



CENTRO DE INVESTIGACION Y DE ESTUDIOS AVANZADOS DEL
INSTITUTO POLITECNICO NACIONAL

DEPARTAMENTO DE INGENIERIA ELECTRICA

SECCION DE COMPUTACION

CENTRO DE INVESTIGACION Y DE
ESTUDIOS AVANZADOS DEL
I. P. N.
BIBLIOTECA
INGENIERIA ELECTRICA

IMPLANTACION DE LAS RECOMENDACIONES

X.3, X.28 Y X.29 DEL CCITT

A UN ENSAMBLADOR/DESENSAMBLADOR DE DATOS (EDD)

CENTRO DE INVESTIGACION Y DE
ESTUDIOS AVANZADOS DEL
I. P. N.
BIBLIOTECA

Tesis que presenta el Ing. Andrés Vega García para obtener el grado de MAESTRO EN CIENCIAS en la especialidad de INGENIERIA ELECTRICA con opción en COMPUTACION.

Trabajo dirigido por el Dr. Manuel E. Guzmán Rentería.

México, D.F., a 1 de Diciembre de 1989.

G R A C I A S

Al Centro de Investigación y de Estudios Avanzados del I.P.N.
por el apoyo que como institución ha representado.

Al Dr. Manuel E. Guzmán Rentería por su asesoría y confianza
en mí depositada.

A todos mis profesores y compañeros de clase.

A mi amigo y compañero de trabajo Rodolfo Rosado por todas
las dificultades técnicas que tuvimos que enfrentar, y por la
disposición que siempre tuvo para emprender nuevas y laboriosas
modificaciones al prototipo de EDD.

A mi amigo y compañero de trabajo Jorge Buenabad por todos
sus consejos y sugerencias que fueron más allá de lo relativo a
los aspectos técnicos.

CENTRO DE INVESTIGACION Y DE ESTUDIOS AVANZADOS
ESTUDIOS AVANZADOS 123
I.P.N.
BIBLIOTECA
INGENIERIA ELECTRICA

Con todo cariño y profundo respeto, dedico el logro de este trabajo, a MIS PADRES: por haber sido el imán que me atrajo a este medio ambiente; porque me han legado la preciosa oportunidad de vivir, oportunidad llena de todo lo que necesito para entrenar y enfrentarme a todo aquello que me llevará, día a día, una medida más arriba en esa inmensa montaña, montaña a cuya cúspide sólo llega el "hombre".

Indice

INDICE	i
OBJETIVO	ii
I. INTRODUCCION	I-1
I.2. Modelo OSI	I-2
I.3. El modelo OSI y la arquitectura del EDD	I-4
I.4. Normas del CCITT para un EDD	I-6
II. RECOMENDACIONES X.3, X.28 Y X.29	II-1
II.1. Parámetros de terminal (X.3)	II-1
II.2. Parámetros de terminal extras	II-9
II.3. Lenguaje de comunicación EDD-Usuarios (X.28)	II-12
II.4. Señales de instrucción y de servicio adicionales al X.28	II-20
II.5. Comandos remotos (X.29)	II-23
III. NIVEL DE TRANSPORTE (INTERFAZ CON EL NIVEL DE RED)	III-1
III.1. Comandos del nivel de transporte	III-3
III.2. Indicaciones del nivel de transporte	III-9
IV. ARQUITECTURA DEL SOFTWARE	IV-1
IV.1. Organización lógica de los módulos de software	IV-1
IV.2. Esquema de intercambio de información y procesos involucrados	IV-3
IV.3. Proceso transmisor/ receptor de caracteres	IV-6
IV.4. Proceso receptor de datos remotos	IV-12
IV.5. Proceso manejador de mensajes de EDD	IV-14
IV.6. Manejador de terminal	IV-17
IV.7. Implantación de la recomendación X.3	IV-20
IV.8. Implantación de las recomendaciones X.28 y X.29	IV-27
IV.9. Teclado y desplegado	IV-28
V. PRUEBAS Y CONCLUSIONES	V-1
BIBLIOGRAFIA	
APENDICE A. ESQUEMA GENERAL DEL HARDWARE	
APENDICE B. TERMINOS	

OBJETIVO

El presente trabajo forma parte de la segunda fase de un proyecto que pretende arrojar un producto comercial en el área de transmisión de datos, el cual tiene en mira los siguientes objetivos:

i) Obtener dos prototipos con el soporte de circuitería necesario para soportar todos los requerimientos que más tarde habría de tener la programación para la implantación de las recomendaciones.

ii) El manejo de hasta 8 terminales asincronas a una velocidad de hasta 19,200 bps.

iii) Capacidad de multiplexaje y demultiplexaje siguiendo la recomendación X.25 del CCITT.

iv) Manejo de un conjunto de parámetros que definen la manera de interactuar del EDD con cada terminal, definido por la recomendación X.3 del CCITT.

v) Implantación de un lenguaje de comunicación usuario-EDD a través de señales de instrucción y señales de servicio de acuerdo a la recomendación X.28 del CCITT.

vi) Implantación de un lenguaje de comunicación remoto entre el EDD y un Equipo Terminal de Datos Remoto que maneja paquetes de X.25, definido por la recomendación X.29 del CCITT.

vii) Implantación de las facilidades de un desplegado alfanumérico y de un teclado que permita examinar/modificar los parámetros de terminal.

La primer fase consistió en lo siguiente:

- * El diseño y construcción de la circuitería, objetivos (i) y (ii).
- * La implantación del nivel de enlace y nivel de paquetes de X.25, objetivo (ii).
- * El logro del objetivo (iii), así como la parte mínima necesaria del objetivo (iv).

Mi participación en este proyecto abarcó tanto aspectos de la circuitería como de programación, en la parte de circuitería, diseñada por Rodolfo Rosado, trabajé en la revisión y el rediseño de una parte de la misma; por el lado de la programación, desarrollé parte del núcleo utilizado para concurrencia.

 En este trabajo limito la tesis únicamente a lo correspondiente a los objetivos restantes de lo inicialmente planteado para el proyecto, que constituyen la segunda fase del mismo, esto es, los objetivos (iv), (v), (vi) y (vii), correspondientes a las recomendaciones X.3, X.28, X.29 y al teclado y desplegado. Cabe señalar que entre las dos fases del proyecto hubo un proceso de asimilación de lo alcanzado en la primer fase, antes de poder continuar con el mismo.

I. INTRODUCCION

En un medio ambiente en el cual la necesidad de información, así como la confiabilidad de la misma y su oportuno acceso a ella son de vital importancia para el desarrollo de la sociedad en sus diferentes aspectos, se vuelve indispensable encaminar esfuerzos hacia el logro de tales objetivos. La comunicación vía satélite o la utilización de la red telefónica son algunos resultados de los esfuerzos realizados por la ingeniería hacia tal fin.

Las redes de datos son otro esfuerzo encaminado a los mismos objetivos, y cada día están cobrando mayor importancia gracias a que su tendencia es: disminuir los costos para transferir información; así como la creciente confiabilidad y rapidéz de acceso a la misma. Todo esto ha tenido como base el creciente desarrollo de la microelectrónica que nos ha entregado circuitos especializados de gran poder y confiabilidad, así como la utilización de técnicas más sofisticadas de programación.

Las redes de datos representan una importante solución a los requerimientos de cómputo y a las necesidades de intercambio de información de hoy día, ya que hacen accesibles, a un usuario dado, los recursos de los sistemas conectados a la red, dando además mayor disponibilidad y confiabilidad, ya que existen más de un camino para interconectar dos puntos dados de la red.

I.1. Conmutación por circuito vs. conmutación por paquete

Podemos distinguir básicamente dos mecanismos de conmutación en una red, éstos son la conmutación por circuito y la conmutación por paquetes. En la conmutación por circuito, existe un ancho de banda asignado durante todo el tiempo que dure la comunicación o llamada, este es el caso de la red telefónica. Se ve en este caso que si hay poca información que cursar, el medio de comunicación que ha establecido el circuito se ve utilizado pobremente, si existe algún otro requerimiento para transferencia de información podría no ser satisfecho aun cuando existiera el ancho de banda para hacerlo; la solución a esta grave deficiencia es el asignar el ancho de banda dinámicamente, dándolo sólo cuando se tiene información que cursar. La conmutación por paquetes requiere de mayor inteligencia para lograr su objetivo así como de la inclusión de información de control para poder distinguir la información que corresponde a cada circuito virtual, vemos pues,

que la conmutación por paquetes introduce el concepto de circuito virtual, y que recibe tal nombre por el hecho de que por un mismo canal de comunicación físico, se intercambia información de diferentes orígenes y destinos, es decir de diferentes llamadas.

I.2. Modelo OSI

Organismos internacionales como la Organización de Estándares Internacionales (ISO), al ver la urgencia de permitir la interconexión de equipos de diferentes fabricantes, creó en 1977 un subcomité (SC16) que normalizara la interconexión de sistemas heterogéneos, al cual llamó "Modelo de Referencia de Interconexión de Sistemas Abiertos" (OSI), el término "abiertos" se utilizó para realzar el hecho de que cualquier sistema que adoptara este estándar, quedaría abierto a todos los otros sistemas que obedecieran ese mismo estándar, pudiendo así interconectarse.

El modelo OSI consta de 7 niveles o capas, que son:

- (1) Nivel Físico
- (2) Nivel de Enlace
- (3) Nivel de Red o de Paquetes
- (4) Nivel de Transporte
- (5) Nivel de Sesión
- (6) Nivel de Presentación
- (7) Nivel de Aplicación

Como se ve en la figura I.1, todos los niveles están presentes en los computadores (host), mientras que en los nodos de la red sólo se tienen los niveles físico, de enlace y de red o paquetes, que conforman la subred de comunicación.

I.2.1. Nivel físico

El protocolo del nivel físico es el involucrado en la transmisión de los bits, en este nivel debe establecerse la manera de representar un 0 o un 1, el tiempo que ha de permanecer cada bit, si la transmisión es half-duplex o full-duplex, el número de terminales que ha de tener el conector de la red, así como la función de cada una de ellas.

I.2.1. Nivel de enlace

El nivel de enlace es el encargado de presentar a la capa inmediata superior, a partir de un canal físico no confiable, un canal libre de errores, la técnica para hacer esto es mediante la división de la información en tramas y la inclusión de una secuencia de verificación de trama, la cual permite detectar errores en la transmisión, se entenderá que el transmisor volverá a enviar la trama mientras ésta no le sea reconocida.

I.2.2. Nivel de red

Mientras que los dos niveles anteriores se verifican entre dos nodos cualesquiera de la red, el nivel de paquetes tiene lugar extremo a extremo de la red, y se encarga de los problemas de enrutamiento y congestión, así como de proveer de circuitos virtuales.

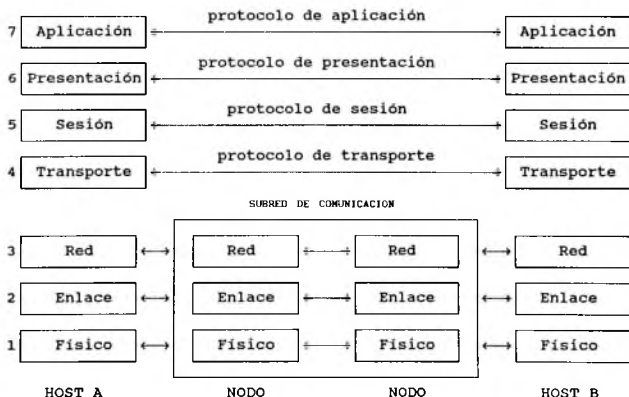


Figura I.1: Modelo de referencia para la interconexión de sistemas abiertos (OSI).

I.2.3. Nivel de transporte

La tarea del nivel de transporte es la de proveer una comunicación confiable host a host, así como la de ocultar todos los detalles de la subred de comunicación, de modo que sea posible cambiar el esquema de la red sin afectar el nivel de sesión.

I.2.4. Nivel de sesión

El nivel de sesión establece y mantiene conexiones, llamadas "sesiones", entre pares de procesos. Este nivel puede permitir la utilización de nombres simbólicos de procesos que se ejecutan en un host en alguna parte de la red.

I.2.5. Nivel de presentación

La principal función del nivel de presentación es la de realizar ciertas transformaciones comunes a los datos antes de ser enviados al nivel de sesión, transformaciones como el encriptado y compactamiento de texto.

I.2.6. Nivel de aplicación

Corresponde a la aplicación en particular de los usuarios.

I.3. El modelo OSI y la arquitectura del EDD

La figura I.2 hace una analogía de los niveles que se manejan en el EDD con la arquitectura OSI.

Los niveles correspondientes a la subred de comunicación están normalizados por el CCITT en su recomendación X.25; los niveles de transporte al de aplicación aún no han sido normalizados. De la figura I.2 se observa que en el EDD se tienen, además de los niveles correspondientes a la subred de comunicación, el nivel de transporte y el nivel de sesión.

El nivel de transporte corresponde a: i) la interfaz entre el nivel de red con el nivel de sesión o "shell", como se le refiere en la programación del EDD, que es donde se especifican los servicios para el establecimiento y liberación de las llamadas virtuales, para el intercambio de información, etc.; y ii) el

intercambio de información de control entre el EDD y un equipo remoto que maneje paquetes, siendo aquí donde entra la recomendación X.29.

El nivel de sesión es el que establece la comunicación entre los usuarios de las terminales y las facilidades del ensamblador/desensamblador de datos, y en nuestra aplicación corresponde, por el lado de los usuarios, a las recomendaciones X.3 y X.28 del CCITT.

EDD		MODELO OSI	
7			APLICACION
6			PRESENTACION
5	X.3	X.28	SESION
4	INTERFAZ / NIVEL RED	X.29	TRANSPORTE
3	PAQUETES		PAQUETES
2	ENLACE		ENLACE
1	FISICO		FISICO

Figura I.2: Analogía del EDD con el modelo OSI.

I.4. Normas del CCITT para un EDD

En el diseño del EDD se han tomado las normas que a este propósito ha establecido el CCITT, éstas son:

X.25, X.3, X.28 y X.29.

X.25

El protocolo X.25 define los 3 primeros niveles del modelo de referencia OSI, esto es, un nivel físico, un nivel de enlace y el nivel de red o de paquetes.

X.3

Define un conjunto de parámetros referidos como parámetros de terminal, los cuales indican la manera en que el EDD ha de tratar a la terminal asíncrona (p.ej. si habrá control de flujo del EDD sobre la terminal, si se hará eco a la terminal, etc.).

X.28

Define un lenguaje de comunicación entre los usuarios de la terminal y el EDD en base a señales de instrucción (comandos enviados por los usuarios al EDD) y señales de servicio (la contestación o indicaciones que hace el EDD a los usuarios) para acceder las facilidades de un EDD.

X.29

Define un lenguaje de comunicación entre un EDD y un ETD remoto que maneje paquetes u otro EDD, en base a mensajes de EDD, el lenguaje de comunicación utiliza el campo de datos de un paquete de datos cuando un circuito virtual existe entre ambos equipos, y permite intercambiar información de control entre los mismos.

II. RECOMENDACIONES X.3, X.28 Y X.29 DEL CCITT

II.1. Parámetros de terminal (X.3)

Los parámetros de terminal son de acuerdo a la revisión de 1984 realizada en Málaga-Torremolinos de la norma X.3 del CCITT, la cual define 22 parámetros. La función de cada uno de los parámetros se aplica independientemente a cada terminal, es decir, cada terminal tiene su propio conjunto de parámetros con sus respectivos valores; adicionalmente a los parámetros de terminal se han definido 6 parámetros más referidos como parámetros extra con funciones adicionales que serán tratados enseguida de los parámetros del CCITT, a continuación se hará la especificación de cada uno de ellos como están implantados en nuestro EDD.

II.1.1. Referencia 1

Rellamada al EDD utilizando un carácter.

Esta función permite al ETD arritmico iniciar un escape desde el estado transferencia de datos para enviar señales de instrucción al EDD.

0	Imposible
1	Carácter DLE
32 a 126	Posible, emplenado un carácter gráfico definido por el usuario

II.1.2. Referencia 2

Eco

Mediante esta función el EDD envía al ETD arritmico los caracteres recibidos del mismo.

0	Sin eco
1	Eco

NOTA: ver el parámetro extra 1

II.1.3. Referencia 3

Elección de carácter(es) de envío de datos (de expulsión)

Mediante esta función, el EDD es capaz de reconocer el (los) carácter(es) para el empaquetado de los datos y su transmisión de acuerdo a la norma X.25.

0	Sin carácter(es) de envío de datos
2	Carácter CR
6	Caracteres CR, ESC, BEL, ENQ, ACK
18	Caracteres CR, EOT, ETX
126	Todos los caracteres de las columnas 0 y 1 del alfabeto internacional número 5 y el carácter DEL

NOTA: las columnas 0 y 1 del A.I. No.5 comprenden los caracteres 0(00H) al 31(1FH) del ASCII.

II.1.4. Referencia 4

Elección de la duración de la temporización de reposo

Esta función permite elegir el tiempo que el EDD ha de esperar sin recibir un nuevo carácter antes de formar un paquete de datos y transmitirlo.

0	Sin temporización
1 a 255	Valor de la duración del temporizador de reposo en veinteavos de segundo

II.1.5. Referencia 5

Control de dispositivo auxiliar

Esta función permite al EDD realizar el control de flujo sobre el ETD arrítmico mediante la transmisión de caracteres especiales. Estos caracteres son los utilizados en el AI No.5 para conmutar a cerrado y abierto un dispositivo auxiliar.

0	Sin utilización de X-ON (DC1) y X-OFF (DC3)
1	Utilización de X-ON y X-OFF (transferencia de datos)
2	Utilización de X-ON y X-OFF (transferencia de datos e instrucción)

NOTA: ver el parámetro extra 8.

II.1.6. Referencia 6 Control de señales de servicio de EDD

Esta función permite elegir si se transmiten o no señales de servicio de EDD y en qué formato.

0	No se transmiten señales de servicio de EDD al ETD arritmico
1	Se transmiten señales de servicio de EDD en el formato normalizado
5	Se transmiten señales de servicio EDD y la señal de servicio EDD de sugerencia ("prompt") en el formato normalizado

II.1.7. Referencia 7 Elección de la operación que ha de efectuar el EDD al recibir la señal de corte

Esta función permite al EDD seleccionar la operación que ha de realizar al recibir la señal de corte (señal Break) del ETD arritmico.

0	Nada
1	Enviar paquete de interrupción
2	Enviar paquete de reiniciación
5	Enviar paquete de interrupción y mensaje de EDD de indicación de corte
8	Escape desde el estado transferencia de datos
21	Descartar salida, enviar paquete de interrupción (indicando que el parámetro 8 se ha puesto a 1) y mensaje de EDD de indicación de corte

II.1.8. Referencia 8 Descartar salida

Esta función permite al EDD, a petición, descartar el contenido de los paquetes de datos, esto es, sin que le sean transmitidos al ETD arritmico.

0	Entrega normal de datos
1	Descartar salida

II.1.9. Referencia 9

Relleno después del retroceso de carro

Esta función prevé la inserción de caracteres de relleno (carácter NULO) que se insertan automáticamente en la secuencia de caracteres enviados al ETD aritmético después de un carácter de retroceso de carro. Esto permite al mecanismo de impresión del ETD aritmético realizar correctamente la función de retroceso de carro.

0	Sin relleno después de CR
1 a 255	Número de caracteres de relleno insertados después del CR

II.1.10. Referencia 10

Nueva línea (o delimitación de línea)

Esta función prevé la inserción automática por el EDD de los determinantes de formato apropiados en el tren de caracteres transmitidos al ETD aritmético. Permite fijar el número de caracteres gráficos por línea.

0	Sin indicación de delimitación de línea
1 a 255	Número de caracteres gráficos por línea

II.1.11. Referencia 11

Velocidad binaria del ETD aritmético

0	110	bits/s
1	134.5	bits/s
2	300	bits/s
4	600	bits/s
5	75	bits/s
6	150	bits/s
7	1800	bits/s
8	200	bits/s
9	100	bits/s
10	50	bits/s
11	1200	bits/s
12	2400	bits/s
13	4800	bits/s
14	9600	bits/s
15	19200	bits/s

NOTA: Este parámetro puede modificarse

II.1.12. Referencia 12

Control de flujo del EDD por el ETD

Esta función permite el control de flujo entre el ETD arritmico y el EDD. El ETD arritmico indica si está o no preparado para recibir caracteres procedentes del EDD mediante la transmisión de caracteres especiales. Estos caracteres son los empleados en el AI No.5 para conmutar a cerrado o a abierto un dispositivo auxiliar.

- | | |
|---|---|
| 0 | Sin utilización de X-ON (DC1) y X-OFF (DC3) para control de flujo |
| 1 | Utilización de X-ON y X-OFF para control de flujo |

II.1.13. Referencia 13

Inserción de cambio de renglón después del retroceso de carro

Esta función prevé la inserción automática por el EDD de un carácter de cambio de renglón después de la transmisión de un carácter de retroceso de carro al ETD arritmico.

- | | |
|---|---|
| 0 | Sin inserción del cambio de renglón |
| 1 | Inserción de cambio de renglón después de CR en los datos remotos |
| 4 | Inserción de cambio de renglón después de la devolución en eco de CR hacia el ETD arritmico |
| 5 | Inserción de cambio de renglón después de la transmisión al ETD arritmico y después de la devolución en eco de CR |
| 6 | Inserción de cambio de renglón en la cadena de datos a transmitir después de CR desde el ETD arritmico y después de la devolución en eco de CR hacia el ETD arritmico |
| 7 | Inserción de cambio de renglón en la cadena de datos hacia y desde el ETD arritmico y después de la devolución en eco de CR hacia el ETD arritmico |

II.1.14. Referencia 14**Relleno después del cambio de renglón**

Esta función prevé la inserción de caracteres de relleno (carácter NULO) que se insertan automáticamente en la secuencia de caracteres enviados al ETD arritmico después de un carácter de cambio de renglón. Esto permite al mecanismo de impresión del ETD arritmico realizar correctamente la función de cambio de renglón.

0	Sin relleno después de CR
1 a 255	Número de caracteres de relleno insertados después del CR

II.1.15. Referencia 15**Edición**

Esta función dota al EDD con la capacidad de edición durante el estado de transferencia de datos para los ETD arritmicos. Durante el estado de instrucción (modo comando), la función de edición siempre está disponible.

0	No se emplea edición en el estado transferencia de datos
1	Se emplea edición en el estado transferencia de datos

NOTA: Las facilidades de edición son:

- Supresión de carácter
- Supresión de línea
- Visualización de línea

Cuando esta función no se aplica, la edición en el estado instrucción se realiza con los caracteres:

- DEL para la supresión de carácter
- CAN para la supresión de línea
- DC2 para la visualización de línea

La puesta a 1 de este parámetro anula la función del parámetro 4

II.1.16. Referencia 16**Carácter de edición de supresión de carácter**

0 a 126	Un carácter del AI No.5
127	Carácter DEL

II.1.17. Referencia 17**Carácter de edición de supresión de línea**

0 a 23	Un carácter del AI No.5
24	Carácter CAN
25 a 127	Un carácter del AI No.5

II.1.18. Referencia 18**Carácter de edición de visualización de línea**

0 a 17	Un carácter del AI No.5
18	Carácter DC2
19 a 127	Un carácter del AI No.5

II.1.19. Referencia 19**Señales de servicio EDD de edición**

Esta función permite seleccionar si se transmiten o no señales de servicio de edición y en que formato hacia el ETD arrítmico.

0	Sin señales de servicio EDD de edición
1	Señales de servicio EDD de edición para terminales impresoras
2	Señales de servicio EDD de edición para terminales de visualización

NOTA: El tipo de señal de servicio especifica se define en la parte de "lenguaje de comunicación EDD-usuarios (X.28)".

II.1.20. Referencia 20**Máscara de eco**

Cuando el eco está habilitado (parámetro 2), esta función permite seleccionar el(los) carácter(es) recibidos del ETD arrítmico a los cuales no se les realiza el eco.

0	Sin máscara de eco (todos los caracteres se devuelven en eco)
1	No se devuelve en eco CR
2	No se devuelve en eco LF
4	No se devuelven en eco VT, HT, FF
8	No se devuelven en eco BEL, BS

16	No se devuelven en eco ESC, ENQ
32	No se devuelven en eco ACK, NAK, STX, SOH, EOT, ETB, ETX
64	No se devuelven en eco los caracteres de edición designados por los parámetros 16, 17 y 18
128	No se devuelven en eco ninguno de los caracteres de las columnas 0 y 1 del AI No.5 ni el carácter DEL

NOTA: Los caracteres de la columna 0 y 1 del AI No.5 son los comprendidos entre 0(00H) y 31(7FH) del ASCII.

II.1.21. Referencia 21 Tratamiento de la paridad

Esta función permite al EDD realizar la verificación de la paridad en los caracteres recibidos del ETD arritmico y/o generar la paridad en la cadena de caracteres transmitidos hacia el ETD arritmico.

0	Sin verificación ni generación de la paridad
1	Verificación de la paridad
2	Generación de la paridad
3	Verificación y generación de la paridad

NOTA: para el tipo de paridad ver el parámetro extra 2.

II.1.22. Referencia 22 Espera de página

Esta función permite al EDD suspender la transmisión de caracteres hacia el ETD arritmico después de que le ha transmitido "n" caracteres de cambio de renglón.

0	No es posible la espera de página
10 a 255	Número de caracteres de cambio de renglón considerados por el EDD para la función de espera de página

II.2. Parámetros de terminal extras

Con el objeto de proveer mayor versatilidad al EDD, se han implantado 6 parámetros extra referidos como X-n, donde n es el número del parámetro extra. Al igual que los parámetros definidos por el X.3, los parámetros extra se aplican independientemente a cada terminal.

II.2.1. Referencia X-1

Elección de mayúsculas y minúsculas

Esta función permite seleccionar si los caracteres transmitidos al ETD arritmico son: todos mayúsculas; todos minúsculas; o bien una combinación de ambos.

0	Los caracteres transmitidos al ETD arritmico es una combinación de mayúsculas y minúsculas
1	Todos los caracteres transmitidos al ETD arritmico son minúsculas
2	Todos los caracteres transmitidos al ETD arritmico son mayúsculas

II.2.2. Referencia X-2

Tipo de paridad

Esta función permite seleccionar, en conjunción con el parámetro 21, el tipo de paridad que ha de verificarse de los caracteres provenientes del ETD arritmico y/o que ha de generarse en los caracteres transmitidos hacia el ETD arritmico.

0	Verifica y/o genera paridad par
1	Verifica y/o genera paridad impar
2	Verifica y/o genera paridad 0
3	Verifica y/o genera paridad 1

NOTA: Este parámetro no tiene efecto si el parámetro 21 está puesto al valor 0.

II.2.3. Referencia X-3 Carga del perfil inicial

Esta función permite al EDD seleccionar el tipo de perfil que se ha de cargar al encender al EDD.

- | | |
|---|---|
| 0 | Carga los parámetros de terminal del perfil almacenado en memoria no volátil |
| 1 | Carga los parámetros de terminal con uno de los perfiles predefinidos y especificado por el parámetro X-4 |

II.2.4. Referencia X-4 Perfil inicial

Esta función indica al EDD el número de perfil predefinido con que han de cargarse los parámetros de terminal al encender el EDD.

- | | |
|--------|--|
| 0 a 20 | Indica el perfil predefinido inicial a cargar en los parámetros de terminal al encender el EDD, no se aplica cuando el parámetro extra X-3 esté puesto al valor cero |
|--------|--|

II.2.5. Referencia X-5 Idioma

Esta función habilita al EDD para enviar al usuario del ETD arritmico las señales de servicio correspondientes, en un idioma seleccionable dentro de la gama disponible.

- | | |
|---|--|
| 0 | Las señales de servicio serán enviadas hacia el ETD arritmico en el idioma ESPAÑOL |
| 1 | Las señales de servicio serán enviadas hacia el ETD arritmico en el idioma INGLES |

II.2.6. Referencia X-6

Será objeto de ulterior estudio la función asociada a este parámetro (se prevé para este parámetro la elección de una dirección por defecto cuando ésta no se especifique durante una petición de llamada).

II.2.7. Referencia X-7

Será objeto de ulterior estudio la función asociada a este parámetro (se prevé para este parámetro la elección de la detección automática de velocidad y tipo de paridad del ETD arritmico).

II.2.8. Referencia X-8
Uso de la señal RTS

Esta función permite al EDD seleccionar, para la señal de control de modem "petición de envío" (RTS), la función que ha de realizar sobre el ETD arritmico.

0	Siempre en nivel activo
1	Como solicitud de envío de datos
2	Para el control de flujo sobre el ETD arritmico

NOTA: Cuando este parámetro tiene el valor 2, solo tendrá efecto si el parámetro 5 es diferente de cero.

II.2.9. Referencia X-9

Será objeto de ulterior estudio la función asociada a este parámetro.

II.2.10. Referencia X-10

Será objeto de ulterior estudio la función asociada a este parámetro.

II.3. Lenguaje de comunicación EDD-Usuarios (X.28)

En esta parte se definen los comandos (o señales de instrucción) que un usuario puede enviar al EDD desde el ETD arritmico cuando se encuentre en el modo comando (o estado instrucción EDD), así como las respuestas (o señales de servicio de EDD) que ha de enviar el EDD hacia el ETD arritmico; el EDD acepta indistintamente caracteres en minúsculas y/o mayúsculas.

Las señales de instrucción (o comandos) proveerán funciones para:

- Selección (petición de llamada)
- Selección del perfil
- Asignación
- Asignación y lectura
- Lectura
- Petición de liberación
- Estado
- Reiniciación
- Interrupción

II.3.1. Formato del determinante de formato

Se enviarán los caracteres (CR) (LF).

II.3.2. Formato del delimitador de la señal de instrucción

Se podrá enviar al EDD como delimitador de instrucción (o terminador) el carácter (CR) o el carácter (+).

II.3.4. Formato normalizado de la señal de servicio de EDD de acuse de recibo

Se enviará como acuse de recibo de las señales de instrucción el determinante de formato.

II.3.5. Señal de servicio de EDD dispuesto (prompt)

Si el parámetro 6 está puesto al valor 5, el EDD indicará al ETD arritmico su disposición de aceptar un nuevo comando (o señal de instrucción) mediante el envío del carácter (*) después de un determinante de formato, en caso contrario no se enviará la señal de EDD dispuesto al ETD arritmico.

II.3.6. Formato de la señal de instrucción de EDD de lectura de los parámetros de terminal del CCITT
< PAR? >

Se transmitirán los caracteres (P)(A)(R)(?) o (p)(a)(r)(?) seguidos de la referencia decimal del parámetro que ha de leerse.

Se transmitirán los caracteres del AI No.5 para representar la referencia de parámetro; por ejemplo: el valor 17 se transmitirá como la secuencia de caracteres (1)(7).

Si no se indica número de referencia de parámetro en la señal de instrucción de EDD de lectura, se asumirá una referencia a todos los parámetros.

Cuando se pida la lectura de más de un parámetro mediante la señal de EDD de lectura, deberá transmitirse el carácter (,) o () entre las referencias decimales de los parámetros.

La señal de respuesta que el EDD envíe al ETD arrítmico será de acuerdo al formato de la señal de servicio de EDD de valor de parámetro.

Ejemplos de señales de instrucción de EDD de lectura válidas son:

```
PAR? 7
par?
par? 1 , 3,4 5
paR? 4 8
```

II.3.7. Formato de la señal de instrucción de EDD de selección de perfil
< PROF >

Deberán enviarse los caracteres (P)(R)(O)(F) o (p)(r)(o)(f) seguidos de el número decimal de referencia del perfil predefinido seleccionado, que comprenden los valores 0 a 20 (las secuencias (0) a (2)(0)).

Ejemplos de señales de instrucción válidas son:

```
PROF 5
prof 20
Prof 14
```

II.3.8. Formato de las señales de instrucción de EDD de asignación de valores y de asignación de valores y lectura de los parámetros de terminal del CCITT
< SET , SET? >

La señal de instrucción de EDD de asignación de valores, estará compuesta por la secuencia de caracteres (S)(E)(T) o (s)(e)(t), seguidos de la referencia decimal del parámetro cuyo valor se desea modificar, seguida del carácter (:) y del valor que se desea asignar al parámetro.

La señal de instrucción de EDD de asignación de valores y lectura, estará compuesta por la secuencia de caracteres (S)(E)(T)(?) o (s)(e)(t)(?), seguidos de la referencia decimal del parámetro cuyo valor se desea modificar, seguida del carácter (:) y del valor que se desea asignar al parámetro.

Si hay más de un parámetro al cual asignar valor o asignar valor y leer, deberá transmitirse el carácter (,) o () entre cada valor de parámetro y la referencia del siguiente parámetro.

Si se envía ya sea la instrucción de EDD de asignación o la instrucción de EDD de asignación y lectura sin especificar referencia alguna a parámetro, la señal de instrucción se interpretará como una señal de instrucción de EDD de lectura con referencia a todos los parámetros.

La señal de respuesta que el EDD envíe al ETD arritmico será de acuerdo al formato de la señal de servicio de EDD de valor de parámetro.

Ejemplos válidos de estas señales de instrucción son:

```
SET 2:0,4:40
set
Set 1:1 7:8 , 10:80
set?
SET? 11:13, 15:1
Set? 5:1, 12:1 , 21:3
```

II.3.9. Formato de la señal de instrucción de EDD de petición de liberación
< CLR >

Deberán transmitirse los caracteres (C)(L)(R) o (c)(l)(r).

Ejemplos válidos de esta señal de instrucción son:

CLR
Clr
clr

II.3.10. Formato de la señal de servicio de confirmación e indicación de liberación

Se transmitirán los caracteres (c)(l)(r)() (d)(t)(e) seguidos del determinante de formato.

II.3.11. Formato de la señal de instrucción de EDD de estado < STAT >

Se transmitirán los caracteres (S)(T)(A)(T) o (s)(t)(a)(t).

II.3.12. Formato de la señal de servicio de EDD de estado ocupado y de estado libre

Para el estado ocupado se transmitirán los siguientes caracteres según el valor del parámetro extra 5.

X-par 5 = 0 (o)(c)(u)(p)(a)(d)(o)
X-par 5 = 1 (e)(n)(g)(a)(g)(e)(d)

Para el estado libre se transmitirán los siguientes caracteres según el valor del parámetro extra 5.

X-par 5 = 0 (l)(i)(b)(r)(e)
X-par 5 = 1 (f)(r)(e)(e)

II.3.13. Formato de la señal de instrucción de EDD de reiniciación < RESET >

Se transmitirán los caracteres (R)(E)(S)(E)(T) o (r)(e)(s)(e)(t).

Ejemplos válidos de esta señal de instrucción son:

RESET
Reset
reset

II.3.14. Formato de la señal de instrucción de EDD de interrupción
< INT >

Se transmitirán los caracteres (I)(N)(T) o (i)(n)(t).

II.3.15. Formato de la señal de servicio de EDD de valor de parámetro

Se transmitirán los caracteres (p)(a)(r)(), seguidos de la referencia decimal, seguidos del carácter (:) y del valor del parámetro apropiado. Si la referencia o valor de parámetro pedidos no son válidos, se transmitirán los caracteres (i)(n)(v) en lugar del valor del parámetro apropiado.

Si la señal de servicio de EDD de valor de parámetro contiene más de un valor de parámetro, se transmitirán los caracteres (,)() entre cada valor de parámetro y la siguiente referencia de parámetro.

Ejemplo de esta señal de servicio es:

par 1:1, 4:0, 15:1, 25:inv, 22:0

II.3.16. Formato de la señal de instrucción de EDD de selección

La señal de instrucción de EDD de selección está formada en el orden siguiente: por un bloque de petición de facilidad; o un bloque de dirección, o ambos; seguidos facultativamente de un campo de datos de usuario.

II.3.16.1. Formato del bloque de petición de facilidad

Los caracteres que representan el código de petición de facilidad se enviarán como se indica a continuación:

(G) o (g)	cadena de GCU	- Grupo cerrado de usuarios
(R) o (r)		- Cobro revertido
(T) o (t)	cadena de EPER	- Selección de red de tránsito EPER
(C) o (c)		- Información de tasación
(N) o (n)		- Identificación de usuario de red

NOTA: El campo de facilidades no deberá exceder de 63 caracteres.

Cuando deba enviarse más de un código de petición de facilidad, se enviará el carácter (,) para separar los códigos de petición de facilidad. El carácter (-) se enviará al final del bloque de petición de facilidad para separarle del campo de dirección.

II.3.16.1.1. Formato de la señal de petición de facilidad IUR

El formato es el siguiente:

```
<petición de facilidad IUR>::= <N><cadena IUR>
<N>::= carácter (N) o (n) del AI No.5
<cadena IUR>::= Uno o más caracteres alfabéticos o numéricos.
```

NOTA: A estos caracteres no se les hace eco sin importar el valor del parámetro 2. El número de caracteres está limitado por la red.

II.3.16.1.2. Formato de la señal de petición de facilidad de selección (red de tránsito) de EPER

```
<petición de facilidad de EPER>::= <T><cadena EPER>
<T>::= carácter (T) o (t) del AI No.5
<cadena EPER>::= El CIRD de la EPER solicitada que consta de
4 cifras decimales (podrán especificarse más
de un CIRD)
```

II.3.16.1.3. Formato de la señal de petición de facilidad de cobro revertido

```
<petición de facilidad de cobro revertido>::= <R>
<R>::= carácter (R) o (r) del AI No.5
```

II.3.16.1.4. Formato de la señal de petición de facilidad de grupo cerrado de usuarios

```
<petición de facilidad de GCU>::= <G><índice>
<G>::= carácter (G) o (g) del AI No.5
<índice>::= 2 o 4 cifras decimales
```

II.3.16.1.5. Petición de facilidad de información de tasación

<petición de facilidad de información de tasación> ::= <C>
 <C> ::= el carácter (C) o (c) del AI No.5

II.3.16.2. Formato del bloque de dirección

<bloque de dirección> ::= <A><dirección>
 <A> ::= carácter (A) o (a) del AI No.5
 <dirección> ::= una secuencia de dígitos decimales

NOTA: La longitud máxima del campo de dirección en los paquetes que lo contienen depende de la red, y estará limitado en el EDD a 16 dígitos decimales.

II.3.16.3. Formato del campo de datos de llamada de usuario

El campo de datos de usuario tiene dos formatos, uno al cual se le hace el eco y otro al que no:

<datos de usuario con eco> ::= <*><D><datos>
 <datos de usuario sin eco> ::= <*><P><datos>
 <*> ::= carácter (*) del AI No.5
 <D> ::= carácter (D) o (d) del AI No.5
 <P> ::= carácter (P) o (p) del AI No.5
 <datos> ::= una secuencia de caracteres alfabéticos de hasta 14 caracteres

II.3.17. Formato de la señal de petición de llamada

Antes de que el EDD pueda aceptar una llamada entrante para una terminal, y antes de que esa terminal pueda tener cualquier interacción con el EDD, el ETD arrítmico deberá, después de poner en nivel activo su señal DTE, enviar hacia el EDD dos caracteres de retroceso de carro (CR), a lo cual el EDD responderá con la señal de presentación (siempre y cuando el parámetro 6 sea diferente de 0).

II.3.18. Formato de la señal de servicio de EDD de error

Se transmitirán los caracteres (e)(r)(r) seguidos del determinante de formato.

II.3.19. Formato de la señal de servicio de EDD de conectado, y de llamada entrante.

Se transmitirán los caracteres (c)(o)(m) seguidos del determinante de formato.

II.3.20. Formato de la señal de servicio de EDD de supresión de carácter

Según el valor del parámetro 19, se aplicará lo siguiente:

valor 0: No se enviará ninguna señal de servicio.

valor 1: Se enviará el carácter (/).

valor 2: Se enviarán los caracteres (BS)(SP)(BS).

II.3.21. Formato de la señal de servicio de EDD de supresión de línea

Según el valor del parámetro 19, se aplicará lo siguiente:

valor 0: No se enviará ninguna señal de servicio.

valor 1: Se enviarán los caracteres (x)(x)(x) seguidos del determinante de formato.

valor 2: Se enviarán los caracteres (BS)(SP)(BS) un número de veces igual al número de caracteres gráficos a borrar.

II.3.22. Formato de la señal de servicio de EDD de espera de página

Se transmitirán los caracteres (p)(a)(g)(e)(.)(.)(.).

II.3.22. Formato de la señal de instrucción de anulación de espera de página

Se transmitirá el carácter (CR) ó (X-ON).

II.4. Señales de instrucción y de servicio adicionales al X.28

Las señales de instrucción adicionales proveerán funciones para:

Lectura de los parámetros extra
 Asignación a los parámetros extra
 Asignación y lectura de los parámetros extra
 Versión
 Salvar/Restaurar parámetros
 Ayuda

II.4.1. Formato de la señal de instrucción de EDD de lectura de los parámetros extra < XPAR? >

Se transmitirán los caracteres (X)(P)(A)(R)(?) o (x)(p)(a)(r)(?) seguidos de la referencia decimal del parámetro extra que ha de leerse.

Si no se indica número de referencia de parámetro en la señal de instrucción de EDD de lectura, se asumirá una referencia a todos los parámetros extra.

Cuando se pida la lectura de más de un parámetro extra mediante la señal de EDD de lectura de los parámetros extra, deberá transmitirse el carácter (,) o () entre las referencias decimales de los parámetros extra.

La señal de respuesta que el EDD envíe al ETD arritmico será de acuerdo al formato de la señal de servicio de EDD de valor de parámetro extra.

Ejemplos de señales de instrucción de EDD de lectura de los parámetros extra válidas son:

```
XPAR? 1
xpar? 1 , 3,4 5
XpaR? 4 2
```

II.4.2. Formato de las señales de instrucción de EDD de asignación de valores y de asignación de valores y lectura de los parámetros extra < XSET , XSET? >

La señal de instrucción de EDD de asignación de valores a los parámetros extra, estará compuesta por la secuencia de caracteres (X)(S)(E)(T) o (x)(s)(e)(t), seguidos de la referencia decimal del parámetro cuyo valor se desea modificar, seguida del carácter (:) y del valor que se desea asignar al parámetro.

La señal de instrucción de EDD de asignación de valores y lectura de los parámetros extra, estará compuesta por la secuencia de caracteres (X)(S)(E)(T)(?) o (x)(s)(e)(t)(?), seguidos de la referencia decimal del parámetro cuyo valor se desea modificar, seguida del carácter (:) y del valor que se desea asignar al parámetro.

Si hay más de un parámetro al cual asignar valor o asignar valor y leer, deberá transmitirse el carácter (,) o () entre cada valor de parámetro y la referencia del siguiente parámetro.

Si se envía ya sea la instrucción de EDD de asignación, o la instrucción de EDD de asignación y lectura a los parámetros extra, sin especificar referencia alguna a parámetro, la señal de instrucción se interpretará como una señal de instrucción de EDD de lectura a los parámetros extra con referencia a todos los parámetros.

La señal de respuesta que el EDD envíe al ETD arritmico será de acuerdo al formato de la señal de servicio de EDD de valor de parámetro extra.

Ejemplos válidos de estas señales de instrucción son:

```
XSET 2:0,3:1,4:20
xset
XSet 1:0 8:0 , 2:0
xset?
XSET? 1:2, 3:0
XSet? 8:1, 5:1 , 3:0
```

II.4.3. Formato de la señal de instrucción de versión < VER >

Se transmitirán hacia el EDD los caracteres (V)(E)(R) o (v)(e)(r).

II.4.4. Formato de la señal de servicio de la señal de instrucción ver

Se enviará la señal de servicio de presentación.

II.4.5. Formato de la señal de instrucción de salvar
< SAVE >

Se transmitirán los caracteres (S)(A)(V)(E) o (s)(a)(v)(e).

II.4.6. Formato de la señal de instrucción de restaurar
< RESTORE >

Se transmitirán los caracteres (R)(E)(S)(T)(O)(R)(E) o (r)(e)(s)(t)(o)(r)(e).

II.4.7. Formato de la señal de instrucción de ayuda
< HELP o ? >

Se transmitirán hacia el EDD los caracteres (H)(E)(L)(P) o (h)(e)(l)(p) o (?).

Si se desea información adicional de alguno de los comandos, la secuencia de caracteres anteriores será seguida de el carácter () y de la secuencia de caracteres que corresponde a la referencia de la señal de instrucción en cuestión.

Si se desea información adicional de más de una señal de instrucción, cada referencia a ellas deberá separarse por el carácter (,) o ().

Ejemplos de señales de ayuda válidas son:

```
HELP
Help
help Stat, par?
? set stat , prof
?
```

II.5. Comandos remotos (X.29)

El intercambio de información de control entre el EDD y un ETD de paquetes remoto u otro EDD se realiza mediante la utilización del campo de datos de paquetes de datos definidos en la recomendación X.25.

El intercambio de la información de control se realiza mediante mensajes de EDD.

Los paquetes de datos que llevan mensajes de EDD, deberán tener el bit calificador (bit Q) puesto a 1; el EDD sólo tomará en cuenta aquellos mensajes de EDD contenidos en paquetes que hayan sido completamente recibidos. Cuando se tengan múltiples referencias a un parámetro en un mensaje de EDD, sólo será tomada en cuenta la última.

Los mensajes de EDD definidos por la recomendación X.29 son los siguientes:

- Mensaje de EDD de asignación de valores
- Mensaje de EDD de lectura
- Mensaje de EDD de asignación de valores y lectura
- Mensaje de EDD de indicación de parámetros
- Mensaje de EDD de invitación a liberar (ver nota)
- Mensaje de EDD de indicación de corte
- Mensaje de EDD de error
- Mensaje de EDD de reelección (ver nota)

NOTA: Este mensaje es ignorado por el EDD

Las respuestas del EDD a los mensajes de asignación de valores, de lectura o de asignación de valores y lectura se muestran en el cuadro II.1.

II.5.1. Mensaje de EDD de asignación de valores

Permite modificar los parámetros de terminal del equipo remoto, si no se indica parámetro alguno, los parámetros son iniciados a un valor por omisión que es aquel con que se inician al encender el equipo.

II.5.2. Mensaje de EDD de lectura

Permite obtener la lectura de los parámetros de terminal indicados en el mensaje de EDD, la no especificación de parámetros causa una lectura a todos los parámetros.

II.5.3. Mensaje de EDD de asignación de valores y lectura

Es exactamente el equivalente a enviar un mensaje de asignación de valores y un mensaje de EDD de lectura.

II.5.4. Mensaje de EDD de indicación de parámetros

Mediante este mensaje el EDD indica al solicitante, el valor actual de los parámetros a los que se hace referencia, teniendo además un mecanismo para indicar errores en la asignación de parámetros (estos errores serán detectados en las asignaciones de un mensaje de asignación de valores o de asignación de valores y lectura previamente recibido).

II.5.5. Mensaje de EDD de invitación a liberar

Este mensaje lo utiliza el DTE de paquetes remoto, para solicitar al EDD que libere la llamada virtual.

II.5.6. Mensaje de EDD de indicación de corte

Si el valor del parámetro 7 está puesto a 21, el EDD transmitirá un paquete de interrupción, seguido de un mensaje de EDD de indicación de corte para indicar que el EDD, a petición del ETD arritmico, está descartando la información recibida (es decir envió la señal de corte). El mensaje de EDD de indicación de corte contendrá la advertencia de que el parámetro 8 se ha puesto a 1 (descartar salida).

La respuesta al mensaje de EDD de indicación de corte será un mensaje de EDD de asignación de valores indicando que el parámetro 8 debe ponerse a 0 (entrega normal de datos).

Si es el EDD quien recibe el mensaje de indicación de corte, responderá de la manera ya indicada y transmitirá la señal de corte al ETD arritmico.

Mensajes de EDD recibidos por el EDD		Operaciones efectuadas con los parámetros de EDD	Mensaje de EDD de indicación de parámetro transmitido al ETD de paquetes
Tipo	Campo de parámetro		
Asignación de valores	Ninguno	Reiniciense todos los parámetros de X.3 a sus valores iniciales	Ninguno
	Lista de los parámetros con los valores deseados	Asignese a los parámetros elegidos los valores dados: a) si no hay errores b) si hay errores	a) Ninguno b) Lista de los parámetros no válidos
Asignación de valores y lectura	Ninguno	Reiniciense todos los parámetros de X.3 a su valor inicial	Lista de todos los parámetros de X.3 con sus valores iniciales
	Lista de los parámetros elegidos con los valores deseados	Asignese a los parámetros elegidos los valores dados	Lista de estos parámetros, con sus nuevos valores vigentes
Lectura	Ninguno	Ninguna	Lista de todos los parámetros de X.3 con sus valores vigentes
	Lista de parámetros elegidos	Ninguna	Lista de estos parámetros de X.3

Cuadro II.1: Mensajes de EDD transmitidos en respuesta mensajes de asignación de valores, lectura, o asignación de valores y lectura.

II.5.7. Mensaje de EDD de error

El EDD transmitirá un mensaje de EDD de error en los siguientes casos:

- a) el EDD recibe un código de mensaje irreconocible;
- b) el campo de parámetros que sigue a un código de mensaje válido es incorrecto o incompatible con el código del mensaje;
- c) el formato del campo de parámetros que sigue a un código de mensaje correcto es inválido;
- d) el EDD recibe un mensaje de EDD de indicación de parámetro no solicitado;
- e) el EDD recibe un mensaje de EDD demasiado largo.

Hay un tratamiento adicional del manejo de los errores en los mensajes de EDD de asignación de valores, de lectura, o de asignación de valores y lectura; los errores consisten en accesos no válidos a un parámetro de EDD; el campo de parámetros del mensaje de EDD de indicación de parámetro transmitido en respuesta a tales mensajes de EDD contendrá la indicación de este hecho.

Las posibles razones de acceso no válido a un parámetro de EDD son:

- a) la referencia de parámetro no se aplica al EDD;
- b) el valor de parámetro no se ha aplicado en el EDD o no puede modificarse del valor vigente;
- c) el parámetro es de lectura solamente (sólo en los mensajes de EDD de asignación y de asignación y lectura).

II.5.8. Mensaje de EDD de reselección

El mensaje de EDD de reselección lo utiliza un ETD de paquetes para pedir al EDD que libere la llamada virtual, después de transmitir al ETD arrítmico todos los datos remotos recibidos previamente, para establecer la comunicación con el ETD reseleccionado.

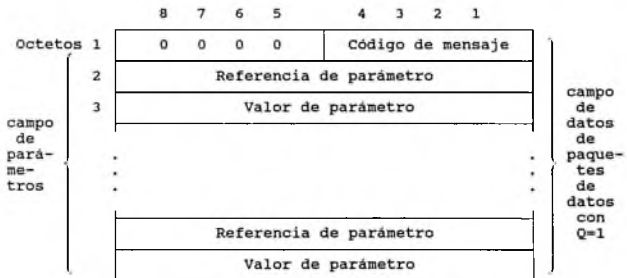
II.5.9. Formato general de los mensajes de EDD

Todos los mensajes de EDD están constituidos por un campo de identificador de control (bits 8, 7, 6 y 5 del octeto 1 iguales a 0000) y un campo de código de mensaje (bits 4, 3, 2 y 1 del octeto 1).

Los mensajes de EDD de asignación de valores, de lectura, de asignación de valores y lectura, y de indicación de parámetro consisten en el octeto 1 que puede ir seguido de uno o más campos de parámetro. Cada campo de parámetro está constituido por un octeto de referencia de parámetro y un octeto de valor de parámetro.

La longitud máxima de un mensaje de EDD depende de la red, pero tendrá como mínimo 128 octetos.

La figura II.2 muestra el formato general de los mensajes de EDD.



Código de mensaje: 0010 - Asignación de valores
 0100 - Lectura
 0110 - Asignación de valores y lectura
 0000 - Indicación de parámetro

Figura II.2: Formato general de los mensajes de EDD.

III. NIVEL DE TRANSPORTE (INTERFAZ CON EL NIVEL DE RED)

Los niveles de sesión o de transporte se comunican con el nivel de paquetes a través del envío de comandos hacia el canal, y del envío de indicaciones del canal hacia el nivel de sesión.

El envío de comandos desde el shell (nivel de sesión) hacia el nivel de paquetes, se realiza a través de un conjunto de operaciones o funciones provistas por el nivel de transporte.

La figura III.1 muestra el esquema de este intercambio de comandos e indicaciones.

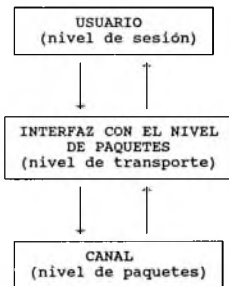


Figura III.1: Envío de comandos del usuario hacia el canal y de indicaciones del canal hacia el usuario.

Los servicios provistos incluyen lo siguiente:

- * - Solicitud de canal lógico
- * - Solicitud de paquete de negociación
- * - Selección (petición de llamada)
- * - Envío de datos
- * - Envío de un paquete de interrupción
- * - Envío de paquete de RNR

- * - Envío de paquete de RR
- * - Solicitud de reinicio del circuito virtual
- * - Petición de liberación de la llamada
- * Indicación de arribo de paquete de datos
- * Indicación de circuito establecido
- * Indicación de reiniciación
- * Indicación de rearranque
- * Indicación de enlace interrumpido
- * Indicación de circuito virtual liberado
- * Indicación de arribo de paquete de interrupción

Como se puede observar, estas funciones dan a un nivel superior todos los elementos para involucrarse en una llamada virtual y su terminación.

Antes de pasar a explicar estos servicios, daremos una breve descripción de las estructuras de datos involucradas y su interrelación, de tal manera que tenga sentido la utilización de tales estructuras en la invocación de los servicios del nivel de transporte; la figura III.2 muestra esta interrelación.

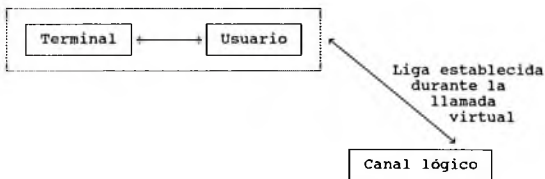


Figura III.2: Interrelación entre las estructuras de datos de un canal lógico y un usuario.

Se puede ver que las estructuras "Terminal" y "Usuario" están íntimamente ligadas, y que sólo durante una llamada virtual se establecen ligas entre la estructura que define a un canal lógico y un usuario-terminal.

Recuérdese que las señales de servicio son las respuestas que el EDD da a señales de instrucción dadas por los usuarios.

III.1. Comandos del nivel de transporte

III.1.1. Solicitud de un canal lógico

Esta función permite solicitarle al nivel de paquetes, un canal lógico libre y preparado para establecer llamadas virtuales (esta función no es accesible directamente al nivel de sesión sino a través de la selección).

Nombre:

EnlazaCanal

Sintaxis:

canal\$logico=EnlazaCanal(Ter.num\$usuario);

canal\$logico: es un entero que indica el número de canal lógico asignado.

Ter.num\$usuario: es el campo que identifica al número de usuario en cuestión, de la estructura terminal.

Descripción:

En la invocación de esta función se indica el número de usuario involucrado, los usuarios se numeran de 0 a 7; devuelve un entero entre 1 y 8 que indica el número de canal lógico asignado; si no es posible satisfacer este requerimiento, se devuelve cero.

III.1.2. Solicitud de paquete de negociación

Permite obtener un área de memoria con la estructura de un paquete de negociación (esta función no es accesible directamente al nivel de sesión).

Nombre:

RecibeVacioBufPaq

Sintaxis:

```
paq$ptr=RecibeVacioBufPaq(usuario.tx$info);
```

paq\$ptr: es un apuntador a una estructura paquete de negociación.

usuario.tx\$info: es un apuntador a buffer de paquetes (paquetes disponibles).

Descripción:

Esta función devuelve un apuntador a estructura tipo paquete de negociación, si temporalmente no es posible satisfacer la petición, el proceso invocador es bloqueado en espera de la disponibilidad de un paquete vacío. Se distinguen dos tipos de paquete, el paquete de negociación y el paquete de X.25, el primero es una estructura de datos general que permite contener cualquier paquete de X.25, mientras que el último tiene sólo los campos requeridos y de la longitud apropiada para cada caso.

III.1.3. Selección (petición de llamada)

Esta función permite hacer una selección (hacer una petición de llamada) con el formato que para este efecto define la recomendación X.28.

Nombre:

```
op$peticion$llamada
```

Sintaxis:

```
paq$ptr=op$peticion$llamada(reg$ptr,token$ptr,paq$ptr);
```

reg\$ptr: apuntador a una estructura tipo terminal

token\$ptr: es el apuntador a una lista de tokens que contienen los elementos de una petición de llamada como lo especifica la recomendación X.28

paq\$ptr: es una estructura tipo "paquete de negociación de petición de llamada" que se va llenando con la información tomada de la cadena de tokens.

Descripción:

Cada vez que se solicita una petición de llamada mediante esta función, se verifica que el usuario no esté ya involucrado en una llamada virtual, de ser así, se envía al usuario la señal de servicio de ocupado. Para proceder a la satisfacción de esta petición, se requiere asignar un canal lógico, si esta asignación no tiene éxito, se envía al usuario la señal de servicio de no más canales; se requiere además solicitar un paquete de negociación, en este caso, si no es posible obtener este paquete, se envía al usuario la señal de servicio de buffer lleno. Una vez que se ha asignado un canal lógico y un paquete de negociación, este último es llenado con los elementos de la petición de llamada (los tokens), y transmitido hacia el proceso que hace el manejo del canal lógico a través de la estructura canal, éste a su vez se encarga de formar, a partir del paquete de negociación, un paquete de petición de llamada, así como de realizar las operaciones necesarias para que sea transmitido por el canal de X.25.

III.1.4. Solicitud de envío de datos

Esta función permite, al nivel de sesión, el envío de datos a través del canal virtual.

Nombre:

op\$envia\$datos

Sintaxis:

```
call op$envia$datos(reg$ptr,cmd_c);
```

reg\$ptr: apuntador a una estructura tipo terminal
 cmd_c: contiene el carácter a transmitir junto con
 un comando

Descripción:

Cada vez que se invoca esta función, el carácter es insertado en un buffer de entrada de la estructura tipo terminal, y cuando ocurre una condición de expulsión (de envío de datos), se solicita y se forma un paquete de

datos que se transfieren al canal lógico para su transmisión; si no es posible obtener un paquete de datos, se envía a la terminal la señal de servicio de buffer lleno y los datos acumulados son descartados. Son condiciones de envío de datos el llenado del buffer de entrada, el recibir un carácter de expulsión o el recibir el comando expulsión.

III.1.5. Envío de un paquete de interrupción

Esta función permite, desde el nivel de sesión, el envío de un paquete de interrupción.

Nombre:

op\$interrupcion

Sintaxis:

call op\$interrupcion(reg\$ptr);

reg\$ptr: es un apuntador a estructura tipo terminal

Descripción:

Cada vez que se solicita el servicio de esta función se verifica, en primer término, que el usuario esté involucrado en una llamada virtual, de no ser así se envía al usuario la señal de servicio de no hay conexión; para proceder a la petición se solicita un paquete de negociación y se envía el comando apropiado a través de la estructura canal hacia el proceso que maneja el canal lógico, si no es posible asignar el paquete de negociación solicitado, se envía al usuario la señal de servicio de buffer lleno.

III.1.6. Envío de un paquete de RNR

Esta función permite el envío de paquetes de RNR (Receiver Not Ready) con fines de control de flujo.

Nombre:

op\$rnr

Sintaxis:

```
call op$rrr(reg$ptr);
```

reg\$ptr: es un apuntador a estructura tipo terminal

Descripción:

Esta función permite hacer control de flujo a nivel paquetes mediante el envío de un paquete de RNR (la recepción de un paquete de RNR, indica a la interfaz ETD/ETCD que el extremo remoto se encuentra indispuerto para aceptar más datos); antes de intentar satisfacer esta petición, se verifica que el usuario en cuestión esté involucrado en una llamada virtual, de no ser así, se envía al usuario la señal de servicio de no conexión, y en caso de proceder la solicitud, se envía el comando apropiado al proceso manejador del canal lógico a través de la estructura canal.

III.1.7. Envío de un paquete de RR

Esta función permite el envío de paquetes de RR (Receiver Ready) con fines de control de flujo.

Nombre:

```
op$rrr
```

Sintaxis:

```
call op$rrr(reg$ptr);
```

reg\$ptr: es un apuntador a estructura tipo terminal

Descripción:

Esta función permite hacer control de flujo a nivel paquetes mediante el envío de un paquete de RR (la recepción de un paquete de RR, indica a la interfaz ETD/ETCD que el extremo remoto se encuentra dispuesto a aceptar más datos); antes de intentar satisfacer esta petición, se verifica que el usuario en cuestión esté involucrado en una llamada virtual, de no ser así, se envía al usuario la señal de servicio de no conexión, y

en caso de proceder la solicitud, se envía el comando apropiado al proceso manejador del canal lógico a través de la estructura canal.

III.1.8. Solicitud de reiniciación del circuito virtual

Esta función permite que se envíe un paquete de petición de reiniciación del circuito virtual.

Nombre:

op\$reinicia

Sintáxis:

call op\$reinicia(reg\$ptr);

reg\$ptr: apuntador a una estructura tipo terminal.

Descripción:

La función indaga si la terminal-usuario en cuestión se encuentra involucrada en una llamada virtual, de no ser así, se envía al usuario la señal de servicio de no hay conexión; en caso de proceder la petición se envía el comando apropiado a la estructura canal correspondiente (hacia el nivel de paquetes), para que el proceso que maneja el canal lógico haga el envío del paquete apropiado de petición de reiniciación.

III.1.9. Solicitud de liberación de la llamada

Esta función permite liberar una llamada virtual.

Nombre:

op\$peticion\$liberacion

Sintáxis:

call op\$peticion\$liberacion(reg\$ptr);

reg\$ptr: apuntador a una estructura tipo terminal

Descripción:

Cuando este servicio es solicitado, se indaga, en primer término, si el usuario se encuentra en una llamada virtual, de no ser así se le envía la señal de servicio de no hay conexión; en caso de proceder la petición, se envía hacia la estructura canal el comando apropiado para que el proceso asociado a ese canal inicie la solicitud de liberación de la llamada en el nivel de red.

III.2. Indicaciones del nivel de transporte

III.2.1. Indicación de arribo de paquete de datos

Anuncia la llegada de un paquete de datos para el usuario.

III.2.2. Indicación de circuito establecido

Anuncia que se ha establecido una llamada virtual, ya sea porque haya sido iniciada por el usuario en cuestión, o bien porque el usuario haya sido seleccionado en una llamada entrante.

III.2.3. Indicación de reiniciación

Anuncia la reiniciación del canal lógico resultado del arribo de un paquete de confirmación de reiniciación o de indicación de reiniciación.

III.2.4. Indicación de rearmar

Anuncia la reiniciación del canal lógico y la liberación de la llamada virtual en caso de haber existido, como resultado del arribo de un paquete de confirmación de rearmar o de indicación de rearmar.

III.2.5. Indicación de enlace interrumpido

Anuncia que el enlace se ha interrumpido (se refiere al nivel 2 de X.25).

III.2.6. Indicación de circuito virtual liberado

Anuncia que la llamada virtual ha sido liberada, es decir, se recibió un paquete de confirmación de liberación o de indicación de liberación.

III.2.7. Indicación de arribo de paquete de interrupción

Anuncia que se ha recibido un paquete de interrupción desde el equipo remoto.

III.3. Resumen de códigos de comandos e indicaciones

Los cuadros III.3a y III.3b muestran los códigos utilizados por las funciones del nivel de transporte para comunicarse con los procesos manejadores de canal lógico, así como los códigos que espera una capa superior de los manejadores de canal lógico.

Comando	Código
Solicitud de canal lógico	sin/cód
Solicitud de paquete de negociación	sin/cód
Solicitud de reinicio del circuito virtual	5
Selección	6
Peticion de liberación de la llamada	7
Envío de datos	8
Envío de un paquete de interrupción	9
Envío de paquete de RNR	10
Envío de paquete de RR	11

Cuadro III.3a: Códigos de comando aceptados por un manejador de canal lógico.

INTERFAZ NIVEL DE RED/NIVEL DE SESION

Indicación	Código
Indicación de arribo de paquete de datos	0
Indicación de circuito establecido	1
Indicación de reiniciación	2
Indicación de rearranque	3
Indicación de enlace interrumpido	4
Indicación de circuito virtual liberado	5
Indicación de arribo de paquete de interrupción	6

Cuadro III.3b: Códigos de indicación enviados por un manejador de canal lógico hacia una capa superior.

IV. ARQUITECTURA DEL SOFTWARE

IV.1. Organización lógica de los módulos de software

La figura IV.1 muestra la interrelación entre los diferentes módulos del software, no muestra a los archivos mismos, ya que en algunos casos, el módulo que aparece en la figura IV.1, está formado por más de un archivo compilado separadamente; la programación se realizó con el lenguaje PL/M-86 y algunas partes con ASM-86.

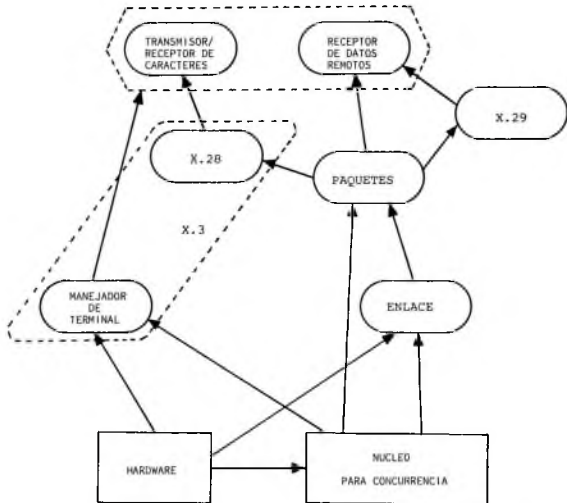


Figura IV.1: Interrelación de los diferentes módulos de software.

IV.1.1. Hardware

La parte de hardware a que se hace referencia en la figura, es la que se relaciona con las interfaces de comunicación serie asincrónica, que proveen de entrada y salida de caracteres al manejador de terminal; al procesador de comunicaciones WD2511, que junto con el módulo enlace proveen el nivel 2 de la recomendación X.25; y al reloj, que permite al módulo núcleo dar el servicio de temporizadores.

IV.1.2. Núcleo para concurrencia

El núcleo para concurrencia permite, básicamente, crear procesos, activarlos y bloquearlos; provee las primitivas de sincronización entre los procesos, como son signal y wait, el manejo de temporizadores y el manejo dinámico de la memoria.

IV.1.3. Manejador de terminal

Provee de un mecanismo de entrada/salida y de control de flujo a nivel caracteres entre la terminal y el EDD, apto para ser utilizado en un ambiente concurrente, es en este módulo donde son tomados en cuenta algunos de los parámetros de terminal definidos por la recomendación X.3.

IV.1.4. Enlace

Este módulo de software, junto con el procesador de comunicaciones (WD2511), proveen la facilidad de transmisión y recepción de tramas de acuerdo con el nivel 2 de la recomendación X.25 que corresponde al nivel 2 del modelo OSI.

IV.1.5. Paquetes

Permite el manejo de canales lógicos, el establecimiento de llamadas virtuales, el intercambio de datos durante las mismas, la liberación de tales llamadas, así como el mecanismo de detección/corrección de errores mediante retransmisión y de control de flujo, todas estas acciones están definidas por la recomendación X.25.

IV.1.6. X.28

El módulo X.28 recibe caracteres del proceso Tx/Rx (que se

verá mas adelante), hace una interpretación de los mismos (como señales de instrucción o bien como datos a transmitir), da las respuestas apropiadas (señales de servicio) y genera las acciones correspondientes (p.ej.: la petición de llamada o el envío de datos cuando se está en modo transferencia de información), todo esto de acuerdo a la recomendación X.28.

IV.1.7. X.29

El módulo X.29 recibe los paquetes de datos que llegan con el bit calificador (bit Q) puesto a 1, ya que éstos contienen mensajes de EDD (comandos remotos); este módulo se encarga de interpretar esos mensajes de EDD y de generar las acciones y/o respuestas apropiadas de acuerdo a la recomendación X.29.

IV.1.8. Transmisor/Receptor de caracteres (Tx/Rx)

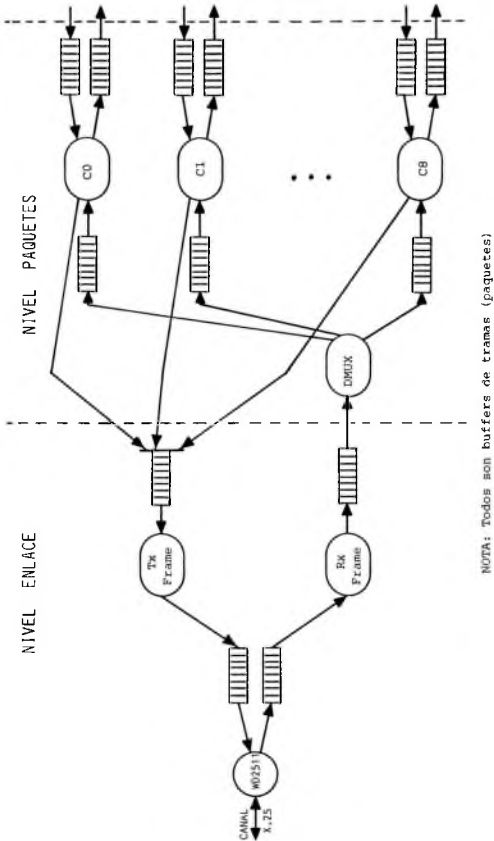
Este módulo es el que interactúa directamente con el usuario, es quien recibe los caracteres del manejador de terminal, para que, en cooperación con el módulo X.28, haga una interpretación de las cadenas de caracteres que le envía el usuario, ya sea para determinar que señal de instrucción es, o para interpretarlas como datos a transmitir por el circuito virtual (esto es, cuando se ha establecido previamente una llamada virtual); si ha de hacerse eco a los caracteres recibidos, es aquí donde se realiza.

IV.1.9. Receptor de datos remotos

Este módulo es el que entrega al usuario los datos recibidos desde el otro extremo de la llamada virtual, los paquetes de datos recibidos con el bit calificador puesto a 1 son transferidos al módulo X.29 para su tratamiento; este módulo recibe además las indicaciones de establecimiento de la llamada, liberación de la llamada, etc. que provienen del nivel paquetes para hacerselas saber al usuario.

IV.2. Esquema de intercambio de información y procesos involucrados

La figura IV.2 muestra las estructuras de datos y los procesos de los niveles de enlace y paquetes donde la unidad de intercambio de información es la trama (el paquete es el campo de información de una trama); la figura IV.3 muestra las estructuras de datos y los procesos de los niveles superiores al de paquetes, donde la unidad de intercambio de información es el carácter.



NOTA: Todos son buffers de tramas (paquetes)

Figura IV.2: Procesos e intercambio de datos a nivel tramas.

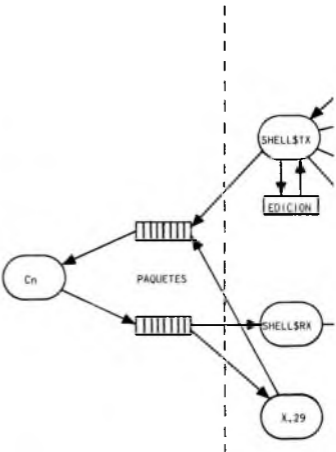
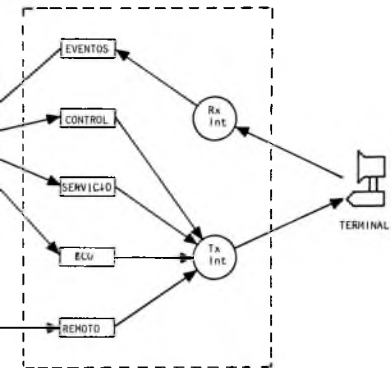


Figura IV.3: Procesos e intercambio de datos a nivel caracteres.

MANEJADOR DE TERMINAL



BUFFERS DE CARACTERES

IV.3. Proceso transmisor/receptor de caracteres

El proceso transmisor/receptor de caracteres, conocido como shell\$tx en el software, está dividido en dos partes, una para cuando el usuario está en modo comando y otra para cuando está en modo transferencia de datos, es en este proceso principalmente donde se hace uso de los servicios dados por el nivel de transporte, servicios que solicita shell\$tx y que serán atendidos por el nivel de paquetes. A continuación se presenta un pseudocódigo que pretende hacer notar las funciones realizadas por este proceso en sus dos modalidades.

```
shell$tx:
```

```
Ciclo
```

```
{
  Espera(petición de servicio);
  Da señal de bienvenida;
  Mientras(terminal activa) {
    Mientras(modo comando) {
      ejecuta(recibe comando);
    }
    Mientras(en estado de transferencia de datos) {
      Espera(evento);
      Switch(evento) {
        carácter:
          Si (rellamada(carácter)) {
            solicita envio de RNR;
            Si (rellamada(nuevo carácter))
              almacena(carácter de rellamada);
            Si no
              ejecuta(recibe comando);
            solicita envio de RR;
          } Si no
            almacena(carácter);
        expulsión:
          los caracteres almacenados deben ser
          transmitidos;
        escape:
          solicita envio de RNR;
          ejecuta(recibe comando);
          solicita envio de RR;
        espera de página:
          Espera(cancelación de espera de página);
          recibe X$OFF del usuario;
          solicita envio de RNR;
          recibe X$ON del usuario;
          solicita envio de RR;
      }
    }
  }
}
```

A continuación se dará una mayor explicación de las funciones que se distinguen en el pseudocódigo anterior:

Petición de servicio

La petición de servicio se entiende como el mecanismo que permite al usuario de una terminal, indicar al EDD que a partir de ese momento el usuario requerirá de sus servicios. La petición de servicio consiste en la transmisión, hacia el EDD, de dos caracteres de retroceso de carro, cualquier carácter diferente de retroceso de carro será ignorado; la petición de servicio permite al EDD dar de alta a una terminal activa; terminales activas, pero que no han sido dadas de alta, no pueden aceptar llamadas entrantes.

Terminal activa

Una terminal estará activa si está encendida, y más estrictamente hablando, si tiene encendida su señal de DTR. El EDD hace un escudriñamiento periódico sobre todas las terminales, de modo que sabe cuando una terminal cambia de estado activo a inactivo o viceversa; esto permite generar las acciones de liberación del circuito virtual cuando la terminal correspondiente es apagada o se descompone antes de que el usuario solicite la liberación de la llamada.

Modo comando

Se considerará al usuario en modo comando siempre que no esté establecida una llamada virtual, o cuando se haya solicitado pero aún no haya sido aceptada la petición de llamada (es posible pasar temporalmente al modo comando cuando se está en modo transferencia de datos).

Recibe comando

Permite obtener una cadena de caracteres de la terminal; sobre esta cadena se realiza un análisis léxico, y si no contiene errores, se le hace un análisis sintáctico, y en caso de no reportar errores devuelve una cadena de unidades sintácticas, las cuales podrán ser ejecutadas; en cada uno de los análisis, si un error es detectado, se le indica al usuario sin darle información

adicional sobre donde se encontró el error. La figura IV.4 muestra el autómata del analizador léxico y la figura IV.5 el autómata del analizador sintáctico, se puede observar allí las diferentes señales de instrucción definidas por la recomendación X.28, así como las señales de instrucción no normalizadas adicionadas a nuestra implantación.

Ejecuta

Recibe una cadena de unidades sintácticas válidas, las cuales son recorridas ejecutando en cada caso la acción que corresponda, aquí se envían las señales de servicio que corresponden a la ejecución de la señal de instrucción recibida; la ejecución comprende desde la lectura de los parámetros de terminal, hasta la construcción de un paquete de petición de llamada, todo esto de acuerdo a la recomendación X.28.

En estado de transferencia de datos

Se estará en el estado de transferencia de datos siempre que se haya establecido una llamada virtual y no se haya escapado al modo comando, en este estado es permitido el intercambio de información extremo a extremo del circuito virtual establecido.

Rellamada

La rellamada permite pasar temporalmente al modo comando; dado un carácter recibido, se analiza si el escape al modo comando es permitido y si el carácter en cuestión es el carácter de rellamada.

Solicita envío de RNR

Permite el envío de un paquete de receptor no listo (RNR) hacia el otro extremo del circuito virtual, con el fin de detener el flujo de paquetes de datos entrantes.

Solicita envío de RR

Permite el envío de un paquete de receptor listo (RR) hacia el otro extremo del circuito virtual, con el fin de rehabilitar el flujo de paquetes de datos entrantes.

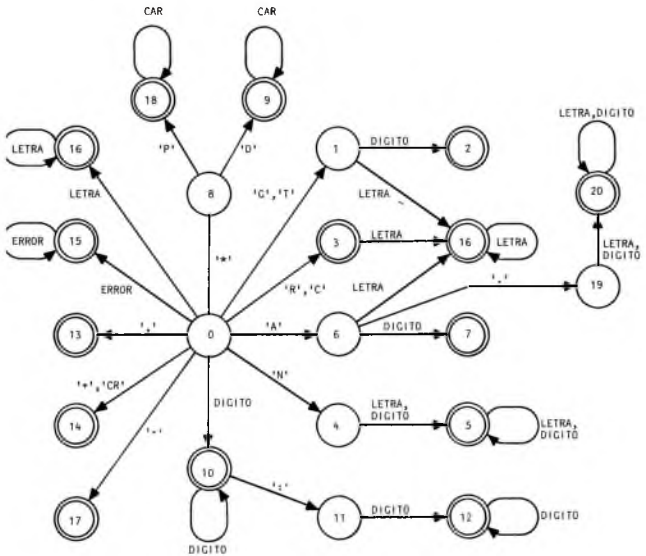


Figura IV.4: Autómata del analizador léxico.

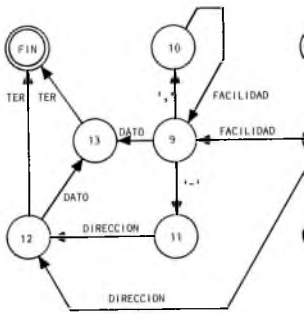
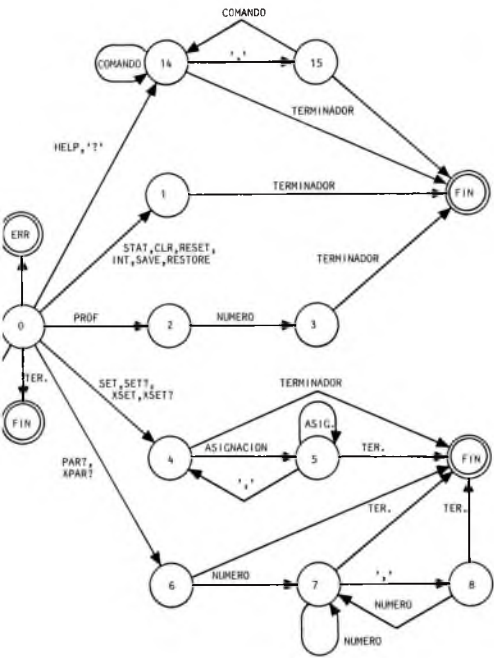


Figura IV.5: Autómata del analizador sintáctico.



ARQUITECTURA DEL SOFTWARE

Almacena

Permite que los caracteres de datos recibidos desde la terminal sean almacenados hasta que se de una condición de expulsión de datos, la cual consiste en el empaquetado de los caracteres y su transmisión a través del circuito virtual; ejemplos de condiciones de expulsión son el llenado del buffer de edición (donde se van almacenando los caracteres) o la expiración del temporizador de reposo (define el tiempo que se espera desde el último carácter recibido antes de que se obligue a la expulsión de los datos).

Los eventos de la instrucción "switch" son los siguientes:

- Carácter.- Se ha recibido un carácter de la terminal.

- Expulsión.- Se le indica a shell\$tx que debe expulsar los caracteres hasta el momento almacenados en su buffer de edición, la ocurrencia de este evento, cuando no haya caracteres en el buffer de edición, no ocasionará que se transmitan paquetes de datos vacíos.

- Escape.- Se ordena a shell\$tx que, sin haber recibido el carácter de rellamada, debe pasar temporalmente al modo comando.

- Espera de página.- Se indica que se ha presentado la condición de espera de página (ver parámetro 22), y por lo tanto, antes de recibir más caracteres del usuario, debe esperar a que se de la condición de cancelación de espera de página.

- Recibe X\$OFF del usuario.- Se le indica a shell\$tx que el usuario ha ordenado la suspensión del flujo de caracteres del EDD hacia su terminal, y por lo tanto se debe prevenir el no arribo de más paquetes de datos desde el otro extremo del circuito virtual.

- Recibe X\$ON del usuario.- Se le indica a shell\$tx que el usuario ha rehabilitado el flujo de caracteres hacia su terminal, y por lo tanto se rehabilita también el flujo de paquetes de datos desde el otro extremo del circuito virtual.

IV.4. Proceso receptor de datos remotos

Este proceso, conocido como shell\$rx en el software, es el que recibe, desde el nivel de paquetes, los paquetes de datos así como las indicaciones que da el nivel de transporte relativas al nivel de paquetes; los paquetes de datos que se le entregan con el bit calificador puesto a 1 son transferidos inmediatamente a otro proceso encargado de su tratamiento.

A continuación se presenta en pseudocódigo, las funciones de este proceso.

```

shell$rx:
Ciclo (
  Espera(evento);
  Switch(evento) {
    arribo de paquete de datos:
      Si (bit Q=1)
        envía paquete a x29$process;
      Si no
        envía datos a usuario;
    circuito establecido:
      saca a shell$tx de (recibe comando);
      inicia buffers de caracteres;
      envía al usuario la señal de servicio de comunicación
        establecida;
    reinicio:
      hace la indicación al usuario de la ocurrencia de este
        evento;
    rearranque:
      hace la indicación al usuario de la ocurrencia de este
        evento;
    enlace interrumpido:
      hace la indicación al usuario de la ocurrencia de este
        evento;
    circuito liberado:
      hace la indicación al usuario de la ocurrencia de este
        evento;
    interrupción:
      hace la indicación al usuario de la ocurrencia de este
        evento;
  }
}

```

A continuación se da una breve explicación de cada una de las opciones de la instrucción "switch".

Arribo de paquete de datos

Cuando se recibe un paquete de datos, éste puede contener información de control entre un equipo remoto y el EDD, esto es, se trata de un mensaje de EDD que ha de ser transferido al proceso adecuado para que lo decodifique y realice las acciones apropiadas; si se trata de un paquete de datos con el bit Q puesto a 0, se trata de datos dirigidos al usuario, y deberá chequearse el parámetro 8 para ver si se entrega al usuario, o simplemente se descarta sin hacer ninguna notificación de ello.

Circuito establecido

Se ha establecido una llamada virtual, ya sea como consecuencia de una solicitud remota, o bien porque el usuario hizo una petición de llamada y ésta está siendo reconocida.

Reinicio

El circuito virtual ha sido reiniciado (las variables para la numeración y reconocimiento de los paquetes de datos se han puesto a cero) como consecuencia de la solicitud remota de esta acción, o bien, por el reconocimiento de la misma solicitud hecha por el usuario.

Rearranque

El nivel de paquetes ha sido reiniciado, esto significa que todas las llamadas virtuales han sido liberadas y que todos los canales lógicos han pasado al estado de listo para establecer llamadas virtuales.

Enlace interrumpido

El nivel 2 de X.25 no está preparado para intercambiar tramas de ningún tipo, para poder hacerlo debe establecerse el enlace.

Circuito liberado

La llamada virtual previamente establecida ha sido liberada, ya sea desde el lado remoto de la llamada, o por el usuario mismo.

Interrupción

Se ha recibido un paquete de interrupción.

IV.5. Proceso manejador de mensajes de EDD

Este proceso conocido en el software como `x29$process`, y al que se hace referencia al tratar sobre el módulo de software de X.29, es el encargado de interpretar los mensajes de EDD y de dar las respuestas apropiadas a ellos; los mensajes que maneja, descritos en II.5 son:

- Mensaje de EDD de asignación de valores
- Mensaje de EDD de lectura
- Mensaje de EDD de asignación de valores y lectura
- Mensaje de EDD de indicación de parámetros
- Mensaje de EDD de indicación de corte
- Mensaje de EDD de error

A continuación se presenta en pseudocódigo la estructura del proceso `x29$process`.

```

x29$process:
Ciclo {
  Espera(evento);
  Switch(evento) {
    mensaje de EDD:
      Switch(mensaje de EDD) {
        indicación de parámetros:
          Si no (esperaba indicación de parámetros)
            envía mensaje de EDD de error;
        invitación a liberar:
          sin acción alguna;
        asignación de valores:
          asigna valores de parámetros;
          Si (hay error en la asignación)
            envía mensaje de EDD de indicación de
              parámetros;
        indicación de corte:
          envía mensaje de EDD de asignación del parámetro 8;
          envía señal de corte a la terminal;
        lectura de parámetros:
          envía mensaje de EDD de indicación de parámetros;
        error:
          sin acción alguna;
        asignación de valores y lectura:
          asigna valores de parámetros;
          envía mensaje de EDD de indicación de parámetros;
        reelección:
          sin acción alguna;
      }
    envía paquete:
      envía un paquete de datos con Q=1;
    corte:
      envía un mensaje de EDD de indicación de corte;
  }
}

```

A continuación se da una breve explicación de algunas de las acciones realizadas por este proceso.

Asignación de valores

Al tratar de realizar la asignación de los parámetros indicados, pueda ser que se trate de un acceso no válido a tal parámetro, en caso de que esto ocurra con una o más de las asignaciones, se deberá enviar un mensaje de EDD de indicación de

parámetro sólo con las referencias a parámetros que resultaron inválidas, los campos de parámetro de este mensaje se codificarán de manera diferente y será de la siguiente manera:

Campo de parámetro	
Referencia de parámetro:	1 número de parámetro
Valor de parámetro:	código de error

El bit 8 puesto al valor 1 (los bits se numeran de 1 a 8 y el bit más significativo es el 8) indica el acceso inválido al parámetro mencionado.

Asignación de valores y lectura

En este caso como en el anterior, también podrá ocurrir que se intenten accesos inválidos a los parámetros; el mensaje de EDD de indicación de parámetros que se envíe como respuesta, contendrá tanto las referencias válidas a parámetro, como las inválidas, codificando las últimas de la manera antes descrita.

Mensajes de EDD sin acción asociada

Los mensajes de EDD que al recibirlos no causan acción alguna en el EDD son:

- i) Mensaje de EDD de invitación a liberar
- ii) Mensaje de EDD de reselección

Para éstos, la recomendación X.29 define ciertas acciones pero no han sido implantadas en la presente versión del EDD.

- iii) Mensaje de EDD de error

La recomendación X.29 solo define los mensajes de EDD que ha de transmitir el EDD en ciertas situaciones, pero no define acción alguna para los mensajes de EDD de error recibidos.

Envía paquete

Esta opción, de la primera instrucción "switch", se utiliza

únicamente para permitir al usuario enviar mensajes de EDD (paquetes de datos con el bit calificador puesto a 1), cosa no permitida según las recomendaciones del CCITT para un EDD.

Corte

Cuando el usuario de la terminal transmite la señal de corte, dependiendo del valor del parámetro 7, se enviará hacia el equipo remoto un mensaje de EDD de indicación de corte.

IV.6. Manejador de terminal

El manejador de terminal es el software que permite la entrada de caracteres de las interfaces de comunicación serie asincrónica, y la salida de los mismos desde los procesos hacia las interfaces, permite además el control de flujo mediante el intercambio de los caracteres DC1 y DC3 para conmutar a cerrado o abierto un dispositivo auxiliar, y el manejo de algunos de los parámetros de terminal.

Como ya se mostró en la figura IV.3, existen 4 buffers de salida de caracteres y un buffer de entrada de eventos (donde un evento es el arribo de un carácter) por cada terminal, la disponibilidad de espacio libre en los buffers de salida está controlada mediante semáforos, así como la disponibilidad de un evento es también indicada mediante un semáforo.

Los 4 buffers de salida son:

- Control
- Servicio
- Eco
- Remoto

Buffer de control

El buffer de control se utiliza para el intercambio de caracteres de control de flujo (DC1 y DC3), este es el buffer de mayor prioridad, es decir, siempre que la interfaz de comunicación serie esté dispuesta a aceptar un nuevo carácter, lo busca primero en el buffer de control, después en el de señales de servicio, luego en el de eco y por último en el de datos remotos.

Buffer de señales de servicio

Toda vez que el EDD desee enviar a la terminal cualquier señal de servicio como respuesta a una señal de instrucción, o por iniciativa del mismo EDD, los caracteres serán insertados en este buffer.

Buffer de eco

En este buffer son insertados todos los caracteres a los cuales ha de hacerse eco, el eco estará sujeto al valor de los parámetros 2 y 20, y dependerá además de algunas situaciones definidas por la recomendación X.28, que dice que durante la introducción de cierta señal de instrucción, el eco debe deshabilitarse independientemente del valor de los parámetros de terminal.

Buffer de datos remotos

Toda vez que estando en el modo transferencia de datos, se reciban paquetes de datos a través del circuito virtual, los caracteres del campo de información de estos paquetes deben ser enviados a la terminal, en caso de proceder, a través de este buffer.

Buffer de eventos (buffer de entrada)

Como se aprecia también en la figura IV.3, existe un buffer de eventos por cada terminal, los eventos pueden ser:

- * Se recibe un carácter de la terminal.
- * Desbloqueo del proceso shell\$tx cuando se encuentra en el modo de transferencia de datos (por causa de la liberación de la llamada virtual desde el extremo remoto).
- * Se ha establecido una llamada virtual, y el proceso shell\$tx, que esperaba por una señal de instrucción, debe desbloquearse para pasar al modo transferencia de datos.
- * Ha expirado el temporizador de reposo, y el proceso shell\$tx deberá expulsar los caracteres almacenados en el buffer de edición (se está en el modo de transferencia de datos).

- * Se le indica al proceso shell\$tx que se ha recibido la señal de corte desde la terminal.

- * El proceso shell\$tx debe escapar al modo comando sin haber recibido el carácter de rellamada.

- * Se ha detectado una condición de espera de página.

- * Se ha hecho control de flujo desde la terminal mediante el carácter DC3 (X\$OFF).

- * Se ha hecho control de flujo desde la terminal mediante el carácter DC1 (X\$ON).

Como puede apreciarse en la figura IV.3, existen dos rutinas de servicio a interrupción, una para entrada y otra para salida, estas interrupciones señalizan al manejador de terminal la disponibilidad de un nuevo carácter que podrá ser introducido al buffer de eventos, y la disposición para transmitir un nuevo carácter de alguno de los buffers de salida.

Cuando el buffer de eventos sobrepasa cierto nivel, se intenta hacer control de flujo (esto estará sujeto al valor de los parámetros de terminal); si el buffer de eventos se llena, sucesivos caracteres serán descartados; en caso de que el buffer de salida al que se hace referencia se llene, el proceso que intenta transmitir caracteres hacia la terminal será bloqueado en espera de espacio en el buffer de salida al que se hace referencia, no teniendo otra consecuencia.

IV.7. Implantación de la recomendación X.3

Los parámetros de terminal son verificados en diferentes módulos de software, cada uno donde más conviene a la eficiencia de su manejo, se da a continuación, de una manera breve, la ubicación del manejo de cada uno de los parámetros de terminal, haciendo referencia al proceso involucrado y a la función donde se checa.

IV.7.1. Referencia 1

Rellamada al EDD utilizando un carácter

El proceso shell\$tx, cuando está en el modo transferencia de datos, analiza, mediante la función "rellamada", si el carácter que se recibe es el definido por este parámetro, para escapar temporalmente al modo comando.

IV.7.2. Referencia 2

Eco

El proceso shell\$tx estando en cualquiera de sus dos estados, al recibir cualquier carácter, determina si ha de hacersele eco, la manera de hacer esto es mediante un atributo ECO, que tiene cada uno de los caracteres de cada terminal, este parámetro opera en conjunción con el parámetro 20.

IV.7.3. Referencia 3

Elección de caracteres de envío de datos

El proceso shell\$tx, mediante un servicio del nivel de transporte (op\$envia\$datos), envía datos hacia el extremo remoto, esta función, cada vez que recibe un nuevo carácter, analiza su atributo CHARACTER_DE_EXPULSION para determinar si el carácter en cuestión es de los identificados con esa propiedad.

IV.7.4. Referencia 4

Elección de la duración de la temporización de reposo

El proceso shell\$tx, al hacer el llamado a la función del nivel de transporte "op\$envia\$datos", dependiendo del valor de este parámetro, podrá rearrancar el temporizador de reposo cada vez que un nuevo carácter es recibido, si este temporizador expira (dejan de recibirse caracteres), enviará un evento a través del

buffer de eventos del manejador de terminal, de modo que el shell\$tx sepa que debe expulsar los caracteres hasta el momento almacenados.

IV.7.5. Referencia 5 **Control de dispositivo auxiliar**

Este parámetro es tomado en cuenta en la rutina de servicio para entrada de caracteres del manejador de terminal, cuando el buffer de entrada de eventos sobrepasa cierto límite (se asume que un alto porcentaje de los eventos será de carácter disponible), se observa el valor de este parámetro, y si es permitido, se hace el control de flujo de acuerdo como se define, se observa también el límite inferior de este buffer para que en caso de estar detenido el flujo de caracteres, éste sea habilitado nuevamente.

IV.7.6. Referencia 6 **Control de señales de servicio de EDD**

Existe un módulo de software que en el esquema general quedaría dentro del manejador de terminal, este módulo es el encargado de manejar las señales de servicio, se utiliza la función "envia\$servicio", la cual chequea el valor de este parámetro para proceder, o no, a insertar en el buffer de servicio los caracteres de la señal de servicio de que se trate; esta función es invocada desde los procesos shell\$tx y shell\$rx, así como desde las funciones provistas por el nivel de transporte, las cuales son invocadas por estos procesos.

IV.7.7. Referencia 7 **Elección de la operación que ha de efectuar el EDD al recibir la señal de corte**

Al recibirse en el manejador de terminal, en la rutina de servicio a interrupción por carácter disponible, la señal de corte, dependiendo del valor de este parámetro, se enviarán eventos al proceso shell\$tx y/o al proceso x29\$process, de modo que se ejecuten las acciones que define este parámetro.

IV.7.8. Referencia 8 **Descartar salida**

En el proceso shell\$rx, cuando se reciben paquetes de datos,

antes de proceder a enviarlos al buffer de datos remotos para ser enviados al usuario, se observa el valor de este parámetro para decidir si tal envío se efectúa o si los datos deberán ser descartados.

IV.7.9. Referencia 9 Relleno después del retroceso de carro

Existe una función en el manejador de terminal, que permite la inserción de caracteres en un buffer de salida especificado, cada vez que recibe un carácter de retroceso de carro, observa este parámetro para determinar cuantos caracteres nulos o ninguno debe insertar después del retroceso de carro.

IV.7.10. Referencia 10 Nueva línea (o delimitación de línea)

Cada vez que la rutina de servicio a interrupción por disponibilidad de la interfaz para aceptar un nuevo carácter para su transmisión, interrumpe, se selecciona un nuevo carácter para su transmisión de los diferentes buffers de salida, en ese momento, si se trata de un carácter gráfico (visible), incrementa un contador de columnas y observa el valor de este parámetro, de modo que sabe si debe insertar o no automáticamente la secuencia de retroceso de carro y avance de línea.

IV.7.11. Referencia 11 Velocidad binaria del ETD arritmico

Este parámetro sólo se observa al iniciar al manejador de terminal, al programar las interfaces de comunicación serie, y cuando este parámetro se modifica, de tal suerte que se deba reprogramar dicha interfaz.

IV.7.12. Referencia 12 Control de flujo del EDD por el ETD

En la rutina de servicio a interrupción por carácter recibido, se analiza si tal carácter es de control de flujo, en cuyo caso se observa este parámetro para ver si el control de flujo sobre el EDD está permitido, y de ser así, se deshabilita mediante una bandera de transmisión de caracteres hacia el ETD arritmico (la terminal).

IV.7.13. Referencia 13

Inserción del cambio de renglón después del retroceso de carro

Este parámetro se observa en 2 puntos del software, y se hace o no la inserción del cambio de renglón dependiendo de la procedencia del carácter de retroceso de carro, haciéndose esta observación en:

- 1) La función que inserta caracteres en uno de los buffers de salida, haciendo la observación en el buffer de eco y el de datos remotos.
- 2) La función del nivel de transporte para envío de datos.

IV.7.14. Referencia 14

Relleno después del cambio de renglón

Existe una función en el manejador de terminal, que permite la inserción de caracteres en un buffer de salida especificado, cada vez que recibe un carácter de cambio de renglón, observa este parámetro para determinar cuantos caracteres nulos o ninguno debe insertar después del carácter de cambio de renglón.

IV.7.15. Referencia 15

Edición

Este parámetro se observa en conjunción con los parámetros 16, 17 y 18; el proceso shell\$tx que ha de consumir caracteres, ya sea en modo comando o modo transferencia de datos, utiliza para cada caso una función del módulo manejador de terminal para la obtención de caracteres, la que utiliza en modo transferencia de datos observa este parámetro para hacer la función de edición apropiada de acuerdo a la facilidad de edición a usar, mientras que en modo comando, se deja al analizador léxico, que consume los caracteres, la observación de este parámetro.

IV.7.16. Referencia 16

Carácter de edición de supresión de carácter

Se observa en conjunción con el parámetro 15.

IV.7.17. Referencia 17**Carácter de edición de supresión de línea**

Se observa en conjunción con el parámetro 15.

IV.7.18. Referencia 18**Carácter de edición de visualización de línea**

se observa en conjunción con el parámetro 15.

IV.7.19. Referencia 19**Señales de servicio de EDD de edición**

Este parámetro se verifica en la función del manejador de terminal encargada de el envío de señales de servicio; este parámetro se observa en conjunción con el parámetro 6.

IV.7.20. Referencia 20**Máscara de eco**

Este parámetro se maneja como un atributo ECO asociado a cada carácter en cada estructura terminal, se maneja como un bit que puede estar o no encendido; cada vez que se utiliza la función `put$char`, que inserta un carácter en un buffer de salida especificado, se observa este atributo para determinar si procede o no la inserción del carácter en el buffer de eco; este parámetro opera en conjunción con el parámetro 2.

IV.7.21. Referencia 21**Tratamiento de la paridad**

La paridad podrá generarse en los caracteres transmitidos hacia la terminal y/o verificarse en los caracteres recibidos de la misma; en el caso de los caracteres de salida, se genera en la función `transmite$carácter`, que es llamada cada vez que la interfaz de comunicación serie genera una interrupción por lista para transmitir; en los caracteres de entrada es verificada por la función de más bajo nivel `"servicio$read"`, que permite a un proceso obtener un nuevo carácter. La paridad se maneja mediante un atributo que indica para cada carácter, la paridad que le corresponde y que debe de generarse y/o verificarse; este parámetro opera en conjunción con el parámetro extra 2.

IV.7.22. Referencia 22
Espera de página

La condición de espera de página se detecta en la función `transmite$caracter` que se invoca desde la rutina de servicio por listo para transmitir, cada vez que se transmite hacia la terminal el carácter de avance de línea, un contador se incrementa si es que está activo este parámetro, de tal suerte que si se da la condición de espera de página, se envía un evento a `shell$tx`, que a su vez es quien espera por la condición de cancelación de espera de página (esta condición se da al recibir el carácter `X$ON` o el carácter de retroceso de carro).

IV.7.23. Referencia x-1
Elección de mayúsculas y minúsculas

Este parámetro se observa en el momento de transmitir un carácter en la función `transmite$caracter`.

IV.7.24. Referencia x-2
Tipo de paridad

Este parámetro se observa en el momento de transmitir un carácter en la función `transmite$caracter` y en la función `servicio$read` al momento de leerlo; este parámetro opera en conjunción con el parámetro 21.

IV.7.25. Referencia x-3
Carga del perfil inicial

Este parámetro se observa al iniciar al manejador de terminal o al ejecutar el comando "restore"; este parámetro opera en conjunción con el parámetro x-4.

IV.7.26. Referencia x-4
Perfil inicial

Este parámetro se observa al iniciar al manejador de terminal o al ejecutar el comando "restore"; este parámetro opera en conjunción con el parámetro x-3.

IV.7.27. Referencia x-5
Idioma

Este parámetro es observado al momento de enviar señales de servicio, en la función envia\$servicio.

IV.7.28. Referencia x-6

Parámetro sin función asignada.

IV.7.29. Referencia x-7

Parámetro sin función asignada.

IV.7.30. Referencia x-8
Uso de la señal RTS

Este parámetro se observa en dos puntos dependiendo de su valor, uno, al transmitir caracteres, dependiendo de si hay o no caracteres pendientes por transmitir; el otro, en las funciones para control de flujo sobre la terminal; este parámetro opera en conjunción con el parámetro 5.

IV.7.31. Referencia x-9

Parámetro sin función asignada.

IV.7.32. Referencia x-10

Parámetro sin función asignada.

IV.8. Implantación de las recomendaciones X.28 y X.29

Ya se dijo donde se manejan los comandos definidos por la recomendación X.28, ya se mostraron también los autómatas del analizador léxico y del analizador sintáctico, sólo quedaría por mostrar los detalles de implantación, lo cual requeriría entrar a un nivel de detalle tal, que sería necesario mostrar el código de los programas, lo cual me parece una labor muy extensa e injustificada para el propósito de este trabajo, por tal razón, he considerado pertinente no incluirlo. Para más detalle sobre esta recomendación remito al lector a la literatura editada por el CCITT.

Lo mismo puedo decir de la implantación de la recomendación X.29, pues ya he dado el esquema general del proceso encargado del manejo de los comandos remotos, quedando por mostrar los detalles de implantación de los programas que realizan cada una de las funciones, cosa que también he considerado pertinente no incluir por la misma razón, remitiendo al lector para más detalle sobre la recomendación X.29, a la literatura editada por el CCITT.

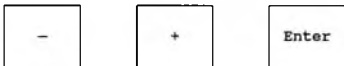
IV.9. Teclado y desplegado

La inclusión de un teclado y un desplegado en nuestro EDD, tiene como principal finalidad la de permitir hacer modificaciones en el EDD o en alguno de los parámetros de terminal de cualquiera de ellas, y además la de hacerlo más atractivo comercialmente.

IV.9.1. Teclado

El teclado se maneja mediante interrupciones; de la rutina de servicio se señala al proceso que maneja el desplegado-teclado, que el usuario ha presionado una tecla, de la misma manera se señala, a manera de una cuarta tecla, la expiración de un temporizador, el cual hace volver al desplegado al modo estado, este temporizador se rearranca cada vez que se presiona una tecla.

El teclado consta de 3 teclas únicamente, que son:



(-). Esta tecla tiene la función de avanzar hacia atrás o de decrementar para hacer alguna selección dependiendo del nivel en que se encuentre el desplegado.

(+). Esta tecla tiene la función de avanzar hacia adelante o de incrementar para hacer alguna selección dependiendo del nivel en que se encuentre el desplegado.

(E). Esta tecla de introducir o ejecutar, permite pasar de un nivel a otro del desplegado o de tomar los cambios o selecciones hechas en un comando.

IV.9.2. Desplegado

El desplegado está formado por una pantalla de 16x1 caracteres alfanuméricos; los diferentes modos en los que puede estar el desplegado se muestran en la figura IV.6.

IV.9.2.1. Modo estado

En este modo se presenta en el desplegado información referente al estado del enlace (si está activo o inactivo) a nivel 2 de X.25, así como las terminales activas (que tienen encendida su señal de DTR); en este modo no existe cursor; en la figura IV.7 se da un ejemplo.



Figura IV.6: Modos del desplegado.

L	I	N	K	U	P	-	-	3	-	-	-	7	-
---	---	---	---	---	---	---	---	---	---	---	---	---	---

Figura IV.7: Modo estado (Enlace activo y terminales 3 y 7 encendidas).

IV.9.2.2. Modo presentación

Este modo permite seleccionar uno de los comandos disponibles, se tienen comandos para:

- Ver/Poner los parámetro de terminal del CCITT.
- Ver/Poner los parámetros extra.
- Seleccionar perfil.
- Salvar los parámetros actuales a memoria no volátil.
- Restaurar los parámetros a su valor inicial.
- Ver el valor de los parámetros de los diferentes perfiles predefinidos disponibles.
- Conmutar el idioma a utilizar para los comandos del desplegado.
- Conmutar la dirección a nivel 2 de X.25 que ha de dar al EDD la identidad de ETD o de ETCD.
- Regresar al modo ESTADO.

En este modo no existe cursor; en la figura IV.8 se da un ejemplo del comando para Ver/Poner los parámetros de terminal del CCITT.

P	O	N	E	R	/	V	E	R		P	A	R	M	S
---	---	---	---	---	---	---	---	---	--	---	---	---	---	---

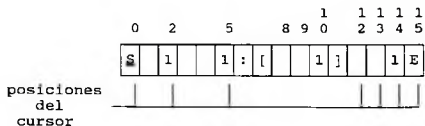
Figura IV.8: Modo presentación (comando para ver/poner los parámetros de terminal del CCITT).

IV.9.2.3. Modo utilización

En este modo es donde efectivamente se hace uso de las facilidades disponibles, en este modo sí hay cursor, y se tendrán dos modalidades: la primera permite moverse a través de los campos que ofrece la pantalla y ubicarse en la parte que se desea modificar, el cursor que se presenta en este caso es () sin parpadeo; la segunda modalidad, a la cual se pasa oprimiendo la tecla (E), permite pasar a modificar el campo seleccionado previamente; aceptar los cambios y volver a la primer modalidad, se logra presionando nuevamente la tecla (E); en esta segunda modalidad el cursor que se presenta es (█) con parpadeo; en la figura IV.9 se presenta como ejemplo el comando para ver/poner los parámetros de terminal del CCITT.

IV.9.3. Automatas para manejo del desplegado

La figura IV.10 muestra el autómata del menú principal (modo presentación) y la figura IV.11 muestra el autómata del comando para Poner/Ver los parámetros de terminal del CCITT; los demás autómatas siguen un esquema semejante.



Posición(es):

- 0 Indica el tipo de comando
- 2 Indica el número de terminal sobre la cual se hacen las modificaciones
- 5 Indica el parámetro actual
- 8 a 10 Indican el valor actual del parámetro
- 12 a 14 Permiten modificar las centenas, decenas y unidades respectivamente
- 15 Actualiza el parámetro en cuestión con el valor de las casillas 12 a 14

Figura IV.9: Modo utilización (comando para ver/poner los parámetros de terminal del CCITT).

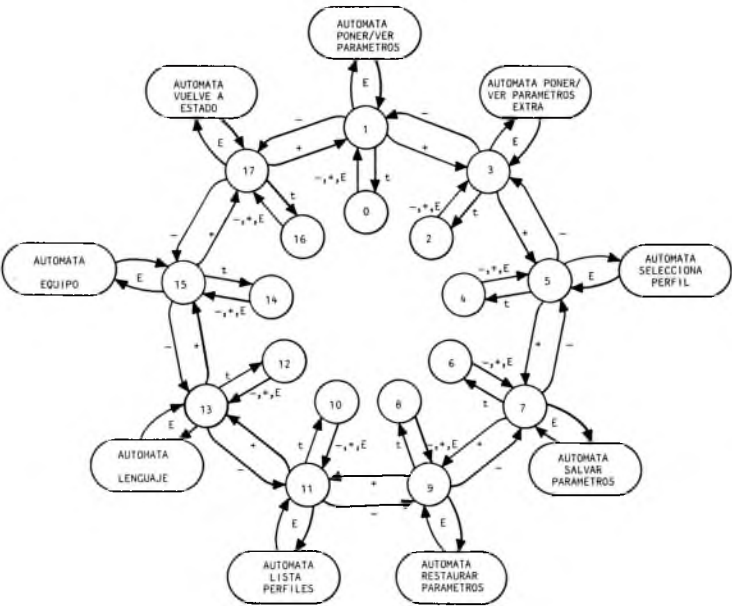


Figura IV.10: Autómata principal del despliegado.

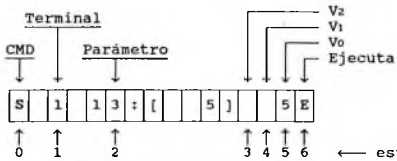
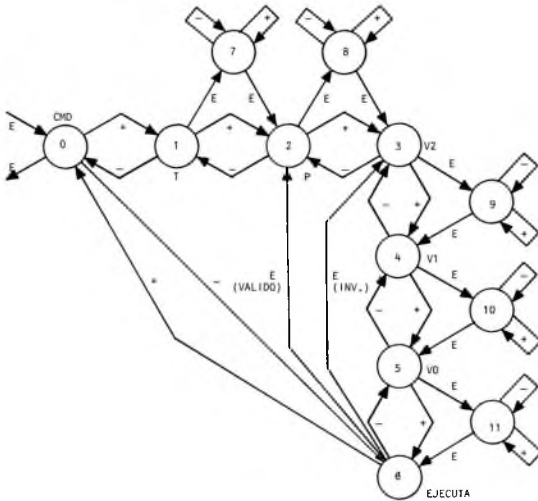


Figura IV.11: Automata del comando para Poner/Ver los parámetros de terminal del CCITT.

V. PRUEBAS Y CONCLUSIONES

Una vez terminada la implantación de las recomendaciones y el manejo del desplegado y teclado, el EDD funcionaba apropiadamente con usuarios humanos, que introducen caracteres a una velocidad relativamente baja (el EDD ha sido concebido para manejar terminales de usuario); al hacer pruebas con dispositivos más veloces (como conectar una computadora que envía un archivo) para enviar caracteres a los puertos del EDD, se observó pérdida de los mismos, las pérdidas a una velocidad de 9,600 bps eran del orden de un 16 % en 36,400 caracteres, este problema hizo plantear soluciones de hardware y de software.

Las soluciones de hardware fueron incrementar la velocidad del procesador en un 60 %, cosa que se esperaba no resolvería el problema por completo, y que se tendría que hacer el resto por software.

Durante el proceso de subir la frecuencia de operación del procesador 80C88-2 de 4.9 MHz a 8 MHz, se detectó un funcionamiento anormal en el procesador de comunicaciones, el circuito WD2511, este problema constituyó un verdadero contratiempo, ya que por la falta de información más especializada sobre el procesador de comunicaciones, algunas interrogantes sobre su exacto funcionamiento en algunos casos no pudieron responderse, limitándonos (el Ing. Rodolfo Rosado y yo) a especular sobre su posible solución, la cual logramos finalmente pero sin encontrar una justificación satisfactoria a la anomalía detectada. Todo lo anterior constituyó un retraso no solo en cuanto al hardware mismo, si no también porque significó involucrarme con cierto detalle en la implantación del nivel de enlace con el objeto de diseñar nuevas pruebas y hacer cierto diagnóstico con las variables que se manejaban a ese nivel.

Además del problema con el procesador de comunicaciones se hizo un replanteamiento de las prioridades que deberían tener las interrupciones por hardware; al hacer el cambio nos sorprendimos al ver cuan significativamente se había mejorado el problema de pérdida de caracteres (del orden de un 25 % en relación al esquema anterior original).

Por el lado del software se observarían dos cosas:

- 1) Velocidad en general del software.
- 2) Disminución de las regiones críticas.

Para el punto 1 se construyó parte del núcleo para concurrencia en lenguaje ensamblador, así como algunas funciones para el manejo de buffers y del módulo manejador de terminal, obteniéndose mejoras del orden de un 5 % a un 50 % en cada función, estas mejoras se evaluaron por el número de instrucciones y por el número de accesos a memoria, sin tomar en cuenta que en algunos casos usé instrucciones más rápidas según fuera el caso (la rapidez ganada en esto último era del orden de unos cuantos ciclos de reloj en la ejecución de la instrucción).

Al utilizar el lenguaje ensamblador para construir ciertas funciones, me encontré con otro problema, un problema sobre una instrucción para manejo de apuntadores, la cual no se comportaba de acuerdo a como yo lo había interpretado de la especificación que hacía el manual de Intel, del microprocesador 8086/88, esto constituyó un obstáculo que me robó bastante tiempo, ya que en un principio estuve seguro de que el problema debería de estar en un error por mi cometido, y lo busqué en la utilización de los registros y en la misma lógica de mi programa, hasta que me di cuenta de esa característica de la instrucción por mi ignorada y de su funcionamiento exacto.

Respecto al punto 2, se han reducido significativamente las regiones críticas mediante un replanteamiento de las mismas y la inclusión de procesos servidores del tic de reloj y del procesador de comunicaciones.

Con las modificaciones del software hasta aquí mencionadas (utilizando la versión del procesador a 4.9 MHz), se ha logrado reducir la pérdida de caracteres a un 3 %; con fundamento puedo pronosticar su completa solución mediante la inclusión de nuevos procesos servidores para recepción y transmisión de caracteres, así como un completo replanteamiento del esquema de entrada de caracteres, esta labor ya la estoy realizando pero no la incluiré como parte de esta tesis.

Una gran satisfacción me ha dejado el trabajar en este proyecto, pues he logrado resolver problemas que en varias ocasiones me parecían muy difíciles y que en algunos casos llegué a pensar que tal vez fueran inherentes al diseño en general

y que por lo tanto no podría salvarlos, sin embargo, he visto pasar diferentes obstáculos ingenieriles seguidos, en mayor o menor tiempo, de su respectiva solución, lo cual reitero, me ha causado gran satisfacción.

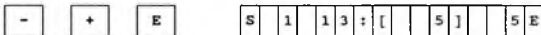
Si tuviera que hacer un nuevo diseño del EDD seguramente elegiría un procesador más veloz (80186 u 80286) y con bus de datos de 16 bits para disminuir el tiempo requerido al acceder la pila, ya sea para los cambios de contexto o durante el llamado de funciones y paso de parámetros, pediría un desplegado que no impusiera restricciones para su acceso, así como las tiene también la memoria no volátil (EEPROM), buscaría interfaces de comunicación serie más versátiles, que por ejemplo incluyeran un pequeño buffer de caracteres, y en cuanto a los programas haría una nueva especificación de muchas de las funciones mediante autómatas dirigidos por tabla o bien con variable de estado, ya que en varios casos hay software que podría construirse con esta técnica, y que sin embargo no lo está.

Resumiendo las características no normalizadas por el CCITT, nuestro EDD cuenta con:

- i) Mayor versatilidad en las señales de instrucción, por ejemplo:
 - * par? 1,2, 5 7, <CR>
- ii) Nuevos comandos y sus respectivas señales de servicio:
 - SAVE, RESTORE
 - XPAR?, XSET, XSET?
 - ? Y HELP
 - VER
- iii) Nuevos parámetros para:
 - Selección de minúsculas y mayúsculas
 - Selección del tipo de paridad
 - Selección del perfil inicial

- Selección del idioma en las señales de servicio
 - Selección del uso de la señal RTS
- Los cuales actuan independientemente en cada terminal.

- iv) Se tienen 20 perfiles predefinidos.
- v) La disponibilidad de un perfil totalmente definido por el usuario.
- vi) Disponibilidad de un teclado y de un desplegado con funciones para:



- Ver o modificar los parámetros de EDD definidos por el CCITT
 - Ver o modificar los parámetros de EDD extra
 - Seleccionar un perfil
 - Salvar los parámetros de EDD
 - Restaurar los parámetros de EDD
 - Ver los valores de los perfiles predefinidos
 - Seleccionar el idioma de los mensajes que aparecen en el desplegado
 - Seleccionar la identidad del EDD como ETD o ETC
 - Ver el estado del enlace y las terminales conectadas
- vii) La selección de la velocidad de la terminal está disponible a través del parámetro 11

El total de líneas de código programadas (incluyendo las líneas de comentarios) es de aproximadamente:

Núcleo para concurrencia	-	3,500 líneas en PL/M-86
	-	900 líneas en ASM-86
Primer fase	-	4,000 líneas en PL/M-86
Segunda fase	-	3,300 líneas en PL/M-86
	-	600 líneas en ASM-86
<hr/>		
Totales:		10,800 líneas en PL/M-86
		1,500 líneas en ASM-86

Han trabajado también en este proyecto:

Dr. Manuel E. Guzmán Rentería	(Dirección)
M. en C. César A. Galindo Legaria	(Primer fase-software)
Ing. Rodolfo Rosado Martínez	(Hardware)
M. en C. Ivan Pech Escalante	
M. en C. Manuel Moreno Higuera	

BIBLIOGRAFIA

CCITT

Libro Rojo, Tomo VIII - Fasciculo VIII.3
REDES DE COMUNICACION DE DATOS
SERVICIOS Y FACILIDADES
Recomendaciones X.20 a X.32
CCITT, 1984

CCITT

Libro Rojo, Tomo VIII - Fasciculo VIII.2
REDES DE COMUNICACION DE DATOS
SERVICIOS Y FACILIDADES
Recomendaciones X.1 a X.15
CCITT, 1984

WESTERN DIGITAL CORPORATION

WD2511 X.25 Packet Network Interface (LAPB)

Intel

Microsystem Components Handbook Vol. I & II
Intel Corporation, 1985

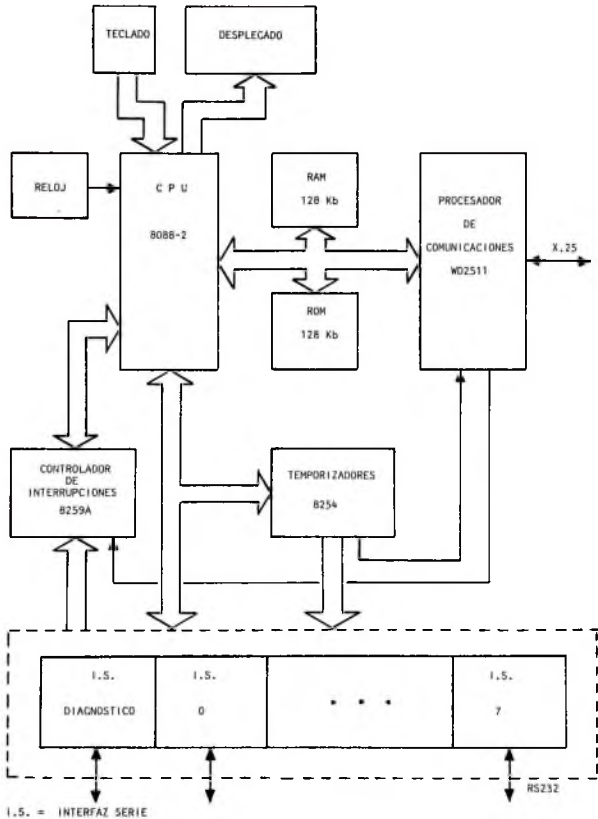
Marshall Abrams - Ira W. Cotton

COMPUTER NETWORKS:

A Tutorial

IEEE Computer Society Press, 1984

Apéndice A. Esquema general del hardware



Apéndice B. Términos

ASCII.- Código Estándar Americano para Intercambio de Información (American Standard Code for Information Interchange).

CCITT.- Comité Consultivo Internacional de Telefonía y Telegrafía.

CIRD.- Código de Identificación de Red de Datos

CTS.- Señal de control de modem "listo para envío" (Clear To Send).

DSR.- Señal de control de modem "modem encendido" (Data Set Ready).

DTR.- Señal de control de modem "terminal encendida" (Data Terminal Ready).

EDD.- Ensamblador/desensamblador de datos (PAD - Packet Assembler/Disassembler)

EPER.- Empresa Privada de Explotación Reconocida.

HOST.- Computador donde se corren programas de usuario.

ISO.- Organización Internacional de Normalización (International Standards Organization).

OSI.- Modelo de referencia de 7 niveles para la interconexión de sistemas abiertos (Open System Interconnection reference model).

RTS.- Señal de control de modem "petición de envío" (Request To Send).

El jurado designado por la Sección de Computación del Departamento de Ingeniería Eléctrica del Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, aprobó esta tesis el 1 de Diciembre de 1989.

Dr. Jan Janeček Hyan



A handwritten signature in black ink, appearing to be 'J. Janeček', written over a horizontal line.

Dr. Manuel E. Guzmán Rentería



A handwritten signature in black ink, appearing to be 'Manuel E. Guzmán', written over a horizontal line.

M. en C. Miguel Angel
Fernández Sánchez



A handwritten signature in black ink, appearing to be 'Miguel Angel Fernández', written over a horizontal line.

M. en C. Jaime Ulises
Barbosa Valencia



A handwritten signature in black ink, appearing to be 'Jaime Ulises Barbosa', written over a horizontal line.

