



T
621.3885
C03

ESCUELA SUPERIOR POLITECNICA DEL LITORAL

Facultad de Ingeniería Eléctrica y Computación

**“DISEÑO DE UNA INTERFASE PARA SENSAR NIVELES DE AUDIO, VIDEO,
Y ALIMENTACION DE ENERGIA EN UNA REPETIDORA DE TV
EMPLEANDO EL SISTEMA DE TRANSMISION CDPD”**

TOPICO DE GRADUACION

Previo a la obtención del título de:

**INGENIERO ELECTRICO
ESPECIALIZACION ELECTRONICO**

Presentado por:

**ROBERTO COBO VELEZ
FREDDY HARO HARO**

GUAYAQUIL-ECUADOR

1998

AGRADECIMIENTO



D-18738

A TODOS QUIENES COLABORARON
DESINTERESADAMENTE
PRESTANDOME SU VALIOSO TIEMPO
Y LA INFRAESTRUCTURA
NECESARIA CON EL FIN DE
AYUDARME A SALIR ADELANTE
CON ESTA TESIS.

DEDICATORIA

MIS PADRES

A MIS HERMANOS

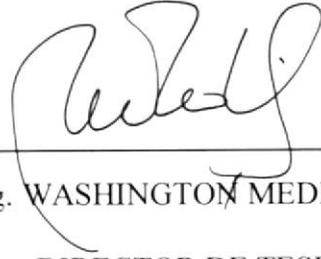
A MIS COMPAÑEROS

DE ESTUDIO

TRIBUNAL DE GRADUACION



Ing. ARMANDO ALTAMIRANO
PRESIDENTE DEL TRIBUNAL



Ing. WASHINGTON MEDINA M.
DIRECTOR DE TESIS



Ing. FABRICIO VELEZ G.

VOCAL



Ing. REBECA ESTRADA

VOCAL

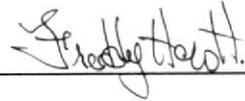
DECLARACION EXPRESA

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

(Reglamento de Graduación de la ESPOL).



ROBERTO COBO VELEZ



FREDDY HARO

RESUMEN

El presente proyecto establece una opción para solucionar el complejo problema de la pérdida de la señal de la transmisión de televisión debido a la falla en alguno de los equipos (audio, video y alimentación de energía) que se encuentran en las estaciones repetidoras (zonas alejadas) y que no se pueden detectar enseguida en la estación central, y de acuerdo a dicha necesidad está fundamentado en ser un trabajo base para la optimización de servicios en transmisión de señales de televisión en un futuro no muy lejano en la empresa al momento de iniciarse.

Se divide en dos secciones: el diseño de la interfase y los sensores; y la programación para comprobar la transmisión que ejecuta la interfase vía módem nulo y a su vez la simulación vía módem en el sistema de comunicación inalámbrico CDPD.

La primera sección enfrenta la meta de diseñar una interfase que permita receptor la señal de los sensores y luego transmitir las en el sistema de comunicación inalámbrica CDPD, evitando la inversión de una computadora y recurriendo a un sistema de comunicación más rápido y efectivo en su operabilidad, de esta manera el proyecto también propone explotar aún más los equipos instalados en la estación central.

La segunda sección es emplear un lenguaje de programación para el funcionamiento y operabilidad de la primera, y establecer un método base que permita crear una base de datos con la información de cada una de las estaciones repetidoras que posee una empresa televisiva.

La solución al problema no es algo fácil, como se puede apreciar, pero es indispensable para el desarrollo de la empresa.

INDICE GENERAL

RESUMEN	II
INDICE GENERAL	III
INDICE DE FIGURAS	IV
INDICE DE TABLAS	V
INTRODUCCION	2

CAPITULO I

CONCEPTOS BASICOS DE UN SISTEMA CDPD DENTRO DEL SISTEMA DE COMUNICACION CELULAR

1.1. Definición del sistema celular.	4
1.1.1. Características del sistema celular.	5
1.2. ¿Qué es CDPD?	6
1.3. Topología típica de una red CDPD.	7
1.4. Comunicación del sistema CDPD en los sistemas AMPS.	11
1.4.1. Comunicación del sistema CDPD a través del área entre celdas.	14
1.5. Comparación con otros servicios de comunicaciones.	18
1.5.1. Comparación con respecto a sus estaciones móviles.	19

1.5.2. Comparación con respecto a sus estaciones bases.	20
1.6. Ventajas del sistema CDPD.	20
1.6.1. Con respecto a los módems celulares.	20
1.6.2. Con respecto a la red privada de radio.	21
1.7. Variedad de los CDPD.	22

CAPITULO II

EMISIONES TELEVISIVAS

2.1. Asignación de frecuencias.	23
2.2. Intensidad de campo y características de propagación.	25
2.3. Normas de transmisión.	26
2.4. Transmisores.	33
2.5. Sistemas de antenas.	35
2.5.1. Consideraciones de las antenas de transmisión.	37
2.5.2. Tipo de antenas de transmisión.	39

CAPITULO III

VISION GLOBAL DEL PROYECTO

3.1. La red de televisión.	41
3.2. El medio de comunicación.	43

3.3. El UBIQUITY.	47
3.3.1. Comunicación entre su computador y el software de control.	47
3.3.2. Código de transmisión.	48
3.3.3. Interfase con el usuario.	51
3.3.4. Cambiando los modos de operación con los comandos AT.	53
3.4. Modos del sistema de comunicación.	55
3.5. Comandos para los modos de operación.	56

CAPITULO IV

CIRCUITOS SENSORES

4.1. Circuito detector de video.	58
4.2. Circuito detector de audio.	59
4.3. Circuito detector de alimentación AC.	60
4.4. Interfase de conexión entre los sensores y el módem CDPD.	60
4.4.1. Interfase con el microcontrolador 8752.	60
4.4.2. Pastillaje de los microcontroladores 8752.	61
4.4.3. Tipo de comunicación.	64
4.4.4. Integrado MAX232.	66
4.4.5. Interfase de conexión.	68
4.5. Descripción de las pruebas.	69
4.5.1. Primera prueba.	69

4.5.2. Segunda prueba.	70
------------------------	----

CAPITULO V

PROGRAMAS DE EJECUCION

5.1. Programa en lenguaje Visual Basic Versión 4.0 que comunica dos computadoras y permite ejecutar la transmisión entre ellos.	72
5.1.1. Programa para el servidor.	73
5.1.2. Programa para el cliente.	87
5.2. Programa en lenguaje Visual Basic Versión 4.0 que permite la comunicación desde la interfase a la computadora.	106
5.2.1. Programación para el servidor.	107
5.2.2. Programa en lenguaje ensamblador para el microcontrolador 87C52.	121
5.3. Programa en Visual Basic Versión 4.0 que permite simular la comunicación a través de módem con la interfase 87C52 al que se encuentran conectados.	127
5.3.1. Programa para el servidor.	128
5.3.2. Programa en lenguaje ensamblador para el microcontrolador 87C52.	143

Conclusiones	147
Recomendaciones	149
Anexos	150
Bibliografía	155

INDICE DE TABLAS

Tabla I	Comparación del servicio del CDPD con otros servicios de transmisión.	18
Tabla II	Comparación con respecto a las estaciones móviles.	19
Tabla III	Comparación con respecto a las estaciones bases.	20
Tabla IV	Canales de televisión y bandas de frecuencia.	24
Tabla V	Potencia máxima en un transmisor.	27
Tabla VI	Modos del sistema de comunicaciones del UBIQUITY.	55
Tabla VII	Comandos para los modos de operación AT.	56

INDICE DE FIGURAS

Fig. # 1	Celda típica del sistema de comunicación celular.	4
Fig. # 2	Topología típica de una red CDPD.	9
Fig. # 3	Transmisión de datos en un sistema CDPD a través de un sistema AMPS.	12
Fig. # 4	Otra manera de apreciar la actividad de la Fig. # 3.	14
Fig. # 5	Comunicación de un sistema CDPD a través de celdas.	15
Fig. # 6	Intensidad de campo en los canales del 2 al 6.	25
Fig. # 7	Espectro típico.	27
Fig. # 8	Sincronización para transmisión a color.	29
Fig. # 9	Sincronización para transmisión a blanco y negro.	30
Fig. # 10	Premodulación standard de señales de televisión a color.	32
Fig. # 11	Transmisor de televisión.	34
Fig. # 12	Factores en el diseño de una antena transmisora.	36
Fig. # 13	Visión global del proyecto.	41
Fig. # 14	Esquema de la red de transmisión de Telecentro.	42
Fig. # 15	Cobertura de BELLSOUTH.	45
Fig. # 16	Cobertura de BISMARCK.	46
Fig. # 17	Patillaje del microcontrolador 87C52.	62
Fig. # 18	Patillaje del integrado MAX 232.	67

Fig. # 19	Ilustración de la conexión entre dos computadoras. Vía módem nulo.	72
Fig. # 20	Ilustración de la conexión entre la computadora y la interfase que contiene al 87C52. Vía módem nulo.	106
Fig. # 21	Ilustración de la conexión entre la computadora y la interfase con los sensores. Vía módem nulo.	127

INTRODUCCION

El presente trabajo es un "Diseño de una interfase para medir niveles de audio, video y energía en una estación repetidora de TV empleando el sistema CDPD, el cual determina la capacidad y el gran potencial que poseen los microcontroladores y los sensores para reducir costos y tener más eficiencia en los instrumentos que transmiten este tipo de señales. Además se ha empleado el sistema CDPD por poseer gran capacidad de transmisión de datos a una velocidad optima de 9.600 bps. El diseño de la interfase está basado en los conocimientos adquiridos a través de nuestros estudios y su simulación en el sistema CDPD, en la experiencia en los técnicos de la empresa dedicada a este servicio. Este trabajo empleó elementos adquiridos en el País y los módems que utilizamos los distribuye y alquila la empresa BISMARK.

En nuestro País se han realizado estudios sobre los diversos potenciales que poseen los sistemas de comunicación inalámbricos, pero casi nadie ha incursionado en este tipo de servicio para las empresas dedicadas a la transmisión de señales de televisión.

CAPITULO I

CONCEPTOS BASICOS DE UN SISTEMA CDPD (CELLULAR DIGITAL PACKET DATA) DENTRO DEL SISTEMA DE COMUNICACIÓN CELULAR

Hoy en día, en el mundo de los negocios, aquel que responde mucho más rápido y más eficazmente que la competencia, tiene una ventaja estratégica. Lo que conlleva un flujo de datos que pueden ir desde lo más simple como un inventario, un estado de órdenes de los clientes, o el mapa de emergencia de un edificio, etc., haciendo que el transporte de estos datos a todas partes sea cada vez más posible. En ese momento es donde la red móvil de datos hace que la atención de los requerimientos de un cliente sea más eficaz e inmediata. No hay la necesidad de regresar a la oficina a buscar los datos en los archivos o en una computadora.

Otra ventaja es la reducción de errores al ingresar la información, la que puede ser enviada en el lugar de la transacción. Como por ejemplo, la policía podría redactar sus informes en el lugar de los hechos, en donde los testigos pueden corroborar los datos ingresados; Los vendedores pueden ingresar las órdenes de sus clientes e

inmediatamente recibir una confirmación del producto. La mayor ventaja es la de tener información actualizada, en el momento que se la necesite.

1.1. DEFINICION DEL SISTEMA DE COMUNICACIÓN CELULAR

Para enlazar una comunicación en un Sistema Celular se debe tener en cuenta cuales son los elementos que intervienen en el y sus propósitos, y es sobre la base de esta premisa que describiremos los elementos de una celda típica

(Fig. # 1):

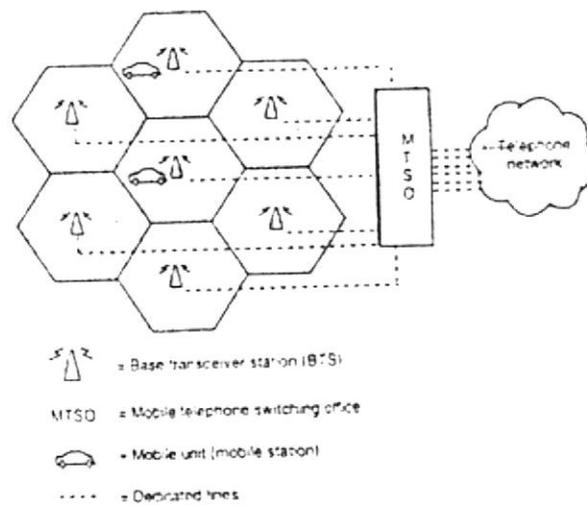


Fig. # 1. Celda Típica del Sistema de Comunicación Celular

- La Celda, su tamaño físico depende esencialmente de su población o tráfico de llamadas, que define el subconjunto fijo de canales de radio del cual dispondría, el tamaño de la celda oscila entre 0.5 y 50 Km., dependiendo del sistema

- La Oficina de conmutación de telefonía móvil (MTSO), que es el elemento de control responsable de la conmutación de las llamadas a celdas, abasteciéndolo para reserva, interconexiones con redes telefónicas, monitoreo del tráfico para actividades cargadas o densas, pruebas de funcionamiento y diagnóstico de servicios, y administración de toda la red.
- La Celda y su Estación Transceptora (BTS), donde se emplea 1 transmisor, 2 receptores por canal, un controlador, un sistema de antena, y además es donde se interrelacionan la Estación Móvil y la MTSO. Se toman las señales receptoras y direccionan desde el MTSO, el BTS las envía y recibe el tráfico a/desde la estación móvil.
- La Unidad Móvil o Estación Móvil, es actualmente el transceptor móvil que está instalado en un automóvil, camión, etc. Este contiene una máquina de frecuencia aguil que le permite cambiar a una frecuencia particular designada para su uso por el MTSO.

1.1.1. CARACTERÍSTICAS DEL SISTEMA CELULAR

Divide el área geográfica relativamente grande en secciones más pequeñas llamadas celdas o células, que se superponen en los bordes con

las celdas adyacentes para una cobertura plena, especialmente en el cambio de celda a celda.

Emplea transmisores de niveles de baja potencia y las frecuencias pueden ser divididas en bandas o celdas con zonas de protección predeterminadas para evitar posibles interferencias y de esta manera poder rehusar las mismas.

1.2. ¿QUÉ ES CDPD?

Como su nombre lo indica CDPD (Cellular Digital Packet Data), es utilizado en una red móvil celular, lo que significa que la red CDPD es una conexión de unidades móviles dentro de un área geográfica. En sí la capacidad de cobertura del CDPD es la misma que la de una red celular.

CDPD emplea la capacidad de voz no utilizada (idle time o idle voice channel) del Sistema de Telefonía Móvil Avanzado (AMPS) para suministrar transmisión de datos.

El CDPD es compatible con los protocolos de Internet (IP) y con los Protocolos de Redes Conectadas (CLNP), las Extensiones de Transportación OSI, el Protocolo de Control de Transmisión (TCP), y el recientemente creado Internet

Protocol versión 6 (IPV6). Esta característica habilita al suscriptor de CDPD a utilizar la red con las aplicaciones populares de Internet.

La industria de la Comunicación de datos está continuamente en desarrollo, por lo que el CDPD garantiza el soporte a los cambios de tecnología de la red, como la adopción del Circuit switched CDPD en 1995, la evolución en áreas como el acceso a tecnologías de algoritmos de seguridad de acuerdo a sus necesidades.

La capacidad de seguridad que provee el CDPD es integral en el sistema. Esta capacidad de seguridad incluye confidencialidad para los dos usuarios y los datos transmitidos, autenticación de usuarios, y claves de usuarios.

1.3. TOPOLOGIA TIPICA DE UNA RED CDPD:

La arquitectura del CDPD esta basada en el sistema OSI, y se deriva del ISO 7498 y las Recomendaciones ITU-T's OSI X.200. Pero sus creadores han usado la terminología y los conceptos OSI hasta donde fue posible, pero los protocolos que se han venido usando hasta ahora están basados en TCP/IP y otros protocolos relacionados a este.

Como vemos en la Figura # 2 los proveedores de servicios CDPD emplean 3 interfaces:

- Interface de enlace aéreo (A): es entre el proveedor de servicios y el suscriptor móvil.
- Interface Externa (E): es entre el proveedor de servicios y las redes externas
- Interface del proveedor de servicios (I): se da entre la cooperación de los proveedores de servicios CDPD.

Dos entidades básicas de red existen en esta arquitectura:

- El Sistema Móvil Final (M-ES), también llamado Huésped (Host) en terminología de Internet. Cada M-ES debe ser identificado al menos con un único Identificador de Entidad de Red (NEI). Durante el enlace de comunicación el NEI es verificado, de otra manera el M-ES no puede decodificar el data enviado desde la red CDPD. Cada M-ES debe pertenecer a un área local fija.

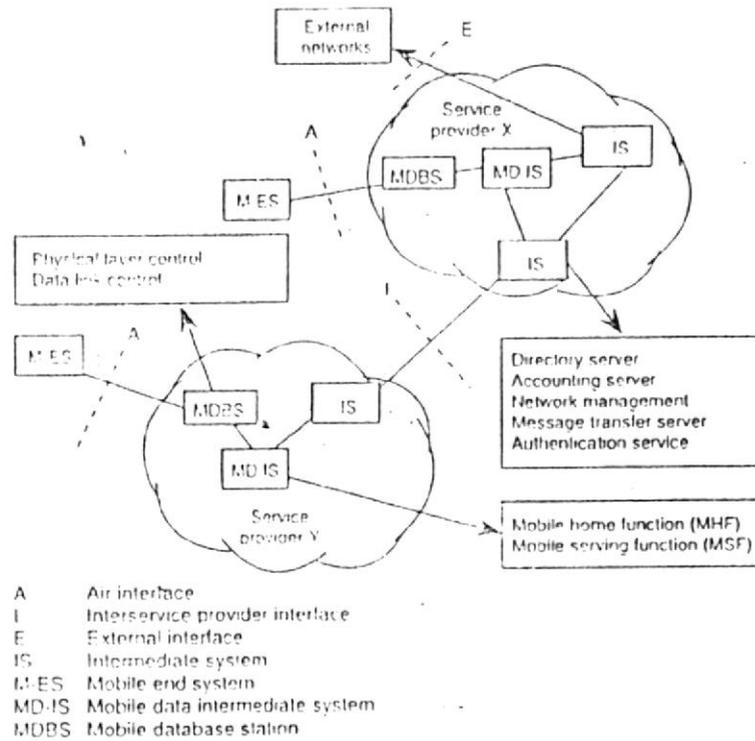


Fig. # 2. Topología Típica de una Red CDPD

- El Sistema Intermedio (IS) es una unidad de trabajo en Internet, llamado Ruteador (Router) en terminología Internet, este también corre una operación definida –CDPD denominada Protocolo de Ubicación de Redes Móviles (MNLP), el que suministra información de ubicación en los sistemas. Este protocolo posteriormente se lo nombró Sistema Intermedio de Data Móvil (MD-IS).

El MD-IS es la única entidad que tiene un reconocedor de la movilidad del Ess.

Este administra la movilidad a través de dos servicios de ruta.

La Función Móvil Local (MHF) provee un paquete de funciones de reenvío, y suministra una base de datos para su área de servicio (su ISs y ESs locales), y siempre mantiene el rastro de la información del M-ES

La Función de Servicio Móvil (MSF) maneja el servicio de transferencia de paquetes para visitantes MSs. Como en muchos sistemas móviles, un M-ES visitante debe registrarse con el servidor MD-IS, el cuál notifica al ES's MD-IS visitante de su común ubicación.

La Estación Base de Datos Móvil (MDBS) crea la interface aérea hacia el M-ES. Este reside en un sitio de la celda AMPS, y usa los equipos de recepción y transmisión AMPS. Este debe transportar el data desde el M-ES en paquetes y reenviárselos al MD-IS para rutas adicionales.

Si el momento de la sesión se pierde el enlace los datos son almacenados y luego retransmitidos un cierto número de veces. Si la comunicación no se pudo volver a realizar la información almacenada es descartada, luego de cumplirse el número de retransmisiones permitidas.

1.4. COMUNICACIÓN DEL SISTEMA CDPD EN LOS SISTEMAS AMPS

El CDPD comparte los equipos y los canales ubicados en las celdas del sistema AMPS, utiliza estos canales como canales de reenvío y de reversa.

El canal de Reenvío es desde el MDBS hasta el M-ES, se lo emite desde el MDBS hacia todas las M-ESs en las celdas o sectores de celdas, esta siempre disponible hasta que recibe contestación.

El canal de Reversa es desde el M-ES hasta el MDBS, en contraste con el canal de reenvío, este debe ser compartido por más de un M-Es y es por ello que posee un protocolo de contención activado para administrar el tráfico de transmisión.

CDPD es una tecnología TDMA que usa espacios de tiempos cortos sobre un canal de voz AMPS cuando no hay actividad de habla en ese canal, pero cuando el canal de habla comienza a activarse el MDBS corta el canal de reenvío y salta a un canal idle, entonces los M-ESs son informados que el canal de reenvío ha sido trasladado a otro canal, esto permite a los M-ESs sintonizar un canal particular, sincronizar sus actividades y continuar la comunicación.

En la Fig. # 3 tenemos una vista general de como las actividades son coordinadas entre el MDBS y el M-ES y el mismo que describiremos a continuación:

En el evento 1, el M-ES esta enviando su data por el canal AMPS número 5.

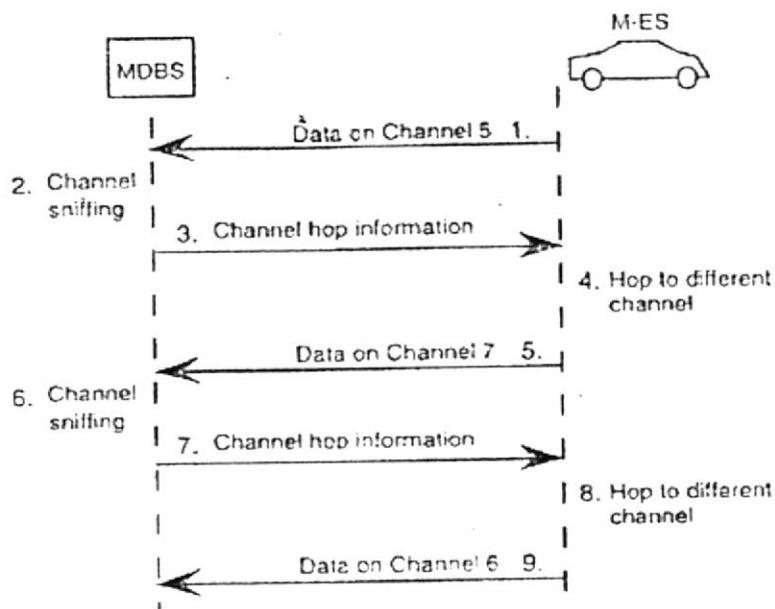


Fig. # 3. Transmisión de data en un Sistema CDPD a través de un Sistema AMPS

En el evento 2, el MDBS, en cambio, esta monitoreando todos los canales mediante el procedimiento de canal de revisión. Durante esta actividad, el MDBS detecta si una conexión de habla esta comenzando a usar el mismo canal que esta ocupando él trafico CDPD, si esto se produce pasa al evento 3.

En el evento 3, el MDBS da las direcciones para que se haga un salto a un canal diferente.

En el evento 4, se produce el salto.

Los saltos de canal requieren que el M-ES vuelva a sincronizar su señal y de esta manera comience a enviar data por otro canal. Esta acción esta descrita en el evento 5 donde el M-ES ha saltado al canal # 7.

En los eventos 6 hasta el 9 se muestran aun otros saltos de canal.

El MDBS detecta el comienzo de una actividad de habla al revisar los niveles bajos de radio frecuencias en el lado de la transmisión del canal AMPS. Obviamente, el MDBS debe trabajar forzosamente en un tiempo muy apretado hasta que se analicen todas las llamadas en la celda o todas las llamadas en un sector.

Las revisiones pueden ser recursos en banda corta o banda ancha:

La revisión en banda corta examina cada canal de 30 kHz, uno a la vez.

La revisión en banda ancha analiza toda la radio frecuencia en el espectro de los 12.5 MHz.

En cualquiera de los dos casos el trabajo del MDBS es ubicar el trafico de data en un canal de habla no utilizado.

En la Fig. # 4 se describe otra forma de apreciar la actividad descrita en la Fig. #

3. En este gráfico, el M-ES esta saltando desde el canal 5 hasta el canal 7 y luego

al canal 6. Esto permite la continuidad y la mezcla de los tráficos de habla en el AMPS y el tráfico de data del CDPD.

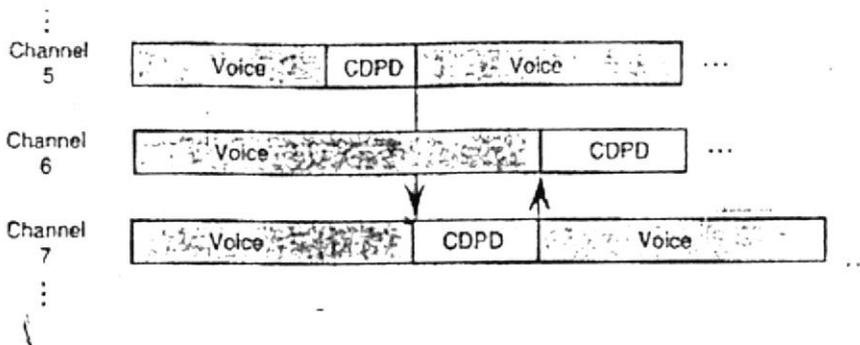


Fig. # 4. Otra manera de apreciar la actividad de la Fig. #3. (Los saltos)

El M-ES y el MDBS tienen una ligera diferencia en sus métodos de rampa ascendente y rampa descendente durante los canales de salto. Para el canal de RF de reenvío, el MDBS tiene un tiempo de rampa ascendente y rampa descendente de 10 ms. Para el canal de reversa, el M-ES tiene un tiempo de rampa ascendente y rampa descendente de 2 ms.

1.4.1. COMUNICACIÓN DEL SISTEMA CDPD A TRAVÉS DEL AREA ENTRE CELDAS.

Para cuando la estación móvil viaja a través de las celdas, CDPD suministra un conjunto de procedimientos para que el M-ES se registre

en un nuevo MD-SI. Estas operaciones se las muestran en la Fig. # 5 bajo la siguiente descripción:

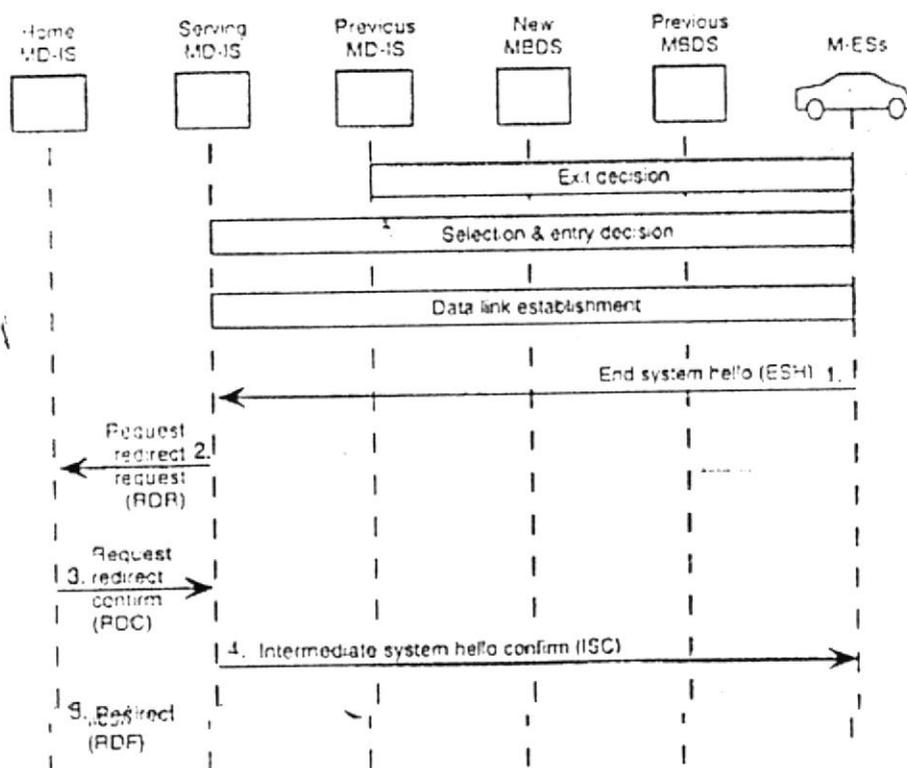


Fig. # 5. Comunicación de un Sistema CDPD a través de celdas

Las 3 primeras áreas marcadas como decisión de salida, decisión de selección y entrada y el establecimiento del enlace de data han sido descritas anteriormente. Daremos esta explicación con el M-ES enviando un sistema final Hola (ESH, mostrado en el evento 1) hacia un nuevo servicio MD-IS. La función de este mensaje Hola es para informar en el nuevo MD-SI de la presencia de la estación móvil y

registrar la dirección de la misma. Este mensaje 2efe contener la dirección única del M-ES, el mismo que es conocido como NEI, y que puede ser ya sea una dirección de las extensiones de redes OSI o una dirección convencional IP en Internet.

En el evento 2, al recibir este mensaje el nuevo servicio de MD-IS envía una unidad de datos al MD-IS de origen en protocolo de solicitud de redireccionamiento (RDR). Este mensaje contiene la dirección de origen del M-ES así como una dirección de red de reenvío. Esta dirección mas tarde es la dirección de la entidad de reenvío y del servicio MD-IS y con la cual el MD-IS de origen le dirá donde el mensaje 2efe ser redireccionado.

En el evento 3, el MD-IS de origen responderá con un mensaje confirmando la solicitud de redireccionamiento al nuevo servicio MD-IS y este le comunicara si los receptores ya obtuvieron el registro.

En el evento 4, el nuevo servicio MD-IS regresara la confirmación del sistema intermedio Hola (ISC) al sistema móvil final, esto se efectúa para confirmar que los registros se dieron realmente.

En el evento 5, la actividad final, es el flujo de redireccion (RDF PDU), que se lo envía al MD-IS de origen para notificar de antemano el

servicio del MD-IS, este mensaje no puede ser más grande que el de reenvío para su ubicación.

1.5. COMPARACION CON OTROS SERVICIOS DE COMUNICACION

	AMPS	CDPD	METRO
TIPO	Celular	Celular	Paging
ACCESO MULTIPLE	FDMA	FH / Paquetes	Simplex
BANDA DE FRECUENCIA	824 – 894 MHz	824 – 894 MHz	Varias
MODULACION	FM	QPSK / BPSK	FSK
ANCHO DE BANDA	30 KHz	30 KHz	12.5 KHz
VELOCIDAD DE TRANSMISION	9600 bps	19200 bps	512 / 1200 bps
PROTOCOLO	IS – 41 y OSI	TCP / IP y MDLP y X.25	POCSAC
COVERTURA	Red Celular	Red Celular	Red Paging
VOLUMEN DE DATOS	MEDIO – ALTO	POCO – MEDIO	POCO
COMPLEJIDAD DEL EQUIPO	MEDIO	MEDIO – ALTO	POCO
COMPLEJIDAD PARA EL USUARIO	FACIL	MEDIA – ALTA	FACIL

Tabla. I Comparación del sistema CDPD con otros sistemas de transmisión.

Además CDPD posee:

- Una eficiencia de espectro de 0.64 (b / Hz)
- Operabilidad en desvanecimiento, soporta 2,2 msg.
- Estrategia para el error por estallido, usa RS 63,47 (6 bits por simbolo)
- Estrategia para el error random, taparlo con protectores de estallido.

1.5.1. COMPARACION CON RESPECTO A SUS ESTACIONES MOVILES.

	TELEFONIA FIJA	TELEFONIA CELULAR (CDPD)	SISTEMA PAGING
RANGO DE COVERTURA	Baja	Alta	Alta
COMPLEJIDAD	Moderada	Alta	Poca
FREC. PORTADORA	< 100 MHz	< 1 GHz	< 1 GHz
OPERABILIDAD	Transceptor	Transceptor	Receptor

Tabla. II Comparación con respecto a sus estaciones móviles.

1.5.2. COMPARACION CON RESPECTO A SUS ESTACIONES BASES.

	TELEFONIA FIJA	TELEFONIA CELULAR (CDPD)	SISTEMA PAGING
RANGO DE COVERTURA	Baja	Alta	Alta
COMPLEJIDAD	Baja	Alta	Alta
FREC. PORTADORA	< 100 MHz	< 1 GHz	< 1 GHz
OPERABILIDAD	Transceptor	Transceptor	Transmisor

Tabla III Comparación con respecto a sus estaciones bases.

1.6. LAS VENTAJAS DEL SISTEMA CDPD

1.6.1. CON RESPECTO A LOS MODEM CELULARES.

El sistema celular es un método fácil de usar, pero para activarlo se necesita de una línea celular, un computador, y el respectivo modem. Cuando un usuario quiere acceder a la red de información, tiene que conectar su LAPTOP al modem a través de la interface requerida, y luego buscar el acceso a la red de información, usando la línea celular.

Además, para recibir datos y luego transmitirlos el usuario tiene que reiniciar el acceso a la red de información lo que hace que este sistema dependa de la efectividad de la conexión.

En cambio en el sistema CDPD, una vez que el usuario es registrado, se convierte en un modo activo de la red de información. Esto significa que cualquier data enviado enviado por un usuario móvil es inmediatamente registrado.

Otra ventaja se basa en que el sistema CDPD es medido en términos de cantidad de datos transferidos y no de la duración de la llamada como en la telefonía celular.

1.6.2. CON RESPECTO A LA RED PRIVADA DE RADIO.

Al crear una red privada de transmisión de datos, se esta involucrando el desarrollo de una infraestructura de transmisiones de RF (gran número de radios bases, licencias de radio, etc.), lo que da como resultado una inversión mucho mayor, al incrementarse el capital invertido al inicio y el costo de funcionamiento del sistema.

Por su parte el uso del sistema CDPD posee una mayor cobertura a menor costo, durante las 24 hrs. del día, ya que emplea la red celular ya existente.

1.7. VARIEDADES DE LOS MODEMS CDPD.

Existe una gran variedad de módems CDPD, como se puede observar en el gráfico adjunto; desde los más fáciles de transportar, hasta aquellos que pueden ser instalados en los carros u oficinas, que pueden ser colocados sobre su LAPTOP, o miniprosesor con módem incluido.

CAPITULO II

2.1. EMISIONES TELEVISIVAS

Las emisiones televisivas consisten en transmisiones de imagen y sonido para el público general en las bandas de frecuencias muy alta (VHF, de very –high – frequency) y ultraalta (UHF, de ultrahigh-frequency). En el sistema compuesto se utilizan portadoras separadas para la imagen, el sonido y la información del color, diseñadas de forma que sea posible obtener recepción compatible en un receptor de blanco y negro o de color.

2.1.1. ASIGNACION DE FRECUENCIAS

En Estados Unidos, las asignaciones de frecuencias se dividen en 68 canales en las bandas VHF y UHF. Para la VHF, los canales del 2 al 13 incluyen de 54 a 72, de 76 a 88, y de 174 a 216 MHz. Para la UHF, los canales del 14 al 69 abarcan de 470 a 806 MHz, y los canales del 70 al 83 incluyen de 806 a 890 MHz. Cada canal consta de una banda de frecuencia de 6 MHz que contiene modulación de imagen, de audio de

color y posiblemente una modulación especial utilizada por espectadores sordos.

En la siguiente tabla se presentan las designaciones del canal y de la banda de frecuencia. Algunos de los canales son compartidos con otros servicios, incluyendo la telemetría biomédica, radio móvil y radioastronomía.

Canal núm.	Banda de frecuencia (MHz)	Canal núm.	Banda de frecuencia (MHz)
2	54-60	43	644-650
3	60-66	44	650-656
4	66-72	45	656-662
5	76-82	46	662-668
6	82-88	47	668-674
7	174-180	48	674-680
8	180-186	49	680-686
9	186-192	50	686-692
10	192-198	51	692-698
11	198-204	52	698-704
12	204-210	53	704-710
13	210-216	54	710-716
14	470-476	55	716-722
15	476-482	56	722-728
16	482-488	57	728-734
17	488-494	58	734-740
18	494-500	59	740-746
19	500-506	60	746-752
20	506-512	61	752-758
21	512-518	62	758-764
22	518-524	63	764-770
23	524-530	64	770-776
24	530-536	65	776-782
25	536-542	66	782-788
26	542-548	67	788-794
27	548-554	68	794-800
28	554-560	69	800-806
29	560-566	70	806-812
30	566-572	71	812-818
31	572-578	72	818-824
32	578-584	73	824-830
33	584-590	74	830-836
34	590-596	75	836-842
35	596-602	76	842-848
36	602-608	77	848-854
37	608-614	78	854-860
38	614-620	79	860-866
39	620-626	80	866-872
40	626-632	81	872-878
41	632-638	82	878-884

Tabla IV Canales de televisión y bandas de frecuencia.

2.2. INTENSIDAD DE CAMPO Y CARACTERÍSTICAS DE PROPAGACION

La intensidad de campo de un transmisor de emisiones televisivas depende de la distancia, altura de la antenna, terreno local, frecuencia y potencia radiada efectiva. (Ver Fig. # 6)

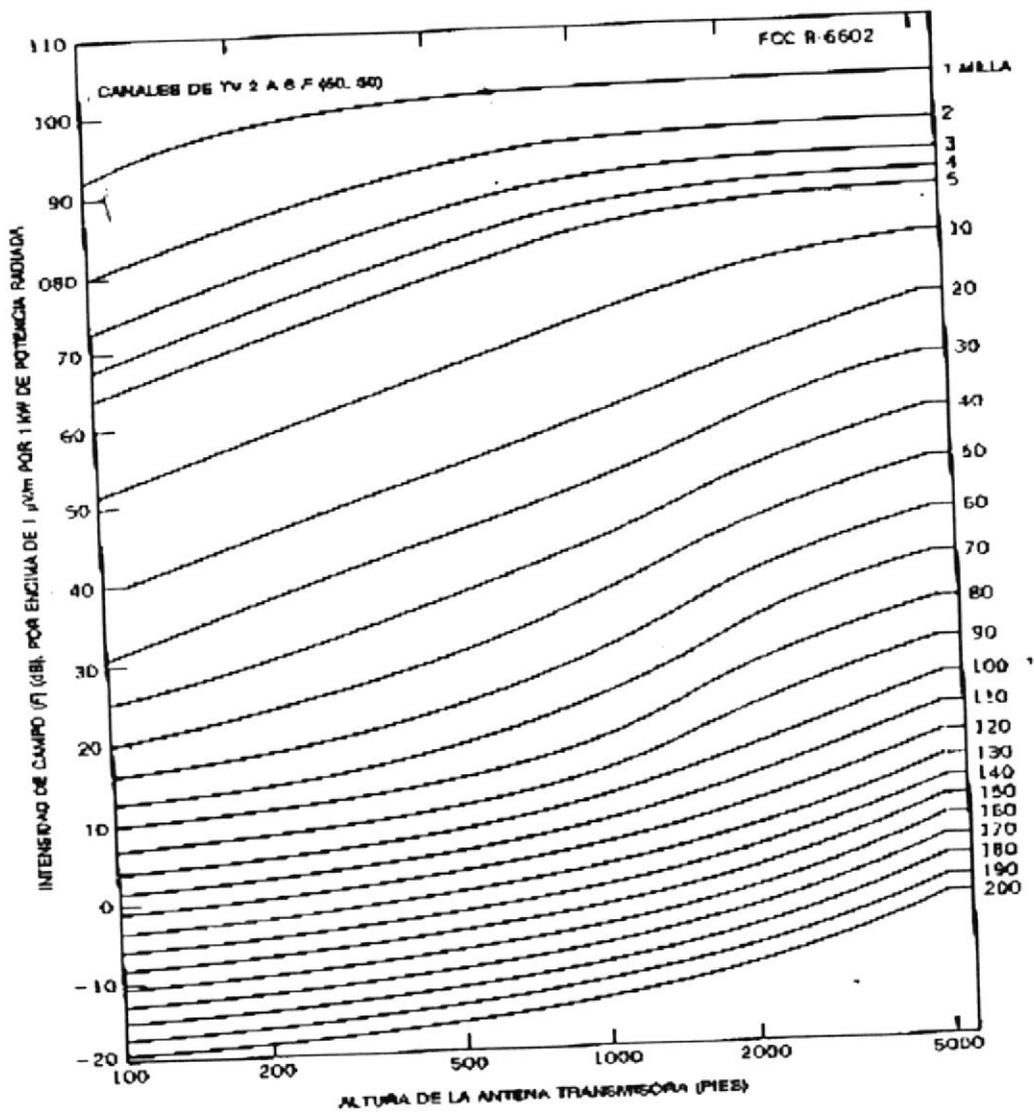


Fig. # 6. muestra una gráfica de la intensidad de campo de los canales 2 a 6.

Las características de propagación de otros canales se proporcionan en la parte 73.699 del documento *Rules and Regulations de la FCC*. La autorización de nuevas estaciones depende de la separación mínima de asignaciones de canal común y varía de 170 a 220 millas para los canales de 2 al 13 y de 155 a 205 millas para los canales del 14 al 83, dependiendo de la localización. Para las asignaciones de canal adyacente, la distancia mínima es 60 millas para los canales del 2 al 13 y 55 millas para los canales del 14 al 88. Este requisito de separación mínima no siempre protege contra la interferencia provocada por la concesión de una nueva estación o el permiso para modificar una ya existente.

2.3. NORMAS DE TRANSMISION

Se requiere que un transmisor tenga un mínimo de 100 W de potencia visual en el plano horizontal con polarización horizontal; la potencia máxima está determinada por los valores que se observan en la tabla V.

La portadora visual suele estar localizada 1.25 MHz por encima del límite inferior del canal.

Canales comunes	Máxima Pot. radiada efectiva visual (dB)por encima de un kW (dBk)
2 a 6	20 dBk (100kW)
7 a 13	25 dBk (316kW)
14 a 83	37 dBk (5000 kW)

Tabla V Potencia máxima en un transmisor.

La subportadora del sonido se encuentra se encuentra 4.5 MHz por encima de la portadora visual, y la subportadora de color está 63/88 partes de 5 MHz (3.57954MHz) por encima de la portadora visual, con una tolerancia de +10Hz y -10Hz y una velocidad de deriva que no debe exceder 0.1 Hz/s. (Ver Fig. # 7).

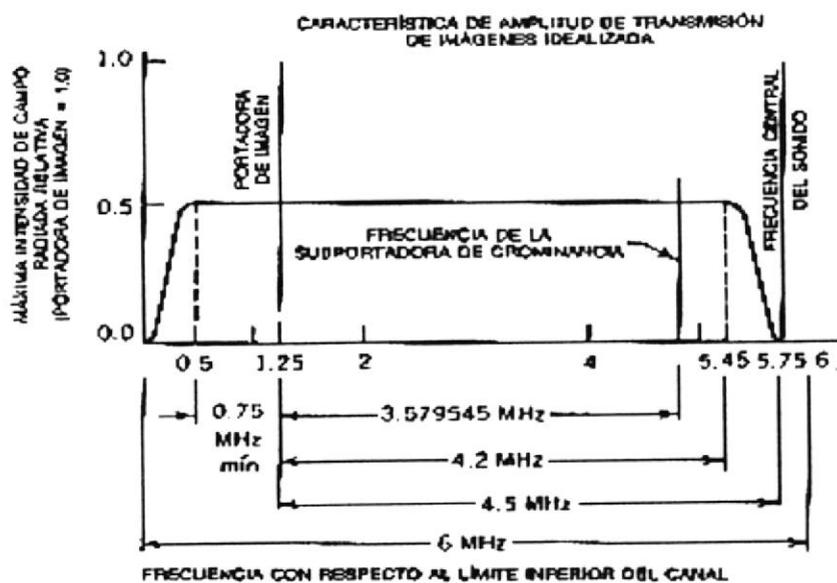


Fig. # 7. Espectro Típico.

El número de líneas de exploración por imagen (cuadro) es 525, entrelazadas dos a uno en campos sucesivos (262.5 líneas por campo) con una frecuencia de exploración horizontal de 15750 Hz para monocromático y 15734.264 ± 0.044 Hz para color. La frecuencia de exploración horizontal para color se obtiene dividiendo la frecuencia de la subportadora de sonido, 4.5 MHz, entre 286. La razón alto/ancho del cuadro es de 4 unidades horizontalmente a 3 unidades verticalmente. Las ondas de forma de sincronización para transmisiones a color y blanco y negro están dadas en las figuras # 8 y figura # 9 respectivamente.

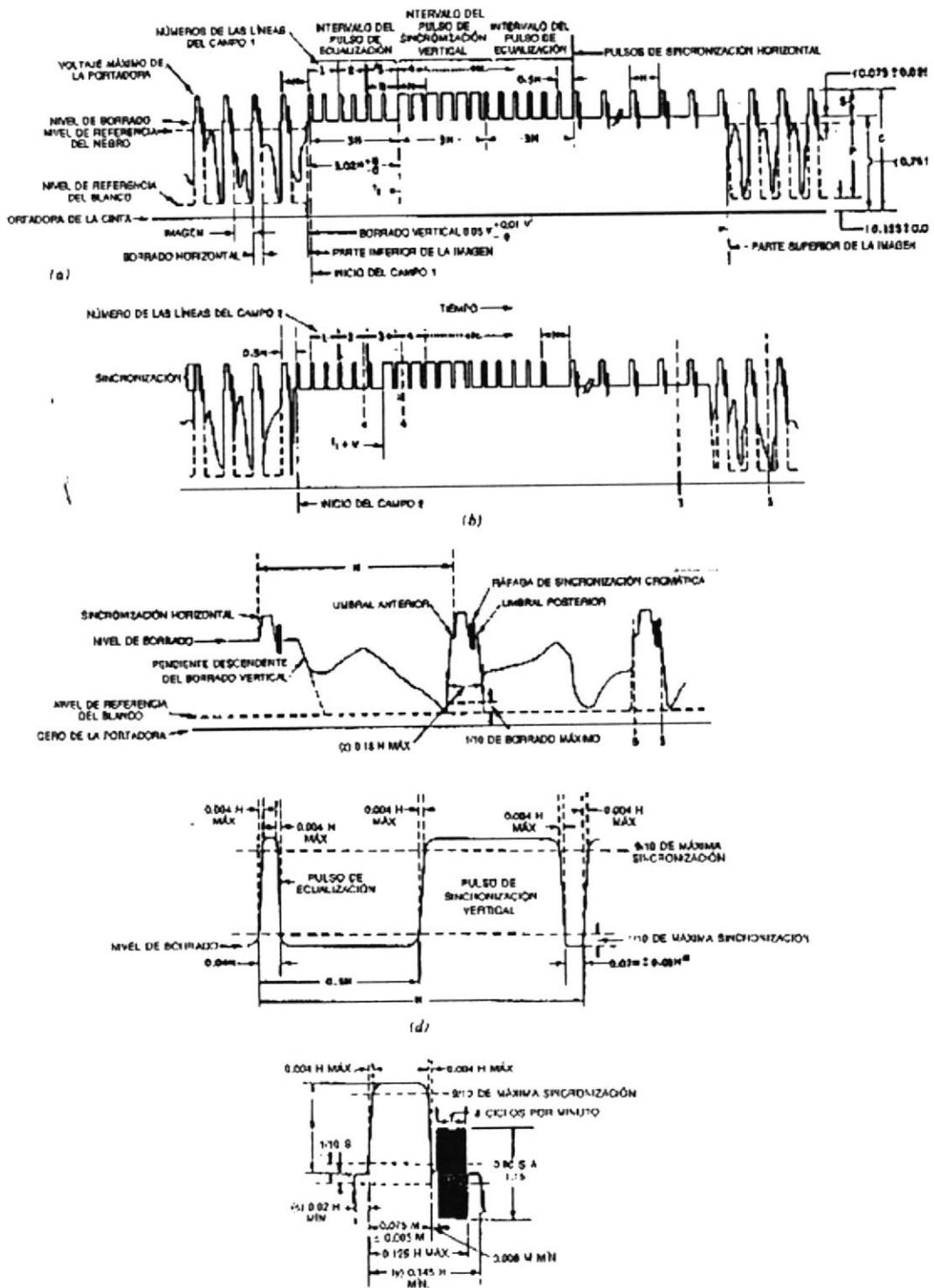


Fig. 8 Sincronización para transmisiones a color

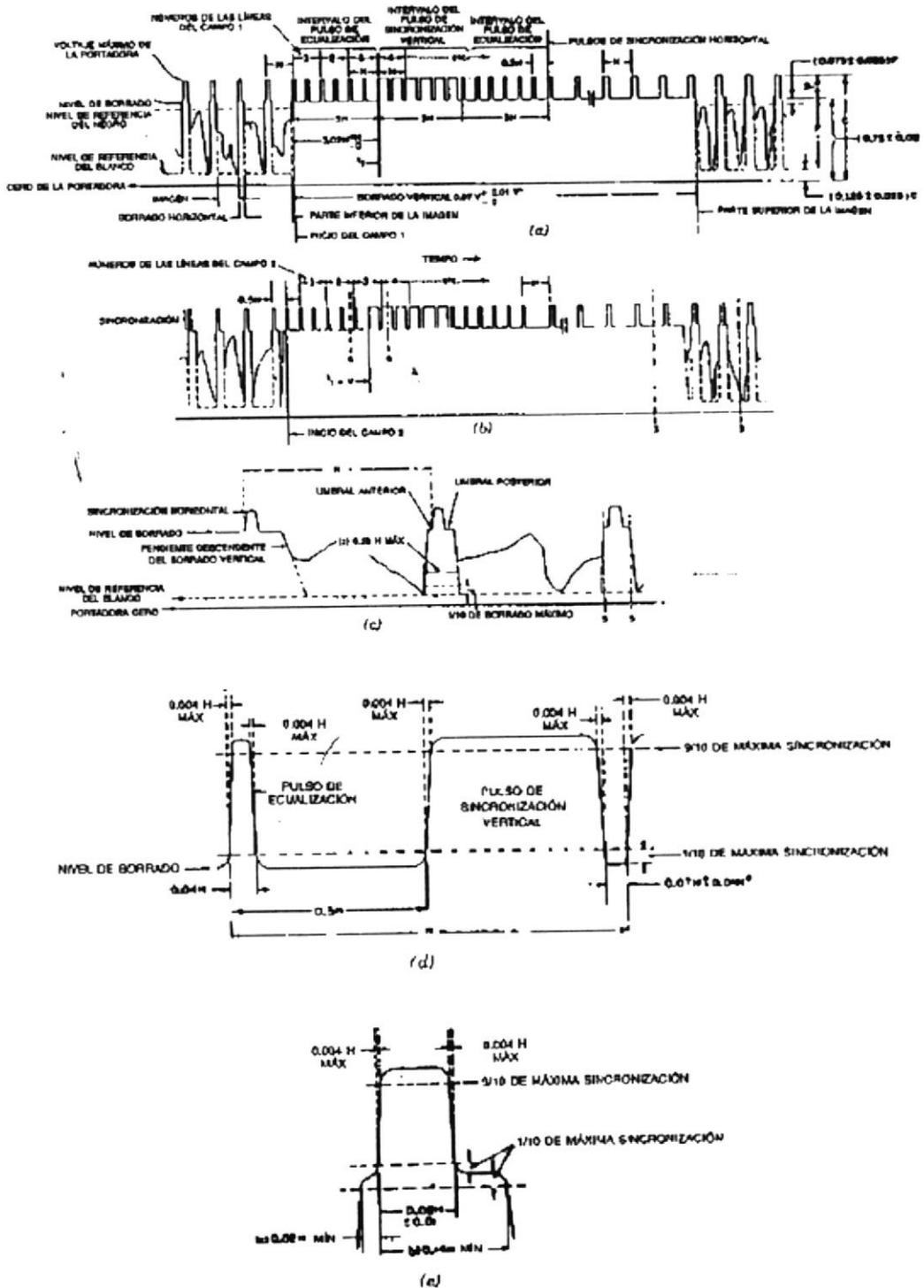


Fig. # 9 Formas de sincronización para transmisiones blanco y negro

La frecuencia de exploración vertical es 60 Hz par monocromático y 59.94 Hz para color. Esa última frecuencia puede obtenerse dividiendo la frecuencia de exploración horizontal de 15734.264 entre el número de líneas por campo, 262.5. El nivel de borrado se transmite a $75 \pm 2.5\%$ del nivel de cresta de la portadora, y el nivel de referencia del negro se encuentra separado del nivel de borrado por $7.5 \pm 2.5\%$ del intervalo de imagen (vídeo) desde el nivel de borrado hasta el nivel de referencia del blanco. Una disminución en la intensidad inicial de la luz provoca un aumento en la potencia radiada. La modulación de la imagen es una forma en la que la banda lateral inferior está significativamente atenuada y se denomina modulación residual. Esto permite que la señal de imagen dentro de aproximadamente la mitad del ancho de banda de RF requerido con la AM ordinaria. La atenuación de la banda lateral inferior se inicia a 0.75 MHz por debajo de la portadora de imagen; por consiguiente, la banda inferior no está atenuada por completo.

El sonido se transmite en una subportadora que se encuentra 4.5 MHz por encima de la portadora de imagen. La modulación que se utiliza es FM con preénfasis a fin de mejorar la razón de señal sobre ruido recibida.

Para la transmisión de color, la información de la imagen se procesa a fin de obtener señales por separado para el contenido de la imagen sin color (luminancia) y para el contenido del color (crominancia). En la figura # 10, se

muestra un diagrama funcional del procesamiento de premodulación estándar de señales de televisión a color.

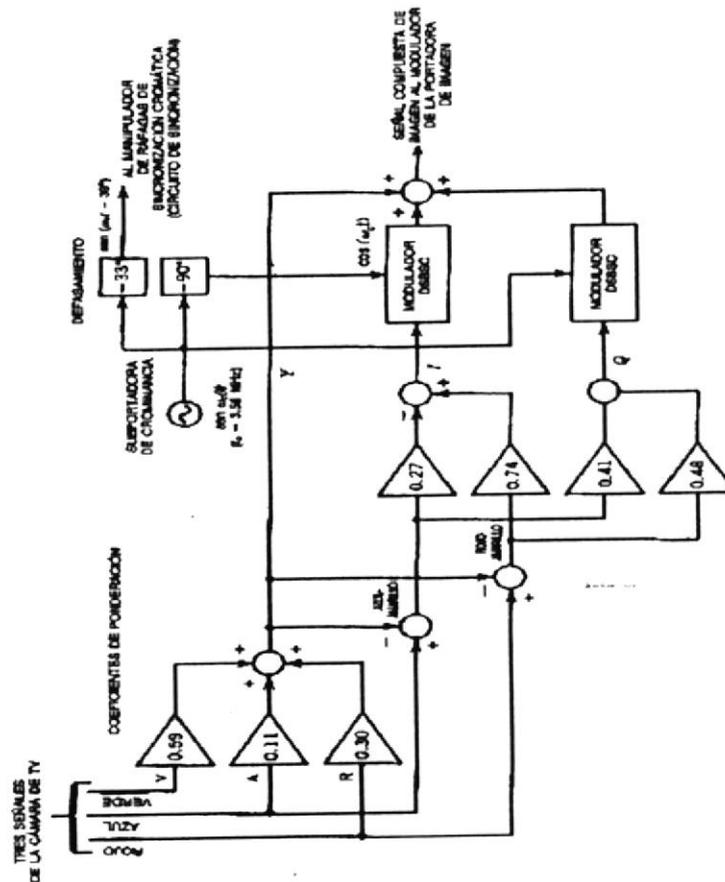


Fig. # 10 Premodulación estándar de señales de televisión a color.

Una cámara de televisión en color produce tres señales simultáneas: una roja, otra azul y otra verde. Estas tres señales se combinan conforme a las normas de la industria ("NTSC Signal Specifications", 1954) a fin de producir una señal de luminancia (designada señal Y) y dos señales de crominancia (I y Q). Las señales I y Q modulan dos componentes de la subportadora de crominancia de 3.58 MHz que se encuentran en cuadratura de fase. Se utiliza la modulación de

doble banda lateral con supresión de la portadora y las salidas de los dos moduladores se suman a la señal de luminancia.

En una señal de imagen compuesta, que modula la portadora de imagen, la señal de luminancia se aplica directamente al modulador. Esta componente de la señal es compatible con los receptores de televisión en blanco y negro. La fase instantánea de la señal de la subportadora de crominancia es función del color de la imagen en ese instante, y la intensidad (saturación) del color está indicada por la amplitud instantánea.

La demodulación de la subportadora de crominancia en el receptor requiere una referencia de fase sincronizada y un detector sincrónico. La información de la referencia de fase se obtiene de la ráfaga de sincronización cromática, que se deriva de la subportadora de crominancia sin modulación y forma parte de la señal de sincronización.

2.4. TRANSMISORES

Un sistema de transmisión por televisión contiene dos transmisores: uno para la portadora de imagen y otro para la portadora de sonido. La amplitud de la portadora de la imagen se modula con una señal de video compuesta (señal de imagen combinada con sincronización y subportadora de color). También se

modula la frecuencia de la portadora del transmisor. Después de la modulación y amplificación, estas dos portadoras se combinan en una señal de televisión.

En la figura # 11 se muestra un diagrama de bloques de un transmisor de televisión.

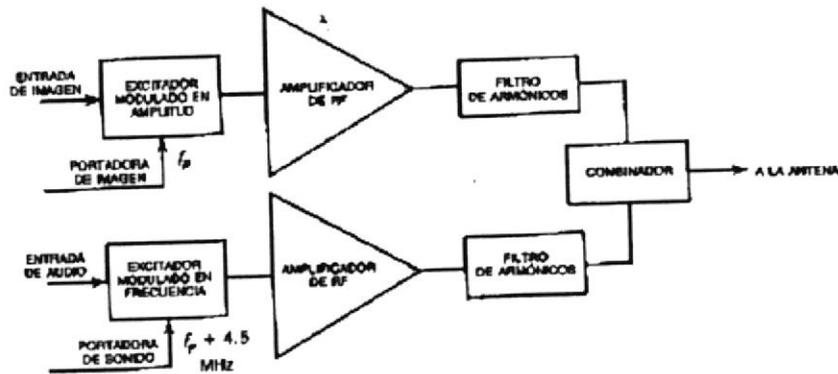


Fig. # 11 Transmisor de televisión.

Los filtros deben insertarse después de los combinadores a fin de prevenir las interferencias a la señal de sonido por la señal de imagen y viceversa. Además, los filtros tienen la capacidad de suprimir los armónicos de las portadoras de radiofrecuencia.

El filtrado de la banda lateral residual puede efectuarse a bajos niveles de potencia si se utiliza un amplificador de potencia lineal para la portadora de imagen. Si se utiliza un amplificador RF clase C para alcanzar mayor eficiencia en el amplificador final de potencia, el filtro de banda lateral residual debe

colocarse después de éste. El último esquema proporciona la máxima eficiencia de potencia en el amplificador final, pero requiere que el filtro de banda lateral residual disipe que el filtro de banda lateral residual disipe mayores cantidades de potencia.

2.5. SISTEMAS DE ANTENAS

En la generalidad de los casos, los sistemas de antena para televisión se diseñan a fin de proporcionar cobertura omnidireccional en el plano horizontal.

Se requiere que la polarización tenga una componente horizontal; sin embargo, el empleo de la polarización circular se está haciendo más popular debido a que es posible obtener mayor cobertura, especialmente por la reducción de desvanecimiento de polarización y el aumento en potencia radiada efectiva. Las ganancias de potencia varían de 6 a 20 dB, dependiendo de la frecuencia y número de elementos. Los elementos pueden estar compuestos de dipolos, ranuras y hélices, así como de otros tipos menos conocidos. Dos factores de suma importancia en el diseño son la altura de la torre (de la antena) y la carga debida al viento.

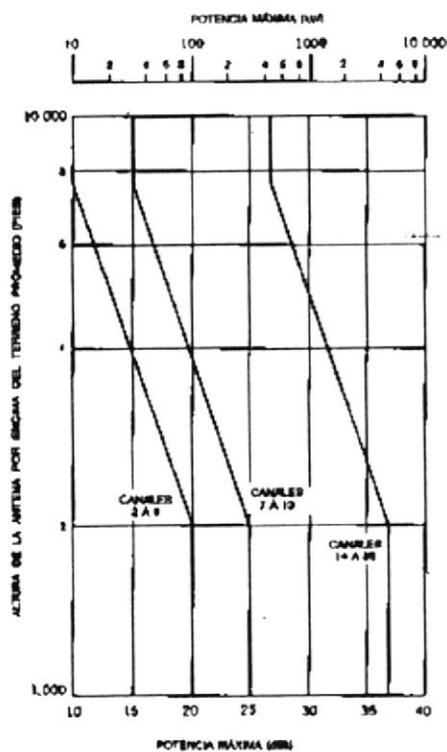
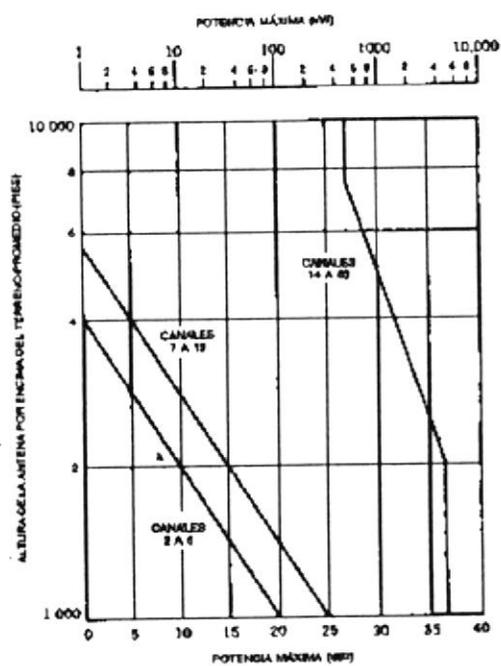


Fig. # 9 Factores en el diseño de una antena transmisora.

2.5.1. CONSIDERACIONES DE LAS ANTENAS DE TRANSMISIÓN

Las antenas son dispositivos que pueden utilizarse, teóricamente, tanto en la recepción como en la transmisión. Para esta utilización, sin embargo es preciso tomar algunas precauciones especiales, más importantes en la transmisión que en la recepción. Veamos algunas de estas precauciones:

1.- *Impedancia característica y acoplamiento con la línea:* La antena y la línea de transmisión deben tener la misma impedancia con el riesgo de que se pueda producir una desadaptación y la aparición de ondas estacionarias. En la recepción, como consecuencia de la desadaptación de impedancias, habrá una pérdida de señal y, por consiguiente, una pérdida en el rendimiento del sistema. En la transmisión esto se agrava: además del menor rendimiento del sistema, es decir, de la menor potencia radiada las ondas estacionarias causan “fantasmas”, y cuando son fuertes pueden dar lugar a la ruptura de la línea, la antena o los componentes de salida del transmisor. Es indispensable, por consiguiente, que la salida del transmisor, la antena y la línea del transmisor tengan la misma impedancia.

2.- *Banda:* La antena debe tener una respuesta plana, es decir, mantener su impedancia dentro de un mínimo de 6 MHz, que es la anchura de banda de transmisión de la señal de video. En otras palabras, por

ejemplo, para transmitir en el canal 2, la antena debe tener una impedancia constante desde 54 MHz hasta 60 MHz.

3.-*Directividad lóbulo de radiación:* Las antenas, en general, son directivas, radiando con mayor eficiencia dentro de un ángulo determinado. Cuando nuestro problema es retransmitir una señal, lo ideal es utilizar una sola antena. Es el caso, por ejemplo, de una estación repetidora de Friburgo para la señal que llega de Río de Janeiro. Esta señal viene de la parte de atrás, debiendo efectuarse su repetición hacia delante. Cuando, por el otro lado, es necesario radiar la señal en todas direcciones, deben emplearse dos dipolos perpendiculares, alimentados con una diferencia de 90 grados, lo que proporciona una radiación circular. Otra solución consiste en utilizar cuatro antenas direccionales, es decir, cada una de ellas con un reflector que concentre la energía radiada en una sola banda, dentro de un ángulo de por ejemplo 100 grados. Este reflector también concentra la energía que sería radiada verticalmente y perdida en el espacio, dirigiéndola en una dirección útil. La combinación de las 4 antenas, una de ellas vuelta hacia cada uno de los 4 lados, y alimentadas con un desfase de 90 grados, también proporciona una radiación circular en el plano horizontal y con más ganancia que en el caso de los dos dipolos.

A veces hay interés en transmitir sólo en una determinada dirección, donde hay grandes concentraciones de población, dejando de radiar energía en otras direcciones de menor interés, como las sierras elevadas o el mar.

2.5.2. TIPO DE ANTENAS DE TRANSMISION

La antena "yagi".- La antena yagui de 3 elementos próximos, o de 5 elementos separados, sin compensación, tiene una impedancia de cerca de 50 ohmios, siendo su banda suficientemente ancha para retransmitir la señal. Tiene las siguientes características:

- 1ª *Frecuencia y anchura de banda.*- La frecuencia se determina por la longitud del dipolo, y la anchura de banda por su diámetro.
- 2ª *Impedancia característica.*- El dipolo simple tiene una impedancia característica de 75 ohmios al aire libre. El reflector dispuesto en la parte posterior de la antena tiene la propiedad de rebajar esta impedancia.

3ª *Directividad*: Se trata de una antena directiva horizontalmente y bastante directiva verticalmente. Su radiación máxima se produce, evidentemente, en el sentido del eje principal.

4ª *Ganancia*: En relación con un dipolo simple, la antena de reflector angular presenta una ganancia de 10 dB en el sentido de su eje principal.

Otras antenas de transmisión.- En las estaciones emisoras de televisión se emplean otras antenas, además de las descritas, como el “panel de dipolos”, el “supermolinete”, etc., que proporcionan una elevada ganancia y permiten distribuir la señal de manera que la energía sea radiada en la dirección deseada, es decir, hacia donde haya mayor concentración de población eliminando el mar y las montañas elevadas.

La fabricación, instalación, y montaje de estas antenas son, sin embargo, bastante complejos, hasta el punto de que sólo pueden ser realizados por personal altamente especializado.

CAPITULO III

VISION GLOBAL DEL PROYECTO

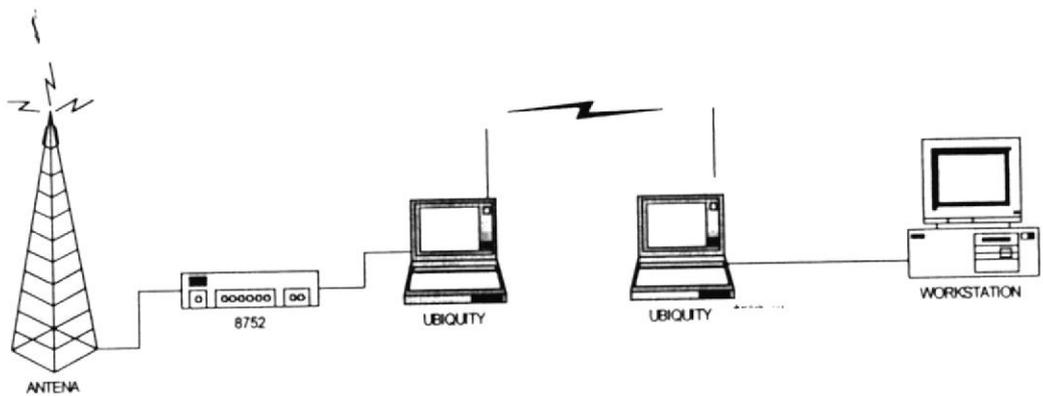


Fig. # 13 Visión global del proyecto.

3.1. RED DE TELEVISION (TC)

La red de Televisión de TC, está compuesta de diferentes antenas alrededor de todo el país. Hay antenas emisoras generalmente ubicadas en los lugares más poblados y antenas repetidoras de la señal en lugares un poco más remotos. Pero toda la señal es generada en la ciudad de Guayaquil, es por eso que si la señal no

llega a otros lugares, la única forma de saberlo es por algún televidente que notifica el hecho.

Las causas de que la señal se pierda, son muchas. Como por ejemplo, la ausencia de alimentación externa de los equipos, ya sea porque las líneas de alimentación externa no funcionan, o porque la alimentación auxiliar de baterías o generadores no se accionó por algún motivo; la caída de la señal de audio o de video, o también de la señal de RF en las antenas repetidoras, etc.

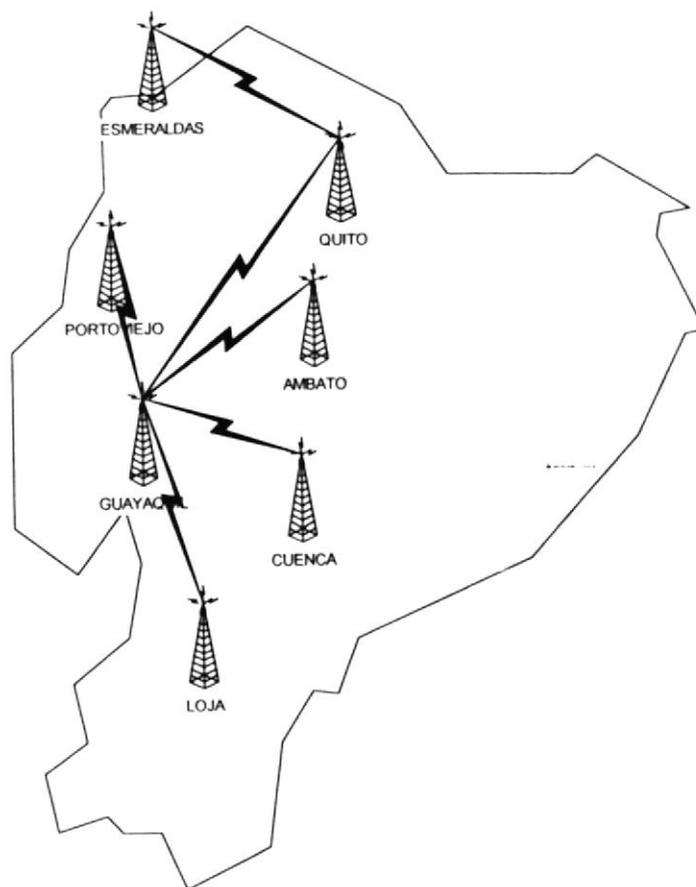


Fig. # 14 Esquema de la red de transmisión de Telecentro.

El proyecto está orientado a comprobar la presencia de estos factores de alimentación, video y audio en una antena, por medio de interfaces que permiten generar un 1 lógico para la presencia, o un 0 para indicar la ausencia de las diferentes señales. Estas señales digitales generadas, deben llegar a una central de control para comprobar el estado de las mismas.

Es así como se veía la necesidad de tener un procesador en cada antena que se iba a sensar, para que realice el manejo y envío de la información.

Es por eso que se escogió el 8752 el mismo que puede ser programado pues posee un CPU interno de 8 bits, 256 bits de memoria RAM, memoria EPROM, comunicación asíncrona full duplex. Suficientes propiedades para reemplazar el procesador. Luego, el medio de comunicación, tiene que ser inalámbrico indudablemente.

3.2. EL MEDIO DE COMUNICACIÓN

Al principio se pensó utilizar los módems internos del computador, utilizando la red telefónica de EMETEL, y utilizar un módem celular en la antena, lo que significaba reducir los costos del mismo, pero la interface entre la telefónica celular y la normal, no nos garantiza la transmisión de datos, ni la velocidad necesaria para recibir los datos.

Luego se pensó en utilizar módems celulares, tanto en la antena como en la central, pero a través de experiencias de diferentes profesionales que habían realizado pruebas de monitoreo utilizando este medio, se seguía con el problema de la frecuencia de transmisión y la no garantía de recibir los datos en la central.

Es por eso que escogimos el servicio de CDPD, que a través de los canales celulares de BELLSOUTH presta la compañía BISMARCK, alrededor del país. En dicha compañía encontramos el UBIQUITY, instrumento que puede ser utilizado de diferentes maneras tanto para transmitir voz y datos, y a diferentes velocidades de transmisión.

La cobertura de CDPD que BISMARCK ofrece lo podemos observar en el siguiente mapa, que en contraste con la cobertura de BELLSOUTH para telefonía celular, es mucho menor ya que tienen diferentes características de transmisión.

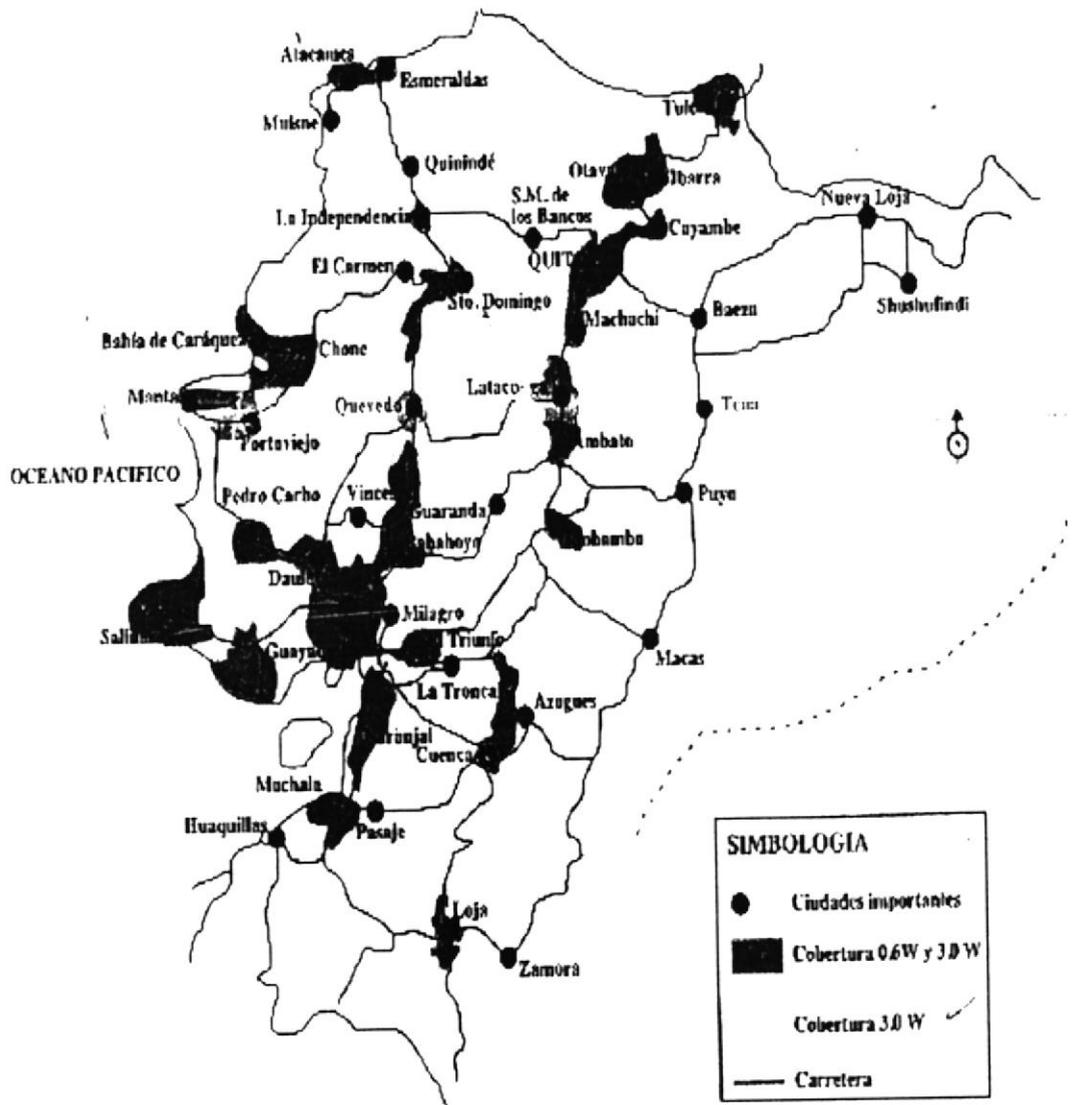


Fig. # 15 Cobertura de BELLSOUTH

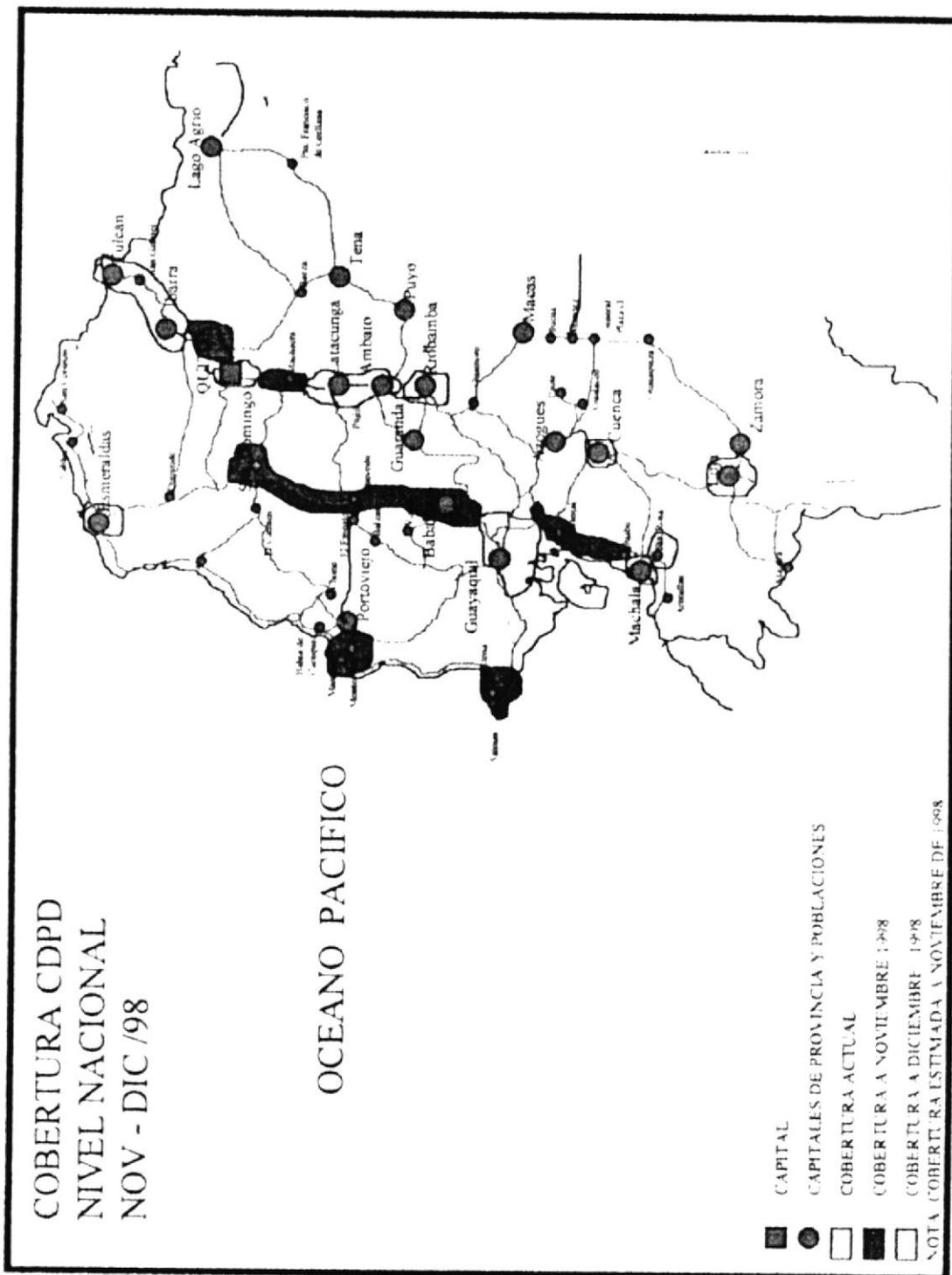


Fig. # 16 Cobertura de BISMARK

3.3. EL UBIQUITY

3.3.1. COMUNICACIÓN ENTRE SU COMPUTADOR Y EL SOFTWARE DE CONTROL

Operación en TCP/IP.- En el modo nativo de TCP/IP, el sistema de comunicaciones es controlado directamente por el software TCP/IP. Esto puede ser realizado a través de:

- Un SLIP driver, provisto con algunas aplicaciones de TCP/IP.

- Una interface WINSOCKET.

Comandos AT.- Para todos los modos de operación, la interface entre el computador y el sistema de comunicaciones son los comandos AT. Los comandos AT pueden ser entrados manualmente a través del teclado utilizando una aplicación de comunicación con un modo de emulación de terminal.

Los comandos AT son también generados por la aplicación de comunicaciones y utilizados para el control de la unidad Ubiquity:

- Cuando las aplicaciones de comunicación son usadas como módem celular, módem alámbrico, como teléfono celular, o teléfono alámbrico.
- Cuando el sistema de comunicaciones es operado como CDPD.
- Cuando las utilidades MCM son utilizados.

Registros-S.- Un control adicional del sistema de comunicaciones es realizado a través de dos sets de Registros S. Estos registros almacenan los parámetros de operación del sistema, y pueden ser configurados por el usuario o dejados en la configuración que viene de fábrica.

3.3.2. CODIGO DE TRANSMISIÓN

El sistema de comunicación está basado en un software de control que puede ser actualizado desde su computadora. Cuando el software de control es transmitido, el módulo de comunicación lo almacena. Cada vez que enciende su computadora, este chequea el software de control del Ubiquity. Si el chequeo indica un error, el computadora automáticamente transmite el código otra vez.

Luego que el software ha sido instalado. Cada vez que encienda el computadora, el sistema de comunicación es inicializado y el software instalado el chequeado. Si el software del Ubiquity no es encontrado o detecta un error, el computadora automáticamente ejecuta la transmisión del software.

Para transmitir manualmente el software al sistema de comunicaciones, se utiliza el siguiente procedimiento:

1. *Típee:* CD C: / UBIQUITY si su clave se encuentra instalado en su directorio por default. Si la clave reside en otro directorio típee el comando CD correspondiente para accederlo.
2. *Típee:* DOWNLOAD para correr el programa que transmite el código de su computadora a la unidad Ubiquity. Cuando utilice DOWNLOAD sin parámetros, el programa corre utilizando los parámetros seteados.

Download (-f) (-pn) (-br) *filename*

Parámetros:

-f Especifica si la transmisión debe ocurrir si el código es válido. El parámetro por default no debe forzar la transmisión.

-pn Especifica el puerto COM asignado para el sistema de comunicación. El valor de n debe ser 1 o 2. El valor por default es 2 par computadoras IBM, y 1 para las demás.

-br Especifica la velocidad de transmisión. El valor de r puede ser 9600, 38400, 57600, o 115200. Por default es 57600.

Filename: Especifica el archivo que contiene el código para transmitir.

2. Durante la transmisión el computadora presenta el mensaje:

```
Download File – Download CP Code Ver ^X.XX From File C:  
/UBIQUITY/CP.DNL.
```

```
Progress – XX percent complete
```

Seguro que el siguiente mensaje aparece, indica que el código ha sido cargado satisfactoriamente en el sistema de comunicaciones:

```
Action – PCSI Ubiquity is Ready
```

Result – EE Code Ver. X.XX, CP Code Ver. X.XX

3.3.3. INTERFACE CON EL USUARIO, UTILIZACION DE LOS COMANDOS AT CON EL SISTEMA DE COMUNICACIONES

Los comandos AT son ingresados directamente de las aplicaciones de comunicación que provee el modo de emulación del terminal. Los comandos estándar AT permiten la comunicación asíncrona de módems asíncronos y el sistema de comunicación.

Luego de ingresar los comandos AT, seguido de un Carriage Return, no ingrese ningún otro comando hasta que el sistema de comunicación responda con el Carriage Return y alimente el comando inicial. Si usted ingresa otro comando antes que el sistema de comunicaciones complete este proceso, los datos comunicados entre el computadora y el sistema de comunicaciones se pueden perder. Comandos extendidos proveen capacidades de control adicionales más allá de aquellos provistos por los comandos AT estándar. Lo común de los comandos AT es que comienzan con “AT”, seguidos de un verbo que define la operación, y un Carriage Return carácter o Enter:

Por ejemplo:

AT&V Enter

AT precede todos los verbos de comando excepto en el caso de los siguientes:

+++AT Acciona el método TIES (Time Independent Scape Sequence), que es una secuencia de escape de cuatro partes que consisten en tres caracteres de escape (típicamente +++), seguido de caracteres AT, luego de una validación del verbo AT, y seteo, seguido de un carácter de terminación.

A/ Este comando es utilizado para reingresar el comando AT previo. Este comando puede ser utilizado para repetir el llamado si el sistema está ocupado, o si no hubo respuesta.

GRUPOS DE COMANDOS AT.- Los comandos AT, retornan solo un mensaje de OK o ERROR, con excepción de los comandos extendidos AT!, que pueden tener hasta 40 caracteres de longitud, seguidos de un Carriage Return o Enter. Cuando un comando AT es agrupado, "AT" debe ser ingresado al principio de la línea. Los comandos AT pueden estar

separados por espacios, los cuales van a ser contados como parte de los 40 caracteres de limitación.

Los comandos AT!, y los comandos que retornan un mensaje que no sea OK o ERROR deben ser ingresados individualmente.

3.3.4. CAMBIANDO LOS MODOS DE OPERACIÓN CON LOS COMANDOS AT

Cuando los comandos AT son utilizados para swichear el sistema de comunicación entre los modos de operación, es importante recordar:

- Solo un modo de operación puede estar activo al mismo tiempo.
- El sistema de comunicaciones debe siempre ser regresado al modo Ideal antes de ser utilizado de otra manera. Esto significa que si el sistema de comunicaciones está en modo de teléfono celular, por ejemplo, y lo quiere utilizar luego como módem celular, usted debe realizar lo siguiente:

1. AT!IDLE (que regresa al sistema de comunicaciones al modo ideal)
2. AT!ACM (que setea al sistema de comunicaciones como módem celular)

3.4. MODOS DEL SISTEMA DE COMUNICACIONES.

MODO	METODO DE COMUNICACION	DESCRIPCION	NOTA
Ideal	No aplicable	El sistema de comunicaciones no está listo para recibir o enviar comunicación	Por default, el sistema de comunicaciones ingresa al modo ideal cuando se inicia, y se mantiene en ese modo hasta que el comando respectivo sea ingresado.
Cellular Packet (CDPD)	Paquetes de datos en canales celulares	El sistema de comunicaciones utiliza el CDPD para comunicar datos sobre una línea celular	El sistema de comunicaciones va a utilizar aplicaciones CDPD vía Winsock, SLIP o Emulación de Módem. La portadora celular debe soportar CDPD.
Cellular Módem (ACM)	Datos en un canal celular	El sistema de comunicaciones utiliza un canal telefónico celular para comunicar datos o fax.	Requiere una línea celular con portadora celular. Utiliza AT&F1 para seteos de fábrica.
Cellular Voice (ACV)	Comunicación de voz en un canal celular	El sistema de comunicaciones utiliza un canal celular para transmitir voz	Un teléfono es requerido para la comunicación en este modo.
Wired Módem (PSTN)	Datos a través de PSTN.	El sistema de comunicaciones opera como módem, comunicando datos o fax a través de PSTN.	El sistema de comunicaciones debe ser conectado a través de un arreglo de un acceso de datos (DAA) al PSTN.
Wired Phone (PHONE)	Comunicación de voz a través del PSTN	El sistema de comunicaciones utiliza el PSTN para comunicar voz.	El sistema de comunicaciones debe ser conectado a través del DAA al PSTN. Un teléfono es requerido para la comunicación en este modo.

Tabla VI Modos del sistema de comunicaciones en el UBIQUITY.

3.5. COMANDOS PARA LOS MODOS DE OPERACION:

COMANDO	DESCRIPCION	RESPUESTA
AT!ACM	Módem Celular	Numérico:0 o 4 Alfabético: OK o ERROR
AT!ACV	Teléfono celular	Numérico: 0 Alfabético: OK o ERROR
AT!CDPD	Celular Packet	Numérico: 0 o 4 Alfabético: OK o ERROR
AT!DIAGn	Diagnóstico	Resultado del diagnóstico de comandos
AT!IDLE	Modo ideal	Numérico: 0 o 4 Alfabético: OK o ERROR
AT!PHONE	Teléfono alámbrico	Numérico: 0 o 4 Alfabético: OK o ERROR
AT!PSTN	Módem alámbrico	Numérico: 0 o 4 Alfabético: OK o ERROR
AT!QUERY	Modo de operación	Numérico: 0,1,2,3,4,5 Alfabético:IDLE, PSTN, ACV, ACM, CDPD, o PHONE.
AT!RESET	Resetea el sistema de comunicación	Numérico:Ninguno Alfabético:Ninguno

Tabla VII Comandos para los modos de operación.

CAPITULO IV

CIRCUITOS SENSORES

En una Antena de Emisiones Televisivas, las señales de AUDIO y VIDEO, son tomadas por separado y luego moduladas para su transmisión. Por lo que al sensor niveles de audio y vídeo es mucho más sencillo que en una Antena Repetidora.

En una Antena Repetidora la señal llega por medio de una transmisión de RF de 1.7 a 2 Ghz, por lo que se hace necesaria la utilización de un demodulador de la señal, para así poder separar el audio del vídeo y poder sensorlos.

Además de las señales de audio y vídeo, se puede sensor la presencia o no de las alimentaciones de los equipos de la Antena, tanto de la línea externa como de las alimentaciones de emergencia que por lo general están compuestas por baterías, en lugares lejanos, o por generadores de emergencia.

4.1. CIRCUITO DETECTOR DE VIDEO:

El siguiente circuito genera una salida de 5 voltios, a la salida, con la presencia de una señal de video en la línea de transmisión. A través de un filtro supresor de 3.58 Mhz, para suprimir las componentes de las subportadoras de color. Se obtiene la señal de video pura en la línea, la misma que es amplificada hasta que los operacionales se saturan y podamos obtener un alto a la salida. También se le ha agregado una constante RC para cuando esta señal se pierda por más de un segundo sea detectada. La no presencia de señal de video hace que esta señal de sincronismo sea nula, con lo que, a la salida del circuito vamos a tener un cero lógico. (ver anexo # 1). El circuito está compuesto por:

- 1 capacitor de 47 uF.
- 1 capacitor de 47 pF.
- 1 inductor de 47 uH
- 2 diodos 1N914.
- 1 resistencia de 470 OHMS.
- 1 resistencia de 1 K.
- 1 resistencia de 100 K.
- 5 resistencia de 1 K.
- 1 capacitor de 0.47 uF.
- 2 opams 741.

4.2. CIRCUITO DETECTOR DE AUDIO

59

La señal de video que ingresa, de alrededor de 2 a 3 KHz, es amplificada y luego por medio de una capacitancia de 0.47 H se logra que el circuito detecte la ausencia de la señal de video luego de un tiempo moderado, y también una pequeña señal DC generada luego de la amplificación. Luego la misma señal es vuelta a amplificar, para lograr la saturación del opam y generar así un alto a la salida. Al momento de no detectar señal de audio se genera un bajo a la salida. (ver anexo #2).

El circuito está compuesto por:

- 1 capacitor de 47 uF.
- 1 capacitor de 47 pF.
- 1 inductor de 47 uH
- 2 diodos 1N914.
- 1 resistencia de 470 OHMS.
- 1 resistencia de 1 K.
- 1 resistencia de 100 K.
- 5 resistencia de 1 K.
- 1 capacitor de 0.47 uF.
- 2 opams 741.

4.3. CIRCUITO DETECTOR DE ALIMENTACION DE C.A.

Consiste en una fuente regulada de 5v, 1 A, la que nos permite saber si la alimentación externa se encuentra activa o no, si no tendremos un bajo a la salida. Si tenemos alimentación trifasica podemos realizar un circuito de estos para cada fase, ya que la interface utilizada para enviar los datos tiene la capacidad de estar conectada a 8 datos digitales. Con lo que en un caso de que se vaya alguna de las fases, pueda ser detectada su ausencia. (ver anexo #3).

4.4. INTERFACE DE CONECCION ENTRE LOS SENSORES Y EL MODEM (CDPD)

4.4.1. INTERFACE CON MICROCONTROLADOR 87C52

Se lo empleo por ser uno de los modelos de microcontroladores mas populares de Intel y por tener las siguientes características generales:

- CPU de 8 bits.
- Procesador booleano (operación sobre bits).
- 4 puertos de 8 bits.
- Para el 8752, 256 bytes de memoria interna RAM útil para el usuario y 384 bytes en total considerando el área de los registros especiales

(SFR). Para el 8751, 128 bytes útiles para el usuario y 256 bytes incluyendo el SFR.

- 8 octetos de ROM (8052) o EPROM (8752), 4K para el 8x51.
- Espacio de memoria de 64K para programa externo.
- Espacio de memoria de 64K para datos externos.
- El 8752 contiene 3 contadores-temporizados (timers). El 8751 presenta dos timers.
- Comunicación asincronicas full-duplex.
- 6 fuentes de interrupciones con niveles de prioridad (5 para el 8751):
 - 2 interrupciones externas
 - 3 interrupciones de los timers (2 para el 8751)
 - 1 interrupción de la comunicación
- Oscilador interno.
- Polarización de 5 voltios.

4.4.2. PATILLAJE DE LOS MICROCONTROLADORES 8752.

En el siguiente gráfico se puede identificar los pines del Microcontrolador 8751, según los formatos de presentación pin DIP (Dual in Line Package):

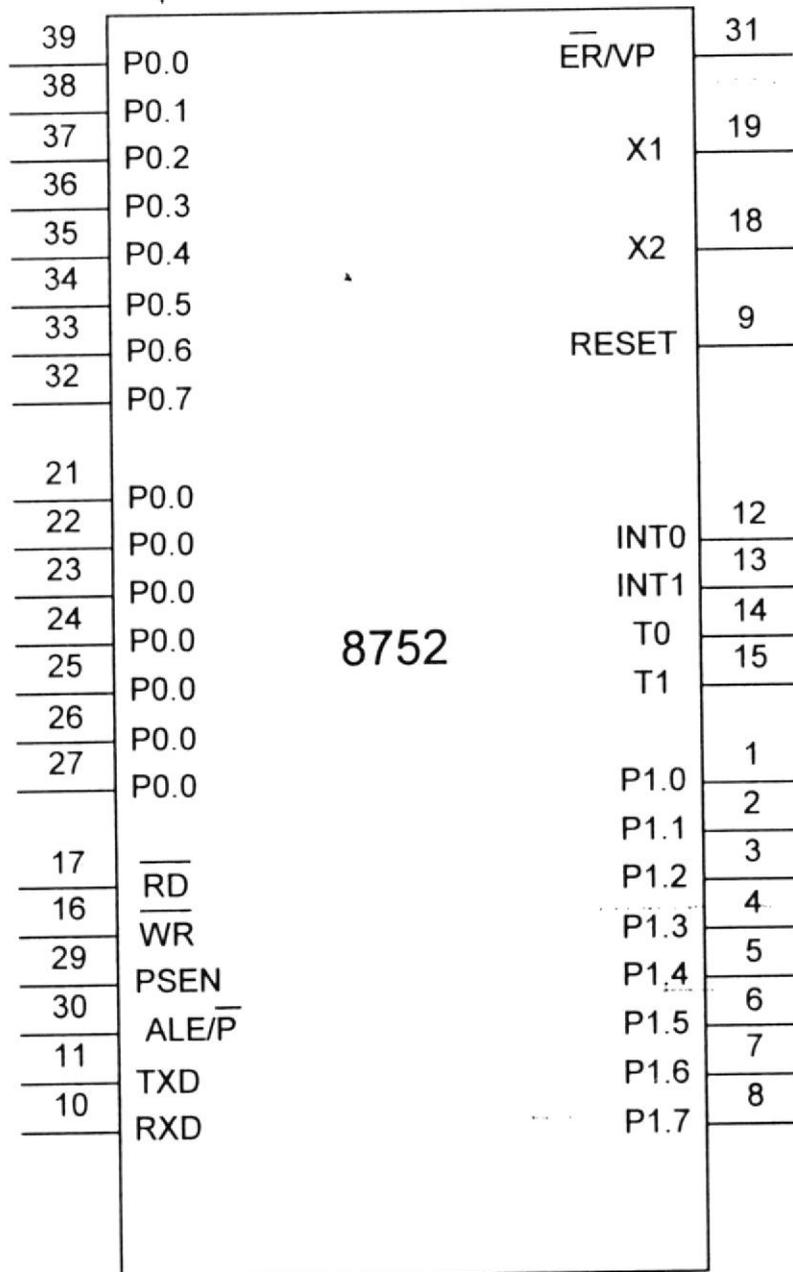


Fig. # 17 Pastillaje Microcontrolador 87C52

- Vcc: Alimentación positiva de +5 voltios.
- Vss: Conexión a tierra (0 voltios).
- Puertos: posee 4 puertos de 8 bits bidireccionales (P0, P1, P2, P3), en otras palabras se pueden programar como entradas o salidas.
- En el puerto 1 (P1):
 - P1.0 T2 (Timer /contador 2. Entrada externa)
 - P1.1 T2EX (Timer/contador 2. Captura e impulso de carga)
- En el puerto 2 (P2); se recibe la parte alta de la dirección, durante la programación y verificación de la memoria EPROM (ver Fig.)
- En el puerto 3 (P3); tiene ciertas funciones especiales:
 - P3.0 RXD (Entrada puerto serie)
 - P3.1 TXD (Salida puerto de serie)
 - P3.2 INT0 (Interrupción 0. Externa)
 - P3.3 INT1 (Interrupción 1. Externa)
 - P3.4 T0 (Entrada externa. Timer 0)
 - P3.5 T1 (Entrada externa. Timer 1)
 - P3.6 WR (Autorización escritura en memoria de datos externa)
 - P3.7 RD (Autorización lectura en memoria de datos externa).

4.4.3. TIPO COMUNICACIÓN

Otra de las facultades es su ventaja en el modo de transmisión. El puerto serie de los Microcontroladores 8752 trabaja en el modo full-duplex, lo que significa que puede recibir o transmitir simultáneamente. Como receptor tiene un buffer que le permite recibir un segundo byte, antes de que el byte previamente recibido haya sido leído por el registro receptor. (Sin embargo, si el primer byte no ha sido leído en el tiempo en que el segundo byte se completa, uno de los dos bytes se pierde.)

El puerto serie puede operar en cuatro modos seleccionados utilizando los bit 6 = SM1 y bit 7 = SM0 del registro SCON , para nuestro caso y con el objetivo de lograr una mejor transmisión empleamos el MODO 1 :

- Modo1: Es una comunicación más acorde con los estándares habituales. Utiliza 10 bits que son transmitidos a través de TXD o recibidos a través de RXD. Los bits utilizados tienen la siguiente función:

- 1 bit de inicio (0) (señala el comienzo)
- 8 bits de datos (LSB = primero)
- 1 bit de parada (1)

La velocidad de transmisión puede ser ajustada por el usuario dentro de un amplio rango y que para nuestro caso lo probamos a 2400 bps, 9600bps y 19200bps en una comunicación directa del microprocesador con el computadora y fue exitosa; para la prueba con los módem UBIQUITI y que realizamos con la empresa Bismark lo hicimos a 9600 bps obteniendo buenos resultados

Sin descartar los otros modos de transmisión que también se deben tomar en cuenta para otra forma de aplicación y que describimos a continuación:

- Modo 0: Los datos en <<serie>> entran y salen a través de la línea RXD. TXD presenta los impulsos de desplazamiento del reloj. La palabra de información a transmitir o recibir es de 8 bits, siendo el primero en salir o en llegar el <<bit menos significativo (LSB = b0)>>. La velocidad de transmisión expresada en baudios (baudio = bits/seg.) está dada por la relación 1/12 de la frecuencia de reloj del microcontrolador.
- Modo 2: Transmite a través de TXD y recibe por RXD como en el caso anterior. Utiliza 11 bits distribuidos de la siguiente forma:
 - 1 bit de inicio (0)
 - 8 bits de datos (LSB = primero)

- 1 bit de dato (9° bit) (programable)
- 1 bit de parada (1)

La velocidad de transmisión es programable bien a 1/32 o 1/64 de la frecuencia del oscilador.

- Modo 3: Como en el caso anterior, 11 bits son transmitidos a través de TXD o recibidos por RXD, de la siguiente manera:

- 1 bit de inicio (0)
- 8 bits de datos (LSB = primero)
- 1 bit de dato (9° bit)
- 1 bit de parada (1)

4.4.4. INTEGRADO MAX 232

Este integrado, mostrado en la figura, se lo utilizó para poder conjuntamente con el 87c52 crear la interface a la que se conectan los sensores y a su vez se conecte al Módem de transmisión (CDPD) , evitando de esta manera la utilización de un computadora por cada estación repetidora ahorrando de esta manera costos de inversión.

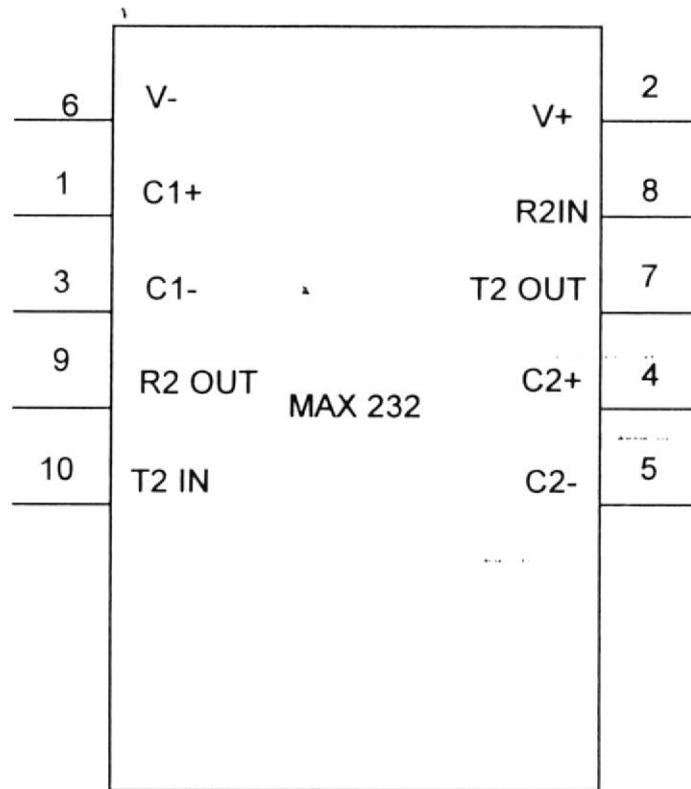


Fig. # 17 Pastillaje del Integrado MAX 232

El MAX 232 es un integrado que permite elevar los niveles de voltaje de TTL a niveles de voltajes RS 232 (conector universal para módems), es decir a +12 y -12 voltios.

Otra característica para su empleo es su polarización a +5 voltios, similar al 87c52, resultando de esta manera una facilidad en su ensamblaje.

Además su capacidad para transmitir y recibir señales.

4.4.5. COMPOSICION DE LA INTERFAZ DE CONEXION

Para la interfaz de conexión se siguió el siguiente esquema:

Donde utilizamos los siguientes elementos:

- 1 microcontrolador 87c52
- 1 integrado MAX 232
- 2 capacitores de 30 picofaradios
- 5 capacitores de tantalio de 10 microfaradios
- 1 capacitor de 1 microfaradio
- 1 resistencia de 1.5 kilohmios
- 1 resistencia de 10 kilohmios
- 1 cristal de 18 Megahertz
- 1 fuente de 5 voltios
- 1 botonera

4.5. DESCRIPCION DE LAS PRUEBAS

4.5.1. PRIMERA PRUEBA

Para poder trabajar con la interface se procedió de la siguiente manera:

- 1.- Armamos la interface como se muestra en la figura anterior.
- 2.- Procedimos a calcular los valores para los cuales se deseaba transmitir.
- 3.- Creamos el programa para la EPROM del microcontrolador, con las velocidades que se deseaba probar (2400, 9600 y 19200 bps) y con la información que se deseaba transmitir, para nuestra primera prueba lo hicimos para que enviara números consecutivos ; este programa se lo ejecuta en lenguaje ensamblador de donde se obtiene la información para grabar la EPROM. (Ver sección 5.2.2)
- 4.- Se conecto la interface directamente a la computadora, desde el MAX 232 a la computadora mediante 3 cables (Transmisión, Recepción y Tierra) y un conector DB25 hembra para realizar una transmisión serial.

- 5.- Creamos un programa en lenguaje Visual Basic versión 4.0 para crear el enlace servidor – cliente, el cual recibiría la información que le envíe el microcontrolador por medio de pulsos de una botonera, mostrándose en la pantalla de la computadora número consecutivos, es decir al primer pulso aparecía el uno, al pulso siguiente el dos y así sucesivamente. (Ver sección 5.2.1).

Esta prueba se la realizó con las tres velocidades de transmisión y fue un éxito.

4.5.2. SEGUNDA PRUEBA

Para poder trabajar con la interface se procedió de la siguiente manera:

- 1.- Armamos la interface como se muestra en la figura anterior.
- 2.- Procedimos a calcular los valores para los cuales se deseaba transmitir.
- 3.- Creamos el programa para la EPROM del microcontrolador, con una dirección TCP/IP y la velocidad que se deseaba probar (9600 bps) y con la información que se deseaba transmitir, para la segunda prueba lo hicimos para que utilizara el puerto 2, específicamente el pin 2.0, el

pin 2.1 y el pin 2.2 donde se recibiría la señal que envían los sensores y posteriormente se enviaría estos datos; este programa se lo ejecuta en lenguaje ensamblador de donde se obtiene la información para grabar la EPROM. (Ver numeral 5.3.2.).

- 4.- Se conectó la interface directamente al módem ubiquity desde el MAX 232 mediante 3 cables (Transmisión, Recepción y Tierra) y un conector DB25 hembra con un adaptador DB25M-DB9H para realizar una transmisión serial hasta el otro módem ubiquity conectado a una computadora.
- 5.- Creamos un programa en lenguaje Visual Basic versión 4.0 para crear el enlace servidor – cliente, el cual recibiría la información que le envíe el microcontrolador por medio de los sensores, mostrándose en la pantalla de la computadora falla en nivel de video, falla en nivel de audio y falla en nivel de energía según se de el caso cuando se produzca un 0 lógico en el pin respectivo del puerto 2 (pin2.0, pin2.1 y pin2.2 respectivamente). (Ver numeral 5.3.1.)

Esta prueba se la realizó y fue un éxito.

CAPITULO V

PROGRAMAS DE EJECUCIÓN

Ejecutamos varios programas, los que describiremos a continuación:

5.1. PROGRAMA EN LENGUAJE VISUAL BASIC VERSION 4.0 QUE COMUNICA DOS COMPUTADORAS Y PERMITE EJECUTAR TRANSMISION ENTRE ELLOS.

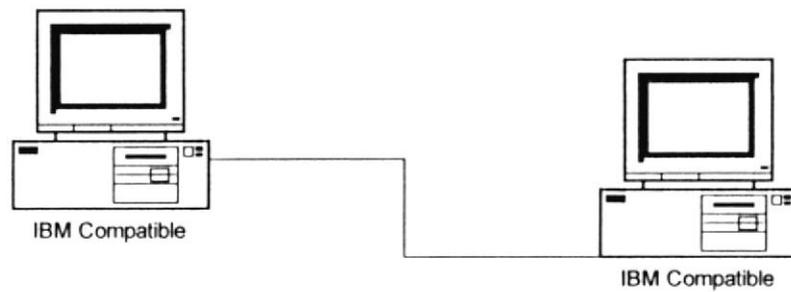


Fig. # 19 Ilustración de la conexión entre dos computadoras.

Vía Módem nulo.

Este programa nos muestra la comunicación vía no módem o módem nulo entre servidor y cliente, permitiendo al cliente (una computadora "B") solicitar cualquier información de archivo que posea el servidor (una computadora "A"), en otras palabras la computadora "B" puede buscar un archivo en el computadora "A" y grabarlo para él; es así que se tienen dos programas uno para cada uno.

Debemos recalcar que la comunicación es de tipo serial, lo que nos permite una transmisión más directa pero hasta cierta distancia ya que para distancias más largas se deben emplear módems, a una velocidad de 9600 bps, y que se emplearán 2 conectores hembras DB25, los mismos que se encontraban conectados con tres cable (Transmisión, Recepción y Tierra), para enlazar los dos computadoras.

5.1.1. PROGRAMA PARA EL SERVIDOR

```
VERSION 4.00
```

```
Begin VB.Form Servidor
```

```
    BorderStyle      = 3  'Fixed Dialog
```

```
    Caption          = "Servidor"
```

```
    ClientHeight     = 3696
```

```
    ClientLeft      = 1092
```

```
ClientTop      = 1188
ClientWidth   = 7392
Height        = 4080
Left          = 1044
LinkTopic     = "Form1"
ScaleHeight   = 3696
ScaleWidth    = 7392
Top           = 852
Width         = 7488
```

```
Begin VB.TextBox Text1
```

```
    Height      = 372
    Left        = 240
    TabIndex    = 3
    Text        = " "
    Top         = 2040
    Width       = 6612
```

```
End
```

```
Begin VB.CommandButton Cerrar
```

```
    Cancel      = -1 'True
    Caption     = "Cerrar"
    Height      = 495
    Left        = 3240
    TabIndex    = 2
```

```
Top = 3000
Width = 855
End
Begin VB.TextBox Estado
BeginProperty Font
name = "MS Sans Serif"
charset = 1
weight = 700
size = 13.8
underline = 0 'False
italic = 0 'False
strikethrough = 0 'False
EndProperty
Height = 492
Left = 240
TabIndex = 1
TabStop = 0 'False
Top = 840
Width = 6612
End
Begin VB.Label Label2
Caption = "LINEA TRANSMITIDA:"
BeginProperty Font
```

```

        name           =   "MS Sans Serif"
        charset        =   1
        weight         =   700
        size           =   12
        underline      =   0   'False
        italic         =   0   'False
        strikethrough  =   0   'False

```

```
EndProperty
```

```

ForeColor          =   &H00000080&
Height             =   372
Left               =   240
TabIndex          =   4
Top                =   1680
Width              =   3132

```

```
End
```

```
Begin VB.Label Label1
```

```

    AutoSize        =   -1   'True
    Caption         =   "ESTADO :"

```

```
BeginProperty Font
```

```

    name           =   "MS Sans Serif"
    charset        =   1
    weight         =   700
    size           =   12

```

```

        underline      = 0 'False
        italic         = 0 'False
        strikethrough  = 0 'False

```

```
EndProperty
```

```

ForeColor      = &H000000FF&
Height         = 300
Left           = 240
TabIndex       = 0
Top            = 480
Width          = 1248

```

```
End
```

```
Begin MSCommLib.MSComm Comunicación
```

```

Left           = 1320
Top            = 3000
_version       = 65536
_extentx       = 677
_extenty       = 677
_stockprops    = 0
cdtimeout      = 0
commport       = 2
ctsttimeout    = 0
dsrtimeout     = 0
dtrenable      = -1 'True

```

```

handshaking      = 0
inbuffersize    = 1024
inputlen        = 0
interval        = 1000
nulldiscard     = 0 'False
outbuffersize   = 512
parityreplate   = ""
rthreshold      = 1
rtsenable       = 0 'False
settings        = "9600,n,8,1"
sthreshold      = 0

```

```
End
```

```
End
```

```
Attribute VB_Name = "Servidor"
```

```
Attribute VB_Creatable = False
```

```
Attribute VB_Exposed = False
```

```
' SERVIDOR - Este programa responderá a las
peticiones
```

```
' del cliente, transmitiéndole informaciones
tales
```

```
' como el directorio actual, la lista de
archivos o el
```

```
' contenido de un archivo
```

Option Explicit

' Cadena en la que se irán almacenando
' los comandos que se reciban del cliente

Dim Cadena As String

' Se ha pulsado el botón Cerrar o la tecla ESC

Private Sub Cerrar_Click()

Unload Me ' terminar

End Sub

' Cada vez que se reciba un carácter del cliente

Private Sub Comunicación_OnComm()

' Si se ha recibido al menos un carácter

If Comunicación.CommEvent = comEvReceive _

And Comunicación.InBufferCount <> 0 Then

' Tomarlo y añadirlo a la cadena

Cadena = Cadena & Comunicación.Input

' Si hemos recibido el retorno de carro

```

If Asc(Right(Cadena, 1)) = 13 Then
ProcesaComando ' hay que procesar el comando
    End If
    ' Si el evento no se ha producido por una
recepción
ElseIf Comunicación.CommEvent <> comEvReceive
Then
    ' Es que se ha producido algún error
MsgBox"Se produce el error " &
Comunicación.CommEvent & _
    " durante la comunicación"
End If

End Sub

' Se ha completado un comando que hay que
ejecutar

Private Sub ProcesaComando()
    ' Sólo comparamos los primeros cuatro caracteres
Select Case Left(Cadena, 4)

Case "IDEN" ' Petición de indentificación

```

```
Estado = "Enviando señal de conexión"
```

```
DoEvents
```

```
Comunicación.Output = "VALE" ' responder
```

```
Case "DIRE" ' Enviar un directorio
```

```
Estado = "Enviando lista de archivos"
```

```
DoEvents
```

```
Dim Archivo As String
```

```
' Obtener una lista de todos los directorios
```

```
Archivo = Dir("*.*", vbDirectory)
```

```
Do While Len(Archivo) <> 0 ' Mientras no hayamos
```

```
llegado al último
```

```
' No incluir la referencia al directorio actual
```

```
If Archivo <> "." Then
```

```
    If Archivo = ".." Then ' Si hay un directorio  
superior
```

```

Comunicación.Output = "[..]" & Chr(13) '
indicarlo

' Si es un directorio
ElseIf GetAttr(Archivo) = vbDirectory Then
' enviarlo
Comunicación.Output = "[" & Archivo & "]" &
Chr(13)

End If

End If

Archivo = Dir ' Leer el siguiente
Loop ' Hasta leer toda la lista

' Ahora leer los archivos
Archivo = Dir("*.*)

' Mientras no obtengamos una cadena vacía
Do While Len(Archivo) <> 0

Do While Comunicación.OutBufferCount+Len(Archivo)+25 > _
Comunicación.OutBufferSize

' Esperar hasta que haya espacio

```

Loop

```
' Enviar el nombre del archivo  
Comunicación.Output = Archivo & Chr(13)
```

```
Archivo = Dir ' y tomar el siguiente
```

Loop

```
' Indicar que se ha terminado  
Comunicación.Output = "FinDirectorio"
```

```
Case "CURD" ' Petición del directorio actual
```

```
Estado = "Enviando directorio actual"
```

```
DoEvents
```

```
' Enviarlo
```

```
Comunicación.Output = CurDir & Chr(13)
```

```
Case "CHDI" ' Petición de cambio de directorio
```

```
Estado = "Cambiando el directorio actual"
```

```
DoEvents
```

```
' Cambiarlo

ChDir Mid(Cadena, 6, Len(Cadena) - 6)

Comunicación.Output = "VALE" ' y notificar que se
ha hecho

Case "CARG" ' Petición del contenido de un
archivo

Estado = "Enviando contenido de un archivo"

DoEvents

' Abrir el archivo solicitado para lectura
Open Mid(Cadena, 6, Len(Cadena) - 6) For Input As
1

' Mientras no hayamos llegado al final
Do While Not EOF(1)

' Leer una línea completa
Line Input #1, Cadena

' Evitar producir un error de desbordamiento
```

```
Do While Comunicación.OutBufferCount +
Len(Cadena) + 25 > Comunicación.OutBufferSize
    ' Esperar hasta que haya espacio
Loop

    ' Enviar la línea leída
Comunicación.Output = Cadena & Chr(13) & Chr(10)

Text1.Text = Left$(Cadena, 51)

Loop

Close #1 ' Cerrar el archivo

    ' Notificar que se ha terminado
Comunicación.Output = "[F]" & Chr(13)

End Select

Estado = "En espera"
Cadena = "" ' Preparar para un nuevo comando

End Sub
```

```
' Al cargar el formulario
Private Sub Form_Load()
    On Error Resume Next
    Comunicación.PortOpen = True ' abrir el puerto de
    comunicaciones
    If Err = 8005 Then
        MsgBox "Puerto ya se encuentra abierto"
    End If
    Estado = "En espera"
End Sub

' Al descargar el formulario
Private Sub Form_Unload(Cancel As Integer)
    On Error Resume Next
    Comunicación.PortOpen = False ' cerrar el puerto
    de comunicaciones
    If Err Then
        MsgBox "Puerto ya se encuentra cerrado"
    End If
End Sub
```

5.1.2. PROGRAMA PARA EL CLIENTE

VERSION\ 4.00

Begin VB.Form Cliente

BorderStyle = 3 'Fixed Dialog

Caption = "Cliente"

ClientHeight = 5055

ClientLeft = 45

ClientTop = 1080

ClientWidth = 9240

ForeColor = &H00FFFFFF&

Height = 5460

Left = -15

LinkTopic = "Form1"

ScaleHeight = 5055

ScaleWidth = 9240

Top = 735

Width = 9360

Begin VB.TextBox NombreArchivoRemoto

Height = 285

Left = 3360

TabIndex = 8

TabStop = 0 'False

```
Top = 360
Width = 3975
End

Begin VB.TextBox ContenidoArchivoRemoto
Height = 3375
Left = 3360
MultiLine = -1 'True
ScrollBars = 2 'Vertical
TabIndex = 6
Top = 720
Width = 5655
End

Begin VB.TextBox CaminoActual
Height = 285
Left = 120
TabIndex = 5
TabStop = 0 'False
Top = 360
Width = 2895
End

Begin VB.PictureBox BarraEstado
Align = 2 'Align Bottom
BackColor = &H00808080&
```

```
BorderStyle = 0 'None
ForeColor = &H00FFFFFF&
Height = 255
Left = 0
ScaleHeight = 255
ScaleWidth = 9240
TabIndex = 3
Top = 4800
Width = 9240
```

End

```
Begin VB.CommandButton Conectar
```

```
Caption = "Conectar"
Height = 495
Left = 120
TabIndex = 2
Top = 4080
Width = 1095
```

End

```
Begin VB.CommandButton LeerDirectorio
```

```
Caption = "Directorio"
Enabled = 0 'False
Height = 495
Left = 1920
```

```
TabIndex      = 1
Top           = 4080
Width        = 1095
```

End

Begin VB.ListBox ListaArchivos

```
Height        = 3375
Left          = 120
TabIndex      = 0
Top           = 720
Width        = 2895
```

End

Begin VB.Timer Reloj

```
Left          = 4080
Top           = 4200
```

End

Begin VB.Label Label2

```
AutoSize      = -1 'True
Caption       = "Archivo remoto"
Height        = 195
Left          = 3360
TabIndex      = 7
Top           = 120
Width        = 1065
```

End

Begin VB.Label Label1

```
AutoSize      = -1  'True
Caption       = "Directorio actual"
Height        = 195
Left          = 120
TabIndex      = 4
Top           = 120
Width         = 1155
```

End

Begin MSCommLib.MSComm Comunicación

```
Left          = 3480
Top           = 4200
_version      = 65536
_extentx      = 847
_extenty      = 847
_stockprops   = 0
cdtimeout     = 0
commport      = 2
ctsttimeout   = 0
dsrtimeout    = 0
dtrenable     = 0  'False
handshaking   = 0
```

```
inbuffersize = 4096
inputlen      = 0
interval      = 10
nulldiscard   = 0 'False
outbuffersize = 512
parityreplace = ""
rthreshold    = 0
rtsenable     = 0 'False
settings      = "9600,n,8,1"
sthreshold    = 0

End

End

Attribute VB_Name = "Cliente"
Attribute VB_Creatable = False
Attribute VB_Exposed = False
Option Explicit

' CLIENTE - Este programa actuará como cliente
en la
' comunicación, permitiendo así acceder a los
archivos
' del sistema en el que se está ejecutando el
SERVIDOR
```

```
' Estas constantes representan las dos
operaciones básicas
```

```
' que es posible realizar
```

```
Const EstLeyendoDirectorio = 1 ' Leer del
directorio del servidor
```

```
Const EstCargandoArchivo = 2 ' y ver un archivo
de texto
```

```
' Esta variable indicará si hay o no conexión
```

```
Dim Conectado As Boolean
```

```
' Cadena para ir recibiendo datos
```

```
Dim Cadena As String
```

```
' Esta variable nos indicará el estado actual
```

```
' de las comunicaciones
```

```
Dim Estado As Byte
```

```
' Cada vez que se reciba algo o se produzca
```

```
' un error se generará el evento OnComm
```

```
Private Sub Comunicación_OnComm()
```

```

' Si aún no se ha comprobado la conexión o
' no se está realizando ninguna de las
' operaciones posibles
If Not Conectado Or Estado = 0 Then Exit Sub ' no
continuar

Dim SubCadena As String ' Para leer el contenido
actual del control

' Comprobar si se ha recibido algo
If Comunicación.CommEvent = comEvReceive _
And Comunicación.InBufferCount <> 0 Then
' leerlo
SubCadena = Comunicación.Input
Cadena = Cadena & SubCadena ' y añadirlo a la
cadena

Select Case Estado ' Según la operación que se
esté realizando
Case EstLeyendoDirectorio ' Si se está leyendo un
directorio

```

```

' Mientras existan retornos de carro en la
cadena
Do While (InStr(Cadena, Chr(13)))
' Ir seccionando
SubCadena = Left(Cadena, InStr(Cadena, Chr(13)) -
1)
' y añadir el elemento a la lista
listaarchivos.AddItem SubCadena
' Eliminar de Cadena la porción extraída
Cadena = Mid(Cadena, InStr(Cadena, Chr(13)) + 1)
Loop ' repetir hasta el fin de la cadena

' Si se ha recibido la señal de fin de
directorio
If Cadena = "FinDirectorio" Then
BarraEstado.Cls ' Indicarlo en la barra de estado
BarraEstado.Print "Directorio completado."
Cadena = ""
LeerDirectorio.Enabled = True ' Activar el botón
Comunicación.RThreshold = 0 ' Desactivar el envío
de mensajes
Estado = 0 ' No se está realizando operación
alguna

```

```
MousePointer = vbArrow ' Devolver el icono normal

End If ,

Case EstCargandoArchivo ' Se está cargando un
programa

' En caso de que se haya recibido la señal de
fin de transmisión

' del archivo

If (Len(Cadena) Mod 512 = 0) Then

ContenidoArchivoRemoto = Left(Cadena, Len(Cadena)
- 4)

End If

If Right(Cadena, 4) = "[F]" & Chr(13) Then

Rem If 1 Then

' Pasarlo a la caja de texto para visualizarlo
ContenidoArchivoRemoto = Left(Cadena, Len(Cadena)
- 4)

Estado = 0 ' No se está realizando operación
alguna
```

```
Cadena = ""
```

```
BarraEstado.Cls ' Indicar que se ha terminado la  
carga
```

```
BarraEstado.Print "Archivo cargado. Servidor en  
espera."
```

```
MousePointer = vbArrow ' Devolver el icono normal  
End If
```

```
End Select
```

```
End If
```

```
End Sub
```

```
' Intentar conectar con el servidor
```

```
Private Sub Conectar_Click()
```

```
MousePointer = vbHourglass ' Activar el reloj de  
espera en el formulario
```

```
Conectar.Enabled = False ' En principio no hay  
conexión
```

```
' Indicar en la barra de estado que se está
' intentando la conexión
BarraEstado.Cls
BarraEstado.Print "Intentando conectar con el
servidor, espere ..."

' No generar el evento
Comunicación.RThreshold = 0
' Enviar la cadena
Comunicación.Output = "IDEN" & Chr(13)
' Esperar la respuesta durante tres segundos
Reloj.Interval = 3000
End Sub

' Al cargar el formulario
Private Sub Form_Load()
On Error Resume Next
Comunicación.PortOpen = True ' Abrir el puerto de
comunicación
If Err = 8005 Then
MsgBox "Puerto ya se encuentra abierto"
```

```
End If

End Sub

' Al descargar el formulario
Private Sub Form_Unload(Cancel As Integer)
Comunicación.PortOpen = False ' Cerrar el puerto
de comunicación
End Sub

' Se ha pulsado el botón de obtención de
directorio
Private Sub LeerDirectorio_Click()
If Not Conectado Then ' Si no hay conexión
MsgBox "El servidor no contesta" ' indicarlo
Exit Sub ' y no continuar
End If

MousePointer = vbHourglass ' Fijar el cursor de
espera

Cadena = "" ' Inicializar la cadena para
recepción
```

```
Comunicación.Output = "CURD" & Chr(13) ' Pedir el
directorio actual
```

```
' Esperar hasta recibir el retorno de carro
Do While Right(Cadena, 1) <> Chr(13)
Cadena = Cadena & Comunicación.Input ' Mientras
tanto ir leyendo los caracteres que llegan
Loop
```

```
' Tomar el camino actual y mostrarlo
CaminoActual = Left(Cadena, Len(Cadena) - 1)
```

```
LeerDirectorio.Enabled = False ' Desactivar el
botón
listaarchivos.Clear ' y limpiar la lista de
archivos
```

```
BarraEstado.Cls ' Indicar que se está obteniendo
el directorio
BarraEstado.Print "Obteniendo directorio del
servidor, espere ..."
```

```
Comunicación.RThreshold = 1 ' Activar el evento
```

```
Cadena = "" ' Inicializar la cadena

' Indicar al servidor que nos facilite un
directorio
Comunicación.Output = "DIRE" & Chr(13)

Estado = EstLeyendoDirectorio ' Estamos leyendo
un directorio

End Sub

' Al hacer una doble pulsación sobre
' un elemento de la lista de archivos
Private Sub listaarchivos_DblClick()

' Si se ha pulsado sobre un directorio
If Left(listaarchivos.Text, 1) = "[" Then

' No permitir otra pulsación
listaarchivos.Enabled = False

' Enviar al servidor la petición de cambio de
directorio
```

```

Comunicación.Output      =      "CHDI      "      &
Mid(listaarchivos.Text, 2, Len(listaarchivos) -
2) & Chr(13)
Do While Comunicación.Input <> "VALE"
    ' Esperar hasta que se haya efectuado el cambio,
    ' momento en el que se recibirá la cadena "VALE"
Loop

LeerDirectorio_Click ' Releer el directorio

listaarchivos.Enabled = True ' Volver a activar
la lista de archivos

ElseIf Right(listaarchivos.Text, 4) = ".TXT" Or 1
Then ' Se ha pulsado sobre un archivo de texto

MousePointer = vbHourglass ' Activar el icono de
espera

    ' Mostrar el nombre del archivo que se va a
cargar

NombreArchivoRemoto = listaarchivos.Text

```

```
BarraEstado.Cls ' Indicar que se está cargando
BarraEstado.Print "Cargando archivo, espere ..."

' Producir el evento OnComm antes
' de que se llene el espacio intermedio de
entrada

Rem Comunicación.RThreshold =
Comunicación.InBufferSize - 512
Comunicación.RThreshold = 1 ' Activar el evento

Cadena = "" ' Inicializar la cadena

' Solicitar al servidor la carga del archivo
Comunicación.Output = "CARG " &
listaarchivos.Text & Chr(13)

Estado = EstCargandoArchivo ' Se está cargando un
archivo

End If
```

End Sub

' Han pasado los tres segundos desde que se
' envió al servidor la petición de conexión

Private Sub Reloj_Timer()

' Contador del número de intentos de conexión

Static Intentos As Byte

' Si no se ha recibido la cadena VALE

If Comunicación.Input <> "VALE" Then

If Intentos < 3 Then ' Si se ha intentado menos
de cuatro veces

Intentos = Intentos + 1 ' intentar una vez más

Comunicación.Output = "IDEN" & Chr(13) ' Enviar
de nuevo la petición de conexión

Exit Sub ' y salir

Else ' Si no es posible la conexión

BarraEstado.Cls ' Indicarlo en la barra de estado

BarraEstado.Print "Imposible conectar, el
servidor no responde"

```
Conectar.Enabled = True ' Visualizar y activar el
botón para
```

```
Conectar.Visible = True ' permitir intentarlo
después
```

```
MousePointer = vbArrow ' Restablecer el icono del
ratón
```

```
End If
```

```
Else ' Si se ha recibido la cadena VALE
```

```
Conectado = True ' se ha conseguido conectar
```

```
LeerDirectorio.Enabled = True ' Activar el botón
de lectura de directorio
```

```
Conectar.Visible = False
```

```
BarraEstado.Cls ' Indicar en la barra de estado
que se ha efectuado la conexión
```

```
BarraEstado.Print "Conexión efectuada, servidor
en espera."
```

```

MousePointer = vbArrow ' Restablecer el icono del
ratón

```

```

End If \

```

```

Reloj.Interval = 0 ' No. generar más veces el
evento

```

```

Intentos = 0 ' Poner a 0 el número de intentos

```

```

End Sub

```

5.2. PROGRAMA EN LENGUAJE VISUAL BASIC VERSION 4.0 QUE PERMITE LA COMUNICACION DESDE LA INTERFACE (87c52 y MAX 232) A LA COMPUTADORA.

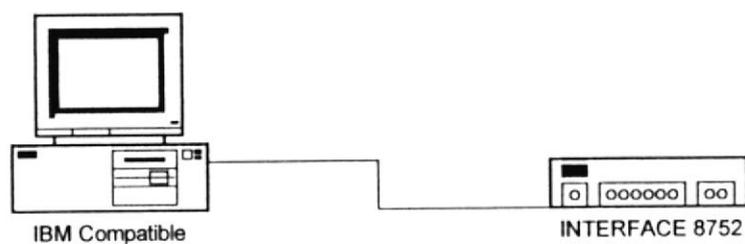


Fig. # 20 Ilustración de la conexión entre la computadora y la interfase que contiene al 8752. Vía Módem nulo.

Para este programa vía módem nulo solamente se utilizó la parte del servidor, ya que el cliente lo hacía la interface, quien era la que enviaba la señal es decir quien originaba la comunicación.

Para ello se creó un programa en lenguaje ensamblador para la interface, que nos permitía enviar bajos mediante pulsos con una botonera los que se mostraban como números en el monitor de la computadora; posteriormente se conectó la interface a la computadora mediante 3 cables (Transmisión, Recepción y Tierra) que iban desde el MAX 232 hacia la computadora conectados por un conector hembra DB25, para utilizar el puerto serial y ejecutar una transmisión del mismo tipo, ya que este tipo de transmisión (serial) es la más óptima para largas distancias cuando se emplean módems.

Y es de esta manera que se crearon los siguientes programas:

5.2.1. PROGRAMA PARA EL SERVIDOR

```
VERSION 4.00
```

```
Begin VB.Form Com8752
```

```
    BorderStyle      = 3    'Fixed Dialog
```

```
    Caption          = "COMUNICACION CON EL 8752"
```

```
    ClientHeight     = 3690
```

```
ClientLeft      = 1125
ClientTop       = 1500
ClientWidth     = 7395
Height          = 4095
Left            = 1065
LinkTopic       = "Form1"
ScaleHeight    = 3690
ScaleWidth     = 7395
Top             = 1155
Width           = 7515
```

```
Begin VB.TextBox Text1
```

```
    Height      = 372
    Left        = 240
    TabIndex    = 3
    Text        = " "
    Top         = 2040
    Width       = 6612
```

```
End
```

```
Begin VB.CommandButton Cerrar
```

```
    Cancel      = -1 'True
    Caption     = "Cerrar"
    Height      = 495
    Left        = 3240
```

```
    TabIndex      = 2
    Top           = 3000
    Width        = 855

End

Begin VB.TextBox Estado...
    BeginProperty Font
        name       = "MS Sans Serif"
        charset    = 1
        weight     = 700
        size       = 13.5
        underline  = 0 'False
        italic     = 0 'False
        strikethrough = 0 'False
    EndProperty
    Height        = 492
    Left         = 240
    TabIndex     = 1
    TabStop      = 0 'False
    Top         = 840
    Width       = 6612

End

Begin VB.Label Label2
    Caption      = "LINEA TRANSMITIDA:"
```

BeginProperty Font

```
name           = "MS Sans Serif"
charset        = 1
weight         = 700
size          = 12
underline      = 0 'False
italic         = 0 'False
strikethrough  = 0 'False
```

EndProperty

```
ForeColor      = &H00000080&
Height         = 372
Left           = 240
TabIndex       = 4
Top            = 1680
Width          = 3132
```

End

Begin VB.Label Label1

```
AutoSize       = -1 'True
Caption        = "ESTADO :"
```

BeginProperty Font

```
name           = "MS Sans Serif"
charset        = 1
weight         = 700
```

```
        size           = 12
        underline      = 0   'False
        \ italic       = 0   'False
        strikethrough  = 0   'False
```

```
EndProperty
```

```
ForeColor      = &H000000FF&
Height         = 300
Left           = 240
TabIndex       = 0
Top            = 480
Width          = 1248
```

```
End
```

```
Begin MSCommLib.MSComm Comunicación
```

```
Left           = 1320
Top            = 3000
_version       = 65536
_extentx       = 847
_extenty       = 847
_stockprops    = 0
cdtimeout      = 0
commport       = 2
ctsttimeout    = 0
dsrtimeout     = 0
```

```
dtrenable      = -1 'True
handshaking    = 0
inbuffersize   = 1024
inputlen       = 0
interval       = 1000
nulldiscard    = 0 'False
outbuffersize  = 512
parityreplace  = ""
rthreshold     = 1
rtsenable      = 0 'False
settings       = "19200,n,8,1"
sthreshold     = 0

End

End

Attribute VB_Name = "Com8752"
Attribute VB_Creatable = False
Attribute VB_Exposed = False

' SERVIDOR - Este programa responderá a las
peticiones

' del cliente, transmitiéndole informaciones
tales

' como el directorio actual, la lista de
archivos o el
```

```
' contenido de un archivo
```

```
Option Explicit
```

```
' Cadena en la que se irán almacenando
```

```
' los comandos que se reciban del cliente
```

```
Dim Cadena As String
```

```
Dim LinTrans As Integer
```

```
' Se ha pulsado el botón Cerrar o la tecla ESC
```

```
Private Sub Cerrar_Click()
```

```
Unload Me ' terminar
```

```
End Sub
```

```
' Cada vez que se reciba un carácter del cliente
```

```
Private Sub Comunicación_OnComm()
```

```
' Si se ha recibido al menos un carácter
```

```
If Comunicación.CommEvent = comEvReceive _
```

```
And Comunicación.InBufferCount <> 0 Then
```

```
' Tomarlo y añadirlo a la cadena
```

```
Cadena = Cadena & Comunicación.Input
```

```

' Si hemos recibido el retorno de carro
-----

If Asc(Right(Cadena, 1)) = 13 Then
  ' If Len(Cadena) >= 1 Then
ProcesaComando ' hay que procesar el comando
End If
  ' Si el evento no se ha producido por una
recepción
ElseIf Comunicación.CommEvent <> comEvReceive
Then
  ' Es que se ha producido algún error
MsgBox"Se produce ..... el error "&
Comunicación.CommEvent & _
      " durante la comunicación"
End If

End Sub

' Se ha completado un comando que hay que
ejecutar
Private Sub ProcesaComando(.)
  ' Sólo comparamos los primeros cuatro caracteres
Dim Cad1 As String

```

```
Comunicación.Output = Str(324156133) ' responder
```

```
LinTrans\ = LinTrans + 1
```

```
Cad1 = ""
```

```
Select Case Left(Cadena, 2)
```

```
Case "I1" ' Petición de identificación
```

```
Estado = "Enviando señal de conexión"
```

```
DoEvents
```

```
Comunicación.Output = Str(100) ' responder
```

```
Cad1 = Cad1 & " 100 "
```

```
Case "I2" ' Enviar un directorio
```

```
Estado = "Enviando lista de archivos"
```

```
DoEvents
```

```
Dim Archivo As String
```

```
Dim NumFil As Integer
```

```
' Obtener una lista de todos los directorios
```

```
Archivo = Dir("*.*", vbDirectory)
```

```
NumFil = 0
```

```

Do While Len(Archivo) <> 0 ' Mientras no hayamos
llegado al último

    ' No incluir la referencia al directorio actual
    If Archivo <> "." Then
        If Archivo = ".." Then ' Si hay un directorio
superior
            Comunicación.Output = "[..]" & Chr(13) '
indicarlo

            ' Si es un directorio
            ElseIf GetAttr(Archivo) = vbDirectory Then
                ' enviarlo
                Comunicación.Output = "[" & Archivo & "]" &
Chr(13)

                End If

            End If

            NumFil = NumFil + 1

            Archivo = Dir ' Leer el siguiente
Loop ' Hasta leer toda la lista

    ' Ahora leer los archivos
    Archivo = Dir("*.*")

```

```

' Mientras no obtengamos una cadena vacía
Do While Len(Archivo) <> 0

'
Do
Comunicación.OutBufferCount+Len(Archivo)+25 > _
Comunicación.OutBufferSize
' Esperar hasta que haya espacio
Loop

' Enviar el nombre del archivo
Comunicación.Output = Archivo & Chr(13)
NumFil = NumFil + 1
Archivo = Dir ' y tomar el siguiente
Loop
NumFil = NumFil Mod 256
' Indicar que se ha terminado
Comunicación.Output = Str(NumFil)
Cad1 = Cad1 & " 101 "

Case "I3" ' Petición del directorio actual

Estado = "Enviando directorio actual"
DoEvents

```

```
' Enviarlo

Comunicación.Output = Str(102)

Cad1 = Cad1 & " 102 "
```

```
' Open Mid(Cadena, 6, Len(Cadena) - 6) For Input
As 1

' Mientras no hayamos llegado al final
' Do While Not EOF(1)

' Leer una línea completa
' Line Input #1, Cadena
Cadena = "1234"

' Evitar producir un error de desbordamiento
Do While Comunicación.OutBufferCount +
Len(Cadena) + 25 > Comunicación.OutBufferSize
' Esperar hasta que haya espacio
Loop

' Enviar la línea leída
Comunicación.Output = Cadena & Chr(13) & Chr(10)

' Text1.Text = Left$(Cadena, 51)

' Loop

' Close #1 ' Cerrar el archivo
```

```
' Notificar que se ha terminado
Comunicación.Output = Str(104)
Cad1 = Cad1 & " 104 "

End Select

' Cadena = Left(Cadena, 5)
Cad1 = Cad1 & Str(LinTrans) & " " & Cadena
Text1.Text = Cad1
Estado = "En espera"
Cadena = "" ' Preparar para un nuevo comando
End Sub

' Al cargar el formulario
Private Sub Form_Load()
On Error Resume Next
Comunicación.PortOpen = True ' abrir el puerto de
comunicaciones

If Err = 8005 Then
MsgBox "Puerto ya se encuentra abierto"

End If

Estado = "En espera"
End Sub
```

```

' Al descargar el formulario
Private Sub Form_Unload(Cancel As Integer)
On Error Resume Next
Comunicación.PortOpen = False ' cerrar el puerto
de comunicaciones
If Err Then
MsgBox "Puerto ya se encuentra cerrado"
End If
End Sub

```

5.2.2. PROGRAMA EN LENGUAJE ENSAMBLADOR PARA EL MICROCONTROLADOR 87c52

```

ljmp e1

org 23h

ljmp e2

org 30h

e1

mov pcon, #80h

mov scon, #40h

mov tmod, #20h

```

```

mov th2,#255
mov rcap2h,#255
mov a,p1
anl a,#24
cjne a,#0,e12
mov th1,#217 ;217 246 251
mov t11,#217 ;217 246 251
mov t12,#22 ;22 197 227
mov rcap21,#22 ;22 197 227
e12
cjne a,#8,e13
mov th1,#246 ;217 246 251
mov t11,#246 ;217 246 251
mov t12,#197 ;22 197 227
mov rcap21,#197 ;22 197 227
e13
cjne a,#16,e14
mov th1,#251 ;217 246 251
mov t11,#251 ;217 246 251
mov t12,#227 ;22 197 227
mov rcap21,#227 ;22 197 227
e14
cjne a,#24,e15

```

```
mov th1,#251          ;217 246 251
mov t11,#251          ;217 246 251
mov t12,#227          ;22 197 227
mov rcap21,#227       ;22 197 227
```

e15

```
mov tcon,#40h
mov t2con,#14h
mov ie,#80h
setb es
setb ren
```

e3

```
mov a,p1
mov b,a
anl b,#3
anl a,#4
cjne a,#4,e3
mov a,#130
lcall delay
```

e5

```
mov a,p1
anl a,#4
cjne a,#0,e5
mov a,#100
```

```
lcall delay
mov a,b
cjne a,#0,e6
mov a,#73
lcall subl
mov a,#49
lcall subl
mov a,#13
lcall subl
ljmp e3
```

e6

```
cjne a,#1,e8
mov a,#73
lcall subl
mov a,#50
lcall subl
mov a,#13
lcall subl
ljmp e3
```

e8

```
cjne a,#2,e9
mov a,#73
lcall subl
```

```
    mov a,#51
    lcall subl
    mov a,#13
    lcall subl
    ljmp e3
e9
    cjne a,#3,e10
    mov a,#73
    lcall subl
    mov a,#52
    lcall subl
    mov a,#13
    lcall subl
e10
    ljmp e3

    org 100h
e2
    clr ri
    mov b,sbuf
    nop
    mov p2,b
```

```
reti
```

```
delay \
```

```
mov r7,a
```

```
del1
```

```
mov r6,#6
```

```
del2
```

```
mov r5,#20
```

```
del3
```

```
djnz r5,del3
```

```
djnz r6,del2
```

```
djnz r7,del1
```

```
ret
```

```
sub1
```

```
clr es
```

```
mov sbuf,a
```

```
e7
```

```
nop
```

```
jnb ti,e7
```

```
nop
```

```
clr ti
```

```
setb es
```

```

ret
end

```

5.3. PROGRAMA EN LENGUAJE VISUAL BASIC VERSION 4.0 QUE PERMITE SIMULAR LA COMUNICACIÓN A TRAVES DE MODEM CON LA INTERFACE 87c52 AL QUE SE ENCUENTRAN CONECTADOS LOS SENSORES DE AUDIO, VIDEO, Y ENERGIA HASTA OTRA COMPUTADORA EN UNA ESTACIÓN CENTRAL .

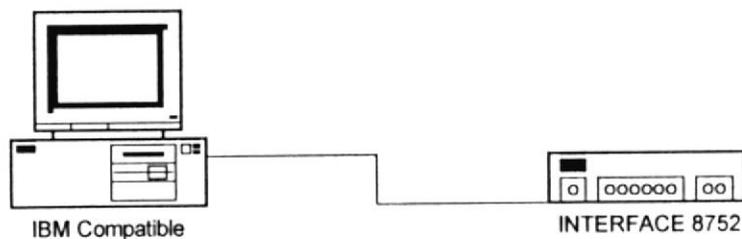


Fig. # 21 Ilustración de la conexión entre la computadora
Y la interfase con los sensores vía Módem.

Para este programa vía módem solamente se utilizó la parte del servidor, ya que el cliente lo hacia la interface, quien era la que enviaba la señal es decir quien originaba la comunicación.

Para ello se creo un programa en lenguaje ensamblador para la interface, que nos permitía sensar el puerto 2, el cual cuando existía niveles normales (1 lógico) de

audio, video y energía no mostraba alteración, pero cuando se perdía una o algunas de las tres señales en conjunto (0 lógico en uno o varios de los pines del puerto 2 donde se encontraban conectados los sensores) se transmitía a 9600 bps y se mostraban como mensajes del tipo de falla producida (falla en nivel de audio, falla en nivel de video o falla en nivel de energía) en el monitor de la computadora receptor; posteriormente se conecto la interface al módem mediante 3 cables (Transmisión, Recepción y Tierra) que iban desde el MAX 232 hacia el módem ubiquity y se receptaba la señal en otro módem, de las mismas características, que se encontraba conectado a una computadora

Y es de esta manera que se crearon los siguientes programas:

5.3.1. PROGRAMA PARA EL SERVIDOR

```

VERSION 4.00

Begin VB.Form Com8752
    BorderStyle       = 3   'Fixed Dialog
    Caption           = "COMUNICACION CON EL 8752"
    ClientHeight      = 3690
    ClientLeft        = 1125
    ClientTop         = 1500
    ClientWidth       = 7395

```

```
Height          = 4095
Left            = 1065
LinkTopic      = "Form1"
ScaleHeight    = 3690
ScaleWidth     = 7395
Top            = 1155
Width          = 7515
```

```
Begin VB.TextBox Text1
```

```
    Height      = 372
    Left        = 240
    TabIndex    = 3
    Text        = " "
    Top         = 2040
    Width       = 6612
```

```
End
```

```
Begin VB.CommandButton Cerrar
```

```
    Cancel      = -1 'True
    Caption     = "Cerrar"
    Height      = 495
    Left        = 3240
    TabIndex    = 2
    Top         = 3000
    Width       = 855
```

End

Begin VB.TextBox Estado

BeginProperty Font

name = "MS Sans Serif"
charset = 1
weight = 700
size = 13.5
underline = 0 'False
italic = 0 'False
strikethrough = 0 'False

EndProperty

Height = 492
Left = 240
TabIndex = 1
TabStop = 0 'False
Top = 840
Width = 6612

End

Begin VB.Label Label2

Caption = "LINEA TRANSMITIDA:"

BeginProperty Font

name = "MS Sans Serif"
charset = 1

```
        weight          = 700
        size            = 12
        underline       = 0 'False
        italic          = 0 'False
        strikethrough   = 0 'False

    EndProperty

    ForeColor          = &H00000080&
    Height             = 372
    Left               = 240
    TabIndex           = 4
    Top               = 1680
    Width             = 3132

End

Begin VB.Label Label1

    AutoSize          = -1 'True
    Caption           = "ESTADO :"

    BeginProperty Font
        name           = "MS Sans Serif"
        charset        = 1
        weight         = 700
        size           = 12
        underline      = 0 'False
        italic         = 0 'False
```

```
        strikethrough = 0 'False
EndProperty
ForeColor = &H000000FF&
Height = 300
Left = 240
TabIndex = 0
Top = 480
Width = 1248
End
Begin MSCommLib.MSComm Comunicación
    Left = 1320
    Top = 3000
    _version = 65536
    _extentx = 847
    _extenty = 847
    _stockprops = 0
    cdtimeout = 0
    commport = 2
    ctsttimeout = 0
    dsrtimeout = 0
    dtrenable = -1 'True
    handshaking = 0
    inbuffersize = 1024
```

```
inputlen      = 0
interval      = 1000
nulldiscard   = 0  'False
outbuffersize = 512
parityreplace = ""
rthreshold    = 1
rtsenable     = 0  'False
settings      = "9600,n,8,1"
sthreshold    = 0

End

End

Attribute VB_Name = "Com8752"
Attribute VB_Creatable = False
Attribute VB_Exposed = False

' SERVIDOR - Este programa responderá a las
peticiones

' del cliente, transmitiéndole informaciones
tales          ' como el directorio actual,
la lista de archivos o el ...
' contenido de un archivo

Option Explicit
```

```

' Cadena en la que se irán almacenando
' los comandos que se reciban del cliente
Dim Cadena As String
Dim LinTrans As Integer

' Se ha pulsado el botón Cerrar o la tecla ESC
Private Sub Cerrar_Click()
Unload Me ' terminar
End Sub

' Cada vez que se reciba un carácter del cliente
Private Sub Comunicación_OnComm()

Cadena = ""

' Si se ha recibido al menos un carácter
If Comunicación.CommEvent = comEvReceive _
And Comunicación.InBufferCount <> 0 Then

' Tomarlo y añadirlo a la cadena
Cadena = Cadena & Comunicación.Input

' Si hemos recibido el retorno de carro
' If Asc(Right(Cadena, 1)) = 13 Then

```

```

' If Len(Cadena) >= 1 Then
ProcesaComando ' hay que procesar el comando
' End If
' Si el evento no se ha producido por una
recepción
ElseIf Comunicación.CommEvent <> comEvReceive
Then
' Es que se ha producido algún error
MsgBox"Se produce el error " &
Comunicación.CommEvent & _
" durante la comunicación"
End If

End Sub

' Se ha completado un comando que hay que
ejecutar

Private Sub ProcesaComando()
' Sólo comparamos los primeros cuatro caracteres
Dim Cad1 As String
Dim Num2 As Integer
Dim Num3 As Integer
Comunicación.Output = Chr$(255) ' responder

```

```
LinTrans = LinTrans + 1

Cad1 = ""

Num2 = Asc(Cadena)

Num3 = Num2 And 1

If Num3 = 0 Then

Cad1 = Cad1 & "Falla en señal de Video. "

End If

Num3 = Num2 And 2

If Num3 = 0 Then

Cad1 = Cad1 & "Falla en señal de Audio. "

End If

Num3 = Num2 And 4

If Num3 = 0 Then

Cad1 = Cad1 & "Falla en señal de Energia. "

End If

'Select Case Left(Cadena, 1)

'Case "I1" ' Petición de identificación

'Estado = "Enviando señal de conexión"

'DoEvents
```

```

'Comunicación.Output = Str(100) " " responder
'Cad1 = Cad1 & " 100 "
'Case "I2" ' Enviar un directorio

'Estado = "Enviando lista de archivos"
'DoEvents

'Dim Archivo As String
'Dim NumFil As Integer
'Obtener una lista de todos los directorios
'Archivo = Dir("*.*", vbDirectory)
'NumFil = 0
'Do While Len(Archivo) <> 0 ' Mientras no
hayamos llegado al último

'No incluir la referencia al directorio actual
'If Archivo <> "." Then
'If Archivo = ".." Then ' Si hay un directorio
superior
'Comunicación.Output = "[..]" & Chr(13) '
indicarlo
'Si es un directorio
'ElseIf GetAttr(Archivo) = vbDirectory Then

```

```

' ' enviarlo

'Comunicación.Output = "[" & Archivo & "]" &
Chr(13)

'End If

'End If

'NumFil = NumFil + 1

'Archivo = Dir '^Leer el siguiente

'Loop ' Hasta leer toda la lista

'Ahora leer los archivos

'Archivo = Dir("*.*)"

'Mientras no obtengamos una cadena vacía

'Do While Len(Archivo) <> 0

'Do While Comunicación.OutBufferCount +
Len(Archivo) + 25 > _
'Comunicación.OutBufferSize
' ' Esperar hasta que haya espacio
'Loop

'Enviar el nombre del archivo

'Comunicación.Output = Archivo & Chr(13)

```

```
'NumFil = NumFil + 1
'Archivo = Dir ' y tomar el siguiente
'Loop '
'NumFil = NumFil Mod 256
' ' Indicar que se ha terminado
'Comunicación.Output = Str(NumFil)
' Cad1 = Cad1 & " 101 "
'
'Case "I3" ' Petición del directorio actual
'Estado = "Enviando directorio actual"
'DoEvents
'Enviarlo
'Comunicación.Output = Str(102)
' Cad1 = Cad1 & " 102 "
'
'Case "I4" ' Petición de cambio de directorio
'Estado = "Cambiando el directorio actual"
'DoEvents
'Cambiarlo
```

```
'ChDir Mid(Cadena, 6, Len(Cadena) - 6)
'Comunicación.Output = Str(103) ' y notificar
que se ha hecho
'Cad1 = Cad1 & " 103 "

'Case "I5" ' Petición del contenido de un
archivo

'Estado = "Enviando contenido de un archivo"
'DoEvents

'Abrir el archivo solicitado para lectura
'Open Mid(Cadena, 6, Len(Cadena) - 6) For Input
As 1

'Mientras no hayamos llegado al final
'Do While Not EOF(1)

'Leer una línea completa
'Line Input #1, Cadena
' Cadena = "1234"
'Evitar producir un error de desbordamiento
```

```

'Do While Comunicación.OutBufferCount +
Len(Cadena) + 25 > Comunicación.OutBufferSize
'Esperar hasta que haya espacio
'Loop

'Enviar la línea leída
'Comunicación.Output = Cadena & Chr(13) &
Chr(10)

'Text1.Text = Left$(Cadena, 51)

'Loop

'Close #1 ' Cerrar el archivo

'Notificar que se ha terminado
'Comunicación.Output = Str(104)
'Cad1 = Cad1 & " 104 "

'End Select

' Cadena = Left(Cadena, 5)

' Cad1 = Cad1 & Str(LinTrans) & " " & Cadena

```

```
Text1.Text = Cad1  
Estado = "En espera"  
Cadena = "" ' Preparar para un nuevo comando  
End Sub
```

```
'Al cargar el formulario
```

```
Private Sub Form_Load()  
On Error Resume Next  
Comunicación.PortOpen = True ' abrir el puerto de  
comunicaciones  
If Err = 8005 Then  
MsgBox "Puerto ya se encuentra abierto"  
End If  
Estado = "En espera"  
End Sub
```

```
'Al descargar el formulario
```

```
Private Sub Form_Unload(Cancel As Integer)  
On Error Resume Next  
Comunicación.PortOpen = False ' cerrar el puerto  
de comunicaciones  
If Err Then  
MsgBox "Puerto ya se encuentra cerrado"
```

```
End If
```

```
End Sub
```

5.3.2. PROGRAMA EN LENGUAJE ENSAMBLADOR PARA EL MICROCONTROLADOR 87c52 QUE CONTROLA EL PUERTO 2 Y TRANSMITE A 9600 bps

```
ljmp e1
    org 23h
    ljmp e2
    org 30h
e1
    mov pcon,#80h
    mov scon,#40h
    mov tmod,#20h
    mov th2,#255
    mov rcap2h,#255
    mov a,p1
    anl a,#24
    cjne a,#0,e12
    mov th1,#217           ;217 246 251
    mov tl1,#217           ;217 246 251
    mov tl2,#22            ;22 197 227
```

```

        mov rcap21,#22          ;22 197 227
e12
        cjne a,#8,e13
        mov th1,#246           ;217 246 251
        mov tl1,#246           ;217 246 251
        mov tl2,#197          ;22 197 227
        mov rcap21,#197       ;22 197 227
e13
        cjne a,#16,e14
        mov th1,#251           ;217 246 251
        mov tl1,#251           ;217 246 251
        mov tl2,#227          ;22 197 227
        mov rcap21,#227       ;22 197 227
e14
        cjne a,#24,e15
        mov th1,#251           ;217 246 251
        mov tl1,#251           ;217 246 251
        mov tl2,#227          ;22 197 227
        mov rcap21,#227       ;22 197 227
e15
        mov tcon,#40h
        mov t2con,#14h
        mov ie,#80h

```

```
        setb es
        setb ren
e10     \
        mov a,#120
        lcall delay
        mov a,p2
        lcall sub1
        ljmp e10
```

```
        org 0a0h
e2
        clr ri
        mov b,sbuf
        nop
        mov p2,b
        reti
```

```
delay
        mov r7,a
del1
        mov r6,#6
del2
```

```
        mov r5,#20
del3
        djnz r5,del3
        djnz r6,del2
        djnz r7,del1
        ret

sub1
        clr es
        mov sbuf,a
e7
        nop
        jnb ti,e7
        nop
        clr ti
        setb es
        ret
end
```

CONCLUSIONES

Con el presente trabajo hemos podido determinar:

1. Que la transmisión empleando el sistema CDPD es optima a la velocidad de 9600 bps, a pesar de existir ciertos momentos en los cuales la velocidad se reduce por efectos de tráfico y del mismo recorrido a través de los diferentes tramos de la transmisión.
2. Que la transmisión en el sistema CDPD se limita, como cualquier sistema de transmisión de dato y voz, a la cobertura de la misma y es por esta característica que la aplicación practica de este proyecto se lo puede efectuar únicamente en la zona anteriormente mencionada.
3. Comprobamos la versatilidad en transmisión y recepción de la interface compuesta por el microcontrolador 87c52 de Intel, permitiéndonos trabajar a diferentes velocidades de transmisión de dato (19200 bps, 9600bps, 2400bps, etc.), así como la utilización de sus puertos para receptor señales de +5 V y 0 V (1

lógico y 0 lógico respectivamente) mediante previa programación de su memoria Eprom en language ensamblador.

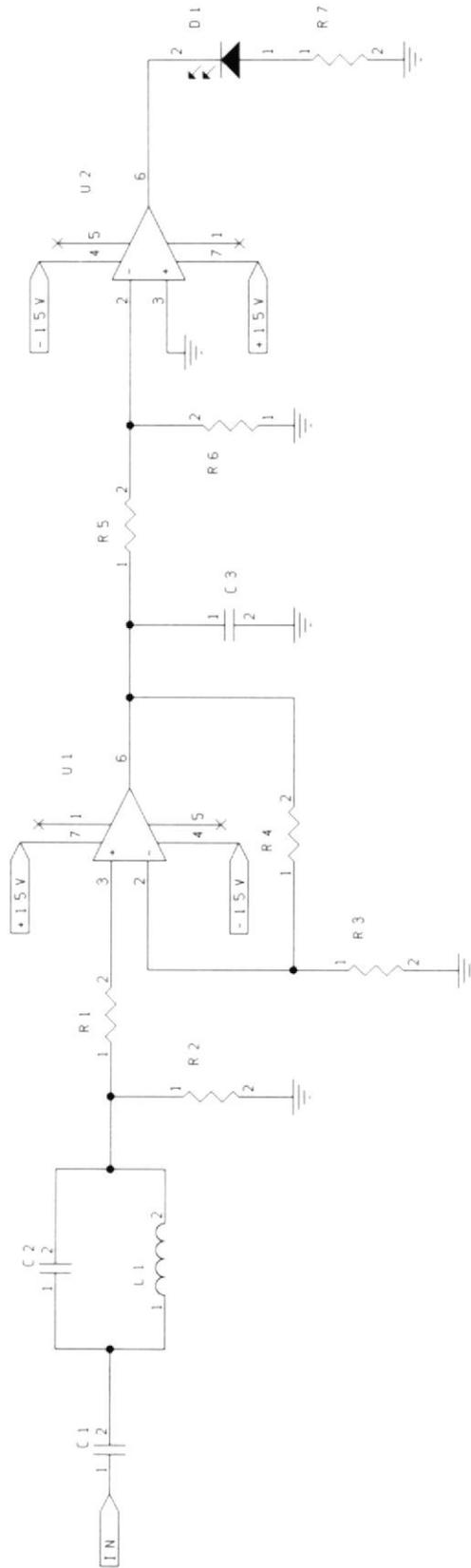
4. Que sin el uso de los sensores de niveles audio, video y energía, no se hubiera podido realizar el proyecto, debido a que los equipos en las estaciones repetidoras de televisión emiten señales de tipo analógico y se las debe convertir a digital para su transmisión.
5. Que el empleo de los microcontroladores y sensores serían los más indicados y de menor costo para este tipo de proyecto en comparación con el empleo de una computadora, por que se debe tomar en cuenta el variado número de antenas repetidoras que poseen instaladas los sistema de televisión.
6. Que el lenguaje de programación Visual Basic es una gran opción para la comunicación, sea vía no módem o módem, debido a sus propiedades de trabajar en base de datos, permitiendo crear un gran archivo para las diferentes estaciones repetidoras de un canal de televisión.

RECOMENDACIONES Y SUGERENCIAS

1. Se recomienda la utilización de este proyecto como base para un equipo mejorado que permita su empleo y comercialización a nivel nacional y como un excelente recurso para las estaciones transmisoras de señales de televisión.
2. Se recomienda emplear en la comunicación el mismo tipo de módem tanto en el lugar de la transmisión (estación repetidora) como en el lugar de la recepción (estación central).
3. Se sugiere que este proyecto se lo emplee como es su propósito, un sistema de alarma, y no como un sistema de monitoreo para reducción de costos.
4. Se sugiere el uso de los microcontroladores y sensores y no el de una computadora para cada estación repetidora por su amplia diferencia de precios.

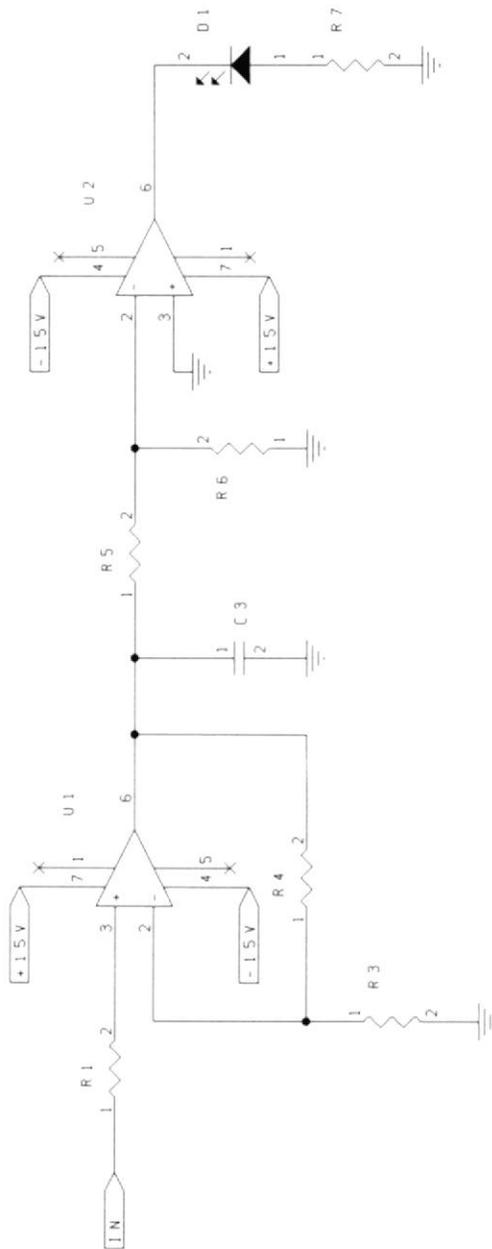
ANEXOS

ANEXO # 1

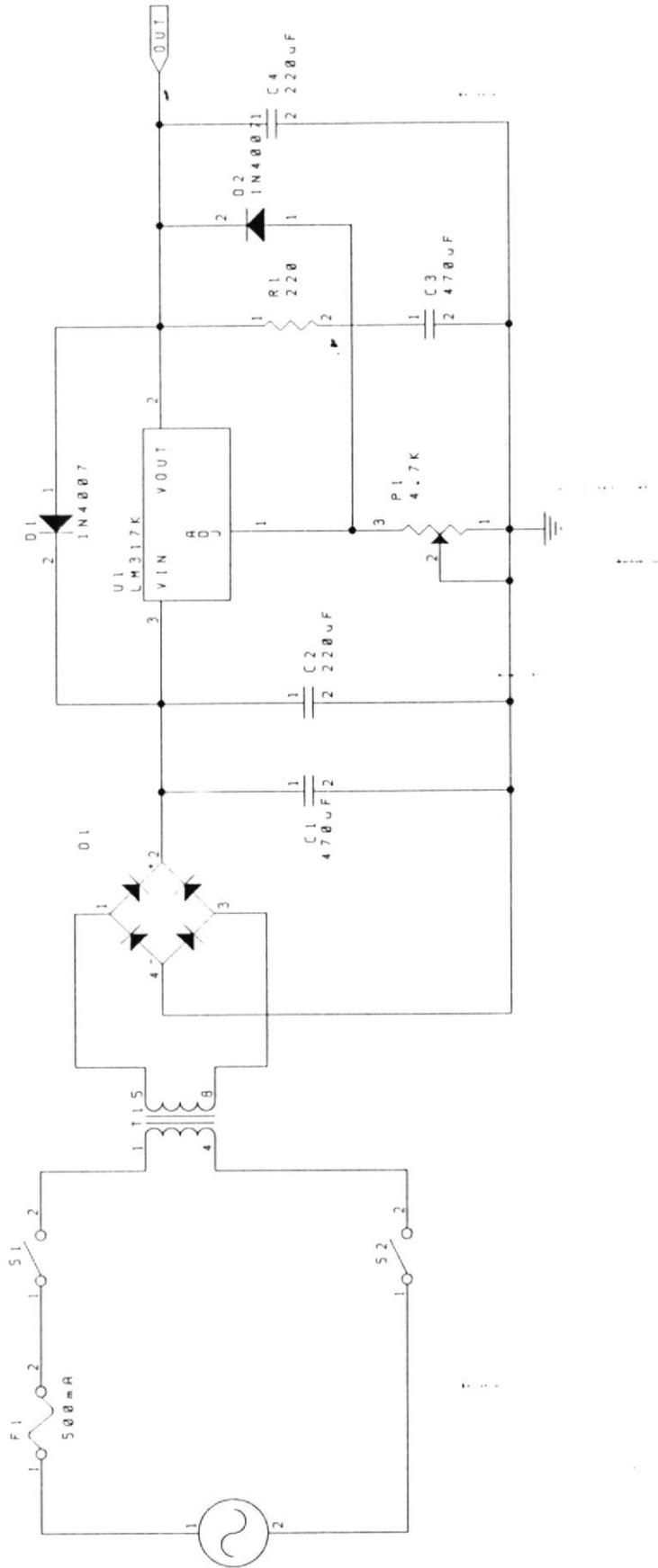


ARMADA DEL ECUADOR DIECAR NIVEL III	
Size	Document Number
A	
Date:	November 13, 1998
Sheet	5
of	
REV	VC2

ANEXO # 2

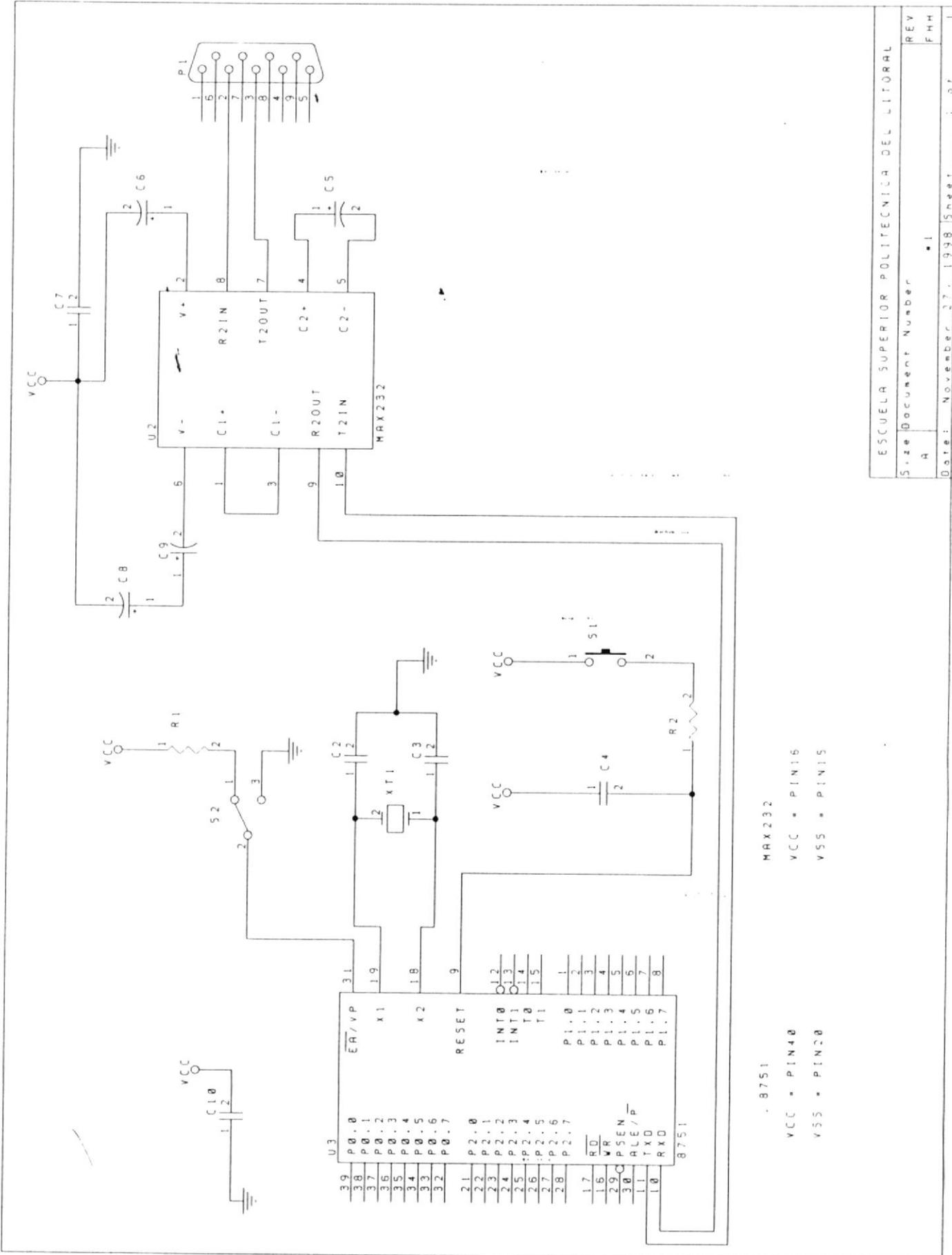


ANEXO 3



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL	
Size	Document Number
A	
Date:	October 12, 1998 Sheet 3 of 3
REV	F.H.H

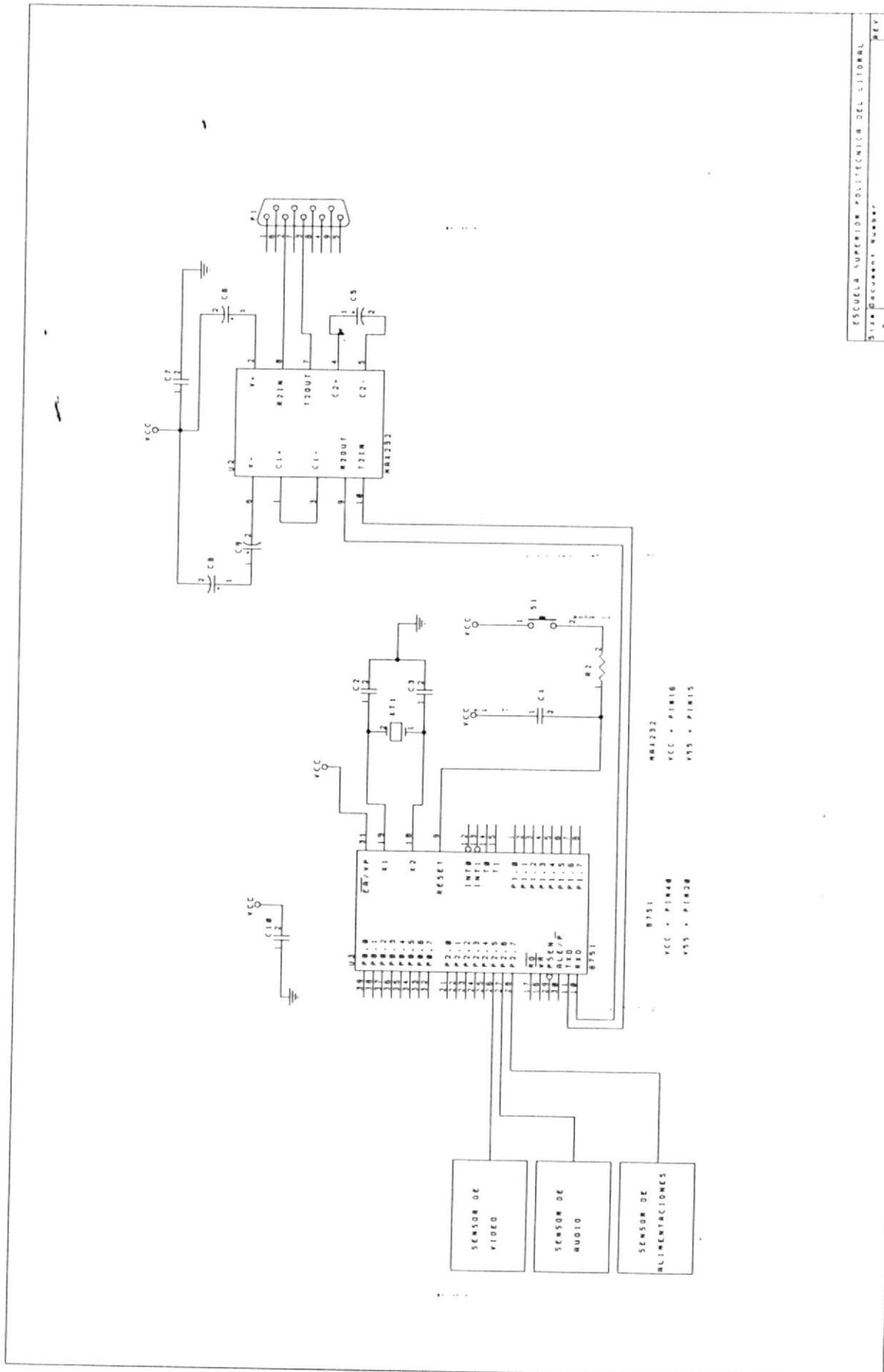
ANEXO 4



8751
VCC = PIN40
VSS = PIN20

MAX232
VCC = PIN16
VSS = PIN15

ANEXO 5



BIBLIOGRAFIA

1. THEODORE S. RAPPAPORT, Wireless Communications principles and practice, Prentice Hall PTR.
2. UYLESS BLACK, Mobile and Wireless Networks, Prentice Hall PTR.
3. WILLIAM J. BEYDA, Data Communications from Basic to Broadband, Prentice Hall PTR.
4. TECHNICAL INFORMATION, Ubiquity celular communications system, Pacific Communication Sciences, Inc.
5. C. BELOVE, Enciclopedia de la Electrónica Ingeniería y Técnica, Oceano/Centrum, Tomo N° 7
6. BC TEL MOBILITY, Celular Digital Packet Data.



A.F. 141843