

DISEÑO DE LA INTERFASE PARA LA RED LOCAL UNIANDES

Leonardo Araújo
Jairo Fernández
Roberto Pardo

Departamento de Ingeniería de Sistemas
Universidad de Los Andes
Apartado Aéreo 4976, Bogotá - Colombia.

OBJETIVO

La tecnología de hardware (1) y software para interconexión de computadores (redes) ha avanzado a pasos gigantescos en la última década. Muchos adelantos aplican a redes de amplia cobertura geográfica, donde los nodos son computadores relativamente grandes y costosos, donde los nodos están dispersos por ejemplo en un país, y donde las decisiones de diseño de hardware y software de comunicación tienden a minimizar un costo inherentemente alto de comunicación.

En los últimos 5 años, con la proliferación de máquinas medianas y pequeñas, se ha despertado un gran interés por la interconexión de computadores localizados en áreas pequeñas (un edificio, un sector de la ciudad). En este tipo de redes locales las decisiones de diseño e implantación tanto del hardware como del software de comunicación son un tanto diferentes. Por ejemplo, en vez de usar conmutadores y/o procesadores de comunicación los cuales son costosos por cuanto el mismo sistema y la comunicación es costosa, se usan Interfases sencillas y poco costosas.

En este artículo se presentan las principales decisiones de diseño de una Interfase que se usará para la interconexión local de varios computadores de la Universidad de Los Andes, Bogotá. Adicionalmente se explicará por qué es interesante realizar este tipo de proyectos en países como los nuestros.

TERMINOS CLAVES : Interfase, Sistemas Distribuidos, Protocolo, Red Local.

(1) Se usarán algunos términos del inglés como "hardware", "software", "buffer" etc., cuando se considere que no tienen una traducción ampliamente aceptada.

1. INTRODUCCION

Este artículo describe el diseño de la Interfase que une los diferentes nodos que conforman la "Red local Uniandes". Una red local, como su nombre lo indica, es una red de comunicación de datos, que cubre un área geográfica pequeña (un edificio o un sector de la ciudad). Este tipo de redes generalmente provee un ancho de banda alto sobre un medio de transmisión poco costoso. (En (CLAR78) se habla en más detalle sobre redes locales).

La red local Uniandes es una red experimental, actualmente en fase de diseño, que nació como un esfuerzo conjunto entre los Departamentos de Ingeniería Eléctrica y de Sistemas de la Universidad de Los Andes. Los objetivos principales del proyecto son :

1. Experimentar en el diseño e implantación de tecnología de hardware y software de redes.
2. Crear una infraestructura de bajo costo para el desarrollo experimental de software distribuido.

El esquema de la red es el de un conjunto de nodos heterogéneos (computadores grandes, medianos y pequeños existentes en la Universidad) que a través de las Interfases se interconectan al compartir un medio de transmisión común (topología tipo bus).

La interfase es un dispositivo (un microcomputador) que continuamente está dispuesto a recibir "paquetes" de los nodos y a ponerlos a viajar por el medio de transmisión compartido a su(s) destino(s). Igualmente la Interfase está continuamente dispuesta a recibir paquetes del medio para dirigirlos al nodo, cuando sea necesario.

Este artículo comienza presentando en la sección 2 una visión general de la Interfase. Se ubica ésta dentro de la jerarquía de protocolos de comunicación y se recalca su importancia. La sección 3 describe con algún detalle los requerimientos mínimos de hardware para una Interfase realista (cuyo hardware esté al alcance de nuestro medio) en el caso de la red Uniandes. La sección 4 presenta el software diseñado para el manejo de la Interfase. Más detalles de este trabajo se encuentran en (ARAU80).

2. CONSIDERACIONES GENERALES

Cuando se desea interconectar dos máquinas normalmente es posible encontrar una solución fácil y realizable a corto plazo : se trata de usar lo ofrecido por las diferentes casas de los fabricantes. Si los equipos son homogéneos es muy sencillo lograr que cada uno "vea" al otro como si fuera una terminal. Si los equipos son heterogéneos es posible estandarizar la comunicación mediante el uso de emuladores como el 3780 de IBM (ofrecido por Texas, Burroughs, WANG, etc) ó 3270 de IBM. Obviamente este tipo de soluciones es aceptable cuando los requerimientos de comunicación son muy elementales. Ahora bien, si se desea conectar una nueva máquina, el problema es aún relativamente fácil de resolver porque bastaría colocar dos líneas más para comunicar el tercer computador con los existen

tes. Conectar un cuarto o más computadores implica pensar adicionalmente en cuáles de ellos se interconectarían con estos, ya que por un lado existe en cada máquina un límite físico de puertos de comunicación y por otro, conectarlos todos entre sí incrementa el costo de canales de comunicación (incluyendo el de modems si es el caso). Adicionalmente es necesario comenzar a introducir software confiable de enrutamiento y evitar que la disponibilidad del sistema dependa de la disponibilidad de unas pocas máquinas. Si seguimos expandiendo la red, estos problemas se tornan aún más complejos (y típicamente en redes de amplia cobertura geográfica se resuelven con hardware y software de comunicación costoso).

En un ambiente de red local las topologías de canal común (v.g. bus o anillo) evitan que sucedan los problemas anteriormente mencionados. La red Uniandes usa un sistema de comunicación local soportado por una topología de tipo bus. Específicamente esta concepción de red ofrece varias ventajas :

- Se crea un ambiente fácil de expandir, porque es sencillo añadir nuevos nodos sin preocuparse por adquirir nuevas líneas, gastar puertos de comunicación y enrutar.
- Se presta esta topología para desarrollar un sistema confiable en el sentido de que la caída de un nodo o Interfase no afecta el funcionamiento del resto de la red.
- Permite tener un protocolo de acceso (método usado para compartir ordenadamente el medio de transmisión) sencillo y distribuido.

2.1. CARACTERISTICAS DE LA INTERFASE

El hardware de comunicación de una red local tiene como objetivo dar un alto rendimiento a bajo costo. Como en este ambiente típicamente se desean interconectar minicomputadores y microcomputadores de bajo costo es necesario hacer tan barato como sea posible el hardware de la Interfase. Es claro que sería más eficiente realizar físicamente la Interfase como un minicomputador de comunicaciones ("front-end") pero este tipo de solución haría que fácilmente el costo de la Interfase fuera superior al costo de algunos nodos utilizados. Es esta la principal razón por la cual se implantan las Interfases como microcomputadores.

En la red Uniandes cada Interfase asociada con una máquina provee el control de la transmisión y recepción de paquetes de bits. Tal función involucra el chequeo de errores causados por problemas en el medio de transmisión y que inciden sobre los paquetes que por él viajan. Adicionalmente la Interfase reconoce direcciones con el fin de determinar cuál es el destino de los paquetes de bits que le llegan (todos los paquetes llegan a cada Interfase). Existe, también, un mecanismo encargado de detectar y resolver las colisiones que pueden ocurrir durante la transmisión de los paquetes.

2.2. LOCALIZACION DE LA INTERFASE

La figura 1 muestra en forma general el diagrama de la red Uniandes, y en ella se aprecia la ubicación física de las Interfases, las cuales están conceptualmente divididas en dos : una parte orientada a la línea (comple

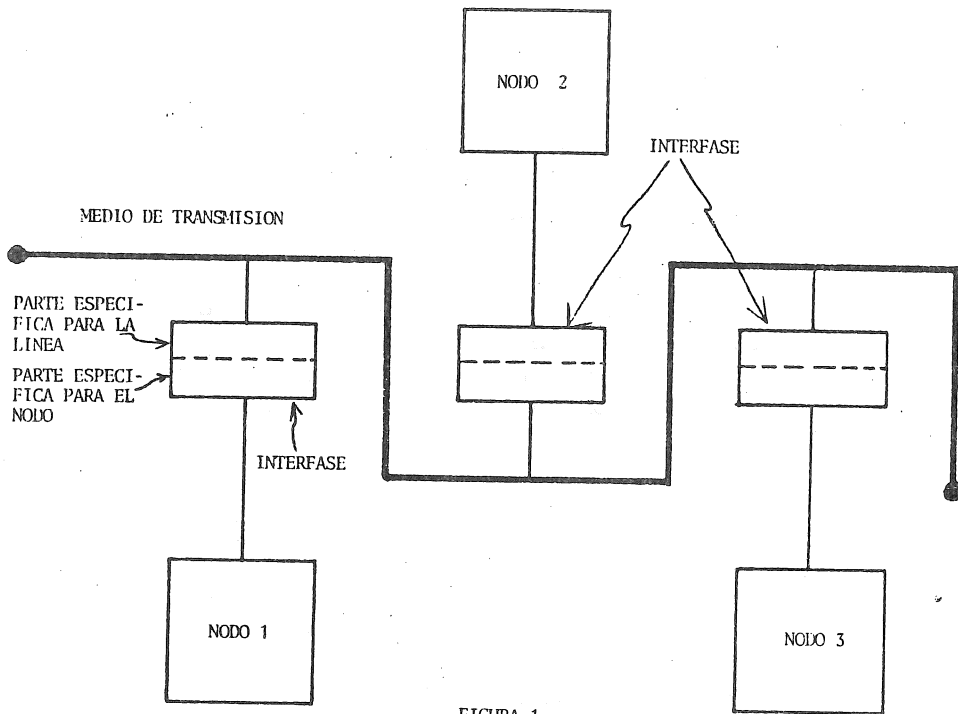


FIGURA 1
ESQUEMA GENERAL

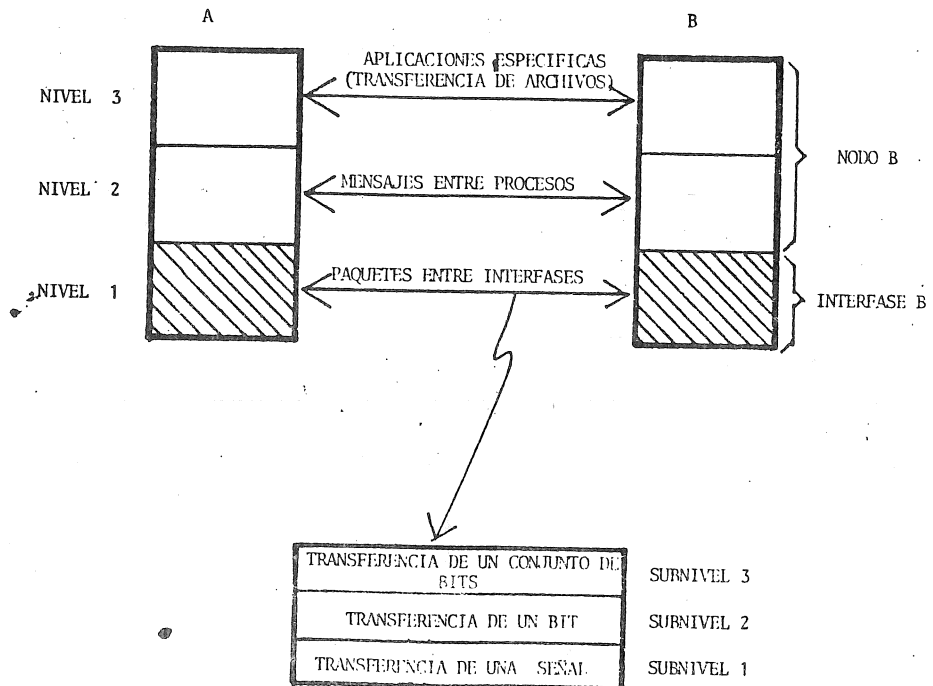


FIGURA 2
LOCALIZACION

tamente estándar), que realiza todas las funciones de control que son requeridas por ésta, y una parte orientada al nodo (varía en detalle según el nodo), que controla el intercambio de paquetes entre el nodo y la parte orientada a la línea.

Dentro de la jerarquía de protocolos de comunicación (ver figura 2) el intercambio de mensajes entre las Interfases pertenece al nivel más bajo (nivel 1 en la figura 2). Sobre éste se encuentran dos niveles más : el de mensajes entre procesos y el de aplicaciones específicas (v.g. transferencia de archivos (GAIT80), sistemas de archivos distribuidos (SANC80)).

Con esta concepción de diseño es posible hacer modular la comunicación nodo a nodo, de tal manera que cambios realizados en un nivel no alteren los demás, siempre y cuando se preserven las mismas Interfases. El nivel 1 se divide, a su vez, en tres subniveles : 1) transferencia de señales, en donde se resuelve el problema de reconocimiento de señales desde el punto de vista eléctrico, 2) transferencia de bits, en donde se resuelve el problema de reconocimiento de bits (las señales del subnivel anterior codifican bits), y 3) transferencia de conjuntos de bits (paquetes). Este artículo se refiere básicamente a los problemas de los subniveles dos y tres (una buena discusión sobre estructura y niveles de protocolos se encuentra en (PARD79), (BOGG80)).

2.3. IMPORTANCIA DE LA INTERFASE

Vale la pena recalcar la importancia de incluir una Interfase en la red : 1) Se estandariza la comunicación entre los diferentes nodos colgados a la red (el hecho de tener una Interfase estandar facilita la expansión de la red, puesto que "colgarle" un nuevo nodo no trae mayores complicaciones), y 2) se libera, hasta cierto punto, al procesador de los trabajos relacionados con comunicación (teniendo en cuenta las capacidades de una Interfase poco costosa).

2.4. TRABAJOS RELACIONADOS

Una de las principales motivaciones en el proyecto de nuestra red es "embarcarnos" en el desarrollo de una tecnología reciente y sobre la cual se está trabajando en muchos sitios. Creemos que hay varias ventajas en este ejercicio : 1) cuando esta tecnología llegue en forma de "producto" a nuestro país podremos evaluarlas al nivel que queramos su calidad técnica (algo que raras veces sucede en tecnología de computadores ya que invertimos esfuerzos en sólo entender lo que nos venden, y peor aún, a veces sólo después de que nos lo han vendido), 2) aunque existan redes "mejores" la experiencia en el diseño e implantación de estos sistemas es invaluable.

La red pionera en sistemas de comunicación local es DCS (FARB72), desarrollada en la Universidad de California, Irvine, la cual interconecta varios nodos heterogéneos en forma de anillo. Sin embargo los adelantos más interesantes en redes locales han sucedido en los últimos 5 años. ETHERNET (METC80, CRAN 80, THAC79), desarrollada en Xerox Palo Alto Research Center, es tal vez la red local más popular hoy en día (aparentemente será un producto conjunto de XEROX/DIGITAL/INTEL) que

interconecta mediante un bus nodos homogéneos. La Interfase es parte del nodo mismo e implanta un protocolo de contención. Otra red con igual topología y protocolo es NBSNET (CARP80); sin embargo la Interfase de ésta es un microcomputador un tanto sofisticado. Otro producto existente en el mercado desde hace varios años es el HYPERCHANNEL (DONN79) el cual interconecta componentes, generalmente de la compañía CDC, a través de un bus usando un protocolo de contención. Prime Computers, ofrece RINGNET (GORD80) como un producto el cual usa topología tipo anillo y un protocolo que circula un mensaje especial ("token") para controlar el acceso al anillo. Zilog anunció una red similar al ETHERNET llamada Z-NET. Finalmente existen compañías no-fabricantes de computadores que venden o cables, o Interfases muy elementales, o ambos.

Muchas universidades y centros de investigación han desarrollado o están desarrollando redes locales. Interesantes entre éstas, el proyecto del MIT de su LCSNET (CLAR78, SALT80) y el de Ohio State con su DDLCN.

3. HARDWARE DE LA INTERFASE

Recordemos que la tarea principal de la Interfase consiste en recibir paquetes de bits del nodo para enviarlos a su destino vía la línea de transmisión, además de transportar los que por ella vienen con destino al nodo. Para realizar esta labor, es necesario comunicar a la Interfase con la línea y con el nodo. En la figura 3 se muestra el hardware mínimo que necesita la Interfase para cumplir con sus requisitos de funcionamiento :

- a) una unidad de microprocesamiento con, por lo menos, cinco puertos de entrada y salida para conectar la Interfase con el nodo, conectar la Interfase con la línea y conectar dos relojes externos (más adelante se explica su uso); además es deseable que el microprocesador posea dos niveles de interrupción (uno enmascarable y otro no enmascarable); b) memoria permanente y temporal, y c) una Interfase para comunicación con el nodo y otra para comunicación con la línea.

3.1. IDENTIFICACION Y SELECCION DE COMPONENTES

a) MICROPROCESADOR : se escogió el Motorola 6800 (MOTO76) por su disponibilidad y fácil uso (lo relevante del artículo no es mostrar una selección óptima de componentes). Este microprocesador tiene las siguientes características :

- bus de datos de 8 bits (bidireccional),
- capacidad de direccionamiento hasta 64 K bytes (bus de direcciones de 16 bits),
- 72 instrucciones,
- 7 modos de direccionamiento,
- interrupción enmascarable,
- interrupción no enmascarable,
- 6 registros internos, y
- periféricos que se referencian como posiciones de memoria.

b) MEMORIA PERMANENTE : se escogió la de tipo PROM (Programmable Read Only Memory) para guardar los programas que manejan la Interfase

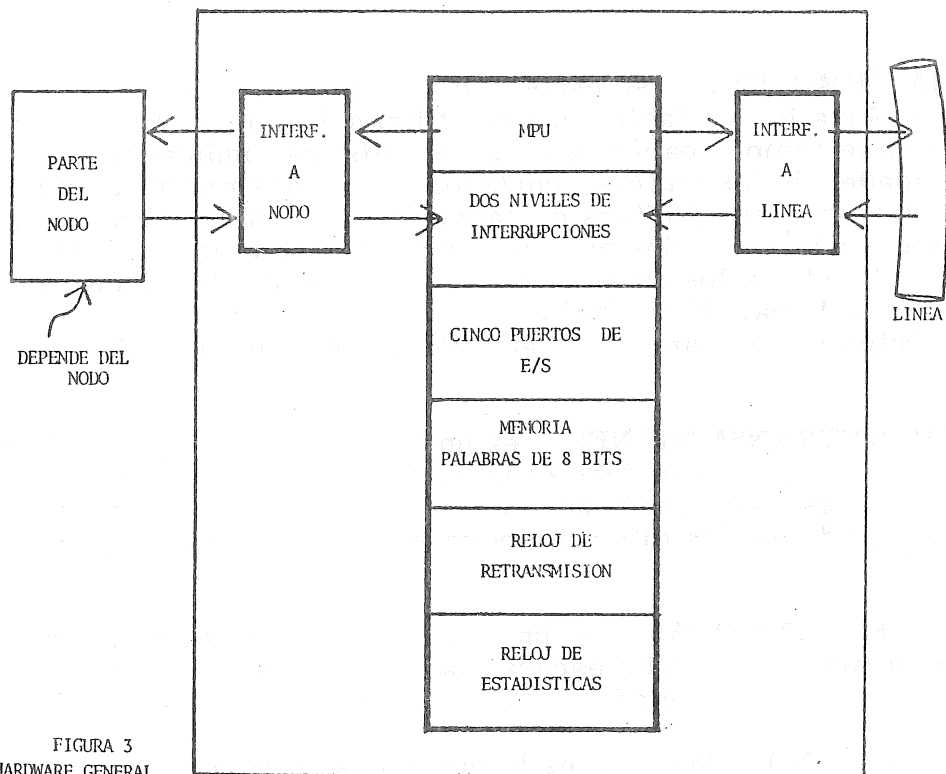


FIGURA 3
HARDWARE GENERAL

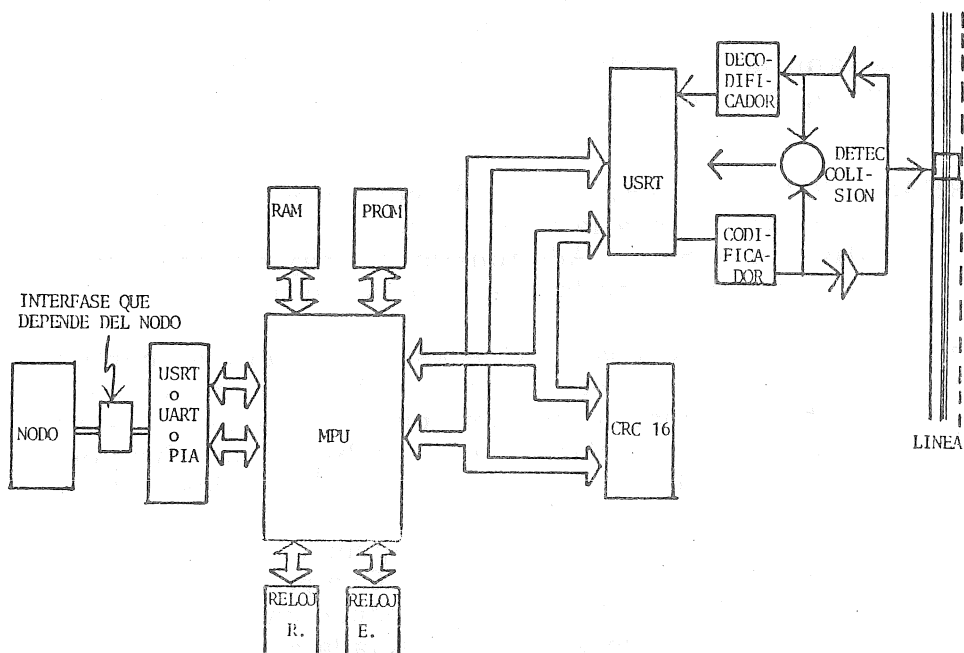


FIGURA 4
INTERCONEXION DE HARDWARE

algunos datos fijos (dirección del nodo y la dirección del nodo de estadísticas), y los apuntadores que proveen las direcciones de los programas que deben ser ejecutados ante la ocurrencia de una interrupción determinada.

c) MEMORIA TEMPORAL : se escogió la de tipo RAM (Random Access Memory). Soporta las variables que se utilizan durante el funcionamiento de la Interfase como : parámetros de estadística, indicadores del estado y prioridades de los buffers, contador de retransmisiones, y los buffers. Se tienen cuatro buffers de capacidad máxima de almacenamiento de 256 bytes cada uno : dos para los paquetes recibidos del nodo y transmitidos al nodo, y los otros para los paquetes recibidos del nodo y transmitidos a la línea. Estos buffers se usan en forma FIFO (First Input First Output) y se manejan independientemente para la línea y para el nodo.

d) RELOJ DE RETRANSMISIONES : es un reloj externo que se utiliza para contabilizar el tiempo que debe esperar la Interfase para retransmitir un mensaje después de ocurrida una colisión en la línea. Cada Interfase tendrá un tiempo de espera diferente porque, de no ser así, ocurriría de nuevo una colisión.

e) RELOJ DE ESTADISTICAS : es un reloj externo que se utiliza para contabilizar el tiempo que debe esperar la Interfase para ensamblar y enviar las estadísticas recolectadas.

f) INTERFASE CON EL NODO : es lo que necesita el microcomputador para comunicarse con el nodo y depende de las facilidades que provea el nodo. Por lo general, la comunicación ofrecida es en serie; si es asincrónica se utiliza un chip del tipo UART (Universal Asynchronous Receiver/Transmitter), o si es sincrónica se utiliza uno de tipo USRT (Universal Synchronous Receiver/Transmitter). En caso de que la comunicación sea en paralelo, la familia Motorola 6800 provee un chip llamado PIA (Peripheral Interface Adapter), el cual se encarga de la recepción y transmisión en paralelo.

g) INTERFASE CON LA LINEA : para la comunicación con la línea se necesita un chip que transfiera la información serial y sincrónicamente. Para ello se utiliza un USRT. Adicionalmente hay que identificar otros componentes que hacen parte de la Interfase con la línea :

- CHIP CRC-16 (Cyclic Redundancy Checksum): es el más confiable de todos los tipos de chequeadores de error en la transmisión de los paquetes. Agrega 16 bits al final del paquete cuando éste se transmite. Al recibirse el paquete, se calculan 16 bits, en base a una función polinomial y se comparan con los 16 bits de chequeo recibidos (WEIS79).
- DETECTOR DE COLISIONES : es una compuerta Or-Exclusivo que realiza la función de detección de colisiones. Compara los datos recibidos con los que están siendo transmitidos y produce una señal de colisión si hay alguna diferencia (esta señal está conectada a la entrada de la interrupción no enmascarable del microprocesador, dada su alta prioridad). Una colisión se produce cuando dos o más Interfases hacen acceso a la

línea al mismo tiempo (CRAN80).

- CODIFICADOR Y DECODIFICADOR DE SEÑALES : dependiendo del medio de transmisión que se utilice, se varía la clase de estos componentes para hacer compatibles los de señales que éste entiende con las que entiende la Interfase (CRAN80).

La figura 4 ilustra la interconexión de los componentes de hardware de la Interfase.

4. DISEÑO DE SOFTWARE

4.1. ESPECIFICACION

La descripción del software juega un papel muy importante en todos los pasos de su desarrollo. En esta sección se intenta explicar claramente y sin ambigüedades (i.e. mediante métodos formales), las funciones que debe desempeñar el software de la Interfase de la red.

La arquitectura de comunicación de un sistema distribuido, para su mejor entendimiento y para lograr buena modularidad, debe estar estructurada como una jerarquía de diferentes niveles. Cada nivel, provee un conjunto particular de servicios a sus usuarios superiores. Desde este punto de vista, el nivel puede verse como una caja negra o máquina que permite un cierto conjunto de interacciones con otros usuarios. A cada usuario debe interesarle la naturaleza del servicio provisto, pero no la forma en que el módulo lo implementa.

Esta descripción del comportamiento de Entrada/Salida de cada nivel es lo que constituye una especificación del servicio prestado por éste.

De la misma forma que es necesario establecer una estructura jerárquica entre los diferentes niveles del software de comunicación, dentro de cada nivel resulta muy útil establecer módulos claramente diferenciados, y especificarlos de acuerdo a las convenciones anteriormente descritas con el fin de obtener una visión completa de las funciones que estos desempeñarán.

El método que será usado en este artículo para especificar formalmente el software serán los "diagramas de estados" (DANT80), por medio de los cuales es posible controlar y entender todos los posibles eventos concurrentes que pueden suceder en un momento dado dentro de la Interfase.

SECUENCIA GENERAL DE EVENTOS.

Existen claramente definidas tres funciones generales que desempeña la Interfase :

- Servir de intermediario para el transporte de paquetes del nodo hacia la línea.
- Servir de intermediario para el envío de paquetes de la línea hacia el no do.

- Ensamblar paquetes de estadísticas y enviarlos con destino al nodo en cargado de su recolección.

La figura 5 muestra la secuencia general de eventos cuando la Interfase se encuentra transportando un paquete desde el nodo hacia la línea. Inicialmente la Interfase está ociosa (no tiene ningún trabajo por realizar) en espera de algún pedido que deba ser atendido. Estos pedidos se efectúan mediante interrupciones. Si la interrupción es causada por un paquete que viene del nodo, la interfase debe disponerse a recibirlo. Una vez se ha detectado su final, es necesario transmitirlo a su destino vía la línea de transmisión; es este momento es cuando se da el control al protocolo, que se encargará de efectuar confiablemente dicha transmisión. De este nuevo estado (protocolo) se sale cuando se detecta el final del paquete que está siendo transmitido, quedando la Interfase nuevamente ociosa.

La segunda función, ilustrada en la figura 6, se inicia cuando llega un pedido (interrupción) por parte de la línea, indicando que hay un paquete que debe ser recibido y que posiblemente se dirige hacia el nodo asociado con esa Interfase. Si este es el caso, es necesario recibir completamente el paquete entrante y analizar su dirección; cuando ésta concuerde con la asignada a tal nodo, el paquete es pasado a él y la Interfase se contrará ociosa. Si la dirección no concuerda, se pasa directamente al estado ocioso en espera de un nuevo pedido.

Finalmente, la función (figura 7) de la Interfase se relaciona con el ensamblaje y envío de las estadísticas que han sido recolectadas durante la ejecución de las demás funciones. Se parte del estado ocioso; cuando llega una interrupción del reloj de estadísticas, es necesario comenzar a ensamblarlas. Una vez se ha terminado, deben ser enviadas al nodo o a la línea, de acuerdo a la dirección del nodo encargado de recolectarlas.

Los diagramas de estados propuestos en las figuras 5, 6 y 7 describen el software que debe ser implantado en la Interfase, pero hacen indivisibles cada una de las funciones anteriormente explicadas. Sin embargo, el hecho de recibir un paquete del nodo o de la línea no debe implicar que sea enviado inmediatamente a su destino, ya que podría dejar de recibirse otros que vinieran seguidamente, es decir, debe haber suficiente independencia entre la recepción y la transmisión de un paquete, con el fin de poder atender recepciones consecutivas de éstos, sin dañar lo ya recibido (sólo retrasando su transmisión).

Para poder diseñar, entonces, un software que tenga capacidad de atender eventos concurrentes, las tres funciones anteriores serán implantadas por los módulos :

- Recepción de la línea;
- Transmisión al nodo;
- Recepción del nodo;
- Protocolo;
- Ensamblaje de estadísticas.

Estos módulos no serán interrumpibles; es decir, una vez que se han comenzado a ejecutar, terminarán su función a menos que una falla los

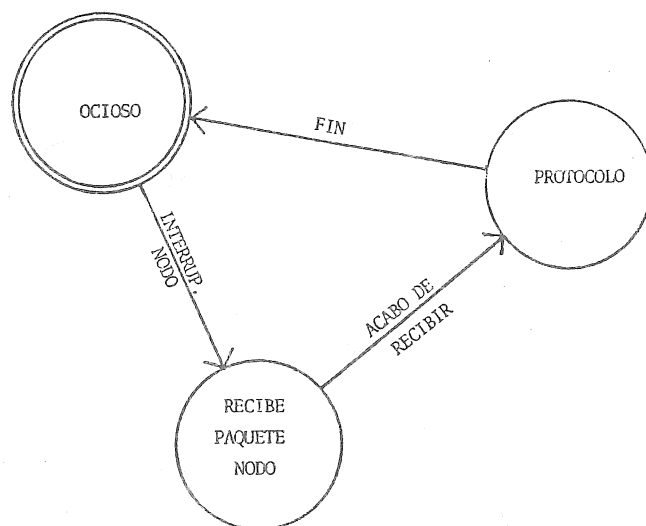


FIGURA 5
TRANSMISION A LINEA

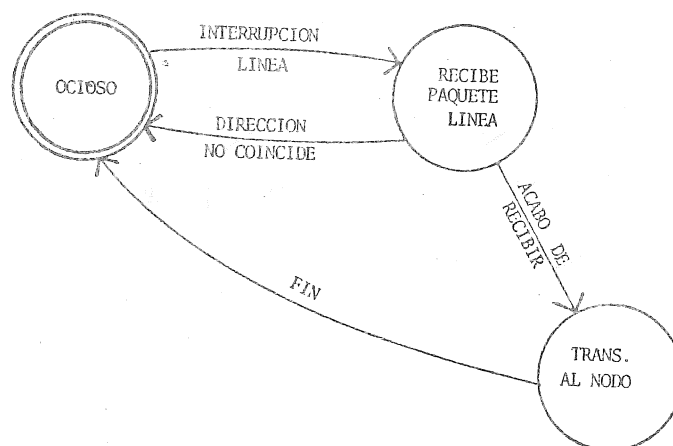


FIGURA 6
RECEPCION DE LINEA

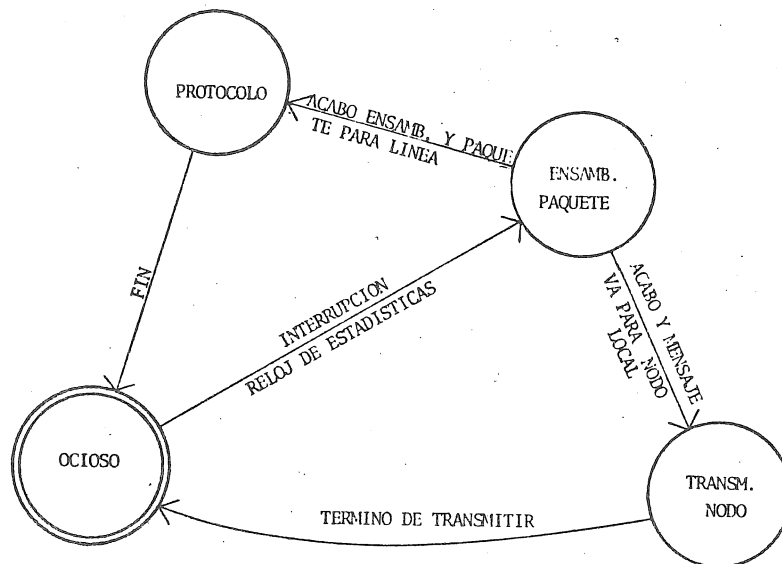


FIGURA 7
ENVIO DE ESTADISTICAS

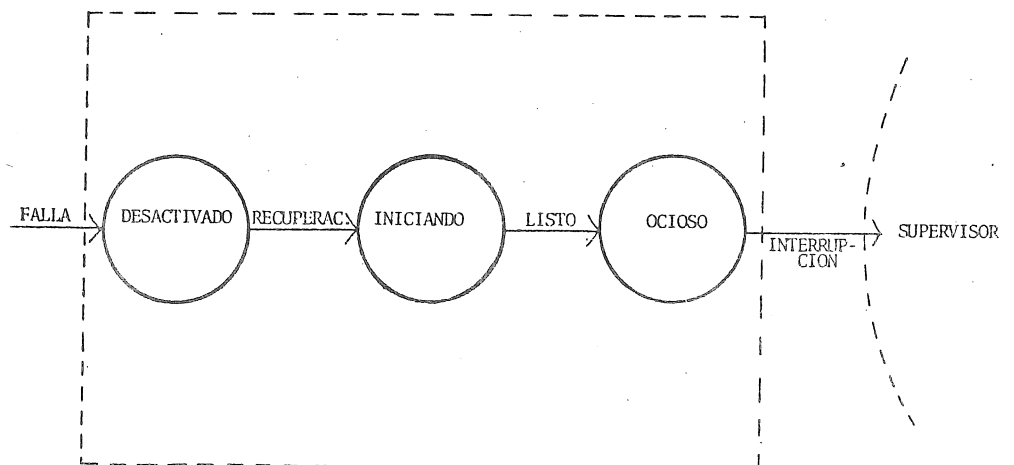


FIGURA 8
MODULO DE INICIACION

suspenda (toda interrupción que se produzca durante su ejecución, será diferida). En resumen, estos módulos con uno adicional de iniciación han lo siguiente :

- MODULO DE INICIACION : Este módulo tiene a su cargo iniciar la Interfase cuando se recupera después de una falla, o simplemente cuando es puesta en funcionamiento.
- MODULO DE RECEPCION DE LA LINEA : Se encarga de resolver los problemas relacionados con la recepción de los paquetes de bits provenientes de la línea (chequeo de errores en la información recibida, control de dirección).
- MODULO DE TRANSMISION AL NODO : Controla la transmisión, hacia el nodo, de los paquetes recibidos correctamente por el módulo de re-cepción de la línea. Este control varía en algunos detalles dependiendo del tipo del nodo al cual está atada la Interfase.
- MODULO DE RECEPCION DEL NODO : Recibe los paquetes provenientes del nodo. Al igual que el módulo anterior éste depende de las facilidades que presente el nodo.
- MODULO PROTOCOLO : Controla la transmisión, hacia la línea, de los paquetes recibidos por el módulo de recepción del nodo. Este control consiste en lograr un acceso ordenado, eficiente y confiable al medio de transmisión compartido.
- MODULO DE ESTADISTICAS : Ensambla las estadísticas recolectadas durante un intervalo predeterminado de tiempo, para luego ser enviadas al nodo designado para que las procese y reporte. Este módulo provee información para medir la eficiencia de la red y es opcional; es decir, existe la posibilidad de iniciar o suspender remotamente la recolección y envío de estadísticas.

Todos estos módulos se relacionan entre sí, y es necesario sincronizarlos para resolver los problemas causados por la concurrencia entre la llegada de paquetes ya sea del nodo o de la línea y lo que en ese momento esté realizando la Interfase. Se debe controlar, también, la escritura y lectura sobre los buffers en los cuales se reciben y se mandan los paquetes tanto de la línea como del nodo. Se hace entonces necesaria la creación de un séptimo módulo que supervise todos los eventos (MODULO SUPERVISOR).

4.2. DESCRIPCION DETALLADA DE LOS MODULOS

MODULO INICIACION :

Como cualquier sistema, la Interfase necesita ser iniciada. Después de ocurrida una falla, o de un receso en su funcionamiento, se entra en el estado de desactividad (ver figura 8). Una vez en funcionamiento el proceso a ejecutar, estado "iniciando", consiste en :

- Iniciar las variables utilizadas : Los indicadores del estado de los buffers y los acumuladores de estadística (ver módulo) se colocan en cero señalando que éstos se encuentran vacíos; la variable que indica si se recolec

tan estadísticas se pone en "cierto".

- Colocar en "cierto" la variable que indica que la Interfase acaba de ponerse en funcionamiento y que luego será enviada al nodo que procesa las estadísticas de la red.
- Cargar el reloj de estadísticas con un tiempo bastante corto para enviar un paquete, que tenga como objetivo informarle al nodo que procesa las estadísticas de la red que la Interfase en cuestión está funcionando. Sin embargo, el paquete no lo envía directamente este módulo ya que hay otro módulo que normalmente envía los paquetes.

Una vez terminado el estado "iniciando" la Interfase se pasa al estado "ocioso" en donde permanece hasta que reciba una interrupción cualquiera.

MODULO DE RECEPCION DE LA LINEA (FIGURA 9) :

A este módulo se llega cuando se va a atender una interrupción causada por un paquete proveniente de la línea. Sus estados son los siguientes :

- Chequear disponibilidad de buffer :
Durante este estado se observan los indicadores del estado de los buffers de recepción de la línea. Si existe por lo menos uno de ellos disponible se pasa al estado "recibiendo paquete"; si por el contrario, los dos buffers destinados a la recepción de la línea se encuentran ocupados, se transfiere el control al módulo supervisor.
- Recibiendo paquete :
En este estado, la Interfase está dedicada exclusivamente a recibir, pasando caracter por caracter desde el periférico encargado de la recepción de la línea hasta el buffer que le fue asignado a tal paquete en la memoria. Al tiempo, el chip de CRC está efectuando sus cálculos con el fin de chequear posibles errores ocurridos al paquete en le medio de transmisión.

Cuando se detecta el fin de texto, se pasa al estado de "chequear CRC". Si por el contrario, al cabo de un tiempo no se detecta un caracter de fin de texto, se pasa a "liberar buffer" bajo el evento de mensaje incompleto.

- Chequear CRC :
En este momento el CRC ya tiene una respuesta acerca de la información recibida, por tanto lo único que la Interfase debe hacer es averiguar por el resultado del chequeo. Así, si CRC está errado se pasa al estado "chequear dirección", de lo contrario se pasa a "liberar buffer".
- Chequear dirección :
En este estado se chequea el destino que lleva el paquete que se acaba de recibir. Existen tres (3) posibilidades :
 - Paquete para la Interfase;
 - Paquete para el nodo;
 - Paquete para el otro nodo.

Si el paquete es para la Interfase, se pasa a un estado de "activar o desactivar estadísticas"; si es para su nodo local se asegura de hacerlo llegar a éste; y finalmente, si es para otro nodo, se pasa a "liberar buffer".

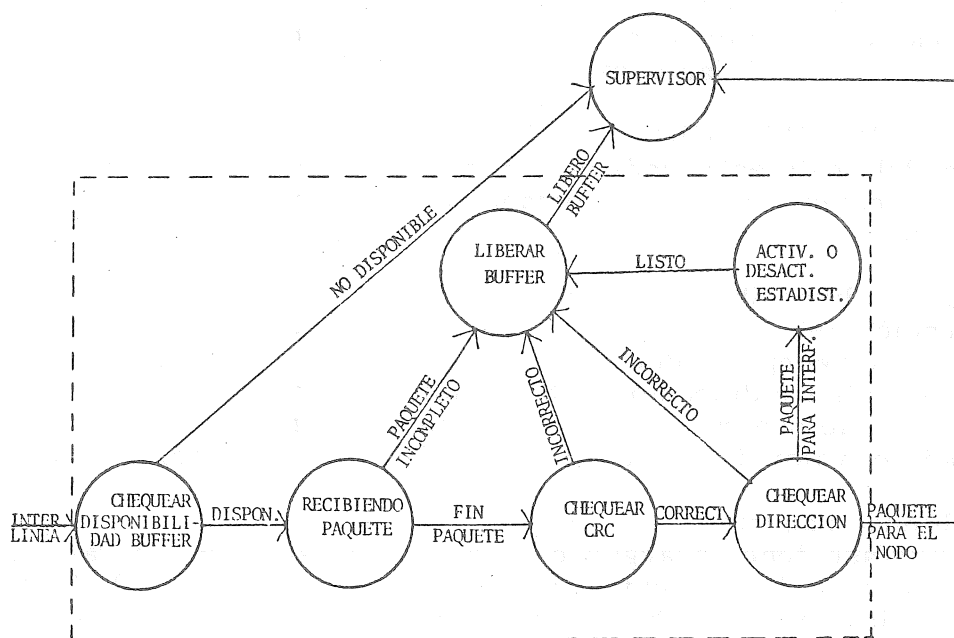


FIGURA 9
MODULO DE RECEPCION DE LA LINEA

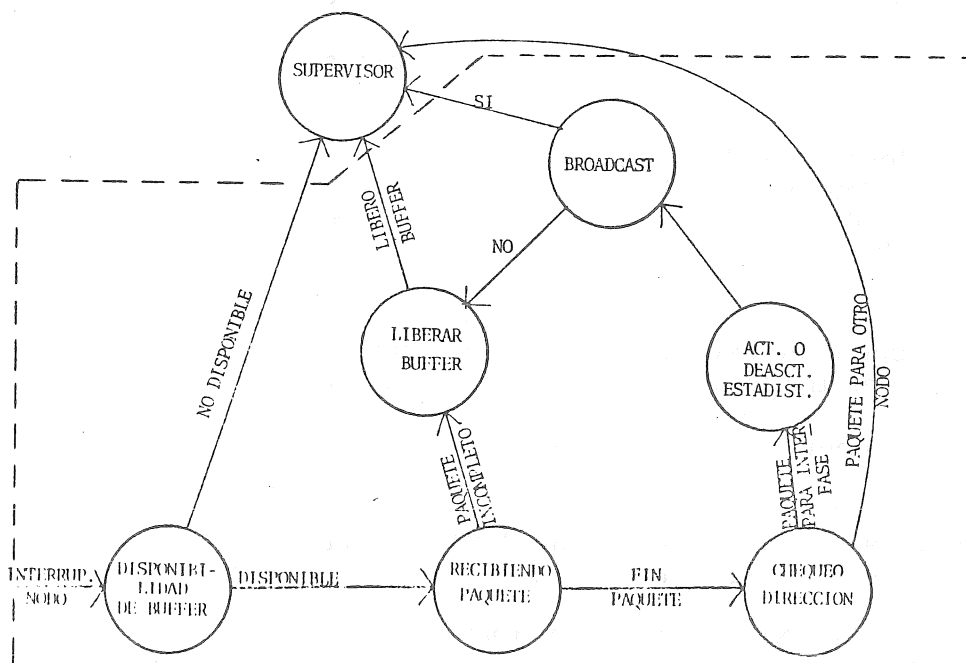


FIGURA 10
MODULO DE RECEPCION DEL NODO

- Liberar buffer :

En este estado se debe alterar el indicador del buffer de recepción de la línea, el cual almacenó el paquete en cuestión. El nuevo valor señalará su disponibilidad para recibir cualquier otro paquete.

MODULO DE RECEPCION DEL NODO (FIGURA 10) :

Este módulo está encargado de atender la interrupción causada por un paquete proveniente del nodo asociado con la Interfase. Sus estados son los siguientes :

- Chequear disponibilidad de buffer :

La función del módulo en este estado es chequear los indicadores de los buffers de recepción del nodo (transmisión a la línea) con el fin de saber si existe alguno disponible. Si es así, se pasa al estado "recibiendo paquete"; de lo contrario, se transfiere el control al supervisor, sin haber variado el valor del indicador, señalándolo como vacío.

- Recibiendo paquete :

La Interfase debe encargarse de recibir caracter por caracter el paquete que el nodo le está enviando, pasándolo al buffer que le ha sido asignado en la memoria. Si durante esta recepción, es detectado el fin de texto, se pasa al estado de "chequeo de dirección". Si por el contrario, al cabo de un tiempo predeterminado, no se ha recibido un nuevo caracter, se asume que el paquete llegó incompleto y se va al estado "liberar buffer".

- Chequear dirección :

En este estado debe chequearse la dirección de destino del paquete, con el fin de saber si se dirige a la Interfase o lleva como destino algún otro nodo. Si se dirige a otro nodo, el control es transferido al supervisor, pero si se dirige hacia la Interfase, se pasa al estado "activar/desactivar estadísticas".

- Activar/Desactivar Estadísticas :

En éste se debe alterar la variable que indica si se debe activar o desactivar la recolección de estadísticas. Una vez se ha hecho esto, se pasa al estado "Broadcast".

- Broadcast :

Lo único que debe hacerse ahora es averiguar si el paquete recibido es uno que se dirige también a todas las demás Interfases. Si no es así, debe darse el control al estado "liberar buffer"; de lo contrario se pasa al módulo supervisor.

- Liberar buffer :

En éste se altera el indicador del buffer de recepción del nodo que tenía el paquete. Su valor lo señala como disponible.

MODULO DE TRANSMISION AL NODO :

A este módulo se entra cuando en el supervisor hay pendiente una transmisión al nodo y, según el orden de prioridades, hay que atenderla (Ver figura 11). Consta de dos estados :

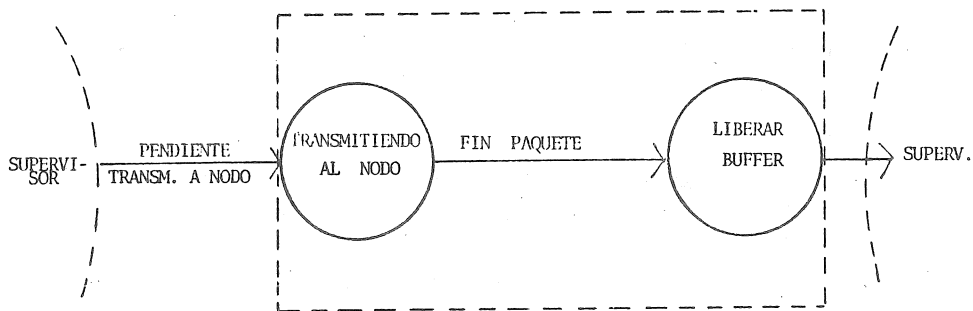


FIGURA 11
MODULO DE TRANSMISION AL NODO

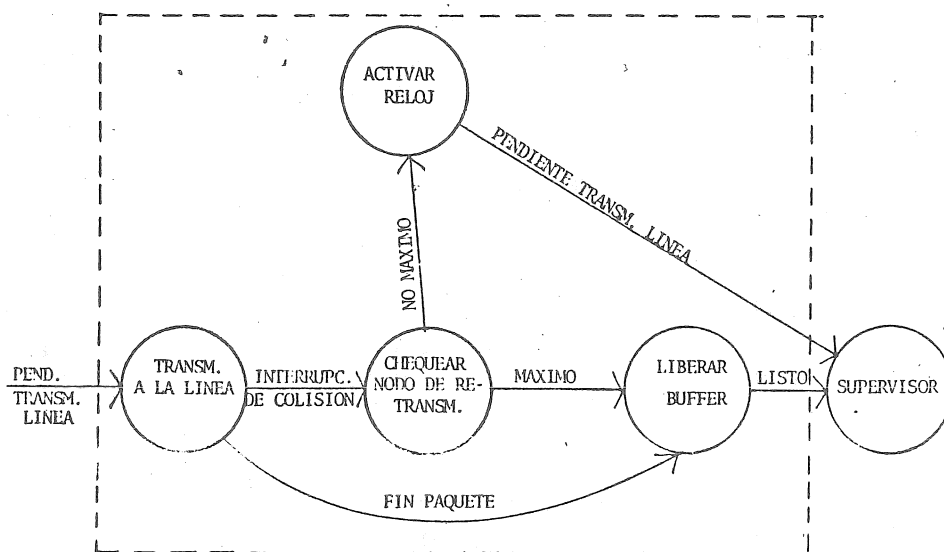


FIGURA 12
MODULO PROTOCOLO

- Transmisión al nodo :
Se averigua, dependiendo de la prioridad, de cuál buffer se va a transmitir. Se coloca luego carácter por carácter en el periférico encargado de la transmisión física hasta que el contador de caracteres llegue al límite.
- Liberar buffer :
El indicador del buffer que se transmitió se pone en vacío y se rearreglan las prioridades de los buffers. La transición para salir de este estado se hace hacia el módulo supervisor.

MODULO PROTOCOLO :

Los protocolos de acceso desde el punto de vista de cuándo se transmite en el canal compartido se dividen en tres grandes clases :

- Los de selección : una Interfase transmite cuando le llega su turno (GORD80), (CLAR78).
- Los de contención : una Interfase transmite sin esperar turno o reservar, lo que implica resolver colisiones (METC80), (CLAR78).
- Los de reservación : una Interfase transmite en el intervalo de tiempo reservado para ella.

En la red se pretende usar un protocolo del tipo contención por ser estos sencillos (menos hardware y software) y por tener control distribuido. Se explica a continuación el diagrama de estados del protocolo de acceso utilizado. Consta de cuatro estados y se llega al inicial cuando en el supervisor hay pendiente una transmisión a la línea y, según el orden de prioridades, hay que atenderla (ver figura 12):

- Transmitiendo a la línea :
Se averigua, dependiendo de la prioridad, de cuál buffer se va a transmitir. Se coloca entonces, carácter por carácter en el periférico encargado de la transmisión física hasta que el contador de caracteres llegue al límite o se detecta una colisión en la línea (produce una interrupción).
- Chequear número de retransmisiones :
Se contabilizan cuántas retransmisiones se han hecho; se sale hacia dos estados dependiendo de si es máximo o no el número de retransmisiones.
- Activar reloj :
Se activa el reloj de espera para retransmitir si el número no ha llegado al máximo. Se va luego al supervisor con el pendiente de una transmisión pero con las demás transmisiones bloqueadas hasta que se acabe el tiempo ya iniciado.
- Liberar buffer :
El indicador del buffer que se transmitió se pone en vacío y se arreglan las prioridades de los buffers. Se llega a este estado cuando se acabó de transmitir o cuando se llega al límite en retransmisiones. En caso de llegar al límite no se hace nada, y el problema debe ser resuelto por un protocolo de más alto nivel. Al salir del estado se va para el supervisor.

MODULO DE ESTADISTICAS :

Es importante hacer una recolección periódica de parámetros que midan en una u otra forma el rendimiento ya que se ayuda a :

1. detectar y corregir fallas de operación;
2. detectar y corregir errores de diseño, y
3. medir el rendimiento para afinar el sistema.

Los parámetros que se pueden medir son los siguientes : número de transmisiones, número de retransmisiones, longitud promedio de los mensajes, número de mensajes recibidos, número de mensajes con errores, número de mensajes que van para el nodo, número de mensajes rechazados por falta de buffer del nodo, número de mensajes rechazados por falta de buffer de la línea, mensajes incompletos, y caídas. De estos parámetros se pueden deducir otros como, por ejemplo, tiempo que dura ociosa la Interfase, etc.

Las estadísticas se recolectan durante el funcionamiento normal de la Interfase; cada intervalo de tiempo T, el reloj de estadísticas interrumpirá para pedir el envío de los datos recolectados. Con anterioridad se ha definido un nodo, en el cual se centralizan las estadísticas de la red para ser procesadas y reportadas. Se tiene de esta forma una visión global del funcionamiento de la red permanente (HEAR72).

La recolección de estadísticas y el envío para su procesamiento es opcional. En caso de que se desee o no hacerlo, se manda desde el nodo de estadísticas un mensaje a la(s) Interfase(s) informándole(s) al respecto. Cabe anotar, que al iniciarse la Interfase, se supone que está recolectando estadísticas.

A continuación se explica el diagrama de estados del ensamblaje de estadísticas (ver figura 13). Cuando sucede una interrupción del reloj de estadísticas y se atiende, no se ensambla de una vez sino que se avisa al supervisor que está pendiente el envío. Se entra, entonces, al módulo cuando hay pendiente un envío y se atiende. El módulo consta de cinco estados :

- Dirección nodo estadísticas :

Dependiendo de la dirección del nodo de estadísticas, el mensaje se transmite hacia la línea o hacia el nodo de la Interfase en cuestión.

- Disponibilidad de buffer de la línea :

Si la dirección del nodo de estadísticas no es la misma de la Interfase se pregunta si hay buffer disponible para mandar hacia la línea. Si no hay, se vuelve al supervisor dejando pendiente el envío; si hay buffer, se continúa con el estado "ensamblar para línea".

- Ensamblar para línea :

Se ensambla el paquete que contendrá la información de las estadísticas recolectadas en uno de los buffers designados para transmitir a la línea. Para el ensamblaje se sigue el formato establecido (ver apéndice). De aquí se sale hacia el supervisor con una transmisión de la línea pendiente.

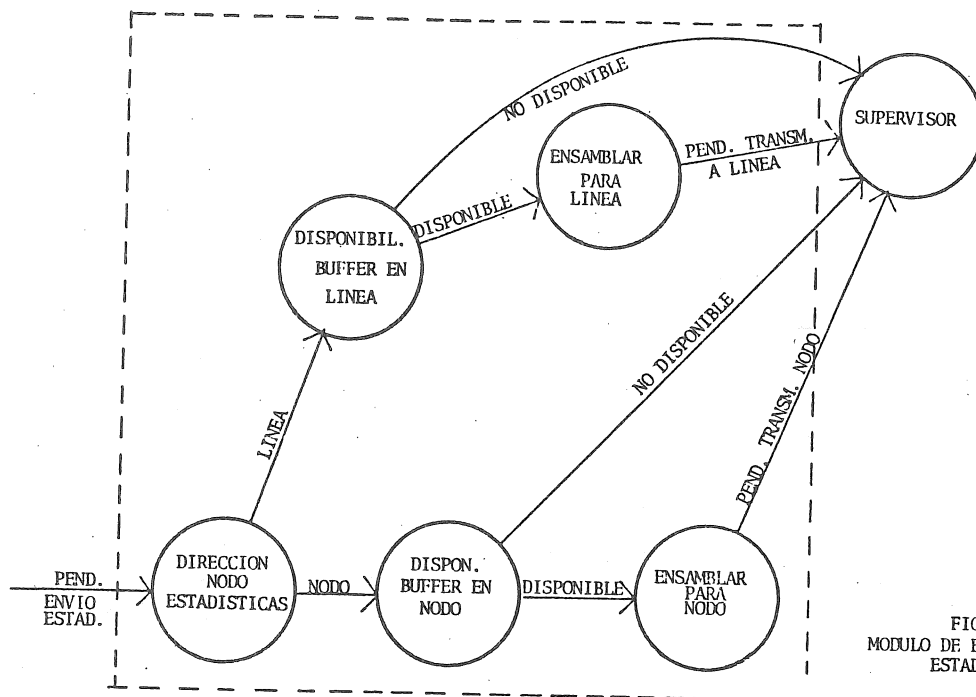


FIGURA 13
MODULO DE ENSAMBLAJE DE
ESTADISTICAS

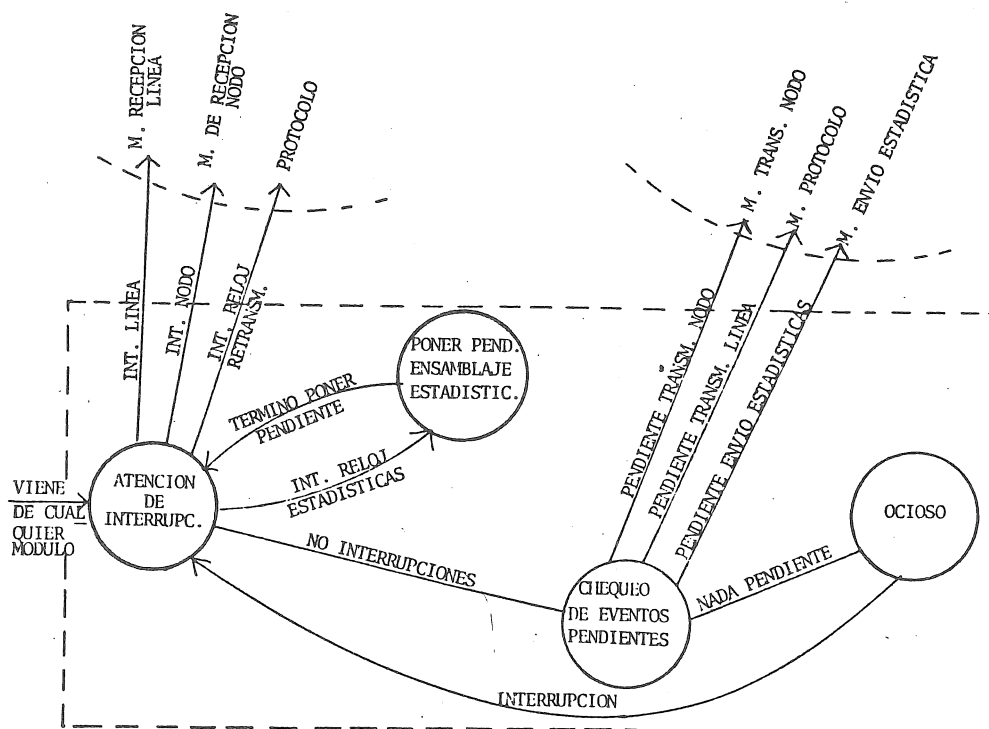


FIGURA 14
MODULO SUPERVISOR

- Disponibilidad de buffer del nodo :

Se entra a este estado cuando la dirección del nodo de estadísticas es igual a la del nodo colgado a la Interfase en cuestión. Se pregunta si hay buffer disponible para mandar hacia el nodo; si no lo hay, se vuelve al supervisor quedando pendiente el envío; si lo hay, se continúa con el estado ensamblar para el nodo.

- Ensamblar para el nodo :

Es similar al estado "ensamblar para la línea" teniendo en cuenta que se transmite hacia el nodo.

MODULO SUPERVISOR (VER FIGURA 14) :

Este módulo, de acuerdo a lo que su nombre indica, tiene como función coordinar la ejecución de las diferentes rutinas que se encargan de atender las interrupciones o ejecutar las tareas pendientes (envíos de mensajes o ensamblaje de estadísticas). (ESTU80).

Al módulo supervisor se entra viniendo de cualquier otro (ver módulos) cuando termina su función o cuando por algún motivo no la puede realizar.

Este módulo se compone de los siguientes estados :

- Atención de interrupciones :

La función del supervisor, en este estado, es "monitorear" las posibles causas de interrupción :

- a. Interrupción de la línea, en cuyo caso se transfiere el control al módulo de recepción de la línea.
- b. Interrupción del nodo, ante este tipo de pedido debe tomar control el módulo de recepción del nodo.
- c. Interrupción del reloj de Retransmisiones, en cuyo caso debe ser atendida por el módulo protocolo.
- d. Interrupción del reloj de estadísticas, la acción a seguir, es únicamente pasar el control al estado de "poner pendiente ensamblaje de estadísticas".

Si no hay interrupciones se pasa a "chequeo eventos pendientes".

- Poner pendiente ensamblaje de estadísticas :

Después de recibir una interrupción del reloj de estadísticas no se va a hacer directamente su ensamblaje; más bien se cambia el valor del indicador que lo señala como pendiente. La razón de hacerlo así, es que sería preferible ensamblar las estadísticas en el momento en que no hubiera interrupciones por atender, ya que su envío no resulta ser prioritario.

- Chequeo eventos pendientes :

En este momento, el supervisor chequea y atiende los eventos que están pendientes en espera de atención. Si hay pendiente una transmisión al nodo o a la línea, debe dársele el control a los módulos encargados de atenderla. Si es un ensamblaje de estadísticas lo que se tiene en espera, debe transferirse el control al módulo que las ensambla. Cuando no hay eventos pendientes, la Interfase entra en un estado "ocioso", hasta ser nuevamente interrumpida.

CONCLUSIONES

En este artículo se han descrito algunos de los principales componentes de la Interfase que conecta un nodo a una red local. Se hizo énfasis en las partes referenciadas al diseño mismo de este tipo de hardware basado en facilidades provistas por partes de relativa alcanzabilidad en nuestro medio. Bien se habría podido tratar de diseñar utópicas conexiones de eficientes módulos completamente inalcanzables. Por eso, fue nuestro parecer restringido y bastante realista.

Se quiso además, evitar detalles no relevantes en la descripción, con el fin de obtener mayor claridad en el diseño.

El software se diseñó de la forma más sencilla posible sin perder eficiencia. Todos los módulos en lenguaje de alto nivel se encuentran en (ARAU80).

La conclusión más importante es que no es muy complicado diseñar este tipo de hardware y software y que por lo tanto es factible de emprender este tipo de proyectos en nuestros países.

AGRADECIMIENTOS

Queremos hacer un reconocimiento especial al Doctor Antonio García, Jefe del Departamento de Ingeniería Eléctrica de la Universidad de Los Andes, por su colaboración con algunas sugerencias técnicas que han sido valiosas para el diseño de la parte hardware de la Interfase.

REFERENCIAS

- (ANDER79). Steven C. Andersen, "A Serial Data Bus Control Method", Computer Networks 3, North-Holland Publishing Company, 1.979, pp. 447-457.
- (ARAU80). Leonardo Araújo R. y Jairo Fernández S., Diseño de la Interfase para la Red Local Uniandes, Proyecto de Grado, Universidad de Los Andes, 1.980.
- (BOGG80). David R. Boggs, "Pup: An Internetwork Architecture", IEEE Transactions on Communications, Vol. Com. 28, No. 4, Abril 1.980, pp. 612-624.
- (CARP80). Robert J. Carpenter y J. Sokol, "Serving Users with a Local Area Network", Computer Networks, Vol. 4, No. 5, North-Holland Publishing Co., Octubre/Noviembre 1.980, pp. 209-214.
- (CLAR78). David D. Clark, et. al., "An Introduction to Local Area Networks", Proceedings of the IEEE, Vol. 66, No. 11, Noviembre 1.978, pp. 1497-1516.
- (CRAN80). Ronald C. Crane y Edward A. Taft, "Practical Considerations in Ethernet Local Network Design", Xerox Corporation, Febrero 1.980.

- (DANT80). André A. S. Danthine, "Protocol Representation with Finite State Models, Vol. Com. 28, No. 4, Abril 1.980.
- (DONN79). James E. Donnelley y Jeffry W. Yeh, "Interaction Between Protocol Levels in a Prioritized CMSA Broadcast Network", Computer Networks 3, North-Holland Publishing Company, 1.979, pp. 9-23.
- (ESTU80). Jacky Estublier, Curso de Sistemas Operacionales, Universidad de Los Andes, Vol. 1, Enero 1.980.
- (FARB72). D. J. Farber y K. C. Larson, "The System Architecture of the Distributed Computer System - The Comunication System" Symposium on Computer Networks, Polytechnic Institute of Brooklyn, Abril 1.972.
- (GAIT80). Laureano Gaitán y Rodrigo Paredes, Diseño de Protocolos para la Red Local Uniandes, Proyecto de Grado, Universidad de Los Andes, 1.980.
- (GORD80). R. L. Gordon, et. al., "Ringnet : A Packet Switched Network with Descentralized Control", Computer Networks 3, North-Holland Publishing Company, 1.980, pp. 373-379.
- (HEAR72). F. E. Heart, et. al., "The Interface Message Processor for the ARPA Computer Network", Computer Communications, 1.972.
- (HSI 80). Peter Hsi y Tsvi Lissack, "Local Networks' Consensus : High Speed", Data Communications, Network Analysis Corporation, Diciembre 1.980, pp. 56-66.
- (METC80). Robert M. Metcalfe y David R. Boogs, "Ethernet : Distributed Packet Switching for Local Computer Networks", Xerox Corporation, 1.980.
- (MOTO76). Motorola, Benchmark Family for Microcomputer Systems M6800 Systems References and Data Sheets. 1.976.
- (PARD79). Roberto Pardo S., "Estructura de Protocolos en Redes de Computadores", Colombia Electrónica, Nos. 7 y 8, 1.979, pp. 23-45.
- (SANC80). Edilberto Sánchez y Edgar Ruiz, Diseño de un Sistema de Archivos Distribuido para la Red Local Uniandes, Proyecto de Grado, Universidad de Los Andes, 1.980.
- (SALT80). J. H. Saltzer y K. T. Pogran, "A Star-Shaped Ring Network with High Maintainability", Computer Networks, Vol. 4, No. 5, North-Holland Publishing Co., Octubre/Noviembre 1.980, PP. 239-244.

- (THAC79). C.P. Thacker, et. al., "Alto : A personal Computer", Xerox Corporation, 1.979.
- (WEIS79). Alan J. Weissberger, "An LSI Implementation of an Intelligent CRC Computer and Programmable Character Comparator", IEEE Transactions on Computers, Vol. C-29, No. 2 , Febrero 1.980, pp. 116-124.

APENDICE A

Formato del paquete en la Red Uniandes :

El paquete comienza con un bit de sincronización cuya llegada capacita al receptor para adquirir la fase de bit (sincroniza la recepción). En seguida se tienen dos campos de dirección de 8 bits, la dirección de destino hacia el cual se dirige el paquete y la dirección del nodo en donde se originó. La dirección de destino posee sólo 7 bits para tal efecto y su octavo bit especifica si el paquete recibido es para la Interfase o para el nodo colgado a ella (uno o cero según el caso). El siguiente byte contiene la longitud del paquete (no incluye el bit de sincronización ni los bytes de CRC). Posteriormente se encuentra el campo de texto cuya longitud varía entre 0 y 253 bytes.

Finalmente, se cuenta con 16 bits de CRC (Cyclical Redundancy Check). El bit de sincronización y el CRC son generados y eliminados por medio del hardware de la Interfase; toda la información restante es transferida a 0 de la memoria de la Interfase.

El campo de texto es transparente para la Interfase y va dirigido a los protolos del nivel superior.

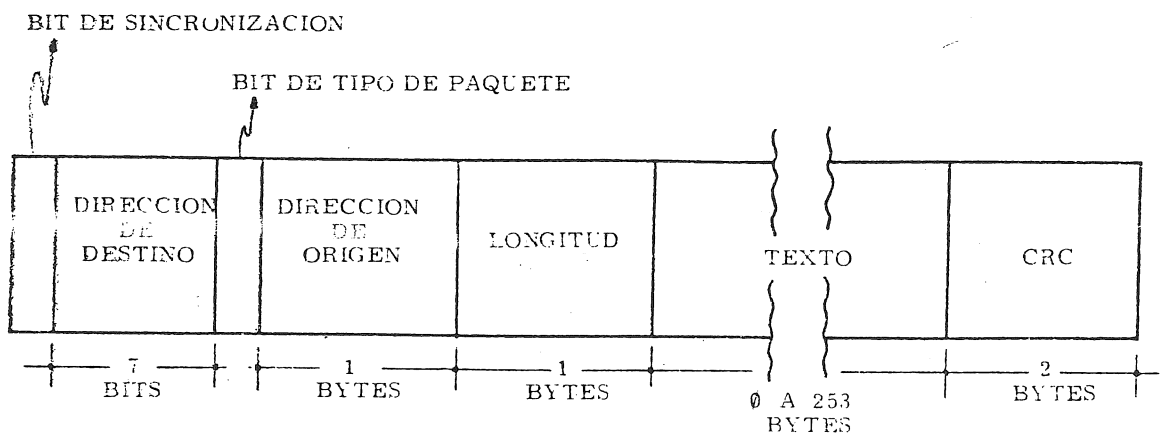


FIGURA 15
FORMATO DEL PAQUETE