



Universidad del Azuay

Facultad de Ciencia y Tecnología

Escuela de Ingeniería Electrónica

“Análisis, Diseño y Construcción de un Circuito de Control de Conmutación Automatizado, para el equipo de medición de las pruebas de las Características Eléctricas de los cables telefónicos de Planta Externa-ETAPA, Fluke 189”.

Trabajo de Graduación previo
a la obtención del Título de
Ingeniero Electrónico.

Autores: María del Carmen García Alvarez.

Luis Enrique Rodas Guapucal.

Director: Ing. Leonel Pérez Rodríguez.

Cuenca – Ecuador.
2006.

AGRADECIMIENTO:

Agradecemos a todas las personas que han aportado con su conocimiento para el desarrollo de este proyecto, a nuestro director Ing. Leonel Pérez, Ing. Edgar Pauta e Ing. Omar Nuñez funcionarios de la empresa ETAPA.

DEDICATORIA:

Sobre todas las cosas agradezco a Dios, más allá de una frase común y sin importar el que dirán. Pues en los momentos más oscuros de mi vida Él puso su corazón en el mío y en lo que más amo. Por eso este proyecto va dedicado a ti mi Amigo y a la persona que me dio la vida y me ha cuidado tiernamente sin limitarse y es mi mami Carmita, a mi fuerte y cariñoso papito Santiago, a mi princesa Toti, a mi risueño Diegui, a mi hermosa Cristinita, a toda mi querida familia y a mis entrañables amigos.

María del Carmen García Álvarez.

Agradezco primeramente a Dios y dedico este trabajo con mucho cariño a mi esposa y mi querido hijo; y de manera especial a mis padres Rosa y Benjamín, por estar siempre junto a mi, demostrándome su amor incondicional.

Luis Enrique Rodas

INDICE**CAPITULO I.****PLANTA EXTERNA.**

1.1 Introducción.....	1
1.2 Configuración de la Red de Planta Externa.....	2
1.2.1 Red Primaria.....	2
1.2.2 Red Secundaria.....	3
1.2.3 Línea del Abonado.....	4
1.3 Características Eléctricas de los Cables Telefónicos.....	4
1.3.1 Parámetros Eléctricos.....	4
1.3.1.1 Parámetros Primarios.....	4
1.3.1.1.1 Resistencia.....	5
1.3.1.1.2 Inductancia.....	6
1.3.1.1.3 Capacitancia.....	7
1.3.1.1.4 Conductancia.....	8
1.3.1.2 Parámetros Secundarios.....	8
1.3.1.2.1 Atenuación (At).....	9
1.3.2 Protocolo de Pruebas de las Características Eléctricas de los Cables Telefónicos.....	9
1.3.2.1 Prueba de continuidad eléctrica de los pares.....	10
1.3.2.2 Prueba de resistencia de bucle.....	12
1.3.2.3 Prueba de capacitancia mutua.....	13
1.3.2.4 Prueba de ruidos metálicos.....	14
1.3.2.5 Prueba de desequilibrio resistivo.....	15
1.3.2.6 Prueba de desequilibrio capacitivo.....	16
1.4 Técnica actual utilizada por ETAPA, para el Protocolo de Pruebas de Las Características Eléctricas de los Cables Telefónicos.....	17

CAPITULO II.**EL MICROCONTROLADOR PIC 16F877.**

2.1 Introducción.	19
2.2 El Microcontrolador PIC 16F877.	19
2.2.1 Características.	20
2.2.2 Periféricos.	21
2.2.3 Pinaje.	22
2.2.4 Memorias incorporadas al Microcontrolador.	26
2.2.4.1 Memoria de Programa.	26
2.2.4.2 Memoria de Datos.	28
2.2.5 Interrupciones por el TIMER0.	32
2.2.6 Temporizador TIMER0.	32
2.2.7 Repertorio de instrucciones.	33
2.2.7.1 Instrucciones orientadas al manejo de <i>bytes</i> .	34
2.2.7.2 Instrucciones orientadas al manejo de <i>bits</i> .	39
2.2.7.3 Instrucciones orientadas a operaciones con literales y de control.	42
2.3 Display de Cristal Líquido LCD.	45
2.3.1 Modos de operación de un LCD.	46
2.3.2 Comandos.	46
2.3.3 Líneas de control.	48
2.3.4 Pinaje del LCD.	49

CAPITULO III**ANALISIS, DISEÑO Y CONSTRUCCION DE UN CIRCUITO DE CONTROL DE CONMUTACION AUTOMATIZADO, PARA EL EQUIPO DE MEDICION DE LAS CARACTERISTICAS ELECTRICAS DE LOS CABLES TELEFONICOS, FLUKE 189.**

3.1 Introducción.	51
3.2 <i>Software</i> de apoyo.	52
3.2.1 MPLAB.	52
3.2.2. IC-PROG.	54
3.2.3 <i>SOFTWARE</i> FLUKE 189.	54
3.2.4. PROTEL.	57
3.3 <i>Software</i> del Circuito de Control de Conmutación Automatizado.	58
3.3.1 Flujograma del Circuito de Control de Conmutación Automatizado.	59
3.3.2 Código de Programa del Circuito de Control de Conmutación Automatizado.	62
3.4 PCB del Circuito de Control de Conmutación Automatizado.	84

CONCLUSIONES Y RECOMENDACIONES GENERALES.	91
--	----

REFERENCIAS.

Glosario.....	93
Bibliografía.....	94
Anexo 1.....	95
Anexo 2.....	97
Anexo 3.....	98
Anexo 4.....	99
Anexo 5.....	99

Resumen del Proyecto.

Análisis, Diseño y Construcción de un Circuito de Control de Conmutación Automatizado, para el equipo de medición de las pruebas de las Características Eléctricas de los cables telefónicos de Planta Externa-ETAPA, Fluke 189.

El presente proyecto está dirigido al Análisis, Diseño y Construcción de un Circuito de Control de Conmutación Automatizado, ha ser incorporado en el equipo de medición de las pruebas de las Características Eléctricas Fluke 189 ; y está dividido en tres puntos:

- El universo de estudio: Planta Externa.
- El microcontrolador PIC 16F877, mediante el cuál se logrará el control automatizado sobre el circuito de conmutación.
- Análisis, Diseño y Construcción del Circuito de Control de Conmutación Automatizado, y las herramientas tanto de software como de hardware utilizadas para el desarrollo de este proyecto.

Abstract.**Analysis, Design and Construction of an Automated Commutation Circuit Control , for the test measurement equipment of the Electrical Characteristics of telephone cables of External Plant - ETAPA, Fluke 189.**

The aim of the present work is to Analysis, Design and Construction of an Automated Commutation Circuit Control. This device will be incorporated in the test measurement equipment of the Electrical Characteristics Fluke 189. This work is divided in three points:

- Universe study: External Plant.
- PIC 16F877 Microcontroller, this device will perform the automated control of the commutation circuit.
- Analysis, design and construction of the Automated Commutation Circuit Control, software and hardware tools used for the development of this project.

García Alvarez María del Carmen

Rodas Guapucal Luis Enrique

Materia: Trabajo de Graduación

Director: Ing. Leonel Pérez Rodríguez

Mayo del 2006.

“Análisis, Diseño y Construcción de un Circuito de Control de Conmutación Automatizado, para el equipo de medición de las pruebas de las Características Eléctricas de los cables telefónicos de Planta Externa-ETAPA, Fluke 189”

CAPITULO I.

PLANTA EXTERNA.

1.1 Introducción.

Este capítulo está dedicado a dar una visión general de la infraestructura y configuración de las Redes de Planta Externa, con el fin de tener un marco de conocimiento y manejo del universo de estudio, para el cual se desarrollará el Circuito de Control de Conmutación Automatizado del equipo de medición de las características eléctricas de los cables, Fluke 189; y a la vez se podrá observar la justificación de su diseño y construcción.

En forma general, las Redes de Planta Externa para la transmisión de los servicios de telecomunicaciones, son redes de cables de cobre que permiten mantener las comunicaciones dentro de un área determinada de una ciudad, un país, o a nivel internacional, por tanto es inevitable exponer la red telefónica a elementos extraños, que influyen en el equilibrio eléctrico, continuidad y estabilidad para la cual fue creada, exponiéndose también a problemas climáticos, agentes químicos en el aire y la tierra; influencias eléctricas y electromagnéticas. Por lo antes señalado, surge la razón de realizar sobre la red, un exigente protocolo de pruebas de las características eléctricas de los cables telefónicos, que comprueben su funcionalidad.

En la actualidad la Planta Externa remarca su importancia, dado que es el medio de enlace entre las centrales telefónicas y el usuario, tomando en cuenta que no sólo son el transporte de telefonía, sino también de datos, por lo que se requiere de una mayor protección, cumpliendo así el objetivo primordial que es la CALIDAD de servicio al usuario.

1.2 Configuración de La Red de Planta Externa.

La Red de Planta Externa basa principalmente su configuración en el bucle del abonado, que está conformado por:

1.2.1 Red Primaria.

1.2.2 Red Secundaria.

1.2.3 Línea del Abonado.

1.2.1 Red Primaria.

Esta comprende desde el repartidor general de la central hasta los armarios de distribución o subrepartidores de una zona. Para una mejor comprensión de los elementos principales mencionados en el concepto de red primaria, se definen a continuación el repartidor general y los armarios de distribución.

Repartidor General.



Foto 1.1 Repartidor general.

Este es el punto donde llegan las líneas de abonados y permiten conectar hacia los equipos de conmutación (foto 1.1).

Sus funciones principales son:

- **Función de mezclado:** Permite conectar las líneas de abonados a los equipos de conmutación; como por ejemplo en los trabajos de instalaciones y traslados.
- **Función de protección:** Mediante la cuál se evita la entrada de sobre tensiones y sobrecorrientes, causados por rayos o líneas de energía eléctrica.

. **Función de corte y prueba:** Esta función permite trabajar sobre las líneas para operación, gestión y mantenimiento.

Armarios de Distribución.



Foto 1.2 Armario de distribución.

Llamados también subrepartidores (foto 1.2), están localizados estratégicamente dentro de una zona o distrito telefónico; entendiéndose como distrito, a las zonas que en función de la red dividen una ciudad geográficamente.

Los armarios están constituidos de una estructura metálica, y son el lugar de conexión entre los cables primarios y los secundarios a través de bloques de conexión, que por lo general son de 50 ó 100 pares.

Este punto de conexión es clave para la localización de averías.

1.2.2 Red Secundaria.

Es la red que va desde los armarios de zona hasta la caja de distribución, para ello, las líneas telefónicas que salen desde los armarios deben pasar primero por cajas de distribución que se encuentran ubicadas en los postes, fachada de edificios o en su interior.

Caja de Distribución.



Foto 1.3 Caja de distribución.

Se define como el punto de conexión entre las líneas individuales de acometida de cada abonado y los cables de la red secundaria (foto 1.3). Estas cajas de distribución están diseñadas por lo general para soportar 10 pares; y pueden tener o no elementos de protección. Este punto de conexión permite realizar labores de operación y mantenimiento.

1.2.3 Línea del Abonado.



Foto 1.4 Línea del abonado.

Son los cables que van desde la caja de distribución hacia el aparato telefónico (foto 1.4). Generalmente se dividen en dos tramos, el primero constituido por cable de acometida tipo exterior y el segundo constituido por un cable tipo interior localizado dentro de la casa del abonado; terminado en un conector, placa o roseta.

1.3 Características Eléctricas de Los Cables Telefónicos.

Siendo la Planta Externa el medio de enlace entre las centrales telefónicas y el usuario; es importante que funcione correctamente para brindar calidad de servicio al usuario. Para ello es necesario realizar un protocolo exigente de pruebas y controles antes de la recepción de la construcción de una red telefónica, con la finalidad de detectar fallas o daños ocurridos en los cables, ya que estas fallas generan perturbaciones a nivel de transmisión de voz y de datos.

El protocolo de pruebas de las características eléctricas está relacionado de forma directa con los parámetros eléctricos.

1.3.1 Parámetros Eléctricos.

Los parámetros eléctricos están divididos en dos grupos:

1.3.1.1 Parámetros Primarios.

1.3.1.2 Parámetros Secundarios.

1.3.1.1 Parámetros Primarios.

La Red de Planta Externa está constituida por cables de cobre de varias capacidades, cada par que lo conforma tiene sus hilos entorchados y aislados tanto entre ellos como de la pantalla y tierra.

Cada conductor o hilo del par presenta una resistencia y una inductancia, y al encontrarse los dos conductores cerca uno del otro, generan entre estos una capacitancia y una conductancia.

Los parámetros primarios de una línea de transmisión son:

1.3.1.1.1 Resistencia.

1.3.1.1.2 Inductancia.

1.3.1.1.3 Capacitancia.

1.3.1.1.4 Conductancia.

Y pueden ser representados a través de un modelo RLC equivalente a un par telefónico (figura 1.1).

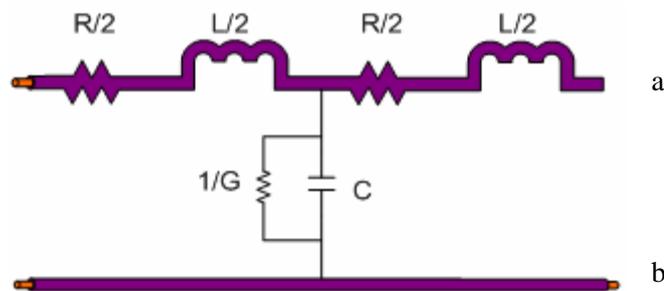


Figura 1.1 Esquema gráfico de los parámetros primarios.

1.3.1.1.1 Resistencia (R).

Conocida como resistencia óhmica, y está definida como “la propiedad que tiene el conductor para oponerse a la circulación de la corriente”, y que varía en función de:

- La longitud del conductor.
- El diámetro del conductor.
- Material del conductor.

La resistencia puede ser determinada mediante la siguiente expresión:

$$R = \frac{\rho \cdot L}{S} [\Omega] \qquad S = \frac{\pi}{4} \cdot d^2 [mm^2]$$

Donde:

- R = Resistencia del conductor, expresada en Ω .
- ρ = Resistividad, expresada $\Omega \text{ mm}^2 / \text{m}$.
- L = Longitud del conductor, expresado en m.
- S = Area de la sección transversal del conductor, expresada en mm^2 .
- d = Diámetro del conductor, expresado en mm.

Dentro de este parámetro primario se debe considerar el **Efecto Pelicular**; el cuál se manifiesta únicamente en corriente alterna, este efecto aparece por el uso de frecuencias altas, permitiendo que la corriente eléctrica se concentre en la superficie del conductor, dentro de un espesor llamado “Profundidad de Penetración”, siendo más estrecha cuando más alta es la frecuencia; mientras que la corriente continua circula por toda la masa del conductor.

1.3.1.1.2 Inductancia (L).

Debido a que los hilos o conductores se comportan como una serie de condensadores conectados en paralelo, y la configuración de estos hilos es simétrica, presentan características que lo hacen análogos a una bobina de inducción con núcleo de aire, por el hecho de ir trenzados y cambiando de posición a través de su trayectoria.

Esta particularidad que presentan los hilos genera con el paso de la corriente campos electromagnéticos que circulan alrededor de los hilos; donde los campos opuestos generados por corrientes de sentidos opuestos se contrarrestan, siempre y cuando los hilos están cercanos y las corrientes del hilo **a** son iguales a las corrientes del hilo **b**.

La inductancia depende del:

- Diámetro del conductor.
- Torcido del par de hilos.
- La distancia entre conductores.

La inductancia puede ser determinada mediante la siguiente expresión:

$$L = \frac{Xl}{\omega} [\text{Henrio}] \quad Xl = \omega.L[\Omega] \quad \omega = 2\pi.f[\text{rad} / \text{seg}]$$

Donde:

L = Inductancia, expresada en henrios.

Xl = Reactancia inductiva, expresada en Ω .

ω = Frecuencia angular, expresada en rad/seg.

f = Frecuencia, expresada en hertz por segundo.

1.3.1.1.3 Capacitancia (C).

Debido a la configuración simétrica de los pares, se comportan como verdaderos condensadores a lo largo de toda su extensión, los cuales tienen la capacidad de almacenar cierta cantidad de cargas eléctricas que se van sumando a medida que aumenta la longitud del cable.

La capacitancia depende del :

- Tipo de aislamiento.
- La distancia y frecuencia aplicada.

La capacitancia puede ser determinada mediante la siguiente expresión:

$$C = \frac{1}{\omega.X_c} [F] \quad \omega = 2\pi.f[\text{rad} / \text{seg}]$$

Donde:

C = Capacitancia, expresada en faradios.

Xc = Reactancia capacitiva, expresada en Ω .

ω = Frecuencia angular, expresada en rad/seg.

f = Frecuencia, expresada en hertz por segundo.

1.3.1.1.4 Conductancia (G).

La conductancia es el parámetro que determina la corriente que se pierde transversalmente a través de la superficie de contacto entre los pares y la capa de protección. Este efecto se debe al hecho de que el aislamiento entre ellos no es perfecto. La conductancia es la conductibilidad que presenta un circuito a la circulación de una corriente eléctrica y que es igual al inverso del valor de aislamiento que existe entre los conductores y la superficie de contacto entre ellos. Su unidad es el MHO o siemens.

La conductancia puede ser determinada mediante la siguiente expresión en corriente alterna:

$$G = 2.\pi.f.C.Cp[Mho]$$

Donde:

G = Conductancia, expresada en Mho.

C = Capacitancia, expresada en faradios.

Cp = Angulo de pérdida o factor de potencia del material aislante.

f = Frecuencia, expresada en hertz por segundo.

1.3.1.2 Parámetros Secundarios.

El comportamiento de una línea de transmisión al ser cruzada por una señal de transmisión alterna se define como parámetros secundarios.

Los parámetros secundarios son:

- Impedancia característica.
- Atenuación.
- Velocidad de propagación Vpm.

A continuación se citará el parámetro de Atenuación debido a que las variables de su expresión matemática pueden ser medidas por el equipo Fluke 189; equipo que es utilizado en el proyecto.

1.3.1.2.1 Atenuación (At).

Desde el punto de vista de la red, la atenuación es la relación logarítmica entre el voltaje de entrada dividido por el voltaje de salida; donde la potencia o voltaje de entrada será siempre mayor que la potencia o voltaje de salida.

Para determinar la atenuación de una línea, en la banda de frecuencia vocal (300 a 3.600Hz) no se consideran los parámetros de inductancia y conductancia ya que son despreciables. Por lo tanto para calcular la atenuación con respecto a la frecuencia, la resistencia de *loop* y la capacitancia se deberá hacer uso de la siguiente expresión:

$$At = 8.696 \sqrt{\frac{\omega.C.R}{2}} [dB] \quad \omega = 2\pi.f [rad / seg]$$

Donde:

At = Atenuación de línea, expresada en dB.

R = Resistencia de bucle, expresada en Ω .

C = Capacitancia mutua, expresada en faradios.

ω = Frecuencia angular, expresada en rad/seg.

f = Frecuencia, expresada en hertz por segundo.

1.3.2 Protocolo de Pruebas de Las Características Eléctricas de Los Cables Telefónicos.

Para que una red telefónica pueda brindar correctamente los servicios de voz y datos, se debe realizar un protocolo de pruebas de verificación a través de la toma de medidas de las características eléctricas. Las pruebas se realizan tanto en una red telefónica nueva, como en una red telefónica en uso, en los cables primarios y en los secundarios.

El protocolo de pruebas de las características eléctricas que se deben realizar sobre una red telefónica son: continuidad eléctrica de los pares, resistencia de bucle, desequilibrio resistivo, resistencia de pantalla, resistencia de aislamiento, capacitancia mutua, desequilibrio capacitivo, niveles de ruidos metálicos y de tierra, impedancia, atenuación, tele y para diafonía, y por último prueba de puesta a tierra.

En el presente trabajo se considerará únicamente aquellas pruebas, que puedan ser realizadas por el equipo de medición téster Fluke 189 y controlado por el Circuito de Control de Conmutación Automatizado (capítulo 3), siendo estas las 4 primeras y las 2 últimas están citadas para un futuro control:

- 1.3.2.1 Prueba de continuidad eléctrica de los pares.
- 1.3.2.2 Prueba de resistencia de bucle.
- 1.3.2.3 Prueba de capacitancia mutua.
- 1.3.2.4 Prueba de los niveles de ruidos metálicos.
- 1.3.2.5 Prueba de desequilibrio resistivo.
- 1.3.2.6 Prueba de desequilibrio capacitivo.

1.3.2.1 Prueba de continuidad eléctrica de los pares.

Esta prueba se realiza sobre la totalidad de los hilos, la misma que permite comprobar la continuidad eléctrica de cada hilo que conforma el cable (desde la central hasta el armario o reserva). Permite detectar par abierto (figura 1.2), hilo de un par abierto (figura 1.3), hilos cruzados **a x b** (figura 1.4) y transpuestos por ejemplo: hilo **a** de un par cruzado con el hilo **a** de otro par (figura 1.5) .



Figura 1.2 Par abierto.



Figura 1.3 Hilo de un par abierto.



Figura 1.4 Cruce axb.

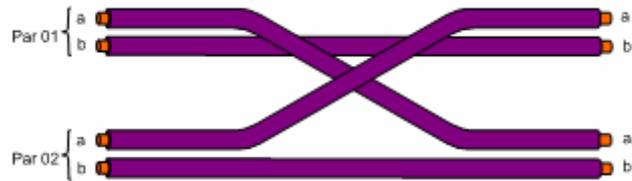


Figura 1.5 Transpuesto.

Las pruebas de continuidad eléctrica, pueden ser realizadas siguiendo el método de medición con dos operadores:

Para la realización de esta prueba, el equipo Fluke 189 debe trabajar en la posición de continuidad (nS). La primera extremidad del par deberá ser conectada al Fluke 189, y en la otra extremidad del par, los hilos **a** y **b** deberán estar en circuito cerrado (figura 1.6).

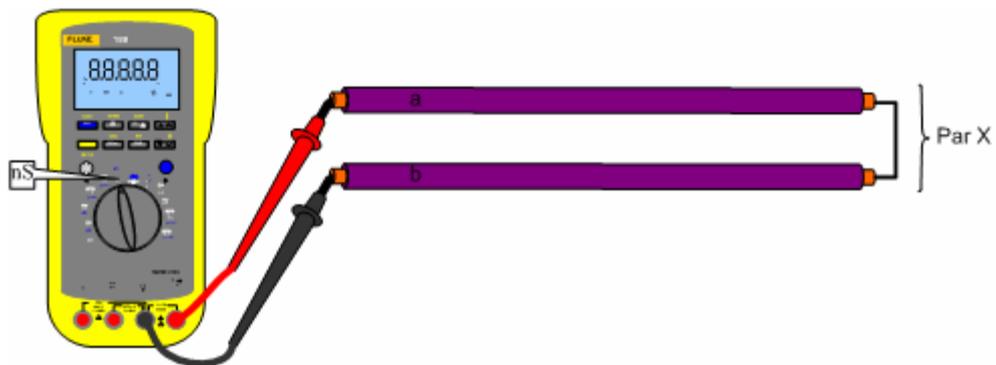


Figura 1.6. Prueba de continuidad eléctrica.

Una vez terminada la prueba con el primer par, se continuará con el resto de los pares.

Otra forma de realizar esta prueba podría ser mediante el uso de dos teléfonos, comprobando a través de la percepción de la voz la continuidad del circuito.

1.3.2.2 Prueba de resistencia de bucle.

La resistencia de bucle se conoce como la suma de las infinitas resistencias conectadas en serie, dentro de un circuito cerrado entre el hilo **a** y **b** de un mismo par. La prueba deberá ser realizada sobre todos los hilos.

Dentro de esta prueba se establece como norma que, los cálculos resistivos del enlace, entre la central telefónica y la placa o cajetín del cliente no debe ser mayor a 1800 Ω .

A continuación se muestra los valores de la resistencia de bucle en función del diámetro del conductor (tabla 1.1).

Diámetro en mm.	Sección En mm ²	Resistencia de bucle. Ω *Km .
0.4	0.1256	270
0.5	0.1963	170
0.6	0.2827	108
0.8	0.5026	70
0.9	0.6361	53

Tabla 1.1 Valores de resistencia de bucle en función del diámetro.

Las pruebas de resistencia de bucle, pueden ser realizadas siguiendo el método de medición con dos operadores:

Para la realización de esta prueba, el equipo Fluke 189 debe trabajar en la posición de resistencia (Ω). La primera extremidad del par deberá ser conectada al Fluke 189, y en la otra extremidad del par, los hilos **a** y **b** deberán estar en circuito cerrado (figura 1.7).

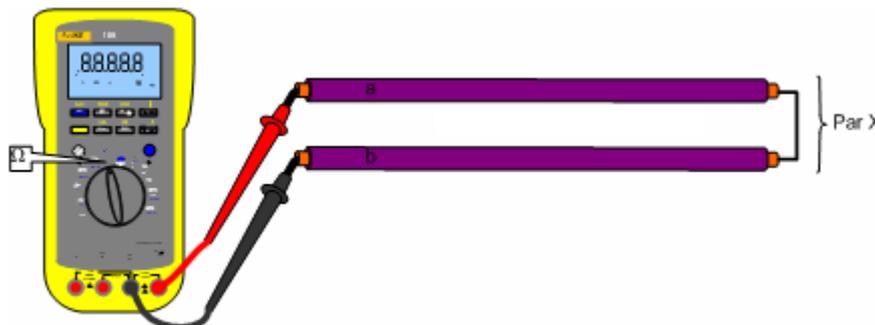


Figura 1.7 Prueba de resistencia de bucle.

Una vez terminada la prueba con el primer par, se continuará con el resto de los pares.

1.3.2.3 Prueba de capacitancia mutua.

La capacitancia es aquella que define la cantidad de electricidad almacenada entre los dos hilos de un mismo circuito. La capacitancia y la inductancia se modifican cuando los dos hilos de un par se separan significativamente. Las reactancias capacitivas e inductivas, que cumplen funciones inversas, no se cancelarán al presentarse esta separación, por lo que, se debe conservar y mantener un circuito únicamente resistivo, que no retrase la intensidad y no adelante el voltaje.

La prueba de capacitancia mutua se realiza sobre todos los hilos. Sus valores límites serán: $52 \pm 2 \text{ nF/Km}$.

La prueba de capacitancia mutua permitirá detectar:

- Una exagerada separación de los hilos de un mismo par.
- Un bajo aislamiento.
- Hilos o circuitos abiertos.
- Hilos transpuestos.

Las pruebas de capacitancia mutua, pueden ser realizadas siguiendo el método de medición con un operador:

Utilizando el Fluke 189 se seleccionará la función capacitiva y se procederá a medir la capacitancia existente entre cada par. El par no deberá estar en circuito cerrado al final (figura 1.8).

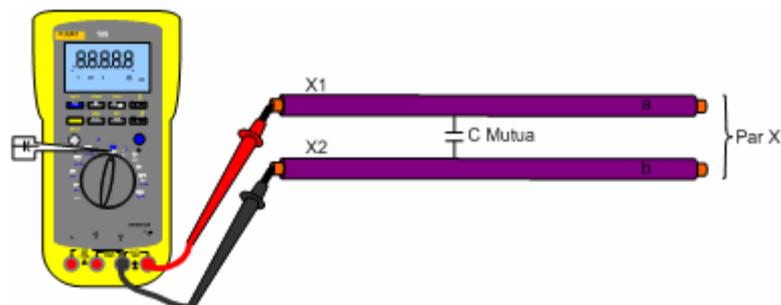


Figura 1.8 Prueba de capacitancia mutua.

El valor obtenido corresponderá a la capacitancia mutua total del par, dicho valor deberá ser dividido por la longitud del cable (en Km.), este resultado deberá estar siempre entre los valores antes citados, por ejemplo:

El valor capacitivo de un par de 3,564 Km. da una capacitancia de 186,75 nF.
Dividiendo el valor capacitivo obtenido por la distancia expresada en Km. tenemos:

$$\frac{186,75nF}{3,564Km.} = 52,40nF / km.$$

Observando que el resultado es aceptable, ya que está dentro del rango permitido por esta prueba.

1.3.2.4 Prueba de ruidos metálicos.

Siendo el ruido cualquier tipo de señal indeseable ajena a la señal de referencia (información transmitida a través del par telefónico) que la hace ininteligible en algún grado. Si la lectura de ruidos no cumple con las especificaciones, es señal de que el cable puede estar expuesto a fuertes inducciones de alta y baja tensión o estaciones radiodifusoras y esto puede ser debido a un bajo aislamiento, falta de continuidad de pantalla en los empalmes, alta resistencia de puesta a tierra ó ausencia de tierra.

En la practica se analizan dos tipos de inducción en los cables telefónicos, una de carácter electrostático, la cuál se produce por efecto interno en las líneas de transmisión o telefónicas, y la segunda de origen electromagnético, que se produce por efecto externo a las líneas telefónicas.

Tomando en cuenta que en la transmisión de datos el máximo de velocidad es directamente proporcional a los niveles de ruidos, se debe considerar lo siguiente:

Ruidos metálicos entre los hilos a y b de un mismo par.

Frente a este problema, se deberá impedir que los campos electromagnéticos ingresen a los hilos **a** y **b** a través de un blindaje total del cable en todo su recorrido

(desde la central hasta el aparato del abonado) y de esta manera impedir en lo posible el ingreso de ruidos metálicos.

La prueba de medición de ruido metálico, puede ser realizada de la siguiente manera (figura 1.9):

Utilizando el Fluke 189 se seleccionará la función de ruidos en V o mV. La primera extremidad del par se deberá conectar al Fluke 189, y la otra extremidad del par no requiere estar en circuito cerrado.

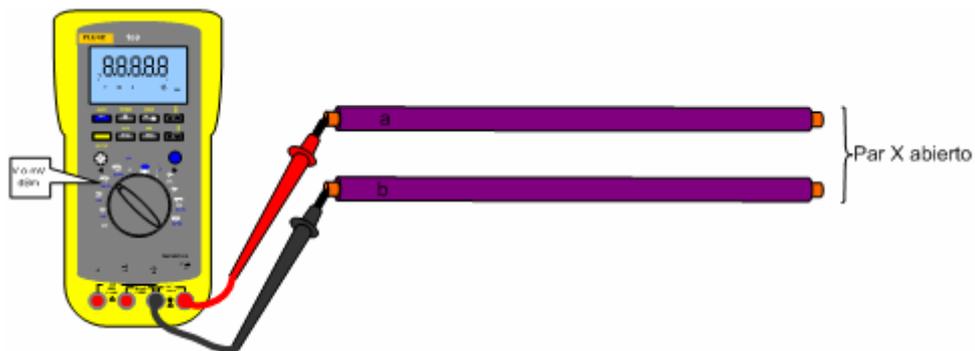


Figura 1.9 Medición de ruido metálico.

A continuación se presenta la designación del ruido, frente a un valor medido por el Fluke 189; (tabla 1.2).

DESIGNACION	RUIDO METALICO (V)	RUIDO METALICO (mV)
EXCELENTE	≤ -48 dBm	≤ -70 dBm
ACEPTABLE	≤ -30 dBm y > -48 dBm	≤ -40 dBm y > -70 dBm
INACEPTABLE	> -30 dBm	> -40 dBm

Tabla 1.2 Ruido metálico.

1.3.2.5 Prueba de desequilibrio resistivo.

Esta prueba se realiza sobre la totalidad de los hilos que conforman el cable y consiste en medir la diferencia resistiva de los hilos **a** y **b** de un par (figura 1.10). El resultado de la diferencia no deberá ser mayor al 2% del valor de la resistencia de bucle, ya que de ocurrir esto indicará presencia de diafonía en el cable.

Esta prueba además de detectar los desequilibrios resistivos (ΔR), permite también detectar fallas de continuidad, inversión de polaridad **a x b** y transpuesto.

Las pruebas de desequilibrio resistivo, pueden ser realizadas siguiendo el método de medición con dos operadores:

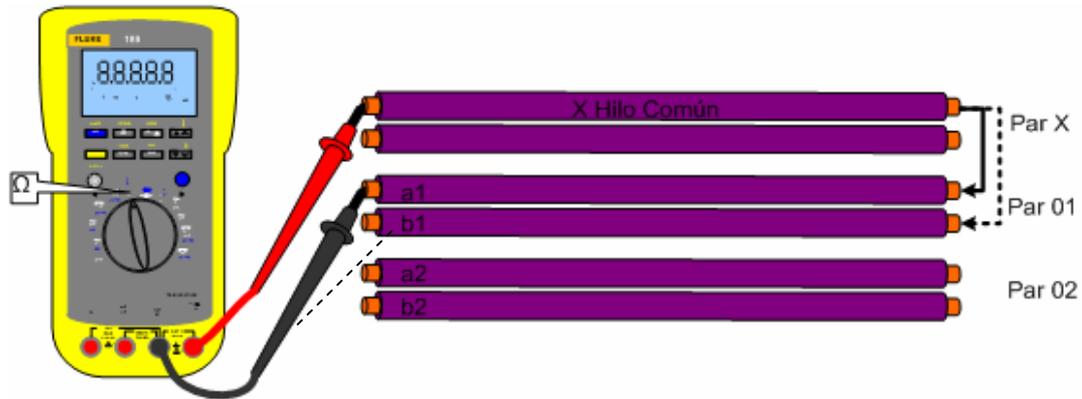


Figura 1.10 Medición del desequilibrio resistivo.

Para la realización de esta prueba, el equipo Fluke 189 debe trabajar en la posición de resistencia (Ω). Luego se deberá elegir como hilo común **X** entre los hilos **a** o **b** de un par **X**. Este hilo **X** será conectado a una punta de prueba del Fluke 189 y la otra punta de prueba será conectada al hilo **a** de un segundo par; mientras que en la otra extremidad deberá estar en circuito cerrado los hilos **a** y **X**, procediendo por último a la medición y obteniendo una respuesta RX_a . Repitiendo el mismo procedimiento con el hilo **b** de este segundo par y obteniendo como respuesta RX_b .

Siendo el desequilibrio resistivo de este par : $\Delta R_{ab} = RX_a - RX_b$

Una vez terminado con este par, se procederá de la misma manera con los pares restantes que conforman el cable, pero manteniendo el mismo hilo común **X**.

1.3.2.6 Prueba de desequilibrio capacitivo.

Esta prueba se realizara sobre la totalidad de los hilos que conforman el cable y consiste en la diferencia capacitiva de los hilos **a** y **b** de un par (figura 1.11).

El valor Máximo de desequilibrio capacitivo del un circuito en referencia con el apantallamiento será \leq a 2 nF/km.

Las pruebas de desequilibrio capacitivo, pueden ser realizadas siguiendo el método de medición con dos operadores:

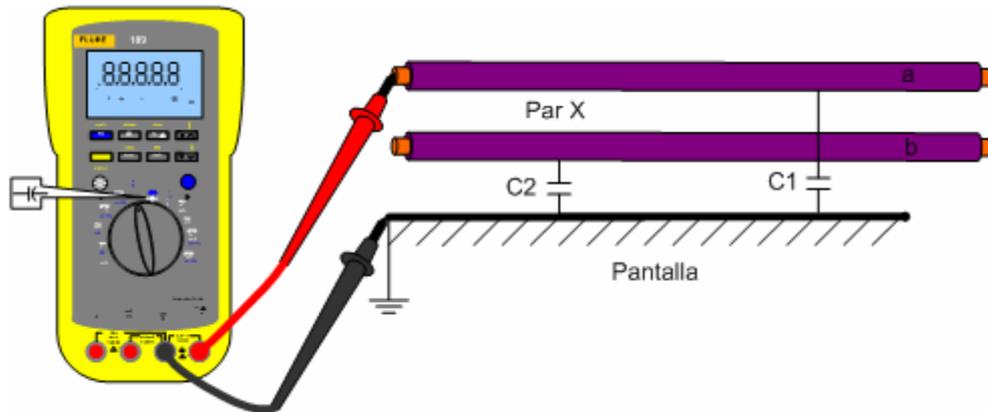


Figura 1.11 Medición del desequilibrio resistivo.

Con la ayuda del téster Fluke 189 en la función capacitiva, se procederá a medir la capacitancia existente entre el hilo **a** de un par y la pantalla, es decir Cap, y luego se medirá la carga capacitiva entre el hilo **b** de este mismo par y la pantalla, obteniendo Cbp. Por último el resultado será la diferencia de estos dos valores: $\Delta C_{ab} = C_{ap} - C_{bp}$. Tomando en cuenta que en la otra extremidad, el circuito deberá estar abierto.

Una vez terminado con este par, se procederá de la misma manera con los pares restantes que conforman el cable.

1.4 Técnica actual utilizada por ETAPA, para el Protocolo de Pruebas de Las Características Eléctricas de los Cables Telefónicos.



Foto 1.5 Medición de Pruebas.

El personal autorizado para la toma de pruebas de las características eléctricas, realiza su trabajo de forma manual, con ayuda del equipo de medición; téster Fluke 189 (foto 1.5). Las mediciones tomadas son registradas en formularios preestablecidos.

Cabe señalar que en la instalación de una red se emplean cables de varias capacidades, como por ejemplo de 1200 pares, y para comprobar su funcionamiento, es necesario verificar par a par y anotar sus mediciones manualmente en el formulario correspondiente a la prueba; utilizando el mismo procedimiento para otra prueba. El tiempo que se emplea para esto, es por lo general de 2 a 3 días.

CAPITULO II

EL MICROCONTROLADOR PIC 16F877.

2.1 Introducción.

Hoy en día la tecnología microcontrolada ha tenido una aceptación muy importante dentro de las diferentes áreas como bioelectrónica, robótica, telecomunicaciones, etc; en razón de las ventajas amplias que ésta presta.

Un microcontrolador, es un circuito integrado programable de alta escala de integración de los elementos que conforman un controlador y que contiene los componentes esenciales de un computador, pero en limitadas prestaciones. El microcontrolador se emplea para controlar el funcionamiento de una tarea determinada, dicho control reside como programa en su memoria. Las líneas de Entrada/Salida (E/S) del microcontrolador pueden estar conectadas por ejemplo, a sensores y actuadores del dispositivo a controlar; es decir llevar el control de aquellos periféricos que se necesiten para cumplir con los objetivos de cierta tarea; como en este caso; la construcción del Circuito de Control de Conmutación Automatizado para el Fluke 189, mediante el microcontrolador PIC 16F877.

2.2 El Microcontrolador PIC 16F877.

El microcontrolador PIC 16F877 pertenece a la serie de los microcontroladores PIC 16F87X. Esta serie está constituida por los PICs 16F870, 16F871, 16F872, 16F873, 16F874 y 16F876 y sus características principales están expuestas de manera comparativa en la tabla 2.1.

PICs	MEMORIA DE PROGRAMA		EEPROM Memoria de Datos Bytes	RAM Byte s	Puertos de E/S	Encapsu lado	Canales 8-bit ADC	PWM 10-bit	Timers WDT	Serial E/S	Veloci dad Max. MHz
	BYTES	OTP/FLASH Palabras									
PIC 16F870	3584 (FLASH)	2048X14 (FLASH)	64	128	22	28SP 28SO 28SS	5 (10- BIT)	1	1-16 BIT 2-8 BIT 1-WDT	USART	20
PIC 16F871	3584 (FLASH)	2048X14 (FLASH)	64	128	33	40P 44L 44PT	8 (10- BIT)	1	1-16 BIT 2-8 BIT 1-WDT	USART	20
PIC 16F872	3584 (FLASH)	2048X14 (FLASH)	64	128	22	28SP 28SO 28SS	5 (10- BIT)	1	1-16 BIT 2-8 BIT 1-WDT	MI ² C/SPI	20
PIC 16F873	7168 (FLASH)	4096X14 (FLASH)	128	192	22	28SP 28SO	5 (10- BIT)	2	1-16 BIT 2-8 BIT 1-WDT	USART MI ² C/SPI	20
PIC 16F874	7168 (FLASH)	4096X14 (FLASH)	128	192	33	40P 44L 44PQ 44PT	8 (10- BIT)	2	1-16 BIT 2-8 BIT 1-WDT	USART MI ² C/SPI	20
PIC 16F876	14336 (FLASH)	8192X14 (FLASH)	256	368	22	28SP 28SO	5 (10- BIT)	2	1-16 BIT 2-8 BIT 1-WDT	USART MI ² C/SPI	20
PIC 16F877	14336 (FLASH)	8192X14 (FLASH)	256	368	33	40P 44L 44PQ 44PT	8 (10- BIT)	2	1-16 BIT 2-8 BIT 1-WDT	USART MI ² C/SPI	20

Tabla 2.1 Serie de los PICs 16F87X.

2.2.1 Características.

En base al análisis de las características de la serie de estos PICs (tabla 2.1) y a los requerimientos necesarios para el desarrollo del Circuito de Control de Conmutación Automatizado (capítulo 3), se escogió el PIC 16F877; cuyas características especiales se describen a continuación:

- Multiplexa hasta 8 entradas analógicas de 10 bits.
- Procesador de arquitectura RISC; es decir, contempla un set de instrucciones reducidas.
- Juego de 35 instrucciones con 14 bits de longitud. La mayoría de las instrucciones se ejecutan en un sólo ciclo de instrucción, excepto CALL, GOTO, RETFIE, RETURN, RETLW que se ejecutan en dos ciclos de instrucción y DECFSZ, INCFSZ, BTFSC, BTFSS que se ejecutan en uno o dos ciclos de instrucción.
- 4 paginas de 2K palabras, lo que hace un total de 8192 palabras de 14 bits c/u.

- Permite hasta 368 *bytes* de memoria de datos RAM.
- Permite hasta 256 *bytes* de memoria de datos EEPROM.
- Está constituido por 40 pines.
- Tiene 5 puertos de E/S (A,B,C,D y E).
- 14 fuentes de interrupción internas y externas.
- Direccionamiento directo e indirecto de la Memoria de Datos.
- *Reset* por conexión de alimentación (POR:*Power-on Reset*).
- Temporizador *Power-on* (POP) y Oscilador Temporizador *Start-Up* (OST).
- Sistema de vigilancia *Watchdog timer* ó perro guardián (WDT).
- Código de protección programable.
- Modo *SLEEP* de bajo consumo.
- Programación y depuración serie "*In-Circuit*" (ICSP).
- Frecuencia de operación hasta 20 MHz (DC a 200 nseg de ciclo de instrucción).
- Rango de voltaje de operación de 2.0 a 5.5 volts.
- Bajo consumo: < 2 mA valor para 5 V y 4 Mhz.
 20 μ A para 3V y 32 Mhz.
 <1 μ A en *standby*.
- Rangos de temperatura: comercial, industrial y extendido.
- Bajo consumo de potencia:
 - Menos de 0.6mA a 3V, 4 Mhz.
 - 20 μ A a 3V, 32 Khz.
 - menos de 1 μ A corriente de *standby*.

2.2.2 Periféricos.

Los periféricos incluidos en el PIC 16F877 son:

- *TIMER0*: Contador/Temporizador de 8 *bits* con prescaler de 8 *bits*.
- *TIMER1*: Contador/Temporizador de 16 *bits* con prescaler.
- *TIMER2*: Contador/Temporizador de 8 *bits* con prescaler y postscaler de 8 *bits* y registro de periodo.
- Dos módulos de captura, comparación y PWM.

- Convertidor A/D (Análogo/Digital) de 10 *bits*, y hasta 8 canales.
- Puerto Serie Síncrono (SSP).
- Puerto Serie Universal (USART/SCI).
- Puerto Paralelo Esclavo (PSP): de 8 *bits* con líneas de protocolo.

2.2.3 Pinaje.

El PIC 16F877 está compuesto por 40 pines que se muestran en la figura 2.1.

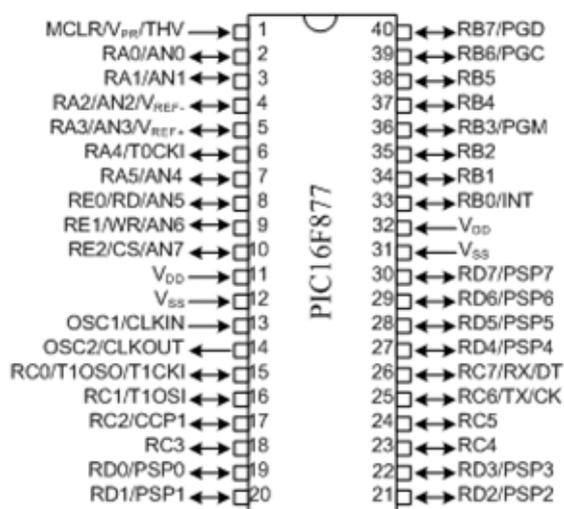


Figura 2.1 Pinaje del PIC.

Oscilador de reloj.

- **OSC1/CLKIN:** Ocupa el pin 13, y trabaja sólo como entrada. Es un oscilador de entrada del cristal que trabaja a través de la señal externa del reloj.
- **OSC2/CLKOUT:** Ocupa el pin 14, y trabaja sólo como salida.

Las salidas del reloj del oscilador 2 deben estar a $\frac{1}{4}$ de la frecuencia del oscilador 1, que es la duración de un ciclo de máquina.

MCLR/V_{pp}: Ocupa el pin 1, y puede trabajar como entrada del *reset* o entrada de voltaje programado.

Puerto A.

Está conformado por 6 puertos bidireccionales, que pueden trabajar como entradas o salidas y son las siguientes:

- **RA0/AN0:** Ocupa el pin 2, puede actuar como línea digital de E/S o entrada analógica (input0).
- **RA1/AN1:** Ocupa el pin 3, puede actuar como línea digital de E/S o entrada analógica (input1).
- **RA2/AN2/VREF- :** Ocupa el pin 4, puede actuar como línea digital de E/S, entrada analógica (input2), o voltaje de referencia analógico negativo.
- **RA3/AN3/VREF+:** Ocupa el pin 5, puede actuar como línea digital de E/S, entrada analógica (input3), o voltaje de referencia analógico positivo.
- **RA4/TOCK1:** Ocupa el pin 6, puede actuar como línea digital de E/S o entrada de reloj para el *TIMERO*.
- **RA5/SS/AN4:** Ocupa el pin 7, puede actuar como línea digital de E/S, entrada analógica (input4), o como selector *slave* para la sincronía del puerto serial.

Puerto B.

Está conformado por 8 puertos bidireccionales, es decir que pueden trabajar como entradas o salidas, y además las entradas del PUERTO B mediante software pueden ser programadas como pull-ups. Cada puerta se describe a continuación:

- **RBO/INT:** Ocupa el pin 33, puede actuar como línea digital de E/S o interrupción externa.
- **RB1:** Ocupa el pin 34, puede actuar como línea digital de E/S.
- **RB2:** Ocupa el pin 35, puede actuar como línea digital de E/S.
- **RB3/PGM:** Ocupa el pin 36, puede actuar como línea digital de E/S y voltaje de programación de entrada alto (LVP=0) o voltaje de programación de entrada bajo (LVP=1).
- **RB4:** Ocupa el pin 37, puede actuar como línea digital de E/S.
- **RB5:** Ocupa el pin 38, puede actuar como línea digital de E/S.

- **RB6/PGC:** Ocupa el pin 39, puede actuar como línea digital de E/S o receptor de las señales de reloj dentro de la programación serie.
- **RB7/PGD:** Ocupa el pin 40, puede actuar como línea digital de E/S o entrada de datos dentro de la programación serie.

Puerto C.

Está conformado por 8 puertas bidireccionales, es decir que pueden trabajar como entradas o salidas.

- **RC0/T1OSO/T1CKI :** Ocupa el pin 11, puede actuar como línea digital de E/S, salida del oscilador del *TIMER1*, o entrada de impulsos para el *TIMER1*.
- **RC1/T1OSI/CCP2:** Ocupa el pin 12, puede actuar como línea digital de E/S, entrada al oscilador del *TIMER1*, entrada del módulo de Captura 2, salida del Comparador 2, o salida del PWM 2.
- **RC2/CCP1:** Ocupa el pin 13, puede actuar como línea digital de E/S, entrada Captura 1, salida Comparador 1, o Salida PWM 1.
- **RC3/SCK/SCL:** Ocupa el pin 14, puede actuar como línea digital de E/S, señal de reloj en modo SPI, o señal de reloj en modo I2C.
- **RC4/SDI/SDA:** Ocupa el pin 15, puede actuar como línea digital de E/S, entrada de datos en modo SPI, o línea de datos en modo I2C.
- **RC5/SDO:** Ocupa el pin 16, puede actuar como línea digital de E/S, o salida de datos en modo SPI.
- **RC6/TX/CK:** Ocupa el pin 17, puede actuar como línea digital de E/S, línea de transmisión en USART asíncrono, o señal de reloj síncrona en transmisión serie.
- **RC7/RX/DT:** Ocupa el pin 18, puede actuar como línea digital de E/S, línea de recepción del USART asíncrono, o línea de datos en transmisión serie síncrona.

Puerto D.

Está conformado por 8 puertas bidireccionales, es decir que pueden trabajar como entradas o salidas. Todos los pines disponen de una entrada de un “*Schmitt Trigger*”.

- **RD0/PSP0:** Ocupa el pin 19, puede actuar como línea digital de E/S o línea de transferencia de información del puerto paralelo esclavo (*bit 0*).
- **RD1/PSP1:** Ocupa el pin 20, puede actuar como línea digital de E/S o línea de transferencia de información del puerto paralelo esclavo (*bit 1*).
- **RD2/PSP2:** Ocupa el pin 21, puede actuar como línea digital de E/S o línea de transferencia de información del puerto paralelo esclavo (*bit 2*).
- **RD3/PSP3:** Ocupa el pin 22, puede actuar como línea digital de E/S o línea de transferencia de información del puerto paralelo esclavo (*bit 3*).
- **RD4/PSP4:** Ocupa el pin 27, puede actuar como línea digital de E/S o línea de transferencia de información del puerto paralelo esclavo (*bit 4*).
- **RD5/PSP5:** Ocupa el pin 28, puede actuar como línea digital de E/S o línea de transferencia de información del puerto paralelo esclavo (*bit 5*).
- **RD6/PSP6:** Ocupa el pin 29, puede actuar como línea digital de E/S o línea de transferencia de información del puerto paralelo esclavo (*bit 6*).
- **RD7/PSP7:** Ocupa el pin 30, puede actuar como línea digital de E/S o línea de transferencia de información del puerto paralelo esclavo (*bit 7*).

Puerto E.

Está conformado por 3 puertas bidireccionales, es decir que pueden trabajar como entradas o salidas.

-
- **RE0/RD/AN5:** Ocupa el pin 8, puede actuar como línea digital de E/S, controlar la lectura del puerto paralelo esclavo, o trabajar como entrada analógica (input5).

- **RE1/WR/AN6**: Ocupa el pin 9, puede actuar como línea digital de E/S, controlar la escritura del puerto paralelo esclavo, o trabajar como entrada analógica (input6).
- **RE2/CS/AN7**: Ocupa el pin 10, puede actuar como línea digital de E/S, seleccionar el control del puerto paralelo esclavo, o trabajar como entrada analógica (input7).

Alimentación.

- **VSS**: Ocupa el pin 8 y 19. Alimentación negativa (0 V).
- **VDD**: Ocupa el pin 20. Alimentación positiva (5 V).

2.2.4 Memorias incorporadas al Microcontrolador.

El PIC 16F877 tiene dos tipos de memoria:

2.2.4.1 Memoria de Programa.

2.2.4.2 Memoria de Datos.

2.2.4.1 Memoria de Programa.

El PIC 16F877 posee un registro Contador del Programa (PC) de 13 *bits*, y tiene la capacidad de direccionar un espacio de 8K x 14 posiciones de memoria, el mismo que está subdividido en 4 páginas de 2K x 14 posiciones como se muestra en la figura 2.2.

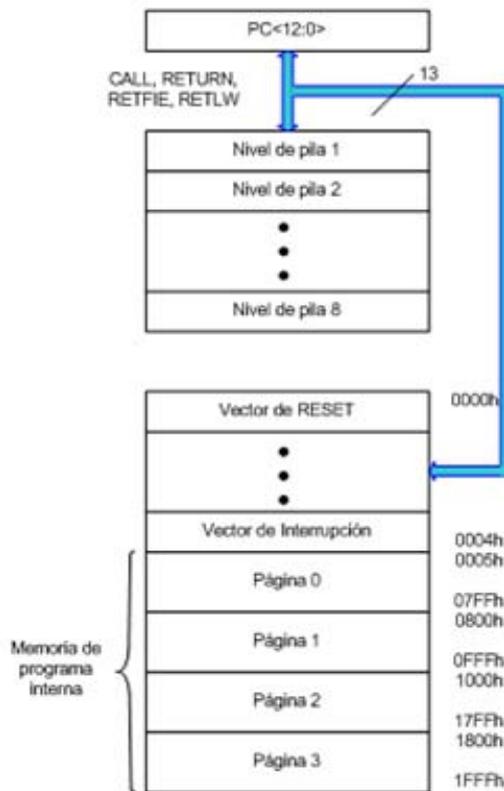


Figura 2.2 Mapa de memoria

Manejo del Contador del Programa (PC).

El contador del programa (PC) especificará una dirección de instrucción que la CPU buscará y una vez encontrada, ejecutarla.

El contador de programa consta de 13 bits, y está dividido en dos partes: el PCLATH y el PCL.

El PCLATH es un registro que se encuentra en la RAM, esta conformado por los 5 bits mas significativos (8:12) del PC. A través del PCLATH se puede leer o escribir los bits antes mencionados.

El PCL es un *byte* de orden bajo, y está conformado por el rango de los *bits* del 0 al 7 y se pueden leer y escribir.

Se puede apreciar gráficamente la estructura del contador de programa en la figura 2.3:

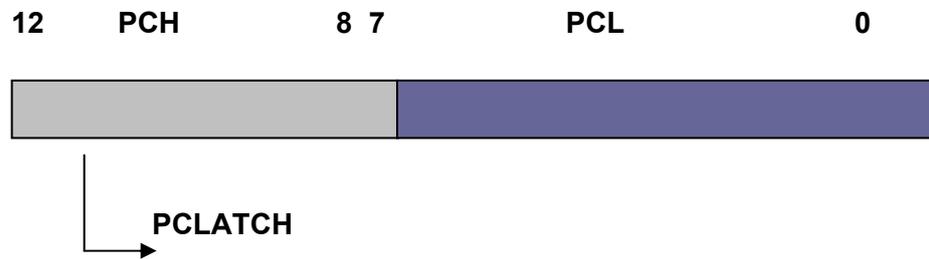


Figura 2.3 Contador de programa.

Como también se observa en la figura 2.2 el mapa de memoria está conformado por dos vectores: *reset* e interrupción; que se explican a continuación:

Vector de Reset.

Al ocurrir un *reset* el contenido del contador de programa (PC) es forzado a cero, siendo ésta la dirección donde la ejecución del programa continuará después del *reset*, por esto se le llama “dirección del vector de *reset*”.

Vector de Interrupción.

Al ocurrir una solicitud de interrupción y el CPU la acepta, provoca un salto a la dirección 0004h (el PC se carga con esta dirección), por lo que se le denomina “dirección del vector de interrupción”.

2.2.4.2 Memoria de Datos.

La Memoria de Datos está dividida en dos áreas: registros de propósito especial (SFR) y registro de propósito general (GPR).

Los SFR son localidades que están asociadas específicamente a los diferentes periféricos y funciones de configuración del PIC y tienen un nombre específico asociado con su función.

Los GPR son memoria RAM de uso general.

Bancos de Memoria.

La Memoria de Datos está organizada en cuatro bancos, cada banco tiene 7Fh posiciones de memoria (128 bytes). Las posiciones más bajas están reservadas para los registros de funciones especiales, y la parte alta para los registros de propósito general.

Los bancos están numerados del 0 al 3 y para seleccionar uno de ellos se hace uso de 2 bits del **Registro STATUS**.

Registro STATUS.

R/W	R/W	R/W	R	R	R/W	R/W	R/W
IRP	RP1	RP0	— TO#	— PD#	Z	DC	C

C: Acarreo en el 8^{vo} bit.

- 1 = Acarreo en la suma y no en la resta.
- 0 = Acarreo en la resta y no en la suma

DC: Acarreo en el 4^{to} bit menos significativo.

- 1 = Acarreo en la suma y no en la resta.
- 0 = Acarreo en la resta y no en la suma.

Z: Señalizador de cero.

- 1 = El resultado de la ultima operación aritmética o lógica, ejecutada fue cero.
- 0 = El resultado fue distinto de 0.

—

PD#: Power Down.

- 1 = Por la conexión de la alimentación, o bien, al ejecutarse la instrucción CLRWDT.
- 0 = Al ejecutarse la instrucción SLEEP.

TO#: Time Out.

1 = Por la conexión de la alimentación, o bien, al ejecutarse las instrucciones CLRWDT o *SLEEP*.

0 = Por el desbordamiento del WDT (*Watchdog Timer*).

IRP.

Se usa concatenando con el *bit* más significativo del registro FSR, para elegir el banco de RAM en el direccionamiento indirecto.

0 : Banco 0 y 1 (00h – FFh, en hexadecimal).

1 : Banco 2 y 3 (100h – 1FFh, en hexadecimal).

RP1 y RP0.

Estos *bits* sirven para la selección de los bancos de la memoria RAM, como se observa en la siguiente tabla 2.2.

BITS DEL REGISTRO DE ESTADO.		BANCOS
RP1	RP0	
0	0	BANCO 0
0	1	BANCO 1
1	0	BANCO 2
1	1	BANCO 3

Tabla 2.2 Configuración de bancos.

Acceso a la Memoria de Datos.

Para el acceso a la Memoria de Datos, se puede realizar de dos maneras:

- Direccionamiento Directo.
- Direccionamiento Indirecto.

Direccionamiento Directo.

Para ello se utiliza los 7 bits del código de instrucción, que son la dirección dentro de cualquiera de los 4 Bancos. Para seleccionar el banco se utilizan los bits RP0 y RP1 del registro STATUS.

Direccionamiento Indirecto.

La localización de memoria se direcciona mediante los registros FSR y INDF. En el registro FSR, los 7 bits menos significativos señalan la dirección, y el bit más significativo del FSR conectado con el bit IRP del registro STATUS determinan el banco. Tomando en cuenta que cualquier instrucción que haga un acceso al registro INDF en realidad se accesa a la dirección apuntada por el registro FSR.

En la figura 2.4 se resumen gráficamente los dos tipos de direccionamiento.

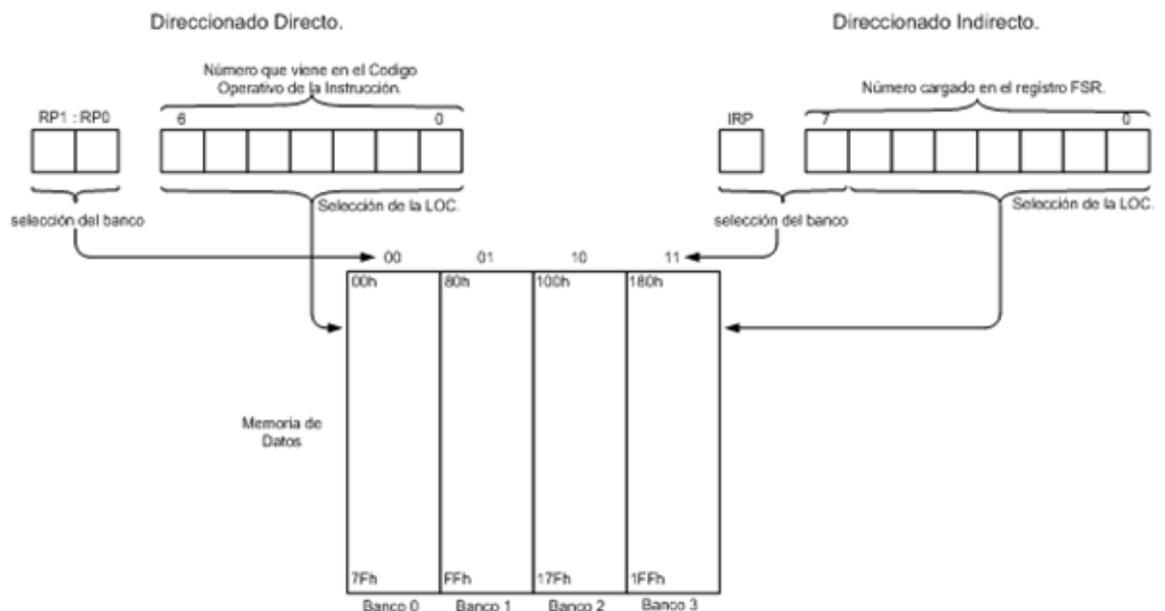


Figura 2.4 Direccionamiento directo e indirecto.

2.2.5 Interrupción por el TMR0.

Dentro del registro INTCON se encuentran tres controles de interrupciones:

- Interrupciones ocasionadas en los pines de RBO/INT.
- Interrupciones ocasionadas por cambios de estado en los pines; RB7:RB4.
- Interrupciones ocasionadas por el desbordamiento del temporizador TMR0.

Dentro de este proyecto se hizo uso de la interrupción por desbordamiento del TMR0 y su estado se registra en la bandera TOIF (*bit* 2 del registro INTCON).

Tomando en cuenta que para la activación del uso de la interrupción por desbordamiento del TMR0, el *bit* GIE (*bit* 7 del registro INTCON) debe encontrarse en uno.

2.2.6 Temporizador *TIMER0*.

El PIC 16F877 tiene 3 temporizadores:

- *TIMER0*.
- *TIMER1*.
- *TIMER2*.

Se trata únicamente el *TIMER0*, por ser el temporizador utilizado en la programación de este proyecto.

Características.

- Puede trabajar como temporizador o contador de 8 *bits*.
- Puede escribirse y leerse.
- Posee Prescaler programable por software de 8 *bits*.
- Dispone de una interrupción por desbordamiento al pasar de FFh a 00h (en hexadecimal).
- Cuenta con la selección de flanco ascendente o descendente para el flanco del reloj externo.

Modo de funcionamiento del *TIMERO* como temporizador.

Para trabajar en modo temporizador se selecciona colocando a cero el *bit T0CS* (*bit* 5 del registro *OPTION_REG*). En este modo el módulo *TIMERO* se incrementa en cada ciclo de instrucción (sin el prescaler). Si el registro *TMR0* se escribe, el incremento se inhibe durante los siguientes dos ciclos de instrucción. Por lo que se debe tomar en cuenta esto y ajustar el valor a cargar en el *TMR0*.

Resumen de registros asociados al *TIMERO*.

En la tabla 2.3 se muestran los registros principales que controlan el comportamiento del *TIMERO* y la distribución de sus *bits*.

Nombre	Dirección	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Valor en POR,BOR	Valor en el resto de Reser
INTCON	0bh, 8Bh 10Bh,18Bh	GIE	PEIE	T0IE	INTE	RBIE	TAIF	INTF	RBIF	0000 000x	0000 000x
OPTION_REG	81h,181h	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
TRISA	85h	---	---	Registro de direccionamiento de datos del PORTA						--11 1111	--11 1111

Leyenda: x = desconocido, u = inalterado; - = no implementado se lee como '0'. Las celdas sombreadas de fondo plomo y letras negras, no son usadas por el *TIMERO*

Tabla 2.3 Registros asociados al *TIMERO*.

2.2.7 Repertorio de instrucciones utilizadas en el proyecto.

El PIC 16F877 consta de 35 instrucciones, cada instrucción en lenguaje máquina (binario) del PIC tiene un código de operación **opcode** que puede ser de 3 a 4 ó 6 *bits*, dependiendo del tipo de instrucción. Estas instrucciones se pueden agrupar en 3 categorías (dentro de cada categoría se revisarán sólo las instrucciones utilizadas en este proyecto):

Instrucción CLRF.

- Esta instrucción borra el contenido del registro indicado por “f”, cargándolo con cero.
- Sintaxis: **CLRF f**
- Operandos: $0 \leq f \leq 127$
- Operación: $0x00 \rightarrow f \quad 1 \rightarrow Z$
- Ciclos de reloj: 1
- Palabras: 1
- Señalizadores:
Z = Su valor se pone en 1, si el resultado de la operación es cero.

Ejemplo 1: **CLRF REG**

Antes REG=d'18'
 Z=0

Después REG=d'0'
 Z=1

Instrucción CLRW.

- Esta instrucción borra el contenido del registro acumulador, cargándolo con 0x00.
- Sintaxis: **CLRW**
- Operandos: No tiene.
- Operación: $0x00 \rightarrow W \quad 1 \rightarrow Z$
- Ciclos de reloj: 1
- Palabras: 1
- Señalizadores:
Z = Su valor se pone en 1, si el resultado de la operación es cero.

Ejemplo 1: **CLRW**

Antes W=d'16'
 Z=0

Después W=d'0'
 Z=1

Instrucción DECF.

- Esta instrucción decrementa en una unidad el contenido del registro “f”.
- Sintaxis: **DECf f**
- Operandos: $0 \leq f \leq 127$
 $d \in [0,1]$
- Operación: $f - 1 \rightarrow \text{destino}$
- Ciclos de reloj: 1
- Palabras: 1
- Señalizadores:
Z = Su valor se pone en 1, si el resultado de la operación es cero.

Ejemplo 1: **DECf CONT,0**

Antes $W=X$ $CONT=d'18'$
Después $W=d'17'$ $CONT=d'18'$

Ejemplo 2: **DECf CONT,1**

Antes $W=X$ $CONT=d'18'$
Después $W=X$ $CONT=d'17'$

Instrucción DECFSZ.

- Esta instrucción decrementa el registro “f” en una unidad, y salta a la siguiente instrucción si el registro “f” llega a 0.
- Sintaxis: **DECFSZ f,d** “d” es el que determina dónde se almacenará el resultado.
- Operandos: $0 \leq f \leq 127$
 $d \in [0,1]$
- Operación: $f - 1 \rightarrow \text{destino}$; Salta si el resultado = 0
- Ciclos de reloj: 1 (2 si se produce el brinco)
- Palabras: 1
- Señalizadores: Ninguno.

Ejemplo 1:

Antes $FRS = d'20'$
NO **DECFSZ FSR,0** ;decrementa en 1 a FSR .
GOTO NO ; Si no es cero.
bsf REG ; Si es cero salta a esta instrucción.
Después $W = d'19'$

Instrucción INCF.

- Esta instrucción incrementa en una unidad el contenido del registro “f”.
- Sintaxis: **INCF f**
- Operandos: $0 \leq f \leq 127$
 $d \in [0,1]$
- Operación: $f + 1 \rightarrow \text{destino}$
- Ciclos de reloj: 1
- Palabras: 1
- Señalizadores:
Z = Su valor se pone en 1, si el resultado de la operación es cero al haber desbordamiento.

Ejemplo 1: **INCF CONT,0**

Antes W=X CONT=d'18'

Después W=d'19' CONT=d'18'

Ejemplo 2: **INCF CONT,1**

Antes W=X CONT=d'18'

Después W=X CONT=d'19'

Instrucción MOVF.

- El contenido del registro “f” se coloca en el destino “d”, dependiendo del valor que tenga “d”, 1 ó 0.
- Sintaxis: **MOVF f,d**
- Operandos: $0 \leq f \leq 127$
 $d \in [0,1]$
- Operación: $f \rightarrow \text{destino}$
- Ciclos de reloj: 1
- Palabras: 1
- Señalizadores:
Z = Su valor se pone en 1, si el resultado de la operación es cero.

Ejemplo 1: **MOVF REG,0**

Antes W=X REG=d'18'

Después W=d'18' REG=d'18'

Ejemplo 2: **MOVF REG,1**

Antes W=X REG=d'18'

Después W=X REG=d'18'

Instrucción MOVWF.

- Esta instrucción mueve el contenido del registro acumulador, hacia el registro "f".
- Sintaxis: **MOVWF f**
- Operandos: $0 \leq f \leq 127$
- Operación: $W \rightarrow f$
- Ciclos de reloj: 1
- Palabras: 1
- Señalizadores: Ninguno.

Ejemplo 1: **MOVWF REG**

Antes W= d'20' REG= d'18'

Después W= d'20' REG= d'20'

Instrucción NOP.

- Esta instrucción no realiza operación alguna. En realidad consume un ciclo de instrucción sin realizar ningún proceso.
- Sintaxis: **NOP**
- Operandos: No tiene.
- Ciclos de reloj: 1
- Palabras: 1
- Señalizadores: Ninguno

Ejemplo 1: **NOP**

Instrucción SUBWF.

- Esta instrucción realiza la resta, del registro acumulador al registro "f" (f-W).
- Sintaxis: **SUBWF f,d**
- Operandos: $0 \leq f \leq 127$
 $d \in [0,1]$

- Operación: $f - W \rightarrow \text{destino}$
- Ciclos de reloj: 1
- Palabras: 1
- Señalizadores:

Z = Su valor se pone en 1, si el resultado de la operación es cero.

DC = Su valor se pone en 1, si se genera un acarreo del *bit* 3 al grupo de 4 *bits* más significativos.

C = Su valor se pone en 1, si se genera un acarreo del *bit* más significativo.

Ejemplo 1: `SUBWF REG,0`

Antes $W = d'16'$ $REG = d'18'$

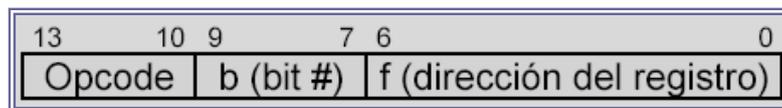
Después $W = d'2'$ $REG = d'18'$

Ejemplo 2: `SUBWF REG,1`

Antes $W = d'16'$ $REG = d'18'$

Después $W = d'16'$ $REG = d'2'$

2.2.7.2 Instrucciones orientadas al manejo de *bits*.



Las instrucciones de esta categoría utilizan la letra “b” para representar el número del *bit* dentro del registro “f”, como todos los registros son de 8 *bits* $0 \leq b \leq 7$; el registro “f” es un número entre 0 y 127 (en decimal).

Instrucción BCF.

- Esta instrucción borra el *bit* “b” del registro “f”.
- Sintaxis: `BCF f,b` “b” es el *bit* a borrar.
- Operandos: $0 \leq f \leq 127$
 $0 \leq b \leq 7$
- Operación: $0 \rightarrow f \text{ “b”}$
- Ciclos de reloj: 1
- Palabras: 1
- Señalizadores: Ninguno.

Ejemplo 1: **BCF** REG,2
 Antes REG= b'11111111'
 Después REG= b'11111**0**11'

Instrucción BSF.

- Esta instrucción coloca en 1 el *bit* “b” del registro “f”.
- Sintaxis: **BSF** f,b “b” es el *bit* a activar (set).
- Operandos: $0 \leq f \leq 127$
 $0 \leq b \leq 7$
- Operación: $1 \rightarrow f$ “b”
- Ciclos de reloj: 1
- Palabras: 1
- Señalizadores: Ninguno.

Ejemplo 1: **BSF** REG,3
 Antes REG= b'00000000'
 Después REG= b'0000**1**000'

Instrucción BTFSC.

- Esta instrucción explora un *bit* “b” del registro “f” y salta a la siguiente instrucción, si “b” vale 0.
- Sintaxis: **BTFSC** f,b “b” es el *bit* a explorar.
- Operandos: $0 \leq f \leq 127$
 $0 \leq b \leq 7$
- Operación: Salta si $f^b = 0$
- Ciclos de reloj: 1 (2 si se produce el salto)
- Palabras: 1
- Señalizadores: Ninguno.

Ejemplo 1: PORTA = b'11110111'

```
NO   BTFSC     PORTA,3;pregunta: ¿es cero el bit 3 del registro PORTA?
      GOTO NO           ;si no es cero salta a una subrutina NO.
      incf  REG         ;si es cero salta a esta instrucción.
```

Ejemplo 2: PORTA = b'11111111'

```
NO   BTFSC     PORTA,3;pregunta: ¿es cero el bit 3 del registro PORTA?
      GOTO NO           ;si no es cero salta a una subrutina NO.
      incf  REG         ;si es cero salta a esta instrucción.
```

Instrucción BTFSS.

- Esta instrucción explora un *bit* “b” del registro “f” y salta a la siguiente instrucción, si “b” vale 1.
- Sintaxis: **BTFSS** f,b “b” es el *bit* a explorar.
- Operandos: $0 \leq f \leq 127$
 $0 \leq b \leq 7$
- Operación: Salta si $f \text{ "b"} = 1$
- Ciclos de reloj: 1 (2 si se produce el salto)
- Palabras: 1
- Señalizadores: Ninguno.

Ejemplo 1: PORTA = b'11111111'

```
NO   BTFSSPORTA,3 ;pregunta: ¿es uno el bit 3 del registro PORTA?
      GOTO NO           ;si es cero salta a una subrutina NO.
      incf  REG         ;si es uno salta a esta instrucción.
```

Ejemplo 2: PORTA = b'11110111'

```
NO   BTFSSPORTA,3 ;pregunta: ¿es uno el bit 3 del registro PORTA?
      GOTO NO           ;si es cero salta a una subrutina NO.
      incf  REG         ;si es uno salta a esta instrucción.
```

2.2.7.3 Instrucciones orientadas a operaciones con literales y de control.



Estas instrucciones utilizan el operador k, que es un número de 8 ó 9 bits, para realizar operaciones aritméticas y lógicas sobre literales, y para cambiar el contenido del contador de programa.

Instrucción CALL.

- Esta instrucción llama a una subrutina que tiene por dirección el valor de “k”.
- Sintaxis: **CALL k**
- Operandos: $0 \leq k \leq 2047$
-
- Operación: $PC+1 \rightarrow \text{Pila}; k \rightarrow PC$
- Ciclos de reloj: 2
- Palabras: 1
- Señalizadores: Ninguno.

Ejemplo 1: ORIGEN **CALL** DESTINO

Antes PC = ORIGEN

Después PC = DESTINO

Instrucción GOTO.

- Esta instrucción realiza un salto incondicional a la dirección especificada por “k”.
- Sintaxis: **GOTO k**
- Operandos: $0 \leq k \leq 2047$
- Operación: $k \rightarrow PC$
- Ciclos de reloj: 2
- Palabras: 1
- Señalizadores: Ninguno.

Ejemplo 1: ORIGEN **GOTO** DESTINO

Antes PC = ORIGEN

Después PC = DESTINO

Instrucción **RETFIE**.

- Esta instrucción realiza un retorno de interrupción, el PC se carga con el contenido de la cima de la pila (TOS), que es la dirección de retorno. Coloca GIE=1 y las interrupciones vuelven a ser habilitadas.
- Sintaxis: **RETFIE**
- Operandos: No tiene
- Operación: TOS \rightarrow PC 1 \rightarrow GIE
- Ciclos de reloj: 2
- Palabras: 1
- Señalizadores: Ninguno.

Ejemplo 1: **RETFIE**

Después PC = dirección de retorno

GIE =1

Instrucción **RETLW**.

- Esta instrucción realiza un retorno de subrutina, carga el valor de “k” en el registro acumulador, el PC se carga con el contenido de la cima de la pila (TOS), que es la dirección de retorno.
- Sintaxis: **RETLW k**
- Operandos: $0 \leq k \leq 255$
- Operación: k \rightarrow W TOS \rightarrow PC
- Ciclos de reloj: 2
- Palabras: 1
- Señalizadores: Ninguno.

Ejemplo 1: **RETLW 16**

Después PC = dirección de retorno

W = 16

Instrucción RETURN.

- Esta instrucción realiza un retorno de una subrutina, el PC se carga con el contenido de la cima de la pila (TOS), que es la dirección de retorno.
- Sintaxis: **RETURN**
- Operandos: No tiene
- Operación: TOS \rightarrow PC
- Ciclos de reloj: 2
- Palabras: 1
- Señalizadores: Ninguno.

Ejemplo 1: **RETURN**

Después PC = dirección de retorno

Instrucción MOVLW.

- Esta instrucción mueve el contenido de “k” hacia el registro acumulador.
- Sintaxis: **MOVLW k**
- Operandos: $0 \leq k \leq 255$
- Operación: k \rightarrow W
- Ciclos de reloj: 1
- Palabras: 1
- Señalizadores: Ninguno.

Ejemplo 1: **MOVLW d'18'**

Antes W = X

Después W = d'18'

Instrucción SUBLW.

- Esta instrucción resta el contenido del registro acumulador al contenido de “k”, guardando el resultado en el registro acumulador.
- Sintaxis: **SUBLW k**
- Operandos: $0 \leq k \leq 255$

- Operación: $k - W \rightarrow W$
- Ciclos de reloj: 1
- Palabras: 1
- Señalizadores:

Z = Su valor se pone en 1, si el resultado de la operación es cero.

DC = Su valor se pone en 1, si se genera un acarreo del *bit* 3 al grupo de 4 *bits* superior.

C = Su valor se pone en 1, si se genera un acarreo del *bit* más significativo.

Ejemplo 1: **SUBLW** d'18'

Antes W = d'16'

Despues W = d'2'

2.3 Display de Cristal Líquido LCD.

El Display de Cristal Líquido LCD, es un dispositivo que tiene la función de visualizar textos y/o gráficos. Su estructura general consiste de un cristal de tinta líquida y un controlador electrónico inteligente montado en una tarjeta de circuito impreso, que se encuentra al reverso del cristal.

En el manejo del LCD orientado a la visualización de caracteres de texto, se debe tomar en cuenta que, cada caracter está formado por una matriz de puntos, y a su vez cada matriz dentro del LCD posee una dirección específica. Por lo tanto cada matriz tiene una dirección de memoria, que puede leerse como escribirse, a esta memoria se le denomina DDRAM (*Display Data RAM*). A la vez los LCDs posee otra memoria llamada ROM (*Read Only Memory*) la misma que es directamente direccionable, escribiendo el caracter a visualizar. Pero antes de proceder a escribir el caracter, se debe escribir la dirección (posición de visualización) dentro de la memoria DDRAM.

2.3.1 Modos de operación de un LCD.

Por lo general los LCDs pueden operar de tres formas:

- Mediante un interfase de 4 *bits* de datos y 3 líneas de control, con transferencia de datos o instrucciones de 4 *bits* en 4 *bits*, utilizando los 4 *bits* más significativos del puerto conectado al LCD (*bits* del 4 al 7).
- Mediante un interfase de 4 *bits* de datos y 3 líneas de control, con transferencia de datos o instrucciones de 4 *bits* en 4 *bits*, utilizando los *bits* menos significativos del puerto conectado al LCD (*bits* del 0 al 3).
- Mediante un interfase de 8 *bits* de datos y 3 líneas de control, tomando en cuenta que en este caso los datos e instrucciones se envían en 8 *bits*, es decir utilizando todo el puerto conectado al LCD (*bits* del 0 al 7).

2.3.2 Comandos.

A continuación se muestran en la tabla 2.4 los comandos generales de un LCD.

Comando	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Borra pantalla	0	0	0	0	0	0	0	0	0	1
Cursor a casa	0	0	0	0	0	0	0	0	1	*
Modo introducción	0	0	0	0	0	0	0	1	I/D	S
Pantalla on/off	0	0	0	0	0	0	1	D	C	B
Modo desplazamiento	0	0	0	0	0	1	S/C	R/L	*	*
Función	0	0	0	0	1	DL	Líneas	Font	*	*
Dirección CGRAM	0	0	0	1	Dirección CGRAM					
Dirección DDRAM	0	0	1	Dirección DDRAM						
Lectura ocupado y dirección contador	0	1	BF	Dirección DDRAM						
Escribe RAM	1	0	Escribe dato							
Lee RAM	1	1	Lee dato							

Tabla 2.4 Comandos generales de un LCD.

Borra Pantalla: Borra la pantalla y sitúa el cursor en su posición inicial (0).

Inicializar Cursor: El cursor va hacia la posición inicial (0), pero sin borrar nada.

Modo instrucción: Configura la dirección del cursor I/D. Cuando I=1 incrementa la posición del cursor, y D=0 la decrementa. Mientras S=1 significa que hay desplazamiento en la pantalla. La operación se ejecuta durante la E/S de los datos.

Pantalla On/Off: Coloca en movimiento al cursor, o genera desplazamiento en la pantalla (D para toda la pantalla, C para cursor On/Off, y B hace parpadear el cursor).

Desplazamiento Cursor/Pantalla: S/C indica el movimiento del cursor, o desplazamiento en la pantalla, R/L indica la dirección del cursor (derecha/izquierda). No varía el contenido de la DDRAM.

Función: DL indica la longitud de datos del interfase; N el número de líneas de la pantalla y F el tipo de caracteres.

Dirección CGRAM: Coloca el dato enviado o recibido en la CGRAM, después de la ejecución de este comando.

Dirección DDRAM: Coloca el dato enviado o recibido en la DDRAM, después de la ejecución de este comando.

Bandera de ocupado BF: Lee BF indicando si hay una operación interna en curso y lee además el contenido de la dirección contador.

Escribe RAM: Escribe un dato en la RAM (ya sea DDRAM o CGRAM).

Lee RAM: Lee datos de la RAM (ya sea DDRAM o CGRAM).

2.3.3 Líneas de control.

Son 3 principales:

- Línea de Control E.
- Línea de Control RS.
- Línea de Control R/W.

Línea de control “E” (*Enable*).

Se debe tomar en consideración lo siguiente:

- Al estar el flanco de subida de esta señal, se procede a almacenar el estado de los pines RS y RW.
- Al estar el flanco de bajada de esta señal, se almacena el dato o la instrucción que haya sido colocado en el bus del LCD.
- En el manejo del LCD, primeramente se debe colocar el dato o la instrucción en el puerto y luego dar un pulso al LCD a través de E. La duración del pulso debe ser como mínimo 250 ns.

Línea de control “RS” (*Register Selection*) .

- RS=1 Significa que la palabra que se colocó en el bus del LCD es un dato.
- RS=0 Significa que la palabra que se colocó en el bus del LCD es una instrucción.

Línea de control “R/W” (*Read/Writer*).

Los datos o instrucciones manejados por la línea de control RS, necesitan ser leídos o escritos; donde:

- R/*W =0 Se procede a escribir en el LCD.
- R/*W =1 Se procede a leer el LCD.

2.3.4 Pinaje del LCD utilizado en el proyecto.



- **PIN 1 - VSS:** Alimentación negativa del módulo LCD.
- **PIN 2 - VDD:** Alimentación positiva del módulo LCD. La tensión utilizada es de 5 voltios.
- **PIN 3 - VC:** Voltaje de ajuste del contraste. Este se puede ajustar mediante un potenciómetro regulando así la intensidad de los caracteres; donde a mayor tensión mayor intensidad. El potenciómetro por lo general puede ser de 10 ó 20 K.
- **PIN 4 - RS:** Este indica si la información que le llega es una instrucción (RS=0), o si la información es un caracter (RS=1).
- **PIN 5 - R/W:** Si el módulo está en 0 se escribe en pantalla el dato que existe en el bus de datos, y si está en 1 se leerá lo que está en el bus de datos.
- **PIN 6 - E:** Este habilita o no el funcionamiento del módulo LCD; es decir si E=0 se deshabilita el LCD y si E=1 entonces se podrá transferir datos y realizar las operaciones correspondientes al módulo.
- **PIN 7 - D0:** *Bit* de datos menos significativo.
- **PIN 8 - D1:** *Bit* de datos.
- **PIN 9 - D2 :** *Bit* de datos.
- **PIN 10 - D3 :** *Bit* de datos.
- **PIN 11 - D4 :** *Bit* de datos.
- **PIN 12 - D5 :** *Bit* de datos.

- **PIN 13 - D6** : *Bit* de datos.
- **PIN 14 - D7** : *Bit* de datos más significativo.
- **PIN 16 Y 17** : Para controlar *back Light*.

CAPITULO III

ANALISIS, DISEÑO Y CONSTRUCCION DE UN CIRCUITO DE CONTROL DE CONMUTACION AUTOMATIZADO, PARA EL EQUIPO DE MEDICION DE LAS CARACTERISTICAS ELECTRICAS DE LOS CABLES TELEFONICOS, FLUKE 189.

3.1 Introducción.

El Circuito de Control de Conmutación Automatizado, conjuntamente con el *software* y *hardware* del Fluke 189 (equipo de medición), permitirán tomar las mediciones de las características eléctricas de las líneas telefónicas que, en fase de prueba contempla la ejecución de un grupo de 10 pares. Para ello, el circuito conmuta par a par, y sus mediciones son capturadas por el téster Fluke 189 y registradas en una base de datos de su *software*.

Se debe tomar en cuenta que, entre el circuito de conmutación, el *software* y *hardware* del Fluke 189, existirá compatibilidad; es decir sincronía con lo que respecta al tiempo. En cuanto al acoplamiento de estos tres elementos están dados de la siguiente forma: el circuito de conmutación se une al téster Fluke189 a través de las puntas de prueba, y este téster a la vez establece comunicación con la base de datos residente en una computadora, mediante el protocolo de comunicación RS232 (figura 3.1).

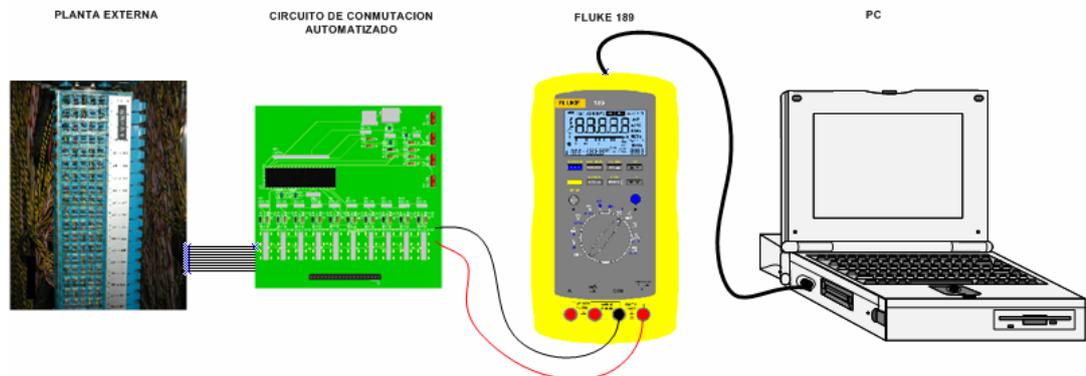


Figura 3.1. Esquema de acoplamiento de los equipos, para la toma de datos y almacenamiento.

El circuito de conmutación está diseñado con componentes tanto analógicos como digitales, que permiten tomar los valores reales de las mediciones de las características que presentan las líneas telefónicas. Para el diseño de la placa o PCB este circuito, se utilizó el *software* PROTEL.

El control del circuito de conmutación se basa en la tecnología microcontrolada; es decir se hace uso de un microcontrolador PIC 16F877; y su programación se realiza a través del *software* MPLAB. Esta programación o programa se debe guardar en el PIC mediante un grabador de PICs, que en el presente caso se utilizó el *software* y *hardware* del IC-PROG.

Para el desarrollo del Circuito de Control de Conmutación Automatizado, se ha dividido en el estudio de los siguientes puntos:

3.2. Software de Apoyo.

3.3. Software del Circuito de Control de Conmutación Automatizado.

3.4. PCB del Circuito de Control de Conmutación Automatizado.

3.2 Software de apoyo.

Para el desarrollo del circuito de conmutación se ha utilizado como herramientas primordiales los siguientes *software*:

3.2.1 MPLAB.

3.2.2 IC-PROG.

3.2.3 SOFTWARE FLUKE 189.

3.2.4 PROTEL.

3.2.1 MPLAB.

MPLAB (figura 3.2) es un *software* que permite desarrollar programas para una serie de microcontroladores o PICs (como el PIC 16F877 utilizado en este proyecto), ya que contempla un entorno de desarrollo integrado que permite escribir y codificar los microcontroladores PIC de *Microchip* para ejecutarlos.

Además MPLAB incluye un editor de texto, funciones para el manejo de proyectos, simulador interno y herramientas que admiten mantener y ejecutar ciertas aplicaciones, también provee una interfase de usuario para todos los productos con lenguaje *Microchip*, programadores de dispositivos, sistemas emuladores y herramientas de tercer orden.

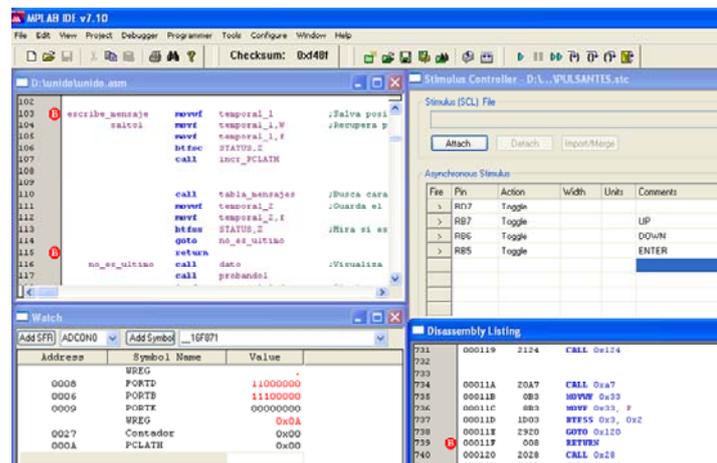


Figura 3.2 Ventanas de simulación del MPLAB.

Por otra parte MPLAB permite además realizar las siguientes tareas:

- Manejar el escritorio de MPLAB.
- Crear un nuevo archivo de código fuente para el ensamble e ingresarlo a un nuevo proyecto.
- Compilación (identificación de errores).
- Simulador de códigos.

3.2.2. IC-PROG.

IC-PROG (figura 3.3) permite grabar un programa de extensión “.hex”, hacia el microcontrolador PIC; para ello es necesario hacer uso de su *software* y *hardware*.

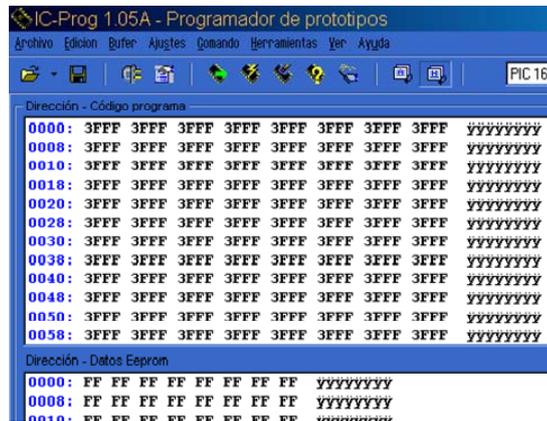


Figura 3.3 IC-PROG, software del grabador de microcontroladores PICs.

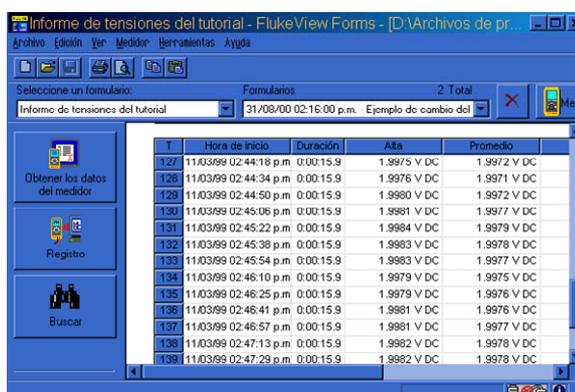
La función específica del *software* IC-PROG dentro de este proyecto, es abrir un archivo de extensión “.hex” desarrollado en MPLAB, mientras que la función del *hardware* IC-PROG o circuito electrónico (anexo 2), es realizar la programación física del microcontrolador.

3.2.3 SOFTWARE FLUKE 189.

El equipo de medición Fluke 189 o téster (anexo 1), está complementado por su *software FlukeView Forms*, el cuál constituye una parte importante en el desarrollo de este proyecto, centrándose básicamente en las opciones fundamentales que posee, lo que ha permitido el desarrollo del Circuito de Control de Conmutación Automatizado.

Base de datos y formularios.

Este punto hace referencia al manejo de la información, y para ello utilizan las plantillas del *FlukeView Forms* (figura 3.4), que sirven de soporte de la información y permiten exhibir los datos de distintas maneras. Entonces al transferir los datos del medidor o téster Fluke 189, el programa los muestra en el formulario activo, previamente escogido. Este formulario exhibido en pantalla y en el cuál se han registrado los datos que transfiere el medidor Fluke 189, se convierte en una base de datos, la misma que puede ser guardada.



T	Hora de inicio	Duración	Acta	Promedio
127	11/03/99 02:44:18 p.m.	0:00:15.9	1.9975 V DC	1.9972 V DC
128	11/03/99 02:44:34 p.m.	0:00:15.9	1.9976 V DC	1.9971 V DC
129	11/03/99 02:44:50 p.m.	0:00:15.9	1.9980 V DC	1.9972 V DC
130	11/03/99 02:45:06 p.m.	0:00:15.9	1.9981 V DC	1.9977 V DC
131	11/03/99 02:45:22 p.m.	0:00:15.9	1.9984 V DC	1.9979 V DC
132	11/03/99 02:45:38 p.m.	0:00:15.9	1.9983 V DC	1.9978 V DC
133	11/03/99 02:45:54 p.m.	0:00:15.9	1.9983 V DC	1.9977 V DC
134	11/03/99 02:46:10 p.m.	0:00:15.9	1.9979 V DC	1.9975 V DC
135	11/03/99 02:46:25 p.m.	0:00:15.9	1.9979 V DC	1.9976 V DC
136	11/03/99 02:46:41 p.m.	0:00:15.9	1.9981 V DC	1.9976 V DC
137	11/03/99 02:46:57 p.m.	0:00:15.9	1.9981 V DC	1.9977 V DC
138	11/03/99 02:47:13 p.m.	0:00:15.9	1.9982 V DC	1.9978 V DC
139	11/03/99 02:47:29 p.m.	0:00:15.9	1.9982 V DC	1.9978 V DC

Exportar Datos.

Una de las ventajas principales con la que cuenta el *FlukeView Forms* es la opción de “Exportar Datos” (figura 3.5), es decir que la base de datos puede ser guardada bajo la extensión “*.csv”, y de esta forma poder trabajar con la base de datos en otras aplicaciones compatibles con ella; como por ejemplo Excel.

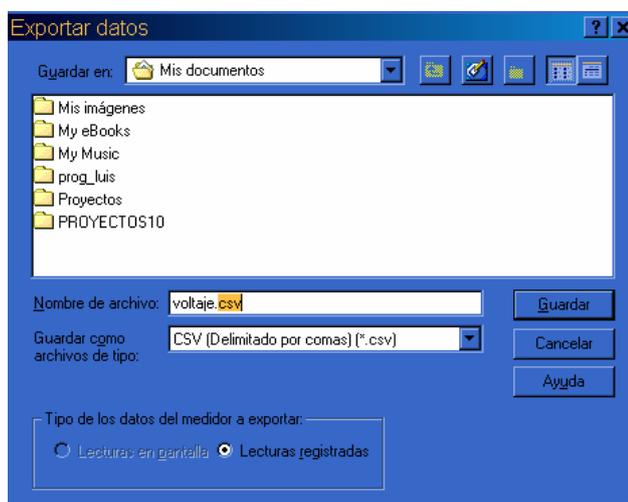


Figura 3.5 Ventana de la opción exportar datos.

Registro.

Esta opción (figura 3.6) es una de las más relevantes, debido a que a través de ella podemos realizar la configuración del medidor o téster Fluke 189, en lo que respecta al tiempo de captura de las lecturas (información) del medidor hacia la base de datos.

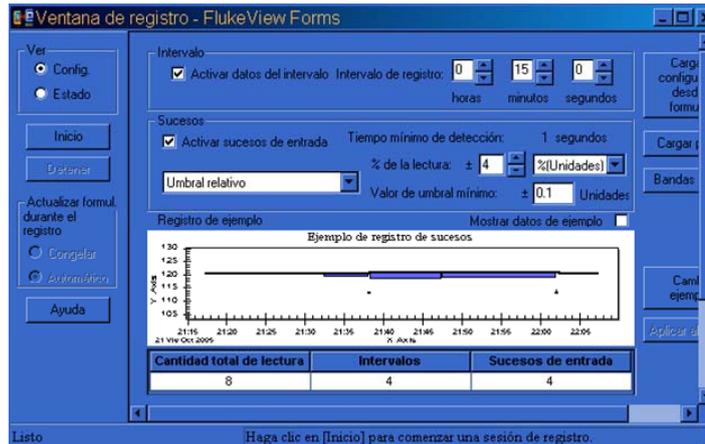


Figura 3.6 Ventana de la opción registro.

Analizando la ventana de la opción de Registro en modo de Configuración, se observan dentro de ella algunas opciones, entre las más importantes para el proyecto:

- Intervalos.
- Sucesos.

Intervalos.

La opción “Intervalos” (figura 3.7), está compuesta por horas, minutos y segundos, modificables, y que representa el intervalo de tiempo o el tiempo de espera para capturar y registrar una lectura del medidor hacia la base de datos. Es decir si por ejemplo la configuración está dada por Horas:1 Minutos:2 y segundos: 3 quiere decir que cada 1 hora con 2 minutos y 3 segundos tomará una nueva lectura del medidor.

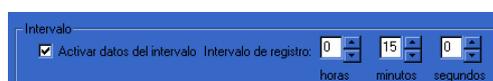


Figura 3.7 Opción intervalos.

Sucesos.

La opción “Sucesos” (figura 3.8) , es la más importante, ya que a través de ella se ha podido resolver el problema de sincronización de reloj entre el Circuito de Control de Conmutación Automatizado y el sistema (*software* y *hardware*) del Fluke 189.

Esta opción permite censar automáticamente un cambio de medición, o la toma de otra lectura, enviándola inmediatamente a la base de datos.

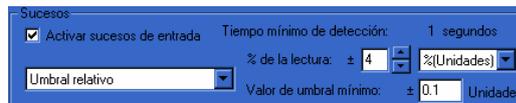


Figura 3.8 Opción de sucesos.

3.2.4. PROTEL.

PROTEL (figura 3.9) es un programa que cuenta con herramientas que han permitido crear, diseñar y generar el PCB (circuito impreso) del Circuito de Control de Conmutación Automatizado.

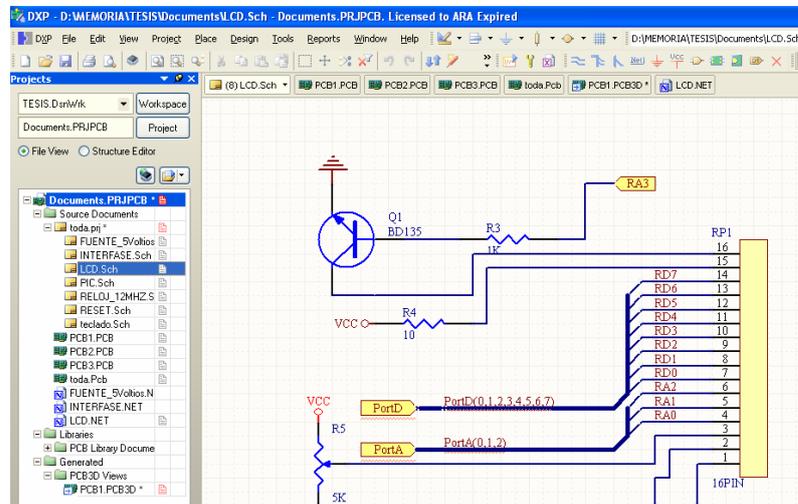


Figura 3.9 Esquema de un circuito desarrollado en PROTEL.

Este programa cuenta con los siguientes componentes :

- Editor de esquemáticos.
- Paneles de navegación, inspección, listado de objetos dentro del documento actual, errores tanto de compilación como de librerías.
- Editor de componentes esquemáticos y huellas (*foot prints*).
- Editor de PCBs.
- Visualización en tercera dimensión de PCBs.
- Herramientas de simulación.

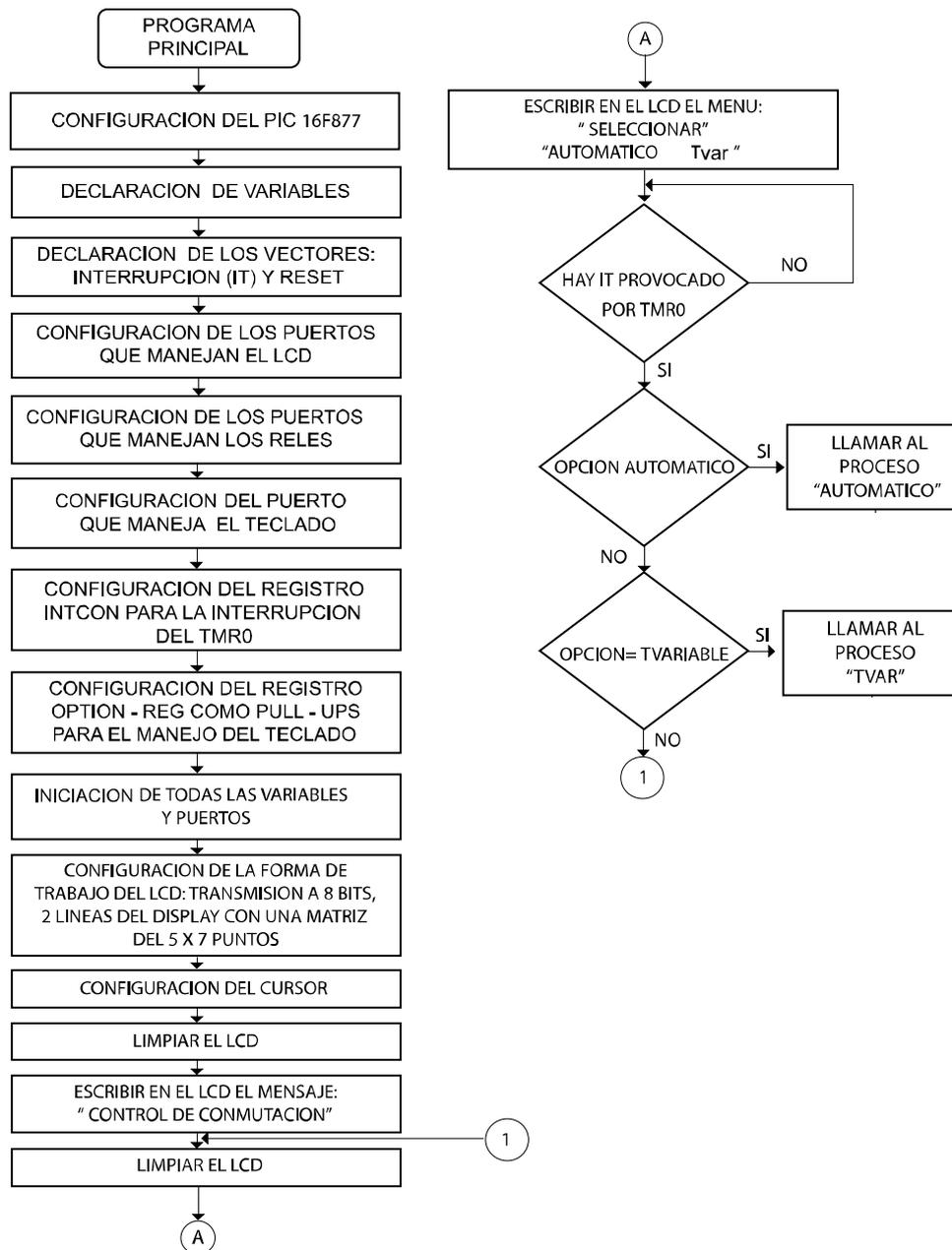
3.3 Software del Circuito de Control de Conmutación Automatizado.

Antes de pasar al código de programa del Circuito de Control de Conmutación Automatizado (desarrollado en ensamblador), se explicará de manera general en que consiste su lógica de programación, mediante un flujograma. Este Flujograma está dividido en tres partes:

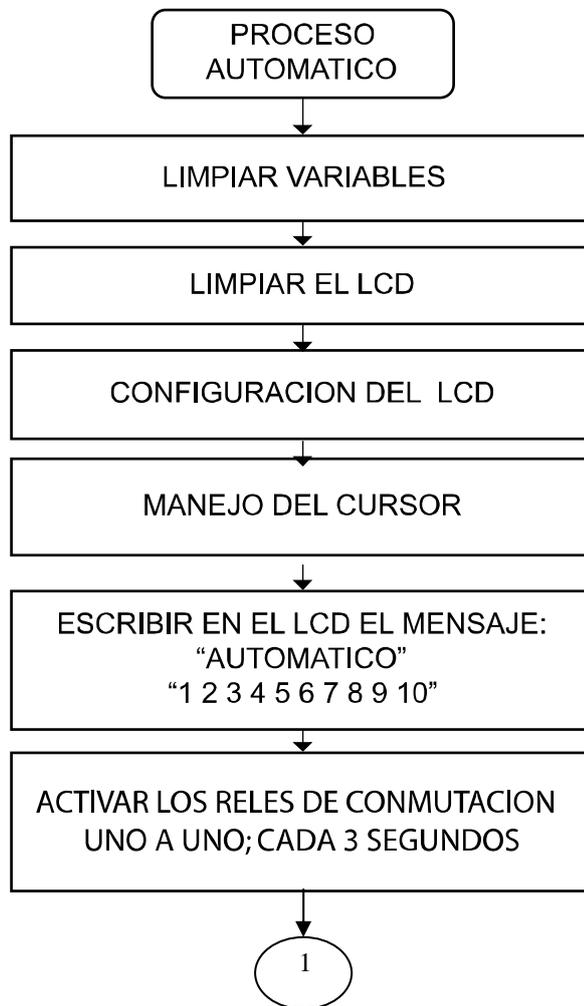
La primera parte contempla en sí el programa principal, basado en un menú de las opciones: automático y tiempo variable (Tvar). La opción automático que es la segunda parte, hace referencia a la conmutación de los 10 pares telefónicos cada 3 segundos, mientras que la tercera parte que es el tiempo variable conmuta los 10 pares telefónicos, según el tiempo escogido por el usuario (entre 3 y 9 segundos).

3.3.1 Flujoograma del Circuito de Control de Conmutación Automatizado.

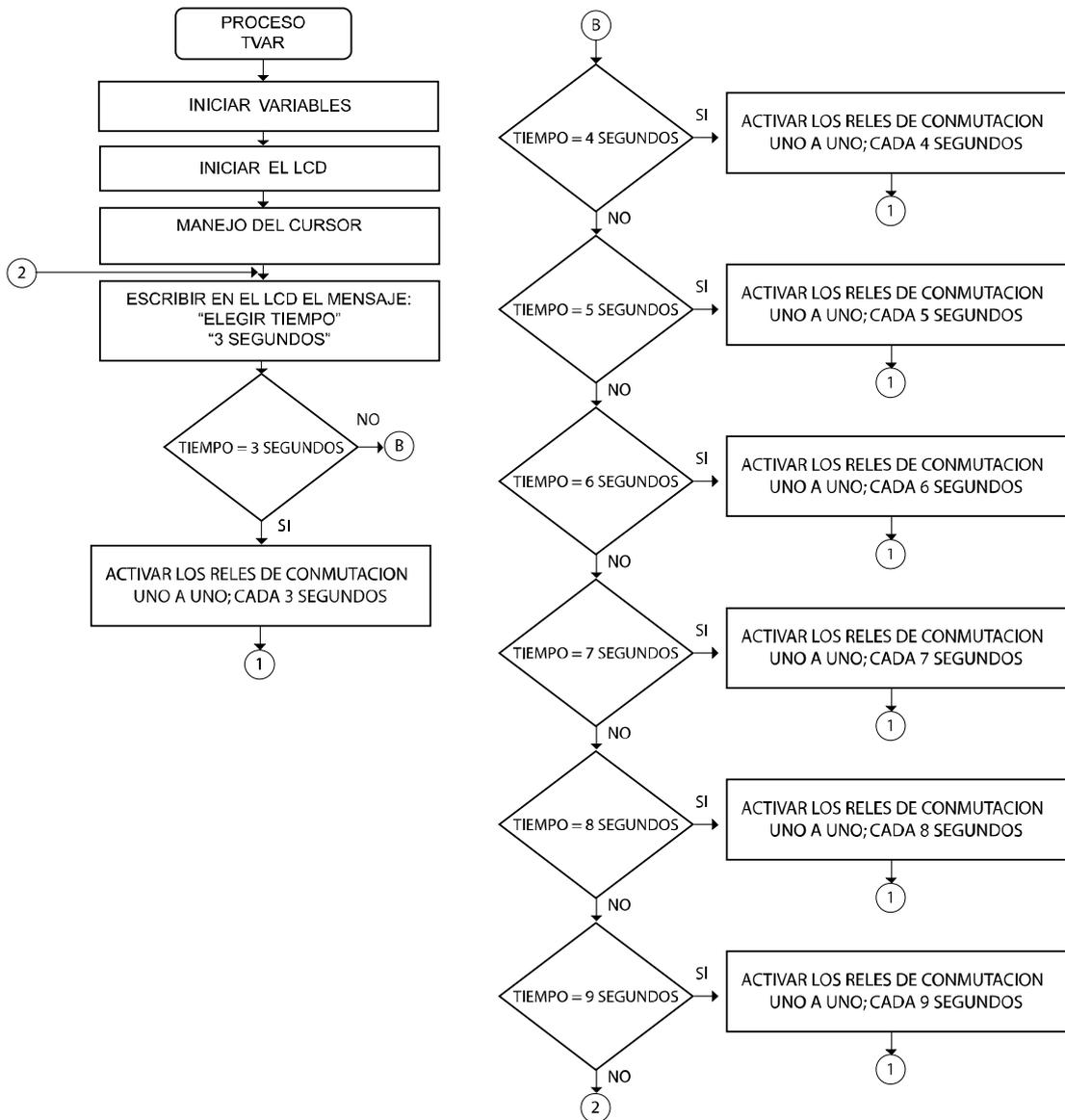
Programa Principal.



Opción Automático.



Opción Tiempo Variable (Tvar).



3.3.2 Código de Programa del Circuito de Control de Conmutación Automatizado.

```

;Control de Conmutación Automatizado.
;Se utiliza las teclas RB7=UP      RB6=DOWN      RB5=ENTER
;Las teclas se activan con ceros
;El LCD transmite a 8 bits
;Los tiempos están calculados para un reloj de 12MHZ

list p=16f871          ;Se indica el tipo de procesador
#include <p16f871.inc>  ;Se incluye la definición de los registros
                       ;internos en una librería
__config 3939         ;Palabra de configuración

LCD_maestro_var EQU 0x20 ;Declaración de variables
Contador EQU 0x29
temporal_1 EQU 0x2A
temporal_2 EQU 0x2B

org 0x00          ;Inicio en el Vector de Reset
goto inicio
org 0x04          ;Vector de Interrupción
goto IT

include"LCD_maestro.inc" ;Incluye rutinas de LCD

;-----
;TABLA DE CODIGOS

;Según el valor contenido en el acumulador, se devuelve el caracter a visualizar
tabla_mensajes movwf PCL ;Calcula el desplazamiento sobre la tabla
;-----
;La directiva dt genera tantas instrucciones retlw como bytes o caracteres ;contenga

mensaje_a1 equ $
dt "CONTROL",0x00

mensaje_a2 equ $
dt "DE CONMUTACION",0x00

mensaje_b1 equ $
dt " SELECCIONAR",0x00

mensaje_b2 equ $
dt " Automático Tvar",0x00

mensaje_c1 equ $
dt " Automático",0x00

mensaje_c2 equ $
dt " 123456789 10",0x00

mensaje_d1 equ $
dt " Elegir tiempo",0x00

```

```

mensaje_d2 equ $
             dt    " 3seg",0x00

mensaje_e1 equ $
             dt    "CONMUTA CADA sg",0x00

;-----
;La rutina "escribe_mensaje" visualiza en el LCD el mensaje cuyo inicio está
;indicado en el acumulador.
;El fin de un mensaje se determina mediante el código 0x00

escribe_mensaje
    movwf temporal_1      ;Salva posición de la tabla en "temporal_1"
salto1 movf  temporal_1,W  ;Recupera posición de la tabla
    movf  temporal_1,f
    btfsc STATUS,Z        ;¿Llegó "temporal_1" a cero?
    call  incr_PCLATH     ;Sí. Cambia a la página siguiente
    call  tabla_mensajes  ;No. Busca caracter de salida
    movwf temporal_2      ;Salva caracter en "temporal_2"
    movf  temporal_2,f
    btfss STATUS,Z        ;¿Es el último caracter?
    goto  no_es_ultimo    ;No. Busca siguiente caracter
    return                ;Sí. Retorno de rutina

no_es_ultimo
    call  dato            ;Visualiza en el LCD
    call  tmp             ;Tiempo para escribir el siguiente caracter
    incf  temporal_1,f    ;Incremento "temporal_1" para encontrar el
                          ;siguiente caracter
    goto  salto1          ;Repite el ciclo

incr_PCLATH
    movlw b'00001'       ;Página 1
    movwf PCLATH
    clrw                    ;Limpia acumulador
    return                ;Retorno de rutina

;-----
;CONFIGURA REGISTROS
inicio
    call  cfg_pic_LCD     ;Configuración de los puertos para
                          ;manejar el LCD
    bsf   STATUS,RP0      ;Banco 1
    bcf   STATUS,RP1
    clrf  TRISE           ;PORTE como salida, para manejar los relés
    clrf  TRISC           ;PORTC como salida, para manejar los relés
    movlw b'11100000'     ;PORTB como entrada, para manejar el teclado
    movwf TRISB           ;bit 7 Entrada UP, bit 6 Entrada DOWN,
                          ;bit 5 Entrada ENTER
    movlw b'00000111'     ;Pull-up activado, TMR0=1:16
    movwf OPTION_REG

;-----
;LIMPIA REGISTROS Y VARIABLES

    bcf   STATUS,RP0     ;Banco 0

```

```

bcf STATUS,RP1
clrf PORTA
clrf PORTD
clrf PORTE
clrf PORTC
clrf CONT1
clrf CONT2
clrf CONT3
clrf N
clrf M
clrf O
clrf P
clrf Contador
clrf TMR0
clrf temporal_1
clrf temporal_2
clrf PCLATH

```

```

;-----
;INICIA LCD Y ESCRIBE PRIMER CARTEL

```

```

call inicia_LCD ;Secuencia de inicio del LCD
call DISPLAY_ONOFF_CONTROL_1 ;Display on, cursor off,
;cursor no parpadea
call SET_THE_DDRAM_ADDRESS_80;Empieza a escribir desde la
;dirección 0X80
call DISPLAY_CLEAR ;Limpia LCD
movlw mensaje_a1 ;Carga la dirección de
;"mensaje_a1" al acumulador
call escribe_mensaje ;Escribe el mensaje
call SET_THE_DDRAM_ADDRESS_C0;Empieza a escribir desde la
;dirección 0xC0
movlw mensaje_a2 ;Carga la dirección de
;"mensaje_a2" al acumulador
call escribe_mensaje ;Escribe el mensaje
call tiempo ;Espera un tiempo para que
;sea leído el mensaje
call desplaza_display ;Desplaza los mensajes

```

menú

```

clrf PCLATH ;Página 0
clrf Contador ;Limpia Contador
clrf PORTC ;Limpia Puerto C
clrf PORTE ;Limpia Puerto E
call DISPLAY_CLEAR ;Limpia LCD
movlw mensaje_b1 ;Carga la dirección de
;"mensaje_b1" al acumulador
call escribe_mensaje ;Escribe el mensaje
call SET_THE_DDRAM_ADDRESS_C0;Empieza a escribir desde la
;dirección 0XC0
movlw mensaje_b2 ;Carga la dirección de
;"mensaje_b2" al acumulador
call escribe_mensaje ;Escribe el mensaje
movlw d'1'
movwf PCLATH ;Página 1

```

```

movf Contador,w
call cur_tab ;Selecciona: Automático (0XC1)
;o Tiempo variable (0XCC)
call pos_cur ;Ubica al cursor en la
;dirección C1 o CC del LCD
call DISPLAY_ONOFF_CONTROL_2 ;Display on, cursor on, cursor
;parpadea
movlw b'10100000' ;GIE=1 permiso de interrupción
;global, TOIE=1 permiso de
;interrupción del TMR0

movwfINTCON
;-----
;INTERRUPCION

lazo goto lazo ;bucle

IT ;Subrutina de Atención a Interrupción
btfss INTCON,TOIF ;¿Se desbordó el TMR0?
goto regresa ;No. Falsa interrupción
bcf STATUS,RP0 ;Sí. Salio del bucle

;-----
;ESCANEA TECLAS

UP
btfsc PORTB,7 ;¿Está pulsada la tecla UP ?
goto DOWN ;No. Salta a "DOWN"
call Delay_20_ms ;Sí. Elimina rebotes
btfss PORTB,7 ;¿A dejado de pulsar la tecla UP ?
goto UP ;No. Salta a "UP"
call Delay_20_ms ;Sí. Elimina rebotes

;-----
;TECLA UP

incf Contador,f
movlw d'2' ;Carga al acumulador con 2, por que son dos
;opciones del menú "Automático y Tvar"

subwf Contador,w
btfss STATUS,Z ;¿Contador es igual a 2?
goto Loop ;No. Selecciona las opciones del menú
clrf Contador ;Sí. Limpia Contador
goto Loop ;Selecciona las opciones del menú

DOWN
btfsc PORTB,6 ;¿Está pulsada la tecla DOWN ?
goto enter ;No. Salta a "enter"
call Delay_20_ms ;Sí. Elimina rebotes

btfss PORTB,6 ;¿A dejado de pulsar la tecla DOWN ?
goto UP ;No. Salta a "UP"
call Delay_20_ms ;Sí. Elimina rebotes

```

```

-----
;TECLA DOWN

    decf  Contador,F
    movlw 0xff
    subwf Contador,W
    btfss STATUS,Z           ;¿Contador es 0 ?
    goto  Loop              ;No. Selecciona las opciones del menú
    movlw 0x01              ;Sí.
    movwf Contador
    goto  Loop              ;Selecciona las opciones del menú

Loop
    movf  Contador,w
    call  cur_tab           ;Selecciona: Automático (0XC1) o Tiempo
                           ;variable (0XCC)
    call  pos_cur          ;Ubica al cursor en la dirección 0XC1 o
                           ;0XCC del LCD

enter
    btfsc PORTB,5         ;¿Está pulsada la tecla ENTER ?
    goto  regresa         ;No. Vuelta de interrupción
    call  Delay_20_ms     ;Sí. Elimina rebotes
    btfss PORTB,5         ;A dejado de pulsar la tecla ENTER ?
    goto  enter           ;No. Salta a "enter"
    call  Delay_20_ms     ;Sí. Elimina rebotes

-----
;TECLA ENTER

    movf  Contador,w
    sublw 0
    btfsc STATUS,Z         ;¿Es Z = 0?
    goto  AUTO             ;No. Ejecuta la rutina automático

    movf  Contador,w      ;Sí
    sublw 1
    btfsc STATUS,Z         ;¿Es Z = 0?
    goto  Tvar             ;No. Ejecuta la rutina tiempo variable
    goto  UP               ;Sí. Vuelvo a revisar las teclas

-----
;TABLAS

cur_tab
    addwf PCL,1           ;Tabla que asigna direcciones:
                           ;Automático (0XC1) o Tiempo variable (0xCC)
    retlw 0XC1            ;El valor se carga en el acumulador
    retlw 0XCC

```

```

tab_num
    addwf PCL,1      ;Tabla que asigna caracteres: 3,4,5,6,7,8,9
    retlw '3'       ;El valor se carga en el acumulador
    retlw '4'
    retlw '5'
    retlw '6'
    retlw '7'
    retlw '8'
    retlw '9'

cur_tab1
    addwf PCL,1      ;Tabla que asigna direcciones:
                    ;0XC2, 0XC3.....0XCA
                    ;El valor se carga en el acumulador
    retlw 0XC2
    retlw 0XC3
    retlw 0XC4
    retlw 0XC5
    retlw 0XC6
    retlw 0XC7
    retlw 0XC8
    retlw 0XC9
    retlw 0XCA

tabla
    addwf PCL,F      ;Tabla que asigna la secuencia de encendido de los
reles
    retlw b'00000001' ;El valor se carga en el acumulador
    retlw b'00000010'
    retlw b'00000100'
    retlw b'00001000'
    retlw b'00010000'
    retlw b'00100000'
    retlw b'01000000'
    retlw b'10000000'

;-----
;AUTOMATICO

AUTO
    clrf temporal_1      ;Iniciando variables
    clrf temporal_2
    clrf Contador
    clrf PCLATH          ;Página 0
    call DISPLAY_CLEAR   ;Inicializar LCD
    call SET_THE_DDRAM_ADDRESS_80;Empieza a escribir desde la
                                ;dirección 0X80
    call DISPLAY_ONOFF_CONTROL_1 ; Display on, cursor off,
                                ;cursor no parpadea
    movlw mensaje_c1     ;Carga la dirección de "mensaje_c1" al
                                ;acumulador
    call escribe_mensaje ;Escribe mensaje
    call SET_THE_DDRAM_ADDRESS_C0;Empieza a escribir desde la
                                ;dirección 0XC0
    movlw mensaje_c2     ;Carga la dirección de "mensaje_c2" al
                                ;acumulador

```

```

    call  escribe_mensaje    ;Escribe mensaje
    movlw d'1'              ;Página 1
    movwf PCLATH
    call  tiempo            ;Espera un tiempo para que sea leído el
                           ;mensaje
    bsf   PORTE,2          ;Pita una vez indicando inicio de secuencia
                           ;"Automático"

    call  tmp_sonido
    bcf   PORTE,2

otro
    call  DISPLAY_ONOFF_CONTROL_2 ;Display on, cursor on, cursor
                           ;parpadea

    movf  Contador,w
    call  cur_tab1          ;Asigna la dirección: 0XC2, 0XC3.....ó
                           ;0XCA al cursor
    call  pos_cur          ;Ubicar al cursor en la dirección que indique
                           ;"cur_tab1"

    movf  Contador,w
    sublw d'8'
    btfss STATUS,Z        ;¿Es el ultimo bits del Puerto C?
    goto  PUERTO_C        ;No. Activa a uno de los relés que están
                           ;conectados en el Puerto C
    clrf  PORTC            ;Sí. Limpio Puerto C
    bsf   PORTE,0        ;Activa al relé nueve, que está conectado en
                           ;el Puerto E
    call  tiempo            ;Espera un tiempo
    bcf   PORTE,0        ;Inactiva al relé nueve, que está conectado en
                           ;el Puerto E

    movlw 0xCC
    call  pos_cur          ;Ubica al cursor en la dirección 0XCC
    bsf   PORTE,1        ;Activa al relé diez, que está conectado en el
                           ;Puerto E
    call  tiempo            ;Espera un tiempo
    bcf   PORTE,1        ;Inactiva al relé diez, que está conectado en
                           ;el Puerto E
    bsf   PORTE,2        ;Pita dos veces indicando fin de secuencia
                           ;"Automático"

    call  tmp_sonido
    bcf   PORTE,2
    call  tmp_sonido
    bsf   PORTE,2
    call  tmp_sonido
    bcf   PORTE,2
    goto  menú            ;Salta al menú principal

PUERTO_C
    movf  Contador,w
    call  tabla            ;Señala al relé que se activará
    movwf PORTC          ;Activa al rele señalado
    call  tiempo            ;Espera un tiempo
    incf  Contador,f      ;Incrementa contador, para seleccionar la
                           ;siguiente dirección en el que se ubicara el
                           ;cursor
    goto  otro            ;Retorna para activar el siguiente rele

```

```
regresa
    clrf    TMR0                ;Vuelta de interrupción. Limpia TMR0
    movlw  b'10100000'        ;TOIE=1 permiso de interrupción del TMR0
    movwf  INTCON
    retfie                       ;Retorno de interrupción
```

```
pos_cur
    bcf    PORTA,0            ;RS=1 (Modo comando)
    bcf    PORTA,1            ;R/w=0 (Modo escritura)
    movwf  PORTD
    call   enable              ;Pulso de duración mayor a 500nseg
    call   DEM_40us            ;Llama demora de 40useg
    bsf    PORTA,0            ;LCD en modo DATO RS=1
    return                       ;Retorno de rutina
```

```
tiempo
    movlw  d'255'
    movwf  N
    movlw  d'50'
    movwf  M
    call   DEMORA
    return
```

```
tmp_sonido
    movlw  d'110'
    movwf  N
    movlw  d'20'
    movwf  M
    call   DEMORA
    return
```

```
tmp
    movlw  d'50'
    movwf  N
    movlw  d'20'
    movwf  M
    call   DEMORA
    return
```

```
Delay_20_ms
    movlw  d'36'
    movwf  N
    movlw  d'15'
    movwf  M
    call   DEMORA
    return
```

```
-----
;TIEMPO VARIABLE
```

```
Tvar
    clrf   temporal_1          ;Limpia variables
    clrf   temporal_2
    clrf   Contador
    call   DISPLAY_CLEAR      ;Limpia LCD
```

```

call SET_THE_DDRAM_ADDRESS_80;Empieza a escribir
                                ;desde la dirección 0X80
call DISPLAY_ONOFF_CONTROL_1 ;Display on, cursor off,
                                ;cursor no parpadea
movlw mensaje_d1                ;Carga la dirección de
                                ;"mensaje_d1" al acumulador
call escribe_mensaje           ;Escribe mensaje
call SET_THE_DDRAM_ADDRESS_C0;Empieza a escribir desde la
                                ;dirección 0XC0
movlw mensaje_d2                ;Carga la dirección de
                                ;"mensaje_d2" al acumulador
call escribe_mensaje           ;Escribe mensaje

```

```

;-----
;ESCANEAR TECLAS

```

```

UPvar

```

```

btfs PORTB,7                ;¿Está pulsada la tecla UP ?
goto DOWNvar                ;No. Salta a "DOWNvar"
call Delay_20_ms            ;Sí. Elimina rebotes
btfs PORTB,7                ;¿A dejado de pulsar la tecla UP ?
goto UPvar                   ;No. Salta a "UPvar"
call Delay_20_ms            ;Sí. Elimina rebotes

```

```

;-----
;TECLA UP

```

```

incf Contador,f
movlw d'7'                  ;Carga el acumulador con 7 por que son siete
                                ;tiempos de conmutación
subwf Contador,w
btfs STATUS,Z                ;¿Contador es igual a 7 ?
goto Loopvar                 ;No. Selecciona tiempos de conmutación
clrf Contador                ;Sí. Limpia Contador
goto Loopvar                 ;Selecciona tiempos de conmutación

```

```

DOWNvar

```

```

btfs PORTB,6                ;¿Esta pulsada la tecla DOWN ?
goto entervar                ;No. Salta a "entervar"
call Delay_20_ms            ;Sí. Elimina rebotes
btfs PORTB,6                ;¿A dejado de pulsar la tecla DOWN ?
goto UPvar                   ;No. Salta a "UPvar"
call Delay_20_ms            ;Sí. Elimina rebotes

```

```

;-----
;TECLA DOWN

```

```

decf Contador,F
movlw 0xff
subwf Contador,W
btfs STATUS,Z                ;¿Contador es 0 ?
goto Loopvar                 ;No. Selecciona tiempos de conmutación
movlw 0x06
movwf Contador
goto Loopvar                 ;Selecciona tiempos de conmutación

```

Loopvar

```

movlw d'1'                ;Página 1
movwf PCLATH
movlw 0XC6
call  pos_cur              ;Ubica al cursor en la dirección 0XC6
movf  Contador,w
call  tab_num              ;Asignar los caracteres: 3,4,5,6,7,8 ó 9 al
                               ;acumulador
call  dato                 ;Visualiza en el LCD los caracteres:
                               ;3,4,,5,6,7,8 ó 9

```

entervar

```

btfs  PORTB,5              ;¿Esta pulsada la tecla ENTER ?
goto  UPvar                ;No. Salta a "UPvar"
call  Delay_20_ms          ;Sí. Elimina rebotes
btfs  PORTB,5              ;¿A dejado de pulsar la tecla ENTER ?
goto  entervar            ;No. Salta a "entervar"
call  Delay_20_ms          ;Sí. Elimina rebotes

```

```

call  DISPLAY_CLEAR        ;Limpia LCD
call  SET_THE_DDRAM_ADDRESS_80;Empieza a escribir desde la
                               ;dirección 0X80
call  DISPLAY_ONOFF_CONTROL_1 ;Display on, cursor off, cursor no
                               ;parpadea
movlw mensaje_e1          ;Carga la dirección de "mensaje_e1" al
                               ;acumulador
call  escribe_mensaje      ;Escribe mensaje
call  SET_THE_DDRAM_ADDRESS_C0;Empieza a escribir desde la
                               ;dirección 0XC0
clrf  PCLATH              ;Pagina 0
movlw mensaje_c2          ;Carga la dirección de "mensaje_C2" al
                               ;acumulador
call  escribe_mensaje      ;Escribe mensaje
call  DISPLAY_ONOFF_CONTROL_2;Display on, cursor on, cursor parpadea
bsf  PORTE,2              ;Pita una vez indicando inicio de secuencia
                               ;"Tiempo variable"

call  tmp_sonido
bcf  PORTE,2

```

```

movlw d'0'
subwf Contador,w
btfs  STATUS,Z            ;¿Es cero el contador?
goto  tiempo3seg         ;Sí. Tiempo de 3seg de conmutación

```

```

movlw d'1'                ;No
subwf Contador,w
btfs  STATUS,Z            ;¿Es uno el contador?
goto  tiempo4seg         ;Sí. Tiempo de 4seg de conmutación

```

```

movlw d'2'                ;No
subwf Contador,w
btfs  STATUS,Z            ;¿Es dos el contador?
goto  tiempo5seg         ;Sí. Tiempo de 5seg de conmutación

```

```

movlw d'3'                ;No

```

```

subwf Contador,w
btfsc STATUS,Z           ;¿Es tres el contador?
goto tiempo6seg         ;Sí. Tiempo de 6seg de conmutación

movlw d'4'               ;No
subwf Contador,w
btfsc STATUS,Z           ;¿Es cuatro el contador?
goto tiempo7seg         ;Sí. Tiempo de 7seg de conmutación

movlw d'5'               ;No
subwf Contador,w
btfsc STATUS,Z           ;¿Es cinco el contador?
goto tiempo8seg         ;Sí. Tiempo de 8seg de conmutación

movlw d'6'               ;No
subwf Contador,w
btfsc STATUS,Z           ;¿Es seis el contador?
goto tiempo9seg         ;Sí. Tiempo de 9seg de conmutación
goto UPvar               ;No. Escanea de nuevo las teclas
;-----
;TIEMPOS DE CONMUTACION
tiempo3seg
movlw d'2'               ;Página 2
movwf PCLATH
movlw 0X8D
call pos_cur             ;Ubica al cursor en la dirección 0X8D
clrf PCLATH              ;Página 0
movlw '3'
call dato                ;Visualiza en el LCD el caracter "3"
movlw d'255'            ;Cargo las variables "N, M" con valores
                          ;fijados para un tiempo de 3seg

movwf O
movlw d'46'
movwf P
movlw d'1'               ;Página 1
movwf PCLATH
clrf Contador
goto otrovar             ;Activación de cada uno de los relés en el
                          ;tiempo de conmutación especificado

tiempo4seg
movlw d'2'               ;Página 2
movwf PCLATH
movlw 0X8D
call pos_cur             ;Ubica al cursor en la dirección 0X8D
clrf PCLATH              ;Página 0
movlw '4'
call dato                ;Visualiza en el LCD el caracter "4"
movlw d'255'            ;Cargo las variables "N, M" con valores
                          ;fijados para un tiempo de 4seg

movwf O
movlw d'62'
movwf P
movlw d'1'               ;Página 1
movwf PCLATH

```

```

    clrf Contador
    goto otrovar ;Activación de cada uno de los relés en el
                 ;tiempo de conmutación especificado

tiempo5seg
    movlw d'2' ;Página 2
    movwf PCLATH
    movlw 0X8D
    call pos_cur ;Ubica al cursor en la dirección 0X8D
    clrf PCLATH ;Página 0
    movlw '5'
    call dato ;Visualiza en el LCD el caracter "5"
    movlw d'255' ;Cargo las variables "N, M" con valores
                 ;fijados para un tiempo de 5seg

    movwf O
    movlw d'77'
    movwf P
    movlw d'1' ;Página 1
    movwf PCLATH
    clrf Contador
    goto otrovar ;Activación de cada uno de los relés en el
                 ;tiempo de conmutación especificado

tiempo6seg
    movlw d'2' ;Página 2
    movwf PCLATH
    movlw 0X8D
    call pos_cur ;Ubica al cursor en la dirección 0X8D
    clrf PCLATH ;Página 0
    movlw '6'
    call dato ;Visualiza en el LCD el caracter "6"
    movlw d'255' ;Cargo las variables "N, M" con valores
                 ;fijados para un tiempo de 6seg

    movwf O
    movlw d'92'
    movwf P
    movlw d'1' ;Página 1
    movwf PCLATH
    clrf Contador
    goto otrovar ;Activación de cada uno de los relés en el
                 ;tiempo de conmutación especificado

tiempo7seg
    movlw d'2' ;Página 2
    movwf PCLATH
    movlw 0X8D
    call pos_cur ;Ubica al cursor en la dirección 0X8D
    clrf PCLATH ;Página 0
    movlw '7'
    call dato ;Visualiza en el LCD el caracter "7"
    movlw d'255' ;Cargo las variables "N, M" con valores
                 ;fijados para un tiempo de 7seg

    movwf O
    movlw d'107'
    movwf P

```

```

movlw d'1' ;Página 1
movwf PCLATH
clrf Contador
goto otrovar ;Activación de cada uno de los relés en el
;tiempo de conmutación especificado

tiempo8seg
movlw d'2' ;Página 2
movwf PCLATH
movlw 0X8D
call pos_cur ;Ubica al cursor en la dirección 0X8D
clrf PCLATH ;Página 0
movlw '8'
call dato ;Visualiza en el LCD el caracter "8"
movlw d'255' ;Cargo las variables "N, M" con valores
;fijados para un tiempo de 8seg

movwf O
movlw d'123'
movwf P
movlw d'1' ;Página 1
movwf PCLATH
clrf Contador
goto otrovar ;Activación de cada uno de los relés en el
;tiempo de conmutación especificado

tiempo9seg
movlw d'2' ;Página 2
movwf PCLATH
movlw 0X8D
call pos_cur ;Ubica al cursor en la dirección 0X8D
clrf PCLATH ;Página 0
movlw '9'
call dato ;Visualiza en el LCD el caracter "9"
movlw d'255' ;Cargo las variables "N, M" con valores
;fijados para un tiempo de 9seg

movwf O
movlw d'138'
movwf P
movlw b'1' ;Página 1
movwf PCLATH
clrf Contador
goto otrovar ;Activación de cada uno de los relés en el
;tiempo de conmutación especificado

;-----
otrovar
movf Contador,w
call cur_tab1 ;Asigna la dirección: 0XC2, 0XC3.....ó
;0XCA al cursor
call pos_cur ;Ubicar al cursor en la dirección que indique
;"cur_tab1"

movf Contador,w
sublw d'8'
btfss STATUS,Z ;¿Se activo el rele ocho del Puerto C?
goto Loop3 ;No. Activa a uno de los siguientes relés que
;están conectados al Puerto C

```

```

clrf Contador ;Sí. Ya se activaron todos los relés del Puerto
;C. Limpia Contador
clrf PORTC ;Limpio Puerto C
bsf PORTE,0 ;Activa al rele nueve que esta conectado al
;Puerto E
call DEMORAconm ;Espera un tiempo determinado
bcf PORTE,0 ;Inactiva al rele nueve que esta conectado al
;Puerto E
movlw 0xCC
call pos_cur ;Ubica al cursor en la dirección 0XCC
bsf PORTE,1 ;Activa al rele diez que esta conectado al
;Puerto E
call DEMORAconm ;Espera un tiempo determinado
bcf PORTE,1 ;Inactiva al rele diez que esta conectado al
;Puerto E
clrf PCLATH ;Página 0
bsf PORTE,2 ;Pita dos veces indicando el fin de secuencia
;"Tiempo variable"
call tmp_sonido
bcf PORTE,2
call tmp_sonido
bsf PORTE,2
call tmp_sonido
bcf PORTE,2
goto menú ;Salto a menú

Loop3
movlw b'1' ;Página 1
movwf PCLATH
movf Contador,w
call tabla ;Señala al rele que se activará
movwf PORTC ;Activa al rele señalado
call DEMORAconm ;Espera un tiempo determinado
incf Contador,f
goto otrovar ;Regresa para activar los relés que todavía no
;han sido activados

end

;*****
;Rutinas para el manejo del LCD
;*****
;
#define habilita_LCD bsf PORTA,2 ;Activa señal E
#define deshabilita_LCD bcf PORTA,2 ;Desactiva señal E
#define LEER bsf PORTA,1 ;Pone LCD en Modo RD
#define ESCRIBIR bcf PORTA,1 ;Pone LCD en Modo WR
#define modo COMANDO bcf PORTA,0 ;Desactiva RS (modo comando)
#define modo DATO bsf PORTA,0 ;Activa RS (modo dato)

```

```

CBLOCK    LCD_maestro_var    ;Inicio de las variables. Será la primera
CONT1    ;dirección libre disponible
CONT2
CONT3
N
M
numcar
endc

```

;Configuración del PIC para manejar al LCD.

```

cfg_pic_LCD
    bsf    STATUS,RP0    ;Banco 1
    bcf    STATUS,RP1
    clrf   TRISD         ;RD <0-7> salidas digitales
    movlw b'00011000'
    movwf TRISA         ;RA0-RA2 salidas, RA3-RA5 entradas
    movlw b'00000111'
    movwf ADCON1       ;Todo el puerto A como e/s digitales
    bcf    STATUS,RP0    ;Banco 0
    bcf    STATUS,RP1
    modo_COMANDO        ;RS=0
    deshabilita_LCD     ;E=0
    return

```

;READ BUSY FLAG & ADDRESS

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	1	BF	DIRECCION DE LA CGRAM O DDRAM						

;Cuando el modulo LCD esta ejecutando cualquiera de estas instrucciones, tarda un cierto tiempo de ejecución en el que no se debe mandar ninguna instrucción para ello dispone de un flag llamado busy (ocupado) que indica que se esta ejecutando una instrucción previa. Esta instrucción de lectura informa del estado de dicho flag además de proporcionar el valor del contador de direcciones de la CG RAM ;o de la DD RAM según la ultima que se haya empleado.

;LCD_BUSY: Lectura del Flag Busy y la dirección.

```

LCD_BUSY
    LEER    ;lectura
    bsf    STATUS,RP0    ;Banco 1
    movlw  H'FF'
    movwf  PORTD         ;Puerta D como entrada
    bcf    STATUS,RP0    ;Banco 0
    habilita_LCD        ;Activa al LCD
    nop

```

```

LCD_BUSY_1
    btfsc PORTD,7           ;Chequea bit de Busy
    goto    LCD_BUSY_1
    deshabilita_LCD        ;Inactiva LCD
    bsf    STATUS,RP0      ;Banco 1
    clrf   PORTD           ;Puerta D salida
    bcf    STATUS,RP0      ;Banco 0
    ESCRIBIR                ;Ubica LCD en modo WR
    return

;*****
;
; "enable": Pulso de Enable. Esta señal debe estar a "0" unos 40uS antes de
; volver a activarse a "1".

enable
    habilita_LCD           ;Activa el LCD
    nop
    nop
    nop
    nop
    deshabilita_LCD       ;Inactiva LCD
    call   DEM_40us
    return

;*****
;
; "LCD_DATO": Escritura de datos en DDRAM o CGRAM. Envía el dato ;presente al
; acumulador

dato
    modo_COMANDO          ; Inactiva RS (modo comando)
    movwf PORTD           ;Valor ASCII a sacar por PORTD
    call   LCD_BUSY       ;Espera a que se libere el LCD
    modo_DATO             ;Activa RS (modo dato)
    goto  enable

;*****
;
; "LCD_REG": Escritura de comandos en el LCD. Envía el comando presente en ;el
; acumulador

LCD_REG
    modo_COMANDO          ;Desactiva RS (modo comando)
    movwf PORTD           ;Código de comando
    call   LCD_BUSY       ;LCD libre?
    goto  enable          ;Si. Genera pulso de E

;*****
;
; inicia_LCD
; "LCD_INI": inicialización del LCD enviando el comando "Function Set" 3 veces
; consecutivas. El LCD queda borrado y el cursor en la primera posición

```

```

;INICIO DE LA SECUENCIA DE RESET DEL CRISTAL
    call DEM_15ms           ;demora de power UP
    call FUNCTION_SET_INST
    call DEM_5ms
    call FUNCTION_SET_INST
    call DEM_100us
    call FUNCTION_SET_INST
    call DISPLAY_CLEAR
    return                 ;Fin de inicialización de LCD

;S = 1   desplaza la visualización cada ves que se escribe un dato
;S = 0   modo normal

;I/D = 1  incremento del cursor
;I/D = 0  decremento del cursor

;S/C = 1  desplaza el display
;S/C = 0  mueve el cursor

;R/L = 1  desplazamiento a la derecha
;R/L = 0  desplazamiento a la izquierda

;BF = 1   modulo ocupado
;BF = 0   modulo disponible

;DL = 1   bus de datos de 8 bits
;DL = 0   bus de datos de 4 bits

;N = 1   LCD de dos líneas
;N = 0   LCD de una línea

;F = 1   caracter de 5x10 puntos
;F = 0   caracter de 5x7 puntos

;B = 1   parpadeo del cursor ON
;B = 0   parpadeo del cursor OFF

;C = 1   cursor ON
;C = 0   cursor OFF

;D = 1   display ON
;D = 0   display OFF

;X =     indeterminado

```

.*****

;FUNCTION SET INST

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	1	DL	N	F	X	X

;Establece el tamaño de interfase con el bus de datos (DL), numero de líneas del display (N) y tipo de caracter (F)

```

FUNCTION_SET_INST
    ;Modo de transferencia de datos: 8bits
    ;display en 2 líneas
    ;matriz de 5x7 puntos
    movlw 0x38
    call LCD_REG
    return
    
```

.*****
 ;
 ;ENTRY MODE SET INST

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	1	I/D	S

;Establece la dirección de movimiento del cursor y especifica si la visualización ;se va desplazando a la siguiente posición de la pantalla o no; estas operaciones se ;ejecutan durante la lectura o escritura de la DD RAM o CG RAM. Para visualizar ;normalmente ponga el bit S=0

```

ENTRY_MODE_SET_INST
    ;la posición del cursor se incrementa
    ;no desplazar el dato
    movlw 6
    call LCD_REG
    return
    
```

.*****
 ;
 ;DISPLAY ON/OFF CONTROL

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	1	D	C	B

;Activa o desactiva poniendo en on/off tanto al display (D) como al cursor (C) y se ;establece si este ultimo debe o no parpadear (B)

```

DISPLAY_ONOFF_CONTROL_1
    ;display on
    ;cursor off
    ;cursor no parpadea
    movlw 0x0C
    call LCD_REG
    return
    
```

```

DISPLAY_ONOFF_CONTROL_3
;display on
;cursor on
;cursor no parpadea
movlw 0x0E
call LCD_REG
return
    
```

```

DISPLAY_ONOFF_CONTROL_2
;display on
;cursor on
;cursor parpadea
movlw 0x0F
call LCD_REG
return
    
```

```

DISPLAY_ONOFF_CONTROL_0
;display off
;cursor off
;cursor no parpadea
movlw 0x08
call LCD_REG
return
    
```

```

;*****
;
;DISPLAY CLEAR
    
```

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	0	1

;Borra el modulo LCD y coloca el cursor en la primera posición (dirección 0) ;Coloca el bit I/D a 1 por defecto

```

DISPLAY_CLEAR
;limpia RAM del Display y ubica al cursor
;en la posición cero
movlw 1
call LCD_REG
call DEM_1640us
return ;Fin de inicialización de LCD
    
```

```

;*****
;
;HOME
    
```

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	1	X

;Coloca el cursor en la posición de inicio (dirección 0) y hace que el display ;comience a desplazarse desde la posición original. El contenido de la Memoria

;RAM de datos de visualización (DD RAM) permanece invariable; la dirección de ;la Memoria RAM de datos para la visualización (DD RAM) es puesta a cero

HOME

```
movlw 0x02
call LCD_REG
call DEM_1640us
return
```

```
.*****
;
;SET THE DD RAM ADDRESS
```

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	1	DIRECCION DE LA DDRAM						

;Los caracteres o datos que se van visualizando, se van almacenando en una ;memoria llamada DD RAM para de aquí pasar a la pantalla, mediante esta ;instrucción se establece la dirección de la memoria DD RAM a partir de la cual ;se irán almacenando los datos a visualizar. Ejecutando este comando, todos los ;datos que se escriban o lean posteriormente lo harán desde esta memoria ;DDRAM. Las direcciones de la 0X80 a la 0X8F corresponden con los 16 caracteres ;del primer región y de la 0XC0 a la 0XCF con los 16 caracteres del segundo región

SET_THE_DDRAM_ADDRESS_80

```
movlw 0x80
call LCD_REG
return
```

SET_THE_DDRAM_ADDRESS_C0

```
movlw 0xC0
call LCD_REG
return
```

```
.*****
;
;CURSOR OR DISPLA SHIFT
```

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	1	S/C	R/L	X	X

;Mueve el cursor y desplaza el display sin cambiar el contenido de la memoria de ;datos de visualización DDRAM

CURSOR_OR_DISPLA_SHIFT

```
movlw b'0000011100'
call LCD_REG
return
```

```

;*****
;DESPLAZA EL MENSAJE CADA 500ms HACIA LA DERECHA

```

desplaza_display

```

    call DISPLAY_ONOFF_CONTROL_0 ;Parpadeo del cursor off
                                     ;cursor off, display off

    call CURSOR_OR_DISPLA_SHIFT ;Mueve el cursor y desplaza el
                                     ;display sin cambiar el ;contenido
                                     de la Memoria de ;Datos de
                                     visualización ;DDRAM

    call DISPLAY_ONOFF_CONTROL_1 ;Parpadeo del cursor off
                                     ;cursor off ,display on

    call proba
    incf numcar
    movfw numcar
    sublw d'16'
    btfss STATUS,Z
    goto desplaza_display
    clrf numcar
    return

```

proba

```

    movlw d'60'
    movwf N
    movlw d'20'
    movwf M
    call DEMORA
    return

```

```

;*****
;SUBROUTINAS DE DEMORAS

```

DEM_1640us

```

    movlw d'47'
    movwf N
    movlw d'1'
    movwf M
    call DEMORA
    return

```

DEM_100us

```

    movlw d'12'
    movwf N
    movlw d'1'
    movwf M
    call DEMORA
    return

```

DEM_5ms

```

movlw d'36'
movwf N
movlw d'6'
movwf M
call DEMORA
return

```

```

DEM_15ms
movlw d'36'
movwf N
movlw d'19'
movwf M
call DEMORA
return

```

```

DEM_40us
movlw d'7'
movwf N
movlw d'1'
movwf M
call DEMORA
return

```

```

DEM_500ms
movlw d'128'
movwf N
movlw d'40'
movwf M
call DEMORA
return

```

```

DEM_1us
nop
nop
nop
nop
return

```

```

;*****
;
;SUBROUTINA DE DEMORA PRINCIPAL

```

```

DEMORA
movf N,w
movwf CONT1
movwf CONT2
movf M,w
movwf CONT3

```

```

LAZO
decfsz CONT1,1
goto LAZO
movf N,w
movwf CONT1
decfsz CONT2,1

```

```

goto LAZO
movf N,w
movwf CONT2
decfsz CONT3,1
goto LAZO
return
    
```

3.4 PCB del Circuito de Control de Conmutación Automatizado.

El PCB (circuito impreso), está subdividido en los siguientes circuitos (figura 3.10):

- PIC 16F877, utilizado para controlar los periféricos de entrada y salida, principalmente el Circuito de Conmutación (relés).
- LCD, circuito para la visualización de información como menús y activación de relés.
- Teclado Lineal, circuito para el manejo de las teclas *enter*, *up* y *down*.
- *Reset*, circuito de reinicialización del PIC.
- Reloj, generador de frecuencia a 20MHz.
- Fuente de Alimentación, este sirve para energizar el circuito a 5 vcc.
- Relés de Conmutación, circuito encargado de la conmutación de los relés.
- Señal auditiva, para indicar el inicio y fin de una prueba de las características eléctricas.

PIC 16F877.

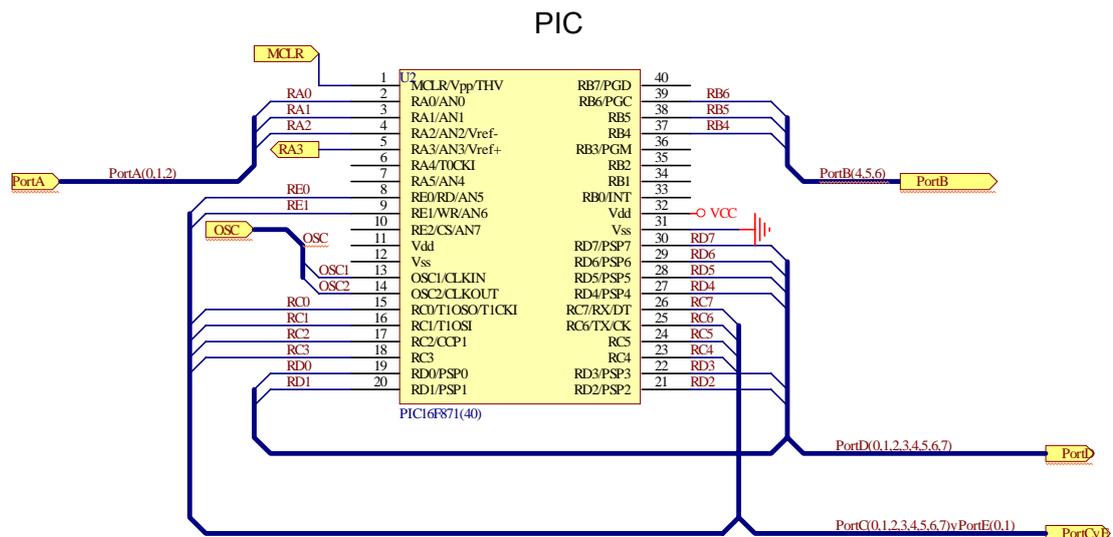


Figura 3.10 Circuito de Control de Conmutación Automatizado.

LCD.

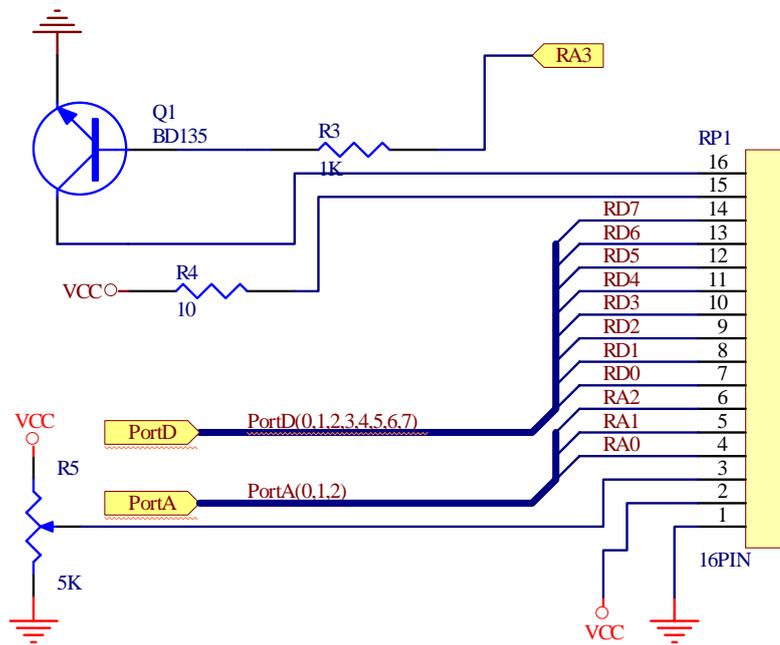


Figura 3.11 Circuito del LCD.

Teclado Lineal.

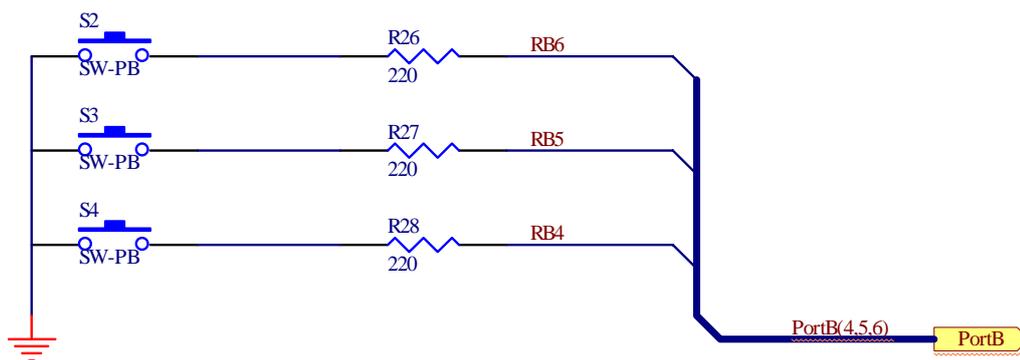


Figura 3.12 Circuito del teclado lineal.

Reset.

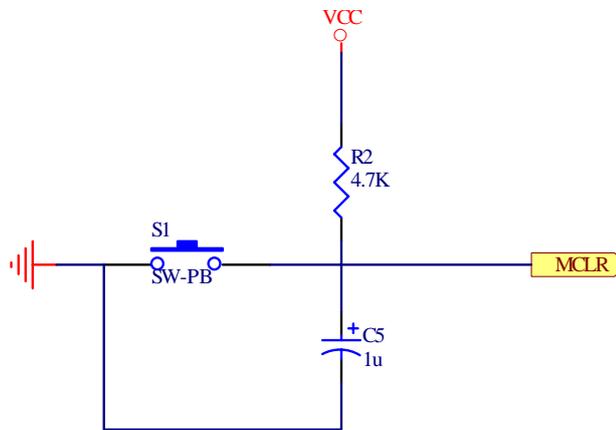


Figura 3.13 Circuito del reset.

Reloj.

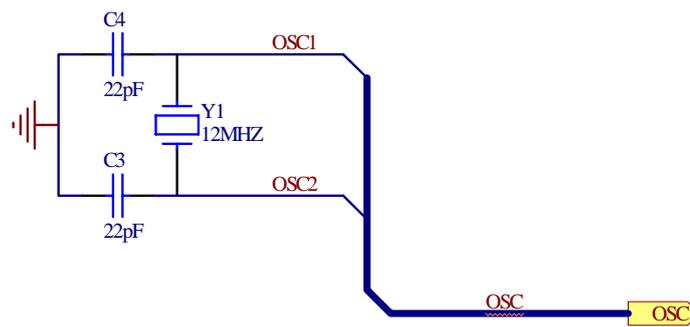


Figura 3.14 Circuito del reloj.

Fuente de Alimentación.

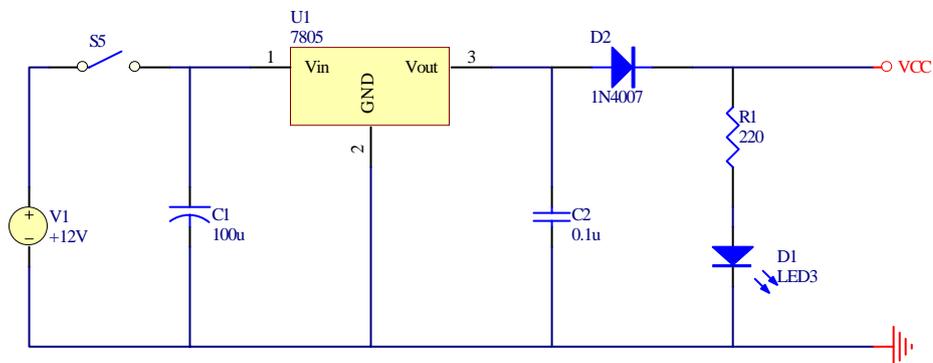


Figura 3.15 Circuito de la fuente de alimentación.

Relés de Conmutación.

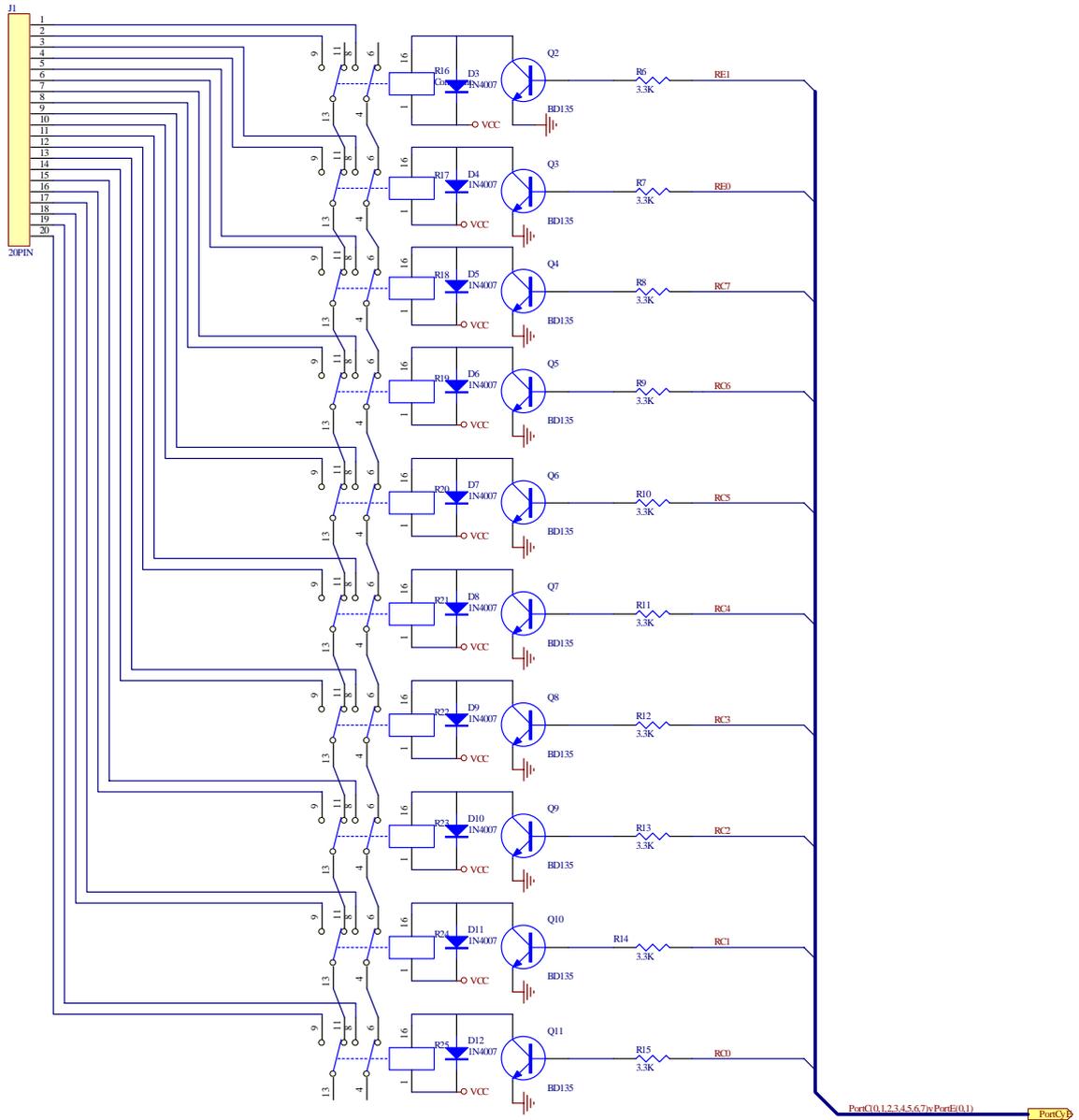


Figura 3.16 Circuito de los relés de conmutación.

Leds de señalización de los relés.

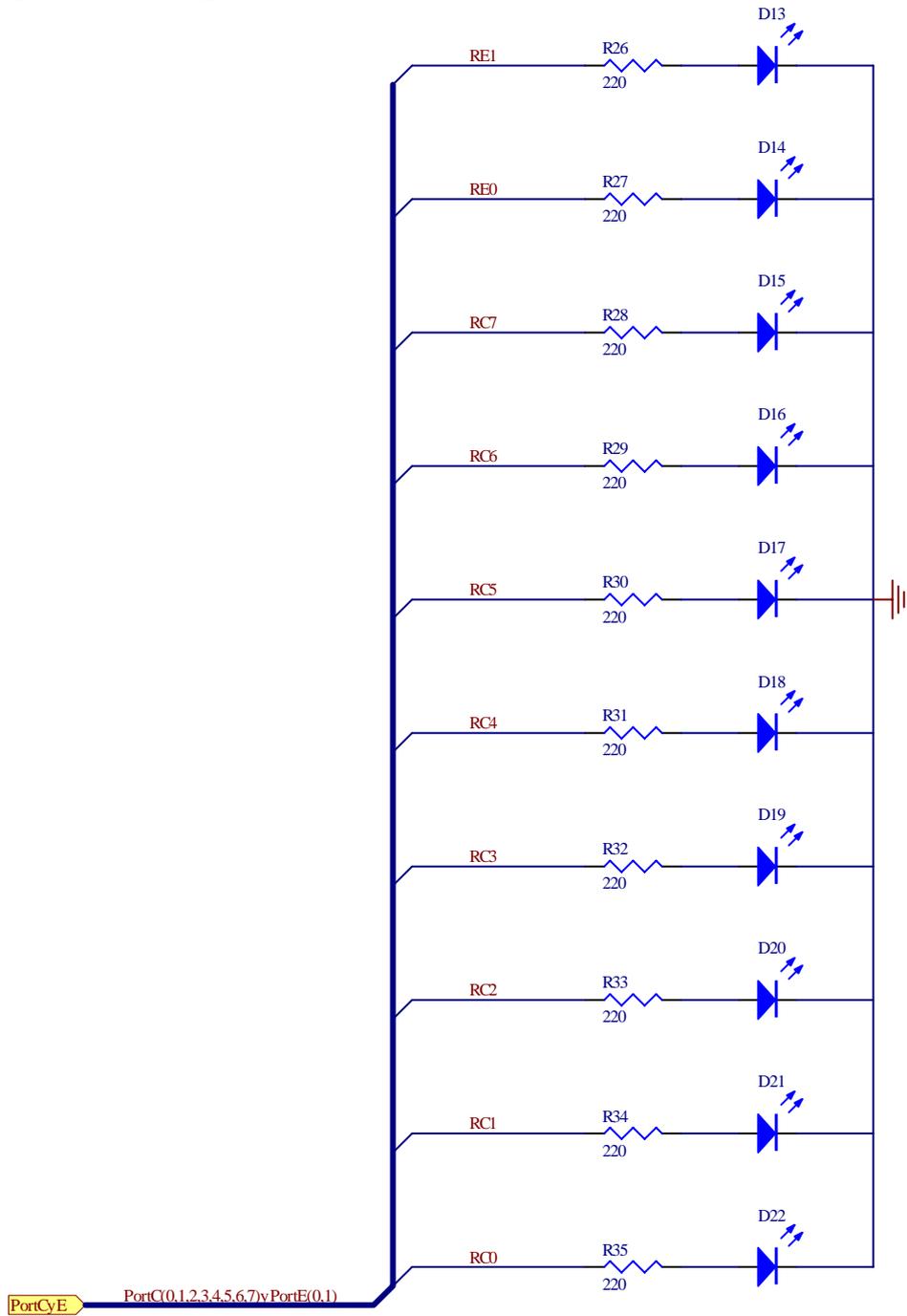


Figura 3.17 Circuito de los Leds de señalización para los relés de conmutación.

Señal Auditiva – Buzzer.

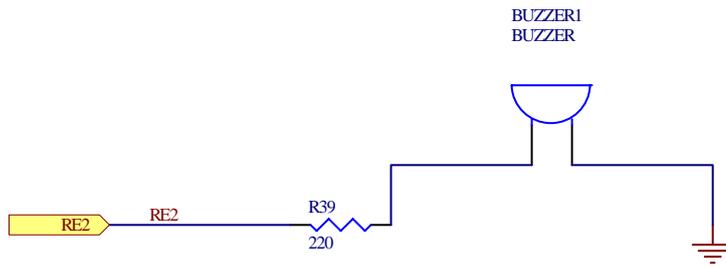
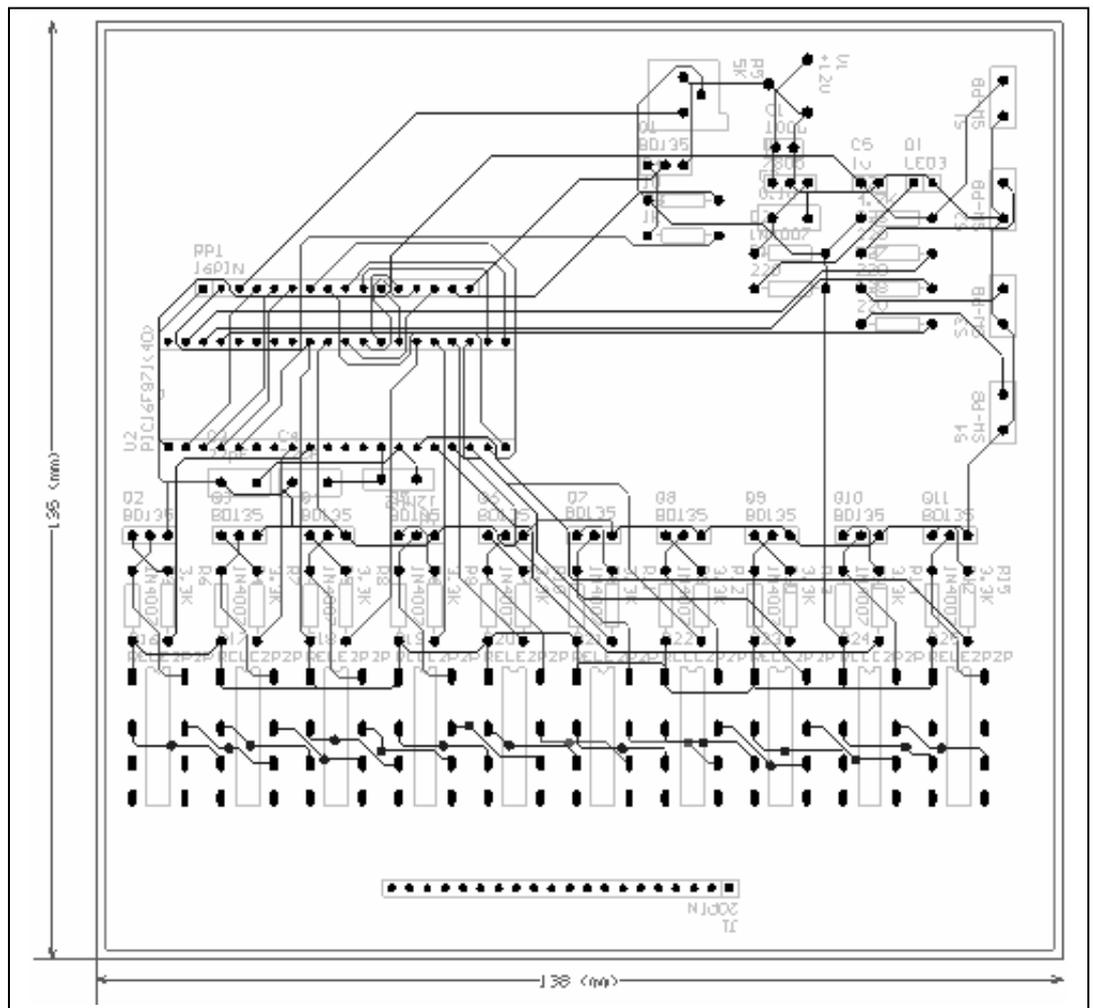


Figura 3.18 Circuito del Buzzer.

PCB del Circuito de Control de Conmutación Automatizado.



Materiales de construcción.

F UENTE

CANTIDAD	ELEMENTOS
1	7805
1	Led
1	Diodo 1N4007
1	Resistencia 220 Ω
1	Capacitor electrolítico 0.1uF
1	Capacitor electrolítico 100uF

RELES

CANTIDAD	ELEMENTOS
10	Relés de 5V
10	Diodos 1N4007
10	Transistores BD135
10	Resistencias 3.3K Ω
10	Resistencias 220 Ω
10	Leds

LCD

CANTIDAD	ELEMENTOS
1	Transistore BD135
1	Resistencia 1K Ω
1	Resistencia 10 Ω
1	Trimer 5K Ω
1	LCD

RELOJ

CANTIDAD	ELEMENTOS
2	Capacitores 22pF
1	Cristal 12MHz

RESET

CANTIDAD	ELEMENTOS
1	Pulsante
1	Resistencia 4.7K Ω
1	Capacitor 1uF

TECLADO

CANTIDAD	ELEMENTOS
3	Pulsantes
3	Resistencia 220 Ω

MICROCONTROLADOR - PIC

CANTIDAD	ELEMENTOS
1	Pic 16F877

SEÑAL AUDITIVA

CANTIDAD	ELEMENTOS
1	Buzzer (5V)
1	Resistencia 3.3K Ω
1	Transistor 2N3904

Conclusiones y Recomendaciones Generales.

Conclusiones:

- La técnica utilizada por la empresa ETAPA para las mediciones de las Pruebas de las Características Eléctricas de los cables telefónicos con la ayuda del téster Fluke 189, son llevadas de forma manual, generando una pérdida de tiempo y desgaste físico del personal encargado; por lo que se puede concluir que existe la necesidad de cambiar esta técnica manual por una técnica automática que resuelva este problema.
- Durante el estudio del equipo de medición Fluke 189 y su software, se observó que su base de datos receptora de las mediciones, puede ser exportada a ciertos programas compatibles con ella; concluyendo que esta base de datos exportada por ejemplo a una hoja electrónica como Excel, puede ser utilizada y generar reportes dirigidos y oportunos para la empresa.
- Para la construcción del Circuito de Control de Conmutación, se utilizó como elemento principal el microcontrolador PIC 16F877, mismo que permite manejar periféricos de entrada y salida (LCD, teclado, etc), así como controlar elementos pertenecientes a la electrónica analógica y digital; optimizando el poder de control del PIC 16F877, la versatilidad y lo económico que resulta su utilización.
- Mediante el Análisis, Diseño y Construcción del Circuito de Control de Conmutación Automatizado, se ha visto la necesidad del estudio y creación futura de dispositivos adicionales (regletas y cables especiales) para este circuito de conmutación, que puedan controlar otras pruebas como desequilibrio resistivo y desequilibrio capacitivo (Capítulo 1), concluyendo así la versatilidad de este Circuito de Conmutación.

Recomendaciones:

- Se recomienda la utilización del software MPLAB para la programación del microcontrolador PIC 16F877, debido a las herramientas que posee como el compilador, simulador, etc (capítulo 3), que permiten verificar errores de programación y el funcionamiento de uno o varios procesos de control.
- Se recomienda el uso del microcontrolador PIC 16F877, por lo versátil y económico que resulta su utilización en este tipo de proyectos.
- Se recomienda la implementación del prototipo del Circuito de Control de Conmutación Automatizado, para aprovechar las ventajas del equipo de medición Fluke 189 y su software, superando a la vez el problema de la pérdida de tiempo y el desgaste físico del personal encargado de tomar las pruebas de las Características Eléctricas de los cables de las redes de telecomunicaciones.

Glosario.

A/D: Analógico/Digital.

ADC: *Analog Digital Conversor.*

CCP: *Capture Compare PWM.*

DDRAM: *Display Data RAM.*

E/S: Entradas/Salidas.

EEPROM: *Electrical Erasable Programmable Read Only Memory.*

GIE: *Global Interrup Enable bit.*

ICSP: *In-circuit Serial Programming.*

INTCON: *Interruptions Control.*

LVP: *Low Voltage Permission.*

OPTION_REG: *Option Register.*

OST: *Osillator Start-up Timer.*

PCB: *Printer Circuit Board.*

PCH: *Program Counter's High significant byte.*

PCL: *Program Counter's Least significant byte.*

PCLATCH: *Program Counter Latch.*

PIC: Programado Integrado Controlador.

POR: *Power-on Reset.*

PSP: *Parallel Slave Port.*

PWM: *Pulse Width Modulation.*

RAM: *Random Access Memory.*

RISC: *Reduced Instruction Set Computers.*

ROM: *Read Only Memory.*

SSP : *Serial Synchronous Port.*

T0CS: *TMRO Clock Source Select bit.*

TMR0: *Timer0 Module's Register.*

USART: *Universal Synchronous Asynchronous Receiver Transmitter.*

WDT: *Watchdog Timer.*

Bibliografía.

ANGULO. José Ma. ANGULO. Ignacio. ROMERO. Susana.
"MICROCONTROLADORES <<PIC>> Diseño práctico de aplicaciones, Segunda
Parte: PIC 16F87X". España. Mc Graw Hill. [s.a.]. 231. 2

CANDIA. Miguel. "PLANTA EXTERNA Cables simétricos y coaxiales para
telecomunicaciones". CHILE. 2000. 270. 1

Manejo del programa IC-PROG. www.TodoTarjetas.com. [consulta agosto de
2005].

Conceptos básicos del PIC. <http://www.micropic.arrakis.es/pag3.htm>. [consulta
agosto de 2005].

Estructura interna del PIC. <http://tapec.uv.es/edupic/doc/pic16f87x.pdf>. [consulta
agosto de 2005].

Estudio de la serie de los Pícs 16F87X
<http://www.learobotics.com/personal/juan/publicaciones/art5/pres-pic-linux.pdf>.
[consulta agosto de 2005].

Conocimiento de Interrupciones del PIC.
<http://www.eie.fceia.unr.edu.ar/ftp/Digital%2520III/2005/Modulo%25204/Interrupcion%25202005.pdf+interrupciones+no+enmascaradas&hl=es&ie=UTF-8&client=pub-5809907729379004> . [consulta septiembre de 2005].

Manuales y Programa del FLUKE 189. <http://www.fluke.com> [Consulta septiembre
de 2005].

Estudio de Planta Externa.
<http://www.plantaexterna.cl/Planta%20Externa/redes.htm>.... [consulta septiembre de
2005].

Conocimiento de Planta Externa.
<http://elsitiodeltelecomunicaciones.iespana.es/plantaexterna.htm>. [consulta
septiembre de 2005].

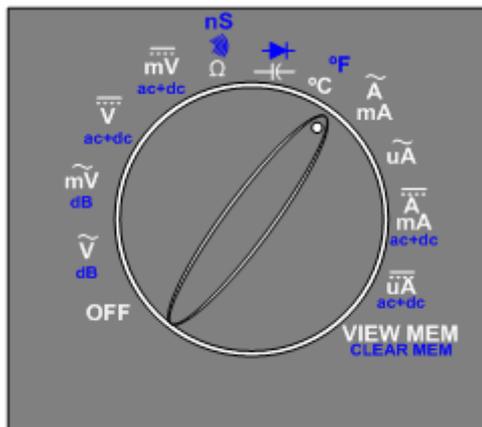
ANEXOS.

ANEXO 1.

Hardware Fluke 189.

Siendo el téster Fluke 189 el equipo utilizado para la toma de mediciones de las características eléctricas, se describe a continuación en forma general su funcionamiento.

Selector Giratorio.



Sirve para escoger cualquiera de las funciones de medición principales o alternas, que se encuentran alrededor del selector giratorio. Las funciones de medición principales están identificadas con letras de color blanco y las funciones alternas identificadas con letras de color azul.

De acuerdo a la posición del selector giratorio, aparecerá en la pantalla la función correspondiente.

Botones Pulsadores.

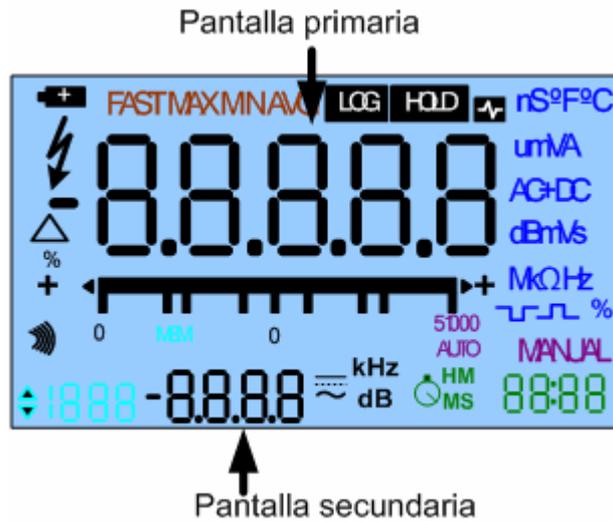


Los botones pulsadores se utilizan para incrementar la capacidad de la función seleccionada con el selector giratorio.

El botón azul () sirve para utilizar las funciones marcadas con letras de color azul, que se encuentran alrededor del selector giratorio.

El botón amarillo () sirve para utilizar funciones adicionales. Estas funciones aparecen con letras de color amarillo sobre las teclas correspondientes.

Explicación de la Pantalla.



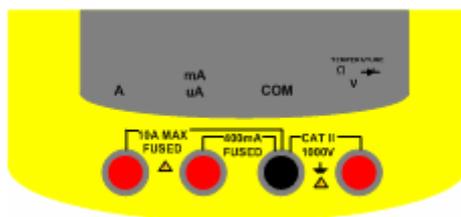
Pantalla Primaria.

La pantalla primaria indica la lectura actual para la función correspondiente en el selector giratorio.

Pantalla Secundaria.

La pantalla secundaria a menudo indica la lectura actual cuando la pantalla primaria esta ocupada mostrando algún resultado.

Utilización de los terminales de entrada.



Todas las funciones excepto la corriente utilizan las entradas  y .

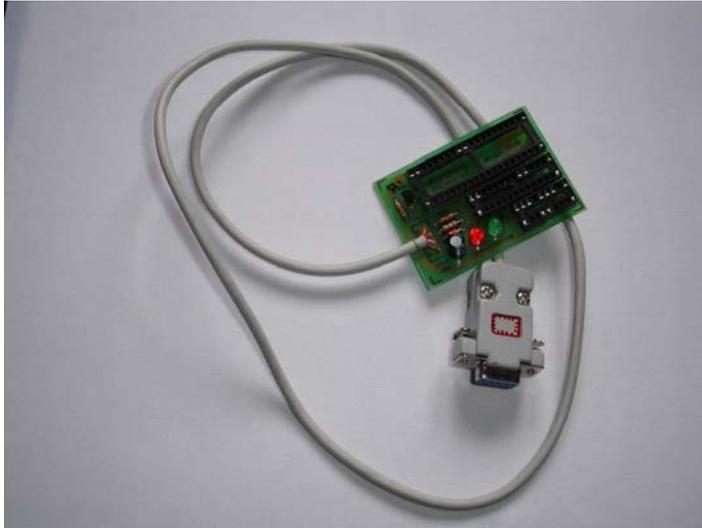
Las funciones de la corriente utilizan las entradas que se identifican a continuación:

Las funciones  o : utilizan las entradas  y  para corrientes de 400mA a 20A y las entradas  y  para valores de entrada ≤ 400mA.

Las funciones  o : utilizan  y  para entradas ≤ 5000uA.

ANEXO 2.

Grabador IC-PROG.



Materiales de construcción:

- 1 DB9 hembra.
- 3 Resistencias de 10K.
- 1 Transistor BC547.
- 4 Diodos 1N4148
- 1 Resistencia de 2k2.
- 2 Capacitores de 100nF.
- 1 Diodo zener de 12V
- 2 Diodos zener de 5V1.
- 1 Capacitor cerámico de 1000uF de 25V.
- 2 Zócalos de 40 pines.

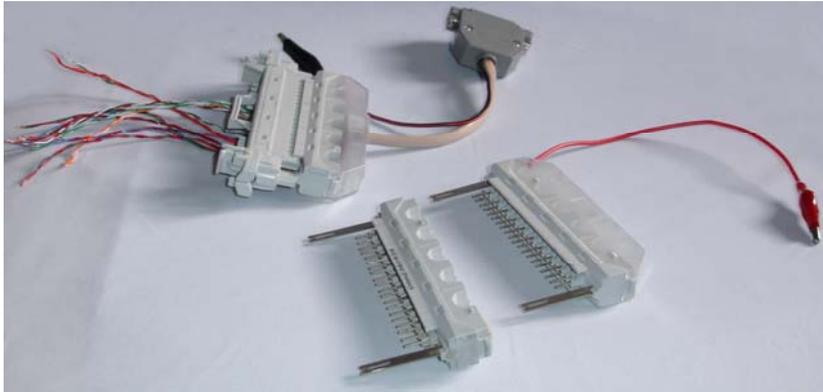
ANEXO 3.

CABLES Y REGLETAS.

Cables.

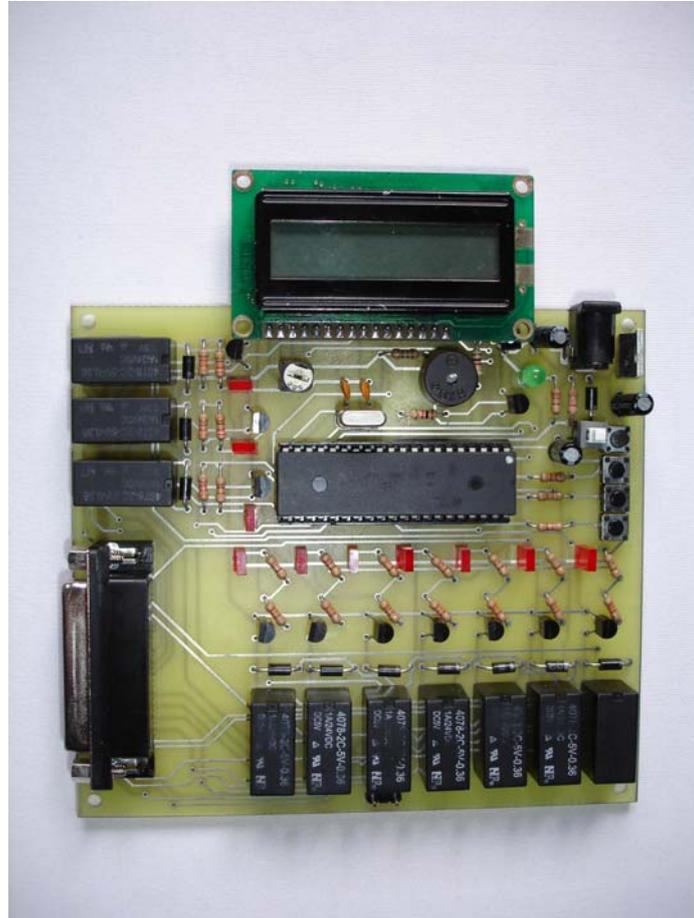


Regletas.



ANEXO 4.

Tarjeta del Circuito de Control de Conmutación Automatizado.



ANEXO 5.

Versiones del software de apoyo.

MPLAB	V7.10
IC-PROG	1.05A
FLUKE	189
PROTEL	DXP 2004