UNIVERSIDAD DE COSTA RICA
Escuela de Ingeniería Eléctrica
IE0624 - Laboratorio de Microcontroladores

Laboratorio 1

Sebastián Vargas Quesada, C18295

28 de agosto del 2024, Ciclo II

1. Resumen

En el presente laboratorio se desarrolló un simulador de tómbola utilizando el microcontrolador PIC12F675. Para su implementación, se emplearon componentes electrónicos adicionales como el 74HC595 y displays de 7 segmentos. La lógica del programa fue codificada en lenguaje C y los resultados se simularon en Simulide. Durante el proyecto, la memoria limitada del microcontrolador presentó un desafío, lo que dificultó la implementación de la función para verificar números duplicados. No obstante, se logró generar números aleatorios de manera satisfactoria. El repositorio del laboratorio se puede consultar en el siguiente enlace: https://github.com/sebasvq106/microcontroladores

2. Nota Teorica

2.1. Información General del Microcontrolador

Para la realización de este laboratorio se utilizó el microcontrolador PIC12F675 el cual es un dispositivo de 8 bits de la familia PIC, este es fabricado por Microchip Technology y posee una arquitectura RISC. Este microcontrolador es uno de los más utilizados para proyectos de electrónica gracias a su bajo costo y versatilidad, sin embargo, no se pueden relizar dispositivos muy complejos por la limitación de su memoria.

Para este laboratorio lo que utilizaremos serán los pines de I/O que posee el microcontrolador. Este cuanta con 6 pines de entrada/salida digital (GPIO), los cuales pueden configurarse dependiendo de la aplicación, por otra parte estos pines pueden funcionar como entradas analógicas, comparadores o salidas PWM. Otra funcionalidad que presenta el PIC12F675 es que incluye un convertidor ADC de 10 bits con 4 canales.

2.2. Diagrama de Bloques

A continuación se muestra el diagrama de bloques del microcontrolador:

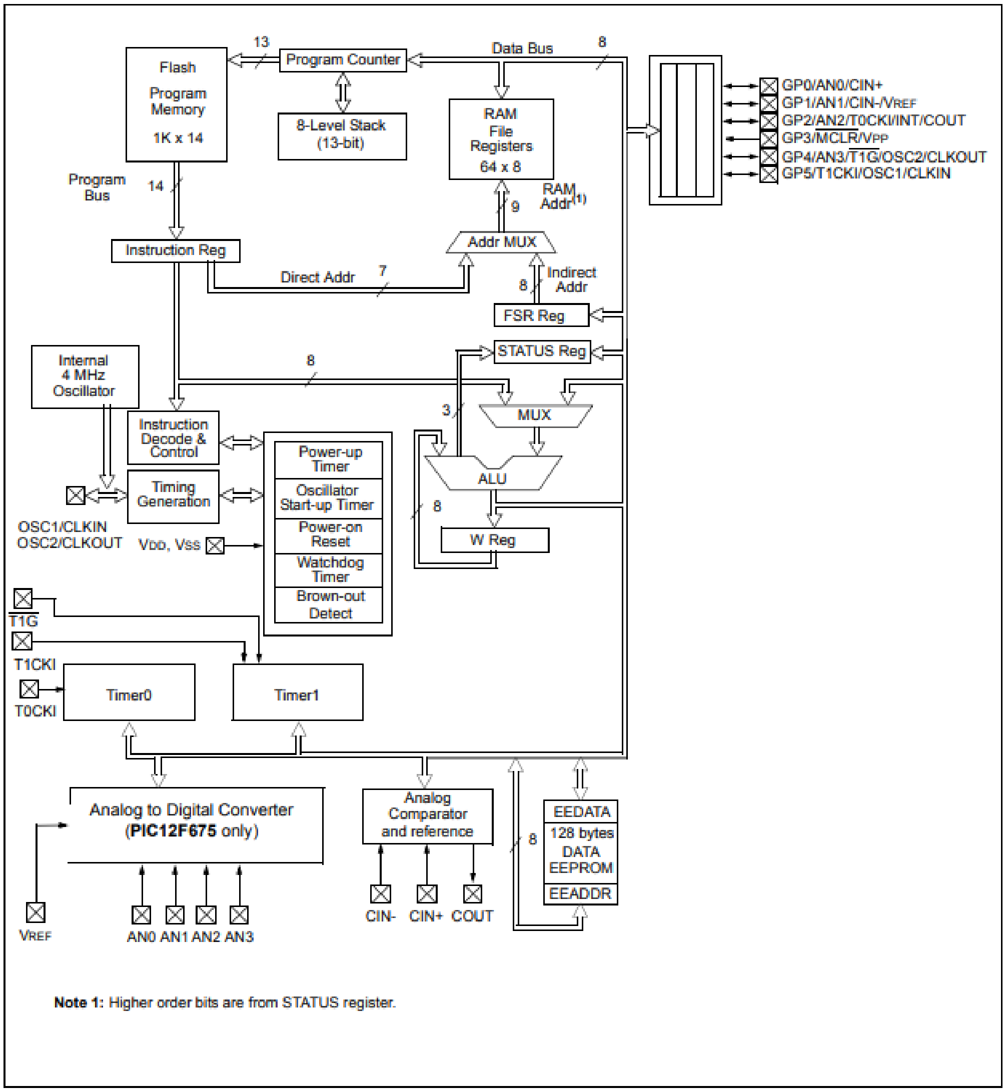


Figura 1: Diagrama de Bloques del PIC12F675 [1]

2.3. Diagrama de Pines

A continuación se muestra el diagrama de pines del microcontrolador:

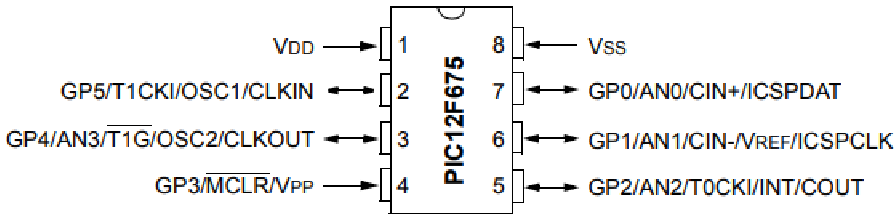


Figura 2: Diagrama de Pines del PIC12F675 [1]

2.4. Características Eléctricas

Ahora se mostrará las características eléctricas encontradas en el Datasheet del microcontrolador:

Absolute Maximum Ratings†

Ambient temperature under bias .... -40 to +125°C
Storage temperature .... -65°C to +150°C
Voltage on V_{DD} with respect to V_{SS} .... -0.3 to +6.5V
Voltage on $\overline{MCLR}$ with respect to Vss .... -0.3 to +13.5V
Voltage on all other pins with respect to V_{SS} .... -0.3V to (V_{DD} + 0.3V)
Total power dissipation(1) .... 800 mW
Maximum current out of V_{SS} pin .... 300 mA
Maximum current into V_{DD} pin .... 250 mA
Input clamp current, I_{IK} ($V_I < 0$ or $V_I > V_{DD}$) .... ± 20 mA
Output clamp current, I_{OK} ($V_O < 0$ or $V_O > V_{DD}$) .... ± 20 mA
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: $P_{DIS} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD}-V_{OH}) \times I_{OH}\} + \sum(V_{OL} \times I_{OL})$.

Figura 3: Características Eléctricas del PIC12F675 [1]

2.5. Periféricos utilizados

Como se mencionó anteriormente para este laboratorio se utilizarán los diferentes pines de entrada y salida que posee el microcontrolador, sin embargo estos se tienen que configurar de una manera especifica para que el circuito funcione, para esto se utilizó el registro TRISIO indicando que pines serán entradas y cuales serán salidas. A continuación se muestra el comando escrito en el archivo .c que produce esto:

```
TRISIO = 0b00100000; // Todos los pines como salida excepto GP5 que es
    entrada
```

En el pin GP5 se conectará el botón, es por esto que se configura como una entrada. Por otra también se inicializa todos los pines de en un valor bajo como se muestra a continuación:

```
GPIO = 0x00; // Inicializar todos los pines en bajo
```

2.6. Componentes Electrónicos Complementarios

Uno de los componentes complementarios de este circuito es el display de 7 segmentos. Este es necesario para mostrar los dígitos luego de presionar el botón, a continuación se muestra el diagrama de bloques:

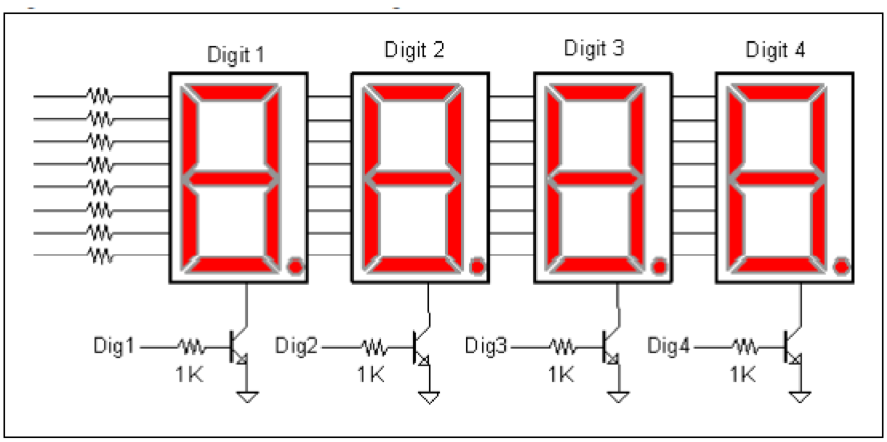


Figura 4: Diagrama de bloques del Display de 7 segmentos [3]

Sin embargo para manejar el display se necesitan 8 salidas, las cuales el microcontrolador no tiene, es por esta situación que se necesita utilizar un componete que permita hacer una conexión entre el microcontraldor y el display. Para este circuito se utilizará el 74HC595 el cual se encarga de recibir los datos por el pin DS y en cada ciclo de reloj este enviará un bit a los diferentes pines de salida. A continuación se muestra el componete:

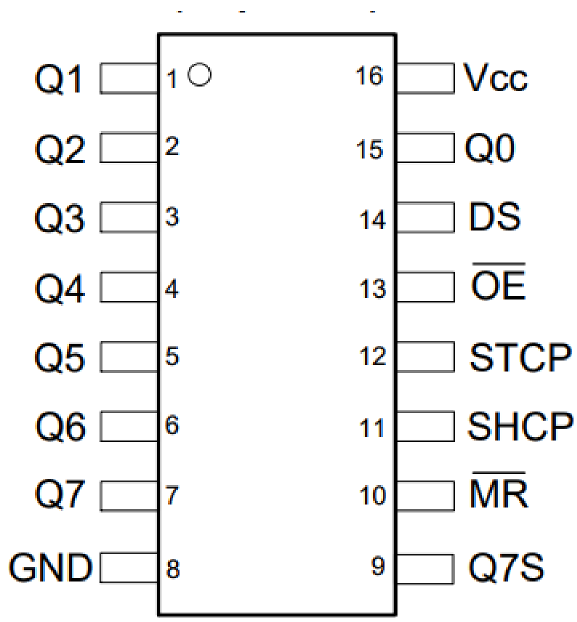


Figura 5: Pines del 74HC595 [2]

Otro apartado importante contar con un presupuesto para realizar el laboratorio, por lo que en la siguiente tabla se las los precios aproximados de los componentes a utilizar:

Componentes	**Precio aproximado**
PIC2F675	2 USD
3 x resistores	1 USD
1 x capacitor	0.5 USD
2 x 74HC595	2 USD
2 x Display 7 segmentos	4 USD

3. Desarrollo

3.1. Implemetación del botón

Lo primero que realizó fue implementar el botón, para esto se utilizó como base el código de holaPIC el cual enciende un LED. Se configura el pin 5 para que funcione como una entrada mientras que los otros se mantienen como salidas. A continuación se muestra una imagen con el circuito implemento y el código que se utilizó:

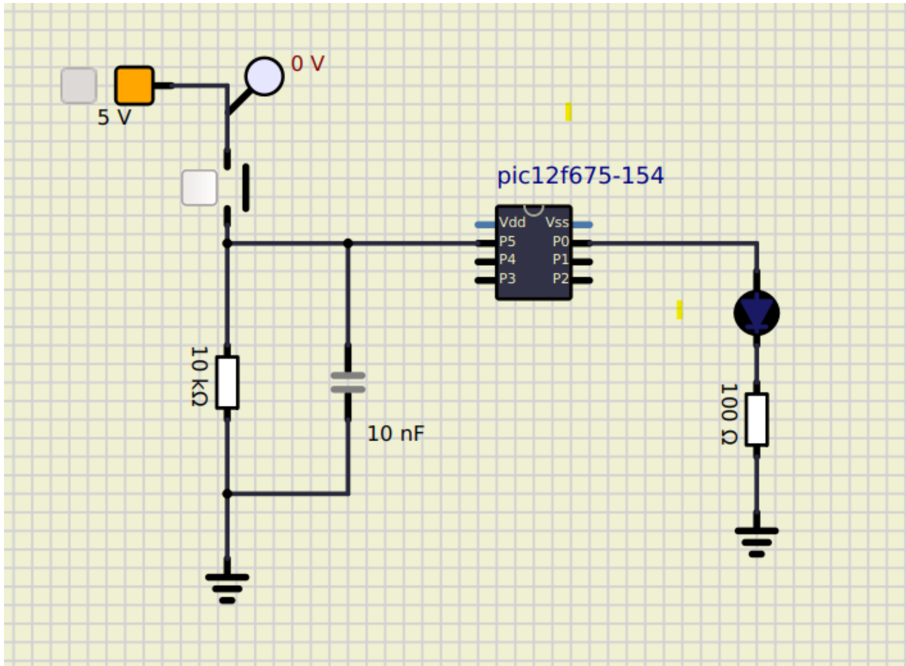


Figura 6: Implementación del botón

```c
#include <pic14/pic12f675.h>
#define push_button GP5 //push button


// Funcion de delay
void delay(unsigned int tiempo)
{
    unsigned int i, j;
    for (i = 0; i < tiempo; i++)
        for (j = 0; j < 1275; j++);
}

void main(void)
{
    TRISIO = 0b00100000; // Todos los pines como salida excepto GP5 que es entrada
    GPIO = 0x00; // Poner todos los pines en bajo inicialmente

    unsigned int time = 200;

    while (1)
    {
        if (push_button) // Si GP5 est  en bajo (bot n presionado)
        {
            GP0 = 1; // Encender el LED conectado a GP0
            delay(time);
        }
```

```
        else // Si GP5 est  en alto (bot n no presionado)
        {
            GP0 = 0; // Asegurar que el LED est  apagado
        }
    }
}
```

3.2. Implementación del Display de 7 segmentos y 74HC595

Como se mencionó en la nota teórica, el display de 7 segmentos no puede conectarse directamente al microcontrolador PIC12F675. Por lo tanto, se decidió utilizar un 74HC595 como interfaz entre el display y el PIC12F675, siguiendo la referencia propuesta por Microcontrollers Lab en su video [4]. Debido a la necesidad de conectar dos displays de 7 segmentos, se implementó una conexión en cadena (Daisy Chain) utilizando dos 74HC595. En este método, ambos 74HC595 comparten sus entradas de STC y SHC, que son los relojes necesarios para la operación, mientras que el pin DS del segundo 74HC595 se conecta al pin Q7' del primero para la transmisión de datos. A continuación se muestra el circuito diseñado:

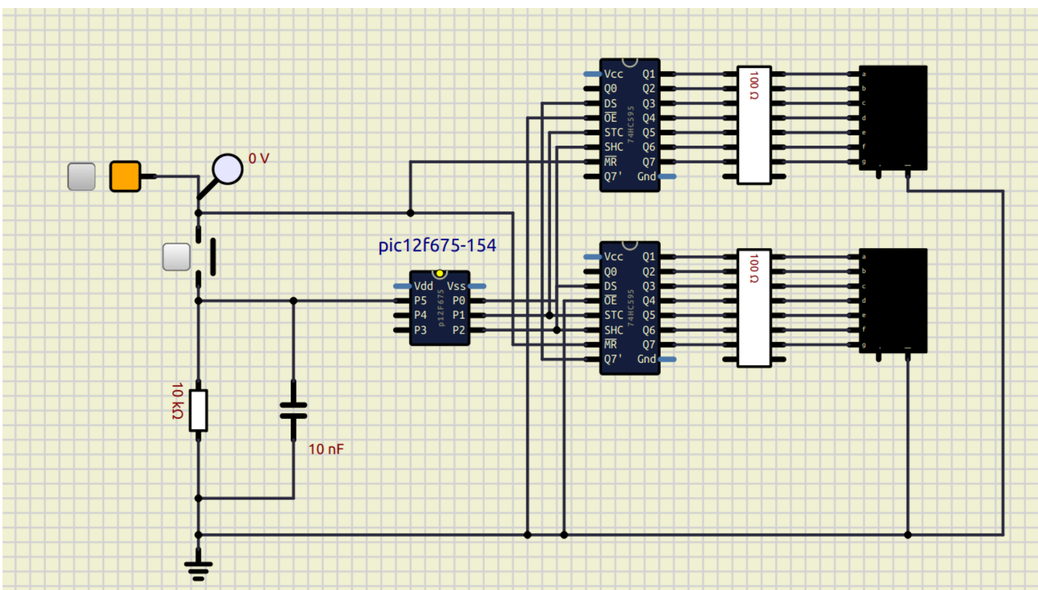


Figura 7: Implementación de los displays de 7 segmentos

3.3. Expliación del Código

```
#define push_button GP5   // Bot n de entrada en GP5
#define DS GP0
#define STC GP1
#define SHC GP2
// Tabla de n meros para el display de 7 segmentos
const unsigned char segment_map[10] = {
    0x7E, // 0
    0x0C, // 1
```

```
    0xB6, // 2
    0x9E, // 3
    0xCC, // 4
    0xDA, // 5
    0xFA, // 6
    0x0E, // 7
    0xFF, // 8
    0xCE  // 9
};
```

Aquí se definen los pines de salida que se conectarán al 74HC595 y se crea un array con los valores que se mostrarán en los displays. Es importante mencionar que estos valores se determinaron a través de pruebas bit a bit para asegurar que cada segmento del display se active correctamente.

```
// Funci n para enviar datos a los dos 74HC595

void shiftOutTwo(unsigned char data1, unsigned char data2) {

    for (unsigned char i = 7; i < 0xFF; i--) { // Usar < 0xFF para evitar problemas de comparaci n
        DS = (data1 >> i) & 0x01; // Enviar cada bit del primer n mero
        SHC = 1;
        delay(5);
        SHC = 0;
        delay(5);
    }

    for (unsigned char i = 7; i < 0xFF; i--) { // Usar < 0xFF para evitar problemas de comparaci n
        DS = (data2 >> i) & 0x01; // Enviar cada bit del segundo n mero
        SHC = 1;
        delay(5);
        SHC = 0;
        delay(5);
    }
    STC = 1;
    delay(5);
    STC = 0;
}

// Funci n para debounce del bot n
unsigned char debounce(void) {
    if (push_button == 1) {
        delay(10);
        if (push_button == 1) {
            return 1;
        }
    }
    return 0;
}
```

```

void main(void) {

    TRISIO = 0b00100000; // Todos los pines como salida excepto GP5 que es entrada
    GPIO = 0x00; // Inicializar todos los pines en bajo
    unsigned char r1 = 0; // Contador para el primer n mero
    unsigned char r2 = 0; // Contador para el segundo n mero
    unsigned char speed1 = 1; // Velocidad de cambio del primer contador
    unsigned char speed2 = 3; // Velocidad de cambio del segundo contador
    unsigned char index = 0; // Contador de pulsaciones del bot n

    while (1) {
        // Incrementar los contadores r pidamente mientras el bot n no est  presionado
        if (push_button == 0) {
            r1 = (r1 + speed1) % 10; // Incremento r pido y ciclado entre 0 y 9
            r2 = (r2 + speed2) % 10;
            delay(10); // Peque o retardo para que los n meros cambien r pidamente
        }
        if (debounce()) { // Verificar el bot n con debounce
            shiftOutTwo(segment_map[r1], segment_map[r2]); // Enviar los valores a los 74HC595
            index++;
            // Esperar hasta que el bot n se libere
            while (push_button == 1);
            // Retardo para evitar m ltiples conteos
            delay(100);
        }
        // Si se ha presionado el bot n 10 veces, mostrar 99 tres veces y reiniciar
        if (index >= 10) {
            for (unsigned char blink = 0; blink < 3; blink++) {
                delay(250); // Pausa
                shiftOutTwo(segment_map[9], segment_map[9]); // Mostrar 99
                delay(250); // Pausa
                shiftOutTwo(0x00, 0x00); // Apagar display
            }

            index = 0; // Reiniciar el contador para los pr ximos n meros
        }
    }
}
```

La función shiftOutTwo es responsable de enviar los datos a los 74HC595. Básicamente, toma cada bit de los datos y los transmite durante cada ciclo del reloj SHC. Una vez que se han enviado todos los datos, se genera un flanco en el reloj STC para transferir los bits a las salidas Q1 a Q7.

La función debounce se encarga de verificar si hubo un rebote en el botón. Por esta razón, si el botón sigue presionado después de un retardo de 10 milisegundos, la función retorna 1, confirmando una pulsación válida.

Por último, en la función main se configuran los pines y se definen las variables a utilizar. Mientras el botón no esté presionado, se utiliza la técnica de contadores para seleccionar dos números aleatorios entre 0 y 9. Una vez que se presiona el botón, los datos se envían a la función shiftOutTwo y el contador aumenta en uno. Si este contador llega a 10, el display muestra un parpadeo del número 99, lo que indica un reinicio del juego.

3.4. Resultados y Análisis

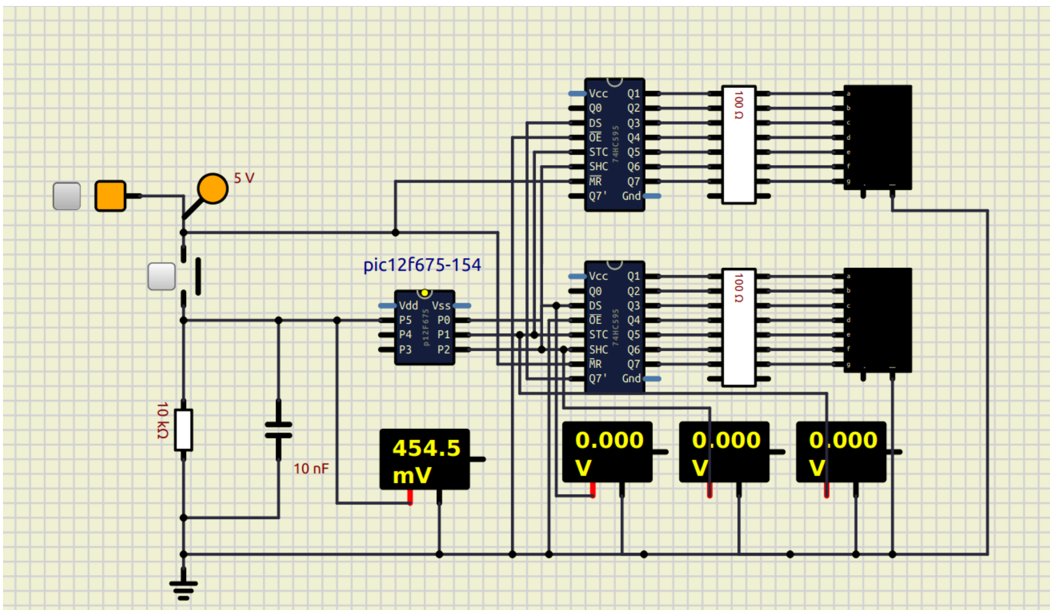


Figura 8: Circuito antes de accionar el botón

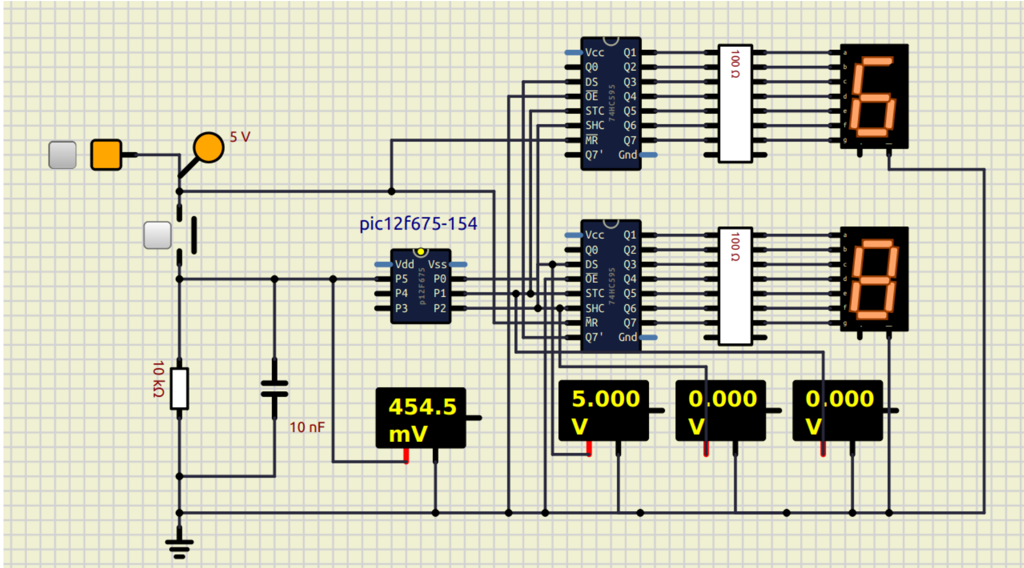


Figura 9: Circuito luego de accionar el botón

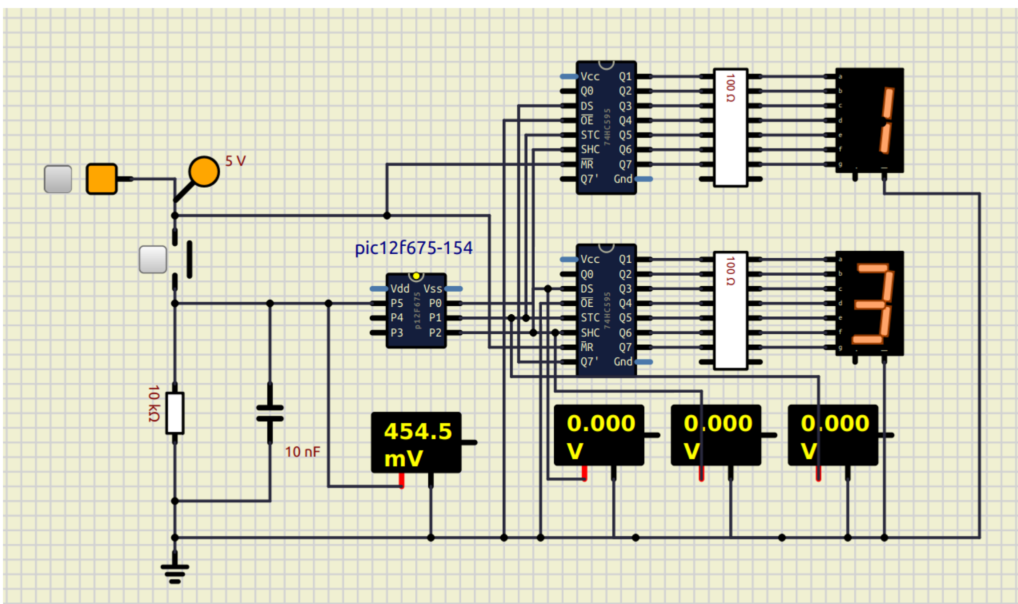


Figura 10: Circuito luego de accionar el botón en varias ocasiones

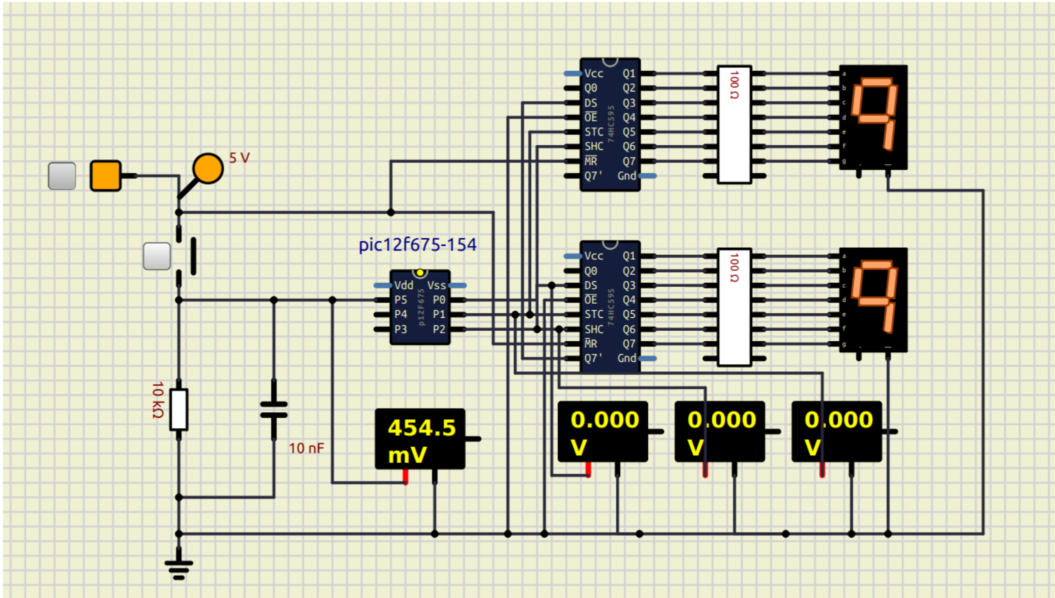


Figura 11: Circuito luego de accionar el botón 10 veces

Luego de analizar los resultados obtenidos, se puede confirmar que el circuito funciona según lo esperado. Cada vez que se acciona el botón, se muestra un número aleatorio entre 0 y 9 en los displays, y tras presionar el botón 10 veces, el número 99 parpadea, indicando el reinicio del juego. Sin embargo, uno de los inconvenientes fue que no se pudo implementar la lógica para evitar números repetidos, debido a las limitaciones de memoria del microcontrolador.

4. Conclusiones y Recomendaciones

- Se logró configurar el PIC12F675 para que un pin funcionara como entrada, permitiendo la conexión de un botón capaz de accionar el sistema.
- Se implementó el display de 7 segmentos en el circuito con la ayuda del 74HC595, que actúa como intermediario entre el display y el microcontrolador.
- Se desarrolló un código que muestra números aleatorios en el display de 7 segmentos cada vez que se presiona el botón.
- No se pudo implementar la función para verificar si la combinación de números se repite, debido a las limitaciones de memoria del microcontrolador.
- Una recomendación es utulizar tipos de variables adecuados para optimizar el programa y minimizar errores relacionados con la memoria.

Referencias

[1] Microchip Technology Inc. (2010). *PIC12F675 Data Sheet*.Recuperado el 27 de agosto del 2024, de https://www.alldatasheet.com/datasheet-pdf/view/348702/MICROCHIP/PIC12F675.html.

[2] Diodes. (2018). *74HC959 Data Sheet*. Recuperado el 27 de agosto del 2024, de https://www.diodes.com/assets/Datasheets/74HC595.pdf

[3] Cypress Semiconductor Corporation. (2014). *7-Segment LED Controller Datasheet*. Recuperado el 27 de agosto del 2024, de https://www.infineon.com/dgdl/Infineon-LED7SEG_User_Module-Software%20Module%20Datasheets-v01_02-EN.pdf?fileId=8ac78c8c7d0d8da4017d0fa9f7d51aaf

[4] Microcontrollers Lab. (2020). *(Demo) 74HC595 interfacing with 4 digit 7 segment display and Pic Microcontroller*.Recuperado el 27 de agosto del 2024, de https://www.youtube.com/watch?v=Ct4vyh07bLM

5. Apéndices

Se presentán los datasheets de los componetes utilizados

PIC12F629/675 Data Sheet

8-Pin, Flash-Based 8-Bit
CMOS Microcontrollers

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Trademarks

Printed on recycled paper.

ISBN: 978-1-60932-160-4

Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

PIC12F629/675

8-Pin Flash-Based 8-Bit CMOS Microcontroller

High-Performance RISC CPU:

- Only 35 Instructions to Learn
  - All single-cycle instructions except branches
- Operating Speed:
  - DC – 20 MHz oscillator/clock input
  - DC – 200 ns instruction cycle
- Interrupt Capability
- 8-Level Deep Hardware Stack
- Direct, Indirect, and Relative Addressing modes

Special Microcontroller Features:

- Internal and External Oscillator Options
  - Precision Internal 4 MHz oscillator factory calibrated to ±1%
  - External Oscillator support for crystals and resonators
  - 5 μs wake-up from Sleep, 3.0V, typical
- Power-Saving Sleep mode
- Wide Operating Voltage Range – 2.0V to 5.5V
- Industrial and Extended Temperature Range
- Low-Power Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Brown-out Detect (BOD)
- Watchdog Timer (WDT) with Independent Oscillator for Reliable Operation
- Multiplexed $\overline{\text{MCLR}}$/Input Pin
- Interrupt-on-Pin Change
- Individual Programmable Weak Pull-ups
- Programmable Code Protection
- High Endurance Flash/EEPROM Cell
  - 100,000 write Flash endurance
  - 1,000,000 write EEPROM endurance
  - Flash/Data EEPROM Retention: > 40 years

Low-Power Features:

- Standby Current:
  - 1 nA @ 2.0V, typical
- Operating Current:
  - 8.5 μA @ 32 kHz, 2.0V, typical
  - 100 μA @ 1 MHz, 2.0V, typical
- Watchdog Timer Current
  - 300 nA @ 2.0V, typical
- Timer1 Oscillator Current:
  - 4 μA @ 32 kHz, 2.0V, typical

Peripheral Features:

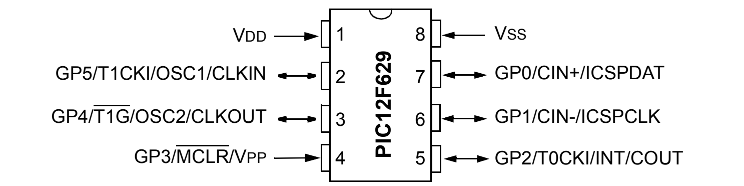
- 6 I/O Pins with Individual Direction Control
- High Current Sink/Source for Direct LED Drive
- Analog Comparator module with:
  - One analog comparator
  - Programmable on-chip comparator voltage reference (CV_{REF}) module
  - Programmable input multiplexing from device inputs
  - Comparator output 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 INTOSC mode selected
- In-Circuit Serial Programming™ (ICSP™) via two pins

Device	Program Memory	Data Memory		I/O	10-bit A/D (ch)	Comparators	Timers 8/16-bit
	Flash (words)	SRAM (bytes)	EEPROM (bytes)				
PIC12F629	1024	64	128	6	—	1	1/1
PIC12F675	1024	64	128	6	4	1	1/1

* 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.

Pin Diagrams

8-pin PDIP, SOIC, DFN-S, DFN



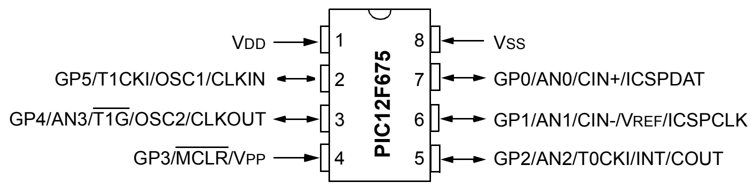


Table of Contents

TO OUR VALUED CUSTOMERS

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

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

Most Current Data Sheet

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

http://www.microchip.com

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

Errata

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

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

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

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

Customer Notification System

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

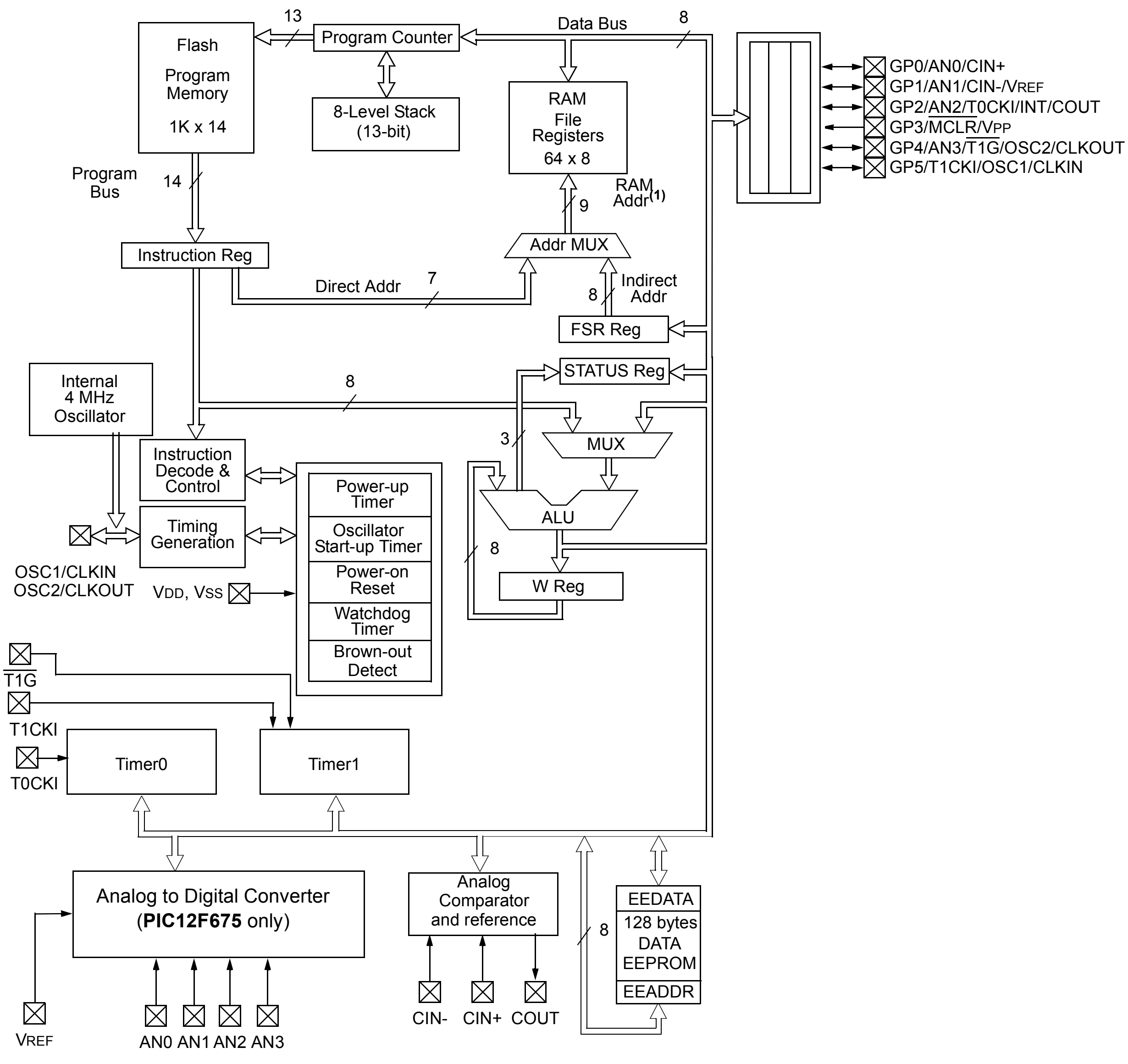
NOTES:

1.0 DEVICE OVERVIEW

This document contains device specific information for the PIC12F629/675. Additional information may be found in the PIC® 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, MLF-S and DFN 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



Note 1: Higher order bits are from STATUS register.

TABLE 1-1: PIC12F629/675 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
GP0/AN0/CIN+/ICSPDAT	GP0	TTL	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change
	AN0	AN		A/D Channel 0 input
	CIN+	AN		Comparator input
	ICSPDAT	TTL	CMOS	Serial programming I/O
GP1/AN1/CIN-/VREF/ ICSPCLK	GP1	TTL	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change
	AN1	AN		A/D Channel 1 input
	CIN-	AN		Comparator input
	VREF	AN		External voltage reference
	ICSPCLK	ST		Serial programming clock
GP2/AN2/T0CKI/INT/COUT	GP2	ST	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change
	AN2	AN		A/D Channel 2 input
	T0CKI	ST		TMR0 clock input
	INT	ST		External interrupt
	COUT		CMOS	Comparator output
GP3/$\overline{\text{MCLR}}$/VPP	GP3	TTL		Input port w/ interrupt-on-change
	$\overline{\text{MCLR}}$	ST		Master Clear
	VPP	HV		Programming voltage
GP4/AN3/$\overline{\text{T1G}}$/OSC2/ CLKOUT	GP4	TTL	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change
	AN3	AN		A/D Channel 3 input
	$\overline{\text{T1G}}$	ST		TMR1 gate
	OSC2		XTAL	Crystal/resonator
	CLKOUT		CMOS	FOSC/4 output
GP5/T1CKI/OSC1/CLKIN	GP5	TTL	CMOS	Bidirectional 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

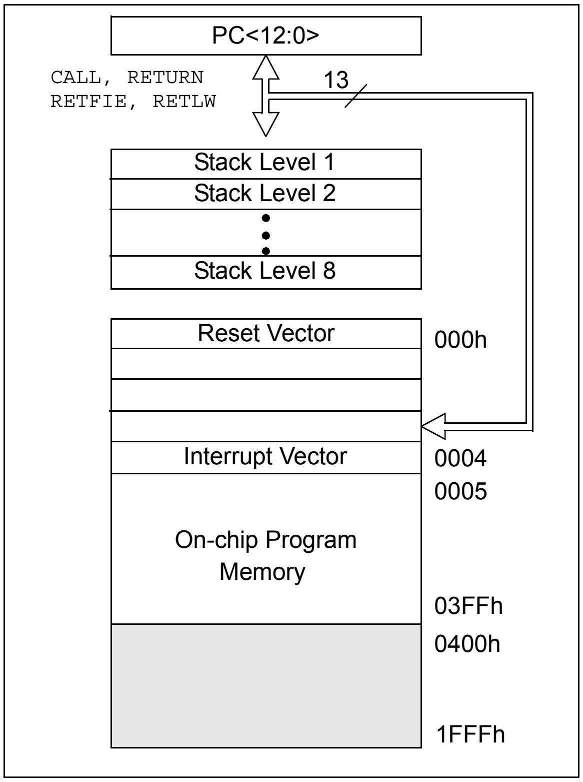
Legend: Shade = PIC12F675 only
TTL = TTL input buffer, ST = Schmitt Trigger input buffer

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 is 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 DSTEMP/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 "Indirect Addressing, INDF and FSR Registers"**).

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

Bank 0	File Address	Bank 1	File Address
Indirect addr.(1)	00h	Indirect addr.(1)	80h
TMR0	01h	OPTION_REG	81h
PCL	02h	PCL	82h
STATUS	03h	STATUS	83h
FSR	04h	FSR	84h
GPIO	05h	TRISIO	85h
	06h		86h
	07h		87h
	08h		88h
	09h		89h
PCLATH	0Ah	PCLATH	8Ah
INTCON	0Bh	INTCON	8Bh
PIR1	0Ch	PIE1	8Ch
	0Dh		8Dh
TMR1L	0Eh	PCON	8Eh
TMR1H	0Fh		8Fh
T1CON	10h	OSCCAL	90h
	11h		91h
	12h		92h
	13h		93h
	14h		94h
	15h	WPU	95h
	16h	IOC	96h
	17h		97h
	18h		98h
CMCON	19h	VRCON	99h
	1Ah	EEDATA	9Ah
	1Bh	EEADR	9Bh
	1Ch	EECON1	9Ch
	1Dh	EECON2(1)	9Dh
ADRESH(2)	1Eh	ADRESL(2)	9Eh
ADCON0(2)	1Fh	ANSEL(2)	9Fh
General Purpose Registers 64 Bytes	20h–5Fh	accesses 20h-5Fh	A0h–DFh
	60h–7Fh		E0h–FFh

Bank 0 Bank 1

▭ Unimplemented data memory locations, read as '0'.
1: Not a physical register.
2: PIC12F675 only.

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, BOD	Page
Bank 0											
00h	INDF(1)	Addressing this Location uses Contents of FSR to Address Data Memory								0000 0000	20,61
01h	TMR0	Timer0 Module's Register								xxxx xxxx	29
02h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	19
03h	STATUS	IRP(2)	RP1(2)	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0001 1xxx	14
04h	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	20
05h	GPIO	—	—	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	--xx xxxx	21
06h	—	Unimplemented								—	—
07h	—	Unimplemented								—	—
08h	—	Unimplemented								—	—
09h	—	Unimplemented								—	—
0Ah	PCLATH	—	—	—	Write Buffer for Upper 5 bits of Program Counter					---0 0000	19
0Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	15
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	17
0Dh	—	Unimplemented								—	—
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit Timer1								xxxx xxxx	32
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit Timer1								xxxx xxxx	32
10h	T1CON	—	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	-000 0000	35
11h	—	Unimplemented								—	—
12h	—	Unimplemented								—	—
13h	—	Unimplemented								—	—
14h	—	Unimplemented								—	—
15h	—	Unimplemented								—	—
16h	—	Unimplemented								—	—
17h	—	Unimplemented								—	—
18h	—	Unimplemented								—	—
19h	CMCON	—	COUT	—	CINV	CIS	CM2	CM1	CM0	-0-0 0000	38
1Ah	—	Unimplemented								—	—
1Bh	—	Unimplemented								—	—
1Ch	—	Unimplemented								—	—
1Dh	—	Unimplemented								—	—
1Eh	ADRESH(3)	Most Significant 8 bits of the Left Shifted A/D Result or 2 bits of the Right Shifted Result								xxxx xxxx	44
1Fh	ADCON0(3)	ADFM	VCFG	—	—	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON	00-- 0000	45,61

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

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

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, BOD	Page
Bank 1											
80h	INDF(1)	Addressing this Location uses Contents of FSR to Address Data Memory								0000 0000	20,61
81h	OPTION_REG	$\overline{GPPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	14,31
82h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	19
83h	STATUS	IRP(2)	RP1(2)	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	14
84h	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	20
85h	TRISIO	—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0	--11 1111	21
86h	—	Unimplemented								—	—
87h	—	Unimplemented								—	—
88h	—	Unimplemented								—	—
89h	—	Unimplemented								—	—
8Ah	PCLATH	—	—	—	Write Buffer for Upper 5 bits of Program Counter					---0 0000	19
8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	15
8Ch	PIE1	EEIE	ADIE	—	—	CMIE	—	—	TMR1IE	00-- 0--0	16
8Dh	—	Unimplemented								—	—
8Eh	PCON	—	—	—	—	—	—	$\overline{POR}$	$\overline{BOD}$	---- --0x	18
8Fh	—	Unimplemented								—	—
90h	OSCCAL	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	—	1000 00--	18
91h	—	Unimplemented								—	—
92h	—	Unimplemented								—	—
93h	—	Unimplemented								—	—
94h	—	Unimplemented								—	—
95h	WPU	—	—	WPU5	WPU4	—	WPU2	WPU1	WPU0	--11 -111	21
96h	IOC	—	—	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0	--00 0000	23
97h	—	Unimplemented								—	—
98h	—	Unimplemented								—	—
99h	VRCON	VREN	—	VRR	—	VR3	VR2	VR1	VR0	0-0- 0000	42
9Ah	EEDATA	Data EEPROM Data Register								0000 0000	49
9Bh	EEADR	—	Data EEPROM Address Register							-000 0000	49
9Ch	EECON1	—	—	—	—	WRERR	WREN	WR	RD	---- x000	50
9Dh	EECON2(1)	EEPROM Control Register 2								---- ----	50
9Eh	ADRESL(3)	Least Significant 2 bits of the Left Shifted A/D Result of 8 bits or the Right Shifted Result								xxxx xxxx	44
9Fh	ANSEL(3)	—	ADCS2	ADCS1	ADCS0	ANS3	ANS2	ANS1	ANS0	-000 1111	46,61

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

Note 1: This is not a physical register.
2: These bits are reserved and should always be maintained as '0'.
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 $\overline{\text{TO}}$ and $\overline{\text{PD}}$ 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 $\overline{\text{Borrow}}$ and $\overline{\text{Digit Borrow}}$ 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	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C
bit 7							bit 0

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

bit 7 **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 **$\overline{\text{TO}}$:** Time-out bit
`1` = After power-up, `CLRWDT` instruction, or `SLEEP` instruction
`0` = A WDT Time-out occurred

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

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

bit 1 **DC**: Digit carry/$\overline{\text{borrow}}$ 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/$\overline{\text{borrow}}$ 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 $\overline{\text{borrow}}$ 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.

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 "Prescaler"**.

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

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

bit 7 **$\overline{\text{GPPU}}$:** 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

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	T0IE	INTE	GPIE	T0IF	INTF	GPIF
bit 7							bit 0

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

bit 7 **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 **T0IE:** 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)
1 = Enables the GPIO port change interrupt
0 = Disables the GPIO port change interrupt

bit 2 **T0IF:** TMR0 Overflow Interrupt Flag bit(2)
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: IOC register must also be enabled to enable an interrupt-on-change.
2: T0IF bit is set when TIMER0 rolls over. TIMER0 is unchanged on Reset and should be initialized before clearing T0IF bit.

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

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

bit 7 **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

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

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

bit 7	**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

2.2.2.6 PCON Register

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

- Power-on Reset (POR)
- Brown-out Detect (BOD)
- Watchdog Timer Reset (WDT)
- External $\overline{\text{MCLR}}$ 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
—	—	—	—	—	—	$\overline{\text{POR}}$	$\overline{\text{BOD}}$
bit 7							bit 0

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

bit 7-2 **Unimplemented:** Read as ‘0’

bit 1 **$\overline{\text{POR}}$:** 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 **$\overline{\text{BOD}}$:** Brown-out Detect Status bit
1 = No Brown-out Detect occurred
0 = A Brown-out Detect occurred (must be set in software after a Brown-out Detect occurs)

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

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

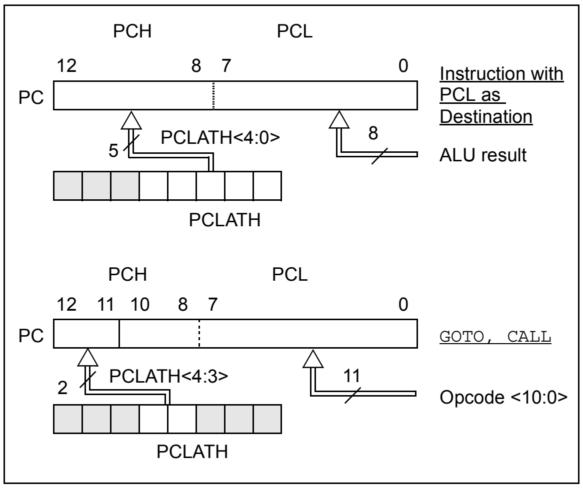
bit 7-2 **CAL5:CAL0:** 6-bit Signed Oscillator Calibration bits
111111 = Maximum frequency
100000 = Center frequency
000000 = Minimum frequency

bit 1-0 **Unimplemented:** Read as ‘0’

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> $\rightarrow$ PCH). The lower example in Figure 2-3 shows how the PC is loaded during a `CALL` or `GOTO` instruction (PCLATH<4:3> $\rightarrow$ 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 PC (`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.

2.4 Indirect Addressing, INDF and FSR Registers

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

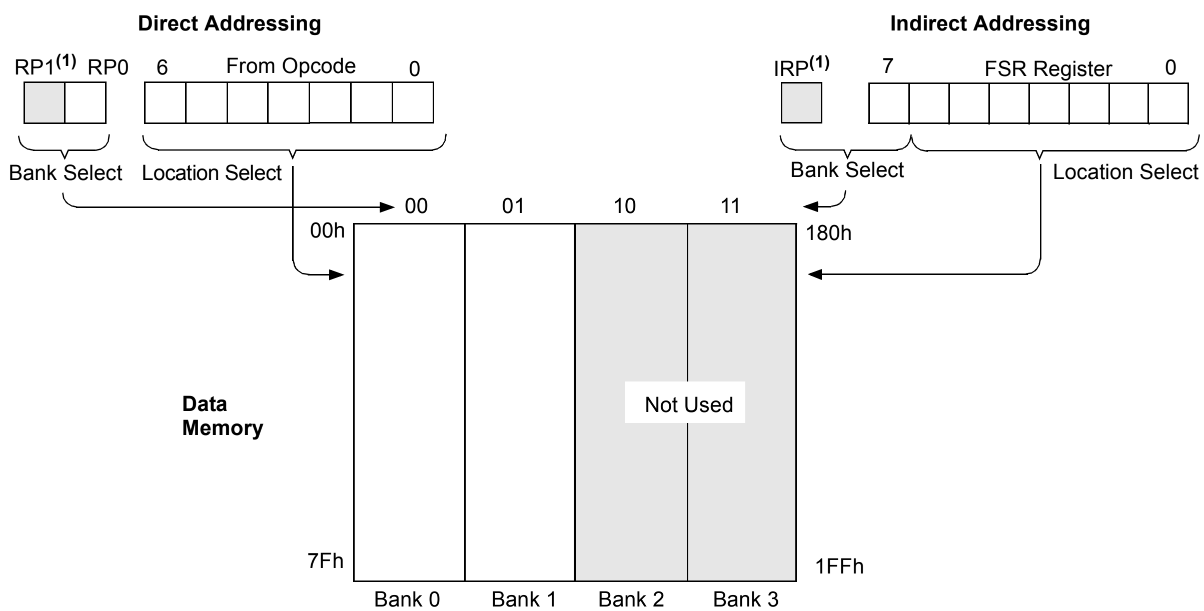
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select Register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no operation (although Status 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 2-2.

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 2-2: DIRECT/INDIRECT ADDRESSING PIC12F629/675



For memory map detail see Figure 2-2.

Note 1: The RP1 and IRP bits are reserved; always maintain these bits clear.

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 PIC® Mid-Range Reference Manual, (DS33023).

3.1 GPIO and the TRISIO Registers

GPIO is an 6-bit wide, bidirectional 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 High-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 TRISIO 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. I/O pins configured as analog inputs always read '0'.

> **Note:** The ANSEL (9Fh) and CMCON (19h) registers (9Fh) must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read '0'. The ANSEL register is defined for the PIC12F675.

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
CLRF    ANSEL           ;Digital I/O
MOVLW   0Ch             ;Set GP<3:2> as inputs
MOVWF   TRISIO          ;and set GP<5:4,1:0>
                        ;as outputs
```

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 3-3. 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 $\overline{\text{GPPU}}$ bit (OPTION<7>).

REGISTER 3-1: GPIO: GPIO REGISTER (ADDRESS: 05h)

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0
bit 7							bit 0

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

bit 7-6 **Unimplemented**: Read as '0'

bit 5-0 **GPIO<5:0>**: General Purpose I/O pin
1 = Port pin is >V_{IH}
0 = Port pin is <V_{IL}

REGISTER 3-2: TRISIO: GPIO TRI-STATE REGISTER (ADDRESS: 85h)

U-0	U-0	R/W-1	R/W-1	R-1	R/W-1	R/W-1	R/W-1
—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0
bit 7							bit 0

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

bit 7-6 **Unimplemented**: Read as '0'

bit 5-0 **TRISIO<5:0>**: General Purpose I/O Tri-State Control bit
1 = GPIO pin configured as an input (tri-stated)
0 = GPIO pin configured as an output

Note: TRISIO<3> always reads '1'.

REGISTER 3-3: 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

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

bit 7-6 **Unimplemented**: Read as '0'

bit 5-4 **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.

2: The weak pull-up device is automatically disabled if the pin is in Output mode (TRISIO = 0).

3.2.2 INTERRUPT-ON-CHANGE

Each of the GPIO pins is individually configurable as an interrupt-on-change pin. Control bits IOC enable or disable the interrupt function for each pin. Refer to Register 3-4. 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, 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 3-4: IOC: 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
—	—	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0
bit 7							bit 0

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

bit 7-6 **Unimplemented**: Read as '0'

bit 5-0 **IOC<5:0>:** Interrupt-on-Change GPIO Control bits
1 = Interrupt-on-change enabled
0 = Interrupt-on-change disabled

Note 1: Global Interrupt Enable (GIE) must be enabled for individual interrupts to be recognized.

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:

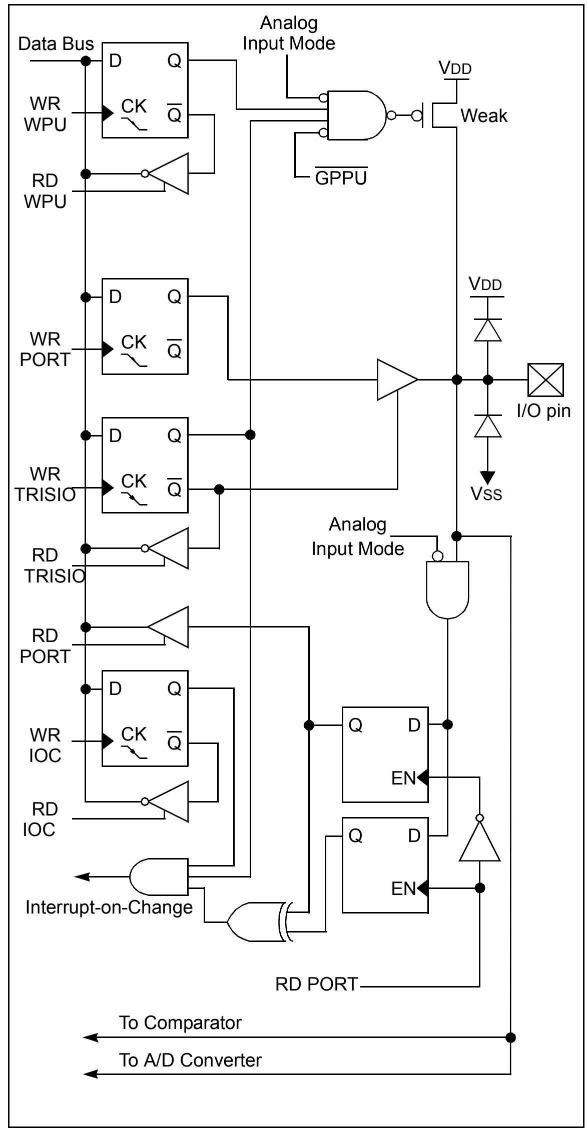
- a general purpose I/O
- an analog input for the A/D (PIC12F675 only)
- an analog input to the comparator

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

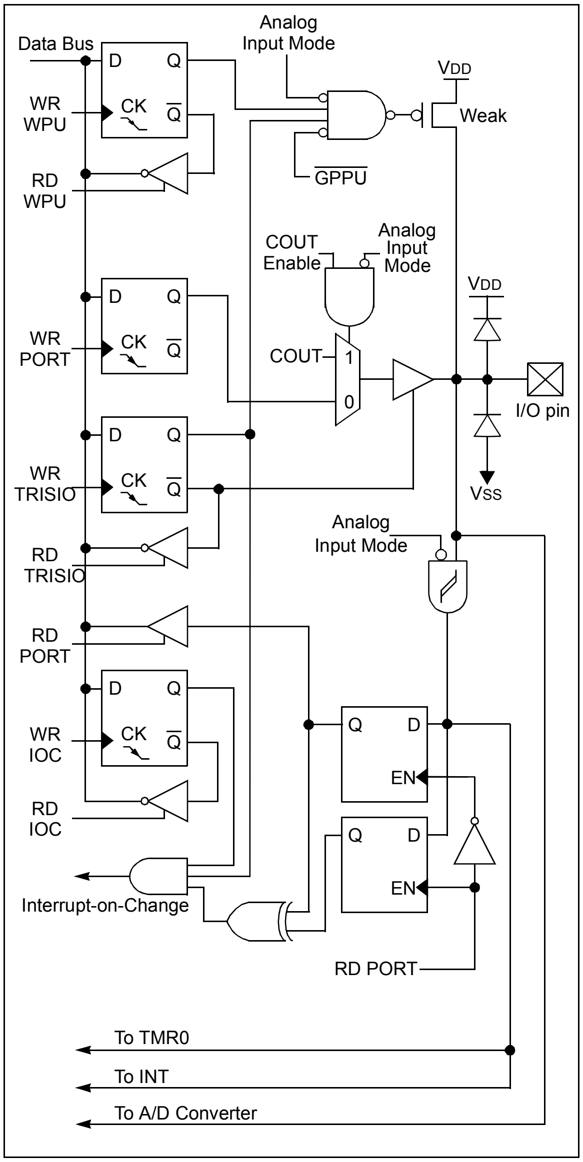


3.3.3 GP2/AN2/T0CKI/INT/COUT

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)
- the clock input for TMR0
- an external edge triggered interrupt
- a digital output from the comparator

FIGURE 3-2: BLOCK DIAGRAM OF GP2

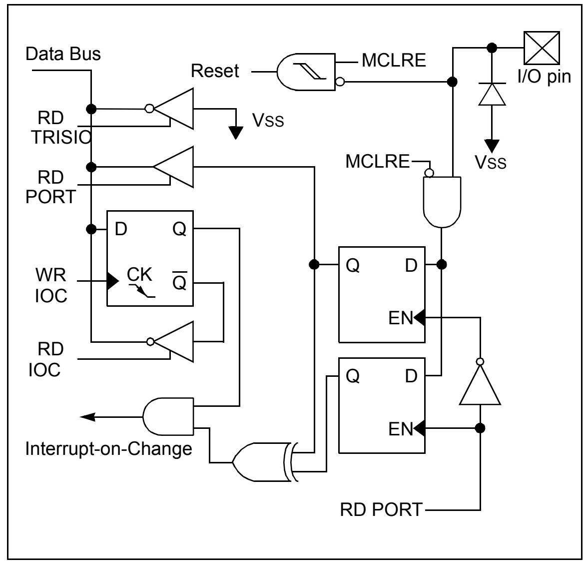


3.3.4 GP3/$\overline{\text{MCLR}}$/VPP

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

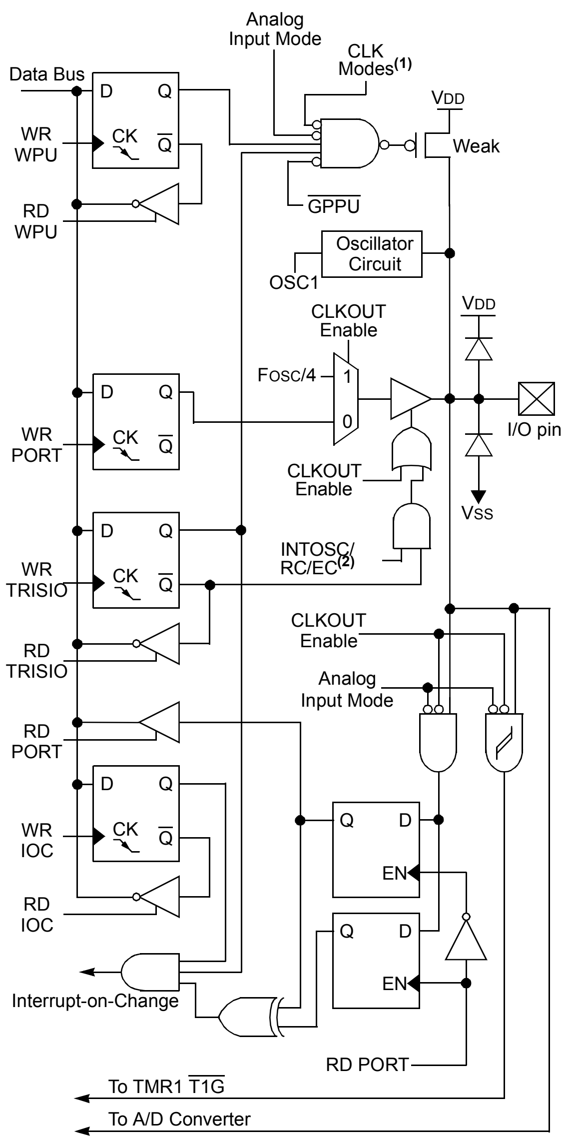


3.3.5 GP4/AN3/$\overline{\text{T1G}}$/OSC2/CLKOUT

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



Note 1: CLK modes are XT, HS, LP, LPTMR1 and CLKOUT Enable.

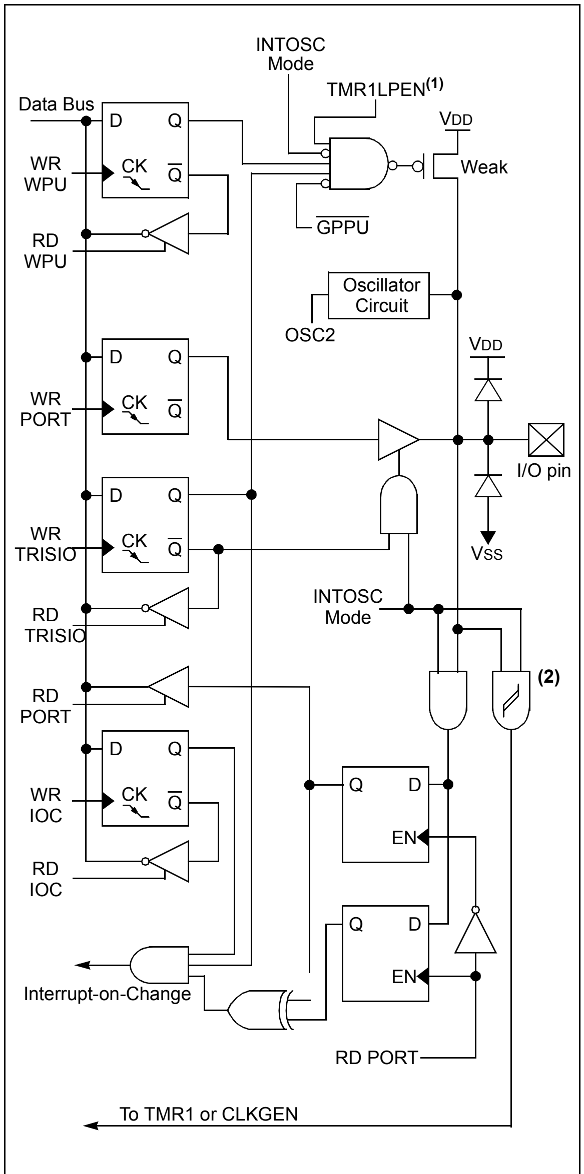
2: With CLKOUT option.

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



Note 1: Timer1 LP Oscillator enabled

2: When using Timer1 with LP oscillator, the Schmitt Trigger is by-passed.

TABLE 3-2: 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, BOD	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	—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0	`--11 1111`	`--11 1111`
95h	WPU	—	—	WPU5	WPU4	—	WPU2	WPU1	WPU0	`--11 -111`	`--11 -111`
96h	IOC	—	—	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0	`--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.

NOTES:

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 PIC® 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.

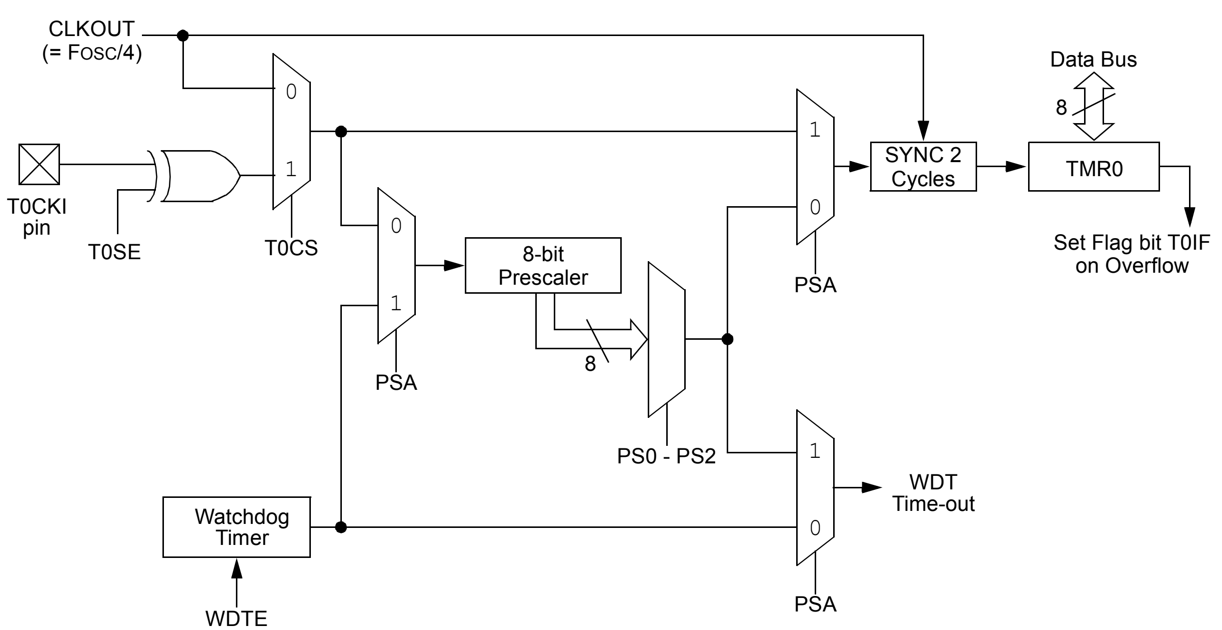
Counter mode is selected by setting the T0CS bit (OPTION_REG<5>). In this mode, the Timer0 module will increment either on every rising or falling edge of pin GP2/T0CKI. The incrementing edge is determined by the source edge (T0SE) control bit (OPTION_REG<4>). Clearing the T0SE bit selects the rising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

Note:	Counter mode has specific external clock requirements. Additional information on these requirements is available in the PIC® 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



Note 1: T0SE, T0CS, PSA, PS0-PS2 are bits in the OPTION register.

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 accomplished 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.

Note:	The ANSEL (9Fh) and CMCON (19h) registers must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read '0'. The ANSEL register is defined for the PIC12F675.

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
$\overline{\text{GPPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

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

bit 7 **$\overline{\text{GPPU}}$:** 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/T0CK 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

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, BOD	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	—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0	--11 1111	--11 1111

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

5.0 TIMER1 MODULE WITH GATE CONTROL

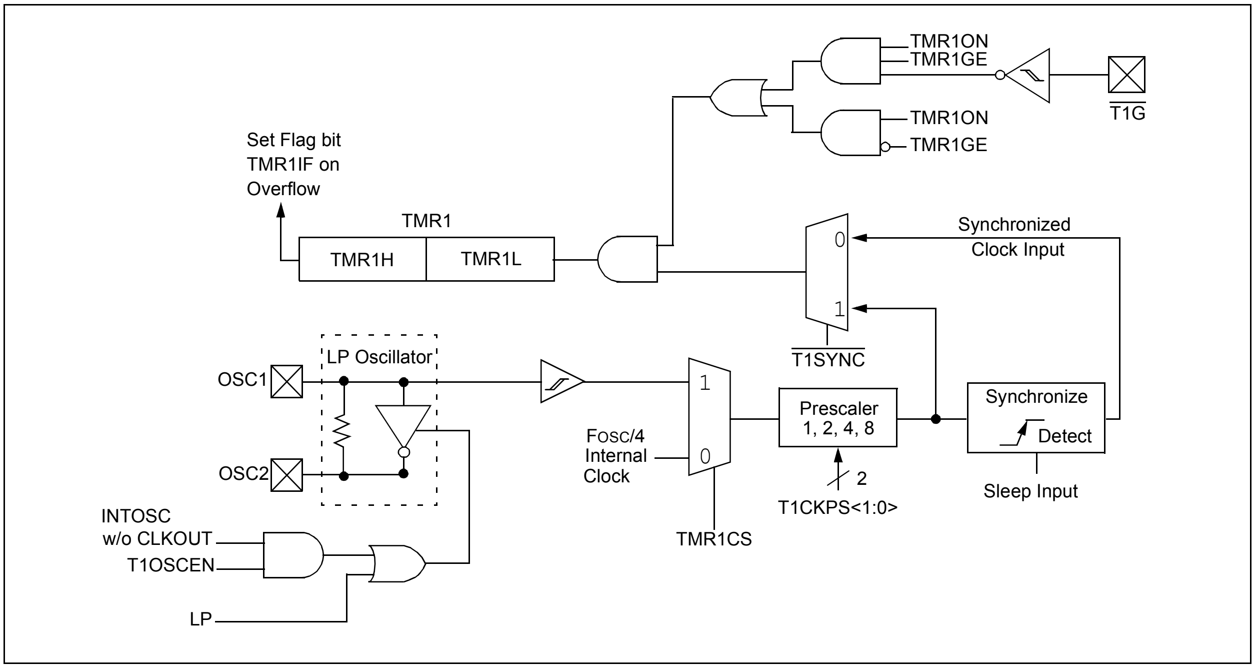
The PIC12F629/675 devices have a 16-bit timer. Figure 5-1 shows the basic block diagram of the Timer1 module. Timer1 has the following features:

- 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 ($\overline{\text{T1G}}$)
- 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 PIC® 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 $\overline{\text{T1G}}$ input.

If an external clock oscillator is needed (and the microcontroller is using the INTOSC 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>).

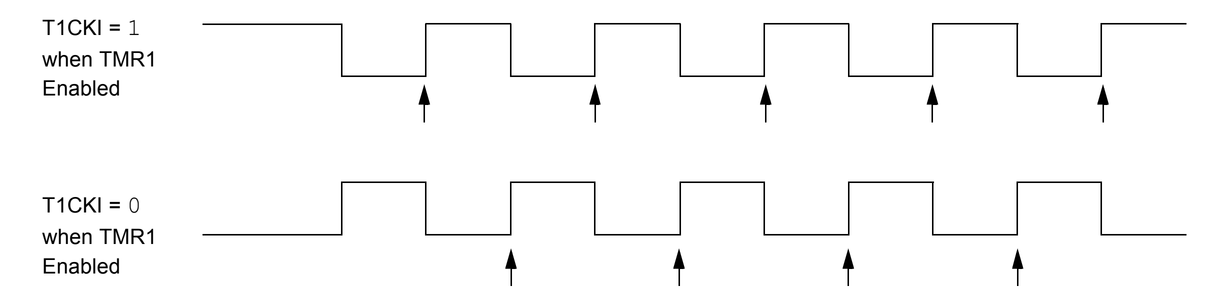
The interrupt is cleared by clearing the TMR1IF in the Interrupt Service Routine.

Note:	The TMR1H:TTMR1L 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



Note 1: Arrows indicate counter increments.
2: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge of the clock.

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	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON
bit 7							bit 0

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

bit 7 **Unimplemented:** Read as ‘0’

bit 6 **TMR1GE:** Timer1 Gate Enable bit
If TMR1ON = 0:
This bit is ignored
If TMR1ON = 1:
1 = Timer1 is on if $\overline{\text{T1G}}$ 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 **$\overline{\text{T1SYNC}}$:** 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 (FOSC/4)

bit 0 **TMR1ON:** Timer1 On bit
1 = Enables Timer1
0 = Stops Timer1

5.4 Timer1 Operation in Asynchronous Counter Mode

If control bit $\overline{\text{T1SYNC}}$ (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 "Reading and Writing Timer1 in Asynchronous Counter Mode"**).

> **Note:** The ANSEL (9Fh) and CMCON (19h) registers must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read '0'. The ANSEL register is defined for the PIC12F675.

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 PIC® 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 37 kHz. It will continue to run during Sleep. It is primarily intended for a 32 kHz crystal. Table 9-2 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.

While enabled, TRISIO4 and TRISIO5 are set. GP4 and GP5 read '0' and TRISIO4 and TRISIO5 are read '1'.

> **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.

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 5-1: 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, BOD	Value on all other Resets
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
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	$\overline{\text{T1SYNC}}$	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.

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 Comparator 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

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

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

bit 6 **COUT**: Comparator Output bit
When CINV = 0:
1 = VIN+ > VIN-
0 = VIN+ < VIN-
When CINV = 1:
1 = VIN+ < VIN-
0 = VIN+ > VIN-

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 = VIN- connects to CIN+
0 = VIN- connects to CIN-

bit 2-0 **CM2:CM0**: Comparator Mode bits
Figure 6-2 shows the Comparator modes and CM2:CM0 bit settings

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 VIN+ is less than the analog input VIN-, the output of the comparator is a digital low level. When the analog input at VIN+ is greater than the analog input VIN-, 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.

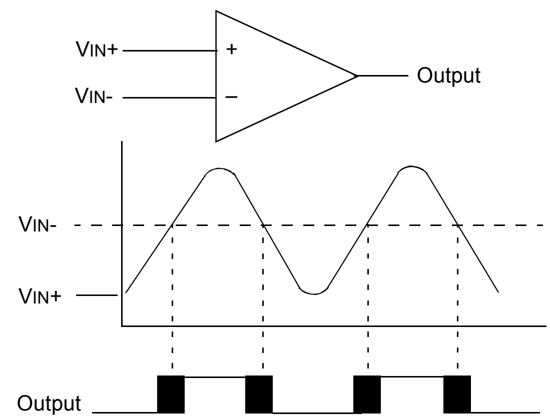
Note:	To use CIN+ and CIN- pins as analog inputs, the appropriate bits must be programmed in the CMCON (19h) register.

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	COUT
VIN- > VIN+	0	0
VIN- < VIN+	0	1
VIN- > VIN+	1	1
VIN- < VIN+	1	0

FIGURE 6-1: SINGLE COMPARATOR



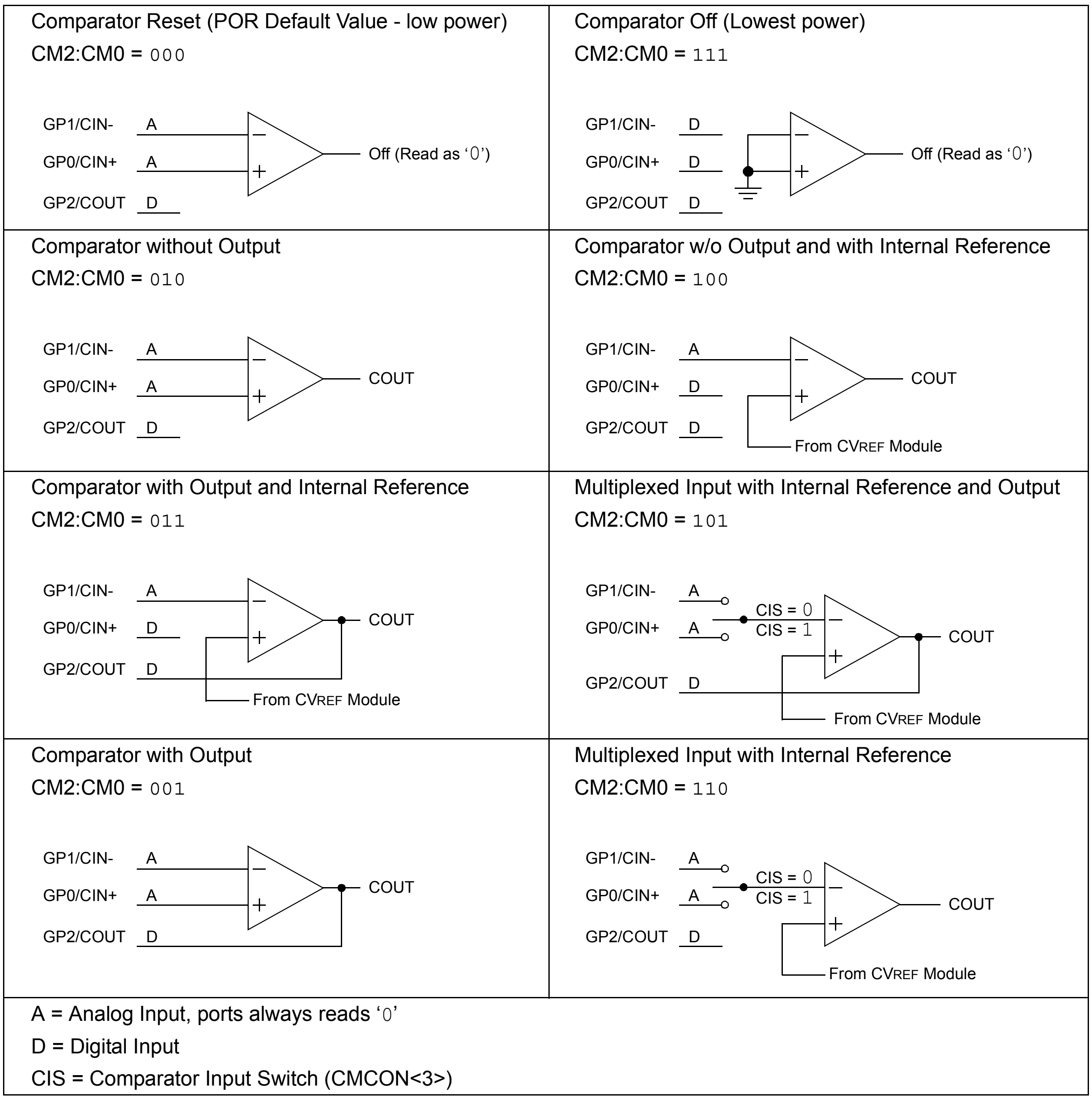
Note: CINV bit (CMCON<4>) is clear.

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 Comparator 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 "Electrical Specifications"**.

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

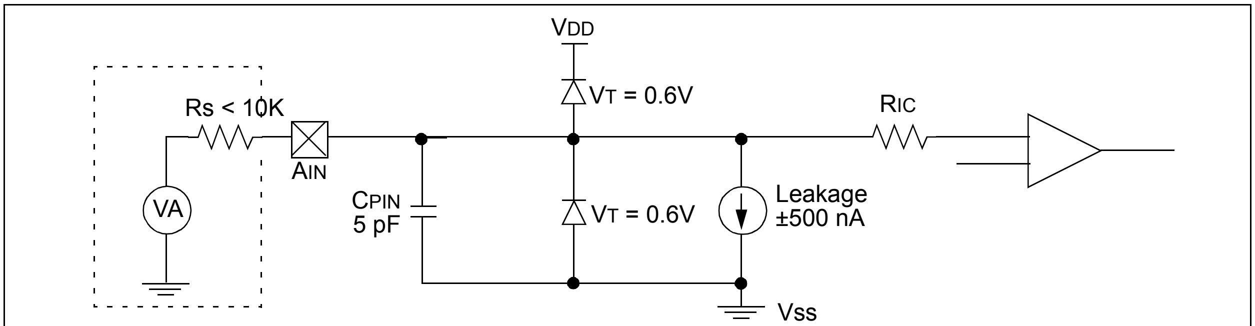
FIGURE 6-2: COMPARATOR I/O OPERATING MODES



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 latch-up may occur. A maximum source impedance of 10 kΩ is recommended for the analog sources. Any external component connected to an analog input pin, such as a capacitor or a Zener diode, should have very little leakage current.

FIGURE 6-3: ANALOG INPUT MODE



Legend:

CPIN	= Input Capacitance
VT	= Threshold Voltage
ILEAKAGE	= Leakage Current at the pin due to Various Junctions
RIC	= Interconnect Resistance
RS	= Source Impedance
VA	= Analog Voltage

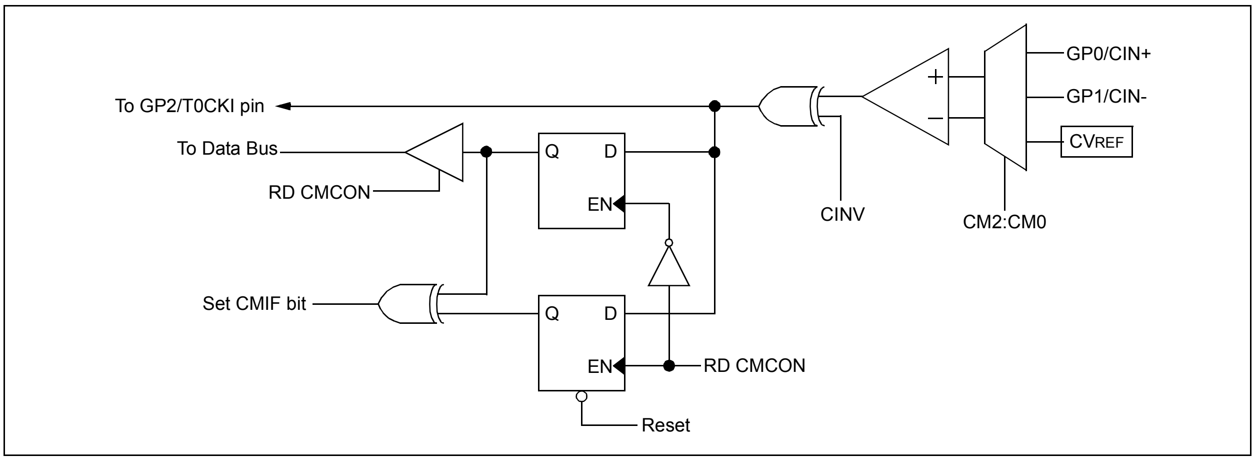
6.4 Comparator Output

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.

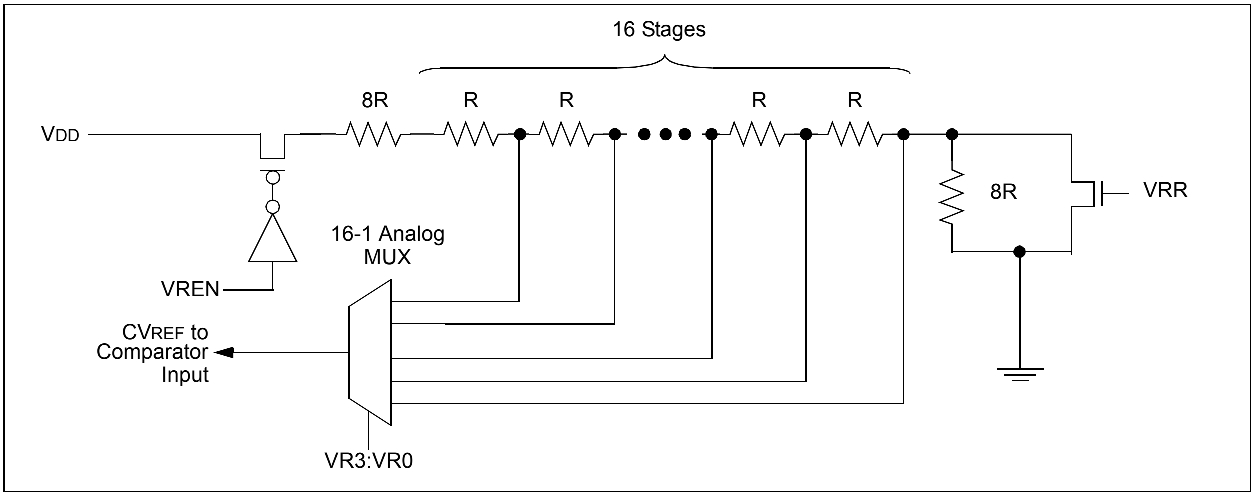
The following equations determine the output voltages:

VRR = 1 (low range): $CV_{REF} = (VR3:VR0 / 24) \times V_{DD}$
VRR = 0 (high range): $CV_{REF} = (V_{DD} / 4) + (VR3:VR0 \times V_{DD} / 32)$

6.5.2 VOLTAGE REFERENCE ACCURACY/ERROR

The full range of V_{SS} to V_{DD} 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 CV_{REF} from approaching V_{SS} or V_{DD}. The Voltage Reference is V_{DD} derived and therefore, the CV_{REF} output changes with fluctuations in V_{DD}. The tested absolute accuracy of the Comparator Voltage Reference can be found in **Section 12.0 "Electrical Specifications"**.

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 12-7).

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.

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

Legend:

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

bit 7 **VREN:** CVREF Enable bit
1 = CVREF 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

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

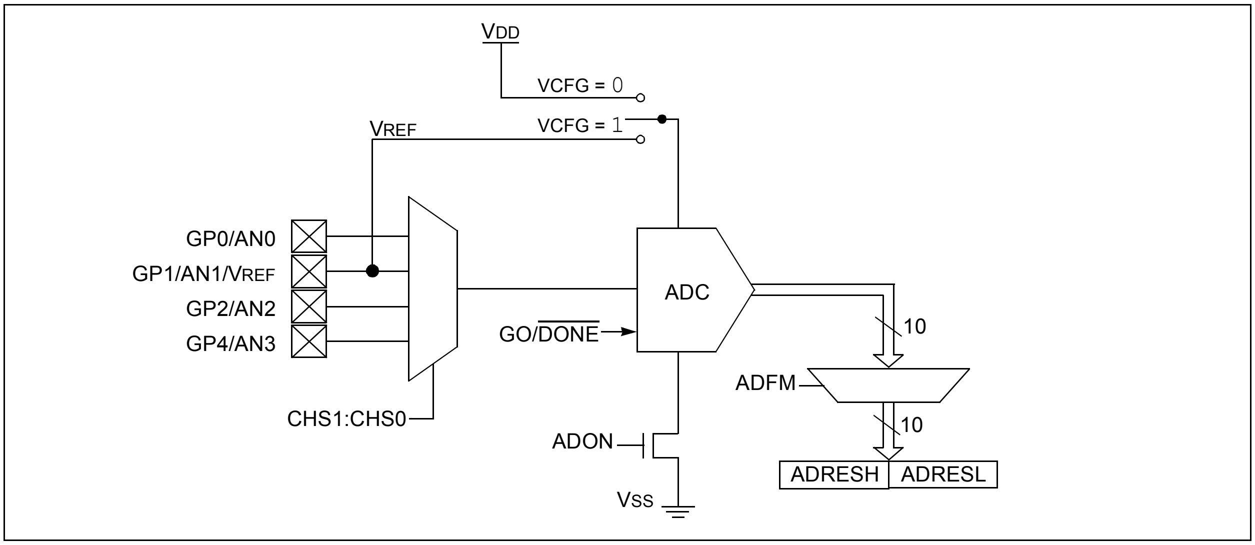
Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOD	Value on all other Resets
0Bh/8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	0000 000u
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	00-- 0--0
19h	CMCON	—	COUT	—	CINV	CIS	CM2	CM1	CM0	-0-0 0000	-0-0 0000
8Ch	PIE1	EEIE	ADIE	—	—	CMIE	—	—	TMR1IE	00-- 0--0	00-- 0--0
85h	TRISIO	—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0	--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 μs. Table 7-1 shows a few TAD calculations for selected frequencies.

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[2]	400 ns[2]	500 ns[2]	1.6 μs
4 TOSC	100	200 ns[2]	800 ns[2]	1.0 μs[2]	3.2 μs
8 TOSC	001	400 ns[2]	1.6 μs	2.0 μs	6.4 μs
16 TOSC	101	800 ns[2]	3.2 μs	4.0 μs	12.8 μs[3]
32 TOSC	010	1.6 μs	6.4 μs	8.0 μs[3]	25.6 μs[3]
64 TOSC	110	3.2 μs	12.8 μs[3]	16.0 μs[3]	51.2 μs[3]
A/D RC	x11	2 - 6 μs[1,4]	2 - 6 μs[1,4]	2 - 6 μs[1,4]	2 - 6 μs[1,4]

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/$\overline{DONE}$ bit (ADCON0<1>). When the conversion is complete, the A/D module:

- Clears the GO/$\overline{DONE}$ bit
- Sets the ADIF flag (PIR1<6>)
- Generates an interrupt (if enabled)

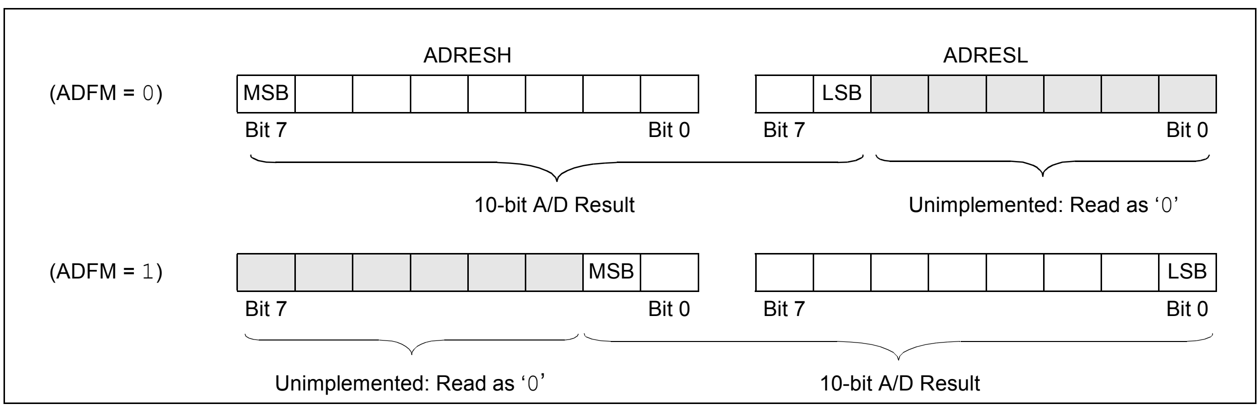
If the conversion must be aborted, the GO/$\overline{DONE}$ bit can be cleared in software. The ADRESH:ADRESL registers will not be updated with the partially complete A/D conversion sample. Instead, the ADRESH:ADRESL registers will retain the value of the previous conversion. After an aborted conversion, 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/$\overline{DONE}$ 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/$\overline{\text{DONE}}$	ADON
bit 7							bit 0

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

bit 7 **ADFM:** A/D Result Formed Select bit
1 = Right justified
0 = Left justified

bit 6 **VCFG:** Voltage Reference bit
1 = V_{REF} pin
0 = V_{DD}

bit 5-4 **Unimplemented**: Read as '0'

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/$\overline{\text{DONE}}$:** 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

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

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

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

bit 6-4 **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.)
1 = Analog input; pin is assigned as analog input(1)
0 = Digital I/O; pin is assigned to port or special function

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.

7.2 A/D Acquisition Requirements

For the A/D converter to meet its specified accuracy, the charge holding capacitor (C_{HOLD}) 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 (R_S) and the internal sampling switch (R_{SS}) impedance directly affect the time required to charge the capacitor C_{HOLD}. The sampling switch (R_{SS}) impedance varies over the device voltage (V_{DD}), see Figure 7-3. **The maximum recommended impedance for analog sources is 10 kΩ.** As the impedance is decreased, the acquisition time may be decreased. After the analog input channel is selected (changed), this acquisition must be done before the conversion can be started.

To calculate the minimum acquisition time, Equation 7-1 may be used. This equation assumes that 1/2 LSb error is used (1024 steps for the A/D). The 1/2 LSb error is the maximum error allowed for the A/D to meet its specified resolution.

To calculate the minimum acquisition time, T_{ACQ}, see the PIC® Mid-Range Reference Manual (DS33023).

EQUATION 7-1: ACQUISITION TIME

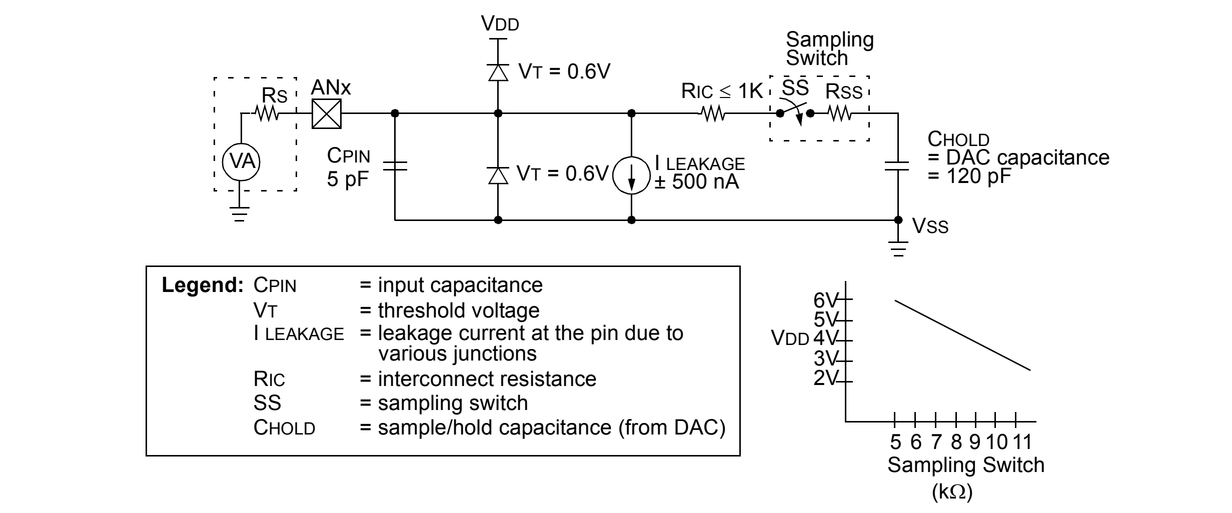
$$
\begin{aligned}
T_{ACQ} &= \text{Amplifier Settling Time} + \\
&\quad\ \text{Hold Capacitor Charging Time} + \\
&\quad\ \text{Temperature Coefficient} \\
&= T_{AMP} + T_C + T_{COFF} \\
&= 2\mu s + T_C + [(\text{Temperature} -25°C)(0.05\mu s/°C)] \\
T_C &= C_{HOLD}\,(R_{IC} + R_{SS} + R_S)\ \text{In}(1/2047) \\
&= -\,120pF\,(1k\Omega + 7k\Omega + 10k\Omega)\ \text{In}(0.0004885) \\
&= 16.47\mu s \\
T_{ACQ} &= 2\mu s + 16.47\mu s + [(50°C -25°C)(0.05\mu s/°C) \\
&= 19.72\mu s
\end{aligned}
$$

Note 1: The reference voltage (V_{REF}) has no effect on the equation, since it cancels itself out.

2: The charge holding capacitor (C_{HOLD}) is not discharged after each conversion.

3: The maximum recommended impedance for analog sources is 10 kΩ. This is required to meet the pin leakage specification.

FIGURE 7-3: ANALOG INPUT MODEL



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 GO/$\overline{\text{DONE}}$ 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, BOD	Value on all other Resets
05h	GPIO	—	—	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	`--xx xxxx`	`--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 xxxx`	`uuuu uuuu`
1Fh	ADCON0	ADFM	VCFG	—	—	CHS1	CHS0	GO	ADON	`00-- 0000`	`00-- 0000`
85h	TRISIO	—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0	`--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 xxxx`	`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.

8.0 DATA EEPROM MEMORY

The EEPROM data memory is readable and writable during normal operation (full V_{DD} 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 PIC® 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

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

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

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

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

bit 7 **Unimplemented**: Should be set to ‘0’

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

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

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

bit 7-4 **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

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
```

Required Sequence

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
BTFSS  STATUS,Z       ;Is data the same
GOTO   WRITE_ERR      ;No, handle error
:                     ;Yes, continue
```

8.5.1 USING THE DATA EEPROM

The data EEPROM is a high-endurance, byte addressable array that has been optimized for the storage of frequently changing information (e.g., program variables or other data that are updated often). Frequently changing values will typically be updated more often than specifications D120 or D120A. If this is not the case, an array refresh must be performed. For this reason, variables that change infrequently (such as constants, IDs, calibration, etc.) should be stored in Flash program memory.

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

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 POR, BOD	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(1)	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 Detect (BOD)
- 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 INTOSC option saves system cost while the LP crystal option saves power. A set of Configuration bits are used to select various options (see Register 9.2).

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 9.2. 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
BG1	BG0	—	—	—	$\overline{\text{CPD}}$	$\overline{\text{CP}}$	BODEN	MCLRE	$\overline{\text{PWRTE}}$	WDTE	F0SC2	F0SC1	F0SC0
bit 13													bit 0

Legend:			
P = Programmed using ICSP™			
R = Readable bit	Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	1 = bit is set	0 = bit is cleared	x = bit is unknown

bit 13-12	**BG1:BG0:** Bandgap Calibration bits for BOD and POR voltage[(1)] 00 = Lowest bandgap voltage 11 = Highest bandgap voltage
bit 11-9	**Unimplemented**: Read as '0'
bit 8	**$\overline{\text{CPD}}$:** Data Code Protection bit[(2)] 1 = Data memory code protection is disabled 0 = Data memory code protection is enabled
bit 7	**$\overline{\text{CP}}$**: Code Protection bit[(3)] 1 = Program Memory code protection is disabled 0 = Program Memory code protection is enabled
bit 6	**BODEN**: Brown-out Detect Enable bit[(4)] 1 = BOD enabled 0 = BOD disabled
bit 5	**MCLRE**: GP3/$\overline{\text{MCLR}}$ Pin Function Select bit[(5)] 1 = GP3/$\overline{\text{MCLR}}$ pin function is $\overline{\text{MCLR}}$ 0 = GP3/$\overline{\text{MCLR}}$ pin function is digital I/O, $\overline{\text{MCLR}}$ internally tied to V_{DD}
bit 4	**$\overline{\text{PWRTE}}$**: 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	**FOSC2:FOSC0**: 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 GP5/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 as specified in the PIC12F629/675 Programming Specification. These bits are reflected in an export of the Configuration Word. Microchip Development Tools maintain all Calibration bits to factory settings.

2: The entire data EEPROM will be erased when the code protection is turned off.

3: The entire program memory will be erased, including OSCCAL value, when the code protection is turned off.

4: Enabling Brown-out Detect does not automatically enable Power-up Timer.

5: When $\overline{\text{MCLR}}$ is asserted in INTOSC or RC mode, the internal clock oscillator is disabled.

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:

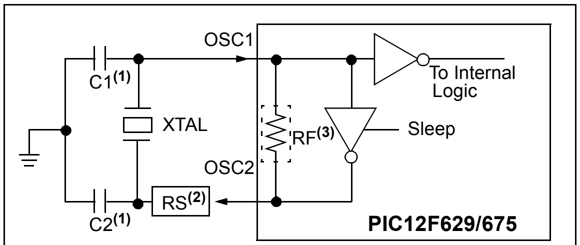
- LP Low-Power Crystal
- XT Crystal/Resonator
- HS High-Speed Crystal/Resonator
- RC External Resistor/Capacitor (2 modes)
- INTOSC Internal Oscillator (2 modes)
- EC External Clock In

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

9.2.2 CRYSTAL OSCILLATOR / CERAMIC RESONATORS

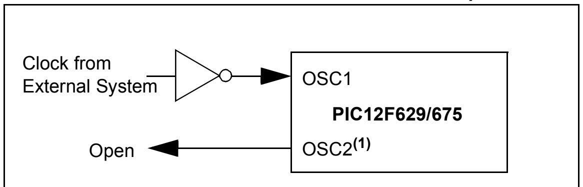
In XT, LP or HS modes a crystal or ceramic resonator is connected to the OSC1 and OSC2 pins to establish oscillation (see Figure 9-1). The PIC12F629/675 oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may yield a frequency outside of the crystal manufacturers 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



Note 1: See Table 9-1 and Table 9-2 for recommended values of C1 and C2.
2: A series resistor may be required for AT strip cut crystals.
3: RF varies with the Oscillator mode selected (Approx. value = 10 MΩ).

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



Note 1: Functions as GP4 in EC Osc mode.

TABLE 9-1: CAPACITOR SELECTION FOR CERAMIC RESONATORS

Ranges Characterized:			
Mode	**Freq.**	**OSC1(C1)**	**OSC2(C2)**
XT	455 kHz 2.0 MHz 4.0 MHz	68-100 pF 15-68 pF 15-68 pF	68-100 pF 15-68 pF 15-68 pF
HS	8.0 MHz 16.0 MHz	10-68 pF 10-22 pF	10-68 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
XT	100 kHz 2 MHz 4 MHz	68-150 pF 15-30 pF 15-30 pF	150-200 pF 15-30 pF 15-30 pF
HS	8 MHz 10 MHz 20 MHz	15-30 pF 15-30 pF 15-30 pF	15-30 pF 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 overdriving crystals with low drive level specification. Since each crystal has its own characteristics, the user should consult the crystal manufacturer for appropriate values of external components.

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 "Electrical Specifications"**. Figure 9-2 shows how an external clock circuit should be configured.

9.2.4 RC OSCILLATOR

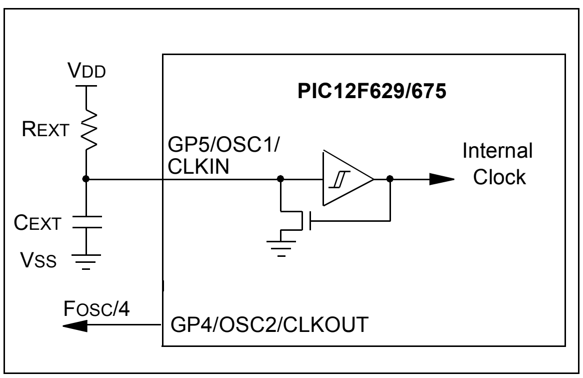
For applications where precise timing is not a requirement, the RC oscillator option is available. The operation and functionality of the RC oscillator is dependent upon a number of variables. The RC oscillator frequency is a function of:

- Supply voltage
- Resistor (REXT) and capacitor (CEXT) 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 CEXT 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.

Two options are available for this Oscillator mode which allow GP4 to be used as a general purpose I/O or to output FOSC/4.

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 "Electrical Specifications"**, for information on variation over voltage and temperature.

Two options are available for this Oscillator mode which allow GP4 to be used as a general purpose I/O or to output FOSC/4.

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. For best operation, decouple (with capacitance) VDD and VSS as close to the device as possible.

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 as specified in the PIC12F629/675 Programming specification. Microchip Development Tools maintain all Calibration bits to factory settings.

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 (FOSC/4) is output on the GP4/OSC2/CLKOUT pin. FOSC/4 can be used for test purposes or to synchronize other logic.

9.3 Reset

The PIC12F629/675 differentiates between various kinds of Reset:

a) Power-on Reset (POR)
b) WDT Reset during normal operation
c) WDT Reset during Sleep
d) $\overline{\text{MCLR}}$ Reset during normal operation
e) $\overline{\text{MCLR}}$ Reset during Sleep
f) 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:

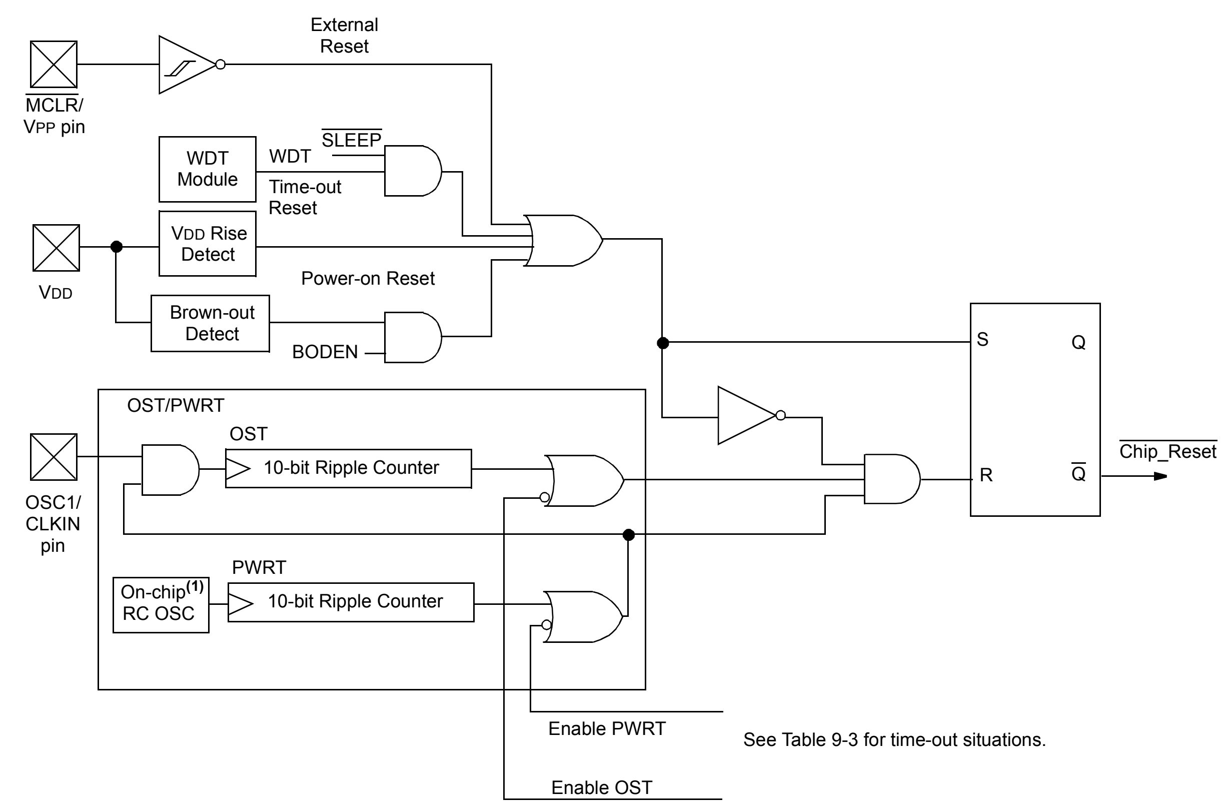
- Power-on Reset
- $\overline{\text{MCLR}}$ Reset
- WDT Reset
- WDT Reset during Sleep
- Brown-out Detect (BOD) Reset

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



Note 1: This is a separate oscillator from the INTOSC/EC oscillator.

9.3.1 MCLR

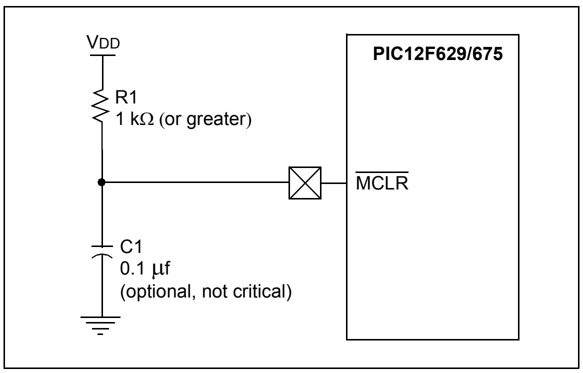
PIC12F629/675 devices have a noise filter in the MCLR Reset path. The filter will detect and ignore small pulses.

It should be noted that a WDT Reset does not drive MCLR pin low.

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

An internal MCLR option is enabled by setting the MCLRE bit in the Configuration Word. When enabled, MCLR is internally tied to VDD. No internal pull-up option is available for the MCLR pin.

FIGURE 9-5: RECOMMENDED 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 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 (see **Section 12.0 "Electrical Specifications"**). If the BOD is enabled, the maximum rise time specification does not apply. The BOD circuitry will keep the device in Reset until VDD reaches VBOD (see **Section 9.3.5 "Brown-Out Detect (BOD)"**).

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 Detect. 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, PWRTE 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 Detect 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 (**Section 12.0 "Electrical Specifications"**).

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)

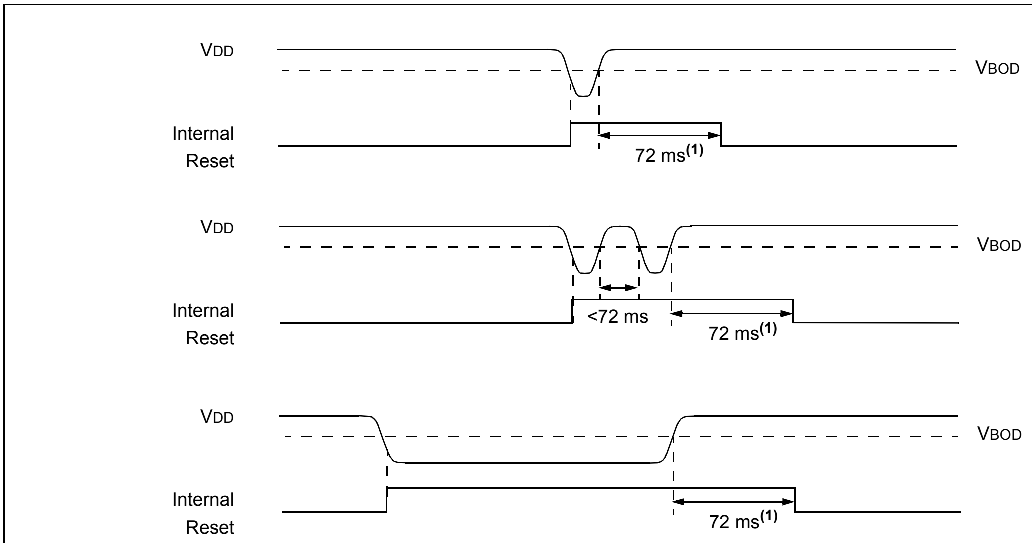
The PIC12F629/675 members have on-chip Brown-out Detect circuitry. A Configuration bit, BODEN, can disable (if clear/programmed) or enable (if set) the Brown-out Detect circuitry. If V_{DD} falls below V_{BOD} for greater than parameter (T_{BOD}) in Table 12-4 (see **Section 12.0 "Electrical Specifications"**), the Brown-out situation will reset the device. This will occur regardless of V_{DD} slew-rate. A Reset is not guaranteed to occur if V_{DD} falls below V_{BOD} for less than parameter (T_{BOD}).

On any Reset (Power-on, Brown-out, Watchdog, etc.), the chip will remain in Reset until V_{DD} rises above BV_{DD} (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.

Note:	A Brown-out Detect does not enable the Power-up Timer if the $\overline{\text{PWRTE}}$ bit in the Configuration Word is set.

If V_{DD} drops below BV_{DD} while the Power-up Timer is running, the chip will go back into a Brown-out Detect and the Power-up Timer will be re-initialized. Once V_{DD} rises above BV_{DD}, the Power-up Timer will execute a 72 ms Reset.

FIGURE 9-6: BROWN-OUT SITUATIONS



Note 1: 72 ms delay only if $\overline{\text{PWRTE}}$ bit is programmed to '0'.

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.

Bit 0 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).

Bit 1 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., V_{DD} may have gone too low).

TABLE 9-3: TIME-OUT IN VARIOUS SITUATIONS

Oscillator Configuration	Power-up		Brown-out Detect		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, BOD	Value on all other Resets(1)
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

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0', q = value depends on condition.
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(1)	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	• $\overline{\text{MCLR}}$ Reset during normal operation • $\overline{\text{MCLR}}$ Reset during Sleep • WDT Reset • Brown-out Detect(1)	• Wake-up from Sleep through interrupt • Wake-up from Sleep through WDT Time-out
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(3)
STATUS	03h/83h	0001 1xxx	000q quuu(4)	uuuq quuu(4)
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 uuqq(2)
PIR1	0Ch	00-- 0--0	00-- 0--0	qq-- q--q(2,5)
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(1,6)	---- --uu
OSCCAL	90h	1000 00--	1000 00--	uuuu uu--
WPU	95h	--11 -111	--11 -111	uuuu uuuu
IOC	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.

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

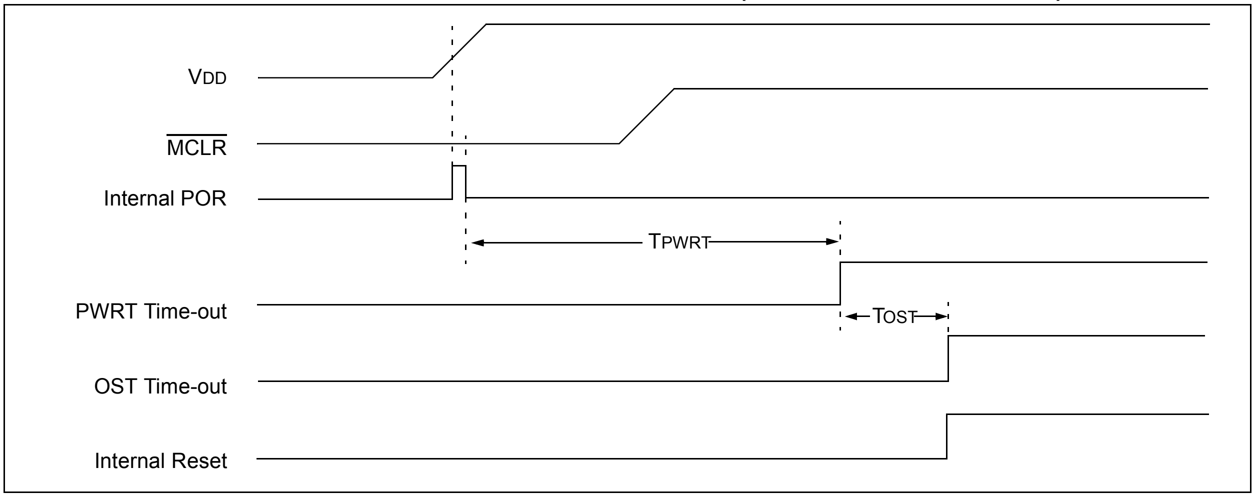


FIGURE 9-8: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ NOT TIED TO V_{DD}): CASE 2

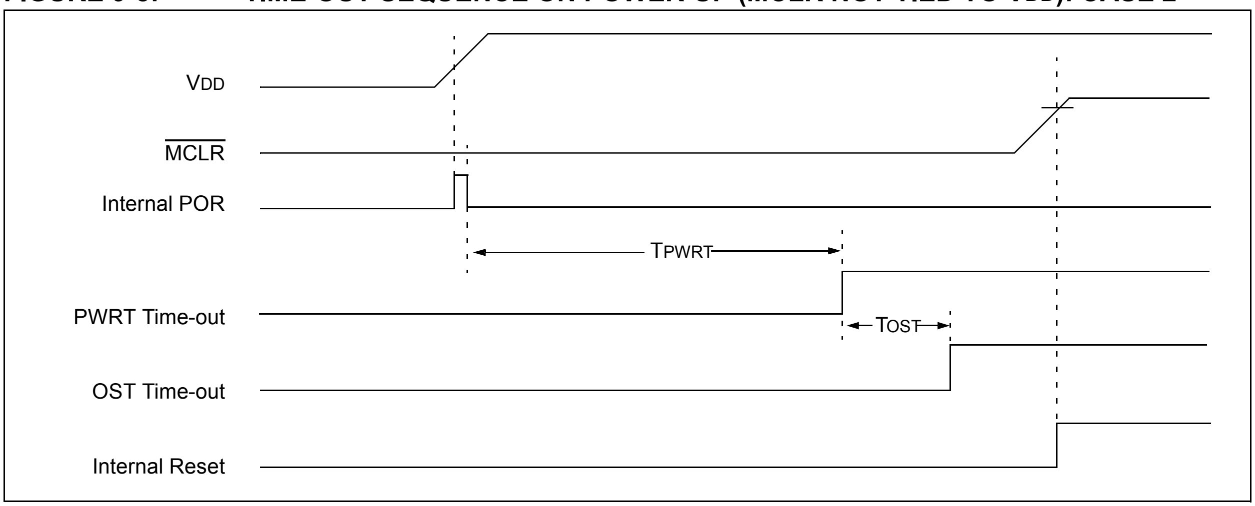
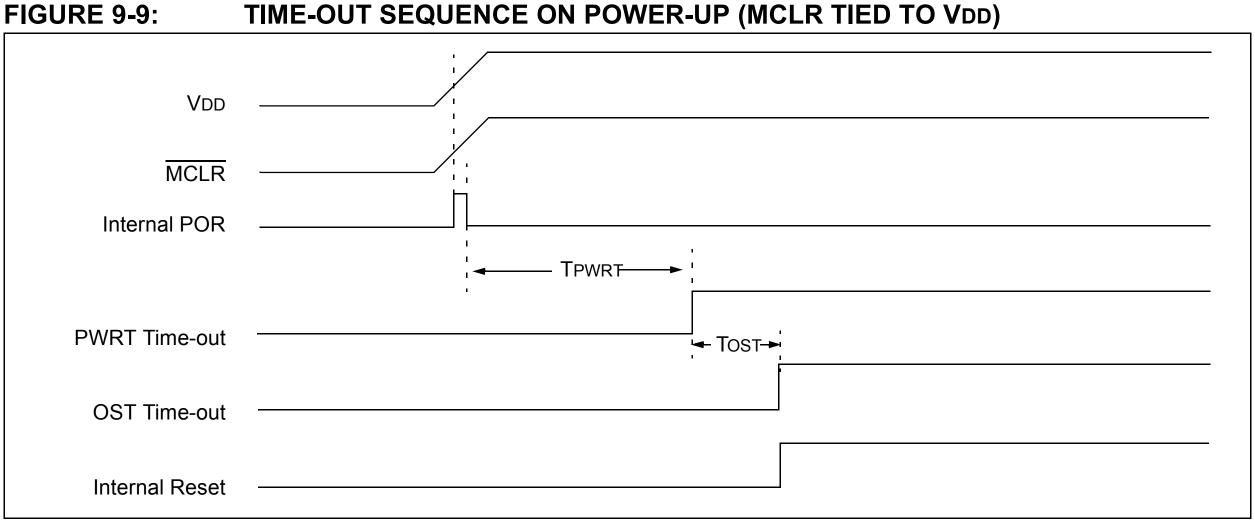


FIGURE 9-9: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ TIED TO V_{DD})



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. The INTCON register also has individual and Global Interrupt Enable (GIE) 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:

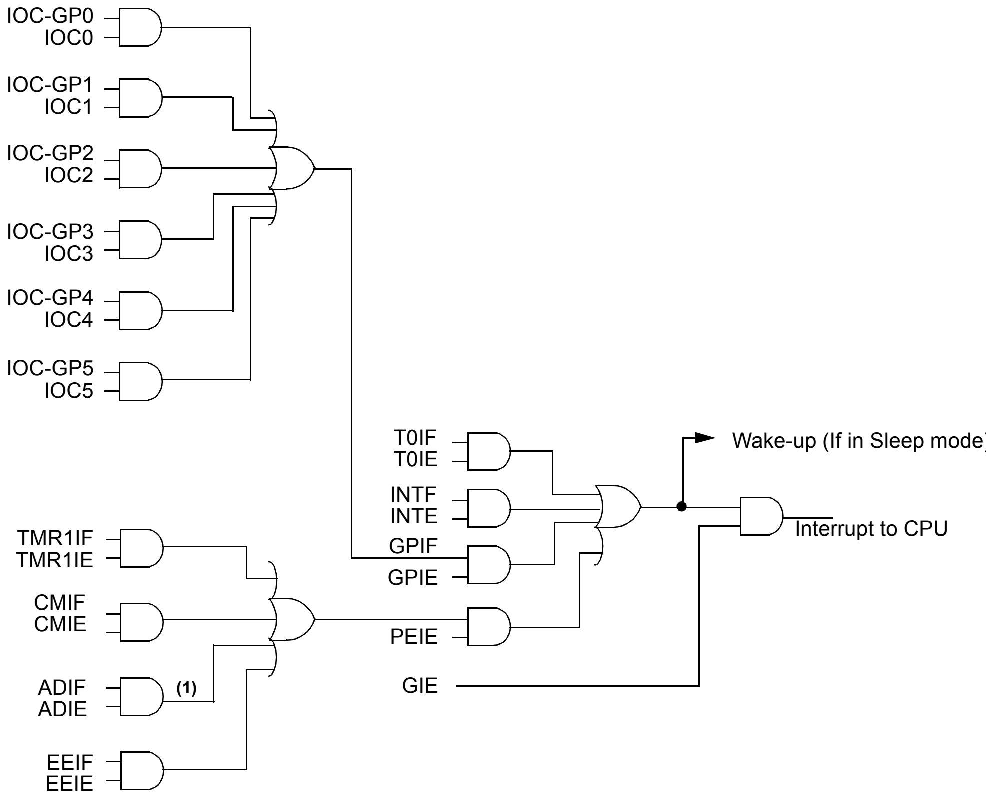
- The GIE is cleared to disable any further interrupt
- The return address is pushed onto the stack
- The PC is loaded with 0004h

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 9-11). 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 9-10: INTERRUPT LOGIC



Note 1: PIC12F675 only.

9.4.1 GP2/INT INTERRUPT

External interrupt on GP2/INT pin is edge-triggered; either rising if INTEDG bit (OPTION<6>) is set, of 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 "Power-Down Mode (Sleep)"** for details on Sleep and Figure 9-13 for timing of wake-up from Sleep through GP2/INT interrupt.

Note:	The ANSEL (9Fh) and CMCON (19h) registers must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read '0'. The ANSEL register is defined for the PIC12F675.

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 "Timer0 Module"**.

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 IOC 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

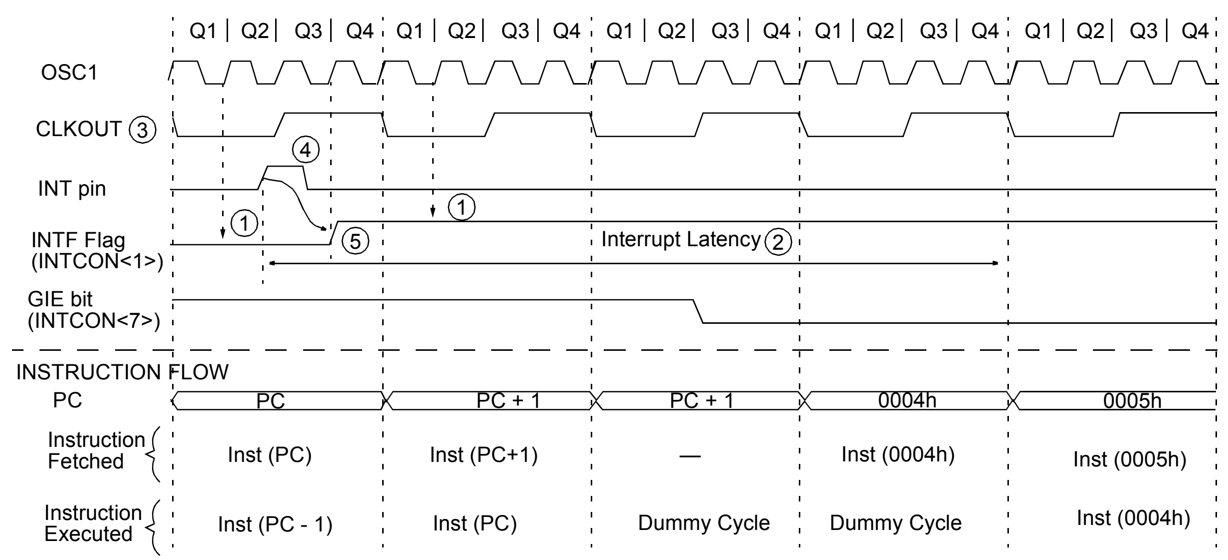
See **Section 6.9 "Comparator Interrupts"** for description of comparator interrupt.

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 "Analog-to-Digital Converter (A/D) Module (PIC12F675 only)"** for operation of the A/D converter interrupt.

FIGURE 9-11: INT PIN INTERRUPT TIMING



Note 1: INTF flag is sampled here (every Q1).
2: Asynchronous interrupt latency = 3-4 TCY. Synchronous latency = 3 TCY, where TCY = instruction cycle time. Latency is the same whether Inst (PC) is a single cycle or a 2-cycle instruction.
3: CLKOUT is available only in RC Oscillator mode.
4: For minimum width of INT pulse, refer to AC specs.
5: INTF is enabled to be set any time during the Q4-Q1 cycles.

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, BOD	Value on all other Resets
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
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 and INTOSC. 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 "Configuration Bits"**).

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, V_{DD} 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.

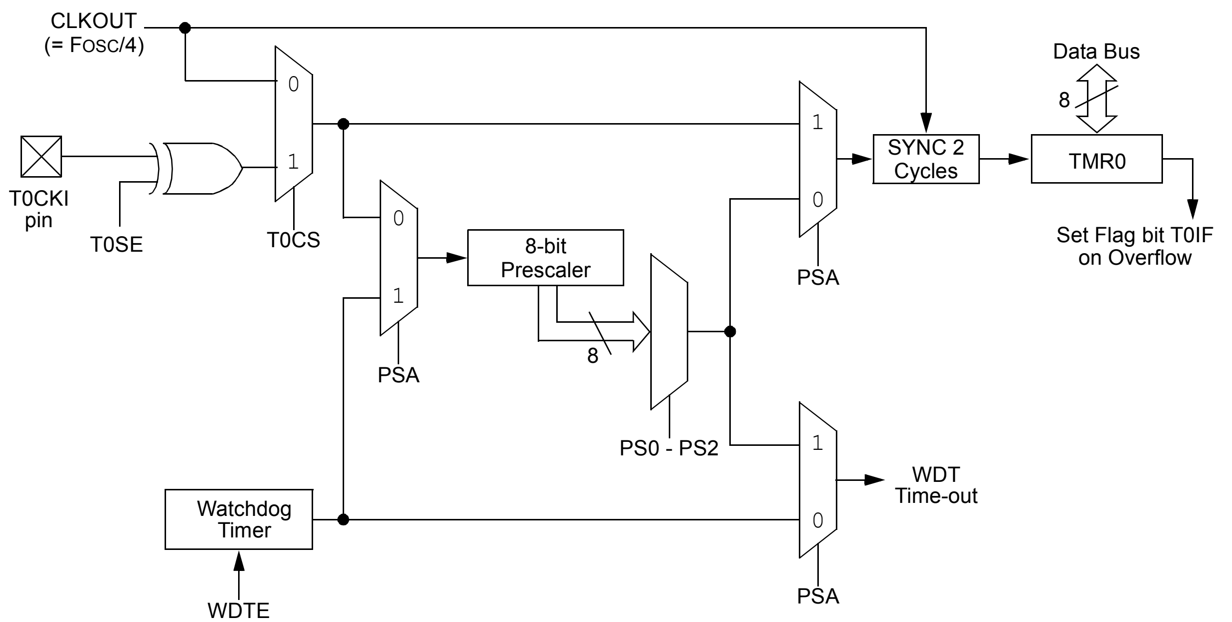
The CLRWDT and SLEEP instructions clear the WDT and the prescaler, if assigned to the WDT, and prevent it from timing out and generating a device Reset.

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., V_{DD} = Min., Temperature = Max., Max. WDT prescaler) it may take several seconds before a WDT Time-out occurs.

FIGURE 9-12: WATCHDOG TIMER BLOCK DIAGRAM



Note 1: T0SE, T0CS, PSA, PS0-PS2 are bits in the OPTION register.

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, BOD	Value on all other Resets
81h	OPTION_REG	$\overline{\text{GPPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
2007h	Config. bits	$\overline{\text{CP}}$	BODEN	MCLRE	$\overline{\text{PWRTE}}$	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
- $\overline{\text{PD}}$ bit in the STATUS register is cleared
- $\overline{\text{TO}}$ bit is set
- Oscillator driver is turned off
- I/O ports maintain the status they had before Sleep was executed (driving high, low, or high-impedance).

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

The $\overline{\text{MCLR}}$ 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 $\overline{\text{MCLR}}$ pin low.

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

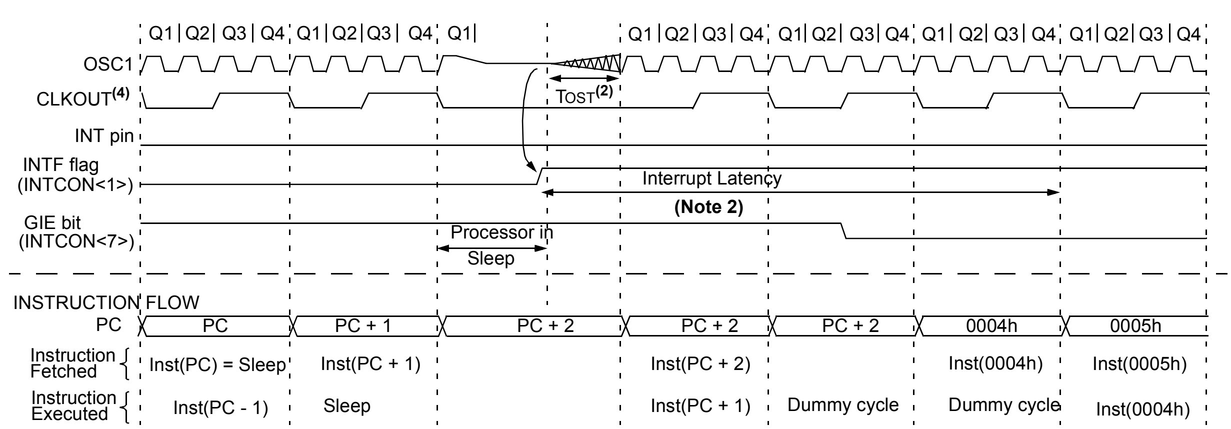
The first event will cause a device Reset. The two latter events are considered a continuation of program execution. The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits in the STATUS register can be used to determine the cause of device Reset. The $\overline{\text{PD}}$ bit, which is set on power-up,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

FIGURE 9-13: WAKE-UP FROM SLEEP THROUGH INTERRUPT



Note 1: XT, HS or LP Oscillator mode assumed.
2: T_{OST} = 1024T_{OSC} (drawing not to scale). Approximately 1 μs delay will be there for RC Osc mode. See Section 12 for wake-up from Sleep delay in INTOSC mode.
3: GIE = `1` assumed. In this case after wake-up, the processor jumps to the interrupt routine. If GIE = `0`, execution will continue in-line.
4: CLKOUT is not available in XT, HS, LP or EC Osc modes, but shown here for timing reference.

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 INTOSC 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 7 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

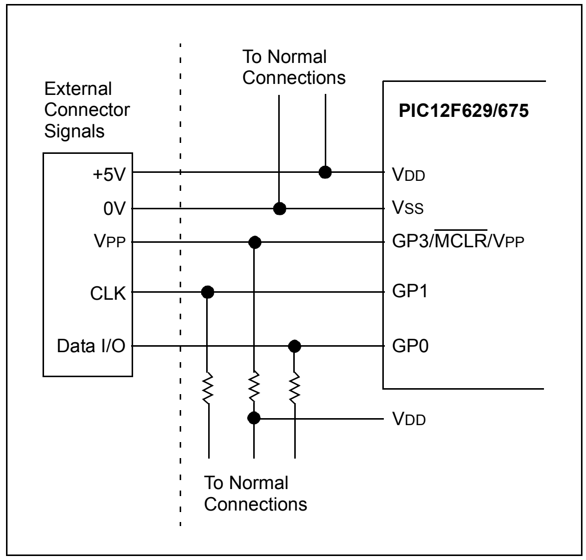
This allows customers to manufacture boards with unprogrammed devices, and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

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

After Reset, to place the device into Programming/Verify mode, the 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 9-14.

FIGURE 9-14: TYPICAL IN-CIRCUIT SERIAL PROGRAMMING CONNECTION



9.11 In-Circuit Debugger

Since in-circuit debugging requires the loss of clock, data and $\overline{\text{MCLR}}$ pins, MPLAB® ICD 2 development with an 8-pin device is not practical. A special 14-pin PIC12F675-ICD device is used with MPLAB ICD 2 to provide separate clock, data and $\overline{\text{MCLR}}$ pins and frees all normally available pins to the user.

This special ICD device is mounted on the top of the header and its signals are routed to the MPLAB ICD 2 connector. On the bottom of the header is an 8-pin socket that plugs into the user's target via the 8-pin stand-off connector.

When the $\overline{\text{ICD}}$ pin on the PIC12F675-ICD device is held low, the In-Circuit Debugger functionality is enabled. This function allows simple debugging functions when used with MPLAB ICD 2. When the microcontroller has this feature enabled, some of the resources are not available for general use. Table 9-10 shows which features are consumed by the background debugger:

TABLE 9-10: DEBUGGER RESOURCES

I/O pins	ICDCLK, ICDDATA
Stack	1 level
Program Memory	Address 0h must be `NOP` 300h-3FEh

For more information, see 8-Pin MPLAB ICD 2 Header Information Sheet (DS51292) available on Microchip's web site (www.microchip.com).

NOTES:

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 PIC12F629/675 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 PIC® 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 `TRISIO` 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

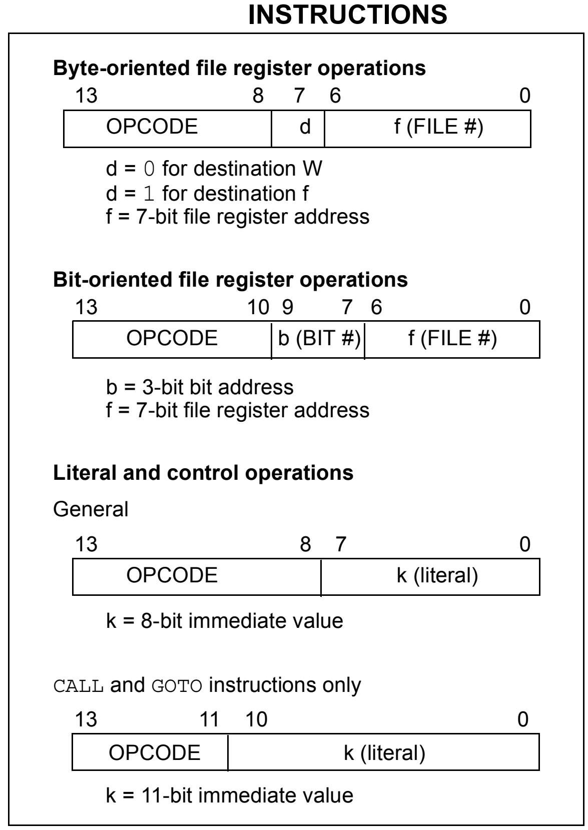


TABLE 10-2: PIC12F629/675 INSTRUCTION SET

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb			14-Bit Opcode LSb	Status Affected	Notes
BYTE-ORIENTED FILE REGISTER OPERATIONS									
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	lfff	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	lfff	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**
BIT-ORIENTED FILE REGISTER OPERATIONS									
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**
LITERAL AND CONTROL OPERATIONS									
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	$\overline{TO}$,$\overline{PD}$	
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	$\overline{TO}$,$\overline{PD}$	
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 PIC® Mid-Range MCU Family Reference Manual (DS33023).

10.2 Instruction Descriptions

ADDLW Add Literal and W

Syntax:	[*label*] ADDLW k
Operands:	$0 \le k \le 255$
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:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	$(W) + (f) \rightarrow$ (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:	$0 \le k \le 255$
Operation:	(W) .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:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) .AND. (f) $\rightarrow$ (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:	$0 \le f \le 127$ $0 \le b \le 7$
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:	$0 \le f \le 127$ $0 \le b \le 7$
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:	$0 \le f \le 127$ $0 \le b < 7$
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 $2T_{CY}$ instruction.

BTFSC	Bit Test, Skip if Clear
Syntax:	[*label*] BTFSC f,b
Operands:	$0 \le f \le 127$ $0 \le b \le 7$
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.

CALL	Call Subroutine
Syntax:	[*label*] CALL k
Operands:	$0 \le k \le 2047$
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:	$0 \le f \le 127$
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:	$0 \le f \le 127$ $d \in [0,1]$
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:	$0 \le f \le 127$ $d \in [0,1]$
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 → (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.

GOTO	Unconditional Branch
Syntax:	[*label*] GOTO k
Operands:	$0 \leq k \leq 2047$
Operation:	k → PC<10:0> PCLATH<4:3> → PC<12:11>
Status Affected:	None
Description:	`GOTO` is an unconditional branch. The eleven-bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. `GOTO` is a two-cycle instruction.

INCF	Increment f
Syntax:	[*label*] INCF f,d
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	(f) + 1 → (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'.

INCFSZ	Increment f, Skip if 0
Syntax:	[*label*] INCFSZ f,d
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	(f) + 1 → (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.

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

IORWF	Inclusive OR W with f
Syntax:	[*label*] IORWF f,d
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	(W) .OR. (f) → (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'.

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

MOVLW	Move literal to W
Syntax:	[*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.
Words:	1
Cycles:	1
Example:	`MOVLW 0x5A` After Instruction W = 0x5A

MOVWF	Move W to f
Syntax:	[*label*] MOVWF f
Operands:	$0 \leq f \leq 127$
Operation:	(W) $\rightarrow$ (f)
Status Affected:	None
Description:	Move data from W register to register ‘f’.
Words:	1
Cycles:	1
Example:	`MOVWF OPTION` Before Instruction OPTION = 0xFF W = 0x4F After Instruction OPTION = 0x4F W = 0x4F

NOP	No Operation
Syntax:	[*label*] NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Description:	No operation.
Words:	1
Cycles:	1
Example:	`NOP`

RETFIE	Return from Interrupt
Syntax:	[*label*] RETFIE
Operands:	None
Operation:	TOS → PC, 1 → GIE
Status Affected:	None
Description:	Return from Interrupt. Stack is POPed and Top-of-Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two-cycle instruction.
Words:	1
Cycles:	2
Example:	`RETFIE` After Interrupt PC = TOS GIE = 1

RETLW	Return with literal in W
Syntax:	[*label*] RETLW k
Operands:	$0 \leq k \leq 255$
Operation:	k → (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.
Words:	1
Cycles:	2
Example:	

```
        CALL TABLE;W contains
        table
            ;offset value
TABLE   •   ;W now has table value
        •
        •
        ADDWF PCL ;W = offset
        RETLW k1  ;Begin table
        RETLW k2  ;
        •
        •
        •
        RETLW kn  ; End of table
```

Before Instruction
W = 0x07
After Instruction
W = value of k8

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.

RLF — Rotate Left f through Carry

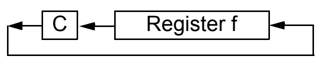
Syntax: [*label*] RLF f,d

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'.



RRF — Rotate Right f through Carry

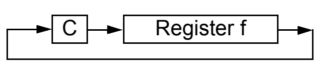
Syntax: [*label*] RRF f,d

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'.



SLEEP

Syntax: [*label*] SLEEP

Operands: None

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

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.

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.

SUBWF — Subtract W from f

Syntax: [*label*] SUBWF f,d

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

Operation: (f) - (W) $\rightarrow$ (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'.

SWAPF — Swap Nibbles in f

Syntax: [*label*] SWAPF f,d

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

Operation: (f<3:0>) $\rightarrow$ (destination<7:4>),
(f<7:4>) $\rightarrow$ (destination<3:0>)

Status Affected: None

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

XORLW	Exclusive OR Literal with W
Syntax:	[*label*] XORLW k
Operands:	$0 \le k \le 255$
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.

XORWF	Exclusive OR W with f
Syntax:	[*label*] XORWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
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'.

NOTES:

11.0 DEVELOPMENT SUPPORT

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

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

11.1 MPLAB Integrated Development Environment Software

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

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

The MPLAB IDE allows you to:

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

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

11.2 MPLAB C Compilers for Various Device Families

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

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

11.3 HI-TECH C for Various Device Families

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

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

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

11.4 MPASM Assembler

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

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code and COFF files for debugging.

The MPASM Assembler features include:

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

11.5 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler and the MPLAB C18 C Compiler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

11.6 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC devices. MPLAB C Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command line interface
- Rich directive set
- Flexible macro language
- MPLAB IDE compatibility

11.7 MPLAB SIM Software Simulator

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

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

11.8 MPLAB REAL ICE In-Circuit Emulator System

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

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

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

11.9 MPLAB ICD 3 In-Circuit Debugger System

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

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

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

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

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

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

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

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

11.12 MPLAB PM3 Device Programmer

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

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

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

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

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

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

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

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

12.0 ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings†

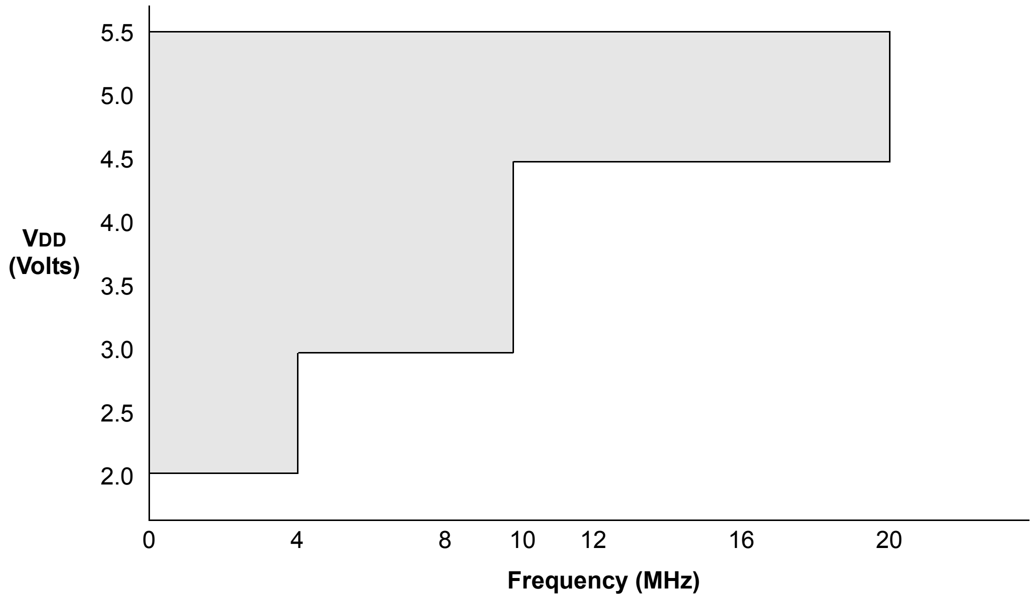
Ambient temperature under bias	-40 to +125°C
Storage temperature	-65°C to +150°C
Voltage on V_{DD} with respect to V_{SS}	-0.3 to +6.5V
Voltage on $\overline{MCLR}$ with respect to Vss	-0.3 to +13.5V
Voltage on all other pins with respect to V_{SS}	-0.3V to (V_{DD} + 0.3V)
Total power dissipation(1)	800 mW
Maximum current out of V_{SS} pin	300 mA
Maximum current into V_{DD} pin	250 mA
Input clamp current, I_{IK} (V_I < 0 or V_I > V_{DD})	± 20 mA
Output clamp current, I_{OK} (Vo < 0 or Vo >V_{DD})	± 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.

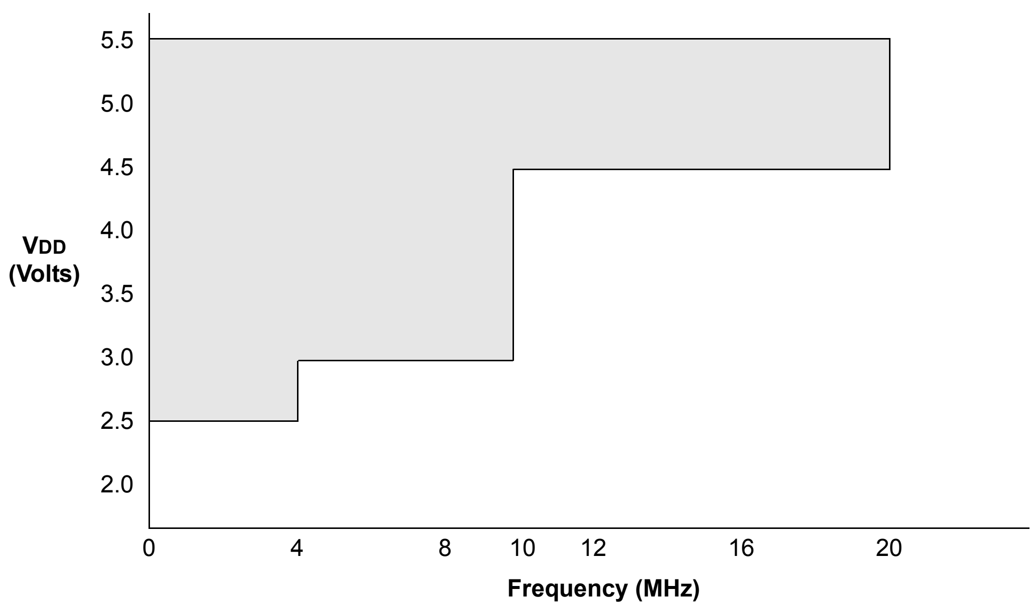
Note: Voltage spikes below V_{SS} at the $\overline{MCLR}$ pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100 Ω should be used when applying a "low" level to the $\overline{MCLR}$ pin, rather than pulling this pin directly to V_{SS}.

FIGURE 12-1: PIC12F629/675 WITH A/D DISABLED VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C



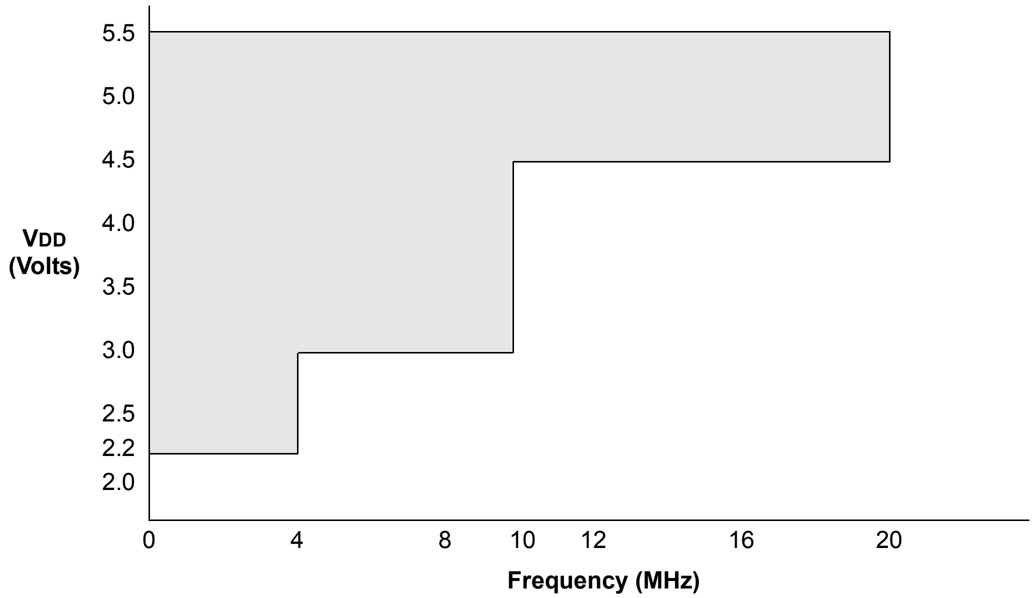
Note 1: The shaded region indicates the permissible combinations of voltage and frequency.

FIGURE 12-2: PIC12F675 WITH A/D ENABLED VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C



Note 1: The shaded region indicates the permissible combinations of voltage and frequency.

FIGURE 12-3: PIC12F675 WITH A/D ENABLED VOLTAGE-FREQUENCY GRAPH, 0°C ≤ T_A ≤ +125°C



Note 1: The shaded region indicates the permissible combinations of voltage and frequency.

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

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**
	VDD	**Supply Voltage**					FOSC < = 4 MHz:
D001			2.0	—	5.5	V	PIC12F629/675 with A/D off
D001A			2.2	—	5.5	V	PIC12F675 with A/D on, 0°C to +125°C
D001B			2.5	—	5.5	V	PIC12F675 with A/D on, -40°C to +125°C
D001C			3.0	—	5.5	V	4 MHz < FOSC < = 10 MHz
D001D			4.5	—	5.5	V	
D002	VDR	**RAM Data Retention Voltage**(1)	1.5*	—	—	V	Device in Sleep mode
D003	VPOR	**VDD Start Voltage** to ensure internal Power-on Reset signal	—	VSS	—	V	See section on Power-on Reset for details
D004	SVDD	**VDD Rise Rate** to ensure internal Power-on Reset signal	0.05*	—	—	V/ms	See section on Power-on Reset for details
D005	VBOD		—	2.1	—	V	

* 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.

12.2 DC Characteristics: PIC12F629/675-I (Industrial)

		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial					
Param No.	**Device Characteristics**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**	
						VDD	**Note**
D010	**Supply Current (IDD)**	—	9	16	μA	2.0	FOSC = 32 kHz LP Oscillator Mode
		—	18	28	μA	3.0	
		—	35	54	μA	5.0	
D011		—	110	150	μA	2.0	FOSC = 1 MHz XT Oscillator Mode
		—	190	280	μA	3.0	
		—	330	450	μA	5.0	
D012		—	220	280	μA	2.0	FOSC = 4 MHz XT Oscillator Mode
		—	370	650	μA	3.0	
		—	0.6	1.4	mA	5.0	
D013		—	70	110	μA	2.0	FOSC = 1 MHz EC Oscillator Mode
		—	140	250	μA	3.0	
		—	260	390	μA	5.0	
D014		—	180	250	μA	2.0	FOSC = 4 MHz EC Oscillator Mode
		—	320	470	μA	3.0	
		—	580	850	μA	5.0	
D015		—	340	450	μA	2.0	FOSC = 4 MHz INTOSC Mode
		—	500	700	μA	3.0	
		—	0.8	1.1	mA	5.0	
D016		—	180	250	μA	2.0	FOSC = 4 MHz EXTRC Mode
		—	320	450	μA	3.0	
		—	580	800	μA	5.0	
D017		—	2.1	2.95	mA	4.5	FOSC = 20 MHz HS Oscillator Mode
		—	2.4	3.0	mA	5.0	

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

Note 1: The 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; $\overline{\text{MCLR}}$ = VDD; WDT disabled.

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.

12.3 DC Characteristics: PIC12F629/675-I (Industrial)

		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial					
Param No.	**Device Characteristics**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**	
						VDD	**Note**
D020	Power-down Base Current (IPD)	—	0.99	700	nA	2.0	WDT, BOD, Comparators, VREF, and T1OSC disabled
		—	1.2	770	nA	3.0	
		—	2.9	995	nA	5.0	
D021		—	0.3	1.5	μA	2.0	WDT Current(1)
		—	1.8	3.5	μA	3.0	
		—	8.4	17	μA	5.0	
D022		—	58	70	μA	3.0	BOD Current(1)
		—	109	130	μA	5.0	
D023		—	3.3	6.5	μA	2.0	Comparator Current(1)
		—	6.1	8.5	μA	3.0	
		—	11.5	16	μA	5.0	
D024		—	58	70	μA	2.0	CVREF Current(1)
		—	85	100	μA	3.0	
		—	138	160	μA	5.0	
D025		—	4.0	6.5	μA	2.0	T1 OSC Current(1)
		—	4.6	7.0	μA	3.0	
		—	6.0	10.5	μA	5.0	
D026		—	1.2	775	nA	3.0	A/D Current(1)
		—	0.0022	1.0	μA	5.0	

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

Note 1: The peripheral current is the sum of the base IDD or IPD and the additional current consumed when this peripheral is enabled. The peripheral Δ current can be determined by subtracting the base IDD or IPD current from this limit. Max values should be used when calculating total current consumption.

2: 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 high-impedance state and tied to VDD.

12.4 DC Characteristics: PIC12F629/675-E (Extended)

		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C for extended					
Param No.	**Device Characteristics**	**Min**	**Typ†**	**Max**	**Units**	**Conditions: VDD**	**Conditions: Note**
D010E	**Supply Current (IDD)**	—	9	16	μA	2.0	FOSC = 32 kHz LP Oscillator Mode
		—	18	28	μA	3.0	
		—	35	54	μA	5.0	
D011E		—	110	150	μA	2.0	FOSC = 1 MHz XT Oscillator Mode
		—	190	280	μA	3.0	
		—	330	450	μA	5.0	
D012E		—	220	280	μA	2.0	FOSC = 4 MHz XT Oscillator Mode
		—	370	650	μA	3.0	
		—	0.6	1.4	mA	5.0	
D013E		—	70	110	μA	2.0	FOSC = 1 MHz EC Oscillator Mode
		—	140	250	μA	3.0	
		—	260	390	μA	5.0	
D014E		—	180	250	μA	2.0	FOSC = 4 MHz EC Oscillator Mode
		—	320	470	μA	3.0	
		—	580	850	μA	5.0	
D015E		—	340	450	μA	2.0	FOSC = 4 MHz INTOSC Mode
		—	500	780	μA	3.0	
		—	0.8	1.1	mA	5.0	
D016E		—	180	250	μA	2.0	FOSC = 4 MHz EXTRC Mode
		—	320	450	μA	3.0	
		—	580	800	μA	5.0	
D017E		—	2.1	2.95	mA	4.5	FOSC = 20 MHz HS Oscillator Mode
		—	2.4	3.0	mA	5.0	

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

Note 1: The 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; $\overline{\text{MCLR}}$ = VDD; WDT disabled.

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.

12.5 DC Characteristics: PIC12F629/675-E (Extended)

		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ T_A ≤ +125°C for extended					
Param No.	Device Characteristics	Min	Typ†	Max	Units	Conditions: V_{DD}	Conditions: Note
D020E	Power-down Base Current (I_{PD})	—	0.00099	3.5	μA	2.0	WDT, BOD, Comparators, V_{REF}, and T1OSC disabled
		—	0.0012	4.0	μA	3.0	
		—	0.0029	8.0	μA	5.0	
D021E		—	0.3	6.0	μA	2.0	WDT Current(1)
		—	1.8	9.0	μA	3.0	
		—	8.4	20	μA	5.0	
D022E		—	58	70	μA	3.0	BOD Current(1)
		—	109	130	μA	5.0	
D023E		—	3.3	10	μA	2.0	Comparator Current(1)
		—	6.1	13	μA	3.0	
		—	11.5	24	μA	5.0	
D024E		—	58	70	μA	2.0	CV_{REF} Current(1)
		—	85	100	μA	3.0	
		—	138	165	μA	5.0	
D025E		—	4.0	10	μA	2.0	T1 Osc Current(1)
		—	4.6	12	μA	3.0	
		—	6.0	20	μA	5.0	
D026E		—	0.0012	6.0	μA	3.0	A/D Current(1)
		—	0.0022	8.5	μA	5.0	

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

Note 1: The peripheral current is the sum of the base I_{DD} or I_{PD} and the additional current consumed when this peripheral is enabled. The peripheral Δ current can be determined by subtracting the base I_{DD} or I_{PD} current from this limit. Max values should be used when calculating total current consumption.

2: 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 high-impedance state and tied to V_{DD}.

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

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**
		Input Low Voltage					
	VIL	I/O ports					
D030		with TTL buffer	VSS	—	0.8	V	4.5V ≤ VDD ≤ 5.5V
D030A			VSS	—	0.15 VDD	V	Otherwise
D031		with Schmitt Trigger buffer	VSS	—	0.2 VDD	V	Entire range
D032		$\overline{\text{MCLR}}$, OSC1 (RC mode)	VSS	—	0.2 VDD	V	
D033		OSC1 (XT and LP modes)	VSS	—	0.3	V	**(Note 1)**
D033A		OSC1 (HS mode)	VSS	—	0.3 VDD	V	**(Note 1)**
		Input High Voltage					
	VIH	I/O ports		—			
D040		with TTL buffer	2.0	—	VDD	V	4.5V ≤ VDD ≤ 5.5V
D040A			(0.25 VDD+0.8)	—	VDD	V	otherwise
D041		with Schmitt Trigger buffer	0.8 VDD	—	VDD		entire range
D042		$\overline{\text{MCLR}}$	0.8 VDD	—	VDD	V	
D043		OSC1 (XT and LP modes)	1.6	—	VDD	V	**(Note 1)**
D043A		OSC1 (HS mode)	0.7 VDD	—	VDD	V	**(Note 1)**
D043B		OSC1 (RC mode)	0.9 VDD	—	VDD	V	
D070	IPUR	**GPIO Weak Pull-up Current**	50*	250	400*	μA	VDD = 5.0V, VPIN = VSS
		Input Leakage Current[3]					
D060	IIL	I/O ports	—	± 0.1	± 1	μA	VSS ≤ VPIN ≤ VDD, Pin at high-impedance
D060A		Analog inputs	—	± 0.1	± 1	μA	VSS ≤ VPIN ≤ VDD
D060B		VREF	—	± 0.1	± 1	μA	VSS ≤ VPIN ≤ VDD
D061		$\overline{\text{MCLR}}$[2]	—	± 0.1	± 5	μA	VSS ≤ VPIN ≤ VDD
D063		OSC1	—	± 0.1	± 5	μA	VSS ≤ VPIN ≤ VDD, XT, HS and LP osc configuration
		Output Low Voltage					
D080	VOL	I/O ports	—	—	0.6	V	IOL = 8.5 mA, VDD = 4.5V (Ind.)
D083		OSC2/CLKOUT (RC mode)	—	—	0.6	V	IOL = 1.6 mA, VDD = 4.5V (Ind.) IOL = 1.2 mA, VDD = 4.5V (Ext.)
		Output High Voltage					
D090	VOH	I/O ports	VDD - 0.7	—	—	V	IOH = -3.0 mA, VDD = 4.5V (Ind.)
D092		OSC2/CLKOUT (RC mode)	VDD - 0.7	—	—	V	IOH = -1.3 mA, VDD = 4.5V (Ind.) IOH = -1.0 mA, VDD = 4.5V (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 $\overline{\text{MCLR}}$ 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.

12.7 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**
		Capacitive Loading Specs on Output Pins					
D100	COSC2	OSC2 pin	—	—	15*	pF	In XT, HS and LP modes when external clock is used to drive OSC1
D101	CIO	All I/O pins	—	—	50*	pF	
		Data EEPROM Memory					
D120	ED	Byte Endurance	100K	1M	—	E/W	-40°C ≤ TA ≤ +85°C
D120A	ED	Byte Endurance	10K	100K	—	E/W	+85°C ≤ TA ≤ +125°C
D121	VDRW	VDD for Read/Write	VMIN	—	5.5	V	Using EECON to read/write VMIN = Minimum operating voltage
D122	TDEW	Erase/Write cycle time	—	5	6	ms	
D123	TRETD	Characteristic Retention	40	—	—	Year	Provided no other specifications are violated
D124	TREF	Number of Total Erase/Write Cycles before Refresh(1)	1M	10M	—	E/W	-40°C ≤ TA ≤ +85°C
		Program Flash Memory					
D130	EP	Cell Endurance	10K	100K	—	E/W	-40°C ≤ TA ≤ +85°C
D130A	ED	Cell Endurance	1K	10K	—	E/W	+85°C ≤ TA ≤ +125°C
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	2.5	ms	
D134	TRETD	Characteristic Retention	40	—	—	Year	Provided no other specifications are violated

* 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 “Using the Data EEPROM”** for additional information.

12.8 TIMING PARAMETER SYMBOLOGY

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

1. TppS2ppS
2. TppS

T			
F	Frequency	T	Time

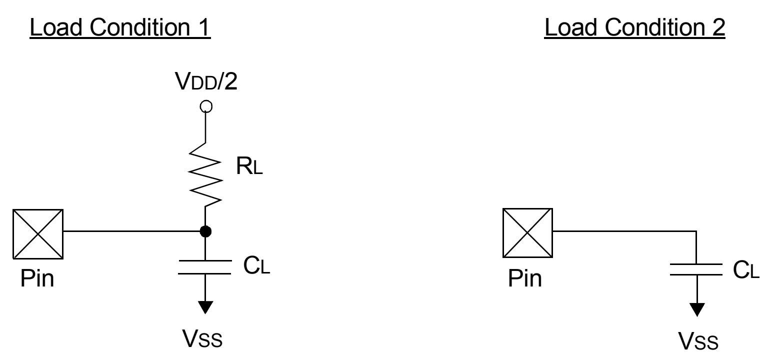
Lowercase letters (pp) and their meanings:

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

Uppercase letters and their meanings:

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

FIGURE 12-4: LOAD CONDITIONS



R_L = 464Ω

C_L = 50 pF for all pins

15 pF for OSC2 output

12.9 AC CHARACTERISTICS: PIC12F629/675 (INDUSTRIAL, EXTENDED)

FIGURE 12-5: EXTERNAL CLOCK TIMING

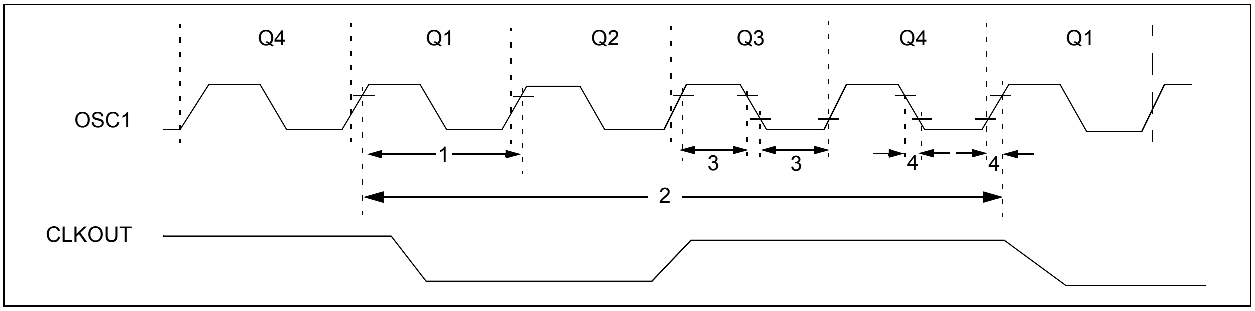


TABLE 12-1: EXTERNAL CLOCK TIMING REQUIREMENTS

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
	FOSC	External CLKIN Frequency(1)	DC	—	37	kHz	LP Osc mode
			DC	—	4	MHz	XT mode
			DC	—	20	MHz	HS mode
			DC	—	20	MHz	EC mode
		Oscillator Frequency(1)	5	—	37	kHz	LP Osc mode
			—	4	—	MHz	INTOSC mode
			DC	—	4	MHz	RC Osc mode
			0.1	—	4	MHz	XT Osc mode
			1	—	20	MHz	HS Osc mode
1	TOSC	External CLKIN Period(1)	27	—	∞	μs	LP Osc mode
			50	—	∞	ns	HS Osc mode
			50	—	∞	ns	EC Osc mode
			250	—	∞	ns	XT Osc mode
		Oscillator Period(1)	27		200	μs	LP Osc mode
			—	250	—	ns	INTOSC mode
			250	—	—	ns	RC Osc mode
			250	—	10,000	ns	XT Osc mode
			50	—	1,000	ns	HS Osc mode
2	TCY	Instruction Cycle Time(1)	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.

TABLE 12-2: PRECISION INTERNAL OSCILLATOR PARAMETERS

Param No.	Sym	Characteristic	Freq. Tolerance	Min	Typ†	Max	Units	Conditions
F10	F_{OSC}	Internal Calibrated INTOSC Frequency	±1	3.96	4.00	4.04	MHz	V_{DD} = 3.5V, 25°C
			±2	3.92	4.00	4.08	MHz	2.5V ≤ V_{DD} ≤ 5.5V 0°C ≤ T_A ≤ +85°C
			±5	3.80	4.00	4.20	MHz	2.0V ≤ V_{DD} ≤ 5.5V -40°C ≤ T_A ≤ +85°C (IND) -40°C ≤ T_A ≤ +125°C (EXT)
F14	T_{IOSCST}	Oscillator Wake-up from Sleep start-up time*	—	—	6	8	μs	V_{DD} = 2.0V, -40°C to +85°C
			—	—	4	6	μs	V_{DD} = 3.0V, -40°C to +85°C
			—	—	3	5	μs	V_{DD} = 5.0V, -40°C to +85°C

* 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.

FIGURE 12-6: CLKOUT AND I/O TIMING

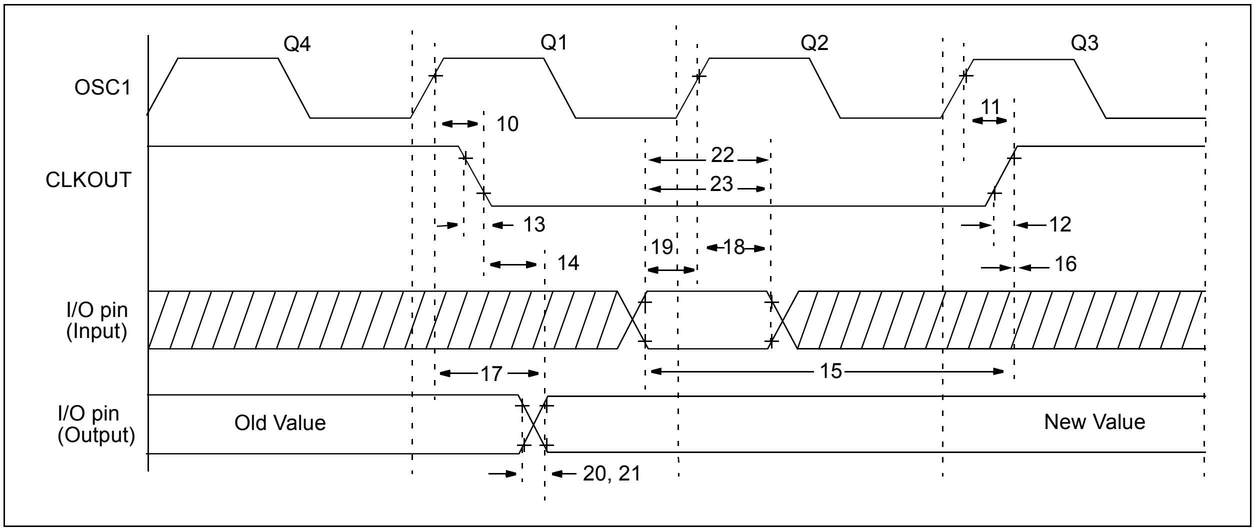


TABLE 12-3: CLKOUT AND I/O TIMING REQUIREMENTS

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
10	TosH2ckL	OSC1↑ to CLOUT↓	—	75	200	ns	**(Note 1)**
11	TosH2ckH	OSC1↑ to CLOUT↑	—	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	TCY	—	—	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.

FIGURE 12-7: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING

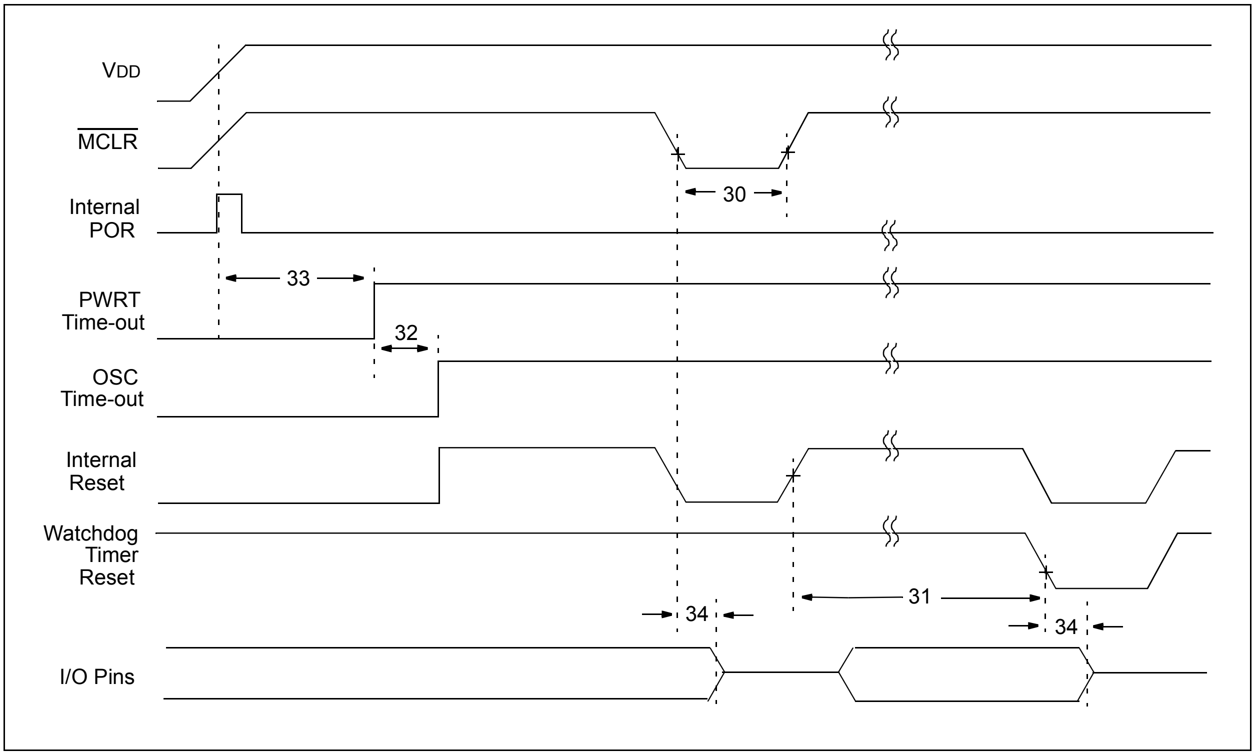
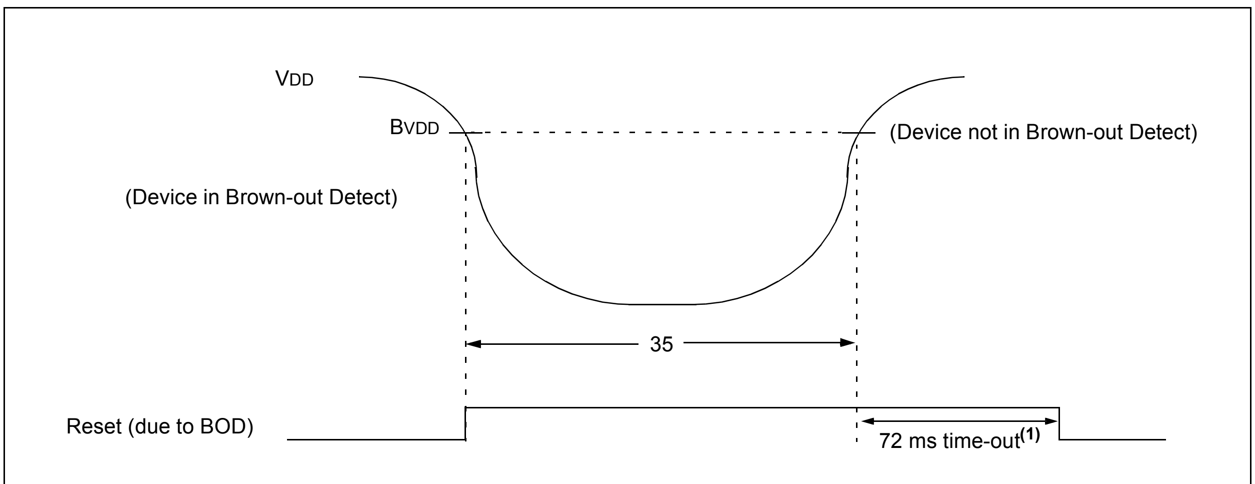


FIGURE 12-8: BROWN-OUT DETECT TIMING AND CHARACTERISTICS



Note 1: 72 ms delay only if $\overline{\text{PWRTE}}$ bit in Configuration Word is programmed to ‘0’.

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

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30	TMCL	$\overline{\text{MCLR}}$ 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)	10 10	17 17	25 30	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 High-impedance from $\overline{\text{MCLR}}$ Low or Watchdog Timer Reset	—	—	2.0	μs	
	BVDD	Brown-out Detect Voltage	2.025	—	2.175	V	
		Brown-out Hysteresis	TBD	—	—	—	
35	TBOD	Brown-out Detect Pulse Width	100*	—	—	μs	VDD ≤ BVDD (D005)

* These parameters are characterized but not tested.

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

FIGURE 12-9: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS

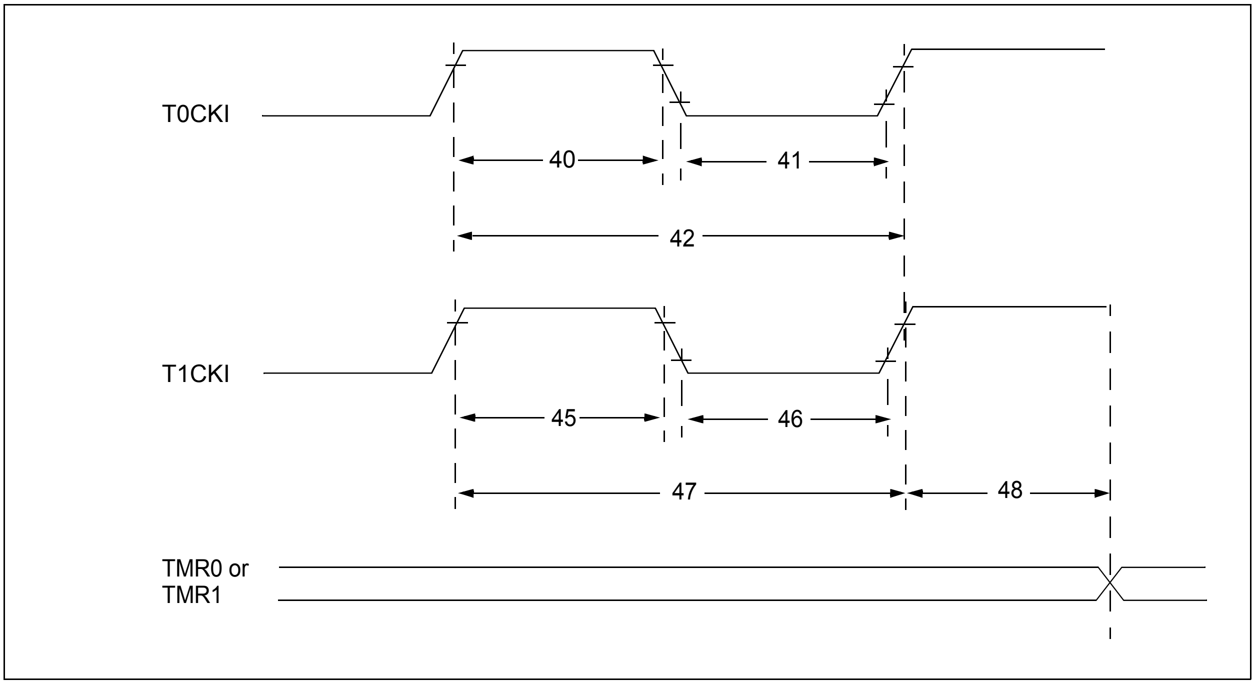


TABLE 12-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

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**
V_{OS}	Input Offset Voltage	—	± 5.0	± 10	mV	
V_{CM}	Input Common Mode Voltage	0	—	V_{DD} - 1.5	V	
C_{MRR}	Common Mode Rejection Ratio	+55*	—	—	db	
T_{RT}	Response Time(1)	—	150	400*	ns	
T_{MC2COV}	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 (V_{DD} - 1.5)/2 while the other input transitions from V_{SS} to V_{DD} - 1.5V.

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	— —	V_{DD}/24* V_{DD}/32	— —	LSb LSb	Low Range (VRR = 1) High Range (VRR = 0)
	Absolute Accuracy	— —	— —	± 1/2 ± 1/2*	LSb LSb	Low Range (VRR = 1) High Range (VRR = 0)
	Unit Resistor Value (R)	—	2k*	—	Ω	
	Settling Time(1)	—	—	10*	μs	

* These parameters are characterized but not tested.

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

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*	—	—	±1	LSb	VREF = 5.0V
A03	EIL	Integral Error	—	—	±1	LSb	VREF = 5.0V
A04	EDL	Differential Error	—	—	±1	LSb	No missing codes to 10 bits VREF = 5.0V
A05	EFS	Full Scale Range	2.2*	—	5.5*	V	
A06	EOFF	Offset Error	—	—	±1	LSb	VREF = 5.0V
A07	EGN	Gain Error	—	—	±1	LSb	VREF = 5.0V
A10	—	Monotonicity	—	guaranteed(3)	—	—	VSS ≤ VAIN ≤ VREF+
A20 A20A	VREF	Reference Voltage	2.0 2.5	—	— VDD + 0.3	V	 Absolute minimum to ensure 10-bit accuracy
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	—	—	10	kΩ	
A50	IREF	VREF Input Current(2)	10 —	— —	1000 10	μA μA	During VAIN acquisition. Based on differential of VHOLD to VAIN. During A/D conversion cycle.

* These parameters are characterized but not tested.

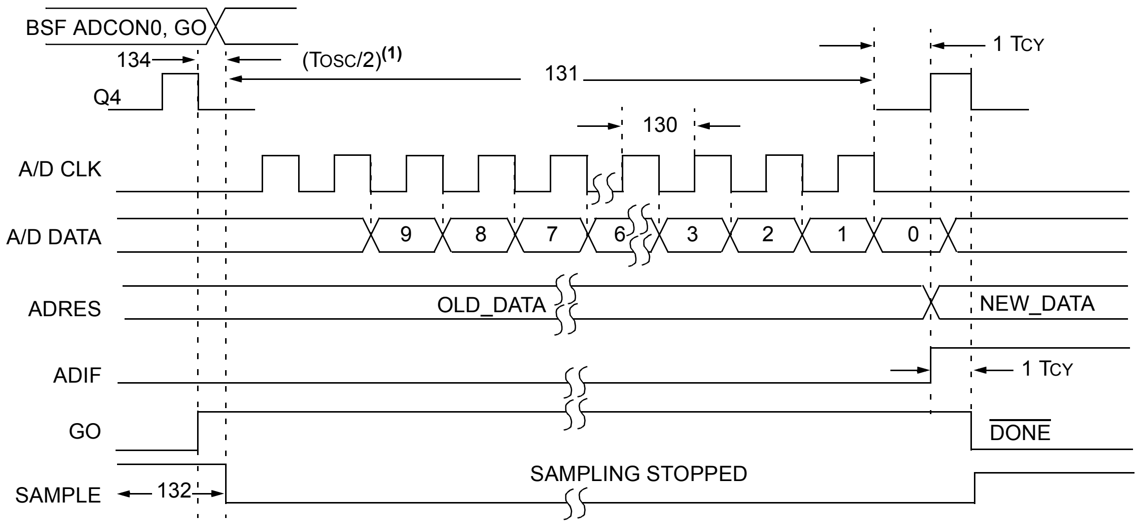
† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

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 12-10: PIC12F675 A/D CONVERSION TIMING (NORMAL MODE)



Note 1: If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the `SLEEP` instruction to be executed.

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	—	—	μs	TOSC based, VREF ≥ 3.0V
			3.0*	—	—	μs	TOSC based, VREF full range
130	TAD	A/D Internal RC Oscillator Period					ADCS<1:0> = 11 (RC mode)
			3.0*	6.0	9.0*	μs	At VDD = 2.5V
			2.0*	4.0	6.0*	μs	At VDD = 5.0V
131	TCNV	Conversion Time (not including Acquisition Time)(1)	—	11	—	TAD	Set GO bit to new data in A/D result register
132	TACQ	Acquisition Time	**(Note 2)**	11.5	—	μs	
			5*	—	—	μs	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).
134	TGO	Q4 to A/D Clock Start	—	TOSC/2	—	—	If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the `SLEEP` instruction to be executed.

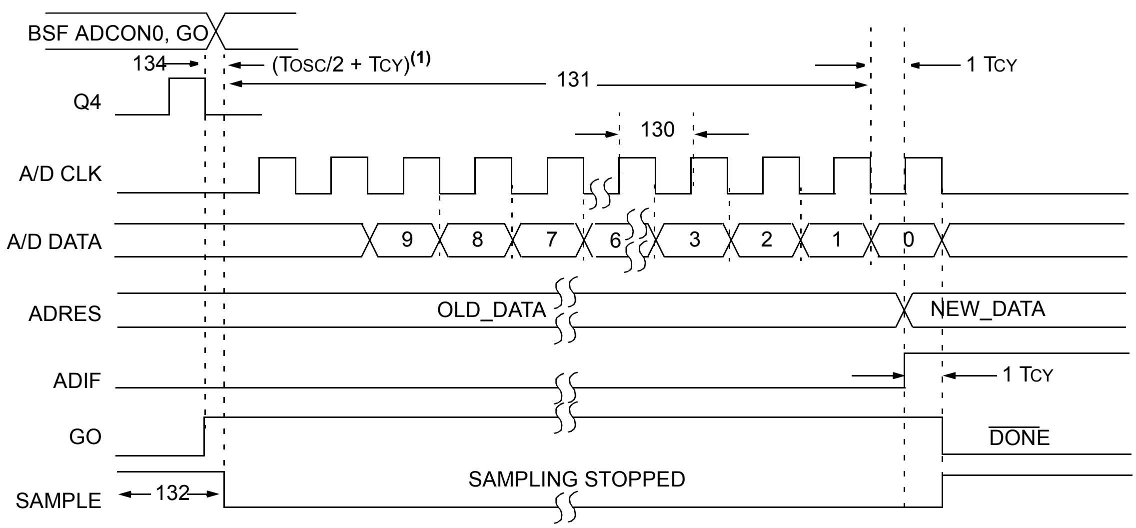
* These parameters are characterized but not tested.

† 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 TCY cycle.

2: See **Section 7.1 "A/D Configuration and Operation"** for minimum conditions.

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



Note 1: If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the `SLEEP` instruction to be executed.

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	—	—	μs	VREF ≥ 3.0V
			3.0*	—	—	μs	VREF full range
130	TAD	A/D Internal RC Oscillator Period					ADCS<1:0> = `11` (RC mode)
			3.0*	6.0	9.0*	μs	At VDD = 2.5V
			2.0*	4.0	6.0*	μs	At VDD = 5.0V
131	TCNV	Conversion Time (not including Acquisition Time)(1)	—	11	—	TAD	
132	TACQ	Acquisition Time	**(Note 2)**	11.5	—	μs	
			5*	—	—	μs	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).
134	TGO	Q4 to A/D Clock Start	—	TOSC/2 + TCY	—	—	If the A/D clock source is selected as RC, a time of TCY 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 TCY cycle.

2: See **Section 7.1 “A/D Configuration and Operation”** for minimum conditions.

NOTES:

13.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES

The graphs and tables provided in this section are for **design guidance** and are **not tested**.

In some graphs or tables, the data presented are **outside specified operating range** (i.e., outside specified VDD range). This is for **information only** and devices are ensured to operate properly only within the specified range.

The data presented in this section is a **statistical summary** of data collected on units from different lots over a period of time and matrix samples. “Typical” represents the mean of the distribution at 25°C. “Max” or “min” represents (mean + 3σ) or (mean - 3σ) respectively, where σ is standard deviation, over the whole temperature range.

FIGURE 13-1: TYPICAL IPD vs. VDD OVER TEMP (-40°C TO +25°C)

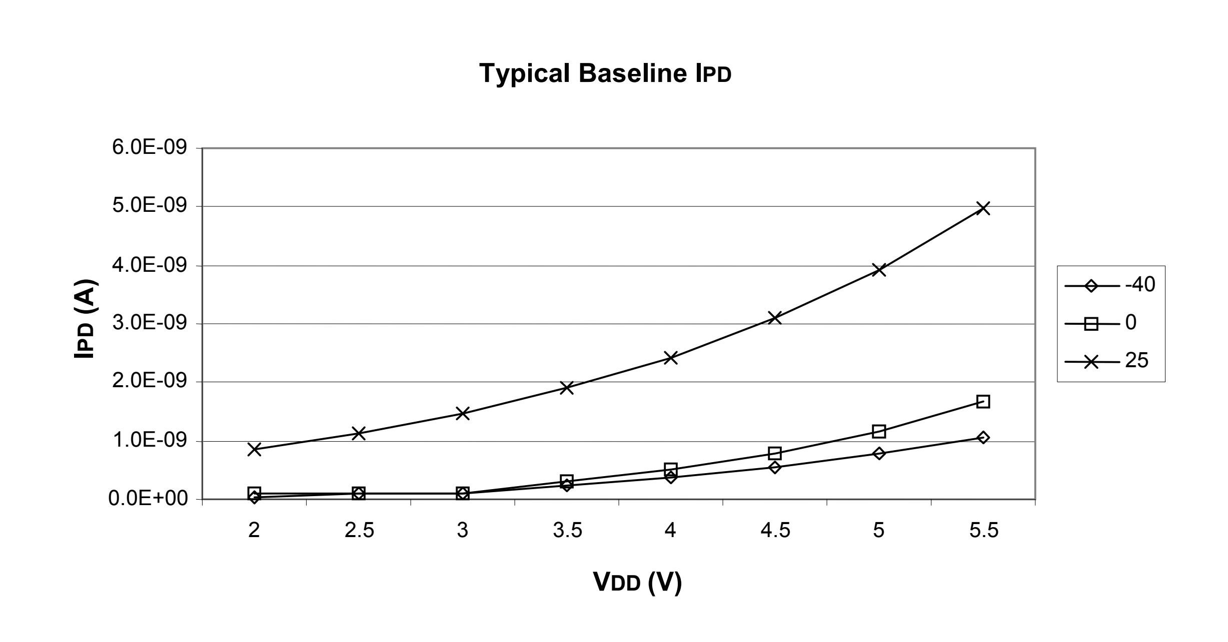


FIGURE 13-2: TYPICAL IPD vs. VDD OVER TEMP (+85°C)

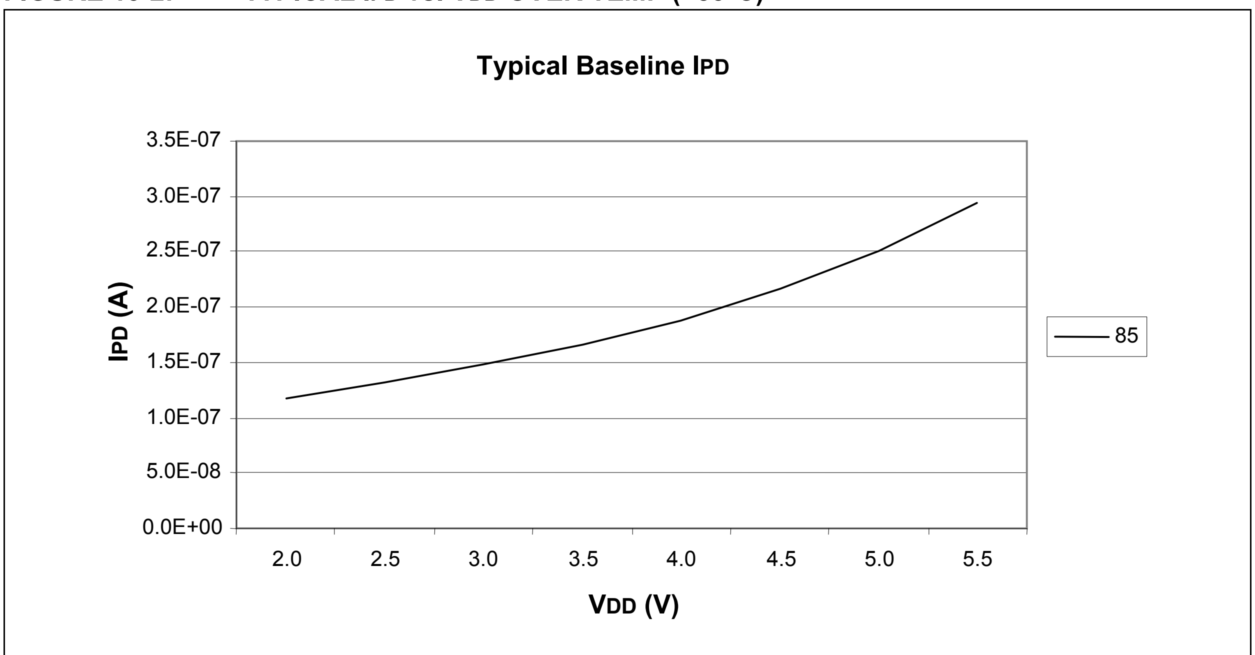


FIGURE 13-3: TYPICAL IPD vs. VDD OVER TEMP (+125°C)

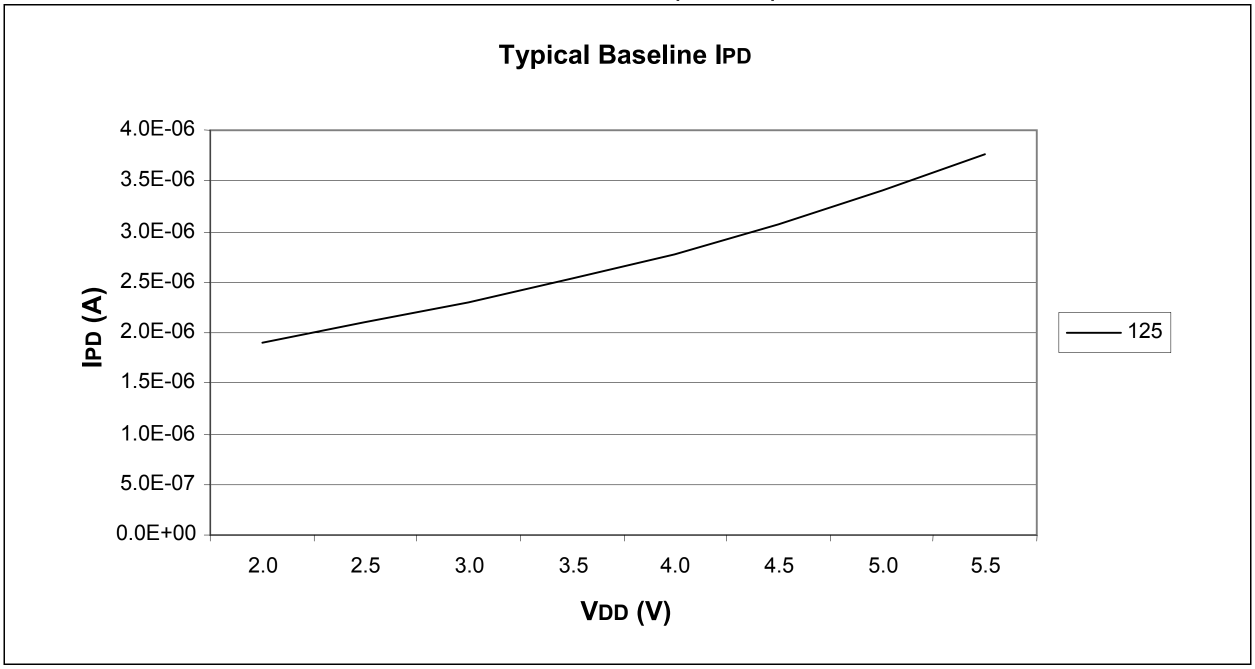


FIGURE 13-4: MAXIMUM IPD vs. VDD OVER TEMP (-40°C TO +25°C)

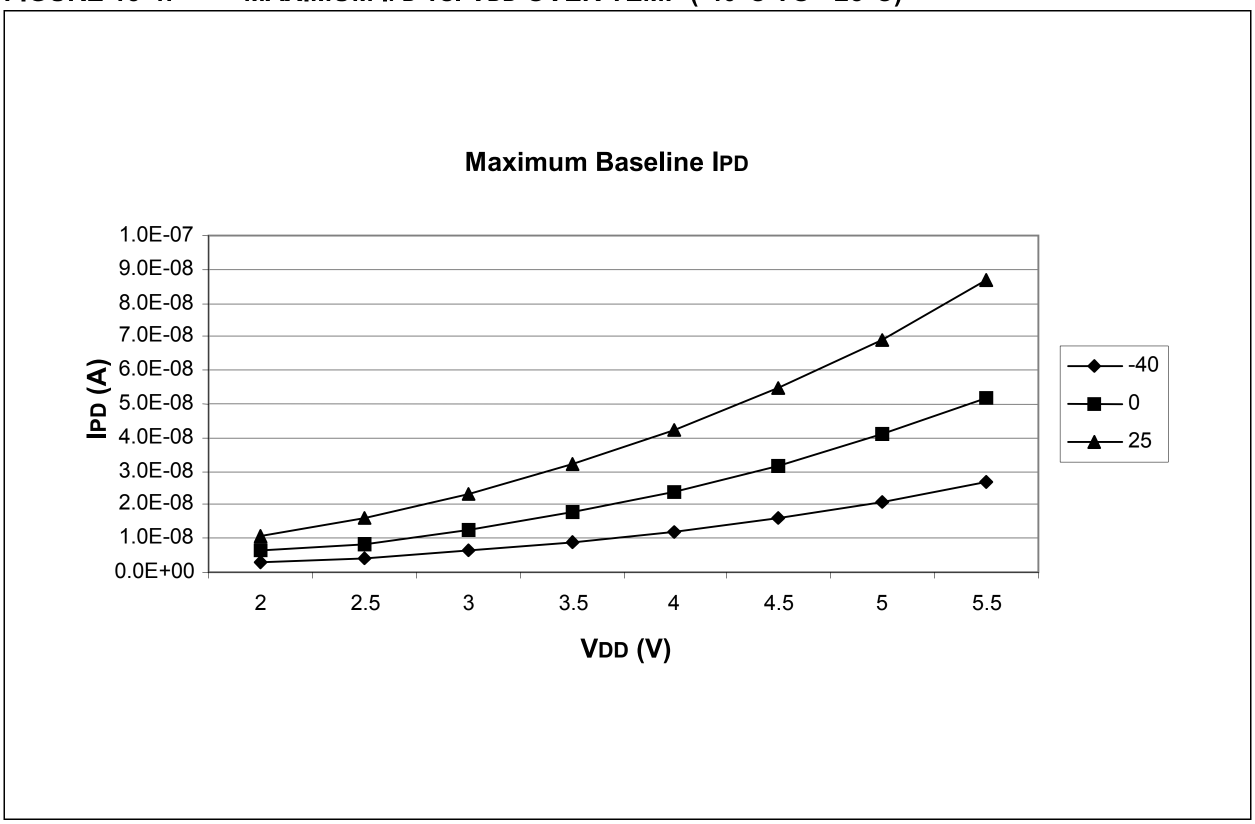


FIGURE 13-5: MAXIMUM IPD vs. VDD OVER TEMP (+85°C)

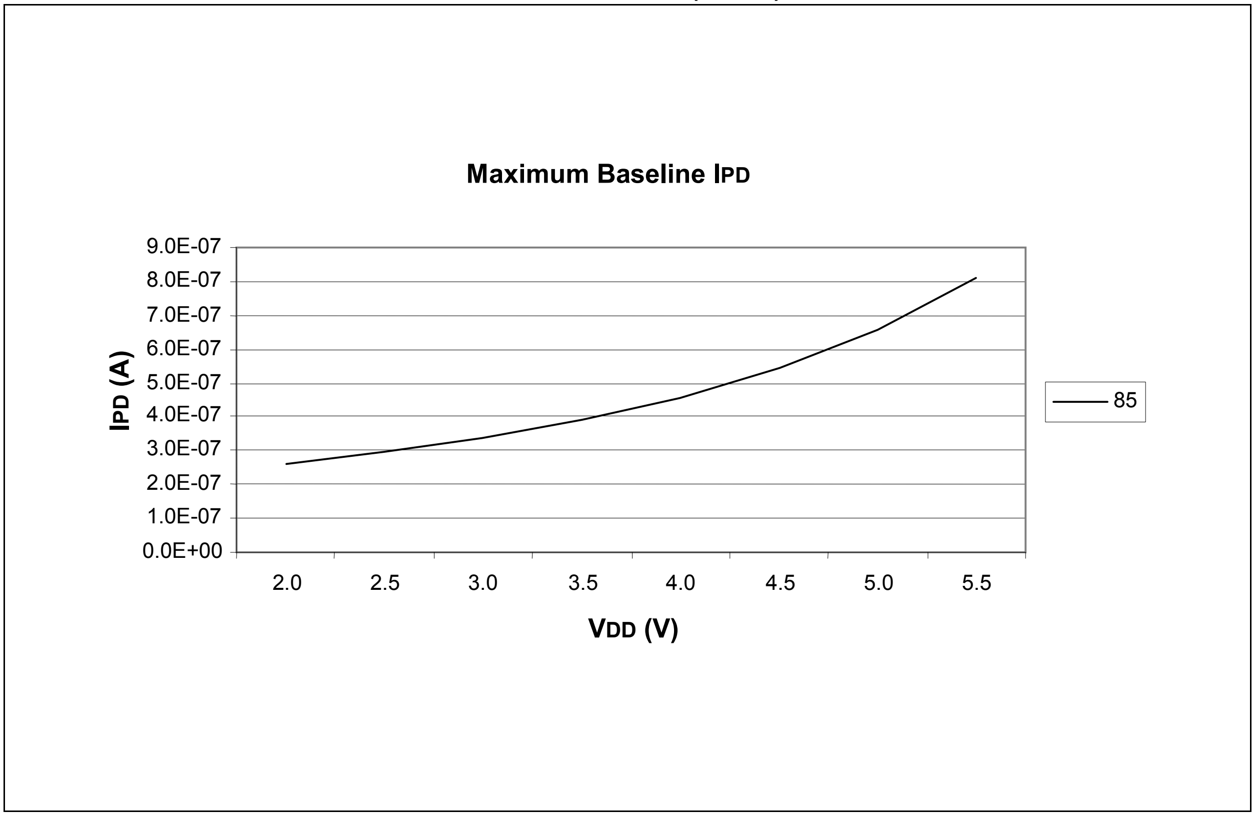


FIGURE 13-6: MAXIMUM IPD vs. VDD OVER TEMP (+125°C)

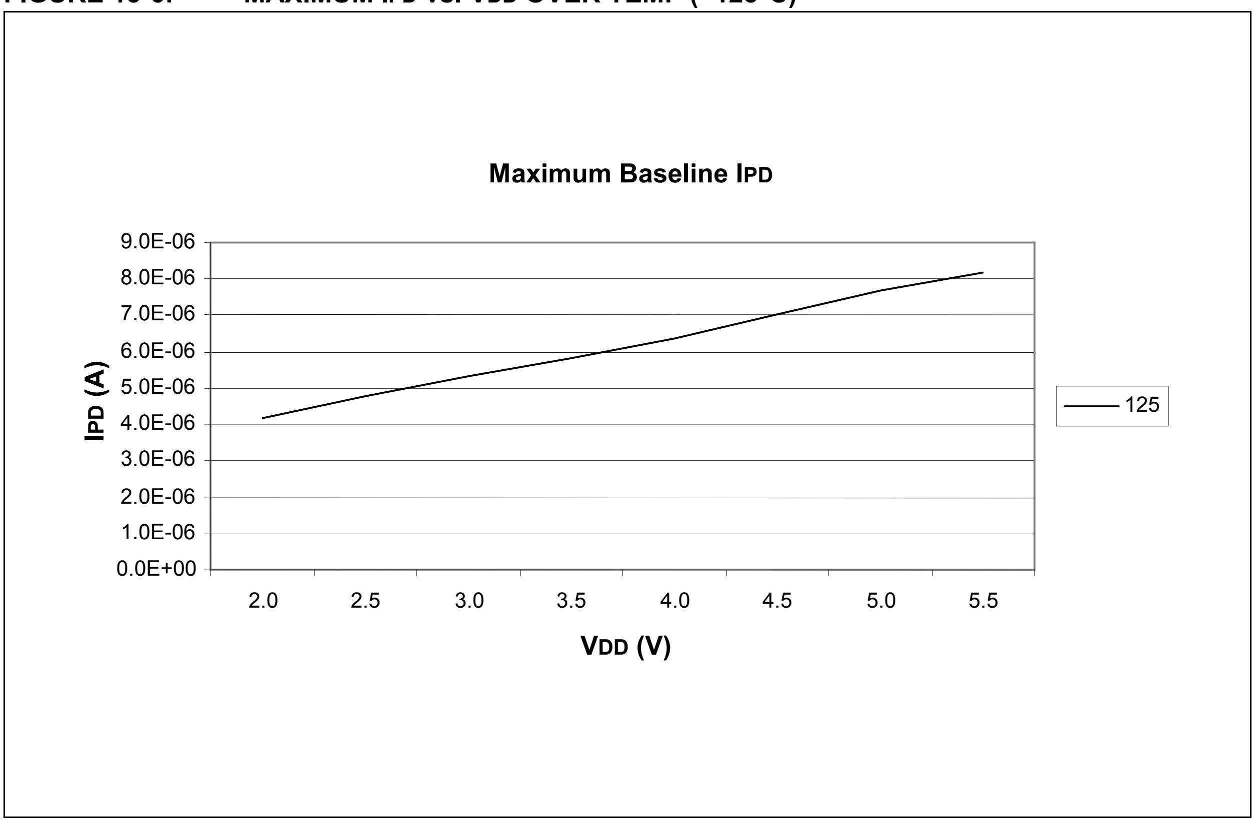


FIGURE 13-7: TYPICAL IPD WITH BOD ENABLED vs. VDD OVER TEMP (-40°C TO +125°C)

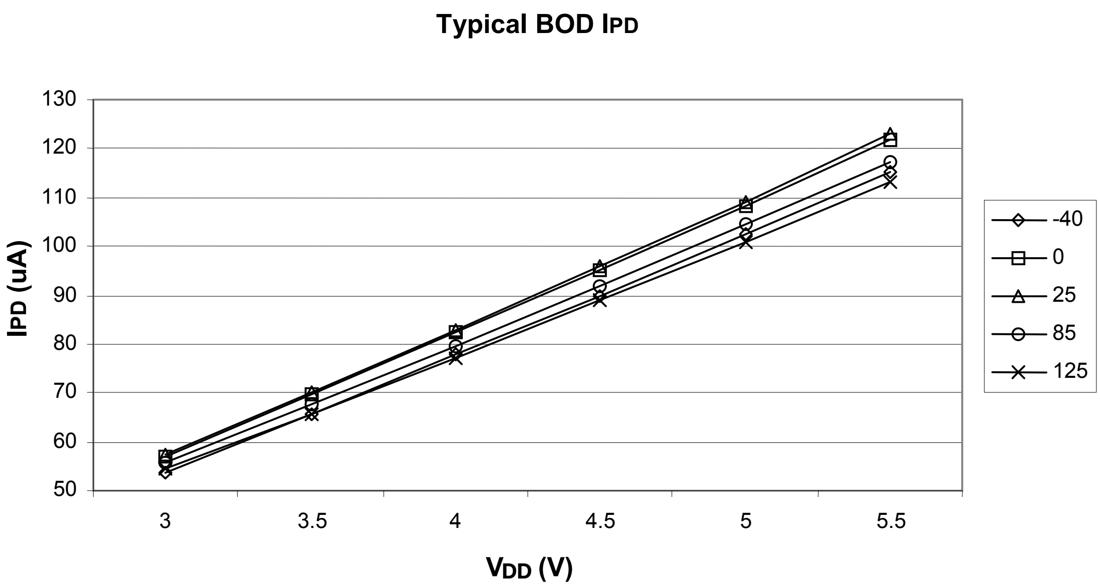


FIGURE 13-8: TYPICAL IPD WITH CMP ENABLED vs. VDD OVER TEMP (-40°C TO +125°C)

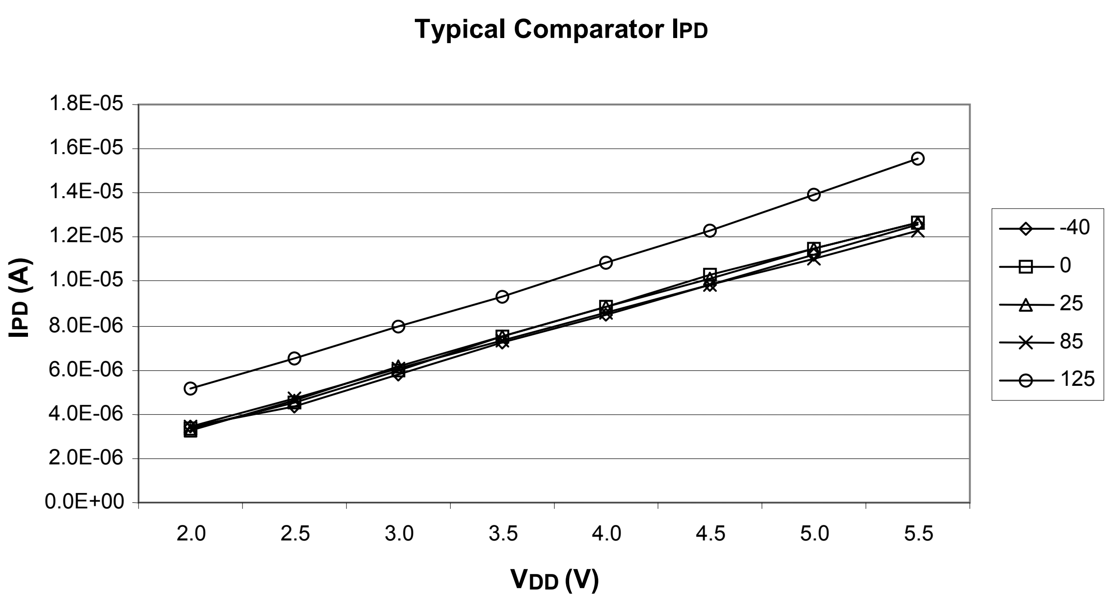


FIGURE 13-9: TYPICAL IPD WITH A/D ENABLED vs. VDD OVER TEMP (-40°C TO +25°C)

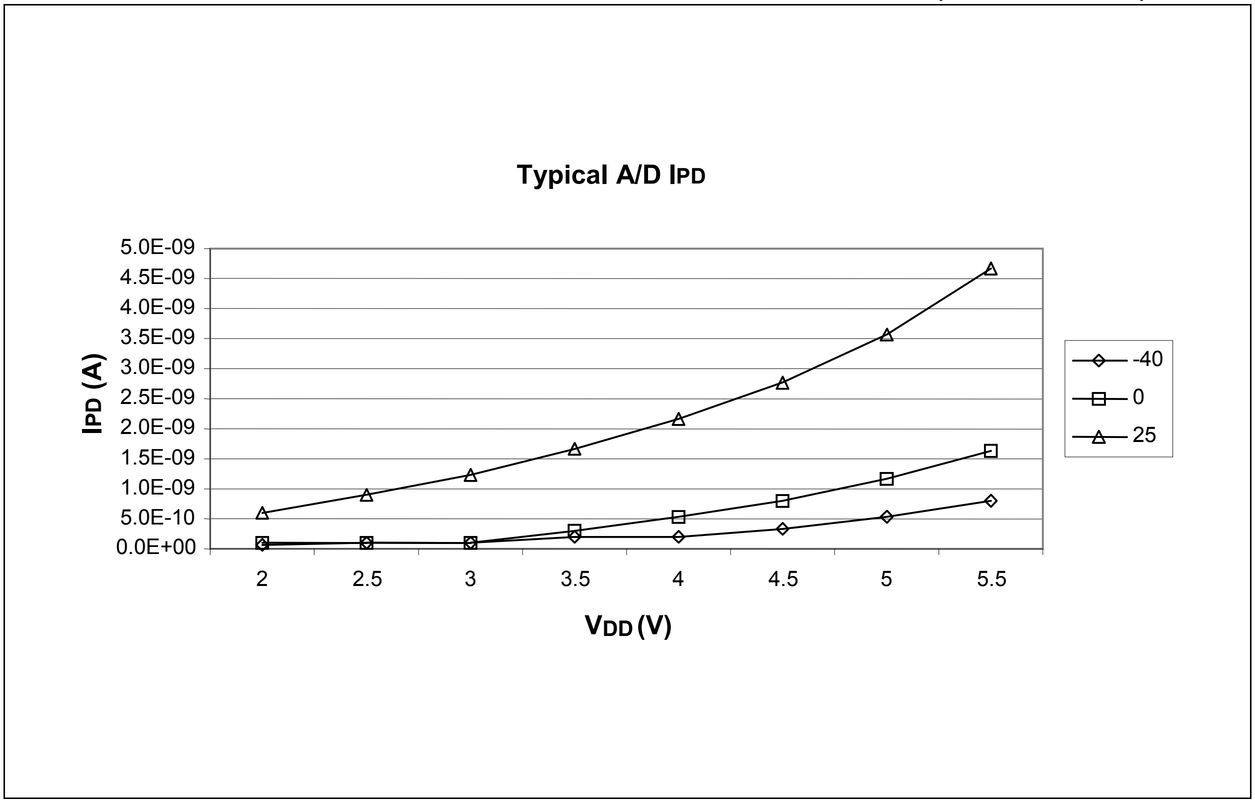


FIGURE 13-10: TYPICAL IPD WITH A/D ENABLED vs. VDD OVER TEMP (+85°C)

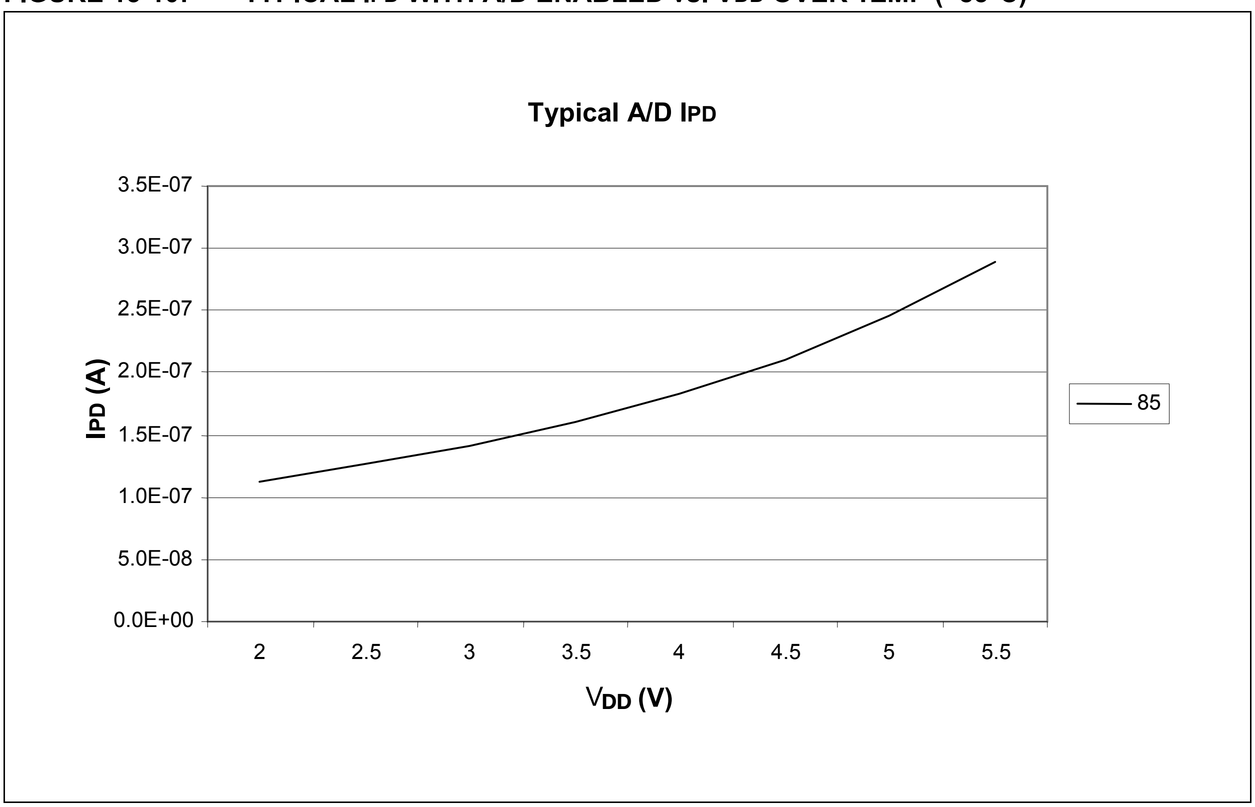


FIGURE 13-11: TYPICAL IPD WITH A/D ENABLED vs. VDD OVER TEMP (+125°C)

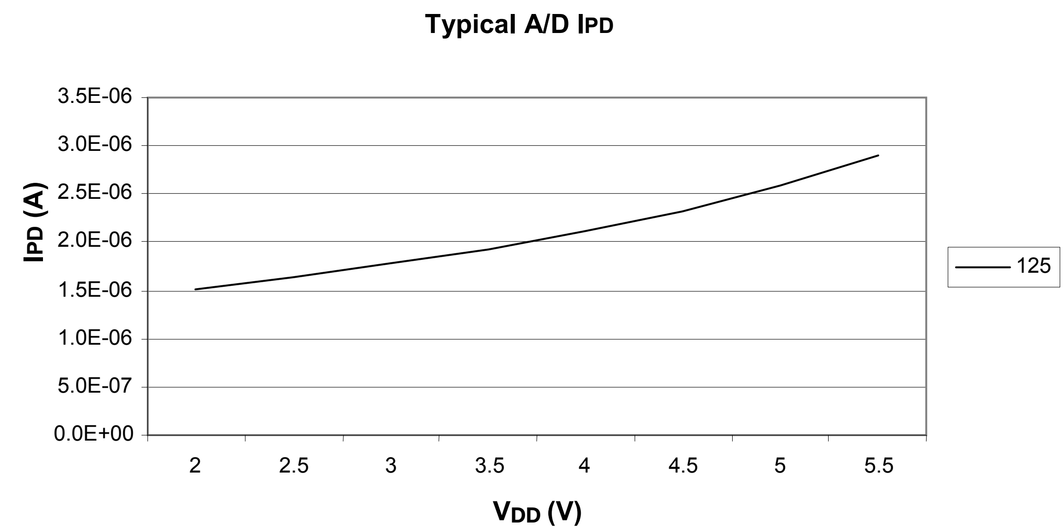


FIGURE 13-12: TYPICAL IPD WITH T1 OSC ENABLED vs. VDD OVER TEMP (-40°C TO +125°C), 32 kHZ, C1 AND C2=50 pF)

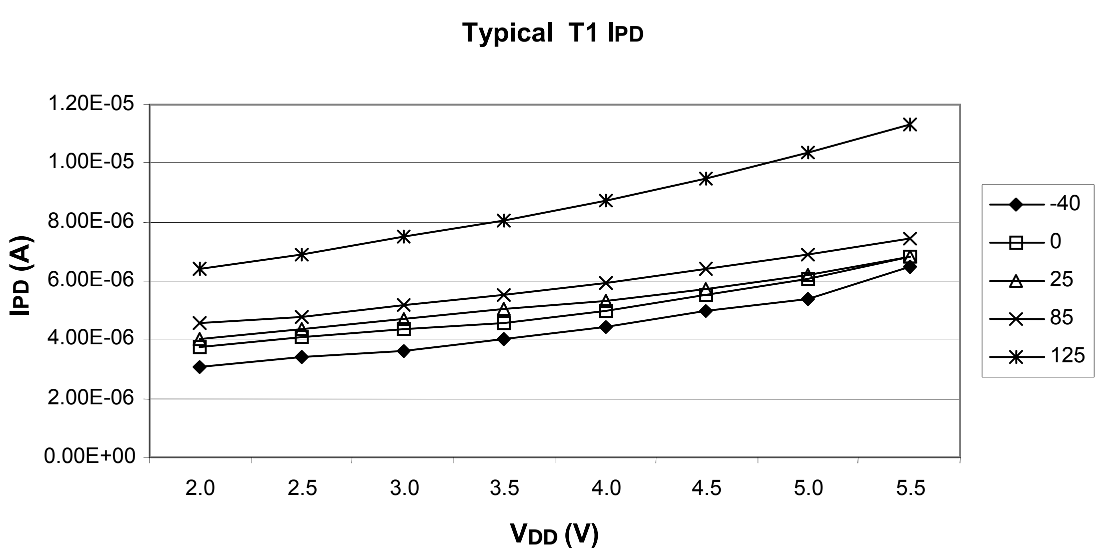


FIGURE 13-13: TYPICAL IPD WITH CVREF ENABLED vs. VDD OVER TEMP (-40°C TO +125°C)

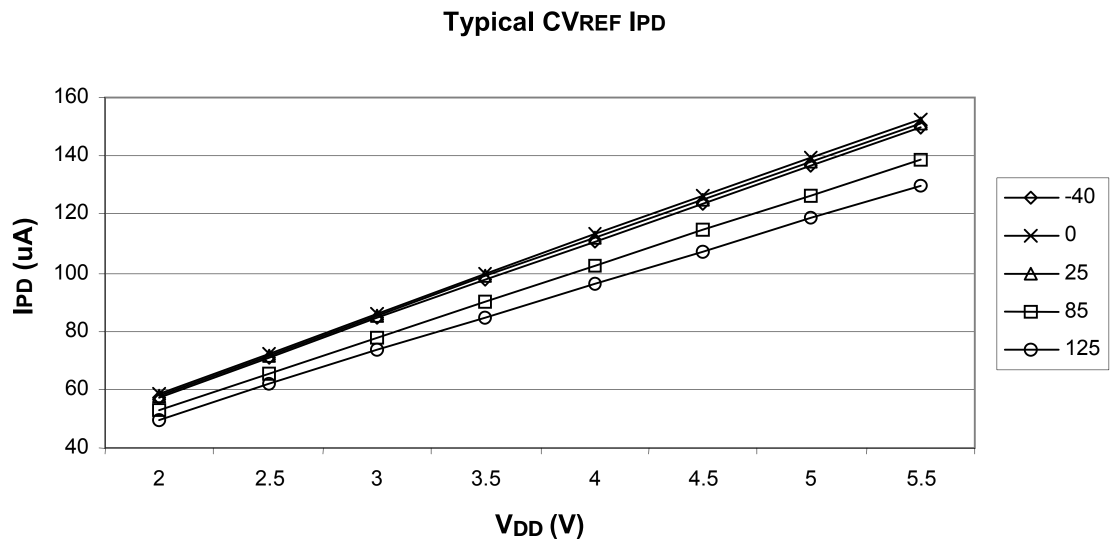


FIGURE 13-14: TYPICAL IPD WITH WDT ENABLED vs. VDD OVER TEMP (-40°C TO +125°C)

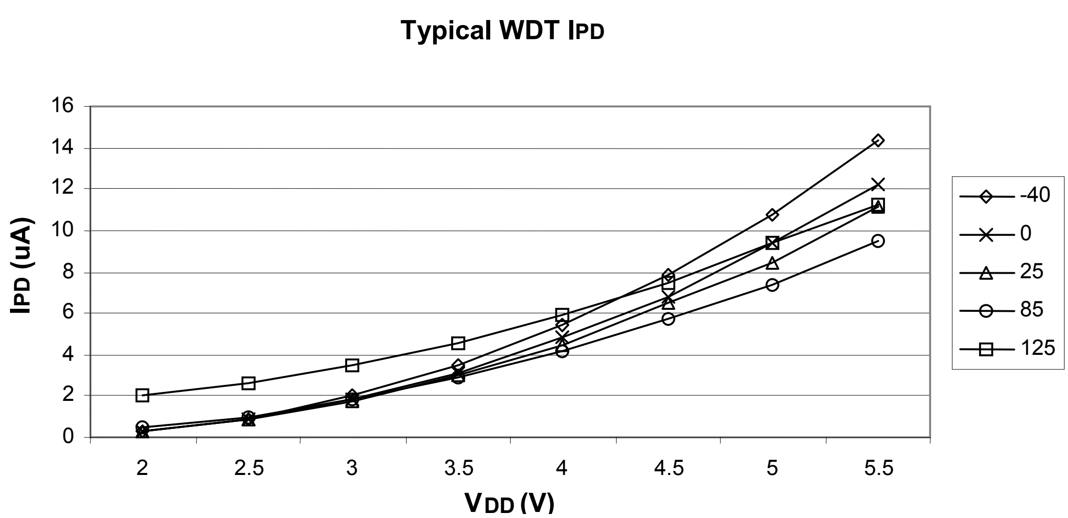


FIGURE 13-15: MAXIMUM AND MINIMUM INTOSC FREQ vs. TEMPERATURE WITH 0.1μF AND 0.01μF DECOUPLING (VDD = 3.5V)

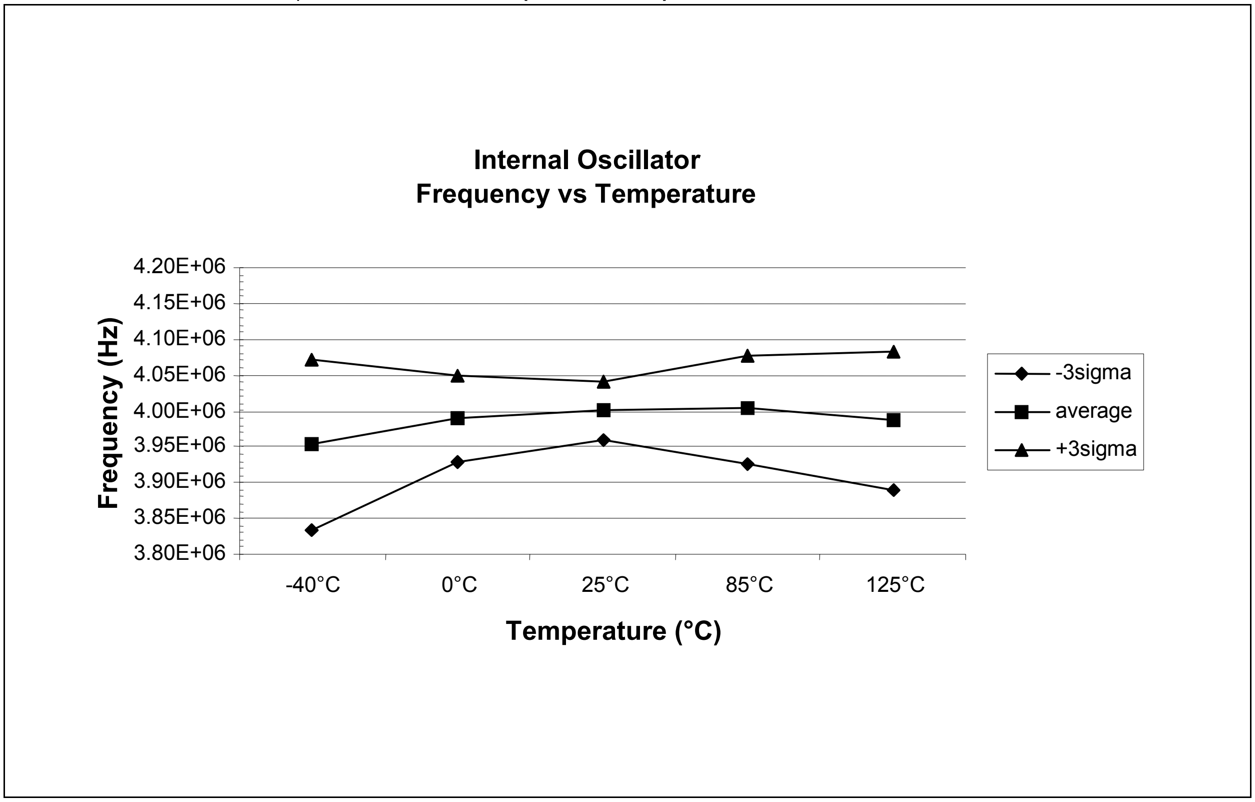


FIGURE 13-16: MAXIMUM AND MINIMUM INTOSC FREQ vs. VDD WITH 0.1μF AND 0.01μF DECOUPLING (+25°C)

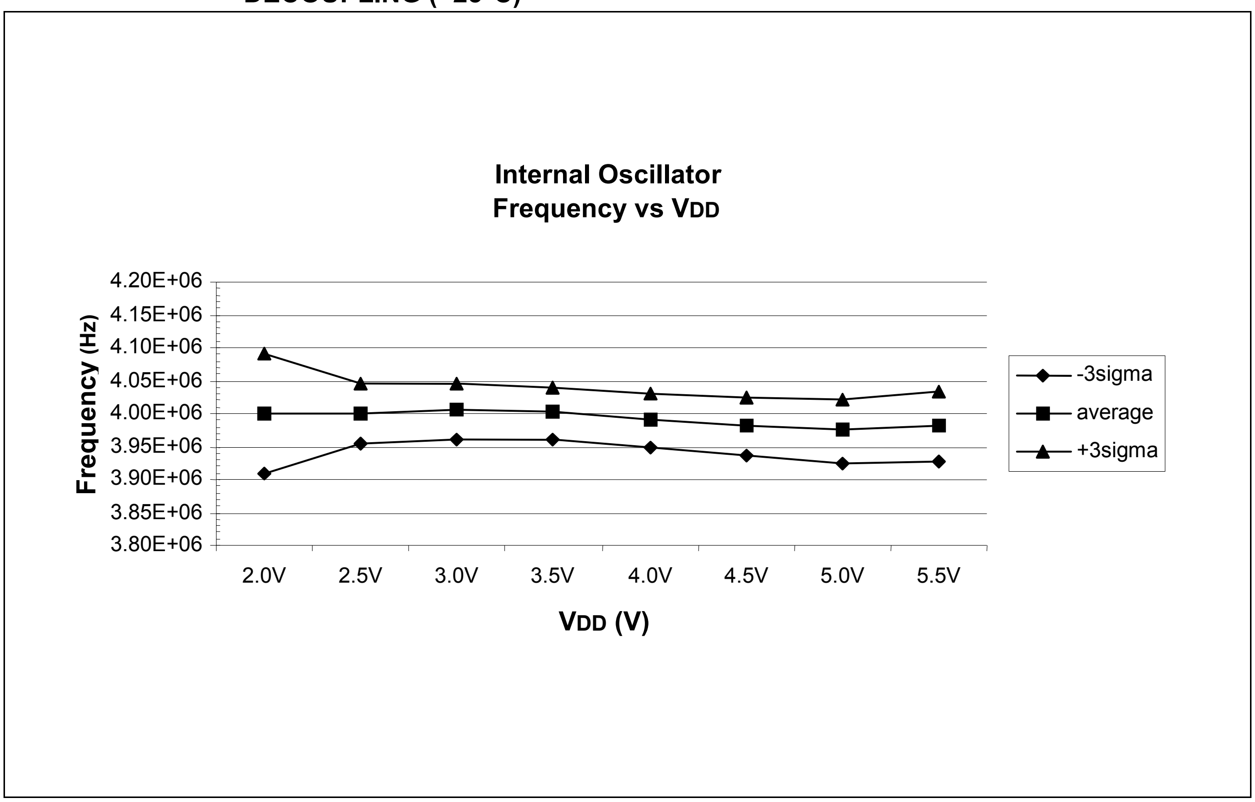
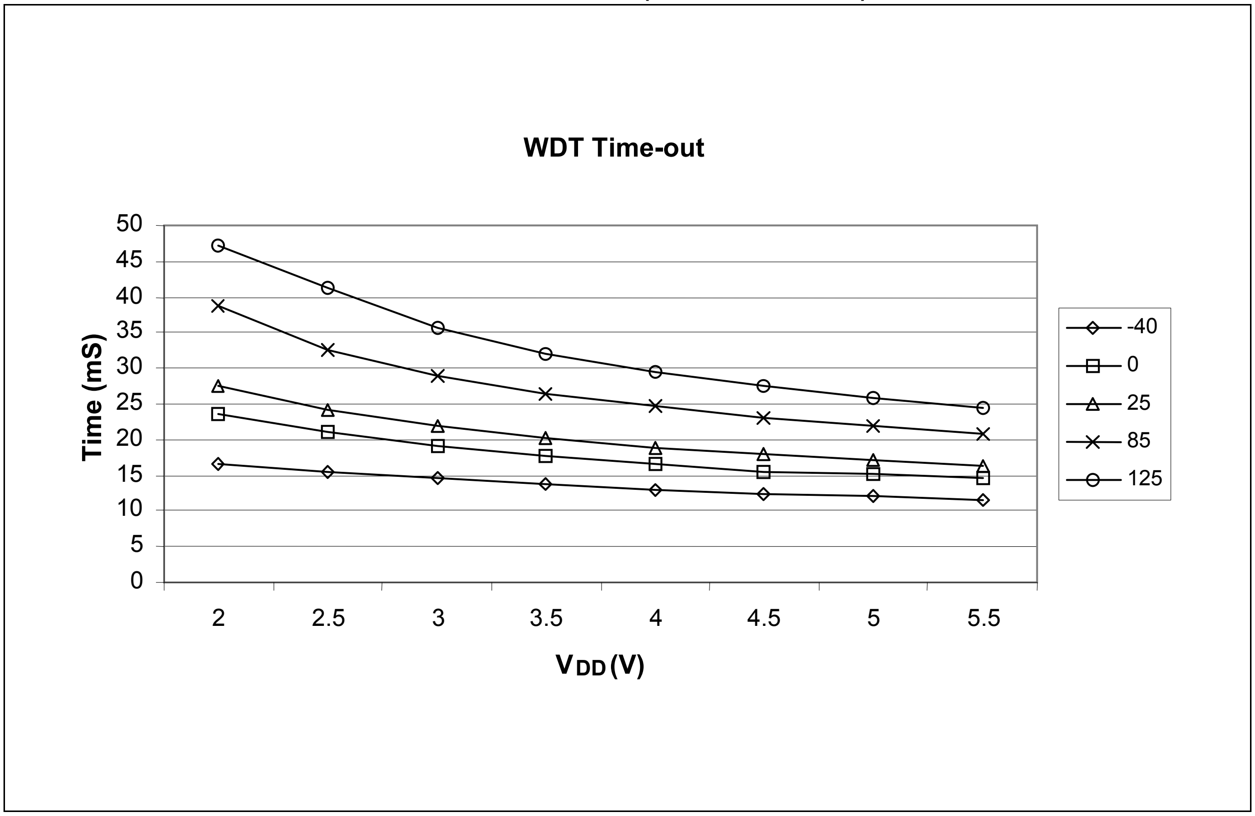


FIGURE 13-17: TYPICAL WDT PERIOD vs. VDD (-40°C TO +125°C)



NOTES:

14.0 PACKAGING INFORMATION

14.1 Package Marking Information

8-Lead PDIP (Skinny DIP)	Example
XXXXXXXX XXXXXNNN YYWW	12F629-I /017 e3 0215

8-Lead SOIC	Example
XXXXXXXX XXXXYYWW NNN	12F629-E /0215 e3 017

8-Lead DFN-S	Example
XXXXXXX XXXXXXX XXYYWW NNN	12F629 -E/021 e3 0215 017

8-Lead DFN (4x4 mm)	Example
XXXXXX XXXXXX YYWW NNN	XXXXXX XXXX e3 0610 017

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

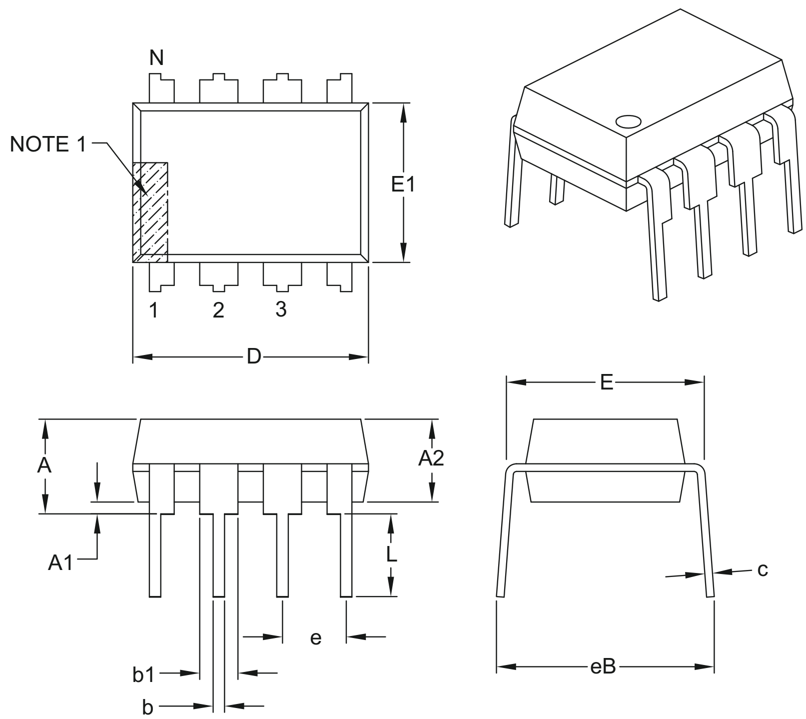
Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

14.2 Package Details

The following sections give the technical details of the packages.

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

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



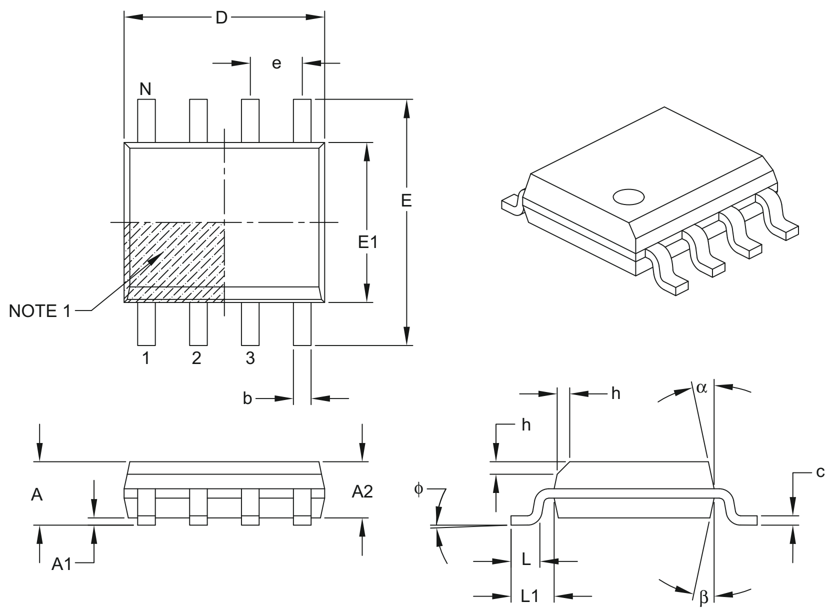
	Units	INCHES		
	Dimension Limits	MIN	NOM	MAX
Number of Pins	N	8		
Pitch	e	.100 BSC		
Top to Seating Plane	A	–	–	.210
Molded Package Thickness	A2	.115	.130	.195
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.290	.310	.325
Molded Package Width	E1	.240	.250	.280
Overall Length	D	.348	.365	.400
Tip to Seating Plane	L	.115	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.040	.060	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	–	–	.430

Notes:
1. Pin 1 visual index feature may vary, but must be located with the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-018B

8-Lead Plastic Small Outline (SN) – Narrow, 3.90 mm Body [SOIC]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	8		
Pitch	e	1.27 BSC		
Overall Height	A	–	–	1.75
Molded Package Thickness	A2	1.25	–	–
Standoff §	A1	0.10	–	0.25
Overall Width	E	6.00 BSC		
Molded Package Width	E1	3.90 BSC		
Overall Length	D	4.90 BSC		
Chamfer (optional)	h	0.25	–	0.50
Foot Length	L	0.40	–	1.27
Footprint	L1	1.04 REF		
Foot Angle	φ	0°	–	8°
Lead Thickness	c	0.17	–	0.25
Lead Width	b	0.31	–	0.51
Mold Draft Angle Top	α	5°	–	15°
Mold Draft Angle Bottom	β	5°	–	15°

Notes:

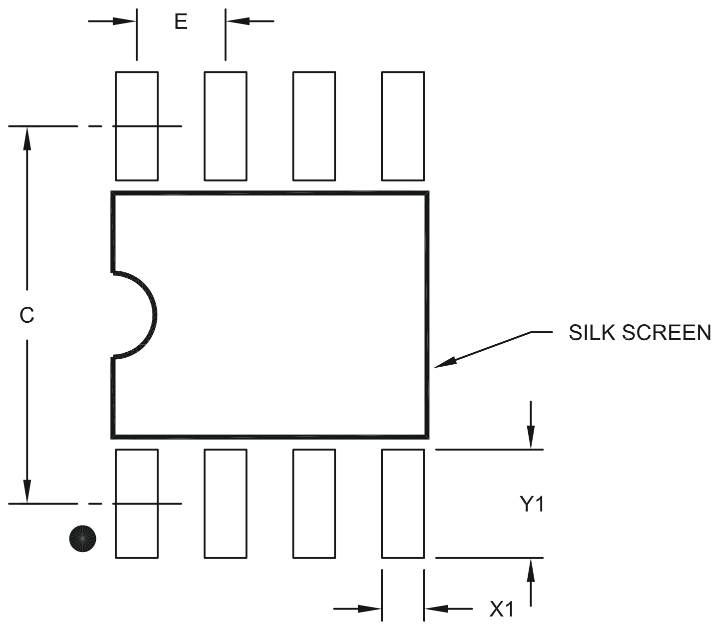
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

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

Microchip Technology Drawing C04-057B

8-Lead Plastic Small Outline (SN) – Narrow, 3.90 mm Body [SOIC]

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



RECOMMENDED LAND PATTERN

		Units	MILLIMETERS		
		Dimension Limits	MIN	NOM	MAX
Contact Pitch	E			1.27 BSC	
Contact Pad Spacing	C			5.40	
Contact Pad Width (X8)	X1				0.60
Contact Pad Length (X8)	Y1				1.55

Notes:

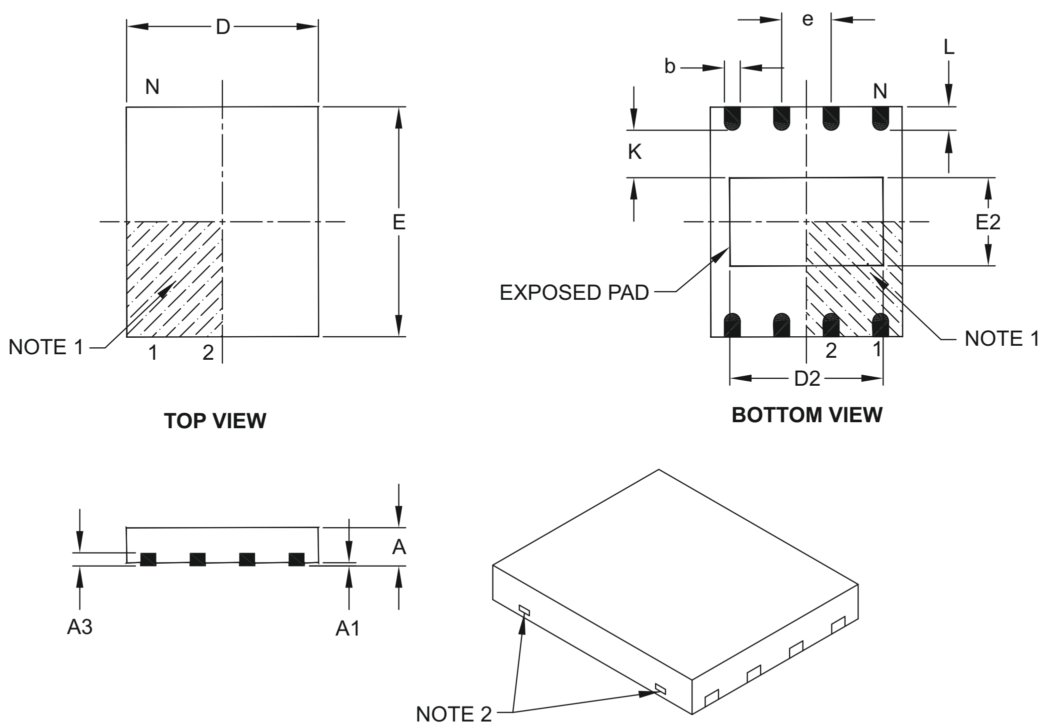
1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2057A

8-Lead Plastic Dual Flat, No Lead Package (MF) – 6x5 mm Body [DFN-S]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		8	
Pitch	e		1.27 BSC	
Overall Height	A	0.80	0.85	1.00
Standoff	A1	0.00	0.01	0.05
Contact Thickness	A3		0.20 REF	
Overall Length	D		5.00 BSC	
Overall Width	E		6.00 BSC	
Exposed Pad Length	D2	3.90	4.00	4.10
Exposed Pad Width	E2	2.20	2.30	2.40
Contact Width	b	0.35	0.40	0.48
Contact Length	L	0.50	0.60	0.75
Contact-to-Exposed Pad	K	0.20	–	–

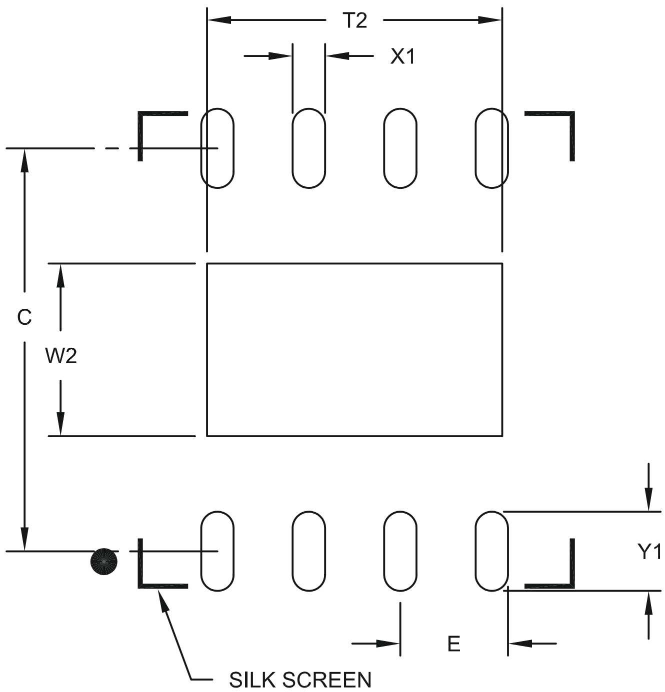
Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package may have one or more exposed tie bars at ends.
3. Package is saw singulated.
4. Dimensioning and tolerancing per ASME Y14.5M.
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

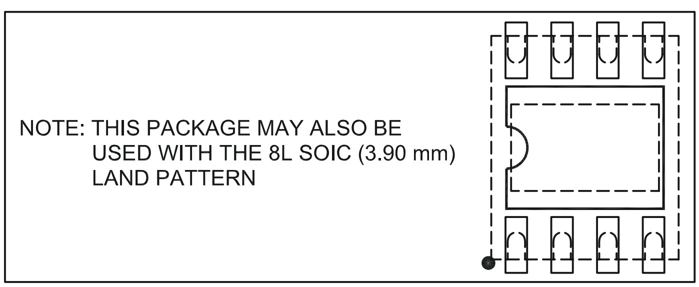
Microchip Technology Drawing C04-122B

8-Lead Plastic Dual Flat, No Lead Package (MF) - 6x5 mm Body [DFN-S]

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



RECOMMENDED LAND PATTERN



	Units	MILLIMETERS		
	Dimension Limits	MIN	NOM	MAX
Contact Pitch	E		1.27 BSC	
Optional Center Pad Width	W2			2.40
Optional Center Pad Length	T2			4.10
Contact Pad Spacing	C		5.60	
Contact Pad Width (X8)	X1			0.45
Contact Pad Length (X8)	Y1			1.10

Notes:

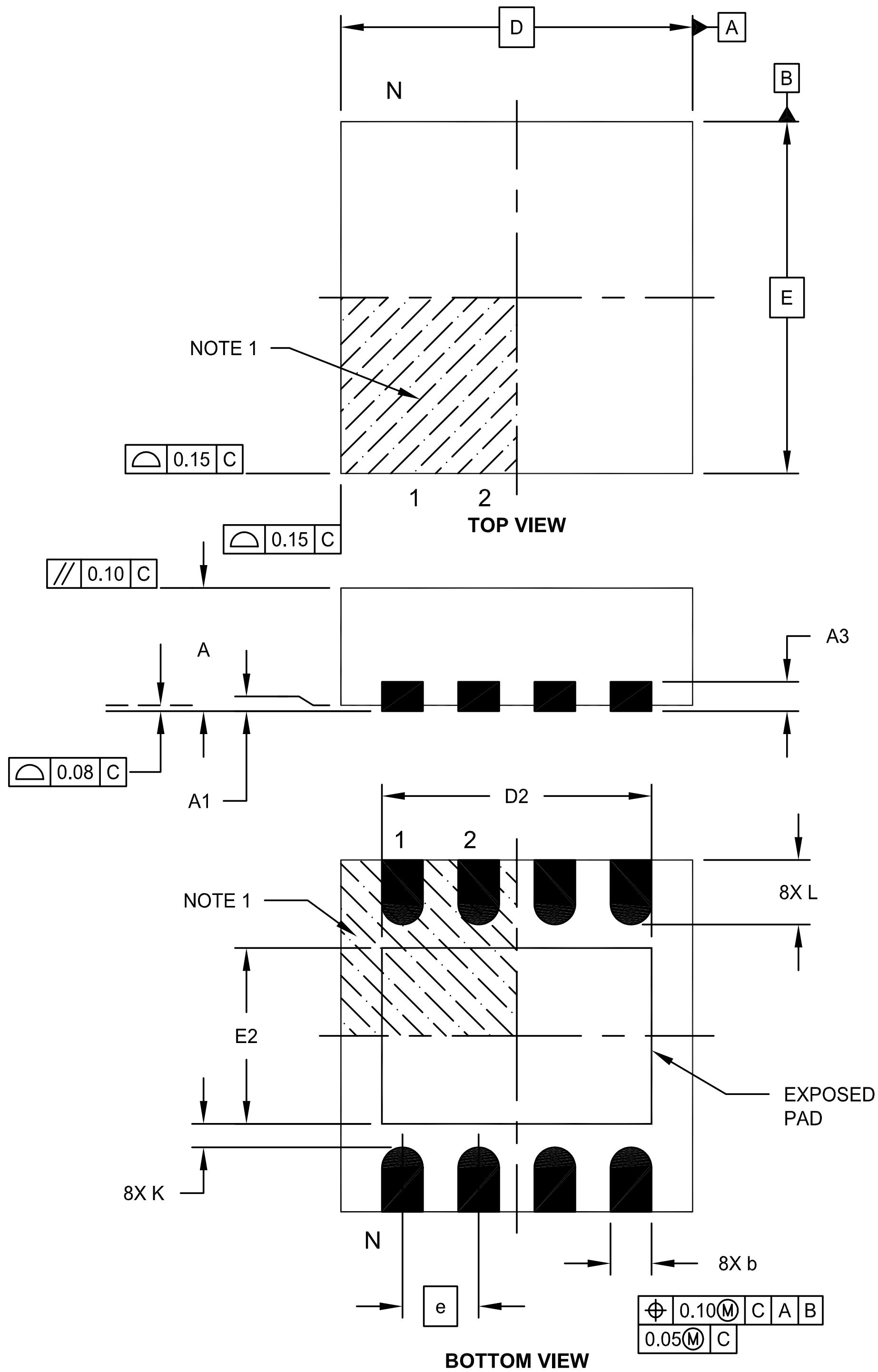
1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2122A

8-Lead Plastic Dual Flat, No Lead Package (MD) – 4x4x0.9 mm Body [DFN]

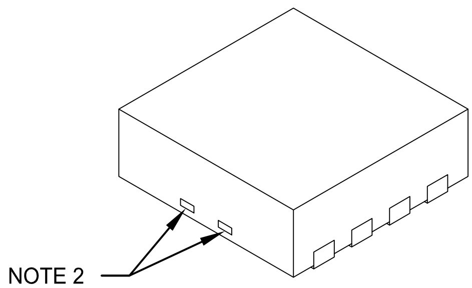
Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Microchip Technology Drawing C04-131E Sheet 1 of 2

8-Lead Plastic Dual Flat, No Lead Package (MD) – 4x4x0.9 mm Body [DFN]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		8	
Pitch	e		0.80 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3		0.20 REF	
Overall Length	D		4.00 BSC	
Exposed Pad Width	E2	2.60	2.70	2.80
Overall Width	E		4.00 BSC	
Exposed Pad Length	D2	3.40	3.50	3.60
Contact Width	b	0.25	0.30	0.35
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	-	-

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package may have one or more exposed tie bars at ends.
3. Package is saw singulated
4. Dimensioning and tolerancing per ASME Y14.5M

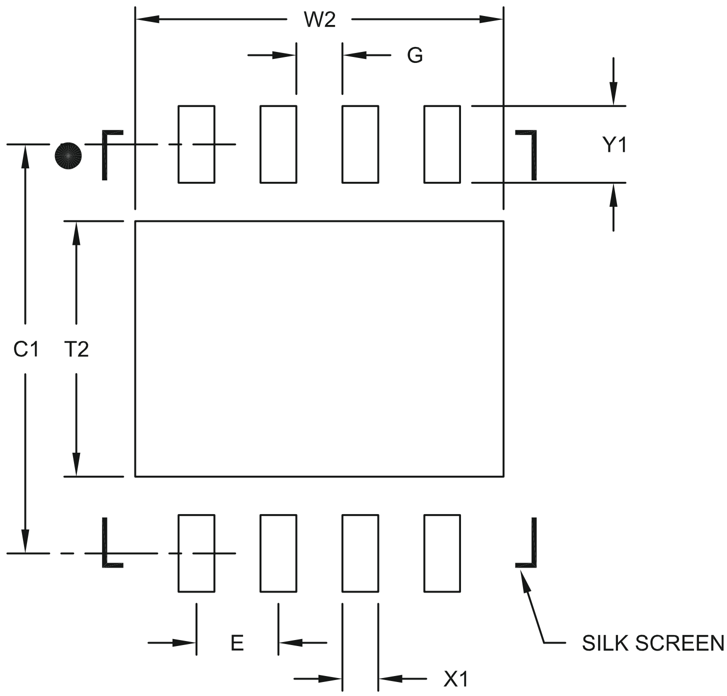
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

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

Microchip Technology Drawing C04-131E Sheet 2 of 2

8-Lead Plastic Dual Flat, No Lead Package (MD) – 4x4x0.9 mm Body [DFN]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.80 BSC		
Optional Center Pad Width	W2			3.60
Optional Center Pad Length	T2			2.50
Contact Pad Spacing	C1		4.00	
Contact Pad Width (X8)	X1			0.35
Contact Pad Length (X8)	Y1			0.75
Distance Between Pads	G	0.45		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2131B

NOTES:

APPENDIX A: DATA SHEET REVISION HISTORY

Revision A

This is a new data sheet.

Revision B

Added characterization graphs.

Updated specifications.

Added notes to indicate Microchip programmers maintain all Calibration bits to factory settings and the PIC12F675 ANSEL register must be initialized to configure pins as digital I/O.

Updated MLF-S package name to DFN-S.

Revision C

Revision D (01/2007)

Updated Package Drawings; Replace PICmicro with PIC; Revised Product ID example (b).

Revision E (03/2007)

Replaced Package Drawings (Rev. AM); Replaced Development Support Section.

Revision F (09/2009)

Updated Registers to new format; Added information to the "Package Marking Information" (8-Lead DFN) and "Package Details" sections (8-Lead Dual Flat, No Lead Package (MD) 4X4X0.9 mm Body (DFN)); Added Land Patterns for SOIC (SN) and DFN-S (MF) packages; Updated Register 3-2; Added MD Package to the Product identification System chapter; Other minor corrections.

Revision G (03/2010)

Updated the Instruction Set Summary section, adding pages 76 and 77.

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

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 PIC® DEVICES

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

D.1 PIC12C67X to PIC12F6XX

TABLE 1: FEATURE COMPARISON

Feature	PIC12C67X	PIC12F6XX
Max Operating Speed	10 MHz	20 MHz
Max Program Memory	2048 bytes	1024 bytes
A/D Resolution	8-bit	10-bit
Data EEPROM	16 bytes	64 bytes
Oscillator Modes	5	8
Brown-out Detect	N	Y
Internal Pull-ups	GP0/1/3	GP0/1/2/4/5
Interrupt-on-change	GP0/1/3	GP0/1/2/3/4/5
Comparator	N	Y

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.

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

INDEX

NOTES:

THE MICROCHIP WEB SITE

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

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

CUSTOMER CHANGE NOTIFICATION SERVICE

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

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

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support
- Development Systems Information Line

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

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

READER RESPONSE

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

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

To: Technical Publications Manager Total Pages Sent ________

RE: Reader Response

From: Name ________________________________

Company ________________________________

Address ________________________________

City / State / ZIP / Country ________________________________

Telephone: (_______) _________ - _________ FAX: (______) _________ - _________

Application (optional):

Would you like a reply?____Y ____N

Device: PIC12F629/675 Literature Number: DS41190G

Questions:

1. What are the best features of this document?

____

____

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

____

____

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

____

____

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

____

____

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

____

____

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

____

____

7. How would you improve this document?

____

____

PRODUCT IDENTIFICATION SYSTEM

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

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

Device:	PIC12F6XX: Standard V_{DD} range PIC12F6XXT: (Tape and Reel)
Temperature Range:	I = -40°C to +85°C (Industrial) E = -40°C to +125°C (Extended)
Package:	P = PDIP SN = SOIC (Gull wing, 3.90 mm body) MF = MLF-S MD = 8-Lead Plastic Dual Flat, No Lead (4X4) (DFN)
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/SN = Industrial temp., SOIC package, 20 MHz.

* JW Devices are UV erasable and can be programmed to any device configuration. JW Devices meet the electrical requirement of each oscillator type.

WORLDWIDE SALES AND SERVICE

AMERICAS

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

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

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

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

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

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

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

Kokomo
Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

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

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

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

ASIA/PACIFIC

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

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

China - Beijing
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

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

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

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

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

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

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

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

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

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

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

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

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

ASIA/PACIFIC

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

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

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

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

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

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

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

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

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

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

Taiwan - Hsin Chu
Tel: 886-3-6578-300
Fax: 886-3-6578-370

Taiwan - Kaohsiung
Tel: 886-7-536-4818
Fax: 886-7-536-4803

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

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

EUROPE

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

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

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

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

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

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

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

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

01/05/10

Description

The 74HC595 is an high speed CMOS device.

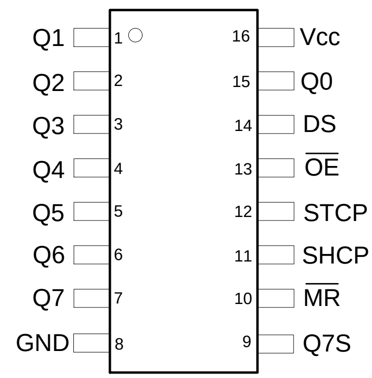
An eight bit shift register accpets data from the serial input (DS) on each positive transition of the shift register clock (SHCP). When asserted low the reset function ($\overline{MR}$) sets all shift register values to zero and is indepent of all clocks.

Data from the input serial shift register is placed in the output register with a rising pulse on the storages resister clock (STCP). With the output enable ($\overline{OE}$) asserted low the 3-state outputs Q0-Q7 become active and present th

All registers capture data on rising edge and change output on the falling edge. If both clocks are connected together the input shift register is always one clock cycle ahead of the output register.

Pin Assignments

(Top View)



SO-16 / TSSOP-16

Features

- Wide Supply Voltage Range from 2.0V to 6.0V
- Sinks or Sources 8mA at V_{CC} = 4.5V
- CMOS Low Power Consumption
- Schmitt Trigger Action at All Inputs
- Inputs Accept up to 6.0V
- ESD Protection Tested per JESD 22
  - Exceeds 200-V Machine Model (A115-A)
  - Exceeds 2000-V Human Body Model (A114-A)
  - Exceeds 1000-V Charged Device Model (C101C)
- Latch-Up Exceeds 250mA per JESD 78, Class II
- **Totally Lead-Free & Fully RoHS Compliant (Notes 1 & 2)**
- **Halogen and Antimony Free. “Green” Device (Note 3)**

Applications

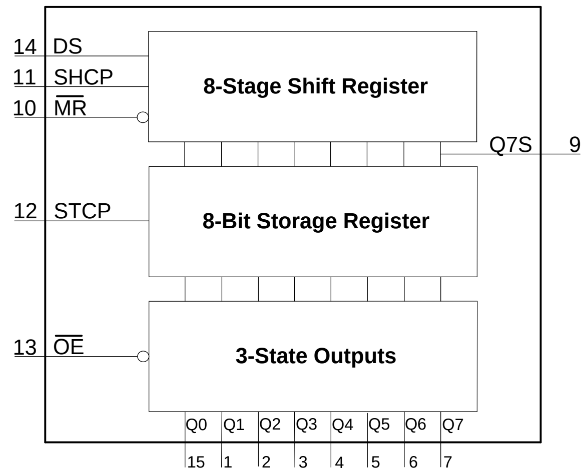
- General Purpose Logic
- Serial to Parallel Data Conversion
- Capture and Hold Data for Extended Periods of Time
- Allow Simple Serial Bit Streams from a Microcontroller to Control as Many Peripheral Lines as Needed
- Wide Array of Products such as:
  - Computer Peripherals
  - Appliances
  - Industrial Control

Notes:
1. No purposely added lead. Fully EU Directive 2002/95/EC (RoHS), 2011/65/EU (RoHS 2) & 2015/863/EU (RoHS 3) compliant.
2. See https://www.diodes.com/quality/lead-free/ for more information about Diodes Incorporated’s definitions of Halogen- and Antimony-free, "Green" and Lead-free.
3. Halogen- and Antimony-free "Green” products are defined as those which contain <900ppm bromine, <900ppm chlorine (<1500ppm total Br + Cl) and <1000ppm antimony compounds.

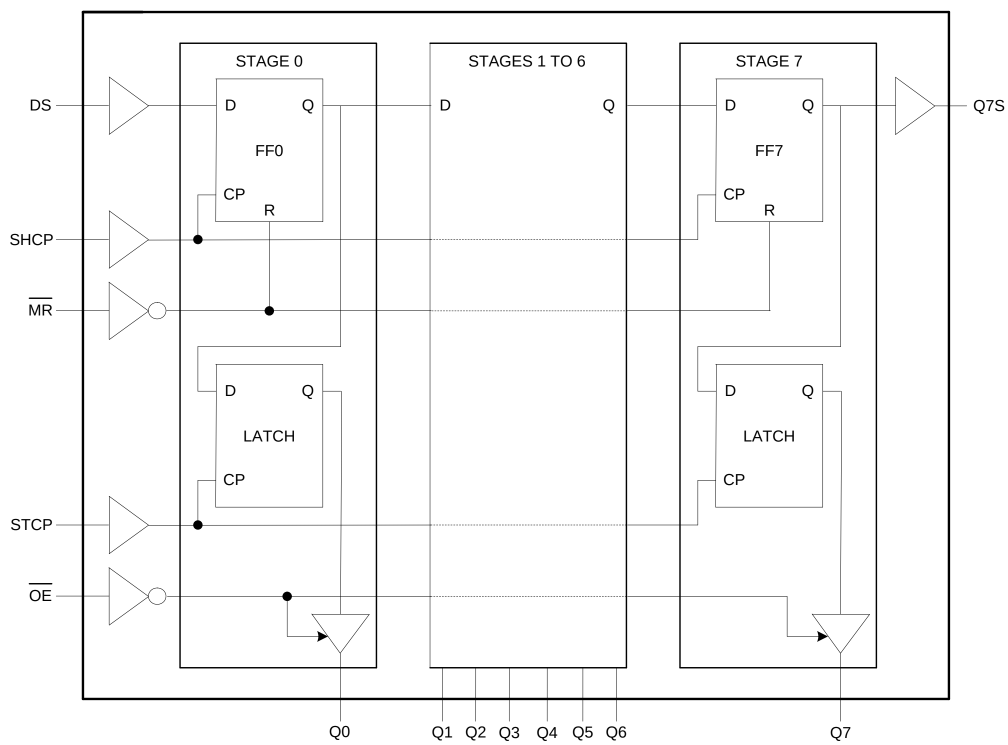
Pin Descriptions

Pin Number	Pin Name	Function
1	Q1	Parallel Data Output 1
2	Q2	Parallel Data Output 2
3	Q3	Parallel Data Output 3
4	Q4	Parallel Data Output 4
5	Q5	Parallel Data Output 5
6	Q6	Parallel Data Output 6
7	Q7	Parallel Data Output 7
8	GND	Ground
9	Q7S	Serial Data Output
10	$\overline{MR}$	Master Reset Input
11	SHCP	Shift Register Clock Input
12	STCP	Storage Register Clock Input
13	$\overline{OE}$	Output Enable Input
14	DS	Serial Data Input
15	Q0	Parallel Data Output 0
16	V_{CC}	Supply Voltage

Functional Diagram



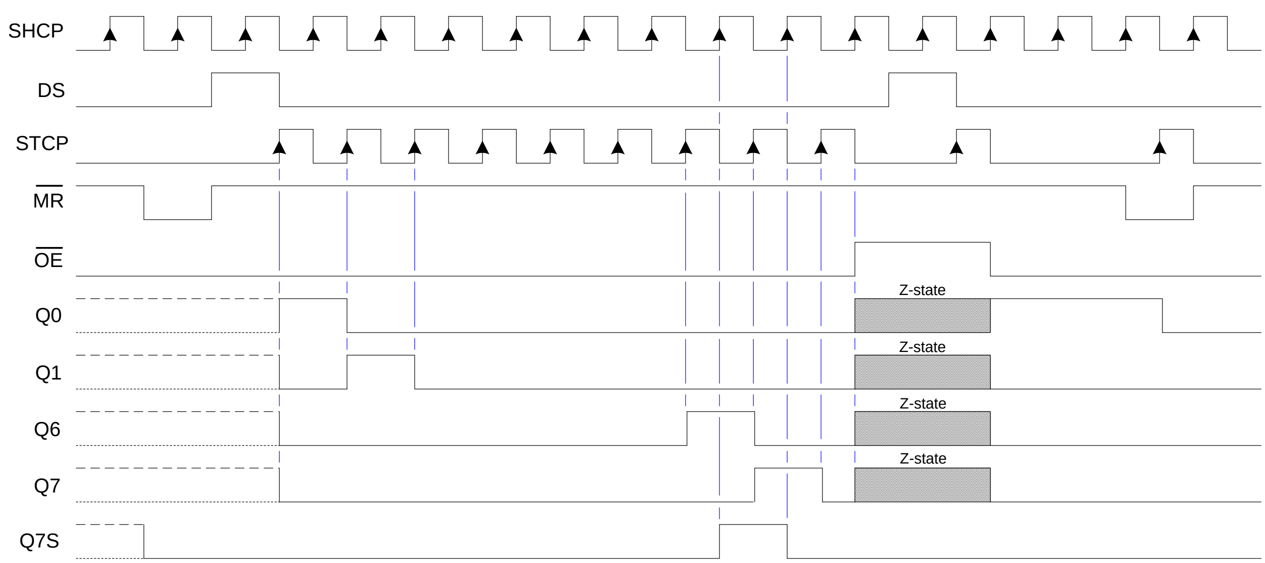
Logic Diagram



Functional Description and Timing Diagram

Control				Input	Output		Function
SHCP	STCP	$\overline{OE}$	$\overline{MR}$	DS	Q7S	Qn	
X	X	L	L	–	L	NC	Low-level asserted on MR clears shift register. Storage register is unchanged.
X	↑	L	L	–	L	L	Empty shift register transferred to storage register.
X	X	H	L	–	L	Z	Shift register remains clear; All Q ouputs in Z state.
↑	X	L	H	–	Q6S	NC	HIGH is shifted into first stage of Shift Register Contents of each register shifted to next register. The content of Q6S has been shifted to Q7S and now appears on device pin Q7S.
X	↑	L	H	–	NC	QnS	Contents of shift register copied to storage register. With output now in active state the storage resister contents appear on Q outputs.
↑	↑	L	H	–	Q6S	QnS	Contents of shift register copied to output register then shift register shifted.

H=HIGH Voltage State
L=LOW Voltage State
↑=LOW to HIGH Transition
X= Don’T Care – High or Low (Not Floating)
NC= No Change
Z= High-Impedance State



Absolute Maximum Ratings (Note 4) (@T_A = +25°C, unless otherwise specified.)

Symbol	Description		Rating	Unit
ESD HBM	Human Body Model ESD Protection		2	kV
ESD CDM	Charged Device Model ESD Protection		1	kV
ESD MM	Machine Model ESD Protection		200	V
V_{CC}	Supply Voltage Range		-0.5 to +7.0	V
V_I	Input Voltage Range		-0.5 to +7.0	V
V_o	Voltage Applied to Output in High or Low State		-0.3 to V_{CC} +0.5	V
I_{IK}	Input Clamp Current V_I < -0.5V		-20	mA
I_{IK}	Input Clamp Current V_I > V_{CC} +0.5V		20	mA
I_{OK}	Output Clamp Current V_O < -0.5V		-20	mA
I_{OK}	Output Clamp Current V_O > V_{CC} +0.5V		20	mA
I_O	Continuous Output Current	Q7 Standard Output	±25	mA
		Qn Bus Driver Outputs	±35	mA
I_{CC}	Continuous Current through Vdd or GND		70	mA
I_{GND}	Continuous Current through Vdd or GND		-70	mA
T_J	Operating Junction Temperature		-40 to +150	°C
T_{STG}	Storage Temperature		-65 to +150	°C
P_{TOT}	Total Power Dissipation		500	mW

Note: 4. Stresses beyond the absolute maximum may result in immediate failure or reduced reliability. These are stress values and device operation should be within recommend values.

Recommended Operating Conditions (Note 5) (@T_A = +25°C, unless otherwise specified.)

Symbol	Parameter	Conditions	Min	Max	Unit
V_{CC}	Supply Voltage	–	2.0	6.0	V
V_I	Input Voltage	–	0	V_{CC}	V
V_O	Output Voltage	Active Mode	0	V_{CC}	V
Δt/ΔV	Input Transition Rise or Fall Rate	V_{CC} = 2.0V	–	1000	ns/V
		V_{CC} = 4.5V	–	500	
		V_{CC} = 6.0V	–	400	–
T_A	Operating Free-Air Temperature	–	-40	+125	°C

Note: 5. Unused inputs should be held at V_{CC} or Ground.

Electrical Characteristics (@T_A = +25°C, unless otherwise specified.)

Symbol	Parameter	Test Conditions	V_{CC}	T_A = +25°C			T_A = -40°C to +85°C		T_A = -40°C to +125°C		Unit
				Min	Typ	Max	Min	Max	Min	Max	
V_{IH}	High-Level Input Voltage	–	2.0V	1.5	1.2	–	1.5	–	1.5	–	V
		–	4.5V	3.15	2.4	–	3.15	–	3.15	–	
		–	6.0V	4.2	3.2	–	4.2	–	4.2	–	
V_{IL}	Low-Level Input Voltage	–	2.0V	–	0.8	0.5	–	0.5	–	0.5	V
		–	4.5V	–	2.1	1.35	–	1.35	–	1.35	
		–	6.0V	–	2.8	1.8	–	1.8	–	1.8	
V_{OH}	High-Level Output Voltage	I_{OH} = -20µA All Outputs	2.0V	1.9	2.0	–	1.9	–	1.9	–	V
			4.5V	4.4	4.5	–	4.4	–	4.4	–	
			6.0V	5.9	6.0	–	5.9	–	5.9	–	
	Q7S Output	I_{OH} = -4.0mA	4.5V	3.84	4.32	–	4.32	–	3.7	–	
		I_{OH} = -5.2mA	6.0V	5.34	5.81	–	5.81	–	5.2	–	
	Qn Bus Outputs	I_{OH} = -6.0mA	4.5V	3.84	4.32	–	4.32	–	3.7	–	
		I_{OH} = -7.8mA	6.0V	5.34	5.81	–	5.81	–	5.2	–	
V_{OL}	Low-Level Output Voltage	I_{OL} = 20µA All Outputs	2.0V	–	0	0.1	–	0.1	–	0.1	V
			4.5V	–	0	0.1	–	0.1	–	0.1	
			6.0V	–	0	0.1	–	0.1	–	0.1	
	Q7S Output	I_{OL} = 4.0mA	4.5V	–	.15	0.33	–	0.33	–	0.4	
		I_{OL} = 5.2mA	6.0V	–	.16	0.33	–	0.33	–	0.4	
	Qn Bus Outputs	I_{OL} = 6.0mA	4.5V	–	.15	0.33	–	0.33	–	0.4	
		I_{OL} = 7.8mA	6.0V	–	.16	0.33	–	0.33	–	0.4	
I_I	Input Current	V_I =GND to 5.5V	6.0V	–	–	±0.1	–	± 1	–	± 1	µA
I_{OZ}	OFF-State Output Current	Qn Internal High or Low Vo = V_{CC} or GND	6.0V	–	–	± 5	–	± 5	–	± 10	µA
I_{CC}	Supply Current	V_I = GND or V_{CC} I_O = 0	6.0V	–	–	8.0	–	80	–	160	µA
C_i	Input Capacitance	V_I = V_{CC} or GND	6.0V	–	4	10	–	10	–	10	pF

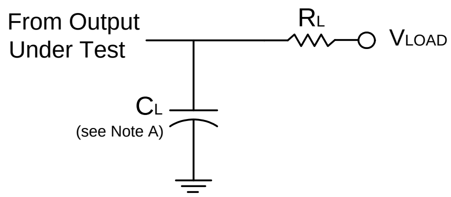
Operating Characteristics (@T_A = +25°C, unless otherwise specified.)

Parameter		Test Conditions	V_{CC} = 5V Typ	Unit
C_{pd}	Power Dissipation Capacitance	f = 1 MHz All Outputs Switching-No Load	43	pF

Switching Characteristics

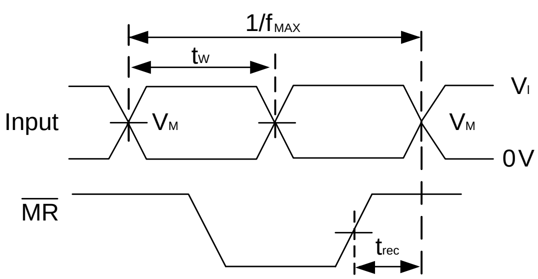
Symbol / Parameter	Pins	Test Conditions	V_{CC}	T_A = +25°C			-40°C to +85°C		-40°C to +125°C		Unit
				Min	Typ	Max	Min	Max	Min	Max	
f_{MAX} Maximum Frequency	SHCP or STCP	Figure 1	2.0V	9	30	–	4.8	–	4	–	MHz
			4.5V	30	91	–	24	–	20	–	
			6.0V	35	108	–	28	–	24	–	
t_W Pulse Width	SHCP HIGH or LOW	Figure 1	2.0V	75	17	–	95	–	110	–	ns
			4.5V	15	6	–	19	–	22	–	
			6.0V	13	5	–	16	–	19	–	
	STCP HIGH or LOW	Figure 1	2.0V	75	11	–	95	–	110	–	
			4.5V	15	4	–	19	–	22	–	
			6.0V	13	3	–	16	–	19	–	
	$\overline{MR}$ LOW	Figure 1	2.0V	75	17	–	95	–	110	–	
			4.5V	15	6	–	19	–	22	–	
			6.0V	13	5	–	16	–	19	–	
t_{SU} Set-up Time	DS to SHCP	Figure 1	2.0V	50	11	–	65	–	75	–	ns
			4.5V	10	4	–	13	–	15	–	
			6.0V	9	3	–	11	–	13	–	
	SHCP tp STCP	Figure 1	2.0V	75	22	–	95	–	110	–	ns
			4.5V	15	8	–	19	–	22	–	
			6.0V	13	7	–	16	–	19	–	
t_H Hold Time	DS to SHCP	Figure 1	2.0V	3	-6	–	3	–	3	–	ns
			4.5V	3	-2	–	3	–	3	–	
			6.0V	3	-2	–	3	–	3	–	
t_{REC} Recovery Time	$\overline{MR}$ to SHCP	Figure 1	2.0V	50	-19	–	65	–	75	–	ns
			4.5V	10	-7	–	13	–	15	–	
			6.0V	9	-6	–	11	–	13	–	
t_{PD} Propagation Delay	SHCP to Q7S	Figure 1 C_L=50pF	2.0V	–	52	160	–	200	–	240	ns
			4.5V	–	19	32	–	40	–	48	
			6.0V	–	15	27	–	34	–	41	
	STCP to Qn	Figure 1 C_L=50pF	2.0V	–	55	175	–	220	–	265	ns
			4.5V	–	20	35	–	44	–	53	
			6.0V	–	16	30	–	37	–	45	
t_{PHL} Propagation Delay	$\overline{MR}$ to Q7S	Figure 1 C_L=50pF	2.0V	–	47	175	–	220	–	265	ns
			4.5V	–	17	35	–	44	–	53	
			6.0V	–	14	30	–	37	–	45	
t_{EN} Enable Time	$\overline{OE}$ to Qn	Figure 1 C_L=50pF	2.0V	–	47	150	–	190	–	225	ns
			4.5V	–	17	30	–	38	–	45	
			6.0V	–	14	26	–	33	–	38	
t_{DIS} Disable Time	$\overline{OE}$ to Qn	Figure 1 C_L=50pF	2.0V	–	41	150	–	190	–	225	ns
			4.5V	–	15	30	–	38	–	45	
			6.0V	–	12	26	–	33	–	38	

Parameter Measurement Information

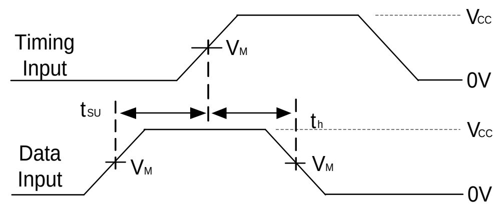


TEST	V_{LOAD}
t_{PLH}/t_{PHL}	Open
t_{PLZ}/t_{PZL}	V_{CC}
t_{PHZ}/t_{PZH}	GND

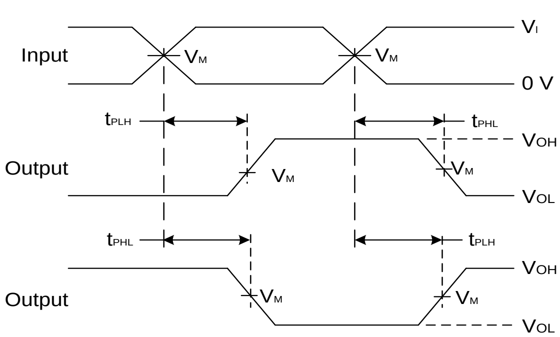
V_{CC}	Inputs		V_M	C_L
	V_I	t_r/t_f		
2.0V	V_{CC}	6ns	$V_{CC}/2$	50pF
4.5V	V_{CC}	6ns	$V_{CC}/2$	50pF
6.0V	V_{CC}	6ns	$V_{CC}/2$	50pF



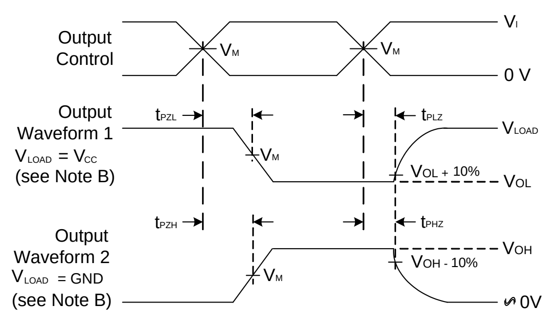
Voltage Waveform
Pulse Duration and Recovery Time



Voltage Waveform
Set-up and Hold Times



Voltage Waveform
Propagation Delay Times
Inverting and Non Inverting Outputs

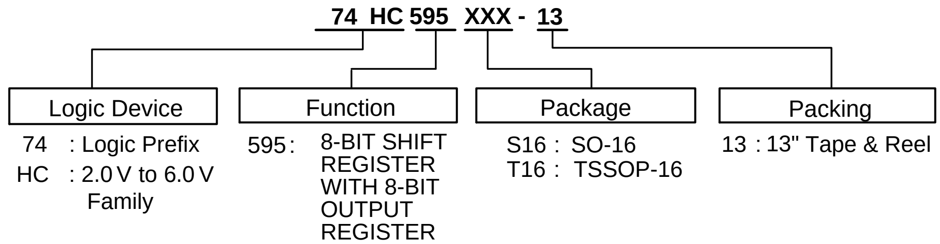


Voltage Waveform
Enable and Disable Times

Notes: A. Includes test lead and test apparatus capacitance.
B. Output Waveform 1 depends on the internal Q_N node being low and behaves in this manner based on OE pin.
Output Waveform 2 depends on the internal Q_N node being high and behaves in this manner based on OE pin.
C. All pulses are supplied at pulse repetition rate ≤ 10MHz.
D. Inputs are measured separately one transition per measurement.
E. t_{PLH} and t_{PHL} are the same as t_{PD}.

Figure 1. Load Circuit and Voltage Waveforms

Ordering Information

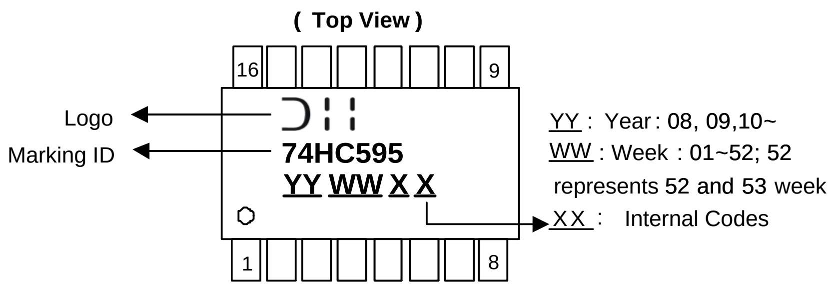


Part Number	Package Code	Packaging	7" Tape and Reel (Note 6)	
			Quantity	**Part Number Suffix**
74HC595S16-13	S16	SO-16	2500/Tape & Reel	-13
74HC595T16-13	T16	TSSOP-16	2500/Tape & Reel	-13

Note: 6. The taping orientation is located on our website at http://www.diodes.com/datasheets/ap02007.pdf.

Marking Information

(1) SO-16, TSSOP16

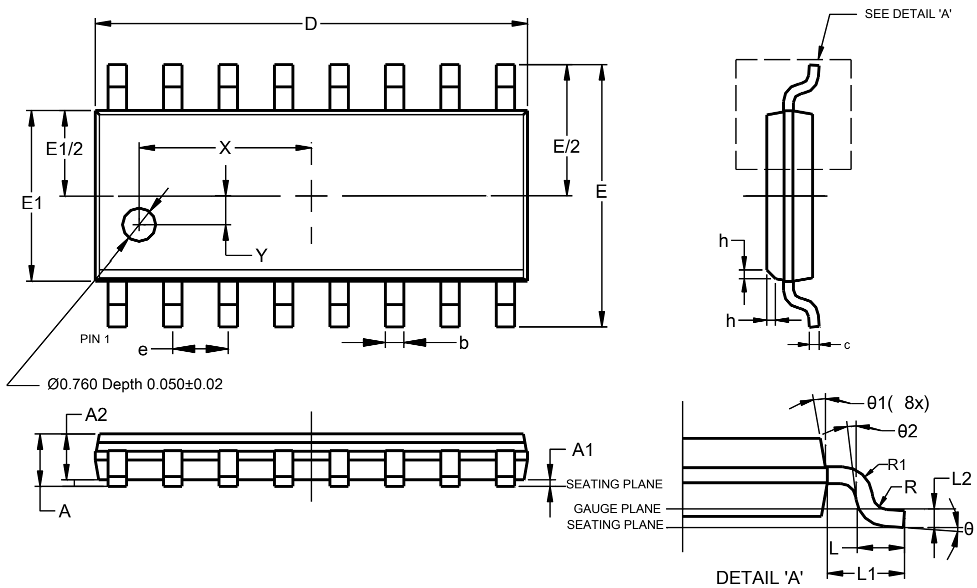


Part Number	Package
74HC595S16	SO-16
74HC595T16	TSSOP-16

Package Outline Dimensions (All dimensions in mm.)

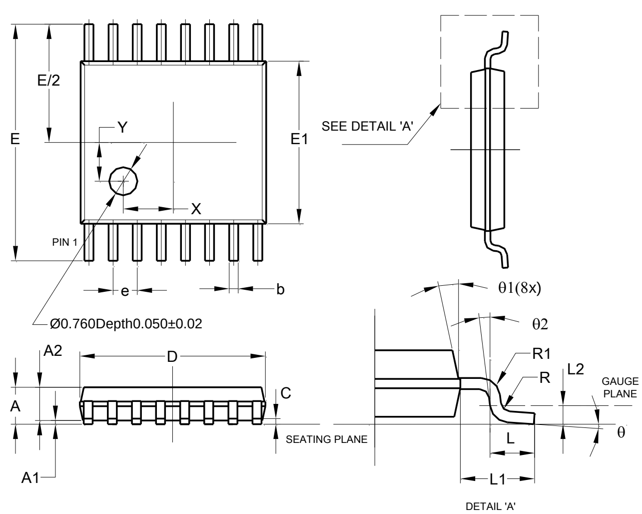
Please see http://www.diodes.com/package-outlines.html for the latest version.

Package Type: SO-16



SO-16			
Dim	**Min**	**Max**	**Typ**
A	--	1.260	--
A1	0.10	0.23	--
A2	1.02	--	--
b	0.31	0.51	--
c	0.10	0.25	--
D	9.80	10.00	--
E	5.90	6.10	--
E1	3.80	4.00	--
e	1.27 BSC		
h	0.15	0.25	0.20
L	0.40	1.27	--
L1	1.04 REF		
L2	0.25 BSC		
R	0.07	--	--
R1	0.07	--	--
X	3.945 REF		
Y	0.661 REF		
θ	0°	8°	--
θ1	5°	15°	--
θ2	0°	--	--
All Dimensions in mm			

Package Type: TSSOP-16

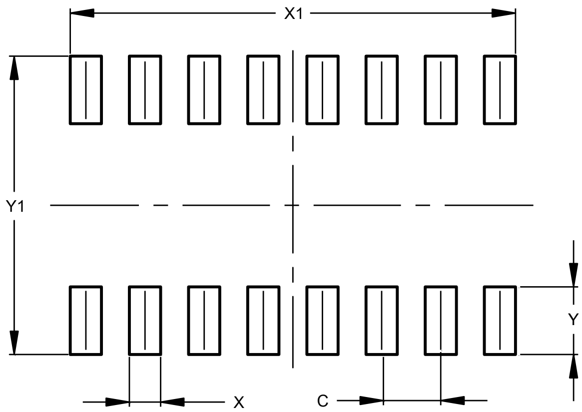


TSSOP-16			
Dim	**Min**	**Max**	**Typ**
A	-	1.08	-
A1	0.05	0.15	-
A2	0.80	0.93	-
b	0.19	0.30	-
c	0.09	0.20	-
D	4.90	5.10	-
E	6.40 BSC		
E1	4.30	4.50	-
e	0.65 BSC		
L	0.45	0.75	-
L1	1.00 REF		
L2	0.25 BSC		
R / R1	0.09	-	-
X	-	-	1.350
Y	-	-	1.050
θ	0°	8°	-
θ1	5°	15°	-
θ2	0°	-	-
All Dimensions in mm			

Suggested Pad Layout

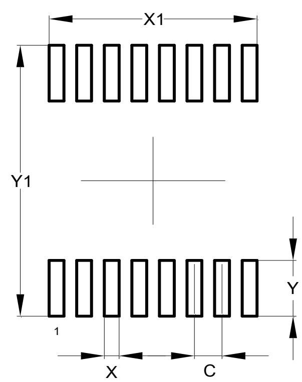
Please see http://www.diodes.com/package-outlines.html for the latest version.

Package Type: SO-16



Dimensions	Value (in mm)
C	1.270
X	0.670
X1	9.560
Y	1.450
Y1	6.400

Package Type: TSSOP-16



Dimensions	Value (in mm)
C	0.650
X	0.350
X1	4.900
Y	1.400
Y1	6.800

IMPORTANT NOTICE

DIODES INCORPORATED MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARDS TO THIS DOCUMENT, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION).

Diodes Incorporated and its subsidiaries reserve the right to make modifications, enhancements, improvements, corrections or other changes without further notice to this document and any product described herein. Diodes Incorporated does not assume any liability arising out of the application or use of this document or any product described herein; neither does Diodes Incorporated convey any license under its patent or trademark rights, nor the rights of others. Any Customer or user of this document or products described herein in such applications shall assume all risks of such use and will agree to hold Diodes Incorporated and all the companies whose products are represented on Diodes Incorporated website, harmless against all damages.

Diodes Incorporated does not warrant or accept any liability whatsoever in respect of any products purchased through unauthorized sales channel.
Should Customers purchase or use Diodes Incorporated products for any unintended or unauthorized application, Customers shall indemnify and hold Diodes Incorporated and its representatives harmless against all claims, damages, expenses, and attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized application.

Products described herein may be covered by one or more United States, international or foreign patents pending. Product names and markings noted herein may also be covered by one or more United States, international or foreign trademarks.

This document is written in English but may be translated into multiple languages for reference. Only the English version of this document is the final and determinative format released by Diodes Incorporated.

LIFE SUPPORT

Diodes Incorporated products are specifically not authorized for use as critical components in life support devices or systems without the express written approval of the Chief Executive Officer of Diodes Incorporated. As used herein:

A. Life support devices or systems are devices or systems which:

1. are intended to implant into the body, or
2. support or sustain life and whose failure to perform when properly used in accordance with instructions for use provided in the labeling can be reasonably expected to result in significant injury to the user.

B. A critical component is any component in a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or to affect its safety or effectiveness.

Customers represent that they have all necessary expertise in the safety and regulatory ramifications of their life support devices or systems, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of Diodes Incorporated products in such safety-critical, life support devices or systems, notwithstanding any devices- or systems-related information or support that may be provided by Diodes Incorporated. Further, Customers must fully indemnify Diodes Incorporated and its representatives against any damages arising out of the use of Diodes Incorporated products in such safety-critical, life support devices or systems.

www.diodes.com

7-Segment LED Controller Datasheet LED7SEG V 1.20

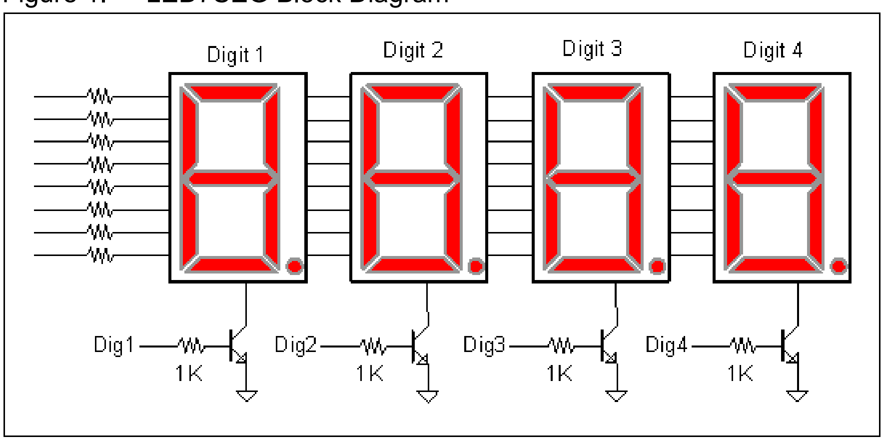
Resources	PSoC® Blocks			API Memory (Bytes)		Pins (per External I/O)
	Digital	Analog CT	Analog SC	Flash	RAM	
CY8C29/27/24/22/21xxx, CY8C23x33, CY8C28xxx, CY8CLED02/04/08/16, CY7C64215, CY8CPLC20, CY8CLED16P01, CYWUSB6953	1[a]	0	0	311	9	1

a. Only requires a digital block if the timer option is selected.

Features and Overview

- Supports 1 to 8 Digits
- Any combination of individual displays up to 8 total digits
- Displays both hex and integer values
- Supports decimal points built into 7-Segment display
- Supports both common cathode and common anode displays
- Configurable for both Active High and Active Low segment and digit drives

Figure 1. LED7SEG Block Diagram

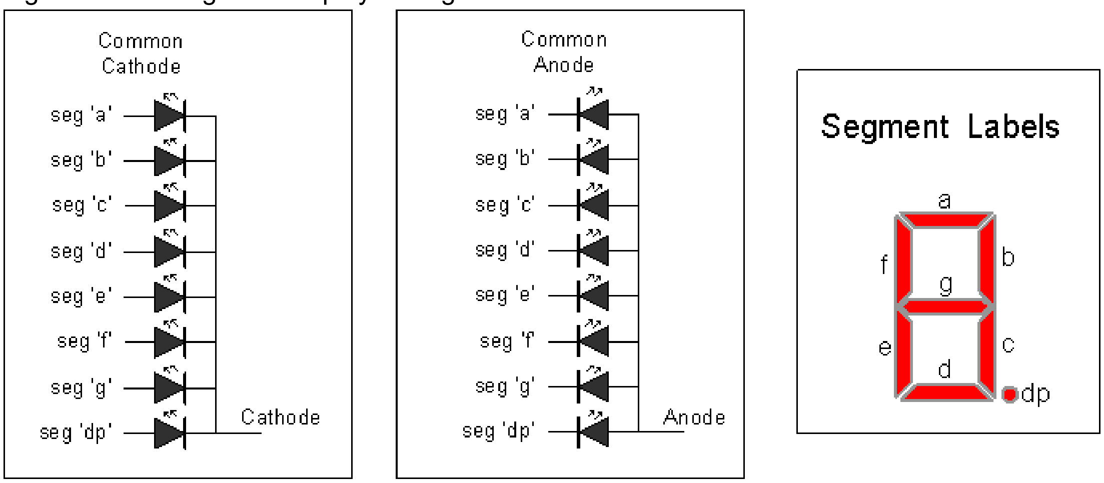


Functional Description

The LED7SEG User Module is capable of multiplexing up to eight 7-segment displays. This user module is compatible with common cathode, common anode, or any drive polarity. This allows a wide range of flexibility with various displays. Digits and segments may be driven directly by PSoC pins without the use of transistors or drivers as long as the current sinking and sourcing limits of the PSoC pins are not exceeded.

The following diagram shows how common anode and cathode 7-segment displays are configured:

Figure 2. 7-Segment Display Configurations



The project using the LED7SEG User Module that manipulates pins on a port shared with an instance of the LED7SEG must avoid direct PRTxDR writes. Use Shadow Registers for such manipulation to prevent incorrect LED7SEG operation.

How Multiplexing Works

When 2 or more displays are multiplexed, only one digit is on at a time, although to the eye it appears that all digits are on continuously. To achieve this illusion, the rate at which the displays are turned on and off must be higher than the response of the human eye. For a four digit display, the multiplex rate should be near 1 kHz; for an eight digit display the multiplex rate should be double that, or 2 kHz. When multiplexing an N-digit display, each digit is bon for 1/N of the total time. For example, for a system with 8 digits and a refresh rate of 2 kHz (500 uS period), each display is on for 500 uSec every 4 mSec (8 digits * 500 uSec/digits).

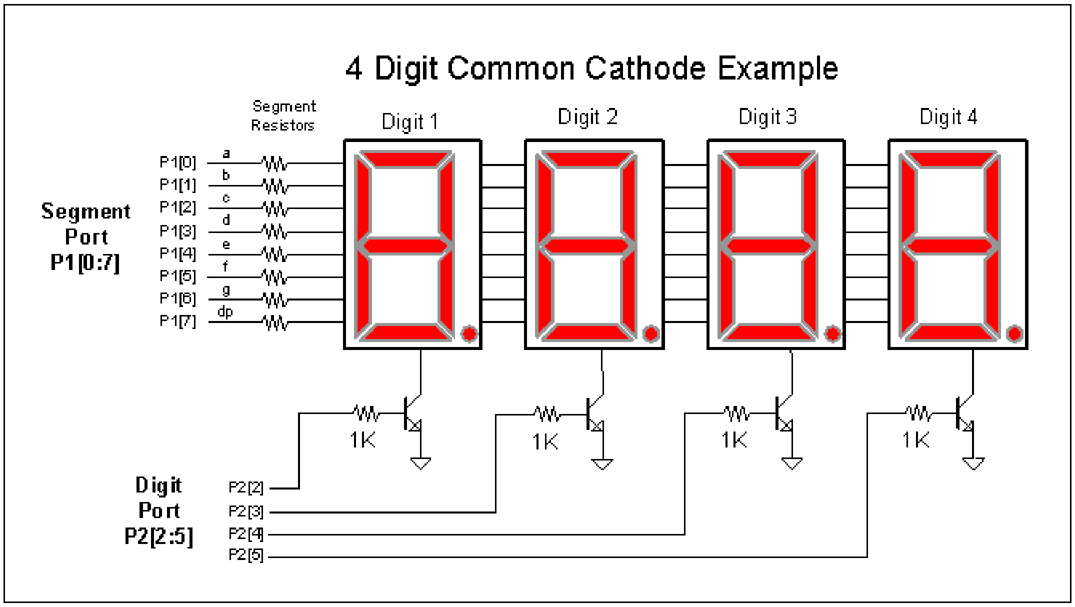
Why Multiplex?

The main reason to multiplex a display is to save I/O pins. A multiplexed eight digit display requires only 16 pins. If the same display were driven directly, 64 I/O pins would be required. On the other hand, multiplexing does require some amount of CPU overhead. In most systems the overhead is small enough not to be of concern. With this user module, if the CPU speed is set to 12 MHz, the overhead is less than 2% with a multiplex rate of 1 kHz.

The following diagram shows an example of a four digit display using this user module. The digit numbering always starts from left (MSB) to right (LSB). Resistors must be used on the segment signals to limit the LED current and the PSoC sinking or sourcing current per pin. The segment resistors are connected between the PSoC port pins and all similar segments for all digits. For example, the segment “a" signals for all displays are connected in parallel, the segment “b" signals are connected in parallel, and so on. A full 8-bit port is required for the port used to drive the display segments (Segment port). In this example, Port 1 is used. The port used to drive the common anode or cathode (Digit Port) only requires as many pins as there are digits, but the pins must be consecutive.

In this example port pins P2[2], P2[3], P2[4], and P2[5] are used. Port pins P2[0], P2[1], P2[2], and P[3] could have equally been used as well as several other combinations.

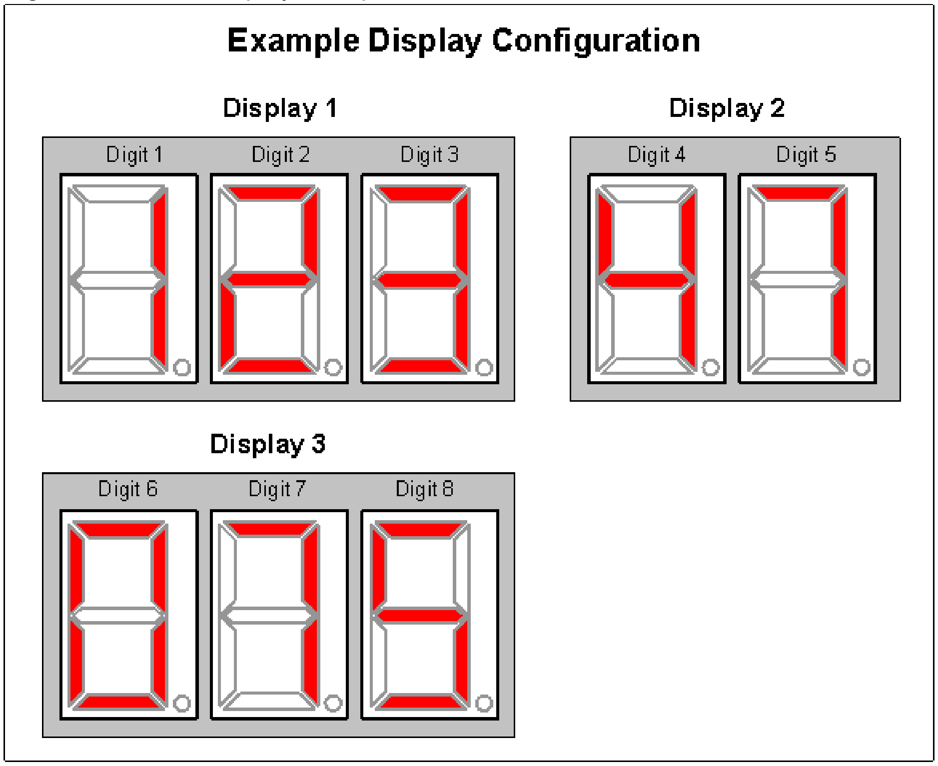
Figure 3. Example 4-Digit Display using LED7SEG User Module



Although the 1 to 8 digit display may look like one contiguous display, it may be arranged in one single eight digit display or eight 1 digit displays and everything in between. For example, you could configure 8 digits into two 3-digit displays and one 2-digit display, or maybe four 2-digit displays. The API can handle any combination as long as the groups of digits are contiguous.

In the following figure, eight digits have been arranged into three display groups. Display 1 contains Digits 1,2, and 3. Display 2 contains digits 4 and 5, and Display 3 contains digits 6, 7, and 8. Notice that although the 8 digits have been arranged in 3 groups, the digit numbers are still consecutive.

Figure 4. LED Display Groups



The following is an example of a code snippet that would work with the configuration defined above:

```
iNum1 = 123;
iNum2 = 47
iNum3 = 75;
LED7SEG_DispInt( iNum1, 1, 3);    // Start at Digit 1, three digits long
LED7SEG_DispInt( iNum2, 4, 2);    // Start at Digit 4, two digits long
LED7SEG_DispInt( iNum3, 6, 3);    // Start at Digit 6, three digits long.
```

DC and AC Electrical Characteristics

See the device datasheet.

Placement

Before placing the LED7SEG User Module, the user must choose between a built in multiplex timer and a user generated multiplex timer. If the build in multiplex timer is selected, a digital block is added to the project for the sole purpose of display multiplexing. This is the simplest option if an extra digital block is available. An extra user parameter “Multiplex Rate" is present, when this option is selected.

If an extra digital block is not available, the user may use any user module’s interrupt service routine that is periodic and close to 1 kHz or more. User modules that may have periodic interrupt service routines are counters, timers, PWMs, and some ADCs. The LED7SEG_Update() function call should be placed in any user module’s interrupt service routine. Unlike most functions, the LED7SEG_Update() function preserves both A and X registers.

Parameters and Resources

Segment Port

This parameter is used to select the GPIO port that drives the display segments (refer to Figure 3). All eight pins of the selected GPIO port are used by this user module. Any GPIO port that has all eight pins available is shown as an option to this parameter.

Digit Port

This parameter selects the GPIO port used for driving the "common" signals of the 7-segment display. Each digit of the display has one common signal that controls whether that digit is on or off. To see how the common signals connect to the 7-segment display hardware, refer to Figure 3. Whenever one of the common signals is in an active state, the cathodes of the LEDs that make up the corresponding digit are connected to ground. Then, if any of the segment signals are driven, the corresponding LEDs will light up. However, if the common signal for a digit is not active, the cathodes of the corresponding digit's LEDs are not connected to ground, and therefore the LEDs for that digit cannot be lit up. So, the common signal for a digit serves as a global on/off signal for the entire digit.

The pins that drive the common signals must all be part of the same GPIO port so that the code for this user module can be simple and efficient.

The selection for this parameter only chooses which GPIO port is to be used for the common signals. The Digit Drive Pins parameter chooses which specific pins within the chosen GPIO port are used.

Digit Drive Pins

This parameter is used to choose the number of digits that make up the 7-segment display and the specific pins that drive the "common" signals that correspond to each digit. Refer to Figure 3 to see how the common signals connect to the 7-segment display hardware.

For example, assume that a 7-segment display with three digits is being used and assume that the desired pins for the common signals are P2[2], P2[3], and P2[4]. In this case, "Port_2" must be chosen for the Digit Port parameter and "3 Digits Px[2:4]" must be chosen for the Digit Drive Pins parameter.

The 'x' character in the text for this parameter's options is a placeholder for the port number that was chosen with the Digit Port parameter. The "[x:y]" text in each parameter option specifies which pins of the port are to be used. The first digit is the first pin and the last digit is the last pin. So, "[2:4]" means GPIO pins 2, 3, and 4 of the GPIO port are to be used. The common signal pins must always be consecutive on the GPIO port. The first common pin (P2[2] in the example) must be connected to the first digit, the second common pin (P2[3] in the example) must be connected to the second digit, and so on.

Digit Drive Polarity

This parameter selects what polarity is required to turn on a single digit.

Option	Description
Active Low	Signal at pin driven low to turn on digit.
Active High	Signal at pin is driven high to turn on digit.

Segment Drive Polarity

This parameter selects what polarity is required to turn on the display's segments.

Option	Description
Active Low	Signal at pin driven low to turn on segment.
Active High	Signal at pin is driven high to turn on segment.

Multiplex Rate (Only valid if Multiplex timer option selected)

This function selects the display refresh rate. The table below shows the valid options for this parameter. The build in timer uses the 32kHz clock so that it would be independent of VC!, VC2, and VC3 clock. Larger displays should use the fastest multiplex rates. Smaller displays (2 to 4 digits) may use the lower rates. If the display seems to flicker, increase the multiplex rate.

Option	Description
500 Hz	Multiplex at a 500 Hz rate (2 to 3 digits)
1 kHz	Multiplex at a 1 kHz rate (3 to 5 digits)
2 kHz	Multiplex at a 2 kHz rate (4 to 6 digits)
4 kHz	Multiplex at a 4 kHz rate (6 to 8 digits)

Application Programming Interface

The Application Programming Interface (API) routines are provided as part of the user module to allow the designer to deal with the module at a higher level.

Though some user module API function may leave A and X unchanged, there is no guarantee they may do so in the future.

For Large Memory Model devices, it is also the caller's responsibility to perserve any value in the CUR_PP, IDX_PP, MVR_PP, and MVW_PP registers. Even though some of these registers may not be modified now, there is no guarantee that will remain the case in future releases.

Each time a user module is placed, it is assigned an instance name. By default, PSoC Designer assigns the LED7SEG_1 to the first instance of this user module in a given project. It can be changed to any unique value that follows the syntactic rules for identifiers. The assigned instance name becomes the prefix of every global function name, variable and constant symbol. In the following descriptions the instance name has been shortened to LED7SEG for simplicity.

The following constants are defined in the user module header:

Constant	Value
LED7SEG_Digit1	0x01
LED7SEG_Digit2	0x02
LED7SEG_Digit3	0x04
LED7SEG_Digit4	0x08
LED7SEG_Digit5	0x10
LED7SEG_Digit6	0x20
LED7SEG_Digit7	0x40
LED7SEG_Digit8	0x80
LED7SEG_DimOn	0x01
LED7SEG_DimOff	0x00
LED7SEG_DpOn	0x01
LED7SEG_DpOff	0x00

LED7SEG_Start

Description:

Clears all digit memory and enables scanning. If the multiplex timer is selected, it also enables the timer and enable its interrupt. Global interrupts need to be enabled for the multiplexing to work.

C Prototype:

```
void  LED7SEG_Start(void)
```

Assembler:

```
lcall  LED7SEG_Start
```

Parameters:

None

Return Value:

None

Side Effects:

None

LED7SEG_Stop

Description:

Stops display scan and turns off all digits. If the multiplexing timer is selected, this function also disables the timer and its interrupt.

C Prototype:

```
void  LED7SEG_Stop(void)
```

Assembler:

```
lcall  LED7SEG_Stop
```

Parameters:

None.

Return Values:

None

Side Effects:

The update ISR no longer scans the display.

LED7SEG_PutHex

Description:

Places a single hex digit (0 through F) at the given digit.

C Prototype:

```
void  LED7SEG_PutHex(BYTE bValue, BYTE bDigit)
```

Assembler:

```
mov  a,0x5     ; Write a '5' to the display
mov  x,0x01    ; Write the '5' at digit 1

lcall  LED7SEG_PutHex
```

Parameters:

bValue: Hex value between 0 and F to display.

bDigit: Digit location to place the given bValue.

Return Value:

None

Side Effects:

The A and X registers may be altered by this function.

LED7SEG_PutPattern

Description:

Writes a byte pattern directly to the display memory. A high bit corresponds to an active segment no matter if the display is common anode or cathode. The LSB is segment ‘a’ and the MSB is the “dp", decimal point.

C Prototype:

```
void  LED7SEG_PutPattern(BYTE bPattern, BYTE bDigit)
```

Assembler:

```
mov  a,0xFF   ; Turn all segments on
mov  x,0x01

lcall  LED7SEG_PutPattern
```

Parameters:

bPattern: A bit pattern to display. For example, 0x80 activates only the decimal point and 0x7F activates all segments other than the decimal point.

bDigit: Digit location to place the given bValue.

Return Value:

None

LED7SEG_DP

Description:

Turn the decimal point on or off at the given location.

C Prototype:

```
void  LED7SEG_DP(BYTE bOnOff, BYTE bDigit)
```

Assembler:

```
mov  a,0x1
mov  x,0x01
lcall  LED7SEG_DP
```

Parameters:

bOnOff: 0 => Turn off decimal point, 1 => turn on decimal point.

bDigit: Digit location for decimal point.

Return Value:

None

Side Effects:

The A and X registers may be altered by this function.

LED7SEG_Dim

Description:

Dim or undim the display. When dim is enabled, the update skips every other refresh cycle.

C Prototype:

```
void  LED7SEG_Dim(BYTE bDim)
```

Assembler:

```
mov   x,0x01         ; Dim the display
lcall  LED7SEG_Dim
```

Parameters:

bDim: 0 => Do not dim display, 1 => Dim display

Return Value:

None

Side Effects:

The A and X registers may be altered by this function.

LED7SEG_DispInt

Description:

Display an integer value on the display. It can be between 1 and 5 digits in length and start at any location. Range of numbers to display, 0 to 65535.

C Prototype:

```
void  LED7SEG_DispInt(int iValue, BYTE bPos, BYTE bLSD)
```

Assembler:

```
mov  X, SP
add   SP, 4
mov   [X], 0x34      ; Load MSB of value to display
mov   [X+1], 0x12  ; Load LSB of value to display
mov   [X+2], 0        ; Set starting location for value in display
mov   [X+3], 4        ; Set number of digits to display
lcall  LED7SEG_DispInt
```

Parameters:

iValue: Integer value to be displayed.

bPos: Digit position in display in which to start displaying the value.

BLSD: Length of digit to display, for example only display 4 digits.

Return Value:

None

Side Effects:

The A and X registers may be altered by this function.

LED7SEG_Update

Description:

The Update function controls the multiplexing rate of the display. Each time it is called, the previous displayed digit is turned off and the next is turned on. This function is automatically called periodically if the “LED7SEG with MPX Timer" is selected in the “User Module Selection Options..." menu. If the “LED7SEG without MPX Timer" option is chosen, the user needs to call this function periodically every 2 to 0.5 mSec to provide flicker free multiplexing. This function can be called in the users main program loop or from an existing interrupt service routine.

C Prototype:

```
void  LED7SEG_Update(void)
```

Assembler:

```
lcall  LED7SEG_Update
```

Parameters:

None

Return Values:

None

Side Effects:

All registers are preserved so there is no need to preserve A and X before calling this function.

Sample Firmware Source Code

Sample Assembly Code

```
;-------------------------------------------------------------------------
; // Example ASM code using LED7SEG User Module
;-------------------------------------------------------------------------

include "m8c.inc"       ; part specific constants and macros
include "memory.inc"    ; Constants & macros for SMM/LMM and Compiler
include "PSoCAPI.inc"   ; PSoC API definitions for all User Modules

export _main

_main:
    lcall  LED7SEG_Start

 ;; Enable multiplex timer here if using
    ;; non-built-in-timer version.

  M8C_EnableGInt         ; Enable global interrupts
     push  X
     mov   A,4              ; Push display size on stack "4"
     push  A
     mov   A,1              ; Push display start on stack "1"
     push  A
     mov   A,>1234          ; Push MSB of value 1234 on stack
     push  A
     mov   A,<1234          ; Push LSB of value 1234 on stack
```

```
     push  A
     lcall LED7SEG_DispInt  ; Display "1234"
     mov   X,3              ; Turn on decimal point at digit 3
     mov   A,1              ; Enable DP
     lcall LED7SEG_DP

.terminate:
    jmp .terminate
```

Sample C Code

```
//--------------------------------------------------------
// Example C code using LED7SEG User Module
//--------------------------------------------------------

#include <m8c.h>
#include "PSoCAPI.h"

void main(void)
{
   LED7SEG_Start();             // Enable display
   M8C_EnableGInt;              // Enable IRQ

// Enable multiplex timer here if using
   // non-built-in-timer version.

M8C_EnableGInt;
LED7SEG_DispInt(1234, 1, 4); // Display "1234"
   LED7SEG_DP(1, 3);            // Turn on DP ( 123.4 )
}
```

Configuration Registers

None

Version History

Version	Originator	Description
1.1	DHA	Added Version History.
1.20	DHA	Added support for CY8C21x12 devices.

Note PSoC Designer 5.1 introduces a Version History in all user module datasheets. This section documents high level descriptions of the differences between the current and previous user module versions.