

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 369 715**

51 Int. Cl.:
G06F 13/28 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 96 Número de solicitud europea: **06815807 .0**
96 Fecha de presentación: **29.09.2006**
97 Número de publicación de la solicitud: **1943595**
97 Fecha de publicación de la solicitud: **16.07.2008**

54 Título: **DMA UNIFICADO.**

30 Prioridad:
29.09.2005 US 238790

45 Fecha de publicación de la mención BOPI:
05.12.2011

45 Fecha de la publicación del folleto de la patente:
05.12.2011

73 Titular/es:
**APPLE INC.
1 INFINITE LOOP
CUPERTINO, CA 95014, US**

72 Inventor/es:
**GO, Dominic;
HAYTER, Mark, D.;
CHEN, Zongijan;
WADHAWAN, Ruchi y
KU, Weichun**

74 Agente: **Fàbrega Sabaté, Xavier**

ES 2 369 715 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín europeo de patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre concesión de Patentes Europeas).

DESCRIPCIÓN

DMA unificado

5 **ANTECEDENTES**

Campo de la invención

10 Esta invención se refiere al campo de circuitos integrados y, más específicamente, al acceso directo a memoria (DMA) en sistemas que comprenden uno o más circuitos integrados.

Descripción de la técnica relacionada

15 En un sistema típico que incluye uno o más procesadores, memoria y dispositivos o interfaces de entrada/salida (E/S), las transferencias de acceso directo a memoria (DMA) se utilizan a menudo para transferir datos entre la E/S y la memoria. En algunos sistemas, se incluyen circuitos individuales de DMA en cada dispositivo o interfaz de E/S que usa el DMA. En otros sistemas, uno o más dispositivos de E/S pueden compartir circuitos de DMA.

20 Algunos sistemas también incluyen un "acarreador de datos" que puede usarse para copiar datos desde un área de memoria a otra. El acarreador de datos puede descargar a los procesadores, que en caso contrario tendrían que ejecutar instrucciones para llevar a cabo el movimiento de datos (por ejemplo, leer y escribir datos con el ancho que usa el procesador, habitualmente de 32 bits o 64 bits a la vez). El modelo de programación para el acarreador de datos es habitualmente distinto al modelo de programación de DMA, que está acondicionado para la comunicación entre los dispositivos de E/S y la memoria.

25 El documento US 2004/0064600 A1 divulga un controlador de disco con acceso directo a memoria usado en operaciones de transferencia de datos asistidas por hardware, que incluye lógica receptora de comandos para recibir un comando de transferencia de datos emitido por un procesador. El documento WO 2004/010314 A2 proporciona un procedimiento, sistema y programa para un sistema de bus local acoplado con un puerto, que está asociado a un espacio de direcciones de memoria.

30 **RESUMEN DE LA INVENCIÓN**

35 En una forma de realización, un aparato comprende un primer circuito de interfaz, un controlador de acceso directo a memoria (DMA) acoplado con el primer circuito de interfaz y un anfitrión acoplado con el controlador de DMA. El primer circuito de interfaz está configurado para comunicarse por una interfaz según un protocolo. El anfitrión comprende al menos un espacio de direcciones mapeado, al menos en parte, a una pluralidad de ubicaciones de memoria en un sistema de memoria del anfitrión. El controlador de DMA está configurado para llevar a cabo transferencias de DMA entre el primer circuito de interfaz y el espacio de direcciones, y el controlador de DMA está adicionalmente configurado para llevar a cabo transferencias de DMA entre una primera pluralidad de la pluralidad de ubicaciones de memoria y una segunda pluralidad de la pluralidad de ubicaciones de memoria. También se contempla un procedimiento.

La invención es según lo definido por las reivindicaciones independientes adjuntas 1 y 9.

45 **BREVE DESCRIPCIÓN DE LOS DIBUJOS**

La siguiente descripción detallada hace referencia a los dibujos adjuntos, que se describen ahora brevemente.

La Fig. 1 es un diagrama de bloques de una forma de realización de un sistema.

50 La Fig. 2 es un diagrama de bloques de una forma de realización de un controlador de DMA mostrado en la Fig. 1.

La Fig. 3 es un diagrama de bloques de una forma de realización de un motor de descarga mostrado en la Fig. 2.

La Fig. 4 es un diagrama de bloques de una forma de realización del DMA en el sistema de la Fig. 1.

La Fig. 5 es un diagrama de bloques de una forma de realización de anillos descriptores y anillos de punteros de memoria intermedia.

55 La Fig. 6 es un diagrama de flujo que ilustra el funcionamiento de una forma de realización de un motor de precaptura de recepción mostrado en la Fig. 2.

La Fig. 7 es un diagrama de flujo que ilustra el funcionamiento de una forma de realización de un circuito de control de recepción mostrado en la Fig. 2.

La Fig. 8 es un diagrama de flujo que ilustra la operación de precaptura de una forma de realización de un circuito de control de transmisión mostrado en la Fig. 2.

La Fig. 9 es un diagrama de flujo que ilustra la operación de transmisión de una forma de realización de un circuito de control de transmisión mostrado en la Fig. 2.

5 La Fig. 10 es un diagrama de bloques que ilustra un anillo descriptor con un descriptor de control incluido con los descriptores de transferencia.

La Fig. 11 es un diagrama de flujo que ilustra una forma de realización del procesamiento de descriptores de control.

La Fig. 12 es un diagrama de bloques que ilustra una forma de realización de un descriptor de DMA de recepción.

La Fig. 13 es un diagrama de bloques que ilustra una forma de realización de un descriptor DMA de transmisión.

10 La Fig. 14 es un diagrama de bloques que ilustra una forma de realización de un descriptor de copia de DMA.

La Fig. 15 es un diagrama de bloques de una forma de realización de un descriptor de descarga de DMA.

La Fig. 16 es un diagrama de bloques de una forma de realización de un descriptor de control.

La Fig. 17 es un diagrama de bloques de una forma de realización de un generador de sumas de control mostrado en la Fig. 3.

15 La Fig. 18 es un diagrama de bloques de una forma de realización de un sumador completo mostrado en la Fig. 17.

Si bien la invención es susceptible a diversas modificaciones y formas alternativas, las formas de realización específicas de la misma se muestran a modo de ejemplo en los dibujos y se describirán en detalle en el presente documento. Debería entenderse, sin embargo, que los dibujos y la descripción detallada para los mismos no están concebidos para limitar la invención a la forma específica revelada, sino que, por el contrario, la intención es cubrir todas las modificaciones, equivalentes y alternativas que caigan dentro del espíritu y el alcance de la presente invención, según lo definido por las reivindicaciones adjuntas.

DESCRIPCIÓN DETALLADA

25 Pasando ahora a la Fig. 1, se muestra un diagrama de bloques de una forma de realización de un sistema 10. En la forma de realización ilustrada, el sistema 10 incluye un anfitrión 12, un controlador 14 de DMA, circuitos 16 de interfaz y una capa de interfaz física (PHY) 36. El controlador 14 de DMA está acoplado con el anfitrión 12 y los circuitos 16 de interfaz. Los circuitos 16 de interfaz están adicionalmente acoplados con la capa 36 de interfaz física. En la realización ilustrada, el anfitrión 12 incluye uno o más procesadores tales como los procesadores 18A-18B, uno o más controladores de memoria tales como los controladores 20A-20B de memoria, un puente de E/S (IOB) 22, una memoria de E/S (IOM) 24, una memoria temporal de E/S (IOC) 26, una memoria temporal 28 de nivel 2 (L2) y una interconexión 30. Los procesadores 18A-18B, los controladores 20A-20B de memoria, el IOB 22 y la memoria temporal 28 de L2 están acoplados con la interconexión 30. El IOB 22 está adicionalmente acoplado con la IOC 26 y la IOM 24. El controlador 14 de DMA también está acoplado con el IOB 22 y la IOM 24. En la forma de realización ilustrada, los circuitos 16 de interfaz incluyen un controlador 32 de interfaz periférica y uno o más circuitos de control de acceso al medio (MAC), tales como los MAC 34A-34B. Los MAC 34A-34B están acoplados con el controlador 14 de DMA y con la capa 36 de interfaz física. El controlador 32 de interfaz periférica también está acoplado con el puente 22 de E/S y la memoria 34 de E/S (y por tanto indirectamente acoplado con el controlador 14 de DMA) y con la capa 36 de interfaz física. Tanto el controlador 32 de interfaz periférica como los MAC 34A-34C incluyen registros 38A-38C de configuración. En algunas formas de realización, los componentes del sistema 10 pueden integrarse sobre un único circuito integrado, como un sistema en un chip. En otras formas de realización, el sistema 10 puede implementarse como dos o más circuitos integrados.

45 El anfitrión 12 puede comprender uno o más espacios de direcciones. Al menos una parte de un espacio de direcciones en el anfitrión 12 puede mapearse a ubicaciones de memoria en el anfitrión 12. Es decir, el anfitrión 12 puede comprender un sistema de memoria mapeado a direcciones en el espacio de direcciones del anfitrión. Por ejemplo, cada uno de los controladores 20A-20B de memoria puede acoplarse con memoria (no mostrada) que comprende las ubicaciones de memoria mapeadas en el espacio de direcciones. En algunos casos, la totalidad del espacio de direcciones puede mapearse a las ubicaciones de memoria. En otros casos, algo del espacio de direcciones puede ser E/S mapeada a memoria (por ejemplo, la interfaz periférica controlada por el controlador 32 de interfaz periférica puede incluir alguna E/S mapeada a memoria).

55 El controlador 14 de DMA está configurado para llevar a cabo transferencias de DMA entre los circuitos 16 de interfaz y el espacio de direcciones del anfitrión. En particular, las transferencias de DMA pueden ser entre ubicaciones de memoria a las cuales está mapeado el espacio de direcciones y los circuitos 16 de interfaz. Adicionalmente, el controlador 14 de DMA, en algunas formas de realización, puede configurarse para llevar a cabo transferencias de DMA entre conjuntos de

ubicaciones de memoria dentro del espacio de direcciones. Es decir, tanto el origen como el destino de una tal transferencia de DMA pueden ser ubicaciones de memoria. La funcionalidad de un acarreador de datos puede por tanto incorporarse al controlador 14 de DMA, y puede no requerirse un acarreador de datos distinto, en algunas formas de realización. El modelo de programación para las transferencias de DMA de memoria a memoria puede ser similar al modelo de programación para otras transferencias de DMA (por ejemplo, descriptores de DMA, descritos en más detalle más adelante). Una transferencia de DMA de memoria a memoria también puede denominarse una transferencia de copia de DMA.

El controlador 14 de DMA puede configurarse para llevar a cabo una o más operaciones (o "funciones") sobre los datos de DMA según se están transfiriendo los datos de DMA, en algunas formas de realización. Las operaciones pueden llevarse a cabo sobre transferencias entre el espacio de direcciones y los circuitos de interfaz, y también pueden llevarse a cabo sobre transferencias de copia de DMA, en algunas formas de realización. Las operaciones llevadas a cabo por el controlador 14 de DMA pueden reducir la carga de procesamiento sobre los procesadores 18A-18B, en algunas formas de realización, ya que los procesadores no necesitan llevar a cabo las operaciones que lleva a cabo el controlador 14 de DMA. En una forma de realización, algunas de las operaciones que lleva a cabo el controlador 14 de DMA son operaciones sobre datos en paquetes (por ejemplo, encriptación/descriptación, generación o comprobación del control de redundancia cíclico, generación o comprobación de la suma de control, etc.). Las operaciones también pueden incluir una operación de O exclusivo (XOR), que puede usarse para el procesamiento de formaciones redundantes de discos baratos (RAID), por ejemplo.

En general, las transferencias de DMA pueden ser transferencias de datos desde un origen a un destino, donde al menos uno de los destinos es una ubicación de memoria u otra(s) dirección(es) en el espacio de direcciones del anfitrión. Las transferencias de DMA se efectúan sin que los datos transferidos atraviesen el procesador, o procesadores, en el sistema (por ejemplo, los procesadores 18A-18B). El controlador 14 de DMA puede efectuar transferencias de DMA leyendo en el origen y escribiendo en el destino. Por ejemplo, una transferencia de DMA desde la memoria a un circuito 16 de interfaz puede ser efectuada por el controlador 14 de DMA, generando solicitudes de lectura de memoria (al IOB 22, en la forma de realización ilustrada, que lleva a cabo transacciones de lectura coherente sobre la interconexión 30 para leer los datos) y transmitiendo los datos leídos como datos de DMA al circuito 16 de interfaz. En una forma de realización, el controlador 14 de DMA puede generar solicitudes de lectura para leer datos en la IOM 24 para una transferencia de DMA a través del controlador 32 de interfaz periférica, y el controlador 32 de interfaz periférica puede leer los datos de la IOM 24 y transmitir los datos. Una transferencia de DMA desde un circuito 16 de interfaz a la memoria puede ser efectuada por el controlador 14 de DMA, recibiendo datos del circuito 16 de interfaz y generando solicitudes de escritura en memoria (al IOB 22, en la forma de realización ilustrada) para transferir los datos de DMA a la memoria. En una forma de realización, el controlador 32 de interfaz periférica puede escribir datos en la IOM 24, y el controlador 14 de DMA puede hacer que los datos se escriban en la memoria. Así, el controlador 14 de DMA puede proporcionar asistencia de DMA para el controlador 32 de interfaz periférica. Las transferencias de copia de DMA pueden efectuarse generando solicitudes de lectura de memoria para las ubicaciones de memoria de origen y solicitudes de escritura en memoria para las ubicaciones de memoria de destino (incluyendo los datos de DMA de las solicitudes de lectura de memoria).

El anfitrión 12 puede comprender, en general, uno o más procesadores y controladores de memoria, configurados para mantener interfaces con la memoria mapeada en el espacio de direcciones del anfitrión 12. El anfitrión 12, optativamente, puede incluir otros circuitos, tales como la memoria temporal 28 de L2, para mejorar las prestaciones de los procesadores en el anfitrión 12. Además, el anfitrión 12 puede incluir sistema de circuitos para mantener interfaces con diversos circuitos de E/S y el controlador 14 de DMA. Si bien se ilustra una implementación del anfitrión 12 en la Fig. 1, otras formas de realización pueden incluir cualquier construcción e interfaz con el controlador 14 y los circuitos 16 de interfaz.

Los procesadores 18A-18B comprenden sistema de circuitos para ejecutar instrucciones definidas en una arquitectura de un conjunto de instrucciones implementadas por los procesadores 18A a 18B. Cualquier arquitectura de conjunto de instrucciones puede implementarse en diversas formas de realización. Por ejemplo, puede implementarse la arquitectura del conjunto de instrucciones PowerPC™. Otras arquitecturas a título de ejemplo de conjuntos de instrucciones pueden incluir el conjunto de instrucciones ARM™, el conjunto de instrucciones MIPS™, el conjunto de instrucciones SPARC™, el conjunto de instrucciones x86 (también denominado IA-32), el conjunto de instrucciones IA-64, etc.

Los controladores 20A-20B de memoria comprenden sistema de circuitos configurado para mantener interfaces con la memoria. Por ejemplo, los controladores 20A-20B de memoria pueden configurarse para mantener interfaces con memoria de acceso aleatorio dinámico (DRAM), tal como la DRAM sincrónica (SDRAM), la SDRAM de doble velocidad de datos (DDR), la SDRAM DDR2, la DRAM Rambus (RDRAM), etc. Los controladores 20A-20B de memoria pueden recibir transacciones de lectura y escritura para la memoria con la cual están acoplados desde la interconexión 30, y pueden llevar a cabo las operaciones de lectura/escritura en la memoria. Las transacciones de lectura y escritura pueden incluir transacciones de lectura y escritura iniciadas por el IOB 22 en beneficio del controlador 14 de DMA y/o el controlador 32 de interfaz periférica. Adicionalmente, las transacciones de lectura y escritura pueden incluir transacciones generadas por los procesadores 18A-18B y/o la memoria temporal 28 de L2.

5 La memoria temporal 28 de L2 puede comprender una memoria temporal configurada para almacenar temporalmente copias de datos correspondientes a diversas ubicaciones de memoria en las memorias con las cuales están acoplados los controladores 20A-20B de memoria, para el acceso de baja latencia por parte de los procesadores 18A-18B y/u otros agentes, por la interconexión 30. La memoria temporal 28 de L2 puede comprender cualquier capacidad y configuración (por ejemplo, directamente mapeada, asociativa por conjuntos, etc.).

10 El IOB 22 comprende sistema de circuitos configurado para comunicar transacciones por la interconexión 30 en beneficio del controlador 14 de DMA y el controlador 32 de interfaz periférica. La interconexión 30 puede dar soporte a la coherencia de memoria temporal, y el IOB 22 puede participar en la coherencia y garantizar la coherencia de las transacciones iniciadas por el IOB 22. En la forma de realización ilustrada, el IOB 22 emplea la IOC 26 para almacenar temporalmente transacciones recientes iniciadas por el IOB 22. La IOC 26 puede tener cualquier capacidad y configuración, en diversas formas de realización, y puede ser coherente. La IOC 26 puede usarse, p. ej., para almacenar temporalmente bloques de datos que están sólo parcialmente actualizados, debido a lecturas/escrituras generadas por el controlador 14 de DMA y el controlador 32 de interfaz periférica. Usando la IOC 26, pueden evitarse secuencias de lectura-modificación-escritura por la interconexión 30, en algunos casos. Adicionalmente, las transacciones por la interconexión 30 pueden evitarse para una coincidencia en memoria temporal en la IOC 26, para una lectura/escritura generada por el controlador 14 de DMA o el controlador 32 de interfaz periférica si la IOC 26 tiene suficiente poder de propiedad sobre el bloque de memoria temporal como para completar la lectura/escritura. Otras formas de realización pueden no incluir la IOC 26.

20 La IOM 24 puede usarse como una memoria intermedia de hospedaje para los datos que están transfiriéndose entre el IOB 22 y la interfaz periférica 32 o el controlador 14 de DMA. Así, la trayectoria de datos entre el IOB 22 y el controlador 14 de DMA, o el controlador 32 de interfaz periférica, puede ser a través de la IOM 24. La trayectoria de control (incluyendo las solicitudes de lectura/escritura, las direcciones en el espacio de direcciones del anfitrión asociadas a las solicitudes, etc.) puede ser directamente entre el IOB 22 y el controlador 14 de DMA/controlador 32 de interfaz periférica. La IOM 24 puede no estar incluida en otras formas de realización.

30 La interconexión 30 puede comprender cualquier medio de comunicación para comunicarse entre los procesadores 18A-18B, los controladores 20A-20B de memoria, la memoria temporal 28 de L2 y el IOB 22. Por ejemplo, la interconexión 30 puede ser un bus con soporte de coherencia. La interconexión 30, alternativamente, puede ser una interconexión punto a punto entre los agentes anteriores, una interconexión basada en paquetes, o cualquier otra interconexión.

35 Los circuitos 16 de interfaz comprenden generalmente circuitos configurados para comunicarse por una interfaz con el sistema 10 según cualquier protocolo de interfaz, y para comunicarse con otros componentes en el sistema 10 para recibir comunicaciones a transmitir por la interfaz, o para proporcionar comunicaciones recibidas desde la interfaz. Los circuitos de interfaz pueden configurarse para convertir las comunicaciones originadas en el sistema 10 al protocolo de interfaz, y para convertir las comunicaciones recibidas desde la interfaz para su transmisión en el sistema 10. Por ejemplo, los circuitos 16 de interfaz pueden comprender circuitos configurados para comunicarse según un protocolo de interfaz periférica (por ejemplo, el controlador 32 de interfaz periférica). Como otro ejemplo, los circuitos 16 de interfaz pueden comprender circuitos configurados para comunicarse según un protocolo de interfaz de red (por ejemplo, los MAC 34A a 34B).

45 Los MAC 34A-34B pueden comprender sistema de circuitos que implementa la funcionalidad del controlador de acceso al medio definida para interfaces de red. Por ejemplo, uno o más de los MAC 34A-34B pueden implementar el estándar Gigabit de Ethernet. Uno o más de los MAC 34A-34B pueden implementar el estándar de la Interfaz de la Unidad de Anexión Ethernet de 10 Gigabits (XAUI). Otras formas de realización pueden implementar otros estándares de Ethernet, tales como los estándares de 10 Megabits o de 100 Megabits, o cualquier otro estándar de red. En una implementación, hay 6 MAC, 4 de los cuales son MAC de Ethernet Gigabit y 2 de los cuales son MAC de XAUI. Otras formas de realización pueden tener más o menos MAC, y cualquier mezcla de tipos de MAC.

50 Entre otras cosas, los MAC 34A-34B que implementan estándares de Ethernet pueden descartar la brecha entre tramas (IFG), el preámbulo y el comienzo del delimitador de trama (SFD) de los paquetes recibidos, y pueden proporcionar los datos restantes del paquete al controlador 14 de DMA para el DMA a la memoria. Los MAC 34A-34D pueden configurarse para insertar la IFG, el preámbulo y el SFD para los paquetes recibidos desde el controlador 14 de DMA como una transferencia de DMA de transmisión, y pueden transmitir los paquetes a la PHY 36 para su transmisión.

60 El controlador 32 de interfaz periférico comprende sistema de circuitos configurado para controlar una interfaz periférica. En una forma de realización, el controlador 32 de interfaz periférica puede controlar una Interfaz Express de interconexión de componente periférico (PCI). Otras formas de realización pueden implementar otras interfaces periféricas (por ejemplo, PCI, PCI-X, bus universal en serie (USB), etc.), además, o en lugar, de la interfaz Express de PCI.

La PHY 36 puede comprender, en general, el sistema de circuitos configurado para comunicarse físicamente, por las

interfaces externas, con el sistema 10, bajo el control de los circuitos 16 de interfaz. En una forma de realización específica, la PHY 36 puede comprender un conjunto de circuitos serializadores/deserializadores (SERDES) que pueden configurarse para su uso como vías Express de PCI o como conexiones de Ethernet. La PHY 36 puede incluir el sistema de circuitos que llevan a cabo la codificación/descodificación 8b/10b para la transmisión a través de los SERDES y las memorias intermedias de sincronización 'primero en entrar, primero en salir' (FIFO), y también el sistema de circuitos que configura lógicamente los enlaces SERDES para su uso como enlaces de comunicación Express de PCI o de Ethernet. En una implementación, la PHY puede comprender 24 SERDES que pueden configurarse como vías Express de PCI o conexiones de Ethernet. Cualquier número deseado de SERDES puede configurarse como Express de PCI y cualquier número deseado puede configurarse como conexiones de Ethernet.

En la forma de realización ilustrada, los registros 38A-38C de configuración se muestran en el controlador 32 de interfaz periférica y los MAC 34A-34B. Puede haber uno o más registros de configuración en cada uno de los controladores 32 de interfaz periférica y los MAC 34A-34B. Pueden existir asimismo otros registros de configuración en el sistema 10, no mostrados en la Fig. 1. Los registros de configuración pueden usarse para configurar diversas características seleccionables de forma programable del controlador 32 de interfaz periférica y los MAC 34A-34B, habilitar o inhabilitar diversas características, configurar el controlador 32 de interfaz periférica y los MAC 34A-34B para su operación, etc. En una forma de realización descrita más adelante, los registros de configuración pueden especificarse en un descriptor de control para la reconfiguración sobre la marcha del controlador 32 de interfaz periférica y los MAC 34A-34B.

Se observa que, en diversas formas de realización, el sistema 10 puede incluir uno, o cualquier número, de cualquiera de los elementos mostrados en la Fig. 1 (por ejemplo, procesadores, controladores de memoria, memorias temporales, puentes de E/S, controladores de DMA y/o circuitos de interfaz, etc.).

Pasando ahora a la Fig. 2, se muestra un diagrama de bloques de una forma de realización del controlador 14 de DMA. Para la forma de realización de la Fig. 2, se expondrá un modelo de software descriptor para hacer transferencias de DMA. En algunas formas de realización, puede darse soporte a un modelo de software basado en registros, además, o en lugar, del modelo del descriptor. En un modelo basado en registros, cada transferencia de DMA puede programarse en el controlador 14 de DMA, y el controlador 14 de DMA puede llevar a cabo la transferencia de DMA. Al completarse la transferencia, el controlador 14 de DMA puede bien interrumpir a uno de los procesadores 18A-18B, o bien proporcionar el estado (por ejemplo, en un registro dentro del controlador 14 de DMA) que el software pueda sondear para determinar cuándo se ha completado la transferencia de DMA.

En el modelo de descriptor, el software puede establecer múltiples transferencias de DMA a llevar a cabo, usando estructuras de datos de descriptor en la memoria. En general, un descriptor de DMA puede comprender una estructura de datos en memoria que describe una transferencia de DMA. La información en el descriptor de DMA, por ejemplo, puede especificar el origen y el destino de la transferencia de DMA, el tamaño de la transferencia y diversos atributos de la transferencia. En algunos casos, el origen o el destino de la transferencia de DMA puede estar implícito. Pueden almacenarse múltiples descriptores en una estructura de datos de descriptor en la memoria (por ejemplo, en un "anillo descriptor"), y el controlador 14 de DMA puede programarse con la dirección del primer descriptor en la estructura de datos. El controlador 14 de DMA puede leer los descriptores y llevar a cabo las transferencias de DMA indicadas. Puede usarse una gran variedad de mecanismos de control para controlar la propiedad de los descriptores entre el software y el hardware. Por ejemplo, los descriptores pueden incluir bits de validación o bits de habilitación que indican al controlador 14 de DMA que la transferencia de DMA descrita en el descriptor está lista para ser llevada a cabo. Un bit de interrupción en un descriptor puede usarse para indicar que el controlador 14 de DMA ha de interrumpir al procesador 18A a 18B al final de una transferencia de DMA dada, o puede usarse un bit de fin de transferencia para indicar que el descriptor describe la última transferencia de DMA y que el controlador 14 de DMA debería hacer una pausa. Alternativamente, el controlador 14 de DMA puede implementar registros contadores de descriptor que pueden ser incrementados por el software para indicar cuántos descriptores están disponibles para que los procese el controlador 14 de DMA. El controlador 14 de DMA puede decrementar un registro contador de descriptor para indicar que se ha generado una precaptura de un descriptor. En otras formas de realización, el controlador 14 de DMA puede decrementar el registro contador de descriptor para indicar el consumo de un descriptor (es decir, la ejecución de la transferencia de DMA especificada). En otras formas de realización adicionales, el controlador 14 de DMA puede usar un registro distinto de contador procesado de descriptor para indicar cuántos descriptores han sido procesados o precapturados.

El controlador 14 de DMA puede llevar a cabo transferencias de DMA de transmisión (Tx) y transferencias de DMA de recepción (Rx). Las transferencias de DMA Tx tienen un espacio de direcciones en el anfitrión 12 como origen (por ejemplo, ubicaciones de memoria en la memoria acoplada con los controladores 20A-20B de memoria). Las transferencias de DMA Rx tienen un espacio de direcciones en el anfitrión 12 como destino. Las transferencias de DMA Tx pueden tener un circuito 16 de interfaz como destino, o pueden tener otra dirección en el espacio de direcciones del anfitrión 12 como destino (por ejemplo, para transferencias de copia de DMA). Las transferencias de DMA Tx que tienen destinos del espacio de direcciones del anfitrión pueden usar la trayectoria de datos del DMA Rx para escribir los datos de DMA leídos de la dirección de origen en la dirección de destino. Un circuito 40 de bucle de ensayo puede proporcionar el enlace entre

la trayectoria de datos del DMA Tx y la trayectoria de datos del DMA Rx. Es decir, un “circuito de bucle de ensayo” comprende circuitos locales al controlador de DMA que está acoplado para recibir datos de DMA Tx desde una trayectoria de datos del DMA de transmisión y para proporcionar datos del DMA Rx por una trayectoria de datos del DMA de recepción. Los datos proporcionados por el circuito 40 de bucle de ensayo en la trayectoria de datos del DMA de recepción pueden ser los datos recibidos desde la trayectoria de datos del DMA de transmisión (por ejemplo, para la función de copia del DMA). En algunas formas de realización, los datos proporcionados por el circuito 40 de bucle de ensayo pueden ser datos transformados por el circuito 40 de bucle de ensayo a partir de los datos recibidos. En algunas formas de realización, los datos proporcionados por el circuito 40 de bucle de ensayo pueden ser los datos recibidos por el circuito 40 de bucle de ensayo, aumentados por un resultado calculado por el circuito 40 de bucle de ensayo sobre los datos (por ejemplo, suma de control, datos de CRC, etc.). Alternativamente, los datos proporcionados por el circuito 40 de bucle de ensayo pueden ser los datos recibidos por el circuito 40 de bucle de ensayo (o bien los datos pueden no ser proporcionados), y el resultado puede almacenarse en el descriptor para la transferencia de DMA. O bien los datos transformados, o bien el resultado calculado e incluido con los datos, o bien escrito en el descriptor del DMA, pueden ser mencionados genéricamente en el presente documento como el “resultado”.

Así, en algunas formas de realización, el circuito 40 de bucle de ensayo puede configurarse para llevar a cabo una o más operaciones (o “funciones”) sobre los datos del DMA Tx a fin de producir un resultado (por ejemplo, datos de DMA transformados, o un resultado generado a partir de los datos). En la forma de realización de la Fig. 2, el circuito 40 de bucle de ensayo puede incluir una FIFO 42 de copia, un motor 44 de descarga y un circuito 46 de O exclusivo (XOR) acoplado con la trayectoria de datos de transmisión. La FIFO 42 de copia puede almacenar datos de transmisión provenientes de la trayectoria de datos del DMA Tx, para su transmisión por la trayectoria de datos del DMA Rx. En consecuencia, la FIFO 42 de copia puede llevar a cabo la operación de copia del DMA. El motor 44 de descarga puede configurarse para llevar a cabo diversas operaciones sobre los datos de DMA, produciendo bien datos transformados o bien un resultado distinto a los datos. El motor 44 de descarga puede configurarse para proporcionar cualquier conjunto deseado de operaciones, en diversas formas de realización. En una forma de realización, el motor 44 de descarga puede configurarse para llevar a cabo operaciones que ayudan al procesamiento de paquetes. Por ejemplo, se han desarrollado diversos protocolos de seguridad de red que proporcionan encriptación y/o la autenticación de paquetes. La autenticación incluye habitualmente generar un mapeo sobre algo de, o todo, el paquete. Por tanto, el motor 44 de descarga puede configurarse para llevar a cabo encriptación/decriptación y/o las funciones de mapeo sobre datos en paquetes en una transferencia de DMA. Adicionalmente, el motor 44 de descarga puede configurarse para llevar a cabo la generación/comprobación de sumas de control y/o la generación/comprobación del CRC. La suma de control y/o la protección del CRC se usan en diversos protocolos de paquetes. El circuito XOR 46 puede efectuar la operación XOR bit a bit sobre datos de DMA (por ejemplo, datos de DMA desde múltiples orígenes). El circuito XOR 46 puede usarse, p. ej., para dar soporte al procesamiento de formaciones redundantes de discos baratos (RAID) y otros tipos de procesamiento que usan funciones de XOR.

El circuito 40 de bucle de ensayo (y, más específicamente, los componentes 42, 44 y 46 de bucle de ensayo) pueden operar sobre los datos de DMA durante la transferencia de DMA que proporciona los datos de DMA al circuito 40 de bucle de ensayo. Es decir, el circuito 40 de bucle de ensayo puede al menos comenzar a llevar a cabo la operación sobre los datos de DMA mientras que la transferencia de DMA Tx proporciona el resto de los datos de DMA. En general, el resultado puede escribirse en la memoria o, más en general, en el espacio de direcciones del anfitrión (por ejemplo, como datos de DMA transformados, adosados a los datos de DMA, o a una ubicación distinta de memoria de resultados, tal como un campo en el descriptor de DMA para la transferencia de DMA Tx).

El circuito 40 de bucle de ensayo también puede incluir FIFOs para el motor 44 de descarga y el circuito XOR 46 (FIFO 48 de descarga acoplada con el motor 44 de descarga y FIFO 50 de XOR acoplada con el circuito 46 de XOR). Las FIFOs 48 y 50 pueden almacenar temporalmente datos del motor 44 de descarga y el circuito XOR 46, respectivamente, hasta que los datos de DMA puedan transmitirse por la trayectoria de datos de DMA de recepción. Se proporciona un árbitro 52 en la forma de realización ilustrada, acoplado con las FIFOs 42, 48 y 50, para arbitrar entre las FIFOs. El árbitro 52 también está acoplado con una FIFO 54 de bucle de ensayo, que puede almacenar temporalmente datos del circuito 40 de bucle de ensayo a escribir en el destino.

En la forma de realización ilustrada, el controlador 14 de DMA comprende un circuito 56 de control Tx en la trayectoria de datos de DMA Tx, y un circuito 58 de control Rx en la trayectoria de datos del DMA Rx. El circuito 56 de control Tx puede precapturar datos del anfitrión 12 para las transferencias de DMA de transmisión. En particular, el circuito 56 de control Tx puede precapturar descriptors de DMA, y puede procesar los descriptors de DMA para determinar la dirección de origen para los datos de DMA. El circuito 56 de control Tx puede entonces precapturar los datos de DMA. Si bien el término precaptura se usa para referirse al funcionamiento del circuito 56 de control Tx, las precapturas pueden generalmente ser operaciones de lectura generadas para leer el descriptor y los datos de DMA desde el espacio de direcciones del anfitrión.

El circuito 56 de control Tx transmite datos de DMA al destino. El destino, en esta forma de realización, puede ser bien uno de los circuitos 16 de interfaz o el circuito 40 de bucle de ensayo (y, más específicamente, uno entre la FIFO 42 de copia,

el motor 44 de descarga y el circuito XOR 46 en la forma de realización ilustrada). El circuito 56 de control Tx puede identificar el destino para los datos transmitidos (por ejemplo, transmitiendo un identificador de destino). Alternativamente, pueden proporcionarse trayectorias físicamente distintas entre el circuito 56 de control Tx y los circuitos 16 de interfaz y entre el circuito 56 de control Tx y los componentes 42, 44 y 46 de bucle de ensayo. El circuito 56 de control Tx puede incluir un conjunto de memorias intermedias 62 para almacenar temporalmente datos a transmitir. El circuito 56 de control Tx puede también proporcionar información diversa de control con los datos. La información de control puede incluir información del descriptor de DMA. La información de control puede incluir, para el circuito 40 de bucle de ensayo, el puntero (o punteros) de memoria(s) intermedia(s) para almacenar datos en el espacio de direcciones de destino. La información de control también puede incluir cualquier otra información de control que pueda incluirse en el descriptor de DMA y pueda ser usada por los circuitos 16 de interfaz o el circuito 14 de bucle de ensayo. Se proporcionarán ejemplos en más detalle más adelante, con respecto a la exposición del descriptor de DMA.

El circuito 58 de control Rx puede recibir datos de DMA a escribir en el espacio de direcciones del anfitrión 12, y puede generar escrituras para almacenar los datos en memoria. En una forma de realización, el software puede adjudicar memorias intermedias en memoria para almacenar los datos de DMA recibidos. El circuito 58 de control Rx puede dotarse de punteros de memoria intermedia (direcciones en el espacio de direcciones del anfitrión que identifican las memorias intermedias). El circuito 58 de control Rx puede usar el puntero de memoria intermedia para generar las direcciones para las escrituras a fin de almacenar los datos. Puede proporcionarse un motor 60 de precaptura Rx para precapturar los punteros de memoria intermedia para el circuito 58 de control Rx. El motor 60 de precaptura Rx está acoplado para proporcionar los punteros de memoria intermedia al circuito 58 de control Rx. El motor 60 de precaptura Rx puede incluir un conjunto de memorias intermedias 64 para almacenar temporalmente punteros de memorias intermedias capturados, para su uso por parte del motor 60 de precaptura Rx. De manera similar, el circuito 58 de control Rx puede incluir un conjunto de memorias intermedias 68 para almacenar temporalmente los datos de DMA recibidos, a escribir en la memoria.

En una forma de realización, el circuito 58 de control Rx puede configurarse para generar descriptores para los datos de DMA recibidos. Es decir, en lugar de hacer que el software cree descriptores de DMA para datos de DMA recibidos, el software puede adjudicar memorias intermedias para almacenar los datos de DMA y puede proporcionar los punteros de memoria intermedia. El circuito 58 de control Rx puede almacenar los datos de DMA recibidos en las memorias intermedias adjudicadas, y puede crear los descriptores para las transferencias de DMA. Los descriptores creados por el circuito 58 de control Rx pueden incluir uno o más punteros de memoria intermedia a una o más memorias intermedias que almacenan los datos de DMA recibidos, así como otra información que describe la transferencia de DMA. Una forma de realización a título de ejemplo del descriptor de DMA de recepción se muestra en la Fig. 12 y se describe en más detalle más adelante. Dado que el circuito 58 de control Rx crea los descriptores para los datos de DMA recibidos, los descriptores pueden ser más eficientes que los creados por el software. Por ejemplo, el software puede tener que crear descriptores de DMA de recepción capaces de recibir la mayor transferencia posible de DMA (o pueden requerirse múltiples descriptores para mayores transferencias), y puede tener que adjudicar memorias intermedias suficientes para almacenar la mayor transferencia posible de DMA. Por otra parte, los descriptores creados por el circuito 58 de control Rx pueden ser lo bastante grandes para la transferencia efectiva recibida (y pueden consumir bastantes memorias intermedias para almacenar los datos recibidos), pero no necesariamente mayores.

En la forma de realización ilustrada, el circuito 58 de control Rx puede recibir los datos de DMA desde un árbitro 66, que está acoplado con la FIFO 54 de bucle de ensayo y asimismo para recibir datos de DMA desde los circuitos 16 de interfaz. El árbitro 66 puede arbitrar entre la FIFO 54 de bucle de ensayo y los datos de DMA recibidos desde los circuitos 16 de interfaz para transferir datos al circuito 58 de control Rx.

Los árbitros 52 y 66 pueden implementar cualquier esquema de arbitraje deseado. Por ejemplo, un esquema basado en prioridades, un esquema de tanda circular, un esquema de tanda circular ponderado, o bien pueden usarse combinaciones de tales esquemas. En algunas formas de realización, el esquema de arbitraje puede ser programable. El esquema, o los esquemas, implementado(s) por el árbitro 52 pueden diferir del esquema, o esquemas, implementado(s) por el árbitro 66.

El circuito 56 de control Tx, el motor 60 de precaptura Rx y el circuito 58 de control Rx están acoplados con una unidad 70 de interfaz de IOM/IOB en la forma de realización ilustrada. La unidad 56 de interfaz de IOM/IOB puede comunicarse con el IOB 22 y la IOM 24 en beneficio del circuito 56 de control Tx, el motor 60 de precaptura Rx y el circuito 58 de control Rx. La unidad 70 de interfaz de IOM/IOB puede recibir solicitudes de lectura y escritura desde el circuito 56 de control Tx, el motor 60 de precaptura Rx y el circuito 58 de control Rx, y puede comunicarse con el IOB 22 y la IOM 24 para satisfacer esas solicitudes.

En particular, la unidad 70 de interfaz de IOM/IOB puede recibir solicitudes de lectura para descriptores y para datos de DMA desde el circuito 56 de control Tx, y solicitudes de lectura para la memoria que almacena punteros de memoria intermedia desde el motor 60 de precaptura Rx, y puede llevar las solicitudes al IOB 22. El IOB 22 puede indicar qué

5 entrada de la IOM 24 almacena una línea de memoria temporal de datos que incluyen los datos solicitados (a continuación de leer los datos desde el espacio de direcciones del anfitrión o de la IOC 26, por ejemplo, o bien los datos ya pueden estar en la IOM 24 provenientes de una solicitud anterior), y la interfaz 70 de IOM/IOB puede leer los datos desde la IOM 24 y proporcionarlos al circuito 56 de control Tx o al motor 60 de precaptura Rx. La unidad 70 de interfaz de IOM/IOB también puede recibir solicitudes de escritura desde el circuito 58 de control Rx, y puede almacenar los datos de escritura en la IOM 24 (en una entrada adjudicada para los datos de escritura por el IOB 22). Una vez que se acumula una línea de memoria temporal de datos en la IOM 24 (o bien se completa la transferencia de DMA, cualquiera que llegue primero), la unidad 70 de interfaz de IOM/IOB puede informar al IOB 22 y puede proporcionar una dirección a la cual ha de escribirse la línea de memoria temporal (obtenida del puntero de memoria intermedia a la memoria intermedia en la que se está escribiendo).

15 En una forma de realización, el controlador 14 de DMA puede dar soporte a diversos canales para transferencias de DMA de transmisión y transferencias de DMA de recepción. Puede darse soporte a cualquier número de canales, en diversas formas de realización. Por ejemplo, en una implementación, pueden proporcionarse 20 canales de DMA de transmisión y pueden proporcionarse 64 canales de DMA de recepción. Cada canal puede ser una trayectoria lógica independiente de datos desde un origen a un destino. Los canales pueden adjudicarse según lo deseado por el software.

20 Más específicamente, cada canal de transmisión puede asignarse a uno de los circuitos 16 de interfaz, o a uno de los circuitos componentes 42, 44 ó 46 de bucle de ensayo. No todos los canales de transmisión deben estar en uso (es decir, algunos canales de transmisión pueden estar inhabilitados). El circuito 56 de control Tx puede precapturar descriptores de DMA y datos de DMA canal por canal. Es decir, el circuito 56 de control Tx puede generar independientemente precapturas para cada canal que tenga descriptores de DMA disponibles para el procesamiento. El circuito 56 de control Tx puede seleccionar entre las precapturas generadas para transmitir solicitudes de lectura a la unidad 70 de interfaz de IOM/IOB.

25 Cada canal de recepción puede asignarse a uno de los circuitos 16 de interfaz. No todos los canales de recepción deben estar en uso (es decir, algunos canales de recepción pueden estar inhabilitados). El circuito 58 de control Rx puede recibir el número de canal con los datos recibidos. El circuito 40 de bucle de ensayo puede proporcionar un puntero de memoria intermedia proveniente del descriptor de DMA para el DMA, y el circuito 58 de control Rx puede usar el puntero de memoria intermedia para escribir los datos de DMA en el espacio de direcciones del anfitrión. Los circuitos 16 de interfaz pueden ser programables con los canales asignados, o bien pueden emplear el filtrado de paquetes para determinar un canal. Los circuitos 16 de interfaz pueden proporcionar el número de canal con los datos de DMA, y el circuito 58 de control Rx puede usar un puntero de memoria intermedia proporcionado desde el motor 60 de precaptura Rx para el canal, para escribir los datos de DMA en el espacio de direcciones del anfitrión.

30 El controlador 14 de DMA puede incluir diversos registros 38D-38H de configuración, según se muestra en la Fig. 2. Los registros 38D-38H de configuración pueden ser programables para habilitar/inhabilitar diversas características programables del controlador 14 de DMA y/o para configurar las características programables, según lo mencionado anteriormente. Por ejemplo, los registros 38D de configuración en el circuito 56 de control Tx pueden incluir direcciones de anillos descriptores para cada canal, así como contadores de descriptores que indican el número de descriptores disponibles. Los registros 38D de configuración pueden incluir adicionalmente asignaciones de canales de transmisión a circuitos 16 de interfaz y funciones componentes de bucle de ensayo. Varias otras configuraciones por canal y configuraciones no vinculadas a canales pueden almacenarse en los registros 38D de configuración. De manera similar, los registros 38E de configuración pueden almacenar direcciones de anillos de punteros de memoria intermedia para cada circuito 16 de interfaz, contadores de anillos de memoria intermedia, etc., así como diversas configuraciones no vinculadas con canales. Los registros 38F de configuración pueden almacenar diversas configuraciones de DMA de recepción. Los registros 38G de configuración pueden almacenar configuraciones para el circuito 40 de bucle de ensayo como un todo, así como configuraciones para cada circuito componente, según se desee. Los registros 38G de configuración también pueden almacenar configuraciones para el árbitro 52 (por ejemplo, selección del esquema de arbitraje, configuración de programación para el esquema de arbitraje seleccionado). Los registros 38H de configuración pueden almacenar configuraciones de arbitraje para el árbitro 66.

35 Se observa que, mientras que el circuito 56 de control Tx implementa la precaptura para obtener descriptores y datos de DMA, otras formas de realización pueden no implementar la precaptura. Así, en general, puede haber un motor 56 Tx o un circuito 56 de control Tx configurado para llevar a cabo transferencias de DMA de transmisión (y transferencias de DMA al circuito 40 de bucle de ensayo).

40 Se observa que la presente descripción se refiere a memorias intermedias y punteros de memorias intermedias para transferencias de DMA. Una memoria intermedia a la que apunta un puntero de memoria intermedia (en contraposición a memorias intermedias de hardware tales como 62, 64 y 68) puede comprender una región de memoria contigua. El software puede adjudicar la región de memoria para almacenar datos de DMA (bien para la transmisión o bien como una región para recibir datos de DMA). El puntero de memoria intermedia puede comprender una dirección de la región de

memoria en el espacio de direcciones del anfitrión. Por ejemplo, el puntero de memoria intermedia puede apuntar a la base de la región de memoria o al límite de la región de memoria.

5 Pasando ahora a la Fig. 3, se muestra un diagrama de bloques de una forma de realización del motor 44 de descarga. En la forma de realización ilustrada, el motor 44 de descarga incluye una memoria intermedia 80 de entrada, una memoria intermedia 82 de salida, un conjunto de circuitos 84A-84D de seguridad, un generador 86 de CRC y un generador 88 de sumas de control. La memoria intermedia 80 de entrada está acoplado con el circuito 56 de control Tx y con los circuitos 84A-84D de seguridad, el generador 86 de CRC y el generador 88 de sumas de control. La memoria intermedia 82 de salida está acoplado con los circuitos 84A-84D de seguridad, el generador 86 de CRC y el generador 88 de sumas de control. La memoria intermedia 82 de salida está acoplada asimismo con la FIFO 48 de descarga. El circuito 84A de seguridad se muestra en mayor detalle en la Fig. 3 para una forma de realización, y los circuitos 84B-84D de seguridad pueden ser similares. El circuito 84A de seguridad incluye un circuito 90 de mapeo y un circuito 92 de cifrado. Tanto el circuito 90 de mapeo como el circuito 92 de cifrado están acoplados con la memoria intermedia 80 de entrada y la memoria intermedia 82 de salida. Adicionalmente, la salida del circuito 90 de mapeo está acoplada como una entrada al circuito 92 de cifrado, y la salida del circuito 92 de cifrado está acoplada como una entrada al circuito 90 de mapeo en una configuración de "mariposa".

Los circuitos 84A-84D de seguridad pueden configurarse para llevar a cabo diversas operaciones a fin de descargar funciones de seguridad del procesamiento de paquetes. En particular, los circuitos 84A-84D de seguridad pueden configurarse para llevar a cabo encriptación/decriptación (denominado colectivamente como cifrado, o funciones de cifrado) y las funciones de mapeo que se incluyen en diversas especificaciones de paquetes seguros (por ejemplo, el protocolo seguro de Internet (IPSec) o la capa de cuencas seguras (SSL)).

Habitualmente, la comunicación que usa un protocolo seguro de paquetes incluye una sesión de negociación en la cual los puntos extremos se comunican los protocolos que pueden usar, los esquemas de seguridad a los que dan soporte, el tipo de encriptación y de mapeo, el intercambio de claves o certificados, etc. Luego hay una fase de transferencia a granel que usa los protocolos, encriptación, etc., acordados. Durante la transferencia a granel, los paquetes pueden recibirse en el anfitrión 12 (por ejemplo, mediante la trayectoria de DMA de recepción desde uno de los circuitos 16 de interfaz). El software puede consultar estructuras de datos en la memoria para obtener las claves, los algoritmos de encriptación, etc., y preparar una transferencia de DMA a través del motor 44 de descarga para decriptar y/o autenticar el paquete. De manera similar, el software puede preparar un paquete para la transmisión segura y usar una transferencia de DMA a través del motor 44 de descarga para encriptar y/o autenticar el paquete.

El circuito 90 de mapeo puede implementar diversas funciones de mapeo que pueden usarse en la autenticación de paquetes. Habitualmente, el mapeo se calcula para al menos una parte del paquete, y el resultado del mapeo se incluye en el paquete. Cuando el paquete se recibe en su destino, puede comprobarse el mapeo para detectar si algunos campos en el paquete han sido cambiados (y detectar así si el paquete fue modificado en tránsito desde su origen). En una forma de realización, las siguientes funciones de mapeo pueden disponer de soporte en el circuito 90 de mapeo: Digesto de Mensaje 5 (MD-5)/algoritmo-1 de mapeo seguro (SHA-1), y código mapeado de autenticación de mensaje (HMAC). Otras formas de realización pueden implementar el SHA-2. Otras formas de realización pueden implementar cualquier otro conjunto de funciones de mapeo, incluyendo subconjuntos o superconjuntos de las funciones anteriores y de otras funciones.

El circuito 92 de cifrado puede configurarse para llevar a cabo funciones de cifrado. Según la especificación segura de paquete, la función de cifrado puede aplicarse al menos a una parte del paquete, que incluye posiblemente los datos de mapeo. Cualquier conjunto de funciones de cifrado puede disponer de soporte en diversas formas de realización. Por ejemplo, en una forma de realización, pueden implementarse los siguientes algoritmos de encriptación/decriptación en el circuito 92 de cifrado: estándar de encriptación de datos (DES), estándar triple de encriptación de datos (3DES), el estándar avanzado de encriptación (AES), Kasumi, código 4 aducido de Ron (ARC4) y/o código 4 de Ron (RC4).

En algunos casos, si se están usando funciones tanto de autenticación como de cifrado, la encriptación se lleva a cabo primero al preparar un paquete para su transmisión, y luego se lleva a cabo el mapeo de autenticación sobre los datos encriptados (por ejemplo, IPSec). En otros casos, el mapeo de autenticación se lleva a cabo primero, y la encriptación del paquete (incluyendo los datos de mapeo) se lleva a cabo en segundo lugar (por ejemplo, SLL). En cualquier caso, el mapeo de autenticación y la encriptación se lleva a cabo en el orden opuesto sobre un paquete recibido.

Los circuitos 84A-84D de seguridad pueden dar soporte a cualquier orden de cifrado y mapeo de datos en una única transferencia de DMA, mediante la conexión de mariposa entre los circuitos 90 y 92. Es decir, si el cifrado ha de llevarse a cabo primero, los datos proporcionados al circuito 84A de seguridad pueden enrutarse al circuito 92 de cifrado, y la salida del circuito 92 de cifrado puede enrutarse a la entrada del circuito 90 de mapeo para calcular la función de mapeo sobre los datos encriptados (o decriptados). Si ha de llevarse a cabo primero el mapeo, los datos proporcionados al circuito 84A de seguridad pueden enrutarse al circuito 90 de mapeo, y la salida del circuito 90 de mapeo puede enrutarse a la entrada

del circuito 92 de cifrado. Los circuitos 84A-84D de seguridad también dan soporte a la ejecución de sólo el mapeo o de sólo la función de cifrado en una transferencia de DMA dada. La información de control desde el descriptor de DMA para la transferencia de DMA dirigida a los circuitos 84A-84D de seguridad puede controlar el enrutamiento de datos a través de los circuitos 84A-84D de seguridad.

5 La forma de realización ilustrada muestra 4 circuitos 84A-84D de seguridad. Otras formas de realización pueden incluir cualquier número de circuitos de seguridad, incluyendo un circuito de seguridad. En una forma de realización, los circuitos 84A-84D de seguridad pueden estar sincronizados al doble de la frecuencia del reloj de sistema usado en el sistema 10 y pueden recibir dos operaciones por ciclo de reloj de sistema (una llevada a cabo en la primera mitad del ciclo de reloj del sistema y la otra en la segunda mitad del ciclo de reloj de sistema). Así, puede haber 8 circuitos lógicos de seguridad que pueden ser seleccionados por el software para llevar a cabo funciones de seguridad.

15 El generador 86 de CRC puede configurarse para generar datos de CRC sobre los datos proporcionados en una transferencia de DMA que especifica generación de CRC. La generación de CRC también puede usarse para comprobar los datos de CRC de un paquete recibido. Por ejemplo, los datos de CRC generados en el generador 86 de CRC pueden compararse con los correspondientes datos de CRC en el paquete recibido. Alternativamente, los datos de CRC en el paquete recibido pueden incluirse en la transferencia de DMA a través del generador 86 de CRC, y el resultado puede comprobarse con respecto a un valor predeterminado para detectar errores en el paquete recibido. En algunas formas de realización, puede haber más de un generador 86 de CRC. Además, el generador, o generadores, 86 de CRC puede(n) sincronizarse al doble de la frecuencia de reloj de sistema, de forma similar a los circuitos 84A-84D de seguridad, para proporcionar más generadores de CRC lógico que los que se proporcionan físicamente en el motor 44 de descarga. En una forma de realización específica, puede haber 4 de los generadores 86 de CRC, sincronizados al doble de la frecuencia de reloj de sistema, para proporcionar un número igual de unidades lógicas (8) a los circuitos 84A-84D de seguridad.

25 El generador 88 de sumas de control puede configurarse para generar una suma de control sobre los datos proporcionados en una transferencia de DMA que especifica generación de sumas de control. La generación de sumas de control también puede usarse para comprobar los datos de suma de control provenientes de un paquete recibido. Por ejemplo, los datos de suma de control generados en el generador 88 de sumas de control pueden compararse con la correspondiente suma de control en el paquete recibido. Alternativamente, los datos de suma de control en el paquete recibido pueden incluirse en la transferencia de DMA a través del generador 88 de sumas de control, y el resultado puede comprobarse con respecto a un valor predeterminado para detectar errores en el paquete recibido. En algunas formas de realización, puede haber más de un generador 88 de sumas de control.

35 La memoria intermedia 80 de entrada puede almacenar temporalmente datos proporcionados por el circuito 56 de control Tx hasta que el circuito 84A-84D, 86 u 88 de destino pueda operar sobre los datos. Los circuitos 84A-84D, 86 y 88 pueden emitir datos a la memoria intermedia 82 de salida para que se escriban en la FIFO 48 de descarga. En otras formas de realización, la memoria intermedia 80 de entrada y/o la memoria intermedia 82 de salida pueden no estar incluidas.

40 La Fig. 4 es un diagrama de bloques que ilustra una forma de realización de un modelo para el DMA que puede implementarse en una forma de realización del sistema 10. Como se ha mencionado anteriormente, una transferencia de DMA puede tener lugar desde un espacio o interfaz de direcciones de origen (bloque 100) hasta un espacio o interfaz de direcciones de destino (bloque 102). La transferencia de DMA está representada por el bloque 104 en la Fig. 4. Optativamente, una transferencia de DMA puede incluir la ejecución de una o más operaciones, o funciones (bloque 106) sobre los datos de DMA para producir un resultado. El resultado se devuelve al bloque 104 de DMA y puede proporcionarse al espacio de direcciones de destino. En algunos casos, el resultado pueden ser datos de DMA transformados que pueden escribirse en el espacio de direcciones de destino. En otros casos, el resultado puede ser distinto a los datos de DMA y puede aumentar los datos de DMA (por ejemplo, agregarse al final de los datos de DMA) o bien puede almacenarse en una ubicación distinta (por ejemplo, en el descriptor de DMA para la transferencia de DMA).

50 La Fig. 4 puede ilustrar una transferencia individual de DMA, en algunos casos. En otros casos, pueden usarse múltiples transferencias de DMA para completar el modelo de la Fig. 4. Por ejemplo, las funciones 106 pueden ser llevadas a cabo por el circuito 40 de bucle de ensayo (o componentes del mismo, para la forma de realización de la Fig. 2). Así, en la forma de realización ilustrada, si se desea que se lleve a cabo una función sobre una transferencia de DMA que tiene como destino un circuito 16 de interfaz, pueden llevarse a cabo dos transferencias de DMA. Una primera transferencia de DMA, tal vez desde un espacio de direcciones de origen a un espacio de direcciones de destino, que especifica las funciones deseadas a llevar a cabo. A continuación, puede llevarse a cabo una segunda transferencia de DMA, usando el espacio de direcciones de destino de la primera transferencia de DMA como el origen y el circuito de interfaz deseado como destino.

60 Por ejemplo, un paquete puede ser preparado por el software para la transmisión mediante uno de los MAC 34A-34B, y el paquete puede almacenarse en memoria en el anfitrión 12. Puede desearse la transmisión segura usando encriptación y/o

mapeo, y por ello el software puede establecer una primera transferencia de DMA desde la primera región de memoria a una segunda región de memoria en el anfitrión 12, y la primera transferencia de DMA puede especificar encriptación y/o mapeo en el motor 44 de descarga. El software también puede preparar una segunda transferencia de DMA desde la segunda región de memoria al MAC 34A-34B de destino. De manera similar, un paquete puede recibirse como un DMA en la memoria en el anfitrión 12, y el software puede establecer una transferencia de DMA a través del motor 44 de descarga para comprobar el mapeo y/o decriptar el paquete.

Otras formas de realización pueden permitir que se apliquen funciones según se transmiten los datos a un circuito 16 de interfaz. En una forma de realización, por ejemplo, puede darse soporte a la generación parcial de sumas de control en el IOB 22, para generar la suma de control incluida en la cabecera del TCP (Protocolo de Control de Transporte) de paquetes del TCP. Cuando se especifica una transferencia de DMA para un tal paquete, el IOB 22 puede acumular los datos del paquete en la IOM 24 y puede generar la suma parcial de control para cada línea de memoria temporal. La suma final de control puede generarse en el controlador 14 de DMA, usando las sumas parciales de control para cada línea de memoria temporal proveniente del IOB 22 y los datos sobre los cuales no se calcularon las sumas parciales de control, y el controlador 14 de DMA puede insertar la suma de control calculada en la cabecera del TCP. Otras formas de realización pueden permitir que la salida del circuito 40 de bucle de ensayo se enrute directamente a los circuitos 16 de interfaz como parte de la transferencia individual de DMA en la cual se llevan a cabo las funciones y se transmiten los datos. Además, otras formas de realización pueden permitir que los datos de la trayectoria de datos de DMA Rx ingresen al circuito 40 de bucle de ensayo para llevar a cabo funciones sobre los datos de DMA recibidos durante la transferencia de DMA de recepción.

Pasando a continuación a la Fig. 5, se muestra un diagrama de bloques de una región 110 de memoria que almacena estructuras de datos de descriptores y estructuras de datos de punteros de memoria intermedia. En la forma de realización de la Fig. 5, las estructuras de datos de descriptores incluyen un conjunto de anillos descriptores 112A-112N. Puede haber un anillo descriptor para cada canal de DMA con soporte por parte del controlador 14 de DMA (por ejemplo, canal 0 a canal N en la Fig. 5). Es decir, puede haber una correspondencia de uno a uno entre los canales de DMA y los anillos descriptores, y las transferencias de DMA para un canal dado de DMA pueden tener descriptores correspondientes en el anillo descriptor 112A-112N asignado a ese canal. Adicionalmente, en la forma de realización de la Fig. 5, las estructuras de datos de punteros de memoria intermedia pueden incluir un conjunto de anillos 114A-114M de punteros de memoria intermedia. Puede haber un anillo de punteros de memoria intermedia por circuito 16 de interfaz (por ejemplo, circuitos 0 a M de interfaz en la Fig. 5, donde M+1 puede ser el número de circuitos 16 de interfaz). Es decir, puede haber una correspondencia de uno a uno entre circuitos de interfaz y anillos descriptores, y los punteros de memoria intermedia usados para los DMA recibidos por esa interfaz pueden tomarse del anillo 114A-114M de punteros de memoria intermedia asignado a ese circuito de interfaz.

Cada anillo descriptor 112A-112N puede comprender un conjunto de descriptores para el correspondiente canal de DMA. Para canales de DMA de transmisión, los descriptores pueden procesarse en el orden incluido dentro del anillo, desde el primer descriptor en el anillo hasta el último, y luego retomar circularmente el primer descriptor en el anillo después de que se ha procesado el último descriptor. Así, en un momento dado, cualquier descriptor en el anillo puede verse como el "descriptor actual", es decir, el próximo a procesar. El software puede controlar el número de descriptores que están disponibles para el procesamiento por el canal de DMA de una gran variedad de maneras, según lo mencionado anteriormente. En consecuencia, si hay descriptores disponibles en un canal dado de DMA de transmisión (en el correspondiente anillo descriptor), el controlador 14 de DMA puede llevar a cabo las transferencias de DMA especificadas (arbitrando los recursos con otros canales de DMA). Para los canales de DMA de recepción en la presente forma de realización, los descriptores en el correspondiente anillo descriptor pueden consumirse según se reciben las transferencias de DMA por ese canal. El controlador 14 de DMA puede escribir en el descriptor actual el puntero, o punteros, de memoria intermedia usado(s) para almacenar los datos de DMA recibidos, así como otra información referida a la transferencia de DMA, tal como la información de estado de transferencia.

Otras formas de realización pueden usar otras estructuras de datos (por ejemplo, listas enlazadas de descriptores). La dirección base de cada anillo descriptor 112A-112N puede proporcionarse al controlador 14 de DMA (por ejemplo, programarse en los registros 38D ó 38F de configuración, según que el canal sea un canal de transmisión o de recepción). Asimismo pueden programarse otros atributos del anillo descriptor 112A-112N (por ejemplo, la extensión). En algunas formas de realización, los descriptores en un anillo dado pueden ser de un tamaño fijo, de modo tal que un descriptor dado pueda estar en un desplazamiento fijo a partir de la dirección base del anillo. En otras formas de realización, los descriptores pueden ser de tamaño variable. En otras formas de realización adicionales, los descriptores pueden ser de tamaño fijo o de tamaño variable, según un atributo programable en los registros 38D ó 38F de configuración. El atributo puede ser programable canal por canal, o bien puede programarse para los canales en su totalidad.

Cada anillo 114A-114M de punteros de memoria intermedia comprende punteros de memoria intermedia que apuntan a memoria intermedias en memoria adjudicada por el software para su uso a fin de almacenar datos de DMA provenientes de transferencias de DMA Rx desde la correspondiente interfaz. De manera similar a los anillos descriptores 112A-112N,

el software puede hacer que los punteros de memoria intermedia en los anillos 114A-114M de punteros de memoria intermedia estén disponibles para el controlador 14 de DMA de cualquier manera deseada. La dirección base del anillo de punteros de memoria intermedia para cada interfaz puede programarse en el controlador 14 de DMA (por ejemplo, en los registros 38E de configuración en el motor 60 de precaptura Rx, en la forma de realización de la Fig. 2) y, en cualquier momento, uno de los punteros de memoria intermedia en el anillo de punteros de memoria intermedia puede ser el próximo a ser consumido para la correspondiente interfaz.

Proporcionando los anillos 114A-114M de punteros de memoria intermedia asociados a los circuitos de interfaz, en lugar de los canales de DMA, el software puede adjudicar memorias intermedias al menor número de circuitos de interfaz, en lugar de al mayor número de canales de DMA, en algunas formas de realización. La adjudicación de memoria, en algunos casos, puede ser más eficiente. A los circuitos de interfaz que están gestionando más tráfico pueden adjudicarse más memorias intermedias, sin que el software tenga conocimiento previo de por qué canales se recibirá ese tráfico. Según se reciben datos de DMA desde una interfaz dada, los datos pueden almacenarse en las memorias intermedias adjudicadas a esa interfaz y los punteros de memoria intermedia pueden escribirse en el descriptor para el canal por el cual se reciben los datos de DMA. El descriptor puede estar en uno de los anillos descriptores 112A-112N, según qué canal de DMA esté asociado a la transferencia de DMA.

Los anillos 114A-114M de punteros de memoria intermedia también pueden incluir un campo de tamaño (Sz en la Fig. 5) para cada puntero de memoria intermedia. El campo de tamaño puede indicar el tamaño de la memoria intermedia al que apunta el correspondiente puntero de memoria intermedia. En consecuencia, el software puede adjudicar memorias intermedias de distintos tamaños en base, p. ej., a la cantidad de memoria disponible, el tamaño esperado de las transferencias de DMA por una interfaz dada, etc.

Pasando ahora a la Fig. 6, se muestra un diagrama de flujo que ilustra el funcionamiento de una forma de realización del motor 60 de precaptura Rx para un circuito de interfaz dado. El motor 60 de precaptura Rx puede incluir circuitos que implementan el funcionamiento mostrado en la Fig. 6 para cada circuito de interfaz, funcionando en paralelo e independientemente. Si bien los bloques se muestran en un orden específico en la Fig. 6, para facilitar la comprensión, los bloques pueden implementarse en paralelo en sistema de circuitos lógicos combinatorios que implementen el funcionamiento mostrado en la Fig. 6. En algunas formas de realización, uno o más de los bloques, o el diagrama de flujo en total, puede secuenciarse sobre múltiples ciclos de reloj.

El motor 60 de precaptura Rx puede determinar si hay punteros de memoria intermedia disponibles para el circuito de interfaz (en el anillo 114A-114M de punteros de memoria intermedia correspondiente al circuito de interfaz) (bloque 120 de decisión) y si se necesitan punteros de memoria intermedia para el circuito de interfaz (bloque 122 de decisión). Si al menos un puntero de memoria intermedia está disponible y es necesario (bloques 120 y 122 de decisión, rama "sí"), el motor 60 de precaptura Rx puede generar una solicitud de precaptura para leer el puntero, o punteros, de memoria intermedia del anillo 114A-114M de punteros de memoria intermedia en la memoria del anfitrión 12 (bloque 124).

Los punteros de memoria intermedia pueden estar generalmente "disponibles" si hay punteros de memoria intermedia en el correspondiente anillo 114A-114M de punteros de memoria intermedia que no han sido precapturados por el motor 60 de precaptura Rx. Los punteros de memoria intermedia pueden ser insertados en el anillo 114A-114M de punteros de memoria intermedia por el software, y el software puede indicar que están disponibles de cualquiera de las maneras anteriormente mencionadas (por ejemplo, usando bits de validación en las entradas del anillo de punteros de memoria intermedia, incrementando un contador de anillo de punteros de memoria intermedia, de manera similar al contador de descriptores de DMA descrito anteriormente, etc.). Los punteros de memoria intermedia pueden verse asimismo como "necesarios" de una gran variedad de formas. Por ejemplo, si se habilita un canal de DMA de recepción y no hay punteros de memoria intermedia precapturados para el canal, puede ser "necesario" un puntero de memoria intermedia. En algunas formas de realización, el motor 60 de precaptura Rx puede ser programable para indicar un cierto número de punteros de memoria intermedia que deberían precapturarse, o bien un número mínimo y máximo de punteros de memoria intermedia que deberían precapturarse. El motor 60 de precaptura Rx puede generar solicitudes de precaptura para punteros de memoria intermedia, a fin de intentar precapturar el número programado de punteros de memoria intermedia.

Como se ha mencionado anteriormente, el funcionamiento de la Fig. 6 puede llevarse a cabo en paralelo para cada circuito de interfaz habilitado. Si se genera concurrentemente más de una solicitud de precaptura, el motor 60 de precaptura Rx también puede incluir sistema de circuitos para seleccionar entre las solicitudes de precaptura. Por ejemplo, en una forma de realización, el motor 60 de precaptura Rx puede implementar un esquema de prioridad fija entre las solicitudes de precaptura. En otras formas de realización, el motor 60 de precaptura Rx puede seleccionar la solicitud de precaptura correspondiente a un circuito de interfaz para el cual estén actualmente precapturados y listos los mínimos punteros de memoria intermedia. Como otro ejemplo, el motor 60 de precaptura Rx puede ponderar las solicitudes en base a qué circuito de interfaz tiene la mayor diferencia entre el número de punteros de memoria intermedia actualmente precapturados y el número deseado de punteros de memoria intermedia, para ese circuito de interfaz. También pueden usarse mecanismos de selección basados en la tanda circular o en la prioridad, y estos mecanismos pueden incluir la

ponderación programable entre los circuitos de interfaz, si se desea. También pueden usarse mecanismos de prevención de inanición, tales como las temporizaciones por interfaz, para garantizar que se precapturen punteros para cada circuito de interfaz.

5 El motor 60 de precaptura Rx puede ser informado por el circuito 70 de interfaz de IOM/IOB cuando estén disponibles en la IOM 24, para su lectura, punteros de memoria intermedia precapturados. El motor 60 de precaptura Rx, en algunas formas de realización, puede leer algunos de, o todos, los punteros de la IOM 24 en las memorias intermedias 64 y puede proporcionar los punteros de memoria intermedia precapturados al circuito 58 de control Rx, según sea necesario.

10 Pasando ahora a la Fig. 7, se muestra un diagrama de flujo que ilustra el funcionamiento de una forma de realización del circuito 58 de control Rx en respuesta a la recepción de datos. Los datos pueden recibirse bien desde el circuito 40 de bucle de ensayo o bien desde un circuito 16 de interfaz. Si bien los bloques se muestran en un orden específico en la Fig. 7, para facilitar la comprensión, los bloques pueden implementarse en paralelo en sistema de circuitos lógicos combinatorios que implementen el funcionamiento mostrado en la Fig. 7. En algunas formas de realización, uno o más de los bloques, o el diagrama de flujo en su totalidad, pueden secuenciarse sobre múltiples ciclos de reloj.

15 El circuito 58 de control Rx puede determinar si una memoria intermedia está ya en uso para recibir los datos de DMA (bloque 130 de decisión). Una memoria intermedia puede estar en uso si ha sido seleccionado anteriormente para almacenar los datos de DMA y no está lleno aún. El circuito 58 de control Rx puede configurarse para mantener memorias intermedias activas para cada canal de DMA Rx y uno o más punteros de memoria intermedia para el circuito 40 de bucle de ensayo. Alternativamente, el circuito 40 de bucle de ensayo puede proporcionar el puntero de memoria intermedia con cada transferencia de datos, y el circuito 58 de control Rx puede no mantener los punteros para el circuito 40 de bucle de ensayo. Si una memoria intermedia no está en uso (bloque 130 de decisión, rama "no"), el circuito 48 de control Rx puede seleccionar el próximo puntero de memoria intermedia para identificar la memoria intermedia a utilizar (bloque 132). Para transferencias de DMA desde el circuito 40 de bucle de ensayo, el próximo puntero de memoria intermedia es proporcionado por el circuito 40 de bucle de ensayo. Para transferencias de DMA desde los circuitos 16 de interfaz, el próximo puntero de memoria intermedia puede ser proporcionado desde el motor 60 de precaptura Rx y el circuito 58 de control Rx puede indicar el consumo del puntero al motor 60 de precaptura Rx.

20 En cualquier caso, el circuito 58 de control Rx puede escribir los datos recibidos en la memoria intermedia (bloque 134). Es decir, el circuito 58 de control Rx puede generar una solicitud de escritura al circuito 70 de interfaz de IOM/IOB para escribir los datos de DMA en la memoria. El circuito 58 de control Rx puede monitorizar el número de octetos escritos en la memoria intermedia para determinar cuándo está lleno la memoria intermedia, al menos para los datos recibidos desde los circuitos 16 de interfaz. En algunas formas de realización, el tamaño de la memoria intermedia puede ser proporcionado por el circuito 40 de bucle de ensayo y el circuito 58 de control Rx puede monitorizar asimismo plenitud de memoria intermedia para transferencias de bucle de ensayo. Si la memoria intermedia está llena, o si la transferencia de datos está indicada por el origen (el circuito 40 de bucle de ensayo o el circuito 16 de interfaz) como el fin de la transferencia de DMA (bloque 136 de decisión, rama "sí"), el circuito 58 de control Rx puede escribir el puntero de memoria intermedia en el descriptor para la transferencia de DMA (bloque 138). Para transferencias de bucle de ensayo, puede no haber un descriptor de DMA para los datos recibidos, ya que los punteros de memoria intermedia pueden ser punteros de memoria intermedia de destino provenientes del descriptor de DMA de origen, y el bloque 138 puede no ejecutarse para tales transferencias. Si la transferencia de datos está indicada como el final de la transferencia de DMA (bloque 140 de decisión, rama "sí"), el circuito 58 de control Rx puede llevar a cabo diversos procesamientos de fin de transferencia (bloque 142). Por ejemplo, para una transferencia de DMA desde los circuitos 16 de interfaz, el circuito 58 de control Rx puede generar diversos estados para la transferencia de DMA, generar la cabecera del descriptor de DMA para la transferencia, y escribir el descriptor de DMA en el anillo descriptor 112A-112N correspondiente al canal de DMA Rx por el cual se recibieron los datos. Adicionalmente, el circuito 58 de control Rx puede señalar el final de la transferencia al circuito 70 de IOM/IOB. La señalización del final de la transferencia puede informar al circuito 70 de IOM/IOB que no se suministrarán más datos por ese canal. Si una línea incompleta de memoria temporal es actualizada por el DMA, el circuito 70 de IOM/IOB puede hacer que los datos actualizados se remitan a la memoria (por ejemplo, en la IOC 26, en algunas formas de realización, para evitar un ciclo de lectura-modificación-escritura por la interconexión 30).

25 Pasando luego a la Fig. 8, se muestra un diagrama de flujo que ilustra el funcionamiento de una forma de realización del circuito 56 de control Tx para un canal dado de DMA Tx. El circuito 56 de control Tx puede incluir circuitos que implementan el funcionamiento mostrado en la Fig. 8 para cada canal de DMA Tx, funcionando en paralelo e independientemente. Si bien los bloques se muestran en un orden específico en la Fig. 8 para facilitar la comprensión, los bloques pueden implementarse en paralelo en sistema de circuitos lógicos combinatorios que implementen el funcionamiento mostrado en la Fig. 8. En algunas formas de realización, uno o más de los bloques, o el diagrama de flujo en su totalidad, pueden secuenciarse sobre múltiples ciclos de reloj.

30 El circuito 56 de control Tx puede determinar si están disponibles para precaptura descriptores para el canal (en el anillo descriptor 112A-112N correspondiente al canal) (bloque 150 de decisión) y si se necesitan descriptores para el canal

(bloque 152 de decisión). Si al menos un descriptor está disponible y es necesario (bloques 150 y 152 de decisión, rama "sí"), el circuito 56 de control Tx puede generar una solicitud para leer los descriptores provenientes del anillo descriptor 112A-112N en la memoria del anfitrión 12 (bloque 154).

5 De manera similar a que los punteros de memoria intermedia estén "disponibles", según lo anteriormente descrito con respecto a la Fig. 6, los descriptores pueden estar generalmente "disponibles" si hay descriptores en el correspondiente anillo descriptor 112A-112N que no han sido precapturados por el circuito 56 de control Tx. Los descriptores para un canal de DMA Tx pueden ser insertados en el anillo descriptor 112A-112N por el software, y el software puede indicar que están disponibles de cualquiera de las maneras anteriormente mencionadas (por ejemplo, usando bits de validación en las
10 entradas del anillo descriptor, incrementando un contador de anillo descriptor, etc.). Los descriptores pueden asimismo verse como "necesarios" de una gran variedad de maneras. Por ejemplo, si un canal de DMA Tx está habilitado y no hay descriptores precapturados para el canal, puede ser "necesario" un descriptor. En algunas formas de realización, el circuito 56 de control Tx puede precapturar descriptores mientras haya sitio en la IOM 24 y/o en las memorias intermedias 62 para almacenar los descriptores. En otras formas de realización, el circuito 56 de control Tx puede ser programable
15 para indicar un cierto número de descriptores que deberían precapturarse, o un número mínimo y máximo de descriptores que deberían precapturarse. El circuito 56 de control Tx puede generar solicitudes de precaptura para descriptores, a fin de intentar precapturar el número programado de descriptores.

20 El circuito 56 de control Tx puede ser informado por el circuito 70 de interfaz de IOM/IOB cuando estén disponibles, para su lectura, descriptores precapturados en la IOM 24. El circuito 56 de control Tx, en algunas formas de realización, puede leer algunos de, o todos, los descriptores provenientes de la IOM 24 en las memorias intermedias 62.

25 El circuito 56 de control Tx puede determinar si hay disponibles para precaptura datos de DMA para el canal (a transmitir por el canal) (bloque 156 de decisión) y si se necesitan datos de DMA para el canal (bloque 158 de decisión). Si los datos de DMA están disponibles y son necesarios (bloques 156 y 158 de decisión, rama "sí"), el circuito 56 de control Tx puede generar una solicitud para leer los datos de DMA del espacio de direcciones del anfitrión 12 (por ejemplo, de las ubicaciones de memoria en el anfitrión 12) (bloque 160).

30 Los datos de DMA pueden considerarse como disponibles para la precaptura si el circuito 56 de control Tx tiene un descriptor para ser procesado (por ejemplo, el descriptor es el siguiente a procesar proveniente del anillo descriptor para el canal), los datos del descriptor están en las memorias intermedias 62 o la IOM 24, y los datos del descriptor describen una transferencia de DMA válida a llevar a cabo. Los datos de DMA pueden ser necesarios si se han transmitido datos anteriores de DMA por el canal (o se transmitirán pronto). En algunas formas de realización, el circuito 56 de control Tx puede ser programable en cuanto a cuántos datos de DMA han de precapturarse en cualquier momento dado, y pueden
35 ser necesarios datos de DMA si se ha precapturado, y no transmitido aún, menos de la cantidad deseada de datos de DMA. En algunas formas de realización, el esquema de arbitraje entre los canales de transmisión también puede afectar si los datos de DMA son necesarios (por ejemplo, si el canal no gana el arbitraje durante un periodo de tiempo relativamente grande, los datos de DMA pueden no ser aún necesarios, ya que no pueden transmitirse hasta que gane el arbitraje).

40 Se observa que el funcionamiento ilustrado por los bloques 156, 158 y 160 puede ser independiente del funcionamiento de los bloques 150, 152 y 154 (fuera de que los descriptores precapturados se usen para determinar si los datos de DMA están disponibles). En consecuencia, el sistema de circuitos que implementa los bloques 156, 158 y 160 puede ser independientes del sistema de circuitos que implementa los bloques 150, 152 y 154, y puede evaluar en paralelo con tal sistema de circuitos.

45 Como se ha mencionado anteriormente, el funcionamiento de la Fig. 8 puede llevarse a cabo en paralelo para cada canal habilitado de DMA Tx. Si se genera concurrentemente más de una solicitud de precaptura, el circuito 56 de control Tx puede también incluir circuitos para seleccionar entre las solicitudes de precaptura. Por ejemplo, el circuito 56 de control Tx puede seleccionar la solicitud de precaptura correspondiente al canal de DMA Tx para el cual estén actualmente precapturados y listos los mínimos descriptores o la mínima cantidad de datos de DMA. Como otro ejemplo, el circuito 56
50 de control Tx puede ponderar las solicitudes en base a qué canal de DMA Tx tiene la mayor diferencia entre los descriptores, o datos de DMA, actualmente precapturados y el número deseado de descriptores, o cantidad de datos de DMA, para ese canal. También pueden usarse mecanismos de selección en base a la tanda circular o a la prioridad, y estos esquemas pueden incluir la ponderación programable entre los canales, si se desea. También pueden usarse
55 mecanismos de prevención de inanición, tales como expiraciones de temporizadores por canal, para garantizar que los descriptores y los datos de DMA sean precapturados para cada canal habilitado.

60 El circuito 56 de control Tx puede ser informado por el circuito 70 de interfaz de IOM/IOB cuando estén disponibles, para su lectura, los datos de DMA precapturados en la IOM 24. El circuito 56 de control Tx, en algunas formas de realización, puede leer algunos de, o todos, los datos de DMA de la IOM 24 en las memorias intermedias 62. Adicionalmente, el circuito 56 de control Tx puede transmitir los datos de DMA precapturados al destino, según lo ilustrado para una forma de realización en la Fig. 9. Si bien los bloques se muestran en un orden específico en la Fig. 9, para facilitar la comprensión,

los bloques pueden implementarse en paralelo en sistema de circuitos lógicos combinatorios que implementen el funcionamiento mostrado en la Fig. 9. En algunas formas de realización, uno o más de los bloques, o el diagrama de flujo en su totalidad, pueden secuenciarse sobre múltiples ciclos de reloj.

5 Si el canal de DMA Tx está asignado al circuito 40 de bucle de ensayo (o, más específicamente, a un componente de bucle de ensayo en el circuito 40 de bucle de ensayo – bloque 162 de decisión, rama “sí”), el circuito 56 de control Tx puede extraer el puntero, o punteros, de memoria intermedia de destino del descriptor de DMA para la transferencia, y por transmitir con los datos de DMA (bloque 164). La información de tamaño para cada puntero de memoria intermedia de destino también puede proporcionarse de modo tal que el circuito 58 de control Rx pueda estar informado del tamaño de las memorias intermedias. El circuito 40 de bucle de ensayo puede recibir el puntero, o punteros, de memoria intermedia de destino y proporcionarlo(s) con los datos a escribir en memoria (bien los datos de DMA originales, p. ej., en una transferencia de copia de DMA, o bien los datos de DMA transformados, p. ej., si se lleva a cabo el cifrado). Alternativamente, puede proporcionarse un puntero al descriptor de DMA si el resultado de bucle de ensayo ha de escribirse en el descriptor de DMA (por ejemplo, un resultado de mapeo, un resultado de CRC o un resultado de suma de control). El circuito 56 de control Tx puede transmitir los datos al circuito 16 de interfaz de destino o al componente de bucle de ensayo (bloque 166) junto con el número de canal, los punteros (si los hubiera) y la información de control (si la hubiera). La información de control también puede extraerse del descriptor de DMA (por ejemplo, información de selección de la(s) operación(es) de bucle de ensayo deseada(s), información de control para los circuitos de interfaz, etc.).

20 El funcionamiento de la Fig. 9 puede llevarse a cabo en paralelo para cada canal de DMA Tx. Es decir, el circuito 56 de control Tx puede incluir circuitos independientes que implementen el funcionamiento de la Fig. 9 en paralelo para cada canal de DMA Tx. Puede usarse un mecanismo de arbitraje para arbitrar entre canales que tienen datos de DMA para transmitir, y para seleccionar un canal que transmita por la trayectoria de datos de transmisión a los circuitos 16 de interfaz y el circuito 40 de bucle de ensayo.

25 Los anillos descriptores 112A-112N han sido descritos anteriormente como almacenadores de diversos descriptores. En una forma de realización, los anillos descriptores 112A-112N pueden almacenar tanto descriptores de transferencia (que describen una transferencia de DMA desde un origen a un destino) como descriptores de control. Los descriptores de control no pueden especificar una transferencia de DMA pero, en cambio, pueden codificar información de control para el canal. Por ejemplo, los descriptores de control pueden usarse para especificar la configuración (por ejemplo, el contenido de uno o más registros 38A-38G de configuración). Así, los descriptores de control pueden usarse para reconfigurar el controlador 14 de DMA, los componentes del mismo u otros componentes del sistema 10 sobre la marcha, entre transferencias de DMA. En consecuencia, el software puede, por ejemplo, establecer uno o más descriptores de transferencia para llevar a cabo transferencias de DMA con una primera configuración, establecer uno o más descriptores de control para cambiar la configuración, establecer uno o más descriptores de transferencia adicionales para llevar a cabo transferencias de DMA con una segunda configuración y a continuación dejar disponibles los descriptores por el canal de DMA como una unidad. Las transferencias de DMA, la reconfiguración y las transferencias adicionales de DMA pueden ser llevadas a cabo sin intervención del procesador.

30 También se contemplan otros tipos de descriptores de control, que controlan al controlador 14 de DMA sin causar explícitamente una transferencia de DMA. Por ejemplo, se contempla un descriptor de activador temporal que causa un retardo temporal por el canal de DMA cuando se procesa el descriptor de activador temporal. Por ejemplo, el sistema 10 puede incluir un temporizador que puede programarse con un valor de retardo, y el procesamiento del próximo descriptor en el anillo descriptor puede retardarse hasta que el temporizador expire. La expiración del temporizador puede generar un suceso de activación para el controlador 14 de DMA, para “despertar” al controlador 14 de DMA. El usuario puede programar el temporizador con el valor de retardo, o bien el valor de retardo puede incluirse en el descriptor de activador temporal, en diversas formas de realización. En otras formas de realización, el controlador 14 de DMA puede implementar el temporizador. El descriptor de activador temporal puede tener una gran variedad de usos. Por ejemplo, los descriptores de activador temporal pueden intercalarse con descriptores de copia de DMA que copian desde un espacio de direcciones de origen a un espacio de direcciones de destino. El espacio de direcciones de origen puede comprender uno o más registros de E/S mapeados a la memoria, u otros registros de estado del sistema. Así, las transferencias repetidas de copia de DMA pueden usarse para leer los registros y escribir los resultados en los destinos, retardadas por el descriptor de activador temporal. Las transferencias de copia de DMA pueden tomar el lugar de un bucle de sondeo de software en los procesadores 18A-18B, liberando los procesadores para otras operaciones.

55 La Fig. 10 es un diagrama de bloques de una forma de realización del anillo descriptor 112A correspondiente al canal 0, que ilustra un ejemplo de inserción de un descriptor de control dentro de los descriptores de transferencia en un anillo descriptor. Puede insertarse más de un descriptor de control a la manera de la Fig. 10 (bien consecutivamente o bien en distintos puntos en el anillo), según se desee.

60 En la Fig. 10, se muestran 2 descriptores 170A-170B de transferencia, seguidos por un descriptor 172 de control, seguido por 2 descriptores 170C-170D más de transferencia. Así, para este ejemplo, las dos transferencias de DMA especificadas

por los descriptores 170A-170B serían llevadas a cabo por el circuito 56 de control Tx, seguidas por el procesamiento del descriptor 172 de control. A continuación del procesamiento del descriptor 172 de control, se llevarían a cabo las dos transferencias de DMA especificadas por los descriptores 170C-170D.

5 La Fig. 11 es un diagrama de flujo que ilustra el procesamiento adicional que una forma de realización del circuito 56 de control Tx puede emplear para los descriptores de control insertados con descriptores de transferencia en un anillo descriptor correspondiente a un canal de DMA Tx. El circuito 56 de control Tx puede incluir sistema de circuitos que implementen el funcionamiento mostrado en la Fig. 11 para cada canal de DMA Tx, funcionando en paralelo e independientemente. Si bien los bloques se muestran en un orden específico en la Fig. 11, para facilitar la comprensión, los bloques pueden implementarse en paralelo en sistema de circuitos lógicos combinatorios que implementen el funcionamiento mostrado en la Fig. 11. En algunas formas de realización, uno o más de los bloques, o el diagrama de flujo en su totalidad, pueden secuenciarse sobre múltiples ciclos de reloj.

15 El circuito 56 de control Tx puede precapturar un descriptor proveniente del anillo descriptor (bloque 174), suponiendo que uno esté disponible y sea necesario para el canal (por ejemplo, según lo descrito con respecto a la Fig. 8). El circuito 56 de control Tx puede procesar el descriptor cuando es devuelto al circuito 56 de control Tx, y puede determinar si el descriptor es un descriptor de control (bloque 176 de decisión). Por ejemplo, la cabecera del descriptor puede incluir un campo de tipo que especifique si el descriptor es un descriptor de transferencia o un descriptor de control. Si el descriptor es un descriptor de transferencia (bloque 176 de decisión, rama "no"), el circuito 56 de control Tx puede procesar el descriptor de transferencia (bloque 177). Por ejemplo, el procesamiento del descriptor de transferencia puede ser similar a la descripción anterior con respecto a las Figs. 8 y 9.

Si el descriptor es un descriptor de control (bloque 176 de decisión, rama "sí"), el circuito de control Tx puede determinar si el descriptor de control es un descriptor de activador temporal (bloque 178 de decisión). Un descriptor de activador temporal puede indicarse mediante una codificación en la cabecera del descriptor de control. Alternativamente, un descriptor de activador temporal puede especificar un valor a cargar en un registro 38A-38G de configuración, de manera similar a otros descriptores de configuración. El valor cargado puede especificar el retardo deseado. El circuito 56 de control Tx puede detectar el activador temporal descodificando la dirección de registro del registro de configuración cargado. Si el descriptor de control es un descriptor de activador temporal (bloque 178 de decisión, rama "sí"), el circuito 56 de control Tx puede retardar el procesamiento del próximo descriptor proveniente del anillo descriptor hasta que expire el retardo especificado por el descriptor de activador temporal (bloque 180). Si el descriptor de control no es un descriptor de activador temporal (bloque 178 de decisión, rama "no"), el circuito 56 de control Tx puede reconfigurar usando los valores del descriptor de control (bloque 182). Por ejemplo, el descriptor de control puede comprender una dirección de registro que identifica un registro 38A-38G de configuración y un valor a escribir en el registro de configuración. El circuito 56 de control Tx puede hacer que tenga lugar la escritura. Alternativamente, el descriptor de control puede incluir un puntero de memoria intermedia a una memoria intermedia de memoria que almacena una lista de direcciones de registro y valores a escribir en esos registros. Así, puede llevarse a cabo una cantidad relativamente grande de configuraciones usando un descriptor de control. Las instrucciones para escribir en cada registro de configuración, ejecutadas en los procesadores 18A-18B, pueden eliminarse usando un descriptor de control, en algunas formas de realización.

40 Las Figs. 12-16 ilustran ejemplos de descriptores de diversos tipos según una forma de realización del controlador 14 de DMA. Generalmente, los descriptores comprenden una cabecera, optativamente un campo de datos para almacenar un resultado (por ejemplo, un resultado generado por el circuito 40 de bucle de ensayo) y uno o más punteros de memoria intermedia que apuntan a memorias intermedias que almacenen datos de DMA (punteros de memoria intermedia de origen) o a memorias intermedias que pueden usarse para almacenar datos de DMA (punteros de memoria intermedia de destino).

50 En la presente forma de realización, los descriptores varían en base a si son DMA de recepción o transmisión, o a la función a llevar a cabo por el circuito 40 de bucle de ensayo, si se selecciona. Los descriptores de recepción se usan para transferencias de DMA Rx, y otros tipos de descriptores son usados por las transferencias de DMA Tx y las funciones de bucle de ensayo. El controlador 14 de DMA (y más específicamente el circuito 56 de control Tx, en una forma de realización) puede determinar el formato de los descriptores en un anillo descriptor para un canal dado de DMA Tx, en base a la asignación de ese canal a los circuitos 16 de interfaz o a una función en el circuito 40 de bucle de ensayo.

55 En las Figs. 12-16, se ilustran diversos campos en detalle (por ejemplo, el campo de cabecera en particular). Si bien se muestra cierta información en las Figs. 12-16, no está concebido excluir el uso de otra información además de la ilustrada, o además de un subconjunto de la ilustrada, o como una alternativa a lo ilustrado. Puede incluirse diversa información adicional, en diversas implementaciones, según se desee.

60 La Fig. 12 es un diagrama de bloques de una forma de realización de un descriptor 190 de recepción. El descriptor 190 de recepción puede ser del formato de los descriptores generados por el circuito 58 de control Rx y ser escrito en los anillos descriptores 112A-112N que corresponden a canales de DMA Rx. En la forma de realización de la Fig. 12, el descriptor

190 de recepción incluye un campo 192 de cabecera (ilustrado en vista expandida en la Fig. 12 para una forma de realización), un campo optativo 194 de resultado de mapeo, y uno o más campos 196A-196N de punteros de memoria intermedia. Cada campo 196A-196N de puntero a de memoria intermedia incluye un campo de tamaño que puede codificarse con el tamaño de la memoria intermedia, y un campo puntero codificado con el puntero a la memoria intermedia. El circuito 58 de control Rx puede configurarse para escribir, en los campos 196A-196N de punteros de memoria intermedia, los punteros de memoria intermedia usados, provenientes de los anillos 114A-114M de punteros de memoria intermedia, para almacenar los datos de DMA recibidos. El campo 194 de resultado de mapeo puede usarse para almacenar un resultado de mapeo.

La vista expandida del campo 196 de cabecera de recepción incluye un campo 192A de tipo, un campo 192B de estilo, un campo 192C de reserva de mapeo, un campo 192D de contador de memoria intermedia, un campo 192E de longitud de paquete y un campo 192F de estado. El campo 192A de tipo puede codificar el tipo de descriptor, p. ej., descriptor de control o descriptor de transferencia. El campo 192A de tipo (y otros campos similares de tipo descritos más adelante para otros descriptores) puede identificar el descriptor como un descriptor de transferencia, excepto por el descriptor de control mostrado en la Fig. 16. El campo 192B de estilo puede codificar el estilo del descriptor, que se refiere a si los punteros de memoria intermedia en los campos 196A-196N de punteros de memoria intermedia incluyen punteros tanto de origen como de destino, o sólo punteros de destino (ya que el origen es un circuito de interfaz para transferencias de DMA Rx). El campo 192C de reserva de mapeo puede usarse para indicar si el campo optativo 194 de resultado de mapeo se incluye o no en el descriptor. El campo 192D de contador de memoria intermedia pueden codificarse con un contador del número de memorias intermedias usadas para almacenar los datos de DMA recibidos y, por tanto, el número de campos 196A-196N de punteros de memoria intermedia. El campo 192E de longitud de paquete puede codificarse con la longitud del paquete almacenado en las memorias intermedias (por ejemplo, en octetos). El campo 192F de estado puede comprender diversos estados de la transferencia, que pueden incluir el estado generado por el circuito 58 de control Rx (por ejemplo, indicaciones de error) y/o el estado proporcionado por el circuito 16 de interfaz de origen. El circuito 58 de control Rx puede escribir la cabecera 192 de recepción al terminar una transferencia de DMA.

La Fig. 13 es un diagrama de bloques de una forma de realización de un descriptor 200 de transmisión. El descriptor 200 de transmisión puede ser del formato de los descriptores usados por el circuito 56 de control Tx para transferencias de DMA Tx a circuitos 16 de interfaz, en particular, a los MAC 34A-34B, y puede ser escrito por el software en los anillos descriptores 112A-112N que corresponden a canales de DMA Tx asignados a los circuitos 16 de interfaz. En la forma de realización de la Fig. 13, el descriptor 200 de transmisión incluye un campo 202 de cabecera (ilustrado en vista expandida en la Fig. 13 para una forma de realización) y uno o más campos 204A-204N de punteros de memoria intermedia. Cada campo 204A-204N de puntero de memoria intermedia incluye un campo de tamaño que puede codificarse con el tamaño de la memoria intermedia, y un campo de puntero codificado con el puntero a la memoria intermedia. El circuito 56 de control Tx puede configurarse para leer los campos 204A-204N de punteros de memoria intermedia, a fin de precapturar los datos de DMA de las memorias intermedias, para su transmisión.

La vista expandida del campo 202 de cabecera de transmisión incluye un campo 202A de tipo, un campo 202B de estilo, un campo 202C de configuración de MAC, un campo 202D de longitud de paquete y un campo 202E de información de paquete. El campo 202A de tipo y el campo 202B de estilo pueden ser similares al campo 192A de tipo y el campo 192B de estilo descritos anteriormente. El campo 202C de configuración de MAC puede codificarse con diversas informaciones de configuración de MAC específicas para el paquete para el MAC 34A-34B que es el destino de la transferencia de DMA. Por ejemplo, el campo 202C de configuración de MAC puede incluir la configuración de la red virtual de área local (VLAN) (por ejemplo, nada, insertar, eliminar o modificar), la configuración del CRC (por ejemplo, nada, insertar CRC, rellenar CRC, modificar CRC) y si se modifica o no la dirección de origen del MAC. El campo 202D de longitud de paquete puede codificarse con la longitud del paquete almacenado en las memorias intermedias (por ejemplo, en octetos). El campo 202E de información de paquete puede codificarse con diversas informaciones que describen el paquete (por ejemplo, longitud de cabecera de IP, longitud de cabecera de Ethernet, tipo de paquete (TCP/UDP), habilitación de suma de control, etc.).

La Fig. 14 es un diagrama de bloques de una forma de realización de un descriptor 210 de copia. El descriptor 210 de copia puede ser del formato de los descriptores usados por el circuito 56 de control Tx para las transferencias de copia de DMA (desde una región de memoria en el anfitrión 12 a otra región de memoria en el anfitrión 12) usando la FIFO 42 de copia. Así, el descriptor 210 de copia puede usarse en los anillos descriptores 112A-112N que corresponden a los canales de DMA Tx asignados a la FIFO 42 de copia. En la forma de realización de la Fig. 14, el descriptor 210 de copia incluye un campo 212 de cabecera (ilustrado en vista expandida en la Fig. 14 para una forma de realización) y uno o más campos 214A-214N de punteros de memoria intermedia. Cada campo 214A-214N de puntero de memoria intermedia incluye un campo de tamaño que puede codificarse con el tamaño de la memoria intermedia, y un campo de puntero codificado con el puntero a la memoria intermedia. Adicionalmente, en esta forma de realización, cada campo 214A-214N de puntero de memoria intermedia incluye un campo de origen/destino (S/D) que identifica el puntero bien como un puntero de origen (que localiza una memoria intermedia que almacena datos de DMA de origen) o bien como un puntero de destino (que localiza una memoria intermedia en la cual los datos de DMA han de almacenarse). El circuito 56 de control Tx puede configurarse para leer los campos 214A-214N de punteros de memoria intermedia a fin de precapturar los datos de DMA

desde las memorias intermedias de origen para su transmisión, y para proporcionar los punteros de destino a la FIFO 42 de copia para su transmisión al circuito 58 de control Rx.

5 En una forma de realización, puede haber más de un puntero de origen para un puntero de destino dado en el descriptor 210 de copia. El controlador 14 de DMA puede copiar datos desde las memorias intermedias de origen, en el orden enumerado en el descriptor 210 de copia, a la memoria intermedia de destino. Así, el controlador 14 de DMA puede dar soporte a la recogida de datos dispersos desde múltiples regiones de memoria a una región de memoria de destino en la operación de copia. De manera similar, en una forma de realización, puede haber más de un puntero de destino para un puntero de origen dado en el descriptor 210 de copia. En tales formas de realización, puede darse soporte a la dispersión de los datos provenientes de la memoria intermedia de origen.

15 La vista expandida del campo 212 de cabecera de transmisión incluye un campo 212A de tipo, un campo 212B de estilo, un campo 212C de tipo de origen, un campo 212D de tipo de destino y un campo 212E de longitud de bloque lógico. El campo 212A de tipo y el campo 212B de estilo pueden ser similares al campo 192A de tipo y el campo 192B de estilo descritos anteriormente. El campo 212C de tipo de origen y el campo 212D de tipo de destino pueden codificarse para indicar cómo deberían modificarse el puntero, o punteros, de memoria intermedia de origen y el puntero, o punteros, de memoria intermedia de destino según avanza la transferencia de DMA. Por ejemplo, cada puntero de memoria intermedia puede ser de uno de los siguientes tipos, en una forma de realización: incremento secuencial; decremento secuencial; o fijo (con diversos anchos fijos, p. ej., 1, 2, 4, 8 ó 16 octetos). El incremento secuencial indica que la dirección se incrementa después de cada transmisión de datos en la cantidad de datos transmitidos. El decremento secuencial es similar, pero la dirección se decrementa. El incremento secuencial o el decremento secuencial pueden usarse para regiones de memoria donde los datos se escriben en ubicaciones secuenciales de memoria. La opción fija puede usarse si una dirección está mapeada por memoria a un registro o puerto de un dispositivo, y el ancho puede ser el ancho de cada transmisión al registro/dispositivo. El campo 212C de tipo de origen también puede tener una codificación para el cero, y puede usarse para escribir un bloque de ceros en el destino. El campo 212D de tipo de destino también puede tener una codificación sólo para captura, en la cual los datos de DMA de origen son precapturados, pero no escritos en un destino. El campo de longitud de bloque lógico puede usarse, en algunas formas de realización, para indicar la longitud de un bloque de DMA lógico que puede abarcar múltiples descriptores de DMA. Es decir, la operación de DMA lógico puede especificarse efectivamente usando múltiples descriptores, y la longitud del bloque de DMA lógico puede ser la longitud de la operación de DMA lógico (por ejemplo, la suma total de la transferencia de datos sobre los múltiples descriptores).

35 El circuito XOR 46 puede usar descriptores que son similares al descriptor 200 de DMA de transmisión. Pueden asignarse múltiples canales al circuito XOR 46, y los descriptores en cada uno de los canales pueden especificar uno de las fuentes de XOR. El primer canal también puede especificar el destino para el resultado del XOR (una memoria o memorias intermedias de destino).

40 La Fig. 15 es un diagrama de bloques de una forma de realización de un descriptor 220 de descarga. El descriptor 220 de descarga puede ser del formato de los descriptores usados por el circuito 56 de control Tx para las transferencias de DMA que especifican el motor 44 de descarga. Así, el descriptor 220 de descarga puede usarse en los anillos descriptores 112A-112N que corresponden a canales de DMA Tx asignados al motor 44 de descarga. En la forma de realización de la Fig. 15, el descriptor 220 de descarga incluye un campo 222 de cabecera (ilustrado en vista expandida en la Fig. 15 para una forma de realización), un campo optativo 224 de reserva de resultado y uno o más campos 226A-226N de punteros de memoria intermedia. Cada campo 226A-226N de puntero de memoria intermedia incluye un campo de tamaño que puede codificarse con el tamaño de la memoria intermedia y un campo de puntero codificado con el puntero a la memoria intermedia. Adicionalmente, en esta forma de realización, cada campo 226A-226N de puntero de memoria intermedia incluye un campo de origen/destino (S/D) que identifica el puntero bien como un puntero de origen (que localiza una memoria intermedia que almacena datos de DMA de origen) o bien como un puntero de destino (que localiza una memoria intermedia en la cual han de almacenarse los datos de DMA). El circuito 56 de control Tx puede configurarse para leer los campos 226A-226N de punteros de memoria intermedia a fin de precapturar los datos de DMA desde las memorias intermedias de origen y para identificar memorias intermedias de destino, si las hubiera. Si los datos de DMA transformados son el resultado del motor 44 de descarga, puede haber punteros de destino para los datos de DMA transformados. Si se genera un resultado distinto a los datos de DMA (por ejemplo, para el almacenamiento en el campo 224 de reserva de resultado), puede no haber ningún puntero de destino en algunos casos y los datos de DMA pueden no escribirse en un destino.

55 La vista expandida del campo 222 de cabecera de descarga incluye un campo 222A de tipo, un campo 222B de estilo, un campo 222C de reserva de resultado, un campo 222D de modalidad de cifrado, un campo 222E de número de función, un campo 222F de longitud de bloque lógico y un campo 222G de control de descarga. El campo 222A de tipo y el campo 222B de estilo pueden ser similares al campo 192A de tipo y el campo 192B de estilo descritos anteriormente, y el campo 222F de longitud de bloque lógico puede ser similar al campo 212E de longitud de bloque lógico descrito anteriormente. El campo 222C de reserva de resultado puede codificarse para indicar si se incluye o no el campo 224 de reserva de resultado en el descriptor 220 de cabecera de descarga, y también puede indicar el tamaño del campo 224 de reserva de

resultado (por ejemplo, 64 bits, 192 bits ó 320 bits, en una forma de realización). El campo 224 de reserva de resultado puede usarse para almacenar el resultado generado por el motor 44 de descarga, si el resultado no son datos de DMA transformados o se genera además de los datos de DMA transformados. El campo 222D de modalidad de cifrado puede codificarse con la modalidad deseada para que el motor 44 de descarga procese los datos, si los hubiera. La modalidad de cifrado puede incluir: nada, sólo firma (por ejemplo, CRC o suma de control en los generadores 86 u 88), sólo encriptación, sólo decriptación, encriptación y mapeo (con opciones para mapear primero o encriptar primero), o decriptación y mapeo (con opciones para mapear primero o decriptar primero). El campo 222E de número de función puede codificarse con el número de función para aquellas modalidades que usan los circuitos 84A- 84D de seguridad (por ejemplo, una entre ocho funciones lógicas formadas a partir de los circuitos 84A-84D de seguridad, según lo mencionado anteriormente). El campo 222G de control de descarga puede incluir información de control adicional para la transferencia de DMA. Por ejemplo, la longitud de cada uno entre la cabecera de firma, la cabecera de cifrado, el epílogo de cifrado y el tamaño de mapeo puede incluirse en el campo 222G de control de descarga. De manera similar, el algoritmo seleccionado de encriptación/decriptación (cifrado), el algoritmo de mapeo y la modalidad de cifrado de bloque pueden codificarse en el campo 222G de control de descarga. Otros diversos bits de control pueden incluirse asimismo en el campo 222G de control de descarga.

La Fig. 16 es un diagrama de bloques de una forma de realización de un descriptor 230 de control. El descriptor 230 de control puede ser del formato de los descriptores de control usados por el circuito 56 de control Tx en cualquier canal de DMA. En la forma de realización de la Fig. 16, el descriptor 230 de control incluye un campo 232 de cabecera (ilustrado en vista expandida en la Fig. 16 para una forma de realización) y un campo 234 de datos o de tamaño/puntero. El campo 234 de datos o de tamaño/puntero puede bien almacenar datos en línea a escribir en un registro 38A-38G de configuración o bien un puntero de memoria intermedia (y tamaño de memoria intermedia) que apunta a una memoria intermedia que almacena una lista de direcciones de registro y datos de configuración a escribir en los registros.

La vista expandida del campo 232 de cabecera de control incluye un campo 232A de tipo, un campo 232B de indicación en línea/puntero, un campo 232C de Identificador de interfaz/función, y un campo 232D de dirección de registro. El campo 232A de tipo puede ser similar al campo 192A de tipo descrito anteriormente, que identifica el descriptor 230 como un descriptor de control. El campo 232B de indicación en línea/puntero, puede codificarse para indicar si el campo 234 de datos o de tamaño/puntero almacena datos en línea a escribir en un registro de configuración, un puntero a una memoria intermedia de direcciones de registro y datos de configuración, o un suceso. El campo 232C de Identificador de interfaz/función se codifica para identificar el destino de los datos de configuración (por ejemplo, un circuito de interfaz, una función en el motor 44 de descarga, etc.). Las direcciones de registro pueden ser locales al Identificador de interfaz/función en esta forma de realización. El campo 232D de dirección de registro puede codificarse con una dirección de registro si el descriptor 230 de control tiene datos en línea.

Pasando ahora a la Fig. 17, se muestra un diagrama de bloques de una forma de realización del generador 88 de sumas de control. En la forma de realización de la Fig. 17, el generador 88 de sumas de control incluye una pluralidad de sumadores de acarreo y resguardo (CSA) 3:2 240A-240G de 16 bits, un sumador completo 242 y un acumulador (ACC) 244 de 16 bits. El generador 88 de sumas de control está acoplado para recibir una entrada de 128 bits (16 octetos) (por ejemplo, desde el circuito 56 de control Tx). La entrada de 128 bits se divide en partes de 16 bits que se suministran como entradas a los CSA 240A-240C. Adicionalmente, la salida del acumulador 24 está acoplada como una entrada al CSA 240C. Cada CSA 240A-240C emite una suma y un término de acarreo, marcados "S" y "C" en la Fig. 17. Las salidas de suma y acarreo de los CSA 240A-240C se entran en los CSA 240D-240E. Las salidas de suma y acarreo del CSA 240D y la salida de acarreo del CSA 240E se entran en el CSA 240F. La salida de suma del CSA 240E y las salidas de suma y acarreo del CSA 240F se entran en el CSA 240G. Las salidas de suma y acarreo del CSA 240G se entran en el sumador completo 242, que produce la suma a almacenar en el acumulador 244.

Los CSA reciben entradas de N bits y producen términos de salida de suma y acarreo que no reflotan el acarreo de posición a posición. Es decir, los bits de salida de la suma son la suma de los bits de entrada en esa posición de bit, independientemente de los bits en las otras posiciones de bits. Cada bit de suma puede ser resultado de la operación XOR de los bits de entrada en esa posición de bit. Los bits de salida de acarreo son el acarreo de la suma de una posición de bit dada, independientemente del acarreo en otras posiciones de bit. Aritméticamente, el término de acarreo se ve como el acarreo a introducir en la siguiente posición de bit más significativa. En consecuencia, el bit de acarreo en la posición de bit menos significativo es cero, y hay lógicamente N+1 bits de acarreo provenientes de un CSA de N bits.

Como se ilustra con las flechas 246, en cada punto donde se proporciona un término de acarreo emitido desde un CSA como entrada a otro CSA o al sumador completo 242, el bit de acarreo más significativo se "desplaza circularmente" al bit menos significativo del término de acarreo. Es decir, el bit más significativo del término de acarreo se borra de la posición de bit más significativo y se inserta en la posición de bit menos significativo, en lugar del bit menos significativo, que es siempre cero. De esta manera, las entradas a cada CSA y al sumador completo 242 son siempre de 16 bits, y todos los bits de acarreo están representados en las entradas. Para los CSA de 16 bits mostrados en la Fig. 17, el término de acarreo emitido puede ser lógicamente un valor de 17 bits con el bit menos significativo igual a 0. El término de acarreo

emitido es ingresado al próximo CSA (o al sumador completo 242) de la siguiente manera:

$$\text{Entrada}[15:0] = \text{Acarreo}[15:1] \parallel \text{Acarreo}[16] \text{ (y se descarta Acarreo}[0] = 0)$$

5 El desplazamiento circular del bit de acarreo más significativo al bit de acarreo menos significativo puede lograrse a través del enrutamiento cableado entre la salida del término de acarreo de un CSA y la entrada al siguiente CSA.

En algunas formas de realización, el generador 88 de sumas de control, según se ilustra en la Fig. 17, puede generar una suma de control de 16 bits en un paso, almacenando la suma de control en el acumulador 244. El acumulador 244 también puede ser de 16 bits en esta forma de realización, ya que los acarreos han sido desplazados circularmente y sumados por los CSA 240A-240G y el sumador completo 242 según se acumula cada entrada de 128 bits en la suma de control.

15 Para la generación de sumas de control se está generando una suma complementaria a unos. Puede mostrarse que el desplazamiento circular de los bits de acarreo emitidos por los CSA desde el bit más significativo al bit menos significativo genera una suma correcta complementaria a unos.

Los CSA 240A-240G pueden verse como un conjunto de niveles de CSA. El primer nivel de los CSA (los CSA 240A-240C) tienen entradas acopladas con las entradas al generador de sumas de control y una entrada acoplada con la salida del acumulador. En otros niveles, las entradas están acopladas con las salidas de los CSA en un nivel precedente (o niveles, p. ej., en el caso del CSA 240G). En cada nivel, las salidas de acarreo de los niveles precedentes tienen sus bits más significativos desplazados circularmente al bit menos significativo a ingresar a ese nivel.

25 La presente forma de realización puede usar los CSA de 16 bits para producir una suma de control de 16 bits que se use en TCP, IP e ICMP. Otras formas de realización pueden usar sumas de control, mayores o menores, y pueden usarse mayores o menores CSA en tales formas de realización. Así, en general, puede usarse un CSA 3:2 de N bits. De manera similar, si bien se usa una entrada de 128 bits en la forma de realización ilustrada, otras formas de realización pueden usar cualquier número de bits de entrada. El número de niveles de los CSA puede cambiarse en consecuencia.

30 Si bien la forma de realización de la Fig. 17 se muestra como implementada en el generador 88 de sumas de control, pueden usarse circuitos similares en el IOB 22 para la generación parcial de sumas de control (según lo descrito anteriormente) y en los circuitos 16 de interfaz para completar la suma parcial de control.

35 Pasando ahora a la Fig. 18, se muestra un diagrama de bloques de una forma de realización del sumador completo 242. En la forma de realización ilustrada, el sumador completo 242 puede incluir un circuito XOR 250 y un par de circuitos 252A-252B de generación de acarreo. El circuito XOR 250 y los circuitos 252A-252B de generación de acarreo están acoplados para recibir las entradas "a" y "b" al sumador completo (que son las salidas de suma y acarreo del CSA 240G en la Fig. 17, con el bit más significativo de la salida de acarreo eliminado e insertado en el bit menos significativo). El generador 252A de acarreo está acoplado para recibir una entrada de acarreo (C_{entrada}) de cero, y el generador 252B de acarreo está acoplado para recibir una entrada de acarreo de uno. La salida de acarreo (C_{salida}) del generador 252A de acarreo se proporciona como el control de selección de un multiplexor (mux) 254, que tiene las salidas de acarreo de los generadores 252A-252B de acarreo como entradas. La salida del mux 254 está acoplada como una entrada a un segundo circuito XOR 256, que también tiene la salida del circuito XOR 250 como una entrada. La salida del circuito XOR 256 es la salida del sumador completo 242.

45 El circuito XOR 250 efectúa la operación lógica XOR bit a bit de las entradas a y b, llevando a cabo efectivamente otra generación de término de suma similar a los sumadores CSA 240. Los generadores 252A-252B de acarreo llevan a cabo una verdadera generación de acarreo sobre las entradas a y b, y si la salida de acarreo del generador 252A de acarreo es un cero, la salida del generador 252A de acarreo se selecciona a través del mux 254 para la operación XOR con el término de suma proveniente del circuito XOR 250, creando por ello la suma final complementaria a unos. Por otra parte, si la salida de acarreo del generador 252A de acarreo es un uno, la salida del generador 252B de acarreo se selecciona a través del mux 254. Como la entrada de acarreo al generador 252B de acarreo es un uno, el generador 252B de acarreo efectivamente ha desplazado circularmente la salida de acarreo desde el bit más significativo de nuevo. Visto de otra manera, los generadores 252A- 252B de acarreo y el mux 254 pueden desplazar circularmente, de manera selectiva, un bit de acarreo más, en base a si se genera o no el acarreo.

Aspectos del objeto descrito en el presente documento se exponen en las siguientes cláusulas numeradas:

60 1. Un aparato que comprende:

un primer circuito de interfaz configurado para comunicarse por una interfaz según un protocolo;

un controlador de acceso directo a memoria (DMA) acoplado con el primer circuito de interfaz; y

un anfitrión acoplado con el controlador de DMA, en el que el anfitrión comprende al menos un espacio de direcciones mapeado, al menos en parte, a una pluralidad de ubicaciones de memoria en un sistema de memoria del anfitrión;

5 en el que el controlador de DMA está configurado para llevar a cabo transferencias de DMA entre el primer circuito de interfaz y el espacio de direcciones, y en el que el controlador de DMA está adicionalmente configurado para llevar a cabo transferencias de DMA entre una primera pluralidad de ubicaciones de memoria y una segunda pluralidad de ubicaciones de memoria.

10 2. El aparato según recitado en la cláusula 1, en el que el controlador de DMA comprende un circuito de control de transmisión, un circuito de control de recepción y un circuito de bucle de ensayo acoplado con el circuito de control de transmisión y el circuito de control de recepción, en el que el circuito de control de transmisión está configurado para leer datos desde la primera pluralidad y proporcionar los datos al circuito de bucle de ensayo, y en el que el circuito de bucle de ensayo está configurado para proporcionar los datos al circuito de control de recepción, y en el que el circuito de control de recepción está configurado para escribir los datos en la segunda pluralidad.

15 3. El aparato según recitado en la cláusula 2, en el que el circuito de control de recepción está adicionalmente configurado para escribir datos del primer circuito de interfaz en el espacio de direcciones, y en el cual el controlador de DMA comprende un árbitro configurado para arbitrar entre el circuito de bucle de ensayo y el primer circuito de interfaz, a fin de proporcionar datos al circuito de control de recepción.

20 4. El aparato según recitado en cualquiera de las cláusulas 1 a 3, en el que el primer circuito de interfaz comprende un controlador de acceso al medio (MAC).

25 5. El aparato según recitado en la cláusula 4, que comprende adicionalmente una pluralidad de circuitos de interfaz, incluyendo el primer circuito de interfaz, en el que la pluralidad de circuitos de interfaz están acoplados con el controlador de DMA.

30 6. El aparato según recitado en cualquiera de las cláusulas 1 a 3, en el que el primer circuito de interfaz comprende un circuito controlador de interfaz periférica.

7. El aparato según recitado en la cláusula 6, que comprende adicionalmente una pluralidad de circuitos de interfaz que incluye el primer circuito de interfaz, en el que la pluralidad de circuitos de interfaz están acoplados con el controlador de DMA.

35 8. El aparato según recitado en las cláusulas 6 ó 7, en el que un segundo circuito de interfaz de la pluralidad de circuitos de interfaz comprende un controlador de acceso al medio (MAC).

40 9. El aparato según recitado en cualquiera de las cláusulas 1 a 8, en el que el controlador de acceso directo a memoria (DMA) está configurado para llevar a cabo una transferencia de DMA desde el espacio de direcciones a un destino, en el que el destino es bien el circuito de interfaz o bien el anfitrión, dependiente de un canal al cual está asignada la transferencia de DMA, y en el que el controlador de DMA está configurado para llevar a cabo al menos una primera operación sobre los primeros datos de DMA leídos desde el espacio de direcciones para la transferencia de DMA, produciendo la primera operación un resultado.

45 10. El aparato según recitado en la cláusula 9, en el que la primera operación transforma los primeros datos de DMA en los segundos datos de DMA, y el controlador de DMA está configurado para transferir los segundos datos de DMA al destino.

50 11. El aparato según recitado en las cláusulas 9 ó 10, en el cual la primera operación comprende encriptación.

12. El aparato según recitado en la cláusula 11, en el que el controlador de DMA está adicionalmente configurado para aplicar una función de mapeo a los primeros datos de DMA, y en el que el resultado comprende la salida de la función de mapeo.

55 13. El aparato según recitado en las cláusulas 9 ó 10, en el que el resultado se almacena en una estructura de datos de un descriptor de DMA que corresponde a los primeros datos de DMA.

14. El aparato según recitado en la cláusula 13, en el que la primera operación es una función de mapeo.

60 15. El aparato según recitado en la cláusula 13, en el que la primera operación es una generación de control de redundancia cíclico.

16. El aparato según recitado en la cláusula 13, en el que la primera operación es una generación de suma de control.

5 17. El aparato según recitado en la cláusula 9, en el que el controlador de DMA está configurado para leer datos desde una pluralidad de canales y para llevar a cabo la operación lógica XOR sobre los datos a fin de producir el resultado.

18. Un procedimiento que comprende:

10 un controlador de acceso directo a memoria (DMA) que lleva a cabo una primera transferencia de DMA entre un primer circuito de interfaz y un espacio de direcciones en un anfitrión, en el que el espacio de direcciones está mapeado, al menos en parte, a una pluralidad de ubicaciones de memoria en el anfitrión, en el que el primer circuito de interfaz está configurado para comunicarse por una interfaz según un protocolo; y

15 el controlador de DMA que lleva a cabo una segunda transferencia de DMA entre una primera pluralidad de la pluralidad de ubicaciones de memoria y una segunda pluralidad de la pluralidad de ubicaciones de memoria.

19. El procedimiento según recitado en la cláusula 18, en el que el primer circuito de interfaz comprende un controlador de acceso al medio (MAC).

20 20. El procedimiento según recitado en las cláusulas 18 ó 19, en el que el controlador de DMA está acoplado con una pluralidad de circuitos de interfaz que incluye el primer circuito de interfaz, en el que la pluralidad de circuitos de interfaz están acoplados con el controlador de DMA.

21. El procedimiento según recitado en cualquiera de las cláusulas 18 a 20, en el que un segundo circuito de interfaz de la pluralidad de circuitos de interfaz comprende un circuito controlador de interfaz periférica.

25 22. Un controlador de acceso directo a memoria (DMA) que comprende:

30 un circuito de control de transmisión configurado para leer primeros datos de DMA desde un espacio de direcciones en un anfitrión y para proporcionar los primeros datos de DMA bien a un motor de descarga en el controlador de DMA o bien a un circuito de interfaz, dependiente de un canal al cual está asignada la transferencia de DMA; y

35 el motor de descarga acoplado para recibir los primeros datos de DMA desde los circuitos de control de transmisión, en el que el motor de descarga está configurado para llevar a cabo al menos una primera operación sobre los primeros datos de DMA para producir un resultado, en el que el motor de descarga está configurado para comenzar al menos a llevar a cabo la primera operación durante una transferencia de DMA que proporciona los primeros datos de DMA al motor de descarga; y

40 un circuito de control de recepción acoplado con el motor de descarga para recibir el resultado, en el que el circuito de control de recepción está configurado para escribir el resultado en el espacio de direcciones en el anfitrión según una estructura de datos de un descriptor de DMA que describe la transferencia de DMA, y en el que el circuito de control de recepción está adicionalmente configurado para escribir datos recibidos desde el circuito de interfaz en el espacio de direcciones.

45 23. El controlador de DMA según recitado en la cláusula 22, en el que la primera operación transforma los primeros datos de DMA en los segundos datos de DMA, y en el que el motor de descarga está acoplado con el circuito de control de recepción y está configurado para proporcionar los segundos datos de DMA al circuito de control de recepción.

24. El controlador de DMA según recitado en las cláusulas 22 ó 23, en el que la primera operación comprende encriptación.

50 25. El controlador de DMA según recitado en cualquiera de las cláusulas 22 a 24, en el que el motor de descarga está adicionalmente configurado para aplicar una función de mapeo a los primeros datos de DMA, y en el que el resultado comprende la salida de la función de mapeo.

55 26. El controlador de DMA según recitado en cualquiera de las cláusulas 22 a 25, en el que el resultado se almacena en una estructura de datos del descriptor de DMA, que corresponde a los primeros datos de DMA.

27. El controlador de DMA según recitado en la cláusula 26, en el que la primera operación es una función de mapeo.

60 28. El controlador de DMA según recitado en la cláusula 26, en el que la primera operación es una generación de control de redundancia cíclico.

29. El controlador de DMA según recitado en la cláusula 26, en el que la primera operación es una generación de suma de control.
- 5 30. El controlador de DMA según recitado en cualquiera de las cláusulas 2,2 a 29, que comprende adicionalmente un circuito de O exclusivo (XOR), en el que el circuito de control de transmisión está configurado para leer datos desde una pluralidad de canales y proporcionar los datos al circuito XOR, en el que el circuito XOR está configurado para efectuar la operación lógica XOR sobre los datos a fin de producir el resultado.
- 10 31. Un controlador de acceso directo a memoria (DMA) configurado para leer los primeros datos de DMA desde un espacio de direcciones en un anfitrión y proporcionar los primeros datos de DMA bien a un motor de descarga en el controlador de DMA o bien a un circuito de interfaz, dependiente de un canal al cual se asigna la transferencia de DMA, en el que el motor de descarga está configurado para llevar a cabo al menos una primera operación sobre los primeros datos de DMA a fin de producir un resultado, en el que el motor de descarga está configurado para iniciar el menos la ejecución de la primera operación durante una transferencia de DMA que proporciona los primeros datos de DMA al motor de descarga, y en el que el controlador de DMA está configurado para escribir el resultado en el espacio de direcciones en el anfitrión según una estructura de datos de un descriptor de DMA que describe la transferencia de DMA, y en el que el controlador de DMA está adicionalmente configurado para escribir datos recibidos desde el circuito de interfaz en el espacio de direcciones.
- 15
- 20 Numerosas variaciones y modificaciones se harán evidentes a los expertos en la técnica una vez que la divulgación anterior sea completamente apreciada. Se pretende que las siguientes reivindicaciones se interpreten como abarcando todas dichas variaciones y modificaciones.

REIVINDICACIONES

1. Un controlador (14) de acceso directo a memoria (DMA) configurado para leer primeros datos de DMA desde un espacio de direcciones en un anfitrión (12) y para proporcionar los primeros datos de DMA bien a un motor (44) de descarga en el controlador (14) de DMA o bien a un circuito (16) de interfaz, dependiente de un canal al cual está asignada la transferencia de DMA,

5 en el que los primeros datos de DMA se leen desde una primera pluralidad de ubicaciones de memoria en una memoria acoplada con un controlador (20A-20B) de memoria dentro del anfitrión (12), en el que la primera pluralidad de ubicaciones de memoria están mapeadas al espacio de direcciones en el anfitrión (12), en el que el motor (44) de descarga está configurado para llevar a cabo al menos una primera operación sobre los primeros datos de DMA a fin de producir un resultado sensible a la recepción de los primeros datos de DMA, en el que el motor (44) de descarga está configurado para iniciar al menos el llevar a cabo de la primera operación durante una transferencia de DMA que proporciona los primeros datos de DMA al motor (44) de descarga,

caracterizado porque

15 el controlador (14) de DMA está configurado para escribir el resultado en el espacio de direcciones en el anfitrión (12) según una estructura (112A-112N) de datos de descriptor de DMA que describe la transferencia de DMA, y en el que el resultado se escribe en una o más segundas ubicaciones de memoria en la memoria acoplada con el controlador (20A-20B) de memoria dentro del anfitrión (12), en el que las segundas ubicaciones de memoria están mapeadas al espacio de direcciones en el anfitrión (12), y en el que la estructura (112A-112N) de datos de descriptor de DMA se almacena en memoria y es leída desde memoria por el controlador (14) de DMA, y en el que el controlador de DMA está adicionalmente configurado para escribir datos recibidos desde el circuito de interfaz en el espacio de direcciones.

2. El controlador (14) de acceso directo a memoria (DMA) según recitado en la reivindicación 1, que comprende:

un circuito (56) de control de transmisión configurado para leer los primeros datos de DMA;

25 el motor (44) de descarga acoplado para recibir los primeros datos de DMA desde el circuito (56) de control de transmisión; y

un circuito (58) de control de recepción acoplado con el motor (44) de descarga para recibir el resultado, en el que el circuito (58) de control de recepción está configurado para escribir el resultado en el espacio de direcciones en el anfitrión (12) según la estructura (112A-112N) de datos de descriptor de DMA que describe la transferencia de DMA, y

30 en el que el circuito (58) de control de recepción está adicionalmente configurado para escribir datos recibidos desde el circuito (16) de interfaz en el espacio de direcciones.

3. El controlador (14) de DMA según recitado en la reivindicación 2, en el que la primera operación transforma los primeros datos de DMA en segundos datos de DMA, y en el que el motor (44) de descarga está acoplado con el circuito (58) de control de recepción y está configurado para proporcionar los segundos datos de DMA al circuito (58) de control de recepción.

4. El controlador (14) de DMA según recitado en cualquiera de las reivindicaciones 1 a 3, en el que la primera operación comprende encriptación.

5. El controlador (14) de DMA según recitado en cualquiera de las reivindicaciones 1 a 4, en el que el motor (44) de descarga está adicionalmente configurado para aplicar una función de mapeo a los primeros datos de DMA, y en el que el resultado comprende la salida de la función de mapeo.

6. El controlador (14) de DMA según recitado en cualquiera de las reivindicaciones 1 a 5, en el que el resultado se almacena en una estructura (112A-112N) de datos de descriptor de DMA que corresponde a los primeros datos de DMA.

7. El controlador (14) de DMA según recitado en la reivindicación 6, en el que la primera operación es una entre una función de mapeo, una generación de control de redundancia cíclico o una generación de suma de control.

45 8. El controlador (14) de DMA según recitado en cualquiera de las reivindicaciones 2 a 7, que comprende adicionalmente un circuito (50) de O exclusivo (XOR), en el que el circuito (58) de control de transmisión está configurado para leer datos desde una pluralidad de canales y proporcionar los datos al circuito XOR, en el que el circuito XOR está configurado para XOR sobre los datos a fin de producir el resultado.

9. Un procedimiento que comprende:

leer primeros datos de DMA desde una primera pluralidad de ubicaciones de memoria en una memoria acoplada con un controlador (20A-20B) de memoria dentro del anfitrión (12) en un controlador (14) de DMA, en el que la primera pluralidad de ubicaciones de memoria están mapeadas al espacio de direcciones en el anfitrión (12);

5 proporcionar los primeros datos de DMA bien a un motor (44) de descarga en el controlador (14) de DMA o bien a un circuito (16) de interfaz, dependiente de un canal al cual está asignada la transferencia de DMA;

llevar a cabo al menos una primera operación sobre los primeros datos de DMA para producir un resultado en el motor (44) de descarga, sensible a la recepción por el motor (44) de descarga de los primeros datos de DMA, en el que el motor (44) de descarga está configurado para comenzar al menos a llevar a cabo la primera operación durante una transferencia de DMA que proporciona los primeros datos de DMA al motor (44) de descarga;

10 **caracterizado por**

escribir el resultado del controlador (14) de DMA en una o más segundas ubicaciones de memoria en la memoria acoplada con el controlador (20A-20B) de memoria dentro del anfitrión (12), en el que las segundas ubicaciones de memoria están mapeadas al espacio de direcciones en el anfitrión (12), y en el que la escritura es según una estructura (112A-112N) de datos de descriptor de DMA que describe la transferencia de DMA, y en el que la estructura (112A-112N) de datos de descriptor de DMA está almacenada en memoria y es leída desde memoria por el controlador (14) de DMA; y

escribir los datos recibidos desde el circuito (16) de interfaz en el espacio de direcciones.

20 **10.** El procedimiento según recitado en la reivindicación 9, en el que la primera operación transforma los primeros datos de DMA en segundos datos de DMA, y el procedimiento comprende adicionalmente escribir los segundos datos de DMA en el espacio de direcciones.

11. El procedimiento según recitado en cualquiera de las reivindicaciones 9 ó 10, en el que la primera operación comprende encriptación.

25 **12.** El procedimiento según recitado en cualquiera de las reivindicaciones 9 a 11, que comprende adicionalmente aplicar una función de mapeo a los primeros datos de DMA en el motor (44) de descarga, y en el que el resultado comprende la salida de la función de mapeo.

13. El procedimiento según recitado en cualquiera de las reivindicaciones 9 a 12, en el que el resultado se almacena en una estructura (112A-112N) de datos de descriptor de DMA que corresponde a los primeros datos de DMA, y en el que la primera operación es una entre una función de mapeo, una generación de control de redundancia cíclico o una generación de suma de control.

30 **14.** El procedimiento según recitado en cualquiera de las reivindicaciones 9 a 13, que comprende adicionalmente leer datos desde una pluralidad de canales y efectuar O exclusivo (XOR) sobre los datos a fin de producir el resultado.

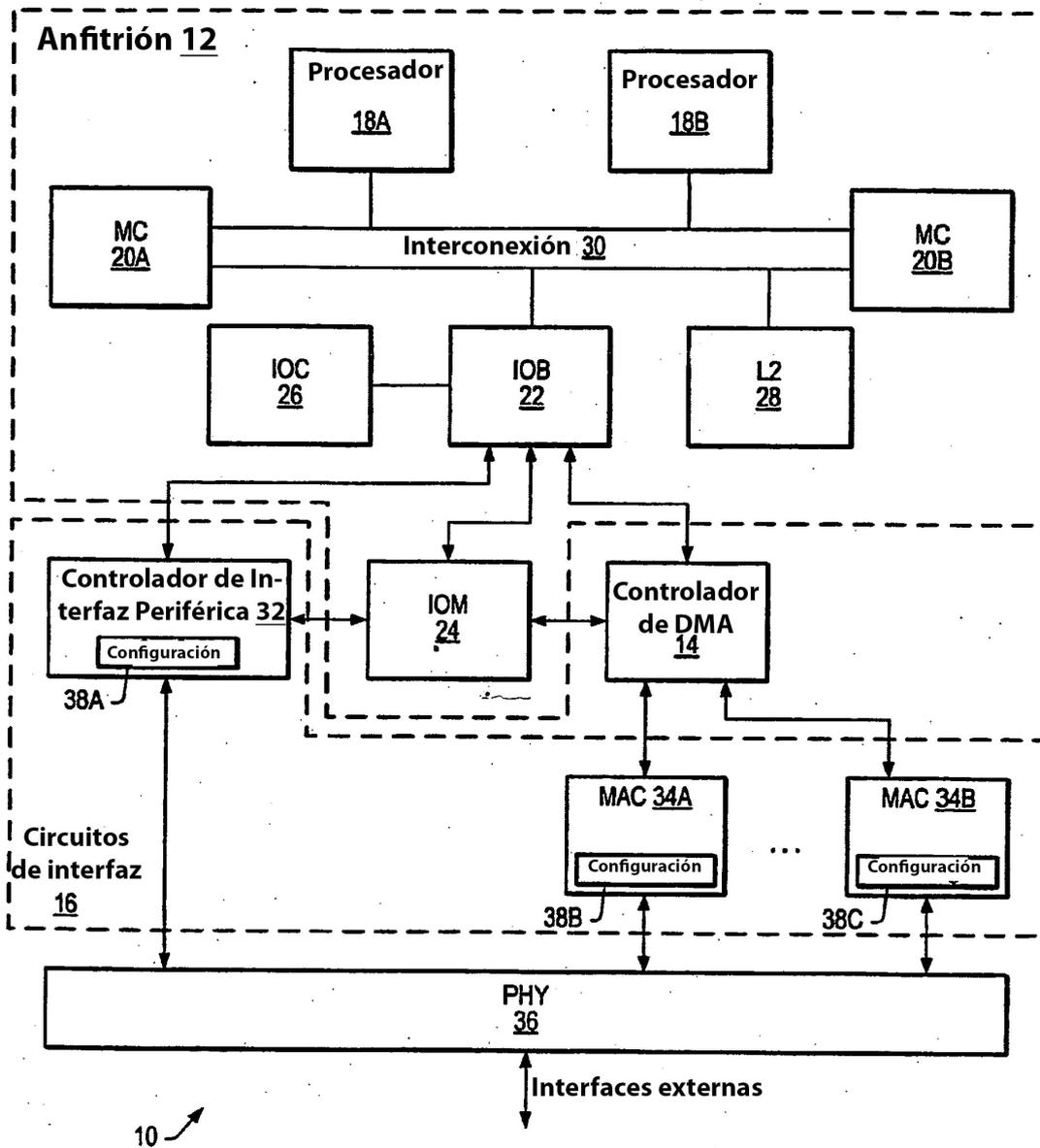


FIG. 1

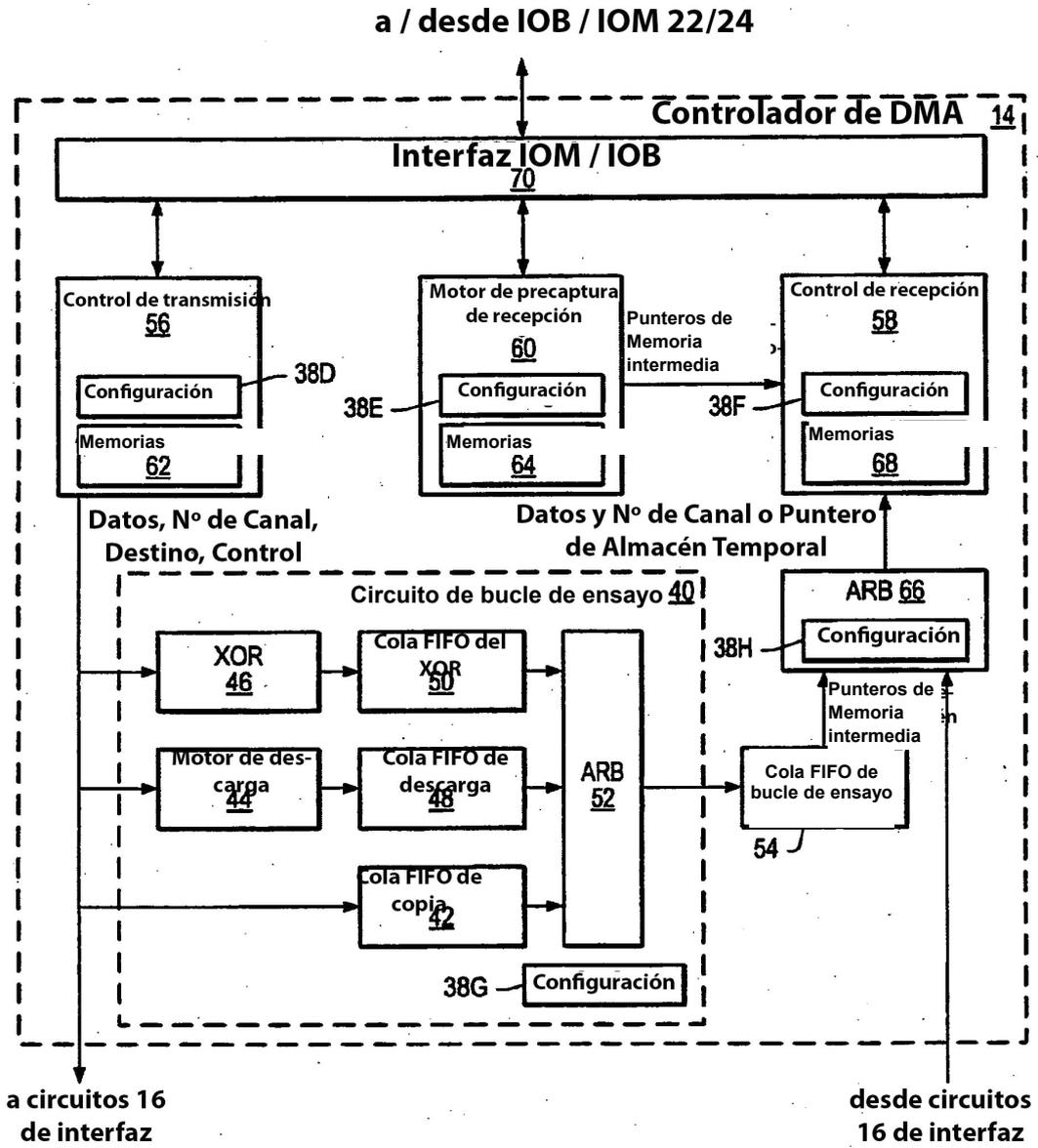


FIG. 2

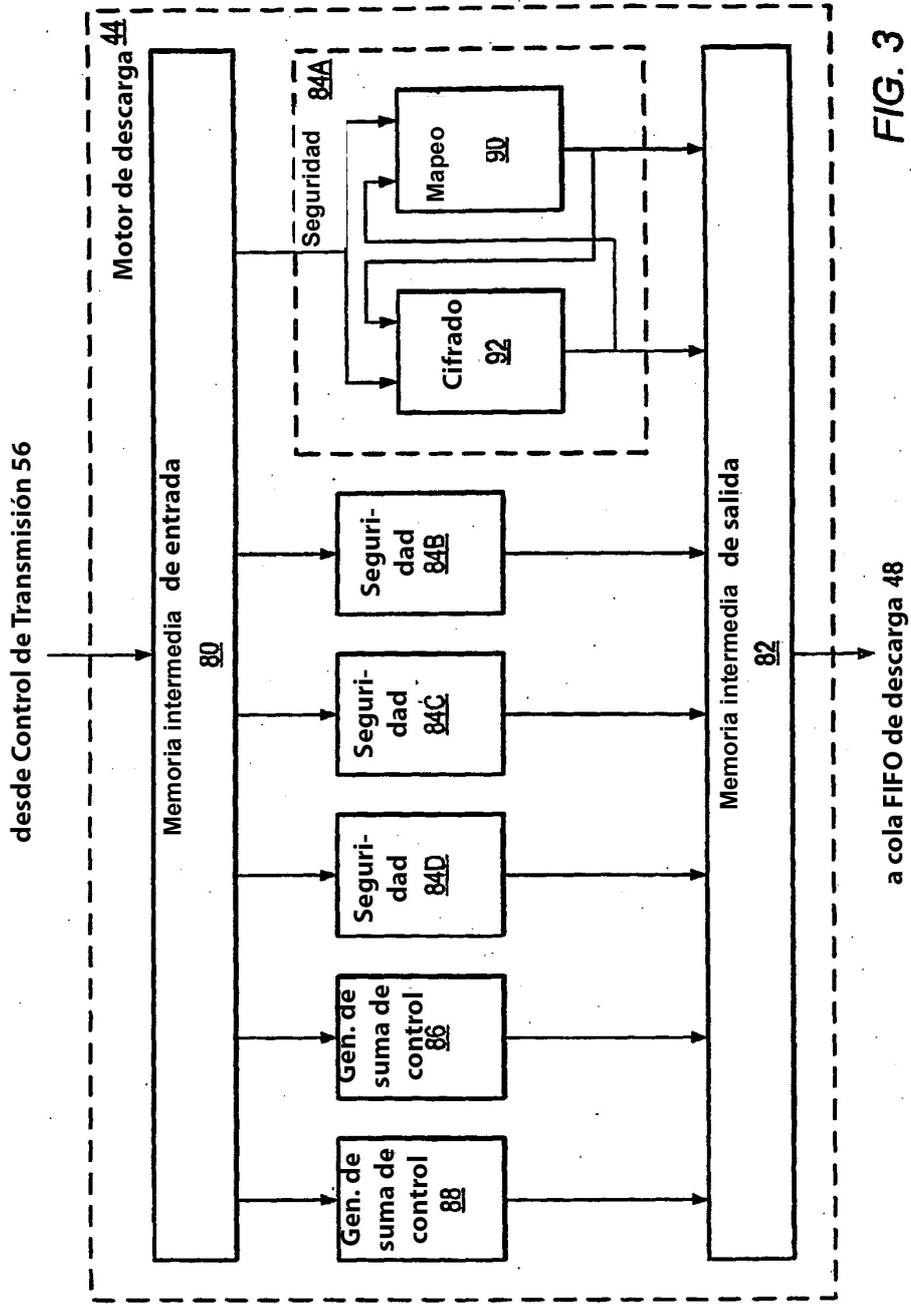


FIG. 3

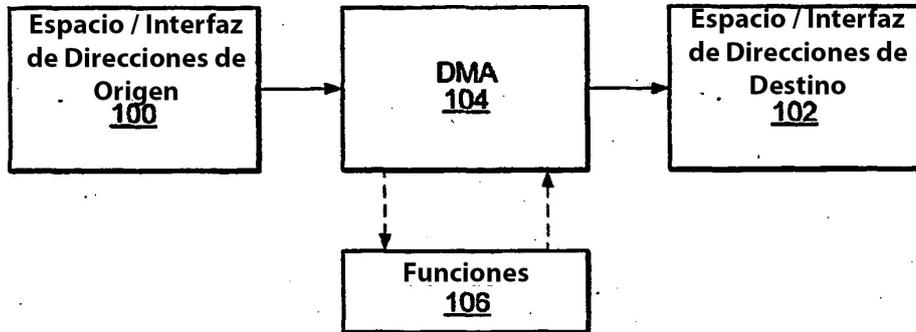


FIG. 4

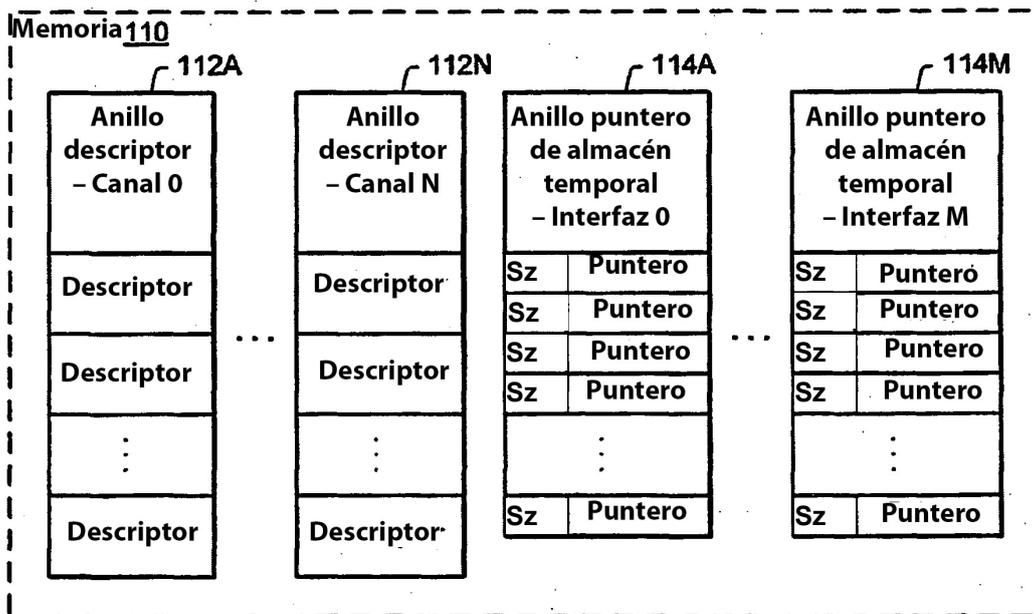


FIG. 5

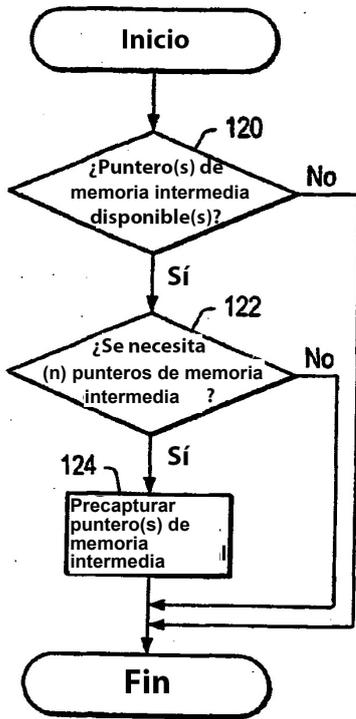


FIG. 6

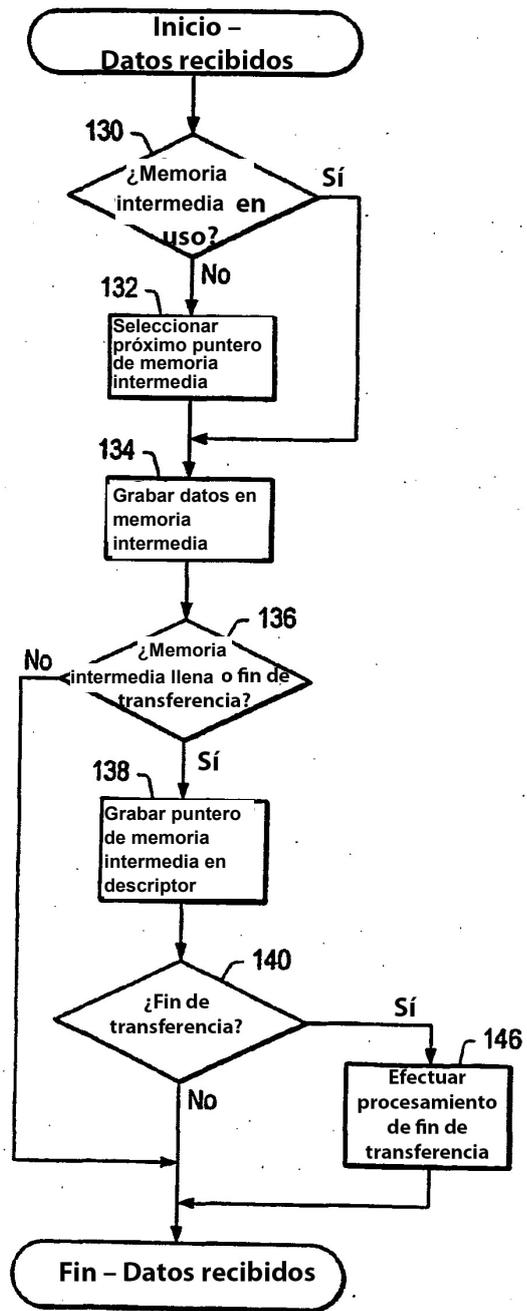


FIG. 7

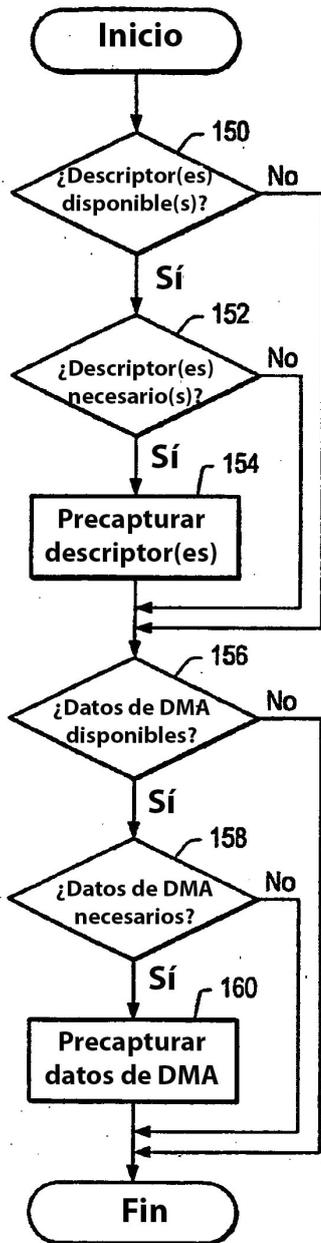


FIG. 8

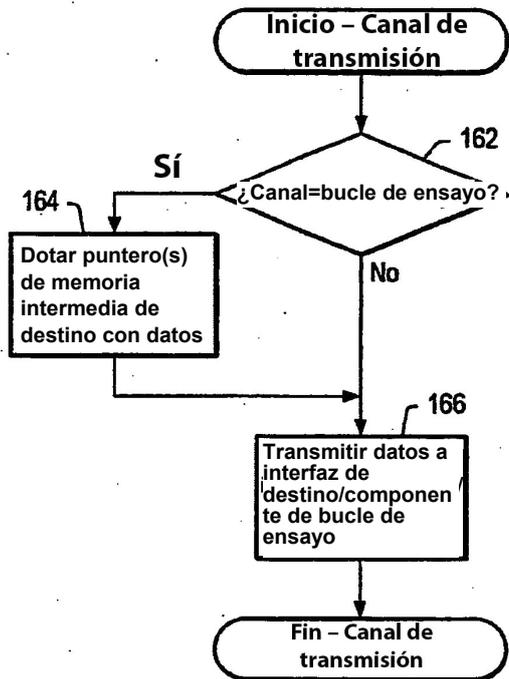


FIG. 9

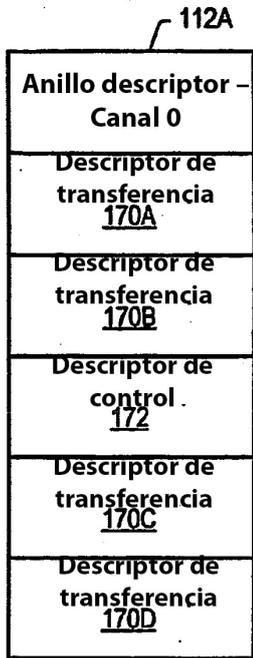


FIG. 10

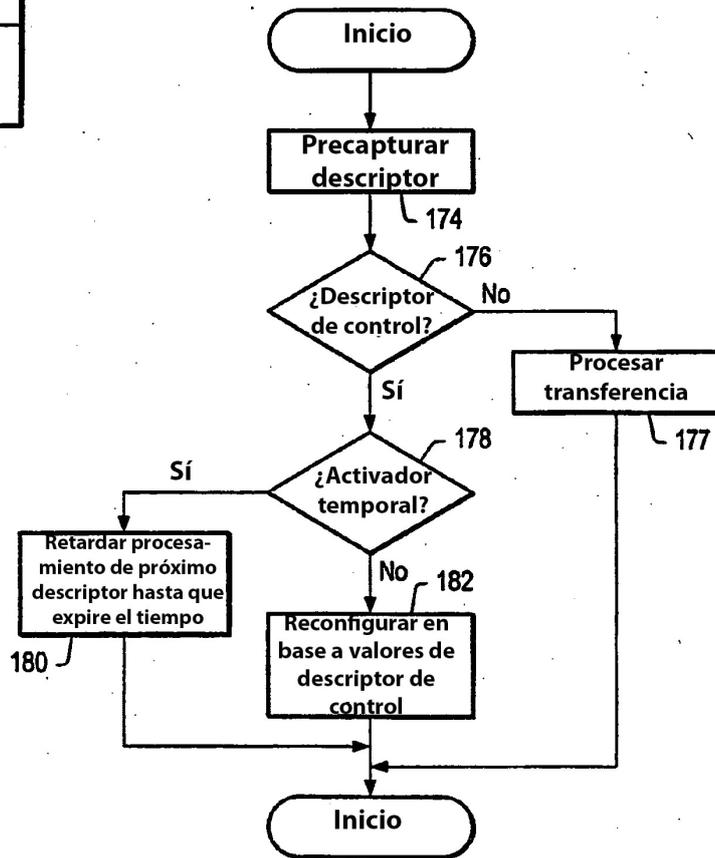


FIG. 11

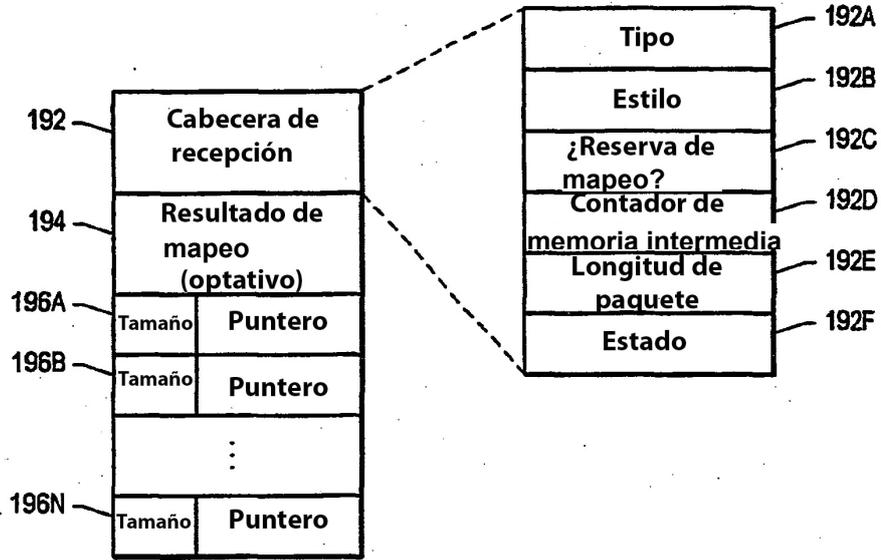


FIG. 12

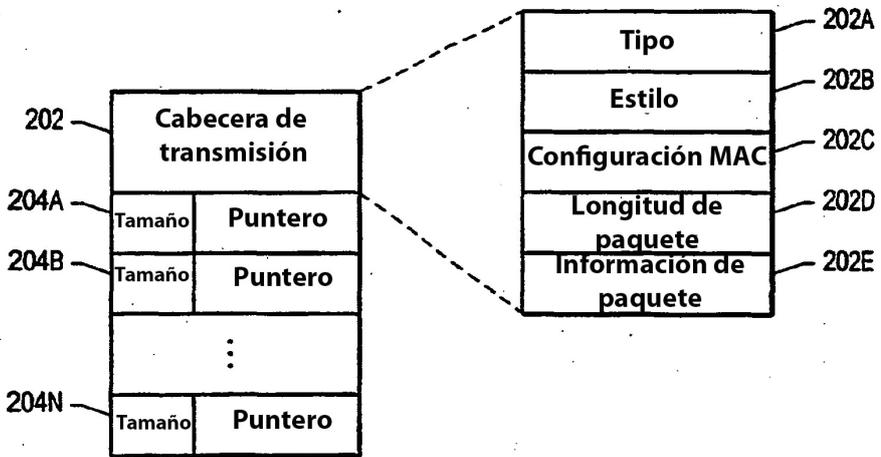


FIG. 13

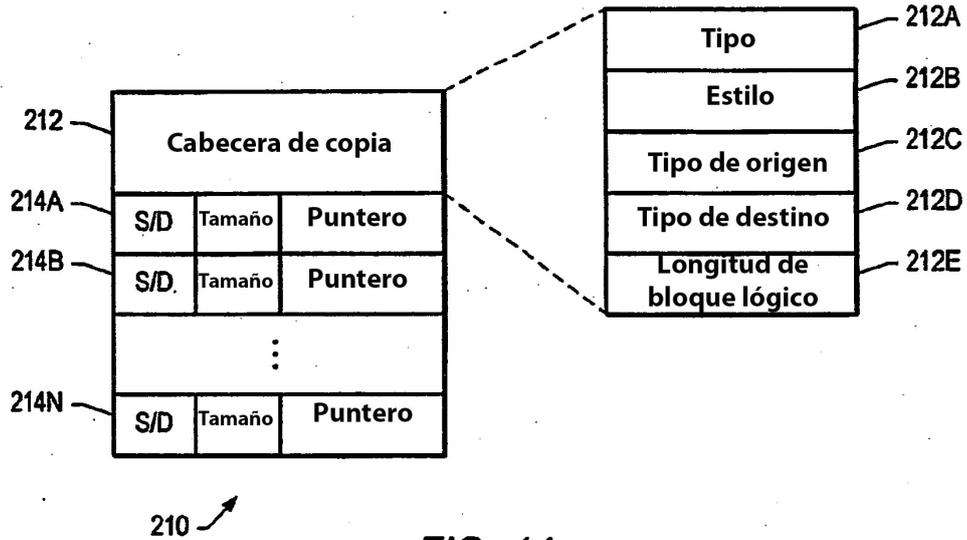


FIG. 14

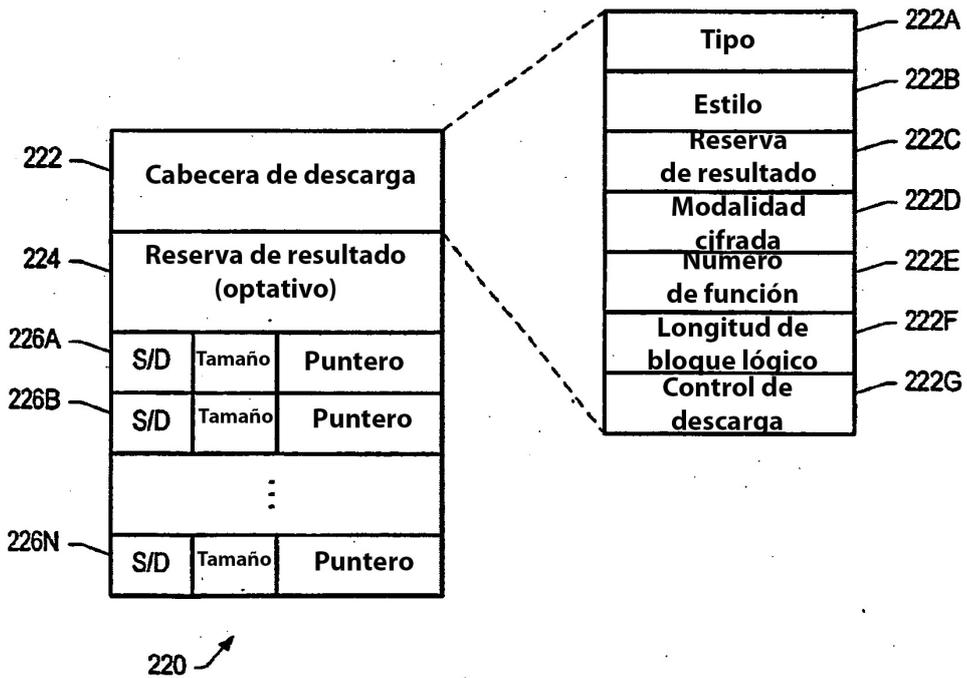
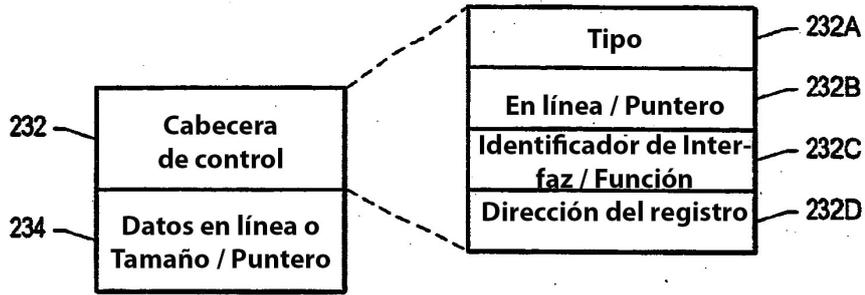


FIG. 15



230 **FIG. 16**

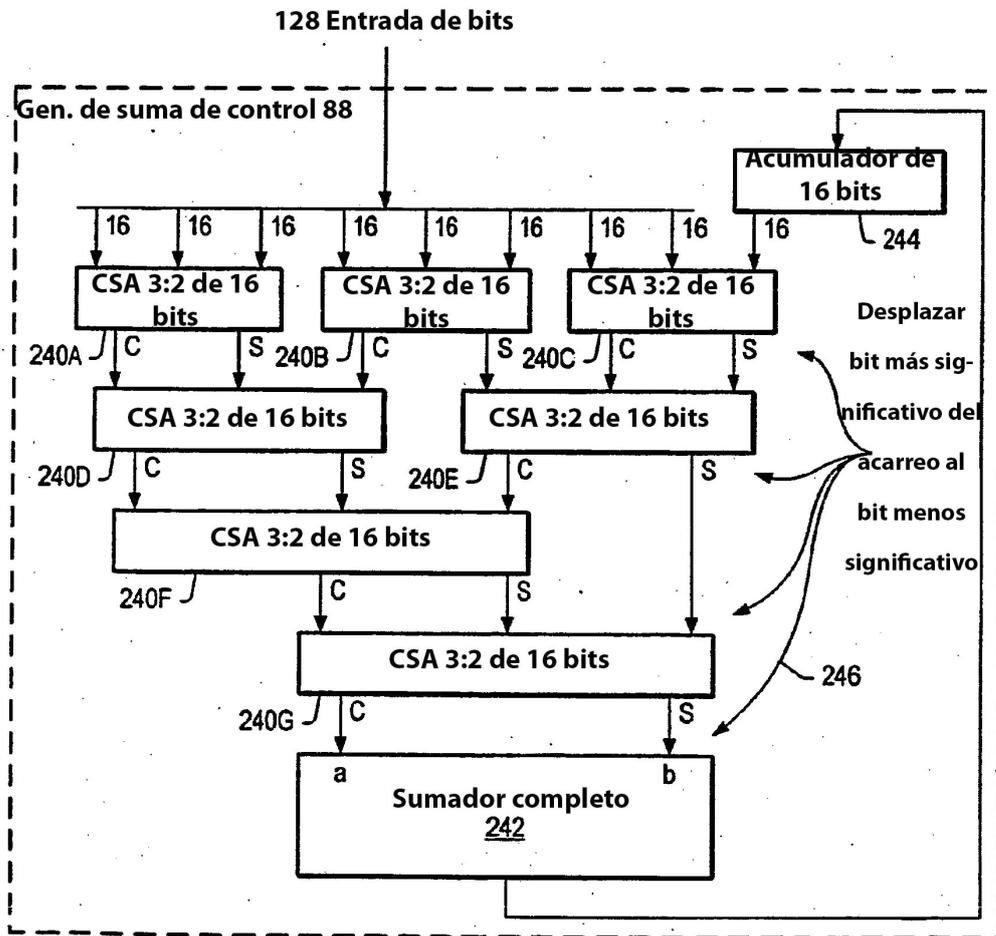


FIG. 17

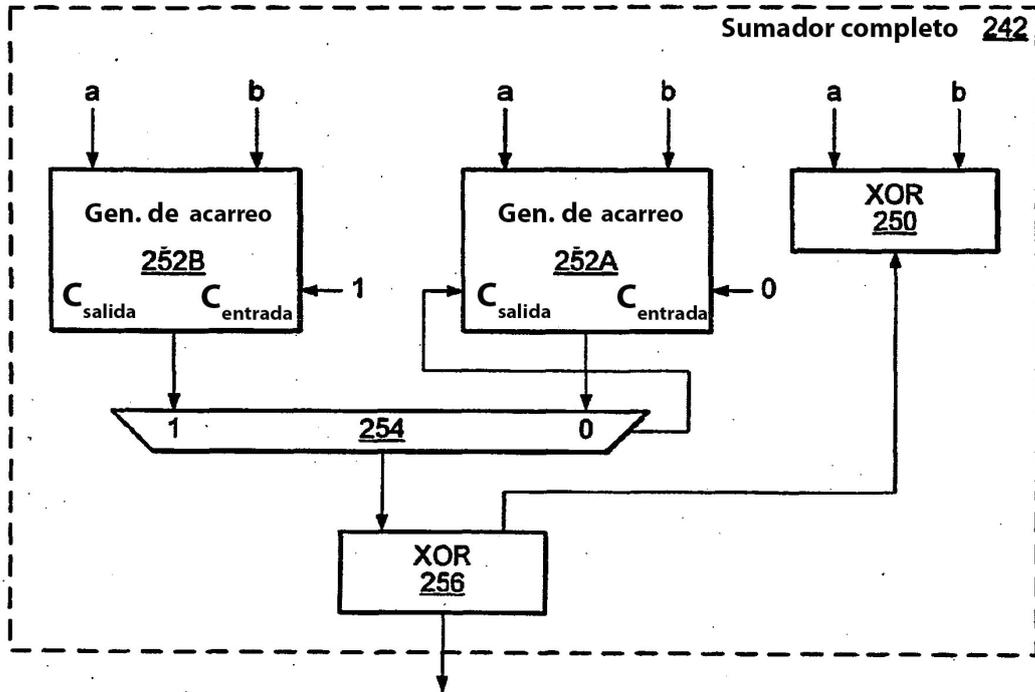


FIG. 18