

DISEÑO Y CONSTRUCCION DE UN SISTEMA RF. DE TRANSMISION DE
MENSAJES

GERARDO MEJÍA MEJÍA

CORPORACION UNIVERSITARIA TECNOLOGICA DE BOLIVAR

FACULTAD DE INGENIERIA ELECTRICA

CARTAGENA DE INDIAS D.T.C.

1998

DISEÑO Y CONSTRUCCION DE UN SISTEMA RF. DE TRANSMISION DE
MENSAJES

GERARDO MEJIA MEJIA

Trabajo de grado presentado como requisito
para optar el titulo de Ingeniero Electricista

Director
OSCAR SANTOS
Ingeniero Electrónico

CORPORACION UNIVERSITARIA TECNOLOGICA DE BOLIVAR
FACULTAD DE INGENIERIA ELÉCTRICA Y ELECTRÓNICA
CARTAGENA DE INDIAS D.T.C.

1998

Cartagena, enero 28 de 1998

Señores:

COMITE DE PROYECTOS DE GRADO

Facultad de Ingeniería Eléctrica

Corporación Universitaria Tecnológica de Bolívar

Ciudad.

Apreciados señores:

Atentamente me permito presentar el proyecto de grado titulado DISEÑO Y CONSTRUCCION DE UN SISTEMA RF. DE TRANSMISION DE MENSAJES, como requisito parcial para optar al titulo de Ingeniero Electricista.

Cordialmente,

GERARDO MEJIA MEJIA

Cartagena, enero 28 de 1998

Señores:

COMITÉ DE PROYECTOS DE GRADO

Facultad de Ingeniería Eléctrica y Electrónica

Corporación Universitaria Tecnológica de Bolívar.

Ciudad.

Apreciados señores:

Como director de tesis de grado titulada DISEÑO Y CONSTRUCCION DE UN SISTEMA DE RF. TRANSMISION DE MENSAJES hago presentación formal de dicha tesis, la cual fue revisada en su totalidad, por lo tanto espero que sea de su total agrado.

Cordialmente,

OSCAR SANTOS BELTRAN

Ingeniero Electrónica.

La Corporación Universitaria Tecnológica de Bolívar, se reserva el derecho de propiedad intelectual de todos los trabajos de grados aprobados, y no pueden ser explotados comercialmente sin su autorización.

AGRADECIMIENTOS

El autor expresa sus agradecimientos a:

Oscar Santos Beltran. Ingeniero Electrónico y Director del proyecto, por su motivación y apoyo.

Billy Kay Paternina. Técnico en Comunicaciones, por su gran colaboración en el área de comunicaciones.

Rafael Medina Sánchez. Estudiante de Ingeniería Electrónica, por su gran ayuda y colaboración en forma incondicional.

A mis compañeros de estudio y trabajo por estar siempre prestos a cualquier colaboración.

A ECOPETROL - USO por sus beneficios en educación.

DEDICATORIA

A mi querida esposa Alíx María Toscano G.

A mis hijos Gerardo David, Angélica y Daniel

A mi querida madre Otilia Mejía

A mis hermanos

A la memoria de mi padre Gerardo Mejía C.

A Dios

GERARDO MEJIA MEJIA

Nota de aceptación

Presidente del Jurado

Jurado

Jurado

Cartagena, 26 de Enero de 1998

CONTENIDO

	pág.
INTRODUCCION	
1. CONCEPTOS BASICOS	4
1.1 ELEMENTOS DE UNA COMUNICACIÓN	4
1.2 COMPONENTES DE UN SISTEMA DE COMUNICACIONES	6
1.2.1 Elementos DTE	6
1.2.2 Elementos DCE	7
1.2.3 Canal	7
1.3 TECNICAS DE TRANSMISION	7
1.3.1 Transmisión paralelo	7
1.3.2 Transmisión serie	8
1.3.2.1 Formas de transmisión en serie	9
1.4 CODIFICACION	11
1.4.1 Tipos de códigos	11
1.5 EL MODELO OSI	12
1.5.1 Nivel físico	12
1.5.2 Nivel de enlace	13

1.5.3 Nivel de red	13
1.5.4 Nivel de transporte	14
1.5.5 Nivel de sesión	14
1.5.6 Capa de presentación	14
1.5.7 Capa de aplicación	15
1.6 LA COMPUTADORA	15
1.7 CONCEPTO DE PROGRAMA	16
1.7.1 Lenguajes de programación	20
1.8 LA NORMA EIA-232-C	25
2. EL MODEM	29
2.1 INTRODUCCION	29
2.2 DESCRIPCION DEL PACKET RADIO	30
2.3 PROTOCOLOS	33
2.4 CONTROLES E INDICADORES	33
2.5 INSTALACION	34
2.6 SELECTORES INTERNOS	39
2.7 CHEQUEOS INICIALES	42
2.8 OPERACION INICIAL	43
2.9 CONECTANDO Y DESCONECTANDO	44
3. EL TRANSCEPTOR	46
3.1 INTRODUCCIÓN	46
3.2 TRANSMISION EN VHF	46
3.3 MODULACIÓN.	47

3.4 MODULACION DE FRECUENCIA	50
3.5 EL TRANSMISOR DE F.M.	52
3.6 RECEPTORES DE F.M.	54
4. TARJETA MICROCONTROLADORA MIC51	59
4.1 INTRODUCCION	59
4.2 HARDWARE	60
4.2.1 Descripción general del sistema MIC51.	61
4.2.1.1 Microcontrolador 80C51	61
4.2.1.2 Lacth 74LS373	64
4.2.1.3 Memoria EPROM 27C64	64
4.2.1.4 Memoria RAM 6264	65
4.2.1.5 Transmisor/Receptor ICL232	66
4.2.1.6 B1, Interface (Puerto paralelo)	66
4.2.1.7 CN1, Interface DB9 (Puerto serie).	67
4.2.1.8 Frecuencia de operación	67
4.2.1.9 Batería	68
4.2.1.10 JP1, Jumper	68
4.2.1.11 RST	68
4.3.2 Descripción del funcionamiento de la tarjeta MIC51	68
4.3.2.1 Arranque del sistema	69
4.3.2.2 Direccionamiento de memoria externa	70
4.3.2.3 Lectura de memoria de programa	72
4.3.2.4 Acceso a la memoria de datos	73

4.3.2.5	Transmisión/recepción serial	73
4.4	SOFTWARE	74
5.	EL DISPLAY	89
5.1	INTRODUCCION	89
5.2	TARJETA DEL DISPLAY	89
5.2.1	Bloque 1. Integrados IC1,IC2, IC3	90
5.2.2	Bloque 2. Integrados IC4,IC5,IC6	92
5.2.3	Bloque 3. Integrado IC7	93
5.2.4	Bloque 4. Etapa de potencia para las columnas	94
5.2.5	Bloque 5. Etapa de potencia para las filas	94
5.2.6	Bloque 6. Pantalla (display)	95
5.3	COMPONENTES DE LA TARJETA DEL DISPLAY	98
5.4	CONSTRUCCION DE UN MENSAJE EN EL DISPLAY	98
5.5	DESPLIEGUE DEL MENSAJE EN EL DISPLAY	99
6.	OPERACIÓN Y DESCRIPCION DEL SISTEMA	105
6.1	INTRODUCCION	105
6.2	MODULOS QUE CONFORMAN EL SISTEMA	105
6.2.1	Módulo 1 o transmisor digital	105
6.2.1.1	Fuente de alimentación	108
6.2.1.1.1	Transformador	109
6.2.1.1.2	Tarjeta rectificadora	110
6.2.1.2	Tarjeta Módem	112
6.2.1.3	Tarjeta del Transceptor	115

6.2.2	Módulo 2 o receptor digital	118
6.2.3	Panel del display	121
7.	SOFTWARE UTILIZADO PARA MANEJAR EL DISPLAY	123
8.	CONCLUSIONES	151
	BIBLIOGRAFIA	153
	ANEXOS	155

LISTA DE TABLAS

	Pág.
Tabla 1. Códigos de bits	11
Tabla 2. Identificación de DB 25 para el módem	38
Tabla 3. Valores de R-32 Vs nivel de ASFK	40

LISTA DE CUADROS

	Pág.
Cuadro 1. Campos de una instrucciones en ensamblador	24

LISTA DE FIGURAS

	Pág.
Figura 1. Elementos de la comunicación.	4
Figura 2. Componentes de una comunicación.	6
Figura 3. Transmisión de datos en paralelo.	8
Figura 4. Transmisión de datos en serie.	9
Figura 5. Formato de datos asíncrono.	10
Figura 6. Formato de datos síncrono.	11
Figura 7. Capas del modelo OSI.	13
Figura 8. Conversor de lenguaje ensamblador a lenguaje de maquina.	21

Figura 9. Conector RS-232.	27
Figura 10. Panel posterior del módem.	35
Figura 11. Conector para radio.	35
Figura 12. Conexión del módem al radio.	37
Figura 13. Conexión del módem al PC.	37
Figura 14. Modulación en A.M.	48
Figura 15. Modulación en F.M.	49
Figura 16. Transmisor típico de F.M.	53
Figura 17. Receptor típico de F.M.	54
Figura 18. Mezclador.	66
Figura 19. Mapa de memoria EPROM.	65
Figura 20. Mapa de memoria de la RAM.	66
Figura 21. Circuito de restablecimiento del microcontrolador.	69
Figura 22. Diagrama de tiempo para acceso a memoria de programa.	71
Figura 23. Diagrama de tiempo acceso a memoria externa.	72
Figura 24. Diagrama de tiempo para comunicación serial.	74
Figura 25. Vista posterior de las tarjetas del display.	90
Figura 26. Diagrama de bloques del display.	91
Figura 27. Punto de la matriz de led's.	95
Figura 28. Identificación de pines de la matriz de diodos.	96
Figura 29. Representación en cátodo común de la matriz de display.	96
Figura 30. conexión de la tarjeta de display.	97

Figura 31. Tablero del display.	101
Figura 32. Visualización de mensajes.	101
Figura 33. Mapa de vídeo de la zona de vídeo.	101
Figura 34. Información para visualizar mensajes.	101
Figura 35. Diagrama de tiempo.	104
Figura 36. Vista del modulo 1.	106
Figura 37. Panel posterior del modulo 1.	108
Figura 38. Partes internas del modulo 1.	109
Figura 39. Transformador de la fuente.	110
Figura 40. Tarjeta rectificadora.	111
Figura 41. Fuente de 12 VDC.	112
Figura 42. Ensamble de la tarjeta módem.	113
Figura 43. Conexión del módem al radio	114
Figura 44. Conexión del módem a la MIC51.	115
Figura 45. Ensamble del transceptor.	116
Figura 46. Panel frontal del transceptor.	117
Figura 47. Vista general del modulo 2.	120
Figura 48. Vista general del modulo 3.	122

LISTA DE ANEXOS

	Pág.
Anexo A. Manual de operaciones del módem KPC-2400.	154
Anexo B. Manual del transceptor IC- 2AT	191
Anexo C. Identificación de pines del microcontrolador 80C51.	208
Anexo D. Especificaciones de la memoria EPROM 27C64	209
Anexo E. Especificaciones de la memoria RAM 6264.	215
Anexo F. Especificaciones del integrado 74LS164.	221
Anexo G. Especificaciones del integrado 74LS373.	224
Anexo H. Especificaciones del integrado 74LS244.	229
Anexo J. Instrucciones en lenguaje mnemónico para el microcontrolador 80C51.	233
Anexo K. Tabla de códigos ASCII.	237
Anexo L. Manual de operación del sistema de transmisión de mensajes.	138
Anexo M. Diagrama a bloques del 80C51.	249
Anexo N. Diagrama a bloques del transceptor IC- 2AT.	250
Anexo Ñ. Diagrama esquemático del transceptor 2AT.	251
Anexo P. Diagrama de la tarjeta del display.	253
Anexo Q. Diagrama de la tarjeta microcontroladora.	256

GLOSARIO

ASK: Manipulación de la señal por corrimiento de amplitud

CCITT: Comité Consultivo para Telefonía y Telegrafía Internacional

DCE: Data Communication Equipment. Dispositivo manipulador o convertidor de la señal.

DTE: Data Terminal Equipment. Dispositivo digital que transmite o recibe la señal.

Full Dúplex (FDX): Dos transmisores transmiten y reciben en ambas direcciones simultáneamente.

Full/Full Dúplex (F/FDX): La transmisión es posible en ambas direcciones al mismo tiempo, pero no entre los mismos dos transmisores. Se requiere de más de dos transmisores.

Hall Dúplex (HDX): La transmisión de datos en ambas direcciones pero no al mismo tiempo.

ISO: Organización de Estándares Internacionales.

Módem: Modulador/Demodulador. Convierte las señales digitales binarias a señales analógicas tal como FSK, PSK, QAM y viceversa.

OSI: Open System Interconexión. Interconexión de sistemas abiertos.

QAM: Modulación de amplitud en cuadratura.

Simplex: Con la operación Simplex, la transmisión de datos no se puede dirigir, la información se puede enviar solo en una dirección.

UART: Transmisor Receptor Universal Asíncrono.

RESUMEN

El objetivo de este trabajo es diseñar un sistema para enviar mensajes desde una computadora a través de ondas de radio y visualizarlos en una matriz de diodos portátil.

En el capítulo 1 se estudiarán algunos conceptos básicos necesarios para conocer como el usuario por medio del teclado puede enviar mensajes desde un PC via radio-frecuencia, lógicamente soportado con el software que es el programa que logra ejecutar las instrucciones.

En el capítulo 2 se tratará el estudio del módem utilizado y sus especificaciones requeridas.

En el capítulo 3 se estudiará el radio transmisor-receptor llamado transceptor con sus especificaciones y sus diferentes protocolos de comunicación.

El capítulo 4 trata el estudio referente al microcontrolador basado en el 8051 y las diferentes memorias utilizadas para la ejecución y almacenamiento de los mensajes.

El capítulo 5 trata el diseño construcción y funcionamiento del Display o matriz de diodos que es la terminal donde se visualiza el mensaje enviado desde el teclado.

En el capítulo 6 se detallan las partes físicas del sistema como son los diferentes módulos utilizados y específicamente el montaje de todos sus dispositivos.

En el capítulo 7 se detalla el software utilizado por el microcontrolador para manipular y almacenar los datos enviados desde el PC.

Se anexan los diferentes planos e información necesaria para el buen manejo y operación del sistema incluyendo el manual de operación del sistema en el anexo L.

Este trabajo es una de las aplicaciones que se pueden dar a tan importante forma de transmisión de datos ya que sus principales ventajas son la transmisión y recepción de información en forma inmediata y la facilidad de recibir la información en una periferia determinada, es decir, se puede trasladar la terminal que visualiza a cualquier punto en un radio aproximado de 500 metros a la redonda.

Esta limitación de la distancia la da la potencia en antena de los transceptores.

Todas estas ventajas se logran gracias a que la comunicación no necesita ningún medio físico (que por lo general son las que generan pérdidas) para enviar los datos.

Para ver la necesidad de este sistema se van a mencionar algunas aplicaciones tales como la supervisión remota de plantas de producción, control remoto de equipos, telemetría, telemática, transmisión de mensajes y los famosos buscapersonas.

Estas son las bases para que futuros estudiantes complementen con otros trabajos similares. También se le puede dar mayor utilidad a este sistema como ejemplo sugiero adaptarle el sistema de medición de temperatura ambiente y la visualización de la hora.

INTRODUCCIÓN

Una de las necesidades más prioritarias durante la historia, desarrollo y evolución del hombre ha sido y será la forma de comunicarse de una manera cada vez más eficiente, confiable e inmediata con sus semejantes. Por éstas razones el hombre ha estado a la vanguardia con sus grandes avances tecnológicos en esta área. Es por eso que hoy vemos la facilidad de comunicarnos y estar informados desde cualquier parte del planeta de una forma rápida, no es para menos ya que tenemos grandes herramientas a las cuales cualquier usuario puede tener disposición. Como ejemplo tenemos *las video conferencias, televisión, Internet y las comunicaciones satelitales*.

Todo esto gracias al descubrimiento y aprovechamiento del espacio libre para la transmisión digital de datos por ondas de radio (R.F.), cuya información viaja a través del espectro electromagnético.

Los sistemas de transmisión de datos por radio-frecuencia han revolucionado las comunicaciones ya que permiten a los usuarios el acceso remoto a bancos de información, así como también la comunicación entre varios usuarios separados por distancias relativamente grande. Estas comunicaciones también se pueden realizar por vía telefónica, par aislado (línea dedicada) pero no tienen una de las principales

ventajas que utiliza las ondas de radio frecuencia cuya propagación no necesita ningún medio físico. Una aplicación de esta transmisión por ondas de radio es la que se va a tratar en este caso donde se van a transmitir mensajes desde la computadora y se van a visualizar en una matriz de diodos.

En un sistema de transmisión de datos generalmente los grandes usuarios tienen centralizada su información en salas de cómputo, sin embargo, tienen la necesidad de tener disponible dicha información actualizada en varios puntos móviles, lo que requiere la existencia de un sistema para comunicar estos puntos transmitiendo y recibiendo dicha información. Si este sistema no es por ondas de radio se tendría que tener la información repetida en todos los sitios lo que resulta poco práctico, engorrosa e inoportuno, además de no ser muy rentable cuando de distancias grandes se trata.

El propósito de este trabajo es dar a conocer los mecanismos utilizados para el diseño y construcción de un sistema para transmitir mensajes basado principalmente en la transmisión de datos por ondas de alta frecuencia, específicamente en la banda para aficionados de dos metros con frecuencia de 144 Mhz. Estos se envían desde un PC y se visualizan en una matriz de diodos.

Esto fue posible gracias a los dispositivos y materias que de una u otra forma se involucraron y que al trabajarlos en conjunto cumplieron el objetivo trazado.

Es como de esta forma se integraron varias áreas de la electrónica como las computadoras, software, radios de comunicación en VHF, módems, microcontroladores y electrónica digital que al unirse forman el sistema titulado DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA RF DE TRANSMISIÓN DE MENSAJES y cuyo desarrollado se expondrá a través de estos siete capítulos.

1 CONCEPTOS BASICOS

1.1 ELEMENTOS DE UNA COMUNICACIÓN.

Los principales elementos que intervienen en una comunicación son:

- Transmisor
- Medio
- Receptor
- Convertidor

En la Figura 1. se observan los elementos que intervienen en un sistema de comunicación, la transformación de las señales o datos a través de ellos y el medio de transporte.

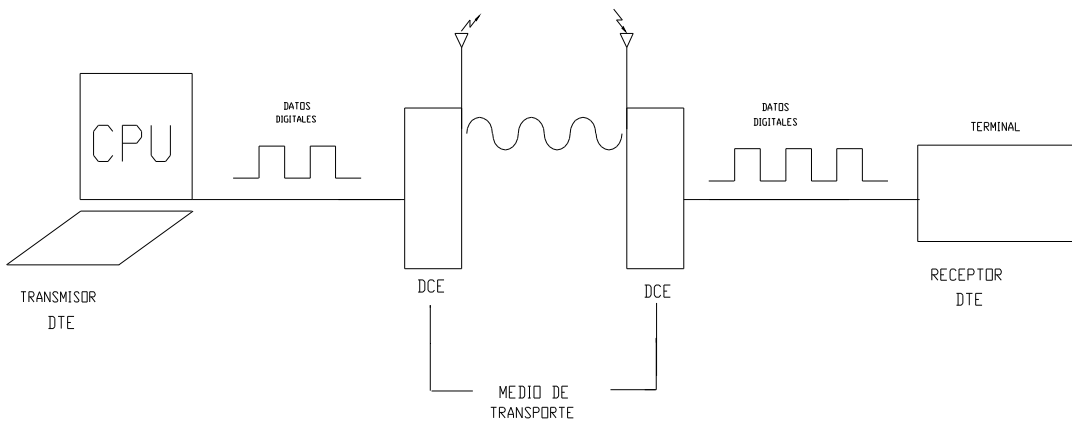


Figura 1. Elementos de una comunicación.

DTE: - *Data Terminal Equipment*: Es el dispositivo digital que transmite y/o recibe la información. Para el gráfico de la figura 1. se observa que la transmisión se hace entre la *computadora y una terminal* cuyos dispositivos DTE son el PC que envía los datos y el display (terminal) que es el que los recibe y los visualiza.

Si la comunicación se realiza de *computadora a computadora* en forma bidireccional los dispositivos DTE son las dos computadoras.

Otra forma de poder transmitir información es entre una *terminal y otra terminal*.

DCE: - *Data communication Equipment*. Es el dispositivo manipulador o convertidor de la señal por donde va la información. Para la figura 1. estos dispositivos son los módem que se encargan de convertir la señal digital a análoga y de análoga a digital.

Los transceptores son los dispositivos encargados de transmitir y/o recibir la información análoga por medio de ondas de radio de alta frecuencia (R.F.) cuya señal se designa como VHF (Very High Frecuence). Este es el medio utilizado en este trabajo para transportar la información.

Hay otros medios utilizados para el transporte de información y es por medio de cable, hilos telefónicos, etc.

El transporte por hilos telefónicos se presenta cuando se envía información desde un PC a través de la línea telefónica a otro PC; para este ejemplo los dispositivos DCE son los módem que se encargan de convertir la señal.

1.2 COMPONENTES DE UN SISTEMA DE COMUNICACIONES

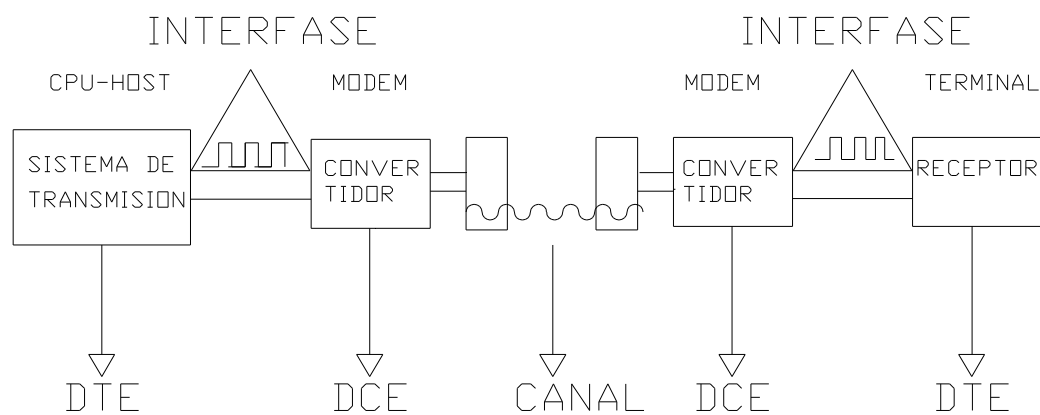


Figura 2. Componentes de un sistema de comunicación.

1.2.1 Elementos DTE: Generalmente se describen los equipos DTE como los usados en las estaciones para adaptar la señales digitales de la computadoras y terminales a una forma más adecuada para transmitir. Como ejemplo tenemos:

- Aparatos terminales: impresoras, cintas, lectoras, teclados
- Procesos industriales: algunas variables como presión, nivel, temperatura
- Computadoras
- Dispositivos especiales: display's (matrices de diodos) como la utilizada en este

trabajo, lápiz de luz, tonos telefónicos, terminales de vídeo, Fax, voz.

1.2.2 Elementos DCE: Generalmente son los equipos que convierten las señales digitales a señales analógicas como también los que hacen la interface del equipo terminal de datos al medio de transmisión analógico. Esencialmente un DCE es un módem (modulador/demodulador) que convierte las señales digitales binarias a señales analógicas tales como FSK, PSK, QAM.

1.2.3 Canal: Es el medio de transporte de la información utilizado para comunicar los dispositivos DCE. Entre los medios de transporte más comunes tenemos:

- Medios físicos: cables, hilos telefónicos, fibra óptica, superficies acústicas, guías de ondas.
- Medios electromagnéticos: señales radiadas por antenas en forma de energía electromagnética que viajan en el espacio libre. Como ejemplo tenemos la transmisión de datos de alta frecuencia VHF y las señales en microondas.

1.3 TECNICAS DE TRANSMISION

La información binaria se puede transmitir paralelamente o en serie:

1.3.1 Transmisión paralelo: Llamada también *paralela por bit*. En la figura 3. se muestra como el código binario se transmite simultáneamente desde T hasta R.

Las principales características utilizadas en esta forma de transmisión son:

- Los bits de datos se transmiten todos al tiempo por lo tanto tiene mayor velocidad.
- Emplea varias líneas de comunicación.
- Se utiliza para comunicaciones a corta distancia
- Se utiliza dentro de una computadora.

En la Figura 3. se observa que para enviar un dato de 8 bits se necesitan 8 líneas.

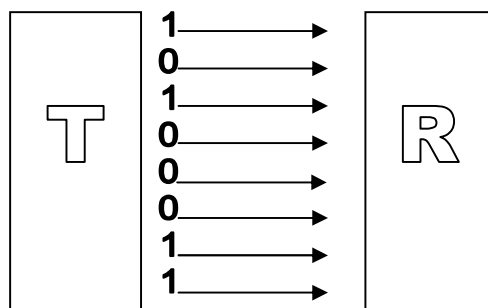


Figura 3. Transmisión de datos en paralelo.

1.3.2 Transmisión serie: Llamado también *serial por carácter*. La transmisión en serie significa el envío de un bit después de otro a través de un único canal. En la Figura 4. se muestra como el código binario se transmite en forma serial, enviando un dato de 8 bits por una sola línea.

Cuando se usa para distancias largas, la transmisión en serie ofrece dos ventajas sobre la transmisión en paralelo:

- Se evita el problema de asegurar que todos los bits enviados en paralelo se reciban también en paralelo en el otro extremo. La diferencia en los tiempos de programación,

que conduce a llegadas desfasadas de bits en canales paralelos, no se presenta si se utiliza un solo canal.

- El costo de enlace, en bits por segundo de capacidad, suele ser menor que el de utilizar transmisión en paralelo. Esto refleja el creciente predominio del costo del circuito, a menudo el costo del hilo de cobre sobre los costos de la interfaz transmisor-receptor en los extremos del circuito a medida que aumenta la distancia. Es más barato aumentar la complejidad del par transmisor-receptor, consiguiendo así mayor capacidad de rendimiento, que buscar el mismo resultado añadiendo circuitos en paralelo.

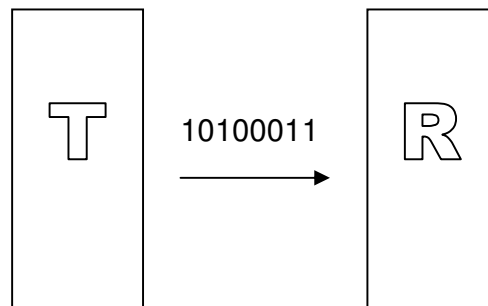


Figura 4. Transmisión de datos en serie.

1.3.2.1. Formas de transmisión en serie: Existen dos formas de transmisión serial:

- Asíncrona: Dentro de la cadena de información por cada carácter se tiene que enviar la señal de inicio y parada (*start* y *stop*). Esta forma de transmisión en serie es la utilizada en este trabajo. Sus principales características son:

- Depende de las señales de start y stop.
- Alta sensibilidad a la distorsión.
- Tiene velocidades muy limitadas.

- Eficiencia de 70%, para pocos volúmenes de datos.
- Bajo costo de diseño.

En la figura 5. se observa un carácter enviado con datos asíncronos.

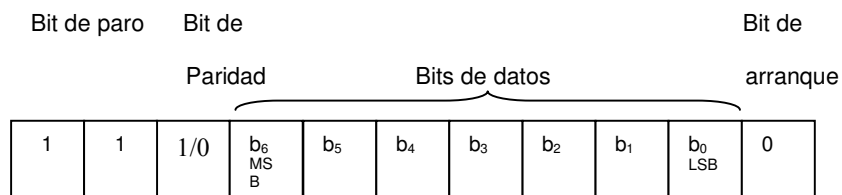


Figura 5. Formato de datos asíncronos.

- Sincrónica: Con los datos síncronos, en vez de entramar cada carácter independientemente con los bits de arranque y parada un carácter de sincronización único llamado SYN se transmite al comienzo de cada mensaje. En la Figura 6. Se muestra el formato de este tipo de sincronización. Sus principales características son:
 - Envío de caracteres durante un tiempo.
 - Receptor y transmisor deben estar sincronizados.
 - Se tiene señal de reloj para sincronizar.
 - Envían bloques de caracteres.
 - Independencia de señal
 - Reducida sensibilidad a distorsión
 - Manejo de altas velocidades
 - Alto costo de diseño.
 - Eficiencia para alto volumen de datos.

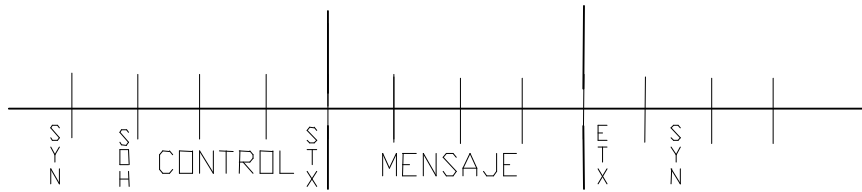


Figura 6. Formato de datos síncronos.

1.4 CODIFICACION

Es la conversión de símbolos en otros para lograr los siguientes objetivos:

- Transmitir símbolos a través de señales eléctricas digitales.
- Posibilidad para detectar errores debidos a distorsión-ruido.

Por lo tanto un código es una configuración de bits ej.

Tabla 1. Código de bits.

Nombre	caracteres	longitud de Bits
Baudot	31	5
BCD	64	6
ASCII	128	7
EBCDIC	256	8

1.4.1 Tipos de códigos. Los códigos de acuerdo a su empleo pueden ser:

- Control STX-ETX-SOH---
- Especiales ; , | * ---

- Alfa A a b c D ----
- Numéricos 1,2,3,4,-----

En el anexo K se encontrara Tabla de código ASCII.

1.5 EL MODELO OSI

El termino *interconexión de sistemas abiertos (OSI)* es el nombre dado a un conjunto de estándares para la comunicación entre computadoras. El objetivo principal de los estándares OSI es servir como guía estructural para intercambiar información entre computadoras, terminales redes. El OSI se apoya por la ISO y CCITT. En la figura 7. se muestra el modelo de siete capas de Interconexión de Sistemas Abiertos ISO. Las principales características son

- Cada nivel entrega un servicio particular al siguiente nivel para que el dato sea entregado correctamente y en forma reconocible.
- Los 3 niveles inferiores 1-2-3- entregan servicios de red-network asociado con los protocolos y aparecen en cada nodo del network. Entregan una comunicación libre de error entre dos nodos.
- Los 4 niveles superiores 4-5-6-7- entregan servicios a los usuarios mismos y están asociados con los desarrollos de los usuarios.

1.5.1 Nivel físico. Especifica los estándares físicos, eléctricos, funcionales y de procedimientos para ingresar en la red de comunicaciones de datos. Como tenemos en las especificaciones de la interface serial RS232C.

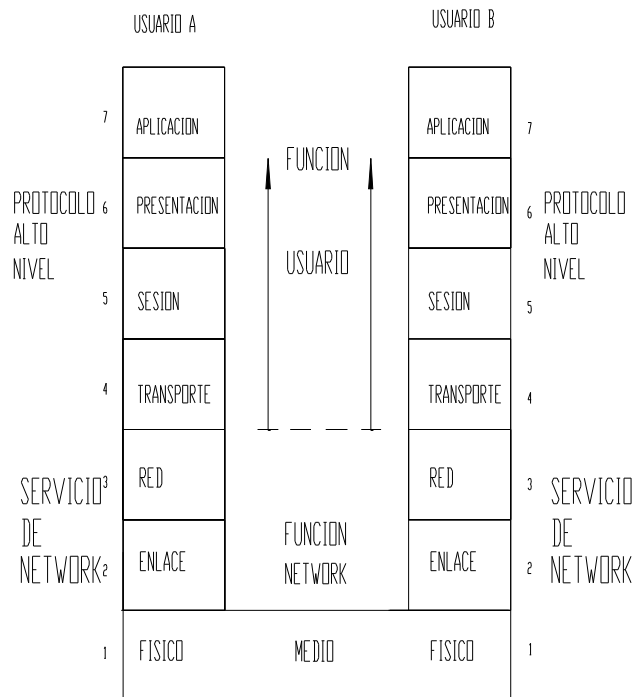


Figura 7. Capas del modelo OSI.

1.5.2 Nivel de enlace. Es la responsable de las comunicaciones entre los nodos primarios y secundarios dentro la red, proporciona un medio de activar, mantener y desactivar el enlace de datos. Como ejemplo tenemos las comunicaciones bisíncronas de IBM y el control de enlace de datos síncronos (SDLC).

1.5.3 Nivel de red. Determina que configuración de red (marcación dedicada o

en paquetes) es la más apropiada para la función que proporciona la red. También define el mecanismo en el cual el mensaje se divide en paquetes de datos y son enrutados desde un nodo de salida a un nodo receptor dentro de una red de comunicaciones.

1.5.4 Nivel de transporte. Controla la integridad de un extremo en el otro extremo del mensaje la cual incluye mensajes de ruteo, segmentación y recuperación de error. Las capas por encima de las capas de transporte no considera los aspectos tecnológicos de la red. Las tres capas superiores se dirigen a los aspectos de aplicaciones de la red, mientras que las tres capas inferiores se dirigen a la transferencia de mensajes. Por lo tanto la capa de transporte actúa como interface entre la red y la capa de sesión.

1.5.5 Nivel de sesión. Es la responsable de la disponibilidad de la red (Es decir almacenamiento de buffers y capacidad de procesamiento). También incluyen procedimientos para el ingreso y abandono de la red así como verificar la autenticidad del usuario. Esta capa también determina la disposición del tipo de diálogo (es decir simplex, half dúplex o full dúplex).

1.5.6 Capa de presentación. Se dirige a cualquier conversión necesaria de códigos o sintaxis para presentar los datos a la red en un formato común para comunicaciones. Sus funciones incluyen el formateo de archivos de datos, codificación (ASCII, EBCDIC, etc.), encriptación y desencriptación de mensajes, procedimientos de

diálogos, compresión de datos, sincronización de datos, interrupción y terminación. La capa de presentación realiza la traslación del conjunto de códigos y caracteres y determina el mecanismo para el desplegado de los mensajes.

1.5.7 Capa de aplicación. Es la más alta de la jerarquía y es análoga al administrador general de la red.

1.6 LA COMPUTADORA

No solo las computadoras utilizan grandes y complicados programas de teleproceso, con procesadores frontales especiales para el manejo de muchos enlaces de comunicación, cada uno con sus propios adaptadores de línea, módems y protocolos, sino que la propia infraestructura para comunicaciones está basada en computadoras. Los canales de comunicación se mezclan en circuitos comunes usando multiplexores controlados por microprocesador, y el tráfico se intercambia en conmutadores controlados por computadora.

Las telecomunicaciones invadieron el campo de las computadoras casi en el mismo grado que las computadoras invadieron las telecomunicaciones. El programador y usuario informado de computadoras deben familiarizarse con esta convergencia de las tecnologías de las telecomunicaciones y la información, con frecuencia denominada TELEMÁTICA, el término telemática se utiliza a veces con significado más restringido para referirse a sistemas basados en textos como teleles o teletex.

La tecnología de la comunicación de datos esta basada, sin embargo en principios más generales que los productos específicos de un fabricante.

Incluye los principios generales para el transporte de datos, los mandatos, la sincronización y otra información por lo general en una sola cadena de bits en serie, las técnicas que permiten la transmisión de datos en forma segura ante la presencia de errores en un canal, el diseño de software que de respuesta rápida a lo que frecuentemente son entradas de alta velocidad y no solicitadas, los métodos de control de acceso de computadoras independientes de un medio común.

1.7 CONCEPTO DE PROGRAMA

Un microprocesador es un diminuto conjunto de miles de elementos lógicos e interconexiones diseñado para responder a las instrucciones de un programa, el cual le indica lo que debe hacer en cada instante.

Sin un programa que le dé vida, un microprocesador y todo el hardware desarrollado sería una masa inerte de componentes electrónicos sin un objetivo específico. De hecho, la función primaria de un microcontrolador es ejecutar programas.

Las funciones realizadas por un microcontrolador quedan definidas por un conjunto de instrucciones diseñadas específicamente para él. El trabajo del programador consiste, precisamente, en elaborar *programas*, es decir combinar estas instrucciones, que son

relativamente simples, de una manera lógica con el fin de permitir que el microcontrolador lleve a cabo la tarea final deseada.

Las instrucciones que constituyen un programa se almacenan en la memoria del sistema como patrones de 1's y 0's, organizados en grupos de 8 bits, siendo esta la longitud de la palabra para el microcontrolador 80C51, y también se utiliza la memoria EPROM 27C64 de 65536 (64 K) posiciones, cada una con capacidad de almacenar un byte o palabra de 8 bits. Esto implica que el sistema utiliza un bus de direcciones de 16 bits (2 bytes).

Una determinada posición de memoria puede contener una instrucción, un dato o parte de un dato, el microcontrolador debe empezar por direccionar la posición o posiciones de memoria donde se encuentra la instrucción, leer o captar esta instrucción por medio de una operación de lectura y decodificarla o interpretarla mediante sus circuitos internos con el fin de determinar la secuencia de microinstrucciones necesarias para su ejecución.

Los pasos anteriores constituyen lo que se denomina un *ciclo de instrucción*. Durante su operación normal, un microcontrolador no hace otra cosa que ejecutar repetidamente ciclos de instrucción. La complejidad y duración de cada ciclo de instrucción dependen de la instrucción particular que el microcontrolador está ejecutando en un momento dado.

Para ejecutar un programa, un microcontrolador debe tenerlo almacenado en forma binaria (patrones de 1's y 0's) en posiciones adyacentes de la memoria del sistema. Esta forma binaria del programa se denomina *lenguaje de máquina* y es el único lenguaje que entiende el microcontrolador.

Para evitar confusión que, desde el punto de vista del programador, pueda generar una larga lista de 1's y 0's, estos códigos se pueden representar en forma hexadecimal, constituyendo lo que se denomina un *programa de código objeto*.

No todos los códigos de un programa objeto o en lenguaje de máquina corresponden a instrucciones ejecutables, hay algunos que corresponden a datos y direcciones que son utilizados por el programa para propósitos particulares. Los códigos correspondientes a instrucciones ejecutables se denominan comúnmente códigos operacionales (*op-codes*). Cada código de operación le indica, directa o indirectamente, al microcontrolador la dirección de memoria donde se encuentra almacenada la siguiente instrucción ejecutable. Al recibir un código de operación, los circuitos internos del microcontrolador lo descifra. La información obtenida le permite al dispositivo identificar la naturaleza de los bytes que siguen al *op-code*. Consideremos por ejemplo, la instrucción 3E0C. En este caso, el op-code es el byte 3E, el cual una vez decodificado, le informa al microcontrolador que debe almacenar en un registro interno del mismo llamado *acumulador* el dato que viene a continuación, es decir 0C, y por lo tanto el acumulador se carga con el dato 00001100. En el anexo I se encuentran las instrucciones en lenguaje de mnemónico para el microprocesador 80C51.

Desde un punto de vista general, las instrucciones de un microcontrolador se pueden agrupar en tres grandes categorías.

- Instrucciones de transferencia de datos.
- Instrucciones de procesamiento de datos.
- Instrucciones de control de programa.

Las instrucciones de transferencia de datos, que son las más numerosas, mueven información sin alterar su contenido de una parte a otra del sistema. Este movimiento de datos puede efectuarse dentro del microcontrolador, entre el microcontrolador y la memoria o entre el sistema de entrada/salida y el microcontrolador y la memoria. Ejemplos de este tipo de instrucciones son: cargar un registro interno del microcontrolador con un dato que se da a continuación del *op-code* o que el microcontrolador debe buscar en la memoria o en un puerto de entrada, transferir el contenido de un registro a otro, a una posición de memoria o a un puerto de salida.

Las instrucciones de procesamientos de datos transforman la información desde el punto de vista lógico y aritmético utilizando circuitos de la unidad aritmético-lógica (ALU) del microcontrolador. Entre las operaciones realizadas por la ALU figuran adición y sustracción, rotación, incremento y decremento, AND, OR, XOR, etc.

Las instrucciones de control de programa determinan la secuencia de ejecución de las instrucciones. Permiten transferir el control de una parte del programa a otra, o entre

diferentes subprogramas. Ejemplos de este tipo de instrucciones son los *bifurques* o saltos, las llamadas y retornos de *subrutinas*, las atenciones de *interrupciones*, etc.

1.7.1 Lenguajes de programación. Un programa en lenguaje de máquina carece de sentido inmediato para un ser humano, quien sólo ve una lista de valores numéricos sin sentido lógico. Además, la escritura de programas en lenguaje de máquina puede resultar pesada y tediosa y está sujeta a errores. Para evitar esto, se han desarrollado una serie de *lenguajes de alto nivel* cuyo objetivo es hacer más agradable la programación de microprocesadores, microcontroladores y microcomputadores desde el punto de vista del usuario.

Los lenguajes de programación se pueden dividir, a groso modo, en las siguientes jerarquías:

- Lenguajes de microprogramación
- Lenguajes de máquina
- Lenguajes ensambladores
- Lenguajes de alto nivel

El lenguaje de microprogramación o microprograma es, normalmente, invisible para el usuario de un microcontrolador porque reside en una parte interna del mismo llamada *ROM del microprograma*. El microprograma interpreta las instrucciones en lenguaje de máquina recibidas por el microcontrolador y es el responsable de generar la microinstrucciones necesarias para ejecutar correctamente cada instrucción. La mayoría de microcontroladores tienen el set de instrucciones fijo e inmodificable.

El lenguaje de máquina (*machine language*), como se explicó anteriormente, es la forma binaria (patrones de 1's y 0's) del programa y es el único lenguaje que, en última instancia, el microcontrolador puede entender. Una forma simplificada del lenguaje de máquina es el código objeto, el cual representa cada código binario por su equivalente hexadecimal.

El lenguaje ensamblador (*assembly*) y los lenguajes de alto nivel (Pascal, Turbo C++, Visual C++, etc.) consisten en una serie de instrucciones especiales (*statements*) que son más fáciles de entender por las personas que las instrucciones en lenguaje de máquina o en código objeto.

Para poder ser comprendidos por el microcontrolador, estas instrucciones deben ser convertidas a lenguaje de máquina por medio de un programa llamado *traductor*, como se ilustra en el Figura 8.

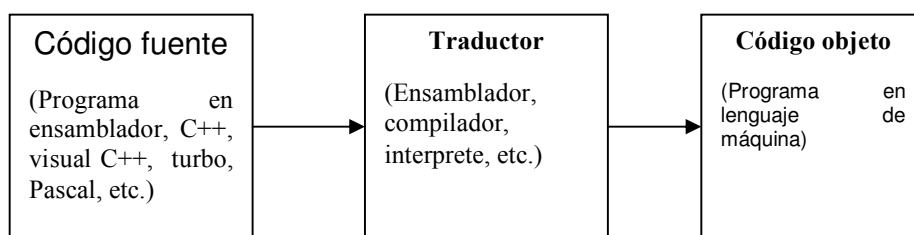


Figura 8. Conversor de lenguaje ensamblador a lenguaje de máquina

El programa traductor que convierte lenguaje ensamblador en lenguaje de máquina se denomina *ensamblador (assembler)* y el que convierte lenguaje de alto nivel en lenguaje de máquina se denomina *compilador o intérprete*. El lenguaje (ensamblador o de alto nivel) que entra en un traductor se denomina *código fuente* y el que sale del mismo, como sabemos, se denomina *código objeto*.

El lenguaje ensamblador fue el utilizado en la programación del microcontrolador 80C51 que maneja los efectos y letra de la matriz de diodos, este lenguaje posee varias e importantes características que hacen de un programa escrito en este lenguaje más fácil de leer, escribir e interpretar que uno escrito en lenguaje de máquina. Algunas de las características claves son las siguientes:

- En lugar de números, cada instrucción ejecutable (op-code) se representa mediante una abreviatura llamada *mnemónico*, fácil de recordar, la cual define la operación realizada. Normalmente un mnemónico consta de dos o cuatro letras y corresponde, usualmente, a las iniciales o la abreviatura de la palabra o palabras en inglés que describen la instrucción. Por ejemplo MOV es el mnemónico de una instrucción que mueve (*move*) o transfiere el contenido de una parte del sistema microcontrolador (un registro, una posición de memoria, etc.) a otra sin modificarla.
- Las direcciones absolutas de memoria se representan también por mnemónicos llamados *rótulos o etiquetas (labels)*, que son símbolos o grupos de símbolos que define el usuario. Generalmente termina en dos puntos (:). Lo anterior obvia la necesidad de conocer durante la escritura del programa, la dirección específica de la memoria donde

quedará almacenada la instrucción a que hace referencia cada etiqueta. Por ejemplo si dentro de un programa existe una instrucción de bifurque que le ordena al microcontrolador saltar a una parte del programa donde se realiza una multiplicación, la dirección de salto puede etiquetarse como MULT; o similar.

- Se pueden usar mnemónicos llamados *seudo-códigos de operación (pseudo-ops)* para facilitar la labor de programación. Estos pseudo-códigos, aunque parecen códigos de operación, no son instrucciones ejecutables, es decir no son convertidos por el ensamblador a lenguaje de máquina. Sin embargo proporcionan información importante que necesita el ensamblador en el proceso de traducción. Los *pseudo ops* se denominan también *directivas del ensamblador*. Como ejemplos tenemos el ORG (origen), utilizado para indicar la ubicación de la primera instrucción ejecutable del programa y la EQU (equivalente a...) utilizado para asignar símbolos a valores numéricos con el fin de que estos últimos posean un significado más evidente ante nosotros. Por lo tanto la pseudo-instrucción ORG 0000 le informa al ensamblador que el programa comienza en la dirección 0000 y la `FREC: EQU 74` le indica que cada vez que se encuentre con el mnemónico `FREC` debe sustituirlo por el valor numérico hexagesimal 74 (0111 0100, en binario).

Los programas en lenguaje ensamblador deben ser escritos en una forma estándar con el fin de que pueda ser fácilmente convertidos a lenguaje de máquina por el programa ensamblador. Desde este punto de vista una instrucción en lenguaje ensamblador consta de cuatro secciones o campos, como se muestra en el cuadro 1.

Cuadro 1. Campos de una instrucción en ensamblador.

ETIQUETA OPERANDOS	OP- CODE	COMENTARIOS	
BUCLE:	IN	01H	; Entrar dato del puerto 1

El primer campo corresponde a la etiqueta, es un mnemónico alfanumérico de cinco o seis caracteres (siendo el primero una letra) para referirse a direcciones que no son específicamente conocidas. Normalmente una etiqueta termina en dos puntos (:) para separarla de los otros campos. Ejemplos: LLAMA:, BUC 10:, ERROR:, CLOCK:, ASTRA:, VI_10:, etc. Este campo no siempre se usa

El segundo campo corresponde al mnemónico del *op-code* (código de operación) de la instrucción a ser ejecutada. Cada microcontrolador tiene su propio conjunto de códigos mnemónicos para designar cada una de las instrucciones a las que responde. Ejemplos de *op-codes* para 80C51 son MOV, MOVX, SETB, CALL, INC, PUSH, etc.

El tercer campo corresponde a los operando o parámetros requeridos por la instrucción. Contiene, separados por comas (,) las designaciones de los registros y/o los datos o direcciones sobre los cuales actúa la misma. Por ejemplo en el lenguaje del 80C51, la instrucción MOV B, A transfiere el contenido del registro A al registro B. En

este caso MOV es el código de operación y A, B los operandos. A actúa como registro fuente y B como registro de destino.

El cuarto campo corresponde a los comentarios. En esta área se describen, entre otras cosas, notas especiales que sirven de referencia no solamente al programador sino a cualquier persona que consulte el programa en un futuro y explicaciones de lo que hace cada instrucción o grupo de instrucciones desde el punto de vista de la aplicación. Los comentarios comienzan generalmente, con un asterisco (*) o un punto y coma (;) y no son ejecutables, es decir son ignorados por el programa ensamblador, y por tanto, no son convertidos a códigos en lenguaje de máquina.

1.8 LA NORMA EIA-232-C

La Electronic Industries Association (EIA) introdujo en el año 1962 la norma RS-232 (RS=Recommended Standard) con el propósito de uniformar el interface entre los DCE (Data communication Equipment) y los DTE (Data Terminal Equipment). El DTE es el equipo que actúa como fuente de los datos, destino de los datos o ambas cosas a la vez (usualmente un ordenador).

El DCE es el equipo que proporciona las funciones necesarias para establecer, mantener y terminar una conexión, y realiza la conversión de las señales o la codificación necesaria para la comunicación (usualmente un MODEM).

El creciente uso de ordenadores personales PC ha creado nuevas áreas de aplicación EIA-232, y la norma ha sido adoptada como interfase de bajo coste, para periféricos, tales como ratones, scanners, impresoras, plotters, digitalizadores y modems.

En el año 1969, la EIA publico la RS-232-C una versión de la norma original. En el año 1986 aparece la norma EIA-232-D que esta en la misma línea de las normas CCITT V.24, V.28 e ISO IS2110. La vieja norma EIA-232 proporciona todavía una solución económica a la comunicación de datos pese a no estar excenta de problemas, como su alta susceptibilidad al ruido, baja velocidad de comunicación y limitada longitud de transmisión.

La norma define las dimensiones y especifica la asignación de señales a las 25 patillas del conector del tipo D, pese a lo cual muchos laptops y PC actuales utilizan un conector de 9 patillas DB9 no contemplado.

Las patillas y señales más importantes se muestran en la figura 9. (el primer número se refiere al conector de 25 patillas y el segundo número de 9 patillas):

- (7/5) GND tierra de señal.
- (2/3) TXD transmisión de datos. Para transmitir un dato es preciso que las señales DTR, DSR, RTS, y CTS estén en ON
- (3/2) RXD recepción de datos.
- (20/4) *DTR Data Terminal Ready*. El DTE dice al DCE que está preparado para recibir o transmitir datos.

- (6/6) DSR *Data Set Ready*. El DCE dice al DTE que esta conectado al canal (se ha marcado el número, se ha establecido la comunicación y se esta en modo de transmisión de datos y no en modo de transmisión de voz).
- (4/7) RTS *Request to send*. El DTE dice al DCE dice que quiere transmitir. También se emplea como control de dirección en modo half dúplex.
- (5/8) CTS *Clear to Send*. El DCE dice al DTE, como respuesta al RTS, que el DTE puede transmitir.

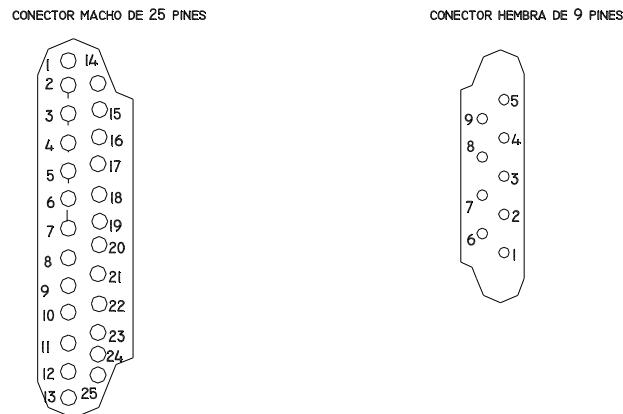


Figura 9. Conectores RS-232

- (8/1) DCD *Data Carrier Detect*. El DCE dice al DTE que está recibiendo señales válidas del canal.
- (22/9) RI *Ring Indicator*. El DCE dice al DTE que está recibiendo una indicación de llamada (se utiliza en sistemas de respuesta automática).

La impedancia de entrada del receptor está comprendida entre 3 Kohmios y 7 Kohmios. La existencia de esta impedancia es la causa de que haya siempre un flujo de corriente

del transmisor hacia el receptor aunque no se estén enviando datos. En equipos alimentados con baterías es deseable desaparecer este flujo inútil.

La máxima velocidad de comunicación es de 20.000 bits por segundo.

La máxima longitud del cable depende de su capacitancia. La norma limita la máxima capacidad de línea a 2.500 pF incluyendo la capacidad del cable. Dada la capacitancia típica de los cables de conexión es de 150 pF/m, la longitud máxima del cable se encuentra alrededor de 15 metros.

2 EL MODEM

2.1 INTRODUCCION

A continuación se detallan las partes más importantes del manual de operación del módem utilizado en este trabajo cuya referencia es KPC-2400 marca KANTRONICS, para mayor información y detalle refiérase al anexo A, también se sugiere que utilice algunas de las referencias listadas en la bibliografía de este manual.

El KPC-2400 es un diseño de Hardware y Software que utiliza todas las características del modelo KPC-2, más una adición para que el módem envíe a 2400 BPS. El Hardware utiliza una versión del microprocesador 6303, y el software es un programa escrito para complementarse con el protocolo ARRL versión 2.0. Para operación a 300 y 1200 Baudios, la unidad usa el circuito integrado AM-7910; para operación a 2400 BPS la unidad usa en el *kantronics differential phase shift keyed* (DPSK).

El KPC-2400 está equipado con 8K RAM en U-13 y en U-14 para un total de 16K.

La RAM puede ser expandida a 32K reemplazando U-13 y U-14 con una sola RAM de 32K en U-14. Si se expande la RAM de 32 K debe tenerse en cuenta que el tiempo de acceso sea de 250 ns o menos.

En el texto de este manual se usa el título genérico TNC (*Controlador Terminal de Nodo*) para hacer referencia al Kantronics KPC-2400. Un TNC es usado para controlar una estación sencilla (Nodo) en un grupo de estaciones que operan juntos regularmente.

2.2 DESCRIPCION DEL PACKET RADIO

La comunicación vía packet radio involucra la transmisión de datos digitales por medio de radio haciendo uso de transceptores típicamente aficionados. Para los diferentes sistemas el dato no-pasa directamente desde la computadora al radio o viceversa, por lo tanto esta es direccionado a través de un TNC donde las características especiales de su nodo son implementadas. El KPC-2400 (TNC) es un dispositivo basado en un microcontrolador el cual es conectado a una fuente de datos digitales por medio de un puerto serial RS-232 y a un transceptor normal vía radio por medio de su puerto de radio. Usualmente el TNC se conecta a un terminal de datos o a una computadora a través del puerto serial y todos los comandos para el TNC así como los datos han de ser transmitidos provendrán de este dispositivo. El TNC manipula todos los tonos involucrados en el actual envío y recepción de datos incluyendo el empaquetamiento de datos dentro de las señales llamadas Paquetes caracterizando y modulando el transmisor y demodulando los paquetes del receptor.

En efecto el TNC hace la mayoría de las tareas conocidas con el mantenimiento del canal del dato. Tales como chequear la integridad de paquetes recibidos y actúa como

una estación de relevo cuando es requerido por otros usuarios del canal. La totalidad de estas funciones son manipulada completamente y en forma automática por el Software y el Hardware del TNC. El usuario solamente necesita saber lo concerniente con la producción y el uso de los datos digitales, descargando la multitud de tareas de comunicación al TNC.

Un paquete típico QSO's comienza con un comando de usuario de su TNC para asegurar un *connect-request* dirigido a la otra estación. El paquete connect-request incluye ambas señales de usuario tal que la estación receptora reconozca que el connect-request es dirigida hacia ella. La estación receptora reconoce la recepción de este paquete y la conexión es establecida. Cada usuario observa que aparece en su terminal `*** CONNECTED TO (CALLSING)` donde ésta es la señal de llamada de la otra estación. EL TNC se conmuta automáticamente al modo de transferencia de datos. Cada paquete es un sándwich consistente de (en orden de transmisión), encabezamiento de información seguido por el dato, seguido por un muy importante número de 16 bits llamado FCC llamado de secuencia de la trama. Este es el empaquetamiento del dato que coloca al Packet radio por encima de todos los tipos anteriores de comunicaciones aficionados, lo cual le da sus propiedades especiales.

Las dos principales características de las comunicaciones en radio Packet son: primera, que esta en "bursty" la mayoría de ocasiones que los ocupantes del canal no están transmitiendo. Sin embargo la excepción a esta regla ocurre, por ejemplo, durante transferencia de archivos de computadora a computadora. Segunda, los datos recibidos

los cuales son enviados por el puerto de datos del TNC, son esencialmente garantizados para estar libres de errores. Esta debido a la detención de error asociada con el seguidor de datos FCS, el cual es usado por la estación receptora para determinar validez del dato. En el evento de que el dato sea incorrecto, la estación transmisora retransmitirá el paquete, hasta un número de veces especificadas por el usuario, en atención de obtener un dato sin errores.

El radio paquete QSO es normalmente terminado cuando una de las estaciones emite un paquete "disconnet-request". Este es reconocido por la estación receptora y ambos usuarios ven ***DISCONNECTED; aparecer en su terminal.

En cualquier instante el TNC puede conmutarse a modo "COMMAND" y uno de los mas de cien parámetros ajustables por el usuario puede ser inspeccionado o cambiado. Estos parámetros controlan todas las facetas de operación del TNC, y la mayoría de ellos están almacenados en una EEPROM la cual es una memoria no volátil, de modo que no necesita resetearse en cada encendido.

2.3 PROTOCOLOS

Las reglas por las cuales el software determina la respuesta al dato recibido, así como los comandos de usuario y datos se conocen como protocolo. Por ejemplo cuando un TNC envía un paquete "CONNECT RESQUEIT" este espera una replica en la forma de un paquete "CONNECT RESQUEIT ACKNOWLEDGED" si todo va bien la estación receptora emitirá tal paquete, la conexión será establecida y cada usuario vera *** connected to (CALLSING) aparecer en su terminal. Sin embargo, si no se recibe la

respuesta apropiada, el TNC original reenviara la llamada por un número de veces predeterminando. Todas estas conductas son dictadas por el protocolo.

Otro ejemplo es cuando dos o más QSO están tomando lugar en el mismo canal (una de las mas grandes ventajas de este modo) y simultáneamente hay transmisión de dos estaciones (colisión), el subsecuente retardo de tiempo y la retransmisión esta determinado por las reglas contenidas en el protocolo.

El Kantronics KPC-2400 usa la versión 2.0 adaptado por el ARRL, en octubre de 1984.

2.4 CONTROLES E INDICADORES

El panel frontal tiene 5 LED's indicadores los cuales suministran la siguiente información:

- Energía –(led verde): Se ilumina cuando se aplica energía. El switch ON/OFF esta localizado en el panel trasero.
- XMIT –(led rojo): Se ilumina cuando el TNC switchea el PPT a envío de datos.
- RCV –(led rojo): Se ilumina cuando el TNC detecta una señal en el canal.
- CON –(led rojo): Se ilumina cuando se establece comunicación con otra estación.
- STA –(led rojo): Se ilumina cuando el TNC ha enviado paquetes los cuales no han sido reconocidos. Cuando esta en operación de múltiple conexión, los led's CON y STA indican estado del flujo de corriente I/O.

2.5 INSTALACION

Antes de comenzar la instalación, por favor lea esta sección. Usted debe cambiar algunos de los jumpers internos, si es necesario.

El TNC esta conectado a su computadora y el transceptor a través de jacks en el panel trasero. Refiriéndonos a la Figura 10. cuatro conexiones son posibles: Potencia, Radio, Audio, Computadora.

Para realizar la interconexión requerida del equipo, se alimenta a 12VDC y se hace su respectivo conexionado. Se debe proveer los conectores y terminales apropiados para los cables desde la computadora y el radio como lo son el jack

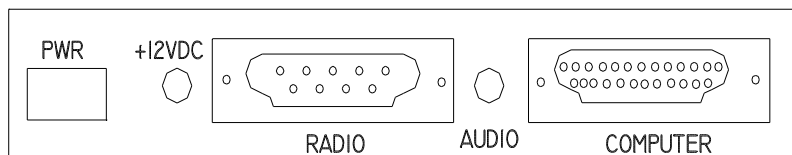


Figura 10. Panel posterior del módem

de mic/sp y puerto serial respectivamente. A continuación se detalla cada uno de los pines del conector:

- **+12VCD** - Conecte la tensión a este jack, alternativamente usted puede también suministra los 12 VDC al TNC a través del conector de radio. El TNC requiere entre +10 y +14 VDC.

- **RADIO** - Las líneas de este jack son usadas para controlar la función PPT del transceptor, entrar los tonos ASFK del KCP-2400 y suministrar otras entradas/salidas alternas como se describe: Ver Figura 11.

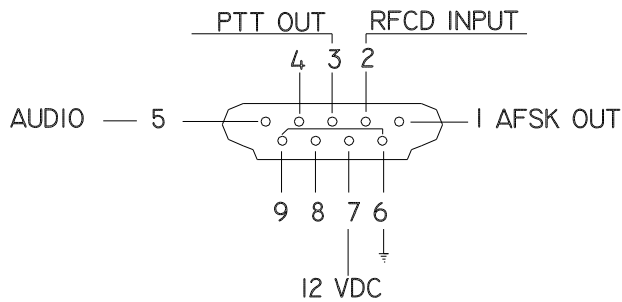


Figura 11. Conector para radio

- PIN 1: (AFSK OUT): Cable de cinco conductores, terminal blanco. Esta línea lleva los tonos AFSK generados por KPC-2400 a la línea de entrada de micrófono del transceptor.
- PIN 2: (RFCD IN): Terminal verde del cable de 5 conductores. Esta línea puede usarse para entrar la línea de replica de su transceptor si lo desea. Esta conexión no será normalmente requerida, no usada, a menos que opere en un canal de voces partidas.
- PIN3: (SALIDA PPT): Terminal marrón de cable de 5 conductores. Esta línea controla la línea PPT de su transceptor, permitiendo al TNC swichear al transceptor desde/a transmisión cuando es necesario enviar paquetes.
- PIN 4: No se conecta.

- PIN 5: (SEÑAL DE AUDIO): Cable de audio de dos conductores/ terminal central. Esta línea es prealambrada para usarse como entrada de audio del jack del parlante externo de transceptor. Si se usa el jack de audio permanece disponible para conectar un parlante externo
- PIN 6 y 9: (TIERRA): Cable de audio de dos conductores/apantallado esta línea suministra tierra entre el TNC y el transceptor.
- PIN 7: (+12VDC): Terminal rojo del cable de 5 conductores. Es suministrado como entrada alternativa de potencia. Puede usarse en lugar del jack de +12 VDC. En este trabajo no se uso esta entrada alterna. Esta conexión no debe utilizarse para alimentar cualquier otro dispositivo.
- PIN 8: No se conecta.

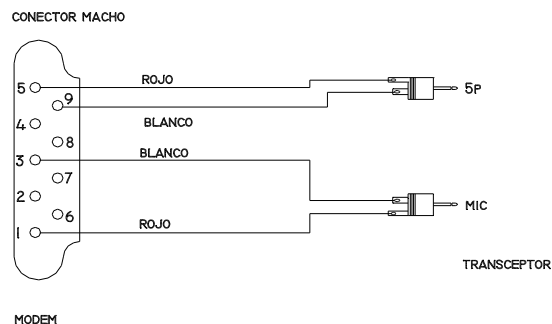


Figura 12. Conexionado del módem al radio

En la figura 12. se observa el conexionado existente entre el terminal DB-25 del KPC-2400 y el transceptor 2AT-ICOM.

- **COMPUTADOR** - El conector DB-25, es cableado con una línea de 5 hilos. Estas conexiones son todas requeridas para la instalación. En la figura 13. se muestra el conector DB-25 y en la tabla 2. la identificación de cada pin.

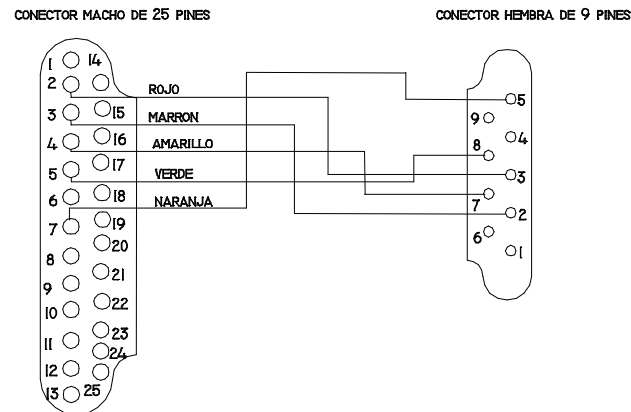


Figura 13. Conexionado del módem al PC.

Las funciones de estas líneas son:

- Transmit Data: Esta línea transmite el dato serial desde el terminal hasta el KPC-2400.
- Recieve Data: Esta línea transmite el dato serial desde el KPC-2400 hasta el terminal.
- Request to Send: Esta línea le dice al KPC-2400 que esta listo y que puede enviar el dato al terminal.

- Clear to Send: Esta línea le dice al terminal cuando o no puede enviar el dato para el KPC-2400. Un nivel ON le dice al terminal que puede enviar el dato mientras que el nivel de OFF no es permisivo para enviar el dato por lo tanto se pierde.
- Data Set Ready: Esta línea desde el KPC-2400 indica que el TNC es leído o habilitado para su uso.

Tabla 2. Señales del módem al PC

Descripción (DCE)	Terminal (DTE)		KPC- 2400	
Descripción	Pin	Dirección	Pin	Color
Transmit data	2	=>=>	3	Rojo
Receive data	3	<=<=	2	Marrón
Request to send	4	=>=>	7	Amarillo
Clear to send	5	<=<=	8	Verde
Data set ready	6	<=<=	-	
Signal ground	7	=====	5	Naranja
Data carrier detect	8	<=<=	-	
Data terminal ready	20	=>=>	-	
+12 VDC IN	25	<=<=	-	

- Signal Ground: Esta línea establece el común de tierra para referencia de potencia.
- Data carrier detect: Esta línea es una salida del KPC-2400 indicando el estado de conexión del KPC. Cuando la línea existe la conexión es confiable.
- Data terminal ready: Esta línea dice que el KPC-2400 esta unido al terminal de la computadora y esta disponible para leer. La corriente del KPC-2400 ignora en el hardware el estado de esta línea.

Una adición especial se implementa en el pin 25 del KPC-2400 para la alimentación de +12 VDC referenciado al pin 7.

Los 12 DVC de entrada se puede usar para alimentar el KPC-2400 alternos de entrada en J2, este pin seria vivo o positivo. En el pin 18, es para probar el procesador, es para uso del personal de la fabrica, únicamente en reparaciones y servicios de operación.

2.6 SELECTORES INTERNOS

Para entregar una máxima facilidad, en la configuración del KPC-2400 están unos selectores para lo siguiente:

- NIVEL DEL AFSK: El selector K1 provisiona un nivel de salida alto o bajo. En la posición, salida alta el nivel es de 44mv pp (en circuito abierto, 600 ohmios nominales). En la posición de salida baja, el nivel es de 10 mv pp. La unidad viene equipada con este selector en el nivel de salida alto. Si quiere un nivel de salida mas alto para su radio lo puede obtener pero cambiando el valor de R32, con el selector en posición HI y en circuito abierto.

Tabla 3. Valores de R-32 vs nivel de ASFK

R32 (ohmios)	ASFK SALIDA
100	4.5 mv
220	10 mv
	150

470	21 mv
470	21 mv
1 K	44 mv
1.5 K	65 mv
2.2 K	94 mv

Si se requiere de un nivel de AFSK mas alto el valor de R32 se puede incrementar aun más para obtener los valores de incrementos aproximados a los mostrados en la tabla. Para obtener un valor máximo de salida los puentes HI y LO se deben quitar. De esta manera el nivel AFSK de salida llegaría aproximadamente a 1.5 Vpp. Ver tabla 3. De esta forma quedaron los módem utilizados en este trabajo.

- RS-232/TTL (SELESTOR K-2): Este selector facilita al KPC-2400 comunicarse con cualquier puerto RS-232 o niveles de voltaje TTL. La unidad viene equipada con este selector en la posición RS-232.

- PRUEBA/NORMA (SELECTOR K3): Este selector inicializa la memoria EEPROM y le coloca valores de inicio. De esta manera queda verificada. La EEPROM y la RAM son las únicas que se chequean en este caso. El equipo es entregado con el selector en la posición normal. Si es necesario reiniciar la EEPROM, apague el equipo y coloque el puente en la posición TEST, luego encienda el equipo. Si la unidad esta conectada a una terminal colóquela a 300 Baudios para que trabaje correctamente, esta puede imprimir el siguiente mensaje: EEPROM INIT OK.

CHECKSUM OK.

RAM OK. 4000H BYTES.

En este instante apague la unidad y coloque el selector en la posición normal.

No intente usar el teclado sin antes apagar la unidad ya que esto activaría una rutina especial de prueba que se realiza en la fábrica. Este selector normalmente no se usa. Usted puede tener cierto permisivo en el TNC si requiere algún cambio, usted puede hacer esto usando este selector. Colóquelo en la posición TEST y prenda el equipo para reinicializar el TNC y colocar sus ajustes a valores de iniciación.

2.7 CHEQUEOS INICIALES

Cuando la instalación de la computadora este totalmente correcta y haya instalado su programa terminal, usted puede determinar que el TNC, cables y computadora, opera correctamente. Prenda la computadora - TNC, en este momento el TNC utiliza un baudio automático de rutina para sincronizar la salida de la computadora este mensaje puede aparecer en el monitor:

PRESS (*) TO SET BAUD RATE (presione * para ajustar los baudios).

Es posible que aparezcan caracteres ilegibles pero no se asuste, esto es normal ya que puede suceder que la computadora este ajustado en un valor de comunicación de baudios diferente.

Usted puede presionar (*) en el teclado en el momento del mensaje legible o un segundo antes de su terminación, después aparece en el monitor el siguiente mensaje:

- KANTRONICS PACKET COMMUNICATOR 2400 V 2,3

En estos momentos usted puede entrar comandos en la pantalla, y observar para familiarizarse con las pantallas del TNC, como ejemplos teclee los siguientes comandos:

- DISP (ENTER): Usted puede observar que el TNC responde con un listado completo en el monitor de todos los comandos presentados en este manual.

Al lado de cada uno puede mostrar los valores de iniciación o modificación de estos comandos y sus parámetros. Al final de la pantalla usted puede observar el comando PROMPT. Este PROMPT indica que el TNC esta en el modo de comando y lectura para instrucciones adicionales.

2.8 OPERACION INICIAL

El primer paso para la operación es colocar la señal de llamado. Para esto entre por el comando PROMPT:

cmd: MY CALL W0XI (CR) (Use nuestra señal de llamado)

A la entrada del comando requerido retorna nuevamente al CMD PROMPT en la pantalla. Para verificar que el TNC acepto el llamado, digite:

MY CALL (CR)

A este comando aparece en pantalla:

MY CALL (your collsing here) (Su llamado esta aquí)

cmd:

El TNC acepta su llamado y la almacena en la RAM. Este permanece en la RAM hasta que se apague el equipo. Si desea que el TNC mantenga el llamado para cada vez que encienda el equipo usted puede hacerlo a través del comando PERM se hace digitando:

PERM (CR)

Su llamado esta almacenado ahora en la EEPROM y estaría aquí después de un cierto periodo de apagado. El comando PERM solo se usa para guardar en la EEPROM algunos cambios hechos en los parámetros y comandos que solo son usados para inicializar su estación.

2.9 CONECTANDO Y DESCONECTANDO

Para hacer uso de los TNC'S en forma ordenada, capaz y confiable para la comunicación de datos, usted puede establecer comunicación con otra estación.

En el momento que usted digita todo para conversar, el TNC automáticamente identifica la estación y los paquetes enviados entre su estación y las otras estaciones, para que puedan ser identificados y reconocidos los mensajes por el receptor.

La estación que envía los mensajes continua enviando o retransmitiendo hasta que se haya recibido correctamente. La estación a conectar para este trabajo sera NODO por Ejemplo. Digite:

CONNECT NODO

Si NODO esta en el aire, y tiene sintonizada la frecuencia con el transmisor, éste recibe un mensaje de notificación desde el otro TNC. Si le tiene conectado un parlante a su radio usted podrá escuchar los paquetes; de otra forma puede observar el led RCV LED (led indicador de recepción). Cuando se requiere una respuesta de conexión (del paquete enviado por el TNC) saber si fue reconocido, el TNC muestra el mensaje en la pantalla:

***CONNECTED TO NODO

Desde este momento usted tiene enlazado los dos módem y puede enviar mensajes de la forma que se explica en el anexo L.

Cualquier información referente al módem KPC-2400 marca Kantronics se puede consultar en el anexo A.

3 EL TRANSCEPTOR

3.1 INTRODUCCIÓN

En este capítulo se estudia lo referente a los equipos utilizados para transmitir, recibir y manipular la información en forma análoga por radio frecuencia (R.F.).

En adelante reciben el nombre de transceptores porque cada uno de los radios tiene las funciones de transmisor y recibir información. La señal análoga es entregada por el Módem al *transmisor* después de hacer la conversión digital-análoga cuando se van a enviar los mensajes desde el teclado del PC. Ésta señal análoga enviada por el transmisor es recibida vía radio por el *receptor* y entregada al Módem para su conversión análoga-digital y recuperar la información digital para presentarla en la matriz de diodos. En este trabajo se utilizaron 2 equipos transceptores marca ICOM 2AT sintonizados a una frecuencia de 144 Mhz. El que se utilizó como transmisor está ubicado en el módulo #1 y el receptor se encuentra en el módulo #2. En el anexo B se encuentran las especificaciones técnicas referentes a este transceptor.

3.7 TRANSMISION EN VHF

Las bandas de muy altas frecuencias (VHF) son las menos afectadas por las variaciones del ciclo de manchas solares y por la capa de Heaviside. Su principal empleo es para comunicaciones a distancias de 150 millas (unos 240 Km), o menores. Estas bandas están ocupadas espaciadamente en las regiones rurales pero en cambio se hallan bastante congestionadas en las zonas urbanas de grandes poblaciones.

En los años anteriores se observado que las señales de muy alta frecuencia son propagadas por otros medios distintos que la transmisión por línea visual, como por ejemplo la reflexión difusa de las señales, la reflexión producida por diversos fenómenos atmosféricos, la incurvación del rayo producido por las masas de aire, etc., permiten que las comunicaciones con estas ondas alcancen hasta unas 1200 millas (unos 2000Km). Las condiciones atmosféricas afectaran frecuentemente a las comunicaciones a larga distancia realizadas en la banda de 2 metros, y todas las bandas de muy alta frecuencia (VHF) son particularmente sensibles a tales variaciones.

En los años 60 se utilizaron las bandas VHF para transmisiones experimentales de rebote en la luna-tierra y para experimentaciones con satélites repetidores (Proyecto Oscar). Las bandas VHF ofrecen un campo de experimentación muy interesante a los radioaficionados para llegar a la región de las microondas.

3.8 MODULACIÓN.

La modulación consiste en variar las características de la onda de radio en correspondencia con la señal que ha de ser transmitida. La naturaleza de esta señal tiene poca importancia en lo referente al proceso de modulación, lo que determina el tipo de modulación a emplear en el procedimiento por el cual esta señal da una característica distintiva a la onda de radio, característica que permite al receptor convertir de nuevo la onda de radio modulada en la señal original.

La Figura 14. representa una portadora de R.F. modulada en amplitud por una tensión de audio senoidal. Después de la modulación se ve que la onda de R.F. modulada varía aproximadamente desde cero a un ritmo constante, pero la intensidad de los ciclos individuales de R.F. es proporcional a la amplitud de la

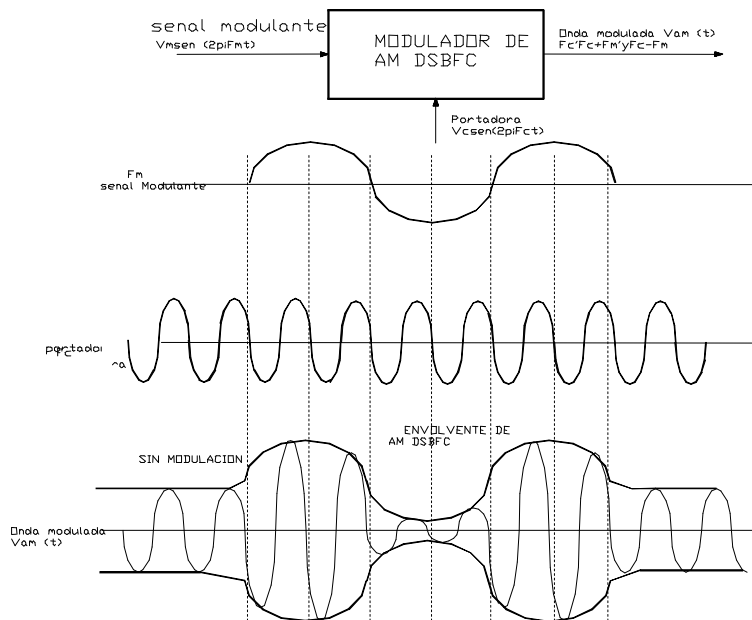


Figura 14. Modulación en AM

tensión de modulación.

La Figura 15. representa la portadora de la Figura 14. y aparece modulada en la frecuencia por la misma tensión de modulación. Cuando la polaridad es igual hace disminuir la frecuencia de la portadora, como se representa por el hecho que los ciclos individuales de R.F. de la portadora aparecen mas espaciados. Una tensión moduladora de polaridad opuesta hace aumentar la frecuencia lo que se aprecia por que los ciclos de R.F. están más juntos, quiere decir que hay mayor número de ellos en un intervalo de tiempo dado.

Las figuras 14. y 15. ponen de manifiesto dos importantes características de las ondas moduladas en amplitud y frecuencia. Primeramente se ve que mientras la amplitud (potencia) de la onda varia en la transmisión con A.M, con la F.M no se produce tal variación. En muchos casos esta ventaja de la F.M. es probablemente

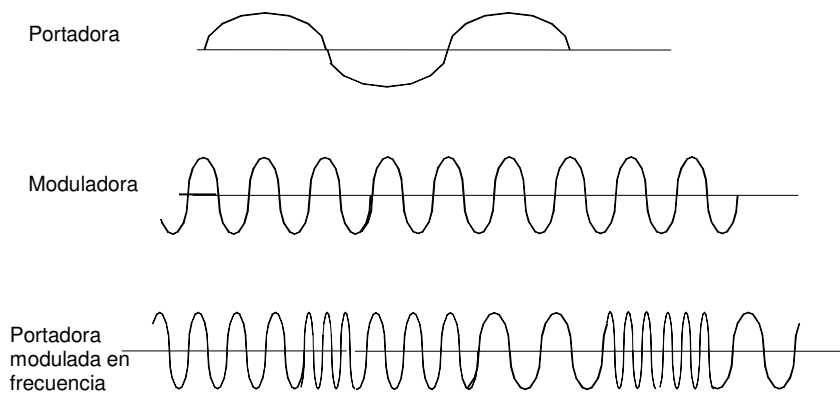


Figura 15. Modulación en FM

de igual o mayor importancia que la conocida capacidad de reducción de ruidos del sistema. Cuando se obtiene una modulación de amplitud de 100%, la potencia media de salida del transmisor debe ser aumentada en 50%. Esta potencia de salida adicional debe ser suministrada, por el modulador mismo, en el sistema de alto nivel. En el sistema de bajo nivel por el funcionamiento de uno o más pasos del transmisor, a su de salida se le suma la potencia para eliminar la distorsión. En cambio un transmisor con frecuencia modulada requiere una cantidad insignificante de potencia de modulador y no necesita suministrar ninguna energía adicional que aumente la potencia en la salida de las crestas de modulación. Todos los pasos entre el oscilador y la antena pueden funcionar como amplificadores clase B o clase C de alto rendimiento, o como multiplicadores de frecuencia.

3.4 MODULACION DE FRECUENCIA

Los primeros experimentos de modulación de frecuencia fueron realizados por el Mayor Edwin H. Armstrong, de la Universidad de Columbia, basándose en la creencia de que el ruido y los estáticos eran variaciones de amplitud a las que no correspondían variaciones ordenadas de frecuencia. En 1934 Armstrong realizó su clásica transmisión de FM en la antigua banda de aficionado de 2.5 metros en N. Y.

Sus siguientes experimentos de aficionado demostraron en 1936 que la F.M. prometía excelentes perspectivas para la comunicación de tipo móvil en las bandas VHF, porque estaban exentas de estáticos y su señal era confiable.

Después de la guerra el desarrollo de la VHF se encontró en la modulación de amplitud en las bandas de aficionado durante dos décadas, a lo que contribuyó la abundancia de equipos militares de VHF sobrantes, y hasta la mitad de la década de los años 60 no fue estimulado el interés de los aficionados por la FM en virtud de la gran cantidad de equipos móviles de FM existentes en el mercado a precios modestos.

Actualmente existen equipos VHF comerciales de dos vías basados en las técnicas de modulación de frecuencia, que proporcionan un excelente rechazo del ruido aleatorio, así como de las frecuencia interferentes en comparación con los sistemas convencionales de A.M. Cuando la amplitud de la señal R.F. es mantenida constante limitada y la información transmitida mediante la variación de la frecuencia o la fase, es imposible eliminar algunos de los efectos disruptivos del ruido. Además los circuitos silenciadores eliminan los picos de ruido y los efectos del ruido de fondo en el receptor hasta que aparece una señal inteligible.

La combinación de rechazo del ruido y el control del silenciamiento provee un margen superior para una potencia primaria dada, en comparación con una asignación de potencia de A.M. equivalente.

3.5 EL TRANSMISOR DE FM

Los diversos métodos directos e indirectos de producir F.M. implican el cambio ya sea de la frecuencia o bien de la fase de una portadora de r.f. en concordancia con la señal moduladora. La señal de F.M es luego elevada hasta la frecuencia de trabajo haciéndola pasar por una serie de multiplicadores de frecuencia. Cuando es multiplicada la frecuencia, la desviación de frecuencia resulta multiplicada por el mismo factor.

Es posible adquirir cristales de poco precio y altamente estables en el margen de 3 a 10 Mhz, y en muchos transmisores de F.M. se utilizan tales cristales para dicha región, multiplicando la frecuencia del cristal por un factor de 12, 18 o 24. A causa de que la amplitud de una señal de F.M. es constante, la señal puede ser amplificada mediante pasos no lineales tales como dobladores y amplificadores clase C sin introducir distorsión de señal. Realmente constituye una ventaja el paso de una señal por etapas no lineales, ya que, en virtud de la acción limitadora inherente a todo amplificador clase C puede ser alisado cualquier vestigio de modulación de amplitud generado en el modulador de fase.

En la Figura 16. se muestra el diagrama de bloques de un transmisor de FM típico. En el Anexo N se encuentra el diagrama a bloques del transceptor ICOM 2AT. El sistema consiste básicamente de una fuente de señal, un procesador de audio, un modulador, un oscilador, varias etapas multiplicadoras de RF, un circuito de CAF y la antena.

El primer amplificador de RF se denomina excitador. La función del CAF (control automático de frecuencia) es estabilizar la frecuencia de salida. En el procesador de audio, la señal proveniente del micrófono se somete inicialmente al proceso de énfasis, luego se amplifica y finalmente se limita.

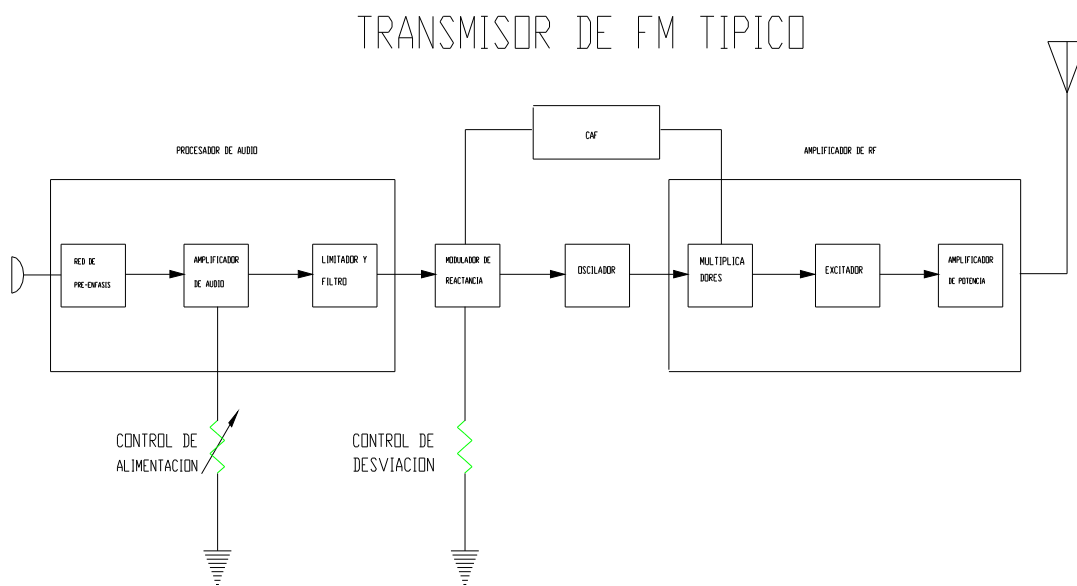


Figura 16. Transmisor de FM.

El procesador de audio incluye el control de limitación de amplitud y el modulador el de desviación de frecuencia. Una vez amplificada y limitada, la señal de audio se aplica al modulador de reactancia, donde se verifica el proceso de modulación de frecuencia anteriormente descrito.

La señal de salida del modulador se aplica a varias etapas multiplicadoras de RF, las cuales desplazan la frecuencia de la portadora hasta obtener la frecuencia final deseada. Una vez obtenida, esta señal se inyecta a los amplificadores de R.F. y se irradia al exterior a través de la antena.

3.6 RECEPTOR DE FM.

En la Figura 17. se muestra el diagrama de bloques de un receptor superheterodino de FM. El sistema consta básicamente de un amplificador de RF, un mezclador, un oscilador local, etapas amplificadoras de FI, un detector o demodulador, un circuito de control automático de frecuencia (CAF) y un amplificador de audio.

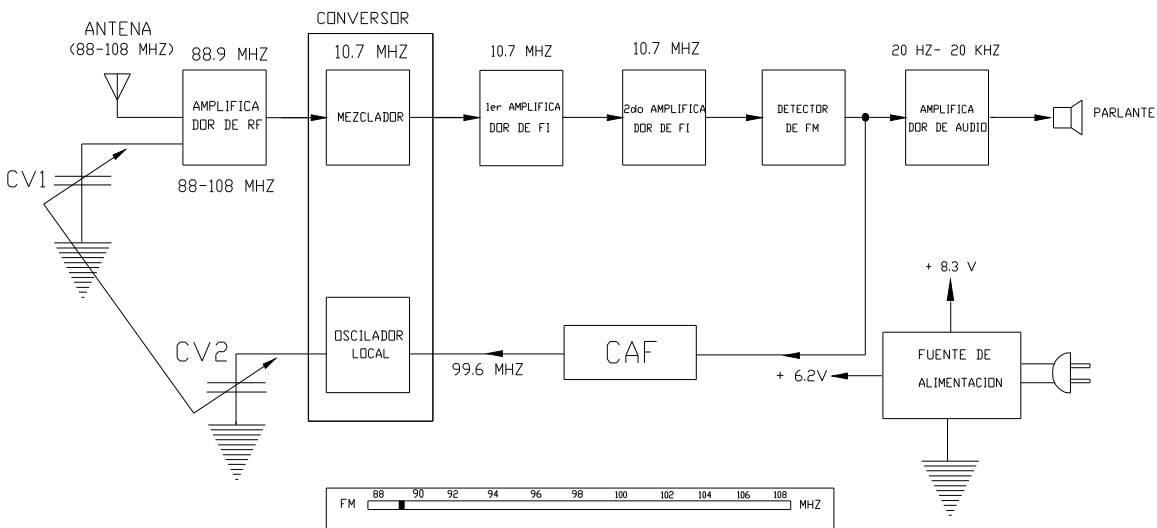


Figura 17. Receptor típico de F.M

Asociado al amplificador de RF esta el circuito de antena. El amplificador de audio maneja el parlante. El mezclador y el oscilador local configuran el convertidor de FI. El receptor utiliza tensiones de alimentación de CC de 8 y 12 voltios extraídos de una fuente regulada, por diseño viene con paquete de baterías recargables.

La frecuencia intermedia del proceso es 10.7 Mhz. El amplificador de R.F. es sintonizado entre 144 y 148 Mhz para radio de 2 metros y el oscilador local entre 154.7

y 158.7 Mhz. El amplificador de audio provee amplificación de señales de baja frecuencia entre 20Hz y 20KHz.

Después de ésta descripción general a continuación se explica como opera el receptor.

Para simplificar la explicación supongamos que el circuito de sintonía esta ajustado para captar la señal enviada por una estación de FM que transmite con una frecuencia de portadora de 144Mhz. En consecuencia, la frecuencia del oscilador local es automáticamente, $146+10.7=156.7$ Mhz.

A la antena convergen las señales provenientes de las diferentes estaciones del área. El circuito de sintonía asociado al amplificador de RF se encarga de seleccionar únicamente la de 146 Mhz e ignora las demás. La señal elegida es entonces amplificada y se inyecta al mezclador.

En el mezclador, la señal modulada de 146 Mhz, proveniente del amplificador de R.F., se mezcla o heterodina con la de 154 Mhz, proveniente del oscilador local.

A la salida del mezclador o conversor se obtiene una señal modulada de F.I. de 107 Mhz. Los demás productos de mezcla son ignorados.

En la figura 18. se ilustra gráficamente el proceso de mezcla. El valor de 10.7 Mhz utilizado como F.I. en nuestro receptor estándar o normalizado en radiodifusión F.M. En radiodifusión AM, la F.I. es de 455 KHz.

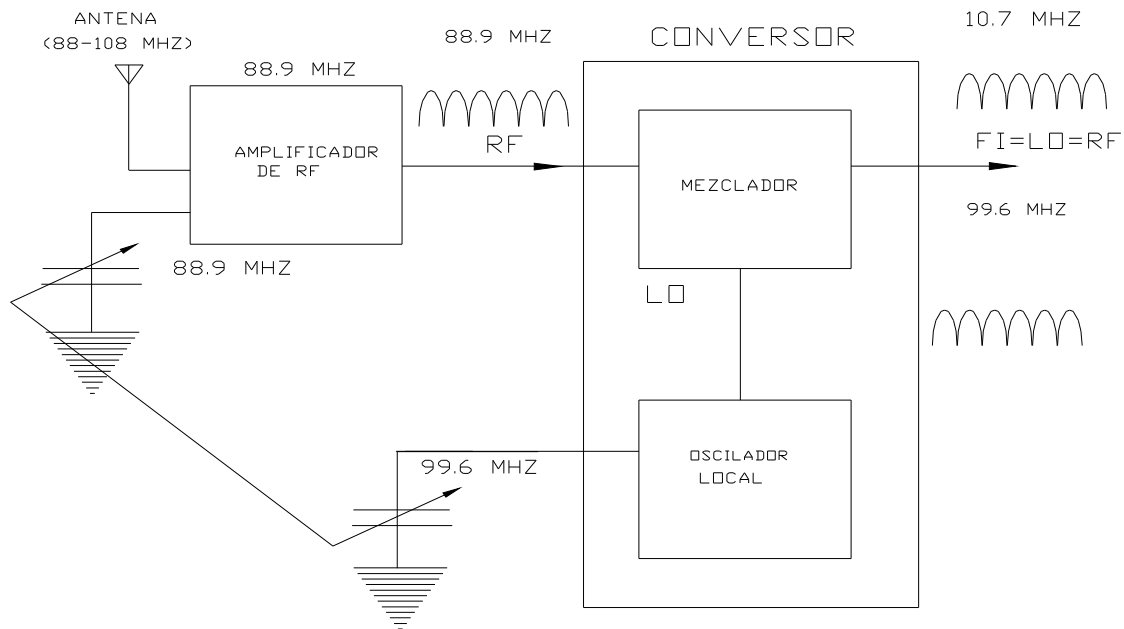


Figura 18. Mezclador.

La frecuencia intermedia (FI) de 10.7 Mhz resulta de la diferencia entre la frecuencia del oscilador local y la frecuencia de la portadora de RF sintonizada. Es decir:

$$FI = LO - RF = 154.7 - 144 = 10.7 \text{ MHz.}$$

En síntesis, el efecto neto del proceso de mezcla es trasladar la modulación original de la portadora (144 Mhz) a una portadora más baja (10.7Mhz). La información transportada por ambas señales es la misma. A continuación, la señal de F.I. se amplifica en dos etapas sucesivas, alcanzando un nivel suficiente como para ser detectada.

El detector o demodulador cumple la función de extraer la información de audio implícita en la señal modulada de F.I., al mismo tiempo que se deshace de la portadora de esta última (10.7mhz). El proceso de detección utilizado en F.M. es sustancialmente diferente del empleado en un receptor de AM.

Una vez obtenida la señal de audio a las salida del detector, se envía al amplificador de audio donde alcanza la potencia suficiente como para impulsar el parlante y convertirse en información audible.

En este trabajo se aprovecha la señal audible (análoga) antes de llegar al parlante para enviarla al módem el cual se encarga de convertir esta señal a digital.

El propósito del circuito de control automático de frecuencia (CAF) es mantener constante la frecuencia del oscilador local, evitando su desplazamiento. Este control se logra mediante un nivel de CC proveniente de la salida del detector.

Finalmente, el receptor necesita de una fuente de alimentación regulada de CC para permitir que operen las diferentes etapas.

La utilización de una fuente regulada garantiza una reproducción de sonido de excelente calidad, sin ruido, y exenta de zumbidos acorde con las características de la señal de F.M. recibida.

En el anexo Ñ se muestra el plano del transceptor 2AT marca ICOM.

TARJETA MICROCONTROLADORA MIC51

INTRODUCCION

Esta tarjeta fue diseñada con el ánimo de obtener uno de los objetivos trazados en este proyecto como el de almacenar la información proveniente del PC. La idea primordial es poder darle otras aplicaciones al PC después de enviar los mensajes sin afectar la información y asegurar que esta se visualice en la matriz de diodos tal como fue enviada, incluyendo efecto, y forma del carácter ya sea mayúscula o minúscula.

El hardware se basa en el microcontrolador 80C51, una memoria EEPROM 27C64, una memoria RAM HY-6264 principalmente, y el software fue diseñado y almacenado en memoria EEPROM para cumplir especialmente las siguientes funciones:

- Visualizar el mensaje C.U.T.B. permanentemente
- Recibir la información o caracteres por el puerto serial.
- Interpretar esta información y convertirla al código ASCII
- Manejar la información y enviarla a memoria RAM
- Manejar el software correspondiente al efecto enviado. Este puede ser de desplazamiento hacia la derecha o hacia la izquierda, de efecto subir o bajar, de

parpadeo por último efecto normal. Cualquiera de ellos se selecciona con el primer carácter (en mayúscula) que se envía desde el PC.

- Inicializa variables y programa temporizadores etc.

La tarjeta MIC51 recibe la información por un terminal serial DB9 del módem receptor y por el puerto 1 del microprocesador sale la información hacia la tarjeta del display para ser visualizada.

A continuación se dará inicio al estudio de hardware y del diagrama de flujo del programa. El listado del programa se puede observar en el capítulo 7.

6.2 HARDWARE

El sistema de la tarjeta MIC51 ha sido desarrollado con unas características tales que puedan permitir el manejo, del puerto serie así como la accesibilidad a memoria de programa y datos de una maneja ágil y eficaz. Se implementa arquitectura HARVAR: espacio separado para memoria de programa y para memoria de datos. En el anexo Q se observa el diagrama esquemático de la tarjeta.

La siguiente es la configuración del sistema de desarrollo

- IC1, Microcontrolador 80C51
- IC2, Latch 74LS373
- IC3, EPROM 27C64

- C4, RAM 6264
- IC5, Transmisor/Receptor Serial ICL232
- B1, Interface DB9 (Puerto paralelo)
- CN1, Interface DB9 (Puerto serie)
- XTAL, Cristal de 11.0592Mhz
- Batería de Ni/Cd de 3.6 vdc
- JP1, Jumper para borrado de la memoria.
- RST, Pulsador de reset

6.2.1 Descripción general del sistema MIC51.

4.2.1.1 Microcontrolador 80C51. El uso de los microprocesadores se ha extendido de tal manera a lo largo y ancho del mundo, que ya es casi imposible encontrar un artefacto electrónico que no disponga en su interior de algún sistema basado en ellos.

La gran versatilidad de los microprocesadores, y su facilidad para cambiar la aplicación a la cual están destinados, con mínimas modificaciones al *hardware*, son en gran parte, las principales responsables de tal popularidad.

Pero aún con sus grandes ventajas, un sistema tradicional basado en microprocesador puede resultar inadecuado en algunas aplicaciones, principalmente debido a su necesidad de utilizar circuitos integrados de apoyo adicionales, tales como memorias y puertos de entrada/ salida.

Dichos circuitos de apoyo significan un aumento en la complejidad y costo de los sistemas, además del mayor espacio físico utilizado. Por esta razón, los fabricantes de microprocesadores han hecho un esfuerzo considerable por desarrollar dispositivos que eliminen estos inconvenientes y permiten disponer de las ventajas del microprocesador en una gran cantidad de aparatos domésticos e industriales.

El resultado ha sido el *microcontrolador*, un dispositivo que incluye, en un mismo circuito integrado, la unidad central de proceso (CPU), memorias RAM y ROM, y puertos de entrada/salida, así como circuitos temporizadores, contadores e incluso, convertidores A/D.

Tal integración permite utilizar el microcontrolador en una gran variedad de artefactos de uso cotidiano: alarmas, sintetizadores musicales, relojes, videograbadoras, hornos microondas etc. También representan, por supuesto, un gran avance en el diseño de sistemas industriales para control automático de procesos.

El microcontrolador de 8 bits 80C51 de la firma Intel, se trata de un microcontrolador optimizado para aplicaciones de control, el cual incorpora, en una misma pastilla de 40 pines, una ROM (o EPROM) de 4 kilobytes, una RAM de 128 bytes, dos temporizadores/contadores de 16 bits y cuatro puertos de entrada/salida. Uno de estos puertos (el #1) puede funcionar alternativamente como canal para comunicación serie.

El 80C51 posee además un conjunto de 111 instrucciones diferentes, facilidades para el tratamiento de interrupciones y capacidad para manejar externamente 64 Kilobytes de memoria de programa y 64 Kilobytes de memoria de datos. En el anexo L se puede apreciar el diagrama de bloques y la configuración de pines del 80C51.

Una de las características importantes del microcontrolador 80C51 es que dispone de cuatro puertos bidireccionales de 8 bits, algunos de los cuales comparten distintas funciones, y otros quedan dedicados a tareas de bus de datos y direcciones cuando se emplea memoria externa.

Con memoria externa, los puertos tienen las siguientes funciones:

- Puerto 0: multiplexa en el tiempo mediante la señal ALE, el bus de datos y la parte baja del bus de DIRECCIONES.
- Puerto 1: Propósito general, y para su uso en el tablero se distribuyó de la siguiente manera.
 - P1.0, P1.6: Maneja las filas del display.
 - P1.7: Pin de conmutación para el display.
- Puerto 2: saca la parte alta del bus de DIRECCIONES.
- Puerto 3: tiene la totalidad de sus patillas compartidas como sigue:
 - P3.0: RxD, entrada de datos del puerto serie.
 - P3.1: TxD, salida de datos del puerto serie. (No se utiliza).
 - P3.2: INT0, interrupción externa 0. (No se utiliza). Pin de RELOJ
 - P3.3: INT1, interrupción externa 1. Pin de salida del DATO serial.
 - P3.4: T0, entrada externa al temporizador/contador 0. (No se utiliza).

- P3.5: T1, entrada externa al temporizador/contador 1. (No se utiliza).
- P3.6: WR, señal de escritura en memoria externa de datos.
- P3.7: RD, señal de lectura en memoria externa de datos.

En el anexo C se muestran la identificación de pines del integrado 80C51

4.2.1.2 Latch 74LS373. Almacena temporalmente la parte baja del bus de direcciones del 8051, con el fin de que el puerto P0 pueda multiplexar la parte baja del bus de DIRECCIONES por el bus DATOS. El latch 74LS373 carga la dirección apuntada por el puerto 0 con la señal de ALE proveniente del 8051 para que el puerto 0 pueda acceder al bus datos sin que se pierda la dirección de memoria. En el anexo G se ve en detalle las características del integrado 74LS373.

4.2.1.3 Memoria EPROM 27C64. En esta memoria reside el programa objeto del sistema MIC51 y tiene una capacidad máxima de almacenamiento de 1Kbyte, todo este espacio es direccionable. En la Figura 19. se muestra el mapa de la memoria 27C64.

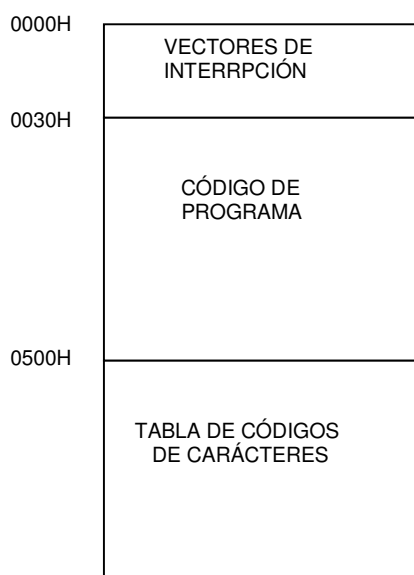


Figura 19. Mapa de la memoria 27C64

En el anexo D se puede apreciar en detalle las características dadas por el fabricante de ésta memoria.

4.2.1.4 Memoria RAM 6264. Esta memoria sirve como unidad de almacenamiento de datos para los mensajes y las zonas de VIDEO y CVIDEO.

Tiene una capacidad de 8Kbytes de los cuales 500 Bytes son utilizados por los mensajes y 128 Bytes por las zonas de VIDEO y CVIDEO, el mapa de memoria esta distribuido según se muestra en la Figura 20. Solo es direccionable hasta 1Kbyte de memoria entre los cuales 396 bytes son utilizables.



Figura 20. Mapa de la memoria RAM 6264

Las características entregadas por el fabricante de esta memoria se muestran en el anexo E.

4.2.1.5 Transmisor/Receptor ICL232. Se emplea el transmisor/receptor ICL232 para adaptar las señales TTL suministradas por el 8051 a los niveles RS232 de la puerta serie del módem. La transmisión se realiza utilizando un solo hilo por señal y un cable de tierra común para todas las señales. La conexión es punto a punto, lo que significa que un transmisor se conecta a un solo receptor.

4.2.1.6 B1, Interface (Puerto paralelo). Este conector exterioriza todos los pines del puerto paralelo P1 al igual que los pines P3.2 y P3.3 del puerto P3 del 80C51. Por medio de esta Interface se comunica la tarjeta MIC51 con el tablero. Es importante señalar que no se hace uso de ninguna norma para la selección de los pines de este conector.

Las patillas que se emplean:

- Pin 1 hasta 7: Manejan las filas del tablero.
- Pin 8: LATCH
- Pin 9: DATO

- Pin 10,11: GND
- Pin 12: CLOCK
- Pin 13, 14: Vcc

4.2.1.7 CN1, Interface DB9 (Puerto serie). Este conector exterioriza las líneas de transmisión y recepción del 80C51 una vez pasados por el transmisor/receptor ICL232 que las convierte en los niveles de tensión adecuados para la comunicación con el módem. De este conector se hacen uso de los siguientes 3 pines.

- Pin 2: RxD recepción de datos
- Pin 3: TxD transmisión de datos.
- Pin 5: GND tierra de señal

4.2.1.8 Frecuencia de operación. El sistema dispone de un cristal de 11.0592Mhz básico para la operación del microcontrolador 8051 y con el cual se pueden obtener las tasas de baudios estándares hasta de 19.200 Baudios, siendo 2400 Baudios la velocidad que se programa para él puerto de comunicación serial RS232 para comunicar la tarjeta con el módem.

4.2.1.9 Batería para el soporte de la memoria. Toda la información que encuentra tanto en la memoria RAM de datos se conserva aunque se desenergice el sistema, debido a la presencia de una batería que la alimenta en el momento de detectarse la ausencia de energía. Esta pila queda alimentando el chip, lo cual nos permite mantener en memoria todos los mensajes.

4.2.1.10 JP1, Jumper para borrado de memoria RAM de datos. Sí por alguna razón se quiere borrar el contenido de la memoria de RAM de datos, solo basta retirar el JP1 que se encuentra dispuesto para tal efecto.

4.2.1.11 RST, Reset del sistema. Este pulsador resetea al microcontrolador 80C51 y permite restablecer el sistema.

4.3.2 Descripción del funcionamiento de la tarjeta MIC51. Para entender claramente el funcionamiento del sistema MIC51 se explica de manera separada cada una de las operaciones que realiza el microcontrolador para efectuar una tarea específica y que en conjunto realizan tareas más complejas, son necesarias para el control de la tarjeta estudiar cada una de ellas

- Arranque del sistema.
- Direccionamiento de memoria externa.
- Lectura de la memoria de programa.
- Acceso a la memoria de datos.
- Transmisión/recepción serial de datos.

4.3.2.1 Arranque del sistema. Todo sistema que posea un microcontrolador requiere para su arranque de un pulso de reset, con el fin de inicializar todas sus variables internas así como también apuntarlo al origen del programa del sistema.

El inicio del programa se ubica siempre a partir de la posición 0000H de la memoria externa o interna y posee la primera instrucción a ejecutar.

Es posible arrancar el sistema ya sea cuando se enciende el equipo o cuando se pulsa el Reset (pulsador RST) como se aprecia en la Figura 21. Para el arranque

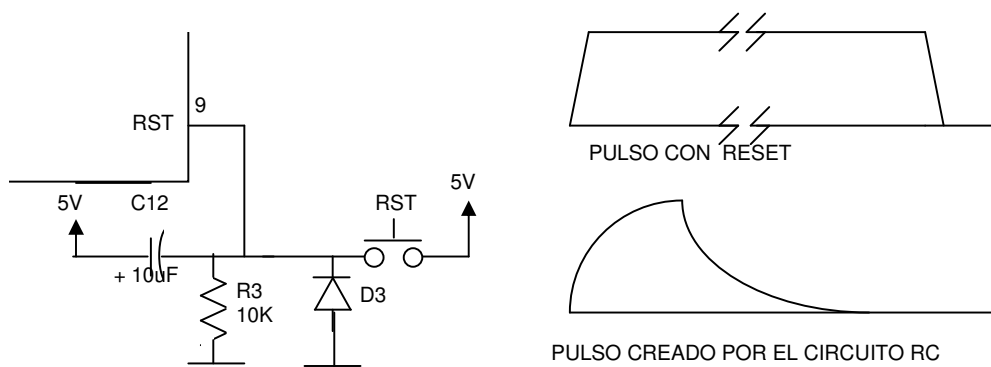


Figura 21. Circuito de restablecimiento del microcontrolador.

con reset, se envía el pin 9 a Vcc a través del pulsador RST. En el caso del encendido, el pulso se crea a partir de un circuito RC, el pulso transitorio así creado tiene el suficiente tiempo para lograr restablecer el procesador después de haber sido energizado.

Una vez inicializado el microcontrolador, está listo para direccionar los buses de datos y direcciones para ejecutar las instrucciones u operaciones indicadas en el código del programa. Todo el proceso de direccionamiento lo realiza internamente el microcontrolador y de su funcionamiento depende básicamente el control de todos los periféricos.

4.3.2.2 Direccionamiento de memoria externa. El microcontrolador 80C51

dispone de cuatro puertos bidireccionales para comunicarse con el exterior, de los cuales dos se dedican a tareas de bus de datos y direcciones cuando se emplea memoria externa u otro tipo de periférico, como los decodificadores de teclados, display, puertos programables, etc. Por medio de los buses, el procesador se puede dirigir a

cualquiera de ellos e intercambiar información, lo cual requiere una completa armonía entre el procesador y todos los periféricos.

La comunicación siempre se realiza entre el microcontrolador y un elemento externo, y nunca involucra a más de dos elementos.

La Figura 22. muestra el diagrama de tiempo de las señales que emplea el

microcontrolador para el acceso a memoria externa o cualquier periférico. Para direccionar a memoria externa, inicialmente el microcontrolador coloca la dirección en la posición a la cual desea acceder para leer o escribir, esto lo realiza con los puertos P0 y P2. Posteriormente envía un pulso por el pin 30 correspondiente a la señal ALE (address latch enable), con el propósito de cargar la parte baja del bus de direcciones en el LATCH 74LS373 para mantener a través de este la dirección. Una vez realizada esta operación, el puerto 0 multiplexa el bus de direcciones por el bus datos y queda listo para leer la instrucción o dato u escribir en memoria externa o cualquier otro periférico.

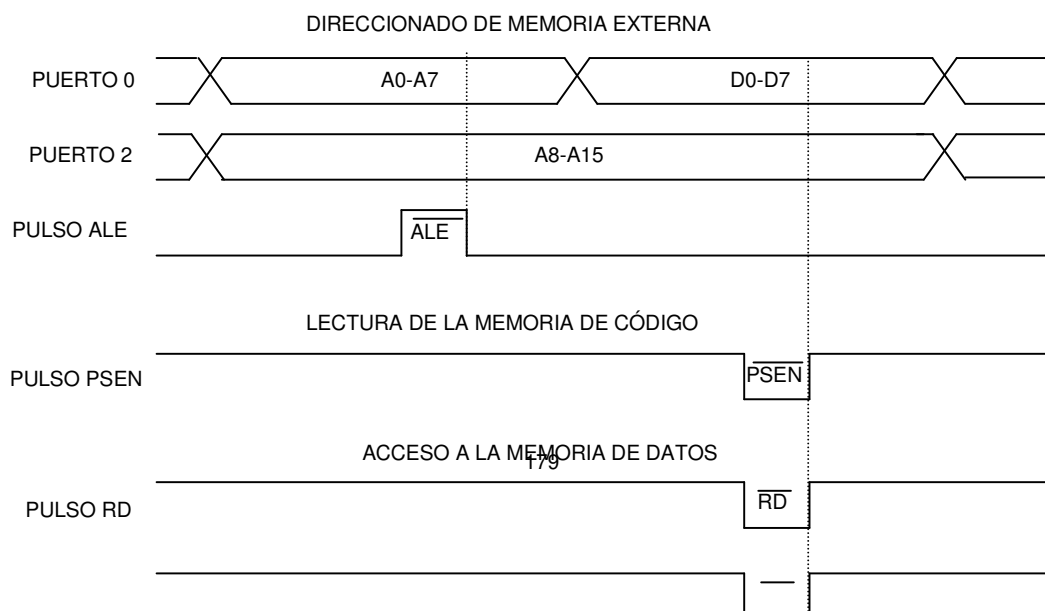


Figura 22. Diagrama de tiempo para acceso a memoria externa

4.3.2.3 Lectura de memoria de programa. Una vez el microcontrolador a direccionado la posición a la cual quiere acceder, este puede leer cualquier dato de la memoria de programa o la memoria de datos.

La Figura 23. muestra que para leer la memoria de programa (EPROM), el microcontrolador pone en bajo un pulso en el pin 29 correspondiente a la señal PSEN (program store enable input), y activa el pin OE (output enable) de la memoria EPROM, con el propósito de habilita sus salidas. La información sacada de la memoria es volcada al bus de datos y leída por el procesador.

La señal PSEN se activa cada vez que se lee una instrucción o cuando se emplea la instrucción MOVC para leer un dató almacenado en la memoria de programa.

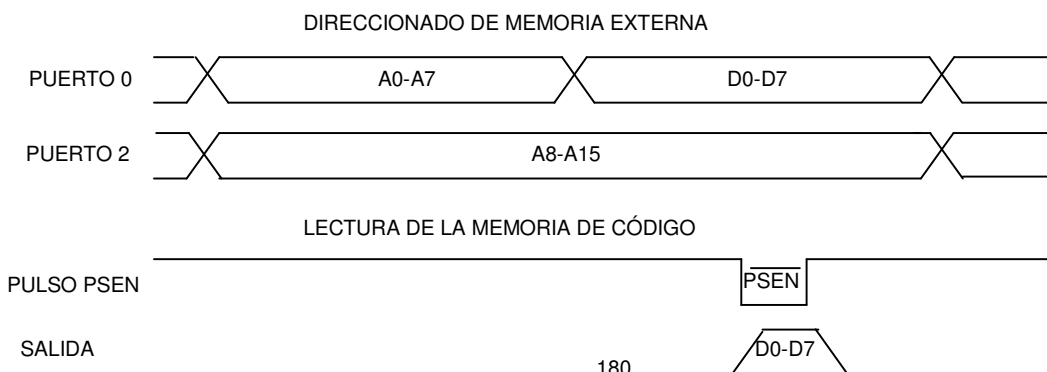


Figura 23. Diagrama de tiempo para acceso a memoria de programa

4.3.2.4 Acceso a la memoria de datos. La manera de acceder a la memoria de datos es similar a la de memoria de programa como se puede ver la Figura 22., la diferencia radica en este caso en que las salidas de la memoria RAM son habilitadas por la señal RD activa en bajo proveniente del pin 17 del microcontrolador. El dato es sacado al bus de dato y leído por el microcontrolador, para leer un dato de la memoria RAM se utiliza la instrucción `MOVX A, @DPTR`.

La otra operación que puede realizar el microcontrolador es la almacenar un dato en la memoria RAM, y se lleva a cabo cuando se ejecuta la instrucción `MOVX @DPTR, A`. De esta forma se consigue poner la señal WD en bajo del pin 16 del procesador, la cual activa la entrada WE (write enable) de la memoria RAM.

Estando habilitada para escritura la información que se encuentra en el bus de dato es escrito y retenido en la memoria.

4.3.2.5 Transmisión/recepción serial. El 8051 está equipado con hardware

interno para comunicaciones serie, que le permite recibir y transmitir datos simultáneamente << full dúplex >>. Los accesos a los registros de recepción y transmisión se realizan mediante el registro SBUF. Al escribir SBUF se carga el registro de transmisión, y al leer SBUF se accede al registro de recepción.

El puerto serie puede operar en cuatro modos diferentes y son programados por software. El sistema MIC51 opera en el modo 2 y el protocolo que sigue es de 1 bit de star, 9 bits de datos, 1 bit de stop. La velocidad de operación es a 2.400 Baudios.

Para enviar o recibir un dato serial el microcontrolador no requiere direccionar los puertos del bus de dato y direcciones, pues esta operación no involucra a los puertos paralelos, en la Figura 24. se aprecia el diagrama de tiempo para la comunicación serial.

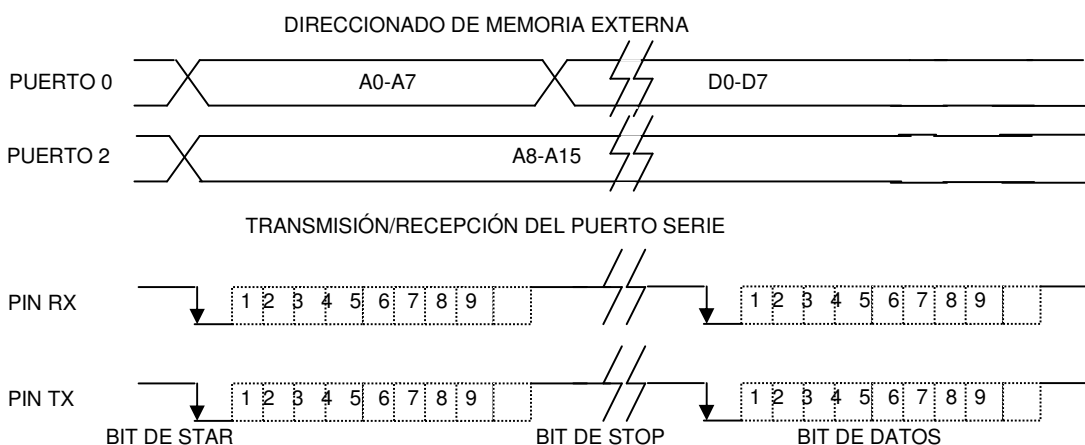
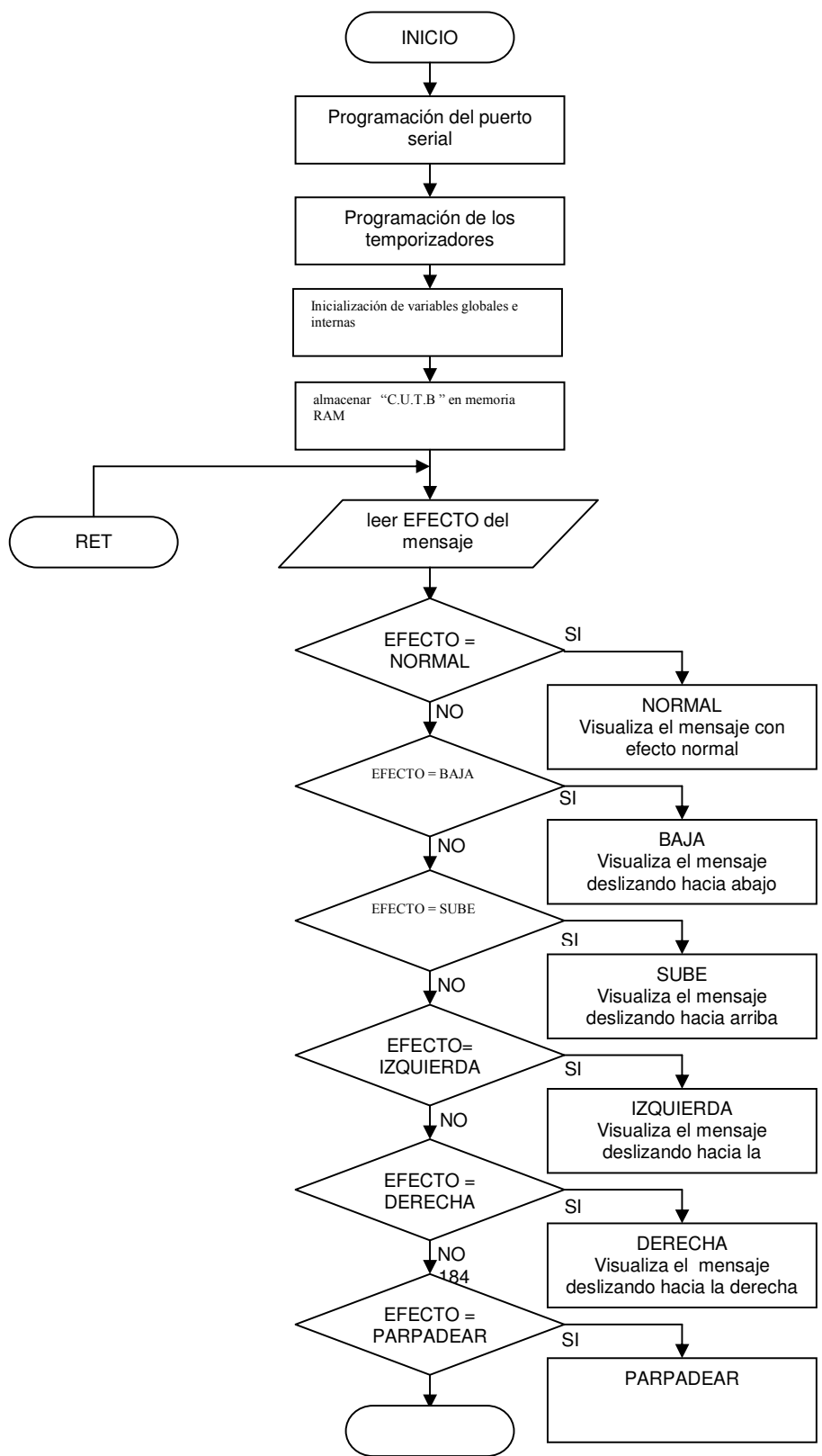


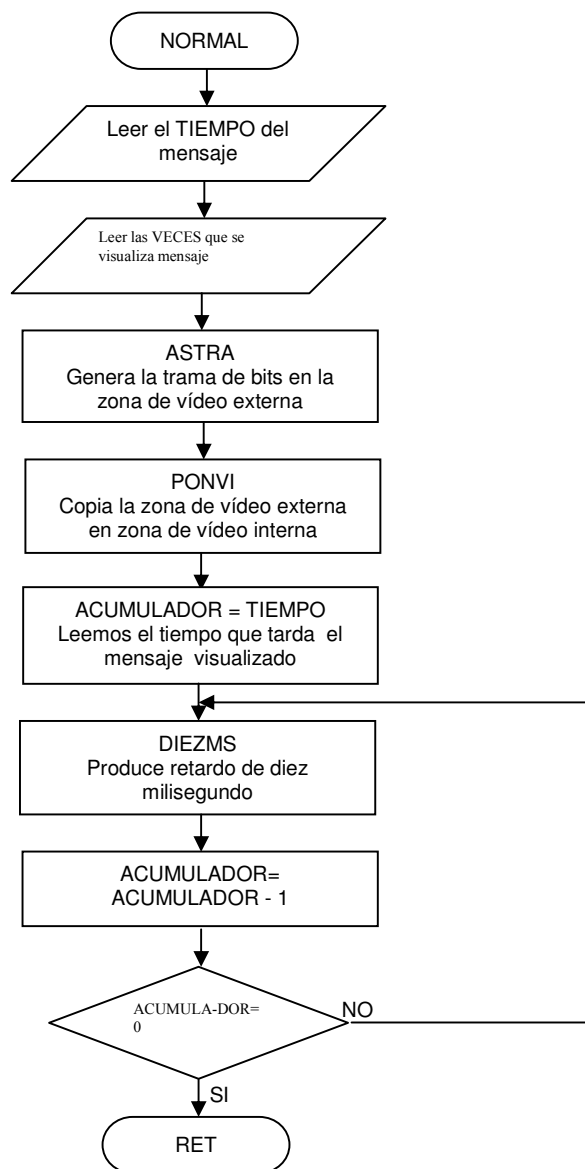
Figura 24. Diagrama de tiempo para la comunicación serial.

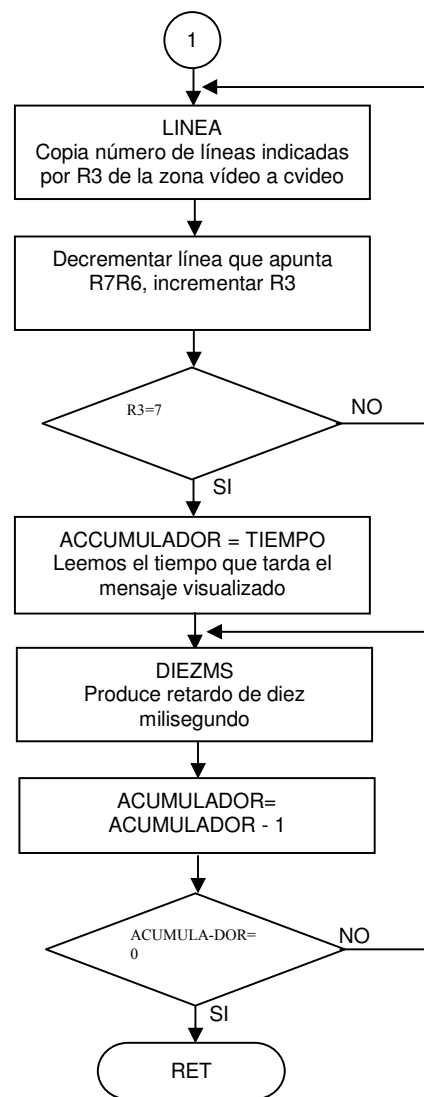
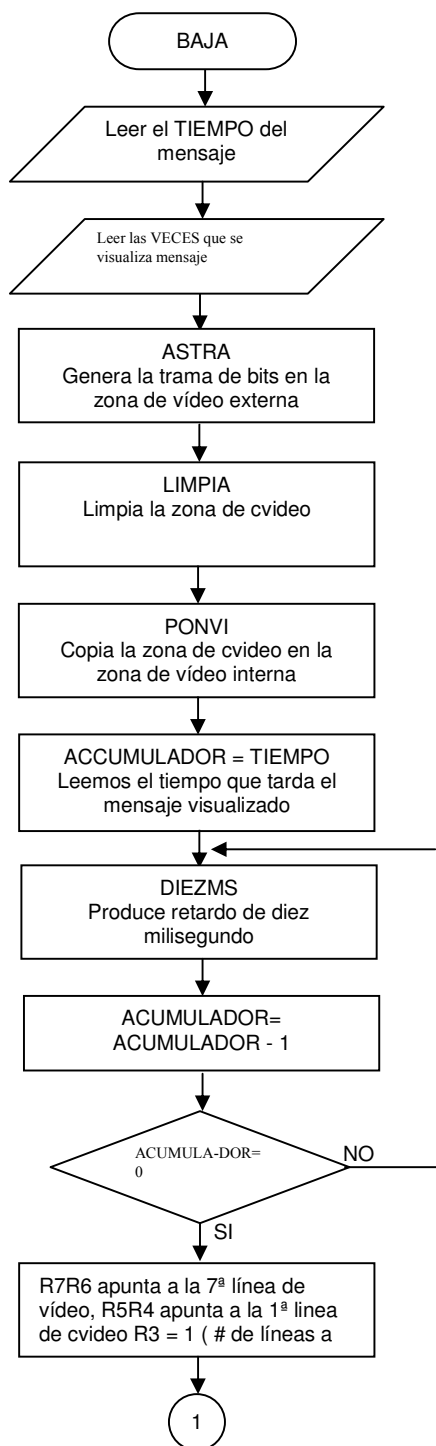
4.4 SOFTWARE

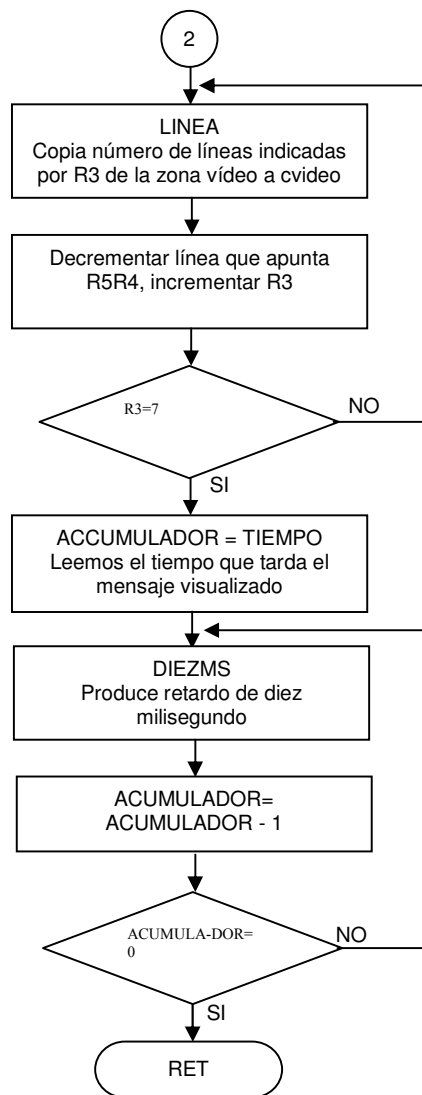
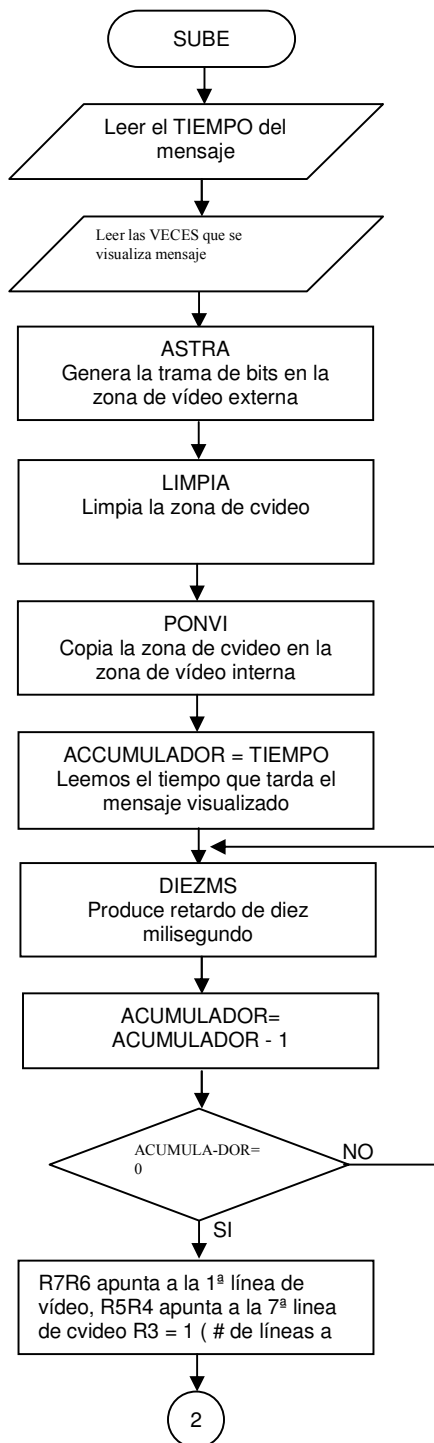
Es esta sección mostraremos el diagrama de flujo simplificado del software utilizado en la tarjeta para ilustrar un poco el funcionamiento de este, no se entra en detalles por la dificultad que representa explicar exactamente los procedimientos matemáticos y lógicos se emplea para procesar la información.

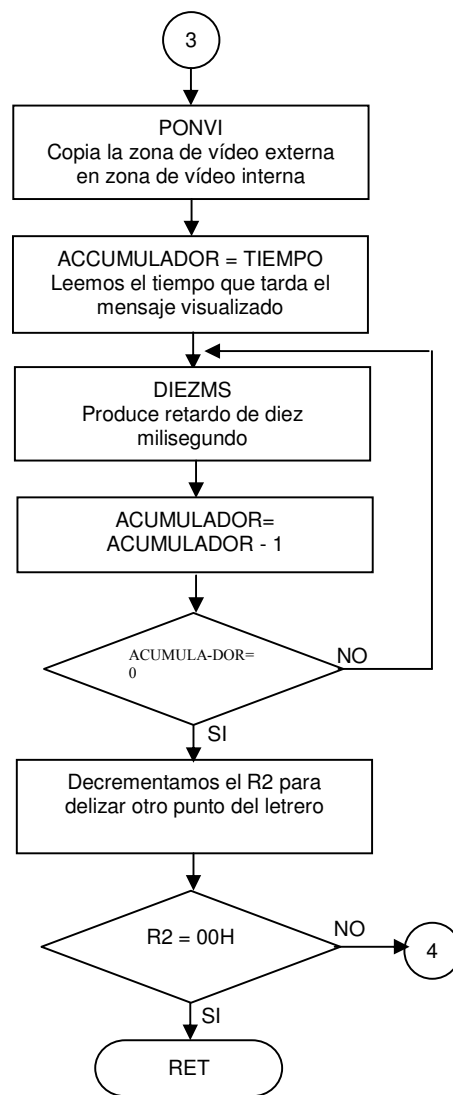
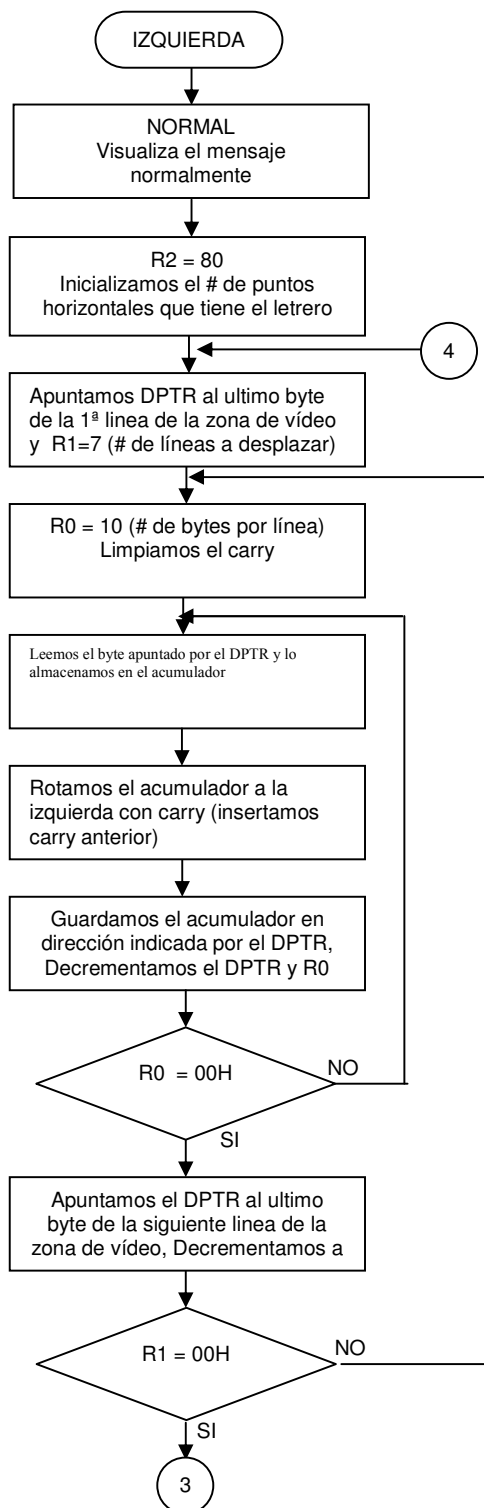
Para mayor información sobre el listado del programa ver el capítulo 7.

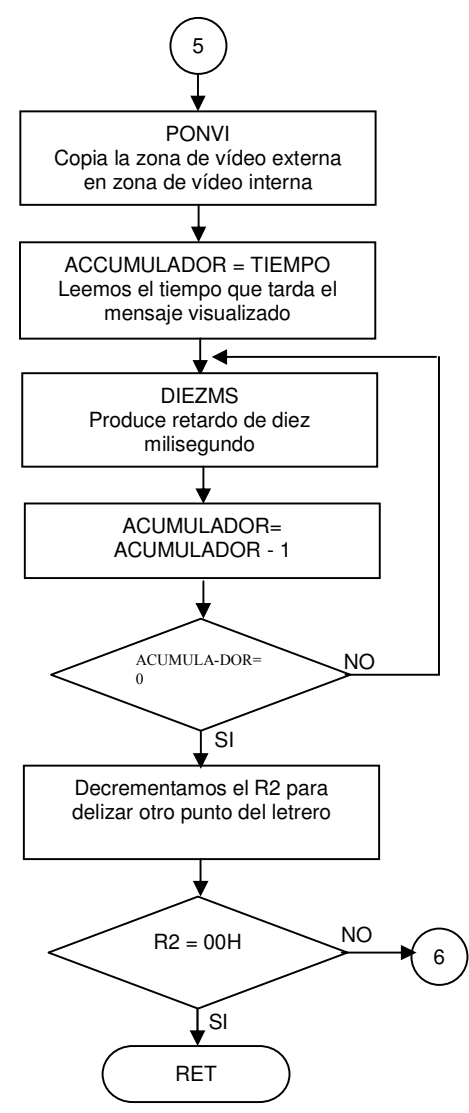
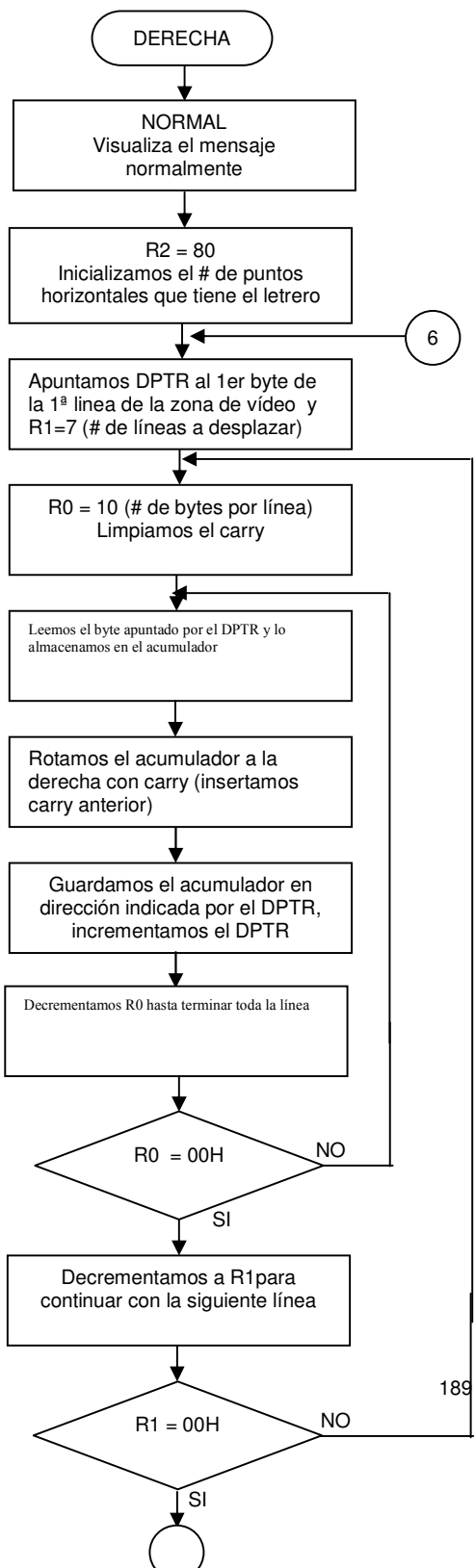


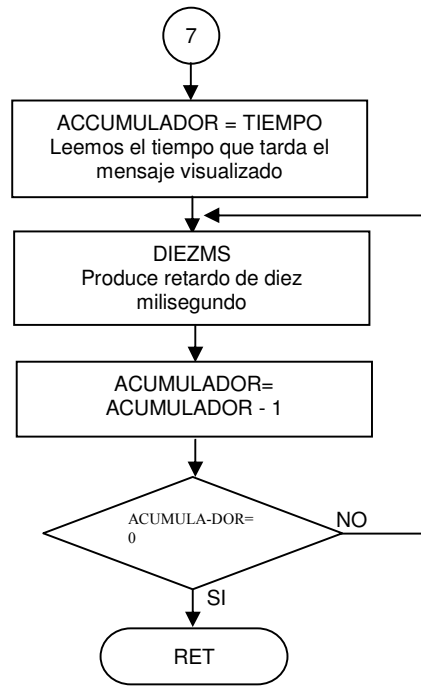
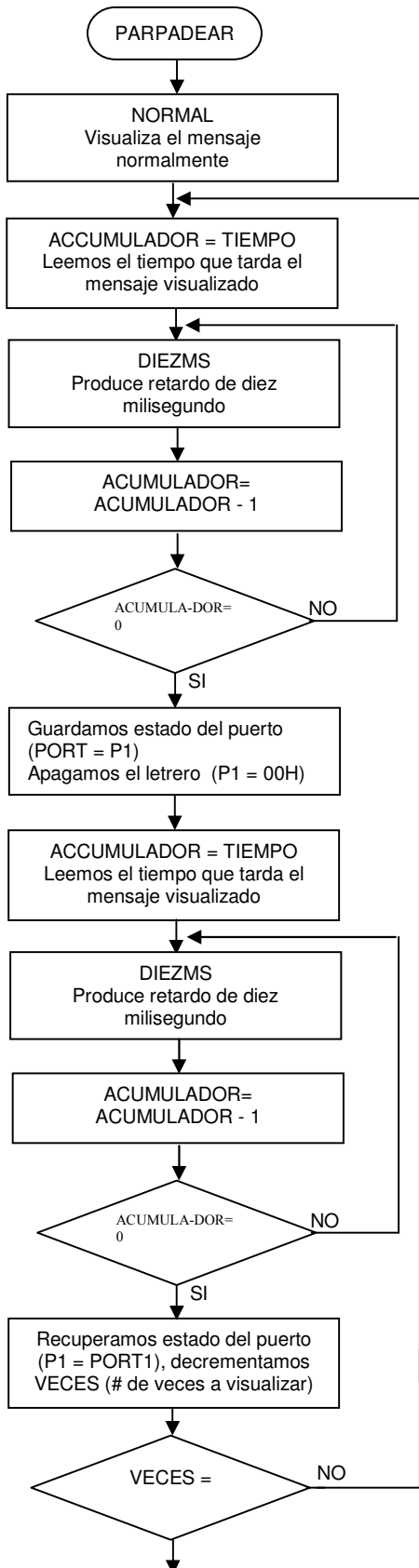


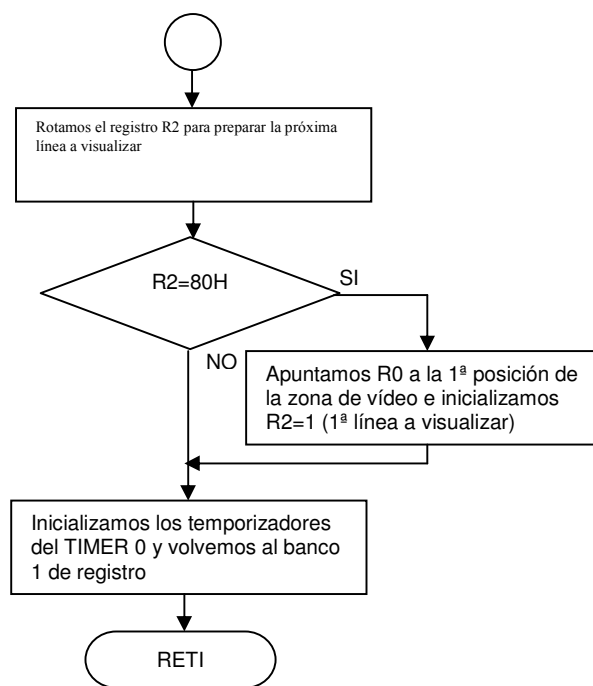
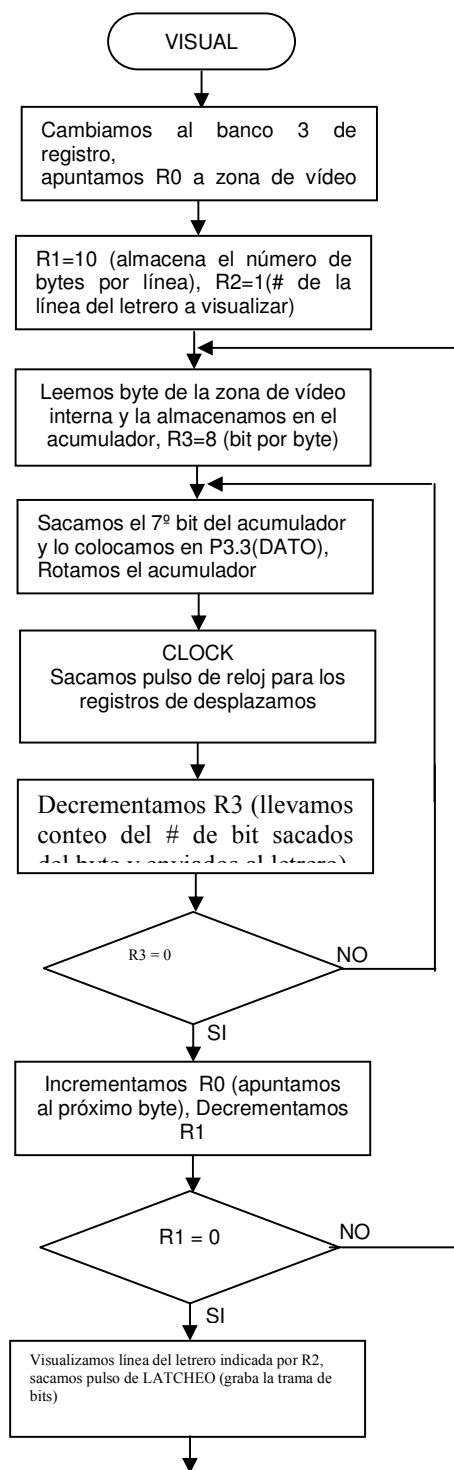


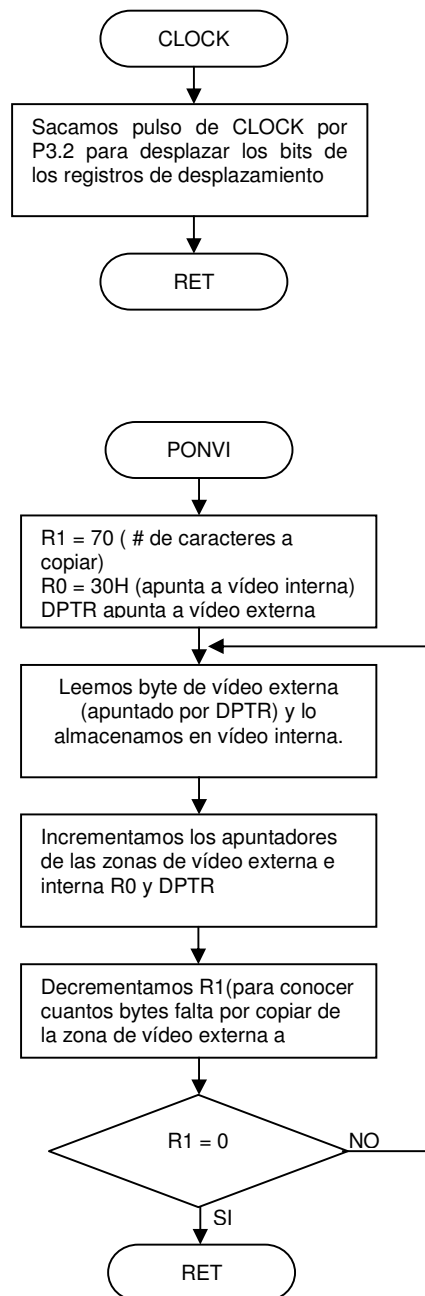


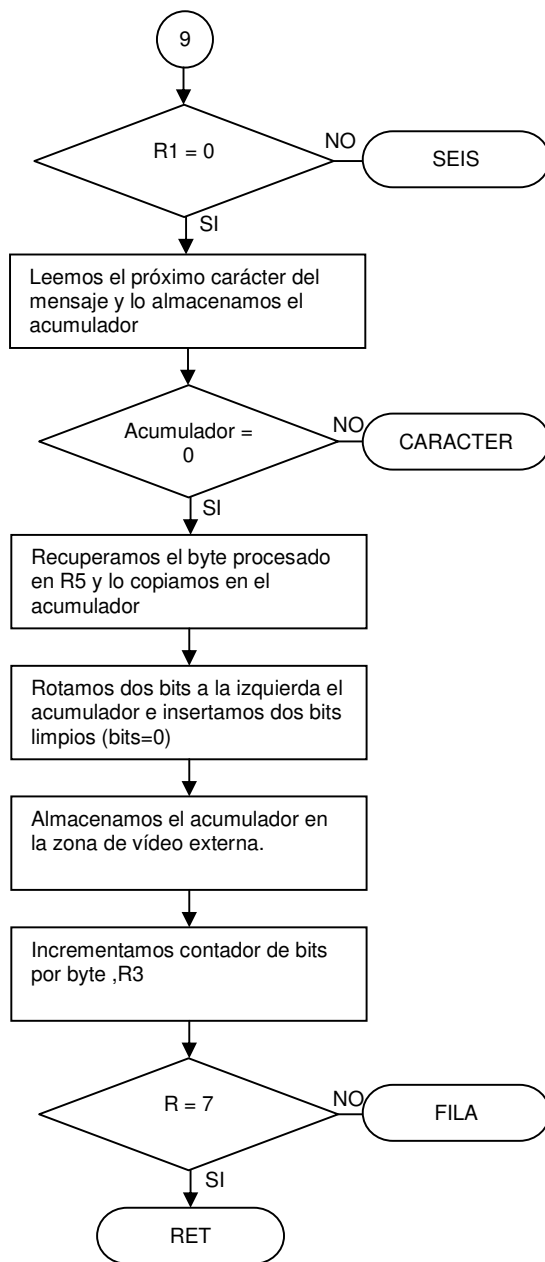
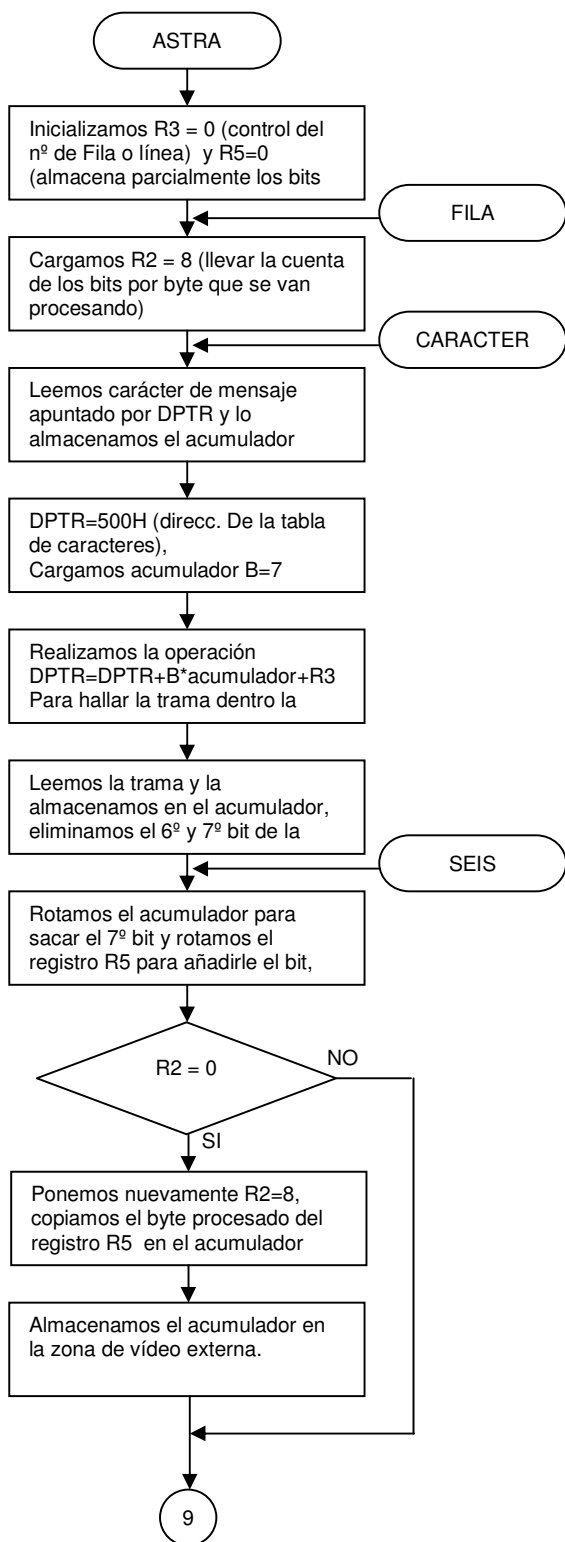


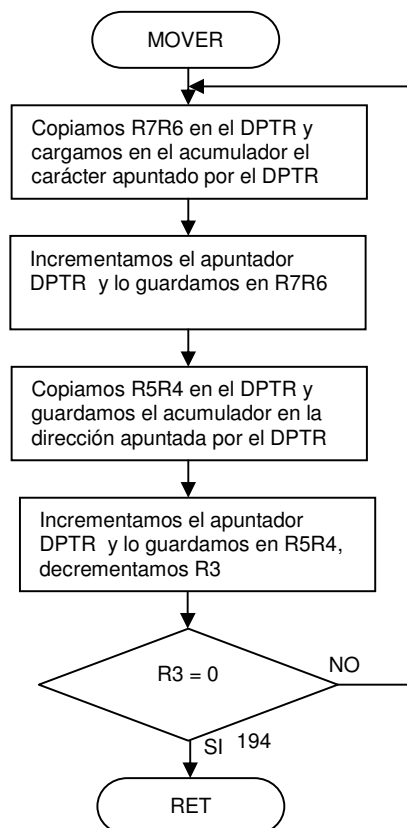
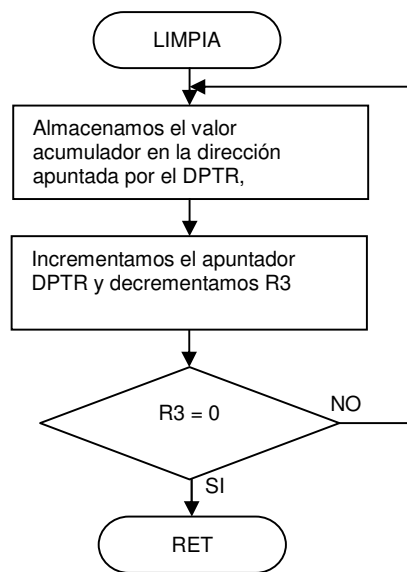


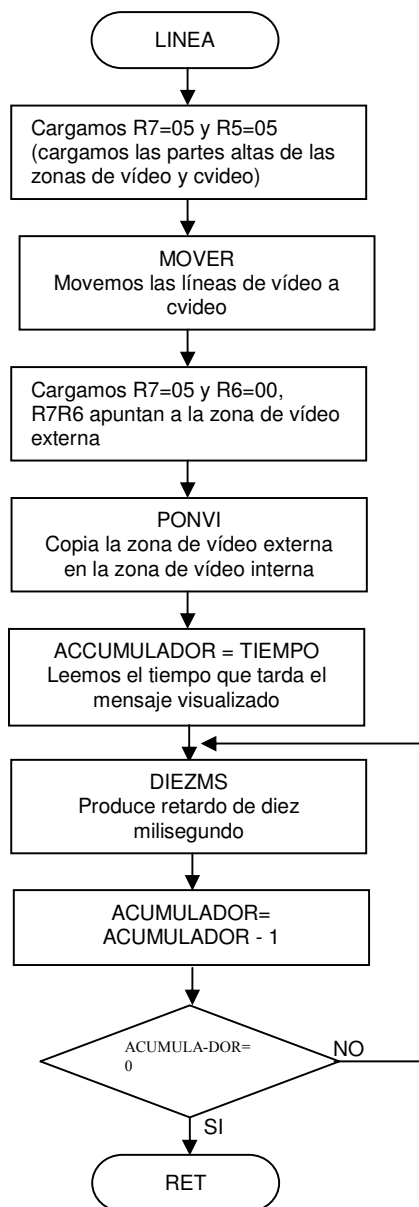


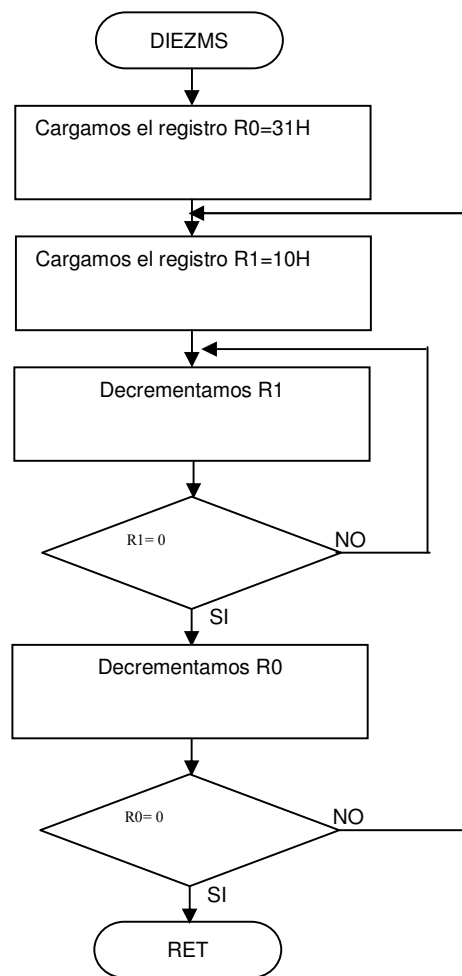












5 EL DISPLAY

5.1 INTRODUCCIÓN

Llamado también matriz de diodos, es la terminal donde se visualizan los mensajes transmitidos, desde el teclado del PC.

Para este fin se utilizaron matrices de diodos Led's, con los cuales se construyen los famosos avisos publicitarios para talleres y negocios que suelen llamarse public's.

El tablero del display esta formado por cuatro tarjetas seguidas una de otra y su información va enlazada por medio de un cable/terminal ribbon como se muestra en la Figura 26. dando una longitud de aproximadamente 1.20m.

En el anexo P, se encuentra el plano de conexionado de las tarjetas del display.

5.2 TARJETA DEL DISPLAY

A continuación se describe el panel donde se visualiza el mensaje; utilizando un diagrama de bloques de describen cada una de sus partes. Ver Figura 26. Este diagrama a bloques representa la tarjeta #1 del display, esto quiere decir que las otras tres tarjetas tienen los mismos elementos y etapas.

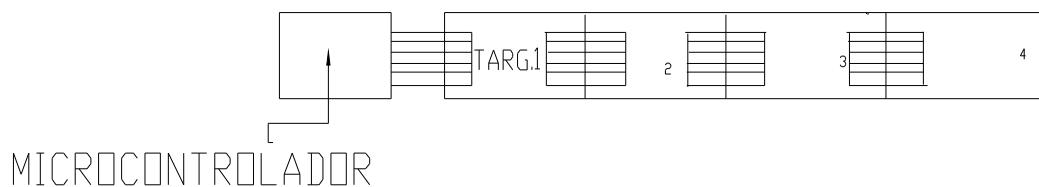


Figura 25. Vista posterior de las tarjetas del display

5.2.1 Bloque 1. Corresponde a los integrado IC-1, IC-2, IC-3 cuya referencia TTL es 74LS164 que es un chip de 14 pines y trabaja como Shift registrador de 8 bit de entrada serial y salida paralelo. La función de este integrado es recibir el dato por los pines 1 y 2 en forma serial proveniente del pin 9 de la bornera de la tarjeta microcontroladora o su equivalente del pin 13 del 80C51 correspondiente al puerto 3 (P3, 3) identificado con el mnemónico INT1 (*Interrupción 1*). Por los pines 1 y 2 se reciben los datos en código de máquina correspondiente a cada una de las columnas de los caracteres a visualizar

Por el pin 8 recibe la señal de reloj proveniente de la tarjeta microcontroladora (pin 12 del integrado IC-8051) correspondiente a la señal INTO (*interrupción 0*) necesario para hacer el desplazamiento de la información de entrada serial (pin 1 y 2) y fijarla a la salida en forma paralelo por los pines 13, 12, 11, 10, 6, 5, 4, 3

Sincronizadamente por el puerto 1 del microcontrolador 80C51 (P1.0, P1.1, P1.2, P1.3,

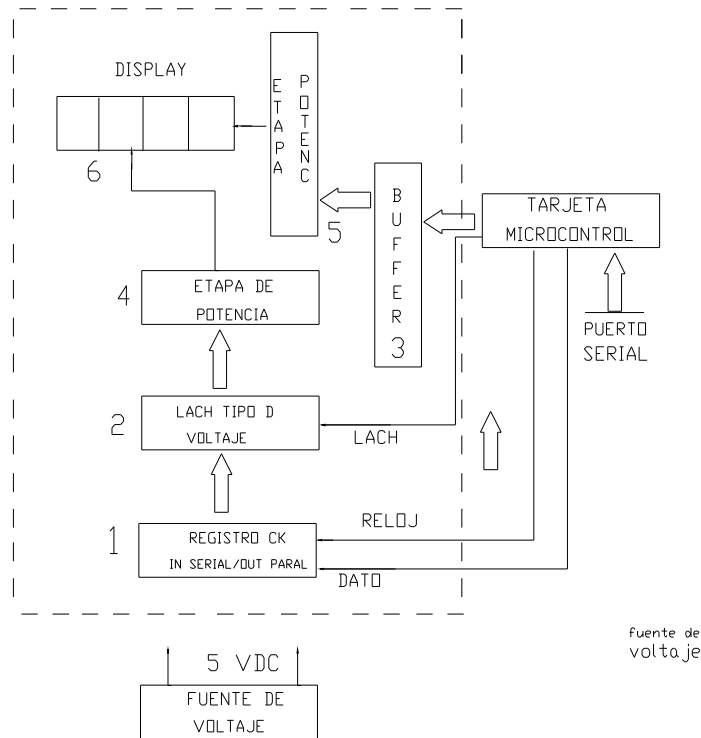


Figura 26. Diagrama a bloques de la tarjeta del display

P1.4, P1.5, P1.6, P1.7) se genera un “1” que va barriendo cada una de las filas del tablero de diodos produciendo el encendido punto a punto de cada uno de los diodos iniciando en la primera fila columna 1 luego fila 1 columna 2 y así sucesivamente hasta barrer toda la fila 1, seguidamente realiza el mismo procedimiento para la fila 2 y así sucesivamente hasta cubrir todo el tablero.

Como se puede apreciar la información que llega por las columnas es la que cierra el circuito para cada uno de los puntos de diodos (ver Figura 27) y de esta manera por ilusión óptica gracias a la frecuencia de barrido se pueden visualizar los diferentes

caracteres. Más adelante se explica con detalle el despliegue de un mensaje en el tablero.

Ver especificaciones del integrado 74LS164 en el anexo F.

5.2.2 Bloque 2. Este bloque corresponde al integrado IC-4, IC-5, IC-6. Chip 74LS373 flip-flops octal tipo D cuya principal función es memoria bit a bit. La información recibida por los pines 3, 4, 7, 8, 13, 14, 17, 18 en forma paralela proveniente del bloque 1.

La información se va a mantener memorizada en los pines de salida 2, 5, 6, 9, 12, 15, 16, 19 y entregada a la etapa de potencia para visualizar en el display cada uno de los "1s" que llegan a las columnas como información correspondiente a los caracteres.

El pin 1 se mantiene en cero lógico (*tierra*) para que las salidas sean controladas por las señales de entrada.

Al pin 11 (*enable*) le llega la señal del puerto 1 P1.7. Este pulso controla el flip-flop, de tal manera que con un "0" mantienen la última información que hay en su salida. Con un "1" permite que la salida sea igual a la entrada.

Ver tabla de función y especificaciones del integrado 74LS373 en el anexo G.

5.2.3 Bloque 3. Esta etapa corresponde al circuito integrado IC-7. Este integrado 74LS244 Buffer es único y se encuentra en la tarjeta #1 del display (ver anexo P, plano A). Este es el encargado de recibir por los pines 2, 4, 6, 8, 11, 13 y 15 las señales del Puerto 1 (P1.0, P1.1.....P1.6 respectivamente) que corresponden a cada una de las 7 filas del carácter a visualizar.

Los pines 18, 16, 14, 12, 9, 7, 5 son las salidas para las filas 1, 2, 3, 4, 5, 6, y 7 respectivamente. Estas señales van a la etapa de potencia para poder visualizar la información en el display.

El pin 19 y el 1 se envían a tierra para permitir que la salida sea igual a la entrada, de lo contrario las salidas se van a impedancia alta.

Ver especificaciones del integrado 74LS244 en el anexo H.

5.2.4 Bloque 4. Corresponde a la Etapa de Potencia utilizada para manejar las columnas. Para esta función se utilizaron transistores ECG 123A NPN, y van del Q1 hasta el Q40.

Estos transistores reciben la señal por base de los integrados que conforman el bloque 2 (74LS323) para poner en corte o saturación cada uno de los transistores que manejan los diodos led's de las 40 columnas ya que sus colectores van polarizados a Vcc (+5 vdc).

5.2.5 Bloque 5: Etapa de potencia es utilizada para manejar las filas del display, corresponden a los transistores Q41 hasta Q54 de referencia ECG-123A NPN. Estos dispositivos reciben por sus bases las señales de salida de IC-4/74LS244, y su salida por emisor polariza los cátodos de los diodos led's correspondientes a las 7 filas de cada carácter ya que sus colectores están conectados a "0" (tierra).

Los elementos de los bloques 4 y 5 correspondientes a las etapas de potencia tienen la función de 2 conmutadores en serie que por medio de las señales de base controlan cada punto de la matriz de led's. En la Figura 27. se observa la forma como se activa un punto de un carácter o letra. Analizando el esquema se concluye que para que se encienda un punto (formado por cuatro led's), se necesita un "1" de la información proveniente de los datos enviados, y en la columna y un "1" proveniente del barrido que realiza el puerto P1 para cada una de las filas.

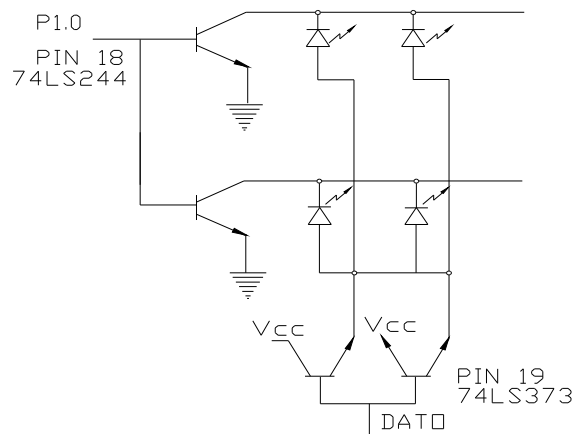


Figura 27. Punto de la matriz de led's

5.2.6. Bloque 6. Corresponde a la pantalla (*display*) y está formado por las matrices de diodos led's que visualizan los caracteres. Estas matrices de diodos son 5x7 (columnas x filas) de cátodo común y de referencia LTP2057AG-NB. Se unieron cuatro de ellas para visualizar una letra o carácter, con el ánimo de ampliar el tamaño de la letra en cuatro veces para visualizarla a una distancia aproximada de 20 metros cumpliendo con esto un objetivo trazado. El tamaño de la cada letra es de aproximadamente 8 x 10 cms.

En el anexo P plano B se observa la distribución de las matrices que se indican con el mnemónico D1 hasta D16. En la Figura 28. se observa una matriz de diodos LTP2057AG-NB con sus respectivos pines identificados.

En la Figura 29. se observa la representación en cátodo común de los diodos de una matriz de led's

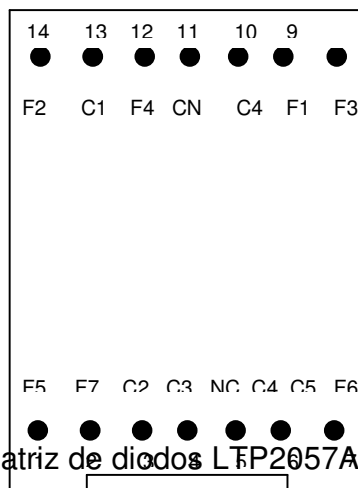


Figura 28. Matriz de diodos LTP2057AG-NB

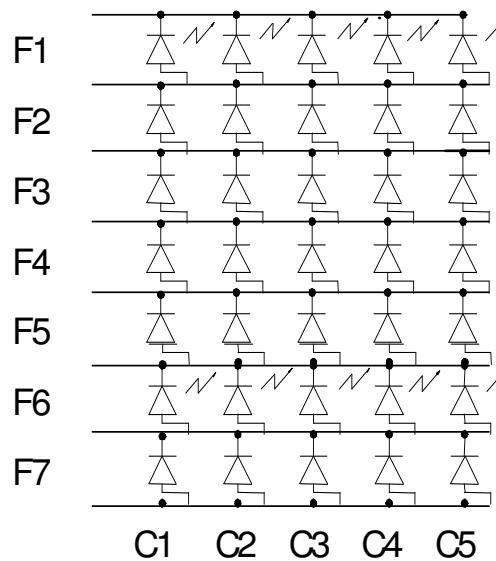


Figura 29. Representación en cátodo común de una matriz de diodos

El diagrama a bloques anteriormente detallado corresponde a cada una de las tarjetas del display. En la Figura 30. se observa como van unidas las tarjetas por medio de un cable/conector ribbon que es el que une las señales seriales y paralelas desde la primera tarjeta hacia la segunda y así sucesivamente hasta la cuarta tarjeta. La bornera B2 de la tarjeta #1 envía las siguientes señales a la bornera B1 de la tarjeta #2:

- Señal serial (*dato*) por el pin 9
- Señales paralelas del puerto P1 por los pines 1, 2, 3, 4, 5, 6, 7
- Señales paralelas de reset y reloj por los pines 8 y 12 respectivamente
- Alimentación de 5 VDC por los pines 10 y 11 para negativo y pines 13 y 14 para positivo.

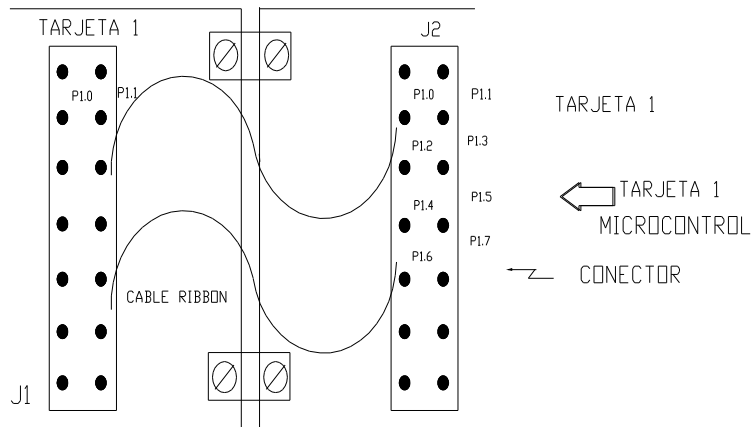


Figura 30. Conexión de tarjetas del display

5.3 COMPONENTES DE LA TARJETA DEL DISPLAY

Los elementos utilizados en una de las tarjetas se listaran a continuación. En cada una de ellas se pueden visualizar cuatro caracteres:

IC1, IC2, IC3-----74LS164

IC4, IC5, IC6-----74LS373

IC7 -----74LS244

Q1,Q2,...Q54-----ECG-123A

C1, C2, C3, C4, C5, C6, C7-----0.1uf

B1 Conector para los datos de entrada

B2 Conector para comunicar la tarjeta siguiente Datos de salida para la tarjeta 2.

D1, D2..... D16: Matrices de diodos led's de color verde de 5x7 con referencia LPT 2057AG-NB de uso comercial con medidas de 3.8 de ancho x 5.3 cm de ancho y de cátodo común.

5.4 CONSTRUCCION DE UN MENSAJE EN EL DISPLAY

El tablero básicamente esta constituido por una gran matriz de diodos de 80x14 led's (un punto se representa por cuatro led's) como se aprecia en la Figura 31. Su tamaño le permite visualizar mensajes que contengan hasta 13 caracteres, puesto que cada carácter ocupa un tamaño de 5x7 puntos y se necesitan de 1x7 puntos para mantenerlo separado de los otros, como se muestra en la Figura 32.

Una imagen en particular se forma cuando se encienden simultáneamente varios puntos del display, cada punto tiene asociado una información que le indica el estado que debe representar, tal información se encuentra almacenado en un bit de la memoria interna del microcontrolador 80C51 y la cual se llamó "*zona de vídeo interna*". Un bit puede almacenar y manejar dos posibles estados, un estado alto o "1" para representar un punto encendido o un estado bajo o "0" para representar un punto apagado.

Una matriz de 560 puntos requiere de 560 bits o de 70 bytes de memoria para guardar la información de la imagen, en la Figura 33. se muestra el mapa de memoria de la zona de vídeo la cual virtualmente representa el tablero y donde cada byte almacena información de 8 puntos de una misma línea del display. Crear una imagen en el letrero

como la que aparece en la Figura 32, necesita tener almacenado en la zona de vídeo la información que aparece en la Figura 34.

5.5 DESPLIEGUE DEL MENSAJE EN EL DISPLAY

Desplegar la información que se tiene en la zona de vídeo a cada uno de los puntos del display requiere de un procedimiento especial, puesto que no se puede acceder a todos ellos a la vez debido a que se poseen pocas líneas. La técnica que se emplea para crear la imagen sin acceder a todos esos puntos requiere de la multiplexación por tiempo, este método permite visualizar una línea a la vez pero a una frecuencia tan alta que el ojo humano no alcanza a notar los cambios.

Con la multiplexación por tiempo se consigue llevar la información a todos los puntos correspondientes a una misma línea para que sean visualizados. Posteriormente durante un tiempo determinado, el tiempo que se usa para visualizar una línea a la vez se emplea para llevarle información a los puntos de la siguiente línea. Una vez cargado los datos, se multiplexa a la línea adyacente para dar paso a la visualización de la misma, en la Figura 35. se muestra al diagrama de tiempo que toma crear una imagen o cuadro.

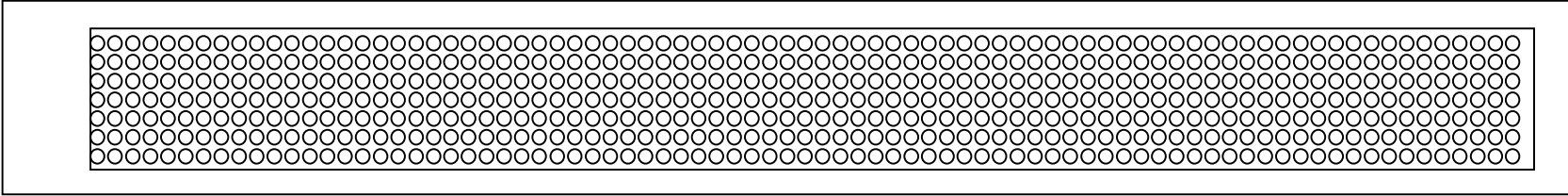


Figura 31. Tablero del display

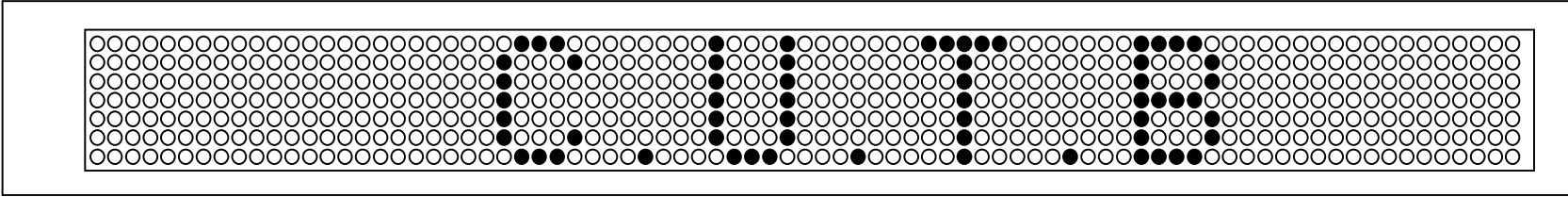


Figura 32. Visualización de mensaje

30H	31H	32H	33H	34H	35H	36H	37H	38H	39H
3AH	3BH	3CH	3DH	3EH	3FH	40H	41H	42H	43H
44H	45H	46H	47H	48H	49H	4AH	4BH	4CH	4DH
4EH	4FH	50H	51H	52H	53H	54H	55H	56H	57H
58H	59H	5AH	5BH	5CH	5DH	5EH	5FH	60H	61H
62H	63H	64H	65H	66H	67H	68H	69H	6AH	6BH
6CH	6DH	6EH	6FH	70H	71H	72H	73H	74H	75H

Figura 33. Zona de Vídeo

00	00	00	E0	11	01	F0	1E	00	00
00	00	01	10	11	00	40	11	00	00
00	00	01	00	11	00	40	11	00	00
00	00	01	00	11	00	40	1E	00	00
00	00	01	00	11	00	40	11	00	00
00	00	01	10	11	00	40	11	00	00
00	00	00	E1	0E	10	41	1E	00	00

Figura 34. Mapa de memoria

El control de la multiplexación y de la carga de datos se hace por software y la secuencia que realiza para crear un cuadro es la siguiente

- Inicialmente se lee el primer byte de la línea de la zona de vídeo. Se procede después a sacar los bits de mayor a menor peso del byte leído y se envían a través del pin 3.3 (DATO) a la entrada del último registro de desplazamiento, por cada bit enviado se manda una señal de reloj por el pin P3.2 (CLOCK) para cargar y correr los bits por los registros. Una vez sacado el byte, se lee la siguiente posición de la zona de vídeo y se continúa con la inserción de los bits en los registros. Esta tarea finaliza cuando se han insertado los 10 bytes de la línea en los registros de desplazamiento.
- Se habilita la línea correspondiente a la información por medio del puerto P1, el cual coloca en nivel alto solo la línea a visualizar.
- La información retenida en los registros se carga en los Latch's por medio del pulso de LATCH, el cual permite que los bits sean almacenados y a la vez sacados hacia el tablero para que sean visualizados.
- Se lee la siguiente línea de la zona de vídeo y se carga en los registros de desplazamiento.
- Se multiplexan las líneas, esto significa que se deshabilita la línea anterior y se habilita la siguiente para su visualización.
- Nuevamente se envía el pulso de LACH para grabar la información en los Latch's para que puedan ser desplegados en el display.

De esta manera se procede con todas las líneas pero a una velocidad elevada, con el fin de crear el efecto visual que muestra toda la imagen. Sabemos que esto es posible por el tiempo de respuesta que presenta el ojo humano para retener una imagen.

Entonces para recrear un cuadro de un mensaje se necesita multiplexar siete veces las líneas de control para ver la información correspondiente en igual número de ocasiones. Este cuadro se repite tantas veces según sea el tiempo definido en el mensaje.

6. OPERACION Y DESCRIPCION DEL SISTEMA

6.1 INTRODUCCION

En este capítulo se describe cada uno de los equipos que componen el sistema de transmisión y recepción de mensajes comprendidos desde el PC hasta el display de diodos. Se describe con detalle cada uno de los equipos referente a su construcción, la implementación de sus respectivos accesorios y conexiones utilizadas, también se incluyen algunas observaciones para su correcto funcionamiento operacional. Se recomienda al usuario leer cada una de estas instrucciones antes de poner en funcionamiento cualquiera de sus partes.

Iniciaremos el estudio de este capítulo describiendo las partes físicas del sistema que contempla tres bloques o módulos en lo referente a la presentación.

6.2 MÓDULOS QUE CONFORMAN EL SISTEMA

6.2.1. El módulo 1 o transmisor digital de RF. Es el módulo encargado de recibir la información desde el puerto serial del PC procesarla y convertirla en señal análoga por medio del módem para poder enviarla por ondas de radio (vía R.F) a través de la antena del transmisor a una frecuencia de 144 MHz.

Este módulo consta de las siguientes partes internas.

- Fuente de suministro de 12 VDC-5A
- Tarjeta Módem PACKET KPC-2400 marca KANTRONICS.
- Radio transceptor de VHF/FM portátil marca ICOM 2AT.

Estos tres dispositivos fueron ensamblados en una caja de uso comercial de color negro con las siguientes medidas A=8cm, B=30cm y C=20 cm y cuyo peso es de aproximadamente 1 Kg.

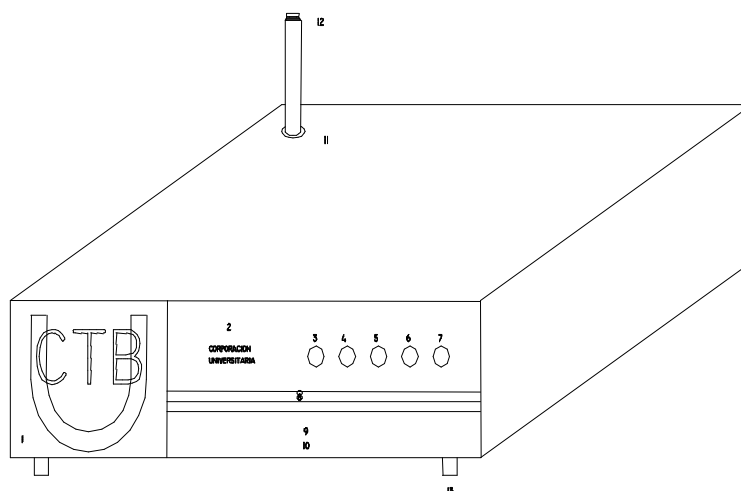


Figura 36. Vista del módulo 1

En la Figura 36. se observa el módulo No.1 y se describe la placa del panel frontal:

- 1- Logotipo de la universidad con sus colores característicos.
- 2- Nombre de la universidad.
- 3- STA: Led de color rojo cuando enciende indica que hay mensajes en espera (STAND- BY).
- 4- CON: Led de color rojo, cuando enciende indica que esta unidad o módulo 1 esta conectado con el módulo 2 (módulo receptor de R.F.) y por lo tanto se puede iniciar el envío de mensajes.
- 5- RCV: Led de color rojo, cuando enciende indica que este módulo esta recibiendo información.
- 6- XMIT: Led de color rojo, cuando enciende indica que este módulo esta enviando información.
- 7- POWER: Led de color verde, cuando enciende indica que la unidad esta energizada con voltaje de 12 VDC. Estos 5 led's van montados en una tarjeta impresa y soportados por una porta led's en el panel frontal.
- 8- Transmisor digital de RF: Es el nombre dado a este módulo porque desde aquí se envía la información al receptor cada vez que el usuario lo requiera.
- 9- Por: GERARDO MEJIA MEJIA: Autor del proyecto.
- 10- Mayo/97: Fecha de entrega del proyecto.
- 11- Conector para la antena de radio tipo BNC que esta en la parte superior de la caja.
- 12- Antena flexible con conector tipo BNC enchufable.

13- Bases de caucho.

En el panel posterior se identifica las siguientes partes. Ver Figura 37.

- 1- Porta-fusible roscable de 1 A.
- 2- Salida de cordón enchufable para 110VAC 60Hz .
- 3- Regulador ECG933 para 12 VDC-5A.
- 4- Conector RS-232 hembra de 9 pines para la conexión tipo seria con el PC.
- 5- Swich on-off para suministro de 110VDC- 60 Hz.

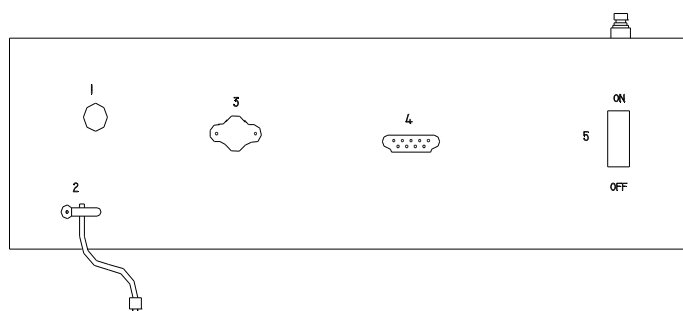


Figura 37. Panel posterior del módulo 1.

A continuación se hará la descripción interior del módulo 1: Ver Figura. 38 donde se detallan las partes internas del módulo 1.

Antes de proceder a quitar la tapa superior cerciórese de que el swich este en off y que el cable de alimentación este desconectado, proceda a quitar los 2 tornillos laterales y los 2 posteriores, levante lentamente la tapa superior gírela hasta quedar en forma vertical y a la vez desplázela hacia el lado izquierdo ya que se une al radio por medio del conector de la antena con una línea de transmisión RG-58 y su respectivo conector BNC, proceda a girar el conector BNC del radio y quedara libre la tapa.

En el interior del módulo se describen los siguientes elementos o dispositivos:

6.2.1.1 Fuente de suministro de 12 VDC-5A

6.2.1.1.1 Transformador marca MAGON Ref: MG. Ver Figura 39.

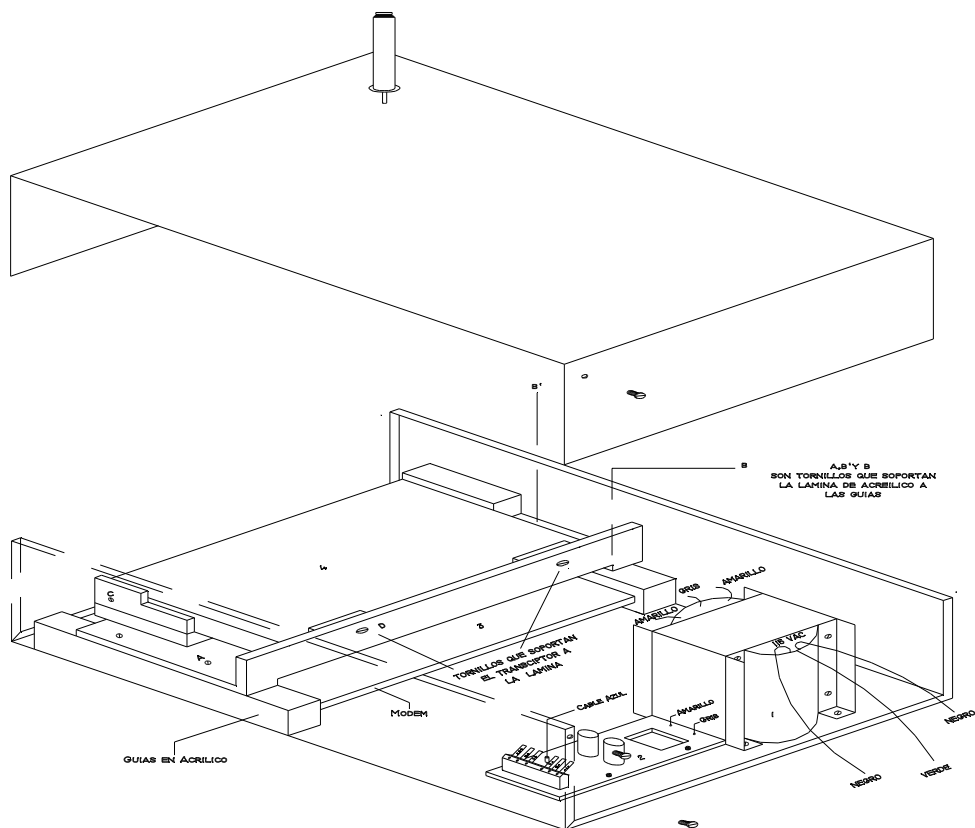


Figura 38. Partes internas del módulo 1

Entrada =115VAC-60Hz. Salida 12VAC/ 24VAC. Potencia 60W. I =5 A

Este transformador esta sujeto con 4 tornillos a la base. Los cables utilizados para la conexión son:

Negro - negro del lado primario para 115VAC-60 Hz.

Amarillo - gris del lado secundario para 12 VAC

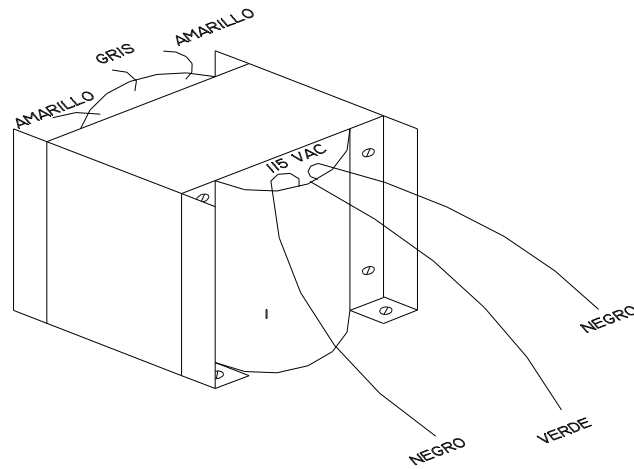


Figura 39. Transformador

6.2.1.1.2. Tarjeta rectificadora. Recibe la señal de 12VAC proveniente del transformador, cuyas líneas son amarillo y gris aquí es rectificada por medio de un puente de 6A y filtrada con un capacitor de 6600 MF/50V. Contiene la bornera de conexionado para alimentar el transeptor y la tarjeta módem a 12VDC. Esta tensión viene del regulador EGG-933 que esta ubicado en el panel posterior de la caja, por medio del cable rojo(+) y negro(-). Ver Figura 40. donde se observa la tarjeta rectificadora de la fuente.

Esta tarjeta esta sujeta a la tapa inferior o base de la caja por cuatro tornillos con sus respectivas tuercas.

En la bornera de conexionado se identifican los cables con las siguientes marquillas:

+ + + : Cable de color rojo correspondiente a la salida del regulador ECG-933. Es el positivo de la fuente de suministro +12VDC.

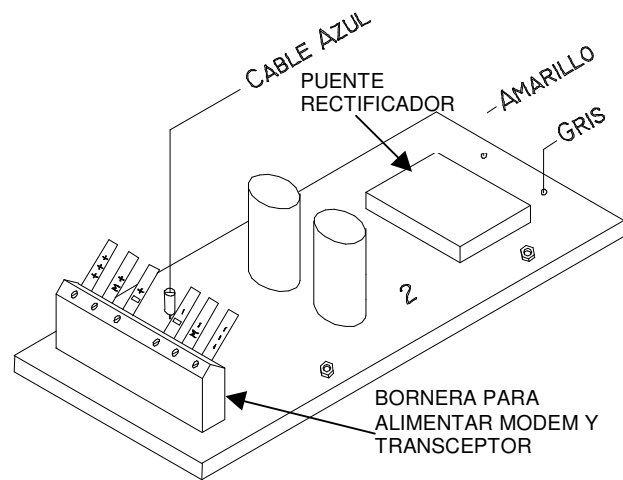


Figura 41. Tarjeta rectificadora

- - - : Cable de color negro corresponde al CASE o carcaza del regulador. Es el negativo de la fuente del suministro.

M + : Cable de color negro/rojo es el positivo para la alimentación del módem a 12VDC.

M - : Cable de color negro es el negativo para la alimentación del módem.

Esta alimentación va por medio de un plug enchufable a la tarjeta módem. Ver. Figura41.

0 + : Cable de color negro/rojo es el positivo para la alimentación a 12 VDC del transmisor.

0 - : Cable de color negro es el negativo para la alimentación del transmisor.

1 : Cable de color azul es la salida (+ 16.5vdc) de la tarjeta rectificadora al regulador ECG933 (12vdc-5A).

En la Figura 41. se observa el plano de la fuente de alimentación para las tarjetas del módem y del transmisor.

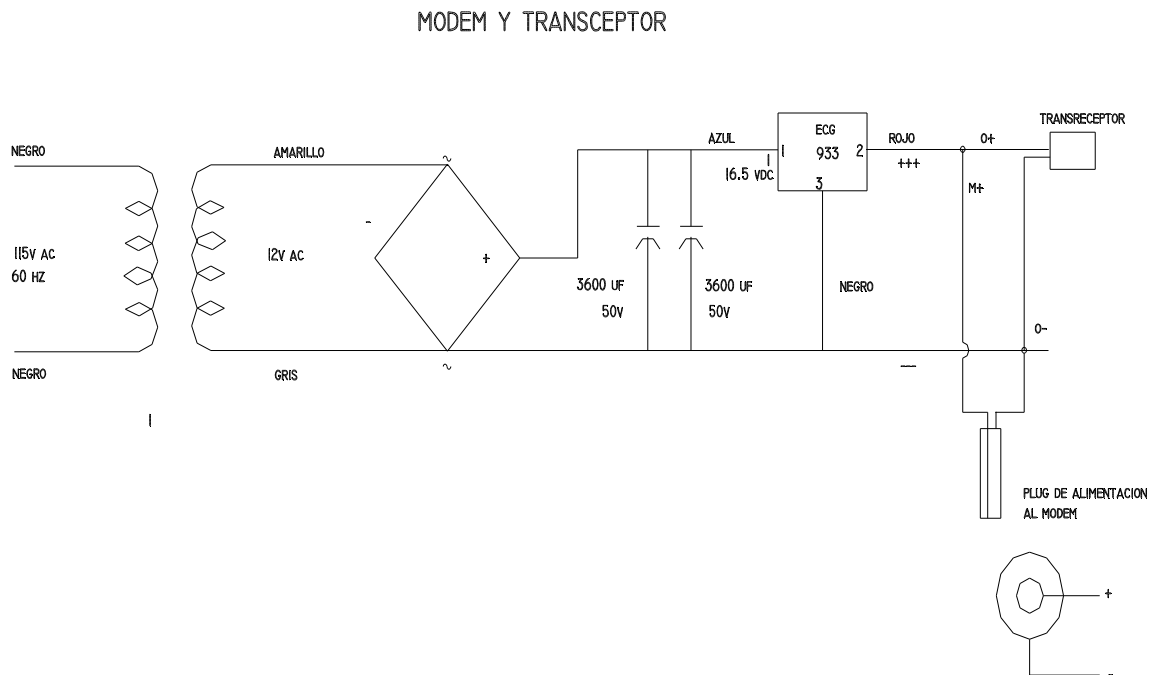


Figura 41. Fuente de 12 Vdc

Esta fuente está diseñada para los requerimientos máximos que se presentan en el momento de la transmisión teniendo un consumo de 3.8 A. Es conveniente no disponer de la fuente para otros uso o aditamentos del sistema.

6.2.1.2 Tarjeta Módem. Marca KANTRONICS referencia KPC-2400. La tarjeta está ensamblada a la caja por un par de guías en acrílico, sujetadas a la parte inferior de la caja con 2 tornillos para cada una de ellas. Ver Figura 42.

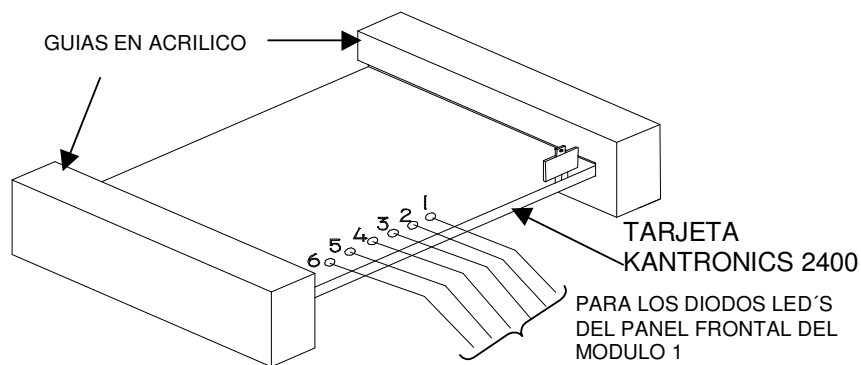


Figura 42. Ensamble de la tarjeta módem

Esta tarjeta tiene las siguientes conexiones: Los cables representados en la Figura 42. con los números 1, 2, 3, 4, 5, 6 son respectivamente:

1. Color naranja, lleva la señal STA al ánodo del led rojo.
2. Color rojo, lleva la señal CON al ánodo de color rojo.
3. Color marrón, lleva la señal RCV al ánodo de color rojo.
4. Color negro, lleva la señal XMIT al ánodo del led roja.
5. Color blanco, lleva la señal POWER al ánodo del led verde.

Estos led's están ubicados en una tarjeta impresa localizada en el panel frontal, su conexión es por medio de un plug, y son los que indican cada una de las funciones descritas en el subíndice 6.2.1.

La tarjeta módem además del plug para la alimentación antes expuesto, tiene el swich de ON-OFF y los conectores tipo DB que son terminales hembra para puerto RS-232 de 9 y 25 pines. Por medio del conector de 9 pines la tarjeta módem recibe y envía información análoga del/al transmisor. Esta conexión se

hace desde la tarjeta módem con su respectivo conector macho de 9 pines por medio de un par de cables blindados a dos plug monofónicos machos que van conectados a MIC (micrófono) y SP (parlante) disponibles en la parte frontal del radio transmisor. Ver Figura 43.

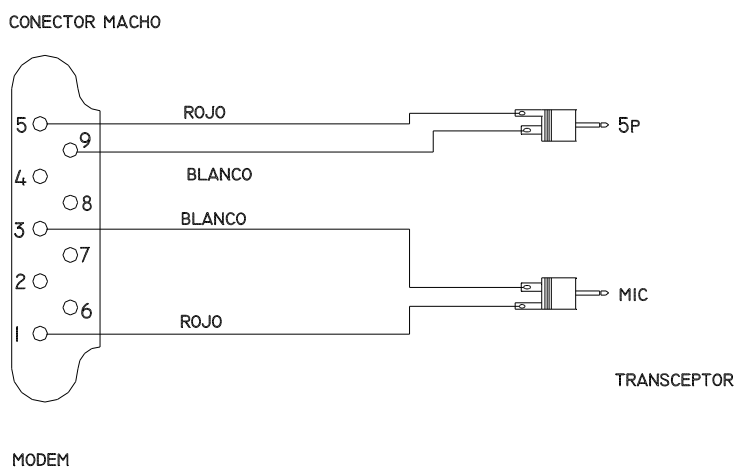


Figura 43. Conexión del módem al transceptor.

Por medio del conector de 25 pines la tarjeta Módem recibe y envía información digital del / al computador (PC). Este conector de 25 pines fue reducido a uno de 9 pines y soportado en el panel posterior del módulo 1, Ver Figura 37. En la Figura 44. se describe la conexión para la reducción del DTE terminal de 25 a 9 pines

6.2.1.3 Tarjeta del transceptor 2AT marca ICOM: Esta parte está representada en la Figura 38. con el número 4 y se refiere a un radio transceptor portátil de 2 metros que trabaja en las frecuencias comprendidas entre 144 a 147 MHz. La frecuencia de trabajo fue ajustada a 144 MHz. Para hacer posible su montaje se

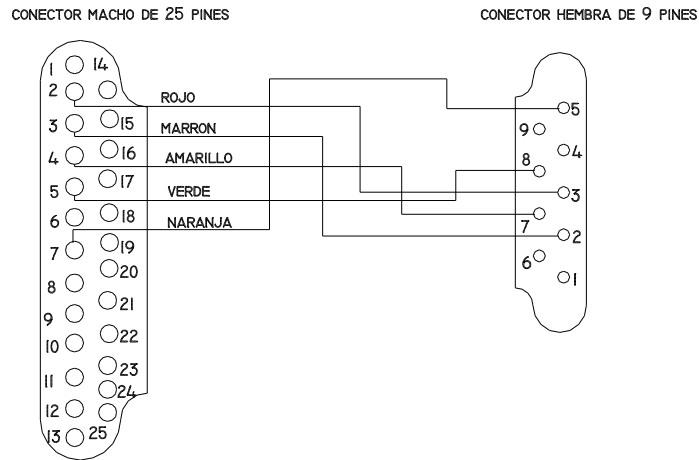


Figura 44. Conexión del módem a la MIC51

adapto este radio a las necesidades eliminando algunos de los accesorios como la batería, el teclado y caja. Para protegerlo de las interferencias se forro con una cubierta de cinta de plomo. El montaje se hizo sobre una lámina de acrílico que esta soportada con 2 tornillos sobre las guías del módem. Ver Figura 45.

El radio transmisor mientras no tenga los plug conectados en MIC y SP puede comunicarse con el otro radio receptor (estando en las mismas condiciones) oprimiendo el swich de PTT para hablar. En estos momentos se está haciendo uso del parlante y micrófono propios de cada radio.

Esto se puede realizar como prueba cuando no hay comunicación de mensajes y se sospeche de los radios transmisor y receptor.

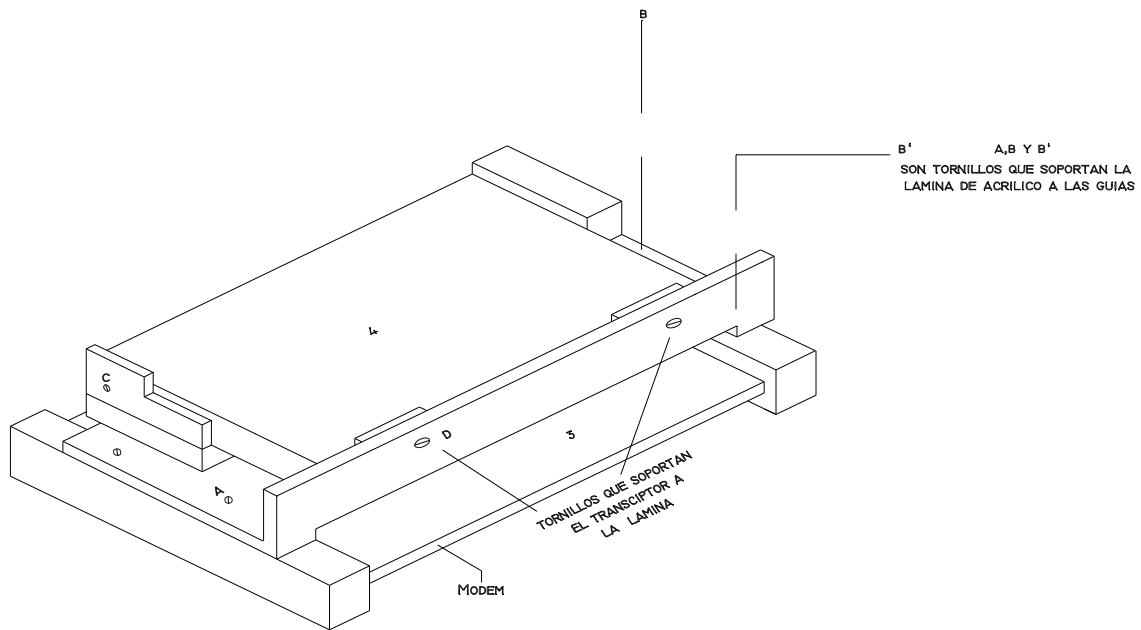


Figura 45. Ensamblaje del transceptor

En el anexo B se tiene la información que el fabricante suministra sobre el manual de funciones del radio marca ICOM 2AT. En el anexo N se encuentra el plano referente al diagrama de bloques y en el anexo Ñ su diagrama esquemático. En la Figura 46 daremos una idea sobre el panel frontal del mismo.

1. Conector tipo BNC de antena, por medio de una línea de transmisión se traslado este conector a la tapa superior de la caja.
2. Led rojo, enciende cuando se esta transmitiendo ya sea oprimiendo el swich de PPT o cuando esta función es reemplazada por el pulso de micrófono externo que lo maneja el módem cuando va a transmitir. También sirve como indicador de batería baja.
3. Squelch: ajuste para eliminar ruido de interferencia en el área.
4. Control de volumen de audio.

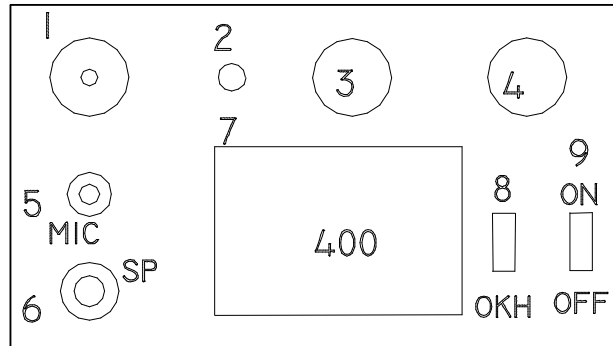


Figura 46. Panel frontal del Transceptor

5. MIC: Conector hembra para micrófono externo.

6. SP: Conector hembra para parlante externo.

El 5 y el 6 se utilizan como muestra la Figura 6.8.

7. Selector de canal con pasos de 10 KHz.

8. Selector de canal con paso cada 5 KHz

9. Swich de ON/OFF para suministro de potencia.

En el panel posterior se ajustaron los swiches de la siguiente forma:

- | | | |
|----|------|-------------|
| 1. | LOW | HIGH |
| 2. | DUP | SIMP |
| 3. | -600 | +600 |

1. Swich de salida de potencia baja (LOW)aproximadamente a 0.1W.

HIGH suministra 1.5 de potencia en antena, en ésta posición quedan ajustados

los radios transceptores.

2. Swich a simplex: debido a que los 2 radios están ajustados o sintonizados a la misma frecuencia.

Dúplex: Se utiliza cuando los radios están trabajando a frecuencias diferentes

3. Conmutador a -600kHz o $+600\text{kHz}$ es indiferente la posición ya que este swich es útil solo en la posición de Dúplex

6.2.2. Modulo 2 o receptor Digital: Este es otro de los módulos necesarios para cumplir el objetivo propuesto, a continuación describiremos esta etapa. El “módulo 2” es el encargado de recibir la información análoga por medio de la antena del radio receptor enviada por la antena del radio transmisor. La información recibida a 144MHz es amplificada y demodulada por el receptor; a través de los conectores de la Figura 43. es transferida al módem para que convierta esta información a señal digital. Obtenida esta señal digital a la salida del módem por medio del terminal de DTE de 25 se ha recuperado la misma información que fue enviada por el transmisor en cuyo proceso se realizó la función inversa, esto quiere decir que el módem del transmisor recibió la señal a enviar en forma digital desde el teclado del PC, éste la pasa a señal análoga y se la entregó al radio transmisor para que la amplifique y la module en frecuencia para luego ser enviada a través de la antena.

Esta información digital obtenida a la salida del módem normalmente es entregada a otra computadora por el puerto serial, esto para transferencia de datos vía R.F. Para nuestro caso esta información es recibida en forma serial por la tarjeta

microcontroladora para que entregue información binaria a las matrices de diodos y poder visualizar cualquier carácter enviado desde el PC.

Con esta breve explicación podemos concluir que para el procesamiento de la señal utilizamos los mismos elementos ya sea en la parte del módulo 1 ó del módulo 2.

A continuación se detallará el módulo 2 de recepción en lo referente al montaje del mismo. Para esta etapa los elementos fueron ensamblados de una forma muy similar al módulo 1 en una caja de uso comercial color blanco con las siguientes características, A=19cm, B=24cm, C=10cm.

La tapa frontal es removida quitando los dos tornillos ubicados en los lados y los dos tornillos que sujetan el terminal DB hembra de 9 pines. Este terminal es la reducción hecha al terminal DB de 25 pines de salida del módem. Ver Figura 44.

En la Figura 47. se describe gráficamente la caja utilizada con sus partes internas, y también se detallará la placa en acrílico del lado frontal. A continuación se identifica cada una de los números colocados en el gráfico.

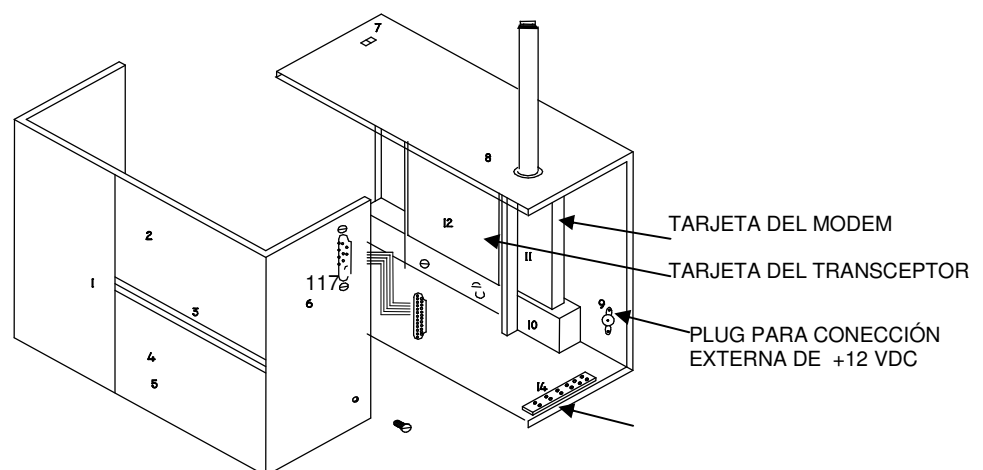


Figura 47. Vista general del módulo 2

1. Logotipo de la universidad con sus colores característicos.
2. Nombre de la Universidad.
3. Receptor digital de RF: Es el dado a este módulo ya que la información es recibida por la antena del radio y digitalizada en el módem para visualizarla en el display.
4. Por GERARDO MEJIA MEJIA: Autor del proyecto
5. Mayo /97: Fecha de entrega.
6. Terminal RS232 de 9 pines es una reducción del terminal del módem de 25 pines para conectar con el puerto serial del P.C.
7. Swich piloto para 12 VDC, suministra el voltaje a las dos tarjetas.
8. Antena flexible con su conector tipo BNC.
9. Conector para recibir la alimentación de 12 VDC externa.
10. Guías en acrílico para sujetar tarjeta módem.
11. Tarjeta módem KANTRONICS. Referencia KPC-2400: tiene los 5 led's acoplados directamente hacia un lado de la tapa, hacen la misma función descrita en la tarjeta del receptor.

12. Tarjeta del radio receptor ICOM 2AT : Es un radio portátil de 2 metros que transmite en FM a 144 MHz adaptado a las necesidades como las expuestas en el radio transmisor.

13. Lámina en acrílico donde se soporta el radio receptor.

14. Bornera: Distribuye la energía de 12 VDC recibidos por el plug (fuente externa) a las tarjeta del módem y del radio receptor.

Los accesorios implementados en este módulo son idénticos a los descritos en el módulo 1, incluidas las conexiones, acrílicos y ajustes del radio.

Este módulo va instalado junto al panel del display ya que al recibir la información debe ser visualizada. Un prototipo de este módulo se puede ver en la Figura 48.

6.2.3 Panel del display: Este es el “módulo 3” y es el terminal donde se visualiza la información enviada desde el PC para cumplir el objetivo trazado. Ver capítulo 4 y 5 donde se explica el diseño de la pantalla (display).

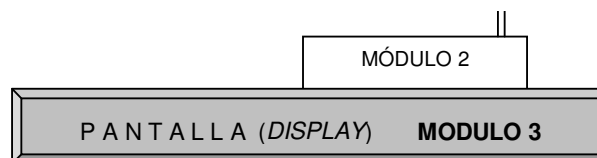


Figura 48. Vista del modulo 3

El módulo 3 contiene: En la Figura 48 se observa la vista frontal del módulo 3 correspondiente a la pantalla. Este módulo contiene:

1. Conector tipo DB de 9 pines para recibir la información a la salida del módulo 2
 2. Tarjeta microcontroladora : Encargada de recibir la información en forma serial, y gracias a un programa (en memoria ROM) también puede interpretar cada uno de los caracteres para almacenarlos en memoria RAM, esta tarjeta suministra en lenguaje binario la información a la tarjeta del display.

3. Tarjetas de display: Son las encargadas de recibir la información binaria en forma serial, pasarla a paralelo, multiplexarla, amplificarla y visualizarla. En el capítulo 5 se detalla cada uno de los elementos que conforman esta tarjeta.

4. Fuente de alimentación: Suministra energía de 12 VDC 4 Amperios al módulo 2 también alimenta la tarjeta microcontroladora y las tarjetas del display a 5VDC 2 Amperios. La fuente adaptada al módulo 3 es de uso comercial para compu

7. SOFTWARE UTILIZADO PARA MANEJAR EL DISPLAY

Dir Decim	Dir Hex	lenguaje de Maquina	lenguaje Nemotecnico	Comentario

**				
Inicio de la zona de programa				

**				
000001	0000	02002F	LJMP	PUERTO
000002	0003	00	NOP	
000003	0004	00	NOP	
000004	0005	00	NOP	
000005	0006	00	NOP	
000006	0007	00	NOP	
000007	0008	00	NOP	
000008	0009	00	NOP	
000009	000A	00	NOP	

**				
Vector de interrupción del TIMER0				

**				
000010	000B	020116	LJMP	VISUAL
000011	000E	00	NOP	

```

000012 000F 00      NOP
000013 0010 00      NOP
000014 0011 00      NOP
000015 0012 00      NOP
000016 0013 00      NOP
000017 0014 00      NOP
000018 0015 00      NOP
000019 0016 00      NOP
000020 0017 00      NOP
000021 0018 00      NOP
000022 0019 00      NOP
000023 001A 00      NOP
000024 001B 00      NOP
000025 001C 00      NOP
000026 001D 00      NOP
000027 001E 00      NOP
000028 001F 00      NOP
000029 0020 00      NOP
000030 0021 00      NOP
000031 0022 00      NOP
*****

```

```

**
Vector de interrupción del puerto serial
*****

```

```

**
000032 0023 02014B      LJMP RECEP
000033 0026 00      NOP
000034 0027 00      NOP
000035 0028 00      NOP
000036 0029 00      NOP
000037 002A 00      NOP
000038 002B 00      NOP
000039 002C 00      NOP
000040 002D 00      NOP
000041 002E 00      NOP
*****

```

```

**
Comienzo del programa principal
*****

```

```

000042 002F 759850  PUERTO  MOV SCON,#50H      ;Programa puerto serie en modo 1
000043 0032 758920      MOV TMOD,#20H      ;Programa el TIMER1 en modo 2 (Autorecarga)
000044 0035 758DF4      MOV TH1,#0F4H      ;Programamos la velocidad a 2400 baudios
000045 0038 53877F      ANL PCON,#7FH      ;No permitimos desdoblamiento de velocidad
000046 003B 43B810      ORL IP,#10H        ;Prioridad de interrupción para el puerto serie
000047 003E D28E        SETB 8EH           ;Permite conteo del TIMER1
000048 0040 D2AC        SETB 0ACH          ;Habilita interrupción del puerto serie
000049 0042 758CF9  INICIO  MOV TH0,#0F9H      ;Programamos los registros de temporización
del
000050 0045 758A00      MOV TL0,#00H       ;TIMER0
000051 0048 758120      MOV SP,#20H        ;Inicializamos el puntero de pila
000052 004B D2AF        SETB 0AFH          ;Habilita todas las interrupciones
000053 004D D2A9        SETB 0A9H          ;Permite interrupción del TIMER0
000054 004F D28C        SETB 8CH           ;Permite conteo del TIMER0
000055 0051 751830      MOV 18H,#30H       ;Carga la dirección de memoria interna

```

000056 0054 751A01		MOV 1AH,#01H	;Apunta a la 1ª línea del display
000057 0057 750800		MOV 08H,#00H	;Inicializamos la dirección de la cola de
mensajes			
			;de la RAM externa en los registros 08H (parte
			; y 09H (parte baja)
alta			
000058 005A 750911		MOV 09H,#11H	
000059 005D 750A00		MOV 0AH,#00H	;módem desconectado (0AH=0)
000060 0060 7800		MOV R0,#00H	;R0 y R1 apuntan al primer mensaje en RAM
000061 0062 7900		MOV R1,#00H	;externa
000062 0064 7C11		MOV R4,#11H	
000063 0066 900105		MOV DPTR,#MENS	;Apuntamos al primer mensaje
000064 0069 E4	GRAB	CLR A	
000065 006A 93		MOVC A,@A+DPTR	;Leemos el carácter
000066 006B A3		INC DPTR	;Incrementamos el apuntador
000067 006C AA82		MOV R2,DPL	;Guardamos el contenido del DPTR
000068 006E AB83		MOV R3,DPH	
000069 0070 8882		MOV DPL,R0	;Leemos la dirección en RAM externa donde sé
			;guardara el carácter leído de memoria de prog.
000070 0072 8983		MOV DPH,R1	
000071 0074 F0		MOVX @DPTR,A	;Grabamos carácter
000072 0075 A3		INC DPTR	;incrementamos el apuntador de RAM externa
000073 0076 A882		MOV R0,DPL	;Guardamos el contenido del DPTR
000074 0078 A983		MOV R1,DPH	
000075 007A 8A82		MOV DPL,R2	
000076 007C 8B83		MOV DPH,R3	
000077 007E DCE9		DJNZ R4,GRAB	;Verifica si sé alcanzado el ultimo carácter de la
			;cadena a grabar en RAM externa
000078 0080 8882		MOV DPL,R0	;Apuntamos a la siguiente dirección donde se
inici			;el segundo mensaje
000079 0082 8983		MOV DPH,R1	
000080 0084 E0		MOVX A,@DPTR	;Leemos dato
000081 0085 B45002		CJNE A,#50H,GRAB2	;Verifica si corresponde al efecto 'P'
(parpadear)			
			;sino intente nuevo efecto (GRAB2)
000082 0088 801C		SJMP REPET	;si es valido inicie programa
000083 008A B44402	GRAB2	CJNE A,#44H,GRAB3	;Verifica si corresponde al efecto 'D' (derecha)
000084 008D 8017		SJMP REPET	
000085 008F B44902	GRAB3	CJNE A,#49H,GRAB4	; Verifica si corresponde al efecto 'I' (izquierda)
000086 0092 8012		SJMP REPET	
000087 0094 B45302	GRAB4	CJNE A,#53H,GRAB5	; Verifica si corresponde al efecto 'S' (subir)
000088 0097 800D		SJMP REPET	
000089 0099 B44202	GRAB5	CJNE A,#42H,GRAB6	; Verifica si corresponde al efecto 'B' (bajar)
000090 009C 8000		SJMP GRAB6	
000091 009E B44E02	GRAB6	CJNE A,#4EH,GRAB7	; Verifica si corresponde al efecto 'N' (normal)
			;sino corresponde a ninguno de los códigos
			;entonces vaya a GRAB7 y corte cadena
			;si corresponde entonces continúe con el
000092 00A1 8003		SJMP REPET	
programa			
000093 00A3 7400	GRAB7	MOV A,#00H	
000094 00A5 F0		MOVX @DPTR,A	
000095 00A6 900000	REPET	MOV DPTR,#0000H	;Apuntamos al primer mensaje en RAM externa
000096 00A9 E0	LAZO	MOVX A,@DPTR	;Leemos el efecto
000097 00AA 60FA		JZ REPET	;Verifica si es el ultimo mensaje
000098 00AC 858213		MOV 13H,DPL	;Guardamos el DPTR
000099 00AF 858314		MOV 14H,DPH	

```

000100 00B2 1200B7      LCALL LLAMA      ;Visualizamos el mensaje
000101 00B5 80F2      SJMP LAZO

```

```

*****
**

```

Nombre : LLAMA

Esta rutina busca la dirección del subprograma que produce el efecto

```

*****
**

```

```

000102 00B7 F8      LLAMA  MOV R0,A      ;Guarda el efecto
000103 00B8 9000EE      MOV DPTR,#ORDEN ;Apuntamos a la tabla de ordenes
000104 00BB 7906      MOV R1,#06H     ;Total de efectos disponibles
000105 00BD E4      BUCLE  CLR A
000106 00BE 93      MOV C A,@A+DPTR ;Leemos código
000107 00BF B50002    CJNE A,00H,BUC10 ;Compara con el efecto almacenado en el
registro
;R0
000108 00C2 8010      SJMP BUC20      ;Código encontrado
000109 00C4 A3      BUC10  INC DPTR      ;Apuntamos al siguiente código del efecto
000110 00C5 D9F6      DJNZ R1,BUCLE   ;Veamos el siguiente
000111 00C7 9000F4    MOV DPTR,#ERROR ;Apunta al mensaje error si no existe efecto
000112 00CA 1202DB    LCALL NORMAL
000113 00CD C28C      CLR 8CH        ;Detenemos visualización del display
000114 00CF 759000    MOV P1,#00H
000115 00D2 80FE      FIN          SJMP FIN        ;Bucle sin fin
000116 00D4 19      BUC20  DEC R1
000117 00D5 E9      MOV A,R1
000118 00D6 29      ADD A,R1
000119 00D7 29      ADD A,R1
000120 00D8 9000DC    MOV DPTR,#SALTO ;DPTR apunta a la tabla de saltos
000121 00DB 73      JMP @A+DPTR    ;SALTAMOS!

```

```

*****
**

```

Tabla de saltos

```

*****
**

```

```

000122 00DC 0202DB    SALTO  LJMP NORMAL
000123 00DF 020304      LJMP BAJA
000124 00E2 020354      LJMP SUBE
000125 00E5 0203A4      LJMP IZQUI
000126 00E8 0203DA      LJMP DERECH
000127 00EB 02040A      LJMP PARPA

```

```

*****
**

```

Código de los efectos

```

*****
**

```

```

000128 00EE 50      ORDEN  DB 'P'
000129 00EF 44      DB 'D'
000130 00F0 49      DB 'I'
000131 00F1 53      DB 'S'
000132 00F2 42      DB 'B'
000133 00F3 4E      DB 'N'

```

**

Mensaje de ERROR

***000134 00F4 4E ERRORDB 'N'

000135 00F5 10	DB 10H
000136 00F6 01	DB 01H
000137 00F7 20	DB ' '
000138 00F8 20	DB ' '
000139 00F9 20	DB ' '
000140 00FA 45	DB 'E'
000141 00FB 52	DB 'R'
000142 00FC 52	DB 'R'
000143 00FD 4F	DB 'O'
000144 00FE 52	DB 'R'
000145 00FF 20	DB ' '
000146 0100 20	DB ' '
000147 0101 20	DB ' '
000148 0102 20	DB ' '
000149 0103 20	DB ' '
000150 0104 00	DB 00H

**

1er. Mensaje

**

000151 0105 4E	MENS DB 'N'
000152 0106 10	DB 10H
000153 0107 01	DB 01H
000154 0108 20	DB ' '
000155 0109 20	DB ' '
000156 010A 20	DB ' '
000157 010B 43	DB 'C'
000158 010C 2E	DB ' '
000159 010D 55	DB 'U'
000160 010E 2E	DB ' '
000161 010F 54	DB 'T'
000162 0110 2E	DB ' '
000163 0111 42	DB 'B'
000164 0112 20	DB ' '
000165 0113 20	DB ' '
000166 0114 20	DB ' '
000167 0115 00	DB 00H

**

Comienzo de rutinas generales de uso

**

**

RUTINA DE INTERRUPCION DEL TIMER0

A cada interrupción se visualiza una nueva línea

R0 : Apunta al byte

R1 : Cuenta el número de bytes por línea

R2 : Número de líneas

R3 : Bits por Bytes

**

```
000168 0116 C0D0  VISUAL  PUSH PSW           ;Salvamos estado de los flancos
000169 0118 C0E0          PUSH ACC
000170 011A D2D3          SETB 0D3H         ;Cambiamos de banco de registros
000171 011C D2D4          SETB 0D4H         ;Banco 3
000172 011E 790A          MOV R1,#0AH       ;Número de bytes por línea
000173 0120 E6  VI10     MOV A,@R0         ;Lee byte
000174 0121 7B08          MOV R3,#08H       ;Bits por Bytes
000175 0123 33  VI20     RLC A             ;Sacamos un bit del acumulador
000176 0124 92B3          MOV 0B3H,C        ;Dato en el registro
000177 0126 1201CE       LCALL CLOK        ;A dentro!
000178 0129 DBF8          DJNZ R3,VI20      ;Otro bit
000179 012B 08          INC R0            ;Apuntamos al próximo byte
000180 012C D9F2          DJNZ R1,VI10
000181 012E 8A90          MOV P1,R2         ;Activamos línea
000182 0130 D297          SETB 97H          ;Cargamos LACHES de salida
000183 0132 C297          CLR 97H
000184 0134 EA          MOV A,R2          ;Preparamos la próxima línea a visualizar
000185 0135 23          RL A
000186 0136 FA          MOV R2,A
000187 0137 30E704       JNB 0E7H,VI30
000188 013A 7A01          MOV R2,#01H       ;Preparamos la primera línea
000189 013C 7830          MOV R0,#30H       ;Apuntamos al 1er byte de RAM interna
000190 013E 758CF9  VI30     MOV TH0,#0F9H     ;Inicializamos registros del TIMER
000191 0141 758A00       MOV TL0,#00H
000192 0144 C2D3          CLR 0D3H
000193 0146 D0E0          POP ACC
000194 0148 D0D0          POP PSW
000195 014A 32          RETI
```

**

RUTINA DE INTERRUPCIÓN DEL PUERTO SERIAL

Esta rutina lee la información que le llega por el puerto serial, la procesa y la clasifica. Si la información recogida es valida (mensaje correcto) la copia en la RAM externa, sino, la procesa como basura y no la toma en cuenta (informaciones de este tipo las envía el módem cuando se enciende, enlaza o desconecta al transmisor)

Banco 1 de resistencia

R0 : Almacena el estado conexión/desconexión con el módem.

R2R1 :Almacenan la dirección final de la cola de mensajes.

R3 :Lleva la cuenta de cuantos caracteres del mensaje se han leído.

**

```
000196 014B C0D0  RECEP  PUSH PSW           ;Salvamos los estados de los Flags
000197 014D C0E0          PUSH ACC
000198 014F C082          PUSH DPL          ;Guardamos el DPTR
000199 0151 C083          PUSH DPH
```

000200 0153 75D000		MOV PSW,#00H	;Cambiamos al Banco 1
000201 0156 D2D3		SETB 0D3H	
000202 0158 E8		MOV A,R0	;Leemos el estado del módem
000203 0159 B4000E		CJNE A,#00H,ENLAC	;Verifica si esta conectado a la tarjeta
			;si A=#00H entonces esta desconectado
000204 015C 3098FD	LEER	JNB 98H,LEER	;Leemos mensaje enviado por módem cuando se
			;enciende por primera vez
000205 015F E599		MOV A,SBUF	
000206 0161 C298		CLR 98H	
000207 0163 B43AF6		CJNE A,#3AH,LEER	;Detecta fin del mensaje
000208 0166 7801		MOV R0,#01H	;módem conectado (R0=direcc. 0AH=#01H)
000209 0168 805B		SJMP SALIR	
000210 016A 3098FD	ENLAC	JNB 98H,ENLAC	;Leemos el 1er carácter del cadena
000211 016D E599		MOV A,SBUF	
000212 016F C298		CLR 98H	
000213 0171 B42A0C		CJNE A,#2AH,GUARD	;Verificamos si es un mensaje para el tablero o un
			;mensaje de enlace entre módem.
000214 0174 3098FD	LEER1	JNB 98H,LEER1	;Leemos los caracteres del mensaje
			;” ****CONECTADO A BASE “
000215 0177 E599		MOV A,SBUF	
000216 0179 C298		CLR 98H	
000217 017B B40AF6		CJNE A,#0AH,LEER1	;Detecta fin del mensaje
000218 017E 8045		SJMP SALIR	
000219 0180 8982	GUARD	MOV DPL,R1	;Recuperamos la dirección de la cola de mensajes
000220 0182 8A83		MOV DPH,R2	
000221 0184 7B0F		MOV R3,#0FH	;Leer 16 caracteres
000222 0186 F0		MOVX @DPTR,A	;Guardamos el 1er carácter del mensaje (EFECTO)
000223 0187 A3	I	NC DPTR	
000224 0188 3098FD	LEER2	JNB 98H,LEER2	;Seguimos leyendo los caracteres del mensaje
000225 018B E599		MOV A,SBUF	
000226 018D C298		CLR 98H	
000227 018F F0		MOVX @DPTR,A	;Guardamos en la pila de mensajes
000228 0190 A3		INC DPTR	;Apuntamos a la siguiente posición
000229 0191 DB02		DJNZ R3,SALT	;Va por otro carácter
000230 0193 8016		SJMP CONT1	
000231 0195 B40AF0	SALT	CJNE A,#0AH,LEER2	;Detecta si se alcanzo fin de línea (LF=#0AH)
000232 0198 7400	COMPL	MOV A,#00H	;Completamos mensaje
000233 019A F0		MOVX @DPTR,A	
000234 019B A3		INC DPTR	
000235 019C DBFA		DJNZ R3,COMPL	;Va por otro carácter
000236 019E 7400		MOV A,#00H	
000237 01A0 F0		MOVX @DPTR,A	;Marcamos fin del mensaje
000238 01A1 A3		INC DPTR	;Apunta a la dirección del próximo mensaje
000239 01A2 A982		MOV R1,DPL	;Guardamos dirección
000240 01A4 AA83		MOV R2,DPH	
000241 01A6 F0		MOVX @DPTR,A	;Marcamos final de la pila de mensaje
000242 01A7 A3		INC DPTR	
000243 01A8 F0		MOVX @DPTR,A	
000244 01A9 80EA		SJMP SALT	
000245 01AB B40A02	CONT1	CJNE A,#0AH,SEG	;Detecta si se alcanzo fin de línea
000246 01AE 8015		SJMP SALIR	
000247 01B0 7400	SEG	MOV A,#00H	;Marcamos fin del mensaje

```

000248 01B2 F0          MOVX @DPTR,A
000249 01B3 A3          INC DPTR          ;Apunta a la próxima dirección
000250 01B4 A982       MOV R1,DPL       ;Guarda la dirección del próximo mensaje
000251 01B6 AA83       MOV R2,DPH
000252 01B8 F0          MOVX @DPTR,A    ; Marcamos final de la pila de mensaje
000253 01B9 A3          INC DPTR
000254 01BA F0          MOVX @DPTR,A
000255 01BB 3098FD LEER3 JNB 98H,LEER3   ;Lee el resto de la cadena que no se va
000256 01BE E599       MOV A,SBUF      ; a utilizar
000257 01C0 C298       CLR 98H
000258 01C2 B40AF6     CJNE A,#0AH,LEER3 ;Detecta si se alcanzo fin de línea
000259 01C5 D083 SALIR POP DPH          ;Recupera el DPRT
000260 01C7 D082       POP DPL
000261 01C9 D0E0       POP ACC
000262 01CB D0D0       POP PSW          ;Recupera los flancos
000263 01CD 32         RETI
000264 01CE C2B2 CLOK CLR 0B2H          ;Pulso de reloj para el registro
000265 01D0 D2B2       SETB 0B2H
000266 01D2 22         RET

```

```

*****
**

```

Nombre : PONVI

R0 : Apunta a zona Vídeo RAM interna

R1 : N° de caracteres

R7R6 : Guarda dirección de RAM externa

```

*****
**

```

Este procedimiento copia en la zona de vídeo RAM interna las zonas de RAM externa que guardan copias de la visualización.

```

*****
**

```

```

000267 01D3 C28C PONVI CLR 8CH
000268 01D5 C083       PUSH DPH
000269 01D7 C082       PUSH DPL
000270 01D9 7946       MOV R1,#46H     ;N° de caracteres a copiar
000271 01DB 7830       MOV R0,#30H
000272 01DD 8E82       MOV DPL,R6     ;Dirección inicial de RAM externa
000273 01DF 8F83       MOV DPH,R7
000274 01E1 E0 PON10 MOVX A,@DPTR   ;Leemos carácter
000275 01E2 F6         MOV @R0,A      ;Copiamos carácter
000276 01E3 08         INC R0         ;Incrementamos apuntadores
000277 01E4 A3         INC DPTR
000278 01E5 D9FA       DJNZ R1,PON10 ;Otro carácter
000279 01E7 D082       POP DPL
000280 01E9 D083       POP DPH
000281 01EB D28C       SETB 8CH
000282 01ED 22         RET

```

```

*****
**

```

Nombre : ASTRA

```

*****
**

```

Esta rutina transforma la cadena ASCII del mensaje, en la trama de bits correspondiente, que volcada sobre la ZVI, permite su visualización

R0 : Se utiliza para realizar operaciones parciales
R1 : Contará el N° de bits útiles de la trama (6)
R2 : Cuenta el N° de bits por byte que se van procesando
R3 : Control del N° de fila o línea
R4 : Registro temporal de la trama
R5 : Contendrá el contenido binario PARCIAL del byte
R7R6 : Apunta a vídeo RAM externa

**

```

000283 01EE C0E0  ASTRA  PUSH ACC
000284 01F0 7F05                MOV R7,#05H      ;Apuntamos a vídeo
000285 01F2 7E46                MOV R6,#46H
000286 01F4 7B00                MOV R3,#00H
000287 01F6 C082                PUSH DPL         ;Guardamos la dirección del mensaje
000288 01F8 C083                PUSH DPH
000289 01FA 7A08  FILA    MOV R2,#08H      ;8 bits por bytes
000290 01FC E0    CARAC  MOVX A,@DPTR     ;Leemos carácter
000291 01FD C082                PUSH DPL
000292 01FF C083                PUSH DPH
000293 0201 900500             MOV DPTR,#0500H ;Apuntamos a la tabla de códigos
000294 0204 75F007             MOV B,#07H      ;7 bytes por código
000295 0207 A4                MUL AB          ;Obtenemos desplazamiento del ASCII dentro
000296 0208 2582                ADD A,DPL      ;la tabla según la formula:
000297 020A F582                MOV DPL,A      ;DPTR=CODIGOS +7*ASCII+FILA
000298 020C E5F0                MOV A,B
000299 020E 2583                ADD A,DPH
000300 0210 F583                MOV DPH,A
000301 0212 C3                CLR C
000302 0213 E582                MOV A,DPL
000303 0215 2B                ADD A,R3       ;Fila
000304 0216 F582                MOV DPL,A
000305 0218 E583                MOV A,DPH
000306 021A 3400                ADDC A,#00H
000307 021C F583                MOV DPH,A
000308 021E E4                CLR A
000309 021F 93                MOVC A,@A+DPTR ;Cogemos la trama
000310 0220 D083                POP DPH
000311 0222 D082                POP DPL
000312 0224 23                RL A          ;Omitimos los dos bits sobrantes
000313 0225 23                RL A
000314 0226 FC                MOV R4,A      ;Guardamos la trama en R4
000315 0227 7906                MOV R1,#06H   ;Añadir los 6 bits a la línea
000316 0229 EC  SEIS    MOV A,R4      ;Recuperamos la trama
000317 022A 5480                ANL A,#80H    ;enmascaramos los bits
000318 022C F8                MOV R0,A
000319 022D EC                MOV A,R4      ;Recuperamos trama
000320 022E 33                RLC A         ;Cogemos un bit
000321 022F FC                MOV R4,A      ;Guardamos Trama
000322 0230 ED                MOV A,R5      ;Cogemos el byte en proceso de relleno
000323 0231 33                RLC A         ;Le añadimos el bit
000324 0232 B88002             CJNE R0,#80H,CONT
000325 0235 4401                ORL A,#01H

```

```

000326 0237 FD      CONT    MOV R5,A           ;Guardamos el parcial del byte
000327 0238 DA14    DJNZ R2,NOCHO     ;Ya se ha completado el byte
000328 023A 7A08    MOV R2,#08H      ;8 bits por byte
000329 023C C082    PUSH DPL         ;Guardamos puntero a mensaje
000330 023E C083    PUSH DPH
000331 0240 8E82    MOV DPL,R6       ;Recuperamos puntero a vídeo
000332 0242 8F83    MOV DPH,R7
000333 0244 F0      MOVX @DPTR,A     ;Guardamos byte a zona de vídeo
000334 0245 A3      INC DPTR         ;Apuntamos a la posición siguiente
000335 0246 AE82    MOV R6,DPL       ;Guardamos puntero al mensaje
000336 0248 AF83    MOV R7,DPH
000337 024A D083    POP DPH          ;Recuperamos puntero al mensaje
000338 024C D082    POP DPL
000339 024E D9D9    NOCHO           DJNZ R1,SEIS     ;A por otro bit de los 6 de la TRAMA
000340 0250 A3      INC DPTR         ;Apuntamos al siguiente carácter
000341 0251 E4      CLR A
000342 0252 E0      MOVX A,@DPTR    ;Leemos carácter del mensaje
000343 0253 6002    JZ ULTI         ;Es el ultimo carácter
000344 0255 80A5    SJMP CARAC      ;A procesar el siguiente carácter
000345 0257 858213  ULTI           MOV 13H,DPL      ;Hacemos una copia del DPTR
000346 025A 858314  MOV 14H,DPH
000347 025D C3      CLR C           ;Borramos los dos puntos finales
000348 025E ED      MOV A,R5        ;80 puntos horizontales se reparten entre:
000349 025F 33      RLC A           ;13 caracteres de 6 puntos y sobran 2
000350 0260 C3      CLR C
000351 0261 33      RLC A
000352 0262 C082    PUSH DPL        ;Guardamos puntero a mensaje
000353 0264 C083    PUSH DPH
000354 0266 8E82    MOV DPL,R6     ;Recuperamos puntero a vídeo
000355 0268 8F83    MOV DPH,R7
000356 026A F0      MOVX @DPTR,A   ;Guardamos byte en zona de vídeo
000357 026B A3      INC DPTR       ;Apuntamos a la posición siguiente
000358 026C AE82    MOV R6,DPL     ;Guardamos puntero a vídeo
000359 026E AF83    MOV R7,DPH
000360 0270 D083    POP DPH        ;Recuperamos puntero al mensaje
000361 0272 D082    POP DPL
000362 0274 0B      INC R3         ;Incrementamos contador de bits
000363 0275 D083    POP DPH
000364 0277 D082    POP DPL
000365 0279 C082    PUSH DPL
000366 027B C083    PUSH DPH
000367 027D BB0702  CJNE R3,#07H,SALT1 ;A por otra fila
000368 0280 4184    AJMP SALT2
000369 0282 21FA    SALT1          AJMP FILA
000370 0284 D0E0    SALT2          POP ACC
000371 0286 D083    POP DPH
000372 0288 D082    POP DPL
000373 028A 22      RET

```

**

Nombre : LIMPIA
R7R6 : Origen
R3 : N° de bytes a mover

Esta rutina rellena con el valor del ACC , a partir de la dirección indicada en R7R6, el N° de veces indicado en R3

```
*****
**
000374 028B F0      LIMPIA MOVX @DPTR,A      ;Escribimos carácter
000375 028C A3      INC DPTR                 ;Apuntamos a la dirección siguiente
000376 028D DBFC    DJNZ R3,LIMPIA          ;A por otro
000377 028F 22      RET
```

```
*****
**
Nombre : MOVER
R7R6 : Origen
R5R4 : Destino
R3   : N° de bytes a mover
*****
```

```
**
000378 0290 8F83    MOVER  MOV DPH,R7        ;Apuntamos a ORIGEN
000379 0292 8E82    MOV DPL,R6
000380 0294 E0      MOVX A,@DPTR            ;leemos carácter
000381 0295 A3      INC DPTR                 ;Incrementamos apuntador
000382 0296 AF83    MOV R7,DPH
000383 0298 AE82    MOV R6,DPL
000384 029A 8D83    MOV DPH,R5              ;Apuntamos a DESTINO
000385 029C 8C82    MOV DPL,R4
000386 029E F0      MOVX @DPTR,A           ;Copiamos carácter
000387 029F A3      INC DPTR                 ;Incrementamos apuntador
000388 02A0 AD83    MOV R5,DPH
000389 02A2 AC82    MOV R4,DPL
000390 02A4 DBEA    DJNZ R3,MOVER          ;A por otro
000391 02A6 22      RET
```

```
*****
**
Nombre : LINEA
R7R6 : Apunta a VIDEO en RAM externa
R5R4 : Apunta a CVIDEO en RAM externa
```

Permite copiar desde VIDEO, a partir de la línea indicada en R6 sobre CVIDEO, a partir de la línea indicada en R4

```
*****
**
000392 02A7 C006    LINEA  PUSH 06H          ;Guardamos registros
000393 02A9 C004    PUSH 04H
000394 02AB C003    PUSH 03H
000395 02AD 7F05    MOV R7,#05H            ;R7R6 apunta a VIDEO
000396 02AF 7D05    MOV R5,#05H
000397 02B1 120290   LCALL MOVER             ;Mueve las líneas
000398 02B4 7F05    MOV R7,#05H            ;Apunta CVIDEO
000399 02B6 7E00    MOV R6,#00H
000400 02B8 1201D3   LCALL PONVI             ;Mueve a RAM interna
000401 02BB E511    MOV A,11H              ;Lee el tiempo
000402 02BD 1202CA   LCALL DIEZMS           ;espera
000403 02C0 D5E0FD   DJNZ ACC,CICL
000404 02C3 D003    POP 03H
000405 02C5 D004    POP 04H
```

```
000406 02C7 D006      POP 06H
000407 02C9 22        RET
```

```
*****
**
```

```
Nombre :DIEZMS
R0 : Contador para generar retardo
R1 : Contador para generar retardo
```

```
Esta rutina genera un retardo de 10 ms por medio de un bucle cerrado formado por los registros R0 y R1
*****
**
```

```
000408 02CA 881A  DIEZMS  MOV 1AH,R0      ;Guardamos el valor actual de R0
000409 02CC 891B          MOV 1BH,R1      ;Guardamos el valor actual de R1
000410 02CE 7831          MOV R0,#31H     ;Inicializamos Registro
000411 02D0 7910  DIEZM05  MOV R1,#10H     ;Inicializamos Registro
000412 02D2 D9FE  DIEZM10  DJNZ R1,DIEZM10 ;Bucle de conteo
000413 02D4 D8FA          DJNZ R0,DIEZM05
000414 02D6 A91B          MOV R1,1BH     ;Recuperamos los registros
000415 02D8 A81A          MOV R0,1AH
000416 02DA 22          RET
```

```
*****
**
```

```
Comienzo de las rutina de efectos
```

```
*****
**
```

```
*****
**
```

```
Efecto : Visualiza el mensaje
Nombre : NORMAL
Código : 'N'
```

```
*****
**
```

```
000417 02DB 851382 NORMAL MOV DPL,13H      ;Recupera dirección del mensaje
000418 02DE 851483          MOV DPH,14H
000419 02E1 A3              INC DPTR        ;Apunta al byte de tiempo
000420 02E2 E0              MOVX A,@DPTR
000421 02E3 F511          MOV 11H,A      ;Lo guarda en la variable
000422 02E5 A3              INC DPTR        ;Apunta al byte de Veces
000423 02E6 E0              MOVX A,@DPTR
000424 02E7 F510          MOV 10H,A      ;Lo guarda, aunque no lo utilice
000425 02E9 A3              INC DPTR        ;Apunta el texto del mensaje
000426 02EA 1201EE          LCALL ASTRA     ;Genera la trama de bits
000427 02ED 7F05          MOV R7,#05H    ;Lo vuelca sobre la RAM interna
000428 02EF 7E46          MOV R6,#46H
000429 02F1 1201D3          LCALL PONVI
000430 02F4 E511          MOV A,11H      ;Recobramos el tiempo
000431 02F6 1202CA  NOR10  LCALL DIEZMS    ;Espera
000432 02F9 D5E0FA          DJNZ ACC,NOR10
000433 02FC 851382          MOV DPL,13H    ;Preparamos el apuntador para que apunte al
000434 02FF 851483          MOV DPH,14H    ;mensaje siguiente
000435 0302 A3              INC DPTR
000436 0303 22          RET
```

```

*****
**
Efecto :Visualiza el mensaje deslizando hacia abajo
Nombre :BAJA
Código : 'B'
*****
**

```

```

000437 0304 851382 BAJA MOV DPL,13H ;Recupera dirección del mensaje
000438 0307 851483 MOV DPH,14H
000439 030A A3 INC DPTR ;Apunta ala byte de tiempo
000440 030B E0 MOVX A,@DPTR
000441 030C F511 MOV 11H,A ;Lo guarda en la variable
000442 030E A3 INC DPTR ;Apunta al byte de veces
000443 030F E0 MOVX A,@DPTR
000444 0310 F510 MOV 10H,A ;Lo guarda, aunque no lo utilice
000445 0312 A3 INC DPTR ;Apunta al texto del mensaje
000446 0313 1201EE LCALL ASTRA ;Genera la trama de bits
000447 0316 7400 MOV A,#00H ;Borramos la zona CVIDEO
000448 0318 7B46 MOV R3,#46H
000449 031A 900500 MOV DPTR,#0500H
000450 031D 12028B LCALL LIMPIA
000451 0320 7F05 MOV R7,#05H ;Lo vuelca sobre la RAM interna
000452 0322 7E00 MOV R6,#00H
000453 0324 1201D3 LCALL PONVI
000454 0327 E511 MOV A,11H ;Recuperamos tiempo
000455 0329 1202CA BAJ05 LCALL DIEZMS ;esperamos
000456 032C D5E0FA DJNZ ACC,BAJ05
000457 032F 7E82 MOV R6,#82H ;Preparamos las líneas a copiar de VIDEO a
;CVIDEO

000458 0331 7C00 MOV R4,#00H
000459 0333 7B0A MOV R3,#0AH
000460 0335 1202A7 BAJ10 LCALL LINEA ;Llama la línea a copiar
000461 0338 EB MOV A,R3
000462 0339 240A ADD A,#0AH ;Incrementamos el número de líneas a copiar
000463 033B FB MOV R3,A
000464 033C C3 CLR C
000465 033D EE MOV A,R6
000466 033E 940A SUBB A,#0AH ;Apuntamos a la siguiente línea
000467 0340 FE MOV R6,A
000468 0341 BE3CF1 CJNE R6,#3CH,BAJ10
000469 0344 E511 MOV A,11H ;Recupera el tiempo
000470 0346 1202CA BAJ15 LCALL DIEZMS ;Espera
000471 0349 D5E0FA DJNZ ACC,BAJ15
000472 034C 851382 MOV DPL,13H ;Preparamos el apuntador para que apunte al
000473 034F 851483 MOV DPH,14H ;mensaje siguiente
000474 0352 A3 INC DPTR
000475 0353 22 RET

```

```

*****
**
Efecto : Visualiza el mensaje deslizando hacia arriba
Nombre : SUBE
Código : 'S'

```

**

```

000476 0354 851382 SUBE      MOV DPL,13H      ;Recupera dirección del mensaje
000477 0357 851483          MOV DPH,14H
000478 035A A3             INC DPTR         ;Apunta al byte de tiempo
000479 035B E0             MOVX A,@DPTR
000480 035C F511          MOV 11H,A        ;Lo guarda en la variable
000481 035E A3             INC DPTR         ;Apunta al byte de veces
000482 035F E0             MOVX A,@DPTR
000483 0360 F510          MOV 10H,A        ;Lo guarda, aunque no lo utilice
000484 0362 A3             INC DPTR
000485 0363 1201EE        LCALL ASTRA      ;Genera la trama de bits
000486 0366 7400          MOV A,#00H      ;Borramos la zona de CVIDEO
000487 0368 7B46          MOV R3,#46H
000488 036A 900500        MOV DPTR,#0500H
000489 036D 12028B        LCALL LIMPIA     ;Limpia
000490 0370 7F05          MOV R7,#05H     ;Lo vuelca sobre la RAM interna
000491 0372 7E00          MOV R6,#00H
000492 0374 1201D3        LCALL PONVI
000493 0377 E511          MOV A,11H       ;Recupera el tiempo
000494 0379 1202CA SUB05    LCALL DIEZMS     ;Espera
000495 037C D5E0FA        DJNZ ACC,SUB05
000496 037F 7E46          MOV R6,#46H     ;Preparamos la líneas a copiar de VIDEO a
                                ;CVIDEO
000497 0381 7C3C          MOV R4,#03CH
000498 0383 7B0A          MOV R3,#0AH
000499 0385 1202A7 SUB10    LCALL LINEA     ;Llama la línea a copiar
000500 0388 EB            MOV A,R3
000501 0389 240A          ADD A,#0AH      ;Incrementamos el número de líneas a copiar
000502 038B FB            MOV R3,A
000503 038C C3            CLR C
000504 038D EC            MOV A,R4
000505 038E 940A          SUBB A,#0AH     ;Apuntamos a la siguiente línea
000506 0390 FC            MOV R4,A
000507 0391 BCF6F1        CJNE R4,#0F6H,SUB10
000508 0394 E511          MOV A,11H       ;Recuperamos tiempo
000509 0396 1202CA SUB15    LCALL DIEZMS     ;Espera
000510 0399 D5E0FA        DJNZ ACC,SUB15
000511 039C 851382          MOV DPL,13H     ;Preparamos el apuntador para que apunte
000512 039F 851483          MOV DPH,14H     ;al mensaje siguiente
000513 03A2 A3             INC DPTR
000514 03A3 22            RET

```

**

Efecto : Visualiza el mensaje y lo borra deslizando hacia la izquierda
Nombre : IZQUIERDA
Código : 'I'

R2 : N° de puntos horizontales del letrero
R1 : N° de líneas a desplazar
R0 : N° de bytes por línea
DPTR : Apunta a vídeo

**

```

000515 03A4 1202DB IZQUI LCALL NORMAL ;Visualiza el mensaje normalmente
000516 03A7 C082 PUSH DPL
000517 03A9 C083 PUSH DPH
000518 03AB 7A50 MOV R2,#50H ;Nº de puntos horizontales del letrero
000519 03AD 90054F IZQ02 MOV DPTR,#054FH ;Apuntamos a video en RAM externa
000520 03B0 7907 MOV R1,#07H ;Número de líneas a desplazar
000521 03B2 780A IZQ05 MOV R0,#0AH ;Nº de bytes por línea
000522 03B4 C3 CLR C ;Iremos apagando los puntos
000523 03B5 E0 IZQ10 MOVX A,@DPTR ;Leemos carácter
000524 03B6 33 RLC A ;Lo desplazamos con carry
000525 03B7 F0 MOVX @DPTR,A ;Lo restituimos
000526 03B8 1582 DEC DPL ;Apuntamos al siguiente
000527 03BA D8F9 DJNZ R0,IZQ10 ;Hasta terminar la línea
000528 03BC E582 MOV A,DPL ;A por la siguiente línea
000529 03BE 2414 ADD A,#14H ;Calculamos la dirección
000530 03C0 F582 MOV DPL,A
000531 03C2 D9EE DJNZ R1,IZQ05
000532 03C4 7F05 MOV R7,#05H ;Copiamos en la RAM interna
000533 03C6 7E46 MOV R6,#46H
000534 03C8 1201D3 LCALL PONVI
000535 03CB E511 MOV A,11H ;Recuperamos tiempo
000536 03CD 1202CA IZQ20 LCALL DIEZMS ;Espera
000537 03D0 D5E0FA DJNZ ACC,IZQ20
000538 03D3 DAD8 DJNZ R2,IZQ02 ;A deslizar otro punto
000539 03D5 D083 POP DPH
000540 03D7 D082 POP DPL
000541 03D9 22 RET

```

**

Efecto : Visualiza el mensaje y lo borra deslizando hacia la derecha
Nombre : DERECHA
Código : 'D'

R2 : Nº de puntos horizontales del letrero

R1 : Nº de líneas a desplazar

R0 : Nº de bytes por línea

DPTR : Apunta a video

**

```

000542 03DA 1202DB DERECH LCALL NORMAL ;Visualiza el mensaje normalmente
000543 03DD C082 PUSH DPL
000544 03DF C083 PUSH DPH
000545 03E1 7A50 MOV R2,#50H ;Nº de líneas a desplazar
000546 03E3 900546 DER02 MOV DPTR,#0546H ;Apuntamos a VIDEO en RAM externa
000547 03E6 7907 MOV R1,#07H ;Nº de líneas a desplazar
000548 03E8 780A DER05 MOV R0,#0AH ;Nº de bytes por línea
000549 03EA C3 CLR C
000550 03EB E0 DER10 MOVX A,@DPTR ;Leemos el carácter
000551 03EC 13 RRC A ;Lo desplazamos con carry
000552 03ED F0 MOVX @DPTR,A ;Lo restituimos
000553 03EE 0582 INC DPL ;Apuntamos al siguiente
000554 03F0 D8F9 DJNZ R0,DER10 ;Hasta terminar la línea
000555 03F2 D9F4 DJNZ R1,DER05 ;A por la siguiente línea
000556 03F4 7F05 MOV R7,#05H ;Copiamos a RAM interna

```

```

000557 03F6 7E46          MOV R6,#46H
000558 03F8 1201D3        LCALL PONVI
000559 03FB E511          MOV A,11H          ;Recupera tiempo
000560 03FD 1202CA DER20  LCALL DIEZMS      ;Espera
000561 0400 D5E0FA        DJNZ ACC,DER20
000562 0403 DADE          DJNZ R2,DER02     ;A deslizar otro punto
000563 0405 D083          POP DPH
000564 0407 D082          POP DPL
000565 0409 22           RET

```

**

Efecto :Visualiza el mensaje y borra, varias veces

Nombre : PARPADEA

Código : 'P'

**

```

000566 040A 1202DB PARPA  LCALL NORMAL     ;Lo visualiza normalmente
000567 040D C082          PUSH DPL
000568 040F C083          PUSH DPH
000569 0411 E511    PAR10  MOV A,11H        ;recupera el tiempo
000570 0413 1202CA PAR20  LCALL DIEZMS     ;espera
000571 0416 D5E0FA        DJNZ ACC,PAR20
000572 0419 C28C          CLR 8CH          ;Apagamos el letrero
000573 041B 859012        MOV 12H,P1
000574 041E 759000        MOV P1,#00H
000575 0421 E511          MOV A,11H        ;Recupera el tiempo
000576 0423 1202CA PAR30  LCALL DIEZMS     ;esperamos
000577 0426 D5E0FA        DJNZ ACC,PAR30
000578 0429 851290        MOV P1,12H      ;Encendemos el letrero
000579 042C D28C          SETB 8CH
000580 042E D510E0        DJNZ 10H,PAR10  ;Nº de veces a parpadear
000581 0431 E511          MOV A,11H        ;Recupera el tiempo
000582 0433 1202CA PAR40  LCALL DIEZMS     ;Espera
000583 0436 D5E0FA        DJNZ ACC,PAR40
000584 0439 D083          POP DPH
000585 043B D082          POP DPL
000586 043D 22           RET

```

JUEGO DE CARACTERES

Tiene definidas la tramas de los caracteres

Se han definido los siguientes caracteres:

!"#\$%&'()*+,-./0123456789:;<=>?

ABCDEFGHIJKLMNÑOPQRSTUVWXYZ

abcdefghijklmnñopqrstuvwxyzáéíóú

Se puede definir el resto de los caracteres de la misma manera

Se pueden definir 256 caracteres, desde el código 00 al FF

Se empezaron a definir a partir de la dirección 5E0H de la memoria de programa (EEPROM) de modo que se inicia con el carácter espacio que tiene el código 20H y como cada código ocupa 7 bytes, entonces:

Solo son útiles los 6 bits de menor peso, los caracteres forman matrices de 7x6 bits

ORG	05E0H	COD. HEX	SIMBOLO
E_20	DB	00000000	00H ;'

	DB	00000000	00H	
	DB	00000000	00H	
	DB	00000000	00H	
	DB	00000000	00H	
	DB	00000000	00H	
	DB	00000000	00H	
E_21	DB	00001000	08H	;!;
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00000000	00H	
	DB	00001000	08H	
E_22	DB	00010100	14H	;"
	DB	00010100	14H	
	DB	00000000	00H	
	DB	00000000	00H	
	DB	00000000	00H	
	DB	00000000	00H	
	DB	00000000	00H	
E_23	DB	00010100	14H	;"
	DB	00010100	14H	
	DB	00111110	3EH	
	DB	00010100	14H	
	DB	00111110	3EH	
	DB	00010100	14H	
	DB	00010100	14H	
E_24	DB	00001000	08H	;\$
	DB	00011110	1EH	
	DB	00100000	20H	
	DB	00011100	1CH	
	DB	00000010	02H	
	DB	00111100	3CH	
	DB	00001000	08H	
E_25	DB	00000000	00H	;%
	DB	00110010	32H	
	DB	00110100	34H	
	DB	00001000	08H	
	DB	00010110	16H	
	DB	00100110	26H	
	DB	00000000	00H	
E_26	DB	00011000	18H	;&
	DB	00100100	24H	
	DB	00101000	28H	
	DB	00010000	10H	
	DB	00101010	2AH	
	DB	00100100	24H	
	DB	00011010	1AH	
E_27	DB	00000110	06H	;;
	DB	00000110	06H	
	DB	00000010	02H	
	DB	00000100	04H	
	DB	00000000	00H	
	DB	00000000	00H	
	DB	00000000	00H	
E_28	DB	00000100	04H	;(
	DB	00001000	08H	
	DB	00010000	10H	
	DB	00010000	10H	
	DB	00010000	10H	
	DB	00001000	08H	
	DB	00000100	04H	

E_29	DB	00010000	10H	;)
	DB	00001000	08H	
	DB	00000100	04H	
	DB	00000100	04H	
	DB	00000100	04H	
	DB	00001000	08H	
	DB	00010000	10H	
E_2A	DB	00001000	08H	;*;
	DB	00101010	2AH	
	DB	00011100	1CH	
	DB	00111110	3EH	
	DB	00011100	1CH	
	DB	00101010	2AH	
	DB	00001000	08H	
E_2B	DB	00000000	00H	;+
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00111110	3EH	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00000000	00H	
E_2C	DB	00000000	00H	;,
	DB	00000000	00H	
	DB	00000000	00H	
	DB	00011000	18H	
	DB	00011000	18H	
	DB	00001000	08H	
	DB	00010000	10H	
E_2D	DB	00000000	00H	;-
	DB	00000000	00H	
	DB	00000000	00H	
	DB	00111110	3EH	
	DB	00000000	00H	
	DB	00000000	00H	
	DB	00000000	00H	
E_2E	DB	00000000	00H	;:
	DB	00000000	00H	
	DB	00000000	00H	
	DB	00000000	00H	
	DB	00000000	00H	
	DB	00001100	18H	
	DB	00001100	18H	
E_2F	DB	00000000	00H	;/
	DB	00000010	02H	
	DB	00000100	04H	
	DB	00001000	08H	
	DB	00010000	10H	
	DB	00100000	20H	
	DB	00000000	00H	
E_30	DB	00011100	1CH	;0
	DB	00100010	22H	
	DB	00100110	26H	
	DB	00101010	2AH	
	DB	00110010	32H	
	DB	00100010	22H	
	DB	00011100	1CH	
E_31	DB	00011000	18H	;1
	DB	00101000	28H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	

	DB	00001000	08H	
	DB	00111110	3EH	
E_32	DB	00011100	1CH	;2
	DB	00100010	22H	
	DB	00000010	02H	
	DB	00000100	04H	
	DB	00001000	08H	
	DB	00010000	10H	
	DB	00111110	3EH	
E_33	DB	00111110	3EH	;3
	DB	00000100	04H	
	DB	00001000	08H	
	DB	00011100	1CH	
	DB	00000010	02H	
	DB	00100010	22H	
	DB	00011100	1CH	
E_34	DB	00000100	04H	;4
	DB	00001100	0CH	
	DB	00010100	14H	
	DB	00100100	24H	
	DB	00111110	3EH	
	DB	00000100	04H	
	DB	00000100	04H	
E_35	DB	00111110	3EH	;5
	DB	00100000	20H	
	DB	00111100	3CH	
	DB	00000010	02H	
	DB	00000010	02H	
	DB	00100010	22H	
	DB	00111110	1EH	
E_36	DB	00000110	06H	;6
	DB	00001000	08H	
	DB	00010000	10H	
	DB	00111100	3CH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00011100	1CH	
E_37	DB	00111110	3EH	;7
	DB	00000010	02H	
	DB	00000100	04H	
	DB	00001000	08H	
	DB	00010000	10H	
	DB	00100000	20H	
	DB	00100000	20H	
E_38	DB	00011100	1CH	;8
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00011100	1CH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00011100	1CH	
E_39	DB	00011100	1CH	;9
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00011110	1EH	
	DB	00000100	04H	
	DB	00001000	08H	
	DB	00110000	30H	
E_3A	DB	00000000	00H	::
	DB	00001100	0CH	
	DB	00001100	0CH	

	DB	00000000	00H	
	DB	00001100	0CH	
	DB	00001100	0CH	
	DB	00000000	00H	
E_3B	DB	00000000	00H	::
	DB	00001100	0CH	
	DB	00001100	0CH	
	DB	00000000	00H	
	DB	00001100	0CH	
	DB	00000100	04H	
	DB	00001000	08H	
E_3C	DB	00000100	04H	; <
	DB	00001000	08H	
	DB	00010000	10H	
	DB	00100000	20H	
	DB	00010000	10H	
	DB	00001000	08H	
	DB	00000100	04H	
E_3D	DB	00000000	00H	; =
	DB	00000000	00H	
	DB	00111110	3EH	
	DB	00000000	00H	
	DB	00111110	3EH	
	DB	00000000	00H	
	DB	00000000	00H	
E_3E	DB	00010000	10H	; >
	DB	00001000	08H	
	DB	00000100	04H	
	DB	00000010	02H	
	DB	00000100	04H	
	DB	00001000	08H	
	DB	00010000	10H	
E_3F	DB	00011100	1CH	; ?
	DB	00100010	22H	
	DB	00000010	02H	
	DB	00001100	0CH	
	DB	00001000	08H	
	DB	00000000	00H	
	DB	00001000	08H	
E_40	DB	00011100	1CH	; @
	DB	00100010	22H	
	DB	00101110	2EH	
	DB	00101010	2AH	
	DB	00101110	2EH	
	DB	00100000	20H	
	DB	00011110	1EH	
E_41	DB	00001000	08H	; A
	DB	00010100	14H	
	DB	00100010	22H	
	DB	00111110	3EH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
E_42	DB	00111100	3CH	; B
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00111100	3CH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00111100	3CH	
E_43	DB	00011100	1CH	; C

	DB	00100010	22H	
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00100010	22H	
	DB	00011100	1CH	
E_44	DB	00111100	3CH	;D
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00111100	3CH	
E_45	DB	00111110	3EH	;E
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00111100	3CH	
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00111110	3EH	
E_46	DB	00111110	3EH	;F
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00111100	3CH	
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00100000	20H	
E_47	DB	00011100	1CH	;G
	DB	00100010	22H	
	DB	00100000	20H	
	DB	00100110	26H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00011110	1EH	
E_48	DB	00100010	22H	;H
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00111110	3EH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
E_49	DB	00011100	1CH	;I
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00011100	1CH	
E_4A	DB	00001110	0EH	;J
	DB	00000100	04H	
	DB	00000100	04H	
	DB	00000100	04H	
	DB	00000100	04H	
	DB	00100100	24H	
	DB	00011000	18H	
E_4B	DB	00100010	22H	;K
	DB	00100100	24H	
	DB	00101000	28H	
	DB	00110000	30H	
	DB	00101000	28H	
	DB	00100100	24H	
	DB	00100010	22H	

E_4C	DB	00100000	20H	;L
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00111110	3EH	
E_4D	DB	00100010	22H	;M
	DB	00110110	36H	
	DB	00101010	2AH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
E_4E	DB	00100010	22H	;N
	DB	00110010	32H	
	DB	00101010	2AH	
	DB	00100110	26H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
E_4F	DB	00011100	1CH	;O
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00011100	1CH	
E_50	DB	00111100	3CH	;P
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00111100	3CH	
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00100000	20H	
E_51	DB	00011100	1CH	;Q
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00101010	2AH	
	DB	00100100	24H	
	DB	00011010	1AH	
E_52	DB	00111100	3CH	;R
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00111100	3CH	
	DB	00101000	28H	
	DB	00100100	24H	
	DB	00100010	22H	
E_53	DB	00011100	1CH	;S
	DB	00100010	22H	
	DB	00100000	20H	
	DB	00011100	1CH	
	DB	00000010	02H	
	DB	00100010	22H	
	DB	00011100	1CH	
E_54	DB	00111110	3EH	;T
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	

	DB	00001000	08H	
	DB	00001000	08H	
E_55	DB	00100010	22H	;U
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00011100	1CH	
E_56	DB	00100010	22H	;V
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00010100	14H	
	DB	00001000	08H	
E_57	DB	00100010	22H	;W
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00101010	2AH	
	DB	00010100	14H	
E_58	DB	00100010	22H	;X
	DB	00100010	22H	
	DB	00010100	14H	
	DB	00001000	08H	
	DB	00010100	14H	
	DB	00100010	22H	
	DB	00100010	22H	
E_59	DB	00100010	22H	;Y
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00010100	14H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
E_5A	DB	00111110	3EH	;Z
	DB	00000010	02H	
	DB	00000100	04H	
	DB	00001000	08H	
	DB	00010000	10H	
	DB	00100000	20H	
	DB	00111110	3EH	
E_61	DB	00000000	00H	;a
	DB	00000000	00H	
	DB	00011110	1EH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00011110	1EH	
E_62	DB	00100000	20H	;b
	DB	00100000	20H	
	DB	00111100	3CH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00111100	3CH	
E_63	DB	00000000	00H	;c
	DB	00000000	00H	

	DB	00011110	1EH	
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00011110	1EH	
E_64	DB	00000010	02H	;d
	DB	00000010	02H	
	DB	00011110	1EH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00011110	1EH	
E_65	DB	00000000	00H	;e
	DB	00000000	00H	
	DB	00011100	1CH	
	DB	00100010	22H	
	DB	00111110	3EH	
	DB	00100000	20H	
	DB	00011110	1EH	
E_66	DB	00000100	04H	;f
	DB	00001010	0AH	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00011110	1EH	
	DB	00001000	08H	
	DB	00001000	08H	
E_67	DB	00000000	00H	;g
	DB	00000000	00H	
	DB	00011110	1EH	
	DB	00100010	22H	
	DB	00011110	1EH	
	DB	00000010	02H	
	DB	00111110	3EH	
E_68	DB	00100000	20H	;h
	DB	00100000	20H	
	DB	00101100	2CH	
	DB	00110010	32H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
E_69	DB	00001000	08H	;i
	DB	00000000	00H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
E_6A	DB	00000010	02H	;j
	DB	00000000	00H	
	DB	00000010	02H	
	DB	00000010	02H	
	DB	00000010	02H	
	DB	00100010	22H	
	DB	00011100	1CH	
E_6B	DB	00100000	20H	;k
	DB	00100010	22H	
	DB	00100100	24H	
	DB	00101000	28H	
	DB	00110000	30H	
	DB	00101000	28H	
	DB	00100110	26H	

E_6C	DB	00001000	08H	;l
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
E_6D	DB	00000000	00H	;m
	DB	00000000	00H	
	DB	00110100	34H	
	DB	00101010	2AH	
	DB	00101010	2AH	
	DB	00101010	2AH	
	DB	00101010	2AH	
E_6E	DB	00000000	00H	;n
	DB	00000000	00H	
	DB	00101100	2CH	
	DB	00110010	32H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
E_6F	DB	00000000	00H	;o
	DB	00000000	00H	
	DB	00011100	1CH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00011100	1CH	
E_70	DB	00000000	00H	;p
	DB	00000000	00H	
	DB	00111100	3CH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00111100	3CH	
	DB	00100000	20H	
E_71	DB	00000000	00H	;q
	DB	00000000	00H	
	DB	00011110	1EH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00011110	1EH	
	DB	00000010	02H	
E_72	DB	00000000	00H	;r
	DB	00000000	00H	
	DB	00101100	2CH	
	DB	00110010	32H	
	DB	00100000	20H	
	DB	00100000	20H	
	DB	00100000	20H	
E_73	DB	00000000	00H	;s
	DB	00000000	00H	
	DB	00011110	1EH	
	DB	00100000	20H	
	DB	00011100	1CH	
	DB	00000010	02H	
	DB	00111100	3CH	
E_74	DB	00010000	10H	;t
	DB	00010000	10H	
	DB	00111000	38H	
	DB	00010000	10H	
	DB	00010000	10H	
	DB	00010010	12H	

	DB	00001100	0CH	
E_75	DB	00000000	00H	;u
	DB	00000000	00H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100110	26H	
	DB	00011010	1AH	
E_76	DB	00000000	00H	;v
	DB	00000000	00H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00010100	14H	
	DB	00001000	08H	
E_77	DB	00000000	00H	;w
	DB	00000000	00H	
	DB	00101010	2AH	
	DB	00101010	2AH	
	DB	00101010	2AH	
	DB	00101010	2AH	
	DB	00010100	14H	
E_78	DB	00000000	00H	;x
	DB	00000000	00H	
	DB	00100010	22H	
	DB	00010100	14H	
	DB	00001000	08H	
	DB	00010100	14H	
	DB	00100010	22H	
E_79	DB	00000000	00H	;y
	DB	00000000	00H	
	DB	00100010	22H	
	DB	00010100	14H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00111000	38H	
E_7A	DB	00000000	00H	;z
	DB	00000000	00H	
	DB	00111110	3EH	
	DB	00000100	04H	
	DB	00001000	08H	
	DB	00010000	10H	
	DB	00111110	3EH	

	ORG	0960H	COD. HEX	SIMBOLO
E_82	DB	00000110	06H	;é
	DB	00000000	00H	
	DB	00011100	1CH	
	DB	00100010	22H	
	DB	00111110	3EH	
	DB	00100000	20H	
	DB	00011110	1EH	
E_A0	DB	00000110	06H	;á
	DB	00000000	00H	
	DB	00011110	1EH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00011110	1EH	
E_A1	DB	00001100	0CH	;í

	DB	00000000	00H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
	DB	00001000	08H	
E_A2	DB	00000110	06H	;ó
	DB	00000000	00H	
	DB	00011100	1CH	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00011100	1CH	
E_A3	DB	00000100	04H	;ú
	DB	00001000	08H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100110	26H	
	DB	00011010	1AH	
E_A4	DB	00011100	1CH	;ñ
	DB	00000000	00H	
	DB	00101100	2CH	
	DB	00110010	32H	
	DB	00100010	22H	
	DB	00100010	22H	
	DB	00100010	22H	
E_A5	DB	00011100	1CH	;Ñ
	DB	00000000	00H	
	DB	00100010	22H	
	DB	00110010	32H	
	DB	00101010	2AH	
	DB	00100110	26H	
	DB	00100010	22H	

tadoras con una potencia de 280 W.

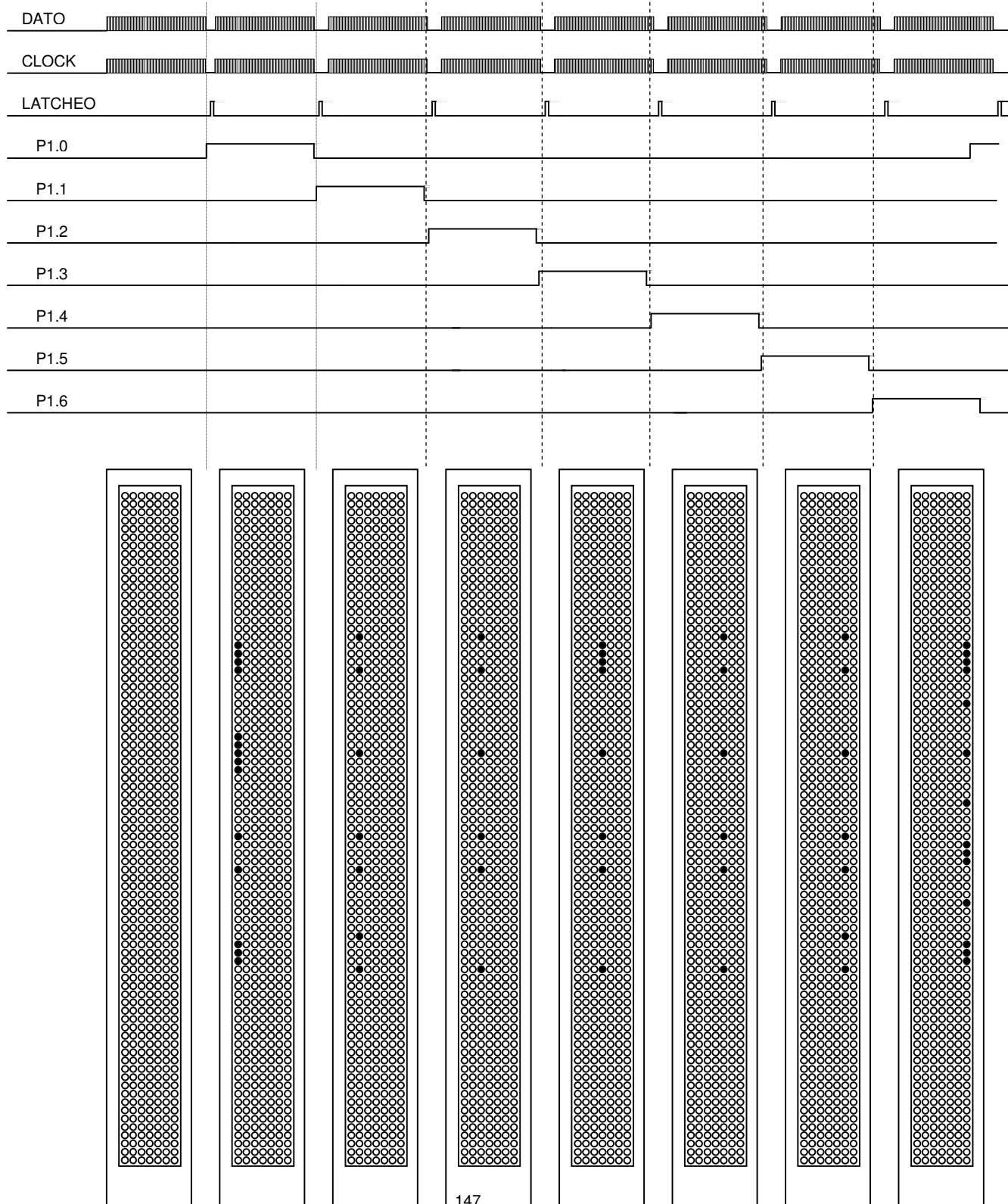


Figura 35. Diagrama de tiempos

