

ESCUELA SUPERIOR POLITECNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

TESIS de GRADO

“Comunicación Inalámbrica”

Previa a la obtención de los Títulos de:

INGENIERO EN COMPUTACIÓN ESPECIALIZACIÓN

SISTEMAS TECNOLÓGICOS

INGENIERO EN ELECTRÓNICA Y

TELECOMUNICACIONES

Rene Iván Ramírez Molina

David Fernando Murillo Miranda

William Vladimir Rosillo Toro

Guayaquil – Ecuador

Año: 2007

AGRADECIMIENTO

A todas las personas que ayudaron a llevar a cabo la realización de este trabajo y muy especialmente al Ing. Edgar Izquierdo y al Ing. Francisco Novillo Directores de nuestro Tópico, por su invaluable aporte.

DEDICATORIA

A MIS PADRES

A MIS HERMANOS

A MI ESPOSA

A MIS HIJOS

TRIBUNAL DE GRADUACIÓN

Ing. Holger Cevallos
SUBDECANO DE LA FIEC
PRESIDENTE

Ing. German Vargas
DIRECTOR DE TESIS

Ing. Rebeca Estrada P.
VOCAL

Ing. Xavier Ochoa Ch.
VOCAL

DECLARACION EXPRESA

“La responsabilidad del contenido de esta Tesis de Grado, nos corresponden exclusivamente; y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL”

(Reglamento de Graduación de la ESPOL).

Rene Ivan Ramírez
Molina

David Fernando Murillo
Miranda

William Vladimir Rosillo
Toro

RESUMEN

El presente trabajo está basado en la implementación de un sistema de comunicación inalámbrica, utilizando un equipo celular para enviar la información y un computador para recibir la misma; el medio de transporte es el aire, la red de telefonía celular y la red de internet, para recibir los mensajes del teléfono debemos conectarnos a una dirección de correo electrónico en internet y de esta manera poder leerlos utilizando un programa diseñado para este fin.

Nuestro prototipo está orientado a brindar un servicio de mensajería a las personas que no posean un equipo celular y que desean comunicarse con aquellas que si lo posean, sin importar con que operadora de telefonía celular trabajen.

En el primer capítulo revisaremos como fue concebida la idea de la tecnología propuesta, la justificación de la misma y las barreras de entrada para su comercialización. En el segundo capítulo se detallan las partes que componen nuestro circuito, la descripción de como se integran en nuestro protoboard y la implementación del prototipo. El tercer capítulo corresponde al plan de negocios que ayudara a llevar a nuestro proyecto a la etapa de comercialización en el mercado ecuatoriano. Por último revisaremos las conclusiones y recomendaciones.

INDICE GENERAL

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA INALÁMBRICO PARA ENVÍO DE INFORMACIÓN”

Resumen	VI
Indice General	VII
Indice de Figuras.....	IX
Indice de tablas.....	X
Introducción	1
1. Capítulo I:	2
<u><i>Generación y análisis de la idea tecnológica</i></u>	
1.1. <u><i>Descripción y Justificación de la idea tecnológica</i></u>	3
1.2. <u><i>Objetivos del Proyecto</i></u>	4
1.3. <u><i>Evaluación preliminar de la idea tecnológica</i></u>	5
1.4. <u><i>Determinación del Modelo de Negocio</i></u>	6
1.5. <u><i>Análisis de diferentes alternativas de negocio</i></u>	7
1.6. <u><i>Selección del modelo de negocio</i></u>	9
1.7. <u><i>Planeación de la aventura de negocio</i></u>	9
2. Capítulo II:	10

Desarrollo del Producto

2.1. <u>Diagrama de bloques del sistema</u>	11
2.2. <u>Diseño e implementación detallada</u>	27
2.2.1 <u>Desarrollo de interfase para controlar teclado del celular</u>	28
2.2.2 <u>Comunicación serial con la computadora utilizando microcontroladores</u>	38
2.2.3 Interfase para comunicación entre el teclado del celular con la computadora.....	39
2.2.4 Programa para leer mensajes enviados por el celular.....	41
2.2.5 Programa para monitoreo de mensajes.	42

3. Capítulo III:43

Plan de Mercadeo

3.1. <u>Resumen Ejecutivo</u>	44
3.2. <u>Análisis del Mercado y Plan de Mercadeo</u>	45
3.3. <u>Análisis Técnico</u>	46
3.4. <u>Análisis Administrativo</u>	48
3.5. <u>Análisis Legal y Social</u>	52
3.6. <u>Análisis de Valores Personales</u>	53
3.7. <u>Análisis Económico y Financiero</u>	54
3.8. <u>Análisis de Riesgos</u>	56
3.9. <u>Evaluación Integral del Proyecto</u>	57

Conclusiones y Recomendaciones.....60

Anexos

A. Plan de Negocios General.....	62
----------------------------------	----

B. Hojas de referencias técnicas de los componentes y equipos utilizados para la infraestructura requerida.....	76
C. Resultado de las entrevistas y encuestas.....	88
D. Rutina para almacenar datos.....	103
E. Programa.....	108
 Bibliografía.-	 204

ÍNDICE DE FIGURAS

Figura 2.1.1 Diagrama de bloques del Sistema.....	11
Figura 2.1.2 Aplicaciones que se pueden implementar.....	12
Figura 2.1.3 Diagrama de flujo del proceso de comunicación.....	13
Figura 2.1.4 Protoboard con dispositivos conectados al PIC(centro).....	15
Figura 2.1.5 Placa para pruebas de configuración y funcionamiento.....	17
Figura 2.1.6 Teléfono Nokia modelo 5125 desarmado e introducido en la placa.....	18
Figura 2.1.7 Placa conectada al proto utilizando un bus de datos.....	20
Figura 2.1.8 Conexión entre la computadora y el proto a través de un cable DB9.....	21
Figura 2.1.9 Conexiones entre la memoria y el Pic en el Proto	23
Figura 2.1.10 Botoneras implementadas en nuestro prototipo para simulación de eventos	26

Figura 2.2.1 Diseño e implementación mas detallada del proyecto.....	27
Figura 2.2.1.1 Conexiones internas del dispositivo switch 4016	32
Figura 2.2.1.2 Conexiones entre los switches 4016 y el teléfono	34
Figura 2.2.1.3 Conexiones entre los switches 4016, el teléfono celular y el PIC	35
Figura 2.2.1.4 Esquemático del PIC 16F877 y todos los demás componentes	36
Figura 2.2.1.5 Conexiones de todos los dispositivos y el PIC en el protoboard.....	37
Figura 2.2.2.1 Esquemático de la comunicación serial.....	39

ÍNDICE DE TABLAS

Tabla 2.1.1 Descripción de las teclas definidas en el PIC.....	19
--	----

INTRODUCCION

El trabajo expuesto a continuación nace del Tópico de Graduación “Desarrollo de Productos con Base Tecnológica”, que fue desarrollado para obtener un producto que pueda ser comercializado en nuestro mercado a partir de equipos, software ya diseñados y de los cuales podamos obtener un sistema integrado novedoso sin necesidad de volverlo a fabricar sino mas bien utilizarlo en beneficio propio para desarrollar nueva tecnología.

El objetivo principal del proyecto es diseñar e implementar un dispositivo inalámbrico para enviar y recibir información, haciendo uso de microcontroladores para hacer interfase con el computador y la telefonía celular como medio de comunicación.

El presente trabajo busca abrir nuevas puertas a la tecnología ya desarrollada para mejorarla y fusionarla con nuevas necesidades en nuestro mercado.

CAPITULO I

GENERACIÓN Y

ANÁLISIS

DE LA

IDEA TECNOLÓGICA

1.1.- Descripción y Justificación de la idea tecnológica.-

La etapa de implementación de la idea tecnológica fue desarrollada luego de un análisis realizado tanto por ingenieros y estudiantes para determinar posibles alternativas tecnológicas, basándonos en dispositivos ya diseñados, tratando de obtener un producto mixto al combinar diferentes tecnologías cuyo resultado pueda ser comercializado.

Luego de diferentes etapas se logró implementar un sistema para enviar información en forma inalámbrica. Dicha comunicación, se establecerá usando un teléfono celular, PIC's y una computadora. A través de una interfase se controlará el teclado del celular para enviar información en mensajes de texto, que a su vez será remitido hacia una dirección de correo electrónico.

Para revisar el mensaje se desarrolló un programa que mostrará la información para que el usuario que lo reciba ejecute una acción determinada.

Esta implementación tecnológica utilizará microcontroladores de mediana escala de integración PICS para controlar el teléfono y ejecutar el proceso automático, luego utilizamos la tecnología SMS, para enviar información a través de estos dispositivos móviles sin necesidad de que los operadores del teléfono intervengan en la conexión ó en el envío de dicha información.

El microcontrolador será el encargado de operar el celular, realizar el marcado y enviar la información. Esta información podrá ser leída y procesada por el destinatario al recibir un correo electrónico.

1.2 Objetivo del Proyecto.-

El objetivo general de nuestro proyecto es diseñar e implementar un dispositivo que enviará y recibirá información de manera inalámbrica. Para lograrlo nos valdremos de la telefonía celular como medio de comunicación y de microcontroladores que servirán para crear una interfase entre el teléfono móvil y el computador.

Detalles técnicos de los objetivos.-

- Desarrollar una interfase entre el computador y el teclado del celular para automatizar el envío de mensajes de texto de manera inalámbrica a un computador remoto o a un dispositivo móvil.

- Desarrollar una interfase para comunicar el computador con el teléfono celular, a través del puerto serial, conexión RS232.

- Desarrollar un programa en la computadora para recibir los mensajes enviados desde el teléfono celular, simulando un browser de correo que interpretará los mensajes recibidos, realizando una acción determinada en la pantalla del computador de destino.
- Elaborar un plan de negocios que permita llevar el prototipo a una etapa de comercialización.

1.3- Evaluación preliminar de la idea tecnológica.-

El desarrollo de la tecnología fue realizado por ingenieros y estudiantes de la ESPOL, en calidad de trabajo de tesis.

Actualmente existe en el mercado un gran número de dispositivos que envían información de manera inalámbrica tales como módems, tarjetas celulares que utilizan diferentes tecnologías. Lo que buscamos es abrir una puerta a nuevos desarrollos tecnológicos que sean ventajosamente competitivos en el mercado.

Sabemos lo difícil que es la competencia ya que nosotros aún estamos en escalas inferiores en comparación con otros países que diseñan ó desarrollan nuevas tecnologías y que además proponen cambios innovadores ayudando a mejorar nuestra calidad de vida.

1.4- Determinación del Modelo de Negocio

La competencia es una barrera muy sólida, difícil de escalar, que no debemos tomar a la ligera, quien como ellos para conocer al cliente, con la experiencia que tienen en el mercado, tendremos que enfocarnos para cambiar los esquemas de los clientes acostumbrados a prototipos diseñados, probados y ya comercializados en muchos mercados nacionales e internacionales, para que prefieran nuestro producto.

El usuario final es escéptico por naturaleza y mucho más con la nueva tecnología. Si el producto no lo vende una compañía de renombre se piensa que no es bueno y prefieren no arriesgar su dinero. Este mercado es muy especial, ya que un factor importante es el costo, el cual varía dependiendo de la aplicación, nuestro sistema sería dirigido para el uso del hogar por ejemplo, para saber si alguna luz quedó encendida luego de abandonar la casa, ó quizá para controlar el garaje de la misma, estar conscientes de la hora de entrada de las personas (hijos) cuando los padres se encuentran fuera, ó en el uso de personas con problemas de salud, en fin es un mercado potencial por la cantidad de hogares (volumen), pero debemos considerar si la inversión a realizar cubrirá esas “necesidades” para que el costo final del producto sea accesible al cliente y si generará la suficiente rentabilidad a la compañía que administraríamos.

Un punto crítico, probablemente, será la confiabilidad que tenga el sistema, muchas veces preferimos comprar un producto ya probado y de marca a algo

nuevo y desconocido. Para lograr una aceptación generalizada debemos realizar algún tipo de alianza estratégica que nos permita ingresar de una mejor manera en el mercado.

Al hacer una alianza debemos considerar la calidad y la garantía de nuestro producto porque dependemos del servicio que brindan otras compañías de telefonía celular, es decir, no podemos garantizar que nunca tendrá problemas, ya que en el momento que nuestro proveedor de servicio celular experimente inconvenientes, nosotros también los tendremos.

Debemos revisar si necesitamos licencias para nuestro sistema, directamente sería con la SUPTEL, CONATEL, y luego revisar estándares de otras compañías para cumplir con todas las normas que requiera la comercialización y distribución de nuestro producto para este tipo de tecnología.

1.5- Análisis de diferentes alternativas de negocio

Algunos de los beneficios proporcionados por el sistema de envío de información inalámbrica son:

- Permite enviar información desde lugares remotos, ya sea para almacenamiento o procesamiento. Esto ayudará a mantenernos informados en todo momento de lo que ocurre en otros lugares en tiempo real.

- Control de personas en cumplimiento de recorridos para mejorar las rutas a seguir de una manera más óptima, realizar cambio en los mismos inmediatamente sin necesidad de retornar a la matriz.

- Mejorar la respuesta humana a cualquier tipo de cambio que ocurre en el medio, para socorro inmediato de pacientes.

- Análisis de información en línea para procesarla inmediatamente y saber que acción tomar en el momento adecuado ya que contamos con la misma y podemos realizar estadísticas basándonos en ella.

- Otro beneficio tomando en cuenta que este dispositivo nos ayudará a conectarnos con muchos lugares que antes eran inaccesibles, será recopilar mayor información para análisis, evaluación y estadísticas que mejoraran el desarrollo en muchos aspectos ya que no necesitaremos montar una infraestructura, ni transportar equipos costosos de comunicación que generalmente son pesados y complejos.

1.6- Selección del modelo de negocio

El sistema a desarrollarse para que sea llevado a una etapa de comercialización será una de las alternativas propuestas en la sección anterior; escogimos un sistema de envío de información a los usuarios que poseen un equipo celular, provenientes de usuarios que no lo poseen, utilizando el mismo concepto de beeper.

La ventaja sera para aquellas personas que poseen ambos dispositivos, el celular y el beeper, ya que su equipo celular se encargara de realizar ambas funciones sin necesidad de depender del beeper, para lograrlo crearemos un call center que recibira las llamadas de los usuarios que deseen ubicarlos y nosotros le enviaremos un mensaje de texto con la información requerida.

1.7- Planificación de la aventura de negocio

La empresa se llamará DATACELULAR y consiste básicamente en ofrecer un servicio de envío de información donde los usuarios serán aquellas personas que posean celular y la ventaja competitiva es que recibirán mensajes de personas que no tienen un equipo celular, independientemente de la compañía con la que tienen el servicio de telefonía celular sea éste Porta, Telefónica ó Alegro, que son las empresas que actualmente prestan el servicio de telefonía celular en nuestro medio.

CAPÍTULO II

DESARROLLO

DEL

PRODUCTO

2.1 Diagrama de bloques del sistema.-

El diagrama de bloques nos muestra en la Figura 2.1.1 de forma macro como está diseñado nuestro prototipo y los diferentes dispositivos que interactúan y se conectan para completar su desarrollo.

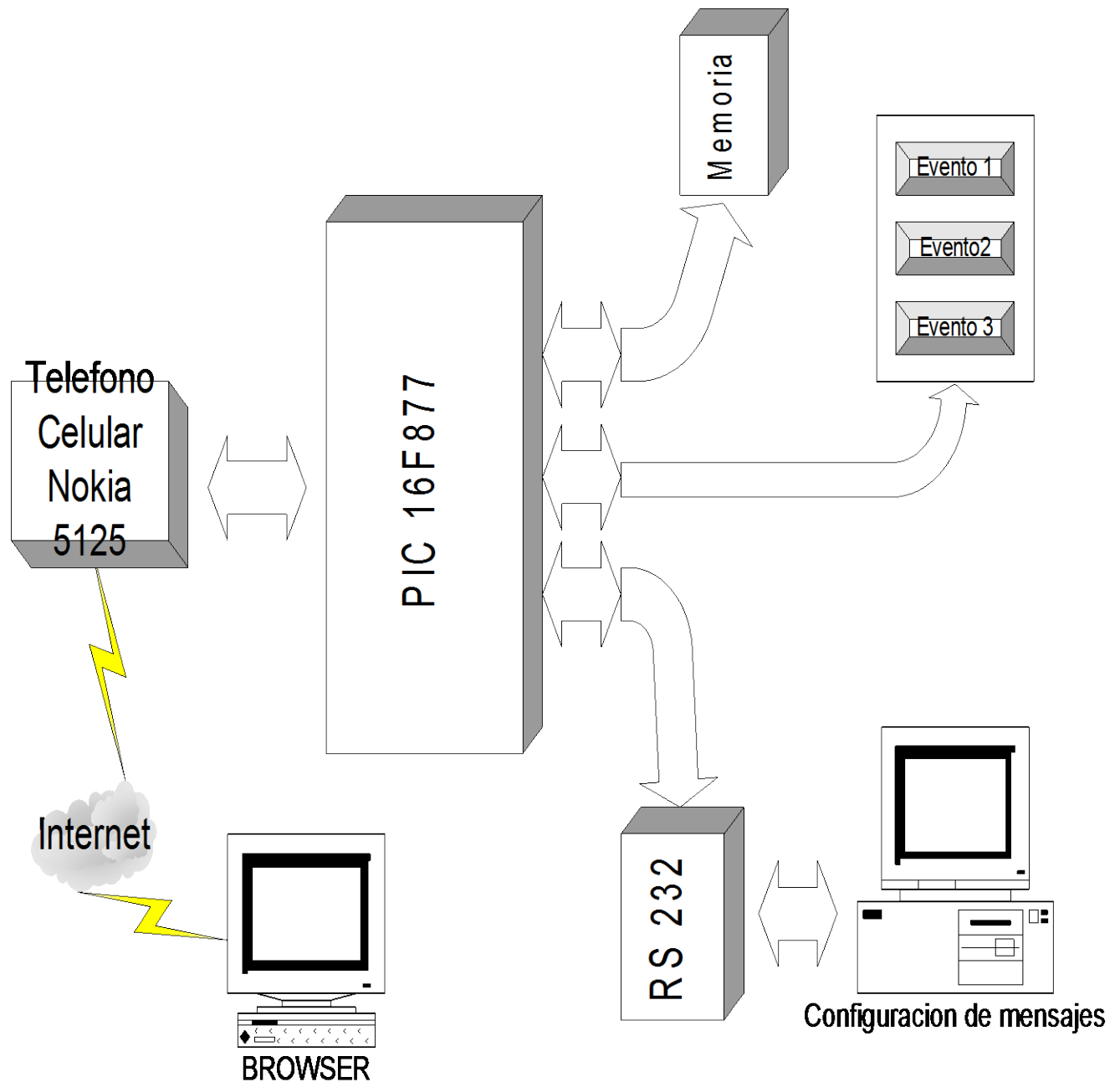


Figura 2.1.1 Diagrama de bloques del Sistema.

El prototipo esta orientado a cubrir varias aplicaciones tales como alarmas de hogar, sensores informativos o eventos no esperados, como se muestra en la figura.

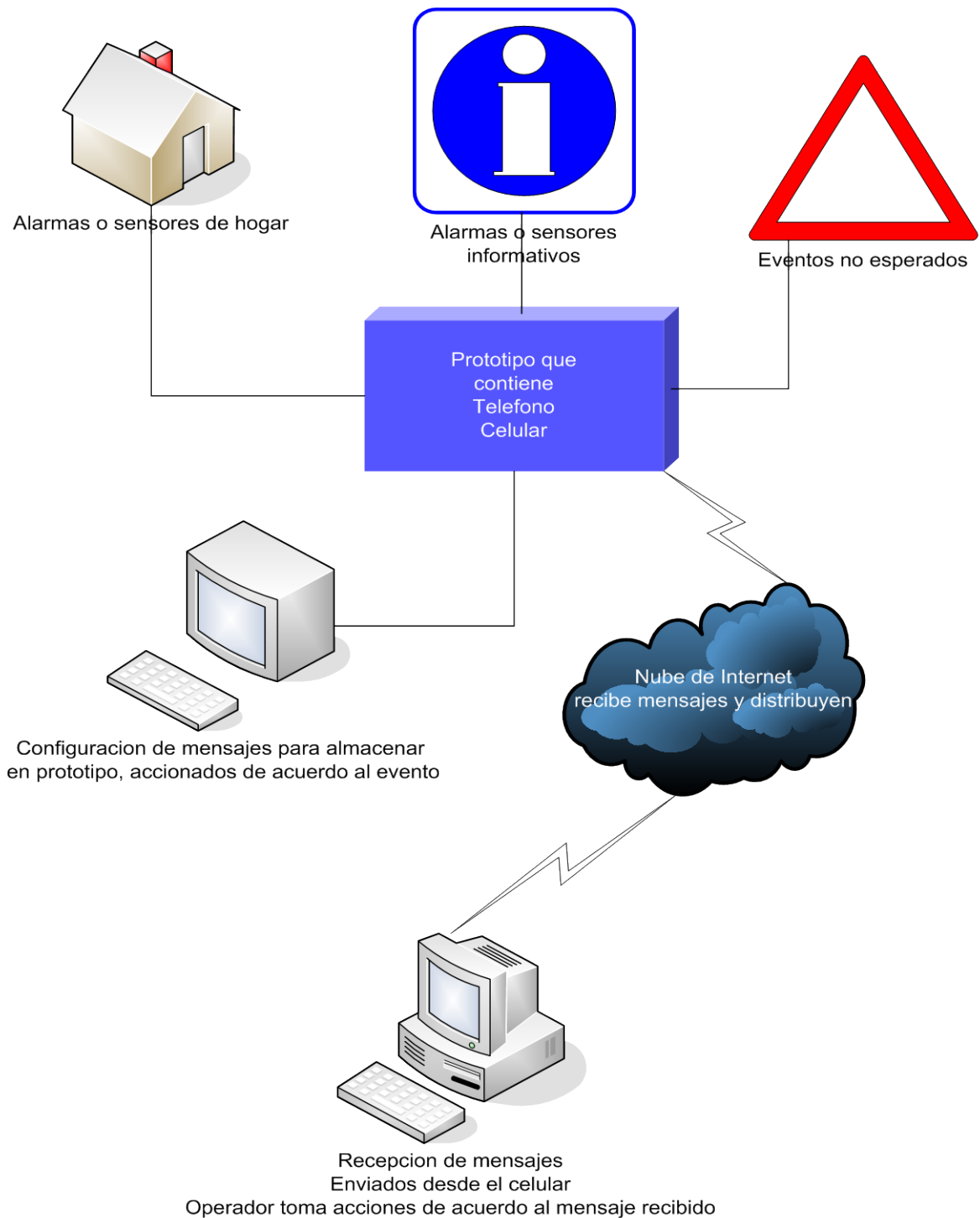


Figura 2.1.2 Aplicaciones que se pueden implementar.

Diagrama de flujo del proceso para la comunicación inalámbrica.

La siguiente figura muestra un digrama del proceso de comunicación inalámbrica que vamos a implementar.

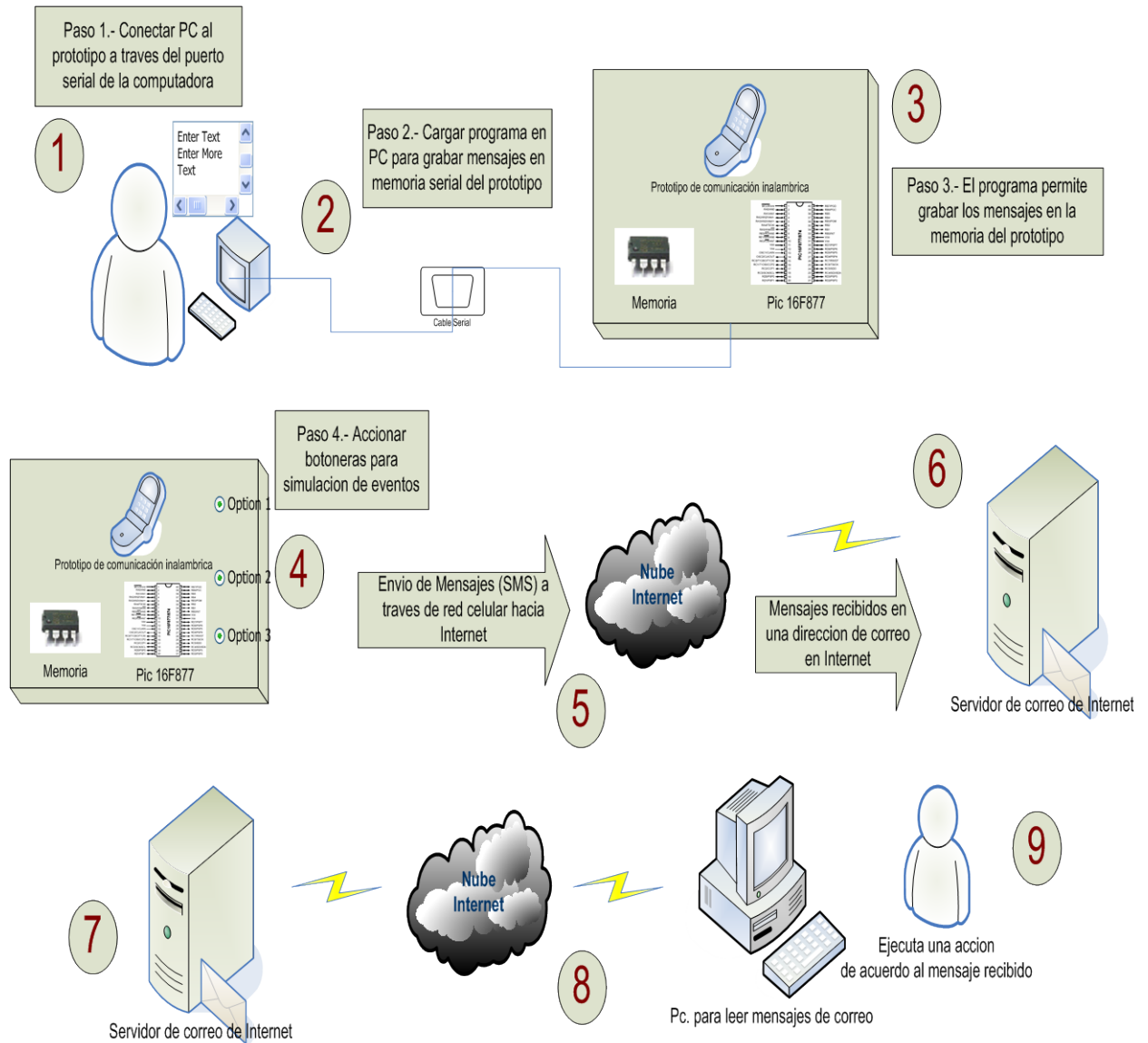


Figura 2.1.3 Diagrama de flujo del proceso de comunicación inalámbrica.

El módulo de control se basará en la arquitectura de la familia de los microcontroladores. Este es un circuito integrado de alta escala de integración que incorpora la mayor parte de los elementos que configuran un controlador.

Existen seis partes principales que componen nuestro sistema, cada una de ellas desempeña un rol muy importante para lograr la comunicación inalámbrica, las mismas que serán revisadas en este capítulo y se las definirá en el desarrollo de cada tema que a continuación detallaremos:

- Microcontrolador PIC16F877
- Teléfono Celular
- Max Rs 232
- Memoria
- Botoneras
- Switches (Quad Swicth 4016)
- Cables serial para conexión del Max RS232 y la computadora

El microcontrolador PIC16F877.-

Es el corazón del proyecto porque contiene el programa que maneja todas las subrutinas para administrar la memoria, el integrado RS232, las botoneras y los switches, que ayudan en conjunto al proceso de envío de información, en la figura 2.4 se muestra las conexiones del PIC con los dispositivos antes mencionados.

Un microcontrolador dispone normalmente de los siguientes componentes:

- Procesador o CPU (Unidad Central de Proceso)
- Memoria RAM para contener los datos
- Memoria para el programa tipo ROM/PROM/EPROM, FLASH.
- Líneas de Entrada y Salida para comunicarse con el exterior.
- Diversos módulos para el control de periféricos (temporizadores, Puertas Serie y Paralelo, CAD: Convertidores Analógico/Digital, CDA: Convertidores Digital/Analógico, etc.)
- Generador de impulsos de reloj que sincronizan el funcionamiento de todo el sistema.

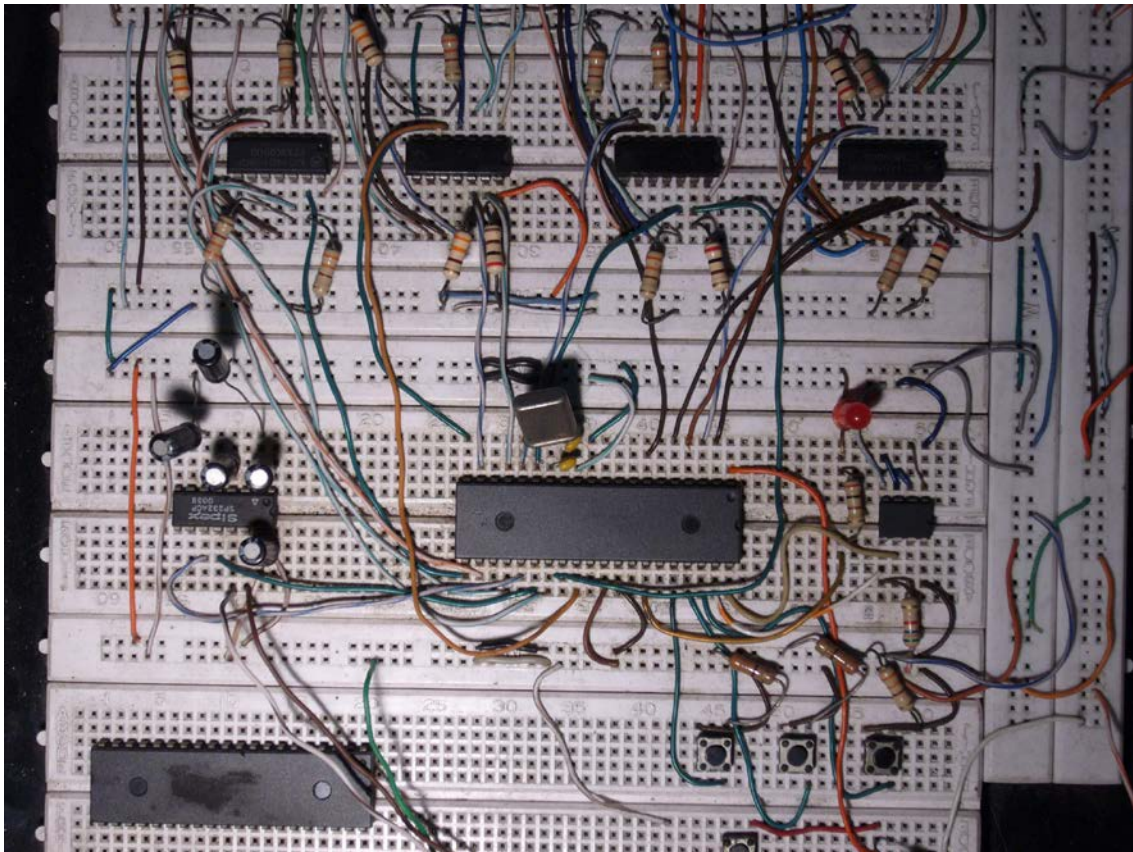


Figura 2.1.4 Protoboard con dispositivos conectados al PIC(centro).

El Módulo de Control comandará las funciones que realizará la comunicación inalámbrica tales como:

- Almacenamiento de datos en Banco de Memoria Serial al recibir una alerta.
- Comunicación serial con el PC, envío de la información almacenada en la memoria serial, y los interruptores digitales que controlan el teclado del teléfono celular.
- Visualización de datos en la pantalla del teléfono celular mientras es enviada la información.

Para el desarrollo del Módulo de Control de la comunicación inalámbrica se utilizó el PIC16F877 de la empresa Microchip (cualquier información adicional la podrá encontrar en el anexo o en la página Web: www.microchip.com).

Escogimos este dispositivo porque posee 40 pines que se acoplan muy bien a las especificaciones que necesitamos; para manejar el teclado del celular requerimos de 16 pines, para controlar el Max RS232 utilizamos 4 pines, para las botoneras 3 pines, para la memoria externa 24LC256, 2 pines, además de las conexiones de reloj y de las conexiones a tierra y la señal de alimentación +Vcc.

Además también consta de 5 diferentes puertos con sus respectivos pines de entrada / salida que se distribuyen en A(0 – 5), B(0 – 7), C(0 – 7), D(0 – 7), E(0 – 3), cada uno de ellos desempeña una función diferente y exclusiva propias de este microcontrolador, los pines del puerto A trabajan de manera independiente como entrada o salida y pueden ser configurados para trabajar de manera analógica o digital según nuestra conveniencia, los pines del puerto B son utilizados para trabajar como interrupciones tanto de entrada como salida, los pines del puerto C nos ayudan a controlar la comunicación serial de los dispositivos externos que se conecten al PIC, tales como el RS23. Por ejemplo, los pines del puerto D trabajan como entrada o salida de manera indiferente así como los pines del puerto E.

Este dispositivo posee muchas ventajas y se acopla perfectamente a nuestras necesidades, por lo que consideramos que es una de las piezas fundamentales de nuestro prototipo.

El teléfono celular.-

Es otro de nuestros componentes principales porque contiene una serie de integrados que no necesitamos identificarlos pero que en conjunto sirven como medio de transporte para enviar la información. La función principal de este dispositivo es enviar mensajes de texto. Para lograr controlar las teclas del teléfono utilizamos una placa propiedad de Nokia, figura 2.5, que fue diseñada para propósitos de configuración y pruebas de funcionamiento de estos equipos,

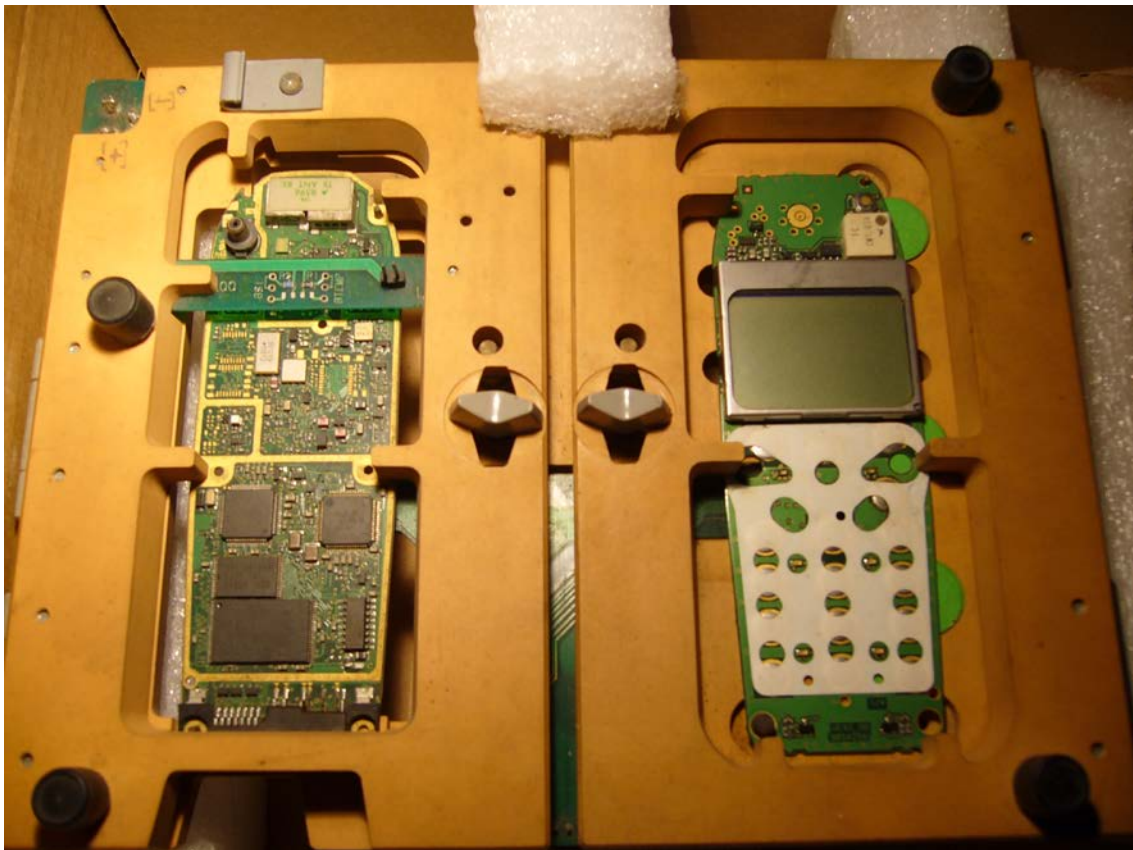


Figura 2.1.5 Placa para pruebas de configuración y funcionamiento.

Escogimos este dispositivo porque es muy comercial y existe una gran cantidad de equipos de este modelo en el mercado; nuestra placa donde reposa el teléfono puede soportar varios modelos de teléfonos.

La secuencia que utilizamos se basa en el modelo 5125 mostrado en la figura 2.6, en esta placa podríamos utilizar otro modelo de celular pero debemos cambiar la configuración de las teclas en nuestro programa, debido a que cada modelo tiene diferentes posiciones en las teclas, por ejemplo en el modelo 6120 la tecla MENU esta en la posición del botón CLEAR del 5125, y las teclas arriba y abajo en la posición de la tecla MENU, es decir si cambiamos el equipo celular, debemos cambiar la secuencia de las teclas en el PIC para obtener los mismos resultados.



Figura 2.1.6 Teléfono Nokia modelo 5125 desarmado e introducido en la placa.

La función del teléfono es ayudar a enviar información a través de mensajes de texto. La placa donde se encuentra empotrado tiene pines que cuando los combinamos y les asignamos un voltaje ejecutan una acción en el teléfono que puede ser observada en la pantalla del celular. Luego de muchas pruebas pudimos descifrar las teclas del celular y que sirven para controlarlo, en total son 16 combinaciones que son voltajes suministrados por el PIC para que el celular ejecute una acción de acuerdo a nuestras necesidades.

Cada tecla del celular es controlada por el PIC, a continuación se detalla en la tabla 2.1 las conexiones para que el teléfono escriba los caracteres enviados por el PIC.

Tabla 2.1.1 Descripción de las teclas definidas en el PIC

<i>Pines del PIC</i>	<i>Teclas del celular</i>
2	<i>Abajo</i>
3	<i>Menú</i>
4	<i>Arriba</i>
5	<i>Clear</i>
7	<i>4 (cuatro)</i>
15	<i>7 (siete)</i>
16	<i>* (asterisco)</i>
17	<i>2 (dos)</i>
19	<i>8 (ocho)</i>
20	<i>0 (cero)</i>
21	<i>3 (tres)</i>
22	<i>6 (seis)</i>
24	<i># (numeral)</i>
27	<i>9 (nueve)</i>
28	<i>1 (uno)</i>
29	<i>5 (cinco)</i>

Por ejemplo para lograr que aparezca el numero 1 en la pantalla del celular, es necesario que el PIC envíe una señal de voltaje positivo o alto (+Vcc= 5 V o señal lógica 1) al Pin 28 (RD5) del PIC, cuando esta señal es recibida por el respectivo pin de control del switch 4016, se enviara una señal en alto, esto ocasionará que en la placa del teléfono se genere un carácter asignado del teléfono de acuerdo al cuadro especificado en el teclado de teléfono celular, en

la siguiente figura se muestra las conexiones de la placa con el proto a través de un bus de datos,

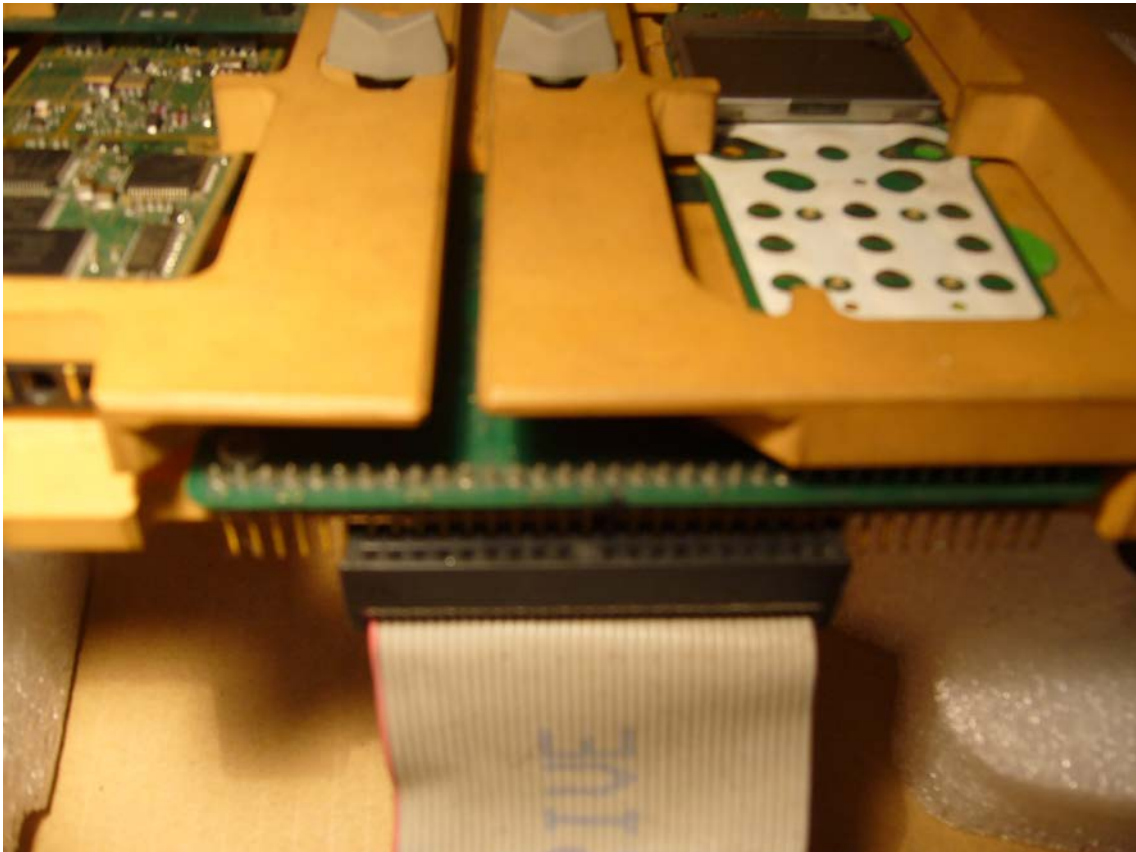


Figura 2.1.7 Placa conectada al proto utilizando un bus de datos.

El Max RS232.-

Este integrado sirve como interfase para lograr la comunicación entre el PC y el microcontrolador, el PIC tiene la particularidad de conectarse a cualquier dispositivo de manera serial o paralela de acuerdo a nuestro requerimiento.

El Rs232 permite unir físicamente cualquier dispositivo a través del puerto serial de una computadora, es decir simulamos una conexión directa entre el PIC y la computadora gracias al RS232 y así logramos enviar caracteres entre ambos dispositivos.

En la computadora se ejecuta un programa para interactuar con el puerto serial de la misma, que al estar conectado con el puerto serial se abre una puerta entre el Pic y la PC, permitiendo que los caracteres que se escriben en la pantalla del monitor a través del programa sean enviados al PIC y almacenados en la memoria externa LC256, en la siguiente figura se muestra el proto con el Max 232 conectado con un cable con terminación DB9 para conectarse al puerto serial de la computadora.

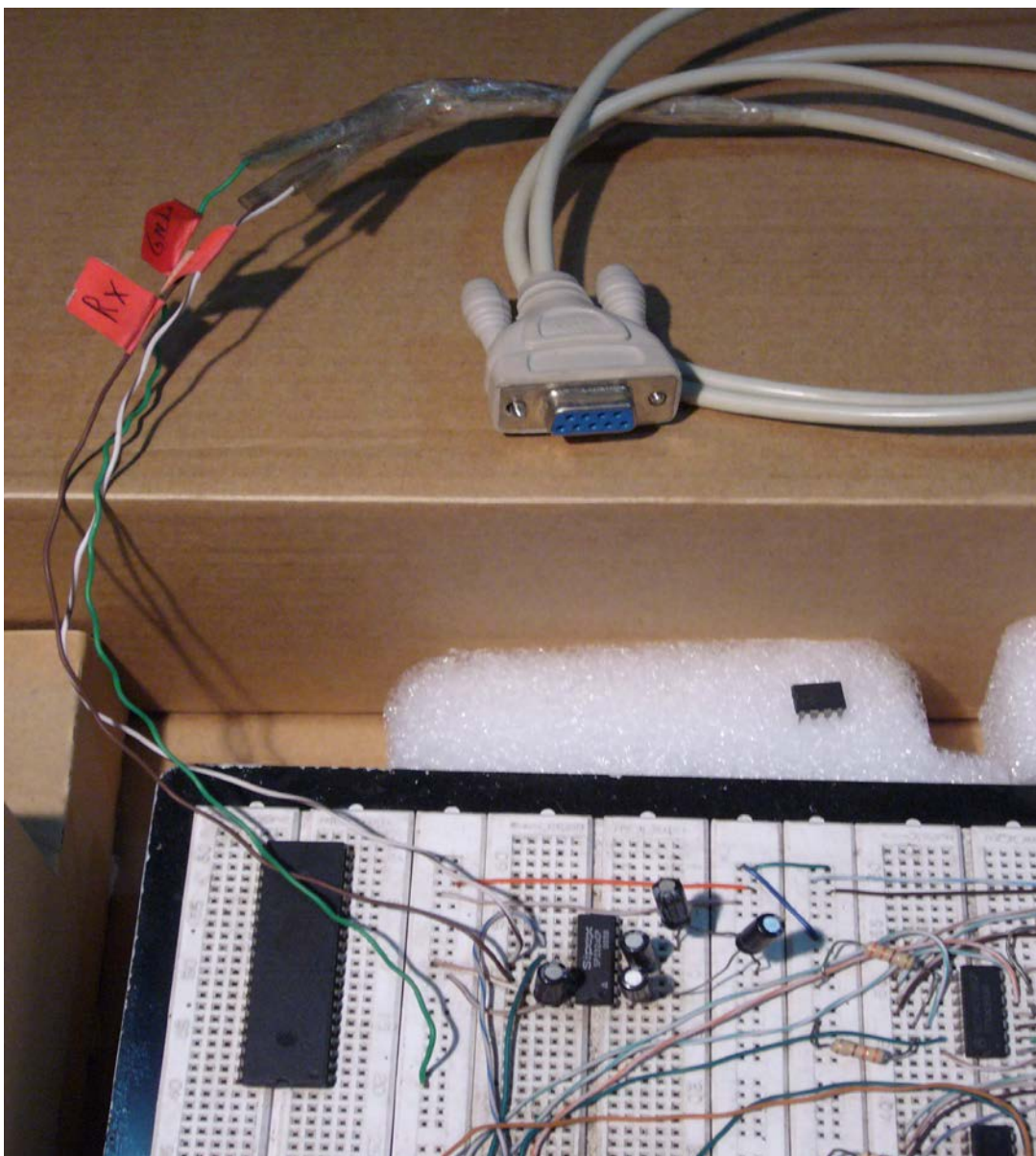


Figura 2.1.8 Conexión entre el computador y el proto a través de un cable DB9.

El Max RS232 para nuestro dispositivo estará basado en el siguiente ítem:

- *Comunicación Serial entre PC y el PIC16F877.*

Para lograr el enlace al PC lo realizaremos por el puerto serial que tiene internamente implementado un módulo UART para el envío y recepción de datos.

Al utilizar la comunicación serial desde el puerto serial de la PC (DB-9), donde se encuentra implementado en un chip el módulo denominado UART, esto es de convertir los BYTES del bus paralelo del PC a un flujo de bits en serie. En tal virtud se tendrá que configurar la comunicación entre la PC y el prototipo de la comunicación inalámbrica en modo de comunicación Serial Asíncrona; La señal que ha sido acondicionada en niveles de voltaje por el circuito integrado MAX232 que es ingresada al PIC16F877 es a través del PIN 26 (RC7 / RX), con ello permite que los caracteres que se escriben en la pantalla del monitor sean enviados al PIC y luego almacenados en la memoria externa LC256.

Memoria.-

Este dispositivo externo lo utilizamos para guardar la información que vamos a enviar a través del teléfono celular.

Su función principal es mantener la información almacenada para luego ser utilizada por el PIC cuando ocurra una interrupción que está definida por alguna de las tres botoneras.

El modo de comunicación con este dispositivo es el I2C en modo Maestro, siendo el PIC16F877 quien actúa como maestro y la memoria 24LC256K como el esclavo. Al utilizar la memoria 24LC256K, esta trabajará en el modo de comunicación I2C, siendo esta esclavo del PIC16F877, a continuación se muestra la figura 2.8 con las conexiones en el proto de la memoria serial.

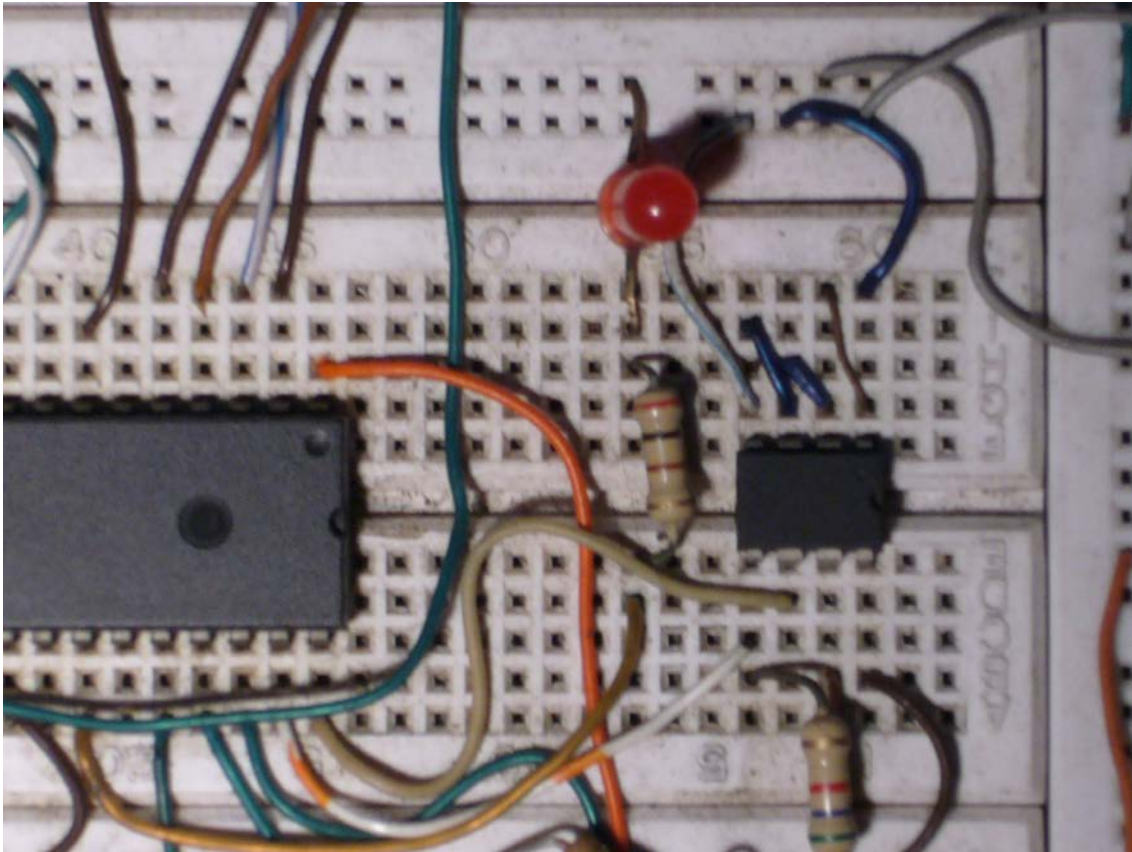


Figura 2.1.9 Conexiones entre la memoria y el Pic en el Proto.

Rutina para almacenar datos del PC a la memoria usando protocolo I²C

Para el almacenamiento de las señales de alarma de nuestro prototipo en la memoria serial, debemos llevar el control de la dirección del último dato almacenado en la memoria 24LC256 en las direcciones 41 y 42 de la memoria eeprom del PIC 16F877; se leen los datos que están almacenados en la memoria eeprom del 16F877 y se van grabando uno a uno dichos datos en la memoria mencionada.

Ver Anexo D

Procedimiento que almacena datos en la memoria I2C

Este procedimiento almacena los datos en la memoria 24LC256, recibe a manera de parámetros el registro slaveor que ayuda a definir con cual de las 4 memorias se realizará la operación de escritura definiendo con los bits del 1 al 3 la memoria en cuestión.

Slaveor tiene el siguiente formato 0000xxx0

000 memoria 0

001 memoria 1

010 memoria 2

011 memoria 3

En primera instancia se genera el bit de start: cambio de alto a bajo en dato con reloj en alto, posterior se envía el byte de control, recibe el bit de aceptado, luego envía byte de dirección el msb primero y luego el lsb, posteriormente se envía el byte del dato para finalizar con un byte de stop:

Características de comunicación de la memoria 24LC256K

Para la conexión de la memoria tomaremos los PINES # 1-2-3 que son las entradas de dirección del chip, A0, A1, A2.- Las entradas A0, A1, A2 son usadas por los 24XX256 para funcionamientos múltiples del dispositivo. Los niveles en estas entradas se comparan con los correspondientes niveles de dirección del esclavo. El chip es seleccionado si la comparación es verdadera.

Pueden ser conectados hasta ocho dispositivos al mismo bus de datos usando diferentes combinaciones de selección del chip.

En nuestro prototipo la información que grabamos en la memoria la recibimos desde el computador, para recibir cada uno de los caracteres enviados por el

PC, conectamos los pines 39 (RB6/PGC) y 40 (RB7/PGD) del PIC y los pines 5 - 6 de la memoria serial; en conjunto el envío de los caracteres que están interactuando con el RS232 con el uso del programa de ingreso de la información le permite al usuario escribir cada uno de los caracteres que desea enviar, luego en el programa presionamos el botón grabar y la información es enviada a la memoria.

Cuando se presiona una de las botoneras o al producirse un indicativo de una acción el PIC se encargara de leer la dirección de memoria correspondiente y enviar cada uno de los caracteres almacenados hacia el equipo celular.

Botoneras.-

Estos dispositivos ayudan a simular una acción externa, emulando un evento de acuerdo a la configuración que hayamos implementado.

Cada botonera le indica al PIC que mensaje debe leer de la memoria externa para enviar la secuencia respectiva al teléfono celular; cada mensaje leído es nuevamente procesado por el Pic y enviado carácter por carácter al telefono celular, para interpretar los comandos enviados hacia el celular, utilizamos los switches 4016 que forman la interfase entre el celular y el PIC.

Cada uno de los botones representa un mensaje que será enviado al celular al momento de accionarlo, para configurar los mensajes utilizamos un programa que se encargara de grabar la información y enviarla al PIC para que este a su vez la envíe a la memoria externa.

Aparte existe una cuarta botonera que nos ayuda a encerrar el proceso de envío del PIC y detenerlo en caso de ser necesario, en la figura 2.9 se muestra las botoneras implementadas en el protoboard.

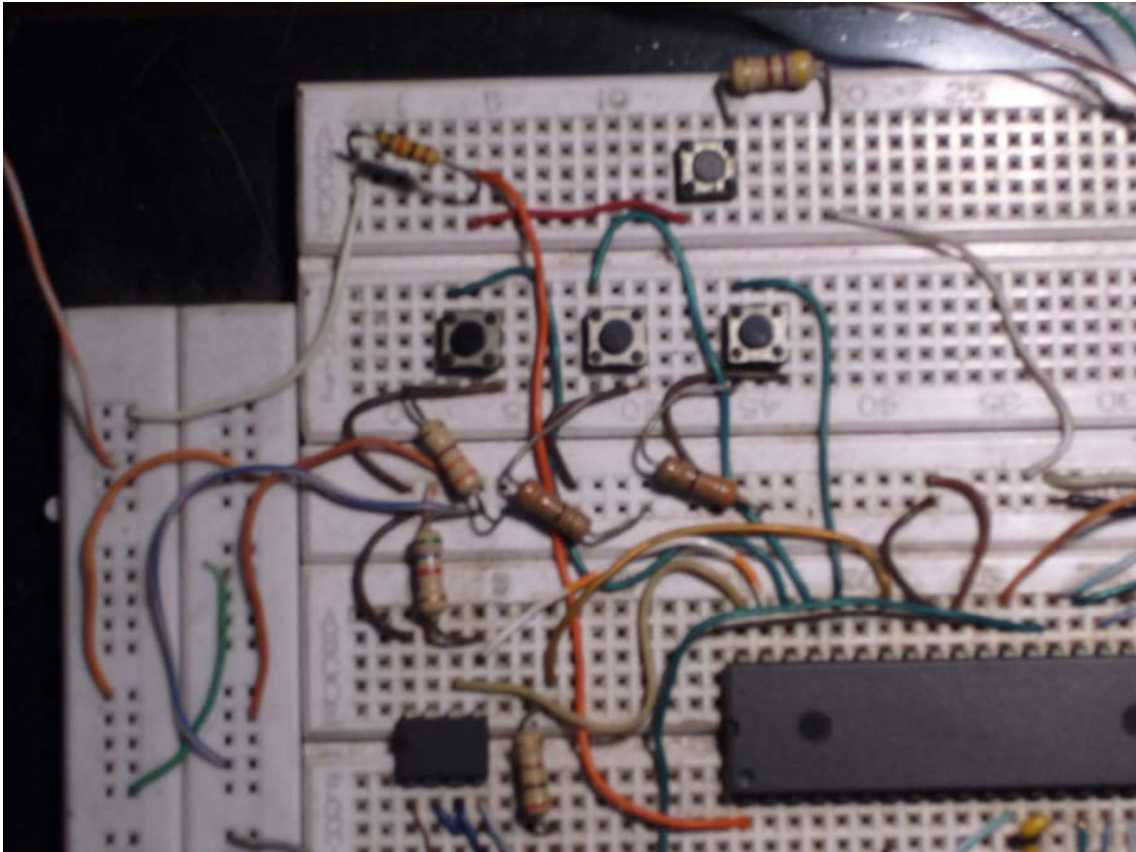


Figura 2.1.10 Botoneras implementadas para simular eventos.

Switches 4016

Estos switches juegan un rol importante para la comunicación entre el PIC y el celular. Los pines del PIC van conectados a los controladores de cada switch para accionar una tecla del teléfono celular.

Los switches van conectados al teléfono celular a través de un bus paralelo generalmente utilizado para la comunicación de los discos duros de las computadoras con la controladora o mainboard.

2.2 Diseño e implementación detallada.-

En la siguiente figura se muestra la implementación mas detallada de nuestro prototipo.

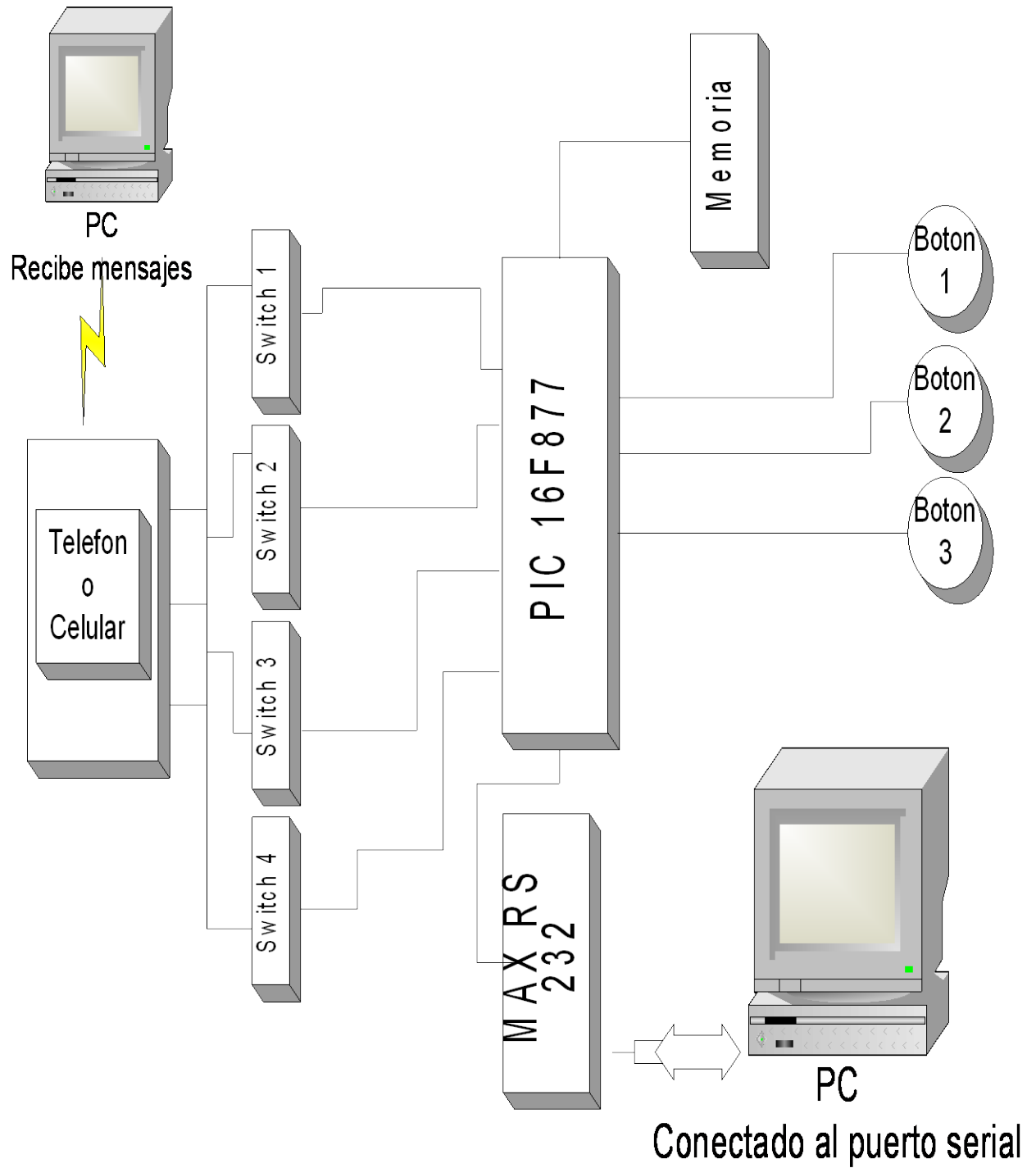


Figura 2.2.1 Diseño e implementación mas detallada del proyecto.

2.2.1 Desarrollo de interfase para controlar teclado del celular.-

Primeramente fue desarrollada la interfase para controlar cada una de las teclas del teléfono celular. Con la ayuda del manual de NOKIA se estableció como estaba configurado dicho teclado y se realizaron las pruebas necesarias para lograr la configuración.

La identificación de cada una de las teclas se la logró utilizando una placa propiedad de Nokia que ayuda a probar y revisar la configuración de los teléfonos celulares de dicha marca, además se puede detectar problemas en el funcionamiento de los mismos para luego cambiar piezas y partes cuando estas presentan inconvenientes.

Esta placa se compone de dos partes para albergar el equipo, para ubicar el teléfono celular es necesario desarmarlo y poner cada una de las placas del mismo en cada una de las ranuras correspondientes como se muestra en el gráfico.

Una vez ensambladas ambas partes del teléfono, en la parte inferior de la placa se encuentra 50 pines que definen la configuración del equipo y además permiten controlar el teclado del celular, he aquí la parte mas importante para desarrollar el proyecto que es identificar cada una de las teclas utilizando la placa anteriormente mencionada.

En esta placa existen 10 pines que controlan el funcionamiento de las teclas del teléfono celular (1, 2, 3, 4, 5, 6, 7, 8, 9, 0, #, *, up, down, menu, clear), para esto se aplican voltajes en dos pines al mismo tiempo, cuando esto ocurre aparece un numero, un símbolo, una letra o una secuencia del teléfono en la pantalla del celular.

Por ejemplo si le damos voltaje a los pines 5 y 6 de la placa en la pantalla del celular aparecerá el número 3

Configuración de las teclas en el programa principal

num1

```
movlw b'00100000'  
movwf PORTD  
return
```

abajo

```
movlw b'00000001'  
movwf PORTA  
return
```

```
;  
; Tecla CLEAR  
;  
; Pin 3 PIC  
;  
; Pin RA1 Salida (Port A Bit 1) '00000010'  
;  
; Control 2 - IC1
```

borra

```
movlw b'00001000'  
movwf PORTA  
return
```

```
;  
; Tecla MENU  
;  
; Pin 4 PIC  
;  
; Pin RA2 Salida (Port A Bit 2) '00000100'  
;  
; Control 3 - IC1
```

menu

```
movlw b'00000010'  
movwf PORTA  
return
```

```
;  
; Pin 5 PIC  
;  
; Pin RA3 Salida (Port A Bit 3) '00001000'  
;  
; Control 4 - IC1
```

arriba

```
movlw b'00000100'  
movwf PORTA  
return
```

```
;  
; Tecla Número 4  
;  
; Pin 6 PIC  
;  
; Pin RA4 Salida (Port A Bit 4) '00010000'  
;  
; Control 1 - IC2
```

num4

```
movlw b'00100000'
```

```
movwf PORTA
return
```

;Configuración para el PUERTO RC (tiene 8 puertos E/S)

```
; Tecla Numero 9
; Pin 15 PIC
; Pin RC0 Salida (Port C Bit 0) '00000001'
; Control 3 - IC2
```

num9

```
movlw b'00010000'
movwf PORTD
return
```

```
; Tecla Numero #
; Pin 16 PIC
; Pin RC1 Salida (Port C Bit 1) '00000010'
; Control 4 - IC2
```

numer

```
movlw b'10100000'
movwf PORTC
return
```

```
; Tecla Numero 7
; Pin 17 PIC
; Pin RC2 Salida (Port C Bit 2) '00000100'
; Control 1 - IC3
```

num7

```
movlw b'10000001'
movwf PORTC
return
```

```
; Tecla Numero '*'
; Pin 18 PIC
; Pin RC3 Salida (Port C Bit 3) '00001000'
; Control 2 - IC3
```

aster

```
movlw b'10000010'
movwf PORTC
return
```

```
; Tecla Numero 2
; Pin 19 PIC
; Pin RD0 Salida (Port D Bit 0) '00000001'
; Control 3 - IC3
```

num2

```
movlw b'10000100'
```



```

movwf PORTC
return

;   Tecla Numero 5
;   Pin 20 PIC
;   Pin RD1 Salida (Port D Bit 1) '00000010'
;   Control 4 - IC3
num5
movlw b'01000000'
movwf PORTD
return

;   Tecla Numero 8
;   Pin 21 PIC
;   Pin RD2 Salida (Port D Bit 2) '00000100'
;   Control 1 - IC4
num8
movlw b'00000001'
movwf PORTD
return

;   Tecla Numero 0
;   Pin 22 PIC
;   Pin RD3 Salida (Port D Bit 3) '000001000'
;   Control 2 - IC4
num0
movlw b'00000010'
movwf PORTD
return

;   Tecla Numero 3
;   Pin 23 PIC
;   Pin RC4 Salida (Port C Bit 4) '00010000'
;   Control 3 - IC4
num3
movlw b'00000100'
movwf PORTD
return

;   Tecla Numero 6
;   Pin 24 PIC
;   Pin RC5 Salida (Port C Bit 5) '00100000'
;   Control 4 - IC4
num6
movlw b'00001000'
movwf PORTD
return

```

Para realizar el control de las teclas previamente identificadas recurrimos a los switches 4016 que son interruptores digitales que trabajan con 4 señales de control además de 4 entradas y 4 salidas (en la figura se muestra la información técnica de estos dispositivos).

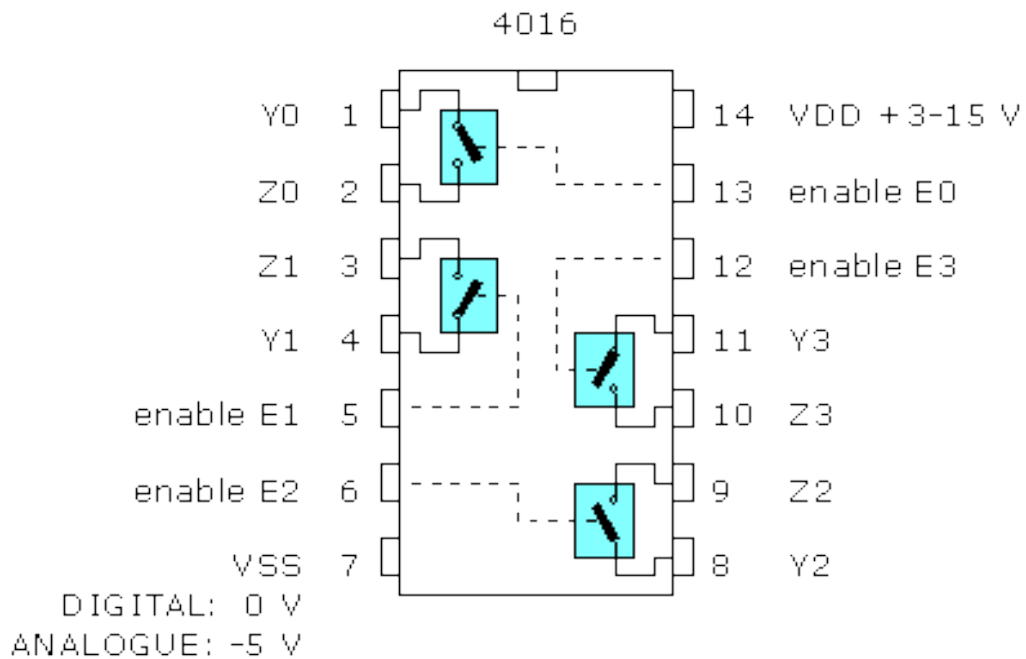


Figura 2.2.1.1 Conexiones internas del dispositivo switch 4016.

Funcionamiento

El funcionamiento del dispositivo y todos sus accesorios se lo detalla en el siguiente diagrama de flujo para un mejor entendimiento del proceso de manera global.

1.- Tenemos una computadora para configurar los mensajes y almacenarlos en la memoria del circuito.

2.- Tenemos un telefono celular que nos ayuda a enviar los mensajes de una forma inalambrica utilizando la tecnología celular.

3.- Existen 3 botoneras que representan cada una, la accion que accionara el dispositivo para que envíe el mensaje a traves del telefono celular.

4.- Ademas tenemos una persona que va a estar monitoreando los mensajes recibidos y se encargara de ejecutar la orden recibida.

Para controlar el teclado utilizamos el chip 4016 que trabaja a +5Voltios DC. En el Pin #14 y en el Pin #7, se lo conecta a tierra con lo cual el chip queda polarizado además tiene 4 entradas, 4 salidas y 4 pines de control en donde dedujimos que el teclado del celular esta establecido en una matriz de distintos niveles de voltaje ya que con el uso de un voltímetro se pudo establecer los niveles de voltaje..

Todas las combinaciones del teclado del celular se conectaron con un grupo de chips 4016 en una parte del protoboard, se definieron los pines controladores para que se pueda identificar cada tecla del celular y que sean manejadas por el microcontrolador 16F877, además en dichos pines se agregaron resistencia de 300 ohmios para que el nivel de voltaje dado por el PIC pueda proporcionar un efecto retardado de accionamiento sobre las teclas del celular.

Conexiones entre los switches 4016 y el telefono celular.

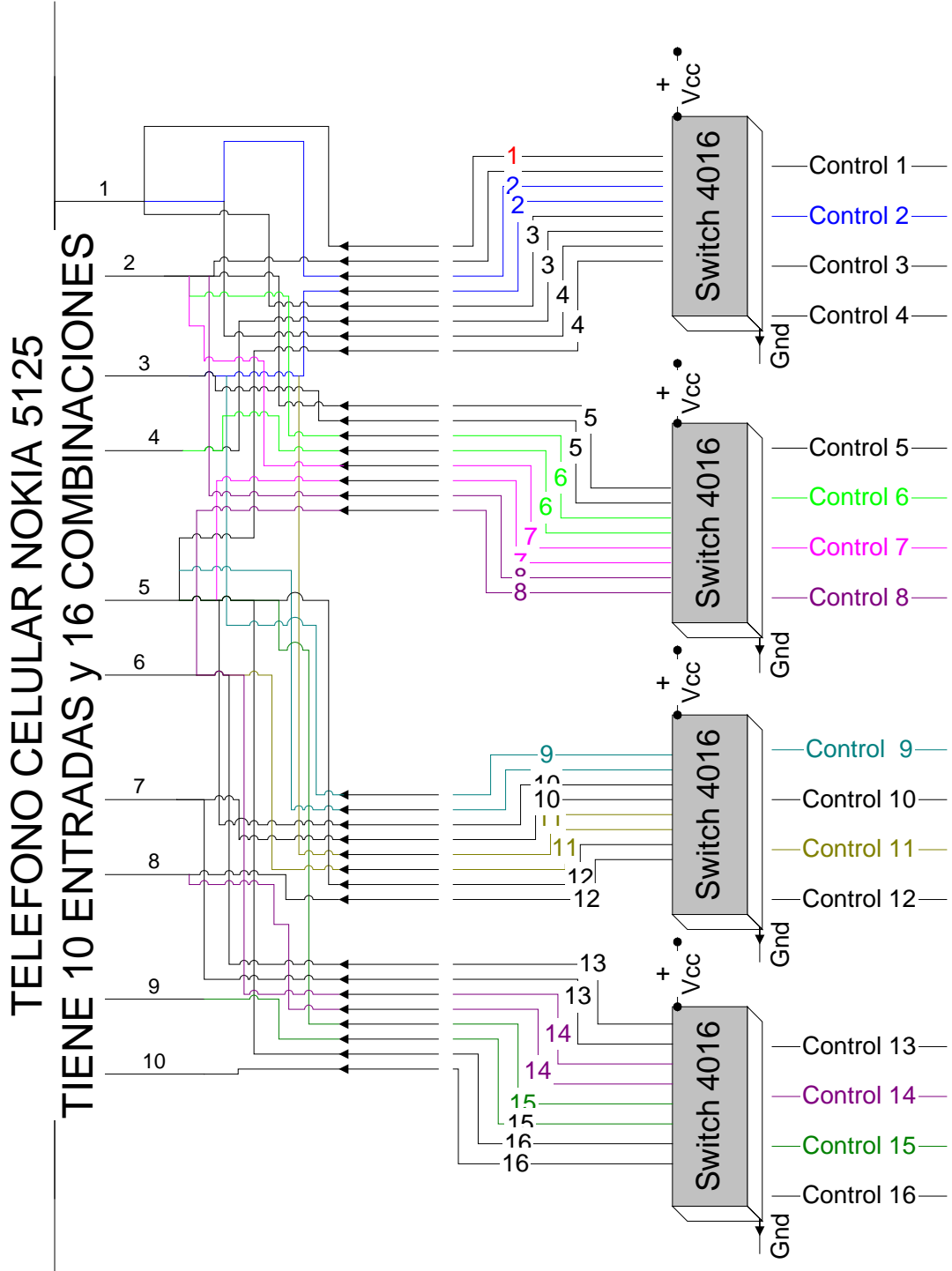


Figura 2.2.1.2 Conexiones entre los switches 4016 y el teléfono.

Conexiones entre el teléfono celular, los switches y el PIC 16F877.

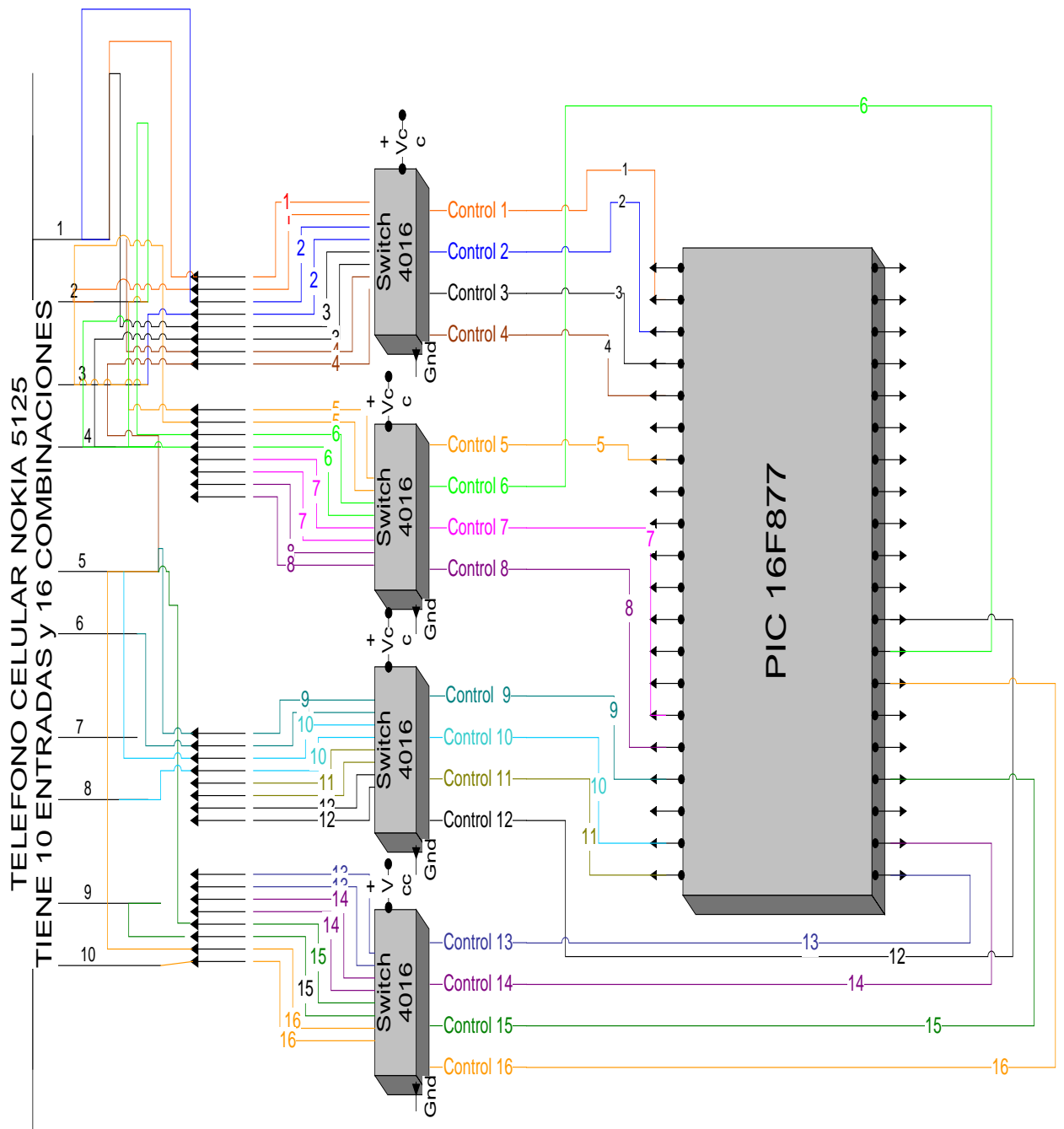


Figura 2.2.1.3 Conexiones entre los switches 4016, el teléfono celular y el PIC.

Conexiones entre el PIC 16F877 y los demás componentes

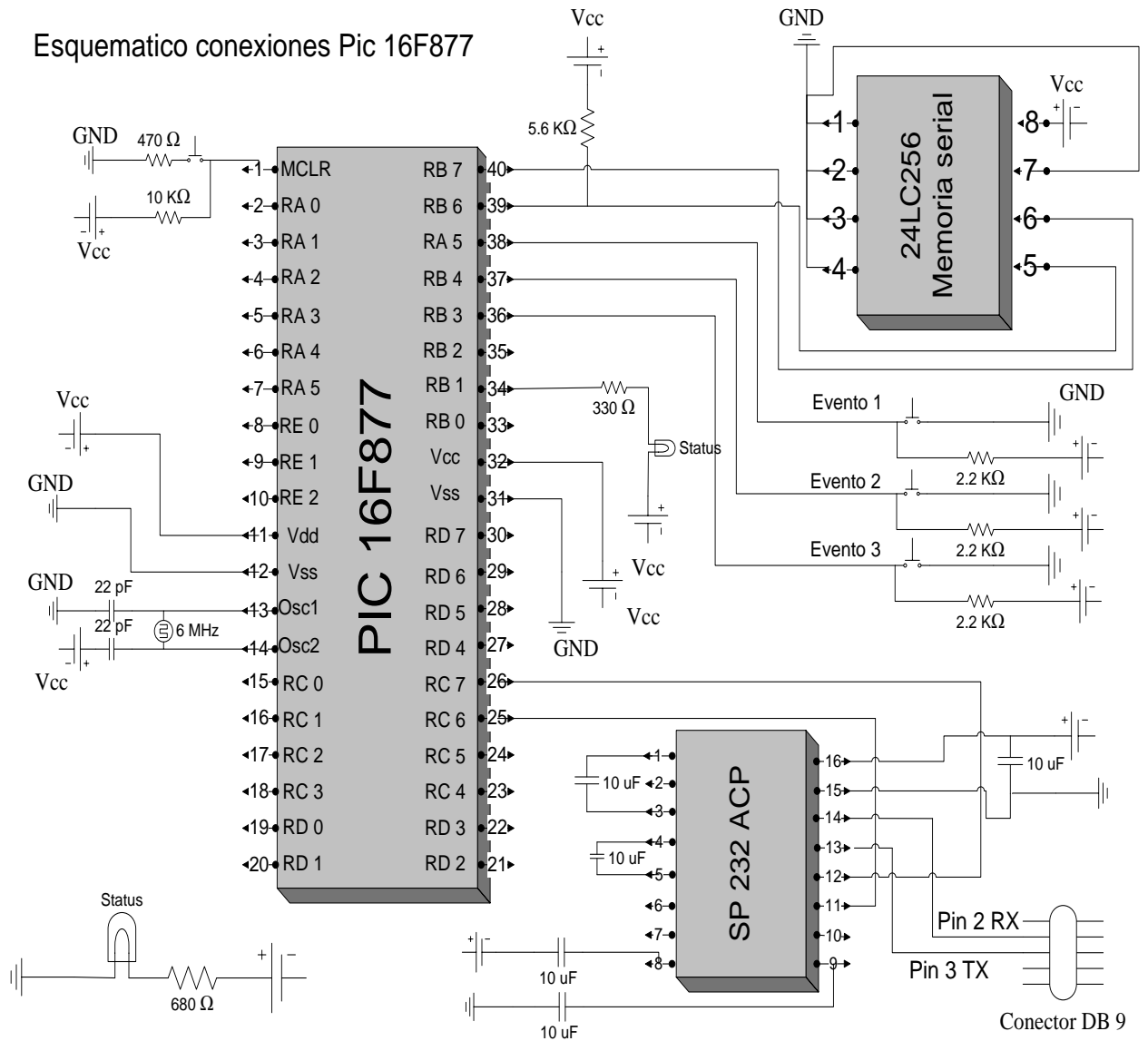


Figura 2.2.1.4 Esquemático del PIC y los demás componentes.

La siguiente figura muestra la implementación de todos los dispositivos en el protoboard

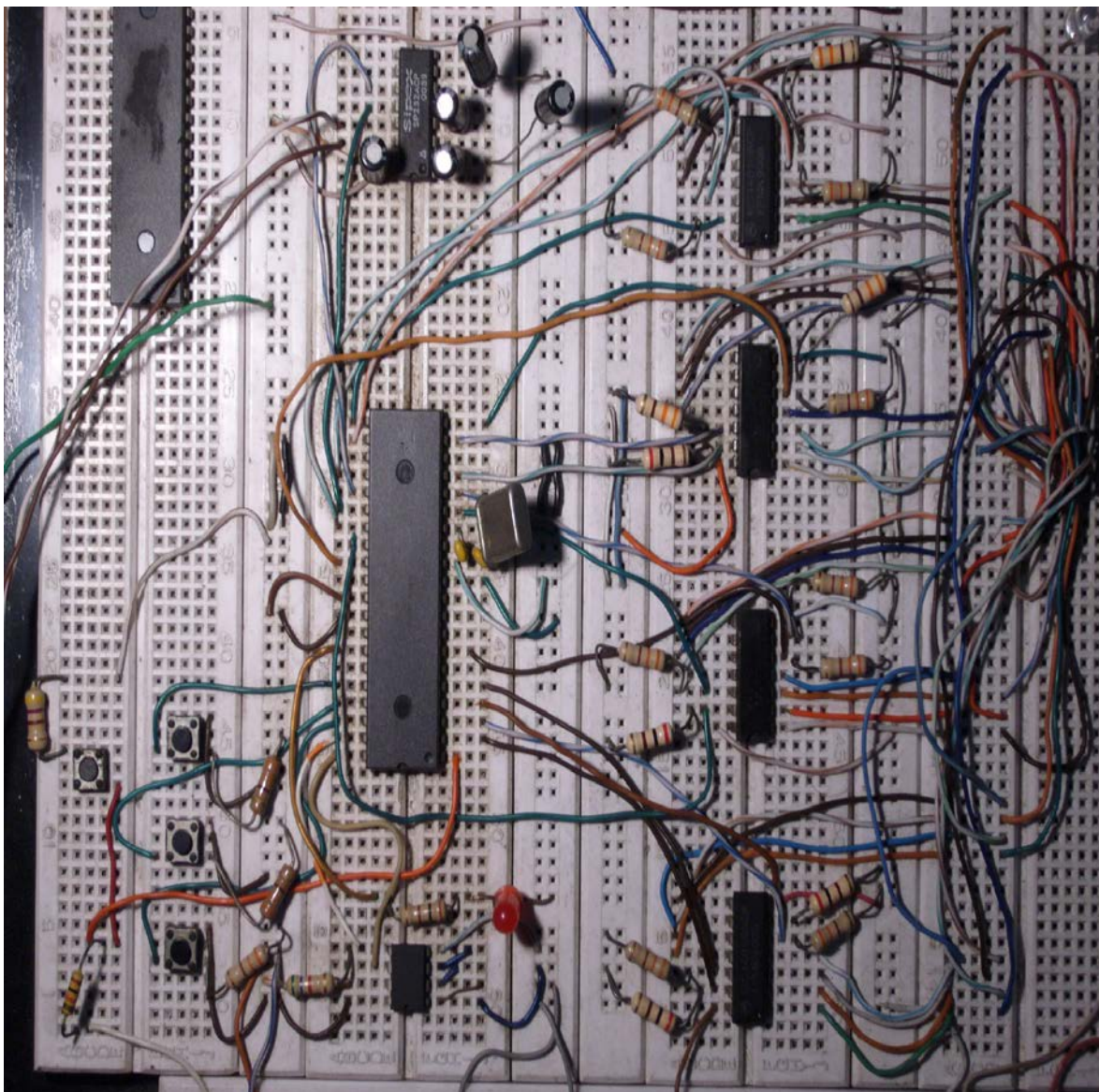


Figura 2.2.1.5 Conexiones de todos los dispositivos y el PIC en el protoboard.

2.2.2 Comunicación serial con la computadora utilizando Microcomputadores.

Para lograr la comunicación con la computadora utilizamos la configuración del PIC con la interfase RS-232 y utilizamos un dispositivo digital como el máx. 233 que hará de interprete entre la comunicación digital y la señal serial tomamos los pines 18, 17 que permiten configurar el PIC con el protocolo RS-232 en el PIC, se establece una línea de programa en la cual está configurado en un formato UART .

Para lograr la comunicación serial al interpretador MAX 233 se lo polariza como lo especifica la ficha técnica de dicho chip y se tomará los pines 11-12 como entradas y salidas para transmisión y recepción de niveles de voltajes.

Hay que considerar que la comunicación serial será a 8Bits y que en el programa del PIC se establecerá ese mismo rango para que se efectúe tanto la transmisión como la recepción.

Para realizar la conexión del MAX-233 utilizaremos los PINES 2 y 3 que se comunican con el puerto serial del PC. Cuando realicemos esta conexión debemos tener mucho cuidado con este puerto ya que cualquier conexión incorrecta puede ocasionar un daño permanente en el integrado.

ESQUEMATICO DE LA COMUNICACIÓN SERIAL .-

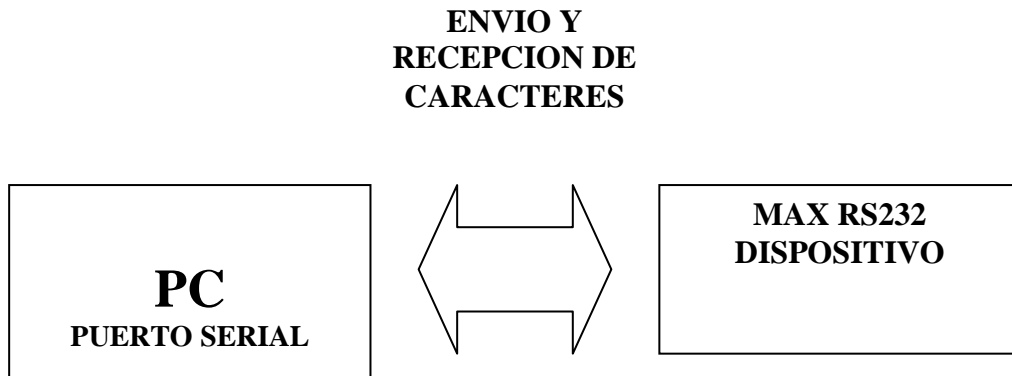


Figura 2.16 Esquemático de la comunicación serial.

2.2.3 Interfase para la comunicación entre el teclado del celular y la computadora

La interfase de comunicación del teclado del celular con la computadora se la logro gracias a la circuitería diseñada que se encuentran entre estos el teléfono celular y el computador.

Desarrollamos un programa en Visual Basic que nos ayuda a enviar los mensajes desde la computadora hacia el PIC y a su vez este último almacena los mensajes en la memoria LC256. La información allí almacenada es leída luego por el PIC y enviada al dispositivo celular a través de una secuencia que puede estar definida por el tipo de mensaje que deseamos enviar.

Este programa tiene además una opción que nos muestra una tabla de los mensajes enviados desde el puerto serial de la computadora hacia la memoria administrada por el PIC, luego utilizando las botoneras puedo decidir que mensaje deseo enviar.

En la pantalla del programa podemos configurar hasta tres mensajes de texto de 120 caracteres máximo, debido a las limitantes de los SMS (Short Message Sending), que esta ligado a la tecnología celular y que solo permite enviar esta cantidad finita de caracteres.

La función principal del programa es definir o configurar los mensajes que deberán estar en la memoria almacenados y que podrán ser enviado en cuanto un evento ocurra.

La simulación de los eventos esta determinada por tres botoneras que representan cada uno de los eventos, así por ejemplo si presionamos el botón 1, el PIC enviara al celular la secuencia del mensaje almacenado en la dirección de memoria asignada para este evento.

El botón 2 así mismo enviara una secuencia diferente que será interpretada por nuestro programa de lectura de mensajes de correo y luego será mostrada en la pantalla al operador para tomar una acción específica.

Hemos tratado de automatizar el proceso y lo hemos logrado en un alto porcentaje aun siendo tan amplia la ciencia de las telecomunicaciones y de la electrónica.

Nuevamente hacemos hincapié en que el desarrollo de este prototipo lo hicimos basándonos en la tecnología existente en nuestro medio, quisimos lograr obtener un producto mixto utilizando tecnología ya desarrollada sin ahondar en descubrir o saber como funciona algo ya probado, simplemente lo utilizamos y nos valimos de la tecnología para desarrollar un nuevo producto.

2.2.4 Programa para leer mensajes enviados por el celular

Para leer los mensajes que el prototipo envía a través del teléfono celular se diseñó un interpretador que consiste en un programa en DoNet que lee el mensaje en background y luego lo muestra en la parte inferior de la pantalla.

Para lograr leer los mensajes que envía el dispositivo tuvimos que idearnos una forma de interpretarlos, al principio quisimos leer los mensajes directamente del teléfono celular pero lamentablemente la tecnología celular no nos permitió ingresar al equipo y obtener la información del dispositivo celular de destino.

Se presento un reto poder interpretar el mensaje que había sido enviado por el teléfono celular, por lo que para poder llevar a cabo la segunda parte de este proyecto tuvimos que decidir enviar el mensaje a una dirección de correo de cualquier dominio y luego a través de funciones ya desarrolladas para leer y enviar correo o email, interpretarlas haciendo un emulador que lea el mensaje en background y luego sea mostrado en la pantalla del operador que estará monitoreando los mensajes que ingresan en forma secuencial. Esta tarea es la más importante de nuestro prototipo ya que no conseguimos nada con solo

enviar los mensajes lo importante es saber que dice el mensaje que fue enviado cuando este sea recibido por su destinatario.

El programa esta desarrollado en DoNet y permite a través de varias funciones leer el mensaje y luego interpretarlo para que sea leído sin que el operador tenga que abrirlo manualmente como un correo normal.

También tenemos la opción de leer el mensaje de manera manual, existen ambas opciones para los usuarios, los mensajes quedan almacenados en el repositorio de correo normal.

2.2.5 Programa para monitoreo de mensajes.

Este programa consiste en una ventana visible en la parte inferior de la pantalla, nos ayuda a mostrar los mensajes que se reciben a una dirección de correo específica.

Cuando recibimos un mensaje nos muestra en la parte inferior durante 20 segundos el mensaje, todas las ventanas de windows quedan deshabilitadas mientras dura el mensaje en la pantalla, luego de recibido el mensaje la ventana se deshabilita, para que el usuario continúe haciendo sus tareas normales.

Todos los mensajes que son enviados a través del dispositivo son almacenados utilizando el mismo programa para enviar mensajes con esto validamos que la información enviada es correctamente recibida por el receptor.

Ambos programas pueden ejecutarse en la misma computadora para realizar las pruebas de comunicación para envío y recepción de mensajes.

CAPÍTULO III:

PLAN

DE

MERCADEO

3.1.- Resumen Ejecutivo.-

El grupo de estudiantes del Tópico de Graduación “Desarrollo de Productos con Base Tecnológica”, desarrollará el prototipo de un sistema que enviará información de manera inalámbrica, utilizando tecnología celular. Este sistema permitirá enviar mensajes de texto que serán recibidos por sus destinatarios, ya sea a un dispositivo celular o a una dirección de correo electrónico.

El proyecto planteado esta orientado a brindar un servicio de mensajería, para las personas que sin tener un equipo celular puedan comunicarse con aquellas que sí lo poseen, sin importar con que operadora de telefonía celular trabajen.

En el estudio de mercado se definirá las pautas a seguir para que esta idea tecnológica sea aceptada en el mercado tecnológico. El análisis de las cinco C's nos muestra en forma global los sectores que se ven involucrados y su repercusión.

El siguiente análisis(FODA) está orientado a describir la situación favorable y/o desfavorable del producto. Calidad, precio, lugar(place), promoción y pláticas positivas son generalidades a tratarse en el análisis de las cinco P's. Por último tenemos el análisis financiero que nos aclara el panorama económico en el cual podemos desarrollar el proyecto.

3.2.- Análisis de Mercado y Plan de Mercadeo

Últimamente nuestro mercado tiene un crecimiento de manera exponencial en cuanto a tecnología se refiere, ya que en la actualidad contamos con sistemas capaces de mostrar información, imágenes y videos en la pantalla de un equipo celular, lo cual implica que el avance tecnológico es muy vertiginoso.

Actualmente existen productos que brindan servicios de envío de mensajes a través de dispositivos de ubicación de personas, nuestra ventaja con respecto a ellos es que el usuario que solicite este servicio no necesitará comprar dicho equipo, sino que su teléfono celular le servirá para este fin.

En el mercado existen aproximadamente cinco millones quinientos mil abonados, que poseen equipo celular, de un total de 7 millones de potenciales usuarios.

Nuestro mercado serán aquellas personas que formen parte de ese grupo privilegiado que poseen celular, y que sus amigos y/o familiares, que no tienen un equipo celular puedan enviarles mensajes, para hacerlo simplemente deberán llamar a nuestro call center y solicitarnos enviar un mensaje a la persona que ellos nos indiquen.

Nuestra competencia son todas aquellas compañías que en la actualidad proveen un servicio de mensajes utilizando beepers, los cuales tienen un costo aproximado de \$50 que sumado a un equipo celular \$100, da un total de \$150, la propuesta será dar un valor agregado al celular, es decir que sirva como beeper, sin necesidad de adquirir uno, ya que su celular le servirá para el mismo fin.

3.3.- Análisis Técnico

Análisis FODA

➤ Fortaleza

-Proceso sencillo y económico. Suponen una base sólida para la financiación de la iniciativa y a favor del cliente.

-Fuente diferenciación de nuestra oferta. Por la originalidad del planteamiento como por lo atractivo de sus prestaciones.

-Variedad de la gama de empresas que dan servicio de telefonía celular lo cual nos favorece extraordinariamente.

➤ Oportunidades.

-Coyuntura absolutamente favorable ya que desarrollamos tecnologías que proponen cambios innovadores que ayudan a mejorar nuestra calidad de vida.

-Feliz precedente de un positivo desarrollo tecnológica lo que nos permite actuar con seguridad, reduciendo errores y logrando aciertos.

-Inexistencia de competencia directa. Se trata de un mercado abierto, sin barreras de entrada

-Amplitud del mercado. La demanda potencial es muy elevada.

-Gran avidez de este tipo de servicios. La índole de necesidades tecnológicas que cubren son importantes.

➤ **Debilidades.**

- Falta de licencia para nuestro sistema por parte de CONATEL, SUPTEL organismos encargados.

-Necesidad de revisar estándares de otras compañías para cumplir con todas las normas que exigen la comercialización y distribución de nuestro producto para este tipo de tecnología.

-Los inconvenientes de la novedad. La novedad puede ocasionar problemas de falta de credibilidad.

-La barrera del precio. La cifra del precio para algunos segmentos va a suponer un obstáculo difícil de superar para muchos.

➤ **Amenazas.**

-Posible aparición de competencia directa. Es de prever que otros grupos o entidades pretendan competir con nosotros.

-Competencia por los propios organismos empresariales. La oferta actual y futura de innovaciones tecnológicas por parte de empresas de telefonía.

-Competencia indirecta de otras entidades. Aunque sus ofertas sean distintas suponen concurrencia.

-Posibles cambios legislativos, políticos, etc. la actual coyuntura puede sufrir cambios en sentido negativo.

3.4 Análisis Administrativo

El grupo empresarial está conformado por:

- Gerente General.
- Vendedores
- Jefe de Ventas
- Operadoras de Call Center
- Operador Técnico.

A continuación se presenta un resumen de la experiencia, capacitación y habilidades de 2 de las expectativas de nuestro personal del grupo empresarial.

Gerente General

Sexo: Masculino.

Edad: 24 años en adelante.

Disponibilidad de Tiempo.

Nivel Académico: Profesional en Economía y carreras Administrativas.

Experiencia Laboral: 3 años en puestos similares.

Habilidades: Organización, Dirección, Planificación.

Poseer alta capacidad de Liderazgo, manejo de personal y técnicas de negociación.

Funciones:

1. Planear, organizar, dirigir y controlar las actividades que se realizan en la empresa.
2. Junto con el Director elaborar el plan estratégico de la compañía.
3. Encargarse de las ventas.
4. Supervisar el presupuesto y su grado de cumplimiento.
5. Planeación del RRHH, junto con el jefe de Operaciones.

Operador Técnico

Estudios realizados: Ingeniero en Computación especialidad Sistemas Tecnológicos, graduado en la Escuela Superior Politécnica del Litoral.

Experiencia Laboral: 2 años en el Sector de Desarrollo de proyectos con microcontroladores.

Funciones:

1. Planear, organizar, dirigir y controlar las actividades que se realizan en el área técnica.
2. Investigación de nuevas tecnologías.

Análisis de las 5 P's.-

Producto.-

El producto en si es el servicio que se da a los usuarios que llamen, pues ya la tecnología esta implementada a nivel de celulares, y utilizaríamos esta tecnología para dar el servicio de call center.

Precio

Esto es algo a estimarse de acuerdo a varios factores pero que siendo el valor mínimo aún así representará una gran ganancia

Lugar (Place).-

El lugar en si seria una oficina donde estaría ubicado el call-center, administración y servicio al cliente. Al Inicio sería un lugar pequeño pero con una visión a expandirse al futuro.

Promoción

La promoción se puede realizar en los mismos estados de cuenta y lugares de expendio de servicios celulares, además se puede conversar con las empresas de telefonía convencional a nivel nacional para llegar a un acuerdo legal.

Pláticas Positivas (Positive W.O.M)

Se espera poder conversar debidamente con los ejecutivos de las compañías telefónicas tanto de celulares como convencionales y nos den facilidades para poder llegar a sus usuarios a través de sus informativos para que sea un beneficio más para ellos mismos, ya que se espera que nos pidan un porcentaje de ganancia por el uso de su tecnología.

3.5.- Análisis Legal y Social

La Empresa DATACELULAR se suma al mercado de las telecomunicaciones que quieren ser líder en el mercado de Ecuador y la red de telecomunicación.

La tecnología de DATACELULAR, está basada en una proyección estadística de que 7 de cada 10 usuarios utilizarían su servicio de telefonía inalámbrica.

La empresa DATACELUAR, busca una cobertura a nivel nacional, para lograr este objetivo dependemos de una alianza estratégica con las empresas que proveen el servicio de telefonía celular, ya que dependemos directamente de ellos para funcionar, esto implica llegar a un acuerdo en el que ambas partes quedan satisfechas desde el punto de vista legal y comercial, promoviendo así

una competencia abierta en caso de existir otras empresas que pretendan brindar el mismo servicio.

Dentro de la investigación de mercado la empresa DATACELULAR crea:

- Un sistema de comunicación inalámbrica para el mercado local.
- La importancia de la mezcla del marketing mix para el posicionamiento en el mercado de la telefonía celular.
- Una forma rápida y eficiente para la cobertura local
- Plantear el problema de la telefonía celular en el mercado local.
- Averiguar las aptitudes y comportamiento de la competencia.
- La hipótesis en la creación y desarrollo del servicio de mensajería en la telefonía celular.

3.6.- Análisis de valores personales.

La empresa DATACELULAR, es un proyecto que trata de llegar al mercado de las telecomunicaciones, sabiendo que en nuestro país éste es realmente competitivo, ya que cuenta con tres competidores Otecel, Conecel y Alegro, que son compañías dedicadas al mercado de la telefonía celular, además de las que usan dispositivos tipo buscapersonas.

Dentro de este análisis enfocamos el valor de la empresa en ofrecer puesto de empleo, en apoyar al cliente interno que es su personal, motivación diaria a su equipo de trabajo, son los aspectos más importantes, que le interesa a nuestra empresa, dentro del marco personal y laboral.

Uno de sus aspectos principales de este proyecto, es el hecho de que todas las familias ecuatorianas no pueden contar con una línea telefónica celular, debido a que en ocasiones es difícil para el usuario adquirir este tipo de servicio, por lo que este proyecto ha sido creado para satisfacer las expectativas de clientes potenciales. Para esto, la empresa DATACELULAR ha considerado en este proyecto el comportamiento del mercado, la cual se ve afectado por la crisis económica que existe en el país. Además de la falta de cobertura en ciertas regiones en donde llamar a un celular es muchas veces caro debido a la realidad económica de comunidades pobres en donde ni siquiera tienen las conocidas cabinas telefónicas ó cybers cafés sino unas cuantas líneas telefónicas convencionales.

3.7.- Análisis económico y financiero

De acuerdo con los análisis del Foda, podemos observar que esta empresa puede tener cabida en este mercado, porque los equipos de telefonía celular implementada por la empresa, son de tecnología de punta, ya que este servicio pueda ser requerido por los clientes, para eso la empresa a tomado en consideración a la competencia en su organización planeación y ejecución de sus estrategias, la empresa DATACELULAR luchará por tener una participación dentro del mercado de la telefonía celular, proyectándose día a día generar servicio y calidad, al entrar en este mercado, reactiva el aparato productivo y al mismo tiempo conceder empleo a varias familias ecuatorianas.

Análisis de las 5 C's.-

➤ La Compañía

La misión de la compañía será llegar a cada una de las personas que no pueden comunicarse a través de equipos celulares por los costos que

estos representan, teniendo como alternativa un servicio de llamada a un numero convencional a través del cual podrá solicitar enviar mensajes a cualquier persona que posea un teléfono celular y que conste en nuestra Base de Datos.

El negocio será netamente de servicio, tratando de ubicarnos en un sitio de prestigio utilizando nuestro dispositivo para envío de mensajes.

➤ **Los Clientes**

Este negocio se basa en prestar un servicio de comunicación para todas las personas que no posean un teléfono celular y deseen enviar mensajes a personas que si posean un celular.

Nuestros clientes serán directamente las personas que tengan celular y contraten nuestros servicios. A partir de ese momento constarán en nuestra base de datos y cualquier persona que llame a nuestra compañía y solicite enviar un mensaje a esta persona podrá hacerlo.

De esta forma se crea una puerta de acceso más rentable para aquellas personas que se ven marginados del sistema de telefonía celular.

➤ **Los Competidores**

En la actualidad hay una competencia bien marcada por saber quien tiene la mejor tecnología lo cual ha llevado a estas empresas a competir por brindar un mejor servicio de telefonía móvil, pero esto cuesta la comunicación entre las operadoras a través de mensaje de texto.

➤ **Los Colaboradores**

Recogiendo estas premisas nuestra empresa utilizará las bondades de cada una y brindará un servicio abierto, para que todas las personas que llamen a nuestra empresa puedan solicitar enviar mensajes de texto a abonados tanto de Porta, Telefónica ó Alegro, logrando así unir la información y a las personas sin importar que tecnología estén utilizando.

➤ **El Contexto/Macroambiente**

El ambiente que nuestra empresa utilizará es dar un servicio abierto, para que todos los posibles clientes potenciales llamen a nuestra

empresa, requiriendo que se les envíe un mensaje de texto al abonado que puede ser de la competencia sea esta Porta, Telefónica ó Alegro, logrando así unir una buena información y a las personas sin importar que tecnología estén utilizando.

3.8.- Análisis de riesgos

Nuestra empresa DATACELULAR es una empresa que ofrece un servicio basado en los servicios de telefonía inalámbrica, conjuntamente con sus competidores que ofrecen el mismo servicio, debemos enfocarnos en el servicio que vamos a ofrecer en este mercado que es extenso debido a que estamos en una globalización de mercado que permite tener competitividad, en el cuál se ofrece los servicios, imagen y calidad de producto.

Esta necesidad nos obliga a cumplir con nuestros clientes, de la manera mas placentera, para con ello satisfacer sus necesidades y requerimientos.

El análisis de mercado está enmarcado en definir las actitudes de los clientes, el comportamiento de la competencia y definir claramente el buen o mal servicio de la telefonía celular el mercado interno o local.

Como en toda aventura de negocios existen muchos riesgos que ya fueron analizados detenidamente en nuestro análisis FODA, pero que es importante volverlo a indicar, se trata de la factibilidad de que tengamos la aceptación de las personas ya sea por el precio o por políticas internas a nivel legal o a nivel empresarial si no se llega a los convenios definidos previamente.

En fin existen los mismos riesgos que pueden existir en cualquier empresa en el sentido de si vamos a tener una fuente de ingresos suficiente para solventar

todos los gastos incurridos, pagar a nuestros empleados y obtener una utilidad que convierta rentable nuestra empresa y el servicio que esta prestará.

3.9.- Evaluación Integral del Proyecto

La empresa DATACELULAR quiere llegar a tener un buen posicionamiento de este mercado, para lo cual pondrá todos sus esfuerzos para la consecución de sus objetivos.

Otro aspecto que es de suma importancia para la empresa es como debe estar difundido el servicio que se va a ofrecer a sus posibles usuarios, dicha difusión, debe estar planificada a través de programas en vivos, en ferias, en revistas, en prensa escrita, estados de cuenta, etc. para así llegar en una manera más directa a nuestros clientes.

La razón de difundir el servicio, es para dar a conocer las características, cualidades y ventajas, que la empresa va a ofrecer para competir en el mercado.

La empresa DATACELULAR ha realizado en este proyecto una implementación en sus estrategias acorde con las condiciones que ofrece el mercado.

Este servicio que la empresa DATACELULAR, está lanzando al mercado, es para un determinado segmento, es decir para las personas que no tienen el servicio de telefonía, para lo cuál la empresa utilizará toda su nueva tecnología, para la consecución de aperturar el mercado interno y externo y así atraer a sus posibles clientes.

La empresa DATACELULAR, busca cubrir una necesidad de personas que no se pueden comunicar a través del celular porque no lo poseen, en este sentido nuestra empresa brindara la conexión o puente que ayudara a estar siempre en contacto con nuestros seres queridos o companeros de trabajo, sin dejar de utilizar la tecnología que cada dia cambia para mejorar nuestros estandares de vida en este mundo globalizado.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

Una vez terminado el análisis y luego de haber concluido con el prototipo necesitamos hacer una aclaración con respecto a su funcionamiento ya que el mismo fue desarrollado con la tecnología de hace aproximadamente 3 años, esto implica que en la actualidad no necesariamente sea un servicio comerciable, debido a los cambios importantes que tenemos en cuanto a telefonía celular se refiere, por ejemplo la entrada de una nueva operadora y además del cambio de tecnología CDMA y GSM, las cuales proporcionan nuevos terminales que traen funciones importantes mejoradas con respecto a la tecnología inicial utilizada en nuestro proyecto, TDMA, la misma que ya no es tan utilizada por las operadoras.

Luego de realizado el plan de negocios y de haber revisado los valores y variables detenidamente podemos concluir que este proyecto es viable, el único detalle es conseguir el dinero para aventurarnos en el mundo de los emprendedores y fundar nuestra propia empresa.

La inversión sería retornada al cabo del primer año por lo que podemos afirmar que nuestro proyecto es rentable, desde varios puntos de vista.

La culminación de este proyecto ha representado para nosotros un verdadero reto porque ha involucrado muchos campos de trabajo, que antes eran desconocidos para nuestra carrera técnica de ingeniería que siempre busca concretar trabajos de índole técnico valga la redundancia.

Este proyecto nos ayudo a conocer el mundo de los emprendedores y además a investigar como poder comercializar un producto, desde su concepción hasta su culminación, hacer análisis de mercado y llevar un proyecto a la practica de una manera real.

Nuestro agradecimiento a nuestros profesores, compañeros de trabajo, de clases, amigos, que nos ayudaron guiándonos en este trabajo y a todas aquellas personas que poniendo un granito de arena supieron orientarnos y en muchas ocasiones los acosamos solicitándoles información a la que no habíamos accedido anteriormente por nuestro enfoque técnico, una vez mas nuestro agradecimiento para todos ustedes que hicieron posible la terminación de este Proyecto presentado como Trabajo de Tesis para obtener nuestro grado.

Anexo A

PROYECTO:

OFRECER UN SERVICIO DE MENSAJES A CELULARES A TRAVES DE NUESTRO CALL CENTER.

INTRODUCCIÓN AL MERCADO DE UN SERVICIO DE MENSAJERIA DIRIGIDO A PERSONAS QUE SIN POSEER UN EQUIPO CELULAR PUEDAN ENVIAR MENSAJES A LAS PERSONAS QUE SI LO POSEEN LLAMANDO A TRAVES DE UN TELEFONO CONVENCIONAL A NUESTRO CALL CENTER

Plan de Negocio de DATACELULAR

ANALISIS ECONOMICO Y FINANCIERO

1. INVERSION INICIAL

RESUMEN DE INVERSIONES	
Inversión Fija	\$76.010,00
Inversión Diferida	\$13.830,00
Capital de Trabajo	\$150.000,00
TOTAL	\$239.840,00

1.1. INVERSION FIJA

1.1.1 INSTALACIONES

Instalaciones Electricas			
Honorarios profesionales		\$ 500,00	
Materiales Electricos		\$ 1.000,00	
Total Inst.Electricas			\$ 1.500,00
Red de Datos			
Honorarios profesionales		\$ 200,00	
Materiales		\$ 300,00	
Total Red de Datos			\$ 500,00
Adecuaciones Red de datos			
Honorarios profesionales		\$ 200,00	
Materiales Red de Datos		\$ 300,00	
Total adecuaciones red de datos			\$ 500,00
Adecuaciones local			
Pintura		\$ 1.000,00	
Estrucutura oficinas		\$ 2.000,00	
Cortinas		\$ 400,00	
Accesorios decoracion		\$ 200,00	
Accesorios baño		\$ 100,00	
Total adecuaciones			\$ 3.700,00
Instalaciones de oficina			
UPS- 4KVA (4 horas)		\$ 3.500,00	
Central de aire acondicionado		\$ 3.500,00	
Cableado estructurado		\$ 2.000,00	
RACK		\$ 300,00	
Total instalaciones oficina			\$ 9.300,00
TOTAL INSTALACIONES			\$ 15.500,00

1.1.2 EQUIPOS DE OFICINA

EQUIPOS DE OFICINA	Unidades	Valor	Total	
Equipos de computacion				
Computadoras	26	\$ 700	\$ 18.200,00	
Emisor telefonico	15	\$ 1000	\$ 15.000,00	
Impresoras	3	\$ 500	\$ 1.500,00	
Total Equipos de computacion				\$ 34.700,00
Centro de computo				
Servidor-Base de Datos			\$ 3.500,00	
Web Correo			\$ 2.000,00	
Unidad de Cinta			\$ 800,00	
Motor Base de Datos			\$ 5.000,00	
Total Centro de computo				\$ 11.300,00
Equipos de comunicaci3n				
Router			\$ 3.000,00	
Switch			\$ 1.500,00	
Central Telef3nica de 10 l3neas			\$ 4.000,00	
Total Equipos de comunicaci3n				\$ 8.500,00

TOTAL EQUIPOS DE OFICINA	\$ 54.500,00
---------------------------------	---------------------

1.1.3 MUEBLES DE OFICINA

MUEBLES DE OFICINA	Unidades	Valor	Total
Escritorio	6	\$ 200	\$ 1.200,00
M3dulos Call Center	15	\$ 100	\$ 1.500,00
M3dulos 3rea administrativa	3	\$ 100	\$ 300,00
Modulos 3rea t3cnica	3	\$ 100	\$ 300,00
Counter Servicio al Cliente	1	\$ 250	\$ 250,00
Sillas con brazo	6	\$ 30	\$ 180,00
Sillas sin brazo	35	\$ 20	\$ 700,00
Mesas de oficina	2	\$ 50	\$ 100,00
Archivadores	11	\$ 50	\$ 550,00
Telefonos con diademas-Call Center	15	\$ 40	\$ 600,00
Tel3fonos de escritorio	11	\$ 30	\$ 330,00

TOTAL MUEBLES DE OFICINA	\$ 6.010,00
---------------------------------	--------------------

1.2 INVERSION DIFERIDA

Gastos constitucion		
Honorarios abogado	\$ 500,00	
Tramites de constitucion	\$ 300,00	
Formularios	\$ 30,00	
Total Gastos constitucion		\$ 830,00
Marketing y Publicidad		
Campaña de medios	\$ 8.000,00	
Diseño de imagen	\$ 2.500,00	
Papeleria y tarjeteria	\$ 500,00	
Total Marketing y Publicidad		\$ 11.000,00
Asesorias		
Consultoria Legal	\$ 1.500,00	
Consultaria Contable	\$ 500,00	
Total Asesorias		\$ 2.000,00

TOTAL INVERSION DIFERIDA		\$ 13.830,00
---------------------------------	--	---------------------

2 . PROYECCION DE VENTAS

2.1. PARAMETROS PARA EL CALCULO DE INGRESOS

Numero Vendedores	30,00
Suscripcion mensuales por vendedor	60,00
Total Suscrip mensual	1.800,00
Valor Suscrip primera vez	\$ 10,00
Mensual cliente US\$	\$ 5,00

2.2. INGRESOS MENSUALES CLIENTES NUEVOS

	MES 1	MES 2	MES 3	MES 4	MES 5	MES 6	MES 7	MES 8	MES 9	MES 10	MES 11	MES 12
Numero Suscriptores	1.800	3.600	5.400	7.200	9.000	10.800	12.600	14.400	16.200	18.000	19.800	21.600
Suscripcion cliente US\$	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00
Mensualidad US\$	9.000,00	18.000,00	27.000,00	36.000,00	45.000,00	54.000,00	63.000,00	72.000,00	81.000,00	90.000,00	99.000,00	108.000,00
TOTAL US\$	27.000,00	36.000,00	45.000,00	54.000,00	63.000,00	72.000,00	81.000,00	90.000,00	99.000,00	108.000,00	117.000,00	126.000,00

TOTAL ANUAL CLIENTES NUEVOS

TOTAL	
Numero Suscriptores	21.600
Suscripcion cliente US\$	216.000,00
Mensualidad US\$	702.000,00
TOTAL US\$	918.000,00

2.3 PROYECCION ANUAL DE CLIENTES

	AÑO 1	AÑO 2	AÑO3	AÑO 4	AÑO 5
Clientes nuevos	21.600	21.600	21.600	21.600	21.600
Clientes actuales		21.600	43.200	64.800	86.400

TOTAL CLIENTES ANUAL	21.600	43.200	64.800	86.400	108.000
-----------------------------	---------------	---------------	---------------	---------------	----------------

2.4 PROYECCION TOTAL DE INGRESOS

	AÑO 1	AÑO 2	AÑO3	AÑO 4	AÑO 5
Suscripcion US\$	\$ 216.000,00	\$ 216.000,00	\$ 216.000,00	\$ 216.000,00	\$ 216.000,00
Mensualidad clientes nuevos US\$	\$ 702.000,00	\$ 702.000,00	\$ 702.000,00	\$ 702.000,00	\$ 702.000,00
Mensualidad clientes actuales US\$	0,00	\$ 1.296.000,00	\$ 2.592.000,00	\$ 3.888.000,00	\$ 5.184.000,00
TOTAL INGRESOS US\$	\$ 918.000,00	\$ 2.214.000,00	\$ 3.510.000,00	\$ 4.806.000,00	\$ 6.102.000,00

3. COSTOS

3.1 PARAMETROS DE COSTOS (Valores en US\$)

Comision vendedor	\$ 10,00	Mensaje x cliente	50,00	Costo men.operador adicional	320,00
Comision Jefe Vtas	\$ 0,50	Mensajes enviado por operador por minuto	3	Costo computador adicional	700,00
Costo debito bancario	\$ 0,40	Mensajes enviado por operador por hora	180	Costo emisor adicional	1.000,00
Costo envio correspondencia	\$ 0,10	Mensajes enviado por operador por dia	1.440		
Comision operadora	\$ 0,06	Mensaj mensua x opera	36.000		
		Mensajes año	432.000		

3.2 COSTOS DE VENTAS

	MES 1	MES 2	MES 3	MES 4	MES 5	MES 6	MES 7	MES 8	MES 9	MES 10	MES 11	MES 12	TOTAL
Suscriptores	1.800	1.800,00	1.800,00	1.800,00	1.800,00	1.800,00	1.800,00	1.800,00	1.800,00	1.800,00	1.800,00	1.800,00	21.600
Comision vendedor US\$	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	18.000,00	216.000,00
Comision Jefe Vtas US\$	900,00	900,00	900,00	900,00	900,00	900,00	900,00	900,00	900,00	900,00	900,00	900,00	10.800,00
Total Comisiones US\$	18.900,00	18.900,00	18.900,00	18.900,00	18.900,00	18.900,00	18.900,00	18.900,00	18.900,00	18.900,00	18.900,00	18.900,00	226.800,00

3.3 COSTOS DE PRODUCCION

3.3.1 COSTOS DE PRODUCCION CLIENTES NUEVOS

	MES 1	MES 2	MES 3	MES 4	MES 5	MES 6	MES 7	MES 8	MES 9	MES 10	MES 11	MES 12	TOTAL
Suscriptores	1.800	3.600	5.400	7.200	9.000	10.800	12.600	14.400	16.200	18.000	19.800	21.600	21.600
Comision operadora US\$	5.400,00	10.800,00	16.200,00	21.600,00	27.000,00	32.400,00	37.800,00	43.200,00	48.600,00	54.000,00	59.400,00	64.800,00	421.200,00
Debito Bancario US\$	720,00	1.440,00	2.160,00	2.880,00	3.600,00	4.320,00	5.040,00	5.760,00	6.480,00	7.200,00	7.920,00	8.640,00	56.160,00
Envio correspondencia US\$	180,00	360,00	540,00	720,00	900,00	1.080,00	1.260,00	1.440,00	1.620,00	1.800,00	1.980,00	2.160,00	14.040,00

3.3.2 COSTOS DE PRODUCCION ANUALES

3.3.2.1 COMISION OPERADORA

	AÑO 1	AÑO 2	AÑO3	AÑO 4	AÑO 5
Com. Operadora clientes nuevos	\$ 421.200,00	\$ 421.200,00	\$ 421.200,00	\$ 421.200,00	\$ 421.200,00
Com. Operadora clientes actual		\$ 777.600,00	\$ 1.555.200,00	\$ 2.332.800,00	\$ 3.110.400,00
Total Com.Operadora US\$	\$ 421.200,00	\$ 1.198.800,00	\$ 1.976.400,00	\$ 2.754.000,00	\$ 3.531.600,00

3.3.2.2 DEBITOS BANCARIOS Y ENVIO CORRESPONDENCIA

	AÑO 1	AÑO 2	AÑO3	AÑO 4	AÑO 5
Debito clientes nuevos	\$ 56.160,00	\$ 56.160,00	\$ 56.160,00	\$ 56.160,00	\$ 56.160,00
Debito clientes actual	\$ 0,00	\$ 103.680,00	\$ 207.360,00	\$ 311.040,00	\$ 414.720,00
Total deb bancarios	\$ 56.160,00	\$ 159.840,00	\$ 263.520,00	\$ 367.200,00	\$ 470.880,00
Corresponden clientes nuevos	\$ 14.040,00	\$ 14.040,00	\$ 14.040,00	\$ 14.040,00	\$ 14.040,00
Corresponden actuales	\$ 0,00	\$ 25.920,00	\$ 51.840,00	\$ 77.760,00	\$ 103.680,00
Total correspondencia	\$ 14.040,00	\$ 39.960,00	\$ 65.880,00	\$ 91.800,00	\$ 117.720,00

3.3.3 PROYECCION DE MENSAJES

3.3.3.1 MENSAJES MENSUALES ENVIADOS POR CLIENTES NUEVOS

	MES 1	MES 2	MES 3	MES 4	MES 5	MES 6	MES 7	MES 8	MES 9	MES 10	MES 11	MES 12
Suscriptores	1.800	3.600	5.400	7.200	9.000	10.800	12.600	14.400	16.200	18.000	19.800	21.600
Mensajes por cliente	90.000	180.000	270.000	360.000	450.000	540.000	630.000	720.000	810.000	900.000	990.000	1.080.000

3.3.3.2 TOTAL DE MENSAJES ENVIADOS AL AÑO

	AÑO 1	AÑO 2	AÑO3	AÑO 4	AÑO 5
Mensajes enviados por clientes nuevos	1.080.000	1.080.000	1.080.000	1.080.000	1.080.000
Mensajes enviados por clientes actuales		12.960.000	25.920.000	38.880.000	51.840.000
Total mensajes enviados	1.080.000	14.040.000	27.000.000	39.960.000	52.920.000

3.3.4 OPERADORES E INVERSION ADICIONAL ANUAL

	AÑO 1	AÑO 2	AÑO3	AÑO 4	AÑO 5
Operadores nuevos	15	30	30	30	30
Costo adicional operadores	\$ 19.200,00	\$ 81.600,00	\$ 144.000,00	\$ 206.400,00	\$ 268.800,00
	AÑO 1	AÑO 2	AÑO3	AÑO 4	AÑO 5
Equipos nuevos	\$ 10.500,00	\$ 21.000,00	\$ 21.000,00	\$ 21.000,00	\$ 21.000,00
Emisor nuevos	\$ 15.000,00	\$ 30.000,00	\$ 30.000,00	\$ 30.000,00	\$ 30.000,00
Total inversion adicional	\$ 25.500,00	\$ 51.000,00	\$ 51.000,00	\$ 51.000,00	\$ 51.000,00

3.4 SUELDOS Y SALARIOS

PRIMER AÑO

	PRECIO UNITARIO	# DE EMPLEADOS	PRECIO TOTAL	PRECIO ANUAL	13º SUELDO	14º SUELDO	FONDO DE RESERVAS	VACACIONES	APORTE IESS (12,35%)	TOTAL
Gerente General	\$ 700,00	1	\$ 700,00	\$ 8.400,00	\$ 700,00	\$ 160,00		\$ 350,00	\$ 1.037,40	\$ 10.647,40
Gerente Tecnico	\$ 550,00	1	\$ 550,00	\$ 6.600,00	\$ 550,00	\$ 160,00		\$ 275,00	\$ 815,10	\$ 8.400,10
Gerente Operativo	\$ 550,00	1	\$ 550,00	\$ 6.600,00	\$ 550,00	\$ 160,00		\$ 275,00	\$ 815,10	\$ 8.400,10
Jefe Call Center	\$ 400,00	1	\$ 400,00	\$ 4.800,00	\$ 400,00	\$ 160,00		\$ 200,00	\$ 592,80	\$ 6.152,80
Personal Call Center	\$ 250,00	15	\$ 3.750,00	\$ 45.000,00	\$ 3.750,00	\$ 160,00		\$ 1.875,00	\$ 5.557,50	\$ 56.342,50
Personal Administrativo	\$ 250,00	2	\$ 500,00	\$ 6.000,00	\$ 500,00	\$ 160,00		\$ 250,00	\$ 741,00	\$ 7.651,00
Personal Serv. Al Cliente	\$ 250,00	2	\$ 500,00	\$ 6.000,00	\$ 500,00	\$ 160,00		\$ 250,00	\$ 741,00	\$ 7.651,00
Jefe de Ventas	\$ 350,00	1	\$ 350,00	\$ 4.200,00	\$ 350,00	\$ 160,00		\$ 175,00	\$ 518,70	\$ 5.403,70
Ejecutivo Vtas	\$ 150,00	30	\$ 4.500,00	\$ 54.000,00						\$ 54.000,00
Jefe Centro de Computo	\$ 400,00	1	\$ 400,00	\$ 4.800,00	\$ 400,00	\$ 160,00		\$ 200,00	\$ 592,80	\$ 6.152,80
Personal tecnico	\$ 250,00	2	\$ 500,00	\$ 6.000,00	\$ 500,00	\$ 160,00		\$ 250,00	\$ 741,00	\$ 7.651,00
TOTAL ANUAL										\$ 178.452,40

SEGUNDO AÑO

	PRECIO UNITARIO	# DE EMPLEADOS	PRECIO TOTAL	PRECIO ANUAL	13º SUELDO	14º SUELDO	FONDO DE RESERVAS	VACACIONES	APORTE IESS (12,35%)	TOTAL
Gerente General	\$ 700,00	1	\$ 700,00	\$ 8.400,00	\$ 700,00	\$ 160,00	\$ 700,00	\$ 350,00	\$ 1.037,40	\$ 11.347,40
Gerente Tecnico	\$ 550,00	1	\$ 550,00	\$ 6.600,00	\$ 550,00	\$ 160,00	\$ 550,00	\$ 275,00	\$ 815,10	\$ 8.950,10
Gerente Operativo	\$ 550,00	1	\$ 550,00	\$ 6.600,00	\$ 550,00	\$ 160,00	\$ 550,00	\$ 275,00	\$ 815,10	\$ 8.950,10
Jefe Call Center	\$ 400,00	1	\$ 400,00	\$ 4.800,00	\$ 400,00	\$ 160,00	\$ 400,00	\$ 200,00	\$ 592,80	\$ 6.552,80
Personal Call Center	\$ 250,00	15	\$ 3.750,00	\$ 45.000,00	\$ 3.750,00	\$ 160,00	\$ 3.750,00	\$ 1.875,00	\$ 5.557,50	\$ 60.092,50
Personal Administrativo	\$ 250,00	2	\$ 500,00	\$ 6.000,00	\$ 500,00	\$ 160,00	\$ 500,00	\$ 250,00	\$ 741,00	\$ 8.151,00
Personal Serv. Al Cliente	\$ 250,00	2	\$ 500,00	\$ 6.000,00	\$ 500,00	\$ 160,00	\$ 500,00	\$ 250,00	\$ 741,00	\$ 8.151,00
Jefe de Ventas	\$ 350,00	1	\$ 350,00	\$ 4.200,00	\$ 350,00	\$ 160,00	\$ 350,00	\$ 175,00	\$ 518,70	\$ 5.403,70
Ejecutivo Vtas	\$ 150,00	30	\$ 4.500,00	\$ 54.000,00						\$ 54.000,00
Jefe Centro de Computo	\$ 400,00	1	\$ 400,00	\$ 4.800,00	\$ 400,00	\$ 160,00	\$ 400,00	\$ 200,00	\$ 592,80	\$ 6.552,80
Personal tecnico	\$ 250,00	2	\$ 500,00	\$ 6.000,00	\$ 500,00	\$ 160,00	\$ 500,00	\$ 250,00	\$ 741,00	\$ 8.151,00
TOTAL ANUAL										\$ 186.302,40

TERCER AÑO

	PRECIO UNITARIO	# DE EMPLEADOS	PRECIO TOTAL	PRECIO ANUAL	13º SUELDO	14º SUELDO	FONDO DE RESERVAS	VACACIONES	APORTE IESS (12,35%)	TOTAL
Gerente General	\$ 800,00	1	\$800,00	\$ 9.600,00	\$ 800,00	\$ 160,00	\$ 800,00	\$ 400,00	\$ 1.185,60	\$ 12.945,60
Gerente Tecnico	\$ 650,00	1	\$ 650,00	\$ 7.800,00	\$ 650,00	\$ 160,00	\$ 650,00	\$ 325,00	\$ 963,30	\$ 10.548,30
Gerente Operativo	\$ 650,00	1	\$ 650,00	\$ 7.800,00	\$ 650,00	\$ 160,00	\$ 650,00	\$ 325,00	\$ 963,30	\$ 10.548,30
Jefe Call Center	\$ 500,00	2	\$ 1.000,00	\$ 12.000,00	\$ 1.000,00	\$ 160,00	\$ 1.000,00	\$ 500,00	\$ 1.482,00	\$ 16.142,00
Personal Call Center	\$ 350,00	15	\$ 5.250,00	\$ 63.000,00	\$ 5.250,00	\$ 160,00	\$ 5.250,00	\$ 2.625,00	\$ 7.780,50	\$ 84.065,50
Personal Administrativo	\$ 350,00	2	\$ 700,00	\$ 8.400,00	\$ 700,00	\$ 160,00	\$ 700,00	\$ 350,00	\$ 1.037,40	\$ 11.347,40
Personal Serv. Al Cliente	\$ 350,00	2	\$ 700,00	\$ 8.400,00	\$ 700,00	\$ 160,00	\$ 700,00	\$ 350,00	\$ 1.037,40	\$ 11.347,40
Jefe de Ventas	\$ 400,00	1	\$ 400,00	\$ 4.800,00	\$ 400,00	\$ 160,00	\$ 400,00	\$ 200,00	\$ 592,80	\$ 6.152,80
Ejecutivo Vtas	\$ 175,00	30	\$ 5.250,00	\$ 63.000,00						\$ 63.000,00
Jefe Centro de Computo	\$ 500,00	1	\$ 500,00	\$ 6.000,00	\$ 500,00	\$ 160,00	\$ 500,00	\$ 250,00	\$ 741,00	\$ 8.151,00
Personal tecnico	\$ 350,00	2	\$ 700,00	\$ 8.400,00	\$ 700,00	\$ 160,00	\$ 700,00	\$ 350,00	\$ 1.037,40	\$ 11.347,40
TOTAL ANUAL										\$ 245.595,70

3.5 DEPRECIACION

Concepto	Inversión Inicial	Tasa de Deprec. Anual(%)	1	2	3	4	5
Activos Fijos							
Instalaciones	\$ 15.500,00	10%	\$ 1.550,00	\$ 1.550,00	\$ 1.550,00	\$ 1.550,00	\$ 1.550,00
Muebles y enseres	\$ 6.010,00	10%	\$ 601,00	\$ 601,00	\$ 601,00	\$ 601,00	\$ 601,00
Computadoras	\$ 54.500,00	33%	\$ 18.166,65	\$ 18.166,65	\$ 18.166,65	\$ 0,05	
Total Depreciacion	\$ 76.010,00		\$ 20.317,65	\$ 20.317,65	\$ 20.317,65	\$ 2.151,05	\$ 2.151,00

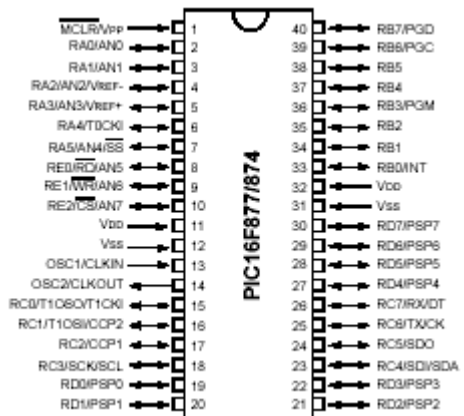
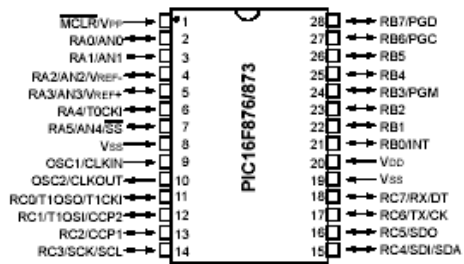
4. FLUJO DE CAJA

AÑOS	0	1	2	3	4	5
INGRESOS						
Suscripciones clientes		\$ 216.000	\$ 216.000	\$ 216.000	\$ 216.000	\$ 216.000
Mens. Nuevos clientes		\$ 702.000	\$ 702.000	\$ 702.000	\$ 702.000	\$ 702.000
Mens. clientes actuales			\$ 1.296.000	\$ 2.592.000	\$ 3.888.000	\$ 5.184.000
Total ingresos		\$ 918.000	\$ 2.214.000	\$ 3.510.000	\$ 4.806.000	\$ 6.102.000
EGRESOS						
Gtos de Administrac.						
Sueldos		\$ -178.452	\$ -186.302	\$ -245.596	\$ -245.596	\$ -245.596
Operadores adic		\$ -19.200	\$ -81.600	\$ -144.000	\$ -206.400	\$ -268.800
Ases. Contable		\$ -2.400	\$ -2.640	\$ -2.880	\$ -2.880	\$ -2.880
Servicios Básicos		\$ -13.200	\$ -15.360	\$ -17.400	\$ -17.400	\$ -17.400
Mantenimiento						
Comunicaciones		\$ -1.200	\$ -1.440	\$ -1.680	\$ -1.680	\$ -1.680
Suministros		\$ -1.800	\$ -2.040	\$ -2.400	\$ -2.400	\$ -2.400
Arriendo		\$ -12.000	\$ -13.200	\$ -14.400	\$ -14.400	\$ -14.400
Seguro		\$ -1.000	\$ -1.000	\$ -1.000	\$ -1.000	\$ -1.000
Otros		\$ -2.400	\$ -3.000	\$ -3.600	\$ -3.600	\$ -3.600
Total Gastos Adm.		\$ -231.652	\$ -306.582	\$ -432.956	\$ -495.356	\$ -557.756
Gastos Producción						
Comision Operadora		\$ -421.200	\$ -1.198.800	\$ -1.976.400	\$ -2.754.000	\$ -3.531.600
Debitos bancarios		\$ -56.160	\$ -159.840	\$ -263.520	\$ -367.200	\$ -470.880
Envio correspondencia		\$ 14.040	\$ 39.960	\$ 65.880	\$ 91.800	\$ 117.720
Total Gasto Producc.		\$ -463.320	\$ -1.318.680	\$ -2.174.040	\$ -3.029.400	\$ -3.884.760
Gastos de Ventas						
Comisiones ventas		\$ -226.800	\$ -226.800	\$ -226.800	\$ -226.800	\$ -226.800
Publicidad		\$ -4.800	\$ -6.000	\$ -7.200	\$ -7.200	\$ -7.200
Total Gasto Ventas		\$ -231.600	\$ -232.800	\$ -234.000	\$ -234.000	\$ -234.000
SUBTOTAL		\$ - 8.572	\$ 355.938	\$ 669.004	\$ 1.047.244	\$ 1.425.484
Depreciación		\$ - 20.318	\$ - 20.318	\$ - 20.318	\$ - 2.151	\$ - 2.151
Ut. antes Impuesto		\$ - 28.890	\$ 335.620	\$ 648.687	\$ 1.045.093	\$ 1.423.333
Participación Empleados			\$ - 50.343	\$ - 97.303	\$ - 156.764	\$ - 213.500
Impuesto a la Renta			\$ - 71.319	\$ - 137.846	\$ - 222.082	\$ - 302.458
Ut. despues impuest		\$ - 28.890	\$ 213.958	\$ 413.538	\$ 666.247	\$ 907.375
Depreciación		\$ 20.318	\$ 20.318	\$ 20.318	\$ 2.151	\$ 2.151
SubtotalUtil +Deprec.		\$ - 8.572	\$ 234.275	\$ 433.855	\$ 668.398	\$ 909.526
Inversion Fija	\$ -76.010,00					
Inversión Diferida	\$ -13.830,00					
Capital de Trabajo	\$ -150.000,00					
Inversión Equipos		\$ -25.500	\$ -51.000	\$ -51.000	\$ -51.000	\$ -51.000
FLUJO DE CAJA	\$ -239.840,00	\$ -34.072,40	\$ 183.275,37	\$ 382.855,39	\$ 617.398,00	\$ 858.525,98

Anexo B

1. CARACTERISTICAS DE LOS MICROCONTROLADORES PIC16F876

Y PIC16F877

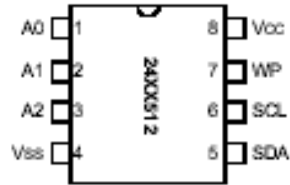


1.1. Características

- Alto rendimiento en el CPU tipo RISC
- Es programado con 35 instrucciones solamente
- Velocidad de operación: reloj de entrada de 20 MHz y 200 ns por ciclo de instrucción
- Memoria FLASH sobre los 8K x 14 palabras
- Memoria RAM sobre los 368 x 8 bytes
- Memoria EEPROM sobre los 256 x 8 bytes
- Distribución de pines compatibles con los PIC16C73B/74B/76/77
- Interrupciones habilitadas (sobre las 14 fuentes)
- Profundidad de pila creada por hardware de 8 niveles
- Modos de direccionamiento directo, indirecto y relativo
- Power-on Reset (POR)
- Power-up Timer (PWRT) and
- Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC
- Protección de código programable
- Ahorrador de energía en modo SLEEP
- Bajo consumo de energía y alta velocidad de las memorias FLASH / EEPROM
- Diseño completamente estático
- Permite al procesador leer y escribir en las memorias
- Amplio rango de voltaje soportado: 2.0V to 5.5V

- Alta fuente de corriente manejada: 25 mA
- Bajo consume de energía:
 - < 0.6 mA típico a 3V, 4 MHz
 - 20 uA típico a 3V, 32 kHz
 - < 1 uA típico cuando no se utiliza al microcontrolador
- Timer0: 8-bit timer/counter con 8-bit pre escalables
- Timer1: 16-bit timer/counter with pre escalables, pueden ser incrementados durante el SLEEP gracias a un reloj / crystal externo
- Timer2: 8-bit timer/counter con 8-bit de registro, pre escalable y post escalable.
- Dos módulos capturadores, comparadores , PWM
 - La captura es 16-bit, max. resolucion es 12.5 ns
 - La comparación es 16-bit, max. resolución es 200 ns
 - PWM max. resolución es 10-bit
- Canal convertidor Analógico a Digital de 10-bit
- Synchronous Serial Port (SSP) con SPI (modo m
- I²C (Maestro / Esclavo)
- Receptor Universal Sincrónico Asincrónico
- Transmisor (USART/SCI) con 9-bit para direccionamiento
- Parallel Slave Port (PSP) 8-bits, con pines externos para control de RD, WR y CS (sólo los pines 40 / 44)

1. CARACTERÍSTICAS DE LAS MEMORIAS 24LC512



1.1. Características

- Bajo consumo de corriente en la tecnología CMOS
 - Máxima corriente en la escritura 5 mA a 5.5V
 - Máxima corriente en la lectura 400 μ A a 5.5V
 - Corriente en el circuito sin uso 100 nA a 5.5V
- Dos interfases de buses seriales, I²C™ compatibles
- Pueden ser unidas en cascada hasta 8 dispositivos
- 5 ms max. en cada ciclo de escritura
- protector contra escritura desarrollado en el hardware para el arreglo de memorias
- Entradas Schmitt Trigger para suppression de ruidos
- Permite hasta 1,000,000 de ciclos de lectura / escritura
- Protección a una carga electrostática > 4000V

- Retención de datos > 200 años
- Rango de Temperaturas:
 - Industrial (I): -40°C to +85°C
 - Automotiva (E): -40°C to +125°C

1.2. Descripción de los pines

1.2.1. A0, A1, A2 Entradas de direccionamiento del Chip

Las entradas A0, A1, A2 son usadas por el 24XX512 para múltiples operaciones con los dispositivos. Los niveles lógicos de estas entradas son comparados con los correspondientes bits en la dirección esclavo. El integrado selecciona si la comparación es verdadera.

Hasta 8 dispositivos pueden ser conectados en el mismo bus usando diferentes combinaciones en el Chip Select.

1.2.2. Serial Data (SDA)

Este es un PIN bidireccional usado para transferir direcciones y datos de entrada y salida del dispositivo.

Para transferencia normal de datos, SDA debe de ser cambiado sólo cuando SCL tiene un bajo. Cambios ocurridos en SDA cuando SCL esta en alto sirve sólo para indicar condiciones de inicio y fin de la transferencia.

1.2.3. Serial Clock (SCL)

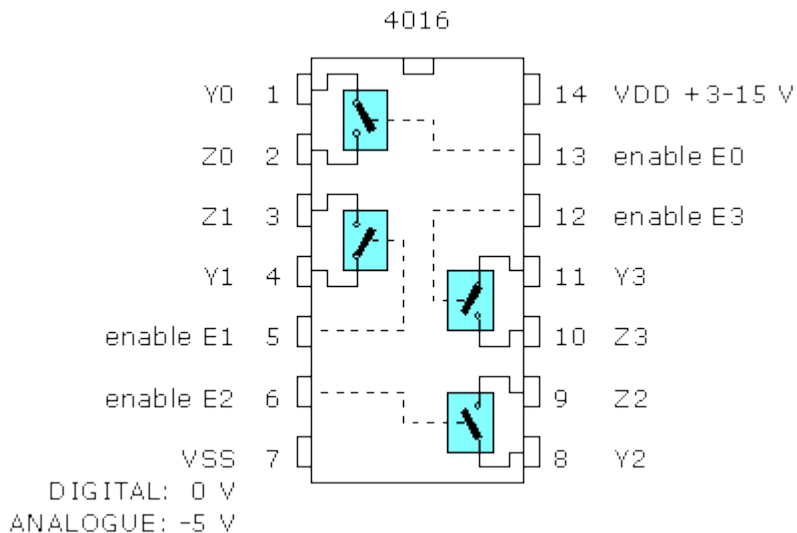
Esta entrada es usada para sincronizar la transferencia de datos desde y hacia el dispositivo.

1.2.4. Write-Protect (WP)

Este PIN puede ser conectado tanto a VSS como a VCC. Si se lo coloca a VSS la operación de la memoria es normal, en cambio si se lo coloca a VCC las operaciones de escrituras están deshabilitadas mientras que las operaciones de lectura pueden seguir realizándose.

1.- Características del Switch 4016

Pin connections



The 4016 contains four switches. Each switch has two input/output terminals, X and Y and an enable terminal, E. When E is LOW, the switch is open. When E is HIGH, the switch is closed.

Y0-Y3, Z0-Z3: input/output terminals

What is a bilateral switch?

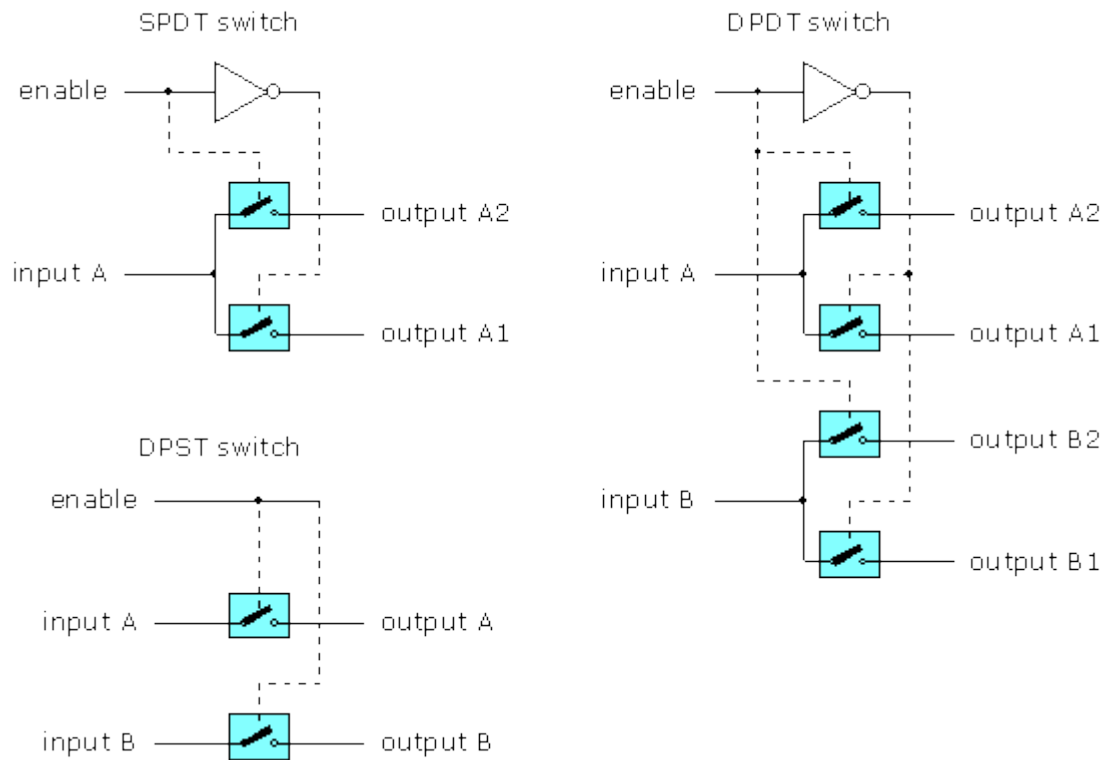
A bilateral switch uses a special CMOS circuit called a **transmission gate**. The circuit behaves as a SPST (single pole/single throw) switch which is under electronic control. When the control signal, or enable, is HIGH, the switch is closed, allowing signals to be transferred between the switch terminals. The switch is 'bilateral' because either terminal can be used as the input. In other words, current flow can be in either direction.

The ON resistance of 4016 switches is around 300 Ω .

In *digital* mode, logic 0 and logic 1 signals are transferred by the 4016 switches. In this mode, the action of the switches can resemble logic gates. For example, if X is used as a switch input, when X=1 AND E=1, output Y=1. This is just like the action of an AND gate.

In *analogue* mode varying signals, such as audio signals are transferred. This opens up interesting methods for the electronic control of signal transfer. The analogue signal must be kept within the limits set by the power supply to the 4016. If the analogue signal is centred around 0 V, the pin 7 voltage of the 4016 should be connected to a negative voltage, say -5 V, with the pin 14 positive voltage connected to +5 V. Alternatively, a DC voltage can be added to the audio signal so that it remains within the limits of the normal power supply.

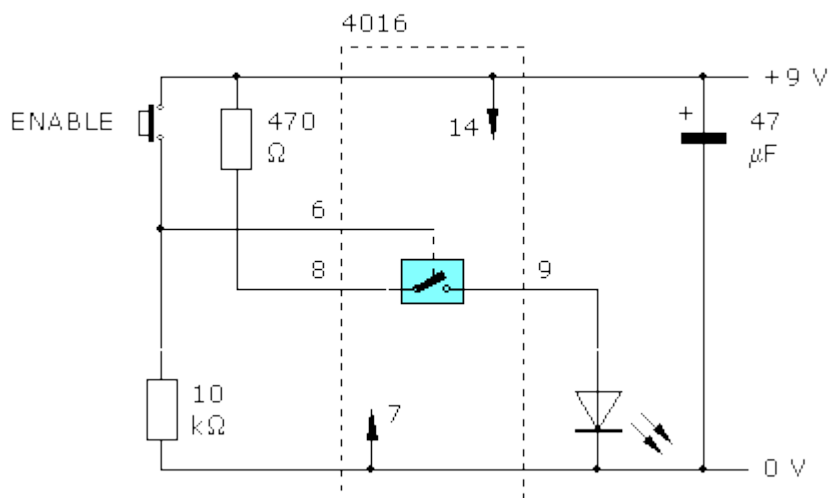
Switches can be linked together to provide SPDT (single pole/double throw), DPST (double pole/single throw) and DPDT (double pole/double throw) action:



For SPDT and DPDT action, a NOT gate is needed. A transistor NOT gate or a spare gate in another integrated circuit can be used. When enable is LOW input A is connected to output A1. When enable is HIGH, input A is connected to output A2.

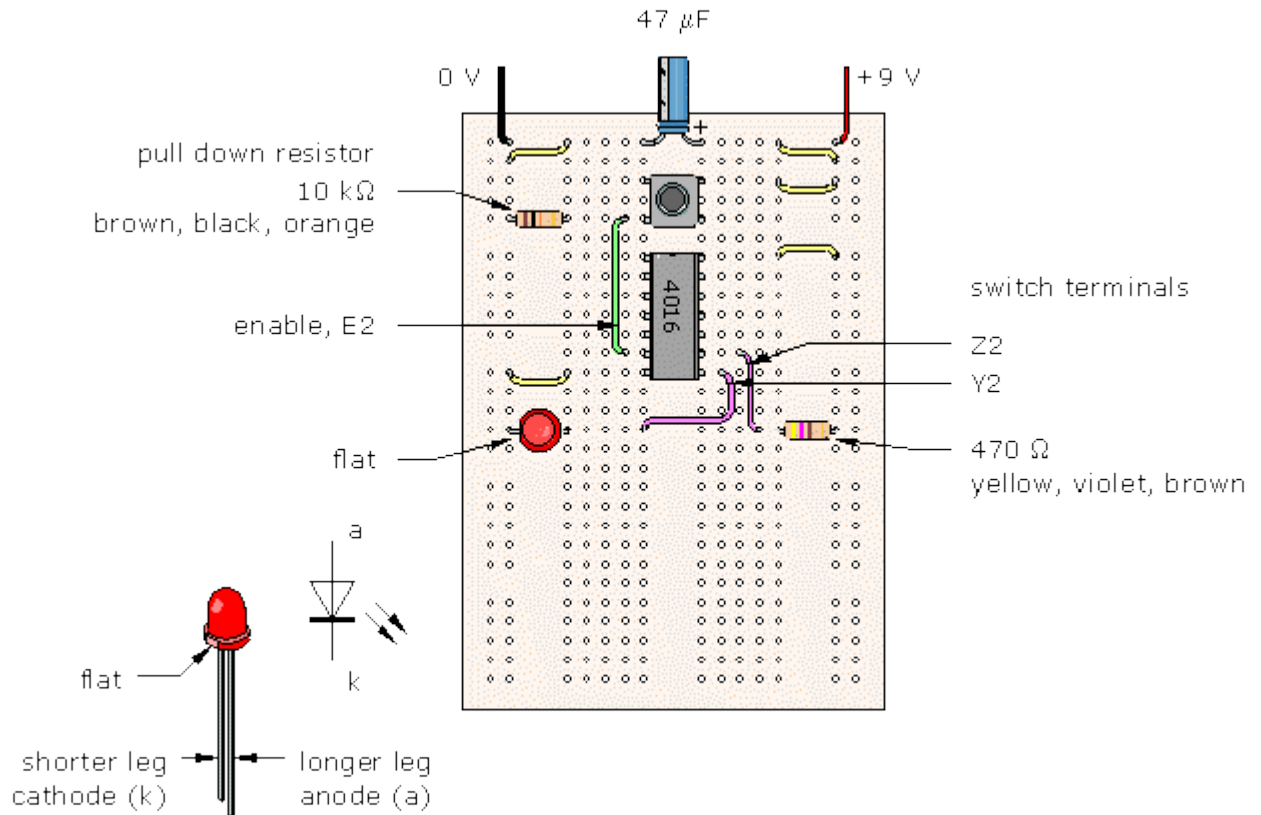
Basic operation

It is easy to test the operation of one of the switches inside the 4016:



The prototype board layout below shows how to build this circuit:

4016



When you press the switch, the LED should illuminate. Once you have made the circuit work, swap the positions of the link wires to Y2 and Z2 and confirm that current can flow in either direction through the switch.

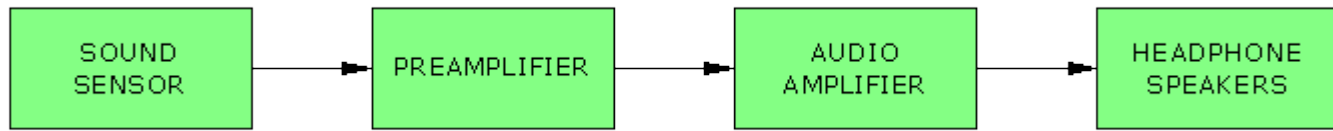
Try modifying the circuit to make SPDT, DPST and DPDT switches.

Applications

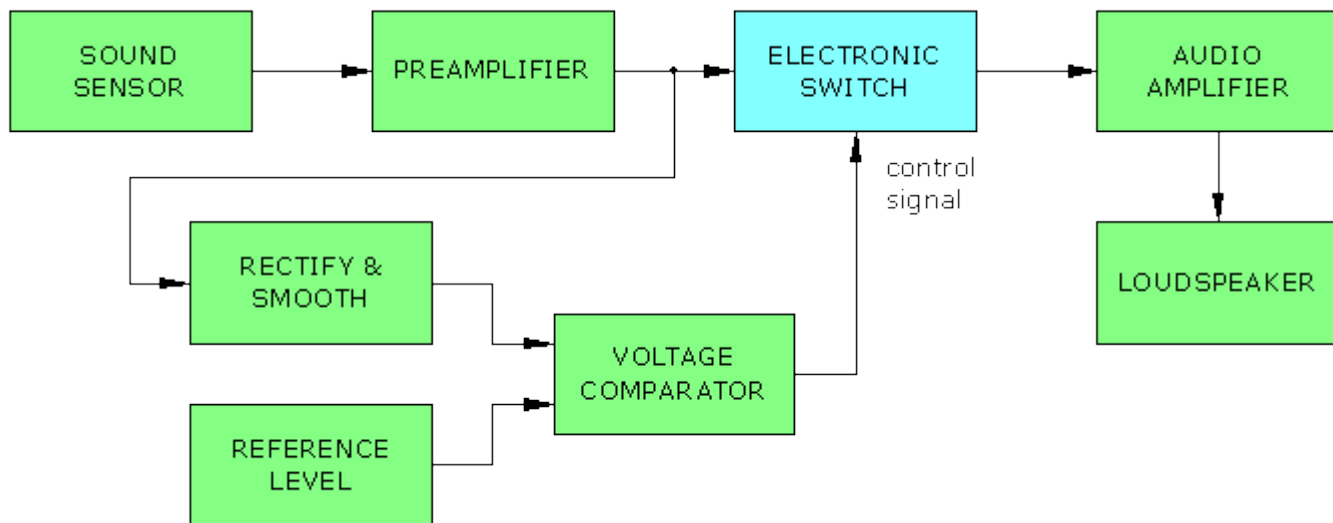
The more interesting applications of the 4016 take advantage of the fact that analogue signals, like audio signals, can be switched ON and OFF.

One idea for a **hearing protection system** uses a 4016 switch to 'disconnect' signals to headphones when loud noises occur. A system like this could be useful in industry where workers are exposed to high noise levels, such as pneumatic drills, or for shooting, to protect against the sudden loud noise of a rifle shot. At other times, low level audio signals are amplified and can be heard in the headphones. People can talk to each other without removing the headphones.

The block diagram for a typical audio system includes a sound sensor, preamplifier, audio amplifier (power amplifier) and the headphone speakers:

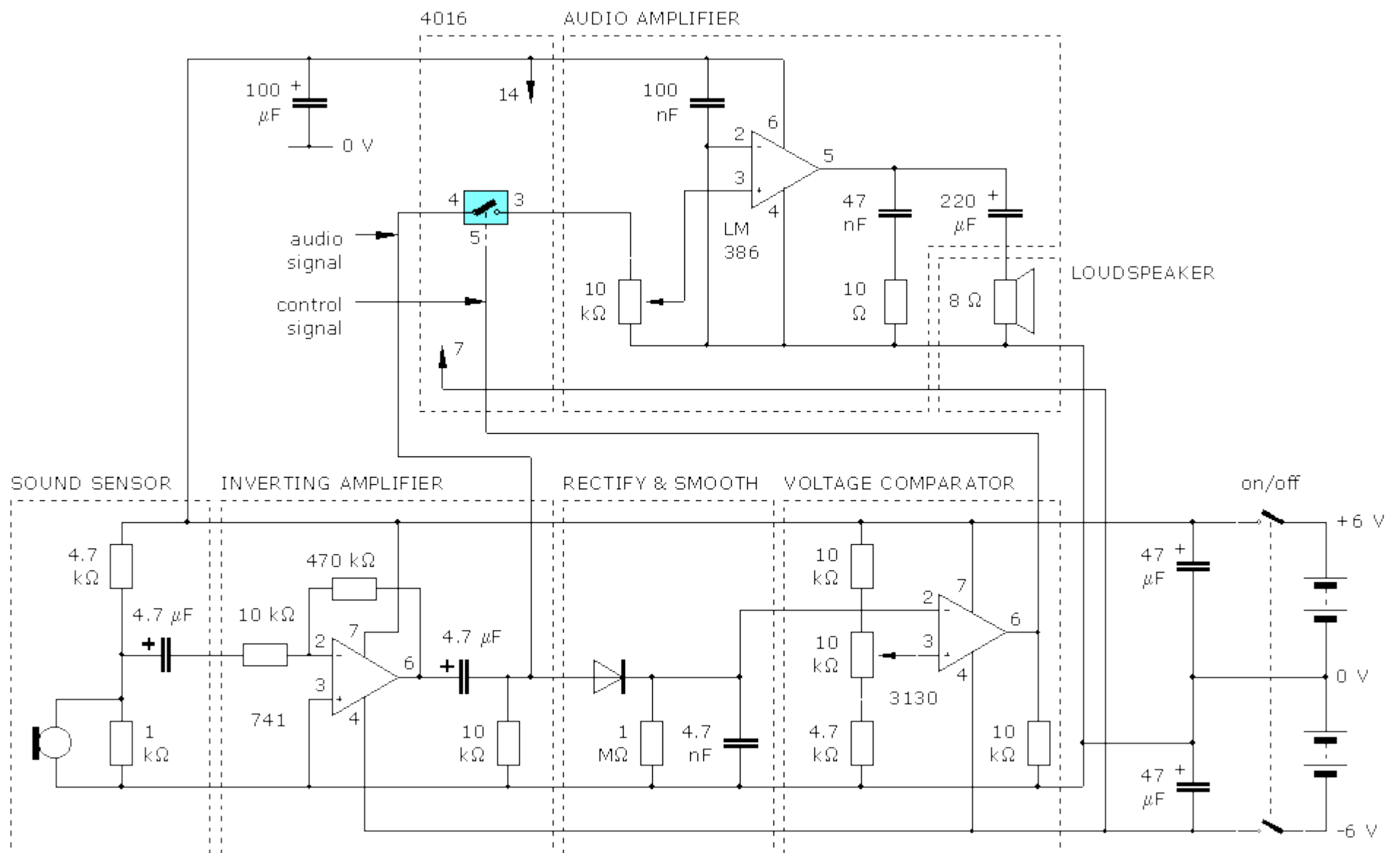


To give electronic control of maximum noise levels, an electronic switch is inserted between the preamplifier and audio amplifier stages:



Audio signals from the preamplifier are rectified and smoothed. This gives a DC level which rises and falls in line with the maximum amplitude of the audio signal. When a loud noise occurs, the DC level rises. If this exceeds the threshold level set by the reference level input to the voltage comparator, the electronic switch is switched OFF, preventing signals from reaching the audio amplifier.

This idea was developed through several prototype board stages and was found to work quite well in practice. Here is the circuit diagram:



HEARING PROTECTION SYSTEM CIRCUIT DIAGRAM

A 741 op-amp is suitable for the inverting amplifier stage but a 3130 op-amp is a better choice for the voltage comparator stage. The 3130 has a faster 'slew rate', meaning that its output voltage changes more quickly. In other words, the 3130 reacts more quickly to cut off potentially damaging sounds when the threshold level is exceeded.

The potentiometer connected to the non-inverting input of the 3130 is adjusted to give an appropriate threshold sound level. This is determined by trial and error while listening in the headphones.

Anexo C

Resultado de las Entrevistas y Encuestas

DATACELULAR S. A.

Encuesta:

Buenas dias/tardes señor/ita, tendría usted la bondad de concederme cinco minutos para contestarme una encuesta, mi nombre es Juan Pérez, trabajo para compañía DATACULAR S.A. estamos investigando acerca de la demanda que tendría un nuevo servicio de mensajería que ofrece la empresa en el mercado local.

Estimamos que su respuesta será de mucha utilidad para la investigación que estamos realizando, le agradezco de antemano.

ENCUESTA

DATACELULAR S.A.

Encuesta para usuarios finales

Esta encuesta está orientada a conocer el uso de un teléfono celular para el envío de mensajes de texto, que ayude a identificar el porcentaje de personas que utilizan este servicio. Además, se desea conocer si a estos usuarios les gustaría tener mejores alternativas al recibir mensajes de texto provenientes de personas que no posean un teléfono celular. Con esta alternativa, la persona puede llamar a un teléfono convencional y solicitar que le envíen un mensaje de texto suyo hacia un teléfono celular específico, teniendo que pagar para el efecto un cierto valor por el servicio recibido.

Esta encuesta está dirigida a personas que posean el servicio celular, así como a aquellas que no tengan dicho servicio.

Datos generales del encuestado

Nombre (opcional):

Edad:

Sexo: Femenino Masculino

• **Ocupación**

- Estudiante
- Trabajador
- Ejecutivo
- Empresario
- Ejecutiva del hogar
- Jubilado
- Otros

1.- ¿En la actualidad, posee usted un equipo celular?

SI

NO

Compañía

.....

2.- ¿Con qué frecuencia utiliza usted el servicio celular?

Nunca Pocas veces Siempre

3.- ¿Con qué frecuencia utiliza usted el servicio de mensajes escritos a través de su teléfono celular?

Nunca Pocas veces Siempre

4.- ¿Cuántas personas que usted conoce poseen un teléfono celular?

Menos de 5 Entre 6 y 10 Entre 11 y 15 Más de 15

5.- ¿Usaría usted el servicio de llamar por un teléfono convencional para enviar mensajes de texto?

SI NO

6.- ¿Le gustaría recibir mensajes de personas que no posean un teléfono celular?

SI NO

7.- ¿Si usted tuviera que pagar por el servicio de recibir mensajes escritos de personas que no posean un teléfono celular, ¿cuánto dinero estaría usted dispuesto a pagar por este servicio?

\$ 3 (mensual) \$ 4 (mensual) \$ 5 (mensual) Costo x mensaje

8.- ¿Le gustaría a usted recibir e-mail de personas que no posean computador y que lo puedan hacer llamando desde un teléfono convencional?

SI NO

Si su respuesta es sí, ¿cuánto dinero estaría usted dispuesto a pagar por este servicio?

\$ 3 (mensual) \$ 4 (mensual) \$ 5 (mensual) Costo x mensaje

DATACELULAR S.A.

ENCUESTA PARA USUARIOS FINALES

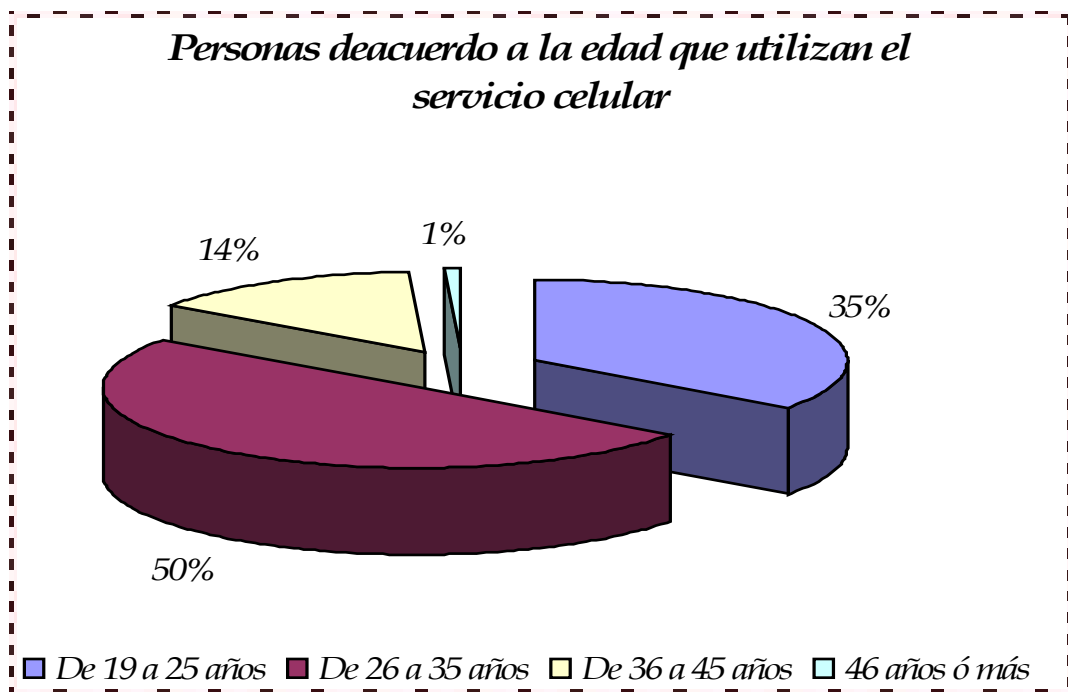
Visualización General (Grafica) :

DATOS GENERALES DE LOS ENCUESTADOS:

Número de Encuesta: 120

Edad:

De 19 a 25 años	De 26 a 35 años	De 36 a 45 años	46 años ó más
42	60	17	1

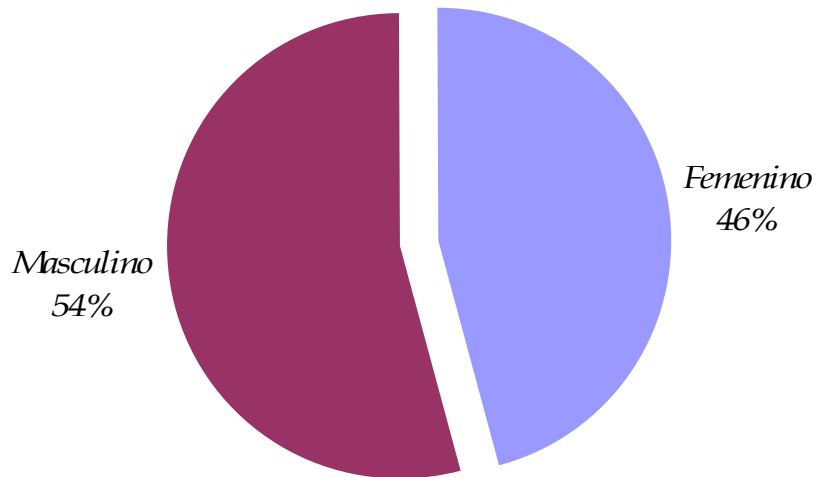


Sexo:

Femenino	Masculino
55	65

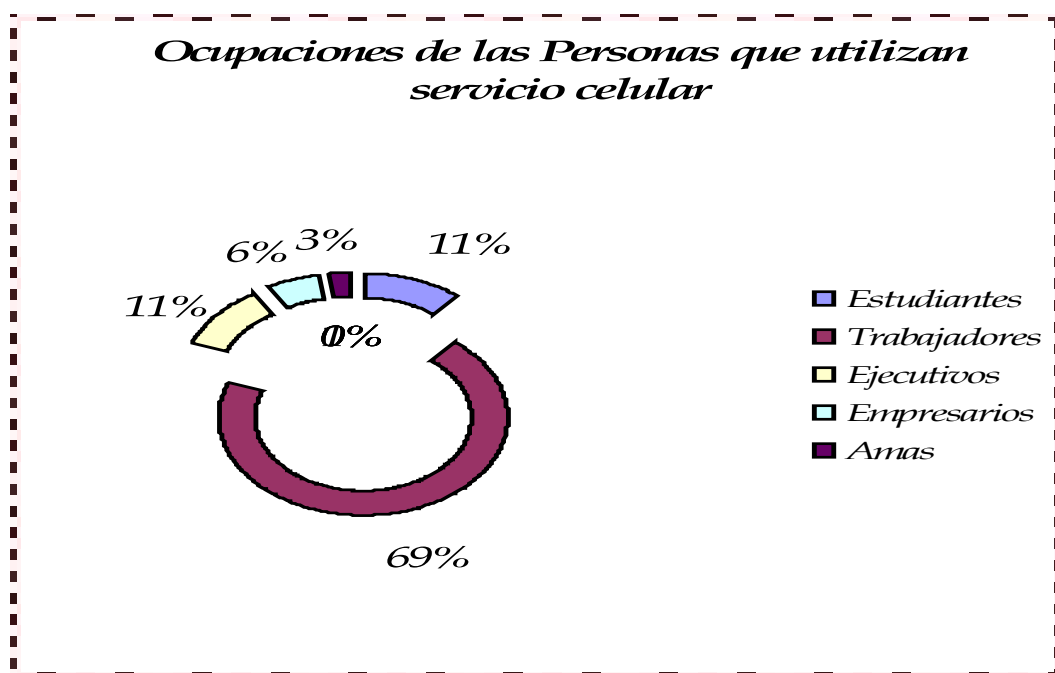
120

Personas de acuerdo al sexo utilizan el servicio celular

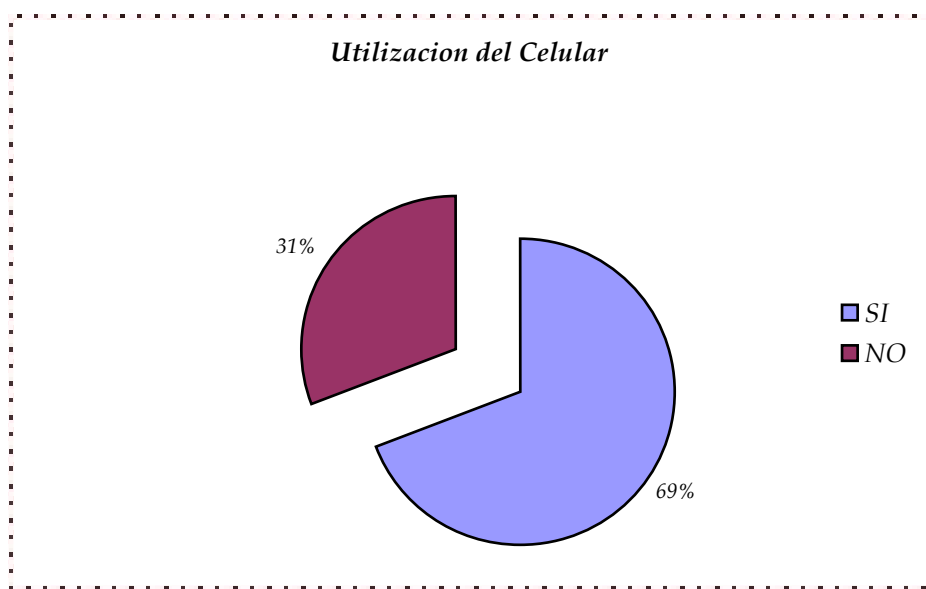


Ocupación:

Estudiantes	Trabajadores	Ejecutivos	Empresarios	Amas de Hogar
13	84	13	7	3

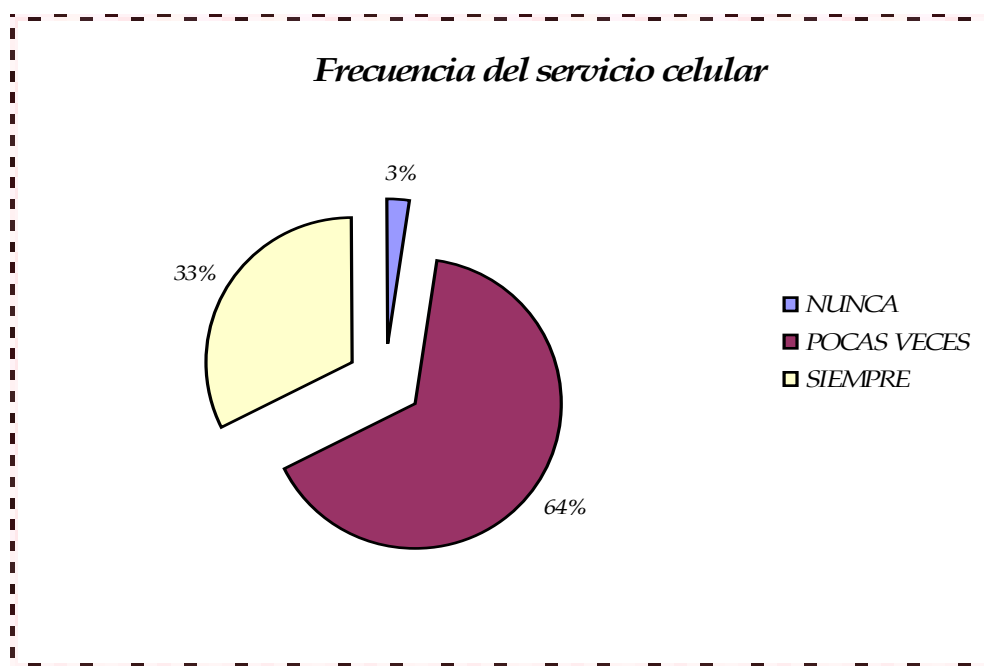


1.- ¿En la actualidad, posee usted un equipo celular?



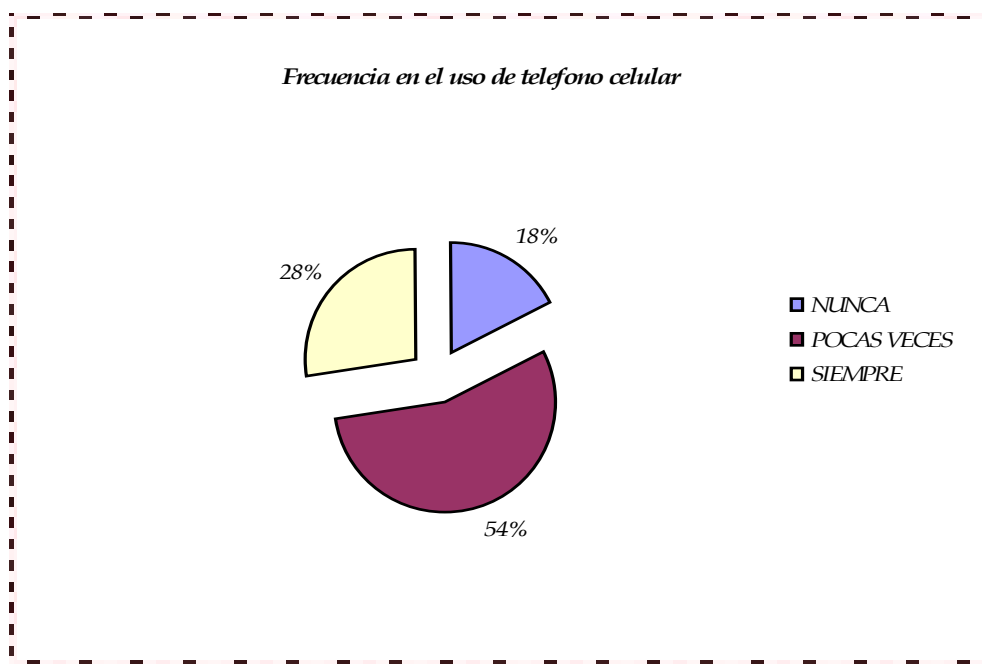
2.- ¿Con qué frecuencia utiliza usted el servicio celular?

NUNCA	POCAS VECES	SIEMPRE
3	78	39



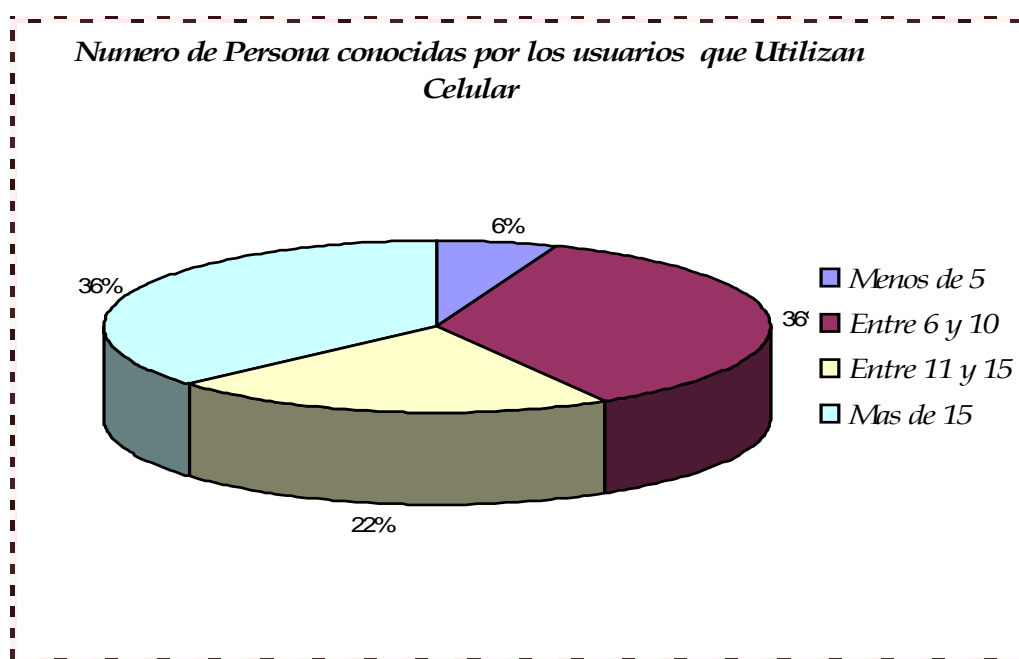
3.- ¿Con qué frecuencia utiliza usted el servicio de mensajes escritos a través de su teléfono celular?

NUNCA	POCAS VECES	SIEMPRE
21	66	33



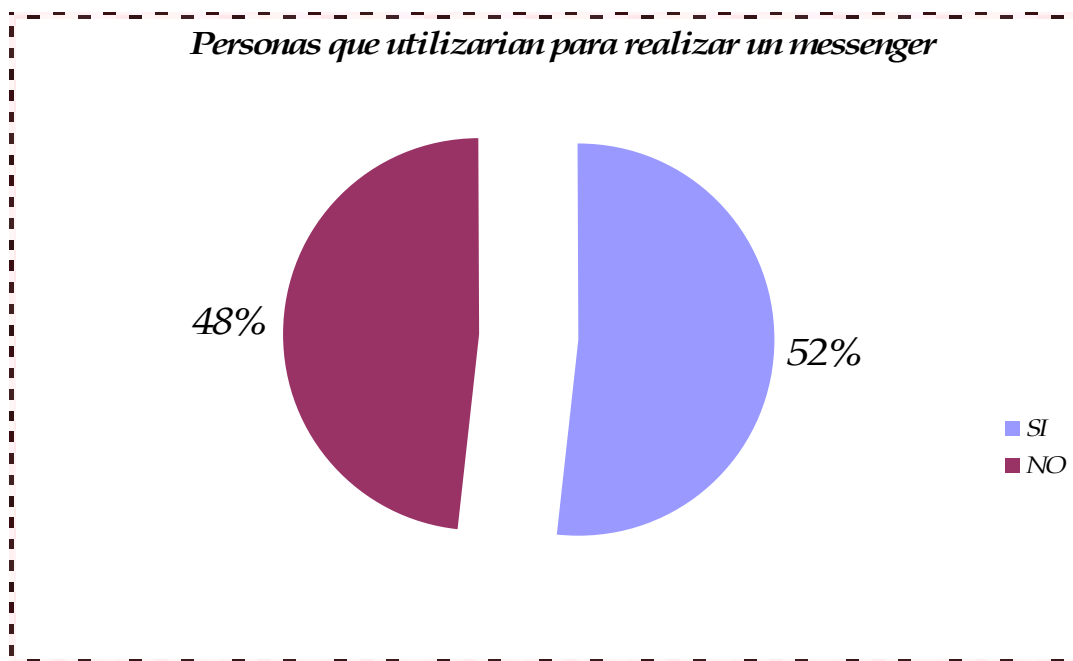
4.- ¿Cuántas personas conocidas suya poseen un teléfono celular?

Menos de 5	Entre 6 y 10	Entre 11 y 15	Mas de 15
7	43	26	44



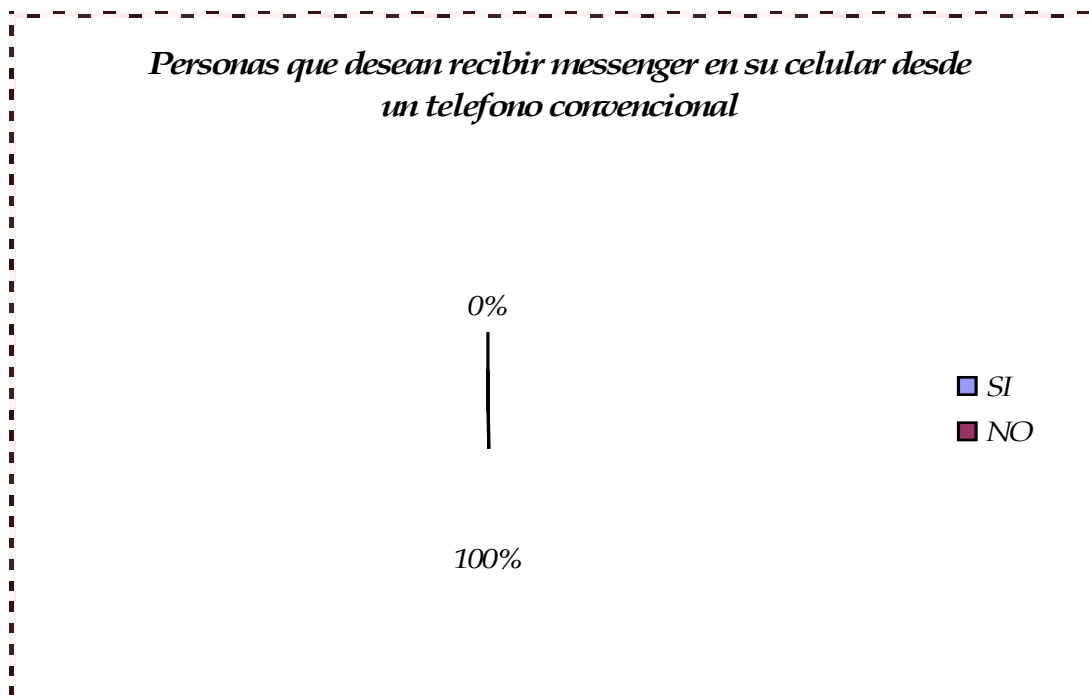
5.- ¿Usaría usted el servicio de llamar por un teléfono convencional para enviar mensajes de texto?

SI	NO
62	58



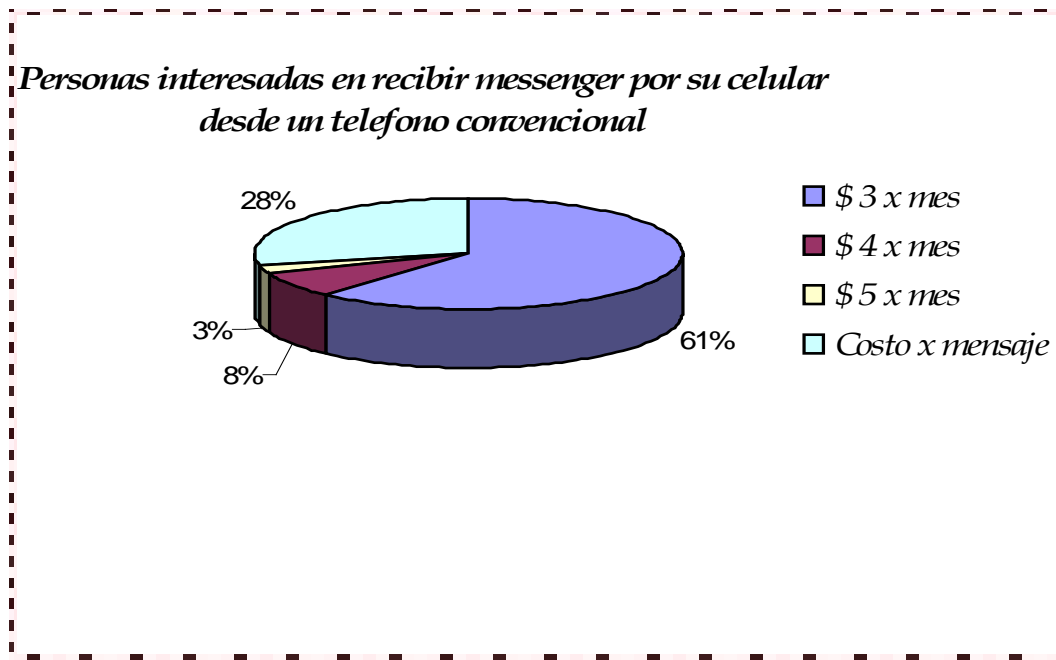
6.- ¿Le gustaría recibir mensajes de personas que no posean un teléfono celular?

SI	NO
120	0



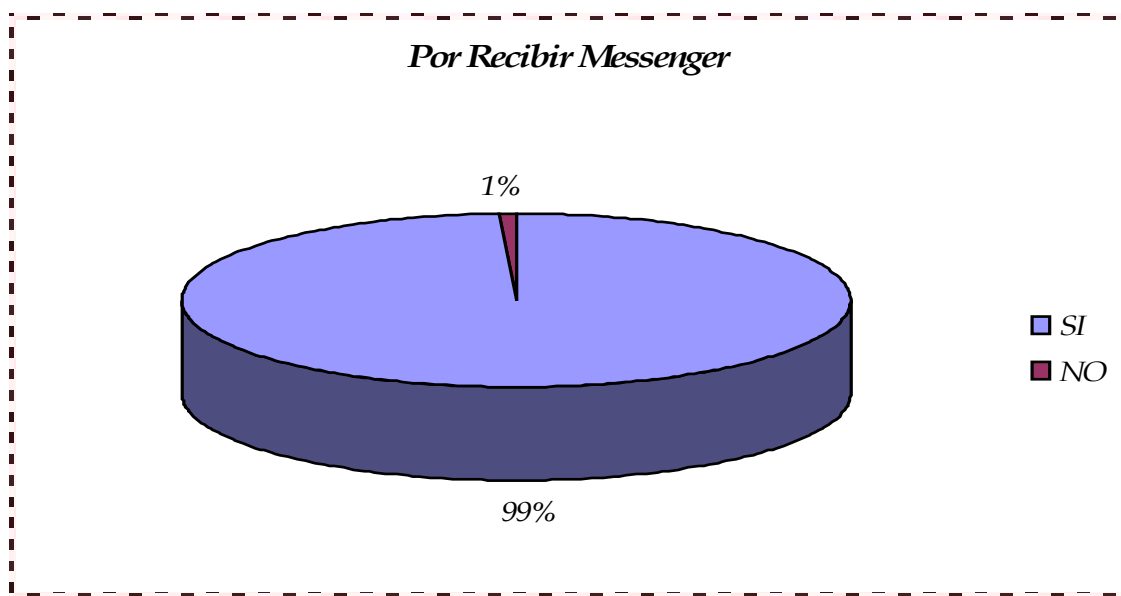
7.- ¿Si usted tuviera que pagar por el servicio de recibir mensajes escritos de personas que no posean un teléfono celular, ¿cuánto dinero estaría usted dispuesto a pagar por este servicio?

\$ 3 x mes	\$ 4 x mes	\$ 5 x mes	Costo x mensaje
74	9	3	34



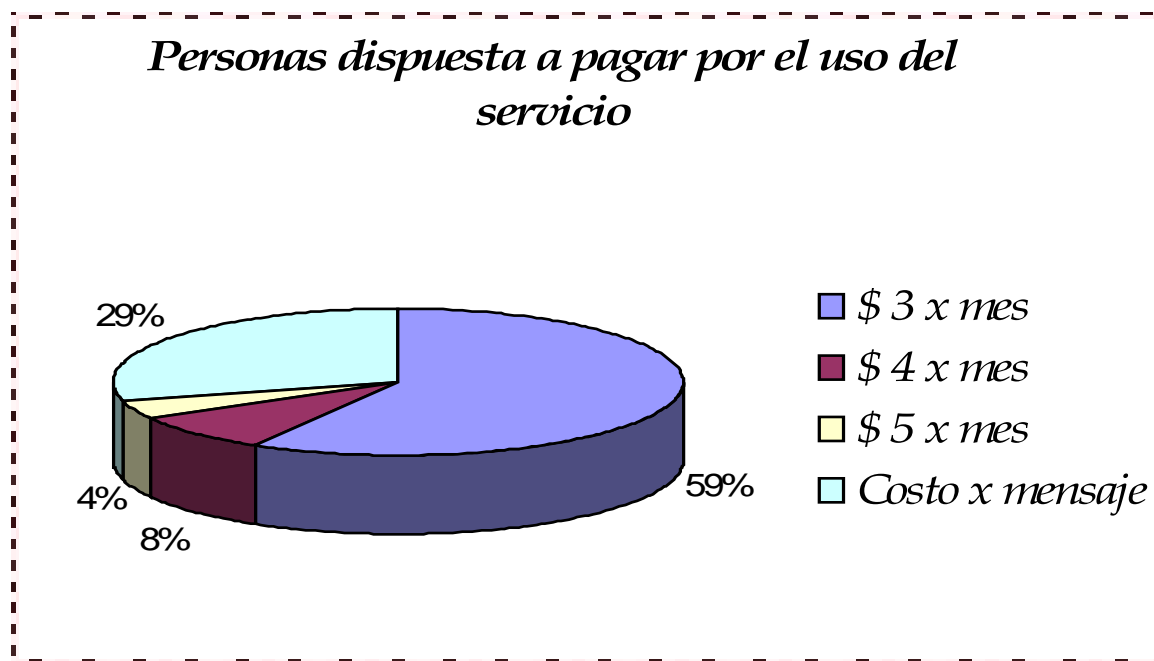
8.- ¿Le gustaría a usted recibir e-mail de personas que no posean computador y que lo puedan hacer llamando desde un teléfono convencional?

SI	NO
119	1



Si su respuesta es sí, ¿cuánto dinero estaría usted dispuesto a pagar por este servicio?

\$ 3 x mes	\$ 4 x mes	\$ 5 x mes	Costo x mensaje
70	10	5	35



ANÁLISIS.-

Por las encuestas realizadas se asume que el 69% de las personas tiene celular es decir la gran mayoría que comprende las edades desde 26 años en adelante ; Aproximadamente el 50%; que pertenece en gran proporción al sexo masculino(54%) y que se desempeñan como trabajadores, comprendiendo a empleados y empleadores, correspondiéndole el 69%.

Anexo D

Rutina para almacenar datos del PC a la memoria usando protocolo I²C

El proceso es el siguiente:

savemem2

```
    pagesel    readpic
    Bank0
    movlw      .42
    movwf     cnt
    call      readpic
    pagesel    savemem2
    xorlw     H'FF' ;    verifico si el msb esta sin setear
    btfsc    STATUS,2
    goto      $+2
    goto      mcontsetdirmem12
;si no tiene valores entonces grabo ceros en
    movlw     .42 ;direcciones 41 y 42 de la memoria ;
    eeprom del
    movwf     cnt ;PIC 16F877
    movlw     h'00'
    movwf     tempdpd
    pagesel    escpic
    call      escpic
    movlw     .41
    movwf     cnt
    movlw     h'00'
    movwf     tempdpd
    call      escpic
mcontsetdirmem12 ; carga en dirmem12 y dirmem22 la Bank0
; dirección máxima del 24lc256 (2)
    movlw     .41
    movwf     cnt
    pagesel    readpic
    call      readpic
    Bank1
    movwf     dirmem12
    Bank0
    movlw     .42
    movwf     cnt
    call      readpic
    Bank1
    movwf     dirmem22
savegpsmem2
    Bank1
    movlw     .30
```



```

movwf    topecprint
movlw    .1
movwf    acumcprint
lazogpsmem2
Bank1
movf     acumcprint,0
Bank0
movwf    cnt
pagesel  readpic
call     readpic
Bank0
movwf    datao
movlw    b'00000100'
movwf    slaveor
Bank1
movf     dirmem12,0 ;direccion de memoria 1 a ;grabar lsb
direc
Bank0
movwf    addr1
Bank1
movf     dirmem22,0 ;direccion de memoria 1 a ;grabar msb
direc
Bank0
movwf    addr2
call     escbyte ;graba en memoria i2c
pagesel  savemem2
Bank1
incf     acumcprint,1
movf     topecprint,0
subwf    acumcprint,0
btfss    STATUS,2
goto     $+2 ; si no llego al tope
goto     finsavemem2 ; si llega al tope ;continua con el
siguiente
Bank1
incf     dirmem12,1
btfss    STATUS,2
goto     $+2 ; salto
incf     dirmem22,1 ; incremento la dirección msb
goto     lazogpsmem2
finsavemem2
Bank0 ;
incf     addr1
btfss    STATUS,2
goto     $+2 ; salto
incf     addr2,1 ; incremento la dirección msb
movlw    .42 ; en las direcciones 42 y 41 de la
memoria ; eeprom del pic16f877 grabo las

```

```

movwf cnt ;direcciones máximas del 24LC256 (2)
movf addr2,0
movwf tempwd
pagesel escpic
call escpic
movlw .41
movwf cnt
movf addr1,0
movwf tempwd
call escpic
return

```

Cambio de bajo a alto en dato con reloj en alto.

escbyte

```

Bank1
movlw 0xff
movwf TRISC
Bank0

movlw b'10100000' ; modo de escritura
movwf slave
movf slaveor,w ; compara con la variable
iorwf slave,1
nop
nop

pagesel BSTART
call BSTART ; generate start bit
Bank0
movf slave,w
movwf txbuf ; guarda la configuración
pagesel TX
call TX ; transmito la dirección de escritura
Bank0
movf addr2,w ; msb dirección mas significativa
movwf txbuf ; guarda en txbuf
pagesel TX
call TX ; transmito la dirección mas significativa en la memoria
movf addr1,w ; lsb dirección menos significativa
movwf txbuf ; guarda en txbuf
pagesel TX
call TX ; transmito la dirección menos significativa en la
memoria
Bank0
movf datao,w ; byte a escribir
movwf txbuf ; guarda en txbuf

```

```

pagesel      TX
call TX      ; transmito el byte a ser escrito en la memoria
pagesel      BSTOP
call BSTOP   ; genero el bit de stop
              ;
pagesel      startmain
movlw .6
movwf loops  ; seteo del retardo
call WAIT    ; rutina de retardo de 6 ms despues de escribir el byte
Bank0
Return

```

BSTART

```

Bank0
bsf PORTC,5 ; asegura dato en alto
Bank1
movlw b'11000100' ; configura a reloj y dato como salida
movwf TRISC
Bank0
bcf PORTC,1 ; reloj en bajo
nop
bsf PORTC,1 ;reloj en alto
nop
nop
nop
nop
nop
nop
bcf PORTC,5 ; dato en bajo
nop
nop
nop
nop
nop
nop
bcf PORTC,1 ;reloj en bajo
nop
nop
return

```

BSTOP

```

Bank0
bcf PORTC,5 ; asegura que dato eeste en bajo
Bank1
movlw b'11000100' ; configura a reloj y dato como salida
movwf TRISC
Bank0

```

```

bcf   PORTC,5   ; dato en bajo
nop
nop
nop
bsf   PORTC,1   ; reloj en alto
nop
nop
nop
bsf   PORTC,5   ;dato en alto
nop
nop
bcf   PORTC,1   ;reloj en bajo
nop
nop
nop
return

```

```

TX
movlw .8
movwf count    ; configura para leer 8 bits

```

```

TXLP
bcf   eeprom,6 ;asume que el bit es bajo
btfsc txbuf,7  ; pregunta si el bit realmente esta en bajo
bsf   eeprom,6 ; pone al bit en bajo
call  BITOUT   ; envia el bit a la eeprom
rlf   txbuf, F ; rota
decfsz count, F ; pregunta si ya estan los 8bits
goto  TXLP     ; si no estan los 8 bits regresa
call  BITIN    ; lee el bit de reconocimiento
btfsc eeprom,7 ; chequea el bit de reconocimiento
;             bsf          PORTB,2 ; indica que no recibio el bit de
reconocimiento
;             ;             enciende RB0

nop
return

```

BITOUT

```

Bank1
movlw b'11000100'
movwf          TRISC
Bank0
btfss eeprom,6 ; chequea el estado del bit
goto  bitlow   ; dato en alto
bsf   PORTC,5  ; dato en bajo
goto  clkout
bitlow

```

```

    bcf   PORTC,5   ; dato en bajo
    clkout
    bsf   PORTC,1   ; reloj alto
    nop
    nop
    nop
    nop
    bcf   PORTC,1   ; reloj en bajo
    return

```

Anexo E

Programa general del prototipo de comunicación inalámbrica.-

```

;*****
;
;   Keyboard Emulator Celular model: NOKIA 5125
;
;           keyboard scan codes

list p=16f877,c=140   ; processor type
#include "p16f877.inc"
errorlevel           1, -(305)

__CONFIG  _XT_OSC & _WDT_OFF & _PWRTE_ON & _CP_OFF & _DEBUG_OFF & _BODEN_OFF &
_LVP_OFF

;Pines RB7,RB6 iran conectados a los pines 6 y 5 de la memoria serial 24LC156.
;Pines RB5,RB4,RB3 iran a las tres botoneras que simularan los tres eventos.
;Pin RB0 ira al led intermitente que indica actividad del circuito.
;Pines RB2,RB1 quedan como dos entradas (o salidas) disponibles para cualquier uso.

; NOTA: Los siguientes 10 caracteres tienen diferente código ASCII bajo DOS y WINDOWS:
; CHARACTER          CODIGO ASCII (DOS)      CODIGO ASCII (WINDOWS)
;   á                 160                    225
;   é                 130                    233
;   í                 161                    237
;   ó                 162                    243
;   ú                 163                    250

```

```

;   ü           129           252
;   ñ           164           241
;   Ñ           165           209
;   ¿           168           191
;   ¡           173           161

; Esto hace que en este programa se les haya asignado codigos que aparentemente no les
; corresponden (procedimiento "RECONOCER")

; Declaración de registros de funciones especiales

prueba      equ      0x21
loops       equ      0x22
loops2      equ      0x23
TEMP        equ      0x24 ;variable utilizada en los lazos de retardo.
TEMP2       equ      0x25 ;variable usada solamente en el procedimiento de
interrupcion.
STATUS_TEMP2 equ      0x26 ;variable usada solamente en el procedimiento de
interrupcion.

;VARIABLES RELACIONADAS CON EL MANEJO DE LA MEMORIA SERIAL:
;=====
datao       equ      0x27 ;registro para escribir datos en la memoria
slave       equ      0x28 ;direccion del dispositivo en el bus I2C(1010xxx0)
txbuf       equ      0x29
count       equ      0x2a
eeprom      equ      0x2b
rxbuf       equ      0x2c
addr        equ      0x2d ;posicion de memoria que se lee o escribe (menos
signif.)
highaddr    equ      0x2e ;posicion de memoria que se lee o escribe (mas signif.)
di          equ      .7 ;bit de entrada desde eeprom
do          equ      .6 ;bit de salida para eeprom
sdata       equ      .6 ;linea de datos seriales (pin RB6)
sclk        equ      .7 ;reloj serial (pin RB7)
;=====
=
ACCION_EJECUTAR equ 0x2f

```

```

BYTE_RECIBIDO equ 0x30
BYTE_RECIBIDO1 equ 0x31
DATO1 equ 0x32
CONTADOR1 equ 0x33 ; contador del numero de bytes de la cadena a enviar al
celular.
W_TEMP equ 0x34 ; context register (ISR)
STATUS_TEMP equ 0x35 ; context register (ISR)
PCLATH_TEMP equ 0x36 ; context register (ISR)
FSR_TEMP equ 0x37 ; context register (ISR)

```

```

;=====

```

```

=

```

```

;declaracion de macros

```

```

;Macros to select the register bank

```

```

;Many bank changes can be optimised when only one STATUS bit changes

```

```

Bank0 macro ;macro to select data RAM bank 0
    bcf STATUS,RP0
    bcf STATUS,RP1
endm

```

```

Bank1 macro ;macro to select data RAM bank 1
    bsf STATUS,RP0
    bcf STATUS,RP1
endm

```

```

Bank2 macro ;macro to select data RAM bank 2
    bcf STATUS,RP0
    bsf STATUS,RP1
endm

```

```

Bank3 macro ;macro to select data RAM bank 3
    bsf STATUS,RP0
    bsf STATUS,RP1
endm

```

```

;=====
=

;Configuracion de teclas
;Configuracion para el PUERTO RA (tiene 6 puertos E/S)

;          Tecla DOWN
;          Pin 2 PIC
;          Pin RA0 Salida (Port A Bit 0) '00000001'
;          Control 1 - IC1

;configuracion OUT DE PIC CON INTERRUPTORES 4016
    org 0
    goto start
nop                ;no operacion (retardo pequeño)
nop                ;no operacion (retardo pequeño)
nop                ;no operacion (retardo pequeño)
nop                ;no operacion (retardo pequeño)
org    0x05        ;direccion del programa del vector de interrupcion.

ISR          ;SUBRRUTINA DE INTERRUPCION=====
;*****
;*** ISR CONTEXT SAVE ***
;*****
nop
    bcf    INTCON,GIE    ; disable all interrupts
    btfsc INTCON,GIE    ; assure interrupts are disabled
    goto  ISR
    movwf W_TEMP        ; context save: W
    swapf STATUS,W      ; context save: STATUS
    movwf STATUS_TEMP   ; context save
    clrf  STATUS        ; bank 0, regardless of current bank
    movfw PCLATH        ; context save: PCLATH
    movwf PCLATH_TEMP   ; context save
    clrf  PCLATH        ; page zero, regardless of current page

```



```

    bcf          STATUS,IRP ; return to bank 0

    movfw  FSR          ; context save: FSR

    movwf  FSR_TEMP    ; context save

    ;*** context save done ***

;-----

    btfsc  PIR1, ADIF    ;Ver si la interrupcion proviene del convertidor.
    goto   INTER_CONVERTER ;si es asi ir a tratamiento de esa interrupcion.

    btfsc  PIR1, RCIF    ;Ver si la interrupcion proviene del receptor del USART.
    goto   INTER_USART   ;si es asi ir a tratamiento de esa interrupcion.
    goto   VOLVER        ;no, falsa interrupcion, ir a VOLVER.

;-----

INTER_CONVERTER

    movf   ADRESH, w    ;mover el resultado de la conversion (ADRESH) al registro w.
    movwf  DATO1

    bcf    PIR1, ADIF   ;borrar el flag de la interrupcion.
    goto   VOLVER

INTER_USART

    bcf    PIR1, RCIF   ;borrar el flag de la interrupcion.
    movf   RCREG, w     ;Lectura del dato recibido.
    movwf  BYTE_RECIBIDO1

;-----

;*****
;*** ISR TERMINATION (CONTEXT RESTORE) ***
;*****

VOLVER

    movfw  FSR_TEMP    ; context restore
    movwf  FSR          ; context restore
    movfw  PCLATH_TEMP ; context restore
    movwf  PCLATH       ; context restore
    swapf  STATUS_TEMP,W ; context restore
    movwf  STATUS       ; context restore
    swapf  W_TEMP,F     ; context restore
    swapf  W_TEMP,W     ; context restore

    RETFIE              ; enable global interrupt (INTCON,GIE)

```

```
;***** END OF INTERRUPT SERVICE ROUTINE *****
```

```
start
```

```
; Bank1
```

```
; clrf OPTION_REG
```

```
; clrf INTCON
```

```
; clrf PIE1
```

```
Bank0
```

```
call Retardo_4164us
```

```
call Retardo_4164us
```

```
call Retardo_4164us
```

```
clrf PORTA
```

```
clrf PORTD
```

```
movlw b'01111111'
```

```
movwf PORTB
```

```
movlw b'10000000'
```

```
movwf PORTC
```

```
Bank1
```

```
; movlw b'11101111' ; CONFIGURACION PREVIA DE INTERRUPCION
```

```
; movwf OPTION_REG
```

```
; movwf TRISA
```

```
movlw b'10000000' ;Pin RC7 es el pin receptor del USART.
```

```
movwf TRISC ;Pin RC6 es el pin transmisor del USART.
```

```
movlw b'00111101'
```

```
movwf TRISB
```

```
clrf TRISA
```

```
clrf TRISD
```

```
movlw b'00000111'
```

```
movwf ADCON1 ; porta inputs = digital not analog
```

```
; INICIALIZACION DEL USART
```

```

movlw  b'00100100'    ;Configuracion del USART, modo asincrono, etc.
movwf  TXSTA

;La formula para calcular el valor X que debe ir en el registro SPBRG es:
;Frecuencia en Baudios = Fosc / (K*(X+1))
;Aqui la Frecuencia en Baudios vale 9600, Fosc=6000000, K=16 (velocidad alta),
;de donde se deduce X=38,0625. En el registro SPBRG se escribira 38, entonces la
;Frecuencia en Baudios con X=38 sale 9615. El error porcentual es 0.16%.

movlw  .38            ;Valor calculado para trabajar a 9600 baudios,
movwf  SPBRG          ;con cristal de 6MHz.
movlw  b'00100000'    ;Activar solamente la interrupcion del receptor del USART.
movwf  PIE1

      Bank0
movlw  b'10010000'    ;configuracion del USART para recepcion continua.
movwf  RCSTA
bcf    PIR1, ADIF     ;borra el flag de interrupcion del convertidor.
bcf    PIR1, TXIF     ;borra el flag de interrupcion del transmisor del USART.
bsf    INTCON, PEIE   ;Permiso de interrupcion a los perifericos.
bsf    INTCON, GIE    ;Permiso global de interrupciones.

      call retardo

movlw  0x00
movwf  ACCION_EJECUTAR
movlw  0x00
movwf  BYTE_RECIBIDO
movlw  0x00
movwf  CONTADOR1
clrf   addr           ;limpia la direccion que apunta a la memoria serial.
clrf   highaddr       ;limpia la direccion que apunta a la memoria serial (mas signif.)
movlw  b'10100000'    ;la memoria serial tiene direccion 000.
movwf  slave

;Entra en un lazo de duracion infinita donde constantemente espera que lleguen del
;computador una secuencia de bytes, y ademas espera que se presione una de las 3
;botoneras para enviar el mensaje correspondiente al celular.(Las 3 botoneras simulan
;entradas provenientes de sensores que detectan intrusos, etc.)

```

```

sec1
    movf    BYTE_RECIBIDO1, w
    movwf   BYTE_RECIBIDO
    ; procede a comparar BYTE_RECIBIDO con el valor 11 que es la bandera
    ; que envia el computador cuando acto seguido va a enviar la direccion de correo 1.
    movlw   .11            ;Bandera.
    subwf   BYTE_RECIBIDO, 0 ;Resta BYTE_RECIBIDO menos W, resultado se guarda en W.
    btfss   STATUS, Z      ;Zero flag indica si BYTE_RECIBIDO es igual a W.
    goto    Label001
    movlw   .11
    movwf   ACCION_EJECUTAR ;11= recibiendo direccion de correo 1.
    goto    Label010
Label001
    ; procede a comparar BYTE_RECIBIDO con el valor 12 que es la bandera
    ; que envia el computador cuando acto seguido va a enviar el subject 1.
    movlw   .12            ;Bandera.
    subwf   BYTE_RECIBIDO, 0 ;Resta BYTE_RECIBIDO menos W, resultado se guarda en W.
    btfss   STATUS, Z      ;Zero flag indica si BYTE_RECIBIDO es igual a W.
    goto    Label002
    movlw   .12
    movwf   ACCION_EJECUTAR ;12= recibiendo subject 1.
    goto    Label010
Label002
    ; procede a comparar BYTE_RECIBIDO con el valor 13 que es la bandera
    ; que envia el computador cuando acto seguido va a enviar el mensaje 1.
    movlw   .13            ;Bandera.
    subwf   BYTE_RECIBIDO, 0 ;Resta BYTE_RECIBIDO menos W, resultado se guarda en W.
    btfss   STATUS, Z      ;Zero flag indica si BYTE_RECIBIDO es igual a W.
    goto    Label003
    movlw   .13
    movwf   ACCION_EJECUTAR ;13= recibiendo mensaje 1.
    goto    Label010
Label003
    ; procede a comparar BYTE_RECIBIDO con el valor 14 que es la bandera
    ; que envia el computador cuando acto seguido va a enviar la direccion de correo 2.

```

```

movlw  .14          ;Bandera.
subwf  BYTE_RECIBIDO, 0 ;Resta BYTE_RECIBIDO menos W, resultado se guarda en W.
btfss  STATUS, Z     ;Zero flag indica si BYTE_RECIBIDO es igual a W.
goto   Label004
movlw  .14
movwf  ACCION_EJECUTAR ;14= recibiendo direccion de correo 2.
goto   Label010

Label004
; procede a comparar BYTE_RECIBIDO con el valor 15 que es la bandera
; que envia el computador cuando acto seguido va a enviar el subject 2.
movlw  .15          ;Bandera.
subwf  BYTE_RECIBIDO, 0 ;Resta BYTE_RECIBIDO menos W, resultado se guarda en W.
btfss  STATUS, Z     ;Zero flag indica si BYTE_RECIBIDO es igual a W.
goto   Label005
movlw  .15
movwf  ACCION_EJECUTAR ;15= recibiendo subject 2.
goto   Label010

Label005
; procede a comparar BYTE_RECIBIDO con el valor 16 que es la bandera
; que envia el computador cuando acto seguido va a enviar el mensaje 2.
movlw  .16          ;Bandera.
subwf  BYTE_RECIBIDO, 0 ;Resta BYTE_RECIBIDO menos W, resultado se guarda en W.
btfss  STATUS, Z     ;Zero flag indica si BYTE_RECIBIDO es igual a W.
goto   Label006
movlw  .16
movwf  ACCION_EJECUTAR ;16= recibiendo mensaje 2.
goto   Label010

Label006
; procede a comparar BYTE_RECIBIDO con el valor 17 que es la bandera
; que envia el computador cuando acto seguido va a enviar la direccion de correo 3.
movlw  .17          ;Bandera.
subwf  BYTE_RECIBIDO, 0 ;Resta BYTE_RECIBIDO menos W, resultado se guarda en W.
btfss  STATUS, Z     ;Zero flag indica si BYTE_RECIBIDO es igual a W.
goto   Label007
movlw  .17
movwf  ACCION_EJECUTAR ;17= recibiendo direccion de correo 3.

```

```

    goto    Label010

Label007
    ; procede a comparar BYTE_RECIBIDO con el valor 18 que es la bandera
    ; que envia el computador cuando acto seguido va a enviar el subject 3.
    movlw  .18          ;Bandera.
    subwf  BYTE_RECIBIDO, 0 ;Resta BYTE_RECIBIDO menos W, resultado se guarda en W.
    btfss  STATUS, Z    ;Zero flag indica si BYTE_RECIBIDO es igual a W.
    goto   Label008
    movlw  .18
    movwf  ACCION_EJECUTAR ;18= recibiendo subject 3.
    goto   Label010

Label008
    ; procede a comparar BYTE_RECIBIDO con el valor 19 que es la bandera
    ; que envia el computador cuando acto seguido va a enviar el mensaje 3.
    movlw  .19          ;Bandera.
    subwf  BYTE_RECIBIDO, 0 ;Resta BYTE_RECIBIDO menos W, resultado se guarda en W.
    btfss  STATUS, Z    ;Zero flag indica si BYTE_RECIBIDO es igual a W.
    goto   Label009
    movlw  .19
    movwf  ACCION_EJECUTAR ;19= recibiendo mensaje 3.
    goto   Label010

Label009
    ; procede a comparar BYTE_RECIBIDO con el valor 20 que es la bandera
    ; que envia el computador cuando ya termino de enviar cualquier cadena de bytes.
    movlw  .20          ;Bandera.
    subwf  BYTE_RECIBIDO, 0 ;Resta BYTE_RECIBIDO menos W, resultado se guarda en W.
    btfss  STATUS, Z    ;Zero flag indica si BYTE_RECIBIDO es igual a W.
    goto   Label011
    bcf    PORTB, 1
        movf  ACCION_EJECUTAR, w
    movwf  highaddr
    movf   BYTE_RECIBIDO, w
    movwf  datao          ;dato a escribir en la EEPROM (24LC256).
    movlw  0x00
    movwf  BYTE_RECIBIDO1
    call   ESCRIBIR_EEPROM
    movlw  0x00

```

```

movwf  ACCION_EJECUTAR ;00= ninguna cadena esta siendo transmitida.

clrf   addr           ;limpia la direccion que apunta a la memoria serial.

clrf   highaddr      ;limpia la direccion que apunta a la memoria serial (mas signif.)

bsf    PORTB, 1

goto   Label012

Label011

; procede a comparar BYTE_RECIBIDO con el valor 0 el cual indica que el computador
; no esta enviando ningun byte.

movlw  .0

subwf  BYTE_RECIBIDO, 0 ;Resta BYTE_RECIBIDO menos W, resultado se guarda en W.

btfss  STATUS, Z      ;Zero flag indica si BYTE_RECIBIDO es igual a W.

goto   Label013

movlw  .0

subwf  ACCION_EJECUTAR, 0 ;Resta ACCION_EJECUTAR menos W, resultado se guarda en
W.

btfss  STATUS, Z      ;Zero flag indica si ACCION_EJECUTAR es igual a W.

goto   sec1

goto   Label012

Label013

;AQUI se llega si la computadora ha enviado un byte que no es igual a ninguna
;de las banderas cuyo valor varia entre 11 y 20, sino que mas bien es uno de los
;caracteres de la cadena actualmente siendo enviada.

;(los caracteres de la cadena tienen codigo ASCII variando entre 32 y 254)

bcf    PORTB, 1

movf   ACCION_EJECUTAR, w

movwf  highaddr

movf   BYTE_RECIBIDO, w

movwf  datao          ;dato a escribir en la EEPROM (24LC256).

;movf  datao, w

;movwf TXREG

;call  Retardo_4164us

;call  Retardo_4164us

;call  Retardo_4164us

```

```

movlw 0x00
movwf BYTE_RECIBIDO1
call ESCRIBIR_EEPROM

    call Retardo_4164us ;enviar eco a la computadora.
call LEER_EEPROM
movf rxbuf, w
movwf TXREG
    call Retardo_4164us
    call Retardo_4164us
    call Retardo_4164us

incf addr,1 ; Incrementar la direccion de la 24LC256.
bsf PORTB, 1
goto sec1

Label010
;Aqui se llega cuando la computadora ha enviado un byte (comando) del 11 al 19:
clrf addr ;limpia la direccion que apunta a la memoria serial.
clrf highaddr ;limpia la direccion que apunta a la memoria serial (mas signif.)
movlw 0x00
movwf BYTE_RECIBIDO1
;envia un ACK a la computadora que en este caso se lo define como el caracter 153:
movlw .153
movwf TXREG
    call Retardo_4164us
    call Retardo_4164us
    call Retardo_4164us
    goto sec1

Label012
btfsc PORTB,3
goto Label014
movlw .11
movwf highaddr
goto secuencia

Label014

```



```

    btfsc    PORTB,4
    goto     Label015
    movlw   .14
    movwf   highaddr
    goto     secuencia
Label015
    btfsc    PORTB,5
    goto     Label016
    movlw   .17
    movwf   highaddr
    goto     secuencia
Label016
    goto     sec1

secuencia
    call encera

    movlw   .0
    movwf   addr
    call    LEER_EEPROM
    movf    rxbuf, w
    movwf   DATO1
    movlw   .10           ;Bandera que indica QUE ES MENSAJE PARA TELEFONO CELULAR.
    subwf   DATO1, 0      ;Resta DATO1 menos W, resultado se guarda en W.
    btfsc   STATUS, Z     ;Zero flag indica si DATO1 es igual a W.
    goto    secuencial    ;EJECUTAR SECUENCIA QUE ES PARA TELEFONO CELULAR.

    call prepara_email

    movlw   .0
    movwf   addr
    call prepara_direcc2

    movlw   .0
    movwf   addr
    incf    highaddr, 1
    call escribe_subject
    movlw   .0

```

```

movwf  addr
incf  highaddr, 1
call  escribe_body
      ;call llama_teclado
      call envia_mensaje
      call salir_menu
      goto secl

```

secuencial

```

call  encera
movlw  .0
movwf  addr
incf  highaddr, 1
incf  highaddr, 1
      call prepara_texto
movlw  .1  ;comienza en la direccion 1 porque en la direccion cero se encuentra
          ;el caracter 10 que es la bandera que nos indica que la direccion de
          ;correo es realmente un numero de telefono celular.
movwf  addr
decf  highaddr, 1
decf  highaddr, 1
call  envia_celular
call  menu
call  retardo
call  retardo
call  retardo
call  retardo
call  retardo
call  retardo
call  retardo
call  retardo
call  retardo
call  retardo
call  retardo
call  retardo
call  abajo
call  retardo
call  abajo
call  retardo

```

```
call menu
call retardo
call borra
call retardo
call borra
call retardo
call borra
call retardo
call borra
call retardo
call borra
call retardo
call borra
call retardo
goto sec1
```

;prepara la ventana para escribir el texto y enviarlo al numero de celular.

prepara_texto

```
call encera
call menu
call retardo
call retardo
call retardo
call abajo
call retardo
call menu
call retardo
call retardo
call menu
call retardo
call retardo
call abajo
call retardo
call abajo
call retardo
call menu
call retardo
call retardo
```

```

    call letmay

    ; Ingresa en un lazo que va extrayendo de la memoria 24LC256 los caracteres a
enviar

    ; al celular, en codigo ASCII, y de acuerdo al valor de este codigo se ejecuta
    ; la secuencia del teclado del celular que corresponda (procedimiento RECONOCER):
Label023
    bcf    PORTB, 1
    call   LEER_EEPROM
    movf   rxbuf, w
    movwf  DAT01
    movlw  .20          ;Bandera que indica fin de cadena.
    subwf  DAT01, 0     ;Resta DAT01 menos W, resultado se guarda en W.
    btfsc  STATUS, Z   ;Zero flag indica si DAT01 es igual a W.
    goto   Label024
    movlw  .80          ;se acepta enviar maximo 80 caracteres en esta cadena.
    subwf  addr, 0      ;Resta addr menos W, resultado se guarda en W.
    btfsc  STATUS, Z   ;Zero flag indica si addr es igual a W.
    goto   Label024
    movf   DAT01, w
    bsf    PORTB, 1
    call   RECONOCER
    incf   addr,1       ; Incrementar la direccion.
    goto   Label023
Label024
    return

; Envia el mensaje escrito a un numero celular "099256998"
envia_celular
    call  encera
    call  retardo
    call  menu
    call  retardo
    call  retardo
    call  menu
    call  retardo
    call  retardo

```

;Aqui el lazo que extrae de la memoria serial el numero del telefono celular al que
se

;le va a enviar el mensaje:

Label026

```
bcf    PORTB, 1
call   LEER_EEPROM
movf   rxbuf, w
movwf  DATO1
movlw  .20          ;Bandera que indica fin de cadena.
subwf  DATO1, 0     ;Resta DATO1 menos W, resultado se guarda en W.
btfsc  STATUS, Z   ;Zero flag indica si DATO1 es igual a W.
goto   Label025
movlw  .70          ;se acepta enviar maximo 70 caracteres en esta cadena.
subwf  addr, 0     ;Resta addr menos W, resultado se guarda en W.
btfsc  STATUS, Z   ;Zero flag indica si addr es igual a W.
goto   Label025
movf   DATO1, w
bsf    PORTB, 1
call   RECONOCER2
incf   addr,1      ; Incrementar la direccion.
goto   Label026
```

Label025

```
call  retardo
call  retardo
;    call num0
;    call retardo
;    call num9
;    call retardo
;    call num9
;    call retardo
;    call num2
;    call retardo
;    call num5
;    call retardo
;    call num6
```

```

;    call retardo
;    call num9
;    call retardo
;    call num9
;    call retardo
;    call num8
;    call retardo
    return

```

```

;configuro la interrupción

```

```

;    movlw  b'10001000'
;    movwf  INTCON

```

```

;lazo1

```

```

;    sleep

```

```

;    call  lazo1

```

```

;incrementa un contador cnt.

```

```

;funcion retardo de acuerdo al valor que tenga

```

```

;demora mas o menos tiempo, llama a la funcion tiempo

```

```

;que tiene un retardo de 250 ms.

```

```

;si se desea aumentar el tiempo del retardo o disminuirlo

```

```

;cambiar parametro que se escribe en variable "prueba":

```

```

retardo

```

```

    movlw  .4

```

```

    movwf  prueba

```

```

    call  lazo2

```

```

Bank0
movlw b'00000000'
movwf PORTA
movwf PORTD

bcf     PORTC, 0
bcf     PORTC, 1
bcf     PORTC, 2
bcf     PORTC, 3
bcf     PORTC, 4
bcf     PORTC, 5
bcf     PORTC, 6

bcf     PORTB, 1
movlw   .4
movwf   prueba
call    lazo2
bsf     PORTB, 1
return

lazo2   decfsz prueba,1
        goto   tiempo
        return

tiempo
movlw   .200           ; para cristal de 6 Mhz.
movwf   loops         ; seteo del retardo
call    WAIT          ; rutina de retardo de 10 ms despues de escribir el byte
goto   lazo2

retardo2
movlw   .4
movwf   prueba
call    lazo22

Bank0
movlw b'00000000'
movwf PORTA
movwf PORTD
bcf     PORTC, 0

```

```

        bcf      PORTC, 1
        bcf      PORTC, 2
        bcf      PORTC, 3
        bcf      PORTC, 4
        bcf      PORTC, 5
        bcf      PORTC, 6

    bcf      PORTB, 1

        movlw   .4
        movwf   prueba
        call    lazo22
    bsf      PORTB, 1

        return

lazo22 decfsz prueba,1
        goto    tiempo2

        return

tiempo2

        movlw   .40          ; para cristal de 6 Mhz.
        movwf   loops       ; seteo del retardo
        call    WAIT        ; rutina de retardo de 10 ms despues de escribir el byte
        goto    lazo22

;Funcion retardo de acuerdo al valor que tenga
;demora mas o menos tiempo, llama a la funcion tiempo
;que tiene un retardo de 250 ms.
;si se desea aumentar el tiempo del retardo o disminuirlo
;cambiar parametro d='xx'

;*****

WAIT
;
top    movlw   .110          ; ajusta variable de tiempo Valor original
;top   movlw   .1           ; ajusta variable de tiempo
        movwf   loops2

```



```

top2    nop

        nop

        nop

        nop

        nop

        nop

        decfsz loops2, F      ; pregunta para salir del lazo

        goto    top2

        decfsz loops, F      ; pregunta para salir del lazo

        goto    top

        retlw   0            ;retorna

```

```

;*****

```

```

encera

```

```

Bank0

movlw b'00000000'

movwf PORTA

movwf PORTD

bcf    PORTC, 0

bcf    PORTC, 1

bcf    PORTC, 2

bcf    PORTC, 3

bcf    PORTC, 4

bcf    PORTC, 5

bcf    PORTC, 6

call  retardo

return

```

```

num1

```

```

movlw b'00100000'

movwf PORTD

return

```

```

abajo

    movlw b'00000001'
    movwf PORTA
    return

;          Tecla CLEAR
;
;          Pin 3 PIC
;
;          Pin RA1 Salida (Port A Bit 1) '00000010'
;
;          Control 2 - IC1
borra

    movlw b'00001000'
    movwf PORTA
    return

;          Tecla MENU
;
;          Pin 4 PIC
;
;          Pin RA2 Salida (Port A Bit 2) '00000100'
;
;          Control 3 - IC1
menu

    movlw b'00000010'
    movwf PORTA
    return

;          Pin 5 PIC
;
;          Pin RA3 Salida (Port A Bit 3) '00001000'
;
;          Control 4 - IC1
arriba

    movlw b'00000100'
    movwf PORTA
    return

;          Tecla Numero 4
;
;          Pin 6 PIC
;
;          Pin RA4 Salida (Port A Bit 4) '00010000'
;
;          Control 1 - IC2
num4

```

```

movlw b'00100000'

movwf PORTA

return

;          Tecla Numero 1
;          Pin 7 PIC
;          Pin RA5 Salida (Port A Bit 5) '00100000'
;          Control 2 - IC2

;Configuracion para el PUERTO RC (tiene 8 puertos E/S)

;          Tecla Numero 9
;          Pin 15 PIC
;          Pin RC0 Salida (Port C Bit 0) '00000001'
;          Control 3 - IC2
num9
movlw b'00010000'

movwf PORTD

return

;          Tecla Numero #
;          Pin 16 PIC
;          Pin RC1 Salida (Port C Bit 1) '00000010'
;          Control 4 - IC2
numer
movlw b'10100000'

movwf PORTC

return

;          Tecla Numero 7
;          Pin 17 PIC
;          Pin RC2 Salida (Port C Bit 2) '00000100'
;          Control 1 - IC3
num7
movlw b'10000001'

movwf PORTC

return

```

```

;   Tecla Numero '*'
;
;       Pin 18 PIC
;
;       Pin RC3 Salida (Port C Bit 3) '00001000'
;
;       Control 2 - IC3
aster
    movlw b'10000010'
    movwf PORTC
    return

;   Tecla Numero 2
;
;       Pin 19 PIC
;
;       Pin RD0 Salida (Port D Bit 0) '00000001'
;
;       Control 3 - IC3
num2
    movlw b'10000100'
    movwf PORTC
    return

;   Tecla Numero 5
;
;       Pin 20 PIC
;
;       Pin RD1 Salida (Port D Bit 1) '00000010'
;
;       Control 4 - IC3
num5
    movlw b'01000000'
    movwf PORTD
    return

;   Tecla Numero 8
;
;       Pin 21 PIC
;
;       Pin RD2 Salida (Port D Bit 2) '00000100'
;
;       Control 1 - IC4
num8
    movlw b'00000001'
    movwf PORTD
    return

```

```

; Tecla Numero 0
;
; Pin 22 PIC
; Pin RD3 Salida (Port D Bit 3) '000001000'
; Control 2 - IC4
num0
    movlw b'00000010'
    movwf PORTD
    return

; Tecla Numero 3
;
; Pin 23 PIC
; Pin RC4 Salida (Port C Bit 4) '00010000'
; Control 3 - IC4
num3
    movlw b'00000100'
    movwf PORTD
    return

; Tecla Numero 6
;
; Pin 24 PIC
; Pin RC5 Salida (Port C Bit 5) '00100000'
; Control 4 - IC4
num6
    movlw b'00001000'
    movwf PORTD
    return

; Salir del modo de escribir mensaje
salir_menu
    call encera
    call abajo
    call retardo
    call abajo
    call retardo
    call abajo

```



```
call borra
call retardo
call borra
call retardo
return
```

```
;Secuencia para mensajes texto vía e-mail
```

```
;Ingreso a la opcion de enviar e-mail
```

```
prepara_email
```

```
call encera
call menu
call retardo
call retardo
call retardo
call abajo
call retardo
call menu
call retardo
call retardo
call menu
call retardo
call retardo
call abajo
call retardo
call abajo
call retardo
call abajo
call retardo
call menu
call retardo
call retardo
call num1      ;se cambio num1 solo para pruebas.
call retardo
call num2      ;se cambio num2 solo para pruebas.
call retardo
```

```

call num3      ;se cambio num3 solo para pruebas.

call retardo

call num4      ;se cambio num4 solo para pruebas.

call retardo

call menu

call retardo

call retardo

return

;Escribe mensaje corto o Letras Palabra "hola"

escribe_subject

    call encera

    call letmay

    ; Ingresa en un lazo que va extrayendo de la memoria 24LC256 los caracteres a
enviar

    ; al celular, en codigo ASCII, y de acuerdo al valor de este codigo se ejecuta

    ; la secuencia del teclado del celular que corresponda (procedimiento RECONOCER):
Label020

    bcf     PORTB, 1

    call    LEER_EEPROM

    movf    rxbuf, w

    movwf   DAT01

    movlw   .20      ;Bandera que indica fin de cadena.

    subwf   DAT01, 0 ;Resta DAT01 menos W, resultado se guarda en W.

    btfsc   STATUS, Z ;Zero flag indica si DAT01 es igual a W.

    goto    Label019

    movlw   .70      ;se acepta enviar maximo 70 caracteres en esta cadena.

    subwf   addr, 0  ;Resta addr menos W, resultado se guarda en W.

    btfsc   STATUS, Z ;Zero flag indica si addr es igual a W.

    goto    Label019

    movf    DAT01, w

    bsf     PORTB, 1

    call    RECONOCER

    incf    addr,1   ; Incrementar la direccion.

```



```

goto    Label020

;call num4          ; MENSAJE "HOLA"
;call retardo2
;call num4
;call retardo
;call num6
;call retardo2
;call num6
;call retardo2
;call num6
;call retardo
;call num5
;call retardo2
;call num5
;call retardo2
;call num5
;call retardo
;call num2
;call retardo

Label019
call menu
call retardo
call retardo
return

escribe_body
call encera
call letmay

; Ingresa en un lazo que va extrayendo de la memoria 24LC256 los caracteres a
enviar

; al celular, en codigo ASCII, y de acuerdo al valor de este codigo se ejecuta
; la secuencia del teclado del celular que corresponda (procedimiento RECONOCER):

Label022

```

```

bcf     PORTB, 1
call    LEER_EEPROM
movf    rxbuf, w
movwf   DATO1
movlw   .20           ;Bandera que indica fin de cadena.
subwf   DATO1, 0      ;Resta DATO1 menos W, resultado se guarda en W.
btfsc   STATUS, Z     ;Zero flag indica si DATO1 es igual a W.
goto    Label021
movlw   .80           ;se acepta enviar maximo 80 caracteres en esta cadena.
subwf   addr, 0       ;Resta addr menos W, resultado se guarda en W.
btfsc   STATUS, Z     ;Zero flag indica si addr es igual a W.
goto    Label021
movf    DATO1, w
bsf     PORTB, 1
call    RECONOCER
incf    addr,1        ; Incrementar la direccion.
goto    Label022

;call num4           ;MENSAJE "HOLA"
;call retardo2
;call num4
;call retardo
;call num6
;call retardo2
;call num6
;call retardo2
;call num6
;call retardo
;call num5
;call retardo2
;call num5
;call retardo2
;call num5
;call retardo
;call num2
;call retardo

```

Label021

```

return

; Envía el mensaje escrito
envia_mensaje
    call encera
    call menu
    call retardo
    call retardo
    call menu
    call retardo
    call retardo
    call retardo
    call retardo
    call retardo
    call retardo
    call retardo
    call retardo
    call retardo
    call retardo
    call retardo
    return

;Ingreso de dirección e-mail: fnovillo@ceibo.fiec.edu.ec
;Dirección de correo
;prepara_direcc
;    call encera
;    letraf
;    call retardo
;    letran
;    call retardo
;    letrao
;    call retardo
;    letrav
;    call retardo
;    letrai

```

; call retardo
; letral
; call retardo
; letral
; call retardo
; letrao
; call retardo
; letarr
; call retardo
; letrac
; call retardo
; letrae
; call retardo
; letrai
; call retardo
; letrab
; call retardo
; letrao
; call retardo
; letpun
; call retardo
; letraf
; call retardo
; letrai
; call retardo
; letrae
; call retardo
; letrac
; call retardo
; letpun
; call retardo
; letrae
; call retardo
; letrad
; call retardo
; letrau
; call retardo

```

; letpun
; call retardo
; letrae
; call retardo
; letrac
; call retardo
; call menu
; call retardo
; call retardo
; return

```

```

;Ingreso de dirección e-mail: ivan_ramz99@hotmail.com

```

```

;Dirección de correo

```

```

prepara_direcc2

```

```

    call encera

```

```

; Ingresa en un lazo que va extrayendo de la memoria 24LC256 los caracteres a
enviar

```

```

; al celular, en codigo ASCII, y de acuerdo al valor de este codigo se ejecuta

```

```

; la secuencia del teclado del celular que corresponda (procedimiento RECONOCER):

```

```

Label018

```

```

    bcf    PORTB, 1

```

```

    call  LEER_EEPROM

```

```

    movf  rxbuf, w

```

```

    movwf DAT01

```

```

    movlw .20          ;Bandera que indica fin de cadena.

```

```

    subwf DAT01, 0     ;Resta DAT01 menos W, resultado se guarda en W.

```

```

    btfsc STATUS, Z   ;Zero flag indica si DAT01 es igual a W.

```

```

    goto  Label017

```

```

    movlw .70          ;se acepta enviar maximo 70 caracteres en esta cadena.

```

```

    subwf addr, 0     ;Resta addr menos W, resultado se guarda en W.

```

```

    btfsc STATUS, Z   ;Zero flag indica si addr es igual a W.

```

```

    goto  Label017

```

```

    movf  DAT01, w

```

```

    bsf   PORTB, 1

```

```

    call  RECONOCER

```

```

    incf  addr,1      ; Incrementar la direccion.

```

```
goto    Label018
```

```
;letrai  
;call retardo  
;letrav  
;call retardo  
;letraa  
;call retardo  
;letran  
;call retardo  
;letsub  
;call retardo  
;letrar  
;call retardo  
;letraa  
;call retardo  
;letram  
;call retardo  
;letraz  
;call retardo  
;letra9  
;call retardo  
;letra9  
;call retardo  
;letarr  
;call retardo  
;letrah  
;call retardo  
;letrao  
;call retardo  
;letrat  
;call retardo  
;letram  
;call retardo  
;letraa  
;call retardo  
;letrai
```

```

;call retardo

;letral

;call retardo

;letpun

;call retardo

;letrac

;call retardo

;letrao

;call retardo

;letram

;call retardo

Label017

call menu

call retardo

call retardo

return

;*****

; RCNCR.ASM

; TOPICO DE GRADUACION

; PARTE DE comprobacion de la comunicaci3n serial RS232

;

; DESCRIPTION:

; =====

;El siguiente programa realiza la comprobacion de los datos enviado del PC y el Pic

;tendra que verificar si el caracter recibido corresponde a la tabla ASCII que posee

;almacenado el PIC para lo cual establecera el comando de control para manejar el

;celular.

;

; Developed and tested on PIC 16F877.

;

;*****

; TABLA DE CODIGO ASCII

RECONOCER

; CHARACTER " @ "

```

```

movf    DAT01, w
        xorlw h'40' ; " @ "
        btfsc STATUS,2
        goto  CTRLA0
        ;goto A0

; CHARACTER " A "
A0
        movf    DAT01, w
        xorlw h'41' ; " A "
        btfsc STATUS,2
        goto  CTRLA1
        ;goto A1

; CHARACTER " B "
A1
        movf    DAT01, w
        xorlw h'42' ; " B "
        btfsc STATUS,2
        goto  CTRLA2
        ;goto A2

; CHARACTER " C "
A2
        movf    DAT01, w
        xorlw h'43' ; " C "
        btfsc STATUS,2
        goto  CTRLA3
        ;goto A3

; CHARACTER " D "
A3
        movf    DAT01, w
        xorlw h'44' ; " D "
        btfsc STATUS,2
        goto  CTRLA4
        ;goto A4

```



```

; CHARACTER " E "
A4
    movf    DAT01, w
    xorlw  h'45' ; " E "
    btfsc  STATUS,2
    goto   CTRLA5
;goto A5

; CHARACTER " F "
A5
    movf    DAT01, w
    xorlw  h'46' ; " F "
    btfsc  STATUS,2
    goto   CTRLA6
;goto A6

; CHARACTER " G "
A6
    movf    DAT01, w
    xorlw  h'47' ; " G "
    btfsc  STATUS,2
    goto   CTRLA7
;goto A7

; CHARACTER " H "
A7
    movf    DAT01, w
    xorlw  h'48' ; " H "
    btfsc  STATUS,2
    goto   CTRLA8
;goto A8

; CHARACTER " I "
A8
    movf    DAT01, w
    xorlw  h'49' ; " I "

```

```

        btfsc STATUS,2
        goto CTRLA9
        ;goto A9

; CHARACTER " J "
A9
        movf DAT01, w
        xorlw h'4A' ; " J "
        btfsc STATUS,2
        goto CTRLA10
        ;goto A10

; CHARACTER " K "
A10
        movf DAT01, w
        xorlw h'4B' ; " K "
        btfsc STATUS,2
        goto CTRLA11
        ;goto A11

; CHARACTER " L "
A11
        movf DAT01, w
        xorlw h'4C' ; " L "
        btfsc STATUS,2
        goto CTRLA12
        ;goto A12

; CHARACTER " M "
A12
        movf DAT01, w
        xorlw h'4D' ; " M "
        btfsc STATUS,2
        goto CTRLA13
        ;goto A13

; CHARACTER " N "

```

A13

```
movf   DAT01, w
xorlw  h'4E' ; " N "
btfsc  STATUS,2
goto   CTRLA14
;goto  A14
```

; CHARACTER " O "

A14

```
movf   DAT01, w
xorlw  h'4F' ; " O "
btfsc  STATUS,2
goto   CTRLA15
;goto  A15
```

; CHARACTER " P "

A15

```
movf   DAT01, w
xorlw  h'50' ; " P "
btfsc  STATUS,2
goto   CTRLA16
;goto  A16
```

; CHARACTER " Q "

A16

```
movf   DAT01, w
xorlw  h'51' ; " Q "
btfsc  STATUS,2
goto   CTRLA17
;goto  A17
```

; CHARACTER " R "

A17

```
movf   DAT01, w
xorlw  h'52' ; " R "
btfsc  STATUS,2
goto   CTRLA18
```

```

                ;goto A18

; CHARACTER " S "
A18
        movf    DAT01, w
        xorlw   h'53' ; " S "
        btfsc  STATUS,2
        goto    CTRLA19
        ;goto  A19

; CHARACTER " T "
A19
        movf    DAT01, w
        xorlw   h'54' ; " T "
        btfsc  STATUS,2
        goto    CTRLA20
        ;goto  A20

; CHARACTER " U "
A20
        movf    DAT01, w
        xorlw   h'55' ; " U "
        btfsc  STATUS,2
        goto    CTRLA21
        ;goto  A21

; CHARACTER " V "
A21
        movf    DAT01, w
        xorlw   h'56' ; " V "
        btfsc  STATUS,2
        goto    CTRLA22
        ;goto  A22

; CHARACTER " W "
A22
        movf    DAT01, w

```

```

        xorlw h'57' ; " W "
        btfsc STATUS,2
        goto CTRLA23
        ;goto A23

; CHARACTER " X "
A23
        movf DAT01, w
        xorlw h'58' ; " X "
        btfsc STATUS,2
        goto CTRLA24
        ;goto A24

; CHARACTER " Y "
A24
        movf DAT01, w
        xorlw h'59' ; " Y "
        btfsc STATUS,2
        goto CTRLA25
        ;goto A25

; CHARACTER " Z "
A25
        movf DAT01, w
        xorlw h'5A' ; " Z "
        btfsc STATUS,2
        goto CTRLA26
        ;goto A26

; CHARACTER " a "
A26
        movf DAT01, w
        xorlw h'61' ; " a "
        btfsc STATUS,2
        goto CTRLA27
        ;goto A27

```

```

; CHARACTER " b "
A27
    movf    DAT01, w
    xorlw  h'62' ; " b "
    btfsc  STATUS,2
    goto   CTRLA28
    ;goto  A28

; CHARACTER " c "
A28
    movf    DAT01, w
    xorlw  h'63' ; " c "
    btfsc  STATUS,2
    goto   CTRLA29
    ;goto  A29

; CHARACTER " d "
A29
    movf    DAT01, w
    xorlw  h'64' ; " d "
    btfsc  STATUS,2
    goto   CTRLA30
    ;goto  A30

; CHARACTER " e "
A30
    movf    DAT01, w
    xorlw  h'65' ; " e "
    btfsc  STATUS,2
    goto   CTRLA31
    ;goto  A31

; CHARACTER " f "
A31
    movf    DAT01, w
    xorlw  h'66' ; " f "
    btfsc  STATUS,2

```

```

        goto    CTRLA32
        ;goto  A32

; CHARACTER " g "
A32
        movf   DAT01, w
        xorlw  h'67' ; " g "
        btfsc  STATUS,2
        goto   CTRLA33
        ;goto  A33

; CHARACTER " h "
A33
        movf   DAT01, w
        xorlw  h'68' ; " h "
        btfsc  STATUS,2
        goto   CTRLA34
        ;goto  A34

; CHARACTER " i "
A34
        movf   DAT01, w
        xorlw  h'69' ; " i "
        btfsc  STATUS,2
        goto   CTRLA35
        ;goto  A35

; CHARACTER " j "
A35
        movf   DAT01, w
        xorlw  h'6A' ; " j "
        btfsc  STATUS,2
        goto   CTRLA36
        ;goto  A36

; CHARACTER " k "
A36

```

```

        movf   DAT01, w
        xorlw  h'6B' ; " k "
        btfsc STATUS,2
        goto   CTRLA37
        ;goto  A37

; CHARACTER " l "

A37
        movf   DAT01, w
        xorlw  h'6C' ; " l "
        btfsc STATUS,2
        goto   CTRLA38
        ;goto  A38

; CHARACTER " m "

A38
        movf   DAT01, w
        xorlw  h'6D' ; " m "
        btfsc STATUS,2
        goto   CTRLA39
        ;goto  A39

; CHARACTER " n "

A39
        movf   DAT01, w
        xorlw  h'6E' ; " n "
        btfsc STATUS,2
        goto   CTRLA40
        ;goto  A40

; CHARACTER " o "

A40
        movf   DAT01, w
        xorlw  h'6F' ; " o "
        btfsc STATUS,2
        goto   CTRLA41
        ;goto  A41

```



```

; CHARACTER " p "
A41
    movf    DAT01, w
    xorlw  h'70' ; " p "
    btfsc  STATUS,2
    goto   CTRLA42
;goto A42

; CHARACTER " q "
A42
    movf    DAT01, w
    xorlw  h'71' ; " q "
    btfsc  STATUS,2
    goto   CTRLA43
;goto A43

; CHARACTER " r "
A43
    movf    DAT01, w
    xorlw  h'72' ; " r "
    btfsc  STATUS,2
    goto   CTRLA44
;goto A44

; CHARACTER " s "
A44
    movf    DAT01, w
    xorlw  h'73' ; " s "
    btfsc  STATUS,2
    goto   CTRLA45
;goto A45

; CHARACTER " t "
A45
    movf    DAT01, w
    xorlw  h'74' ; " t "

```

```

        btfsc STATUS,2
        goto CTRLA46
        ;goto A46

; CHARACTER " u "
A46
        movf DAT01, w
        xorlw h'75' ; " u "
        btfsc STATUS,2
        goto CTRLA47
        ;goto A46

; CHARACTER " v "
A47
        movf DAT01, w
        xorlw h'76' ; " v "
        btfsc STATUS,2
        goto CTRLA48
        ;goto A48

; CHARACTER " w "
A48
        movf DAT01, w
        xorlw h'77' ; " w "
        btfsc STATUS,2
        goto CTRLA49
        ;goto A49

; CHARACTER " x "
A49
        movf DAT01, w
        xorlw h'78' ; " x "
        btfsc STATUS,2
        goto CTRLA50
        ;goto A50

; CHARACTER " y "

```

A50

```
movf    DAT01, w
xorlw   h'79' ; " y "
btfsc   STATUS,2
goto    CTRLA51
;goto   A51
```

; CHARACTER " z "

A51

```
movf    DAT01, w
xorlw   h'7A' ; " z "
btfsc   STATUS,2
goto    CTRLA52
;goto   A52
```

; CHARACTER " 0 "

A52

```
movf    DAT01, w
xorlw   h'30' ; " 0 "
btfsc   STATUS,2
goto    CTRLA53
;goto   A53
```

; CHARACTER " 1 "

A53

```
movf    DAT01, w
xorlw   h'31' ; " 1 "
btfsc   STATUS,2
goto    CTRLA54
;goto   A54
```

; CHARACTER " 2 "

A54

```
movf    DAT01, w
xorlw   h'32' ; " 2 "
btfsc   STATUS,2
goto    CTRLA55
```

```

                ;goto A55

; CHARACTER " 3 "
A55
    movf    DAT01, w
    xorlw   h'33' ; " 3 "
    btfsc   STATUS,2
    goto    CTRLA56
    ;goto   A56

; CHARACTER " 4 "
A56
    movf    DAT01, w
    xorlw   h'34' ; " 4 "
    btfsc   STATUS,2
    goto    CTRLA57
    ;goto   A57

; CHARACTER " 5 "
A57
    movf    DAT01, w
    xorlw   h'35' ; " 5 "
    btfsc   STATUS,2
    goto    CTRLA58
    ;goto   A58

; CHARACTER " 6 "
A58
    movf    DAT01, w
    xorlw   h'36' ; " 6 "
    btfsc   STATUS,2
    goto    CTRLA59
    ;goto   A59

; CHARACTER " 7 "
A59
    movf    DAT01, w

```

```

        xorlw h'37' ; " 7 "
        btfsc STATUS,2
        goto CTRLA60
        ;goto A60

; CHARACTER " 8 "
A60
        movf DAT01, w
        xorlw h'38' ; " 8 "
        btfsc STATUS,2
        goto CTRLA61
        ;goto A61

; CHARACTER " 9 "
A61
        movf DAT01, w
        xorlw h'39' ; " 9 "
        btfsc STATUS,2
        goto CTRLA62
        ;goto A62

; CHARACTER ESPECIAL " RETORNO DE CARRO CR ( <-- )"
A62
        movf DAT01, w
        xorlw h'13' ; " <-- "
        btfsc STATUS,2
        goto CTRLA63
        ;goto A63

; CHARACTER ESPECIAL " SUPR/DEL "
A63
        movf DAT01, w
        xorlw h'FF' ; " DEL "
        btfsc STATUS,2
        goto CTRLA64
        ;goto A64

```

```

; CHARACTER ESPECIAL " BACKSPACE "
A64
    movf   DAT01, w
    xorlw  h'08' ; " BACKSPACE "
    btfsc  STATUS,2
    goto   CTRLA65
    ;goto  A65

```

```

; CHARACTER ESPECIAL " * "
A65
    movf   DAT01, w
    xorlw  h'2A' ; " * "
    btfsc  STATUS,2
    goto   CTRLA66
    ;goto  A66

```

```

; CHARACTER ESPECIAL " + "
A66
    movf   DAT01, w
    xorlw  h'2B' ; " + "
    btfsc  STATUS,2
    goto   CTRLA67
    ;goto  A67

```

```

; CHARACTER ESPECIAL " - "
A67
    movf   DAT01, w
    xorlw  h'2D' ; " - "
    btfsc  STATUS,2
    goto   CTRLA68
    ;goto  A68

```

```

; CHARACTER ESPECIAL " / "
A68
    movf   DAT01, w
    xorlw  h'2F' ; " / "
    btfsc  STATUS,2

```

```

        goto    CTRLA69
        ;goto   A69

; CARACTER ESPECIAL " ( "
A69

        movf   DATO1, w
        xorlw  h'28' ; " ( "
        btfsc  STATUS,2
        goto   CTRLA70
        ;goto  A70

; CARACTER ESPECIAL " ) "
A70

        movf   DATO1, w
        xorlw  h'29' ; " ) "
        btfsc  STATUS,2
        goto   CTRLA71
        ;goto  A71

; CARACTER ESPECIAL " # "
A71

        movf   DATO1, w
        xorlw  h'23' ; " # "
        btfsc  STATUS,2
        goto   CTRLA72
        ;goto  A72

; CARACTER ESPECIAL " : "
A72

        movf   DATO1, w
        xorlw  h'3A' ; " : "
        btfsc  STATUS,2
        goto   CTRLA73
        ;goto  A73

; CARACTER ESPECIAL " ; "
A73

```

```
    movf   DAT01, w
    xorlw  h'3B' ; " ; "
    btfsc  STATUS,2
    goto   CTRLA74
```

```
; CARACTER ESPECIAL " . "
```

A74

```
    movf   DAT01, w
    xorlw  h'2E' ; " . "
    btfsc  STATUS,2
    goto   CTRLA75
```

```
; CARACTER ESPECIAL " _ "
```

A75

```
    movf   DAT01, w
    xorlw  h'5F' ; " _ "
    btfsc  STATUS,2
    goto   CTRLA76
```

```
; CARACTER ESPECIAL " ¿ "
```

A76

```
    movf   DAT01, w
    xorlw  .191 ; " ¿ "
    btfsc  STATUS,2
    goto   CTRLA77
```

```
; CARACTER ESPECIAL " ? "
```

A77

```
    movf   DAT01, w
    xorlw  .63 ; " ? "
    btfsc  STATUS,2
    goto   CTRLA78
```

```
; CARACTER ESPECIAL " ¡ "
```

A78

```
    movf   DAT01, w
    xorlw  .161 ; " ¡ "
```



```

        btfsc STATUS,2
        goto CTRLA79

; CARACTER ESPECIAL " ! "
A79
        movf DAT01, w
        xorlw .33 ; " ! "
        btfsc STATUS,2
        goto CTRLA80

; CARACTER ESPECIAL " , "
A80
        movf DAT01, w
        xorlw .44 ; " , "
        btfsc STATUS,2
        goto CTRLA81

; CARACTER ESPECIAL " ESPACIO "
A81
        movf DAT01, w
        xorlw .32 ; " ESPACIO EN BLANCO "
        btfsc STATUS,2
        goto CTRLA82

; CARACTER ESPECIAL '
A82
        movf DAT01, w
        xorlw .39 ; Comilla simple
        btfsc STATUS,2
        goto CTRLA83

; CARACTER ESPECIAL "
A83
        movf DAT01, w
        xorlw .34 ; Comilla doble
        btfsc STATUS,2
        goto CTRLA84

```

```

; CHARACTER ESPECIAL &
A84
    movf   DAT01, w
    xorlw  .38   ; " & "
    btfsc STATUS,2
    goto  CTRLA85

; CHARACTER ESPECIAL $
A85
    movf   DAT01, w
    xorlw  .36   ; " $ "
    btfsc STATUS,2
    goto  CTRLA86

; CHARACTER ESPECIAL %
A86
    movf   DAT01, w
    xorlw  .37   ; " % "
    btfsc STATUS,2
    goto  CTRLA87

; CHARACTER ESPECIAL <
A87
    movf   DAT01, w
    xorlw  .60   ; " < "
    btfsc STATUS,2
    goto  CTRLA88

; CHARACTER ESPECIAL >
A88
    movf   DAT01, w
    xorlw  .62   ; " > "
    btfsc STATUS,2
    goto  CTRLA89

; CHARACTER ESPECIAL =

```

A89

```
movf   DAT01, w
xorlw  .61   ; " = "
btfsc  STATUS,2
goto   CTRLA90
```

; CARACTER ESPECIAL á

A90

```
movf   DAT01, w
xorlw  .225  ; " á "
btfsc  STATUS,2
goto   CTRLA91
```

; CARACTER ESPECIAL é

A91

```
movf   DAT01, w
xorlw  .233  ; " é "
btfsc  STATUS,2
goto   CTRLA92
```

; CARACTER ESPECIAL í

A92

```
movf   DAT01, w
xorlw  .237  ; " í "
btfsc  STATUS,2
goto   CTRLA93
```

; CARACTER ESPECIAL ó

A93

```
movf   DAT01, w
xorlw  .243  ; " ó "
btfsc  STATUS,2
goto   CTRLA94
```

; CARACTER ESPECIAL ú

A94

```
movf   DAT01, w
```

```

        xorlw .250 ; " ú "
        btfsc STATUS,2
        goto CTRLA95

; CARACTER ESPECIAL ü
A95
        movf DAT01, w
        xorlw .252 ; " ü "
        btfsc STATUS,2
        goto CTRLA96

; CARACTER ESPECIAL ñ
A96
        movf DAT01, w
        xorlw .241 ; " ñ "
        btfsc STATUS,2
        goto CTRLA97

; CARACTER ESPECIAL Ñ
A97
        movf DAT01, w
        xorlw .209 ; " Ñ "
        btfsc STATUS,2
        goto CTRLA98

        return

; CONFIGURACION DE CONTROL DEL TECLADO CELULAR

CTRLA0
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "@" DEL CELULAR>
        call letarr
        return

CTRLA1

```

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "A" DEL CELULAR>  
  
    call letmay  
  
    call letraa  
  
    call letmay  
  
    return
```

CTRLA2

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "B" DEL CELULAR>  
  
    call letmay  
  
    call letrab  
  
    call letmay  
  
    return
```

CTRLA3

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "C" DEL CELULAR>  
  
    call letmay  
  
    call letrac  
  
    call letmay  
  
    return
```

CTRLA4

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "D" DEL CELULAR>  
  
    call letmay  
  
    call letrad  
  
    call letmay  
  
    return
```

CTRLA5

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "E" DEL CELULAR>  
  
    call letmay  
  
    call letrae  
  
    call letmay  
  
    return
```

CTRLA6

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "F" DEL CELULAR>  
  
    call letmay
```

```
call letraf
call letmay
return
```

CTRLA7

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "G" DEL CELULAR>
call letmay
call letrag
call letmay
return
```

CTRLA8

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "H" DEL CELULAR>
call letmay
call letrah
call letmay
return
```

CTRLA9

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "I" DEL CELULAR>
call letmay
call letrai
call letmay
return
```

CTRLA10

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "J" DEL CELULAR>
call letmay
call letraj
call letmay
return
```

CTRLA11

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "K" DEL CELULAR>
call letmay
call letrak
call letmay
```

return

CTRLA12

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "L" DEL CELULAR>  
    call letmay  
    call letral  
    call letmay  
    return
```

CTRLA13

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "M" DEL CELULAR>  
    call letmay  
    call lettram  
    call letmay  
    return
```

CTRLA14

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "N" DEL CELULAR>  
    call letmay  
    call lettran  
    call letmay  
    return
```

CTRLA15

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "O" DEL CELULAR>  
    call letmay  
    call letrao  
    call letmay  
    return
```

CTRLA16

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "P" DEL CELULAR>  
    call letmay  
    call lettrap  
    call letmay  
    return
```

CTRLA17

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "Q" DEL CELULAR>  
  
    call letmay  
  
    call letraq  
  
    call letmay  
  
    return
```

CTRLA18

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "R" DEL CELULAR>  
  
    call letmay  
  
    call letrar  
  
    call letmay  
  
    return
```

CTRLA19

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "S" DEL CELULAR>  
  
    call letmay  
  
    call letras  
  
    call letmay  
  
    return
```

CTRLA20

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "T" DEL CELULAR>  
  
    call letmay  
  
    call letrat  
  
    call letmay  
  
    return
```

CTRLA21

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "U" DEL CELULAR>  
  
    call letmay  
  
    call letrau  
  
    call letmay  
  
    return
```

CTRLA22

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "V" DEL CELULAR>
```



```
call letmay
call letrav
call letmay
return
```

CTRLA23

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "W" DEL CELULAR>
call letmay
call letraw
call letmay
return
```

CTRLA24

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "X" DEL CELULAR>
call letmay
call letrax
call letmay
return
```

CTRLA25

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "Y" DEL CELULAR>
call letmay
call letray
call letmay
return
```

CTRLA26

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "Z" DEL CELULAR>
call letmay
call letraz
call letmay
return
```

CTRLA27

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "a" DEL CELULAR>
call letraa
```

return

CTRLA28

; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "b" DEL CELULAR>

call letrab

return

CTRLA29

; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "c" DEL CELULAR>

call letrac

return

CTRLA30

; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "d" DEL CELULAR>

call letrad

return

CTRLA31

; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "e" DEL CELULAR>

call letrae

return

CTRLA32

; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "f" DEL CELULAR>

call letraf

return

CTRLA33

; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "g" DEL CELULAR>

call letrag

return

CTRLA34

; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "h" DEL CELULAR>

call letrah

return

```
CTRLA35
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "i" DEL CELULAR>
    call letrai
    return

CTRLA36
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "j" DEL CELULAR>
    call letraj
    return

CTRLA37
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "k" DEL CELULAR>
    call letrak
    return

CTRLA38
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "l" DEL CELULAR>
    call letral
    return

CTRLA39
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "m" DEL CELULAR>
    call letram
    return

CTRLA40
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "n" DEL CELULAR>
    call letran
    return

CTRLA41
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "o" DEL CELULAR>
    call letrao
    return

CTRLA42
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "p" DEL CELULAR>
```

```

        call letrap
        return

CTRLA43
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "q" DEL CELULAR>
        call letraq
        return

CTRLA44
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "r" DEL CELULAR>
        call letrar
        return

CTRLA45
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "s" DEL CELULAR>
        call letras
        return

CTRLA46
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "t" DEL CELULAR>
        call letrat
        return

CTRLA47
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "u" DEL CELULAR>
        call letrau
        return

CTRLA48
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "v" DEL CELULAR>
        call letrav
        return

CTRLA49
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "w" DEL CELULAR>
        call letraw
        return

```

```
CTRLA50
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "x" DEL CELULAR>
    call letrax
    return

CTRLA51
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "y" DEL CELULAR>
    call letray
    return

CTRLA52
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "z" DEL CELULAR>
    call letraz
    return

CTRLA53
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "0" DEL CELULAR>
    call letra0
    return

CTRLA54
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "1" DEL CELULAR>
    call letra1
    return

CTRLA55
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "2" DEL CELULAR>
    call letra2
    return

CTRLA56
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "3" DEL CELULAR>
    call letra3
    return

CTRLA57
```

```

; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "4" DEL CELULAR>
    call letra4
    return

CTRLA58
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "5" DEL CELULAR>
    call letra5
    return

CTRLA59
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "6" DEL CELULAR>
    call letra6
    return

CTRLA60
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "7" DEL CELULAR>
    call letra7
    return

CTRLA61
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "8" DEL CELULAR>
    call letra8
    return

CTRLA62
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "9" DEL CELULAR>
    call letra9
    return

CTRLA63
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA " <-- " DEL CELULAR>
    ;call borra
;call retardo
    return

CTRLA64
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "DEL" DEL CELULAR>

```

```

        ;call borra
;call retardo
        return

CTRLA65
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "BACKSPACE" DEL CELULAR>
        ;call borra
;call retardo
        return

CTRLA66
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "*" DEL CELULAR>
        call aster
        call retardo
        call caracter_42
        ;call aster
;call retardo
        return

CTRLA67
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "+" DEL CELULAR>
        call aster
        call retardo
        call caracter_43
        ;call aster
;call retardo2
        ;call aster
;call retardo
        return

CTRLA68
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "-" DEL CELULAR>
        call letmen
        return

CTRLA69
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "/" DEL CELULAR>

```

```
call aster
call retardo
    call caracter_47
return
```

CTRLA70

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "(" DEL CELULAR>
call aster
call retardo
    call caracter_40
return
```

CTRLA71

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA ")" DEL CELULAR>
call aster
call retardo
    call caracter_41
return
```

CTRLA72

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "#" DEL CELULAR>
call aster
call retardo
    call caracter_35
return
```

CTRLA73

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA ":" DEL CELULAR>
call aster
call retardo
    call caracter_58
return
```

CTRLA74

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA ";" DEL CELULAR>
call aster
call retardo
```



```

        call caracter_59
        return

CTRLA75
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "." DEL CELULAR>
        call letpun
        return

CTRLA76
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "_" DEL CELULAR>
        call aster
        call retardo
        call letsub
        return

CTRLA77
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "¿" DEL CELULAR>
        call caracter_168
        return

CTRLA78
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "?" DEL CELULAR>
        call caracter_63
        return

CTRLA79
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA ";" DEL CELULAR>
        call caracter_173
        return

CTRLA80
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "!" DEL CELULAR>
        call caracter_33
        return

CTRLA81
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA "," DEL CELULAR>

```

```

        call caracter_44
        return

CTRLA82
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA LA TECLA " " DEL CELULAR>
        call caracter_32
        return

CTRLA83
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER ' DEL CELULAR>
        call aster
        call retardo
        call caracter_39
        return

CTRLA84
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER " DEL CELULAR>
        call aster
        call retardo
        call caracter_34
        return

CTRLA85
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER & DEL CELULAR>
        call aster
        call retardo
        call caracter_38
        return

CTRLA86
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER $ DEL CELULAR>
        call aster
        call retardo
        call caracter_36
        return

CTRLA87

```

```

; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER % DEL CELULAR>
    call aster
    call retardo
        call caracter_37
    return

CTRLA88
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER < DEL CELULAR>
    call aster
    call retardo
        call caracter_60
    return

CTRLA89
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER > DEL CELULAR>
    call aster
    call retardo
        call caracter_62
    return

CTRLA90
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER = DEL CELULAR>
    call aster
    call retardo
        call caracter_61
    return

CTRLA91
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER á DEL CELULAR>
    call caracter_160
    return

CTRLA92
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER é DEL CELULAR>
    call caracter_130
    return

```

CTRLA93

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER Í DEL CELULAR>  
    call caracter_161  
    return
```

CTRLA94

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER Ó DEL CELULAR>  
    call caracter_162  
    return
```

CTRLA95

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER Ú DEL CELULAR>  
    call caracter_163  
    return
```

CTRLA96

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER Û DEL CELULAR>  
    call caracter_129  
    return
```

CTRLA97

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER Ñ DEL CELULAR>  
    call caracter_164  
    return
```

CTRLA98

```
; <SE CONFIGURA LA SEÑAL DE CONTROL QUE ACTIVA EL CARACTER Ñ DEL CELULAR>  
    call letmay  
    call caracter_164  
    call letmay  
    return
```

;Procedimiento "RECONOCER2":

;Similar al procedimiento "RECONOCER" con la diferencia de que solo procesa numeros

;y es el utilizado cuando se pide el ingreso del NUMERO TELEFONICO CELULAR al cual

;se le va a enviar el mensaje:

RECONOCER2

```
; CARACTER " 0 "  
movf    DAT01, w  
        xorlw  h'30' ; " 0 "  
        btfss STATUS,2 ;revisa la ZERO flag.  
        goto   B0  
  
call    num0  
call    retardo2  
  
return
```

```
; CARACTER " 1 "
```

B0

```
movf    DAT01, w  
        xorlw  h'31' ; " 1 "  
        btfss STATUS,2 ;revisa la ZERO flag.  
        goto   B1  
  
call    num1  
call    retardo2  
  
return
```

```
; CARACTER " 2 "
```

B1

```
movf    DAT01, w  
        xorlw  h'32' ; " 2 "  
        btfss STATUS,2 ;revisa la ZERO flag.  
        goto   B2  
  
call    num2  
call    retardo2  
  
return
```

```
; CARACTER " 3 "
```

B2

```
movf    DAT01, w  
        xorlw  h'33' ; " 3 "  
        btfss STATUS,2 ;revisa la ZERO flag.  
        goto   B3
```

```

call    num3
call    retardo2
return

; CARACTER " 4 "
B3
movf    DAT01, w
        xorlw  h'34' ; " 4 "
        btfss STATUS,2 ;revisa la ZERO flag.
        goto  B4
call    num4
call    retardo2
return

; CARACTER " 5 "
B4
movf    DAT01, w
        xorlw  h'35' ; " 5 "
        btfss STATUS,2 ;revisa la ZERO flag.
        goto  B5
call    num5
call    retardo2
return

; CARACTER " 6 "
B5
movf    DAT01, w
        xorlw  h'36' ; " 6 "
        btfss STATUS,2 ;revisa la ZERO flag.
        goto  B6
call    num6
call    retardo2
return

; CARACTER " 7 "
B6
movf    DAT01, w

```

```

        xorlw h'37' ; " 7 "
        btfss STATUS,2 ;revisa la ZERO flag.
        goto B7
    call num7
    call retardo2
    return

; CARACTER " 8 "
B7
    movf DATO1, w
        xorlw h'38' ; " 8 "
        btfss STATUS,2 ;revisa la ZERO flag.
        goto B8
    call num8
    call retardo2
    return

; CARACTER " 9 "
B8
    movf DATO1, w
        xorlw h'39' ; " 9 "
        btfss STATUS,2 ;revisa la ZERO flag.
        goto B9
    call num9
    call retardo2
B9
    return

;***** PROCESSOR DECLARATION & CONFIGURATION *****

;llama_teclado
;        call USART
;        call MENSAJE
;        return

;lazoini
;        call RX_DATO

```

```

;      movwf  datao
;
;      call           IN_DATO
;
;      Bank0
;
;      movf      datao,0
;
;      xorlw  h'5C' ; ends of character " \ "
;
;      btfsc  STATUS,2
;
;      goto           recibido
;
;      call RECONOCER
;
;      ;goto           lazoini

```

```

;recibido

```

```

;      movlw 'R'
;
;      call   TX_DATO
;
;      movlw 'E'
;
;      call   TX_DATO
;
;      movlw 'C'
;
;      call   TX_DATO
;
;      movlw 'I'
;
;      call   TX_DATO
;
;      movlw 'B'
;
;      call   TX_DATO
;
;      movlw 'I'
;
;      call   TX_DATO
;
;      movlw 'D'
;
;      call   TX_DATO
;
;      movlw 'O'
;
;      call   TX_DATO
;
;      call NOTE
;
;      ;goto fin

```

```

;MENSAJE

```

```

;      movlw 'H'
;
;      call           TX_DATO
;
;      movlw 'o'
;
;      call           TX_DATO
;
;      movlw 'l'
;
;      call           TX_DATO

```



```

;         movlw  'a'
;         call   TX_DATO
;         return

```

```

;NOTE

```

```

;         movlw  '@'
;         call   TX_DATO
;         movlw  'E'
;         call   TX_DATO
;         movlw  'C'
;         call   TX_DATO
;         movlw  'O'
;         call   TX_DATO
;         return

```

```

; @@@@ PROCESO DE RECEPCION @@@@

```

```

;RX_DATO

```

```

;         Bank0
;         btfss PIR1,5
;         goto  $-1
;         movf  RCREG,0
;         return

```

```

; </// PROCESO DE TRANSMISION Y SALIDA DE CARACTER ///>

```

```

;TX_DATO

```

```

;         Bank0
;         movwf TXREG
;         Bank1
;         btfss TXSTA,1
;         goto  $-1
;         Bank0
;         return

```

```

;**** CONFIGURACION DE USART ****

;USART

;          Bank1
;          movlw    .25          ; 9600 baud @ 4.0 Mhz Fosc
;          movwf    SPBRG
;          movlw    b'00100100'  ; brgh = 1
;          movwf    TXSTA        ; enable Async Transmission, set brgh
;          Bank0
;          movlw    b'10010000'
;          movwf    RCSTA        ; enable Async Reception
;          movf     RCREG,w
;          movf     RCREG,w
;          movf     RCREG,w      ; flush receive buffer
;          return

```

```

; %%% PROCESO DE ENTRADA DE DATO %%%

```

```

;IN_DATO

;          Bank0
;          movwf    PORTB
;          return

```

```

;SUBROUTINAS RELACIONADAS CON LA MEMORIA SERIAL:

```

```

;=====°

```

```

;Esta rutina genera el start bit para la comunicacion serial

```

```

BSTART

    bsf     STATUS, RP0    ;Seleccion del Banco 1

    movlw  b'00111101'

    movwf  TRISB

    bcf     STATUS, RP0    ;Seleccion del Banco 0

    bcf     PORTB, sclk

    bsf     PORTB, sdata

    nop

```



```

        nop

        nop

        retlw  0

;=====°

;Esta rutina genera el stop bit para la comunicacion serial

BSTOP

        bsf    STATUS, RP0    ;Seleccion del Banco 1

        movlw  b'001111101'

        movwf  TRISB

        bcf    STATUS, RP0    ;Seleccion del Banco 0

        bcf    PORTB, sdata

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        bsf    PORTB, sclk

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        nop

        bsf    PORTB, sdata

        nop

        nop

        nop

        nop

        nop

```

```

nop
nop
nop
nop
nop
nop
bcf    PORTB, sclk
nop
nop
nop
nop
nop
nop
nop
retlw  0

;=====°
;Esta rutina toma el bit que se debe transmitir y lo saca al puerto
BITOUT
    bsf    STATUS, RP0    ;Seleccion del Banco 1
    movlw  b'00111101'
    movwf  TRISB
    bcf    STATUS, RP0    ;Seleccion del Banco 0
    bsf    PORTB, sdata
    nop
    nop
    btfss  eeprom, do
    bcf    PORTB, sdata
clkout
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    bsf    PORTB, sclk
    nop
    nop

```



```

    btfsc txbuf, 7
    bsf    eeprom, do
    call   BITOUT
    rlf    txbuf, 1
    decfsz count, 1
    goto   TXLP
    call   BITIN
    retlw  0

;=====°
;Esta rutina recibe un byte y lo entrega en el registro rxbuf
RX
    clrf   rxbuf
    movlw  .8
    movwf  count
    bcf    STATUS, 0

RXLP
    rlf    rxbuf, F
    call   BITIN
    btfsc  eeprom, di
    bsf    rxbuf, 0
    decfsz count, 1
    goto   RXLP
    bsf    eeprom, do
    call   BITOUT
    retlw  0

;=====°
;Esta rutina recibe la direccion que se desea LEER
;y devuelve el dato que tiene grabado
LEER_EEPROM
    call   BSTART
    nop
    nop
    bcf    slave, 0    ;va a escribir (comando y direccion)
    movf   slave, w
    movwf  txbuf
    call   TX

```



```

movf  highaddr, w ;linea adicional para el caso 24LC256 (antes era el 24LC04).
movwf txbuf      ;linea adicional para el caso 24LC256 (antes era el 24LC04).
call  TX         ;linea adicional para el caso 24LC256 (antes era el 24LC04).

movf  addr, w
movwf txbuf
call  TX
nop
nop
call  BSTART
nop
nop
bsf  slave, 0    ;va a leer.
movf  slave, w
movwf txbuf
call  TX
nop
call  RX
;bsf  eeprom, do
;call BITOUT
nop
nop
nop
call  BSTOP
retlw 0

```

;=====0

;Esta rutina escribe un dato en la posicion de

;memoria que se le indique en el registro addr

ESCRIBIR_EEPROM

```

call  BSTART
nop
nop
bcf  slave, 0
movf  slave, w
movwf txbuf
call  TX

```

```

movf    highaddr, w ;linea adicional para el caso 24LC256 (antes era el 24LC04).
movwf   txbuf       ;linea adicional para el caso 24LC256 (antes era el 24LC04).
call    TX          ;linea adicional para el caso 24LC256 (antes era el 24LC04).

movf    addr, w
movwf   txbuf
call    TX
nop
nop
nop
nop
movf    datao, w
movwf   txbuf
call    TX
call    BSTOP
call    Retardo_4164us
call    Retardo_4164us
call    Retardo_4164us
call    Retardo_4164us
retlw   0

```

Retardo_4164us

;El siguiente lazo realiza un retardo de $208 * 20 + 2 = 4162$ us.

```

movlw   0xd0        ; carga registro w con el valor 208.
movwf   TEMP        ; carga TEMP con valor de w.

```

label24

;Con un cristal de 4 Mhz se ejecuta la instruccion "nop" en 1 us.

```

nop
nop
nop
nop
nop
nop
nop
nop
nop
nop

```

```

nop

nop

nop

nop

nop

nop

nop

nop

nop ;17 nops.

decfsz TEMP, 1 ;toma 1 ciclo, excepto cuando salta si la condicion se cumple.

goto label24 ;toma 2 ciclos.

nop

return

; A CONTINUACION ESTAN LOS PROCEDIMIENTOS QUE PONEN CADA UNO DE LOS CARACTERES
; QUE SE PUEDEN ESCRIBIR EN LOS MENSAJES DEL CELULAR (ANTES ERAN MACROS):

; Mayusculas
letmay
    call numer
    call retardo2
    call retardo2
    return

; Letras ¿, ?, ¡, !, -, ,, .., @, 1
letral
    call num1
    call retardo2

letarr ;caracter arroba.
    call num1
    call retardo2

letpun ;caracter punto.
    call num1
    call retardo2

caracter_44
    call num1

```

```

        call retardo2
letmen
        call num1
        call retardo2
caracter_33
        call num1
        call retardo2
caracter_173
        call num1
        call retardo2
caracter_63
        call num1
        call retardo2
caracter_168
        call num1
        call retardo2
        call num0
        call retardo2
        call borra
        call retardo2
return

;   Letras a, b, c, á, 2
letra2
        call num2
        call retardo2
caracter_160
        call num2
        call retardo2
letrac
        call num2
        call retardo2
letrab
        call num2
        call retardo2
letraa
        call num2

```

```

        call retardo2
        call num0
        call retardo2
        call borra
        call retardo2
    return

;   Letras d, e, f, é, 3
letra3
    call num3
    call retardo2
caracter_130
    call num3
    call retardo2
letraf
    call num3
    call retardo2
letrae
    call num3
    call retardo2
letrad
    call num3
    call retardo2
    call num0
    call retardo2
    call borra
    call retardo2
return

;   Letras g, h, i, í, 4
letra4
    call num4
    call retardo2
caracter_161
    call num4
    call retardo2
letraí

```

```
    call num4
    call retardo2
letrah
    call num4
    call retardo2
letrag
    call num4
    call retardo2
    call num0
    call retardo2
    call borra
    call retardo2
return
```

```
; Letras j, k, l, 5
```

```
letra5
    call num5
    call retardo2
letral
    call num5
    call retardo2
letrak
    call num5
    call retardo2
letraj
    call num5
    call retardo2
    call num0
    call retardo2
    call borra
    call retardo2
return
```

```
; Letras m, n, o, ñ, ó, 6
```

```
letra6
```

```

        call num6
        call retardo2
caracter_162
        call num6
        call retardo2
caracter_164
        call num6
        call retardo2
letrao
        call num6
        call retardo2
letran
        call num6
        call retardo2
letram
        call num6
        call retardo2
        call num0
        call retardo2
        call borra
        call retardo2
return

;   Letras p, q, r, s, 7
letra7
        call num7
        call retardo2
letras
        call num7
        call retardo2
letrar
        call num7
        call retardo2
letraq
        call num7
        call retardo2

```

```

letrap
    call num7
    call retardo2
    call num0
    call retardo2
    call borra
    call retardo2
    return

;   Letras t, u, v, ú, ü, 8
letra8
    call num8
    call retardo2
caracter_129
    call num8
    call retardo2
caracter_163
    call num8
    call retardo2
letrav
    call num8
    call retardo2
letrau
    call num8
    call retardo2
letrat
    call num8
    call retardo2
    call num0
    call retardo2
    call borra
    call retardo2
    return

;   Letras w, x, y, z, 9
letra9

```



```

        call num9
        call retardo2
letraz
        call num9
        call retardo2
letray
        call num9
        call retardo2
letrax
        call num9
        call retardo2
letraw
        call num9
        call retardo2
        call num0
        call retardo2
        call borra
        call retardo2
    return

;   Letras ESPACIO y CERO
letra0
    call num0
    call retardo2
caracter_32
    call num0
    call retardo2
    call num1
    call retardo2
    call borra
    call retardo2
    return

;   Letras : ; + # *
caracter_42
    call abajo

```

```

        call retardo2
caracter_35
        call abajo
        call retardo2
caracter_43
        call abajo
        call retardo2
        call abajo
        call retardo2
caracter_59
        call abajo
        call retardo2
caracter_58
        call abajo
        call retardo2
        call abajo
        call retardo2
        call abajo
        call retardo2
        call abajo
        call retardo2
        call abajo
        call retardo2
        call abajo
        call retardo2
        call menu
        call retardo2
return

;      Letras ( ) _ / ' " & $ % < > =
caracter_40
        call arriba
        call retardo2

```

```
caracter_41
    call arriba
    call retardo2
caracter_39
    call arriba
    call retardo2
caracter_34
    call arriba
    call retardo2
letsub
    call arriba
    call retardo2
caracter_38
    call arriba
    call retardo2
caracter_36
    call arriba
    call retardo2
caracter_37
    call arriba
    call retardo2
caracter_47
    call arriba
    call retardo2
caracter_60
    call arriba
    call retardo2
caracter_62
    call arriba
    call retardo2
caracter_61
    call arriba
    call retardo
    call menu
    call retardo
    return
```

```

;      Letra Arroba @
;letarr
;      call aster
;      call retardo
;      call abajo
;      call retardo
;      call abajo
;      call retardo
;      call menu
;      call retardo
;      return

;      Letra Menos -
;letmen
;      call aster
;      call retardo
;      call abajo
;      call retardo
;      call abajo
;      call retardo
;      call abajo
;      call retardo
;      call abajo
;      call retardo
;      call abajo
;      call retardo
;      call abajo
;      call retardo
;      call abajo
;      call retardo
;      call abajo
;      call retardo
;      call menu
;      call retardo
;      return

;      Letra Punto .
;letpun
;      call aster

```

```
;    call retardo
;    call menu
;    call retardo
;    return
```

```
;=====
```

```
end
```

BIBLIOGRAFÍA.-

Direcciones electrónicas.

<http://www.innovateur.com.mx> Página que contiene herramientas para la elaboración de planes de negocio.

<http://winred.com/EP/recursos/rplandenegocio/010027.html> Pagina que contiene planes de negocio y software de ejemplo.

<http://www.microchip.com> Pagina de microchip la cual contiene descripción de los microcontroladores y memorias seriales.

<http://www.phanderson.com/PIC/PICC/> Pagina con varios ejemplos en lenguaje C para programar ciertos dispositivos.

<http://www.piclist.com/techref/piclist/index.htm> Lista Oficial FAQ para programadores de microcontroladores.

<http://www.doctrionics.co.uk/4016.htm> Configuracion de switches 4016

Libros:

Commercializing New Technologies

Vijay K. Jolly

Harvard Business School Press, 1997

Copyright (c) 1997 by the President and Fellows of Harvard College.