



**Universidad del Azuay**

**Facultad de Ciencia y Tecnología**

**Escuela de Ingeniería Electrónica**

**MANDO REMOTO DE UNA CASA VIA TELEFÓNICA**

**Trabajo de graduación previo a la obtención del título de  
Tecnólogo Electrónico**

**Autor: Andrés Argudo Amores.**

**Director: Dr. Hugo Torres**

**Cuenca, Ecuador  
2010**

## DEDICATORIA

A mi Señor, Jesús, quien me dio la fe, la fortaleza, la salud y la esperanza para terminar este trabajo.

A mi adorado hijo Carlitos Argudo y su dulce mama “Mary”, quienes me prestaron el tiempo que les pertenecía para terminar y me motivaron siempre con sus besos y abrazos.

A mis padres Carlos Argudo Vicuña que está en cielo ¡el siempre anhelo este momento! PROMESA CUMPLIDA PAPA, A mi madre Eva Amores Aguirre quien me enseñó y me demostró que aun no estoy vencido. Mi triunfo es el de ustedes,

A los que nunca dudaron que lograría este triunfo: mis hermanos Mauricio y Angélica.

Que Dios los bendiga a todos ustedes.

## **AGRADECIMIENTO**

Agradezco a mi Director de Tesis el Doctor Hugo Torres, por haber aceptado dirigir el presente trabajo con toda su capacidad y profesionalismo, digno de admiración y respeto, que Dios siga iluminando su prodigiosa mente al servicio de la juventud.

## **Resumen**

Este Mando Remoto de Casa vía Telefónica utiliza el control a distancia, que presenta muchas opciones y beneficios para los usuarios residenciales.

Dentro de sus principales ventajas se encuentran la comodidad que ofrece el no tener que desplazarse hasta algún sitio para hacer alguna tarea, ya que utiliza una línea telefónica normal; considerando factores de que se pueden realizar labores peligrosas sin acercarse a la vivienda y la rapidez con que se pueden tomar acciones respecto a alguna condición específica, entre otros. Además de incorporar un medio de control y visualización alternativa por Red LAN o WAN.

El proyecto realizado ofrece una infinidad de aplicaciones las que dependerán de la imaginación y necesidades de cada aplicación ya que puede ser modificado para alternativas comerciales e industriales.

## **Abstract**

This Telephonic House Remote Control offers many options and benefits for residential user.

Among its main advantages they are the comfort offered by not having to go certain places in order to perform certain tasks as it uses a normal phone line; as well as factors such as dangerous tasks can be taken with respect to some specific condition, amongst other. Furthermore, it incorporates a means of control and alternative visualization for LAN or WAN networks..

This project offers an infinite variety of applications, which will depend only on the imagination and the needs of each application given that it can be modified for alternative commercial and industrial uses.

## Índice de Contenidos

Dedicatoria.....	ii
Agradecimiento.....	iii
Resumen.....	iv
Abstract.....	v
Indice de Contenidos.....	vi
<b>INTRODUCCIÓN.....</b>	<b>1</b>

### CAPITULO 1

<b>LA DOMÓTICA SEGURIDAD Y CONFORT .....</b>	<b>2</b>
1.1 ¿Qué es la domótica? .....	3
1.2 Beneficios .....	4
1.3 Aplicaciones.....	5
1.3.1 En el ámbito del ahorro energético.....	5
1.3.3 En el ámbito de la protección personal y patrimonial.....	7
1.3.4 En el ámbito de las comunicaciones.....	8
1.4 Seguridad.....	9
1.5 Sistema de monitorización personal (video-vigilancia) .....	10
1.6 Confort (automatismos).....	12

### CAPITULO 2

<b>GENERALIDADES DE LAS FRECUENCIAS DTMF.....</b>	<b>14</b>
2.1 Funcionamiento general del teléfono.....	13
2.2 DTMF .....	13
2.3 Tipos de códigos.....	14
2.4 Formas de onda.....	15
2.5 Parámetros y valores .....	17

2.6 Estructura interna del MT8870 .....	18
2.7 Funcionamiento y Configuraciones del MT8870 .....	21

### **CAPITULO 3**

<b>3.0 CONTROL DE SEÑALES POR PUERTOS DE LA PC.....</b>	<b>23</b>
3.1 Puertos en Serie. ....	23
3.3 Comunicaciones serie asíncronas.....	23
3.3.1 Forma de puerto serial .....	24
3.4 El Conector DB9 del PC.....	24
3.4.1 Características .....	26
3.5 Ubicación en el sistema informativo. ....	26
3.6 Puerto paralelo.....	26
3.6.1 Forma de puerto paralelo. ....	27
3.7 Puerto USB (Universal Serial Bus).....	27
3.7.1 Características del Puerto USB.....	28
3.7. 2 Forma del Puerto usb.....	28
3.7.3 Ubicación en el Sistema Informático.....	28
3.7.4 Tipos de Transferencia.....	29
3.8 EL PIC 16F628A .....	29
3.8.1 Arquitectura del PIC 16F628A.....	30
3.8.2 Características Generales. ....	31
3.8.3 Diagrama de Pines y Funciones.....	32
3.9 Fuente de Alimentación.....	33
3.10 Detector de Señal de Timbre Telefónico. ....	33
3.11 EL PIC como decodificador de señales BDC.....	35
3.12 circuito Modulador de Tonos .....	36
3.13 Flujograma de Programación del Circuito Control. ....	37
3.14 Código Fuente de Programa de Control del PIC 16F628A .....	42

3.15 Entrada y Salida de datos entre PIC y puerto serial. ....	58
3.16 Interfaz de Entrada y Salida de datos entre el PIC y puerto serial .....	59
3.17 Conexión de un Microcontrolador al puerto serie del PC. ....	59
3.18 Circuito de Salida de Potencia.....	61

## **CAPITULO 4**

<b>PROGRAMACIÓN EN VISUAL C++ .....</b>	<b>63</b>
4.1 Programación orientada a objetos.....	64
4.1.1 Conceptos preliminares.....	64
4.2 El modelo de programación Windows .....	67
4.3 El entorno de desarrollo .....	68
4.4 El concepto de proyecto (workspace).....	69
4.4.1 Posibilidades de programación.....	69
4.5 El generador de aplicaciones. ....	70
4.6 Diseño de software de aplicación de mando de una casa vía telefónica a travez de interfaz serial USB. ....	71
4.6.1 Casos de uso .....	71
4.7 Código fuente del programa en visual c .....	72

## **CAPITULO 5**

<b>DISEÑO Y CONSTRUCCIÓN DEL PROYECTO .....</b>	<b>76</b>
5.1 Listado de componentes .....	78
5.2 Tarjeta electrónica.....	79
5.3 Protocolo de comunicación .....	82
5.4 Claves de activación y desactivación. ....	82
5.5 Conexión de aparatos a 110 v.....	83
5.6 Conexión y mando a travez de escritorio remoto.....	83
5.7 VNC .....	83
5.7.1 Características principales.....	84

5.7.2 Funcionamiento.....	85
5.8 Pruebas realizadas.....	87
<b>CONCLUSIONES.....</b>	<b>88</b>
<b>RECOMENDACIONES .....</b>	<b>89</b>
<b>BIBLIOGRAFÍA .....</b>	<b>90</b>

Argudo Amores Andrés Adrian

Trabajo de Graduación

Dr. Hugo Torres

Noviembre 2010

## **MANDO REMOTO DE UNA CASA VÍA TELEFÓNICA**

### **INTRODUCCIÓN**

El tipo de vida actual obliga a una continua evolución de todo aquello que nos rodea, evolución a la que no escapan nuestros hogares, oficinas e industrias; desde la búsqueda de nuevos estilos de diseño, técnicas de edificación y nuevos materiales hasta la incorporación de tecnologías que gestionen el uso de la energía, aumenten el confort y la seguridad y que permitan un total control desde el interior o el exterior de la casa.

Las necesidades de seguridad, junto con las de cobijo, fueron el origen de la aparición del concepto de vivienda. No obstante, las necesidades actuales son muy diferentes de las de hace cincuenta años. La vivienda es, en la mayoría de los casos, la principal inversión que realiza una familia a lo largo de su vida y las características del entorno social en el que nos movemos hacen que nuestra preocupación principal se centre actualmente en posibles agresiones exteriores, robo, incendio, intrusión, fugas de agua, etc.

Esta inseguridad provocada por la alta peligrosidad actual nos ha obligado a crear nuevos medios de comando mucho más autónomos y controlables desde cualquier medio de transmisión desde un teléfono convencional hasta el más sofisticado equipo celular.

En el entorno doméstico existen una multitud de aplicaciones y servicios destinados a mejorar nuestra seguridad.

Aprovechando los medios de comunicación montados en las redes de telefonía públicas y privadas podremos acceder a controlar los aparatos de una vivienda solucionado el control que podemos tener sobre aparatos que pueden hacer pensar la presencia de un ser humano dentro de casa, convirtiéndola además en una solución para manejar elementos de un hogar en el que se necesite activar luces puertas persianas, electrodomésticos, alarmas en un momento determinado.

## CAPITULO 1

### LA DOMÓTICA SEGURIDAD Y CONFORT

La domótica del latín domus<sup>1</sup> (hogar) es el conjunto de sistemas automatizados de una vivienda que aportan servicios de gestión energética, seguridad, bienestar y comunicación, y que pueden estar integrados por medio de redes interiores y exteriores de comunicación. Se centra en los servicios de bienestar, seguridad y comunicaciones que pueden facilitarse en la vivienda a sus habitantes.

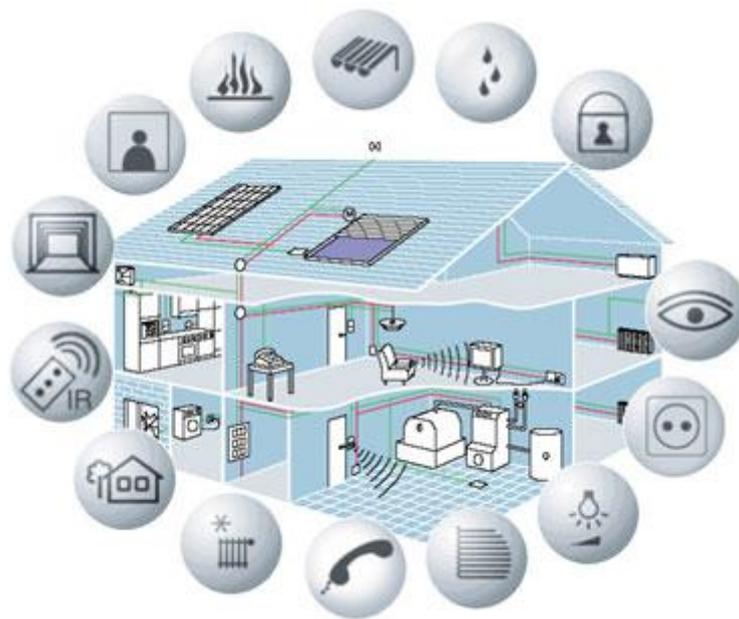


FIG 1: Gestión de la Domótica en el Hogar

Hasta la actualidad el usuario de una instalación eléctrica convencional se conformaba sencillamente con iluminarse, calentarse y disponer de acceso al tendido eléctrico para conectar los electrodomésticos y demás componentes eléctricos.

Los requisitos de una instalación se limitaban a proteger las líneas y las personas contra los riesgos eléctricos. Posteriormente, a las funciones tradicionales se han

<sup>1</sup>Fuente: Romero Morales, Cristobal, 2001, Domótica e Imnótica, Alfaomega, México.

añadido nuevas funciones y productos que gestionan la energía y el confort como aparatos que permiten aplicaciones específicas como programar la calefacción, regular la temperatura ambiental, gestionar el consumo de energía, etc.

La incorporación de estas instalaciones singulares ha supuesto mayor complejidad, aumentando el cableado interno de la vivienda y provocando que una ampliación y/o modificación de dicha instalación se traduzca en largas y costosas intervenciones por parte del instalador.

### 1.1 ¿Qué es la domótica?

Entendemos como tal la incorporación al equipamiento de nuestras viviendas y edificios una práctica, intuitiva e innovadora tecnología que permita gestionar de forma energéticamente eficiente, segura y confortable para el usuario, los distintos aparatos e instalaciones domésticas tradicionales que conforman una vivienda ( la calefacción, la lavadora, la iluminación...)<sup>2</sup>

El usuario nota que tiene el control, y ese control lo ejerce mediante el sistema para ahorrar o derrochar (yo controlo, yo decido). Podemos hacer que una luz se encienda al abrir la puerta, o que se cierren automáticamente el gas y el agua, se bajen las persianas y se apague la calefacción cuando activamos la alarma al salir de casa, pero todo ello porque yo quiero, no porque el *sistema* quiera (el usuario establece de forma intuitiva, centralizada y eficiente como quiere que su hogar se automatic). Podemos considerar la domótica como la utilización simultánea de electricidad, electrónica e informática en la gestión técnica de las viviendas o bien decir sencillamente que es el lenguaje mediante el cual el usuario y la vivienda se comunican. Entendemos por gestión técnica la modificación, ya sea local o remota, de los parámetros de los servicios y funciones implementados en la instalación.

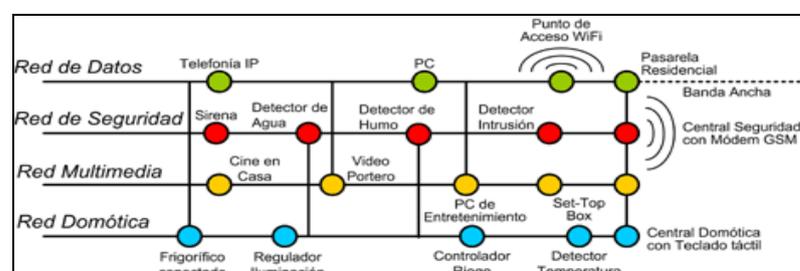


FIG 2: Modelo conceptual de la integración de sistemas del Hogar Digital

<sup>2</sup>Fuente: Millan Tejedor, Ramon, 2003, Domótica Edificios Inteligentes, Cre, Madrid.

Según el destinatario podemos hacer distintas definiciones:

**a)Para el usuario:** Aquella que permite una mayor calidad de vida a través de la tecnología, ofreciéndole un aumento del bienestar y la seguridad de los habitantes, a la vez que una reducción de las tareas domésticas y una racionalización de los distintos consumos.

**b)Para el técnico:** Aquella que incluye agrupaciones automatizadas de equipos normalmente asociadas por funciones, que disponen de la capacidad de comunicarse interactivamente entre ellas a través de un soporte de comunicaciones que las integra.

Antiguamente el usuario de una instalación eléctrica convencional se conformaba solamente con iluminarse, calentarse y disponer de acceso al tendido eléctrico para conectar los distintos dispositivos asociados.

Los requisitos de la instalación se limitaban a proteger las líneas y las personas contra los riesgos eléctricos.

Posteriormente, a las funciones tradicionales se han añadido nuevas funciones y productos que gestionan la energía y el confort (primordialmente).

Para dar respuesta a estas nuevas demandas, han aparecido en el mercado una extensa variedad de dispositivos que permiten aplicaciones específicas como programar la calefacción, regular la temperatura ambiental, gestionar el consumo de energía, etc.

## **1.2 Beneficios**

Los beneficios que aporta la domótica son múltiples, y se podría afirmar que cada día surgen nuevos. Por ello los agruparemos en los siguientes apartados:

**1-** El ahorro energético gracias a una gestión tarifaria e "inteligente" de los sistemas y consumos.

**2-** La potenciación y enriquecimiento de la propia red de comunicaciones.

**3-** La más contundente seguridad personal y patrimonial.

**4-** La tele-asistencia.

5- La gestión remota (vía teléfono, radio, Internet, etc.) de instalaciones y equipos domésticos.

6- Aumento del bienestar y en definitiva, del confort.

### **1.3 Aplicaciones**

La domótica busca el aprovechamiento al máximo de la energía y luz solar adecuando su comportamiento a nuestras necesidades.

Las posibles aplicaciones son innumerables dadas las posibilidades de la domótica, podemos decir tranquilamente que las posibilidades son tan extensas como puedan ser las pretensiones de los propios usuarios, por ello trataremos de agruparlas en las más comunes:

#### **1.3.1 En el ámbito del ahorro energético**

a) Programación y zonificación de la climatización. El usuario personaliza a qué hora y que zonas de la vivienda desea que estén gestionadas por el control central

b) Racionalización de cargas eléctricas: desconexión de equipos de uso no prioritario en función del consumo eléctrico en un momento dado. (Reduce la potencia contratada).

c) Gestión de tarifas, derivando el funcionamiento de algunos aparatos a horas de tarifa reducida.

#### **1.3.2 En el ámbito del nivel de confort**

a) Control de todos los dispositivos instalados y operativos desde un dispositivo central simplificando su gestión y optimizando su uso.

b) Apagado general de todas las luces de la vivienda.

c) Automatización del apagado/encendido en cada punto de luz. La forma de encender y apagar la iluminación de la vivienda puede ser automatizada y controlada de formás complementarias al control tradicional a través del interruptor clásico; se puede en está manera conseguir un incremento del confort y ahorro energético.

La iluminación puede ser regulada en función del nivel de luminosidad ambiente, evitando su encendido innecesario o adaptándola a las necesidades del usuario. La activación de ésta se realiza siempre cuando el nivel de luminosidad pasa un determinado umbral, ajustable por parte del usuario. Esto garantiza un nivel de iluminación mínima, que puede ser especialmente útil para por ejemplo un pasillo o la iluminación exterior.



FIG 3: Aplicación Domótica en Iluminación

La iluminación puede ser activada en función de la presencia de personas en la estancia; se activa la iluminación cuando un sensor detecta presencia. Esto garantiza una buena iluminación para por ejemplo zonas de paso como pasillos. Asegura que luces no se quedan encendidas en habitaciones cuando no hace falta.

**d)** Regulación de la iluminación según el nivel de luminosidad ambiente.

**e)** Automatización de todos los distintos sistemas/ instalaciones / equipos dotándolos de control eficiente y de fácil manejo. El hecho de que los sistemas de la vivienda se pueden programar ya sea para que realicen ciertas funciones con sólo tocar un botón o que las lleven a cabo en función de otras condiciones del entorno (hora, temperatura interior o exterior, etc.) produce un aumento del confort y un ahorro de tiempo.

**f)** Integración del portero al teléfono, o del video-portero al televisor. La señal de audio y control del portero automático se puede integrar en la red de telefonía

interior de la vivienda, para permitir utilizar el teléfono en lugar de la habitual consola de control de esta instalación.

Cualquier llamada desde el portero automático puede ser atendida desde un terminal telefónico, entablando conversación con la persona visitante y, si es preciso, abrirle la puerta. La señal de vídeo y control del video-portero automático se puede integrar en la red de televisión de la vivienda y edificio, para permitir utilizar el televisor en lugar de la habitual consola de control de esta instalación. Cualquier llamada desde el video-portero automático puede ser atendida desde el televisor, reconociendo la persona visitante y, si es preciso, abrirle la puerta mediante el propio mando a distancia del televisor (u otro de uso específico).

Opcionalmente, y cuando no hay nadie en la vivienda, podría pensarse en desviar la llamada desde el portero automático a un número de abonado telefónico, simulando la presencia de un usuario en casa o abrirle la puerta de acceso de la calle a por ejemplo un mensajero.

**g)** El riego automático es una aplicación muy utilizada por la gente que vive en viviendas unifamiliares. El riego puede ser gestionado por un controlador que normalmente se limita a regar según la programación horaria. Pero el riego puede ser más desarrollado y avanzado que eso. Puede ser activado de forma automática según programación horaria, pero también según la humedad en el césped, el día de la semana o cualquier otro valor. Además si el riego está integrado en el sistema de domótica se puede controlar el riego de forma remota o según otros eventos como incendios o robos.

Además existe la posibilidad de realizar actuaciones puntuales y personalizadas como por ejemplo regar por la tarde en vez de por la noche si el dueño planifica una barbacoa con los amigos por la noche.

### **1.3.3 En el ámbito de la protección personal y patrimonial**

**a)** Detección de un posible intruso. En caso de intruso el control central se encarga de hacer saltar las alarmas, a la vez que avisa al propietario del inmueble y las autoridades.

**b)** Simulación de presencia. Gestión del control de acceso y control de presencia, así como la simulación de presencia.

**c)** Detección de conatos de incendio, fugas de gas, escapes de agua. Mediante el nodo telefónico, se puede tener acceso (mediante un pulsador radio-frecuencia que se lleve encima por ejemplo) a los servicios de Salud, Policía, etc. A través del nodo telefónico es posible desviar la alarma hacia los bomberos, por ejemplo.

**d)** Servicios de información, tele-compra, tele-banco, alerta médica (Tele-asistencia), etc. Para ciertos colectivos estos servicios pueden ser de gran utilidad (por ejemplo, unidades familiares donde ambos cónyuges trabajan) ya que producen un ahorro de tiempo.

**e)** Cerramiento de persianas puntual y seguro.

**f)** Se puede detectar averías en los accesos, en los ascensores, etc.

#### **1.3.4 En el ámbito de las comunicaciones.**

**a)** Dentro de la vivienda

A través de un esquema de comunicación con los distintos equipos (mando a distancia, bus de comunicación, etc.). Reduce la necesidad de moverse dentro de la vivienda, este hecho puede ser particularmente importante en el caso de personas de la tercera edad o minusválidos.

**b)** Fuera de la vivienda

Presupone un cambio en los horarios en los que se realizan las tareas domésticas (por ejemplo: la posibilidad de que el usuario pueda activar la cocina desde el exterior de su vivienda, implica que previamente ha de preparar los alimentos) y como consecuencia permite al usuario un mejor aprovechamiento de su tiempo.

**c)** Transmisión de alarmas.

**d)** Intercomunicaciones entre las habitaciones.

**f)** Telefonía IP. Las comunicaciones de voz por Internet utiliza la conexión a Internet como red de transporte de los datos, para realizar una comunicación *VoIP* (Voice Over IP). Se puede realizar las llamadas desde el ordenador personal hasta otro PC remoto o bien hasta cualquier tipo de teléfono, basta con disponer de un PC, conexión a Internet, un equipo multimedia (altavoces y micrófono) y el software

necesario para ello. Pero también existe la posibilidad de integrar, o hasta sustituir la telefonía tradicional con la telefonía IP. Como *terminal* para realizar las llamadas se puede utilizar por parte del que tiene contratado el servicio:

- El PC, aprovechando los altavoces y micrófonos internos o externos.
- El PC, con un teléfono especial conectado al puerto USB.
- Un teléfono normal conectado a un concentrador que a su vez está conectado a un router.
- Un Teléfono o SmartPhone dotado de tecnología WiFi que directamente integra el software de telefonía IP.



FIG 4 :Esquema conceptual sobre el sistema VoIP.  
Fuente: Catalogo de Instalaciones IP DLink

## 1.4 Seguridad

Las aplicaciones de seguridad contemplan tanto la seguridad de las personas como la seguridad de los bienes materiales. Hay que destacar:

- El sistema anti-intrusión: existen dos tipos de sistemas anti-intrusión que son los siguientes:

**1) Protección perimetral:** protege de accesos a la parcela y a la misma vivienda a través de puertas y ventanas. Principalmente se utiliza barreras infrarrojas de exterior en vallas, el jardín y ventanas y puertas; y sensores de contacto magnético de puerta/ventana y sensores de rotura de cristal.

**2) Protección del interior:** protege de intrusión dentro de la misma vivienda. Se utiliza normalmente sensores de detección de movimiento con tecnologías infrarroja y ultrasónica.

La alarma técnica consiste en un conjunto de detectores (incendio, gas, CO, inundación, etc.) que en caso de producirse fuegos o escapes de fluidos, posibles errores en la instalación o mal uso involuntario detectan la anomalía y proceden al corte del suministro mediante la válvula de fluidos, la desconexión de los circuitos eléctricos y otras actuaciones previstas. También pueden configurarse para que den un aviso de entre los siguientes:

**1) Local:** sirenas, timbres, luces, mensajes hablados etc.

**2) Remoto:** a las Centrales Receptoras de Alarmas y/o al usuario final directamente, a través del teléfono convencional, móvil, correo electrónico o similar.

- La *simulación de presencia* consiste en la activación aleatoria de algunos elementos como la iluminación y las persianas en los periodos en que la vivienda no está ocupada, creando una sensación de actividad en su interior.

- Las *alarmas personales*: existen dos tipos de alarmas personales destinadas al hogar.

**Aviso SOS o pánico:** se utiliza en casos de emergencias graves cuando hay intrusos, de robo o ataques personales realizados dentro o justo fuera de la vivienda.

**Avisos de asistencia:** se utiliza para llamar la atención de necesidad de asistencia personal principalmente para personas de tercera edad o gente discapacitada.

### **1.5 Sistema de monitorización personal (video-vigilancia)**

Cualquier sistema de seguridad, esté o no homologado, puede ser instalado en una vivienda y configurado para que avise directamente al usuario o propietario de la misma. En este caso es el propio usuario quien gestiona que hacer en caso de que se produzca un evento en la vivienda.

Cada vez más están saliendo al mercado productos o sistemas que pueden ser configurados para avisar directamente al usuario final. Además suelen incorporar la opción de darse alta en una *Central Receptora de Alarmas* y según el tipo de evento, avisar a uno o a ambos.

Adicionalmente se utilizan cámaras de seguridad que pueden ser consultadas y/o grabadas remotamente o de forma local para posteriormente identificar a los intrusos. Las cámaras de vigilancia pueden ser cámaras convencionales analógicas que están cableadas a una central capaz de tratar la imagen para acceso local o a través de Internet (video servidores con o sin IP). O cada cámara puede tener conexión a Internet (cámaras IP) para el acceso directo a la cámara a través de Internet o para incluir la imagen en otra interfaz global del control y la vigilancia de la casa.

En las familias modernas normalmente ya no hay una persona con la responsabilidad global del hogar en la casa todo el día. Y muchas veces cuando no estamos en casa nos gustaría saber lo que está pasando por ejemplo cuando llega la asistenta, lo que hace la muchacha con nuestros hijos, si han llegado los niños del colegio, lo que hacen los niños cuando les dejamos solos en la casa por la noche. Podemos definir estas *alarmas emocionales* como ejemplo:

*Avisos de actividades:* como la llegada o salida de terceros (asistenta, muchacha, jardinero, fontanero, etc.) o de los familiares (hijos, padres, etc.) a la vivienda.

*Avisos de ausencia de actividad:* si se queda alguien en la vivienda (niños, ancianos, etc.) sin realizar ninguna actividad en un determinado intervalo de tiempo, algo que puede ser una indicación de que ha pasado algo, como una caída o similar, o que una persona mayor no se ha levantado por la mañana.

El tipo de aviso se puede estructurar en dos tipos:

Mensajes de texto o hablados, guardados en la misma central, o avisos en tiempo real a teléfonos fijos, móviles, correo electrónico (e-mail), etc. que avisan de la conexión o desconexión de la alarma, accesos a zonas específicas, etc.

Mensajes con imágenes enviados como MMS, o mediante la tecnología digital *streaming*, al móvil o por correo electrónico, o películas grabadas guardadas en el video, PC, según programación horaria o según los eventos dentro de la casa.

Adicionalmente podemos monitorizar la vivienda en tiempo real de forma local a través de la televisión, PC o similar o remotamente a través de Internet para ver las actividades que ocurren dentro del hogar con cámaras distribuidas por distintas zonas y habitaciones la casa.

### **1.6 Confort (automatismos)**

El campo de automatización de sistemas e instalaciones eléctricas es muy amplio, dependiendo de las necesidades e ideas del usuario, como más significativos destacamos:

Accionamiento automático de persianas para evitar los efectos atmosféricos (viento, lluvia, nieve, etc.) sobre estos elementos para limitar sus pérdidas térmicas.

Accionamiento automático de la iluminación en función del nivel de iluminación y de la presencia de personas.

Centralización y supervisión de la información del estado de los sistemas instalados.

Posibilidad de agrupar un conjunto de acciones en una sola orden. Por ejemplo: apagar todas las luces, cerrar las persianas y cortar el suministro de agua, gas y poner el sistema de alarma en estado de vigilancia con un mismo interruptor con llave.

## CAPITULO 2

### GENERALIDADES DE LA FRECUENCIAS DTMF

#### 2.1 Funcionamiento general del teléfono

Para que pueda funcionar un teléfono, las centrales telefónicas actuales disponen de bancos de baterías a 48 VCC, las cuales energizan los equipos terminales (aparatos telefónicos) conectados a ellas. Dependiendo del abonado y la central, la impedancia de la línea en DC puede variar entre 400 a 1750 ohm, por otra parte la impedancia en AC, es alrededor de 600 ohm la cual corresponde a la característica de un cable bifilar, usado comúnmente en telefonía.

Cuando el teléfono está colgado la impedancia del equipo terminal es grande, no drenando la corriente de las baterías de la central telefónica, al descolgarse el teléfono empieza a circular corriente. De esta manera la central detecta que el usuario desea comunicarse y envía una señal o tono de 350 a 450 ha de invitación a marcar ; enviando en adelante los tonos se los decodifica mediante filtros especiales conociendo en forma instantánea que digito se marco, si el teléfono es descolgado la central elimina la señal de timbrado del teléfono destino y la señal del teléfono de origen de esta manera se cierra el lazo de unión entre los dos suscriptores permitiéndose así la comunicación.

#### 2.2 DTMF

DTMF significa: Tonos Duales de multi frecuencia. Una señal DTMF generada por cualquier dispositivo, es en sí la suma de dos ondas sinusoidales con diferentes frecuencias<sup>3</sup>

Un sistema de este tipo (como el del teléfono) usa pares de tonos para representar los diferentes números del teclado, es decir, existe un par de tonos asociado a cada botón, un *tono bajo* y un *tono alto*. En la figura siguiente se

---

<sup>3</sup> Fuente: Martínez, Procopio, 2006, Diseño de Herramientas de Comunicación, Limusa, México

muestra la distribución de frecuencias en un teclado completo, los *tonos bajos* varían con la línea horizontal, mientras que los *tonos altos* lo hacen con la vertical.

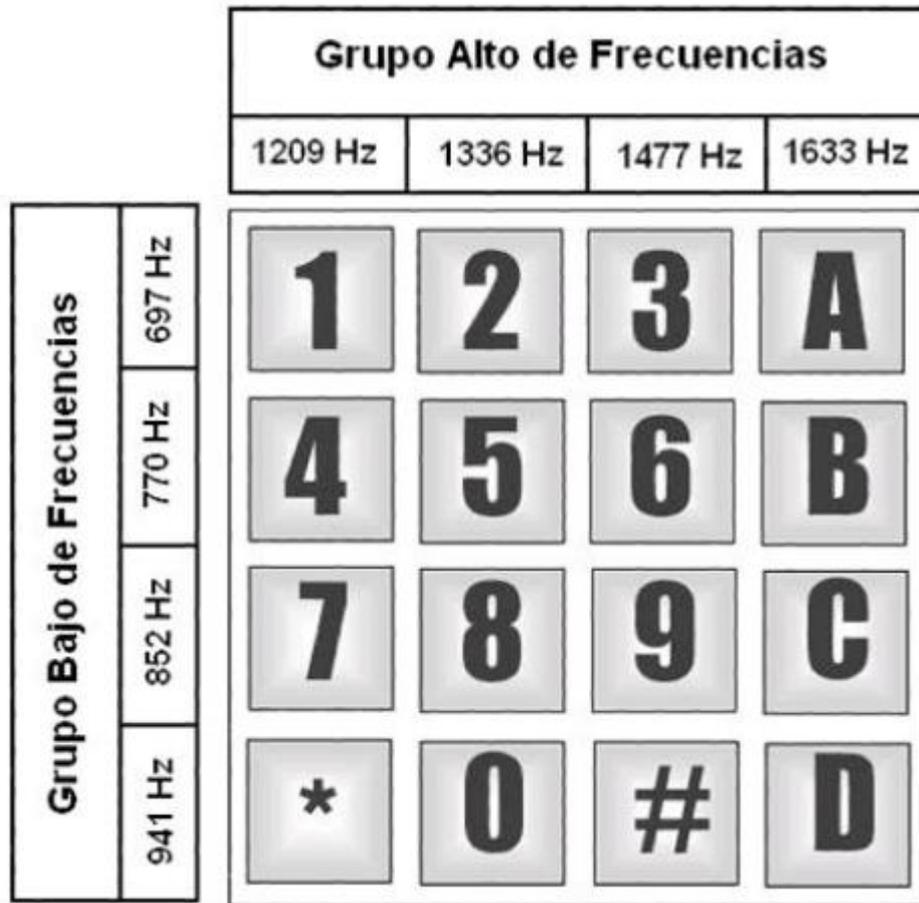


FIG5: Diagrama de distribución de los pares de frecuencia para DTMF.

### 2.3 Tipos de códigos.

Los tonos DTMF se dividen en dos tipos de códigos; el código estándar y el extendido. El código estándar de DTMF comprende los números (0-9) y los símbolos \* y #; mientras que el código extendido agrega al código estándar las letras (A-D). Sin duda el código estándar es el más usado en la industria, principalmente por su aplicación en los teléfonos; en nuestro país, se ha incrementado el uso de los tonos gracias a la conversión de líneas analógicas por digitales, permitiendo aplicaciones de selección automática de opciones en empresas bancarias, así como en el servicio información a clientes y/o usuarios de empresas principalmente.

El código extendido fue creado en la red telefónica *Autovon* del ejército de los Estados Unidos. El uso que se le daba a este código era para establecer un nivel de prioridad en la llamada (siendo "A" el más alto nivel de prioridad), permitiendo así cortar una llamada, de cierta línea telefónica, con un cierto nivel de prioridad menor al de la llamada entrante<sup>4</sup>. Actualmente estos botones se usan principalmente en aplicaciones especiales como repetidores de radio amateur para su control de señalización. Los Módems y los circuitos de DTMF están tendiendo a incluir estos tonos, aunque todavía no están siendo usados en servicios públicos generales, por lo pasarán años antes de que estos tonos se puedan usar en líneas de servicios de información a clientes (entre otros), debido a que dichos sistemas deben ser compatibles con el código de 12 dígitos.

De la figura anterior podemos observar que cuando se presiona el número 3 (por ejemplo) se genera una suma de dos ondas sinusoidales con frecuencias de 697 y 1477 Hz. Respectivamente, que son las que representan a dicho número. De esta manera cualquier sistema puede decodificar un número a través de sus frecuencias asignadas.

Las frecuencias de los tonos fueron diseñadas para evitar armónicos y otros problemas que pueden surgir cuando dos tonos son enviados y recibidos. Es muy importante asegurar una transmisión y recepción precisa, libres de errores en el proceso de comunicación de transmisor-receptor para que un sistema funcione adecuadamente, por lo que los tonos deben estar en un rango de  $\pm 1.5\%$  de su valor nominal, la frecuencia alta debe estar, por lo menos, al mismo nivel de potencia que la frecuencia baja; aunque es recomendable que la frecuencia alta esté 3 dB por arriba de la frecuencia baja, con un máximo de 4 dB .

Los decodificadores de DTMF más usados, generan un código binario en su salida, que corresponde al número captado a través de su entrada analógica.

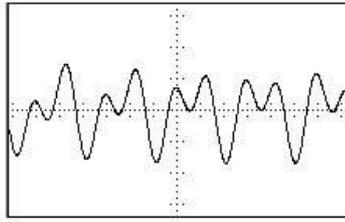
## 2.4 Formas de onda.

Con el fin de enfatizar la diferencia entre las diferentes señales DTMF, a continuación se muestra la forma de onda para la suma de cada par de tonos del código estándar:

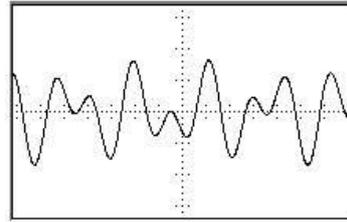
---

<sup>4</sup>Fuente: Smile, Philip H, 2000, Introducción a los Sistemas de Telecomunicaciones, Trillas, Mexico. [www.taringa.net/posts/ebooks-tutoriales/1631123/Libros-de-Electronica-y-Microcontroladores-PIC.html](http://www.taringa.net/posts/ebooks-tutoriales/1631123/Libros-de-Electronica-y-Microcontroladores-PIC.html)

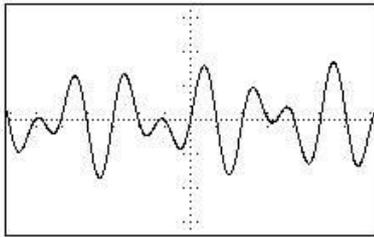
1



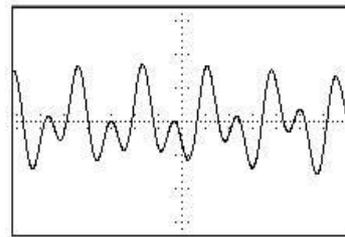
2



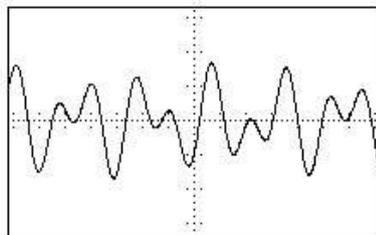
3



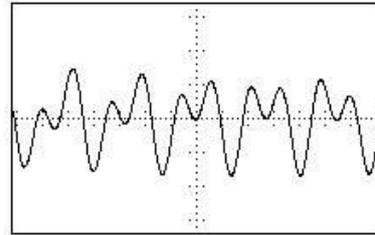
4



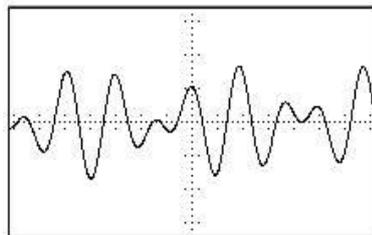
5



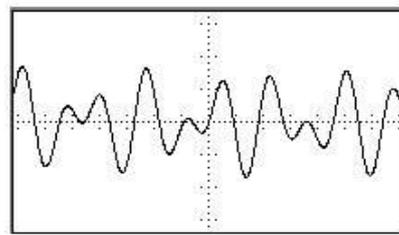
6



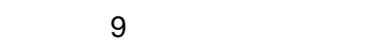
7



8



9



0



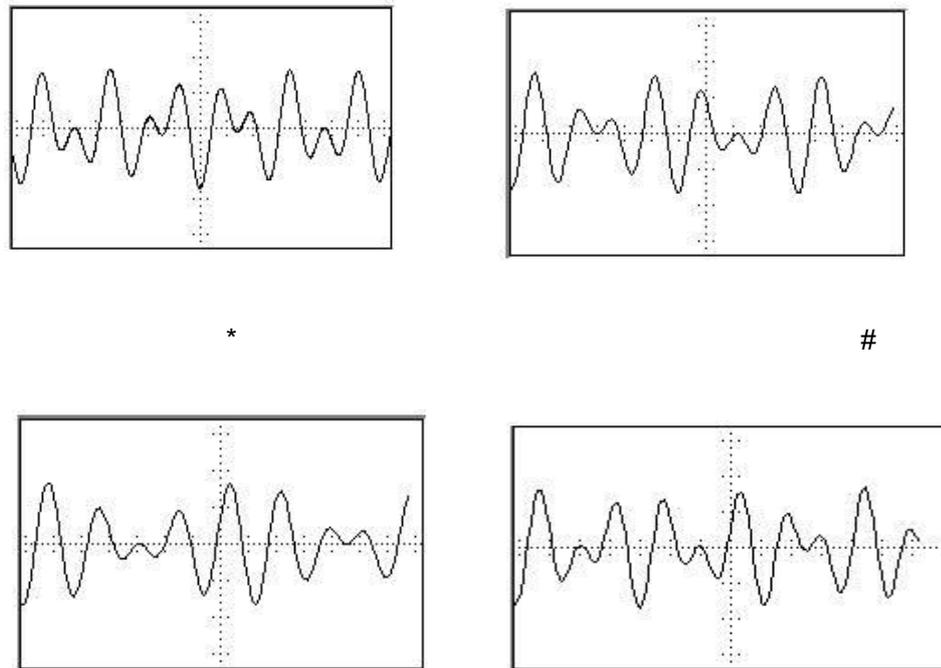


FIG 6: Formás de onda de tonos del código estándar.  
Fuente: Cebek Electronic Circuit DTMF

Cabe mencionar que debido a las características de los tonos DTMF, se pueden utilizar en sistemas de transmisión cuyos requerimientos nos conduzcan a la necesidad de dispositivos confiables (alámbricos e inalámbricos). Además se observa la versatilidad que tienen intrínsecos los tonos DTMF, por lo que pueden ayudar para que un sistema adopte esas mismas características multitareas.

## 2.5 Parámetros y valores

Parámetros	Valores
Frecuencia de señales.	Grupo bajo: 697, 770, 852, 941 Hz. Grupo alto: 1209, 1336, 1477, 1633 Hz.
Tolerancia de frecuencia.	< 1.8%
Niveles de potencia por frecuencia.	0 a -25dBm.
Diferencia de potencia entre señales.	+4dB a -8dB.
Duración mínima de la señal.	40 ms (para ser aceptada).
Pausa mínima entre dígitos.	40ms.
Tolerancia al ruido.	-12dB.
Nivel alto de voltaje de entrada mínimo.	3.5 Volts.

Nivel bajo de voltaje de entrada máximo.	1.5 Volts.
Señal simulada por voz en el sistema.	Para los códigos 0-9, 1 falsa por cada 3000 llamadas. Para los códigos 0-9, *, #, 1 falsa por cada 2000 llamadas. Para los códigos 0-9, *, #, A-D, 1 falsa por cada 1500 llamadas.
Interferencia por ecos.	Debe tolerar ecos retrasados más de 20ms. Y con al menos 10 dB de atenuación.

TABLA 1: Parámetros principales de los DTMF.

Al ser utilizados los DTMF en sistemas telefónicos, es muy importante que se eviten los errores por simulación de voz, ya que como vemos en la siguiente figura, algunas frecuencias utilizadas por el sistema corresponden al rango de frecuencias cubiertas por la voz. Se conoce que las frecuencias más bajas alcanzadas por la voz humana son del orden de 80 Hz, mientras que un soprano puede alcanzar frecuencias de hasta 1200Hz.

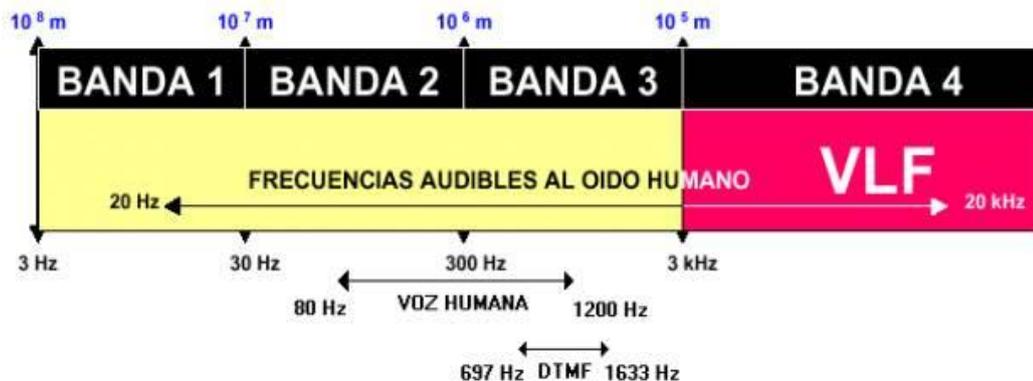


FIG 7: Espectro de los DTMF y de la voz humana.

## 2.6 Estructura interna del MT8870

El MT8870 es un completo receptor que integra ambos filtros tanto de frecuencia altas como el de frecuencias bajas y además funciones de decodificador digital, para ello utiliza las técnicas de contador digital para descubrir y descifrar los 16 tonos pares del DTMF en un código binario de cuatro dígitos. La construcción de este integrado es muy compacta ofrece bajo consumo de potencia y gran desempeño, su arquitectura consiste en una sección de filtro de bandsplit que

separa los grupos de frecuencias altas y bajas seguida por una sección de contador digital que verifica la frecuencia y duración de los tonos antes de pasar el código correspondiente al bus de salida.

La separación de los tonos bajos de los altos se logra aplicando a la entrada una señal DTMF por un filtro pasa banda mediante el cual se filtran los anchos de banda correspondientes a las frecuencias del grupo alto y bajo.

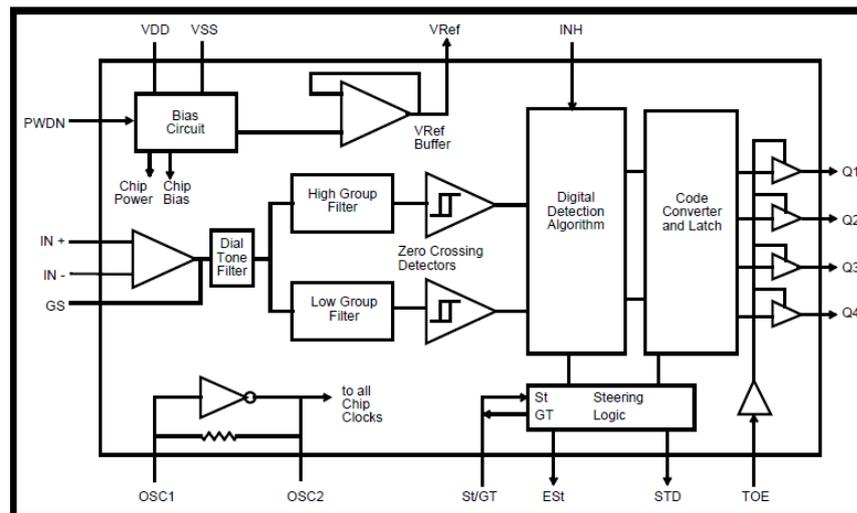


FIG 8: ESTRUCTURA DE BLOQUES DEL MT 8870

Fuente: Datasheet, Hoja de datos del MT8870, [www.datashet.com](http://www.datashet.com)

La sección del filtro es un decodificador que emplea las técnicas de contadores digitales para determinar las frecuencias de los tonos entrantes y verifica que ellos correspondan a las frecuencias DTMF normales. Un algoritmo protege contra la simulación de tono por signos extraños como la voz mientras proporciona la tolerancia a las pequeñas desviaciones de frecuencia.

El algoritmo se ha desarrollado para asegurar la combinación óptima de inmunidad de confusiones, cuando se reconoce la presencia de los tonos válidos el pin EST irá a un estado activo cualquier pérdida subsecuente de está generara un estado inactivo del pin EST.

Pin N°	Nombre	Descripción
1	IN+	<b>Non-Inverting Op-Amp. (Input).</b> Entrada de señal no invertida.
2	IN-	<b>Inverting Op-amp (Input).</b> Entrada de señal invertida.
3	GS	<b>Gain Select.</b> Ajuste de ganancia.
4	V <sub>Ref</sub>	<b>Reference Voltaje (Output).</b> Es una salida con un voltaje igual a la mitad de la fuente.
5	INH	<b>Inhibit (Input).</b> Un lógico alto en este pin prohíbe la detección de los tonos correspondientes a las teclas A, B, C y D.
6	PWDN	<b>Power Down (Input).</b> Un lógico alto en este pin pone el dispositivo en modo de bajo consume.
7	OSC1	<b>Clock (Input).</b>
8	OSC2	<b>Clock (Output).</b> Conexión del cristal de 3.579545 MHz entre los pines OSC1 y OSC2 completa el circuito del oscilador interior.
9	V <sub>SS</sub>	<b>Ground (Input).</b>
10	TOE	<b>Three State Output Enable (Input).</b> Un lógico bajo en este pin pone las salidas en alta impedancia, un alto las habilita.
11 – 14	Q1 – Q4	<b>Three State Data (Output).</b> Salida de datos, mantiene en latch el último código recibido.
15	StD	<b>Delayed Steering (Output).</b> Genera un pulso alto cuando recibe un tono válido y la salida actualiza el código recibido.
16	Est	<b>Early Steering (Output).</b> Presenta un lógico alto cuando se detecta un tono válido.
17	St/GT	<b>Steering Input/Guard time (Output) Bidirectional.</b> Ajusta el nivel de sensibilidad.
18	V <sub>DD</sub>	<b>Positive power supply (Input).</b> Fuente positiva entre 4.75 y 5.25 V

TABLA 2: Descripción de pines del MT 8870

El MT8870 fue seleccionado de entre una amplia variedad de circuitos integrados debido principalmente a sus características; eléctricas, disponibilidad e interfaz de comunicación paralela con salida de tercer estado que lo hacen óptimo para este proyecto ya que, debido a que los recursos del controlador principal son compartidos por todos los módulos que componen el sistema en donde la implementación de un canal de comunicación de datos virtual es clave en la manipulación de la información necesaria para atender los requerimientos específicos de cada uno de los componentes que integran el sistema.

Este decodificador de señales DTMF se encuentra provisto además de las características ya mencionadas, de una terminal que indica si una señal DTMF válida ha sido recibida, decodificada y su correspondiente código binario capturado en el registro de salida, esta terminal es llamada StD y permite llevar a cabo una interacción en tiempo real con el sistema de control empleándola para provocar una interrupción de manera tal que el controlador atiende esta acción prioritaria ejecutando la rutina de atención a interrupción, misma que se limita a leer el código

binario presente en el registro de salida del decodificador DTMF, para que posteriormente el programa principal de el tratamiento de comando o dato según sea el caso y efectuar la acción correspondiente dentro del sistema de control remoto.

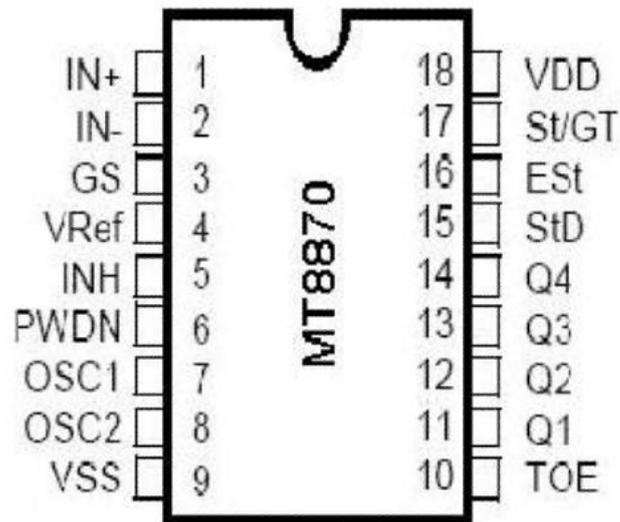


FIG 9: DIAGRAMA DE PINES MT 8870

Fuente: Datasheet,2010, Diagrama de Pines, [www.datasheetcatalog.net/es/datasheet](http://www.datasheetcatalog.net/es/datasheet)

Los ingenieros de Laboratorios Bell eligieron pares de frecuencias específicas para evitar armónicos y otros problemas que pudieran presentarse cuando estos tonos son enviados o recibidos, además de la dificultad que presentan para ser imitados por la voz humana y puedan accidentalmente activar el decodificador del otro lado de la línea.

## 2.7 Funcionamiento y Configuraciones del MT8870

Existen varias formas de detectar y decodificar estos tonos, una forma podría ser; con ocho filtros sintonizados en combinación con circuitos de detección. Resulta innecesario decir que esto no es práctico considerando la cantidad de circuitos integrados de diferentes fabricantes que hay que conseguir y el complejo ajuste que hay que realizar para sintonizarlos.

En este caso particular la decodificación se encuentra a cargo de un circuito integrado MT8870CE fabricado por MITEL, el cual reconoce los tonos dobles generados por el teclado de un teléfono en modalidad digital, estos tonos

son decodificados y después convertidos a cantidades binarias de 4 bits que corresponden a un valor numérico determinado. Estos valores binarios funcionan como comandos o datos de acuerdo al orden con que son ingresados al sistema de control.

El circuito mostrado en la figura siguiente ilustra el uso de un dispositivo MT8870 en un sistema de recepción típico. Las especificaciones de British Telecom definen las señales de entrada menores de -34 dBm como niveles no operativos. Esta condición puede ser evitada seleccionando valores adecuados de R1 y R2 para proporcionar una atenuación de 3 dB, de tal forma que la señal entrada de -34 dBm corresponderá a -37 dBm en la terminal de programación de ganancia (GS). En el diagrama se muestran los valores de los componentes de R3 y C2 para los requerimientos de tiempo de guarda cuando la tolerancia total de los componentes es del 6%.

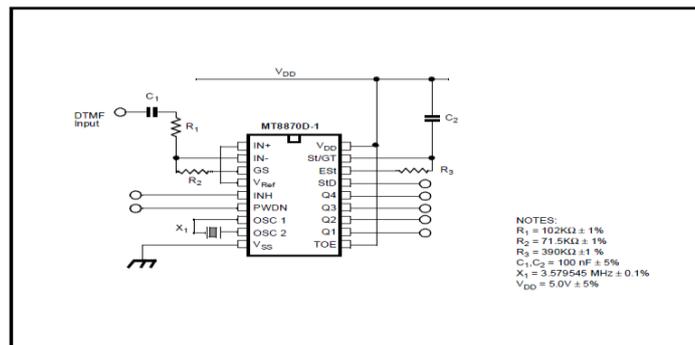


FIG 10 : Conexión General del MT 8870

Fuente: DATASHEET, 2010, Diagrama de Pines,

[www.datasheetcatalog.net/es/datasheets\\_pdf/M/T/8/8/MT8870.shtml](http://www.datasheetcatalog.net/es/datasheets_pdf/M/T/8/8/MT8870.shtml)

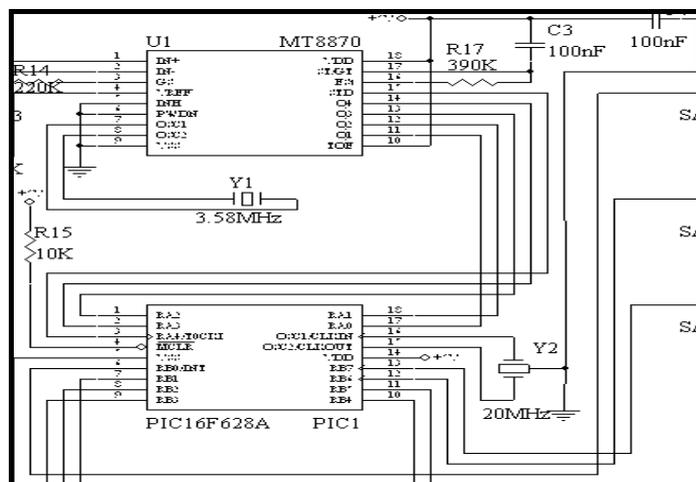


Fig 11: Circuito de Interconexión con el PIC 16F628A

## CAPITULO 3

### 3.0 CONTROL DE SEÑALES POR PUERTOS DE LA PC

Un puerto es una conexión, el cual es utilizado para conectar dispositivo de Hardware como impresoras o Mouse, permitiendo el intercambio de datos con otro dispositivo. También existen puertos internos definidos mediante el Software.

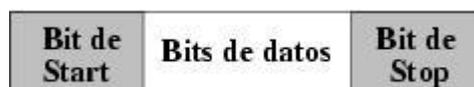
Normalmente estos puertos se encuentran en la parte trasera del computador, aunque en la actualidad muchos computadores incorporan puertos USB y audio en la parte delantera.

#### 3.1 Puertos en Serie.

El puerto en serie de un ordenador es un adaptador asíncrono utilizado para poder intercomunicar varios ordenadores entre si. Un puerto serie recibe y envía información fuera del ordenador mediante un determinado software de comunicación o un drive del puerto serie.

#### 3.3 Comunicaciones serie asíncronas

Los datos serie se encuentran encapsulados en tramas de la forma:



Primero se envía un **bit de start**, a continuación los **bits de datos** (primero el bit de mayor peso) y finalmente los **bits de STOP**.

El número de bits de datos y de bits de Stop es uno de los parámetros configurables, así como el criterio de paridad par o impar para la detección de errores. Normalmente, las comunicaciones serie tienen los siguientes parámetros: 1 bit de Start, 8 bits de Datos, 1 bit de Stop y sin paridad.

En esta figura se puede ver un ejemplo de la transmisión del dato binario 10011010. La línea en reposo está a nivel alto:

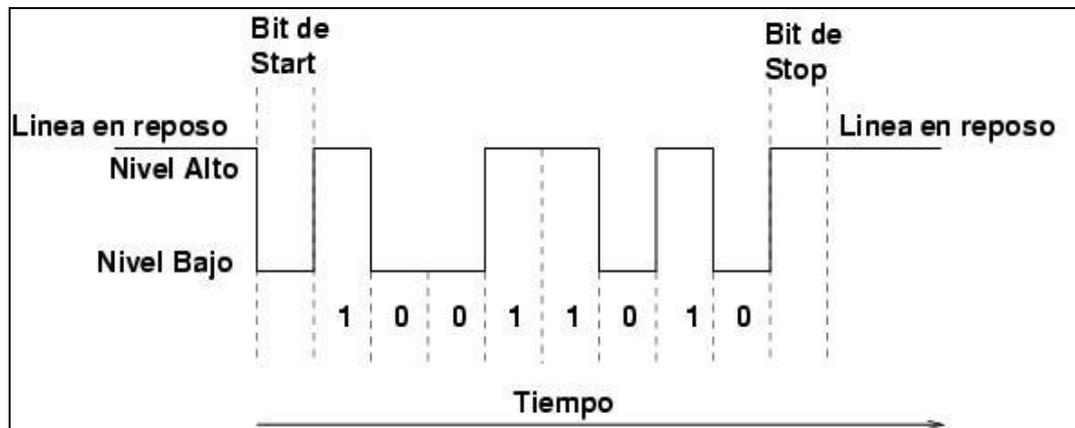


FIG 12: Ejemplo de transmisión de un dato binario

El Software envía la información al puerto, carácter a carácter, convirtiendo en una señal que puede ser enviada por cable serie o un módem. Cuando se ha recibido un carácter, el puerto serie envía una señal por medio de una interrupción indicando que el carácter está listo. Cuando el ordenador ve la señal, los servicios del puerto serie leen el carácter.

### 3.3.1 Forma de puerto serial

En la mayoría de los casos hay 2 tamaños de puertos, el primero sería de 25 pines, que tiene una longitud de alrededor de 38 mm; y otro de 9 pines que tiene una longitud de 17 mm que es el que se utiliza en este trabajo.

En los PC's, se emplea como conector del interface serie, un Terminal macho, al que se lo llama DTE (Dato Terminal Equipment), que a través de un cable conectaremos a un periférico que posee un conector hembra al que llamaremos DCE (Data Communications Equipment).

### 3.4 El Conector DB9 del PC

En los PCs hay conectores DB9 macho, de 9 pines, por el que se conectan los dispositivos al puerto serie. Los conectores hembra que se enchufan tienen una colocación de pines diferente, de manera que se conectan el pin 1 del macho con el pin 1 del hembra, el pin2 con el 2, etc.

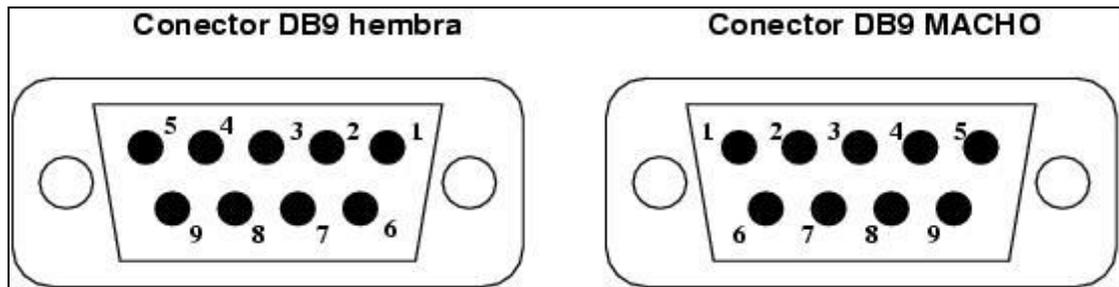


FIG 13: CONECTOR MACHO – HEMBRA DB9

Fuente: Electronica Facil, 2010, [www.electronicaestudio.com/interfase.htm](http://www.electronicaestudio.com/interfase.htm)

La información asociada a cada uno de los pines es la siguiente en el conector DB9 a utilizar en este trabajo:

Número de pin	Señal
1	<b>DCD</b> (Data Carrier Detect)
2	<b>RX</b>
3	<b>TX</b>
4	<b>DTR</b> (Data Terminal Ready)
5	<b>GND</b>
6	<b>DSR</b> (Data Sheet Ready)
7	<b>RTS</b> (Request To Send)
8	<b>CTS</b> (Clear To Send)
9	<b>RI</b> (Ring Indicator)

TABLA 3: DESCRIPCION DE PINES CONECTOR DB9

En nuestro Terminal DTE, Tenemos las siguientes CONEXIONES (para un conector DB-25):

PIN	Nombre	Dirección	Función
1	P.G	--	Tierra de seguridad
2	TD	--> DCE	Salida de DTE
3	RD	--> DTE	Entrada de datos DTE
4	RTS	--> DCE	Petición de emisión DTE
5	CTS	--> DTE	Listo para transmitir DCE
6	DSR	--> DTE	CE listo para com. Con DTE
7	GND	--	Mása común del circuito

8	DCD	--> DTE	Detención de portadora
20	DTR	--> DCE	Señal de Terminal disponible
23	DSRD	--	Indicador de velocidad de TX.

TABLA 4 : DESCRIPCION DE PINES DE CONECTOR DB 25

### 3.4.1 Características

1. Normalmente estos suelen ser 2 en una placa base y son denominados COM 1 y COM2.
2. Estos puertos funcionan con un chip llamado UART, que es un controlador serie.
3. El término serie quiere decir que la comunicación con este tipo de conector se realiza sólo en una dirección: o envío, o recepción de datos, pero no las dos al mismo tiempo debido a que envía los datos uno detrás de otro.
4. El puerto serie utiliza direcciones y una línea de señales, un IRQ para llamar la atención del procesador. Además el Software de control debe conocer la dirección.
5. La mayoría de los puertos serie utilizan direcciones Standard predefinidas. Éstas están descritas normalmente en base hexadecimal.
6. Para el protocolo de transmisión de datos, sólo se tiene en cuenta dos estados de la línea, 0 y 1, también llamados Low y High.
7. El conector tiene sus extremos en ángulo de manera que el enchufe podrá introducirse de una manera solamente.

### 3.5 Ubicación en el sistema informático.

Se ubican en la parte trasera del case, podremos identificar estos puertos por los nombres COM 1, COM 2, COM 3. La cantidad de puertos de serie dependen de la tarjeta, ya que hay algunas tarjetas que son capaces de tener 4 u 8 puertos.

### 3.6 Puerto paralelo.

Este puerto de E/S envía datos en formato paralelo (donde 8 bits de datos, forman un byte, y se envían simultáneamente sobre ocho líneas individuales en un solo cable.) El puerto paralelo usa un conector tipo D-25 (es de 25 pin) El puerto paralelo se utiliza principalmente para impresoras.

La mayoría de los software usan el término LPT (impresor en línea) más un número para designar un puerto paralelo (por ejemplo, LPT1). Un ejemplo donde se utiliza la designación del puerto es el procedimiento de instalación de software donde se incluye un paso en que se identifica el puerto al cual se conecta a una impresora.

### **3.6.1 Forma de puerto paralelo.**

´-Es un conector de tipo hembra; los conectores hembras disponen de uno o más receptáculos diseñados para alojar las clavijas del conector macho.

-Mide 38mm de longitud en ambos extremos, de largo y de alto 5mm.

-Tiene forma de rectangular.

-Contiene 25 pines.

### **3.7 Puerto USB (Universal Serial Bus).**

El puerto USB fue creado a principio de 1996. La sigla USB significa Bus Serie Universal (Universal Serial Bus) Se llama universal, porque todos los dispositivos se conecten al puerto. Conexión que es posible, porque es capaz de hacer conectar hasta un total de 127 dispositivos.

Unas de las razones más importantes dieron origen a este puerto fueron:

´-Conexión del PC con el teléfono.

-Fácil uso.

-Expansión del puerto.

Unas de las principales características más importantes de este puerto es que permite la conexión entre el PC y el teléfono, además, nos elimina la incomodidad al momento de ampliar el PC.

Cabe destacar que para hacer esto, se necesita abrir el case e introducir las tarjetas de expansión o cualquier dispositivo deseado y después configurar y reiniciar el PC. Por lo tanto se puede decir que con este puerto tienes la capacidad de almacenar hasta de 127 dispositivos periféricos simultáneamente.

### **3.7.1 Características del Puerto USB.**

1. Todos los dispositivos USB tienen el mismo tipo de cable y el mismo tipo de conector, independientemente de la función que cumple.
2. Los detalles de consumo y administración electrónica del dispositivo son completamente transparentes para el usuario.
3. El computador identifica automáticamente un dispositivo agregado mientras opera, y por supuesto lo configura.
4. Los dispositivos pueden ser también desconectados mientras el computador está en uso.
5. Comparten un mismo bus tanto dispositivos que requieren de unos pocos KBPS como los que requieren varios MBPS.
6. Hasta 127 dispositivos diferentes pueden estar conectados simultáneamente y operando con una misma computadora sobre el Bus Serial Universal.
7. El bus permite periféricos multifunción, es decir aquellos que pueden realizar varias tareas a la vez, como lo son algunas impresoras que adicionalmente son fotocopiadoras y máquinas de fax.
8. Capacidad para manejo y recuperación de errores producido por un dispositivo cualquiera.
9. Soporte para la arquitectura conectar y operar (Plug&Play).
10. Bajo costo.

### **3.7. 2 Forma del Puerto USB.**

El puerto USB es el puerto más pequeño de los que existen en la parte trasera de nuestro ordenador. El conector USB, es un conector con tan sólo 4 pin. Este conector es individual, aunque también, nos podemos encontrar conectores compuestos para más de una conexión.

### **3.7.3 Ubicación en el Sistema Informático.**

El puerto USB está ubicado en la mayoría de los case en la parte frontal o lateral y en la parte trasera del mismo. Pero hay otros case que poseen este puerto únicamente en la parte trasera del case.

### 3.7.4 Tipos de Transferencia.

El puerto USB permite cuatro tipos de transferencia, que son:

- **Transferencias de control:**

Es una transferencia no esperada, no se realiza periódicamente, sino que la realiza el software para iniciar una petición/respuesta de comunicación. Normalmente se utiliza para operar operaciones de control o estado.

- **Transferencias Isocrónicas:**

Es periódica, una comunicación continua entre el controlador y el dispositivo, se usa normalmente para información.

Este tipo de transferencia envía la señal de reloj encapsulando en los datos, mediante comunicaciones NZRI.

- **Transferencias Continúa:**

Son datos pequeños no muy frecuentes, que provocan la espera de otras transferencias hasta que son realizadas.

- **Transferencias de Volumen:**

No son transferencias periódicas. Se trata de paquetes de gran tamaño, usados en aplicaciones donde se utiliza todo el ancho de banda disponible en la comunicación. Estas transferencias pueden quedar a la espera de que el ancho de banda quede disponible.

### 3.8 EL PIC 16F628A

Los microcontroladores PIC (Peripheral Interface Controller), son fabricados por la empresa MICROCHIP Technology INC, esta empresa ocupa el primer lugar en venta de microcontroladores desde el año 2002. El éxito radica en la gran variedad (más de 180 modelos) gran versatilidad, velocidad, bajo costo, bajo consumo de potencia y gran disponibilidad de herramientas de programación siendo uno de los más populares el PIC16F628A; pueden soportar hasta 100000 ciclos de escritura en su memoria Flash y 1000.000 de ciclos en su memoria Eeprom. Además de algunas ventajas adicionales como el oscilador interno RC de 4 mhz, MCLR programable, mayor capacidad de corriente; programación en bajo voltaje hacen de este microcontrolador más sencillo de ponerlo en funcionamiento para este trabajo final de graduación

### 3.8.1 Arquitectura del PIC 16F628A

Este microcontrolador utiliza un procesador con arquitectura Harvard, consiguiendo mayor rendimiento en el procesamiento de las instrucciones, está arquitectura a diferencia de la Von Neumann utiliza dos bloques de memorias independientes, una contiene instrucciones y la otra solo datos, cada una con su respectivo sistema de buses de acceso, 8 líneas para los datos y 14 líneas para las instrucciones, con lo que es posible realizar operaciones de acceso lectura escritura simultáneamente en las 2 memorias, a esto se conoce como paralelismo.

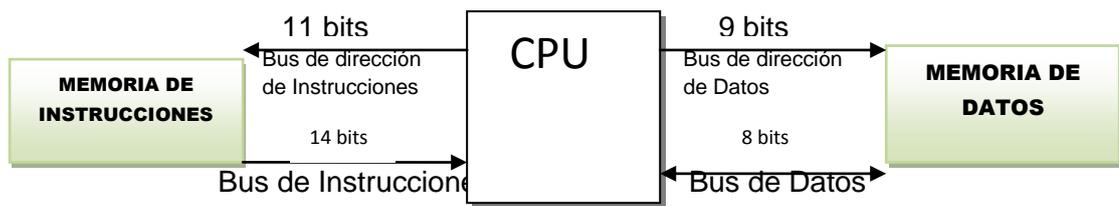


Fig 14: Arquitectura Harvard de PIC 16F628

El CPU del microcontrolador emplea una avanzada arquitectura RISC con un set de 35 instrucciones poderosas pertenecientes a la gama media de la familia e los microcontroladores PIC, la mayoría de instrucciones se ejecutan en un ciclo de instrucción a excepción de los saltos que requieren 2 ciclos, dentro de su procesador existe una pila de 8 niveles que permite el anidamiento de subrutinas, esto quiere decir que puede retomar 8 lugares diferentes de línea de programa e ir regresando a cada uno en el orden inverso al que fueron anidados.

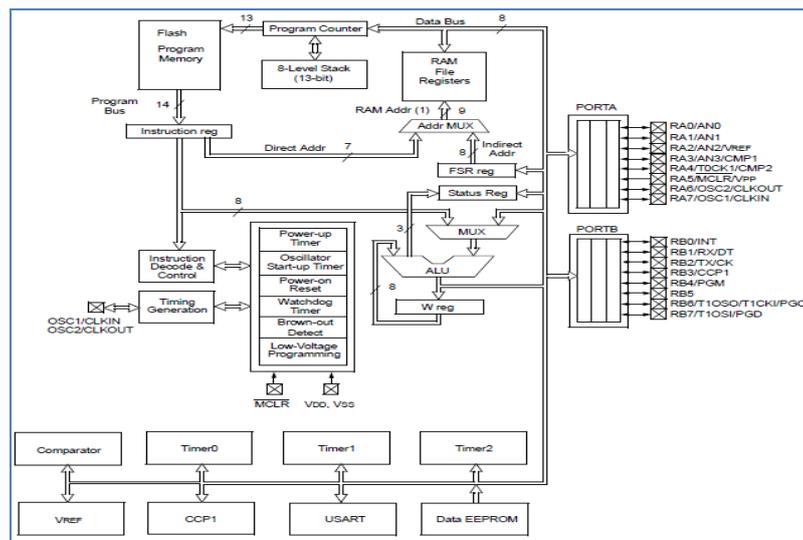


Fig 15: Diagrama de bloques funcionales del PIC 16F628

Fuente: PICs Programacion, 2009, Microcontroladores PIC Programación, Carlos A Reyes

### 3.8.2 Características Generales.

Se las puede resumir como las más relevantes a las siguientes:

- Velocidad de operación de hasta 20 MHz con oscilador externo.
- Oscilador interno RC (resistencia condensador) de 4MHz calibrado de fábrica al  $\pm 1\%$ .
- Admite 8 configuraciones de oscilador.
- 8 niveles de PILA.
- Procesador con arquitectura HARDVARD.
- Conjunto reducido de instrucciones RISC (35) gama media.
- Instrucciones de un ciclo excepto los saltos (200 ns por instrucción a 20 MHz).
- Resistencias PULL UP programables en el puerto B.
- Pin RA5 MCLR programable como reset externo o pin de entrada.
- Rango de operación desde 3V hasta 5.5 V.
- 15 pines de I/O y 1 solo de entrada RA5.
- Temporizador perro guardián WDT independiente del oscilador.
- Programable con bajo voltaje LPV (5V).
- Programación serial en circuito ICSP por pines:RB6 reloj y RB7 datos.
- Código de protección programable por sectores.
- Memoria de programa Flash 2048K de 100000 ciclos escritura/borrado.
- Memoria de datos EEPROM de 1000000 ciclos escritura/borrado de 100 años de retención.
- 2 circuitos comparadores análogos con entradas multiplexadas.
- 3 timers, timer 0 a 8 bits, timer 1 a 16 bits y timer 2 a 8 bits.
- Módulo CCP, captura compara 16 bits y PWM, modulación de ancho de pulso 10 bits.
- 10 fuentes de interrupción.
- Módulo de comunicación serial USART/SCI.
- Capacidad de corriente para encender leds directamente (25 mA I/O) por pin.

### 3.8.3 Diagrama de Pines y Funciones

Excluyendo los dos pines de alimentación, todos los 16 pines restantes pueden ser configurados como entradas o salidas, algunas de ellos tienen funciones especiales.

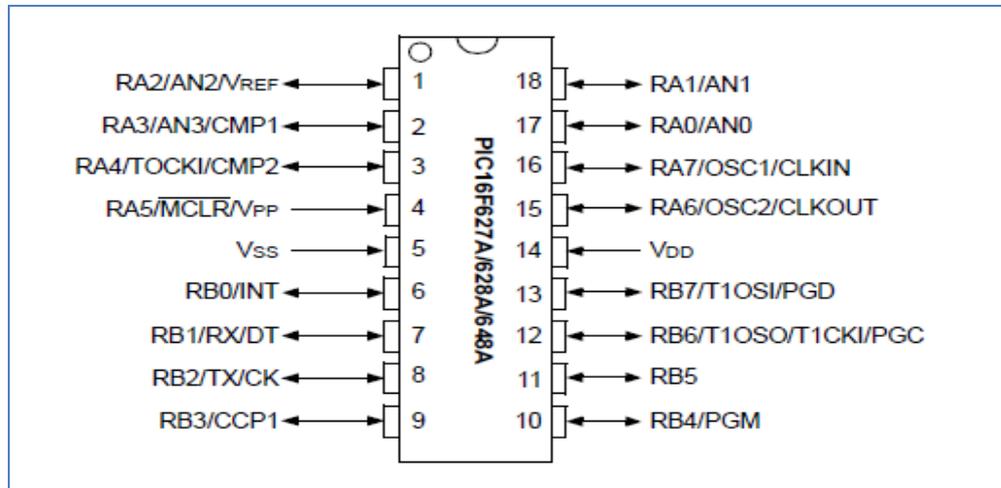


Fig 16 : Diagrama de Pines

Fuente:Microchip,2010, PIC 16F628A, [www.microchip.com/downloads](http://www.microchip.com/downloads)

	NOMBRE	DESCRIPCION
17	RA0/AN0	pin bidireccional I/O entrada comparador analógico
18	RA1/AN1	pin bidireccional I/O entrada comparador analógico
1	RA2/AN2/VREF	pin bidireccional I/O entrada comp. analógico y voltaje de referen.
2	RA3/AN3/CMP1	pin I/O entrada comp analógico y salida del comp. analógico 1
3	RA4/TOCKI/CLKOUT	pin I/O reloj TIMER 0 y salida del comparador analógico 2
4	RA5/MCLR/VPP	pin de entrada en modo MCLR activa RESET externo.
15	RA6/OSC2/CLKOUT	pin I/O entrada osc. externo salida de 1/4 de la frecuencia OSC 1.
16	RA7/OSC1/CLKIN	pin I/O entrada e oscilador externo, entrada del reloj externo.
6	RB0/INT	pin I/O resistencia pull-up programable entrada de interrupción ext,
	RB1/RX/DT	pin I/O resistencia pull-up entrada dato RS 232, I/O dato serial

7		asíncrono
8	RB2/TX/CK	pin I/O resistencia pull-up salida dato RS 232, I/O señal de reloj asincron.
9	RB3/CCP1	pin I/O resistencia pull-up Módulo CCP/PWM entrada o salida
10	RB4/PGM	pin I/O resistencia pull-up entrada de voltaje bajo de programación.
11	RB5	pin I/O resistencia pull-up programable.
12	RB6/T1OSO/T1CKI	pin I/O res. pull-up salida oscilador TIMER 1 entrada reloj de ICSP.
13	RB7/T1OSI	pin I/O resistencia pull-up entrada TIMER 1 , I/O datos de ICSP

TABLA 5 :Tabla de pines con sus funciones

### 3.9 Fuente de Alimentación.

En la cual regulamos el voltaje de entrada con la ayuda de los reguladores positivos 7812 y 7805, los cuales nos entregan una tensión continua de 12v y 5v que alimentará a todo el circuito

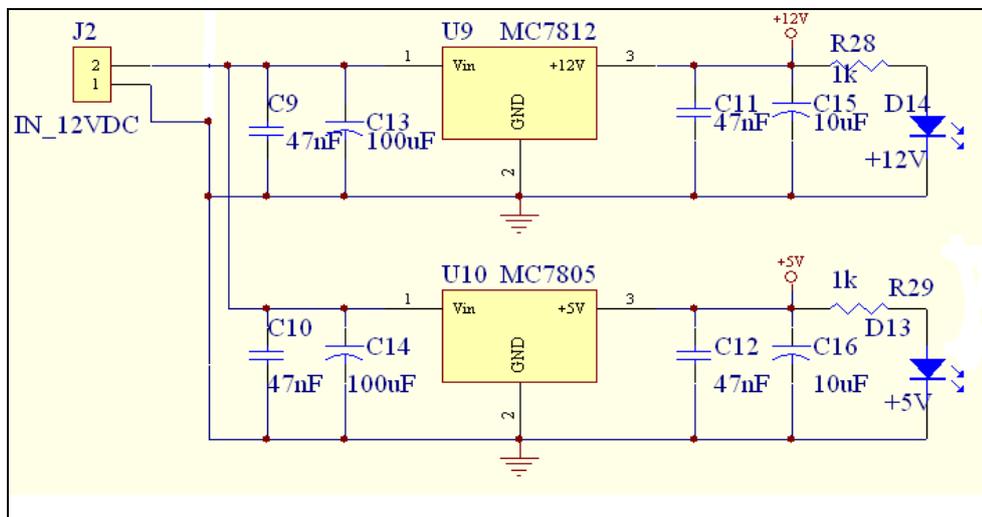


Fig 17 : Fuente de Alimentación

### 3.10 Detector de Señal de Timbre Telefónico.

Se diseño teniendo en cuenta que nuestro sistema de telefonía da como máximo un total de 6 timbres, y luego de esto se asienta el teléfono, la programación de nuestro PIC estará diseñada para contestar al quinto timbre; para ello el circuito electrónico está compuesto por dos partidores de tensión cuyos voltajes ingresan a

un circuito operacional, el cual al tener voltajes diferentes en sus entradas, activa la salida que comanda al transistor que cierra o abra el opto transistor cuya salida se conecta al microcontrolador para detectar la caída a 0v.

Como sabemos la línea telefónica mientras no recibe una llamada tiene un voltaje de aproximadamente 53v.

Cuando recibe una llamada este voltaje sube 65 v aproximadamente, si vemos el circuito, el primer partidor de tensión formado entre R4, R5, D1, es un partidor Fijo cuyo valor de tensión a su salida se mantiene estable. Caso contrario el partidor formado por R1, R25, R22 y D7, es un partidor variable que depende el voltaje en la entrada que variará entre 53 y 65 voltios. Mientras no existe llamada entrante el valor a la salida del operacional TL081 es 0v. Sin embargo cuando existe una llamada el voltaje aumenta y el partidor variable genera un voltaje diferente al partidor fijo, produciéndose un voltaje de salida que cierra el opto acoplador 4N35, haciendo que la salida del opto acoplador se vaya a tierra y por tanto el microcontrolador puede sensar la cantidad de pulsos a bajo, para activar el relé que cierra la resistencia de 560  $\Omega$  simulando así la impedancia para el alza del auricular.

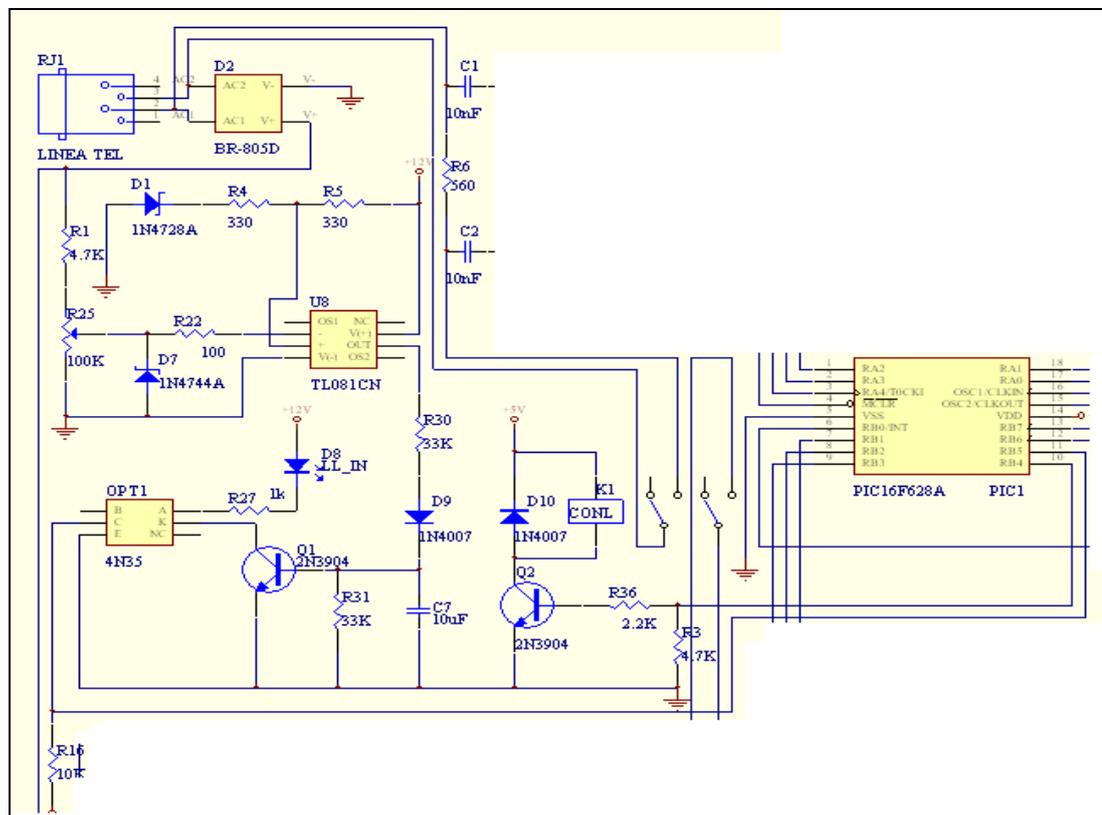


Fig 18 : Circuito Detector de Timbre Telefónico

### 3.11 EL PIC como decodificador de señales BDC

El integrado 8870, es el encargado con la ayuda de dos filtros conformados por C1, R7 y C2, R8 de filtrar el rango de frecuencias asignadas a los tonos de el teclado telefónico. Estos filtros no son más que dos filtros pasa bandas tanto para la frecuencia baja como para la frecuencia alta que formará el tono. Los zener D3, y D4 se conectan en anti paralelo a la línea para generar el voltaje de referencia que se unirá a la tierra de todo el circuito para que el 8870 pueda decodificar los tonos entrantes.

Cuando el relé k1 se cierra todos los pulsos que se reciben se transforman en código BCD de 4 bits que serán leídos en las entradas del puerto A, pines 17(RA0), pin 18(RA1), pin 1 (RA2), pin 2 (RA3), del PIC16F628, ahora para que ese código pueda ser entendido y procesado en una subrutina del microcontrolador debemos convertirlo a código de 8 bits esto se lo realiza agregando 4 bits y dejando de derecha a izquierda los bits más significativos, a esto se lo realiza utilizando el comando AND para que luego el PIC confirme la tecla ingresada si es correcta y lo guarde en la subrutina correspondiente denominada GUARDAR\_DATO

GUARDAR\_DATO

.\*\*\*\*\*

```

BTFSC    PORTA,4
GOTO     GUARDAR_DATO
MOVF     PORTA,W
MOVWF    MEMORIA
MOVLW   B'00001111'
ANDWF    MEMORIA,F
RETURN

```

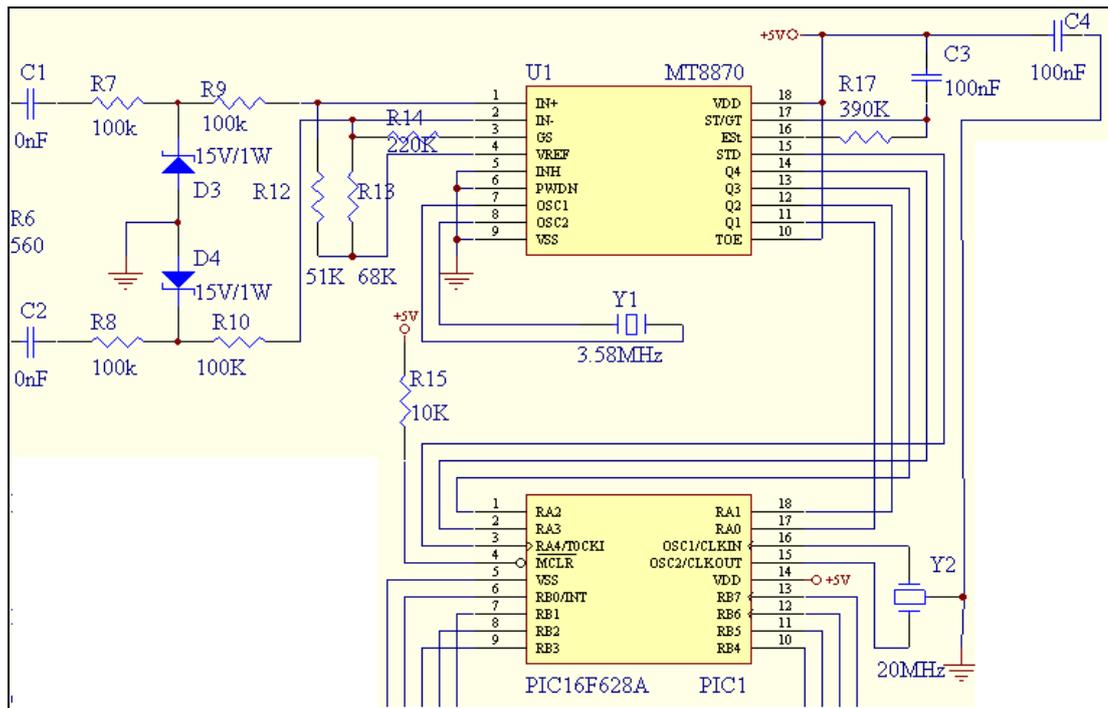


Fig 19: Circuito de Decodificación BCD

### 3.12 Circuito Modulador de Tonos

Para confirmar el hecho de que las claves se han digitado correctamente es necesario generar un tono y poder transmitirlo por la línea telefónica para que el usuario confirme su acción. Para ello se desarrollo un circuito modulador conformado por R33, R34, C5, Q6, R38, R39, D12, C8, R11, R26, C6 R2, R35 Y Q3. El tono es generado en el microcontrolador, mediante la salida RB3 que oscila entre 1 y 0 con intervalos de 500us, por 200 veces, ya que

$$\text{FRECUENCIA} = 1/\text{PERIODO}$$

$$F = 1 / 0,000500$$

$$F = 2000\text{HZ} = 2\text{KHz}$$

Por tanto hemos generado un tono dentro del rango audible para el ser humano que va desde los 20 y 20KHz. Este tono no es más que un tren de pulsos, que debe primero ser modulado para que pueda ser escuchado en la línea telefónica, para ello la señal entrante primero es amplificada gracias a la configuración de Q6 y luego es modulada gracias a Q3, transmitiéndose la señal modulada hacia la línea telefónica mediante el positivo del puente de greatz que elimina las componente alterna de la señal telefónica, y está polarizado directamente sobre la

línea. Cabe destacar que los transistores Q4 y Q5 están conectados en configuración Darlington para amplificar el tono generado.

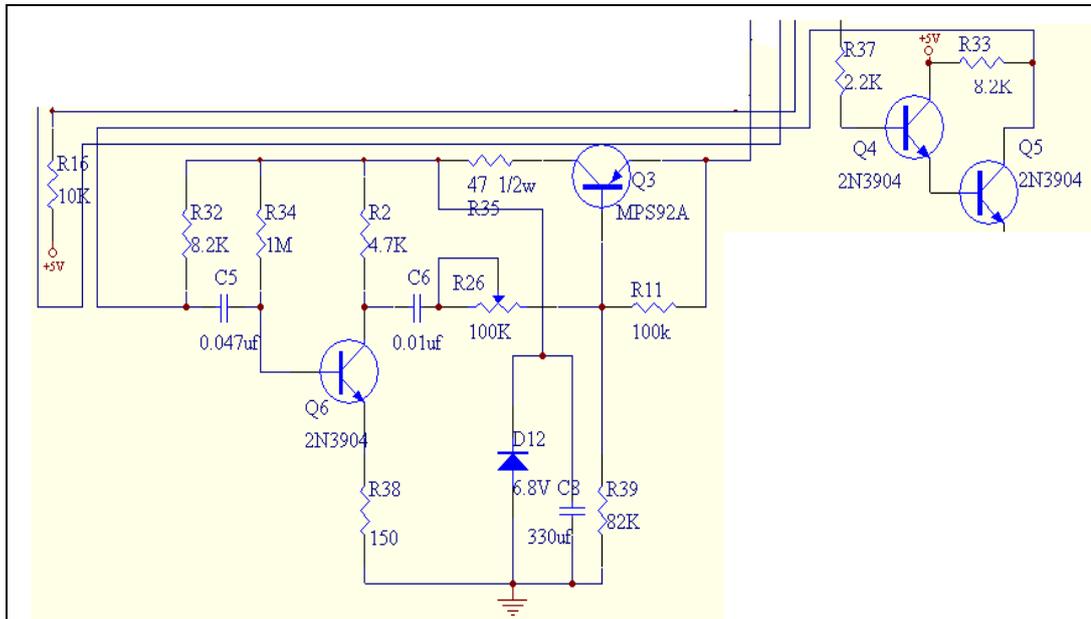


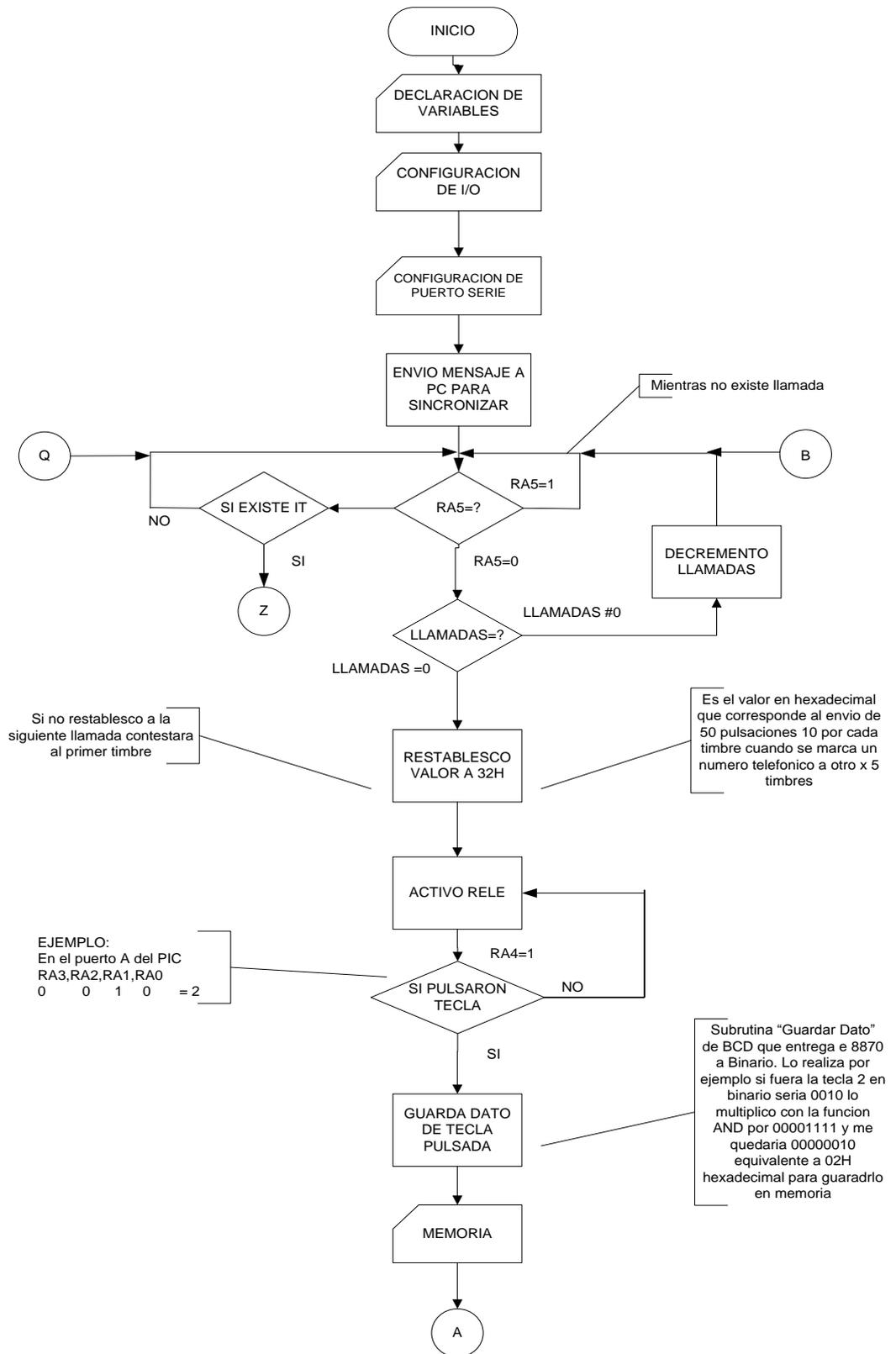
Fig 20: Circuito Modulador de Tonos

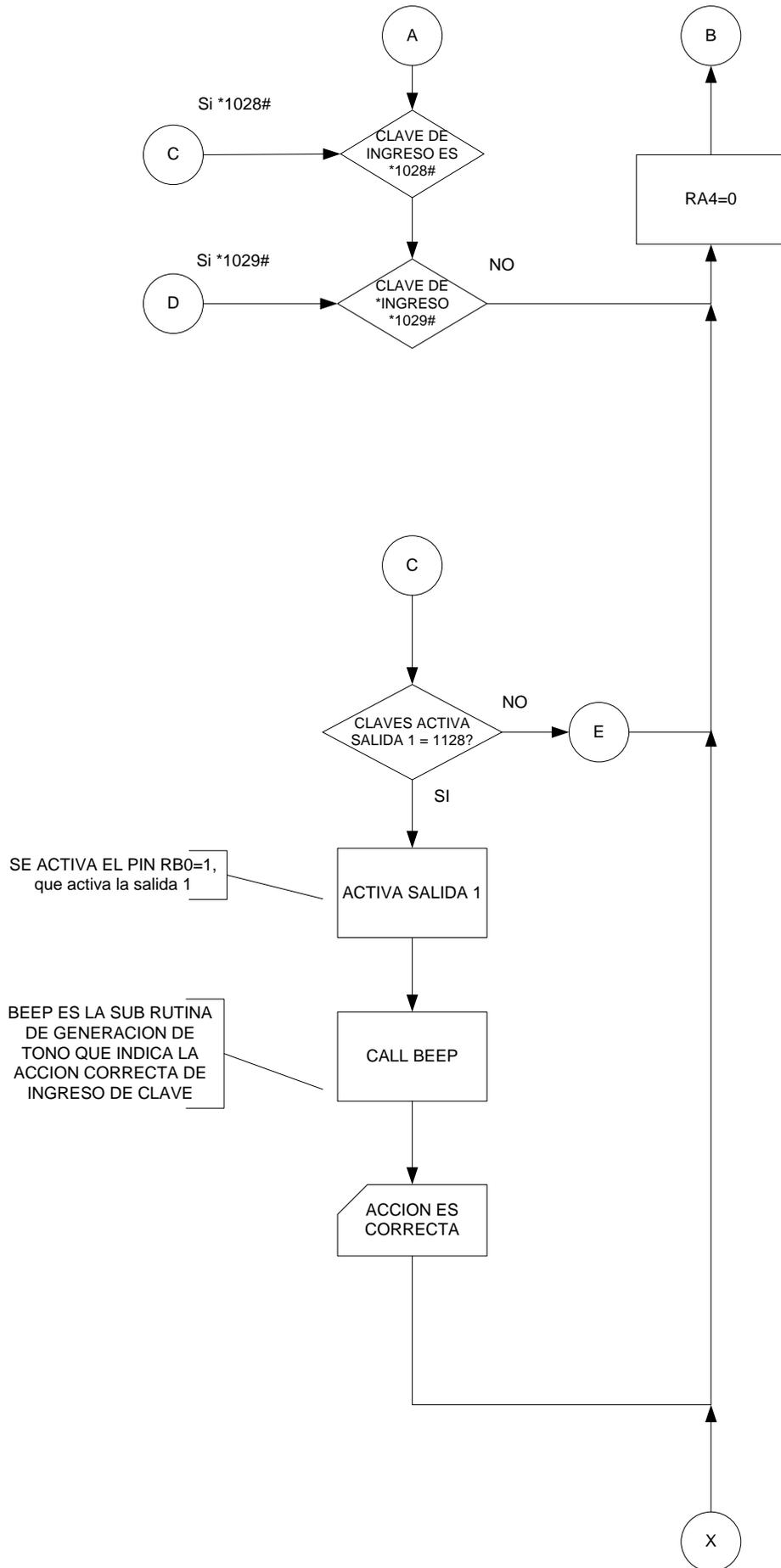
### 3.13 Flujograma de Programación del Circuito Control.

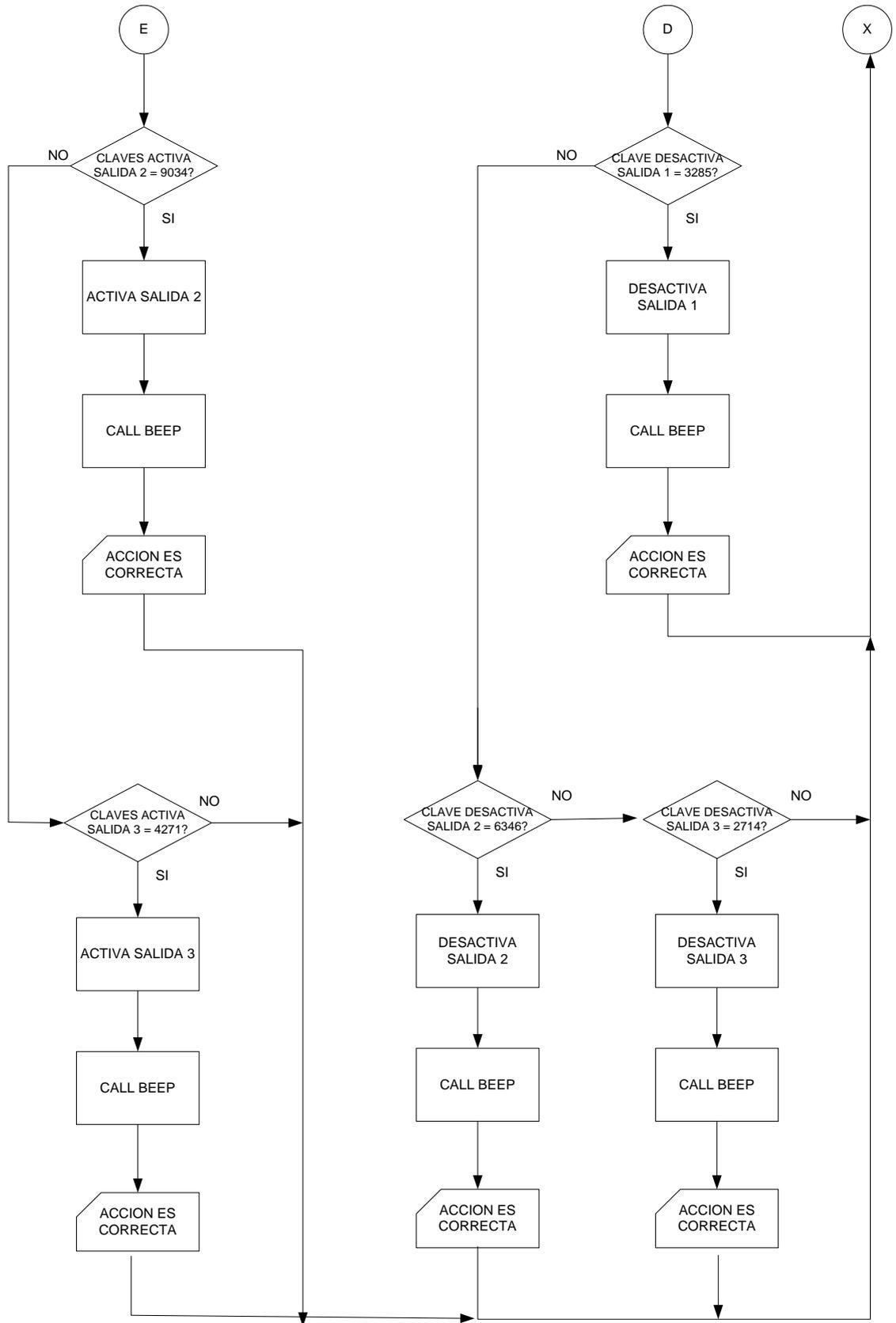
Está desarrollado en función del PIC, y es el encargado de controlar y generar el proceso lógico para a activación desactivación de las salidas, mediante la llamada telefónica y/o mediante el software de control por el puerto serial. El proceso es el siguiente:

La entrada del pin 11 (RB5) mientras está en alto espera en un bucle cerrado que el voltaje se vaya a nivel bajo para contar el número de veces que timbra el teléfono, en este caso se programo para que a la quinta vez que suene el teléfono la salida pin 10 (RB4) se activara y el relé se cerrará, colocando en paralelo la resistencia de 560 que simula la impedancia de levantado del auricular. Para explicar más eficientemente todo el proceso ponemos a consideración los siguientes gráfico:

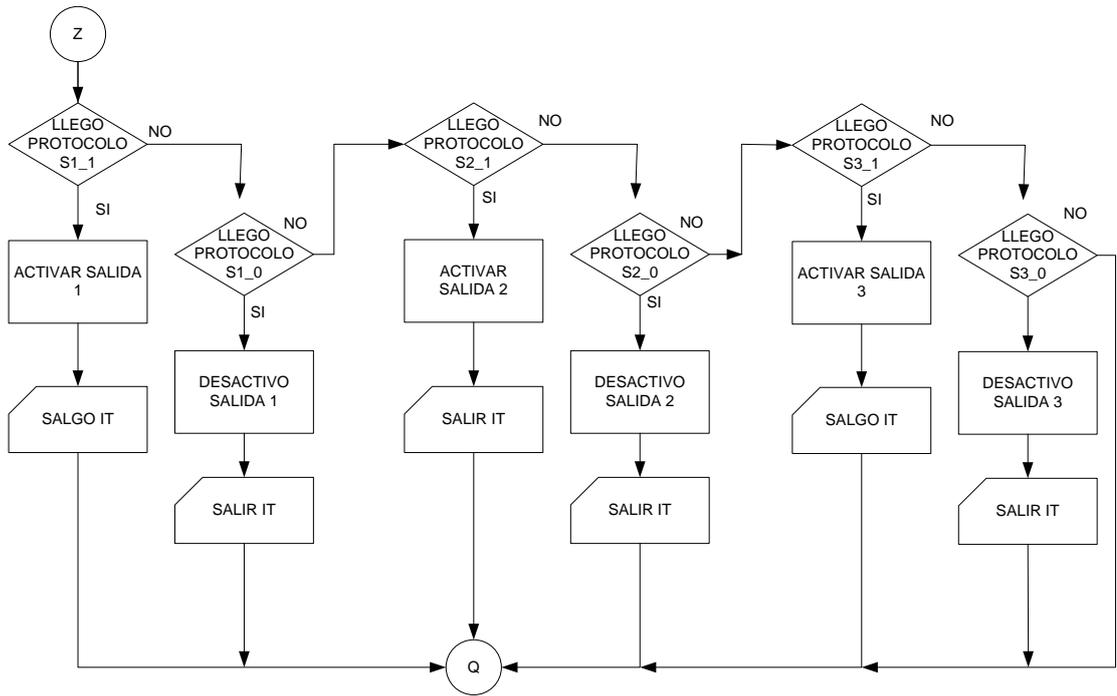
Diagrama de flujo de programación de PIC 16F628A







PROTOCOLO DE TRASMISION  
 PARA COMUNICACION CON  
 LA PC A TRAVEZ DEL PUERTO  
 SERIAL:  
 S1\_1= ACTIVA SALIDA 1  
 S1\_0=DESACTIVO SALIDA 1  
 S2\_1=ACTIVO SALIDA 2  
 S2\_0=DESACTIVO SALIDA 2  
 S3\_1 ACTIVO SALIDA 3  
 S3\_0 DESACTIVO SALIDA 3



### 3.14 Código Fuente de Programa de Control del PIC 16F628A

```

;PROGRAMACIÓN DE PIC 16F28A
;ANDRES ARGUDO
;TESIS DE TECNOLOGIA ELECTRÓNICA
;MANDO REMOTO DE UNA CASA VÍA TELEFÓNICA
;*****DECLARACION DEL MICROCONTROLADOR A UTILIZAR*****
list p=16F628A
#include <p16F628A.inc>
errorlevel -302

;*****PALABRA DE CONFIGURACION*****
;
;   __CONFIG_CP_OFF&_PWRTE_ON&_WDT_OFF&_LVP_OFF&_BODEN_OFF&_HS_OSC
;*****
;*****VARIABLES A UTILIZAR EN NUESTRO PROGRAMA*****
;
MEMORIA      EQU   0X20
PASO         EQU   0X21
ACT          EQU   0X22
DESACT       EQU  0X23
N            EQU  0X24
M            EQU  0X25
CONT1        EQU  0X26
CONT2        EQU  0X27
CONT3        EQU  0X28
LLAMADAS     EQU  0X29
RESP_RX      EQU  0X2A
WTEMP        EQU  0X2B
RESP_SAL1    EQU  0X2C
RESP_SAL2    EQU  0X2D
RESP_SAL3    EQU  0X2E
EST          EQU  0X2F
BEEP         EQU  0X30
;*****
;
;   ORG      0
;   GOTO INICIO
;   ORG      4
;   GOTO IT
;
;.....
;.....INTERRUPCION;.....
IT
;-----
;           MOVWF      WTEMP
;-----
;           BTFSS PIR1,RCIF      ;CONFIRMO IT
;           GOTO SALIR_IT
;-----
;           BSF      STATUS,RP0
;           BCF      PIE1,5      ;DESABILITO IT POR RX DE USART
;           BCF      STATUS,RP0
;-----
I1          BTFSS PIR1,RCIF
;           GOTO I1
;           MOVF      RCREG,W
;           MOVWF    RESP_RX
;           MOVLW 'S'
;           SUBWF    RESP_RX,W
;           BTFSC    STATUS,2
;           GOTO I2
;-----
;           MOVLW 'E'
;           SUBWF    RESP_RX,W
;           BTFSS    STATUS,2
;           GOTO SALIR_IT
I9          BTFSS PIR1,RCIF
;           GOTO I9

```

```

        MOVF  RCREG,W
        MOVWF RESP_RX
        MOVLW 'S'
        SUBWF RESP_RX,W
        BTFSS STATUS,2
        GOTO  I9
I10    BTFSS  PIR1,RCIF
        GOTO  I10
        MOVF  RCREG,W
        MOVWF RESP_RX
        MOVLW 'T'
        SUBWF RESP_RX,W
        BTFSS STATUS,2
        GOTO  I10
;-----
        BSF      EST,0
        GOTO  SALIR_IT
;-----
I2     BTFSS  PIR1,RCIF
        GOTO  I2
        MOVF  RCREG,W
        MOVWF RESP_RX
        MOVLW '1'
        SUBWF RESP_RX,W
        BTFSS STATUS,2
        GOTO  CASO_1
I3     BTFSS  PIR1,RCIF
        GOTO  I3
        MOVF  RCREG,W
        MOVWF RESP_RX
        MOVLW '_'
        SUBWF RESP_RX,W
        BTFSS STATUS,2
        GOTO  I3
I4     BTFSS  PIR1,RCIF
        GOTO  I4
        MOVF  RCREG,W
        MOVWF RESP_RX
        MOVLW '1'
        SUBWF RESP_RX,W
        BTFSS STATUS,2
        GOTO  CASO_3
;-----
        BSF      PORTB,0           ;ACTIVO LA SALIDA 1
        BSF      RESP_SAL1,0
        GOTO  SALIR_IT
;-----
CASO_3
;-----
        MOVLW '0'
        SUBWF RESP_RX,W
        BTFSS STATUS,2
        GOTO  SALIR_IT
;-----
        BCF      PORTB,0           ;DESACTIVO LA SALIDA 1
        BCF      RESP_SAL1,0
        GOTO  SALIR_IT
;-----
CASO_1
;-----
        MOVLW '2'
        SUBWF RESP_RX,W
        BTFSS STATUS,2
        GOTO  CASO_2
I5     BTFSS  PIR1,RCIF
        GOTO  I5
        MOVF  RCREG,W

```

```

MOVWF RESP_RX
MOVLW '_'
SUBWF RESP_RX,W
BTFSS STATUS,2
GOTO I5
16 BTFSS PIR1,RCIF
GOTO I6
MOVF RCREG,W
MOVWF RESP_RX
MOVLW '1'
SUBWF RESP_RX,W
BTFSS STATUS,2
GOTO CASO_4
;-----
BSF PORTB,6 ;ACTIVO LA SALIDA 2
BSF RESP_SAL2,0
GOTO SALIR_IT
;-----
CASO_4
;-----
MOVLW '0'
SUBWF RESP_RX,W
BTFSS STATUS,2
GOTO SALIR_IT
;-----
BCF PORTB,6 ;DESACTIVO LA SALIDA 2
BCF RESP_SAL2,0
GOTO SALIR_IT
;-----
CASO_2
;-----
MOVLW '3'
SUBWF RESP_RX,W
BTFSS STATUS,2
GOTO SALIR_IT
17 BTFSS PIR1,RCIF
GOTO I7
MOVF RCREG,W
MOVWF RESP_RX
MOVLW '_'
SUBWF RESP_RX,W
BTFSS STATUS,2
GOTO I7
18 BTFSS PIR1,RCIF
GOTO I8
MOVF RCREG,W
MOVWF RESP_RX
MOVLW '1'
SUBWF RESP_RX,W
BTFSS STATUS,2
GOTO CASO_5
;-----
BSF PORTB,7 ;ACTIVO LA SALIDA 3
BSF RESP_SAL3,0
GOTO SALIR_IT
;-----
CASO_5
;-----
MOVLW '0'
SUBWF RESP_RX,W
BTFSS STATUS,2
GOTO SALIR_IT
;-----
BCF PORTB,7 ;DESACTIVO LA SALIDA 2
BCF RESP_SAL3,0
;-----
SALIR_IT

```

```

;-----
MOVF WTEMP,W
BSF STATUS,RP0 ;BANCO 1
BSF PIE1,5 ;HABILITO LA IT POR RX DEL USART
BCF STATUS,RP0 ;BANCO 0
CLRF RCREG
CLRF RESP_RX
CLRF WTEMP
RETFIE
;*****
;
INICIO
;*****
;
BSF STATUS,RP0 ;BANCO 1
MOVLW B'00111111' ;RA0,RA1,RA2,RA3,INGRESOS CÓDIGO
BINARIO
MOVWF TRISA ;RA4, INGRESO SEÑAL DE DURACION DE
PULSO
MOVLW B'00100000' ;RB0 = SALIDA CONTROL 1, RB1 = RX, RB2 =
TX, RB6 = SALIDAS DE CONTROL 2, RB7 = SALIDA DE CONTROL 3
MOVWF TRISB ;RB4, SALIDA RELE, RB5, ENTRADA
DETECTOR DE LLAMADAS
MOVLW B'10000111' ;PULL- UP DESABILITADAS
MOVWFOPTION_REG
BCF STATUS,RP0 ;BANCO 0
MOVLW B'01000000' ;HABILITACION DE IT DE PERIFERICOS = ON,
INTERRUPCION POR EL PIN RB0 OFF;INTERRUPCIONES GLOBALES = OFF
MOVWFINTCON
MOVLW B'00000111'
MOVWFCMCON
CLRF PORTA
CLRF PORTB
CLRF MEMORIA
CLRF ACT
CLRF DESACT
CLRF RESP_RX
CLRF WTEMP
CLRF RESP_SAL1
CLRF RESP_SAL2
CLRF RESP_SAL3
CLRF EST
MOVLW 32H ;19H
MOVWFLLAMADAS
MOVLW 0C8H
MOVWFBEEP
CALL DEM_200ms
BSF STATUS,5 ;BANCO 1
;*****
;
CONFIG_PTO_SERIE ;PARÁMETROS DEL PUERTO SERIE
;*****
;
MOVLW b'00100010' ;TRANSMISION DE 8 BITS, TX
HABILITADA,MODO ASINCRONO,BAJA VALOCIDAD
MOVWFTXSTA
MOVLW d'36' ;VALOR 32,VELOCIDAD DE 9600 BPS PARA 20 Mhz
MOVWFSPBRG
MOVLW B'00100000' ;IT POR RX DE USART
MOVWFPIE1
BCF STATUS,RP0 ;BANCO 0
MOVLW b'10010000'
MOVWFRCSSTA ;PUERTO SERIAL HABILITADO,8 BITS, RECEPCION
CONTINUA HABILITADA
;-----
MOVLW "S"
CALL ENVÍAR
MOVLW "I"
CALL ENVÍAR
MOVLW "S"
CALL ENVÍAR

```

```

MOVLW      "T"
CALL  ENVÍAR
MOVLW      "E"
CALL  ENVÍAR
MOVLW      "M"
CALL  ENVÍAR
MOVLW      "A"
CALL  ENVÍAR
MOVLW      " "
CALL  ENVÍAR
MOVLW      "E"
CALL  ENVÍAR
MOVLW      "N"
CALL  ENVÍAR
MOVLW      " "
CALL  ENVÍAR
MOVLW      "L"
CALL  ENVÍAR
MOVLW      "I"
CALL  ENVÍAR
MOVLW      "N"
CALL  ENVÍAR
MOVLW      "E"
CALL  ENVÍAR
MOVLW      "A"
CALL  ENVÍAR
CALL  RET_CARRO
MOVLW      "C"
CALL  ENVÍAR
MOVLW      "O"
CALL  ENVÍAR
MOVLW      "N"
CALL  ENVÍAR
MOVLW      "T"
CALL  ENVÍAR
MOVLW      "R"
CALL  ENVÍAR
MOVLW      "O"
CALL  ENVÍAR
MOVLW      "L"
CALL  ENVÍAR
MOVLW      " "
CALL  ENVÍAR
MOVLW      "D"
CALL  ENVÍAR
MOVLW      "O"
CALL  ENVÍAR
MOVLW      "M"
CALL  ENVÍAR
MOVLW      "O"
CALL  ENVÍAR
MOVLW      "T"
CALL  ENVÍAR
MOVLW      "I"
CALL  ENVÍAR
MOVLW      "C"
CALL  ENVÍAR
MOVLW      "O"
CALL  ENVÍAR
CALL  RET_CARRO
MOVLW      "A"
CALL  ENVÍAR
MOVLW      "N"
CALL  ENVÍAR
MOVLW      "D"
CALL  ENVÍAR
MOVLW      "R"

```

```

CALL ENVÍAR
MOVLW "E"
CALL ENVÍAR
MOVLW "S"
CALL ENVÍAR
MOVLW " "
CALL ENVÍAR
MOVLW "A"
CALL ENVÍAR
MOVLW "R"
CALL ENVÍAR
MOVLW "G"
CALL ENVÍAR
MOVLW "U"
CALL ENVÍAR
MOVLW "D"
CALL ENVÍAR
MOVLW "O"
CALL ENVÍAR
MOVLW " "
CALL ENVÍAR
MOVLW "V"
CALL ENVÍAR
MOVLW "1"
CALL ENVÍAR
MOVLW "."
CALL ENVÍAR
MOVLW "1"
CALL ENVÍAR
MOVLW "1"
CALL RET_CARRO
;-----
PROG_PRIN_ADD
;-----
CLRF PASO
BCF PORTB,4
;-----
PROG_PRIN
;-----
BTFSS PORTB,5
GOTO P0
;-----
P0 CALL ESTÁDO
GOTO PROG_PRIN
BTFSS PORTB,5
GOTO P0
CALL DEM_100ms
DECF LLAMADAS
BTFSS STATUS,2
GOTO PROG_PRIN
BSF PORTB,4
MOVLW 09H
MOVWF LLAMADAS
CALL ESPERA
CALL GUARDAR_DATO
CALL CLAVE_INGRESO
BTFSC PASO,0

```

```

GOTO PROG_PRIN_ADD
BTFSS ACT,0
GOTO CASO_DESACTIVAR
CLRF ACT
CALL ACT_SALIDAS
GOTO PROG_PRIN_ADD
;-----
ESTÁDO
;-----
BTFSS EST,0
RETURN
;-----
CLRF EST
;-----
MOVLW "E"
CALL ENVÍAR
MOVLW "S"
CALL ENVÍAR
MOVLW "T"
CALL ENVÍAR
MOVLW ":"
CALL ENVÍAR
MOVLW " "
CALL ENVÍAR
MOVLW "S"
CALL ENVÍAR
MOVLW "1"
CALL ENVÍAR
MOVLW " _"
CALL ENVÍAR
BTFSS RESP_SAL1,0
GOTO MEN_1
MOVLW "O"
CALL ENVÍAR
MOVLW "N"
CALL ENVÍAR
MOVLW ","
CALL ENVÍAR
GOTO SALIDA_2
;-----
MEN_1
;-----
MOVLW "O"
CALL ENVÍAR
MOVLW "F"
CALL ENVÍAR
MOVLW "F"
CALL ENVÍAR
MOVLW ","
CALL ENVÍAR
;-----
SALIDA_2
;-----
MOVLW "S"
CALL ENVÍAR
MOVLW "2"
CALL ENVÍAR
MOVLW " _"
CALL ENVÍAR
BTFSS RESP_SAL2,0
GOTO MEN_2
MOVLW "O"
CALL ENVÍAR
MOVLW "N"
CALL ENVÍAR
MOVLW ","
CALL ENVÍAR

```

```

                GOTO SALIDA_3
;-----
MEN_2
;-----
    MOVLW      "O"
    CALL ENVÍAR
    MOVLW      "F"
    CALL ENVÍAR
    MOVLW      "F"
    CALL ENVÍAR
    MOVLW      " ,"
    CALL ENVÍAR
;-----
SALIDA_3
;-----
    MOVLW      "S"
    CALL ENVÍAR
    MOVLW      "3"
    CALL ENVÍAR
    MOVLW      " _"
    CALL ENVÍAR
    BTFSS RESP_SAL3,0
    GOTO MEN_3
    MOVLW      "O"
    CALL ENVÍAR
    MOVLW      "N"
    CALL ENVÍAR
    MOVLW      " ,"
    CALL ENVÍAR
    CALL RET_CARRO
    GOTO PROG_PRIN
;-----
MEN_3
;-----
    MOVLW      "O"
    CALL ENVÍAR
    MOVLW      "F"
    CALL ENVÍAR
    MOVLW      "F"
    CALL ENVÍAR
    MOVLW      " ,"
    CALL ENVÍAR
    CALL RET_CARRO
    RETURN
;-----
CASO_DESACTIVAR
;-----
    BTFSS DESACT,0
    GOTO PROG_PRIN_ADD
    CLRF DESACT
    CALL DESACT_SALIDAS
    GOTO PROG_PRIN_ADD
;-----
;*****
ESPERA
;*****
    BTFSS PORTA,4
    GOTO ESPERA
    RETURN
;*****
GUARDAR_DATO
;*****
    BTFSC PORTA,4
    GOTO GUARDAR_DATO
    MOVF PORTA,W
    MOVWF MEMORIA
    MOVLW B'00001111'

```

```

ANDWF MEMORIA,F
RETURN
;*****
;CLAVE_INGRESO                                ;LA CLAVE POR DEFECTO ES *1028#
PARA ACTIVAR Y *1129# PARA DESACTIVAR
;*****
;
;   MOVLW B'00001011'                            ;CÓDIGO BINARIO ASIGNADO A LA TECLA *
;   SUBWF MEMORIA,W
;   BTFSC STATUS,2
;   GOTO  TECLA_1
;   BSF      PASO,0
;   RETURN
;-----
;TECLA_1
;-----
;   CALL  ESPERA
;   CALL  GUARDAR_DATO
;   MOVLW B'00000001'                            ;CÓDIGO BINARIO ASIGNADO A LA TECLA 1
;   SUBWF MEMORIA,W
;   BTFSC STATUS,2
;   GOTO  TECLA_0
;   BSF      PASO,0
;   RETURN
;-----
;TECLA_0
;-----
;   CALL  ESPERA
;   CALL  GUARDAR_DATO
;   MOVLW B'00001010'                            ;CÓDIGO BINARIO ASIGNADO A LA TECLA 0
;   SUBWF MEMORIA,W
;   BTFSC STATUS,2
;   GOTO  TECLA_2
;   BSF      PASO,0
;   RETURN
;-----
;TECLA_2
;-----
;   CALL  ESPERA
;   CALL  GUARDAR_DATO
;   MOVLW B'00000010'                            ;CÓDIGO BINARIO ASIGNADO A LA TECLA 2
;   SUBWF MEMORIA,W
;   BTFSC STATUS,2
;   GOTO  TECLA_8_9
;   BSF      PASO,0
;   RETURN
;-----
;TECLA_8_9
;-----
;   CALL  ESPERA
;   CALL  GUARDAR_DATO
;   MOVLW B'00001000'                            ;CÓDIGO BINARIO ASIGNADO A LA TECLA 8
;   SUBWF MEMORIA,W
;   BTFSC STATUS,2
;   GOTO  TECLA_#_A
;   MOVLW B'00001001'                            ;CÓDIGO BINARIO ASIGNADO A LA TECLA 9
;   SUBWF MEMORIA,W
;   BTFSC STATUS,2
;   GOTO  TECLA_#_D
;   BSF      PASO,0
;   RETURN
;-----
;TECLA_#_A
;-----
;   BSF      ACT,0
;   CALL  ESPERA
;   CALL  GUARDAR_DATO
;   MOVLW B'00001100'                            ;CÓDIGO BINARIO ASIGNADO A LA TECLA #

```

```

SUBWF MEMORIA,W
BTFSC STATUS,2
RETURN
BSF PASO,0
CLRF ACT
RETURN
;-----
TECLA_#_D
;-----
BSF DESACT,0
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00001100' ;CÓDIGO BINARIO ASIGNADO A LA TECLA #
SUBWF MEMORIA,W
BTFSC STATUS,2
RETURN
BSF PASO,0
CLRF DESACT
RETURN
.*****
ACT_SALIDAS ;SALIDA1:1128 SALIDA2:9034
SALIDA3:4271
.*****
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000001' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 1
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO ACT_SALIDA1
MOVLW B'00001001' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 9
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO ACT_SALIDA2
MOVLW B'00000100' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 4
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO ACT_SALIDA3
BSF PASO,1
RETURN
.*****
ACT_SALIDA1
.*****
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000001' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 1
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S1_T2
BSF PASO,1
RETURN
;-----
S1_T2
;-----
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000010' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 2
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S1_T8_T9
BSF PASO,1
RETURN
;-----
S1_T8_T9
;-----
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00001000' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 8

```

```

SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S1_ACT
MOVLW B'00001001'
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S1_ACT_DEM
BSF PASO,1
RETURN
;-----
S1_ACT
;-----
BSF PORTB,0
BSF RESP_SAL1,0
CALL S_BEEP
RETURN
;-----
S1_ACT_DEM
;-----
BSF PORTB,0
BSF RESP_SAL1,0
CALL S_BEEP
CALL DEM_2s
BCF PORTB,0
RETURN
.*****
ACT_SALIDA2
.*****
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00001010'
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S2_T3
BSF PASO,1
RETURN
;-----
S2_T3
;-----
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000011'
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S2_T4_T5
BSF PASO,1
RETURN
;-----
S2_T4_T5
;-----
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000100'
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S2_ACT
MOVLW B'00000101'
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S2_ACT_DEM
BSF PASO,1
RETURN
;-----
S2_ACT
;-----
BSF PORTB,6
BSF RESP_SAL2,0

```

;CÓDIGO BINARIO ASIGNADO A LA TECLA 9

;CÓDIGO BINARIO ASIGNADO A LA TECLA 0

;CÓDIGO BINARIO ASIGNADO A LA TECLA 3

;CÓDIGO BINARIO ASIGNADO A LA TECLA 4

;CÓDIGO BINARIO ASIGNADO A LA TECLA 5

```

CALL S_BEEP
RETURN
;-----
S2_ACT_DEM
;-----
BSF PORTB,6
BSF RESP_SAL2,0
CALL S_BEEP
CALL DEM_2s
BCF PORTB,6
RETURN
;*****
ACT_SALIDA3
;*****
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000010' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 2
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S3_T7
BSF PASO,1
RETURN
;-----
S3_T7
;-----
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000111' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 7
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S3_T1_T2
BSF PASO,1
RETURN
;-----
S3_T1_T2
;-----
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000001' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 1
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S3_ACT
MOVLW B'00000010' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 2
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S3_ACT_DEM
BSF PASO,1
RETURN
;-----
S3_ACT
;-----
BSF PORTB,7
BSF RESP_SAL3,0
CALL S_BEEP
RETURN
;-----
S3_ACT_DEM
;-----
BSF PORTB,7
BSF RESP_SAL3,0
CALL S_BEEP
CALL DEM_2s
BCF PORTB,7
RETURN
;*****
DESACT_SALIDAS ;SALIDA1:3285 SALIDA2:6346
SALIDA3:2714

```

```

*****
;
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000011' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 3
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO DESACT_SALIDA1
MOVLW B'00000110' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 6
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO DESACT_SALIDA2
MOVLW B'00000010' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 2
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO DESACT_SALIDA3
BSF PASO,1
RETURN
*****
;
DESACT_SALIDA1
*****
;
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000010' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 2
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO D_S1_T8
BSF PASO,1
RETURN
;-----
;
D_S1_T8
;-----
;
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00001000' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 8
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO D_S1_T5
BSF PASO,1
RETURN
;-----
;
D_S1_T5
;-----
;
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000101' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 5
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S1_DESACT
BSF PASO,1
RETURN
;-----
;
S1_DESACT
;-----
;
BCF PORTB,0
BCF RESP_SAL1,0
CALL S_BEEP
RETURN
*****
;
DESACT_SALIDA2
*****
;
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000011' ;CÓDIGO BINARIO ASIGNADO A LA TECLA 3
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO D_S2_T4
BSF PASO,1

```

```

RETURN
;-----
D_S2_T4
;-----
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000100'           ;CÓDIGO BINARIO ASIGNADO A LA TECLA 4
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO D_S2_T6
BSF PASO,1
RETURN
;-----
D_S2_T6
;-----
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000110'           ;CÓDIGO BINARIO ASIGNADO A LA TECLA 6
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S2_DESACT
BSF PASO,1
RETURN
;-----
S2_DESACT
;-----
BCF PORTB,6
BCF RESP_SAL2,0
CALL S_BEEP
RETURN
;-----
*****
DESACT_SALIDA3
*****
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000111'           ;CÓDIGO BINARIO ASIGNADO A LA TECLA 7
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO D_S3_T1
BSF PASO,1
RETURN
;-----
D_S3_T1
;-----
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000001'           ;CÓDIGO BINARIO ASIGNADO A LA TECLA 1
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO D_S3_T4
BSF PASO,1
RETURN
;-----
D_S3_T4
;-----
CALL ESPERA
CALL GUARDAR_DATO
MOVLW B'00000100'           ;CÓDIGO BINARIO ASIGNADO A LA TECLA 4
SUBWF MEMORIA,W
BTFSC STATUS,2
GOTO S3_DESACT
BSF PASO,1
RETURN
;-----
S3_DESACT
;-----
BCF PORTB,7

```

```

        BCF          RESP_SAL3,0
        CALL   S_BEEP
        RETURN
;-----
S_BEEP
;-----
        BCF          PORTB,3
        DECF   BEEP,F
        BTFSC STATUS,2
        GOTO   RVB
        BSF          PORTB,3
        CALL   DEM_500us
        BCF          PORTB,3
        CALL   DEM_500us
        GOTO   S_BEEP
;-----
RVB
;-----
        MOVLW 0C8H
        MOVWF BEEP
        RETURN
;-----
S_BEEP_E
;-----
        BCF          PORTB,3
        DECF   BEEP,F
        BTFSC STATUS,2
        GOTO   S_B
        BSF          PORTB,3
        CALL   DEM_500us
        BCF          PORTB,3
        CALL   DEM_500us
        GOTO   S_BEEP_E
;-----
S_B
;-----
        CALL   DEM_300ms
        MOVLW 0C8H
        MOVWF BEEP
;-----
S_BEEP_E1
;-----
        BCF          PORTB,3
        DECF   BEEP,F
        BTFSC STATUS,2
        GOTO   RVB_E
        BSF          PORTB,3
        CALL   DEM_500us
        BCF          PORTB,3
        CALL   DEM_500us
        GOTO   S_BEEP_E1
;-----
RVB_E
;-----
        MOVLW 0C8H
        MOVWF BEEP
        RETURN
;-----
ENVIAR          ;SUBROUTINA DE ENVIO DE DATOS
;-----
        MOVWFTXREG
        CALL   BAN_TX
        RETURN
;-----
BAN_TX          ;PIR1=1?
;-----
        BCF          STATUS,RP0

```

```

B1    BTFSS PIR1,TXIF
      GOTO B1
      RETURN
.....
RET_CARRO ;ENTER
.....
      MOVLW    d'13'
      CALL    ENVÍAR
      CALL    DEM_100us
      MOVLW    d'10'
      CALL    ENVÍAR
      RETURN
.....
;*****
;DEMORAS:
;*****
DEM_100us                ;(100.00us)
.....
      MOVLW    d'12'
      MOVWF    N
      MOVLW    d'1'
      MOVWF    M
      CALL    DEMORA
      RETURN
.....
DEM_500us                ;(496.80us)
.....
      MOVLW    d'28'
      MOVWF    N
      MOVLW    d'1'
      MOVWF    M
      CALL    DEMORA
      RETURN
.....
DEM_100ms                ;(101.29ms)
.....
      MOVLW    d'85'
      MOVWF    N
      MOVLW    d'23'
      MOVWF    M
      CALL    DEMORA
      RETURN
.....
DEM_200ms                ;(200.94ms)
.....
      MOVLW    d'120'
      MOVWF    N
      MOVLW    d'23'
      MOVWF    M
      CALL    DEMORA
      RETURN
.....
DEM_300ms                ;(305.02ms)
.....
      MOVLW    d'148'
      MOVWF    N
      MOVLW    d'23'
      MOVWF    M
      CALL    DEMORA
      RETURN
.....
DEM_2s                   ;(2.35s)
.....
      MOVLW    d'255'
      MOVWF    N
      MOVLW    d'60'
      MOVWF    M
      CALL    DEMORA

```

```

                RETURN
                .....
                DEMORA                ;RETURN                ;Contador de la demora
                .....
                MOVF N,w
                MOVWF CONT1
                MOVWF CONT2
                MOVF M,w
                MOVWF CONT3
LAZO
                DECFSZ CONT1,1
                GOTO LAZO
                MOVF N,w
                MOVWF CONT1
                DECFSZ CONT2,1
                GOTO LAZO
                MOVF N,w
                MOVWF CONT2
                DECFSZ CONT3,1
                GOTO LAZO RETURN END

```

### 3.15 Entrada y Salida de datos entre PIC y puerto serial.

La comunicación serial se la transmite bit por bit y tiene la ventaja de necesitar menor cantidad de hilos y además se puede extender la comunicación a mayor distancia por ejemplo en la norma RS232 a 15 mts y en la norma RS422/485 a 1200mts.

Existen dos formas de realizar la comunicación serial: síncrona y la asíncrona la diferencia entre ellas es que en la síncrona además de la línea de transmisión de datos necesita otra línea que contenga los pulsos de reloj. El que se utiliza en este trabajo es asíncrona ya que no necesita pulsos de reloj en su lugar utiliza mecanismo de referencia tierra (RS232).

La norma RS232 se incluye actualmente en los computadores, conocido como puerto serial y sirve para comunicarse con otras computadoras además del mouse, programadores, impresoras etc. Esta señal se caracteriza por permanecer en un nivel lógico alto mientras no realiza ninguna transferencia de datos. Para empezar a transmitir datos el transmisor coloca la línea en nivel bajo durante el tiempo de un bit (416 us para 2400 bits/seg) éste se llama el bit de arranque, a continuación empieza a transmitir con el mismo intervalo de tiempo los bits de datos, que pueden ser de 7 u 8 bits, comenzando por los bits menos significativos y

terminando por los más significativos. Al final de la transmisión de datos se envía el bit de paridad, si estuviera activa está opción y por último los bits de parada, que pueden ser 1 o 2 después de esto la línea vuelve a un estado lógico alto y el transmisor está listo para enviar el siguiente dato.

Como el receptor no está sincronizado con el transmisor desconoce el momento en que se empieza la transmisión por lo que siempre debe estar en espera del cambio de estado o sea el bit de arranque, una vez que se da este bit, medio bit después vuelve a verificar si está en bajo, si no lo está no lo recibe ya que no pudo ser ocasionado por un ruido en la línea, caso contrario si el estado sigue siendo bajo, empieza a recibir la transmisión hasta el bit de parada. Para que la lectura de los datos sea correcta ambos equipos deben estar configurados a la misma velocidad y además parámetros y no exceder más allá de los 2 metros, pasando esta distancia los datos recibidos pueden no ser correctos debido a la pérdida de voltaje en el cable.

### **3.16 Interfaz de Entrada y Salida de datos entre el PIC y puerto serial**

Una vez explicada la teoría de la comunicación serial y su protocolo RS232 se pasa a describir la comunicación que lleva nuestro trabajo que lo realizamos con los pines de salida del PIC 16F28 y un CI MAX 232

### **3.17 Conexión de un Microcontrolador al puerto serie del PC.**

Para conectar el PC a un microcontrolador por el puerto serie se utilizan las señales **Tx**, **Rx** y **GND**. El PC utiliza la norma RS232, por lo que los niveles de tensión de los pines están comprendidos entre +15 y -15 voltios. Los microcontroladores normalmente trabajan con niveles TTL (0-5v). Es necesario por tanto intercalar un circuito que adapte los niveles:

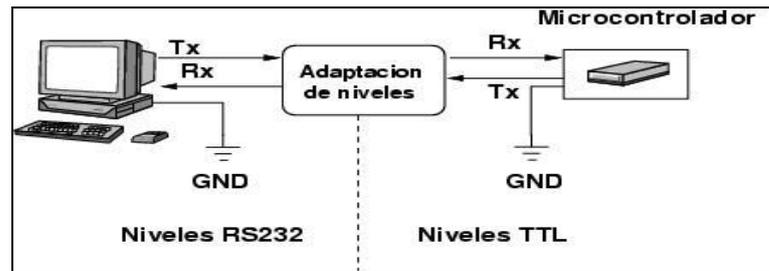


FIG 21 : CONEXIÓN DE PC CON EL PIC

El CI MAX 232 es la solución para transmitir a mayor distancia ya que incrementa los niveles de voltaje de 5 a +-10v gracias a un juego de capacitores que le ayuda a doblar los voltajes, por lo que su alimentación solo requiere una fuente de 5V. que puede ser la misma que utiliza el PIC además dispone de de 2 juegos de transmisores y receptores de los cuales solo ocuparemos un par de ellos.

El MAX 232 en este caso nos ayudará a convertir los voltajes TTL del PIC en voltajes de la norma RS232 quiere decir que si enviamos un estado lógico alto (5V) a la salida del T out del MAX 232 tendremos -10 V y si enviamos un 0 lógico desde el PIC (0v)el MAX 232 enviará +10V por lo tanto debemos invertir el dato de la salida del PIC y eso lo realizamos con la programación del PIC. La comunicación de datos entre el PIC y el puerto serial de la computadora se lo realiza de los pines 7 (RB1/TX/DT) y pin 8 (RB2/RX/CK) con los pines 11 (RX in) y 12 (RX out) del CI MAX 232, para luego utilizar las salidas 13 y 14 de este a un conector DB9 (pines 2 RX y 3 TX), utilizando la subrutina siguiente:

```

CONFIG_PTO_SERIE                ;PARÁMETROS DEL PUERTO SERIE
;*****
MOV LW    b'00100010'           ;TRANSMISION DE 8 BITS, TX
HABILITADA,MODO ASINCRONO,BAJA VALOCIDAD
MOVWF    TXSTA
MOV LW    d'32'                 ;VALOR 32,VELOCIDAD DE 9600 BPS
PARA 20 Mhz
MOVWF    SPBRG
MOV LW    B'00100000'           ;IT POR RX DE USART
MOVWF    PIE1
BCF      STATUS,RP0            ;BANCO 0
MOV LW    b'10010000'
MOVWF    RCSTA                 ;PUERTO SERIAL
HABILITADO,8 BITS, RECEPCION CONTINUA HABILITADA

```

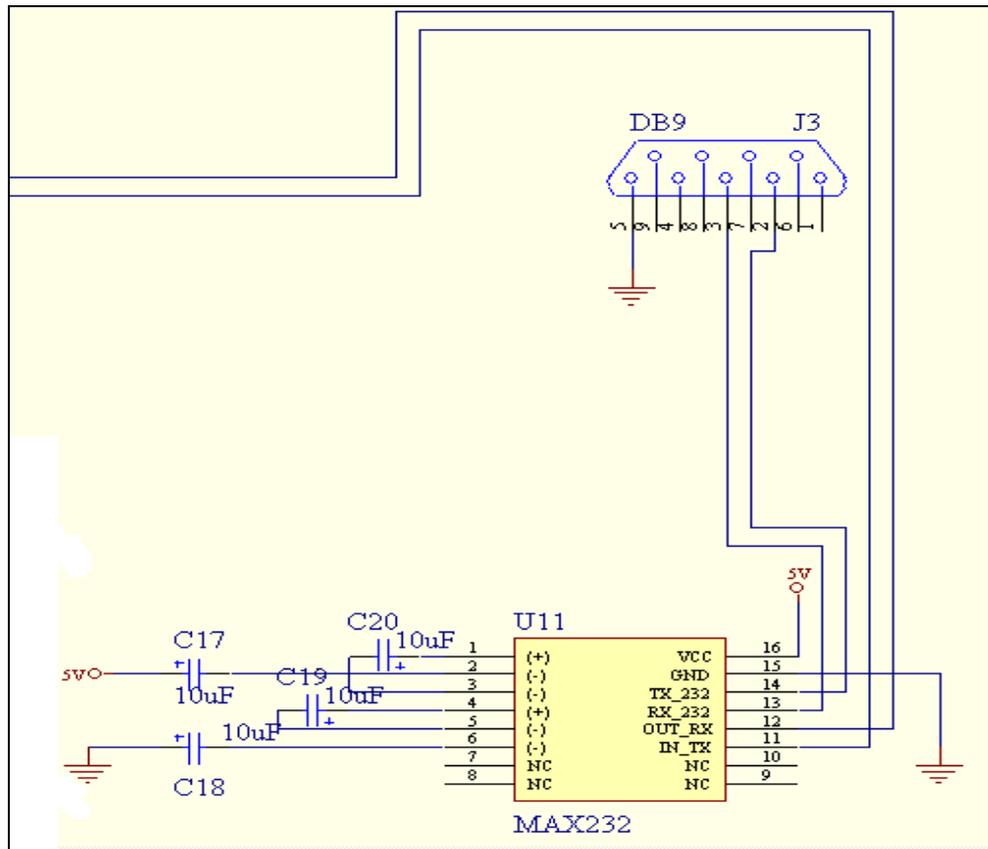


Fig 22 : Circuito de Conexión PIC / Puerto Serial

### 3.18 Circuito de Salida de Potencia

El circuito de potencia es de tipo DC/CA ya que utiliza una entrada DC para controlar 3 cargas de salida AC, utilizando elementos de estado sólido que tienen características que los hace muy rápidos silenciosos y compactos, teniendo una vida útil más elevada ya que son inmunes a los choques eléctricos y vibraciones.

Para este circuito de Mando Remoto se han configurado los pines 7 (RB0), 12 (RB6), y 13 (RB7) del PIC 16F628A como salida, está señal pasa a un led que actúa de Visualizador y a su vez ingresa a los pines 1 de cada uno de los Optoacopladores MOC3021 aislando la parte de mando con la de potencia, está señal de disparo pasa al gate de los Triacs BT136, para que finalmente de conducción a la carga.

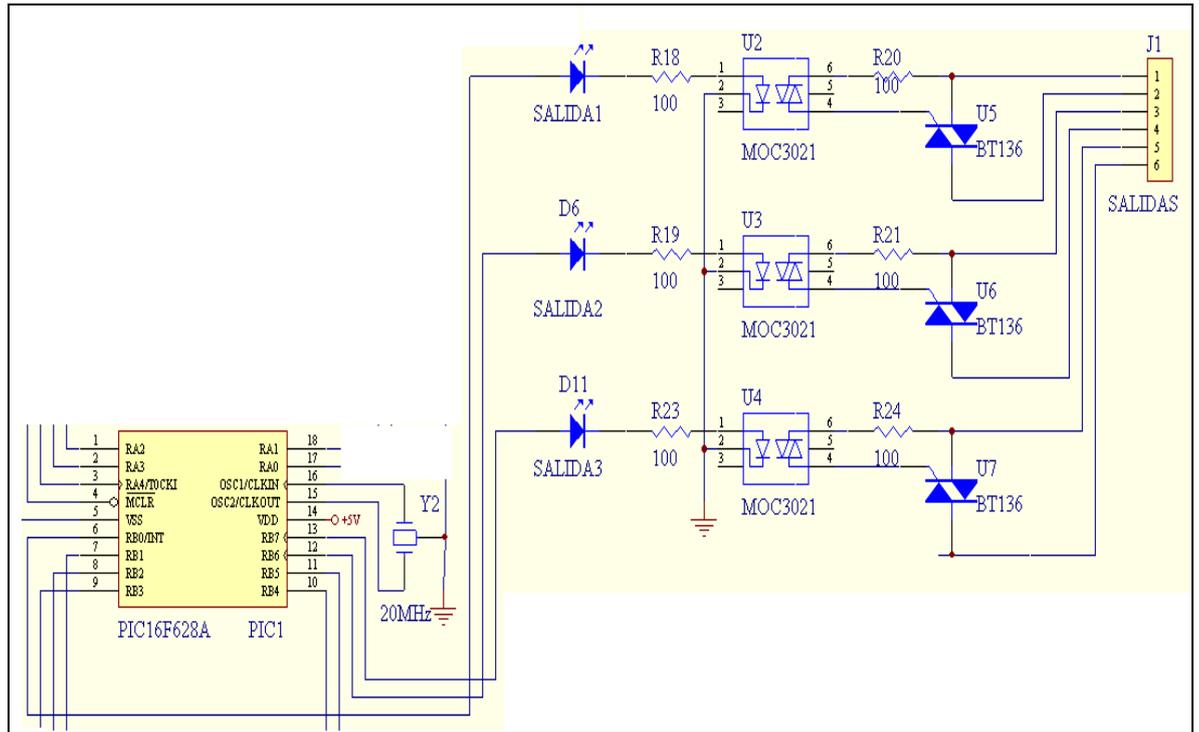


Fig 23 : Circuito de Salida de Potencia

## CAPITULO 4

### PROGRAMACIÓN EN VISUAL C++

A diferencia de los Mandos Remotos que se ha podido consultar y que han sido realizados, en este trabajo de grado se incluyó un interfaz de comunicación y monitoreo de tipo gráfico, que nos permite revisar el estado de las salidas de nuestro Mando Remoto y a su vez también nos permite modificar el estado de salida en forma independiente de la línea telefónica, Esto nos posibilita tener una segunda alternativa de comando a través de un enlace por RED local o internet siempre que el equipo disponga de una dirección IP fija para el internet, y una dirección de red interna de red local. Este diseño se lo efectuó en Visual C++ 2008.

Como sabemos, Windows es el entorno más popular de interfaz gráfico de usuario (GUI). Desde este punto de vista, Windows es un entorno multitarea basado en ventanas, que representan programas, y que permite ejecución concurrente.

Para desarrollar programas, Windows provee una librería de rutinas y funciones (SDK - Kit de desarrollo de software) que permiten gestionar componentes como menús, diálogos, ventanas, etc.

Visual C++ es un entorno integrado de desarrollo que permite la programación orientada a objetos (POO) conjuntamente con el sistema de desarrollo SDK (también denominado API) de Windows. Al ser un entorno integrado Visual C++ incluye, entre otras, las siguientes herramientas de desarrollo:

- Editor de texto
- Compilador/Enlazador
- Depurador

- Visor de datos y dependencias (Browser)

Pero si desde el punto de vista del usuario Windows es un sistema amigable, desde el punto de vista del desarrollador observaremos todo lo contrario. El SDK de Windows no es más que un complejo conjunto de funciones que añade además numerosas definiciones de tipos de datos nuevos para cualquier programador de C/C++ para DOS. Para solucionar este problema, Visual C++ incluye la librería de clases MFC (Microsoft Foundation Classes) que permite crear y gestionar de manera intuitiva componentes típicos de Windows. Esto es, la MFC es una implementación que utiliza el API encapsulando todas las estructuras y llamadas a funciones en objetos fáciles de utilizar. Basándose en la potencia de la MFC, Visual C++ se convierte en un generador de programas C++ para Windows.

Como todos sabemos, "C" es un lenguaje de alto nivel, basado en funciones, que permite desarrollos estructurados. Entre otras muchas características contempla la definición de estructuras de datos, recursividad o direcciones a datos o código (punteros). "C ++", por su parte, es un superconjunto de "C", al que recubre con una capa de soporte a la POO. Permite por tanto la definición, creación y manipulación de objetos.

## **4.1 Programación orientada a objetos**

### **4.1.1 Conceptos preliminares**

La POO es una nueva filosofía de programación que se basa en la utilización de objetos. El objetivo de la POO no es sino la meta de cualquier modelo de programación estructurada convencional además de "imponer" una serie de normas de desarrollo que aseguren y faciliten la mantenibilidad y reusabilidad del código.

Los mecanismos básicos de la POO son: objetos, mensajes, métodos y clases.

- **Objetos.** Un objeto es una entidad que tiene unos atributos particulares (datos) y unas formas de operar sobre ellos (los métodos o funciones miembro). Es decir, un objeto incluye, por una parte una serie de operaciones que definen su comportamiento, y una serie de variables manipuladas por esas funciones que definen su estado. Por ejemplo, una ventana Windows contendrá operaciones como "maximizar" y variables como "ancho" y "alto" de la ventana.
- **Mensajes.** En C++, un mensaje se corresponde con el nombre de uno de los métodos de un objeto. Cuando se pasa un mensaje a un objeto, este responde ejecutando el código de la función asociada.
- **Método.** Un método (función miembro) se implementa dentro de un objeto y determina como tiene que actuar el objeto cuando se produce el mensaje asociado. En C++ un método se corresponde con la definición de la función miembro del objeto. La estructura más interna de un objeto está oculta, de tal manera que la única conexión con el exterior son los mensajes.
- **Clases.** Una clase es la definición de un tipo de objetos. De esta manera, una clase "Empleado" representaría todos los empleados de una empresa, mientras que un objeto de esa clase (también denominado instancia) representaría a uno de esos empleados en particular.

Las principales características de la POO son: abstracción, encapsulamiento, herencia y polimorfismo:

- **Abstracción.** Es el mecanismo de diseño en la POO. Nos permite extraer de un conjunto de entidades datos y comportamientos comunes para almacenarlos en clases.
- **Encapsulamiento.** Mediante esta técnica conseguiremos que cada clase sea una caja negra, de tal manera que los objetos de esa clase se puedan manipular como unidades básicas. Los detalles de la implementación se encuentran dentro de la clase, mientras que desde el exterior, un objeto será

simplemente una entidad que responde a una serie de mensajes públicos (también denominados interfaz de la clase).

- **Herencia.** Es el mecanismo que nos permite crear clases derivadas (especialización) a partir de clases bases (generalización). Es decir, podríamos tener la clase "Empleado" (clase base) y la clase "Vendedor" derivando de la anterior. Una librería de clases (como la MFC) no es más que un conjunto de definiciones de clases interconectadas por múltiples relaciones de herencia.
- **Polimorfismo.** Esta característica nos permite disponer de múltiples implementaciones de un mismo método de clase, dependiendo de la clase en la que se realice. Es decir, podemos acceder a una variedad de métodos distintos (con el mismo nombre) mediante el mismo mecanismo de acceso. En C++ el polimorfismo se consigue mediante la definición de clases derivadas, funciones virtuales y el uso de punteros a objetos.

Otros dos conceptos muy importantes en la POO son relativos a la creación y destrucción de objetos. En lenguajes estructurados convencionales, cuando se define una variable se le reserva espacio en memoria y, si no se inicializa expresamente, se hace por defecto (por ejemplo, en C una variable global siempre se inicializa a 0, pero una automática no, por lo que si no se inicializa expresamente su contenido inicial será basura); por otra parte, cuando se destruye una variable (por que se abandona el ámbito de su definición - scope -) se libera la memoria que estaba ocupando. Si ahora hacemos el paralelismo obligado entre variables y objetos para los lenguajes POO nos daremos cuenta de que deben existir procedimientos especiales de construcción y destrucción de objetos. En concreto, cada clase tiene dos funciones miembro especiales denominadas constructor y destructor.

- **Constructor** : Función miembro que es automáticamente invocada cada vez que se define un objeto, su objetivo es la inicialización del mismo. Toma el

mismo nombre que la clase, puede recibir parámetros y podemos tener varios constructores definidos.

- **Destructor** : Función miembro invocada automáticamente cada vez que se destruye un objeto. Su objetivo es realizar operaciones como liberación de memoria, cerrar ficheros abiertos, etc. Toma el mismo nombre de la clase comenzado primero por el carácter "~", no toma parámetros y no admite la sobrecarga (sólo puede existir uno en cada clase).

En muchos casos, para las clases más sencillas, podemos encontrar clases que no tienen constructor o destructor, ó ninguno de los dos. En C++, siempre existen constructores y destructores por defecto que realizan una inicialización/liberación estándar.

#### 4.2 El modelo de programación Windows

El modelo de programación propuesto por Windows es totalmente diferente al modelo de ejecución secuencial de DOS. Al ser Windows un entorno multitarea los programas tienen que estar preparados para compartir los recursos de la maquina (procesador, memoria, teclado, ratón etc). Ésto supone que Windows ha de disponer de métodos que permitan suspender tareas para activar otras en función de las circunstancias del momento (por ejemplo, por acción del usuario).

Pero por parte de las aplicaciones, este hecho supone que han de cooperar en la compartición de esos recursos. Las aplicaciones Windows se limitan a "esperar" mensajes procedentes del sistema, procesarlos y volver al estado de espera. Este modelo de programación se conoce como "orientado al evento".

- **Mensaje**. Es una notificación a la aplicación de que ha ocurrido algo de interés y que por lo tanto debe de realizarse alguna acción específica. El origen del mensaje puede ser el usuario (haciendo click con el ratón dentro de una ventana), la propia aplicación (mandándose un mensaje a si misma) o Windows (pidiendo,

por ejemplo, que se repinte la ventana tras ocultarse otra que tuviese delante). Dado que la unidad mínima de ejecución en Windows es una ventana, los mensajes van realmente dirigidos a ellas.

- **Ventana y procedimiento de ventana.** En Windows, una aplicación se representa físicamente por su ventana principal (aunque después pueda desplegar diversas ventanas hijas). Cada una de esas ventanas dispone de una serie de propiedades y un código asociado (lo que concuerda con el principio de la POO, en el concepto de objeto). Al código asociado a cada ventana se le denomina procedimiento de ventana. Es una función que recibe los mensajes, los procesa y devuelve el control a Windows para quedar en espera.

Otra de las características específicas de Windows frente a DOS es el uso de recursos por parte de las aplicaciones, como son iconos, menús, mapas de bits, cursores, plantillas de diálogos, etc. Las aplicaciones Windows disponen por tanto de recursos (gráficos generalmente) propios almacenados en lo que se llama el fichero de recursos). El proceso de construcción de programas en Windows incorpora una fase adicional al compilado y enlazado de los módulos objeto y las librerías. Hay un proceso final de compilación y de enlazado (bind) del fichero de recursos.

### 4.3 EL ENTORNO DE DESARROLLO

El entorno de desarrollo viene representado por el icono "Developer Studio". En él se integran entre otras las siguientes herramientas:

- Editor orientado a la codificación C/C++ (resaltando palabras claves ...)
- Compilador/Enlazador incremental, que acelera el proceso de construcción de los programas.
- Depurador Visual, que permite Visualizar y modificar el contenido de variables y áreas de memoria.

- Visor de datos (browser) que permite fácilmente controlar dependencias y referencias a funciones, datos, clases, etc. Además permite Visualizar la jerarquía de las clases utilizadas en los programás.
- Herramientas complementarias como un analizador de ventanas (Spy ++ ) o un trazador de funciones MFC.

En Visual C++ la construcción de cualquier tipo de programa se inscribe dentro del concepto de proyecto (workspace). Un proyecto define los pasos a seguir para alcanzar la construcción de un objetivo (un programa, una DLL, etc.), en realidad es un concepto análogo a lo que se conoce como "makefile" en otros entornos tíPICos de desarrollo en C. En realidad, Visual C++ genera para cada proyecto dos ficheros que lo definen, el fichero de workspace (con extensión wsp) y un makefile (con extensión mak) estándar que permitiría la utilización del mismo proyecto en otro entorno distinto.

Desde el punto de vista funcional, el proyecto contiene referencias a cada uno de los ficheros fuentes (C/C++, con extensiones c y cpp respectivamente), objetos, librerías o ficheros de recursos (extensión rc) que se deben utilizar para construir el objetivo final del proyecto.

En definitiva, para crear cualquier programa con Visual C++ debemos comenzar creando un proyecto para él, codificando y añadiendo los módulos necesarios a dicho proyecto, y definiendo los recursos asociados.

## **4.4 EL CONCEPTO DE PROYECTO (WORKSPACE)**

### **4.4.1 Posibilidades de programación**

Cuando se crea un nuevo proyecto (desde la opción "Nuevo" del menú "Fichero" aparece un diálogo que nos permite especificar que se cree un nuevo workspace), lo primero que solicita el sistema es determinar el tipo de objetivo que se persigue con este proyecto.

. Destacar las siguientes posibilidades:

- Aplicación (.EXE) basada en la MFC (MFC AppWizard)
- Librería de enlace dinámico (.DLL) basada en la MFC. (MFC AppWizard)
- Aplicación (.EXE) estándar para Windows (basada en el SDK)
- Librería de enlace dinámico (.DLL) (basada en el SDK)
- Aplicación (.EXE) modelo DOS (Console application)
- Librería estática (.LIB)

Como ya hemos destacado anteriormente, el objetivo del presente curso es el manejo de la librería MFC, por lo que nos centraremos básicamente en el primer punto (la construcción de una DLL con MFC es totalmente similar).

#### 4.5 EL GENERADOR DE APLICACIONES.

Ya sabemos que Visual C++, apoyado en la potencia de la MFC, es capaz de convertirse en un generador de aplicaciones. Para ello dispone de dos herramientas integradas complementarias:

- **AppWizard**, que es el generador de aplicaciones propiamente dicho. Con él podremos generar esqueletos de programas para Windows basados en la MFC.
- **ClassWizard**, herramienta de mantenimiento de los programas generados con la anterior. Permite añadir o eliminar clases, modificar los comportamientos de las mismas, etc.

Pero el código generado mediante este método presenta una complejidad añadida a la natural de cualquier programa; junto con el código C/C++ y el de la MFC aparecen líneas (generalmente entre comentarios) que son totalmente necesarias para el funcionamiento de las dos herramientas anteriores, modificar cualquiera de esas líneas de código dará muchos problemas a la hora de utilizar ClassWizard

para modificarlo. De todas maneras, este "defecto" es bastante popular entre los usuarios de cualquier generador de código, para cualquier lenguaje.

El formato general de los proyectos generados con estas herramientas suele tener las siguientes características:

- Cada clase de nuestro programa dispondrá de dos ficheros: Un fichero de cabecera (extensiones .h o .hpp) y un fichero de implementación (.cpp). El fichero de cabecera contiene la definición de la clase (definiciones de sus miembros datos y funciones - los mensajes -), mientras que el fichero fuente contiene la implementación de esas funciones miembro (los métodos de la clase).
- Un fichero de recursos (extensión .rc), aunque éste es opcional.
- Módulos objetos (.obj) y librerías estáticas (.lib) necesarias para crear nuestro programa.

#### **4.6 DISEÑO DE SOFTWARE DE APLICACIÓN DE MANDO DE UNA CASA VÍA TELEFÓNICA A TRAVEZ DE INTERFAZ SERIAL USB.**

##### **4.6.1 Casos de uso**

Un caso de uso es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema. Para especificar la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas. O lo que es igual, un diagrama que muestra la relación entre los actores y los casos de uso en un sistema. Una relación es una conexión entre los elementos del modelo, por ejemplo la especialización y la generalización son relaciones. Los diagramas de casos de uso se utilizan para ilustrar los requerimientos del sistema al mostrar cómo reacciona a eventos que se producen en su ámbito o en él mismo. El diagrama de casos de uso sirven que hemos usado, ha sido realizado para comprender la interacción entre la Persona o Usuario y el Dispositivo electrónico controlado por el PIC 16F28A

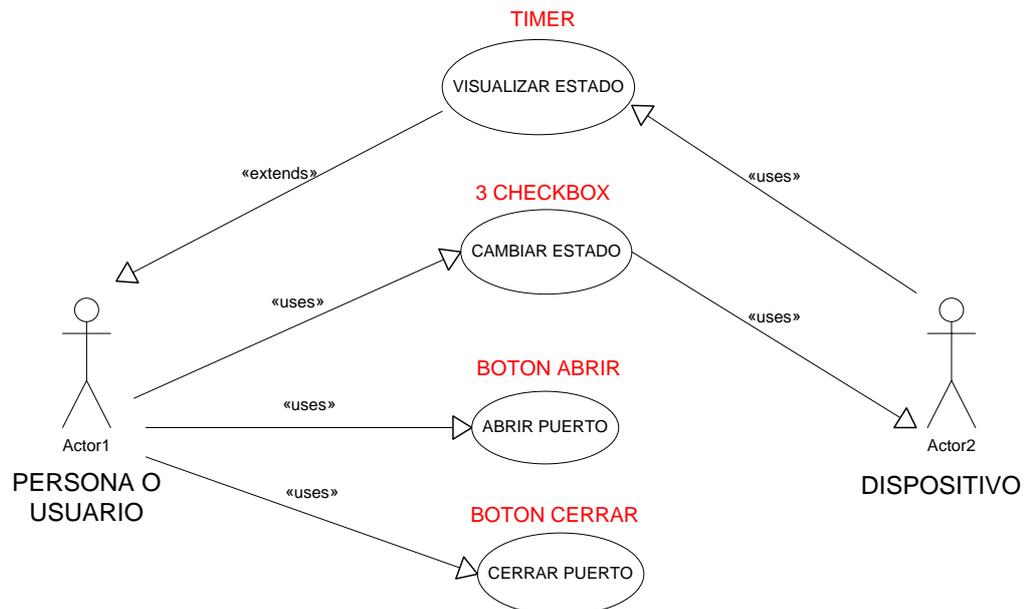


FIG 24 : DIAGRAMA DE CASOS DE USOS

#### 4.7 CÓDIGO FUENTE DEL PROGRAMA EN VISUAL C

```
private: System::Void button1_Click(System::Object^ sender, System::EventArgs^ e)
```

```

    {panel2->Enabled = false;

    timer1->Enabled = false;

    serialPort1->Close();

    serialPort1->PortName = ("COM" + numericUpDown1->Text);

    serialPort1->Open();

    panel2->Enabled = true;

    timer1->Enabled = true; }

```

#### PROGRAMACIÓN DEL BOTON ABRIR

```
private: System::Void timer1_Tick(System::Object^ sender, System::EventArgs^ e)
```

```

    {serialPort1->WriteLine("EST");

    String^ sw1;

    String^ sw2;

```

```

        String^ sw3;
    If (temp->Contains("EST:")){
        sw1 = temp->Split(';')[0];
        sw2 = temp->Split(';')[1];
        sw3 = temp->Split(';')[2];
    if (sw1->Substring(8)==="ON"){
        checkBox1->Checked = true; }
    else
        {checkBox1->Checked = false; }
    If (sw2->Substring(3)==="ON"){
        checkBox2->Checked = true; }
    else
        {checkBox2->Checked = false;}
    If (sw3->Substring(3)==="ON")
        {checkBox3->Checked = true;}
    else
        {checkBox3->Checked = false;}
        textBox1->Text = temp; } }

```

### REVISA EL ESTADO DE LOS SWICHS(TIMER)

```

private: System::Void serialPort1_DataReceived(System::Object^ sender,
System::IO::Ports::SerialDataReceivedEventArgs^ e)
    { temp = serialPort1->ReadLine();}

```

### LEE LA INFORMACIÓN DEL PUERTO ENTREGADA CADA VEZ QUE ENVÍA EL DISPOSITIVO

```

private: System::Void button2_Click(System::Object^ sender, System::EventArgs^
e)
    {serialPort1->Close();
        timer1->Enabled = false;

```

```
panel2->Enabled = false; }
```

### CIERRA EL PUERTO

```
private: System::Void checkBox2_Click_1(System::Object^ sender,
System::EventArgs^ e)
```

```
{ timer1->Enabled = false;
```

DESABILITO EL TIMER POR NO PODER HACER LAS FUNCIONES DE CHEQUEAR Y ESCRIBIR EN EL PIC AL MISMO TIEMPO

```
If (checkBox2->Checked)
```

```
{serialPort1->WriteLine("S2_1"); } SI ESTÁ CHEQUEADO MANDO A PRENDER
```

```
else
```

```
{serialPort1->WriteLine("S2_0"); } SI NO LO ESTÁ MANDO APAGAR
```

```
}
```

```
timer1->Enabled = true; }
```

HABILITO EL TIMER PRA QUE SIGA LEYENDO EL ESTADO DEL PIC

```
private: System::Void checkBox1_Click_1(System::Object^ sender,
System::EventArgs^ e)
```

```
{timer1->Enabled = false;
```

```
If (checkBox1->Checked)
```

```
{serialPort1->WriteLine("S1_1"); } SI ESTÁ CHEQUEADO MANDO A PRENDER
```

```
else
```

```
{serialPort1->WriteLine("S1_0"); } SI NO LO ESTÁ MANDO APAGAR
```

```
timer1->Enabled = true;}
```

```
private: System::Void checkBox3_Click_1(System::Object^ sender,
System::EventArgs^ e)
```

```
{timer1->Enabled = false;
```

```
If (checkBox3->Checked)
```

```
{serialPort1->WriteLine("S3_1"); } SI NO LO ESTÁ MANDO A PRENDER
```

```
else
```

```
{serialPort1->WriteLine("S3_0");} SI NO LO ESTÁ MANDO APAGAR
```

```
timer1->Enabled = true; };}}}
```

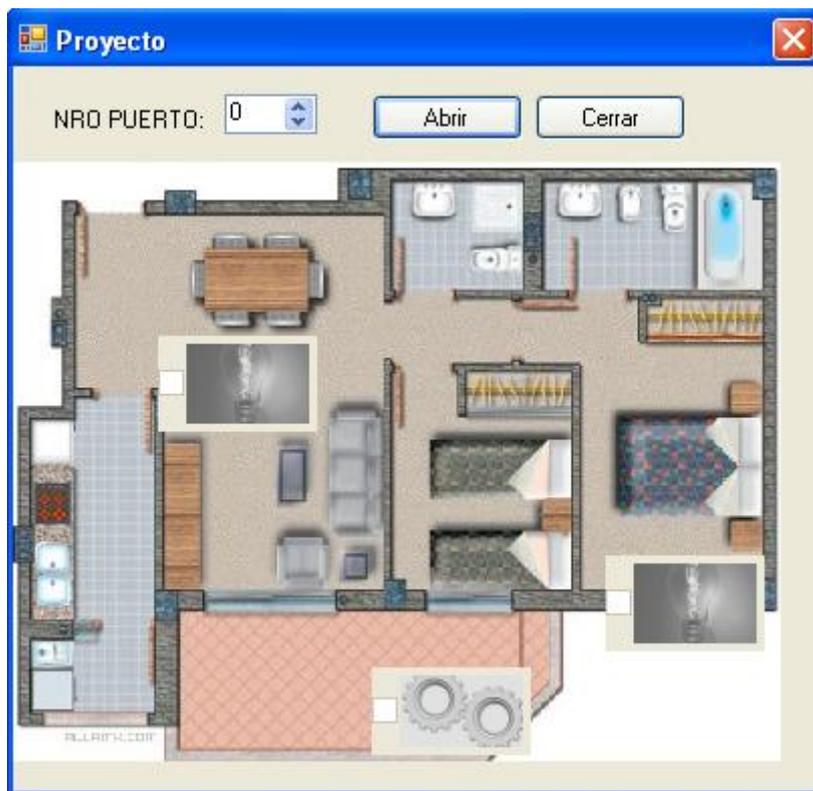


FIG 25 : Esquema final de programación para comando y visualización

## CAPITULO 5

### DISEÑO Y CONSTRUCCIÓN DEL PROYECTO

Para elaborar el circuito electrónico del Mando Remoto de una Casa Vía Telefónica, se utilizó Protel 99, tomando en cuenta todos los aspectos relacionados con el diseño e implementación basados en los siguientes elementos:

- a) Una fuente de poder formada con un puente de Diodos BR 805D, y reguladores integrados 7805 y 7812,
- b) Una etapa de detección de timbrado compuesta por un CI TL081 y un optoacoplador 4N35,
- c) Una sección de simulación de levantado de Línea que lo integra un relé K1, y una resistencia de 560 ohm
- d) El circuito Integrado MT8870 (decodificador de Tonos DTMF)
- e) El PIC 16F628A como circuito principal de control,
- f) El interfaz de comunicaciones con el Max 232,
- g) Una etapa de potencia de 3 salidas formados por los opto acopladores MOC3021 y los Triacs BT136.



### 5.1 Listado de componentes

Se utilizaron los siguientes elementos que fueron de fácil consecución:

ELEMENTO:	DESCRIPCION:
U1	MT8870
U2, U3, U4	MOC3021
U5,U6,U7	BT136
U8	TL081CN
U9	MC7812
U10	MC7805
U11	MAX3232
OPT1	4N35
PIC1	PIC16F628A
D1	1N4728A
D2	BR-805D
D3,D4,D7	1N4744A
D5,D6,D11	Leds Salidas 1, 2 y 3
D8	Led Llamada Entrante
D9,D10	1N4007
D12	1N4734
D13,D14	Leds Fuente de 12v, 5v
Q1, Q2, Q4, Q5	2N3409
Q3	2N5401
Y1	Cristal 3.58Mhz
Y2	Resonador de 20Mhz
K1	Relé 5v
RJ1	Plug Entrada Línea Telefónica
J1	Salidas 110 V
J2	Entrada 12V DC
J3	DB9 Puerto Serial
R1, R2, R3	4,7K $\Omega$
R4, R5	330 $\Omega$
R6	560 $\Omega$ 2W
R7, R8, R9, R10, R11	100K $\Omega$
R12	51K $\Omega$
R13	68K $\Omega$
R14	220K $\Omega$
R15, R16	10K
R17	390K $\Omega$
R18, R19, R20, R21, R22, R23, R24	100 $\Omega$
R25, R26	Potenciómetro 100k $\Omega$
R27, R28, R29	1K $\Omega$
R30, R31	33K $\Omega$
R32, R33	8,2K $\Omega$
R34	1M $\Omega$
R35	47 $\Omega$
R36, R37	2.2K $\Omega$
R38	150 $\Omega$
R39	82K $\Omega$

C1, C2	10nF 400V
C3, C4	100nF
C5	0,047uF
C6	0,01uF
C7, C15, C16, C17, C18, C19, C20	10uF
C8	330uF
C9, C10, C11, C12	47nF
C13, C14	100uF

TABLA 6: Listado de componentes

## 5.2 TARJETA ELECTRÓNICA

Se la realizó en Prótel 99 a continuación se indica las caras superior e inferior así como la asignación de elementos en la parte superior.

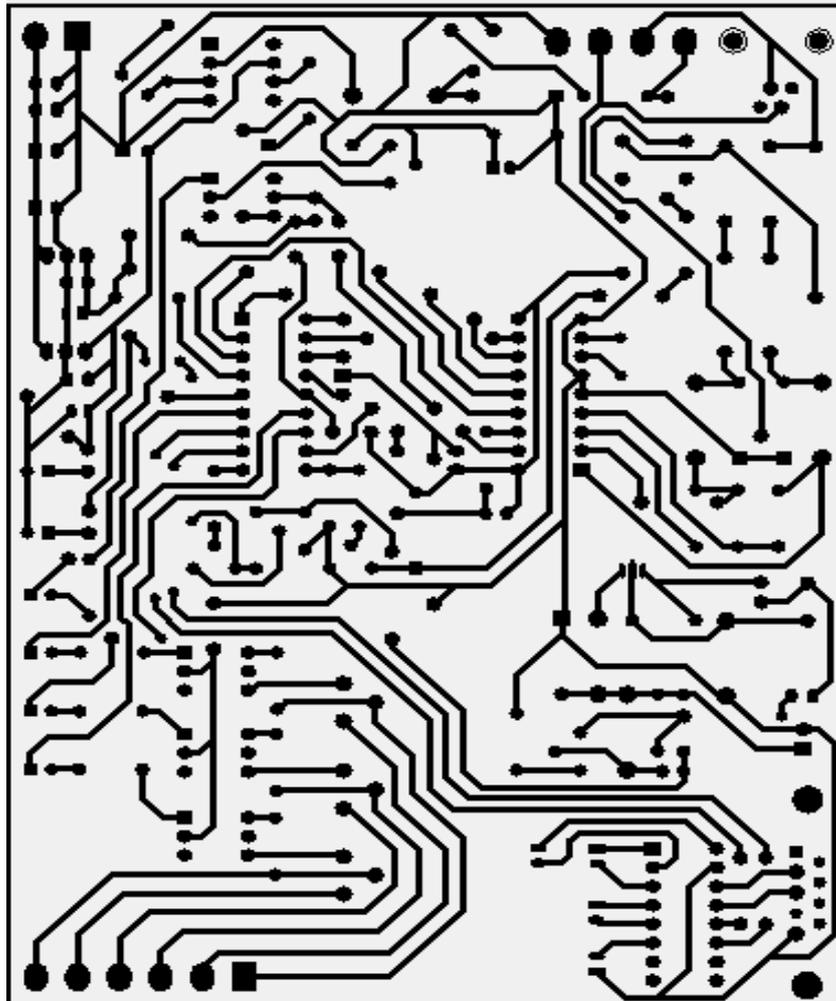


FIG 27: CARA INFERIOR DE TARJETA ELECTRÓNICA

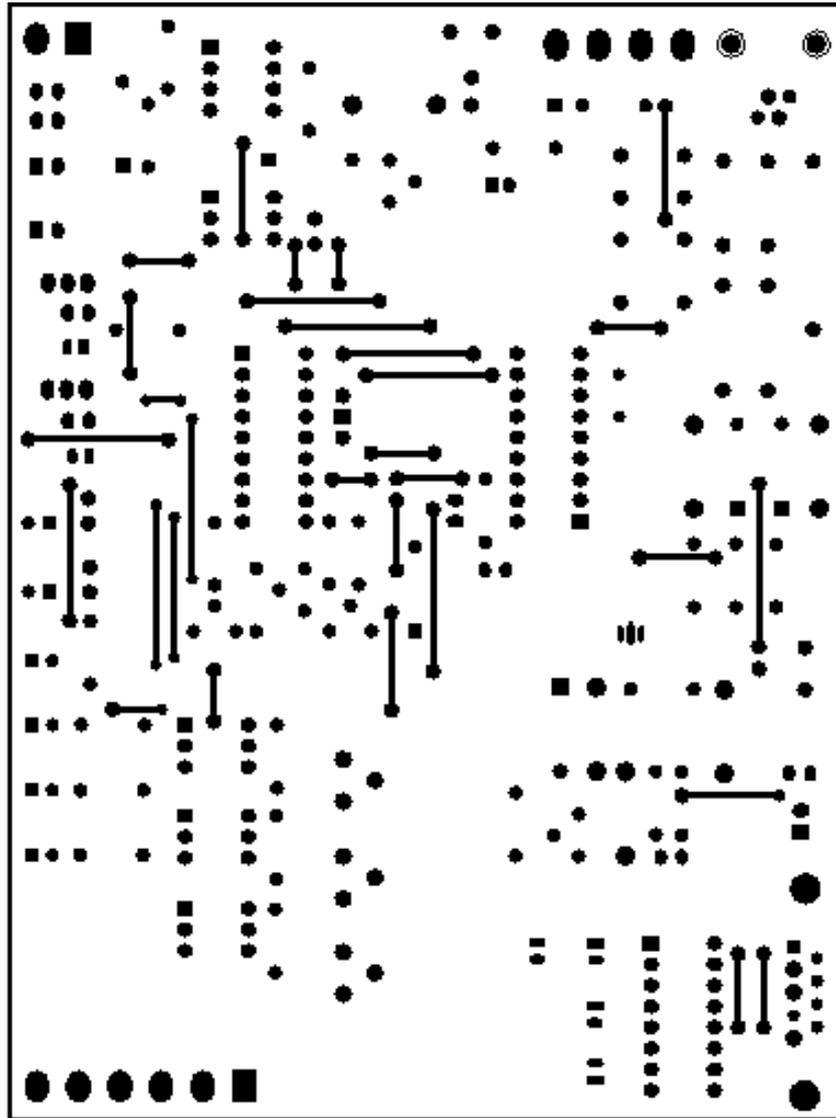


FIG 28 : CARA SUPERIOR Y JUMPERS DE CONECCION



### 5.3 PROTOCOLO DE COMUNICACIÓN

Para realizar el enlace de comunicación entre el PIC y el Puerto serial de la PC debemos tener el envío desde el PIC de el siguiente protocolo

PROTOCOLO CONTROL TELEFÓNICO

MENSAJE DE INICIO:

SISTEMA EN LINEA  
CONTROL DOMÓTICO  
ANDRES ARGUDO .  
V1.1

PROTOCOLO:

S1\_1 SALIDA 1 ACTIVADA  
S1\_0 SALIDA 1 DESACTIVADA

S2\_1 SALIDA 2 ACTIVADA  
S2\_0 SALIDA 2 DESACTIVADA

S3\_1 SALIDA 3 ACTIVADA  
S3\_0 SALIDA 3 DESACTIVADA

ESTÁDO:

EST: S1\_ON/OFF;S2\_ON/OFF;S3\_ON/OFF;

### 5.4 CLAVES DE ACTIVACIÓN Y DESACTIVACIÓN.

CLAVE DE INGRESO AL SISTEMA :

\*1028# PARA ACTIVAR SALIDAS  
\*1029# PARA DESACTIVAR SALIDAS

CLAVES INDIVIDUALES PARA ACTIVAR:

SALIDA 1: 1128  
SALIDA 2: 9034  
SALIDA 3: 4271

CLAVES INDIVIDUALES PARA ACTIVAR LAS SALDAS POR 2 SEGUNDOS:

SALIDA 1: 1129      5 SEGUNDOS  
SALIDA 2: 9035      10 SEGUNDOS  
SALIDA 3: 4272      20 SEGUNDOS

CLAVES INDIVIDUALES PARA DESACTIVAR:

SALIDA 1: 3285  
SALIDA 2: 6346  
SALIDA 3: 2714

### 5.5 CONEXIÓN DE APARATOS A 110 V.

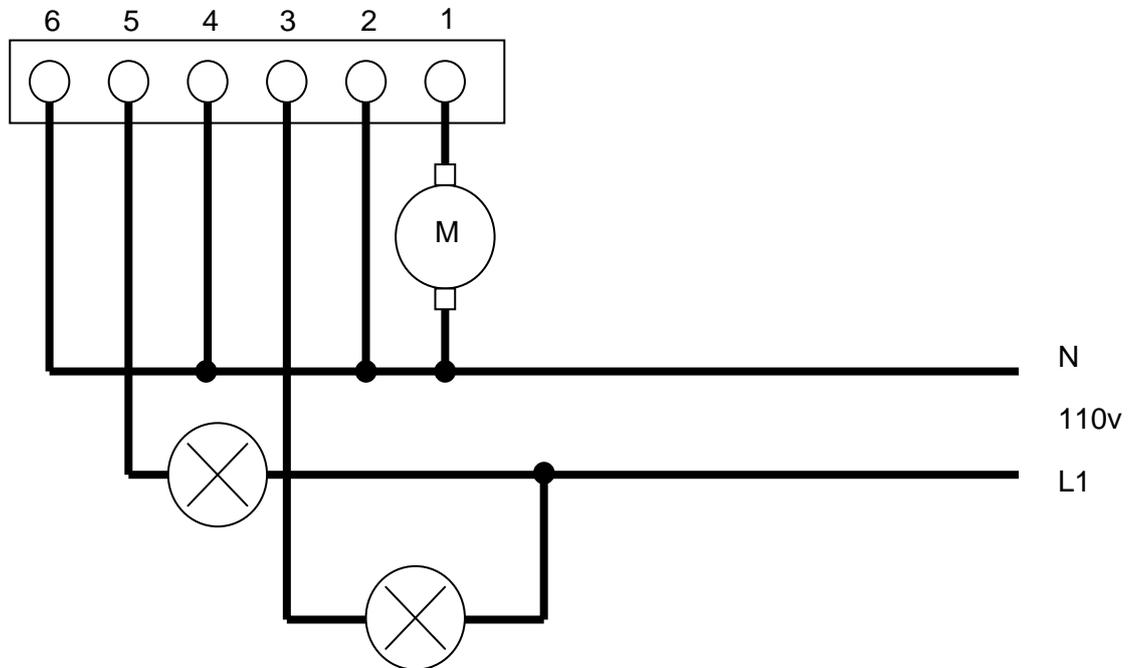


FIG 30: ESQUEMA DE CONECCION A 110 VCA

### 5.6 CONEXIÓN Y MANDO A TRAVEZ DE ESCRITORIO REMOTO.

La conexión de acceso remoto se lo puede efectuar de varias maneras, la más normal es usar el que tiene Windows, pero en el caso de este trabajo se utilizó un software que por sus características de funcionalidad y seguridad estaba mejor dotado que la herramienta de acceso remoto que tiene Windows, este tiene una misma estructura de funcionamiento Cliente – Servidor.

### 5.7 VNC

Son las siglas en inglés de *Virtual Network Computing* (Computación Virtual en Red ). **VNC** es un programa de software libre basado en una estructura cliente-servidor el cual nos permite tomar el control del ordenador servidor remotamente a través de un ordenador cliente. También llamado software de escritorio remoto.

VNC no impone restricciones en el sistema operativo del ordenador servidor con respecto al del cliente; es posible compartir la pantalla de una máquina con cualquier sistema operativo que soporte VNC conectándose desde otro ordenador o dispositivo que disponga de un cliente VNC portado.

La versión original del **VNC** se desarrolló en Reino Unido, concretamente en los laboratorios AT&T, en Cambridge. El programa era de código abierto por lo que cualquiera podía modificarlo y existen hoy en día varios programás para el mismo uso.

Es un software que permite acceder de manera remota a un computador conectado a una red, ya sea local (LAN) o bien Internet, permitiendo tomar su control y realizar actividades como si se estuviera frente a él. Este acceso puede realizarse desde múltiples plataformas (Windows, Linux, etc.) a través de un software cliente o de un navegador.

### **5.7.1 Características principales**

- Instalación del software servidor a través de asistente, además pudiendo habilitarse como servicio en Windows XP/2000.
- El software cliente está disponible para diferentes sistemas operativos: Windows, Linux, Unix, Java, Macintosh.
- Según la configuración del computador a administrar (Servidor) permite tomar el control de éste y realizar todas las actividades como si se estuviera frente a él, o bien realizar sólo tareas de Visualización de lo que está pasando en el servidor.
- Permite definición de contraseña de conexión y modo de sólo Visualización.

### 5.7.2 Funcionamiento

Se instaló VNC en cada equipo y se corre la aplicación en el computador que se desea administrar remotamente (Server), el cual puede ser de muchos sistemas operativos (ver imagen); luego, desde cualquier parte del mundo, a través de Internet, se utiliza un software (Viewer) en el cliente, para ver y administrar el equipo definido.

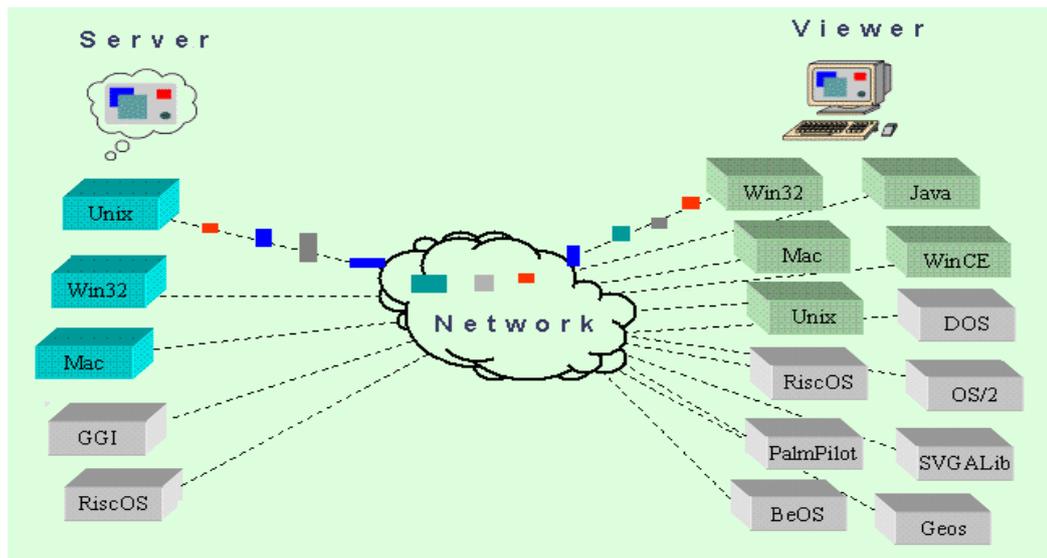


FIG 31: ENLACE SERVIDOR / CLIENTE

Para la demostración y presentación se instaló VNC versión 4 el mismo que se lo pudo bajar desde [www.realvnc.com](http://www.realvnc.com) en forma gratuita el paquete servidor y el paquete cliente, en el caso de el equipo al cual estará conectada la tarjeta de control se la deberá configurar como servidor. Los parámetros que se tienen en estas opciones son las relacionadas con la autenticación y asignación de clave, la conexión, el escritorio, el método de captura y el compartimiento del equipo el momento de acceso, de ellos lo más importantes son la clave de acceso, y el si se bloquea o no el equipo servidor cuando el equipo remoto toma el control.

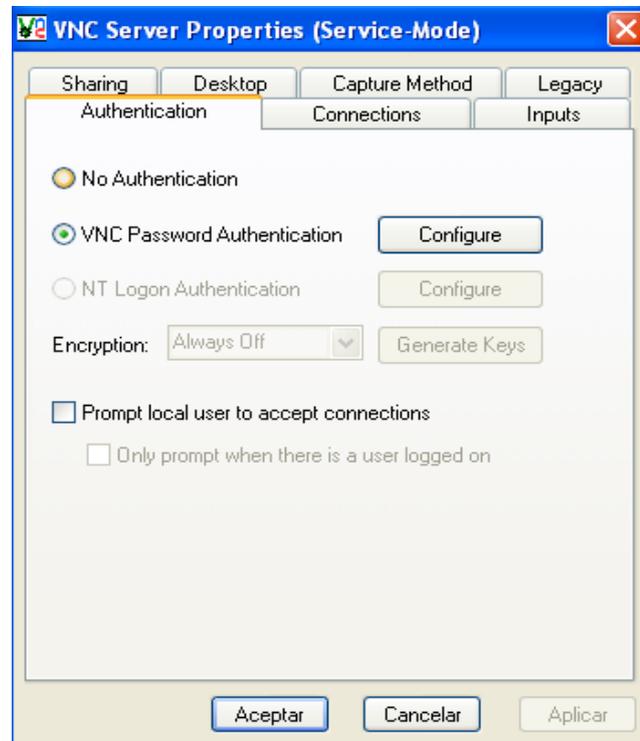


Fig: 32 Configuración de VNC Server

Para acceder al equipo servidor se corre la opción VNC Viewer que se instala el momento de correr la instalación del paquete VNC y tan solo digitando la dirección IP asignada al equipo remoto se tomará el control del equipo para el comando de Mando Remoto de una Casa. Ejemplo

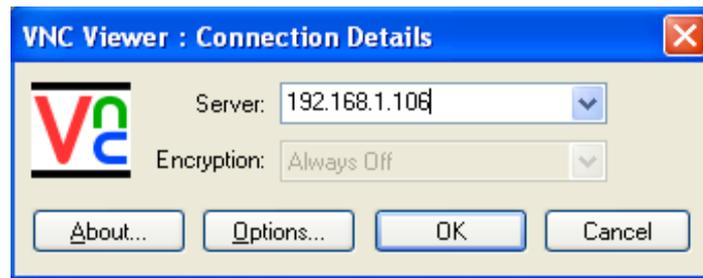


Fig 33: Configuración de IP de equipo Remoto

## 5.8 Pruebas realizadas

Se realizaron pruebas destinadas a determinar el correcto funcionamiento y a tomar el control y realizar cambios como si se encontrara frente a el, y el programa respondió a las expectativas siendo una buena herramienta de comunicación.

Básicamente los pasos generales que se probaron son los siguientes:

Conexión de la línea telefónica al sistema

Conexión del equipo telefónico a la salida del sistema

Marcación con la segunda línea telefónica

Ingreso al sistema

Activación de la carga, por códigos

Desactivación del sistema, por códigos

Ingreso y verificación de funcionamiento a través de acceso remoto.

Activación y Desactivación por Internet.

## CONCLUSIONES

El presente trabajo tuvo como objetivo realizar el Mando Remoto de una Casa Vía Telefónica, este fue cumplido a satisfacción, se pudo agregar e incorporar opciones de seguridad tanto al comando vía telefónica creando claves de acceso para cada una de las cargas para su activación y desactivación; además de crear una opción que permite Visualizar y manejar en forma remota una casa a través de una PC, dando a este trabajo una herramienta aprovechable no solo en viviendas si no en oficinas industrias garantizando un control eficiente.

En forma general en el presente trabajo de graduación se concentraron conocimientos de electrónica analógica, microcontroladores, informática y redes, dando como resultado final un proyecto que se lo puede implementar en actividades domóticas, comerciales e industriales que requieren comandar cargas en forma remota con seguridad con opciones de ampliación y crecimiento según los requerimientos que necesite cada aplicación en particular.

## RECOMENDACIONES

Las aplicaciones de la electrónica en el campo Residencial, Industrial y Comercial sin lugar a duda marginan el control mecánico y electromecánico por aplicaciones más cómodas, inteligentes que contribuyen a brindar seguridad confort y eficiencia. Se recomienda trabajar con este mando únicamente con líneas telefónicas normales para la recepción de llamadas, ya que está diseñado para cubrir una necesidad de un casa que tiene un servicio básico como es la telefonía. El sistema puede funcionar realizando la llamada desde cualquier teléfono normal o celular ya que las operadoras telefónicas locales y celulares conservan el sistema DTMF, se puede agregar más salidas utilizando un PIC más grande si las necesidades de la aplicación crecieran; recomendando que para mantener una comunicación vía Internet el usuario doméstico en lo posible debería solicitar una dirección IP fija a el proveedor de Internet lo que haría posible mantener un enlace de red continuo siempre a una dirección fija, caso contrario utilizara una IP dinámica que le concederá el Proveedor el momento del enlace, lo cual dificultaría el enlace en forma autónoma.

## BIBLIOGRAFÍA

### REFERENCIAS BIBLIOGRÁFICAS:

- FRENZEL, Louis, 2003, Sistemas Electrónicos de Comunicaciones, 3ra edición, ALFAOMEGA, Mexico.
- HUIDROBO Moya, Jode Manuel, 2001 ,Domotica Edificios Inteligentes, 2da edición, LIMUSA Noriega Editores, México.
- ROMERO Morales, Cristobal, 2003, Domotica e Inmotica, 6ta edición, ALFAOMEGA, México.
- REYES A, Carlos, 2008, Microcontroladores PIC, 3ra edición, DISPERGRAF, Ecuador
- RASHID, M.H., 2004, Electrónica de Potencia, 4ta edición PRENTICE HALL, Argentina
- TOKHEIM, Roger, 2000, Fundamentos de los Microprocesadores, MC GRAW HILL, México.

### REFERENCIAS ELECTRÓNICAS:

- Datasheet, 2010, MT8870, en línea [www.datasheetcatalog.net/es/datasheets\\_pdf/M/T/8/8/MT8870](http://www.datasheetcatalog.net/es/datasheets_pdf/M/T/8/8/MT8870)
- Datasheet, 2010, PIC 16F628A, [www.datasheetcatalog.net/es/datasheets](http://www.datasheetcatalog.net/es/datasheets)
- MICROCHIP, 2010, PIC 16F628A, en línea [www.todoPIC.com.ar](http://www.todoPIC.com.ar),
- Programación de Puertos Serie , 2010, Visual C, en línea [www.lawebdelprogramador.com](http://www.lawebdelprogramador.com),
- Programación en C, 2010, Visual C ++, en línea, [www.lawebdelprogramador.com/cursos/mostrar.php?id=13&texto=C/Visual +C](http://www.lawebdelprogramador.com/cursos/mostrar.php?id=13&texto=C/Visual+C),

- Universidad de Pereira, 2006, Mando Remoto de Vivienda, Colombia, en línea, <http://www.biblioteca.utp.edu.co/tesisdigitales>
- Domótica y control de Viviendas, 2010, Proyectos de Control, en línea, [www.proyectosdomotica.com](http://www.proyectosdomotica.com)
- Telefonica, 2010, Investigacion y Desarrollo Telefónico, en línea, [www.tid.es](http://www.tid.es)
- Optimus Tronic, 2009, Proyectos Electrónicos, en línea, [www.proyctoselectronics.blogspot.com/search/label/Videos%20Electronica](http://www.proyctoselectronics.blogspot.com/search/label/Videos%20Electronica)
- Domodesk, 2010, Todo en Domótica, [www.domodesk.com/content.aspx?co=51&t=21&c=43](http://www.domodesk.com/content.aspx?co=51&t=21&c=43).
- Electrónica Estudio, 2009, Interface USB , en línea, <http://www.electronicaestudio.com/interfase.htm>.