

**MÓDULO DE CONTROL PARA EL DESBLOQUEO DE EQUIPOS DE
MONITOREO VÍA TELEFÓNICA**

LEONARDO AGUDELO DUSSAN

JEAN CARLO SALAZAR CORTES

**UNIVERSIDAD AUTÓNOMA DE OCCIDENTE
FACULTAD DE INGENIERIA
DEPARTAMENTO DE AUTOMÁTICA Y ELECTRÓNICA
PROGRAMA INGENIERÍA ELECTRÓNICA
SANTIAGO DE CALI
2007**

**MÓDULO DE CONTROL PARA EL DESBLOQUEO DE EQUIPOS DE
MONITOREO VÍA TELEFÓNICA**

LEONARDO AGUDELO DUSSAN

JEAN CARLO SALAZAR CORTES

Pasantía para optar al título de Ingeniero Electrónico

**Director
ADRIANA CADAVID SANCHEZ
Ingeniero Electrónico
Especialista en Electromedicina y Gestión Tecnológica Hospitalaria
Docente Hora Cátedra
Departamento de Automática y Electrónica**

**UNIVERSIDAD AUTÓNOMA DE OCCIDENTE
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE AUTOMÁTICA Y ELECTRÓNICA
PROGRAMA INGENIERÍA ELECTRÓNICA
SANTIAGO DE CALI
2007**

Nota de aceptación:

Aprobado por el Comité de Grado en cumplimiento de los requisitos exigidos por la Universidad Autónoma de Occidente para optar al título de Ingeniero Electrónico.

Ing. WILMAN FRANCO

Jurado

Fecha: Santiago de Cali, 16 de Julio de 2007

AGRADECIMIENTOS

Queremos agradecer a Emcali E.I.C.E. E.S.P , especialmente a la planta telefonica de Salomia y a sus ingenieros, en cabeza inicialmente del Ingeniero Ruben Chávez y el Ingeniero Gustavo Montaña y luego del Ingeniero Julio César Osorio por abrirnos las puertas y permitirnos desarrollar este proyecto, que ha sido muy beneficioso y productivo tanto para la empresa misma como para nosotros.

También cabe resaltar la ayuda y apoyo de parte de los demás funcionarios de la planta, y en específico del Ingeniero Luciano Cuellar quien nos colaboró en la etapa de montaje y realización de pruebas del prototipo final.

Del mismo modo queremos agradecer a la Ingeniero Electrónico Adriana Cadavid por dirigir esta pasantia quien nos orientó y dedicó tiempo para la realización de este proyecto.

Muchas Gracias.

CONTENIDO

	pág.
GLOSARIO	13
RESUMEN	17
INTRODUCCIÓN	18
1. SISTEMA DE MONITOREO RED PRIMARIA MODELO 5325B	20
1.1. EQUIPO DE MONITOREO RED PRIMARIA MODELO 5325B	21
2. FUENTE CONMUTADA 52 VDC A 5 VDC Y 2 A	26
2.1. CONVERTIDOR DC-DC	26
2.1.1. Topología Buck	28
2.1.2. Topología Boost	32
2.1.3. Topología FlyBack	36
2.2. ETAPA DE CONTROL	37
2.3. ESPECIFICACIONES	38
2.4. ETAPA DE DISEÑO	39
2.4.1. Fuente Conmutada	39
2.4.2. Etapa de Reinicialización	41
2.5. RESULTADOS FINALES	42
3. IDENTIFICACIÓN DE LLAMADAS USANDO DTMF	45
3.1. DUAL TONE MULTI FREQUENCY (DTMF)	45
3.2. PROTOCOLO DTMF	47
3.3. CIRCUITO DE IDENTIFICACIÓN	48
4. IDENTIFICADOR DE LLAMADAS BINA	50
4.1. FUNCIONES DEL SERVICIO	50
4.2. RESULTADOS FINALES	50
5. EL MODEM	53

5.1. COMO FUNCIONA EL MODEM	53
5.2. VELOCIDADES DE TRANSMISIÓN DEL MODEM	53
5.3. MODEM HANDSHAKING	54
5.4. LIMITACIÓN FÍSICA DE LA VELOCIDAD DE TRANSMISIÓN EN LA LÍNEA TELEFÓNICA	54
5.5. ESTÁNDARES DE CONTROL DE ERRORES	55
5.6. CONTROL DE FLUJO	55
5.7. COMANDOS DE CONTROL DEL MODEM	56
5.8. MODOS DE OPERACIÓN DEL MODEM	56
5.8.1. Formato de comandos Hayes	57
5.9. MODEM: XECOM XE2496	59
5.10. CONEXIÓN RS232 ENTRE EL MICROCONTROLADOR Y MÓDEM	60
5.10.1. Desarrollo de una Conexión a través de Módem	60
5.11. CÓDIGOS DE RESULTADOS	62
5.12. RESULTADOS FINALES	62
6. SISTEMA DE COMUNICACIÓN	66
6.1. MICROCONTROLADOR MICROCHIP PIC 16F877	69
6.2. RESULTADOS FINALES	72
7. PROGRAMA DE MANEJO DEL MÓDULO DE REINICIO	76
7.1. FUNCIONAMIENTO	76
8. SISTEMA FINAL DESARROLLADO	80
9. COSTOS DEL PROTOTIPO	82
10. CONCLUSIONES	84
11. RECOMENDACIONES	85
BIBLIOGRAFÍA	86
ANEXOS	88

LISTA DE TABLAS

	pág.
Tabla 1. Valores de frecuencias por dígito	44
Tabla 2. Valor decimal y binario correspondiente a la frecuencia DTMF	44
Tabla 3. Características del protocolo DTMF	46
Tabla 4. Estándares de modulación	53
Tabla 5. Secuencia de conexión entre módems	60
Tabla 6. Respuestas del módem	61
Tabla 7. Costos del prototipo	82

LISTA DE FIGURAS

	pág.
Figura 1. Diagrama de instalación	20
Figura 2. Estado del sistema	21
Figura 3. Equipo de monitoreo red primaria	21
Figura 4. Equipo de monitoreo red primaria	22
Figura 5. Tarjeta direccionable	23
Figura 6. Tarjeta multiplexora	23
Figura 7. Tarjeta fuente de alimentación	24
Figura 8. Arquitectura Direccionable	24
Figura 9. Conversor DC-DC más simple, y forma de onda V_{Ro}	26
Figura 10. Filtro de salida inductivo. Filtro de Salida capacitivo e inductivo	27
Figura 11. Convertidor Buck	28
Figura 12. (a) Circuito equivalente durante $0-\delta T$, y (b) durante $\delta T-T$	29
Figura 13. Forma de ondas principales, modo de conducción continua	29
Figura 14. Circuito Buck y bloque PWM	30
Figura 15. (a) Circuito equivalente en $0 < t < dT$. (b) Circuito Equivalente en $dT < t < qT$. (c) Circuito equivalente en $qT < t < T$	31
Figura 16. Forma de ondas relevantes, modo de Conducción discontinua	31
Figura 17. Conversor Boost, elevador o Step Up	33
Figura 18. a) circuito equivalente cuando S Conduce ($0 < t < \delta T$). b) circuito equivalente cuando S bloquea ($\delta T < t < T$)	33
Figura 19. Principales formas de onda en MCC	33
Figura 20. (a) Intervalo $0 < t < \delta T$. (b) Intervalo $\delta T < t < \theta T$. (c) Intervalo $\theta T < t < T$	35

Figura 21. Formas de Onda relevantes en MDC	35
Figura 22. Topología de Retroceso o FlyBack.	37
Figura 23. Bloques componentes de un Buck.	38
Figura 24. Generación de la señal PWM	38
Figura 25. Diseño base de un Back	39
Figura 26. Configuración interna del TIP 147	40
Figura 27. Circuito de reinicio en los -48 Vdc	41
Figura 28. Montaje de la fuente en protoboard	42
Figura 29. Circuito Impreso, vista de pistas	43
Figura 30. Circuito Impreso, vista de componentes	43
Figura 31. Circuito Impreso Terminado.	44
Figura 32. Fuente Terminada	44
Figura 33. Identificación del tono DTMF con CM 8870	46
Figura 34. Transmisión del tono DTMF	46
Figura 35. Protocolo DTMF	47
Figura 36. Intervalo de tiempo de los tonos DTMF	48
Figura 37. Circuito de identificación de tonos DTMF	49
Figura 38. Circuito identificador de llamadas	51
Figura 39. Aislamiento Galvánico en entrada del identificador	52
Figura 40. Montaje de la etapa de aislamiento del identificador	52
Figura 41. Modo de funcionamiento del módem	57
Figura 42. Configuración de pines en el módem	59
Figura 43. Diagrama de conexión entre el PIC y el módem	60
Figura 44. Circuito para el control y configuración del módem	63
Figura 45. Simulación del microcontrolador con el MODEM Físico a través del puerto serie	64
Figura 46. Esquema general del modulo de reinicio	66
Figura 47. Configuración de pines del PIC 16F877	68
Figura 48. Circuito central del modulo de reinicio	70

Figura 49. Conmutación de la línea telefónica	71
Figura 50. Circuito central de comunicación	72
Figura 51. Circuito impreso del circuito central	73
Figura 52. Circuito impreso del circuito central	73
Figura 53. Conmutación de la línea telefónica.	74
Figura 54. Circuito impreso de conmutación de la línea telefónica	74
Figura 55. Pantalla de bienvenida	76
Figura 56. Menú principal del programa	77
Figura 57. Configuración del puerto serie	77
Figura 58. Directorio telefónico de las centrales	78
Figura 59. Números telefónicos almacenados en el PIC	79
Figura 60. Prototipo desarrollado	80
Figura 61. Prototipo identificando una llamada	81
Figura 62. Prototipo en funcionamiento	81

LISTA DE ANEXOS

	pág.
Anexo A. Sistema de monitoreo red primaria modelo 5325b	89
Anexo B. Código fuente programado en el microcontrolador en PIC C	93
Anexo C. Código fuente del programa en Visual Basic	112
Anexo D. Datasheets	142

LISTA DE ECUACIONES

	pág.
Ecuación 1. Almacenamiento de Energía Intervalo $0 < t < \delta T$	29
Ecuación 2. Almacenamiento de Energía Intervalo $\delta T < t < T$	30
Ecuación 3. Corriente media entregada por la fuente	32
Ecuación 4. Ciclo de trabajo de la fuente	33
Ecuación 5. Almacenamiento de energía Intervalo $0 < t < dT$	34
Ecuación 6. Almacenamiento de energía Intervalo Intervalo $dT < t < T$	35
Ecuación 7. Ciclo de trabajo de la fuente	35
Ecuación 8. Transferencia de tensión entre la entrada y la salida	37
Ecuación 9. Cálculo de R_f	40
Ecuación 10. Cálculo de R_t	40
Ecuación 11. Cálculo de R_t	41
Ecuación 12. Cálculo de la bobina L	41
Ecuación 13. Cálculo del condensador de salida	41
Ecuación 14. Cálculo del número de vueltas al toroide	42
Ecuación 15. Cálculo del número de vueltas al núcleo	42

GLOSARIO

AISLAMIENTO GALVÁNICO: es el aislamiento en el que se utilizan transformadores para aislar magnéticamente, es muy usado para aislar tierras que tienen diferencias de potencial.

ASCII: American Standard Code for Information Interchange (Código Estadounidense Estándar para el Intercambio de Información) es un código de caracteres basado en el alfabeto latino tal como se usa en inglés moderno y otras lenguas occidentales. Creado aproximadamente en 1963 por el Comité Estadounidense de Estándares (ASA)

BAUDIOS: unidad de medida que determina el número de cambios de estados en una señal por segundo. En la transmisión de datos, baudio es la cantidad de veces que cambia el estado del medio de transmisión por segundo. Cada cambio de estado afecta a más de un bit de información, por lo tanto la tasa de bits puede ser superior a la tasa de baudios.

BIT: acrónimo de Binary Digit (dígito binario). Un bit es una señal electrónica que puede estar encendida (1) o apagada (0). Es la unidad más pequeña de información que utiliza un ordenador. Son necesarios 8 bits para crear un byte. La mayoría de las veces los bits se utilizan para describir velocidades de transmisión, mientras que los bytes se utilizan para describir capacidad de almacenamiento o memoria.

BINA: sistema de identificación de número llamante que usa DTMF, La sigla BINA, que viene de: "B" Identifica el Numero de "A", donde "B" es el teléfono que esta siendo llamado y "A" quien esta llamando, empleado en equipamientos mundialmente conocidos como CALLER-Id.

BLOQUEO DE TARJETAS: es el problema que se presentan en las tarjetas SPARTON cuando su memoria RAM se ha llenado completamente y se empiezan a reescribir espacios de memoria que contienen información que se va a usar en el proceso, debido a esto la tarjeta se bloquea.

BYTE: también llamado "octeto", es el conjunto de 8 bits que se necesitan para codificar un carácter, al ser un valor muy pequeño, se utilizan sus múltiplos: kilobyte, megabyte, gigabyte.

BOOST: en este convertidor siempre se cumple que el voltaje de salida es mayor que el voltaje de entrada.

BUCK: en este convertidor siempre se cumple que el voltaje de salida es menor o igual que el voltaje de entrada, razón por la que se le llama Reductor.

CENTRAL AXE: es una central digital telefónica fabricada por Siemens.

CENTRAL EWSD: es una central digital telefónica fabricada por Ericsson.

CENTRAL DE MONITOREO: es la oficina telefónica donde se realizan los procesos de monitorización de los cables presurizados.

COMANDOS HAYES: es una cadena ASCII que se envía al módem para que este realice alguna acción o configure alguno de sus parámetros. Esta comienza con los caracteres 'AT', seguida de nombres de comando y sus respectivos parámetros, y que concluye con un carácter de retorno de carro.

CONVERTIDOR DC/DC: son circuitos de potencia donde los semiconductores, o llaves de potencia, conmutan a una frecuencia mucho mayor que la de variación de las formas de onda de entrada y salida del conversor. Esto permite emplear filtros pasa-bajos para eliminar los componentes armónicos no deseados.

CPU RISC: es una filosofía de diseño de CPU para que está a favor de conjuntos de instrucciones pequeños y simples que toman menor tiempo para ejecutarse. En arquitectura computacional, RISC del inglés Reduced Instruction Set Computer (Computadora con Conjunto de Instrucciones Reducido). Tipo de microprocesadores con las siguientes características fundamentales:

- ✓ Instrucciones de tamaños fijos y presentados en un reducido número de formatos.
- ✓ Sólo las instrucciones de carga y almacenamiento acceden a la memoria por datos.

Además estos microprocesadores suelen disponer de muchos registros de propósito general.

DCE: es el Equipo de Terminación del Circuito de Datos.

DTE: es el Equipo Terminal de Datos.

DTMF: Dual Tone Multi Frequency usa dos tonos para representar cada dígito en el teclado. Cuando una tecla se presiona el tono de la columna y el tono de la fila son generados, por lo tanto es tono dual. Por ejemplo, al presionar el botón ' 5 ' se generan los tonos 770 Hz y 1336 Hz.

FLUJO: en física e ingeniería, caudal es la cantidad de fluido que circula por unidad de tiempo en determinado sistema o elemento. Se expresa en la unidad de volumen dividida por la unidad de tiempo.

FLYBACK: es un convertidor DC a DC con aislamiento galvánico entre la entrada y la salida. Tiene la misma estructura que un convertidor Boost con un transformador en lugar de un inductor. Gracias a ello se pueden alcanzar altos ratios de conversión. Debido a las limitaciones intrínsecas este convertidor solo se usa en aplicaciones de baja potencia.

HANDSHAKING: técnica de sincronización de comunicaciones entre dos terminales de datos; el procedimiento se ejecuta cuando se establece una conexión entre dos dispositivos de comunicaciones de datos, antes de cualquier transferencia de datos; la terminal de llamada verifica que se haya establecido una comunicación satisfactoria con la terminal llamada.

HUMEDAD: La humedad es la cantidad de vapor de agua presente en el aire. Se puede expresar de forma absoluta mediante la humedad absoluta, o de forma relativa mediante la humedad relativa o el grado de humedad.

MICROCONTROLADOR: es un circuito integrado o chip que incluye en su interior las tres unidades funcionales de una computadora: CPU, Memoria y Unidades de E/S, es decir, se trata de un computador completo en un solo circuito integrado. Aunque sus prestaciones son limitadas, además de dicha integración, su característica principal es su alto nivel de especialización.

MÓDEM: es un acrónimo de Modulador-Demodulador; es decir, que es un dispositivo que transforma las señales digitales del ordenador en señal telefónica analógica y viceversa, con lo que permite al ordenador transmitir y recibir información por la línea telefónica.

PAR DE COBRE: el bucle de abonado es el par de cobre que conecta el terminal telefónico del usuario con la central local de la que depende. El bucle de abonado proporciona el medio físico por medio del cuál el usuario accede a la red telefónica y por tanto recibe el servicio telefónico.

PIC: son los circuitos integrados de Microchip Technology Inc., que pertenece a la categoría de los microcontroladores es decir, aquellos componentes que integran en un único dispositivo todos los circuitos necesarios para realizar un completo sistema digital programable.

PRESIÓN: en física y disciplinas afines la presión, también llamada presión absoluta en aquellos casos que es necesario evitar interpretaciones ambiguas, se

define como la fuerza por unidad de superficie, su unidad se denomina PSI que quiere decir Libra por pulgada cuadrada en el sistema métrico inglés.

PWM: es una señal cuadrada que se modula por ancho de pulso, esto quiere decir que en un período la duración en alto de esta señal varía, a esta variación se le denomina ciclo de trabajo de la señal PWM.

RED TELEFÓNICA PRIMARIA: Es la conexión física de cables telefónicos que unen el bastidor con los armarios.

RS232: también conocido como EIA RS-232C, designa una norma para el intercambio serie de datos binarios entre un DTE (Equipo Terminal de Datos) y un DCE (Equipo de Terminación del Circuito de Datos), aunque existen otras situaciones en las que también se utiliza el interfaz RS-232.

SEÑAL RAMPA: es una señal cuya amplitud empieza desde cero y luego aumenta constantemente hasta que al final del período vuelve a ser cero. Esta señal se denomina también diente de sierra, con una pendiente m.

TRANSDUCTOR: Dispositivo que puede detectar cambios de presión, flujo, temperatura, humedad a través de un sensor y a su vez generar una señal eléctrica medible.

THROUGHPUT: Define la cantidad de datos que pueden enviarse a través de un módem en un cierto período de tiempo. Un módem de 9600 baudios puede tener un throughput distinto de 9600 BPS debido al ruido de la línea (que puede ralentizar) o a la compresión de datos (que puede incrementar la velocidad hasta 4 veces el valor de los baudios). Para mejorar la tasa efectiva de transmisión o throughput se utilizan técnicas de compresión de datos y corrección de errores.

RESUMEN

Este documento muestra el proceso realizado para supervisar remotamente el estado de los cables telefónicos por medio del módulo de control para el desbloqueo de los equipos de monitoreo sparton serie 5325B vía telefónica; empezando por el funcionamiento y características del equipo utilizado por EMCALI para la realización de monitoreo de la red telefónica primaria. Entender cuál es el proceso que desarrollan y la importancia de este para el mantenimiento preventivo de las líneas telefónicas y garantizar el servicio.

El primer capítulo trata de dar a conocer al público el funcionamiento y características del equipo utilizado por EMCALI para la realización de monitoreo de la red telefónica primaria. Entender cuál es el proceso que desarrollan y la importancia de este para el mantenimiento preventivo de las líneas telefónicas y garantizar el servicio.

Posteriormente se realiza el estudio de la fuente de alimentación para nuestro prototipo, cuáles fueron las necesidades planteadas para nuestro diseño. Explica cuál fue el método de conversión de voltaje recomendado y los resultados obtenidos en la elaboración del dispositivo.

En el capítulo siguiente se comenta acerca de los tonos DTMF, cómo se genera, en qué frecuencias se sitúan, cómo se transmiten por la línea telefónica, cómo está desarrollado el protocolo de comunicaciones BINA utilizado por EMCALI para la identificación de llamadas. Con dicha información desarrollamos un circuito capaz de identificar el número telefónico que hace una llamada telefónica, circuito que forma una parte importante en el prototipo desarrollado.

En la siguiente parte del trabajo la investigación se centra en el MODEM, dando a conocer sus características y su importancia en la elaboración del prototipo. Mediante este elemento se establece una comunicación RS232 con el microcontrolador para realizar una transferencia de información por la línea telefónica con un equipo remoto, por medio de esta comunicación se realizan procesos de configuración en el prototipo, esta se destaca por ser una de las funciones más importantes del dispositivo final.

En los capítulos finales se ilustra el diseño final con base a la información y necesidades planteadas del prototipo, elementos empleados en la elaboración, las dificultades que se presentaron. Además se proporciona información del software programado para realizar los procesos de configuración con el prototipo.

INTRODUCCIÓN

El monitoreo de la red telefónica es un sistema automático de control que se encarga de supervisar remotamente el estado de los cables telefónicos en cuanto a presión, flujo y humedad para prevenir daños masivos.

Lo que ocurre en estos momentos es el bloqueo de las tarjetas SPARTON para el monitoreo de las redes, cuando estas están en funcionamiento; este bloqueo consiste en dos fallos: a) cuando la tarjeta es llamada por el gestor y esta no responde y, b) cuando después de hacer la gestión en el equipo este no da por terminada la llamada es decir que no cuelga.

En la actualidad cuando suceden estos tipos de inconvenientes la única forma de solucionarlo es desplazarse hasta la ubicación del equipo y reinicializarlo manualmente; esto presenta el inconveniente de tener que ir hasta el sitio donde está ubicado el equipo, perdiendo tiempo que se puede usar en hacer mantenimiento de otros cables y; que si la tarjeta no cuelga la llamada el gestor no puede continuar su trabajo con las demás tarjetas ya que tiene bloqueado su sistema.

Lo que se busca es hacer este proceso desde el mismo sitio donde está ubicado el monitor de la red, es decir, llevar a cabo remotamente la reinicialización de las tarjetas bloqueadas y así ahorrar en tiempo y en desplazamientos de operarios hasta el sitio donde se encuentra ubicado el equipo.

Teniendo en cuenta que existen un gran número de tarjetas de monitoreo ubicadas por toda la ciudad comunicándose con la central a través de una línea telefónica y algunas tarjetas no logran comunicarse por fallos en la comunicación de las mismas o por el estado de antigüedad de los equipos SPARTON 5325B, el sistema suele bloquearse y por consiguiente pierde información del estado de las líneas telefónicas, lo que conlleva a realizar un desplazamiento de personal al sitio donde se encuentran ubicadas las tarjetas para realizar manualmente el reinicio del equipo, aunque este se encuentre en sitios muy alejados por ejemplo Jamundí. Esto ocasiona a la empresa pérdidas económicas, retraso en actividades normales de trabajo ocasionado por los desplazamientos (tiempo) y perdida en el control del estado en las líneas telefónicas. Para solucionar estos inconvenientes se requiere crear un equipo electrónico capaz de realizar la reinicialización de las tarjetas de monitoreo en forma remota y así establecer una comunicación exclusiva con el modulo y la central de monitoreo cuando el administrador de la red lo requiera.

Nuestro objetivo principal es el Diseño e implementación de un dispositivo que permita la reinicialización de los equipos de monitoreo 5325B de forma remota,

actuando sobre estos a través de una línea telefónica; para lograrlo nos propusimos los siguientes objetivos específicos:

- ∇ Elaborar un método de comunicación entre la central de monitoreo y el prototipo a diseñar a través de la línea telefónica para realizar el proceso de desbloqueo de los equipos de monitoreo.
- ∇ Diseñar y desarrollar el sistema de alimentación del prototipo.
- ∇ Evaluar y seleccionar componentes adecuados para el prototipo.
- ∇ Diseñar e implementar el sistema de control de reinicialización en las tarjeta Spartan 5325B.
- ∇ Realizar las pruebas y verificación de los resultados obtenidos del prototipo.

Para realizar este propósito la investigación y aprendizaje sobre el funcionamiento de las líneas telefónicas y de los medios de transmisión sobre estas forman parte fundamental en la realización del proyecto, corrigiendo las dificultades en transmisión de datos por un par telefónico, logramos de esta forma abrir un canal de comunicación para nuestro propósito.

Las limitaciones de hardware determinan cambios importantes en el diseño final, siendo solucionadas por estrategias diferentes en el modo de operación del microprocesador y la forma de alimentación de voltaje. Todo para lograr como resultado los objetivos planteados.

1. SISTEMA DE MONITOREO RED PRIMARIA MODELO 5325B

El monitoreo de la red es un sistema automático de control que se encarga de supervisar remotamente el estado de los cables telefónicos en cuanto a presión, flujo y humedad para prevenir daños masivos; el realizar un monitoreo constante las 24 horas del día ofrece las siguientes ventajas:

- Visualizar el estado de los cables en cada central.
- Programar el mantenimiento y tomar decisiones según las prioridades.
- Facilitar la localización de fugas neumáticas.
- Incrementar la productividad de los grupos de mantenimiento preventivo.
- Disminuir los costos de mantenimiento correctivo.
- Establecer criterios para determinar la reposición de redes por envejecimiento.
- Permitir en algunos casos determinar el sitio de corte de un cable por vandalismo.

Figura 1. Diagrama de instalación.

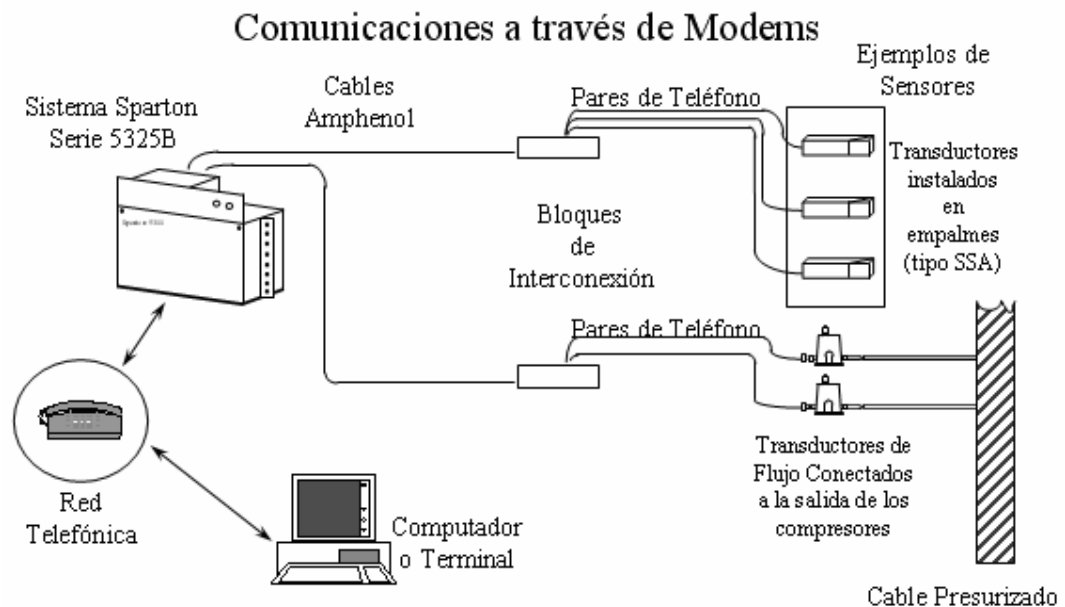
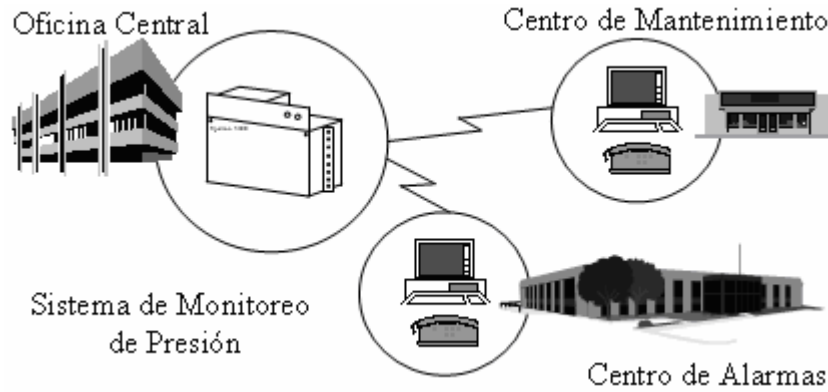


Figura 2. Estado del sistema.



1.1. EQUIPO DE MONITOREO RED PRIMARIA MODELO 5325B

Los sistemas 5325 son sistemas a base de microprocesadores que monitorean, miden y reportan el estado individual de transductores de presión y de flujo, contactores y módulos direccionables de alarmas de contacto instalados en pares dedicados direccionables. Los sistemas Sparton 5325 permiten a través de programación hacer que el sistema imprima la información en cualquier orden deseado sin importar en qué entrada está el transductor físicamente conectado.

Figura 3. Equipo de monitoreo red primaria



Figura 4. Equipo de monitoreo red primaria



Un sistema 5325 básico consiste de un chasis, fuente de alimentación, control, comunicaciones, conversión analógica a digital, almacenaje electrónico de datos, módem.

Características Control Principal:

- Memoria PROM de 512 KB.
- Memoria RAM de 512 KB.
- Memoria RAM de respaldo de 512 KB.
- Comunicación con Módem 2400 Baudios.
- Comunicación Directa 2400 Baudios.
- Respaldo con Batería Interna para la Memoria RAM.

Modulo Direccional:

- Ejecuta las funciones de direccionar y leer los transductores direccionables.
- Módulo control a 128 Transductores por cada una de las entradas a este módulo.
- 4 ENTRADAS - $128 \times 4 = 512$ Transductores.
- 24 ENTRADAS - $128 \times 24 = 3072$ Transductores.

Figura 5. Tarjeta direccionable.



Tarjeta multiplexora:

- Es la que convierte el módulo direccionable de 4 entradas en uno de 24 entradas, multiplexando por 6 para mayor capacidad.
- Esta se utiliza para las centrales con mayor cantidad de cables.
- El módulo de 24 entradas se obtiene reemplazando la tarjeta puente por la tarjeta direccionable.

Figura 6. Tarjeta multiplexora.



Tarjeta fuente de alimentación:

- Se alimenta con -48VDC. Voltaje Filtrado de las Baterías (Rectificadores).
- Produce + 5 VDC
+12 VDC
-12 VDC

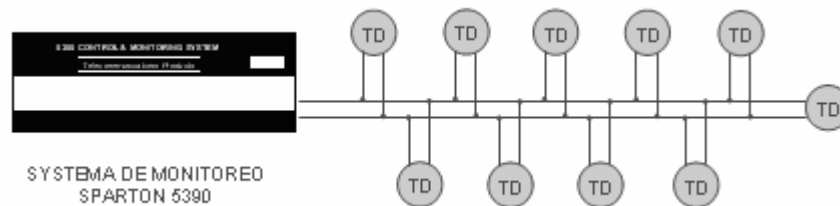
Figura 7. Tarjeta fuente de alimentación.



Tecnología direccionable:

Se puede conectar 128 transductores en un par de alambre de cobre.

Figura 8. Arquitectura Direccionable.



- Los Transductores pueden ser instalados en un sólo par hasta una distancia de 90 Millas o 145 Km.
- Tecnología de Estado Sólido provee:
 - Más Confiabilidad
 - Más Exactitud de Medición (1%).
- Inversión mínima de Cobre.

El código binario del Transductor a medirse es transmitido por el sistema de Monitoreo a través del par, se envía como una señal cuadrada de baja frecuencia, El transductor con el código binario correspondiente responde transmitiendo con una señal de 20 Hz para 9.5 PSI, hasta 40 Hz para 0.0 PSI.

2. FUENTE CONMUTADA 52 VDC A 5 VDC Y 2 A

2.1. CONVERTIDOR DC/DC

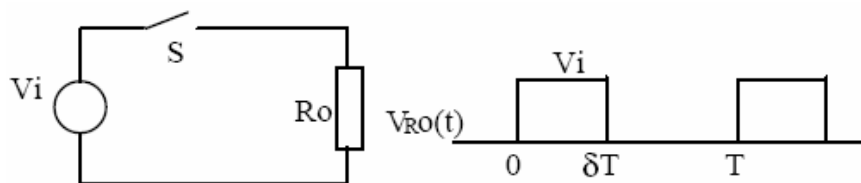
Es de mucha importancia para la electrónica de potencia el estudio y desarrollo de los convertidores conmutados o fuentes conmutadas.

Los conversores conmutados en alta frecuencia son circuitos de potencia donde los semiconductores, o llaves de potencia, conmutan a una frecuencia mucho mayor que la de variación de las formas de onda de entrada y salida del convertidor. Esto permite emplear filtros pasa-bajos para eliminar los componentes armónicos no deseados. No podemos usar la inversión de la forma de onda de entrada para bloquear los semiconductores, como se hace con los conmutados por la red. Los interruptores deberán entonces controlarse al conducir y bloquear.

Los conversores DC-DC forman parte de este grupo de convertidores, controlan el flujo de energía entre dos sistemas de continua. Ejemplos de aplicación son: fuentes de alimentación CC (Switch Mode Power Supplies, SMPS) para equipamiento electrónico, control de máquinas eléctricas de corriente continua, etc.

La Figura 9.(a) presenta la topología más simple para un convertidor DCDC, una carga resistiva se alimenta desde una fuente de tensión CC (V_i), interruptor S conmuta a una frecuencia $f=1/T$.

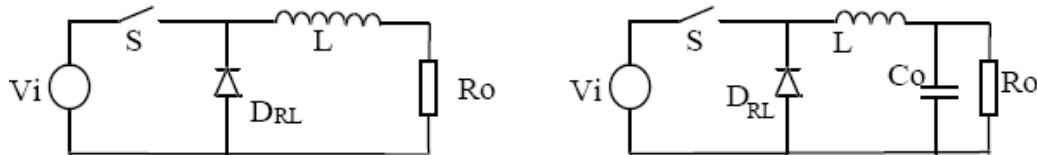
Figura 9. Conversor DC-DC más simple, y forma de onda V_{Ro} .



La tensión en la resistencia R_o será rectangular, Figura 9(b). Su valor medio es: $V_{Ro} = \delta V_i$ donde δ se llama ciclo de trabajo (duty cycle), y es la fracción de tiempo que conduce el interruptor S. Se observa que la tensión y corriente en la carga presentan un alto contenido armónico. Pocas aplicaciones aceptan una fuente con esas características.

Para obtener corriente y tensión en la carga constantes es necesario agregar un filtro, Figura 10. Encontramos así una de las topología básicas, el convertor BUCK, o REDUCTOR o STEP-DOWN.

Figura 10. Filtro de salida inductivo. Filtro de salida capacitivo e inductivo.



Los convertidores conmutados de continua a continua se dividen en diferentes topologías así:

- Reductora o modelo Buck.
- Elevadora o modelo Boost.
- Y en tercer lugar un híbrido de las dos anteriores el Flyback,

El funcionamiento básico de las fuentes conmutadas consiste en el almacenamiento temporal de energía y la cesión de esta en un período de tiempo; este período de tiempo determinará la cantidad de energía entregada a la carga.

Los conmutadores se dividen en tres bloques:

- Conmutación: este bloque se encarga de fraccionar la señal de entrada según la frecuencia y el ciclo de trabajo que se le quiera dar al elemento conmutador.
- Elemento de acumulación de energía: este elemento está ligado al primer bloque, ya que ahí se determina cuándo será liberada la energía hacia la carga.
- Filtrado de la señal: este es el bloque que filtra la señal conmutada a alta frecuencia.

Las ventajas de las fuentes conmutadas frente a las fuentes lineales son:

- Su rendimiento tiene márgenes del 60% y el 90% contra el 14% de las fuentes de alimentación lineales.
- Algunas veces se suelen utilizar como limitadores de terceros armónicos.
- Tienen pequeñas dimensiones.

En desventajas tenemos:

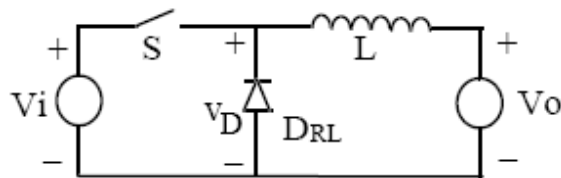
- Generación de EMI (emisión electromagnética), tanto conducida como radiada.
- Aumento de las pérdidas cuando la frecuencia de conmutación es alta.
- Altos picos de corriente inicial.

Esta topología de fuentes conmutadas se utiliza en el campo de los microprocesadores, que cada vez son más rápidos, de tal manera que se necesitan corrientes más elevadas y tensiones más bajas.

Debido a su alto rendimiento y que sus componentes pueden soportar temperaturas elevadas ($\sim 80^\circ\text{C}$), las hacen idóneas para la reducción de tensión y para el aumento de corriente.

2.1.1. Topología Buck. Su principio básico se centra en la reducción de la tensión de entrada, mediante una frecuencia de conmutación en el elemento conmutador, y un ciclo de trabajo que nos determinará el porcentaje de señal de entrada que representaremos a la salida, en la figura 11 se representa un diagrama circuital de esta topología:

Figura 11. Convertidor Buck



✓ Modo de conducción continua – CCM. En $t=0$ comienza a conducir el interruptor S, el circuito equivalente de esta etapa está representado en la Figura 12(a). Como la tensión de salida V_o es menor que V_i la corriente por L será creciente durante este intervalo. La corriente que circula por S es igual a la de L.

Un tiempo δT después se apaga el interruptor S. Se genera entonces una sobretensión que hace conducir al diodo DRL (diodo de rueda libre) manteniendo así la continuidad de la corriente por L. El nuevo circuito está representado en la Figura 12(b).

La corriente por L es ahora decreciente.

La Figura 13 presenta las formas de onda principales.

Figura 12. (a) Circuito equivalente durante $0-\delta T$, y (b) durante $\delta T-T$

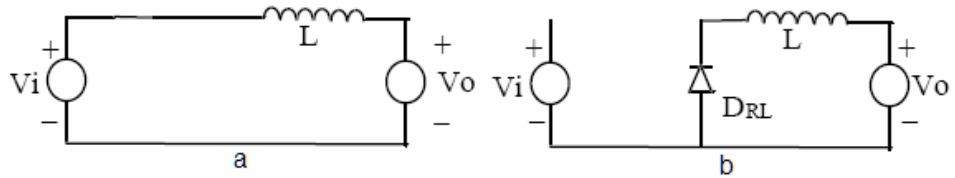
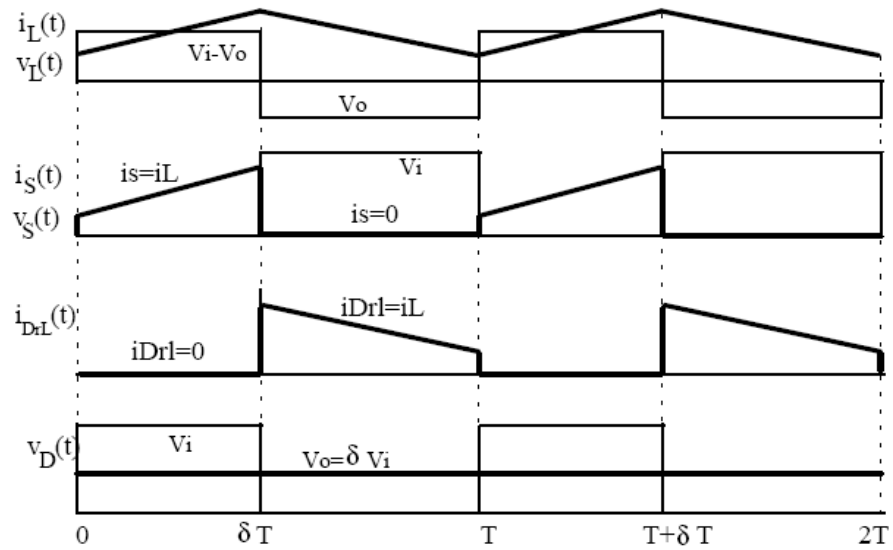


Figura 13. Forma de ondas principales, modo de conducción continua



Se verifican la siguientes ecuaciones:
Intervalo $0 < t < \delta T$:

$$\frac{di_L}{dt} = \frac{V_i - V_o}{L}$$

$$i_S = i_L$$

$$i_D = 0$$

$$v_D = V_i$$

$$v_L = V_i - V_o \quad (\text{Ecu. 1. Almacenamiento de Energía Intervalo } 0 < t < \delta T)$$

Intervalo $\delta T < t < T$:

$$\frac{di_L}{dt} = \frac{-V_o}{L}$$

$$i_D = i_L$$

$$i_s = 0$$

$$v_D = 0$$

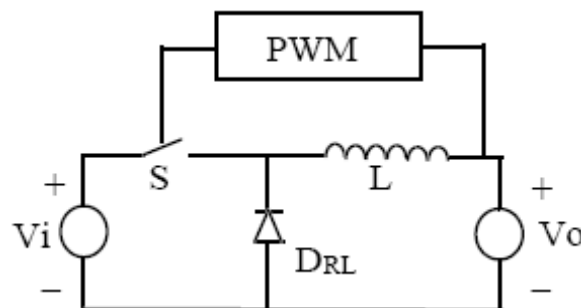
$$v_L = -V_o \quad (\text{Ecu. 2. Almacenamiento de Energía Intervalo } \delta T < t < T)$$

El valor medio de la tensión v_D es dV_i y, como la tensión media sobre L debe ser nula cuando el circuito opera en estado estacionario, tenemos que: $V_o = dV_i$

La tensión de salida V_o puede ser entonces controlada variando el ciclo de trabajo (d) del interruptor S , para compensar las variaciones de la tensión de entrada V_i .

La Figura 14 representa al Buck junto con un bloque de control, denominado PWM (Pulse Width Modulator, modulador de ancho de pulso). Este bloque se encarga de calcular el ancho de pulso δT de conducción del interruptor, necesario para obtener una dada tensión de salida V_o , y enviarlo al interruptor S .

Figura 14. Circuito Buck y bloque PWM



✓ Modo de conducción discontinua – MCD. El comportamiento descrito anteriormente se basa en asumir que la corriente por L no se anula durante el intervalo en que conduce el diodo de rueda libre. Esta hipótesis no se cumple si la inductancia L es suficientemente “chica”, o si la corriente entregada por la fuente de salida V_o a la carga es muy “chica”. Supondremos ahora que la corriente i_L se anula a partir de algún instante del intervalo $dT-T$, hasta $t=T$.

En $t=0$ comienza a conducir el interruptor S , el circuito resultante está representado en la Figura 15(a) (es el mismo de la Figura 12(a)). El valor inicial de la corriente i_L es cero.

Un tiempo δT después se apaga el interruptor S, y conduce el diodo de rueda libre. El nuevo circuito está representado en la Figura 15(b) (es el mismo de la Figura 12(b)).

En el instante $t=\theta T$ la corriente i_L se anula, el circuito equivalente está representado en la Figura 15(c).

La Figura 16 representa las nuevas formas de onda.

Figura 15. (a) Circuito equivalente en $0 < t < dT$. (b) Circuito equivalente en $dT < t < qT$. (c) Circuito equivalente en $qT < t < T$.

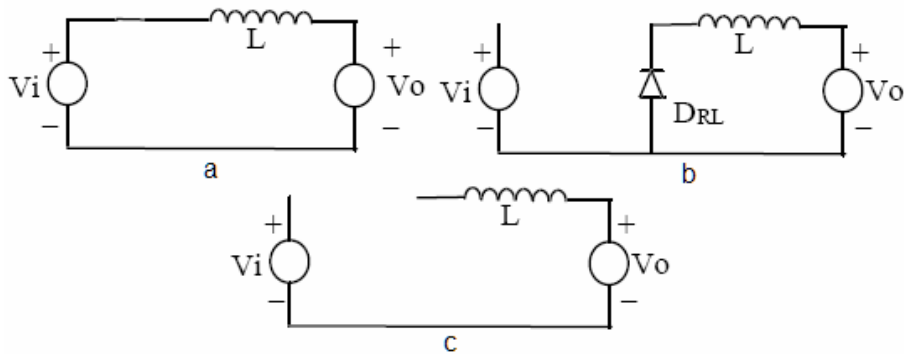
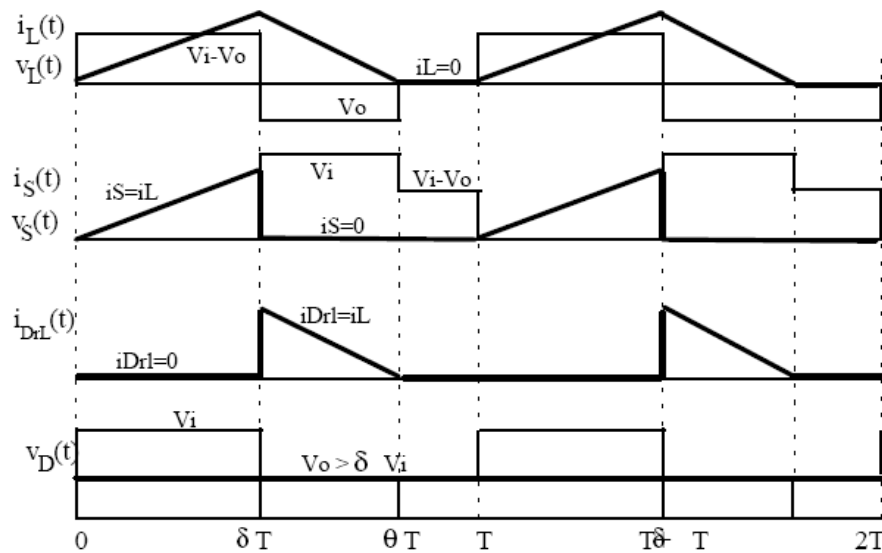


Figura 16. Forma de ondas relevantes, modo de conducción discontinua.



Se observa de la forma de onda de v_D en la Figura 16, que en el modo de conducción discontinua la tensión de salida V_o (igual al valor medio de v_D) es

mayor que en el modo de conducción continua (δV_i), si se considera el mismo ciclo de trabajo δ . Se cumple: $\delta V_i = \theta V_o$

Sea I_o la corriente media entregada por la fuente de salida V_o a la carga, será igual al valor medio de la corriente por L .

$$\begin{aligned} \langle i_L \rangle &= I_o \\ \langle i_L \rangle &= \theta T \cdot \frac{V_i - V_o}{L} \cdot \delta T \cdot \frac{1}{2T} \\ \theta &= \frac{2LIof}{(V_i - V_o)\delta} \end{aligned} \quad (\text{Ecu 3. Corriente media entregada por la fuente})$$

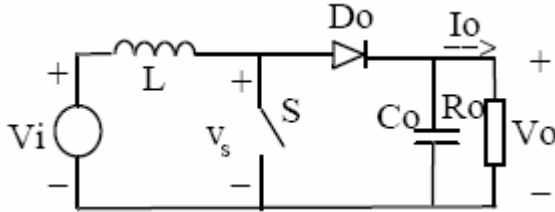
Operando con las últimas 4 ecuaciones se obtiene la tensión de salida en función de la tensión de entrada, del ciclo de trabajo, de la corriente de salida I_o , de la frecuencia f y de la inductancia L :

$$\begin{aligned} V_o &= \frac{\delta^2 V_i^2}{\delta^2 V_i + 2Lfo} \\ \frac{V_o}{V_i} &= \frac{1}{1 + \frac{2Lfo}{\delta^2 V_i}} \end{aligned} \quad (\text{Ecu. 4. Ciclo de trabajo de la fuente})$$

2.1.2. Topología Boost: El circuito del convertor Boost (o Elevador o Step-Up) está representado en la Figura 17. V_i es la tensión continua de entrada, V_o es la tensión continua de salida. Se supondrá que el capacitor de salida es suficientemente grande para considerar constante la tensión en bornes, despreciando el rizado de tensión. En este convertor siempre se cumple que V_o es mayor que V_i , razón por la que se le llama “Elevador”. El interruptor S conmuta a una frecuencia $f = 1/T$. Se distingue al igual que en el Buck, dos modos de operación, según la corriente por el inductor L se anule en el período de operación T : MCC y MCD.

Cuando el interruptor S está conduciendo la energía entregada por la fuente de entrada V_i es acumulada en el inductor L , y cuando el interruptor es bloqueado esa energía, junto con la proveniente de V_i , es transferida a la salida.

Figura 17. Conversor Boost, elevador o Step Up



✓ Modo de conducción continua – MCC. La Figura 18 presenta los circuitos equivalentes para las dos etapas de operación de este modo. La Figura 19 presenta las formas de onda principales.

Figura 18. a) circuito equivalente cuando S conduce ($0 < t < \delta T$)

b) circuito equivalente cuando S bloquea ($\delta T < t < T$)

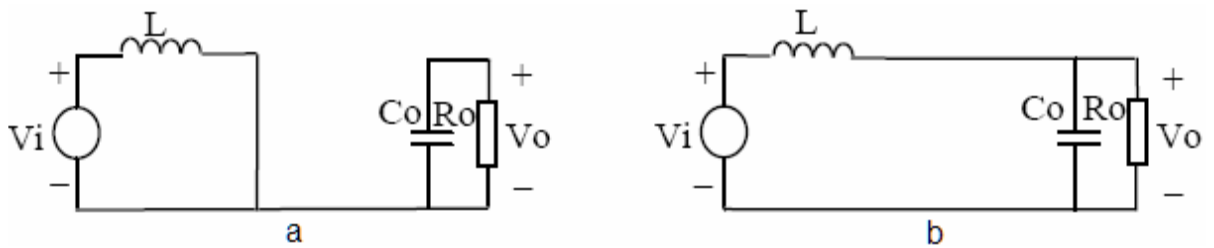
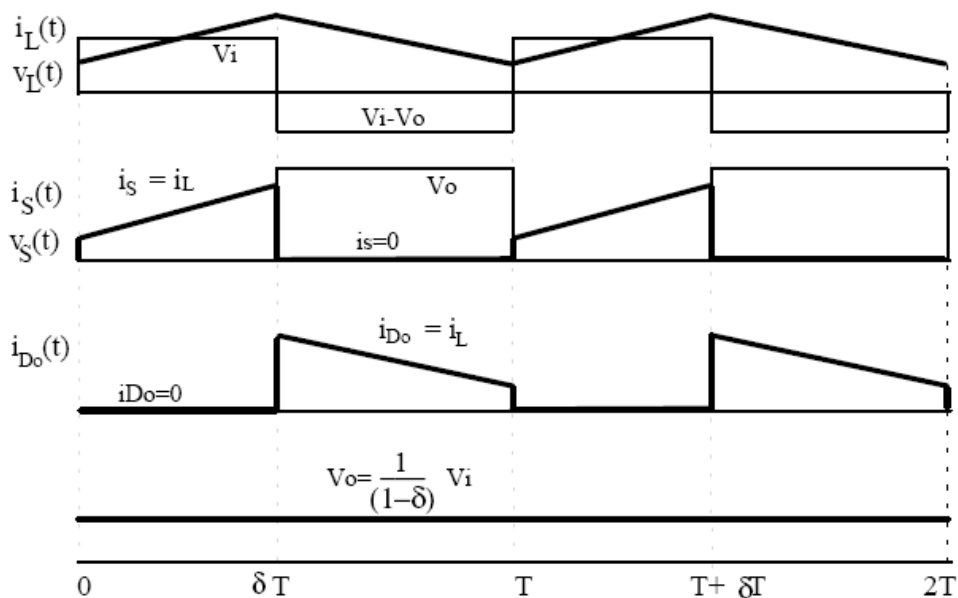


Figura 19. Principales formas de onda en MCC.



Se verifican las siguientes ecuaciones:

Intervalo $0 < t < dT$:

$$\frac{di_L}{dt} = \frac{V_i}{L}$$

$$v_s(t) = 0$$

$$i_s = i_L$$

$$i_D = 0 \quad (\text{Ecu. 5. Almacenamiento de energía Intervalo } 0 < t < dT)$$

Intervalo $dT < t < T$:

$$\frac{di_L}{dt} = \frac{V_i - V_o}{L}$$

$$v_s(t) = V_o$$

$$i_s = 0$$

$$i_D = i_L \quad (\text{Ecu. 6. Almacenamiento de energía Intervalo Intervalo } dT < t < T)$$

Para que el funcionamiento del circuito sea estable debe ser V_o mayor que V_i , en caso contrario i_L crece indefinidamente. Para encontrar la transferencia de tensión para el circuito operando en estado estacionario se considera que el valor medio de la tensión en el inductor debe ser cero, el valor medio de la tensión sobre el interruptor $u_S(t)$ es entonces igual a la tensión de entrada V_i .

$$\langle v_s(t) \rangle = V_i$$

$$(1 - \delta)V_o = V_i$$

$$\frac{V_o}{V_i} = \frac{1}{1 - \delta} \quad (\text{Ecu. 7. Ciclo de trabajo de la fuente})$$

Se observa que al igual que en el convertor Buck, en conducción continua la transferencia de tensión depende solo del ciclo de trabajo, no depende de la corriente de carga. Es posible mantener constante la tensión de salida frente a variaciones de la tensión de entrada variando el ciclo de trabajo.

✓ Modo de conducción discontinua – MCD. Si el inductor L o la corriente de carga lo son muy “chicos” el convertor opera en conducción discontinua, es decir que la corriente por L se anula durante el intervalo en que conduce D_o . La Figura 20 presenta los circuitos equivalentes para las tres etapas de este modo de operación. La Figura 21 presenta las formas de onda relevantes.

Figura 20. (a) Intervalo $0 < t < \delta T$. (b) Intervalo $\delta T < t < \theta T$. (c) Intervalo $\theta T < t < T$.

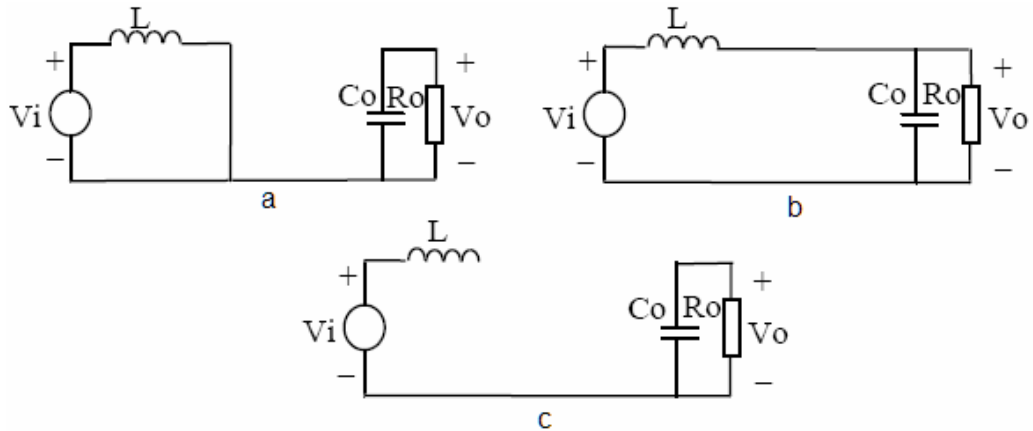
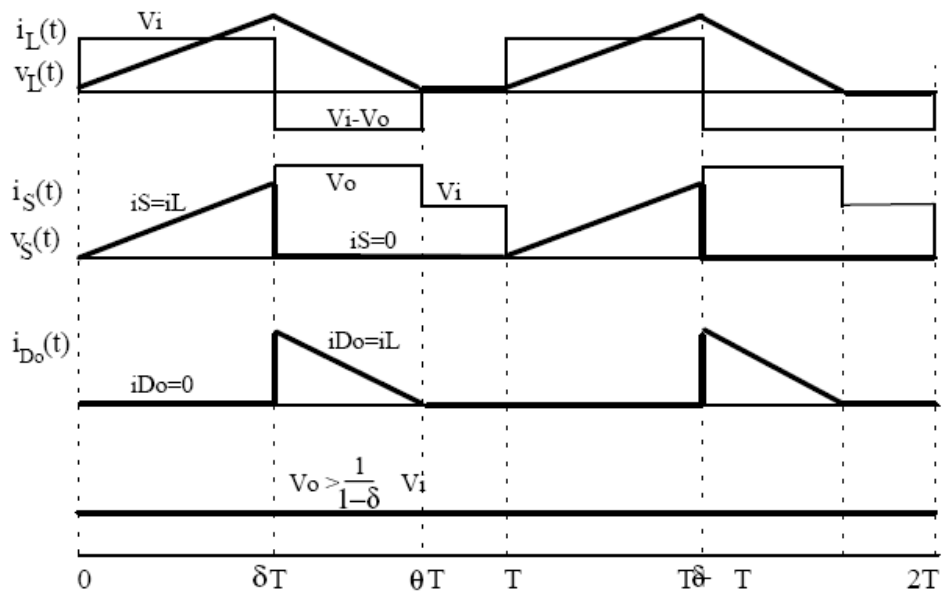


Figura 21. Formas de Onda relevantes en MDC



Se verifican las siguientes expresiones:

$$\langle v_s(t) \rangle = V_i \Rightarrow \frac{V_o}{V_i} = \frac{\theta}{\theta - \delta}$$

$$\langle i_L(t) \rangle = I_o = \frac{V_o}{R_o}$$

$$\langle i_L(t) \rangle = \frac{V_i}{2Lf} \delta(\theta - \delta)$$

$$\frac{V_o}{V_i} = 1 + \frac{V_i \delta^2}{2L f I_o}$$

(Ecu. 8. Transferencia de tensión entre la entrada y la salida)

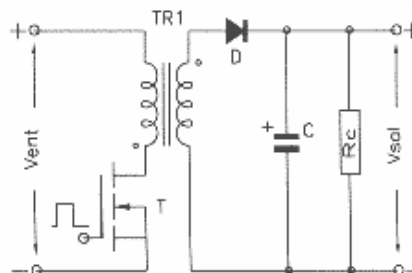
Otra vez encontramos que en MCD la transferencia de tensión entre entrada y salida depende de la corriente de carga.

2.1.3. Topología FlyBack. El convertidor Flyback es un convertidor DC a DC con aislamiento galvánico entre la entrada y la(s) salida(s). Tiene la misma estructura que un convertidor Boost con un transformador en lugar de un inductor. Gracias a ello se pueden alcanzar altos ratios de conversión. Debido a las limitaciones intrínsecas este convertidor solo se usa en aplicaciones de baja potencia.

Dada su sencillez y bajo costo, es la topología preferida en la mayoría de los convertidores de baja potencia (hasta 100 w). En la Figura 22 se muestran los principios de esta topología de fuente conmutada. Cuando «T» conduce, la corriente crece linealmente en el primario del transformador, diseñado con alta inductancia para almacenar energía a medida que el flujo magnético aumenta.

La disposición del devanado asegura que el diodo «D» está polarizado en sentido inverso durante este período, por lo que no circula corriente en el secundario. Cuando «T» se bloquea, el flujo en el transformador cesa generando una corriente inversa en el secundario que carga el condensador a través del diodo alimentando la carga. Es decir, en el campo magnético del transformador se almacena la energía durante el período «ON» del transistor y se transfiere a la carga durante el período «OFF» (FLYBACK). El condensador mantiene la tensión en la carga durante el período «ON».

Figura 22. Topología de Retroceso o FlyBack



La regulación de tensión en la salida se obtiene mediante comparación con una referencia fija, actuando sobre el tiempo «ON» del transistor, por tanto la energía transferida a la salida mantiene la tensión constante independientemente del valor de la carga o del valor de la tensión de entrada.

La variación del período «ON» se controla por modulación de ancho de pulso (PWM) a frecuencia fija, o en algunos sistemas más sencillos por autooscilación variando la frecuencia en función de la carga.

La transferencia de energía requiere un núcleo del transformador mayor que otros convertidores. Comparado con otras topologías, el interruptor debe soportar voltajes más altos.

2.2. ETAPA DE CONTROL

Para garantizar una regulación de tensión de salida adecuada se implementa una etapa de control, que modifica el ciclo de trabajo dependiendo del error en la tensión de salida.

La señal modulada por PWM (modulación por ancho de pulso) controla el encendido y el apagado del interruptor. Esta señal PWM se crea a partir de una señal rampa y el valor del ciclo de trabajo, como se aprecia en la siguiente figura.

Figura 23. Bloques componentes de un Buck.

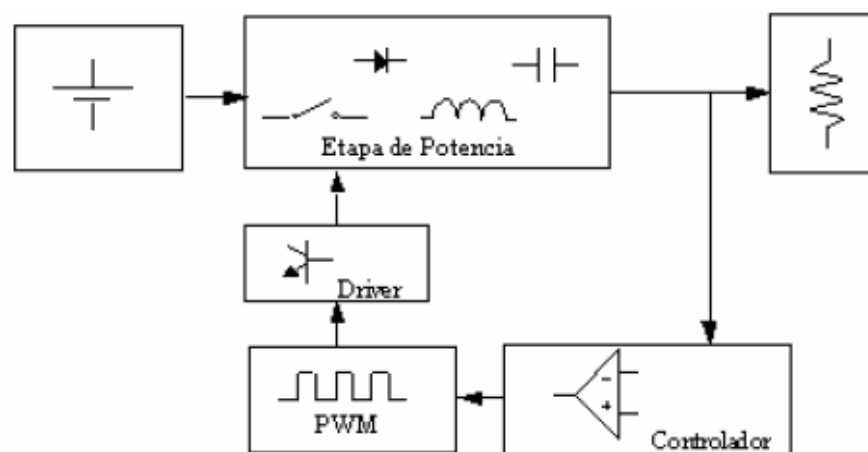
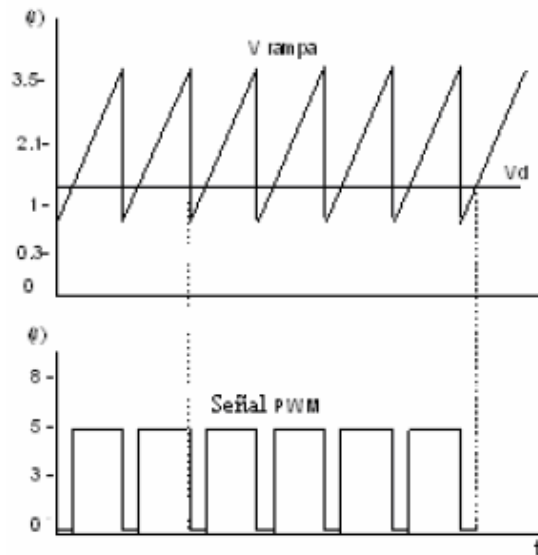


Figura 24. Generación de la señal PWM.



2.3. ESPECIFICACIONES

- Tensión de salida 12 VDC.
- Tensión de entrada 48 VDC.
- Frecuencia de conmutación 30 KHz.
- Rizado de tensión de salida 100 mV.
- Número de salidas reguladas a 5 V: 3.

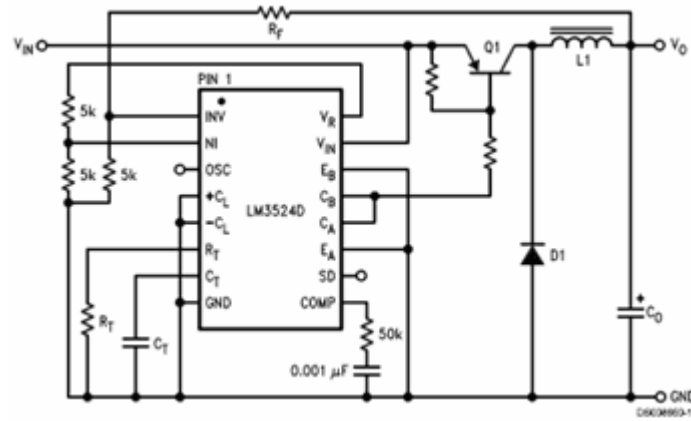
2.4. ETAPA DE DISEÑO

2.4.1. Fuente conmutada. Se requiere que tenga una salida de 12 VDC y 2 amperios para luego pasar por 3 reguladores de 5 VDC, teniendo así una fuente que parte de 48 VDC y entrega 5 VDC por tres salidas.

Para el diseño se tomó como base el circuito integrado SG3524 de Texas Instruments, en cuyo datasheet se plantea una topología buck.

Esta topología es la siguiente:

Figura 25. Diseño base de un Buck.



Para el cálculo de los componentes se usan las siguientes ecuaciones suministradas por el fabricante del circuito integrado.

$$R_F = 5K\Omega \cdot \left(\frac{V_0}{2.5} - 1 \right) \rightarrow 5K\Omega \cdot \left(\frac{12}{2.5} - 1 \right) \rightarrow R_F = 20K\Omega \text{ (Ecu. 9. Cálculo de } R_f.)$$

Donde R_f es la resistencia de retroalimentación para el circuito de control que monitorea la tensión de salida para que se mantenga en los deseados 12 VDC.

$$f_{osc} = \frac{1.30}{R_T \cdot C_T} \Rightarrow R_T = \frac{1.30}{f_{osc} \cdot C_T} \text{ (Ecu. 10. Cálculo de } R_t.)$$

Aquí se definen la frecuencia de oscilación (f_{osc}) a 30KHz y C_t a 0.1 μ F.

$$R_T = \frac{1.30}{30KHz \cdot 0.1\mu F} \rightarrow R_T \approx 433\Omega \text{ (Ecu. 11. Cálculo de } R_t.)$$

La resistencia R_t es la que nos definirá junto con C_t la señal rampa que será comprada con el error para formar la señal de control PWM.

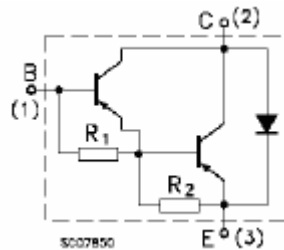
$$L = \frac{2.5 \cdot V_0 \cdot (V_{in} - V_0)}{I_0 \cdot V_{in} \cdot f_{osc}} = \frac{2.5 \cdot 12 \cdot (48 - 12)}{3 \cdot 48 \cdot 30KHz} \approx 250\mu H \text{ (Ecu. 12. Cálculo de la bobina } L)$$

Esta bobina es el elemento de almacenamiento de la energía que luego será transmitida a la carga.

$$C_0 = \frac{(V_{in} - V_0) \cdot V_0 \cdot T^2}{8 \cdot \Delta V_0 \cdot V_{in} \cdot L} = \frac{(48 - 12) \cdot 12 \cdot \left(\frac{1}{30 \text{ KHZ}}\right)^2}{8 \cdot 100 \text{ mV} \cdot 48 \cdot 250 \mu\text{H}} \rightarrow C_0 \approx 47 \mu\text{F} \text{ (Ecu. 13. Cálculo del condensador de salida)}$$

Como elemento interruptor se optó por usar un transistor BJT PNP en configuración Darlington, específicamente el TIP 147 que soporta 100V entre base - colector y colector - emisor, una corriente de colector de 10A y una disipación total de 125W. Su configuración interna se puede ver en la siguiente figura.

Figura 26. Configuración interna del TIP 147.



El TIP 147 tiene un tiempo de apagado de 4μS y un tiempo de encendido de 0.9μS, probado con una corriente de colector de 10A y una RL de 3Ω.

Para hacer la bobina se usó un toroide de ferrita de color Amarillo – Blanco, que va hasta los 100HKz y un alambre de 20 mm de diámetro.

Para calcular el número de vueltas que se le dieron al toroide con el alambre se uso la siguiente formula:

$$N = 100 \sqrt{\frac{\text{Induc.}}{A_L}} \text{ (Ecu. 14. Cálculo del número de vueltas al toroide)}$$

Donde el índice AL es la inductancia que se obtiene al enrollar cien espiras en el núcleo. Y viene dado por el material y las dimensiones del mismo, en este caso por tratarse de un núcleo Amarillo – Blanco y de 76 mm de diámetro exterior, 48 mm de diámetro interior y 13 mm de altura es de 900.

$$N = 100 \sqrt{\frac{250}{900}} \approx 60 \text{ (Ecu. 15. Cálculo del número de vueltas al núcleo)}$$

2.4.2. Etapa de Reinicialización. Consta principalmente de dos relevos que mantienen conectados los equipos de monitoreo a su alimentación, y que según el microcontrolador les mande una señal de estado alto desconectan inmediatamente los equipos de su alimentación hasta que esta señal sea puesta de nuevo en estado bajo por el microcontrolador.

Para que este procedimiento fuera óptimo la señal proveniente de microcontrolador no se podía conectar directamente a los relevos, porque no le alcanzaría la corriente para accionar la bobina y además la presencia de EMI puede afectar el funcionamiento del microcontrolador.

Para eso se usó una compuerta NOT con Smith Trigger, para aislar la señal del microcontrolador y para que sea la compuerta la que aporte la corriente necesaria, además se diseñó un circuito de tal manera que un transistor accionara las bobinas de los relevos cuando le llegara la señal de alto proveniente de la compuerta.

Figura 27. Circuito de reinicio en los -48 Vdc.

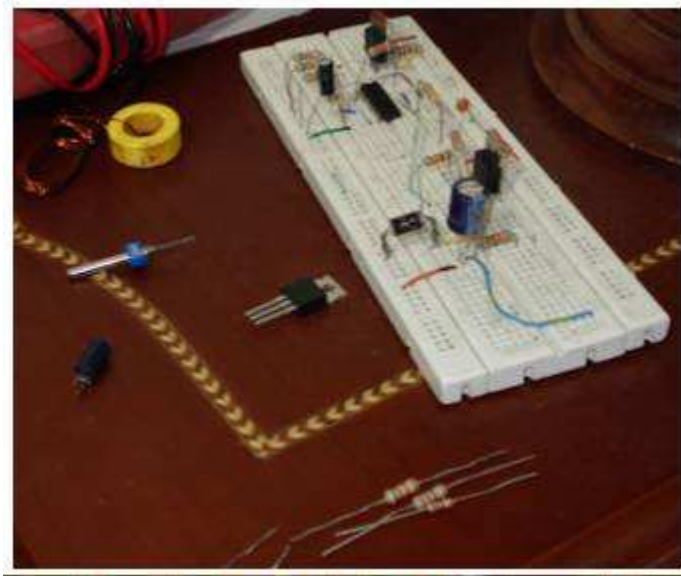


Nota: El reinicio se realiza sobre la alimentación del los -48 Vdc de los equipos de monitoreo.

2.5. RESULTADOS FINALES

Teniendo todos los cálculos hechos, se procede a hacer las pruebas pertinentes en la protoboard con todos los elementos mencionados anteriormente. En la siguiente figura se presenta cómo se montó la fuente en protoboard para pruebas.

Figura 28. Montaje de la fuente en protoboard.



En estas pruebas se presentaron algunos problemas, tales como:

- Daño de diodos zenner: fue causado por un error en el diseño que no contempló cuánta corriente soportaban los zenner que regulaban el voltaje al circuito integrado SG3524.
- Debido a esto también se dañaron 2 SG3524 ya que el voltaje que les llegaba superaba su límite de 40 VDC.
- También por esta razón al tratar de sacarle los 2 amperios a la fuente se nos dañaban los transistores que no eran los TIP 147 sino los TIP 132C que no son Darlington y que se usaron inicialmente.

Estos inconvenientes se solucionaron cambiando un poco el diseño, se quitaron los diodos zenner y se hizo un divisor de tensión, bajando el voltaje a 20 VDC y luego poniendo un regulador LM 7815 para alimentar el integrado, de esta manera se estabilizó la alimentación y no volvió a fallar.

Después de esto se seguían presentando fallas con el transistor de conmutación, y la solución que se dio fue cambiarlo por un Darlington, ya que la corriente que salía del SG3524 era muy poca para polarizar el transistor, además de recalcular las resistencias de polarización.

Con estos cambios se dieron por finalizadas las pruebas y ajustes del prototipo ya que se lograron los resultados esperados para los que fue diseñado.

Luego de terminadas las pruebas en la protoboard se procede a diseñar el circuito impreso, y para eso se usó la herramienta de Proteus que se llama Ares.

Figura 29. Circuito Impreso, vista de pistas.

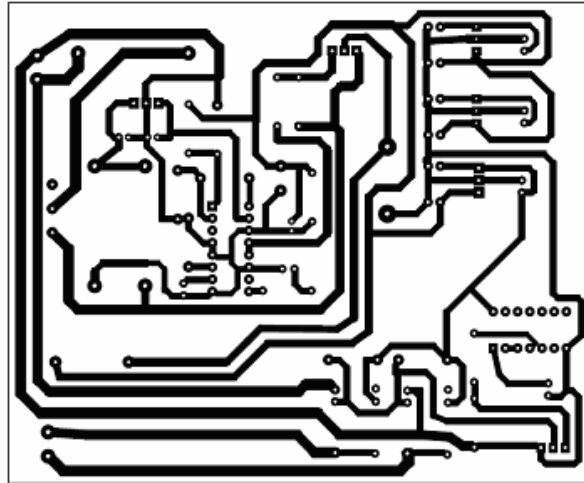
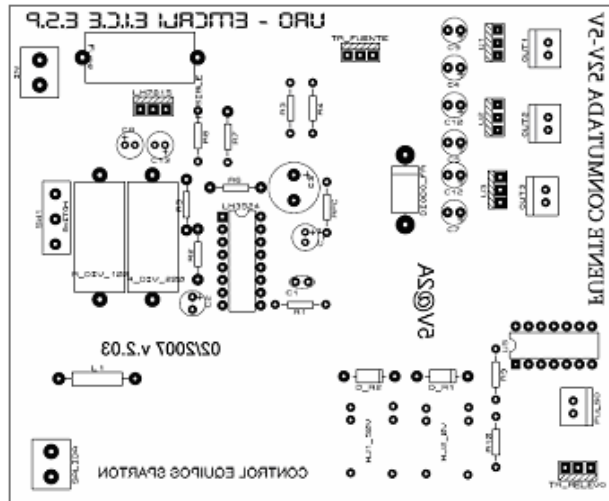


Figura 30. Circuito Impreso, vista de componentes.



El circuito impreso terminado:

Figura 31. Circuito Impreso Terminado.

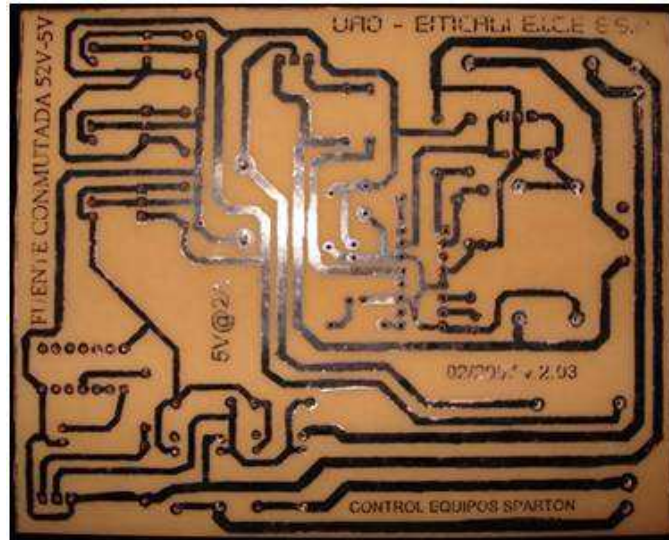
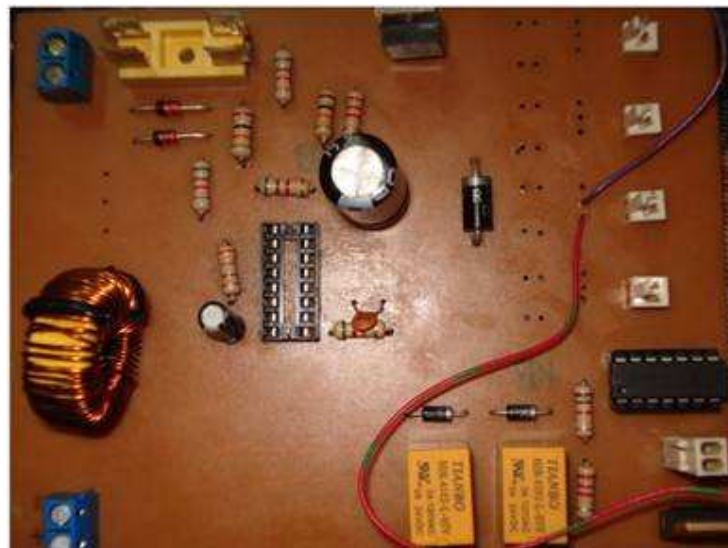


Figura 32. Fuente Terminada.



Cabe anotar que junto a la fuente conmutada, en el mismo circuito impreso se encuentra localizada la etapa del proyecto que se encarga de hacer el reset de los equipos de monitoreo.

3. IDENTIFICACIÓN DE LLAMADAS USANDO DTMF

3.4. DUAL TONE MULTI FREQUENCY (DTMF)

DTMF, o Dual Tone Multi Frequency usa dos tonos para representar cada dígito en el teclado. Cuando una tecla se presiona el tono de la columna y el tono de la fila son generados, por lo tanto es tono dual. Por ejemplo, al presionar el botón ' 5 ' se generan los tonos 770 Hz y 1336 Hz. Las frecuencias fueron elegidas para evitar armónicos (no hay frecuencia que sea múltiplo de otra, la diferencia entre dos frecuencias no es igual a otra de las frecuencias, y la suma de dos frecuencias no es igual a otra de las frecuencias). Las frecuencias generadas deben ser con +/- 1.5%, la frecuencia más alta debe tener el mismo nivel que la frecuencia más baja, dicho nivel es de 4 db. El dígito A, el B, el C y el D fueron utilizados para el sistema de teléfono de Autovon de los militares de los EE.UU.

Tabla 1. Valores de frecuencias por dígito.

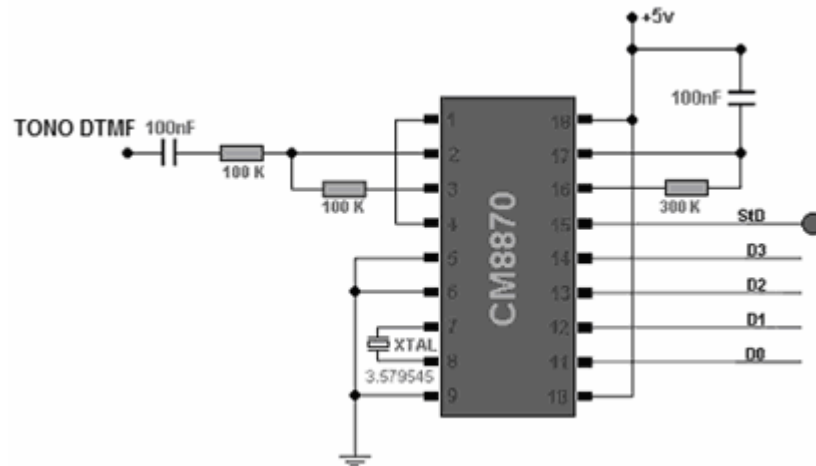
1	2	3	A	697 Hz
4	5	6	B	770 Hz
7	8	9	C	852 Hz
*	0	#	D	941 Hz
1209 Hz	1336 Hz	1477 Hz	1633 Hz	

Tabla 2. Valor decimal y binario correspondiente a la frecuencia DTMF.

Tecla	Salida Digital				Valor Decimal	Frecuencia Baja (Hz)	Frecuencia Alta (Hz)
	D3	D2	D1	D0			
1	0	0	0	1	1	697	1209
2	0	0	1	0	2	697	1336
3	0	0	1	1	3	697	1477
4	0	1	0	0	4	770	1209
5	0	1	0	1	5	770	1336
6	0	1	1	0	6	770	1477
7	0	1	1	1	7	852	1209
8	1	0	0	0	8	852	1336
9	1	0	0	1	9	852	1477
0	1	0	1	0	10	941	1336
*	1	0	1	1	11	941	1209
#	1	1	0	0	12	941	1477
A	1	1	0	1	13	697	1633
B	1	1	1	0	14	770	1633
C	1	1	1	1	15	852	1633
D	0	0	0	0	0	941	1633

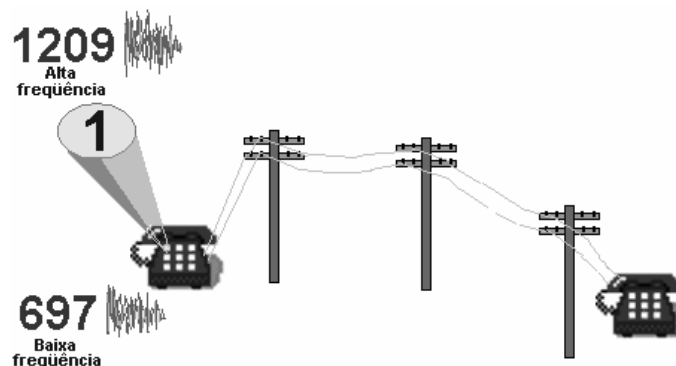
La tabla 2. De arriba muestra el valor decimal y binario de la tecla correspondiente a las frecuencias DTMF (baja y alta). El circuito integrado CM8870 reconoce una frecuencia DTMF y la convierte en un número binario de 4 bits (1 nibble). El software en un microcontrolador es el responsable en convertir ese número binario en una tecla equivalente al sistema DTMF. Observe que el valor decimal 10 representa la tecla "0".

Figura 33. Identificación del tono DTMF con CM8870.



La configuración sugerida por el fabricante del CM8870 para detectar frecuencias DTMF, es la esquematizada en la figura 22. donde la frecuencia entra a través del condensador cerámico de 100 nF y prosigue por las resistencias de 100 k hasta los pines 1 y 3 del CM8870. Cuando una frecuencia es reconocida, el CM8870, convierte este dato en un valor binario y lo representa en lo pines 11, 12,13 y 14. Después de eso, el CM8870 activa el pin 15, manteniéndolo en nivel alto por algunos milisegundos. Después esto el pin retorna al nivel bajo.

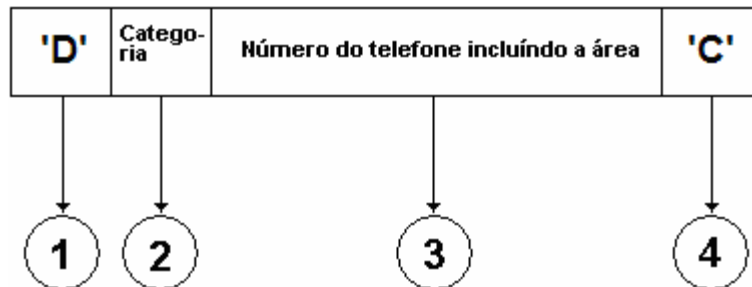
Figura 34. Transmisión del tono DTMF.



La figura 34 muestra la transmisión la transmisión de la tecla "1", tecleada en un aparato telefónico, y detectada en otro aparato telefónico distante a través de la Red telefónica. Ambos aparatos tienen circuitos integrados codificadores y decodificadores de DTMF. Vea que las dos frecuencias son sobrepuestas y sólo después transmitidas.

3.5. PROTOCOLO DTMF

Figura 35. Protocolo DTMF



La figura 35 muestra un gráfico del protocolo DTMF para reconocimiento del número del teléfono del llamador. Esas informaciones son enviadas antes del teléfono tocar, o sea, la central telefónica envía primero el paquete de identificación del teléfono llamador, y sólo después de algunos segundos es enviada la señal para tocar el teléfono.

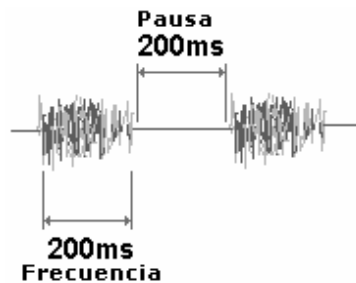
Importante: Para que ese recurso de identificación de llamadas esté disponible en la línea telefónica, es necesario estar habilitado el servicio por la empresa de telefonía.

Tabla 3. Características del protocolo DTMF.

Campo	Descripción
1	Ese es un valor constante. El carácter "D" o 0(decimal), identifica el inicio del protocolo.
2	Categoría puede ser: 1 – teléfono residencial o comercial; 4 – teléfono público; 7 – teléfono público a tarjeta; Especiales: 2,3,5,6,8,9,10,11,12,13,14 y 15.

3	Ese campo almacena el número del teléfono propiamente dicho, inclusive el código de área Por ejemplo: 11 9999-9999.
4	Ese es un valor constante. El carácter "C" o 15(decimal), identifica el fin del protocolo.

Figura 36. Intervalo de tiempo de los tonos DTMF.



En la figura 36 tenemos una representación de los intervalos de tiempo del paquete de identificación enviado por la central telefónica, tanto de la frecuencia como de la pausa. El tiempo entre ellos es de 200 ms. Si la distancia entre el suscriptor y la Central telefónica, es importante destacar que el intervalo de tiempo podrá sufrir alteraciones debido a una distancia muy larga.

3.6. CIRCUITO DE IDENTIFICACIÓN

Para facilitar la comprensión del circuito de la figura 37. Dividimos el mismo en 2 bloques. El bloque fundamental del circuito es el numero 2, el cual es el que valida la frecuencia recibida y decodifica en binario, El bloque restante funcionan como un recurso auxiliar, pero que es de bastante importancia para el buen funcionamiento del proyecto. El varistor en paralelo con la línea telefónica protege el circuito de una ocasional sobrecarga de tensión. La función del bloque 1 es de amplificar la señal DTMF y llevarlo al CM8870.

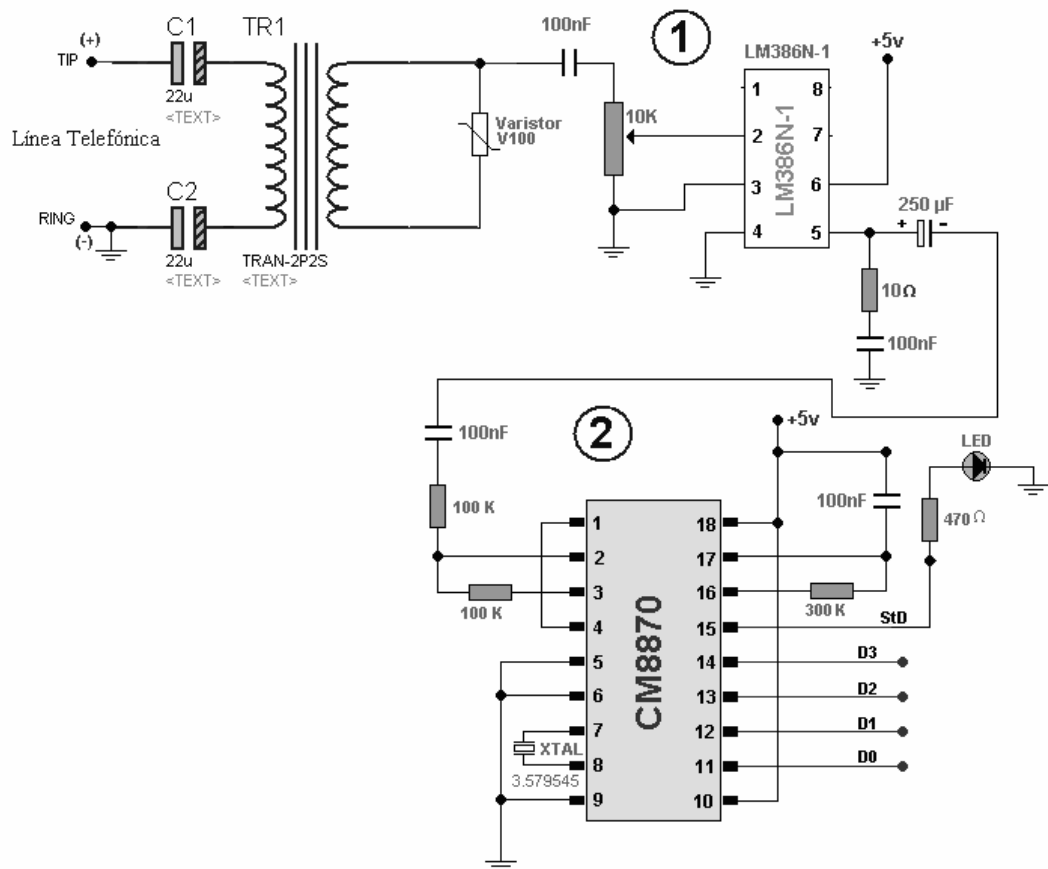
En la red de telefónica:

RING es polo negativo.

TIP es el polo positivo.

La tensión en la Red telefónica puede llegar a 90 Vac.

Figura 37. Circuito de identificación de tonos DTMF.



4. IDENTIFICADOR DE LLAMADAS BINA



La sigla BINA, que viene de: "B" Identifica el Numero de "A", donde "B" es el teléfono que esta siendo llamado y "A" quien está llamando, empleado en equipamientos mundialmente conocidos como CALLER-Id.

El BINA es un servicio especial que ofrecen las centrales telefónicas de tecnología digital (AXE y EWSD). Permiten identificar el número telefónico de la persona que llama, antes de contestar.

4.1. FUNCIONES DEL SERVICIO

Para disfrutar de este servicio se requiere de un dispositivo especial (equipo) llamada identificador de llamadas, que lo provee EMCALI y que para lo cual a realizado alianzas con proveedores quienes garantizan la disponibilidad y el funcionamiento de los dispositivos.

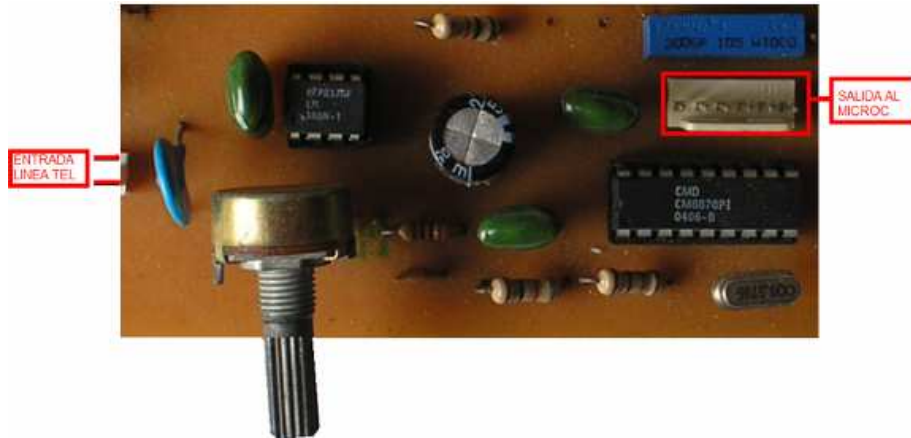
Entre los beneficios que presenta este servicio podemos destacar los siguientes:

- Brinda elementos para la toma de decisión de atender o no una llamada telefónica.
- Permite disponer de elementos de seguridad.
- Establece base de datos, conociendo quien llama.
- El equipo puede conectarse a cualquier extensión de una línea telefónica.

4.2. RESULTADOS FINALES

Una vez considerado todos los datos anteriores se realizaron las respectivas pruebas ensamblando el circuito en protoboard para corregir las dificultades que se podían presentar, posteriormente se realizo el monte ensamblado en un circuito impreso.

Figura 38. Circuito identificador de llamadas.



En estas pruebas se presentaron algunos problemas, tales como:

❖ La ganancia en el amplificador operacional LM386 era modificada por el potenciómetro de 10 K Ω lo que ocasionaba que la señal alterna de la llamada tuviera mas amplitud hasta el punto que el amplificador operacional se saturara cortando los picos del la señal y de esta forma el tono no era identificado por el CM8870, o de forma contraria cuando se reducía la ganancia la señal se atenuaba haciendo imposible el proceso de identificación de los tonos.

❖ Después de ser montado en la plaqueta, el circuito era probado con un transformador conectado a la tensión de 110 Vac para obtener el voltaje de alimentación de 5 V (adaptador comercial de voltaje), funcionando sin ningún inconveniente, cuando se probó esta etapa en conjunto con la etapa del conversor DC/DC, se presento un problema con las tierras del circuito, debido a que esta etapa de identificación aterriza una de las líneas telefónicas a la tierra común del circuito, para tener una referencia cuando entraban los dígitos DTMF, esta tierra interfería con la tierra del convertidor dc-dc que tomaba la referencia de la alimentación de los -48 Vdc provenientes de la central telefónica. Es decir que la tierra de la línea telefónica era diferente a la tierra de los -48 Vdc de la central ocasionando una diferencia de potencial entre las dos tierras, lo que ocasionaba que el circuito no funcionara y que la línea telefónica perdiera la señal al presentarse un voltaje en la línea.

Este inconveniente fue solucionado cambiando el diseño del circuito, para lo cual tuvimos que agregar un transformador a la entrada para generar un aislamiento galvanico y así separa la tierra de la línea telefónica de la del circuito.

Figura 39. Aislamiento Galvanico en entrada del identificador.

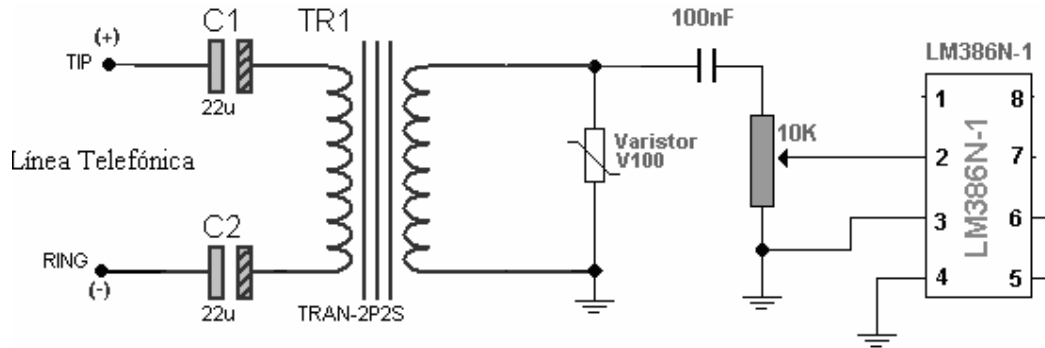


Figura 40. Montaje de la etapa de aislamiento del identificador.



Los condensadores a la entrada del transformador sirven para desacoplar la parte de DC de la línea telefónica dejando pasar solo la componente AC. El transformador puede tener una relación de transformación 1:1 funcionando solo como aislamiento, pero debido que al ser un transformador poco común no es muy comercial, por lo tanto decidimos utilizar un transformador reductor con una relación de transformación de 1:6 por ser mas comercial, la reducción de la señal es compensada por el amplificador operacional dándole mas ganancia a la señal reestableciéndola a sus condiciones normales.

5. EL MODEM

El módem es un dispositivo que permite conectar dos ordenadores remotos utilizando la línea telefónica de forma que puedan intercambiar información entre sí. El módem es uno de los métodos más extendidos para la interconexión de ordenadores por su sencillez y bajo costo.

La gran cobertura de la red telefónica convencional posibilita la casi inmediata conexión de dos ordenadores si se utiliza módems. El módem es por todas estas razones el método más popular de acceso a la Internet por parte de los usuarios privados y también de muchas empresas.

5.1. CÓMO FUNCIONA EL MODEM

Debe enviar los datos digitales de la computadora a través de líneas telefónicas análogas. Logra esto modulando los datos digitales para convertirlos en una señal análoga; es decir, el módem varía la frecuencia de la señal digital para formar una señal análoga continua. Y cuando el módem recibe señales análogas a través de la línea telefónica, hace el opuesto: demodula, o quita las frecuencias variadas de la onda análoga para convertirlas en impulsos digitales. De estas dos funciones, Modulación y Demodulación, surgió el nombre del módem.

5.2. VELOCIDADES DE TRANSMISIÓN DEL MODEM

Las velocidades de módem se miden por el número de bits que el módem puede enviar o recibir por segundo. Por ejemplo un módem de 33,6 Kbps puede enviar y recibir 33,6 kilobits (un kilobit consiste en 1.000 bits) de datos por segundo.

A veces, la velocidad del módem es expresada en baudio.

BAUDIOS. Numero de veces de cambio en el voltaje de la señal por segundo en la línea de transmisión. Los módem envían datos como una serie de tonos a través de la línea telefónica. Los tonos se "encienden"(ON) o "apagan"(OFF) para indicar un 1 o un 0 digital. El baudio es el número de veces que esos tonos se ponen a ON o a OFF. Los módem modernos pueden enviar 4 o más bits por baudio.

BITS POR SEGUNDO (BPS). Es el número efectivo de bits/seg que se transmiten en una línea por segundo. Como hemos visto un módem de 600 baudios puede transmitir a 1200, 2400 o, incluso a 9600 BPS.

La señal está formada por diferentes tonos que viajan hasta el otro extremo de la línea telefónica, donde se vuelven a convertir a datos digitales.

5.3. MÓDEM HANDSHAKING

Un proceso de intercambio de señales es necesario para establecer la conexión entre dos módem. Estas señales son requeridas para iniciar y terminar una llamada, y un tipo de señalización usada es determinante de acuerdo con uno de los tres mayores patrones: EIA (Electronic Industry Association), RS-232C o RS-449 y el CCITT v.24. Estos dos últimos son idénticos y usados por el 98% de los actuales fabricantes de módem.

Tabla 4. Estándares de modulación.

TIPO	CARACTERÍSTICAS
Bell 103	Especificación del sistema Bell para un módem de 300 baudios, asíncrono y full-duplex
Bell 201	Especificación del sistema Bell para un módem de 2400 BPS, síncrono, y Full- duplex.
Bell 212	Especificación del sistema Bell para un módem de 2400 BPS, asíncrono, y Full-duplex.
V.22 bis	Módem de 2400 BPS, síncrono/asíncrono y full-duplex
V.29	Módem de 4800/7200/9600 BPS, síncrono y full-duplex
V.32	Módem de 4800/9600 BPS, síncrono/asíncrono y full-duplex
V.32 bis	Módem de 4800/7200/9600/7200/12000/14400 BPS, síncrono/asíncrono y full-duplex
Hayes Express	Módem de 4800/9600 BPS, síncrono/asíncrono y half-duplex. Sólo compatibles consigo mismo aunque los mas modernos soportan
V.32	
USR-HST	Módem de USRobotics de 9600/14400 BPS. Sólo compatibles consigo mismo aunque los mas modernos soportan V.32 y
V.32bis	
Vfast	Vfast es una recomendación de la industria de fabricantes de módem. La norma Vfast permite velocidades de transferencia de hasta 28.800 bps
V34	Estándar del CCITT para comunicaciones de módem en velocidades de hasta 28.800 bps

5.4. LIMITACIÓN FÍSICA DE LA VELOCIDAD DE TRANSMISIÓN EN LA LÍNEA TELEFÓNICA

Las leyes físicas establecen un límite para la velocidad de transmisión en un canal ruidoso, con un ancho de banda determinado. Por ejemplo, un canal de banda 3000Hz, y una señal de ruido 30dB (que son parámetros típicos del sistema telefónico), nunca podrán transmitir a más de 30.000 bps.

Throughput: Define la cantidad de datos que pueden enviarse a través de un módem en un cierto período de tiempo. Un módem de 9600 baudios puede tener un throughput distinto de 9600 BPS debido al ruido de la línea (que puede ralentizar) o a la compresión de datos (que puede incrementar la velocidad hasta 4 veces el valor de los baudios).

Para mejorar la tasa efectiva de transmisión o throughput se utilizan técnicas de compresión de datos y corrección de errores.

Compresión de datos: Describe el proceso de tomar un bloque de datos y reducir su tamaño. Se emplea para eliminar información redundante y para empaquetar caracteres empleados frecuentemente y representarlos con sólo uno o dos bits.

Control de errores: La ineludible presencia de ruido en las líneas de transmisión provoca errores en el intercambio de información que se debe detectar introduciendo información de control. Así mismo puede incluirse información redundante que permita además corregir los errores cuando se presenten.

5.5. ESTÁNDARES DE CONTROL DE ERRORES

El problema de ruido puede causar pérdidas importantes de información en módems a velocidades altas, existen para ello diversas técnicas para el control de errores. Cuando se detecta un ruido en un módem con control de errores, todo lo que se aprecia es una breve inactividad o pausa en el enlace de la comunicación, mientras que si el módem no tiene control de errores lo que ocurre ante un ruido es la posible aparición en la pantalla de caracteres "basura" o, si se está transfiriendo un fichero en ese momento, esa parte del fichero tendría que retransmitirse otra vez.

5.6. CONTROL DE FLUJO

El control de flujo es un mecanismo por el cual módem y ordenador gestionan los intercambios de información. Estos mecanismos permiten detener el flujo cuando uno de los elementos no puede procesar más información y reanudar el proceso. Los métodos más comunes de control de flujo son:

- Control de flujo hardware: RTS y CTS permiten al PC y al módem parar el flujo de datos que se establece entre ellos de forma temporal. Este sistema es el más seguro y el que soporta una operación adecuada a altas velocidades.
- Control de flujo software: Xon/Xoff: Aquí se utilizan para el control dos caracteres especiales XON y XOFF (en vez de las líneas hardware RTS y CTS) que controlan el flujo. Cuando el PC quiere que el módem pare su envío de datos, envía XOFF. Cuando el PC quiere que el módem le envíe más datos, envía XON. Los mismos caracteres utilizan el módem para controlar los envíos del PC. Este sistema no es adecuado para altas velocidades.

5.7. COMANDOS DE CONTROL DEL MODEM.

La mayoría de los módem se controlan y responden a caracteres enviados a través del puerto serie. El lenguaje de comandos para módem más extendido es de los comandos Hayes que fue inicialmente incorporado a los módem de este fabricante. Existen dos tipos principales de comandos.

- Comandos que ejecutan acciones inmediatas (ATD marcación, ATA contestación o ATH desconexión).
- Comandos que cambian algún parámetro del módem (por ejemplo AT\$7=90).

5.8. MODOS DE OPERACIÓN DEL MODEM

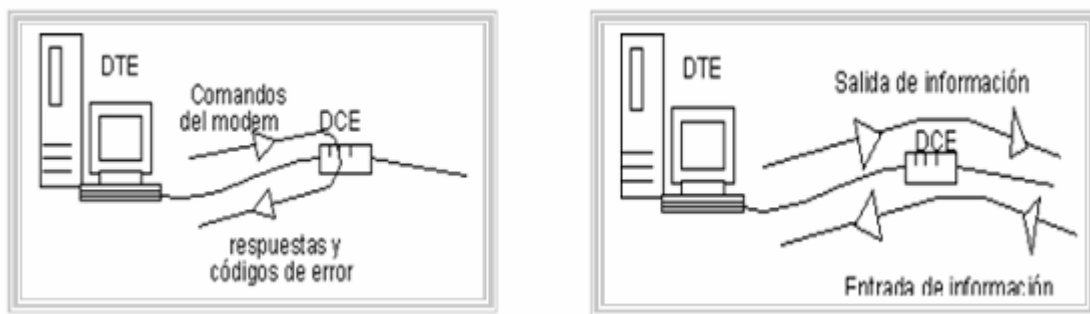
El módem tiene dos modos de funcionamiento:

El módem está en "estado de comandos" el módem responde a los comandos que envía el computador. En este modo es posible configurar el módem o realizar las operaciones de marcado y conexión. Antes de que se puedan enviar un comando al módem este debe estar en el "estado de comandos".

Cuando el módem se conecta con otro módem pasa al “modo en línea”. En este modo cualquier información que reciba del ordenador será enviada al módem distante. En este modo el módem no procesa la información y simplemente la trasmite a través de la línea de comunicación.

Para salir del modo en línea y pasar de nuevo al modo comandos se envía al módem +++ (petición de atención) precedidos por un segundo de inactividad.

Figura 41. Modo de funcionamiento del modem.



Modo comando

Modo línea

5.8.1. Formato de comandos HAYES. Un comando Hayes es una cadena ASCII que se envía al módem para que este realice alguna acción o configure alguno de sus parámetros. Esta comienza con los caracteres 'AT', seguida de nombres de comando y sus respectivos parámetros, y que concluye con un carácter de retorno de carro. No debe sobrepasar 256 caracteres, y puede incluir espacios para separar comandos y parámetros.

A continuación se describen algunos comandos típicos que suelen estar incluidos en la mayoría de módems.

♣ **A (Answer)**; responder. Se utiliza en conexiones manuales para indicar al módem que descuelgue y responda a un posible módem remoto, pasando de modo voz a modo datos. Se sigue este proceso:

- Paso 1. El módem que contesta envía portadora como respuesta al que llama.
- Paso 2. Cuando el módem que llama recibe portadora pasa a modo de intercambio de datos.
- Paso 3. Si el proceso se completa satisfactoriamente ambos módem envían un mensaje "CONNECT..." a los DTE's.

Paso 4. Si vence el tiempo de espera de portadora (especificado en el registro S7) sin recibirse portadora el módem envía el mensaje "NO CARRIER" al DTE.

♣ **D (Dial)**; marcar. Permite establecer conexión con un destino. Existen dos opciones de uso:

Comando 'D' seguido del número de teléfono destino. Sin especificar número del destino; el módem descuelga e intenta conectar directamente. Como parámetros más importantes se pueden utilizar los siguientes dígitos y modificadores tras el comando:

'0' - '9': Dígitos del número a marcar.

'A', 'B', 'C', 'D', '#', '*': Tonos específicos para marcación por tonos.

'T': Utiliza marcación por tonos.

'P': Utiliza marcación por pulsos.

',': Genera un tiempo muerto antes de marcar el siguiente símbolo, según el valor especificado en el registro S8.

':': Empleado como último dígito de una cadena antes del CR, sirve para pasar a modo comando sin cortar a comunicación tras el marcado, dejando el módem en espera de respuesta.

Todos los comandos Hayes empiezan con la secuencia AT. La excepción es el comando A/. Tecleando A/ se repite el último comando introducido. El código AT consigue la atención del módem y determina la velocidad y formato de datos.

Los comandos más simples:

- **ATH** dice al módem que cuelgue el teléfono.
- **ATDT** dice al módem que marque un número de teléfono determinado empleando la marcación por tonos.
- **ATDP** lo mismo que ATDT pero la marcación es por pulsos.
- **ATE** Eco.
- **ATI** Revisa la ROM del módem (checksum).
- **ATL** Programa el volumen del altavoz.
- **ATM** Programa conexión/desconexión del altavoz.
- **ATO** Vuelve a estado on-line desde el estado de comandos.
- **ATQ** Programa los códigos de resultado a ON/OFF.
- **ATS** Visualiza/cambia contenidos de los registros S.
- **ATV** Envía códigos de resultado en palabras o números.
- **ATW** Envía "códigos del progreso de la negociación".
- **ATX** Programa códigos de resultado.
- **ATZ** Reset.

- **AT&C** Programa detección de portadora.
- **AT&D** Programa control de DTR.
- **AT&K** Programa control de flujo.
- **AT&W** Almacena perfil configuración del usuario.
- **AT&Y** Especifica que perfil de configuración usuario de los almacenados se va a utilizar.

Los módem actuales incluyen muchas funciones adicionales como marcado automático, métodos de compresión y corrección de los datos, distintos protocolos de intercambio de datos con el DTE, etc. Por ello, los módem son configurables por software a través de comandos especiales aunque salen de fábrica con una configuración por defecto.

5.9. MODEM: XECOM XE2496

Características:

- Velocidad de datos: 2400, 1200 y 300 bps.
- Envía fax a 9600 bps.
- Recibe fax a 4800 bps.
- Acepta comandos "AT" para el control y Configuración de modem.
- Suministro de alimentación: +5 V.
- Potencia CMOS: En operación: 200 mW (Typ.)
 En modo sleep: 50 mW (Typ.).
- Dimensiones: XE9624F: 2.28" por 1.08" por 0.42"
 XE9624FS: 1.53" por 0.46" por 0.92"

Figura 42. Configuración de pines en el módem.

XE2496F Pin Configuration		
RESET	1	40 +5V
AR	2	39 /DSR
RXD	3	38 /DCD
	4	37
	5	36
	6	35
	7	34
	8	33
/DTR	9	32
/AA	10	31
/CTS	11	30
OH	12	29
TXD	13	28
/RTS	14	27
/HS	15	26
/RI	16	25
	17	24
TIP	18	23
	19	22 AMP
RING	20	21 GND

5.10. CONEXIÓN RS232 ENTRE EL MICROCONTROLADOR Y MÓDEM

Los módem se conectan con el microcontrolador a través de un puerto de comunicaciones serie. Estos puertos siguen comúnmente la norma RS232.

A través del protocolo RS232 utilizado entre el microcontrolador y módem estos se comunican. Hay varios pines independientes en el RS232. Dos de estos pines, el de transmitir datos (TD), y el de recibir datos (RD) forman la conexión de datos entre PIC y Módem. Hay otros pines en la interfaz que permiten leer y controlar la comunicación.

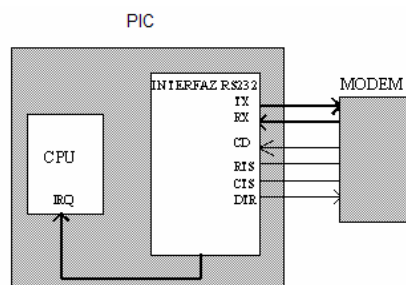
Data Terminal Ready (DTR). Esta señal indica al módem que el PIC está conectado y listo para comunicar. Si la señal se pone a OFF mientras el módem esta en on-line, el módem termina la sesión y cuelga el teléfono.

Carrier Detect (CD). El módem indica al PIC que esta on-line, es decir conectado con otro módem.

Request to send (RTS). Normalmente en ON Se pone OFF si el módem no puede aceptar más datos del PIC, por estar en esos momentos realizando otra operación.

Clear to send (CTS). Normalmente en ON Se pone OFF cuando el PIC no puede aceptar datos del módem.

Figura 43. Diagrama de conexión entre el PIC y el módem.



5.10.1. Desarrollo de una Conexión a través de Módem. La secuencia de comunicación entre dos módems, empieza con el paso 1 y termina con el paso 12.

Tabla 5. Secuencia de conexión entre módems

Paso	Usuario	Software	Módem llama	Módem responde
1	Selecciona "dial" en el menú del programa o teclea en la línea de comandos.	Pone a ON la señal DTR y envía al módem el comando de marcación ATDT 055	El módem conecta el altavoz, descuelga la línea, espera el tono de llamada y marca el número de teléfono.	
2		Comienza observando los códigos de resultados del módem.	Espera una respuesta durante tiempo según configuración del registro S7.	
3				La línea de teléfono suena.
4				El módem detecta la llamada, y contesta situando el tono de respuesta en línea.
5			El módem detecta el modo de respuesta y sitúa la portadora de comienzo en línea.	
6			Los módems se ponen de acuerdo en la modulación y velocidad a utilizar.	Los módems se ponen de acuerdo en la modulación y velocidad a utilizar.
7			Los módems determinan la técnica de compresión y control de errores a utilizar	Los módems determinan la técnica de compresión y control de errores a utilizar
8			Envía el código de rtdo. "Connect" al PC, apaga el altavoz, y pone a ON la señal CD.	
9		Detecta el código de rtdo. y/o la señal CD; Informa al usuario que la conexión está establecida.		
10	Comienza la comunicación con el host.	Gestiona la sesión de comunicaciones; vigila la pérdida de portadora	Envía y recibe datos.	Envía y recibe datos.

		monitorizando la señal CD.		
11	Completa la sesión de comunicaciones y selecciona el comando "disconnect".	Pone a OFF la señal DTR, o envía +++ seguidos por ATH.		
12			Cuelga el teléfono.	Detecta la pérdida de portadora y cuelga.

Para comunicarse con el modem se debe hacer uso de un protocolo basado en los comandos Hayes.

5.11. CÓDIGOS DE RESULTADOS

Cuando se envía un comando al módem, este responde con un código de resultado: "CONNECT", "OK" o "ERROR".

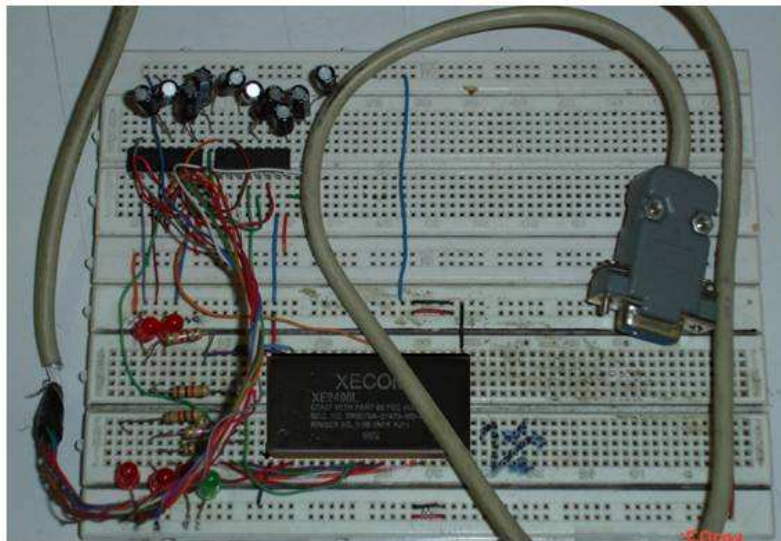
Tabla 6. Respuestas del módem

Digito	Código	Significado del código
0	OK	Comando de línea correctamente ejecutado
1	CONNECT	Conexión de 300 bps establecida
2	RING	Señal de ring detectada
3	NO CARRIER	Portadora no detectada
4	ERROR	Error en la línea de comandos
5	CONNECT 1200	Conexión de 1200 bps establecida
6	NO DIAL TONE	No se detecto tono de marcado
7	BUSY	Señal de ocupado detectada
8	NO ANSWER	No se detectaron 5 segundos de silencio
10	CONNECT 2400	Conexión de 2400 bps establecida
13	DATA	Conectado en modo DATOS después de la respuesta automática
15	FAX	Conectado en modo FAX después de respuesta automática
+F4	+FCERROR	Error de la portadora de FAX detectado

5.12. RESULTADOS FINALES.

La configuración y el manejo del modem se realizo mediante un montaje en protoboard, para poder determinar de esta forma los códigos de programación del modem y la respuesta que generaba a dichos códigos.

Figura 44. Circuito para el control y configuración del modem.

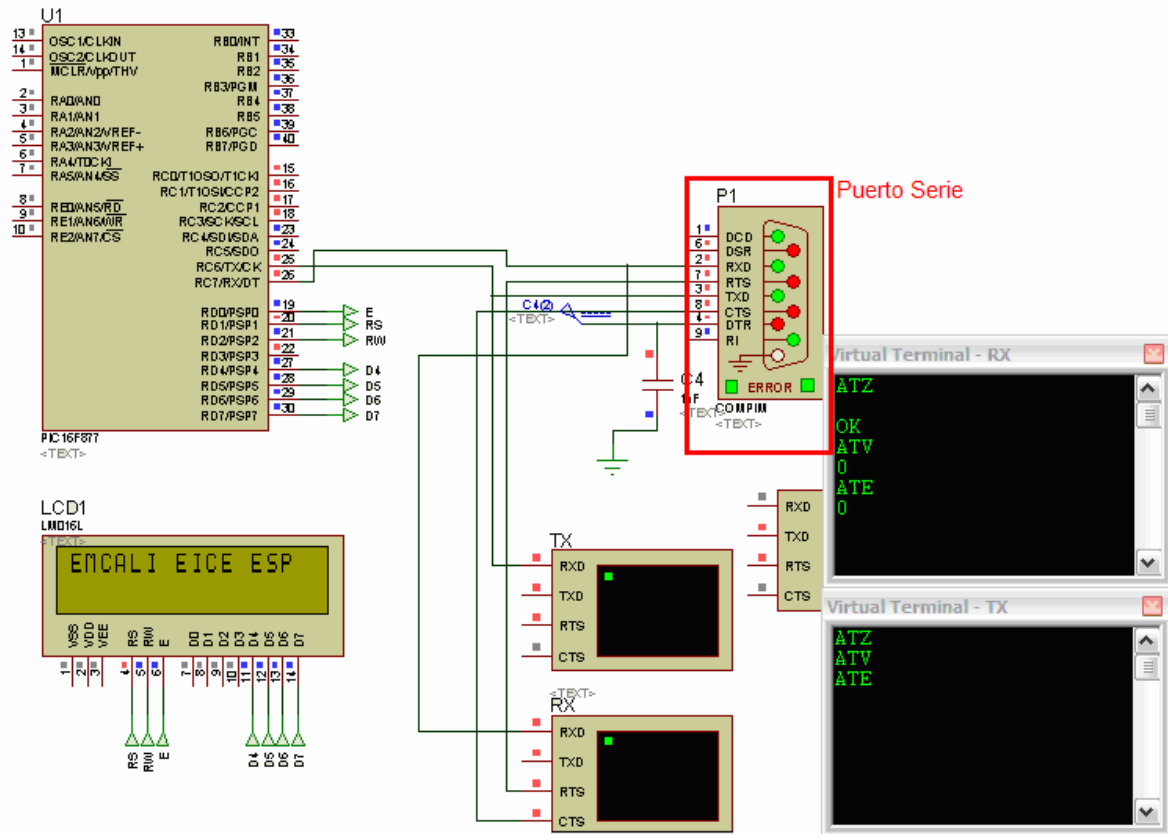


En la figura 44 se observa el montaje en protoboard, utilizado para conocer el funcionamiento del modem, el modem era conectado a un PC a través de un puerto serie, pero para lograr esto era necesario emplear el MAX232 que se encargaba de generara los niveles de voltajes (12Vdc, -12Vdc) necesarios para realizar la comunicación RS232 entre el PC y el modem.

Una vez se lograba la comunicación entre el modem y el PC, se realizaba la configuración del modem desde el HYPER TERMINAL mediante los comandos HAYES, de esta forma podíamos determinar el estado del modem.

Después el modem fue configurado desde el simulador ISIS PROTEUS, ya que este simulador nos brinda la posibilidad del manejo de los puertos serie que tiene el PC.

Figura 45. Simulación del microcontrolador con el modem físico a través del puerto serie.



Con la ayuda de esta herramienta podíamos interactuar de forma virtual con el microcontrolador que se encontraba en el simulador y el modem físico a través del puerto serie, logrando de esta forma examinar lo siguiente:

- ◆ Simular el programa cargado en el microcontrolador.
- ◆ Evaluar el funcionamiento del modem al interactuar con el microcontrolador.
- ◆ Examinar los errores en la transmisión y recepción de la comunicación RS232.
- ◆ Realizar pruebas de comunicación por la línea telefónica, transmitiendo información desde el microcontrolador simulado en ISIS PROTEUS por el puerto serie hasta el modem, y este estableciendo la comunicación con un equipo remoto (otro PC).

Los problemas que se presentaron en esta etapa estaban relacionados con la comunicación del microcontrolador y el modem, la transferencia de información, estos problemas formaban parte del software desarrollado en el PIC.

- ♣ Las velocidades de transmisión no eran las mismas por lo tanto la información no era identificada.
- ♣ La comunicación RS232 no estaba debidamente configurada ocasionando fallos en el funcionamiento del microcontrolador.
- ♣ La velocidad con la que el PIC recibía la información del modem no era lo suficiente lo que ocasionaba pérdida de la información recibida.
- ♣ El búfer de recepción del PIC se saturaba por información que provenía del modem, interrumpiendo el funcionamiento del PIC.

Todos los problemas presentados pertenecían a la programación del microcontrolador, pero fue de esta forma que logramos examinar de forma minuciosa que estaba ocurriendo en la comunicación, ya que gracias al simulador lográbamos depurar el programa paso a paso para evaluar el contenido de los registros internos del microcontrolador y así corregir las dificultades encontradas en el software diseñado.

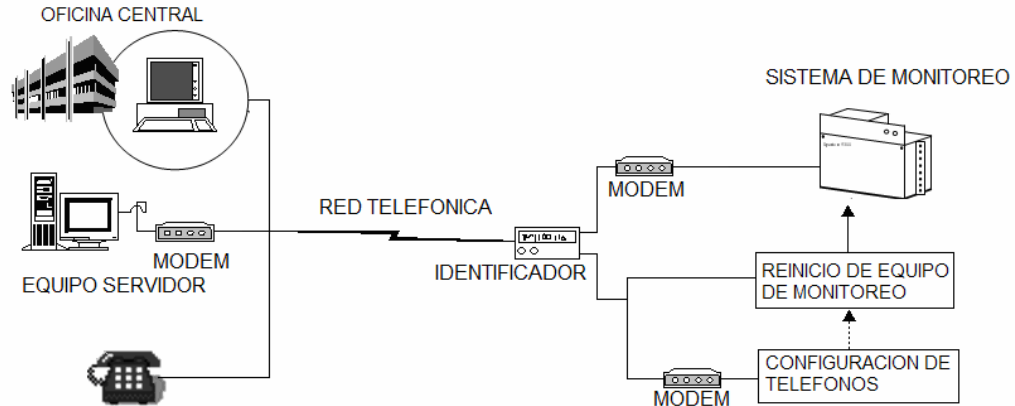
6. SISTEMA DE COMUNICACIÓN

El sistema a desarrollar consta de un “*identificador*” que se encarga de reconocer el número de la llamada entrante para conmutarla entre el sistema de monitoreo y el modulo de reinicio dependiendo del número que este llamando, para evitar que los dos módems entren en interferencia al tratar de contestar la llamada y establecer una comunicación con el equipo servidor.

En el caso en que el número que haga la llamada sea del equipo servidor de la central de monitoreo, el “*identificador*” envía la llamada al sistema de monitoreo *SPARTON*, y contesta el modem de las tarjetas monitoras; en caso contrario, en el que el número llamante sea identificado como “*reset*”, la llamada es enviada al “*modulo de reinicio*” y este establece la comunicación.

En la siguiente grafica se observa un esquema general del sistema final desarrollado:

Figura 46. Esquema general del modulo de reinicio.



Una vez el número telefónico que esta haciendo la llamada es transferido al modulo de reinicio por el “*identificador*”, la llamada pasa a ser analizada, existen dos tipos de números telefónicos en el modulo de reinicio:

❖ **Números telefónicos de reinicio directo:** Son los teléfonos fijos o celulares que han sido programados en la memoria interna del microcontrolador para realizar el reinicio en el equipo de monitoreo de forma inmediata, el modulo de reinicio responde a estas llamadas simulado alzar el auricular para hacer creer al

llamante que han contestado la llamada, y hacer creer a la central que un teléfono ha sido levantado.

❖ **Números telefónicos de configuración:** Estos teléfonos realizan la comunicación con el microcontrolador a través del modem para realizar configuración en la memoria interna EEPROM. Estableciendo una comunicación con un ordenador, el sistema permite realizar cambios de los números telefónicos programados y reinicio del sistema.

A continuación se puede apreciar un diagrama (figura 47) que representa el circuito central del proyecto, donde está incluido el microcontrolador PIC16F877 como el procesador central de todo el módulo y el encargado de realizar las órdenes de control y configuración del sistema.

6.1. MICROCONTROLADOR MICROCHIP PIC 16F877

Las características de este microcontrolador son las siguientes:

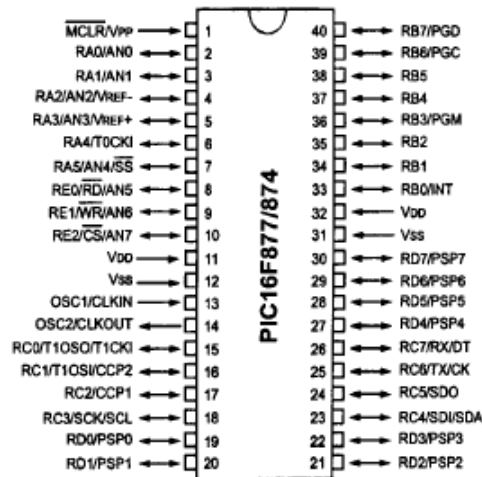
- CPU RISC.
- Sólo 35 instrucciones que aprender.
- Todas las instrucciones se ejecutan en un ciclo de reloj, excepto los saltos que requieren dos.
- Frecuencia de operación de 0 a 20 MHz (DC a 200 nseg de ciclo de instrucción).
- Hasta 8k x 14 bits de memoria Flash de programa.
- Hasta 368 bytes de memoria de datos (RAM).
- Hasta 256 bytes de memoria de datos EEPROM.
- Hasta 4 fuentes de interrupción.
- Stack de hardware de 8 niveles.
- Reset de encendido (POR).
- Timer de encendido (PWRT).
- Timer de arranque del oscilador (OST).
- Sistema de vigilancia Watchdog timer.
- Protección programable de código.
- Modo SEP de bajo consumo de energía.
- Opciones de selección del oscilador.
- Programación y depuración serie "In-Circuit" (ICSP) a través de dos patitas.
- Lectura/escritura de la CPU a la memoria flash de programa.
- Rango de voltaje de operación de 2.0 a 5.5 volts.
- Alta disipación de corriente de la fuente: 25mA.
- Rangos de temperatura: Comercial, Industrial y Extendido.
- Bajo consumo de potencia:
 - Menos de 0.6mA a 3V, 4 Mhz
 - 20 μ A a 3V, 32 Khz
 - Menos de 1 μ A corriente de standby.

Periféricos:

- Timer0: Contador/Temporizador de 8 bits con pre-escalador de 8 bits.
- Timer1: Contador/Temporizador de 16 bits con pre-escalador.
- Timer0: Contador/Temporizador de 8 bits con pre-escalador y post-scalador de 8 bits y registro de periodo.
- Dos módulos de Captura, Comparación y PWM.
- Convertidor Analógico/Digital: de 10 bits, hasta 8 canales.

- Puerto Serie Síncrono (SSP).
- Puerto Serie Universal (USART/SCI).
- Puerto Paralelo Esclavo (PSP): de 8 bits con líneas de protocolo

Figura 48. Configuración de pines del PIC 16F877



El circuito central se encuentra simulado en ISIS PROTEUS y contiene todo el funcionamiento del microcontrolador, la programación interna, la interrupción por los tonos identificados, la secuencia de configuración con el modem, la señal para la conmutación de la línea telefónica, la orden de reinicio de los equipos de monitoreo.

En la figura 47. El circuito se encuentra dividido en varios bloques los cuales representan lo siguiente:

Tono del identificador: números decodificados e identificados provenientes de modulo identificador.

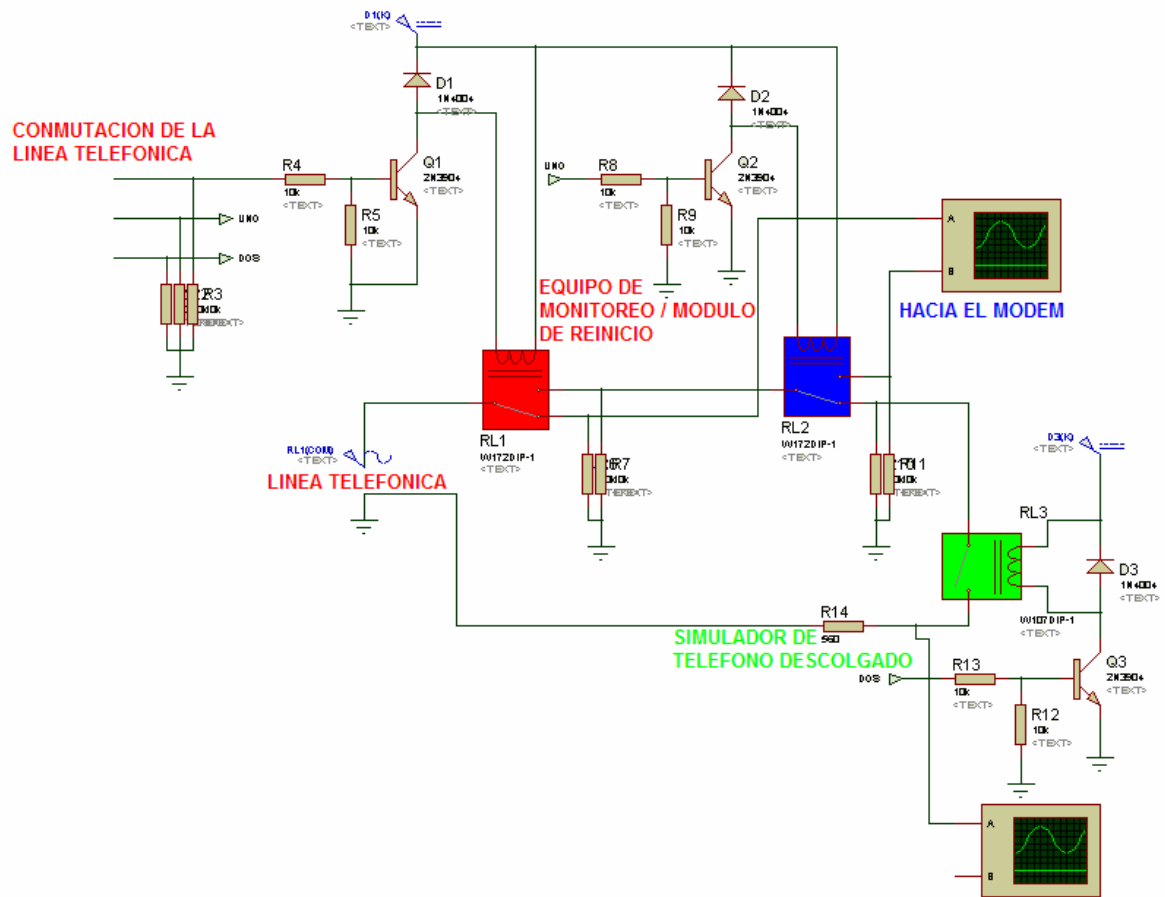
Menú: pulsadores para realizar un control manual sobre el equipo, estos permiten realizar lo siguiente:

- ✓ Visualizar el número identificado.
- ✓ Forzar al módem a entrar en modo de comunicación con el comando ATA.
- ✓ Borrar los registros temporales internos de memoria del microcontrolador.
- ✓ Reiniciar el microcontrolador y la memoria EEPROM interna.

Señal de reset: señal que se envía al circuito de reinicio para realizar el proceso de conmutación sobre la línea de alimentación del equipo de monitoreo y de esta forma reiniciarlo.

Conmutación de la línea telefónica: Circuito encargado de conmutar la línea telefónica entre el módulo de reinicio y el equipo de monitoreo, en conmutar la línea para definir la forma de recepción de la llamada, entre números telefónicos de reinicio directo y números telefónicos de configuración.

Figura 49. Conmutación de la línea telefónica.



La línea telefónica será conmutada por relevos que reciben la orden del microcontrolador de la siguiente forma:

Paso 1. Cuando se produce una llamada al equipo de monitoreo (forma normal). El número se identifica.

Paso 2. Si el numero pertenece a la central para realizar un proceso de monitoreo (lectura de transductores, alarmas, etc.). el primer relevo (rojo) no se activara dejando que el proceso de monitoreo funcione normalmente.

Paso 3. Si el numero identificado es para realizar un reset, entonces este relevo (rojo) desviara la llama hacia el modem o el reset directo (relevo azul) pero esta orden la determina el microcontrolador por el numero telefónico identificado (numero telefónico de reinicio directo o de configuración).

Paso 4. Si el numero es de configuración el relevo azul se activa y enviara la llama al modem para establecer la comunicación con este.

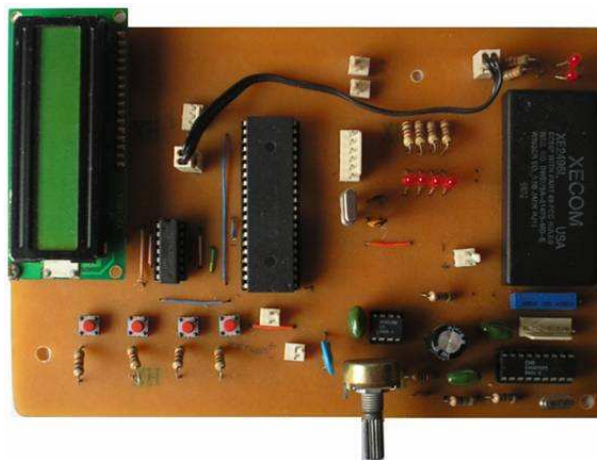
Paso 5. Si el numero es de reinicio directo este relevo azul permanecerá inactivo dejando que la llamada pase al simulador de teléfono descolgado. El último relevo (verde) se encarga de simular contestar la llamada, conmutando la llamada a una resistencia de 600Ω para hacer creer que un teléfono a sido levantado.

NOTA: cuando el usuario llega a este punto el equipo de monitoreo ya esta siendo reiniciado por una orden del microcontrolador, posteriormente hace creer al usuario que el teléfono ha ido descolgado para confirmar la orden de reinicio.

6.2. RESULTADOS FINALES.

Después de diseñar el circuito central en el simulador se realizo el montaje de esta parte dando como resultado lo siguiente:

Figura 50. Circuito central de comunicación.



NOTA: en la figura anterior se observa el circuito central, en este circuito se consideraron elementos que no forman parte esencial en el diseño pero que eran de gran ayuda para el desarrollo del proyecto como LCD, pulsadores, leds indicadores. Proporcionando información del estado del microcontrolador, modem y las rutinas de comunicación.

A continuación se ilustran el circuito impreso diseñado en ARES PROTEUS.

Figura 51. Circuito impreso del circuito central

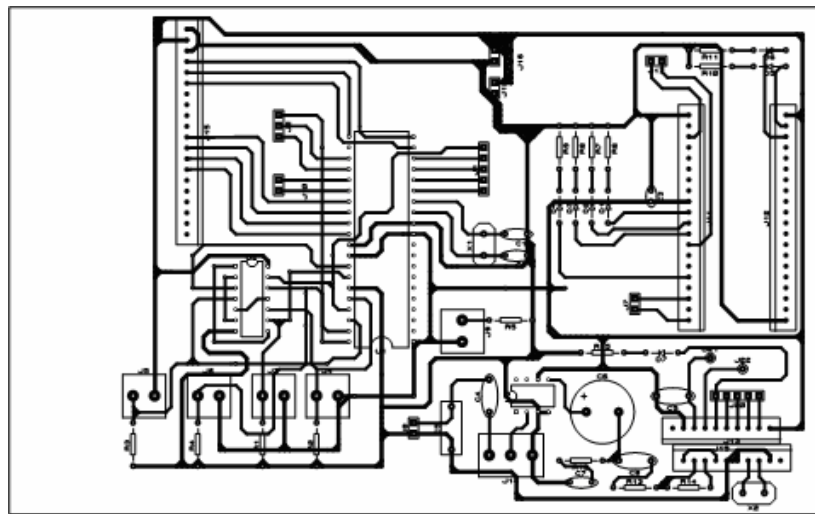


Figura 52. Circuito impreso del circuito central.

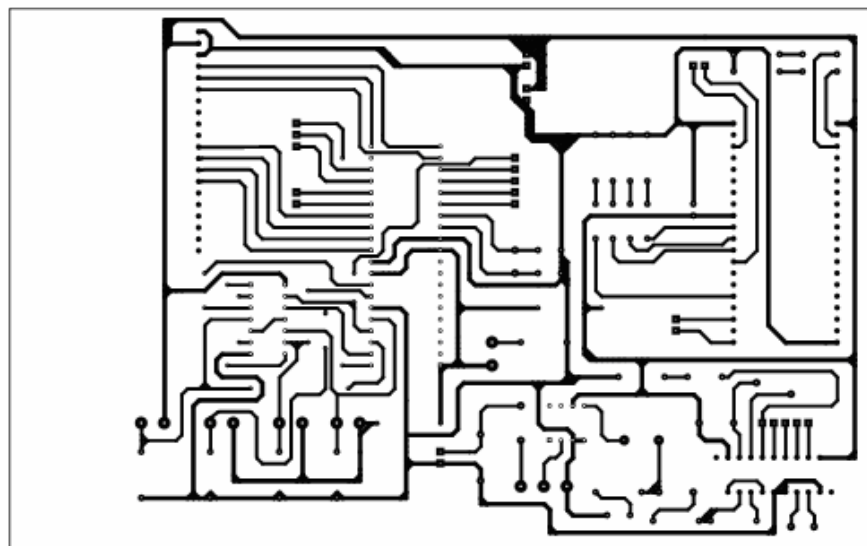


Figura 53. Conmutación de la línea telefónica.

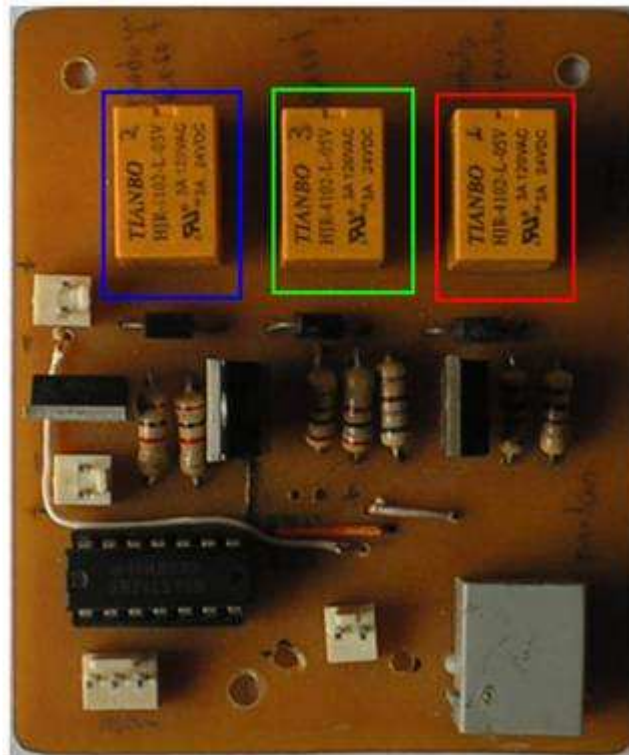
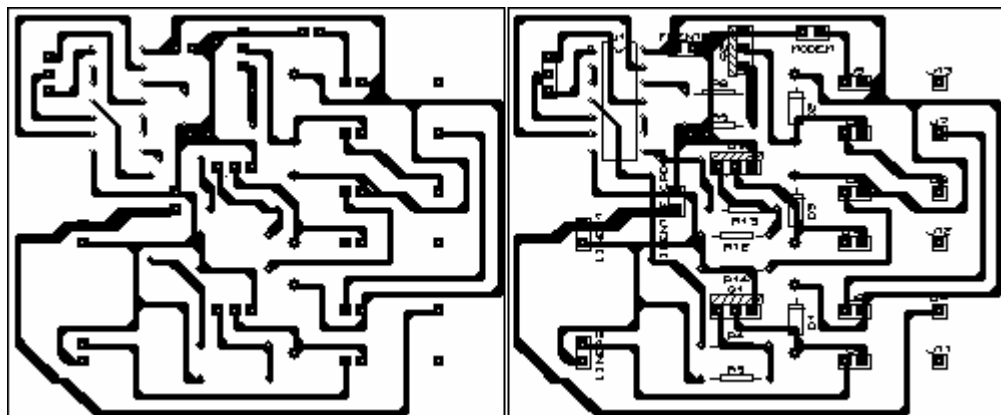


Figura 54. Circuito impreso de conmutación de la línea telefónica



En estas pruebas se presentaron algunos problemas, tales como:

- ✓ El ensamble del circuito central presentaba ciertos inconvenientes en el momento de realizar las pruebas por fallas en el montaje y soldadura de los componentes.

✓ Por ser el circuito central debía funcionar correctamente para ello debíamos realizar pruebas del funcionamiento del controlador como prueba de los puertos, interrupciones, temporizadores, comunicación serial, lo que retrazaba un poco el diseño del prototipo.

✓ La parte mas critica de esta parte era el software desarrollado y cargado en el microcontrolador, ya que el programa iba siendo mejorado corrigiendo los fallos de sincronización del microcontrolador con el LCD, la comunicación serial con el modem, y fallos que se iban presentando.

Lo que conllevaba a esta extrayendo el microcontrolador de la tarjeta para ser reprogramado y reiniciar las pruebas nuevamente, hasta satisfacer todo lo errores y necesidades.

7. PROGRAMA DE MANEJO DEL MÓDULO DE REINICIO

El manejo del módulo de reinicio por medio de un software surge de la necesidad de poder hacer la configuración del mismo remotamente. Ya que ocasionalmente se necesitará configurar los números telefónicos que harán el reset directo y el reset o configuración usando el módem equipado en el prototipo.

Este programa ha sido desarrollado usando un lenguaje de programación muy versátil como es Visual Basic, ya que provee todas las facilidades requeridas para desarrollar aplicaciones complejas en poco tiempo comparado con lo que demora hacerlo en otros lenguajes de programación como por ejemplo C++ ó Visual C++, aunque con una menor velocidad o eficiencia en las aplicaciones.

Visual Basic es un lenguaje *basado en objetos*, aunque no *orientado a objetos* como C++ o Java. La diferencia es que Visual Basic utiliza objetos con propiedades y métodos, pero carece de mecanismos de herencia y polimorfismo propios de los verdaderos lenguajes orientados a objetos como C++ y Java.

7.1. FUNCIONAMIENTO

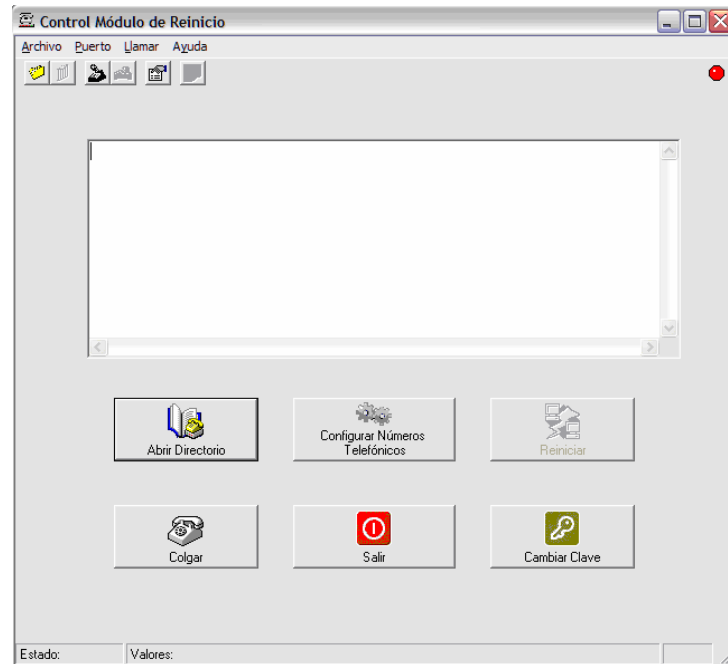
Al iniciar el programa lo primero que aparece es una pantalla de presentación con el logo de la empresa y los datos del programa, etc.

Figura 55. Pantalla de bienvenida.



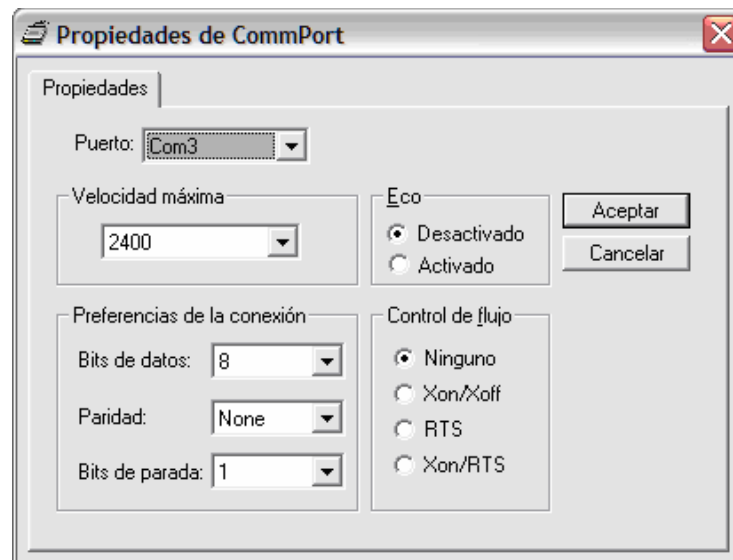
Este es la pantalla principal del programa:

Figura 56. Menú principal del programa.



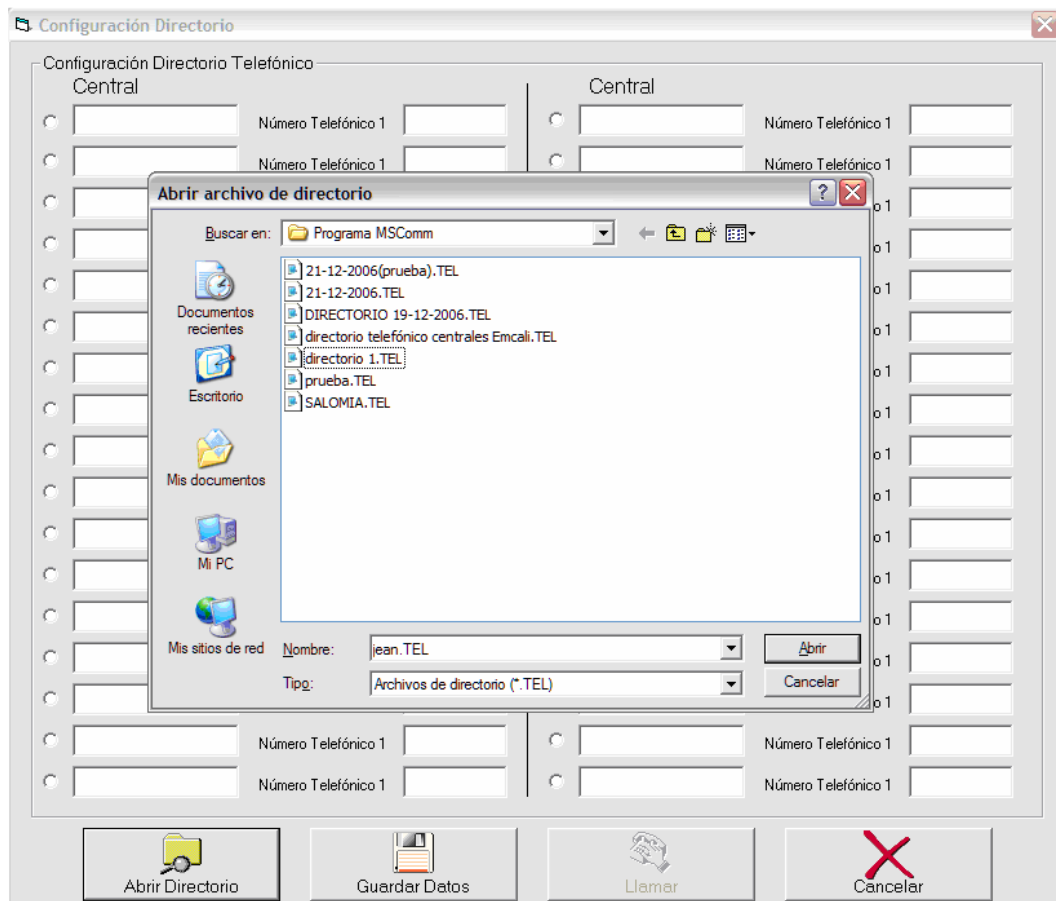
Aquí se puede configurar el puerto serie del computador al cual esta conectado el módem del equipo.

Figura 57. Configuración del puerto serie.



También se puede abrir el directorio telefónico para hacer la comunicación al módulo de reinicio:

Figura 58. Directorio telefónico de las centrales.



Después de estar conectado al módulo de reinicio puede configurarle los números telefónicos que usará para comunicarse con él, ya sea por medio de módem o si prefiere que se haga un reset directo.

✓ **Reset Directo:** Es el tipo de reset que se hace usando solamente un aparato telefónico normal, ya sea desde un teléfono fijo o desde un teléfono celular.

✓ *Reset por Módem:* es el tipo de reset que se hace estableciendo primero una comunicación por módem, también se usa para configurar el módulo de reinicio y abre una posibilidad futuras de monitorear el voltaje, corriente, temperatura, etc.

Figura 59. Números telefónicos almacenados en el PIC.

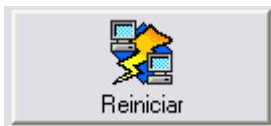
Configuración de los números telefónicos

CONFIGURACIÓN DE NÚMEROS TELEFÓNICOS EN EL MÓDULO DE REINICIO

Tipo de Reset		Tipo de Reset	
Número Tel.	[Dropdown]	Número Tel.	[Dropdown]
Número Tel.	[Dropdown]	Número Tel.	[Dropdown]
Número Tel.	[Dropdown]	Número Tel.	[Dropdown]
Número Tel.	[Dropdown]	Número Tel.	[Dropdown]
Número Tel.	[Dropdown]	Número Tel.	[Dropdown]
Número Tel.	[Dropdown]	Número Tel.	[Dropdown]
Número Tel.	[Dropdown]	Número Tel.	[Dropdown]
Número Tel.	[Dropdown]	Número Tel.	[Dropdown]
Número Tel.	[Dropdown]	Número Tel.	[Dropdown]
Número Tel.	[Dropdown]	Número Tel.	[Dropdown]

Consultar Abrir Lista Guardar esta Lista Enviar Cancelar

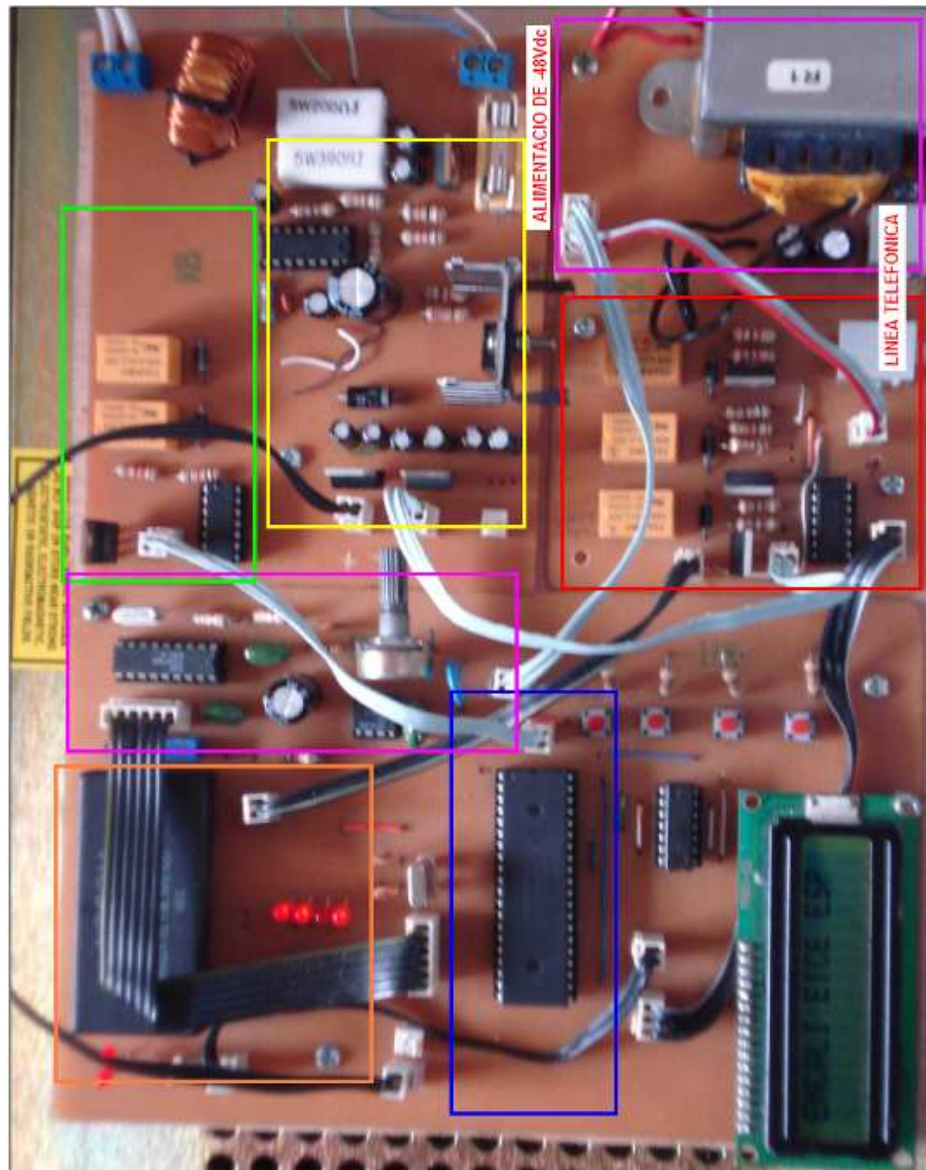
Para finalizar, se tiene el botón que hace el reinicio del equipo de monitoreo Sparton, este botón sólo estará habilitado cuando se esté conectado a un módulo de reinicio.



Además el programa también cuenta con funciones para guardar las listas de números telefónicos, cargar esos datos para mayor facilidad, cambiar la clave de acceso.

8. SISTEMA FINAL DESARROLLADO

Figura 60. Prototipo desarrollado.









- | | |
|---|--|
|  Microcontrolador. |  Conmutación de la línea telefónica. |
|  Módem. |  Convertidor dc-dc |
|  Identificador. |  Circuito de reinicio en los -48 Vdc. |

Figura 61. Prototipo identificando una llamada.

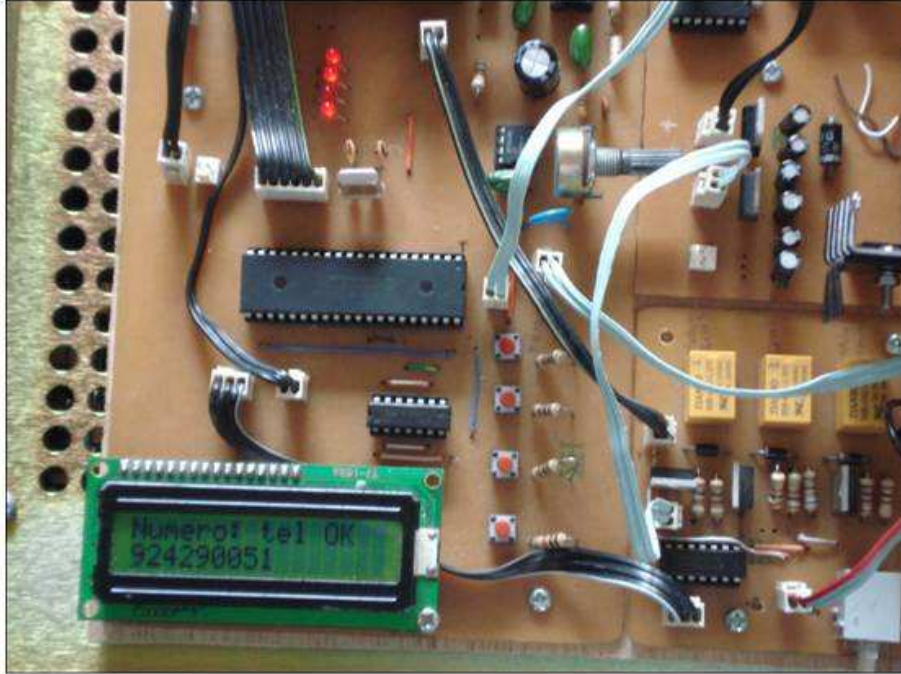
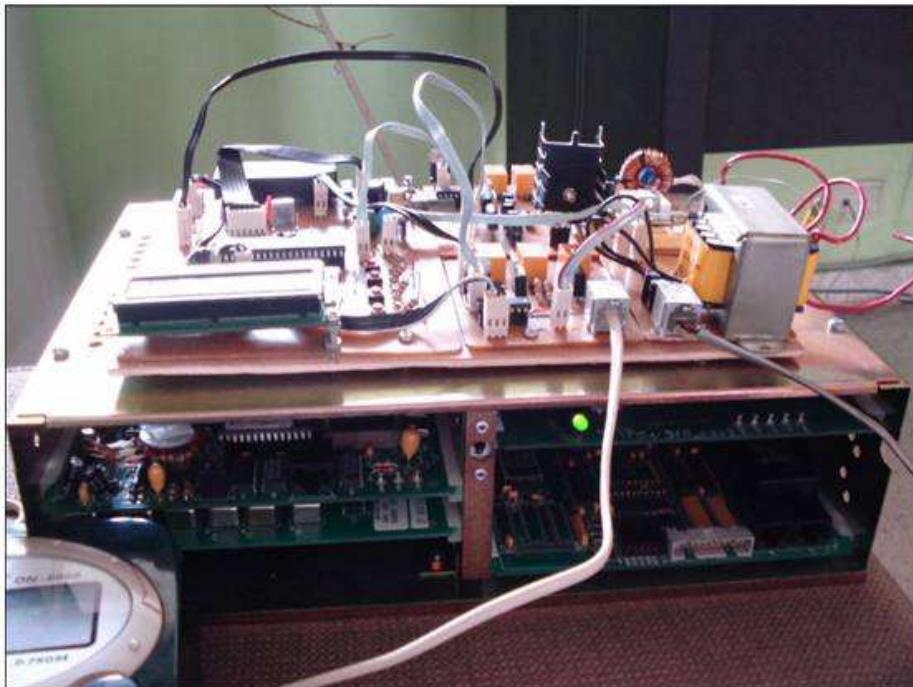


Figura 62. Prototipo en funcionamiento.



9. COSTOS DEL PROTOTIPO

La siguiente tabla muestra el costo detallado de los elementos empleados en la elaboración del prototipo:

Tabla 7. Costos del prototipo.

Descripción	Cantidad	Costo/unidad	Total
Microcontrolador PIC16F877.	1	\$18000	\$18.000
Cristal 4MHz	1	\$1400	\$1.400
Condensador de 22pf	2	\$80	\$160
Modem XECOM XE2496	1	\$120000	\$120000
Potenciómetro 100k Ω	1	\$800	\$800
AO LM386	1	\$1500	\$1.500
Resistencia de 100 k Ω	2	\$150	\$300
Resistencia de 10 Ω	1	\$100	\$100
Resistencia de 300k Ω	1	\$150	\$150
Condensador 100nf	3	\$150	\$450
Condensador 250uF	1	\$300	\$300
Varistor de 100V	1	\$800	\$800
Decodificador de tonos dtmf CM8870	1	\$2700	\$2.700
Cristal de 3.5795 Mhz	1	\$1150	\$1.150
Relevos de 110Vac	5	\$1800	\$9.000
TIP31C	4	\$1400	\$5.600
Diodo 1n4004	4	\$50	\$200
Resistencia 2k Ω	9	\$100	\$900
Resistencia de 600 Ω	1	\$100	\$100
Compuerta 74LS14	2	\$1100	\$2.200
Puerto RJ11	2	\$800	\$1.600
Condensador 22uf 100V	2	\$400	\$800
Transformador	1	\$2800	\$2.800
Regulador 7805	2	\$1100	\$2.200
Condensador 0.1uf 50v	6	\$100	\$600
Condensador 2.2uf 160v	1	\$300	\$300
Condensador 47uf 160v	1	\$600	\$600
Resistencia 200 Ω 5W	1	\$300	\$300
Resistencia 390 Ω 5W	1	\$300	\$300
Regulador 7815	1	\$1050	\$1.050

Transistor TIP127	1	\$1150	\$1.150
Fusible	1	\$150	\$150
Porta fusible	1	\$500	\$500
SG3524	1	\$2600	\$2.600
Reasistencia 5k Ω	3	\$150	\$450
Resistencia 470 Ω	1	\$100	\$100
Resistencia 20k Ω	1	\$150	\$150
Resistencia 330 Ω	1	\$100	\$100
Condensador 0.001uf	1	\$100	\$100
Bobina de 250Uh	1	\$1000	\$1.000
Diodo FR307	1	\$550	\$550
Interruptor	1	\$700	\$700
Base para impreso 40pines	2	\$400	\$800
Base para impreso 18pines	1	\$400	\$400
Base para impreso 16pines	1	\$400	\$400
Plaqueta cobrizaza	1	\$5000	\$5.000
TOTAL			\$190510
Pulsadores	4	\$500	\$2.000
Resistencia de 1k Ω	10	\$150	\$1.500
Diodo led	6	\$150	\$900
LCD 2X16	1	\$18500	\$18.500
Compuerta 74LS32	1	\$1000	\$1.000
Conectores de 2 pines hembra y macho	14	\$400	\$5.600
Conectores de 6 pines hembra y macho	2	\$800	\$1.600
Conectores de 3 pines hembra y macho	2	\$600	\$1.200
TOTAL			\$32.300
TOTAL			\$222810

NOTA: Los valores de los elementos en rojo representan los elementos que son indispensables en la elaboración del proyecto, por lo tanto son obligatorios, los elementos que aparecen en azul no son necesarios, son elementos que consideramos para poder desarrollar el prototipo pero que no forman parte esencial del dispositivo final.

10. CONCLUSIONES

- ▶ Por medio de la realización de este proyecto como opción de grado adquirimos experiencia y conocimiento en el campo laboral y en el sector de telecomunicaciones en donde aplicamos lo aprendido en la carrera universitaria.
- ▶ Se dio solución satisfactoriamente a una problemática planteada por la empresa EMCALI.
- ▶ La elaboración de este proyecto es muy significativo para EMCALI porque permite a los operarios del sistema de monitoreo de la red primaria tener una herramienta más para corregir las condiciones adversas de trabajo y de esta forma ayudar al buen estado de las líneas telefónicas y el mantenimiento preventivo.
- ▶ Por medio de la creación de un canal de comunicación independiente entre la unidad de monitoreo Sparton y la central telefonica, logramos crear un enlace para cumplir nuestros objetivos, ademas este canal de comunicación puede ser acondicionado y utilizado para monitorear el estado de variables como voltaje, corriente y temperatura en el equipo esparton, todo con el fin de garantizar el estado optimo de los equipos.
- ▶ Lograr crear la alimentación de voltaje de nuestro prototipo representa una etapa importante en nuestro diseño, debido que nos permitio identificar inconvenientes como la diferencia de potencial entre las tierras de voltaje de la central de -48Vcd y la tierra de la linea telefonica, que eran provenientes de la misma central pero que no estaban a una misma referencia, ocasionando problemas de funcionamiento en nuestro prototipo, pero que fueron solucionadas satisfactoriamente con metodos de aislamiento galvanico.

11. RECOMENDACIONES

- ◆ Se recomienda un seguimiento periódico al prototipo para detectar anomalías que se puedan corregir para la elaboración de más dispositivos como este.
- ◆ Si se desea agregarle a este dispositivo otras funciones como monitorear temperaturas, voltajes o corrientes, se puede llegar a hacer ya que cuenta con los medios para lograr transmitir más información hacia la central.
- ◆ **Importante:** Para que el modulo de control de reinicio actúe sobre el numero telefónico que esta intentando acceder al sistema el recurso de identificación de llamadas debe estar disponible en la línea telefónica, por lo tanto es necesario estar habilitado el servicio por la empresa de telefonía.

BIBLIOGRAFÍA

ANGULO USATEGUI, Jose Manuel. Microcontroladores PIC Diseño práctico de aplicaciones. Bogotá: Mc Graw Hill, 1999. 389 p.

CallID Unit based on Motorola MC68HC11A1P & MC145447P chips compatible with the standard protocols [en línea]. Paris: Matthieu Benoit Electrical Engeneering Home Page, 2007. [Consultado 11 de Enero de 2007]. Disponible por Internet: <http://matthieu.benoit.free.fr/cidfr.htm> caller ID display service

Circuito detector de tonos DTMF [en línea]. Buenos Aires: Portal de Tecnología de Habla Hispana, 2006. [Consultado 25 de Septiembre de 2006]. Disponible por internet: <http://www.pablin.com.ar/electron/circuito/telefon/decdtmf/index.htm>

Circuito simulador de teléfono descolgado [en línea]. Buenos Aires: Portal de Tecnología de Habla Hispana, 2006. [Consultado 25 de Septiembre de 2006]. Disponible por internet: <http://www.pablin.com.ar/electron/circuito/telefon/hangup/index.htm>

Comunicação Com A Porta Serial [en línea]. Brasilia: Antonio Rogério Messias ROGERCOM, 2007. [Consultado 30 de Septiembre de 2006]. Disponible por Internet: <http://www.rogercom.com/index.htm>

Consulta de Hojas de datos de dispositivos electrónicos, LM3524, CM8870, TIP31C, XE2496 [en línea]. Miami: Alldatasheet.com 2003 – 2007. Buscador de hojas de datos, 2006. [Consultado 26 de Junio de 2006]. Disponible por Internet: <http://www.alldatashet.com>

CREUS, Antonio. Instrumentación industrial. 6 ed. Madrid: Alfaomega, 1998. 620 p.

DAUENPORT, William. Comunicaciones Modernas de Datos. Madrid: McGraw Hill, 1997. 654 p.

GONZÁLEZ DE LA GARZA, Manuel. Módems. Barcelona: Prentice Hall Hispanoamericana, 1999. 523 p.

HAKALA, David. Módems a su alcance. Madrid: McGraw Hill, 1988. 324 p.

HUIDOBRO, Jose Manuel. Manual de Telecomunicaciones. Madrid: Alfaomega, 2001. 132 p.

Módulo de Telecomunicaciones [en línea]. Trujillo: Escuela Académico Profesional de Informática Universidad Nacional de Trujillo, 2206. [Consultado 07 de Octubre de 2006]. Disponible por Internet:
<http://www.inf.unitru.edu.pe/docs/telep/cap5/index.html#modula>

PALLAS, Ramón. Sensores y Acondicionadores de Señal. Madrid: Alfaomega, 1995. 523 p.

Qué es un Módem [en línea]. Puerto Rico: Pontificia Universidad Católica Facultad de Ingenierías, 2001. [Consultado 30 de Noviembre de 2006]. Disponible por Internet: <http://www.pucpr.edu/facultad/apagan/que-es/modem.htm>

RASHID, Muhammad H. Electrónica de Potencia circuitos, dispositivos y aplicaciones. 6 ed. México: Prentice Hall, 2004. 904 p.

Red telefónica básica [en línea]. Valencia: Futurnet & Globalnet S.A., 2006. [Consultado 24 de Septiembre de 2006]. Disponible por Internet: <http://www.futurnet.es/adsl/intro/index.htm>

ANEXOS

Anexo A. Sistema de monitoreo - red primaria Modelo 5325b

OBJETIVOS

- * Comprender las ventajas de monitorear el estado de los cables telefónicos
- * Identificar los elementos constitutivos del sistema automático de control y su funcionamiento

OBJETIVOS

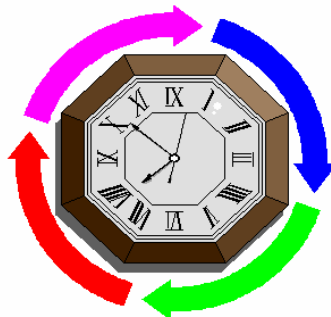
- * Aprender a direccionar transductores con base en el código binario
- * Interpretar los diferentes reportes generados por el equipo de monitoreo

DEFINICIÓN

- * El monitoreo de la red es un sistema automático de control que se encarga de supervisar remotamente el estado de los cables telefónicos en cuanto a presión, flujo y humedad para prevenir daños masivos

VENTAJAS

- * Permite visualizar el estado de los cables en cada central
- * Programar el mantenimiento y tomar decisiones según las prioridades
- * Facilita la localización de fugas neumáticas
- * Incremento en la productividad de los grupos de mantenimiento preventivo
- * Disminución de los costos de mantenimiento correctivo
- * Establece criterios para determinar la reposición de redes por envejecimiento
- * Permite en algunos casos determinar el sitio de corte de un cable por vandalismo



MONITOREO CONSTANTE LAS 24 HORAS DEL DIA

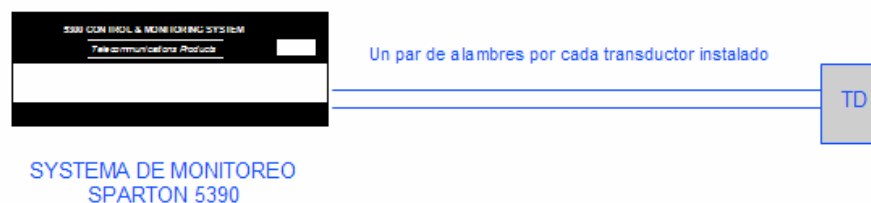
TIPOS DE MONITOREO

- En pares Dedicados
- En pares Suscriptores
- En pares Direccionables
- En pares Binarios

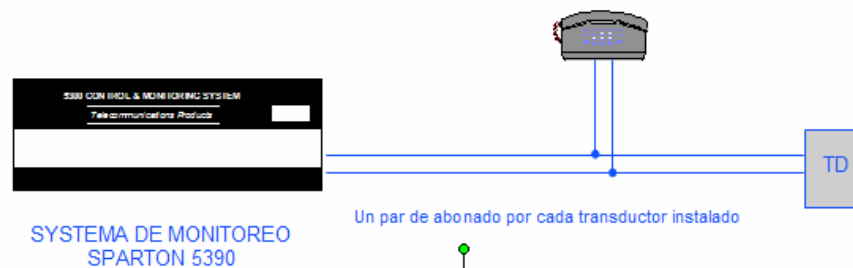
PARES DEDICADOS

Este método de monitorear los transductores sería muy eficiente ya que se tiene un par de alambres por cada transductor y si algo le sucede a un par de alambres sólo afectaría a un solo transductor.

La desventaja es que se utilizan muchos pares de los cables telefónicos para este propósito de monitorear transductores.



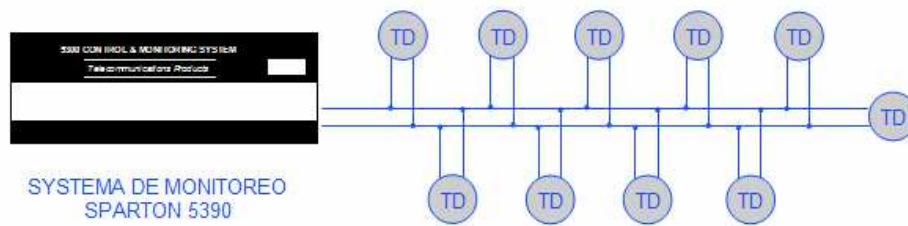
PARES SUSCRIPTORES



Este método de monitorear los transductores no es muy eficiente ya que se comparte el par alambres con el abonado. En muchos lugares esto no se usa en ningún caso.

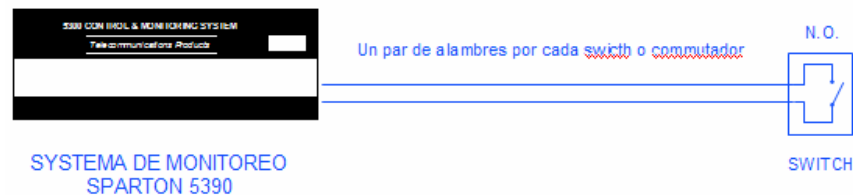
La mayor desventaja es que el abonado conecta muchos aparatos de diversos usos en la misma línea, causando problemas para el sistema de monitoreo.

PARES DIRECCIONABLES



Este método de monitorear los transductores sería el más eficiente ya que utiliza una mínima cantidad de pares de alambres para monitorear todo tipo de transductor. En un sistema de cuatro (4) entradas se podría monitorear 512 sensores (4 X 128 = 512).

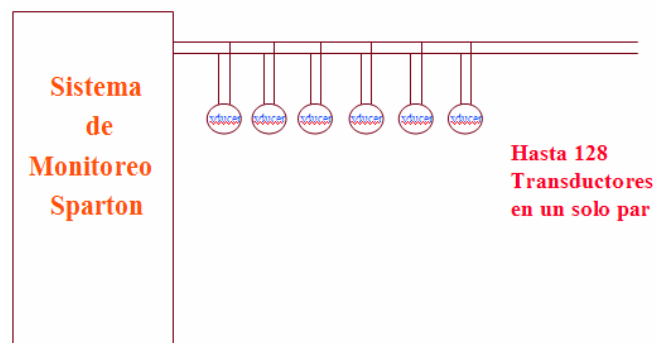
PARES BINARIOS



El sistema puede monitorear el estado de un switch. No importa si el estado normal del switch es ABIERTO o CERRADO, se puede monitorear cualquiera de los dos casos. Se podría monitorear cualquier aparato que cambie el estado de un switch, como los compresores de aire en una central telefónica o el estado de una puerta o ventana (abierta o cerrada).

TECNOLOGÍA DIRECCIONABLE

Se pueden conectar 128 Transductores en un solo par de alambres de cobre

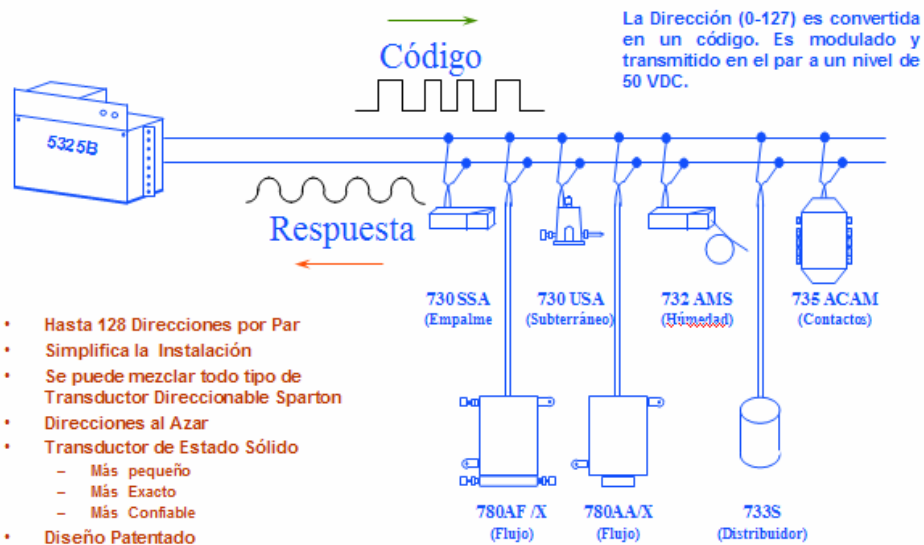


- Los Transductores pueden ser instalados en un sólo par hasta una distancia de 90 Millas o 145 Km
- Tecnología de Estado Sólido provee:
 - Más Confiabilidad
 - Más Exactitud de Medición (1%)
- Inversión mínima de Cobre
- El código binario del Transductor a medirse es transmitido por el sistema de Monitoreo a través del par, es una señal cuadrada de baja frecuencia



- El transductor con el código binario correspondiente responde con una señal FM (frecuencia modulada)

20 HZ - 40HZ
9.5 psi - 0.0 psi



Anexo B. Código Programación microcontrolador

El siguiente anexo contiene el código fuente programado en el microcontrolador:

```
#include <16f877.h>
#include <math.h>
#include <lcd2.h>
#include <serial.h>
#include <dtmf.h>
#fuses XT,NOPROTECT,NOLVP,NOWDT,NOBROWNOUT
#use delay(clock=4000000)
#use rs232(baud=2400, xmit=PIN_C6, rcv=PIN_C7)// RS232 Estándar

int temp, intento; //variables para el timer

////////////////////////////////////
#INT_TIMER1
void tiempo()
{
    set_timer1(0x0000);
    temp=temp+1;
}
//-----int. por recepcion de datos-----
#int_RDA
void serial_isr() {           // Interrupción recepción serie USART
    rcvchar=0x00;           // Inicializo caracter recibido

    if (xbuff>=0x0F){
        inicbuff();
    }
    if(kbhit()){           // Si hay algo pendiente de recibir ...
        rcvchar=getc();           // lo descargo y ...
        addcbuff(rcvchar);           // lo añado al buffer y ...
    }
}

//-----seleccion de menu por el tipo de int.-----
#INT_EXT
void isr_ext(){
    if (input(PIN_B1)){ //Pin_B1 = Contestar ring llamada entrante
        menu=4;
    }
    if (input(PIN_B3)){
```

```

    menu=8;    //reset numeros telefonicos guardados en memoria
}
if (input(PIN_B4)){ //Pin_B4 = entrada de numero llamante
    tonos();    //numeros DTMF
}
if (input(PIN_B5)){
    menu=7;    //VISUALIZAR TONOS menu=7
}
if (input(PIN_B6)){
    menu=1;    //BORRAR TONOS menu=1
}
}

////////////////////////////////////
conf_inicial(){

    output_bit(PIN_B7,0);
    output_bit(PIN_B2,0);

    SET_TRIS_D(0x00);
    SET_TRIS_B(0x7D);
    SET_TRIS_C(0x8F);

    lcd_init();

    setup_timer_1(T1_INTERNAL | T1_DIV_BY_8);
    set_timer1(0x0000);

    enable_interrupts(int_rda);
    enable_interrupts(INT_EXT);
    enable_interrupts(global);

//-----cifiguracion inicial del modem-----

    printf("ATZ\r"); //reset modem
    delay_ms(500);

    inicbuff(); // Borra buffer al inicio

    menu=0;

```

```

        enable_interrupts(INT_TIMER1);
unoatv:
    printf("ATV\r"); //respuesta del modem: numerica
    delay_ms(500);
uno:  modemrecive();
    if (menu==5){
        if (temp>=6){
            temp=0;
            if (++intento==3){
                menu=9;
                disable_interrupts(INT_TIMER1);
                goto fallo;
            }
            goto unoatv;
        }
        goto uno;
    }
    intento=0;
    temp=0;
    disable_interrupts(INT_TIMER1);
    inicbuff();

```

```

        enable_interrupts(INT_TIMER1);
dosate:
    printf("ATE\r"); //modem respesta sin eco
    delay_ms(500);
dos:  modemrecive();
    if (menu==5){
        if (temp>=6){
            temp=0;
            if (++intento==3){
                menu=9;
                disable_interrupts(INT_TIMER1);
                goto fallo;
            }
            goto dosate;
        }
        goto dos;
    }
    intento=0;
    temp=0;
    disable_interrupts(INT_TIMER1);

```

```

fallo:
    lcd_gotoxy(1,1);
    printf(lcd_putc,"ESTADO SISTEMA");
    delay_ms(2000);

}
////////////////////////////////////inicio del programa////////////////////////////////////
void main() {

    conf_inicial();

inicio:

switch (menu){
case 0:
    lcd_init();
    printf(lcd_putc,"EMCALI EICE ESP");
    delay_ms(2000);
    break;
case 1:
    borrado(); //rutina dtmf
    menu=0;
    break;
case 2:
    identificar(); //compara el numero identificado con los numeros almacenados
    borrado();
    break;
case 3:
    comunicacion();
    break;
case 4: //cuando se detecta un timbre el modem se pone en modo
    lcd_gotoxy(1,2); //respuesta
    printf(lcd_putc,"ATA ");
    inicbuff();
    enable_interrupts(INT_TIMER1);
    printf("ATA\r"); //Modem modo respuesta ATA
    delay_ms(100);
estadoata:
    modemrecive();
    if (menu==5){
        if (temp>=120){
            temp=0;

```

```

        menu=0;
        disable_interrupts(INT_TIMER1);
        output_bit(PIN_B7,0);
        output_bit(PIN_C4,0);
        output_bit(PIN_C5,0);
        goto tiempoata;
    }
    goto estadoata;
}
tiempoata:
    intento=0;
    temp=0;
    disable_interrupts(INT_TIMER1);
    break;
case 5:
    modemrecive();
    break;
case 6:
    break;
case 7:
    visualizar();
    break;
case 8:
    resettel();
    break;
case 9:
    lcd_gotoxy(1,2);
    printf(lcd_putc,"FALLO CON MODEM");
    delay_ms(2000);
    break;
default:
    break;
}
goto inicio;
}

```

```

+++++
+++++
+++++
#include <string.h>
#include <math.h>

```



```

int const teldato=12;
int k=0, n=0, num=15;
int dato[teldato], telefono[teldato];
int1 trama=0;
int contatele, posiciotele;
//-----
tonos(){
    num=(input(PIN_C0))+(input(PIN_C1)*2)+(input(PIN_C2)*4)+(input(PIN_C3)*8);
    if ((num >= 1) && (num <= 9) && (trama==1)){
        dato[k]=num+48;    //convierte los numeros a ascii
        k=k+1;
        goto tonosiguiente;
    }
    if (num==0){ //D
        trama=1;
        goto tonosiguiente; //inicio de trama el numero pasa a ser identificado
    }
    if ((num==13)&&(trama==1)){    // A
        dato[k]=65;
        k=k+1;
        goto tonosiguiente;
    }
    if ((num==11)&&(trama==1)){
        dato[k]=42; /*
        k=k+1;
        goto tonosiguiente;
    }
    if ((num==12)&&(trama==1)){
        dato[k]=35; //#
        k=k+1;
        goto tonosiguiente;
    }
    if ((num==14)&&(trama==1)){ //B
        dato[k]=66;
        k=k+1;
        goto tonosiguiente;
    }
    if ((num==15)&&(trama==1)){ //C
        trama=0;
        menu=2;
        goto finprotocolo;
    }
    if ((num==10)&&(trama==1)){

```

```

        dato[k]=48;
        k=k+1;
        goto tonosiguiente;
    }
tonosiguiente:
    menu=0;
finprotocolo:
    #asm
    nop;
    #endasm
}
//-----identificacion del # telefonico-----
identificar(){
lcd_putc('\f');
lcd_gotoxy(1,1);
printf(lcd_putc,"Numero:");

for (n=0;n<=teldato;++n){
    telefono[n]=0;
}

for (n=0;n<k;++n){
    lcd_gotoxy(n+1,2);
    printf(lcd_putc,"%c", dato[n]);
}

contatele=0;
posiciotele=0;

nuevonumero:

for (n=0;n<k;++n){
    telefono[n] = read_EEPROM (n+posiciotele);
}
if ((strcmp(telefono, dato))==0){
    lcd_gotoxy(9,1);
    printf(lcd_putc,"tel OK");
    if (read_EEPROM (posiciotele+k)=='M'){
        output_bit(PIN_B7,1);
        delay_ms(2000);
        output_bit(PIN_C4,1);
        delay_ms(5000);
        menu=4;
    }
}

```

```

        goto telmodem;
    }
    else{
        output_bit(PIN_B2,1);
        delay_ms(6000);
        output_bit(PIN_B7,1);
        delay_ms(1000);
        output_bit(PIN_C5,1);
        goto teltele;
    }
}
else{
    if (++contatele>=21){
        lcd_gotoxy(9,1);
        printf(lcd_putc,"Malo ");
        delay_ms(3000);
        menu=0;
        goto telmodem;
    }
    else{
        posiciotele=posiciotele+12;
        goto nuevonumero;
    }
}

teltele:
    menu=0;
    delay_ms(3000);
    output_bit(PIN_B7,0);
    output_bit(PIN_C4,0);
    output_bit(PIN_C5,0);
    output_bit(PIN_B2,0);

telmodem:
    #asm
    nop;
    #endasm
}
//-----
borrado(){
for (n=0;n<=teldato;++n){
    dato[n]=0;
    telefono[n]=0;
}
}

```

```

n=0;
k=0;
}
//-----
resettel(){
menu=0;
for (n=0;n<250;++n){
    write_eeprom(n,0x00);
}
for (n=0;n<=teldato;++n){
    telefono[n]=0;
}
sprintf(telefono,"924290059M");
for (n=0;n<teldato;++n){
    write_eeprom(n,telefono[n]);
}
for (n=0;n<=teldato;++n){
    telefono[n]=0;
}
sprintf(telefono,"924290057");
for (n=0;n<teldato;++n){
    write_eeprom(12+n,telefono[n]);
}
for (n=0;n<=teldato;++n){
    telefono[n]=0;
}
sprintf(telefono,"924290056");
for (n=0;n<teldato;++n){
    write_eeprom(24+n,telefono[n]);
}
for (n=0;n<=teldato;++n){
    telefono[n]=0;
}
sprintf(telefono,"924290058M");
for (n=0;n<teldato;++n){
    write_eeprom(36+n,telefono[n]);
}
for (n=0;n<=teldato;++n){
    telefono[n]=0;
}
sprintf(telefono,"924290055");
for (n=0;n<teldato;++n){
    write_eeprom(48+n,telefono[n]);
}

```

```

    }
    for (n=0;n<=teldato;++n){
        telefono[n]=0;
    }
    sprintf(telefono,"924290054");
    for (n=0;n<teldato;++n){
        write_eeprom(60+n,telefono[n]);
    }
    for (n=0;n<=teldato;++n){
        telefono[n]=0;
    }
    sprintf(telefono,"924290053");
    for (n=0;n<teldato;++n){
        write_eeprom(72+n,telefono[n]);
    }
    for (n=0;n<=teldato;++n){
        telefono[n]=0;
    }
    sprintf(telefono,"924290052");
    for (n=0;n<teldato;++n){
        write_eeprom(84+n,telefono[n]);
    }
    for (n=0;n<=teldato;++n){
        telefono[n]=0;
    }
    sprintf(telefono,"924290051");
    for (n=0;n<teldato;++n){
        write_eeprom(96+n,telefono[n]);
    }
    for (n=0;n<=teldato;++n){
        telefono[n]=0;
    }
    sprintf(telefono,"93136469468");
    for (n=0;n<teldato;++n){
        write_eeprom(108+n,telefono[n]);
    }
    for (n=0;n<=teldato;++n){
        telefono[n]=0;
    }
    sprintf(telefono,"924290050");
    for (n=0;n<teldato;++n){
        write_eeprom(120+n,telefono[n]);
    }

```

```

for (n=0;n<=teldato;++n){
    telefono[n]=0;
}
sprintf(telefono,"921111111");
for (n=0;n<teldato;++n){
    write_eeprom(132+n,telefono[n]);
}
for (n=0;n<=teldato;++n){
    telefono[n]=0;
}
sprintf(telefono,"922222222");
for (n=0;n<teldato;++n){
    write_eeprom(144+n,telefono[n]);
}
for (n=0;n<=teldato;++n){
    telefono[n]=0;
}
sprintf(telefono,"923333333");
for (n=0;n<teldato;++n){
    write_eeprom(156+n,telefono[n]);
}
for (n=0;n<=teldato;++n){
    telefono[n]=0;
}
sprintf(telefono,"924444444");
for (n=0;n<teldato;++n){
    write_eeprom(168+n,telefono[n]);
}
for (n=0;n<=teldato;++n){
    telefono[n]=0;
}
sprintf(telefono,"925555555");
for (n=0;n<teldato;++n){
    write_eeprom(180+n,telefono[n]);
}
for (n=0;n<=teldato;++n){
    telefono[n]=0;
}
sprintf(telefono,"926666666M");
for (n=0;n<teldato;++n){
    write_eeprom(192+n,telefono[n]);
}
for (n=0;n<=teldato;++n){

```

```

        telefono[n]=0;
    }
    sprintf(telefono,"927777777");
    for (n=0;n<teldato;++n){
        write_eeprom(204+n,telefono[n]);
    }
    for (n=0;n<=teldato;++n){
        telefono[n]=0;
    }
    sprintf(telefono,"928888888");
    for (n=0;n<teldato;++n){
        write_eeprom(216+n,telefono[n]);
    }
    for (n=0;n<=teldato;++n){
        telefono[n]=0;
    }
    sprintf(telefono,"929999999");
    for (n=0;n<teldato;++n){
        write_eeprom(228+n,telefono[n]);
    }
    for (n=0;n<=teldato;++n){
        telefono[n]=0;
    }
    sprintf(telefono,"920000000");
    for (n=0;n<teldato;++n){
        write_eeprom(240+n,telefono[n]);
    }
    for (n=0;n<=teldato;++n){
        telefono[n]=0;
    }
    menu=0;
}
//-----
visualizar(){
    lcd_putc('\f');
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Numero:");
    for (n=0;n<k;++n){
        lcd_gotoxy(n+1,2);
        printf(lcd_putc,"%c", dato[n]);
    }
    delay_ms(2000);
    menu=0;
}

```

```

}
+++++
+++++
+++++
#use delay(clock=4000000)
#use rs232(baud=2400, xmit=PIN_C6, rcv=PIN_C7)
#include <string.h>
#include <stdlib.h>

int const lenbuff=16;

int xbuff=0x00;           // Índice: siguiente char en cbuff
char cbuff[lenbuff];     // Buffer

char palabra[6];        // Buffer para comparar las respuestas del modem

char rcvchar=0x00;      // último caracter recibido
int1 flagcommand=0;    // Flag para indicar comando disponible

void inicbuff(void);    // Borra buffer
int addcbuff(char c);   // añade caracter recibido al buffer
void echos(char c);     // Eco selectivo sobre RS232
void procesa_comando(void); // Procesa comando

int j, m;

int menu;
int vista=0;
int clave=0;
int newclave;
int nuevonum=0;
int const newteldat=12;
int newposicio=0;
int vertel=0;

// Desarrollo de Funciones //////////////////////////////////////

void inicbuff(void){    // Inicia a \0 cbuff -----
    int i;
    for(i=0;i<lenbuff;i++){ // Bucle que pone a 0 todos los
        cbuff[i]=0x00;     // caracteres en el buffer
    }
}

```



```

xbuff=0x00;          // Inicializo el indice de siguiente
                    // caracter

for(i=0;i<8;i++){
    palabra[i]=0x00;
}
}
//-----
int addcbuf(char c){    // Añade a cbuf -----
    switch(c){
        case 0x0D:      // Enter -> Habilita Flag para procesar
            flagcommand=1;    // Comando en Main
            break;
        case 0x08:      // Del-> Borra último caracter del Buffer
            //if(xbuff>0) cbuf[--xbuff]=0x00;
            inicbuf();
            break;
        //case 0x01B:    // Esc -> Borra el Buffer completamente
        // inicbuf();
        // break;
        default:
            cbuf[xbuff++]=c;    // Añade caracter recibido al Buffer
    }
}
}
////////// Comunicacion con la central //////////
void comunicacion(){

for (vista=0;vista<=15;vista++){
    lcd_gotoxy(vista+1,1);
    printf(lcd_putc,"%c", cbuf[vista]);
}

if ((rcvchar==0x20)&&(clave==0)&&(nuevonum==0)&&(newclave==0)){
    printf("\r\n");
    printf("Binevenido!\r\n");
    printf("UNO - reiniciar equipo\r\n");
    printf("DOS - cambiar numero telefonico\r\n");
    printf("TRE - cambiar clave\r\n");
    inicbuf();
}
if (rcvchar==0x0D){
    sprintf(palabra,"3");
    if ((strcmp(cbuf, palabra))==0){
        lcd_gotoxy(4,1);

```

```

        printf(lcd_putc,"NCARRIER");
        goto fallocom;
    }
}
if ((rcvchar==0x0D)&&(clave==0)&&(nuevonum==0)&&(newclave==0)){
    sprintf(palabra,"UNO");
    if ((strcmp(cbuff, palabra))==0){
        printf("\r\n");
        printf("entre clave para reiniciar:\r\n");
        clave=1;
        inicbuff();
    }
    sprintf(palabra,"DOS");
    if ((strcmp(cbuff, palabra))==0){
        nuevonum=1;
        printf(" OK\r\n");
        inicbuff();
    }
    sprintf(palabra,"TRE");
    if ((strcmp(cbuff, palabra))==0){
        newclave=1;
        printf("\r\n");
        printf("introduzca la nueva clave\r\n");
        inicbuff();
    }
    sprintf(palabra,"CUA");
    if ((strcmp(cbuff, palabra))==0){
        printf("\r\n");
        printf("telefonos en la EEPROM:\r\n");
        vernuevo:
        printf("A");
        for (m=0;m<12;++m){
            putc(read_EEPROM (m+newposicio));
        }

        printf("A\r\n");
        newposicio=newposicio+12;
        nuevonum=nuevonum+1;
        if (nuevonum<21){
            goto vernuevo;
        }
        newposicio=0;
        nuevonum=0;
    }
}

```

```

        vertel=0;
        inicbuff();
    }
    inicbuff();
}
if ((rcvchar==0x0D)&&(nuevonum>=1)&&(nuevonum<=21)){
if((cbuff[0]==0x41)&&((cbuff[10]==0x41)||((cbuff[11]==0x41)||((cbuff[12]==0x41))))){
    for (m=1;m<xbuff-1;++m){
        write_eeprom(m+newposicio-1,cbuff[m]);
    }
    for (m=xbuff-1;m<newteldat;++m){
        write_eeprom(m+newposicio-1,0x00);
    }
    newposicio=newposicio+12;
    nuevonum=nuevonum+1;
    printf("OK\r\n");
    inicbuff();
}else{
    printf("ERROR\r\n");
}
    inicbuff();
}
if (nuevonum>=22){
    nuevonum=0;
    newposicio=0;
    printf("\r\n");
    printf("numeros cambiados\r\n");
    inicbuff();
}
if ((rcvchar==0x0D)&&(clave>=1)){
    for (m=0;m<4;++m){
        palabra[m] = read_EEPROM (m+252);
    }
    if (((strcmp(cbuff, palabra))==0)&&(clave<=3)){
        printf("reiniciado...\r\n");
        clave=0;
        output_bit(PIN_B2,1);
        delay_ms(10000);
        output_bit(PIN_B2,0);
        printf("OK\r\n");
        inicbuff();
    }else{
        clave=clave+1;
    }
}

```

```

    if (clave<=3){
        printf("clave incorrecta!\r\n");
        printf("reingrese clave:\r\n");
        inicbuff();
    }
}
}
if ((rcvchar==0x0D)&&(newclave>=1)){
    if((cbuff[0]==0x41)&&(cbuff[5]==0x41)){
        for (m=1;m<xbuff-1;++m){
            write_eeprom(m+251,cbuff[m]);
        }
        newclave=0;
        printf("OK\r\n");
    }else{
        printf("ERROR\r\n");
    }
    inicbuff();
}
if ((rcvchar==0x1B)||((clave>=4))){ // Esc -> finaliza la comunicacion
falocom:
    clave=0;
    nuevonum=0;
    newposicio=0;
    newclave=0;
    printf("\r\n");
    printf("finalizando....\r\n");
    lcd_gotoxy(7,2);
    printf(lcd_putc,"FIN  ");
    printf("+");
    delay_ms(2000);
    printf("+");
    delay_ms(2000);
    printf("+\r\n");
    delay_ms(2000);
    printf("ATH0\r\n");
    delay_ms(5000);
    output_bit(PIN_B7,0);
    output_bit(PIN_C4,0);
    output_bit(PIN_C5,0);
    menu=1;
    inicbuff();
}

```

```
}
```

```
////////// identificacion de respuesta de config. del modem //////////
```

```
#SEPARATE
```

```
void modemrecive(){ //Comprueba el estado de la llamada
```

```
if (rcvchar==0x0D){
```

```
    sprintf(palabra,"0");
```

```
    if ((strcmp(cbuff, palabra))==0){
```

```
        inicbuff();
```

```
        lcd_gotoxy(7,2);
```

```
        printf(lcd_putc,"OK  ");
```

```
        menu=0;
```

```
        goto termina;
```

```
    }
```

```
    sprintf(palabra,"ATV0");
```

```
    if ((strcmp(cbuff, palabra))==0){
```

```
        inicbuff();
```

```
        lcd_gotoxy(7,2);
```

```
        printf(lcd_putc,"OK  ");
```

```
        menu=0;
```

```
        goto termina;
```

```
    }
```

```
    sprintf(palabra,"ATE0");
```

```
    if ((strcmp(cbuff, palabra))==0){
```

```
        inicbuff();
```

```
        lcd_gotoxy(7,2);
```

```
        printf(lcd_putc,"OK  ");
```

```
        menu=0;
```

```
        goto termina;
```

```
    }
```

```
    sprintf(palabra,"4");
```

```
    if ((strcmp(cbuff, palabra))==0){
```

```
        inicbuff();
```

```
        lcd_gotoxy(7,2);
```

```
        printf(lcd_putc,"ERROR ");
```

```
        delay_ms(2000);
```

```
        output_bit(PIN_B7,0);
```

```
        output_bit(PIN_C4,0);
```

```
        output_bit(PIN_C5,0);
```

```
        menu=0;
```

```
        goto termina;
```

```
    }
```

```
    sprintf(palabra,"3");
```

```

    if ((strcmp(cbuff, palabra))==0){
    inicbuff();
    lcd_gotoxy(7,2);
    printf(lcd_putc,"NCARRIER");
    delay_ms(2000);
    output_bit(PIN_B7,0);
    output_bit(PIN_C4,0);
    output_bit(PIN_C5,0);
    menu=0;
    goto termina;
}
sprintf(palabra,"10");
if ((strcmp(cbuff, palabra))==0){
    inicbuff();
    lcd_gotoxy(7,2);
    printf(lcd_putc,"C 2400 ");
    delay_ms(4000);
    lcd_init();
    printf(lcd_putc,"en linea..");
    delay_ms(1000);
    printf("en linea..\r\n");
    delay_ms(1000);
    menu=3;
    inicbuff();
    goto termina;
}
sprintf(palabra,"6");
if ((strcmp(cbuff, palabra))==0){
    inicbuff();
    lcd_gotoxy(7,2);
    printf(lcd_putc,"NO DIAL ");
    delay_ms(2000);
    menu=0;
    goto termina;
}
}
menu=5;
termina:
    #asm
    nop;
    #endasm
}

```

Anexo C. Código fuente del programa de Visual Basic.

Pantalla inicial del programa:

```
Option Explicit
Private Sub Form_KeyPress(KeyAscii As Integer)
    Unload Me
End Sub

Private Sub Form_Load()
    lblVersion.Caption = "Versión " & App.Major & "." & App.Minor & "." &
App.Revision
    lblProductName.Caption = "Módulo de Contro de Reinicio" & Chr(10) & "de
Equipos SPARTON mediante" & Chr(10) & " la línea telefónica"
End Sub

Private Sub Frame1_Click()
    Unload Me
End Sub
```

Pantalla del Menú principal del programa:

```
' VBTerm - Este es un programa de demostración del
' control ActiveX de comunicaciones MSComm.
'
' Copyright (c) 1994, Crescent Software, Inc.
' por Don Malin y Carl Franklin.
'
' Actualizado por Mike Maddox
'-----
Public pass As String
Public recibe As Variant
Public LoginSucceeded As Boolean

'Public rsTelefonos As ADODB.Recordset
Option Explicit

Public call_number As String
Dim Ret As Integer    ' Entero auxiliar.
Dim Temp As String    ' Cadena auxiliar.
Dim hLogFile As Integer ' Controlador de archivo de registro abierto.
Dim StartTime As Date ' Almacena la hora de inicio del cronómetro del puerto
```

```

Private Type num_telefonico
    central As String * 20
    numero1 As String * 8
    numero2 As String * 8
End Type

Private Sub Cam_Clave_Click()
    Dim new_pass
    MSComm1.Output = "SEC"
    new_pass = InputBox("Introduzca Nueva Clave: ")
    If new_pass = pass Then
        MsgBox ("Clave Correcta")
    Else
        MsgBox ("Clave Incorrecta")
    End If
End Sub

Private Sub Command1_Click()
    Dim respuesta As String
    respuesta = MsgBox("Está Seguro?", vbYesNo + vbCritical + vbDefaultButton2)

    If respuesta = vbYes Then
        frmTerminal.MSComm1.Output = "UNO" & Chr(13) 'Hace el reinicio de la
        maquina
    End If
End Sub

Private Sub Command2_Click()
    Cam_tel.Show vbModal
End Sub

Private Sub Colgar_Click()
    'RUTINA PARA COLGAR
    frmTerminal.MSComm1.Output = Chr(27)
End Sub

Private Sub Command3_Click()

End Sub

Private Sub config_tels_Click()

```



```

Config.Show vbModal
End Sub

Private Sub Form_Load()

'While LoginSucceeded = False
  frmLogin.Show vbModal
'Wend

frmSplash.Show vbModal

  Dim CommPort As String, Handshaking As String, Settings As String

  On Error Resume Next

  ' Establece el color predeterminado del terminal
  txtTerm.SelLength = Len(txtTerm)
  txtTerm.SelText = ""
  txtTerm.ForeColor = vbBlue

  ' Establece el título
  App.Title = "Terminal de Visual Basic"
  ' Establece la luz indicadora de estado
  imgNotConnected.ZOrder

  ' Centra el formulario
  frmTerminal.Move (Screen.Width - Width) / 2, (Screen.Height - Height) / 2

  ' Carga la configuración del registro

  Settings = GetSetting(App.Title, "Properties", "Settings", "")
  frmTerminal.MSComm1.Settings\
  If Settings <> "" Then
    MSComm1.Settings = Settings
    If Err Then
      MsgBox Error$, 48
      Exit Sub
    End If
  End If

  CommPort = GetSetting(App.Title, "Properties", "CommPort", "")
  frmTerminal.MSComm1.CommPort
  If CommPort <> "" Then MSComm1.CommPort = CommPort

```

```

    Handshaking = GetSetting(App.Title, "Properties", "Handshaking", "")
    frmTerminal.MSComm1.Handshaking
    If Handshaking <> "" Then
        MSComm1.Handshaking = Handshaking
        If Err Then
            MsgBox Error$, 48
            Exit Sub
        End If
    End If

    Echo = GetSetting(App.Title, "Properties", "Echo", "") ' Echo
    On Error GoTo 0

```

```
End Sub
```

```
Private Sub Form_Resize()
```

```

    ' Cambia el tamaño del control Term (ventana)
    "tbrToolBar.Height , frmTerminal.ScaleWidth, frmTerminal.ScaleHeight -
    sbrStatus.Height - tbrToolBar.Height
    "txtTerm.Move 0,
    ' Sitúa la luz indicadora de estado
    Frame1.Left = ScaleWidth - Frame1.Width * 1.5
End Sub

```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    Dim Counter As Long
```

```
    If MSComm1.PortOpen Then
```

```
        ' Espera 10 segundos para transmitir los datos.
```

```
        Counter = Timer + 10
```

```
        Do While MSComm1.OutBufferCount
```

```
            Ret = DoEvents()
```

```
            If Timer > Counter Then
```

```
                Select Case MsgBox("Imposible enviar los datos", 34)
```

```
                    ' Cancelar.
```

```
                    Case 3
```

```
                        Cancel = True
```

```
                        Exit Sub
```

```
                    ' Reintentar.
```

```
                    Case 4
```

```
                        Counter = Timer + 10
```

```

        ' Ignorar.
        Case 5
        Exit Do
    End Select
End If
Loop

MSComm1.PortOpen = 0
End If

' Si el archivo de registro está abierto, vuelca y lo cierra.
If hLogFile Then mnuCloseLog_Click
End
End Sub

Private Sub imgConnected_Click()

    ' Llama a la rutina mnuOpen_Click para alternar entre conectar y desconectar.
    Call mnuOpen_Click
End Sub

Private Sub imgNotConnected_Click()
    ' Llama a la rutina mnuOpen_Click para alternar entre conectar y desconectar.
    Call mnuOpen_Click
End Sub

Private Sub mnuAcerca_Click()

frmAbout.Show vbModal
End Sub

Private Sub mnuCloseLog_Click()

    ' Cierra el archivo de registro.
    Close hLogFile
    hLogFile = 0
    mnuOpenLog.Enabled = True
    tbrToolBar.Buttons("OpenLogFile").Enabled = True
    mnuCloseLog.Enabled = False
    tbrToolBar.Buttons("CloseLogFile").Enabled = False
    frmTerminal.Caption = "Terminal de Visual Basic"
End Sub

```

```

Public Sub mnuDial_Click()

    On Local Error Resume Next
    Static Num As String

    ' Abre el puerto si no está abierto ya.
    If Not MSComm1.PortOpen Then
        mnuOpen_Click
        If Err Then Exit Sub
    End If

    Num = frmTerminal.call_number
    ' Habilita el botón de colgar y el elemento de menú correspondiente.
    mnuHangUp.Enabled = True
    tbrToolBar.Buttons("HangUpPhone").Enabled = True

    ' Marca el número.
    MSComm1.Output = "ATDT" & Num & vbCrLf

    ' Inicia el cronómetro del puerto.

    StartTiming
End Sub

Private Sub mnuFileExit_Click()
    ' Utiliza Form_Unload, ya que tiene código para comprobar si hay datos
    ' no enviados y si hay un archivo de registro abierto.
    Form_Unload Ret
End Sub

' Alterna la propiedad DTREnable para colgar la línea.
Private Sub mnuHangup_Click()
    On Error Resume Next

    MSComm1.Output = "ATH"    ' Envía la cadena de colgar
    Ret = MSComm1.DTREnable   ' Guarda el valor actual.
    MSComm1.DTREnable = True  ' Activa DTR.
    MSComm1.DTREnable = False ' Desactiva DTR.
    MSComm1.DTREnable = Ret   ' Restablece el valor anterior.
    mnuHangUp.Enabled = False
    tbrToolBar.Buttons("HangUpPhone").Enabled = False

    ' Si el puerto continúa abierto, lo cierra

```

```

If MSComm1.PortOpen Then MSComm1.PortOpen = False

' Notifica el error al usuario
If Err Then MsgBox Error$, 48

mnuSendText.Enabled = False
tbrToolBar.Buttons("TransmitTextFile").Enabled = False
mnuHangUp.Enabled = False
tbrToolBar.Buttons("HangUpPhone").Enabled = False
mnuDial.Enabled = True
tbrToolBar.Buttons("DialPhoneNumber").Enabled = True
sbrStatus.Panels("Settings").Text = "Valores:"

' Apaga la luz indicadora y quita la marca al menú
mnuOpen.Checked = False
imgNotConnected.ZOrder

' Detiene el cronómetro del puerto
StopTiming
sbrStatus.Panels("Status").Text = "Estado:"
On Error GoTo 0
End Sub

' Este procedimiento establece la propiedad InputLen, que determina
' el número de bytes de datos leídos cada vez que se usa Input para
' recuperar datos del búfer de entrada.
' Al establecer 0 en InputLen se especifica que debe leerse todo el
' contenido del búfer.
Private Sub mnulInputLen_Click()
' On Error Resume Next
"      Temp = InputBox$("Escriba nuevo InputLen:", "InputLen",
Str$(MSComm1.InputLen))
" If Len(Temp) Then
"     MSComm1.InputLen = Val(Temp)
"     If Err Then MsgBox Error$, 48
" End If
"End Sub

Private Sub mnuProperties_Click()
' Muestra el formulario de propiedades de CommPort
frmProps.Show vbModal

End Sub

```

```

' Alterna el estado del puerto (abierto o cerrado).
Private Sub mnuOpen_Click()
    On Error Resume Next
    Dim OpenFlag

    MSComm1.PortOpen = Not MSComm1.PortOpen
    If Err Then MsgBox Error$, 48

    OpenFlag = MSComm1.PortOpen

    mnuOpen.Checked = OpenFlag
    mnuSendText.Enabled = OpenFlag
    tbrToolBar.Buttons("TransmitTextFile").Enabled = OpenFlag

    If MSComm1.PortOpen Then
        ' Habilita el botón de marcar y el elemento de menú asociado
        mnuDial.Enabled = True
        tbrToolBar.Buttons("DialPhoneNumber").Enabled = True

        ' Habilita el botón de colgar y el elemento de menú asociado
        mnuHangUp.Enabled = True
        tbrToolBar.Buttons("HangUpPhone").Enabled = True

        imgConnected.ZOrder
        sbrStatus.Panels("Settings").Text = "Valores:" & MSComm1.Settings
        StartTiming
    Else
        ' Habilita el botón de marcar y el elemento de menú asociado
        mnuDial.Enabled = True
        tbrToolBar.Buttons("DialPhoneNumber").Enabled = True
        ' deshabilita el botón de colgar y el elemento de menú asociado
        mnuHangUp.Enabled = False
        tbrToolBar.Buttons("HangUpPhone").Enabled = False
        imgNotConnected.ZOrder
        sbrStatus.Panels("Settings").Text = "Valores:"
        StopTiming
    End If

End Sub

Private Sub mnuOpenLog_Click()
    Dim replace

```

```

On Error Resume Next
openlog.Flags = cdlOFNHideReadOnly Or cdlOFNExplorer
openlog.CancelError = True

' Obtiene del usuario el nombre de archivo largo.
openlog.DialogTitle = "Abrir archivo de registro de comunicaciones"
openlog.Filter = "Archivos de registro (*.LOG)|*.log|Todos los archivos (*.*)|*.*"
Do
    openlog.FileName = ""
    openlog.ShowOpen
    If Err = cdlCancel Then Exit Sub
    Temp = openlog.FileName

    ' Si el archivo ya existe, pregunta al usuario si desea
    ' sobrescribirlo o agregarle datos.
    Ret = Len(Dir$(Temp))
    If Err Then
        MsgBox Error$, 48
        Exit Sub
    End If
    If Ret Then
        replace = MsgBox("Reemplazar el archivo - " + Temp + "?", 35)
    Else
        replace = 0
    End If
Loop While replace = 2
' El usuario ha hecho clic en el botón Sí, así que elimina el archivo.
If replace = 6 Then
    Kill Temp
    If Err Then
        MsgBox Error$, 48
        Exit Sub
    End If
End If

frmTerminal.Caption = "Terminal de Visual Basic - " + openlog.FileTitle
mnuOpenLog.Enabled = False
tbrToolBar.Buttons("OpenLogFile").Enabled = False
mnuCloseLog.Enabled = True
tbrToolBar.Buttons("CloseLogFile").Enabled = True
End Sub

' Este procedimiento establece la propiedad ParityReplace, que

```

```

' contiene el carácter que reemplazará a los caracteres
' incorrectos recibidos a causa de un error de paridad.

"Private Sub mnuParRep_Click()
"   On Error Resume Next

"       Temp = InputBox$("Escriba el carácter que desea reemplazar",
"ParityReplace", frmTerminal.MSComm1.ParityReplace)
"   frmTerminal.MSComm1.ParityReplace = Left$(Temp, 1)
"   If Err Then MsgBox Error$, 48
"End Sub

' Este procedimiento establece la propiedad RThreshlold, que
' determina el número de bytes que pueden llegar al búfer de
' recepción antes de disparar el evento OnComm y de que se
' establezca comEvReceive en la propiedad CommEvent.
"Private Sub mnuRThreshold_Click()
"   On Error Resume Next

"       Temp = InputBox$("Introduzca el nuevo RThreshold:", "RThreshold",
Str$(MSComm1.RThreshold))
"   If Len(Temp) Then
"       MSComm1.RThreshold = Val(Temp)
"       If Err Then MsgBox Error$, 48
"   End If

"End Sub

' El evento OnComm se usa para interceptar eventos y errores de comunicaciones.
Private Static Sub MSComm1_OnComm()

    Dim EVMsg$
    Dim ERMsg$
    Dim k, i As Integer

    ' Bifurca según la propiedad CommEvent.
    Select Case MSComm1.CommEvent
        ' Mensajes de evento.
        Case comEvReceive
            Dim Buffer As Variant

            Buffer = MSComm1.Input
            ShowData txtTerm, (StrConv(Buffer, vbUnicode))
    
```



```

Debug.Print "Recibir - "; (StrConv(Buffer, vbUnicode))

recibe = StrConv(Buffer, vbUnicode)

For k = 1 To 20

    If Mid(recibe, k, 3) = "CON" Then 'Cuando se Conecta

        MSComm1.Output = Chr(32) & Chr(32) & Chr(13)

        Command1.Enabled = True
        Command2.Enabled = True
    End If

Next k

Debug.Print "Recibir - "; (StrConv(recibe, vbUnicode))

Case comEvSend
Case comEvCTS
    EVMsg$ = "Detectado cambio en CTS"
Case comEvDSR
    EVMsg$ = "Detectado cambio en DSR"
Case comEvCD
    EVMsg$ = "Detectado cambio en CD"
Case comEvRing
    EVMsg$ = "El teléfono está sonando"
Case comEvEOF
    EVMsg$ = "Detectado el final del archivo"
' Mensajes de error.
Case comBreak
    ERMsg$ = "Parada recibida"
Case comCDTO
    ERMsg$ = "Sobrepasado el tiempo de espera de detección de portadora"
Case comCTSTO
    ERMsg$ = "Sobrepasado el tiempo de espera de CTS"
Case comDCB
    ERMsg$ = "Error recibiendo DCB"
Case comDSRTO
    ERMsg$ = "Sobrepasado el tiempo de espera de DSR"
Case comFrame
    ERMsg$ = "Error de marco"

```

```

Case comOverrun
    ERMsg$ = "Error de sobrecarga"
Case comRxOver
    ERMsg$ = "Desbordamiento en el búfer de recepción"
Case comRxParity
    ERMsg$ = "Error de paridad"
Case comTxFull
    ERMsg$ = "Búfer de transmisión lleno"
Case Else
    ERMsg$ = "Error o evento desconocido"
End Select

If Len(EVMsg$) Then
    ' Muestra los mensajes de evento en la barra de estado.
    sbrStatus.Panels("Status").Text = "Estado:" & EVMsg$

    ' Activa el cronómetro para que el mensaje de la barra
    ' de estado se borre después de dos segundos.
    Timer2.Enabled = True

Elseif Len(ERMsg$) Then
    ' Muestra los mensajes de evento en la barra de estado.
    sbrStatus.Panels("Status").Text = "Estado:" & ERMsg$

    ' Muestra los mensajes de error en un cuadro de alerta.
    Beep
    Ret = MsgBox(ERMsg$, 1, "Haga clic en Cancelar para salir, clic en Aceptar
para ignorar.")

    ' Si el usuario hace clic en Cancelar (2)...
    If Ret = 2 Then
        MSComm1.PortOpen = False ' Cierra el puerto y sale.
    End If

    ' Activa el cronómetro para que el mensaje de la barra
    ' de estado se borre después de dos segundos.
    Timer2.Enabled = True
End If
End Sub

Private Sub mnuSendText_Click()
    Dim hSend, BSize, LF&

```

On Error Resume Next

```
mnuSendText.Enabled = False  
tbrToolBar.Buttons("TransmitTextFile").Enabled = False
```

```
' Obtiene del usuario el nombre del archivo de texto.  
openlog.DialogTitle = "Enviar archivo de texto"  
openlog.Filter = "Archivos de texto (*.TXT)|*.txt|Todos los archivos (*.*)|*.*"
```

Do

```
    openlog.CancelError = True  
    openlog.FileName = ""  
    openlog.ShowOpen  
    If Err = cdICancel Then  
        mnuSendText.Enabled = True  
        tbrToolBar.Buttons("TransmitTextFile").Enabled = True  
        Exit Sub  
    End If  
    Temp = openlog.FileName
```

```
' Si el archivo no existe, vuelve.  
Ret = Len(Dir$(Temp))  
If Err Then  
    MsgBox Error$, 48  
    mnuSendText.Enabled = True  
    tbrToolBar.Buttons("TransmitTextFile").Enabled = True  
    Exit Sub  
End If  
If Ret Then  
    Exit Do  
Else  
    MsgBox Temp + " no encontrado.", 48  
End If
```

Loop

```
' Abre el archivo de registro.  
hSend = FreeFile  
Open Temp For Binary Access Read As hSend
```

```
If Err Then  
    MsgBox Error$, 48  
Else  
    ' Muestra el cuadro de diálogo Cancelar.
```

```

CancelSend = False
frmCancelSend.Label1.Caption = "Transmitiendo archivo de texto - " + Temp
frmCancelSend.Show

' Lee el archivo en bloques del tamaño del búfer de transmisión.
BSize = MSComm1.OutBufferSize
LF& = LOF(hSend)
Do Until EOF(hSend) Or CancelSend
    ' No lee demasiado al final.
    If LF& - Loc(hSend) <= BSize Then
        BSize = LF& - Loc(hSend) + 1
    End If

    ' Lee un bloque de datos.
    Temp = Space$(BSize)
    Get hSend, , Temp

    ' Transmite el bloque.
    MSComm1.Output = Temp
    If Err Then
        MsgBox Error$, 48
        Exit Do
    End If

    ' Espera a que se envíen todos los datos.
    Do
        Ret = DoEvents()
    Loop Until MSComm1.OutBufferCount = 0 Or CancelSend
Loop
End If
Close hSend
mnuSendText.Enabled = True
tbrToolBar.Buttons("TransmitTextFile").Enabled = True
CancelSend = True
frmCancelSend.Hide
End Sub

' Este procedimiento establece la propiedad SThreshold, que
' determina el número máximo de caracteres que deben estar
' esperando en el búfer de salida para que se establezca
' comEvSend en la propiedad CommEvent y se dispare el evento
' OnComm.

```

```

"Private Sub mnuSThreshold_Click()
"   On Error Resume Next

"   Temp = InputBox$("Escriba el nuevo valor de SThreshold", "SThreshold",
Str$(MSComm1.SThreshold))
"   If Len(Temp) Then
"       MSComm1.SThreshold = Val(Temp)
"       If Err Then MsgBox Error$, 48
"   End If
"End Sub

```

```

Private Sub ESPACIOS_Click()
Dim ESPACIOS As String

```

```

ESPACIOS = Chr(32) & Chr(32) & Chr(32) & Chr(32) & Chr(32) & Chr(32)

```

```

MSComm1.Output = (ESPACIOS)

```

```

End Sub

```

```

' Este procedimiento agrega datos a la propiedad Text del
' control Term. También filtra los caracteres de control,
' como RETROCESO, retorno de carro y avances de línea, y
' escribe datos en un archivo de registro.
' Los caracteres RETROCESO eliminan el carácter situado a
' su izquierda, ya sea en la propiedad Text o en la cadena
' pasada. Se agregan caracteres de avance de línea a todos
' los retornos de carro. El tamaño de la propiedad Text del
' control Term también se controla para que nunca exceda de
' MAXTERMSIZE caracteres.

```

```

Private Static Sub ShowData(Term As Control, Data As String)
    On Error GoTo Handler
    Const MAXTERMSIZE = 16000
    Dim TermSize As Long, i
    ' Se asegura que el texto existente no se haga demasiado largo.
    TermSize = Len(Term.Text)
    If TermSize > MAXTERMSIZE Then
        Term.Text = Mid$(Term.Text, 4097)
        TermSize = Len(Term.Text)
    End If

```

```

' Apunta al final de los datos de Term.

```

```

Term.SelStart = TermSize

' Filtra y procesa los caracteres RETROCESO.
Do
  i = InStr(Data, Chr$(8))
  If i Then
    If i = 1 Then
      Term.SelStart = TermSize - 1
      Term.SelLength = 1
      Data = Mid$(Data, i + 1)
    Else
      Data = Left$(Data, i - 2) & Mid$(Data, i + 1)
    End If
  End If
Loop While i

' Elimina los avances de línea.

Do
  i = InStr(Data, Chr$(10))
  If i Then
    Data = Left$(Data, i - 1) & Mid$(Data, i + 1)
  End If
Loop While i
' Se asegura de que todos los retornos de carro tengan un
' avance de línea.
i = 1
Do
  i = InStr(i, Data, Chr$(13))
  If i Then
    Data = Left$(Data, i) & Chr$(10) & Mid$(Data, i + 1)
    i = i + 1
  End If
Loop While i

' Agrega los datos filtrados a la propiedad SelText.
Term.SelText = Data
' Registra los datos en un archivo si así se solicita.
If hLogFile Then
  i = 2
  Do
    Err = 0
    Put hLogFile, , Data
  
```

```

        If Err Then
            i = MsgBox(Error$, 21)
            If i = 2 Then
                mnuCloseLog_Click
            End If
        End If
    Loop While i <> 2
End If
Term.SelStart = Len(Term.Text)
Exit Sub
Handler:
    MsgBox Error$
    Resume Next
End Sub
Private Sub Timer2_Timer()
sbrStatus.Panels("Status").Text = "Estado:"
Timer2.Enabled = False
End Sub

' Las pulsaciones interceptadas aquí se envían
' al control MSComm, donde se devuelven a través
' del evento OnComm (comEvReceive), y se muestran
' con el procedimiento ShowData.

Private Sub txtTerm_KeyPress(KeyAscii As Integer)
    ' Si el puerto está abierto...
    If MSComm1.PortOpen Then
        ' Envía la pulsación al puerto.
        MSComm1.Output = Chr$(KeyAscii)
        ' Si el eco no está activado, no hay
        ' necesidad de que el control de texto
        ' muestre la tecla. Normalmente, el módem
        ' devolverá el carácter.
        If Not Echo Then
            ' Sitúa la posición al final del terminal
            txtTerm.SelStart = Len(txtTerm)
            KeyAscii = 0
        End If
    End If
End Sub

Private Sub tbrToolBar_ButtonClick(ByVal Button As MSComctlLib.Button)
Select Case Button.Key

```

```

Case "OpenLogFile"
    Call mnuOpenLog_Click
Case "CloseLogFile"
    Call mnuCloseLog_Click
Case "DialPhoneNumber"
    Call mnuDial_Click
Case "HangUpPhone"
    Call mnuHangup_Click
Case "Propiedades"
    Call mnuProperties_Click
Case "TransmitTextFile"
    Call mnuSendText_Click
End Select
End Sub

```

```

Private Sub Timer1_Timer()

```

```

    ' Muestra la hora de conexión
    sbrStatus.Panels("ConnectTime").Text = Format(Now - StartTime, "hh:nn:ss") &
    " "

```

```

End Sub

```

```

' Llama a esta función para iniciar el cronómetro ConnectTime

```

```

Private Sub StartTiming()

```

```

    StartTime = Now
    Timer1.Enabled = True

```

```

End Sub

```

```

' Llama a esta función para detener el cronometraje

```

```

Private Sub StopTiming()

```

```

    Timer1.Enabled = False
    sbrStatus.Panels("ConnectTime").Text = ""

```

```

End Sub

```

Pantalla del menú de configuración del puerto para el módem

```

Private iFlow As Integer, iTempEcho As Boolean

```

```

Sub LoadPropertySettings()

```

```

    Dim i As Integer, Settings As String, Offset As Integer

```

```

    ' Carga las configuraciones de puerto

```



```

For i = 1 To 16
    cboPort.AddItem "Com" & Trim$(Str$(i))
Next i

' Carga las configuraciones de velocidad
cboSpeed.AddItem "110"
cboSpeed.AddItem "300"
cboSpeed.AddItem "600"
cboSpeed.AddItem "1200"
cboSpeed.AddItem "2400"
cboSpeed.AddItem "4800"
cboSpeed.AddItem "9600"
cboSpeed.AddItem "14400"
cboSpeed.AddItem "19200"
cboSpeed.AddItem "28800"
cboSpeed.AddItem "38400"
cboSpeed.AddItem "56000"
cboSpeed.AddItem "57600"
cboSpeed.AddItem "115200"
cboSpeed.AddItem "128000"
cboSpeed.AddItem "256000"

' Carga las configuraciones de bits de datos
cboDataBits.AddItem "4"
cboDataBits.AddItem "5"
cboDataBits.AddItem "6"
cboDataBits.AddItem "7"
cboDataBits.AddItem "8"
' Carga las configuraciones de paridad
cboParity.AddItem "Even"
cboParity.AddItem "Odd"
cboParity.AddItem "None"
cboParity.AddItem "Mark"
cboParity.AddItem "Space"
' Carga las configuraciones de bits de parada
cboStopBits.AddItem "1"
cboStopBits.AddItem "1.5"
cboStopBits.AddItem "2"
' Establece la configuración predeterminada
Settings = frmTerminal.MSComm1.Settings

' En todos los casos, el componente más a la derecha de Settings
' será un solo carácter, excepto cuando haya 1,5 bits de parada.

```

```

If InStr(Settings, ".") > 0 Then
    Offset = 2
Else
    Offset = 0
End If
cboSpeed.Text = Left$(Settings, Len(Settings) - 6 - Offset)
Select Case Mid$(Settings, Len(Settings) - 4 - Offset, 1)
Case "e"
    cboParity.ListIndex = 0
Case "m"
    cboParity.ListIndex = 1
Case "n"
    cboParity.ListIndex = 2
Case "o"
    cboParity.ListIndex = 3
Case "s"
    cboParity.ListIndex = 4
End Select
cboDataBits.Text = Mid$(Settings, Len(Settings) - 2 - Offset, 1)
cboStopBits.Text = Right$(Settings, 1 + Offset)
cboPort.ListIndex = frmTerminal.MSComm1.CommPort - 1
optFlow(frmTerminal.MSComm1.Handshaking).Value = True
If Echo Then
    optEcho(1).Value = True
Else
    optEcho(0).Value = True
End If

End Sub

Private Sub cmdCancel_Click()

Unload Me
End Sub

Private Sub cmdOK_Click()

Dim OldPort As Integer, ReOpen As Boolean
On Error Resume Next
Echo = iTempEcho
OldPort = frmTerminal.MSComm1.CommPort
NewPort = cboPort.ListIndex + 1
If NewPort <> OldPort Then ' Si cambia el número de puerto, cierra el antiguo.

```

```

If frmTerminal.MSComm1.PortOpen Then
    frmTerminal.MSComm1.PortOpen = False
    ReOpen = True
End If
frmTerminal.MSComm1.CommPort = NewPort      ' Establece el nuevo número
de puerto.

If Err = 0 Then
    If ReOpen Then
        frmTerminal.MSComm1.PortOpen = True
        frmTerminal.mnuOpen.Checked = frmTerminal.MSComm1.PortOpen
        frmTerminal.mnuSendText.Enabled = frmTerminal.MSComm1.PortOpen
        frmTerminal.tbrToolBar.Buttons("TransmitTextFile").Enabled =
frmTerminal.MSComm1.PortOpen
    End If
End If
If Err Then
    MsgBox Error$, 48
    frmTerminal.MSComm1.CommPort = OldPort
    Exit Sub
End If
End If
frmTerminal.MSComm1.Settings = Trim$(cboSpeed.Text) & "," &
Left$(cboParity.Text, 1) _
& "," & Trim$(cboDataBits.Text) & "," & Trim$(cboStopBits.Text)
If Err Then
    MsgBox Error$, 48
    Exit Sub
End If
frmTerminal.MSComm1.Handshaking = iFlow
If Err Then
    MsgBox Error$, 48
    Exit Sub
End If
SaveSetting App.Title, "Properties", "Settings", frmTerminal.MSComm1.Settings
SaveSetting App.Title, "Properties", "CommPort",
frmTerminal.MSComm1.CommPort
SaveSetting App.Title, "Properties", "Handshaking",
frmTerminal.MSComm1.Handshaking
SaveSetting App.Title, "Properties", "Echo", Echo
Unload Me
End Sub

```

```

Private Sub Form_Load()

' Establece el tamaño del formulario
Me.Left = (Screen.Width - Me.Width) / 2
Me.Top = (Screen.Height - Me.Height) / 2
' Ajusta el tamaño del marco al control tabstrip
fraSettings.Move tabSettings.ClientLeft, tabSettings.ClientTop
' Se asegura de que el marco sea el control situado encima
fraSettings.ZOrder
' Carga la configuración actual de las propiedades
LoadPropertySettings
End Sub

Private Sub optEcho_Click(Index As Integer)
If Index = 1 Then
    iTempEcho = True
Else
    iTempEcho = False
End If
End Sub
Private Sub optFlow_Click(Index As Integer)
iFlow = Index
End Sub

```

Pantalla del menú del directorio de centrales

```

Dim Ret As Integer    ' Entero auxiliar.
Dim Temp As String    ' Cadena auxiliar.
Dim hLogFile As Integer ' Controlador de archivo de registro abierto.
Dim StartTime As Date ' Almacena la hora de inicio del cronómetro del puerto
Private Type num_telefonico
    central As String * 20
    numero1 As String * 7
End Type

Private Sub Cargar_Click()
    Dim replace
    Dim strRow As String
    Dim strField As String
    Dim telefono As num_telefonico
    On Error Resume Next
    openlog.Flags = cdIOFNHideReadOnly Or cdIOFNExplorer
    openlog.CancelError = True

```

```

'Obtiene del usuario el nombre de archivo largo.
openlog.DialogTitle = "Abrir archivo de directorio"
openlog.Filter = "Archivos de directorio (*.TEL)|*.TEL|Todos los archivos (*.*)|*.*"

Do
    openlog.FileName = ""
    openlog.ShowOpen
    If Err = cdICancel Then Exit Sub
    Temp = openlog.FileName
    Loop While replace = 2
replace = 0
"Abrir el archivo para mostrar los números para marcar
Open Temp For Random As #1 Len = Len(telefono)
For i = 0 To 33
    Get #1, i + 1, telefono
    central(i).Text = UCase(telefono.central)
    numero_tel1(i).Text = UCase(telefono.numero1)
Next
Close All
End Sub

Private Sub Command1_Click()
Close All
Unload Me
End Sub

Private Sub Form_Load()
Dim i As Integer
llamar.Enabled = False
For i = 0 To 33
    central(i).Text = ""
    numero_tel1(i).Text = ""
Next
End Sub

Private Sub guardar_datos_Click()

Dim replace
Dim strRow As String
Dim strField As String
Dim telefono As num_telefonico
On Error Resume Next
openlog.Flags = cdIOfNHideReadOnly Or cdIOfNExplorer

```

```

    openlog.CancelError = True
' Obtiene del usuario el nombre de archivo largo.
openlog.DialogTitle = "Guardar archivo de directorio"
openlog.Filter = "Guardar el directorio (*.TEL)|*.TEL|Todos los archivos (*.*)|*.*"

Do
    openlog.FileName = ""
    openlog.ShowSave
    If Err = cdlCancel Then Exit Sub
    Temp = openlog.FileName
    Loop While replace = 2

' Abrir el archivo para guardar los números para marcar

Open Temp For Random As #1 Len = Len(telefono)
For i = 0 To 33
    If (central(i).Text <> "") Or (numero_tel1(i).Text <> "") Then
        telefono.central = UCase(central(i).Text)
        telefono.numero1 = (numero_tel1(i).Text)
        Put #1, i + 1, telefono
    End If
Next i
Close All
End Sub

Private Sub llamar_Click()

Dim i As Integer
For i = 0 To 33
    If (Option1(i).Value = True) Then
        frmTerminal.call_number = numero_tel1(i).Text
    End If
Next i
Close All
Call frmTerminal.mnuDial_Click
Hide
End Sub

Private Sub Option1_Click(Index As Integer)
llamar.Enabled = True
End Sub

```

Pantalla del menú de cambio de números telefónicos

```

Private Sub Command1_Click()
'Enviar Nuevos Números Al Módulo
frmTerminal.MSComm1.Output = "CAMB_NUM"
End Sub

Private Sub Command2_Click()
Unload Me
End Sub

Private Sub consult_Click()
Dim i, k As Integer
i = 0
For k = 1 To 200
If Mid(recibe, k, 1) = "A" And Mid(recibe, k + 13, 1) = "A" Then 'Número Celular
    Cam_tel.Text1(i).Text = Mid(recibe, k + 2, 11) 'Toma el número celular y lo muestra
    i = i + 1
        End If
        If Mid(recibe, k, 1) = "A" And Mid(recibe, k + 12, 1) = "A" Then
'Número Celular
            Cam_tel.Text1(i).Text = Mid(recibe, k + 2, 11) 'Toma el número celular y lo muestra
            i = i + 1
        End If
        If Mid(recibe, k, 1) = "A" And Mid(recibe, k + 14, 1) = "A" Then
'Número Celular
            Cam_tel.Text1(i).Text = Mid(recibe, k + 2, 11) 'Toma el número celular y lo muestra
            i = i + 1
        End If
        If Mid(recibe, k, 1) = "A" And Mid(recibe, k + 11, 1) = "A" Then
'Número fijo
            Cam_tel.Text1(i).Text = Mid(recibe, k + 2, 9) 'Toma el número fijo y lo muestra
            i = i + 1
        End If
        If Mid(recibe, k, 1) = "A" And Mid(recibe, k + 10, 1) = "A" Then
'Número para módem
            Cam_tel.Text1(i).Text = Mid(recibe, k + 2, 10) 'Toma el número para módem y lo muestra
            i = i + 1
        End If

```

```
Next k
End Sub
```

```
Private Sub Form_Load()
Dim i As Integer
For i = 0 To 19
    With Combo1(i)
        .AddItem "Directo"
        .AddItem "Módem"
    End With
Next i
frmTerminal.MSComm1.Output = "CUA" & Chr(13)
End Sub
```

```
Private Sub Send_Numbers_Click()
Dim i As Integer
For i = 0 To 19
frmTerminal.MSComm1.Output = "A" & Text1(i).Text & "A" & Chr(13)
Next i
End Sub
```

Pantalla del Acerca de

Option Explicit

```
' Opciones de seguridad de clave del Registro...
Const READ_CONTROL = &H20000
Const KEY_QUERY_VALUE = &H1
Const KEY_SET_VALUE = &H2
Const KEY_CREATE_SUB_KEY = &H4
Const KEY_ENUMERATE_SUB_KEYS = &H8
Const KEY_NOTIFY = &H10
Const KEY_CREATE_LINK = &H20
Const KEY_ALL_ACCESS = KEY_QUERY_VALUE + KEY_SET_VALUE + _
    KEY_CREATE_SUB_KEY + KEY_ENUMERATE_SUB_KEYS + _
    KEY_NOTIFY + KEY_CREATE_LINK + READ_CONTROL
' Tipos ROOT de clave del Registro...
Const HKEY_LOCAL_MACHINE = &H80000002
Const ERROR_SUCCESS = 0
Const REG_SZ = 1 ' Cadena Unicode terminada en valor nulo
Const REG_DWORD = 4 ' Número de 32 bits
Const gREGKEYSYSINFOLOC = "SOFTWARE\Microsoft\Shared Tools Location"
Const gREGVALSYSINFOLOC = "MSINFO"
```



```

Const gREGKEYSYSINFO = "SOFTWARE\Microsoft\Shared Tools\MSINFO"
Const gREGVALSYSINFO = "PATH"
Private Declare Function RegOpenKeyEx Lib "advapi32" Alias "RegOpenKeyExA"
(ByVal hKey As Long, ByVal lpSubKey As String, ByVal ulOptions As Long, ByVal
samDesired As Long, ByRef phkResult As Long) As Long

Private Declare Function RegQueryValueEx Lib "advapi32" Alias
"RegQueryValueExA" (ByVal hKey As Long, ByVal lpValueName As String, ByVal
lpReserved As Long, ByRef lpType As Long, ByVal lpData As String, ByRef
lpcbData As Long) As Long
Private Declare Function RegCloseKey Lib "advapi32" (ByVal hKey As Long) As
Long

Private Sub cmdSysInfo_Click()
    Call StartSysInfo
End Sub

Private Sub cmdOK_Click()
    Unload Me
End Sub

Private Sub Form_Load()
    Me.Caption = " Módulo de Control de Equipos de Monitoreo SPARTON"
    lblVersion.Caption = "Versión 1.0"
End Sub

Public Sub StartSysInfo()
    On Error GoTo SysInfoErr

    Dim rc As Long
    Dim SysInfoPath As String

    ' Intentar obtener ruta de acceso y nombre del programa de Info. del sistema a
partir del Registro...
    If GetKeyValue(HKEY_LOCAL_MACHINE, gREGKEYSYSINFO,
gREGVALSYSINFO, SysInfoPath) Then
        ' Intentar obtener sólo ruta del programa de Info. del sistema a partir del
Registro...
        ElseIf GetKeyValue(HKEY_LOCAL_MACHINE, gREGKEYSYSINFOLOC,
gREGVALSYSINFOLOC, SysInfoPath) Then
            ' Validar la existencia de versión conocida de 32 bits del archivo
            If (Dir(SysInfoPath & "\MSINFO32.EXE") <> "") Then
                SysInfoPath = SysInfoPath & "\MSINFO32.EXE"
            End If
        End If
    End If
End Sub

```

```

        ' Error: no se puede encontrar el archivo...
    Else
        GoTo SysInfoErr
    End If
' Error: no se puede encontrar la entrada del Registro...
Else
    GoTo SysInfoErr
End If

Call Shell(SysInfoPath, vbNormalFocus)

Exit Sub

SysInfoErr:
    MsgBox "La información del sistema no está disponible en este momento",
vbOKOnly

End Sub

Public Function GetKeyValue(KeyRoot As Long, KeyName As String, SubKeyRef
As String, ByRef KeyVal As String) As Boolean
    Dim i As Long                ' Contador de bucle
    Dim rc As Long               ' Código de retorno
    Dim hKey As Long             ' Controlador de una clave de
Registro abierta
    Dim hDepth As Long           '
    Dim KeyValType As Long       ' Tipo de datos de una clave de
Registro
    Dim tmpVal As String         ' Almacenamiento temporal para un
valor de clave de Registro
    Dim KeyValSize As Long       ' Tamaño de variable de clave de
Registro
    '-----
    ' Abrir clave de registro bajo KeyRoot {HKEY_LOCAL_MACHINE...}
    '-----
    rc = RegOpenKeyEx(KeyRoot, KeyName, 0, KEY_ALL_ACCESS, hKey) ' Abrir
clave de Registro

    If (rc <> ERROR_SUCCESS) Then GoTo GetKeyError            ' Error de
controlador...

    tmpVal = String$(1024, 0)                ' Asignar espacio de variable

```

```

KeyValSize = 1024                                ' Marcar tamaño de variable

'-----
' Obtener valor de clave de Registro...
'-----
rc = RegQueryValueEx(hKey, SubKeyRef, 0, _
                    KeyValType, tmpVal, KeyValSize) ' Obtener o crear valor de
clave

If (rc <> ERROR_SUCCESS) Then GoTo GetKeyError    ' Controlar errores

If (Asc(Mid(tmpVal, KeyValSize, 1)) = 0) Then      ' Win95 agregar cadena
terminada en valor nulo...
    tmpVal = Left(tmpVal, KeyValSize - 1)         ' Encontrado valor nulo, se va
a quitar de la cadena
Else                                              ' En WinNT las cadenas no terminan en
valor nulo...
    tmpVal = Left(tmpVal, KeyValSize)             ' No se ha encontrado valor
nulo, sólo se va a extraer la cadena
End If

'-----
' Determinar tipo de valor de clave para conversión...
'-----
Select Case KeyValType                          ' Buscar tipos de datos...
Case REG_SZ                                     ' Tipo de datos String de clave de
Registro
    KeyVal = tmpVal                             ' Copiar valor de cadena
Case REG_DWORD                                 ' Tipo de datos Double Word de
clave del Registro
    For i = Len(tmpVal) To 1 Step -1            ' Convertir cada bit
        KeyVal = KeyVal + Hex(Asc(Mid(tmpVal, i, 1))) ' Generar valor carácter a
carácter
    Next
    KeyVal = Format$("&h" + KeyVal)                ' Convertir Double Word a
cadena
End Select

GetKeyValue = True                              ' Se ha devuelto correctamente
rc = RegCloseKey(hKey)                          ' Cerrar clave de Registro
Exit Function                                    ' Salir

GetKeyError: ' Borrar después de que se produzca un error...
KeyVal = ""                                     ' Establecer valor a cadena vacía

```

```
    GetKeyValue = False                                ' Fallo de retorno
    rc = RegCloseKey(hKey)                             ' Cerrar clave de Registro
End Function
```

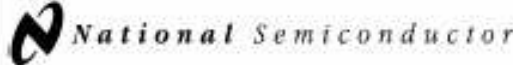
Código del módulo

```
' Variables públicas
Public Echo As Boolean    ' indicador de echo Encendido/Apagado.
Public CancelSend As Integer ' indicador para parar el envío de un archivo de
texto.

Declare Sub SetWindowPos Lib "user32" (ByVal hWnd As Long, ByVal
hWndInsertAfter As Long, ByVal X As Long, ByVal Y As Long, ByVal cx As Long,
ByVal cy As Long, ByVal wFlags As Long)
```

Anexo D. Datasheets

A continuación se muestra el datasheet del circuito integrado LM3524D.


June 1999

LM2524D/LM3524D Regulating Pulse Width Modulator

General Description

The LM3524D family is an improved version of the industry standard LM3524. It has improved specifications and additional features yet is pin for pin compatible with existing 3524 families. New features reduce the need for additional external circuitry often required in the original version.

The LM3524D has a $\pm 1\%$ precision 5V reference. The current carrying capability of the output drive transistors has been raised to 200 mA while reducing $V_{CE(sat)}$ and increasing V_{CE} breakdown to 60V. The common mode voltage range of the error-amp has been raised to 5.5V to eliminate the need for a resistive divider from the 5V reference.

In the LM3524D the circuit bias line has been isolated from the shut-down pin. This prevents the oscillator pulse amplitude and frequency from being disturbed by shut-down. Also at high frequencies (≈ 300 kHz) the max. duty cycle per output has been improved to 44% compared to 35% max. duty cycle in other 3524s.

In addition, the LM3524D can now be synchronized externally, through pin 3. Also a latch has been added to insure

one pulse per period even in noisy environments. The LM3524D includes double pulse suppression logic that insures when a shut-down condition is removed the state of the T-flip-flop will change only after the first clock pulse has arrived. This feature prevents the same output from being pulsed twice in a row, thus reducing the possibility of core saturation in push-pull designs.

Features

- Fully interchangeable with standard LM3524 family
- $\pm 1\%$ precision 5V reference with thermal shut-down
- Output current to 200 mA DC
- 60V output capability
- Wide common mode input range for error-amp
- One pulse per period (noise suppression)
- Improved max. duty cycle at high frequencies
- Double pulse suppression
- Synchronize through pin 3

Block Diagram

LM2524D/LM3524D Regulating Pulse Width Modulator

Absolute Maximum Ratings (Note 5)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Supply Voltage	40V
Collector Supply Voltage (LM2524D)	55V
(LM3524D)	40V
Output Current DC (each)	200 mA
Oscillator Charging Current (Pin 7)	5 mA

Internal Power Dissipation	1W
Operating Junction Temperature Range (Note 2)	
LM2524D	-40°C to +125°C
LM3524D	0°C to +125°C
Maximum Junction Temperature	150°
Storage Temperature Range	-55°C to +150°C
Lead Temperature (Soldering 4 sec.) M, N Pkg.	260°C

Electrical Characteristics

(Note 1)

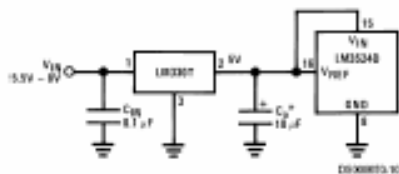
Symbol	Parameter	Conditions	LM2524D			LM3524D			Units
			Typ	Tested Limit (Note 3)	Design Limit (Note 4)	Typ	Tested Limit (Note 3)	Design Limit (Note 4)	
REFERENCE SECTION									
V_{REG}	Output Voltage		5	4.85 5.15	4.80 5.20	5	4.75 5.25		V_{REF} V_{MAX}
V_{OLR}	Line Regulation	$V_{IN} = 5V$ to 40V	10	15	30	10	25	50	mV/V _{REF}
V_{OLR}	Load Regulation	$I_L = 0$ mA to 20 mA	10	15	25	10	25	50	mV/V _{REF}
$\frac{\Delta V_{IN}}{\Delta V_{REF}}$	Ripple Rejection	$f = 120$ Hz	86			86			dB
I_{SC}	Short Circuit Current	$V_{REG} = 0$	50	25 150		50	25 200		mA Min mA Max
N_O	Output Noise	10 Hz $\leq f \leq$ 10 kHz	40		100	40		100	μV_{REF} Max
	Long Term Stability	$T_A = 125^\circ C$	20			20			mV/HR
OSCILLATOR SECTION									
f_{osc}	Max. Freq.	$R_f = 1k, C_f = 0.001 \mu F$ (Note 7)	550		500	350			kHz _{Max}
f_{osc}	Initial Accuracy	$R_f = 5.6k, C_f = 0.01 \mu F$ (Note 7)	20	17.5 22.5		20	17.5 22.5		kHz _{Max} kHz _{Max}
		$R_f = 2.7k, C_f = 0.01 \mu F$ (Note 7)	30	34 42		30	30 45		kHz _{Max} kHz _{Max}
Δf_{osc}	Freq. Change with V_{IN}	$V_{IN} = 5$ to 40V	0.5	1		0.5	1.0		% _{Max}
Δf_{osc}	Freq. Change with Temp.	$T_A = -55^\circ C$ to $+125^\circ C$ at 20 kHz $R_f = 5.6k,$ $C_f = 0.01 \mu F$	5			5			%
V_{osc}	Output Amplitude (Pin 3) (Note 8)	$R_f = 5.6k, C_f = 0.01 \mu F$	3	2.4		3	2.4		V_{REF}
t_{pw}	Output Pulse Width (Pin 3)	$R_f = 5.6k, C_f = 0.01 \mu F$	0.5	1.5		0.5	1.5		μs_{Max}
	Sawtooth Peak Voltage	$R_f = 5.6k, C_f = 0.01 \mu F$	3.4	3.6	3.8		3.8		V_{REF}
	Sawtooth Valley Voltage	$R_f = 5.6k, C_f = 0.01 \mu F$	1.1	0.8	0.6		0.6		V_{REF}
ERROR-AMP SECTION									
V_{io}	Input Offset Voltage	$V_{CM} = 2.5V$	2	8	10	2	10		mV _{REF}
I_b	Input Bias	$V_{CM} = 2.5V$	1	8	10	1	10		μA_{Max}

Functional Description

INTERNAL VOLTAGE REGULATOR

The LM3524D has an on-chip 5V, 50 mA, short circuit protected voltage regulator. This voltage regulator provides a supply for all internal circuitry of the device and can be used as an external reference.

For input voltages of less than 8V the 5V output should be shorted to pin 15, V_{IH} , which disables the 5V regulator. With these pins shorted the input voltage must be limited to a maximum of 6V. If input voltages of 6V–8V are to be used, a pre-regulator, as shown in Figure 1, must be added.



*Minimum C_2 of 10 μF required for stability.

FIGURE 1.

OSCILLATOR

The LM3524D provides a stable on-board oscillator. Its frequency is set by an external resistor, R_T , and capacitor, C_T . A graph of R_T , C_T vs oscillator frequency is shown in Figure 2. The oscillator's output provides the signals for triggering an internal flip-flop, which directs the PWM information to the outputs, and a blanking pulse to turn off both outputs during transitions to ensure that cross conduction does not occur. The width of the blanking pulse, or dead time, is controlled by the value of C_T , as shown in Figure 3. The recommended values of R_T are 1.8 k Ω to 100 k Ω , and for C_T , 0.001 μF to 0.1 μF .

If two or more LM3524D's must be synchronized together, the easiest method is to interconnect all pin 3 terminals, tie all pin 7's (together) to a single C_T , and leave all pin 6's open except one which is connected to a single R_T . This method works well unless the LM3524D's are more than 6" apart.

A second synchronization method is appropriate for any circuit layout. One LM3524D, designated as master, must have its R_T , C_T set for the correct period. The other slave LM3524D(s) should each have an R_T , C_T set for a 10% longer period. All pin 3's must then be interconnected to allow the master to properly reset the slave units.

The oscillator may be synchronized to an external clock source by setting the internal free-running oscillator frequency 10% slower than the external clock and driving pin 3 with a pulse train (approx. 3V) from the clock. Pulse width should be greater than 50 ns to insure full synchronization.

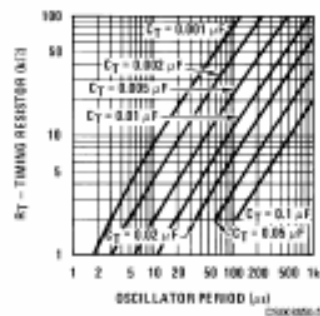


FIGURE 2.

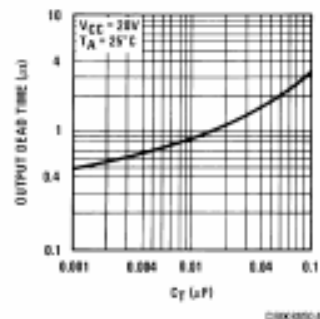


FIGURE 3.

ERROR AMPLIFIER

The error amplifier is a differential input, transconductance amplifier. Its gain, nominally 86 dB, is set by either feedback or output loading. This output loading can be done with either purely resistive or a combination of resistive and reactive components. A graph of the amplifier's gain vs output load resistance is shown in Figure 4.

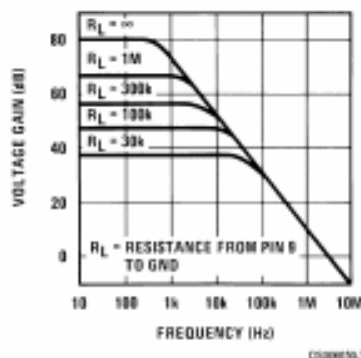


FIGURE 4.

The output of the amplifier, or input to the pulse width modulator, can be overridden easily as its output impedance is very high ($Z_o = 5 \text{ M}\Omega$). For this reason a DC voltage can be

Functional Description (Continued)

applied to pin 9 which will override the error amplifier and force a particular duty cycle to the outputs. An example of this could be a non-regulating motor speed control where a variable voltage was applied to pin 9 to control motor speed. A graph of the output duty cycle vs the voltage on pin 9 is shown in Figure 5.

The duty cycle is calculated as the percentage ratio of each output's ON-time to the oscillator period. Paralleling the outputs doubles the observed duty cycle.

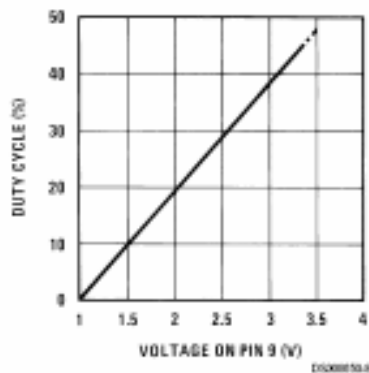


FIGURE 5.

The amplifier's inputs have a common-mode input range of 1.5V–5.5V. The on board regulator is useful for biasing the inputs to within this range.

CURRENT LIMITING

The function of the current limit amplifier is to override the error amplifier's output and take control of the pulse width. The output duty cycle drops to about 25% when a current limit sense voltage of 200 mV is applied between the +CL and -CL sense terminals. Increasing the sense voltage approximately 5% results in a 0% output duty cycle. Care should be taken to ensure the -0.7V to +1.0V input common-mode range is not exceeded.

In most applications, the current limit sense voltage is produced by a current through a sense resistor. The accuracy of this measurement is limited by the accuracy of the sense resistor, and by a small offset current, typically 100 μ A, flowing from +CL to -CL.

OUTPUT STAGES

The outputs of the LM3524D are NPN transistors, capable of a maximum current of 200 mA. These transistors are driven 180° out of phase and have non-committed open collectors and emitters as shown in Figure 6.

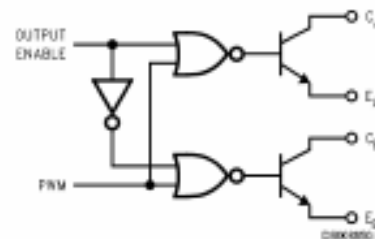
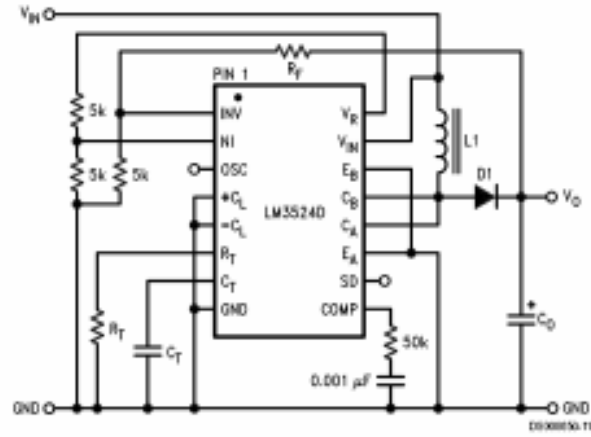


FIGURE 6.

Typical Applications



Design Equations

$$R_F = 5k \left(\frac{V_O}{2.5} - 1 \right)$$

$$f_{OSC} = \frac{1}{R_T C_T}$$

$$L1 = \frac{2.5V_{IN}^2 (V_O - V_{IN})}{f_{OSC} V_O^2}$$

$$C_O = \frac{I_O (V_O - V_{IN})}{f_{OSC} \Delta V_O V_O}$$

$$I_{O(MAX)} = I_{IN} \frac{V_{IN}}{V_O}$$

FIGURE 7. Positive Regulator, Step-Up Basio Configuration ($I_{IN(MAX)}$) = 80 mA)

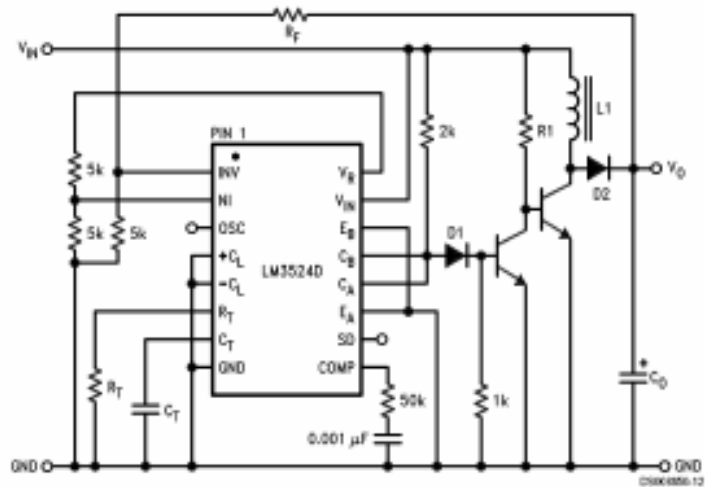
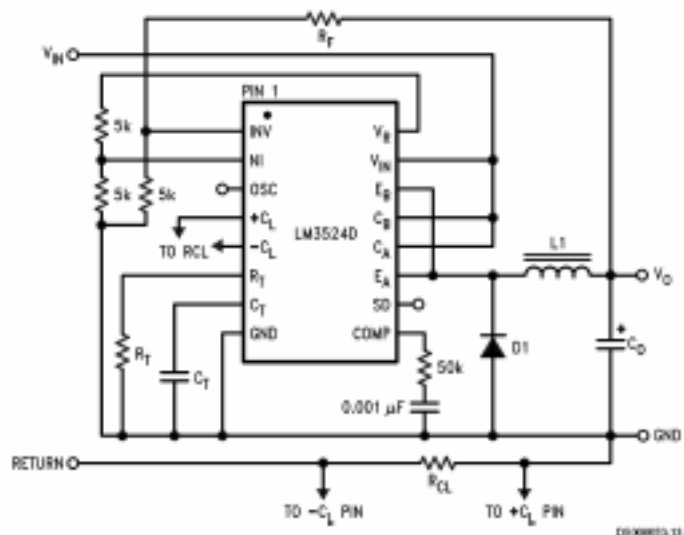


FIGURE 8. Positive Regulator, Step-Up Boosted Current Configuration

Typical Applications (Continued)



Design Equations

$$R_T = 5 \text{ k}\Omega \left(\frac{V_O}{2.5} - 1 \right)$$

$$R_{CL} = \frac{\text{Current Limit Sense Volt}}{I_{O(\text{MAX})}}$$

$$f_{\text{OSC}} = \frac{1}{R_T C_T}$$

$$L1 = \frac{2.5V_O (V_{IN} - V_O)}{I_O V_{IN} f_{\text{OSC}}}$$

$$C_O = \frac{(V_{IN} - V_O) V_O I_T^2}{8 \Delta V_O V_{IN} L1}$$

$$I_{O(\text{MAX})} = I_{IN} \frac{V_{IN}}{V_O}$$

FIGURE 9. Positive Regulator, Step-Down Basic Configuration ($I_{IN(\text{MAX})} = 80 \text{ mA}$)

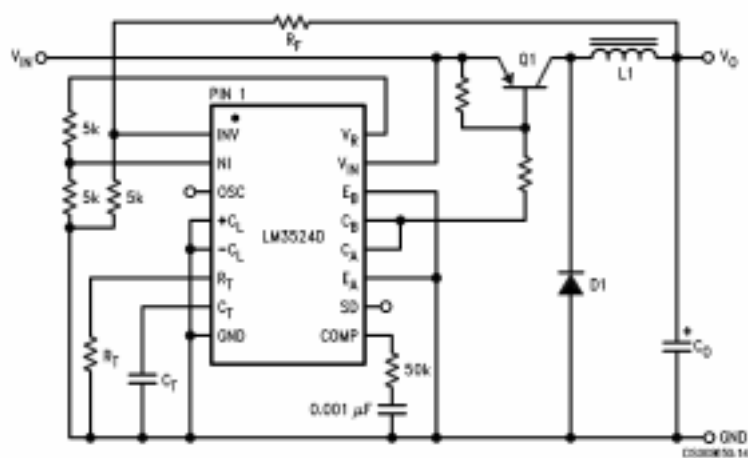


FIGURE 10. Positive Regulator, Step-Down Boosted Current Configuration

Typical Applications (Continued)

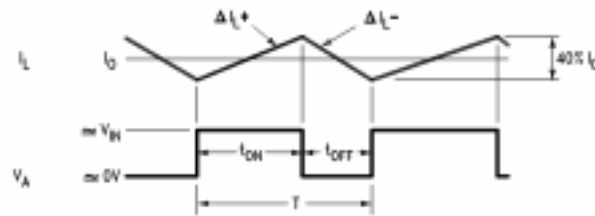


FIGURE 13. Relation of Switch Timing to Inductor Current in Step-Down Regulator

$$\text{From the relation } V_L = L \frac{di}{dt}, \Delta I_L \approx \frac{V_L T}{L1}$$

$$\Delta I_L^+ = \frac{(V_{IN} - V_o) t_{ON}}{L1}, \Delta I_L^- = \frac{V_o t_{OFF}}{L1}$$

Neglecting V_{SAT} , V_{DI} , and setting $\Delta I_L^+ = \Delta I_L^-$:

$$V_o \approx V_{IN} \left(\frac{t_{ON}}{t_{OFF} + t_{ON}} \right) = V_{IN} \left(\frac{t_{ON}}{T} \right);$$

where T = Total Period

The above shows the relation between V_{IN} , V_o and duty cycle.

$$I_{IN(DC)} = I_{OUT(DC)} \left(\frac{t_{ON}}{t_{ON} + t_{OFF}} \right),$$

as Q1 only conducts during t_{ON} .

$$P_{IN} = I_{IN(DC)} V_{IN} = (I_o(DC)) \left(\frac{t_{ON}}{t_{ON} + t_{OFF}} \right) V_{IN}$$

$$P_o = I_o V_o$$

The efficiency, η , of the circuit is:

$$\eta_{MAX} = \frac{P_o}{P_{IN}} = \frac{I_o V_o}{I_o \left(\frac{t_{ON}}{T} \right) V_{IN} + \frac{(V_{SAT} t_{ON} + V_{DI} t_{OFF})}{T} I_o}$$

$$= \frac{V_o}{V_o + 1} \text{ for } V_{SAT} = V_{DI} = 1V.$$

η_{MAX} will be further decreased due to switching losses in Q1. For this reason Q1 should be selected to have the maximum possible f_T , which implies very fast rise and fall times.

CALCULATING INDUCTOR L1

$$t_{ON} \approx \frac{(\Delta I_L^+) \times L1}{(V_{IN} - V_o)}, t_{OFF} = \frac{(\Delta I_L^-) \times L1}{V_o}$$

$$t_{ON} + t_{OFF} = T = \frac{(\Delta I_L^+) \times L1}{(V_{IN} - V_o)} + \frac{(\Delta I_L^-) \times L1}{V_o}$$

$$= \frac{0.4 I_o L1}{(V_{IN} - V_o)} + \frac{0.4 I_o L1}{V_o}$$

Since $\Delta I_L^+ = \Delta I_L^- = 0.4 I_o$

Solving the above for L1

$$L1 = \frac{2.5 V_o (V_{IN} - V_o)}{I_o V_{IN} f}$$

where: L1 is in Henrys

f is switching frequency in Hz

Also, see LM1578 data sheet for graphical methods of inductor selection.

CALCULATING OUTPUT FILTER CAPACITOR C_o :

Figure 13 shows L1's current with respect to Q1's t_{ON} and t_{OFF} times (V_A is at the collector of Q1). This current must flow to the load and C_o . C_o 's current will then be the difference between i_L and I_o .

$$i_{C_o} = i_L - I_o$$

From Figure 13 it can be seen that current will be flowing into C_o for the second half of t_{ON} through the first half of t_{OFF} , or a time, $t_{ON}/2 + t_{OFF}/2$. The current flowing for this time is $\Delta I_L/4$. The resulting ΔV_c or ΔV_o is described by:

$$\Delta V_{opp} = \frac{1}{C} \times \frac{\Delta I_L}{4} \times \left(\frac{t_{ON}}{2} + \frac{t_{OFF}}{2} \right)$$

$$= \frac{\Delta I_L}{4C} \left(\frac{t_{ON} + t_{OFF}}{2} \right)$$

$$\text{Since } \Delta I_L = \frac{V_o(T - t_{ON})}{L1} \text{ and } t_{ON} = \frac{V_o T}{V_{IN}}$$

$$\Delta V_{opp} = \frac{V_o \left(T - \frac{V_o T}{V_{IN}} \right)}{4C L1} \left(\frac{T}{2} \right) = \frac{(V_{IN} - V_o) V_o T^2}{8 V_{IN} C_o L1} \text{ or}$$

$$C_o = \frac{(V_{IN} - V_o) V_o T^2}{8 \Delta V_o V_{IN} L1}$$

where: C is in farads, T is $\frac{1}{\text{switching frequency}}$

ΔV_o is p-p output ripple

For best regulation, the inductor's current cannot be allowed to fall to zero. Some minimum load current I_o and thus inductor current, is required as shown below:

$$I_{o(MIN)} = \frac{(V_{IN} - V_o) t_{ON}}{2L1} = \frac{(V_{IN} - V_o) V_o}{2 V_{IN} L1}$$

Typical Applications (Continued)

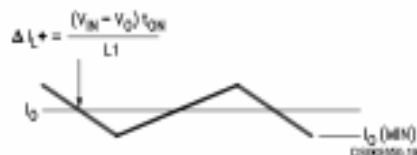


FIGURE 14. Inductor Current Slope in Step-Down Regulator

A complete step-down switching regulator schematic, using the LM3524D, is illustrated in Figure 15. Transistors Q1 and Q2 have been added to boost the output to 1A. The 5V regulator of the LM3524D has been divided in half to bias the error amplifier's non-inverting input to within its common-mode

range. Since each output transistor is on for half the period, actually 45%, they have been paralleled to allow longer possible duty cycle, up to 90%. This makes a lower possible input voltage. The output voltage is set by:

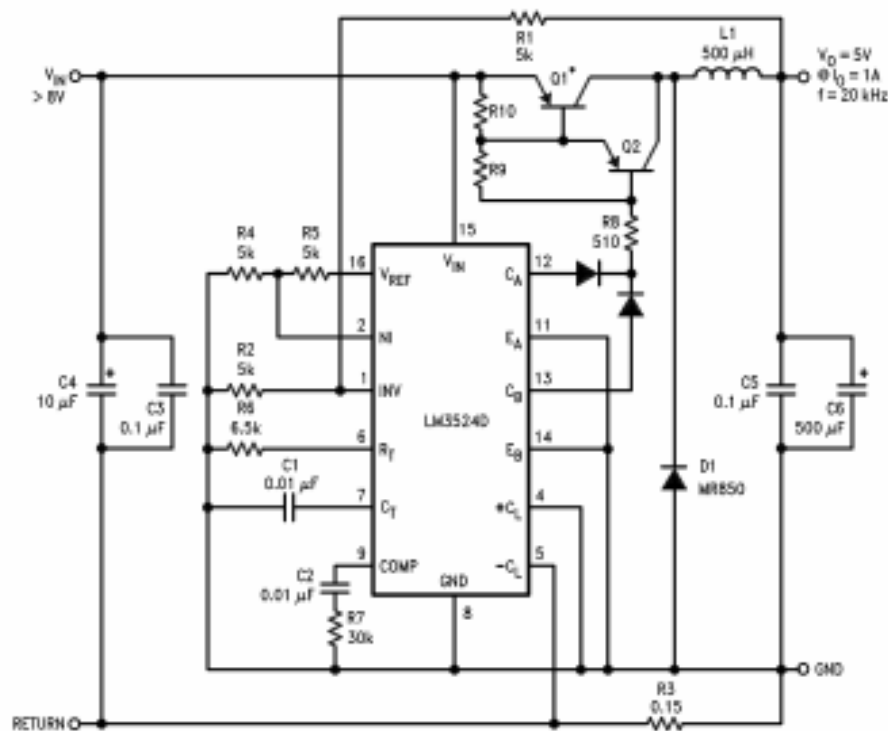
$$V_O = V_{NI} \left(1 + \frac{R1}{R2} \right),$$

where V_{NI} is the voltage at the error amplifier's non-inverting input.

Resistor R3 sets the current limit to:

$$\frac{200 \text{ mV}}{R3} = \frac{200 \text{ mV}}{0.15} = 1.3\text{A},$$

Figures 16, 17 and show a PC board layout and stuffing diagram for the 5V, 1A regulator of Figure 15. The regulator's performance is listed in Table 1.



*Mounted to Silver Heatlink No. VS-1.

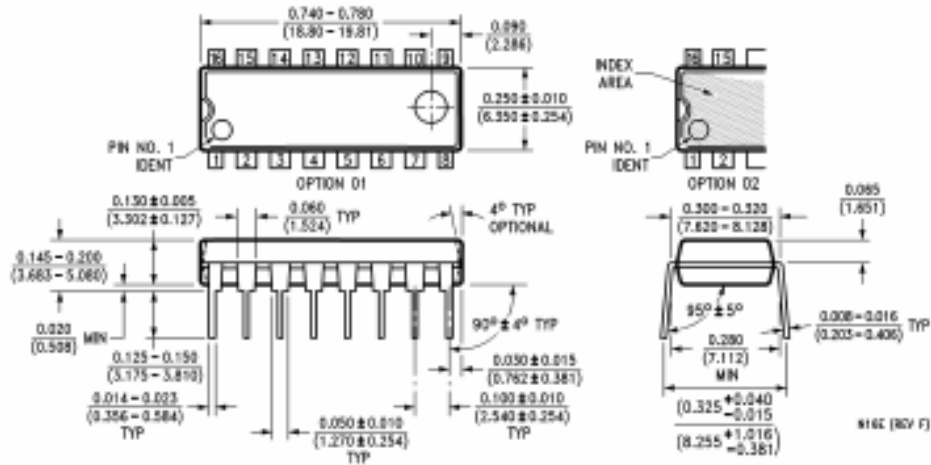
Q1 = 2N344

Q2 = 2N5023

L1 = #40 turns No. 22 wire on Ferracube No. K300502 Toroid core.

FIGURE 15. 5V, 1 Amp Step-Down Switching Regulator

Physical Dimensions inches (millimeters) unless otherwise noted (Continued)



Molded Dual-In-Line Package (N)
Order Number LM2524DN or LM3524DN
N8 Package Number N18E

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

 National Semiconductor Corporation
 America
 Tel: 1-800-273-9659
 Fax: 1-800-737-7210
 Email: support@nsc.com
 www.national.com

National Semiconductor Europe
 Fax: +49 (0) 1 90-530 05 05
 Email: europe.support@nsc.com
 Deutsch Tel: +49 (0) 1 80-532 05 05
 English Tel: +49 (0) 1 80-532 78 32
 Francais Tel: +49 (0) 1 80-532 93 58
 Italiano Tel: +49 (0) 1 80-534 16 90

National Semiconductor Asia Pacific Customer Response Group
 Tel: 65-2544499
 Fax: 65-2544466
 Email: asa.support@nsc.com

National Semiconductor Japan Ltd.
 Tel: 81-3-5639-7580
 Fax: 81-3-5639-7507

A continuación el datasheet de circuito integrado CM8870.



CMOS Integrated DTMF Receiver

Features

- Full DTMF receiver
- Less than 35mW power consumption
- Industrial temperature range
- Uses quartz crystal or ceramic resonators
- Adjustable acquisition and release times
- 18-pin DIP, 18-pin DIP EIAJ, 18-pin SOIC, 20-pin PLOC
- CM8870C
 - Power down mode
 - Inhibit mode
 - Buffered OSC3 output (PLOC package only)
- CM8870C is fully compatible with CM8870 for 18-pin devices by grounding pins 5 and 6

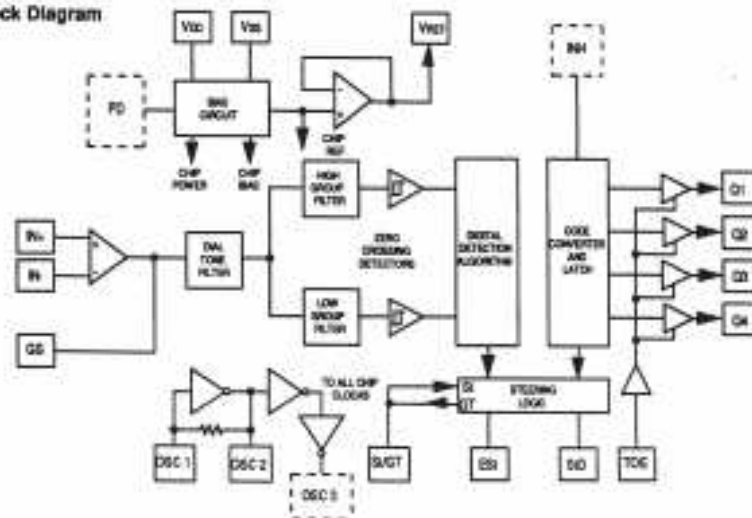
Applications

- PABX
- Central office
- Mobile radio
- Remote control
- Remote data entry
- Call limiting
- Telephone answering systems
- Paging systems

Product Description

The CAMD CM8870/70C provides full DTMF receiver capability by integrating both the bandpass filter and digital decoder functions into a single 18-pin DIP, SOIC, or 20-pin PLOC package. The CM8870/70C is manufactured using state-of-the-art CMOS process technology for low power consumption (35mW, max.) and precise data handling. The filter section uses a switched capacitor technique for both high and low group filters and dial tone rejection. The CM8870/70C decoder uses digital counting techniques for the detection and decoding of all 16 DTMF tone pairs into a 4-bit code. This DTMF receiver minimizes external component count by providing an on-chip differential input amplifier, clock generator, and a latched three-state interface bus. The on-chip clock generator requires only a low cost TV crystal or ceramic resonator as an external component.

Block Diagram





Absolute Maximum Ratings: (Note 1)

ABSOLUTE MAXIMUM RATINGS		
Parameter	Symbol	Value
Power Supply Voltage (V_{DD} - V_{SS})	V_{DD}	5.0V Max
Voltage on any Pin	V_{dc}	$V_{SS}-0.3V$ to $V_{DD}+0.3V$
Current on any Pin	I_{DD}	10mA Max
Operating Temperature	T_A	-40°C to +85°C
Storage Temperature	T_S	-65°C to +150°C

This device contains input protection against damage due to high static voltages or electric fields; however, precautions should be taken to avoid application of voltages higher than the maximum rating.

Notes:

1. Exceeding these ratings may cause permanent damage, functional operation under these conditions is not implied.

DC Characteristics: All voltages referenced to V_{SS} , $V_{DD} = 5.0V \pm 5\%$, $T_A = -40^\circ C$ to $+85^\circ C$ unless otherwise noted.

DC CHARACTERISTICS						
Parameter	Symbol	Min	Typ	Max	Units	Test Conditions
Operating Supply Voltage	V_{DD}	4.75		5.25	V	
Operating Supply Current	I_{DD}		3.0	7.0	mA	
Standby Supply Current	I_{DDQ}			25	μA	$PD = V_{DD}$
Power Consumption	P_D		15	35	mW	$f = 3.579$ MHz; $V_{DD} = 5.0V$
Low Level Input Voltage	V_{IL}			1.5	V	$V_{DD} = 5.0V$
High Level Input Voltage	V_{IH}	3.5			V	$V_{DD} = 5.0V$
Input Leakage Current	I_{IH}/I_{IL}		0.1		μA	$V_{IH} = V_{SS} = V_{DD}$ (Note 1)
Pull Up (Source) Current on TOE	I_{SS}		6.5	20	μA	TOE=0V, $V_{DD} = 5.0V$
Input Impedance, (IN+, IN-)	R_{IN}	8	10		M Ω	@ 1KHz
Steering Threshold Voltage	V_{TST}	2.2		2.5	V	$V_{DD} = 5.0V$
Low Level Output Voltage	V_{OL}			0.03	V	$V_{DD} = 5.0V$, No Load
High Level Output Voltage	V_{OH}	4.97			V	$V_{DD} = 5.0V$, No Load
Output Low (Sink) Current	I_{OL}	1.0	2.5		mA	$V_{OUT} = 0.4V$
Output High (Source) Current	I_{OH}	0.4	0.8		mA	$V_{OUT} = 4.6V$
Output Voltage	V_{REF}	2.4		2.7	V	$V_{DD} = 5.0V$, No Load
Output Resistance	R_{OUT}		10		K Ω	

Operating Characteristics: All voltages referenced to V_{SS} , $V_{DD} = 5.0V \pm 5\%$, $T_A = -40^\circ C$ to $+85^\circ C$ unless otherwise noted.

Gain Setting Amplifier

OPERATING CHARACTERISTICS						
Parameter	Symbol	Min	Typ	Max	Units	Test Conditions
Input Leakage Current	I_{IN}			± 100	nA	$V_{SS} < V_{IN} < V_{DD}$
Input Resistance	R_{IN}	10			M Ω	
Input Offset Voltage	V_{OS}			± 25	mV	
Power Supply Rejection	PSRR	50			dB	1KHz (Note 12)
Common Mode Rejection	CMRR	40			dB	$-3.0V < V_{IN} < 3.0V$
DC Open Loop Voltage Gain	A_{VOL}	32			dB	
Open Loop Unity Gain Bandwidth	f_c	0.3			MHz	
Output Voltage Swing	V_O	4.0			V $_{PP}$	$R_L \geq 100K\Omega$ to V_{SS}
Maximum Capacitive Load (GS)	C_L			100	pF	
Maximum Resistive Load (GS)	R_L			50	K Ω	
Common Mode Range (No Load)	V_{cm}	2.5			V $_{PP}$	No Load



AC Characteristics: All voltages referenced to V_{DD} , $V_{DD}=5.0V \pm 5\%$, $T_A=-40^\circ C$ to $+85^\circ C$, $f_{CLK}=3.579545$ MHz using test circuit (Fig. 1) unless otherwise noted.

AC CHARACTERISTICS						
Parameter	Symbol	Min	Typ	Max	Units	Notes
Valid Input Signal Levels (each tone of composite signal)		-29		+1	dBm	1,2,3,4,5,8
		27.5		869	mV _{RMS}	
Positive Tolerant Accept				10	dB	2,3,4,8
Negative Tolerant Accept				10	dB	
Freq. Deviation Accept Limit				1.5%±2Hz	Nom.	2,3,5,8,10
Freq. Deviation Reject Limit		±3.5%			Nom.	2,3,5
Third Tone Tolerance			-16		dB	2,3,4,5,8,9,13,14
Noise Tolerance			-12		dB	2,3,4,5,6,8,9
Dial Tone Tolerance			+22		dB	2,3,4,5,7,8,9
Tone Present Detection Time	t _{DP}	5	8	14	mS	Refer to Timing Diagram
Tone Absent Detection Time	t _{DA}	0.5	3	8.5	mS	
Min Tone Duration Accept	t _{DAc}			40	mS	(User Adjustable) Times shown are obtained with circuit in Fig. 1)
Max Tone Duration Reject	t _{DRc}	20			mS	
Min. Interdigit Pause Accept	t _{IP}			40	mS	
Max. Interdigit Pause Reject	t _{IPR}	20			µS	
Propagation Delay (St to Q)	t _{PQ}		6	11	µS	TOE = V _{DD}
Propagation Delay (St to Sd)	t _{PSD}		9	16	µS	
Output Data Set Up (Q to Sd)	t _{QSD}		3.4		µS	
Propagation Delay (TOE to Q)	Enable	t _{TE}		50	nS	R _L = 10KΩ C _L = 50pF
	Disable	t _{TD}		300	nS	
Crystal/Clock Frequency	f _{CLK}	3.5759	3.5795	3.5831	MHz	
Clock Output (OSC 2)	Capacitive Load	C _{LO}			30	pF

Notes:

- | | |
|--|---|
| <p>1. dBm = decibels above or below a reference power of 1 mW into a 600 ohm load.</p> <p>2. Digit sequence consists of all 16 DTMF tones.</p> <p>3. Tone duration = 40mS. Tone pause = 40 mS.</p> <p>4. Nominal DTMF frequencies are used.</p> <p>5. Both tones in the composite signal have an equal amplitude.</p> <p>6. Bandwidth limited (0 to 3 KHz) Gaussian Noise.</p> <p>7. The precise dial tone frequencies are (350 Hz and 440 Hz) ±2%.</p> <p>8. For an error rate of better than 1 in 10,000</p> | <p>9. Referenced to lowest level frequency component in DTMF signal.</p> <p>10. Minimum signal acceptance level is measured with specified maximum frequency deviation.</p> <p>11. Input pins defined as IN+, IN-, and TOE.</p> <p>12. External voltage source used to bias V_{DD}.</p> <p>13. This parameter also applies to a third tone injected onto the power supply.</p> <p>14. Referenced to Figure 1. Input DTMF tone level at -28 dBm.</p> |
|--|---|



Pin Function Table

PIN FUNCTION		
Name	Description	
IN+	Non-inverting Input	Connection to the front-end differential amplifier
IN-	Inverting Input	
GS	Gain Select	Gives access to output of front-end differential amplifier for connection of feedback resistor.
V _{REF}	Reference voltage output (nominally V _{DD} /2). May be used to bias the inputs at mid-rail.	
INH	Inhibits detection of tones represents keys A, B, C, and D	
OSC3	Digital buffered oscillator output.	
PD	Power Down	Logic high powers down the device and inhibits the oscillator.
OSC1	Clock Input	3.579545 MHz crystal connected between these pins completes internal oscillator.
OSC2	Clock Output	
V _{SS}	Negative power supply (normally connected to GND).	
TOE	Three-state output enable (input). Logic high enables the outputs Q ₁ -Q ₄ . Internal pull-up.	
Q ₁ Q ₂ Q ₃ Q ₄	Three-state outputs. When enabled by TOE, provides the code corresponding to the last valid tone pair received. (See Fig. 2).	
StD	Delayed steering output. Presents a logic high when a received tone pair has been registered and the output latch is updated. Returns to logic low when the voltage on St/GT falls below V _{REF} .	
Est	Early steering output. Presents a logic high immediately when the digital algorithm detects a recognizable tone pair (signal condition). Any momentary loss of signal condition will cause Est to return to a logic low.	
St/Gt	Steering input/guard time output (bidirectional). A voltage greater than V _{REF} detected a St causes the device to register the detected tone pair. The GT output acts to reset the external steering time constant, and its state is a function of Est and the voltage on St. (See Fig. 2)	
V _{DD}	Positive power supply.	
IC	Internal Connection.	Must be tied to V _{SS} (for 8870 configuration only)

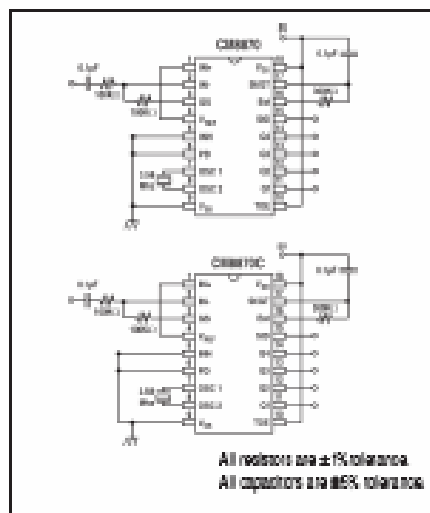


Figure 1.
Single Ended Input Configuration

F _{LOW}	F _{HIGH}	KEY	TOW	Q ₄	Q ₃	Q ₂	Q ₁
697	1209	1	H	0	0	0	1
697	1336	2	H	0	0	1	0
697	1477	3	H	0	0	1	1
770	1209	4	H	0	1	0	0
770	1336	5	H	0	1	0	1
770	1477	6	H	0	1	1	0
852	1209	7	H	0	1	1	1
852	1336	8	H	1	0	0	0
852	1477	9	H	1	0	0	1
941	1209	0	H	1	0	1	0
941	1336	.	H	1	0	1	1
941	1477	#	H	1	1	0	0
697	1633	A	H	1	1	0	1
770	1633	B	H	1	1	1	0
852	1633	C	H	1	1	1	1
941	1633	D	H	0	0	0	0
-	-	ANY	L	Z	Z	Z	Z

L = Logic Low, H = Logic High, Z = High Impedance

Figure 2.
Function of Diode Table

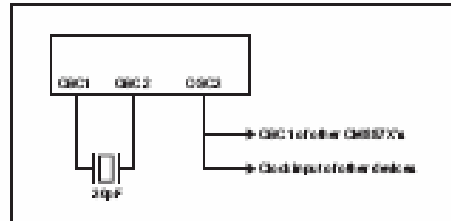


Figure 7. CM8870C Crystal Connection (PLCC Package Only)

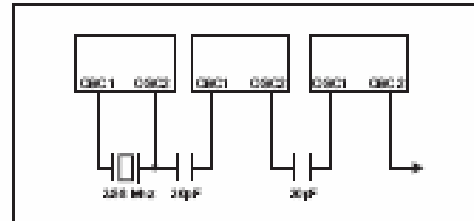
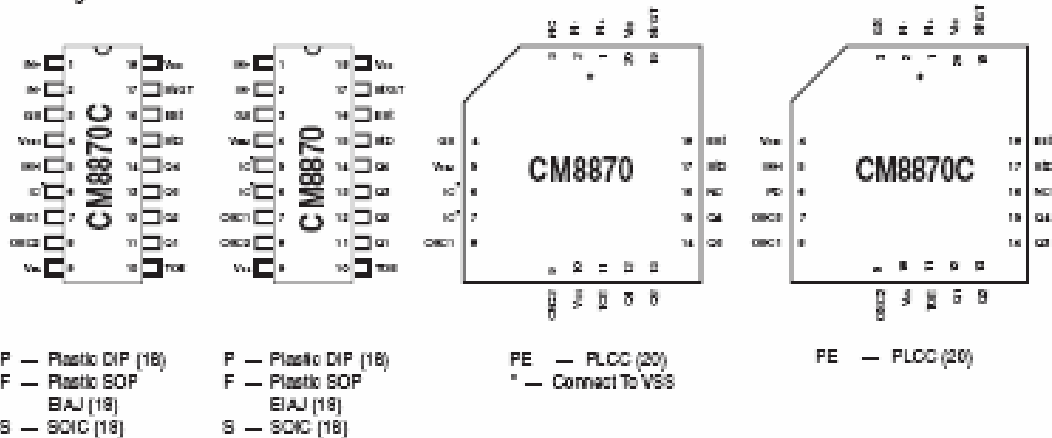
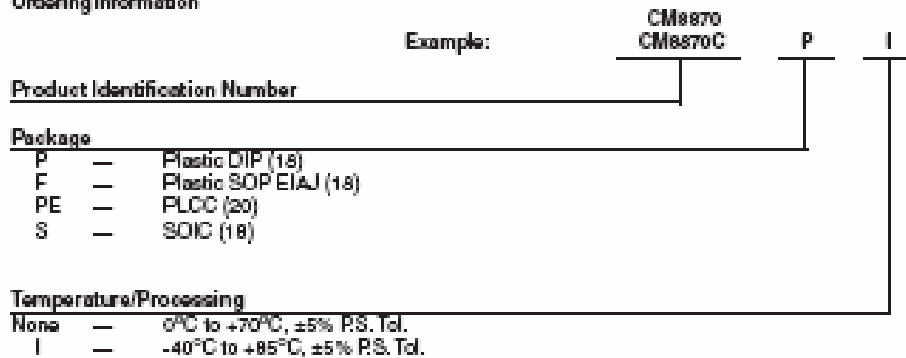


Figure 8. CM8870/70C Crystal Connection

Pin Assignments



Ordering Information



A continuación el datasheet del transistor TIP31C.

UTC TIP31C NPN EXPTAXIAL PLANAR TRANSISTOR

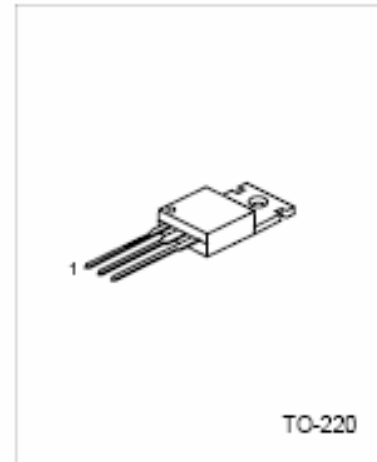
NPN EXPTAXIAL PLANAR TRANSISTOR

DESCRIPTION

The UTC TIP31C is a NPN epitaxial planar transistor, designed for using in general purpose amplifier and switching applications.

FEATURE

*Complement to tip32C



1:BASE 2:COLLECTOR 3:EMITTER

ABSOLUTE MAXIMUM RATINGS

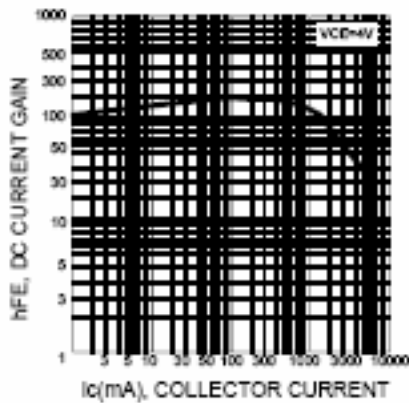
PARAMETER	SYMBOL	VALUE	UNIT
Collector Base Voltage	V _{CB0}	100	V
Collector to Emitter Voltage	V _{CE0}	100	V
Emitter To Base Voltage	V _{EB0}	5	V
Collector Current(DC)	I _C	3	A
Collector Current(Pulse)	I _C	5	A
Base Current	I _B	1	A
Collector Dissipation(T _c =25°C)	P _C	40	W
Collector Dissipation(T _a =25°C)	P _C	2	W
Junction Temperature	T _J	150	°C
Storage Temperature	T _{stg}	-65 ~ +150	°C

ELECTRICAL CHARACTERISTICS (T_c=25°C)

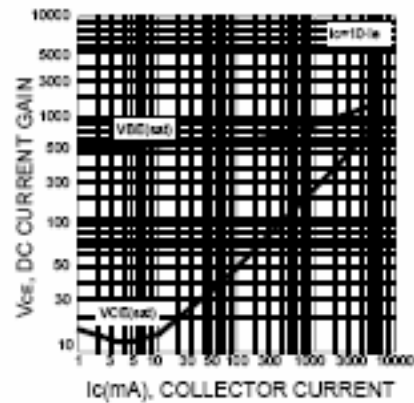
PARAMETER	SYMBOL	TEST CONDITIONS	MIN	TYP	MAX	UNIT
Collector Emitter Sustaining Voltage(*)	BV _{CEO}	I _C =30mA, I _B =0	100			V
Collect Cutoff Current	I _{CES}	V _{CB} =100V, V _{EB} =0			200	μA
Collector Cutoff Current	I _{CEO}	V _{CE} =60V, I _B =0			0.3	mA
Emitter Cutoff Current	I _{EB0}	V _{BE} =5V, I _C =0			1	mA
Collector-Emitter Saturation Voltage(*)	V _{CE(sat)}	I _C =3A, I _B =375mA			1.2	V
Base-Emitter On Voltage(*)	V _{BE(on)}	I _C =3A, V _{CE} =4V			1.8	V
DC Current Gain(*)	h _{FE}	I _C =1A, V _{CE} =4V	25			
		I _C =3A, V _{CE} =4V	10		50	
Current Gain Bandwidth Product	f _T	I _C =0.5A, V _{CE} =10V f=1MHz	3			MHz

*Pulse Test: PW<=300μs, Duty Cycle<=2%

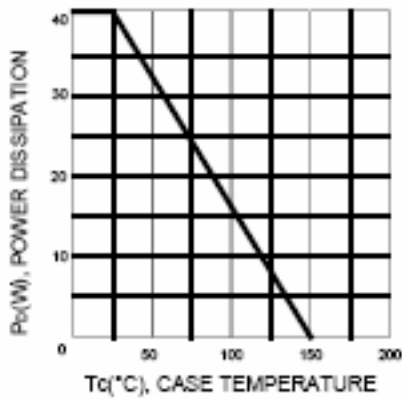
DC CURRENT GAIN



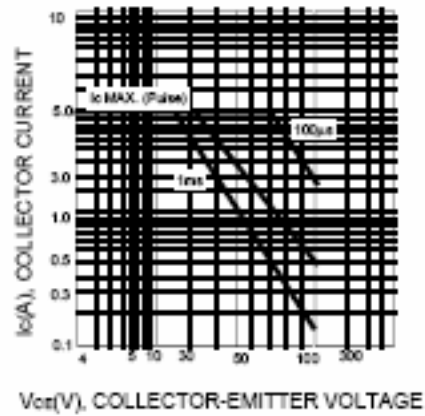
BASE-EMITTER SATURATION VOLTAGE
COLLECTOR-EMITTER SATURATION VOLTAGE



POWER DERATING



SAFE OPERATING AREA



UTC assumes no responsibility for equipment failures that result from using products at values that exceed, even momentarily, rated values (such as maximum ratings, operating condition ranges, or other parameters) listed in products specifications of any and all UTC products described or contained herein. UTC products are not designed for use in life support appliances, devices or systems where malfunction of these products can be reasonably expected to result in personal injury. Reproduction in whole or in part is prohibited without the prior written consent of the copyright owner. The information presented in this document does not form part of any quotation or contract, is believed to be accurate and reliable and may be changed without notice.



XE9624F & XE9624FS

10-03-95

A Complete Send and Receive Fax & Data Modem

Description

Xecom's XE9624F and XE9624FS include a 2400bps data modem, a Group III Send and Receive fax modem and the telephone interface in one compact component. Both models include user transferable FCC Part 68 registration.

Xecom puts the XE9624F in our dual inline package. The XE9624F is pin compatible with other Xecom modems including the XE2401, XE2496, and XE9624E.

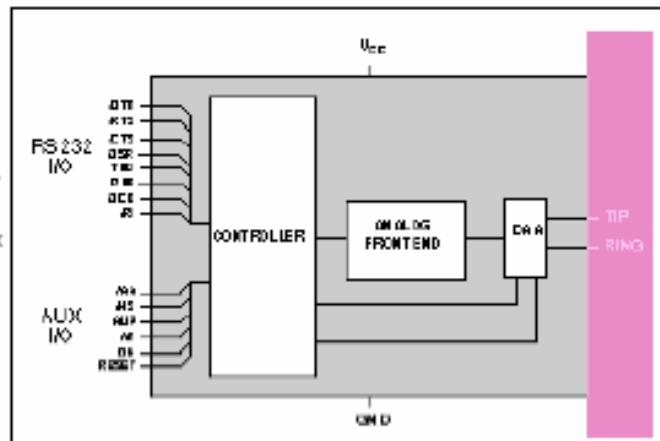
The XE9624FS comes in a Single Inline Package. The XE9624FS can be ordered for either vertical (XE9624FS1) or horizontal (XE9624FS2) mounting to optimize space utilization.

Xecom designed the XE9624F and XE9624FS specifically to provide computer, notebook, and industrial systems manufacturers with a complete, highly integrated, compact solution for adding data and fax communications.

Features

- Data Rates: 2400, 1200 and 300 bps
- Send fax to 9600 bps
- Receive fax to 4800 bps
- Enhanced "AT" commands for modem control and configuration.
- Class 1 commands for facsimile control.
- Single +5V supply
- Low power CMOS:
Operating: 200 mW (Typ.)
Sleep mode: 50 mW (Typ.)
- Dimensions:
XE9624F: 2.28" by 1.08" by 0.42"
XE9624FS: 1.53" by 0.46" by 0.92"

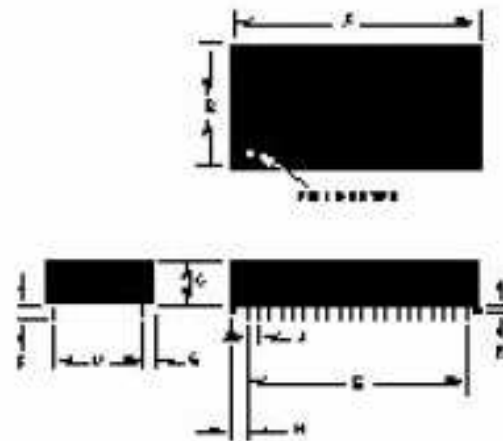
Block Diagram



XE9624F Mechanical Specifications

PIN	INCHES		METRIC(MM)	
	MIN	MAX	MIN	MAX
A	2.270	2.290	57.66	58.17
B	1.070	1.090	27.18	27.69
C	0.420	0.430	10.67	10.92
D	0.890	0.910	22.61	23.11
E	1.890	1.910	48.01	48.51
F	0.125	0.200	3.18	5.08
G	0.080	0.100	2.03	2.54
H	0.180	0.200	4.57	5.08
J	0.090	0.110	2.29	2.79
K	0.020	0.025	0.51	0.64

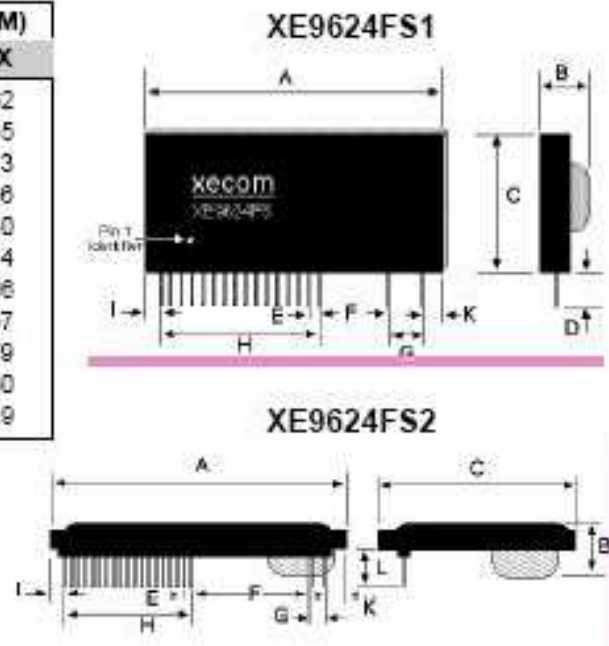
Pins = 0.025 inch square pin
All pins tin-plated



XE9624FS Mechanical Specifications

PIN	INCHES		METRIC(MM)	
	MIN	MAX	MIN	MAX
A	1.500	1.560	38.10	39.62
B	0.430	0.490	10.92	12.45
C	0.890	0.950	22.61	24.13
D	0.090	0.140	2.23	3.56
E	0.045	0.055	1.14	1.40
F	0.340	0.360	8.64	9.14
G	0.140	0.160	3.56	4.06
H	0.790	0.810	20.07	20.57
I	0.050	0.110	1.27	2.79
K	0.200	0.260	5.08	6.60
L	0.420	0.490	10.67	12.19

Pins = 0.018 inch diameter pin
All pins tin-plated



2/XECOM

XE9624F/XE9624FS

Pin Descriptions

NAME	I/O	DESCRIPTION
RESET	I	Hardware reset pin, Schmitt input, active HI, TTL. Use of an external reset is not required
AR	O	Auxiliary Data/Voice Relay output, active HI, TTL/CMOS. When high, AR drives an external auxiliary telephone set relay closed to allow the same telephone line to be used for voice communications.
RXD	O	Serial data output to the DTE (i.e. external UART). A logic "high" represents a "mark" and a logic "low" represents a "space", TTL.
!DTR	I	Data Terminal Ready, input, active LO, TTL. The function of this pin is set by the &D command and the value in register S21. Normally the modem ignores DTR.
!AA	O	Auto Answer enable indicator, output, active LO, TTL/CMOS. A low indicates the modem is set to automatically answer an incoming call.
!CTS	O	Clear to Send, output, active LO, TTL/CMOS.
OH	O	DAA hookswitch relay is closed in the "off-hook" position connecting the DAA to the telephone line.
TXD	I	Serial data input from the DTE (i.e. external UART). A logic "high" represents a "mark" and a low represents a "space", TTL.
!RTS	I	Request to Send, input, active LO, TTL. Not used.
!HS	O	High Speed indicator, output, active LO, TTL/CMOS. Low when operating at 2400bps rate, high otherwise.
!RI	O	Ring Indicator, output, active LO, TTL. When low indicates the modem is receiving a ring signal.
TIP	—	Tip connection to the phone line(RJ11 pin3) from the internal DAA.
RING	—	Ring connection to the phone line(RJ11 pin4) from the internal DAA. Caution: Observe design rules for Tip & Ring trace layout

NAME	I/O	DESCRIPTION
GND	—	Ground (0 volts).
AMP	O	Audio output to speaker. Function is determined by L & M commands and the value in register S22. This output can drive a 50Kohm load.
/DCD	O	Data Carrier Detect, output, active LO, TTL/CMOS. Function is set by the &C command and the value in register S21.
/DSR	O	Data Set Ready, output, active LO, TTL/CMOS. Function is set by the &S command and the value in register S21.
Vcc	—	+5 Volts

Pin Configurations

XE9624F Pin Configuration			
RESET	1	40	+5V
AR	2	39	/DSR
RXD	3	38	/DCD
	4	37	
	5	36	
	6	35	
	7	34	
	8	33	
/DTR	9	32	
/AA	10	31	
/CTS	11	30	
CH	12	29	
TXD	13	28	
/RTS	14	27	
/HS	16	26	
/RI	16	25	
	17	24	
TIP	18	23	
	19	22	AMP
RING	20	21	GND

XE9624FS Pin Configuration	
VCC	1
Reset	2
TXD	3
RXD	4
/AA	6
/RTS	8
/DTR	7
/RI	8
/HS	9
/DCD	10
/CTS	11
/DSR	12
CH	13
AR	14
AMP	15
Gnd	16
Tip	17
Ring	18

4/XECOM

XE9624F/XE9624FS

Datasheet del transistor TIP 147.



TIP145/146/147

Monolithic Construction With Built In Base-Emitter Shunt Resistors

- High DC Current Gain: $h_{FE} = 1000$ @ $V_{CE} = -4V, I_C = -5A$ (Min.)
- Industrial Use
- Complement to TIP140/141/142

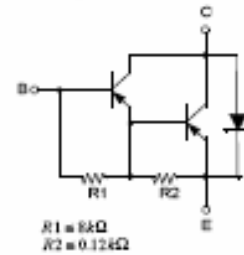


PNP Epitaxial Silicon Darlington Transistor

Absolute Maximum Ratings $T_C=25^\circ C$ unless otherwise noted

Symbol	Parameter	Value	Units
V_{CBO}	Collector-Base Voltage : TIP145	-60	V
	: TIP146	-80	V
	: TIP147	-100	V
V_{CEO}	Collector-Emitter Voltage : TIP145	-60	V
	: TIP146	-80	V
	: TIP147	-100	V
V_{EBO}	Emitter-Base Voltage	-5	V
I_C	Collector Current (DC)	-10	A
I_{CP}	Collector Current (Pulse)	-15	A
I_B	Base Current (DC)	-0.5	A
P_C	Collector Dissipation ($T_C=25^\circ C$)	125	W
T_J	Junction Temperature	150	$^\circ C$
T_{STG}	Storage Temperature	-65 ~ 150	$^\circ C$

Equivalent Circuit



Electrical Characteristics $T_C=25^\circ C$ unless otherwise noted

Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Units	
$V_{CE(sus)}$	Collector-Emitter Sustaining Voltage	$I_C = -30mA, I_B = 0$	-60			V	
			-80			V	
			-100			V	
I_{CBO}	Collector Cut-off Current	$V_{CE} = -30V, I_B = 0$			-2	mA	
			$V_{CE} = -40V, I_B = 0$			-2	mA
			$V_{CE} = -50V, I_B = 0$			-2	mA
I_{CBO}	Collector Cut-off Current	$V_{CB} = -60V, I_E = 0$			-1	mA	
			$V_{CB} = -80V, I_E = 0$			-1	mA
			$V_{CB} = -100V, I_E = 0$			-1	mA
I_{EBO}	Emitter Cut-off Current	$V_{BE} = -5V, I_C = 0$			-2	mA	
h_{FE}	DC Current Gain	$V_{CE} = -4V, I_C = -5A$	1000				
		$V_{CE} = -4V, I_C = -10A$	500				
$V_{CE(sat)}$	Collector-Emitter Saturation Voltage	$I_C = -5A, I_B = -10mA$			-2	V	
		$I_C = -10A, I_B = -40mA$			-3	V	
$V_{BE(sat)}$	Base-Emitter Saturation Voltage	$I_C = -10A, I_B = -40mA$			-3.5	V	
$V_{BE(on)}$	Base-Emitter ON Voltage	$V_{CE} = -4V, I_C = -10A$			-3	V	
t_D	Delay Time	$V_{CC} = -30V, I_C = -5A$ $I_{B1} = -20mA, I_{B2} = 20mA$ $R_L = 6\Omega$		0.15		μs	
t_R	Rise Time			0.55		μs	
t_{STG}	Storage Time			2.5		μs	
t_f	Fall Time			2.5		μs	

TIP145/146/147

Typical Characteristics

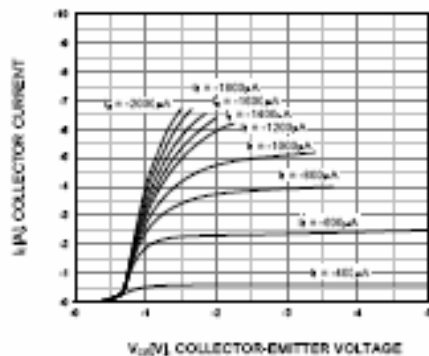


Figure 1. Static Characteristic

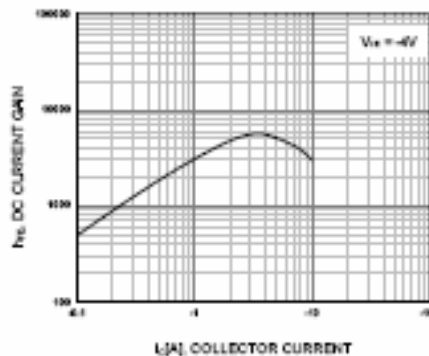


Figure 2. DC current Gain

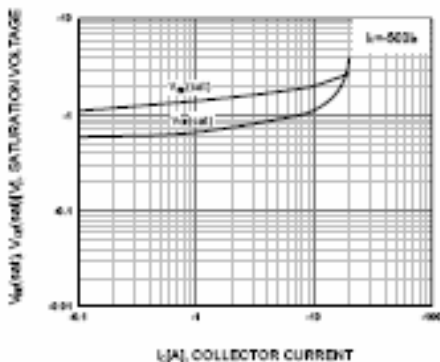


Figure 3. Collector-Emitter Saturation Voltage
Base-Emitter Saturation Voltage

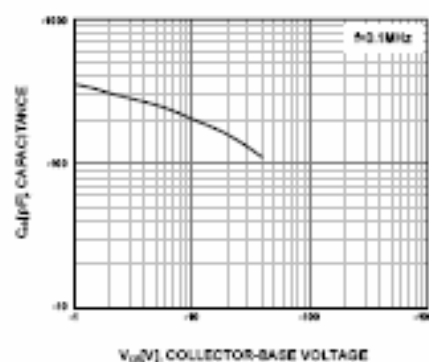


Figure 4. Collector Output Capacitance

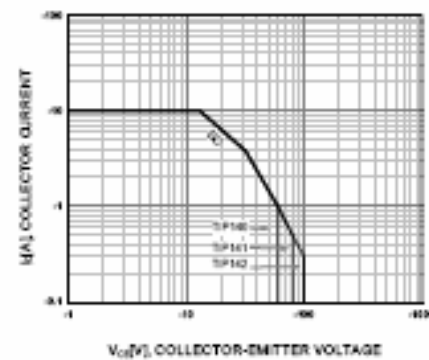


Figure 6. Safe Operating Area

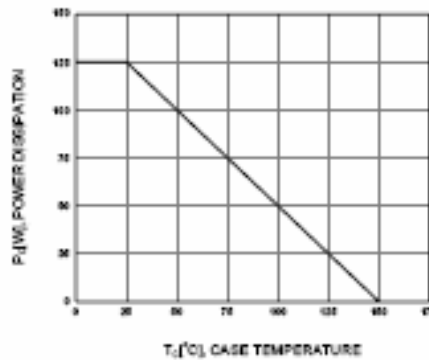
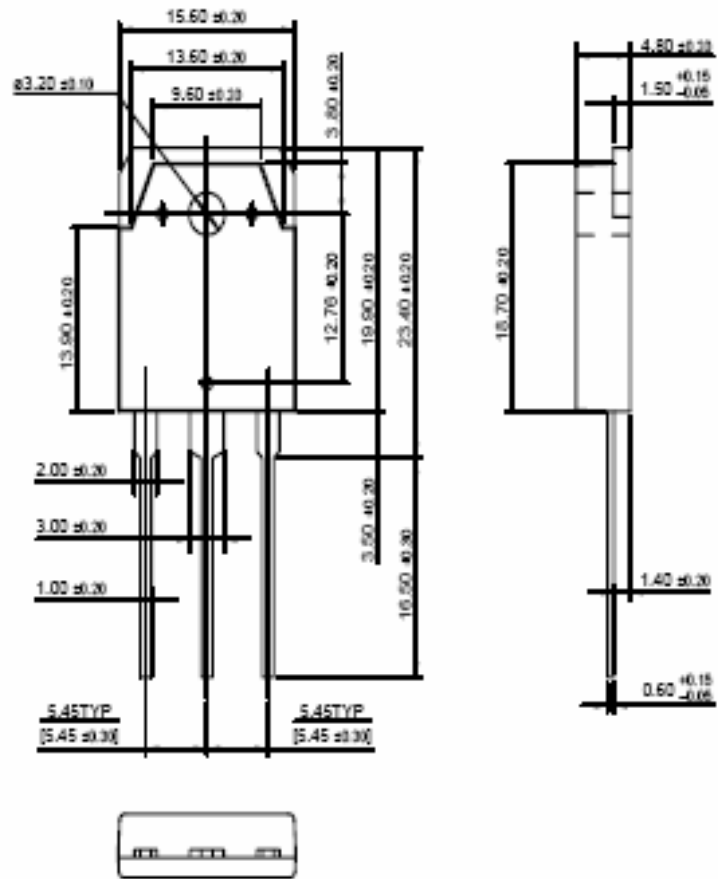


Figure 8. Power Derating

Package Dimensions

TO-3P



Dimensions in Millimeters

TRADEMARKS

The following are registered and unregistered trademarks Fairchild Semiconductor owns or is authorized to use and is not intended to be an exhaustive list of all such trademarks.

ACEx™	HiSeC™	SuperSOT™-8
Bottomless™	ISOPLANAR™	SyncFET™
CoolFET™	MICROWIRE™	TinyLogic™
CROSSVOLT™	POP™	UHC™
E ² CMOS™	PowerTrench®	VCX™
FACT™	QFET™	
FACT Quiet Series™	QS™	
FAST®	Quiet Series™	
FASTr™	SuperSOT™-3	
GTO™	SuperSOT™-8	

DISCLAIMER

FAIRCHILD SEMICONDUCTOR RESERVES THE RIGHT TO MAKE CHANGES WITHOUT FURTHER NOTICE TO ANY PRODUCTS HEREIN TO IMPROVE RELIABILITY, FUNCTION OR DESIGN. FAIRCHILD DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT OR CIRCUIT DESCRIBED HEREIN; NEITHER DOES IT CONVEY ANY LICENSE UNDER ITS PATENT RIGHTS, NOR THE RIGHTS OF OTHERS.

LIFE SUPPORT POLICY

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF FAIRCHILD SEMICONDUCTOR INTERNATIONAL.

As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, or (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

PRODUCT STATUS DEFINITIONS

Definition of Terms

Datasheet Identification	Product Status	Definition
Advance Information	Formative or In Design	This datasheet contains the design specifications for product development. Specifications may change in any manner without notice.
Preliminary	First Production	This datasheet contains preliminary data, and supplementary data will be published at a later date. Fairchild Semiconductor reserves the right to make changes at any time without notice in order to improve design.
No Identification Needed	Full Production	This datasheet contains final specifications. Fairchild Semiconductor reserves the right to make changes at any time without notice in order to improve design.
Obsolete	Not in Production	This datasheet contains specifications on a product that has been discontinued by Fairchild semiconductor. The datasheet is printed for reference information only.

Datasheet de los relevos HJR4102.

NINGBO TIANBO GANGLIAN ELECTRONICS CO., LTD

HJR4102

SPECIFICATION

FILE NUMBER: TB10S-T002A

DATE: 2004/06/25

■ CONTACT DATA

Contact Form	1A.1C
Contact Material	Silver Alloy
Contact Ratings	3A 120VAC/24VDC
Max Switching Voltage	240VAC/60VDC
Max Switching Current	5A
Max Switching Power	360VA/90W
Contact Resistance	100mΩ Max at 6VDC 1A
Life Expectancy Electrical	100,000 Operations (at 30 Operations/minute)
Mechanical	10,000,000 Operations

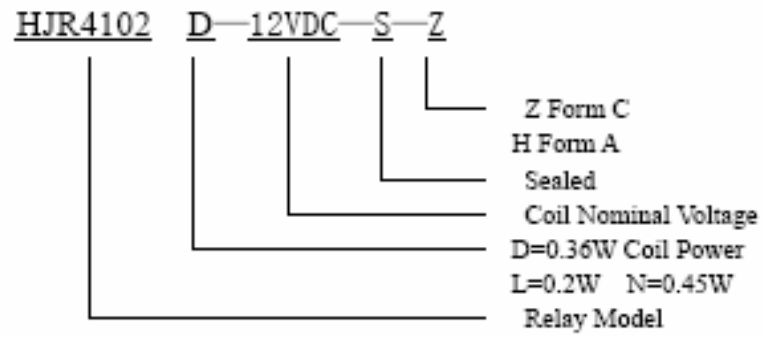
■ GENERAL DATA

Insulation Resistance	100MΩ Min at 500VDC	
Dielectric Strength Between Open Contacts	500VAC (for one minute)	
Between Contacts and coil	1000VAC (for one minute)	
Operate Time	5ms	
Release Time	5ms	
Temperature Range	-30℃ to +70℃	
Shock Resistance	Operating Extremes	10G
	Damage Limits	50G
Vibration Resistance	10-55Hz, 1.5mm	
Humidity	40-85%	
Weight	Approx. 3.5g	
Safety Standard	UL E173485	

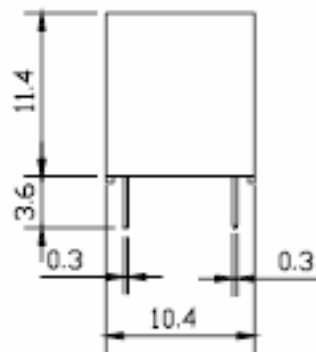
■ COIL DATA

Nominal Voltage (VDC)	Coil Resistance at 20℃ ±10% (Ω)			Max Operate Voltage VDC	Min Release Voltage VDC	Max Applicable Voltage VDC
	0.2W	0.36W	0.45W			
3	45	25	20	2.25	0.3	3.9
5	125	70	56	3.75	0.5	6.5
6	160	100	80	4.5	0.6	7.6
9	405	220	180	6.75	0.9	11.7
12	720	400	320	9	1.2	15.6
24	2800	1600	1280	18	2.4	31.2
48		5650±15%	4520±15%	36	4.8	62.4

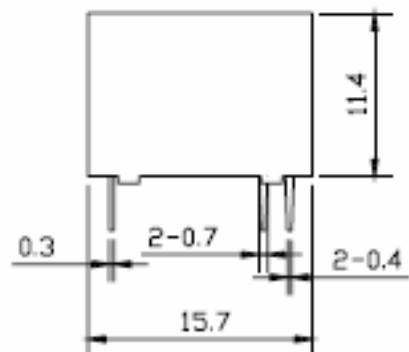
■ ORDERING CODE



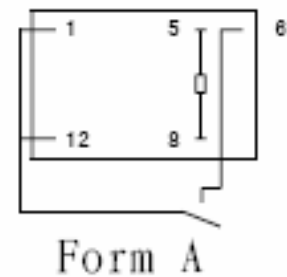
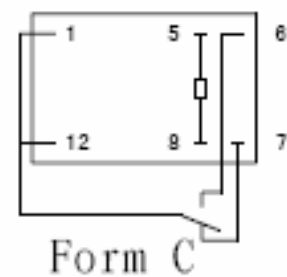
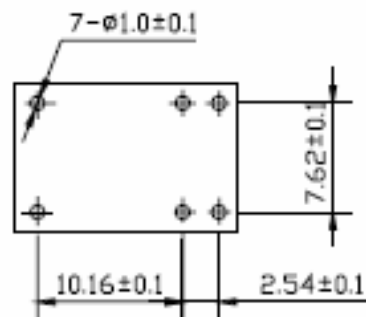
■ OVERALL AND MOUNTING DIMENSIONS



安装图

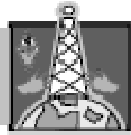


接线图



A continuación se presenta un documento con información sobre los núcleos toroidales.

NÚCLEOS TOROIDALES



1.- INTRODUCCIÓN

Para la construcción de bobinas fijas para su utilización en circuitos de baja potencia en las bandas de HF y VHF se utilizan casi exclusivamente núcleos toroidales. Los radioaficionados utilizan cada vez más este tipo de núcleo en lugar de las clásicas bobinas cilíndricas con núcleo ajustable que suelen ir encerradas en un blindaje metálico.

Los motivos por los cuales se prefieren las inductancias con núcleo toroidal a las clásicas bobinas cilíndricas son varias y les podemos resumir en los siguientes:

- Los núcleos toroidales, debido a su forma, no tienen campo magnético disperso y por tanto no es necesario encerrarlos en un blindaje metálico.
- El Q (factor de calidad) de un bobinado sobre un toroide es mayor que sobre una bobina convencional. Valores de 150 — 200 son fáciles de alcanzar, siempre que se utilice el núcleo adecuado. Esto es muy conveniente para la construcción de determinados circuitos, como por ejemplo, filtros pasabanda donde el factor de calidad de las bobinas empleadas influyen grandemente en su respuesta de frecuencia.
- El valor de la inductancia depende fuertemente del material empleado en el núcleo y del número de espiras, por lo que es muy fácil calcular el valor de la inductancia.
- Con los núcleos toroidales se consiguen inductancias muy

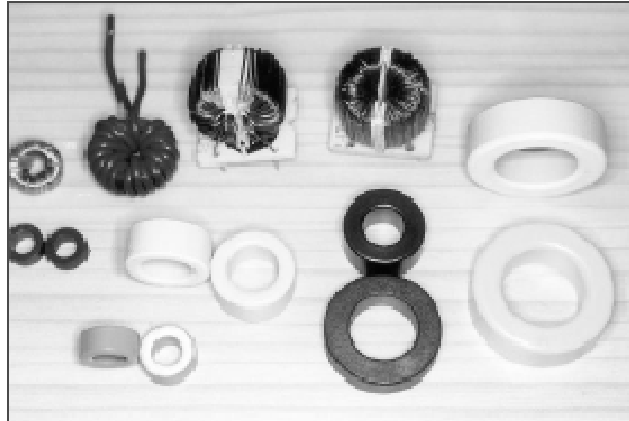


Fig. 1 Distintos tipos de bobinas

compactas y de reducido tamaño.

- Los núcleos toroidales son muy útiles desde algunos cientos de Hz hasta muy arriba en el espectro de UHF.

En la figura número 1 se pueden observar distintos tipos de toroides, algunos desnudos y otros bobinados con un cierto número de espiras. Como se puede observar claramente, los núcleos toroidales tienen forma de rosquilla. En la parte superior de dicha figura se aprecian dos unidades colocadas sobre un soporte para su mejor instalación sobre un circuito impreso. Así mismo se aprecia que el color de los toroides no es igual en las distintas unidades. El color indica el tipo de material utilizado en su fabricación y que tiene que ver, entre otras cosas, con el margen de frecuencias dentro del cual se puede utilizar el núcleo en cuestión.

2.- NÚMEROS

Los núcleos toroidales se identifican mediante la letra "T" seguida de dos otros números, que indican el diámetro exterior del núcleo en centésimas de pulgada, y otros cifras que indican el tipo de mezcla empleado en su fabricación. Así, un núcleo con la denominación T33/2 indica un núcleo con un diámetro exterior de 33 centésimas de pulgada, esto es, 8,5 milímetros, y una mezcla tipo 2 apropiada para frecuencias entre 1 y 30 MHz.

En la figura número 2 tenemos una vista frontal de un núcleo toroidal. En ella se indican el diámetro exterior y el diámetro interior. La sección AA se obtiene restando al diámetro exterior menos el diámetro interior y dividiendo por dos.

$$SE = \frac{DE - DI}{2}$$

SE = Sección horizontal
DE = Diámetro exterior
DI = Diámetro interior

En la figura número 3 tenemos una vista lateral del toroide, donde se indica la altura del núcleo. Con las anteriores dimen-

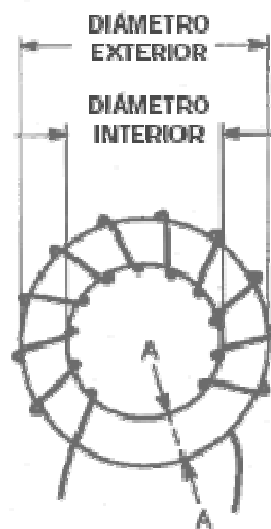


Fig. 2 Vista frontal

sienes, diámetro exterior, diámetro interior y altura, podemos calcular la longitud que tendrá cada espira. Esta longitud será la suma de dos veces la sección AA más dos veces la altura.

$$LO = DE - DI + 2AL$$

LO = Longitud de cada espira
AL = Altura del núcleo

Otro aspecto a tener en cuenta es el número máximo de espiras que podremos bobinar en una sola capa sobre un cierto núcleo. Este número vendrá determinado por diámetro interior y por el grueso del hilo que utilizamos y lo podemos calcular con la siguiente fórmula.

$$NE = \frac{3,14DI}{DH}$$

NE = Número de espiras
DI = Diámetro interior
DH = Diámetro del hilo

En la figura número 4 se observa la forma exacta de contar las espiras sobre el núcleo y en la figura número 5 tenemos la disposición más conveniente del bobinado sobre el toroide, esto es, dejando un espacio entre el comienzo y el final del bobinado aproximadamente de unos 30 grados. De esta forma se disminuye la capacidad parásita y se obtiene el mejor factor de calidad (Q) posible.

En la figura número 6 se indica el modo de bobinar un toroide. Para ello haremos una lanzadera a partir de un trozo

ALTURA DEL NÚCLEO

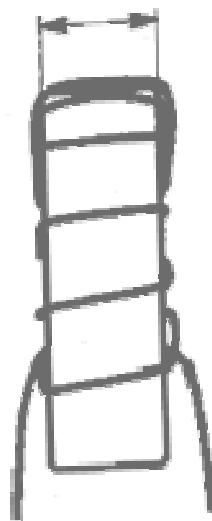


Fig. 3 Medida lateral

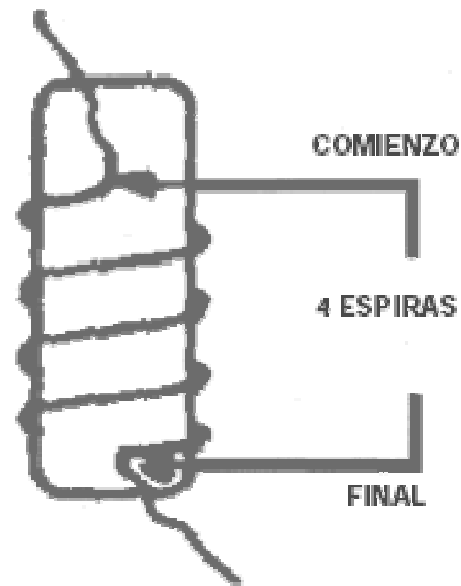


Fig. 4 Comienzo y fin del bobinado

de madera o cartón fuerte. El ancho de esta lanzadera deberá ser el adecuado para que pase por el centro del núcleo. En sus extremos haremos unas rendijas sobre las que bobinaremos previamente el hilo de cobre que formará el bobinado. La cantidad de hilo necesaria vendrá dada por las dimensiones del núcleo y el número de espiras a bobinar, y se puede calcular con las fórmulas dadas anteriormente.

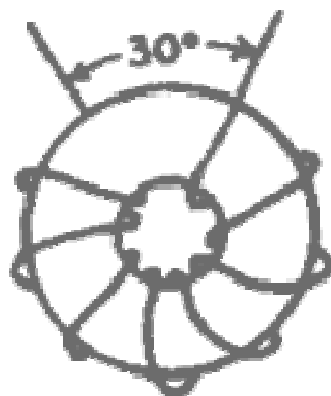


Fig. 5 Disposición del bobinado

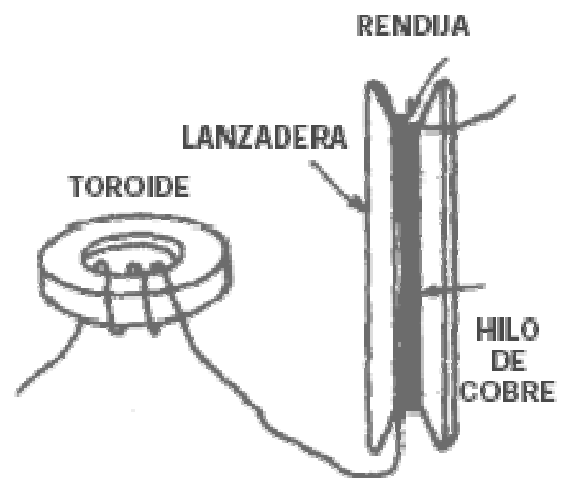


Fig. 6 Bobinado de un toroide

3.- DATOS

Los datos sobre núcleos toroidales que aparecen en distintas publicaciones a veces son algo confusos. En las siguientes líneas he intentado recopilar el mayor número de datos sobre esta cuestión, ordenándoles para una mejor comprensión por parte del lector.

En la tabla número 1 se indica el tipo de toroide, el diámetro exterior, diámetro interior, altura y desarrollo, es decir, la longitud de cada espira.

TABLA NÚMERO UNO				
TIPO	DIÁMETRO EXTERIOR	DIÁMETRO INTERIOR	ALTURA	DESARROLLO
	mm	mm	mm	mm
T-12	3,1	1,5	1,3	4,2
T-16	4,1	1,8	1,5	5,3
T-20	5,1	2,0	1,8	6,7
T-25	6,4	3,0	2,5	8,4
T-27	6,9	3,4	3,1	9,7
T-30	7,6	3,8	3,3	10,4
T-37	9,4	5,1	3,3	10,9
T-44	11,2	5,8	4,0	13,4
T-50	12,7	7,6	4,8	14,7
T-60	15,2	8,4	4,8	16,4
T-68	17,3	9,4	4,8	17,5
T-80	20,3	12,4	6,4	20,7
T-94	23,9	14,2	7,9	25,5
T-105	26,0	14,2	11,2	35,1
T-130	33,0	19,0	11,2	36,4
T-157	39,9	24,1	14,5	44,8
T-184	46,7	24,4	18,0	58,3
T-200	50,8	31,7	14,0	47,1
T-200A	50,8	31,7	25,4	69,0
T-225	57,1	35,6	14,0	49,5
T-225A	57,1	35,6	25,4	72,3
T-300	76,2	48,8	12,7	52,8
T-300A	76,2	48,8	25,4	78,2
T-400	101,6	57,1	16,5	77,5
T-400A	101,6	57,1	33,0	110,5
T-520	132,1	78,2	20,3	94,5

Todos los datos están expresados en milímetros. Estas cifras son aproximadas debido a la conversión de pulgadas a milímetros. Así mismo las dimensiones de los núcleos pueden tener ciertas tolerancias, por lo que sus dimensiones reales pueden diferir ligeramente de las indicadas.

Señalar también que hoy en el mercado núcleos toroidales que no tienen las dimensiones indicadas en la tabla número 1, bien en su totalidad o parcialmente. Por ejemplo, el toroide de color gris que se encuentra encima de los dos pequeños de color rojo, en la parte izquierda de la figura número 7, tiene un diámetro exterior de 20,3 milímetros y un diámetro interior de 12,4 milímetros, por lo que parece que debería corresponder al tipo T-60. Sin embargo su altura es de 9,4 milímetros y no de 6,4 milímetros como indica la tabla para la altura del toroide tipo T-60.

Como ya se ha dicho anteriormente, el color o colores del toroide indican el tipo de material empleado en su construc-

ción y por tanto el margen de frecuencias dentro del cual debe ser utilizada preferentemente un determinado núcleo. En la tabla número 2 se seleccionan estos colores y sus frecuencias asociadas.

TABLA NÚMERO DOS		
COLOR(ES)	NÚMERO	FRECUENCIAS
Amarillo-Blanco	26	Hasta 100KHz
Gris	3	50 KHz - 1 MHz
Rojo-Blanco	15	0,1 - 3 MHz
Azul	1	0,5-5 MHz
Rojo	2	1 - 30 MHz
Amarillo	6	2 - 50 MHz
Negro	10	5 - 100 MHz
Azul-Amarillo	12	20 - 200 MHz
Verde-Blanco	17	20-200 MHz
Verde-Naranja	22	20-200 MHz
Marrón	0	50-300 MHz

En la primera columna se relacionan los colores del núcleo, en primer lugar el color que ocupa tres de las cuatro caras y en segundo lugar el que ocupa la cuarta cara, si lo hay. La segunda columna indica el número normalizado que define el material del núcleo. En la tercera columna se indica el margen de frecuencias donde preferentemente se debe utilizar el núcleo.

Los datos relativos a las frecuencias de utilización varían entre las distintas publicaciones consultadas. No obstante es posible utilizar un núcleo en un margen de frecuencias distinto del especificado, aunque con un factor de calidad menor.

Otro dato a tener en cuenta es el índice AL, que es la inductancia que se obtiene, en milihenrios, al bobinar sobre un núcleo n-espiras. Este índice varía con el tamaño del núcleo y con el tipo de mezcla. En la tabla número tres se relacionan los distintos núcleos y su índice AL.

En la fila superior se indica el tipo de mezcla, mientras que en la columna de la izquierda aparece el tamaño del núcleo. En los núcleos muy pequeños no se tiene el dato ya que no se pueden bobinar cien espiras sobre un núcleo tan pequeño.

Conociendo el parámetro AL de un determinado núcleo, es posible calcular el número de espiras que es preciso bobinar para obtener una determinada inductancia. Para ello utilizaremos la siguiente fórmula:

$$N = 100 \sqrt{\frac{\text{INDUCTANCIA}}{A_L}}$$

TABLA NÚMERO TRES

	26	3	15	1	2	6	10	17	0
T-12		60	50	8	20	17	12	7	3
T-16		61	5	44	22	19	13	0	3
T-20		90	6	52	27	22	16	10	3
T-25		100	100	70	34	27	19	12	5
T-27									
T-30	325	140	93	85	43	36	25	16	6
T-37	275	120	90	80	40	30	25	15	5
T-44	360	180	160	105	52	42	33	19	7
T-50	320	175	135	100	49	40	31	18	6
T-60									
T-68	420	195	180	115	57	47	32	21	8
T-80	450	190	170	115	55	45	32	22	9
T-94	590	248	200	160	84	70	58	32	11
T-106	900	450	345	325	135	116			19
T-130	785	350	250	200	110	96			15
T-157	970	420	330	320	140	115			
T-184	1640	720		500	240				
T-200	895	425		250	120	100			
T-200A	1550				218	180			
T-225	950	424			120	100			
T-225A	1600				215				
T-300	825				115				
T-300A	1600				228				
T-400	1320				185				
T-400A	2600				360				
T-520	1460				20				

La inductancia de un determinado bobinado se puede calcular utilizando la siguiente fórmula:

$$L(\text{mH}) = \frac{A_L \cdot N^2}{10000}$$

El parametro AL de un determinado núcleo se calcula mediante la siguiente fórmula:

$$A_L = \frac{10000 \cdot L}{N^2}$$

4.- CONCLUSIÓN

En las líneas anteriores se ha intentado recopilar los datos más significativos que permitan la colección de bobinados sobre núcleos toroidales. Estos datos no pretenden ser exhaustivos ni totalmente exactos, ya que se han encontrado discrepancias entre las diversas publicaciones consultadas. No obstante pueden ser un buen punto de partida para obtener buenos resultados.

Como es natural, quedo enteramente a la disposición de quien necesite cualquier aclaración o detalle sobre lo aquí tratado. Buena suerte a todos.

Luis Sanchez Perez, EA4-NH
 Aparado 421, TOLEDO
 Tlf. 925-35-34-66
 Email : lusape@ndbener.es

