overline{\text{MCLR}}$ Reset during normal operation	000h	000u uuuu	---- --uu
$\overline{\text{MCLR}}$ Reset during SLEEP	000h	0001 0uuu	---- --uu
WDT Reset	000h	0000 uuuu	---- --uu
WDT Wake-up	PC + 1	uuu0 0uuu	---- --uu
Brown-out Detect	000h	0001 1uuu	---- --10
Interrupt Wake-up from SLEEP	PC + 1 <sup>(1)</sup>	uuu1 0uuu	---- --uu

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0'.

**Note 1:** When the wake-up is due to an interrupt and global enable bit GIE is set, the PC is loaded with the interrupt vector (0004h) after execution of PC+1.

**TABLE 9-7: INITIALIZATION CONDITION FOR REGISTERS**

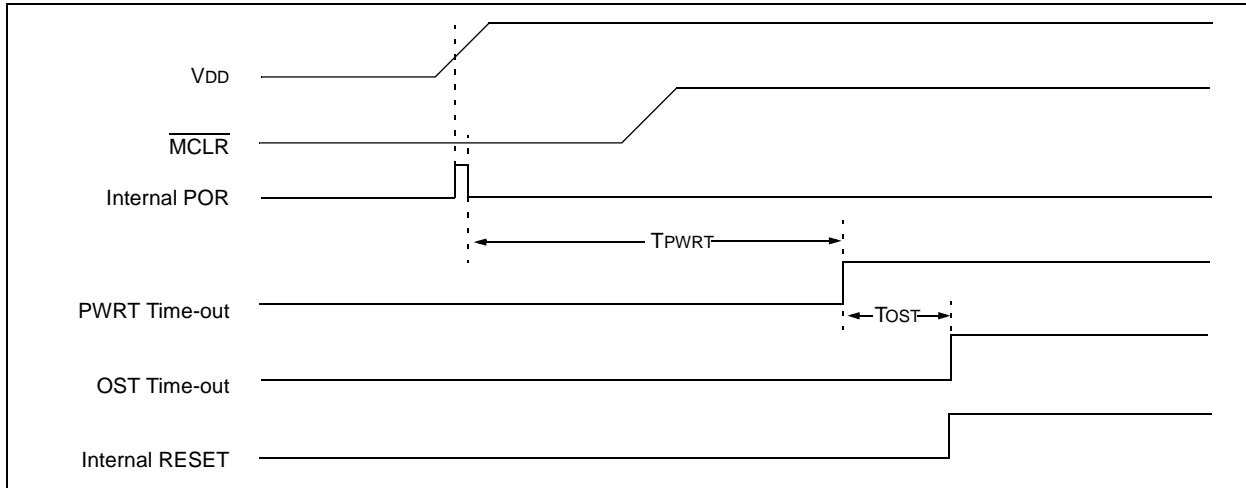
Register	Address	Power-on Reset	<ul style="list-style-type: none"> <li>• MCLR Reset during normal operation</li> <li>• MCLR Reset during SLEEP</li> <li>• WDT Reset</li> <li>• Brown-out Detect<sup>(1)</sup></li> </ul>	<ul style="list-style-type: none"> <li>• Wake-up from SLEEP through interrupt</li> <li>• Wake-up from SLEEP through WDT time-out</li> </ul>
W	—	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	00h/80h	—	—	—
TMR0	01h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02h/82h	0000 0000	0000 0000	PC + 1 <sup>(3)</sup>
STATUS	03h/83h	0001 1xxx	000q quuu <sup>(4)</sup>	uuuq quuu <sup>(4)</sup>
FSR	04h/84h	xxxx xxxx	uuuu uuuu	uuuu uuuu
GPIO	05h	--xx xxxx	--uu uuuu	--uu uuuu
PCLATH	0Ah/8Ah	---0 0000	---0 0000	---u uuuu
INTCON	0Bh/8Bh	0000 0000	0000 000u	uuuu uuq <sup>(2)</sup>
PIR1	0Ch	00-- 0--0	00-- 0--0	qq-- q--q <sup>(2,5)</sup>
T1CON	10h	-000 0000	-uuu uuuu	-uuu uuuu
CMCON	19h	-0-0 0000	-0-0 0000	-u-u uuuu
ADRESH	1Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0	1Fh	00-- 0000	00-- 0000	uu-- uuuu
OPTION_REG	81h	1111 1111	1111 1111	uuuu uuuu
TRISIO	85h	--11 1111	--11 1111	--uu uuuu
PIE1	8Ch	00-- 0--0	00-- 0--0	uu-- u--u
PCON	8Eh	---- --0x	---- --uu <sup>(1,6)</sup>	---- --uu
OSCCAL	90h	1000 00--	1000 00--	uuuu uu--
WPU	95h	--11 -111	--11 -111	uuuu uuuu
IOCB	96h	--00 0000	--00 0000	--uu uuuu
VRCON	99h	0-0- 0000	0-0- 0000	u-u- uuuu
EEDATA	9Ah	0000 0000	0000 0000	uuuu uuuu
EEADR	9Bh	-000 0000	-000 0000	-uuu uuuu
EECON1	9Ch	---- x000	---- q000	---- uuuu
EECON2	9Dh	---- ----	---- ----	---- ----
ADRESL	9Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
ANSEL	9Fh	-000 1111	-000 1111	-uuu uuuu

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

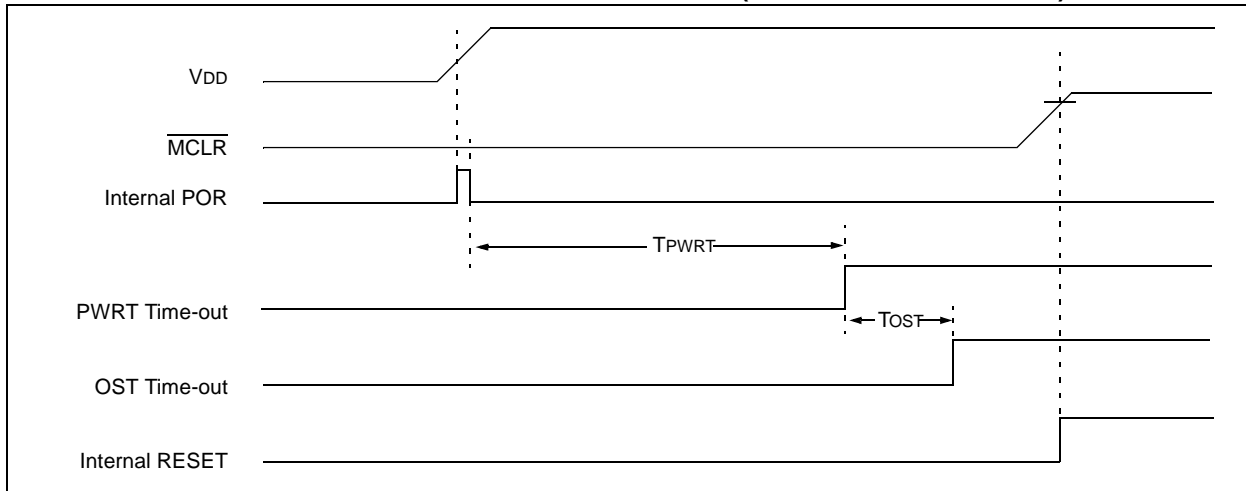
- Note 1:** If VDD goes too low, Power-on Reset will be activated and registers will be affected differently.
- 2:** One or more bits in INTCON and/or PIR1 will be affected (to cause wake-up).
- 3:** When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).
- 4:** See Table 9-6 for RESET value for specific condition.
- 5:** If wake-up was due to data EEPROM write completing, bit 7 = 1; A/D conversion completing, bit 6 = 1; Comparator input changing, bit 3 = 1; or Timer1 rolling over, bit 0 = 1. All other interrupts generating a wake-up will cause these bits to = u.
- 6:** If RESET was due to brown-out, then bit 0 = 0. All other RESETS will cause bit 0 = u.

# PIC12F629/675

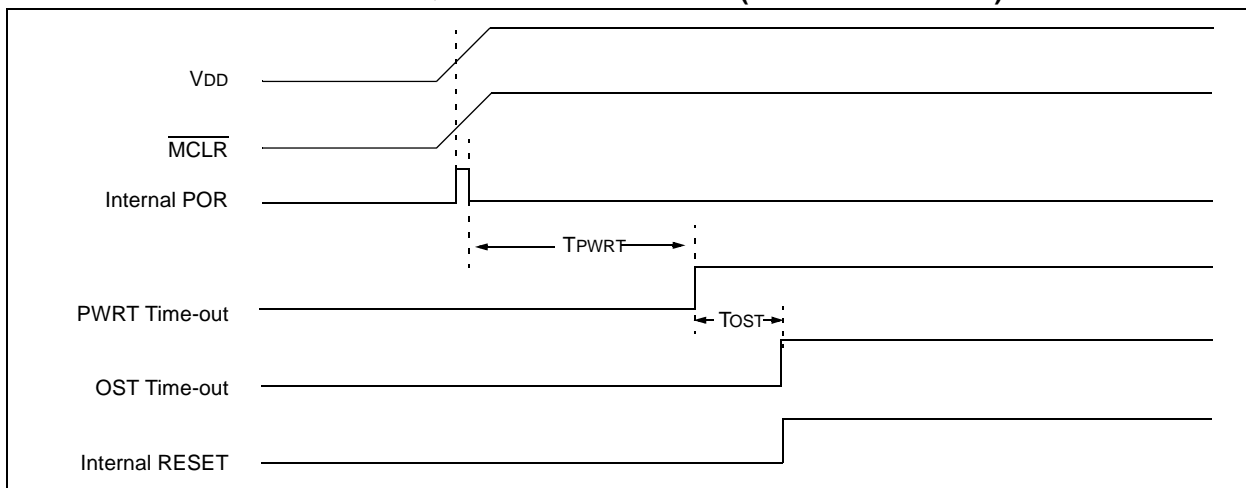
**FIGURE 9-7: TIME-OUT SEQUENCE ON POWER-UP ( $\overline{\text{MCLR}}$  NOT TIED TO  $V_{\text{DD}}$ ): CASE 1**



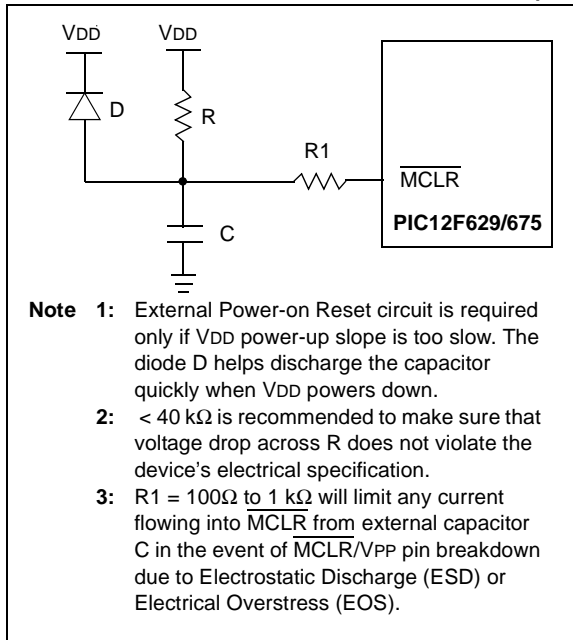
**FIGURE 9-8: TIME-OUT SEQUENCE ON POWER-UP ( $\overline{\text{MCLR}}$  NOT TIED TO  $V_{\text{DD}}$ ): CASE 2**



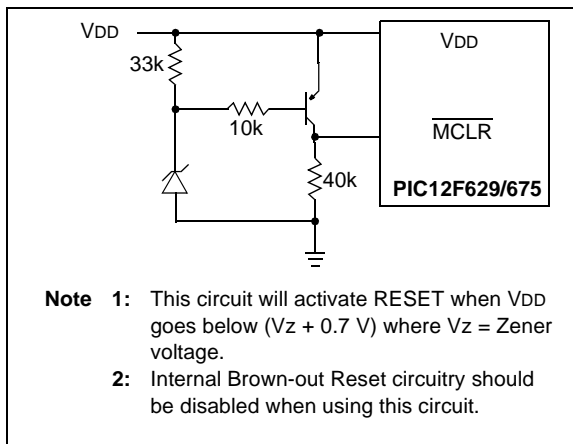
**FIGURE 9-9: TIME-OUT SEQUENCE ON POWER-UP ( $\overline{\text{MCLR}}$  TIED TO  $V_{\text{DD}}$ )**



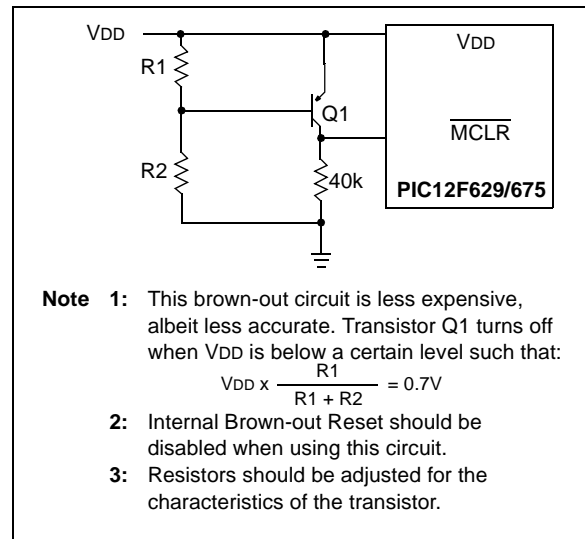
**FIGURE 9-10: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW V<sub>DD</sub> POWER-UP)**



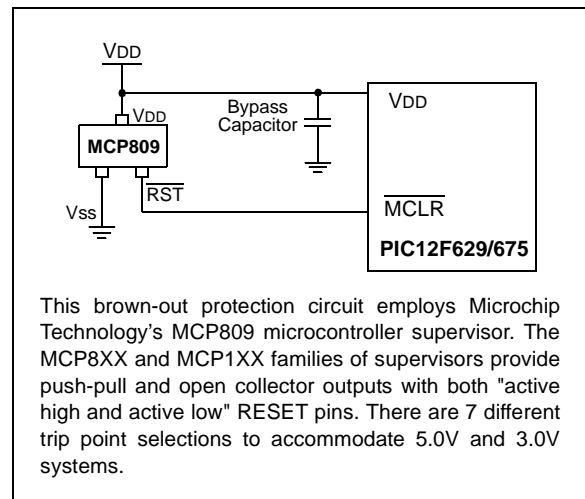
**FIGURE 9-11: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 1**



**FIGURE 9-12: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 2**



**FIGURE 9-13: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 3**



# PIC12F629/675

---

## 9.4 Interrupts

The PIC12F629/675 has 7 sources of interrupt:

- External Interrupt GP2/INT
- TMR0 Overflow Interrupt
- GPIO Change Interrupts
- Comparator Interrupt
- A/D Interrupt (PIC12F675 only)
- TMR1 Overflow Interrupt
- EEPROM Data Write Interrupt

The Interrupt Control register (INTCON) and Peripheral Interrupt register (PIR) record individual interrupt requests in flag bits. It also has individual and global interrupt enable bits.

A global interrupt enable bit, GIE (INTCON<7>) enables (if set) all unmasked interrupts, or disables (if cleared) all interrupts. Individual interrupts can be disabled through their corresponding enable bits in INTCON register and PIE register. GIE is cleared on RESET.

The return from interrupt instruction, `RETFIE`, exits interrupt routine, as well as sets the GIE bit, which re-enables unmasked interrupts.

The following interrupt flags are contained in the INTCON register:

- INT pin interrupt
- GP port change interrupt
- TMR0 overflow interrupt.

The peripheral interrupt flags are contained in the special register PIR1. The corresponding interrupt enable bit is contained in Special Register PIE1.

The following interrupt flags are contained in the PIR register:

- EEPROM data write interrupt
- A/D interrupt
- Comparator interrupt
- Timer1 overflow interrupt

When an interrupt is serviced:

- The GIE is cleared to disable any further interrupt
- The return address is pushed onto the stack
- The PC is loaded with 0004h.

Once in the Interrupt Service Routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid GP2/INT recursive interrupts.

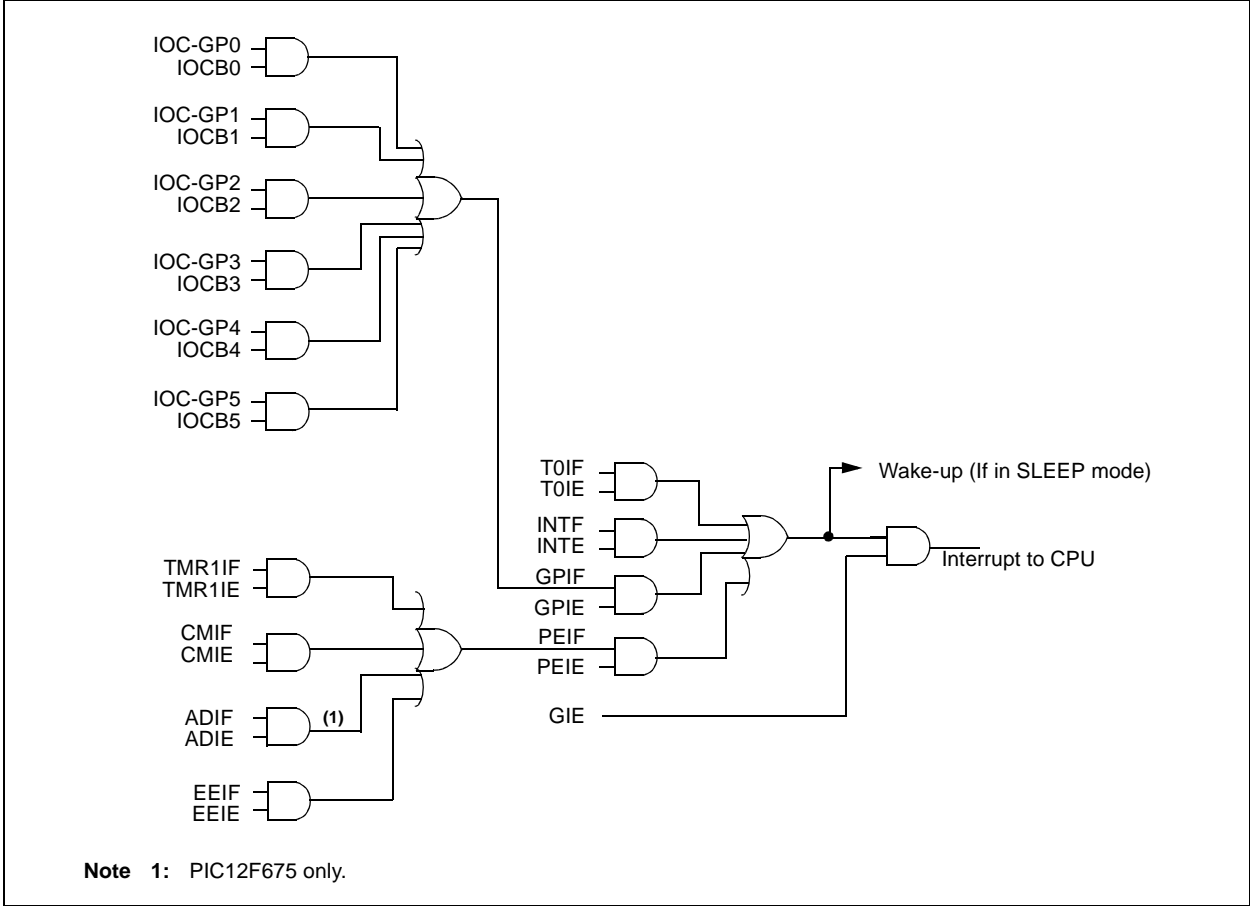
For external interrupt events, such as the INT pin, or GP port change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends upon when the interrupt event occurs (see Figure 9-15). The latency is the same for one or two-cycle instructions. Once in the Interrupt Service Routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid multiple interrupt requests.

**Note 1:** Individual interrupt flag bits are set, regardless of the status of their corresponding mask bit or the GIE bit.

**2:** When an instruction that clears the GIE bit is executed, any interrupts that were pending for execution in the next cycle are ignored. The interrupts which were ignored are still pending to be serviced when the GIE bit is set again.



FIGURE 9-14: INTERRUPT LOGIC



# PIC12F629/675

## 9.4.1 GP2/INT INTERRUPT

External interrupt on GP2/INT pin is edge-triggered; either rising if INTEDG bit (OPTION<6>) is set, or falling, if INTEDG bit is clear. When a valid edge appears on the GP2/INT pin, the INTF bit (INTCON<1>) is set. This interrupt can be disabled by clearing the INTE control bit (INTCON<4>). The INTF bit must be cleared in software in the Interrupt Service Routine before re-enabling this interrupt. The GP2/INT interrupt can wake-up the processor from SLEEP if the INTE bit was set prior to going into SLEEP. The status of the GIE bit decides whether or not the processor branches to the interrupt vector following wake-up. See Section 9.7 for details on SLEEP and Figure 9-17 for timing of wake-up from SLEEP through GP2/INT interrupt.

## 9.4.2 TMR0 INTERRUPT

An overflow (FFh → 00h) in the TMR0 register will set the T0IF (INTCON<2>) bit. The interrupt can be enabled/disabled by setting/clearing T0IE (INTCON<5>) bit. For operation of the Timer0 module, see Section 4.0.

## 9.4.3 GPIO INTERRUPT

An input change on GPIO change sets the GPIF (INTCON<0>) bit. The interrupt can be enabled/disabled by setting/clearing the GPIE (INTCON<3>) bit. Plus individual pins can be configured through the IOCB register.

**Note:** If a change on the I/O pin should occur when the read operation is being executed (start of the Q2 cycle), then the GPIF interrupt flag may not get set.

## 9.4.4 COMPARATOR INTERRUPT

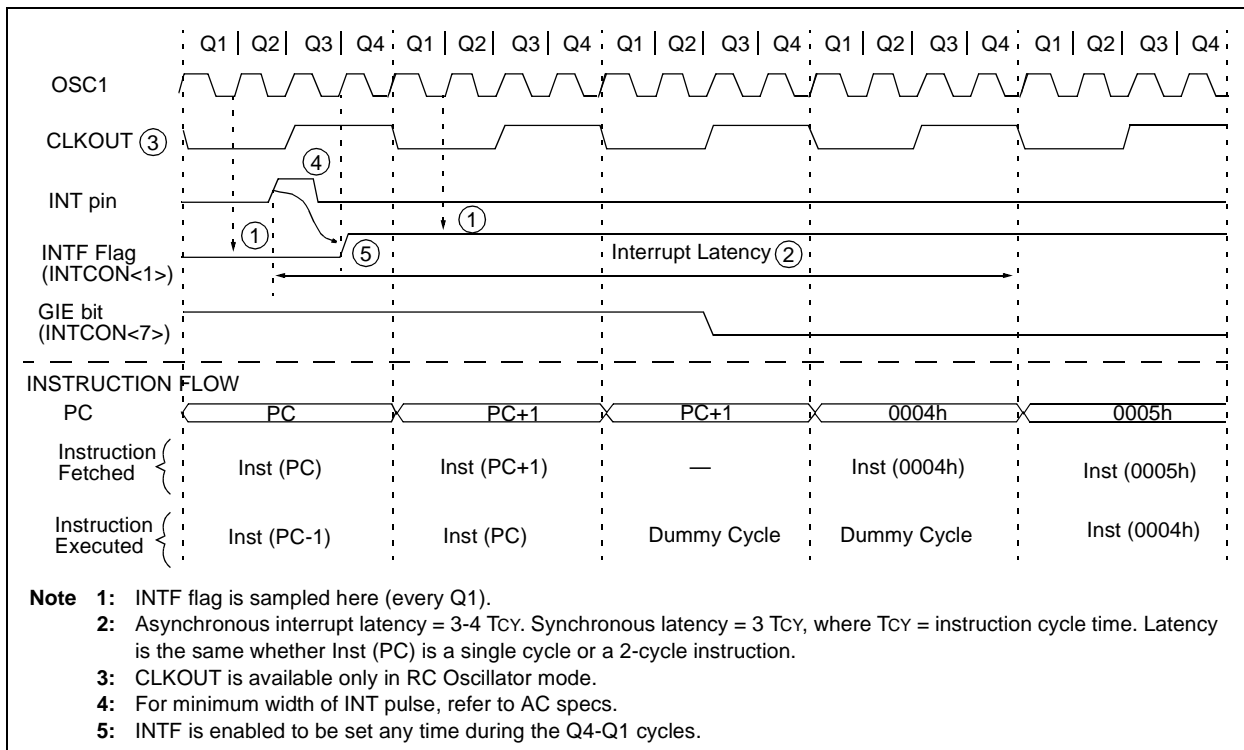
See Section 6.9 for description of comparator interrupt.

## 9.4.5 A/D CONVERTER INTERRUPT

After a conversion is complete, the ADIF flag (PIR<6>) is set. The interrupt can be enabled/disabled by setting or clearing ADIE (PIE<6>).

See Section 7.0 for operation of the A/D converter interrupt.

**FIGURE 9-15: INT PIN INTERRUPT TIMING**



**TABLE 9-8: SUMMARY OF INTERRUPT REGISTERS**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR Reset	Value on all other RESETS
0Bh, 8Bh	INTCON	GIE	PEIE	TOIE	INTE	GPIE	TOIF	INTF	GPIF	0000 0000	0000 000u
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	00-- 0--0
8Ch	PIE1	EEIE	ADIE	—	—	CMIE	—	—	TMR1IE	00-- 0--0	00-- 0--0

Legend: x = unknown, u = unchanged, - = unimplemented read as '0', q = value depends upon condition.  
 Shaded cells are not used by the Interrupt module.

## 9.5 Context Saving During Interrupts

During an interrupt, only the return PC value is saved on the stack. Typically, users may wish to save key registers during an interrupt, e.g., W register and STATUS register. This must be implemented in software.

Example 9-2 stores and restores the STATUS and W registers. The user register, W\_TEMP, must be defined in both banks and must be defined at the same offset from the bank base address (i.e., W\_TEMP is defined at 0x20 in Bank 0 and it must also be defined at 0xA0 in Bank 1). The user register, STATUS\_TEMP, must be defined in Bank 0. The Example 9-2:

- Stores the W register
- Stores the STATUS register in Bank 0
- Executes the ISR code
- Restores the STATUS (and bank select bit register)
- Restores the W register

### EXAMPLE 9-2: SAVING THE STATUS AND W REGISTERS IN RAM

```

MOVWF  W_TEMP      ;copy W to temp register,
                   ;could be in either bank
SWAPF  STATUS,W    ;swap status to be saved into W
BCF    STATUS,RP0  ;change to bank 0 regardless of
                   ;current bank
MOVWF  STATUS_TEMP ;save status to bank 0 register
:
: (ISR)
:
SWAPF  STATUS_TEMP,W;swap STATUS_TEMP register into
                   ;W, sets bank to original state
MOVWF  STATUS      ;move W into STATUS register
SWAPF  W_TEMP,F    ;swap W_TEMP
SWAPF  W_TEMP,W    ;swap W_TEMP into W
    
```

## 9.6 Watchdog Timer (WDT)

The Watchdog Timer is a free running, on-chip RC oscillator, which requires no external components. This RC oscillator is separate from the external RC oscillator of the CLKIN pin. That means that the WDT will run, even if the clock on the OSC1 and OSC2 pins of the device has been stopped (for example, by execution of a SLEEP instruction). During normal operation, a WDT time-out generates a device RESET. If the device is in SLEEP mode, a WDT time-out causes the device to wake-up and continue with normal operation. The WDT can be permanently disabled by programming the configuration bit WDTE as clear (Section 9.1).

### 9.6.1 WDT PERIOD

The WDT has a nominal time-out period of 18 ms, (with no prescaler). The time-out periods vary with temperature, VDD and process variations from part to part (see DC specs). If longer time-out periods are 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, time-out periods up to 2.3 seconds can be realized.

The CLRWDT and SLEEP instructions clear the WDT and the prescaler, if assigned to the WDT, and prevent it from timing out and generating a device RESET.

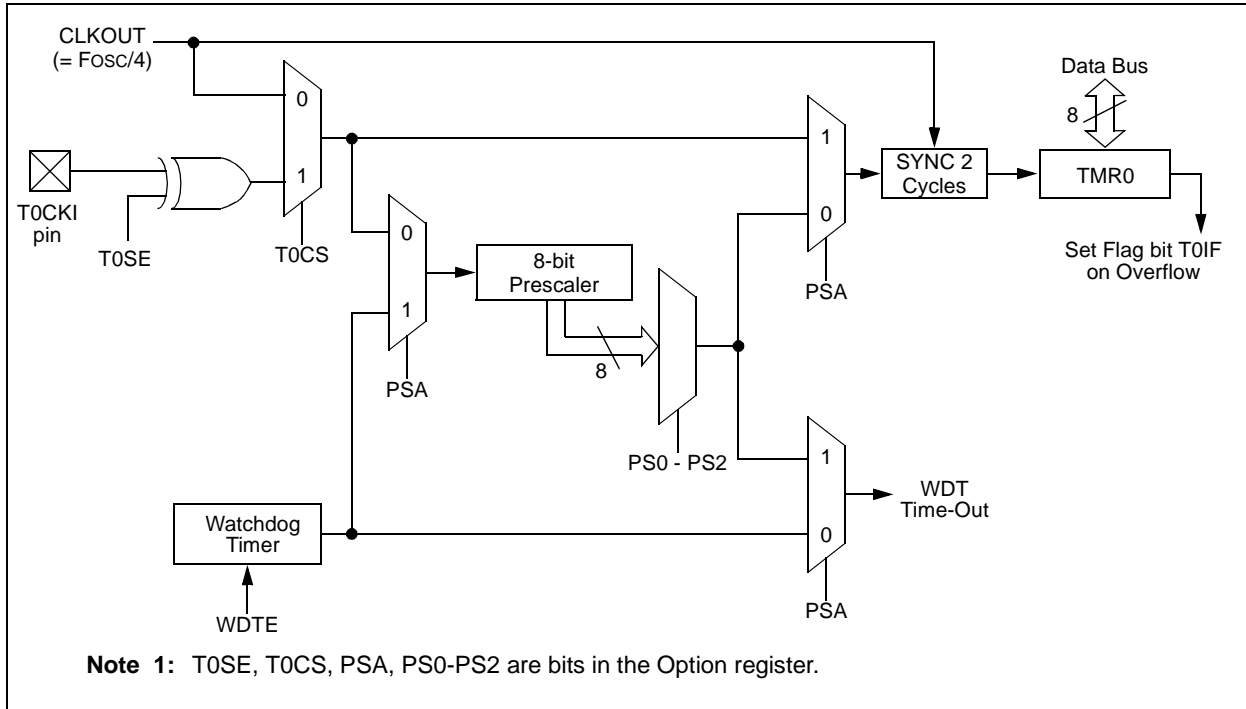
The  $\overline{TO}$  bit in the STATUS register will be cleared upon a Watchdog Timer time-out.

### 9.6.2 WDT PROGRAMMING CONSIDERATIONS

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

# PIC12F629/675

**FIGURE 9-16: WATCHDOG TIMER BLOCK DIAGRAM**



**TABLE 9-9: SUMMARY OF WATCHDOG TIMER REGISTERS**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR Reset	Value on all other RESETS
81h	OPTION_REG	GPPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
2007h	Config. bits	$\overline{CP}$	BODEN	MCLRE	PWRTE	WDTE	F0SC2	F0SC1	F0SC0	uuuu uuuu	uuuu uuuu

Legend: u = Unchanged, shaded cells are not used by the Watchdog Timer.

## 9.7 Power-Down Mode (SLEEP)

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

If the Watchdog Timer is enabled:

- WDT will be cleared but keeps running
- $\overline{PD}$  bit in the STATUS register is cleared
- $\overline{TO}$  bit is set
- Oscillator driver is turned off
- I/O ports maintain the status they had before SLEEP was executed (driving high, low, or hi-impedance).

For lowest current consumption in this mode, all I/O pins should be either at VDD, or VSS, with no external circuitry drawing current from the I/O pin and the comparators and CVREF should be disabled. I/O pins that are hi-impedance inputs should be pulled high or low externally to avoid switching currents caused by floating inputs. The T0CKI input should also be at VDD or VSS for lowest current consumption. The contribution from on chip pull-ups on GPIO should be considered.

The  $\overline{MCLR}$  pin must be at a logic high level ( $V_{IHMC}$ ).

**Note:** It should be noted that a RESET generated by a WDT time-out does not drive  $\overline{MCLR}$  pin low.

The first event will cause a device RESET. The two latter events are considered a continuation of program execution. The  $\overline{TO}$  and  $\overline{PD}$  bits in the STATUS register can be used to determine the cause of device RESET. The  $\overline{PD}$  bit, which is set on power-up, is cleared when SLEEP is invoked.  $\overline{TO}$  bit is cleared if WDT Wake-up occurred.

When the SLEEP instruction is being executed, the next instruction (PC + 1) is pre-fetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set (enabled). Wake-up is regardless of the state of the GIE bit. If the GIE bit is clear (disabled), the device continues execution at the instruction after the SLEEP instruction. If the GIE bit is set (enabled), the device executes the instruction after the SLEEP instruction, then branches to the interrupt address (0004h). In cases where the execution of the instruction following SLEEP is not desirable, the user should have an NOP after the SLEEP instruction.

**Note:** If the global interrupts are disabled (GIE is cleared), but any interrupt source has both its interrupt enable bit and the corresponding interrupt flag bits set, the device will immediately wake-up from SLEEP. The SLEEP instruction is completely executed.

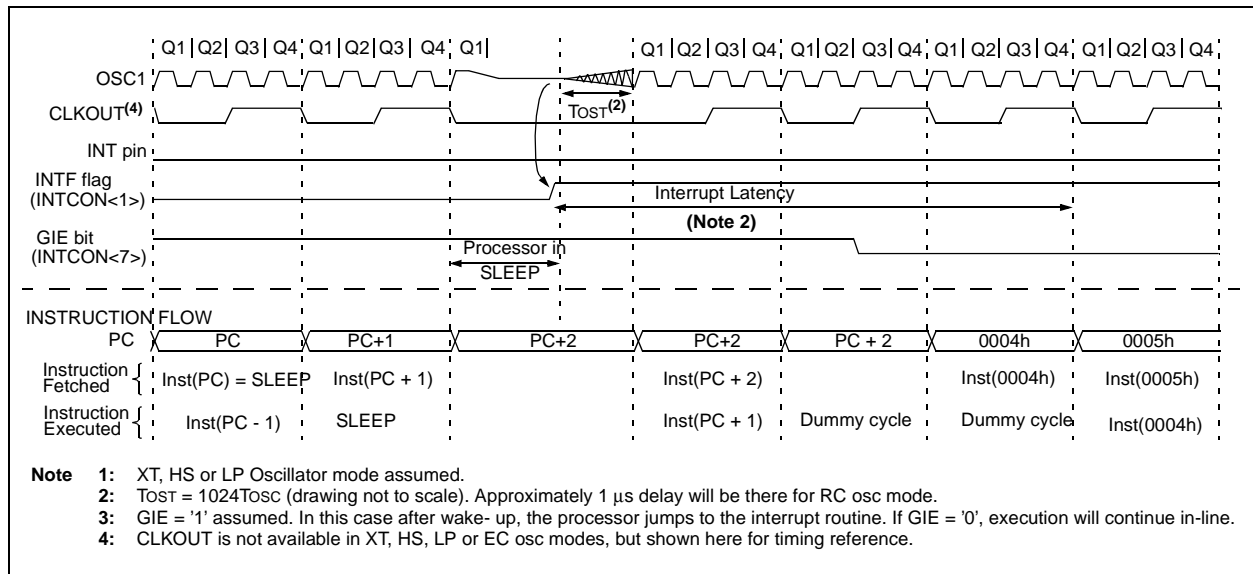
The WDT is cleared when the device wakes up from SLEEP, regardless of the source of wake-up.

### 9.7.1 WAKE-UP FROM SLEEP

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

1. External RESET input on  $\overline{MCLR}$  pin
2. Watchdog Timer Wake-up (if WDT was enabled)
3. Interrupt from GP2/INT pin, GPIO change, or a peripheral interrupt.

**FIGURE 9-17: WAKE-UP FROM SLEEP THROUGH INTERRUPT**



# PIC12F629/675

## 9.8 Code Protection

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

**Note:** The entire data EEPROM and FLASH program memory will be erased when the code protection is turned off. The INTRC calibration data is also erased. See PIC12F629/675 Programming Specification for more information.

## 9.9 ID Locations

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

## 9.10 In-Circuit Serial Programming

The PIC12F629/675 microcontrollers can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data, and three other lines for:

- power
- ground
- programming voltage

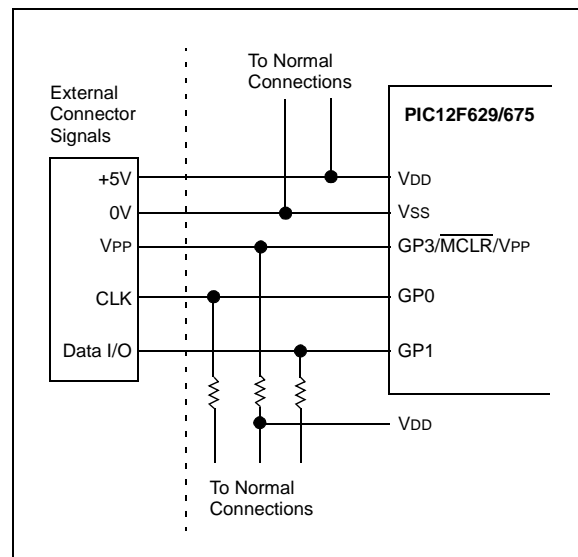
This allows customers to manufacture boards with unprogrammed devices, and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

The device is placed into a Program/Verify mode by holding the GP0 and GP1 pins low, while raising the MCLR (VPP) pin from  $V_{IL}$  to  $V_{IH}$  (see Programming Specification). GP0 becomes the programming clock and GP1 becomes the programming data. Both GP0 and GP1 are Schmitt Trigger inputs in this mode.

After RESET, to place the device into Programming/Verify mode, the program counter (PC) is at location 00h. 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 on whether the command was a load or a read. For complete details of serial programming, please refer to the Programming Specifications.

A typical In-Circuit Serial Programming connection is shown in Figure 9-18.

**FIGURE 9-18: TYPICAL IN-CIRCUIT SERIAL PROGRAMMING CONNECTION**



## 10.0 INSTRUCTION SET SUMMARY

The PIC12F629/675 instruction set is highly orthogonal and is comprised of three basic categories:

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

Each PIC12 instruction is a 14-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 10-1, while the various opcode fields are summarized in Table 10-1.

Table 10-2 lists the instructions recognized by the MPASM™ assembler. A complete description of each instruction is also available in the PICmicro™ Mid-Range Reference Manual (DS33023).

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 zero, the result is placed in the W register. If 'd' is one, 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 bit affected by the operation, while 'f' represents the address of the file in which the bit is located.

For **literal and control** operations, 'k' represents an 8-bit or 11-bit constant, or literal value

One instruction cycle consists of four oscillator periods; for an oscillator frequency of 4 MHz, this gives a normal instruction execution time of 1 μs. 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. When this occurs, the execution takes two instruction cycles, with the second cycle executed as a NOP.

**Note:** To maintain upward compatibility with future products, do not use the `OPTION` and `TRIS` instructions.

All instruction examples use the format '0xhh' to represent a hexadecimal number, where 'h' signifies a hexadecimal digit.

## 10.1 READ-MODIFY-WRITE OPERATIONS

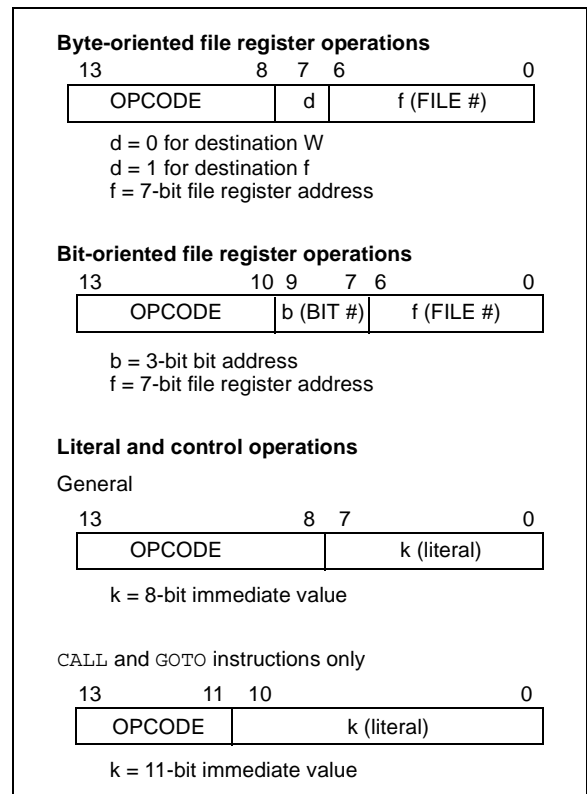
Any instruction that specifies a file register as part of the instruction performs a Read-Modify-Write (R-M-W) operation. The register is read, the data is modified, and the result is stored according to either the instruction, or the destination designator 'd'. A read operation is performed on a register even if the instruction writes to that register.

For example, a `CLRF GPIO` instruction will read GPIO, clear all the data bits, then write the result back to GPIO. This example would have the unintended result that the condition that sets the GPIF flag would be cleared.

**TABLE 10-1: 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.
PC	Program Counter
TO	Time-out bit
PD	Power-down bit

**FIGURE 10-1: GENERAL FORMAT FOR INSTRUCTIONS**



# PIC12F629/675

**TABLE 10-2: PIC12F629/675 INSTRUCTION SET**

Mnemonic, Operands	Description	Cycles	14-Bit Opcode			Status Affected	Notes		
			MSb	LSb					
<b>BYTE-ORIENTED FILE REGISTER OPERATIONS</b>									
ADDWF	f, d	Add W and f	1	00	0111	dfff	ffff	C,DC,Z	1,2
ANDWF	f, d	AND W with f	1	00	0101	dfff	ffff	Z	1,2
CLRF	f	Clear f	1	00	0001	1fff	ffff	Z	2
CLRW	-	Clear W	1	00	0001	0xxx	xxxx	Z	
COMF	f, d	Complement f	1	00	1001	dfff	ffff	Z	1,2
DECf	f, d	Decrement f	1	00	0011	dfff	ffff	Z	1,2
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1,2,3
INCF	f, d	Increment f	1	00	1010	dfff	ffff	Z	1,2
INCFSZ	f, d	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1,2,3
IORWF	f, d	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	1,2
MOVF	f, d	Move f	1	00	1000	dfff	ffff	Z	1,2
MOVWF	f	Move W to f	1	00	0000	1fff	ffff		
NOP	-	No Operation	1	00	0000	0xx0	0000		
RLF	f, d	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	1,2
RRF	f, d	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	1,2
SUBWF	f, d	Subtract W from f	1	00	0010	dfff	ffff	C,DC,Z	1,2
SWAPF	f, d	Swap nibbles in f	1	00	1110	dfff	ffff		1,2
XORWF	f, d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	1,2
<b>BIT-ORIENTED FILE REGISTER OPERATIONS</b>									
BCF	f, b	Bit Clear f	1	01	00bb	bfff	ffff		1,2
BSF	f, b	Bit Set f	1	01	01bb	bfff	ffff		1,2
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01	10bb	bfff	ffff		3
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01	11bb	bfff	ffff		3
<b>LITERAL AND CONTROL OPERATIONS</b>									
ADDLW	k	Add literal and W	1	11	111x	kkkk	kkkk	C,DC,Z	
ANDLW	k	AND literal with W	1	11	1001	kkkk	kkkk	Z	
CALL	k	Call subroutine	2	10	0kkk	kkkk	kkkk		
CLRWDt	-	Clear Watchdog Timer	1	00	0000	0110	0100	$\overline{TO,PD}$	
GOTO	k	Go to address	2	10	1kkk	kkkk	kkkk		
IORLW	k	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
MOVLW	k	Move literal to W	1	11	00xx	kkkk	kkkk		
RETFIE	-	Return from interrupt	2	00	0000	0000	1001		
RETLW	k	Return with literal in W	2	11	01xx	kkkk	kkkk		
RETURN	-	Return from Subroutine	2	00	0000	0000	1000		
SLEEP	-	Go into Standby mode	1	00	0000	0110	0011	$\overline{TO,PD}$	
SUBLW	k	Subtract W from literal	1	11	110x	kkkk	kkkk	C,DC,Z	
XORLW	k	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Z	

- Note 1:** When an I/O register is modified as a function of itself (e.g., `MOVF GPIO, 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'.
- 2:** If this instruction is executed on the TMR0 register (and, where applicable,  $d = 1$ ), the prescaler will be cleared if assigned to the Timer0 module.
- 3:** If Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a `NOP`.

**Note:** Additional information on the mid-range instruction set is available in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023).



## 10.2 Instruction Descriptions

### **ADDLW**      **Add Literal and W**

**Syntax:**            *[label]* ADDLW    *k*  
**Operands:**         $0 \leq k \leq 255$   
**Operation:**         $(W) + k \rightarrow (W)$   
**Status Affected:**    C, DC, Z  
**Description:**        The contents of the W register are added to the eight-bit literal 'k' and the result is placed in the W register.

### **ADDWF**      **Add W and f**

**Syntax:**            *[label]* ADDWF    *f,d*  
**Operands:**         $0 \leq f \leq 127$   
                            $d \in [0,1]$   
**Operation:**         $(W) + (f) \rightarrow (\text{destination})$   
**Status Affected:**    C, DC, Z  
**Description:**        Add 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'.

### **ANDLW**      **AND Literal with W**

**Syntax:**            *[label]* ANDLW    *k*  
**Operands:**         $0 \leq k \leq 255$   
**Operation:**         $(W) .\text{AND.} (k) \rightarrow (W)$   
**Status Affected:**    Z  
**Description:**        The contents of 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:**         $0 \leq f \leq 127$   
                            $d \in [0,1]$   
**Operation:**         $(W) .\text{AND.} (f) \rightarrow (\text{destination})$   
**Status Affected:**    Z  
**Description:**        AND 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'.

### **BCF**            **Bit Clear f**

**Syntax:**            *[label]* BCF      *f,b*  
**Operands:**         $0 \leq f \leq 127$   
                            $0 \leq b \leq 7$   
**Operation:**         $0 \rightarrow (f<b>)$   
**Status Affected:**    None  
**Description:**        Bit 'b' in register 'f' is cleared.

### **BSF**            **Bit Set f**

**Syntax:**            *[label]* BSF      *f,b*  
**Operands:**         $0 \leq f \leq 127$   
                            $0 \leq b \leq 7$   
**Operation:**         $1 \rightarrow (f<b>)$   
**Status Affected:**    None  
**Description:**        Bit 'b' in register 'f' is set.

### **BTFSS**        **Bit Test f, Skip if Set**

**Syntax:**            *[label]* BTFSS    *f,b*  
**Operands:**         $0 \leq f \leq 127$   
                            $0 \leq b < 7$   
**Operation:**        skip if  $(f<b>) = 1$   
**Status Affected:**    None  
**Description:**        If bit 'b' in register 'f' is '0', the next instruction is executed. If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead, making this a 2TCY instruction.

### **BTFSC**        **Bit Test, Skip if Clear**

**Syntax:**            *[label]* BTFSC    *f,b*  
**Operands:**         $0 \leq f \leq 127$   
                            $0 \leq b \leq 7$   
**Operation:**        skip if  $(f<b>) = 0$   
**Status Affected:**    None  
**Description:**        If bit 'b' in register 'f' is '1', the next instruction is executed. If bit 'b', in register 'f', is '0', the next instruction is discarded, and a NOP is executed instead, making this a 2TCY instruction.

# PIC12F629/675

---

---

## **CALL**                      **Call Subroutine**

---

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

## **CLRF**                      **Clear f**

---

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

## **CLRW**                      **Clear W**

---

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

## **CLRWDT**                  **Clear Watchdog Timer**

---

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

## **COMF**                      **Complement f**

---

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

## **DECF**                      **Decrement f**

---

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

---

## DECFSZ      Decrement f, Skip if 0

---

Syntax:      [ *label* ] DECFSZ f,d

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

Operation:     $(f) - 1 \rightarrow (\text{destination})$ ;  
 skip if result = 0

Status Affected: None

Description:    The contents of register 'f' are decremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.  
 If the result is 1, the next instruction is executed. If the result is 0, then a NOP is executed instead, making it a 2TCY instruction.

---

## INCFSZ      Increment f, Skip if 0

---

Syntax:      [ *label* ] INCFSZ f,d

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

Operation:     $(f) + 1 \rightarrow (\text{destination})$ ,  
 skip if result = 0

Status Affected: None

Description:    The contents of register 'f' are incremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.  
 If the result is 1, the next instruction is executed. If the result is 0, a NOP is executed instead, making it a 2TCY instruction.

---

## GOTO      Unconditional Branch

---

Syntax:      [ *label* ] GOTO k

Operands:     $0 \leq k \leq 2047$

Operation:     $k \rightarrow \text{PC}\langle 10:0 \rangle$   
 $\text{PCLATH}\langle 4:3 \rangle \rightarrow \text{PC}\langle 12:11 \rangle$

Status Affected: None

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

---

## IORLW      Inclusive OR Literal with W

---

Syntax:      [ *label* ] IORLW k

Operands:     $0 \leq k \leq 255$

Operation:     $(W) .OR. k \rightarrow (W)$

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.

---

## INCF      Increment f

---

Syntax:      [ *label* ] INCF f,d

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

Operation:     $(f) + 1 \rightarrow (\text{destination})$

Status Affected: Z

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

---

## IORWF      Inclusive OR W with f

---

Syntax:      [ *label* ] IORWF f,d

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

Operation:     $(W) .OR. (f) \rightarrow (\text{destination})$

Status Affected: Z

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

# PIC12F629/675

---

---

## **MOVF**      **Move f**

---

Syntax:      [ *label* ] MOVF f,d  
Operands:     $0 \leq f \leq 127$   
               $d \in [0,1]$   
Operation:    (f) → (destination)  
Status Affected: Z  
Description:    The contents of register f are moved to a destination dependant upon the status of d. If d = 0, destination is W register. If d = 1, the destination is file register f itself. d = 1 is useful to test a file register, since status flag Z is affected.

## **NOP**      **No Operation**

---

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

## **MOVLW**      **Move Literal to W**

---

Syntax:      [ *label* ] MOVLW k  
Operands:     $0 \leq k \leq 255$   
Operation:     $k \rightarrow (W)$   
Status Affected: None  
Description:    The eight-bit literal 'k' is loaded into W register. The don't cares will assemble as 0's.

## **RETFIE**      **Return from Interrupt**

---

Syntax:      [ *label* ] RETFIE  
Operands:    None  
Operation:    TOS → PC,  
              1 → GIE  
Status Affected: None

## **MOVWF**      **Move W to f**

---

Syntax:      [ *label* ] MOVWF f  
Operands:     $0 \leq f \leq 127$   
Operation:    (W) → (f)  
Status Affected: None  
Description:    Move data from W register to register 'f'.

## **RETLW**      **Return with Literal in W**

---

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

## RLF Rotate Left f through Carry

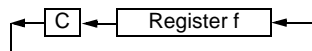
**Syntax:** [ *label* ] RLF f,d

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

**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'.



## SLEEP

**Syntax:** [ *label* ] SLEEP

**Operands:** None

**Operation:** 00h → WDT,  
 0 → WDT prescaler,  
 1 →  $\overline{TO}$ ,  
 0 →  $\overline{PD}$

**Status Affected:**  $\overline{TO}$ ,  $\overline{PD}$

**Description:** The power-down status bit,  $\overline{PD}$  is cleared. Time-out status bit,  $\overline{TO}$  is set. Watchdog Timer and its prescaler are cleared. The processor is put into SLEEP mode with the oscillator stopped.

## RETURN Return from Subroutine

**Syntax:** [ *label* ] RETURN

**Operands:** None

**Operation:** TOS → PC

**Status Affected:** None

**Description:** Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two-cycle instruction.

## SUBLW Subtract W from Literal

**Syntax:** [ *label* ] SUBLW k

**Operands:**  $0 \leq k \leq 255$

**Operation:**  $k - (W) \rightarrow (W)$

**Status Affected:** C, DC, Z

**Description:** The W register is subtracted (2's complement method) from the eight-bit literal 'k'. The result is placed in the W register.

## RRF Rotate Right f through Carry

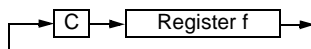
**Syntax:** [ *label* ] RRF f,d

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

**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'.



## SUBWF Subtract W from f

**Syntax:** [ *label* ] SUBWF f,d

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

**Operation:**  $(f) - (W) \rightarrow (\text{destination})$

**Status Affected:** C, DC, Z

**Description:** Subtract (2's complement method) 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'.

# PIC12F629/675

---

---

## **SWAPF**      **Swap Nibbles in f**

---

Syntax:      [*label*] SWAPF f,d

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

Operation:    ( $f\langle 3:0 \rangle \rightarrow$  (destination $\langle 7:4 \rangle$ ),  
                  ( $f\langle 7:4 \rangle \rightarrow$  (destination $\langle 3:0 \rangle$ ))

Status Affected: None

Description:    The upper and lower nibbles of register 'f' are exchanged. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed in register 'f'.

## **XORWF**      **Exclusive OR W with f**

---

Syntax:      [*label*] XORWF f,d

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

Operation:    (W) .XOR. (f)  $\rightarrow$  (destination)

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'.

## **XORLW**      **Exclusive OR Literal with W**

---

Syntax:      [*label*] XORLW k

Operands:     $0 \leq k \leq 255$

Operation:    (W) .XOR. k  $\rightarrow$  (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.

## 11.0 DEVELOPMENT SUPPORT

The PICmicro<sup>®</sup> microcontrollers are supported with a full range of hardware and software development tools:

- Integrated Development Environment
  - MPLAB<sup>®</sup> IDE Software
- Assemblers/Compilers/Linkers
  - MPASM<sup>™</sup> Assembler
  - MPLAB C17 and MPLAB C18 C Compilers
  - MPLINK<sup>™</sup> Object Linker/  
MPLIB<sup>™</sup> Object Librarian
- Simulators
  - MPLAB SIM Software Simulator
- Emulators
  - MPLAB ICE 2000 In-Circuit Emulator
  - ICEPIC<sup>™</sup> In-Circuit Emulator
- In-Circuit Debugger
  - MPLAB ICD
- Device Programmers
  - PRO MATE<sup>®</sup> II Universal Device Programmer
  - PICSTART<sup>®</sup> Plus Entry-Level Development Programmer
- Low Cost Demonstration Boards
  - PICDEM<sup>™</sup> 1 Demonstration Board
  - PICDEM 2 Demonstration Board
  - PICDEM 3 Demonstration Board
  - PICDEM 17 Demonstration Board
  - KEELOQ<sup>®</sup> Demonstration Board

### 11.1 MPLAB Integrated Development Environment Software

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

- An interface to debugging tools
  - simulator
  - programmer (sold separately)
  - emulator (sold separately)
  - in-circuit debugger (sold separately)
- A full-featured editor
- A project manager
- Customizable toolbar and key mapping
- A status bar
- On-line help

The MPLAB IDE allows you to:

- Edit your source files (either assembly or 'C')
- One touch assemble (or compile) and download to PICmicro emulator and simulator tools (automatically updates all project information)
- Debug using:
  - source files
  - absolute listing file
  - machine code

The ability to use MPLAB IDE with multiple debugging tools allows users to easily switch from the cost-effective simulator to a full-featured emulator with minimal retraining.

### 11.2 MPASM Assembler

The MPASM assembler is a full-featured universal macro assembler for all PICmicro MCU's.

The MPASM assembler has a command line interface and a Windows shell. It can be used as a stand-alone application on a Windows 3.x or greater system, or it can be used through MPLAB IDE. The MPASM assembler generates relocatable object files for the MPLINK object linker, Intel<sup>®</sup> standard HEX files, MAP files to detail memory usage and symbol reference, an absolute LST file that contains source lines and generated machine code, and a COD file for debugging.

The MPASM assembler features include:

- Integration into MPLAB IDE projects.
- User-defined macros to streamline assembly code.
- Conditional assembly for multi-purpose source files.
- Directives that allow complete control over the assembly process.

### 11.3 MPLAB C17 and MPLAB C18 C Compilers

The MPLAB C17 and MPLAB C18 Code Development Systems are complete ANSI 'C' compilers for Microchip's PIC17CXXX and PIC18CXXX family of microcontrollers, respectively. These compilers provide powerful integration capabilities and ease of use not found with other compilers.

For easier source level debugging, the compilers provide symbol information that is compatible with the MPLAB IDE memory display.

## 11.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK object linker combines relocatable objects created by the MPASM assembler and the MPLAB C17 and MPLAB C18 C compilers. It can also link relocatable objects from pre-compiled libraries, using directives from a linker script.

The MPLIB object librarian is a librarian for pre-compiled code to be used with the MPLINK object linker. When a routine from a library is called from another 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 MPLIB object librarian manages the creation and modification of library files.

The MPLINK object linker features include:

- Integration with MPASM assembler and MPLAB C17 and MPLAB C18 C compilers.
- Allows all memory areas to be defined as sections to provide link-time flexibility.

The MPLIB object librarian features include:

- Easier linking because single libraries can be included instead of many smaller files.
- Helps keep code maintainable by grouping related modules together.
- Allows libraries to be created and modules to be added, listed, replaced, deleted or extracted.

## 11.5 MPLAB SIM Software Simulator

The MPLAB SIM software simulator allows code development in a PC-hosted environment by simulating the PICmicro series microcontrollers on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a file, or user-defined key press, to any of the pins. The execution can be performed in single step, execute until break, or trace mode.

The MPLAB SIM simulator fully supports symbolic debugging using the MPLAB C17 and the MPLAB C18 C compilers and the MPASM assembler. The software simulator offers the flexibility to develop and debug code outside of the laboratory environment, making it an excellent multi-project software development tool.

## 11.6 MPLAB ICE High Performance Universal In-Circuit Emulator with MPLAB IDE

The MPLAB ICE universal in-circuit emulator is intended to provide the product development engineer with a complete microcontroller design tool set for PICmicro microcontrollers (MCUs). Software control of the MPLAB ICE in-circuit emulator is provided by the MPLAB Integrated Development Environment (IDE), which allows editing, building, downloading and source debugging from a single environment.

The MPLAB ICE 2000 is a full-featured emulator system with enhanced trace, trigger and data monitoring features. Interchangeable processor modules allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the MPLAB ICE in-circuit emulator allows expansion to support new PICmicro microcontrollers.

The MPLAB ICE in-circuit emulator system has been designed as a real-time emulation system, with advanced features that are generally found on more expensive development tools. The PC platform and Microsoft® Windows environment were chosen to best make these features available to you, the end user.

## 11.7 ICEPIC In-Circuit Emulator

The ICEPIC low cost, in-circuit emulator is a solution for the Microchip Technology PIC16C5X, PIC16C6X, PIC16C7X and PIC16CXXX families of 8-bit One-Time-Programmable (OTP) microcontrollers. The modular system can support different subsets of PIC16C5X or PIC16CXXX products through the use of interchangeable personality modules, or daughter boards. The emulator is capable of emulating without target application circuitry being present.



## 11.8 MPLAB ICD In-Circuit Debugger

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

## 11.9 PRO MATE II Universal Device Programmer

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

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

## 11.10 PICSTART Plus Entry Level Development Programmer

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

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

## 11.11 PICDEM 1 Low Cost PICmicro Demonstration Board

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

## 11.12 PICDEM 2 Low Cost PIC16CXX Demonstration Board

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

## 11.13 PICDEM 3 Low Cost PIC16CXXX Demonstration Board

The PICDEM 3 demonstration board is a simple demonstration board that supports the PIC16C923 and PIC16C924 in the PLCC package. It will also support future 44-pin PLCC microcontrollers with an LCD Module. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM 3 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer with an adapter socket, and easily test firmware. The MPLAB ICE in-circuit emulator may also be used with the PICDEM 3 demonstration board to test firmware. A prototype area has been provided to the user for adding hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push button switches, a potentiometer for simulated analog input, a thermistor and separate headers for connection to an external LCD module and a keypad. Also provided on the PICDEM 3 demonstration board is a LCD panel, with 4 commons and 12 segments, that is capable of displaying time, temperature and day of the week. The PICDEM 3 demonstration board provides an additional RS-232 interface and Windows software for showing the demultiplexor LCD signals on a PC. A simple serial interface allows the user to construct a hardware demultiplexer for the LCD signals.

## 11.14 PICDEM 17 Demonstration Board

The PICDEM 17 demonstration board is an evaluation board that demonstrates the capabilities of several Microchip microcontrollers, including PIC17C752, PIC17C756A, PIC17C762 and PIC17C766. All necessary hardware is included to run basic demo programs, which are supplied on a 3.5-inch disk. A programmed sample is included and the user may erase it and program it with the other sample programs using the PRO MATE II device programmer, or the PICSTART Plus development programmer, and easily debug and test the sample code. In addition, the PICDEM 17 demonstration board supports downloading of programs to and executing out of external FLASH memory on board. The PICDEM 17 demonstration board is also usable with the MPLAB ICE in-circuit emulator, or the PICMASTER emulator and all of the sample programs can be run and modified using either emulator. Additionally, a generous prototype area is available for user hardware.

## 11.15 KEELOQ Evaluation and Programming Tools

KEELOQ evaluation and programming tools support Microchip's HCS Secure Data Products. The HCS evaluation kit includes a LCD display to show changing codes, a decoder to decode transmissions and a programming interface to program test transmitters.

**TABLE 11-1: DEVELOPMENT TOOLS FROM MICROCHIP**

Development Tool	PIC12CXXXX	PIC14000	PIC16C5X	PIC16C6X	PIC16CXX	PIC16C7X	PIC16C8X	PIC16F8XX	PIC16C9XX	PIC17C4X	PIC17C7XX	PIC18CXX2	PIC18FXX	24CXX/ 25CXX/ 93CXX	HCSXX	MCRFXXX	MCP2510
<b>Software Tools</b>																	
MPLAB® Integrated Development Environment	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MPLAB® C17 C Compiler																	
MPLAB® C18 C Compiler																	
MPASM™ Assembler/ MPLINK™ Object Linker	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MPLAB® ICE In-Circuit Emulator	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
ICEPIC™ In-Circuit Emulator	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
<b>Debugger</b>																	
MPLAB® ICD In-Circuit Debugger			✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*
<b>Programmers</b>																	
PICSTART® Plus Entry Level Development Programmer	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
PRO MATE® II Universal Device Programmer	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
<b>Demo Boards and Eval Kits</b>																	
PICDEM™ 1 Demonstration Board			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
PICDEM™ 2 Demonstration Board				✓†									✓				
PICDEM™ 3 Demonstration Board								✓									
PICDEM™ 14A Demonstration Board		✓															
PICDEM™ 17 Demonstration Board										✓							
KEELOQ® Evaluation Kit															✓		
KEELOQ® Transponder Kit															✓		
microID™ Programmer's Kit																✓	
125 kHz microID™ Developer's Kit																✓	
125 kHz Anticollision microID™ Developer's Kit																✓	
13.56 MHz Anticollision microID™ Developer's Kit																✓	
MCP2510 CAN Developer's Kit																✓	✓

\* Contact the Microchip Technology Inc. web site at [www.microchip.com](http://www.microchip.com) for information on how to use the MPLAB® ICD In-Circuit Debugger (DV164001) with PIC16C62, 63, 64, 65, 72, 73, 74, 76, 77.

\*\* Contact Microchip Technology Inc. for availability date.

† Development tool is available on select devices.

# PIC12F629/675

---

NOTES:

## 12.0 ELECTRICAL SPECIFICATIONS

### Absolute Maximum Ratings†

Ambient temperature under bias .....	-40 to +125°C
Storage temperature .....	-65°C to +150°C
Voltage on VDD with respect to VSS .....	-0.3 to +6.5V
Voltage on $\overline{\text{MCLR}}$ with respect to VSS .....	-0.3 to +13.5V
Voltage on all other pins with respect to VSS .....	-0.3V to (VDD + 0.3V)
Total power dissipation <sup>(1)</sup> .....	800 mW
Maximum current out of VSS pin .....	300 mA
Maximum current into VDD pin .....	250 mA
Input clamp current, I <sub>IK</sub> (V <sub>I</sub> < 0 or V <sub>I</sub> > VDD) .....	± 20 mA
Output clamp current, I <sub>OK</sub> (V <sub>O</sub> < 0 or V <sub>O</sub> > VDD) .....	± 20 mA
Maximum output current sunk by any I/O pin .....	25 mA
Maximum output current sourced by any I/O pin .....	25 mA
Maximum current sunk by all GPIO .....	125 mA
Maximum current sourced all GPIO .....	125 mA

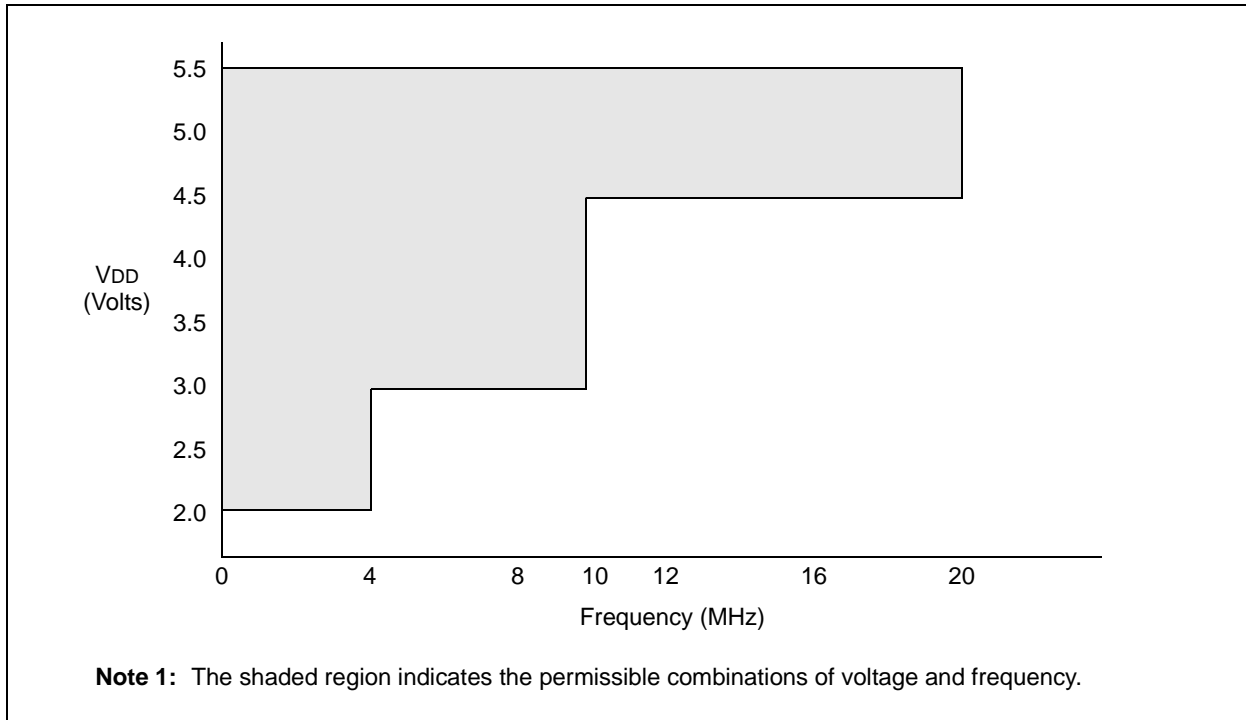
**Note 1:** Power dissipation is calculated as follows:  $P_{DIS} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$ .

† **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.

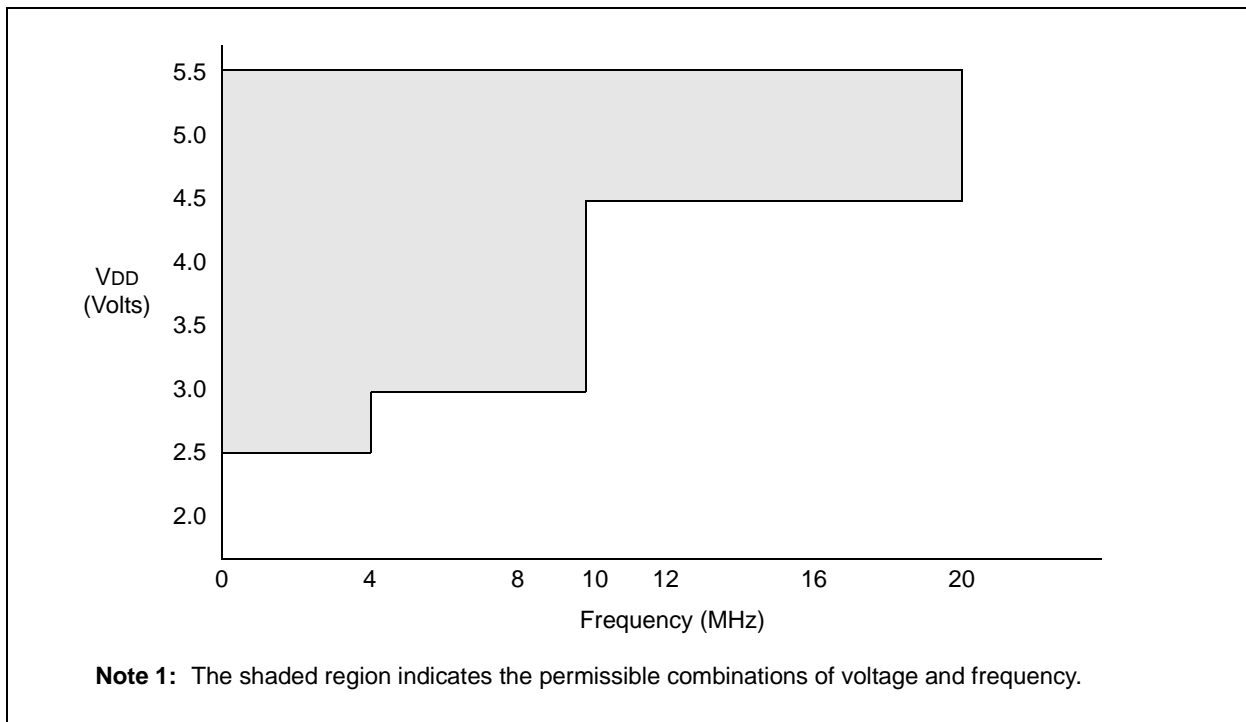
**Note:** Voltage spikes below VSS at the  $\overline{\text{MCLR}}$  pin, inducing currents greater than 80 mA, may cause latchup. Thus, a series resistor of 50-100Ω should be used when applying a "low" level to the  $\overline{\text{MCLR}}$  pin, rather than pulling this pin directly to VSS

# PIC12F629/675

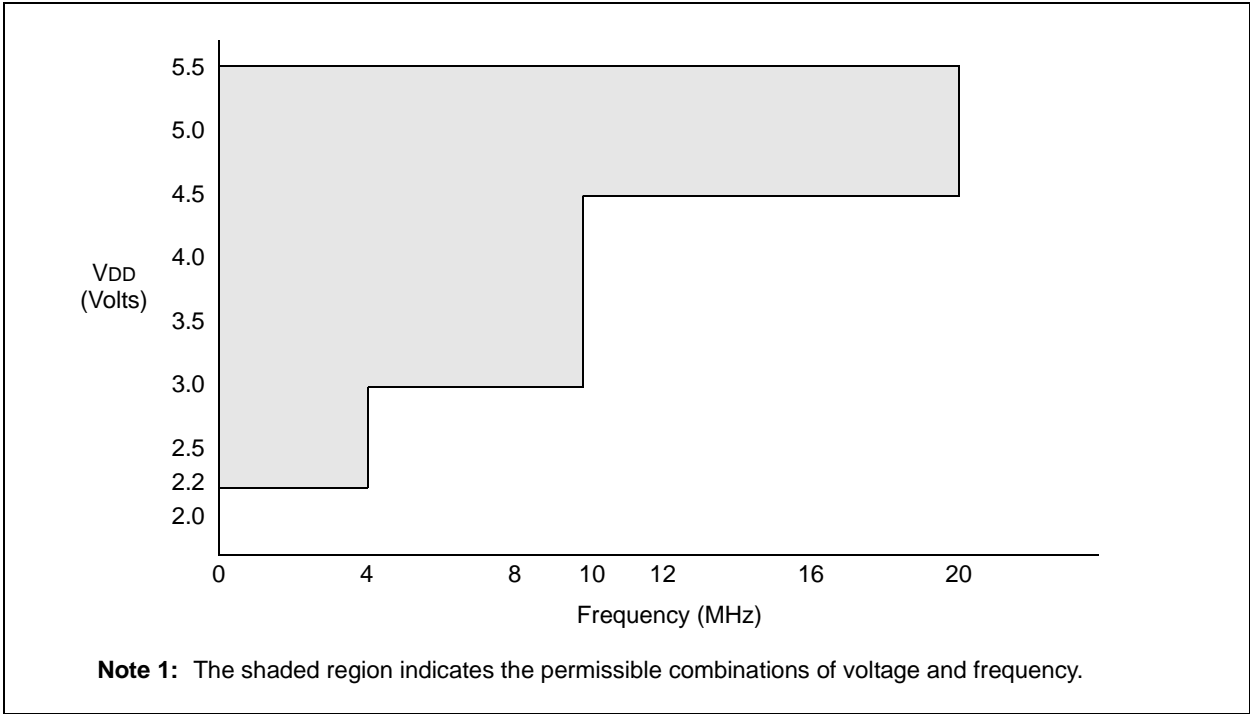
**FIGURE 12-1: PIC12F629/675 WITH A/D DISABLED VOLTAGE-FREQUENCY GRAPH,  $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$**



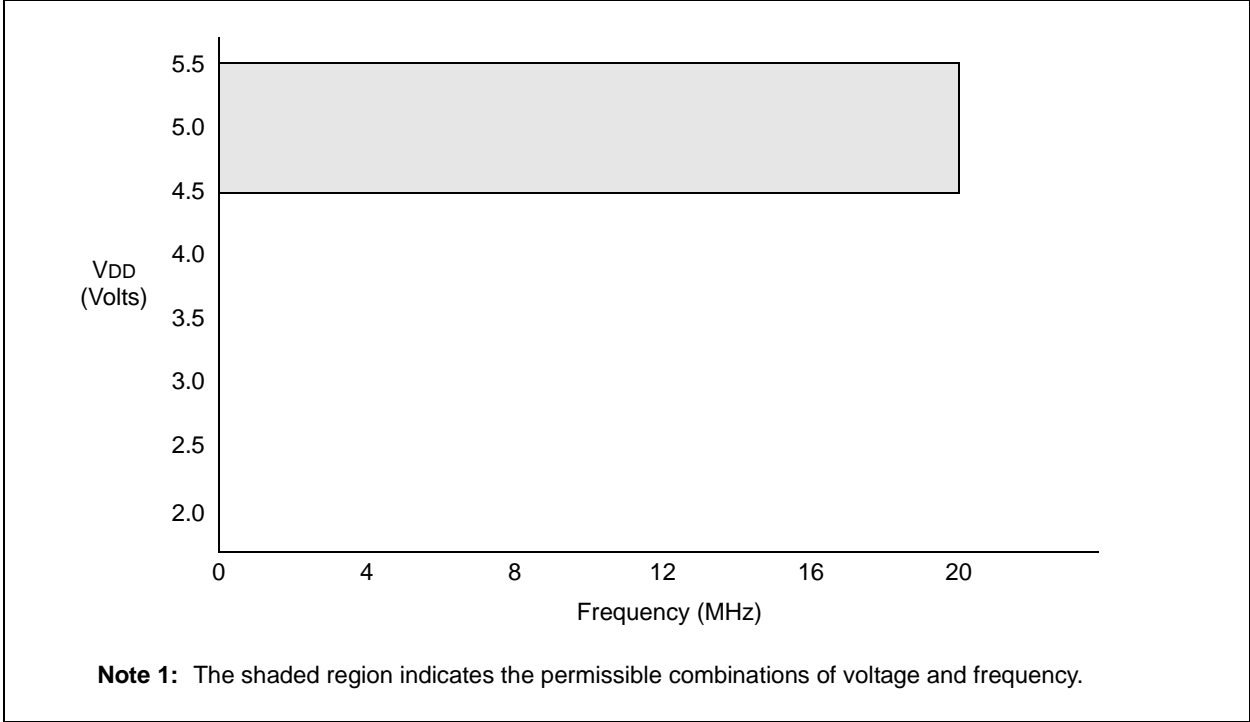
**FIGURE 12-2: PIC12F675 WITH A/D ENABLED VOLTAGE-FREQUENCY GRAPH,  $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$**



**FIGURE 12-3: PIC12F675 WITH A/D ENABLED VOLTAGE-FREQUENCY GRAPH,  $0^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$**



**FIGURE 12-4: PIC12F629/675 VOLTAGE-FREQUENCY GRAPH,  $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$**



# PIC12F629/675

## 12.1 DC Characteristics: PIC12F629/675-I (Industrial)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial				
Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
D001 D001A D001B D001C D001D	VDD	<b>Supply Voltage</b>	2.0 2.2 2.5 3.0 4.5	— — — — —	5.5 5.5 5.5 5.5 5.5	V V V V V	FOSC ≤ 4 MHz: PIC12F629/675 with A/D off PIC12F675 with A/D on, 0°C to 85°C PIC12F675 with A/D on, -40°C to 85°C 4 MHz < FOSC ≤ 10 MHz
D002	VDR	<b>RAM Data Retention Voltage<sup>(1)</sup></b>	1.5*	—	—	V	Device in SLEEP mode
D003	VPOR	<b>VDD Start Voltage</b> to ensure internal Power-on Reset signal	—	VSS	—	V	See section on Power-on Reset for details
D004	SVDD	<b>VDD Rise Rate</b> to ensure internal Power-on Reset signal	0.05*	—	—	V/ms	See section on Power-on Reset for details
D005	VBOR		—	2.0	—	V	
D010 D011 D012 D013	IDD	<b>Supply Current<sup>(2,3)</sup></b>	— — — —	0.4 20 0.9 5.2	2.0 48 4 15	mA μA mA mA	XT, RC osc configurations FOSC = 4 MHz, VDD = 2.0V LP osc configuration FOSC = 32 kHz, VDD = 2.0V, WDT disabled XT, RC osc configurations FOSC = 4 MHz, VDD = 5.5V HS osc configuration FOSC = 20 MHz, VDD = 5.5V
D020 D021 D022 D023 D024 D025 D026	IPD	<b>Power Down Current<sup>(4)</sup></b>	— — — — — — —	0.9 — TBD 1 TBD TBD 5	TBD 153 TBD 18 TBD TBD TBD	μA μA μA μA μA μA μA	VDD = 2.0V, WDT disabled VDD = 5.5V, BOR enabled VDD = 2.0V, Comparator enabled VDD = 2.0V, A/D on, not converting VDD = 2.0V, Timer1 on, 32 kHz ext. drive VDD = 2.0V, CVREF enabled VDD = 2.0V, WDT enabled

\* 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:** This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

**2:** The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

**3:** 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 VDD; MCLR = VDD; WDT disabled.

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



## 12.2 DC Characteristics: PIC12F629/675-E (Extended)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for industrial				
Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
D001A	VDD	<b>Supply Voltage</b>	4.5	—	5.5	V	$-40^{\circ}\text{C}$ to $+125^{\circ}\text{C}$
D002	VDR	<b>RAM Data Retention Voltage<sup>(1)</sup></b>	1.5*	—	—	V	Device in SLEEP mode
D003	VPOR	<b>VDD Start Voltage</b> to ensure internal Power-on Reset signal	—	VSS	—	V	See section on Power-on Reset for details
D004	SVDD	<b>VDD Rise Rate</b> to ensure internal Power-on Reset signal	0.05*	—	—	V/ms	See section on Power-on Reset for details
D005	VBOR		—	2.0	—	V	
D012	IDD	<b>Supply Current<sup>(2,3)</sup></b>	—	0.9	4	mA	XT, RC osc configurations FOSC = 4 MHz, VDD = 5.5V HS osc configuration FOSC = 20 MHz, VDD = 5.5V
D013			—	5.2	15	mA	
D020	IPD	<b>Power Down Current<sup>(4)</sup></b>	—	TBD	TBD	$\mu\text{A}$	VDD = 4.5V, WDT disabled
D021			—	TBD	TBD	$\mu\text{A}$	VDD = 5.0V, BOR enabled
D022			—	TBD	TBD	$\mu\text{A}$	VDD = 4.5V, Comparator enabled
D023			—	TBD	TBD	$\mu\text{A}$	VDD = 4.5V, A/D on, not converting
D024			—	TBD	TBD	$\mu\text{A}$	VDD = 4.5V, Timer1 on, 32 kHz ext. drive
D025			—	TBD	TBD	$\mu\text{A}$	VDD = 4.5V, CVREF enabled
D026			—	12	TBD	$\mu\text{A}$	VDD = 4.5V, WDT enabled

\* 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:** This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

**2:** The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

**3:** 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 VDD; MCLR = VDD; WDT disabled.

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

# PIC12F629/675

## 12.3 DC Characteristics: PIC12F629/675-I (Industrial), PIC12F629/675-E (Extended)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended				
Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
D030 D030A D031 D032 D033 D033A	$V_{IL}$	<b>Input Low Voltage</b> I/O ports with TTL buffer with Schmitt Trigger buffer MCLR, OSC1 (RC mode) OSC1 (XT and LP modes) OSC1 (HS mode)	$V_{SS}$ $V_{SS}$ $V_{SS}$ $V_{SS}$ $V_{SS}$ $V_{SS}$	— — — — — —	0.8 $0.15 V_{DD}$ $0.2 V_{DD}$ $0.2 V_{DD}$ 0.3 $0.3 V_{DD}$	V V V V V V	$4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ Otherwise Entire range <b>(Note 1)</b> <b>(Note 1)</b>
D040 D040A D041 D042 D043 D043A D043B	$V_{IH}$	<b>Input High Voltage</b> I/O ports with TTL buffer with Schmitt Trigger buffer MCLR, GP2/AN2/T0CKI/ INT/COU OSC1 (XT and LP modes) OSC1 (HS mode) OSC1 (RC mode)	2.0 $(0.25 V_{DD} + 0.8)$ $0.8V_{DD}$ $0.8V_{DD}$ 1.6 $0.7V_{DD}$ $0.9V_{DD}$	— — — — — — —	$V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$ $V_{DD}$	V V V V V V V	$4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ otherwise entire range <b>(Note 1)</b> <b>(Note 1)</b>
D070	IPUR	<b>GPIO Weak Pull-up Current</b>	50*	250	400*	$\mu\text{A}$	$V_{DD} = 5.0\text{V}$ , $V_{PIN} = V_{SS}$
D060 D060A D060B D061 D063	$I_{IL}$	<b>Input Leakage Current<sup>(3)</sup></b> I/O ports Analog inputs $V_{REF}$ MCLR <sup>(2)</sup> OSC1	— — — — —	— — — — —	$\pm 1$ $\pm TBD$ $\pm TBD$ $\pm 5$ $\pm 5$	$\mu\text{A}$ $\mu\text{A}$ $\mu\text{A}$ $\mu\text{A}$ $\mu\text{A}$	$V_{SS} \leq V_{PIN} \leq V_{DD}$ , Pin at hi-impedance $V_{SS} \leq V_{PIN} \leq V_{DD}$ $V_{SS} \leq V_{PIN} \leq V_{DD}$ $V_{SS} \leq V_{PIN} \leq V_{DD}$ $V_{SS} \leq V_{PIN} \leq V_{DD}$ , XT, HS and LP osc configuration
D080 D083	$V_{OL}$	<b>Output Low Voltage</b> I/O ports OSC2/CLKOUT	— —	— —	0.6 0.6	V V	$I_{OL} = 8.5\text{mA}$ , $V_{DD} = 4.5\text{V}$ (Ind.) $I_{OL} = 1.6\text{mA}$ , $V_{DD} = 4.5\text{V}$ (Ind.) $I_{OL} = 1.2\text{mA}$ , $V_{DD} = 4.5\text{V}$ (Ext.)
D090 D092	$V_{OH}$	<b>Output High Voltage</b> I/O ports OSC2/CLKOUT	$V_{DD} - 0.7$ $V_{DD} - 0.7$	— —	— —	V V	$I_{OH} = -3.0\text{mA}$ , $V_{DD} = 4.5\text{V}$ (Ind.) $I_{OH} = -1.3\text{mA}$ , $V_{DD} = 4.5\text{V}$ (Ind.) $I_{OH} = -1.0\text{mA}$ , $V_{DD} = 4.5\text{V}$ (Ext.)

\* 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:** In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended to use an external clock in RC mode.
- 2:** The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.
- 3:** Negative current is defined as current sourced by the pin.

# PIC12F629/675

## 12.3 DC Characteristics: PIC12F629/675-I (Industrial), PIC12F629/675-E (Extended) (Cont.)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				
Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
<b>Capacitive Loading Specs on Output Pins</b>							
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	—	—	50*	pF	
D101A	CAN	All analog input pins	—	—	TBD	pF	
D101B	CVR	VREF	—	—	TBD	pF	
<b>Data EEPROM Memory</b>							
D120	ED	Cell Endurance <sup>(1)</sup>	100K	1M	—	E/W	-40°C ≤ TA ≤ +85°C
D120A	ED	Cell Endurance <sup>(1)</sup>	10K	100K	—	E/W	
D121	VDRW	VDD for read	VMIN	—	5.5	V	VMIN = Minimum operating voltage
D122	TDEW	VDD for Erase/Write Erase/Write cycle time	4.5 —	— 4	5.5 8	V ms	
<b>Program FLASH Memory</b>							
D130	EP	Endurance <sup>(1)</sup>	10K	100K	—	E/W	-40°C ≤ TA ≤ +85°C
D130A	EP	Endurance <sup>(1)</sup>	1000	10K	—	E/W	
D131	VPR	VDD for read	VMIN	—	5.5	V	VMIN = Minimum operating voltage
D132	VPEW	VDD for Erase/Write	4.5	—	5.5	V	
D133	TPEW	Erase/Write cycle time	—	2	4	ms	

\* 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:** See Section 8.5.1 for additional information.

# PIC12F629/675

## 12.4 TIMING PARAMETER SYMBOLOGY

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

1. TppS2ppS
2. TppS

<b>T</b>			
F	Frequency	T	Time

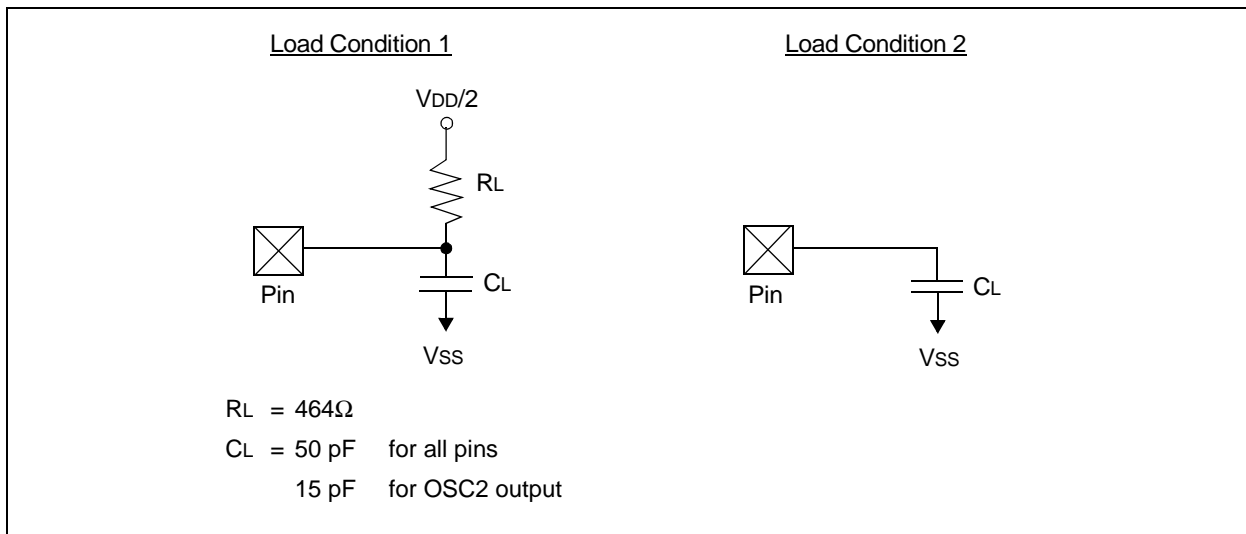
Lowercase letters (pp) and their meanings:

<b>pp</b>			
cc	CCP1	osc	OSC1
ck	CLKOUT	rd	$\overline{RD}$
cs	$\overline{CS}$	rw	$\overline{RD}$ or $\overline{WR}$
di	SDI	sc	SCK
do	SDO	ss	$\overline{SS}$
dt	Data in	t0	T0CKI
io	I/O port	t1	T1CKI
mc	$\overline{MCLR}$	wr	$\overline{WR}$

Uppercase letters and their meanings:

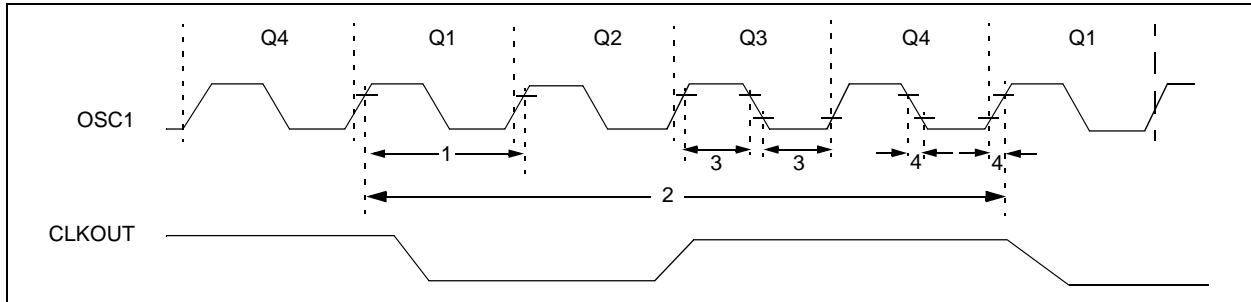
<b>S</b>			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-impedance

**FIGURE 12-5: LOAD CONDITIONS**



## 12.5 AC CHARACTERISTICS: PIC12F629/675 (INDUSTRIAL, EXTENDED)

**FIGURE 12-6: EXTERNAL CLOCK TIMING**



**TABLE 12-1: EXTERNAL CLOCK TIMING REQUIREMENTS**

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
	Fosc	External CLKIN Frequency <sup>(1)</sup>	DC	—	200	kHz	LP osc mode
			DC	—	4	MHz	XT mode
			DC	—	20	MHz	HS mode
			DC	—	20	MHz	EC mode
		Oscillator Frequency <sup>(1)</sup>	5	—	200	kHz	LP osc mode
				4		MHz	INTRC mode
			TBD	—	4	MHz	RC osc mode
			0.1	—	4	MHz	XT osc mode
		1	—	20	MHz	HS osc mode	
1	Tosc	External CLKIN Period <sup>(1)</sup>	5	—	∞	μs	LP osc mode
			50	—	∞	ns	HS osc mode
			50	—	∞	ns	EC osc mode
			250	—	∞	ns	XT osc mode
		Oscillator Period <sup>(1)</sup>	5	—	200	μs	LP osc mode
				250		ns	INTRC mode
		250	—	TBD	ns	RC osc mode	
		250	—	10,000	ns	XT osc mode	
		50	—	1,000	ns	HS osc mode	
2	Tcy	Instruction Cycle Time <sup>(1)</sup>	200	Tcy	DC	ns	Tcy = 4/Fosc
3	TosL, TosH	External CLKIN (OSC1) High External CLKIN Low	2*	—	—	μs	LP oscillator, TOSC L/H duty cycle
			20*	—	—	ns	HS oscillator, TOSC L/H duty cycle
			100*	—	—	ns	XT oscillator, TOSC L/H duty cycle
4	TosR, TosF	External CLKIN Rise External CLKIN Fall	—	—	50*	ns	LP oscillator
			—	—	25*	ns	XT oscillator
			—	—	15*	ns	HS oscillator

\* These parameters are characterized but not tested.

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

**Note 1:** Instruction cycle period (Tcy) equals four times the input oscillator time-base period. 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. All devices are tested to operate at 'min' values with an external clock applied to OSC1 pin. When an external clock input is used, the 'max' cycle time limit is 'DC' (no clock) for all devices.

# PIC12F629/675

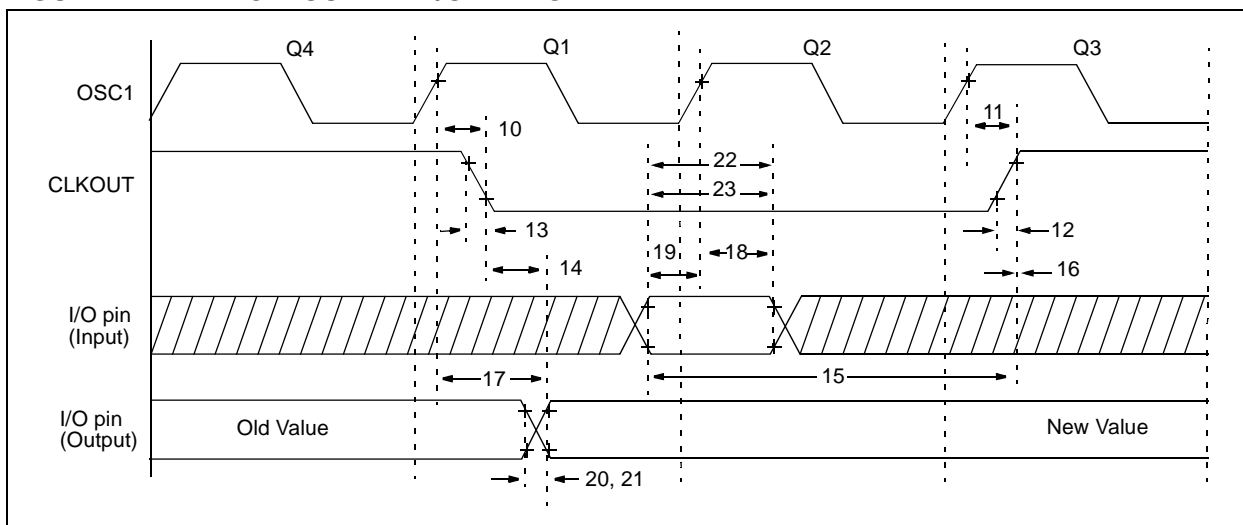
**TABLE 12-2: CALIBRATED INTERNAL RC FREQUENCIES**

AC Characteristics			Standard Operating Conditions (unless otherwise specified)				
			Operating Temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (Industrial), $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (Extended)				
			Operating Voltage $V_{DD}$ range is described in Section 12.1 and Section 12.2.				
Param No.	Sym	Characteristic	Min*	Typ <sup>(1)</sup>	Max*	Units	Conditions
		Internal Calibrated RC Frequency	3.92	4.00	4.08	MHz	$V_{DD} = 5.0\text{V}$ , $+85^{\circ}\text{C}$ (Ind.) $V_{DD} = 5.0\text{V}$ , $+125^{\circ}\text{C}$ (Ext.)
		Internal Calibrated RC Frequency	3.80	4.00	4.20	MHz	$2.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (Ind.) $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (Ext.)

\* These parameters are characterized but not tested.

**Note 1:** Data in the Typical ("Typ") column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**FIGURE 12-7: CLKOUT AND I/O TIMING**



**TABLE 12-3: CLKOUT AND I/O TIMING REQUIREMENTS**

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
10	TosH2ckL	OSC1↑ to CLKOUT↓	—	75	200	ns	(Note 1)
11	TosH2ckH	OSC1↑ to CLKOUT↑	—	75	200	ns	(Note 1)
12	TckR	CLKOUT rise time	—	35	100	ns	(Note 1)
13	TckF	CLKOUT fall time	—	35	100	ns	(Note 1)
14	TckL2ioV	CLKOUT↓ to Port out valid	—	—	20	ns	(Note 1)
15	TioV2ckH	Port in valid before CLKOUT↑	TOSC + 200 ns	—	—	ns	(Note 1)
16	TckH2ioI	Port in hold after CLKOUT↑	0	—	—	ns	(Note 1)
17	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid	—	50	150 *	ns	
			—	—	300	ns	
18	TosH2ioI	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	100	—	—	ns	
19	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)	0	—	—	ns	
20	TioR	Port output rise time	—	10	40	ns	
21	TioF	Port output fall time	—	10	40	ns	
22	Tinp	INT pin high or low time	25	—	—	ns	
23	Trbp	GPIO change INT high or low time	T <sub>CY</sub>	—	—	ns	

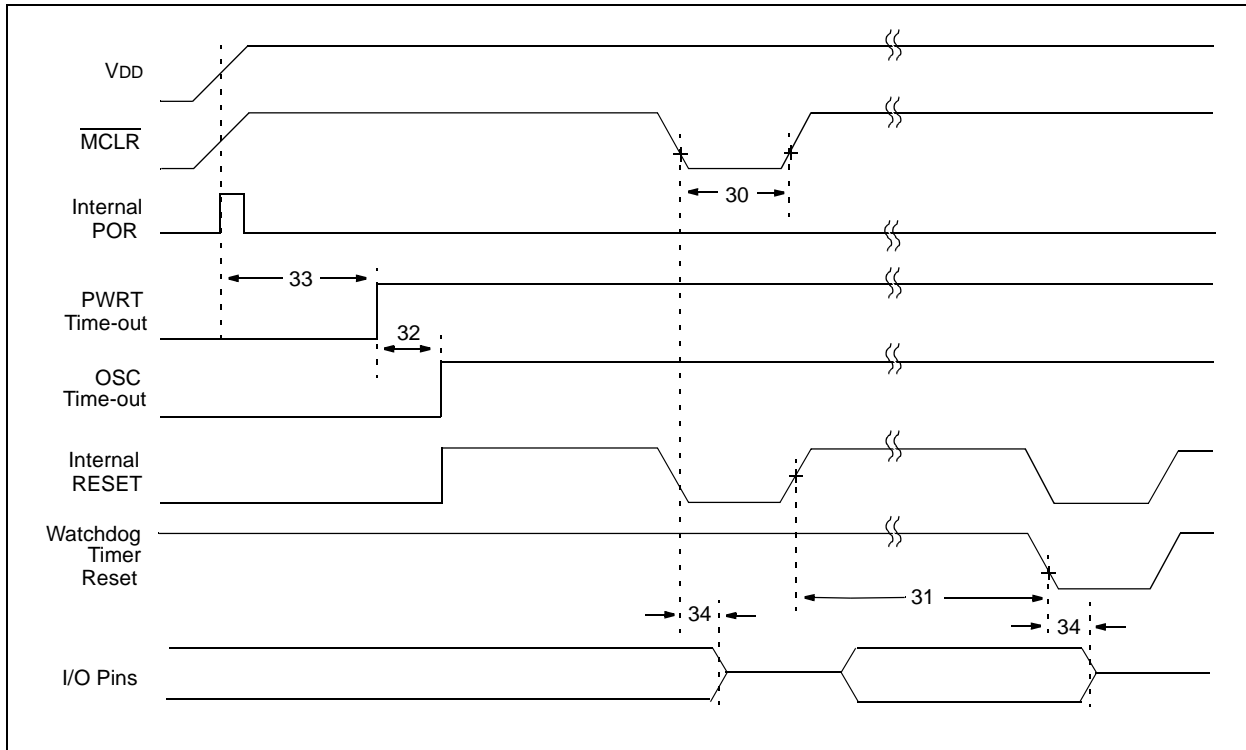
\* These parameters are characterized but not tested.

† Data in 'Typ' column is at 5.0V, 25°C unless otherwise stated.

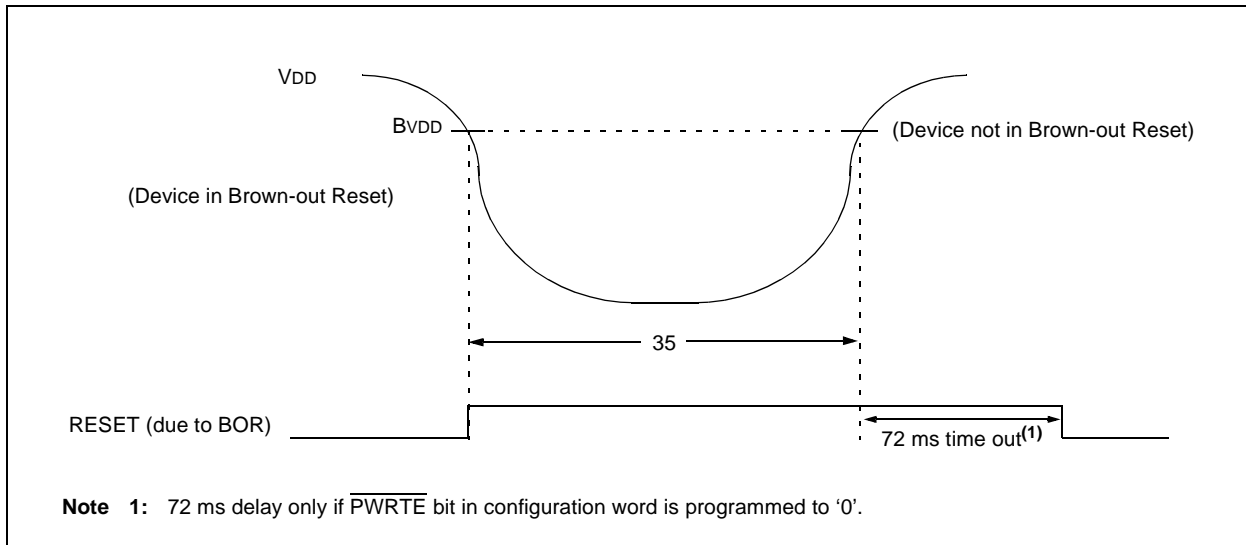
**Note 1:** Measurements are taken in RC mode where CLKOUT output is 4xTOSC.

# PIC12F629/675

**FIGURE 12-8: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING**



**FIGURE 12-9: BROWN-OUT RESET TIMING AND CHARACTERISTICS**





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

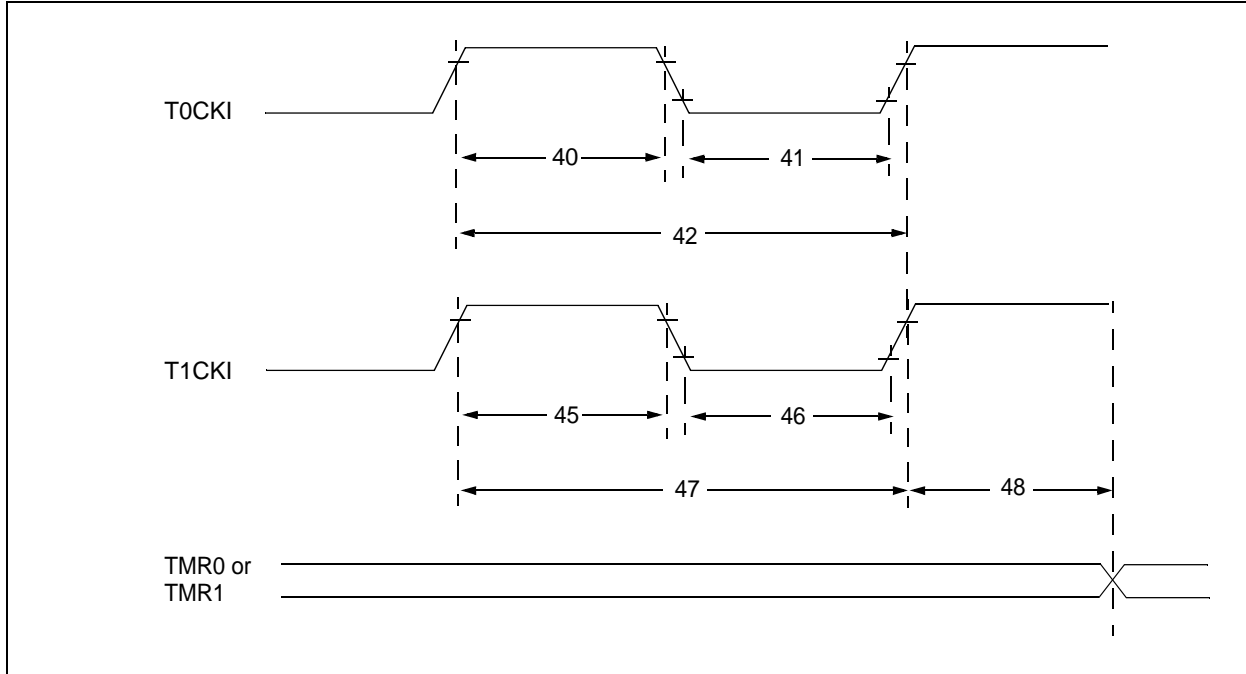
Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30	TMCL	MCLR Pulse Width (low)	2 TBD	— TBD	— TBD	μs ms	VDD = 5V, -40°C to +85°C Extended temperature
31	TWDT	Watchdog Timer Time-out Period (No Prescaler)	7* TBD	18 TBD	33* TBD	ms ms	VDD = 5V, -40°C to +85°C Extended temperature
32	TOST	Oscillation Start-up Timer Period	—	1024Tosc	—	—	Tosc = OSC1 period
33*	TPWRT	Power up Timer Period	28* TBD	72 TBD	132* TBD	ms ms	VDD = 5V, -40°C to +85°C Extended Temperature
34	TIOZ	I/O Hi-impedance from MCLR Low or Watchdog Timer Reset	—	—	2.0	μs	
	BVDD	Brown-out Reset Voltage	2.0		2.1	V	
		Brown-out Hysteresis	TBD				
35	TBOR	Brown-out Reset Pulse Width	100*	—	—	μs	VDD ≤ BVDD (D005)

\* These parameters are characterized but not tested.

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

# PIC12F629/675

**FIGURE 12-10: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS**



**TABLE 12-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS**

Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
40*	Tt0H	T0CKI High Pulse Width	No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			With Prescaler	10	—	—	ns	
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			With Prescaler	10	—	—	ns	
42*	Tt0P	T0CKI Period		Greater of: $20$ or $\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (2, 4, ..., 256)
45*	Tt1H	T1CKI High Time	Synchronous, No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			Synchronous, with Prescaler	15	—	—	ns	
			Asynchronous	30	—	—	ns	
46*	Tt1L	T1CKI Low Time	Synchronous, No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			Synchronous, with Prescaler	15	—	—	ns	
			Asynchronous	30	—	—	ns	
47*	Tt1P	T1CKI Input Period	Synchronous	Greater of: $30$ or $\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1, 2, 4, 8)
			Asynchronous	60	—	—	ns	
	Ft1	Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN)		DC	—	200*	kHz	
48	TCKEZtmr1	Delay from external clock edge to timer increment		$2 T_{OSC}^*$	—	$7 T_{OSC}^*$	—	

\* These parameters are characterized but not tested.

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

**TABLE 12-6: COMPARATOR SPECIFICATIONS**

Comparator Specifications		Standard Operating Conditions -40°C to +125°C (unless otherwise stated)				
Sym	Characteristics	Min	Typ	Max	Units	Comments
VOS	Input Offset Voltage	—	± 5.0	± 10	mV	
VCM	Input Common Mode Voltage	0	—	VDD - 1.5	V	
CMRR	Common Mode Rejection Ratio	+55*	—	—	db	
TRT	Response Time <sup>(1)</sup>	—	150	400*	ns	
TMC2COV	Comparator Mode Change to Output Valid	—	—	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.

**TABLE 12-7: COMPARATOR VOLTAGE REFERENCE SPECIFICATIONS**

Voltage Reference Specifications		Standard Operating Conditions -40°C to +125°C (unless otherwise stated)				
Sym	Characteristics	Min	Typ	Max	Units	Comments
	Resolution	—	VDD/24*	—	LSb	Low Range (VRR = 1)
		—	VDD/32	—	LSb	High Range (VRR = 0)
	Absolute Accuracy	—	—	± 1/4*	LSb	Low Range (VRR = 1)
		—	—	± 1/2*	LSb	High Range (VRR = 0)
	Unit Resistor Value (R)	—	2k*	—	Ω	
	Settling Time <sup>(1)</sup>	—	—	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.

# PIC12F629/675

**TABLE 12-8: PIC12F675 A/D CONVERTER CHARACTERISTICS:**

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
A01	NR	Resolution	—	—	10 bits	bit	
A02	EABS	Total Absolute Error*	—	—	TBD	LSb	VREF = 3.0V
A03	EIL	Integral Error	—	—	TBD	LSb	VREF = 3.0V
A04	EDL	Differential Error	—	—	TBD	LSb	No missing codes to 10 bits VREF = 3.0V
A05	EFS	Full Scale Range	2.2*	—	5.5*	V	
A06	E0FF	Offset Error	—	—	TBD	LSb	VREF = 3.0V
A07	EGN	Gain Error	—	—	TBD	LSb	VREF = 3.0V
A10	—	Monotonicity	—	guaranteed <sup>(3)</sup>	—	—	VSS ≤ VAIN ≤ VREF+
A21	VREF	Reference V High (VDD or VREF)	VSS	—	VDD	V	
A25	VAIN	Analog Input Voltage	VSS	—	VREF	V	
A30	ZAIN	Recommended Impedance of Analog Voltage Source	—	—	2.5	kΩ	
A50	IREF	VREF Input Current <sup>(2)</sup>	10	—	1000	μA	During VAIN acquisition. Based on differential of VHOLD to VAIN. During A/D conversion cycle.
			—	—	10	μA	

\* These parameters are characterized but not tested.

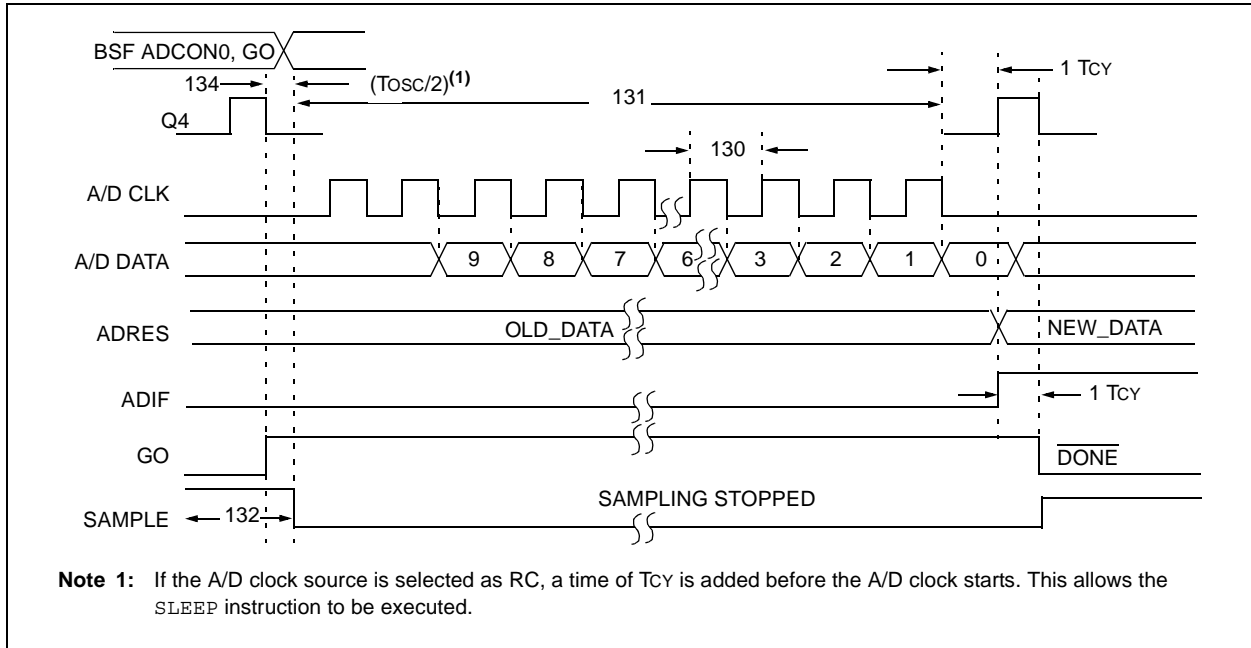
† Data in 'Typ' column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**Note 1:** When A/D is off, it will not consume any current other than leakage current. The power-down current spec includes any such leakage from the A/D module.

**2:** VREF current is from External VREF or VDD pin, whichever is selected as reference input.

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

**FIGURE 12-11: PIC12F675 A/D CONVERSION TIMING (NORMAL MODE)**



**TABLE 12-9: PIC12F675 A/D CONVERSION REQUIREMENTS**

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
130	TAD	A/D Clock Period	1.6	—	—	$\mu\text{s}$	TOSC based, $V_{REF} \geq 3.0\text{V}$
130	TAD	A/D Internal RC Oscillator Period	3.0*	—	—	$\mu\text{s}$	TOSC based, $V_{REF}$ full range
			3.0*	6.0	9.0*	$\mu\text{s}$	ADCS<1:0> = 11 (RC mode) At $V_{DD} = 2.5\text{V}$
			2.0*	4.0	6.0*	$\mu\text{s}$	At $V_{DD} = 5.0\text{V}$
131	TCNV	Conversion Time (not including Acquisition Time) <sup>(1)</sup>	—	11	—	TAD	Set GO bit to new data in A/D result register
132	TACQ	Acquisition Time	(Note 2)	11.5	—	$\mu\text{s}$	The minimum time is the amplifier settling time. This may be used if the "new" input voltage has not changed by more than 1 LSB (i.e., 4.1 mV @ 4.096 V) from the last sampled voltage (as stored on CHOLD).
			5*	—	—	$\mu\text{s}$	
134	TGO	Q4 to A/D Clock Start	—	TOSC/2	—	—	If the A/D clock source is selected as RC, a time of $T_{CY}$ is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

\* These parameters are characterized but not tested.

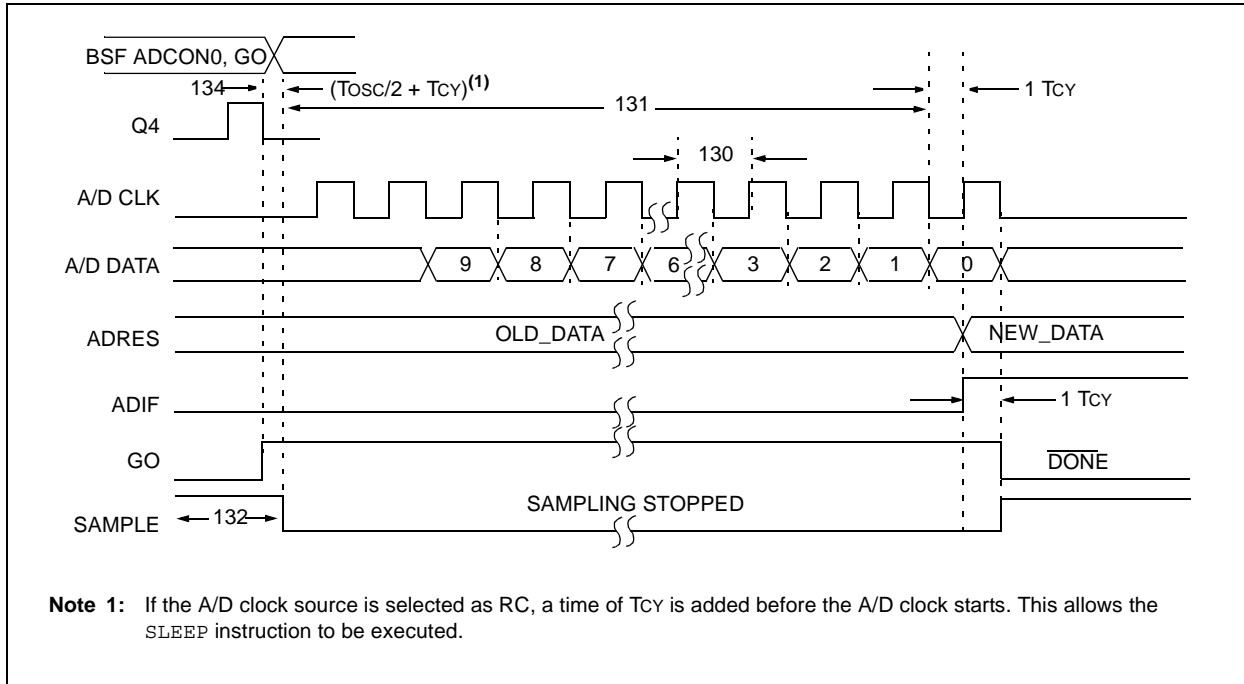
† Data in 'Typ' column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**Note 1:** ADRES register may be read on the following  $T_{CY}$  cycle.

**Note 2:** See Section 7.1 for minimum conditions.

# PIC12F629/675

**FIGURE 12-12: PIC12F675 A/D CONVERSION TIMING (SLEEP MODE)**



**TABLE 12-10: PIC12F675 A/D CONVERSION REQUIREMENTS (SLEEP MODE)**

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
130	TAD	A/D Clock Period	1.6	—	—	$\mu\text{s}$	$V_{REF} \geq 3.0\text{V}$
130	TAD	A/D Internal RC Oscillator Period	3.0*	—	—	$\mu\text{s}$	$V_{REF}$ full range
			3.0*	6.0	9.0*	$\mu\text{s}$	ADCS<1:0> = 11 (RC mode)
			2.0*	4.0	6.0*	$\mu\text{s}$	At $V_{DD} = 2.5\text{V}$
							At $V_{DD} = 5.0\text{V}$
131	T <sub>CV</sub>	Conversion Time (not including Acquisition Time) <sup>(1)</sup>	—	11	—	TAD	
132	T <sub>ACQ</sub>	Acquisition Time	(Note 2)	11.5	—	$\mu\text{s}$	The minimum time is the amplifier settling time. This may be used if the “new” input voltage has not changed by more than 1 LSb (i.e., 4.1 mV @ 4.096V) from the last sampled voltage (as stored on CHOLD).
			5*	—	—	$\mu\text{s}$	
134	T <sub>GO</sub>	Q4 to A/D Clock Start	—	$T_{OSC}/2 + T_{CY}$	—	—	If the A/D clock source is selected as RC, a time of $T_{CY}$ is added before the A/D clock starts. This allows the <code>SLEEP</code> instruction to be executed.

\* 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:** ADRES register may be read on the following  $T_{CY}$  cycle.

**Note 2:** See Section 7.1 for minimum conditions.

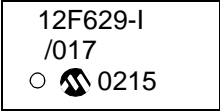
## 13.0 PACKAGING INFORMATION

### 13.1 Package Marking Information

8-Lead PDIP (Skinny DIP)



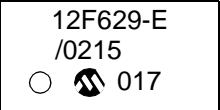
Example



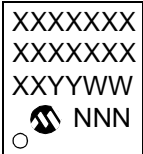
8-Lead SOIC



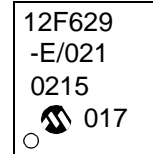
Example



8-Lead MLF-S



Example



<p><b>Legend:</b> 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</p>
<p><b>Note:</b> 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.</p>

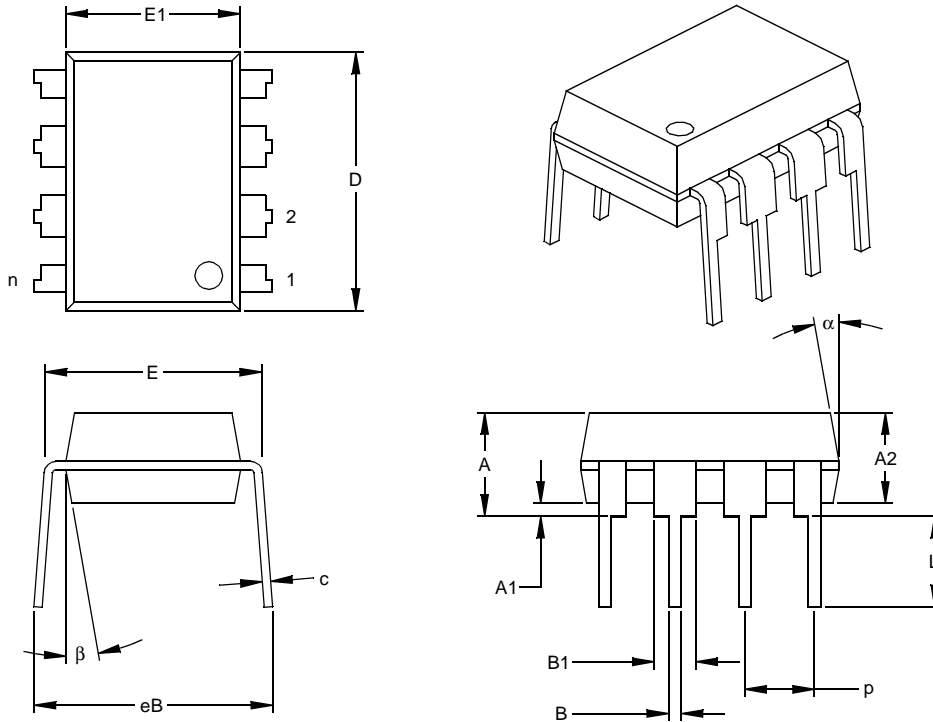
\* Standard PICmicro device marking consists of Microchip part number, year code, week code, and traceability code. For PICmicro device marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

# PIC12F629/675

## 13.2 Package Details

The following sections give the technical details of the packages.

### 8-Lead Plastic Dual In-line (P) – 300 mil (PDIP)



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		8			8	
Pitch	p		.100			2.54	
Top to Seating Plane	A	.140	.155	.170	3.56	3.94	4.32
Molded Package Thickness	A2	.115	.130	.145	2.92	3.30	3.68
Base to Seating Plane	A1	.015			0.38		
Shoulder to Shoulder Width	E	.300	.313	.325	7.62	7.94	8.26
Molded Package Width	E1	.240	.250	.260	6.10	6.35	6.60
Overall Length	D	.360	.373	.385	9.14	9.46	9.78
Tip to Seating Plane	L	.125	.130	.135	3.18	3.30	3.43
Lead Thickness	c	.008	.012	.015	0.20	0.29	0.38
Upper Lead Width	B1	.045	.058	.070	1.14	1.46	1.78
Lower Lead Width	B	.014	.018	.022	0.36	0.46	0.56
Overall Row Spacing	§ eB	.310	.370	.430	7.87	9.40	10.92
Mold Draft Angle Top	α	5	10	15	5	10	15
Mold Draft Angle Bottom	β	5	10	15	5	10	15

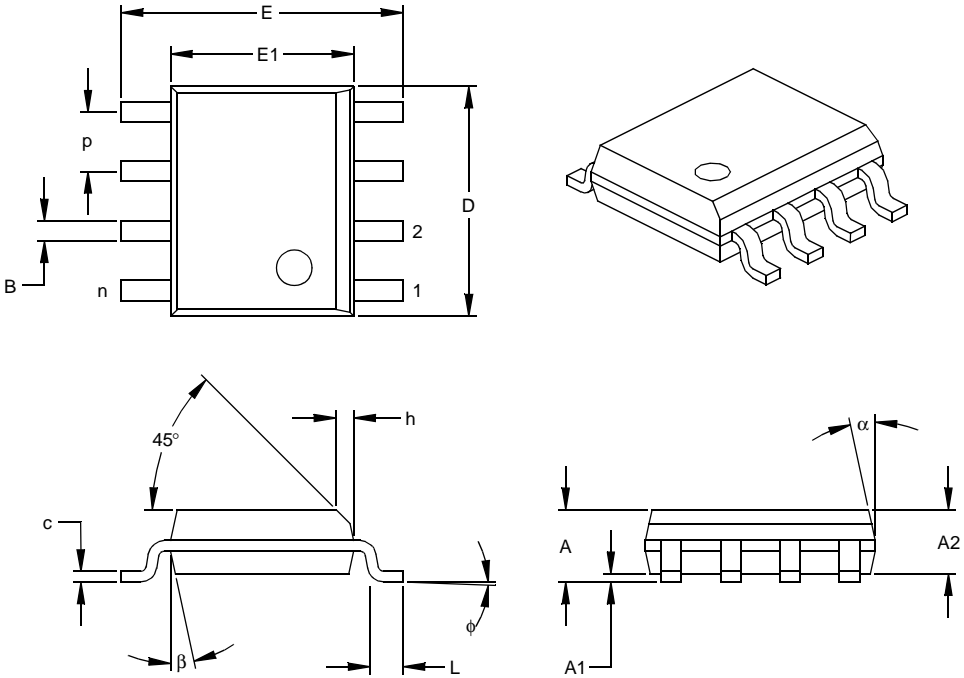
\* Controlling Parameter  
 § Significant Characteristic

Notes:  
 Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.  
 JEDEC Equivalent: MS-001  
 Drawing No. C04-018



# PIC12F629/675

## 8-Lead Plastic Small Outline (SN) – Narrow, 150 mil (SOIC)



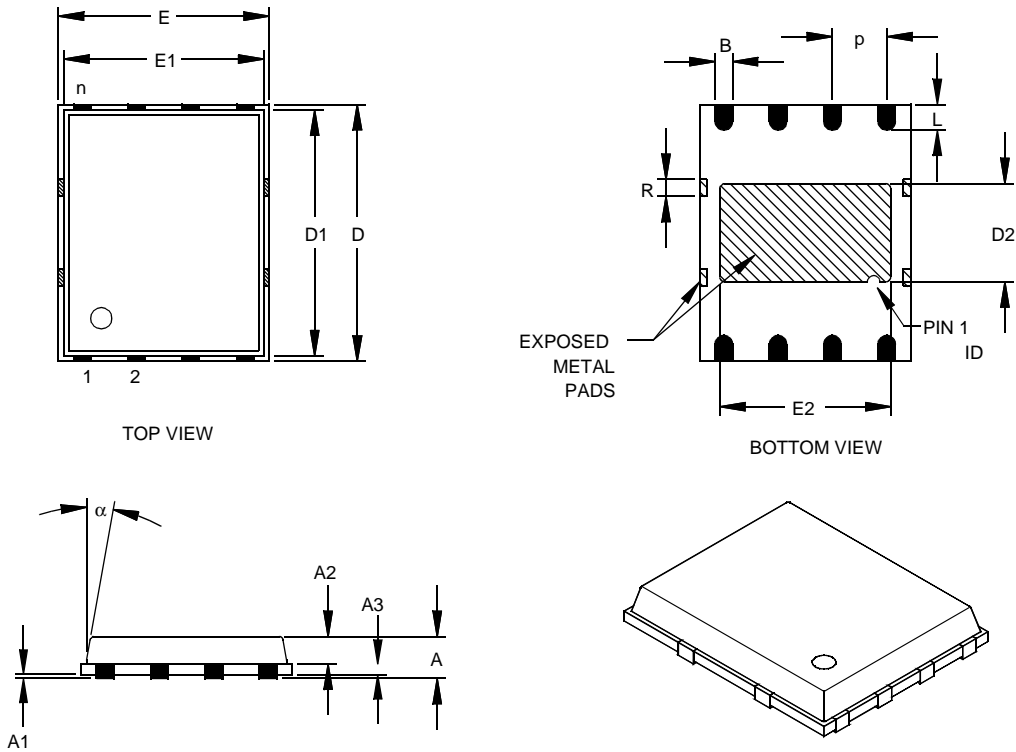
Units		INCHES*			MILLIMETERS		
Dimension	Limits	MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		8			8	
Pitch	p		.050			1.27	
Overall Height	A	.053	.061	.069	1.35	1.55	1.75
Molded Package Thickness	A2	.052	.056	.061	1.32	1.42	1.55
Standoff §	A1	.004	.007	.010	0.10	0.18	0.25
Overall Width	E	.228	.237	.244	5.79	6.02	6.20
Molded Package Width	E1	.146	.154	.157	3.71	3.91	3.99
Overall Length	D	.189	.193	.197	4.80	4.90	5.00
Chamfer Distance	h	.010	.015	.020	0.25	0.38	0.51
Foot Length	L	.019	.025	.030	0.48	0.62	0.76
Foot Angle	φ	0	4	8	0	4	8
Lead Thickness	c	.008	.009	.010	0.20	0.23	0.25
Lead Width	B	.013	.017	.020	0.33	0.42	0.51
Mold Draft Angle Top	α	0	12	15	0	12	15
Mold Draft Angle Bottom	β	0	12	15	0	12	15

\* Controlling Parameter  
 § Significant Characteristic

Notes:  
 Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.  
 JEDEC Equivalent: MS-012  
 Drawing No. C04-057

# PIC12F629/675

## 8-Lead Plastic Micro Leadframe Package (MF) 6x5 mm Body (MLF-S)



Dimension Limits	Units	INCHES			MILLIMETERS*		
		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		8			8	
Pitch	p	.050 BSC			1.27 BSC		
Overall Height	A		.033	.039		0.85	1.00
Molded Package Thickness	A2		.026	.031		0.65	0.80
Standoff	A1	.000	.0004	.002	0.00	0.01	0.05
Base Thickness	A3	.008 REF.			0.20 REF.		
Overall Length	E	.194 BSC			4.92 BSC		
Molded Package Length	E1	.184 BSC			4.67 BSC		
Exposed Pad Length	E2	.152	.158	.163	3.85	4.00	4.15
Overall Width	D	.236 BSC			5.99 BSC		
Molded Package Width	D1	.226 BSC			5.74 BSC		
Exposed Pad Width	D2	.085	.091	.097	2.16	2.31	2.46
Lead Width	B	.014	.016	.019	0.35	0.40	0.47
Lead Length	L	.020	.024	.030	0.50	0.60	0.75
Tie Bar Width	R		.014			.356	
Mold Draft Angle Top	$\alpha$			12°			12°

\*Controlling Parameter

Notes:

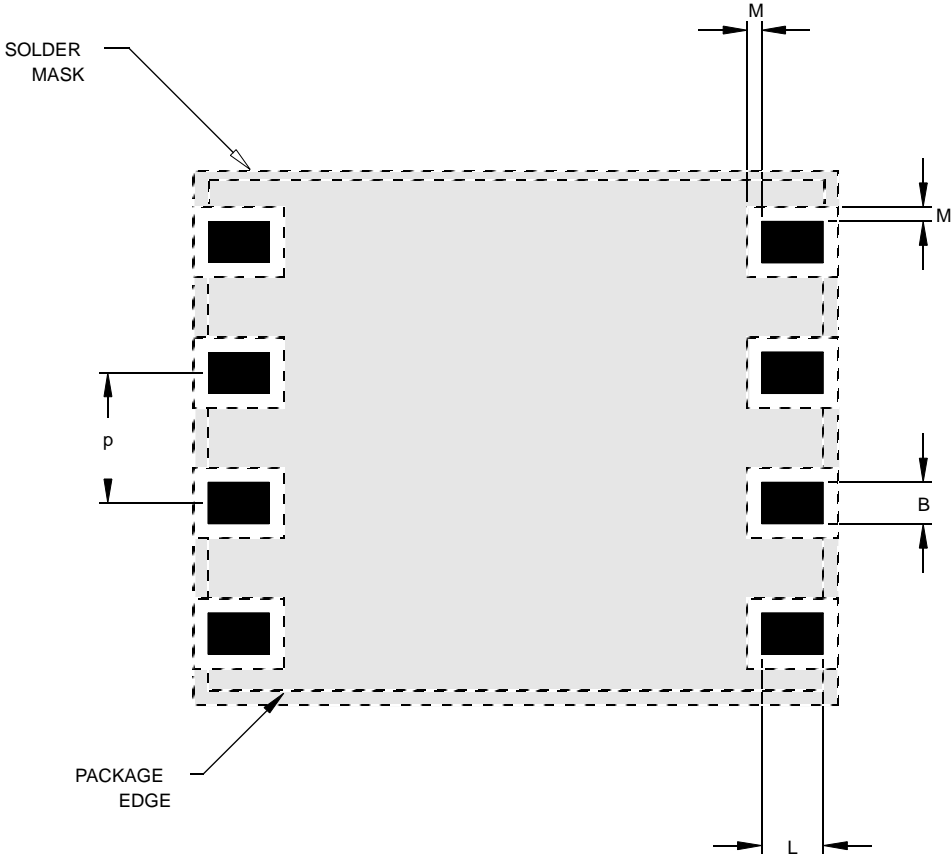
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.

JEDEC equivalent: pending

Drawing No. C04-113

# PIC12F629/675

## 8-Lead Plastic Micro Leadframe Package (MF) 6x5 mm Body (MLF-S)



Dimension Limits	Units	INCHES			MILLIMETERS*		
		MIN	NOM	MAX	MIN	NOM	MAX
Pitch	p	.050 BSC			1.27 BSC		
Pad Width	B	.014	.016	.019	0.35	0.40	0.47
Pad Length	L	.020	.024	.030	0.50	0.60	0.75
Pad to Solder Mask	M	.005		.006	0.13		0.15

\*Controlling Parameter

Drawing No. C04-2113

# PIC12F629/675

---

NOTES:

## APPENDIX A: DATA SHEET REVISION HISTORY

### Revision A

This is a new data sheet.

## APPENDIX B: DEVICE DIFFERENCES

The differences between the PIC12F629/675 devices listed in this data sheet are shown in Table B-1.

**TABLE B-1: DEVICE DIFFERENCES**

Feature	PIC12F629	PIC12F675
A/D	No	Yes

# PIC12F629/675

---

## APPENDIX C: DEVICE MIGRATIONS

This section is intended to describe the functional and electrical specification differences when migrating between functionally similar devices (such as from a PIC16C74A to a PIC16C74B).

**Not Applicable**

## APPENDIX D: MIGRATING FROM OTHER PICmicro® DEVICES

This discusses some of the issues in migrating from other PICmicro devices to the PIC12F6XX family of devices.

### D.1 PIC12C67X to PIC12F6XX

See Microchip website for availability ([www.microchip.com](http://www.microchip.com)).

**Note:** 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.

## APPENDIX E: DEVELOPMENT TOOL VERSION REQUIREMENTS

This lists the minimum requirements (software/firmware) of the specified development tool to support the devices listed in this data sheet.

**MPLAB® IDE:** TBD

**MPLAB® SIMULATOR:** TBD

**MPLAB® ICE 3000:**

PIC12F629/675 Processor Module:

Part Number - TBD

PIC12F629/675 Device Adapter:

Socket Part Number

8-pin SOIC TBD

8-pin PDIP TBD

8-pin MLF-S TBD

**MPLAB® ICD:** TBD

**PRO MATE® II:** TBD

**PICSTART® Plus:** TBD

**MPASM™ Assembler:** TBD

**MPLAB® C18 C Compiler:** TBD

**Note:** Please read all associated README.TXT files that are supplied with the development tools. These "read me" files will discuss product support and any known limitations.

# PIC12F629/675

---

NOTES:



## INDEX

### A

A/D .....	39
Acquisition Requirements .....	43
Block Diagram .....	39
Configuration and Operation .....	39
Effects of a RESET .....	45
Internal Sampling Switch (Rss) Impedence .....	43
Operation During SLEEP .....	45
PIC12F675 Converter Characteristics .....	98
Sampling Time .....	44
Source Impedance .....	43
Summary of Registers .....	45
Absolute Maximum Ratings .....	83
AC Characteristics	
Industrial and Extended .....	91
Additional Pin Functions .....	19
Interrupt-on-Change .....	20
Weak Pull-up .....	19
Analog Input Connection Considerations .....	36
Analog-to-Digital Converter. <i>See</i> A/D	
Assembler	
MPASM Assembler .....	77

### B

Block Diagram	
TMR0/WDT Prescaler .....	25
Block Diagrams	
Analog Input Mode .....	36
Comparator Output .....	36
Comparator Voltage Reference .....	37
GP0 and GP1 Pins .....	21
GP2 .....	22
GP3 .....	22
GP4 .....	23
GP5 .....	23
On-Chip Reset Circuit .....	55
RC Oscillator Mode .....	54
Timer1 .....	28
Watchdog Timer .....	66
Brown-out	
Associated Registers .....	58
Brown-out Detect (BOD) .....	56
Brown-out Reset Timing and Characteristics .....	94

### C

Calibrated Internal RC Frequencies .....	92
CLKOUT .....	54
Code Examples	
Changing Prescaler .....	27
Data EEPROM Read .....	49
Data EEPROM Write .....	49
Initializing GPIO .....	19
Saving STATUS and W Registers in RAM .....	65
Write Verify .....	49
Code Protection .....	68

Comparator .....	33
Associated Registers .....	38
Configuration .....	35
Effects of a RESET .....	37
I/O Operating Modes .....	35
Interrupts .....	38
Operation .....	34
Operation During SLEEP .....	37
Output .....	36
Reference .....	37
Response Time .....	37
Comparator Specifications .....	97
Comparator Voltage Reference Specifications .....	97
Configuration Bits .....	52
Configuring the Voltage Reference .....	37
Crystal Operation .....	53

### D

Data EEPROM Memory	
Associated Registers/Bits .....	50
Code Protection .....	50
EEADR Register .....	47
EECON1 Register .....	47
EECON2 Register .....	47
EEDATA Register .....	47
Data Memory Organization .....	7
DC Characteristics	
Extended .....	87
Extended and Industrial .....	88
Industrial .....	86
Development Support .....	77
Development Tool Version Requirements .....	109
Device Differences .....	107
Device Migrations .....	108
Device Overview .....	5

### E

EEPROM Data Memory	
Reading .....	49
Spurious Write .....	49
Write Verify .....	49
Writing .....	49
Electrical Specifications .....	83
Errata .....	3

### F

Firmware Instructions .....	69
-----------------------------	----

### G

General Purpose Register File .....	7
GPIO	
Associated Registers .....	24
GPIO Port .....	19
GPIO, TRISIO Registers .....	19

# PIC12F629/675

## I

ICEPIC In-Circuit Emulator .....	78
ID Locations .....	68
In-Circuit Serial Programming .....	68
Indirect Addressing, INDF and FSR Registers.....	18
Instruction Format .....	69
Instruction Set .....	69
ADDLW .....	71
ADDWF .....	71
ANDLW .....	71
ANDWF .....	71
BCF .....	71
BSF .....	71
BTFSC .....	71
BTFSS .....	71
CALL .....	72
CLRF .....	72
CLRW .....	72
CLRWDT .....	72
COMF .....	72
DECF .....	72
DECFSZ .....	73
GOTO .....	73
INCF .....	73
INCFSZ .....	73
IORLW .....	73
IORWF .....	73
MOVF .....	74
MOVLW .....	74
MOVWF .....	74
NOP .....	74
RETFIE .....	74
RETLW .....	74
RETURN .....	75
RLF .....	75
RRF .....	75
SLEEP .....	75
SUBLW .....	75
SUBWF .....	75
SWAPF .....	76
XORLW .....	76
XORWF .....	76
Summary Table .....	70
Internal 4 MHz Oscillator .....	54
Internal Sampling Switch (Rss) Impedance .....	43
Interrupts .....	62
A/D Converter .....	64
Comparator .....	64
Context Saving.....	65
GP2/INT .....	64
GPIO .....	64
Summary of Registers .....	65
TMR0 .....	64

## K

KEELOQ Evaluation and Programming Tools .....	80
---	----

## M

MCLR.....	56
Memory Organization	
Data EEPROM Memory.....	47
Migrating from other PICmicro Devices.....	108
MPLAB C17 and MPLAB C18 C Compilers .....	77
MPLAB ICD In-Circuit Debugger .....	79
MPLAB ICE High Performance Universal In-Circuit Emulator with MPLAB IDE .....	78
MPLAB Integrated Development Environment Software .....	77
MPLINK Object Linker/MPLIB Object Librarian .....	78

## O

OPCODE Field Descriptions.....	69
Oscillator Configurations.....	53
Oscillator Start-up Timer (OST) .....	56

## P

Packaging .....	101
Details.....	102
Marking.....	101
PCL and PCLATH.....	17
Computed GOTO.....	17
Stack.....	17
PICDEM 1 Low Cost PICmicro Demonstration Board .....	79
PICDEM 17 Demonstration Board.....	80
PICDEM 2 Low Cost PIC16CXX Demonstration Board .....	79
PICDEM 3 Low Cost PIC16CXXX Demonstration Board .....	80
PICSTART Plus Entry Level Development Programmer.....	79
Pin Descriptions and Diagrams .....	21
Pinout Descriptions	
PIC12F629 .....	6
PIC12F675 .....	6
Power Control/Status Register (PCON).....	57
Power-Down Mode (SLEEP) .....	67
Power-on Reset (POR).....	56
Power-up Timer (PWRT) .....	56
Prescaler.....	27
Switching Prescaler Assignment .....	27
PRO MATE II Universal Device Programmer .....	79
Program Memory Organization.....	7
Programming, Device Instructions.....	69

## R

RC Oscillator .....	54
Read-Modify-Write Operations .....	69
Registers	
ADCON0 (A/D Control) .....	41
ANSEL (Analog Select) .....	42
CMCON (Comparator Control) .....	33
CONFIG (Configuration Word) .....	52
EEADR (EEPROM Address) .....	47
ECON1 (EEPROM Control) .....	48
EEDAT (EEPROM Data) .....	47
INTCON (Interrupt Control) .....	13
IOCB (Interrupt-on-Change GPIO) .....	20
Maps	
PIC12F629 .....	8
PIC12F675 .....	8
OPTION_REG (Option) .....	12, 26
OSCCAL (Oscillator Calibration) .....	16
PCON (Power Control) .....	16
PIE1 (Peripheral Interrupt Enable 1) .....	14
PIR1 (Peripheral Interrupt 1) .....	15
STATUS .....	11
T1CON (Timer1 Control) .....	30
VRCON (Voltage Reference Control) .....	38
WPU (Weak Pull-up) .....	19
RESET .....	55
Revision History .....	107

## S

Software Simulator (MPLAB SIM) .....	78
Special Features of the CPU .....	51
Special Function Registers .....	8
Special Functions Registers Summary .....	9

## T

Time-out Sequence .....	57
Timer0 .....	25
Associated Registers .....	27
External Clock .....	26
Interrupt .....	25
Operation .....	25
T0CKI .....	26

## Timer1

Associated Registers .....	31
Asynchronous Counter Mode .....	31
Reading and Writing .....	31
Capacitor Selection .....	31
Interrupt .....	29
Modes of Operations .....	29
Operation During SLEEP .....	31
Oscillator .....	31
Prescaler .....	29
Timer1 Module with Gate Control .....	28
Timing Diagrams	
CLKOUT and I/O .....	93
External Clock .....	91
INT Pin Interrupt .....	64
PIC12F675 A/D Conversion (Normal Mode) .....	99
PIC12F675 A/D Conversion Timing (SLEEP Mode) .....	100
RESET, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer .....	94
Time-out Sequence on Power-up (MCLR not Tied to VDD)/	
Case 1 .....	60
Case 2 .....	60
Time-out Sequence on Power-up (MCLR Tied to VDD) .....	60
Timer0 and Timer1 External Clock .....	96
Timer1 Incrementing Edge .....	29
Timing Parameter Symbology .....	90

## V

Voltage Reference Accuracy/Error .....	37
--	----

## W

### Watchdog Timer

Summary of Registers .....	66
Watchdog Timer (WDT) .....	65
WWW, On-Line Support .....	3

# PIC12F629/675

---

NOTES:

## ON-LINE SUPPORT

Microchip provides on-line support on the Microchip World Wide Web (WWW) site.

The web site is used by Microchip as a means to make files and information easily available to customers. To view the site, the user must have access to the Internet and a web browser, such as Netscape or Microsoft Explorer. Files are also available for FTP download from our FTP site.

### Connecting to the Microchip Internet Web Site

The Microchip web site is available by using your favorite Internet browser to attach to:

**[www.microchip.com](http://www.microchip.com)**

The file transfer site is available by using an FTP service to connect to:

**<ftp://ftp.microchip.com>**

The web site and file transfer site provide a variety of services. Users may download files for the latest Development Tools, Data Sheets, Application Notes, User's Guides, Articles and Sample Programs. A variety of Microchip specific business information is also available, including listings of Microchip sales offices, distributors and factory representatives. Other data available for consideration is:

- Latest Microchip Press Releases
- Technical Support Section with Frequently Asked Questions
- Design Tips
- Device Errata
- Job Postings
- Microchip Consultant Program Member Listing
- Links to other useful web sites related to Microchip Products
- Conferences for products, Development Systems, technical information and more
- Listing of seminars and events

## Systems Information and Upgrade Hot Line

The Systems Information and Upgrade Line provides system users a listing of the latest versions of all of Microchip's development systems software products. Plus, this line provides information on how customers can receive any currently available upgrade kits. The Hot Line Numbers are:

1-800-755-2345 for U.S. and most of Canada, and

1-480-792-7302 for the rest of the world.

013001

# PIC12F629/675

---

---

## 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 Data Sheet.

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: **PIC12F629/675** Literature Number: **DS41190A**

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 data sheet easy to follow? If not, why?

---

---

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

---

---

5. What deletions from the data sheet 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?

---

---

8. How would you improve our software, systems, and silicon products?

---

---

## PRODUCT IDENTIFICATION SYSTEM

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

<u>PART NO.</u>	-	<u>X</u>	<u>/XX</u>	<u>XXX</u>
Device		Temperature Range	Package	Pattern
Device		PIC12F6XX: Standard VDD range 2.0V to 5.5V PIC12F6XXT VDD range 2.0V to 5.5V (Tape and Reel)		
Temperature Range	I	= -40°C to +85°C		
	E	= -40°C to +125°C		
Package	P	= PDIP		
	SN	= SOIC (Gull Wing, 150 mil body)		
	MF	= MLF-S		
Pattern				3-Digit Pattern Code for QTP (blank otherwise).

**Examples:**

- a) PIC12F629 - E/P 301 = Extended Temp., PDIP package, 20 MHz, QTP pattern #301.
- b) PIC12F675 - I/SO = Industrial Temp., SOIC package, 20 MHz.

## Sales and Support

### Data Sheets

Products supported by a preliminary Data Sheet may have an errata sheet describing minor operational differences and recommended workarounds. To determine if an errata sheet exists for a particular device, please contact one of the following:

1. Your local Microchip sales office
2. The Microchip Corporate Literature Center U.S. FAX: (480) 792-7277
3. The Microchip Worldwide Site ([www.microchip.com](http://www.microchip.com))

Please specify which device, revision of silicon and Data Sheet (include Literature #) you are using.

### New Customer Notification System

Register on our web site ([www.microchip.com/cn](http://www.microchip.com/cn)) to receive the most current information on our products.



## 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: 480-792-7627  
Web Address: <http://www.microchip.com>

#### Rocky Mountain

2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 480-792-7966 Fax: 480-792-7456

#### Atlanta

500 Sugar Mill Road, Suite 200B  
Atlanta, GA 30350  
Tel: 770-640-0034 Fax: 770-640-0307

#### Boston

2 Lan Drive, Suite 120  
Westford, MA 01886  
Tel: 978-692-3848 Fax: 978-692-3821

#### Chicago

333 Pierce Road, Suite 180  
Itasca, IL 60143  
Tel: 630-285-0071 Fax: 630-285-0075

#### Dallas

4570 Westgrove Drive, Suite 160  
Addison, TX 75001  
Tel: 972-818-7423 Fax: 972-818-2924

#### Detroit

Tri-Atria Office Building  
32255 Northwestern Highway, Suite 190  
Farmington Hills, MI 48334  
Tel: 248-538-2250 Fax: 248-538-2260

#### Kokomo

2767 S. Albright Road  
Kokomo, Indiana 46902  
Tel: 765-864-8360 Fax: 765-864-8387

#### Los Angeles

18201 Von Karman, Suite 1090  
Irvine, CA 92612  
Tel: 949-263-1888 Fax: 949-263-1338

#### New York

150 Motor Parkway, Suite 202  
Hauppauge, NY 11788  
Tel: 631-273-5305 Fax: 631-273-5335

#### San Jose

Microchip Technology Inc.  
2107 North First Street, Suite 590  
San Jose, CA 95131  
Tel: 408-436-7950 Fax: 408-436-7955

#### Toronto

6285 Northam Drive, Suite 108  
Mississauga, Ontario L4V 1X5, Canada  
Tel: 905-673-0699 Fax: 905-673-6509

### ASIA/PACIFIC

#### Australia

Microchip Technology Australia Pty Ltd  
Suite 22, 41 Rawson Street  
Epping 2121, NSW  
Australia  
Tel: 61-2-9868-6733 Fax: 61-2-9868-6755

#### China - Beijing

Microchip Technology Consulting (Shanghai)  
Co., Ltd., Beijing Liaison Office  
Unit 915  
Bei Hai Wan Tai Bldg.  
No. 6 Chaoyangmen Beidajie  
Beijing, 100027, No. China  
Tel: 86-10-85282100 Fax: 86-10-85282104

#### China - Chengdu

Microchip Technology Consulting (Shanghai)  
Co., Ltd., Chengdu Liaison Office  
Rm. 2401, 24th Floor,  
Ming Xing Financial Tower  
No. 88 TIDU Street  
Chengdu 610016, China  
Tel: 86-28-6766200 Fax: 86-28-6766599

#### China - Fuzhou

Microchip Technology Consulting (Shanghai)  
Co., Ltd., Fuzhou Liaison Office  
Unit 28F, World Trade Plaza  
No. 71 Wusi Road  
Fuzhou 350001, China  
Tel: 86-591-7503506 Fax: 86-591-7503521

#### China - Shanghai

Microchip Technology Consulting (Shanghai)  
Co., Ltd.  
Room 701, Bldg. B  
Far East International Plaza  
No. 317 Xian Xia Road  
Shanghai, 200051  
Tel: 86-21-6275-5700 Fax: 86-21-6275-5060

#### China - Shenzhen

Microchip Technology Consulting (Shanghai)  
Co., Ltd., Shenzhen Liaison Office  
Rm. 1315, 13/F, Shenzhen Kerry Centre,  
Renminnan Lu  
Shenzhen 518001, China  
Tel: 86-755-2350361 Fax: 86-755-2366086

#### Hong Kong

Microchip Technology Hongkong Ltd.  
Unit 901-6, Tower 2, Metroplaza  
223 Hing Fong Road  
Kwai Fong, N.T., Hong Kong  
Tel: 852-2401-1200 Fax: 852-2401-3431

#### India

Microchip Technology Inc.  
India Liaison Office  
Divyasree Chambers  
1 Floor, Wing A (A3/A4)  
No. 11, O'Shaugnessey Road  
Bangalore, 560 025, India  
Tel: 91-80-2290061 Fax: 91-80-2290062

### Japan

Microchip Technology Japan K.K.  
Benex S-1 6F  
3-18-20, Shinyokohama  
Kohoku-Ku, Yokohama-shi  
Kanagawa, 222-0033, Japan  
Tel: 81-45-471- 6166 Fax: 81-45-471-6122

### Korea

Microchip Technology Korea  
168-1, Youngbo Bldg. 3 Floor  
Samsung-Dong, Kangnam-Ku  
Seoul, Korea 135-882  
Tel: 82-2-554-7200 Fax: 82-2-558-5934

### Singapore

Microchip Technology Singapore Pte Ltd.  
200 Middle Road  
#07-02 Prime Centre  
Singapore, 188980  
Tel: 65-6334-8870 Fax: 65-6334-8850

### Taiwan

Microchip Technology Taiwan  
11F-3, No. 207  
Tung Hua North Road  
Taipei, 105, Taiwan  
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

### EUROPE

#### Denmark

Microchip Technology Nordic ApS  
Regus Business Centre  
Lautrup høj 1-3  
Ballerup DK-2750 Denmark  
Tel: 45 4420 9895 Fax: 45 4420 9910

#### France

Microchip Technology SARL  
Parc d'Activite du Moulin de Massy  
43 Rue du Saule Trapu  
Batiment A - 1er Etage  
91300 Massy, France  
Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

#### Germany

Microchip Technology GmbH  
Gustav-Heinemann Ring 125  
D-81739 Munich, Germany  
Tel: 49-89-627-144 0 Fax: 49-89-627-144-44

#### Italy

Microchip Technology SRL  
Centro Direzionale Colleoni  
Palazzo Taurus 1 V. Le Colleoni 1  
20041 Agrate Brianza  
Milan, Italy  
Tel: 39-039-65791-1 Fax: 39-039-6899883

#### United Kingdom

Arizona Microchip Technology Ltd.  
505 Eskdale Road  
Winnersh Triangle  
Wokingham  
Berkshire, England RG41 5TU  
Tel: 44 118 921 5869 Fax: 44-118 921-5820

03/01/02