

# **ESCUELA POLITÉCNICA NACIONAL**

## **ESCUELA DE FORMACIÓN DE TECNÓLOGOS**

### **CONSTRUCCIÓN E IMPLEMENTACIÓN DE UNA ALARMA ANTIRROBO CON INTERFAZ DTMF DE LÍNEA TELEFÓNICA UTILIZANDO UN MICROCONTROLADOR PARA LA BIBLIOTECA MANNA PROJECT INTERNATIONAL**

#### **PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN ELECTRÓNICA Y TELECOMUNICACIONES**

**BYRON HERMÓGENES NOGALES MINGA**  
**byrin29@hotmail.com**

**DIRECTOR: ING. CARLOS ROMO**  
**cromo36@hotmail.com**

**Quito, Febrero 2011**

## DECLARACIÓN

Yo, Byron Hermógenes Nogales Minga, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

---

**Byron Hermógenes Nogales Minga**

## **CERTIFICACIÓN**

Certifico que el presente trabajo fue desarrollado por Byron Hermógenes Nogales Minga, bajo mi supervisión.

---

**Ing. Carlos Romo**

**DIRECTOR DE PROYECTO**

## AGRADECIMIENTOS

A Dios por brindarme la luz de un nuevo día y así ver realizado uno de mis triunfos profesionales, le doy gracias también por permitirme tener a mi lado a los seres más excepcionales, constantes, cariñosos, comprensivos y pacientes a los cuales respeto y amo con todo mi corazón como lo son mis padres Nanci Elizabeth y Elicio Hermógenes que con tanto esfuerzo y pese a tantas adversidades de la vida se dieron modos para regalarme la mejor y más preciada herencia que es la educación.

A mis hermanos Darío Benjamín y Santiago Ricardo por su constante e incomparable apoyo en todo momento, por estar presente en cada triunfo así como en cada derrota y por ser un ejemplo a seguir

A mi cuñada Valeria, a mi sobrina Estefanía y a la luz de mi vida Santiago Isaac mi sobrino que con el pasar del tiempo se arraiga en mi corazón y que han sido una gran fuente de inspiración, amor y ternura.

A mis grades amigos y amigas que han sido parte de los buenos y malos momentos en el transcurso de la vida especialmente a Verónica Gordillo, Paola Guerra, Eduardo Guevara, Jorge Pillajo y Michel Castro.

Al Ing. Carlos Romo por su acertada dirección y guía continua en el desarrollo del presente proyecto.

## **DEDICATORIA**

Para quienes a través de los años me han sabido brindar su amor y apoyo incondicional; que me han enseñado que aun entre los mayores problemas y adversidades, la solución nunca será rendirse; que han respetado mi individualidad indicando que solo yo debo resolver mis problemas, aleccionándome a tener sangre fría en los momentos de crisis, y cautela en los momentos de triunfo.

A mi familia mil gracias.

## CONTENIDO

<b>RESUMEN</b> .....	<b>IX</b>
----------------------	-----------

<b>PRESENTACIÓN</b> .....	<b>XV</b>
---------------------------	-----------

### CAPITULO 1

#### FUNDAMENTOS TEÓRICOS

<b>1.1</b>	<b>RED DE TELECOMUNICACIONES</b> .....	<b>1</b>
	1.1.1 COMPONENTES DE UNA RED DE TELECOMUNICACIONES .....	1
	1.1.1.1 Red de acceso .....	2
	1.1.1.2 Nodo de conmutación .....	2
	1.1.1.3 Red de transporte.....	3
	1.1.1.4 Equipos terminales .....	3
<b>1.2</b>	<b>RED DE TELEFÓNIA PÚBLICA CONMUTADA</b> .....	<b>3</b>
	1.2.1 TELEFONÍA FIJA ALÁMBRICA .....	4
	1.2.1.1 Inicio de llamada.....	5
	1.2.1.2 Comunicación con la central.....	8
	1.2.1.3 Comunicación entre centrales.....	9
	1.2.1.4 Final de la llamada .....	10
	1.2.2 TELEFONÍA FIJA INALÁMBRICA .....	11
<b>1.3</b>	<b>TELEFONÍA MÓVIL</b> .....	<b>12</b>
	1.3.1 INTRODUCCIÓN.....	12
	1.3.2 FUNCIONAMIENTO .....	13
<b>1.4</b>	<b>EL MICROCONTROLADOR</b> .....	<b>15</b>
	1.4.1 INTRODUCCIÓN.....	15
	1.4.2 DIFERENCIA ENTRE MICROCONTROLADOR Y MICROPROCESADOR .....	16
	1.4.3 RECURSOS COMUNES A TODOS LOS MICROCONTROLADORES .....	19
	1.4.4 RECURSOS ESPECIALES.....	25
	1.4.5 COMO SABER QUE MICROCONTROLADOR UTILIZAR .....	29
<b>1.5</b>	<b>MICROCONTROLADOR PIC 16F877A</b> .....	<b>31</b>
	1.5.1 CARACTERISTICAS .....	32
	1.5.2 NOMENCLATURA .....	33
	1.5.3 ARQUITECTURA.....	34
	1.5.4 ORGANIZACIÓN DE LA MEMORIA .....	38
	1.5.5 DIAGRAMA DE PINES Y FUNCIONES.....	43

## CAPITULO 2

### CONSTRUCCIÓN DEL HARDWARE Y DESARROLLO DEL SOFTWARE DEL SISTEMA

<b>2.1</b>	<b>INTRODUCCIÓN.....</b>	<b>46</b>
<b>2.2</b>	<b>CONSTRUCCIÓN DEL HARDWARE DEL SISTEMA .....</b>	<b>46</b>
2.2.1	ETAPA DE CONTROL.....	47
2.2.2	ETAPA DE ALIMENTACIÓN .....	51
2.2.3	ETAPA DE INICIALIZACIÓN .....	54
2.2.4	ETAPA DE DETECCIÓN.....	58
2.2.5	ETAPA DE ACTIVACIÓN .....	59
<b>2.3</b>	<b>DESARROLLO DEL SOFTWARE DEL SISTEMA .....</b>	<b>62</b>
2.3.1	DIAGRAMA DE FLUJO.....	62
2.3.2	LENGUAJE DE PROGRAMACIÓN UTILIZADO .....	64
2.3.3	ELABORACIÓN DEL PROGRAMA PARA EL SISTEMA .....	69

## CAPITULO 3

### IMPLEMENTACIÓN Y PRUEBAS DE FUNCIONAMIENTO

<b>3.1</b>	<b>IMPLEMENTACIÓN DEL HARDWARE.....</b>	<b>84</b>
3.1.1	DISEÑO DE PLANO PARA INSTALACIÓN DE SENSORES Y UNIDAD CENTRAL DEL SISTEMA.....	84
3.1.2	INSTALACIÓN DE CANALETA Y CABLE PARA SENSORES .....	85
3.1.3	INTALACIÓN DE SENSORES .....	85
3.1.4	INSTALACIÓN DE SIRENA Y CABLE TELEFÓNICO .....	87
3.1.5	INSTALACIÓN DE LA UNIDAD CENTRAL DEL SISTEMA .....	88
3.1.6	ALIMENTACIÓN DE VOLTAJE AL SISTEMA.....	90
<b>3.2</b>	<b>PRUEBAS DE FUNCIONAMIENTO DEL SISTEMA.....</b>	<b>91</b>
3.2.1	ENCENDIDO E INICIALIZACIÓN DEL SISTEMA.....	91
3.2.2	MENU DEL SISTEMA.....	95
3.2.3	PRUEBA DE SENSORES .....	102
3.2.4	PRUEBA DE SIRENA Y LLAMADA TELEFÓNICA.....	113

## CAPITULO 4

### CONCLUSIONES Y RECOMENDACIONES

<b>4.1</b>	<b>CONCLUSIONES .....</b>	<b>116</b>
<b>4.2</b>	<b>RECOMENDACIONES .....</b>	<b>117</b>
	<b>BIBLIOGRAFÍA .....</b>	<b>118</b>

**ANEXOS:**

**ANEXO A:** Especificaciones Técnicas del Regulador LM7805.

**ANEXO B:** Planos diseñados para la implementación de la alarma antirrobo.



## INDICE DE TABLAS

### CAPITULO 1

TABLA 1-1. CARACTERÍSTICAS PERIFÉRICAS.....	33
TABLA 1-2. NOMENCLATURA SEGÚN EL TIPO DE MEMORIA.....	34
TABLA 1-3. NOMENCLATURA SEGÚN EL RANGO DE VOLTAJE. ....	34
TABLA 1-4. REGISTRO STATUS (DIRECCIONES 03H, 83H, 103H, 183H) .....	40
TABLA 1-5. REGISTRO INTCON (DIRECCIONES 0BH, 8BH, 10BH, 18BH) .....	41
TABLA 1-6. NOMBRE Y DESCRIPCIÓN DE CADA UNO DE LOS PINES DEL PIC16F877A .....	43

### CAPITULO 2

TABLA 2-1. DISTRIBUCIÓN DE PINES PARA CADA ETAPA CON SUS RESPECTIVAS FUNCIONES.....	50
TABLA 2-2. DECLARACIONES DISPONIBLES EN EL COMPILADOR PICBASIC PRO.....	67

### CAPITULO 3

TABLA 3-1. DISTRIBUCIÓN DE SENSORES EN LA BIBLIOTECA.....	84
---	----

## INDICE DE FIGURAS

### CAPITULO 1

FIGURA 1-1. COMPONENTES DE UNA RED DE TELECOMUNICACIONES.....	1
FIGURA 1-2. ESTRUCTURA DE LA TELEFONÍA FIJA ALÁMBRICA.....	4
FIGURA 1-3. CIRCUITO DE MARCACIÓN POR PULSOS.....	6
FIGURA 1-4. TIEMPOS EN EL DISCADO POR PULSOS.....	6
FIGURA 1-5. CIRCUITO DE MARCACIÓN POR TONOS.....	7
FIGURA 1-6. CODIFICACIÓN DMTF PARA TECLADO TELEFÓNICO.....	7
FIGURA 1-7. TIEMPO DE UNA SEÑAL LIBRE.....	8
FIGURA 1-8. TIEMPO DE SEÑAL OCUPADA.....	9
FIGURA 1-9. ESQUEMA DE LA COMUNICACIÓN ENTRE CENTRALES.....	10
FIGURA 1-10. ARQUITECTURA GENÉRICA DE WLL <sup>11</sup> .....	11
FIGURA 1-11. DIAGRAMA DEL FUNCIONAMIENTO DE LA TELEFONÍA MÓVIL.....	14
FIGURA 1-12. MICROPROCESADOR.....	16
FIGURA 1-13. ESTRUCTURA DE UN MICROCONTROLADOR.....	18
FIGURA 1-14. LA ARQUITECTURA HARVARD DISPONE DE DOS MEMORIAS INDEPENDIENTES PARA DATOS Y PARA INSTRUCCIONES, PERMITIENDO ACCESOS SIMULTÁNEOS.....	20
FIGURA 1-15. TIPOS DE ARQUITECTURA.....	34
FIGURA 1-16. DIAGRAMA DE LA ARQUITECTURA INTERNA DEL PIC 16F877A.....	36
FIGURA 1-17. MAPA DE LA MEMORIA DE PROGRAMA Y STACK DEL PIC16F877A ESPECIALES.....	38
FIGURA 1-18. MAPA DEL ARCHIVO DE REGISTROS.....	39
FIGURA 1-19. DISTRIBUCIÓN DE PINES DEL MICROCONTROLADOR PIC16F877A.....	43

## **CAPITULO 2**

FIGURA 2-1. DIAGRAMA EN BLOQUES DE LA ALARMA ANTIRROBO. ....	<b>47</b>
FIGURA 2-2. ETAPA DE CONTROL (MICROCONTROLADOR 16F877A) .....	<b>48</b>
FIGURA 2-3. DIAGRAMA DEL CIRCUITO DE RESET. ....	<b>48</b>
FIGURA 2-4. DIAGRAMA DEL CIRCUITO PARA CONECTAR EL OSCILADOR EXTERNO .....	<b>49</b>
FIGURA 2-5. ALIMENTACIÓN DE TENSIÓN ALTERNIA Y CONEXIÓN DE FUSIBLE. ....	<b>52</b>
FIGURA 2-6. CIRCUITO DE ALIMENTACIÓN AL SISTEMA. ....	<b>52</b>
FIGURA 2-7. DIAGRAMA DE CONEXIÓN DE UN MÓDULO LCD.....	<b>55</b>
FIGURA 2-8. DIAGRAMA DE CONEXIÓN DEL TECLADO.....	<b>56</b>
FIGURA 2-9. CIRCUITO DE CONEXIÓN PARA EL SENSOR DE PRESENCIA. ....	<b>58</b>
FIGURA 2-10. CIRCUITO DE CONEXIÓN DE UN SENSOR MAGNÉTICO.....	<b>59</b>
FIGURA 2-11. CIRCUITO DE CONEXIÓN A LA SIRENA. ....	<b>60</b>
FIGURA 2-12. CIRCUITO DE CONEXIÓN A LA LÍNEA TELEFÓNICA.....	<b>61</b>
FIGURA 2-13. NIVELES DE PROGRAMACIÓN.....	<b>64</b>
FIGURA 2-14. PARTES DEL EDITOR MICROCODE .....	<b>66</b>

## **CAPITULO 3**

FIGURA 3-1. CONEXIÓN DEL SENSOR DE PRESENCIA. ....	<b>85</b>
FIGURA 3-2. SELECCIÓN DE LA SENSIBILIDAD EN EL SENSOR.....	<b>86</b>
FIGURA 3-3. MOVIMIENTOS PARA VARIAR EL RANGO DE DETECCIÓN DEL SENSOR. ....	<b>86</b>
FIGURA 3-4. INSTALACIÓN DE SENSOR MAGNÉTICO.....	<b>87</b>
FIGURA 3-5. SIRENA.....	<b>87</b>

FIGURA 3-6. PLACA ELECTRÓNICA DE LA ALARMA DONDE SE INDICA LA CONEXIÓN DE LOS DIFERENTES ELEMENTOS DEL SISTEMA.....	88
FIGURA 3-7. CONEXIÓN DEL SENSOR DE PRESENCIA A LA UNIDAD CENTRAL .....	89
FIGURA 3-8. CONEXIÓN DE LOS SENSORES MAGNÉTICOS A LA UNIDAD CENTRAL .....	89
FIGURA 3-9. CONEXIÓN DE BUSES DE DATOS PARA EL LCD Y EL TECLADO .....	90
FIGURA 3-10. INGRESO DE VOLTAJE ALTERNO A LA UNIDAD CENTRAL DESDE UN TOMACORRIENTE CERCANO.....	90
FIGURA 3-11. CONEXIÓN PARA LA ALIMENTACIÓN DEL SISTEMA Y SWITCH DE ENCENDIDO .....	91
FIGURA 3-12. ENCENDIDO DEL SISTEMA .....	92
FIGURA 3-13. PROCESO DE INGRESO DE CLAVE PARA ACCEDER AL MENÚ PRINCIPAL. ....	93
FIGURA 3-14. PRUEBA DE UN MAL INGRESO DE CLAVE.....	95
FIGURA 3-15. MENÚ PRINCIPAL.....	95
FIGURA 3-16. OPCIÓN A DEL MENÚ PRINCIPAL PARA ACTIVAR LA ALARMA.....	96
FIGURA 3-17. OPCIÓN A DEL MENÚ PRINCIPAL PARA DESACTIVAR LA ALARMA. ....	98
FIGURA 3-18. INGRESO DE UN NUEVO NÚMERO TELEFÓNICO FIJO. ....	99
FIGURA 3-19. INGRESO DE UN NUEVO NÚMERO TELEFÓNICO MÓVIL. ....	101
FIGURA 3-20. CAMBIO DE CLAVE PARA INGRESAR AL SISTEMA. ....	102
FIGURA 3-21. PRUEBA DEL SENSOR DE PRESENCIA 1.....	103
FIGURA 3-22. PRUEBA DE SENSOR DE PRESENCIA 3. ....	104
FIGURA 3-23. PRUEBA DE SENSOR DE PRESENCIA 2. ....	105
FIGURA 3-24. PRUEBA DE SENSOR DE PRESENCIA 4 .....	106
FIGURA 3-25. PRUEBA DE SENSOR DE PRESENCIA 5. ....	107
FIGURA 3-26. PRUEBA DE SENSOR MAGNÉTICO 1.....	108

FIGURA 3-27. PRUEBA DE SENSOR DE PRESENCIA 2. ....	<b>109</b>
FIGURA 3-28. PRUEBA DE SENSOR MAGNÉTICO 3. ....	<b>110</b>
FIGURA 3-29. PRUEBA DE SENSOR MAGNÉTICO 4. ....	<b>111</b>
FIGURA 3-30. PRUEBA DE SENSOR MAGNÉTICO 5. ....	<b>112</b>
FIGURA 3-31. PROCESO Y PRUEBA DE ACTIVACIÓN DE LA ALARMA. ....	<b>114</b>
FIGURA 3-32. LLAMADA TELEFÓNICA.....	<b>115</b>

## RESUMEN

El presente proyecto desarrolla la construcción e implementación de una alarma antirrobo con interfaz DTMF de línea telefónica utilizando un microcontrolador para la Biblioteca Manna Project International, en la cual a más de colocar una sirena se incorporó una nueva función, la cual consiste en que el sistema realice automáticamente una llamada a un delegado de la Biblioteca, mismo que al momento de contestar escuchará la emisión de tonos a diferentes frecuencias e intervalos lo que le permitirá identificar que ha sucedido un imprevisto, permitiendo de esta manera que el propietario de la biblioteca se organice para tomar las medidas adecuadas.

El proyecto utiliza diferentes dispositivos eléctricos y electrónicos para conseguir el funcionamiento correcto de la alarma, con la única finalidad de proporcionar un ambiente de trabajo seguro, para los niños y jóvenes del sector de Rumiloma, que utilizan los servicios de la Biblioteca Manna Project International.

Para presentar detalladamente cómo se logró el objetivo de este proyecto, la parte escrita se ha conformado de cuatro capítulos, dispuestos de la siguiente manera:

En el Capítulo 1: se describen temas comprendidos para el desarrollo de la alarma con interfaz DTMF de línea telefónica, se detalla un estudio de los microcontroladores y en particular se describe el microcontrolador 16F877A.

En el Capítulo 2: se explica principalmente la construcción del hardware de la alarma, es decir se describen las etapas que conforman el circuito con sus respectivos diagramas, además se da a conocer los principales componentes electrónicos que interviene en la construcción e implementación de la alarma. Conjuntamente se encuentra el programa para el microcontrolador con su diagrama de flujo.

En el Capítulo 3: se detalla la implementación de la alarma y las pruebas de funcionamiento realizadas a los distintos dispositivos utilizados, obteniendo así resultados satisfactorios una vez que el sistema de seguridad se ha implementado completamente en la Biblioteca Manna Project International, comprobando así la validez del sistema que se construyó.

En el Capítulo 4: se mencionan las conclusiones obtenidas durante el proceso de elaboración y las recomendaciones del trabajo realizado.

Los resultados obtenidos tanto a nivel de hardware y software cumplen con las tareas de registrar la activación de los sensores de presencia y sensores magnéticos, por ende encender la sirena y simultáneamente comenzar a llamar a los teléfonos previamente grabados en el microcontrolador.

## PRESENTACIÓN

El hombre siempre busca mejorar sus condiciones de vida, en todos los aspectos que le sean posibles. La seguridad dentro de una edificación es uno de estos aspectos, y con los sistemas de detección y alarma se puede evitar pérdidas humanas y materiales al alertar rápidamente de una incursión ajena.

En la actualidad en todas las edificaciones es muy importante este tipo de sistemas, ya que cada vez es más necesario contar con sistemas de seguridad física que garanticen la estabilidad de personas y activos, y la tendencia indica que un método muy elegido son las alarmas de seguridad antirrobo, ya que permiten alertar a los propietarios para tomar medidas adecuadas.

Identificada la necesidad de proveer mayor seguridad a las edificaciones y que cumpla con las necesidades de resguardar los bienes existentes, me he visto en la necesidad de construir e implementar una alarma antirrobo en un lugar que presta servicios a niños y jóvenes del sector de Rumiloma esta es en la Biblioteca Manna Project Internacional. Este proyecto consta de una sirena, la cual al momento de activarse será una señal para conocer que se ha suscitado un imprevisto, a esto se suma una nueva función al sistema de detección, que consiste en que el sistema automáticamente realizará una llamada al propietario y al momento de contestar escuchará la emisión de tonos a diferentes frecuencias e intervalos.

Por tanto, este proyecto desarrolla la construcción e implementación de una alarma la cual esta comandada por un microcontrolador PIC 16F877A.



# CAPITULO 1

## FUNDAMENTOS TEÓRICOS

### 1.1 RED DE TELECOMUNICACIONES<sup>1</sup>

Es un conjunto de terminales conectados a un punto. Este punto se denomina NODO. Su función es establecer un único camino entre cada par de terminales que deseen establecer una comunicación.

La red proporciona las vías de comunicación necesarias para establecer las interconexiones. Para ello dispone de un conjunto organizado de recursos que se comparten entre todos los usuarios. La red solamente establece las conexiones que se le solicitan, a diferencia de la Conectividad Total, en que las conexiones son permanentes aunque los abonados no se comuniquen. El acceso de la señal a la red, su encaminamiento y transmisión debe atenerse a una organización o disciplina. Tales normas de funcionamiento se denominan Protocolos. Los protocolos se realizan mediante señales especiales que no son de información propiamente dicha, pero que resultan esenciales para la telecomunicación. Al conjunto de estas señales se les denomina señalización.

#### 1.1.1 COMPONENTES DE UNA RED DE TELECOMUNICACIONES

Para los sistemas conmutados los componentes básicos de la red son:

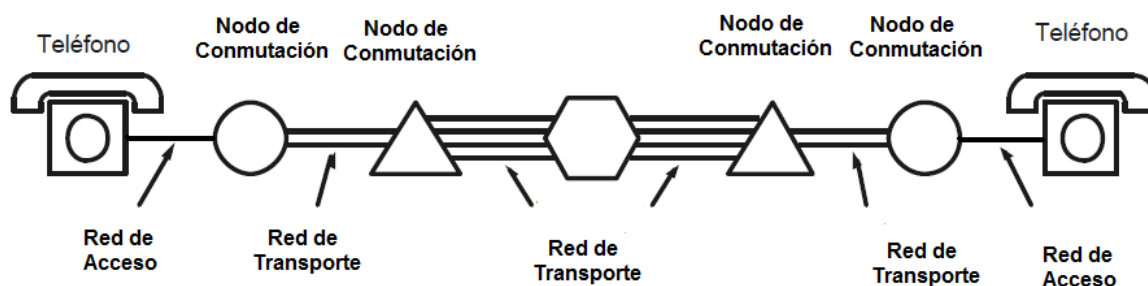


Figura 1-1. Componentes de una red de telecomunicaciones.<sup>2</sup>

<sup>1</sup> AULESTIA, Hugo, Apuntes de conmutación

<sup>2</sup> <http://www.rootshell.be/~wcruzy/cd/sistel1.pdf>

### **1.1.1.1 Red de acceso**

Es aquella porción de los medios de transmisión que sirve para conectar un equipo terminal con el nodo de conmutación. Pueden clasificarse de la siguiente manera:

1. Cableadas:
  - Cobre
  - Fibra
2. Inalámbricas no móviles o con movilidad restringida:
  - WLAN 802.11
  - WiMAX 802.16
3. Inalámbricas móviles/celulares:
  - GSM / GPRS
  - CDMA
  - 3G

### **1.1.1.2 Nodo de conmutación**

Es la parte de una red de telecomunicaciones cuyo objetivo básico es el de conectar una entrada de cualquier tipo (equipo terminal, línea de enlace, etc.) que llegue hacia él, con cualquier salida que conduzca al destino deseado por el usuario llamante. Una forma de clasificar a los nodos de conmutación sería la siguiente:

1. Por su estructura:
  - Electromecánica
  - Semielectrónica
  - Electrónica
2. Por su tecnología:
  - Analógica
  - Digital
3. Por las técnicas de conmutación:
  - De circuitos
  - De paquetes

### **1.1.1.3 Red de transporte**

Es aquella porción de los medios de transmisión que sirve para conectar entre si los equipos de conmutación.

Las redes de transporte, según el medio físico se pueden clasificar en:

1. Alámbricas
2. Inalámbricas

Las redes de transporte generalmente utilizan medios multiplexados:

1. Analógicos
2. Digitales

### **1.1.1.4 Equipos terminales**

Son los elementos que sirven para el ingreso a una red de telecomunicaciones y que podría considerarse como el interfaz hombre-red. Los equipos terminales pueden ser teléfono, videoteléfono, facsímil, computador, etc.

## **1.2 RED DE TELEFÓNIA PÚBLICA CONMUTADA<sup>3</sup>**

La telefonía es la ciencia que tiene por objeto la transmisión de sonidos a distancia. Trata por tanto, del estudio y utilización de los medios y procedimientos para el transporte del sonido. Por ello, las compañías telefónicas se pueden considerar empresas de transportes. Los clientes les entregan sonidos y la obligación de las empresas es transportarlos con el mínimo deterioro posible, a la dirección que el cliente diga. Como en el modelo general, en el de telefonía, hay un elemento donde se genera la información y otro destinatario, donde se presenta de forma comprensible para los hombres. Como la comunicación telefónica, se establece en los dos sentidos, los dos terminales serán idénticos, y los conocemos por terminal de abonado o teléfono.

---

<sup>3</sup> [http://elsitiodetelecomunicaciones.iespana.es/telefonía\\_basica.htm](http://elsitiodetelecomunicaciones.iespana.es/telefonía_basica.htm)

La Red de Telefonía Pública Conmutada fue creada, para la transmisión de sonido (voz). La forma es idéntica a la Red de Telecomunicación General que hemos visto, salvo que, en la Red Telefónica, al Nodo se le llama Central.

La red telefónica es el sistema que permite reducir el coste económico necesario para efectuar una comunicación entre dos terminales de abonados, ya que los medios y equipos de transmisión están compartidos. La finalidad de la red es el transporte de la información desde un origen hasta un destino. Esta red es Conmutada, ya que un terminal origen al seleccionar un terminal destino, la red se encarga de proporcionar un camino entre ambos, efectuando las conmutaciones o cambios de camino necesarios en las centrales.

### 1.2.1 TELEFONÍA FIJA ALÁMBRICA<sup>4</sup>

Es un servicio de telecomunicaciones que permite el intercambio bidireccional de tráfico de voz en tiempo real, entre diferentes usuarios a través de una red de conmutación de circuitos, que es la técnica que permite que el equipo terminal emisor se conecte físicamente con el equipo terminal receptor mediante un circuito único y específico asignado previamente al inicio de la comunicación y que permanecerá activo durante todo el tiempo que dure dicha comunicación.

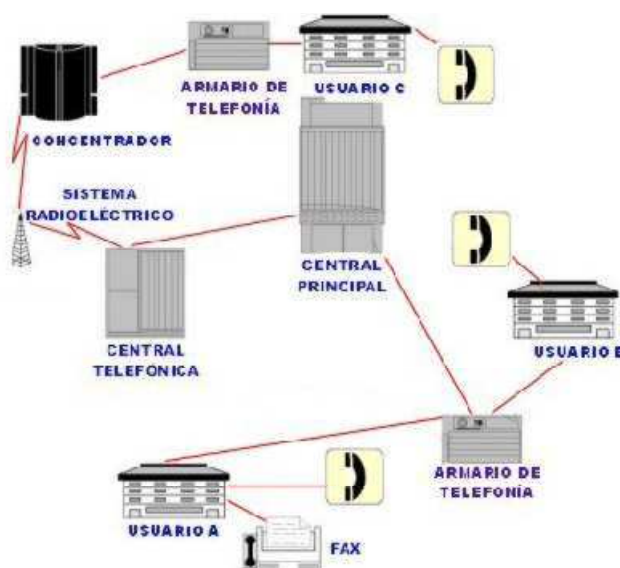


Figura 1-2. Estructura de la telefonía fija alámbrica.

<sup>4</sup> <http://www.supertel.gov.ec/index.php/servicios-de-telecomunicaciones/62-telefonía-fija/>

Para que se establezca una llamada telefónica, debe establecerse un circuito de comunicaciones entre dos puntos, este circuito establece en base a un proceso de señalización que se inicia una vez que el abonado que llama levanta el auricular. Las centrales telefónicas o de conmutación constituyen la parte operativa que permiten que se establezca esta comunicación, y son las encargadas de enrutar las llamadas a sus destinos correspondientes.

### 1.2.1.1 Inicio de llamada<sup>5</sup>

La línea telefónica que va al abonado y está conectada al terminal telefónico tiene una tensión continua de 50v. Al descolgar el terminal, un interruptor conecta a la línea una impedancia de 600 ohmios cayendo la tensión continua a 10v de tal manera que la central detecta este estado, poniéndose en disposición de recibir instrucciones para realizar la llamada y emitiendo un tono de indicación en espera. Este estado se mantiene hasta que envían instrucciones de llamada o pasa el tiempo predeterminado por la central y envía tono de comunicar para pasar luego a desconectar la línea.

Las instrucciones de llamadas se pueden realizar por dos métodos.

- **Por Pulsos.**<sup>6</sup> (O pulsación decádica) es el clásico sistema de disco el cual consiste en un disco giratorio provisto de diez agujeros, de aquí lo de decádica, numerados del 0 al 9, este teléfono hace que por el mismo par de alambres de cobre vaya una señalización que indica el destino de la conversación. Esto se realiza interrumpiendo por periodos cortos de tiempo la corriente de lazo, tantas veces como el dígito discado es decir que se produce emitiendo pulsos cuadrados con una impedancia muy baja y otra muy alta.

---

<sup>5</sup> <http://www.cinit.org.mx/articulo.php?idArticulo=7>

<sup>6</sup> [http://es.wikipedia.org/wiki/Marcaci%C3%B3n\\_dec%C3%A1dica\\_por\\_pulsos](http://es.wikipedia.org/wiki/Marcaci%C3%B3n_dec%C3%A1dica_por_pulsos)

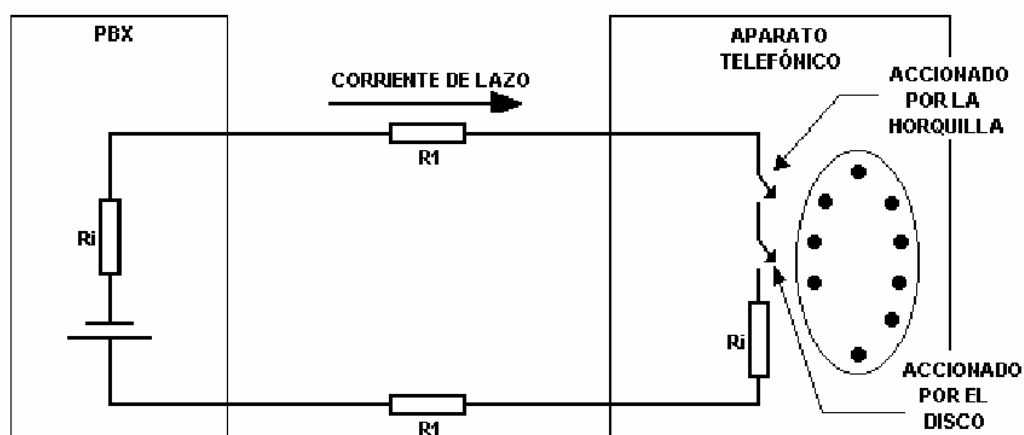


Figura 1-3. Circuito de marcación por pulsos.

El discado se realiza a diez pulsos por segundo, siendo cada dígito representado por 60ms de lazo abierto y 40ms de lazo cerrado. Entre cada dígito deben transcurrir como mínimo 250ms (ver Figura 1-4.). Actualmente, este tipo solo se usan por compatibilidad con los antiguos sistemas, pero nos permiten a veces saltarse las protecciones para evitar llamadas puestas por los sistemas de tonos.

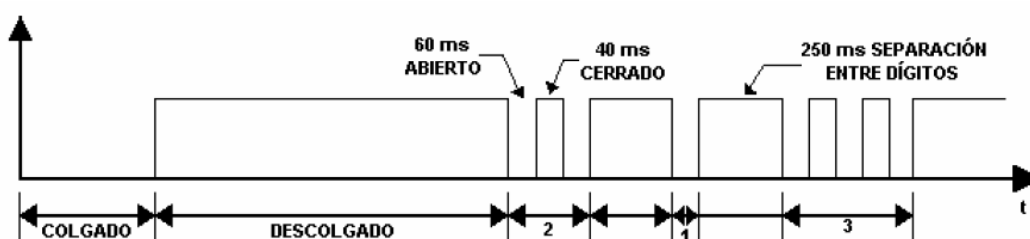


Figura 1-4. Tiempos en el discado por pulsos.

- **Por Tonos Multifrecuencia.**<sup>7</sup> Están constituidos por un par de frecuencias asignadas a cada dígito. Son los mismos que los DTMF (Dual Tono MultiFrecuencia) que emite cualquier modem. Al pulsar un dígito el sistema genera una señal de audio compuesta por la suma de dos frecuencias (la correspondiente a la fila mas la correspondiente a la columna), que pueden ser luego detectadas en la central, por medio de filtros adecuados.

<sup>7</sup> REYES, Carlos, Microcontroladores PIC, NARVAEZ, Carlos, Interfaces telefónicas.

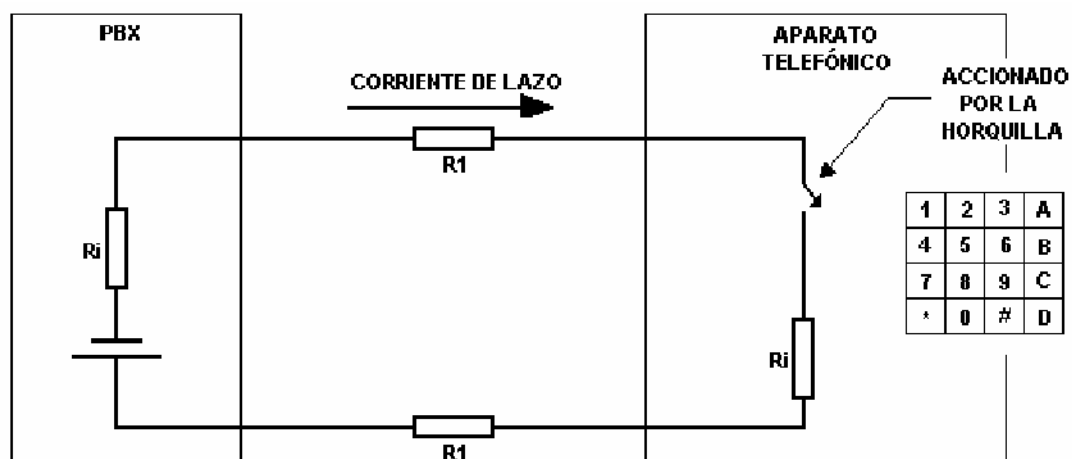


Figura 1-5. Circuito de marcación por tonos.

Las frecuencias estandarizadas presentan dos grupos uno de frecuencias altas que corresponden a las columnas y otro de bajas que corresponde a las filas como se muestra en la Figura 1-6, el tiempo de duración de cada dígito es de 100ms y el espacio entre ellos es de 40ms.

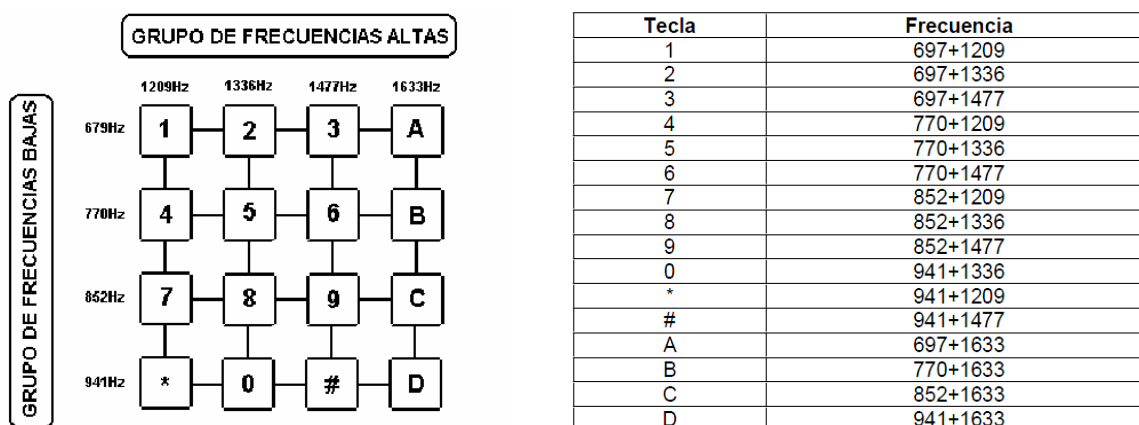


Figura 1-6. Codificación DTMF para teclado telefónico.<sup>8</sup>

Este sistema supera al de marcación por pulsos por cuanto disminuye la posibilidad de errores de marcación, al no depender de un dispositivo mecánico. Por otra parte es mucho más rápido ya que no hay que esperar tanto tiempo para que la central detecte las interrupciones, según el número marcado.

<sup>8</sup> <http://platea.pntic.mec.es/rrodrigu/es/Interface%20DTMF.pdf>

No obstante, las modernas centrales telefónicas de conmutación digital, controladas por ordenador, siguen admitiendo la conexión de terminales telefónicos con ambos tipos de marcación.

### 1.2.1.2 Comunicación con la central

La central a la que tenemos conectado el equipo terminal (teléfono) se comunica con nosotros a través de unos tonos que nos indican el estado de la línea. Los tonos que nos envía la centralita son:

- **Tono de invitación a marcar.** Es enviado cuando descolgamos el equipo terminal sin haber recibido una llamada. Nos indica que podemos marcar un número para que se realice la llamada o introducir alguna función en las líneas multiservicio. Es un tono continuo de 400 Hz.
- **Tono de llamada.** Se produce a raíz de haber terminado de marcar el número al que queremos llamar y nos indica el número de RINGS que se están produciendo en el otro extremo de la llamada. La frecuencia es de 400 Hz y una cadencia de un segundo y medio sonando y cuatro segundos de silencio tal como se muestra en la Figura 1-7.



Figura 1-7. Tiempo de una señal libre

- **Tono de ocupado.** Se produce cuando el teléfono al que estamos llamando tiene la línea ocupada (descolgado). La frecuencia es de 400 Hz con un tono audible intermitente como indica la Figura 1-8.



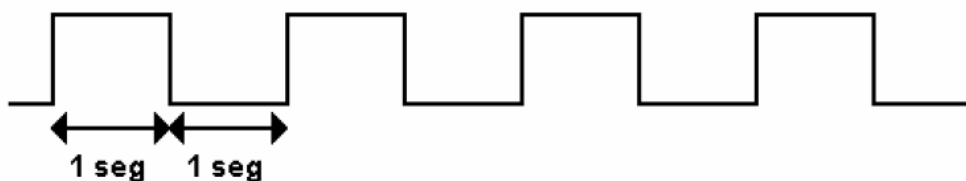


Figura 1-8. Tiempo de señal ocupada.

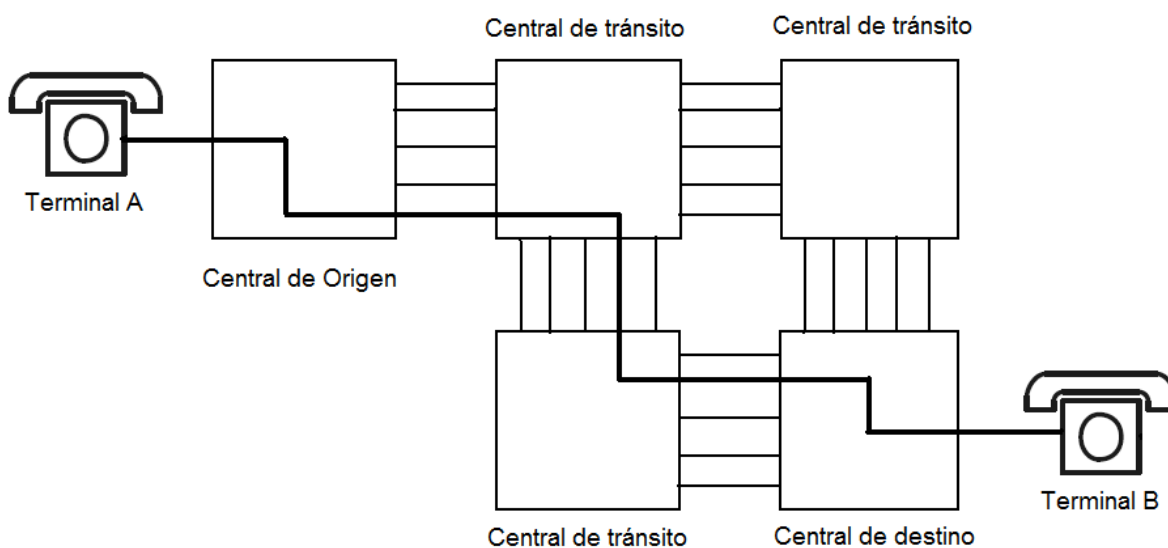
- **Tono de saturación.** Se produce cuando la llamada no llega a su destino por algún problema técnico. Los técnicos siguen la cadencia de serie 400 Hz y silencio, con un intervalo de 200 ms, durante tres segundos.
- **Señal de llamada.** La central usa una señal de bastante potencia ya que puede soportar un consumo grande para indicar la llamada de alguien a nuestro equipo terminal. Es la que activa nuestro teléfono y es una señal alterna de 25 Hz con 75 v eficaces, con una cadencia de un segundo y medio activa y cuatro segundos inactiva.

### 1.2.1.3 Comunicación entre centrales

La central inicial con los datos de la llamada que se le proporcionado desde el equipo terminal del usuario inicia la comunicación conectándose con otras centrales e informándolas del número al que tiene que llegar la llamada.

Para ello pone en modo de comandos a la otra central enviándole FC (Frecuencia de Control, un tono de 1700 Hz) y cuando esta pone en silencio la línea, se le envía los cuatro primeros números del número de teléfono destino, seguidamente la otra central envía un tono de 1700 Hz para confirmar que ha recibido los datos. La central inicial manda otra FC y la central de recepción corta el tono para recibir el resto del número y es así que la central inicial manda los últimos dígitos del número de teléfono.

Esta operación se repite, convirtiéndose cada vez la central de destino en la inicial del próximo tramo hasta que la llamada llega a su destino como se muestra en la Figura 1-9.



**Figura 1-9.** Esquema de la comunicación entre centrales<sup>9</sup>

En el momento que la llamada llega a su destino se abre esta línea y tenemos conexión directa con el otro teléfono (podemos oír como suenan los tonos en el otro extremo) y continuará en este estado hasta que alguno de los terminales cuelgue. Cuando una de las centrales detecta que se ha cortado introduce una frecuencia FD (Frecuencia de Desconexión) para dejar esa línea en estado de recibir instrucciones o estado de TRUNK<sup>10</sup>. Este proceso se repite entre las centrales hasta que todas las líneas queden en este estado.

#### 1.2.1.4 Final de la llamada

La central a la que está conectada el terminal destino envía una señal de llamada al terminal telefónico de destino para indicarle que está recibiendo una llamada. Esta señal se mantiene hasta que se descuelga el terminal telefónico de destino, hasta que se cuelga en origen o hasta que se supera el tiempo máximo programado en la central para que suene el terminal telefónico.

<sup>9</sup> AULESTIA, Hugo, Apuntes de conmutación

<sup>10</sup> Se le llama a estado de trunk cuando alguna central se queda en modo recepción de datos y podemos desde el terminal telefónico enviar comandos como si fuéramos otra central.

## 1.2.2 TELEFONÍA FIJA INALÁMBRICA<sup>11</sup>

La Unidad de Suscriptor Fija (FSU). Es una interfaz entre los dispositivos cableados del suscriptor y la red de WLL. Los dispositivos cableados con alambre pueden ser computadoras o facsímiles así como teléfonos. Varios sistemas utilizan otras siglas para el FSU tal como la unidad fija de acceso inalámbrico (WAFU), la unidad de radio del suscriptor (RSU), o la unidad de interfaz de red inalámbrica fija (FWNIU). El FSU realiza la codificación/decodificación del canal, modulación/demodulación y transmisión/recepción de la señal vía radio, según la especificación de interfaz de aire. En caso de necesidad, el FSU también realiza la codificación/decodificación en la fuente.

Cuando se usa un teléfono simple, el FSU puede realizar la función de generación de tono de marcado para los usuarios que no se enterarán que se trata de un sistema de WLL. El FSU también apoya los dispositivos computarizados para conectarse a la red usando módem de banda de voz o canales de datos dedicados.

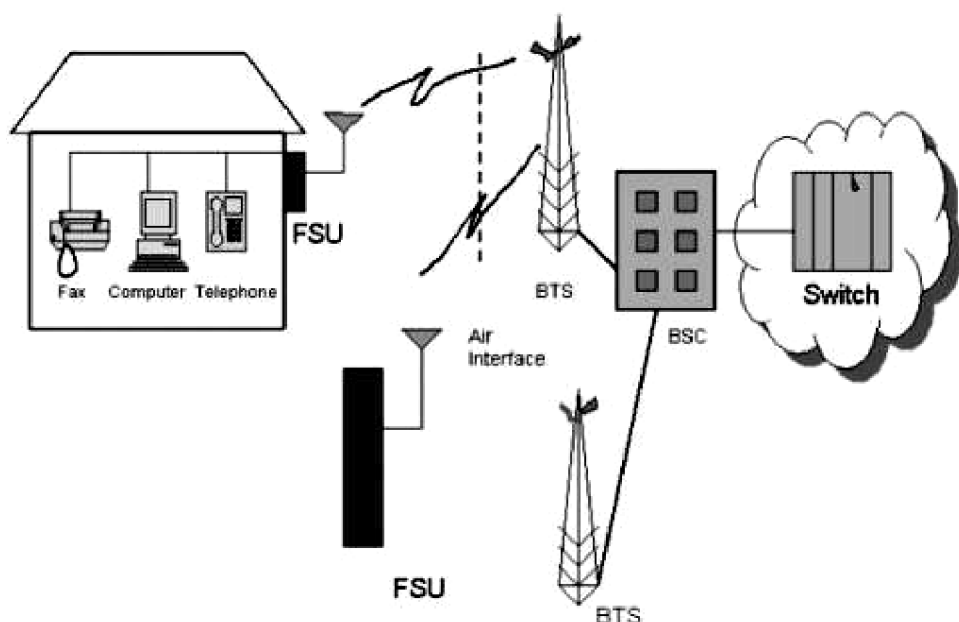


Figura 1-10. Arquitectura genérica de WLL<sup>11</sup>

<sup>11</sup> <http://www.supertel.gov.ec/index.php/servicios-de-telecomunicaciones/62-telefonía-fija/>

- **Interface de Aire.** Un FSU está conectado con la estación base vía banda de radio que está a varios centenares de MHz o alrededor de 2GHz; se pueden utilizar antenas direccionales fijas de alta ganancia entre el FSU y la estación base, teniendo en cuenta la línea de vista (o por lo menos, cercana), así, la señal de WLL es un canal con ruido Gaussiano, lo que aumenta drásticamente la eficiencia del canal y la capacidad del sistema.
- **La Estación Base.** La estación base está conformada usualmente por dos partes, el sistema estación base transmisor-receptor (BTS) y el controlador estación base (BSC). En muchos sistemas, el BTS realiza la codificación/decodificación y la modulación/demodulación del canal así como la transmisión/recepción de la señal vía radio. El BTS también se refiere como el puerto de radio (RP) o la unidad transmisor-receptor de radio (RTU). Un BSC controla uno o más BTS y proporciona una interfaz para el intercambio local (conmutación) en la oficina central. Un papel importante del BSC es transcodificar los códigos de fuente usados en redes alámbricas y el interfaz aéreo. De los roles mencionados, un BSC a menudo se denomina la unidad de control de puerto de radio (RPCU) o la unidad del interfaz entre el transcodificador y la red (TNU).

Los sistemas de WLL no necesitan ofrecer servicios básicamente móviles, sin embargo algunos sistemas proporcionan servicios móviles limitados. Por ejemplo, no hay registro de la localización del “home” y “visitor” (HLR/VLR) en un sistema WLL por lo que su arquitectura total puede ser más simple que la de los sistemas móviles

## 1.3 TELEFONÍA MÓVIL

### 1.3.1 INTRODUCCIÓN<sup>12</sup>

La telefonía móvil básicamente está formada por dos grandes partes: Una red de comunicaciones o red de telefonía móvil y los terminales o teléfonos celulares que permiten el acceso a dicha red.

---

<sup>12</sup> <https://www.ibercom.com/soporte/index.php?>

El teléfono móvil o celular, es un dispositivo de comunicación electrónico con las mismas capacidades básicas de un teléfono de línea telefónica convencional. Además de ser portátil es inalámbrico al no requerir cables conductores para su conexión a la red telefónica.

### **1.3.2 FUNCIONAMIENTO<sup>13</sup>**

La red de telefonía móvil o celular consiste en un sistema telefónico en el que mediante la combinación de una red de estaciones transmisoras-receptoras de radio (estaciones base) y una serie de centrales telefónicas de conmutación, se posibilita la comunicación entre terminales telefónicos portátiles (teléfonos móviles) o entre terminales portátiles y teléfonos de la red fija tradicional.

El empleo de la palabra celular referido a la telefonía móvil, deriva del hecho de que las estaciones base, que enlazan vía radio los teléfonos móviles con los controladores de estaciones base, están dispuestas en forma de una malla, formando células o celdas (teóricamente como un panal de abejas). Así, cada estación base está situada en un nudo de estas células y tiene asignado un grupo de frecuencias de transmisión y recepción propio. Como el número de frecuencias es limitado, con esta disposición es posible reutilizar las mismas frecuencias en otras células, siempre que no sean adyacentes, para evitar interferencia entre ellas.

Las antenas están conectadas a una estación base que recoge todas las comunicaciones que reciban las antenas de la célula de cobertura en la que nos encontramos. La estación base toma los datos que emitimos desde nuestro teléfono móvil y los procesa para establecer la comunicación con el teléfono al que queremos llamar. Esta información es introducida en la Red Telefónica Conmutada o red convencional de telefonía, que procede a la conexión con las centrales digitales o analógicas asociadas al teléfono que recibe la llamada.

En el caso de que el teléfono receptor sea móvil, la red telefónica alertará a la estación base de la célula en la que se encuentre el móvil con el que queremos

---

<sup>13</sup> <http://ec.kalipedia.com/informatica/tema/realiza-comunicacion-mediante-telefono>

comunicar, procediendo a la emisión de la señal de radiofrecuencia a través de sus antenas, que captará el teléfono móvil que hemos marcado.

En el caso de que el receptor sea un teléfono fijo, la llamada es realizada directamente a través de la red telefónica convencional.



**Figura 1-11.** Diagrama del funcionamiento de la telefonía móvil<sup>14</sup>

Básicamente existen dos tipos de redes de telefonía móvil:

- Red de telefonía móvil analógica (TMA). Como su propio nombre indica, en esta red la comunicación se realiza mediante señales vocales analógicas tanto en el tramo radioeléctrico como en el terrestre. En su primera versión funcionó en la banda radioeléctrica de los 450 MHz, trabajando posteriormente en la banda de los 900 MHz.
- Red de telefonía móvil digital. En esta red la comunicación se realiza mediante señales digitales, lo que permite optimizar tanto el aprovechamiento de las bandas de radiofrecuencia como la calidad de transmisión. Su exponente más significativo en el ámbito público es el estándar GSM y su tercera generación, UMTS.

<sup>14</sup> <http://www.monografias.com/trabajos34/telefonía-celular/telefonía-celular>

Funciona en las bandas de 850/900 y 1800/1900 MHz. En 2004 llegó a los 1000 millones de usuarios. Hay otro estándar digital, presente en América y Asia, denominado CDMA. En el ámbito privado y de servicios de emergencias como policía, bomberos y servicios de ambulancias se utilizan los estándares Tetrapol y Terrestrial Trunked Radio (TETRA) en diferentes bandas de frecuencia.

## **1.4 EL MICROCONTROLADOR**

### **1.4.1 INTRODUCCIÓN<sup>15</sup>**

Un microcontrolador es un circuito integrado programable el cual contiene todos los componentes de un computador. Es un computador completo de limitadas prestaciones, que está contenido en un único chip.

Se emplea para controlar el funcionamiento de una única tarea y gracias a su reducido tamaño suele incorporarse en el propio dispositivo que controla.

Un microcontrolador normalmente dispone de una memoria pequeña, en la que se almacena un solo programa. Las líneas de entrada y salida se conectan con sensores y actuadores al dispositivo físico que controlan. Una vez programado el microcontrolador sólo sirve para atender la tarea para la que ha sido programado.

Las ventajas de los microcontroladores son:

- Aumento de servicios y utilidades para el usuario.
- Aumento de la fiabilidad.
- Reducción de tamaño en el producto acabado.
- Mayor flexibilidad.

---

<sup>15</sup> Ing. COSTALES, Alcívar, Apuntes de Control con Microcontroladores.

## 1.4.2 DIFERENCIA ENTRE MICROCONTROLADOR Y MICROPROCESADOR<sup>16</sup>

Ciertamente la gran mayoría de nosotros hemos escuchado hablar acerca de microprocesadores y microcontroladores y tal vez no sepamos cual es la diferencia entre uno y otro, o peor aún, nos podemos confundir al emplear cualquiera de las dos palabras.

Existe mucha información en internet acerca de microprocesadores y microcontroladores. Pero a veces la información es muy extensa y no muy fácil de comprender.

### 1.4.2.1 El Procesador

Definamos a un microprocesador o procesador como un elemento de hardware que se encarga como su nombre mismo lo dice procesar datos a partir de ciertas instrucciones de entrada arrojando cierto resultado, todo esto en base a su set de instrucciones que incorpora.



Figura 1-12. Microprocesador<sup>17</sup>

Un microprocesador por si solo sería prácticamente inservible, para que funcione necesita estar conectado mediante diferentes buses a otros elementos como la memoria RAM, dispositivos de entrada y salida (E/S) y otro tipo de memoria especial llamada EEPROM que almacena los datos de forma permanente aun sin energía eléctrica.

---

<sup>16</sup> <http://www.monografias.com/trabajos12/microco/microco.shtml>

<sup>17</sup> <http://www.smartsystemselectronics.com.mx/computacion/39-hardware/56-diferencias-entre-microprocesador-y-microcontrolador.html>



La mayoría de los procesadores de las PCs personales suelen utilizar la arquitectura Von Neuman. En esta arquitectura el procesador comparte la memoria RAM para almacenar las instrucciones del programa y los datos.

Debemos comprender que en una PC, cuando se carga un programa en memoria, a éste se le asigna un espacio de direcciones de la memoria que se divide en segmentos, de los cuales típicamente tendremos los siguientes: código (programa), datos y pila. Es por ello que podemos hablar de la memoria como un todo, aunque existan distintos dispositivos físicos en el sistema (HDD, RAM, CD, FLASH).

Utilizar la arquitectura tiene ventajas y desventajas, una de ellas es que al utilizar la misma memoria para almacenar los datos de programa y las instrucciones, solo se utilizan tres buses (de control, de datos y de direcciones) de estar separadas utilizaríamos otros tres buses mas, que con respecto al diseño seria más costosa.

Como desventaja principal de esta arquitectura encontramos que el procesador no puede acceder a los datos de programa y a las instrucciones al mismo tiempo, sino que debe almacenar una de ellos en los registros del procesador y volver a consultar las instrucciones para ejecutar, esto se traduce en pérdida de tiempo ya que el procesador tendrá que acceder dos veces a la memoria para poder ejecutar una sola instrucción.

También existe otro tipo de arquitectura llamada Harvard que es utilizada comúnmente por supercomputadoras o Mainframes. Esta define que la memoria de datos de programa y las instrucciones deben estar separadas. Con esto se tienen que crear otros tres buses a la otra memoria que incorporamos. Pero vale la pena ya que podemos procesar la información mas rápido, pues se puede acceder a la memoria de datos y a las instrucciones al mismo tiempo.

#### **1.4.2.2 Set de instrucciones del procesador**

Define las operaciones básicas que puede realizar el procesador, que conjugadas y organizadas forman lo que conocemos como software. El conjunto de instrucciones vienen siendo como las letras del alfabeto, el elemento básico del

lenguaje, que organizadas adecuadamente permiten escribir palabras, oraciones y cuanto programa se le ocurra.

Existen dos tipos de set de instrucciones: CISC y RISC

Las instrucciones CISC (Complex Instruction Set Computer, Computadora de Conjunto de Instrucciones Complejo) se caracterizan por tener un repertorio amplio de instrucciones, que pueden hacer cálculos muy complejos. Pero también se complica su uso.

El tipo RISC (Reduced Instruction Set Computer, Computadora con Conjunto de Instrucciones Reducido) se distingue por sus pocas instrucciones, y tamaño un tamaño fijo por instrucción además permite el paralelismo en la ejecución.

### 1.4.2.3 El Microcontrolador

Fue necesario definir algunas de las características principales de un procesador porque conociendo lo anterior definiremos al Microcontrolador.

Un Microcontrolador en realidad no es otra cosa más que una mini computadora embebida dentro de un solo encapsulado del tamaño de un chip. Este cuenta con procesador, memoria RAM, Memoria ROM y periféricos de entrada y salida.

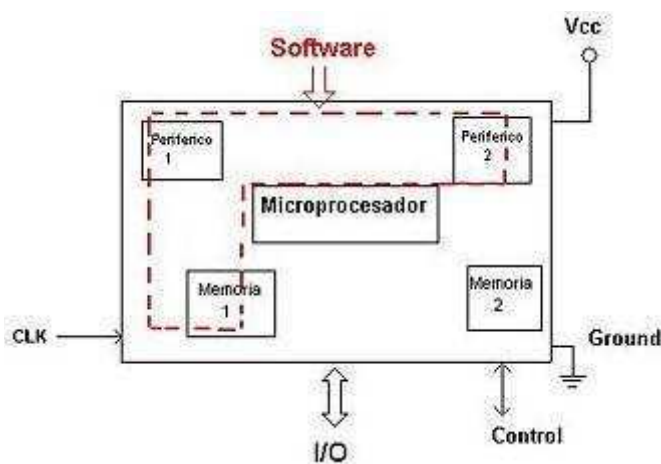


Figura 1-13. Estructura de un microcontrolador<sup>18</sup>

<sup>18</sup> <http://www.smartsystemselectronics.com.mx/computacion/39-hardware/56-diferencias-entre-microprocesador-y-microcontrolador.html>

Técnicamente un Microcontrolador lleva dentro un Microprocesador. Este a su vez cuenta con buses de datos, control y dirección hacia la memoria y hacia los periféricos o puertos.

La arquitectura de estos dispositivos es la Harvard quiere decir que poseen 2 memorias una memoria RAM que almacena los datos y otra de tipo EPROM o EEPROM que almacena el programa. Al usar esta arquitectura tiene la ventaja de poder usar simultáneamente los dos tipos de memoria y en lugar de ser una desventaja como lo era con una PC normal puesto que todo se encuentra en un solo encapsulado puede implementarse fácilmente.

El set de instrucciones es del tipo RISC las cuales ya vimos anteriormente. El lenguaje nativo de estos dispositivos es el lenguaje ensamblador. Aunque en la actualidad hay muchas empresas que se dedican a desarrollar compiladores que funcionan con la sintaxis de C o Basic que después traducirán a lenguaje máquina para programar el Microcontrolador.

Existen muchos fabricantes como Intel, Motorola, Texas Instruments, etc. Pero los más conocidos son los PICs de Microchip y los AVR de Atmel.

Actualmente a los microcontroladores se les puede hallar en muchísimos electrodomésticos como lavadoras, hornos de micro ondas, DVDs etc. Por su bajo precio, fácil programación y la extensa información que circula en internet estos se han vuelto muy utilizados en la electrónica y ramas afines.

### **1.4.3 RECURSOS COMUNES A TODOS LOS MICROCONTROLADORES**

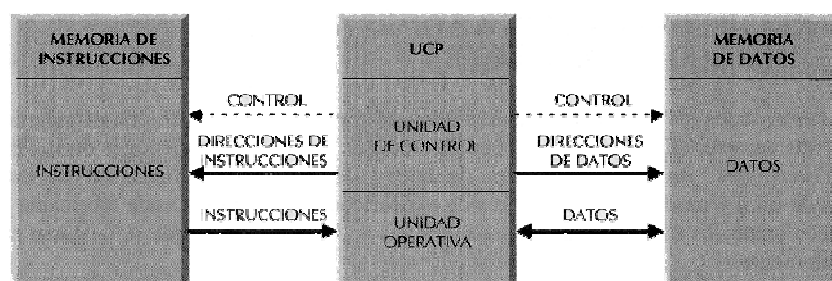
Al estar todos los microcontroladores integrados en un chip, su estructura fundamental y sus características básicas son muy parecidas. Todos deben disponer de los bloques esenciales Procesador, memoria de datos y de instrucciones, líneas de E/S, oscilador de reloj y módulos controladores de periféricos. Sin embargo, cada fabricante intenta enfatizar los recursos más idóneos para las aplicaciones a las que se destinan preferentemente.

En este apartado se hace un recorrido de todos los recursos que se hallan en todos los microcontroladores describiendo las diversas alternativas y opciones que pueden encontrarse según el modelo seleccionado.

#### 1.4.3.1 Arquitectura básica

Aunque inicialmente todos los microcontroladores adoptaron la arquitectura clásica de von Neumann, en el presente se impone la arquitectura Harvard. La arquitectura de von Neumann se caracteriza por disponer de una sola memoria principal donde se almacenan datos e instrucciones de forma indistinta. A dicha memoria se accede a través de un sistema de buses único (direcciones, datos y control).

La arquitectura Harvard dispone de dos memorias independientes una, que contiene sólo instrucciones y otra, sólo datos. Ambas disponen de sus respectivos sistemas de buses de acceso y es posible realizar operaciones de acceso (lectura o escritura) simultáneamente en ambas memorias. Los microcontroladores PIC responden a la arquitectura Harvard. Figura 1.3.



**Figura 1-14.** La arquitectura Harvard dispone de dos memorias independientes para datos y para instrucciones, permitiendo accesos simultáneos.<sup>19</sup>

#### 1.4.3.2 El procesador o UCP

Es el elemento más importante del microcontrolador y determina sus principales características, tanto a nivel hardware como software.

Se encarga de direccionar la memoria de instrucciones, recibir el código OP de la instrucción en curso, su decodificación y la ejecución de la operación que implica

<sup>19</sup> <http://www.monografias.com/trabajos12/microco/microco.shtml#DIFER>

la instrucción, así como la búsqueda de los operandos y el almacenamiento del resultado.

Existen tres orientaciones en cuanto a la arquitectura y funcionalidad de los procesadores actuales.

**CISC:** Un gran número de procesadores usados en los microcontroladores están basados en la filosofía CISC (Computadores de Juego de Instrucciones Complejo). Disponen de más de 80 instrucciones máquina en su repertorio, algunas de las cuales son muy sofisticadas y potentes, requiriendo muchos ciclos para su ejecución.

Una ventaja de los procesadores CISC es que ofrecen al programador instrucciones complejas que actúan como macros.

**RISC:** Tanto la industria de los computadores comerciales como la de los microcontroladores están decantándose hacia la filosofía RISC (Computadores de Juego de Instrucciones Reducido). En estos procesadores el repertorio de instrucciones máquina es muy reducido y las instrucciones son simples y, generalmente, se ejecutan en un ciclo.

La sencillez y rapidez de las instrucciones permiten optimizar el hardware y el software del procesador.

**SISC:** En los microcontroladores destinados a aplicaciones muy concretas, el juego de instrucciones, además de ser reducido, es "específico", o sea, las instrucciones se adaptan a las necesidades de la aplicación prevista. Esta filosofía se ha bautizado con el nombre de SISC (Computadores de Juego de Instrucciones Específico).

### 1.4.3.3 Memoria

En los microcontroladores la memoria de instrucciones y datos está integrada en el propio chip. Una parte debe ser no volátil, tipo ROM, y se destina a contener el programa de instrucciones que gobierna la aplicación. Otra parte de memoria será tipo RAM, volátil, y se destina a guardar las variables y los datos.

Hay dos peculiaridades que diferencian a los microcontroladores de los computadores personales:

No existen sistemas de almacenamiento masivo como disco duro o disquetes.

Como el microcontrolador sólo se destina a una tarea en la memoria ROM, sólo hay que almacenar un único programa de trabajo.

La RAM en estos dispositivos es de poca capacidad pues sólo debe contener las variables y los cambios de información que se produzcan en el transcurso del programa. Por otra parte, como sólo existe un programa activo, no se requiere guardar una copia del mismo en la RAM pues se ejecuta directamente desde la ROM.

Los usuarios de computadores personales están habituados a manejar Megabytes de memoria, pero, los diseñadores con microcontroladores trabajan con capacidades de ROM comprendidas entre 512 bytes y 8 kbytes y de RAM comprendidas entre 20 y 512 bytes.

Según el tipo de memoria ROM que dispongan los microcontroladores, la aplicación y utilización de los mismos es diferente. Se describen las cinco versiones de memoria no volátil que se pueden encontrar en los microcontroladores del mercado.

#### *1.4.3.3.1 ROM con máscara*

Es una memoria no volátil de sólo lectura cuyo contenido se graba durante la fabricación del chip. El elevado coste del diseño de la máscara sólo hace aconsejable el empleo de los microcontroladores con este tipo de memoria cuando se precisan cantidades superiores a varios miles de unidades.

#### *1.4.3.3.2 OTP*

El microcontrolador contiene una memoria no volátil de sólo lectura "programable una sola vez" por el usuario. OTP (One Time Programmable). Es el usuario quien puede escribir el programa en el chip mediante un sencillo grabador controlado por un programa desde un PC.

La versión OTP es recomendable cuando es muy corto el ciclo de diseño del producto, o bien, en la construcción de prototipos y series muy pequeñas.

Tanto en este tipo de memoria como en la EPROM, se suele usar la encriptación mediante fusibles para proteger el código contenido.

#### *1.4.3.3.3 EPROM*

Los microcontroladores que disponen de memoria EPROM (Erasable Programmable Read Only Memory) pueden borrarse y grabarse muchas veces. La grabación se realiza, como en el caso de los OTP, con un grabador gobernado desde un PC. Si, posteriormente, se desea borrar el contenido, disponen de una ventana de cristal en su superficie por la que se somete a la EPROM a rayos ultravioleta durante varios minutos. Las cápsulas son de material cerámico y son más caros que los microcontroladores con memoria OTP que están hechos con material plástico.

#### *1.4.3.3.4 EEPROM*

Se trata de memorias de sólo lectura, programables y borrables eléctricamente EEPROM (Electrical Erasable Programmable Read Only Memory). Tanto la programación como el borrado, se realizan eléctricamente desde el propio grabador y bajo el control programado de un PC. Es muy cómoda y rápida la operación de grabado y la de borrado. No disponen de ventana de cristal en la superficie.

Los microcontroladores dotados de memoria EEPROM una vez instalados en el circuito, pueden grabarse y borrarse cuantas veces se quiera sin ser retirados de dicho circuito. Para ello se usan "grabadores en circuito" que confieren una gran flexibilidad y rapidez a la hora de realizar modificaciones en el programa de trabajo.

El número de veces que puede grabarse y borrarse una memoria EEPROM es finito, por lo que no es recomendable una reprogramación continua. Son muy idóneos para la enseñanza y la Ingeniería de diseño.

Se va extendiendo en los fabricantes la tendencia de incluir una pequeña zona de memoria EEPROM en los circuitos programables para guardar y modificar cómodamente una serie de parámetros que adecuan el dispositivo a las condiciones del entorno. Este tipo de memoria es relativamente lenta.

#### *1.4.3.3.5 FLASH*

Se trata de una memoria no volátil, de bajo consumo, que se puede escribir y borrar. Funciona como una ROM y una RAM pero consume menos y es más pequeña.

A diferencia de la ROM, la memoria FLASH es programable en el circuito. Es más rápida y de mayor densidad que la EEPROM.

La alternativa FLASH está recomendada frente a la EEPROM cuando se precisa gran cantidad de memoria de programa no volátil. Es más veloz y tolera más ciclos de escritura/borrado.

Las memorias EEPROM y FLASH son muy útiles al permitir que los microcontroladores que las incorporan puedan ser reprogramados "en circuito", es decir, sin tener que sacar el circuito integrado de la tarjeta. Así, un dispositivo con este tipo de memoria incorporado al control del motor de un automóvil permite que pueda modificarse el programa durante la rutina de mantenimiento periódico, compensando los desgastes y otros factores tales como la compresión, la instalación de nuevas piezas, etc. La reprogramación del microcontrolador puede convertirse en una labor rutinaria dentro de la puesta a punto.

#### **1.4.3.4 Puertas de Entrada y Salida**

La principal utilidad de las patitas que posee la cápsula que contiene un microcontrolador es soportar las líneas de E/S que comunican al computador interno con los periféricos exteriores.

Según los controladores de periféricos que posea cada modelo de microcontrolador, las líneas de E/S se destinan a proporcionar el soporte a las señales de entrada, salida y control.



#### **1.4.3.5 Reloj principal**

Todos los microcontroladores disponen de un circuito oscilador que genera una onda cuadrada de alta frecuencia, que configura los impulsos de reloj usados en la sincronización de todas las operaciones del sistema.

Generalmente, el circuito de reloj está incorporado en el microcontrolador y sólo se necesitan unos pocos componentes exteriores para seleccionar y estabilizar la frecuencia de trabajo. Dichos componentes suelen consistir en un cristal de cuarzo junto a elementos pasivos o bien un resonador cerámico o una red R-C.

Aumentar la frecuencia de reloj supone disminuir el tiempo en que se ejecutan las instrucciones pero lleva aparejado un incremento del consumo de energía.

#### **1.4.4 RECURSOS ESPECIALES**

Cada fabricante oferta numerosas versiones de una arquitectura básica de microcontrolador. En algunas amplía las capacidades de las memorias, en otras incorpora nuevos recursos, en otras reduce las prestaciones al mínimo para aplicaciones muy simples, etc. La labor del diseñador es encontrar el modelo mínimo que satisfaga todos los requerimientos de su aplicación. De esta forma, minimizará el coste, el hardware y el software.

Los principales recursos específicos que incorporan los microcontroladores son:

- Temporizadores o "Timers".
- Perro guardián o "Watchdog".
- Protección ante fallo de alimentación o "Brownout".
- Estado de reposo o de bajo consumo.
- Conversor A/D.
- Conversor D/A.
- Comparador analógico.
- Modulador de anchura de impulsos o PWM.
- Puertas de E/S digitales.

- Puertas de comunicación.

#### **1.4.4.1 Temporizadores o "Timers"**

Se emplean para controlar periodos de tiempo (temporizadores) y para llevar la cuenta de acontecimientos que suceden en el exterior (contadores).

Para la medida de tiempos se carga un registro con el valor adecuado y a continuación dicho valor se va aumentando o disminuyendo al ritmo de los impulsos de reloj o algún múltiplo hasta que se desborde y llegue a 0, momento en el que se produce un aviso.

Cuando se desean contar acontecimientos que se materializan por cambios de nivel o flancos en alguna de las patitas del microcontrolador, el mencionado registro se va aumentando o disminuyendo al ritmo de dichos impulsos.

#### **1.4.4.2 Perro guardián o "Watchdog"**

Cuando el computador personal se bloquea por un fallo del software u otra causa, se pulsa el botón del reset y se reinicializa el sistema. Pero un microcontrolador funciona sin el control de un supervisor y de forma continuada las 24 horas del día. El Perro guardián consiste en un temporizador que, cuando se desborda y pasa por 0, provoca un reset automáticamente en el sistema.

Se debe diseñar el programa de trabajo que controla la tarea de forma que refresque o inicialice al Perro guardián antes de que provoque el reset. Si falla el programa o se bloquea, no se refrescará al Perro guardián y, al completar su temporización, "ladrará y ladrará" hasta provocar el reset.

#### **1.4.4.3 Protección ante fallo de alimentación o "Brownout"**

Se trata de un circuito que resetea al microcontrolador cuando el voltaje de alimentación (VDD) es inferior a un voltaje mínimo ("brownout"). Mientras el voltaje de alimentación sea inferior al de brownout el dispositivo se mantiene reseteado, comenzando a funcionar normalmente cuando sobrepasa dicho valor.

#### **1.4.4.4 Estado de reposo o de bajo consumo**

Son abundantes las situaciones reales de trabajo en que el microcontrolador debe esperar, sin hacer nada, a que se produzca algún acontecimiento externo que le ponga de nuevo en funcionamiento. Para ahorrar energía, (factor clave en los aparatos portátiles), los microcontroladores disponen de una instrucción especial (SLEEP en los PIC), que les pasa al estado de reposo o de bajo consumo, en el cual los requerimientos de potencia son mínimos. En dicho estado se detiene el reloj principal y se "congelan" sus circuitos asociados, quedando sumido en un profundo "sueño" el microcontrolador. Al activarse una interrupción ocasionada por el acontecimiento esperado, el microcontrolador se despierta y reanuda su trabajo.

#### **1.4.4.5 Conversor A/D (CAD)**

Los microcontroladores que incorporan un Conversor A/D (Analógico/Digital) pueden procesar señales analógicas, tan abundantes en las aplicaciones. Suelen disponer de un multiplexor que permite aplicar a la entrada del CAD diversas señales analógicas desde las patitas del circuito integrado.

#### **1.4.4.6 Conversor D/A (CDA)**

Transforma los datos digitales obtenidos del procesamiento del computador en su correspondiente señal analógica que saca al exterior por una de las patitas de la cápsula. Existen muchos efectores que trabajan con señales analógicas.

#### **1.4.4.7 Comparador analógico**

Algunos modelos de microcontroladores disponen internamente de un Amplificador Operacional que actúa como comparador entre una señal fija de referencia y otra variable que se aplica por una de las patitas de la cápsula. La salida del comparador proporciona un nivel lógico 1 ó 0 según una señal sea mayor o menor que la otra.

También hay modelos de microcontroladores con un módulo de tensión de referencia que proporciona diversas tensiones de referencia que se pueden aplicar en los comparadores.

#### **1.4.4.8 Modulador de anchura de impulsos o PWM**

Son circuitos que proporcionan en su salida impulsos de anchura variable, que se ofrecen al exterior a través de las patitas del encapsulado.

#### **1.4.4.9 Puertos de E/S digitales**

Todos los microcontroladores destinan algunas de sus patitas a soportar líneas de E/S digitales. Por lo general, estas líneas se agrupan de ocho en ocho formando Puertos.

Las líneas digitales de los Puertos pueden configurarse como Entrada o como Salida cargando un 1 ó un 0 en el bit correspondiente de un registro destinado a su configuración.

#### **1.4.4.10 Puertos de comunicación**

Con objeto de dotar al microcontrolador de la posibilidad de comunicarse con otros dispositivos externos, otros buses de microprocesadores, buses de sistemas, buses de redes y poder adaptarlos con otros elementos bajo otras normas y protocolos. Algunos modelos disponen de recursos que permiten directamente esta tarea, entre los que destacan:

UART, adaptador de comunicación serie asíncrona.

USART, adaptador de comunicación serie síncrona y asíncrona

Puerta paralela esclava para poder conectarse con los buses de otros microprocesadores.

USB (Universal Serial Bus), que es un moderno bus serie para los PC.

Bus I2C, que es un interfaz serie de dos hilos desarrollado por Philips.

CAN (Controller Area Network), para permitir la adaptación con redes de conexionado multiplexado desarrollado conjuntamente por Bosch e Intel para el cableado de dispositivos en automóviles. En EE.UU. se usa el J1850.

### 1.4.5 COMO SABER QUE MICROCONTROLADOR UTILIZAR<sup>20</sup>

A la hora de escoger el microcontrolador a emplear en un diseño concreto hay que tener en cuenta multitud de factores, como la documentación y herramientas de desarrollo disponibles y su precio, la cantidad de fabricantes que lo producen y por supuesto las características del microcontrolador (tipo de memoria de programa, número de temporizadores, interrupciones, etc.):

**Costes.** Como es lógico, los fabricantes de microcontroladores compiten duramente para vender sus productos. Y no les va demasiado mal ya que sin hacer demasiado ruido venden 10 veces más microcontroladores que microprocesadores.

Para que nos hagamos una idea, para el fabricante que usa el microcontrolador en su producto una diferencia de precio en el microcontrolador de algunas pesetas es importante (el consumidor deberá pagar además el coste del empaquetado, el de los otros componentes, el diseño del hardware y el desarrollo del software). Si el fabricante desea reducir costes debe tener en cuenta las herramientas de apoyo con que va a contar: emuladores, simuladores, ensambladores, compiladores, etc. Es habitual que muchos de ellos siempre se decanten por microcontroladores pertenecientes a una única familia.

**Aplicación.** Antes de seleccionar un microcontrolador es imprescindible analizar los requisitos de la aplicación:

- **Procesamiento de datos:** puede ser necesario que el microcontrolador realice cálculos críticos en un tiempo limitado. En ese caso debemos asegurarnos de seleccionar un dispositivo suficientemente rápido para ello. Por otro lado, habrá que tener en cuenta la precisión de los datos a manejar: si no es suficiente con un microcontrolador de 8 bits, puede ser necesario acudir a microcontroladores de 16 ó 32 bits, o incluso a hardware de coma flotante. Una alternativa más barata y quizá suficiente es usar librerías para manejar los datos de alta precisión. -

- **Entrada/Salida:** para determinar las necesidades de Entrada/Salida del sistema es conveniente dibujar un diagrama de bloques del mismo, de tal forma que sea

<sup>20</sup> <http://www.monografias.com/trabajo12/microco/microco.shtml>

sencillo identificar la cantidad y tipo de señales a controlar. Una vez realizado este análisis puede ser necesario añadir periféricos hardware externos o cambiar a otro microcontrolador más adecuado a ese sistema.

- **Consumo:** algunos productos que incorporan microcontroladores están alimentados con baterías y su funcionamiento puede ser tan vital como activar una alarma antirrobo. Lo más conveniente en un caso como éste puede ser que el microcontrolador esté en estado de bajo consumo pero que despierte ante la activación de una señal (una interrupción) y ejecute el programa adecuado para procesarla.
- **Memoria:** para detectar las necesidades de memoria de nuestra aplicación debemos separarla en memoria volátil (RAM), memoria no volátil (ROM, EPROM, etc.) y memoria no volátil modificable (EEPROM). Este último tipo de memoria puede ser útil para incluir información específica de la aplicación como un número de serie o parámetros de calibración.

El tipo de memoria a emplear vendrá determinado por el volumen de ventas previsto del producto: de menor a mayor volumen será conveniente emplear EPROM, OTP y ROM. En cuanto a la cantidad de memoria necesaria puede ser imprescindible realizar una versión preliminar, aunque sea en pseudo-código, de la aplicación y a partir de ella hacer una estimación de cuánta memoria volátil y no volátil es necesaria y si es conveniente disponer de memoria no volátil modificable.

- **Ancho de palabra:** el criterio de diseño debe ser seleccionar el microcontrolador de menor ancho de palabra que satisfaga los requerimientos de la aplicación. Usar un microcontrolador de 4 bits supondrá una reducción en los costes importante, mientras que uno de 8 bits puede ser el más adecuado si el ancho de los datos es de un byte. Los microcontroladores de 16 y 32 bits, debido a su elevado coste, deben reservarse para aplicaciones que requieran sus altas prestaciones (Entrada/Salida potente o espacio de direccionamiento muy elevado).

- **Diseño de la placa:** la selección de un microcontrolador concreto condicionará el diseño de la placa de circuitos. Debe tenerse en cuenta que quizá usar un microcontrolador barato encarezca el resto de componentes del diseño.

Los microcontroladores más populares se encuentran, sin duda, entre las mejores elecciones:

8048 (Intel). Es el padre de los microcontroladores actuales, el primero de todos. Su precio, disponibilidad y herramientas de desarrollo hacen que todavía sea muy popular.

8051 (Intel y otros). Es sin duda el microcontrolador más popular. Fácil de programar, pero potente. Está bien documentado y posee cientos de variantes e incontables herramientas de desarrollo.

80186, 80188 y 80386 EX (Intel). Versiones en microcontrolador de los populares microprocesadores 8086 y 8088. Su principal ventaja es que permiten aprovechar las herramientas de desarrollo para PC.

68HC11 (Motorola y Toshiba). Es un microcontrolador de 8 bits potente y popular con gran cantidad de variantes.

683xx (Motorola). Surgido a partir de la popular familia 68k, a la que se incorporan algunos periféricos. Son microcontroladores de altísimas prestaciones.

PIC (MicroChip). Familia de microcontroladores que gana popularidad día a día. Fueron los primeros microcontroladores RISC.

Es preciso resaltar en este punto que existen innumerables familias de microcontroladores, cada una de las cuales posee un gran número de variantes.

## 1.5 MICROCONTROLADOR PIC 16F877A<sup>21</sup>

En este proyecto se utilizó el PIC 16F877A. Este microcontrolador es fabricado por Microchip Technology Inc. familia a la cual se le denomina PIC. El modelo 16F877A posee varias características que hacen a este microcontrolador un

---

<sup>21</sup> <http://www.monografias.com/trabajos18/descripción-pic>

dispositivo muy versátil, eficiente y práctico para ser empleado en la aplicación que en el capítulo siguiente será detallada.

Algunas de estas características se muestran a continuación:

- Soporta modo de comunicación serial, posee dos pines para ello.
- Amplia memoria para datos y programa.
- Memoria reprogramable: La memoria en este PIC es la que se denomina FLASH; este tipo de memoria se puede borrar electrónicamente (esto corresponde a la "F" en el modelo).
- Set de instrucciones reducido (tipo RISC), pero con las instrucciones necesarias para facilitar su manejo.

### 1.5.1 CARACTERÍSTICAS<sup>22</sup>

#### **CPU:**

- Tecnología RISC
- Sólo 35 instrucciones que aprender
- Todas las instrucciones se ejecutan en un ciclo de reloj, excepto los saltos que requieren dos.
- Frecuencia de operación de 0 a 20 MHz (200 nseg de ciclo de instrucción)
- Opciones de selección del oscilador

#### **Memoria:**

- Hasta 8k x 14 bits de memoria Flash de programa
- Hasta 368 bytes de memoria de datos (RAM)
- Hasta 256 bytes de memoria de datos EEPROM
- Lectura/escritura de la CPU a la memoria flash de programa
- Protección programable de código
- Stack de hardware de 8 niveles

#### **Reset e interrupciones:**

- Hasta 14 fuentes de interrupción
- Reset de encendido (POR)

---

<sup>22</sup> <http://www.scribd.com/doc/101172/pic16f877-en-espanol1>



- Timer de encendido (PWRT)
- Timer de arranque del oscilador (OST)
- Sistema de vigilancia Watchdog timer.

#### Otros:

- Modo SLEEP de bajo consumo de energía
- Programación y depuración serie “In-Circuit” (ICSP) a través de dos patitas
- Rango de voltaje de operación de 2.0 a 5.5 volts
- Alta disipación de corriente de la fuente: 25mA
- Rangos de temperatura: Comercial, Industrial y Extendido
- Bajo consumo de potencia:
  - Menos de 0.6mA a 3V, 4 Mhz
  - 20  $\mu$ A a 3V, 32 KHz
  - menos de 1 $\mu$ A corriente de standby (modo SLEEP).

#### Periféricos:

Tabla 1-1. Características periféricas.<sup>23</sup>

Periférico	PIC16F873 PIC16F876	PIC16F874 PIC16F877	Características
<b>3 a 5 Puertos paralelos</b>	PortA,B,C	PortA, B,C,D,E	con líneas digitales programables individualmente
<b>3 Timers</b>	Timer0	Timer0	Contador/Temporizador de 8 bits con pre-escalador de 8 bits
	Timer1	Timer1	Contador/Temporizador de 16 bits con pre-escalador
	Timer2	Timer2	Contador/Temporizador de 8 bits con pre-escalador y post-escalador de 8 bits y registro de periodo
<b>2 módulos CCP</b>	Captura	Captura	16 bits, 1.5 nseg de resolución máxima
	Comparación	Comparación	16 bits, 200 nseg de resolución máxima
	PWM	PWM	10 bits
<b>1 Convertidor A/D</b>	AN0,...,AN4	AN0,...,AN7	de 10 bits, hasta 8 canales
<b>Puertos Serie</b>	SSP	SSP	Puerto Serie Síncrono
	USART/SCI	USART/SCI	Puerto Serie Universal
	ICSP	ICSP	Puerto serie para programación y depuración “in circuit”
<b>Puerto Paralelo Esclavo</b>	PSP	PSP	Puerto de 8 bits con líneas de protocolo

### 1.5.2 NOMENCLATURA<sup>24</sup>

En el nombre específico del microcontrolador pueden aparecer algunas siglas que dependen del rango de voltaje manejado y del tipo de memoria ROM incluida, como se muestra en la siguiente tabla:

<sup>23</sup> <http://www.scribd.com/doc/101172/pic16f877-en-espanol1>

<sup>24</sup> <http://www.apuntes20%intro20%pic.pdf.com>

**Tabla 1-2.** Nomenclatura según el tipo de memoria.

Tipo de memoria	Rango de voltaje	
	Estándar (4.5 a 6 volts)	Extendido (2.5 a 6 volts)
<b>EPROM, OTP</b>	PIC16CXXX	PIC16LCXXX
<b>ROM</b>	PIC16CRXXX	PIC16LCRXXX
<b>Flash</b>	PIC16FXXX	PIC16LFXXX

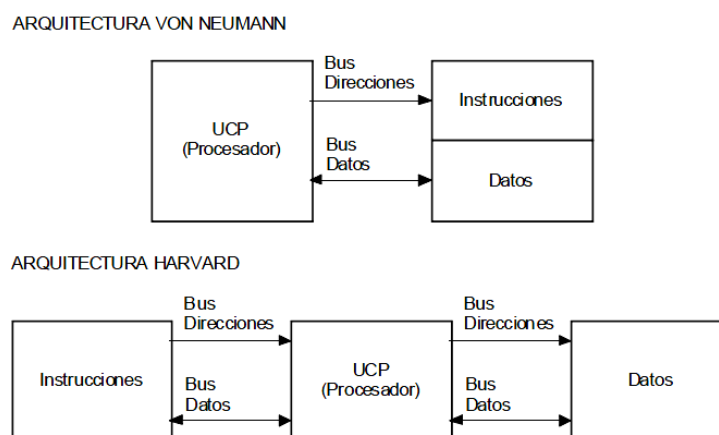
En la siguiente tabla se especifican los rangos de voltaje estándar y extendido manejados por los dispositivos.

**Tabla 1-3.** Nomenclatura según el rango de voltaje.

Rango de voltaje	EPROM		ROM		Flash	
Estándar	<b>C</b>	4.5 a 6v	<b>CR</b>	4.5 a 6v	<b>F</b>	4.5 a 6v
Extendido	<b>LC</b>	2.5 a 6v	<b>LCR</b>	2.5 a 6v	<b>LF</b>	2 a 6v

### 1.5.3 ARQUITECTURA<sup>25</sup>

Los PIC16F877A de Microchip pertenecen al tipo de procesador RICS que es un procesador de instrucciones reducidas, se caracteriza por que el número de instrucciones es pequeño y además casi todas se realizan en la misma cantidad de tiempo, por otro lado posee unidades que trabajan en paralelo conectadas por pipes o tuberías. Este tipo de procesador emplea una arquitectura Harvard lo que significa que trabaja las zonas de memoria de programa y datos en forma separada. En el siguiente diagrama se muestra la arquitectura Von Neuman frente a la Harvard:

**Figura 1-15.** Tipos de Arquitectura

<sup>25</sup> [http:// www.scribd.com/doc/2939885/pic16f877-PERALTA?autodown=pdf](http://www.scribd.com/doc/2939885/pic16f877-PERALTA?autodown=pdf)

En ambas arquitecturas observamos bloques de memoria, cada bloque tiene posiciones y cada posición un valor. Para recoger o dejar un valor en una determinada posición es necesario primero indicar cuál es la dirección a leer o escribir de la memoria, en consecuencia hay un grupo de líneas que nos permiten hacer esa función conocida como el bus de direcciones, también existe un bus de datos que son líneas paralelas por donde discurren los valores de cada dirección.

En el caso de la arquitectura Von Neuman podemos apreciar que existe un único bus de direcciones y de datos. Podemos apreciar como cada posición de memoria tiene una dirección, a su vez la memoria se divide en memoria de programa (conocida como ROM) y memoria de datos (conocida como RAM).

En el caso de la arquitectura Harvard existen dos bloques de memoria separados. Un bloque para instrucciones y otro para datos. Note como hay dos buses independientes de direcciones y el bus de instrucciones solo tiene una dirección, a diferencia del bus de datos que es de naturaleza bidireccional.

Todo esto sugiere que puede existir una dirección por ejemplo la 0. Entonces tenemos una instrucción en la posición 0 y también un dato en la 0. En el caso de la arquitectura Von Neumann esa dirección es de programa o de instrucción pero no de ambas.

La arquitectura Harvard mejora el ancho de banda por que el bus de datos es de 14 bits frente a los de 8 de un bus tradicional Von Neumann por tanto en una sola lectura puede llevar mayor cantidad de datos.

### **1.5.3.1 Arquitectura Interna**

Hemos señalado que el microcontrolador posee varios elementos en forma interna: el procesador, memoria de programa, memoria de datos, periféricos, contadores. Observemos el siguiente diagrama de bloques del PIC16F877A:

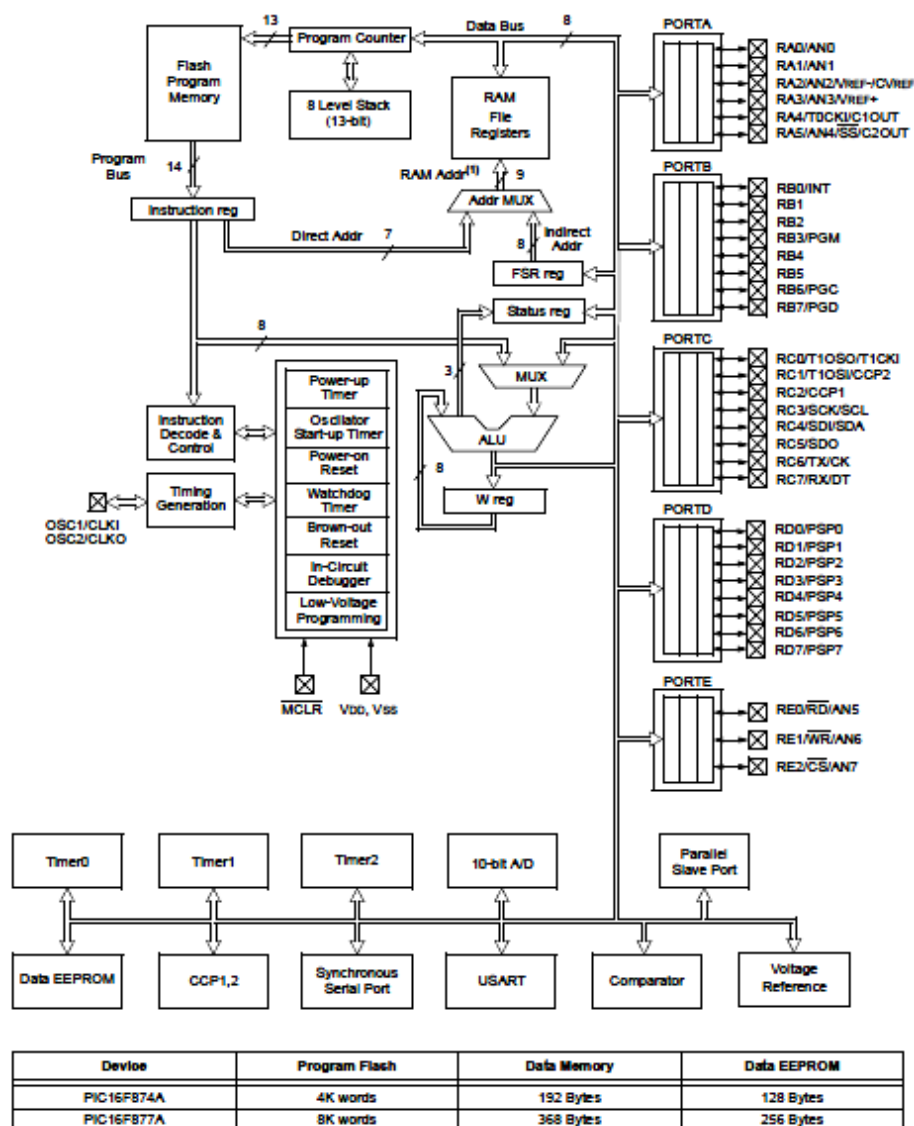


Figura 1-16. Diagrama de la Arquitectura interna del PIC 16F877A<sup>26</sup>

En el diagrama podemos identificar la memoria del Programa en la parte superior izquierda con 8K posiciones por 14 bits, esta también presenta la memoria de datos (RAM) de 368 posiciones por 8 bits. La memoria EEPROM 256 posiciones x 8 bits. El procesador propiamente dicho está formado por la ALU (unidad aritmética lógica) el registro de trabajo W. Tenemos los periféricos I/O Port A, B, C, D, E el TMR0 (temporizador contador de eventos), TMR1 y TMR2 entre otros módulos. También contamos con un registro de instrucción que se carga cada vez que la ALU solicita una nueva instrucción a procesar. En la parte intermedia encontramos algunos bloques como son el Status Reg. que es el registro de

<sup>26</sup> <http://www.datasheet.com>

estado encargado de anotar el estado actual del sistema, cada vez que se ejecuta una instrucción se llevan a cabo cambios dentro del microcontrolador como desborde, acarreo, etc. Cada uno de esos eventos está asociado a un bit de este registro. Existe un registro de vital importancia que se llama el Program Counter o contador de programa este registro indica la dirección de la instrucción a ejecutar. El registro en cuestión no es necesariamente secuencial, esto es no se incrementa necesariamente de uno en uno ya que puede darse el caso en el que salte dependiendo si hay una instrucción de bifurcación de por medio o puede haber alguna instrucción de llamada a función y/o procedimiento. También se observa el bloque de la pila, la función de la pila es ser un buffer temporal en el que se guarda el contador de programa cada vez que se suscita una llamada a un procedimiento y/o función (incluyendo interrupciones). Por tanto el nivel de anidamiento es de hasta 8 llamadas. También está presente el FSR reg. que es el registro que cumple una función similar a la del contador de programa direccionando en este caso la RAM, el FSR es un puntero a una dirección de la RAM. La aparición de multiplexores se debe a que los datos pueden tener diferentes fuentes.

Cuando programamos el microcontrolador debemos siempre tener en mente que es lo que él hace. Cuando lo prendemos asume un valor por defecto, el contador de programa asume la posición cero por tanto el microcontrolador toma la instrucción que se encuentra en esa posición en la memoria de programa y la ejecuta. Al momento de ejecutarla procede a informar si se ha llevado a cabo alguna operación en particular registrándola en el registro de estado (STATUS). Si la instrucción es de salto o bifurcación evaluará las condiciones para saber si continua o no con la siguiente instrucción, en caso que no sea así saltará a otra posición de memoria. En caso el programa haga un llamado a una función guardará en la pila el valor del contador de programa ejecutará la rutina y al momento que termina restituirá el valor correspondiente para seguir con la siguiente instrucción.

## 1.5.4 ORGANIZACIÓN DE LA MEMORIA<sup>27</sup>

Existen bloques de memoria dentro de un PIC16F87X. La memoria de programa y la memoria de datos tienen buses separados por lo que es posible el acceso a las mismas en forma concurrente. El tercer bloque de memoria es la memoria de datos EEPROM.

### 1.5.4.1 Organización de la memoria de programa

El dispositivo PIC16F87X tiene un program counter de 13-bit capaz de direccionar 8K x 14 direcciones de memoria. Los dispositivos PIC16F877/876 tienen 8K x 14 palabras de FLASH program memory y los dispositivos PIC16F873/874 tienen 4K x 14. El vector de RESET (por donde comienza a ejecutar el PIC) está en la dirección 0000h y el vector de interrupciones se encuentra en la dirección 0004h.

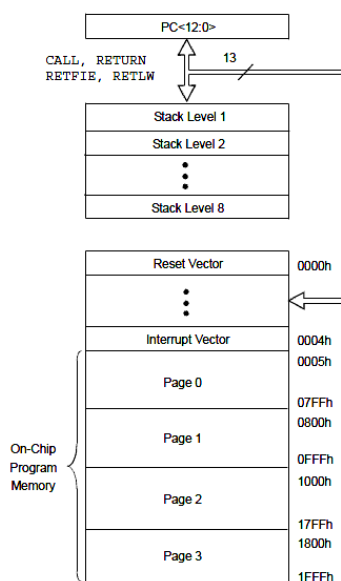


Figura 1-17. Mapa de la memoria de programa y STACK del PIC16F877A especiales.<sup>28</sup>

### 1.5.4.2 Organización de la memoria de datos

La memoria de datos está particionada en múltiples bancos que contienen los registros de propósito general (General Purpose Registers) y los registros de funciones especiales (Special Function Registers).

<sup>27</sup> <http://www.fing.edu.uy/inco/cursos/firmware/laboratorio/cartillaPIC16F877.pdf>

<sup>28</sup> <http://www.datasheet.com>

El bit RP1 (STATUS <6>) y RP0 (STATUS<5>) son los bits de selección de banco. Cada banco se extiende hasta 7Fh (128 bytes). Los lugares más bajos de cada banco están reservados para registros de funciones. Debajo de estos se encuentran los registros de propósito general, implementados como RAM estática. Todos los bancos implementados contienen registros de funciones especiales. Algunos de los registros de funciones especiales que son accedidos frecuentemente desde un banco pueden ser espejados en otro banco para reducir el código y lograr un acceso más rápido a los mismos.

### 1.5.4.3 Registros de funciones especiales

File Address		File Address		File Address		File Address	
Indirect addr. <sup>(*)</sup>	00h	Indirect addr. <sup>(*)</sup>	80h	Indirect addr. <sup>(*)</sup>	100h	Indirect addr. <sup>(*)</sup>	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h		105h		185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
PORTC	07h	TRISC	87h		107h		187h
PORTD <sup>(1)</sup>	08h	TRISD <sup>(1)</sup>	88h		108h		188h
PORTE <sup>(1)</sup>	09h	TRISE <sup>(1)</sup>	89h		109h		189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	EEDATA	10Ch	EECON1	18Ch
PIR2	0Dh	PIE2	8Dh	EEADR	10Dh	EECON2	18Dh
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reserved <sup>(2)</sup>	18Eh
TMR1H	0Fh		8Fh	EEADRH	10Fh	Reserved <sup>(2)</sup>	18Fh
T1CON	10h		90h		110h		190h
TMR2	11h	SSPCON2	91h				
T2CON	12h	PR2	92h				
SSPBUF	13h	SSPAD	93h				
SSPCON	14h	SSPSTAT	94h				
CCPR1L	15h		95h				
CCPR1H	16h		96h				
CCP1CON	17h		97h				
RCSTA	18h	TXSTA	98h				
TXREG	19h	SPBRG	99h				
RCREG	1Ah		9Ah				
CCPR2L	1Bh		9Bh				
CCPR2H	1Ch	CMCON	9Ch				
CCP2CON	1Dh	CVRCON	9Dh				
ADRESH	1Eh	ADRESL	9Eh				
ADCON0	1Fh	ADCON1	9Fh				
	20h		A0h		120h		1A0h
General Purpose Register 96 Bytes		General Purpose Register 96 Bytes		accesses 20h-7Fh		accesses A0h - FFh	
	7Fh		FFh		10Fh		1EFh
Bank 0		Bank 1		Bank 2		Bank 3	
					17Fh		1FFh

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

**Note 1:** These registers are not implemented on the PIC16F873A.  
**2:** These registers are reserved; maintain these registers clear.

Figura 1-18. Mapa del archivo de registros<sup>29</sup>

<sup>29</sup> <http://www.datasheet.com>

Los registros de funciones especiales son usados por la CPU y los módulos periféricos para controlar las operaciones del dispositivo. Los registros de funciones especiales pueden ser clasificados en dos conjuntos: core (CPU) y periféricos.

#### 1.5.4.3.1 Registro: STATUS

El registro STATUS contiene el estado de la aritmética de la ALU, es estado del RESET y el banco elegido para los datos de la memoria (SRAM). Puede ser el destino de cualquier instrucción como cualquier otro registro. Si el registro de estado es destino de una instrucción los bits Z, DC o C son afectados. Los bits TO y PD son sólo de lectura.

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

Tabla 1-4. Registro STATUS (Direcciones 03h, 83h, 103h, 183h)<sup>30</sup>

bit 7 **IRP**: Registro de selección de Banco.(usado para direccionamiento indirecto)

1 = Bank 2, 3 (100h - 1FFh)

0 = Bank 0, 1 (00h - FFh)

bit 6-5 **RP1:RP0**: Registro de selección de banco (usado para direccionamiento directo)

00 = Bank 0 (00h - 7Fh)

01 = Bank 1 (80h - FFh)

10 = Bank 2 (100h - 17Fh)

11 = Bank 3 (180h - 1FFh)

bit 4 **TO**: bit de Timeout

1 = Después de encender, una instrucción CLRWDT , o una instrucción SLEEP

0 = ocurrió un timeout de WDT

<sup>30</sup> <http://www.datasheet.com>



bit 3 **PD**: bit de Apagado

1 = Después de encendido o por una instrucción CLRWDT

0 = Por una ejecución de la instrucción SLEEP

bit 2 **Z**: Zero bit

1 = El resultado de una operación aritmética fue 0.

0 = El resultado de una operación aritmética no fue 0.

bit 1 **DC**: Dígito de acarreo (para instrucciones ADDWF, ADDLW, SUBLW, SUBWF instructions)

1 = Hubo acarreo del cuarto bit de orden bajo en el resultado.

0 = No hubo acarreo del cuarto bit de orden bajo en el resultado.

bit 0 **C**: bit de acarreo o préstamo (instrucciones ADDWF, ADDLW, SUBLW, SUBWF)

1 = Ocurrió acarreo en el bit más significativo del resultado.

0 = No ocurrió acarreo en el bit más significativo del resultado.

#### 1.5.4.3.2 Registro: INTCON

El registro INTCON es un registro legible y escribible que contiene las llaves para habilitar las fuentes de interrupción.

Nota: las banderas de interrupción se setearán independientemente de la forma en que esté seteado el INTCON, pero si el GIE no está habilitado la interrupción no se provocará.

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	TOIE	INTE	TBIE	TOIF	INTF	TBIF
bit 7							bit 0

Tabla 1-5. Registro INTCON (Direcciones 0Bh, 8Bh, 10Bh, 18Bh)<sup>31</sup>

<sup>31</sup> <http://www.datasheet.com>

bit 7 **GIE**: bit de interrupciones globales

1 = Habilita las interrupciones no enmascaradas.

0 = Deshabilita las interrupciones no enmascaradas.

bit 6 **PEIE**: bit de Interrupciones periféricas

1 = Habilita las interrupciones periféricas no enmascaradas.

0 = Deshabilita las interrupciones periféricas no enmascaradas.

bit 5 **TOIE**: bit de habilitación de interrupciones por overflow del timer 0.

1 = Habilita interrupciones por TMR0

0 = Deshabilita interrupciones por TMR0

bit 4 **INTE**: bit de habilitación de la interrupción externa en RB0

1 = Habilita interrupciones externas en RB0

0 = Deshabilita interrupciones externas en RB0

bit 3 **RBIE**: bit de habilitación de interrupciones por cambios en el puerto B.

1 = Habilita interrupciones por cambio en puertoB

0 = Deshabilita interrupciones por cambio en puertoB

bit 2 **TOIF**: bit bandera de interrupción por TMR0

1 = en el TMR0 ha ocurrido overflow (esta bandera debe volverse a 0 en software)

0 = en el TMR0 no ha ocurrido overflow.

bit 1 **INTF**: bit bandera de interrupción externa en RB0

1 = ha ocurrido una interrupción externa por RB0 (esta bandera debe volverse a 0 en software)

0 = no ha ocurrido una interrupción externa por RB0

bit 0 **RBIF**: bit bandera de interrupción por cambio en el puerto B

1 = Al menos una de las patas entre RB7:RB4 a cambiado de valor (esta bandera debe volverse a 0 en software)

0 = Ninguna de las patas entre RB7:RB4 ha cambiado de valor

### 1.5.5 DIAGRAMA DE PINES Y FUNCIONES

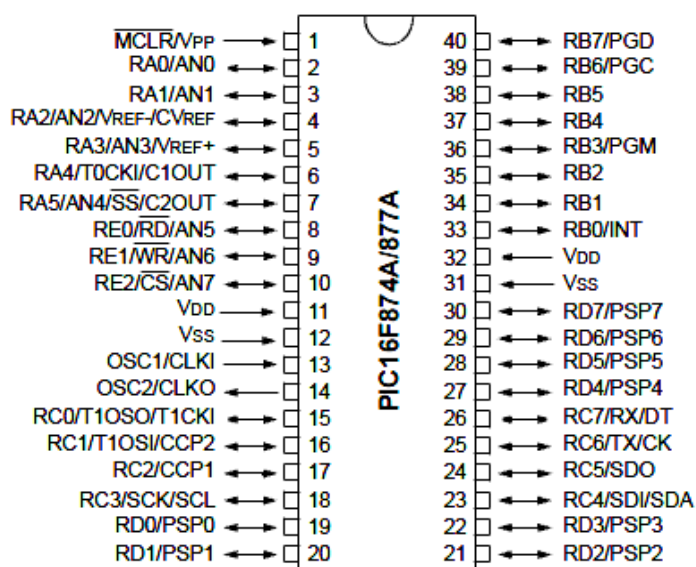


Figura 1-19. Distribución de pines del Microcontrolador PIC16F877A<sup>32</sup>

Tabla 1-6. Nombre y descripción de cada uno de los pines del PIC16F877A<sup>33</sup>

NOMBRE DEL PIN	PIN	TIPO	TIPO DE BUFFER	DESCRIPCIÓN
OSC1/CLKIN	13	I	ST/MOS	Entrada del oscilador de cristal / Entrada de señal de reloj externa
OSC2/CLKOUT	14	O	-	Salida del oscilador de cristal
MCLR/Vpp/THV	1	I/P	ST	Entrada del Master clear (Reset) o entrada de voltaje de programación o modo de control high voltaje test
RA0/AN0	2	I/O	TTL	PORTA es un puerto I/O bidireccional  RA0: puede ser salida analógica 0  RA1: puede ser salida analógica 1  RA2: puede ser salida analógica 2 o referencia negativa de voltaje  RA3: puede ser salida analógica 3 o referencia positiva de voltaje
RA1/AN1	3	I/O	TTL	
RA2/AN2/ Vref-	4	I/O	TTL	
RA3/AN3/Vref+	5	I/O	TTL	

<sup>32</sup> <http://www.datasheet.com>

<sup>33</sup> <http://www.monografias.com/trabajos18/descripcion-pic/descripcion-pic.shtml>

<b>RA4/T0CKI</b>	6	I/O	ST	RA4: puede ser entrada de reloj el timer0.
<b>RA5/SS/AN4</b>	7	I/O	TTL	RA5: puede ser salida analógica 4 o el esclavo seleccionado por el puerto serial síncrono.
<b>RBO/INT</b>	33	I/O	TTL/ST	PORTB es un puerto I/O bidireccional. Puede ser programado todo como entradas RB0 puede ser pin de interrupción externo.
<b>RB1</b>	34	I/O	TTL	
<b>RB2</b>	35	I/O	TTL	
<b>RB3/PGM</b>	36	I/O	TTL	RB3: puede ser la entrada de programación de bajo voltaje
<b>RB4</b>	37	I/O	TTL	Pin de interrupción
<b>RB5</b>	38	I/O	TTL	Pin de interrupción
<b>RB6/PGC</b>	39	I/O	TTL/ST	Pin de interrupción. Reloj de programación serial
<b>RB7/PGD</b>	40	I/O	TTL/ST	
<b>RCO/T1OSO/T1CKI</b>	15	I/O	ST	PORTC es un puerto I/O bidireccional RCO puede ser la salida del oscilador timer1 o la entrada de reloj del timer1
<b>RC1/T1OS1/CCP2</b>	16	I/O	ST	RC1 puede ser la entrada del oscilador timer1 o salida PWM 2
<b>RC2/CCP1</b>	17	I/O	ST	RC2 puede ser una entrada de captura y comparación o salida PWN
<b>RC3/SCK/SCL</b>	18	I/O	ST	RC3 puede ser la entrada o salida serial de reloj síncrono para modos SPI e I2C
<b>RC4/SD1/SDA</b>	23	I/O	ST	RC4 puede ser la entrada de datos SPI y modo I2C
<b>RC5/SD0</b>	24	I/O	ST	RC5 puede ser la salida de datos SPI
<b>RC6/Tx/CK</b>	25	I/O	ST	RC6 puede ser el transmisor asíncrono USART o el reloj síncrono.

<b>RC7/RX/DT</b>	26	I/O	ST	RC7 puede ser el receptor asíncrono USART o datos síncronos
<b>RD0/PSP0</b> <b>RD1/PSP1</b> <b>RD2/PSP2</b> <b>RD3/PSP3</b> <b>RD4/PSP4</b> <b>RD5/PSP5</b> <b>RD6/PSP6</b> <b>RD7/PSP7</b>	19 20 21 22 27 28 29 30	I/O I/O I/O I/O I/O I/O I/O I/O	ST/TTL ST/TTL ST/TTL ST/TTL ST/TTL ST/TTL ST/TTL ST/TTL	PORTD es un puerto bidireccional paralelo
<b>RE0/RD/AN5</b>  <b>RE1/WR/AN</b>  <b>RE2/CS/AN7</b>	8  9  10	I/O  I/O  I/O	ST/TTL  ST/TTL  ST/TTL	PORTE es un puerto I/O bidireccional  RE0: puede ser control de lectura para el puerto esclavo paralelo o entrada analógica 5  RE1: puede ser escritura de control para el puerto paralelo esclavo o entrada analógica 6  RE2: puede ser el selector de control para el puerto paralelo esclavo o la entrada analógica 7.
<b>Vss</b>	12.3 1	P	-	Referencia de tierra para los pines lógicos y de I/O
<b>Vdd</b>	11.3 2	P	-	Fuente positiva para los pines lógicos y de I/O
<b>NC</b>	-	-	-	No está conectado internamente

## **CAPITULO 2**

### **CONSTRUCCIÓN DEL HARDWARE Y DESARROLLO DEL SOFTWARE DEL SISTEMA**

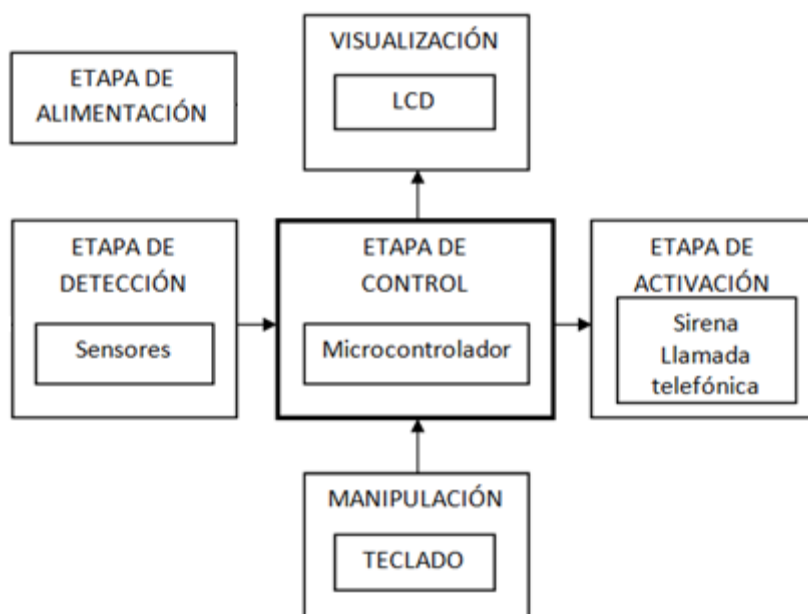
#### **2.1 INTRODUCCIÓN**

Esta alarma antirrobo tiene un microcontrolador como elemento principal y es el encargado de tomar todas las decisiones teniendo en cuenta los eventos que puedan suscitarse para que los sensores de movimiento y magnéticos se activen, los mismos que están distribuidos estratégicamente en todo el espacio de la biblioteca. Si algún sensor es accionado el microcontrolador activa la línea telefónica para generar tonos DMTF que nos permiten realizar llamadas a números telefónicos previamente grabados y de esta manera dar aviso de robo a la directora de la Biblioteca, sin embargo en ese mismo momento el microcontrolador enciende una sirena de alerta para todo el vecindario.

Este sistema también consta de un teclado y un LCD (Display de Cristal Líquido), el teclado nos permite ingresar la clave para activar o desactivar la alarma y a su vez para cambiar de clave y de números telefónicos a los cuales va a llamar el sistema, todo esto se lo podrá visualizar a través de un LCD que normalmente se utiliza para mostrar mensajes que indican al usuario el estado del sistema tales como inicialización, ejecución o cambios en el mismo.

#### **2.2 CONSTRUCCIÓN DEL HARDWARE DEL SISTEMA**

A la estructura de este sistema se lo puede dividir por etapas como se muestra en la Figura 2-1, de esta manera se facilita la explicación de todos los circuitos inmersos en cada una de las etapas.



**Figura 2-1.** Diagrama en bloques de la alarma antirrobo.

### 2.2.1 ETAPA DE CONTROL

La etapa de control es una de las más indispensables en este proyecto, es el cerebro del sistema, todo gira alrededor de ésta, es como un paso obligatorio de todas las etapas para que logren cumplir su respectiva y correcta función en donde el elemento principal es el microcontrolador PIC16F877A que se puede ver en la Figura 2-2, el cual se encarga de tomar las decisiones a la hora de activar la alarma, basándose en la etapa de detección y dando una alerta de robo con una llamada telefónica a la Directora de la Biblioteca y a su vez encendiendo la sirena, es por eso que el microcontrolador es una parte muy primordial para el funcionamiento del sistema.

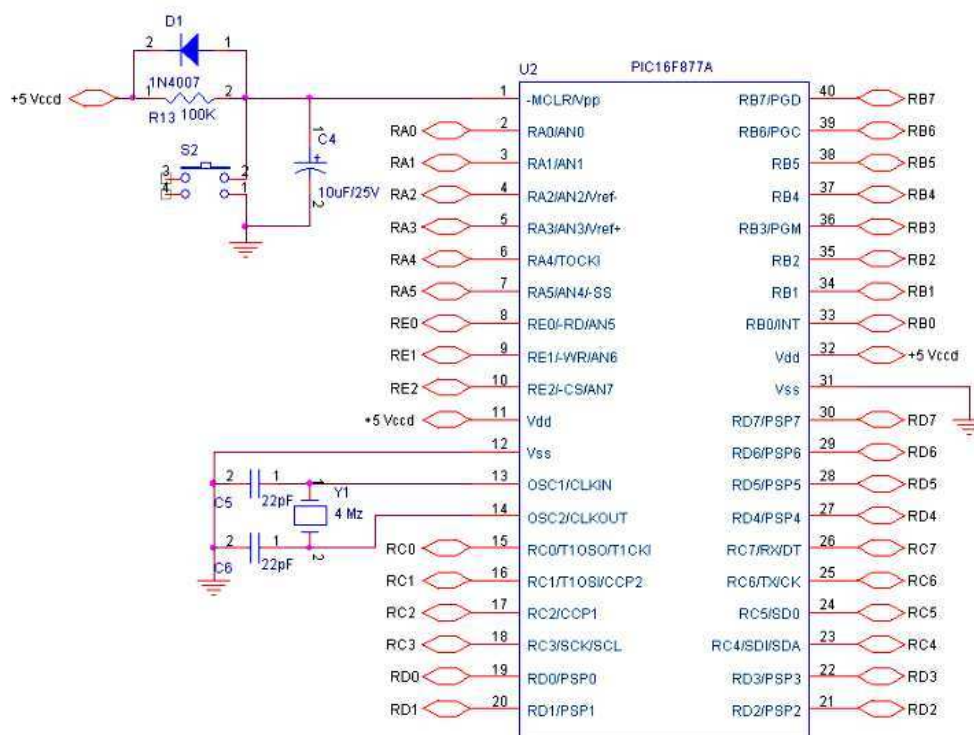


Figura 2-2. Etapa de control (Microcontrolador 16F877A)

### 2.2.1.1 Circuito de Reset

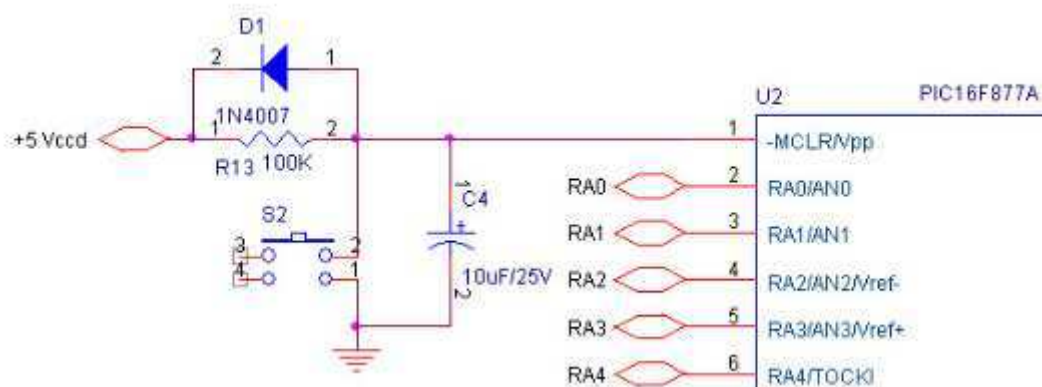


Figura 2-3. Diagrama del circuito de reset.

La función del capacitor es suavizar las transiciones de voltaje de entrada mientras el capacitor se está cargando en el PIN1 se tendrá un uno lógico por lo cual siempre va a estar en ese estado hasta el momento que se presiona el pulsador y el capacitor se descarga haciendo que el PIN1 baje a cero siendo este el estado necesario para activar al reset del PIC y cuando se deja de presionar el capacitor comienza a cargarse y el reset vuelve a su estado normal.



El diodo es de protección para descargas o en caso de corrientes no previstas.

La entrada MCLR tiene la función de reiniciar el estado del microcontrolador, teniendo como consecuencia dos acciones importantes:

- Se carga un 0 en el contador de programa, de forma que después de un Reset siempre se ejecuta la instrucción que está en la posición 0 de la memoria de programa.
- Los registros de estado y control toman un estado conocido y determinado.

### 2.2.1.2 Circuito para conectar el oscilador externo

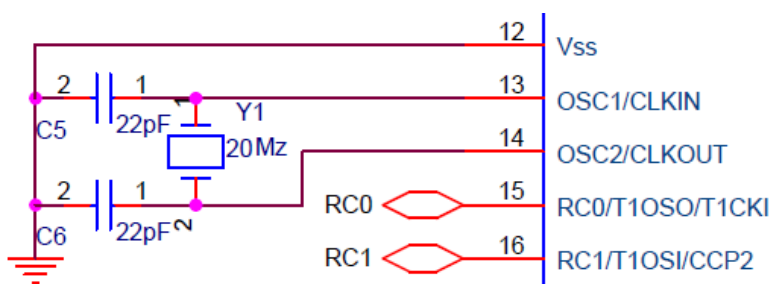


Figura 2-4. Diagrama del circuito para conectar el oscilador externo

Debido a que el oscilador interno que posee el PIC es un RC (resistencia condensador), esto no es muy preciso y para que los tonos que genera el PIC sean validos al cien por ciento es necesario conectar un oscilador externo como en la Figura 2-4 que garantiza mayor precisión y buen arranque del microcontrolador, ya que la frecuencia que genera este viene dada por el oscilador.

El circuito como se muestra en la Figura 2-4 posee un Oscilador de 20MHz basado en un cristal de cuarzo conectado a dos capacitores de 22µf puestos a tierra lo que se llama un circuito oscilador tipo XT (Cristal/Resonador cerámico externo, media frecuencia), el oscilador debe estar conectado a los PINES 13 y 14 que son OSC1/CLKIN (entrada del circuito oscilador externo) y OSC2/CLKOUT (auxiliar del circuito oscilador) respectivamente.

### 2.2.1.3 Pines del microcontrolador usados en las diferentes etapas

Como la etapa de control es la que va a intervenir en todas las demás etapas es necesario hacer una descripción rápida de los pines del microcontrolador destinados para cada una de las mismas.

**Tabla 2-1.** Distribución de pines para cada etapa con sus respectivas funciones

ETAPA	PINES	NOMBRE	FUNCIÓN		
DETECCIÓN	3	RA1	SENSORES MAGNÉTICOS		
	4	RA2			
	5	RA3			
	6	RA4			
	7	RA5			
	8	RE0			
	9	RE1			
	10	RE2			
	ACTIVACIÓN	22		RD3	SENSORES DE MOVIMIENTO
		27		RD4	
28		RD5			
29		RD6			
30		RD7			
ACTIVACIÓN	19	RD0	LLAMADA TELEFÓNICA SIRENA		
	20	RD1			
VISUALIZACIÓN	33	RB0	LCD		
	34	RB1			
	35	RB2			
	36	RB3			
	37	RB4			
	38	RB5			
	39	RB6			
MANIPULACIÓN	40	RB7	TECLADO		
	15	RC0			
	16	RC1			
	17	RC2			
	18	RC3			
	23	RC4			
	24	RC5			
	25	RC6			
ALIMENTACIÓN	26	RC7	FUENTE		
	11	Vdd			
	32	Vdd			
	12	Vss			
	31	Vss			

## 2.2.2 ETAPA DE ALIMENTACIÓN

El sistema requiere de manera primordial una tensión fija y estable de un valor determinado, para lo cual se utiliza la corriente alterna que por medio de un circuito que después se describirá se convierte en una fuente de alimentación continua que tiene de apoyo una batería que en caso de fallo o pérdida de fluido eléctrico entra en funcionamiento caso contrario se recarga mediante el mismo circuito al que va conectada, de tal manera que siempre se tendrá la tensión necesaria para las diferentes etapas del sistema.

### 2.2.2.1 Circuito de alimentación continua de 5v y 12v

Para obtener la alimentación continua el circuito es alimentado por una tensión alterna que nos provee la Empresa Eléctrica Quito la misma que normalmente es de 110v.

Como se puede apreciar en la Figura 2-5 los 110v ingresan por la bornera JP3 para luego ser conectado al primario del transformador, el fusible F1 es de 2A, se denomina fusible a un dispositivo, constituido por un soporte adecuado, un filamento o lámina de un metal o aleación de bajo punto de fusión que se intercala en un punto determinado de una instalación eléctrica para que se funda, cuando la intensidad de corriente supere, por un cortocircuito o un exceso de carga, un determinado valor que pudiera hacer peligrar la integridad de la placa electrónica con el consiguiente riesgo de incendio o destrucción de otros elementos y se lo debe conectar en la fase de la tensión alterna. Los cálculos para encontrar el valor del fusible se los debe hacer tomando en cuenta la intensidad total del sistema a máximo funcionamiento la misma que resulta de la suma de todas las intensidades de cada circuito que interviene en la alarma.

$I_t$  = intensidad total del circuito

$I$  = intensidad máxima que el fusible puede soportar

$$I = I_t + 25\% \quad \text{Ecuación 2.1}$$

$$I = 1.5A + 25\%$$

$$I = 1.875 \approx 2A$$

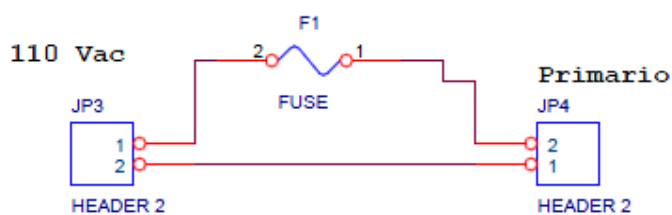


Figura 2-5. Alimentación de tensión alterna y conexión de fusible.

El transformador es un dispositivo usado para variar tensiones y corrientes alternas está formado por dos bobinas de hilo conductor enrolladas sobre un núcleo de hierro común, el transformador usado en este proyecto es un reductor donde el número de espiras de la bobina secundaria es menor que el número de espiras de la bobina primaria por tanto la tensión en el secundario será menor que en el primario, tiene las siguientes características tomado en cuenta los requerimientos del sistema:

VOLTAJE DE ENTRADA	VOLTAJE DE SALIDA	CORRIENTE
110V	12-0-12	2000 mA

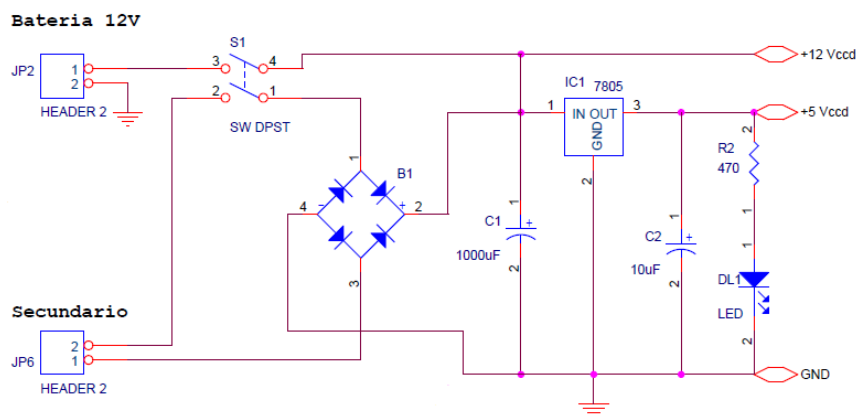


Figura 2-6. Circuito de alimentación al sistema.

El secundario del transformador con un voltaje de 12v ingresa por la bornera JP6, como se puede observar en la figura 2-6 esta nueva tensión es rectificadora a una señal de onda completa mediante el puente de diodos B1, para luego ser filtrada por el capacitor C1, consiguiéndose una tensión continua no estabilizada, esta conexión permite que la batería se cargue siempre que el switch de dos

posiciones este cerrado ya que se tiene un voltaje suficiente para cumplir este propósito, dicha tensión es ingresada al regulador de voltaje.

El regulador 7805 que es un regulador de tensión positiva de 5 Volts 2A la tensión justa y mucho más corriente de la que necesita nuestro PIC para funcionar, se sabe que el buen funcionamiento del firmware que grabemos en el PIC está sujeto no solo a la buena programación que hayamos hecho a la hora de diseñarlo sino que también a una alimentación fija, constante y regulada a la hora de ejecutarlo, entonces la manera más segura, económica y sencilla de obtener ese voltaje, es la utilización de un integrado regulador de voltaje, y el 7805 es el más indicado ya que mantendrá fija la tensión en 5V siempre y cuando en su entrada reciba al menos 6V. Por lo tanto a la entrada podremos despreocuparnos de la alimentación superando por mucho el voltaje de trabajo del PIC. El regulador de voltaje fijo 7805 es un circuito integrado que posee tres terminales, uno para la entrada de la tensión no regulada, otro para la salida regulada y la tercera es la masa común para ambas, este proporciona un voltaje de salida de DC constante que es prácticamente independiente del voltaje de entrada, la corriente de carga de salida y la temperatura.

El regulador 7805 es capaz de producir un voltaje de salida de 5v con una corriente de 2A cuando se usa un disipador de calor apropiado, en cuanto al voltaje de entrada debe ser por lo menos de 7v y como máximo 17v a fin de mantener la regulación. El capacitor C2 disminuye la tensión de rizado de salida, a la vez que evita oscilaciones.

La fuente de alimentación de 12v se la toma antes de ingresar al regulador como se puede apreciar en la Figura 2-6 esta tensión tiene un poco de rizado pero es suficiente para la alimentación de sensores, reles, sirena y por supuesto para cargar la batería la cual en caso de que el servicio eléctrico sea suspendido por cualquier imprevisto entrará en funcionamiento.

La R2 trabaja como limitador de corriente para proteger al diodo led y se la encuentra con el siguiente cálculo:

$$R_2 = \frac{V_{cc} - V_{led}}{10 \text{ mA}} \quad \text{Ecuación 2.3}$$

$$R_2 = \frac{5\text{v} - 1.6\text{v}}{10 \text{ mA}}$$

$$R_2 = 340 \approx 330 \Omega$$

### 2.2.3 ETAPA DE INICIALIZACIÓN

La etapa de inicialización se encarga de la interacción entre el usuario y el sistema, conformado por dos partes primordiales como se puede ver en la Figura 2-1, la visualización que consta de un LCD (Display de Cristal Líquido) y la manipulación que es un teclado para el ingreso de datos.

Esta etapa es la que nos permite mediante el teclado activar la alarma o realizar algún cambio antes de su activación y todo esto poder observarlo en el LCD.

#### 2.2.3.1 Visualización

La visualización consta de un módulo LCD (Display de Cristal Líquido) que es utilizado para mostrar mensajes que indican al usuario el estado del sistema. El LCD permite la comunicación entre el sistema y el usuario, este puede mostrar cualquier caracter ASCII, y consumen mucho menos que los displays a 7 segmentos.

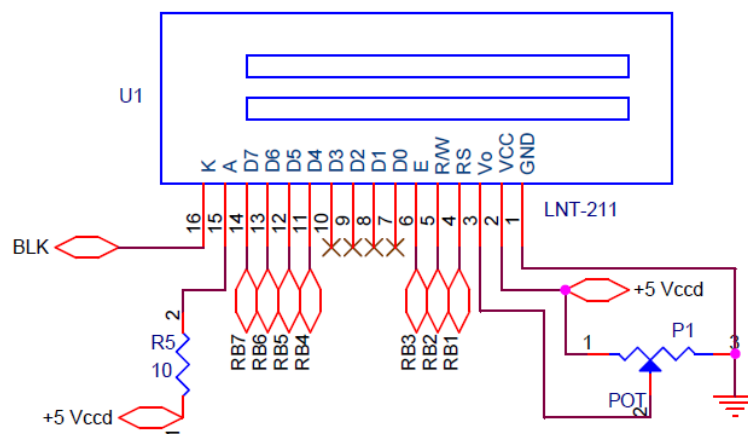
Solamente permiten visualizar mensajes cortos de texto. Existen algunos modelos estandarizados en la industria, en función de su tamaño medido en número de líneas y columnas de texto. Existen modelos de una, dos y cuatro filas únicamente. El número de columnas típico es de ocho, dieciséis, veinte y cuarenta caracteres.

El controlador Hitachi HD44780 se ha convertido en un estándar de industria cuyas especificaciones funcionales son imitadas por la mayoría de los fabricantes. Este controlador cuenta con los siguientes interfaces eléctricos:

- **D0-D7:** ocho señales eléctricas que componen un bus de datos.

- **R/W**: una señal que indica si se desea leer o escribir en la pantalla (generalmente solamente se escribe).
- **RS**: una señal que indica si los datos presentes en D0-D7 corresponden bien a una instrucción o bien a sus parámetros.
- **E**: una señal para activar o desactivar la pantalla.
- **V0**: señal eléctrica para determinar el contraste de la pantalla. Generalmente en el rango de cero a cinco voltios. Cuando el voltaje es de cero voltios se obtienen los puntos más oscuros.
- **Vss y Vdd**: señales de alimentación. Generalmente a cinco voltios.

Estas señales son fácilmente controladas desde un ordenador a través de un interfaz paralelo, típicamente a través del interfaz IEEE 1284, también conocido como “Centronics”. El mismo que se utiliza para conectar impresoras.



**Figura 2-7.** Diagrama de conexión de un módulo LCD

El LCD está conectado al circuito de alarma mediante un bus de datos de 20 canales de los cuales solo 16 son utilizados ya que es un LCD de 2x16, 2 líneas de 16 caracteres cada una.

Como se puede ver en la Figura 2-7 el LCD necesita ser alimentado con 5v que ingresan por el PIN 2, este dispositivo es un modelo con iluminado de pantalla o backlight, lo cual se utilizó por si la central de la alarma se encuentra en un lugar poco luminoso y por simple estética, los pines utilizados para este fin son el PIN 15 para la alimentación del backlight con una resistencia de 10Ω y el PIN 16.

Los LCD se pueden conectar con el PIC con un bus de 4 u 8 bits para el envío de los datos la diferencia está en el tiempo que se demora, pues la comunicación de 4 bits, primero envía los 4 bits más altos y luego los 4 bits más bajos, mientras que la de 8 bits envía todo al mismo tiempo, para la alarma se optó por usar un bus de 4 bits ya que la gran ventaja son los pocos cables que se deben conectar, los PINES de el LCD utilizados para esta función son 11, 12, 13, 14 y que a su vez son conectados a RB4, RB5, RB6, RB7 del PIC respectivamente.

PIN 4 RS es utilizado para la selección del registro control/datos, 0 para registro de control y 1 para registro de datos.

PIN 5 R/W es el pin de lectura/escritura, 0 para escritura y 1 para lectura.

PIN 6 E es el de habilitación con este podemos conectar o desconectar el LCD cuando queramos, de acuerdo con el programa grabado en el PIC.

El potenciómetro 10K $\Omega$  es una resistencia variable, que colocamos para el control de contraste, con lo cual se asegura que la corriente mínima de ingreso sea 0,5 mA, el terminal variable va conectado al PIN 3 del LCD y los dos fijos entre la tierra del PIN 1 y la alimentación de Vcc del PIN 2.

### 2.2.3.2 Manipulación

En esta parte tenemos un teclado matricial que lo conectaremos a nuestro sistema mediante un bus de datos de 20 canales.

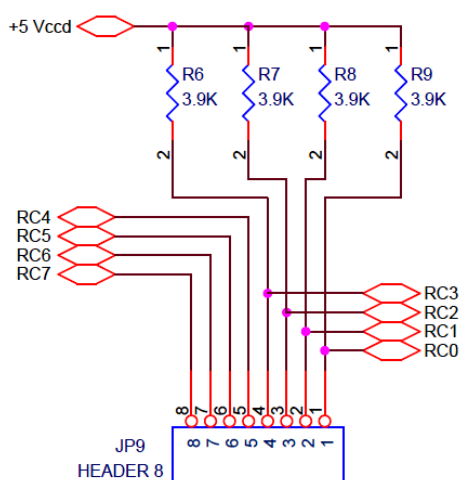


Figura 2-8. Diagrama de conexión del teclado.



Un teclado matricial es un simple arreglo de botones conectados en filas y columnas, de modo que se pueden leer varios botones con el mínimo número de pines requeridos. Un teclado matricial 4×4 solamente ocupa 4 líneas de un puerto para las filas y otras 4 líneas para las columnas, de este modo se pueden leer 16 teclas utilizando solamente 8 líneas de un microcontrolador. Si asumimos que todas las columnas y filas inicialmente están en alto (1 lógico), la pulsación de un botón se puede detectar al poner cada fila a en bajo (0 lógico) y checar cada columna en busca de un cero, si ninguna columna está en bajo entonces el 0 de las filas se recorre hacia la siguiente y así secuencialmente.

El circuito para la conexión del teclado se lo puede apreciar en la Figura 2-8, para asegurar que la corriente que ingresa al PIC no le cause ningún daño se ha colocado resistencias de 3,9 KΩ.

El PIC puede soportar una corriente máxima de entrada de 25 mA, por tanto al calcular la resistencia se tiene:

$$R = \frac{V}{I} \quad \text{Ecuación 2.4}$$

$$R = \frac{5 \text{ v}}{25 \text{ mA}}$$

$$R = 200 \Omega$$

Entonces la resistencia mínima sería 200 Ω, pero esto no es muy aconsejable porque estaríamos al límite de corriente del PIC, por lo que se optó por trabajar con resistencias de 3,9 KΩ que se tendría una corriente de:

$$I = \frac{V}{R} \quad \text{Ecuación 2.5}$$

$$I = \frac{5 \text{ v}}{3.9 \text{ K}\Omega}$$

$$I = 1.28 \text{ mA}$$

Entonces trabaja con una corriente de 1,28 mA y con esto no se sobre exige al PIC.

## 2.2.4 ETAPA DE DETECCIÓN

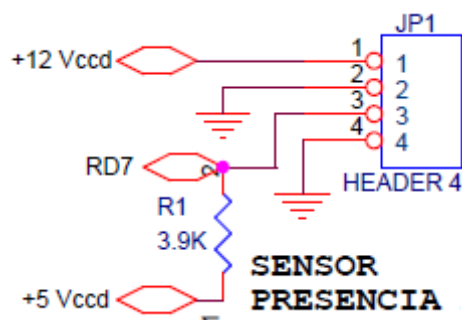
Para la etapa de detección se han colocado 5 sensores magnéticos y 5 sensores de presencia distribuidos estratégicamente alrededor de la biblioteca, los cuales al ser accionados envían una señal al microcontrolador, el mismo que dará aviso de robo mediante una llamada telefónica y también al sonar la sirena.

### 2.2.4.1 Detección de movimiento

Esta sección consta de sensores de presencia infrarrojos PIR que detectan cambios de temperatura (calor corporal) y movimiento. Si estos sensores detectan movimiento estando el sistema conectado cambia el nivel lógico de un pin por consiguiente se disparará la alarma. Existen detectores regulados para no detectar mascotas, tales como perros y gatos.

Estos sensores de movimiento brindan una protección en el interior de la propiedad.

Estos se alimentan con 12v de corriente continua, tienen en su interior contactos del relé normalmente cerrados y su conexión se muestra en la Figura 2.9



**Figura 2-9.** Circuito de conexión para el sensor de presencia.

JP1 es una bornera en la cual va conectado el sensor, de donde los terminales 1(+12v) y 2(0v) son la polarización mientras que los dos restantes son las entradas del contacto normalmente cerrado, a una de las entradas de dicho contacto conectamos RD7 del micro controlador, una resistencia de 3,9 KΩ (Ecuación 2.5) y 5 v de corriente continua, manteniendo así el 0L en RD7 mientras el contacto este cerrado ya que el otro terminal está conectado a tierra, de esta manera cuando el sensor detecte algún movimiento el contacto del relé se abrirá y tendremos un 1L en RD7 activando así la alarma.

### 2.2.4.2 Detección de apertura de puertas y ventanas

Para este bloque hemos usado sensores magnéticos que son detectores que sirven para proteger todos los accesos de la casa que dan al exterior, como las puertas ó ventanas de uso normal, pudiendo ser instalados en distintos tipos de aberturas de metal o de madera, siempre y cuando las mismas no tengan movimiento con el viento.

Normalmente, se componen de dos partes: el imán actuador, y el interruptor propiamente dicho. El interruptor consiste en unas lengüetas de material magnético, aleación especial de ferro níquel, con zona de contacto protegida por un recubrimiento de metal noble, generalmente de oro, con una separación entre ambas partes de 0,2 a 0,3 mm. y funcionan cuando se realiza la apertura de una puerta o ventana al separarse del dispositivo imán.

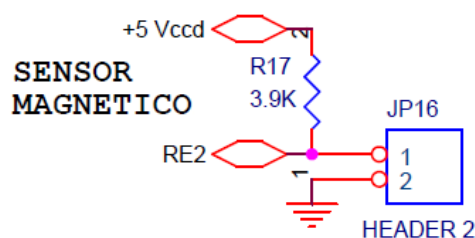


Figura 2-10. Circuito de conexión de un sensor magnético

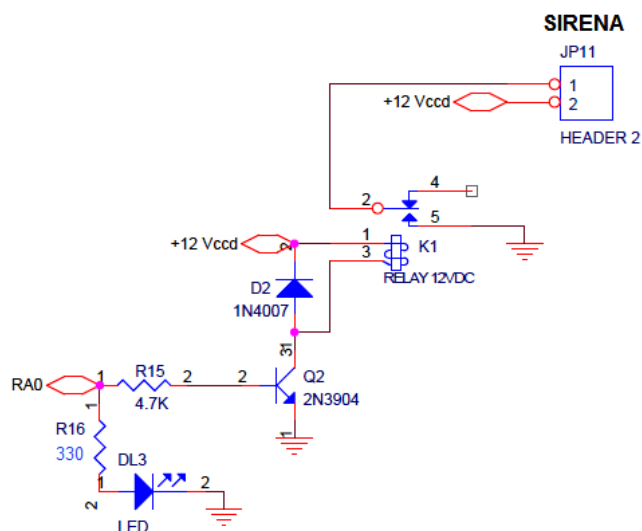
JP16 es la bornera en la cual conectamos el sensor magnético, éste sensor funciona como un interruptor normalmente cerrado es por eso que en uno de los terminales conectamos RE2, una resistencia de 3.9 K $\Omega$  (Ecuación 2.5) y 5v de corriente continua de forma similar al sensor de presencia esto es para mantener un 0L en la entrada RE2, ya que el otro terminal está conectado a tierra. Cuando la puerta o una ventana se abran el interruptor magnético se abrirá dejando pasar un 1L en RE2 activando el sistema.

### 2.2.5 ETAPA DE ACTIVACIÓN

Esta etapa cumple la función de alerta o aviso al usuario de una incursión ajena en la Biblioteca y se la realiza mediante dos formas que a continuación se describen.

### 2.2.5.1 Circuito de conexión a la sirena

La sirena es el dispositivo con el cual el sistema da aviso a los vecinos que la Biblioteca está siendo robada y con esto se logrará un pronto auxilio.



**Figura 2-11.** Circuito de conexión a la sirena.

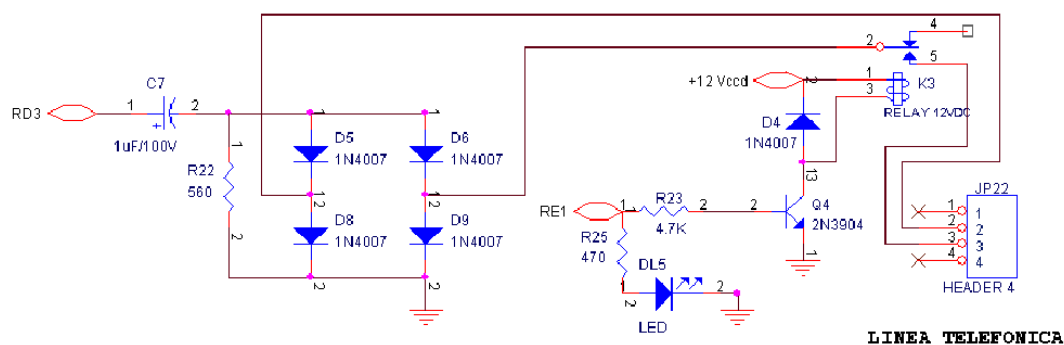
El circuito de la Figura 2-11 tiene la función de activar la sirena, cuando se acciona cualquier sensor el micro envía un pulso al transistor 2N3904 para que éste entre en saturación funcionando como un interruptor cerrado y dejando pasar lo que se encuentra en su emisor, polarizando así el relé el cual es activado, el mismo que deja pasar tierra logrado polarizar la sirena y a su vez haciéndola sonar.

El diodo 1N4007 es un elemento unidireccional el cual deja pasar la corriente en un solo sentido además se colocó en paralelo con el relé para asegurar la disipación de energía de este cuando pase de estado encendido – apagado.

El diodo led DL3 nos indica que la sirena está encendida.

### 2.2.5.2 Circuito de conexión a la línea telefónica

El circuito está diseñado básicamente para generar una llamada mediante la conexión del microcontrolador PIC con la línea telefónica, como se puede ver en la Figura 2-12, donde cada uno de los elementos tiene su función específica.



**Figura 2-12.** Circuito de conexión a la línea telefónica

El PIC genera tonos DTMF según su programación y son enviados a través del PIN RD3 para hacer la llamada telefónica, una vez marcado el número deseado el PIC envía por el mismo PIN señales o tonos diferentes según el sensor que fue activado dando así una alerta de robo, en caso de que el usuario no conteste el programa está desarrollado para que el PIC llame a otro teléfono y así rotatoriamente hasta que el usuario o algún delegado llegue a la biblioteca y desactive el sistema.

La resistencia de  $560\Omega$  a 1vatio colocada paralelamente a la red telefónica sirve para simular la carga de un teléfono normal, y con esto podemos obtener el tono de marcado necesario para hacer la llamada, asimismo se notará que esta empieza a disipar calor en el momento en que se une a la red telefónica, esto debido a los casi 50v DC que circulan por dicha red.

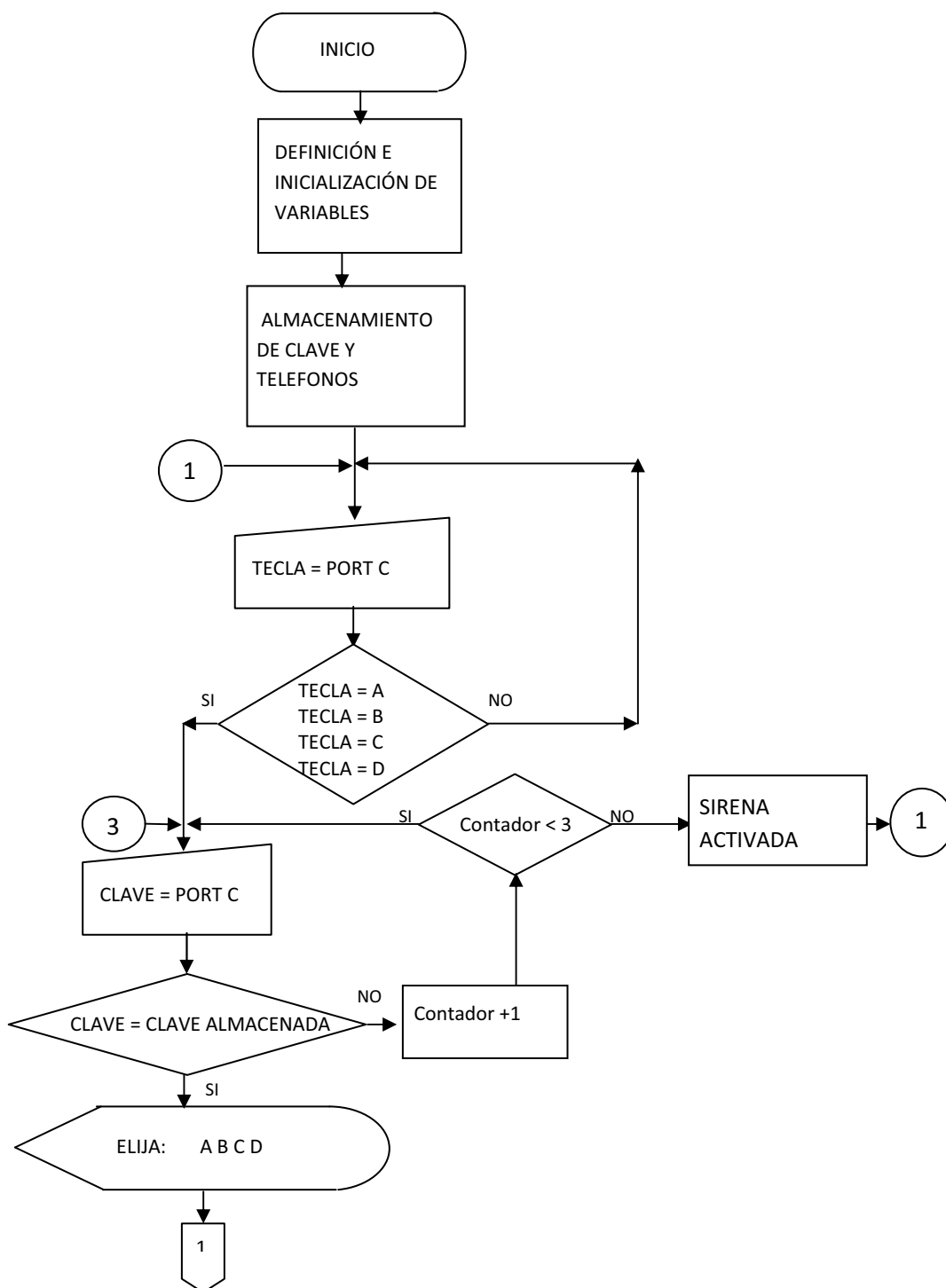
El relé hace la conexión y desconexión de la red telefónica, en este caso sería como el auricular que cuelga y descuelga el teléfono, el funcionamiento de este proceso es similar al de la activación de la sirena. El capacitor electrolítico de  $1\mu\text{f}$  sirve para poder mejorar la onda que sale del PIC y además como protección para el mismo. Como ya se mencionó la red telefónica suministra alrededor de 50v DC, es así que el capacitor debe pasar de este voltaje, por tal razón el valor de dicho elemento es de  $1\mu\text{f}$  a 100v.

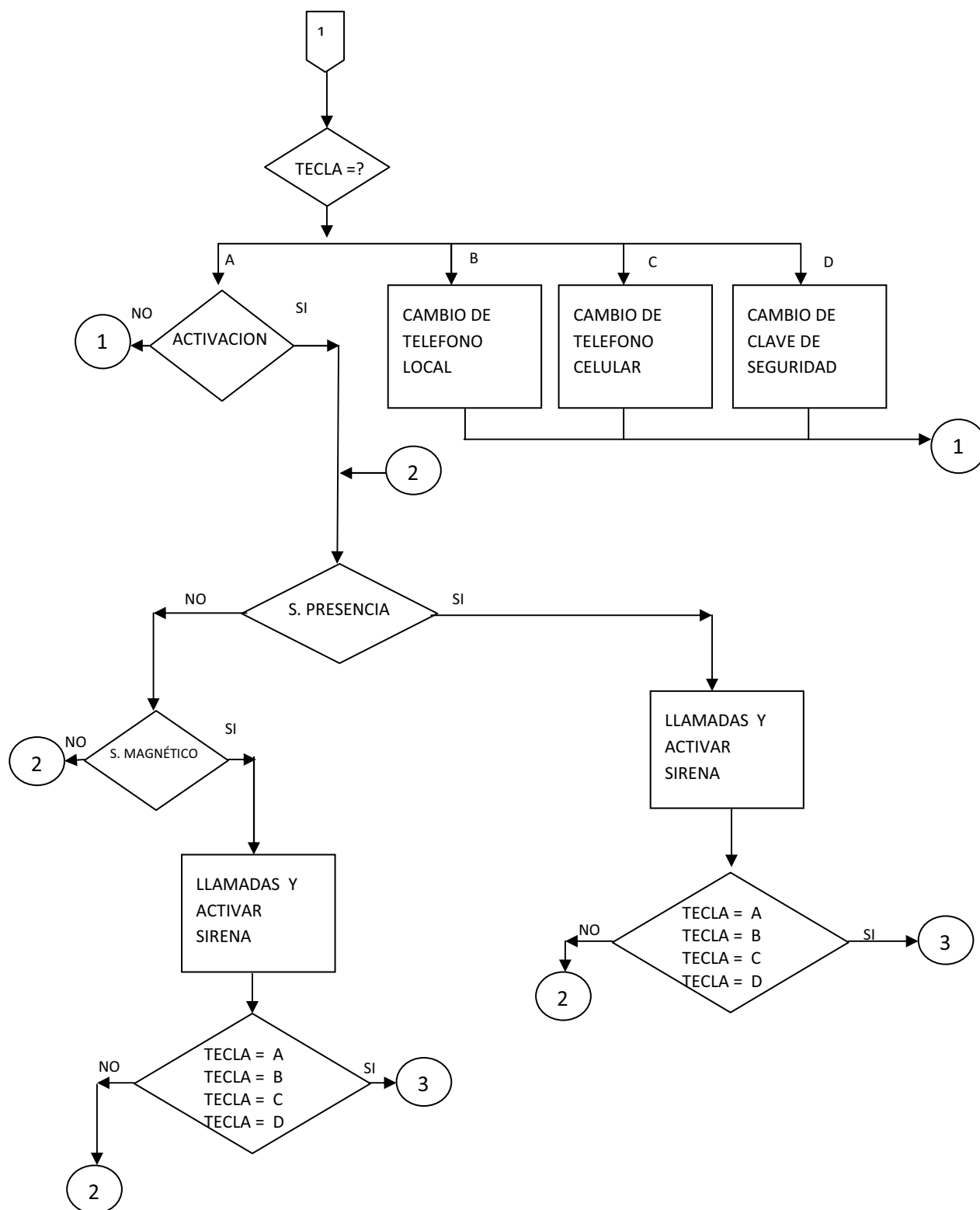
Para no tener inconveniente al momento de identificar la polaridad de la red telefónica para la conexión hacia el PIC, se coloca un puente de diodos a la entrada de la red, en donde el lado positivo ya está unido a tierra y el lado negativo va a través del filtro hacia el PIC.

## 2.3 DESARROLLO DEL SOFTWARE DEL SISTEMA

En el presente tema se hará una descripción del software desarrollado el cual logrará que cada etapa entre en correcto funcionamiento según los requerimientos propuestos, es decir dará un sentido lógico al hardware que se analizó anteriormente.

### 2.3.1 DIAGRAMA DE FLUJO





### 2.3.2 LENGUAJE DE PROGRAMACIÓN UTILIZADO

Para entender la diferencia entre el lenguaje de programación basic y el ensamblador, se debe tener en claro qué es un lenguaje de alto nivel y qué es un lenguaje de bajo nivel, a través del siguiente cuadro podemos ver los niveles de programación.

<b>LENGUAJE HUMANO</b>		
Lenguaje de alto nivel	HIGH led	Lenguaje BASIC
	.....	Otros lenguajes
	.....	
	bsf porta,0	Lenguaje Ensamblador
	3F83 345F	Código Hexadecimal
Lenguaje de bajo nivel	10110011011110	Código Binario
<b>LENGUAJE DE MÁQUINA</b>		

Figura 2-13. Niveles de programación.

La programación en un lenguaje de alto nivel (como el C ó el Basic) que es el que vamos a usar para este proyecto permite disminuir el tiempo de desarrollo de un producto. No obstante, si no se programa con cuidado, el código resultante puede ser mucho más ineficiente que el programado en ensamblador. Las versiones más potentes suelen ser muy caras, aunque para los microcontroladores más populares pueden encontrarse versiones demo limitadas e incluso compiladores gratuitos.

El lenguaje BASIC original fue inventado en 1964 por John George Kemeny (1926-1993) y Thomas Eugene Kurtz (1928-) en el Dartmouth College. En los años subsiguientes, mientras que otros dialectos de BASIC aparecían, el BASIC original de Kemeny y Kurtz era conocido como BASIC Dartmouth.

BASIC fue diseñado para permitir escribir programas usando terminales de computador de tiempo compartido. BASIC estaba intencionado para facilitar los problemas de complejidad de los lenguajes anteriores, con un nuevo lenguaje diseñado específicamente para la clase de usuarios que los sistemas de tiempo compartido permitían: un usuario más sencillo, a quien no le interesaba tanto la velocidad, sino el hecho de ser capaz de usar la máquina. Los diseñadores del



lenguaje también querían que permaneciera en el dominio público, lo que contribuyó a que se diseminara.

Los ocho principios de diseño de BASIC fueron:

- Ser fácil de usar para los principiantes.
- Ser un lenguaje de propósito general.
- Permitir que los expertos añadieran características avanzadas, mientras que el lenguaje permanecía simple para los principiantes.
- Ser interactivo.
- Proveer mensajes de error claros y amigables.
- Responder rápido a los programas pequeños.
- No requerir un conocimiento del hardware de la computadora.
- Proteger al usuario del sistema operativo.

El compilador usado para la elaboración del software y utilizando el lenguaje basic es el PicBasic Pro de Micro Engineering Labs Inc. es uno de los más conocidos. Este poderoso compilador pone al alcance del usuario potentes instrucciones para comunicación serie, matemática de 16 bits, mediciones de sensores analógicos, PWM, sonido, y muchísimas más.

Además de generar los files "hex" y también es capaz de generar los files "asm". De tal manera que sí se pueden hacer modificaciones de bajo nivel.

Otra magnífica característica de este compilador es que además de soportar al PIC16F84 también soporta a muchos otros de la gran familia de MICROCHIP. Por ejemplo los micros Flash PIC16F628, 16F876 y el 16F877.

### **2.3.2.1 Manejo del MicroCode Studio**

Para editar el programa utilizamos MicroCode Studio que es un Entorno de desarrollo Integrado (IDE), diseñado exclusivamente para facilitar la programación de los PIC debido a que es un software muy sencillo, utilizaremos el compilador PicBasic Pro (PBP) mencionado anteriormente que es el lenguaje de programación que hace más fácil y rápido la programación de microcontroladores,

los procedimientos para programar y editar son muy sencillos y a continuación se detallan:

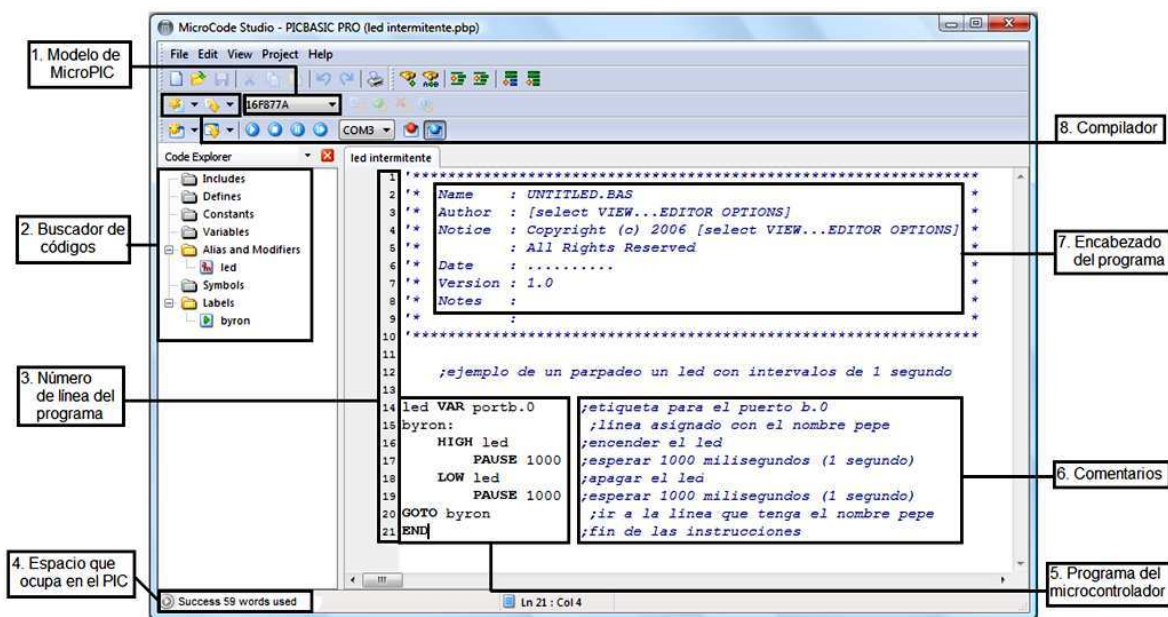


Figura 2-14. Partes del editor Microcode

1. **Modelo de MicroPIC.-** Esto es lo primero que debe seleccionar antes de empezar a programar, seleccione de acuerdo al modelo de PIC que se va a programar para nuestro caso será el 16F877A.
2. **Buscador de códigos.-** Aquí se van adicionando cada vez que se crea una variable, al incluir un define, o crear algún nombre de línea, sirve para saber qué componentes influyen en el programa y también como buscador de líneas, para esto basta con dar un clic en el nombre de la línea que desea encontrar y automáticamente le indicara donde esta dicha línea.
3. **Número de línea del programa.-** Esto por defecto no viene habilitado, se lo debe habilitar previamente, y es muy útil a la hora de encontrar errores, porque le indica el número de la línea en donde se halla un error. Esto no es el tamaño de líneas que ocupa el PIC, sino el que ocupa en Basic.
4. **Espacio que ocupa en el PIC.-** Es el espacio que se requiere en la memoria FLASH del PIC y aparece una vez que se compila el programa, debe fijarse si alcanza en el PIC que se selecciono o debe ser remplazado por otro de mayor capacidad.

5. **Programa del microcontrolador.-** En esta parte se debe escribir el programa, MicroCode reconoce palabras claves como **VAR, HIGH, LOW**, etc., y los pinta con mayúsculas y negrillas, por lo que no se debe utilizar estas palabras como nombres de subrutinas o variables.
6. **Comentarios.-** Es recomendable usar comentarios todo el tiempo, aunque sea obvio para el programador, alguien podría necesitarlo.
7. **Encabezado del programa.-** No son nada más que comentarios en los que se puede incluir: nombre, fecha, autor y una explicación breve de cómo y para qué sirve el programa.
8. **Compilador.-** Estos botones sirven básicamente para compilar el programa y crear el archivo .ASM, .MAC, y el .HEX, el .HEX sirve para grabar en el micro, el .MAC sólo sirve para el PICBasic y el .ASM, para personas interesadas en ver como lo hizo el compilador en assembler.

### 2.3.2.2 Declaraciones disponibles en el Compilador PicBasic PRO

En la siguiente Tabla se pone a disposición las 83 instrucciones disponibles con una breve explicación estas declaraciones son cada una de las palabras que el compilador tiene reservado para realizar una tarea específica.

**Tabla 2-2.** Declaraciones disponibles en el Compilador PicBasic PRO.

DECLARACIÓN	APLICACIÓN
@	Inserta una línea de código ensamblador
ADCIN	Lee el conversor analógico
ASM..ENENDASM	Insertar una sección de código ensamblador
BRANCH	GOTO Computado (equivale a ON...GOTO)
BRANCHL	BRANCH Fuera de pagina (BRANCH largo)
BUTTON	Anti-rebote y auto-repetición de entrada en el pin especifico
CALL	Llamada de subrutina a ensamblador
CLEAR	Hace cero todas las variables
CLEARWDT	Hace cero el contador del Watchdog timer
COUNT	Cuenta el numero de pulsos en el pin
DATA	Define el contenido inicial en un Chip EEPROM
DEBUG	Señal sincrónica de salida en un pin fijo y baud
DEBUGIN	Señal Asincrónica de salida en un pin fijo y baud
DISABLE	Deshabilita el procesamiento de ON INTERRUPT, ON DEBUG

DISABLE DEBUG	Deshabilita el procesamiento de ON DEBUG
DISABLE INTERRUPT	Deshabilita el procesamiento de ON INTERRUPT
DTMFUOT	Produce tonos telefónicos en un pin
EEPROM	Define el contenido inicial en un Chip EEPROM
ENABLE	Habilita el procedimiento ON INTERRUPT, ON DEBUG
ENABLE DEBUG	Habilita el procedimiento ON DEBUG
ENABLE INTERRUPT	Habilita el procesamiento de ON INTERRUPT
END	Detiene la ejecución e ingresa en modo de baja potencia
FOR...NEXT	Ejecuta declaraciones en forma repetitiva
FREQOUT	Produce hasta dos frecuencias en un pin
GOSUB	Llama a una subrutina BASIC en la línea especificada
GOTO	Continúa la ejecución n en la línea especificada
HIGH	Saca el 1 Lógico (5V) por un pin
HPWM	Salida de hardware con ancho de pulsos modulados
HSERIN	Entrada serial asincrónica (hardware)
HSEROUT	Salida serial asincrónica (hardware)
12CWRITE	Graba bytes de dispositivos 12C
12CWRITE	Graba bytes de dispositivos 12C
IF...THEN..ELSE...ENDIF	Ejecuta declaraciones en forma condicional
INPUT	convierte un pin en entrada
LCDIN	Lee caracteres desde una RAM de un LCD
LCDOUT	Muestra caracteres en un LCD
LET	Asigna el resultado de una expresión a una variable
LOOKDOWN	Busca el valor en una tabla de constantes
LOOKDOWN2	Busca el valor en una tabla de constantes o variables
LOOKUP	Obtiene el valor constante de una tabla
LOOKUP2	Obtiene el valor constante o variable de una tabla
LOW	Hace cero lógico un pin específico
NAP	Apaga el procesador por un corto periodo de tiempo
ON DEBUG	Ejecuta un DEBUG en BASIC
ON INTERRUPT	Ejecuta una subrutina Basic en un interrupt
OUTPUT	convierte un pin en salida
OWIN	Entrada de dispositivos un – alambre
OWOUT	salida a dispositivos un-alambre
PAUSE	Demora con resolución de un milisegundo
PAUSEUS	Demora con resolución de un microsegundo
PEEK	Lee un byte de registro
POKE	Graba un byte en el registro
POT	Lee el potenciómetro en el pin especificado
PULSIN	Mide el ancho de pulso en un pin
PULSOUT	Genera un pulso hacia un pin
PWM	Salida modulada en ancho de pulso por un pin especificado
RANDOM	genera numero Pseudo-aleatorio
RCTIME	Mide el ancho de pulso en un pin

READ	Lee un byte de un chip EEPROM
READCODE	Lee palabra desde un código de memoria
RESUME	Continúa la ejecución desde una interrupción
RETURN	Continúa en la declaración que sigue al último GOSUB
REVERSE	Convierte un pin de salida en entrada, o uno de entrada en salida
SELECT CASE	Compara una variable con diferentes valores
SERIN	Entrada serial asincrónica (tipo BASIC Stamp1)
SERIN2	Entrada serial asincrónica (tipo BASIC Stamp2)
SEROUT	salida serial asincrónica (tipo BASIC Stamp1)
SEROUT2	Salida serial asincrónica (tipo BASIC Stamp2)
SHIFTIN	Entrada serial sincrónica
SHIFTOUT	Salida serial sincrónica
SLEEP	Apaga el procesador por un periodo de tiempo
SOUND	Genera un tono o ruido blanco en un pin
STOP	Detiene la ejecución del programa
SWAP	Intercambia los valores de dos variables
TOGGLE	Hace salida a un pin y cambia de estado
USBIN	Entrada de USB
USBINIT	Inicializar USB
USBOUT	Salida de USB
WHILE...WEND	Ejecuta declaraciones mientras la condición sea cierta
WRITE	Graba bytes en un chip EEPROM
WRITECODE	Escribe palabra en código de memoria
XIN	Entrada X-10
XOUT	Salida X-10

### 2.3.3 ELABORACIÓN DEL PROGRAMA PARA EL SISTEMA

```

*****
!* Name   : AlarmaDTMF5.BAS *
!* Author : Byron Hermógenes Nogales Minga *
!* Notice : Copyright (c) 2009 [set under view...options] *
!*       : All Rights Reserved *
!* Date   : 15/11/2009 *
!* Version : 1.0 *
!* Notes  : *
!*       : *
*****

'Programa para hacer una llamada telefónica, cuando
'se activan los sensores

'DEFINICIONES INICIALES fusibles de configuración
'Fusibles
@ DEVICE PIC16F877A, HS_OSC, WDT_OFF, PWRT_OFF, BOD_OFF, LVP_OFF
@ DEVICE PIC16F877A, CPD_OFF, WRT_OFF, DEBUG_OFF, PROTECT_OFF

'DEFINICIONES INICIALES PARA CONFIGURAR EL HARDWARE DEL LCD*****
' Define el pòrtico de Datos
DEFIne LCD_DREG PORTB

```

```
' Define el Bit de inicio de los Datos (0 or 4) si el bus es de 4-bit
DEFINE LCD_DBIT 4
' Define el p rtico en donde se encuentra el Bit E (Enable) del LCD
DEFINE LCD_EREG PORTB
' Define la posici n del bit en el p rtico del Bit E (Enable) del LCD
DEFINE LCD_EBIT 3
' Define el p rtico en donde se encuentra el Bit R/W (Read/Write) del LCD
DEFINE LCD_RWREG PORTB
' Define la posici n del bit en el p rtico del Bit R/W del LCD
DEFINE LCD_RWBIT 2
' Define el p rtico en donde se encuentra el Bit RS (Register Select) del LCD
DEFINE LCD_RSREG PORTB
' Define la posici n del bit en el p rtico del Bit RS (Register Select) del LCD
DEFINE LCD_RSBIT 1
' Define el tama o del bus de datos del LCD (4 or 8 bits)
DEFINE LCD_BITS 4
' Define el n mero de l neas del LCD
DEFINE LCD_LINES 4
' Define el tiempo de retardo para el env o del comando en microsegundos (us)
DEFINE LCD_COMMANDUS 2000
' Define el tiempo de retardo para el env o del dato en microsegundos (us)
DEFINE LCD_DATAUS 50
```

'DEFINICIONES INICIALES PARA CONFIGURAR EL TECLADO \*\*\*\*\*

```
F1 VAR PORTC.4 'nombres para los pines de las filas
F2 VAR PORTC.5
F3 VAR PORTC.6
F4 VAR PORTC.7
```

```
C1 VAR PORTC.0 'nombres para los pines de las columnas
C2 VAR PORTC.1
C3 VAR PORTC.2
C4 VAR PORTC.3
```

' DEFINICIONES DE LAS VARIABLES

```
TonoDTMF var portd.0 'pin que genera tonos DTMF
LiTef VAR portd.1 'nombre rel  para el pin D.1, conecta L nea Telef nica
Alarma VAR porta.0 'nombre rel  para el pin A.0 conectado ALARMA
SM1 var Porte.2 'Sensor Magn tico 1
SM2 var Porte.1 'Sensor Magn tico 2
SM3 var Porte.0 'Sensor Magn tico 3
SM4 var Porta.5 'Sensor Magn tico 4
SM5 var Porta.4 'Sensor Magn tico 5
SM6 var Porta.3 'Sensor Magn tico 6
SM7 var Porta.2 'Sensor Magn tico 7
SM8 var Porta.1 'Sensor Magn tico 8

SP1 var PortD.7 'Sensor Presencia 1
SP2 var PortD.6 'Sensor Presencia 2
SP3 var PortD.5 'Sensor Presencia 3
SP4 var PortD.4 'Sensor Presencia 4
SP5 var PortD.3 'Sensor Presencia 5

Led1 var PortD.2 'Led Indicador
BK var Portb.0 'BK
```

```
Tiempo VAR word
Tiempo1 VAR word
Pausa VAR word
```

```

v      VAR BYTE
w      VAR BYTE
x      VAR BYTE
y      VAR BYTE
Z      VAR BYTE
Tecla  VAR BYTE
Cont   var byte
Cont1  var byte
Cont2  var byte
CLAVE  VAR BYTE [4]
CLAVEDEF VAR BYTE [4]
Teléfono VAR BYTE [10]
D1     VAR BYTE
D2     VAR BYTE
D3     VAR BYTE
D4     VAR BYTE
flag1  var bit
flag2  var bit
flag3  var bit
flag4  var bit
flag5  var bit
flag6  var bit
Activar var bit
Espera var bit

Tini1  con 30 'Constante Inicial espera 90 segundos
Tini2  con 90 'Constante Inicial espera 90 segundos, activar ALARMA
delay1 con 3500 'Constante entre mensajes (mseg.)
Mil    con 1000 'Constante de 1000
Retardo con 6000 '6000 = 1min. Constante 30000 aproximadamente 5 minutos
TiempoSonido con 50 'repetir 200 veces, equivale aproximadamente a 48 segundos
A0 VAR BYTE
A1 VAR BYTE
A2 VAR BYTE
A3 VAR BYTE
A4 VAR BYTE
A5 VAR BYTE
A6 VAR BYTE
A7 VAR BYTE
A8 VAR BYTE

B0 VAR BYTE
B1 VAR BYTE
B2 VAR BYTE
B3 VAR BYTE
B4 VAR BYTE
B5 VAR BYTE
B6 VAR BYTE
B7 VAR BYTE
B8 VAR BYTE

***** GUARDA CLAVE ACTIVACIÓN ON/OFF *****
EEPROM 0, [1,2,3,4] 'cargar la memoria EEPROM desde la dirección 0,CLAVE
FOR X=0 TO 3
  READ X,CLAVEDEF[X]
NEXT X
***** GUARDA TELÉFONO DE PRUEBA1 *****
EEPROM 8, [ 2,5,4,1,0,7,1] 'cargar la memoria EEPROM desde la dirección 8
*****

```

```

READ 8,A0
READ 9,A1
READ 10,A2
READ 11,A3
READ 12,A4
READ 13,A5
READ 14,A6
'READ 15,A7
'READ 16,A8
***** GUARDA TELÉFONO DE PRUEBA2 *****
EEPROM 24, [ 0,8,4,0,0,7,5,4,8] 'cargar la memoria EEPROM desde la dirección 24
*****

READ 24,B0
READ 25,B1
READ 26,B2
READ 27,B3
READ 28,B4
READ 29,B5
READ 30,B6
READ 31,B7
READ 32,B8

DEFINE OSC 20      'definir oscilador externo de 20 MHZ
TRISA = %00111110  'Setea PORTA
TRISB = %00000000  'Setea PORTB
TRISC = %00001111  'Setea PORTC
TRISD = %11111000  'Setea PORTD
TRISE = %00000111  'Setea PORTE
ADCON1 = 7         'Pórtico A, como digitales

' INICIALIZACION DE VARIABLES
PORTC=255          'Inicializa el teclado
alarma=0           'Alarma apagada
LiTef=0            'Apagado interface línea telefónica
LED1=0             'Apagado Led Indicador
BK=0               'Apagado BK del lcd
TonoDTMF=0         'Apagado Tonos DTMF
CONT=0
CONT1=0
CONT2=0
flag3=0
flag5=0
Activar=0
Espera=0
Tiempo=Retardo
Tiempo1=0
Iniciar:
    Pause 500      'Espera inicialización del LCD
    bk=1
    Lcdout $fe, 1  'Limpia la pantalla
    Lcdout "  ALARMA  "
    Lcdout $fe, $c0 'Ir a la Segunda línea
    Lcdout "REALIZADO POR:"
    Pause delay1   'Esperar 4 segundos
    Lcdout $fe, 1  'Limpia la pantalla
    Lcdout "  BYRON  "
    Lcdout $fe, $c0 'Ir a la Segunda línea
    Lcdout "  NOGALES  "
    Pause delay1   'Esperar 4 segundos

```



```

Lcdout $fe, 1      'Limpia la pantalla
Lcdout "INICIALIZANDO..."
Lcdout $fe, $c0    'Ir a la Segunda línea
Lcdout "SENSORES ",dec3 Tini1," SEG" *****
Z=Tini1
FOR V=1 TO Tini1
  Z=Z-1
  PAUSE mil      'Pausa de un segundo
  Lcdout $fe, $c0  'Ir a la Segunda línea
  Lcdout "SENSORES ",DEC3 Z," SEG"
NEXT V
Pause mL         'Esperar 1 segundo
Lcdout $fe, 1    'Limpia la pantalla
Inicio:
if activar=0 then
  if flag5=1 then
    Lcdout $FE,$80," ALARMA " 'Ir a la Primera línea
    Lcdout $fe,$C0," ACTIVADA " 'Ir a la Segunda línea
  else
    Lcdout $FE,$80," ALARMA " 'Ir a la Primera línea
    Lcdout $fe,$C0," DESACTIVADA " 'Ir a la Segunda línea
  endif
  ' pause delay1
  bk=0
else
  if espera=0 then
    Lcdout $FE,$80," Revisando " 'Ir a la Primera línea
    Lcdout $fe,$C0," Sensores.. " 'Ir a la Segunda línea
    bk=0
    gosub sensores
  else
    if flag6=0 then
      pauseus 1'mil
      Tiempo=Tiempo-1
      Tiempo1=Tiempo/100
      Lcdout $FE,$80,"ESPERANDO 1 MIN." 'Ir a la Primera línea
      Lcdout $fe,$C0," ",DEC3 Tiempo1," SEG. " 'Ir a la Segundo línea
      if Tiempo=0 then
        Tiempo=Retardo
        espera=0
      endif
    endif
  endif
endif
if flag6=0 then
  tecla=255
  gosub BARRIDO
  gosub REBOTE
endif
if tecla>9 and tecla<14 then
  flag3=1
  flag6=0
  bk=1
else
  flag3=0
endif

while flag3=1
  TECLA=255

```

```

IF cont=0 then
  LCDOUT $fe,$80,"INGRESE LA CLAVE"
  LCDOUT $FE,$C0,"      "
  LCDOUT $FE,$C6
endif
gosub BARRIDO
gosub REBOTE

select case TECLA
  IF CONT<4 and tecla < 10 THEN
    CLAVE[CONT]=TECLA
    LCDOUT "*"
    cont=cont+1
  ENDIF
  case 15
    if cont=4 then
      Flag1=1
      FOR X=0 TO 3
        IF CLAVEDEF[X] <> CLAVE[X] THEN
          FLAG1=0
        ENDIF
      NEXT X
      IF FLAG1=1 THEN
        LCDOUT $FE,1," CLAVE CORRECTA "
        pause delay1
        LCDOUT $FE,1,"ELIJA: A B C D "
        gosub teclasabcd
        flag3=0
        CONT1=0
      else
        LCDOUT $FE,1,"CLAVE INCORRECTA"
      ENDIF
      PAUSE delay1
    else
      LCDOUT $FE,1,"CLAVE INCORRECTA"
      PAUSE delay1
    endif
    CONT=0
    CONT1=CONT1+1
    if cont1>3 then
      LCDOUT $FE,1,"Alarma Activada"
      cont1=0
      alarma=1
      flag3=0
      flag5=1
      PAUSE delay1
    endif
  case 14
    CONT=0
end select
wend
GOTO Inicio

BARRIDO:
  F1=0
  IF C1 = 0 THEN TECLA =1 :RETURN
  IF C2 = 0 THEN TECLA =2 :RETURN
  IF C3 = 0 THEN TECLA =3 :RETURN
  IF C4 = 0 THEN TECLA =10 :RETURN

```

```

    F1=1
    F2=0
    IF C1 = 0 THEN TECLA =4 :RETURN
    IF C2 = 0 THEN TECLA =5 :RETURN
    IF C3 = 0 THEN TECLA =6 :RETURN
    IF C4 = 0 THEN TECLA =11 :RETURN
    F2=1
    F3=0
    IF C1 = 0 THEN TECLA =7 :RETURN
    IF C2 = 0 THEN TECLA =8 :RETURN
    IF C3 = 0 THEN TECLA =9 :RETURN
    IF C4 = 0 THEN TECLA =12 :RETURN
    F3=1
    F4=0
    IF C1 = 0 THEN TECLA =14:RETURN
    IF C2 = 0 THEN TECLA =0 :RETURN
    IF C3 = 0 THEN TECLA =15:RETURN
    IF C4 = 0 THEN TECLA =13 :RETURN
    F4=1
    pauseus 500
return

```

```

REBOTE:
  ESPERAR:
  IF C1 = 0 THEN ESPERAR
  IF C2 = 0 THEN ESPERAR
  IF C3 = 0 THEN ESPERAR
  IF C4 = 0 THEN ESPERAR
  PAUSEus 500
RETURN

```

```

Sensores:
if flag6=0 then
  If Sm1=1 then
    bk=1
    D1=100
    D2=10
    D3=50
    D4=10
    Alarma=1
    Lcdout $fe, 1          'Limpia la pantalla
    Lcdout "S.Magnético1 ON "
    gosub Llamadas
  endif
endif
if flag6=0 then
  If Sm2=1 then
    bk=1
    D1=150
    D2=10
    D3=80
    D4=10
    Alarma=1
    Lcdout $fe, 1          'Limpia la pantalla
    Lcdout "S.Magnetico2 ON "
    gosub Llamadas
  endif
endif
if flag6=0 then

```

```
If Sm3=1 then
  bk=1
  D1=100
  D2=10
  D3=50
  D4=10
  Alarma=1
  Lcdout $fe, 1      'Limpia la pantalla
  Lcdout "S.Magnético3 ON "
  gosub Llamadas
endif
endif
if flag6=0 then
  If Sm4=1 then
    bk=1
    D1=150
    D2=10
    D3=80
    D4=10
    Alarma=1
    Lcdout $fe, 1      'Limpia la pantalla
    Lcdout "S.Magnético4 ON "
    gosub Llamadas
  endif
endif
if flag6=0 then
  If Sm5=1 then
    bk=1
    D1=100
    D2=10
    D3=50
    D4=10
    Alarma=1
    Lcdout $fe, 1      'Limpia la pantalla
    Lcdout "S.Magnético5 ON "
    gosub Llamadas
  endif
endif
if flag6=0 then
  If Sm6=1 then
    bk=1
    D1=150
    D2=10
    D3=80
    D4=10
    Alarma=1
    Lcdout $fe, 1      'Limpia la pantalla
    Lcdout "S.Magnético6 ON "
    gosub Llamadas
  endif
endif
if flag6=0 then
  If Sm7=1 then
    bk=1
    D1=100
    D2=10
    D3=50
    D4=10
    Alarma=1
```

```

        Lcdout $fe, 1          'Limpia la pantalla
        Lcdout "S.Magnético7 ON "
        gosub Llamadas
    endif
endif
if flag6=0 then
    If Sm8=1 then
        bk=1
        D1=150
        D2=10
        D3=80
        D4=10
        Alarma=1
        Lcdout $fe, 1          'Limpia la pantalla
        Lcdout "S.Magnético8 ON "
        gosub Llamadas
    endif
endif
*****
if flag6=0 then
    If SP1=1 then
        bk=1
        D1=100
        D2=10
        D3=50
        D4=10
        Alarma=1
        Lcdout $fe, 1          'Limpia la pantalla
        Lcdout "S.Presencia1 ON "
        gosub Llamadas
    endif
endif
if flag6=0 then
    If SP2=1 then
        bk=1
        D1=150
        D2=10
        D3=80
        D4=10
        Alarma=1
        Lcdout $fe, 1          'Limpia la pantalla
        Lcdout "S.Presencia2 ON "
        gosub Llamadas
    endif
endif
if flag6=0 then
    If SP3=1 then
        bk=1
        D1=100
        D2=10
        D3=50
        D4=10
        Alarma=1
        Lcdout $fe, 1          'Limpia la pantalla
        Lcdout "S.Presencia3 ON "
        gosub Llamadas
    endif
endif
if flag6=0 then

```

```

If SP4=1 then
  bk=1
  D1=100
  D2=10
  D3=50
  D4=10
  Alarma=1
  Lcdout $fe, 1      'Limpia la pantalla
  Lcdout "S.Presencia4 ON "
  gosub Llamadas
endif
endif
if flag6=0 then
  If SP5=1 then
    bk=1
    D1=100
    D2=10
    D3=50
    D4=10
    Alarma=1
    Lcdout $fe, 1      'Limpia la pantalla
    Lcdout "S.Presencia5 ON "
    gosub Llamadas
  endif
endif
RETURN

TECLASabcd:
flag4=1
while flag4=1
  tecla=255
  gosub barrido
  gosub rebote
  select case TECLA
  case 10
    if flag5=1 then
      flag5=0
      alarma=0
      flag4=0
      ' activar=0
    else
      activar=~activar
      z=Tini2
      alarma=0
      if activar=1 then
        LCDOUT $FE,$80," ACTIVANDO "
        Lcdout $fe,$C0,"ALARMA - ",DEC3 tini2," SEG" *****
        FOR V=1 TO TIni2
          Z=Z-1
          PAUSE Mil      'Pausa de un segundo
          Lcdout $fe, $c0      'Ir a la Segunda línea
          Lcdout "ALARMA - ",DEC3 Z," SEG"
        NEXT V
        PAUSE mil      'Pausa de un segundo
      else
        LCDOUT $FE,$80," DESACTIVANDO "
        Lcdout $fe,$C0," ALARMA " 'Ir a la Segunda línea
        ESPERA=0
        tiempo=retardo
      '
    end case
  end while

```

```

        PAUSE mil          'Pausa de un segundo
    endif
    alarma=0
    TECLA=255
    flag4=0
endif
case 11
    cont2=0
    flag2=1
    while flag2=1
        tecla=255
        if cont2=0 then
            LCDOUT $FE,$80," CAMBIAR TELF.1 "
            LCDOUT $FE,$C0,"          "
            LCDOUT $FE,$C0
        endif
        gosub BARRIDO
        gosub REBOTE
        select case TECLA
            IF CONT2 < 7 and tecla < 10 THEN
                Telefono[CONT2]=TECLA
                LCDOUT #TECLA
                cont2=cont2+1
            ENDIF
        case 15
            if cont2=7 then
                for x=0 to 6
                    write x+8,Telefono[x]
                    pause 10
                next x
                READ 8,A0
                READ 9,A1
                READ 10,A2
                READ 11,A3
                READ 12,A4
                READ 13,A5
                READ 14,A6
                'READ 15,A7
                'READ 16,A8
                LCDOUT $FE,1,"NUEVO TELEFONO1"
                LCDOUT $FE,$c0," GUARDADO OK "
                pause delay1
            else
                LCDOUT $FE,1,"NUEVO TELEFONO1"
                LCDOUT $FE,$c0," NO GRABADO....."
                pause delay1
            endif
        flag2=0
        cont2=0
        FLAG4=0
    case 14
        cont2=0
    end select
wend
case 12
    cont2=0
    flag2=1
    while flag2=1
        tecla=255

```

```

if cont2=0 then
  LCDOUT $FE,$80," CAMBIAR TELF.2 "
  LCDOUT $FE,$C0,"          "
  LCDOUT $FE,$C0
endif
gosub BARRIDO
gosub REBOTE
select case TECLA
  IF CONT2 < 9 and tecla < 10 THEN
    Telefono[CONT2]=TECLA
    LCDOUT #TECLA
    cont2=cont2+1
  ENDIF
case 15
  if cont2=9 then
    for x=0 to 8
      write x+24,Telefono[x]
      pause 10
    next x
    READ 24,B0
    READ 25,B1
    READ 26,B2
    READ 27,B3
    READ 28,B4
    READ 29,B5
    READ 30,B6
    READ 31,B7
    READ 32,B8
    LCDOUT $FE,1,"NUEVO TELEFONO2"
    LCDOUT $FE,$c0," GUARDADO OK "
    pause delay1
  else
    LCDOUT $FE,1,"NUEVO TELEFONO2 "
    LCDOUT $FE,$c0," NO GRABADO....."
    pause delay1
  endif
  flag2=0
  cont2=0
  FLAG4=0
case 14
  cont2=0
end select
wend
case 13
  cont2=0
  flag2=1
  while flag2=1
    tecla=255
    if cont2=0 then
      LCDOUT $FE,$80," CAMBIAR CLAVE "
      LCDOUT $FE,$C0,"          "
      LCDOUT $FE,$C0
    endif
    gosub BARRIDO
    gosub REBOTE
    select case TECLA
      IF CONT2 < 4 and tecla < 10 THEN
        Teléfono[CONT2]=TECLA
        LCDOUT #TECLA

```



```

        cont2=cont2+1
    ENDIF
case 15
    if cont2=4 then
        for x=0 to 3
            write x,Teléfono[x]
            pause 10
        next x
        FOR X=0 TO 3
            READ X,CLAVEDEF[X]
        NEXT X
        LCDOUT $FE,1," NUEVA CLAVE "
        LCDOUT $FE,$c0," GUARDADO OK "
        pause delay1
    else
        LCDOUT $FE,1," NUEVA CLAVE "
        LCDOUT $FE,$c0," NO GRABADO...."
        pause delay1
    endif
    flag2=0
    cont2=0
    FLAG4=0
case 14
    cont2=0
end select
wend
end select
wend
return
Llamadas:
    Lcdout $fe, $c0      'Ir a la Segunda línea
    Lcdout "Llamando TELF.1"
    gosub llamada2
    if flag6=0 then
        Lcdout $fe, $c0      'Ir a la Segunda línea
        Lcdout "PAUSA 5 SEGUNDOS"
        pause 5000
        Lcdout $fe, $c0      'Ir a la Segunda línea
        Lcdout "Llamando TELF.2"
    endif
    gosub llamada1
    if flag6=0 then
        Lcdout $fe, $c0      'Ir a la Segunda línea
        Lcdout "PAUSA 5 SEGUNDOS"
        pause 5000
        Lcdout $fe, $c0,"FIN DE LLAMADAS"
        PAUSE Mil           'Pausa de un segundo
    endif
    Espera=1
' alarma=0
return
Llamada1:
    if flag6=0 then
        gosub Pausa2Seg
    endif
' pause 2000           'espera de 2 segundos antes de empezar
    if flag6=0 then
        LiTef=1           'se conecta a la línea telefónica
    endif

```

```

if flag6=0 then
  gosub Pausa2Seg
endif
' PAUSE 2000      'espera 2 segundo hasta que exista tono de marcar
if flag6=0 then
  DTMFOUT TonoDTMF,[A0,A1,A2,A3,A4,A5,A6] 'número al cual el PIC va a llamar
endif
if flag6=0 then
  gosub Pausa2Seg
endif
if flag6=0 then
  gosub Pausa2Seg
endif
if flag6=0 then
  gosub Pausa2Seg
endif
' pause 4000      'esperar 4 seg. hasta que alguien conteste
if flag6=0 then
  FOR x = 1 to TiempoSonido  'repetir 200 veces, equivale aproximadamente a 48 segundos
    sound TonoDTMF,[D1,D2,D3,D4] 'enviar sonido por el teléfono
  NEXT
endif
LiTef=0          'desconecta el relé, el cual cierra la llamada
return

```

Llamada2:

```

if flag6=0 then
  gosub Pausa2Seg
endif
' pause 2000      'espera de 2 segundos antes de empezar
if flag6=0 then
  LiTef=1         'se conecta a la línea telefónica
endif
if flag6=0 then
  gosub Pausa2Seg
endif
' PAUSE 2000      'espera 2 segundo hasta que exista tono de marcar
if flag6=0 then
  DTMFOUT TonoDTMF,[B0,B1,B2,B3,B4,B5,B6,B7,B8] 'número al cual el PIC va a llamar
endif
if flag6=0 then
  gosub Pausa2Seg
endif
if flag6=0 then
  gosub Pausa2Seg
endif
if flag6=0 then
  gosub Pausa2Seg
endif
' pause 4000      'esperar 4 seg. hasta que alguien conteste
if flag6=0 then
  FOR x = 1 to TiempoSonido  'repetir 200 veces, equivale aproximadamente a 48 segundos
    sound TonoDTMF,[D1,D2,D3,D4] 'enviar sonido por el teléfono
  NEXT
endif
LiTef=0          'desconecta el relé, el cual cierra la llamada
return

```

Pausa2Seg:

```
for pausa=1 to 2000
  tecla=255
  gosub barrido
  gosub rebote
  if tecla>9 and tecla<14 then
    flag6=1
    PAUSA=65535
  endif
next pausa
return
END
```

## CAPITULO 3

### IMPLEMENTACIÓN Y PRUEBAS DE FUNCIONAMIENTO

#### 3.1 IMPLEMENTACIÓN DEL HARDWARE

Una vez terminado el hardware y software del sistema se da comienzo a la implementación de todos los elementos que intervienen en la alarma, tratando de distribuir los sensores en las partes más vulnerables a una incursión ajena.

##### 3.1.1 DISEÑO DE PLANO PARA INSTALACIÓN DE SENSORES Y UNIDAD CENTRAL DEL SISTEMA

Para la instalación de sensores se ha dibujado un plano de la biblioteca en el cual se detalla la localización de cada uno de los sensores en el que sobresalen los de presencia, ya que los sensores magnéticos van a puertas y ventanas por lo que no es necesario tratar de encontrar una localización adecuada y precisa como sucede con los sensores de presencia.

En el diseño de instalación de los sensores de presencia se ha tomado en cuenta la distancia máxima y altura a la que los sensores pueden ser instalados para detectar un movimiento de acuerdo con sus características técnicas, teniendo en cuenta estos aspectos se pasa a buscar los lugares en donde la incursión ajena podría ser más eminente. Es así que en el Plano 1 ubicado en el ANEXO B se puede apreciar con claridad cómo están distribuidos los sensores.

**Tabla 3-1.** Distribución de sensores en la biblioteca.

SENSORES	UBICACIÓN
Sensor de presencia 1	Sala de lectura (Adultos)
Sensor de presencia 2	Entrada principal
Sensor de presencia 3	Sala de lectura (Niños)
Sensor de presencia 4	Salón de juegos
Sensor de presencia 5	Sala de sesiones
Sensor magnético 1	Ventana en la sala de lectura
Sensor magnético 2	Puerta de ingreso a la sala de lectura
Sensor magnético 3	Puerta principal de ingreso a la biblioteca
Sensor magnético 4	Ventana en el salón de juegos
Sensor magnético 5	Puerta de salida al balcón

Para ubicar la unidad central del sistema lo que se ha hecho es buscar un lugar estratégico y accesible para su fácil uso, teniendo en cuenta que en el lugar a instalarse no dificulte las labores diarias que se realizan en este establecimiento.

### 3.1.2 INSTALACIÓN DE CANALETA Y CABLE PARA SENSORES

Una vez lista la ubicación de todos y cada uno de los sensores se procede con la instalación de canaletas plásticas DEXSON autoadhesivas, tomando en cuenta el número de cables que van a pasar por su interior y la ubicación de la unidad central del sistema.

En el Plano 2 en el ANEXO B podemos apreciar el camino de las canaletas por las cuales van a pasar los cables desde cada uno de los sensores alrededor de la biblioteca hasta donde va a ser instalada la unidad central del sistema.

### 3.1.3 INTALACIÓN DE SENSORES

Una vez lista la instalación de cable se procede a instalar los sensores según la ubicación hecha en el Plano 1 de diseño de instalación.

#### 3.1.3.1 Instalación de sensores de presencia.

Los sensores de presencia van conectados mediante un cable multipar de 2 pares categoría 3, de los cuales un par es utilizado para alimentación de 12v DC y el otro par está conectado al contacto normalmente cerrado del sensor como se puede apreciar en la Figura 3-1.



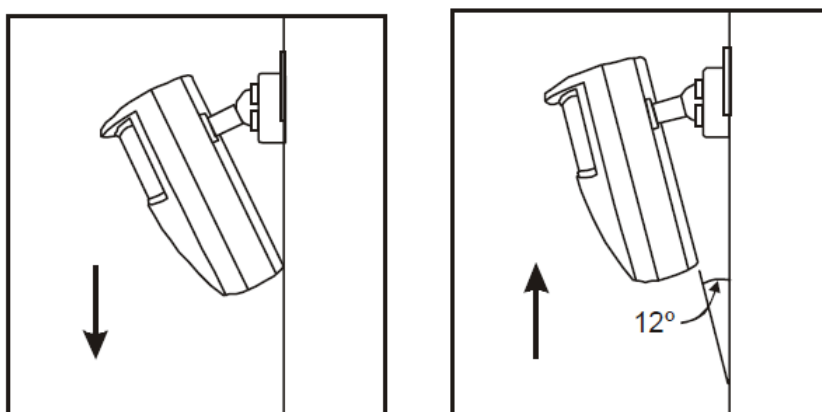
**Figura 3-1.** Conexión del sensor de presencia.

La sensibilidad del sensor de presencia a instalarse es ajustable y tiene tres opciones, para este proyecto se seleccionó en la opción normal como se indica en la figura 3-2.



**Figura 3-2.** Selección de la sensibilidad en el sensor

Se recomienda montar el sensor a una altura de 5 a 6 pies y el ángulo del sensor para controlar la distancia de detección que se requiere e indica en el Plano 3 se puede variar de la siguiente manera; para reducir el rango de detección simplemente se mueve el sensor hacia abajo y hacia arriba para incrementar el rango, incrementando el ángulo cada vez por unos 12 grados como se muestra en la Figura 3-3. Este proceso incrementará al máximo el rango de detección.



**Figura 3-3.** Movimientos para variar el rango de detección del sensor.

El sensor no debe de ser instalado directamente en frente a la luz del sol y/o instalado cerca de aparatos que emita calor o frío lo cual pueda activar falsamente el sensor (ejemplo: unidades de aire acondicionado, salidas de aire caliente, ventiladores, hornos, calentadores, etc.)

### 3.1.3.2 Instalación de sensores magnéticos

Para la instalación de los sensores magnéticos en puertas y ventanas se ha utilizado cable gemelo 22 AWG y se lo debe hacer tomando en cuenta que la parte del switch va a la central y el imán actuador va pegado a la parte de la puerta o ventana que se abre, es así que si el sistema detecta la apertura de cualquiera de estos sensores como se muestra en la Figura 3-4 la alarma será activada.

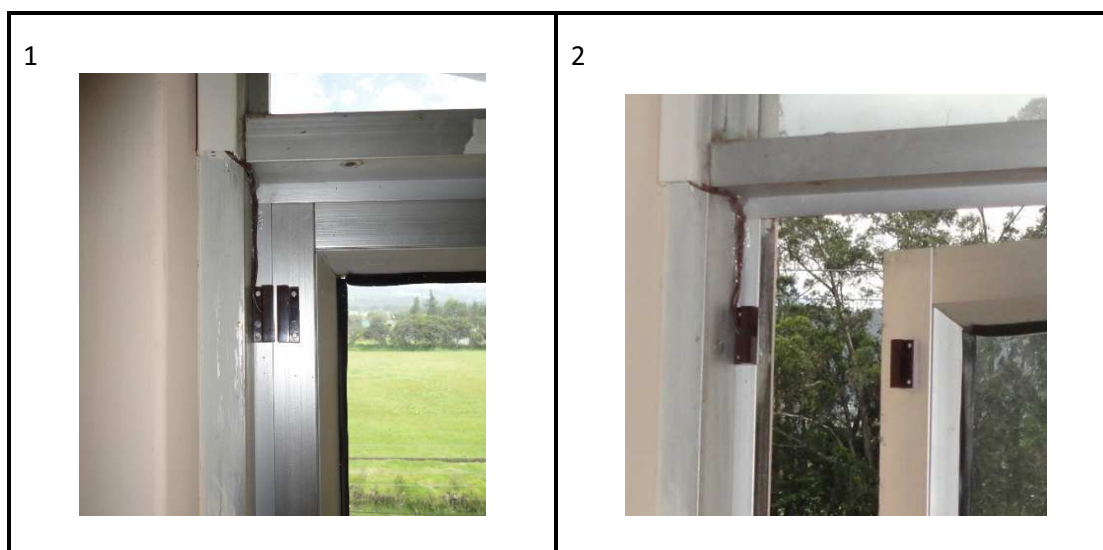


Figura 3-4. Instalación de sensor magnético

### 3.1.4 INSTALACIÓN DE SIRENA Y CABLE TELEFÓNICO

La sirena está ubicada en la parte exterior de la biblioteca con la bocina apuntando a la calle principal para que en caso de robo suene claro y fuerte dando un aviso al vecindario para un pronto auxilio, esta sirena no necesita protección o ser instalada en una caja metálica ya que se encuentra en el tercer piso bajo cubierta y no existe manera de acceder a ésta para su destrucción.

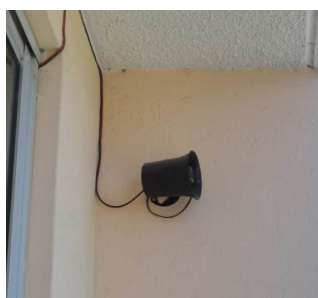
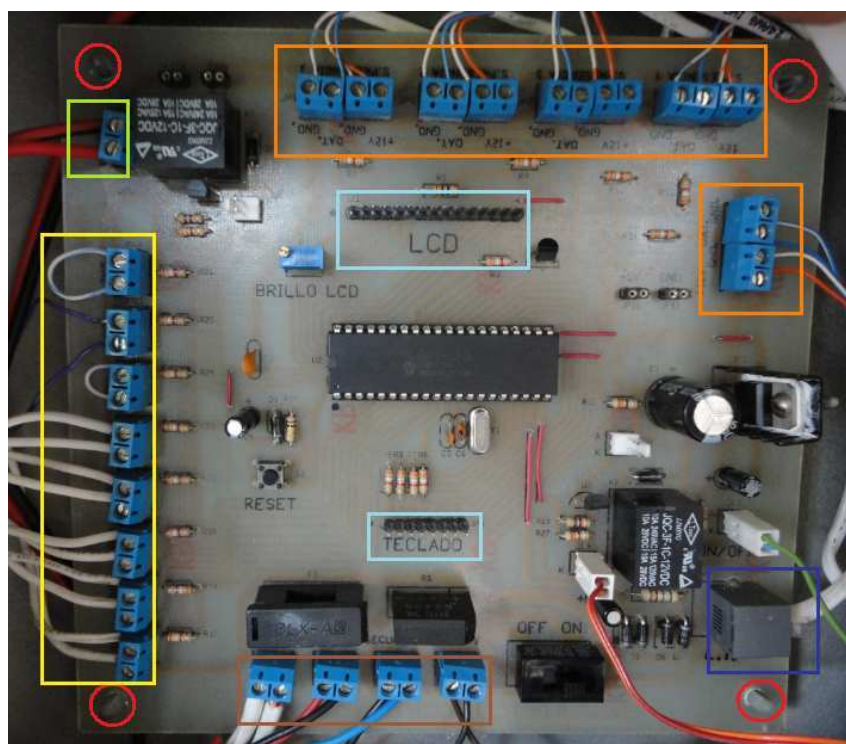


Figura 3-5. Sirena

El cable telefónico va instalado individualmente con una canaleta su recorrido se lo puede apreciar en el Plano 2 en el ANEXO B.

### 3.1.5 INSTALACIÓN DE LA UNIDAD CENTRAL DEL SISTEMA

La unidad central va instalada en una caja metálica de 25x25x10 cm<sup>3</sup> fijada a la pared a una altura de 1,65 m impidiendo así la fácil manipulación de los niños que acuden a este centro, después se coloca la placa electrónica aislada 1cm aproximadamente de la superficie metálica con unos soportes plásticos para evitar cortocircuitos, una vez lista la placa se comienza a conectar a esta unidad los cables de la sirena, teléfono y de todos los sensores.

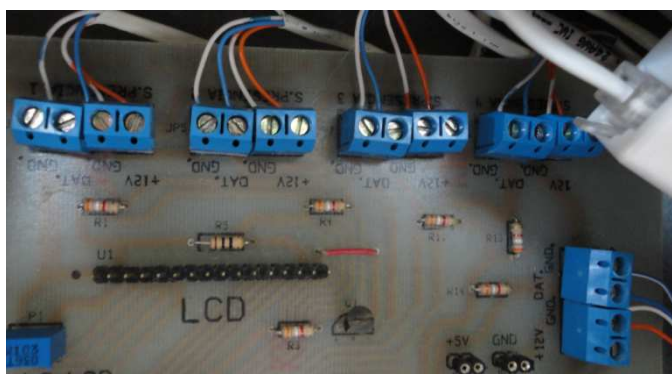


**Figura 3-6.** Placa electrónica de la alarma donde se indica la conexión de los diferentes elementos del sistema.

	Soportes plásticos
	Conexión de sensores magnéticos
	Conexión de sensores de presencia
	Conexión e ingreso de voltaje
	Conexión del cable telefónico
	Socatos de conexión para bus de datos
	Conexión de sirena



Los sensores se deben conectar según la enumeración existente en la placa y de acuerdo a la distribución indicada en el plano, los sensores de presencia se conectan en las borneras teniendo en cuenta los colores de cada cable de acuerdo con la conexión interna que se realizó en la instalación de los sensores como se muestra en la siguiente Figura 3-7.



	PLACA	SENSOR
Naranja	12v	12v
Blanco/Naranja	GND	GND
Azul	DAT	C
Blanco/Azul	GND	NC

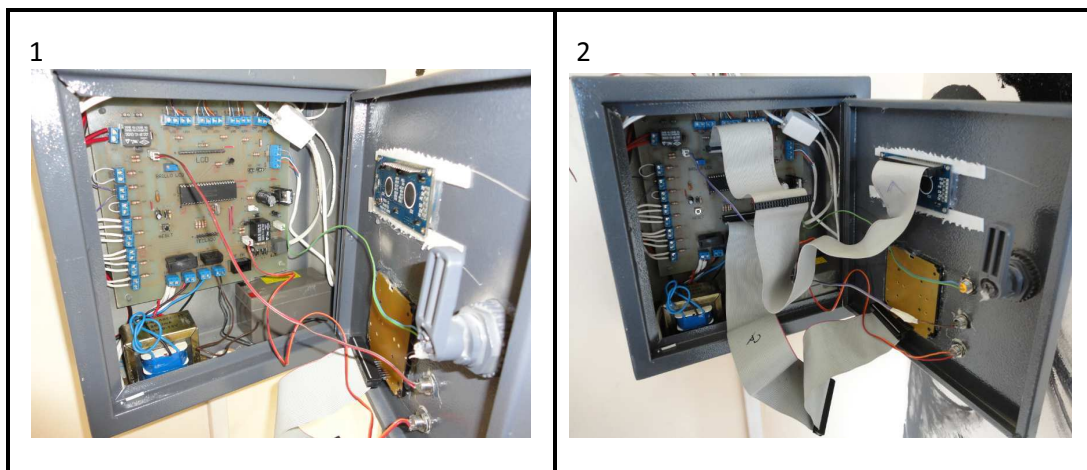
**Figura 3-7.** Conexión del sensor de presencia a la unidad central

En el caso de los sensores magnéticos la conexión es simple como estos son switches no tienen polaridad o forma de conexión se los conecta como en la Figura 3-8, en la placa existe espacio para ocho sensores magnéticos de los cuales usamos solo cinco, los tres restantes se los puentea para simular el switch cerrado y así la alarma no dará un falso aviso.



**Figura 3-8.** Conexión de los sensores magnéticos a la unidad central.

A la placa también van conectados dos buses de datos los mismos que servirán para instalar en la tapa de la caja metálica el LCD y el teclado respectivamente, con la instalación de estos buses se nos facilita la apertura de caja así como la visualización de la placa electrónica ya que son fáciles de conectar y desconectar.



**Figura 3-9.** Conexión de buses de datos para el LCD y el teclado

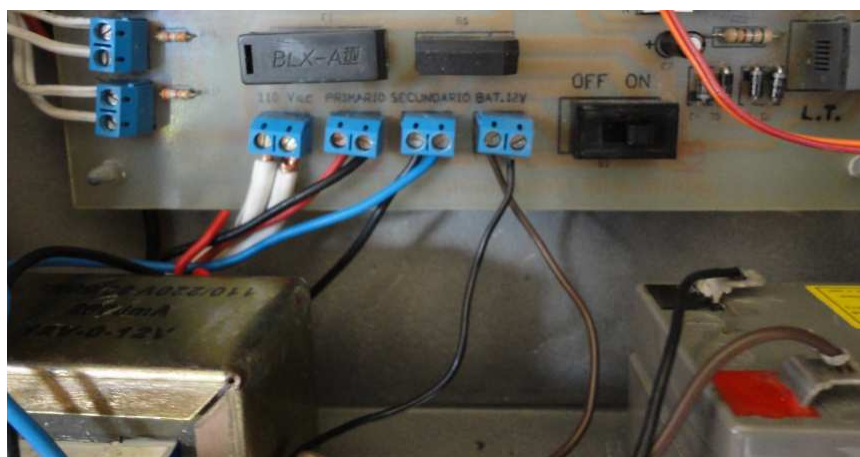
### 3.1.6 ALIMENTACIÓN DE VOLTAJE AL SISTEMA

El sistema necesita ser alimentado con 110v AC para lo cual se ha tratado de buscar el lugar más cercano de donde se pueda obtener dicho voltaje encontrando un tomacorriente y haciendo las adecuaciones necesarias se logro ingresar a la unidad central el voltaje alterno tal y como se muestra en la Figura 3-10.



**Figura 3-10.** Ingreso de voltaje alterno a la unidad central desde un tomacorriente cercano.

Como se puede observar en la Figura 3-11 el sistema también es alimentado con una batería que en caso de falla eléctrica entra en funcionamiento, al conectar la alimentación de voltaje en sus respectivas borneras el sistema está listo para ser encendido con el switch de dos posiciones el cual activa el ingreso de voltaje hacia cada una de las etapas ya descritas anteriormente.



**Figura 3-11.** Conexión para la alimentación del sistema y switch de encendido.

## 3.2 PRUEBAS DE FUNCIONAMIENTO DEL SISTEMA

Al finalizar la implementación del sistema de seguridad, se realizó una serie de pruebas, para verificar el correcto funcionamiento del mismo.

### 3.2.1 ENCENDIDO E INICIALIZACIÓN DEL SISTEMA

Al conectar la unidad central a la línea telefónica y presionando su interruptor de encendido, se mostrará en el LCD un mensaje indicando el encendido de la alarma entonces comenzará la inicialización de sensores, es decir el sistema realiza el sondeo a todos los sensores para luego mostrar el mensaje de ALARMA DESACTIVADA, también tenemos leds, los cuales nos indican cuando la alarma está encendida, cuando la sirena está sonando y cuando está llamando, este proceso se muestra en la Figura 3-12.

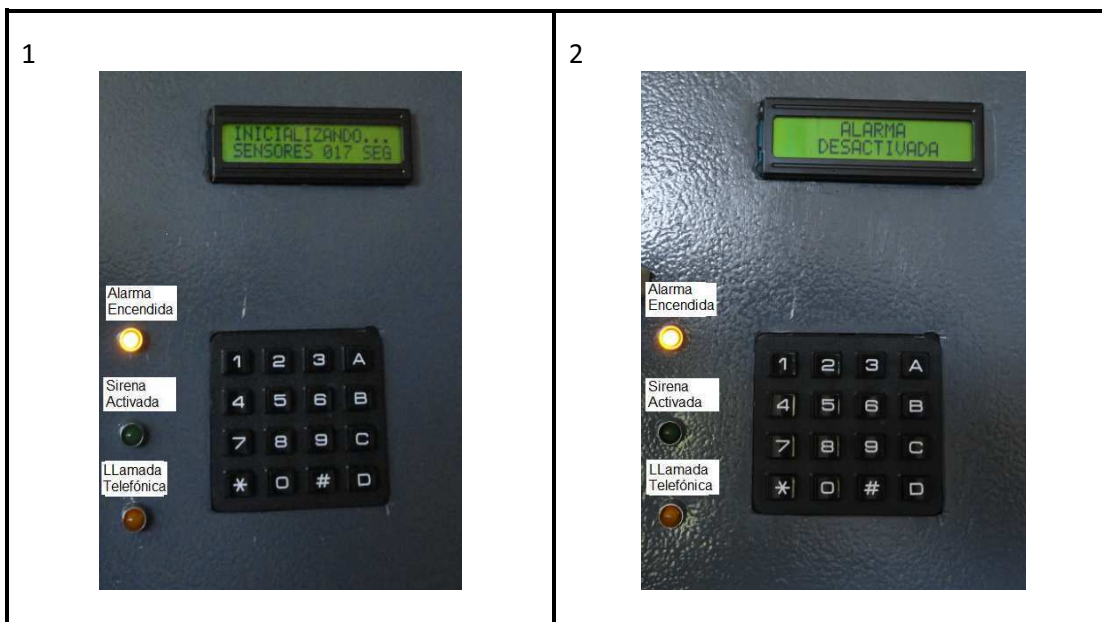
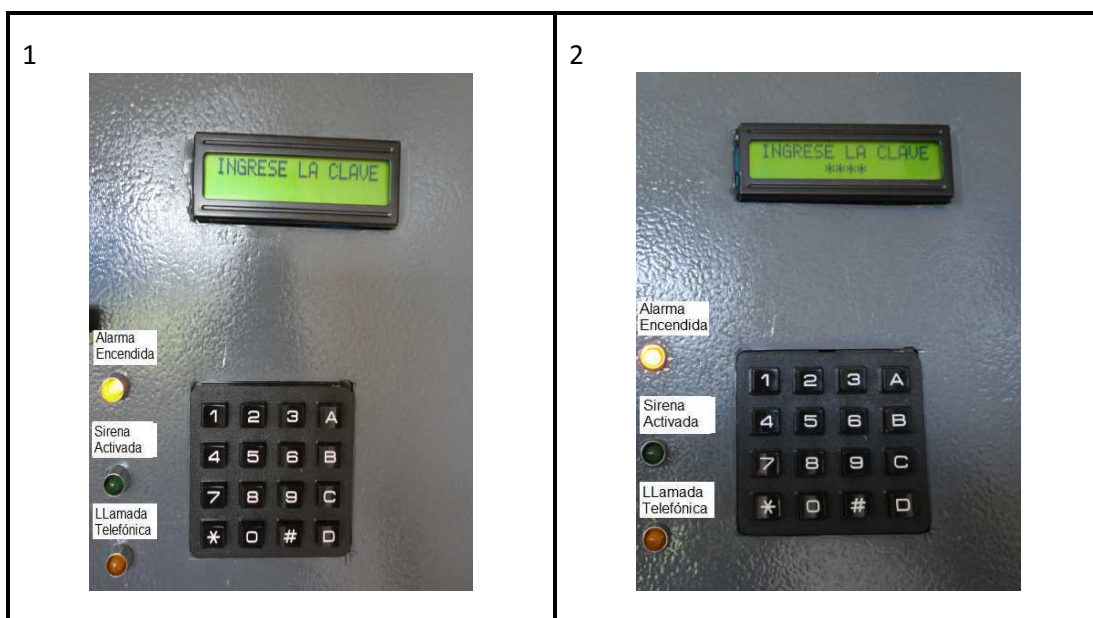


Figura 3-12. Encendido del sistema

A través del teclado se puede acceder al menú principal del sistema presionando la tecla A, en donde el sistema nos pedirá la clave de acceso, la cual debe ser de cuatro dígitos y debe ser la correcta dicho proceso se muestra en la Figura 3-13.





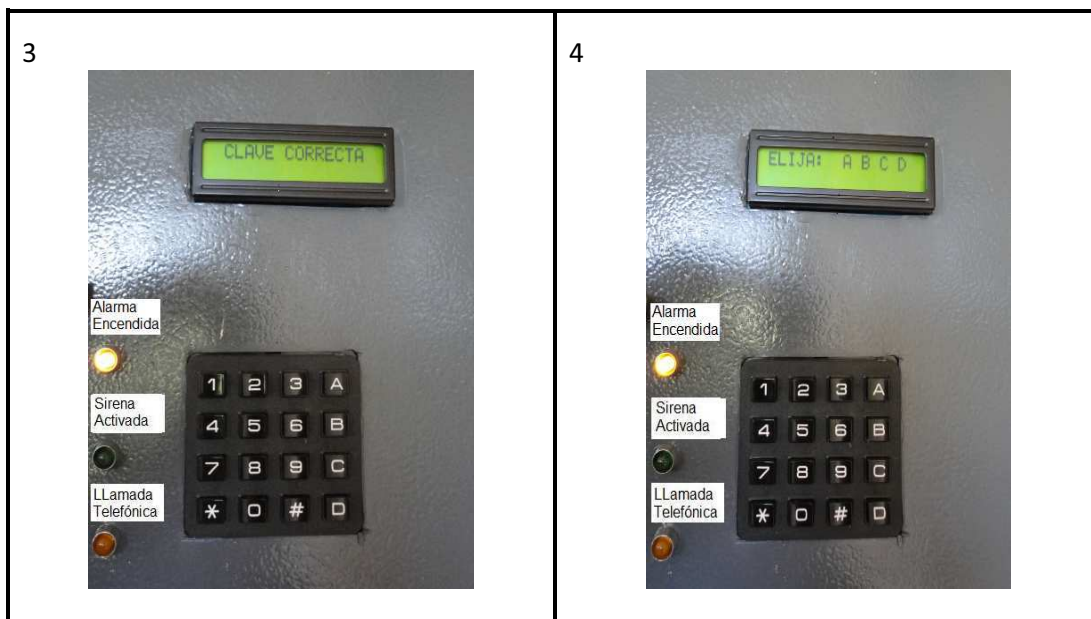
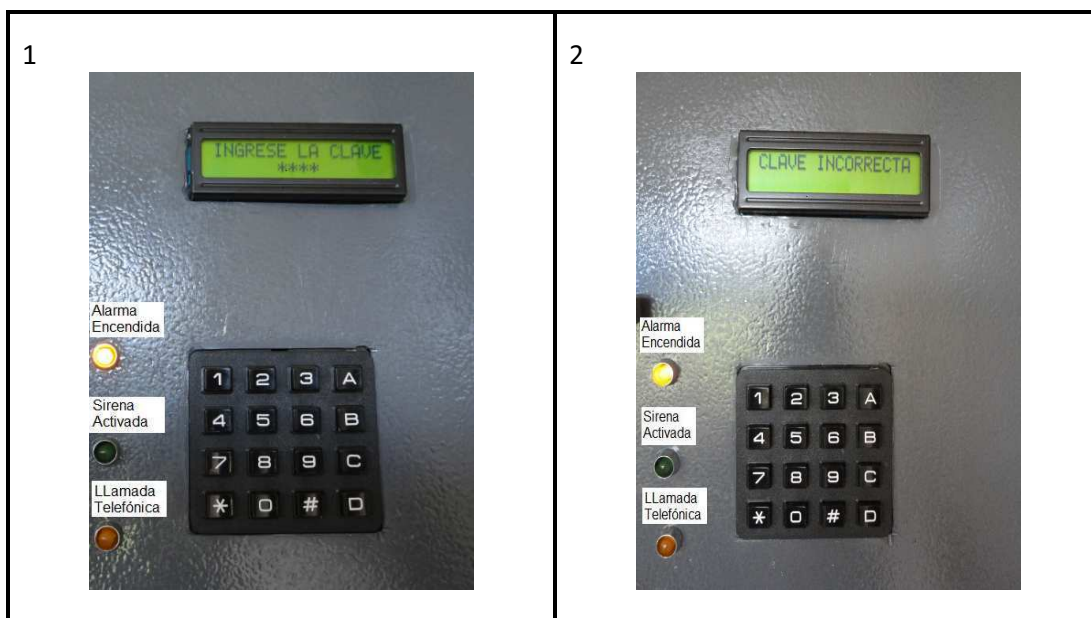
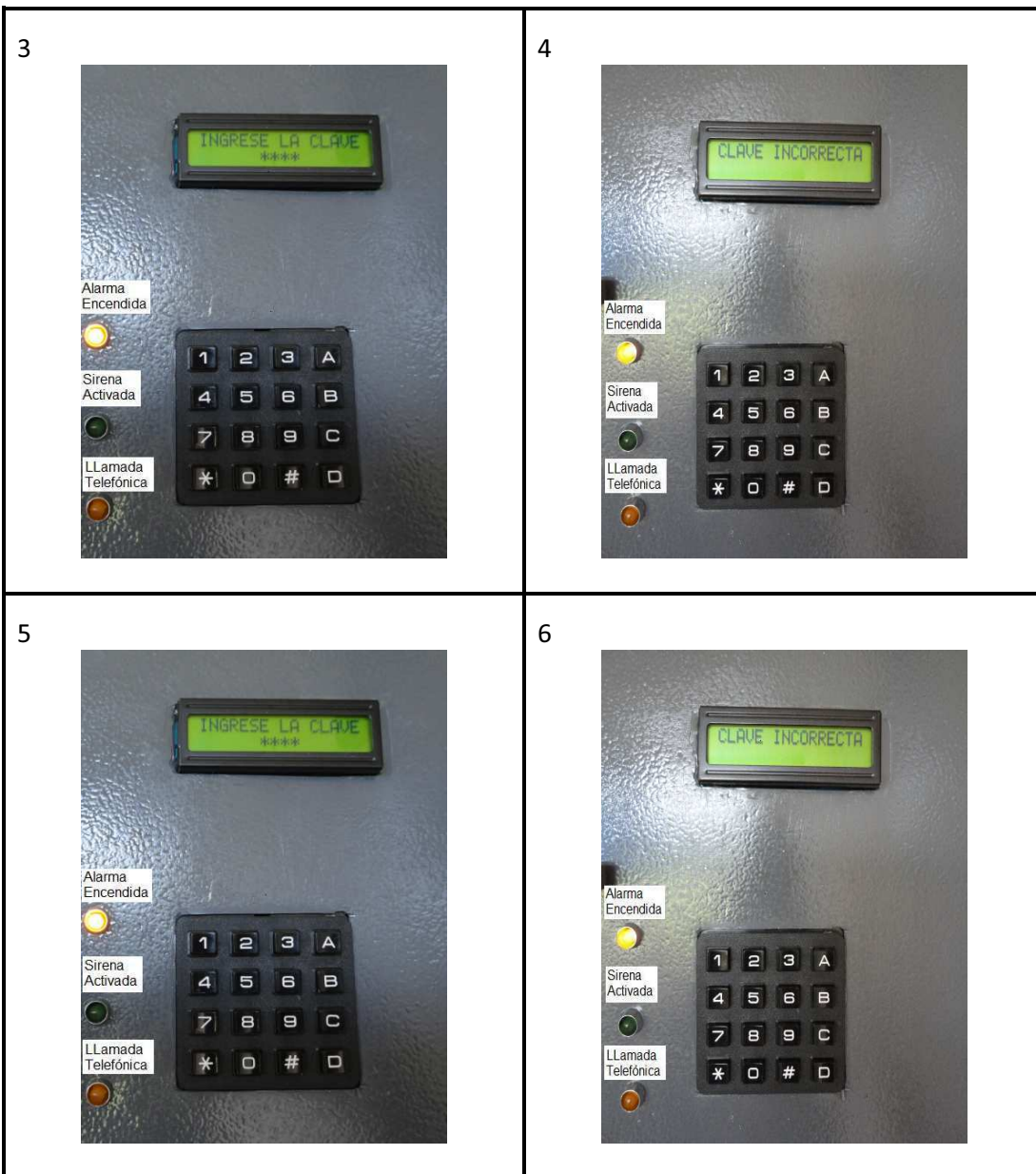


Figura 3-13. Proceso de ingreso de clave para acceder al menú principal.

En caso de que no se ingrese la clave correcta, el sistema proporcionará tres oportunidades para ingresar la clave de cuatro dígitos, caso contrario la alarma será activada como se puede apreciar en la siguiente Figura 3-14.





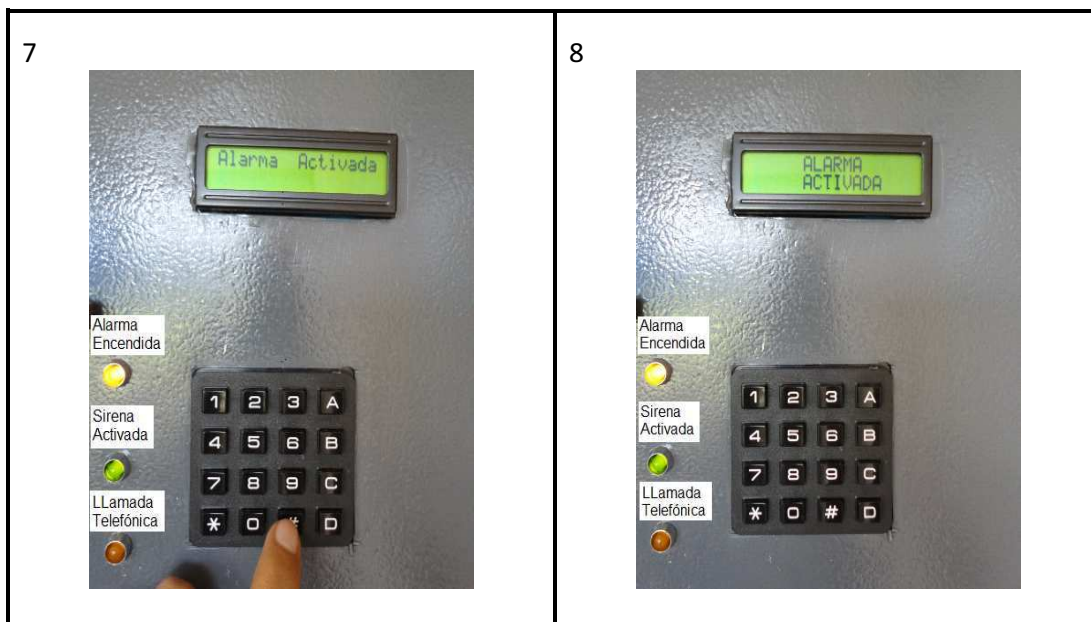


Figura 3-14. Prueba de un mal ingreso de clave.

### 3.2.2 MENU DEL SISTEMA

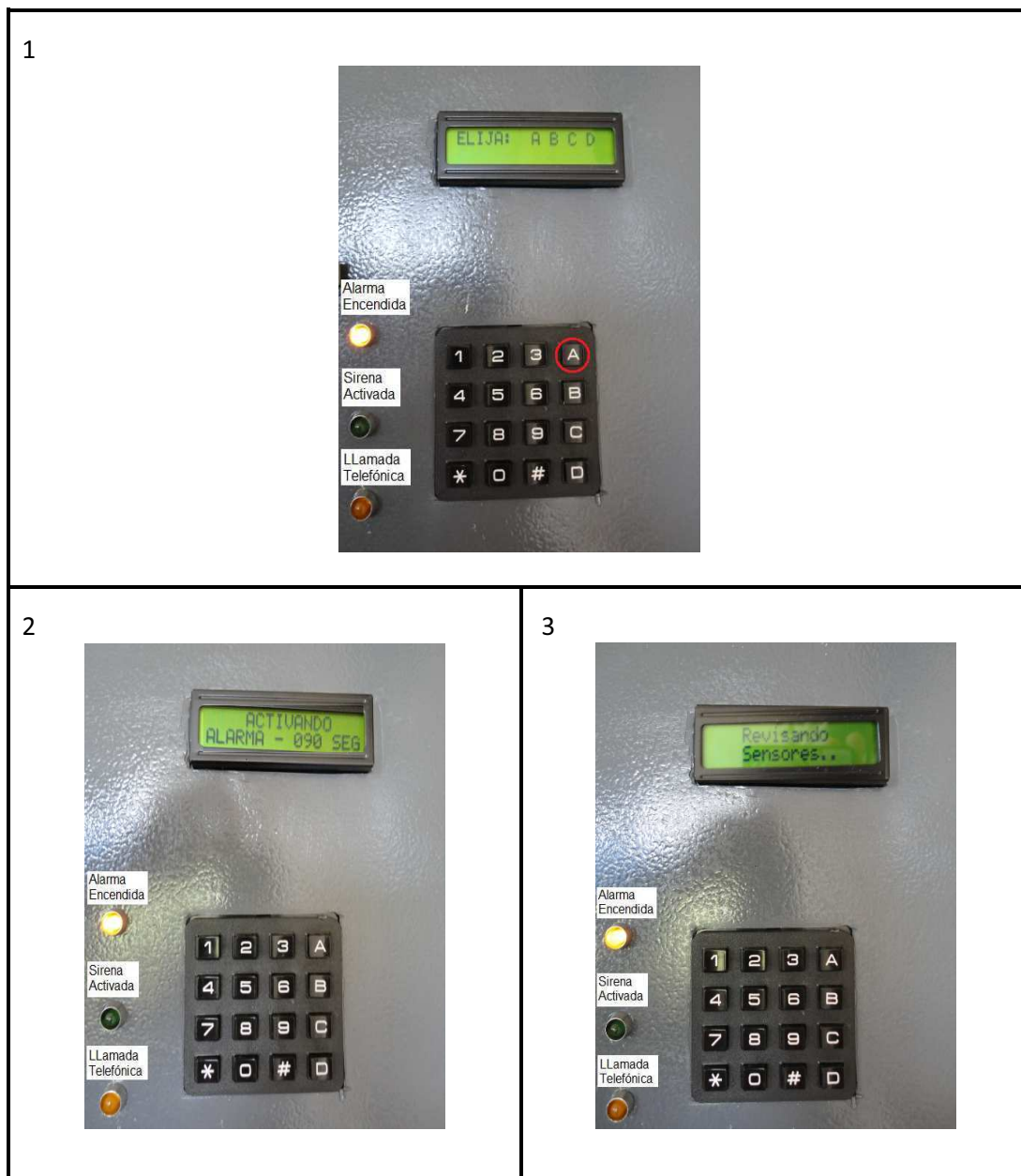
Al ingresar la clave correctamente, en la pantalla del LCD aparecerá el menú con cuatro opciones como se muestra en la Figura 3-15, las cuales nos permitirán hacer cambios en el sistema.



Figura 3-15. Menú principal

#### 3.2.2.1 Opción A

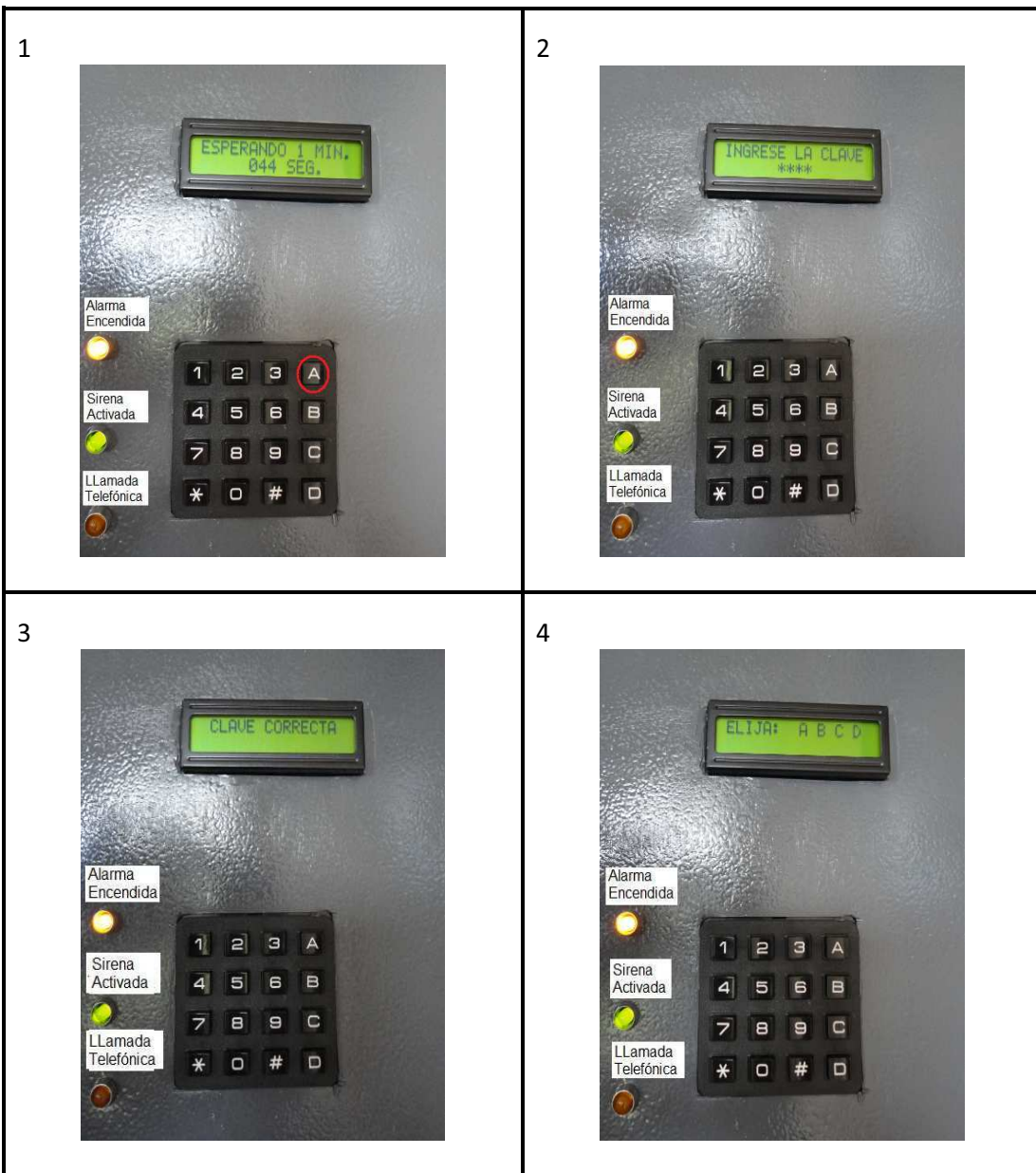
Al presionar la tecla A es la opción de activación/desactivación de la alarma, para activar, el sistema procede a hacer un conteo regresivo de 90 segundos para que el usuario revise puertas y ventanas antes de salir de la biblioteca, después comenzará a revisar los sensores en busca de algún sensor activado. Este proceso se muestra en la Figura 3-16.



**Figura 3-16.** Opción A del menú principal para activar la alarma.

Para cancelar el proceso de activación se presiona la tecla A cuando finalice la segunda llamada, enseguida en el LCD se mostrará el mensaje de ingreso de clave para acceder al menú y desactivar la alarma presionando la tecla A, en todo este transcurso la sirena sonará pero las llamadas serán canceladas como nos indican los leds. Este proceso se muestra en la figura 3-17.





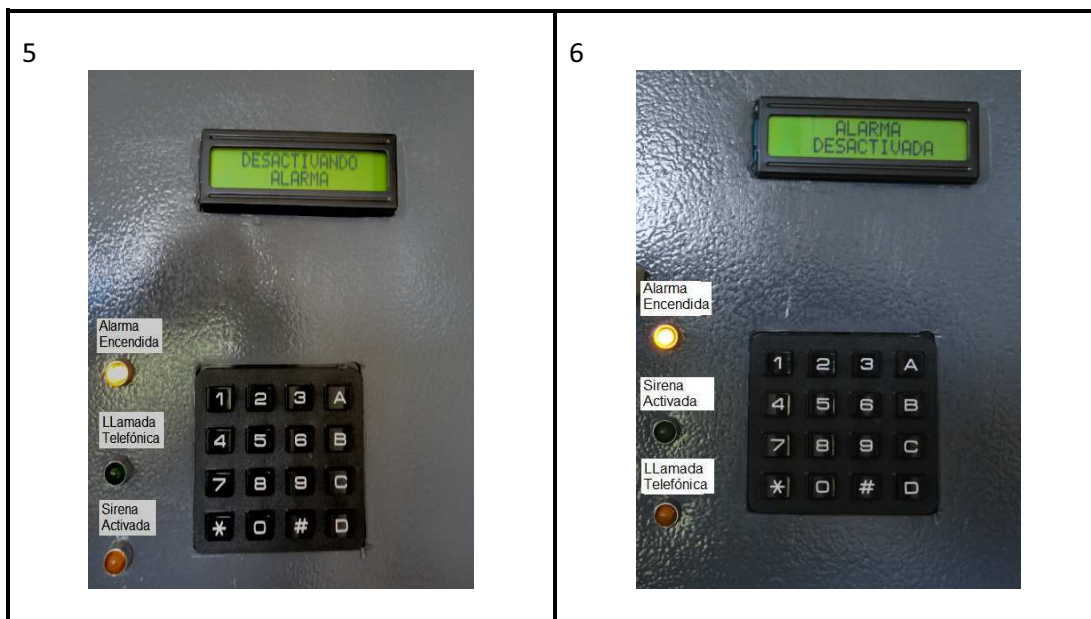
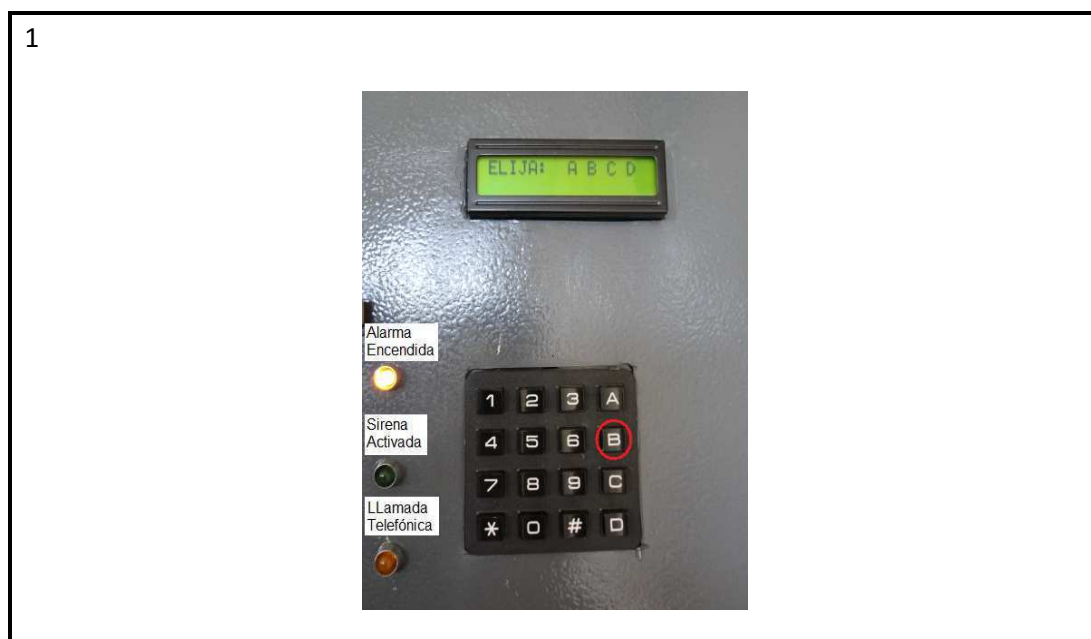


Figura 3-17. Opción A del menú principal para desactivar la alarma.

### 3.2.2.2 Opción B

Al presionar la tecla B, aparece en la pantalla LCD la opción para ingresar un número telefónico fijo mediante el teclado, al cual el sistema va a realizar una llamada en caso de robo, el número debe ser de 7 dígitos para que sea grabado correctamente, pero si el número es incorrecto no será grabado, este proceso se lo muestra en la Figura 3-18.



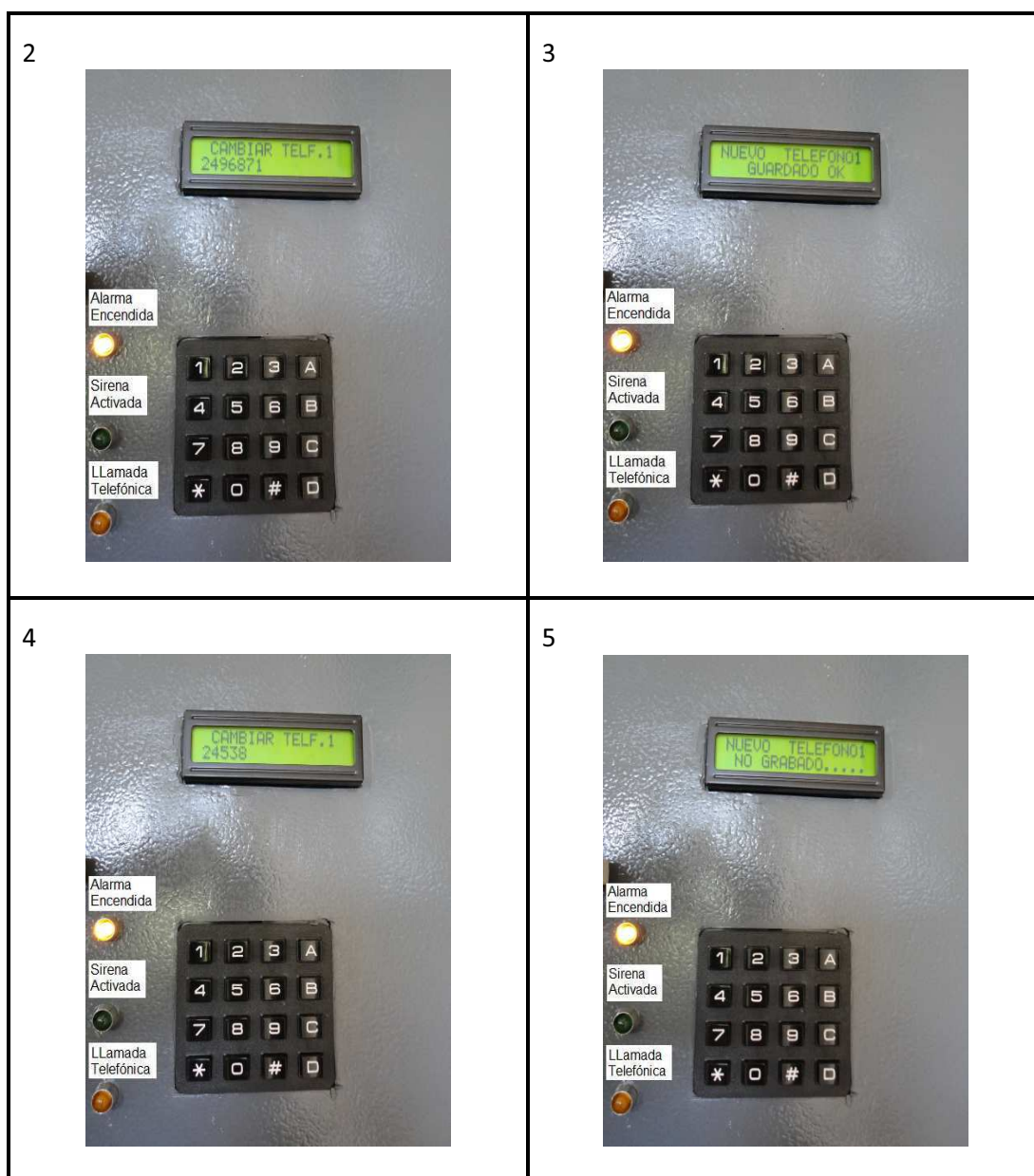


Figura 3-18. Ingreso de un nuevo número telefónico fijo.

### 3.2.2.3 Opción C

Al presionar la tecla C el procedimiento es el mismo que para la opción B pero en este caso se ingresa un número telefónico móvil el cual debe tener 9 dígitos y será grabado correctamente caso contrario no, como se muestra en la Figura 3-19.

1



2



3





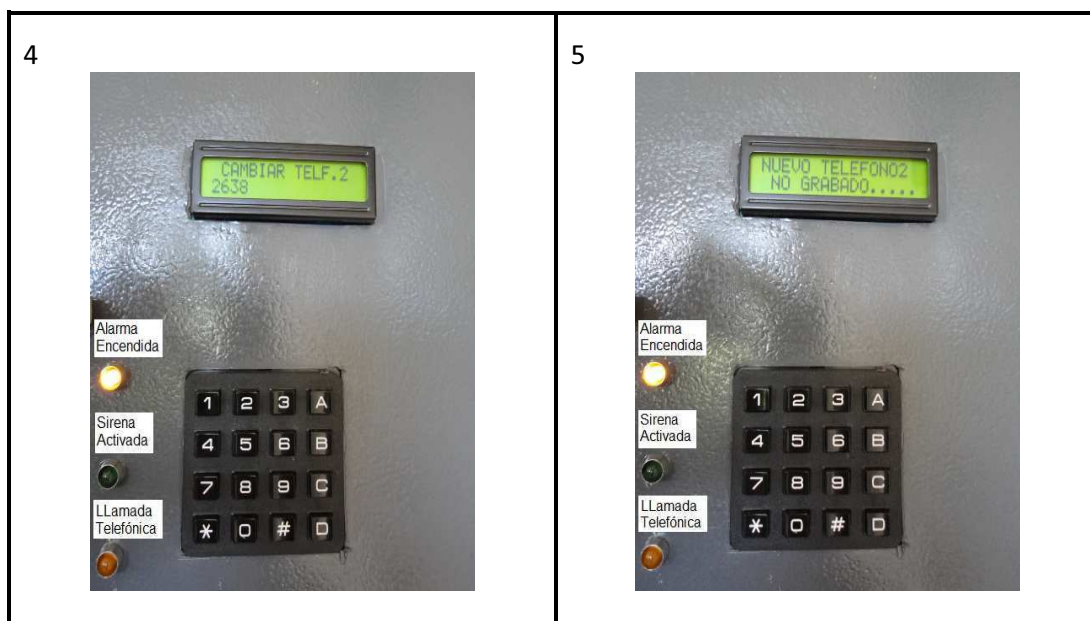
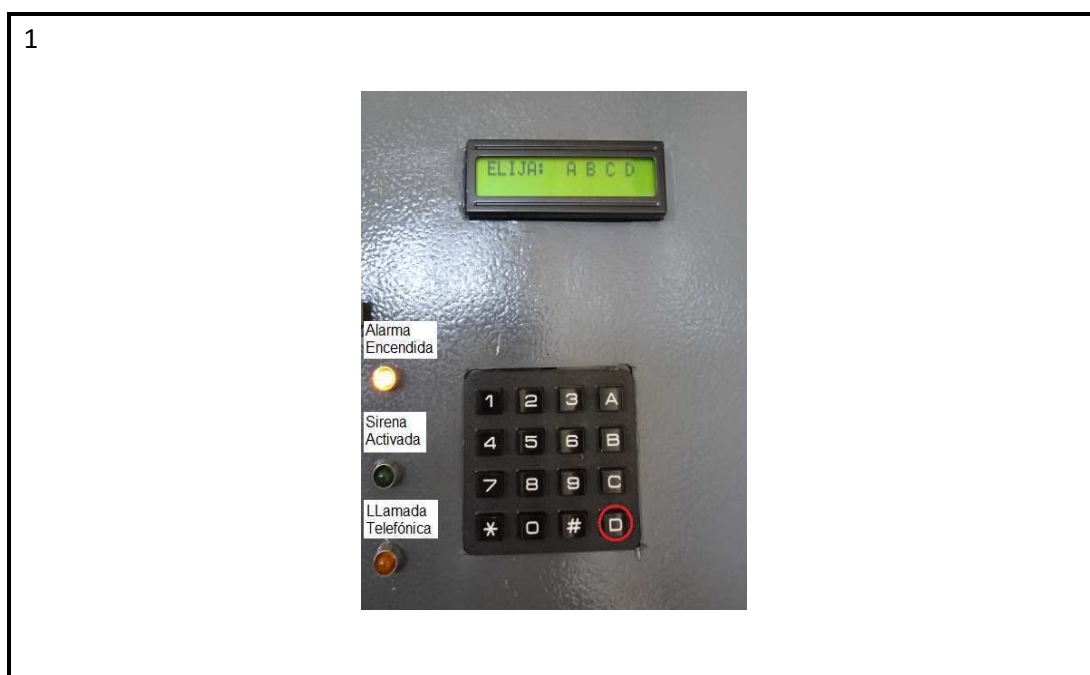


Figura 3-19. Ingreso de un nuevo número telefónico móvil.

#### 3.2.2.4 Opción D

Presionando la tecla D la pantalla LCD muestra la opción de cambio de clave, esta tiene que ser de cuatro dígitos y fácil de recordar para el usuario, así la siguiente vez que se quiera ingresar al sistema, se deberá ingresar la nueva clave almacenada como se muestra en la Figura 3-20.





**Figura 3-20.** Cambio de clave para ingresar al sistema.

### 3.2.3 PRUEBA DE SENSORES

Una vez activada la alarma tal como describimos en la opción A, el sistema se encuentra en total alerta en busca de alguna incursión ajena para dar aviso mediante la llamada telefónica y a su vez activando la sirena. Para que todo este proceso funcione correctamente, se debe comprobar que todos los sensores estén trabajando bien, por eso hacemos la prueba con cada sensor como se muestra en las siguientes Figuras.

**SENSOR DE PRESENCIA 1: SALA DE LECTURA (ADULTOS)**

1



2

**Figura 3-21.** Prueba del Sensor de Presencia 1.

**SENSOR DE PRESENCIA 3: SALA DE LECTURA (NIÑOS)**

1



2

**Figura 3-22.** Prueba de Sensor de Presencia 3.



### SENSOR DE PRESENCIA 2: INGRESO A LA BIBLIOTECA



1



2



Figura 3-23. Prueba de Sensor de Presencia 2.

**SENSOR DE PRESENCIA 4: SALÓN DE JUEGOS**

1



2

**Figura 3-24. Prueba de Sensor de Presencia 4**

### SENSOR DE PRESENCIA 5: SALA DE REUNIONES



1



2



Figura 3-25. Prueba de Sensor de Presencia 5.



**SENSOR MAGNÉTICO 1: VENTANA EN LA SALA DE LECTURA**

1



2

**Figura 3-26.** Prueba de Sensor Magnético 1.

**SENSOR MAGNÉTICO 2: PUERTA DE INGRESO A LA SALA DE LECTURA**

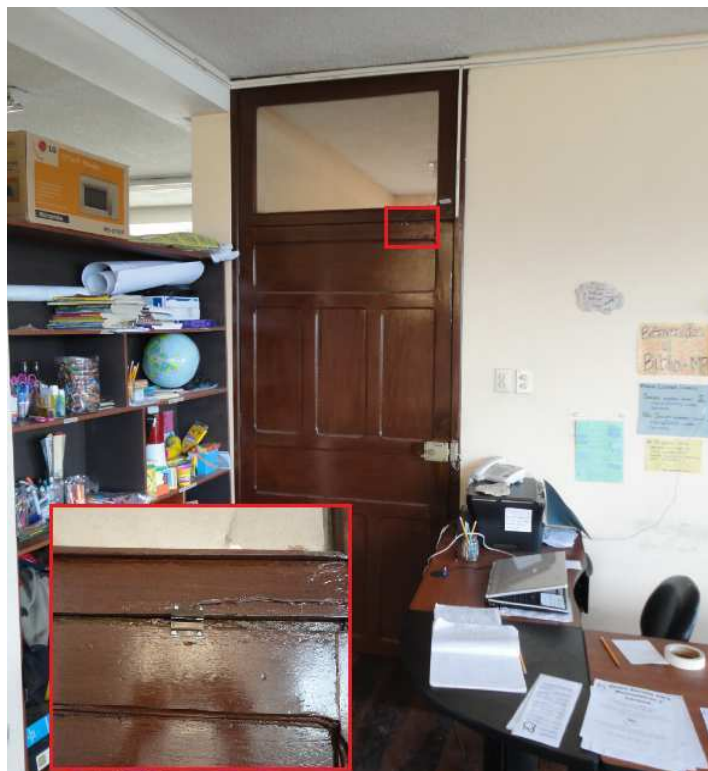
1



2



Figura 3-27. Prueba de Sensor de Presencia 2.

**SENSOR MAGNÉTICO 3: PUERTA DE INGRESO A LA BIBLIOTECA**

1



2

**Figura 3-28.** Prueba de Sensor Magnético 3.



**SENSOR MAGNÉTICO 4: VENTANA EN EL SALÓN DE JUEGOS**

1



2

**Figura 3-29.** Prueba de Sensor Magnético 4.

**SENSOR MAGNÉTICO 5: PUERTA DE SALIDA AL BALCÓN**

1



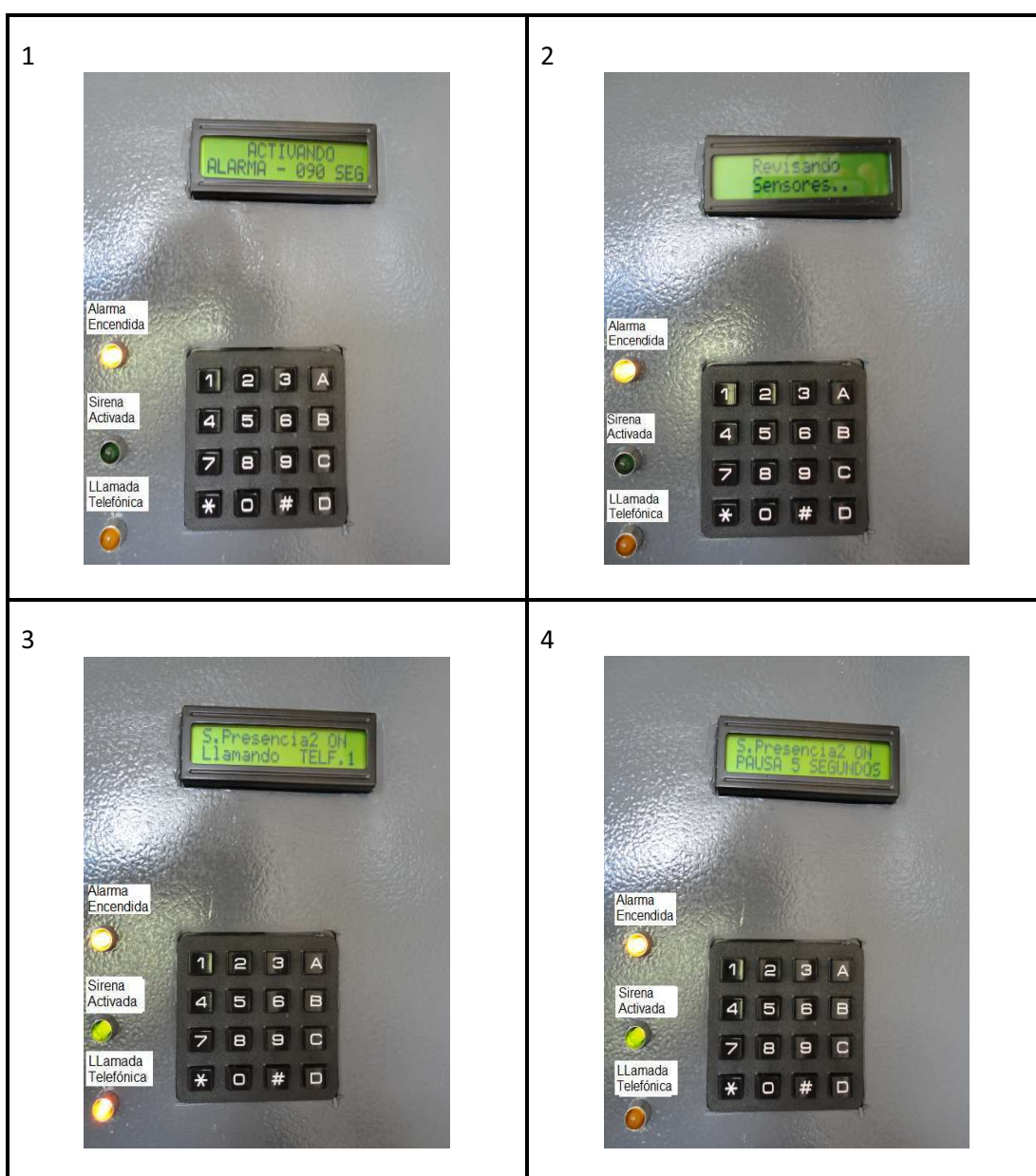
2

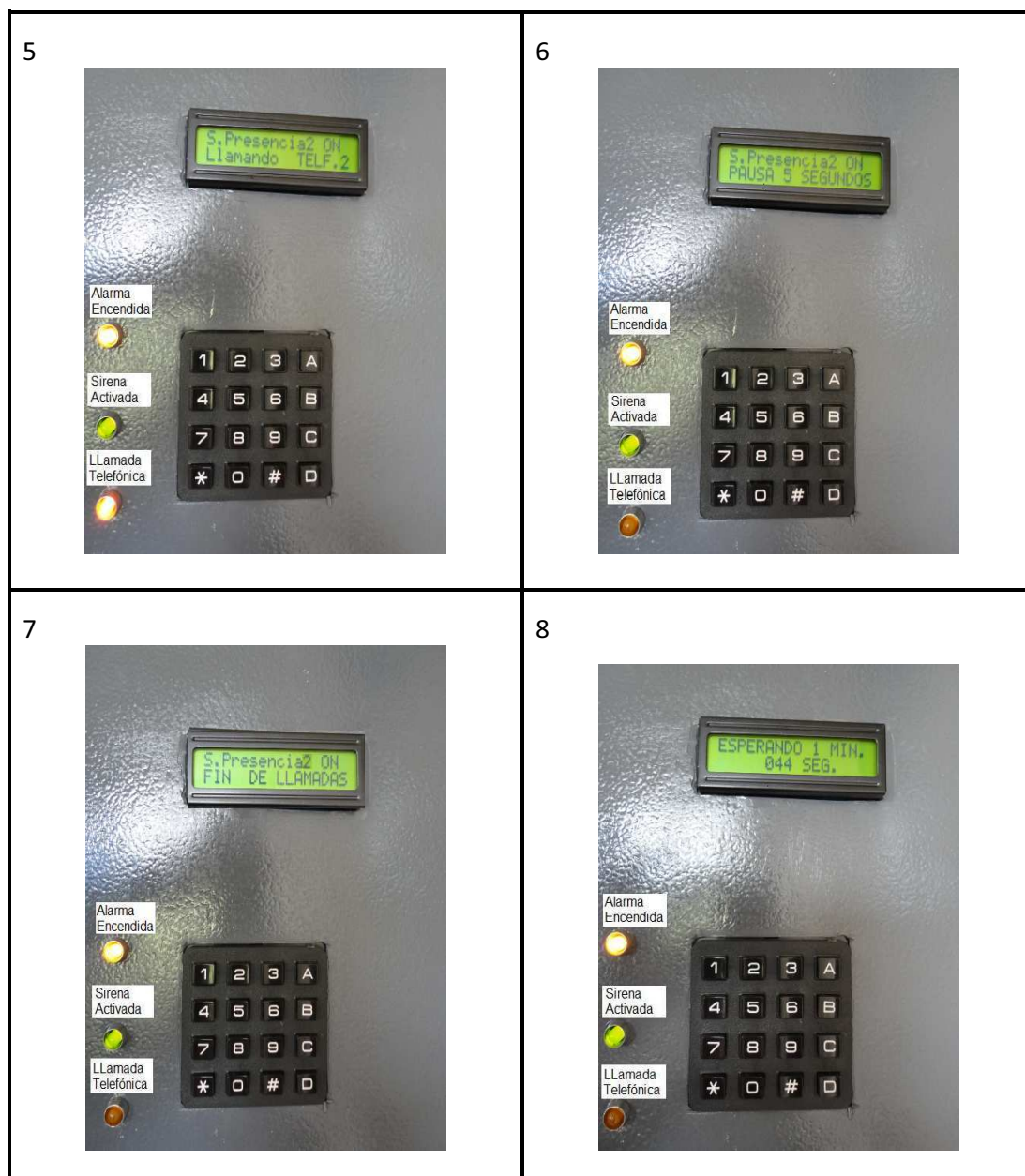
**Figura 3-30.** Prueba de Sensor Magnético 5.



### 3.2.4 PRUEBA DE SIRENA Y LLAMADA TELEFÓNICA

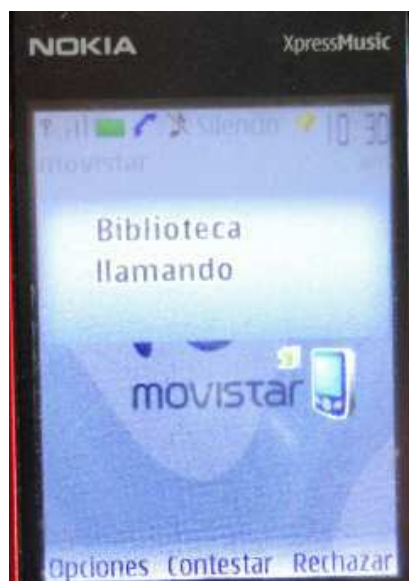
Al ser activado cualquiera de los sensores el sistema se enciende activando la sirena y simultáneamente comienza a llamar a los teléfonos previamente grabados. En la pantalla LCD se muestra que sensor fue activado y el transcurso de las llamadas telefónicas teniendo una pausa de 5 segundos entre cada llamada, al finalizar las llamadas el sistema da una pausa de 1 minuto para luego comenzar a revisar sensores y si alguno de ellos sigue alterado comenzará de nuevo el proceso ya mencionado, a continuación en la siguiente Figura se hace un detalle completo de la alarma activada pero con un sensor ya que para todos los demás el procedimiento es el mismo.





**Figura 3-31.** Proceso y prueba de activación de la alarma.

Cuando la llamada es recibida por el teléfono móvil o fijo escucharemos por el auricular los tonos que indican que la alarma ha sido activada, estos tonos son previamente grabados en el microcontrolador y varían de acuerdo al sensor activado.



**Figura 3-32.** Llamada telefónica

## CAPITULO 4

### CONCLUSIONES Y RECOMENDACIONES

#### 4.1 CONCLUSIONES

El objetivo de este proyecto de tesis es la Construcción e Implementación de una alarma antirrobo con interfaz DTMF de línea telefónica utilizando un microcontrolador, donde se puede concluir lo siguiente

- El sistema implementado cumple correctamente con las tareas señaladas descritas anteriormente como: ingreso de una clave a través de un teclado matricial para evitar falsas alarmas, chequeo interno a través de los sensores para la seguridad, leds indicador de estado, sonido de sirena de 12V DC, llamada telefónica al propietario para tomar las medidas pertinentes.
- Es una excelente alternativa para todos los usuario que no están en capacidad de pagar sistemas antirrobo sofisticados, tales como sistemas de monitoreo a través de cámaras, internet, que a su vez necesitan de un operador.
- El de fácil manejo, hace posible que el usuario sea capaz de operar los distintos dispositivos con toda confianza.
- La construcción e Implementación de una alarma antirrobo con interfaz DTMF de línea telefónica y sirena, es un elemento de seguridad pasiva, ya que no evita una intrusión, pero sí es capaz de advertir de ella, cumpliendo así una función disuasoria frente a posibles intrusos, además reduce el tiempo de ejecución de la intrusión, reduciendo así las pérdidas.
- Con la elaboración del proyecto nos podemos dar cuenta que se puede realizar en el país proyectos tan poco comunes y prácticos a través de la recopilación de información y buscando las necesidades propias y de los demás, tratando de cuidar la economía.

## 4.2 RECOMENDACIONES

- Al colocar el sensor de presencia se debe tomar muchas consideraciones tales como: no se debe colocar el detector de tal forma que reciba en forma directa luz solar ni frente a ventanas ni espejos, ya que puede detectar presencias o sombras innecesarias para la activación de la alarma, además hay que tener en cuenta que el funcionamiento de un sensor de movimiento se ve afectado por la distribución de temperaturas del lugar, por lo que no debe haber corrientes de aire bruscas que activen el sensor y produzca falsas alarmas.
- Para mayor seguridad se recomienda mantener, dos bocinas de alarma. una a la vista, sin presencia de cables de conexión de carácter persuasivo y otra oculta de respaldo de preferencia con una luz ya que ayudará a identificar la fuente de la alarma.
- Para no tener inconveniente al momento de identificar la polaridad de la red telefónica para la conexión hacia el PIC, se coloca un puente de diodos a la entrada de la red, en donde el lado positivo ya está unido a tierra y el lado negativo va a través del filtro hacia el PIC.
- En caso de tener problemas como por ejemplo datos erróneos en la visualización de mensajes en el display se recomienda hacer un RESET al Microcontrolador.
- El manejo de los pulsadores para el ingreso de datos debe ser de manera pausada debido a que necesitamos darle tiempo al microcontrolador de procesar la información de lo contrario visualizaremos valores erróneos en el display.

## BIBLIOGRAFÍA

- [1] “Apuntes de conmutación”  
AULESTIA, Hugo,
- [2] “Componentes de una red de telecomunicaciones”  
<http://www.rootshell.be/~wcruzy/cd/sistel1.pdf>
- [3] “Red de telefonía pública conmutada”  
[http://elsitiodetelecomunicaciones.iespana.es/telefonía\\_basica.htm](http://elsitiodetelecomunicaciones.iespana.es/telefonía_basica.htm)
- [4] “Telefonía fija alámbrica”  
<http://www.supertel.gov.ec/index.php/servicios-de-telecomunicaciones/62-telefonía-fija/>
- [5] “Los teléfonos y su funcionamiento”  
<http://www.cinit.org.mx/articulo.php?idArticulo=7>
- [6] “Marcación telefónica por pulsos”  
[http://es.wikipedia.org/wiki/Marcac\\_dec%C3%A1dica\\_por\\_pulsos](http://es.wikipedia.org/wiki/Marcac_dec%C3%A1dica_por_pulsos)
- [7] “Marcación telefónica por pulsos”  
REYES, Carlos; NARVAEZ, Carlos, Microncontroladores PIC.
- [8] “Codificación DMTF”  
<http://platea.pntic.mec.es/rrodrigu/es/Interface%20DTMF.pdf>
- [9] “Apuntes de conmutación”  
AULESTIA, Hugo
- [11] “Telefonía fija inalámbrica”  
<http://www.supertel.gov.ec/index.php/servicios-de-telecomunicaciones/62-telefonía-fija/>
- [12] “Funcionamiento simplificado de la telefonía móvil”  
<https://www.ibercom.com/soporte/index.php?>

- [13] “Comunicación mediante un teléfono móvil”  
<http://ec.kalipedia.com/informatica/tema/realiza-comunicacion-mediante-telefono>
- [14] “Funcionamiento de la telefonía móvil”  
<http://www.monografias.com/trabajos34/telefonía-celular/telefonía-celular>
- [15] “Apuntes de Control con Microcontroladores”  
Ing. COSTALES, Alcívar
- [16] “Microprocesador y microcontrolador”  
<http://www.monografias.com/trabajos12/microco/microco.shtml>
- [17] “El microprocesador”  
<http://www.smartsystemselectronics.com.mx/computacion/39-hardware/56-diferencias-entre-microprocesador-y-microcontrolador.html>
- [18] “Estructura de un microcontrolador”  
<http://www.smartsystemselectronics.com.mx/computacion/39-hardware/56-diferencias-entre-microprocesador-y-microcontrolador.html>
- [19] “Arquitectura de un microcontrolador”  
<http://www.monografias.com/trabajos12/microco/microco.shtml#DIFER>
- [20] “Como saber que microcontrolador utilizar”  
<http://www.monografias.com/trabajo12/microco/microco.shtml>
- [21] “Microcontrolador PIC 16F877A”  
<http://www.monografias.com/trabajos18/descripción-pic>
- [22] “Características del microcontrolador PIC 16F877A”  
<http://www.scribd.com/doc/101172/pic16f877-en-espanol1>
- [23] “Características periféricas del PIC 16F877A”  
<http://www.scribd.com/doc/101172/pic16f877-en-espanol1>
- [24] “Nomenclatura de un microcontrolador”  
<http://www.apuntes20%intro20%pic.pdf.com>

- [25] “Arquitectura del PIC 16F877A”  
[http:// www.scribd.com/doc/2939885/pic16F877-PERALTA?autodown=pdf](http://www.scribd.com/doc/2939885/pic16F877-PERALTA?autodown=pdf)
- [26] “Arquitectura interna”  
<http://www.datasheet.com>
- [27] “Organización de la memoria de programa”  
<http://www.fing.edu.uy/inco/cursos/firmware/cartillaPIC16F877.pdf>
- [28] “Mapa de la memoria de programa”  
<http://www.datasheet.com>
- [29] “Registro de funciones especiales”  
<http://www.datasheet.com>
- [30] “Registro STATUS”  
<http://www.datasheet.com>
- [31] “Registro INTCON”  
<http://www.datasheet.com>
- [32] “Diagrama de pines y funciones del PIC 16F877A”  
<http://www.datasheet.com>
- [33] “Descripción de pines del PIC 16F877A”  
<http://www.monografias.com/trabajos18/descripcion-pic/descripcion-pic.shtml>



**ANEXOS**

# **ANEXO A**

## **DATA SHEET**

### **Especificaciones Técnicas del Regulador LM7805**

# MC78XX/LM78XX/MC78XXA

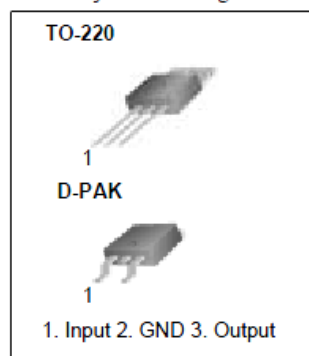
## 3-Terminal 1A Positive Voltage Regulator

### Features

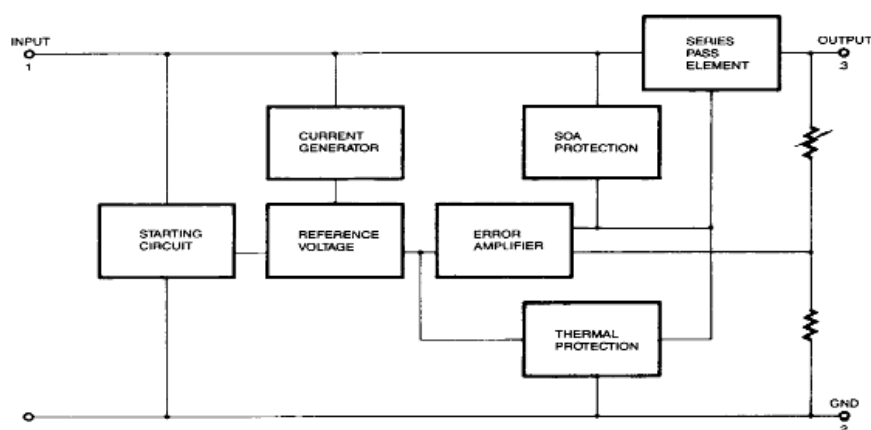
- Output Current up to 1A
- Output Voltages of 5, 6, 8, 9, 10, 12, 15, 18, 24V
- Thermal Overload Protection
- Short Circuit Protection
- Output Transistor Safe Operating Area Protection

### Description

The MC78XX/LM78XX/MC78XXA series of three terminal positive regulators are available in the TO-220/D-PAK package and with several fixed output voltages, making them useful in a wide range of applications. Each type employs internal current limiting, thermal shut down and safe operating area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.



### Internal Block Diagram



## Absolute Maximum Ratings

Parameter	Symbol	Value	Unit
Input Voltage (for $V_O = 5V$ to $18V$ ) (for $V_O = 24V$ )	$V_I$	35	V
	$V_I$	40	V
Thermal Resistance Junction-Cases (TO-220)	$R_{\theta JC}$	5	$^{\circ}C/W$
Thermal Resistance Junction-Air (TO-220)	$R_{\theta JA}$	65	$^{\circ}C/W$
Operating Temperature Range	TOPR	0 ~ +125	$^{\circ}C$
Storage Temperature Range	TSTG	-65 ~ +150	$^{\circ}C$

## Electrical Characteristics (MC7805/LM7805)

(Refer to test circuit , $0^{\circ}C < T_J < 125^{\circ}C$ ,  $I_O = 500mA$ ,  $V_I = 10V$ ,  $C_I = 0.33\mu F$ ,  $C_O = 0.1\mu F$ , unless otherwise specified)

Parameter	Symbol	Conditions	MC7805/LM7805			Unit	
			Min.	Typ.	Max.		
Output Voltage	$V_O$	$T_J = +25^{\circ}C$	4.8	5.0	5.2	V	
		$5.0mA \leq I_O \leq 1.0A$ , $P_O \leq 15W$ $V_I = 7V$ to $20V$	4.75	5.0	5.25		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}C$	$V_O = 7V$ to $25V$	-	4.0	100	mV
			$V_I = 8V$ to $12V$	-	1.6	50	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}C$	$I_O = 5.0mA$ to $1.5A$	-	9	100	mV
			$I_O = 250mA$ to $750mA$	-	4	50	
Quiescent Current	$I_Q$	$T_J = +25^{\circ}C$	-	5.0	8.0	mA	
Quiescent Current Change	$\Delta I_Q$	$I_O = 5mA$ to $1.0A$	-	0.03	0.5	mA	
		$V_I = 7V$ to $25V$	-	0.3	1.3		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5mA$	-	-0.8	-	$mV / ^{\circ}C$	
Output Noise Voltage	$V_N$	$f = 10Hz$ to $100KHz$ , $T_A = +25^{\circ}C$	-	42	-	$\mu V / V_O$	
Ripple Rejection	RR	$f = 120Hz$ $V_O = 8V$ to $18V$	62	73	-	dB	
Dropout Voltage	$V_{Drop}$	$I_O = 1A$ , $T_J = +25^{\circ}C$	-	2	-	V	
Output Resistance	$r_O$	$f = 1KHz$	-	15	-	$m\Omega$	
Short Circuit Current	ISC	$V_I = 35V$ , $T_A = +25^{\circ}C$	-	230	-	mA	
Peak Current	IPK	$T_J = +25^{\circ}C$	-	2.2	-	A	

### Note:

1. Load and line regulation are specified at constant junction temperature. Changes in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

### Typical Performance Characteristics

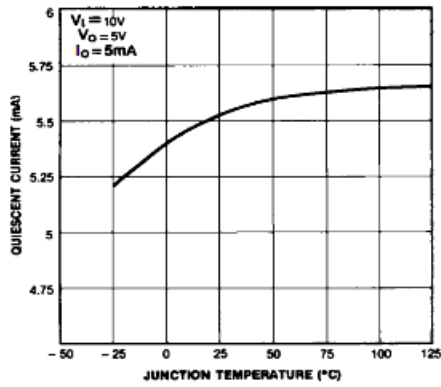


Figure 1. Quiescent Current

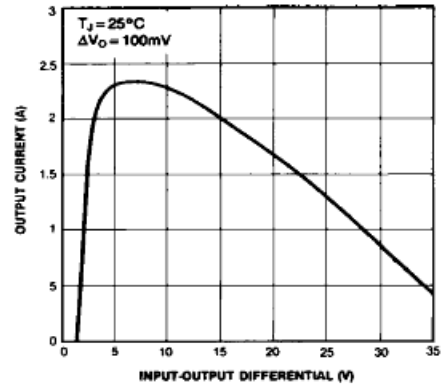


Figure 2. Peak Output Current

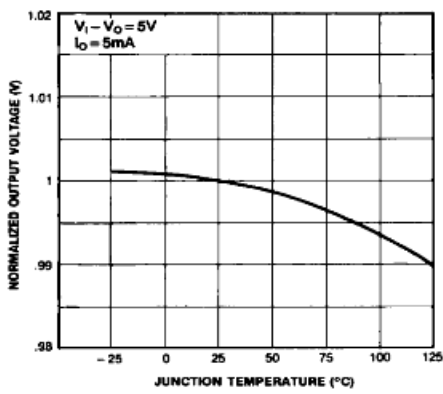


Figure 3. Output Voltage

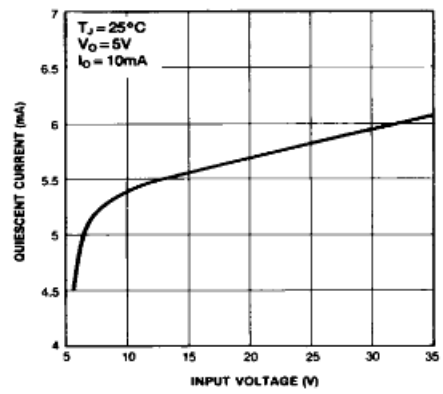


Figure 4. Quiescent Current

### Typical Applications

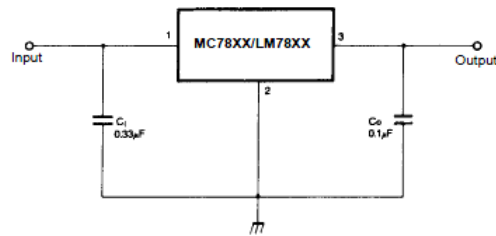


Figure 5. DC Parameters

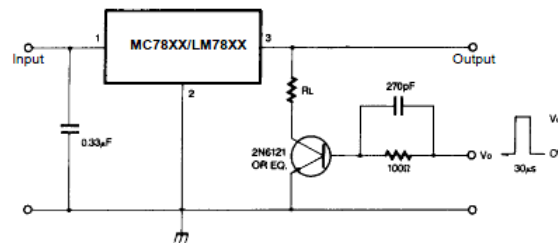


Figure 6. Load Regulation

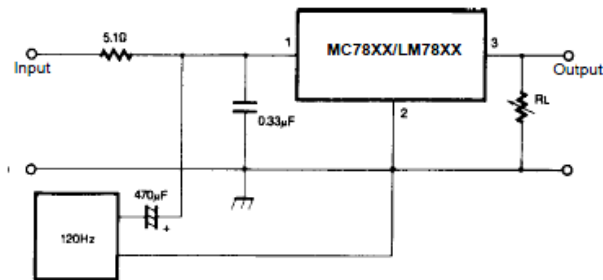


Figure 7. Ripple Rejection

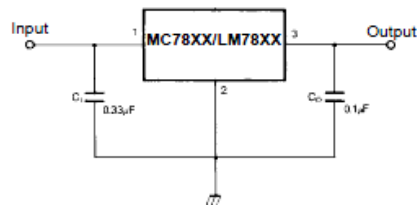


Figure 8. Fixed Output Regulator

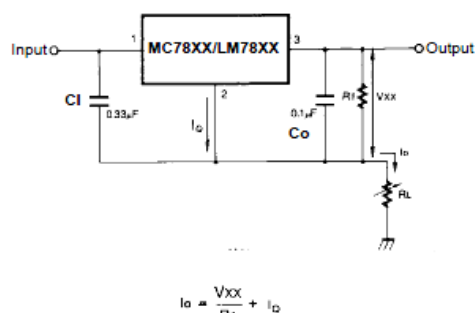
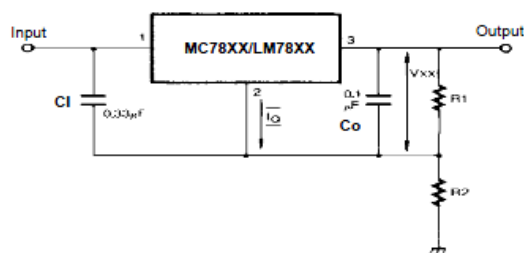


Figure 9. Constant Current Regulator

**Notes:**

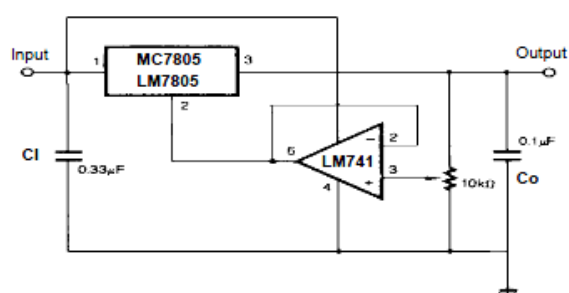
- (1) To specify an output voltage, substitute voltage value for "XX." A common ground is required between the input and the Output voltage. The input voltage must remain typically 2.0V above the output voltage even during the low point on the input ripple voltage.
- (2) C<sub>1</sub> is required if regulator is located an appreciable distance from power Supply filter.
- (3) C<sub>0</sub> improves stability and transient response.



$$I_{R1} \geq 5I_Q$$

$$V_O = V_{XX}(1+R_2/R_1) + I_Q R_2$$

Figure 10. Circuit for Increasing Output Voltage



$$I_{R1} \geq 5I_Q$$

$$V_O = V_{XX}(1+R_2/R_1) + I_Q R_2$$

Figure 11. Adjustable Output Regulator (7 to 30V)

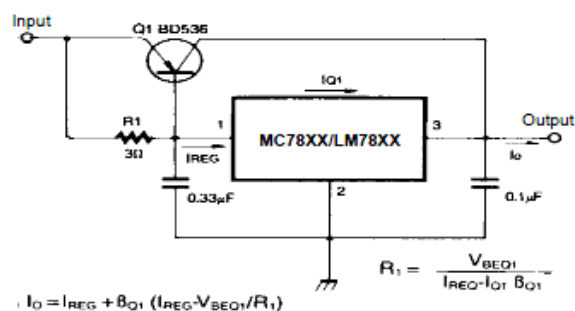


Figure 12. High Current Voltage Regulator

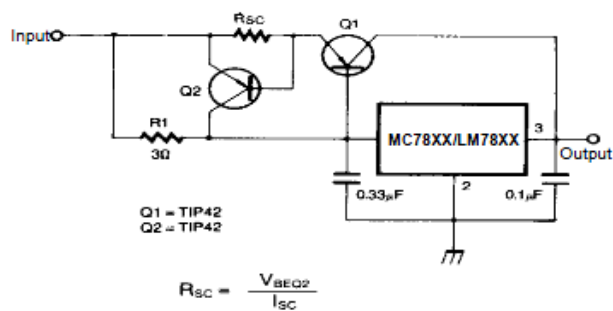


Figure 13. High Output Current with Short Circuit Protection

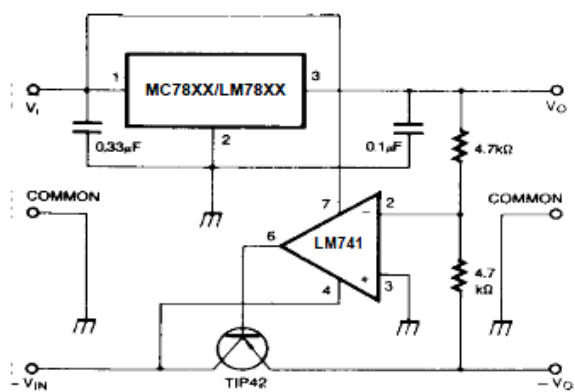


Figure 14. Tracking Voltage Regulator



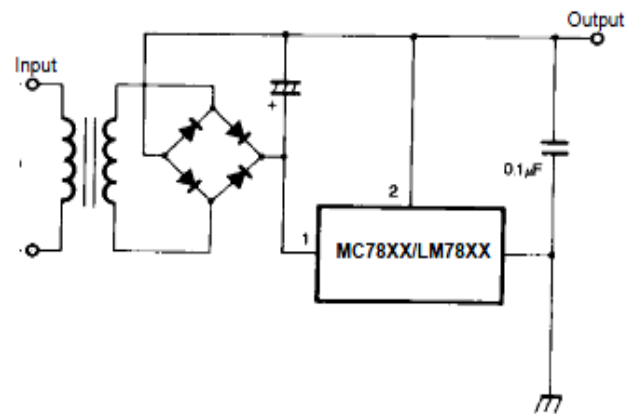


Figure 16. Negative Output Voltage Circuit

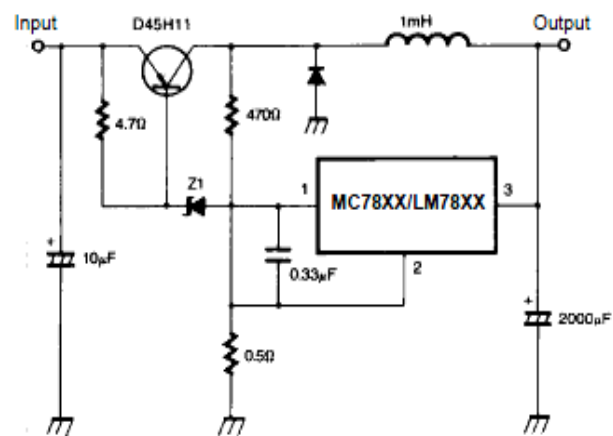


Figure 17. Switching Regulator



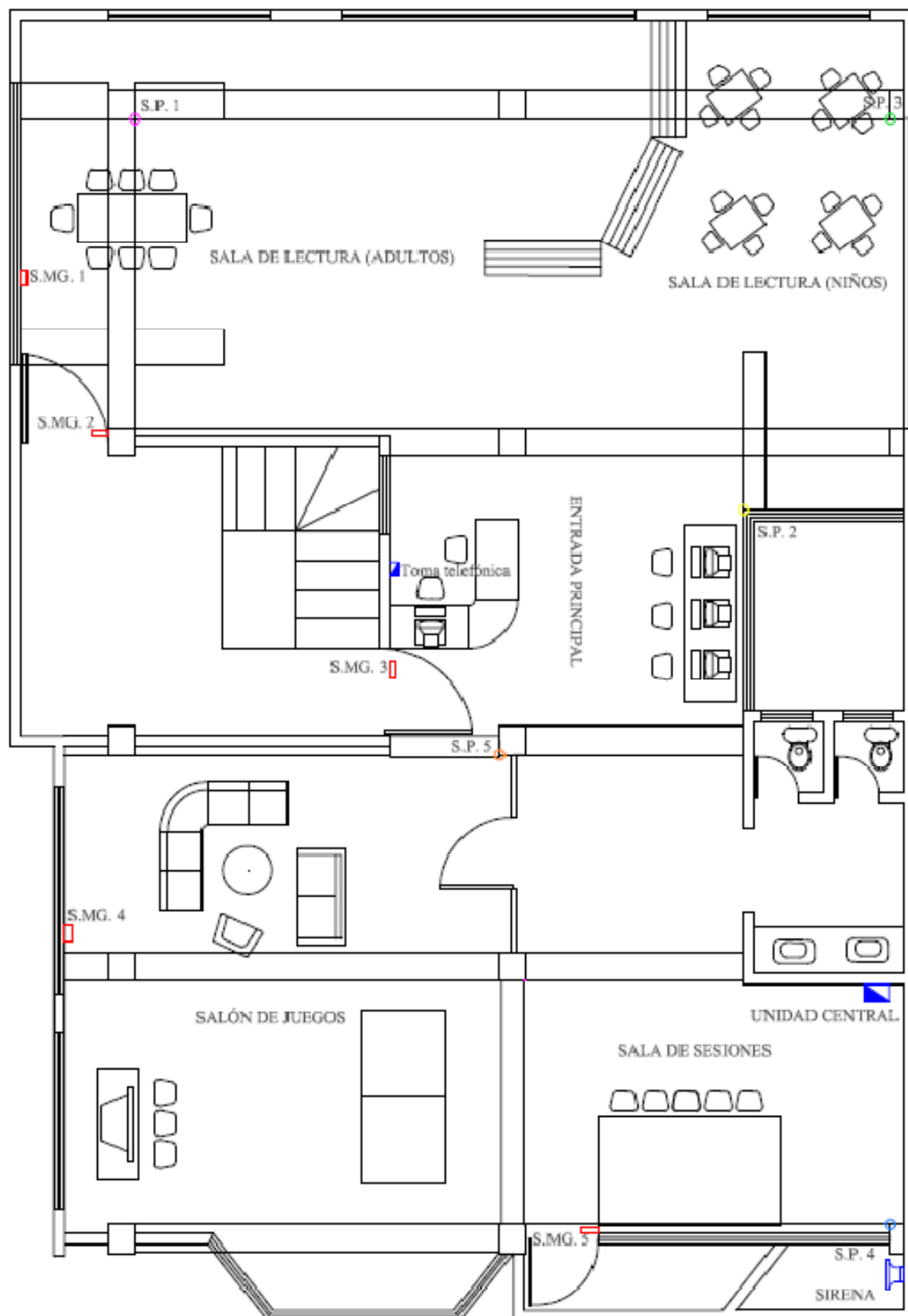
### Ordering Information

Product Number	Output Voltage Tolerance	Package	Operating Temperature
LM7805CT	±4%	TO-220	0 ~ + 125°C

Product Number	Output Voltage Tolerance	Package	Operating Temperature
MC7805CT	±4%	TO-220	0 ~ + 125°C
MC7806CT			
MC7808CT			
MC7809CT			
MC7810CT			
MC7812CT			
MC7815CT			
MC7818CT			
MC7824CT			
MC7805CDT			
MC7806CDT			
MC7808CDT			
MC7809CDT			
MC7810CDT			
MC7812CDT			
MC7805ACT	±2%	TO-220	
MC7806ACT			
MC7808ACT			
MC7809ACT			
MC7810ACT			
MC7812ACT			
MC7815ACT			
MC7818ACT			
MC7824ACT			

# **ANEXO B**

Planos diseñados para la  
implementación de la alarma  
antirrobo.



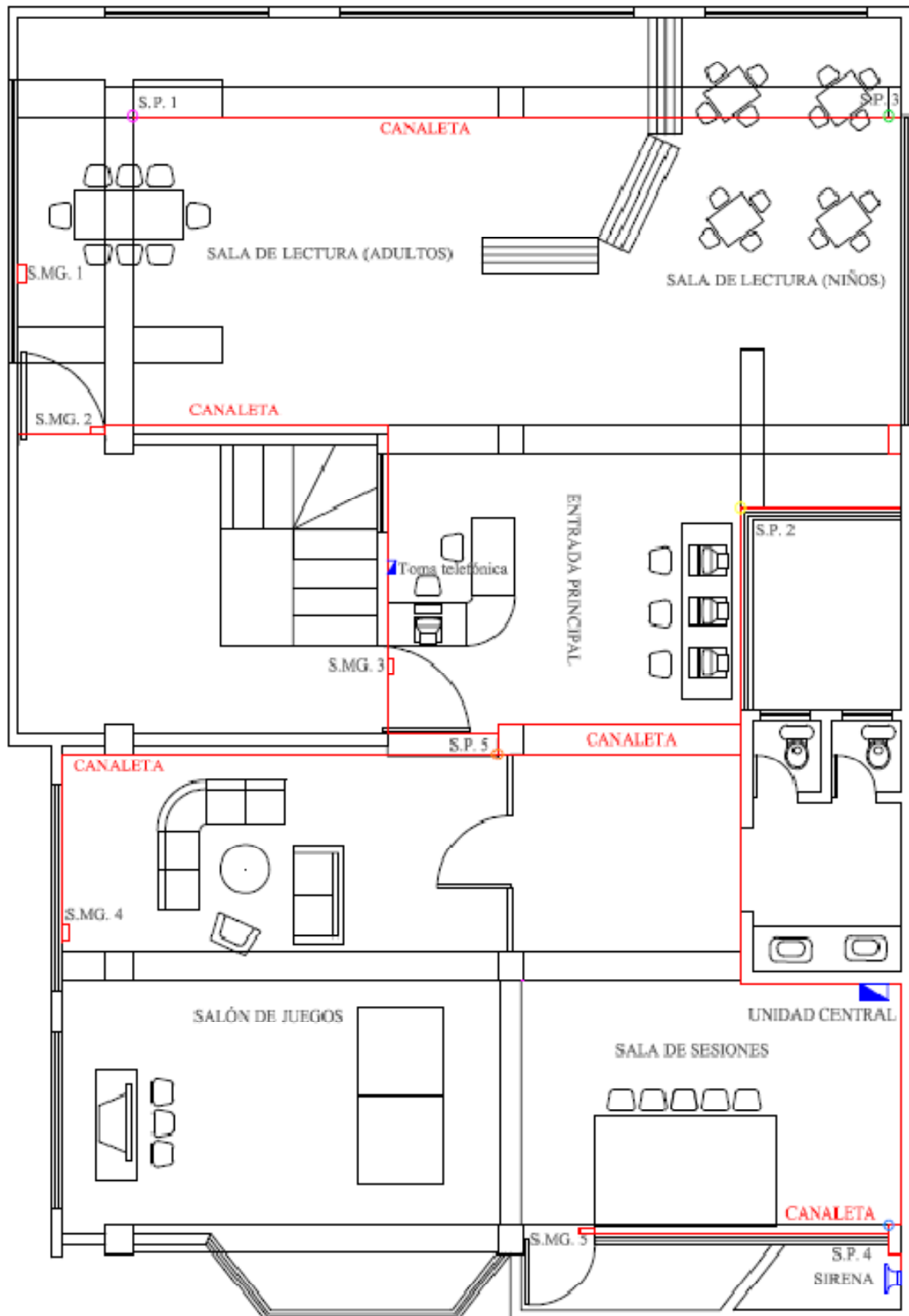
Construcción e implementación de una alarma antirrobo con interfaz DTMF de línea telefónica utilizando un microcontrolador para la Biblioteca "Manna Project International"

Plano para la instalación de sensores, unidad central y sirena

Dib: BYRON HERMOGENES NOGALES MINGA

E.S.F.O.T.

PLANO N° 1



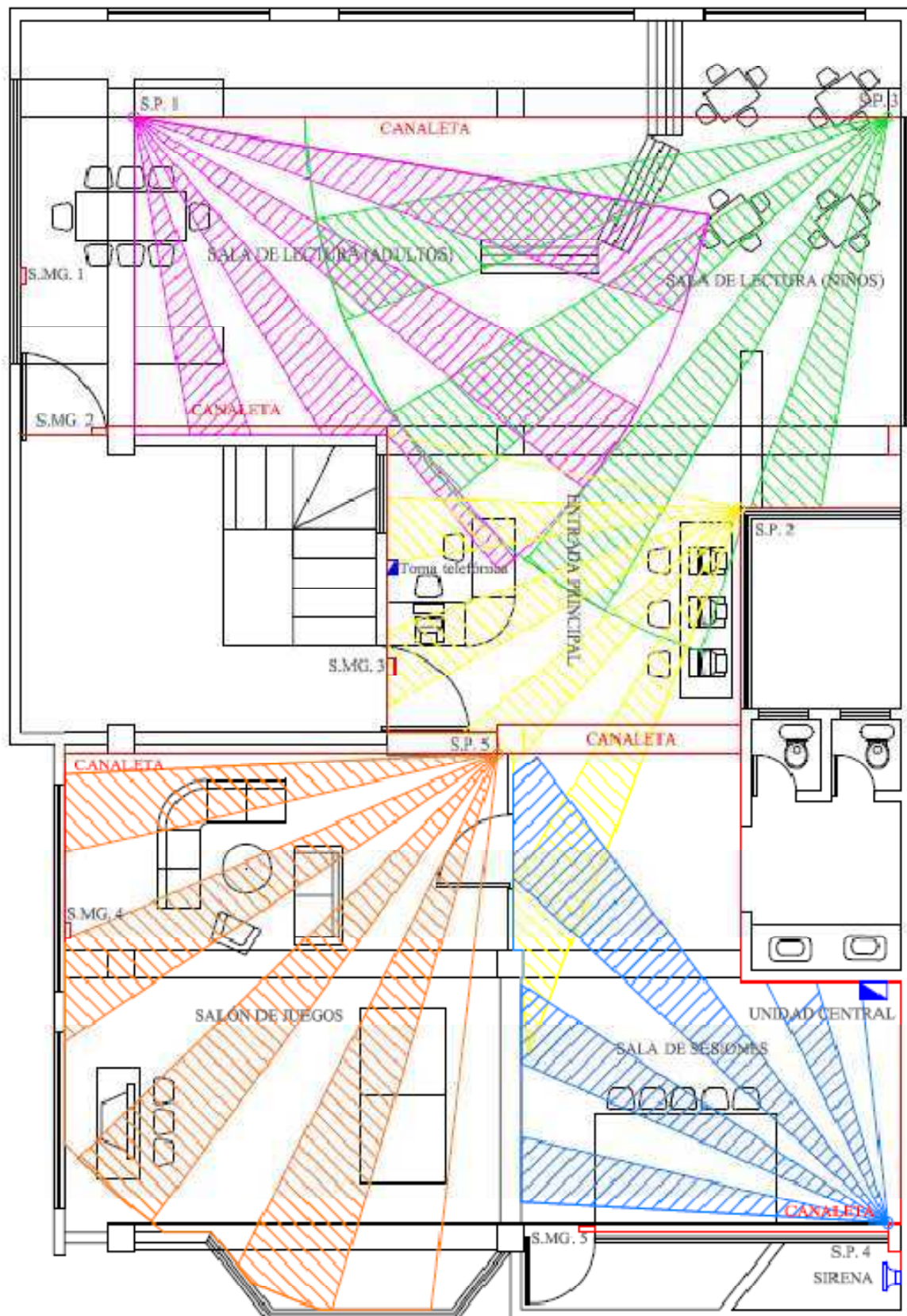
**Construcción e implementación de una alarma antirrobo con interfaz DTMF de línea telefónica utilizando un microcontrolador para la Biblioteca "Manna Project International"**

**Plano para la instalación de canaleta e intalación de cable para los sensores**

**Dib: BYRON HERMOGENES NOGALES MINGA**

**E.S.F.O.T.**

**PLANO N° 2**



ERN

**Construcción e implementación de una alarma antirrobo con interfaz DTMF de línea telefónica utilizando un microcontrolador para la Biblioteca "Manna Project International"**

**Zonas de detección de los sensores de presencia**

**Dib: BYRON HERMOGENES NOGALES MINGA**

**E.S.F.O.T.**

**PLANO N° 3**