



**UNIVERSIDAD NACIONAL  
AUTÓNOMA DE MÉXICO**

**FACULTAD DE INGENIERÍA**

**CIRCUITO DE ILUMINACIÓN ACOPLADO A  
UN SISTEMA RECONOCEDOR DE VOZ.**

**TESINA  
QUE PARA OBTENER EL TÍTULO DE  
INGENIERO ELÉCTRICO ELECTRÓNICO  
PRESENTA  
ARTURO GONZÁLEZ URQUIZA**

**DIRECTOR DE TESIS  
DR. JOSÉ ABEL HERRERA CAMACHO**



**MÉXICO, D.F. 2009**

***Agradecimientos.***

*A mis padres, a mis hermanos, a mi tía y a mis amigos.*

*Gracias.*

*Arturo González Urquiza.*

# **Circuito de iluminación acoplado a un sistema reconocedor de voz.**

## **Índice.**

<b>Capítulo 1 - Introducción al proyecto realizado.</b>	<b>1</b>
1 – 1 Justificación del proyecto.	2
1 – 2 Objetivo y estructura de esta tesina.	2
<b>Capítulo 2 –</b>	
<b>Método utilizado para el procesamiento de las señales de voz.</b>	<b>3</b>
2 – 1 Sonido.	4
2 – 2 La voz humana.	5
2 – 3 Sistema de reconocimiento de voz de palabras aisladas y dependiente del locutor utilizando Multisection Bookcode.	7
<b>Capítulo 3 - Hardware para la implementación del sistema de reconocimiento de voz.</b>	<b>12</b>
3 – 1 Procesador Digital de Señales.	13
3 – 2 Hardware utilizado.	15
<b>Capítulo 4 - Descripción de los elementos utilizados en el proyecto.</b>	<b>18</b>
4 – 1 Filtro Paso Altas.	19
4 – 2 Comparadores.	21
4 – 3 Microcontrolador.	26
4 – 4 Programa.	40
4 – 5 Tiristores.	46

4 – 6 Optoaisladores.	57
4 – 7 Esquemático Final.	58
<b>Capítulo 5 – Conclusiones.</b>	<b>60</b>
<b>Bibliografía.</b>	<b>62</b>

# Capítulo 1

## Introducción al proyecto realizado

Se tiene programado un sistema reconocedor de voz en un TMS320C6711 DSP Starter Kit de Texas Instruments, dicha tarjeta carece de puertos de salida de datos digitales, pero sí tiene un puerto para salida de datos analógicos.

El sistema reconocedor de voz es capaz de identificar doce comandos de voz, los cuales son frases de dos palabras necesarias para prender o apagar la luz en los cuartos de una casa habitación.

Por el puerto de salida analógico de la tarjeta ya mencionada se saca una señal sinusoidal de una duración de un segundo, con una frecuencia asignada de acuerdo al comando de voz utilizado. Además se ha asignado una frecuencia adicional al caso en que la señal no haya sido reconocida. Dicha señal posee una amplitud de 2 Vpp y un offset de 3 V. En la tabla 1.1 se muestran los doce comandos de voz que el sistema es capaz de reconocer y la frecuencia de la señal sinusoidal asociada a cada comando, además de la frecuencia asociada a las señales desconocidas.

Comando.	Frecuencia de la señal sinusoidal [Hz].
Apaga baño.	300
Apaga cocina.	500
Apaga comedor.	700
Apaga jardín.	900
Apaga recamara.	1100
Apaga sala.	1300
Enciende baño.	1500
Enciende cocina.	1700
Enciende comedor.	1900
Enciende jardín.	2100
Enciende recamara.	2300
Enciende sala.	2500
Señal desconocida.	2700

**Tabla 1.1**

Comandos de voz y la frecuencia de la señal sinusoidal asociada a ellos.

### **1 – 1 Justificación del proyecto.**

Los usos y aplicaciones de los sistemas de reconocimiento de voz son múltiples, pero las razones por las cuales se planteo la realización de este proyecto son las siguientes:

- **El control por comandos**, con lo cual se logra controlar un proceso por medio del habla, reconociendo un vocabulario muy reducido y teniendo un rendimiento elevado comparado con sistemas dedicados al reconocimiento de gran cantidad de fonemas, como el sistema de dictado automático.
- **Sistema diseñado para discapacitados**, útil para personas con problemas de movimiento les sea más fácil realizar un proceso, en este caso el prendido y apagado de las luces en una casa habitación.

### **1 – 2 Objetivo y estructura de esta tesina.**

En esta parte del proyecto se requiere diseñar un circuito electrónico capaz de reconocer a que comando corresponde la señal sinusoidal, y además prender o apagar un foco correspondiente al comando reconocido por el DSP, los seis focos estarán alimentados con la línea, es decir, 127 Vrms de corriente alterna.

En el segundo capítulo se explica lo que es el sonido, se estudian las características de la voz, y se da una breve descripción del proceso de reconocimiento de voz usado para reconocer los comandos de voz en este proyecto.

En el tercer capítulo se explica lo que es un DSP, se dan algunas notas relevantes relacionadas con estos, y se presentan las características del DSK TMS320C6711 de Texas Instruments, que es el que se usó para implementar el sistema reconocedor de voz del capítulo 2.

En el cuarto capítulo, que es la parte central de este trabajo, se describe todo el proceso de reconocimiento de la señal sinusoidal ya comentada, se describe a detalle el acondicionamiento de dicha señal para que un microcontrolador la pueda interpretar, se describen algunas de las características del PIC16F877A, se describe el funcionamiento del programa del microcontrolador, y además se presenta la etapa de potencia usada para prender o apagar los focos.

Finalmente se dan las conclusiones correspondientes a este proyecto.

## Capítulo 2

### **Método utilizado para el procesamiento de las señales de voz.**

En este capítulo se definen los conceptos de sonido y de voz, se presentan algunas de sus características, se muestra también un modelo digital para la voz, se describe un sistema de reconocimiento de voz de palabras aisladas y dependiente del locutor, se presentan los diagramas a bloques de las etapas de entrenamiento y reconocimiento, y se explican a grandes rasgos en qué consisten los bloques de cada una de estas etapas.

## 2 – 1 Sonido.

En física, **sonido** es cualquier fenómeno que involucre la propagación en forma de ondas elásticas audibles o casi audibles, generalmente a través de un fluido (u otro medio elástico) que este generando movimiento vibratorio de un cuerpo. El sonido humanamente audible consiste en ondas sonoras consistentes en oscilaciones de la presión del aire, que son convertidas en ondas mecánicas en el oído humano y percibidas por el cerebro. La propagación del sonido es similar en los fluidos, donde el sonido toma la forma de fluctuaciones de presión. En los cuerpos sólidos la propagación del sonido involucra variaciones del estado tensional del medio.

La propagación del sonido involucra transporte de energía sin transporte de materia, en forma de ondas mecánicas que se propagan a través de la materia sólida, líquida o gaseosa. Como las vibraciones se producen en la misma dirección en la que se propaga el sonido, se trata de una onda longitudinal.

### Características o cualidades del sonido.

Las cuatro cualidades básicas del sonido son:

- El **tono**: viene determinado por la frecuencia fundamental de las ondas sonoras (es lo que permite distinguir entre sonidos graves, agudos o medios), medida en ciclos por segundo o hercios (Hz).

Para que los humanos podamos percibir un sonido, éste debe estar en el rango de audición, de 20 y 20,000 Hz. Por debajo de este rango tenemos los infrasonidos y por encima los ultrasonidos. A esto se le denomina rango de *frecuencia audible*. Cuanta más edad se tiene, este rango va reduciéndose tanto en graves como en agudos.

- La **intensidad**: es la cantidad de energía acústica que contiene un sonido. La intensidad viene determinada por la potencia, que a su vez está determinada por la amplitud y nos permite distinguir si el sonido es fuerte o débil.

Los sonidos que percibimos deben superar el umbral auditivo (0 dB) y no llegar al umbral de dolor (140 dB). Esta cualidad la medimos con el sonómetro y los resultados se expresan en decibelios (dB).

- El **timbre**: es la cualidad que confiere al sonido los armónicos que acompañan a la frecuencia fundamental.

Esta cualidad es la que permite distinguir dos sonidos, por ejemplo, entre la misma nota (tono), con igual intensidad, producida por dos instrumentos musicales distintos.

- La **duración**: es la cualidad que determina el tiempo de vibración de un objeto. Por ejemplo, podemos escuchar sonidos largos, cortos, muy cortos, etc.

## 2 – 2 La voz humana.

La voz humana se produce por la vibración de las cuerdas vocales, lo cual genera una onda sonora que es combinación de varias frecuencias y sus correspondientes armónicos. La cavidad buco-nasal sirve para crear ondas cuasiestacionarias, por lo que ciertas frecuencias son denominadas formantes. Cada segmento de sonido del habla viene caracterizado por un cierto espectro de frecuencias o distribución de la energía sonora en las diferentes frecuencias. El oído humano es capaz de identificar diferentes formantes de dicho sonido y percibir cada sonido con formantes diferentes como cualitativamente diferentes, eso es lo que permite por ejemplo distinguir dos vocales. Típicamente el primer formante, el de frecuencia más baja está relacionado con la abertura de la vocal que, en última instancia, está relacionada con la frecuencia de las ondas estacionarias que vibran verticalmente en la cavidad. El segundo formante está relacionado con la vibración en la dirección horizontal y está relacionado con si la vocal es anterior, central o posterior.

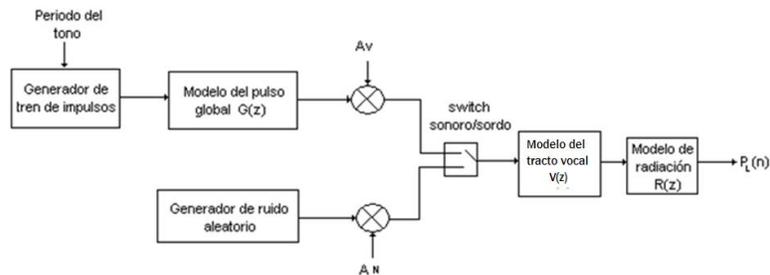
La voz masculina tiene un tono fundamental de entre 100 y 200 Hz, mientras que la voz femenina es más aguda, típicamente está entre 150 y 300 Hz. Las voces infantiles son aún más agudas. Sin el filtrado por resonancia, que produce la cavidad buco nasal, nuestras emisiones sonoras no tendrían la claridad necesaria para ser audibles. Ese proceso de filtrado es precisamente lo que permite generar los diversos formantes de cada unidad segmental del habla.

### El modelo digital para la voz.

Un modelo de la voz que se considere completo debe incluir: los cambios en la señal de excitación, la respuesta del tracto vocal y los efectos de los labios en la radiación. Tal modelo es el *fuentes-filtro* que ha sido usado por casi todos los sistemas de procesamiento de voz.

En la figura 2.1, la excitación glotal<sup>1</sup> es la entrada del filtro. Los cinco tipos de excitación son reducidos, de manera general, a dos: *señales periódicas* (sonoras) y *ruido turbulento* (sonidos sordos) con distribución gamma o laplaciana y espectro plano. El espectro glotal es un tren de impulsos espaciados a frecuencias iguales a la frecuencia del tono fundamental. El efecto es, aproximadamente, una caída de 12 dB/octava alrededor de 0.8 a 1 kHz.

<sup>1</sup> La glotis es la porción más estrecha de la luz laríngea, espacio que está limitado por las cuerdas vocales, la porción vocal de los aritenoides y el área interaritenoides. En el ser humano, la laringe tiene diversas funciones interviniendo en la fonación, la deglución y como mecanismo de defensa. Cuando las cuerdas vocales vibran, el sonido resultante tiene una cualidad, llamada voz o sonoridad. Como contraste, cuando no vibran se dice que el sonido es sordo. Los sonidos en cuya producción interviene sólo la glotis se llaman *glotales*.



**Figura 2.1**

Modelo digital para la voz.

El tracto vocal puede ser modelado aproximadamente por la función de transferencia:

$$H(z) = \frac{G}{1 - \sum_{i=1}^N \alpha_i z^{-i}}$$

Donde  $G$  representa el factor de ganancia total y  $\alpha_i$  las ubicaciones de los polos. Los polos de  $H(z)$  corresponden a las resonancias o formantes de la voz.

La radiación de la voz tiene la propiedad que a bajas frecuencias la presión del sonido es proporcional a la derivada de la velocidad volumétrica. Esto introduce un levantamiento de 6 dB/octava en el espectro que puede ser modelado por

$$R(z) = 1 - z^{-1}$$

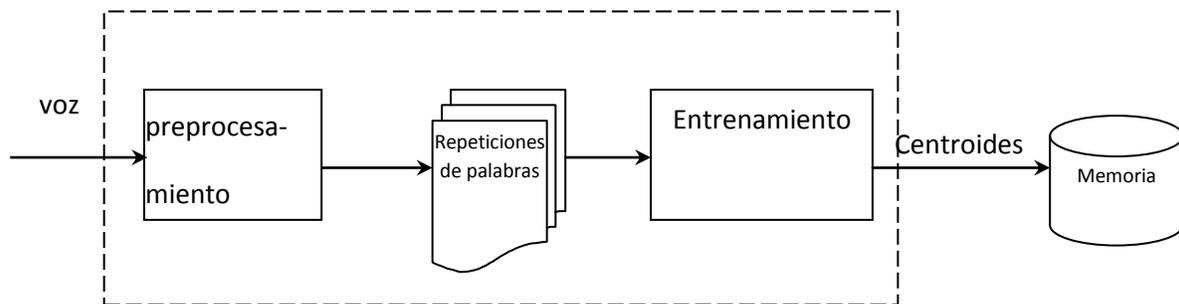
Los tres efectos pueden ser representados por una sola función de transferencia todo-polos llamada espectro de envolvente. En su forma refleja la información principal de la señal de voz y casi todos los sistemas de voz tratan de generarla o recuperarla.

## 2 – 3 Sistema de reconocimiento de voz de palabras aisladas y dependiente del locutor utilizando Multisection Bookcode.

### Descripción general.

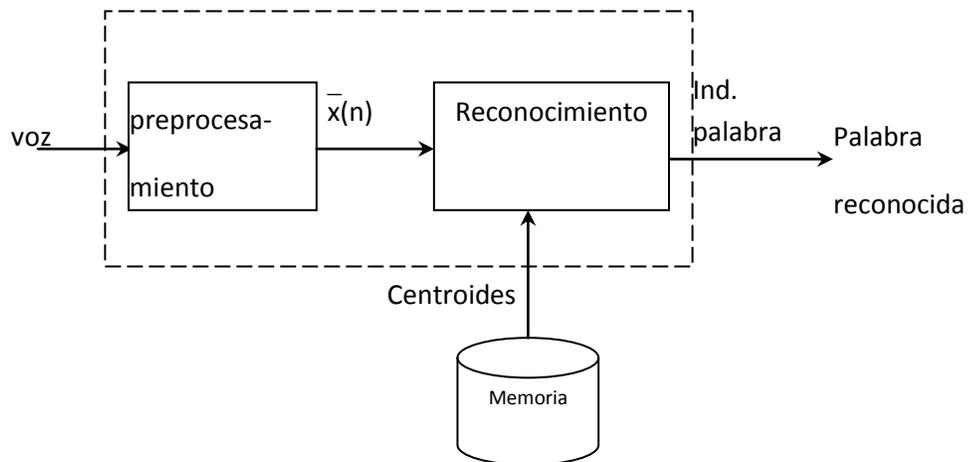
El sistema para el reconocimiento de palabras aisladas utiliza las técnicas del procesamiento de voz más comunes.

Los diagramas de bloques que se muestran en las figuras 2.2 y 2.3 ilustran, de forma general, los procesos que se realizan en el entrenamiento y reconocimiento de palabras aisladas, respectivamente.



**Figura 2.2**

Etapa de entrenamiento.



**Figura 2.3**

Etapa de reconocimiento.

Como se observa en los diagramas, existe un módulo común en ambas etapas: el preprocesamiento. Por esta razón, el preprocesamiento se manejará como un módulo independiente, para efecto del análisis.

Como se muestra en la figura 2.2, durante la etapa de entrenamiento se capturan diferentes señales de voz, a la que denominaremos *repeticiones de palabras*. A cada repetición se le aplica preprocesamiento para obtener una palabra delimitada. Posteriormente, se agrupan todas las palabras recortadas y con ellas se obtienen sus centroides<sup>2</sup> correspondientes. Estos centroides se guardan en memoria.

En la etapa de reconocimiento, como se observa en la figura 2.3, se captura la palabra que se desea reconocer. A esta palabra también se le aplica preprocesamiento para recortarla. Utilizando los centroides, calculados durante el entrenamiento y almacenados en la memoria, se realizan las comparaciones necesarias, para efectuar el reconocimiento. El éxito del evento dependerá de ciertos parámetros estadísticos obtenidos mediante el análisis de una población de resultados.

### **Preprocesamiento.**

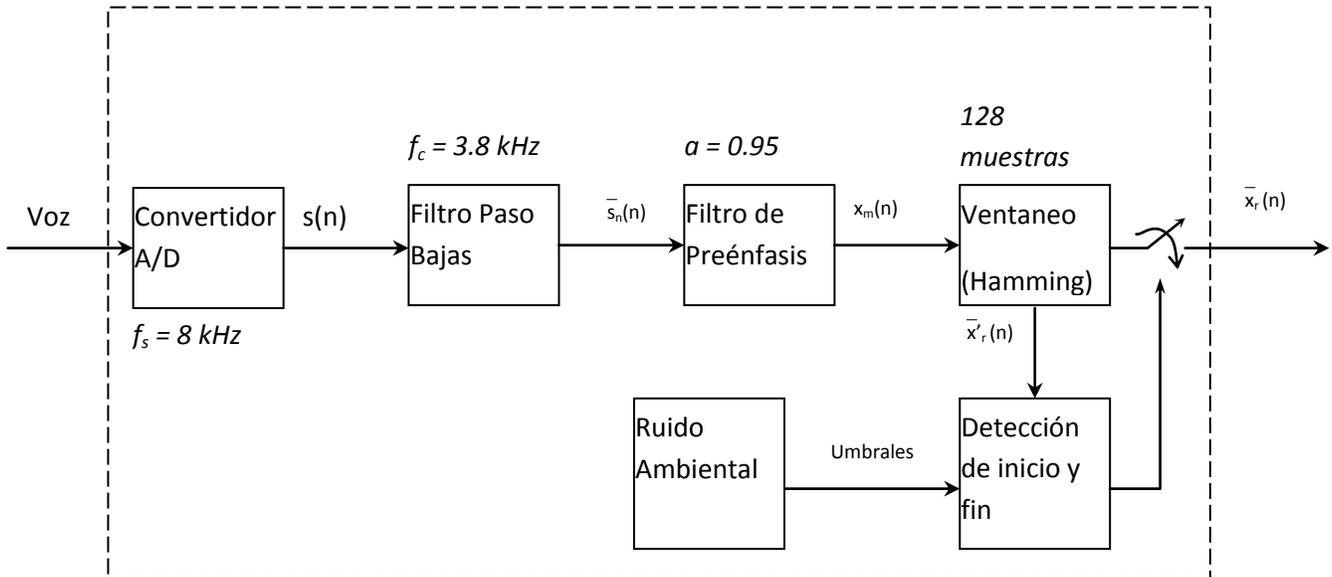
En el preprocesamiento se recibe una señal de voz y se determinan sus límites. A la señal se le aplican dos tipos de filtros: paso bajas y de preénfasis para eliminar ruidos de altas frecuencias y realzar las frecuencias altas presentes en la voz, respectivamente. Enseguida se divide la señal en tramas de 128 muestras y a cada trama se le aplica una ventana de tipo Hamming, para suavizar el espectro.

Es necesario aclarar que, cuando se inicializa el sistema, se determinan ciertos parámetros (*umbrales*) necesarios para la detección de inicio y fin de la palabra. Estos parámetros se calculan recolectando muestras del ruido presente en el ambiente circundante. El módulo encargado de realizar este proceso, se denomina ruido ambiental.

Finalmente, con las tramas provenientes del ventaneo y con los umbrales originados por el ruido ambiental, se determina el inicio y fin de cada palabra.

La figura 2.4 muestra el diagrama de bloques para este módulo.

<sup>2</sup> Los centroides son vectores que se obtienen a partir de los vectores de LPC, a los que se les aplica un algoritmo de cuantización (en este caso K-medias), para tener como resultado un menor número de vectores que representan a los vectores iniciales en el reconocimiento.



**Figura 2.4**

Diagrama de bloques para el módulo de preprocesamiento.

Como ya se mencionó, el preprocesamiento es un módulo común en el entrenamiento y en el reconocimiento. Sin embargo, existe una diferencia sutil: durante el entrenamiento, cada palabra recortada se almacena en un archivo independiente, que posteriormente se utilizará para determinar los centroides representativos. Mientras que en el reconocimiento, la palabra recortada se utiliza directamente para hacer las comparaciones con los centroides.

El preprocesamiento consta de los siguientes módulos: filtro paso bajas, filtro de preénfasis, ventaneo, detección de ruido ambiental y detección de inicio y fin de la palabra.

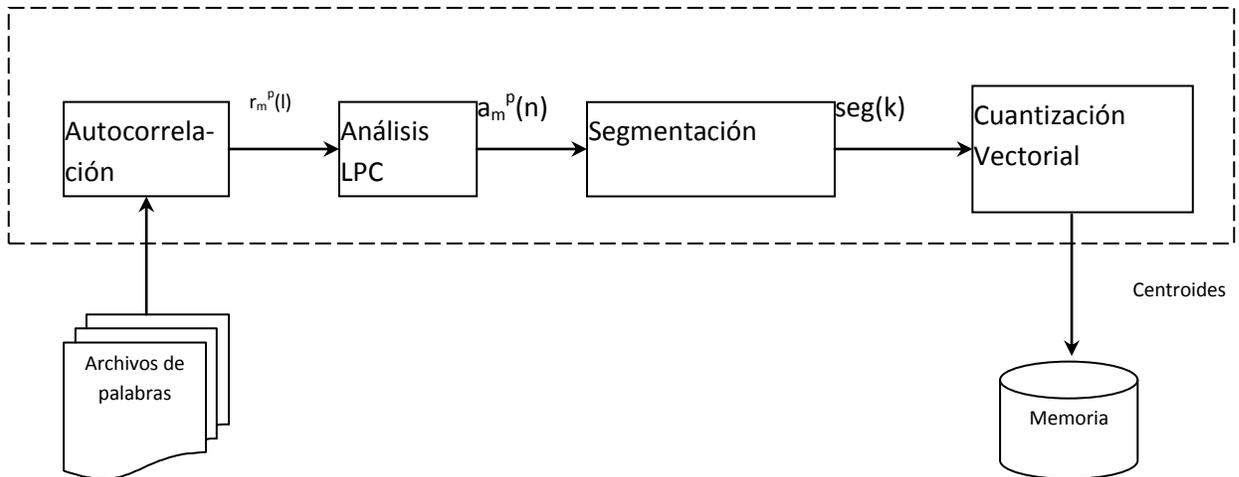
### Entrenamiento.

Para el entrenamiento, se deben capturar diversas palabras recortadas, por medio del módulo de preprocesamiento, que representan las diferentes *repeticiones*. A partir de estas repeticiones, se obtiene un conjunto de centroides que son almacenados en la memoria.

Los archivos de palabras, previamente almacenados, son utilizados para aplicarle varios procesos a cada uno. Primero se determinan sus vectores de autocorrelación para obtener los Coeficientes de Predicción Lineal (LPC). Enseguida, estos coeficientes son segmentados en los distintos fonemas.

Se sigue el mismo procedimiento para todas las repeticiones de una misma palabra.

Posteriormente, una vez que se tienen segmentadas todas las repeticiones, se agrupan en conjuntos del mismo segmento. Para cada conjunto se obtienen centroides, que representarán al segmento. Finalmente, estos centroides son almacenados en memoria. La figura 2.5 esquematiza el proceso completo del entrenamiento.



**Figura 2.5**  
Módulo de Entrenamiento.

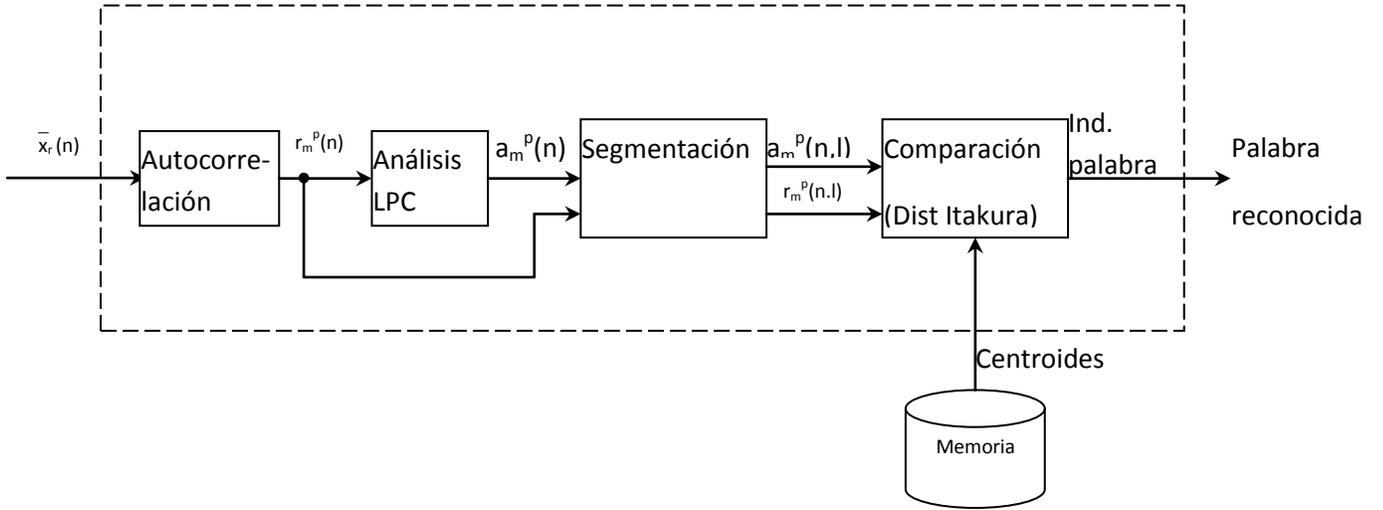
El entrenamiento es la etapa que nos permite obtener los patrones de comparación para las señales de voz. Consta de los siguientes módulos: autocorrelación, análisis LPC, segmentación y cuantización vectorial.

### Reconocimiento

El reconocimiento, recibe las tramas de una señal de voz, proveniente del preprocesamiento, para efectuar comparaciones con los centroides y obtener un indicador a la palabra reconocida.

Cada trama recibida es utilizada para calcular su autocorrelación y sus coeficientes de predicción lineal. Este proceso se repite hasta que el preprocesamiento detecta el fin de la palabra. Una vez detectado el final, se segmentan los vectores LPC y los vectores de autocorrelación correspondientes, de forma lineal y con segmentos iguales al patrón de comparación. Con cada segmento se realiza una comparación, utilizando la distancia de *Itakura*, con los centroides que correspondan al mismo segmento.

El resultado de esta comparación, aunado con ciertos parámetros estadísticos, determina el éxito o fracaso del reconocimiento. La figura 2.6 muestra el diagrama de bloques del reconocimiento.



**Figura 2.6**  
Módulo de Reconocimiento.

El reconocimiento permite identificar una señal de voz similar a ciertos patrones definidos previamente. Consta de los siguientes módulos: autocorrelación, análisis LPC, segmentación y comparación.

## Capítulo 3

# Hardware para la implementación del sistema de reconocimiento de voz

En el capítulo anterior se planteó el procedimiento para procesar las grabaciones de señales de voz, tanto para el entrenamiento como para el reconocimiento o clasificación. Se planteó que el sistema sería dependiente del locutor, y de hecho, se utilizó un solo hablante. También se diseñó el sistema de reconocimiento.

El siguiente paso es la implementación del propio sistema de reconocimiento, y para esto fue necesario utilizar un DSP, el cual que procesa las señales de voz a una velocidad suficiente de tal forma que se trabaje en tiempo real.

A continuación se explica lo que es un DSP, se dan algunos datos relevantes acerca de estos, también se explica lo que es un DSK y se presentan las características generales del usado en este proyecto, y por último, se enlistan algunas características del sistema de reconocimiento de voz.

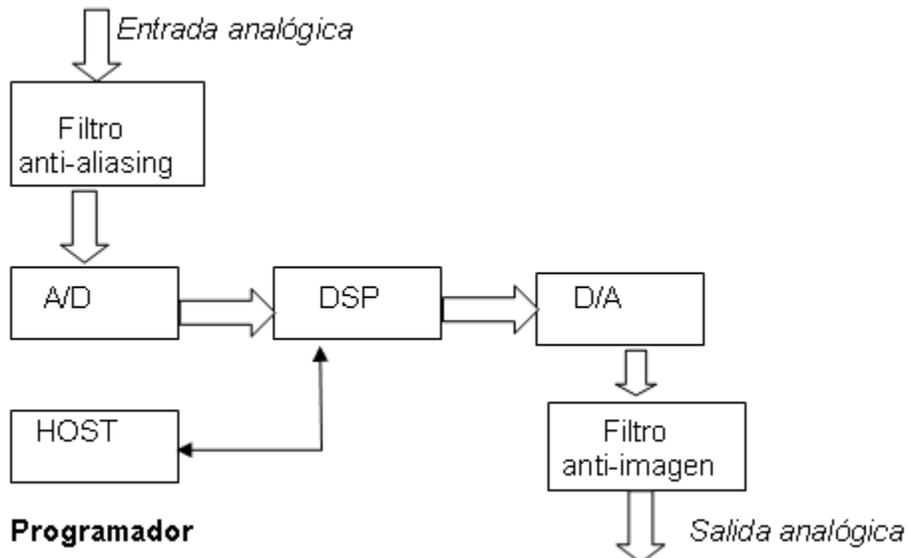
### 3 – 1 Procesador digital de señales.

Un **DSP** (acrónimo de *Digital Signal Processor*) es un sistema basado en un procesador o microprocesador que posee un juego de instrucciones, un hardware y un software optimizados para aplicaciones que requieran operaciones numéricas a muy alta velocidad. Debido a esto, un DSP es especialmente útil para el procesamiento y representación de señales analógicas en tiempo real: en un sistema que trabaje de esta forma (tiempo real) se reciben muestras (**samples** en inglés), normalmente provenientes de un convertidor analógico/digital (ADC).

Se ha dicho que puede trabajar con señales analógicas, pero es un sistema digital, por lo tanto necesitará un convertidor analógico/digital a su entrada y digital/analógico en la salida. Como todo sistema basado en un procesador programable necesita una memoria donde almacenar los datos con los que trabajará y el programa que ejecuta.

Si se tiene en cuenta que un DSP puede trabajar con varios datos en paralelo y un diseño e instrucciones específicas para el procesamiento digital, se puede dar una idea de su enorme potencia para este tipo de aplicaciones. Estas características constituyen la principal diferencia de un **DSP** y otros tipos de procesadores.

Para describir su funcionamiento se pondrá el ejemplo de un filtro: el **DSP** recibirá valores digitales o muestras procedentes de la señal de entrada, calcula la salida correspondiente a esos valores con el filtro que se le ha programado. Un posible sistema basado en un **DSP** puede ser el de la figura 3.1, donde la entrada analógica pasa por un filtro antialiasing, y después es convertida en una señal digital por medio del convertidor analógico-digital. La señal digital es procesada por el DSP y posteriormente se transforma en una señal analógica con el convertidor digital – analógico. Por último la señal pasa por un filtro anti-imagen.



**Figura 3.1**  
Sistema basado en un DSP.

La señal analógica entra directamente en un filtro antialiasing para evitar frecuencias superiores a la de muestreo del convertidor analógico-digital. Después se lleva a cabo el procesado digital en el módulo DSP, y se vuelve a convertir en analógica para presentarse a la salida.

La fabricación de DSP's se remonta a finales de los años 1970. Intel y AMI presentaron dispositivos para procesar señales sin mucho éxito comercial. En 1979, Bell Labs presentó el primer chip para procesar digitalmente señales, el *Mac 4 Microprocessor*. En los años 1981 NEC Corporation y AT&T presentaron dispositivos parecidos. Hoy en día los tres fabricantes más importantes son Texas Instruments, Analog Devices y Motorola.

Un **DSP** se puede programar tanto en ensamblador como en C. Cada familia de **DSP** tiene su propio lenguaje ensamblador y sus propias herramientas suministradas por el fabricante. En los últimos tiempos hay otros fabricantes que están intentando que sus aplicaciones programen también **DSP** como son Matlab y Labview.

Las aplicaciones más habituales en las que se emplean **DSP** son el procesado de audio y video; y cualquier otra aplicación que requiera el procesado en tiempo real. Con estas aplicaciones se puede eliminar el eco en las líneas de comunicaciones, lograr hacer más claras imágenes de órganos internos en los equipos de diagnóstico médico, cifrar conversaciones en teléfonos celulares para mantener privacidad, analizar datos sísmicos para encontrar nuevas reservas de petróleo, hacer posible las comunicaciones wireless LAN, el reconocimiento de voz, los reproductores digitales de audio, los modems inalámbricos, las cámaras digitales, y una larga lista de elementos que pueden ser relacionados con el proceso de señales.

Un **DSP** está diseñado teniendo en cuenta las tareas más habituales del procesado digital: sumas, multiplicaciones y retrasos (almacenar en memoria).

Tienen arquitectura Harvard y algunos pueden manejar números con punto flotante.

Los elementos básicos que componen un **DSP** son:

- Conversores en las entradas y salidas
- Memoria de datos, memoria de programa y DMA.
- MACs: multiplicadores y acumuladores.
- ALU: Unidad aritmético-lógica.
- Registros.

### 3 – 2 Hardware utilizado.

Para este proyecto se utilizó el DSK (DSP starter kit) TMS320C6711, que proporciona a los ingenieros diseñadores de sistemas una forma “fácil y económica” de llevar sus diseños de alto desempeño, basados en DSP’s de la familia C6000, de la concepción a la producción. El DSK TMS320C6711 no solo introduce la tecnología C6000, sino que también es suficientemente poderoso para ser usado en aplicaciones de alto desempeño en los campos de las telecomunicaciones, procesamiento de imágenes y muchos otros.

Los DSP’s TMS320C67x (incluyendo los TMS320C6711, TMS320C6711B, TMS320C6711C, TMS320C6711D) forman la familia de punto flotante en la plataforma TMS320C6000. Los DSP’s C6711, C6711B, C6711C y C6711D tienen una arquitectura desarrollada por Texas Instruments llamada “palabra de instrucción muy larga” (por sus siglas en inglés VLIW), la cual hace de esta familia una buena elección para aplicaciones multicanal y multifunción.

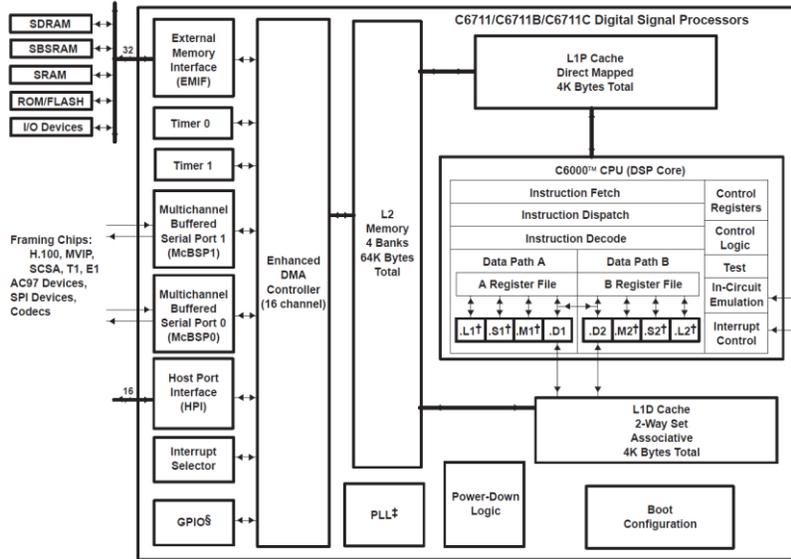
Con un desempeño de hasta 900 millones de operaciones de punto flotante por segundo (MFLOPS) con un reloj de 150 MHz, los dispositivos C6711/C6711B ofrecen soluciones económicas y efectivas para retos de programación de alto rendimiento. Los DSP’s C6711/C6711B poseen la flexibilidad operacional de los controladores a alta velocidad y la capacidad numérica de los procesadores tipo array (su set de instrucciones incluye operaciones matemáticas que se pueden realizar sobre múltiples elementos de forma simultánea). Estos procesadores tienen 32 registros de propósito general de 32 bits cada uno y ocho unidades funcionales altamente independientes. Las ocho unidades funcionales proporcionan cuatro ALU’s de punto fijo/flotante, dos ALU’s de punto fijo, y dos multiplicadores de punto fijo/flotante. Los dispositivos C6711/C6711B pueden producir dos MAC’s (multiplica y acumula) por ciclo para un total de 300 MMACS.

Con un desempeño de hasta 1200 millones de operaciones de punto flotante por segundo (MFLOPS) utilizando un reloj de 200 MHz (para los dispositivos C6711C/D), o hasta 1500 MFLOPS con un reloj de 250 MHz (para el C6711D), el C6711C ofrece también soluciones económicas y efectivas para retos de programación de alto rendimiento. El DSP C6711C posee la flexibilidad operacional de los controladores a alta velocidad y la capacidad numérica de los procesadores tipo array. Este procesador tiene 32 registros de propósito general de 32 bits cada uno y ocho unidades funcionales altamente independientes. Las ocho unidades funcionales proporcionan cuatro ALU’s de punto fijo/flotante, dos ALU’s de punto fijo, y dos multiplicadores de punto fijo/flotante. El dispositivo C6711C puede producir dos MAC’s por ciclo para un total de 400 MMACS.

Los dispositivos C6711, C6711B, C6711C tienen también hardware de propósito específico, memoria integrada y periféricos adicionales integrados.

Los dispositivos C6711, C6711B, C6711C usan una arquitectura basada en dos niveles de cache y tienen un set de poderosos y diversos periféricos. El set de periféricos incluye dos puertos seriales multicanal (McBSPs), dos timers de propósito general, una interface host-port (HPI) y una interfaz de memoria externa compatible con SDRAM, SBSRAM y periféricos asíncronos.

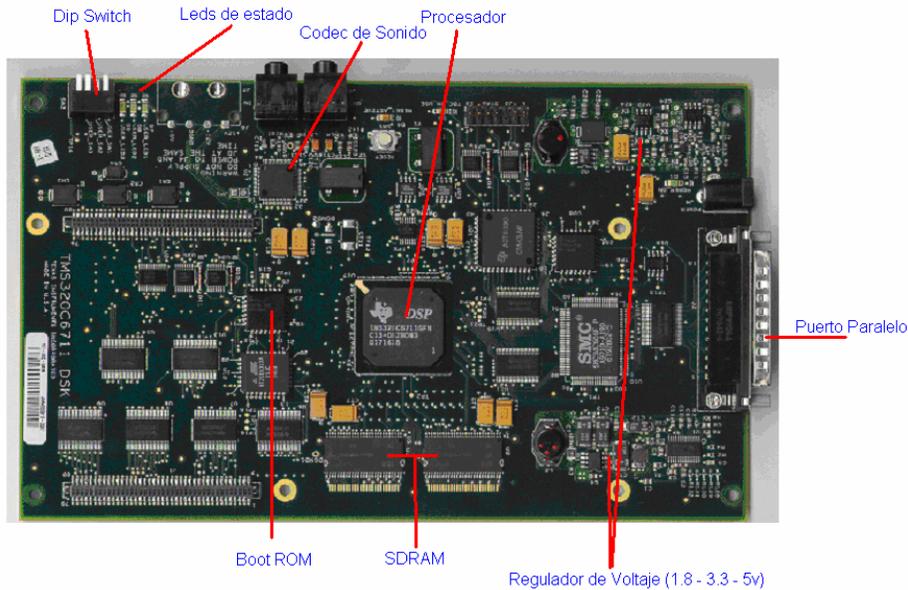
En la figura 3.2 se muestra un diagrama donde se observa el CPU del TMS320C6711 y la manera en que está conectado con los periféricos.



**Figura 3.2**  
Diagrama del CPU y periféricos del TMS320C6711.

Los dispositivos C6711, C6711B, C6711C tienen un set completo de herramientas de desarrollo que incluye: un compilador de C, un optimizador de ensamblador para simplificar la programación, y un debugger ejecutable en Windows.

En particular, la tarjeta DSK usada en el desarrollo de este proyecto contiene un DSP TMS320C6711 y se muestra en la figura 3.3.



**Figura 3.3**  
DSK TMS320C6711.

Cabe mencionar también que esta familia de DSP's salió al mercado en 1999 y ya no está disponible debido a que han salido DSP's más veloces y más poderosos. Texas Instruments en su página electrónica recomienda el TMS320C6713 DSP Starter Kit como reemplazo.

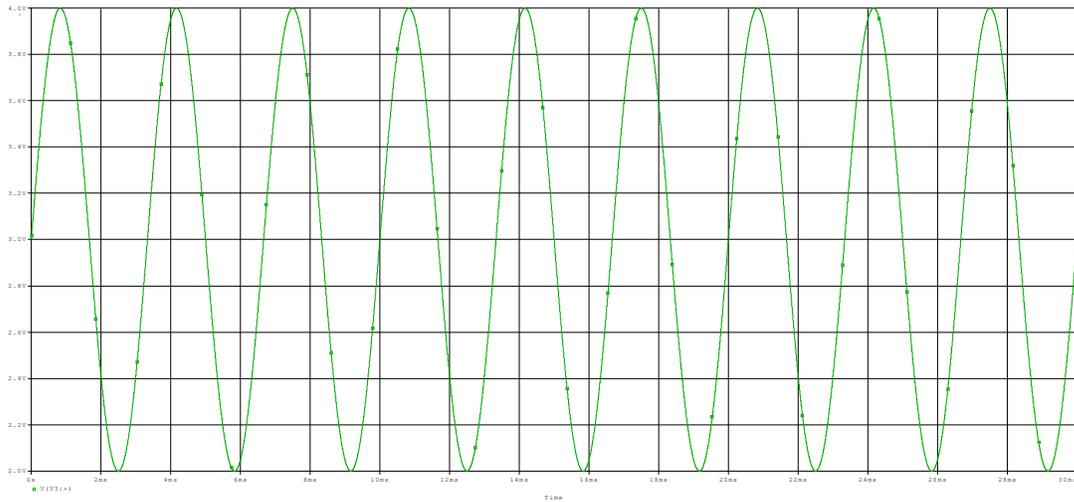
Algunas de las características del sistema de reconocimiento de voz que se programó en el DSP son las siguientes.

- Sistema dependiente del locutor (único locutor).
- Reconocimiento en tiempo real.
- 13 comandos que se reconocen.
  1. Apaga baño.
  2. Apaga cocina.
  3. Apaga comedor.
  4. Apaga jardín.
  5. Apaga recamara.
  6. Apaga sala.
  7. Enciende baño.
  8. Enciende cocina.
  9. Enciende comedor.
  10. Enciende jardín.
  11. Enciende recamara.
  12. Enciende sala.
  13. Señal desconocida.
- Precisión mayor al 98%, en presencia de ruido (en el laboratorio de voz).

## Capítulo 4

### Descripción de los elementos usados en el proyecto

Como se mencionó en la introducción, la señal proveniente del DSP está compuesta por 3 V de offset más la señal que nos interesa, que es la sinusoidal de diferentes frecuencias (ver Figura 4.1).



**Figura 4.1**

Señal proveniente del DSP.

Para poder identificar la señal con el microcontrolador, la señal proveniente del DSP debe ser acondicionada; para llevar a cabo dicho acondicionamiento es necesario eliminar el voltaje de offset y convertirla a un tren de pulsos de la misma frecuencia de la señal sinusoidal, de esta forma el microcontrolador será capaz de contar los pulsos por un segundo y así saber la frecuencia de la señal enviada por el DSP.

A continuación se describe la manera como se logra quitar el offset de la señal proveniente del DSP y como esta señal sinusoidal es convertida en un tren de pulsos.

#### 4 – 1 Filtro paso altas.

Un filtro paso altas es una red selectiva de frecuencia que deja pasar solamente señales de frecuencia mayor a su frecuencia de corte  $f_c$ . Para este proyecto se busca un filtro con una frecuencia de corte baja, inferior a los 300 Hz, correspondientes a la frecuencia de la señal sinusoidal de más baja frecuencia. Para el diseño del filtro se eligió una frecuencia de corte de 100 Hz y un filtro paso altas RC, y se propone un capacitor de 220 nF. La ecuación de la frecuencia de corte de este filtro es

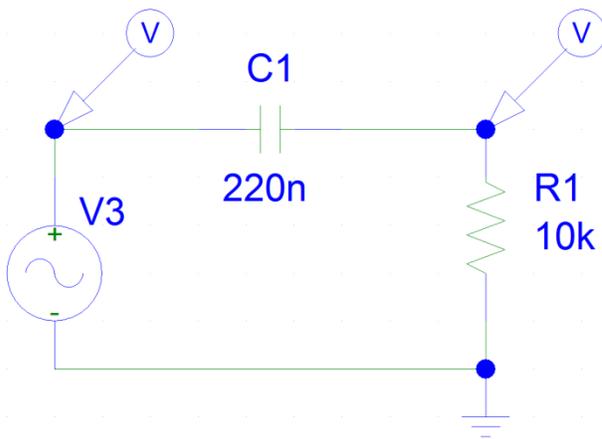
$$f_c = \frac{1}{2\pi RC}$$

despejando el valor de R, se obtiene

$$R = \frac{1}{2\pi f_c C} = 7.234k\Omega$$

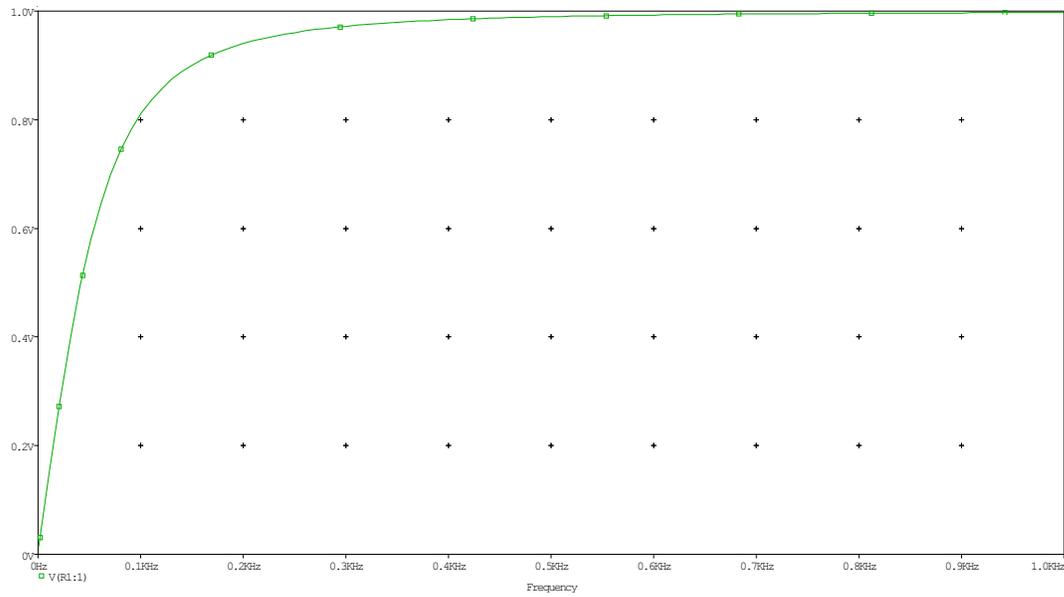
Si la resistencia aumenta la frecuencia de corte disminuye, entonces se elige un resistor de 10 k $\Omega$  que es un valor comercial muy fácil de conseguir, y con este valor se arma un filtro que cumple con el propósito de eliminar el voltaje de directa, y además se asegura que no elimina las señales de interés.

En la figura 4.2 se muestra el filtro paso altas elegido, que es un filtro de primer orden; además, en la Figura 4.3 podemos ver su respuesta en frecuencia.



**Figura 4.2**

Filtro paso altas usado en el proyecto.

**Figura 4.3**

Respuesta en frecuencia del filtro.

En la Figura 4.3 se puede observar que el offset que acompañaba a la señal sinusoidal ya no está presente a la salida del filtro, ya que todas las señales menores a aproximadamente 0.1 KHz son eliminadas. Como todas las señales que nos interesan son de 300 Hz o mayores, este filtro cumple con los requerimientos.

#### 4 – 2 Comparadores.

Un comparador es un circuito que analiza una señal de voltaje en una entrada con el voltaje de referencia presente en la otra entrada, dependiendo que voltaje de las señales sea mayor, la salida del comparador alcanza el valor de la fuente positiva o negativa con el que es alimentado.

Se pueden utilizar amplificadores operacionales para implementar comparadores, también existen circuitos integrados diseñados especialmente para trabajar como comparadores.

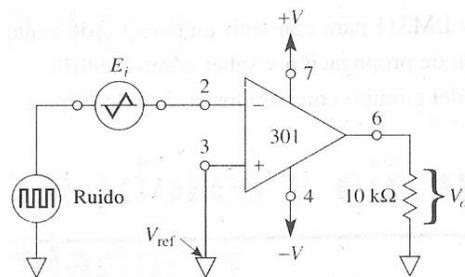
Tanto el amplificador operacional de propósito general, como el comparador, no funcionan adecuadamente si hay ruido en cualquier entrada. Para resolver este problema se emplea la retroalimentación positiva. Hay que tomar en cuenta que la retroalimentación positiva no elimina el ruido, sino que logra que el amplificador operacional responda menos a él.

#### Efecto del ruido en los circuitos comparadores.

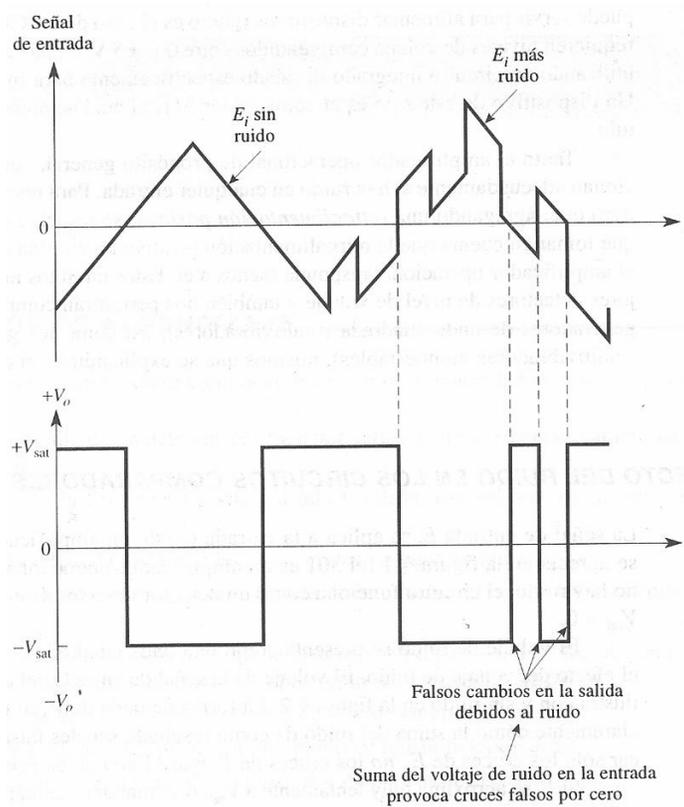
La señal de entrada  $E_i$  se aplica en la entrada (-) de un amplificador operacional LM301 como se aprecia en la Figura 4.4 (el 301 es un amplificador operacional de propósito general). Si no hay ruido, el circuito funciona como un detector de cruce por cero, debido a que  $V_{ref} = 0$ .

El voltaje de ruido se presenta como una onda cuadrada en serie con  $E_i$ , para mostrar el efecto del voltaje de ruido. El voltaje de la señal de entrada del amplificador operacional se ilustra con y sin ruido en la Figura 4.5. La forma de onda  $V_o$ , en función del tiempo, muestra claramente como la suma del ruido da como resultado señales falsas a la salida.  $V_o$  debe indicar sólo los cruces de  $E_i$ , no los cruces de  $E_i$  más el voltaje de ruido.

Si  $E_i$  se aproxima muy lentamente a  $V_{ref}$ , o se mantiene cercano a este valor,  $V_o$  puede seguir todas las oscilaciones del voltaje de ruido, o bien iniciar bruscamente una oscilación de alta frecuencia. Estos cruces en falso se pueden eliminar mediante la retroalimentación positiva.



**Figura 4.4**  
Detector inversor de cruce por cero.



**Figura 4.5**  
Efecto del ruido en un detector de cruce por cero.

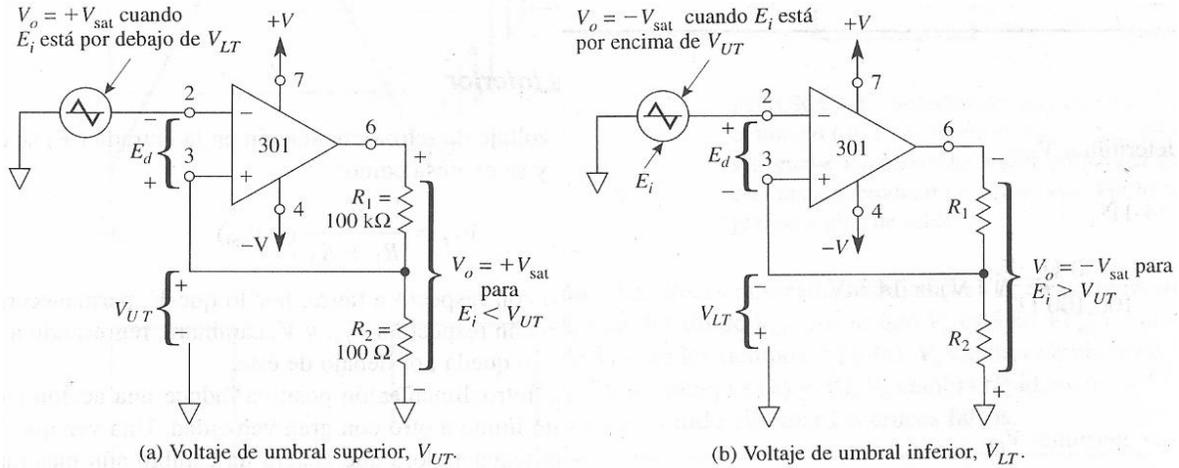
### Retroalimentación positiva.

La retroalimentación positiva se realiza tomando una fracción del voltaje de salida  $V_o$  y aplicando esta a la entrada (+). En la Figura 4.6(a), el voltaje de salida  $V_o$  se divide entre  $R_1$  y  $R_2$ . Una fracción de  $V_o$  se retroalimenta a la entrada (+) y crea así un voltaje de referencia que depende de  $V_o$ . Ahora se describirá que es la retroalimentación positiva y como servirse de ella para eliminar cambios falsos en la salida provocados por el ruido.

### Voltaje de umbral superior.

En la Figura 4.6(a), como se mencionó, el voltaje de salida  $V_o$  se divide entre  $R_1$  y  $R_2$  y la fracción de  $V_o$  resultante se retroalimenta a la entrada (+). Cuando  $V_o = +V_{sat}$ , el voltaje retroalimentado se denomina umbral superior de voltaje,  $V_{UT}$ . Este voltaje se expresa en función del divisor de voltaje de la siguiente manera:

$$V_{UT} = \frac{R_2}{R_1 + R_2} (+V_{sat})$$

**Figura 4.6**

$R_1$  y  $R_2$  retroalimentan un voltaje de referencia de la salida a la terminal de entrada (+).

Para valores de  $E_i$  inferiores a  $V_{UT}$ , el voltaje en la entrada (+) es mayor que el voltaje en la entrada (-). Por lo tanto,  $V_o$  se fija como  $+V_{sat}$ .

Si  $E_i$  se vuelve ligeramente más positivo que  $V_{UT}$ , la polaridad de  $E_d$ , como se muestra, se invierte y el valor de  $V_o$  comienza a descender. Ahora la fracción de  $V_o$  desciende con mayor rapidez y llega pronto al valor  $-V_{sat}$ ; de esta manera, el circuito es estable, de acuerdo con lo que se muestra en la Figura 4.6(b).

### Voltaje de umbral inferior.

Cuando  $V_o$  está en  $-V_{sat}$ , el voltaje de retroalimentación en la entrada (+) se denomina umbral inferior de voltaje,  $V_{LT}$ , y se expresa como:

$$V_{LT} = \frac{R_2}{R_1 + R_2} (-V_{sat})$$

Observe que  $V_{LT}$  es negativo con respecto a tierra, por lo que  $V_o$  permanecerá en  $-V_{sat}$ , en tanto que  $E_i$  sea mayor o positivo con respecto a  $V_{LT}$ , y  $V_o$  cambiara, regresando a  $+V_{sat}$  si  $E_i$  se vuelve más negativo que  $V_{LT}$ , o queda por debajo de éste.

Se concluye así que la retroalimentación positiva induce una acción casi instantánea que permite cambiar  $V_o$  de un límite a otro con gran velocidad. Una vez que  $V_o$  comienza a cambiar, provoca una acción regeneradora que genera un cambio aun más rápido de  $V_o$ . Si los voltajes de umbral resultan mayores que los voltajes pico del ruido, la retroalimentación positiva eliminará las transiciones falsas de salida.

### Comparador seleccionado.

Para este proyecto no resulta importante el ancho de pulso, ni es necesario conservar el mismo signo entre la señal sinusoidal y la del tren de pulsos, lo único necesario es que la frecuencia del tren de pulsos sea la misma que la de la señal sinusoidal.

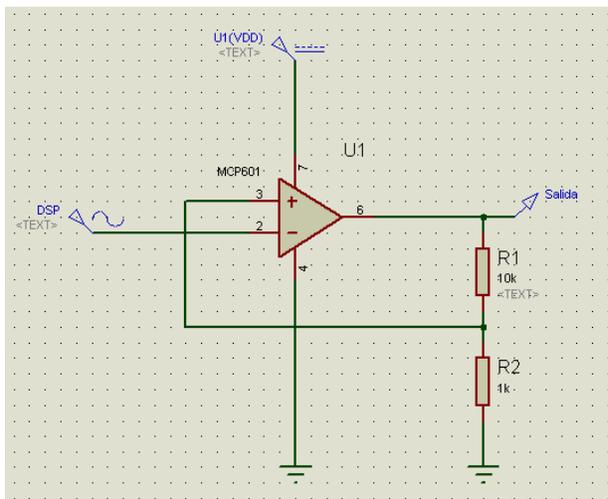
Debido a que las amplitudes del voltaje de ruido observadas en la etapa de pruebas nunca superaron los 0.4 V se considera un  $V_{UT}$  de 0.45 V, y como el voltaje de alimentación del amplificador operacional usado (MCP601) es de 5 V,  $+V_{sat}$  es igual a 5 V, para determinar las resistencias que se usaron en el circuito se propuso una  $R_1$  de 10 k $\Omega$  y despejando de la ecuación de  $V_{UT}$  resultó una  $R_2$  de 0.989 k $\Omega$ . Al final se eligió el valor de resistencia existente de 1 k $\Omega$ , quedando  $V_{UT}$  con este nuevo valor igual a 0.4545 V.

Y como el voltaje de alimentación negativa para el circuito se establece como 0 V o tierra,  $-V_{sat}$  es igual a 0 V.

El circuito usado es el que se muestra en la Figura 4.7, quedando sus voltajes de umbral como sigue:

$$V_{UT} = \frac{1K\Omega}{10K\Omega + 1K\Omega} (5) = 0.4545V$$

$$V_{LT} = \frac{1K\Omega}{10K\Omega + 1K\Omega} (0) = 0V$$

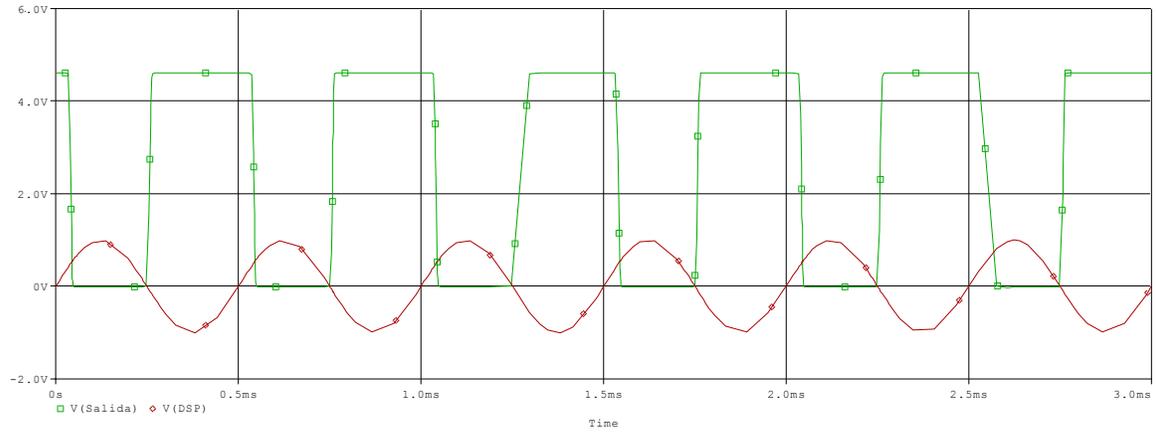


**Figura 4.7**

Comparador usado en este proyecto.

Se utilizó un amplificador operacional MCP601 debido a que este amplificador operacional trabaja con 5V, mismo voltaje con el que trabajan el resto de los circuitos integrados usados en este proyecto; además, así se genera la señal digital que se requiere con los voltajes permitidos en los pines del microprocesador.

La forma de onda que tiene este comparador a la salida se muestra en la Figura 4.8, y como se puede ver, esta lista para ser introducida en un microcontrolador, ya que es una señal lógica que varía de 0 a 5 V.



**Figura 4.8**

Señal en la entrada (-) y señal en la salida del comparador con retroalimentación positiva.

#### 4 – 3 Microcontrolador.

##### Introducción.

Un **microcontrolador** es un circuito integrado o chip que incluye en su interior las tres unidades funcionales de una computadora: CPU, Memoria y Unidades de E/S.

##### Características importantes del microcontrolador PIC16F877A.

En la Tabla 4.1 se resumen las características claves que posee este microcontrolador.

Características clave.	PIC16F877A.
Frecuencia de operación.	DC – 20MHz
Resets (y retardos).	POR, BOR (PWRT, OST)
Memoria de programa Flash (Palabras de 14 bits).	8K
Memoria de datos (bytes).	368
Memoria de datos EEPROM (bytes).	256
Interrupciones.	15
Puertos de entrada/salida.	Puertos A, B, C, D, E
Temporizadores (Timers).	3
Módulos de Captura/Comparación/PWM.	2
Comunicación serial.	MSSP, USART
Comunicación en paralelo.	PSP
Convertidores Analógico/Digital.	8
Comparadores Analógicos.	2
Set de instrucciones.	35 instrucciones.
Empaquetados.	PDIP de 40 pines. PLCC de 44 pines. TQFP de 44 pines. QFN de 44 pines.

**Tabla 4.1**

Principales características del PIC16F877A.

Para este proyecto solamente hacemos uso de algunas de las características que incluye el PIC16F877A, estas son:

- Memoria de programa.
- Memoria de datos.
- Set de instrucciones.
- Puerto B como entrada/salida de datos digitales.
- Puerto D como salida de datos digitales.
- Timer 0.
- Timer 1.

- Modo de bajo consumo de energía (Sleep).
- Interrupción por evento externo.
- Interrupción por desborde del Timer 0.

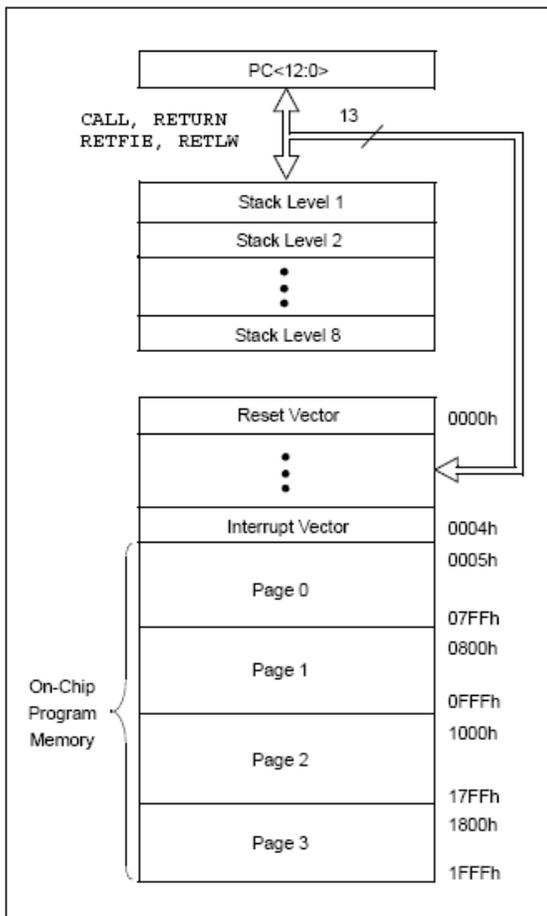
### Mapa de memoria.

Existen tres bloques de memoria en el PIC16F877A. La memoria de programa y la memoria de datos, las cuales tienen buses distintos, de tal forma que se puede tener acceso a ellos de manera concurrente. El otro bloque es el de la memoria EEPROM, el cual no se usa en este proyecto.

Memoria de programa.

El PIC16F877A tiene un contador de programa de 13 bits, capaz de acceder a cualquier dirección de la memoria de programa de 8K palabras x 14 bit. Si se intenta acceder a una dirección arriba de las físicamente existentes, se producirá un sobreflujo y se accederá a un espacio de memoria de programa no deseado.

El vector de reset está localizado en la dirección 0000h y el vector de interrupciones en la 0004h.



**Figura 4.9**

Mapa de memoria y Stack para el PIC16F877A.

Memoria de datos.

La memoria de datos esta colocada en diferentes bancos que contienen registros de función específica (SFRs) y registros de propósito general. Los bits RP1 (Status<6>) y RP0 (Status<5>) son los bits para seleccionar el banco.

RP1:RP0	Bank
00	0
01	1
10	2
11	3

**Tabla 4.2**

Estado de los bits RP1 y RP0 y bancos asociados.

Cada banco se extiende hasta 7Fh (128 bytes). Las localidades más bajas de cada banco están reservadas para los SFRs. Enseguida de los SFRs están los registros de propósito general implementados como RAM estática. Algunos SFRs frecuentemente usados se encuentran en más de un banco para tener acceso a ellos más rápidamente.

Registros de propósito general.

Se puede acceder a estos registros de manera directa o indirecta, a través del "File Select Register" (FSR).

Registros de función específica (SFR).

Son registros usados por el CPU y los módulos periféricos para controlar y definir el comportamiento del microprocesador. Estos registros son implementados como RAM estática. Estos registros pueden ser clasificados en dos categorías: del CPU y de los periféricos. Algunos de estos registros se explican en las secciones siguientes.

File Address		File Address		File Address		File Address				
Indirect addr. <sup>(*)</sup>	00h	Indirect addr. <sup>(*)</sup>	80h	Indirect addr. <sup>(*)</sup>	100h	Indirect addr. <sup>(*)</sup>	180h			
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h			
PCL	02h	PCL	82h	PCL	102h	PCL	182h			
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h			
FSR	04h	FSR	84h	FSR	104h	FSR	184h			
PORTA	05h	TRISA	85h		105h		185h			
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h			
PORTC	07h	TRISC	87h		107h		187h			
PORTD <sup>(1)</sup>	08h	TRISD <sup>(1)</sup>	88h		108h		188h			
PORTE <sup>(1)</sup>	09h	TRISE <sup>(1)</sup>	89h		109h		189h			
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah			
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh			
PIR1	0Ch	PIE1	8Ch	EEDATA	10Ch	EECON1	18Ch			
PIR2	0Dh	PIE2	8Dh	EEADR	10Dh	EECON2	18Dh			
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reserved <sup>(2)</sup>	18Eh			
TMR1H	0Fh		8Fh	EEADRH	10Fh	Reserved <sup>(2)</sup>	18Fh			
T1CON	10h		90h		110h		190h			
TMR2	11h	SSPCON2	91h	General Purpose Register 16 Bytes		General Purpose Register 16 Bytes		191h		
T2CON	12h	PR2	92h				111h		192h	
SSPBUF	13h	SSPADD	93h				112h		193h	
SSPCON	14h	SSPSTAT	94h				113h		194h	
CCPR1L	15h		95h				114h		195h	
CCPR1H	16h		96h				115h		196h	
CCP1CON	17h		97h				116h		197h	
RCSTA	18h	TXSTA	98h				117h		198h	
TXREG	19h	SPBRG	99h				118h		199h	
RCREG	1Ah		9Ah				119h		19Ah	
CCPR2L	1Bh		9Bh		11Ah		19Bh			
CCPR2H	1Ch	CMCON	9Ch		11Bh		19Bh			
CCP2CON	1Dh	CVRCON	9Dh		11Ch		19Ch			
ADRESH	1Eh	ADRESL	9Eh		11Dh		19Dh			
ADCON0	1Fh	ADCON1	9Fh		11Eh		19Eh			
	20h		A0h		11Fh		19Fh			
General Purpose Register 96 Bytes		General Purpose Register 80 Bytes		General Purpose Register 80 Bytes		General Purpose Register 80 Bytes		1A0h		
										1EFh
			accesses 70h-7Fh		accesses 70h-7Fh			accesses 70h - 7Fh		1F0h
Bank 0	7Fh	Bank 1	FFh	Bank 2	17Fh	Bank 3	1FFh			

Unimplemented data memory locations, read as '0'.  
 \* Not a physical register.

**Note 1:** These registers are not implemented on the PIC16F876A.  
**Note 2:** These registers are reserved; maintain these registers clear.

Figura 4.10

Mapa de memoria datos (Registros de función específica y de propósito general).

## Puertos.

Algunos pines de los puertos de entrada/salida digital están multiplexados con algún periférico del microcontrolador. En general, cuando dicho periférico está habilitado, ese pin no debe ser usado como pin de entrada/salida digital.

Para este proyecto solo son usados los puertos B y D del microcontrolador, y son los que se describen a continuación.

### Puerto B.

El puerto B consta de 8 bits y es bidireccional. El registro correspondiente de datos es el TRISB. Poner un bit del TRISB en 1 hará el mismo bit del puerto B un bit de entrada. Poner un bit del TRISB en 0 hará el mismo bit del puerto B un bit de salida.

Tres de los bits del puerto B están multiplexados con las funciones de "In-Circuit Debugger" y "Low-Voltage Programming": RB3/PGM, RB6/PGC y RB7/PGD.

Cuatro de los pines del puerto B, RB7:RB4, pueden ser programados para que cuando haya un cambio en alguno de estos pines se provoque una interrupción. Solo los pines configurados como entradas pueden causar la interrupción.

RB0/INT es el pin que se puede configurar para que haya una interrupción cuando haya un cambio. Esta interrupción es abordada con mayor detalle en la sección de interrupciones de este trabajo.

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
06h, 106h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
86h, 186h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
81h, 181h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

**Tabla 4.3**

Registros asociados con el Puerto B.

### Puerto D.

El puerto D consta de 8 bits con buffers de entrada tipo Schmitt Trigger. Cada pin es individualmente configurable como entrada o salida.

Este puerto puede ser configurado como un puerto paralelo de 8 bits poniendo en 1 el bit PSPMODE (TRISE<4>). En este modo los buffers de entrada son de tipo TTL.

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
08h	PORTD	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx xxxx	uuuu uuuu
88h	TRISD	PORTD Data Direction Register								1111 1111	1111 1111
89h	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction Bits			0000 -111	0000 -111

**Tabla 4.4**

Registros asociados con el Puerto D.

### Temporizadores.

El PIC16F877A dispone de un potente conjunto de temporizadores para manejar eficientemente todas las operaciones que involucran al tiempo y al conteo. Dichos temporizadores son tres y se denominan técnicamente TMR0, TMR1 y TMR2.

En este proyecto se hace uso únicamente del TMR0 y del TMR1.

TMR0.

Las características principales de este temporizador son las siguientes:

- Es un contador/temporizador de 8 bits.
- Legible y escribible.
- Reloj interno o externo.
- Selección del flanco en el reloj externo.
- Previsor de la frecuencia de reloj programable.
- Generación de interrupción opcional en el desbordamiento.

El registro de función específica (SFR), OPTION\_REG se encarga de controlar el funcionamiento de este temporizador, a continuación se presenta este registro que ocupa las direcciones 81h y 181h, y se describen los bits que tienen que ver con el temporizador.

### OPTION\_REG REGISTER

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

**Figura 4.11**

Bits que componen al registro OPTION\_REG.

Bit 5 – T0CS: Bit de selección del reloj para el TMR0.

1 = Por transición en el pin TOCKI

0 = Reloj interno de ciclo de instrucción (CLKO).

Bit 4 – TOSE: Flanco con el cual se produce un incremento.

1 = Incremento cuando el flanco es de bajada en el pin TOCKI.

0 = Incremento cuando el flanco es de subida en el pin TOCKI.

Bit 3 – PSA: Asignación del preescalador.

1 = Preescalador asignado al Watchdog Timer (WDT).

0 = Preescalador asignado al TMR0.

Bit 2-0 – PS2:PS0: Selección de la preescala.

Valor de los bits 2 – 0.	Asignado al TMR0.	Asignado al WDT.
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

**Tabla 4.5**

Significado de los valores de los bits PS2:PS0 cuando son asignados al TMR0 y al WDT.

El TMR0 para este proyecto se usa para contar un segundo cuando existe una interrupción por evento externo que despierta al microcontrolador, esto se hace cargándole el valor 59,  $59 = 255 - 196$ , al registro TMR0 y además configurando el preescalador con el valor de 256 de tal forma que solo cuente  $196 \times 256 = 50176$  ciclos de instrucción antes de que se desborde. Se sabe que un ciclo de instrucción sucede cada cuatro ciclos de reloj, y como en este proyecto se usó un reloj de cuarzo de 4 MHz, pasarán 50 ms antes de que se desborde el Timer 1 ( $50ms = 4 \times (\frac{1}{4MHz}) \times 196 \times 256$ ), y con la ayuda de un registro auxiliar (al cual se le asignó el nombre de “retardo”) podemos hacer que este proceso se realice 20 veces, y así contar un segundo.

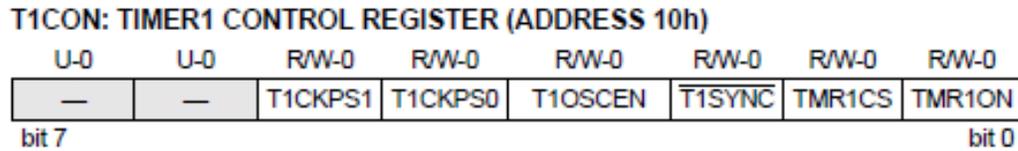
TMR1.

Las características principales de este temporizador son las siguientes:

- Es un contador/temporizador de 16 bits.
- Legible y escribible.
- Reloj interno o externo.
- Predivisor de la frecuencia de reloj programable.
- Generación de interrupción opcional en el desbordamiento.

- Posible reinicialización desde los módulos de CCP.

El SFR T1CON se encarga de controlar el funcionamiento de este temporizador, a continuación se presenta este registro que ocupa la dirección 10h, y se describe el funcionamiento de los bits.



**Figura 4.12**

Bits que componen al registro T1CON.

Bit 7-6 – No implementados: se leen siempre como un 0.

Bit 5-4 – T1CKPS1: T1CKPS0: Asignación de la preescala.

Valor de los bits 5 – 4.	Valor de la preescala.
00	1:1
01	1:2
10	1:4
11	1:8

**Tabla 4.6**

Preescala dependiendo de los valores de los bits 5 y 4.

Bit 3 – T1OSCEN: Habilitación del oscilador del Timer1.

1 = Oscilador habilitado.

0 = Oscilador apagado.

Bit 2 – T1SYNC: Sincronización con reloj externo.

- Cuando TMR1CS = 1:

1 = No sincronizar con reloj externo.

0 = Sincronizar con reloj externo.

- Cuando TMR1CS = 0:

Este bit es ignorado.

Bit 1 – TMR1CS: Selección de fuente de reloj para el Timer1.

1 = Reloj externo conectado al pin RC0/T1OSO/T1CKI (con flanco de subida).

0 = Reloj interno ( $F_{osc}/4$ ).

Bit 0 – TMR1ON: Activación del Timer1.

1 = Enciende el Timer1

0 = Detiene el Timer1.

En este proyecto en TMR1 es usado para llevar el conteo de los pulsos que llegan desde el DSP.

### **Modo de bajo consumo de energía (Sleep).**

Se entra a este modo ejecutando la instrucción SLEEP.

Los puertos de entrada/salida mantienen el mismo estado en que se encontraban antes de entrar al modo Sleep.

Para despertar al microprocesador debe ocurrir alguno de los siguientes eventos:

- Reset externo en el pin  $\overline{MCLR}$ .
- Por medio del Watchdog Timer (si está habilitado).
- Interrupción externa, interrupción por cambio en el puerto RB o interrupción de algún periférico.

Cuando la instrucción SLEEP está siendo ejecutada, la siguiente instrucción (PC + 1) se empieza a cargar. Para despertar al dispositivo por medio de una interrupción es necesario que esta interrupción este habilitada. El despertar se produce sin importar el estado del bit GIE. Si el bit GIE está en 0, el dispositivo continúa con la ejecución después de la instrucción SLEEP. Si este bit esta en 1 se ejecuta la instrucción después del SLEEP y salta a la dirección del vector de interrupción (0004h). En el caso de que la ejecución de la instrucción después del SLEEP no se requiera se debe poner un NOP después del SLEEP.

### **Interrupciones.**

Es una señal recibida por un procesador, indicando que debe "interrumpir" el curso de ejecución actual y pasar a ejecutar código específico para tratar esta situación.

Las interrupciones surgen de las necesidades que tienen los dispositivos periféricos de enviar información al procesador principal de un sistema. La primera técnica que se empleó fue que el propio procesador se encargara de sondear (polling) el dispositivo cada cierto tiempo para

averiguar si tenía pendiente alguna comunicación para él. Este método presentaba el inconveniente de ser muy ineficiente, ya que el procesador constantemente consumía tiempo en realizar todas las instrucciones de sondeo.

El mecanismo de interrupciones fue la solución que permitió al procesador desentenderse de esta problemática, y delegar en el dispositivo la responsabilidad de comunicarse con el procesador cuando lo necesitaba. El procesador, en este caso, no sondea a ningún dispositivo, sino que queda a la espera de que estos le avisen (le "interrumpan") cuando tengan algo que comunicarle (ya sea un evento, una transferencia de información, una condición de error, etc.).

Funcionamiento del mecanismo de interrupciones.

Cada dispositivo que desea comunicarse con el procesador por interrupciones debe tener asignada una línea única capaz de avisar a éste de que le requiere para una operación. Esta línea es la llamada IRQ ("Interrupt ReQuest", petición de interrupción).

Las IRQ son líneas que llegan al controlador de interrupciones, un componente hardware dedicado a la gestión de las interrupciones, y que puede estar integrado en el procesador principal o ser un circuito separado conectado al procesador principal. El controlador de interrupciones debe ser capaz de habilitar o inhibir líneas de interrupción (operación llamada comúnmente enmascarar por la utilización de una máscara), y establecer prioridades entre las distintas interrupciones habilitadas. Cuando varias líneas de petición de interrupción se activan a la vez, el controlador de interrupciones utilizará estas prioridades para escoger la interrupción sobre la que informará al procesador principal. Sin embargo hay interrupciones que no se pueden enmascarar o deshabilitar, las conocidas como interrupciones no enmascarables o NMI.

Un procesador principal (sin controlador de interrupciones integrado) suele tener una única línea de interrupción llamada habitualmente INT. Esta línea es activada por el controlador de interrupciones cuando tiene una interrupción que servir. Al activarse esta línea, el procesador consulta los registros del controlador de interrupciones para averiguar qué IRQ es la que ha de atender. A partir del número de IRQ busca en el vector de interrupciones qué rutina debe llamar para atender una petición del dispositivo asociado a dicha IRQ.

Las rutinas de interrupción generalmente toman un pequeño tiempo de ejecución y la mayoría no pueden ser interrumpidas cuando se están atendiendo, porque al entrar en ellas se almacena el estado de los registros en una pila y si se interrumpen muchas veces, la pila se puede desbordar.

Pasos típicos para el procesamiento de una IRQ:

- 1) Terminar la ejecución de la instrucción máquina en curso.
- 2) Salva el valor de contador de programa, PC, en la pila, de manera que en la CPU, al terminar el proceso, pueda seguir ejecutando el programa a partir de la última instrucción.
- 3) La CPU salta a la dirección donde está almacenada la rutina de servicio de interrupción (ISR, Interrupt Service Routine) y ejecuta esa rutina que tiene como objetivo atender al dispositivo que generó la interrupción.
- 4) Una vez que la rutina de la interrupción termina, el procesador restaura el estado que había guardado en la pila en el paso 2 y retorna al programa que se estaba usando anteriormente.



**PIR1 REGISTER (ADDRESS 0Ch)**

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	
PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	
bit 7								bit 0

**PIE2 REGISTER (ADDRESS 8Dh)**

U-0	R/W-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	
—	CMIE	—	EEIE	BCLIE	—	—	CCP2IE	
bit 7								bit 0

**PIR2 REGISTER (ADDRESS 0Dh)**

U-0	R/W-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	
—	CMIF	—	EEIF	BCLIF	—	—	CCP2IF	
bit 7								bit 0

**Figura 4.13**

Bits que componen a los registros INTCON, PIE1, PIR1, PIE2 y PIR2.

De las quince causas de interrupción que posee el PIC16F877A se hace uso solamente de dos, de la interrupción por evento externo y de la interrupción por desborde del Timer 0.

Interrupción por evento externo.

Esta interrupción es habilitada con el flanco de subida o de bajada de una señal digital conectada al pin RB0/INT del microcontrolador. Si el bit INTEDG (OPTION\_REG<6>) está en 1, se activa con flanco de subida, y si esta en cero con flanco de bajada. Cuando un flanco valido ocurre en el pin RB0/INT, la bandera INTF (INTCON<1>), se pone en 1. Esta interrupción puede ser deshabilitada poniendo en cero el bit INTE (INTCON<4>). La bandera INTF debe ser puesta en cero durante la rutina de interrupción antes de habilitar nuevamente esta interrupción. Esta interrupción puede despertar al procesador del estado de Sleep (bajo consumo de energía) si el bit INTE fue puesto en 1 antes de que el microprocesador fuera mandado a dicho estado. El estado del bit de habilitación global de interrupciones, GIE, decide si se va o no a la rutina de interrupción una vez que el microprocesador sale del modo Sleep.

Interrupción por desborde del Timer 0.

Un desborde (FFh → 00h) en el registro TMR0 pondrá en 1 la bandera de interrupción TMR0IF (INTCON<2>). Esta interrupción puede ser habilitada/deshabilitada poniendo en 1/0 el bit de habilitación, TMR0IE (INTCON<5>).

### Configuraciones del oscilador (reloj para el microcontrolador).

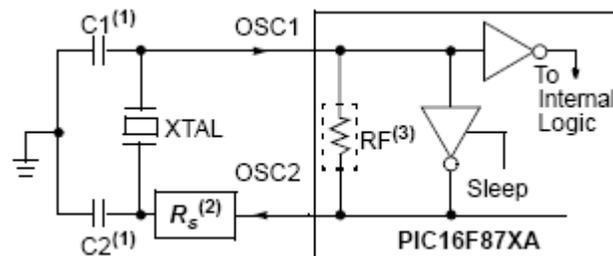
El PIC16F877A puede ser operado con cuatro diferentes modos de oscilación. El usuario puede programar dos bits de configuración (FOSC1 y FOSC0) para seleccionar una de estos cuatro modos:

- LP: Low-Power Crystal
- XT: Crystal/Resonator
- HS: High-Speed Crystal/Resonator
- RC: Resistor/Capacitor

Para este proyecto se usa el modo XT, ya que se usa un cristal de cuarzo de 4 MHz.

Oscilador por medio de cristal de cuarzo.

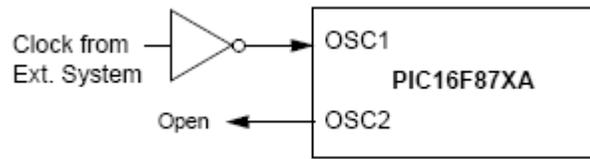
En los modos XT, LP o HS un cristal de cuarzo o un circuito integrado resonante cerámico es conectado en los pines OSC1/CLKI y OSC2/CLKO para establecer la oscilación (ver figura 4.14). Cabe mencionar que en los modos XT, LP o HS el PIC puede tener un reloj externo conectado a el pin OSC1/CLKI (ver figura 4.15).



**Figura 4.14**  
Circuito para los modos HS, XT o LP.

#### Notas

- 1: Ver tabla 4.7 para los valores de C1 y C2.
- 2: El uso de  $R_s$  no es estrictamente necesario y se puede reemplazar por un corto circuito.



**Figura 4.15**  
Forma de conectar un reloj externo al PIC.

Osc Type	Crystal Freq.	Cap. Range C1	Cap. Range C2
LP	32 kHz	33 pF	33 pF
	200 kHz	15 pF	15 pF
XT	200 kHz	47-68 pF	47-68 pF
	1 MHz	15 pF	15 pF
	4 MHz	15 pF	15 pF
HS	4 MHz	15 pF	15 pF
	8 MHz	15-33 pF	15-33 pF
	20 MHz	15-33 pF	15-33 pF

**Tabla 4.7**  
Valores recomendados de los capacitores para los osciladores con cristal.

#### 4 – 4 Programa.

##### Introducción.

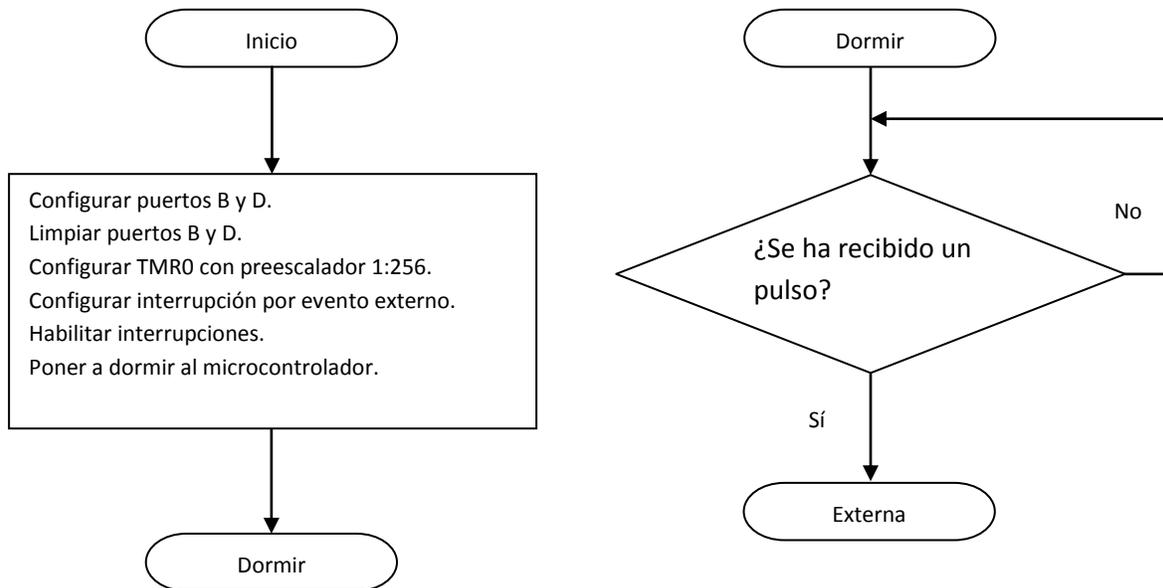
Cuando el microcontrolador es despertado por el primer pulso del tren de pulsos, cuenta por un segundo el número de pulsos, con lo que determina la instrucción y prende o apaga un pin del puerto D, después vuelve a dormir, esperando así a la próxima instrucción para ser despertado y realizar todo el proceso una vez más.

Como ya se ha mencionado se hace uso de interrupciones, contadores con preescaladores y de puertos. Además es conveniente mencionar que este programa está diseñado para que el PIC realice la función descrita con anterioridad usando un reloj de cuarzo de 4 MHz.

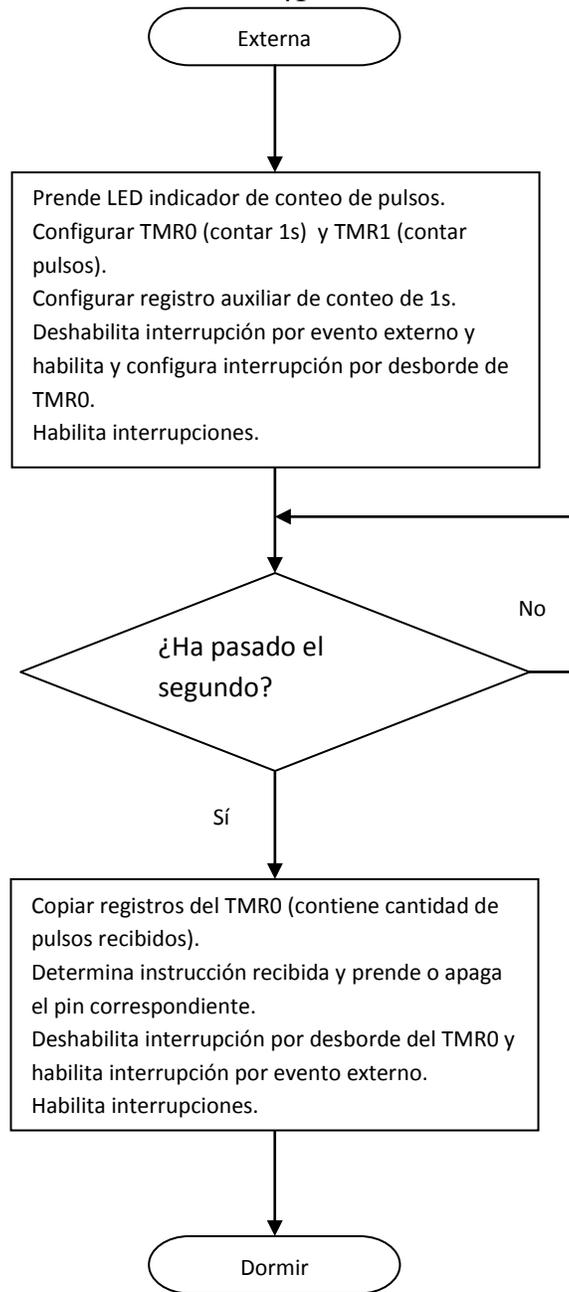
Un LED verde es prendido mientras el microcontrolador está recibiendo información del DSP y está en proceso de conteo, también un LED rojo es prendido cuando el DSP no reconoció el comando de voz, o cuando la frecuencia de la onda sinusoidal mandada por el DSP es de mayor frecuencia de las preestablecidas.

Los bits de configuración para seleccionar el modo de oscilación y para apagar el Watchdog Timer son configurados por medio del programa usado para programar el microcontrolador, que es el MPLAB de Microchip, por esta razón no aparece código alguno para establecer el estado de los bits correspondientes a las funciones mencionadas.

A continuación se presenta un diagrama de flujo que ilustra el funcionamiento del programa.



41



**Programa.**

```

retardo    equ 21h           ; localidades de necesarias para almacenar variables
cuenta_h   equ 22h
cuenta_l   equ 23h

list p=16f877a

include "p16f877a.inc"

org 0      ;vector de reset
goto inicio

org 4      ;vector d interrupción
goto intvect

org 5      ;programa

inicio     bcf STATUS, RP1
           bsf STATUS, RP0      ;cambia al banco 1
           movlw 03h           ;configura el puerto B
           movwf TRISB
           movlw 00h           ;configura el puerto D (todas salidas)
           movwf TRISD
           movlw 0C7h          ;configura TIMER0 con preescalador 1:256
           movwf OPTION_REG
           bcf STATUS,RP0

           clrf PORTB          ;limpia puerto B
           clrf PORTD          ;limpia puerto D
           clrf TMR1L          ;limpia contenido del timer1
           clrf TMR1H
           movlw 90h           ;configura interrupciones
           movwf INTCON        ;interrupción externa y permiso global de interrupciones
           sleep               ;duerme y espera el primer pulso del tren de pulsos.
           nop

cuenta     nop                 ;ciclo infinito que espera a que se desborde el timer0
           goto cuenta         ;cada 50ms

intvect    btfsc INTCON,TMR0IF ;checa causa de interrupción
           goto timer          ;la causa fue el desborde del timer0
           goto externa        ;la causa fue el INT

timer      decfsz retardo,F
           goto aunno          ;todavía no es un segundo
           bcf T1CON,0         ;ya es un segundo
           movf TMR1L,W        ;guardar el resultado del timer1

```

```

movwf cuenta_l
movf TMR1H,W
movwf cuenta_h
clrf TMR1L           ;limpia registros del timer1
clrf TMR1H
bcf PORTD,6         ;apaga pin6 del puerto D

movf cuenta_h,F
btfss STATUS,Z      ;determinar si la instrucción fue de las tres frecuencias mas
                    ; altas
goto grande

movf cuenta_l,W     ; para evitar confusiones por pulsos de
sublw .20           ; baja frecuencia causados por inestabilidad del
                    ; comparador

btfsc STATUS,C
goto setreg         ;si la causa es ruido no ejecutar acción alguna

movf cuenta_l,W     ;resultado negativo, y seguimos buscando
sublw .40           ;para 300Hz
btfsc STATUS,C
goto primera       ;resultado positivo, y es la frecuencia buscada

movf cuenta_l,W     ;resultado negativo, y seguimos buscando
sublw .65           ;para 500Hz
btfsc STATUS,C
goto segunda

movf cuenta_l,W
sublw .90           ;700Hz
btfsc STATUS,C
goto tercera

movf cuenta_l,W
sublw .115          ;900Hz
btfsc STATUS,C
goto cuarta

movf cuenta_l,W
sublw .140          ;1100Hz
btfsc STATUS,C
goto quinta

movf cuenta_l,W
sublw .165          ;1300Hz
btfsc STATUS,C
goto sexta

```

```

movf cuenta_I,W
sublw .190           ;1500Hz
btfsc STATUS,C
goto septima

movf cuenta_I,W
sublw .215           ;1700Hz
btfsc STATUS,C
goto octava

movf cuenta_I,W
sublw .240           ;1900Hz
btfsc STATUS,C
goto novena

grande  movf cuenta_I,W           ;busca entre las tres mas altas frecuencias
sublw .10           ;2100Hz
btfsc STATUS,C
goto decima           ;resultado positivo y es la frecuencia buscada

movf cuenta_I,W           ;resultado negativo, y seguimos buscando
sublw .35           ;2300Hz
btfsc STATUS,C
goto once

movf cuenta_I,W
sublw .60           ;2500Hz
btfsc STATUS,C
goto doce
goto trece           ;fueron 2700 pulsos o mas, comando no reconocido por el DSP

primera  bcf PORTD,0           ;apaga baño
goto setreg

segunda  bcf PORTD,1           ;apaga cocina
goto setreg

tercera  bcf PORTD,2           ;apaga comedor
goto setreg

cuarta  bcf PORTD,3           ;apaga jardín
goto setreg

quinta  bcf PORTD,4           ;apaga recamara
goto setreg

sexta  bcf PORTD,5           ;apaga sala
goto setreg

septima  bsf PORTD,0           ;enciende baño
goto setreg

octava  bsf PORTD,1           ;enciende cocina
goto setreg

```

```

novena    bsf PORTD,2           ;enciende comedor
          goto setreg
decima    bsf PORTD,3           ;enciende jardín
          goto setreg
once      bsf PORTD,4           ;enciende recamara
          goto setreg
doce      bsf PORTD,5           ;enciende sala
          goto setreg
trece     bsf PORTD,6           ;enciende indicador de comando desconocido
          goto setreg

setreg    bcf INTCON, TMR0IE      ;deshabilita interrupción por desborde de timer0
          bcf INTCON, TMR0IF      ;limpia bandera de timer0
          bcf PORTB, 3           ;apaga LED indicador de conteo
          bsf INTCON, INTE       ;habilita interrupción externa
          bcf INTCON, INTF       ;limpia bandera por interrupción externa
          bsf INTCON, GIE       ;habilita permiso global de interrupciones
          sleep                  ;duerme otra vez esperando una interrupción externa
          nop

aunno     movlw ~.196           ;para que tarde 50 ms
          movwf TMR0
          bcf INTCON, TMR0IF      ;limpia banderas de interrupción
          bsf INTCON, GIE       ;habilita permiso global de interrupciones
          goto cuenta

externa   movlw 37h             ;configura timer1
          movwf T1CON
          bsf PORTB, 3           ;prende LED indicador de conteo
          movlw ~.196           ;para que tarde 50ms en desbordarse con un reloj de 4MHz
          movwf TMR0
          movlw .20              ; 20 por 50ms es un segundo
          movwf retardo
          bcf INTCON, INTE       ;deshabilita interrupción externa
          bsf INTCON, TMR0IE      ;habilita interrupción por desborde de TIMER0
          bcf INTCON, TMR0IF      ;limpia bandera de interrupción del TIMER0
          bsf INTCON, GIE       ;habilita permiso global de interrupciones
          goto cuenta

end

```

#### 4 – 5 Tiristores.

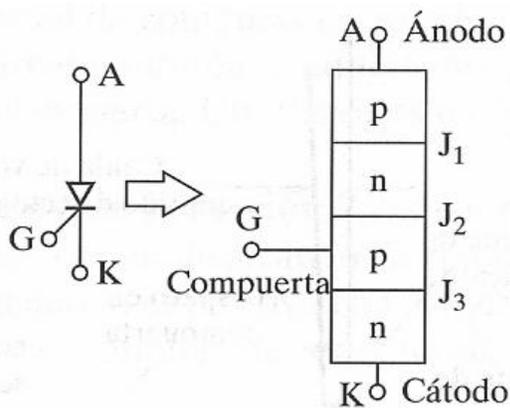
##### Introducción.

Los tiristores son una familia de dispositivos semiconductores de potencia. Se usan mucho en circuitos electrónicos de potencia. Se manejan como interruptores biestables, que funcionan de un estado no conductor a un estado conductor. Se puede suponer que los tiristores son interruptores ideales en muchas aplicaciones, pero los tiristores prácticos tienen ciertas características y limitaciones.

Los tiristores convencionales se diseñan sin la función de apagado controlada por la compuerta, y en ese caso, el tiristor puede pasar de su estado conductor hasta un estado no conductor sólo cuando se hace bajar la corriente a cero por otros medios.

##### Características de los tiristores.

Un tiristor es un dispositivo semiconductor con cuatro capas, de estructura *pnpn*, con tres uniones *pn*. Tiene tres terminales: ánodo, cátodo y compuerta. La figura 4.16 muestra el símbolo del tiristor y el corte de las tres uniones *pn*. Los tiristores se fabrican por dispersión difusión.

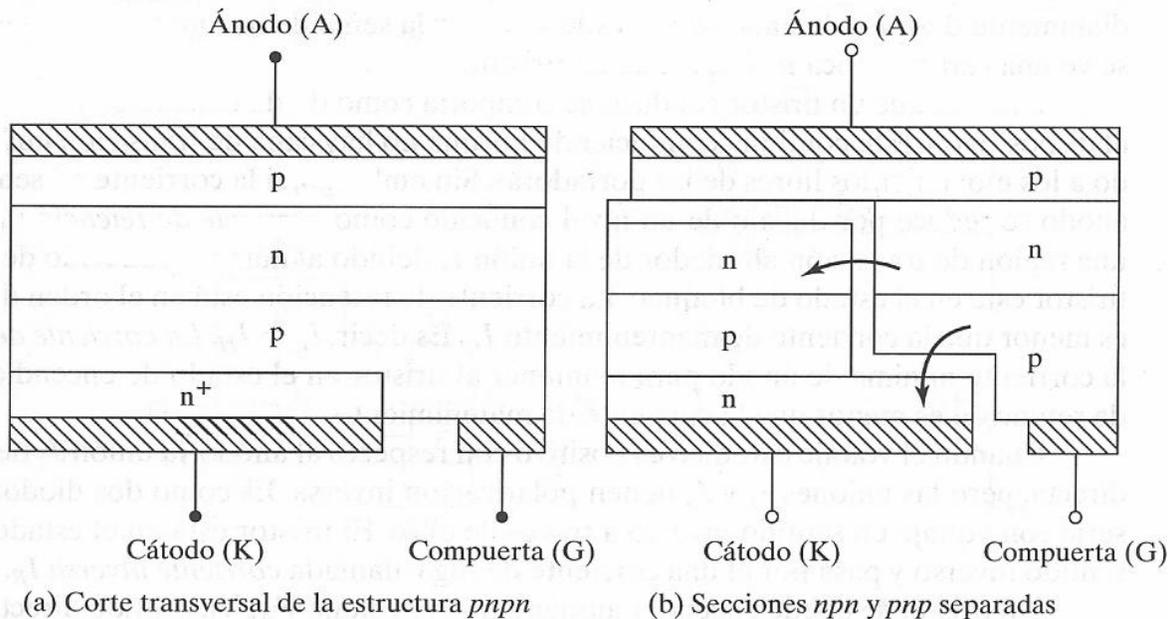


**Figura 4.16**

Símbolo del tiristor y tres uniones *pn*.

El corte transversal de un tiristor se ve en la Figura 4.17a, que se puede dividir en dos secciones de *pnp* y *pnp*, como se ve en la figura 4.17b.

Cuando el voltaje del ánodo se hace positivo con respecto al del cátodo, las uniones  $J_1$  y  $J_3$  tienen polarización directa. La unión  $J_2$  tiene polarización inversa, y solo pasa una pequeña corriente de fuga de ánodo a cátodo. En este caso, se dice que el tiristor está en la condición de bloqueo directo o en estado apagado, y la corriente de fuga se llama corriente en estado apagado,  $I_D$ . Si aumenta el voltaje de polarización inversa  $V_{AK}$  de ánodo a cátodo hasta un valor suficientemente grande, se daña de forma permanente la unión  $J_2$ . A esto se le llama ruptura por avalancha, y el voltaje correspondiente se llama voltaje  $V_{BO}$ , voltaje de avalancha directo. Ya que las otras uniones  $J_1$  y  $J_3$  ya están polarizadas directamente, hay un flujo libre de portadores a través de las tres uniones, dando como resultado una gran corriente anódica directa. Entonces, el dispositivo está en estado conductor, o estado encendido. La caída de voltaje se debe a la caída óhmica en las cuatro capas, y es pequeña, normalmente de 1 V. En el estado de conducción, la corriente anódica

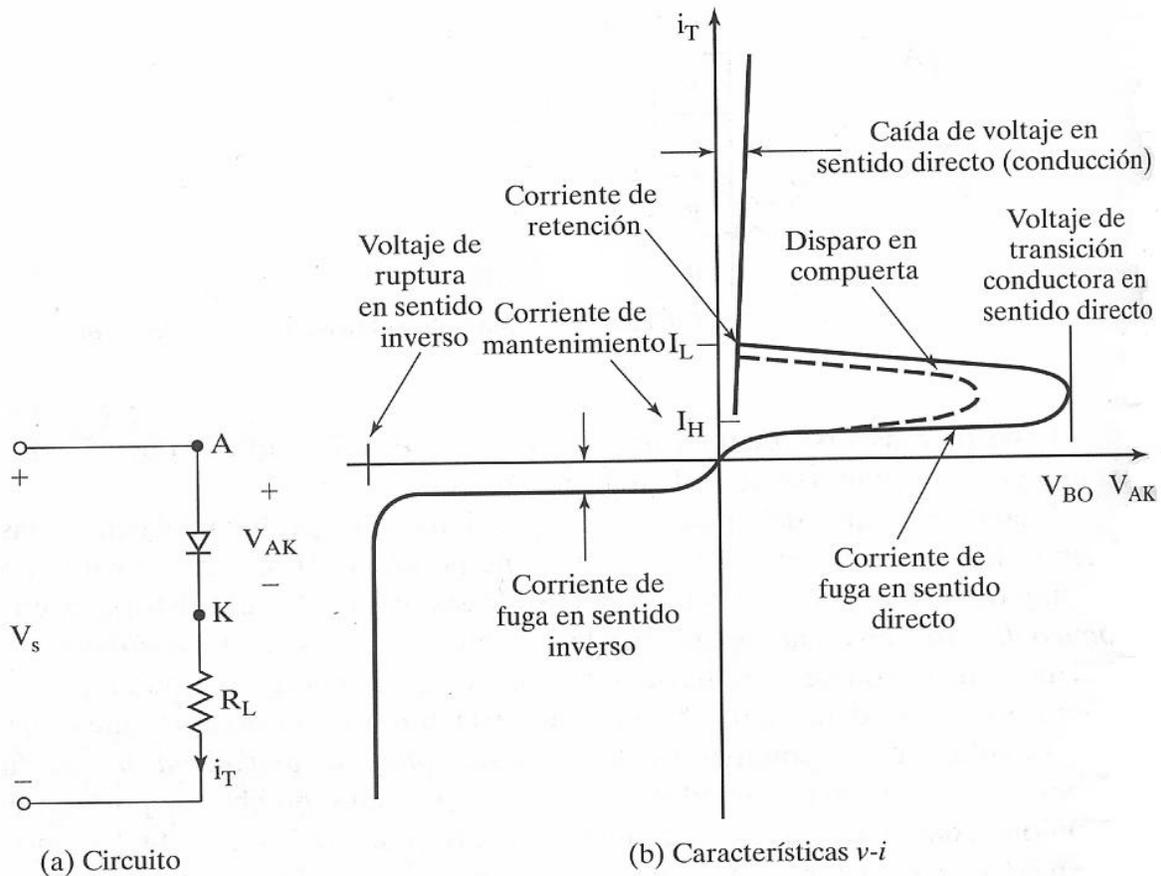
**Figura 4.17**

Secciones transversales de un tiristor.

se limita por una impedancia externa o una resistencia externa  $R_L$  como se ve en la figura 4.18a. La corriente anódica debe ser mayor que un valor llamado corriente de retención  $I_L$  para mantener el flujo necesario de portadores a través de la unión; en caso contrario, el dispositivo regresa a la condición de bloqueo, cuando se reduce el voltaje de ánodo a cátodo. La corriente de retención  $I_L$  es la corriente anódica mínima necesaria para mantener el tiristor en estado de encendido, inmediatamente después de haberse activado y retirar la señal de compuerta. En la figura 4.18b se ve una característica  $v-i$  típica de un tiristor.

Una vez que un tiristor conduce se comporta como un diodo conductor y no hay control sobre el dispositivo y este continúa conduciendo porque no hay capa de transición en la unión  $J_2$  debido a los movimientos libres de los portadores. Sin embargo, si la corriente en sentido directo del ánodo se reduce de un nivel conocido como corriente de mantenimiento  $I_H$ , se desarrolla una región de transición alrededor de la unión  $J_2$  debido al número reducido de portadores y el tiristor está en el estado de bloqueo. La corriente de mantenimiento está en el orden de miliamperes y es menor que la corriente de retención  $I_L$ . Es decir  $I_L > I_H$ . La corriente de mantenimiento  $I_H$  es la corriente mínima de ánodo para mantener al tiristor en el estado de encendido. La corriente de mantenimiento es menor que la corriente de retención.

Cuando el voltaje catódico es positivo con respecto al ánodo, la unión  $J_2$  tiene polarización directa, pero las uniones  $J_1$  y  $J_3$  tienen polarización inversa. Es como dos diodos conectados en serie con voltaje en sentido inverso a través de ellos. El tiristor está en el estado de bloqueo en sentido inverso y pasa por él una corriente de fuga, llamada corriente inversa  $I_R$ .

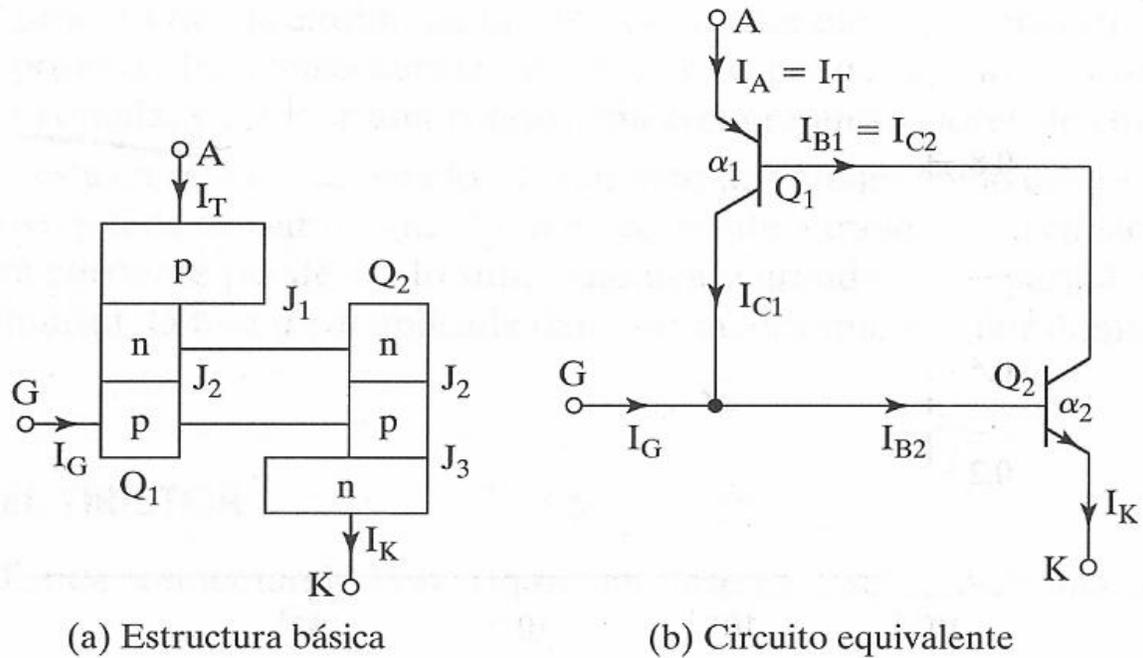


**Figura 4.18**  
Circuito de tiristor y características  $v-i$ .

Un tiristor se puede encender aumentando el voltaje  $V_{AK}$  en sentido directo a más de  $V_{BO}$ , pero ese encendido podría ser destructivo. En la práctica, el voltaje en sentido directo se mantiene menor que  $V_{BO}$  y el tiristor se activa aplicando un voltaje positivo entre su compuerta y su cátodo. Una vez encendido el tiristor por una señal de compuerta, y siendo mayor su corriente anódica que la corriente de retención, el dispositivo continúa conduciendo, por retroalimentación positiva, aun cuando se retire la señal de compuerta. Un tiristor es un dispositivo de retención.

#### Modelo del tiristor empleando dos transistores.

La acción regenerativa o de retención debida a la retroalimentación positiva se puede demostrar con un modelo de tiristor hecho con dos transistores. Se puede considerar que un tiristor equivale a dos transistores complementarios, un transistor  $pnp$ ,  $Q_1$ , y otro transistor  $npn$ ,  $Q_2$ , como se ve en la figura 4.19a. El modelo del circuito equivalente se ve en la figura 4.19b.



**Figura 4.19**  
Modelo de dos transistores para el tiristor.

La corriente  $I_C$  del colector de un tiristor se relaciona, en general, con la corriente del emisor  $I_E$  y la corriente de fuga de la unión colector-base,  $I_{CBO}$ , como sigue:

$$I_C = \alpha I_E + I_{CBO} \quad (4.1)$$

Y la ganancia en corriente en base común se define como  $\alpha = I_C/I_E$ . Para el transistor  $Q_1$ , la corriente del emisor es la corriente anódica  $I_A$ , y la corriente del colector  $I_{C1}$  se puede determinar con la ecuación:

$$I_{C1} = \alpha_1 I_A + I_{CBO1} \quad (4.2)$$

Donde  $\alpha_1$  es la ganancia en corriente e  $I_{CBO1}$  es la corriente de fuga para  $Q_1$ . De igual modo para el transistor  $Q_2$ , la corriente del colector  $I_{C2}$  es

$$I_{C2} = \alpha_2 I_K + I_{CBO2} \quad (4.3)$$

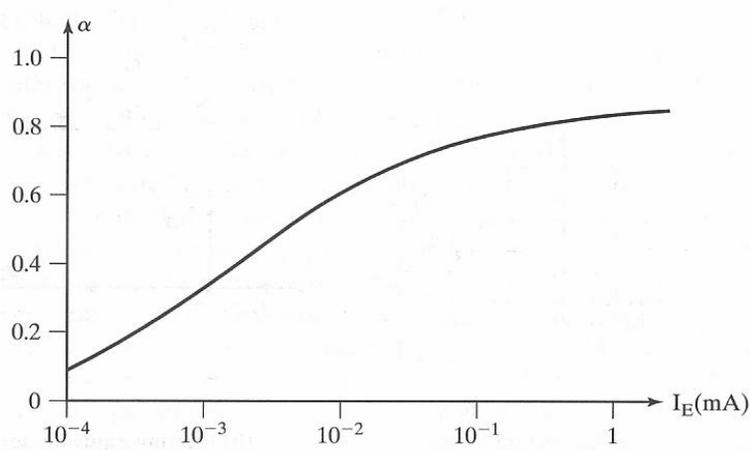
Donde  $\alpha_2$  es la ganancia en corriente e  $I_{CBO2}$  es la corriente de fuga para  $Q_2$ . Se combinan  $I_{C1}$  e  $I_{C2}$  para obtener

$$I_A = I_{C1} + I_{C2} = \alpha_1 I_A + I_{CBO1} + \alpha_2 I_K + I_{CBO2} \quad (4.4)$$

Si la corriente de disparo, control, de compuerta es  $I_G$ ,  $I_K = I_A + I_G$ , y despejando  $I_A$  de la ecuación (4.4), se obtiene

$$I_A = \frac{\alpha_2 I_G + I_{CBO1} + I_{CBO2}}{1 - (\alpha_1 + \alpha_2)}$$

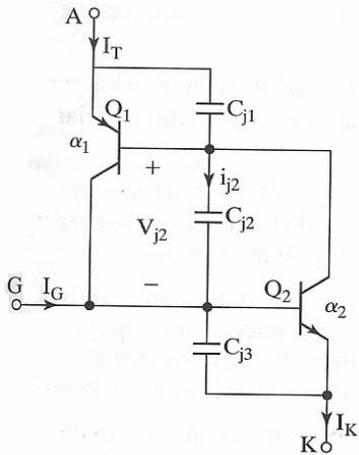
La ganancia en corriente  $\alpha_1$  varía con la corriente de emisor  $I_A = I_E$ ;  $\alpha_2$  varía con  $I_K = I_A + I_G$ . En la figura 4.20 se ve una variación típica de la ganancia en corriente  $\alpha$  con la corriente del emisor  $I_E$ . Si aumenta de forma repentina la corriente de compuerta  $I_G$ , por ejemplo de 0 a 1 mA, aumenta de inmediato la corriente anódica  $I_A$ , que hace aumentar más  $\alpha_1$  y  $\alpha_2$ . La ganancia en corriente  $\alpha_2$  depende de  $I_A$  e  $I_G$ . El aumento en los valores de  $\alpha_1$  y  $\alpha_2$  hace aumentar más  $I_A$ . En consecuencia, hay un efecto regenerativo, o de retroalimentación positiva. Si  $(\alpha_1 + \alpha_2)$  tiende a la unidad, el denominador de la ecuación (4.5) tiende a cero, resulta un valor grande de la corriente anódica  $I_A$ , y el tiristor se enciende con una pequeña corriente de compuerta.



**Figura 4.20**

Variación típica de la ganancia de corriente en función de la corriente de emisor.

Bajo condiciones transitorias, las capacitancias de las uniones  $pn$ , que se ven en la figura 4.21, influyen sobre las características del tiristor. Si un tiristor está en estado de bloqueo, un voltaje que aumente con rapidez aplicado a través del dispositivo causaría un flujo alto de corriente a través de los capacitores de la unión.

**Figura 4.21**

Modelo de dos transistores para tiristor en estado transitorio.

La corriente por el capacitor  $C_{j2}$  se puede expresar con la ecuación

$$i_{j2} = \frac{d(q_{j2})}{dt} = \frac{d(C_{j2}V_{j2})}{dt} = V_{j2} \frac{d(C_{j2})}{dt} + C_{j2} \frac{d(V_{j2})}{dt}$$

En la que  $C_{j2}$  y  $V_{j2}$  son la capacitancia y el voltaje de la unión  $J_2$ , respectivamente;  $q_{j2}$  es la carga en la unión. Si la rapidez de aumento de la tasa  $dv/dt$  es grande, entonces  $i_{j2}$  sería grande, y eso causaría mayores corrientes de fuga  $I_{CBO1}$  e  $I_{CBO2}$ . De acuerdo con la ecuación (4.5), los valores suficientemente altos de  $I_{CBO1}$  e  $I_{CBO2}$  pueden causar que  $(\alpha_1 + \alpha_2)$  tiendan a la unidad, y causen un encendido no deseado del tiristor. Sin embargo, una corriente grande a través de los capacitores de la unión también puede dañar al dispositivo.

#### Activación del tiristor.

Un tiristor se enciende aumentando la corriente anódica. Esto se hace de las siguientes maneras.

**Térmica.** Si la temperatura de un tiristor es alta, hay un aumento en la cantidad de pares electrón – hueco, que aumentan las corrientes de fuga. Este aumento en las corrientes hace aumentar a  $\alpha_1$  y  $\alpha_2$  (modelando el tiristor empleando transistores,  $\alpha_1$  es la ganancia en corriente para Q1 y  $\alpha_2$  es la ganancia en corriente para Q2). Debido a la acción regenerativa,  $\alpha_1 + \alpha_2$  pueden tender a la unidad, y el tiristor se puede activar. Este tipo de activación puede causar avalancha térmica, y en el caso normal se evita.

**Luz.** Si se deja incidir luz en las uniones de un tiristor, aumentan los pares electrón – hueco y el tiristor puede activarse. Los tiristores activados con luz se encienden dejando que la luz incida sobre la oblea de silicio.

**Alto voltaje.** Si el voltaje en sentido directo, de ánodo a cátodo, es mayor que el voltaje de ruptura en sentido directo  $V_{BO}$ , pasa una corriente de fuga suficiente para iniciar la activación regenerativa. Esta clase de activación es destructiva y se debe evitar.

$dv/dt$ . Si la rapidez de aumento en el voltaje ánodo – cátodo es alta, la corriente de carga de las uniones capacitivas puede bastar para activar al tiristor. Un valor alto de la corriente de carga puede dañar al tiristor, y se debe proteger contra una alta tasa  $dv/dt$ . Los fabricantes especifican la tasa  $dv/dt$  máxima admisible en sus tiristores.

Corriente de compuerta. Si un tiristor está polarizado en sentido directo, la inyección de corriente de compuerta al aplicar voltaje de compuerta positivo, entre las terminales de la compuerta y el cátodo, enciende al tiristor. Al aumentar la corriente de compuerta, disminuye el voltaje de bloqueo en sentido directo.

Se deben tener en cuenta los siguientes puntos para diseñar el circuito de control de compuerta.

1. La señal de compuerta debe retirarse después de que haya encendido el tiristor. Una señal de control continua aumentaría la pérdida de potencia en la unión de la compuerta.
2. Aunque el tiristor está polarizado en sentido inverso, no debe haber señal de compuerta, porque de lo contrario puede fallar a causa de un aumento en la corriente de fuga.
3. El ancho de pulso en la compuerta  $t_G$  puede ser mayor que el tiempo necesario para que la corriente anódica aumente hasta el valor de la corriente de retención  $I_H$ . En la práctica, el ancho  $T_G$  del pulso se hace, en el caso normal, mayor que el tiempo de activación  $t_{ON}$  del tiristor.

### **Tipos de tiristores.**

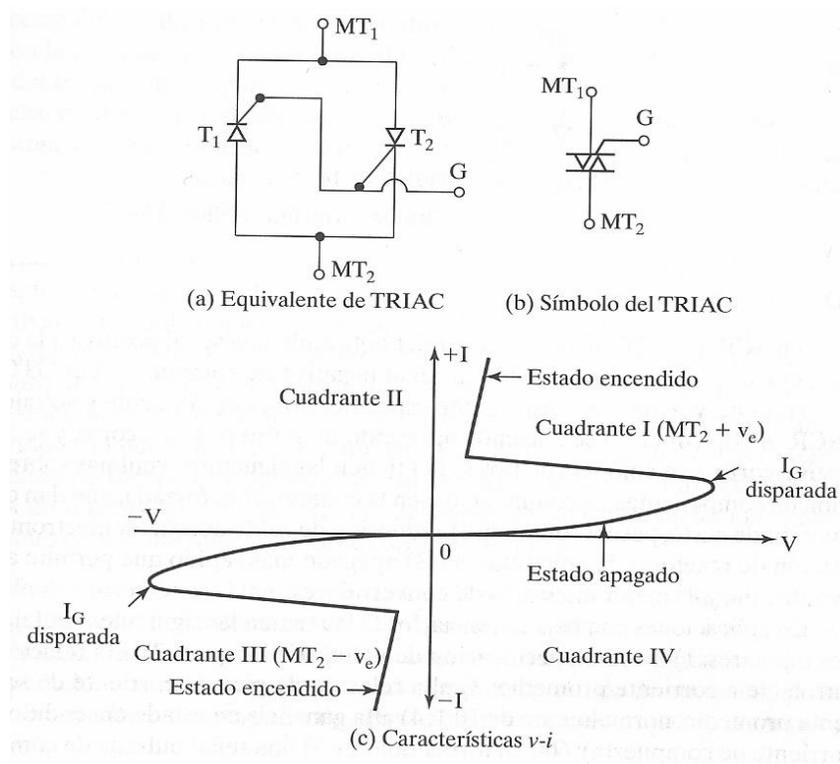
Existen varios tipos de tiristores, dependiendo de la construcción física y el comportamiento en el encendido y el apagado, se pueden clasificar los tiristores, en forma amplia, en 13 categorías:

1. Tiristores controlados por fase (SCR de sus siglas en inglés silicon controlled rectifier).
2. Tiristores bidireccionales controlados por fase (BCT, bidirectional phase-controlled thyristor).
3. Tiristores de conmutación rápida (ASCR).
4. Rectificadores controlados de silicio fotoactivados (LASCR, light-activated SCR).
5. Tiristores de trío bidireccional (TRIAC).
6. Tiristores de conducción en sentido inverso (RCT, reverse-conducting thyristor).
7. Tiristores apagados por compuerta (GTO).
8. Tiristores controlados por FET (FET-CTH, de FET-controlled thyristor).
9. Tiristores de apagado por MOS (MTO, MOS turn off).
10. Tiristores de apagado (control) por emisor (ETO, emitter turn off).
11. Tiristores conmutados por compuerta integrada (IGCT, integrated gate-commutated thyristor).
12. Tiristores controlados por MOS (MCT, de MOS-controlled thyristor).
13. Tiristores de inducción estática (SITH, static induction thyristor).

### **Tiristores de trío bidireccional (TRIACS).**

Un TRIAC puede conducir en ambas direcciones, y se usa normalmente para control por fase (por ejemplo, en controladores de AC). Se puede considerar como dos SCR conectados en antiparalelo con una conexión de compuerta común, como se ve en la figura 4.22a. Las características  $v-i$  se ven en la figura 4.22c.

Como un TRIAC es un dispositivo bidireccional, no se puede decir que sus terminales sean ánodo y cátodo. Si la terminal MT2 es positiva con respecto a la terminal MT1, el TRIAC se puede encender aplicando una señal positiva entre la compuerta G y la terminal MT1. Si la terminal MT2 es negativa con respecto a la terminal MT1, se enciende aplicando una señal negativa entre la compuerta G y la terminal MT1. No es necesario tener las dos polaridades de señal de compuerta, y un TRIAC se puede encender con una señal de compuerta que puede ser positiva o negativa. En la práctica las sensibilidades varían de uno a otro cuadrante, y los TRIAC se suelen operar en el cuadrante I (voltaje de compuerta y corriente de compuerta positivos), o en el cuadrante III (voltaje y corriente de compuerta negativos).



**Figura 4.22**  
Características de un TRIAC.

### Protección contra $di/dt$ .

Un tiristor requiere un tiempo mínimo para repartir por igual la conducción de la corriente en las uniones. Si la rapidez de aumento de la corriente anódica es muy alta, en comparación con la velocidad de repartición de un proceso de activación, puede presentarse un calentamiento

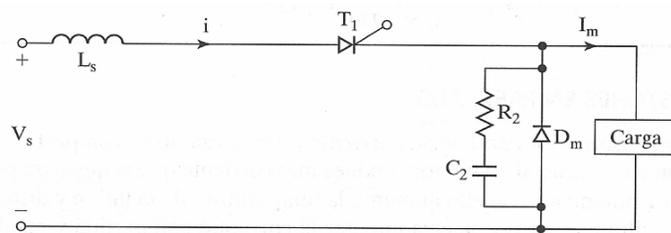
localizado, o “punto caliente” debido a la alta densidad de corriente, y el dispositivo puede fallar como resultado de una temperatura excesiva.

Se deben proteger los dispositivos prácticos contra una alta tasa  $di/dt$ . Por ejemplo veamos el circuito de la figura 4.23. Bajo operación de estado permanente,  $D_m$  conduce cuando el tiristor  $T_1$  está apagado. Si se dispara  $T_1$  cuando  $D_m$  todavía está conduciendo, la tasa  $di/dt$  puede ser muy alta, y solo la limita la inductancia parásita del circuito.

En la práctica, la tasa  $di/dt$  se limita agregando un inductor  $L_s$  en serie, como se ve en la figura 4.23. La tasa  $di/dt$  es sentido directo es

$$\frac{di}{dt} = \frac{V_s}{L_s}$$

Donde  $L_s$  es la inductancia en serie, que incluye cualquier inductancia parásita.



**Figura 4.23**

Circuito de conmutación para tiristor, con inductores limitadores de la tasa  $di/dt$ .

### Protección contra $dv/dt$ .

Si se cierra el interruptor  $S_1$  en la figura 4.24a cuando  $t=0$ , se puede aplicar un escalón de voltaje a través del tiristor  $T_1$ , y puede ser que la tasa  $dv/dt$  sea suficientemente alta como para encender el dispositivo. Se puede limitar la tasa  $dv/dt$  conectando el capacitor  $C_s$  como se ve en la figura 4.24a. Cuando el tiristor  $T_1$  se activa, la corriente de descarga del capacitor se limita con el resistor  $R_s$  como se ve en la figura 4.24b.

Con un circuito RC, llamado circuito amortiguador, el voltaje en el tiristor aumenta en forma exponencial, como se ve en la figura 4.24c, y la tasa  $dv/dt$  del circuito se puede determinar en forma aproximada con

$$\frac{dv}{dt} = \frac{0.632V_s}{\tau} = \frac{0.632V_s}{R_s C_s}$$

El valor de la constante de tiempo del amortiguador,  $\tau=R_s C_s$ , se puede determinar con esta ecuación, para un valor conocido de la tasa  $dv/dt$ . El valor de  $R_s$  se determina a partir de la corriente de descarga  $I_{TD}$ .

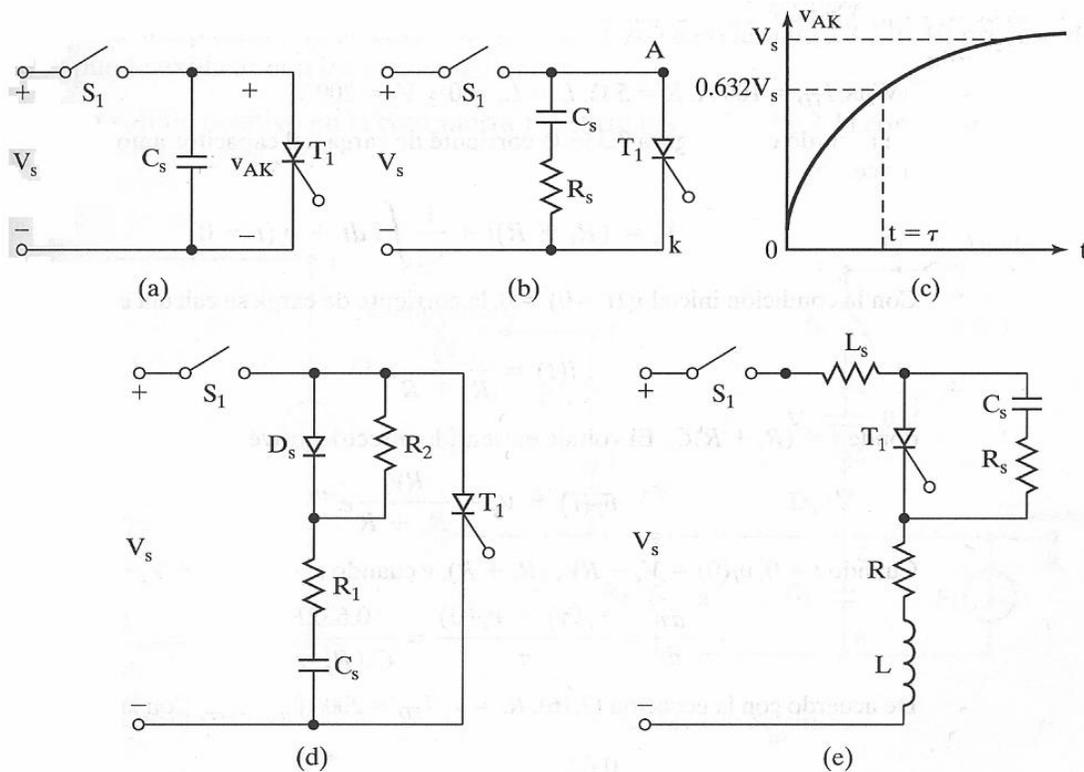
$$R_s = \frac{V_s}{I_{TD}}$$

Es posible usar más de un resistor para la tasa  $dv/dt$  y descargar, como se ve en la figura 4.24d. La tasa  $dv/dt$  se limita con  $R_1$  y  $C_s$ . La suma  $(R_1+R_2)$  limita la corriente de descarga, de tal modo que

$$I_{TD} = \frac{V_s}{R_1 + R_2}$$

La carga puede formar un circuito en serie con la red amortiguadora, como se ve en la figura 4.24e. La relación de amortiguamiento  $\delta$  de una ecuación de segundo orden es

$$\delta = \frac{\alpha}{\omega_0} = \frac{R_s + R}{2} \sqrt{\frac{C_s}{L_s + L}}$$



**Figura 4.24**  
Circuitos de protección contra la tasa  $dv/dt$ .

Donde  $L_s$  es la inductancia parasita y  $L$  y  $R$  son la inductancia y la resistencia de la carga, respectivamente.

Para limitar el sobredisparo de voltaje pico aplicado a través del tiristor, se usa una relación de amortiguamiento en el intervalo de 0.5 a 1.0. Si la inductancia de la carga es alta, lo cual suele ser el caso,  $R_s$  puede ser alta y  $C_s$  puede ser pequeña, para conservar el valor deseado de la relación de amortiguamiento. Un valor grande de  $R_s$  reduce la corriente de descarga, y un valor pequeño de  $C_s$  reduce la pérdida por el amortiguador. Se deben analizar con detenimiento los circuitos de la figura 4.24 para determinar el valor requerido de la relación de amortiguamiento que limite la tasa  $dv/dt$  al valor deseado. Una vez conocida la relación de amortiguamiento, se pueden determinar  $R_s$  y  $C_s$ . En el caso normal, se usa la misma red o amortiguador RC para protección contra la tasa  $dv/dt$  y para suprimir el voltaje transitorio debido al tiempo de recuperación en sentido inverso.

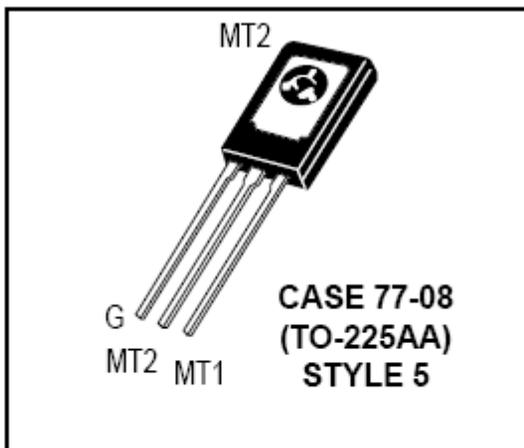
#### Características del TRIAC seleccionado.

En la tabla 4.8 se muestran los máximos valores que soporta el TRIAC 2N6071A, usado en este proyecto.

MAXIMUM RATINGS: ( $T_J=25^\circ\text{C}$ unless otherwise noted)		2N6071	2N6073	2N6075	
		2N6071A	2N6073A	2N6075A	
	SYMBOL	2N6071B	2N6073B	2N6075B	UNITS
Peak Repetitive Off-State Voltage	$V_{DRM}, V_{RRM}$	200	400	600	V
RMS On-State Current ( $T_C=85^\circ\text{C}$ )	$I_{T(RMS)}$		4.0		A
Peak One Cycle Surge (60Hz, $T_J=110^\circ\text{C}$ )	$I_{TSM}$		30		A
$I^2t$ Value for Fusing ( $t=8.3\text{ms}$ )	$I^2t$		3.7		$\text{A}^2\text{s}$
Peak Gate Power ( $T_C=85^\circ\text{C}$ )	$P_{GM}$		10		W
Average Gate Power ( $t=8.3\text{ms}$ , $T_C=85^\circ\text{C}$ )	$P_{G(AV)}$		0.5		W
Peak Gate Voltage ( $T_C=85^\circ\text{C}$ )	$V_{GM}$		5.0		V
Storage Temperature	$T_{stg}$		-40 to +150		$^\circ\text{C}$
Junction Temperature	$T_J$		-40 to +110		$^\circ\text{C}$
Thermal Resistance	$\theta_{JC}$		3.5		$^\circ\text{C}/\text{W}$
Thermal Resistance	$\theta_{JA}$		75		$^\circ\text{C}/\text{W}$
Maximum Lead Temperature	$T_L$		260		$^\circ\text{C}$

**Tabla 4.8**

Máximos valores que soporta el TRIAC 2N6071A.



**Figura 4.25**  
TRIAC 2N6071A.

#### 4 – 6 Optoaisladores.

Se trata de un CI encapsulado que contiene tanto un LED infrarrojo como un fotodetector, como un diodo de silicio, un transistor par Darlington o un SCR. La respuesta de longitud de onda de cada dispositivo se adapta para ser lo más similar posible para permitir el mayor nivel de acoplamiento posible. Estos dispositivos se encuentran diseñados con tiempos de respuesta tan pequeños que pueden utilizarse para transmitir datos en el rango de los megahertz.

El uso de éste dispositivo proporciona un aislamiento eléctrico completo entre el circuito eléctrico conectado a la entrada del aislador y el circuito conectado a su salida. Este aislamiento es útil para reducir el efecto de la interferencia eléctrica sobre la transmisión de la señal en un sistema y, por tanto, los optoaisladores suelen utilizarse en el diseño de sistemas digitales. También se usan en el diseño de instrumentos médicos para reducir el riesgo de descargas eléctricas en pacientes.

En este proyecto se utilizaron optoaisladores MOC3010, que tienen salida a TRIAC, para aislar el circuito de reconocimiento de la señal proveniente del DSP y la etapa de potencia. En la figura 4.26 se presenta el diagrama de conexiones utilizado.

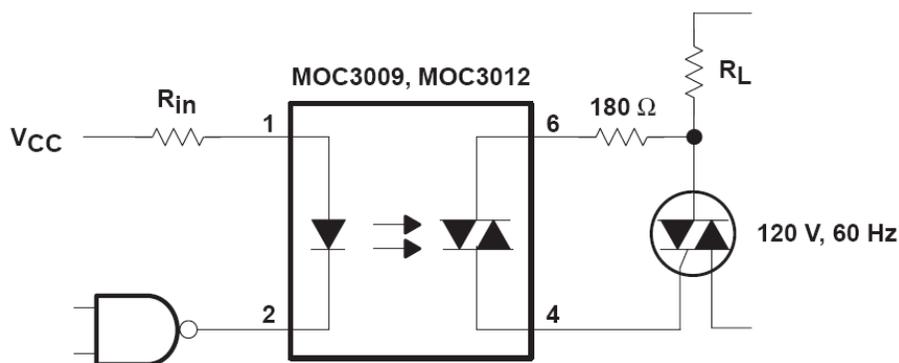


Figura 4.26

MOC3010 usado en el proyecto, y forma como se conecta a los pines del microcontrolador y al TRIAC que controla el encendido del foco.

En las terminales número uno de los optoaisladores están conectados los pines del microcontrolador que se prenden o apagan según lo reconocido, las terminales numero dos van conectadas a la misma tierra del microcontrolador, a la salida de los optoaislador se tienen conectados TRIACs por los que circulan las corrientes que también lo hacen por los focos de 100 W @ 127 VAC ( $R_L$ ).

#### **4 – 7 Esquemático final.**

En la figura 4.27 se muestra el esquemático final del circuito que se usa para reconocer la señal proveniente del DSP y prender los focos, además de un circuito, que se usó para amplificar la señal de voz que entra al DSP, este circuito ocupa un amplificador operacional LM386, recomendable para amplificar señales de audio.

En la figura 4.28 se observa el diagrama de conexiones de los optoacopladores, TRIACs y focos y finalmente como estos están conectados a la línea de 120 VRMS.

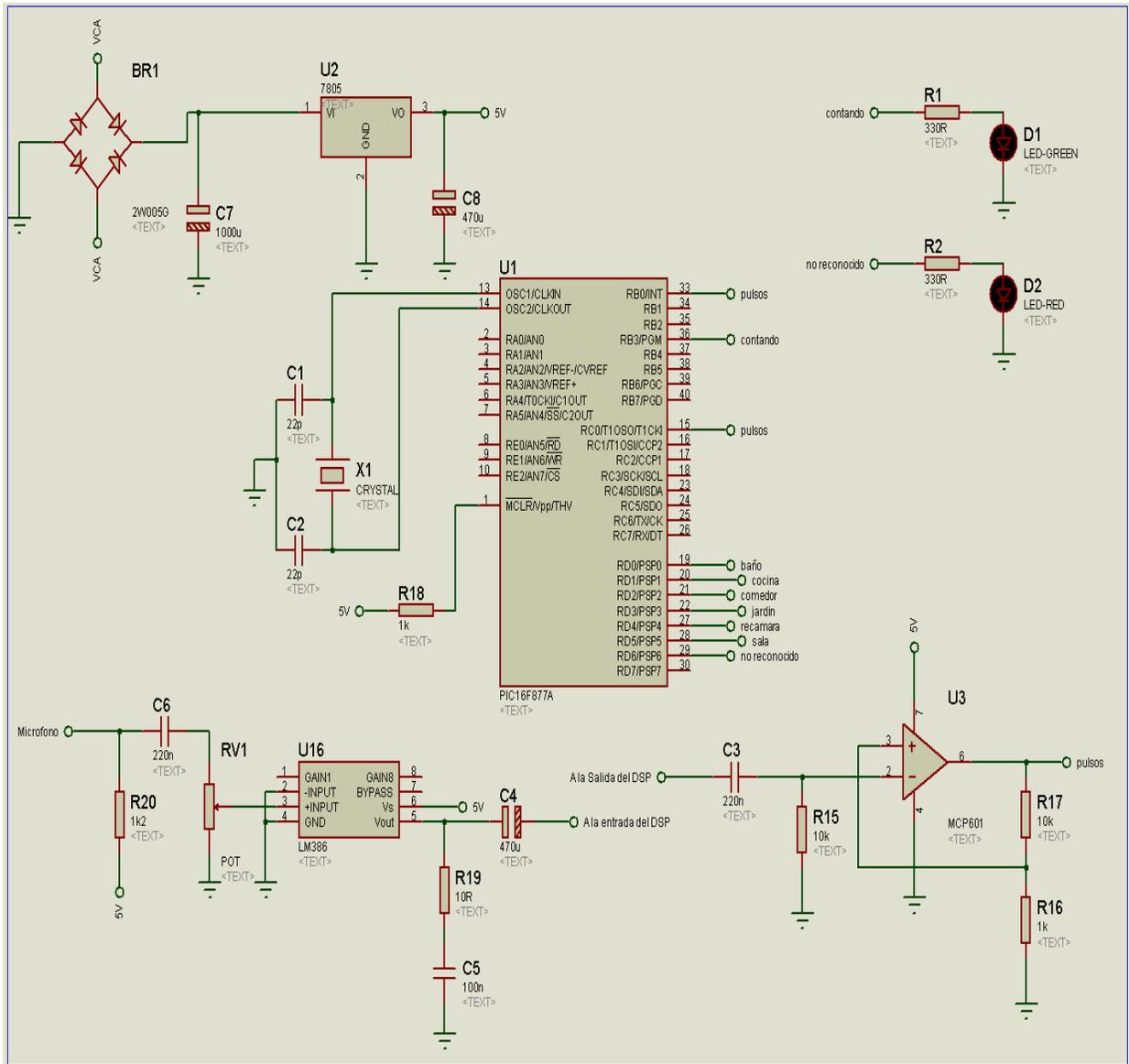
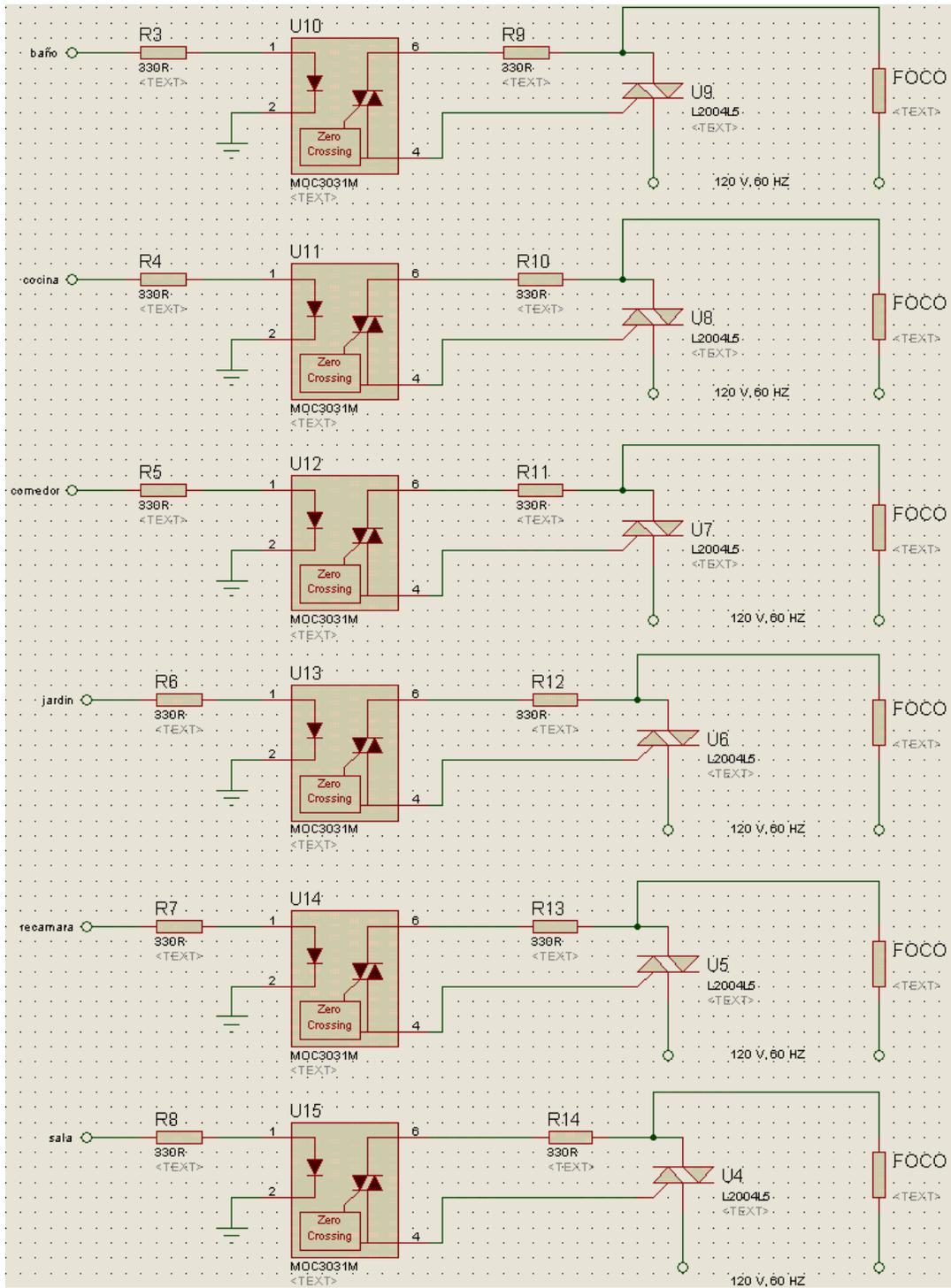


Figura 4.27  
Esquemático final del circuito.



**Figura 4.28**

Diagrama de conexiones de los optocopladores con los TRIACS y los focos.

## Capítulo 5

### Conclusiones

Se logró implementar un sistema que reconoce comandos de voz relacionados con la iluminación de una casa habitación y que interactúa con otro sistema capaz de identificar una señal emitida por el sistema de reconocimiento de voz, y en base a ésta prender o apagar focos de la misma tensión y corriente de los que se utilizan en una casa.

Antes de hacer este proyecto, se tenía armado un amplificador para la señal del micrófono en una protoboard, y se tenía un 30% de reconocimientos falsos debidos a que la señal amplificada que se enviaba al DSP era muy ruidosa. Cuando se hizo el circuito impreso de este proyecto, se integro este amplificador, logrando reducir considerablemente el ruido de la señal amplificada, lo que tiene como consecuencia que todo el sistema completo tenga un 95% de reconocimientos exitosos.

Cuando el DSP ha reconocido en comando de voz y ha mandado la señal sinusoidal de un segundo de duración, este está listo para reconocer un nuevo comando. En este momento el PIC apenas ha terminado de contar los pulsos que le llegaron, pero aun no ha determinado de que instrucción se trata. A pesar de esto, con el reloj de 4 MHz usado, el proceso de determinar la instrucción recibida es suficientemente rápido, y no es necesario esperar un tiempo significativo después del segundo ocupado mientras el DSP le manda la señal sinusoidal al PIC, para darle el siguiente comando al DSP.

Si se modificara en el programa del DSP la duración de la señal sinusoidal a un tiempo menor, también se tendría que modificar el programa del PIC, pero no considerablemente. En el caso de que se quisiera una velocidad mayor también se podría hacer que el PIC trabajara con un reloj de hasta 16 MHz, teniéndose que hacer ligeros cambios en el programa del PIC. De estas dos formas sería posible aumentar la velocidad de respuesta del sistema hasta un punto donde se pudieran dar los comandos de voz con un espaciamiento menor a un segundo.

Este sistema se puede usar con un micrófono inalámbrico para dar las instrucciones. También es importante mencionar que el sistema que se tiene, podría controlar no solo la iluminación de una casa, sino también el encendido de aparatos que tengan motores o alguna otra clase de actuadores, como para prender la bomba de agua de una casa, abrir y cerrar puertas/ventanas, etc. Para esto se requiere grabar el comando de voz apropiado, extraer sus características e implementarlo en el DSP, además de calcular bien las protecciones para el TRIAC o incluso cambiar el TRIAC por uno que soporte más corriente.

Sin embargo, este proyecto tiene aun muchos aspectos que se pueden mejorar, ya que se podría meter todo el proyecto en un solo dsPIC que se encargara de hacer el procesamiento de voz y que además en base al comando reconocido prendiera o apagara pines del mismo chip para después acoplarlo a una etapa de potencia que finalmente prenda o apague los focos.

Otro de los puntos que se puede mejorar es hacer que este sistema reconocedor de voz sea multilocutor, ya que en una aplicación para la vida diaria esto sería necesario. Para lograr lo anterior sería necesario una cantidad de memoria considerablemente mayor a la usada en este proyecto.

Una vez hechas las mejoras propuestas, creo que es factible comercializar este producto, ya que proporciona comodidad y podría ser de gran ayuda para personas con limitaciones físicas, permitiéndoles controlar varios aparatos sin la necesidad de caminar o apretar botones.

Con este proyecto terminé de aprender a usar los PIC's, que realmente se pueden volver imprescindibles en la vida de un ingeniero eléctrico electrónico, ya que además de ser baratos, son relativamente fáciles de usar y se pueden encontrar en distintas clases de encapsulados. Para este proyecto utilizamos el encapsulado PDIP, con el que es posible hacer pruebas en protoboard y al hacer el circuito impreso casero, las pistas no quedan tan cercanas unas con otras como quedarían con algún otro encapsulado, lo cual es una ventaja a la hora de soldar los dispositivos.

Es satisfactorio poder utilizar los conocimientos adquiridos en varias de las materias cursadas durante la carrera en un problema planteado al inicio del Seminario de Titulación. Para este proyecto se utilizaron tanto la electrónica digital como la analógica; el trabajo no hubiese podido ser ejecutado sin usar ambas, iniciando con la señal analógica que es la señal de voz y terminando con el uso de TRIACS.

## Bibliografía.

### Capítulo 2.

- Lawrence Rabiner y Biing-Hwang Juang: Fundamentals of Speech Recognition, Prentice Hall, first edition.
- Apuntes de la materia de Procesamiento de Voz.
- Prácticas de laboratorio de la materia Procesamiento de voz.
- [http://es.wikipedia.org/wiki/Cualidades\\_del\\_sonido](http://es.wikipedia.org/wiki/Cualidades_del_sonido)

### Capítulo 3.

- Hoja de datos de la familia de DSP's TMS320C6000.
- [http://www.kanecomputing.co.uk/dsk\\_c6711.htm](http://www.kanecomputing.co.uk/dsk_c6711.htm).
- <http://focus.ti.com/docs/toolsw/folders/print/tmds320006711.html>.
- [http://es.wikipedia.org/wiki/Procesador\\_digital\\_de\\_se%C3%B1al](http://es.wikipedia.org/wiki/Procesador_digital_de_se%C3%B1al).

### Capítulo 4.

- James W. Nilsson y Susan A. Riedel: Circuitos eléctricos, Prentice Hall, séptima edición.
- Robert F. Coughlin y Frederick Driscoll: Amplificadores Operacionales y circuitos integrados lineales, Prentice Hall, quinta edición.
- José María Angulo, Susana Romero e Ignacio Angulo: Microcontroladores PIC: diseño practico de aplicaciones 2ª Parte, McGraw Hill, segunda edición.
- Robert L. Boylestad y Louis Nashelsky: Electrónica: teoría de circuitos y dispositivos electrónicos, Prentice Hall, octava edición.
- Adel S. Sedra y Kenneth C. Smith: Circuitos microelectrónicos, McGraw-Hill, quinta edición.
- Muhammad H. Rashid: Electrónica de potencia: circuitos, dispositivos y aplicaciones, Prentice Hall, tercera edición.
- Hojas de datos de los siguientes circuitos integrados o dispositivos.
  - PIC16F877A.
  - MOC3010.
  - 2N6071A.
- <http://es.wikipedia.org/wiki/Microcontrolador>.
- <http://es.wikipedia.org/wiki/Interrupcion>