(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 826 404**

(51) Int. CI.:

H04L 12/46 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **23.06.2016 PCT/US2016/038839**

(87) Fecha y número de publicación internacional: **29.12.2016 WO16210017**

(96) Fecha de presentación y número de la solicitud europea: **23.06.2016 E 16744948 (7)**

(97) Fecha y número de publicación de la concesión europea: **22.07.2020 EP 3298739**

(54) Título: **Protocolo de transporte ligero**

(30) Prioridad:

26.06.2015 US 201514752713

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
18.05.2021

(73) Titular/es:

MICROSOFT TECHNOLOGY LICENSING, LLC (100.0%)
One Microsoft Way
Redmond, Washington 98052-6399, US

(72) Inventor/es:

CAULFIELD, ADRIAN;
CHUNG, ERIC;
BURGER, DOUG y
CHIOU, DEREK

(74) Agente/Representante:

ELZABURU, S.L.P

DESCRIPCIÓN

Protocolo de transporte ligero

Antecedentes

Las redes de datos implican hardware y software. En el lado del software, los protocolos de red a menudo se diseñan para capacidades de hardware actuales o a corto plazo. Los protocolos a menudo se adaptan ampliamente al mismo tiempo que el hardware de la red mejora. Los procesadores se vuelven más eficientes y capaces y los medios de comunicación ganan en fiabilidad y capacidad. Con el tiempo, los protocolos de red diseñados en el pasado pueden llegar a ser menos adecuados para el hardware disponible en el presente. Por ejemplo, un protocolo de transporte podría tener mecanismos para garantizar la fiabilidad, la capacidad de respuesta a la congestión y el suministro solicitado. Dichos mecanismos no suelen ser muy adecuados para los nuevos tipos de redes. Sin embargo, para mantener la operatividad continua entre las redes, no es práctico modificar los protocolos más antiguos. Existe una necesidad de técnicas que puedan traducir los avances en la tecnología de redes en mejoras del rendimiento general de los protocolos posiblemente más antiguos sin tener que modificar esos protocolos o sus aplicaciones actuales.

Además, algunas capacidades del hardware de red no se han apreciado y realizado plenamente. Las denominadas tarjetas de interfaz de red (NIC) inteligentes, por ejemplo, las NIC FPGA (conjunto de puertas programables en campo) se han vuelto más comunes. Estos nuevos interfaces, como las NIC tradicionales, proporcionan conectividad física y de medios. También incluyen una capacidad de procesamiento adicional, a veces en forma de circuitos reconfigurables (por ejemplo, FPGA). Estas NIC aumentadas de procesamiento pueden permitir que las características de algunos protocolos se descarguen de la CPU (unidad central de procesamiento) del anfitrión a la NIC. Algunas NIC inteligentes pueden incluso permitir que un protocolo de transporte completo se descargue completamente de la CPU de un anfitrión a la NIC inteligente. Sin embargo, este enfoque a menudo requiere cambios significativos en el lado del anfitrión. Por ejemplo, podría ser necesario reescribir el software del lado del anfitrión para comunicarse directamente con su NIC por medio de una interfaz de programación de aplicaciones (API) personalizada en lugar de por medio de un protocolo de transporte estándar. Además, la mayoría de las NIC inteligentes con funcionalidad de descarga no funcionan como dispositivos "bump in the line" ("en línea") (es decir, dispositivos que se conectan a la NIC existente de un anfitrión). Por lo tanto, para actualizar desde un anfitrión una NIC ordinaria a una NIC inteligente, la NIC ordinaria del anfitrión se debe sustituir físicamente por una nueva NIC inteligente. Las actualizaciones a gran escala pueden ser costosas y perturbadoras. Existe una necesidad de NIC inteligentes que se puedan proporcionar a los anfitriones para mejorar el rendimiento de la red y del anfitrión sin que sea necesario realizar cambios en el software o el hardware del lado del anfitrión.

Además, como sólo los inventores han podido comprobar, el desarrollo de las NIC inteligentes también ha permitido descargar la funcionalidad del nivel de aplicación a las NIC inteligentes. Por ejemplo, una aplicación distribuida podría tener elementos en todo un centro de datos que necesitan intercambiar datos. Algunos de esos elementos o sus agentes ayudantes podrían ser capaces de ser ejecutados por una NIC inteligente. Sin embargo, no se ha visto que el código de nivel de aplicación (u otro código) en las NIC inteligentes en línea pueda ser capaz de una comunicación de red directa sin tener que atravesar sus respectivos anfitriones y pilas de red de anfitrión. El documento US2014/0254594 describe métodos, aparatos y sistemas para implementar en la conmutación de los flujos del Controlador de Interfaz de Red. El documento WO96/00468 describe el método que utiliza un protocolo de transporte ligero para hacer un túnel de protocolo punto a punto sobre una red de malla imperfecta.

A continuación, se describen las técnicas relacionadas con los problemas mencionados anteriormente.

Resumen

Se proporciona una tarjeta de interfaz de red de acuerdo con la reivindicación independiente 1 y un método de acuerdo con la reivindicación independiente 8. El siguiente resumen se incluye sólo para introducir algunos conceptos descritos en la Descripción detallada que figura a continuación. Este resumen no es exhaustivo y no tiene por objeto delimitar el alcance de la materia de estudio reivindicada, que se establece en las reivindicaciones presentadas al final.

Se proporciona una NIC (Tarjeta de Interfaz de Red) inteligente con características que permiten que la NIC inteligente funcione como una NIC en línea entre la NIC de un anfitrión y una red. La NIC inteligente proporciona una transmisión de paso de flujos de red para el anfitrión. Los paquetes enviados hacia y desde el anfitrión pasan a través de la NIC inteligente. Como punto de paso, la NIC inteligente es capaz de acelerar el rendimiento de los flujos de red de paso mediante el análisis de paquetes, la inserción de paquetes, el descarte de paquetes, la inserción o el reconocimiento de la información de congestión, etc. Además, la NIC inteligente proporciona un módulo de protocolo de transporte ligero (LTP) que le permite comunicarse directamente con otras NIC inteligentes sin pasar el tráfico de la red a través de sus respectivos anfitriones.

Muchas de las características relacionadas se explicarán a continuación con referencia a la siguiente descripción detallada considerada en relación con los dibujos adjuntos.

Breve descripción de los dibujos

La presente descripción se entenderá mejor a partir de la siguiente descripción detallada, leída a la luz de los dibujos adjuntos, en donde se utilizan números de referencia similares para designar partes similares en la descripción adjunta.

La Figura 1 muestra los anfitriones dispuestos a comunicarse entre sí por medio de una red de datos.

La Figura 2 muestra las características de alto nivel de una NIC inteligente en línea.

La Figura 3 muestra otra vista de una NIC inteligente.

La Figura 4 muestra los detalles de un conmutador.

La Figura 5 muestra los detalles de un módulo de protocolo de transporte ligero (LTP).

La Figura 6 muestra un proceso de transmisión de datos de aplicación LTP.

Descripción detallada

Las formas de realización que se describen a continuación se refieren a las NIC inteligentes en línea que aceleran los flujos de red de los anfitriones a los que proporcionan conectividad de red. Las formas de realización que se describen a continuación también se refieren a la habilitación de la comunicación directa entre las NIC. A continuación, se describen también los detalles técnicos que respaldan y mejoran esas características.

La Figura 1 muestra los anfitriones 100 dispuestos para comunicarse entre sí por medio de una red de datos 102. Los anfitriones 100 pueden ser cualquier tipo de dispositivo informático que pueda funcionar como puntos finales de flujo de red. La forma de un anfitrión 100 no es importante (por ejemplo, un servidor blade, un servidor, una estación de trabajo, un ordenador portátil, etc.), pero asume el hardware de procesamiento (por ejemplo, una CPU, unidades de procesamiento gráfico, etc.), el hardware de almacenamiento (por ejemplo, memoria, unidades de disco, etc.) y el hardware para su cooperación (buses, controladores y puertos de entrada/salida, etc.). A los efectos de la presente memoria, los detalles de la red de datos 102 no son significativos. Por ejemplo, la red de datos 102, a nivel físico/enlace, podría incluir segmentos de medios compartidos (por ejemplo, Ethernet), una o más switched fabric (por ejemplo, canal de fibra Infiniband o switched fabric), token ring, etc. Se puede utilizar cualquier tipo de red de datos 102 conocida capaz de proporcionar enrutamiento a nivel de red entre los anfitriones 100 u otros dispositivos. A título de ejemplo, se supondrá que la red de datos 102 es una red de protocolo de Internet (IP) que proporciona enrutamiento IP a los anfitriones 100, a los que se han asignado las respectivas direcciones IP.

Los anfitriones 100 se dotan con las respectivas NIC inteligentes 104. Los anfitriones 100 pueden tener sus propias NIC (no mostradas), y las NIC inteligentes 104 (que también se pueden denominar "NIC en línea") se configuran para intermediar los flujos de red 106 entre las NIC del anfitrión y la red de datos 102. Los flujos de red 106 pueden ser flujos del Protocolo de Control de Transmisión (TCP)), por ejemplo. Las NIC inteligentes 104 pueden intercambiar paquetes de red con la red de datos 102 por medio de medios/enlaces físicos 105 y pueden intercambiar paquetes de red con sus respectivos anfitriones 100 por medio de medios/enlaces físicos (no mostrados) a las NIC anfitrionas. Según se muestra en la Figura 1, los anfitriones 100 se pueden comunicar entre sí (y con otros nodos) utilizando los flujos de red intermedios 106. En resumen, una NIC inteligente 104 de un anfitrión 100 intermediara paquetes de flujos de red 106 que terminan en el anfitrión 100. Según se describe con más detalle a continuación, el tipo de intermediación proporcionado por una NIC inteligente 104 puede variar desde la lógica de paso básica que no intenta afectar a sus flujos de red 106 hasta la lógica de paso capaz de mejorar el rendimiento de sus flujos de red 106.

Como también se describirá con más detalle a continuación, las NIC inteligentes 104 pueden implementar un protocolo de transporte ligero (LTP). El LTP es un protocolo orientado a la conexión que proporciona fiabilidad y orden con una baja sobrecarga. Según se muestra en la Figura 1, las conexiones LTP 108 comienzan y terminan en las NIC inteligentes 104 y, por lo tanto, proporcionan una forma de comunicación entre las NIC que no supone una carga para los anfitriones 100, ya que los anfitriones 100 no reciben ni indexan de ninguna otra forma paquetes LTP. Las NIC inteligentes 104 son capaces, por iniciativa propia, de (i) abrir y cerrar las conexiones LTP 108, (ii) generar paquetes LTP, y (iii) enviar y recibir paquetes LTP por medio de las conexiones LTP 108.

La Figura 2 muestra las características de alto nivel de una NIC inteligente 104 en línea. Según se mencionó anteriormente, una NIC inteligente 104 se puede disponer en línea entre un anfitrión 100 y la red de datos 102 (según se utiliza en la presente memoria "en línea" es un término que se utiliza para identificar un tipo de NIC inteligente y no implica que un NIC de este tipo se conecte actualmente a un anfitrión y a una red). En particular, una NIC inteligente 104 puede tener una primera conexión física/enlace 120 que conecta físicamente la NIC inteligente 104 con su anfitrión 100. La NIC inteligente también tiene una segunda conexión física/enlace 122 que conecta a la red de datos 102. Las conexiones físicas/enlaces pueden ser cada uno de cualquier tipo, por ejemplo, Ethernet, Fibre Channel, Infiniband, etc. Una conexión física/enlace también puede ser un medio inalámbrico. Según se describió con referencia a la Figura 3, la NIC inteligente se dota con controladores de acceso a los medios (MAC) para interactuar con las conexiones físicas/enlaces 120, 122.

Para facilitar la intermediación de paso de los flujos de red 106, las NIC inteligentes 104 se dotan con uno o más componentes de paso 124. Un componente de paso 124 proporciona un almacenamiento en el búfer de almacenamiento y envío de paquetes de flujo. Un componente de paso 124 también puede incluir lógica para mejorar el rendimiento de los flujos que indexa. Según se describe con más adelante a continuación, un componente de paso 124 puede utilizar diversos factores para mejorar el rendimiento de un flujo, incluyendo, por ejemplo, notificaciones explícitas de congestión (ECN), clasificaciones de flujos (por ejemplo, con pérdidas y sin pérdidas), información sobre el estado de la red 102, características de los propios flujos, etc. Según se describe también a continuación, se pueden adoptar diversas medidas para afectar el comportamiento de los flujos, tales como la inserción de paquetes en los flujos, la inserción de marcadores de control de flujo prioritario (PFC) en los paquetes de flujo, la pausa o aceleración de los flujos, el descarte aleatorio anticipado (RED) de los paquetes de flujo, etc.

Un flujo de red 106 utilizado por aplicaciones para intercambiar datos puede pasar a través de una NIC inteligente de la siguiente manera. Una aplicación basada en anfitrión 126 (cualquier código de capa de aplicación en un anfitrión 100) tiene datos que transmitir. Los datos se pasan a través de una API/instalación del sistema operativo (por ejemplo, un flujo o un zócalo) a una pila de red 128, donde los datos se colocan en paquetes de transporte (por ejemplo, paquetes TCP), que se encapsulan en paquetes de red (por ejemplo, paquetes IP con la dirección IP del anfitrión como remitente), que a su vez se colocan en la(s) carga(s) útil(es) de las tramas de la capa física (por ejemplo, tramas Ethernet). Las tramas se pasan a través de la primera conexión física/enlace 120 a la NIC inteligente 104. La NIC inteligente 104 desmonta las tramas de red, almacena los paquetes de transporte y, opcionalmente, comprueba la IP y/o los paquetes de transporte en busca de características tales como marcadores ECN, banderas de clasificación de flujos, etc. Cuando los paquetes almacenados en el búfer están listos para ser enviados, la NIC inteligente los encapsula en un paquete IP con las mismas direcciones de origen y destino que los paquetes IP recibidos del anfitrión. Los paquetes IP se estructuran a continuación para la segunda conexión de enlace/media 122 y se transmiten sobre la misma. En efecto, los paquetes IP transmitidos por la NIC inteligente a la red de datos son generalmente los mismos que los recibidos originalmente del anfitrión.

Los paquetes de flujo recibidos por una NIC inteligente de la red de datos 102 se envían generalmente al anfitrión de la misma manera, y la pila de red 128 del anfitrión proporciona de forma similar los datos de la aplicación a la aplicación 126 basada en anfitrión. Cabe destacar que la disposición en línea de la NIC inteligente y el paso bidireccional de paquetes permite que una NIC inteligente y su anfitrión utilicen la misma dirección IP. Esto puede permitir que la NIC inteligente del anfitrión se añada o se elimine de forma transparente al anfitrión y al enrutamiento de la red de datos. Según se describe con más detalle a continuación, los paquetes LTP también pueden utilizar las direcciones IP de los anfitriones como direcciones de origen y destino para el enrutamiento IP, aunque los paquetes LTP no sean enviados o recibidos por los anfitriones.

Para facilitar las conexiones LTP entre las NIC 108, las NIC inteligentes 104 se pueden dotar con los respectivos módulos LTP 130. Los módulos LTP 130 implementan los LTP. Un módulo LTP 130 recibe una solicitud de una aplicación basada en NIC 132 para enviar datos. La aplicación basada en NIC 132 puede ser cualquier tipo de código. Por ejemplo, la aplicación basada en NIC 132 podría ser una aplicación que respalde la cooperación entre NIC inteligentes. La aplicación basada en NIC podría ser parte de una aplicación distribuida que, a través de otros medios, comparte datos con una aplicación basada en anfitrión. En una forma de realización, las NIC inteligentes tienen una interfaz de bus para conectarse con un bus de anfitrión (por ejemplo, un bus de Interconexión de Componentes Periféricos Express). El software del anfitrión puede enviar datos a través del bus a su NIC inteligente. Un agente en la NIC inteligente interactúa con el módulo LTP 130 local: solicitando una conexión a un módulo LTP remoto (especificando la solicitud la dirección de red de destino y proporcionando datos al módulo LTP 130), que a su vez empaqueta los datos en paquetes LTP y gestiona la transmisión de los paquetes LTP, según el LTP, al módulo LTP remoto. En una forma de realización, las conexiones LTP 108 son unidireccionales (no bidireccionales). En otra forma de realización, los paquetes LTP utilizados por los módulos LTP 130 son paquetes UDP (protocolo de datagramas de usuario) que en última instancia se enrutan en paquetes IP. Según se describe a continuación, una NIC inteligente 104 puede incluir un FPGA 131 u otra forma de dispositivo lógico programable. La FPGA 131 se puede configurar para implementar el módulo LTP 130, los componentes de paso 124, la aplicación basada en NIC 132, u otros elementos de una NIC inteligente.

La Figura 3 muestra otra vista de una NIC inteligente 104. La NIC inteligente 104 incluye un primer MAC 150, un segundo MAC 152, un módulo de derivación 154, un conmutador 156, el módulo LTP 130, y una aplicación basada en NIC 132. El módulo de derivación 154 puede implementar un indexado básico de paso "tonto" de paquetes de flujo para la NIC inteligente. El conmutador 156 puede ampliar el indexado de paso para incluir funciones de procesamiento de flujo tales como la inyección de paquetes en los flujos, la eliminación de paquetes de los flujos, la alteración del comportamiento de los flujos, etc.

El circuito de derivación 154 permite a la NIC inteligente 104 funcionar como un puente básico en condiciones en las que el procesamiento de los flujos podría no estar disponible. Por ejemplo, cuando la NIC inteligente se encuentra en un determinado estado como un estado de arranque, un estado de reinicio, un estado de error, etc. Se puede señalar un control de derivación 158 para hacer que los MAC 150, 152 se comuniquen o bien directamente por medio de sus respectivos búferes FIFO, o bien indirectamente pasando paquetes a través del conmutador 156 para el procesamiento en línea. El conmutador 156 también puede señalar el control de derivación 158 (véase la línea discontinua desde el conmutador 156 hasta el control de derivación 158).

Más en concreto, la NIC inteligente puede incluir servicios de comprobación de errores para determinar si se debe activar la derivación. Si se detecta una condición de error, entonces se introduce el modo de derivación. Las condiciones detectables podrían ser: se está transmitiendo un paquete que no tiene datos listos (tartamudeo de datos), se producen dos banderas de fin de paquete sin una bandera de inicio de paquete entre ellas, o se producen dos banderas de inicio de paquete sin una bandera de fin de paquete entre ellas.

Además de las condiciones fijas anteriores, la NIC inteligente puede rastrear los recuentos de entrada y salida de paquetes en ambos MAC y puede proporcionar niveles mínimos configurables de paquetes por intervalo que pueden indicar condiciones de error. Los contadores de entrada/salida se pueden utilizar para determinar si el tráfico en cualquier dirección a través de la NIC inteligente ha dejado de fluir, o si el tráfico producido internamente se ha detenido.

La Figura 4 muestra los detalles del conmutador 156 (las líneas sólidas representan las trayectorias de datos, y las líneas discontinuas representan las señales de control). El conmutador 156 proporciona características para permitir a la NIC inteligente 104 añadir paquetes a los flujos salientes con destino la red y evitar que se envíen paquetes LTP al sistema anfitrión. Si la red de datos 102 soporta varias clases de tráfico sin pérdidas, el conmutador 156 se puede configurar para que proporcione suficiente respaldo para almacenar en el búfer y pausar los flujos sin pérdidas entrantes para permitirle insertar su propio tráfico en la red. Para ello, el conmutador 156 se puede configurar para distinguir las clases de tráfico sin pérdidas (por ejemplo, el acceso directo a memoria remota (RDMA)) de las clases de flujos con pérdidas (por ejemplo, TCP/IP). Se puede utilizar un campo en la cabecera de un paquete para identificar a qué clase de tráfico pertenece el paquete.

El conmutador 156 tiene un primer puerto 170 (del lado del anfitrión) para conectarse al primer MAC 150, y un segundo puerto 172 (del lado de la red) para conectarse al segundo MAC 152. Un tercer puerto local 174 proporciona servicio interno a la NIC inteligente, por ejemplo, al módulo LTP 130. Este puerto puede aplicar contrapresión si es necesario. El conmutador 156 puede funcionar en general como un conmutador de red, con algunas limitaciones. En concreto, el conmutador 156 se puede configurar para pasar los paquetes recibidos en el puerto local 174 (por ejemplo, paquetes LTP) sólo al segundo puerto 172 (no al primer puerto 170). De forma similar, el conmutador 156 se puede diseñar para no entregar paquetes del primer puerto 170 al puerto local 174.

Internamente, el conmutador 156 tiene dos búferes de paquetes; uno para el primer puerto de recepción (Rx) 170 y otro para el segundo puerto de recepción 172. Los búferes de paquetes se dividen en varias regiones. Cada región corresponde a una clase de tráfico de paquetes. A medida que los paquetes llegan y son extraídos de sus tramas (por ejemplo, tramas Ethernet) son clasificados por clasificadores de paquetes en una de las clases de paquetes disponibles (con pérdidas, sin pérdidas, etc.) y escritos en un búfer de paquetes correspondiente. Si no hay espacio disponible en el búfer para un paquete entrante, entonces el paquete será descartado. Una vez que un paquete está almacenado y listo para transmitir, un árbitro selecciona de entre los paquetes disponibles y transmite el paquete (el paquete se puede estructurar mediante el conmutador 156 u otro elemento de la NIC inteligente). Un bloque de inserción PFC permite al conmutador 156 insertar tramas PFC entre los paquetes de flujo en la mitad de la transmisión de cualquiera de los puertos 170, 172.

El conmutador 156 puede indexar una clase de tráfico sin pérdidas de la siguiente manera. Todos los paquetes que lleguen a la mitad receptora del primer puerto 170 y a la mitad receptora del segundo puerto 172 se deben transmitir eventualmente en las correspondientes mitades transmisoras (Tx) de los puertos. Los paquetes se enrutan por almacenamiento y envío. Se puede implementar un control de flujo prioritario (PFC) para evitar la pérdida de paquetes. Para las clases de tráfico sin pérdidas, el conmutador 156 puede generar mensajes PFC y enviarlos en las partes de transmisión de los puertos primero y segundo 170, 172. En una forma de realización, los mensajes PFC se envían cuando se llena un búfer de paquetes. Cuando un buffer está lleno o a punto de estarlo, se envía un mensaje PFC al socio de enlace solicitando que el tráfico de esa clase se detenga. Los mensajes PFC también se pueden recibir y llevar a cabo. Si se recibe una trama de control PFC para una clase de tráfico sin pérdidas en la parte receptora bien del primer o bien del segundo puerto 170, 172, el conmutador 156 suspenderá el envío de paquetes en la parte transmisora del puerto que recibió la trama de control. Los paquetes se almacenarán en el búfer internamente hasta que los búferes estén llenos, momento en el que se generará una trama PFC para el socio de enlace.

El conmutador 156 puede indexar una clase de tráfico con pérdidas de la siguiente manera. El tráfico con pérdidas (todo lo que no está clasificado como sin pérdidas) se envía en base al mejor esfuerzo. El conmutador 156 es libre de dejar paquetes si hay congestión. Se pueden implementar los descartes aleatorios anticipados (RED) en la parte de recepción del primer puerto 170. Los RED pueden ayudar a limitar el tráfico con pérdidas generado por el anfitrión a un ancho de banda objetivo. Limitar el ancho de banda del anfitrión garantiza que cuando la NIC inteligente empiece a introducir su propio tráfico no habrá una correspondiente caída repentina del ancho de banda disponible, que interactuaría mal con los flujos TCP/IP que utilizan el enlace. Se puede proporcionar un parámetro de control para establecer los umbrales RED.

Como se puede observar a partir de las características del conmutador 156, una NIC inteligente 104 es capaz de actuar como una tasa de transferencia para su anfitrión al mismo tiempo que mejora el rendimiento de los flujos del anfitrión. Se pueden detectar señales de congestión en los paquetes o flujos antes de que los paquetes atraviesen un anfitrión y su pila de red. Una NIC inteligente puede reaccionar a esas señales más rápidamente que el anfitrión. Por

ejemplo, si se detecta un marcador de congestión en un paquete en su camino hacia el anfitrión, la NIC inteligente puede detener o iniciar rápidamente el flujo, aumentar o disminuir el ancho de banda disponible, estrangular otros flujos/conexiones, etc., antes de que los efectos de la congestión comiencen a manifestarse en el anfitrión.

Además, los marcadores de congestión o similares se pueden eliminar de un flujo antes de que lleguen al anfitrión, reduciendo por lo tanto la carga de indexado de los flujos en el anfitrión. Como resultado, el anfitrión puede ejecutar una pila de red ordinaria (por ejemplo, TCP/IP) y muchas de las características de alta sobrecarga de esa pila pueden permanecer inactivas. Por ejemplo, los ajustes de tamaño de la ventana se pueden minimizar o evitar, puede ser necesario un menor almacenamiento en el búfer de paquetes, el indexado de paquetes malformados o inconsistentes puede no ser necesario (la NIC inteligente puede interceptar y eliminar dichos paquetes y posiblemente solicitar su retransmisión), la señalización de la congestión se puede hacer innecesaria en el anfitrión, la necesidad de fragmentación de paquetes o ajustes de MTU, etc. Las transformaciones de flujo tales como la compresión, la codificación y los cuadros de flujo también se pueden asumir de forma transparente.

Antes de explicar el módulo LTP 130, será útil entender la función del LTP y cómo se puede utilizar. Las NIC inteligentes, en general, son NIC ordinarias que se aumentan con hardware de procesamiento que podría no ser necesario para las funciones de red mínimas de una NIC. Los procesadores de propósito general acompañados de un software apropiado en memoria estática pueden ser una forma de hardware de procesamiento utilizado para aumentar una NIC. Los circuitos digitales reconfigurables son otra forma de hardware de procesamiento que son convenientes debido a su flexibilidad y velocidad. Dichos circuitos reconfigurables, denominados en la presente memoria como dispositivos lógicos programables (PLD), pueden ser FPGA o equivalentes conocidos. El procesamiento complementario basado en NIC también se puede proporcionar con ASIC (circuitos integrados específicos para la aplicación), ASSP (piezas estándar específicas para la aplicación), SoC (sistema en un chip), etc.

Una NIC inteligente, en particular una con un PLD, se puede configurar para actuar como un nodo autónomo delgado en una red. En una forma de realización, una NIC inteligente se puede configurar con un entorno o un intérprete de comandos dentro del cual se pueden crear instancias de procesos arbitrarios o unidades de ejecución. En cualquier caso, el código (o hardware) de nivel de aplicación basado en NIC puede ser ventajoso debido a la proximidad a la red y a la falta de cargas basadas en anfitrión, tal como una pila de red compleja, el indexado de interrupciones, el intercambio de recursos, etc. En resumen, las aplicaciones basadas en NIC pueden ser capaces de comunicarse con latencias y tasas de transferencia cercanas al máximo posible en una red. Además, pueden ser capaces de garantizar el rendimiento gracias al uso de hardware dedicado. Como se señaló anteriormente, dicho código/circuito de aplicación basado en NIC también puede cooperar con el software basado en anfitrión a altas velocidades de datos si la NIC inteligente tiene una conexión de bus con el anfitrión. Como se puede observar, una NIC inteligente puede ser en sí misma una especie de agente que genera y consume el tráfico de red para sus propios fines. Para aprovechar las capacidades de procesamiento y locales de una NIC inteligente, puede ser ventajoso proporcionar a dichas NIC inteligentes su propio mecanismo o protocolo para intercambiar directamente datos por medio de la red de datos.

La Figura 5 muestra detalles del módulo LTP 130. El módulo LTP 130 puede incluir uno o más búferes de datos 190 para almacenar en el búfer la carga útil, un búfer de transmisión 192, una tabla de conexiones 194, una lógica de gestión de conexiones 196 y una lógica de transmisión de datos 198. Estos elementos pueden funcionar para proporcionar una comunicación entre las NIC inteligentes basada en una conexión eficiente y fiable.

La lógica de gestión de conexiones 196 proporciona una interfaz con la que una aplicación basada en NIC 132 puede abrir una conexión. La aplicación pasa una dirección IP de la red de destino a la lógica de gestión de conexiones 196. La lógica de gestión de conexiones 196 crea una nueva entrada en la tabla de conexiones 194. La entrada incluye un índice o handle, la dirección IP de destino y el número de secuencia actual de la conexión (inicialmente cero). A continuación, se envía una solicitud de conexión LTP a la IP de destino. El NIC inteligente del destinatario responde añadiendo una entrada correspondiente en su tabla de conexiones. En una forma de realización, cada NIC inteligente también mantiene una tabla de conexiones receptoras para las conexiones abiertas por otras NIC inteligentes. Si bien no es necesario, el uso de ambas tablas de conexiones de envío y recepción puede reducir la sobrecarga y posiblemente evitar la identificación explícita de la comunicación unidireccional frente a la bidireccional entre dispositivos.

En una forma de realización, la NIC inteligente puede incluir servicios para proporcionar y gestionar flujos o canales virtuales para aplicaciones. En esta forma de realización, la solicitud de conexión de la aplicación también puede especificar o asociar un flujo 200 con la conexión solicitada. Entonces, los datos de los flujos se pasan automáticamente a la conexión a medida que están disponibles. En otra forma de realización, la propia aplicación pasa datos a su conexión utilizando el índice/handle para especificar la conexión. Una entrada de conexión se puede eliminar de la tabla de conexiones 194, ya sea mediante una llamada explícita de la aplicación o ya sea mediante un supervisor que elimina las conexiones que no tienen actividad durante un período de tiempo definido.

La Figura 6 muestra un proceso de transmisión de datos de aplicación 202 realizado por la lógica de transmisión de datos 198. Como se señaló anteriormente, los datos de aplicación 202 se pasan al módulo LTP 130 bien directamente desde la aplicación o bien desde un flujo 200 asociado a la conexión. En el primer caso, la aplicación pasa los datos de aplicación 202 y el handle de la conexión (por ejemplo, el índice "5" de la Figura 5). En la etapa 220 se reciben los datos de la aplicación y en la etapa 222 los datos entrantes se añaden a un búfer correspondiente en el búfer de datos

190. En la etapa 224 se comprueba el búfer. En una forma de realización, el búfer se considera lleno cuando una siguiente parte de datos (o los datos de aplicación actuales 202) llenaría el búfer hasta un tamaño tal como la unidad máxima de transmisión (MTU) de la red de datos. En resumen, los datos de aplicación entrantes que se van a transmitir se almacenan en el búfer hasta que hay suficientes datos para formar un paquete LTP. Si en la etapa 224 se determina que el búfer no está listo para transmitir, entonces la lógica de transmisión de datos 198 espera datos de aplicación adicionales. Para permitir que se envíen pequeños mensajes con menor latencia, el módulo LTP 130 también se puede configurar para recibir una bandera o señal que indique que los datos de aplicación (por ejemplo, un mensaje) está completo, lo que desencadenará la transmisión de un paquete LTP.

Suponiendo que haya una cantidad suficiente de datos almacenados en el búfer, en la etapa 226 se forma una carga útil LTP a partir de los datos almacenados en el búfer y se reinicializa el búfer (los datos entrantes se incluyen bien en la carga útil o bien en el búfer vacío). En la etapa 228 se forma un paquete LTP 204 para la carga útil. El número de secuencia para la conexión relevante se incrementa en la tabla de conexiones 194 y se añade a la cabecera del nuevo paquete LTP. La dirección IP de destino se obtiene de la entrada apropiada en la tabla de conexiones 194 y se añade a la cabecera del paquete LTP. En una forma de realización, el paquete LTP 204 se formatea como un paquete UDP estándar. El formato del paquete UDP es sólo una conveniencia, y el estándar UDP en sí mismo no necesita ser implementado. Debido a que los paquetes LTP se enrutan en paquetes IP y sólo son analizados por las NIC inteligentes, se puede utilizar cualquier formato para el LTP. Si se utiliza el formato de paquetes UDP, se puede definir un número de puerto para identificar los paquetes LTP. De esta manera, la NIC inteligente puede diferenciar entre los paquetes LTP y los paquetes UDP que no son paquetes LTP y que, por lo tanto, deben pasar a través de ella. Los paquetes de control/sobrecarga de LTP, como los paquetes de reconocimiento, los paquetes de establecimiento y retirada de conexiones, etc., también se formatean para que sean identificables como paquetes LTP (por ejemplo, los paquetes UDP con un número de puerto específico). Obsérvese que, en una forma de realización, el módulo LTP también forma los paquetes IP que transportarán los paquetes LTP. Es decir, el conmutador recibe un paquete completo, bien estructurado y formateado.

En la etapa 230, el paquete LTP 204 se copia en el búfer de transmisión 192 y luego se envía al puerto local 174 del conmutador 156, que estructura y transmite el paquete LTP 204 desde la parte de transmisión de su segundo puerto 172 (el puerto orientado a la red). El módulo LTP proporciona un suministro fiable y ordenado. En una forma de realización, los paquetes LTP 206 de acuse de recibo (ACK) se transmiten para confirmar la recepción. Cada LTP ACK incluye el número de secuencia del paquete LTP del que se está acusando recibo. Cuando se acusa recibo de un paquete LTP, se puede eliminar del búfer de transmisión 192. En una forma de realización, los acuses de recibo negativos (NAKS) y/o los tiempos muertos se pueden utilizar para activar la retransmisión de los paquetes LTP almacenados en el búfer.

El módulo LTP 130 también indexa los paquetes LTP entrantes. Cuando los paquetes llegan al puerto local 174 desde el conmutador 130, el módulo LTP 130 determina qué entrada de la tabla de conexiones 194 se empareja con el paquete. Si no existe una entrada que coincida, el paquete se descarta. El módulo LTP 130 comprueba que el número de secuencia entrante coincide con el valor esperado de la entrada de la tabla de conexiones 194. Si el número de secuencia es mayor que la entrada de la tabla de conexiones, el paquete se descarta. Si el número de secuencia es menor que el esperado, se genera un LTP ACK y el paquete se descarta. Si el número de secuencia del paquete LTP entrante coincide con la entrada de la tabla de conexiones 194, entonces la carga útil del paquete se descarga y pasa a una aplicación o a un flujo asociado con la conexión (según sea el caso). Si el módulo LTP 130 no puede pasar los datos de la carga útil, entonces el paquete LTL entrante se puede descartar. Una vez que el paquete LTP entrante ha sido aceptado, se genera un LTP ACK y el número de secuencia para la conexión se incrementa en la tabla de conexiones 194. En otra forma de realización, los paquetes LTP entrantes se pueden almacenar en el búfer para reordenar los paquetes que no están en orden. Además, se puede generar un NACK cuando se detecta un paquete descartado (por medio de un número de secuencia omitido, etc.).

Dado que el tráfico LTP se indexa mediante el conmutador 156, los paquetes LTP se pueden clasificar y gestionar de la misma manera que los paquetes de los flujos de red basados en anfitrión. Por lo tanto, las conexiones LTP se pueden adherir a puestos PFC y ECN, por ejemplo. El tráfico LTP también se puede dotar con una cantidad de ancho de banda o una calidad de configuración de servicio. El espacio de cabecera no utilizado, si está disponible, se puede utilizar para enviar información ECN y/o QCN/DC (notificación de congestión cuantificada con detección de congestión basada en retardo).

REIVINDICACIONES

1. Una primera tarjeta de interfaz de red en línea, NIC (104), para indexar los flujos de red (106), comprendiendo la primera NIC en línea (104):

un primer controlador de acceso a los medios, MAC (150); un segundo MAC (152);

hardware de procesamiento (124) configurado para proporcionar la transmisión de paso (154, 156) de los paquetes de los flujos de red (106) mediante la transmisión de los paquetes del primer MAC (150) recibidos por el segundo MAC (152) y mediante la transmisión de los paquetes del segundo MAC (152) recibidos por el primer MAC (150);

un primer módulo (130) configurado para implementar un protocolo de transporte ligero, LTP; y un segundo módulo (132) configurado para comunicarse con una segunda NIC en línea arbitraria por medio del primer módulo (130) especificando una dirección de red correspondiente a la segunda NIC en línea para permitir que el primer módulo (130) establezca una conexión LTP (108) con puntos finales en la primera NIC en línea (104) y en la segunda NIC en línea, en donde la primera NIC en línea (104) se conecta a un primer anfitrión y a la red de datos, y la segunda NIC en línea se conecta a un segundo anfitrión y a la red de datos, en donde el hardware de procesamiento se configura para

proporcionar conectividad de red entre el primer anfitrión y el segundo anfitrión, realizando una transmisión de paso de los paquetes recibidos que se ha determinado que no son paquetes LTP entre NIC, y proporcionar conectividad LTP entre NIC entre la primera NIC en línea (104) y la segunda NIC en línea, y en donde los paquetes recibidos por la primera NIC en línea que se determina que son paquetes LTP entre IC se consumen por la primera NIC en línea (104) y no se reenvían al primer o segundo anfitrión mediante la primera NIC en línea, y en donde los paquetes LTP entre NIC se originan por la primera o segunda NIC en línea.

2. Un primer NIC en línea (104) de acuerdo con la reivindicación 1, en donde el segundo módulo (132) se configura además para proporcionar datos (202) al primer módulo (130), y el primer módulo (130) se configura además para hacer que el segundo MAC (152) transmita un paquete LTP (204) que contiene los datos como una carga útil.

3. Una primera NIC en línea (104) de acuerdo con la reivindicación 2, en donde los paquetes de los flujos de red se reciben y transmiten por el primer MAC (150) y el segundo MAC (152) como cargas útiles de las respectivas tramas de medios, en donde el paquete LTP se transmite por el segundo MAC (152) como una carga útil de una trama de medios, y en donde los paquetes de los flujos de red y el paquete LTP comprenden paquetes de Protocolo Internet.

4. Una primera NIC en línea (104) de acuerdo con la reivindicación 3, en donde los paquetes de los flujos de red comprenden paquetes de protocolo de control de transmisión, TCP/IP, en donde el paquete LTP comprende un paquete de protocolo de datagrama de usuario, UDP/IP, en donde los paquetes TCP/IP y el paquete UDP comprenden una dirección IP de una computadora anfitrión conectada a la primera NIC en línea, en donde la computadora anfitrión es un punto final para los flujos de red, y en donde el paquete UDP/IP comprende la dirección IP de la computadora anfitrión y una dirección IP de una computadora anfitrión remota.

5. Una primera NIC en línea (104) de acuerdo con la reivindicación 4, en donde la segunda NIC en línea se conecta a la computadora anfitrión remota, en donde la computadora anfitrión remota corresponde a una dirección IP remota, en donde uno de los flujos de red tiene puntos finales que comprenden la dirección IP anfitrión y la dirección IP remota, respectivamente, en donde el segundo NIC en línea recibe el paquete UDP/IP y, en función de una indicación de la conexión LTP, pasa los datos de la carga útil a un módulo de punto final en la segunda NIC en línea y no pasa los datos de la carga útil al anfitrión remoto, y en donde los paquetes TCP/IP de uno de los flujos de red se pasan al anfitrión remoto mediante el segundo NIC en línea.

6. Un primer NIC en línea (104) de acuerdo con la reivindicación 1, en donde el primer módulo (130) se configura para implementar la LTP manteniendo los números de secuencia para las respectivas conexiones LTP, en donde se incrementa un número de secuencia para cada paquete LTP enviado en una conexión LTP correspondiente, y en donde el primer módulo (130) incorpora un valor de un número de secuencia de la conexión LTP en el paquete LTP.

7. Una primera NIC en línea (104) de acuerdo con la reivindicación 1, en donde el primer módulo (130) se configura para implementar la LTP mediante i) el almacenamiento en el búfer de los paquetes LTP hasta que se reciban los correspondientes paquetes LTP de reconocimiento, y ii) la retransmisión de los paquetes LTP almacenados en el búfer para los que no se reciban paquetes LTP de reconocimiento dentro de un período de tiempo muerto.

8. Un método para permitir la comunicación entre NIC en una red de datos (102), comprendiendo el método:

recibir y transmitir paquetes mediante tarjetas de interfaz de red en línea, NIC (104), conectada cada NIC (104) a un anfitrión respectivo (100) y a la red de datos (102), en donde cada anfitrión se configura con una dirección de red anfitrión respectiva, en donde una primera NIC en línea se conecta a un primer anfitrión y a la red de datos y una segunda NIC en línea se conecta a un segundo anfitrión y a la red de datos, y comprendiendo cada NIC un primer controlador de acceso a los medios, MAC (150), y un segundo MAC (152), e implementando el protocolo de transporte ligero, LTP; proporcionando conectividad LTP entre NIC mediante la primera NIC en línea, en donde cualquiera de las

NIC puede iniciar una conexión LTP entre NIC (108) con cualquiera de las otras NIC, y en donde los paquetes recibidos por la primera NIC en línea que se determinan como paquetes LTP entre NIC se consumen por la primera NIC en línea y no se reenvían a los anfitriones primero o segundo por la primera NIC en línea, y en donde los paquetes LTP entre NIC se originan mediante la primera o segunda NIC en línea, y en donde los paquetes LTP entre NIC se originan mediante la primera o segunda NIC en línea; y proporcionan conectividad de red para los anfitriones primero y segundo mediante la primera NIC en línea que realiza transmisión de paso de los paquetes recibidos que se determina que no son paquetes LTP entre NIC.

9. Un método de acuerdo con la reivindicación 8, en donde cada anfitrión comprende una NIC anfitrión, en donde cada NIC (104) se conecta a la NIC anfitrión de su anfitrión correspondiente, y en donde las NIC anfitrión reciben de las NIC los paquetes que no son paquetes entre NIC.

10. Un método de acuerdo con la reivindicación 8, en donde tanto los paquetes LTP entre NIC como los paquetes que no son entre NIC se enrutan como paquetes de protocolo de internet, IP, a las NIC mediante la red en función de los paquetes IP que comprenden las direcciones de red del anfitrión.

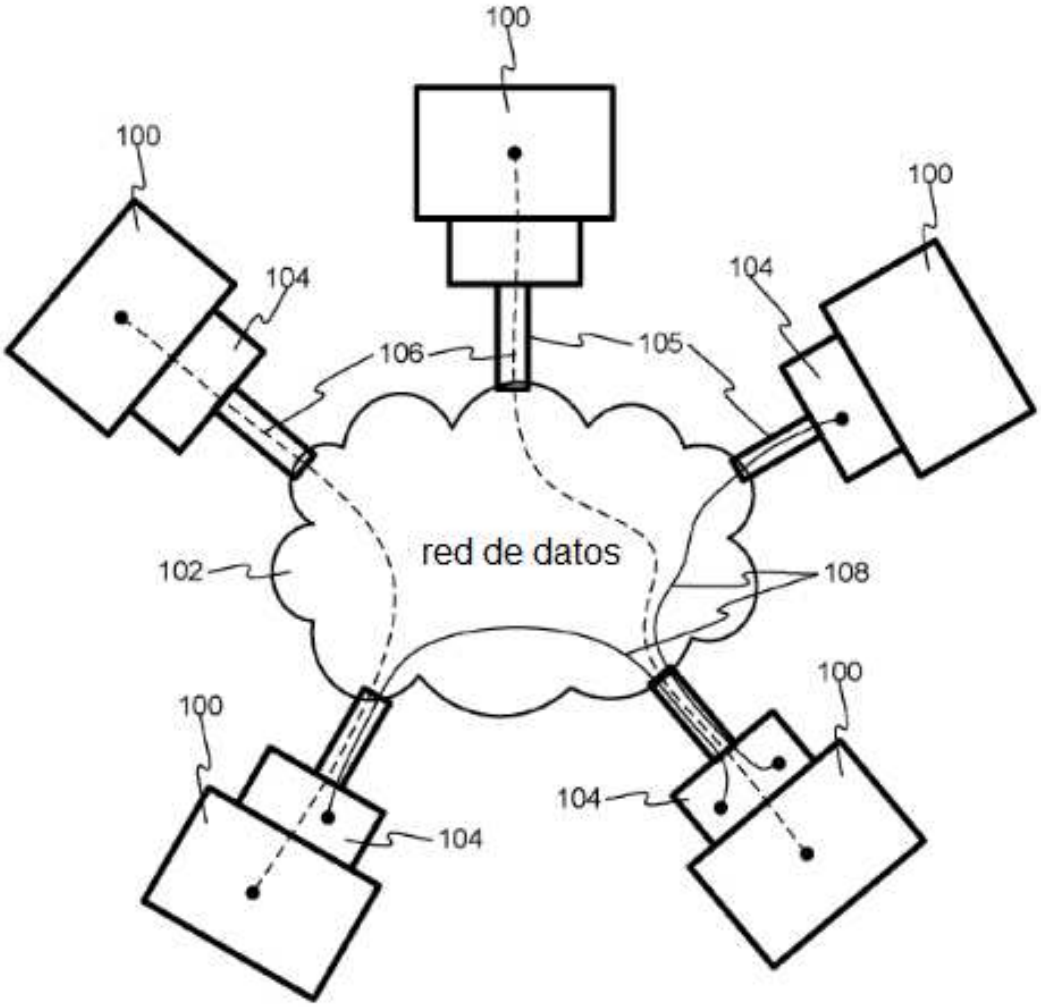

FIG. 1

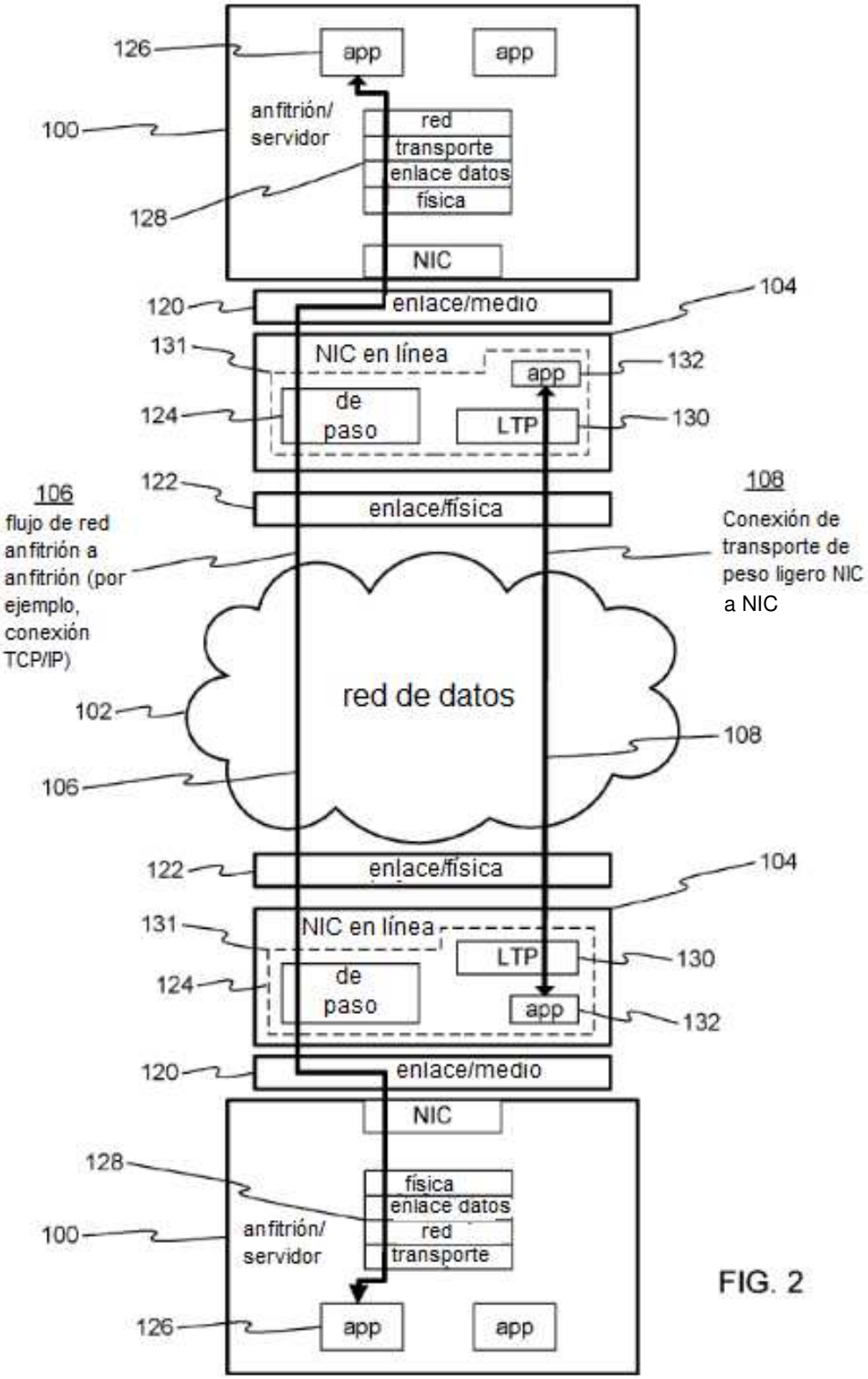
126
app
app
100
anfitrión/
servidor
red
transporte
enlace datos
física
128
NIC
120
enlace/medio
104
131
NIC en línea
app
132
124
de
paso
LTP
130
122
enlace/física
106
flujo de red
anfitrión a
anfitrión (por
ejemplo,
conexión
TCP/IP)
108
Conexión de
transporte de
peso ligero NIC
a NIC
red de datos
102
108
106
122
enlace/física
104
131
NIC en línea
LTP
130
124
de
paso
app
132
120
enlace/medio
NIC
128
física
enlace datos
100
anfitrión/
servidor
red
transporte
126
app
app

FIG. 2

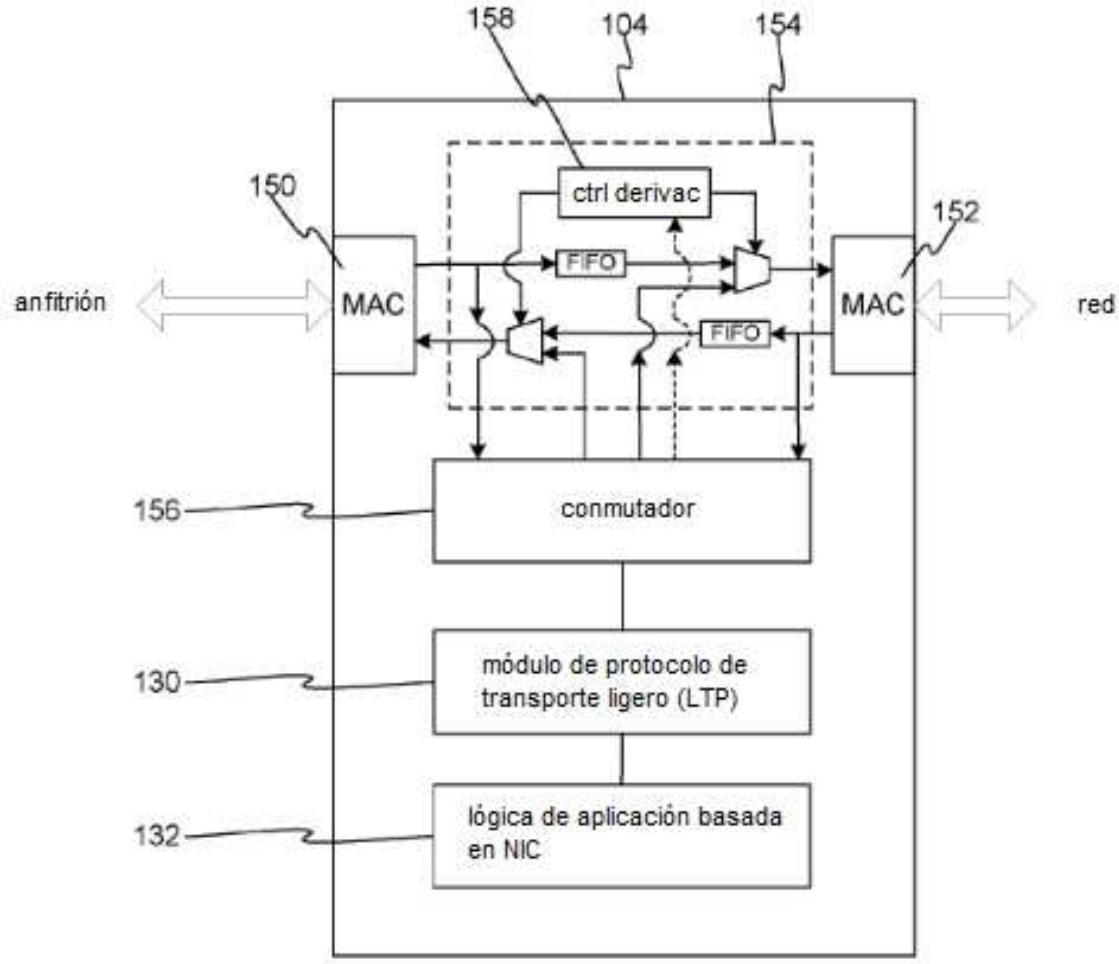
158
104
154
150
ctrl derivac
152
FIFO
anfitrión
MAC
MAC
red
FIFO
156
conmutador
130
módulo de protocolo de transporte ligero (LTP)
132
lógica de aplicación basada en NIC

FIG. 3

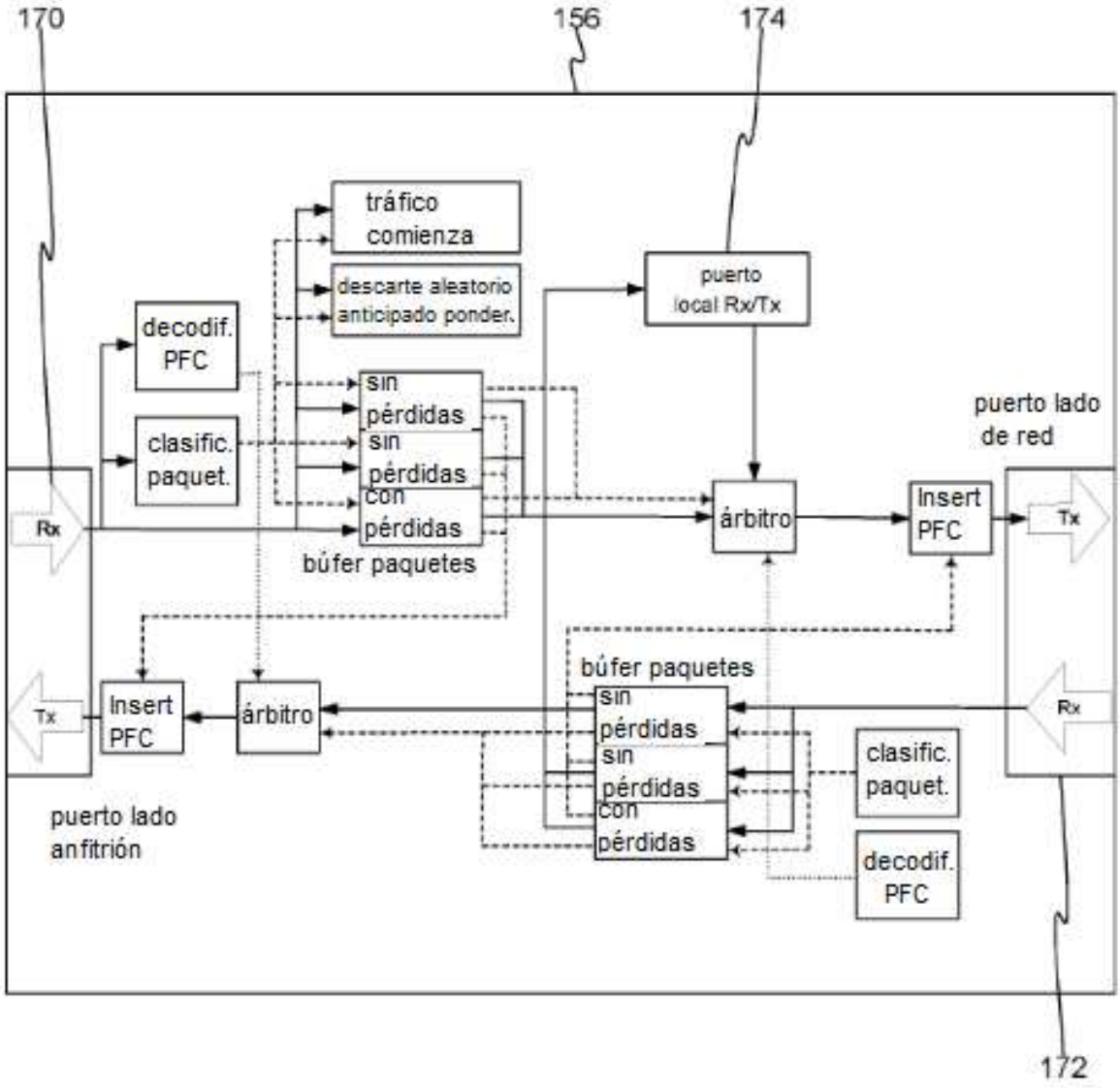
170
156
174
tráfico comienza
descarte aleatorio anticipado ponder.
puerto local Rx/Tx
decodif. PFC
clasific. paquet.
sin pérdidas
sin pérdidas
con pérdidas
búfer paquetes
puerto lado de red
árbitro
Insert PFC
Tx
Rx
Rx
Insert PFC
árbitro
Tx
búfer paquetes
sin pérdidas
sin pérdidas
con pérdidas
clasific. paquet.
decodif. PFC
puerto lado anfitrión
172

FIG. 4

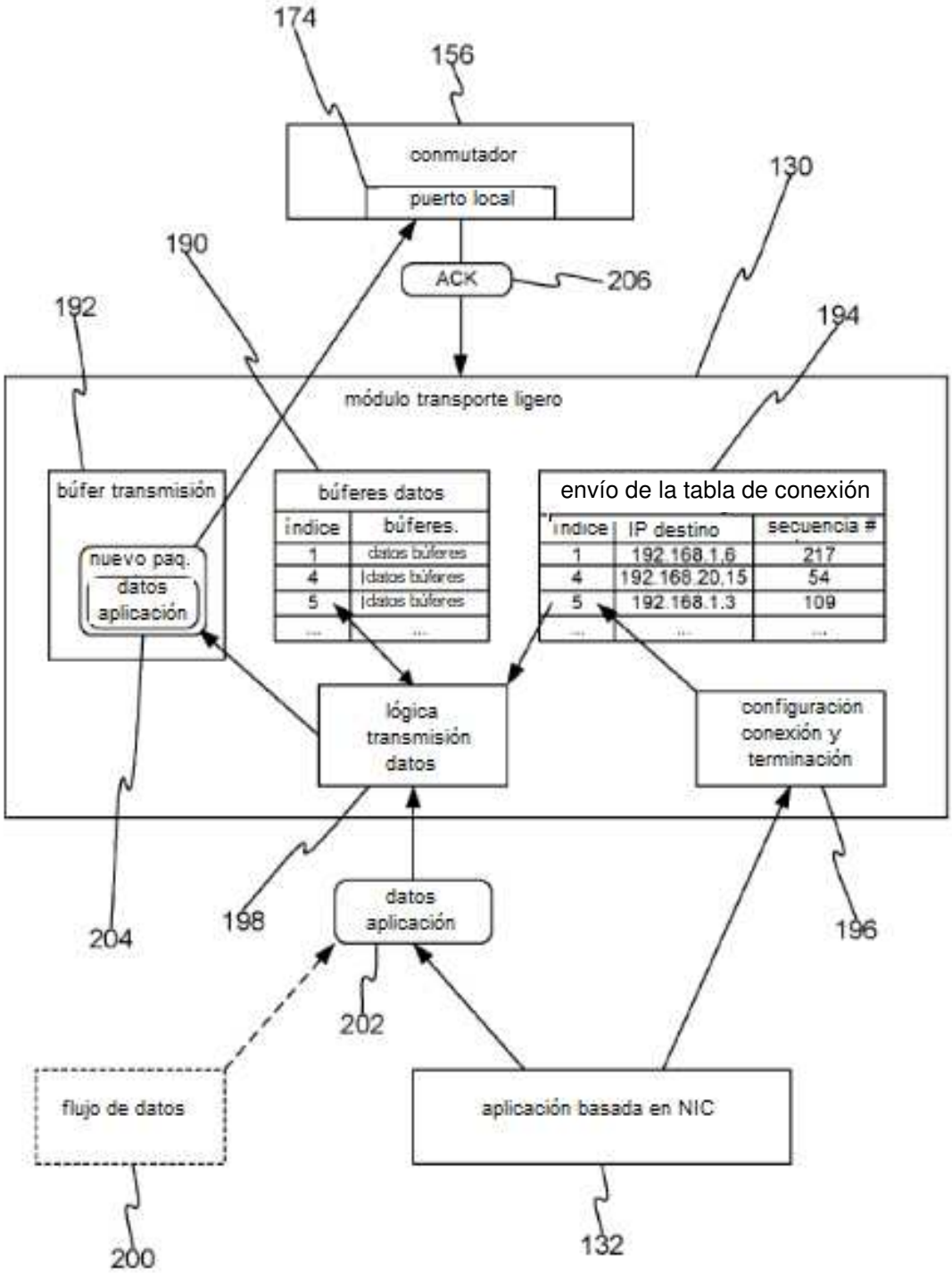
174
156
conmutador
puerto local
130
190
ACK
206
192
194
módulo transporte ligero
búfer transmisión
nuevo paq.
datos aplicación
búferes datos
índice
búferes.
1
datos búferes
4
datos búferes
5
datos búferes
envío de la tabla de conexión
índice
IP destino
secuencia #
1
192.168.1.6
217
4
192.168.20.15
54
5
192.168.1.3
109
lógica transmisión datos
configuración conexión y terminación
204
198
datos aplicación
196
202
flujo de datos
aplicación basada en NIC
200
132

FIG. 5

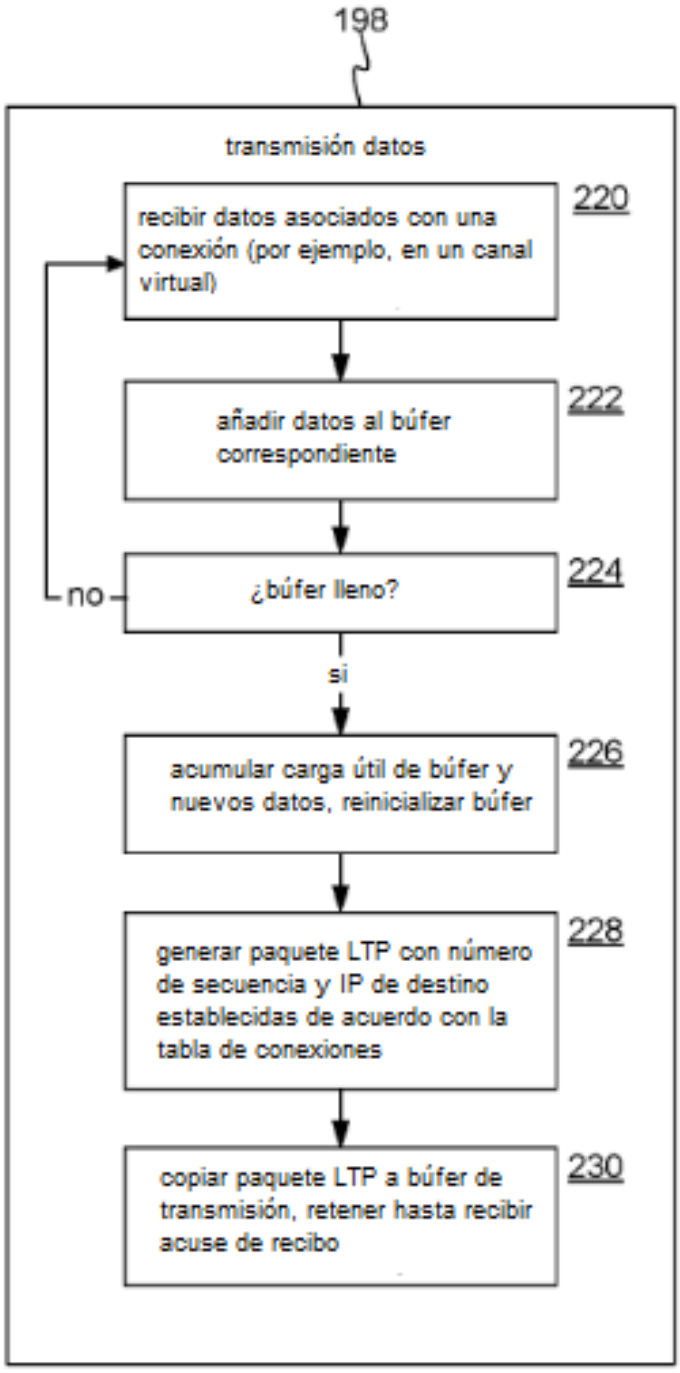
198
transmisión datos
recibir datos asociados con una conexión (por ejemplo, en un canal virtual)
220
añadir datos al búfer correspondiente
222
¿búfer lleno?
224
no
si
acumular carga útil de búfer y nuevos datos, reinicializar búfer
226
generar paquete LTP con número de secuencia y IP de destino establecidas de acuerdo con la tabla de conexiones
228
copiar paquete LTP a búfer de transmisión, retener hasta recibir acuse de recibo
230

FIG. 6