



**UNIVERSIDAD NACIONAL
AUTONOMA DE MEXICO**

FACULTAD DE QUIMICA

INTERCONEXION DE DOS MICROCOMPUTADORAS
PERSONALES Y COMPARACION DE LAS
TELECOMUNICACIONES DE DOS PAQUETES
INTEGRADOS: FRAMEWORK Y SYMPHONY.

TRABAJO ESCRITO
Que para obtener el Título de
INGENIERO QUIMICO
P r e s e n t a
SERGIO JESUS PARRA SORIA.

México, D. F.

1992

TESIS CON
FALLA DE ORIGEN



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

1	INTRODUCCION.	1
1.1	OBJETIVOS DEL TRABAJO.	1
1.1.1	OBJETIVO PRINCIPAL.	1
1.1.2	LENGUAJE DE PROGRAMACION BASIC.	1
1.1.3	PAQUETE INTEGRADO FRAMEWORK.	1
1.1.4	PAQUETE INTEGRADO SYMPHONY.	1
1.2	RESUMEN GENERAL.	2
1.2.1	CAPITULO I.	2
1.2.2	CAPITULO II.	2
1.2.3	CAPITULO III.	2
1.2.4	APENDICE A.	2
1.2.5	BIBLIOGRAFIA.	2
2	CAPITULO I. GENERALIDADES.	3
2.1	COMPUTADORAS Y MICROCOMPUTADORAS.	3
2.1.1	HISTORIA DE LAS COMPUTADORAS.	3
2.1.1.1	ANTECEDENTES DE LAS COMPUTADORAS.	3
2.1.1.2	PRIMERA GENERACION (1949-1959).	3
2.1.1.3	SEGUNDA GENERACION (1959-1965).	4
2.1.1.4	TERCERA GENERACION (1965-1970).	5
2.1.1.5	CUARTA GENERACION (1970-1977).	5
2.1.2	HISTORIA DE LAS MICROCOMPUTADORAS.	6
2.1.2.1	SISTEMAS DE MICROCOMPUTADORAS.	6
2.1.2.2	DESARROLLO DE MICROCOMPUTADORAS.	7
2.1.2.3	PRIMERA MICROCOMPUTADORA PERSONAL.	7
2.1.2.4	CARACTERISTICAS PRINCIPALES.	7
2.1.2.5	SOFTWARE DE MICROCOMPUTADORAS.	8
2.1.2.6	MARCAS DE MICROCOMPUTADORAS.	9
2.2	COMUNICACION DE DATOS.	9
2.2.1	COMUNICACION HUMANA.	9
2.2.2	COMUNICACION EN MICROCOMPUTADORAS.	10
2.2.3	VELOCIDAD.	12
2.2.4	SISTEMA DE TRANSMISION DE DATOS.	12
2.2.5	SINCRONISMO.	13
2.2.6	LINEAS DE COMUNICACION.	14
2.2.7	PUERTOS SERIE.	15
2.2.8	EL CODIGO ASCII.	15
2.3	MODEMS.	17
2.3.1	GENERALIDADES.	17
2.3.2	CLASIFICACION DE LOS MODEMS.	18
2.3.3	MODEMS SEMI-INTELIGENTES.	18
2.3.4	MODEMS INTERNOS Y EXTERNOS.	18
2.3.5	MODEMS DE BAJA Y ALTA VELOCIDAD.	19
2.4	PROTOCOLOS DE COMUNICACION.	19
2.4.1	PROTOCOLO.	19
2.4.2	PROTOCOLO XON/XOFF.	19
2.4.3	PROTOCOLO MODEM.	20
2.5	LA INTERFASE RS-232C.	21
2.5.1	DEFINICIONES.	21
2.5.2	SEÑALES INTERNAS Y EXTERNAS.	22
2.5.3	IDENTIFICACION DE PINES.	22
2.5.4	PINES EN UNA MICROCOMPUTADORA.	24
2.6	PAQUETES INTEGRADOS.	25
2.6.1	GENERALIDADES.	25
2.6.2	FRAMEWORK II.	25
2.6.3	SYMPHONY.	28
3	CAPITULO II. INTERCONEXION DIRECTA.	29
3.1	HARDWARE Y CABLES UTILIZADOS.	29
3.1.1	CABLE DE MODEM NULO MODIFICADO.	29
3.1.2	CONECTORES DB-25.	30
3.1.3	CONSTRUCCION DEL CABLE.	30

3.1.4	MICROCOMPUTADORAS UTILIZADAS.	30
3.2	<u>COMUNICACIONES CON BASIC.</u>	31
3.2.1	SOFTWARE UTILIZADO.	31
3.2.2	INTRODUCCION.	31
3.2.2.1	ENTRADA/SALIDA (I/O) EN BASIC.	31
3.2.2.2	FUNCIONES DE ENTRADA/SALIDA.	31
3.2.2.3	FUNCION INPUTS PARA ARCHIVOS COM.	32
3.2.3	PROGRAMA DE COMUNICACIONES EN BASIC.	33
3.2.3.1	CONSIDERACIONES.	33
3.2.4	PROGRAMA TTY DE COMUNICACIONES.	33
3.2.5	DESCRIPCION DEL PROGRAMA TTY.	34
3.2.5.1	ANTECEDENTES.	34
3.2.5.2	DESCRIPCION.	34
3.2.5.3	ERRORES EN COMUNICACIONES.	36
3.2.6	PRUEBAS DEL PROGRAMA.	37
3.2.6.1	TRANSMISION DE TEXTO.	37
3.2.6.2	TRANSMISION DE ARCHIVOS ASCII.	38
3.2.6.3	ARCHIVOS DE PROGRAMAS.	39
3.3	<u>COMUNICACIONES CON FRAMEWORK II.</u>	40
3.3.1	INTRODUCCION.	40
3.3.1.1	SOFTWARE UTILIZADO.	40
3.3.1.2	PREPARACION DEL EQUIPO.	40
3.3.1.3	MARCOS DE TELECOMUNICACIONES.	40
3.3.1.4	CONEXION CON EL FRAMEWORK II.	41
3.3.2	PRIMERA PRUEBA.- TRANSMISION DE TEXTO.	42
3.3.2.1	DESCRIPCION.	42
3.3.2.2	MARCO TELEPC.	42
3.3.3	SEGUNDA PRUEBA.- USO DEL XMODEM.	43
3.3.3.1	DESCRIPCION.	43
3.3.3.2	MARCO TELENV2.	45
3.3.4	TERCERA PRUEBA.- USO DEL BATCH XMODEM.	46
3.3.4.1	DESCRIPCION.	46
3.3.4.2	MARCO TELENV3.	47
3.3.4.3	MARCO TELRES3.	50
3.4	<u>COMUNICACIONES CON SYMPHONY.</u>	53
3.4.1	INTRODUCCION.	53
3.4.1.1	SOFTWARE UTILIZADO.	53
3.4.1.2	PREPARACION DEL EQUIPO.	53
3.4.1.3	ESPECIFICACIONES DE COMUNICACION.	53
3.4.1.4	CONEXION CON EL SYMPHONY.	54
3.4.2	PRIMERA PRUEBA.- TRANSMISION DE TEXTO.	55
3.4.2.1	DESCRIPCION.	55
3.4.3	SEGUNDA PRUEBA.- TRANSMISION XMODEM.	55
3.4.3.1	DESCRIPCION.	55
4	<u>CAPITULO III. CONCLUSIONES.</u>	56
4.1	<u>CONCLUSIONES CON RESPECTO AL HARDWARE.</u>	56
4.1.1	CABLES Y CONECTORES.	56
4.1.2	INTERCONEXION DE EQUIPOS.	56
4.2	<u>CONCLUSIONES CON RESPECTO AL GWBASIC.</u>	57
4.2.1	VENTAJAS.	57
4.2.2	DESVENTAJAS.	57
4.3	<u>CONCLUSIONES CON RESPECTO AL FRAMEWORK.</u>	57
4.3.1	VENTAJAS.	57
4.3.2	DESVENTAJAS.	58
4.4	<u>CONCLUSIONES CON RESPECTO AL SYMPHONY.</u>	58
4.4.1	VENTAJAS.	58
4.4.2	DESVENTAJAS.	58
4.5	<u>CONCLUSIONES FINALES.</u>	58
4.5.1	LA MEJOR OPCION DE COMUNICACIONES.	58
4.5.2	OTROS PAQUETES DE COMUNICACIONES.	59

5	APENDICE A. GLOSARIO Y TERMINOLOGIA.	60
	5.1 <u>GLOSARIO Y TERMINOS USADOS.</u>	60
6	BIBLIOGRAFIA.	69
	6.1 <u>BIBLIOGRAFIA.</u>	69

1 INTRODUCCION.

1.1 OBJETIVOS DEL TRABAJO.

1.1.1 OBJETIVO PRINCIPAL.

El presente trabajo describe la interconexión de dos Microcomputadoras Personales (PC) con el objeto de probar y comparar las opciones de transmisión de datos a través del puerto serie de Comunicaciones de la Interfase RS-232C.

Los Lenguajes y Paquetes probados fueron los siguientes:

- a) Lenguaje de Programación GW-BASIC.
- b) Paquete Integrado FRAMEWORK II.
- c) Paquete Integrado SYMPHONY.

1.1.2 LENGUAJE DE PROGRAMACION BASIC.

La primera etapa de este trabajo pretende demostrar que cualquier usuario con conocimientos básicos en microcomputadoras puede llevar a cabo pruebas de comunicaciones, ya que además del puerto serie (interfase RS-232C) sólo se requiere un programa sencillo escrito en GW-BASIC (o BASICA). En estas pruebas no influye la memoria de las microcomputadoras, ya que se puede realizar en dos equipos con 256 Kb de memoria.

Esta primera etapa consistió en probar la transmisión de datos mediante un programa escrito en el lenguaje de programación GW-BASIC (Versión 3.2).

Las pruebas realizadas en esta primera fase fueron tres:

- a) Transmisión de mensajes.
- b) Transmisión de archivos ASCII.
- c) Transmisión de archivos de programas.

1.1.3 PAQUETE INTEGRADO FRAMEWORK.

Si el usuario dispone de dos microcomputadoras con al menos 384 Kb de memoria (aunque son recomendables 512 Kb) con sus respectivos puertos serie de comunicaciones (interfase RS-232C) y también dispone del paquete integrado Framework II, puede pasar a la segunda etapa, la cual abarcó las siguientes pruebas:

- a) Transmisión de mensajes entre ambas microcomputadoras.
- b) Transmisión de un archivo utilizando el protocolo XMODEM.
- c) Transmisión de un conjunto de archivos con el Protocolo Batch XMODEM, agrupados bajo una extensión común (por ejemplo *.FW2, *.WK1, etc.).

1.1.4 PAQUETE INTEGRADO SYMPHONY.

Si el usuario dispone de dos microcomputadoras con al menos 384 Kb de memoria (aunque son recomendables 512 Kb) con sus respectivos puertos serie de comunicaciones (interfase RS-232C) y

dispone del paquete integrado Symphony, puede pasar a la tercera etapa, la cual consistió en las siguientes pruebas:

- a) Transmisión de mensajes entre ambas microcomputadoras.
- b) Transmisión de un archivo utilizando el protocolo XMODEM.

1.2 RESUMEN GENERAL.

1.2.1 CAPITULO I.

En el Capítulo I se incluyen generalidades sobre Microcomputadoras Personales, Líneas de Transmisión de Datos (tanto Directas como Telefónicas), Equipos de Comunicaciones (Módems), Protocolos de Comunicación (XON/XOFF, XMODEM, Batch XMODEM, etc.) y sobre Paquetes Integrados (FRAMEWORK, SYMPHONY).

1.2.2 CAPITULO II.

En el capítulo II se indica la manera de interconectar las dos microcomputadoras por cable directo (Cable de Módem Nulo Modificado).

También se explican las pruebas efectuadas con un programa en GW-BASIC, así como la prueba de algunas opciones de Telecomunicaciones de los paquetes integrados Framework II y Symphony.

1.2.3 CAPITULO III.

Las conclusiones a que se llegó con el presente trabajo se indican en el capítulo III, así como algunos campos donde se pueden aplicar las pruebas realizadas.

Asimismo se dan algunas ideas sobre pruebas adicionales no cubiertas en este trabajo en diferentes campos de la Transmisión de Información entre Microcomputadoras Personales.

1.2.4 APENDICE A.

En este apéndice se incluye un glosario de los términos más usuales empleados en la Transmisión de Datos mediante Microcomputadoras Personales.

1.2.5 BIBLIOGRAFIA.

Finalmente se incluye una Bibliografía que puede ser consultada por los interesados en el tema de la Transmisión de Datos entre Microcomputadoras Personales.

2 CAPITULO I. GENERALIDADES.

2.1 COMPUTADORAS Y MICROCOMPUTADORAS.

2.1.1 HISTORIA DE LAS COMPUTADORAS.

2.1.1.1 ANTECEDENTES DE LAS COMPUTADORAS.

Las computadoras son dispositivos electrónico-mecánicos capaces de procesar automáticamente la información representada por impulsos eléctricos, mediante un programa almacenado. Generalmente se clasifican, según el tipo de datos que procesan, en **analógicas, digitales e híbridas**. En las primeras se maneja la información en forma de cantidades continuas que se miden físicamente; los datos que se obtienen por medición son datos continuos. Las **computadoras digitales** por su parte manejan la información en forma de estados físicos, los cuales se representan mediante un código especial denominado código binario; los datos que procesan estas máquinas se llaman datos discretos y se obtienen por conteo. Cuando se combinan las características de las computadoras analógicas y las digitales resultan los **sistemas híbridos**. Por ejemplo, en la unidad de cuidados intensivos de un hospital, los dispositivos analógicos pueden medir el funcionamiento del corazón del paciente, la temperatura y otros signos vitales. Estas mediciones pueden ser convertidas en números y enviadas a un componente digital del sistema, el cual puede controlar los signos vitales y enviar una señal al puesto de enfermeras si se detectan lecturas anormales.

En 1937, Howard G. Aiken, de la Universidad de Harvard, desarrolló algunas ideas que comprendían cálculos mecánicos y que vendió a la IBM. En mayo de 1944 se puso en servicio una calculadora automática controlada en secuencia, llamada **Harvard Mark I** y diseñada por Aiken. Tenía contadores mecánicos movidos por embragues electromagnéticos controlados con relevadores electromecánicos. A esta máquina se le ha llamado "cerebro mecánico" y suma, resta, multiplica, divide y compara cantidades. Posteriormente Aiken construyó los modelos II, III y IV.

En 1945, Mauchly y Prosper Eckert construyeron una gran computadora para fines generales llamada **ENIAC** (integradora y calculadora electrónumerica). Fue la primera computadora totalmente electrónica y se consideraba bastante rápida para cálculos prolongados. Esta máquina funcionó hasta el 2 de octubre de 1955.

2.1.1.2 PRIMERA GENERACION (1949-1959).

Las computadoras de la primera generación fueron de tubos al vacío y su época abarcó de 1949 hasta 1959, aproximadamente. En 1947, Eckert y Mauchly comenzaron a perfeccionar la **UNIVAC** (Computadora Automática Universal), que entregaron a la Oficina de Censos el 14 de junio de 1951; esta máquina se utilizó constantemente 12 horas diarias durante 12 años.

En Inglaterra, la máquina **MADAM** (máquina digital automática de Manchester) fue construida en la Universidad de Manchester por Ferranti, Limited, mientras que en la Universidad de Londres se perfeccionaron las máquinas **SEC** (computadora electrónica sencilla) y **APEC** (computadora electrónica para fines generales).

Otros fabricantes construyeron computadoras de la primera generación. Así, la IBM construyó la serie Mark a mediados de la década de 1940. presentó la perforadora y calculadora electrónica 604 en 1948 y los modelos 702, 704, 705 y 709 de computadoras comerciales a mediados de la década de 1950 y que se vendieron hasta 1958. En 1953 vendió más de 1000 computadoras IBM 650 de tambor magnético, una de las cuales fue la primera computadora que existió en México en junio de 1958 y que se instaló en el Centro Electrónico de Cálculo de la UNAM. Los lenguajes que utilizaba esta computadora eran el lenguaje-máquina, los ensambladores SOAP II y III, el intérprete BELL-2 y el compilador RUNCIBLE. Tenía una lectora y una perforadora de tarjetas, una memoria de 20,000 caracteres y tableros para leer y perforar los datos en diferentes columnas.

En 1957, la Burroughs Corporation construyó una computadora de tambor magnético, la E101. En ese mismo año Honeywell presentó la Datamatic 1000. La RCA Fue el primer fabricante de una memoria de núcleo magnético que incorporó en su computadora BIZMAC, si bien resultó anticuada cuando se entregó en 1956.

Comparadas con las computadoras posteriores, las de primera generación eran voluminosas, inflexibles y muy sensibles a los cambios de temperatura, requiriendo instalaciones especiales de aire acondicionado. Sin embargo, tenían grandes ventajas sobre sus predecesoras las máquinas electromecánicas. Eran de mayor velocidad debido al uso de tubos al vacío y fueron los primeros sistemas prácticos que permitían una programación de tipo interno. Esta característica permitió hacer comparaciones y tomar "decisiones lógicas".

2.1.1.3 SEGUNDA GENERACION (1959-1965).

Con la sustitución de los bulbos electrónicos mediante transistores, surgió la segunda generación de computadoras, las cuales ya eran más compactas, con más memoria, tiempos de acceso de microsegundos y mecanismos para detección y corrección de errores. Los requerimientos de aire acondicionado fueron menos estrictos y se mejoró el equipo periférico, con lectoras, perforadoras e impresoras de alta velocidad.

En 1959, la IBM produjo las series 7070, 7080 y 7090; en 1960 las series 1400 y 1600 y en 1962 las series 7040 y 7044. Un ejemplo de esta segunda generación lo constituye la IBM 1620 instalada en la Facultad de Química de la UNAM (aproximadamente en el año de 1967) y que ya utilizaba el compilador FORTRAN.

La Minneapolis-Honeywell presento su modelo 800 y en 1963 la Burroughs presentó su modelo 5000, más tarde modificado para ser el B5500. De este modelo se instaló una computadora en el entonces Centro de Investigación en Matemáticas Aplicadas, Sistemas y Servicios (CIMASS) de la UNAM. La NCR produjo su modelo 304, en tanto que la RCA presentó el modelo 501. En 1957, la Philco construyó su modelo 2000, computadora binaria de alta velocidad. La Univac presentó al inicio de la década de 1960 el modelo UNIVAC III y la 1107 de película delgada en 1962. La CDC presentó su modelo 1604 en 1962 y el 600 en 1963; casi todas las computadoras

en gran escala instaladas hasta mediados de la década de 1960, eran CDC-6600 y en 1968 apareció la CDC-7600.

2.1.1.4 TERCERA GENERACION (1965-1970).

La transición de la segunda a la tercera generación no es muy clara. Se consideran de tercera generación casi todas las computadoras posteriores a 1965. Sus principales características son los circuitos monolíticos integrados, terminales de tiempo compartido, multiprogramación (ejecución en paralelo de varios programas), multiprocesamiento, procesamiento en tiempo real y mayor miniaturización del equipo. También se utilizan más eficazmente los mecanismos de entrada/salida y los de acceso aleatorio (memorias de disco). Igualmente, los equipos de comunicación de datos facilitan la transmisión de información con estaciones remotas. Utilizan ampliamente el rastreo óptico y el reconocimiento de caracteres escritos en tinta magnética (MICR) y las velocidades de acceso se miden en nanosegundos.

Se cree que la IBM 360 fue la primera de las computadoras basadas en el uso de la tecnología lógica sólida (SLT); su unidad aritmética efectúa cálculos en milmillonésimas de segundo. Por su parte la RCA presentó su serie Spectra 70. En diciembre de 1963, Minneapolis-Honeywell produjo la Honeywell 200 y su serie completa desde la 100 hasta la 1200. Otras computadoras de la tercera generación incluyen las Burroughs 6500, 7500 y 8500, de las cuales la UNAM todavía poseía la B7800 en 1989. También se incluyen la serie Univac 9000, la serie Century de NCR (1968) y la serie 3000 de la CDC.

2.1.1.5 CUARTA GENERACION (1970-1977).

Se considera que a fines de 1970 surgió la cuarta generación de computadoras, con sutiles y profundos cambios para el proceso de datos. Desde el punto de vista del diseño, la cuarta generación ofrece al usuario una mayor capacidad en operaciones de entrada/salida, separando estas funciones de las de procesamiento. También ofrece mayor duración de los componentes y mayor confiabilidad en los sistemas.

Desde un punto de vista funcional, se tienen nuevos y poderosos lenguajes que ensanchan el empleo de la multiprogramación y del procesamiento múltiple, así como un cambio importante del proceso por lotes al procesamiento en línea remoto e interactivo.

Desde un punto de vista administrativo, las computadoras de la cuarta generación facilitan y hacen más efectivo el trabajo gerencial, con lenguajes orientados a la administración con cadenas de sistemas compatibles en localidades centrales, divisionales e individuales de los usuarios, conectados por terminales remotas capaces de manipular datos y voz que facilita la comunicación hombre-máquina. La adquisición y retroalimentación de información con grandes bancos de datos y archivos centrales dan a la administración acceso inmediato con ahorro de tiempo, cambios de hábitos de trabajo, reducción entre los niveles administrativos, costos más bajos para el usuario, reemplazo paulatino de los archivos tradicionales y cambios en la toma de decisiones. En esta generación se hacen más comunicaciones que proceso de datos.

La serie IBM de la cuarta generación se caracteriza por un concepto microológico variable, usado para dar instrucciones y para ejecución en el procesamiento con memorias amortiguadoras de alta velocidad, mayor facilidad en el diagnóstico de fallas y la disponibilidad de capacidades interconstruidas de comunicación entre el hombre y las máquinas. La micrológica variable es única en la forma en que altera una serie de instrucciones mediante la programación, ya que puede alterarse para producir resultados diferentes con los mismos datos disponibles.

Con respecto al tamaño de la memoria interna, varía de 32 Kb a 2 Mb, aunque puede ser mayor. Se compone de núcleos de ferrita y de una memoria amortiguadora MOS (circuito integrado semiconductor de óxido metálico), en donde almacena instrucciones y datos. Durante el procesamiento la computadora funciona a mayor velocidad con la memoria amortiguadora.

Además de la disponibilidad de terminales remotas para aplicaciones de tiempo compartido y conversacional que interactúan con grandes bases de datos, otras características de la cuarta generación incluyen: mayor utilización de la multiprogramación (operación coincidente de dos o más programas), mayor disponibilidad del sistema (reducción de las interrupciones), almacenamiento compartido de la memoria, operación en varios modos de la unidad central de proceso (CPU) y la mayor utilización de programas y paquetes que aumentan la eficacia del sistema.

2.1.2 HISTORIA DE LAS MICROCOMPUTADORAS.

2.1.2.1 SISTEMAS DE MICROCOMPUTADORAS.

Una microcomputadora es el sistema más pequeño de propósito general capaz de ejecutar instrucciones de un programa almacenado para efectuar una amplia variedad de tareas. Un sistema de microcomputadora tiene todos los elementos funcionales que se encuentran en cualquier sistema grande. Esto es, está organizado con las siguientes unidades: de almacenamiento, aritmético-lógica, de control y la de entrada/salida. Aunque las UCP completas de algunas microcomputadoras se pueden empaçar en una sola pastilla (chip) de silicio, como las de 8 bits, la mayoría son más grandes y emplean varias pastillas (chips). Un chip de microprocesador, por ejemplo, lleva a cabo las funciones de lógica, de aritmética y de control. Varios chips de RAM (memoria de acceso aleatorio) están disponibles para ser utilizados por el operador conforme se van necesitando para manejar las funciones primarias de almacenamiento. Además se utilizan chips adicionales de ROM (memoria sólo de lectura) para almacenar en forma permanente datos o instrucciones programados con anticipación.

La mayoría de las microcomputadoras son unidades tan compactas y ligeras, que pueden ser trasladadas con facilidad. Están diseñadas para ser utilizadas por una sola persona. Además de la UCP, la microcomputadora común tiene un teclado para que el operador introduzca la información. Unidades lectoras/grabadoras de discos flexibles o cintas se usan para introducir datos y programas y para recibir la salida procesada. Los discos rígidos y los discos compactos (CD-ROM) proporcionan gran cantidad de almacenamiento secundario fuera de línea. Una pantalla de despliegue vi-

sual (monocromática o en color), una impresora de caracteres (o LASER) y graficadores (PLOTTERS), se utilizan para preparar la salida en forma legible para los humanos.

2.1.2.2 DESARROLLO DE MICROCOMPUTADORAS.

Antes de 1968 y durante varios años, Victor Poor, ingeniero electrónico al servicio de Datapoint Corporation, había estado trabajando en el diseño de computadoras de propósito especial. Cada vez que se necesitaba un dispositivo especial, comenzaban el diseño desde el principio, lo que a Poor y colaboradores les parecía un gran desperdicio de tiempo. En vez de eso, pensó, si pudieran colocar en una sola pastilla de silicio los elementos básicos de control y de lógica aritmética, la pastilla podría producirse en forma masiva y entonces se le programaría de diferentes formas para llevar a cabo las tareas especiales para las cuales sería utilizada.

Así fue como en 1969 Poor y Harry Pyle desarrollaron un modelo de microprocesador en una sola pastilla y lo presentaron a dos fabricantes de componentes: Texas Instruments e Intel Corporation. No se decidió nada en esas fechas, pero a principios del decenio de 1970, un grupo de ingenieros de Intel construyó el primer microprocesador para un fabricante japonés de calculadoras de escritorio. Este primer microprocesador en una sola pastilla, el Intel 4004, tenía un número limitado de instrucciones y sólo podía manejar "palabras" de 4 bits de datos al mismo tiempo.

2.1.2.3 PRIMERA MICROCOMPUTADORA PERSONAL.

No pasó mucho tiempo, sin embargo, antes de que los ingenieros de Intel y de otras compañías produjeran microprocesadores capaces de operar con 8 bits. En 1974 este avance motivó a su vez la introducción de microsistemas individuales para los aficionados a los pasatiempos.

En el mes de enero de 1975 fue anunciada en la revista Popular Electronics la que se considera como la primera microcomputadora, la ALTAIR 8800 inventada por Ed Roberts. La ALTAIR original era una caja con interruptores y luces en el frente. Dentro de la caja había un tablero con un microprocesador INTEL 8080 de 8 bits y 256 bytes de memoria (0.25 Kb). Se vendía en 397 dlls. y la compañía que la fabricaba era la MITS. La estructura interna se conoció como el bus ALTAIR, conocido posteriormente como bus S-100. Sólo se podía programar en lenguaje-máquina y únicamente aceptaba instrucciones en números binarios, los cuales se introducían a la minicomputadora (el término microcomputadora todavía no se inventaba) por medio de los interruptores. No tenía unidades de entrada (sólo interruptores), ni de salida (sólo luces). No obstante se vendieron 800 en un año.

2.1.2.4 CARACTERISTICAS PRINCIPALES.

En general, todas las UCP de las microcomputadoras utilizan el enfoque de almacenamiento en bytes direccionables. Actualmente, la mayoría de las microcomputadoras existentes están construidas con base a unos pocos microprocesadores populares, de 8

y 16 bits: Zilog Z80; Mos Technology 6502; Intel 8080, 8088, 80286, 80386 y 80486; Motorola 6809; Texas Instruments 9900. Los chips de 8 bits pueden manejar solamente un byte de 8 bits al mismo tiempo, mientras que los de 16 bits manipulan 2 bytes. Dado que las computadoras grandes pueden consultar, manejar y almacenar dos, cuatro y hasta ocho bytes como una sola unidad, es fácil entender por qué las microcomputadoras son relativamente lentas. Sin embargo, los microprocesadores de 16 bits y el uso de tecnologías avanzadas (AT), han mejorado el desempeño y velocidad de las microcomputadoras. Además, algunos fabricantes de componentes como Intel, Hewlett-Packard y National Semiconductor, han diseñado microprocesadores de 32 bits, que pueden manejar 4 bytes al mismo tiempo. El diseño de Hewlett-Packard comprime 450 mil transistores en una pastilla del tamaño de la letra mayúscula "M".

Con respecto al almacenamiento principal en RAM, las microcomputadoras más pequeñas con microprocesadores Z80, 6502 o TI-9900, de baja velocidad (hasta 4 MHz), principian en 4096 bytes (o 4 Kb, donde 1 Kb = 1024 bytes, es decir 2^{10}) y llegan hasta 128 Kb. Las de tipo intermedio con microprocesador 8088 de 4.77-10 MHz van desde 128 Kb hasta 512 Kb. Las de tipo AT, con microprocesador 80286 de 12-16 MHz varían entre 640 Kb y 4 Megabytes de memoria. Finalmente las que incluyen microprocesadores 80386 de 25 MHz y 80486 de 33 MHz principian en 4 Megabytes en su memoria principal, llegando a 32 Mb. A menudo el procesador del sistema y los programas de control se almacenan permanentemente en chips de ROM.

Muchos proveedores ofrecen una amplia variedad de dispositivos periféricos o "adicionales" para microcomputadoras. Sin embargo, una determinada microcomputadora no puede soportar los dispositivos de E/S de una computadora mayor. Un "bus de interfase" del sistema es un dispositivo que sirve como conexión eléctrica entre la UCP y los diferentes periféricos. Muchos fabricantes de computadoras personales han adoptado como estándar el bus de interfase S-100 que fue utilizado inicialmente en la computadora ALTAIR. Ensambalar una microcomputadora a partir de diferentes proveedores puede resultar problemático, de aquí la amplia variedad de marcas de microcomputadoras.

2.1.2.5 SOFTWARE DE MICROCOMPUTADORAS.

Con respecto al software, los programas de aplicación de las microcomputadoras se escriben normalmente en un lenguaje de programación de alto nivel. El más popular es el BASIC, si bien se usan también el PASCAL, FORTRAN, COBOL y el lenguaje C.

Existe una amplia gama de sistemas operativos como el CPM, MS-DOS, DR-DOS, UNIX, PIC, etc., aunque el más utilizado por su facilidad es el MS-DOS, que ha llegado a las versiones 4.01 y 5.0 orientadas a ventanas y menús, y con una potente interfase de denominada "SHELL" que facilita su aprendizaje y ejecución.

Los usuarios de microcomputadoras también tienen acceso a una amplia variedad de programas unitarios de aplicaciones como los procesadores de texto, hojas de cálculo, telecomunicaciones, bases de datos, graficación, procesadores de ideas, o bien a paquetes integrados como el Symphony y el Framework que ofrecen en un sólo paquete todas las aplicaciones anteriores. La variedad de

software ofrecido por los proveedores es actualmente casi interminable.

2.1.2.6 MARCAS DE MICROMPUTADORAS.

Desde la aparición de la primera microcomputadora, han surgido infinidad de computadoras hogareñas, personales y para pequeñas empresas, algunas ya desaparecidas. Entre las más representativas utilizadas por los individuos para diversos usos, se encuentran los diferentes modelos II y III de Apple; los modelos 400 y 800 de ATARI; las líneas PET, CBM, VIC, 16, 64 y AMIGA de Commodore; la H-89 de Heath; los sistemas personales de IBM; la serie Challenger de Ohio Scientific; los diferentes modelos TRS de Radio Shack; los modelos 99-4, 99-4A y Profesional de Texas Instruments.

En relación con México, generalmente las marcas japonesas, estadounidenses y algunas europeas, son las más utilizadas, aunque algunas ya son obsoletas. Así, existen modelos de Apple (II, II+, IIe y IIc); de IBM (PC Junior, PC, PS-2, etc.); de Texas Instruments (TI-99/4, 99/4A, Profesional), de Printaform (de Compañías tales como Columbia, Multitech y Acer en una gran variedad de modelos); de Acer (II+, III+, 910 y 915-P y V, etc.); de Philips y de otras Compañías cuyos modelos sería muy largo enumerar.

2.2 COMUNICACION DE DATOS.

2.2.1 COMUNICACION HUMANA.

Antes de aparecer las primeras computadoras, ya existían muchos medios de comunicación, con la siguiente clasificación:

1.- **Comunicación unidireccional.** Es aquella en la cual no se puede conversar con el productor de la información. Aquí se incluyen los libros, revistas y periódicos, discos y cassetes de audio, videocassetes y videodiscos, TV y TV por cable, cine y teatro, radio, propaganda y finalmente carteles y señales.

2.- **Comunicación bidireccional.** Esta es cuando se permite iniciar una conversación y dialogar con una segunda parte. Pertenecen a esta clase el correo, teléfono, Télex y el telégrafo.

3.- **Comunicación multidireccional.** Es aquella que permite interactuar con los miembros de un grupo. Abarca las reuniones, mesas redondas, videoconferencias, banda de radio civil o ciudadana (CB) y a los radioaficionados.

Se puede notar que la mayoría de estos medios de comunicación son electrónicos.

Cuando surgió la revolución de las computadoras apareció un nuevo medio de comunicación, que ni siquiera tiene un nombre. La palabra más usada es Videotex, pero se utilizan otros términos para describirlo como los siguientes: Red de computadoras, comunicación de datos, comunicación de microcomputadoras personales, correo electrónico, publicación electrónica, base de datos "on-line", utilidad de información y Viewdata.

Por lo anterior se ve que no existe una palabra que describa el fenómeno entero, ya que existen muchos aspectos diferentes del mismo.

2.2.2 COMUNICACION EN MICROCOMPUTADORAS.

En la actualidad uno de los medios de comunicación más usados es el teléfono, con el cual se tiene acceso a la red telefónica de todo el mundo. En la comunicación entre computadoras, la **Microcomputadora Personal (PC)** es el equivalente al teléfono. La PC es el instrumento para acceder una red internacional de computadoras y a grandes bancos de información. Una PC puede llamar a otra microcomputadora y pueden intercambiar datos.

Los primeros sistemas de comunicación de computadoras implicaban terminales conectados a grandes computadoras principales o a minicomputadoras. Al principio estas terminales eran impresoras con teclados integrados, como el Teletipo Modelo 33, que todavía se usa. Pronto se introdujeron terminales más silenciosas, seguidas por terminales de video. De esta forma las terminales mudas no eran otra cosa que pantallas de video con teclado integrado, sobre las cuales se imprimían las líneas de texto de una en una que desaparecían por arriba. Más tarde aparecieron las terminales inteligentes que aceptaban órdenes de borrado de pantalla, inserción o eliminación de texto, posicionamiento de texto en la pantalla y de protección de áreas de la misma contra modificaciones accidentales. Estos dos tipos de terminales no hacían nada útil, a menos que estuvieran conectadas a una computadora.

La invención de la **Microcomputadora Personal** dejó obsoletas a muchas terminales de video. La PC podía hacer lo mismo que las terminales y mucho más. Además de trabajar mejor, ejecuta casi cualquier tipo de programas de aplicación en forma independiente y en la actualidad han llegado a ser muy poderosas, a veces más que algunas computadoras principales a las que se pueden conectar.

LA COMUNICACION ASINCRONA.

La comunicación asincrónica es el enlace entre una microcomputadora personal y el mundo exterior, y es un descendiente directo del telégrafo. Inclusive se puede conectar directamente un Teletipo Modelo 33 a la PC. Conceptualmente, el enlace asincrónico son dos cables, uno en cada dirección, por los cuales se envían caracteres secuencialmente, uno a la vez. En la siguiente figura se muestra la representación del carácter "B", tal como aparece en un osciloscopio, y su equivalencia en el código ASCII.

telefónica. En cambio cuando se utiliza un módem externo o se conectan dos PC's, sí se requiere un cable adecuado. Para las microcomputadoras IBM o Compatibles, sólo existen dos tipos de cables: El cable de módem y el cable de módem nulo. Estos cables tienen en cada extremo un conector estándar de 25 pines ("patas"), llamado conector DB-25, aunque sólo se utilizan realmente los pines 2 al 9, por lo que se puede usar un conector de 9 pines. Posteriormente se detallan estos cables.

2.2.3 VELOCIDAD.

Cuando ya se tenían bastantes conocimientos sobre comunicaciones analógicas y digitales, surgió la comunicación de datos entre computadoras. Esta es el movimiento de la información codificada de alguna manera desde un punto hacia otro punto, por medio de líneas y sistemas de transmisión eléctrica, electrónica u óptica.

Una de las principales consideraciones que hay que tomar en cuenta en la comunicación de datos, es la velocidad con la que se transmite la información, existiendo las siguientes clases:

1.- La velocidad de modulación se usa para fijar las características de la línea de transmisión y se puede definir como el número máximo de veces por segundo que puede cambiar de estado la señalización en línea. Se mide en baudios, siendo un baudio equivalente a un intervalo significativo por segundo.

2.- Por otra parte, la velocidad de transmisión en serie se define como el número máximo de bits (dígitos binarios) que se transmiten por segundo al través de un determinado circuito de datos. Se mide en bits por segundo. Si la información es tal que a cada estado significativo en línea corresponde un bit de información y se usa un código con bits de igual longitud, el número de bits/seg coincide con el de baudios.

3.- La velocidad de transferencia de datos representa la cantidad de información que se puede transmitir por unidad de tiempo. También se define como el promedio de bits, caracteres o bloques por unidad de tiempo que pasan entre dos equipos que pertenecen a un sistema de transmisión de datos.

Una interfase serie asíncrona opera a una velocidad fija, y ambos extremos de la línea deben estar a la misma velocidad. En general, para obtener la velocidad en caracteres por segundo basta con dividir entre 10 la velocidad en bits por segundo. Así, una velocidad de 300 bps es equivalente a transmitir 30 caracteres por segundo, lo cual es útil conocer cuando se transfieren archivos y se requiere calcular el tiempo de transmisión. La microcomputadora personal admite un amplio rango de velocidades, pero las más usuales son 300, 1200 y 9600 bps.

2.2.4 SISTEMA DE TRANSMISION DE DATOS.

La transmisión de datos es el movimiento de información que ya se procesó o que se va a procesar, codificada de alguna forma (generalmente binaria), en algún sistema de transmisión eléctrica

o electrónica. Los medios empleados varían en función de la distancia sobre la cual se va a transmitir la información.

Los elementos que integran un sistema de transmisión de datos, son los que se indican a continuación.

1.- **Equipo Terminal de Datos (DTE).** Este equipo cumple la función de ser origen o destino de los datos y además controla la comunicación. Se utilizan por lo menos dos, uno para enviar y otro para recibir información y viceversa. Puede ser una computadora grande (Mainframe) y una microcomputadora, o bien dos microcomputadoras personales.

2.- **Equipo de Comunicación de Datos (DCE).** Es el equipo de terminación del circuito de datos o Módem, y tiene por objeto transformar las señales portadoras de la información a transmitir (utilizadas por los DTE's) en otras que sean susceptibles de ser enviadas hasta otro DTE remoto, utilizando los medios de comunicaciones clásicos. Generalmente el DCE agrega alguna información adicional y sólo se dedica a transmitir los datos.

3.- **Líneas o medios de transmisión.** Es el conjunto de medios de transmisión que une a dos DCE's. Su constitución depende de algunos factores como la distancia, velocidad, etc.. Deben cumplir con determinadas especificaciones y siempre se apoyan en la infraestructura de comunicaciones disponibles.

4.- **Enlace de Datos (ED).** Es la unión entre la fuente y el colector de datos y está formado por los controladores DCE y la línea.

5.- **Circuito de Datos, (CD).** Es el conjunto formado por los módems (DCE's) y la línea y su misión consiste en entregar en la interfase con el DTE colector las señales que recibió en la interfase del DTE origen, en la misma forma y con idéntica información.

De acuerdo al medio de transmisión, se puede tener en serie o en paralelo. La primera es cuando los datos son transmitidos bit a bit utilizando un único canal; generalmente se usa para enviar datos a largas distancias. La segunda es cuando se transmiten simultáneamente todos los bits de un carácter o de una palabra, para lo cual el medio de transmisión debe tener tantos canales o conductores, como bits contenga el elemento base. Esto aumenta la complejidad del medio, pero aumenta la velocidad de transmisión.

2.2.5 SINCRONISMO.

Dentro del circuito de datos puede haber dos tipos de transmisión:

1.- **Transmisión Asíncrona o de Arranque/Parada (Start/Stop).** Se basa en la existencia dentro del equipo receptor de un tiempo teóricamente igual al que existe en el transmisor o emisor. La señal que se transmite es cuando los bits que forman la palabra-código correspondiente van precedidos de un bit de arranque (Start) y terminados de uno o dos bits de parada (Stop). Se emplea en velocidades bajas e implica equipos más baratos, si bien

utiliza menos eficientemente la línea. Esto significa que se transmite un carácter a la vez y las computadoras deben estar equipadas con un puerto (adaptador de comunicaciones) asíncrono serie y un módem asíncrono, un aparato que conecta la PC a la línea telefónica.

2.- Transmisión Síncrona. Es cuando los datos se transmiten del emisor al receptor con tiempos fijos y constantes, marcados por una base de tiempos común para todos los elementos que intervienen en la transmisión. Los caracteres se transmiten en bloques usando procedimientos complejos. Se precisa un puerto síncrono serie y módems más complejos, pero mejora la utilización de la línea telefónica y permite mayores velocidades, además de ser menos sensible al ruido.

En cualquier forma de las anteriores en que se transmitan los datos, es preciso que ambas microcomputadoras tengan una misma base de tiempo para dar el mismo valor a cada instante. Esto se conoce como sincronización y se debe hacer a tres niveles, en toda transmisión de datos:

a) Sincronismo de bit. Este sirve para determinar el instante en que se comienza a detectar un bit, al menos teóricamente.

b) Sincronismo de carácter. Este se utiliza para que el equipo receptor conozca cuantos bits corresponden a un carácter, generalmente 7 u 8, así como otras características de la información.

c) Sincronismo de mensaje o de bloque. Con éste se define el conjunto de caracteres que constituye la unidad base que forma parte del protocolo de comunicaciones.

2.2.6 LINEAS DE COMUNICACION.

Las líneas de comunicación generalmente están constituidas por dos o cuatro hilos. Las primeras son cuando en todo su recorrido o parte de él se utiliza un mismo circuito físico (2 conductores) para transmitir información en ambos sentidos, pero no al mismo tiempo. Generalmente se llama transmisión Half-Duplex.

En las líneas de 4 hilos se utilizan canales independientes en todo su recorrido para cada sentido de transmisión. En ellas se puede enviar información en ambos sentidos al mismo tiempo. En general se llaman Full-Duplex.

Con respecto al tipo de las líneas, pueden existir las siguientes categorías:

1.- Línea automática conmutada. La línea de transmisión es la misma que la que se utiliza para la comunicación telefónica ordinaria y consiste en un conjunto de medios de transmisión encadenados automáticamente al llamar al equipo remoto y mientras se esté en comunicación con él. Es el método de conexión más simple, pero la probabilidad de errores es alta.

2.- Línea con dedicación exclusiva. Es cuando los circuitos de datos se establecen en forma permanente. En este caso primero

se determina la ruta que una ambos equipos de comunicación de datos, luego se fija el tipo de línea, después se seleccionan y finalmente se asignan los medios de transmisión individualmente de forma rígida y permanente. Existen diferentes calidades en este tipo de líneas.

3.- Líneas de alta velocidad. Se utilizan generalmente con velocidades superiores a los 9600 bits/seg y requieren módems especiales.

2.2.7 PUERTOS SERIE.

Uno de los elementos más importantes en la comunicación de datos entre computadoras es el puerto serie. En el caso de que la microcomputadora no lo tenga se le puede incorporar una tarjeta del Asynchronous Communications Adapter (Adaptador para Comunicaciones Asíncronas) o bien una tarjeta multifuncional conteniendo la circuitería para un puerto serie. La tarjeta se inserta en una ranura de expansión (slot) de la PC, y un conector (generalmente DB-25 o DB-9) permite enchufar un cable para un módem o para otra microcomputadora personal. Los paneles multifuncionales son muy útiles, ya que además del puerto serie incorporan otro puerto serie, un reloj fechador alimentado por batería, un puerto paralelo, un puerto de juegos y ampliaciones de memoria.

2.2.8 EL CODIGO ASCII.

Los archivos de datos y de texto utilizados en las microcomputadoras, corresponden a alguna de las dos siguientes categorías: archivos en lenguaje-máquina, o bien archivos ASCII (American Standard Code for Information Interchange). Ambos están contruidos con números binarios, o sea secuencias de ceros y unos. La diferencia es que los dígitos binarios en un archivo ASCII estándar representan texto y números que la mayoría de la gente puede leer y comprender, mientras que los números binarios de un archivo en lenguaje-máquina representan instrucciones que únicamente la computadora puede leer. Estos sólo tienen sentido cuando son interpretados y desplegados por un programa de aplicación.

Esta diferencia es crucial debido a que un determinado programa de aplicación no puede leer e interpretar archivos en lenguaje-máquina creados por otro programa. Por otro lado, la mayoría de los programas de aplicación pueden leer y comprender los archivos ASCII.

El código ASCII, uno de los pocos denominadores comunes en el mundo de las computadoras, es muy simple en sí mismo. El código asigna un número diferente a todas las letras mayúsculas y minúsculas, a todos los números del 0 al 9, y a cada uno de los símbolos de puntuación (estos números en código ASCII, convertidos a forma binaria, son de lo que se componen los archivos ASCII). El código ASCII estándar va desde 0 hasta 127 (2 elevado a la séptima potencia); el primer carácter reconocible, el signo de cerrar admiración, aparece con el número 33. Los códigos del 0 al 32 son principalmente códigos de control, con Control-A en el 1 y Control-Z en el 26. Estos códigos representan funciones como Retorno (Control-M) o Alimentación de Línea (Control-J). Otros

códigos de control representan Espacio, Retroceso, Alimentación de Formas o Tabulador; otro código hace sonar un bip, etc..

Si todos los programas que almacenan texto en archivos ASCII usaran únicamente los códigos ASCII estándar desde 0 hasta 127, el procedimiento para transferir archivos sería menos complejo. Pero en el sistema binario, sólo se requieren 7 bits para representar estos códigos y todas las computadoras tienen al menos 8 bits con los cuales trabajar. Esto significa que pueden representar números tan grandes como 255 y pueden generar 128 códigos ASCII adicionales.

Los fabricantes de computadoras y de software usan estos códigos adicionales para una variedad de funciones. Como se puede suponer, no se ponen de acuerdo sobre lo que significan estos "códigos altos" (del 128 al 255). En microcomputadoras IBM y Comodore, por ejemplo, los códigos altos representan caracteres gráficos, mientras que en las APPLE los códigos 128 al 255 simplemente repiten los caracteres asignados a los códigos 0 al 127.

Los productores de Software a menudo usan los códigos altos para simbolizar información que es importante a un programa específico. En el WordStar de MicroPro, un código alto marca un "guión fantasma" ("soft hyphen") - aquel que el programa inserta o elimina automáticamente siempre que se ajusta el texto a una postura de margen particular. Otros programas usan los códigos altos para almacenar archivos más compactamente. El BASIC de Microsoft, por ejemplo, almacena automáticamente sus programas en un formato comprimido usando códigos altos para simbolizar instrucciones BASIC.

Los archivos ASCII que utilizan códigos altos son comprendidos únicamente por el programa que los creó. Aunque sólo se coloque texto en ellos, pueden ser tan ilegibles a otros programas como lo son los archivos en lenguaje-máquina. Afortunadamente, los programas que usan los códigos altos ofrecen usualmente una opción de texto puro en código ASCII directo, o sea sin usar los códigos altos, que sí puede ser leído por otros programas.

FORMATO DE LOS ARCHIVOS ASCII.

La manera con la cual un programa o una computadora usan el código ASCII es el primer factor a considerar cuando se desea transferir archivos, y cuando se quiere transferir archivos entre dos programas de procesamiento de palabras, algunas veces es el único factor que se debe considerar.

Pero con otros tipos de programas de aplicación, un segundo e igualmente importante factor es aquel sobre cómo los datos están arreglados, o formateados, dentro del archivo. Por ejemplo, para seguir la pista a un archivo de direcciones de clientes, un programa administrador de bases de datos debe conocer dónde principia y termina cada registro de direcciones, e igualmente donde principia y termina cada nombre, estado, código postal o cualquier otro campo.

El modo más simple para que un programa conozca esta información es usar un carácter ASCII como marcador. Algunos programas de bases de datos marcan el fin de cada registro en un archivo

insertando un caracter de retorno de carro (ASCII 13) o un retorno de carro pareado con un caracter de alimentación de línea (ASCII 10). Cada campo de datos en el registro es típicamente separado de sus alrededores con una coma. Los campos de texto son usualmente enmarcados entre dobles comillas para hacer posible usar las comas como parte de la información de la base de datos.

Los archivos ASCII con este tipo de formato se llaman archivos de "valores separados por comas" (CSV, "comma-separated value"). Algunas veces se llaman también "archivos delimitados", debido a que cada campo está delimitado por un caracter, usualmente una coma. Y debido a que los campos y registros pueden ser de cualquier longitud especificada, se llaman también archivos de "formato libre" o de "registros de longitud variable". El programa simplemente cuenta las comas para saber dónde se encuentra, si dentro de un campo o dentro de un registro.

Otros programas, incluyendo al dBASE II de Ashton-Tate, toman una aproximación diferente para rastrear campos y registros. En lugar de contar las comas, cuentan los caracteres. Para todos los archivos de datos en dBASE, por ejemplo, se deben especificar los nombres y longitudes de cada campo. Así, si se especifican 15 espacios para el campo NOMBRE, el dBASE asigna 15 caracteres espacio (ASCII 32) para ese campo. Ya sea que contenga datos o no, la longitud del campo NOMBRE en todos los registros es siempre de 15 caracteres. Lo mismo es cierto para los demás campos en la base de datos. Por esta razón los archivos con este tipo de formato son llamados archivos de "longitud fija" o de "formato fijo".

Toda esta información sobre archivos ASCII y formato de los datos es muy importante. Con estos fundamentos se puede tener la habilidad para crear un archivo ASCII que sea aceptable para un programa o paquete en particular, una técnica necesaria cuando todo lo demás falla. Aún cuando no sea necesario crear un archivo ASCII, esta información tendrá algunas ventajas sobre otras opciones disponibles, lo cual puede llegar a ser crítico en la Transferencia de Archivos entre Microcomputadoras usando líneas de comunicación de datos.

2.3 MODEMS.

2.3.1 GENERALIDADES.

Además de las líneas telefónicas de comunicación, existen otros sistemas de conexión como los cables directos y coaxiales, rayos infrarrojos, radiofrecuencia, fibras ópticas y comunicación por satélite. Pero los usuarios de microcomputadoras utilizan más los módems para conectarse a la red telefónica pública y acceder otras computadoras.

Un módem es un equipo de comunicación de datos que se conecta a un dispositivo digital (terminal o microcomputadora) y que sirve para convertir la señal digital en una señal analógica, la cual puede ser transmitida por la línea de comunicación hacia otra computadora remota. Su nombre es un acrónimo de los vocablos "MODulador/DEModulador", ya que modula o inserta una información en la señal portadora y la convierte en analógica; en el otro extremo de la línea la señal es demodulada o interpretada

por el otro módem y convertida en digital para ser aceptada por la otra computadora o terminal.

2.3.2 CLASIFICACION DE LOS MODEMS.

Los módems se pueden clasificar de acuerdo a la manera como se modula la señal portadora, existiendo de esta forma modulación de amplitud, de frecuencia y de fase.

La modulación de amplitud (AM) se efectúa cuando el voltaje de pico a pico de la señal portadora varía con la información que se desea transmitir. Esta permite usar el ancho de banda disponible de una línea para transmisión de voz.

La modulación de frecuencia (FM) es la forma más común de modulación con velocidades de transmisión de hasta 1800 bps (bits por segundo). Aquí la señal portadora se modula a diferentes frecuencias. Las frecuencias específicas que se utilizan dependen de los equipos receptor y transmisor utilizados y es menos propensa a errores que la modulación de amplitud.

En la modulación de fase (PM), la fase de la señal portadora varía de acuerdo a los datos que se van a transmitir, y se desplaza un cierto número de grados como respuesta al patrón de bits transmitidos. Esta comienza a reemplazar a las anteriores.

2.3.3 MODEMS SEMI-INTELIGENTES.

Además de modular/demodular datos para transmitirlos, un módem puede tener otras características. Algunos pueden efectuar llamadas a computadoras remotas en forma automática (llamada automática) o se pueden configurar para permanecer en estado de alerta continuo, de modo que puedan ser llamados desde un equipo remoto en cualquier momento (respuesta automática). Otros se pueden utilizar para transmisión de voz o datos alternada o simultáneamente, lo cual es útil para localizar fallas.

Por las características arriba anotadas y otras adicionales, este tipo de módems se conoce como módems semi-inteligentes.

2.3.4 MODEMS INTERNOS Y EXTERNOS.

Los módems también se clasifican en internos y externos. Los módems internos son aquellos que se conectan en una ranura de expansión (slot) dentro de la microcomputadora. Generalmente vienen con su propio Software de comunicaciones y la interfase está contenida dentro de la tarjeta, por lo que únicamente necesitan el conector RJ11 para conectarlos a la línea telefónica. Como ejemplos de éstos se encuentran el Multimódem en sus diferentes modelos (224 PC, 224 EC y PC3), el Micromódem PC Half/Card de ITSA y el FDX/PC de Paradyne.

Los módems externos por otro lado se conectan fuera de la microcomputadora y requieren una tarjeta de interfase serie, generalmente la RS-232C, para comunicarse con ella. Como ejemplos cabe mencionar los modelos Duplex 2 Hilos e ITD-24 de ITSA.

2.3.5 MODEMS DE BAJA Y ALTA VELOCIDAD.

Los módems también se clasifican según su velocidad de transferencia de datos, en módems de baja o alta velocidad. Los primeros operan a velocidades de hasta 1800 bps (bits por segundo). Como ejemplo se pueden mencionar el Micromódem PC (módem interno) y el Duplex 2 Hilos (módem externo), ambos de ITSA.

Los módems de alta velocidad trabajan a velocidades superiores a 1800 bps; generalmente emplean la modulación de fase y en la transmisión emplean una metodología de díbits. Se utilizan para conectar terminales remotas que tienen muchos dispositivos de entrada/salida. Son equipos más costosos que los de baja velocidad. Como ejemplos se puede mencionar el Codex MX 2400, el ITSA ITD 24, el SISCO M 24 D y el PARADINE FDX-2400.

2.4 PROTOCOLOS DE COMUNICACION.

2.4.1 PROTOCOLO.

Un Protocolo de comunicaciones es un conjunto de reglas establecidas para la transmisión de datos entre dos equipos que se comunican entre sí.

Si los datos se transmiten en diferentes tiempos sin ser controlados por un reloj, se denominan protocolos asíncronos. Si los datos aparecen en tiempos determinados y están controlados por un reloj se denominan protocolos síncronos.

Existen protocolos más generales para transferencias de archivos, como los siguientes: XON/XOFF, MODEM7, XMODEM, BATCH XMODEM, YMODEM, TELINK, KERMIT, COMPUSERVE-B, WYXMODEM, etc. A continuación se mencionan los usados en este trabajo.

2.4.2 PROTOCOLO XON/XOFF.

Con este protocolo sólo se pueden enviar archivos ASCII, o sea archivos de texto. Se pueden transmitir mensajes teclados de un sistema emisor a otro receptor, o de una terminal remota a una computadora central.

Parecería difícil pensar en instruir a una computadora remota para que se detenga mientras se transfiere un buffer de memoria a un disco de la microcomputadora receptora que actúa como terminal. La mayoría de los servicios de información y sistemas de tableros de boletines (BBS, o sea Bulletin Board System), utilizan un convenio llamado control de flujo, que actúa como un vigilante de tráfico durante la transmisión de la información. La mayoría de los esquemas comunes de control de flujo utilizan el carácter generado al oprimir las teclas Ctrl-S del teclado como la señal para detener temporalmente el envío de información. Cuando la terminal no tiene la tecla CONTROL se localiza en el manual de los programas de terminal para una tecla equivalente. Todos los programas de terminal tienen alguna manera de enviar códigos de control estándar. La secuencia Ctrl-S es la señal XOFF para detener la transmisión, mientras que la secuencia Ctrl-Q es la señal XON para continuar la transmisión.

Estas combinaciones de teclas permiten detener y reiniciar el flujo de información que corre al través de la pantalla de la terminal receptora. Igualmente, el software de ambas computadoras (sistema remoto y terminal) puede usar automáticamente el XON/XOFF para asegurar que los datos entrantes se detengan si cualquiera de las computadoras está lista para manipular alguna otra tarea por unos momentos.

El protocolo XON/XOFF es generalmente una opción que se puede seleccionar en la mayoría de los programas de terminal comerciales y debe ser inicializada generalmente. Los caracteres XON/XOFF usualmente no son visibles en la pantalla, de manera que pasan desapercibidos para la mayoría de los usuarios.

2.4.3 PROTOCOLO XMODEM.

Ward Christensen, un coautor del primer Sistema de Tablero de Boletines (BBS, Bulletin Board System), desarrolló un programa de dominio público para terminales con sistemas operativos C/PM llamado **MODEM7**. Cuando los usuarios de los BBS de Christensen preguntaron cómo transferir un archivo binario, se instruyó a los BBS para correr un programa de utilería llamado **XMODEM**.

Las reglas fundamentales del **XMODEM** básico establecidas por Christensen desde 1976, son todavía usadas por casi todos los sistemas de boletines (BBS), sin consideración de la computadora a la cual estén conectados.

El protocolo XMODEM, como así se ha llamado, también está disponible para transferencia de archivos en los servicios de información CompuServe y Delphi.

El protocolo **XMODEM** requiere ciertas especificaciones de comunicaciones: palabra de ocho bits de longitud, sin paridad, un bit de inicio y un bit de parada. Algunos programas de terminales colocan estos parámetros automáticamente durante la transferencia con el **XMODEM**, mientras que otros requieren colocarlos manualmente.

El **XMODEM** transfiere archivos enviando y recibéndolos en bloques. Cada bloque consta de los siguientes cinco elementos:

1.- Un caracter especial que marca el principio de encabezado (SOH, Start Of Header).

2.- Un número que identifica el número secuencial del bloque, que se incrementa en uno por cada bloque enviado.

3.- Un número que asegura que el número del bloque esté correcto.

4.- Un segmento de 128 caracteres (o bytes) del archivo que se está transfiriendo.

5.- Por último, un número para verificar que el segmento de datos de 128 bytes recibidos sea igual al enviado. Este número se llama "checksum". Es la suma de los valores hexadecimales de todos los 128 caracteres (bytes).

RECEPCION DE UN ARCHIVO.

Después de notificar al sistema remoto (o emisor) que el sistema receptor desea recibir un cierto archivo con el protocolo XMODEM, el emisor prepara el archivo para la transmisión y luego espera a que el receptor esté listo para principiar la transferencia. Para iniciar ésta, el sistema receptor envía una señal de reconocimiento que consiste en un carácter llamado ACK (ACKnowledgment). El primer ACK significa que está listo para recibir el primer bloque.

El sistema emisor transmite entonces el primer bloque a la computadora receptora. El software de ésta busca el carácter de principio de encabezado (SOH), después checa el número del bloque, tanto para validez como para secuencia adecuada. A continuación examina los 128 caracteres de datos y calcula un checksum. Este checksum calculado lo compara con el checksum realmente enviado por el sistema emisor; si ambos son iguales, el sistema receptor supone que los datos se enviaron sin errores.

A medida que cada bloque es exitosamente recibido, otro carácter ACK se envía. Si se detecta un error, o el siguiente bloque no es enviado después de cierto tiempo, el sistema receptor envía un carácter de reconocimiento negativo llamado NAK (Negative Acknowledgement). Esto obliga al emisor a enviar nuevamente el último bloque. La transferencia se aborta después de que un mismo bloque se envía usualmente diez veces. Cuando se completa la transmisión, el sistema remoto envía al receptor una señal llamada EOT (End Of Transmission) y se termina la transferencia.

Si bien es cierto que los estándares originales de transferencia de archivos con el protocolo XMODEM están bien establecidos, no todos los programas de terminales o microcomputadoras están realmente establecidos conforme al protocolo estándar y existen variaciones entre ellos.

2.5 LA INTERFASE RS-232C.

2.5.1 DEFINICIONES.

Los puertos serie de las microcomputadoras personales IBM o Compatibles, son genéricos. Esto quiere decir que el puerto serie de la tarjeta IBM Asynchronous Communications Adapter es el mismo que el puerto de un panel multifuncional o el mismo de una microcomputadora compatible. Este puerto serie se conoce como Interfase RS-232C. Es una interfase eléctrica estándar definida por la Electronic Industries Association (EIA) en octubre de 1969 y se utiliza para conectar diferentes componentes periféricos a un sistema de computación, como módems, impresoras seriales, ratones y computadoras. Las iniciales "RS" significan "Recommended Standard" (Estándar Recomendado); "232" es el número del estándar y "C" es el nivel de revisión. Es decir, RS-232C es la tercera revisión del estándar básico, existiendo 13 revisiones desde la "A" hasta la "M" que tratan de cubrir todos los requerimientos de sistemas definidos. Este estándar es parecido al ISO International Standard 2110-1972, aunque hay incompatibilidades entre los dos en la asignación de algunos pines.

El estándar establecido para la interfase RS-232C define un camino de señal de 25 conductores que conforman 18 circuitos con retorno al través de tierra y los voltajes usados en todos los circuitos, es decir los rangos del 0 y del 1 lógicos. La razón de que la interfase RS-232C requiere 25 conductores fue debido a que al definirse el estándar tenía que satisfacer las necesidades de toda la industria eléctrica. En términos prácticos, esto significa que algo que tiene un puerto RS-232C posee un conector de 25 pines, 13 de los cuales están en una fila y 12 en otra. Estos conectores se llaman "DB-25". La mayoría de las terminales, módems y microcomputadoras, tienen conectores hembra y requieren conectores macho-macho para conectarlos, aunque existen excepciones a esta regla según las diferentes marcas. El puerto serie de una microcomputadora utiliza cuando mucho 9 conductores, aunque a veces sólo se requieren 3, por lo que existen conectores de 9 pines llamados "DB-9" con 5 y 4 pines en ambas hileras.

2.5.2 SEÑALES INTERNAS Y EXTERNAS.

Las señales lógicas dentro de una microcomputadora se ajustan a lo que se llama "niveles TTL", un estándar para la interconexión de los chips de circuitos integrados. Un voltaje interno entre 2 y 5 voltios es un "1" lógico, mientras que un voltaje de 0 a 0.8 voltios es un "0" lógico. Estos voltajes no se usan fuera de la PC debido a que no son inmunes al ruido eléctrico.

Por otra parte, una PC y la mayoría del equipo de comunicaciones producen señales de salida de ± 12 voltios, si bien una señal de ± 3 voltios es suficiente para definir un estado lógico. Aunque una señal es limpia al salir de la PC, al ser recibida por otra PC después de recorrer un conductor de 30 metros ya tiene ruido eléctrico, que desaparece al ser reconstruida dentro de la PC receptora.

Siempre hay que recordar que la señal de la interfase RS-232C tiene la polaridad cambiada con respecto a la señal TTL interna de la PC; es decir, un "1" lógico TTL de +3 voltios equivale a un "1" lógico o "marca" de -12 voltios.

2.5.3 IDENTIFICACION DE PINES.

El conjunto completo de conexiones sobre un conector RS-232C, se muestra en la siguiente tabla:

PIN	CIRCUITO	FUNCION Y DESCRIPCION
TIERRAS:		
1	AA	Tierra de Protección, soldada al chasis.
7	AB	Señal de Tierra, retorno común para todos los circuitos de intercambio (excepto el AA).
DATOS:		
2	BA	TD (Transmitted Data, Transmitir Datos), señales de transmisión de datos, generadas por la PC (DTE).
3	BB	RD (Received Data, Recibir Datos), entrada de datos a la PC desde el equipo de comunicacio-

nes (DCE).

SEÑALES DE CONTROL:

4	CA	RTS (Request To Send, petición de emisión), establecida por la PC cuando quiere emitir datos para condicionar al DCE local para la transmisión de datos.
5	CB	CTS (Clear To Send, listo para recibir), recibida por la PC cuando está lista para recibir datos. Es generada por el DCE para indicar que los datos están listos para ser transmitidos.
6	CC	DSR (Data Set Ready, conjunto de datos listo) señal recibida por la PC para indicar que el que el conjunto local está listo y que el módem está encendido y conectado.
20	CD	DTR (Data Terminal Ready, terminal de datos lista), establecida por la PC siempre que la comunicación de datos esté activa.
22	CE	RI (Ring Indicator, indicador de timbre), recibida por la PC cuando el módem está recibiendo una señal de timbre (sólo opera en modo de respuesta).
8	CF	CD (Carrier Detect ó Received Line Signal Detector, detector de portadora ó detector de señal de línea recibida), recibida por la PC cuando el módem detecta una señal que reconoce ciertos criterios establecidos por el fabricante y que se puede demodular.
21	CG	SQD (Signal Quality Detector, detector de la calidad de la señal), indica la calidad de la señal recibida del DCE, para indicar si existe alta probabilidad de errores.
23	CH	DSR (Data Signal Rate, velocidad de la señal de datos), indica la velocidad de la señal de datos del DTE origen.
23	CI	DSRS (Data Signal Rate Detector, detector de velocidad de la señal de datos), indica el selector de la velocidad de datos del DCE.

SEÑALES DE TIEMPO:

24	DA	TSET (Transmit Signal Element Timing, transmitir señal elemental de tiempo), indica la señal de sincronía.
15	DB	TSETDCE (Transmission Signal Element Timing DCE Source, señal de transmisión elemental de tiempo), indica el tiempo de la señal de ---- transmisión.
17	DD	RSET (Receiver Signal Element Timing, tiempo elemental de la señal de recepción), indica el tiempo de la señal de recepción.

DATOS SECUNDARIOS:

14	SBA	STD (Secondary Transmitted Data, transmisión secundaria de datos), se utiliza para transmitir datos por una vía secundaria o alterna y las señales de este circuito son generadas por el equipo terminal de datos (DTE).
16	SBB	SRD (Secondary Received Data, recepción secundaria de datos), usada para recibir datos

por un canal secundario o alternativo y es emitida por el DCE.

CONTROL SECUNDARIO:

19	SCA	SRTS (Secondary Request To Send, petición de emisión secundaria), utilizado para establecer un camino secundario para enviar datos y es equivalente al circuito CA.
13	SCB	SCTS (Secondary Clear To Send, listo para recibir secundario), es equivalente al circuito CB, pero de carácter secundario.
12	SCF	SRLS (Secondary Received Line Signal, secundario de señal de línea recibida), es un detector secundario de señales recibidas y es equivalente al circuito CF.

PINES RESERVADOS O NO ASIGNADOS: 9, 10, 11, 18 y 25.

2.5.4 PINES EN UNA MICROCOMPUTADORA.

En la comunicación de datos entre microcomputadoras, la mayoría de las tareas que se ejecutan no utilizan todos los circuitos mostrados en la tabla de la sección anterior. Los más importantes son Transmitir Datos (pin 2, circuito TD) y Recibir Datos (3, RD), al igual que la señal de tierra (7, Gnd).

El resto de los circuitos son circuitos de control y están controlados de formas diferentes según el Software de comunicaciones disponible. Las señales de control RTS y CTS no se usan tanto como las otras, habiéndose diseñado para controlar la comunicación Half-Duplex.

En la siguiente tabla se muestran los pines y señales utilizados al conectar dos microcomputadoras personales:

PIN	CIRCUITO	FUNCION Y DESCRIPCION
TIERRAS:		
1	AA	Tierra de Protección, soldada al chasis.
7	AB	Señal de Tierra, retorno común para todos los circuitos de intercambio (excepto el AA).
DATOS:		
2	BA	TD (Transmitted Data, Transmitir Datos), señales de transmisión de datos, generadas por la PC (DTE).
3	BB	RD (Received Data, Recibir Datos), entrada de datos a la PC desde el equipo de comunicaciones (DCE).
SEÑALES DE CONTROL:		
4	CA	RTS (Request To Send, petición de emisión), establecida por la PC cuando quiere emitir datos para condicionar al DCE local para la transmisión de datos.
5	CB	CTS (Clear To Send, listo para recibir), recibida por la PC cuando está lista para

		recibir datos. Es generada por el DCE para indicar que los datos están listos para ser transmitidos.
6	CC	DSR (Data Set Ready, conjunto de datos listo) señal recibida por la PC para indicar que el que el conjunto local está listo y que el módem está encendido y conectado.
20	CD	DTR (Data Terminal Ready, terminal de datos lista), establecida por la PC siempre que la comunicación de datos esté activa.
22	CE	RI (Ring Indicator, indicador de timbre), recibida por la PC cuando el módem está recibiendo una señal de timbre (sólo opera en modo de respuesta).
8	CF	CD (Carrier Detect ó Received Line Signal Detector, detector de portadora ó detector de señal de línea recibida), recibida por la PC cuando el módem detecta una señal que reconoce ciertos criterios establecidos por el fabricante y que se puede demodular.

2.6 PAQUETES INTEGRADOS.

2.6.1 GENERALIDADES.

Desde el año de 1984 han aparecido paquetes de software que contienen integrados varios programas de aplicación que anteriormente se vendían por separado, como hojas de cálculo, procesadores de texto, bases de datos, etc.. Estos paquetes se conocen con el nombre genérico de Software de Cuarta Generación o Paquetes Integrados. Los más conocidos son el FRAMEWORK de Ashton-Tate y el SYMPHONY de Lotus Development, cada uno en varias versiones.

Ambos paquetes contienen integradas en un solo programa las siguientes cinco funciones comunes:

- 1.- Procesamiento de texto.
- 2.- Hojas de cálculo.
- 3.- Bases de datos.
- 4.- Graficación.
- 5.- Telecomunicaciones.

Además de las anteriores, el FRAMEWORK soporta una sexta función conocida como Esquemas Dinámicos y un lenguaje de programación llamado FRED basado en Macros. Por su parte el SYMPHONY tiene su propio lenguaje de Macros y Telecomunicaciones.

2.6.2 FRAMEWORK II.

El paquete integrado Framework II (versión utilizada en este trabajo, aunque la impresión se hizo con el Framework III), incluye como nuevas prestaciones una gran capacidad de almacenamiento, mayor velocidad, mayor capacidad de integración, persona-

lización de programas, un programa de adaptación ya preparado con indicadores, una mayor integración de funciones y numerosas facilidades para la gestión de datos. También se han agregado nuevas funciones al lenguaje de programación FRED.

El Framework II proporciona versiones a escala completa de las seis herramientas de Software más populares en un único sistema, junto con la posibilidad de interactuar con otros sistemas de Software. Con esto, el sistema sirve como una herramienta de propósito general y como un ambiente multitarea capaz de alojar diferentes programas de aplicación para cada usuario. La versión II es bastante diferente de la I, por lo que se le cambió el nombre. Actualmente ya existe también la versión III.

Las principales mejoras del Framework II con respecto al Framework I, se indican en los siguientes incisos.

1.- Mayor capacidad de almacenamiento.

a) Memoria virtual. El Framework II permite ampliar la memoria de trabajo en un disco rígido, disco RAM o con tarjeta de ampliación de memoria. Con esta ampliación de memoria las hojas electrónicas pueden llegar a 32000 filas y 32000 columnas (1024 millones de celdas).

b) Mejor uso de la memoria de la microcomputadora. Mediante una forma nueva de representar y almacenar los marcos (frames), el Framework II utiliza más eficientemente la memoria interna disponible en la microcomputadora personal. Se ha duplicado la capacidad total de los archivos; para hojas electrónicas o bases de datos que tengan algunas celdas vacías, se ha incrementado de 2 a 50 veces.

c) Búsqueda de memoria. Si una aplicación está próxima al límite de la memoria interna de la microcomputadora, el Framework II busca automáticamente y utiliza la siguiente fuente de memoria virtual más rápida, ya sea un disco RAM, una tarjeta de ampliación de memoria o un disco duro.

2.- Mayor velocidad.

Se han mejorado los algoritmos y los cálculos con hojas electrónicas y bases de datos se ejecutan en Framework II de forma significativamente más rápida.

3.- Nuevas posibilidades.

a) Ganchos de aplicaciones. El Framework II incluye "ganchos" para añadir programas. Estos ganchos facilitan la instalación de programas de aplicaciones DOS y de programas personales en lenguaje de programación FRED, directamente ejecutables desde un menú del Framework II.

b) Importación y exportación de datos. Las opciones de un menú único permiten al usuario importar y exportar datos y textos desde y hacia archivos de otros sistemas, como dBASE II y III, WordStar, Multimate, IBM Displaywriter y Lotus 1-2-3.

c) Analizador ortográfico. El Framework II incluye un analizador ortográfico con un diccionario de aproximadamente 80000

palabras, la posibilidad de buscar palabras y además el usuario puede construir su propio diccionario de términos especiales. También analiza hojas electrónicas.

d) Simultanear telecomunicaciones. El Framework II permite ejecutar procesos de telecomunicaciones y realizar simultáneamente otras funciones.

e) Procesador de textos mejorado. El procesador de textos del Framework II incluye rupturas de página visibles, rupturas de página forzadas, guiones de continuación, espacios forzados y un contador de palabras.

f) Biblioteca personal. Esta biblioteca se activa automáticamente al iniciar el Framework II y se muestra en pantalla continuamente. Se utiliza para almacenar macros, abreviaturas, máscaras (templates), "hipótesis" para hojas electrónicas y otras utilidades.

g) Una nueva opción en gráficas. El Framework II permite crear gráficas de Alto-Bajo-Cierre empleadas en cotizaciones accionarias, además de los otros tipos de gráficas normales (de barras, circulares, XY, etc.).

4.- Mejoras en la integración, personalización y en las facilidades.

a) Telecomunicaciones integradas. El sistema de telecomunicaciones está completamente integrado en el Framework II y disponible continuamente. Todos los parámetros de comunicaciones están disponibles desde menús del Framework II.

b) Formatos de impresión basados en menús. La mayoría de las opciones de formatos de impresión se pueden utilizar desde menús. Además se pueden almacenar en la biblioteca personal donde están continuamente disponibles.

c) Almacenar macros. En cualquier instante se pueden almacenar macros para automatizar operaciones o combinaciones de teclas, pulsando la secuencia de teclas apropiada. Además están disponibles en el momento de activar el Framework II.

d) Imprimir etiquetas e insertar en cartas. Directamente desde un menú funciona una prestación para insertar en cartas y para imprimir etiquetas, combinando las posibilidades del Framework II para el procesamiento de textos y para la administración de bases de datos. Incluso puede operar con archivos creados en dBASE II y III.

e) Configuración personalizada. Un programa de configuración que solicita los parámetros, permite personalizar las diferentes prestaciones del Framework II. Simplifica los cambios y se acomoda casi a cualquier configuración de microcomputadora personal, monitor, unidad de disco, impresora y graficador.

f) Precauciones. El Framework II puede leer y procesar archivos del Framework I (versiones 1.0 y 1.1), pero sólo puede grabar archivos Framework II, mientras que el Framework I no puede leer archivos del Framework II.

2.6.3 SYMPHONY.

El paquete Symphony contiene los siguientes cinco ambientes integrados:

- 1.- Hoja de cálculo electrónica.
- 2.- Graficación.
- 3.- Procesador de texto.
- 4.- Base de datos.
- 5.- Telecomunicaciones.

Además de dichas funciones posee un lenguaje de comunicaciones y de macros.

3 CAPITULO II. INTERCONEXION DIRECTA.

3.1 HARDWARE Y CABLES UTILIZADOS.

3.1.1 CABLE DE MODEM NULO MODIFICADO.

La forma más sencilla de comunicaciones es conectar dos microcomputadoras personales utilizando únicamente tres conductores soldados a los pines 2 (transmisión), 3 (recepción) y 7 (tierra) de los conectores DB-25 correspondientes. El pin 2 de la PC1 se conecta al pin 3 de la PC2 y el pin 7 de ambas se conecta directamente; éste actúa como retorno común de tierra para completar los circuitos transmisor y receptor. Con un cable de tres conductores y dos conectores DB-25 se puede construir este tipo de conexión y se le llama Cable de Módem Nulo. Se debe observar que los pines 2 y 3 están cruzados.

El cable de módem nulo funciona bien con algunos paquetes de comunicaciones como el Crosstalk, pero no con todos. Algunos de ellos inspeccionan las líneas CTS (pin 5), CD (pin 8) y RTS (pin 4) y no funcionan a menos que alguna o todas estas señales sean "0" lógico u "ON". Sin embargo, se puede "engañarlos" conectando la señal RTS (pin 4) a las señales CTS (pin 5) y CD (pin 8); también se conecta la señal DSR (pin 20) a la DTR (pin 6). Estos puentes se efectúan en ambos conectores DB-25. Una desventaja de esta conexión es que ninguna de las PC's sabe si la otra está lista para la comunicación, por lo que se deben ejecutar simultáneamente los programas o paquetes de comunicaciones.

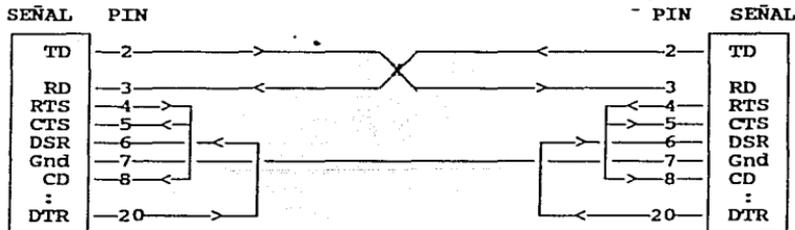
El cable descrito en el párrafo anterior se denomina Cable de Módem Nulo Modificado y fue el que se utilizó para conectar las dos microcomputadoras personales con las cuales se efectuaron las pruebas de comunicaciones en este trabajo. En la siguiente figura se muestra el diagrama esquemático de esta conexión.

MICROCOMPUTADORA 1

MICROCOMPUTADORA 2

(CONECTOR DB-25)

(CONECTOR DB-25)



Cable de Módem Nulo Modificado.

1.- Dos microcomputadoras personales Printaform con micro-procesador 8088 y 640 Kb de memoria, con sus respectivos puertos serie RS-232C. La primera con una unidad de disco flexible de 5.25 pulgadas para discos de 360 Kb y con una unidad de disco rígido de 20 Mb. La segunda con dos unidades de disco de 5.25 pulgadas y 360 Kb. Los conectores de ambas eran DB-25 machos.

2.- El Cable de Módem Nulo Modificado, descrito en la sección 3.1.1, convirtiendo sus conectores en DB-25 hembras.

3.2 COMUNICACIONES CON BASIC.

3.2.1 SOFTWARE UTILIZADO.

Con el objeto de efectuar las pruebas de comunicaciones con el lenguaje GW-BASIC, además del hardware anteriormente descrito, se utilizaron los siguientes elementos de software:

- 1.- El sistema operativo MS-DOS, versión 3.2.
- 2.- El Lenguaje Interpretador GW-BASIC, versión 3.20.
- 3.- El Programa TTY.BAS, que se describe en las siguientes secciones.
- 4.- Un archivo de texto (DISTEX.TXT), escrito con el Procesador de Texto WordStar, versión 3.02 en español.
- 5.- Un programa escrito en GW-BASIC (archivo DISCOXY.BAS).

3.2.2 INTRODUCCION.

3.2.2.1 ENTRADA/SALIDA (I/O) EN BASIC.

Debido a que cualquier puerto de comunicaciones se abre como cualquier archivo en BASIC, todos los comandos de Entrada/Salida que son válidos para archivos en disco son válidos para archivos COM.

Los comandos de entrada secuenciales en COM son los mismos que aquéllos para los archivos en disco. Estos son: INPUT #*número de archivo*, LINE INPUT #*número de archivo* y la función INPUT\$.

Los comandos de salida secuenciales en COM son los mismos que aquéllos para los archivos en disco y son: PRINT #*número de archivo* y PRINT #*número de archivo* USING.

3.2.2.2 FUNCIONES DE ENTRADA/SALIDA.

El aspecto más difícil de la comunicación asíncrona es su habilidad para procesar los caracteres tan rápido como sean recibidos. A velocidades arriba de 2400 bps, es necesario suspender la transmisión de caracteres desde la computadora anfitriona el tiempo suficiente para almacenarlos o procesarlos. Esto puede hacerse enviando a la anfitriona el caracter XOFF (o sea Ctrl-S). Cuando la microcomputadora receptora ha terminado de almacenar o

procesar los caracteres envía a la anfitriona el caracter XOM (Ctrl-Q) para reanudar la transmisión. El BASIC proporciona tres funciones que ayudan a detectar cuando una condición de "over-run" es inminente. Estas son:

LOC(x).- Esta función regresa el número de caracteres en la cola de entrada que esperan ser leídos. La cola de entrada puede contener más de 255 caracteres (determinado por el switch /C:). Si hay más de 255 caracteres en la cola, LOC(x) regresa 255. Puesto que una cadena está limitada a 255 caracteres, este límite práctico alivia al programador de la necesidad de probar el tamaño de la cadena antes de leer datos en ella. Si hay menos de 255 caracteres que permanecen en la cola, LOC(x) retorna la cuenta actual.

LOF(x).- Esta función regresa la cantidad de espacio libre en la cola de entrada. Esto es, el tamaño de la cola (/C:) menos el valor de la función LOC(x). La función LOF(x) se usa para detectar cuándo se va a llenar la cola de entrada.

EOF(x).- El valor CIERTO (o sea -1) de esta función, indica que la cola está vacía. O bien regresa el valor FALSO (o sea 0) si hay caracteres que esperan ser leídos.

3.2.2.3 FUNCION INPUT\$ PARA ARCHIVOS COM.

Es preferible usar la función INPUT\$ en lugar de los comandos INPUT # y LINE INPUT # cuando se leen archivos COM, ya que todos los caracteres ASCII pueden ser significativos en Comunicaciones. INPUT # es la menos deseable debido a que detiene la entrada cuando una coma o un retorno de carro aparecen, y LINE INPUT # se termina cuando aparece un retorno de carro.

En cambio, la función INPUT\$ permite que todos los caracteres leídos sean asignados a una cadena. INPUT\$ regresará X caracteres del archivo #Y. Las siguientes proposiciones, por lo tanto, serán más eficientes para leer un archivo COM:

```
10 WHILE NOT EOF(1)
20 A$=INPUT$(LOC(1),#1)
.
.   Proceso de los datos almacenados en A$
.
60 WEND
```

La secuencia anterior de enunciados se lee como: "... Mientras haya algo en la cola de entrada, regresar el número de caracteres en la cola y almacenarlos en A\$. Si hay más de 255 caracteres, solamente serán regresados 255 a la vez para evitar el derrame de la cadena ("String Overflow"). Además, si este es el caso, EOF(1) es FALSO y la entrada continúa hasta que la cola de entrada queda vacía ...".

3.2.3 PROGRAMA DE COMUNICACIONES EN BASIC.

3.2.3.1 CONSIDERACIONES.

Con el objeto de probar las opciones de comunicaciones en BASIC, se preparó el Programa TTY que se muestra en la siguiente Sección. Este programa habilita a una de las dos microcomputadoras para que actúe como una terminal convencional con respecto a la otra. Además de comunicarse con una computadora anfitrión, este programa permite que los datos recibidos sean "descargados" (downloaded) en un archivo. También un archivo puede ser "cargado" (uploaded) hacia otra máquina.

Además de demostrar los elementos que intervienen en la comunicación asíncrona, este programa es útil para transferir programas o archivos ASCII de datos de una a otra microcomputadora.

El programa original fue tomado del manual del lenguaje BASIC y fue usado inicialmente para comunicarse con un sistema DEC SYSTEM-20, por lo cual tuvo que ser modificado para ser utilizado en dos Microcomputadoras Personales Compatibles Printaform.

3.2.4 PROGRAMA TTY DE COMUNICACIONES.

```
5 'PROGRAMA TTY (PRUEBA DE TELECOMUNICACIONES)
10 SCREEN 0,0:WIDTH 80
15 KEY OFF:CLS:CLOSE
20 DEFINT A-Z
25 LOCATE 25,1
30 PRINT STRING$(60," ")
40 FALSE=0:TRUE=NOT FALSE
50 MENU=5 ' VALOR DE LA TECLA MENU (CTRL E)
60 XOFF$=CHR$(19):XON$=CHR$(17)
100 LOCATE 25,1:PRINT "PROGRAMA TTY ASINCRONO"
110 LOCATE 1,1:LINE INPUT "DAR VELOCIDAD: ";SPEED$
120 COMFIL$="COM1:"+SPEED$+"N,8,1,DS,CD"
130 OPEN COMFIL$ AS #1
190 '***** EMULACION DE TERMINAL *****
200 PAUSE=FALSE
210 A$=INKEY$:IF A$="" THEN 230
215 PRINT A$;
220 IF ASC(A$)=MENU THEN 300 ELSE PRINT #1,A$;
230 IF EOF(1) THEN 210
240 IF LOC(1) > 128 THEN PAUSE=TRUE:PRINT #1,XOFF$;
250 A$=INPUT$(LOC(1),#1)
260 PRINT A$;;IF LOC(1) > 0 THEN 240
270 IF PAUSE THEN PAUSE=FALSE:PRINT #1,XON$;
280 GOTO 210
290 '*** TRANSFERENCIA DE ARCHIVOS ASCII ***
295 CLS
300 LOCATE 1,1:PRINT STRING$(50," "):LOCATE 1,1
310 LINE INPUT"NOMBRE DEL ARCHIVO: ";DSKFIL$
400 LOCATE 1,1:PRINT STRING$(50," "):LOCATE 1,1
410 LINE INPUT "(T)ransmitir o (R)ecibir: ";TXRX$
420 IF TXRX$="T" THEN OPEN DSKFIL$ FOR INPUT AS #2:
GOTO 1000
425 '*** RECEPCION DEL ARCHIVO ***
430 OPEN DSKFIL$ FOR OUTPUT AS #2
```

```

440 PRINT #1,CHR$(13);
500 IF EOF(1) THEN GOSUB 600
510 IF LOC(1)>128 THEN PAUSE=TRUE: PRINT #1,XOFF$;
520 A$=INPUT$(LOC(1),#1)
530 PRINT #2,A$; IF LOC(1)>0 THEN 510
540 IF PAUSE THEN PAUSE=FALSE: PRINT #1,XON$;
550 GOTO 500
590 '*** SUBROUTINA DE TIME-OUT ***
600 FOR I=1 TO 5000
610 IF NOT EOF(1) THEN I=9999
620 NEXT I
630 IF I>9999 THEN RETURN
640 CLOSE #2: CLS: LOCATE 25,10
645 PRINT "**** RECEPCION COMPLETA ****";
650 GOTO 200
900 '*** TRANSMISION DEL ARCHIVO ***
1000 WHILE NOT EOF(2)
1010 A$=INPUT$(1,#2) 'LEER UN CARACTER
1015 PRINT A$;
1020 PRINT #1,A$; 'ENVIAR EL CARACTER A COM
1030 WEND
1040 PRINT #1,CHR$(26);'CTRL-Z PARA CERRAR EL ARCHIVO
1050 CLOSE #2: CLS: LOCATE 25,10
1055 PRINT "**** TRANSMISION COMPLETA ****"
1060 GOTO 200
9999 CLOSE: KEY ON

```

3.2.5 DESCRIPCION DEL PROGRAMA TTY.

3.2.5.1 ANTECEDENTES.

La comunicación asíncrona implica entrada/salida (I/O) de caracteres, lo cual es opuesto a entrada/salida de líneas o bloques. Por lo tanto, todas las impresiones (PRINT's), ya sea hacia el archivo COM o a la pantalla, están terminadas con un punto y coma (;). Esto suprime el retorno de carro y la alimentación de línea normalmente suministrados al final de un comando PRINT.

3.2.5.2 DESCRIPCION.

En esta sección se suministra una descripción general del programa TTY, de acuerdo a los bloques que lo integran.

Primeramente en la línea 5 se identifica al programa como TTY y en la 10 se coloca la pantalla en modo alfabético, blanco y negro y de 80 columnas. Después en la 15 se apaga el despliegue de teclas de función, se limpia la pantalla y se asegura de que todos los archivos estén cerrados.

La línea 20 define todas las variables numéricas como enteras. Esto es principalmente para la subrutina en 600-620. Cualquier programa que busque la optimización de la velocidad, usará contadores enteros siempre que sea posible. En 25-30 se limpia la línea 25 de la pantalla, principiando en la columna 1. La 40 define lógicas las variables TRUE y FALSE.

En la 50 se define el valor ASCII de la variable MENU (Ctrl-E). La 60 asigna el valor ASCII 19 (Ctrl-S) a la variable XOFF\$ y el valor ASCII 17 (Ctrl-Q) a la variable XON\$.

La línea 100 imprime la identificación del programa; en la 110 se solicita la velocidad (300, 900, 2400 bps, etc.). La 120 forma la cadena para el archivo COM1 de comunicaciones, sin paridad (N), 8 bits de datos, 1 bit de parada y se apagan las señales DS y CD. En la 120 se abre el archivo # 1 para comunicaciones.

En las líneas 200-280, se efectúa la transmisión entre la pantalla de video y el dispositivo conectado a la interfase RS-232C, de la siguiente forma:

1.- En primer lugar se lee un caracter del teclado dentro de la variable A\$ (línea 210). Nótese que la función INKEY\$ regresa un caracter nulo si ningún caracter está esperando.

2.- Si ningún caracter está esperando, entonces va a ver si algún caracter está siendo recibido. Si un caracter está esperando en el teclado, entonces:

a) Si el caracter fue la tecla MENU, entonces el usuario está listo para descargar un archivo (línea 220), por lo cual el programa solicita el nombre del archivo (líneas 300-310).

b) Si el caracter (A\$) no fue la tecla MENU, lo envía para escritura al archivo de comunicaciones (línea 220, ... ELSE PRINT #1,A\$).

3.- En la línea 230 se ve si algún caracter está esperando en el buffer COM. Si no lo hay, se regresa a revisar el teclado (a la línea 210).

4.- En la 240, si más de 128 caracteres están esperando, entonces se almacena el valor CIERTO (de la variable TRUE) para decir que se está suspendiendo la entrada y se envía a la microcomputadora anfitriona la señal XOFF (caracter ASCII 19) para que detenga cualquier transmisión adicional.

5.- En las líneas 250-260 se lee y se despliega en la pantalla el contenido del buffer COM hasta que se vacía (en 240). Se suspende la transmisión si caemos atrás.

6.- Finalmente se reanuda la transmisión de la micro anfitriona enviándole la señal de XON, sólo si estaba suspendida por alguna señal previa de XOFF. El proceso se repite hasta que se oprime la tecla MENU (Ctrl-E).

En las líneas 300-310 se solicita el nombre del archivo, se almacena en DSKFIL\$ y será el #2.

En 400-420 se pregunta si el archivo va a ser transmitido a la microcomputadora anfitriona (Uploaded), o bien recibido de ella (Downloaded).

Si el archivo se recibe de la emisora (se descarga), primeramente se abre para salida el archivo #2 (línea 430) y se le envía un caracter de Retorno de Carro (ASCII 13, línea 440) para que principie la descarga. Este programa supone que el último

comando enviado a la anfitriona fue para principiar la transferencia y únicamente le estaba faltando el caracter de retorno de carro.

En la 500, cuando no hay más caracteres que estén siendo recibidos (la función LOC(1) regresa 0), entonces se efectúa una rutina de fuera de tiempo ("time-out", ... GOSUB 600) explicada después.

En la 510 nuevamente se señala una pausa, si más de 128 caracteres están esperando, y se envía un XOFF a la anfitriona para poder almacenar los caracteres en el archivo.

En las líneas 520-530 se leen todos los caracteres de la cola COM (LOC(1)) y se graban en el disco (PRINT #2, A\$) hasta que se terminan.

En las líneas 540-550, si una pausa fue suministrada, la anfitriona reanuda la transmisión enviándole una señal XON y limpiando la pausa. El proceso continúa hasta que no más caracteres se reciben en un tiempo determinado.

En las líneas 600-650 se encuentra la subrutina de tiempo-fuera. La cuenta para el ciclo FOR se determinó experimentalmente. En resumen, si ningún caracter se recibe de la anfitriona por cerca de 17-20 segundos, entonces se supone que la transmisión está completa. Si cualquier caracter se recibe durante este tiempo (línea 610), se establece la variable que controla (I) muy arriba del rango del ciclo FOR, para salirse del ciclo y regresar al programa que llamó a la subrutina. Si la transmisión de la microcomputadora anfitriona se ha terminado, se cierra el archivo en disco (línea 640 CLOSE #2 ...) de salida y el programa regresa a seguir actuando como una terminal (línea 200).

En las líneas 1000-1060 está la rutina de transmisión del archivo, previamente abierto de entrada para lectura en la línea 420 como #2. Primero se lee un caracter en A\$ con el comando INPUT\$. Se envía el caracter hacia el dispositivo COM en la 1020. Se envía el caracter Ctrl-Z (ASCII 26) en caso de que el dispositivo receptor necesite uno para cerrar su archivo. Finalmente, en las líneas 1050-1060 se cierra el archivo en disco #2, se imprime el mensaje de terminación y se regresa al modo de conversación en la línea 200.

La línea 9999 no se ejecuta en este programa. Se pueden modificar las líneas 400-420 para terminar el programa con esta línea. En ésta se cierra el archivo COM dejado abierto y se restaura el despliegue de las teclas de funciones.

3.2.5.3 ERRORES EN COMUNICACIONES.

Las siguientes condiciones de error pueden ocurrir en los archivos de comunicaciones:

- 1.- Cuando se abre un archivo, puede aparecer el mensaje "Device Timeout". Este se presenta si alguna de las señales probadas (CTS, DSR, o CD) está faltando. Para evitarlo el programa se modificó para no detectarlas.

2.- Cuando se lee un archivo pueden presentarse los dos errores siguientes:

a) "Communication Buffer Overflow". Se presenta si ocurre una condición de overflow; ésto indica que el Buffer de Comunicaciones necesita ser incrementado (con la opción /C: en la inicialización), o bien indica que el programa necesita leer caracteres del buffer mas rápidamente.

b) "Device I/O Error". Aparece si ocurren errores de derrame ("overrun"), rotura ("break"), paridad o de delimitación ("framing").

3.- Cuando se escribe un archivo puede aparecer el error "Device Timeout". Se presenta si se pierden las señales CTS, DSR o CD mientras se envían datos desde el buffer de salida.

3.2.6 PRUEBAS DEL PROGRAMA.

3.2.6.1 TRANSMISION DE TEXTO.

Una vez terminado el programa TTY y depurado de la mayoría de errores, se procedió a efectuar la primera prueba consistente en transmitir mensajes de texto de una primera microcomputadora emisora hacia una segunda receptora y viceversa. Previamente se habían conectado ambas micros con el Cable de Módem Nulo Modificado. Los pasos seguidos fueron los siguientes:

1.- Conexión de ambas microcomputadoras con el cable de Módem nulo modificado.

2.- Carga en ambas micros del sistema operativo MS-DOS (Ver. 3.2).

3.- Carga en ambas micros del lenguaje intérprete GW-BASIC (Ver. 3.2).

4.- Carga en ambas computadoras del programa TTY de comunicaciones.

5.- Ejecución simultánea del programa TTY en ambas micros.

Al ejecutarse, el programa TTY primero escribe en la pantalla la identificación "PROGRAMA TTY ASINCRONO". A continuación solicita la velocidad de transmisión. Esta velocidad inicialmente se suministró de 300 bps, pero se hicieron pruebas hasta de 2400 bps. La manera de especificarla era dar el valor en las dos microcomputadoras y oprimir Enter simultáneamente en ambas.

En las primeras pruebas del programa se corrigieron los errores normales y se procedió a enviar algunos caracteres desde la micro 1 emisora, hasta que fueron aceptados por la micro 2 receptora. Cuando ya todo estaba en orden, se transmitieron varios renglones de texto terminados con la tecla Enter: Primero desde la micro 1 a la 2 y después desde la 2 hacia la 1, con lo cual se terminó con éxito la primera prueba.

En el siguiente cuadro se transcribe la primera prueba de transmisión de texto realizada con el programa TTY de comunicaciones, impresa con Shift-PrtSc.

DAR VELOCIDAD: 1200
MEXICO, D. F. A 31 DE MAYO DE 1989.
ESTA ES UNA PRUEBA DE TELECOMUNICACIONES ENTRE DOS COMPUTADORAS
USANDO UN CABLE DE MODEM NULO.
AHORA SE ESTA TRANSMITIENDO DESDE LA OTRA MICRO.
ESTO SE ESTA ENVIANDO DESDE LA MICRO 1.
ESTO SE ENVIA DESDE LA MICRO 2.
La prueba resultó exitosa.

Esto se imprimió con Shift-PrtSc y también funcionó.

En esta primera prueba no se enviaron archivos completos, mas bien sirvió para depurar el programa, comprobar el alambrado del cable y transmitir caracteres a varias velocidades.

3.2.6.2 TRANSMISION DE ARCHIVOS ASCII.

Una vez terminada la primera prueba del programa TTY, se procedió a efectuar la segunda prueba consistente en transmitir un archivo de texto de una primera microcomputadora emisora hacia una segunda receptora y viceversa. Este archivo fue elaborado con el procesador de texto WordStar (Ver. 3.02 en español), en modo de no documento para escribir sólo texto ASCII puro y se le dio el nombre DISTEX.TXT. Previamente se habían conectado ambas micros con el cable de Módem nulo modificado. Los pasos seguidos fueron los siguientes:

- 1.- Conexión de ambas microcomputadoras con el cable de Módem nulo modificado.
- 2.- Carga en ambas micros del sistema operativo MS-DOS (Ver. 3.2).
- 3.- Carga en ambas micros del lenguaje intérprete GW-BASIC (Ver. 3.2).
- 4.- Carga en ambas computadoras del programa TTY de comunicaciones.
- 5.- Ejecución simultánea del programa TTY en ambas micros.

Al ejecutarse el programa TTY primero escribe en la pantalla la identificación "PROGRAMA TTY ASINCRONO". A continuación solicita la velocidad de transmisión. Esta velocidad inicialmente se suministró de 1200 bps. La manera de especificarla era dar el valor en las dos microcomputadoras y oprimir Enter simultáneamente en ambas.

En esta segunda prueba del programa se procedió a enviar el archivo ASCII DISTEX.TXT desde la micro 1 emisora hacia la micro 2 receptora. Primeramente se transmitieron varios renglones de

texto terminados con la tecla Enter primero desde la micro 1 a la 2 y después desde la 2 hacia la 1 para probar la conexión. Después se oprimieron las teclas Ctrl-E en ambas micros para terminar la transmisión de caracteres y poder transmitir el archivo.

Al solicitarse el nombre del archivo, en la micro 1 se dio el nombre DISTEX.TXT, mientras que en la micro 2 (receptora) se dio el nombre DISTEX2.TXT con el objeto de diferenciarlos.

Al aparecer el mensaje "(T)ransmitir o (R)ecibir", en la micro 1 (emisora) se dio la opción "T" para Transmitir y en la micro 2 (receptora) se dio la opción "R" para recibir el archivo.

Al terminar de transferirse el archivo, en la micro emisora (# 1) apareció el mensaje "**** TRANSMISION COMPLETA ****", mientras que en la micro receptora se indicó "**** RECEPCION COMPLETA ****". Con esto se terminó exitosamente la segunda prueba de comunicaciones del programa TTY:

El siguiente cuadro se muestra el archivo de texto de la segunda prueba realizada con el programa TTY de comunicaciones.

ESTE ES UN ARCHIVO DE TEXTO PARA PRUEBA DEL PROGRAMA TTY. SERA PARA QUE SE TRANSMITA DE UNA MICROCOMPUTADORA A OTRA. AMBAS MICROS ESTAN CONECTADAS CON UN CABLE DE MODEM NULO. LOS PINES 2 Y 3 ESTAN CRUZADOS. Los pines 4 y 5 están puenteados con el pin 8. El pin 6 está puenteadado al pin 20. El pin 7 está unido directamente.

Se usa el protocolo XON/XOFF

Después de recibirse el archivo DISTEX2.TXT en la micro 2 se procedió a listarlo y se comprobó que la transmisión estuvo correcta, con lo cual se terminó la segunda prueba de comunicaciones en BASIC.

3.2.6.3 ARCHIVOS DE PROGRAMAS.

La tercera y última prueba efectuada con el programa TTY fue para transmitir un programa escrito en GW-BASIC (DISCOXY.BAS, para crear un archivo de datos en BASIC). Esta prueba resultó un fracaso, ya que incluso se dañaron todos los archivos del disco flexible desde el cual se trataba de enviar ese programa.

Después de investigar se llegó a la conclusión de que el programa contenía caracteres de control en código ASCII superiores al ASCII 127, utilizados por el programa intérprete GW-BASIC. Esto se debió a que el programa fue respaldado con el comando SAVE "DISCOXY". En lugar de esto se debió haber salvado con el comando SAVE "DISCOXY", A con el objeto de tener el archivo en código ASCII directo, o sea en texto puro sin caracteres de control indeseables. El archivo ASCII así generado ya no se probó.

Si bien esta tercera prueba no terminó con éxito, sirvió para que se tenga cuidado al transmitirse los archivos ASCII, sobre todo cuando se trata de programas.

3.3 COMUNICACIONES CON FRAMEWORK II.

3.3.1 INTRODUCCION.

3.3.1.1 SOFTWARE UTILIZADO.

Con el objeto de efectuar las pruebas de comunicaciones con el paquete integrado Framework II, además del hardware anteriormente descrito, se utilizaron los siguientes elementos de software:

- 1.- El sistema operativo MS-DOS, versión 3.2.
- 2.- El paquete integrado Framework II, versión en inglés.
- 3.- Un archivo ASCII (PRUDUP), creado con el comando COPY del sistema operativo MS-DOS, en un curso previo de Framework II.
- 4.- Un archivo creado en forma de "esquema" con el paquete Framework II (\FRAME\CURSO\PRACT8.FW2), consistente en tres hojas de cálculo y una gráfica circular (tipo "pie"), generado durante el curso mencionado.
- 5.- Un conjunto de 6 archivos de diferentes tipos, creados con el Framework II, todos ellos con la extensión común FW2.
- 6.- Un conjunto de 10 archivos creados con el paquete LOTUS 1-2-3 (V. 2.01), generados en un curso previo de dicho paquete y con la extensión común WK1.

3.3.1.2 PREPARACION DEL EQUIPO.

Con el objeto de preparar la conexión entre las dos microcomputadoras, los pasos comunes seguidos para cada una de las pruebas, fueron los siguientes:

- 1.- Conexión física de ambas microcomputadoras con el cable de módem nulo modificado, descrito en la sección 3.1.1, antes de prender los equipos.
- 2.- Carga del sistema operativo (Ver. 3.2) en ambas microcomputadoras.
- 3.- Carga del paquete integrado Framework II en las dos microcomputadoras.

3.3.1.3 MARCOS DE TELECOMUNICACIONES.

Una vez conectadas físicamente ambas microcomputadoras y cargados tanto el sistema operativo como el Framework II, el siguiente proceso común para todas las pruebas fue crear un marco de telecomunicaciones, tanto para la microcomputadora emisora como para la receptora, con el objetivo de guardar los resultados

de las pruebas en archivos. Para ello, se siguieron los siguientes pasos:

- 1.- Ctrl-C para llamar al menú Create.
- 2.- Selección de la opción Empty/Word Frame y Enter, para crear un marco vacío o de texto.
- 3.- A continuación se tecló el nombre en la etiqueta del marco, terminándolo con la tecla Enter.

Para cada prueba se crearon dos marcos vacíos (uno para cada microcomputadora), con nombres diferentes para que se pudieran diferenciar y respaldar al final de la prueba.

3.3.1.4 CONEXION CON EL FRAMEWORK II.

Con el objeto de establecer mediante Software la conexión entre ambas microcomputadoras con el Framework II, y después de haber creado los marcos de telecomunicaciones, se procedió de la forma siguiente:

1.- De la etiqueta del marco creado se pasó al interior del mismo, oprimiendo la tecla DownLevel (tecla + gris).

2.- Ya dentro del marco, se procedió a establecer la conexión con el Framework II, tecleando Ctrl-A para llamar al menú de aplicaciones (Apps, Applications).

3.- A continuación se seleccionó la opción de telecomunicaciones (Telecommunications), oprimiendo la tecla T. Con esto apareció una tabla con ocho opciones de comunicaciones. Las primeras seis son para conectarse a diferentes servicios, tales como el Dow Jones, CompuServe, The Source, etc.. La séptima opción se refiere a llamar a otra microcomputadora personal, mientras que la octava es para responder a la microcomputadora que llama.

4.- En el caso de la microcomputadora que llama (micro 1, emisora), se seleccionó la opción 7 (PC-to-PC). Así apareció el mensaje "Use this frame for Telecommunications Session? (y/n)", con el que el Framework II pide permiso para usar el marco actual para telecomunicaciones. Puesto que ya se había creado y etiquetado el marco, se tecló "y" para responder afirmativamente.

5.- Para la microcomputadora que recibe (micro 2, receptora), se seleccionó la opción 8 (ANSWER mode). Con ello apareció nuevamente el mensaje "Use this frame for Telecommunications Session? (y/n)", con el que el Framework II pide permiso para usar el marco actual para telecomunicaciones. Puesto que ya se había creado el marco, se tecló "y" para responder afirmativamente.

Los pasos anteriores (1 a 3) fueron realizados en forma simultánea en las dos microcomputadoras. El paso 4 sólo se hizo en la micro 1 emisora, y el paso 5 sólo para la micro 2 receptora.

De esa forma se procedió para cada una de las pruebas.

3.3.2 PRIMERA PRUEBA.- TRANSMISION DE TEXTO.

3.3.2.1 DESCRIPCION.

Los objetivos principales de esta primera prueba de comunicaciones con el Framework II, fueron probar el cable construido y la manera de establecer la comunicación entre las dos microcomputadoras con el Framework II; posteriormente transmitir una serie de líneas de texto en código ASCII puro, sin incluir caracteres superiores al código 127 para evitar complicaciones. Los pasos fueron los siguientes:

1.- En primer lugar se conectaron ambas microcomputadoras, como se describe en la sección 3.3.1.2.

2.- A continuación se crearon los marcos de telecomunicaciones, como se indica en la sección 3.3.1.3. Para la microcomputadora emisora el nombre del marco fue TELEPC, mientras que para la microcomputadora receptora el nombre fue TELERESP.

3.- En tercer lugar se procedió a establecer la conexión por Software entre las dos microcomputadoras, tal como se describe en la sección 3.3.1.4.

Una vez que la conexión con el Framework II queda establecida entre ambas microcomputadoras, suceden varias cosas. Primeramente, en el marco TELEPC (micro 1, emisora) aparecen los mensajes de bienvenida del Framework II ("ATSO=1", "Welcome to Framework Telecommunications" y "Hi! This is XXXX's PC"), los cuales se pueden observar en los primeros renglones del marco TELEPC mostrado en la siguiente sección. En segundo lugar, ya que se estaba en el marco activo de comunicaciones, en la parte inferior de la pantalla aparece el mensaje "Telecommunications is active". En el caso de estar fuera del marco activo, sólo aparecería el indicador "Tcom" debajo de la bandeja del nivel superior, en el espacio donde normalmente aparece el indicador "CAPS".

Al quedar establecida la comunicación con el Framework II, se procedió a teclear toda la información que se muestra en el marco TELEPC de la sección siguiente. Al terminar este proceso se terminó la sesión de telecomunicaciones con el Framework II, seleccionando la opción "End Session" del menú Telecommunications. Finalmente se procedió a guardar en disco los marcos TELEPC y TELERESP, aunque se debe aclarar que el último marco se perdió por una falla al salvarlo.

En la siguiente sección se muestra un listado del marco TELEPC, con los mensajes del Framework II y la información teclada, si bien se corrigió para fines de presentación en este trabajo, sobre todo los errores ortográficos.

3.3.2.2 MARCO TELEPC.

ATSO=1

Welcome to Framework Telecommunications

Hi! This is XXXX's PC.

ESTA ES UNA PRUEBA DE LAS TELECOMUNICACIONES DEL FRAMEWORK II.

Los pasos seguidos fueron los siguientes:

- 1.- En primer lugar se crearon dos marcos: el TELEPC y el TELERESP.
- 2.- El marco TELEPC se usó para llamar a la otra micro.
Los pasos fueron: a) Ctrl-A, para llamar al menú Apps.
b) Se seleccionó la opción telecomunicaciones.
c) Del segundo menú se usó la opción PC-TO-PC.
d) Se guardó en el disco B: el marco TELEPC.
- 3.- El marco TELERESP se usó para la micro 2 en modo respuesta. los pasos fueron:
a) Ctrl-C para crear el marco TELERESP.
b) Ctrl-A para llamar al menú Apps.
c) Selección de la opción Telecomunicaciones.
d) Del segundo menú se usó la opción ANSWER.
e) En el disco B: se guardó el marco TELERESP.
- 4.- Se conectaron las dos micros con un cable de módem nulo.
- 5.- Se cargó el FRAMEWORK II en las dos micros.
- 6.- En la micro 1 se seleccionó la opción PC-TO-PC.
- 7.- En la micro 2 se seleccionó el modo ANSWER.
- 8.- Una vez conectadas ambas micros se procedió a teclear esta información desde la micro 1 (en modo origen) y se recibió en la pantalla de la micro 2 (en modo respuesta).

3.3.3 SEGUNDA PRUEBA.- USO DEL XMODEM.

3.3.3.1 DESCRIPCION.

Los objetivos principales de esta segunda prueba de comunicaciones con el Framework II, fueron: a) Transmitir algunas líneas de texto y b) Transmitir un archivo con el protocolo XMODEM. Los pasos fueron los siguientes:

1.- En primer lugar se conectaron ambas microcomputadoras, como se describe en la sección 3.3.1.2.

2.- A continuación se crearon los marcos de telecomunicaciones, como se indica en la sección 3.3.1.3. Para la microcomputadora emisora el nombre del marco fue TELENV2, mientras que para la microcomputadora receptora el nombre fue TELRES2.

3.- En tercer lugar se procedió a establecer la conexión entre las dos microcomputadoras, tal como se describe en la sección 3.3.1.4.

El archivo seleccionado para transmitirse fue un archivo generado en un curso previo de Framework II, consistente en tres hojas de cálculo y una gráfica circular (tipo "pie"), cuyo nombre era \FRAME\CURSO\PRACT8.FW2.

En la siguiente sección se muestra el marco TELENV2 de la microcomputadora emisora. Los primeros cuatro párrafos se incluyeron para fines de documentación únicamente.

Una vez que la conexión con el Framework II quedó establecida entre ambas microcomputadoras, primeramente en el marco TELENV2 (micro 1, emisora) aparecieron los mensajes de bienvenida del Framework II ("ATSO=1", "Welcome to Framework Telecommunications" y "Hi! This is XXXX's PC"), los cuales se pueden observar después de los cuatro primeros párrafos del marco TELENV2 mostrado en la siguiente sección. En segundo lugar, ya que se estaba en el marco activo de comunicaciones, en la parte inferior de la pantalla apareció el mensaje "Telecommunications is active".

Al quedar establecida la comunicación con el Framework II, se procedió a transmitir el archivo mencionado, según se muestra en el marco TELENV2 de la sección siguiente. Para ello, en primer término se seleccionó el menú de aplicaciones con Ctrl-A, escogiendo la opción de Telecommunications y dentro de ésta la opción File Transfer. En este punto, para la micro 1 emisora se seleccionó la opción Send a File, dando el nombre del archivo a enviar (\FRAME\CURSO\PRACT8.FW2). Para la micro 2 receptora se escogió la opción Receive a File, dando el nombre del archivo a recibir (PRACT8.DUP). Se puede observar que aparece el mensaje "No such file", lo cual se debió a que el nombre del archivo era demasiado largo, ya que incluía dos subdirectorios (\FRAME\CURSO\). El problema se solucionó dando primero los nombres de los subdirectorios, antes del nombre del archivo. El segundo problema que surgió se debió a que inicialmente en la micro 2 receptora no se había dado la opción de ANSWER, lo cual originó el mensaje "Transmission Aborted"; en este punto se escribió el mensaje "NO SE PUDO ENVIAR EL ARCHIVO" por el autor de este trabajo. El Framework por su parte imprimió la secuencia "CCC*****", lo cual indica que había problemas de transmisión, ya que cada letra C y cada asterisco indican que los bloques de datos no podían enviarse. Al dar la opción ANSWER en la micro 2 se solucionó este segundo problema.

Resueltos los problemas anteriores, ya fue posible enviar el archivo. El primer renglón que aparece correcto es "Attempting to Synchronize", que indica que el Framework II intenta sincronizar ambas microcomputadoras. Al lograrlo aparece el mensaje "Now Sending PRACT8.FW2", que indica que principia la descarga del archivo de la micro 1. A continuación aparece el mensaje del tiempo aproximado de transferencia ("Approximate Transfer Time"), que para este archivo fue de 2 minutos con 11 segundos. Después aparecen dos renglones de puntos (en total 107 puntos); cada punto indica un bloque de datos transmitido exitosamente. En el caso de que hubiera existido algún error en la transmisión, el Framework II habría retransmitido el bloque en cuestión y habría escrito una "R" o una "T" en la pantalla, lo cual no sucedió en esta prueba. Por último, aparece el mensaje "File Sent", lo cual indica que la transmisión del archivo terminó con éxito. Para la micro 2 casi todos los mensajes son los mismos, con excepción del segundo y el último, que son respectivamente "Now receiving PRACT8.FW2" y "File Received", respectivamente.

Al terminar este proceso se terminó la sesión de telecomunicaciones con el Framework II, seleccionando la opción "End Session" del menú Telecommunications. Finalmente se procedió a guardar en disco los marcos TELENV2 y TELRES2, aunque se debe aclarar que el último marco se perdió por una falla al salvarlo, debido a fallas en la micro 2 receptora.

En la siguiente sección se muestra un listado del marco TELENV2, con los mensajes del Framework II y la información teclada, si bien se corrigió para fines de presentación en este trabajo, sobre todo los errores ortográficos.

3.3.3.2 MARCO TELENV2.

Este es un marco de telecomunicaciones en FRAMEWORK II. Será utilizado para enviar una hoja de cálculo completa y un rango de la misma.

Esta es la micro 1 y es la que se establece en modo de llamada o de envío.

Esta es la segunda prueba de telecomunicaciones y se efectúa el V/23/JUN/89.

El primer paso es establecer este marco en modo de telecomunicaciones, con la opción de enviar.

ATSO=1

Welcome to Framework Telecommunications

Hi! This is XXXX's PC.

No such file

No such file

El archivo que se envía es A:\FRAME\CURSO\PRACT8.FW2

Attempting to Synchronize

Transmission Aborted

NO SE PUDO ENVIAR EL ARCHIVO.

CCCCSSSS

Attempting to Synchronize

Now Sending PRACT8.FW2

Approximate Transfer Time: 2 minutes : 11 seconds

.....
.....
File Sent

El siguiente paso fue con las siguientes opciones:

- 1.- En primer lugar se seleccionó el menú Apps.
- 2.- Después se escogió la opción Telecommunications y dentro de ésta la opción File Transfer.
- 3.- A continuación se seleccionó la opción Send a File, y se dio el nombre del archivo a enviar: A:\FRAME\CURSO\PRACT8.FW2. Esto se estableció para la micro 1 (la emisora).
- 4.- Para la micro 2 se siguieron los mismos pasos:
 - a) Selección de Telecommunications.
 - b) Selección de File Transfer.

c) Se seleccionó Receive a File, y se le dio el nombre A:RACTS.DUP.

5.- Con estas condiciones establecidas sí se pudo enviar el archivo desde la micro 1 y también se pudo recibir en la micro 2.

6.- Al finalizar se dio la opción End Session en ambas micros y se respaldó los marcos generados, el TELENV2 y el TELRES2.

3.3.4 TERCERA PRUEBA.- USO DEL BATCH XMODEM.

3.3.4.1 DESCRIPCION.

Los objetivos principales de esta tercera prueba de comunicaciones con el Framework II, fueron: a) Transmitir algunas líneas de texto; b) Transmitir un primer conjunto de 6 archivos generados con el Framework II, todos con la extensión común FW2; c) Transmitir un segundo conjunto de 10 archivos generados con el LOTUS 1-2-3 (V. 2.01), todos con la extensión común WK1. Los archivos se transfirieron con el protocolo Batch XMODEM. Los pasos fueron los siguientes:

1.- En primer lugar se conectaron ambas microcomputadoras, como se describe en la sección 3.3.1.2.

2.- A continuación se crearon los marcos de telecomunicaciones, como se indica en la sección 3.3.1.3. Para la microcomputadora emisora el nombre del marco fue TELENV3, mientras que para la microcomputadora receptora el nombre fue TELRES3.

3.- En tercer lugar se procedió a establecer la conexión entre las dos microcomputadoras, tal como se describe en la sección 3.3.1.4.

Los 6 archivos seleccionados para transmitirse se generaron en un curso previo de Framework II y en las pruebas anteriores de comunicaciones. Igualmente los 10 archivos del segundo conjunto se generaron en un curso previo de LOTUS 1-2-3.

En la siguiente sección se muestra el marco TELENV3 de la microcomputadora emisora, mientras que en la sección que la sigue se muestra el marco TELRES3 de la microcomputadora receptora.

En el marco TELENV3 se puede observar que los primeros tres mensajes ("AT DTEnter Phone Number of Remote PC", "XXXXX's PC is calling" y "CCCCCCCCCCCCCCCCCCCCCCCC") son diferentes a los de las otras dos pruebas anteriores. Esto se debió a que se realizó una simulación de llamar a un número telefónico, utilizando algunas opciones del menú Setup.

Al quedar establecida la comunicación con el Framework II, se procedió a transmitir el primer conjunto de 6 archivos con la extensión FW2, según se muestra en el marco TELENV3 de la sección siguiente. Para ello, en primer término se seleccionó el menú de Aplicaciones con Ctrl-A, escogiendo la opción de Telecommunications y dentro de ésta la opción File Transfer. En este punto, para la micro 1 emisora se seleccionó la opción Batch XMODEM y después Send a File, dando el nombre genérico del primer conjunto de archivos a enviar (*.FW2). Para la micro 2 receptora se escogió la opción Receive a File, utilizando los mismos nombres de los archivos a transferirse.

El primer renglón que aparece correcto es "Attempting to Synchronize", que indica que el Framework II intenta sincronizar ambas microcomputadoras. Al lograrlo aparece el mensaje "Now Sending TELERESP.FW2 - Using CRC", que indica que principia la descarga del primer archivo del primer lote desde la micro 1. A continuación aparece el mensaje del tiempo aproximado de transferencia ("Approximate Transfer Time"), que para este archivo fue de 0 minutos con 2 segundos. Después aparecen dos puntos; cada punto indica un bloque de datos transmitido exitosamente. Por último, aparece el mensaje "File Sent", lo cual indica que la transmisión del archivo terminó con éxito. Para la micro 2 casi todos los mensajes son los mismos, según se observa en el marco TELRES3, con excepción del segundo y el último, que son respectivamente "Now receiving TELERESP.FW2 - Using CRC" y "File Received", respectivamente. De la misma forma se transfirieron los archivos TELEPC, PRUDUP, TELECOMM, TELENV2 y DOS_ACCE, todos con la misma extensión FW2. Al terminar el último aparece el mensaje "All files received" en el marco TELRES3 de la micro 2 receptora.

Al terminar la transmisión del primer lote hubo un error, por lo cual se observan los mensajes "Transmission aborted" y "CC", por lo cual se tuvo que reanudar la transmisión. Enseguida se procedió a enviar el segundo lote de archivos, colocando los caracteres *.WK1 en la opción Send a File y dejando las demás opciones sin cambio, para transferirlos con el mismo protocolo Batch XMODEM. De esta forma se transmitieron los 10 archivos siguientes, todos con la extensión WK1: MONEDAS, RAUL, COSPOL, R, CE_FQ89, MARCO, MATERIAL, REGRESIO, REGRELIN y REGREMIN.

Una vez enviado el segundo conjunto de archivos, se procedió a respaldar ambos marcos de telecomunicaciones (TELENV3 y TELRES3) para no perderlos. A continuación se volvió a establecer la conexión con el Framework II y a documentar esta tercera prueba, por lo que se observan nuevamente los mensajes de bienvenida ("ATS0=1", etc.).

Al terminar este proceso se terminó la sesión de telecomunicaciones con el Framework II, seleccionando la opción "End Session" del menú Telecommunications. Finalmente se procedió a guardar en disco los marcos TELENV3 y TELRES3.

En la siguiente sección se muestra un listado del marco TELENV3, con los mensajes del Framework II y la información teclada, si bien se corrigió para fines de presentación en este trabajo, sobre todo los errores ortográficos. Igualmente se muestra en la sección que lo sigue el marco TELRES3, de la micro 2 receptora, corregido también para presentación.

3.3.4.2 MARCO TELENV3.

AT DTEnter Phone Number of Remote PC

XXXXX's PC is calling

CCCCCCCCCCCCCCCCCCCCCCCC

```

Attempting to Synchronize
Now Sending TELERESP.FW2           - using CRC
Approximate Transfer Time: 0 minutes : 2 seconds
..
File Sent
Attempting to Synchronize
Now Sending TELEPC.FW2             - using CRC
Approximate Transfer Time: 0 minutes : 16 seconds
.....
File Sent
Attempting to Synchronize
Now Sending PRUDUP.FW2             - using CRC
Approximate Transfer Time: 0 minutes : 4 seconds
....
File Sent
Attempting to Synchronize
Now Sending TELECOMM.FW2          - using CRC
Approximate Transfer Time: 0 minutes : 3 seconds
...
File Sent
Attempting to Synchronize
Now Sending TELENV2.FW2           - using CRC
Approximate Transfer Time: 0 minutes : 21 seconds
.....
File Sent
Attempting to Synchronize
Now Sending DOS ACCE.FW2          - using CRC
Approximate Transfer Time: 0 minutes : 21 seconds
.....
File Sent

Attempting to Synchronize
Transmission Aborted
CC
Attempting to Synchronize
Now Sending MONEDAS.WK1           - using CRC
Approximate Transfer Time: 2 minutes : 19 seconds
.....
File Sent
Attempting to Synchronize
Now Sending RAUL.WK1              - using CRC
Approximate Transfer Time: 0 minutes : 29 seconds
.....
File Sent
Attempting to Synchronize
Now Sending COSPOL.WK1            - using CRC
Approximate Transfer Time: 2 minutes : 17 seconds
.....

```

File Sent
Attempting to Synchronize

Now Sending R.WK1
Approximate Transfer Time: 2 minutes : - using CRC
31 seconds

.....
File Sent
Attempting to Synchronize

Now Sending CE FQ89.WK1
Approximate Transfer Time: 2 minutes : - using CRC
31 seconds

.....
File Sent
Attempting to Synchronize

Now Sending MARCO.WK1
Approximate Transfer Time: 0 minutes : - using CRC
21 seconds

.....
File Sent
Attempting to Synchronize

Now Sending MATERIAL.WK1
Approximate Transfer Time: 3 minutes : - using CRC
3 seconds

.....
File Sent
Attempting to Synchronize

Now Sending REGRESIO.WK1
Approximate Transfer Time: 2 minutes : - using CRC
31 seconds

.....
File Sent
Attempting to Synchronize

Now Sending REGRELIN.WK1
Approximate Transfer Time: 3 minutes : - using CRC
7 seconds

.....
File Sent
Attempting to Synchronize

Now Sending REGREMIN.WK1
Approximate Transfer Time: 3 minutes : - using CRC
6 seconds

.....
File Sent

ATSO=1

Welcome to Framework Telecommunications

Hi! This is XXXX's PC.

El archivo TELENV3 no se pudo respaldar
Ahora sí se pudo respaldar.

Los pasos seguidos en esta tercera prueba fueron los siguientes.

- 1.- En primer lugar se crearon dos marcos vacíos:
 - a) Para la micro 1 se creó el marco TELENV3 y para la micro 2 se creó el marco TELRES3.
 - b) El marco TELENV3 se usó para que el FRAMEWORK II escribiera los mensajes de la micro 1 emisora.
 - c) El marco TELRES3 se usó para la micro 2 en modo respuesta.
- 2.- Para la micro 1 se escogieron las opciones siguientes:
 - a) Telecommunications.
 - b) File Transfer.
 - c) Send a File.
 - d) El nombre del archivo a enviar fue: *.FW2, con lo cual se enviaron todos los archivos cuya extensión fue FW2. Antes se puso en ON la opción Batch XMODEM.
- 3.- Para la micro 2 en modo de respuesta se seleccionaron las siguientes opciones:
 - a) Telecommunications.
 - b) File Transfer.
 - c) Se colocó en ON la opción Batch XMODEM.
 - d) Se seleccionó la opción Receive a File.
- 4.- Con los pasos anteriores se enviaron todos los archivos con la extensión FW2. Los mensajes del FRAMEWORK II se indican al principio de este marco.
- 5.- El siguiente proceso consistió en enviar con las mismas opciones de telecomunicaciones todos los archivos cuya extensión era WK1, colocando en Send a File *.WK1. Estos archivos fueron generados con el paquete LOTUS 123, Versión 2.01.
- 6.- Todas las pruebas anteriores resultaron correctas y satisfactorias, y fueron realizadas el Ma/27/JUN/89.

3.3.4.3 MARCO TELRES3.

ESTA ES LA TERCERA PRUEBA DE TELECOMUNICACIONES EN FRAMEWORK II, REALIZADA EL Ma/27/JUN/89.

```
Now Receiving TELERESP.FW2           - using CRC
Attempting to Synchronize
..
File Received
```

```
Now Receiving TELEPC .FW2           - using CRC
Attempting to Synchronize
.....
File Received
```

```
Now Receiving PRUDUP .FW2          - using CRC
Attempting to Synchronize
....
```

File Received

Now Receiving TELECOMM.FW2 - using CRC
Attempting to Synchronize

File Received

Now Receiving TELENV2 .FW2 - using CRC
Attempting to Synchronize

File Received

Now Receiving DOS_ACCE.FW2 - using CRC
Attempting to Synchronize

File Received

All files received

Now Receiving MONEDAS .WK1 - using CRC
Attempting to Synchronize

File Received

Now Receiving RAUL .WK1 - using CRC
Attempting to Synchronize

File Received

Now Receiving COSPOL .WK1 - using CRC
Attempting to Synchronize

File Received

Now Receiving R .WK1 - using CRC
Attempting to Synchronize

File Received

Now Receiving CE_FQ89 .WK1 - using CRC
Attempting to Synchronize

File Received

Now Receiving MARCO .WK1 - using CRC
Attempting to Synchronize

File Received

Now Receiving MATERIAL.WK1 - using CRC
Attempting to Synchronize

File Received

Now Receiving MATERIAL.WK1 - using CRC
Attempting to Synchronize

File Received

Now Receiving REGRESIO.WK1 - using CRC
Attempting to Synchronize

.....
File Received

Now Receiving REGRELIN.WK1 - using CRC
Attempting to Synchronize

.....
File Received

Now Receiving REGREMIN.WK1 - using CRC
Attempting to Synchronize

.....
File Received

All files received

Aquí se terminó la transferencia de archivos.

Welcome to Framework Telecommunications

Hi! This is XXXX's PC.

El archivo TELENV3 no se pudo respaldar

Ahora sí se pudo respaldar.

Los pasos seguidos en esta tercera prueba fueron los siguientes.

1.- En primer lugar se crearon dos marcos vacíos:

- a) Para la micro 1 se creó el marco TELENV3 y para la micro 2 se creó el marco TELRES3.
- b) El marco TELENV3 se usó para que el FRAMEWORK II escribiera los mensajes de la micro 1 emisora.
- c) El marco TELRES3 se usó para la micro 2 en modo respuesta.

2.- Para la micro 1 se escogieron las opciones siguientes:

- a) Telecommunications.
- b) File Transfer.
- c) Send a File
- d) El nombre del archivo a enviar fue: *.FW2, con lo cual se enviaron todos los archivos cuya extensión fue FW2. Antes se puso en ON la opción Batch XMODEM.

3.- Para la micro 2 en modo de respuesta se seleccionaron las siguientes opciones:

- a) Telecommunications.
- b) File Transfer.
- c) Se colocó en ON la opción Batch XMODEM.
- d) Se seleccionó la opción Receive a File.

4.- Con los pasos anteriores se enviaron todos los archivos con la extensión FW2. Los mensajes del FRAMEWORK II se indican al principio de este marco.

- 5.- El siguiente proceso consistió en enviar con las mismas opciones de telecomunicaciones todos los archivos cuya extensión era WK1, colocando en Send a File *.WK1. Estos archivos fueron generados con el paquete LOTUS 123, Versión 2.01.
- 6.- Todas las pruebas anteriores resultaron correctas y satisfactorias, y fueron realizadas el Ma/27/JUN/89.

3.4 COMUNICACIONES CON SYMPHONY.

3.4.1 INTRODUCCION.

3.4.1.1 SOFTWARE UTILIZADO.

Con el objeto de efectuar las pruebas de comunicaciones con el paquete integrado Symphony, además del hardware anteriormente descrito, se utilizaron los siguientes elementos de software:

- 1.- El sistema operativo MS-DOS, versión 3.2.
- 2.- El paquete integrado Symphony, Versión 1.1 en inglés.
- 3.- Un archivo creado con el paquete Symphony (PAR891C.WR1), consistente en una base de datos documentada con el procesador de texto y con algunas macros, generada durante un trabajo previo.

3.4.1.2 PREPARACION DEL EQUIPO.

Con el objeto de preparar la conexión entre las dos microcomputadoras, los pasos comunes seguidos para cada una de las pruebas, fueron los siguientes:

- 1.- Conexión física de ambas microcomputadoras con el Cable de Módem Nulo Modificado, descrito en la sección 3.1.1, antes de prender los equipos.
- 2.- Carga del sistema operativo (Ver. 3.2) en ambas microcomputadoras.
- 3.- Carga del paquete integrado Symphony (V. 1.1) en las dos microcomputadoras.

3.4.1.3 ESPECIFICACIONES DE COMUNICACION.

Una vez conectadas físicamente ambas microcomputadoras y cargados tanto el sistema operativo como el Symphony, el siguiente proceso fue crear las hojas de especificaciones de comunicaciones, tanto para la microcomputadora emisora como para la receptora. Para ello, se siguieron los siguientes pasos:

- 1.- Al cargar el paquete Symphony, se entró directamente al ambiente de hoja (Sheet).
- 2.- En segundo término se oprimieron las teclas Alt-F10 juntas y luego la tecla C (de Communications), para abrir la ventana del ambiente de Comunicaciones.

3.- En tercer lugar se oprimió la tecla F10 (tecla de menú) para llamar al menú de comunicaciones.

4.- En el menú de comunicaciones se tecléo S(ettings) para llamar a la hoja de especificaciones y cambiar algunas de las opciones que por defecto (default) tiene el Symphony.

5.- A continuación se procedió a cambiar las opciones de interfase tecléando lo siguiente (sólo la primera letra):

- a) I(nterface) B(auds) 5 [RETURN], para seleccionar una velocidad de 1200 bauds.
- b) I(nterface) L(ength) 2 [RETURN], para establecer la longitud de la palabra en 8 bits.
- c) Las otras opciones de la interfase no se cambiaron.

6.- Enseguida se configuraron algunas opciones de la terminal, de la siguiente forma:

- a) T(erminal) S(creen) F(ull screen), para cambiar la ventana a pantalla completa.
- b) T(erminal) E(cho) Y(es), para establecer la modalidad eco.
- c) T(erminal) L(ine feed) Y(es), para establecer la opción de salto de línea.

7.- Las demás opciones se quedaron en sus valores por default.

8.- Una vez modificada la hoja de especificaciones de comunicaciones, se procedió a respaldarla en dos archivos con nombres diferentes y en dos discos diferentes, puesto que se usaría la misma hoja en las dos microcomputadoras interconectadas. Para ello se tecléo lo siguiente:

- a) N(ame) S(ave) SYMTELE1 [RETURN], para respaldar en el disco 1 a utilizarse en la micro 1 emisora.
- b) N(ame) S(ave) SYMTELR1 [RETURN], para respaldar en el disco 2 a utilizarse en la micro 2 receptora.

El programa Symphony automáticamente le agregó la extensión .CCF, para reconocerlos como archivos de comunicaciones. Lo anterior se hizo para que ambas microcomputadoras (1 y 2) tuviesen las mismas opciones de comunicaciones.

3.4.1.4 CONEXION CON EL SYMPHONY.

Después de haber creado las hojas de especificaciones de comunicaciones, quedó establecida con el Symphony la interconexión entre ambas microcomputadoras y se procedió a realizar las pruebas de comunicaciones. Puesto que ya se tenían experiencias previas con el BASIC y el Framework, se efectuó una sola prueba

transmitiendo algunos renglones de texto y un archivo.

3.4.2 PRIMERA PRUEBA.- TRANSMISION DE TEXTO.

3.4.2.1 DESCRIPCION.

En esta primera parte se procedió a enviar algunas líneas de texto. Para ello, establecida ya la comunicación, se efectuó lo siguiente:

Se transmitieron algunos renglones de texto desde la micro emisora hacia la receptora y viceversa, con el objeto de probar la transmisión de datos entre ambas. El texto transmitido consistió en la fecha de la prueba (L/3/JUL/89) y los pasos realizados para la interconexión, descritos en las secciones 3.4.1.2 y 3.4.1.3.

3.4.3 SEGUNDA PRUEBA.- TRANSMISION XMODEM.

3.4.3.1 DESCRIPCION.

En la segunda parte de la prueba se procedió a enviar un archivo consistente en una base de datos preparada en un trabajo previo (PAR891C.WR1). Para ello, establecida ya la comunicación, se efectuó lo siguiente:

1.- Primero se oprimió la tecla F10 para acceder el menú, en ambas microcomputadoras.

2.- Después se tecléo F(ile-transfer), en las dos micros, para establecer la transferencia de archivos.

3.- Enseguida se tecléo:

a) S (end) en la micro emisora, para enviar el archivo.

b) R (eceive) en la micro receptora, para recibirlo.

4.- A continuación se escribió el nombre del archivo, usando el mismo nombre (PAR891C.WR1) para ambas microcomputadoras.

5.- En la parte superior de la pantalla apareció:

a) En el primer renglón, el mensaje "Intent of conec-tion" en ambas computadoras, empezando a continuación la transmisión del archivo de la emisora a la receptora.

b) En el segundo renglón de la micro emisora apareció el número de bytes enviados (variando en múltiplos de 128 caracteres), así como el número total de bytes del archivo (41997 bytes) y el número de errores corregidos (xxxx de 41997 bytes send, 0 errors corrected).

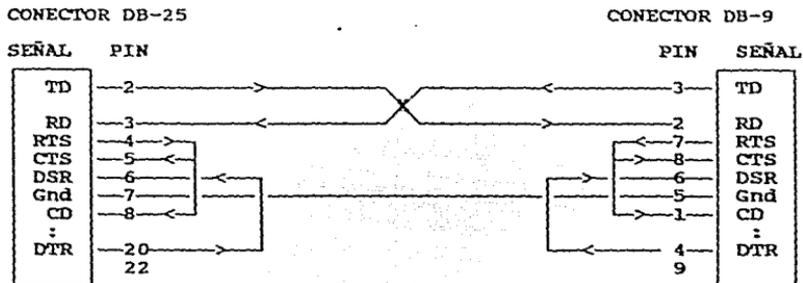
c) En el segundo renglón de la micro receptora apareció el número de bytes recibidos (variando en múltiplos de 128 caracteres) y el número de errores corregidos (xxxx bytes received, 0 errors corrected).

4 CAPITULO III. CONCLUSIONES.

4.1 CONCLUSIONES CON RESPECTO AL HARDWARE.

4.1.1 CABLES Y CONECTORES.

En las sección 3.1.1. se especificó la construcción del Cable de Módem Nulo Modificado y su esquema correspondiente, utilizando conectores DB25. Puesto que las Microcomputadoras más recientes vienen equipadas generalmente con dos puertos seriales, uno de 25 pines (Conector DB25) y otro de 9 pines (Conector DB9), es de gran utilidad conocer los pines equivalentes de ambos conectores. A continuación se muestra esta equivalencia:



Cable de Módem Nulo Modificado.
(Conexión DB25 a DB9).

Por otra parte, si el usuario no desea construir sus propios cables, siempre quedará la opción de comprarlos o mandarlos a hacer con los diagramas presentados en el presente trabajo.

4.1.2 INTERCONEXION DE EQUIPOS.

Si bien en el presente trabajo únicamente se probaron dos Microcomputadoras con microprocesador 8088, se pueden efectuar con otros equipos. Mientras más grandes sean, será más sencilla y rápida la transmisión, como en el caso de microprocesadores 80286, 80386 y 80486, puesto que además tienen discos duros con tiempos de acceso menores por lo cual son más rápidas.

Por otra parte, también se pueden interconectar micros más pequeñas como las Apple. El autor del presente trabajo ha interconectado Microcomputadoras Apple (Apple II-E y Apple II-C) a una micro Printaform, si bien la construcción del conector de la Apple fue diferente por tener 5 pines (tipo D). Además se consideró a la micro Printaform como si fuese una impresora serial.

4.2 CONCLUSIONES CON RESPECTO AL GWBASIC.

4.2.1 VENTAJAS.

La principal ventaja de realizar comunicaciones con el lenguaje BASIC es su disponibilidad. Casi todas las microcomputadoras lo tienen, ya que generalmente viene con el sistema operativo (MS-DOS, por lo menos).

Por otra parte, existen varios programas para comunicaciones escritos en BASIC, por lo que se puede escoger entre ellos. Además se tiene más control sobre la transmisión por manejarse cadenas de caracteres ASCII.

4.2.2 DESVENTAJAS.

La principal desventaja de las comunicaciones con el BASIC es su dificultad para transmitir archivos binarios, ya que puede llegar a detener los equipos o destruir otros archivos si no se toman algunas precauciones, como sucedió en la tercera prueba del presente trabajo.

4.3 CONCLUSIONES CON RESPECTO AL FRAMEWORK.

4.3.1 VENTAJAS.

Las principales ventajas de la intercomunicación entre microcomputadoras utilizando el paquete integrado Framework, se indican a continuación:

1.- Las especificaciones de comunicaciones se pueden variar a voluntad mediante menús de fácil acceso y comprensión. En caso de duda se tiene ayuda integrada en el paquete (tecla F1).

2.- Los resultados de las pruebas son fácilmente almacenados en marcos, que después se almacenan como archivos independientes o como ventanas de un Esquema para consultas posteriores. Además se pueden imprimir para análisis más detallados.

3.- Por cable directo aceptan la máxima velocidad del puerto serial, acelerándose de este modo la transmisión, sobre todo de grupos de archivos.

4.- El protocolo Batch XMODEM del Framework es muy poderoso y confiable, ya que acepta las opciones comodín (*.*). De este modo se pueden transmitir todos los archivos de un subdirectorio, como lo hizo el autor del presente trabajo el 1º de mayo de 1991 para transferir todos los archivos de un disco duro de 30 Mega-bytes de una microcomputadora XT Printaform (8088 de 4.77 MHz) hacia un disco duro de 80 Mb vacío de una microcomputadora Pine 80386-SX de 16 MHz recién instalada.

5.- Existen otros protocolos de comunicación entre los que se puede escoger, algunos de los cuales se indican a continuación: XON/XOFF, XMODEM, YMODEM, Batch XMODEM, etc..

6.- Puede emular otros dispositivos, como en el caso descrito de la Apple a la Printaform (Sección 4.1.2) en el cual la

microcomputadora Apple (II-E ó II-C) transmitía la información a una impresora serial, siendo ésta la micro Printaform que recibía los datos en marcos establecidos con el Framework (II y III) y posteriormente se grababan en disco para procesos ulteriores.

4.3.2 DESVENTAJAS.

Tal vez la única desventaja del Framework sea su adquisición, por ser aplicaciones costosas; pero una vez que se tiene, la intercomunicación entre computadoras es de lo más sencillo. Además por ser paquete integrado se pueden corregir los marcos para su mejor comprensión, documentación y respaldo.

4.4 CONCLUSIONES CON RESPECTO AL SYMPHONY.

4.4.1 VENTAJAS.

Las principales ventajas de la intercomunicación entre microcomputadoras utilizando el paquete integrado Symphony, se indican a continuación:

1.- Las especificaciones de comunicaciones se pueden variar a voluntad mediante menús de fácil acceso y comprensión. En caso de duda se tiene ayuda integrada en el paquete (tecla F1).

2.- Los resultados de las pruebas son almacenados en ventananas (aunque no fácilmente) que después se almacenan como archivos independientes para consultas posteriores. Además se pueden imprimir para análisis más detallados.

3.- Por cable directo aceptan la máxima velocidad del puerto serial, acelerándose de este modo la transmisión, sobre todo de grandes archivos.

4.4.2 DESVENTAJAS.

Tal vez una desventaja del Symphony sea su adquisición, por ser una aplicación costosa; pero una vez que se tiene, la intercomunicación entre computadoras no es difícil. Además por ser paquete integrado se pueden corregir los marcos para su mejor comprensión, documentación y respaldo. Otra desventaja, al menos en la versión 1.1 probada en este trabajo, es que no acepta el protocolo Batch XMODEM como en el Framework, siendo muy difícil transferir grupos de archivos con nombres y extensiones comodín (*.*). Tal vez sea posible en otras versiones.

4.5 CONCLUSIONES FINALES.

4.5.1 LA MEJOR OPCION DE COMUNICACIONES.

De los dos Paquetes Integrados y el lenguaje GW-BASIC que se probaron en el presente trabajo, el más sencillo, poderoso y confiable, lo fué el Framework. Desde las primeras pruebas ha sido utilizado repetidamente por el autor del presente trabajo.

De esta manera se utilizó nuevamente el 11 de mayo de 1992, transfiriendo archivos con el protocolo Batch XMODEM del Framework III, versión 1.0 en español. Se transmitieron todos los archivos de dos subdirectorios de un disco duro de 40 Megabytes de una microcomputadora AT Printaform (modificada con un motherboard M-209 de 20 MHz y microprocesador 80286) usando el puerto 1, hacia un disco duro de 80 Mb vacío de una microcomputadora Gama AT-286, al través del puerto 2. La velocidad de transmisión se fijó en 19,200 bauds y los resultados se grabaron en disco.

4.5.2 OTROS PAQUETES DE COMUNICACIONES.

Si bien en el presente trabajo sólo se utilizaron para comunicaciones el lenguaje GW-BASIC y los paquetes integrados Framework y Symphony, ésto no implica que sean los únicos.

De este modo han aparecido otros paquetes como los siguientes: PCTools (Versiones 6.0 y 7.0), Excel (V. 3.0), Works (V. 2.0), Organizer Link II (V. 1.0), etc. Además se tienen otros paquetes para intercambiar información con grandes computadoras. Con todos ellos se pueden intercomunicar computadoras, lo cual será más sencillo en el futuro.

Como se indicó al principio del presente trabajo, el objetivo principal fue que cualquier usuario con mínimos conocimientos de computación pudiese efectuar pruebas de comunicaciones. Si es de utilidad para algún lector, este trabajo habrá cumplido sus fines.

5 APENDICE A. GLOSARIO Y TERMINOLOGIA.

5.1 GLOSARIO Y TERMINOS USADOS.

Adaptador. Cualquier tarjeta de circuito impreso que se coloca en alguna ranura (slot) de la tarjeta maestra (mother board).

ALGOL (ALGOrithmic Language). Un lenguaje de alto nivel usado para programar aplicaciones de carácter matemático. Desde 1968 se usó en las computadoras Burroughs (B-5500, B-6500) de los varios centros de cómputo de la UNAM (CIMASS, Dirección de Cómputo Académico).

ASCII (American national Standard Code for Information Interchange). Un código de 7 bits usado para transmitir datos dentro y entre computadoras. El código estándar define 128 (2⁷) caracteres gráficos y de control, los cuales pueden ser representados por los números decimales 0-127.

ASCII extendido. Ver Conjunto extendido de caracteres.

Archivo. Un conjunto de información relacionada. Un archivo en disco contiene información. Los programas también se guardan en archivos.

BASIC (Beginner's All-Purpose Symbolic Instruction Code). Un lenguaje de programación de propósito general, diseñado para ser fácil de aprender, recordar y usar. Se caracteriza por construcciones simples, un mínimo de reglas especiales y está orientado al uso interactivo.

BASICA. Lenguaje BASIC Avanzado, usado por IBM en sus computadoras personales.

Batch XMODEM. Es el protocolo XMODEM modificado para aceptar las opciones "comodín" (*.*). Con el se pueden transferir todos los archivos de un subdirectorio de una microcomputadora emisora a una receptora, como se hizo en el presente trabajo en una de las pruebas usando el paquete Framework.

Baud. Una unidad la cual define el número de veces que una señal de datos cambia de estado cada segundo. A pesar del uso popular, no es sinónimo de "bits por segundo", aunque las dos son prácticamente iguales en aplicaciones de comunicación entre PC's.

BELL-2. Lenguaje interpretador desarrollado por el laboratorio de Bell & Telephone en la década de los 50's y usado en la computadora IBM-650 del Centro de Cálculo Electrónico de la UNAM en 1959-1965.

BIOS (Basic Input/Output System). Un conjunto de programas rutinarios escritos para manipular varias funciones rudimentarias de entrada/salida. Los archivos ocultos IBMBIO.COM (de IBM) e IO.SYS (de Microsoft) contienen adiciones y cambios al BIOS grabado en ROM de la tarjeta maestra del sistema. El término "basic" no se debe confundir el lenguaje del mismo nombre.

Bit (BINary digiT). Acrónimo de dígito binario. En notación binaria, el incremento más pequeño de datos, representado por 0 ó 1.

Boot. Cuando una computadora personal se prende, un pequeño programa construido en el hardware del sistema se usa para cargar un programa ligeramente más grande desde el disco flexible o desde la partición principal del DOS del disco duro. Este programa carga otro todavía más grande a medida que el sistema va arrancando. En español se considera sinónimo de "arranque" o "reinicio". De esta forma "dar boot" al sistema significa iniciar este proceso, ya sea conectando el switch de corriente ("arrancar en frío"), o si ya está encendida, manteniendo oprimidas las teclas Control (Ctrl) y Alternate (Alt) juntas y oprimiendo la tecla Delete (Del) ("arranque en caliente" o "reinicio"). Esta combinación de teclas usualmente se abrevia como Ctrl+Alt+Del.

Buffer. Un área en la memoria de la computadora utilizada por el MS-DOS para almacenar información.

Byte. Un carácter binario consistente en 8 bits, usualmente operado (almacenado o recuperado) como una sola unidad de información.

C. Lenguaje de alto nivel, muy usado en la elaboración de paquetes de software por ser casi independiente del equipo a utilizar.

Cable de módem. Un cable generalmente de 10 conductores utilizado para conectar una microcomputadora (DTE) a un módem (DCE).

Cable de módem nulo. Un cable de 3 conductores usado para conectar dos microcomputadoras (DTE's) y transmitir información entre ellas.

Cable de módem nulo modificado. Un cable de por lo menos 3 conductores utilizado para interconectar dos microcomputadoras, en el cual se puentean algunos pines de los conectores DB25 o DB9, con el objeto de "engañar" algunos paquetes de software que usan dichas señales. Es el que se utilizó en el presente trabajo.

Carácter. Una letra, número o símbolo que se presiona en el teclado o se ve en la pantalla. En el código ASCII estándar, un carácter gráfico puede ser un signo de puntuación, los dígitos 0-9 y las letras A-Z mayúsculas y minúsculas del alfabeto inglés; en cambio un carácter oculto representa un código de control y generalmente no se puede observar en pantalla.

CD-ROM. Un dispositivo conectado a una microcomputadora y que se usa para leer información grabada en discos compactos.

Checksum. En comunicación de datos, es un número calculado para verificar que el segmento de datos de 128 bytes recibidos en una microcomputadora sea igual al enviado por otra y es la suma de los valores hexadecimales de todos los 128 caracteres (bytes) transmitidos. Se utiliza en algunos protocolos, como en el XMO-DEM. A veces se le llama incorrectamente "dígito verificador".

Cinta perforada. Una cinta continua de papel usada en los antiguos teletipos conectados a las primeras minicomputadoras, que se perforaba para almacenar información.

Cinta magnética. Una cinta continua de plástico (Mylar) recubierta con material magnetizable, generalmente de 0.75", y utilizada para almacenar grandes cantidades de información. Muy usada en grandes computadoras; en las microcomputadoras se usa como respaldo (backup) de discos y generalmente son cassettes más pequeños.

COBOL (Common Business Oriented Language). Lenguaje de alto nivel utilizado para manipular grandes cantidades de información. Se utilizó desde la década de los '60 (en México desde 1963, en la computadora IBM 1401, de la UNAM), generalmente en bancos y empresas públicas y privadas. Actualmente existen versiones para microcomputadoras.

Comando. Un programa corto que le dice al MS-DOS cómo realizar una tarea específica.

Comodín (Wild Card). Un asterisco o signo de cerrar interrogación, usado para significar cualquier palabra o cualquier carácter, respectivamente. Por ejemplo, la expresión DIR *.BAK desplegará un directorio de todos los archivos con la extensión BAK, mientras que DIR A????? mostrará todos los archivos de 6 ó menos caracteres que principien con la letra "A".

Compilador. Es un lenguaje que sirve para traducir programas al lenguaje-máquina de las computadoras. Ejemplos de compiladores son el FORTRAN, BASIC compilado, etc..

Computadora. En general, una computadora es un dispositivo electrónico-mecánico capaz de procesar automáticamente la información representada por impulsos eléctricos, mediante un programa almacenado. Generalmente se clasifican, según el tipo de datos que procesan, en analógicas, digitales e híbridas.

Computadora analógica. Es un dispositivo que maneja la información en forma de cantidades continuas que se miden físicamente; los datos que se obtienen por medición son datos continuos.

Computadora digital. Es un dispositivo que maneja la información en forma de estados físicos, los cuales se representan mediante un código especial denominado código binario; los datos que se procesan se llaman datos discretos y se obtienen por conteo.

Computadora híbrida. Cuando se combinan las características de las computadoras analógicas y las digitales resultan los sistemas híbridos.

Conector. En general es un conjunto de enchufes usado para conectar cualquier dispositivo interno o externo a una computadora. Existen de diferentes tipos (de "peine", de pines, etc.). Algunos tienen "género" (macho o hembra), como los DB-9 y DB-25 muy usados en comunicación de datos.

Conector DB-9. Es un conector en forma de "D" alargada con 5 pines arriba y 4 abajo. Se usa para conectar dispositivos seriales (generalmente ratones) a una microcomputadora.

Conector DB-25. Es un conector en forma de "D" alargada con 13 pines arriba y 12 abajo. Se utiliza para conectar dispositivos periféricos (impresoras, módems, ratones, FAX, etc.) a las computadoras.

Conjunto extendido de caracteres. Un código de 8 bits de 256 (2⁸) caracteres usado en microcomputadoras para ensanchar el código ASCII estándar con 128 caracteres adicionales, consistentes en letras de idiomas ajenos al inglés (como las cinco vocales acentuadas y la letra Ñ del español, algunas letras griegas, etc.), símbolos gráficos (generalmente matemáticos como þ, ¸, ¨, etc.) y otros caracteres no encontrados en el código ASCII estándar. A veces se le llama incorrectamente "ASCII extendido".

CPU (Central Process Unit). Nombre dado a la parte de las computadoras que se encarga de las operaciones principales de cálculo. Aunque el término se usa algunas veces como sinónimo de microprocesador, la unidad central de proceso comprende al microprocesador, al coprocesador matemático (si lo hay) y todos los circuitos electrónicos asociados. Para complicar todavía más la nomenclatura, el microprocesador Intel i486 incorpora un coprocesador matemático 80387 en el mismo paquete, así como un caché de 8 Kb, posiblemente conjuntando un microprocesador y una CPU.

Dirección. La localidad lógica en la cual se puede localizar un byte. Por ejemplo, los primeros cien bytes de RAM se encuentran en las localidades 0-99.

Disco duro. Un dispositivo de almacenamiento en el cual el medio de almacenamiento es un disco rígido de aluminio cubierto con un material magnetizable y que está instalado en forma permanente.

Disco flexible (floppy disk). Término usado para nombrar a los discos magnéticos, generalmente de 5" o de 3" pulgadas.

Disco magnético. Un disco magnético flexible, encerrado permanentemente en una cubierta protectora de plástico duro o de PVC. Los más conocidos son de 5" y 3" pulgadas, en baja (360 Kb y 720 Kb) y alta densidad (1.2 Mb y 1.44 Mb).

Dispositivo. Un equipo que realiza una función específica. Una impresora es un ejemplo de dispositivo.

DOS (Sistema Operativo en Disco). Un grupo de programas que actúan como traductores entre el usuario y la computadora. El MS-DOS es un sistema operativo en disco de MicroSoft Corporation.

EBCDIC (Extended Binary-Coded Decimal Interchange Code). Un código de 8 bits usado para intercambiar datos dentro y entre computadoras. Fué desarrollado por IBM para su sistema 360.

EISA (Extended Industrial Standard Architecture). Un sistema propuesto como una versión actualizada de la arquitectura ISA de las microcomputadoras originales de IBM (PC, XT y AT).

Ensamblador. Es un lenguaje de bajo nivel para elaborar programas de computadoras que dá un completo dominio sobre el equipo.

Equipo de comunicación de datos (DCE, Data Communication Equipment). Un equipo usado para convertir las señales digitales de una computadora a las señales analógicas de las líneas de transmisión (generalmente telefónicas) y que se utiliza para interconectar computadoras. El caso más usual es el módem.

Equipo terminal de datos (DTE, Data Terminal Equipment). Se dá este nombre a una microcomputadora personal cuando se usa en aplicaciones de comunicaciones.

Error de paridad. Una condición en la cual la paridad par o impar no se mantiene, indicando que probablemente el dato almacenado sea incorrecto.

FORTRAN (FORMula TRANslator). Lenguaje de alto nivel muy parecido a la notación algebraica y con palabras en inglés, que apareció a mediados de la década de los '60 y se utilizó en la computadora de segunda generación IBM-1620. En el año de 1966 se instaló una de éstas en la Facultad de Química de la UNAM.

Framework. Un paquete integrado que abarca aplicaciones de proceso de texto, hojas de cálculo, bases de datos, graficación, comunicaciones, esquemas dinámicos y un lenguaje de programación llamado FRED. Existen las versiones I, II y III, tanto en inglés como en español, y fué desarrollado por la compañía Ashton-Tate.

Graficador. Un dispositivo de salida que se usa para elaborar gráficas, dibujos y texto, en diferentes colores y tamaños. Las plumillas son intercambiables y algunos elaboran planos de más de un metro cuadrado.

Gw-Basic. Lenguaje de computadora de propósito general. Frecuentemente, el Basic (o GW-Basic) es el primer lenguaje que las personas aprenden.

Hardware. El equipo que constituye un sistema de computación. No se debe confundir con los programas o Software.

I/O (Input/Output, Entrada/Salida). Términos utilizados para los dispositivos conectados a las computadoras, existiendo dispositivos de entrada (teclado, ratón), de salida (impresoras, graficadores) o de ambos (discos, cintas). También se refieren a operaciones en los lenguajes de programación.

Impresora. Un dispositivo de salida utilizado para obtener en papel (o acetatos, sobres, etc.) la información interna de las computadoras y sus periféricos.

Interfase RS-232C. Es una interfase eléctrica estándar definida por la Electronic Industries Association (EIA) en octubre de 1969 y se utiliza para conectar diferentes componentes periféricos a un sistema de computación, como módems, impresoras seriales, ratones y computadoras. Las iniciales "RS" significan "Recommended Standard" (Estándar Recomendado); "232" es el número del estándar y "C" es el nivel de revisión. Es decir, RS-232C es la tercera revisión del estándar básico, existiendo 13 revisiones

desde la "A" hasta la "M" que tratan de cubrir todos los requerimientos de sistemas definidos.

Kilobyte (Kb). La letra mayúscula "K" equivale a 2^{10} , es decir 1 024. Por lo tanto 1 Kb representa 1 024 bytes, mientras que 64 Kb son realmente 65 536 bytes y no 64 000.

ISA (Industrial Standard Architecture). La arquitectura del sistema introducida en la primera computadora personal (PC) de IBM y posteriormente usada las XT y en las AT.

Lenguaje. En computación, es un conjunto de instrucciones que se aplican a ciertos elementos (caracteres, operadores, etc.) que siguen una serie de reglas de sintaxis.

Lenguaje de programación. Es un lenguaje que se utiliza para elaborar programas de computadoras. Se clasifican en lenguajes de bajo y alto nivel, según sus objetivos.

Lenguaje-máquina. Es aquel lenguaje que únicamente es comprendido por las computadoras, ya que se encuentra en código binario. Se puede interpretar, pero es sumamente difícil.

Megabyte (Mb). La letra mayúscula "M" equivale a 2^{20} , es decir 1 048 576. Por lo tanto 1 Mb representa 1 048 576 bytes, mientras que 60 Mb son realmente 62 914 560 bytes (60 x 1 048 576 y no 60 000 000).

Memoria. La parte activa del almacenamiento en la computadora, utilizada por ésta cuando ejecuta un programa o un comando.

Microcomputadora. Es el sistema más pequeño de propósito general capaz de ejecutar instrucciones de un programa almacenado para efectuar una amplia variedad de tareas. Un sistema de microcomputadora tiene todos los elementos funcionales que se encuentran en cualquier sistema grande. Esto es, está organizado con las siguientes unidades: de almacenamiento, aritmético-lógica, de control y la de entrada/salida.

Microprocesador. Es un "chip" que lleva a cabo las funciones de lógica, de aritmética y de control en una microcomputadora.

Minicomputadora. Es una computadora intermedia entre una microcomputadora (PC) y una gran computadora (Mainframe).

MODEM (MODulator/DEMODulator). Un dispositivo de hardware que actúa como interfase entre una computadora y la línea telefónica, usado para comunicación de datos entre dos computadoras. Se llama así debido a que modula la señal digital transmitida y demodula la señal analógica recibida.

Monitor. Un dispositivo que consiste en un tubo de rayos catódicos, usado como la interfase visual entre una microcomputadora y el usuario.

Nombre de archivo. Un nombre de archivo puede tener de 1 a 8 caracteres de longitud y puede tener una extensión de hasta tres caracteres, separados del nombre por un punto.

Nombre de unidad. Consiste en una letra de unidad y dos puntos. El nombre de la unidad le dice al MS-DOS en que unidad buscar un archivo.

Palabra. Es un término usado en comunicación de datos y comprende los 8 bits de datos que forman un carácter, al cual se le agregan dos bits, uno de START y otro de STOP.

Pantalla. La interfase visual entre una microcomputadora y el usuario. Existen pantallas monocromáticas (de color verde, ámbar o blanco), de color (CGA, EGA y VGA), de crystal líquido (LCD), de plasma, etc.

Paralelo. En transmisión de datos, una descripción del método en el cual los ocho bits de un byte de datos se transmiten simultáneamente, requiriendo de esta forma 8 líneas de datos separadas. Así, una impresora paralela requiere 8 conductores, además de otros para diferentes funciones de soporte. Comparar con Serial.

Paridad. En una computadora, la convención de agregar 1 bit a un byte de tal forma que el número total de bytes sea siempre par (paridad par) o impar (paridad impar).

PASCAL. Es un lenguaje de programación de alto nivel de tipo matemático, nombrado así en honor del matemático Blaise Pascal.

Pin. Son aquellas espigas ("patitas") que tienen los conectores; se aplica el término generalmente a los conectores DB-9 y DB-25 de los puertos seriales o paralelos.

POST (Power-On Self Test). La serie de chequeos del sistema que se efectúan cada vez que una computadora personal se enciende.

Programa. Un conjunto de instrucciones, escritas en lenguaje de la computadora, que le indican a la computadora cómo realizar algunas tareas.

Protocolo. Un acuerdo o conjunto de reglas convenidas para establecer comunicaciones entre computadoras.

Puerto serie. Es un adaptador de comunicaciones usado para conectar una microcomputadora a otra por cable directo o a un módem asíncrono, aparato que conecta la PC a la línea telefónica.

Paquete integrado. Un conjunto de programas, generalmente en lenguaje-máquina, que están orientados a procesar en un solo paquete varias aplicaciones que anteriormente se vendían por separado, tales como proceso de textos, hojas electrónicas, bases de datos, graficación, comunicaciones y algún lenguaje de programación. Ejemplo de ellos son el Framework, Symphony, Excel, Works, etc.

RAM (Random-Access Memory, memoria de acceso aleatorio). La porción de memoria en la cual el usuario puede escribir instrucciones o datos. La porción más baja de la RAM se usa para almacenar temporalmente el sistema operativo en disco (DOS) cuando se enciende la microcomputadora.

ROM (Read-Only Memory, memoria sólo de lectura). Es aquella memoria cuyo contenido se graba durante el proceso de manufactura de los chips y no puede ser cambiado por el usuario (excepto con el reemplazo físico del chip de ROM por otro).

RUNCIBLE. Lenguaje compilador de alto nivel usado en la computadora IBM-650 del Centro Electrónico de Cálculo de la UNAM, en el año de 1960. Fué el antepasado del FORTRAN.

Señal (Prompt). Una palabra o símbolo que el MS-DOS muestra en la pantalla para indicar que está listo para que el usuario escriba algo. El símbolo estándar del MS-DOS consiste en la letra de la unidad predeterminada (usualmente A, B, o C) y un signo de mayor que.

Serial. En transmisión de datos, una descripción del método en el cual los 8 bits de un byte de datos se transmiten secuencialmente al través de una sola línea de datos. Un módem es un dispositivo serial, puesto que una línea telefónica transmite una sola señal a la vez. Comparar con Paralelo.

SOAP. Un lenguaje ensamblador usado en la computadora IBM-650, en el año de 1959.

Sistema operativo. Una serie de programas que traducen sus comandos para la computadora, ayudándole a realizar tareas tales como crear archivos, ejecutar programas o imprimir documentos.

Software. Los programas, rutinas o instrucciones que le permiten a la computadora realizar tareas. Algunos ejemplos de software incluyen: Sistemas operativos, programas de procesamiento de palabras y paquetes integrados.

Symphony. Un paquete integrado que abarca aplicaciones de proceso de texto, hojas de cálculo, bases de datos, graficación, comunicaciones y un lenguaje de programación de macros. Existen las versiones 1, 1.1 y 2.0, tanto en inglés como en español, y fué desarrollado por la compañía Lotus Development.

Tarea. Algo que la computadora realiza cuando el usuario suministra un comando. La ejecución de un programa es un ejemplo de tarea.

Tarjeta perforada. Una tarjeta de cartón delgado en la cual se hacían perforaciones redondas o rectangulares con equipos periféricos (perforadoras de tarjetas) para registrar información que después se leía en las computadoras de las primeras generaciones. La más usual era de 80 columnas con 12 renglones, aunque también existían de 90 columnas.

Transmisión de datos. Es el movimiento de información que ya se procesó o que se va a procesar, codificada de alguna forma (generalmente binaria), en algún sistema de transmisión eléctrica o electrónica. Los medios empleados varían en función de la distancia sobre la cual se va a transmitir la información.

Transmisión Half-Duplex. Es cuando en todo el recorrido o parte de él se utiliza un mismo circuito físico de 2 conductores

de una línea telefónica para transmitir información en ambos sentidos, pero no al mismo tiempo.

Transmisión Full-Duplex. Es cuando en las líneas telefónicas de 4 hilos se utilizan canales independientes en todo su recorrido para cada sentido de la transmisión. En ellas se puede enviar información en ambos sentidos al mismo tiempo.

Velocidad. Una interfase serie asíncrona opera a una velocidad fija, y ambos extremos de la línea deben estar a la misma velocidad. En general, para obtener la velocidad en caracteres por segundo basta con dividir entre 10 la velocidad en bits por segundo. Así, una velocidad de 300 bps es equivalente a transmitir 30 caracteres por segundo, lo cual es útil conocer cuando se transfieren archivos y se requiere calcular el tiempo de transmisión. La microcomputadora personal admite un amplio rango de velocidades, pero las más usuales son 300, 1200 y 9600 bps.

Velocidad de modulación. Es el número máximo de veces por segundo que puede cambiar de estado la señalización en línea. Se mide en baudios, siendo un baudio equivalente a un intervalo significativo por segundo.

Velocidad de transmisión en serie. Se define como el número máximo de bits (dígitos binarios) que se transmiten por segundo al través de un determinado circuito de datos. Se mide en bits por segundo. Si la información es tal que a cada estado significativo en línea corresponde un bit de información y se usa un código con bits de igual longitud, el número de bits/seg coincide con el de baudios.

Velocidad de transferencia de datos. Representa la cantidad de información que se puede transmitir por unidad de tiempo. También se define como el promedio de bits, caracteres o bloques por unidad de tiempo que pasan entre dos equipos que pertenecen a un sistema de transmisión de datos.

Unidad de disco. Un dispositivo conectado a la computadora para leer o grabar información en discos flexibles (floppies), discos fijos o duros (hard disks) o discos compactos (CD-ROM).

XMODEM. Es un protocolo de comunicaciones desarrollado por Ward Christensen, para transferir archivos binarios. Las reglas fundamentales del XMODEM básico establecidas desde 1976, son todavía usadas por casi todos los sistemas de boletines (BBS), sin consideración de la computadora a la cual estén conectados. El protocolo XMODEM requiere ciertas especificaciones de comunicaciones: palabra de ocho bits de longitud, sin paridad, un bit de inicio y un bit de parada. Algunos programas de terminales colocan estos parámetros automáticamente durante la transferencia con el XMODEM, mientras que otros requieren colocarlos manualmente.

XON/XOFF. Es un protocolo de comunicaciones en el cual la secuencia Ctrl-S es la señal XOFF para detener la transmisión, mientras que la secuencia Ctrl-Q es la señal XON para continuar la transmisión.

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

6 BIBLIOGRAFIA.

6.1 BIBLIOGRAFIA.

Amozorrutia de Maria y Campos, José, "Curso de LOTUS 1-2-3", Apuntes y Material Computacional, Facultad de Química, Junio de 1989.

Awad, Elías M., "Proceso de Datos en los Negocios", Trad. José Meza Nieto Segunda edición, Quinta impresión, Edit.. Diana, S. A., Septiembre de 1982.

Bates, Williams, "The Computer Cookbook", 1983/84 Edition, Edit. Prentice-Hall, Inc., Englewood Cliffs, New Jersey 07632, U. S. A., 1983.

Columbia Data Products, "Manual de BASIC, Versión 2.0", Editado por Columbia Data Products, Inc., 1984.

Glossbrenner, Alfred, "The Fine Art of Data Transfer", Revista POPULAR COMPUTING, Volumen 4, Número 11, Septiembre de 1985, Editorial McGraw-Hill Publication.

Harrison, Bill, "Introducción al Framework II", Traducción de Antonio Loris Ruiz y Antonio Vaquero Sánchez, Editorial McGraw-Hill, México, 1987.

Kruglinski, David, "Guía a las Comunicaciones del IBM/PC", Traducción de Luis Hernández Yáñez y Antonio Vaquero Sánchez, Editorial McGraw-Hill, México, 1985.

Levitan, Arlan R., "Telecomputing Today: Inside XMODEM", Revista COMPUTE!, publicación 61, volumen 7, número 6, junio de 1985, editorial COMPUTE! Publications Inc. abc.

Moreno Padilla, Caritino, "Curso de Framework II", Apuntes y Material Computacional, Facultad de Química, Febrero de 1989.

Salazar Selvas, Edith del Carmen, "Diseño y Desarrollo de un Conmutador Lógico Inteligente para la Interconexión e Intercomunicación entre Computadoras, utilizando una Computadora Personal y un Paquete de Comunicación de Datos", Tesis, Facultad de Ingeniería, UNAM, México, Enero de 1989.

Sanders, Donald H., "Informática: Presente y futuro", Editorial McGraw-Hill, México, 1985.

Woram, John, "The PC Configuration Handbook: A Complete Guide To Troubleshooting, Enhancing And Maintaining Your PC", 2nd Edition, Editorial Bantam Books, USA, 1990.