ESCUELA POLITECNICA NACIONAL

FACULTAD DE INGENIERIA ELECTRICA Y ELECTRONICA

TESIS DE GRADO

CONVERSION DE UN COMPUTADOR PERSONAL EN UN TERMINAL INTELIGENTE

TESIS PREVIA A LA OBTENCION DEL TITULO DE

INGENIERO EN

ELECTRONICA Y TELECOMUNICACIONES

ALBERTO ANDRADE VAREA

JUNIO 1987

CERTIFICO QUE LA PRESENTE TESIS
HA SIDO ELABORADA EN SU TOTALIDAD
POR EL SR. ALBERTO ANDRADE VAREA.

ING. JAIME VELARDE

CERTIFICO QUE LA PRESENTE TESIS
HA SIDO ELABORADA EN SU TOTALIDAD
POR EL SR. ALBERTO ANDRADE VAREA.

ING. JAIME VELARDE

INDICE

PAGINA

CAPITULO V

CAPITULO VI

APENDICES

1.1.1 INTRODUCCION

En este capítulo se describirá un sistema transaccional de procesamiento de datos basado en equipos marca NCR.

En general, una transacción se entiende como un evento independiente de otros que se realiza mediante un intercambio de información entre un terminal y un computador central.

Un sistema transaccional está orientado a procesar estos eventos o transacciones como un proceso completo, es decir como si este evento es el único o el último a procesarse.

Una transacción la inicia normalmente el operador del terminal, quien digita los datos requeridos los mismos que se transmiten al computador central dentro de un mensaje. Este mensaje es recibido, validado, identificado, procesado y finalmente aceptado o rechazado. Esto último es informado al operador mediante un mensaje de respuesta apropiado.

La configuración más simple está constituída por un computador central con capacidad de manejar una o más líneas de comunicación a las cuales estarán conectados uno o más terminales (red). Figura 1.1.1.

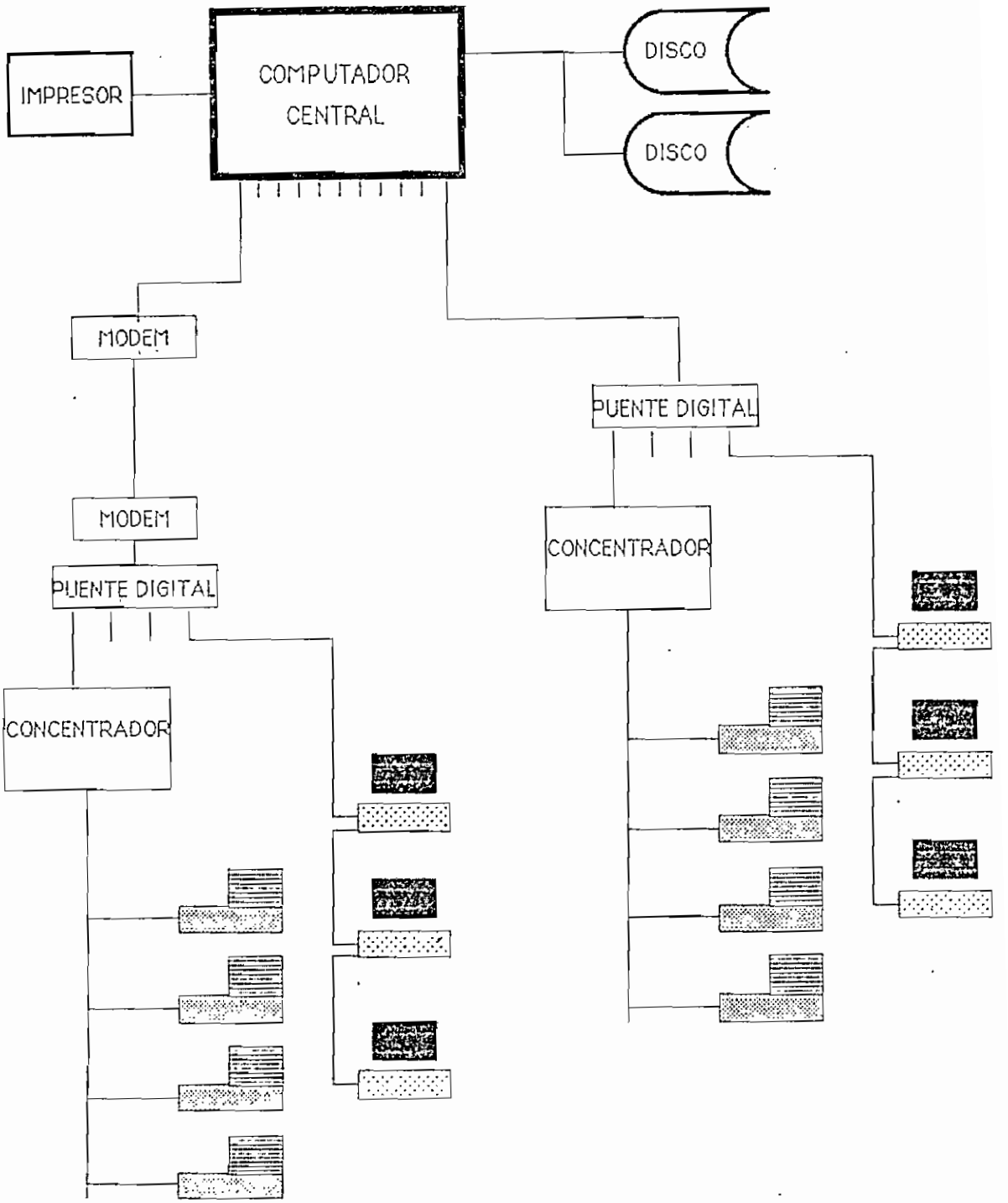

IMPRESOR
COMPUTADOR CENTRAL
DISCO
DISCO
MODEM
MODEM
PUENTE DIGITAL
CONCENTRADOR
PUENTE DIGITAL
CONCENTRADOR

FIGURA 1.1.1.

1.1.2 PROCESADOR CENTRAL.

La función del computador central es principalmente la de permitir el acceso a una base de datos para consulta actualización de información desde cualquiera de los terminales de la red. Para lograrlo , el computador debe disponer de dispositivos comunes de procesamiento de datos tales como discos, impresoras, etc. y de elementos de "hardware" de comunicación necesarios tales como adaptadores de comunicación.

En cuanto al "software" requerido, este consta generalmente de los siguientes módulos:

a) CONTROL DE LA RED

Este componente es responsable por el control de las líneas de comunicación y de los terminales.

Debido a que esta Tesis trata sobre la conexión de un PC a una línea de protocolo NCR ISO ASINCRONICO este trabajo se limitará a describir características relacionadas unicamente con dicho protocolo (ver capitulo 3).

CONTROL DE LAS LINEAS DE COMUNICACION

Una línea de comunicación ISO ASINCRONICA, es una línea multipunto, es decir que a esta se pueden conectar múltiples dispositivos direccionables. Un dispositivo direccionable debe entenderse como un terminal , un concentrador

o controlador de terminales con la capacidad de responder a una secuencia de interrogaciòn y selecciòn.

Una secuencia de interrogaciòn es una secuencia de caracteres que es interpretada por un terminal o dispositivo direccionable como la invitaciòn a enviar un mensaje si este tiene uno listo. La secuencia de selecciòn es tambièn una secuencia de caracteres que es interpretada por el terminal como el permiso que solicita el computador central para enviar un mensaje a dicho terminal.

En una lìnea de estas caracterìsticas, es el computador central quien cumple la funciòn de control de la lìnea; esto es: el computador central es quien invita a los terminales a enviar sus mensajes y en ningùn caso un terminal podrà enviar un mensaje sin haber sido invitado o interrogado por el computador central.

El control de la lìnea entonces radica principalmente en mantenerla activa mediante la emisiòn continua de secuencias de interrogaciòn y el analisis de las respuestas recibidas que pueden ser: mensajes con una transacciòn o, la simple indicaciòn del terminal de no disponer de uno.

Este control puede residir ya sea en el "software" del computador central o en los procesadores de comunicaciones dependiendo del equipo central y de su configuraciòn.

CONTROL DE TERMINALES

El control de los terminales tiene que ver con el manejo de los mensajes enviados por los terminales y de los mensajes enviados por los programas de aplicación hacia los terminales.

Cuando se recibe un mensaje desde un terminal, este mensaje debe ser verificado, identificado y procesado; la verificación incluye el control de paridad de cada uno de los caracteres del mensaje, el control de consistencia del mensaje (BCC Block Check Caracter), y la validez de la dirección del terminal y que forma parte del mensaje. La identificación consiste en analizar el contenido del mensaje para determinar la transacción requerida por el operador del terminal. El proceso está condicionado a los requerimientos del usuario del sistema y normalmente incluye el acceso a la base de datos, cálculos, etc..

b) ACCESO DE ARCHIVOS.

Este componente permite a los programas de aplicación tener acceso a los archivos de datos ya sea para consulta o actualización.

c) RECUPERACION.

Este componente procura garantizar la integridad de archivos en caso de producirse fallas eléctricas, del "hardware", del "software" o de los programas de aplicación.

1.1.3 RED DE TERMINALES.

Por red de terminales debe entenderse no solo a los terminales sino también todos los componentes que hacen posible la conexión de los mismos a los adaptadores de comunicación del computador central, esto incluye concentradores, modems, puentes, líneas telefónicas y cables.

La comunicación entre computador y terminal, se rige a las reglas impuestas por el protocolo utilizado. En este caso el protocolo NCR ISO ASINCRONICO.

El número de líneas de comunicación que salen del procesador central, está generalmente relacionado con el número de terminales, la actividad de los mismos, la extensión promedio de los mensajes, la velocidad de transmisión de la línea y el tiempo de respuesta esperado por el usuario del sistema.

Cada línea de comunicación a su vez permite conectar un número de terminales que está limitado por el ancho de banda de la línea de comunicaciones es decir que se podrá añadir terminales a una línea hasta que el incremento en el tiempo de respuesta producido por esta razón sea aceptable por el usuario. El incremento en dicho tiempo de respuesta depende del número de dispositivos direccionables, del número de terminales, de la actividad de los mismos (transacciones por minuto), de la extensión de los mensajes y del ancho de banda de la línea (velocidad de transmisión).

El tiempo de respuesta a una transacciòn es el sumatorio de lo siguiente:

- tiempo de proceso en el computador central, que incluye acceso a discos, etc.
- tiempo de transmisiòn de los mensajes (el enviado y el recibido), el mismo que depende unicamente de la velocidad de la linea y de la extensiòn de los mismos.
- tiempo de espera del terminal hasta que reciba la secuencia de interrogaciòn y pueda enviar su mensaje.
- tiempo que el computador central debe esperar hasta poder enviar la secuencia de selecciòn, lo cual depende del nùmero de terminales en la linea, de la velocidad de transmisiòn de la misma ,de la actividad de los terminales y de la extensiòn de los mensajes; y
- tiempo que puede existir en reintentos por errores de comunicaciòn.

Una linea de comunicaciones puede atender un grupo de terminales locales, un grupo de terminales remotos o una combinaciòn de terminales locales y remotos.

Se consideran terminales locales aquellos que por la distancia al computador central no requieren de modems para comunicarse.

Existen dos formas en que un terminal puede conectarse a una línea de comunicaciones: directa o indirectamente. La conexión es directa cuando el terminal tiene la capacidad de responder a la secuencia de interrogación emitida por el computador central (dispositivo direccionable). La conexión es indirecta cuando se requiere de un concentrador de terminales o controlador de terminales.

Un concentrador de terminales es un dispositivo que realiza las siguientes funciones:

- de interfase con la línea de comunicaciones: en lo que se refiere a la atención a la secuencia de interrogación, al envío de mensajes de los terminales y a la recepción de mensajes desde el computador central.

- de interfase con los terminales: mediante la utilización de un control basado en prioridades y mensajes.

- de multiplexer al permitir que varios terminales compartan no solo la línea de comunicaciones sino también un mismo código de interrogación.

Un controlador cumple todas las funciones de un concentrador pero adicionalmente permite una mayor capacidad de proceso a nivel de sucursal, sin intervención del computador central.

En cuanto a terminales, se describe brevemente algunos de los màs utilizados en redes transaccionales en el paìs.

NCR-2261: Este es un terminal financiero basado en un microprocesador y que està formado por los siguientes componentes:

- un teclado alfanumèrico de 44 teclas,
- una pantalla de 16 lìneas por 40 columnas,
- una impresora de auditoria de 40 columnas, y con capacidad de certificaciòn de documentos,
- capacidad de comunicaciòn directa o a travès del concentrador NCR-751,
- programaciòn a traves de paràmetros y tablas,
- un "buffer" de comunicaciòn de 256 caracteres y
- utiliza protocolo de comunicaciòn ISO ASINCRONICO con formato de mensajes standard (SMF).

NCR-2251: Este es un terminal financiero basado en un microprocesador y que tiene las siguientes caracteristicas:

- un teclado alfanumèrico de 44 teclas,
- un display nùmerico y luces de indicaciòn (LED'S),
- una impresora de libretas de ahorro o tarjetas de 90 columnas,
- una impresora de auditoria de 40 columnas con capacidad de certificaciòn de documentos,
- capacidad de comunicaciòn directa o a travès de un concentrador NCR-751,
- programaciòn a travès de tablas,
- un buffer de comunicaciòn de 100 caracteres y
- utiliza protocolo ISO-ASINCRONICO y formato de mensaje standard (SMF).

NCR-2262: Este es un terminal financiero basado en un microprocesador que tiene los siguientes componentes y caracteristicas:

- un teclado alfanumèrico programable de 44 teclas,
- una pantalla de 16 lineas por 40 columnas,
- una impresora de auditoria de 40 columnas y con capacidad de certificaciòn de documentos,
- programaciòn en lenguaje BASIC y
- comunicaciòn a travès del "router", que hace las funciones de un concentrador.

NCR-7900-3: Este es un terminal de propòsito general con las siguientes caracteristicas:

- teclados alfanumèrico, numèrico y de funciones,
- pantalla de 24 lìneas de 80 columnas y una lìnea de estado del terminal,
- buffer de comunicaciòn de 2000 caracteres,
- capacidad de conexiòn en cadena (DAYSY CHAIN),
- trabaja en base a formatos de pantalla que deben ser enviados desde el computador central y que define àreas protegidas en la pantalla. El operador puede ingresar informaciòn ùnicamente en las areas desprotegidas. Mediante la tecla de transmisiòn, el terminal envìa el contenido de todos los campos desprotegidos al computador central,
- no es un terminal programable,
- utiliza el protocolo ISO ASINCRONICO con un formato de mensaje propio y
- no ejerce ninguna validaciòn sobre el tipos de datos ingresados. Ejem. numèricos, fechas, etc.

NCR-6440: Esta es una impresora de propòsito general con las siguientes caracteristicas:

- 132 columnas de impresiòn,
- buffer de comunicaciòn de 2048 caracteres e
- interfase de comunicaciones ISO ASINCRONICO.

1.2.1 INTRODUCCION

En este capitulo se describen de manera general las ideas utilizadas en el diseño e implementaciòn de este producto, asì como tambièn sus dos ambientes de trabajo, esto es el de definiciòn de los paramètros que identifican una transacciòn y el de utilizaciòn del PC como terminal.

La idea principal de este producto es la de utilizar los recursos del PC estos son: el procesador, la pantalla, el teclado, la impresora, los medios de almacenamiento y el sistema operativo, para optimizar los recursos de una red transaccional NCR, esto es, el computador central y la utilizaciòn del ancho de banda de las lìneas de comunicaciòn y adicionalmente proveer de un terminal de caracteristicas muy superiores a productos existentes.

Las caracteristicas de este nuevo terminal pueden ser resumidas en las siguientes:

- CONTROL DE INGRESO DE INFORMACION: El PC debe tener la capacidad de controlar el ingreso de datos que constituyen una transacción. Este ingreso debe realizarse en los siguientes pasos:
  - desplegar un formato de pantalla con instrucciones al operador.
  - realizar un proceso de validación de cada uno de los campos ingresados respecto a parámetros especificados,
  - añadir caracteres de control a los campos ingresados para ensamblar un mensaje que debe enviarse al computador transaccional, y
  - procesar el mensaje de respuesta enviado por el computador transaccional.

- CONTROL DE LA LINEA Y DEL PROTOCOLO: El PC debe atender en tiempo real la línea de comunicaciones y el protocolo NCR-ISO ASINCRONICO. Para lograr esto se debe incluir un "DRIVER" de comunicaciones que trabaje utilizando interrupciones. Algunos parámetros del "DRIVER" deben poder ser modificados por el operador: velocidad, código de interrogación.

- HERRAMIENTAS DE MANTENIMIENTO DE PARAMETROS: El usuario de este "nuevo" terminal debe poder definir sus propios formatos de pantalla, las reglas de validaciòn de los campos de entrada, y los caracteres de control que deben añadirse a los datos ingresados para permitir la identificaciòn del mensaje enviado por parte del computador central.

- SEGURIDAD DE ACCESO: El PC debe adicionar caracterìsticas de seguridad de acceso a las provistas por el computador central limitando el acceso unicamente a personal autorizado.

- CONTROL REMOTO: Se debe proveer al computador central de la capacidad de controlar los recursos de PC mediante còdigos de control.

1.2.2 FUNCIONAMIENTO COMO TERMINAL

Durante la utilización del PC como terminal, este se conecta a la línea de comunicaciones mediante el adaptador serial RS-232 y gracias al "driver" de comunicaciones,. se convierte en un dispositivo que responde a una secuencia de interrogacion y una secuencia de selección.

Independientemente del proceso de atención de la linea y protocolo de comunicaciones, y concurrentemente con dicho proceso, el PC debe atender los requerimientos del usuario realizados a través del teclado.

El usuario de este "terminal" puede realizar las siguientes funciones:

- FUNCIONES DE SUPERVISOR:
  - procesos de inicio de día: ingreso de fecha y hora
  - reconfiguración del terminal: cambio del código de interrogación y la velocidad
  - iniciación del "driver" de comunicaciones

- SOLICITUD DE AYUDA:

  Que consiste en consultar las transacciones disponibles por el operador.

- SOLICITUD DE ACCESO A LAS TRANSACCIONES:

  Proceso que permite verificar la clave de acceso de un operador y además permite al operador selecionar un grupo

- SOLICITUD DE UN FORMATO ESPECIFICO:

  Proceso mediante el cual el operador solicita la presentaciòn de un formato de pantalla. Luego de que el PC despliega dicho formato, el operador puede ingresar los datos que cosntituyen la transacciòn.

- FIN DEL PROCESO:

  Mediante esta funciòn, el operador regresa a las funciones normales del PC.

1.2.3 AMBIENTE DE DEFINICION DE TRANSACCIONES

En esta modalidad de trabajo, el operador está en capacidad de definir o eliminar información referente a lo siguiente:

- transacciones disponibles
- formatos de pantalla

Los parámetros que identifican una transacción, son los siguientes:

- número de la transacción
- nombre de la transacción
- grupo al que pertenece
- nivel de seguridad
- número de formato de pantalla asociado.

Los parámetros que identifican un formato de pantalla son los siguientes:

- número del formato
- datos constantes del formato (títulos)
- campos de entrada (línea, posición, longitud, tipo, código de control)
- campos de salida (línea, posición, longitud)

El programa que controla la definiciòn de estos parámetros permite además la impresiòn de los mismos.

Los parámetros ingresados son almacenados en archivos de organizaciòn relativa y de acceso directo lo que permite operar eficientemente en la modalidad de terminal.

2.1 INTRODUCCION

En este capítulo se describen las características principales del computador personal NCR PC-4i y algunos de los componentes principales del mismo.

El computador personal NCR PC-4i ha sido diseñado para proveer compatibilidad con el computador personal IBM-XT. Gracias a ello, un porcentaje muy alto de productos de "hardware" y "software" disponibles en el mercado pueden ser utilizados.

El computador personal NCR PC-4i consiste de una unidad de pantalla y procesador, y de un teclado. Existen varios modelos y cada uno de ellos tiene una serie de opciones que pueden añadirse.

La unidad de pantalla standard consta de los siguientes elementos:

- gabinete,
- fuente de poder,
- una o dos unidades de disco flexible de 5 1/4 pulgadas,
- pantalla monocromática o de color de 12 pulgadas,
- tarjeta de control con los siguientes componentes:
  - microprocesador INTEL 8088,
  - 256 kb de memoria RAM,
  - 16 kb de memoria ROM,
  - interfase serial RS-232-C,
  - interfase paralelo CENTRONICS,

2.2 PROCESADOR

La figura 2.1 ilustra los componentes principales del procesador del computador personal NCR PC-4i. Dichos componentes se describen a continuación:

CLOCK: Toda la coordinación entre los componentes del sistema está dada por señales de reloj. Para ello se utiliza un cristal de 14.31 Mhz como entrada al controlador de reloj 8284A el cual divide esta frecuencia para obtener una señal de 4.77 Mhz para el microprocesador y otras señales para otros componentes.

TIMER: Es un circuito integrado que se puede programar para generar diferentes señales a varios intervalos. El "timer" utiliza el reloj, y mediante divisiones de frecuencia apropiadas obtiene un exactitud aceptable. El "timer" es utilizado para refrescar la memoria RAM y puede ser utilizado por el usuario para diferentes aplicaciones, por ejemplo, un reloj en la pantalla que funcione independientemente del programa de aplicación, o en aplicaciones de tiempo real como el control de una línea de comunicaciones.

CPU: Es un microprocesador INTEL 8088. Es quien ejecuta las instrucciones dadas en un programa, y es quien transfiere dichas instrucciones a otras unidades o componentes del sistema. Opcionalmente, se puede utilizar el co-procesador

INTEL 8087, quien a su vez tiene la capacidad de realizar operaciones de punto flotante como respuesta a una simple instrucciòn.

MEMORIA: El computador personal hace uso de dos tipos de memoria:

-ROM:(read only memory) que contiene instrucciones que permiten la preparaciòn inicial y carga del sistema operativo al momento de encendido. Contiene ademàs una serie de programas conocidos como BIOS (basic input/output system).

-RAM (random access memory): que puede contener ya sea instrucciones o datos que estàn siendo procesados por un programa. Esta memoria es dinàmica por lo que es necesario refrescar el contenido de la memoria a intervalos fijos de tiempo. Este proceso es manejado por el "timer".

DMA (direct memory access): Este controlador es utilizado para transferir bloques de datos de y hacia la memoria RAM sin intervenciòn del CPU. Gracias al DMA, el CPU requiere unicamente especificar desde o hacia donde los datos deben trasladarse, la direcciòn del primer caràcter de datos y el nùmero de caracteres. La transferencia tiene luego lugar sin intervenciòn del CPU. DMA es utilizado para el acceso a perifèricos tales como discos.

2.3 PERIFERICOS

El computador personal NCR PC4i puede ser configurado con los siguientes perifericos:

UNIDADES DE ACCESO DIRECTO:

- Una o dos unidades de diskettte integradas de 5 1/4 pulgadas.
- Un disco duro dentro del gabinete de 10 MB.
- Discos duros exteriores de 11 MB, 22 MB, 32 MB o 64 MB.
- Discos flexibles de 8 pulgadas.

IMPRESORAS:

Existe una gran variedad de impresoras que pueden conectarse. El computador personal dispone de un puerto serial y un puerto paralelo que pueden utilizarse para el manejo de la impresora.

DISPOSITIVOS DE COMUNICACIONES:

Los siguientes adaptadores se encuentran disponibles:

- RS-232-C ASINCRONICO: que permite conectar dispositivos tales como: impresoras, "ploters", modems, terminales, computadores, etc.

- IN-HOUSE DLC: Este adaptador permite la conexiòn del computador personal a una lìnea DLC (data link control) ya sea como estaciòn primaria o secundaria.

- BSC (Binary synchronous communications): que permite la conexiòn del computador personal a una lìnea BSC. Esta lìnea podrìa utilizar protocolos punto a punto tales como BSC 2780 o BSC 3780 o multipunto como el BSC 3270.

2.4 INTERFASE ASINCRONICO RS-232-C.

El interfase asincrònico RS-232-C puede ser utilizado para una gran variedad de conexiones y aplicaciones. Mediante esta interfase el computador personal NCR PC4i puede conectarse a impresoras seriales, lìneas de comunicaciòn asincrònica punto a punto o multipunto, dispositivos tales como: balanzas, alarmas, relojes, etc..

Un màximo de dos interfases pueden trabajar simultàneamente en este computador.

El primer interfase esta incluido dentro de la tarjeta del procesador central, mientras que el segundo es una tarjeta independiente que puede conectarse en uno de los conectores de la tarjeta del procesador.

La conexiòn exterior se realiza mediante un conector EIA de 25 contactos, de los cuales tienen interès para comunicaciones del tipo ISO ASINCRONICO los siguientes:

CONECTOR		DESCRIPCION
1	-	FG (FRAME GROUND)
2	>	TD (TRANSMIT DATA)
3	<	RD (RECEIVE DATA)
4	>	RTS (REQUEST TO SEND)
5	<	CTS (CLEAR TO SEND)
6	<	DTS (DATA SET READY)
7	-	SG (SIGNAL GROUND)
8	<	DCD (DATA CARRIER DETECT)
20	>	DTR (DATA TERMINAL READY)

El adaptador RS-232-C contiene el transmisor / receptor WESTERN DIGITAL 8250 lo cual le convierte en un adaptador programable. La figura F 2.4.1 ilustra la estructura lògica de este circuito integrado.

Las señales de control, direcciòn y data del 8250 se describen a continuaciòn:

- terminales 1-8 (D0-D7): Bus de datos utilizados para transferir informaciòn de datos y control en forma paralela desde y hacia el CPU.

- terminal 9 (RCLK): señal de reloj de 16 veces la velocidad del receptor. Esta señal està dada por el terminal 15 (BAUDOUT).
- terminal 10 (SIN): recibe datos seriales desde RD (receive data).
- terminal 11 (SOUT): envia datos seriales hacia TD (transmit data).
- terminales 12-14 (CS0,CS1,CS2): señales que permiten seleccionar al circuito 8250 (chip select). De estas terminales, solamente es utilizado el 14. Los terminales 12 y 13 esta conectados a 5 Voltios.
- terminal 15 (BAUDOUT): señal de reloj de 16 veces la velocidad del transmisor. Esta es una señal de salida que se conecta al terminal 9. La frecuencia de esta señal està dada por la divisiòn de la frecuencia de reloj ingresada en el terminal 16 (XTAL1) por el contenido de los registros divisores (divisor latches).

- terminales 16,17 (XTAL1,XTAL2) : señal de reloj que ingresa a un circuito divisor dentro de 8250. La señal de reloj conectada a XTAL1 es de 1.8 Mhz. XTAL2 no está conectado.
- terminales 18,19 (DOSTR): señales que permiten cargar datos en los registros del 8250. Solamente el terminal 18 es utilizado, el 19 esta conectado a tierra.
- terminal 20 (Vss): señal de tierra.
- terminal 21,22 (DISTR): señales que permiten leer el contenido de los registros del 8250. Solamente el terminal 21 está conectado, el 22 está a tierra.
- terminal 23 (DDIS) (driver disable): señal que baja durante la lectura de datos desde el 8250. Este terminal no está conectado.
- terminal 24 (CSOUT): esta señal se levanta para confirmar la selección del 8250. Esta señal no está conectada.
- terminal 25 (ADS) (address strobe) : señal requerida en caso de que las lineas de dirección de registros internos del 8250 no son estables. Este terminal está conectado a tierra.
- terminales 26,27,28 (A2,A1,A0): bus de selección de un registro interno del 8250.
- terminal 29: no utilizado.
- terminal 30 (INTRPT): señal que es activada cuando una interrupción al procesador es requerida. Esta señal es utilizada junto a la terminal 31 (OUT2).

- terminal 31 (OUT2): señal que es activa si el tercer bit del registro de control del modem es 1. Esta señal es utilizada junto al terminal 30 (INTRPT).
- terminal 32 (RTS) (request to send): señal que indica a dispositivo exterior que el 8250 esta listo a transmitir.
- terminal 33 (DSR) (data set ready): señal que indica al dispositivo exterior que el 8250 está listo para comunicarse.
- terminal 34 (OUT1): señal que es activa si el segundo bit del registro de control de modem es 1.
- terminal 35 (MR) (master reset): señal que permite borrar los registros internos del 8250.
- terminal 36 (CTS) (clear to send): señal que envia el dispositivo exterior indicando al 8250 que dicho dispositivo está listo para transmitir.
- terminal 37 (DSR) (data set ready): señal desde el dispositivo exterior que indica que está listo para comunicarse.
- terminal 38 (RSLD) (carrier detect): señal enviada por el dispositivo exterior indicando la presencia de señales adecuadas.
- terminal 39 (RI) (ring indicator): señal que indica que el dispositivo exterior está recibiendo la señal de llamada.
- terminal 40 (Vcc): +5 Vdc

PROGRAMACION DEL 8250

El computador personal NCR PC4i puede tener hasta dos interfases seriales asincrònicos. Para ello, estàn reservados las direcciones 3F8-3FF para el primer adaptador y las direcciones 2F8-2FF para el segundo.

A travès de la direcciòn 3F8, se envia o recibe datos al o del registro de dato del 8250 (receiver buffer/ transmit holding register). Esta direciòn es utilizada tambièn para programar la velocidad de transmisiòn cuando el sèptimo bit del registro de control de la lìnea es 1.

La direcciòn 3F9 permite habilitar las interrupciones. El 8250 mantiene un registro interno para conocer los tipos de interrupciones habilitadas en un momento dado. Esta direcciòn, al igual que la anterior es utilizado tambièn para programar la velocidad de transmisiòn de la linea mientras el sèptimo bit del registro de control de la lìnea es 1.

La direcciòn 3FA permite leer la identificaciòn de la ùltima interrupciòn.

La direcciòn 3FB permite escribir en el registro de control de la lìnea.

La direcciòn 3FC permite escribir en el registro de control de modem.

La direcciòn 3FD permite leer el registro de estado de la lìnea.

La direcciòn 3FE permite leer el registro de estado del modem.

La direcciòn 3FF no es utilizada.

Las direcciones 2F8 a 2FF, son las equivalentes para el segùndo interfase asincrònico.

REGISTRO DE INTERRUPCIONES HABILITADAS

Es un registro de 8 bits, de los cuales solamente los 4 menos significativos son utilizados y tienen el siguiente significado:

- bit 0 : si este bit es 1, el 8250 generarà una interrupciòn cuando ha recibido un caràcter de datos.
- bit 1 : si este bit es 1, el 8250 generarà una interrupciòn cuando ha terminado de transmitir un caràcter.
- bit 2 : si este bit es 1, el 8250 generarà una interrupciòn cuando ha detectado un error.
- bit 3 : si este bit es 1, el 8250 generarà una interrupciòn cuando detecte un cambio de estado en las siguientes señales: CTS, DSR, RI, RLSD

REGISTRO DE IDENTIFICACION DE LA INTERRUPCION

Es un registro de 8 bits, de los cuales son utilizados unicamente los 3 menos significativos y que tienen el siguiente significado.

- bit 0 : tiene un valor de 0 mientras existe una interrupción que debe ser atendida.
- bit 1 y 2 : el valor binario de estos 2 bits representa la identificación de la interrupción mediante la siguiente codificación:

  b2=1, b1=1 indica que la interrupción se debe a la detección de un error en recepción. Este error puede ser por ejemplo de paridad.

  b2=1, b1=0 indica que se ha recibido un carácter de datos. Se debe leer el registro de datos para obtener el carácter.

  b2=0, b1=1 indica que el último caracter escrito en el registro de datos ha sido serializado y transmitido.

  b2=0, b1=0 indica que se ha detectado un cambio en alguna de las siguientes señales del modem : CTS, DSR, RI, RLSD.

REGISTRO DE CONTROL DE LA LINEA.

Este es un registro de ocho bits que tienen el siguiente significado:

- bits 1 y 0: identifican la longitud del carácter de acuerdo al valor binario especificado:

  00 = 5 bits,

  01 = 6 bits,

10 = 7 bits, y

11 = 8 bits.

- bit 2: especifica el nùmero de stop bits de acuerdo a la siguiente codificaciòn:

  0 = 1 stop bit,

  1 = 2 stop bits. (1.5 para el caso de caràcteres de 5 bits)

- bit 3: indica si el control de paridad està habilitado o no.
- bits 4 y 5: especifican el tipo de paridad: par o impar.
- bit 6 genera una señal baja en el transmisor (break). Esta señal se mantiene baja hasta que este bit cambie.
- bit 7 : este bit permite que las direcciones 3F8 y 3F9 (2F8 y 2F9) puedan utilizarse para programar la velocidad de transmisiòn. Si este bit es 0, los puertos especificados tienen sus funciones normales.

REGISTRO DE CONTROL DE MODEM

De los ocho bits de este registro, unicamente los cinco menos significativos son utilizados y tienen el siguiente significado:

- bits 0,1,2,3: estos bits permiten controlar las siguientes lineas del 8250: DTR,RTS,OUT1,OUT2.
- bit 4: activa las funciones de diagnòstico interno del 8250.

REGISTRO DE ESTADO DE LA LINEA

De este registro se utilizan los 7 bits menos significativos, y tinen el siguiente significado:

- bit 0: indica que un carácter ha sido recibido, se ha convertido en paralelo, y que esta listo para ser leído.
- bit 1: indica que se ha producido una condición de "overrun", lo cual quiere decir que ha llegado un nuevo carácter antes de que el microprocesador haya leído el anterior.
- bit 2: indica un error de paridad en recepción.
- bit 3: indica un error en la detección de bits de parada (stop bit).
- bit 4: indica la detección de una señal de "break".
- bit 5: indica que el último carácter ya ha sido enviado, y que un nuevo carácter puede ser escrito en el registro de datos del 8250.
- bit 6: indica el estado del registro de datos (transmit shift register).

REGISTRO DE ESTADO DEL MODEM

Los ocho bits de este registro tienen el siguiente significado:

- bit 0: indica que se ha detectado un cambio en la línea CTS desde la última lectura a este registro.
- bit 1: indica que se ha detectado un cambio en la línea DSR desde la última lectura a este registro.

- bit 2: indica que se ha detectado una señal de llamada (ring indicator).
- bit 3: indica que se ha detectado un cambio en la portadora (RLSD) desde la última lectura de este registro.
- bits 4-7: representan los complementos de las señales CTS, DSR, RI, RLSD respectivamente. En modo de diagnostico, estos bits representan las señales RTS, DTR, OUT1, OUT2.

2.5 SOFTWARE.

El computador personal NCR PC4i, gracias a su caracteristica de compatibilidad, tiene la capacidad de ejecutar gran cantidad de programas (software) disponibles en el mercado. De ellos, esta Tesis describira unicamente los utilizados en el desarrollo del producto.

NCR-DOS

Es el sistema operativo utilizado y es operacionalmente compatible con MS-DOS y PC-DOS. El sistema operativo es un interfase entre el usuario y el computador. Es un grupo de programas que controlan la ejecución de programas en el computador, controla también los recursos del computador, lo que incluye la memoria, el espacio en disco, la impresora, etc..

NCR-DOS provee una serie de comandos que permiten al usuario ordenar la ejecución de tareas tales como la ejecución de un programa, listar un directorio, copiar un archivo.

El sistema operativo es también una ayuda muy importante en el manejo de información, por que es el quien controla la forma de almacenamiento de la información en las unidades de disco. Todos los programas, textos, y datos residen en discos en forma de archivos.

Un archivo es un conjunto de información relacionada, por ejemplo toda la información de las cuentas corrientes de un Banco puede estar almacenada en un archivo.

Un archivo esta identificado por un nombre, y existe una estructura de directorios que continene información de dichos nombres, del tipo de archivo y del la ubicación de los datos dentro del disco.

Adicionalmente a los comandos del sistema operativo y a las rutinas de control de periféricos tales como discos, impresoras, teclado, pantalla, parlante, etc., NCR-DOS provee una serie de programas o rutinas que se denominan comandos externos, y que son de utilidad para realizar funciones de mantenimiento, desarrollo, prueba, de sistemas, entre ellas cabe destacar las siguientes:

- FORMAT: que permite formatear un disco flexible o rigido para que pueda ser reconocido por NCR-DOS y asi pueda ser utilizado para almacenar información.

- SORT: que permite clasificar archivos.

- DISKCOPY: que permite copiar información de un disco a otro.

- CHKDSK: que permite detectar y en ocaciones corregir errores o inconsistencias en un disco.

- EDLIN: que es un editor de linea que permite crear, modificar y presentar en pantalla archivos de texto o programas.

- LINK: que permite convertir uno o mas modulos objeto en un modulo ejecutable.

GW-BASIC

Este es un producto de MICROSOFT que permite crear, probar y ejecutar programas escritos en este lenguaje. La caracteristica fundamental de este producto es la de permitir accesar con facilidad los diferentes dispositivos del computador personal, esto incluye a los discos, impresoras, comunicaciones, gràficos, y mùsica.

GW-BASIC incluye un editor de pàgina que es de gran utilidad al momento de desarrollo de programas o aplicaciones.

Gracias a extenso nùmero de instrucciones disponibles, este lenguaje ha sido seleccionado para el desarrollo de los programas principales de esta Tesis.

Los programas que forman parte de esta Tesis, hacen uso extensivo de instrucciones de acceso directo a memoria, de funciones de acceso a impresoras , pantalla y discos, de instrucciones de manejo de listas de caracteres (strings), de funciones binarias, etc..

Debido a que la velocidad de ejecución de programas interpretados es significativamente mayor que programas compilados, el compilador QUICK-BASIC de MICROSOFT ha sido seleccionado para generar los programas objeto finales.

QUICK BASIC

Este producto presenta gran compatibilidad con programas escritos para GW-BASIC pero permiten mejorar los tiempos de ejecución significativamente.

MICROSOFT MACRO ASSEMBLER

Este producto permite crear programas que pueden ejecutarse en computadores que tengan uno de los siguientes microprocesadores: 8086,8088,80186,80286 y también para computadores que tengan los siguientes coprocesadores: 8087, 80287.

Las rutinas de atención de interrupciones han sido escritas en asembler y ensambladas gracias a este producto.

3.1 INTRODUCCION

En este capítulo se describen: el interfase a la línea de comunicaciones y las características de comunicación que incluyen: el código de caracteres utilizados, los caracteres de control, el modo de transmisión, los controles de paridad , la detección y corrección de errores. También, se detalla el formato de los mensajes utilizados en las redes transaccionales de NCR.

INTERFASE A LA LINEA

El computador personal NCR-PC4i (en adelante llamado "terminal") puede conectarse a una línea de comunicaciones de la red ya sea directamente o a través de modems; en los dos casos, la conexión se realiza mediante el interfase RS-232-C del terminal.

CODIGO DE CARACTERES

El terminal utiliza el código ASCII (American Standard Code for Information Interchange) para la transmisión y recepción. El código ASCII, consiste de 128 caracteres codificados que incluyen los caracteres alfabéticos, los numéricos, caracteres especiales y códigos de control. De los ocho bits que forman un carácter, los siete bits menos significativos representan el código del carácter y el bit más significativo es el bit de paridad utilizado durante la transmisión (ver FIGURA F.3.1.1).

Right Hex / Left Hex		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL 0	SOH 1	STX 2	ETX 3	EOT 4	ENQ 5	ACK 6	BEL 7	BS 8	HT 9	LF 10	VT 11	FF 12	CR 13	SO 14	SI 15
0001	1	DLE 16	DC1 17	DC2 18	DC3 19	DC4 20	NAK 21	SYN 22	ETB 23	CAN 24	EM 25	SUB 26	ESC 27	FS 28	GS 29	RS 30	US 31
0010	2	Ø 32	! 33	" 34	# 35	$ 36	% 37	& 38	' 39	(40	) 41	* 42	+ 43	, 44	— 45	. 46	/ 47
0011	3	0 48	1 49	2 50	3 51	4 52	5 53	6 54	7 55	8 56	9 57	: 58	; 59	< 60	= 61	> 62	? 63
0100	4	@ 64	A 65	B 66	C 67	D 68	E 69	F 70	G 71	H 72	I 73	J 74	K 75	L 76	M 77	N 78	O 79
0101	5	P 80	Q 81	R 82	S 83	T 84	U 85	V 86	W 87	X 88	Y 89	Z 90	[91	\ 92	] 93	^ 94	← 95
0110	6	` 96	a 97	b 98	c 99	d 100	e 101	f 102	g 103	h 104	i 105	j 106	k 107	l 108	m 109	n 110	o 111
0111	7	p 112	q 113	r 114	s 115	t 116	u 117	v 118	w 119	x 120	y 121	z 122	{ 123	¦ 124	} 125	~ 126	DEL 127
1000	8	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
1001	9	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
1010	A	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
1011	B	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
1100	C	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
1101	D	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
1110	E	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
1111	F	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

FIGURA F.3.1.1

CARACTERES DE CONTROL

Un caràcter de control es aquel que es utilizado para crear un protocolo, es decir es un caràcter que debe ser interpretado tanto por el terminal como por el computador central como un comando, una respuesta a un comando, el inicio de datos, el fin de datos, etc.. Los siguientes son los utilizados por el terminal producto de esta tesis:

- STX (start of text) 02H : Este carácter es utilizado como indicador de que los siguientes caracteres son de datos.

- ETX (end of text) 03H: Este carácter indica que el carácter anterior era el último del mensaje, y .que el siguiente es el carácter de consistencia del mensaje (BCC block check character).

- NAK (negative acknowledgement) 15H: Este carácter es transmitido por quien recibe un mensaje como respuesta a quien envía, en el caso de haber detectado errores de transmisión tales como: error de paridad, error de BCC.

- ACK (positive acknowledgement) 06H: Este carácter es una respuesta de aceptación de un mensaje que es transmitido por el que recibe dicho mensaje.

- EOT (end of transmission) 04H: Este carácter es utilizado para iniciar una secuencia de interrogación o de selección, y también es utilizado como respuesta a un ACK.

- ENQ (enquiry) 05H: Este carácter es utilizado para terminar una secuencia de interrogación o de selección y también como una solicitud de respuesta en caso de que una estación no haya recibido una notificación positiva o negativa a un mensaje.

- US (unit separator) 1FH: Este carácter, es utilizado

dentro del mensaje (entre STX y ETX) para separar diferentes campos de informaciòn.

MODO DE TRANSMISION

El terminal transmite y recibe datos asincrònicamente. Cada caràcter esta formado por siete bits del còdigo, un bit de paridad, un bit de inicio (0) y uno de fin (1). Los diez bits son transmitidos serialmente bit por bit, y el tiempo entre la transmisiòn de dos caracteres puede ser variable.

El orden de transmisiòn de los bits es el siguiente: bit de inicio, bit menos significativo del còdigo del caràcter seguido por los seis siguientes bits, bit de paridad y bit de fin ver FIGURA F.3.1.2.

BF	P	7	6	5	4	3	2	1	BI

BF= BIT DE FIN
P= BIT DE PARIDAD
BI= BIT DE INICIO

FIGURA F.3.1.2

CONTROL DE PARIDAD DE UN CARACTER

La generaciòn del bit de paridad, y el control del mismo se realizan de forma tal que el nùmero de bits con valor de 1 de los ocho bits que conforman el caràcter (incluyendo el de paridad) sea un numero par.

DETECCION DE ERRORES

Los siguientes procedimientos son utilizados para la detecciòn de errores:

- PARIDAD: que consiste en la verificaciòn de cada uno de los caracteres recibidos. Un caràcter errado causa que todo el mensaje sea errònso.

- BCC (block check caracter): Todo mensaje de datos incluye un caràcter adicional que es transmitido a continuaciòn del ETX y que es generado por la estaciòn transmisora y verificado por la estaciòn receptora. Tanto la estaciòn transmisora como la receptora utilizan, lògicamente, la misma forma de càlculo y que consiste en llevar un sumatorio de los bits de todos los caracteres del mensaje (se excluye STX y BCC). Este sumatorio se realiza sin "carrier" y entre los bits que ocupan una misma posiciòn es decir que el bit nùmero dos del BCC es la suma de todos los bits nùmero dos de todos los caracteres del mensaje a partir de STX e incluyendo a ETX. Lo mismo es aplicable para los siete bits

que conforman el BCC, el cual tambièn tiene un bit de paridad.

- FORMATO DEL MENSAJE

Que consiste en verificar cierta estructura del mensaje. En el caso de nuestro terminal, esta estructura corresponde a formato standard de mensaje (SMF) descrita màs adelante.

CORRECCION DE ERRORES

La correcciòn de errores consiste en la retransmisiòn de los mensajes cuando se ha detectado un error de transmisiòn.

En el caso de envio de un mensaje desde el terminal, este espera durante un determinado tiempo un respuesta positiva o negativa; se considera un error si se recibe una notificaciòn negativa o nada. En el caso de un NAK, el terminal retransmite el mensaje, en el caso de no tener una respuesta, el terminal transmite un ENQ.

En el caso de recepciòn de un mensaje en el terminal, este verifica el mensaje de acuerdo a los procedimientos descritos anteriormente, y responde positiva o negativamente al computador central. Es responsabilidad del computador central realizar las funciones de retransmisiòn.

FORMATO DE MENSAJE STANDARD

Este es uno de los formatos de mensaje màs comunmente utilizados actualmente en las redes NCR.

El formato de mensaje se refiere a la estructura que tienen los mensajes de datos, es decir de aquellos mensajes que se inician con STX y terminan con ETX.

- MENSAJES ENVIADOS DESDE EL TERMINAL

Los mensajes enviados desde el terminal tienen la siguiente estructura:

- STX : caràcter de inicio del mensaje.
- VLI (indicador de la longitud del mensaje): son dos caracteres que tienen un valor constante hexadecimal de 7E7E.
- RLTC (relationship code): es un caràcter que contiene el valor hexadècimal 24. Este caràcter no tiene mayor importancia.

- FRMT (format code): es un caràcter que contiene el valor hexadècimal 31. Este caràcter tampoco tiene mayor importancia.
- TA1: que representa el primer caràcter de identificaciòn del terminal.
- TA2: que es el segundo caràcter de identificaciòn del terminal.

- MMC: (message code characters): son dos caracteres conocidos como MMC1 y MMC2 que definen ciertas caracteristicas del mensaje mediante la configuración de sus bits. La siguiente es la descripción del significado de cada uno de los bits de estos caracteres.

  - MMC1 BIT 8: es el bit de paridad de este carácter.
  - MMC1 BIT 7: siempre es 1.
  - MMC1 BIT 6: es 1 si el terminal esta trabajando en modo de reingreso automático de transacciones desde medio magnético.
  - MMC1 BIT 5: siempre 0
  - MMC1 BIT 4: es 1 si el terminal esta trabajando en modo de reingreso manual de transacciones.
  - MMC1 BIT 3: es 1 si el terminal esta siendo operado en modo de diagnóstico.
  - MMC1 BIT 2: es 1 si este mensaje es un mensaje de anulación.
  - MMC1 BIT 1: es 1 si este es el último segmento del mensaje.
  - MMC2 BIT 8: es el bit de paridad de este carácter.
  - MMC2 BIT 7: siempre es 1.
  - MMC2 BIT 6: no usado , es 0.
  - MMC2 BIT 5: no usado , es 0.
  - MMC2 BIT 4: no usado , es 0.
  - MMC2 BIT 3: indica cuando es 1 que este es un mensaje de autorización (override).
  - MMC2 BIT 2: no usado , es 0.
  - MMC2 BIT 1: no usado , es 0.

- TRS#: son tres caracteres que contienen el nùmero de transacciòn del terminal.
- TRM#: es un caràcter que contiene un nùmero consecutivo de transmisiòn.
- FC: es còdigo de funciòn que puede tener los siguientes valores:
  - 71 que indica que el operador nùmero 1 esta operando el terminal.
  - 72 que indica que el operador nùmero 2 esta operando el terminal.
  - 73 que indica que el supervisor esta operando el terminal.
- ID: son 4 caracteres que contienen el nùmero pùblico del operador que esta realizando esta transacciòn.
- DATOS: es el area del mensaje que contiene los datos ingresados por el operador. cada campo de informaciòn es transmitido junto a un còdigo de funciòn, y es separado del siguiente campo con un US (unit separator).

-ETX : caràcter de fin del mensaje.

El siguiente es un ejemplo de un mensaje enviado desde el terminal:

STX_H7E_H7E_H24_H31_TA1_TA2_H41_H30_"001"_"1"_H71_"0005"..
.. _US_H31_"123456"_US_H32_"50000"_US_H33_"321554"_ETX_BCC

- MENSAJES ENVIADOS AL TERMINAL

El formato de los mensajes enviados por el computador central se rige al siguiente formato:

- STX: caràcter de inicio del mensaje.
- VLI: (variable length indicator) son dos caracteres que contienen el valor haxadecimal 7E7E.
- RLTC: es un caràcter que contiene el valor hexadècimal de 30.
- MDC (message definition character): es un caràcter que contiene informaciòn de ciertas caracterìsticas del mensaje. Cada uno de los bits de esta caràcter tienen el siguiente significado:
  - BIT 8: paridad del caràcter.
  - BIT 7: siempre 1.
  - BIT 6,5,4 : siempre 0.
  - BIT 3: tiene un valor de 1, si este es un mensaje de diagnòstico.
  - BIT 2: tiene un valor de 1 si este es un mensaje no solicitado.
  - BIT 1: tiene un valor de 1 si este es el ùltimo segmento del mensaje.
- DATOS: es el area del mensaje en que estàn los datos enviados por el computador central. El mensaje puede contener màs de un dato, por lo que cada uno de ellos debe ir asociado a un còdigo de funciòn y debe estar separado con un US (unit separator). El còdigo de funciòn informa al terminal que proceso debe realizarse con el dato asociado.
- ETX : caràcter de fin del mensaje.

3.2 STANDARDS PARA COMUNICACION DE DATOS

Los siguientes standards para comunicaciones de datos estàn relacionados con el protocolo de comunicaciòn NCR ISO ASINCRONICO. Una copia de todos los standards referidos se encuentra en el apendice "A".

ISO 646: 7-bit coded character set for information processing interchange.

Este standard contiene 128 caracteres codificados que incluyen caracteres de control y caracteres gràficos tales como letras, nùmeros y simbolos.

ISO 1155: Information processing - Use of longitudinal parity to detect errors in information messages.

Este standard define las reglas para generar y verificar el BCC (block check character).

ISO 1177: Information processing - Character structure for start/stop and synchronous transmission.

Este standard define la estructura de un caràcter durante la transmisiòn lo cual incluye los siguientes bits: de inicio, de fin , de paridad y 7 del còdigo del caràcter.

ISO 2110: Data communications - 25-pin DTE/DCE interface connector and pin assgnments.

Este standard define las caracterìsticas mecànicas del conector y la asignaciòn de señales a cada uno de los 25 terminales.

ISO 1745: Information processing - Basic mode control procedures for data communication systems.

Este standard define una serie de conceptos relacionados con transmisiòn de datos utilizando còdigos de siete bits, define caracteres de control, el formato bàsico de mensajes tanto de supervisiòn como de informaciòn, define tambièn conceptos de "polling" y "selecting".

ISO 2628: Basic mode control procedures- Complements.

Este standard define procedimientos de recuperaciòn de errores, la terminaciòn (abort) de la transmisiòn de mensajes, y selecciòn mùltiple de terminales.

EIA STANDARD RS-232-C

Este standard define las caracterìsticas de las señales elèctricas, las caracteristicas del interfase mecànico y contiene una descripciòn de funcionamiento de los diferentes circuitos de intercambio.

3.3 STANDARD ISO ASINCRONICO NCR.

En esta secciòn se describe de manera general y detallada los procedimientos de control utilizados para transmisiòn de datos implementada por NCR.

Esta tesis tratarà principalmente lo relacionado a la forma de comunicaciòn implementada por los programas que constituyen este trabajo, razòn por lo cual ciertos conceptos pueden no ser aplicables a la forma en que algunos terminales la han implementado.

Tambièn cabe anotar que muchos aspectos del protocolo estàn enfocados desde el punto de vista del terminal y no desde el punto de vista del computador central.

La transferencia de datos entre el terminal y el computador central requiere que una disciplina o protocolo sea establecido. Esta disciplina es establecida por el computador central mediante el envio de secuencias de interrogaciòn (polling) y de selecciòn (selection).

"Polling" es el macanismo utilizado por el computador central para preguntar a un terminal si este tiene mensajes listos para enviar. "Selection" es el mecanismo utilizado por el computador central para informar al terminal que existe un mensaje listo para ser enviado a èl.

La transferencia de datos desde el terminal al computador central ocurre cuando el terminal detecta una secuencia de interrogaciòn que contiene su còdigo de identificaciòn (poll code) y en caso de disponer de un mensaje para enviar, el terminal responde a la secuencia de interrogaciòn con la transmisiòn del mensaje. Este mensaje empieza con STX, un encabezado (header) que contiene su identificaciòn (TA1. TA2), los datos, y ETX. El BCC es generado por el terminal y transmitido como ùltimo caràcter. Si el computador central recibe el mensaje sin errorres, un ACK es enviado al terminal. El terminal finaliza la transmisiòn con el envio de un EOT para dar fin a la comunicaciòn con el computador central hasta que este envie un nuevo còdigo de interrogaciòn o selecciòn a este terminal. En el caso de que el computador central detecte errores de transmisiòn del mensaje, enviarà un NAK; el terminal entonces retransmite el mensaje. Si luego de tres reintentos, el mensaje no es aceptado por el computador central, el terminal anula la transacciòn en proceso.

Cuando el terminal recibe un secuencia de interrogaciòn que incluye su identificaciòn pero no dispone de un mensaje para enviar, entonces transmite un EOT lo cual termina la comunicaciòn con el computador central hasta que este envie una nueva secuencia de interrogaciòn o selecciòn al terminal.

La transferencia de datos desde el computador central al terminal es iniciada por el envio de la secuencia de selecciòn a un terminal, este, en caso de estar listo a recibir dicho mensaje, envia uno que contiene la identificaciòn del terminal y un ACK (TA1-TA2-ACK), al recibir este mensaje, el computador central envia el STX, datos, ETX y BCC. Si el terminal recibe este mensaje sin errores, enviarà un ACK al que el computador central responde con un EOT para terminar la comunicaciòn con dicho terminal hasta que nuevamente envie una secuencia de interrogaciòn o selecciòn. En el caso de que el terminal detectè errores en el mensaje de datos, enviarà un NAK, a lo cual el computador central podrà responder con la retransmisiòn del mensaje o terminar la comunicaciòn.

Cuando el terminal detecta una secuencia de selecciòn pero no està listo a recibir un mensaje, este responde con un NAK.

MONITOREO DE LA LINEA

El computador central transmite secuencia de interrogaciòn o selecciòn solicitando a los terminales que envien o reciban mensajes. Este intercambio de mensajes entre computador central y terminales debe producirse dentro de ciertos limites de tiempo cortos y màximos, razòn por la cual los terminales deben realizar una funciòn de monitoreo de todo lo que el computador central envia a travès de la linea de comunicaciòn para poder identificar secuencias de

interrogaciòn o selecciòn que contengan la identificaciòn de ellos y asì poder responder eficientemante a los requerimientos del computador central.

Cada terminal dentro de un lìnea de comunicaciones tiene asignado los siguientes còdigos:

POLL CODE: Que es el còdigo de interrogaciòn al que debe responder. Si el terminal esta conectado a travès de un concentrador o controlador, compartirà este còdigo con todos los terminales conectados a dicho concentrador o controlador.

SELECT CODE: Este còdigo puede estar formado por uno o dos caracteres dependiendo de si està conectado directamente o a travès de un concentrador o controlador. En el caso de conexiòn a travès de concentradores, el primer caràcter selecciona al concentrador mientras que el segundo permite seleccionar al terminal dentro del concentrador esto quiere decir que todos los terminales conectados a un concentrador deben tener asignado un mismo primer caràcter de còdigo de selecciòn, pero el segundo tiene que ser diferente.

TA1-TA2 (terminal address 1 y 2): Estos caracteres son utilizados para identificar el origen de un mensaje recibido a travès de una lìnea de comunicaciones. Estos caracteres deben ser ùnicos dentro de un lìnea de comunicaciones y son transmitidos por el terminal dentro de dos tipos de mensajes: los de datos y los de aceptaciòn de secuencias de

selección (TA1-TA2-ACK).

Si bien en las últimas implementaciones de productos de comunicaciones tanto en computadores centrales como en terminales, existe libertad para asignar estos codigos independientemente unos de otros, productos anteriores imponen ciertas restricciones a los mismos. Debido a que existe una gran cantidad de terminales que se rigen a dichas restricciones, conviene describirlas.

- Los còdigos de interrogaciòn deben tener el quinto bit en cero (xxx0 xxxx).

- El primer caràcter del còdigo de selecciòn debe ser igual al còdigo de interrogaciòn pero con el quinto bit en uno (xxx1 xxxx).

- El segundo caràcter del còdigo de selecciòn debe existir unicamente si el terminal està conectado a travès de un concentrador o controlador, y debe ser igual a la suma hexadècimal del valor hexadècimal "40" màs el puerto o direcciòn del terminal en el concentrador o controlador (usualmente un valor hexadecimal entre 1 y E).

- TA1 debe ser igual al primer caràcter del còdigo de selecciòn.

- TA2 debe ser igual a TA1 en el caso de conexiòn directa. En el caso de conexiòn a travès de un concentrador o

controlador, debe ser igual a la suma hexadècimal del valor hexadecimal "40" màs el producto hexadecimal de 2 por el puerto o direcciòn del terminal en el concentrador o controlador.

SECUENCIA DE INTERROGACION

La secuencia de interrogaciòn es un mensaje enviado por el computador central que tiene el siguiente formato.

EOT - POLL_CODE - ENQ

El terminal debe detectar esta secuencia para poder responder a la misma. Una vez que detecte esta secuencia, debe responder ya sea con un EOT en caso de no disponer de un mensaje o con el mensaje. El computador central espera una respuesta durante un determinado tiempo (menor a 500 milisegundos) antes de enviar la secuencia de interrogaciòn al siguiente terminal.

SECUENCIA DE SELECCION

La secuencia de selecciòn es tambièn un mensaje enviado por el computador central y que tiene el siguiente formato:

EOT - SELECT_CODE - ENQ

El terminal debe detectar esta secuencia para poder responder a la misma. Una vez que la misma ha sido

detectada, el terminal debe responder ya sea con el mensaje TA1 - TA2 - ACK o con un NAK.

El terminal puede recibir mensajes cuando no está en medio de una transacción (idle) y cuando ha enviado un mensaje y está en espera de la respuesta. No puede recibir mensajes mientras está procesando una transacción y tampoco mientras está procesando un mensaje anterior.

Existen varios tipos de mensaje que el computador central puede enviar a un terminal los cuales se decriben a continuación:

- MENSAJES SOLICITADOS: Estos son mensajes enviados como respuesta a una transacción. En este caso el terminal esta en espera de dicho mensaje.

- MENSAJES NO SOLICITADOS: Estos son mensajes enviados al terminal que no constituyen una respuesta a una transacción iniciada desde dicho terminal.

- MENSAJES CON INDICACION DE ULTIMO: Estos mensajes contiene dentro de su encabezado la indicación de que este es el último mensaje. Esta indicación le sirve al terminal para no esperar más mensajes durante esta transacción. Si este no es el Ultimo, el terminal debe esperar por otros mensajes y no terminar la transacción.

SECUENCIA DE ENVIO DE UN MENSAJE

Cuando el terminal ha detectado una secuencia de interrogación y ha enviado su mensaje, el terminal sigue monitoreando la línea por una espuesta desde el computador central. En este momento pue ocurrir cualquiera de las siguientes posibilidades:

- ACK: Si el computador cent recibió el mensaje sin detectar ningun error de paridad n el BCC, envia un ACK. Al recibir el ACK, el terminal ia un EOT.

CONTADOR	TERMINAL	COMPUTADOR
		EOT-POLL_CODE-ENQ
X=0	STX-MENSAJE-EC	
		ACK
	EOT	

- **NACK:** en el caso de que central detecte algun error en la recepciòn del mensaje, envia un NACK pidiendo la retransmisiòn.

CONTADOR	TERMINAL	COMPUTADOR
		EOT-POLL_CODE-ENQ
X=0	STX-MENSAJE-ETX-BCC	
		NACK
X=1	STX-MENSAJE-ETX-BCC	
		NACK
X=2	STX-MENSAJE-ETX-BCC	
		NACK
X=3	DA POR TERMINADA LA COMUNICACION	

- **NO RESPONDE:** Si el computador central no responde en un tiempo determinado, el terminal envia un ENQ solicitando el computador central una respuesta.

CONTADOR	TERMINAL	COMPUTADOR
		EOT-POLL_CODE-ENQ
X=0	STX-MENSAJE-ETX-BCC	
		NO RESPONDE
X=1	ENQ	
		NO RESPONDE
X=2	ENQ	
		NO RESPONDE
X=3	DA POR TERMINADA LA COMUNICACION	

-EOT: Si por alguna razòn el terminal recibe un EOT como respuesta, este incrementa el contador de errores y espera un nueva secuencia de interrogaciòn para retransmitir su mensaje.

CONTADOR	TERMINAL	COMPUTADOR
		EOT-POLL_CODE-ENQ
X=0	STX-MENSAJE-ETX-BCC	
		EOT
		EOT-POLL_CODE-ENQ
X=1	STX-MENSAJE-ETX-BCC	
		EOT
X=2		
		EOT-POLL_CODE-ENQ
X=2	STX-MENSAJE-ETX-BCC	
		EOT
X=3	DA POR TERMINADA LA COMUNICACION	

SECUENCIA DE RECEPCION DE UN MENSAJE

Cuando el terminal ha sido seleccionado por el computador central y esta listo para recibir un mensaje envia en mensaje de aceptaciòn (TA1-TA2-ACK). En este momento, el computador central envia el mensaje. Las siguientes condiciones pueden ocurrir:

- ACK: Si el terminal no ha detectado ningun error, este transmite un ACK a lo que el computador central envia un EOT.

CONTADOR	TERMINAL	COMPUTADOR
		EOT-SELECT_CODE-ENQ
	TA1-TA2-ACK	
		STX-MENSAJE-ETX-BCC
	ACK	
		EOT

Como respuesta al ùltimo ACK, tambièn se puede recibir un ENQ, al cual el terminal debe retransmitir el ACK.

CONTADOR	TERMINAL	COMPUTADOR
		EOT-SELECT_CODE-ENQ
	TA1-TA2-ACK	
		STX-MENSAJE-ETX-BCC
	ACK	
		ENQ
	ACK	
		EOT

Existe otra posibilidad, y es que central no responda al ùltimo ACK transmitido, en este caso, el terminal asume que fue una transmisiòn normal.

-NACK: En el caso de que el terminal detecte un error en la recepciòn del mensaje transmite un NACK, a lo que el computador central debe retransmitir el mensaje.

CONTADOR	TERMINAL	COMPUTADOR
	_____	EOT-SELECT_CODE-ENQ
	TA1-TA2-ACK	_____
	_____	STX-MENSAJE-ETX-BCC
	NACK	_____
	_____	STX-MENSAJE-ETX-BCC
	ACK	_____
	_____	EOT

En este caso tambièn se puede recibir un ENQ como respuesta al NACK.

CONTADOR	TERMINAL	COMPUTADOR
	_____	EOT-SELECT_CODE-ENQ
	TA1-TA2-ACK	_____
	_____	STX-MENSAJE-ETX-BCC
	NACK	_____
	_____	ENQ
	NACK	_____
	_____	STX-MENSAJE-ETX-BCC
	ACK	EOT

Tambièn podria recibirse un EOT.

TERMINAL	COMPUTADOR
_____	EOT-SELECT_CODE-ENQ
TA1-TA2-ACK	_____
_____	STX-MENSAJE-ETX-BCC
NACK	_____
_____	EOT

En caso de no obtener respuesta al NACK, el tratamiento es similar al de EOT.

4.1 DISEÑO GENERAL

El diseño de este producto se dividio en los siguientes aspectos:

- diseño funcional,
- diseño de estructura de datos y archivos y
- diseño de programas.

4.1.1 DISEÑO FUNCIONAL

Bajo este título, se describen los siguientes criterios de funcionamiento considerados en este producto:

- El protocolo de comunicaciones exige un monitoreo continuo de la línea de comunicaciones, razón por la cual este es un trabajo que debe funcionar en "tiempo real". Indiscutiblemente, la ùnica alternativa eficiente es la de ùtilizar tècnicas de interrupciones disponibles en el computador personal NCR PC4i, que incluye dentro de sus componentes el microprocesador 8088, el controlador de interrupciones 8259A y el adaptador RS-232-C con un 8250. Estas rutinas de interrupciòn deben tener la capacidad de atender el protocolo independientemente del trabajo que realice el operador del PC.

- Debe funcionar como un producto completo en el sentido de que el usuario solamente requiera definir sus transacciones, pero no requiera realizar modificaciones a los programas producto de esta tesis.

- El programa generador de transacciones y formatos de pantalla debe brindar facilidades al usuario.

- El proceso de una transacciòn debe incluìr los siguientes pasos:
  - presentaciòn del formato de pantalla asociado a la transacciòn seleccionada por el operador,
  - control de ingreso y validaciòn de datos,
  - envìo del mensaje al computador central, y
  - procesamiento del mensaje del computador central

- Se debe permitir al computador central ciertas funciones de control de pantalla e impresora a travès de còdigos de funciòn, enviados dentro del mensaje de respuesta tales como:
  - cambiar el formato de pantalla,
  - llenar datos sobre un formato,
  - llenar datos en una linea y posiciòn especificada por el computador central,
  - imprimir el contenido de la pantalla,
  - anular una transacciòn,
  - imprimir una linea en la impresora,
  - saltar pàgina,
  - avanzar "n" lineas.

4.1.2 DISEMO DE ESTRUCTURAS DE DATOS Y ARCHIVOS

Bajo este título se describe la razòn de ser y la conformaciòn de cada uno de los archivos y estructuras de datos utilizados por los diferentes programas que constituyen este trabajo.

Los archivos son requeridos por varias razones:

- el proyecto està dividido en dos programas principales y una rutina de manejo de comunicaciones. De los dos programas principales, el primero permite definir paràmetros, formatos de pantalla, etc que el segundo programa debe utilizarlos, por lo tanto el primer programa debe almacenar dicha informaciòn en un medio magnètico,

- este trabajo està orientado a permitir al usuario definir sus propias transacciones, por lo que todas las necesidades de un usuario especifico debe almacenarse en medio magnètico y

- los requerimientos de formatos y transacciones que un usuario requiera no puede estar limitado a la memoria del equipo por lo que un almacenamiento magnetico es requerido.

Los siguientes archivos han sido incluìdos en este trabajo y van ha ser descritos a continuaciòn:

- archivo de formatos de pantalla (formatos.fl)
- archivo de definiciòn de transacciones (deftran.fl)
- archivo de definiciòn de reglas de validaciòn (defval.fl)
- archivo de grupos de transacciones (grupos.fl)
- archivo de paràmetros (param.fl)

El lenguaje utilizado (BASIC) permite ùnicamente la utilizaciòn de archivos relativos y secuenciales; todos los archivos ya mencionados requieren de acceso directo, razòn por la que todos tienen organizaciòn relativa.

Si bien los archivos relativos permiten un acceso directo, este se realiza ùnicamente a travès de su nùmero de registro relativo el cual no siempre es la llave de la informaciòn deseada. Por ejemplo, si se desea definir el formato de pantalla nùmero 600, existen dos alternativas: la primera: almacenar el formato en el registro 600, lo cual constituye una ineficiencia de utilizaciòn de espacio en disco; la segunda es almacenar dicho formato en el siguiente registro no utilizado en el archivo en ese momento, y crear un puntero que corresponda al formato 600 en una tabla. Este tipo de mecanismos utilizados se describirà tambièn en detalle màs adelante.

ARCHIVO DE FORMATOS DE PANTALLA: formatos.fl

ORGANIZACION: RELATIVA

LONGITUD DEL REGISTRO: 777 CARACTERES

Este archivo contiene información de control de asignación de registros del archivo e información de todos los formatos de pantalla definidos.

Existen tres tipos diferentes de registros en este archivo: de control, de índice y de datos.

REGISTRO DE CONTROL

Este es el primer registro del archivo y su única función es la de llevar un control de que registros han sido utilizados en el archivo y de cuales estan libres, para lo cual se mantiene un bit para representar cada registro, si el bit correspondiente es igual a cero (0), el registro está libre y puede ser asignado, si es uno (1), el registro está ocupado y no puede ser utilizado.

La información de este registro es utilizada únicamente por el programa de definición de formatos, el cual prende un bit cuando utiliza un registro y lo apaga cuando el formato almacenado en el registro correspondiente ha sido eliminado.

La información inicial de este registro es también grabada por el programa de definición de formatos al detectar que este registro no existe (la primera vez).

La siguente es la definición de este registro.

- campo 1: 2 caracteres (bytes) que contienen el número de este registro, en este caso un 1.
- campo 2: 10 caracteres que en este registro no son utilizados.
- campo 3: 765 caracteres que representan 6120 bits, cada uno de los cuales representa un registro del archivo, por ejemplo el tercer bit del primer carácter representa al tercer registro; el tercer bit del tercer carácter representa al registro número 19.

REGISTROS DE INDICE

Estos registros son utilizados para almacenar punteros a los registros de datos. Un puntero es un entero (2 caracteres) que contiene la llave de registro.

Existen tres registros de este tipo en el archivo, y corresponden a los registros 2,3,4 los mismos que forman una tabla de 2262 caracteres en la cual se pueden almacenar 1131 punteros. Cada posición de esta tabla está reservada para un número de formato, es decir los primeros dos caracteres de esta tabla contienen el puntero a los datos del formato

de pantalla nùmero uno, los dos siguientes caracteres de la tabla contienen el puntero para el formato nùmero dos y asi consecutivamente. Si un formato no ha sido definido, el puntero correspondiente tiene un valor de cero (0).

La siguiente es la definiciòn de este tipo de registro:

- campo 1: 2 caràcteres que contienen el nùmero de registro. Puede ser 2,3 o 4.
- campo 2: 10 caràcteres no utilizados en este tipo de registro.
- campo 3: 755 caracteres de los cuales los primeros 754 constituyen la tabla de 2262 caracteres. El registro nùmero 2 contiene los primeros 754 caracteres de la tabla, el registro nùmero 3 los siguientes, y el registro nùmero 4 los ùltimos.

REGISTRO DE DATOS

Este tipo de registro contiene datos de un formato de pantalla, el mismo que que definido mediante la siguiente informaciòn:

- informaciòn constante del formato: que representan titulo, nombres de campos, etc., en general es informaciòn que no puede ser modificada por el operador.
- informaciòn de datos de entrada: es informaciòn que permite al programa de terminal, aceptar un campo en una posiciòn de la pantalla especifica (lìnea y posiciòn), por una longitud dada, y ademàs permite verificar el tipo de informaciòn (nùmerico, alfanùmerico, etc). asociado a cada

dato de entrada; està tambièn el còdigo de funciòn que debe enviarse al computador central para que la transacciòn pueda ser identificada.

- informaciòn de campos de salida: que constituye la lìnea y posiciòn en la pantalla y la longitud.

Debido a que los formatos de pantalla son definidos por el usuario, estos no pueden ajustarse a definiciones fijas, por lo que es necesario manejar datos de longitudes variables. Como se detallo anteriormente, existen tres tipos de informaciòn que defines un formato: campos constantes, campos de entrada, y campos de salida. No es conveniente limitar ni el nùmero de constantes, ni el nùmero de campos de entrada y tampoco el de salida, por lo que la estructura del registro que se describe a continuaciòn es una buena soluciòn.

- campo 1: 2 caracteres que contienen el nùmero de registro.
- campo 2: 10 caracteres que tienen el siguiente significado:
  - 2 caracteres que contienen el nùmero del formato,
  - 2 caracteres que contienen el puntero al siguiente registro de datos de este formato. Utilizado unicamente en formatos muy extensos.
  - 2 caracteres que contienen la longitud de la definiciòn de los datos constantes.
  - 2 caracteres que contienen la longitud de las definiciones de datos constantes y de campos de entrada,

- 2 caracteres que contienen la longitud de las definiciones de datos constantes, de campos de entrada y de campos de salida.
- campo 3: 755 caracteres en que se almacena la informaciòn de campos constantes seguida por la definiciòn de campos de entrada y de la definiciòn de campos de salida.

La definiciòn de un campo constante esta dada por la siguiente informaciòn:

- 1 caràcter que contiene la linea en que esta constante debe ser presentada,
- 1 caràcter que contiene la posiciòn inicial en esa linea en que la constante debe ser presentada,
- 1 caràcter que contiene la longitud de la constante y
- Los "n" caractsres que contienen la constante. Cabe anotar que màs de tres espacios consecutivos no son almacena dos en esta estructura.

La definiciòn de un campo de entrada està dada por la siguiente informaciòn:

- 1 caràcter que contiene el còdigo de funciòn asociado a este campo y que debe ser transmitido al computador central,
- 1 caràcter que contiene la longitud de este campo,
- 1 caràcter que contiene la linea en que este campo debe ser aceptado,
- 1 caràcter que contiene la posiciòn inicial en esa linea en que este campo debe ser aceptado,
- 1 caràcter que contiene un puntero a una tabla de

reglas de validaciòn de este campo. Esta tabla està almacenada en el archivo defval.fl.

- Los "n" caracteres que contienen el valor inicial de este campo (default) y que es opcional.

La definiciòn de un campo de salida esta dada por la siguiente informaciòn:

- 1 caràcter que contiene la longitud del campo,
- 1 caràcter que contiene la linea en que debe presentarse dicho campo y
- 1 caràcter que contiene la posiciòn inicial en esa linea de este campo.

ARCHIVO DE DEFINICION DE TRANSACCIONES: deftran.fl

ORGANIZACION: RELATIVA

LONGITUD DEL REGISTRO: 102 CARACTERES

Este archivo contiene dos tipos de registro: de control y de datos. El de control permite asignar los registros del archivo de la manera màs eficiente, los de datos, contienen informaciòn relacionada a la transacciòn.

REGISTRO DE CONTROL

Este es el primer registro del archivo y su ùnica funciòn es la de llevar un control de los registros asignados y de los registros libres del archivo. Para lograr lo anterior, este registro mantiene un bit asociado a cada

uno de los registros del archivo, este bit tiene el valor de cero (0) cuando el registro está libre y tiene el valor de uno (1) cuando está asignado.

El contenido de este registro es manejado únicamente por el programa de definición de transacciones.

La siguiente es la definición de este registro:

- campo 1: 2 caracteres que contienen el número del registro, en este caso un 1.
- campo 2: 100 caracteres que contiene 800 bits, cada uno de los cuales representa un registro del archivo, por ejemplo: el primer bit del segundo carácter representa al registro 9.

REGISTRO DE DATOS

Existe un registro de este tipo por cada transacción definida por el usuario. En este tipo de registro se almacena la información que constituye la definición de una transacción la misma que incluye lo siguiente:

- grupo al que pertence,
- nombre de la transacción,
- nivel de seguridad, y
- pantalla asociada.

La siguiente es la definicòn de este registro:

- campo 1: 2 caracteres que contienen el número del registro.

- campos 2: 1 caràcter que contiene el grupo al que esta transacciòn pertenece. puede tener un valor entre 1 y 9.
- campo 3: 2 caracteres que contirnen el nùmero de la transacciòn.
- campo 4: 2 caracteres que contienen el nivel de seguridad de la transacciòn. puede tener un valor entre 1 y 132.
- campo 5: 30 caracteres que contienen el nombre de la transacciòn.
- campo 6: 6 caracteres que identifican la transacciòn en el computador central.
- campo 7: 2 caracteres que contienen el nùmero del formato asociado con esta transacciòn.
- campo 8: 2 caracteres que contienen un puntero al registro del archivo de formatos que contiene el formato asociado con esta transacciòn.

ARCHIVO DE DEFINICION DE REGLAS DE VALIDACION: defval.fl

ORGANIZACION: RELATIVA

LONGITUD DEL REGISTRO: 102 CARACTERES

Este archivo contiene las reglas de validaciòn de campos de entrada. Una regla de validaciòn es un conjunto de banderas que definen los procedimientos a ejecutarse cuando el operador ingrese un campo de entrada de un formato de pantalla.

La idea principal de mantener esta informaciòn

separada del archivo de formatos, es la de permitir que uno o más formatos compartan la definición de un campo de entrada.

La siguiente información constituye una regla de validación:

- tipo de campo: el cual puede ser número, alfanúmerico, alfabético, clave, fecha, cuenta.
- formato de fecha: que identifica si el formato es AAMMDD, DDMMAA, MMDDAA.
- tipo de dígito verificador: que permite especificar hasta tres formas de cálculo del dígito verificador de una cuenta.
- indicador de valor inicial del campo: que permite procesar campos de entrada con un valor inicial.

Este archivo tiene dos tipos de registros, el de control y el de datos.

REGISTRO DE CONTROL

Este registro permite controlar la asignación de los registros del archivo, para lo cual se mantienen bits que representan registros. La siguiente es la definición de este registro:

- campo 1: 2 caracteres que contienen el número de este registro, en este caso un 1.

- campo 2: 100 caracteres que contienen 800 bits, cada uno de los cuales representa un registro del archivo. Cada bit tiene el valor de cero (0) cuando el registro està libre y un valor de uno (1) cuando el registro està ocupado.

REGISTRO DE DATOS

En este tipo de registro se almacenan todas las reglas de validaciòn que un usuario requiera. La creaciòn de una nueva regla es automàtica y se realiza al momento de definir los campos de entrada en la definiciòn de los formatos de pantalla.

En un registro se puede almacenar hasta diez reglas de validaciòn. Para cada regla de validaciòn se han reservado diez caracteres, de los cuales solo uno es utilizado actualmente.

La definiciòn de este tipo de registro es la siguiente:

- campo 1: 2 caracteres que contienen el nùmero de registro.
- campo 2: 100 caracteres en que se agrupan 10 reglas de validaciòn. Cada regla de validaciòn contienen diez caracteres de los cuales solo el primero tiene utilidad y el siguiente significado:

- primer caràcter: cada uno de los ocho bits del primer caràcter tienen el siguiente significado:
  - bits 7 y 6: tiene la siguiente codificaciòn:

    11 = campo alfanùmerico

    10 = campo alfabètico

    01 = campo nùmerico
  - bit 5: es 1 cuando el campo es una fecha
  - bit 4 y 3 : tiene la siguiente codificaciòn cuando el campo no es una fecha:

    00 el campo no es una clave

    11 clave alfanùmerica

    Si el campo es una fecha, estos dos bits indican el formato de la fecha de acuerdo a la siguiente codificaciòn:

    01 AAMMDD

    10 DDMMAA

    11 MMDDAA
  - bit 2 y 1: no utilizados
  - bit 0: es 1 si el campo tiene un valor inicial.

ARCHIVO DE GRUPOS DE TRANSACCIONES: grupos.fl

ORGANIZACION: RELATIVA

LONGITUD DEL REGISTRO: 510 CARACTERES

La funciòn de este archivo es la de agrupar las diferentes transacciones en grupo de acuerdo a las necesidades del usuario.

Este archivo tiene ùnicamente nueve registros, uno por cada grupo que pueden existir. Dentro de cada registro se almacena los numeros de transacciones y los punteros al archivo de definiciòn de transacciones.

La definiciòn de estos registros es la siguiente:

- campo 1: 2 caracteres que contienen el nùmero del registro (1-9).
- campo 2: 254 caracteres que permiten almacenar 127 nùmeros de transacciòn ya que cada nùmero se almacena en 2 caracteres.
- campo 3: 254 caracteres que permiten almacenar los 127 punteros correspondientes a los nùmeroa de transacciòn almacenados en el campo 2.

ARCHIVO DE PARAMETROS: param.fl

ORGANIZACION: RELATIVA

LONGITUD DEL REGISTRO: 102 CARACTERES

La función principal de este archivo es la de almacenar parámetros referentes a un terminal en particular, estos son : el código de interrogación y la velocidad de transmisión. Sin embargo, este archivo podría utilizarse para otros propositos, tales como almacenar códigos de operadores y claves de acceso, etc.

El archivo tiene un solo registro el cual tiene la siguiente definición:

- campo 1: 2 caracteres que contienen el valor del número del registro, en este caso un 1.
- campo 2: 2 caracteres que contienen el código de interrogación.
- campo 3: 5 caracteres que contienen la velocidad de transmisión.

4.1.3 DISEÑO DE PROGRAMAS.

Existen dos ambientes de trabajo, el de definiciòn y el de ejecuciòn. En el ambiente de definiciòn, existe un solo programa que controla el mantenimiento de los datos y paràmetros requeridos por el ambiente de ejecuciòn, mientras que en el ambiente de ejecuciòn, existen dos funciones que deben ejecutarse concurrentemente, la primera, el "driver" de comunicaciones y la segunda el "interprete" de los parametros y datos definidos por el usuario.

PROGRAMA DE MANTENIMIENTO DE DATOS

Este programa ha sido diseñado para trabajar de manera interactiva con el operador, para lo cual hace uso de menus. El programa consta de siete mòdulos principales:

- MODULO DE CONTROL: las funciones de este mòdulo son las de llamar a uno de los nueve mòdulos siguientes dependiendo de la opciòn selecionada por el operador.

- MODULO DE CREACION DE TRANSACCIONES: Este mòdulo permite al operador ingresar todos los datos requeridos para la definiciòn de una transacciòn, verificar dichos datos y almacenar la informaciòn en los archivos correspondientes.

- ELIMINACION DE TRANSACCIONES: Este mòdulo permite eliminar transacciones.

RUTINA DE COMUNICACIONES

Esta rutina esta diseñada para funcionar mediante interrupciones. Para su funcionamiento, es necesario cargar esta rutina , alterar la direcciòn de la rutina de servicio de la interrupciòn ,programar el 8250 y activar la interrupciòn en el mismo.

Al utilizar interrupciones, esta rutina utilizara el minimo tiempo de procesador requerido por lo que la ejecuciòn del programa "intèrprete" se realizarà eficientemente.

Las funciones de esta rutina son:

- las de recibir el caràcter que causò la interrupciòn,
- analizar el estado en que la comunicaciòn se encontraba,
- identificar secuencias de interrogaciòn y de selecciòn,
- calcular el BCC (block check character) ya sea para transmitirlo o verificarlo,
- detectar errores de paridad,
- informar al intèrprete la presencia de un mensaje,
- recibir mensajes del intèrprete para enviarlos al computador central,
- aceptar o rechazar secuencias de selecciòn dependiendo de instrucciones del interprete.
- procesar errores de transmisiòn y recepciòn de acuerdo a las reglas del protocolo.

INTERPRETE

Este programa procesa los paràmetros definidos por el usuario y almacenados en los archivos. Las funciones principales de este programa son:

- presentar los formatos de pantalla al operador,
- controlar el ingreso de la informaciòn de una transacciòn y verificar dicha informaciòn respecto a paràmetros o reglas de validaciòn definidas,
- comunicarse apropiadamente con el "driver" para enviar mensajes al computador central,
- recibir mensajes desde el "driver", y procesar cada uno de los campos o datos enviados de acuerdo a reglas fijas,
- preguntar al operador por una nueva transacciòn.

Los mòdulos principales que forman este programa son:

- INICIALIZACION: El cual abre los archivos requeridos, crea ciertas estructuras, inicializa el driver de comunicaciones y pasa el control al siguiente mòdulo.

- "IDLE" : La ùnica funciòn de este mòdulo es la de esperar el ingreso de un còdigo de transacciòn por parte del operador.

- PROCESO DE LA TRANSACCION: Este mòdulo controla el flujo de la transacciòn el cual consta de la presentaciòn del

4.2 GENERADOR DE TRANSACCIONES Y FORMATOS

A continuación se incluye el listado del programa que permite definir las transacciones y los formatos de pantalla asociados.

```
1 '  ==================================================
2 '  PROGRAMA DE DEFINICION DE TRANSACCIONES
5 '  ESTE PROGRAMA FORMA PARTE DE LA TESIS DE GRADO DE
6 '  ALBERTO ANDRADE VAREA.
10 ' ==================================================
20 '_____________________________________________________'
22 '     RUTINA DE CONTROL:  ESTA RUTINA ABRE LOS        '
24 '        ARCHIVOS, CREA LOS ARREGLOS REQUERIDOS (DIM) '
26 '        LLAMA AL MENU PRINCIPAL, Y DEPENDIENDO DE LA '
28 '        RESPUESTA (Q$), LLAMA A UNO DE LOS SIES  MO- '
30 '        DULOS: DEFINICION DE TRANSACCIONES,          '
32 '               ELIMINACION DE TRANSACCIONES,         '
34 '               IMPRESION DE TRANSACCIONES,           '
36 '               DEFINICION DE FORMATOS,               '
38 '               ELIMINACION DE FORMATOS               '
40 '               IMPRESION DE FORMATOS                 '
42 '_____________________________________________________'
50 DISK$="a"
100 DEFVALFL$=DISK$+":DEFVAL.FL"
150 GRUPOSFL$=DISK$+":GRUPOS.FL"
200 DEFTRANFL$=DISK$+":DEFTRAN.FL"
250 FORMATOSFL$=DISK$+":FORMATOS.FL"
300 PARAMFL$=DISK$+":PARAM.FL"
350 DIM FORD$(20):DIM FORDP$(20):DIM PD$(60):DIM FORREC2$(3)
400 DIM ACC$(10):DIM MESS$(12)
450 KEY OFF
500 FOR I=1 TO 10:KEY(I) OFF:NEXT I
550 DIM RV$(200):DIM WRV$(60):DIM WOV$(60):DIM DEFULT$(60)
600 GOSUB 16000    ' ABRIR ARCHIVOS
650 GOSUB 850      ' MENU PRINCIPAL
700 IF Q$="0" THEN CLS:END
750 ON VAL(Q$) GOSUB 2000,6250,34150,8600,14050,35850
800 GOTO 650
850 '_____________________________________________________'
900 '     MENU PRINCIPAL: ESTA RUTINA PRESENTA UN MENU CON '
950 '     LAS DIFERENTES OPCIONES Y ACEPTA DEL OPERADOR    '
1000 '     EL NUMERO DE SELECCION, EL MISMO QUE ES VERI-    '
1050 '     FICADO. LA RUTINA REGRESA EL CONTROL AL MODULO   '
1100 '     DE CONTROL.                                      '
1150 '_____________________________________________________'
1200 COLOR 7,0:CLS:GOSUB 41050
1250 LOCATE 2,22:PRINT "DEFINICION DE TRANSACCIONES";
1300 LOCATE 3,22:PRINT "==========================";
```

```
1350 LOCATE 5,22:PRINT "0..FIN";
1400 LOCATE 6,22:PRINT "1..DEFINIR TRANSACCIONES";
1450 LOCATE 7,22:PRINT "2..ELIMINAR TRANSACCIONES";
1500 LOCATE 8,22:PRINT "3..IMPRIMIR TRANSACCIONES";
1550 LOCATE 9,22:PRINT "4..DEFINIR FORMATOS";
1600 LOCATE 10,22:PRINT "5..ELIMINAR FORMATOS";
1650 LOCATE 11,22:PRINT "6..IMPRIMIR FORMATOS";
1700 LOCATE 17,22:PRINT "OPCION = ";
1750 LOCATE 17,34,1:Q$=INKEY$:LOCATE 17,34:DEF SEG=&HB000
1800 POKE -32767+2*(13*13),112:LOCATE 17,34:PRINT Q$;
1850 IF LEN(Q$)=0 THEN GOTO 1750
1900 IF Q$>"6" THEN MESS=1:GOSUB 42150:GOTO 1750
1950 RETURN
2000 '_____________________________________________________'
2050 '    DEFINICION DE TRANSACCIONES: ESTA RUTINA          '
2100 '    ACEPTA LOS DATOS REQUERIDOS PARA DEFINIR UNA      '
2150 '    TRANSACCION, VERIFICA QUE LA TRANSACCION NO       '
2200 '    EXISTA, Y QUE EL NUMERO DE PANTALLA SI EXISTA     '
2250 '    Y ACTUALIZA LOS ARCHIVOS                          '
2300 '_____________________________________________________'
2350 KEY(1) OFF
2400 KEY(2) OFF
2450 KEY(3) OFF
2500 COLOR 7,0,1:CLS:GOSUB 41050
2550 LOCATE 2,22:PRINT "DEFINICION DE TRANSACCIONES";
2600 LOCATE 3,22:PRINT "==========================";
2650 LOCATE 5,22:PRINT "GRUPO:";
2700 LOCATE 6,22:PRINT "NUMERO DE TRANSACCION:";
2750 LOCATE 7,22:PRINT "NOMBRE DE LA TRANSACCION:";
2800 LOCATE 8,22:PRINT "PROGRAMA DE APLICACION:";
2850 LOCATE 9,22:PRINT "NUMERO DE PANTALLA:";
2900 LOCATE 23,1
2950 PRINT CHR$(16)+"GRUPO DE TRANSACCION  "+CHR$(17)
3000 II=5:JJ=34:WLEN=1:GOSUB 20950    '____ GRUPO
3050 IF Q%=0 THEN BEEP:GOTO 2900
3100 GR=Q%
3150 IF GR=0 THEN GOTO 2900
3200 TGIRRN=GR
3250 GET #3,TGIRRN
3300 LOCATE 23,1
3350 PRINT CHR$(16)+"0001 - 9998.  ENTRE 9999 PARA CANCELAR"
     +CHR$(17)
3400 II=6:JJ=49:WLEN=4:GOSUB 20950    '____ NUM TRAN
3450 IF Q%=0 THEN BEEP:GOTO 3300
3500 IF Q%=9999 THEN MESS=8:GOSUB 42150:RETURN
3550 NUMT=Q%
3600 I=1
3650 IF MID$(TGI1$,I,2) = STRING$(2,255) THEN GOTO 3800
3700 IF  MID$(TGI1$,I,2)  =  MKI$(NUMT)  THEN  MESS=9:
     GOSUB 42150:GOTO 3300
3750 I=I+2:GOTO 3650
3800 TOFF%=I
3850 LOCATE 23,1:
3900 PRINT CHR$(16)+"NOMBRE DE LA TRANSACCION "+CHR$(17)
3950 II=7:JJ=54:WLEN=30:GOSUB 21600    '____ NOMBRE TRAN
```

```
4000 NOMT$=QQ$
4050 LOCATE 23,1
4100 PRINT CHR$(16)+"PROGRAMA DE APLICACION EN CENTRAL"+CHR$(17
4150  II=8:JJ=54:WLEN=6:GOSUB  21600  '__ PROGRAMA  DE CENTRAL
4200 PROG$=QQ$
4250 LOCATE 23,1
4300 PRINT CHR$(16)+"001 - 998. ENTRE 999 PARA CANCELAR"+CHR$(1
4350 II=9:JJ=54:WLEN=3:GOSUB 20950  '__ NUMERO DE PANTALLA
4400 IF Q%=999 THEN MESS=8:GOSUB 42150:RETURN
4450 IF Q%=0 THEN BEEP:GOTO 4250
4500 FORRNUM%=Q%
4550 Q%=2*Q%
4600 WIDX%=1+INT((Q%-2)/764)
4650 WOFF%=Q%
4700 IF WOFF%>764 THEN WOFF%=WOFF%-764:GOTO 4700
4750 WOFF%=WOFF%-1
4800   IF  MID$(FORREC2$(WIDX%),WOFF%,2)=STRING$(2,0)   THEN
       MESS=4:GOSUB 42150:GOTO 4250
4850 J=0
4900 J=J+1
4950 IF MID$(TDP$,J,1)=CHR$(255) THEN GOTO 4900
5000 WRRN=(J-1)*8:WRRN1=0
5050 WRRN1=8-INT(LOG((255 XOR ASC(MID$(TDP$,J,1))))/LOG(2))
5100 WRRN=WRRN+WRRN1
5150 Q$=SPACE$(100)
5200 MID$(Q$,1,1)=CHR$(GR)
5250 MID$(Q$,2,2)=MKI$(NUMT)
5300 MID$(Q$,4,2)=MKI$(SECLEV)
5350 MID$(Q$,6,30)=NOMT$
5400 MID$(Q$,36,6)=PROG$
5450 MID$(Q$,42,2)=MKI$(FORRNUM%)
5500 MID$(Q$,44,2)=MID$(FORREC2$(WIDX%),WOFF%,2)
5550 LSET TD$=Q$
5600 RSET TDF$=MKI$(WRRN)
5650 PUT #4,WRRN
5700 TDRRN=1
5750 Q%=ASC(MID$(TDP$,J,1))+INT(256/(2^(WRRN1))):
     MID$(TDP$,J,1)=CHR$(Q%)
5800 LSET TD$=TDP$
5850 RSET TDF$=MKI$(TDRRN)
5900 PUT #4,TDRRN
5950 MID$(TGI1$,TOFF%,2)=MKI$(NUMT)
6000 MID$(TGI2$,TOFF%,2)=MKI$(WRRN)
6050 RSET TGIF$=MKI$(TGIRRN)
6100 LSET TGI1$=TGI1$:LSET TGI2$=TGI2$
6150 PUT #3,TGIRRN
6200 MESS=11:GOSUB 42150:RETURN
6250 '_____________________________________________________'
6300 '     ELIMINACION DE TRANSACCIONES: ESTA RUTINA       '
6350 '     ACEPTA EL GRUPO Y EL NUMERO DE LA TRANSACCION   '
6400 '     QUE DEBE ELIMINARSE, VERIFICA QUE DICHA TRAN-   '
6450 '     SACCION EXISTA, Y PROCEDE A ELIMINAR LA INFOR-  '
6500 '     MACION DE LOS DIFERENTES ARCHIVOS.              '
6550 '_____________________________________________________'
6600 COLOR 7,0,1:CLS:GOSUB 41050
```

```
6650 LOCATE 2,22:PRINT "ELIMINACION DE TRANSACCIONES";
6700 LOCATE 3,22:PRINT "============================";
6750 LOCATE 5,22:PRINT "GRUPO:";
6800 LOCATE 6,22:PRINT "NUMERO DE TRANSACCION:";
6850 LOCATE 23,1
6900 PRINT CHR$(16)+"GRUPO DE TRANSACCION:  1 - 9"+CHR$(17)
6950 II=5:JJ=34:WLEN=1:GOSUB 20950     '____ GRUPO
7000 IF Q%=0 THEN BEEP:GOTO 6850
7050 GR=Q%
7100 TGIRRN=GR
7150 GET #3,TGIRRN
7200 LOCATE 23,1
7250 PRINT CHR$(16)+"0001 - 9998.   ENTRE 9999 PARA CANCELAR
     "+CHR$(17)
7300 II=6:JJ=49:WLEN=4:GOSUB 20950     '____ NUM TRAN
7350 IF Q%=0 THEN BEEP:GOTO 7200
7400 IF Q%=9999 THEN MESS=8:GOSUB 42150:RETURN
7450 NUMT=Q%
7500 I=1
7550  IF MID$(TGI1$,I,2) = STRING$(2,255) THEN  MESS=2:
     GOSUB 42150:GOTO 7200
7600 IF MID$(TGI1$,I,2) = MKI$(NUMT) THEN GOTO 7700
7650 I=I+2:GOTO 7550
7700 TDRRN%=CVI(MID$(TGI2$,I,2))
7750 GET #4,TDRRN%
7800 FORRNUM%=CVI(MID$(TD$,42,2))     '____ NUM PANTALLA
7850 FOR I=I TO 250 STEP 2
7900 MID$(TGI1$,I,2)=MID$(TGI1$,I+2,2)
7950 MID$(TGI2$,I,2)=MID$(TGI2$,I+2,2)
8000 NEXT I
8050 MID$(TGI1$,252,2)=STRING$(2,255)
8100 MID$(TGI2$,252,2)=STRING$(2,255)
8150 LSET TGI1$=TGI1$:LSET TGI2$=TGI2$
8200 PUT #3,TGIRRN
8250 I%=INT((TDRRN%-1)/8)
8300 J%=8-TDRRN%+8*I%
8350 MID$(TDP$,I%+1,1)=CHR$(ASC(MID$(TDP$,I%+1,1))-2^J%)
8400 TDRRN=1
8450 LSET TD$=TDP$
8500 PUT #4,TDRRN
8550 MESS=11:GOSUB 42150:RETURN
8600 '_____________________________________________________'
8650 '    DEFINICION DE FORMATOS: ESTA RUTINA ACEPTA EL     '
8700 '    NUMERO DE FORMATO A DEFINIRSE, VERIFICA QUE       '
8750 '    DICHO NUMERO NO EXISTA, LLAMA A LA RUTINA DE      '
8800 '    ACEPTAR LOS DATOS DEL FORMATO, LLAMA A LAS        '
8850 '    RUTINAS QUE PROCESAN LOS CAMPOS DE ENTRADA, DE    '
8900 '    SALIDA Y CONSTANTES, Y A LA RUTINA QUE GRABA      '
8950 '    LA INFORMACION EN DISCO.                          '
9000 '_____________________________________________________'
9050 KEY(1) ON        'DEFINICION DE CAMPOS DE ENTRADA
9100 KEY(2) ON        'DEFINICION DE CAMPOS DE SALIDA
9150 KEY(3) ON     'DEFINICION DE CAMPOS DE ENTRADA Y SALIDA
9200 COLOR 7,0,1:CLS:GOSUB 41300
9250 LOCATE 2,22:PRINT "DEFINICION DE FORMATOS";
```

```
9300 LOCATE 3,22:PRINT "=======================";
9350 LOCATE 5,22:PRINT "NUMERO DE PANTALLA:";:II=5:JJ=49:WLEN=
     QQ$="":GOSUB 20950
9400 IF Q%=0 THEN BEEP:GOTO 9350
9450 IF Q%=999 THEN MESS=8:GOSUB 42150:RETURN
9500 FORNUM%=Q%     '__ LAS SIGUIENTES INSTRUCCIONES VERIFICAN
9550  Q%=2*Q%       '    QUE  EL  FORMATO  NO EXISTA
9600 WIDX%=1+INT((Q%-2)/764)
9650 WOFF%=Q%
9700 IF WOFF%>764 THEN WOFF%=WOFF%-764:GOTO 9700
9750 WOFF%=WOFF%-1
9800 WNULL$=CHR$(0)+CHR$(0)
9850 IF  MID$(FORREC2$(WIDX%),WOFF%,2)<>WNULL$     THEN
     MESS=3:GOSUB 42150:GOTO 9350
9900 MESS=11:GOSUB 42150:COLOR 0,2,1:CLS:LOCATE 25,1
9950 PRINT "F1=ENTRADA     F2=SALIDA     F3=ENTRADA/SALIDA";
10000 FOR I=1 TO 20
10050 FORD$(I)=SPACE$(80):FORDP$(I)=FORD$(I)
10100 NEXT I
10150 I=1:J=1:II=1:JJ=1
10200 DEF SEG=&HB000:
      WATRIB%=PEEK(-32767+2*(80*(I-1)+(J-1))):
      ATR%=WATRIB%
10250 FOR RVIDX = 1 TO 60
10300 WRV$(RVIDX)="":DEFULT$(RVIDX)="":WOV$(RVIDX)=""
10350 NEXT RVIDX
10400 RVIDX=1:OVIDX=1
10450 GOSUB 10700    '____INGRESAR DATOS
10500 GOSUB 32200    '____PROCESAR CAMPOS DE ENTRADA
10550 GOSUB 33550    '____PROCESAR CAMPOS DE SALIDA
10600 GOSUB 29000    '____COMPACTAR DATOS Y GRABAR EN ARCHIVOS
10650 RETURN
10700 '_____________________________________________________________'
10750 '  INGRESO DE DATOS DE LA PANTALLA:  ESTA RUTINA  CON-  '
10800 '  TROLA EL INGRESO DE LOS DATOS DE UN FORMATO.    EL   '
10850 '  OPERADOR PUEDE MOVER EL CURSOR LIBREMENTE  USANDO   '
10900 '  LAS TECLAS DE CONTROL.  MEDIANTE LAS  TECHAS <F1>   '
10950 '  <F2> Y <F3>,  EL OPERADOR PUEDE DEFINIR CAMPOS  DE  '
11000 '  ENTRADA,   SALIDA   O   ENTRADA   Y   SALIDA.        '
11050 '_____________________________________________________________'
11100 ON KEY(1) GOSUB 22200    '__ SI <F1> CAMPO DE ENTRADA
11150 ON KEY(2) GOSUB 22350    '__ SI <F2> CAMPO DE SALIDA
11200 ON KEY(3) GOSUB 22500    '__ SI <F3> CAMPO DE ENTRADA Y
                                   SALIDA
11250 IF J>80 THEN J=J-80:I=I+1:IF I>20 THEN RETURN
11300 LOCATE 25,60:PRINT I;J;RVIDX-1;OVIDX-1;:LOCATE I,J
11350 DEF SEG=&HB000:POKE -32767+2*(80*(I-1)+(J-1)),15
11400 Q$=INKEY$
11450 IF LEN(Q$)=0 THEN GOTO 11400
11500 IF LEN(Q$)=2 THEN Q$= RIGHT$(Q$,1) ELSE GOTO 11900
11550 IF Q$=CHR$(72) THEN GOSUB 12900 '__CURSOR ARRIBA
11600 IF Q$=CHR$(75) THEN GOSUB 13200 '__CURSOR IZQUIERDA
11650 IF Q$=CHR$(77) THEN GOSUB 13500 '__CURSOR DERECHA
11700 IF Q$=CHR$(80) THEN GOSUB 13800 '__CURSOR ABAJO
11750  '_______ WIO SE CARGA CON UN VALOR EN LAS RUTINAS QUE
```

```
                ATIENDEN
11800 ' LAS TECLAS DE FUNCION <F1>,  <F2>,  <F3>  CON
11850 ' LOS VALORES 1,2,3        RESPECTIVAMENTE
11900  IF  WIO = 1 OR WIO = 3 THEN GOSUB  22650:IF  OK$<>"S"
       THEN GOTO 12000
11950  IF WIO = 2 THEN GOSUB 27300 ELSE IF WIO =3 THEN
       J=J-FLEN%:GOSUB 28200
12000 WIO=0:KEY(1) ON:KEY(2) ON:KEY(3) ON
12050 IF  LEN(Q$)=0 THEN GOTO 11250  '__ CURSOR  O <Fn>
12100 IF ASC(Q$)<>13 THEN GOTO 12200  '__ 13= ENTER
12150 IF I<20 THEN LOCATE I,J:
       PRINT MID$(FORD$(I),J,1);:J=1:I=I+1:GOTO 11250
12200 IF ASC(Q$)<32 OR ASC(Q$)>126 THEN BEEP:GOTO 11250
12250 MID$(FORD$(I),J)=Q$:WIO=0
12300 LOCATE I,J
12350 PRINT Q$;
12400 IF LEFT$(FORD$(I),3)="..." THEN
      FORD$(I)=SPACE$(80):RETURN
12450 J=J+1:IF J=81 THEN I=I+1:J=1
12500 DEF SEG=&HB000:WATRIB%=PEEK(-32767+2*(80*(I-1)+(J-1)))
12550 GOTO 11250
12600 '_____________________________________________________'
12650 '  RUTINAS DE CONTROL DEL  CURSOR:  LAS  SIGUIENTES '
12700 '  CONTROLAN EL MOVIMIENTO DEL CURSOR EN LAS CUATRO '
12750 '  DIRECCIONES POSIBLES Y DENTRO DE LAS LINEAS 1 A  '
12800 '-----------------------------------------------------'
12850 '_________________________ CURSOR ARRIBA
12900 Q$="":IF I=1 THEN BEEP:GOTO 13100
12950 POKE -32767+2*(80*(I-1)+(J-1)),WATRIB%:I=I-1
13000 WATRIB%=PEEK(-32767+2*(80*(I-1)+(J-1)))
13050 IF MID$(FORDP$(I),J,1)<> " " THEN GOTO 12900
13100 RETURN
13150 '_________________________ CURSOR IZQUIERDA
13200 Q$="":IF J=1 AND I=1 THEN BEEP:GOTO 13400
13250  POKE  -32767+2*(80*(I-1)+(J-1)),WATRIB%:J=J-1:
      IF  J=0 THEN J=80:I=I-1
13300 WATRIB%=PEEK(-32767+2*(80*(I-1)+(J-1)))
13350 IF MID$(FORDP$(I),J,1)<> " " THEN GOTO 13200
13400 RETURN
13450 '_________________________ CURSOR DERECHA
13500 Q$="":IF I=20 AND J=80 THEN BEEP:GOTO 13700
13550  POKE -32767+2*((I-1)*80+(J-1)),WATRIB%:J=J+1:
      IF  J=81 THEN J=1:I=I+1
13600 WATRIB%=PEEK(-32767+2*(80*(I-1)+(J-1)))
13650 IF MID$(FORDP$(I),J,1)<> " " THEN GOTO 13500
13700 RETURN
13750 '_________________________ CURSOR ABAJO
13800 Q$="":IF I=20 THEN BEEP:GOTO 14000
13850 POKE -32767+2*(80*(I-1)+(J-1)),WATRIB%:I=I+1
13900 WATRIB%=PEEK(-32767+2*(80*(I-1)+(J-1)))
13950 IF MID$(FORDP$(I),J,1)<> " " THEN GOTO 13800
14000 RETURN
14050 '_____________________________________________________'
14100 '  ELIMINACION DE FORMATOS:  ESTA RUTINA ACEPTA  EL  '
14150 '  NUMERO DEL FORMATO A ELIMINARSE,  VERIFICA QUE EL '
```

```
14200 '  FORMATO EXISTA,  Y ELIMINA DE LOS ARCHIVOS  TODA  '
14250 '  INFORMACION REFERENTE A DICHO FORMATO.  EL OPERA- '
14300 '  DOR DEBE ELIMINAR LAS TRANSACCIONES  QUE  HAGAN  '
14350 '  DE     ESTE    NUMERO     DE     FORMATO.         '
14400 '_____________________________________________________'
14450 COLOR 7,0,1:CLS
14500 LOCATE 2,22:PRINT "ELIMICACION DE FORMATOS";
14550 LOCATE 2,22:PRINT "ELIMICACION DE FORMATOS";
14600 LOCATE 3,22:PRINT "========================";
14650 LOCATE 6,22:PRINT "NUMERO DE FORMATO:";:II=6:
      JJ=42:WLEN=3:QQ$="":GOSUB 20950
14700 IF Q%=0 THEN 14650
14750 IF Q%=999 THEN RETURN
14800 FORNUM%=Q%
14850 Q%=2*Q%
14900 WIDX%=1+INT((Q%-2)/764)
14950 WOFF%=Q%
15000 IF WOFF%>764 THEN WOFF%=WOFF%-764:GOTO 15000
15050 WOFF%=WOFF%-1
15100 WNULL$=CHR$(0)+CHR$(0)
15150 IF MID$(FORREC2$(WIDX%),WOFF%,2)=WNULL$ THEN
       MESS=4:GOSUB 42150:RETURN
15200 WRRN%=CVI(MID$(FORREC2$(WIDX%),WOFF%,2))
15250 MID$(FORREC2$(WIDX%),WOFF%,2)=WNULL$
15300 I%=INT((WRRN%-1)/8)
15350 J%=8-WRRN%+8*I%
15400 MID$(FORREC1$,I%+1,1)=
      CHR$(ASC(MID$(FORREC1$,I%+1,1))-2^J%)
15450 WRRN%=1
15500 LSET FORDH$=STRING$(10,0)
15550 LSET FORDATA$=FORREC1$
15600 LSET FORDF$=MKI$(WRRN%)
15650 PUT #1,WRRN%
15700 FOR WRRN%=2 TO 4
15750 LSET FORDF$=MKI$(WRRN%)
15800 LSET FORDATA$=FORREC2$(WRRN%-1)
15850 PUT #1,WRRN%
15900 NEXT WRRN%
15950 RETURN
16000 '_____________________________________________________'
16050 ' LA SIGUIENTE RUTINA ABRE LOS ARCHIVOS REQUE-       '
16100 ' DOS POR ESTE PROGRAMA. EN CASO DE NO EXISTIR,      '
16150 ' CREA LOS REGISTROS DE CONTROL Y DE INDICE          '
16200 ' EN LOS MISMOS.                                     '
16250 '_____________________________________________________'
16300 '  PARAMFL. TIENE 3 TIPOS DE REGISTROS
16350 ' A: REC #1 : PARAMETROS DEL TERMINAL (POLL,VELOCIDAD)
16400 OPEN "R", #5,PARAMFL$,102
16450 FIELD #5,2 AS SPF$,100 AS SPD$
16500 SPRRN%=1
16550 GET #5,SPRRN%
16600  IF  CVI(SPF$) <> SPRRN% THEN GOSUB 33850
16650 '  FORMATOSFL. TIENE 3 TIPOS DE REGISTRO:
16700 ' A:   REC #1  CONTROL:  TIENE 1 BIT POR
      ' CADA REGISTRO DEL ARCHIVO
```

```
16750 '        SI 1 ==>> OCUPADO ; SI 0 ==>> LIBRE
16800 '   B:   REC  # 2,3,4  PUNTEROS :
              2 BYTES  PARA  CADA PANTALLA QUE APUNTA
16850 '        AL   REGISTRO   DONDE EMPIEZA EL 1ER REGISTRO
16900 ' C:  REC # N  : REGISTRO DE DATOS DE LA PANTALLA
16950 OPEN "R" ,#1,FORMATOSFL$,777
17000 FIELD #1,2 AS FORDF$,10 AS FORDH$,765 AS FORDATA$
17050 FORRRN%=1
17100 GET #1,FORRRN%
17150  IF  ASC(FORDF$)=0  AND  ASC(RIGHT$(FORDF$,1))=0  THEN
              GOSUB 19450 '__ ARCH NUEVOS
17200 FOR FORRRN%=1 TO 4
17250 GET #1,FORRRN%
17300 I=FORRRN%
17350 IF I=1 THEN FORREC1$=FORDATA$
17400 IF I>1 THEN FORREC2$(I-1)=FORDATA$
17450 NEXT FORRRN%
17500 ' DEFVAL.FL :   ARCHIVO DE RV'S(REGLAS DE VALIDACION)
17550 '               TIENE 3 TIPOS DE REGISTRO
17600 ' A: HEADERS:   1 BIT POR CADA REGISTRO DEL ARCHIVO;
17650 '               INDICA SI UNA RV EXISTE
17700 ' B: RV'S:      TIENE 10 RV'S
17750 OPEN "R",#2,DEFVALFL$,102
17800 FIELD #2,2 AS RVF$,100 AS RVD$
17850 RVRRN=1:K=1
17900 GET #2,RVRRN
17950 IF CVI(RVF$) <> RVRRN THEN GOSUB 20050
18000 RVRRN=2
18050 GET #2,RVRRN
18100 WHILE RVRRN =  CVI(RVF$)
18150 FOR J=1 TO 10
18200 RV$(K)=MID$(RVD$,1+(J-1)*10,10):K=K+1
18250 NEXT J
18300 RVRRN=RVRRN+1
18350 GET #2,RVRRN
18400 WEND
18450 RVMAX = K-1:IF RVMAX=0 THEN RVMAX=1:
      RV$(1)=STRING$(10,255)
18500 IF  RV$(RVMAX)=STRING$(10,0) THEN  RVMAX=RVMAX-1:
       GOTO 18500
18550 '------ GRUPOS.FL (LISTA DE TRANSACCIONES POR GRUPO)
18600  '------ EL  ARCHIVO TIENE 9 REGISTROS ,
       1  POR  CADA GRUPO DE TRANSACCIONES
18650  '------ CADA  REGISTRO TIENE UNA LISTA DE  HASTA
      100 TRANSAC.Y APUNTADORES
18700 OPEN "R",#3,GRUPOSFL$,510
18750 FIELD #3,2 AS TGIF$,254 AS TGI1$,254 AS TGI2$
18800 TGIRRN=1
18850 GET #3,TGIRRN
18900  IF  CVI(TGIF$) <> TGIRRN THEN GOSUB  20250
18950 '------ DEFTRAN.FL  (DEFINICION DE TRANSACCIONES)
19000 '------ 1ER REGISTRO DE CONTROL (1 BIT POR REGISTRO)
19050  '------ UN  REGISTRO POR TRANSACCION
       (APUNTADO  DESDE GRUPOS.FL)
19100 OPEN "R",#4,DEFTRANFL$,102
```

```
19150 FIELD #4,2 AS TDF$,100 AS TD$
19200 TDRRN=1
19250 GET #4,TDRRN
19300 IF CVI(TDF$) <> TDRRN THEN GOSUB 20650
19350 TDP$=TD$
19400 RETURN
19450 '_________________________ INICIALIZAR ARCHIVOS
19500 RSET FORDF$=MKI$(FORRRN%)
19550 LSET FORDH$=STRING$(10,0)
19600 LSET SPD$=SPD$
19650 LSET FORDATA$=STRING$(1,240)+STRING$(764,0)
19700 PUT #1,FORRRN%
19750 LSET FORDATA$=STRING$(765,0)
19800 FOR FORRRN%=2 TO 4
19850 RSET FORDF$=MKI$(FORRRN%)
19900 PUT #1,FORRRN%
19950 NEXT FORRRN%
20000 RETURN
20050 RSET RVF$=MKI$(RVRRN)
20100 LSET RVD$=CHR$(128)+STRING$(99,0)
20150 PUT #2,RVRRN
20200 RETURN
20250 '_____________________________ INICIALIZAR GRUPOSFL
20300 FOR TGIRRN =1 TO 9
20350 RSET TGIF$=MKI$(TGIRRN)
20400 LSET TGI1$=STRING$(254,255)
20450 LSET TGI2$=STRING$(254,255)
20500 PUT #3,TGIRRN
20550 NEXT TGIRRN
20600 RETURN
20650 '_____________________________ INICIALIZAR DEFTRANFL
20700 TDRRN=1
20750 RSET TDF$=MKI$(TDRRN)
20800 LSET TD$=CHR$(128)+STRING$(99,0)
20850 PUT #4,TDRRN
20900 RETURN
20950 '_________________________ ACEPTAR CAMPOS NUMERICOS
21000 QQ$="":Q%=0
21050   LOCATE  II,JJ:DEF   SEG=&HB000:
       POKE   -32767+2*((II-1)*80+(JJ-1)),15
21100 Q$=INKEY$
21150 IF LEN(Q$) = 2 THEN Q$=RIGHT$(Q$,1) ELSE GOTO 21300
21200 IF Q$=CHR$(75) THEN IF LEN(QQ$)>0 THEN WLEN=WLEN+1:
      JJ=JJ-1:QQ$=LEFT$(QQ$,LEN(QQ$)-1):GOTO 21050
21250 BEEP:GOTO 21050
21300 IF LEN(Q$)=0 THEN GOTO 21050
21350 IF ASC(Q$)=13 THEN RETURN
21400 IF ASC(Q$)<48 OR ASC(Q$)>57 THEN BEEP:GOTO 21050
21450 LOCATE II,JJ:PRINT Q$;
21500 QQ$=QQ$+Q$:Q%=VAL(QQ$):JJ=JJ+1:WLEN=WLEN-1:
21550 IF WLEN=0 THEN RETURN ELSE GOTO 21050
21600 '_________________________ ACEPTAR CAMPOS ALFANUMERICOS
21650 QQ$=""
21700 LOCATE  II,JJ:DEF   SEG=&HB000:
      POKE   -32767+2*((II-1)*80+(JJ-1)),15
```

```
21750 Q$=INKEY$
21800 IF LEN(Q$) = 2 THEN Q$=RIGHT$(Q$,1) ELSE GOTO 21950
21850 IF Q$=CHR$(75) THEN IF LEN(QQ$)>0 THEN WLEN=WLEN+1:
      JJ=JJ-1:QQ$=LEFT$(QQ$,LEN(QQ$)-1):GOTO 21700
21900 BEEP:GOTO 21700
21950 IF LEN(Q$)=0 THEN GOTO 21700
22000 IF ASC(Q$)=13 THEN RETURN
22050 IF ASC(Q$) < 32 OR ASC(Q$)>128 THEN BEEP :GOTO 21700
22100 LOCATE II,JJ:PRINT Q$;
22150  QQ$=QQ$+Q$:JJ=JJ+1:WLEN=WLEN-1:IF WLEN=0 THEN
       RETURN ELSE GOTO 21700
22200 '_____________________________ <F1> CAMPO DE ENTRADA
22250 WIO=1:Q$="1":KEY(1) OFF:KEY(2) OFF:KEY(3) OFF
22300 RETURN 11900
22350 '_____________________________ <F2> CAMPO DE SALIDA
22400 WIO=2:Q$="2":KEY(1) OFF:KEY(2) OFF:KEY(3) OFF
22450 RETURN 11900
22500  '_____________________________ <F3> CAMPO DE ENTRADA Y
                                        SALIDA
22550 WIO=3:Q$="3":KEY(1) OFF:KEY(2) OFF:KEY(3) OFF
22600 RETURN 11900
22650  '_____________________________ DEFINICION DE CAMPOS DE
                                       ENTRADA
22700 FILL$=" "
22750 LOCATE 21,1:
      PRINT "CAMPO DE ENTRADA DE DATOS NUMERO:"; RVIDX
22800 LOCATE  22,1:PRINT  "TIPO?:"  TAB(12)
        "LONGITUD?:" TAB(25) "V.INIC.?:"
22850  LOCATE  23,1:PRINT "FC?:" TAB(25)  "FECHA?:"
       TAB(35) "CDV?:" TAB(45) "OK?:"
22900 WRV$(RVIDX)="                          "
22950 LOCATE 25,1
23000  PRINT  "N=NUM,  A=ALFA,  B=A/N,  F=FECHA,
              Q=CLV.ALFA, C=CUENTA,";
23050 PRINT " X=CANCELAR";
23100 A$="NABFQCXnabfqcx"
23150 IL=22:JL=1:NLEN=9:II=22:JJ=8:WLEN=1:GOSUB 42500:
        GOSUB 21600
23200 GOSUB 42650'      ACCEPT TIPO
23250 IF INSTR(A$,Q$) = ZERO THEN GOTO 23150
23300 IF INSTR(A$,Q$) > 8 THEN Q$=MID$(A$,INSTR(A$,Q$)-8,1)
23350 FTIP$=Q$
23400  IF  FTIP$="X" THEN  WIO=0:Q$="":GOTO  27050
23450 LOCATE 25,1
23500 PRINT "01-16 SI NUMERICO , 01-50 SI ALFABETICO"
      +SPACE$(41);
23550 IF FTIP$="F" THEN FLEN%=6:LOCATE 22,22 :PRINT "6";:
      GOTO 23900
23600 IL=22:JL=12:NLEN=11:JJ=22:WLEN=2:GOSUB 42500:
      GOSUB 20950
23650 GOSUB 42650                        '__ ACEPTAR LONGITUD
23700 IF Q% = 0 OR Q%>50 THEN BEEP:GOTO 23600
23750 IF Q% > 81-J THEN BEEP:GOTO 23600
23800 IF (FTIP$="N" OR FTIP$="P" OR FTIP$="C") AND Q%  > 16
            THEN BEEP:GOTO 23600
```

```
23850 FLEN%=Q%
23900 IF   FTIP$="F"  OR  FTIP$="Q"  OR   FTIP$="P"   THEN
            QQ$=SPACE$(FLEN%):GOTO 24000
23950 IL=22:JL=25:NLEN=10+FLEN%:JJ=35:WLEN=FLEN%:
      GOSUB  42500:GOSUB  21600:GOSUB  42650
24000 DEFULT$(RVIDX)=QQ$:IF LEN(QQ$)<FLEN% THEN
      DEFULT$(RVIDX)=DEFULT$(RVIDX)+SPACE$(FLEN%-LEN(QQ$))
24050 LOCATE I,J:PRINT DEFULT$(RVIDX);
24100 JJ=FLEN%
24150 MID$(FORD$(I),J,JJ)=QQ$
24200 MID$(FORDP$(I),J,JJ)=STRING$(JJ,49)
24250 DEF SEG=&HB000
24300 JJ=J+JJ-1
24350 FOR J=J TO JJ
24400 POKE -32767+2*((I-1)*80+(J-1)),112
24450 NEXT J
24500 LOCATE 25,1
24550  PRINT "CODIGO PARA CENTRAL:       00=NO  FC  ASOCIADO;
           HH=FC"+SPACE$(31);
24600 IL=23:JL=1:NLEN=8:II=23:JJ=6:WLEN=2:GOSUB 42500:
       GOSUB 21600
24650  GOSUB  42650            '__  ACEPTAR  CODIGO  DE
                                    CONTROL PARA CENTRAL
24700 IF QQ$="00" THEN 24850
24750 IF QQ$ < "20"  OR QQ$  > "7E" THEN BEEP :GOTO 24500
24800 IF LEFT$(QQ$,1) > "F" OR RIGHT$(QQ$,1) > "F" THEN
         BEEP :GOTO 24500
24850 Q1%=INT(ASC(LEFT$(QQ$,1)))-48:IF Q1%>10 THEN Q1%=Q1%-7
24900 Q2%=INT(ASC(RIGHT$(QQ$,1)))-48:IF Q2%>10 THEN
      Q2%=Q2%-7
24950 FC$=CHR$((16*Q1%+Q2%) AND 127)
25000 IF FTIP$ <> "F" THEN 25350
25050 LOCATE 25,1
25100 PRINT "FORMATO DE FECHA: 1=AAMMDD, 2=DDMMAA,
              3=MMDDAA"+SPACE$(34);
25150 IL=23:JL=25:NLEN=8:II=23:JJ=32:WLEN=1:
      GOSUB 42500:GOSUB 21000
25200 GOSUB 42650       '-- FORMATO DE FECHA
25250 IF Q% =0 OR Q%>3 THEN BEEP:GOTO 25050
25300 FF%=Q%     'FORMATO DE FECHA
25350 IF FTIP$ <> "C" THEN GOTO 25650
25400 LOCATE 25,1:PRINT "DIGITO VERIFICADOR:
               1,2,3"+SPACE$(54);
25450 IL=23:JL=35:NLEN=8:II=23:JJ=40:WLEN=1:
      GOSUB 42500:GOSUB 21000
25500 GOSUB 42650       '___ CODIGO DE DIGITO VERIFICADOR
25550 IF Q%=0 OR Q%>3 THEN BEEP:GOTO 25400
25600 FCDV%=Q%
25650 Q$=""
25700 LOCATE 25,1
25750 PRINT "OK  S/N"+SPACE$(73);
25800 IL=23:JL=45:NLEN=5:II=23:JJ=50:WLEN=1:
      GOSUB 42500:GOSUB 21600
25850 GOSUB 42650       '___ OK?
25900 IF Q$="s" THEN Q$="S"
```

```
25950 IF Q$<> "N" AND Q$<>"S" THEN BEEP:GOTO 25700
26000 OK$=Q$:Q$=""
26050 IF OK$="S" THEN GOTO 26400
26100 DEF SEG=&HB000
26150 FOR J=J-1 TO J-FLEN% STEP -1
26200 POKE -32767+2*(80*(I-1)+(J-1)),ATR%
26250 NEXT J
26300 Q$="":J=J+1:MID$(FORDP$(I),J,FLEN%)=SPACE$(FLEN%)
26350 LOCATE I,J:PRINT SPACE$(FLEN%):GOTO 27000
26400 RV1%=0:RV2%=0:RV4%=0:FILL$=" "
26450 IF FTIP$="B" THEN RV1%=RV1%+192:GOTO 26800
26500 IF FTIP$="A" THEN RV1%=RV1%+128:GOTO 26800
26550 RV1%=RV1%+64
26600 IF FTIP$="C" THEN RV1%=RV1%+8:RV4%=FCDV%:GOTO 26800
26650 IF FTIP$="P" THEN RV1%=RV1%+16:GOTO 26800
26700 IF FTIP$="Q" THEN RV1%=RV1%+24
26750 IF FTIP$="F" THEN RV1%=RV1%+32+8*(FF%-1).
26800 IF DEFULT$(RVIDX)<>SPACE$(FLEN%) THEN RV1%=RV1%+1
26850 WRV$(RVIDX)=FC$+CHR$(FLEN%)+CHR$(I)+
      CHR$(J-FLEN%)+CHR$(0)+CHR$(RV1%)
26900 WRV$(RVIDX)=WRV$(RVIDX)+CHR$(RV2%)+FILL$+CHR$(RV4%)
                  +STRING$(6,0)
26950 RVIDX=RVIDX+1
27000 LOCATE 25,1
27050 PRINT "F1=ENTRADA   F2=SALIDA
             F3=ENTRADA/SALIDA"+SPACE$(38);
27100 FOR II=21 TO 25
27150 LOCATE II,1:PRINT SPACE$(79);
27200 NEXT II
27250 RETURN
27300  '_________________________   DEFINICION  DE  CAMPOS  DE
                                   SALIDA
27350 LOCATE 21,1:PRINT "CAMPO DE SALIDA NUMERO:"; OVIDX
27400 LOCATE 22,1:PRINT "LONGITUD?:              OK?";
27450 LOCATE 25,1
27500 PRINT "01-80        99=ANULAR"+SPACE$(59);
27550 IL=22:JL=1:NLEN=13:II=22:JJ=12:WLEN=2:
            GOSUB 42500:GOSUB 20950:GOSUB 42650
27600 IF Q%=99 THEN MESS=8:GOSUB 42150:
             GOTO   28650 '___ANULAR
27650 IF Q%=0 OR Q%>80 THEN BEEP:GOTO 27450
27700 IF Q%>81-J THEN BEEP:GOTO 27450
27750 FLEN%=Q%
27800 LOCATE 25,1
27850 PRINT "OK  S/N"+SPACE$(73);
27900 IL=22:JL=23:NLEN=5:II=22:JJ=28:WLEN=1:
      GOSUB 42500:GOSUB 21600
27950 GOSUB 42650        '___ OK?
28000 IF Q$="s" THEN Q$="S"
28050 IF Q$<> "N" AND Q$<>"S" THEN BEEP:GOTO 27800
28100 OK$=Q$:Q$=""
28150 IF OK$<> "S" THEN GOTO 28700
28200 JJ=J+FLEN%-1
28250 MID$(FORDP$(I),J,FLEN%)=STRING$(FLEN%,49)
28300 DEF SEG=&HB000
```

```
28350 FOR J=J TO JJ
28400 IF WIO=2 THEN POKE -32767+2*((I-1)*80+(J-1)),180:
         GOTO 28500
28450 POKE -32767+2*((I-1)*80+(J-1)),195
28500 NEXT J
28550 WOV$(OVIDX)=CHR$(FLEN%)+CHR$(I)+CHR$(J-FLEN%)
28600 OVIDX=OVIDX+1
28650 Q$=""
28700 FOR II=21 TO 25
28750 LOCATE II,1:PRINT SPACE$(79);
28800 NEXT II
28850 LOCATE 25,1
28900 PRINT "F1=ENTRADA   F2=SALIDA
             F3=ENTRADA/SALIDA"+SPACE$(23);
28950 RETURN
29000 '_________________________  COMPACTAR DATOS
29050 CLS:PRINT "ESPERE  ";
29100 FOR I=1 TO RVIDX-1   ' CLEAR DEFAULTS
29150 A1=ASC(MID$(WRV$(I),2,1))
29200 A2=ASC(MID$(WRV$(I),3,1))
29250 A3=ASC(MID$(WRV$(I),4,1))
29300 MID$(FORD$(A2),A3,A1)=SPACE$(ASC(WRV$(I)))
29350 NEXT I
29400 FORDLEN=0
29450 FOR I=1 TO 20
29500 PD$(I)=""
29550 IF FORD$(I)=SPACE$(80) THEN GOTO 30000
29600 J=1:FORD$(I)=FORD$(I)+"   "
29650 K=INSTR(J,FORD$(I),"   ")
29700 IF K=J THEN J=J+1:GOTO 29650
29750 IF K=0 THEN GOTO 30000
29800 Q$=MID$(FORD$(I),J,K-J)
29850 IF LEFT$(Q$,1)=" " THEN Q$=MID$(Q$,2):J=J+1:GOTO 29850
29900 IF Q$ <> SPACE$(LEN(Q$)) THEN
         PD$(I)=PD$(I)+CHR$(I)+CHR$(J)+CHR$(K-J)+Q$
29950 J=K:GOTO 29650
30000 FORDLEN=FORDLEN+LEN(PD$(I))
30050 NEXT I
30100 '_____________________  ENCONTRAR REGISTRO DISPONIBLE
30150 I=0
30200 I=I+1
30250 J=0
30300 J=J+1
30350 IF MID$(FORREC1$,J,1)<>CHR$(255) THEN GOTO 30450
30400 IF J=255 THEN GOTO 30200 ELSE GOTO  30300
30450 WRRN=(I-1)*2040+(J-1)*8:WRRN1=0
30500 WRRN1=8-INT(LOG((255 XOR
               ASC(MID$(FORREC1$,J,1))))/LOG(2))
30550 WRRN=WRRN+WRRN1
30600 '_________________________  GRABAR EN DISCO
30650 Q$=MID$(FORREC1$,J,1):
      Q%=ASC(Q$)+INT(256/(2^(WRRN1))):Q$=CHR$(Q%)
30700 MID$(FORREC1$,J,1)=Q$
30750 MID$(FORREC2$(WIDX%),WOFF%,2)=MKI$(WRRN)
30800 SH$=MKI$(FORRNUM%)+MKI$(NXTRRN)+MKI$(FORDLEN)+
```

```
          MKI$(FORDLEN+RVLEN)
30850 SH$=SH$+MKI$(FORDLEN+RVLEN+OVLEN):SX$=""
30900 J=1
30950 FOR I=1 TO 18+RVIDX+OVIDX
31000 SX$=SX$+PD$(I)
31050 NEXT I
31100 SX$=SX$+CHR$(255)
31150 IF LEN(SX$) < 760 THEN GOTO 31300
31200  LOCATE  20,1:PRINT "FORMATO MUY  LARGO.
                    ELIMINE  SU TRANSACCION";
31250 RETURN
31300 GOSUB 31400     '___ GRABAR REGISTRO
31350 RETURN
31400 '_____________________________ GRABAR REGISTRO
31450 LSET FORDF$=MKI$(WRRN)
31500 LSET FORDH$=SH$
31550 LSET FORDATA$=SX$
31600 PUT #1,WRRN
31650 WRRN =1
31700 LSET FORDF$=MKI$(WRRN)
31750 LSET FORDATA$=FORREC1$
31800 PUT #1,WRRN
31850 FOR I=2 TO 4
31900 WRRN=I
31950 LSET FORDF$=MKI$(WRRN)
32000 LSET FORDATA$=FORREC2$(I-1)
32050 PUT #1,WRRN
32100 NEXT I
32150 RETURN
32200 '_____________________ PROCESAR CAMPOS DE ENTRADA
32250 K=RVMAX
32300 RVLEN=0
32350 FOR J=1 TO RVIDX-1
32400 FOR I=1 TO RVMAX
32450 IF MID$(WRV$(J),6,10)=RV$(I) THEN
          MID$(WRV$(J),5,1)=CHR$(I):I=RVMAX
32500 NEXT I
32550 IF MID$(WRV$(J),5,1)<>CHR$(0) THEN GOTO 32650
32600 RVMAX=RVMAX+1:RV$(RVMAX)=MID$(WRV$(J),6,10):
      MID$(WRV$(J),5,1)=CHR$(RVMAX)
32650 IF DEFULT$(J)=SPACE$(LEN(DEFULT$(J))) THEN
         DEFULT$(J)=""
32700 PD$(20+J)=LEFT$(WRV$(J),5)+DEFULT$(J):RVLEN=RVLEN
                +LEN(PD$(20+J))
32750 NEXT J
32800  IF K <> RVMAX THEN GOSUB 32900
32850 RETURN
32900 '_____________________ ACTUALIZAR DEFVAL.FL
32950 RVRRN = 2
33000 J=1
33050 Q$=""
33100 FOR I=1 TO 10
33150 IF LEN(RV$((J-1)*10+I))=0 THEN RV$((J-1)*10+I)=
         STRING$(10,0)
33200 Q$=Q$+RV$((J-1)*10+I)
```

```
33250 NEXT I
33300 LSET RVF$=MKI$(RVRRN)
33350 LSET RVD$=Q$
33400 PUT #2,RVRRN:RVRRN=RVRRN+1
33450 IF RVMAX > (J-1)*10  THEN J=J+1:GOTO 33050
33500 RETURN
33550 '_____________________ PROCESAR CAMPOS DE SALIDA
33600 OVLEN=0
33650 FOR I=1 TO OVIDX-1
33700 PD$(20+RVIDX-1+I)=WOV$(I):OVLEN=OVLEN+3
33750 NEXT I
33800 RETURN
33850 '_____________________ INIT PARAMFL.
33900 Q$="40 9600"+STRING$(93,0)
33950 LSET SPD$=Q$
34000 LSET SPF$=MKI$(1)
34050 PUT #5,SPRRN%
34100 RETURN
34150 '_____________________________________________________'
34200 '    IMPRESION DE TRANSACCIONES: ESTA RUTINA GENERA   '
34250 '    UNA LISTA DE TRANSACCIONES DEL GRUPO SELECIONA   '
34300 '    POR EL OPERADOR                                  '
34350 '_____________________________________________________'
34400 CLOSE #1,#2,#5:MESS=7:CA=1:GOSUB 42150
34450 IF Q$="X" THEN MESS=8:GOSUB 42150:GOSUB 41300:RESET:
        GOSUB 16000:RETURN
34500 WIDTH "LPT1:",80
34550 COLOR 0,2,1:CLS:GOSUB 41050
34600 LOCATE 2,22:PRINT "IMPRESION DE TRANSACCIONES";
34650 LOCATE 3,22:PRINT "==========================";
34700 LOCATE 4,22:PRINT "GRUPO:"
34750 II=4:JJ=34:WLEN=1:GOSUB 20950    '___ ACEPTAR GRUPO
34800 GR$=STR$(Q%)
34850 IF Q%=0 THEN 35450
34900 GET #3,Q%
34950 GOSUB 35550
35000 FOR I=1 TO 253 STEP 2
35050 IF ASC(MID$(TGI2$,I,1)) = 255 THEN 35350
35100 TT%=CVI(MID$(TGI2$,I,2))
35150 GET #4,TT%:Q$=TD$
35200 IF I=50 OR I=100 OR I=150 OR I=200 THEN GOSUB 35550
35250 LPRINT CVI(MID$(Q$,2,2));TAB(8) MID$(Q$,6,30);
35300 LPRINT MID$(Q$,36,6); TAB(50) CVI(MID$(Q$,42,2))
35350 NEXT I
35400 LPRINT CHR$(12)
35450 RESET:GOSUB 16000
35500 MESS=11:GOSUB 42150:RETURN
35550 '_____________________ TITULOS
35600 LPRINT CHR$(15)+CHR$(12)+CHR$(14);
35650 LPRINT "TRANSACCIONES DEL GRUPO" + GR$
35700 LPRINT STRING$(80,95)+CHR$(10)+CHR$(10)
35750 LPRINT "NUM    NOMBRE                                    PROG.
              FORMATO"
35800 RETURN
35850 '_____________________________________________________'
```

```
35900 '      IMPRESION DE FORMATOS: ESTA RUTINE GENERA     '
35950 '      UN LISTADO DE FORMATOS DEFINIDOS. EL OPERA-   '
36000 '      DOR INGRESA EL RANGO DE FORMATOS REQUERIDO    '
36050 '____________________________________________________'
36100 CLOSE #2,#3,#5:MESS=7:CA=1:GOSUB 42150
36150 IF Q$="X" THEN MESS=8:GOSUB 42150:
         GOSUB 41300:RESET:GOSUB 16000:RETURN
36200 WIDTH "LPT1:",82
36250 COLOR 0,2,1:CLS:GOSUB 41050
36300 LOCATE 2,22:PRINT "IMPRESION DE FORMATOS"
36350 LOCATE 3,22:PRINT "======================"
36400 LOCATE 4,22:PRINT "FORMATO INICIAL:"
36450 LOCATE 5,22:PRINT "FORMATO FINAL:"
36500 II=4:JJ=46:WLEN=3:GOSUB 20950    '___ INICIAL
36550 IF Q%=999 OR Q%=0 THEN 37300
36600 PI%=(Q%)
36650 II=5:JJ=46:WLEN=3:GOSUB 20950    '___ FINAL
36700 IF Q%=999 THEN GOSUB 41300:GOTO 37300
36750 PF%=(Q%)
36800 IF PF%<PI% THEN 36250
36850 FOR PI% = PI% TO PF%
36900 Q%=PI%
36950 FORRNUM%=Q%
37000 Q%=Q%*2
37050 WIDX%=1+INT((Q%-2)/764)
37100 WOFF%=Q%
37150 IF WOFF%>764 THEN WOFF%=WOFF%-764:GOTO 37150
37200 WOFF%=WOFF%-1
37250  IF MID$(FORREC2$(WIDX%),WOFF%,2)<>  (CHR$(0)+CHR$(0))
          THEN GOSUB 37450
37300 NEXT PI%
37350 RESET:GOSUB 16000
37400 MESS=11:GOSUB 42150:RETURN
37450 '_________________________  IMPRIMIR FORMATO PI%
37500 LPRINT CHR$(15)+CHR$(12)+CHR$(14);
37550 LPRINT "FORMATO: "+STR$(PI%)
37600 LPRINT STRING$(80,95)+CHR$(10)
37650 FORI%=CVI(MID$(FORREC2$(WIDX%),WOFF%,2))
37700 GET #1,FORI%
37750 FOR I=1 TO 20:FORD$(I)=SPACE$(80):NEXT I
37800 FORDLEN%=CVI(MID$(FORDH$,5,2))
37850 RVLEN%=CVI(MID$(FORDH$,7,2))
37900 I=1
37950 WHILE I< FORDLEN%
38000 J=I:J1=I+1:J2=I+2:J3=I+3
38050 MID$(FORD$(ASC(MID$(FORDATA$,J,1))),
      ASC(MID$(FORDATA$,J1,1)))=MID$(FORDATA$,J3,
      ASC(MID$(FORDATA$,J2,1)))
38100 I=I+3+ASC(MID$(FORDATA$,J2,1))
38150 WEND
38200 RVLEN%=CVI(MID$(FORDH$,7,2))
38250 WHILE I<RVLEN%
38300 J=I:J1=I+1:J2=I+2:J3=I+3:J4=I+4:J5=I+5
38350 MID$(FORD$(ASC(MID$(FORDATA$,J2,1))),
      ASC(MID$(FORDATA$,J3,1) ))=
```

```
      STRING$(ASC(MID$(FORDATA$,J1,1)),105)
38400 IF (1 AND ASC(LEFT$(RV$(ASC(MID$(FORDATA$,J4,1))),1)))
         <> 0 THEN I=I+ASC(MID$(FORDATA$,J1,1))
38450 I=I+5
38500 WEND
38550 OVLEN%=CVI(MID$(FORDH$,9,2))
38600 WHILE I<OVLEN%
38650 J=I:J1=I+1:J2=I+2
38700 IF MID$(FORD$(ASC(MID$(FORDATA$,J1,1))),
         ASC(MID$(FORDATA$,J2,1)),
         ASC(MID$(FORDATA$,J,1)))=
         STRING$(ASC(MID$(FORDATA$, J,1)),105) THEN
         MID$(FORD$(ASC(MID$(FORDATA$,J1,1))),
             ASC(MID$(FORDATA$,J2,1) ))=
         STRING$(ASC(MID$(FORDATA$,J,1)),117):GOTO 38800
38750 MID$(FORD$(ASC(MID$(FORDATA$,J1,1))),
           ASC(MID$(FORDATA$,J2,1) ))=
      STRING$(ASC(MID$(FORDATA$,J,1)),111)
38800 I=I+3
38850 WEND
38900 LPRINT STRING$(82,249);
38950 FOR I=1 TO 20:LPRINT CHR$(249)+FORD$(I)+CHR$(249);:
      NEXT I
39000 LPRINT STRING$(82,249);
39050 LPRINT CHR$(10)
39100 LPRINT "CAMPOS DE ENTRADA"
39150 LPRINT "FC  TIPO  LIN  POS  LONG  CDV  NOTA
               V.INICIAL"
39200 I=FORDLEN%+1
39250 RVLEN%=CVI(MID$(FORDH$,7,2))
39300 WHILE I<RVLEN%
39350 J=I:J1=I+1:J2=I+2:J3=I+3:J4=I+4:J5=I+5
39400 Q$=SPACE$(80)
39450 MID$(Q$,1,2)=HEX$(ASC(MID$(FORDATA$,J,1)))
39500 MID$(Q$,11)=STR$(ASC(MID$(FORDATA$,J2,1)))
39550 MID$(Q$,15)=STR$(ASC(MID$(FORDATA$,J3,1)))
39600 MID$(Q$,20)=STR$(ASC(MID$(FORDATA$,J1,1)))
39650 IF (1 AND ASC(LEFT$(RV$(ASC(MID$(FORDATA$,J4,1))),1)))
         <> 0 THEN I=I+ASC(MID$(FORDATA$,J1,1)):
         MID$(Q$,38)=
              MID$(FORDATA$,J5,AS C(MID$(FORDATA$,J1,1)))
39700 RV1= ASC(LEFT$(RV$(ASC(MID$(FORDATA$,J4,1))),1))
39750 RV4= ASC(MID$(RV$(ASC(MID$(FORDATA$,J4,1))),4,1))
39800 IF RV1 > 191 THEN MID$(Q$,6)="A/N":GOTO 40250
39850 IF RV1 > 127 THEN MID$(Q$,6)="ALFA":GOTO 40250
39900 IF RV1 > 111 THEN MID$(Q$,6)="FECH":
         MID$(Q$,31)="MMDDAA":GOTO 40250
39950 IF RV1 > 103 THEN MID$(Q$,6)="FECH":
         MID$(Q$,31)="DDMMAA":GOTO 40250
40000 IF RV1 > 95 THEN MID$(Q$,6)="FECH":
         MID$(Q$,31)="AAMMDD":GOTO 40250
40050 IF RV1 > 87 THEN  MID$(Q$,6)="PASS":
         MID$(Q$,31)="ALFA-NUM":GOTO 40250
40100 IF RV1 > 79 THEN MID$(Q$,6)="PASS":
         MID$(Q$,31)="NUM":GOTO 40250
```

```
40150 IF RV1 > 71 THEN MID$(Q$,6)="CTA":
         MID$(Q$,27)=STR$(RV4):GOTO 40250
40200 IF RV1 > 63 THEN MID$(Q$,6)="NUM":GOTO 40250
40250 I=I+5
40300 LPRINT Q$
40350 WEND
40400 LPRINT CHR$(10)
40450 LPRINT "CAMPOS DE SALIDA"
40500 LPRINT "LIN  POS  LONG"
40550 WHILE I<OVLEN%
40600 J=I:J1=I+1:J2=I+2
40650 Q$=SPACE$(80)
40700 MID$(Q$,1)=STR$(ASC(MID$(FORDATA$,J1,1)))
40750 MID$(Q$,6)=STR$(ASC(MID$(FORDATA$,J2,1)))
40800 MID$(Q$,10)=STR$(ASC(MID$(FORDATA$,J)))
40850 I=I+3
40900 LPRINT Q$
40950 WEND
41000 RETURN
41050 '_________RUTINA DE LINEA DE STATUS
41100 LOCATE 24,46:PRINT "MENSAJE--"+CHR$(16);
41150 DEF SEG=&HB000
41200 FOR PK=0 TO 79:POKE -32767+2*(80*(23)+PK),112:NEXT PK
41250 RETURN
41300 '_________TIMER USO GENERAL
41350 FOR PK=1 TO 100:NEXT PK
41400 RETURN
41450 '________MENSAJES
41500 ON MESS GOTO
      41550,41600,41650,41700,41750,41800,41850,
      41900,41950,42000,42050
41550 MESS$(1)="DIGITACION ERROR    ":RETURN
41600 MESS$(2)="TRANSACCION NO EXISTE":RETURN
41650 MESS$(3)="PANTALLA YA EXISTE  ":RETURN
41700 MESS$(4)="PANTALLA NO EXISTE  ":RETURN
41750 MESS$(5)="OPERADOR YA EXISTE  ":RETURN
41800 MESS$(6)="OPERADOR NO EXISTE  ":RETURN
41850 MESS$(7)="IMPRESORA           ":RETURN
41900 MESS$(8)="CANCELADO           ":RETURN
41950 MESS$(9)="TRANSACCION YA EXISTE":RETURN
42000 MESS$(10)="                     ":RETURN
42050 MESS$(11)="LISTO...?            ":RETURN
42100 RETURN
42150 '________ MENSAJE
42200 LOCATE 24,58:GOSUB 41450:PRINT MESS$(MESS);:
       DEF SEG=&HB000:MA$=""
42250 FOR PK=29 TO 41:X%=SCREEN(24,PK):MA$=MA$+CHR$(X%):
      NEXT PK
42300 FOR PK=58 TO 78:
      POKE   -32767+2*(80*(23)+(PK-1)),240:NEXT PK
42350 Q$=INKEY$:IF LEN(Q$)=0 THEN  42350
42400 IF Q$<>CHR$(13) THEN 42350
42450 RETURN
42500 '_______CAMBIO DE ATRIBUTOS PARA SELECCION
42550 FOR PK=JL TO JL+NLEN:
```

```
          POKE  -32767+2*(80*(IL-1)+(PK-1)),112:NEXT PK
42600 RETURN
42650 '_______CAMBIO DE ATRIBUTOS DESPUES DE SELECCION
42700 FOR PK=JL TO JL+NLEN:
          POKE  -32767+2*(80*(IL-1)+(PK-1)),38:NEXT PK
42750 RETURN
```

4.2 PROGRAMA DE CONTROL DEL TERMINAL

A continuación se incluye el listado del programa que realiza las funciones de terminal inteligente.

```
10 '_____________________________________________________________'
20 '    PROGRAMA DE TERMINAL.                                    '
30 '    ESTE PROGRAMA PERMITE A UN COMPUTADOR PERSONAL           '
32 '    COMPATIBLE CONECTARSE A UNA RED DE PROCESAMIENTO         '
34 '    TRANSACCIONAL UTILIZANDO PROTOCOLO DE COMUNICACIO-       '
36 '    NES NCR ISO ASINCRONICO.                                 '
38 '    ESTE PROGRAMA FORMA PARTE DE LA TESIS DE GRADO           '
40 '    DE ALBERTO ANDRADE VAREA.                                '
42 '_____________________________________________________________'
50 CLEAR 20000
100 DEF SEG=&H0        '__ SEGMENTO DE DIRECCION DE
                           INTERRUPCIONES
150  DRIVERSEG=PEEK(&H33)*256+PEEK(&H32)'__  SEGMENTO DE  LA
                                        RUTINA DE INTERRUPCION
200 COLPAN1=7:COLPAN2=0
250 COLSTL1=8:COLSTL2=7
300 COLINP1=0:COLINP2=7
350 COLOUT1=0:COLOUT2=15
400 DEFVALFL$="A:DEFVAL.FL"
450 PARAMFL$="A:PARAM.FL"
500 FORMATOSFL$="A:FORMATOS.FL"
550 GRUPOSFL$="A:GRUPOS.FL"
600 DEFTRANFL$="A:DEFTRAN.FL"
650 GOSUB 950            '__  INICIALIZACION
700 GOSUB 3850           '__  LINEA DE STATUS
750 WHILE ONOFF$<>"FIN"
800 GOSUB 5300           '__  LAZO PRINCIPAL
850 WEND
900 CLS:SYSTEM
950 '_____________________________________________________________'
1000 '   RUTINA DE INICIALIZACION:                                '
1050 '   ESTA RUTINA ABRE LOS ARCHIVOS REQUERIDOS, LEE            '
1100 '   LOS REGISTROS DEL ARCHIVO DEFVAL.FL, Y PONE EN           '
1150 '   MEMORIA EL CONTENIDO.                                    '
1200 '_____________________________________________________________'
1250 ON ERROR GOTO 30850
1300 CLS
1350 DIM A%(20),B%(20),NDM%(12)
1400 NDM%(1)=31:NDM%(2)=28:NDM%(3)=31:NDM%(4)=30:
     NDM%(5)=31:NDM%(6)=30:NDM%(7)=31:NDM%(8)=31:NDM%(9)=30:
     NDM%(10)=31:NDM%(11)= 30:NDM%(12)=31
1450 FOR I=1 TO 10:KEY I,"":NEXT I
1500 ER$="    "              '___CODIGO DE ERROR
1550 GR$=" "                 '___GRUPO
1600 RT$=""                  '__ TRANSACCIONES RESIDENTES EN
```

```
                                        MEMORIA
1650 RS$=""                        '__ FORMATOS RESIDENTES EN
                                        MEMORIA
1700 POLL$="XX":SPEED$="99999"
1750 DIM RV$(200),T1$(10),T2$(10),DEFTRAN$(50),SCDATA$(40),
     SH$(20),IN$(40)
1800 RV%=0:DEFTRAN=0:SC=0
1850 OPEN "R",#1,FORMATOSFL$,777
1900 FIELD #1,2 AS SDF$,10 AS SDH$,765 AS SD$
1950 OPEN "R",#2,DEFVALFL$,102
2000 FIELD #2,2 AS RVF$,100 AS RVD$
2050 OPEN "R",#3,GRUPOSFL$,510
2100 FIELD #3,2 AS GRUPOSF$,254 AS GRUPOS1$,254 AS GRUPOS2$
2150 OPEN "R",#4,DEFTRANFL$,102
2200 FIELD #4,2 AS DEFTRANF$,100 AS DEFTRAND$
2250 OPEN "R",#5,PARAMFL$,102
2300 FIELD #5,2 AS SPF$,100 AS SPD$
2350 SPRRN=1
2400 GET #5,SPRRN
2450 IF CVI(SPF$)<>SPRRN THEN ER$="E01":E1%=1
2500 POLL$=MID$(SPD$,1,2)
2550 Q1%=INT(ASC(LEFT$(POLL$,1)))-48:IF Q1%>10 THEN
     Q1%=Q1%-7
2600 Q2%=INT(ASC(RIGHT$(POLL$,1)))-48:IF Q2%>10 THEN
     Q2%=Q2%-7
2650 PC$=CHR$((16*Q1%+Q2%) AND 127)
2700 DEF SEG=DRIVERSEG
2750 POKE 4,ASC(PC$)            '__ POLL CODE EN EL DRIVER
2800 SC$=CHR$((16*Q1%+Q2%) AND 127)
2850 POKE 2,ASC(SC$)            '__ SELECT CODE EN EL DRIVER
2900 TA1$=SC$:TA2$=TA1$
2950 SPEED$=MID$(SPD$,3,5)
3000 RVRRN=2:K=1
3050 GET #2,RVRRN
3100 WHILE RVRRN = CVI(RVF$)
3150 FOR J=1 TO 10
3200 RV$(K)=MID$(RVD$,1+(J-1)*10,10):K=K+1
3250 NEXT J
3300 RVRRN=RVRRN+1
3350 GET #2,RVRRN
3400 WEND
3450 CLOSE #2
3500 FOR GRUPOSRRN=1 TO 9
3550 GET #3,GRUPOSRRN
3600 IF CVI(GRUPOSF$) <>GRUPOSRRN THEN ER$="301" ELSE
          T1$(GRUPOSRRN)=GRUPOS1$:T2$(GRUPOSRRN)=GRUPOS2$
3650 NEXT GRUPOSRRN
3700 CLOSE #3
3750 GOSUB 6350            '___ INICIALIZAR EL DRIVER DE
                                COMUNICACIONES
3800 RETURN
3850 '_____________________________________________________'
3900 '      LINEA DE STATUS                                '
3950 '_____________________________________________________'
4000 KEY OFF
```

```
4050 COLOR COLPAN1,COLPAN2:CLS
4100 GR$=RIGHT$(STR$(GR),1)
4150 COLOR COLSTL1,COLSTL2
4250 LOCATE 25,1
4300 Q1%=INT(ASC(LEFT$(POLL$,1)))-48:IF Q1%>10 THEN
     Q1%=Q1%-7
4350 Q2%=INT(ASC(RIGHT$(POLL$,1)))-48:IF Q2%>10 THEN
     Q2%=Q2%-7
4400 PC$=CHR$((16*Q1%+Q2%) AND 127)
4450 PRINT "F1=CONFG\F2=AYUDA\F3=SELEC\F10=FIN"
            +ER$+POLL$+PC$+SPEED$+"    GR:"+GR$+
            "          FUNCION:    (     )";
4500 DEF SEG=DRIVERSEG
4550 POKE 4,ASC(PC$)        '__ POLL CODE EN EL DRIVER
4600 Q1%=Q1% OR 1
4650 SC$=CHR$((16*Q1%+Q2%) AND 127)
4700 POKE 2,ASC(SC$)        '__ SELECT CODE EN EL DRIVER
4750 TA1$=SC$:TA2$=TA1$
4800 HDR$=CHR$(2)+CHR$(126)+CHR$(126)+CHR$(36)+CHR$(49)+
          TA1$+TA2$+CHR$(65)+CHR$(64)+"0010"+CHR$(113)
          +"0001"+CHR$(31)
4850 RETURN
4900 '_____________________________________________________'
4950 '   LINEA DE  STATUS DE UNA TRANSACCION                '
5000 '_____________________________________________________'
5050 CLS:LOCATE 25,1
5100 COLOR COLSTL1,COLSTL2
5150 PRINT "F1=TRAN/F2=LIMP/F3=BORR/F5=IMPR";
     POLL$;" ";SPEED$;" GR:";GR$;
5200 COLOR COLPAN1,COLPAN2
5250 RETURN
5300 '_____________________________________________________'
5350 '   LAZO PRINCIPAL                                     '
5400 '   ESTA RUTINA ESTA A LA ESPERA DE ALGUNA ACCION      '
5450 '   DEL OPERADOR                                       '
5500 '_____________________________________________________'
5550 ON KEY(1) GOSUB 6050            ' CONFIGURAR EL TERMINAL
5600 ON KEY(2) GOSUB 8500            ' AYUDA
5650 ON KEY(3) GOSUB 9200            ' SELECCIONAR GRUPO
5700 ON KEY(10) GOSUB 10850           ' FIN
5750 COLOR COLPAN1,COLPAN2:LOCATE 24,1:PRINT "          ";
5800 KEY(1) ON:KEY(2) ON:KEY(3) ON:KEY(10) ON
5850 RV1=200:II=25:JJ=76:LL=4:QQ$="":Q%=0:V0=0:I=0:
     GOSUB 26750        '___INGRESAR NUMERICO
5900 IF Q%>0 THEN CT%=Q% ELSE GOTO 6000
5950 GOSUB 11600          '___ PROCESAR UNA TRANSACCION
6000 RETURN
6050 '_____________________ CONFIGURAR EL TERMINAL
6100 FOR I=1 TO 10:KEY(I) OFF:NEXT I
6150 GOSUB 7500
6200 GOSUB 3850
6250 Q%=0
6300 RETURN 6000
6350 '_____________________________________________________'
6400 '    LAS SIGUIENTES INSTRUCCIONES PROGRAMAN           '
```

```
6450 '     EL 8250 (COM1) Y LLAMAN A LA RUTINA QUE      '
6500 '     CAMBIA LA DIRECCION DE INTERRUPCION          '
6550 '_____________________________________________________'
6600 DEF SEG=DRIVERSEG          '__ SEGMENTO DE LA RUTINA DE
                                    INTERRUPCION
6650 POKE 4,ASC(PC$)            '__ POLL CODE
6700 POKE 6,&H1                 '__ DRIVER STATUS (ESPERE EOT)
6750 POKE 0,1                   '__ "ENVMSG" NO HAY MENSAJE QUE
                                    ENVIAR
6800 POKE 1,1                   '__ NO ACEPTE SELECCION
6850 OUT &H3FB,128              '__ HABILITAR PROGRAMACION
                                    VELOCIDAD EN EL 8250
6900 IF SPEED$=" 1200" THEN OUT &H3F8,&H60  '__ PROGRAMAR
                                                VELOCIDAD
6950 IF SPEED$=" 1800" THEN OUT &H3F8,&H40  '    EN EL BYTE
                                        MENOS SIGNIFICATIVO
7000 IF SPEED$=" 2400" THEN OUT &H3F8,&H30
7050 IF SPEED$=" 4800" THEN OUT &H3F8,&H18
7100 IF SPEED$=" 9600" THEN OUT &H3F8,&HC
7150 IF SPEED$="19200" THEN OUT &H3F8,&H6
7200 OUT &H3F9,0                '__  BYTE MAS SIGNIFICATIVO
7250 OUT &H3FB,&H1A             '__  PROGRAMAR PARIDAD,
                                     STOP BITS
7300  OUT &H3F9,1                '__ HABILITAR INTERRUPCION
                                     POR DATOS
7350 OUT &H3FC,13               '__ LEVANTAR DTR
7400 OUT &H21,&HAC              '__ HABILITA LA INTERRUPCION EN
                                    EL PIC
7450 RETURN
7500 '_____________________ CONFIGURACION DEL TERMINAL
7550 CLS:LOCATE 1,1:COLOR COLSTL1,COLSTL2
7600 PRINT SPACE$(25)+
     "CONFIGURACION    DEL    TERMINAL "+SPACE$(26);
7650 COLOR COLPAN1,COLPAN2
7700 LOCATE 4,10:PRINT "POLL CODE ACTUAL:    "+POLL$+
     "  ENTRE EL NUEVO:          ";
7750 RV1=200:QQ$="":II=4:JJ=52:LL=2:I=0:GOSUB 25000
7800 IF LEN(QQ$) < 2 THEN BEEP:GOTO 7700
7850 FOR K=1 TO 2
7900 IF MID$(QQ$,I,1) > "F" OR MID$(QQ$,I,1) < "0" THEN
        QQ$="XX"
7950 NEXT K
8000 IF QQ$ = "XX" THEN BEEP:GOTO 7700
8050 IF QQ$ < "20" THEN BEEP:GOTO 7700
8100 POLL$=QQ$
8150 LOCATE 5,10:PRINT "BAUD RATE ACTUAL:   "+SPEED$+
        "  ENTRE EL NUEVO:          ";
8200 QQ$="":II=5:JJ=52:LL=5:I=0:GOSUB 25000
8250 IF LEN(QQ$)<5 THEN QQ$=SPACE$(5-LEN(QQ$))+QQ$
8300 IF  QQ$ = " 1200" OR QQ$ = " 1800" OR QQ$ = " 2400"  OR
        QQ$="  4800"  OR QQ$=" 9600" OR QQ$ =  "19200"  THEN
        GOTO 8350 ELSE BEEP:GOTO 8150
8350 SPEED$=QQ$:SPRRN=1:Q$=POLL$+SPEED$+STRING$(93,0):
        LSET SPD$=Q$:LSET SPF$=MKI$(SPRRN):PUT #5,SPRRN
8400 GOSUB 6350     '  ____ TRABAJAR ON LINE
```

```
8450 RETURN
8500 '_____________________ AYUDA
8550  FOR I=1 TO 10:KEY(I)  OFF:NEXT I
8600 J=3:CLS:LOCATE 1,1:COLOR COLSTL1,COLSTL2
8650 PRINT SPACE$(28)+"TRANSACCIONES DISPONIBLES"
        +SPACE$(27);:COLOR COLPAN1,COLPAN2
8700 IF GR=0 THEN LOCATE J,11:
        PRINT "<F3>          SELECCION DE GRUPO";:GOTO 9100
8750 FOR I=1 TO 253 STEP 2
8800 IF ASC(MID$(T2$(GR),I,1)) = 255 THEN 9100
8850 TT%=CVI(MID$(T2$(GR),I,2))
8900 GET #4,TT%:Q$=DEFTRAND$
8950 IF J>20 THEN II=24:JJ=76:LL=1:GOSUB 26750:J=3
9000 LOCATE J,10:PRINT CVI(MID$(Q$,2,2)); TAB(20)
        MID$(Q$,6,30);:J=J+1
9050 NEXT I
9100 GOSUB 4100
9150 RETURN 6000
9200 '_____________________ SELECCION DE GRUPO
9250 FOR I=1 TO 10:KEY(I) OFF:NEXT I
9300 CLS:LOCATE 1,1:COLOR COLSTL1,COLSTL2
9350 PRINT SPACE$(30)+
        "SELECCION DE GRUPO "+SPACE$(30);:
        COLOR COLPAN1,COLPAN2
9400 LOCATE 4,1:PRINT "CLAVE:"
9450 LOCATE 5,1:PRINT "GRUPO:"
9500  QQ$="":II=4:JJ=15:LL=6:RV1=0:I=0:GOSUB 25000
9550 IF LEN(QQ$)<6 THEN QQ$=STRING$(6-LEN(QQ$),71)+QQ$
9600 K=0
9650  IF  QQ$ <> "987654" THEN LOCATE  22,1:
        PRINT "PASSWORD INVALIDO        ";:GOTO 9500
9700 QQ$="":II=5:JJ=15:LL=1:I=0:GOSUB 26750
9750 GR=Q%
9800 GOSUB 3850
9850 Q%=0
9900 RETURN 6000
9950 '_____________________ IMPRIMIR FORMATOS
10000 ON ERROR GOTO 31600
10050 COLOR COLINP1+16,COLINP2:LOCATE 24,1:PRINT "IMPRESOR";
10100 FOR I%=1 TO 5:KEY(I%) OFF:NEXT I%
10150 DEF SEG=&HB000
10200 WIDTH "LPT1:",80
10250 LPRINT CHR$(12)
10300 FOR I%=1 TO 25
10350 K%=80*(I%-1)
10400 Q$=SPACE$(80)
10450 FOR J%=0 TO 79:
        MID$(Q$,J%+1,1)=CHR$(PEEK(-32768!+2*(K%+(J%))))
        :NEXT J%
10500 IF Q$=SPACE$(80) THEN Q$=CHR$(10)
10550 LPRINT Q$;
10600 NEXT I%
10650 LPRINT CHR$(13)
10700 ON ERROR GOTO 30850
10750 FOR I%=1 TO 5:KEY(I%) ON:NEXT I%
```

```
10800 RETURN
10850 '_____________________ FIN DEL PROGRAMA
10900 ONOFF$="FIN"
10950 RETURN 6000
11000 '_________________________ INGRESAR NUMERICO
11050 QQ$=""
11100 LOCATE II,JJ,1
11150 IF LEN(QQ$)=0 THEN KEY(1) ON:KEY(2) ON
      :KEY(3) ON:KEY(4) ON:KEY(5) ON:KEY(10) ON
      ELSE KEY(1) OFF:KEY(2) OFF :KEY(3) OFF:KEY(4) OFF
      :KEY(5) OFF:KEY(10) OFF
11200 Q$=INKEY$
11250 IF LEN(Q$) =2 THEN Q$=RIGHT$(Q$,1) ELSE GOTO 11350
11300 IF Q$=CHR$(75) THEN IF LEN(QQ$)>0 THEN
        WLEN=WLEN-1:JJ=JJ-1:QQ$=LEFT$(QQ$,LEN(QQ$)-1):
        GOTO 11100
11350 IF LEN(Q$)=0 THEN GOTO 11100
11400 IF ASC(Q$)=13 THEN RETURN
11450 IF ASC(Q$)<48 OR ASC(Q$)>57 THEN BEEP:GOTO 11100
11500 LOCATE II,JJ:PRINT Q$;
11550 QQ$=QQ$+Q$:Q%=VAL(QQ$):JJ=JJ+1:WLEN=WLEN-1:
        IF WLEN<=0 THEN RETURN ELSE GOTO 11100
11600 '_____________________________________________________'
11650 '      PROCESO DE UNA TRANSACCION                     '
11700 '_____________________________________________________'
11750 FOR I=1 TO 10:KEY(I)  OFF:NEXT I
11800 GOSUB 12150                '___ VALIDAR LA TRANSACCION
11850 IF V0=0 THEN GOSUB 12750     '___ PRESENTAR EL FORMATO
11900 IF V0=0 THEN GOSUB 14350          '___ INGRESO DE DATOS
11950 IF V0=0 THEN GOSUB 16200          '___ COMUNICACIONES
12000  IF  V0=0  THEN GOSUB 31400       '___  PROCESO  DEL
                                         MENSAJE DE CENTRAL
12050 IF V0 < 9 THEN V0=0:GOTO 11900    '____ INGRESAR NUEVOS
                                              DATOS
12100 RETURN
12150 '_____________________ VALIDAR LA TRANSACCION
12200  V0=9:IF  GR=0 THEN LOCATE  24,11:
             PRINT "TERMINAL NO INGRESADO";:RETURN
12250 I=1
12300 IF MID$(T1$(GR),I,2)=STRING$(2,255) THEN
        LOCATE 24,11:PRINT "TRANSACCION NO EXISTE";:RETURN
12350 IF MID$(T1$(GR),I,2)=MKI$(CT%) THEN GOTO 12450
12400 I=I+2:GOTO 12300
12450 LOCATE 24,1:PRINT SPACE$(80);
12500 TT%=CVI(MID$(T2$(GR),I,2))
12550 K=INT((4+INSTR(RT$,CHR$(255)+MKI$(CT%)))/5)
12600 GOSUB 23600          '___ACTUALIZAR TRANSACCIONES
                                RESIDENTES
12650 V0=0
12700 RETURN
12750 '_____________________ PRESENTAR EL FORMATO
12800 GOSUB 4900
12850 SC%=CVI(MID$(DEFTRAN$(K),44,2))
12900 K=INT((4+INSTR(RS$,CHR$(255)+MKI$(SC%)))/5)
12950 GOSUB 24150             '__ ACTUALIZAR FORMATOS
```

```
13000 I=1
13050 SDLEN%=CVI(MID$(SH$(K),5,2))
13100 RVLEN%=CVI(MID$(SH$(K),7,2))
13150 OVLEN%=CVI(MID$(SH$(K),9,2))
13200 OC$="":IF OVLEN%>RVLEN% THEN
            OC$=MID$(SCDATA$(K),RVLEN%+1,OVLEN%-RVLEN%)
13250 COLOR COLPAN1,COLPAN2
13300 WHILE I < SDLEN%
13350 J=I:J1=I+1:J2=I+2:J3=I+3
13400 LOCATE ASC(MID$(SCDATA$(K),J,1)),
            ASC(MID$(SCDATA$(K),J1,1))
13450 PRINT MID$(SCDATA$(K),J3,ASC(MID$(SCDATA$(K),J2,1)))
13500 I=I+3+ASC(MID$(SCDATA$(K),J2,1))
13550 WEND
13600 COLOR COLINP1,COLINP2
13650 IN%=1
13700 WHILE I < RVLEN%
13750 J=I:J1=I+1:J2=I+2:J3=I+3:J4=I+4:J5=I+5
13800 LOCATE ASC(MID$(SCDATA$(K),J2,1)),
            ASC(MID$(SCDATA$(K),J3,1))
13850 IF (1 AND
         ASC(LEFT$(RV$(ASC(MID$(SCDATA$(K),J4,1))),1)))   = 0
         THEN GOTO 14050
13900 S$=MID$(SCDATA$(K),J5,ASC(MID$(SCDATA$(K),J1,1)))
13950 I=I+ASC(MID$(SCDATA$(K),J1,1))
14000 GOTO 14100
14050 S$=SPACE$(ASC(MID$(SCDATA$(K),J1,1)))
14100 PRINT S$;
14150 IN$(IN%)=MID$(SCDATA$(K),J,1)+MID$(SCDATA$(K),J1,1)
        +MID$(SCD ATA$(K),J2,1)+MID$(SCDATA$(K),J3,1)
        +MID$(SCDATA$(K),J4,1)+"0 "+S$:IN%=IN%+1
14200 I=I+5
14250 WEND
14300 RETURN
14350 '_____________________ INGRESO DE DATOS
14400 I=1:EOI=0
14450 WHILE EOI=0
14500 COLOR COLINP1,COLINP2
14550 ON KEY(1) GOSUB 29550    '___ TRANSMITIR
14600 ON KEY(2) GOSUB 29700    '___ LIMPIAR CAMPOS
14650 ON KEY(3) GOSUB 30450    '___ BORRAR LA PANTALLA
                               (REGRESA A LAZO PRINCIPAL)
14700 ON KEY(5) GOSUB 30700    '___ IMPRIMIR LA PANTALLA
14750 IF I=IN% THEN I=1:KEY(1) ON
14800 DEF SEG=&HB000
14850 SQQ$=""
14900 FOR SI%=ASC(MID$(IN$(I),4,1)) TO
         ASC(MID$(IN$(I),4,1))+ASC(MID$(IN$(I),2,1))
14950 SQQ$=SQQ$
        +CHR$(PEEK(-32768! +2*(80*(ASC(MID$(IN$(I),3,1))-1)
        +(SI%-1))))
15000 NEXT SI%
15050 IF LEN(SQQ$)>0 THEN IF RIGHT$(SQQ$,1)=" " THEN
         SQQ$=LEFT$(SQQ$,LEN(SQQ$)-1):GOTO 15050
```

```
15100 QQ$=MID$(IN$(I),7)
15150 QQ$=SQQ$
15200 IF LEN(QQ$)>0 THEN IF RIGHT$(QQ$,1)=" " THEN
        QQ$=LEFT$(QQ$,LEN(QQ$)-1):GOTO 15200
15250 IF QQ$=SPACE$(LEN(QQ$)) THEN QQ$=""
15300 II=ASC(MID$(IN$(I),3,1))
15350 JJ=ASC(MID$(IN$(I),4,1))+LEN(QQ$)
15400 LL=ASC(MID$(IN$(I),2,1))-LEN(QQ$)
15450 MID$(IN$(I),6,1)="0"
15500 RV1=ASC(LEFT$(RV$(ASC(MID$(IN$(I),5,1))),1))
15550 LOCATE 24,1:COLOR COLINP1+16,COLINP2
15600 MODULO$="INGRESAR DATA":KEY(2) ON:KEY(3) ON
        :KEY(4) ON:KEY(5) ON:KEY(6) ON
15650 IF RV1>191 THEN PRINT "ALFA-NUM";:
        COLOR COLINP1,COLINP2:GOSUB 25000:
        GOTO 16100   '___ INGRESAR ALFA-NUM
15700 IF RV1>127 THEN PRINT "ALFA    ";:
        COLOR  COLINP1,COLINP2:GOSUB 25850:
        GOTO 16100   '___ INGRESAR ALFA
15750 IF RV1>111 THEN PRINT "MMDDAA  ";:
        COLOR COLINP1,COLINP2:GOSUB 27650:
        GOTO 16100   '___ INGRESAR FECHA 1
15800 IF RV1>103 THEN PRINT "DDMMAA  ";:
        COLOR  COLINP1,COLINP2:GOSUB 27950:
        GOTO 16100   '___ INGRESAR FECHA 2
15850 IF RV1>95 THEN PRINT "AAMMDD  ";:
        COLOR COLINP1,COLINP2:GOSUB 28250:
        GOTO 16100   '___ INGRESAR FECHA 3
15900 IF RV1>87 THEN PRINT "CLAVE   ";
        :COLOR COLINP1,COLINP2:GOSUB 25000:
        GOTO 16100   '___ INGRESAR CLAVE
15950 IF RV1>79 THEN PRINT "PASSWORD";
        :COLOR COLINP1,COLINP2:GOSUB 26750:
        GOTO 16100   '___ INGRESAR PASSWORD
16000 IF RV1>71 THEN PRINT "CUENTA  ";:
        COLOR COLINP1,COLINP2:GOSUB 29000:
        GOTO 16100   '___ INGRESAR CUENTA
16050 IF RV1>63 THEN PRINT "NUM     ";:
        COLOR  COLINP1,COLINP2:GOSUB 26750:
        GOTO 16100   '___ INGRESAR NUM
16100 WEND
16150 RETURN
16200 '____________________ COMUNICACIONES
16250 KEY (1) OFF
16300 US$=CHR$(31):Q$=""
16350 IOB$=HDR$+Q$
16400 FOR I=1 TO IN%-1
16450 IOB$=IOB$+LEFT$(IN$(I),1)+MID$(IN$(I),7)+US$
16500 NEXT I
16550 IOB$=IOB$+CHR$(3)+CHR$(255)
16600 CALL STRTOM(IOB$)            '___ ENVIAR AL DRIVER
16650 LOCATE 24,1: PRINT "COMM    ";
16700 DEF SEG=DRIVERSEG
16750 LOCATE 24,11:COLOR COLPAN1,COLPAN2:PRINT SPACE$(70);
16800 IOB$=SPACE$(512)
```

```
16850 POKE 0,2                '__ "ENVMSG" INFORMAR AL DRIVER
                                   QUE ENVIE
16900 POKE 1,2                     '__ ACEPTE SELECCION
16950 POKE 5,0                     '__ PONER 0 EN STATUS DE LA
                                       COMUNICACION
17000 I=0
17050 IF PEEK (5)>31 THEN I=18000 ELSE I=I+1  '__ VER
                                                STATUS(32 ES OK)
17100 IF I < 18000 THEN GOTO 17050    '__ SIGA EN LAZO
17150 POKE 0,1                     '__ NO EXISTE MENSAJE
17200 POKE 1,1                     '__ NO ACEPTE SELECCION
17250 IF PEEK (5)<32 THEN Q$="error en comunicaciones":
        GOSUB 18900:SOUND 930,13:V0=1:GOTO 17550
17300 CALL MTOSTR(IOB$):J=INSTR(IOB$,CHR$(3))  '__ RECIBIR
                                                  DEL DRIVER
17350 IF J> 0 THEN IOB$=LEFT$(IOB$,J)
17400 UL$=MID$(IOB$,6,1)
17450 GOSUB 17600            '____  PROCESAR MENSAJE DE CENTRAL
17500 IF V0=0 THEN IF UL$<>"A" THEN POKE 0,1:POKE 1,2:
      POKE  5,0:GOTO 17000    '____  GOTO RECIBIR (
                                   NO ES EL ULTIMO MENSAJE)
17550 RETURN
17600 '_____________________ PROCESAR MENSAJE DE CENTRAL
17650 IOB$ = MID$(IOB$,6)          '__ eliminar encabezado
17700 IF LEFT$(IOB$,1) = CHR$(31) THEN IOB$=MID$(IOB$,2)
17750 IF RIGHT$(IOB$,1) = CHR$(31) THEN
         IOB$=LEFT$(IOB$,LEN(IOB$)-1)
17800    X=INSTR(IOB$,CHR$(31)):IF  X=0  THEN   X=LEN(IOB$)+1
17850 IF LEN(IOB$)<>0 THEN Q$=MID$(IOB$,2,X-2):
         FC$=LEFT$(IOB$,1) ELSE RETURN
17900 IF V0=0 THEN GOSUB 18050    '___ PROCESAR 1 FC
17950 IF V0=0 THEN IF X=LEN(IOB$)+1 THEN IOB$="" ELSE
         IOB$=RIGHT$(IOB$,LEN(IOB$)-X):GOTO 17700
18000 RETURN
18050 '_____________________ PROC 1 FC
18100 IF LEN(Q$) =0 THEN Q$=" "
18150 IF FC$=CHR$(113) THEN V0=1:SOUND 930,13:RETURN
         '__FC 71 (VOID)
18200 IF FC$=CHR$(33) THEN GOSUB 18900:RETURN     '__FC 21
18250 IF FC$=CHR$(34) THEN GOSUB 19100:RETURN     '__FC 22
18300 IF FC$=CHR$(35) THEN GOSUB 19350:RETURN     '__FC 23
18350 IF FC$=CHR$(36) THEN GOSUB 20050:RETURN     '__FC 24
18400 IF FC$=CHR$(37) THEN GOSUB 20350:RETURN     '__FC 25
18450 IF FC$=CHR$(38) THEN GOSUB 20500:RETURN     '__FC 26
18500 IF FC$=CHR$(48) THEN GOSUB 20800:RETURN     '__FC 30
18550 IF FC$=CHR$(49) THEN GOSUB 20950:RETURN     '__FC 31
18600 IF FC$=CHR$(52) THEN GOSUB 21400:RETURN     '__FC 34
18650 IF FC$=CHR$(54) THEN GOSUB 21950:RETURN     '__FC 36
18700 IF FC$=CHR$(55) THEN GOSUB 22550:RETURN     '__FC 37
18750 IF FC$=CHR$(56) THEN GOSUB 23000:RETURN     '__FC 38
18800 V0=1:Q$="FC NO SOPORTADO  "+HEX$(ASC(FC$)):
        GOSUB 18900:RETURN
18850 RETURN
18900 '_____________________ FC 21: PRESENTAR DATO EN LINEA 24
18950 LOCATE 24,11:COLOR COLOUT1,COLOUT2:PRINT SPACE$(70);
```

```
19000 LOCATE 24,11:PRINT Q$;
19050 RETURN
19100 ' FC 22: PRESENTAR UNA  PANTALLA ASOCIADA A UNA TRAN.
19150 IF LEN(Q$) < 4 THEN LOCATE 23,11:
      PRINT "fc22, numero de transaccion errada";:RETURN
19200 CT%=VAL(LEFT$(Q$,4))
19250 GOSUB 12150:IF V0=0 THEN GOSUB 12750
19300 RETURN
19350 ' FC 23: LLENAR DESDE UN CAMPO DE SALIDA DADO
19400 IF LEN(Q$) < 2 THEN LOCATE 23,11:
        PRINT "fc23, longitud errada";:RETURN
19450 IF VAL(LEFT$(Q$,2)) = 0 OR
         VAL(LEFT$(Q$,2)) > LEN(OC$)/3 THEN LOCATE 23,11:
         COLOR  COLOUT1,COLOUT2:
         PRINT "fc23: numero de campo errado";:RETURN
19500 W1%=VAL(LEFT$(Q$,2)):Q$=MID$(Q$,3)
19550 OC1$=MID$(OC$,3*(W1%-1)+1)
19600 COLOR COLOUT1,COLOUT2
19650 WHILE LEN(Q$) > 0
19700 LOCATE ASC(MID$(OC1$,2,1)),ASC(MID$(OC1$,3,1))
19750 IF LEN(Q$)<ASC(MID$(OC1$,1,1)) THEN
         Q$=Q$+SPACE$(ASC(MID$(OC1$,1,1))-LEN(Q$))
19800 PRINT MID$(Q$,1,ASC(MID$(OC1$,1,1)));
19850 IF ASC(MID$(OC1$,1,1))>=LEN(Q$) THEN Q$=""
      ELSE Q$=MID$(Q$,1+ASC(MID$(OC1$,1,1)))
19900 IF LEN(OC1$)>3 THEN OC1$=MID$(OC1$,4) ELSE Q$=""
19950 WEND
20000 RETURN
20050 ' FC 24: DISPLAY EN LINEA N,POSICION M
20100  IF LEN(Q$) < 4 THEN LOCATE 23,11:
        PRINT "fc24, campo muy corto";:RETURN
20150 IF VAL(LEFT$(Q$,2)) > 24 OR VAL(LEFT$(Q$,2))=0 THEN
         MID$(Q$,1,2)="22"
20200 IF VAL(MID$(Q$,3,2)) > 76-LEN(Q$) OR
         VAL(MID$(Q$,3,2))=0 THEN MID$(Q$,3,2)="01"
20250 LOCATE VAL(LEFT$(Q$,2)),VAL(MID$(Q$,3,2)):
      COLOR COLOUT1,COLOUT2:PRINT MID$(Q$,5);
20300 RETURN
20350 ' FC 25: BORRAR CAMPOS DE LA PANTALLA
20400 GOSUB 29700
20450 RETURN
20500 ' fc 26: borrar desde linea n a linea m
20550 IF VAL(LEFT$(Q$,2)) = 0 OR
         VAL(LEFT$(Q$,2)) > VAL(MID$(Q$,3,2)) OR
         VAL(MID$(Q$,3,2))  >  21  THEN  LOCATE  23,11:
         PRINT "fc26: numero de lineas errados";:RETURN
20600 FOR W1%=VAL(LEFT$(Q$,2)) TO VAL(MID$(Q$,3,2))
20650 LOCATE W1%,1:PRINT SPACE$(80);
20700 NEXT W1%
20750 RETURN
20800 '_____________________ FC 30: IMPRIMIR LA PANTALLA
20850 GOSUB 9950
20900 RETURN
20950 '_____________________ FC 31: IMPRIMIR UNA LINEA
21000 ON ERROR GOTO 21250
```

```
21050 COLOR COLINP1+16,COLINP2:LOCATE 24,1:PRINT "IMPRESOR";
21100 LPRINT Q$
21150 ON ERROR GOTO 30850
21200 RETURN
21250 '_____________________ON ERROR
21300 GOSUB 31100
21350 IF  I$="X" OR I$="x" THEN RESUME 21150 ELSE RESUME
21400 '_____________________ FC 34: AVANZAR N LINEAS
21450 ON ERROR GOTO 21800
21500 I=VAL(LEFT$(Q$,2))
21550 IF I=0 OR I > 60 THEN I=1
21600 COLOR COLINP1+16,COLINP2:LOCATE 24,1:PRINT "IMPRESOR";
21650 FOR J=1 TO I:LPRINT CHR$(10):NEXT J
21700 ON ERROR GOTO 30850
21750 RETURN
21800 '_____________________ON ERROR
21850 GOSUB 31100
21900 IF  I$="X" OR I$="x" THEN RESUME 21700 ELSE RESUME
21950 ' FC 36: PONER CAMPO EN POSICION -N- E IMPRIMIR LINEA
22000 ON ERROR GOTO 22400
22050 IF VAL(LEFT$(Q$,3)) > 215-LEN(Q$) THEN
         MID$(Q$,1,3)="001"
22100 MID$(QQ$,VAL(LEFT$(Q$,3)))=MID$(Q$,4)
22150 COLOR COLINP1+16,COLINP2:LOCATE 24,1:PRINT "IMPRESOR";
22200 LPRINT QQ$
22250 QQ$=SPACE$(210)
22300 ON ERROR GOTO 30850
22350 RETURN
22400 '_____________________ON ERROR
22450 GOSUB 31100
22500 IF  I$="X" OR I$="x" THEN RESUME 22300 ELSE RESUME
22550 '_____________________ FC 37: AVANZAR PAGINA
22600 ON ERROR GOTO 22850
22650 COLOR COLINP1+16,COLINP2:LOCATE 24,1:PRINT "IMPRESOR";
22700 LPRINT CHR$(12);
22750 ON ERROR GOTO 30850
22800 RETURN
22850 '_____________________ON ERROR
22900 GOSUB 31100
22950 IF  I$="X" OR I$="x" THEN RESUME 22750 ELSE RESUME
23000 '_____________________ FC 38: IMPRESION NORMAL
23050 ON ERROR GOTO 23300
23100 COLOR COLINP1+16,COLINP2:LOCATE 24,1:PRINT "IMPRESOR";
23150 LPRINT CHR$(18);:WIDTH "LPT1:",132
23200 ON ERROR GOTO 30850
23250 RETURN
23300 '_____________________ON ERROR
23350 GOSUB 31100
23400 IF  I$="X" OR I$="x" THEN RESUME 23200 ELSE RESUME
23450 '_____________________ON ERROR
23500 GOSUB 31100
23550 IF  I$="X" OR I$="x" THEN RESUME 23800 ELSE RESUME
23600 '_____________________ ACTUALIZAR TRANSACCIONES RESIDENTES
23650 TN%=TN%+1
23700 IF LEN(RT$)=250 THEN GOSUB 23900
```

```
23750 IF K=0 THEN RT$=RT$+CHR$(255)+MKI$(CT%)+MKI$(TN%):
         K=LEN(RT$)/5:GET #4,TT%:DEFTRAN$(K)=DEFTRAND$
23800 MID$(RT$,5*(K-1)+4,2)=MKI$(TN%)
23850 RETURN
23900 '_________________ ELIMINAR UNA TRANSACCION RESIDENTE
23950 FOR I=1 TO 50
24000 IF CVI(MID$(RT$,5*(I-1)+4,2)) < TN%-49 THEN
         MID$(RT$,5*(I-1)+1,5)=RIGHT$(RT$,5):RT$=LEFT$(RT$,245)
24050 NEXT I
24100 RETURN
24150 ' ACTUALIZAR LISTA DE FORMATOS RESIDENTES
24200 SN%=SN%+1
24250 IF LEN(ST$)=100 THEN GOSUB 24450
24300 IF K=0 THEN RS$=RS$+CHR$(255)+MKI$(SC%)+MKI$(SN%):
         K=LEN(RS$)/5:GET #1,SC%:SH$(K)=SDH$:
         SCDATA$(K)=SD$:IF    CVI(MID$(SH$(K),3,2))>0   THEN
         GOSUB 24700
24350 MID$(RS$,5*(K-1)+4,2)=MKI$(SN%)
24400 RETURN
24450 '___________________ ELIMINAR FORMATOS RESIDENTES
24500 FOR I=1 TO 20
24550 IF CVI(MID$(RS$,5*(I-1)+4,2)) < SN%-19 THEN
         MID$(RS$,5*(I-1)+1,5)=RIGHT$(RS$,5):
         RS$=LEFT$(RS$,95):SCDATA$(I)=SCDATA$(2 0):
         SCDATA$(20)=""
24600 NEXT I
24650 RETURN
24700 '__________________ LEER SIGUIENTE RECORD
24750 SCRRN=CVI(MID$(SH$(K),3,2))
24800 WHILE SCRRN <> 0
24850 GET #1,SCRRN
24900 SCDATA$(K)=SCDATA$+SD$
24950 WEND
25000 '_____________________ INGRESAR ALFA-NUMERICO
25050 LOCATE II,JJ,1
25100 IF LL>0 THEN PRINT " ";:LOCATE II,JJ,1
25150 IF V0>0 THEN RETURN
25200 Q$=INKEY$
25250 IF LEN (Q$)=0 THEN GOTO 25150
25300 IF LEN(Q$)=2 THEN Q$=RIGHT$(Q$,1) ELSE IF ASC(Q$) <> 8
          THEN GOTO 25500
25350 IF Q$=CHR$(75) OR ASC(Q$) = 8 THEN IF LEN(QQ$)>0  THEN
         LL=LL+1:JJ=JJ-1:QQ$=LEFT$(QQ$,LEN(QQ$)-1):
         KEY(1) OFF:GOTO 25050 ELSE BEEP:GOTO 25050
25400 IF Q$=CHR$(72) THEN IF I>1 THEN
         IN$(I)=LEFT$(IN$(I),5)+"0"+QQ$:I=I-1:GOTO 25800
      ELSE BEEP:GOTO 25150
25450 BEEP:GOTO 25150
25500 IF ASC(Q$)=13 THEN GOTO 25750
25550 IF LL=0 THEN BEEP:GOTO 25050
25600 IF ASC(Q$)<32 OR ASC(Q$)>128 THEN BEEP:GOTO 25050
25650 LOCATE II,JJ:IF RV1 > 191 THEN PRINT Q$; ELSE
         PRINT "*";
25700 QQ$=QQ$+Q$:JJ=JJ+1:LL=LL-1:IF LL>0 THEN GOTO 25050
25750 IN$(I)=LEFT$(IN$(I),5)+"1"+QQ$:I=I+1
```

```
25800 RETURN
25850 '_____________________ INGRESAR ALFA
25900 LOCATE II,JJ,1
25950 IF LL>0 THEN PRINT " ";:LOCATE II,JJ,1
26000 IF VO>0 THEN RETURN
26050 Q$=INKEY$
26100 IF LEN (Q$)=0 THEN GOTO 26000
26150 IF LEN(Q$)=2 THEN Q$=RIGHT$(Q$,1) ELSE IF ASC(Q$) <> 8
        THEN GOTO 26350
26200 IF Q$=CHR$(75) OR ASC(Q$) = 8 THEN IF LEN(QQ$)>0  THEN
      LL=LL+1:JJ=JJ-1:QQ$=LEFT$(QQ$,LEN(QQ$)-1):KEY(1) OFF:
      GOTO 25900 ELSE BEEP:GOTO 25900
26250 IF Q$=CHR$(72) THEN IF I>1 THEN
      IN$(I)=LEFT$(IN$(I),5)+"0"+QQ$:I=I-1:GOTO 26700 ELSE
      BEEP:GOTO 26000
26300 BEEP:GOTO 26000
26350 IF ASC(Q$)=13 THEN GOTO 26650
26400 IF LL=0 THEN BEEP:GOTO 25900
26450 IF ASC(Q$)<32 OR ASC(Q$)>122 THEN BEEP:GOTO 25900
26500 IF ASC(Q$)>32 AND ASC(Q$)<64 THEN BEEP:GOTO 25900
26550 LOCATE II,JJ:PRINT Q$;
26600 QQ$=QQ$+Q$:JJ=JJ+1:LL=LL-1:IF LL>0 THEN GOTO 25900
26650 IN$(I)=LEFT$(IN$(I),5)+"1"+QQ$:I=I+1
26700 RETURN
26750 '_____________________ INGRESAR NUMERICO
26800 LOCATE II,JJ,1
26850 IF LL>0 THEN PRINT " ";:LOCATE II,JJ,1
26900 IF VO>0 THEN RETURN
26950 Q$=INKEY$
27000 IF LEN (Q$)=0 THEN GOTO 26900
27050 IF LEN(Q$)=2 THEN Q$=RIGHT$(Q$,1) ELSE IF ASC(Q$) <> 8
        THEN GOTO 27300
27100  IF Q$=CHR$(75) OR ASC(Q$) = 8 THEN IF LEN(QQ$)>0 THEN
       LL=LL+1:JJ=JJ-1:QQ$=LEFT$(QQ$,LEN(QQ$)-1):KEY(1) OFF:
       GOTO 26800 ELSE BEEP:GOTO 26800
27150 IF Q$=CHR$(72) THEN IF I>1 THEN
        IN$(I)=LEFT$(IN$(I),5)+"0"+QQ$:I=I-1:GOTO 27600
        ELSE BEEP:GOTO 26900
27200 BEEP:GOTO 26900
27250 IF LEN (Q$)=0 THEN GOTO 26900
27300 IF ASC(Q$)=13 THEN GOTO 27550
27350 IF LL=0 THEN BEEP:GOTO 26800
27400 IF ASC(Q$)<48 OR ASC(Q$)>57 THEN BEEP:GOTO 26800
27450 LOCATE II,JJ:IF RV1 > 79 AND RV1 < 88 THEN PRINT  "*";
      ELSE PRINT Q$;
27500 QQ$=QQ$+Q$:JJ=JJ+1:LL=LL-1:IF LL>0 THEN GOTO 26800
27550 IN$(I)=LEFT$(IN$(I),5)+"1"+QQ$:I=I+1:IF VAL(QQ$)<30000
      THEN Q%=VAL(QQ$)
27600 RETURN
27650 '_____________________ INGRESAR FECHA 1  MMDDAA
27700 GOSUB 26750      '_______ INGRESAR NUMERICO
27750 IF Q$=CHR$(72) OR VO>0 THEN RETURN
27800 IF LEN(QQ$) = 6 THEN
        QQ$=MID$(QQ$,5,2)+MID$(QQ$,1,2)+MID$(QQ$,3,2)
27850 GOSUB 28500      '____ val fecha
```

```
27900 RETURN
27950 '_____________________ INGRESAR FECHA 2  DDMMAA
28000 GOSUB 26750      '_______ INGRESAR NUMERICO
28050 IF Q$=CHR$(72) OR VO>0 THEN RETURN
28100 IF LEN(QQ$) = 6 THEN
         QQ$=MID$(QQ$,5,2)+MID$(QQ$,3,2)+MID$(QQ$,1,2)
28150 GOSUB 28500      '____ val fecha
28200 RETURN
28250 '_____________________ INGRESAR FECHA 3  AAMMDD
28300 GOSUB 26750      '_______ INGRESAR NUMERICO
28350 IF Q$=CHR$(72) OR VO>0 THEN RETURN
28400 GOSUB 28500      '____ val fecha
28450 RETURN
28500 '_____________________ VALIDAR FECHA
28550 IF LEN(QQ$) <> 6 THEN GOTO 28850
28600 IF VAL(MID$(QQ$,3,2)) = 0 OR VAL(MID$(QQ$,3,2)) > 12
         THEN GOTO 28850
28650 IF (VAL(MID$(QQ$,1,2)) MOD 4) = 0 THEN NDM%(2) = 29
         ELSE NDM%(2) =28
28700 IF VAL(MID$(QQ$,5,2)) = 0 OR VAL(MID$(QQ$,5,2)) >
         NDM%(VAL(MID$(QQ$,3,2))) THEN 28850
28750 LOCATE 24,11:PRINT SPACE$(70);
28800 RETURN
28850 I=I-1
28900 LOCATE 24,11:PRINT "fecha errada";:BEEP
28950 RETURN
29000 '_____________________ INGRESAR CUENTA
29050 GOSUB 26750      '_______ INGRESAR NUMERICO
29100 IF Q$=CHR$(72) OR VO>0 THEN RETURN
29150 CDV=0:LOCATE 24,11:COLOR COLOUT1,COLOUT2:
      PRINT SPACE$(70);
29200 FOR I%=1 TO LEN(QQ$)
29250 CDV=CDV+VAL(MID$(QQ$,I%,1))
29300 NEXT I%
29350 IF 10*(INT(CDV/10))=CDV THEN RETURN
29400 LOCATE 24,11:COLOR COLOUT1,COLOUT2:
      PRINT "digito verificador errado";:BEEP
29450 I=I-1
29500 GOTO 29050
29550 '_____________________ END OF INPUT   (XMIT)
29600 EOI=1
29650 RETURN 14450
29700 '_____________________ LIMPIAR PANTALLA
29750 FOR I% =1 TO 6:KEY(I%) OFF:NEXT I%
29800 COLOR COLINP1,COLINP2
29850 FOR I%=1 TO IN%-1
29900 LOCATE ASC(MID$(IN$(I%),3,1)),ASC(MID$(IN$(I%),4,1))
29950 IN$(I%)=LEFT$(IN$(I%),5)+"0"
            +SPACE$(ASC(MID$(IN$(I%),2,1)))
30000 PRINT MID$(IN$(I%),7);
30050 NEXT I%
30100 EOI=1:I=1:VO=1:Q$="":KEY(1) OFF
30150 COLOR COLOUT1,COLOUT2
30200 FOR I%=1 TO LEN(OC$) STEP 3
30250 LOCATE ASC(MID$(OC$,1+I%,1)),ASC(MID$(OC$,2+I%,1))
```

```
30300 PRINT SPACE$(ASC(MID$(OC$,I%,1)));
30350 NEXT I%
30400 RETURN
30450 '_____________________ BORRAR PANTALLA
30500 FOR I% =1 TO 6:KEY(I%) OFF:NEXT I%
30550 V0=9:EOI=1
30600 GOSUB 3850
30650 RETURN 14450
30700 '_____________________ IMPRIMIR FORMATO
30750 GOSUB 9950
30800 RETURN 14450
30850 '_____________________ ON ERROR INIT
30900 PRINT "SYSTEM ERROR" ERR ERL
30950 FOR I=1 TO 15000:NEXT I
31000 RESET
31050 RETURN 50
31100 '_____________________ error en impresora
31150 COLOR COLPAN1,COLPAN2
31200 IF ERR<>24 AND ERR<>25 THEN LOCATE 24,1:
        PRINT "error en la impresora";:I$="X":RETURN
31250 LOCATE 23,1:
        PRINT "la impresora no esta lista            ";
31300 LOCATE 23,40:PRINT "reintento (r),cancel?(x)   *";
31350 LOCATE 23,66:I$=INKEY$
31400 IF LEN(I$)=0 THEN 31350
31450 LOCATE 23,1:PRINT SPACE$(80);
31500 IF I$="x" OR I$="X" THEN RETURN
31550 IF I$="r" OR I$="R" THEN RETURN ELSE GOTO 31250
31600 '_____________________ ERROR AL IMPRIMIR EL FORMATO
31650 GOSUB 31150
31700 IF I$="X" OR I$="x" THEN RESUME 10700 ELSE RESUME
```

Las rutinas que permiten transaferir el mensaje al driver de comunicaciones, y recibir los mensajes de comunicación desde el driver, se incluyen a continuación.

```
;_____________________________________________________________
;   ESTE PROGRAMA FORMA PARTE DE LA TESIS DE GRADO DE
;   ALBERTO ANDRADE VAREA.
;   ESTE PROGRAMA ESTA FORMADO POR DOS RUTINAS: LA
;   PRIMERA (STRTOM) PERMITE MOVER DATOS DESDE UN STRING DE
;   BASIC  A UNA LOCALIZACION DE  MEMORIA  ESTABLECIDA.  LA
;   SEGUNDA (MTOSTR) PERMITE REALIZAR LO CONTRARIO.
;   ESTAS  RUTINAS  SON LLAMADAS POR EL PROGRAMA  PRINCIPAL
;   PARA  COMUNICARSE CON EL DRIVER DE  COMUNICACION,  PARA
;   ENVIARLE  DATOS Y PARA RECIBIR DATOS
;_____________________________________________________________
;
;
CODE      SEGMENT  BYTE   PUBLIC 'CODE'
          ASSUME   CS:CODE
;_____________________________________________________________
;         MOVER EL STRING A$ DE BASIC A MEMORIA
;_____________________________________________________________
          PUBLIC  STRTOM        ; NOMBRE DE LA RUTINA
STRTOM    PROC  FAR             ;
          PUSH  BP              ; GUARDAR BP EN STACK
          MOV   BP,SP           ; CARGAR EN BP EL APUNTADOR A LA
                                ; BASE DEL STACK
          MOV   BX,[BP]+6       ; DIRECCION DEL STRING DESCRIPTOR
          MOV   BX,[BX]+2       ; BX  APUNTA  AL  PRIMER
                                ; CARACTER DEL STRING
          PUSH  ES              ; GUARDAR ES EN STACK
          PUSH  DI              ; GUARDAR DI EN STACK
          MOV   AX,0
          MOV   ES,AX
          PUSH  SI
          MOV   SI,32H
          MOV   AX,ES:[SI]
          POP   SI
          MOV   ES,AX           ; ES = 3100H
          MOV   DI,50H          ; OFFSET DENTRO DEL SEGMENTO
X1:       MOV   AL,DS:[BX]      ; LAZO PARA MOVER EL STRING.
          MOV   ES:[DI],AL      ; EL LAZO TERMINA  CUANDO  SE
                                ; ENCUENTRA
          INC    BX             ; EL CARACTER FFH  O  SE  HAN
                                ; MOVIDO 512
          INC   DI              ; CARACTERES.
          CMP   DI,250H         ; ES 512?
          JE    X2              ; EXIT SI 512
```

```
        CMP   AL,0FFH       ; ES FFH?
        JNE   X1            ; CONTINUE EN LAZO
X2:     POP   DI            ; RECUPERE DI
        POP   ES            ; RECUPERE ES
        POP   BP            ; RECUPERE BP
        RET   2             ; REGRESE A BASIC
STRTOM  ENDP
;_____________________________________________________________________
        PUBLIC MTOSTR
MTOSTR  PROC  FAR
        PUSH  BP            ; GUARDAR BP EN STACK
        MOV   BP,SP         ; CARGAR EN BP EL APUNTADOR
                            ; A LA BASE DEL
                            ; STACK
        MOV    BX,[BP]+6    ; DIRECCION  DEL  "STRING
                            ; DESCRIPTOR"
        MOV    BX,[BX]+2    ; BX  APUNTA  AL  PRIMER
                            ; CARACTER DEL STRING
        PUSH  ES            ; GUARDAR ES EN STACK
        PUSH  DI            ; GUARDAR DI EN STACK
        MOV   AX,0
        MOV   ES,AX
        PUSH  SI
        MOV   SI,32H
        MOV   AX,ES:[SI]
        POP   SI
        MOV   ES,AX
        MOV   DI,50H        ; OFFSET DENTRO DEL SEGMENTO
                            ; EN DONDE
                            ; EMPIEZA EL BUFFER
Y1:     MOV   AL,ES:[DI]    ; LAZO PARA MOVER LOS DATOS
        MOV    DS:[BX],AL   ; EL LAZO TERMINA  CUANDO  SE
                            ; ENCUSTRE
        INC    BX           ; EL ETX O SE HAN MOVIDO  512
                            ; CARACTERES
        INC   DI
        CMP   DI,250H       ; SON 512 CARACTERES?
        JE    Y2            ; EXIT
        CMP   AL,03H        ; ES ETX
        JNE   Y1            ; CONTINUE EL LAZO
Y2:     POP   DI            ; RECUPERE DI
        POP   ES            ; RECUPERE ES
        POP   BP            ; RECUPERE BP
        RET   2             ; REGRESE A BASIC
MTOSTR  ENDP
CODE    ENDS
        END
```

4.4 RUTINAS DE COMUNICACIONES

A continuación se incluye un listado del driver de comunicaciones.

```
;_____________________________________________________________
; ESTE PROGRAMA FORMA PARTE DE LA TESIS DE GRADO DE
; ALBERTO ANDRADE VAREA.
; LA FUNCION DE ESTE PROGRAMA ES LA DE ACTUAR COMO UN DRIVER
; DE  COMUNICACIONES DE PROTOCOLO NCR ISO ASYNCRONICO  ESTA
; RUTINA SE EJECUTA AUTOMATICAMENTE CUANDO LLEGA UN CARACTER
; DESDE EL COMPUTADOR CENTRAL O CUANDO UN CARACTER  TERMINA
; DE SER ENVIADO POR EL DRIVER.
; LA  EJECUCION  DE  ESTA  RUTINA  SIEMPRE  EMPIEZA  EN  LA
; REFERENCIA INICIO
;_____________________________________________________________
; DEFINICION DE CARACTERES DE CONTROL NECESARIOS. SE DEFINEN
; COMO CONSTANTES PARA CLARIDAD EN EL PROGRAMA
;_____________________________________________________________
EOT         EQU     04H             ; EOT
ACK         EQU     06H             ; ACK
NACK        EQU     15H             ; NACK
STX         EQU     02H             ; STX
ETX         EQU     03H             ; ETX
ENQ         EQU     05H             ; ENQ
;_____________________________________________________________
;  DEFINICION DE PUERTOS DEL ADAPTADOR RS-232-C
;  NOTA:  ESTAS  DIRECCIONES DEBEN SER  MODIFICADAS  PARA
;         UTILIZAR EL SEGUNDO ADAPTADOR (COMM2)
;_____________________________________________________________
PRECTRM     EQU     3F8H            ; RECIBE/TRANSMITE
PLINST      EQU     3FDH            ;  LINE  STATUS
PINTENA     EQU     3F9H            ; INTERRUPT ENABLE
PMODCON     EQU     3FCH            ; CONTROL DE MODEM
PMODST      EQU     3FEH            ; STATUS DEL MODEM
;
CODI        SEGMENT
    ASSUME CS:CODI,DS:CODI
;_____________________________________________________________
; CAMPOS VARIABLES
;_____________________________________________________________
    ORG OH
ENVMSG      DB 1                ; ES UNA BANDERA MANEJADA POR EL
                                ; PROGRAMA PRINCIPAL
                                ; TIENE UN VALOR DE 1 CUANDO NO
                                ; EXISTE UN MENSAJE
                                ; PARA SER ENVIADO Y UN VALOR DE
                                ; 2 HAY UN MENSAJE
                                ; PARA SER ENVIADO AL PROCESADOR
                                ; CENTRAL. ESTA BANDERA LE
                                ; INDICA AL DRIVER SI DEBE
                                ; RESPONDER  CON EOT O  CON  EL
                                ; MENSAJE A UNA SECUENCIA DE
```

```
                                ; INTERROGACION
;
RCVFLG      DB  1               ; ES  TAMBIEN  UNA  BANDERA
                                ; CONTROLADA POR EL PROGRAMA
                                ; PRINCIPAL. ESTA BANDERA SIRVE
                                ; PARA INDICAR AL DRIVER
                                ; SI EL PROGRAMA PRINCIPAL PUEDE
                                ; O NO RECIBIR UN
                                ; MENSAJE.  TIENE UN VALOR DE 1
                                ; SI EL PROGRAMA PRINCIPAL ESTA
                                ; OCUPADO Y NO PUEDE RECIBIR, EL
                                ; DRIVER  ENVIARA UN NACK  COMO
                                ; RESPUESTA A LA SELECCION.
                                ; TIENE  UN  VALOR DE 2  SI  EL
                                ; PROGRAMA PRINCIPAL ESTA DISPUEST
                                ; A RECIBIR EL MENSAJE,
                                ; EL DRIVER ACEPTARA LA SELECCION.
;
SELCOD      DB 51h              ; CODIGO DE SELECCION. EL DRIVER
                                ; ASUME QUE TA1=TA2=SEL1
                                ; DEBIDO  A  QUE EL  DRIVER  ES
                                ; DIRECTAMENTE DIRECCIONABLE
                                ; EL  CODIGO DE SELECCION ES DE
                                ; UN SOLO CARACTER. EL CODIGO DE
                                ; SELECCION PUEDE SER MODIFICADO
                                ; POR EL
SELCOD2 DB 0                    ; SEGUNDO CARACTER DE SELECCION
                                ; (NO UTILIZADO)
;
POLLCOD     DB 41h              ; CODIGO DE INTERROGACION. ESTE
                                ; CODIGO PUEDE SER
                                ; MODIFICADO  POR  EL  PROGRAMA
                                ; PRINCIPAL
;
COMST       DB  0               ; STATUS DE LA  COMUNICACION.
                                ; ESTA BANDERA INDICA
                                ; AL PROGRAMA PRINCIPAL EL
                                ; STATUS DEL ENVIO Y RECEP-
                                ; CION  DE  UN  MENSAJE.  PUEDE
                                ; TENER LOS SIGUIENTES
                                ; VALORES:  05H  = SE ENVIO  EL
                                ; MENSAJE, SE ESPERA ACK
                                ; 06H = SE HA INTENTADO ENVIAR
                                ;       3 VECES
                                ; 07H = NO LLEGO ACK NI NACK
                                ;       DESDE CENTRAL
                                ; 10H = SE ENVIO EL MENSAJE Y SE
                                ;       ESPERA SELECCION Y
                                ;       RESPUESTA
                                ; 11H = EL DRIVER YA FUE
                                ;       SELECCIONADO, SE
                                ;       ESPERA EL MENSAJE
                                ; 12H = NO SE PUDO RECIBIR
                                ;       EL MENSAJE
                                ; 20H = LA COMUNICACION HA
```

```
                                ;        SIDO CORRECTA
;
DRIVST      DB  1h              ;  ESTE CARACTER  CONTIENE  EL
                                ;  ESTADO DEL DRIVER.
                                ;  DEBIDO A QUE LA EJECUCION DEL
                                ;  DRIVER ES INTERMITENTE
                                ;  YA QUE ESTA GUIADA POR LAS
                                ;  INTERRUPCIONES, ESTE
                                ;  CARACTER  ES  UTILIZADO  COMO
                                ;  MECANISMO DE COMUNICACION
                                ;  ENTRE  2 EJECUCIONES CONSECUTIVA
                                ;  DEL DRIVER. EN ESTE CARACTER
                                ;  SE ALMACENA MEDIANTE UN
                                ;  CODIGO,  LA FUNCION QUE DEBE
                                ;  REALIZAR EL DRIVER LA
                                ;  SIGUIENTE VEZ QUE SE EJECUTE.
                                ;  PUEDE  TENER  LOS  SIGUIENTES
                                ;  VALORES Y SIGNIFICADOS:
                                ;  01H = DEBE ESPERAR EOT
                                ;  02H = DEBE ESPERAR POLLCODE O
                                ;        SELECT CODE
                                ;  03H = DEBE ESPERAR ENQ
                                ;        (EOT_POLL_ENQ)
                                ;  04H = SEBE ESPERAR ACK O NACK
                                ;        A MENSAJE ENVIADO
                                ;  05H = DEBE ENVIAR EL SIGUIENTE
                                ;        CARACTER DEL MENSAJE
                                ;  06H = DEBE ESPERAR ENQ
                                ;        (EOT_SELECT_ENQ)
                                ;  07H  = ESPERA STX DE MENSAJE
                                ;         DE CENTRAL
                                ;  08H  = ESPERA CARACTERES  DEL
                                ;         MENSAJE DE CENTRAL
                                ;  09H = ESPERA BCC DEL MENSAJE
                                ;        DE CENTRAL
;
BCC         DB 0                ;  CAMPO EN QUE SE CALCULA EL BCC
                                ;  (BLOCK CHECK CHARACTER)
                                ; PARA SER ENVIADO O VERIFICADO
;_____________________________________________________________
; LAS SIGUIENTES INSTRUCCIONES CONSTITUYEN UNA TABLA QUE
; CONTIENE LA DIRECCION DE ENTRADAS A RUTINAS ESPECIFICAS
; DEL PROGRAMA QUE PROCESAN CADA UNO DE LOS ESTADOS DEL
; DRIVER.
;_____________________________________________________________
JEST1       DW OFFSET EST1      ; RUTINA QUE ESPERA EOT
JEST2       DW OFFSET EST2      ; RUTINA QUE  ESPERA  POLL  O
                                ; SELECT
JEST3       DW OFFSET EST3      ; RUTINA QUE ESPERA ENQ DEL POLL
JEST4       DW OFFSET EST4      ; RUTINA ESPERA ACK O NACK  DE
                                ; CENTRAL
JEST5       DW OFFSET EST5      ; RUTINA QUE DEBE ENVIAR  EL
                                ; SIGUIENTE CARACTER A CENTRAL
JEST6       DW OFFSET EST6      ; RUTINA QUE ESPERA ENQ DE SELECT
JEST7       DW OFFSET EST7      ; RUTINA QUE ESPERA STX
```

```
JEST8        DW OFFSET EST8  ; RUTINA QUE ESPERA  CARACTERES
                             ; DEL MENSAJE DE CENTRAL
JEST9        DW OFFSET EST9  ; RUTINA QUE ESPERA  BCC  DE
                             ; CENTRAL
;____________________________________________________________
; OTRAS VARIABLES
;____________________________________________________________
ESTDI        DW  0           ; PARA GUARDAR APUNTADOR A BU
NACKT        DB  0           ; CONTADOR  DE   NACK   EN
                             ; TRANSMISION
        ORG 300H
;INICIO DEL PROGRAMA
;____________________
; ALMACENAMIENTO EN STACK DE TODOS LOS REGISTROS QUE SE
; UTILIZAN
INICIO:       PUSH AX
              PUSH DX
              PUSH DS
              PUSH DI
;____________________________________________________________
;  CARGA  DEL REGISTRO DS CON LA DIRRECCION INICIAL DEL AREA
;  DE DATOS
;____________________________________________________________
              MOV  AX,CS      ;  SEGMENTO  EN QUE SE DEBE
                              ;  CARGAR  EL PROGRAMA
              MOV DS,AX
;____________________________________________________________
;CALCULO DE LA DIRECCION DE LA RUTINA QUE DEBE EJECUTARSE
;____________________________________________________________
INICIA:       MOV AL,DRIVST   ; STATUS DEL DRIVER
              MOV AH,0
              MOV DI,6H       ; LA TABLA DE OFFSETS ESTA EN OFFSE
              ROL AX,1        ; AX = 2*DRIVST
              ADD  DI,AX      ; DI  = 8,10,12,..ETC  SI
                              ; DRIVST  = 1,2,3,..ETC
;____________________________________________________________
; LECTURA DEL CARACTER QUE PRODUJO LA INTERRUPCION
;____________________________________________________________
              MOV DX,PRECTRM  ; PUERTO DE RECEPCION
                              ; /TRANSMISION (3F8/2F8)
              IN AL,DX        ; LEER PUERTO
              JMP  [DI]       ;  JMP  AL CONTENIDO  DE  DI
                              ;  (RUTINA ESPECIFICA)
;____________________________________________________________
;ESTADO 1:   SE DEBE ESPERAR EOT.  SI EL CARACTER RECIBIDO
;            ES EOT
;            SE  PASA AL ESTADO 2,  CASO CONTRARIO EL  DRIVER
;            DEBE
;            PERMANECER EN ESTE ESTADO.
;____________________________________________________________
EST1:         CMP AL,EOT      ; COMPARA CON EOT. AL CONTIENE
                              ; EL CARACTER LEIDO
              JE  EST11       ; SALTE SI EOT
              JMP REGRE       ; SI <> REGRESE
EST11:        MOV DRIVST,02H  ; CAMBIE A ESTADO 2
```

```
                JMP REGRE
;_____________________________________________________________
;ESTADO 2:  DEBE ESPERAR POLL O SELECT CODES. SI POLL, CAMBIA
;           A ESTADO 3, SI SELECT CAMBIA A ESTADO 6
;_____________________________________________________________
EST2:           CMP AL,POLLCOD ; COMPARE CON POLL CODE
                JNE COMPSEL    ; SI <> COMPARAR CON SELECT
                MOV DRIVST,03H ; CAMBIE A ESTADO 3
                JMP REGRE      ; REGRESE
COMPSEL:        CMP AL,SELCOD  ; COMPARE CON SELECT CODE
                JNE CHAEST1    ;  ERROR,  CAMBIE A ESTADO 1
                               ;  (PUEDE  SER POLL DE OTRO
                               ; TERMINAL
                MOV DRIVST,06H ; CAMBIE A ESTADO 6
                JMP REGRE
CHAEST1:        MOV DRIVST,01H
                JMP REGRE
;_____________________________________________________________
;ESTADO 3:  SI ENQ DE POLL,  DEBE TRASMITIR EOT O MENSAJE
;_____________________________________________________________
EST3:           CMP AL,ENQ
                JNE CHAEST1    ; ERROR, REGRESE A ESTADO 1
                CMP ENVMSG,02H ;  HAY MENSAJE QUE TRANSMITIR?
                               ;  1=NO HAY, 2=SI HAY
                JNE SNDEOT     ; NO HAY: TRANSMITA EOT
;_____________________________________________________________
;SI HAY MENSAJE QUE TRASMITIR. SE DEBE HACER LO SIGUIENTE:
;    -INICIALIZA   CONTADOR  DE  REINTENTOS  DE  TRANSMISION
;     (NACKT),
;  - HABILITA INTERRUPCION POR CARACTER TRASMITIDO,
;  - SE LEVANTA PORTADORA
;  - SE  INICIALIZA EL APUNTADOR AL BUFFER  DE  TRANSMISION
;    (DI).
;_____________________________________________________________
                MOV NACKT,0H   ; CONTADOR DE REINTENTOS
TRASM1:         MOV  AL,2H     ; CODIGO DE INTERRUPCION  POR
                               ; TRANSMISION
                MOV DX,PINTENA ; PUERTO PARA HABILITAR INTERRUPCI
                OUT DX,AL      ; HABILITAR INTERRUPCION POR
                               ; TRANSMISION
                CALL SUBIR     ; LEVANTAR PORTADORA PARA TRANSMIT
                MOV DI,50H     ; OFFSET DE BUFFER DE DATOS
                MOV  AL,[DI]   ; AL CONTIENE SIGUIENTE CARACTER
                               ; DEL BUFFER
                MOV  DX,PRECTRM;  PUERTO DE RECEPCION  Y
                               ;  TRANSMISION (3F8)
                OUT DX,AL      ; ENVIO DEL CARACTER
                MOV ESTDI,DI   ; GUARDAR APUNTADOR AL BUFFER
                MOV BCC,0H     ; ENCERAR BCC
                MOV DRIVST,05H ; CAMBIAR A ESTADO 5
                JMP REGRE      ; REGRESO.  EL DRIVER SE EJECUTARA
                               ; OTRA VEZ
                               ; CUANDO SE TERMINE LA TRANSMISION
                               ; DEL CARACTER
;_____________________________________________________________
```

```
;ESTADO 5: TRANSMISION DE CARACTERES POSTERIORES A STX. ESTA
RUTINA
;  CALCULA EL BCC HASTA ENCONTRAR FFH, QUE ES PUESTO EN EL
;  BUFFER COMO BANDERA QUE INDICA EL FIN DEL MENSAJE.
;_____________________________________________________________
EST5:          MOV  DX,PRECTRM      ;  PUERTO  DE  TRANSMISION
                                    ;  Y RECEPCION
               MOV DI,ESTDI    ; POINTER AL BUFFER
               INC DI          ; INCREMENTAR POINTER
               MOV AL,[DI]     ; AL CONTIENE AL SIGUIENTE CARACTER
               CMP AL,0FFH     ; VER SI ES EL ULTIMO
               JE MANBCC       ; SI = , ENVIAR BCC
               OUT DX,AL       ; ENVIAR CARACTER
               XOR BCC,AL      ; CACULAR BCC
               MOV ESTDI,DI    ; ALMACENAR APUNTADOR AL BUFFER
               JMP REGRE       ; REGRESO .
;_____________________________________________________________
;  - ENVIO BCC
;  - DESHABILITACION INTERRUPCION DE ENVIO DE CARACTERES
;  - CAMBIO A ESTADO 4 BAJAR PORTADORA
;_____________________________________________________________
MANBCC:        MOV AL,BCC      ; CARGAR BCC CALCULADO
               OUT DX,AL       ; ENVIAR BCC
               MOV DX,PINTENA  ; PUERTO PARA HABILITAR INTERRUPCIC
               MOV AL,1H       ; HABILITAR  INTERRUPCION  SOLO
                               ; POR RECEPCION
               OUT DX,AL       ; Y DESHABILITAR POR TRANSMISION
               MOV DRIVST,04H  ; CAMBIO A ESTADO 4, ESPERAR ACK /N
               MOV COMST,05H   ; ESTATUS COM = 5, MENSAJE ENVIADO
               CALL BAJAR      ; BAJAR PORTADORA
               JMP REGRE       ; REGRESO
;_____________________________________________________________
; ENVIO DE EOT LUEGO DE POLL Y LUEGO DE RECIBIR UN ACK
;_____________________________________________________________
SNDEOT:        CALL SUBIR      ; SUBIR PORTADORA
               MOV DX,PRECTRM  ; PUERTO DE TRANSMISION /RECEPCION
               MOV AL,EOT      ; CARGAR EOTC
               OUT DX,AL       ; ENVIAR EOT
               CMP DRIVST,04H  ; COMPARE ESTADO CON 4 (ESPERA ACK)
               JNE SEOT        ; SALTE SI NO ES 4
               MOV COMST,10    ; COMST = 10, ESPERA SELECCION
SEOT:          MOV DRIVST,01H  ; REGRESE A ESTADO 1. ESPERE EOT
               CALL BAJAR      ; BAJAR PORTADORA
               JMP REGRE       ; REGRESE
;_____________________________________________________________
; ESTADO 4: ESPERA RECIBIR ACKO NACK A MENSAJE ENVIADO.
; - SI ES NACK INCREMENTA EL NUMERO DE REINTENTOS (NACKT) Y
;    SALTA  A  TRASMITIR  NUEVAMENTE  EL  MENSAJE.  ENVMSG=1
; - SI EL CARACTER RECIBIDO NO ES NACK NI ACK REPORTA STATUS
;   07 Y REGRESA A ESTADO 1
;_____________________________________________________________
EST4:          MOV  ENVMSG,01H ;  ENVMSG=1 ==> NO HAY MENSAJE
                               ;  QUE TRANSMITIR
               CMP AL,ACK      ; ES ACK ?
               JE SNDEOT       ; ENVIE EOT
```

```
                CMP AL,NACK     ; ES NACK ?
                JE TRASM11      ; REINTENTE
                MOV COMST,07H   ; ERROR 7, NI ACK NI NACK RECIBIDO
                MOV DRIVST,01H  ; ESPERE EOT
                JMP REGRE       ; REGRESE
TRASM11:        ADD NACKT,1     ; REINTENTOS. INCREMENTE REINTENTOS
                CMP NACKT,3     ; TERCERA VEZ?
                JE TRASM12      ; SI, ABORTE
                JMP TRASM1      ; NO, REINTENTE
TRASM12:        MOV COMST,06H   ; TERMINE LA COMUNICACION.
                                ; ERROR 06H
                MOV DRIVST,01H  ; ESPERE EOT
                JMP REGRE       ; REGRESE
;_____________________________________________________________
;ESTADO 6: ESPERA ENQ DE SELECCION
;_____________________________________________________________
EST6:           CMP AL,ENQ      ; ES ENQ ?
                JE  EST61       ; OK .
                MOV DRIVST,01H  ; ESPERE EOT
                JMP REGRE       ; REGRESE
EST61:          CMP RCVFLG,02H  ; SE PUEDE RECIBIR? RCVFLG = 2
                JE ASELE        ; SI SE PUEDE
                CALL SUBIR      ; SUBIR PORTADORA PARA ENVIAR NACK
                MOV DX,PRECTRM  ; PUERTO DE TRANSMISION Y RECEPCION
                MOV AL,NACK     ; MOVER NACK
                OUT DX,AL       ; ENVIAR NACK
                MOV DRIVST,01H  ; ESPERAR EOT
                CALL BAJAR      ; BAJAR PORTADORA
                JMP REGRE       ; REGRESAR
;_____________________________________________________________
; SI SE PUEDE RECIBIR. SE DEBE ENVIAR TA1_TA2_ACK
;_____________________________________________________________
ASELE:          CALL SUBIR      ; SUBIR PORTADORA
                MOV AH,0H       ; CONTADOR
ASELE1:         MOV DX,PRECTRM  ; PUERTO DE ENVIO/RECEPCION
                MOV AL,SELCOD   ; TA1 Y TA2 SON IGUALES A SELECT CO
                OUT DX,AL       ; ENVIAR TA1
                MOV DX,PLINST   ; CHEQUEAR EL ESTATUS DE LA LINEA
CHSEL:          IN AL,DX        ; LEER STATUS
                AND AL,20H      ; VER SI SE PUEDE PONER OTRO
                                ; CARACTER EN EL
                CMP AL,20H      ; REGISTRO DE TRANSMISION
                JNE CHSEL       ; NO SE PUEDE, LOOP
                INC AH          ; INCREMENTAR CONTADOR
                CMP AH,2H       ; SI CONTADOR ES 2 SE DEBE ENVIAR A
                JNE ASELE1      ; ENVIAR TA2
                MOV DX,PRECTRM  ; PUERTO DE ENVIO/RECEPCION
                MOV AL,ACK      ; CARGAR ACK
                OUT DX,AL       ; ENVIAR ACK
                MOV DRIVST,07H  ; DRIVER STATUS 7 = ESPERA MENSAJE
                                ; (STX)
                MOV COMST,11    ; STATUS DE COMUNICACION 11.
                CALL BAJAR      ; BAJAR PORTADORA
                JMP REGRE       ; REGRESE
;_____________________________________________________________
```

```
;ESTADO 7: ESPERA RECIBIR STX
;_____________________________________________________________
EST7:          CMP AL,STX      ; ES STX
               JE RECIB        ; SI, ALMACENAR EN BUFFER
               CMP AL,EOT      ; ES EOT
               JNE EST71       ; NO, REGRESE
               MOV DRIVST,02H  ; ESPERE POLL O SELECT CODE
EST71:         JMP REGRE
;_____________________________________________________________
; - RECIBIO STX: INICIALIZA BCC, INICIALIZA PUNTERO A BUFFER
;   (DI)
;_____________________________________________________________
RECIB:         MOV DI,50H      ; APUNTADOR A BUFFER
               MOV [DI],AL     ; MOVE CARACTER RECIBIDO A BUFFER
               MOV DRIVST,08H  ; CAMBIE A ESTADO 8
               MOV ESTDI,DI    ; ALMACENE APUNTADOR
               MOV BCC,0H      ; INICIALIZAR BCC
               JMP REGRE       ; REGRESE
;_____________________________________________________________
;ESTADO 8: RECIBE CARACTERES POSTERIORES A STX HASTA ETX Y
;          CALCULA BCC
;_____________________________________________________________
EST8:          MOV DI,ESTDI    ; RECUPERA APUNTADOR
               INC DI          ; INCREMENTA APUNTADOR
               MOV [DI],AL     ; MUEVE CARACTER RECIBIDO
               XOR BCC,AL      ; CALCULA BCC
               CMP AL,ETX      ; ES ETX
               JE EST9         ; SI, ESPERE BCC
               MOV ESTDI,DI    ; NO, ALMACENE APUNTADOR
               JMP REGRE       ; REGRESE
;_____________________________________________________________
; ESTADO 9: ESPERA BCC SI ES CORRECTO MANDA ACK RCVFLG=1
;_____________________________________________________________
EST9:          MOV DX,PLINST    ; VER STATUS DE LA LINEA
VIEBC:         IN AL,DX        ; LEER STATUS
               AND AL,01H      ; VER SI LLEGO UN CARACTER
               CMP AL,01H
               JNE VIEBC       ; NO LLEGO
               MOV DX,PRECTRM  ; SI LLEGO, HAY QUE LEER
               IN AL,DX        ; LEER CARACTER
               CMP AL,BCC      ; ES IGUAL AL BCC CALCULADO
               JNE MALME       ; NO, ENVIAR NACK
               CALL SUBIR      ; SI, ENVIAR ACK
               MOV AL,ACK      ; CARGAR ACK
               MOV DX,PRECTRM  ; PUERTO PARA TRANSMISION
               OUT DX,AL       ; ENVIAR ACK
               MOV DRIVST,01H  ; ESPERAR EOT
               MOV COMST,20H   ; COMST = 20: BUEN STATUS
               MOV RCVFLG,01H  ; RCVFLG= 1, NO SE PUEDE RECIBIR OT
                               ; MENSAJE HASTA QUE EL PROGRAMA
                               ; PRINCIPAL CAMBIE A 2
               CALL BAJAR      ; BAJAR LA PORTADORA
               JMP REGRE       ; REGRESAR
;_____________________________________________________________
;EL MENSAJE RECIBIDO ES MALO SE ENVIA NACK
```

```
;
MALME:          CALL SUBIR      ; SUBIR PORTADORA
                MOV AL,NACK     ; CARGAR NACK
                MOV DX,PRECTRM  ; PUERTO DE TRANSMISION
                OUT DX,AL       ; ENVIAR NACK
                MOV  DRIVST,07H; ESPERE OTRA VEZ EL MENSAJE 0
                                ; EOT
                CALL BAJAR      ; BAJAR PORTADORA
                JMP REGRE       ; REGRESE
;
;LEVANTA PORTADORA SE CHEQUEA CTS
;
    PUBLIC SUBIR
SUBIR           PROC NEAR
                MOV DX,PMODCON  ; PUERTO DE CONTROL DE MODEM
                MOV AL,15       ; PRENDER OUT1,OUT2,RTS,DTR
                OUT DX,AL       ; OUTPUT
                MOV DX,PMODST   ; PUERTO DE STATUS DEL MODEM
SUB1:           IN AL,DX        ; LEER STATUS
                AND AL,30H      ; ESTA DSR Y CTS  ARRIBA
                CMP AL,30H
                JNE SUB1        ; NO, ESPERE
                MOV AX,02FFH    ; TIMER
SUB2:           DEC AX          ; DECREMENTE
                CMP AX,0        ; ES 0
                JNE SUB2        ; NO, LOOP
                RET             ; REGRESE DE CALL
SUBIR           ENDP
;
;SE BAJA PORTADORA
;
    PUBLIC BAJAR
BAJAR           PROC NEAR
                MOV DX,PMODCON  ; PUERTO DE CONTROL DE MODEM
                MOV AL,13       ; PRENDER OUT1,OUT2,DTR, APAGAR RTS
                OUT DX,AL       ; OUTPUT
                RET             ; REGRESE DE CALL
BAJAR           ENDP
;
; - REGRESO DE LA INTERRUPCION
;
REGRE:          POP DI          ; RECUPERAR REGISTROS
                POP DS
                POP DX
                MOV AL,64H      ; FIN DE LA INTERRUPCION
                OUT 20H,AL      ; RECONOCER INTERRUPCION
                POP AX
                IRET
FINPROG DB 0
LOADER:         MOV AX,CS
                MOV DS,AX
                MOV AH,25H
                MOV AL,0CH
                MOV DX,OFFSET INICIO
                INT 21H
```

5.1 INSTALACION Y CONEXIONES.

Las pruebas se han realizado utilizando un computador central NCR 9300 trabajando con el sistema operativo ITX-TRANPRO.

El PC se ha conectado a una línea con protocolo NCR ISO ASINCRONICO en dos modalidades diferentes: local y remota.

PRUEBAS DE CONEXION LOCAL.

Para la conexión local se ha utilizado un cable con la siguiente configuración de conectores.

TERMINAL	COMPUTADOR CENTRAL
1	1
2	3
3	2
4 Y 5	8
6	20
7	7
8	4 Y 5
20	6

Las pruebas se han realizado a las siguientes velocidades de transmisión: 1200, 1800, 2400, 4800 y 9600 bits por segundo.

PRUEBAS DE CONEXION REMOTA

La conexión remota se realizó utilizando modems asincrónicos a velocidades de transmisión de 1200 y 1800 bits por segundo. La configuración del cables requerido para la conexión del PC al modem es la siguiente:

TERMINAL	COMPUTADOR CENTRAL
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
20	20

5.2 COMPARACION CON OTRAS ALTERNATIVAS

Las siguientes son las alternativas que un PC compatible con los programas producto de esta tesis existen:

a) El terminal NCR 7900-3.

b) Un PC compatible con un emulador del terminal NCR 7900-3.

Las dos alternativas son funcionalmente similares, sin embargo, la segunda tiene la ventaja de que el equipo puede ser utilizado como un computador personal cuando no está trabajando como terminal.

Los siguientes puntos resumen las ventajas que en mi opinión tiene el PC compatible y los programas producto de esta tesis sobre el terminal NCR 7900-3 o su emulador.

TIEMPO DE PRESENTACION DE UN FORMATO DE PANTALLA.

El terminal NCR 7900-3 es un terminal de limitadas capacidades, es un terminal que no dispone de ningun medio de almacenamiento magnético, y que tiene una limitada cantidad de memoria RAM. Para su operación requiere que el computador central le envie los formatos de pantalla a través de la línea de comunicaciones en un mensaje. Este mensaje contiene información de los datos constantes del formato y también de los campos de entrada y salida. El computador central debe enviar inclusive los espacios en

blanco.

Para realizar una comparación lo más real posible, se han seleccionado parámetros considerados comunes y promedios y que son los siguientes:

- Un formato que requiere aproximadamente 800 caracteres de transmisión,
- Una velocidad de transmisión de 1800 bits por segundo,
- que existan 8 dispositivos direccionables en la línea de comunicaciones,
- que todos los dispositivos direccionables respondan apropiadamente a la secuencia de interrogación,
- que no exista actividad en los teminales de esa línea,
- que no existan errores de comunicación en el envío del mensaje, y
- que la respuesta del computador central a la solicitud del mensaje sea casi inmediata.

El tiempo mínimo requerido para la presentación del formato en el terminal NCR 7900-3 está dado por el sumatorio de lo siguiente:

- tiempo de espera hasta que el terminal sea interrogado por el computador central. Este tiempo se ha calculado en 130 milisegundos mediante las siguientes consideraciones:

- el tiempo que toma la transmisión de un carácter es de 5.55 milisegundos a una velocidad de 1800 bits por segundo.
- una secuencia de interrogación tiene 3 caracteres, por lo que se requieren 16.66 milisegundos.
- la respuesta a una secuencia de interrogación toma aproximadamente 12 milisegundos.
- El terminal debe esperar de 1 a 7 secuencias de interrogación hasta que él sea interrogado. El promedio de interrogaciones que debe esperar es de 4. En esta consideración se asume que todos los dispositivos direccionables tienen la misma prioridad de interrogación.
- ningun terminal envía o recibe un mensaje durante este tiempo.

- envío del mensaje de solicitud del formato. Este mensaje tiene 58 caracteres, por lo que 32.2 milisegundos son requeridos.
- tiempo de proceso en el computador central. Este tiempo es variable dependiendo de la actividad del sistema en el momento de la solicitud del formato, sin embargo, para una primera consideración se asume este valor en 1.5 segundos, lo cual puede considerarse excelente.
- tiempo de transmisión del mensaje. Los 800 caracteres del mensaje toman 4.45 segundos.

De las consideraciones anteriores, se desprende que el menor tiempo de presentación del formato es de aproximadamente 6.5 segundos. El peor tiempo, será cuando todos los terminales de la línea requieren enviar o recibir mensajes al mismo tiempo, y el computador central se encuentra sobrecargado y sus tiempos de respuesta son altos. Como ejemplo, consideremos que existen 8 terminales 7900-3 en la línea, y que todos solicitan el formato al mismo tiempo, consideremos también que el tiempo de respuesta es de 1.5 segundos, en estas condiciones, uno de los terminales tendría el formato en 6.5 segundos , pero el último tendrá el formato en 8 veces dicho tiempo (52 segundos).

De lo anterior se desprende que la solicitud de un formato por un terminal, no solo afecta al operador de dicho terminal, sino a todos los operadores de todos los terminales de la línea, ya que durante la transmisión del mensaje (aproximadamente 5 segundos), ningún otro terminal puede enviar o recibir mensajes. También se desprende que el tiempo de presentación de un formato es un tiempo variable y dependiente de los recursos de la línea y del computador central en un momento dado, y que la variación puede tener un rango muy grande.

La presentación de un formato de características similares en el PC con los programas producto de esta tesis toma 1.4 segundos cuando no es requerido accesar al disco

flexible, y 2.1 segundos cuando el formato no esta residente en memoria. Este tiempo es siempre constante ya que no hace uso de los recursos del computador central ni de la línea de comunicaciones. Al utilizar únicamente los recursos del PC, permite que la línea de comunicaciones pueda ser utilizada más eficientemente, y que los requerimientos del procesador central por parte de los terminales disminuyan.

VALIDACION DE CAMPOS DE ENTRADA

El terminal NCR 7900-3 no tiene ninguna capacidad de validación de la información ingresada por el operador. Para detectar un error de digitación, el operador debe enviar el mensaje y esperar un minimo de 3 segundos por la respuesta con la indicación del error dada por el computador central. En el procedimiento anterior se utilizó ovbiamente recursos de la línea de comunicaciones, del computador central y tiempo del operador del terminal. El PC y los programas de esta tesis permiten realizar una serie de validaciones utilizando la inteligencia que este "terminal" tiene, evitando asi el envio de mensajes que tienen errores simples.

MEJOR UTILIZACION DE LOS RECURSOS DE COMUNICACIONES

Puesto que terminal no requiere de la transamisión de formatos de pantalla, y que además realiza procesos de validación de los datos ingresados, se reduce el número de mensajes erróos enviados. Esto reduce también los recursos que se requieren para procesar dichos mensajes.

5.3 SIMULACION DE FALLAS

Para propósito de este trabajo, son de interés las fallas referentes a comunicaciones entre el terminal y el computador central.

Con objeto de analizar las fallas y generar las mismas, se ha utilizado el siguiente equipo:

TEKTRONIX 834: que es un analizador de líneas de comunicación y que se ha utilizado para realizar las funciones de un computador central.

INTERVIEW 3100: que es también un analizador de líneas de comunicación y que se ha utilizado como un DATASCOPE y que permitió analizar el resultado de las diferentes pruebas.

Con la ayuda del equipo indicado, se realizaron las siguientes pruebas:

NACK DEL COMPUTADOR CENTRAL A UN MENSAJE DEL TERMINAL

El 834 de TEKTRONIX realiza las funciones de interrogación del terminal, y cuando recibe un mensaje, responde con un NACK. El driver de comunicación, al recibir el NACK, envía nuevamente el mensaje.

Al recibir un NACK por tercera vez consecutiva, envía una identificación del error al programa intérprete.

El flujo de mensajes entre el computador y el terminal, se aprecia en el la figura 5.2-A1.

```
pin 2  04      02 7E 7E 24 31 31 31 41 40 30 30 30 30 71 30 30 31
       ET      SX ^ ^ $ 1 1 1 A @ 0 0 0 0 q 0 0 1
pin 3  05 04 21 05
       EQ ET ! EQ

pin 2  1F 31 43 43 43 43 43 43 43 43 31 33 32 31 32 32 31 31 31 31
       US 1 C C C C C C C C 1 3 2 1 2 2 1 1 1 1
pin 3
pin 2  31 31 1F 32 31 32 33 34 35 36 37 38 39 30 20 20 20 20 20 20
       1 1 US 2 1 2 3 4 5 6 7 8 9 0 SP SP SP SP SP SP
pin 3

pin 2  20 20 20 20 03 54 02 7E 7E 24 31 31 31 41 40 30 30 30 30 71
       SP SP SP SP EX T SX ^ ^ $ 1 1 1 A @ 0 0 0 0 q
pin 3              15
                   NK

pin 2  30 30 31 1F 31 43 43 43 43 43 43 43 43 31 33 32 31 32 32 31
       0 0 1 US 1 C C C C C C C C 1 3 2 1 2 2 1
pin 3
pin 2  31 31 31 31 31 1F 32 31 32 33 34 35 36 37 38 39 30 20 20 20
       1 1 1 1 1 US 2 1 2 3 4 5 6 7 8 9 0 SP SP SP
pin 3

pin 2  20 20 20 20 20 20 20 03 54 02 7E 7E 24 31 31 31 41 40 30 30
       SP SP SP SP SP SP SP EX T SX ^ ^ $ 1 1 1 A @ 0 0
pin 3                        15
                             NK

pin 2  30 30 71 30 30 31 1F 31 43 43 43 43 43 43 43 43 31 33 32 31
       0 0 q 0 0 1 US 1 C C C C C C C C 1 3 2 1
pin 3
```

ENVIO DE UN MENSAJE ERRADO

En esta prueba, el computador central, selecciona al terminal y envia un mensaje erròneo (paridad o BCC). El driver de comunicaciones, detecta el error y envia un NACK. Es responsabilidad del computador reintentar o dar por terminada la comunicaciòn mediante el envio de un EOT.

La figura 5.2.A2 indica el flujo de mensajes entra computador y terminal.

```
pin 2  54 04    31 31 06
       T ET     1  1  AK
pin 3  06 04 31 05    02 50 52 55 45 42 41 20 4D 41 4C 20 42 43 43
       AK ET 1  EQ    SX P  R  U  E  B  A  SP M  A  L  SP B  C  C

pin 2    15                                                   15
         NK                                                   NK
pin 3  03 21 02 50 52 55 45 42 41 20 4D 41 4C 20 42 43 43 03 21 02
       EX !  SX P  R  U  E  B  A  SP M  A  L  SP B  C  C  EX !  SX

pin 2                                                 15
                                                      NK
pin 3  50 52 55 45 42 41 20 4D 41 4C 20 42 43 43 03 21 02 50 52 55
       P  R  U  E  B  A  SP M  A  L  SP B  C  C  EX !  SX P  R  U

pin 2                                                 04       04
                                                      ET       ET
pin 3  45 42 41 20 4D 41 4C 20 42 43 43 03 21 04 21 05 04 21 05 04
       E  B  A  SP M  A  L  SP B  C  C  EX !  ET !  EQ ET !  EQ ET

pin 2    04       04       04       04
         ET       ET       ET       ET
pin 3  21 05 04 21 05 04 21 05 04 21 05 04
```

FIGURA 5.2.A2

MENSAJE DEL COMPUTADOR SIN ETX

En esta prueba, el computador envía una secuencia de selección, y luego un mensaje que no tiene un ETX. El driver recibe el mensaje y lo ignora al recibir el EOT de la siguiente secuencia de interrogación.

La secuencia de los mensajes puede apreciarse en la figura 5.2.A3

```
pin 2  04    31 31 06
       ET    1  1  AK
pin 3  04 31 05    02 50 52 55 45 42 41 20 53 49 4E 20 45 54 58 20
       ET 1  EQ    SX P  R  U  E  B  A  SP S  I  N  SP E  T  X  SP

pin 2                 04       04       04       04       04
                      ET       ET       ET       ET       ET
pin 3  20 04 21 05 04 21 05 04 21 05 04 21 05 04 21 05 04 21 05 04
       SP ET !  EQ ET !  EQ ET !  EQ ET !  EQ ET !  EQ ET !  EQ ET

pin 2    04       04
         ET       ET
pin 3  21 05 04 21 05 04 21 05
       !  EQ ET !  EQ ET !  EQ
```

FIGURA 5.2.A3

MENSAJE SIN STX

En esta prueba, el computador selecciona al terminal y envia un mensaje que no contiene STX. El driver recibe el mensaje pero lo ignora al recibir el EOT de la siguiente secuencia de interrogación.

El flujo de mensajes entre computador y terminal se presenta en la figura 5.2.A4.

```
pin 2  04    31 31 06
       ET    1  1  ACK
pin 3  04 31 05    50 52 55 45 42 41 20 53 49 4E 20 53 54 58 03 20
       ET 1 EQ    P  R  U  E  B  A  SP S  I  N  SP S  T  X  EX SP

pin 2        04       04       04       04       04       04
             ET       ET       ET       ET       ET       ET
pin 3  04 21 05 04 21 05 04 21 05 04 21 05 04 21 05 04 21 05 04 21
       ET !  EQ ET !  EQ ET !  EQ ET !  EQ ET !  EQ ET !  EQ ET !
```

FIGURA 5.2.A4

6. CONCLUSIONES Y RECOMENDACIONES

El desarrollo de este trabajo ha sido muy interesante ya que para el mismo ha sido necesario investigar en diferenes areas:

- protocolos de comunicaciones,
- microprocesadores,
- adaptadores de comunicaciones,
- lenguajes de programación,
- redes de procesamiento transaccional en teleproceso,
- sistemas operativos transaccionales,
- sistemas aplicativos transaccionales, y
- necesidades de usuarios para optimizar los recursos de los computadores centrales y de las líneas de comunicación.

Este trabajo incluye también una fase de diseño del funcionamiento de los programas, de la interrelación de los mismos, de la estructura de archivos, y de la lógica de los mismos.

Como recomendaciones, puedo sugerir las siguientes:

a) El protocolo NCR ISO ASYNCRONICO es utilizado unicamente por equipos de marca NCR. Esto limita la utilidad de este producto a redes que tengan computadores de esta marca. Por esta razón, debería analizarse la posibilidad de utilización de otros protocolos de comunicación.

b) El interfase RS-232-C no permite la conexión de terminales en cadena, lo cual es normalmente un requisito en redes multipunto. Es posible modificar el interfase para permitir dicha conexión.

c) Pese a que el lenguaje Q-BASIC tiene un rendimiento aceptable, no es el más eficiente, por lo que el utilizar lenguajes tales como "C" o PASCAL, permitirían obtener un mejor rendimiento.

d) La tendencia futura es la utilizar redes de alta velocidad entre los terminales de una localidad, agrupados en un controlador que tiene comunicaciones al computador central, por lo que el driver de este trabajo debe ser reemplazado por un driver que permita la conexión a un concentrador local.

e) Computadores personales del futuro podrán manejar otros dispositivos, tales como impresoras, lectoras de caracteres ópticos o magnéticos, lectores de banda magnética, etc, por lo que el diseño de este producto debería modificarse para soportar dichos dispositivos.

EIA STANDARD

INTERFACE BETWEEN DATA
TERMINAL EQUIPMENT AND
DATA COMMUNICATION EQUIPMENT
EMPLOYING SERIAL BINARY
DATA INTERCHANGE

RS-232-C

(Revision of RS-232-B)

1305

OCTOBER 1969

(Reaffirmed JUNE 1981)

Engineering Department

ELECTRONIC INDUSTRIES ASSOCIATION

Published by

ELECTRONIC INDUSTRIES ASSOCIATION

Engineering Department

2001 Eye Street, N. W., Washington, D. C. 20006

PRICE : $ 6.90

Printed in U.S.A.

1306

FOREWORD

This foreword is not a part of EIA Standard RS-232-C and has been added for information purposes only.

This standard is published to provide information relating to the interface characteristics of existing and new equipment designed to meet the requirements of this standard. Co-existing with this standard is EIA Standard RS-449, together with the current versions of RS-422 and RS-423. RS-232-C has long served the user community in areas where speeds up to 20 kbit/s and cable lengths up to 15 meters (50 feet) are required. RS-449 offers speeds up to 2 Mbit/s, cable lengths up to 60 meters (200 feet) and some additional functions. See the foreword of RS-449 for additional details.

Existing RS-232-C equipment may be interoperated with, equipment designed to meet RS-449. Guidance concerning interoperability may be found in Industrial Electronicas Bulletin No. 12, "Application Notes on Interconnection between Interfaces Using RS-449 and RS-232-C".

A revision of ISO International Standard IS2110-1980 specifying the 25 pin connector and associated pin assignments has now been processed. The revision introduces some pin assignment incompatibilities with RS-232-C, due to the addition of test features. RS-232-C has not been revised to follow ISO IS2110-1980. Rather EIA Technical Committees have prepared Industrial Electronics Bulletin IEB #14, "Application Notes on Loop Test Interface Circuits using RS-232-C", as a guide for the joint use of RS-232C and the test circuits -- test indicator, local loopback and loopback/maintenance test, as shown in ISO IS2110-1980.

CCITT, ISO and EIA groups are now working on a "mini" interface which is intended to serve as a common DTE/DCE interface in a wide range of applications including public data networks and public telephone networks.

INTERFACE BETWEEN DATA TERMINAL EQUIPMENT AND DATA COMMUNICATION EQUIPMENT EMPLOYING SERIAL BINARY DATA INTERCHANGE

(From EIA Standard RS-232-B and Standards Proposal No. 1012 formulated under the cognizance of EIA Subcommittee TR-30.2 on Interface.)

INDEX

1308

PAGE

Section Four – Continued

Section Five

Section Six

Section Seven

SECTION ONE

1. SCOPE

1.1 This standard is applicable to the interconnection of data terminal equipment (DTE) and data communication equipment (DCE) employing serial binary* data interchange. It defines:

Section 2 – Electrical Signal Characteristics:–

Electrical characteristics of the interchange signals and associated circuitry.

Section 3 – Interface Mechanical Characteristics:–

Definition of the mechanical characteristics of the interface between the data terminal equipment and the data communication equipment.

Section 4 – Functional Description of Interchange Circuits:–

Functional description of a set of data, timing and control interchange circuits for use at a digital interface between data terminal equipment and data communication equipment.

1310

Section 5 – Standard Interfaces for Selected Communication System Configurations:–

Standard subsets of specific interchange circuits are defined for a specific group of data communication system applications.

In addition, the standard includes:

Section 6 – Recommendations and Explanatory Notes
Section 7 – Glossary of New Terms

1.2 This standard includes thirteen specific interface configurations intended to meet the needs of fifteen defined system applications. These configurations are identified by type, using alphabetic characters A through M. In addition, type Z has been reserved for applications not covered by types A through M, and where the configuration of interchange circuits is to be specified, in each case, by the supplier.

1.3 This standard is applicable for use at data signalling rates in the range from zero to a nominal upper limit of 20,000 bits per second.

1.4 This standard is applicable for the interchange of data, timing and control signals when used in conjunction with electronic equipment, each of which has a single common return (signal ground),

*See section 6.1.

that can be interconnected at the interface point. It does not apply where electrical isolation between equipment on opposite sides of the interface point is required.

1.5 This standard applies to both synchronous and nonsynchronous serial binary data communication systems.

1.6 This standard applies to all classes of data communication service, including:

1.6.1 Dedicated leased or private line service, either two wire or four wire. Consideration is given to both point-to-point and multipoint operation.

1.6.2 Switched network service, either two-wire or four-wire. Consideration is given to automatic answering of calls; however, this standard does not include all of the interchange circuits required for automatically originating a connection. (See EIA Standard RS-366 "Interface Between Data Terminal Equipment and Automatic Calling Equipment for Data Communication".)

1.7 The data set may include transmitting and receiving signal converters as well as control functions. Other functions, such as pulse regeneration, error control, etc., may or may not be provided. Equipment to provide these additional functions may be included in the data terminal equipment or in the data communication equipment, or it may be implemented as a separate unit interposed between the two.

1.7.1 When such additional functions are provided within the data terminal equipment or the data communication equipment, this interface standard shall apply only to the interchange circuits between the two classes of equipment.

1.7.2 When additional functions are provided in a separate unit inserted between the data terminal equipment and the data communication equipment, this standard shall apply to both sides (the interface with the data terminal equipment and the interface with the data communication equipment — See Section 3.1.1) of such separate unit.

1.8 This standard applies to all of the modes of operation afforded under the system configurations indicated in Section 5, Standard Interfaces for Selected Communication System Configurations.

SECTION TWO

2. ELECTRICAL SIGNAL CHARACTERISTICS

2.1 Figure 2.1, Interchange Equivalent Circuit, shows the electrical parameters which are specified in the subsequent paragraphs of this section. The equivalent circuit shown in Figure 2.1 is applicable to all interchange circuits regardless of the category (data, timing, or control) to which they belong. The equivalent circuit is independent of whether the driver is located in the data communication [illegible]

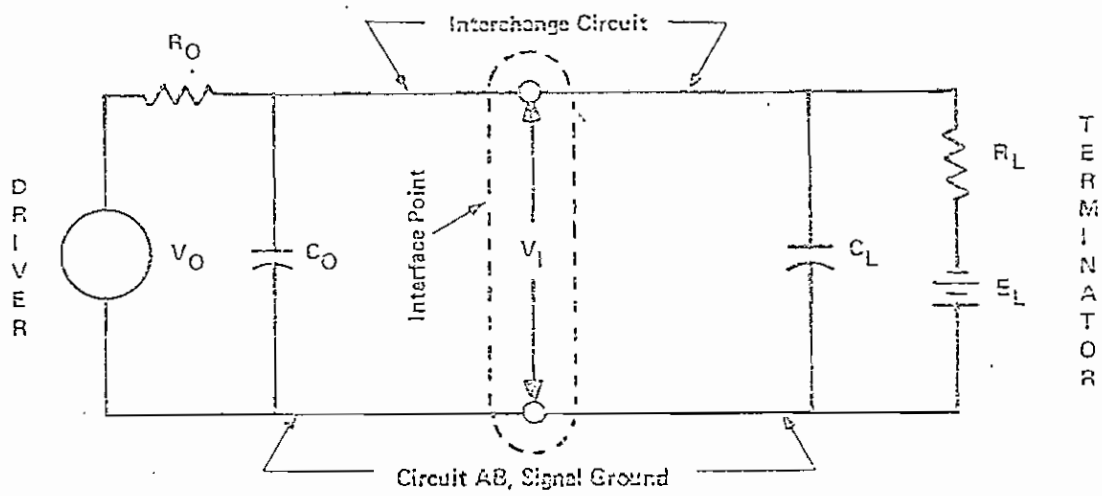


FIGURE 2.1 - INTERCHANGE EQUIVALENT CIRCUIT

V_O is the open-circuit driver voltage.

R_O is the driver internal dc resistance.

C_O is the total effective capacitance associated with the driver, measured at the interface point and including any cable to the interface point.

1312

V_I is the voltage at the interface point.

C_L is the total effective capacitance associated with the terminator, measured at the interface point and including any cable to the interface point.

R_L is the terminator load dc resistance.

E_L is the open circuit terminator voltage (bias).

2.2 The driver on an interchange circuit shall be designed to withstand an open circuit, a short circuit between the conductor carrying that interchange circuit in the interconnecting cable and any other conductor in that cable, or any passive non-inductive load connected between that interchange circuit and any other interchange circuit including Circuit AB (Signal Ground), without sustaining damage to itself or its associated equipment. The terminator on an interchange circuit shall be designed to withstand any input signal within the 25 volt limit specified in section 2.6. (see Section 6.6).

2.3 For data interchange circuits, the signal shall be considered in the marking condition when the voltage (V_I) on the interchange circuit, measured at the interface point, is more negative than minus three volts with respect to Circuit AB (Signal Ground). The signal shall be considered in the spacing condition when the voltage (V_I) is more positive than plus three volts with respect to circuit AB (see 6.3). The region between plus three volts and minus three volts is defined as the transition region. The signal state is not uniquely defined when the voltage (V_I) is in this transition region.

During the transmission of data, the marking condition shall be used to denote the binary state ONE and the spacing condition shall be used to denote the binary state ZERO.

For timing and control interchange circuits, the function shall be considered ON when the voltage (V_1) on the interchange circuit is more positive than plus three volts with respect to circuit AB, and shall be considered OFF when the voltage (V_1) is more negative than minus three volts with respect to Circuit AB. The function is not uniquely defined for voltages in the transition region between plus three volts and minus three volts.

Notation	Interchange Voltage	
	Negative	Positive
Binary State	1	0
Signal Condition	Marking	Spacing
Function	OFF	ON

1313

This specification neither implies nor precludes the use of terminator circuits which utilize hysteresis techniques to enhance their noise immunity; however, the requirements of section 2.5 must also be satisfied.

2.4 The load impedance (R_L and C_L) of the terminator side of an interchange circuit shall have a dc resistance (R_L) of not less than 3000 Ohms, measured with an applied voltage not greater than 25 volts in magnitude, nor more than 7000 Ohms, measured with an applied voltage of 3 to 25 volts in magnitude. The effective shunt capacitance (C_L) of the terminator side of an interchange circuit, measured at the interface point, shall not exceed 2500 picofarads. The reactive component of the load impedance shall not be inductive. The open circuit terminator voltage (E_L) shall not exceed 2 volts in magnitude. (See sections 6.4, 6.5, and 6.6).

2.5 The following interchange circuits, where implemented, shall be used to detect either a power-off condition in the equipment connected across the interface, or the disconnection of the interconnecting cable:

Circuit CA (Request to Send)

Circuit CC (Data Set Ready)

Circuit CD (Data Terminal Ready)

Circuit SCA (Secondary Request to Send)

The power-off source impedance of the driver side of these circuits shall not be less than 300 Ohms, measured with an applied voltage not greater than 2 volts in magnitude referenced to Circuit AB (Signal Ground). The terminator for these circuits shall interpret the power-off condition or the disconnection of the interconnecting cable as an OFF condition.

2.6 The open-circuit driver voltage (V_O) with respect to Circuit AB (Signal Ground) on any interchange circuit shall not exceed 25 volts in magnitude. The source impedance (R_O and C_O) of the driver side of an interchange circuit including any cable to the interface point is not specified: however, the combination of V_O and R_O shall be selected such that a short circuit between any two conductors (including ground) in the interconnecting cable shall not result in a current in excess of one-half ampere. Additionally, the driver design shall be such that, when the terminator load resistance (R_L) is in the range between 3000 Ohms and 7000 Ohms and the terminator open circuit voltage (E_L) is zero, the potential (V_I) at the interface point shall not be less than 5 volts nor more than 15 volts in magnitude (see section 6.5).

2.7 The characteristics of the interchange signals transmitted across the interface point, exclusive of external interference, shall conform to the limitations specified in this section. These limitations shall be satisfied at the interface point when the interchange circuit is terminated with any receiving circuit which meets the requirements given in Section 2.4. These limitations apply to all interchange signals (Data, Control and Timing) unless otherwise specified.

1314

(1) All interchange signals entering into the transition region shall proceed through the transition region to the opposite signal state and shall not reenter the transition region until the next significant change of signal condition.

(2) There shall be no reversal of the direction of voltage change while the signal is in the transition region.

(3) For Control Interchange Circuits, the time required for the signal to pass through the transition region during a change in state shall not exceed one millisecond.

(4) For Data and Timing interchange Circuits, the time required for the signal to pass through the transition region shall not exceed one millisecond or four percent of the nominal duration of a signal element on that interchange circuit, whichever is the lesser.

(5) The maximum instantaneous rate of voltage change shall not exceed 30 volts per microsecond.

SECTION THREE

3. INTERFACE MECHANICAL CHARACTERISTICS

3.1 The interface between the data terminal equipment and data communication equipment is located at a pluggable connector signal interface point between the two equipments. The female connector shall be associated with, but not necessarily physically attached to the data communication equipment and should be mounted in a fixed position near the data terminal equipment. The use of an

extension cable on the data communication equipment is permitted. An extension cable with a male connector shall be provided with the data terminal equipment. The use of short cables (each less than approximately 50 feet or 15 meters) is recommended; however, longer cables are permissible, provided that the resulting load capacitance (C_L of Fig. 2.1), measured at the interface point and including the signal terminator, does not exceed 2500 picofarads. (See section 2.4 and 6.5.)

3.1.1 When additional functions are provided in a separate unit inserted between the data terminal equipment and the data communication equipment (See section 1.7), the female connector, as indicated above shall be associated with the side of this unit which interfaces with the data terminal equipment while the extension cable with the male connector shall be provided on the side which interfaces with the data communication equipment.

Pin Number	Circuit	Description
1	AA	Protective Ground
2	BA	Transmitted Data
3	BB	Received Data
4	CA	Request to Send
5	CB	Clear to Send
6	CC	Data Set Ready
7	AB	Signal Ground (Common Return)
8	CF	Received Line Signal Detector
9	–	(Reserved for Data Set Testing)
10	–	(Reserved for Data Set Testing)
11		Unassigned (See Section 3.2.2)
12	SCF	Sec. Rec'd. Line Sig. Detector
13	SCB	Sec. Clear to Send
14	SBA	Secondary Transmitted Data
15	DB	Transmission Signal Element Timing (DCE Source)
16	SBB	Secondary Received Data
17	DD	Receiver Signal Element Timing (DCE Source)
18		Unassigned
19	SCA	Secondary Request to Send
20	CD	Data Terminal Ready
21	CG	Signal Quality Detector
22	CE	Ring Indicator
23	CH/CI	Data Signal Rate Selector (DTE/DCE Source)
24	DA	Transmit Signal Element Timing (DTE Source)
25		Unassigned

Figure 3.1

Interface Connector Pin Assignments

.1315

3.2 Pin Identification

3.2.1 Pin assignments listed in Figure 3.1 shall be used.

3.2.2 Pin assignments for circuits not specifically defined in section 4 (See section 4.1.1.) are to be made by mutual agreement. Preference should be given to the use of unassigned pins, but in the event that additional pins are required extreme caution should be taken in their selection.

SECTION FOUR

4. FUNCTIONAL DESCRIPTION OF INTERCHANGE CIRCUITS

4.1 General

This section defines the basic interchange circuits which apply, collectively, to all systems.

4.1.1 Additional interchange circuits not defined herein, or variations in the functions of the defined interchange circuits may be provided by mutual agreement. See sections 3.2.2. and 5.2.

4.2 Categories

1316

Interchange circuits between data terminal equipment and data communication equipment fall into four general categories.

Ground or Common Return
Data Circuits
Control Circuits
Timing Circuits

4.2.1 A list of circuits showing category as well as equivalent C.C.I.T.T. identification in accordance with Recommendation V.24 is presented in Figure 4.1.

4.3 Signal Characteristics, General

4.3.1 Interchange circuits transferring data signals across the interface point shall hold marking (binary ONE) or spacing (binary ZERO) conditions for the total nominal duration of each signal element.

In synchronous systems using synchronous data communication equipment, distortion tolerances as specified in RS-334[1] shall apply. Acceptable distortion tolerances for data terminal equipment in synchronous and start-stop (i.e. asynchronous) systems using non-synchronous

[1] RS-334 "Signal Quality at Interface Between Data Processing Terminal Equipment and Synchronous Data Communication Equipment for Serial Data Transmission" – March 1967.

data communication equipment are under consideration for a future companion standard to RS-334.

4.3.2 Interchange circuits transferring timing signals across the interface point shall hold ON and OFF conditions for nominally equal periods of time, consistent with acceptable tolerances as specified in RS-334. During periods when timing information is not provided on a timing interchange circuit, this interchange circuit shall be clamped in the OFF condition.

4.3.3 Tolerances on the relationship between data and associated timing signals shall be in accordance with RS-334.

Interchange Circuit	C.C.I.T.T. Equivalent	Description	Gnd	Data		Control		Timing	
				From DCE	To DCE	From DCE	To DCE	From DCE	To DCE
AA	101	Protective Ground	X						
AB	102	Signal Ground/Common Return	X						
BA	103	Transmitted Data			X				
BB	104	Received Data		X					
CA	105	Request to Send					X		
CB	106	Clear to Send				X			
CC	107	Data Set Ready				X			
CD	108.2	Data Terminal Ready					X		
CE	125	Ring Indicator				X			
CF	109	Received Line Signal Detector				X			
CG	110	Signal Quality Detector				X			
CH	111	Data Signal Rate Selector (DTE)					X		
CI	112	Data Signal Rate Selector (DCE)				X			
DA	113	Transmitter Signal Element Timing (DTE)							X
DB	114	Transmitter Signal Element Timing (DCE)						X	
DD	115	Receiver Signal Element Timing (DCE)						X	
SBA	118	Secondary Transmitted Data			X				
SBB	119	Secondary Received Data		X					
SCA	120	Secondary Request to Send					X		
SCB	121	Secondary Clear to Send				X			
SCF	122	Secondary Rec'd Line Signal Detector				X			

Figure 4.1

Interchange Circuits by Category

4.4 Interchange Circuits

Circuit AA – Protective Ground (C.C.I.T.T. 101)
Direction: Not applicable

This conductor shall be electrically bonded to the machine or equipment frame. It may be furthe connected to external grounds as required by applicable regulations.

Circuit AB – Signal Ground or Common Return (C.C.I.T.T. 102)
Direction: Not applicable

This conductor establishes the common ground reference potential for all interchange circuits excep Circuit AA (Protective Ground). Within the data communication equipment, this circuit shall b brought to one point, and it shall be possible to connect this point to Circuit AA by means of a wir strap inside the equipment. This wire strap can be connected or removed at installation, as may b required to meet applicable regulations or to minimize the introduction of noise into electroni circuitry.

Circuit BA – Transmitted Data (C.C.I.T.T. 103)
Direction: TO data communication equipment

Signals on this circuit are generated by the data terminal equipment and are transferred to the loca transmitting signal converter for transmission of data to remote data terminal equipment.

1318

The data terminal equipment shall hold Circuit BA (Transmitted Data) in marking condition durin intervals between characters or words, and at all times when no data are being transmitted.

In all systems, the data terminal equipment shall not transmit data unless an ON condition is preser on all of the following four circuits, where implemented.

1. Circuit CA (Request to Send)
2. Circuit CB (Clear to Send)
3. Circuit CC (Data Set Ready)
4. Circuit CD (Data Terminal Ready)

All data signals that are transmitted across the interface on interchange circuit BA (Transmitted Dat during the time an ON condition is maintained on all of the above four circuits, where implemente shall be transmitted to the communication channel.

See Section 4.3, for signal characteristics.

Circuit BB – Received Data (C.C.I.T.T. 104)
Direction: FROM data communication equipment

Signals on this circuit are generated by the receiving signal converter in response to data sign received from remote data terminal equipment via the remote transmitting signal converter. Circuit

(Received Data) shall be held in the binary ONE (Marking) condition at all times when Circuit CF (Received Line Signal Detector) is in the OFF condition.

On a half duplex channel, Circuit BB shall be held in the Binary One (Marking) condition when Circuit CA (Request to Send) is in the ON condition and for a brief interval following the ON to OFF transition of Circuit CA to allow for the completion of transmission (See Circuit BA -- Transmitted Data) and the decay of line reflections. See section 4.3 for signal characteristics.

Circuit CA -- Request to Send (C.C.I.T.T. 105)
Direction: TO data communication equipment

This circuit is used to condition the local data communication equipment for data transmission and, on a half duplex channel, to control the direction of data transmission of the local data communication equipment.

On one way only channels or duplex channels, the ON condition maintains the data communication equipment in the transmit mode. The OFF condition maintains the data communication equipment in a non-transmit mode.

On a half duplex channel, the ON condition maintains the data communication equipment in the transmit mode and inhibits the receive mode. The OFF condition maintains the data communication equipment in the receive mode.

A transition from OFF to ON instructs the data communication equipment to enter the transmit code (see Section 6.8). The data communication equipment responds by taking such action as may be necessary and indicates completion of such actions by turning ON Circuit CB (Clear to Send), thereby indicating to the data terminal equipment that data may be transferred across the interface point on interchange Circuit BA (Transmitted Data).

A transition from ON to OFF instructs the data communication equipment to complete the transmission of all data which was previously transferred across the interface point on interchange Circuit BA and then assume a non-transmit mode or a receive mode as appropriate. The data communication equipment responds to this instruction by turning OFF Circuit CB (Clear to Send) when it is prepared to again respond to a subsequent ON condition of Circuit CA.

> *NOTE:* A non-transmit mode does not imply that all line signals have been removed from the communication channel. See section 6.8.

When Circuit CA is turned OFF, it shall not be turned ON again until Circuit CB has been turned OFF by the data communication equipment.

An ON condition is required on Circuit CA as well as on Circuit CB, Circuit CC (Data Set Ready) and, where implemented, Circuit CD (Data Terminal Ready) whenever the data terminal equipment transfers data across the interface on interchange Circuit BA.

It is permissible to turn Circuit CA ON at any time when Circuit CB is OFF regardless of the condition of any other interchange circuit.

Circuit CB – Clear to Send (C.C.I.T.T. 106)
Direction: FROM data communication equipment

Signals on this circuit are generated by the data communication equipment to indicate whether or not the data set is ready to transmit data.

The ON condition together with the ON condition on interchange circuits CA, CC and, where implemented, CD, is an indication to the data terminal equipment that signals presented on Circuit BA (Transmitted Data) will be transmitted to the communication channel.

The OFF condition is an indication to the data terminal equipment that it should not transfer data across the interface on interchange Circuit BA.

The ON condition of Circuit CB is a response to the occurrence of a simultaneous ON condition on Circuits CC (Data Set Ready) and Circuit CA (Request to Send), delayed as may be appropriate to the data communication equipment for establishing a data communication channel (including the removal of the MARK HOLD clamp from the Received Data interchange circuit of the remote data set) to a remote data terminal equipment.

Where Circuit CA (Request to Send) is not implemented in the data communication equipment with transmitting capability, Circuit CA shall be assumed to be in the ON condition at all times, and Circuit CB shall respond accordingly.

1320

Circuit CC – Data Set Ready (C.C.I.T.T. 107)
Direction: FROM data communication equipment

Signals on this circuit are used to indicate the status of the local data set.

The ON condition on this circuit is presented to indicate that –

a) the local data communication equipment is connected to a communication channel ("OFF HOOK" in switched service),

AND b) the local data communication equipment is not in test (local or remote), talk (alternate voice) or dial* mode, (See section 6.10).

AND c) the local data communication equipment has completed, where applicable

1. any timing functions required by the switching system to complete call establishment, and

2. the transmission of any discreet answer tone, the duration of which is controlled solely by the local data set.

---

* The data communication equipment is considered to be in the dial mode when circuitry directly associated with the call origination function is connected to the communication channel. These functions include signalling to the central office (dialing) and monitoring the communication channel for call progress or answer back signals.

Where the local data communication equipment does not transmit an answer tone, or where the duration of the answer tone is controlled by some action of the remote data set, the ON condition is presented as soon as all the other above conditions (a, b, and c-1) are satisfied.

This circuit shall be used only to indicate the status of the local data set. The ON condition shall not be interpreted as either an indication that a communication channel has been established to a remote data station or the status of any remote station equipment.

The OFF condition shall appear at all other times and shall be an indication that the data terminal equipment is to disregard signals appearing on any other interchange circuit with the exception of Circuit CE (Ring Indicator). The OFF condition shall not impair the operation of Circuit CE or Circuit CD (Data Terminal Ready).

When the OFF condition occurs during the progress of a call before Circuit CD is turned OFF, the data terminal equipment shall interpret this as a lost or aborted connection and take action to terminate the call. Any subsequent ON condition on Circuit CC is to be considered a new call.

When the data set is used in conjunction with Automatic Calling Equipment, the OFF to ON transition of Circuit CC shall not be interpreted as an indication that the ACE has relinquished control of the communication channel to the data set. Indication of this is given on the appropriate lead in the ACE interface (see EIA Standard RS-366).

> *Note:* Attention is called to the fact that if a data call is interrupted by alternate voice communication, Circuit CC will be in the OFF condition during the time that voice communication is in progress. The transmission or reception of the signals required to condition the communication channel or data communication equipment in response to the ON condition of interchange Circuit CA (Request to Send) of the transmitting data terminal equipment will take place after Circuit CC comes ON, but prior to the ON condition on Circuit CB (Clear to Send) or Circuit CF (Received Line Signal Detector).

1321

Circuit CD – Data Terminal Ready (C.C.I.T.T. 108.2)
Direction: To data communication equipment

Signals on this circuit are used to control switching of the data communication equipment to the communication channel. The ON condition prepares the data communication equipment to be connected to the communication channel and maintains the connection established by external means (e.g., manual call origination, manual answering or automatic call origination).

When the station is equipped for automatic answering of received calls and is in the automatic answering mode, connection to the line occurs only in response to a combination of a ringing signal and the ON condition of Circuit CD (Data Terminal Ready); however, the data terminal equipment is normally permitted to present the ON condition on Circuit CD whenever it is ready to transmit or receive data, except as indicated below.

The OFF condition causes the data communication equipment to be removed from the communication channel following the completion of any "in process" transmission. See Circuit BA (Transmitted Data). The OFF condition shall not disable the operation of Circuit CE (Ring Indicator).

In switched network applications, when circuit CD is turned OFF, it shall not be turned ON again until Circuit CC (Data Set Ready) is turned OFF by the data communication equipment.

Circuit CE – Ring Indicator (C.C.I.T.T. 125)
Direction: FROM data communication equipment

The ON condition of this circuit indicates that a ringing signal is being received on the communication channel.

The ON condition shall appear approximately coincident with the ON segment of the ringing cycle (during rings) on the communication channel.

The OFF condition shall be maintained during the OFF segment of the ringing cycle (between "rings") and at all other times when ringing is not being received. The operation of this circuit shall not be disabled by the OFF condition on Circuit CD (Data Terminal Ready).

Circuit CF – Received Line Signal Detector (C.C.I.T.T. 109)
Direction: FROM data communication equipment

The ON condition on this circuit is presented when the data communication equipment is receiving a signal which meets its suitability criteria. These criteria are established by the data communication equipment manufacturer.

1322

The OFF condition indicates that no signal is being received or that the received signal is unsuitable for demodulation.

The OFF condition of Circuit CF (Received Line Signal Detector) shall cause Circuit BB (Received Data) to be clamped to the Binary One (Marking) condition.

The indications on this circuit shall follow the actual onset or loss of signal by appropriate guard delays.

On half duplex channels, Circuit CF is held in the OFF condition whenever Circuit CA (Request to Send) is in the ON condition and for a brief interval of time following the ON to OFF transition of Circuit CA. (See Circuit BB.)

Circuit CG – Signal Quality Detector (C.C.I.T.T. 110)
Direction: FROM data communication equipment

Signals on this circuit are used to indicate whether or not there is a high probability of an error in the received data.

An ON condition is maintained whenever there is no reason to believe that an error has occurred.

An OFF condition indicates that there is a high probability of an error. It may, in some instances, be used to call automatically for the retransmission of the previously transmitted data signal. Preferably the response of this circuit shall be such as to permit identification of individual questionable signal elements on Circuit BB (Received Data).

Circuit CH – Data Signal Rate Selector (DTE Source) (C.C.I.T.T. 111)
Direction: TO data communication equipment

Signals on this circuit are used to select between the two data signalling rates in the case of dual rate synchronous data sets or the two ranges of data signalling rates in the case of dual range non-synchronous data sets.

An ON condition shall select the higher data signalling rate or range of rates.

The rate of timing signals, if included in the interface, shall be controlled by this circuit as may be appropriate.

Circuit CI – Data Signal Rate Selector (DCE Source) (C.C.I.T.T. 112)
Direction: FROM data communication equipment

Signals on this circuit are used to select between the two data signalling rates in the case of dual rate synchronous data sets or the two ranges of data signalling rates in the case of dual range non-synchronous data sets.

An ON condition shall select the higher data signalling rate or range of rates.

The rate of timing signals, if included in the interface, shall be controlled by this circuit as may be appropriate.

Circuit DA – Transmitter Signal Element Timing (DTE Source) (C.C.I.T.T. 113)
Direction: TO data communication equipment

Signals on this circuit are used to provide the transmitting signal converter with signal element timing information.

The ON to OFF transition shall nominally indicate the center of each signal element on Circuit BA (Transmitted Data). When Circuit DA is implemented in the DTE, the DTE shall normally provide timing information on this circuit whenever the DTE is in a POWER ON condition. It is permissible for the DTE to withhold timing information on this circuit for short periods provided Circuit CA (Request to Send) is in the OFF condition. (For example, the temporary withholding of timing information may be necessary in performing maintenance tests within the DTE.)

Circuit DB – Transmitter Signal Element Timing (DCE Source) (C.C.I.T.T. 114)
Direction: FROM data communication equipment

Signals on this circuit are used to provide the data terminal equipment with signal element timing information. The data terminal equipment shall provide a data signal on Circuit BA (Transmitted Data) in which the transitions between signal elements nominally occur at the time of the transitions from OFF to ON condition of the signal on Circuit DB. When Circuit DB is implemented in the DCE, the DCE shall normally provide timing information on this circuit whenever the DCE is in a POWER ON condition. It is permissible for the DCE to withhold timing information on this circuit for short periods provided Circuit CC (Data Set Ready) is in the OFF condition. (For example, the withholding of timing information may be necessary in performing maintenance tests within the DCE.)

Circuit DD – Receiver Signal Element Timing (DCE Source) (C.C.I.T.T. 115)
Direction: FROM data communication equipment.

Signals on this circuit are used to provide the data terminal equipment with received signal element timing information. The transition from ON to OFF condition shall nominally indicate the center of each signal element on Circuit BB (Received Data). Timing information on Circuit DD shall be provided at all times when Circuit CF (Received Line Signal Detector) is in the ON condition. It may, but need not be present following the ON to OFF transition of Circuit CF (See section 4.3.2).

Circuit SBA – Secondary Transmitted Data (C.C.I.T.T. 118)
Direction: TO data communication equipment

This circuit is equivalent to Circuit BA (Transmitted Data) except that it is used to transmit data via the secondary channel.

Signals on this circuit are generated by the data terminal equipment and are connected to the local secondary channel transmitting signal converter for transmission of data to remote data terminal equipment.

The data terminal equipment shall hold Circuit SBA (Secondary Transmitted Data) in marking condition during intervals between characters or words and at all times when no data are being transmitted.

1324

In all systems, the data terminal equipment shall not transmit data on the secondary channel unless an ON condition is present on all of the following four circuits, where implemented:

1. Circuit SCA – Secondary Request to Send
2. Circuit SCB – Secondary Clear to Send
3. Circuit CC – Data Set Ready
4. Circuit CD – Data Terminal Ready

All data signals that are transmitted across the interface on interchange Circuit SBA during the time when the above conditions are satisfied shall be transmitted to the communication channel. See Section 4.3.

When the secondary channel is useable only for circuit assurance or to interrupt the flow of data in the primary channel (less than 10 Baud capability), Circuit SBA (Secondary Transmitted Data) is normally not provided, and the channel carrier is turned ON or OFF by means of Circuit SCA (Secondary Request to Send). Carrier OFF is interpreted as an "Interrupt" condition.

Circuit SBB – Secondary Received Data (C.C.I.T.T. 119)
Direction: FROM data communication equipment

This circuit is equivalent to Circuit BB (Received Data) except that it is used to receive data on the secondary channel.

When the secondary channel is useable only for circuit assurance or to interrupt the flow of data in the primary channel, Circuit SBB is normally not provided. See interchange Circuit SCF (Secondary Received Line Signal Detector).

Circuit SCA – Secondary Request to Send (C.C.I.T.T. 120)
Direction: TO data communication equipment

This circuit is equivalent to Circuit CA (Request to Send) except that it requests the establishment of the secondary channel instead of requesting the establishment of the primary data channel.

Where the secondary channel is used as a backward channel, the ON condition of Circuit CA (Request to Send) shall disable Circuit SCA and it shall not be possible to condition the secondary channel transmitting signal converter to transmit during any time interval when the primary channel transmitting signal converter is so conditioned. Where system considerations dictate that one or the other of the two channels be in transmit mode at all times but never both simultaneously, this can be accomplished by permanently applying an ON condition to Circuit SCA (Secondary Request to Send) and controlling both the primary and secondary channels, in complementary fashion, by means of Circuit CA (Request to Send). Alternatively, in this case, Circuit SCB need not be implemented in the interface.

When the secondary channel is useable only for circuit assurance or to interrupt the flow of data in the primary data channel, Circuit SCA shall serve to turn ON the secondary channel unmodulated carrier. The OFF condition of Circuit SCA shall turn OFF the secondary channel carrier and thereby signal an interrupt condition at the remote end of the communication channel.

Circuit SCB – Secondary Clear to Send (C.C.I.T.T. 121)
Direction: FROM data communication equipment

This circuit is equivalent to Circuit CB (Clear to Send), except that it indicates the availability of the secondary channel instead of indicating the availability of the primary channel. This circuit is not provided where the secondary channel is useable only as a circuit assurance or an interrupt channel.

Circuit SCF – Secondary Received Line Signal Detector (C.C.I.T.T. 122)
Direction: FROM data communication equipment

This circuit is equivalent to Circuit CF (Received Line Signal Detector) except that it indicates the proper reception of the secondary channel line signal instead of indicating the proper reception of a primary channel received line signal.

Where the secondary channel is useable only as a circuit assurance or an interrupt channel (see Circuit SCA – Secondary Request to Send), Circuit SCF shall be used to indicate the circuit assurance status or to signal the interrupt. The ON condition shall indicate circuit assurance or a non-interrupt condition. The OFF condition shall indicate circuit failure (no assurance) or the interrupt condition.

SECTION FIVE

5. STANDARD INTERFACES FOR SELECTED COMMUNICATION SYSTEM CONFIGURATIONS

5.1 This section describes a selected set of data transmission configurations. For each of these configurations a standard set of interchange circuits (defined in section 4) is listed. (See section 6.2.)

5.1.1 Provision is made for additional data transmission configurations not defined herein. Interchange circuits for these applications must be specified separately, for each application, by the supplier.

5.2 Drivers shall be provided for every interchange circuit included in the standard interface. Terminators need not be provided for every interchange circuit included in the standard interface; however, the designer of the equipment which does not provide all of the specified terminators must be aware that any degradation in service due to his disregard of a standard interchange circuit is his responsibility.

In the interest of minimizing the number of different types of equipment, additional interchange circuits may be included in the design of a general unit capable of satisfying the requirements of several different applications.

5.2.1 For a given configuration, interchange circuits which are included in the standard list and for which drivers are provided, but which the manufacturer of equipment at the receiving side of the interface chooses not to use, shall be suitably terminated by means of a dummy load impedance in the equipment which normally provides the terminator. See Section 2.4.

1326

5.2.2 Where interchange circuits not on the standard list are provided for a given configuration, the designer of this equipment must be prepared to find an open circuit on the other side of the interface, and the system shall not suffer degradation of the basic service.

Interference due to unterminated drivers in this category is the responsibility of the designer who includes these drivers.

Terminators shall not interfere with or degrade system performance as a result of open circuited input terminals.

5.3 Circuit configurations for which standard sets of interchange circuits are defined are listed in Figure 5.1.

5.4 The use of Circuit AA (Protective Ground) is optional. Where it is used, attention is called to the applicable Underwriters' regulation applying to wire size and color coding. Where it is not used, other provisions for grounding equipment frames should be made in accordance with good engineering practice.

5.5 The use of Circuit AB (Signal Ground) is mandatory in all systems. See section 1.4.

5.6 Secondary channels, where involved in the standard interfaces, are shown as Auxiliary Channels.

5.6.1 Where secondary channels are intended for use as backward channels, Circuit SCA (Secondary Request to Send) shall be interconnected with Circuit CA (Request to Send) within the data communication equipment and need not be brought out to the interface. See Section 4.4, Interchange Circuit SCA (Secondary Request to Send) for detailed information.

5.6.2 Where secondary channels are useable only for circuit assurance or to interrupt the flow of data in the primary channel, they transmit no actual data and depend only on the presence or absence of the secondary channel carrier. For this application only, Circuit SBA (Secondary Transmitted Data), SBB (Secondary Received Data) and SCB (Secondary Clear to Send) are not provided. Circuit SCA (Secondary Request to Send) turns secondary channel carrier ON and OFF as required and Circuit SCF (Secondary Received Line Signal Detector) recognizes its presence or absence. See definitions of Circuits SCA and SCF in Section 4.4 for details.

Data Transmission Configuration	Interface Type
Transmit Only	A
Transmit Only*	B
Receive Only	C
Half Duplex	D
Duplex*	D
Duplex	E
Primary Channel Transmit Only * / Secondary Channel Receive Only	F
Primary Channel Transmit Only / Secondary Channel Receive Only	H
Primary Channel Receive Only / Secondary Channel Transmit Only*	G
Primary Channel Receive Only / Secondary Channel Transmit Only	I
Primary Channel Transmit Only* / Half Duplex Secondary Channel	J
Primary Channel Receive Only / Half Duplex Secondary Channel	K
Half Duplex Primary Channel / Half Duplex Secondary Channel	L
Duplex Primary Channel* / Duplex Secondary Channel*	L
Duplex Primary Channel / Duplex Secondary Channel	M
Special (Circuits specified by Supplier)	Z

Note: Data Transmission Configurations identified with an asterisk (*) indicate the inclusion of Circuit CA (Request to Send) in a One Way Only (Transmit) or Duplex Configuration where it might ordinarily not be expected, but where it might be used to indicate a non-transmit mode to the data communication equipment to permit it to remove a line signal or to send synchronizing or traning signals as required.

Figure 5.1

Interface Types for Data Transmission Configurations

Interchange Circuit		Interface Type													
		A	B	C	D	E	F	G	H	I	J	K	L	M	Z
AA	Protective Ground	—	—	—	—	—	—	—	—	—	—	—	—	—	—
AB	Signal Ground	x	x	x	x	x	x	x	x	x	x	x	x	x	x
BA	Transmitted Data	x	x		x	x	x		x		x		x	x	°
BB	Received Data			x	x	x		x		x		x	x	x	°
CA	Request to Send		x		x		x				x		x		°
CB	Clear to Send	x	x		x	x	x		x		x		x	x	°
CC	Data Set Ready	x	x	x	x	x	x	x	x	x	x	x	x	x	°
CD	Data Terminal Ready	s	s	s	s	s	s	s	s	s	s	s	s	s	°
CE	Ring Indicator	s	s	s	s	s	s	s	s	s	s	s	s	s	°
CF	Received Line Signal Detector			x	x	x		x		x		x	x	x	°
CG	Signal Quality Detector														°
CH/CI	Data Signalling Rate Selector (DTE) ,(DCE)														°
DA/DB	Transmitter Sig. Element Timing (DTE) ,(DCE)	t	t		t	t	t		t		t	t	t	t	°
DD	Receiver Signal Element Timing (DCE)			t	t	t		t		t		t	t	t	°
SBA	Secondary Transmitted Data							x		x	x	x	x	x	°
SBB	Secondary Received Data						x		x		x	x	x	x	°
SCA	Secondary Request to Send							x			x	x	x		°
SCB	Secondary Clear to Send							x		x	x	x	x	x	°
SCF	Secondary Received Line Signal Detector						x		x		x	x	x	x	°

Legend: ° - To be specified by the supplier
— - optional
s - Additional Interchange Circuits required for Switched Service
t - Additional Interchange Circuits required for Synchronous Channel
x - Basic Interchange Circuits, All Systems

Figure 5.2

Standard Interfaces For
Selected Communication Systems Configurations

SECTION SIX

6. RECOMMENDATIONS AND EXPLANATORY NOTES

6.1 The data are to be serialized by the data terminal equipment so that the design of the data communication equipment may be independent of the character length and code used by the data terminal equipment. The data communication equipment shall place no restrictions on the arrangement of the sequence of bits provided by the data terminal equipment.

6.2 The control interchange circuits at the interface point are arranged to permit the alternate use of a higher class of communication service as follows:

A. Data terminal equipment designed for Transmit-Only or Receive-Only service may also use either Half-duplex or Duplex service.

B. Data terminal equipment designed for Half-duplex service may also use Duplex service.

6.3 The electrical specifications are intended to provide a two-volt margin in rejecting noise introduced either on interchange circuits or by a difference in reference ground potential across the interface. The equipment designer should maintain this margin of safety on all interchange circuitry.

6.4 To avoid inducing voltage surges on interchange circuits, signals from interchange circuits should not be used to drive inductive devices, such as relay coils. (Note that relay or switch contacts may be used to generate signals on an interchange circuit, with appropriate measures to assure that signals so generated comply with Section 2.7.)

1330

6.5 Alphabetical parenthetical designations are added to the terms used in Sections 2.3, 2.4, and 2.6 to better tie them in with the equivalent circuit of Section 2.1 and stress the point that the 2500 picofarad capacitance (C_L) is defined for the receiving end of the interchange circuit and that the capacitance (C_O) at the driving end of the interchange circuit, including cable, is not defined. It is the responsibility of the designer to build a circuit capable of driving all of the capacitance in the driver circuitry plus the capacitance in his part of the interconnecting cable (not specified) plus 2500 pF in the load (including the cable on the load side of the interface point).

6.6 The user is reminded that the characteristics of an equivalent load (terminator) circuit used to test for compliance with each of the electrical specifications in section 2 are a function not only of the parameter under test, but also of the tolerance limit to be tested. For example, a driver which just delivers a minimum of 5 Volts into a 7,000 Ohm test load may fail the test if the load is reduced to 3,000 Ohms, whereas, a driver with an output within the 15 Volt limit when driving a 3,000 Ohm load may exceed this limit when driving a 7,000 Ohm load. The 5 Volt tolerance should therefore be tested with a 3,000 Ohm load while the 15 Volt limit should be tested using a 7,000 Ohm load.

6.7 The operation of the transmitting and receiving circuits should minimize the effects of any circuit time constants which would delay the circuit response and introduce time distortion of the signals.

6.8 The turning ON of Circuit CA (Request to Send) does not necessarily imply the turning ON of a line signal on the communication channel. Some data sets might not have a line signal as it is understood in this standard, e.g. the signal can be a modified digital base-band signal.

Conversely, in data sets which do transmit a "line signal", the turning OFF of Circuit CA does not necessarily command the removal of that line signal from the communication channel. On a duplex channel, the data set might autonomously transmit a training signal to hold AGC Circuits or automatic equalizers in adjustment, or to keep timing locked (synchronized) when Circuit CA is OFF.

It is not within the scope of this standard to specify in detail what occurs on the communication channel (line) side of the data communication equipment. Therefore the definition for Circuit CA uses the terminology "assume the transmit mode" intentionally avoiding reference to "carrier" or "line signals".

However, the continued requirement for multipoint systems is recognized. Data sets intended for this type of operation should permit the sharing of a communication channel by more than one data set transmitter and should, when in a non-transmit mode, place no signal on the communication channel which might interfere with the transmission from another data set in the network.

6.9 It is important that, at an answering data station, Circuit CC (Data Set Ready) be turned ON independently of any event which might occur at the remote (calling) data station. This independence permits the use of the OFF to ON transition of Circuit CC to start an "abort timer" in the data terminal equipment. This timer would cause termination of an automatically answered call (by causing Circuit CD to be turned OFF) if other expected events such as Circuit CF ON or proper exchange of data do not occur in a predetermined time interval. Such independence is necessary to assure the starting of the abort timer when an automatically answered incoming call is the result of a wrong number reached from a regular (non data station) telephone instrument.

6.10 Although the method of operation for multi-line automatic calling equipment, RS-366 (when assigned) Interface Type V, has not yet been fully defined, it appears that a situation could arise during call origination where both the DCE and the ACE appear to be idle (at the interface) even though actively engaged in establishing a connection.

One possible solution to this problem requires that circuit CC be turned ON upon completion of dialing to provide continuity of signalling during call origination. When multiline automatic calling equipment is used, Circuit CC would thus turn ON earlier than specified in Section 4.4 herein. This solution is subject to further study; however, data terminal equipment which may, in the future, be used in systems with multi-line automatic calling equipment should not be adversely affected by this early "Data Set Ready" indication.

1331

SECTION SEVEN

7. GLOSSARY OF NEW TERMS

7.1 This section defines terms used in this standard which are new or are used in a special sense.

1. Data Transmission Channel

The transmission media and intervening equipment involved in the transfer of information between data terminal equipments. A data transmission channel includes the signal conversion equipment. A data transmission channel may support the transfer of information in one direction only, in either direction alternately, or in both directions simultaneously and the channel is accordingly classified as defined in the following sections. When the data communications equipment has more than one speed capability associated with it, for example 1200 baud transmission in one direction and 150 baud transmission in the opposite direction, a channel is defined for each speed capability.

2. Primary Channel

The data transmission channel having the highest signaling rate capability of all the channels sharing a common interface connector. A primary channel may support the transfer of information in one direction only, either direction alternately or both directions simultaneously and is then classified as "one way only", "half duplex" or "duplex" as defined herein.

1332

3. Secondary Channel

The data transmission channel having a lower signaling rate capability than the primary channel in a system in which two channels share a common interface connector. A secondary channel may be either one way only, half duplex or duplex as defined later. Two classes of secondary channels are defined, auxiliary and backward.

4. Auxiliary Channel

A secondary channel whose direction of transmission is independent of the primary channel and is controlled by an appropriate set of secondary control interchange circuits.

5. Backward Channel

A secondary channel whose direction of transmission is constrained to be always opposite to that of the primary channel. The direction of transmission of the backward channel is restricted by the control interchange circuit (Circuit CA – Request to Send) that controls the direction of transmission of the primary channel.

6. One Way Only (Unidirectional) Channel

A primary or secondary channel capable of operation in only one direction. The direction is fixed and cannot be reversed. The term "one way only" used to describe a primary channel

does not imply anything about the type of secondary channel or the existence of a secondary channel; similary, the use of the term to describe a secondary channel implies nothing about the type of primary channel present.

7. Half Duplex Channel

A primary or secondary channel capable of operating in both directions but not simultaneously. The direction of transmission is reversible. The term half duplex used to describe a primary channel does not imply anything about the type of secondary channel; similarly, the use of the term to describe a secondary channel implies nothing about the type of primary channel present. (Note that as a result of the definitions, both directions of a half duplex channel have the same signaling rate capability.)

8. Duplex Channel (Full Duplex Channel)

A primary or secondary channel capable of operating in both directions simultaneously. The term duplex used to describe a primary channel does not imply anything about the type of secondary channel or the existence of a secondary channel; similarly, the use of the term to describe a secondary implies nothing about the type of primary channel present. (Note that a full duplex channel has the same signaling rate capability in both directions. A system with different rates would be considered to be a one way only primary channel in one direction and a one way only secondary channel in the opposite direction.)

9. Synchronous Data Transmission Channel

A data channel in which timing information is transferred between the data terminal equipment and the data communication equipment. Transmitter Signal Element Timing signals can be provided by either the data terminal equipment or by the data communication equipment. Receiver Signal Element Timing is normally recovered in and provided by the Data Communication Equipment. A synchronous data channel will not accommodate Start/Stop data signals unless they are transmitted isochronously and timing signals are interchanged at least at the transmitting station.

10. Nonsynchronous Data Transmission Channel

A data channel in which no timing information is transferred between the data terminal equipment and the data communication equipment.

11. Dedicated Line

A communications channel which is nonswitched, i.e., which is permanently connected between two or more data stations. These communication channels are also referred to as "leased" or "private"; however, since leased and private switched networks do exist, the term "dedicated" is preferred herein to define a nonswitched connection between two or more stations.

12. Interchange Circuit

A circuit between the data terminal equipment and the data communication equipment for the purpose of exchanging data, control or timing signals. Circuit AB (signal ground) is a common reference for all interchange circuits.

13. Driver

a. The electronic circuitry or relay contact at the transmitting end (source) of an interchange circuit which transmits binary digital signals to a terminator via an interconnecting cable.

b. The transmitter of a binary digital signal.

14. Terminator

a. The electronic circuitry at the receiving end (sink) of an interchange circuit which receives binary digital signals from a driver via an interconnecting cable.

b. The receiver of a binary digital signal.

15. Signal Conversion Equipment

1334 Those portions of the data communication equipment which transform (e.g., modulate, shape, etc.) the data signals exchanged across the interface into signals suitable for transmission through the associated communication media or which transform (e.g., demodulate, slice, regenerate, etc.) the received line signals into data signals suitable for presentation to the data terminal equipment.

APPENDIX I

INTERFACE CONNECTOR

While no industry standard exists which defines a suitable interface connector, it should be noted that commercial products are available which will perform satisfactorily as electrical connectors for interfaces specified in RS-232C, such as those connectors meeting Military Specification MIL-C-24308 (MS-18275) or equivalent.

It is not intended that the above reference be considered as part of RS-232C or as a standard for the devices to which reference is made.

INTERNATIONAL STANDARD ISO 646

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION

7-bit coded character set for information processing interchange

First edition — 1973-07-01

533

UDC 681.3.042 : 003.62

Ref. No. ISO 646-1973 (E)

Descriptors : data processing, data transmission, character sets, coding, information interchange.

Price based on 12 pages

FOREWORD

ISO (the International Organization for Standardization) is a worldwide federation of national standards institutes (ISO Member Bodies). The work of developing International Standards is carried out through ISO Technical Committees. Every Member Body interested in a subject for which a Technical Committee has been set up has the right to be represented on that Committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work.

Draft International Standards adopted by the Technical Committees are circulated to the Member Bodies for approval before their acceptance as International Standards by the ISO Council.

International Standard ISO 646 was drawn up by Technical Committee ISO/TC 97, *Computers and information processing*, and circulated to the Member Bodies in April 1972.

It has been approved by the Member Bodies of the following countries :

Belgium	Ireland	Sweden
Brazil	Italy	Switzerland
Canada	Japan	Thailand
Czechoslovakia	Netherlands	United Kingdom
Denmark	Portugal	U.S.A.
Egypt, Arab Rep. of	Romania	U.S.S.R.
France	South Africa, Rep. of	
Germany	Spain	

No Member Body expressed disapproval of the document.

534

Printed in Switzerland

7-bit coded character set for information processing interchange

535

1 SCOPE AND FIELD OF APPLICATION

1.1 This International Standard contains a set of 128 characters (control characters and graphic characters such as letters, digits and symbols) with their coded representation. Most of these characters are mandatory and unchangeable, but provision is made for some flexibility to accommodate special national and other requirements.

1.2 The need for graphics and controls in data processing and in data transmission has been taken into account in determining this character set.

1.3 This International Standard consists of a general table with a number of options, notes, a legend and explanatory notes. It also contains a specific international reference version, guidance on the exercise of the options to define specific national versions and application oriented versions.

1.4 This character set is primarily intended for the interchange of information among data processing systems and associated equipment, and within message transmission systems.

1.5 This character set is applicable to all latin alphabets.

1.6 This character set includes facilities for extension where its 128 characters are insufficient for particular applications.

1.7 The definitions of some control characters in this International Standard assume that data associated with them is to be processed serially in a forward direction. Their effect when included in strings of data which are processed other than serially in a forward direction or included in data formatted for fixed record processing may have undesirable effects or may require additional special treatment to ensure that the control characters have their desired effect.

2 IMPLEMENTATION

2.1 This character set should be regarded as a basic alphabet in an abstract sense. Its practical use requires definitions of its implementation in various media. For example, this could include punched tapes, punched cards, magnetic tapes and transmission channels, thus permitting interchange of data to take place either indirectly by means of an intermediate recording in a physical medium, or by local electrical connection of various units (such as input and output devices and computers) or by means of data transmission equipment.

2.2 The implementation of this coded character set in physical media and for transmission, taking into account the need for error checking, is the subject of other ISO publications. (See Appendix Y.)

3 BASIC CODE TABLE

TABLE 1 – Basic code table

				b_7	0	0	0	0	1	1	1	1
				b_6	0	0	1	1	0	0	1	1
				b_5	0	1	0	1	0	1	0	1
				column / row	0	1	2	3	4	5	6	7
b_4	b_3	b_2	b_1									
0	0	0	0	0	NUL	TC_7 (DLE)	SP	0	@	P	` ⊙	p
0	0	0	1	1	TC_1 (SOH)	DC_1	!	1	A	Q	a	q
0	0	1	0	2	TC_2 (STX)	DC_2	" ⊙	2	B	R	b	r
0	0	1	1	3	TC_3 (ETX)	DC_3	£(#) ⊙	3	C	S	c	s
0	1	0	0	4	TC_4 (EOT)	DC_4	$(¤) ⊙	4	D	T	d	t
0	1	0	1	5	TC_5 (ENQ)	TC_8 (NAK)	%	5	E	U	e	u
0	1	1	0	6	TC_6 (ACK)	TC_9 (SYN)	&	6	F	V	f	v
0	1	1	1	7	BEL	TC_{10} (ETB)	' ⊙	7	G	W	g	w
1	0	0	0	8	FE_0 (BS)	CAN	(	8	H	X	h	x
1	0	0	1	9	FE_1 (HT)	EM	)	9	I	Y	i	y
1	0	1	0	10	FE_2 (LF)⊙	SUB	*	:	J	Z	j	z
1	0	1	1	11	FE_3 (VT)⊙	ESC	+	;	K	⊙	k	⊙
1	1	0	0	12	FE_4 (FF)⊙	IS_4 (FS)	, ⊙	<	L	⊙	l	⊙
1	1	0	1	13	FE_5 (CR)⊙	IS_3 (GS)	–	=	M	⊙	m	⊙
1	1	1	0	14	SO	IS_2 (RS)	.	>	N	^ ⊙⊙	n	– ⊙⊙
1	1	1	1	15	SI	IS_1 (US)	/	?	O	_	o	DEL

536

NOTES ABOUT TABLE 1

① The format effectors are intended for equipment in which horizontal and vertical movements are effected separately. If equipment requires the action of CARRIAGE RETURN to be combined with a vertical movement, the format effector for that vertical movement may be used to effect the combined movement. For example, if NEW LINE (symbol NL, equivalent to CR + LF) is required, FE_2 shall be used to represent it. This substitution requires agreement between the sender and the recipient of the data.

The use of these combined functions may be restricted for international transmission on general switched telecommunication networks (telegraph and telephone networks).

② The symbol £ is assigned to position 2/3 and the symbol $ is assigned to position 2/4. In a situation where there is no requirement for the symbol £ the symbol # (number sign) may be used in position 2/3. Where there is no requirement for the symbol $ the symbol ¤ (currency sign) may be used in position 2/4. The chosen allocations of symbols to these positions for international information interchange shall be agreed between the interested parties. It should be noted that, unless otherwise agreed between sender and recipient, the symbols £, $ or ¤ do not designate the currency of a specific country.

③ National use positions. The allocations of characters to these positions lies within the responsibility of national standardization bodies. These positions are primarily intended for alphabet extensions. If they are not required for that purpose, they may be used for symbols.

④ Positions 5/14, 6/0 and 7/14 are provided for the symbols UPWARD ARROW HEAD, GRAVE ACCENT and OVERLINE. However, these positions may be used for other graphical characters when it is necessary to have 8, 9 or 10 positions for national use.

⑤ Position 7/14 is used for the graphic character ‾ (OVERLINE), the graphical representation of which may vary according to national use to represent ~(TILDE) or another diacritical sign provided that there is no risk of confusion with another graphic character included in the table.

⑥ The graphic characters in positions 2/2, 2/7, 2/12 and 5/14 have respectively the significance of QUOTATION MARK, APOSTROPHE, COMMA and UPWARD ARROW HEAD; however, these characters take on the significance of the diacritical signs DIAERESIS, ACUTE ACCENT, CEDILLA and CIRCUMFLEX ACCENT when they are preceded or followed by the BACKSPACE character (0/8).

E37

4 LEGEND

4.1 Control characters

Abbreviation	Note	Meaning	Position in the code table
ACK		Acknowledge	0/6
BEL		Bell	0/7
BS		Backspace	0/8
CAN		Cancel	1/8
CR	1	Carriage Return	0/13
DC		Device control	—
DEL		Delete	7/15
DLE		Data Link Escape	1/0
EM		End of Medium	1/9
ENQ		Enquiry	0/5
EOT		End of Transmission	0/4
ESC		Escape	1/11
ETB		End of Transmission Block	1/7
ETX		End of Text	0/3
FE		Format Effector	—
FF	1	Form Feed	0/12
FS		File Separator	1/12
GS		Group Separator	1/13
HT		Horizontal Tabulation	0/9
IS		Information Separator	—
LF	1	Line Feed	0/10
NAK		Negative Acknowledge	1/5
NUL		Null	0/0
RS		Record Separator	1/14
SI		Shift-In	0/15
SO		Shift-Out	0/14
SOH		Start of Heading	0/1
SP		Space (see 7.2)	2/0
STX		Start of Text	0/2
SUB		Substitute Character	1/10
SYN		Synchronous Idle	1/6
TC		Transmission Control	—
US		Unit Separator	1/15
VT	1	Vertical Tabulation	0/11

4.2 Graphic characters

Graphic	Note	Name	Position in the code table
(space)		Space (see 7.2)	2/0
!		Exclamation mark	2/1
"	6	Quotation mark, Diaeresis	2/2
£	2	Pound sign	2/3
#	2	Number sign	2/3
$	2	Dollar sign	2/4
¤	2	Currency sign	2/4
%		Percent sign	2/5
&		Ampersand	2/6
'	6	Apostrophe, acute accent	2/7
(		Left parenthesis	2/8
)		Right parenthesis	2/9
*		Asterisk	2/10
+		Plus sign	2/11
,	6	Comma, Cedilla	2/12
-		Hyphen, Minus sign	2/13
.		Full stop (period)	2/14
/		Solidus	2/15
:		Colon	3/10
;		Semi-colon	3/11
<		Less than sign	3/12
=		Equals sign	3/13
>		Greater than sign	3/14
?		Question mark	3/15
^	4, 6	Upward arrow head, Circumflex accent	5/14
_		Underline	5/15
`	4	Grave accent	6/0
~	4, 5	Overline, Tilde	7/14

5 EXPLANATORY NOTES

5.1 Numbering of the positions in Table 1

Within any one character the bits are identified by b_7, $b_6 \ldots b_1$, where b_7 is the highest order, or most significant bit, and b_1 is the lowest order, or least significant bit.

If desired, these may be given a numerical significance in the binary system, thus :

Bit identification :	b_7	b_6	b_5	b_4	b_3	b_2	b_1
Significance :	64	32	16	8	4	2	1

In the table the columns and rows are identified by numbers written in binary and decimal notations.

Any one position in the table may be identified either by its bit pattern, or by its column and row numbers. For instance, the position containing the digit 1 may be identified :

- by its bit pattern in order of decreasing significance, i.e. 011 0001;
- by its column and row numbers, i.e. 3/1.

The column number is derived from bits b_7, b_6 and b_5 giving them weights of 4, 2 and 1 respectively. The row number is derived from bits b_4, b_3, b_2 and b_1 giving them weights of 8, 4, 2 and 1 respectively.

5.2 Diacritical signs

In the 7-bit character set, some printing symbols may be designed to permit their use for the composition of accented letters when necessary for general interchange of information. A sequence of three characters, comprising a letter, "backspace" and one of these symbols, is needed for this composition; the symbol is then regarded as a diacritical sign. It should be noted that these symbols take on their diacritical significance when they are preceded or followed by one "backspace" character; for example the symbol corresponding to the code combination 2/7 normally has the significance of "apostrophe", but becomes the diacritical sign "acute accent" when it precedes or follows a "backspace" character.

In order to increase efficiency, it is possible to introduce accented letters (as single characters) in the positions marked by Note ③ in the code table. According to national requirements, these positions may contain special diacritical signs.

5.3 Names, meanings and fonts of graphic characters

This International Standard assigns at least one name to denote each of the graphic characters displayed in Tables 1 and 2. The names chosen to denote graphic characters are intended to reflect their customary meanings. However, this International Standard does not define and does not restrict the meanings of graphic characters. In addition, it does not specify a particular style or font design for the graphic characters.

Under the provision of Note ③ of Table 1, graphic characters which are different from the characters of the international reference version may be assigned to the national use positions. When such assignments are made, the graphic characters shall have distinct forms and be given distinctive names which are not in conflict with any of the forms or the names of any of the graphic characters in the international reference version.

5.4 Uniqueness of character allocation

A character allocated to a position in Table 1 may not be placed elsewhere in the table. For example, in the case of position 2/3 the character not used cannot be placed elsewhere. In particular the POUND sign (£) can never be represented by the bit combination of position 2/4.

6 VERSIONS OF TABLE 1

6.1 General

6.1.1 In order to use Table 1 for information interchange, it is necessary to exercise the options left open, i.e. those affected by Notes ② to ⑤. A single character must be allocated to each of the positions for which this freedom exists or it must be declared to be unused. A code table completed in this way is called a "version".

6.1.2 The notes to Table 1, the explanatory notes and the legend apply in full to any version.

6.2 International reference version

This version is available for use when there is no requirement to use a national or an application-oriented version. In international information processing interchange the international reference version (Table 2) is assumed unless a particular agreement exists between sender and recipient of the data. The following characters are allocated to the optional positions of Table 1 :

#	Number sign	2/3
¤	Currency sign	2/4
@	Commercial at	4/0
[	Left square bracket	5/11
\	Reverse solidus	5/12
]	Right square bracket	5/13
{	Left curly bracket	7/11
|	Vertical line	7/12
}	Right curly bracket	7/13

It should be noted that no substitution is allowed when using the international reference version.

TABLE 2 — International reference version

				b_7	0	0	0	0	1	1	1	1
				b_6	0	0	1	1	0	0	1	1
				b_5	0	1	0	1	0	1	0	1
				column / row	0	1	2	3	4	5	6	7
b_4	b_3	b_2	b_1									
0	0	0	0	0	NUL	TC_7 (DLE)	SP	0	@	P	`	p
0	0	0	1	1	TC_1 (SOH)	DC_1	!	1	A	Q	a	q
0	0	1	0	2	TC_2 (STX)	DC_2	"	2	B	R	b	r
0	0	1	1	3	TC_3 (ETX)	DC_3	#	3	C	S	c	s
0	1	0	0	4	TC_4 (EOT)	DC_4	¤	4	D	T	d	t
0	1	0	1	5	TC_5 (ENQ)	TC_8 (NAK)	%	5	E	U	e	u
0	1	1	0	6	TC_6 (ACK)	TC_9 (SYN)	&	6	F	V	f	v
0	1	1	1	7	BEL	TC_{10} (ETB)	'	7	G	W	g	w
1	0	0	0	8	FE_0 (BS)	CAN	(	8	H	X	h	x
1	0	0	1	9	FE_1 (HT)	EM	)	9	I	Y	i	y
1	0	1	0	10	FE_2 (LF)	SUB	*	:	J	Z	j	z
1	0	1	1	11	FE_3 (VT)	ESC	+	;	K	[	k	{
1	1	0	0	12	FE_4 (FF)	IS_4 (FS)	,	<	L	\	l	|
1	1	0	1	13	FE_5 (CR)	IS_3 (GS)	-	=	M	]	m	}
1	1	1	0	14	SO	IS_2 (RS)	.	>	N	^	n	~
1	1	1	1	15	SI	IS_1 (US)	/	?	O	_	o	DEL

6.3 National versions

6.3.1 The responsibility for defining national versions lies with the national standardization bodies. These bodies shall exercise the options available and make the required selection.

6.3.2 If so required, more than one national version can be defined within a country. The different versions shall be separately identified. In particular when for a given national use position, for example, 5/12 or 6/0, alternative characters are required, two different versions shall be identified, even if they differ only by this single character.

6.3.3 If there is in a country no special demand for specific characters, it is strongly recommended that the characters of the international reference version be allocated to the same national use positions.

6.4 Application-oriented versions

Within national or international industries, organizations or professional groups, application-oriented versions can be used. They require precise agreement among the interested parties, who will have to exercise the options available and to make the required selection.

7 FUNCTIONAL CHARACTERISTICS RELATED TO CONTROL CHARACTERS

Some definitions in this section are stated in general terms and more explicit definitions of use may be needed for specific implementation of the code table on recording media or on transmission channels. These more explicit definitions and the use of these characters are the subject of other ISO Publications.

7.1 General designations of control characters

The general designation of control characters involves a specific class name followed by a subscript number.

They are defined as follows :

TC Transmission control characters

Control characters intended to control or facilitate transmission of information over telecommunication networks.

The use of the TC characters on the general telecommunication networks is the subject of other ISO publications.

The transmission control characters are :

ACK, DLE, ENQ, EOT, ETB, ETX, NAK, SOH, STX and SYN.

FE Format effectors

Control characters mainly intended for the control of the layout and positioning of information on printing and/or display devices. In the definitions of specific format effectors, any reference to printing devices should be interpreted as including display devices.

The definitions of format effectors use the following concept :

a) a page is composed of a number of lines of characters;

b) the characters forming a line occupy a number of positions called character positions;

c) the active position is that character position in which the character about to be processed would appear, if it were to be printed. The active position normally advances one character position at a time.

The format effector characters are : BS, CR, FF, HT, LF and VT (see also Note 1).

DC Device control characters

Control characters for the control of a local or remote ancillary device (or devices) connected to a data processing and/or telecommunication system. These control characters are not intended to control telecommunication systems; this should be achieved by the use of TCs.

Certain preferred uses of the individual DCs are given in 7.2.

IS Information separators

Control characters that are used to separate and qualify data logically. There are four such characters. They may be used either in hierarchical order or non-hierarchically; in the latter case their specific meanings depend on their applications.

When they are used hierarchically, the ascending order is :

US, RS, GS, FS.

In this case, data normally delimited by a particular separator cannot be split by a higher order separator but will be considered as delimited by any higher order separator.

7.2 Specific control characters

Individual members of the classes of controls are sometimes referred to by their abbreviated class name and a subscript number (for example, TC_5) and sometimes by a specific name indicative of their use (for example, ENQ).

Different but related meanings may be associated with some of the control characters but in an interchange of data this normally requires agreement between the sender and the recipient.

ACK Acknowledge

A transmission control character transmitted by a receiver as an affirmative response to the sender.

BEL Bell

A control character that is used when there is a need to call for attention; it may control alarm or attention devices.

BS Backspace

A format effector which moves the active position one character position backwards on the same line.

CAN Cancel

A character, or the first character of a sequence, indicating that the data preceding it is in error. As a result, this data is to be ignored. The specific meaning of this character must be defined for each application and/or between sender and recipient.

CR Carriage return

A format effector which moves the active position to the first character position on the same line.

Device controls

DC_1 A device control character which is primarily intended for turning on or starting an ancillary device. If it is not required for this purpose, it may be used to restore a device to the basic mode of operation (see also DC_2 and DC_3), or for any other device control function not provided by other DCs.

DC_2 A device control character which is primarily intended for turning on or starting an ancillary device. If it is not required for this purpose, it may be used to set a device to a special mode of operation (in which case DC_1 is used to restore the device to the basic mode), or for any other device control function not provided by other DCs.

DC_3 A device control character which is primarily intended for turning off or stopping an ancillary device. This function may be a secondary level stop, for example wait, pause, stand-by or halt (in which case DC_1 is used to restore normal operation). If it is not required for this purpose, it may be used for any other ancillary device control function not provided by other DCs.

DC_4 A device control character which is primarily intended for turning off, stopping or interrupting an ancillary device. If it is not required for this purpose, it may be used for any other device control function not provided by other DCs.

Examples of use of the device controls:

1) One switching		
	on—DC_2	off—DC_4
2) Two independent switchings		
first one	on—DC_2	off—DC_4
second one	on—DC_1	off—DC_3
3) Two dependent switchings		
general	on—DC_2	off—DC_4
particular	on—DC_1	off—DC_3
4) Input and output switching		
output	on—DC_2	off—DC_4
input	on—DC_1	off—DC_3

DEL Delete

A character used primarily to erase or obliterate an erroneous or unwanted character in punched tape. DEL characters may also serve to accomplish media-fill or time-fill. They may be inserted into or removed from a stream of data without affecting the information content of that stream but then the addition or removal of these characters may affect the information layout and/or the control of equipment.

DLE Data link escape

A transmission control character which will change the meaning of a limited number of contiguously following characters. It is used exclusively to provide supplementary data transmission control functions. Only graphic characters and transmission control characters can be used in DLE sequences.

EM End of medium

A control character that may be used to identify the physical end of a medium, or the end of the used portion of a medium, or the end of the wanted portion of data recorded on a medium. The position of this character does not necessarily correspond to the physical end of the medium.

ENQ Enquiry

A transmission control character used as a request for a response from a remote station — the response may include station identification and/or station status. When a "Who are you" function is required on the general switched transmission network, the first use of ENQ after the connection is established shall have the meaning "Who are you" (station identification). Subsequent use of ENQ may, or may not, include the function "Who are you", as determined by agreement.

EOT End of transmission

A transmission control character used to indicate the conclusion of the transmission of one or more texts.

ESC Escape

A control character which is used to provide additional control functions. It alters the meaning of a limited number of contiguously following bit combinations. The use of this character is specified in ISO 2022.

ETB End of transmission block

A transmission control character used to indicate the end of a transmission block of data where data is divided into such blocks for transmission purposes.

ETX End of text

A transmission control character which terminates a text.

FF Form feed

A format effector which advances the active position to the same character position on a pre-determined line of the next form or page.

HT Horizontal tabulation

A format effector which advances the active position to the next pre-determined character position on the same line.

Information separators

IS_1 (US) A control character used to separate and qualify data logically; its specific meaning has to be defined for each application. If this character is used in hierarchical order as specified in the general definition of IS, it delimits a data item called a UNIT.

IS_2 (RS) A control character used to separate and qualify data logically; its specific meaning has to be defined for each application. If this character is used in hierarchical order as specified in the general definition of IS, it delimits a data item called a RECORD.

IS_3 (GS) A control character used to separate and qualify data logically; its specific meaning has to be defined for each application. If this character is used in hierarchical order as specified in the general definition of IS, it delimits a data item called a GROUP.

IS_4 (FS) A control character used to separate and qualify data logically; its specific meaning has to be defined for each application. If this character is used in hierarchical order as specified in the general definition of IS, it delimits a data item called a FILE.

LF Line feed

A format effector which advances the active position to the same character position of the next line.

NAK Negative acknowledge

A transmission control character transmitted by a receiver as a negative response to the sender.

NUL Null

A control character used to accomplish media-fill or time-fill. NUL characters may be inserted into or removed from a stream of data without affecting the information content of that stream; but then the addition or removal of these characters may affect the information layout and/or the control of equipment.

SI Shift-in

A control character which is used in conjunction with SHIFT-OUT and ESCAPE to extend the graphic character set of the code. It may reinstate the standard meanings of the bit combinations which follow it. The effect of this character when using code extension techniques is described in ISO 2022.

SO Shift-out

A control character which is used in conjunction with SHIFT-IN and ESCAPE to extend the graphic character set of the code. It may alter the meaning of the bit combinations of columns 2 to 7 which follow it until a SHIFT-IN character is reached. However, the characters SPACE (2/0) and DELETE (7/15) are unaffected by SHIFT-OUT. The effect of this character when using code extension techniques is described in ISO 2022.

SOH Start of heading

A transmission control character used as the first character of a heading of an information message.

SP Space

A character which advances the active position one character position on the same line.

This character is also regarded as a non-printing graphic.

STX Start of text

A transmission control character which precedes a text and which is used to terminate a heading.

SUB Substitute character

A control character used in the place of a character that has been found to be invalid or in error. SUB is intended to be introduced by automatic means.

SYN Synchronous idle

A transmission control character used by a synchronous transmission system in the absence of any other character (idle condition) to provide a signal from which synchronism may be achieved or retained between data terminal equipment.

VT Vertical tabulation

A format effector which advances the active position to the same character position on the next pre-determined line.

544

INTERNATIONAL STANDARD ISO 1155

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION•МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ•ORGANISATION INTERNATIONALE DE NORMALISATION

Information processing — Use of longitudinal parity to detect errors in information messages

Traitement de l'information — Emploi de la parité longitudinale pour la détection d'erreurs dans les messages d'information

Second edition — 1978-11-15

545

UDC 681.3.042.4

Ref. No. ISO 1155-1978 (E)

Descriptors : data processing, error detecting codes, parity check, block check.

Price based on 1 page

FOREWORD

ISO (the International Organization for Standardization) is a worldwide federation of national standards institutes (ISO member bodies). The work of developing International Standards is carried out through ISO technical committees. Every member body interested in a subject for which a technical committee has been set up has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work.

Draft International Standards adopted by the technical committees are circulated to the member bodies for approval before their acceptance as International Standards by the ISO Council.

International Standard ISO 1155 was developed by Technical Committee ISO/TC 97, *Computers and information processing*.

This second edition was submitted directly to the ISO Council, in accordance with clause 6.13.1 of the Directives for the technical work of ISO. It cancels and replaces the first edition (i.e. ISO 1155-1973), which had been approved by the member bodies of the following countries :

Australia	Germany, F.R.	Romania
Belgium	Greece	Spain
Brazil	Israel	Sweden
Canada	Italy	Switzerland
Czechoslovakia	Japan	Thailand
Denmark	New Zealand	United Kingdom
Egypt, Arab Rep. of	Peru	U.S.A.
France	Poland	U.S.S.R.

No member body had expressed disapproval of the document.

546

Printed in Switzerland

Information processing — Use of longitudinal parity to detect errors in information messages

0 INTRODUCTION

In data communication systems the information formats and the redundancy in the data to be transmitted differ widely from one application to another. It is therefore clear that a number of classes of error protection systems may be required.

This International Standard defines one method of error detection which satisfies a wide range of applications. It consists of accompanying the data block or text by one checking character (in addition to character parity) and it is often referred to as the "Longitudinal Parity Method".

1 SCOPE AND FIELD OF APPLICATION

This International Standard specifies a method for detecting errors in information messages by attaching one block check character to the transmitted information block (or text) and checking this character when it is received. The method of correcting errors when they are detected is subject to the particular application and is performed at a higher level.

The method is applicable to systems which use the 7-bit coded character set which is the subject of ISO 646, *7-bit coded character set for information processing interchange*, and the basic mode of implementing this 7-bit code in data communication systems, which is the subject of ISO 1745, *Information processing — Basic mode control procedures for data communication systems*.

The rules for generating the character parity bits, according to ISO 1177, *Information processing — Character structure for start/stop and synchronous transmission*, are that the character parity sense shall be odd in synchronous systems and even in asynchronous systems.

2 RULES FOR GENERATING THE LONGITUDINAL PARITY BLOCK CHECK CHARACTER

2.1 Block check character

2.1.1 The block check character shall be composed of 7 bits plus a parity bit.

2.1.2 Each of the first 7 bits of the block check character shall be the modulo 2 binary sum of every element in the same bit 1 to bit 7 column of the successive character of the transmitted block.

2.1.3 The longitudinal parity of each column of the block, including the block check character, shall be even.

2.1.4 The sense of the parity bit of the block check character shall be the same as for the information characters (odd for synchronous transmission, even for asynchronous transmission).

2.2 Summation

2.2.1 The summation to obtain the block check character shall be started by the first appearance of either SOH (Start of Heading) or STX (Start of Text).

2.2.2 The starting character shall not be included in the summation.

2.2.3 If an STX character appears after the summation has been started by SOH, then the STX character shall be included in the summation as if it were a text character.

2.3 With the exception of SYN (Synchronous Idle), all the characters which are transmitted after the start of the block check summation shall be included in the summation, including the ETB (End of Transmission/Block) or ETX (End of Text) control character which signals that the next following character is the block check character.

2.4 No character, SYN or otherwise, shall be inserted between the ETB or ETX character and the block check character.

547

INTERNATIONAL STANDARD ISO 1177

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION

Information processing — Character structure for start/stop and synchronous transmission

First edition — 1973-11-01

UDC 681.3.042

Ref. No. ISO 1177-1973 (E)

Descriptors : data processing, data transmission, start-stop data transmission, synchronous data transmission, character signal.

Price based on 4 pages

FOREWORD

ISO (the International Organization for Standardization) is a worldwide federation of national standards institutes (ISO member bodies). The work of developing International Standards is carried out through ISO technical committees. Every member body interested in a subject for which a technical committee has been set up has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work.

Draft International Standards adopted by the technical committees are circulated to the member bodies for approval before their acceptance as International Standards by the ISO Council.

Prior to 1972, the results of the work of the Technical Committees were published as ISO Recommendations; these documents are now in the process of being transformed into International Standards. As part of this process, Technical Committee ISO/TC 97, *Computers and information processing*, has reviewed ISO Recommendation R 1177-1970 and found it technically suitable for tansformation. International Standard ISO 1177 therefore replaces ISO Recommendation R 1177-1970, which was approved by the Member Bodies of the following countries :

Australia	Greece	Sweden
Belgium	Israel	Switzerland
Brazil	Italy	Thailand
Canada	Japan	Turkey
Czechoslovakia	New Zealand	United Kingdom
Denmark	Poland	U.S.A.
France	Romania	U.S.S.R.
Germany	Spain	

No Member Body expressed disapproval of the Recommendation.

Printed in Switzerland

Information processing — Character structure for start/stop and synchronous transmission

1 SCOPE AND FIELD OF APPLICATION

This International Standard specifies the character structure to be used for serial-by-bit start/stop and synchronous data transmission systems using the 7-bit coded character set which is the subject of ISO 646, *7-bit coded character set for information processing interchange*[1].

It applies to the information transfer through the interface standardized by the CCITT and the IEC/ISO between the data terminal equipment and data communications equipment as defined in CCITT Recommendation V 24, and the relevant CCITT modem recommendations.

2 BIT SEQUENCING — START/STOP AND SYNCHRONOUS OPERATION

In serial working data transmission systems, the chronological order of transmission of the information bits shall correspond to the bit identification b_1 to b_7 as defined in the 7-bit code table of ISO 646 with least significant bit transmitted first.

When the rank in the combination represents the order of the bit in binary numbering, the bits shall be transmitted in serial, working with the low order bit first.

The numerical meaning corresponding to each information bit considered in isolation is that of the digit :

0 for a unit corresponding to condition A (Travail = Space), and

1 for a unit corresponding to condition Z (Repos = Mark),

in accordance with the definitions of these conditions for two-condition transmission systems.

3 PARITY BIT — START/STOP AND SYNCHRONOUS OPERATION

A parity bit is added to every character and is located in the eight position, b_8, and is therefore transmitted after the seven signifiant bits for the character.

4 PARITY SENSE — START/STOP AND SYNCHRONOUS OPERATION

For asynchronous systems, the parity bit is chosen in such a way that the number of "ONE" bits is even in the sequence of eight bits thus formed.

For synchronous systems, the parity bit is chosen in such a way that the number of "ONE" bits is odd in the sequence of eight bits thus formed.

5 CHARACTER FRAMING

5.1 Start/Stop operation

In start/stop systems using the 7-bit coded character set (see ISO 646), ten or eleven unit elements shall be used per character.

The first information bit of the transmitted coded combinations shall be preceded by a start element corresponding to condition A (Travail = Space). The duration of this start element shall be one unit interval at the data signalling rate at the transmitting interface.

1) This character set is also standardized by CCITT : International Telegraphic Alphabet No. 5 Recommendation V 3.

The combination of seven information elements completed by its parity element shall be followed by a stop element corresponding to condition Z (Repos = Mark).

For systems using the 7-bit coded character set over the general switched telephone and telegraph networks with electromechanical data terminal equipment operating at modulation rates up to and including 200 bauds, the stop element duration at the transmitter shall be TWO unit intervals at the data signalling rate of the transmitter.

In other cases the use of a stop element with a duration of ONE unit interval is preferable. However, this is subject to mutual agreement between the administrations and/or recognized private operating agencies concerned.

Similar situations when a ONE unit interval stop element can be used may apply to leased circuits.

The start/stop receivers should be capable of correctly receiving start/stop signals comprising a single-unit stop element, whose duration will be reduced by a time interval equal to the deviation corresponding to the degree of gross start/stop distortion permitted at the receiver input. However, for electromechanical equipment which must use a two-unit stop element (11-unit alphabet) with a modulation rate of 200 bauds or less, receivers should be capable of correctly receiving signals with a stop element reduced to one unit.

The time between the end of the stop element of a character and the beginning of the start element of the next character may be of any duration; the polarity of the signal during this time is the same as that of the stop element.

5.2 Synchronous operation

In synchronous systems using the 7-bit coded character set (see ISO 646), eight bits per character shall be used : the seven information bits followed by the parity bit.

The time between the end of the last bit of a character and the beginning of the first bit of the next character shall be zero time or a multiple of the unit interval at the data signalling rate of the transmitter. When character synchronism must be maintained, this time interval shall be zero or a multiple of the character interval. Where necessary, parity sense should be maintained.

551

INTERNATIONAL STANDARD ISO 1745

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION · МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ · ORGANISATION INTERNATIONALE DE NORMALISATION

Information processing — Basic mode control procedures for data communication systems

Traitement de l'information — Procédures de commande pour transmission de données en mode de base

First edition — 1975-02-01

UDC 681.327.18.01 Ref. No. ISO 1745-1975 (E)

Descriptors : data processing, data transmission, control procedures, control characters.

Price based on 19 pages

FOREWORD

ISO (the International Organization for Standardization) is a worldwide federation of national standards institutes (ISO Member Bodies). The work of developing International Standards is carried out through ISO Technical Committees. Every Member Body interested in a subject for which a Technical Committee has been set up has the right to be represented on that Committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work.

Draft International Standards adopted by the Technical Committees are circulated to the Member Bodies for approval before their acceptance as International Standards by the ISO Council.

International Standard ISO 1745 was drawn up by Technical Committee ISO/TC 97, *Computers and information processing*, and circulated to the Member Bodies in May 1973.

It has been approved by the Member Bodies of the following countries :

Australia	Japan	Switzerland
Brazil	Netherlands	Thailand
Canada	New Zealand	Turkey
Czechoslovakia	Poland	United Kingdom
Egypt, Arab Rep. of	Romania	U.S.A.
France	South Africa, Rep. of	U.S.S.R.
Hungary	Spain	Yugoslavia
Italy	Sweden	

The Member Bodies of the following countries expressed disapproval of the document on technical grounds :

Bulgaria
Germany

This International Standard cancels and replaces ISO Recommendation R 1745-1971, of which it constitutes a technical revision.

553

Printed in Switzerland

CONTENTS

554

Information processing — Basic mode control procedures for data communication systems

0 INTRODUCTION

0.1 General

A data communication system may be considered as the ensemble of the terminal installations and the interconnecting network that permits information to be exchanged.

A data link concept is identifiable when considering terminal installations connected to the same network, operating at the same speed, in the same code. Whenever actions on the respective transmission control characters take place, a separation of data links is constituted. Typical examples where this applies are : store and forward switching centres, concentrators, intermediate reformatting and speed-change devices.

The information transfer in a data link is monitored by data link control procedures where some characters, selected within a code set, are given particular meanings according to the transmission phase and are used for various purposes such as to delineate information, to reverse the direction of transmission, to ask questions, to answer, etc.

The data link control procedures are categorized in classes which are referred to as modes of operation. The present considerations relate to one class called "basic mode", which is defined as follows :

> In the basic mode all the necessary transmission control information (for example, message framing and supervisory instructions) passing from one station to another is carried over the link by discrete control characters selected from the ten transmission control characters which are defined in the ISO/CCITT 7-bit code (ISO 646). The information exchanges are carried out in the alternate mode on standard communication facilities. The control of the data link is not affected by any characters other than the ten transmission control characters. Other codes than the ISO/CCITT code may therefore be transmitted provided that they do not contain any of the ten transmission control characters in either heading or text. Sequences of transmission control character combinations such as DLE.XXX are not permitted, with the one exception DLE.EOT which is defined as "Disconnect".

Extensions to the basic mode are contained in the following International Standards :

ISO 2111, *Basic mode control procedures — Code independent information transfer;*

ISO 2628, *Basic mode control procedures — Complements;*

ISO 2629, *Basic mode control procedures — Conversational information message transfer;*

and also in annexes B and C of this International Standard.

The following considerations have been taken into account in developing the rules for the basic mode :

The rules are based on the assumption that one of the stations in each connection would be either a computer or a device capable of handling automatically an exchange of information. The rules are designed to allow the complexity of operation to be increased from a basic level by adding options. These options are designed so that any number of stations can still communicate even though they normally operate at different levels of complexity.

It is desirable to reduce optional features in this International Standard to a minimum, but still retain a balance between an economic solution for the "low cost systems" and extendability for encompassing more complex systems. The rules may be difficult to implement in very simple systems involving low cost devices and human control. On the other hand, in complex high speed computer links, the rules may seriously restrict the throughput of information. These two cases are regarded as the upper and lower fringes of the present International Standard and may be the subject of future International Standards.

With the above considerations, typical limitations of basic mode control procedures are :

- restriction of efficiency by the time delay which is due to the alternate mode of operation;
- single link operation only.

555

0.2 Communication phases

The table below shows the various possible phases and sub-phases of a data communication.

Phases 1 and 5, which relate to the establishment and clearing of connections over the general switched network, are under the responsibility of the CCITT and are therefore not covered by this International Standard.

In each phase, one of the stations directs the operation and is responsible for the continuity of the communication. The other station or stations only react to the actions of the responsible station.

The transmission control characters which are shown alongside the various sub-phases are those which are involved in the basic mode of operation.

EOT is shown in parentheses in Phases 2 and 3 because its use within the phases initiates a changeover to Phase 4.

1 SCOPE AND FIELD OF APPLICATION

1.1 General

This International Standard specifies the method of implementation of the ISO/CCITT 7-bit coded character set[1] for information interchange on data transmission channels. It also defines the formats of the transmitted messages and the supervisory sequences which are part of the transmission control procedures. It covers the majority of existing data transmission systems and data link configurations used in conjunction with data processing systems.

These control procedures deal with transmission over one link at a time and do not describe the operation of data links in "tandem". They relate to the class of control procedures which is known as the basic mode and apply at the interface between data communication equipment and data terminal equipment.

Table of phases

Phase		Function: Action / Reaction	Station's name: Responsible / Responsive	Transmission control characters used in basic mode: Forward / Backward	Notes
1 Establishment of connection over general network	a) Switching b) Identification				CCITT Responsibility
2 Establishment of data link	a) Switching	Call / Answer	Calling / Called		Not covered at present
	b) Polling	Poll / Reply	Control / Tributary	(EOT), ENQ / (EOT)	
	c) Selecting	Select / Reply	Master / Slave	(EOT), ENQ / ACK, NAK	
3 Information transfer		Transfer / Supervision	Master / Slave	SOH, STX ETB, ETX, (EOT) / ACK, NAK, (EOT)	
4 Termination	a) Return to neutral state	Terminate / Interrupt	Master / Slave	EOT / EOT	
	b) Return to control station	Terminate / Interrupt	Master / Slave	EOT / EOT	
	c) Disconnect	Disconnect / Disconnect	Master / Slave	DLE, EOT / DLE, EOT	
5 Clearing of connection					CCITT Responsibility

1) See ISO 646. CCITT : Alphabet No. 5.

It is accepted that, in their present form, the control procedures are a framework upon which a system can be built and that, before the successful interconnection of equipment from different supplies can be ensured, it will be necessary to define additional details, such as :

— structure of prefixes or addresses when used;

— "time-out" procedures and the recovery procedures which follow the various time-out conditions (see ISO 2628).

This International Standard must be considered in conjunction with the following ISO publications :

1) ISO 1177, *Information processing — Character structure for start/stop and synchronous transmission;*

2) ISO 1155, *Information processing — Use of longitudinal parity to detect errors in information messages.*

1.2 Assumptions

1) The information to be transmitted will normally be coded in accordance with the 7-bit ISO/CCITT code.

2) All transmission control functions will be performed by the use of ten specific transmission control characters which are defined in this code as TC 1 to TC 10.

3) No recommendation is made regarding

— the technique used (hardware or software);

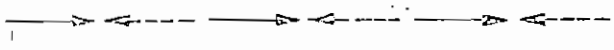
— the part of the terminal installation where the information messages and supervisory sequences are generated and recognized.

4) Transmission may be at any data transfer rate, either serial or parallel and either start/stop or synchronous.

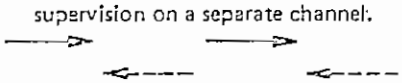
5) Responses to an information message or a supervisory sequence may be either by turn around of the channel or by using another channel.

6) The basic mode control procedures are applicable to systems of varied complexity based on either-way transmission using :

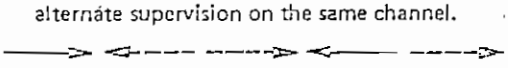
a) One-way transfer of information with alternate supervision on the same channel.



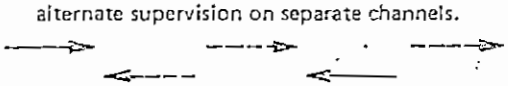
b) One-way transfer of information with alternate supervision on a separate channel.



c) Alternate two-way transfer of information with alternate supervision on the same channel.



d) Alternate two-way transfer of information with alternate supervision on separate channels.



7) The following cases will be the subject of further study :

a) One-way transfer of information with simultaneous supervision.

b) Alternate two-way transfer of information with simultaneous supervision.

c) Two-way simultaneous transfer of information with alternate supervision.

d) Two-way simultaneous transfer of information with simultaneous supervision.

2 DEFINITIONS OF THE TRANSMISSION CONTROL CHARACTERS

The basic definitions of the ten transmission control characters, as taken from ISO 646, are listed below (see clause 5 for description of use).

(TC1) SOH Start of heading

A transmission control character used as the first character of a heading of an information message.

(TC2) STX Start of text

A transmission control character which precedes a text and which is used to terminate a heading.

(TC3) ETX End of text

A transmission control character which terminates a text.

(TC4) EOT End of transmission

A transmission control character used to indicate the conclusion of the transmission of one or more texts.

(TC5) ENQ Inquiry

A transmission control character used as a request for a response from a remote station — the response may include station identification and/or station status. When a "Who are you" function is required on the general switched transmission network, the first use of ENQ after the connection is established shall have the meaning "Who are you" (station identification). Subsequent use of ENQ may, or may not, include the function "Who are you", as determined by agreement.

(TC6) ACK Acknowledge

A transmission control character transmitted by a receiver as an affirmative response to the sender.

(TC7) DLE Data link escape

A transmission control character which will change the meaning of a limited number of contiguously following characters. It is used exclusively to provide supplementary data transmission control functions. Only graphic characters and transmission control characters can be used in DLE sequences.

(TC8) NAK Negative acknowledge

A transmission control character transmitted by a receiver as a negative response to the sender.

(TC9) SYN Synchronous idle

A transmission control character used by a synchronous transmission system in the absence of any other character (idle condition) to provide a signal from which synchronism may be achieved or retained between data terminal equipment.

(TC10) ETB End of transmission block

A transmission control character used to indicate the end of a transmission block of data where data is divided into such blocks for transmission purposes.

3 MESSAGE FORMATS

The various possible messages are categorized as follows :

- information messages;
- forward supervisory sequences;
- backward supervisory sequences.

558

3.1 General rules

Every transferred sequence of characters contains at least one transmission control character. These are used either to define the nature of the information contained in a sequence of data or to convey supervisory functions.

- They must not be considered information. Therefore, they must not be transmitted as part of the text or heading of an information message with the exception of SYN which may be inserted as required but which must not be regarded as information.
- When used singly or at the end of a message or sequence, they invite the station receiving them to take action.

a) *Information messages*

Information messages consist of a text which can be preceded by a heading; the heading is delivered with the text. Routing indication, for intermediate points in particular, must be in the heading. Other auxiliary information may be either in the heading or in the text.

SOH, STX, ETB and ETX are used as information framing characters. They cannot be sent singly.

Information messages, or information blocks, may be accompanied by a block checking character in accordance with ISO 1155. The use of this block checking character, shown in parentheses, is optional and therefore subject to prior agreement.

b) *Supervisory sequences*

All supervisory sequences except DLE.EOT are composed of either a single transmission control character or a single transmission control character preceded by one or several graphics.

In some of the following supervisory sequences the meaning of the character or characters which precede the transmission control character is defined (for example Polling address). In others, it is simply shown as a prefix which may include one or more of the following :

- identity information;
- address information;
- status information;
- any other qualifier as necessary (for example response number).

The use of these prefixes and their description is subject to prior agreement. They may be standardized at a later date.

EOT, ENQ, ACK and NAK are used for supervision. They can never appear contiguously.

The prefix must not contain more than 15 characters.

3.2 Information messages

a) S T X — TEXT — E T X (B C C)

b) S T X — TEXT — E T B (B C C)

(See note 2, below)

c) S O H — HEADING — S T X — TEXT — E T X (B C C)

d) S O H — HEADING — S T X — TEXT — E T B (B C C)

(See note 2, below)

e) S O H — HEADING — E T B (B C C)

(See note 2, below)

NOTES

1 Fillers may be inserted in the heading and the text (for example SYN).

2 In formats b), d) and e) above which end with ETB, some continuation is required.

3 All the above messages can be aborted by terminating them at any point with EOT. Future study may lead to the specifying of another method for aborting which allows the continuation of the communication.

3.3 Forward supervisory sequences

a) Polling

Polling address — ENQ

(See note below)

b) Selecting

1) Station selection

Selecting address — ENQ

(See note below)

If a reply is not required, ENQ is not used and the selecting sequence is immediately followed by the information message.

2) Identification and status

(Prefix) — ENQ

3) Out of neutral

(Prefix) — ENQ

c) Return to control station — Return to neutral state

(Prefix) — EOT

(See note below)

d) Disconnect

(Prefix) — DLE EOT

NOTE — Polling sequences are always preceded by EOT except in systems involving Phase 1 where the omission of EOT is optional. Selecting sequences may also be preceded by EOT.

Some systems may not be able to tolerate a polling or selecting sequence immediately following EOT. In such cases it may be necessary to ensure a short delay between the EOT and the address by using, for example, a number of "filler" characters.

3.4 Backward supervisory sequences

a) Positive reply to :

— an information message

— selecting

(Prefix) — ACK

b) Negative reply to :

— an information message

(Prefix) — NAK

c) Negative reply to :

— a polling supervisory sequence

(Prefix) — EOT

d) Negative reply to :

— a selecting supervisory sequence

(Prefix) — NAK

e) Request for :

— an interruption

(Prefix) — EOT

— a return of responsibility to the control station

(Prefix) — EOT

— return to neutral state

(Prefix) — EOT

(See note 2, below)

f) Disconnect

(Prefix) — DLE EOT

NOTES

1 The procedures for the cases of "no reply" are covered in 4.3.

2 Future study may lead to replacing the interruption by EOT with another method.

4 DESCRIPTION OF PHASES

The operational procedures of a complete system can be constructed from the following separate phases and sub-phases :

Phase 1[1]. Establishment of connection over the general network

a) Switching

b) Identification

1) This phase is under the responsibility of the CCITT.

Phase 2 Establishment of data link

a) Switching

b) Polling

c) Selecting

Phase 3 Information transfer

Phase 4 Termination

a) Return to neutral state

b) Return to control station

c) Disconnect

Phase 5[1] Clearing of connection

4.1 Phase linkage

Figure 1 represents the various phases of a communication which are linked (thick lines) to achieve one transmission, or information transfer, in the most general case encompassed by the basic mode control procedures.

The sequence of events for such a communication would be as follows :

Phase 1 a), b) Establishment of connection over the general switched network

Here the connection is established by the telecommunications administration and this is likely to be divided into two sub-phases : "Switching" and "Identification". They will both be under the responsibility of the administration.

Unless otherwise stipulated by the administration, once this phase is achieved, the calling station takes on the responsibility for the communication and acts as master station or control station.

Means for signalling the completion of Phase 1 will be defined with reference to Recommendations on interfaces (for example, CCITT-V24, Circuit 107).

Phase 2 a), b), c) Establishment of data link

After establishing the Connection on the general network it is required to establish the data link. This procedure may involve some private line Switching performed by a private Switching exchange or a line concentrator before polling and selecting.

The "Polling" procedure, carried out by the control station, invites a tributary station to transmit any message it may have.

This procedure transfers the responsibility of the communication to the polled station, which takes the status of master station.

The "Selecting" procedure, carried out by the so-designated master station, invites in turn another station to get ready to receive an information message.

This procedure gives to the selected station the status of slave station.

Phase 3 Information transfer

Assuming the slave station(s) has accepted to receive the information message, the master station commences its transmission.

During this phase there are no changes of station status or responsibility.

Phase 4 a), b), c) Termination

When the information message has been transmitted and satisfactorily received by the slave station(s), the master station sends EOT to announce to the control station that its transmission requirement has temporarily ceased. By doing so, the master station relinquishes its master status and returns the responsibility of the communication to the control station.

If there are no further transmission requirements, the control station, by sending DLE.EOT, releases the possibly involved private switching equipment.

Phase 5 Clearing of connection (General network)

The Disconnect function (DLE.EOT) of the termination phase will initiate the clearing of the connection over the general switched network. The procedure for so doing is under the responsibility of the administration.

As a matter of fact, in most systems, several data link establishments and several information transfers take place in sequence within a communication.

This is illustrated by the phase linkage arrows marked PL 1, 2,...,6. An example of such a multiple communication could be :

Phase 1 a), b) — We reach a multistation link via the general network;

Phase 2 c) — We try to select station X;

Phase 4 a) — Station X refuses to receive;

Phase 2 b) — We poll station Y;

Phase 3 — Station Y transmits information to us;

Phase 4 — Station Y terminates its transmission;

Phase 4 c) — We decide to disconnect;

Phase 5 — The general network is cleared.

All the permitted phase linkages are shown on the phase diagrams in 4.2, along with more detailed descriptions of the phases and sub-phases.

In some systems, not all the phases or sub-phases shown on the phase linkage diagram will be required. Examples are illustrated by different by-passes :

By-pass 1 (BP1)

This by-pass applies to systems composed entirely of leased or private circuits not connected to the general switched network.

1) This phase is under the responsibility of the CCITT.

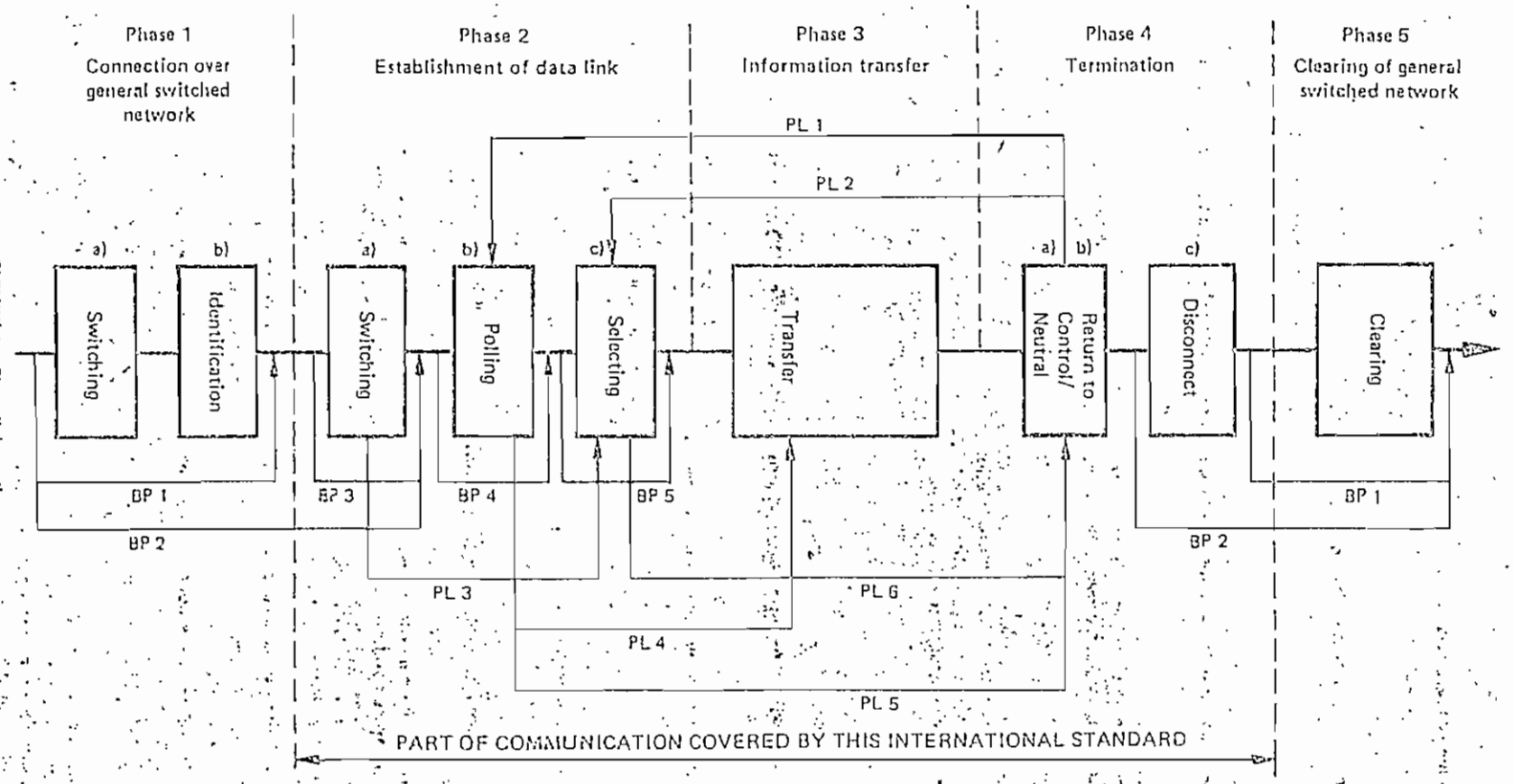


FIGURE 1 – Phase linkage diagram

By-pass 2 (BP2)

This by-pass applies to systems which do not involve line switching.

By-pass 3 (BP3)

This by-pass applies to systems that do not involve private line switching.

By-pass 4 (BP4)

In systems with control station, the suppression of the polling sub-phase allows only the sending of information from the control station to the others.

In systems without control station, each station can still select the other in order to transfer information but cannot poll in order to request transmission from the other end.

By-pass 5 (BP5)

This by-pass applies to systems with control station in which only the transfer of information from the tributary stations to the control station is required.

4.2 Phase diagrams

The detailed procedures for the phases 2 to 4 are given in the following text and illustrated by flow diagrams.

562

4.2.1 *Establishment of data link* (Phase 2) (see figure 2)

a) SWITCHING

A private switching process may be used. This, however, is not described in this International Standard.

b) POLLING

Polling is the process of inviting stations, one at a time in an orderly fashion, to transmit messages. The basic function of polling is to prevent contention by ensuring that only one station transmits at a time.

The polling process can only be performed by a control station, following EOT.

When a station receives its appropriate polling supervisory sequence, it becomes the master station.

Each polling supervisory sequence must uniquely identify one station on the data link. However, a given station may be assigned more than one address (for example, to distinguish between different precedences of originating traffic).

If no response or a non-valid response is received after transmission of a polling supervisory sequence, the control station must clear the possibly established data link by sending a termination supervisory sequence (EOT, see 4.2.3, termination phase, and 4.3, recovery procedures).

Some systems may not be able to tolerate a polling or selecting sequence immediately following EOT. In such cases it may be necessary to assure a short delay between the EOT character and the polling or selecting sequence using, for example, a number of "filler" characters.

Polling sequences are always preceded by EOT except in systems involving Phase 1 where the omission of EOT is optional. Some selecting sequences may be preceded by EOT.

c) SELECTING

Selecting is the normal process for inviting one or more stations to receive an information message. However, it can be used for the sole purpose of checking the identification of a station and/or for obtaining its status.

The selecting process can only be performed by a master station.

When used on a multistation data link, the selecting supervisory sequence uniquely identifies, by means of its address, one or several stations. This function is called station selection. The address may include information other than that indicating the address of the desired station, for example, priority, device selections, etc. The address of the selecting supervisory sequence may identify either a single station on the link or a group of stations on the link.

The station selection sequence may be sent either by the control station taking the master status or by a station previously polled.

When used on a point-to-point data link, the selecting supervisory sequence, which may be limited to the character ENQ, essentially turns the data link out of its neutral state.

In all cases, the selecting supervisory sequence calls for a status reply from the selected or opposed station. If no response or a non-valid response is received, the master station must take action to recover the communication (see 4.3, recovery procedures).

NOTES

1 An exception to the above is the use of the so-called "Fast select" method in which the information message follows immediately the selecting address (ENQ is not used). The use of this method requires special agreement between sender and recipient.

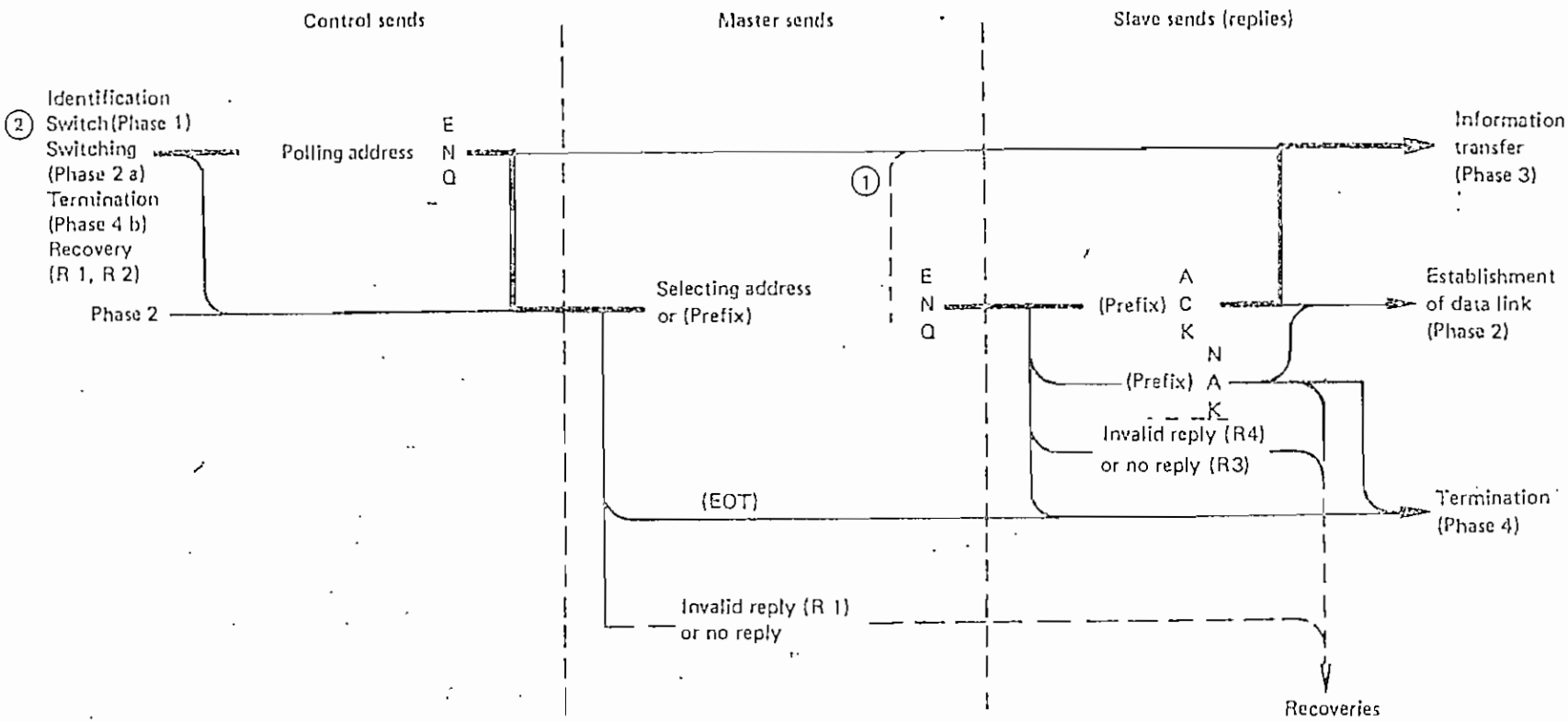
2 The method for achieving the sequential or group selection of a number of stations with the purpose of transmitting the same message to all stations is not completely defined by this International Standard.

4.2.2 *Information transfer* (phase 3) (see figure 3)

a) HEADING

The heading of an information message is a sequence of characters sent by a master station which constitutes the auxiliary information pertinent to the communication of a text. Such auxiliary information may include, for instance, characters representing routing, priority, security, message numbering, and associated characters. The definition of specific portions of a heading is not within the scope of this International Standard.

The sequence of characters which constitutes the heading is prefixed by the start of heading (SOH) character and is terminated only with the start of text (STX) character.



(1) For those systems where the information message follows the selecting address (Fast Select path), ENQ is not used.

(2) In most cases, the transmission of EOT is necessary after the connection has been established and before a polling or selecting supervisory sequence is transmitted.

FIGURE 2 — Phase 2 — Establishment of data link

The heading is not a "stand alone" message but must always be immediately followed by a text and is applicable only to that text. Any arrangement for association of one heading with more than one text can only be made by prior agreement between the affected parties and is not within the scope of this International Standard.

The heading may be sub-divided into more than one information block by terminating each such block with the end of transmission block (ETB) character and by prefixing the next following portion of the heading with the SOH character. The block check character (BCC), if used, immediately follows the ETB character (see ISO 1155).

b) TEXT

The text portion of an information message contains the information that is to be transmitted as an entity from the sender to the recipient(s). A text is always embodied between the STX and ETX characters, and is always transmitted by a master station.

The text may be sub-divided into more than one information block. Each of the blocks is terminated by an end of block control character (ETB, or ETX if it is the last block of text). The following portion of the text must be prefixed with the STX character.

If block checking is employed, the block check character immediately follows the ETB or ETX character.

564

The master station stops sending after each text or block has been sent, and normally does not resume transmission until the reply has been received.

c) REPLIES

Replies are used by the slave station to inform the master station of the status of the slave station and of the validity of the received message.

If the information message, or block, was acceptable, and the slave station is ready to receive, the slave station replies by transmitting the acknowledge character (ACK). The master station then transmits the next information message or block. If the master station has no more to transmit, it passes to termination phase.

If the information message or block is unacceptable, the slave station responds with the negative acknowledge character (NAK). This negative acknowledgement indicates to the master station that the data was unacceptable and also that the slave station is ready to receive. The next block of data transmitted is normally a retransmission of the previous information message or block.

If, during the information transfer, the slave station becomes unable to receive further, this station waits for the end of the transmitted information message or block and then responds by EOT. This EOT shall be interpreted by the other station(s) as a request for interruption and, according to the type of system used, either returns the responsibility of the communication to the control station or returns the data link to neutral state (see 4.2.3, termination phase).

Up to 15 characters may precede the final character of the reply sequence to convey information of a qualifying nature. The nature of this information is not a subject of this International Standard (see 3.1b), supervisory sequences formats).

4.2.3 *Termination* (Phase 4) (see figure 4).

There are essentially three situations when a station may elect to terminate the transmission in progress :

a) When a station refuses the establishment of a data link, either because it has nothing to transmit (negative reply to polling), or because it is unable to transmit (negative reply to polling), or because it is unable to receive (negative reply to selecting).

b) When the master station has successfully transmitted all of the data it desires to send :

The master station then transmits the end of transmission control character (EOT), indicating to the slave station that the master station has no more data to transmit. The master station thus relinquishes its right to transmit (unless it is also the control station).

c) When an unusual situation arises where either the master or the slave station desires to stop the transmission in progress :

If a master station sends EOT at any time other than after a terminated transmission, the transmission in progress is said to be aborted.

If a slave station sends EOT instead of a normal reply (ACK, NAK), the transmission in progress is said to be interrupted.

In all circumstances, such as a), b), c), the sending of EOT by any of the stations terminates the transmission, that is to say :

— in systems comprising a control station, returns the responsibility and the control of the communication to that control station; this function is called "Return to Control";

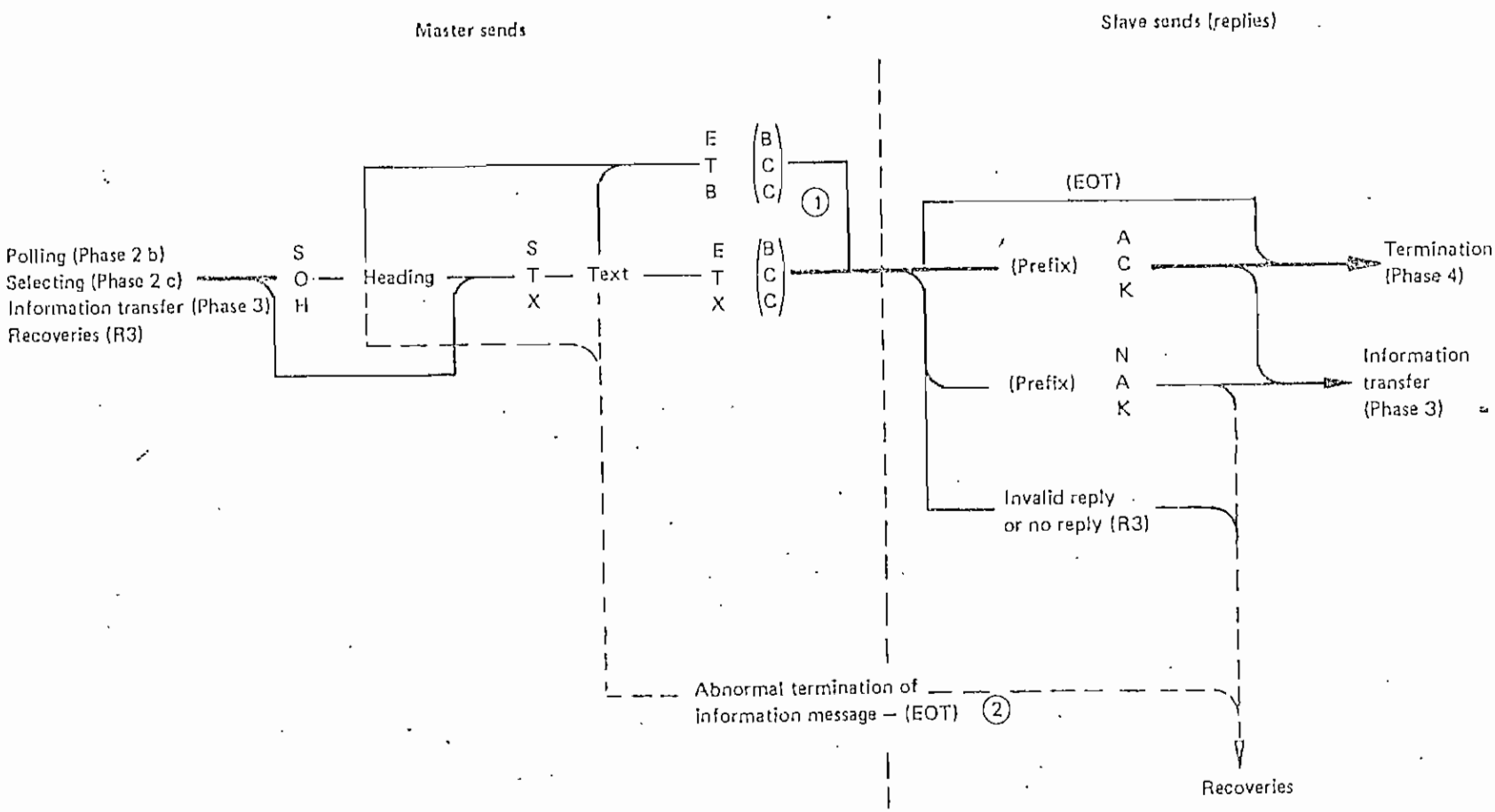
— in point-to-point systems, without control station, returns the data link to neutral state; this function is called "Return to Neutral".

In addition, in all the cases of termination, if the clearing of the connection (private and/or general network) is the intended consequence of the termination, the "Disconnect" supervisory sequence (DLE.EOT) shall be sent, either instead of EOT or following the receipt of an EOT.

4.3 Recovery procedures

It has been recognized that a number of recovery procedures are required to deal with various abnormal situations. Some recovery procedures are outlined in the following and their linkage to the appropriate phase is outlined in the diagrams in 4.2.

In all cases, after appropriate time-out periods, it shall be the responsibility of either the control station or the master station (never of a slave station) to take action.



(1) The BCC in brackets is optional.

(2) EOT can abort the transfer at any point in the transfer phase.

FIGURE 3 – Phase 3 – Information transfer

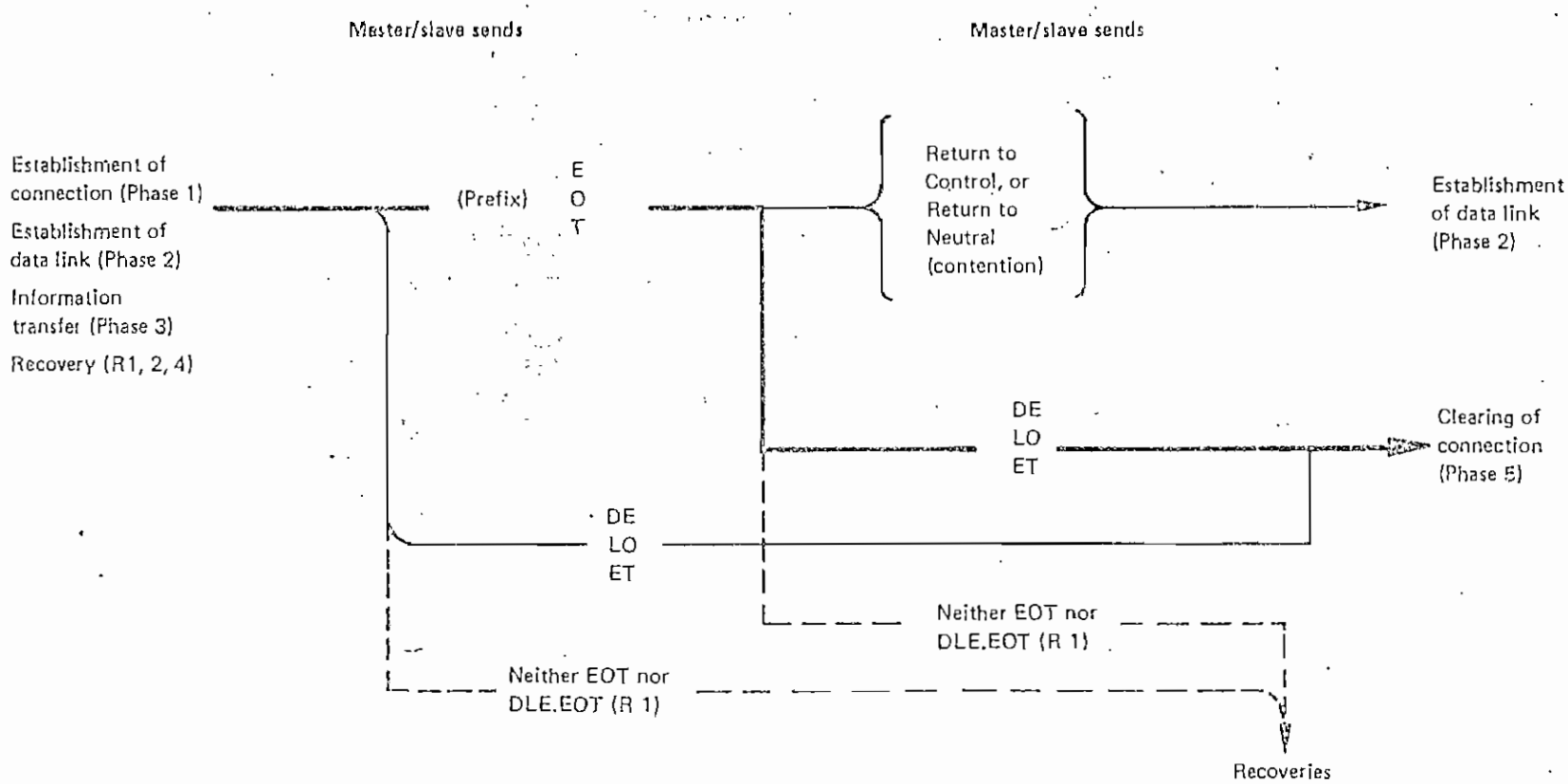


FIGURE 4 — Phase 4 — Termination

4.3.1 *Recovery procedures by control station*

R1 — In the case of :

— no reply or invalid reply to a polling supervisory sequence, or

— invalid, or absence of termination supervisory sequence,

the control station must transmit EOT.

R2 — In the case of repeated unsuccessful polling of one, several or all stations, the control station should, for example, set an alarm or report to the operator. Subsequent manual or automatic continuation is system-dependent and falls outside the scope of this International Standard.

4.3.2 *Recovery procedures by master station*

R3 — In the case of :

— no reply or invalid reply to an information message, or

— no reply or invalid reply to a selecting supervisory sequence,

the master station must repeat the previous transmission.

This procedure can lead to duplication of blocks. A possible alternative is :

In the case of no reply or an invalid reply to an information message, ENQ can be transmitted by the master station as a request to the slave station to repeat its previous response (ACK or NAK). This procedure is preferably used in conjunction with a response numbering scheme to ensure that blocks are neither added nor deleted.

R4 — In the case of :

— repeated unsuccessful transmission of an information message, or

— repeated unsuccessful transmission of a selecting supervisory sequence, or

— repeated negative replies (NAK) to an information message,

the master station should set an alarm or report to the operator and go to the termination phase. Subsequent manual or automatic continuation is system-dependent and falls outside the scope of this International Standard.

5 DESCRIPTION OF USE OF THE TRANSMISSION CONTROL CHARACTERS

In the following, the definition of the ten transmission control characters already given in clause 2 is recalled and the functional description of their use is summarized. See also 3.1.

SOH — Start of heading — TC 1

Definition

A transmission control character used as the first character of a heading of an information message.

Description of use

1) SOH is transmitted by a master station.

2) If a heading is used it must be preceded by SOH.

3) If a heading is sub-divided into transmission blocks, each block must be preceded again by SOH.

4) If block checking is used, see ISO 1155.

5) SOH is not permitted in text.

STX — Start of text — TC 2

Definition

A transmission control character which precedes a text and which is used to terminate a heading.

Description of use

1) STX is transmitted by the master station.

2) A text must be preceded by STX.

3) If a heading is used, STX in the message indicates the end of the heading.

4) If the text is sub-divided into transmission blocks, each block must be preceded again by STX.

5) If block checking is used, see ISO 1155.

6) STX is not permitted in the heading.

ETX — End of text — TC 3

Definition

A transmission control character which terminates a text.

Description of use

1) ETX is only transmitted by the master station.

2) ETX indicates the end of the text of each information message.

3) If an information message is sub-divided into transmission blocks, ETX is used to terminate the last block.

4) ETX calls for a reply from the slave station.

5) If block checking is used, ETX signals that the next following character is a block check character. See ISO 1155.

6) ETX is not permitted in text or heading.

EOT — End of transmission — TC 4

Definition

A transmission control character used to indicate the conclusion of the transmission of one or more texts.

Description of use

1) EOT may be transmitted by a control, master, or slave station.

2) The control station transmits EOT to condition all tributary stations to anticipate the reception of a forward supervisory sequence.

3) The master station, in a system with control station, transmits EOT to relinquish its right to transmit in favour of the control station.

4) The master station, in a system without control station, transmits EOT to indicate either the end or the aborting of a transmission and resets the master and slave stations to the neutral state.

5) The slave station transmits EOT to indicate its inability to receive further information messages. This is an abnormal reply or interruption and leads into termination phase.

6) EOT is not permitted in text or heading.

NOTE — Depending upon the system characteristics and configuration, EOT transmitted by a tributary station (master or slave) may reset some or all tributary stations.

ENQ — Enquiry — TC 5

Definition

A transmission control character used as a request for a response from a remote station — the response may include station identification and/or station status. When a "Who are you" function is required on the general switched transmission network, the first use of ENQ after the connection is established shall have the meaning "Who are you" (station identification). Subsequent use of ENQ may, or may not, include the function "Who are you", as determined by agreement.

Description of use

1) ENQ is transmitted by the control station during polling and by the master station during selecting.

2) In the polling sub-phase, ENQ is used to indicate the end of a polling address.

3) In the selecting sub-phase, ENQ is used to indicate the end of a selecting address or prefix when a reply is required from the slave station.

More specifically, ENQ can in this sub-phase

- terminate a station selection sequence;
- request identification and/or status;
- turn a data link out of neutral state.

4) ENQ is not permitted in text or heading.

ACK — Acknowledge — TC 6

Definition

A transmission control character transmitted by a receiver as an affirmative response to the sender.

Description of use

1) ACK is transmitted only by a slave station as an affirmative reply to a master station.

2) When supplementary information is included in the reply (for example, station identification or status information) it is prefixed to ACK.

3) In the selecting sub-phase, ACK is transmitted as a reply to a selecting supervisory sequence to indicate that the slave station is ready to receive.

4) In the information transfer phase, ACK indicates that the last transmitted information message or block was received correctly, and that the slave station is ready to receive the next one.

5) ACK is not permitted in text or heading.

DLE — Data link escape — TC 7

Definition

A transmission control character which will change the meaning of a limited number of contiguously following characters. It is used exclusively to provide supplementary data transmission control functions. Only graphics and transmission control characters can be used in DLE sequences.

Description of use

1) DLE immediately followed by EOT is transmitted by a master or a slave station to "disconnect", that is, to initiate the clearing of the connection over private and/or general switched network.

2) Other uses of DLE require prior agreement until additional DLE sequences are defined in future International Standards for modes other than the basic mode.

NAK — Negative acknowledge — TC 8

Definition

A transmission control character transmitted by a receiver as a negative response to the sender.

Description of use

1) NAK is transmitted only by a slave station as a negative reply to the master station.

2) When supplementary information is included in the reply (for example, station identification or status information) it is prefixed to NAK.

3) The slave station transmits NAK after receipt of a selecting supervisory sequence to indicate its inability to receive an information message.

4) In the information transfer phase, NAK indicates that the last transmitted information message or block was not received correctly, and the slave station is ready to receive the same one.

5) NAK is not permitted in text or heading.

SYN — Synchronous idle — TC 9

Definition

A transmission control character used by a synchronous transmission system in the absence of any other character (idle condition) to provide a signal from which synchronism may be achieved or retained between terminal equipments.

Description of use

1) SYN may be transmitted by a control, master, or slave station.

2) SYN is used to achieve character synchronization in synchronous data communication systems. At least two SYN characters must be transmitted prior to the transmission of any information message or supervisory sequence.

NOTE — It is assumed that the receiving station requires two consecutive SYN characters to reliably achieve character synchronization.

3) SYN can be used as "time-fill" to maintain, for instance, synchronization during periods when no other characters are available for transmission. When used as "time-fill", SYN may be added at any point in a character sequence, except :

a) between ETX or ETB and the block check character when block checking is implemented;

b) within DLE sequences.

4) SYN is generally removed at the receiving terminal installation.

5) If block checking is used, see ISO 1155.

ETB — End of transmission block — TC 10

Definition

A transmission control character used to indicate the end of a transmission block of data where data is divided into such blocks for transmission purposes.

Description of use

1) ETB is transmitted only by the master station.

2) If an information message is sub-divided into transmission blocks, ETB is used to terminate each block, with the exception of the final one.

3) ETB calls for a reply from the slave station.

4) If block checking is used, ETB indicates that the next following character is a block check character (BCC). See ISO 1155.

569

ANNEX A
(not part of the standard)

DEFINITIONS

A.1 The following definitions have been included for use with this International Standard since these definitions are not in accordance with those given in the corresponding sections of the ISO Information Processing Vocabulary (ISO 2382) :

coded character set : A finite set of characters arranged in a specified order according to given rules and conventions.

filler : A character that is used as a time or space fill when a block of a specified size is required and there are insufficient heading and/or text characters for this purpose.

heading : A sequence of characters preceding the text of a message. It enables the receiving station to handle the text(s).

information block : A sequence of characters of fixed or variable length which is a subdivision of an information message formed for the purpose of meeting transmission requirements.

transmission control characters (TCC) : Characters used either to define the nature of the information contained in a sequence of data characters or to convey supervisory instructions. They must not be transmitted as part of the text or heading.

A.2 The following definitions have been drawn in part from the corresponding sections of the ISO Information Processing Vocabulary (ISO 2382).

address : A sequence of characters to select or poll another station.

asynchronous transmission[1] : A transmission process such that between any two significant instants in the same group (block or character) there is always an integral number of unit intervals. Between two significant intervals located in different groups there is not always an integral number of unit intervals.

570

backward supervision : The use of supervisory sequences sent from the slave to the master station.

block check[1] : A system of error control based on the check that some preset rules for the formation of blocks are observed.

both-way : A mode of operation on two channels so that transmission may occur simultaneously. One of the channels is equipped for transmission in one direction while the other is equipped for transmission in the opposite direction.

channel[1] : A means of one-way transmission. Several channels may shere a common path as in carrier systems; in this case, each channel is allotted a particular frequency band which is reserved to it.

code transparent transmission : A transmission process which is capable of handling any character set or binary arrangement.

connection : The established path between two or more terminal installations. It is a permanent connection when it is established without using switching facilities, and a temporary connection when it is established by using switching facilities. It may consist of one or more channels in tandem.

contention : A condition arising on a communication channel when two or more stations try to transmit at the same time.

control station : The station on a network which supervises the procedures such as polling, selecting and recovery. It is also responsible for establishing order on the line in the event of contention, or any other abnormal situation, arising between any stations on the network.

data communication system : One or more data links each of which may be operating in the same or a different mode.

data link : An ensemble of terminal installations and the interconnecting network operating in a particular mode that permits information to be exchanged between terminal installations.

data terminal equipment[1] : Equipment comprising the data source, the data link or both.

either-way : A mode of operation of a channel to permit the transmission of signals in either direction. These transmissions cannot take place simultaneously.

error : Any received character or sequence of characters that does not conform to those transmitted.

1) Definition taken from CCITT yellow book and appropriate supplements.

error control or error protection : A procedure for detecting and reducing the effects of errors generated during the process of data transmission.

forward supervision : Use of supervisory sequences sent from the master to the slave station.

identifier : A sequence of one or more characters transmitted by a station in order to identify itself.

information message : A sequence of characters conveying the test. It may also convey supplementary information forming a heading.

invalid reception : A character or sequence of characters that has not been recognized in accordance with the expected character or sequence.

master station : The station which, at a given instant, has the right to select and to transmit an information message to a slave station and the responsibility of ensuring the information transfer. There should be only one master station on a data link at one time.

multi-point network : A configuration in which a connection is established between more than two terminal installations. The connection may include switching facilities.

network : The ensemble of equipment through which connections are made between terminal installations. These equipments operate in real time and do not introduce store and forward delays.

one-way : A mode of operation of a channel for one-way transmission of signals in a preassigned direction.

parity bit : A bit associated with a character or block for the purpose of checking the absence of error within that character or block. This is chosen to make the modulo 2 sum of the bits (including the parity bit) in the character or block a "0" or a "1" as required.

parallel transmission[1] : The simultaneous transmission of a certain number of signal elements constituting the same telegraph or data signal.

point-to-point connection : A configuration in which a connection is established between two, and only two, terminal installations. The connection may include switching facilities.

polling : The process of inviting another station to become a master station.

prefix : A sequence of characters (other than TCCs) used in a supervisory sequence to define or qualify the meaning of the supervisory sequence.

query : The process by which a master station asks a slave station to identify itself and to give its status.

recovery procedure : A process by which a responsible station within the network attempts to resolve either conflicting, or erroneous conditions arising in the communication process. The control or master station is responsible for this procedure.

route : The selected path between master station and slave station for the purpose of information transfer.

selecting : The process of inviting a station to receive.

serial transmission[1] : Transmission, at successive intervals, of signal elements constituting the same telegraph or data signal. The sequential elements may be transmitted with or without interruption provided that they are not transmitted simultaneously.

slave station : A station which, at a given instant, is intended to receive an information message from a master station.

start element (in a start/stop system) : Binary element serving to prepare the terminal installation for the reception of a character.

start/stop transmission[1] : Asynchronous transmission in which a group of code elements corresponding to a character signal is preceded by a start signal which serves to prepare the receiving mechanism for the reception and registration of a character and is followed by a stop signal which serves to bring the receiving mechanism to rest in preparation for the reception of the next character.

station : See terminal installation.

status : The capability at a given instant of a station to receive or transmit. It is indicated by a sequence of one or more characters which may be an acknowledgement of the previous data exchange.

1) Definition taken from CCITT yellow book and appropriate supplements.

stop element (in a start/stop system) : Binary element serving to bring the terminal installation to rest.

supervisory sequence : A sequence of one or more characters used for transmission control, whose structure does not follow the formatting rules applied to the information message.

switching[1] : Operations involved in interconnecting circuits in order to establish a temporary communication between two or more stations.

synchronous transmission[1] : A transmission process such that between any two significant instants there is always an integral number of unit intervals.

terminal installation (for data transmission) or station[1] : Installation comprising

- the data terminal equipment,
- the signal conversion equipment,
- and any intermediate equipment.

NOTE — In some instances the data terminal equipment may be connected directly to a data processing machine or may be a part of it.

text : A sequence of characters forming part of a transmission, which is transmitted as an entity to the ultimate destination and which contains the information to be conveyed.

tributary station : Any station on a non-switched multi-point network, other than the control station.

1) Definition taken from CCITT yellow book and appropriate supplements.

ANNEX B
(not part of the standard)

EXTENSIONS OF TRANSMISSION CONTROL FUNCTIONS USING DLE SEQUENCES

This annex describes a standard method for providing additional transmission control functions through the definition of Extension Sequences using DLE.

The character DLE is used as a prefix to one or more additional ISO 7-bit characters to form a sequence to represent a communication control function not directly represented by a single control character.

Preference should be given to two-character sequences.

Two-character sequences consist of DLE followed by one "final" character. When Extension Sequences formed by more than two characters are required, the Extention Sequence begins with DLE, continues with any number of "intermediate" characters and end with one "final" character.

The intermediate characters are those in column 2 of the ISO 7-bit code table.

The final characters are the TC's and those in columns 3, 4, 5, 6 and 7, with the exception of DEL (7/15) of the ISO 7-bit code table, i.e. the alphabetics, the numerics and several of the special graphics.

The final characters in columns 4, 5, 6 are for private use. The TC's and the characters in columns 3 and 7 with the exception of DEL (7/15) are reserved for future international standardization.

ANNEX C
(not part of the standard)

ALTERNATIVE POSITIVE ACKNOWLEDGEMENT OPTION

The following basic rules are recommended for alternate response numbering, if used :

1) Use of DLE 3/0 instead of ACK in phase 2.

2) Use of DLE 3/0 alternating with DLE 3/1 instead of ACK in phase 3, starting with DLE 3/1 as the acknowledgement for the first information block.

3) Use of DLE 3/0 or DLE 3/1, as appropriate, in response to ENQ in phase 3.

International Standard ISO 2110

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION•МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ•ORGANISATION INTERNATIONALE DE NORMALISATION

Data communication — 25-pin DTE/DCE interface connector and pin assignments

Téléinformatique — Affectation des broches et description du connecteur 25 broches à la jonction entre ETTD et ETCD

Second edition — 1989-07-01

600

UDC 681.327.8 : 621.316.541

Ref. No. ISO 2110-1989 (E)

Descriptors : data processing equipment, data processing, data transmission, electrical connections, connector pins, electric connectors, numbering, numerical designations, dimensions, describing, instructions.

Price based on 9 pages

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards institutes (ISO member bodies). The work of developing International Standards is carried out through ISO technical committees. Every member body interested in a subject for which a technical committee has been set up has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work.

Draft International Standards adopted by the technical committees are circulated to the member bodies for approval before their acceptance as International Standards by the ISO Council.

International Standard ISO 2110 was developed by Technical Committee ISO/TC 97, *Computers and information processing*, and was circulated to the member bodies in February 1979.

It has been approved by the member of the following countries :

Australia	Germany, F. R.	Romania
Belgium	Hungary	South Africa, Rep. of
Canada	Italy	Spain
Czechoslovakia	Japan	Sweden
Denmark	Libyan Arab Jamahiriya	Switzerland
Egypt, Arab Rep. of	Mexico	USA
Finland	Netherlands	
France	Poland	

The member bodies of the following countries expressed disapproval of the document on technical grounds :

United Kingdom
USSR

This second edition cancels and replaces the first edition (i.e. ISO 2110-1972).

Printed in Switzerland

Data communication — 25-pin DTE/DCE interface connector and pin assignments

1 Scope and field of application

This International Standard specifies the 25-pin connector and the assignment of connector pin numbers at the interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) or automatic calling equipment (ACE). It is applicable to voice band modems, public data network (PND) facilities, telegraph signal converters, and automatic calling equipment where CCITT[1] Recommendations V.24 and V.28 are applicable.

In the case of the PDN attachment through the X.20 interface, the functions of the interchange circuits are in accordance with CCITT Recommendation X.24.

In the case of the V.20 type outstation interface, the electrical characteristics are in accordance with CCITT Recommendation V.31.

602

2 References

ISO 4902, *Data communication — 37-pin and 9-pin DTE/DCE interface connectors and pin assignments.*

ISO 4903, *Data communication — 15-pin DTE/DCE interface connector and pin assignments.*

CCITT Recommendation S.16, *Automatic calling and/or answering on the telex network.*

CCITT Recommendation V.10 (or X.26), *Electrical characteristics for unbalanced double-current interchange circuits for general use with integrated circuit equipment in the field of data communications.*

CCITT Recommendation V.11 (or X.27), *Electrical characteristics for balanced double-current interchange circuits for general use with integrated circuit equipment in the field of data communication.*

CCITT Recommendation V.19, *Modems for parallel data transmission using telephone signalling frequencies.*

CCITT Recommendation V.20, *Parallel data transmission modems standardized for universal use in the general switched telephone network*

CCITT Recommendation V.21, 200-baud *modem standardized for use in the general switched telephone network.*

CCITT Recommendation V.23, 600/1 200-baud *modem standardized for use in the general switched telephone network.*

CCITT Recommendation V.24, *List of definitions for interchange circuits between data terminal equipment (DTE) and data circuit-terminating equipment (DCE).*

CCITT Recommendation V.25, *Automatic calling and/or answering equipment on the general switched telephone network, including disabling of echo suppressors on manually established calls.*

CCITT Recommendation V26, 2 400 bits *per second modem standardized for use on 4-wire leased telephone-type circuits.*

CCITT Recommendation V.26 bis, 2 400/1 200 bits *per second modem standardized for use in the general switched telephone network.*

CCITT Recommendation V.27, 4 800 bits *per second modem with manual equalizer standardized for use on leased telephone-type circuits.*

CCITT Recommendation V.27 bis, 4 800 bits *per second modem with automatic equalizer standardized for use on leased telephone-type circuits.*

CCITT Recommendation V.27 ter, 4 800/2 400 bits *per second modem standardized for use in the general switched telephone network.*

CCITT Recommendation V.28, *Electrical characteristics for unbalanced double-current interchange circuits.*

CCITT Recommendation V.29, 9 600 bits *per second modem standardized for use on leased telephone-type circuits.*

CCITT Recommendation V.31, *Electrical characteristics for single-current interchange circuits controlled by contact closure.*

CCITT Recommendation X.20, *Interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) for start-stop transmission services on public data networks (PDN).*

1) International Telegraph and Telephone Consultative Committee.

CCITT Recommendation X.20 bis, *V.21-compatible interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) for start-stop transmission services on public data networks (PDN).*

CCITT Recommendation X.21 bis, *Use on public data networks (PDN) of data terminal equipment (DTE) which are designed for interfacing to synchronous V-series modems.*

CCITT Recommendation X.24, *List of definitions for interchange circuits between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) on public data networks (PDN).*

3 Connector

A 25-pin connector shall be provided for the DTE/DCE interface. A separate 25-pin connector shall be provided for the automatic calling equipment interface if this facility is used.

Figure 1 illustrates the DTE interface connector which has 25 male contacts and a female shell. Figure 2 illustrates the DCE interface connector which has 25 female contacts and a male shell. Contact numbering is specified in figures 1 and 2. Figures 3, 4 and 5 illustrate contact spacing and dimensions.

In view of the multiplicity of known arrangements currently in use for fastening these connectors together, no preferred version is documented herein. The user is advised to determine the method available in his area.

The specification for the connector in this International Standard is provided for mechanical compatibility only. It is also intended to be mechanically compatible with the detailed connector specification currently being developed by the IEC.

4 Assignment of pin numbers

The pin assignments are given in table 2.

The list of the interchange circuits is given in table 1. Their provision and use shall be in conformity with CCITT Recommendations, referred to in the heading of table 2.

Table 1 — List of interchange circuits

Circuit number	Description
102	Signal ground or common return
103	Transmitted data
104	Received data
105	Request to send
106	Ready for sending
107	Data set ready
108/1	Connect data set to line
108/2	Data terminal ready
109	Data channel received line signal detector
110	Data signal quality detector
111	Data signalling rate selector (DTE source)
113	Transmitter signal element timing (DTE source)
114	Transmitter signal element timing (DCE source)
115	Receiver signal element timing (DCE source)
116	Select standby
118	Transmitted backward channel data
119	Received backward channel data
120	Transmit backward channel line signal
121	Backward channel ready
122	Backward channel received line signal detector
124	Select frequency groups
125	Calling indicator
126	Select transmit frequency
129	Request to receive
130	Transmit backward tone
131	Received character timing
132	Return to non-data mode
140	Remote loopback for point to point circuits
141	Local loopback
142	Test indicator
191	Transmitted voice answer
192	Received voice answer
201	Signal ground or common return
202	Call request
203	Data line occupied
204	Distant station connected
205	Abandon call
206	Digit signal (2^0)
207	Digit signal (2^1)
208	Digit signal (2^2)
209	Digit signal (2^3)
210	Present next digit
211	Digit present
213	Power indication
G	Signal ground or common return
T	Transmit
R	Receive

604

Table 2 — Assigment of pin numbers

Pin number	Interchange circuit numbers and remarks											
	Voice band modems					Public data networks			Telegraph		Automatic calling	
	Asynchronous		Synchronous	Parallel								
	A V.21	B V.23	C V.26, V.26 bis V.27, V.27 bis V.27 ter, V.29	D V.19, V.20 Instation	E V.20 Outstation	F X.20 bis	G X.21 bis	H X.20[8]	I Telex	J Other	K Telephone V.25	L Telex S.16
1	See note 1	See note 1	See note 1	See note 1	See note 1	See note 1	See note 1	See note 1	See note 1	See note 1	See note 1	See note 1
2	103	103	103	See note 5	192-A	103	103	T	103	103	211	211
3	104	104	104	A1[3]	A1[4]	104	104	R	104	104	205	205
4	105	105	105	A2[3]	A2[4]	F	105	F	N	N	202	202
5	106	106	106	A3[3]	A3[4]	106	106	F	106	106	210	210
6	107	107	107	A4[3]	B1[4]	107	107	F	107	107	213	213
7	102	102	102	131	B2[4]	102	102	G	102	102	201	201
8	109	109	109	109	B3[4]	109	109	F	109	109	F	F
9	N	N	N	C1[3]	C1[4]	N	N	N	N	N	N	N
10	N	N	N	C2[3]	C2[4]	N	N	N	N	N	N	N
11	126	N	N	C3[3]	C3[4]	F	N	N	N	N	F	F
12	F	122	122	C4[3]	192-B	F	F	N	F	F	F	F
13	F	121	121	B1[3]	See note 4	F	F	N	F	F	204	204
14	F	118	118	B2[3]	125-A	F	F	N	F	F	206	206
15	F	See note 2	114	B3[3]	125-B	F	114	N	F	F	207	207
16	F	119	119	B4[3]	105-A	F	F	N	F	F	208	208
17	F	See note 2	115	191-A	105-B	F	115	N	F	F	209	209
18	141	141	141	191-B	129-A	N	N	F	132	F	F	F
19	F	120	120	130	129-B	F	F	F	F	F	F	F
20	108*	108*	108*	105	119-A	108*	108*	F	108/2	108/2	F	F
21	140	140	140	125	119-B	N	N	F	F	F	F	F
22	125	125	125	108*	107-A	125	125	F	125	125	203	203
23	N	111	111	107	107-B	N	N	F	N	N	N	N
24	N	N	113[9]	102	108-A	N	F	F	N	N	N	N
25	142	142	142	124	108-B	N	142	F	F	F	F	F
Electrical characteristics	V.28[7]	V.28[7]	V.28[7]	V.28[6]	V.31[6]	V.28	V.28	V.28	V.28	V.28	V.28	V.28

Legend : N — Pin number permanently reserved for national use.
F — Pin number reserved for future international standardization and should not be used for national use.
* — CT 109/1 or CT 109/2.

NOTES

1 Pin 1 is assigned for connecting the shields between tandem sections of shielded interface cable. The shield may be connected either to protective ground or to signal ground at either the DTE or DCE or both in accordance with national regulations.

Signal ground may be further connected to protective ground in accordance with national safety regulations. Caution should be exercised to prevent establishment of ground loops carrying high currents.

2 Where signal element timing is provided in the DCE, pin 15 will be used for circuit 114, and pin 17 will be used for circuit 115.

3 Received data circuits 104 are designated A1 to A4, B1 to B4, and C1 to C4 corresponding to their relevant frequency.

4 Transmitted data circuits 103 are designated A1 to A3, B1 to B3, and C1 to C3 corresponding to their relevant frequency, and all use pin 13 as the common return according to V.31.

5 Pin 2 is reserved for national use. Where circuit 110 is provided in the DCE, it will use pin 2.

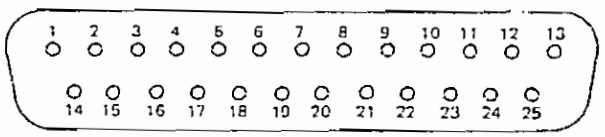
6 The electrical characteristics of circuits 191 and 192 are subject to specification within V.19 and V.20.

7 Alternative use of V.10 and V.11 electrical characteristics has been recognized by CCITT for V.21, V.23, V.26, V.26 bis, V.27, V.27 bis, V.27 ter, and V.29. The connectors and pin assignments associated with application of V.10 and V.11 to these interfaces are provided by ISO 4902.

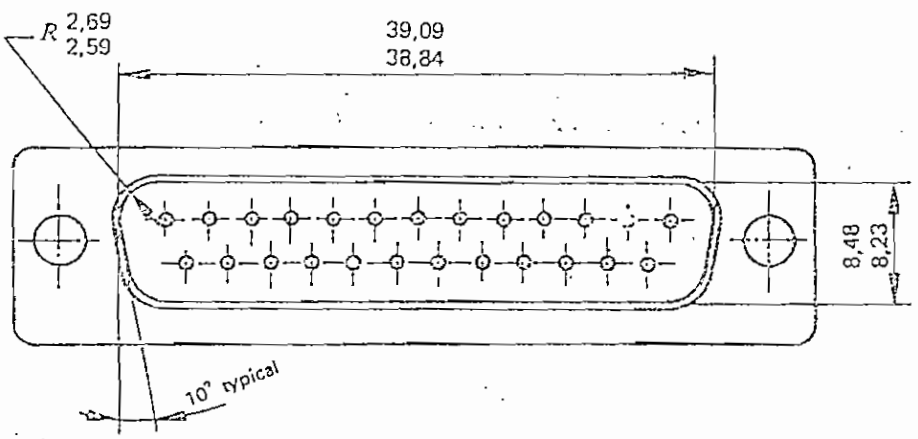
8 The functions of the interchange circuits are in accordance with X.24. This column refers only to X.20 type DTE with V.28 electrical characteristics since it may not interconnect with an X.20/V.10 DCE.

9 In some countries pin 24 is allocated to another interchange circuit such as circuit 116 (select standby).

Dimensions in millimetres



DTE connector face
contact numbering



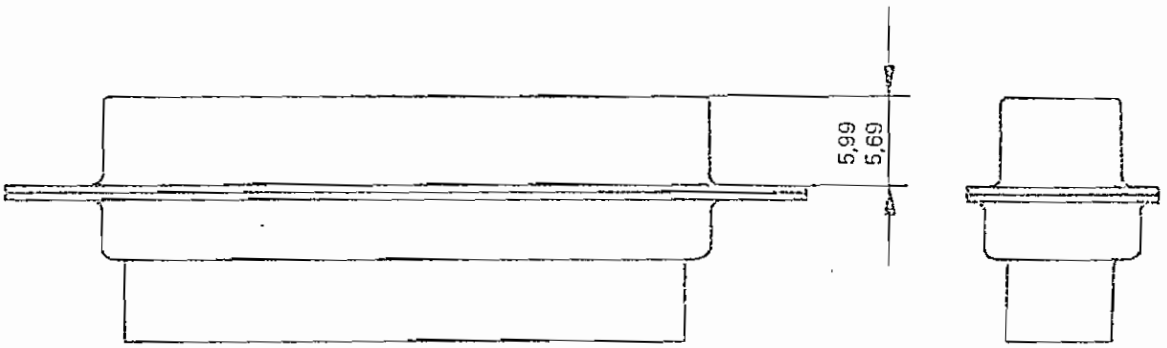
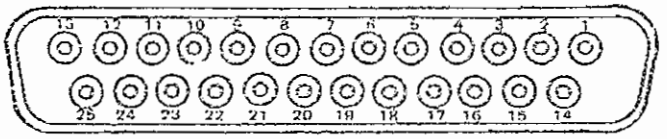
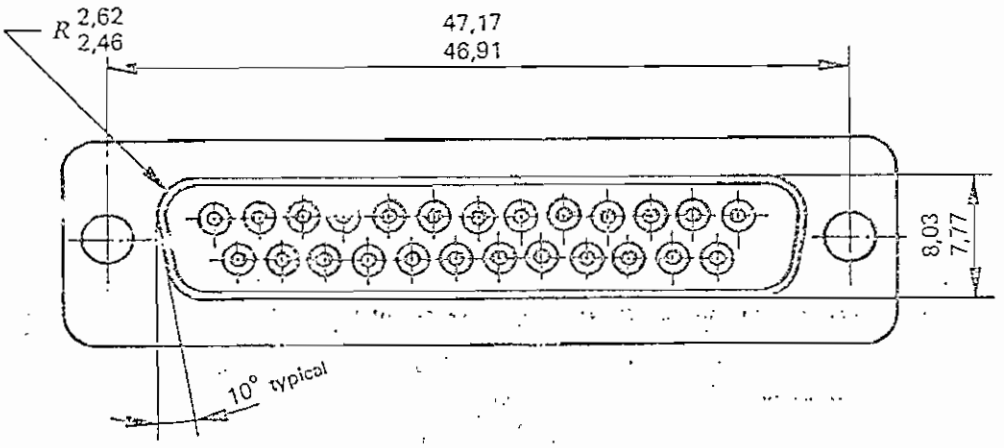


Figure 1 — DTE interface connector

Dimensions in millimetres



DCE connector face
contact numbering



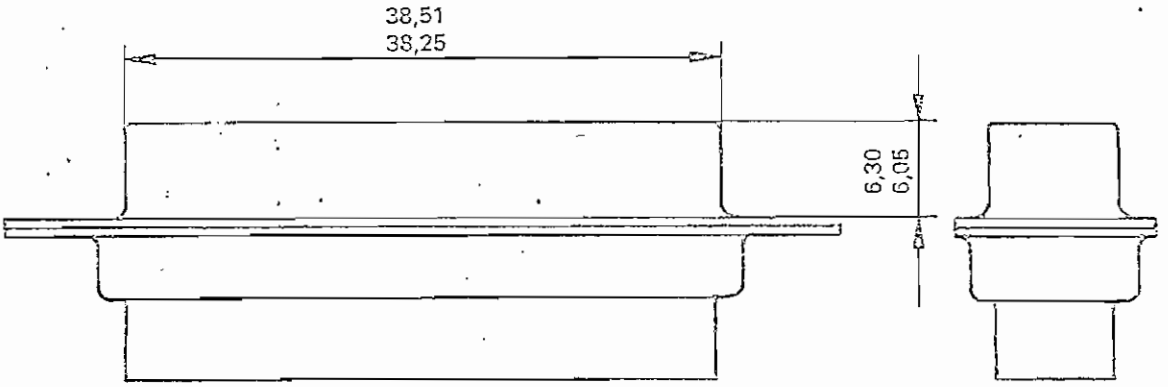


Figure 2 — DCE interface connector

Dimensions in millimetres

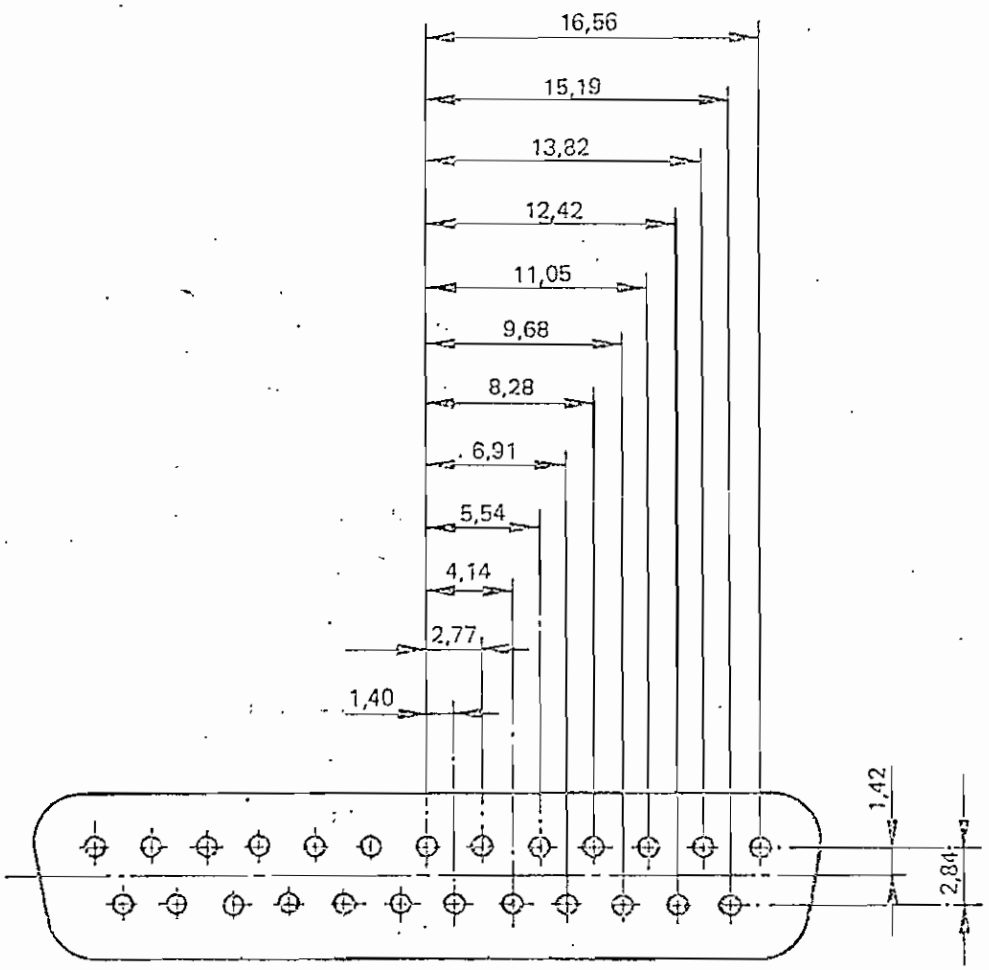


Figure 3 — Insert arrangement

Dimensions in millimetres

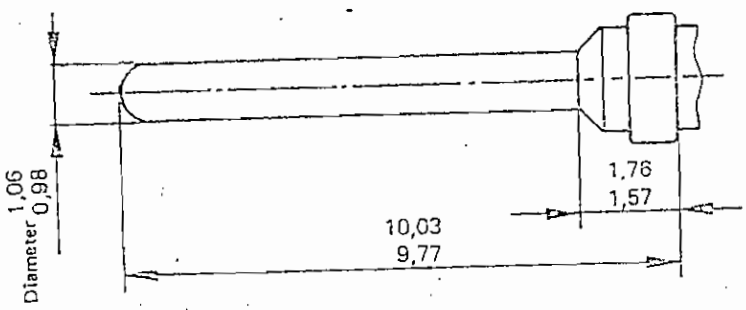
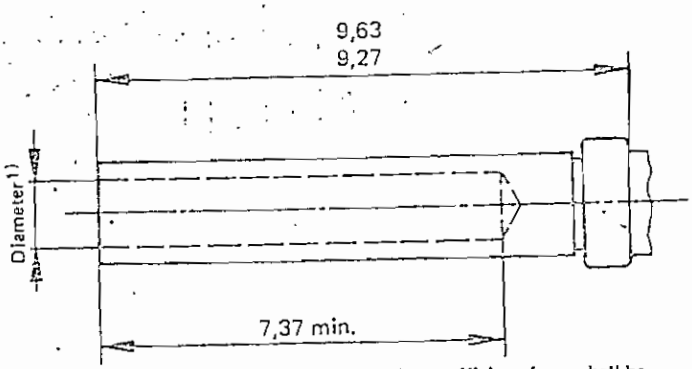


Figure 4 — Male contact

Dimensions in millimetres



1) When the pin is mated with the socket, sufficient force shall be applied by the socket to ensure proper electrical contact.

Figure 5 — Female contact

INTERNATIONAL STANDARD ISO 2628

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION · МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ · ORGANISATION INTERNATIONALE DE NORMALISATION

Basic mode control procedures — Complements

First edition — 1973-06-01

UDC 681.14

Ref. No. ISO 2628-1973 (E)

Descriptors : data processing, data transmission, control procedures, character sets, codes.

Price based on 8 pages

FOREWORD

ISO (the International Organization for Standardization) is a worldwide federation of national standards institutes (ISO Member Bodies). The work of developing International Standards is carried out through ISO Technical Committees. Every Member Body interested in a subject for which a Technical Committee has been set up has the right to be represented on that Committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work.

Draft International Standards adopted by the Technical Committees are circulated to the Member Bodies for approval before their acceptance as International Standards by the ISO Council.

International Standard ISO 2628 was drawn up by Technical Committee ISO/TC 97, *Computers and information processing*, and circulated to the Member Bodies in May 1972.

It has been approved by the Member Bodies of the following countries :

Australia	Ireland	Spain
Belgium	Italy	Sweden
Canada	Japan	Switzerland
Czechoslovakia	Netherlands	Thailand
Egypt, Arab Rep. of	New Zealand	United Kingdom
France	Portugal	U.S.A.
Germany	South Africa, Rep. of	U.S.S.R.

No Member Body expressed disapproval of the document.

Printed in Switzerland

Basic mode control procedures — Complements

0 INTRODUCTION

A data communication system may be considered as the set of the terminal installations and the interconnecting network that permits information to be exchanged.

A data link comprises terminal installations connected to the same network, operating at the same speed, in the same code. Any "store and forward" delay or intermediate data processing separates data links. Any system is constituted of one or several data links.

The information transfer in a data link is monitored by data link control procedures where some characters, selected within a coded character set, are given particular meanings according to the transmission phase and are used for various purposes such as to delineate information, to reverse the direction of transmission, to ask questions, to answer, etc.

This International Standard defines complements to the basic mode and its extensions :

1) Recovery procedures

   — System guidelines are given for the use of timers, counters, etc.;

2) Abort and interrupt procedures

   — Defines abort procedures which are always initiated by the master station, and interrupt procedures which are always initiated by the slave station;

3) Multiple station selection

   — Gives means whereby a master station may select more than one slave station so that all the selected slave stations receive the same transmission at the same time.

1 SCOPE AND FIELD OF APPLICATION

This International Standard extends the digital basic mode control procedures as defined in ISO/R 1745 and ISO 2111, to allow the following features :

1) Recovery procedures;

2) Abort and interrupt procedures;

3) Multiple station selection.

Those systems which conform to ISO/R 1745 do not necessarily have to include the functions described in this International Standard. However, those systems implementing the functions described in this International Standard and conforming to ISO/R 1745 and ISO 2111, must follow these recommendations.

2 REFERENCES

ISO/R 1745, *Basic mode control procedures for data communication systems.* (At present under revision.)

ISO 2111, *Data communication — Basic mode control procedures — Code independent information transfer.*

CCITT Recommendation V24, *Function and electrical characteristics of circuits at the interface between data terminal equipment and data communication equipment.*

3 RECOVERY PROCEDURES

3.1 General

These recovery procedures are system guidelines which should be used by all stations, as applicable. However, it is recognized that the detailed method of station mechanization, absolute value of timers, etc., may vary with applications and communication facilities.

In some cases, these recovery procedures can only detect the error condition and then notify the operator or the processor program, or both. In more sophisticated cases, automatic recovery is partially or completely possible. In other cases, only operators can perform the recovery procedures. Operator recovery procedures are not part of this International Standard. However, the operator may do such things as retry *n* more times, establish voice communication to the distant station in order to determine trouble, etc.

For a good system, the functions of timers A, B, and C defined below, must be utilized. The value of the timer may vary over a wide range depending upon whether they are implemented via hardware, software, or human operator.

It is recognized that in some systems additional timers may be required for such purposes as aiding synchronization procedures, added reliability, etc.

3.2 Timers and counters

Timers are primarily used as aids in recovery procedures when recognition of specific control characters does not occur. The action taken following a time-out is specified in

general terms to provide system protection. The absolute values of the timers are dependent upon such things as manual use, non-manual data entry, speed of transmission, type of data source/sink, etc.

Counting is primarily used as an aid in determining what recovery alternative is applicable in each error condition. The number of consecutive negative or invalid replies and the number of consecutive attempts to recover using one recovery procedure before an alternative is chosen depends upon the network configuration, quality of the channel, and application.

3.2.1 *Timer A* (No-response timer)

Where implemented : control station, master station, or both.

Purpose : protection against an invalid response or no response.

Start : after transmitting any ending character where a reply is expected; for example ENQ, ETB, ETX, DLE ETB, DLE ETX.

Stop : upon receipt of a valid reply from the communication line; for example ACK, NAK, STX, EOT, DLE STX.

When time-out occurs :

1) — retransmit same information (up to *n* times)[1], or

   — transmit different information; for example ENQ, different polling/selection sequence;

2) transmit EOT, when station abort procedures are used;

3) notify operator or processor program, or both;

4) return to non-transparent mode, if applicable.

3.2.2 *Timer B* (Receive timer)

Where implemented : slave station.

Purpose : protection against non-recognition of any block terminating character, for example ETB, ETX, ENQ, DLE ETB or DLE ETX received from the communication line.

Start :

1) receipt of SOH, STX (if not preceded by SOH), DLE SOH, DLE STX or other opening characters or sequence as required.

2) this timer may be restarted to permit receipt of variable length blocks.

Stop : upon receipt of a valid terminating character or sequence; for example ETB, ETX, ENQ, DLE ETB, DLE ETX.

When time-out occurs :

1) remain in slave status and initiate search for character synchronization in synchronous systems;

2) prepare to receive another transmission;

3) notify operator or processor program or both, and discard the incomplete block;

4) return to non-transparent mode, if applicable.

NOTE — For maximum system efficiency, the duration of the no-response timer (Timer A) should be short and the receive timer (Timer B) should time-out before the no-response timer.

3.2.3 *Timer C* (No-activity timer for switched lines)

Where implemented : all stations.

Purpose : facilitates disconnection procedures of the communication line if data transmission stops due to not recognizing DLE EOT, or due to remote station or communication facility problems.

Start or restart :

1) upon receipt of indication of circuit connection; for example receipt of ON condition of circuit 107 (data set ready[2]) or circuit 125 (calling indicator[2]) and circuit 108.2 (data terminal ready[2]).

2) upon receipt or transmission of any character in asynchronous systems or the synchronizing sequence in synchronous systems.

Stop :

1) upon receipt or transmission of DLE EOT, or

2) loss of circuit 107 (data set ready[2])

When time-out occurs :

1) disconnect communication circuit;

2) notify operator or processor program, or both;

3) return to control mode, if applicable;

4) return to non-transparent mode, if applicable.

3.2.4 *Timer D* (No-activity timer for non-switched lines)

Where implemented : control station

Purpose : serves as a "no-activity" time-out for all stations in a system.

Start or restart : upon receipt or transmission of any character in asynchronous systems or after the synchronizing sequence in synchronous systems.

Stop : upon receipt or transmission of EOT

1) Retransmission of a data block can result in duplication of a block at the receiving location if a block numbering or other protective scheme is not used.

2) CCITT — V 24 designation.

When time-out occurs :

1) notify operator or processor program, or both;

2) return to control mode, if applicable;

3) return to non-transparent mode, if applicable.

3.3 Recovery procedures

Some recovery procedures are outlined in the following with their linkage to the appropriate phase diagrams in 4.2 of ISO/R 1745 and to the timers A, B and C described in this International Standard.

In all cases, after the appropriate time-out periods, it shall be the final responsibility of either the control station or the master station to take action.

3.3.1 *Recovery procedures by control station*

R1 — In the case of :

1) invalid or absence of termination supervisory sequence detected by time-out of either timer A or timer C, the control station must transmit EOT or DLE EOT whichever is appropriate;

2) invalid or no response to a polling/selection sequence detected by time-out of timer A, the control station may transmit the same or a different polling/selection sequence following the transmission of an EOT and/or notify operator or processor program, or both.

R2 — In the case of :

repeated unsuccessful polling of one, several or all stations, the control station should notify operator or processor program, or both.

3.3.2 *Recovery procedures by master station*

R3 — In the case of :

1) invalid or no response to a selecting supervisory sequence detected by the time-out of timer A, the master station may

a) terminate by transmitting EOT;

b) transmit same or another selecting supervisory sequence (up to *n* times);

c) notify operator or processor program, or both.

2) invalid or no response to information message detected by time-out of timer A the master station may

a) repeat the previous transmission (up to *n* times). This procedure can lead to duplication of blocks;

b) transmit prefix ENQ (up to *n* times) which requests the slave station to repeat its previous response (ACK or NAK). This procedure can lead to loss of blocks unless used in conjunction with a response numbering scheme to ensure that blocks are neither added nor deleted.

R4 — In the case of :

1) repeated negative replies (NAK) or invalid or no responses to a selection supervisory sequence, the master station should notify the operator or processor program, or both;

2) repeated negative replies (NAK) or failure to receive a valid reply for an information block, the master station may transmit an EOT (if master station abort is used) and/or notify the operator or processor program, or both.

3.3.3 *Recovery procedures by a slave station*

Recovery procedures by a slave station are explained by the functions of timer B (see 3.2.2).

4 ABORT AND INTERRUPT PROCEDURES

4.1 General

Abort procedures are always initiated by the master station wishing either

1) to stop transmitting a block of information before its normal end (ETB or ETX) but without returning to control mode or neutral; or

2) to stop transmitting at any time during the information transfer phase and then return to control mode or neutral.

Interrupt procedures are always initiated by the slave station which desires to stop receiving either instantaneously or within a short period of time.

4.2 Abort procedures

4.2.1 *Block abort*

4.2.1.1 DESCRIPTION

The master station decides to terminate a block in an unusual way so that the slave station rejects this block. There is no return to control mode or neutral and the master station resumes transmission to the same slave station.

4.2.1.2 PROCEDURE

When the master station decides to abort a block, it terminates it immediately with ENQ (DLE ENQ if applicable). The slave station replies with NAK which is the only valid acknowledgement in this case. The master station then resumes transmission beginning with STX (or SOH). If the reply from the slave station is invalid, or if there is no reply, the normal recovery procedures may apply (*n* retries, time-out). (See Figure 1.)

NOTE — As examples, block abort may be used in the following cases :

— the master station determines that invalid data have been sent;

for example, errors are detected at the buffer storage level, or when reading data from their media, or by the source (operator).

— with fixed length blocks when, due to transmission, programming or operator errors, the block being transmitted overflows normal length.

— when the master station determines that the block being transmitted will not be accepted by the slave station.

4.2.2 *Station abort*

4.2.2.1 DESCRIPTION

The master station is sending a message and decides, either while a block of information is being sent or between two blocks of information, to stop transmitting and return to control mode or neutral.

4.2.2.2 PROCEDURE

a) While a block is being sent

When the master station decides to abort a transmission, it immediately sends the transmission control character ENQ (DLE ENQ). The slave station detects this unusual termination with ENQ (instead of ETB or ETX) and then replies with NAK which is the only valid reply in this case. After receiving NAK, the master station sends EOT and the communication link returns to control mode or neutral.

When there is no answer or an invalid answer, the normal recovery procedures may apply (*n* retries, time-out). (See Figure 2.)

b) Between two blocks of information

The master station terminates the block being transmitted in the usual way. The usual answer of the slave station is ACK. The master station then sends EOT and the communication link goes back to control mode or neutral.

If the answer is NAK or if there is no answer or an invalid one, the master station may or may not decide to use the normal recovery procedures (*n* retries, time-out) before transmitting EOT with the resulting return to control mode or neutral. (See Figure 3.)

NOTES

1 In switched line applications, DLE EOT may be used in place of EOT.

2 Examples of use : when it is intended to disconnect the line.

Master station abort may be used in the following cases :

— Master station detects its own malfunction, or a malfunction of the transmitting media.

— Master station detects a failure in the slave station or in the link (persisting NAK, or invalid reply, or absence of reply) or the master station detects that the slave station is no longer in a position to receive.

— Master station is notified that the transmission media are urgently required for another purpose.

4.3 Interrupt procedures

4.3.1 *Block interrupt*

4.3.1.1 DESCRIPTION

The slave station, at the end of a message or of an information block, is no longer in a position to receive and wishes the master station to cease transmission immediately.

4.3.1.2 PROCEDURE

The slave station replies EOT instead of its normal reply. EOT indicates a negative acknowledgement of the last received block and the conclusion of the current transmission. The communication link returns to control mode or neutral. (See Figure 4.)

NOTE — The transmission systems fall into one of the following classes :

1 Control station is also master station.

2 Control station is also slave station.

3 Control station is neither master nor slave but is monitoring only the transmissions of the master station.

4 Control station, being neither master nor slave, is monitoring all data exchange within the system.

The block interrupt procedure, as described in 4.3.1.2 above, can only be used in classes 1, 2 and 4. As regards class 3, the control station is not aware of the EOT sent by the slave station and there is no way to return to control mode or neutral other than through recovery procedures (control station time-out, for instance).

For this reason, the use of block interrupt procedure is not recommended for systems falling in class 3 above. Concerning systems falling in classes 1, 2 and 4, block interrupt is not recommended for frequent utilization; its use should be reserved for emergency situations.

4.3.2 *Station interrupt*

4.3.2.1 DESCRIPTION

Station interrupt is the means whereby a slave station can request the master station to stop transmitting as soon as possible.

4.3.2.2 PROCEDURE

Station interrupt is accomplished by the slave transmitting the control sequence DLE < instead of the normal positive acknowledgement. This reply has a double meaning :

1) it includes the positive acknowledgement which would have been normally sent;

2) it means a request from the slave station to have the current transmission terminated at the earliest possible time (by the master station sending EOT). However, the master station may not stop transmitting immediately

and may, for instance, continue to transmit so that its buffers are cleared and readily available for further transmissions. The point where the master station effectively stops is system dependent.

NOTES

1 Example of use. The control station being also the slave station may want to interrupt so as to be able to urgently poll or select another tributary station.

2 Recovery procedures. The possibility of a station interrupt sequence being garbled by line disturbances should be considered. In particular, if backward supervision numbering should be established, this may lead either to use the same numbering scheme for the station interrupt sequence as for the supervisory sequence, or to impose other rules to preserve the correct information blocks sequencing.

For instance, it should not be permitted to continuously send the station interrupt sequence.

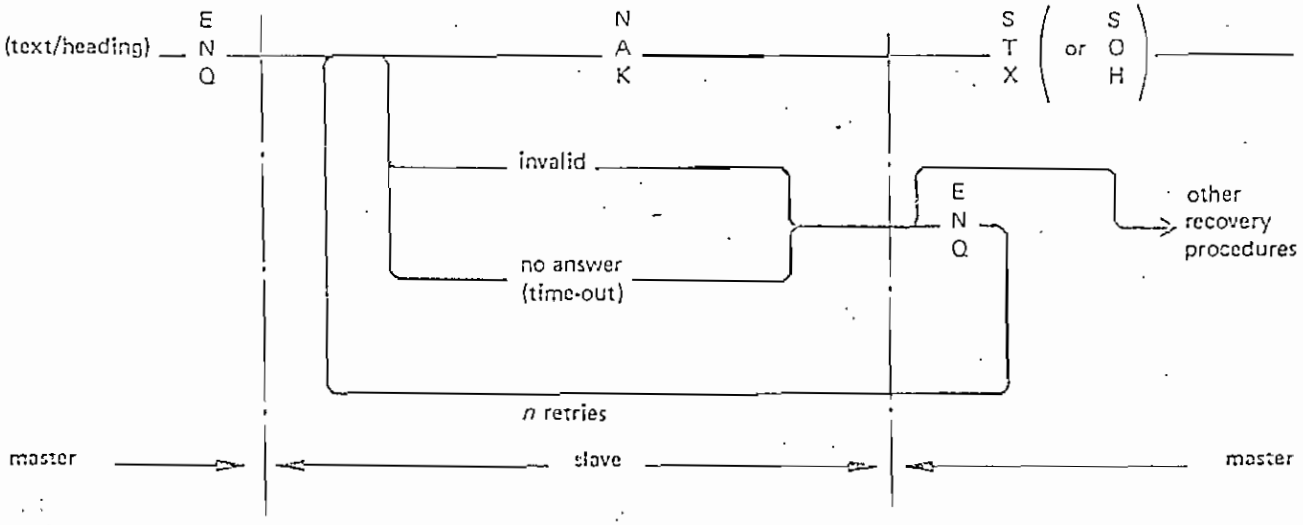


FIGURE 1 — Block abort

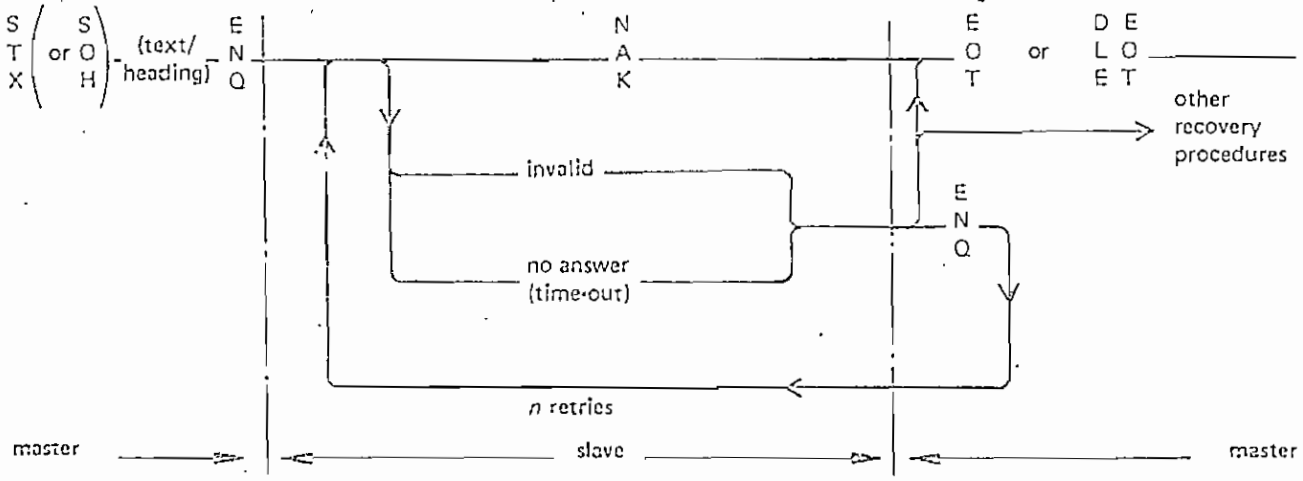


FIGURE 2 — Station abort (while a block is being sent)

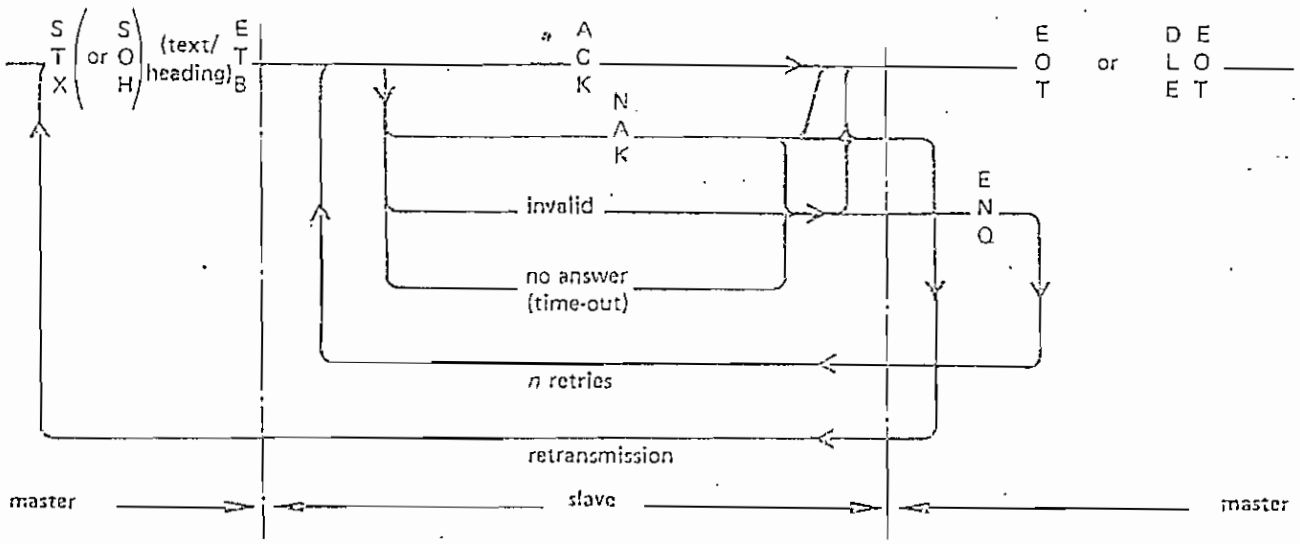


FIGURE 3 — Master station abort (between two information blocks)

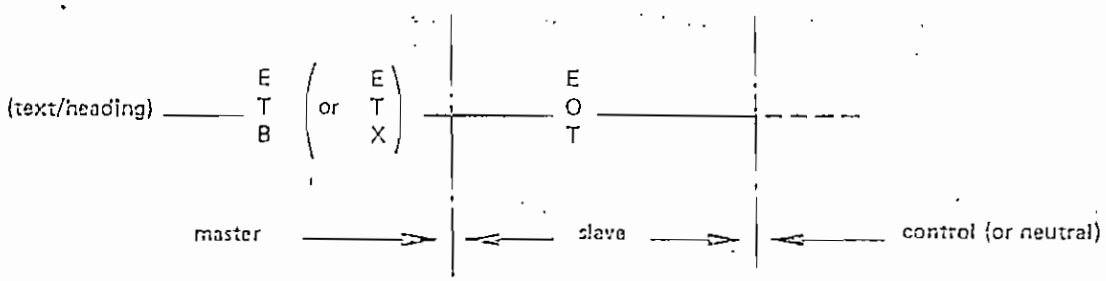


FIGURE 4 — Block interrupt

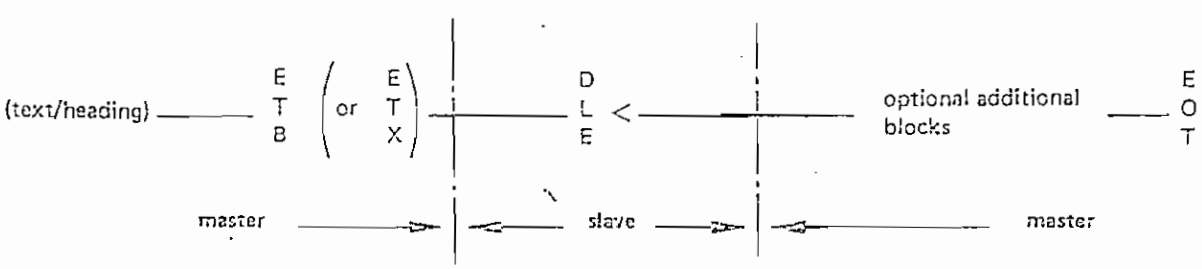


FIGURE 5 — Station interrupt

5 MULTIPLE STATION SELECTION

5.1 General

5.1.1 Multiple selection is a means whereby a master station may select more than one slave station so that all the selected slave stations receive the same transmission at the same time.

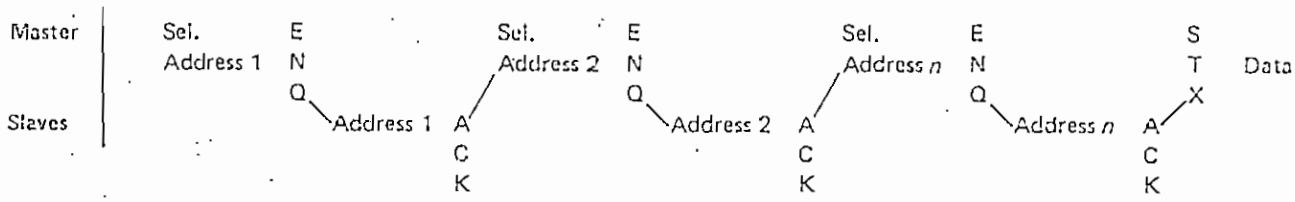
5.1.2 The procedures for multiple selection are not covered by the basic mode control procedures, hence they are considered to be an extension to them.

5.1.3 When a system is designed to work both with and without the multiple selection procedure, some means must be provided for the master station to notify the slave stations which procedure is to be entered. For example, by assigning two different addresses to each station having both facilites.

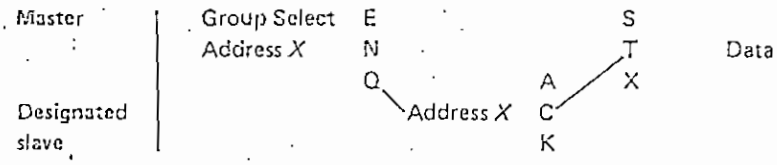
5.2 Selection sub-phase

Three methods providing various degrees of protection are proposed for the selection sub-phase. They are listed in the order of decreasing degree of protection.

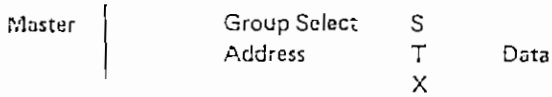
5.2.1 *Sequential selection with individual replies from the selected stations.*



5.2.2 *Group selection with reply from one designated station*, for example, the most distant one, or strategically located, or any station indicated within the selection sequence.



5.2.3 *Group selection with fast select.*

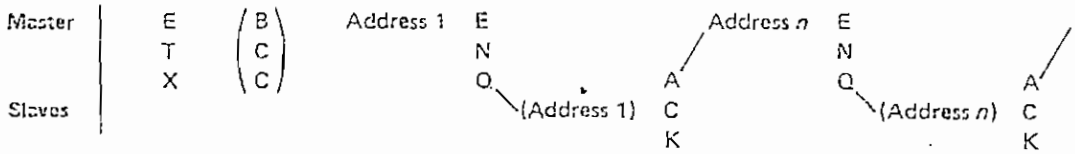


5.3 Information transfer phase

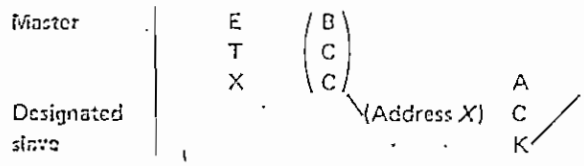
Three methods providing various degrees of protection are proposed for the information transfer phase. They are listed in the order of decreasing degree of protection.

5.3.1 *Information transfer with individual replies from the slave station*

After each transmission block, the master station sends a delivery verification supervisory sequence, consisting of a prefix identifying a single slave station, followed by ENQ. Only tributary stations having slave status should respond to delivery verification supervisory sequences.



5.3.2 *Information transfer with reply from one designated station*, for example the most distant, or strategically located.



X being the address of the designated station.

5.3.3 *No reply*

Although the "no reply" case is not considered by the basic mode, it is recognized that it may conveniently be used for general announcement (for example conference) and the broadcasting of messages of a "clear text" type.

5.4 Relations between selection procedures and information transfer procedures

Although the adoption of one of the three selection procedures does not preclude the adoption of any one of the three procedures for information transfer, it is recognized that some pairings would not be realistic. Straightforward pairings could be, for example, 2.1 with 3.1, 2.2 with 3.2, and 2.3 with 3.3.

8259A/8259A-2/8259A-8
PROGRAMMABLE INTERRUPT CONTROLLER

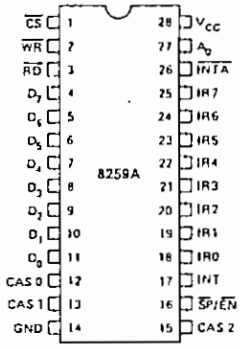
- 8086/8088 Compatible
- MCS-80/85™ Compatible
- Eight-Level Priority Controller
- Expandable to 64 Levels
- Programmable Interrupt Modes
- Individual Request Mask Capability
- Single +5V Supply (No Clocks)
- 28-Pin Dual-In-Line Package

The Intel® 8259A Programmable Interrupt Controller handles up to eight vectored priority interrupts for the CPU. It is cascadable for up to 64 vectored priority interrupts without additional circuitry. It is packaged in a 28-pin DIP, uses NMOS technology and requires a single +5V supply. Circuitry is static, requiring no clock input.

The 8259A is designed to minimize the software and real time overhead in handling multi-level priority interrupts. It has several modes, permitting optimization for a variety of system requirements.

The 8259A is fully upward compatible with the Intel® 8259. Software originally written for the 8259 will operate the 8259A in all 8259 equivalent modes (MCS-80/85, Non-Buffered, Edge Triggered).

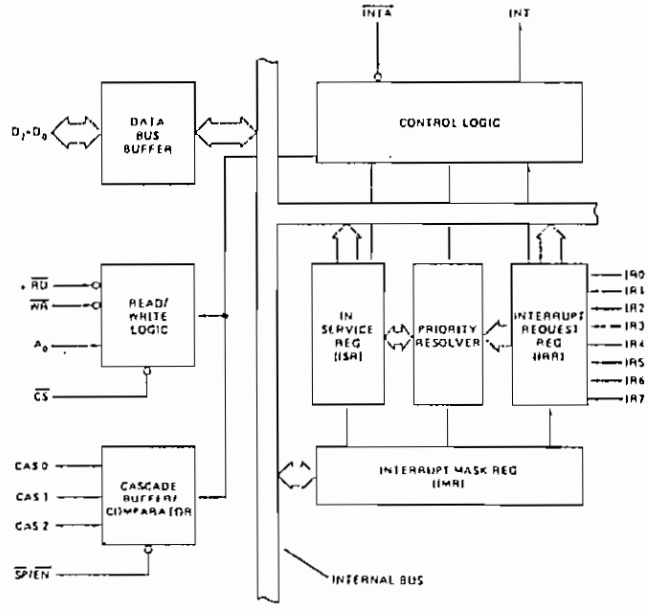
PIN CONFIGURATION



PIN NAMES

D_7-D_0	DATA BUS (BI DIRECTIONAL)
$\overline{RD}$	READ INPUT
$\overline{WR}$	WRITE INPUT
A_0	COMMAND SELECT ADDRESS
$\overline{CS}$	CHIP SELECT
CAS2 CAS0	CASCADE LINES
$\overline{SP/EN}$	SLAVE PROGRAM/ENABLE BUFFER
INT	INTERRUPT OUTPUT
$\overline{INTA}$	INTERRUPT ACKNOWLEDGE INPUT
IR0-IR7	INTERRUPT REQUEST INPUTS

BLOCK DIAGRAM



INTERRUPTS IN MICROCOMPUTER SYSTEMS

Microcomputer system design requires that I/O devices such as keyboards, displays, sensors and other components receive servicing in an efficient manner so that large amounts of the total system tasks can be assumed by the microcomputer with little or no effect on throughput.

The most common method of servicing such devices is the *Polled* approach. This is where the processor must test each device in sequence and in effect "ask" each one if it needs servicing. It is easy to see that a large portion of the main program is looping through this continuous polling cycle and that such a method would have a serious, detrimental effect on system throughput, thus limiting the tasks that could be assumed by the microcomputer and reducing the cost effectiveness of using such devices.

A more desirable method would be one that would allow the microprocessor to be executing its main program and only stop to service peripheral devices when it is told to do so by the device itself. In effect, the method would provide an external asynchroncus input that would inform the processor that it should complete whatever instruction that is currently being executed and fetch a new routine that will service the requesting device. Once this servicing is complete, however, the processor would resume exactly where it left off.

This method is called *Interrupt*. It is easy to see that system throughput would drastically increase, and thus more tasks could be assumed by the microcomputer to further enhance its cost effectiveness.

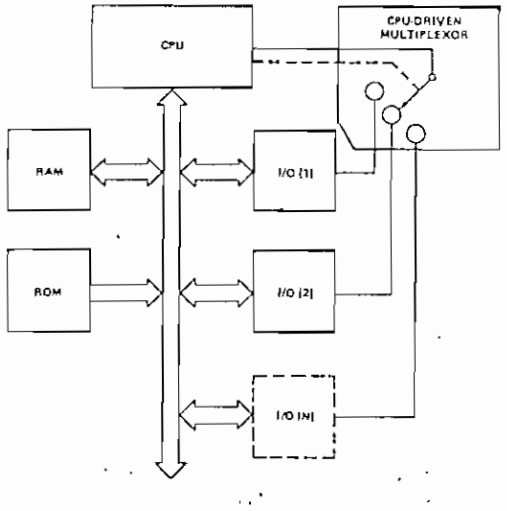
The Programmable Interrupt Controller (PIC) functions as an overall manager in an Interrupt-Driven system environment. It accepts requests from the peripheral equipment, determines which of the incoming requests is of the highest importance (priority), ascertains whether the incoming request has a higher priority value than the level currently being serviced, and issues an interrupt to the CPU based on this determination.

Each peripheral device or structure usually has a special program or "routine" that is associated with its specific functional or operational requirements; this is referred to as a "service routine". The PIC, after issuing an interrupt to the CPU, must somehow input information into the CPU that can "point" the Program Counter to the service routine associated with the requesting device. This "pointer" is an address in a vectoring table and will often be referred to, in this document, as vectoring data.

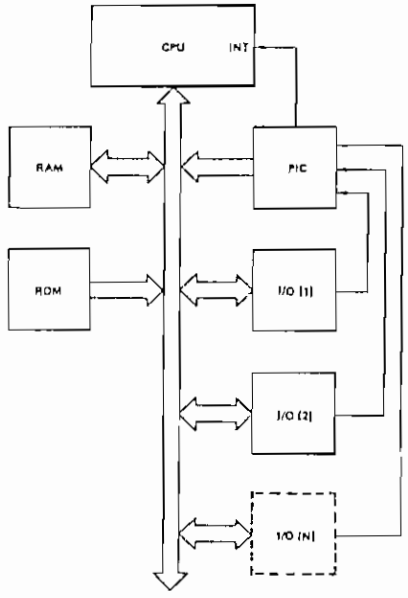
8259A BASIC FUNCTIONAL DESCRIPTION

GENERAL

The 8259A is a device specifically designed for use in real time, interrupt driven microcomputer systems. It manages eight levels or requests and has built-in features for expandability to other 8259A's (up to 64 levels). It is programmed by the system's software as an I/O peripheral. A selection of priority modes is available to the programmer so that the manner in which the requests are processed by the 8259A can be configured to match his system requirements. The priority modes can be changed or reconfigured dynamically at any time during the main program. This means that the complete interrupt structure can be defined as required, based on the total system environment.



Polled Method



Interrupt Method

INTERRUPT REQUEST REGISTER (IRR) AND IN-SERVICE REGISTER (ISR)

The interrupts at the IR input lines are handled by two registers in cascade, the Interrupt Request Register (IRR) and the In-Service Register (ISR). The IRR is used to store all the interrupt levels which are requesting service; and the ISR is used to store all the interrupt levels which are being serviced.

PRIORITY RESOLVER

This logic block determines the priorities of the bits set in the IRR. The highest priority is selected and strobed into the corresponding bit of the ISR during $\overline{INTA}$ pulse.

INTERRUPT MASK REGISTER (IMR)

The IMR stores the bits which mask the interrupt lines to be masked. The IMR operates on the IRR. Masking of a higher priority input will not affect the interrupt request lines of lower priority.

INT (INTERRUPT)

This output goes directly to the CPU interrupt input. The V_{OH} level on this line is designed to be fully compatible with the 8080A, 8085A, 8086 and 8088.

$\overline{INTA}$ (INTERRUPT ACKNOWLEDGE)

$\overline{INTA}$ pulses will cause the 8259A to release vectoring information onto the data bus. The format of this data depends on the system mode (μPM) of the 8259A.

DATA BUS BUFFER

This 3-state, bidirectional 8-bit buffer is used to interface the 8259A to the system Data Bus. Control words and status information are transferred through the Data Bus Buffer.

READ/WRITE CONTROL LOGIC

The function of this block is to accept OUTput commands from the CPU. It contains the Initialization Command Word (ICW) registers and Operation Command Word (OCW) registers which store the various control formats for device operation. This function block also allows the status of the 8259A to be transferred onto the Data Bus.

$\overline{CS}$ (CHIP SELECT)

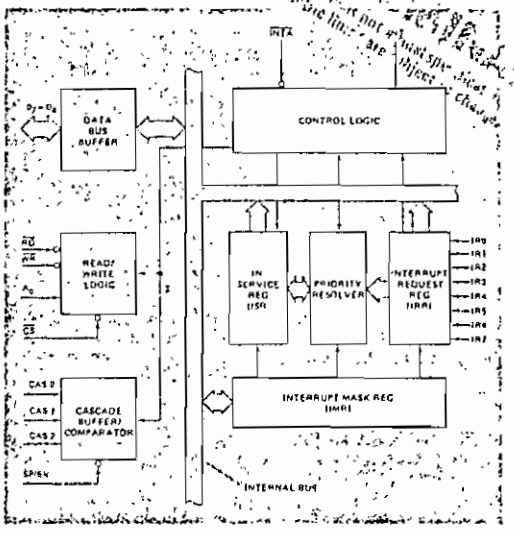
A LOW on this input enables the 8259A. No reading or writing of the chip will occur unless the device is selected.

$\overline{WR}$ (WRITE)

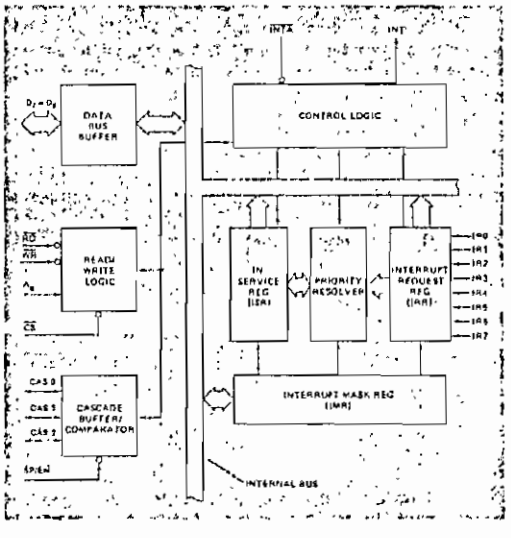
A LOW on this input enables the CPU to write control words (ICWs and OCWs) to the 8259A.

$\overline{RD}$ (READ)

A LOW on this input enables the 8259A to send the status of the Interrupt Request Register (IRR), In Service Register (ISR), the Interrupt Mask Register (IMR), or the Interrupt level onto the Data Bus.



8259A Block Diagram



8259A Block Diagram

A_0

This input signal is used in conjunction with $\overline{WR}$ and $\overline{RD}$ signals to write commands into the various command registers, as well as reading the various status registers of the chip. This line can be tied directly to one of the address lines.

THE CASCADE BUFFER/COMPARATOR

This function block stores and compares the IDs of all 8259A's used in the system. The associated three I/O pins (CAS0-2) are outputs when the 8259A is used as a master and are inputs when the 8259A is used as a slave. As a master, the 8259A sends the ID of the interrupting slave device onto the CAS0-2 lines. The slave thus selected will send its preprogrammed subroutine address onto the Data Bus during the next one or two consecutive $\overline{INTA}$ pulses. (See section "Cascading the 8259A".)

INTERRUPT SEQUENCE

The powerful features of the 8259A in a microcomputer system are its programmability and the interrupt routine addressing capability. The latter allows direct or indirect jumping to the specific interrupt routine requested without any polling of the interrupting devices. The normal sequence of events during an interrupt depends on the type of CPU being used.

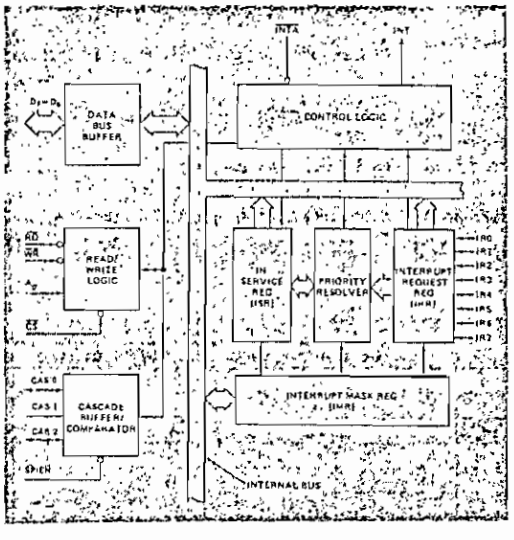
The events occur as follows in an MCS-80/85 system:

1. One or more of the INTERRUPT REQUEST lines (IR7-0) are raised high, setting the corresponding IRR bit(s).
2. The 8259A evaluates these requests, and sends an INT to the CPU, if appropriate.
3. The CPU acknowledges the INT and responds with an $\overline{INTA}$ pulse.
4. Upon receiving an $\overline{INTA}$ from the CPU group, the highest priority ISR bit is set, and the corresponding IRR bit is reset. The 8259A will also release a CALL instruction code (11001101) onto the 8-bit Data Bus through its D7-0 pins.
5. This CALL instruction will initiate two more $\overline{INTA}$ pulses to be sent to the 8259A from the CPU group.
6. These two $\overline{INTA}$ pulses allow the 8259A to release its preprogrammed subroutine address onto the Data Bus. The lower 8-bit address is released at the first $\overline{INTA}$ pulse and and the higher 8-bit address is released at the second $\overline{INTA}$ pulse.
7. This completes the 3-byte CALL instruction released by the 8259A. In the AEOI mode the ISR bit is reset at the end of the third $\overline{INTA}$ pulse. Otherwise, the ISR bit remains set until an appropriate EOI command is issued at the end of the interrupt sequence.

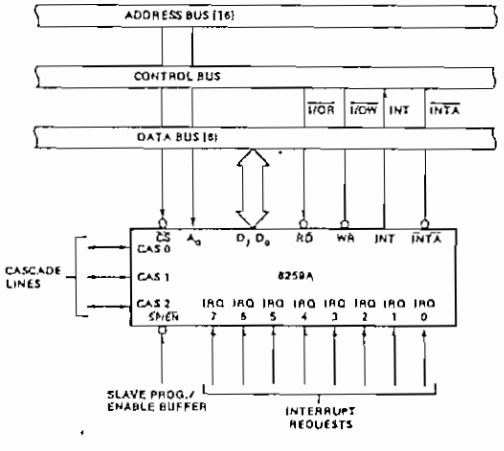
The events occuring in an 8086/8088 system are the same until step 4.

4. Upon receiving an $\overline{INTA}$ from the CPU group, the highest priority ISR bit is set and the corresponding IRR bit is reset. The 8259A does not drive the Data Bus during this cycle.
5. The 8086/8088 CPU will initiate a second $\overline{INTA}$ pulse. During this pulse, the 8259A releases an 8-bit pointer onto the Data Bus where it is read by the CPU.
6. This completes the interrupt cycle. In the AEOI mode the ISR bit is reset at the end of the second $\overline{INTA}$ pulse. Otherwise, the ISR bit remains set until an appropriate EOI command is issued at the end of the interrupt subroutine.

If no interrupt request is present at step 4 of either sequence (i.e., the request was too short in duration) the 8259A will issue an interrupt level 7. Both the vectoring bytes and the CAS lines will look like an interrupt level 7 was requested.



8259A Block Diagram



8259A Interface to Standard System Bus

INTERRUPT SEQUENCE OUTPUTS

MCS-80/85 MODE

This sequence is timed by three $\overline{INTA}$ pulses. During the first $\overline{INTA}$ pulse the CALL opcode is enabled onto the data bus.

Content of First Interrupt Vector Byte

	D7	D6	D5	D4	D3	D2	D1	D0
CALL CODE	1	1	0	0	1	1	0	1

During the second $\overline{INTA}$ pulse the lower address of the appropriate service routine is enabled onto the data bus. When Interval = 4 bits A_5-A_7 are programmed, while A_0-A_4 are automatically inserted by the 8259A. When Interval = 8 only A_6 and A_7 are programmed, while A_0-A_5 are automatically inserted.

Content of Second Interrupt Vector Byte

IR	Interval = 4							
	D7	D6	D5	D4	D3	D2	D1	D0
7	A7	A6	A5	1	1	1	0	0
6	A7	A6	A5	1	1	0	0	0
5	A7	A6	A5	1	0	1	0	0
4	A7	A6	A5	1	0	0	0	0
3	A7	A6	A5	0	1	1	0	0
2	A7	A6	A5	0	1	0	0	0
1	A7	A6	A5	0	0	1	0	0
0	A7	A6	A5	0	0	0	0	0

IR	Interval = 8							
	D7	D6	D5	D4	D3	D2	D1	D0
7	A7	A6	1	1	1	0	0	0
6	A7	A6	1	1	0	0	0	0
5	A7	A6	1	0	1	0	0	0
4	A7	A6	1	0	0	0	0	0
3	A7	A6	0	1	1	0	0	0
2	A7	A6	0	1	0	0	0	0
1	A7	A6	0	0	1	0	0	0
0	A7	A6	0	0	0	0	0	0

During the third INTA pulse the higher address of the appropriate service routine, which was programmed as byte 2 of the initialization sequence (A_8 – A_{15}), is enabled onto the bus.

Content of Third Interrupt Vector Byte

D7	D6	D5	D4	D3	D2	D1	D0
A15	A14	A13	A12	A11	A10	A9	A8

8086/8088 Mode

8086/8088 mode is similar to MCS80/85 mode except that only two Interrupt Acknowledge cycles are issued by the processor and no CALL opcode is sent to the processor. The first Interrupt acknowledge cycle is similar to that of MCS-80/85 systems in that the 8259A uses it to internally freeze the state of the interrupts for priority resolution and as a master it issues the interrupt code on the cascade lines at the end of the $\overline{INTA}$ pulse. On this first cycle it does not issue any data to the processor and leaves its data bus buffers disabled. On the second interrupt acknowledge cycle in 8086/8088 mode the master (or slave if so programmed) will send a byte of data to the processor with the acknowledged interrupt code composed as follows (note the state of the ADI mode control is ignored and A_5-A_{11} are unused in 8086/8088 mode):

	D7	D6	D5	D4	D3	D2	D1	D0
IR7	T7	T6	T5	T4	T3	1	1	1
IR6	T7	T6	T5	T4	T3	1	1	0
IR5	T7	T6	T5	T4	T3	1	0	1
IR4	T7	T6	T5	T4	T3	1	0	0
IR3	T7	T6	T5	T4	T3	0	1	1
IR2	T7	T6	T5	T4	T3	0	1	0
IR1	T7	T6	T5	T4	T3	0	0	1
IR0	T7	T6	T5	T4	T3	0	0	0

PROGRAMMING THE 8259A

The 8259A accepts two types of command words generated by the CPU:

1. *Initialization Command Words (ICWs):* Before normal operation can begin, each 8259A in the system must be brought to a starting point — by a sequence of 2 to 4 bytes timed by $\overline{WR}$ pulses. This sequence is described in Figure 1.

2. *Operation Command Words (OCWs):* These are the command words that are sent to the 8259A for various forms of operation, such as:

- Interrupt Masking
- End of Interrupt
- Priority Rotation
- Interrupt Status

The OCWs can be written into the 8259A anytime after initialization.

INITIALIZATION

GENERAL

Whenever a command is issued with A0=0 and D4=1, this is interpreted as Initialization Command Word 1 (ICW1). ICW1 starts the initialization sequence during which the following automatically occur.

a. The edge sense circuit is reset, which means that following initialization, an interrupt request (IR) input must make a low-to-high transition to generate an interrupt.
b. The Interrupt Mask Register is cleared.
c. R7 input is assigned priority 7.
d. The slave mode address is set to 7.
e. Special Mask Mode is cleared and Status Read is set to IRR.
f. If IC4=0, then all functions selected in ICW4 are set to zero. (Non-Buffered mode*, no Auto-EOI, MCS-80/85 system).

*Note: Master/Slave in ICW4 is only used in the buffered mode.

A_0	D_4	D_3	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	INPUT OPERATION (READ)
0			0	1	0	IRR, ISR or Interrupting Level → DATA BUS (Note 1)
1			0	1	0	IMR → DATA BUS
						OUTPUT OPERATION (WRITE)
0	0	0	1	0	0	DATA BUS → OCW2
0	0	1	1	0	0	DATA BUS → OCW3
0	1	X	1	0	0	DATA BUS → ICW1
1	X	X	1	0	0	DATA BUS → OCW1, ICW2, ICW3, ICW4 (Note 2)
						DISABLE FUNCTION
X	X	X	1	1	0	DATA BUS — 3-STATE (NO OPERATION)
X	X	X	X	X	1	DATA BUS — 3-STATE (NO OPERATION)

Notes: 1. Selection of IRR, ISR or Interrupting Level is based on the content of OCW3 written before the READ operation
2. On-chip sequencer logic queues these commands into proper sequence

8259A Basic Operation

INITIALIZATION COMMAND WORDS 1 AND 2 (ICW1, ICW2)

A_5-A_{15}: *Page starting address of service routines.* In an MCS 80/85 system, the 8 request levels will generate CALLs to 8 locations equally spaced in memory. These can be programmed to be spaced at intervals of 4 or 8 memory locations, thus the 8 routines will occupy a page of 32 or 64 bytes, respectively.

The address format is 2 bytes long (A_0-A_{15}). When the routine interval is 4, A_0-A_4 are automatically inserted by the 8259A, while A_5-A_{15} are programmed externally. When the routine interval is 8, A_0-A_5 are automatically inserted by the 8259A, while A_6-A_{15} are programmed externally.

The 8-byte interval will maintain compatibility with current software, while the 4-byte interval is best for a compact jump table.

In an MCS-86 system T7-T3 are inserted in the five most significant bits of the vectoring byte and the 8259A sets the three least significant bits according to the interrupt level. A_{10}-A_5 are ignored and ADI (Address Interval) has no effect.

LTIM: If LTIM = 1, then the 8259A will operate in the level interrupt mode. Edge detect logic on the interrupt inputs will be disabled.

ADI: CALL address interval. ADI = 1 then interval = 4; ADI = 0 then interval = 8.

SNGL: Single. Means that this is the only 8259A in the system. If SNGL = 1 no ICW3 will be issued.

IC4: If this bit is set — ICW4 has to be read. If ICW4 is not needed, set IC4 = 0.

INITIALIZATION COMMAND WORD 3 (ICW3)

This word is read only when there is more than one 8259A in the system and cascading is used, in which case SNGL = 0. It will load the 8-bit slave register. The functions of this register are:

a. In the master mode (either when $\overline{SP}$ = 1, or in buffered mode when M/S = 1 in ICW4) a "1" is set for each slave in the system. The master then will release byte 1 of the call sequence (for MCS-80/85 system) and will enable the corresponding slave to release bytes 2 and 3 (for 8086/8088 only byte 2) through the cascade lines.

b. In the slave mode (either when $\overline{SP}$ = 0, or if BUF = 1 and M/S = 0 in ICW4) bits 2-0 identify the slave. Tha slave compares its cascade input with these bits and if they are equal, bytes 2 and 3 of the call sequence (or just byte 2 for 8086/8088) are released by it on the Data Bus.

INITIALIZATION COMMAND WORD 4 (ICW4)

SFNM: If SFNM = 1 the special fully nested mode is programmed.

BUF: If BUF = 1 the buffered mode is programmed. In buffered mode $\overline{SP}/\overline{EN}$ becomes an enable output and the master/slave determination is by M/S.

M/S: If buffered mode is selected: M/S = 1 means the 8259A is programmed to be a master, M/S = 0 means the 8259A is programmed to be a slave. If BUF = 0, M/S has no function.

AEOI: If AEOI = 1 the automatic end of interrupt mode is programmed.

μPM: Microprocessor mode: μPM = 0 sets the 8259A for MCS-80/85 system operation, μPM = 1 sets the 8259A for MCS-86 system operation.

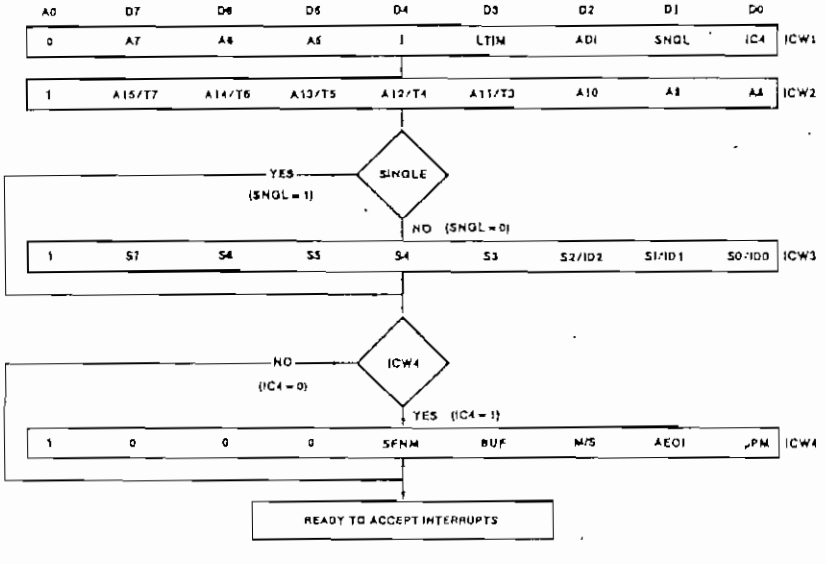
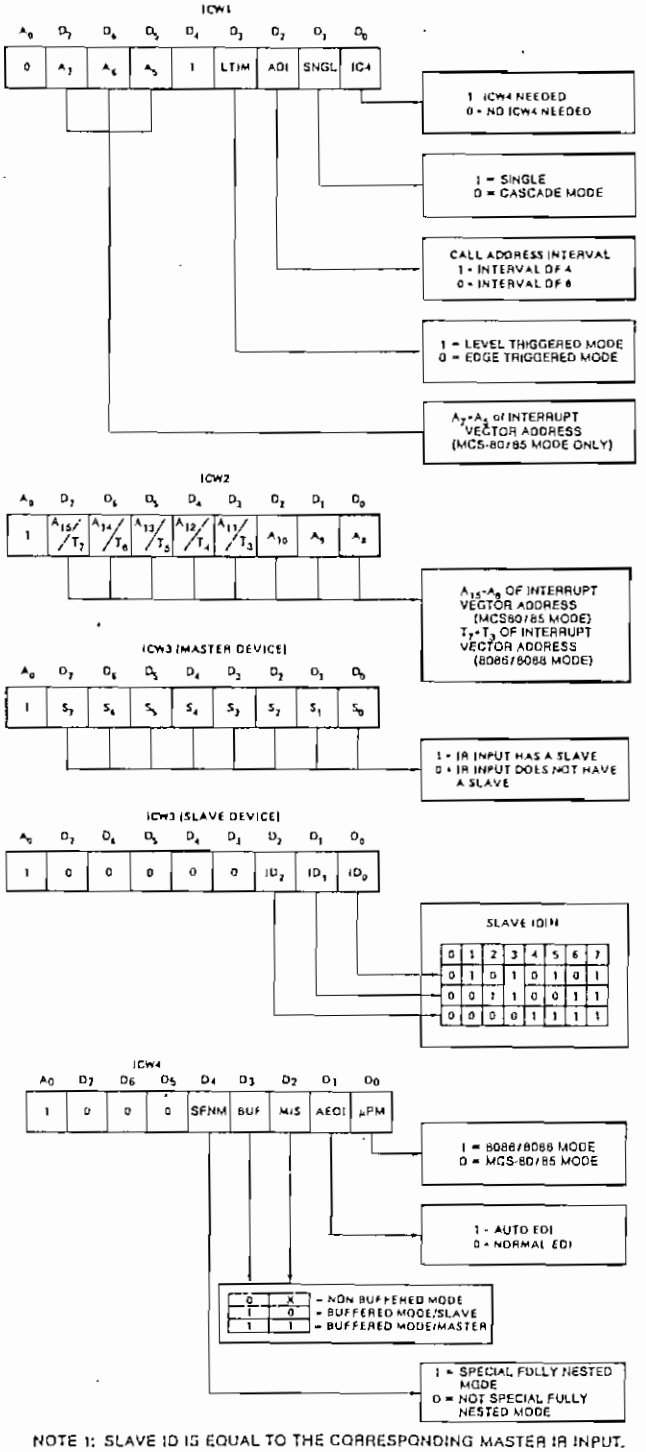


Figure 1. Initialization Sequence



NOTE 1: SLAVE ID IS EQUAL TO THE CORRESPONDING MASTER IR INPUT.

Initialization Command Word Format

OPERATION COMMAND WORDS (OCWs)

After the Initialization Command Words (ICWs) are programmed into the 8259A, the chip is ready to accept interrupt requests at its input lines. However, during the 8259A operation, a selection of algorithms can command the 8259A to operate in various modes through the Operation Command Words (OCWs).

OPERATION CONTROL WORDS (OCWs)

	OCW1							
A0	D7	D6	D5	D4	D3	D2	D1	D0
1	M7	M6	M5	M4	M3	M2	M1	M0

	OCW2							
0	R	SL	EOI	0	0	L2	L1	L0

	OCW3							
0	0	ESMM	SMM	0	1	P	RR	RIS

OPERATION CONTROL WORD 1 (OCW1)

OCW1 sets and clears the mask bits in the Interrupt Mask Register (IMR). $M_7 - M_0$ represent the eight mask bits. M = 1 indicates the channel is masked (inhibited), M = 0 indicates the channel is enabled.

OPERATION CONTROL WORD 2 (OCW2)

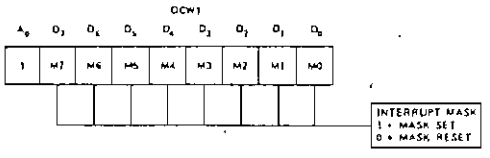
R, SL, EOI — These three bits control the Rotate and End If Interrupt modes and combinations of the two. A chart of these combinations can be found on the Operation Command Word Format.

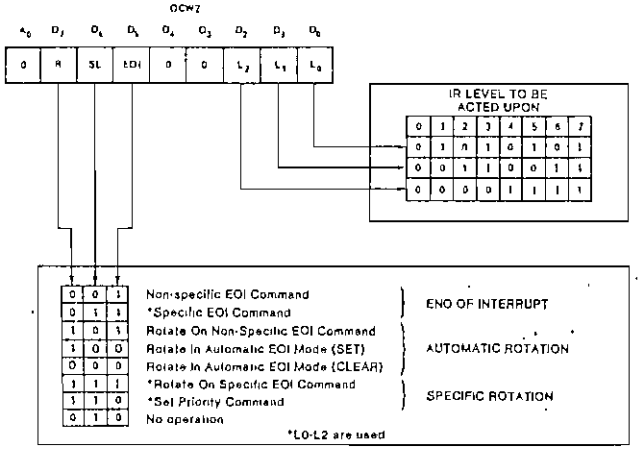
L_2, L_1, L_0 — These bits determine the interrupt level acted upon when the SEOI bit is active.

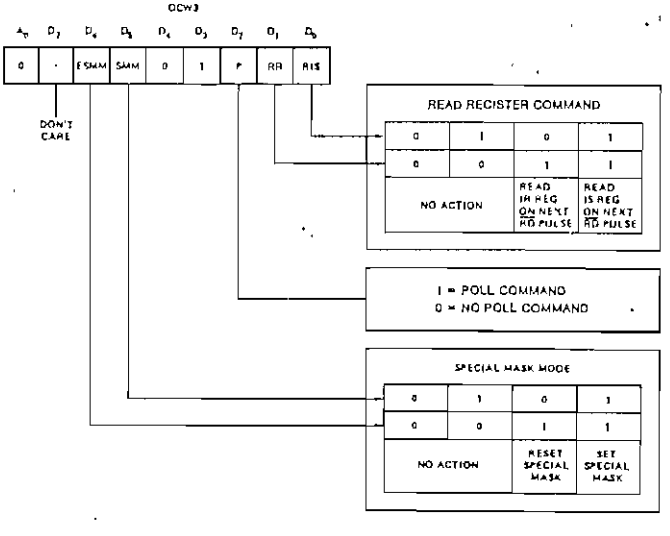
OPERATION CONTROL WORD 3 (OCW3)

ESMM — Enable Special Mask Mode. When this bit is set to 1 it enables the SMM bit to set or reset the Special Mask Mode. When ESMM = 0 the SMM bit becomes a "don't care".

SMM — Special Mask Mode. If ESMM = 1 and SMM = 1 the 8259A will enter Special Mask Mode. If ESMM = 1 and SMM = 0 the 8259A will revert to normal mask mode. When ESMM = 0, SMM has no effect.







Operation Command Word Format

INTERRUPT MASKS

Each Interrupt Request input can be masked individually by the Interrupt Mask Register (IMR) programmed through OCW1. Each bit in the IMR masks one Interrupt channel if it is set (1). Bit 0 masks IR0, Bit 1 masks IR1 and so forth. Masking an IR channel does not affect the other channels operation.

SPECIAL MASK MODE

Some applications may require an interrupt service routine to dynamically alter the system priority structure during its execution under software control. For example, the routine may wish to inhibit lower priority requests for a portion of its execution but enable some of them for another portion.

The difficulty here is that if an Interrupt Request is acknowledged and an End of Interrupt command did not reset its IS bit (i.e., while executing a service routine), the 8259A would have inhibited all lower priority requests with no easy way for the routine to enable them

That is where the Special Mask Mode comes in. In the special Mask Mode, when a mask bit is set in OCW1, it inhibits further interrupts at that level *and enables* interrupts from *all other* levels (lower as well as higher) that are not masked.

Thus, any interrupts may be selectively enabled by loading the mask register.

The special Mask Mode is set by OCW3 where: SMM = 1, SMM = 1, and cleared where SMM = 1, SMM = 0.

BUFFERED MODE

When the 8259A is used in a large system where bus driving buffers are required on the data bus and the cascading mode is used, there exists the problem of enabling buffers.

The buffered mode will structure the 8259A to send an enable signal on $\overline{SP}/\overline{EN}$ to enable the buffers. In this mode, whenever the 8259A's data bus outputs are enabled, the $\overline{SP}/\overline{EN}$ output becomes active.

This modification forces the use of software programming to determine whether the 8259A is a master or a slave. Bit 3 in ICW4 programs the buffered mode, and bit 2 in ICW4 determines whether it is a master or a slave.

FULLY NESTED MODE

This mode is entered after initialization unless another mode is programmed. The interrupt requests are ordered in priority form 0 through 7 (0 highest). When an interrupt is acknowledged the highest priority request is determined and its vector placed on the bus. Additionally, a bit of the Interrupt Service register (ISO-7) is set. This bit remains set until the microprocessor issues an End of Interrupt (EOI) command immediately before returning from the service routine, or if AEOI (Automatic End of Interrupt) bit is set, until the trailing edge of the last INTA. While the IS bit is set, all further interrupts of the same or lower priority are inhibited, while higher levels will generate an interrupt (which will be acknowledged only if the microprocessor internal interrupt enable flip-flop has been re-enabled through software).

After the initialization sequence, IR0 has the highest priority and IR7 the lowest. Priorities can be changed, as will be explained, by priority rotation.

THE SPECIAL FULLY NESTED MODE

This mode will be used in the case of a big system where cascading is used, and the priority has to be conserved within each slave. In this case the special fully nested mode will be programmed to the master (using) ICW4). This mode is similar to the normal fully nested mode with the following exceptions:

a. When an interrupt request from a certain slave is in service this slave is not locked out from the master's priority logic and further interrupt requests from higher priority IR's within the slave will be recognized by the master and will initiate interrupts to the processor. (In the normal nested mode a slave is masked out when its request is in service and no higher requests from the same slave can be serviced.)

b. When exiting the Interrupt Service routine the software has to check whether the interrupt serviced was the only one from that slave. This is done by sending a non-specific End of Interrupt (EOI) command to the slave and then reading its In-Service register and checking for zero. If it is empty, a non-specific EOI can be sent to the master too. If not, no EOI should be sent.

POLL

In this mode the microprocessor internal Interrupt Enable flip-flop is reset, disabling its interrupt input. Service to devices is achieved by programmer initiative using a Poll command.

The Poll command is issued by setting P = "1" in OCW3. The 8259A treats the next $\overline{RD}$ pulse to the 8259A (i.e., $\overline{RD} = 0$, $\overline{CS} = 0$) as an interrupt acknowledge, sets the appropriate IS bit if there is a request, and reads the priority level. Interrupt is frozen from $\overline{WR}$ to $\overline{RD}$.

The word enabled onto the data bus during $\overline{RD}$ is:

D7	D6	D5	D4	D3	D2	D1	D0
I	—	—	—	—	W2	W1	W0

W0-W2: Binary code of the highest priority level requesting service.

I: Equal to a "1" if there is an interrupt.

This mode is useful if there is a routine command common to several levels so that the $\overline{INTA}$ sequence is not needed (saves ROM space). Another application is to use the poll command to expand the number of priority levels to more than 64.

END OF INTERRUPT (EOI)

The In Service (IS) bit can be reset either automatically following the trailing edge of the last in sequence $\overline{INTA}$ pulse (when AEOI bit in ICW1 is set) or by a command word that must be issued to the 8259A before returning from a service routine (EOI command). An EOI command must be issued twice, once for themaster and once for the corresponding slave if slaves are in use.

There are two forms of EOI command: Specific and Non-Specific. When the 8259A is operated in modes which preserve the fully nested structure, it can determine which IS bit to reset on EOI. When a Non-Specific EOI command is issued the 8259A will automatically reset the highest IS bit of those that are set, since in the nested mode the highest IS level was necessarily the last level acknowledged and serviced.

However, when a mode is used which may disturb the fully nested structure, the 8259A may no longer be able to determine the last level acknowledged. In this case a Specific End of Interrupt (SEOI) must be issued which includes as part of the command the IS level to be reset. EOI is issued whenever EOI = 1, in OCW2, where L0-L2 is the binary level of the IS bit to be reset. Note that although the Rotate command can be issued together with an EOI where EOI = 1, it is not necessarily tied to it.

It should be noted that an IS bit that is masked by an IMR bit will not be cleared by a non-specific EOI if the 8259A is in the Special Mask Mode.

AUTOMATIC END OF INTERRUPT (AEOI) MODE

If AEOI = 1 in ICW4, then the 8259A will operate in AEOI mode continuously until reprogrammed by ICW4. In this mode the 8259A will automatically perform a non-specific EOI operation at the trailing edge of the last interrupt acknowledge pulse (third pulse in MCS-80/85, second in MCS-86). Note that from a system standpoint, this mode should be used only when a nested multilevel interrupt structure is not required within a single 8259A.

To achieve automatic rotation within AEOI, there is a special rotate flip-flop. It is set by OCW2 with R = 1, SL = 0, EOI = 0, and cleared with R = 0, SEOI = 0, EOI = 0.

AUTOMATIC ROTATION
(Equal Priority Devices)

In some applications there are a number of interrupting devices of equal priority. In this mode a device, after being serviced, receives the lowest priority, so a device requesting an interrupt will have to wait, in the worst case until each of 7 other devices are serviced at most once. For example, if the priority and "in service" status is:

Before Rotate (IR4 the highest priority requiring service)

	IS7	IS6	IS5	IS4	IS3	IS2	IS1	IS0
"IS" Status	0	1	0	1	0	0	0	0
	Lowest Priority							Highest Priority
Priority Status	7	6	5	4	3	2	1	0

After Rotate (IR4 was serviced, all other priorities rotated correspondingly)

	IS7	IS6	IS5	IS4	IS3	IS2	IS1	IS0
"IS" Status	0	1	0	0	0	0	0	0
	Highest Priority							Lowest Priority
Priority Status	2	1	0	7	6	5	4	3

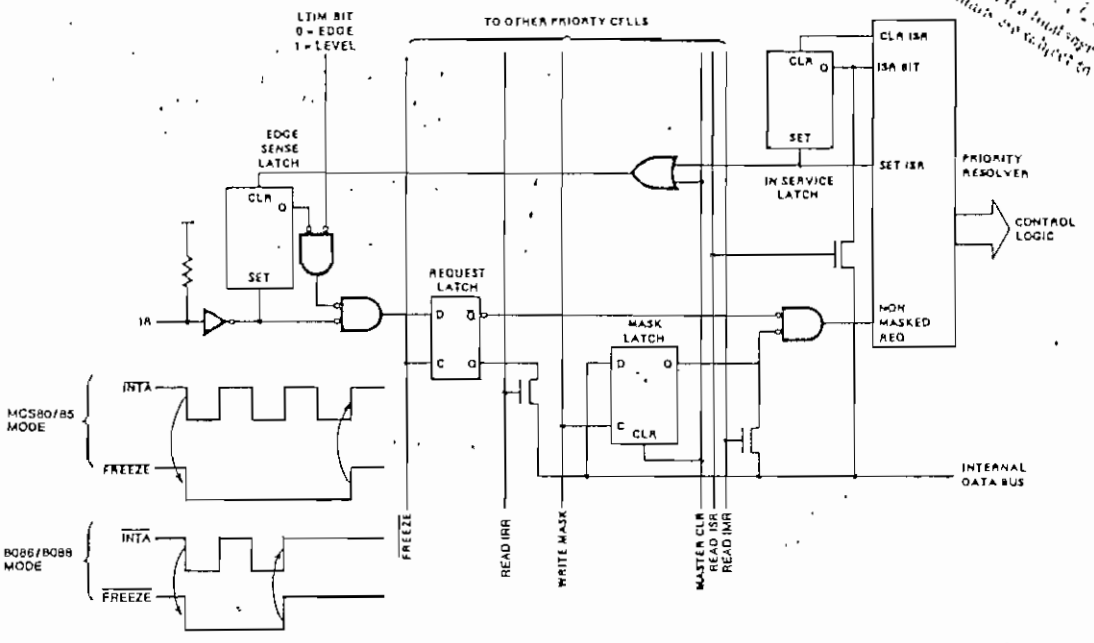
There are two ways to accomplish Automatic Rotation using OCW2, the Rotate on Non-Specific EOI Command (R = 1, SL = 0, EOI = 1) and the Rotate in Automatic EOI Mode which is set by (R = 1, SL = 0, EOI = 0) and cleared by (R = 0 SL = 0, EOI = 0).

SPECIFIC ROTATION
(Specific Priority)

The programmer can change priorities by programming the bottom priority and thus fixing all other priorities; i.e., if IR5 is programmed as the bottom priority device, then IR6 will have the highest one.

The Set Priority command is issued in OCW2 where: R = 1, SEOI = 1; L0-L2 is the binary priority level code of the bottom priority device.

Observe that in this mode internal status is updated by software control during OCW2. However, it is independent of the End of Interrupt (EOI) command (also executed by OCW2). Priority changes can be executed during an EOI command by using the Rotate on Specific EOI Command in OCW2 (R = 1, SL = 1, EOI = 1 and L0-L2 = IR level to receive bottom priority).



NOTES
1. MASTER CLEAR ACTIVE ONLY DURING ICW1
2. FREEZE/IS ACTIVE DURING $\overline{INTA}$ AND POLL SEQUENCES ONLY
3. TRUTH TABLE FOR Q-LATCH

C	D	Q	OPERATION
1	Q_i	Q_i	FOLLOW
0	X	Q_{n-1}	HOLD

Priority Cell — Simplified Logic Diagram

LEVEL TRIGGERED MODE

This mode is programmed using bit 3 in ICW1.

If LTIM = '1,' an interrupt request will be recognized by a 'high' level on IR input, and there is no need for an edge detection. The interrupt request must be removed before the EOI command is issued or the CPU interrupt is enabled to prevent a second interrupt from occurring.

The above figure shows a conceptual circuit to give the reader an understanding of the level sensitive and edge sensitive input circuitry of the 8259A. Be sure to note that the request latch is a transparent D type latch.

READING THE 8259A STATUS

The input status of several internal registers can be read to update the user information on the system. The following registers can be read via OCW3 (IRR and ISR or OCW1 (IMR).

Interrupt Request Register (IRR): 8-bit register which contains the levels raquesting an interrupt to be acknowledged. The highest request level is reset from the IRR when an interrupt is acknowledged. (Not affected by IMR).

In-Service Register (ISR): 8-bit register which contains the priority levels that are being serviced. The ISR is updated when an End of Interrupt command is issued.

Interrupt Mask Register: 8-bit register which contains the interrupt request lines which are masked.

The IRR can be read when, prior to the $\overline{RD}$ pulse, a Read Register Command is issued with OCW3 (RR = 1, RIS = 0).

The ISR can be read when, prior to the $\overline{RD}$ pulse, a Read Register Command is issued with OCW3 (RR = 1, RIS = 1).

There is no need to write an OCW3 before every status read operation, as long as the status read corresponds with the previous one; i.e., the 8259A "remembers" whether the IRR or ISR has been previously selected by the OCW3. This is not true when poll is used.

After initialization the 8259A is set to IRR.

For reading the IMR, no OCW3 is needed. The output data bus will contain the IMR whenever $\overline{RD}$ is active and A0 = 1 (OCW1).

Polling overrides status read when P = 1, RR = 1 in OCW3.

SUMMARY OF 8259A INSTRUCTION SET

Inst. #	Mnemonic		A0	D7	D6	D5	D4	D3	D2	D1	D0	Operation Description	
1	ICW1	A	0	A7	A6	A5	1	0	1	1	0		Format = 4, single, edge triggered
2	ICW1	B	0	A7	A6	A5	1	1	1	1	0		Format = 4, single, level triggered
3	ICW1	C	0	A7	A6	A5	1	0	1	0	0	Byte 1 Initialization	Format = 4, not single, edge triggered
4	ICW1	D	0	A7	A6	A5	1	1	1	0	0		Format = 4, not single, level triggered
5	ICW1	E	0	A7	A6	0	1	0	0	1	0	No ICW4 Required	Format = 8, single, edge triggered
6	ICW1	F	0	A7	A6	0	1	1	0	1	0		Format = 8, single, level triggered
7	ICW1	G	0	A7	A6	0	1	0	0	0	0		Format = 8, not single, edge triggered
8	ICW1	H	0	A7	A6	0	1	1	0	0	0		Format = 8, not single, level triggered
9	ICW1	I	0	A7	A6	A5	1	0	1	1	1		Format = 4, single, edge triggered
10	ICW1	J	0	A7	A6	A5	1	1	1	1	1		Format = 4, single, level triggered
11	ICW1	K	0	A7	A6	A5	1	0	1	0	1	Byte 1 Initialization	Format = 4, not single, edge triggered
12	ICW1	L	0	A7	A6	A5	1	1	1	0	1		Format = 4, not single, level triggered
13	ICW1	M	0	A7	A6	0	1	0	0	1	1	ICW4 Required	Format = 8, single, edge triggered
14	ICW1	N	0	A7	A6	0	1	1	0	1	1		Format = 8, single, level triggered
15	ICW1	O	0	A7	A6	0	1	0	0	0	1		Format = 8, not single, edge triggered
16	ICW1	P	0	A7	A6	0	1	1	0	0	1		Format = 8, not single, level triggered
17	ICW2		1	A15	A14	A13	A12	A11	A10	A9	A8	Byte 2 Initialization	
18	ICW3	M	1	S7	S6	S5	S4	S3	S2	S1	S0	Byte 3 initialization — master	
19	ICW3	S	1	0	0	0	0	0	S2	S1	S0	Byte 3 initialization — slave	
20	ICW4	A	1	0	0	0	0	0	0	0	0	No action, redundant	
21	ICW4	B	1	0	0	0	0	0	0	0	1	Non-buffered mode, no AEOI, 8086/8088	
22	ICW4	C	1	0	0	0	0	0	0	1	0	Non-buffered mode, AEOI, MCS-80/85	
23	ICW4	D	1	0	0	0	0	0	0	1	1	Non-buffered mode, AEOI, 8086/8088	
24	ICW4	E	1	0	0	0	0	0	1	0	0	No action, redundant	
25	ICW4	F	1	0	0	0	0	0	1	0	1	Non-buffered mode, no AEOI, 8086/8088	
26	ICW4	G	1	0	0	0	0	0	1	1	0	Non-buffered mode, AEOI, MCS-80/85	
27	ICW4	H	1	0	0	0	0	0	1	1	1	Non-buffered mode, AEOI, 8086/8088	
28	ICW4	I	1	0	0	0	0	1	0	0	0	Buffered mode, slave, no AEOI, MCS-80/85	
29	ICW4	J	1	0	0	0	0	1	0	0	1	Buffered mode, slave, no AEOI, 8086/8088	
30	ICW4	K	1	0	0	0	0	1	0	1	0	Buffered mode, slave, AEOI, MCS-80/85	
31	ICW4	L	1	0	0	0	0	1	0	1	1	Buffered mode, slave, AEOI, 8086/8088	
32	ICW4	M	1	0	0	0	0	1	1	0	0	Buffered mode, master, no AEOI, MCS-80/85	
33	ICW4	N	1	0	0	0	0	1	1	0	1	Buffered mode, master, no AEOI, 8086/8088	
34	ICW4	O	1	0	0	0	0	1	1	1	0	Buffered mode, master, AEOI, MCS-80/85	
35	ICW4	P	1	0	0	0	0	1	1	1	1	Buffered mode, master AEOI, 8086, 8088	
36	ICW4	NA	1	0	0	0	1	0	0	0	0	Fully nested mode, MCS-80, non buffered, no AEOI	
37	ICW4	NB	1	0	0	0	1	0	0	0	1	ICW4 NB through ICW4 ND are identical to	
38	ICW4	NC	1	0	0	0	1	0	0	1	0	ICW4 B through ICW4 D with the addition of	
39	ICW4	ND	1	0	0	0	1	0	0	1	1	Fully Nested Mode	
40	ICW4	NE	1	0	0	0	1	0	1	0	0	Fully Nested Mode, MCS-80/85, non-buffered, no AEOI	
41	ICW4	NF	1	0	0	0	1	0	1	0	1		
42	ICW4	NG	1	0	0	0	1	0	1	1	0		
43	ICW4	NH	1	0	0	0	1	0	1	1	1		
44	ICW4	NI	1	0	0	0	1	1	0	0	0		
45	ICW4	NJ	1	0	0	0	1	1	0	0	1		
46	ICW4	NK	1	0	0	0	1	1	0	1	0	ICW4 NF through ICW4 NP are identical to	
47	ICW4	NL	1	0	0	0	1	1	0	1	1	ICW4 F through ICW4 P with the addition of	
48	ICW4	NM	1	0	0	0	1	1	1	0	0	Fully Nested Mode	
49	ICW4	NN	1	0	0	0	1	1	1	0	1		
50	ICW4	NO	1	0	0	0	1	1	1	1	0		
51	ICW4	NP	1	0	0	0	1	1	1	1	1		
52	OCW1		1	M7	M6	M5	M4	M3	M2	M1	M0	Load mask register, read mask register	
53	OCW2	E	0	0	0	1	0	0	0	0	0	Non-specific EOI	
54	OCW2	SE	0	0	1	1	0	0	L2	L1	L0	Specific EOI, L0-L2 code of IS FF to be reset	
55	OCW2	RE	0	1	0	1	0	0	0	0	0	Rotate on Non-Specific EOI	
56	OCW2	RSE	0	1	1	1	0	0	L2	L1	L0	Rotate on Specific EOI L0-L2 code of line	
57	OCW2	R	0	1	0	0	0	0	0	0	0	Rotate in Auto EOI (set)	
58	OCW2	CR	0	0	0	0	0	0	0	0	0	Rotate in Auto EOI (clear)	
59	OCW2	RS	0	1	1	0	0	0	L2	L1	L0	Set Priority Command	
60	OCW3	P	0	0	0	0	0	1	1	0	0	Poll mode	
61	OCW3	RIS	0	0	0	0	0	1	0	1	1	Read IS register	

SUMMARY OF 8259A INSTRUCTION SET (Cont.)

Inst. #	Mnemonic	A0	D7	D6	D5	D4	D3	D2	D1	D0	Operation Description
46	OCW3 RR	0	0	0	0	0	1	0	1	0	Read request register
47	OCW3 SM	0	0	1	1	0	1	0	0	0	Set special mask mode
48	OCW3 RSM	0	0	1	0	0	1	0	0	0	Reset special mask mode

Note: 1 In the master mode $\overline{SP}$ pin = 1, in slave mode $\overline{SP}$ = 0

Cascading

The 8259A can be easily interconnected in a system of one master with up to eight slaves to handle up to 64 priority levels.

A typical MCS-80/85 system is shown in Figure 2. The master controls, through the 3 line cascade bus, which one of the slaves will release the corresponding address.

As shown in Figure 2, the slave interrupt outputs are connected to the master interrupt request inputs. When a slave request line is activated and afterwards acknowledged, the master will enable the corresponding slave to release the device routine address during bytes 2 and 3 of $\overline{INTA}$. (Byte 2 only for 8086/8088).

The cascade bus lines are normally low and will contain the slave address code from the trailing edge of the first $\overline{INTA}$ pulse to the trailing edge of the third pulse. It is obvious that each 8259A in the system must follow a separate initialization sequence and can be programmed to work in a different mode. An EOI command must be issued twice: once for the master and once for the corresponding slave. An address decoder is required to activate the Chip Select ($\overline{CS}$) input of each 8259A.

The cascade lines of the Master 8259A are activated for any interrupt input, even if no slave is connected to that input.

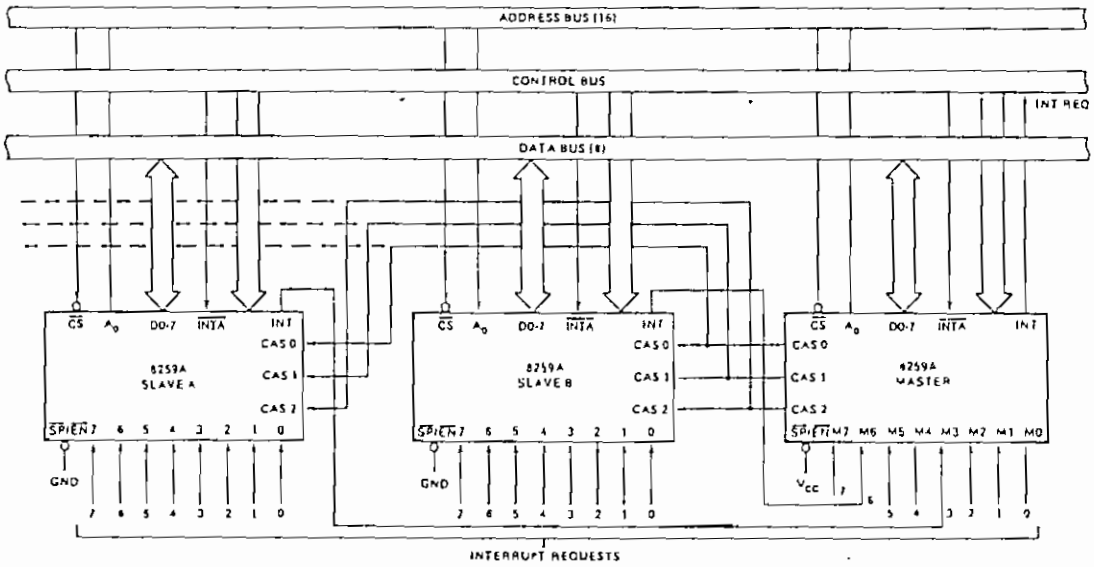


Figure 2. Cascading the 8259A

PIN FUNCTIONS

NAME	I/O	PIN#	FUNCTION
V_{CC}	I	28	+5v supply
GND	I	14	Ground
$\overline{CS}$	I	1	Chip Select A low on this pin enables RD and WR communication between the CPU and the 8259A. INTA functions are independent of CS.
$\overline{WR}$	I	2	Write: A low on this pin when CS is low, enables the 8259A to accept command words from the CPU.
$\overline{RD}$	I	3	Read: A low on this pin when CS is low enables the 8259A to release status onto the data bus for the CPU.
D_7-D_0	I/O	4-11	Bidirectional Data Bus: Control, status and interrupt-vector information is transferred via this bus.
CAS_0-CAS_2	I/O	12,13,15	Cascade Lines: The CAS lines form a private 8259A bus to control a multiple 8259A structure. These pins are outputs for a master 8259A and inputs for a slave 8259A.
$\overline{SP}/\overline{EN}$	I/O	16	Slave Program/Enable Buffer: This is a dual function pin. When in the Buffered Mode it can be used as an output to control buffer transceivers (EN). When not in the buffered mode it is used as an input to designate a master (SP = 1) or slave (SP = 0).
INT	O	17	Interrupt: This pin goes high whenever a valid interrupt request is asserted. It is used to interrupt the CPU, thus it is connected to the CPU's interrupt pin.
IR_0-IR_7	I	18-25	Interrupt Requests: Asynchronous inputs. An interrupt request can be generated by raising an IR input (low to high) and holding it high until it is acknowledged (Edge Triggered Mode), or just by a high level on an IR input (Level Triggered Mode).
$\overline{INTA}$	I	26	Interrupt Acknowledge: This pin is used to enable 8259A interrupt-vector data onto the data bus. This is done by a sequence of interrupt acknowledge pulses issued by the CPU.
A_0	I	27	A0 Address Line: This pin acts in conjunction with the $\overline{CS}$, $\overline{WR}$, and $\overline{RD}$ pins. It is used by the 8259A to decipher between various Command Words the CPU writes and status the CPU wishes to read. It is typically connected to the CPU A0 address line (A1 for 8086/8088).

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias ..... −40°C to 85°C
Storage Temperature ..... −65°C to +150°C
Voltage On Any Pin
With Respect to Ground ..... −0.5V to +7V
Power Dissipation ..... 1 Watt

*COMMENT
Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

D.C. CHARACTERISTICS

$T_A = 0°C$ to $70°C$, $V_{CC} = 5V \pm 10\%$ (8259-A), $V_{CC} = 5V \pm 10\%$ (8259A)

Symbol	Parameter	Min.	Max.	Units	Test Conditions
V_{IL}	Input Low Voltage	-.5	V		
V_{IH}	Input High Voltage	2.0	V_{CC} + .5V	V	
V_{OL}	Output Low Voltage		.45	V	$I_{OL} = 2.2$ mA
V_{OH}	Output High Voltage	2.4		V	$I_{OH} = -400\ \mu A$
$V_{OH(INT)}$	Interrupt Output High Voltage	3.5 2.4		C V	$I_{OH} = -100\ \mu A$ $I_{OH} = -400\ \mu A$
I_{LI}	Input Load Current		10	μA	$V_{IN} = V_{CC}$ to 0V
I_{LOL}	Output Leakage Current		-10	μA	$V_{OUT} = 0.45V$
I_{CC}	V_{CC} Supply Current		85	mA	
I_{LIR}	IR Input Load Current		-300 10	μA μA	$V_{IN} = 0$ $V_{IN} = V_{CC}$

8259A A.C. CHARACTERISTICS

$T_A = 0°C$ to $70°C$ $V_{CC} = 5V \pm 5\%$ (8259A-8) $V_{CC} = 5V \pm 10\%$ (8259A)

TIMING REQUIREMENTS

Symbol	Parameter	8259A-8		8259A		8259A-2		Units	Test Conditions
		Min.	Max.	Min.	Max.	Min.	Max.		
TAHRL	A0/$\overline{CS}$ Setup to $\overline{RD}$/$\overline{INTA}$	50		0		0		ns	
TRHAX	A0/$\overline{CS}$ Hold after $\overline{RD}$/$\overline{INTA}$	5		0		0		ns	
TRLRH	$\overline{RD}$ Pulse Width	420		235		160		ns	
TAHWL	A0/$\overline{CS}$ Setup to $\overline{WR}$	50		0		0		ns	
TWHAX	A0/$\overline{CS}$ Hold after $\overline{WR}$	20		0		0		ns	
TWLWH	$\overline{WR}$ Pulse Width	400		290		190		ns	
TDVWH	Data Setup to $\overline{WR}$	300		240		160		ns	
TWHDX	Data Hold after $\overline{WR}$	40		0		0		ns	
TJLJH	Interrupt Request Width (Low)	100		100		100		ns	See Note 1
TCVIAL	Cascade Setup to Second or Third $\overline{INTA}$ (Slave Only)	55		55		40		ns	
TRHRL	End of $\overline{RD}$ to Next Command	160		160		160		ns	
TWHRL	End of $\overline{WR}$ to Next Command	190		190		190		ns	

Note: This is the low time required to clear the input latch in the edge triggered mode.

TIMING RESPONSES

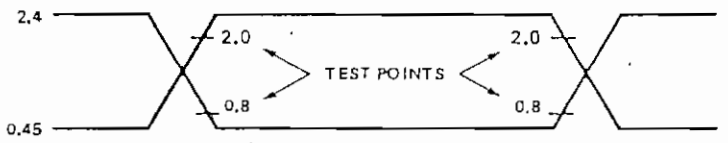
Symbol	Parameter	8259A-8		8259A		8259A-2		Units	Test Conditions
		Min.	Max.	Min.	Max.	Min.	Max.		
TRLDV	Data Valid from $\overline{RD}$/$\overline{INTA}$		300		200		120	ns	C of Data Bus = 100 pF
TRHDZ	Data Float after $\overline{RD}$/$\overline{INTA}$	10	200		100		85	ns	C of Data Bes Max Iext C = 100 pF Min. Iext C = 15 pF
TJHIH	Interrupt Output Delay		400		350		300	ns	
TIAHCV	Cascade Valid from First $\overline{INTA}$ (Master Only)		565		565		360	ns	C_{INT} = 100 pF
TRLEL	Enable Active from $\overline{RD}$ or $\overline{INTA}$		160		125		100	ns	$C_{CASCADE}$ = 100 pF
TRHEH	Enable Inactive from $\overline{RD}$ or $\overline{INTA}$		325		150		d150	ns	
TAHDV	Data Valid from Stable Address		350		200		200	ns	
TCVDV	Cascade Valid to Valid Data		300		300		200	ns	

CAPACITANCE

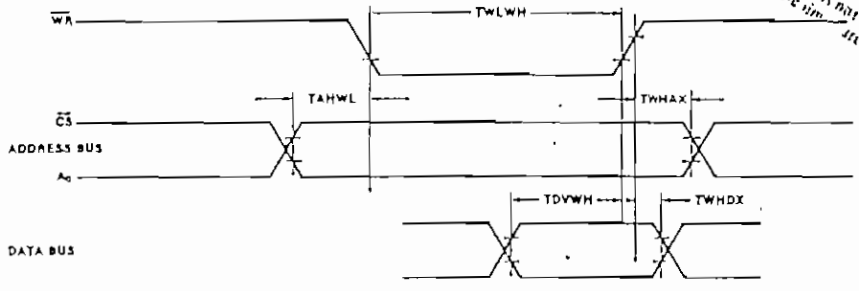
$T_A = 25°C$; $V_{CC} = GND = 0V$

Symbol	Parameter	Min.	Typ.	Max.	Unit	Test Conditions
C_{IN}	Input Capacitance			10	pF	fc = 1 MHz
$C_{I/O}$	I/O Capacitance			20	pF	Unmeasured pins returned to V_{SS}

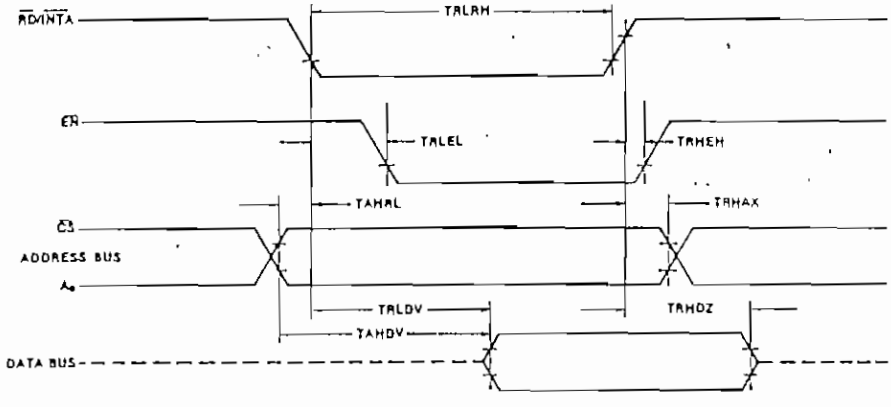
Input and Output Waveforms for A.C. Tests



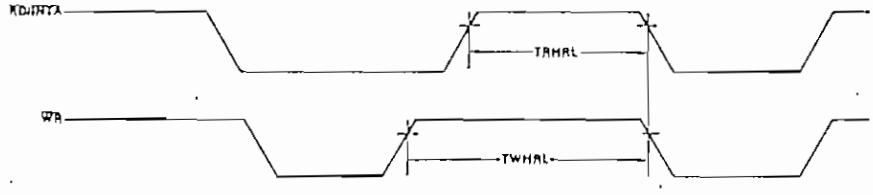
WRITE MODE



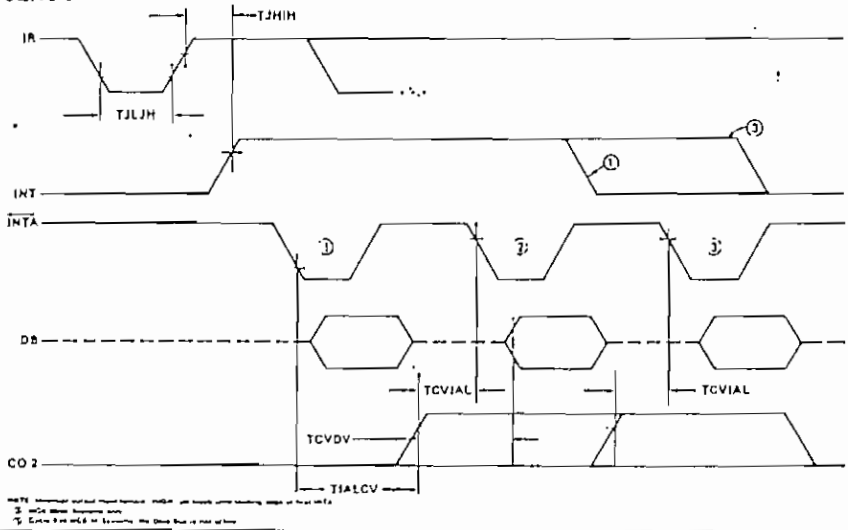
READ/INTA MODE



OTHER TIMING



INTA SEQUENCE



C1. GENERADOR DE TRANSACCIONES Y FORMATOS

Para la ejecución de este proceso, se debe instalar el diskette que contiene los programas y archivos en la unidad "A" y ejecutar el comando "gen-tran".

Cuando el programa "gen-tran" toma control, el operador es guiado a través de menús y de mensajes de ayuda que le permiten definir, eliminar e imprimir pantallas y transacciones.

Cuando este programa es ejecutado por primera vez, crea los archivos y graba en los registros de control datos que permitirán controlar la asignación de registros de los diferentes archivos.

Las transacciones definidas, pueden ser ejecutadas por el programa "proc-tran".

C3 PROGRAMA DE CONTROL DEL TERMINAL

Para que el PC trabaje como terminal el operador debe haber cargado a memoria el driver de comunicaciones, y posteriormente digitar el comando "proc-tran".

Cuando el programa "proc-tran" adquiere control, el operador puede realizar las siguientes funciones:

- reconfiguración del terminal: cambio de código de interrogación y de velocidad,
- habilitación del terminal,
- terminar las funciones de terminal.

las funciones arriba indicadas se realizan mediante las teclas de control de acuerdo a indicaciones dadas por el programa al operador.

La ejecución de una transacción se realiza en los siguientes pasos:

- el operador debe ingresar el código de una transacción existente.
- el programa presenta el formato de la pantalla asociada a esta transacción.
- el operador debe ingresar los datos solicitados por el programa de acuerdo a los campos de entrada definidos para ese formato.

BIBLIOGRAFIA

- PROGRAMMING THE 8086 8088. JAMES COFFRON SYBEX INC. BERKELEY CALIFORNIA
- EL MICROPROCESADOR 8086/8088. A. FONTAINE MASSON. BARCELONA
- NCR PC4i OWNER'S MANUAL
- NCR PC4i SERVICE MANUAL
- NCR PC4i GW-BASIC
- BASCOM MICROSOFT
- MACRO ASSEMBLER MICROSOFT
- NCR 270 REFERENCE MANUAL
- NCR 796 REFERENCE MANUAL
- NCR 6640 REFERENCE MANUAL
- NCR 2261 REFERENCE MANUAL
- NCR 2262 REFERENCE MANUAL
- NCR ITX TRANPRO REFERENCE MANUAL
- NCR COMTEM 721 REFERENCE MANUAL