



①9



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA

①1 Número de publicación: **2 327 325**

⑤1 Int. Cl.:
H04N 7/16 (2006.01)

⑫

TRADUCCIÓN DE PATENTE EUROPEA

T3

⑨6 Número de solicitud europea: **03000543 .3**

⑨6 Fecha de presentación : **09.01.2003**

⑨7 Número de publicación de la solicitud: **1345435**

⑨7 Fecha de publicación de la solicitud: **17.09.2003**

⑤4 Título: **Procedimientos y sistemas de tarjeta de vídeo segura.**

③0 Prioridad: **16.01.2002 US 52840**

④5 Fecha de publicación de la mención BOPI:
28.10.2009

④5 Fecha de la publicación del folleto de la patente:
28.10.2009

⑦3 Titular/es: **MICROSOFT CORPORATION**
One Microsoft Way
Redmond, Washington 98052-6399, US

⑦2 Inventor/es: **Evans, Glen F.**

⑦4 Agente: **Carpintero López, Mario**

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

Procedimientos y sistemas de tarjeta de vídeo segura.

5 **Campo técnico**

La presente invención se refiere a procedimientos y sistemas para procesar datos de vídeo usando tarjetas de vídeo.

Antecedentes

10

Normalmente, un autor de contenido, tal como un estudio de cine o un usuario que publique contenido en la web, publicará el contenido de vídeo con restricciones sobre cómo pueden verlo los usuarios. Este contenido normalmente puede verse o reproducirse en un ordenador, tal como un ordenador personal. Personas y organizaciones sin escrúpulos gastan una gran cantidad de tiempo, esfuerzo y dinero cada año intentando robar u obtener de otro modo y de forma inapropiada tal contenido de vídeo.

15

Uno de los puntos de ataque puede ser el ordenador en el que va a verse o reproducirse tal contenido de vídeo. Es decir, programas o dispositivos malintencionados pueden, y a menudo intentan, obtener de forma inapropiada contenido de vídeo tras haber sido recibido en un ordenador, tal como un ordenador personal. Entre otros componentes informáticos, este ataque puede hacerse contra la tarjeta de vídeo que procesa el contenido de vídeo y/o el bus que transporta el contenido de vídeo hacia y desde la tarjeta de vídeo.

20

La figura 1 muestra una tarjeta 100 de vídeo (o gráfica) ejemplar que incluye un conector 102 de bus que se inserta en un puerto en un ordenador normal. La tarjeta 100 de vídeo también incluye un conector 104 de monitor (por ejemplo un enchufe de 15 patillas) que recibe un cable que se conecta a un monitor. La tarjeta 100 de vídeo puede incluir un zócalo 106 de salida de vídeo digital que puede usarse para enviar imágenes de vídeo a monitores LCD y de pantalla plana y similares.

25

La tarjeta de vídeo en la actualidad consiste en cuatro componentes principales: la unidad 108 de procesador gráfico (GPU), la memoria 110 de vídeo, el convertidor 112 digital a analógico de memoria de acceso aleatorio (RAMDAC) y el software de controlador que puede estar incluido en el BIOS 114 de vídeo.

30

La GPU 108 es un chip de procesamiento gráfico dedicado que controla todos los aspectos de resolución, profundidad de color y todos los elementos asociados con la renderización de imágenes en la pantalla del monitor. La unidad de procesamiento central o CPU del ordenador (no mostrada) envía un conjunto de instrucciones y datos de trazado, que se interpretan por el controlador propietario de la tarjeta gráfica y se ejecutan por la GPU 108 de la tarjeta. La GPU 108 realiza tales operaciones como dibujo y transferencias de mapa de bits, redimensionamientos y reposicionamiento de ventanas, trazado de líneas, ajuste a escala de fuentes y trazado de polígonos. La GPU 108 está diseñada para tratar estar tareas en hardware a velocidades mucho mayores que el software que se ejecuta en la CPU del sistema. La GPU escribe entonces los datos de cuadros en la memoria intermedia de cuadros (o en la memoria 110 de vídeo integrada). La GPU reduce en gran medida la carga de trabajo de la CPU del sistema.

35

40

La memoria que contiene la imagen de vídeo también se denomina memoria intermedia de cuadros y habitualmente está implementada en la propia tarjeta de vídeo. En este ejemplo, la memoria intermedia de cuadros está implementada en la tarjeta de vídeo en forma de memoria 110. Los sistemas iniciales implementaban la memoria de vídeo en DRAM estándar. Sin embargo, esto requiere una renovación continua de los datos para evitar que se pierdan y no pueden modificarse durante este proceso de renovación. La consecuencia, particularmente a las velocidades de reloj muy elevadas que requieren las tarjetas gráficas modernas, es que el rendimiento se ve perjudicado negativamente.

45

Una ventaja de implementar la memoria de vídeo en la propia tarjeta de vídeo es que puede adaptarse para su tarea específica y, de hecho, esto ha dado como resultado una proliferación de nuevas tecnologías de memoria:

50

55

60

65

- RAM de vídeo (VRAM): un tipo especial de DRAM de puerto doble, que puede escribirse y de la que puede leerse al mismo tiempo. También requiere una renovación mucho menos frecuente que la DRAM ordinaria y, por consiguiente, tiene un rendimiento mucho mejor;
- Windows RAM (WRAM): usada por la tarjeta Matrox Millennium, también tiene puerto doble y puede funcionar ligeramente más rápido que la VRAM convencional;
- EDO DRAM: que proporciona un ancho de banda superior a la DRAM, puede tener ciclos de reloj superiores a la DRAM normal y gestiona los ciclos de lectura/escritura de manera más eficaz;
- SDRAM: Similar a la EDO RAM salvo porque la memoria y los chips gráficos funcionan con un reloj común usado para sincronizar datos, lo que permite a la SDRAM funcionar más rápido que la EDO RAM normal;
- SGRAM: Igual que la SDRAM pero también soporta bloques de escritura y escritura por bit, que tienen un mejor rendimiento en chips gráficos que soportan estas características mejoradas; y

ES 2 327 325 T3

- DRDRAM: RDRAM directa es una arquitectura de memoria totalmente nueva, de propósito general que promete una mejora de rendimiento 20 veces superior frente a la DRAM convencional.

Algunos diseños integran los sistemas de circuitos gráficos en la propia placa base y usan una parte de la RAM del sistema para la memoria intermedia de cuadros. Esto se denomina “arquitectura de memoria unificada” y se usa por motivos de reducción de costes únicamente y puede llevar a un rendimiento gráfico inferior.

La información en la memoria intermedia de cuadros de la memoria de vídeo es una imagen de lo que aparece en la pantalla, almacenada como un mapa de bits digital. Sin embargo, mientras que la memoria de vídeo contiene información digital, su medio de salida, el monitor, puede usar señales analógicas. Las señales analógicas requieren más que simplemente una señal “on” u “off”, ya que se utiliza para determinar dónde, cuándo y con qué intensidad han de dispararse los cañones de electrones a medida que realizan un barrido por y bajando por el frontal del monitor. Aquí es donde entra en juego la RAMDAC 112 como se describe más adelante. Algunas RAMDAC también soportan salidas de interfaz de vídeo digital (DVI) para dispositivos de visualización digitales tales como monitores LCD. En tales configuraciones, la RAMDAC convierte la representación digital interna en una forma comprensible por el dispositivo de visualización digital.

La RAMDAC desempeña el papel de un “convertidor de visualización” puesto que convierte los datos digitales internos a una forma que entiende el dispositivo de visualización.

A pesar de que la cantidad total de memoria de vídeo instalada en la tarjeta de vídeo puede no ser necesaria para una resolución particular, la memoria extra se usa a menudo para almacenar en caché información para la GPU 108. Por ejemplo, el almacenamiento en caché de elementos gráficos habitualmente usados, tal como fuentes de texto e iconos o imágenes, evita la necesidad de que el subsistema gráfico los cargue cada vez que se escribe una nueva letra o se mueve un icono y mejora así el rendimiento. Las imágenes almacenadas en caché pueden usarse para poner en cola secuencias de imágenes para que las presente la GPU, dejando así libre a la CPU para realizar otras tareas.

Muchas veces por segundo, la RAMDAC 112 lee el contenido de la memoria de vídeo, lo convierte en una señal y lo envía por el cable de vídeo al monitor. Para dispositivos de visualización analógicos, normalmente hay un convertidor digital a analógico (DAC) para cada uno de los tres colores primarios que usa el CRT para crear un espectro completo de colores. Para dispositivos de visualización digitales, la RAMDAC emite un único flujo de datos RGB para que lo interprete y visualice el dispositivo de salida. El resultado pretendido es la mezcla correcta necesaria para crear el color de un píxel individual. La tasa a la que la RAMDAC 112 puede convertir la información, y el propio diseño de la GPU 108, dicta el intervalo de tasas de renovación que puede soportar la tarjeta gráfica. La RAMDAC 112 también dicta el número de colores disponibles en una resolución dada, dependiendo de su arquitectura interna.

El conector 102 de bus puede soportar uno o más buses que se usan para conectarse con la tarjeta de vídeo. Por ejemplo, un bus de puerto gráfico acelerado (AGP) puede permitir a la tarjeta de vídeo acceder directamente a la memoria de sistema. El acceso directo a la memoria ayuda a hacer el ancho de banda pico muchas veces superior que el bus de interconexión de componentes periféricos (PCI). Esto puede permitir a la CPU del sistema realizar otras tareas mientras que la GPU en la tarjeta de vídeo accede a la memoria de sistema.

Durante la operación, los datos contenidos en la memoria de vídeo integrada pueden proporcionarse a la memoria de sistema del ordenador y pueden gestionarse como si fueran parte de la memoria de sistema. Esto incluye cosas tales como técnicas de gestión de memoria virtual que emplea el gestor de memoria del ordenador. Además, cuando se necesitan los datos contenidos en la memoria del sistema para una operación gráfica en la tarjeta de vídeo, los datos pueden enviarse sobre un bus (tal como un bus PCI o AGP) a la tarjeta de vídeo y almacenarse en la memoria 110 de vídeo integrada. En ella, la GPU 108 puede acceder a los datos y manipularlos como se ha descrito anteriormente.

Cuando los datos se transfieren desde la memoria de sistema a la memoria de vídeo en la tarjeta de vídeo y viceversa, es posible que dispositivos PCI conectados al bus PCI “escuchen” los datos mientras se transfieren. El bus PCI también hace la memoria de vídeo “visible” al resto del sistema, como si existiera como memoria de sistema. Como resultado, es posible que un dispositivo PCI adquiera el bus PCI y simplemente copie el contenido de la memoria de vídeo a otro dispositivo. Si el dispositivo PCI está sincronizado con el vídeo entrante, en potencia podría capturar todo el contenido.

Existen dos opciones previas para proteger el contenido una vez en la memoria de vídeo en la tarjeta de vídeo.

En primer lugar, la memoria de vídeo puede seguir siendo accesible, pero el contenido se almacena en una forma protegida cifrada de modo que sea ilegible para dispositivos y aplicaciones malintencionados. Aunque esto evita que se lean los datos, también requiere mantener los datos de manera continua en forma cifrada. Si la tarjeta de vídeo (es decir la GPU) desea procesar los datos, debe descifrarlos de manera atómica mientras los lee, procesarlos y volver a cifrarlos en cada nueva escritura en la memoria de vídeo. Para datos de vídeo, los datos descomprimidos podrían requerir más de 300 mb por segundo sólo para visualizarse. Por consiguiente, el dispositivo de cifrado/descifrado tendría que operar a estas tasas de transmisión de datos elevadas. Normalmente, se procesarán varios flujos de vídeo en un único flujo de salida. Por ejemplo, una visualización de imagen dentro de imagen (PIP) o multicanal mezclaría ocho canales en una única visualización. Esto requeriría ocho dispositivos de descifrado simultáneos y un dispositivo

de cifrado que funcione a 300 mb por segundo (un total de aproximadamente 2,4 gigabytes por segundo). Por tanto, este enfoque no es muy deseable debido a los elevados requisitos informáticos.

En segundo lugar, el contenido o datos en la memoria de vídeo pueden hacerse simplemente inaccesibles. Normalmente esto no es posible debido al diseño de los buses PCI y AGP, puesto que la memoria de vídeo está en correspondencia con la memoria física (es decir parece como si fuera memoria de sistema normal). La memoria de vídeo es por tanto accesible a cualquier dispositivo PCI que puede entonces adquirir el bus PCI y realizar la transferencia de datos sin el conocimiento de la CPU. Por tanto, este enfoque no es muy deseable puesto que el controlador de memoria (o GPU) no puede determinar con fiabilidad quién está accediendo a la memoria.

Por consiguiente, la presente invención surge por la preocupación asociada a proporcionar sistemas y procedimientos de procesamiento de vídeo seguros.

El documento US-A-6 064 739 se refiere a la protección de datos digitales usando una combinación de técnicas criptográficas y encapsulación física de datos digitales desde el momento en el que se remiten para el procesamiento hasta el momento en el que se proporcionan al monitor en forma analógica. El sistema informático descrito incluye un subsistema de vídeo que incluye además un procesador de contenido de vídeo seguro (SVCP), que está rodeado por una barrera de hardware que impide la manipulación indebida con los sistemas electrónicos del SVCP. Cuando el SVCP recibe un contenido de vídeo cifrado, los sistemas de circuitos de descifrado y los sistemas de circuitos de descompresión dentro del SVCP descifran y descomprimen el flujo de datos. Un procesador gráfico procesa entonces los datos descifrados para generar una representación digital de una imagen que se almacena en una memoria intermedia de cuadros hasta que se necesite por el dispositivo de visualización. En el caso en el que la memoria intermedia de cuadros esté situada fuera de la barrera de hardware que rodea al SVCP, los cuadros de imagen digital generados por el procesador gráfico se cifran antes de transmitirse fuera de la barrera de hardware.

Sumario

El objetivo de la invención es proporcionar un procedimiento y sistema de procesamiento de vídeo seguro más flexible.

Este objetivo se soluciona por la presente invención según se reivindica en las reivindicaciones independientes.

En las reivindicaciones dependientes se definen realizaciones preferidas.

Se describen procedimientos y sistemas para proteger datos previstos para su uso y procesamiento en tarjetas de vídeo o gráficas. En diversas realizaciones, los datos previstos para su uso por una tarjeta de vídeo pueden cifrarse de manera selectiva de manera que cada vez que los datos se proporcionan a un bus (por ejemplo un bus PCI o AGP) entre la tarjeta de vídeo y el sistema informático, los datos estén cifrados. Por ejemplo, cuando los datos se mueven desde la memoria en la tarjeta de vídeo a la memoria del sistema (por ejemplo cuando está en correspondencia con la memoria física del sistema) y viceversa, los datos están en forma cifrada y, por tanto, protegidos.

Cuando los datos cifrados se transfieren de la memoria del sistema a la tarjeta de vídeo para su procesamiento por una unidad de procesador gráfico (GPU) en la tarjeta de vídeo, pueden descifrarse y disponerse en memoria protegida en la tarjeta de vídeo.

En una realización, la tarjeta de vídeo tiene partes de memoria que están protegidas. Pares de claves de cifrado/descifrado están asociados con cada parte de memoria protegida. Cuando se reciben datos cifrados en la tarjeta de vídeo, los datos se descifran automáticamente con una clave de descifrado asociada con una parte de memoria protegida en la que se escriben los datos descifrados. Entonces la GPU puede operar libremente sobre los datos descifrados. Si los datos van a moverse a una parte de memoria no protegida o memoria que no se encuentra físicamente en la tarjeta de vídeo, los datos pueden cifrarse con una clave de cifrado asociada y después moverse. Puesto que los datos pueden cifrarse cuando se transfieren, no es necesario almacenar en caché toda la imagen cifrada antes de enviarse.

En otra realización, se reciben datos cifrados en una tarjeta de vídeo y se disponen en una parte de memoria no protegida. Los datos pueden descifrarse entonces en una parte protegida de la memoria en la tarjeta de vídeo. La GPU puede operar entonces sobre los datos descifrados. Si los datos van a moverse a una parte no protegida de la memoria de la tarjeta de vídeo, o, fuera de la tarjeta de vídeo en la memoria de sistema, por ejemplo, los datos pueden volver a cifrarse.

Por tanto, las diversas realizaciones pueden proteger datos garantizando que cada vez que los datos se dispongan en un bus que pueda ser objeto de ataque (por ejemplo un bus PCI o un bus AGP), los datos están cifrados y por tanto protegidos.

Breve descripción de los dibujos

La figura 1 es un diagrama de bloques que muestra diversos componentes de una tarjeta de vídeo o gráfica ejemplar prevista para usarse en un sistema informático.

ES 2 327 325 T3

La figura 2 es un diagrama de bloques de un sistema informático ejemplar que puede emplear tarjetas de vídeo según la realización descrita.

La figura 3 es un diagrama de flujo que describe etapas en un procedimiento según una realización.

La figura 4 es un diagrama de flujo que describe etapas en un procedimiento según una realización.

La figura 5 es un diagrama de bloques que muestra diversos componentes de una tarjeta de vídeo o gráfica ejemplar según una realización.

La figura 6 es un diagrama de bloques que muestra diversos componentes de la tarjeta de vídeo de la figura 5.

La figura 7 es un diagrama de flujo que describe etapas en un procedimiento según una realización.

La figura 8 es un diagrama de bloques que es útil para entender ciertos aspectos de las realizaciones descritas.

La figura 9 es un diagrama de flujo que describe etapas en un procedimiento según una realización.

La figura 10 es un diagrama de bloques que muestra diversos componentes de una tarjeta de vídeo según una realización.

Descripción detallada

Sistema informático ejemplar

La figura 2 ilustra un ejemplo de un entorno 200 informático adecuado en el que pueden implementarse el sistema y procedimientos relacionados descritos a continuación.

Ha de observarse que el entorno 200 informático es tan sólo un ejemplo de un entorno informático adecuado y no pretende sugerir limitación alguna en cuanto al ámbito de uso o funcionalidad del sistema de procesamiento de medios. El entorno 200 informático tampoco debe interpretarse como que tiene dependencia o requisito alguno en relación a uno cualquiera de los componentes o una combinación de los mismos ilustrados en el entorno 200 informático ejemplar.

Las diversas realizaciones descritas pueden ser operativas con muchos otros entornos o configuraciones de sistema informático de propósito general o de propósito específico. Ejemplos de sistemas, entornos y/o configuraciones informáticos ampliamente conocidos que pueden ser adecuados para su uso con el sistema de procesamiento de medios incluyen, sin limitarse a, ordenadores personales, ordenadores servidores, clientes livianos, clientes pesados, dispositivos manuales o portátiles, sistemas de multiprocesador, sistemas basados en multiprocesador, módulos de codificadores, sistemas electrónicos de consumo programables, PC en red, miniordenadores, ordenadores centrales, entornos informáticos distribuidos que incluyen cualquiera de los anteriores sistemas o dispositivos, y similares.

En ciertas implementaciones, el sistema y los procedimientos relacionados pueden describirse perfectamente en el contexto general de instrucciones ejecutables por ordenador, tales como módulos de programa, que se ejecutan por un ordenador. Por lo general, los módulos de programa incluyen rutinas, programas, objetos, componentes, estructuras de datos, etc. que realizan tareas particulares o implementan tipos de datos abstractos particulares. Las realizaciones también pueden ponerse en práctica en entornos informáticos distribuidos en los que las tareas se realizan por dispositivos de procesamiento remotos que están enlazados a través de una red de comunicaciones. En un entorno informático distribuido, los módulos de programa pueden ubicarse en medios de almacenamiento informáticos tanto locales como remotos incluyendo dispositivos de almacenamiento de memoria.

Según la realización de ejemplo ilustrada de la figura 2, se muestra un sistema 200 informático que comprende uno o más procesadores o unidades 202 de procesamiento, una memoria 204 de sistema y un bus 206 que acopla varios componentes de sistema incluyendo la memoria 204 de sistema con el procesador 202.

El bus 206 está previsto para representar uno o más de cualquiera de varios tipos de estructuras de bus, incluyendo un bus de memoria o controlador de memoria, un bus periférico, un puerto gráfico acelerado y un procesador o bus local que usa cualquiera de una variedad de arquitecturas de bus. A modo de ejemplo, y no de limitación, tales arquitecturas incluyen bus de arquitectura estándar industrial (ISA), bus de arquitectura de microcanal (MCA), bus ISA mejorado (EISA), bus local de Video Electronics Standards Association (VESA) y bus de interconexión de componentes periféricos (PCI) también conocido como bus Mezzanine.

El ordenador 200 normalmente incluye una variedad de medios legibles por ordenador. Tales medios pueden ser cualquier medio disponible accesible local y/o remotamente por el ordenador 200, e incluye medios tanto volátiles como no volátiles, medios extraíbles y no extraíbles.

En la figura 2, la memoria 204 de sistema incluye medios legibles por ordenador en forma de memoria volátil, tal como memoria 210 de acceso aleatorio (RAM), y/o memoria no volátil, tal como memoria 208 de solo lectura (ROM).

ES 2 327 325 T3

Un sistema 212 básico de entrada/salida (BIOS), que contiene las rutinas básicas que ayudan a transferir información entre elementos dentro del ordenador 200, tal como durante el arranque, está almacenado en ROM 208. La RAM 210 normalmente contiene datos y/o módulos de programa a los que puede acceder de manera inmediata y/o que pueden operarse en la actualidad por una unidad (o unidades) 202 de procesamiento.

El ordenador 200 puede incluir además otros medios de almacenamiento informáticos extraíbles/no extraíbles, volátiles/no volátiles. A modo de ejemplo únicamente, la figura 2 ilustra una unidad 228 de disco duro para leer desde y escribir en un medio magnético no extraíble, no volátil (no mostrado y denominado habitualmente “disco duro”), una unidad 230 de disco magnético para leer desde y escribir en un disco 232 magnético extraíble, no volátil (por ejemplo, un “disco flexible”), y una unidad 234 de disco óptico para leer desde o escribir en un disco 236 óptico extraíble, no volátil, tal como un CD-ROM, DVD-ROM u otros medios ópticos. La unidad 228 de disco duro, la unidad 230 de disco magnético y la unidad 234 de disco óptico están conectadas, cada una, al bus 206 mediante una o más interfaces 226.

Las unidades y su medios legibles por ordenador asociados proporcionan almacenamiento no volátil de instrucciones legibles por ordenador, estructuras de datos, módulos de programa y otros datos para el ordenador 200. Aunque el entorno ejemplar descrito en este caso emplea un disco 228 duro, un disco 232 magnético extraíble y un disco 236 óptico extraíble, los expertos en la técnica apreciarán que también pueden usarse en el entorno operativo ejemplar otros tipos de medios legibles por ordenador que pueden almacenar datos accesibles por un ordenador, tal como cintas magnéticas, tarjetas de memoria *flash*, discos de vídeo digital, memorias de acceso aleatorio (RAM), memorias de solo lectura (ROM), y similares.

Varios módulos de programa pueden almacenarse en el disco 228 duro, disco 232 magnético, disco 236 óptico, ROM 208 o RAM 210, incluyendo, a modo de ejemplo, y no de limitación, un sistema 214 operativo, uno o más programas 216 de aplicación (por ejemplo, programa 224 de aplicación multimedia), otros módulos 218 de programa y datos 220 de programa. Un usuario puede introducir órdenes e información en el ordenador 200 a través de dispositivos de entrada tales como un teclado 238 y un dispositivo 240 de puntero (tal como un “ratón”). Otros dispositivos de entrada pueden incluir un dispositivo (dispositivos) 253 de entrada de audio/vídeo, un micrófono, un *joystick*, un mando de juego, una antena parabólica, un puerto serie, un escáner o similares (no mostrados). Estos y otros dispositivos de entrada están conectados a la unidad (unidades) 202 de procesamiento a través de la interfaz (interfaces) 242 de entrada que está(n) acoplada(s) al bus 206, aunque pueden estar conectados mediante otras estructuras de interfaz y bus, tales como un puerto paralelo, puerto de juego, o un bus serie universal (USB).

Un monitor 256 u otro tipo de dispositivo de visualización también está conectado al bus 206 a través de una interfaz, tal como un adaptador de vídeo o tarjeta 244 de vídeo/gráfica. Además del monitor, los ordenadores personales normalmente incluyen otros dispositivos de salida periféricos (no mostrados), tales como altavoces e impresoras, que pueden conectarse a través de la interfaz 246 periférica de salida.

El ordenador 200 puede operar en un entorno en red usando conexiones lógicas a uno o más ordenadores remotos, tales como un ordenador 250 remoto.

Como se muestra en la figura 2, el sistema 200 informático está acoplado en comunicación a dispositivos remotos (por ejemplo, el ordenador 250 remoto) a través de una red 251 de área local (LAN) y una red 252 de área amplia general (WAN). Tales entornos en red son habituales en oficinas, redes informáticas en empresas, intranets e Internet.

Cuando se usa en un entorno en red LAN, el ordenador 200 está conectado a la LAN 251 a través de una interfaz o adaptador 248 de red adecuado. Cuando se usa en un entorno de red WAN, el ordenador 200 normalmente incluye un módem 254 u otro medio para establecer comunicaciones sobre la WAN 252. El módem 254, que puede ser interno o externo, puede conectarse al bus 206 de sistema a través de la interfaz 242 de entrada de usuario, u otro mecanismo apropiado.

En un entorno en red, los módulos de programa ilustrados en relación con el ordenador 200 personal, o partes de los mismos, pueden almacenarse en un dispositivo de almacenamiento de memoria. A modo de ejemplo, y no de limitación, la figura 2 ilustra programas 216 de aplicación remotos residentes en un dispositivo de memoria del ordenador 250 remoto. Se apreciará que las conexiones de red mostradas y descritas son ejemplares y que pueden usarse otros medios de establecimiento de un enlace de comunicaciones entre los ordenadores.

Exposición general

En las realizaciones descritas a continuación, los datos previstos para su uso por una tarjeta de vídeo pueden cifrarse de tal manera que cada vez que los datos se proporcionen a un bus (por ejemplo el bus PCI o AGP) entre la tarjeta de vídeo y el sistema informático, los datos están cifrados. Esto es ventajoso porque dispositivos de “fisgoneo” o aplicaciones malintencionadas que adquieran el bus no pueden acceder a los datos en su forma no cifrada. Por tanto, cuando los datos se mueven desde la memoria de vídeo en la tarjeta de vídeo a la memoria del sistema (por ejemplo cuando está en correspondencia con la memoria física del sistema) y viceversa, los datos están en forma cifrada y por tanto protegidos.

Cuando los datos cifrados se transfieren desde la memoria del sistema a la tarjeta de vídeo para el procesamiento

ES 2 327 325 T3

por la unidad de procesador gráfico (GPU), pueden descifrarse y disponerse en memoria protegida en la tarjeta de vídeo. Por tanto, la GPU puede realizar operaciones sobre los datos no cifrados, que es para lo que la GPU es más adecuada.

5 En las realizaciones descritas a continuación, se describen dos implementaciones específicas que se dirigen a proteger datos cifrándolos siempre que van a desplazarse sobre un bus entre la tarjeta de vídeo y el sistema informático. Ha de apreciarse y entenderse que las realizaciones específicamente descritas son sólo ejemplos y que pueden realizarse variaciones de las implementaciones específicas sin alejarse del ámbito del contenido reivindicado.

10 La figura 3 es un diagrama de flujo que describe etapas en un procedimiento según una realización. Este procedimiento puede implementarse en cualquier hardware, software, *firmware* adecuado o combinación de los mismos. En este ejemplo particular, el procedimiento describe etapas de procesamiento ejemplares que se realizan cuando se mueven datos desde la tarjeta de vídeo al sistema o la memoria de sistema.

15 La etapa 300 proporciona datos no cifrados en la tarjeta de vídeo. Estos datos son los datos que normalmente se procesan por la GPU de la tarjeta de vídeo y se proporcionan a la memoria de la tarjeta de vídeo. La etapa 302 cifra los datos no cifrados en la tarjeta de vídeo. Esta etapa puede implementarse en respuesta a una petición de transferencia de los datos a la memoria del sistema. Por ejemplo, la CPU puede enviar una instrucción para mover los datos desde la tarjeta de vídeo a la memoria del sistema. Alternativamente, la GPU puede determinar que ya no necesita los datos y
20 por lo tanto inicia una transferencia a la memoria del sistema. Los datos pueden cifrarse en la tarjeta de vídeo usando cualquier técnica de cifrado adecuada. La etapa 304 envía entonces los datos cifrados sobre un bus entre la tarjeta de vídeo y el sistema. Buses ejemplares pueden ser el bus PCI y el bus AGP. La etapa 306 proporciona los datos cifrados a la memoria de sistema.

25 En el proceso anteriormente descrito, los datos que se mueven desde la tarjeta de vídeo a la memoria del sistema están cifrados y, por tanto, cualquier dispositivo de “fisgoneo” o aplicación malintencionada que intente adquirir el bus y robar o copiar los datos recibirá únicamente datos cifrados que, para los dispositivos de “fisgoneo” o aplicaciones malintencionadas, será matemáticamente inviable descifrar.

30 La figura 4 es un diagrama de flujo que describe etapas en un procedimiento según una realización. Este procedimiento puede implementarse en cualquier hardware, software, *firmware* adecuado o combinación de los mismos. En este ejemplo particular, el procedimiento describe etapas de procesamiento ejemplares que se realizan cuando se mueven datos desde el sistema o memoria de sistema a la tarjeta de vídeo.

35 La etapa 400 proporciona datos a la memoria de sistema. Tales datos normalmente son datos que se utilizan por la tarjeta de vídeo para el procesamiento. Los datos pueden proporcionarse a la memoria de sistema o bien como datos cifrados o bien como datos no cifrados. La etapa 402 puede cifrar los datos (si están no cifrados en la memoria de sistema) y puede enviar los datos a un bus entre la memoria de sistema y la tarjeta de vídeo. Esta etapa puede implementarse en respuesta a un deseo de procesar los datos en la tarjeta de vídeo. La etapa 404 recibe los datos
40 cifrados en la tarjeta de vídeo. La etapa 406 descifra entonces los datos cifrados en la tarjeta de vídeo y la etapa 408 proporciona los datos descifrados a la memoria de vídeo para su procesamiento por la GPU.

En el proceso anteriormente descrito, los datos que se mueven desde la memoria del sistema a la tarjeta de vídeo están cifrados y, por tanto, cualquier dispositivo de “fisgoneo” o aplicación malintencionada que intente adquirir el bus
45 y robar o copiar los datos recibirá únicamente datos cifrados que, para los dispositivos de “fisgoneo” o aplicaciones malintencionadas, será matemáticamente inviable descifrar.

Primera realización de tarjeta de vídeo ejemplar

50 La figura 5 muestra una tarjeta 500 de vídeo (o gráfica) ejemplar según una realización. La tarjeta 500 incluye un conector 502 de bus que se inserta en un puerto en un ordenador típico. La tarjeta 500 de vídeo también incluye un conector 504 de monitor (por ejemplo un enchufe de 15 pines) que recibe un cable que se conecta a un monitor. La tarjeta 500 de vídeo puede, aunque no es necesario, incluir un zócalo 506 de salida de vídeo digital (por ejemplo DVI) que puede usarse para enviar imágenes de vídeo a dispositivos de visualización digitales y similares.

55 Como la tarjeta de vídeo de la figura 1, la tarjeta 500 de vídeo comprende una unidad 508 de procesador gráfico (GPU), una memoria 510 de vídeo, un convertidor 512 digital a analógico de memoria de acceso aleatorio (RAMDAC), y software de controlador que puede estar incluido en el BIOS 514 de vídeo.

60 La GPU 508 es un chip de procesamiento gráfico dedicado que controla todos los aspectos de resolución, profundidad de color y todos los elementos asociados con la renderización de imágenes en la pantalla del monitor. El controlador de memoria (en ocasiones integrado en la GPU) gestiona la memoria en la tarjeta de vídeo. La unidad de procesamiento central o CPU del ordenador (no mostrada) envía un conjunto de instrucciones y datos de trazado, que se interpretan por el controlador propietario de la tarjeta gráfica y se ejecutan por la GPU 508 de la tarjeta. La GPU 508
65 realiza tales operaciones como dibujos y transferencias de mapas de bits, redimensionamiento y reposicionamiento de ventanas, trazado de líneas, ajuste a escala de fuentes y trazado de polígonos. La GPU puede entonces escribir los datos de cuadro en la memoria intermedia de cuadros (o memoria 510 de vídeo integrada).

ES 2 327 325 T3

La información en la memoria intermedia de cuadros de la memoria de vídeo es una imagen de lo que aparece en la pantalla, almacenada en un mapa de bits digital. El RAMDAC 512 se utiliza para convertir el mapa de bits digital en una forma que pueda usarse para la renderización en el monitor, como se ha descrito anteriormente.

Además de estos componentes, en esta realización, la tarjeta 500 de vídeo comprende un controlador 516 de memoria y un gestor 518 de claves. La tarjeta de vídeo también puede incluir un dispositivo 520 de descifrado. Estos componentes pueden implementarse en cualquier hardware, software, *firmware* adecuado o combinación de los mismos.

El controlador 516 de memoria recibe datos cifrados en la tarjeta de vídeo y descifra los datos en partes o regiones 510 protegidas de memoria de vídeo. Los datos descifrados están ahora en un estado en el que la GPU 508 puede operar sobre ellos. El controlador de memoria también puede ser responsable de garantizar que las transferencias de datos en la tarjeta de vídeo se realicen entre regiones protegidas o regiones que tienen un nivel de protección compatible. Esto es, a menudo durante el procesamiento de los datos en la tarjeta de vídeo, la GPU 508 operará sobre los datos en la memoria de vídeo (por ejemplo, realizando una operación de mezclado) y hará que los datos resultantes se escriban en una ubicación de memoria de vídeo diferente. En este caso, puede haber regiones de memoria de vídeo que no estén protegidas o que estén protegidas en un nivel diferente. En este caso, el controlador de memoria puede garantizar que las transferencias de datos dentro de la tarjeta de vídeo tengan lugar de una manera que garantice la protección de los datos no cifrados. Ejemplos de cómo puede hacerse esto se describen a continuación en más detalle.

El gestor 518 de claves controla las claves de cifrado y de descifrado que se usan para cifrar y descifrar datos en la tarjeta de vídeo. En una implementación, el gestor 518 de claves comprende un chip separado. El gestor 518 de claves está conectado en comunicación con el controlador de memoria y puede programar el controlador de memoria con las diversas claves de cifrado/descifrado. Ventajosamente, la comunicación entre el gestor 518 de claves y el controlador 516 de memoria tiene lugar a través de un canal de comunicación cifrado, autenticado, garantizando así que el gestor de claves es la única entidad que puede programar el controlador de memoria.

La tarjeta 500 de vídeo también puede incluir un dispositivo 520 de descifrado que está configurado o puede configurarse para descifrar datos. Por ejemplo, en algunos casos datos que no están cifrados en la tarjeta de vídeo pueden escribirse en una región de memoria que no está protegida. Por ejemplo, tales datos pueden escribirse en una denominada “superficie primaria” o superficie de escritorio que contiene datos que se usan por el RAMDAC 512 para renderizar una imagen en el monitor. En este caso, puede ser deseable proteger los datos no cifrados de modo que no estén sujetos a ataque. Por consiguiente, cuando los datos se mueven desde una región de memoria protegida a una región de memoria no protegida, el controlador 516 de memoria puede cifrar los datos. En este ejemplo, el gestor 518 de claves hace un seguimiento y conoce qué claves están asociadas con los datos cifrados en la región de memoria no protegida. Cuando los datos cifrados van a proporcionarse al RAMDAC 512, el gestor de claves puede dar instrucciones entonces al dispositivo 520 de descifrado acerca de qué clave o claves usar para descifrar los datos cifrados. Los datos cifrados se proporcionan entonces al dispositivo 520 de descifrado, se descifran y entonces se proporcionan al RAMDAC 512 para su procesamiento adicional.

Alternativamente, el dispositivo de descifrado puede eliminarse y el RAMDAC 512 puede estar configurado para realizar la funcionalidad de descifrado. En este caso, el gestor 518 de claves puede dar instrucciones al RAMDAC acerca de qué claves usar para efectuar el descifrado.

La figura 6 muestra, según una realización, componentes seleccionados de la tarjeta 500 de vídeo (figura 5) en más detalle, en general en 600. En este caso, el controlador 516 de memoria comprende un módulo 602 de descifrado que está configurado para recibir datos cifrados desde el bus (o bien PCI o bien AGP en este ejemplo) y descifrar los datos en la memoria 510 de vídeo. Además, se proporciona una tabla 604 de protección de memoria y una tabla 606 de decisión de equivalencia.

En este ejemplo, la tabla 604 de protección de memoria incluye una parte 604a de tabla que contiene entradas que asocian pares de claves de cifrado/descifrado con partes individuales de la memoria 510 de vídeo. Por ejemplo, el par de claves E_1/D_1 de cifrado/descifrado está asociado con la parte 608 de memoria, el par de claves E_2/D_2 de cifrado/descifrado está asociado con la parte 610 de memoria, etc. Cuando los datos descifrados en la memoria de vídeo van a escribirse en la memoria de sistema desde la tarjeta de vídeo, o en cualquier otro momento en el que los datos en la tarjeta de vídeo vayan a proporcionarse sobre el bus (por ejemplo el bus PCI o AGP), la CPU puede hacer que se cifren los datos con una de las claves de cifrado en la parte 604a de tabla. Los datos cifrados pueden disponerse entonces sobre el bus y proporcionarse, por ejemplo, a la memoria de sistema. Cuando el controlador 516 de memoria recibe datos cifrados desde la memoria de sistema que se han cifrado, por ejemplo, con la clave E_1 , el module 602 de descifrado puede localizar la clave D_1 de descifrado asociada y descifrar los datos en la parte 608 de memoria.

Puede haber varias partes protegidas de memoria 510 de vídeo. En el presente ejemplo, hay cuatro partes protegidas designadas como 608-614. Los datos no cifrados en estas partes protegidas pueden procesarse por la GPU 508 (figura 5) de la manera habitual. Las partes protegidas de la memoria de vídeo pueden protegerse mediante una lista de control de acceso. Por ejemplo, la tabla 604 de protección de memoria incluye una parte 604b de tabla que puede definir una lista de control de acceso para las diversas partes de memoria de vídeo. Cada parte de la memoria de vídeo puede tener definidas, en su lista de control de acceso asociada, qué entidades pueden acceder a la parte de memoria. Por tanto,

ES 2 327 325 T3

cualquier intento de acceso por entidades distintas a las contenidas en cada lista de control de acceso de la parte de memoria no se permitirá por el controlador de memoria.

5 Obsérvese también que la memoria 510 de vídeo puede incluir partes que no están protegidas. En este ejemplo, las partes 616-624 no están protegidas. Por consiguiente, no hay pares de claves de cifrado/descifrado asociados con estas partes de memoria.

10 La figura 7 es un diagrama de flujo que describe etapas en un procedimiento según una realización. El procedimiento puede implementarse en cualquier hardware, software, *firmware* adecuado o combinación de los mismos. En el ejemplo ilustrado, el procedimiento puede implementarse, al menos en parte, con un sistema tal como el descrito en conexión con la figura 6.

15 La etapa 700 proporciona uno o más pares de claves. Cada par de claves comprende una clave de cifrado y una clave de descifrado asociada. La etapa 702 asocia el par (pares) de claves con partes individuales de la memoria de vídeo. Puede utilizarse cualquier modo adecuado para asociar los pares de claves con las partes de la memoria de vídeo. En el ejemplo de la figura 6, se emplea una tabla de protección de memoria que tiene múltiples entradas diferentes. Las entradas de tabla individuales corresponden a un par de claves particular, y cada entrada de tabla está asociada con una parte de la memoria de vídeo.

20 La etapa 704 cifra datos no cifrados usando una clave de cifrado de un par de claves. Por ejemplo, si datos no cifrados en la memoria de vídeo van a escribirse en la memoria de sistema, la CPU puede hacer que los datos se cifren con una clave de cifrado que está asociada con la parte de la memoria de vídeo en la que se encuentran los datos. La etapa 706 envía entonces los datos cifrados sobre el bus a, por ejemplo, la memoria de sistema.

25 La etapa 708, que sigue a la etapa 702, descifra datos cifrados usando una clave de descifrado de un par de claves. Por ejemplo, supóngase que los datos están cifrados con una clave de cifrado y se proporcionan entonces a la memoria del sistema. Si la GPU necesita los datos cifrados, los datos pueden enviarse, por la CPU, a la tarjeta de vídeo. Al recibir los datos cifrados, el controlador de memoria puede localizar una clave de descifrado que está asociada con la clave de cifrado que se usó para cifrar los datos, y descifrar los datos usando la clave de descifrado. La etapa 710
30 puede proporcionar entonces los datos a una parte de memoria de vídeo asociada que está asociada con la clave de descifrado. De esta manera, los datos pueden estar protegidos siempre que vayan a proporcionarse sobre uno o más de los buses de sistema.

35 La tabla 606 de decisión de equivalencia (figura 6) se proporciona y se usa por el controlador 516 de memoria para garantizar que las transferencias de datos dentro de la memoria 510 de la tarjeta de vídeo se realizan de manera que se preserva la protección de los datos.

Como ejemplo, considérese la figura 8. A menudo cuando la GPU procesa datos en la tarjeta de vídeo, mueve los datos desde una ubicación de memoria a otra. Por ejemplo, cuando la GPU realiza una operación de mezclado, puede
40 tomar datos de dos partes de memoria diferentes en la tarjeta de vídeo, mezclar los datos y entonces escribir los datos mezclados resultantes en una parte de la memoria de vídeo totalmente diferente. Esto se indica esquemáticamente mediante la operación 800 GPU. Supóngase en este caso que la GPU está realizando una operación que tiene dos entradas y una salida. Cada una de las entradas corresponde a datos que están almacenados en una parte protegida de la memoria de vídeo. En este ejemplo, una de las entradas está almacenada en una parte protegida de memoria de vídeo que está asociada con la clave E_1 de cifrado, y la otra de las entradas está almacenada en una parte protegida de memoria de vídeo que está asociada con la clave E_2 de cifrado (por tanto las entradas E_1 y E_2). Una vez realizada la operación, los datos de salida resultantes van a almacenarse en una parte protegida de la memoria de vídeo asociada con la clave E_4 de cifrado. Ahora bien, siempre que la GPU realiza una transferencia de memoria como ésta, el controlador 516 de memoria se asegura de conocer de dónde proceden los datos y dónde van a almacenarse los datos.
50 El controlador de memoria realiza entonces una comprobación para garantizar que las partes de memoria de vídeo en las que van a almacenarse los datos de salida son consistentes, en cuanto al nivel de protección del que están dotados los datos, con las protecciones conferidas por las partes de memoria de las que proceden los datos constitutivos.

55 En este ejemplo, la consistencia de la protección puede imponerse averiguando si las claves asociadas con las partes de memoria que contenían los datos de entrada son las mismas que o equivalentes a la clave asociada con la parte de memoria que va a contener los datos de salida. La equivalencia, en este ejemplo, puede determinarse basándose en las restricciones asociadas con las partes de memoria, es decir, si hay el mismo nivel de protección que entre las diferentes partes de memoria. Las claves podrían no ser equivalentes si, por ejemplo, hay dos programas diferentes con dos contenidos diferentes. Por ejemplo, supóngase que cada contenido está cifrado con una clave diferente y que
60 los programas asociados con las claves no pueden compartir contenido. En este ejemplo, las claves de cifrado no son equivalentes.

Una forma de averiguar la equivalencia de claves asociadas con las diversas partes de memoria es usar una tabla de decisión de equivalencia como la tabla 606. En la misma, la tabla define una matriz que contiene entradas para cada una de las claves. Una "X" en una de las celdas de la matriz indica que las claves son equivalentes. Por ejemplo, consultando en primer lugar la clave E_1 , esta clave es equivalente a las claves E_1 , E_2 y E_4 . Por tanto, como se muestra mejor en la figura 6, los datos pueden transferirse libremente entre las partes 608, 610, y 614 de memoria. La clave E_1 , sin embargo, no es equivalente a la clave E_3 . Por tanto, los datos no pueden transferirse desde la parte 608 de memoria

a la parte 612 de memoria. En el ejemplo de la figura 8, puesto que los datos de entrada proceden de las partes 608 y 610 de memoria y los datos resultantes van a almacenarse en la parte 614 de memoria, la equivalencia no es un problema y la operación puede tener lugar.

5 La figura 9 es un diagrama de flujo que describe etapas en un procedimiento según una realización. El procedimiento puede implementarse en cualquier hardware, software, *firmware* adecuado o combinación de los mismos. En el ejemplo ilustrado, el procedimiento puede implementarse, al menos en parte, con un sistema como el descrito en conexión con la figura 6.

10 La etapa 900 lee datos desde las partes de memoria de vídeo en una tarjeta de vídeo. Esta etapa puede implementarse por la GPU. La etapa 902 registra pares de claves asociados con las partes de memoria de vídeo desde las que se han leído los datos. Esta etapa puede implementarse por el controlador de memoria. La etapa 904 averigua si los pares de claves registrados son equivalentes a un par de claves asociado con una parte de memoria de vídeo que va a servir como destino para los datos de salida que resulten de la operación de la GPU. Si los pares de claves son
15 equivalentes, entonces la etapa 906 proporciona los datos de salida a la parte de destino de la memoria de vídeo. Si, por otra parte, los pares de claves no son equivalentes, entonces la etapa 908 busca una parte de destino de memoria de vídeo que tenga un par de claves equivalente o, posiblemente, la etapa 910 emite datos en blanco al destino de memoria especificado.

20 *Segunda realización de tarjeta de vídeo ejemplar*

La figura 10 muestra una tarjeta 1000 de vídeo (o gráfica) ejemplar según otra realización. En este caso, una parte de la figura se designa “Tarjeta de vídeo” y una parte de la figura se designa “Sistema”. Tal notación pretende designar aquellos componentes que se encuentran, respectivamente, en la tarjeta de vídeo y el sistema. El lado “Sistema” de la
25 figura incluye una CPU 1002 y una memoria 1004 de sistema. Obsérvese también que en la memoria 1004 de sistema hay datos 1006 previstos para usarse en la tarjeta 1000 de vídeo.

En esta realización, los datos 1006 que se encuentran en la memoria 1004 de sistema no están cifrados. La CPU 1002 está configurada para cifrar los datos siempre que vayan a enviarse sobre el bus a la tarjeta de vídeo. Esto
30 normalmente se realiza por la operación de CPU conocida como una “cifrar transferencia de memoria”. Por tanto, los datos cifrados se disponen ahora en una parte 624 no protegida de la memoria 510 de vídeo. En este ejemplo, los datos podrían cifrarse por la CPU con la clave E_1 de cifrado.

Recuérdese que la GPU 508 está configurada para procesar datos no cifrados. Por consiguiente, cuando la GPU 508 desea procesar datos cifrados en la parte no protegida de la memoria 510 de vídeo, los datos deben descifrarse. En esta
35 realización, la GPU 508 contiene funcionalidad de descifrado que está representada en la figura mediante el módulo 602 de descifrado. El gestor 518 de claves está conectado en comunicación con la GPU 508 y puede programar la GPU con las claves de descifrado apropiadas. Por supuesto, la comunicación entre el gestor de claves y la GPU puede tener lugar a través de un enlace autenticado.

40 Por consiguiente, cuando la GPU desea operar sobre datos 1006 cifrados en la memoria de vídeo, puede acceder a los datos cifrados y descifrarlos en una parte protegida de la memoria de vídeo, en este caso la parte 614 protegida. El descifrado puede tener lugar mediante una operación conocida como “descifrar transferencia de memoria”. Ahora que estos y otros datos están en partes protegidas de la memoria de vídeo, la GPU puede operar libremente sobre ello
45 de la manera habitual.

En esta realización, el controlador 516 de memoria puede imponer la protección de la memoria controlando el acceso a las partes protegidas de la memoria de vídeo mediante una lista de control de acceso (no mostrada específicamente). Por tanto, cada vez que una entidad como la GPU 508 (o partes de la misma) desea acceder a partes protegidas
50 de la memoria de vídeo, obtendrá acceso a través del controlador 516 de memoria. La lista de control de acceso puede describir qué clase de protección tienen los datos y qué entidades pueden acceder a los mismos. Por ejemplo, la lista de control de acceso puede definir qué partes de la GPU pueden acceder a la memoria protegida y una o más aplicaciones que “poseen” la memoria protegida y por tanto tener acceso.

55 Ahora bien, cuando los datos en las partes protegidas de la memoria de vídeo van a moverse a una parte no protegida de la memoria de vídeo, pueden cifrarse. Por ejemplo, supóngase que los datos en la parte 614 de memoria van a moverse a la parte 620 de memoria no protegida (que, en este ejemplo, corresponde a la superficie de escritorio que va a procesarse por el RAMDAC para la renderización). La GPU 508 puede acceder a la parte de memoria protegida a través del controlador 516 de memoria y realizar una operación de cifrado de transferencia de memoria
60 para cifrar los datos en la parte 620 de memoria. Cuando los datos cifrados van a proporcionarse al RAMDAC para su procesamiento, el gestor 518 de claves puede comunicarse con el dispositivo 520 de descifrado y proporcionar al dispositivo de descifrado la clave de descifrado apropiada para descifrar los datos cifrados en la parte 620 de memoria. Tras descifrar los datos, el dispositivo 520 de descifrado puede proporcionar los datos al RAMDAC para su procesamiento de la manera habitual.

65 Obsérvese también que el controlador 516 de memoria también comprende una tabla de decisión de equivalencia (no designada específicamente). Esta tabla es similar, en concepto, a la tabla comentada en conexión con las figuras 6 y 8. En este caso, sin embargo, la tabla se usa para garantizar que las listas de control de acceso que están asociadas

con cada una de las partes de memoria de la memoria 510 de vídeo son equivalentes cuando van a transferirse datos entre las mismas. Por ejemplo, supóngase que los datos en la parte 614 de memoria van a transferirse a la parte 620 de memoria. Por consiguiente, en este ejemplo, el controlador 516 de memoria realiza una comprobación para garantizar que la lista de control de acceso asociada con la parte 614 de memoria es equivalente a la lista de control de acceso asociada con la parte 620 de memoria. Si es así, se realiza la transferencia de datos.

Conclusión

Las diversas realizaciones descritas anteriormente pueden garantizar que los datos previstos para su uso por una tarjeta de vídeo pueden cifrarse de manera que cada vez que los datos se proporcionan a un bus (por ejemplo el bus PCI o AGP) entre la tarjeta de vídeo y el sistema informático, los datos están cifrados. Por tanto, dispositivos de “fiscgoneo” o aplicaciones malintencionadas que adquieran el bus no pueden acceder a los datos y por tanto robar los datos en su forma no cifrada. Por consiguiente, puede proporcionarse otro nivel de protección para contenido que va a procesarse por la tarjeta de vídeo del sistema.

Aunque la invención se ha descrito en lenguaje específico de características estructurales y/o etapas metodológicas, ha de entenderse que la invención definida en las reivindicaciones adjuntas no está limitada necesariamente a las características o etapas específicas descritas. Más bien, las características y etapas específicas se dan a conocer como formas preferidas de implementar la invención reivindicada.

REIVINDICACIONES

1. Un procedimiento para proporcionar datos desde una memoria (204, 1004) de un sistema informático a una tarjeta de vídeo para su procesamiento, comprendiendo el procedimiento:
 - proporcionar (400) datos a la memoria de sistema que es otra memoria distinta a una memoria (510) de tarjeta de vídeo;
 - cifrar (402) los datos que están almacenados en la memoria de sistema;
 - transferir (402) los datos cifrados sobre un bus (206) a la tarjeta (500, 1000) de vídeo;
 - descifrar (406) los datos cifrados en la tarjeta de vídeo, teniendo lugar tal descifrado independientemente de si una unidad (508) de procesador gráfico, GPU, en la tarjeta necesita procesar los datos cifrados; y
 - almacenar (408) datos descifrados en partes (608-614) protegidas de la memoria de la tarjeta de vídeo para su procesamiento por la GPU, teniendo asociada cada parte protegida una clave de cifrado individual;
 - en el que el procedimiento comprende además cifrar datos no cifrados que están almacenados en las partes protegidas de la memoria en la tarjeta de vídeo usando claves de cifrado asociadas respectivas de manera que los datos cifrados pueden moverse a una parte (616 a 624) no protegida de la memoria en la tarjeta de vídeo; y
 - en el que el procedimiento comprende además transferir datos no cifrados en la tarjeta de vídeo entre partes protegidas que tienen un nivel de protección compatible, en el que la compatibilidad se determina basándose al menos en parte en las claves de cifrado asociadas con las respectivas partes protegidas.
2. El procedimiento según la reivindicación 1, en el que la acción de descifrado se realiza de manera automática siempre que se transfieran datos cifrados a la tarjeta de vídeo.
3. El procedimiento según la reivindicación 1 ó 2, en el que la acción de descifrado comprende descifrar los datos cifrados usando una clave de descifrado que está asociada con una parte de memoria en la que van a almacenarse los datos no cifrados.
4. El procedimiento según la reivindicación 1, que comprende:
 - proporcionar la GPU en la tarjeta de vídeo para procesar datos que van a renderizarse en un monitor;
 - proporcionar la memoria de la tarjeta de vídeo a la tarjeta de vídeo asociada de manera operativa con la GPU para contener datos que van a procesarse o se han procesado por la GPU;
 - proporcionar un convertidor (512) de visualización para convertir datos digitales en señales para su uso en la renderización de los datos en el monitor.
5. El procedimiento según la reivindicación 4, en el que la compatibilidad se determina por el controlador de memoria, basándose al menos en parte, en listas (604b) de control de acceso que están asociadas con partes protegidas individuales.
6. El procedimiento según las reivindicaciones 4 ó 5, que comprende además proporcionar un gestor (518) de claves en la tarjeta de vídeo para controlar uno o más pares (604a) de claves de cifrado/descifrado que se usan para cifrar y descifrar respectivamente datos en la tarjeta de vídeo.
7. El procedimiento según la reivindicación 6, en el que la acción de proporcionar el gestor de claves comprende proporcionar un chip de circuito integrado.
8. El procedimiento según la reivindicación 6, en el que la acción de proporcionar el gestor de claves comprende enlazar en comunicación el gestor de claves con el controlador de memoria de manera que el gestor de claves pueda programar el controlador de memoria para cifrar y descifrar datos en la tarjeta de vídeo.
9. El procedimiento según la reivindicación 1, que comprende además proporcionar un dispositivo de descifrado asociado de manera operativa con la memoria en la tarjeta de vídeo y configurado para recibir datos cifrados desde la memoria y descifrar los datos cifrados de manera que los datos descifrados puedan proporcionarse a un convertidor (512) de visualización para su procesamiento.
10. El procedimiento según la reivindicación 9, que comprende además proporcionar un gestor de claves para controlar uno o más pares (604a) de claves de cifrado/descifrado que se usan para cifrar y descifrar respectivamente datos en la tarjeta de vídeo, estando configurado el gestor de claves para dar instrucciones al dispositivo de descifrado acerca de qué clave usar para descifrar datos cifrados.

11. El procedimiento según una de las reivindicaciones 1 y 4 a 10, en el que proporcionar el controlador de memoria comprende proporcionar un controlador de memoria que comprende una tabla de protección de memoria que tiene una o más entradas de tabla, asociando las entradas de tabla individuales al menos un par (604a) de claves de cifrado/descifrado con una parte de memoria que va a contener datos que pueden cifrarse y descifrarse usando el par de claves asociado con la parte de memoria.

12. El procedimiento según una de las reivindicaciones 1 y 4 a 10, en el que proporcionar el controlador de memoria comprende proporcionar un controlador de memoria que comprende una o más listas (604b) de control de acceso, estando las listas de control de acceso individuales asociadas con partes individuales de la memoria y definiendo qué entidades pueden acceder a una parte de memoria particular.

13. El procedimiento según la reivindicación 1, que comprende además:
disponer los datos cifrados en una parte no protegida de memoria en la tarjeta de vídeo; y
disponer los datos descifrados en las partes protegidas de memoria en la tarjeta de vídeo.

14. El procedimiento según la reivindicación 13, en el que la acción de descifrado se realiza por la unidad de procesador gráfico en la tarjeta de vídeo.

15. El procedimiento según la reivindicación 13 ó 14, que comprende además programar la unidad de procesador gráfico en la tarjeta de vídeo con capacidades de descifrado suficientes para permitir a la unidad de procesador gráfico descifrar datos cifrados que se encuentran en la parte de memoria no protegida de la tarjeta de vídeo.

16. El procedimiento según la reivindicación 15, en el que la acción de programación se realiza por un gestor de claves en la tarjeta de vídeo, estando el gestor de claves configurado para gestionar una o más claves de cifrado/descifrado.

17. El procedimiento según la reivindicación 16, en el que el gestor de claves comprende un chip de circuito integrado.

18. El procedimiento según una de las reivindicaciones 13 a 17, que comprende además proteger las partes protegidas de la memoria en la tarjeta de vídeo usando el controlador de memoria.

19. El procedimiento según la reivindicación 18, en el que la acción de protección se realiza usando, al menos en parte, una o más listas de control de acceso que definen la protección para las partes protegidas de la memoria de vídeo.

20. El procedimiento según una de las reivindicaciones 13 a 19, que comprende además, tras dicha acción de disposición:

operar sobre los datos en la parte protegida de la memoria de vídeo para proporcionar datos resultantes;
cifrar los datos resultantes; y
disponer los datos resultantes cifrados en una parte protegida de la memoria de vídeo.

21. El procedimiento según la reivindicación 20, en el que dichas acciones de cifrar los datos resultantes y disponer los datos resultantes se realizan, al menos en parte, por una unidad de procesador gráfico en la tarjeta de vídeo.

22. El procedimiento según la reivindicación 21, que comprende además:

descifrar los datos resultantes cifrados usando un dispositivo de descifrado en la tarjeta de vídeo; y
tras dicho descifrado de los datos resultantes cifrados, proporcionar los datos resultantes descifrados a un convertidor de visualización para su procesamiento adicional.

23. El procedimiento según la reivindicación 1 ó 4, en el que la GPU comprende funcionalidad de cifrado y descifrado suficiente para cifrar y descifrar datos en la tarjeta de vídeo; en el que la memoria comprende la parte protegida y no protegida; en el que el convertidor de visualización está proporcionado en la tarjeta de vídeo; y en el que proporcionar el controlador de memoria comprende proporcionar, en la tarjeta de vídeo, el controlador de memoria configurado para proteger las partes protegidas de la memoria de vídeo.

24. El procedimiento según la reivindicación 23, en el que proporcionar el controlador de memoria comprende proporcionar un controlador de memoria que comprende una o más listas de control de acceso que se utilizan para proteger las partes protegidas de la memoria.

ES 2 327 325 T3

25. El procedimiento según la reivindicación 24, en el que las listas de control de acceso definen entidades que pueden acceder a las partes protegidas de la memoria de vídeo.

26. El procedimiento según una de las reivindicaciones 23 a 25, que comprende además proporcionar un gestor de claves en la tarjeta de vídeo configurado para programar la GPU con la funcionalidad de cifrado/descifrado.

27. El procedimiento según una de las reivindicaciones 23 a 25, en el que proporcionar la GPU comprende proporcionar una GPU configurada para descifrar datos en partes no protegidas de la memoria en partes protegidas de la memoria.

28. El procedimiento según una de las reivindicaciones 23 a 25, en el que proporcionar la GPU comprende proporcionar una GPU configurada para cifrar datos en partes protegidas de la memoria en partes no protegidas de la memoria.

29. El procedimiento según la reivindicación 1, que comprende además:

cifrar (302) los datos no cifrados en las partes de protección de la tarjeta de vídeo; y

transferir (304) los datos cifrados fuera de la tarjeta de vídeo.

30. El procedimiento según la reivindicación 29, en el que la parte de memoria está protegida con una lista de control de acceso.

31. El procedimiento según la reivindicación 29 ó 30, en el que la acción de proporcionar comprende proporcionar los datos no cifrados a una parte de memoria de vídeo que tiene un par (604a) de claves de cifrado/descifrado asociado que puede usarse para cifrar y descifrar respectivamente los datos proporcionados en dicha parte de memoria de vídeo.

32. El procedimiento según una de las reivindicaciones 29 a 31, en el que la acción de cifrado se realiza cada vez que los datos no cifrados van a proporcionarse a la memoria (1004) de sistema desde la tarjeta de vídeo.

33. El procedimiento según una de las reivindicaciones 29 a 32, en el que la acción de cifrado se realiza cada vez que los datos no cifrados van a proporcionarse a un bus (206) conectado entre la tarjeta de vídeo y una unidad (202, 1002) de procesador central del sistema informático, comprendiendo la tarjeta de vídeo parte del sistema informático.

34. El procedimiento según la reivindicación 1 ó 29, que comprende:

proporcionar uno o más pares de claves, comprendiendo los pares de claves individuales una clave de cifrado que puede usarse para cifrar datos y una clave de descifrado que puede usarse para descifrar datos cifrados con la clave de cifrado; y

asociar pares de claves individuales con las partes protegidas individuales de memoria que comprenden parte de la memoria de tarjeta de vídeo.

35. El procedimiento según la reivindicación 34, en el que dichas acciones de proporcionar y asociar se realizan en la tarjeta de vídeo.

36. El procedimiento según la reivindicación 34 ó 35, en el que dicha acción de asociación comprende definir una tabla en la tarjeta de vídeo, teniendo la tabla entradas individuales que asocian pares de claves individuales con las partes protegidas individuales de la memoria.

37. El procedimiento según la reivindicación 36, en el que la tabla se define como parte del controlador de memoria en la tarjeta de vídeo.

38. El procedimiento según una de las reivindicaciones 34 a 37, que comprende además usar una clave de cifrado para cifrar datos que están almacenados en una parte asociada de la memoria.

39. El procedimiento según la reivindicación 38, en el que la acción de usar se realiza antes de transferir los datos fuera de la tarjeta de vídeo.

40. El procedimiento según la reivindicación 39, en el que la acción de usar se realiza antes de transferir los datos a la parte no protegida de la memoria en la tarjeta de vídeo.

41. El procedimiento según una de las reivindicaciones 34 a 40, que comprende además usar una clave de descifrado para descifrar datos cifrados que se han recibido sobre el bus externo a la tarjeta de vídeo.

42. El procedimiento según la reivindicación 41, que comprende además proporcionar los datos descifrados a una parte de la memoria asociada con la clave de descifrado que se usó para descifrar los datos cifrados.

ES 2 327 325 T3

43. El procedimiento según la reivindicación 1 ó 29, que comprende:

proporcionar datos a una primera parte de memoria en la tarjeta de vídeo, teniendo la primera parte de la memoria un primer nivel de protección;

operar sobre los datos para proporcionar datos resultantes en la tarjeta de vídeo;

determinar si una segunda parte de la memoria en la tarjeta de vídeo tiene un nivel de protección que es compatible con el primer nivel de protección; y

si los niveles primero y segundo de protección son compatibles, proporcionar los datos resultantes a la segunda parte de la memoria en la tarjeta de vídeo.

44. El procedimiento según la reivindicación 43, en el que los niveles de protección están definidos, al menos en parte, por las claves de cifrado asociadas con las partes de memoria primera y segunda.

45. El procedimiento según la reivindicación 43 ó 44, en el que los niveles de protección están definidos, al menos en parte, por listas de control de acceso asociadas con las partes de memoria primera y segunda.

46. El procedimiento según una de las reivindicaciones 43 a 45, en el que la acción de operar se realiza por la unidad de procesador gráfico que comprende parte de la tarjeta de vídeo.

47. El procedimiento según una de las reivindicaciones 43 a 45, en el que los datos resultantes son diferentes de los datos que se proporcionaron a la primera parte de la memoria en la tarjeta de vídeo.

48. El procedimiento según la reivindicación 1 ó 29, que comprende:

leer datos de una o más partes de memoria en la tarjeta de vídeo, teniendo las partes individuales de la memoria un par de claves de cifrado/descifrado asociadas;

grabar pares de claves asociados con las partes de memoria desde las que se han leído los datos;

operar sobre los datos leídos desde la una o más partes de la memoria para proporcionar datos de salida;

averiguar si los pares de claves asociados con las partes de memoria desde las que se han leído los datos son equivalentes a un par de claves asociado con una parte de memoria de vídeo que va a servir como destino para los datos de salida; y

si los pares de claves son equivalentes, proporcionar los datos de salida a la parte de memoria de vídeo de destino.

49. El procedimiento según la reivindicación 1 ó 29, que comprende:

recibir, en partes de memoria no protegidas de la tarjeta de vídeo, datos cifrados previstos para procesarse por la GPU en la tarjeta;

descifrar los datos cifrados en las partes de memoria protegidas de la tarjeta de vídeo; y

cifrar datos no cifrados en partes de memoria protegidas de la tarjeta de vídeo en partes de memoria no protegidas de la tarjeta de vídeo.

50. El procedimiento según la reivindicación 49, en el que las acciones de cifrado y descifrado se realizan por la GPU.

51. El procedimiento según la reivindicación 49 ó 50, en el que partes de memoria protegidas están protegidas por el controlador de memoria usando una o más listas de control de acceso que definen qué entidades pueden acceder a las partes de memoria protegidas.

52. El procedimiento según una de las reivindicaciones 49 a 51, que comprende además cifrar datos en la tarjeta de vídeo cada vez que los datos van a proporcionarse desde la tarjeta de vídeo y a un bus conectado con una unidad de procesador central, CPU, del sistema informático.

53. El procedimiento según una de las reivindicaciones 49 a 52, que comprende además cifrar datos en la tarjeta de vídeo cada vez que los datos van a proporcionarse desde la tarjeta de vídeo y a un bus conectado con una memoria del sistema informático.

54. Uno o más medios legibles por ordenador que tienen instrucciones legibles por ordenador en los mismos que, cuando se ejecutan por uno o más procesadores, hacen que el uno o más procesadores realicen el procedimiento según una de las reivindicaciones 1 a 53.

ES 2 327 325 T3

55. Una tarjeta (500, 1000) de vídeo que comprende:

una unidad (508) de procesador gráfico, GPU, para procesar datos que van a renderizarse en un monitor;

5 memoria (510) asociada de manera operativa con la unidad de procesador gráfico para contener datos que van a procesarse o se han procesado por la GPU;

10 un convertidor (512) de visualización para convertir datos digitales en señales para su uso en la renderización de los datos en el monitor; y

15 un controlador (516) de memoria configurado para actuar según el procedimiento de una de las reivindicaciones 1 a 53.

15

20

25

30

35

40

45

50

55

60

65

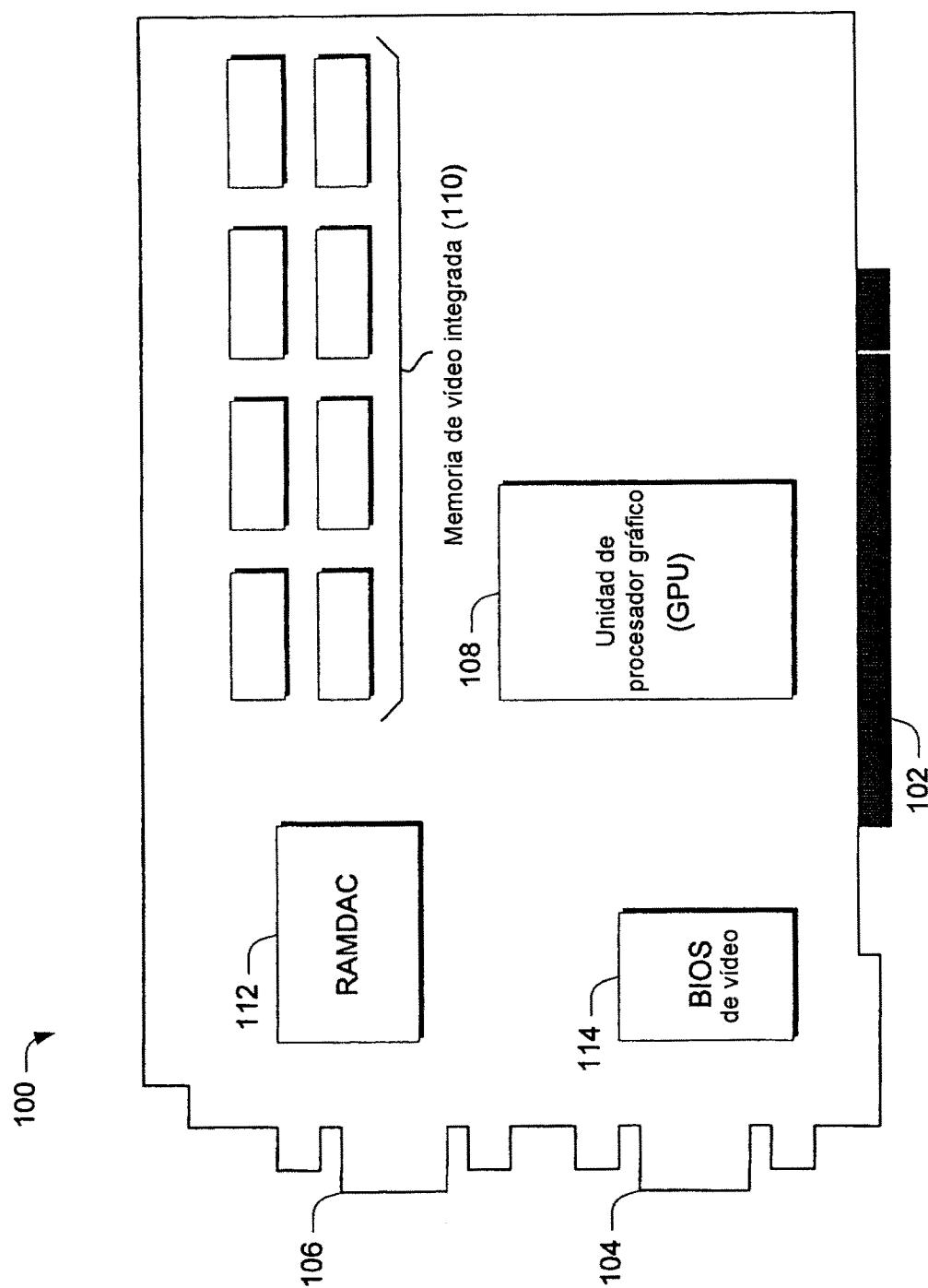
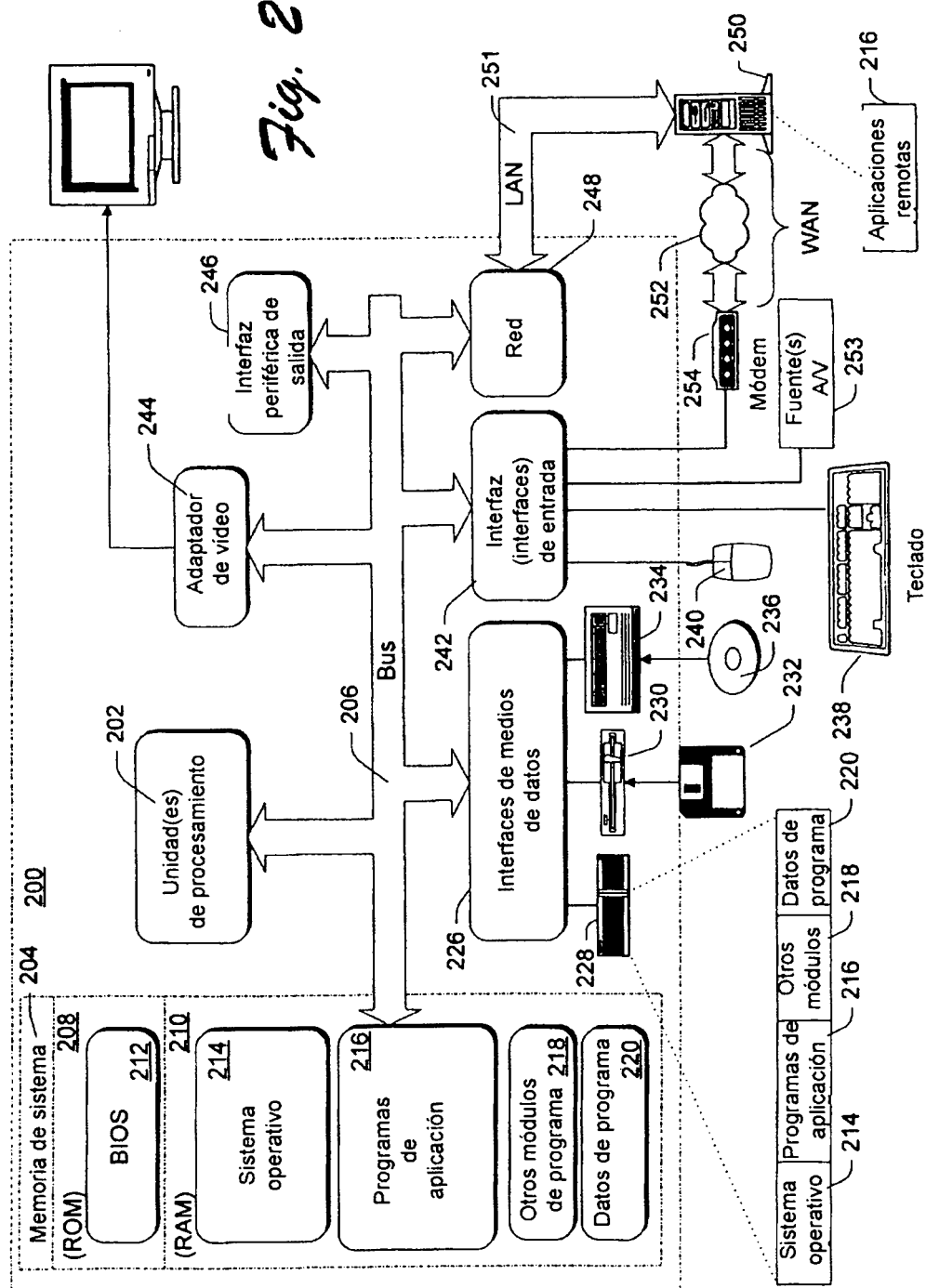


Fig. 1

Fig. 2



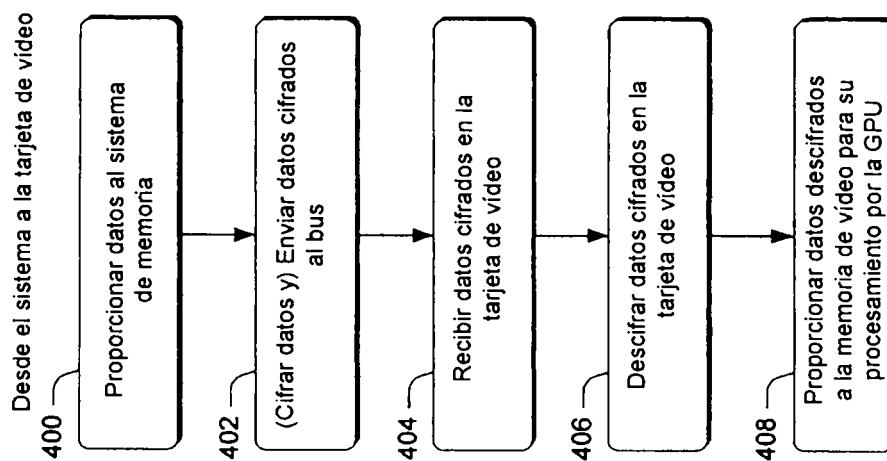


Fig. 4

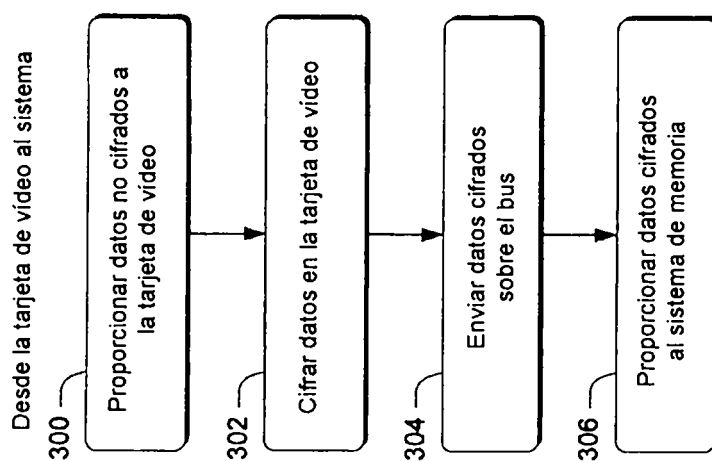


Fig. 3

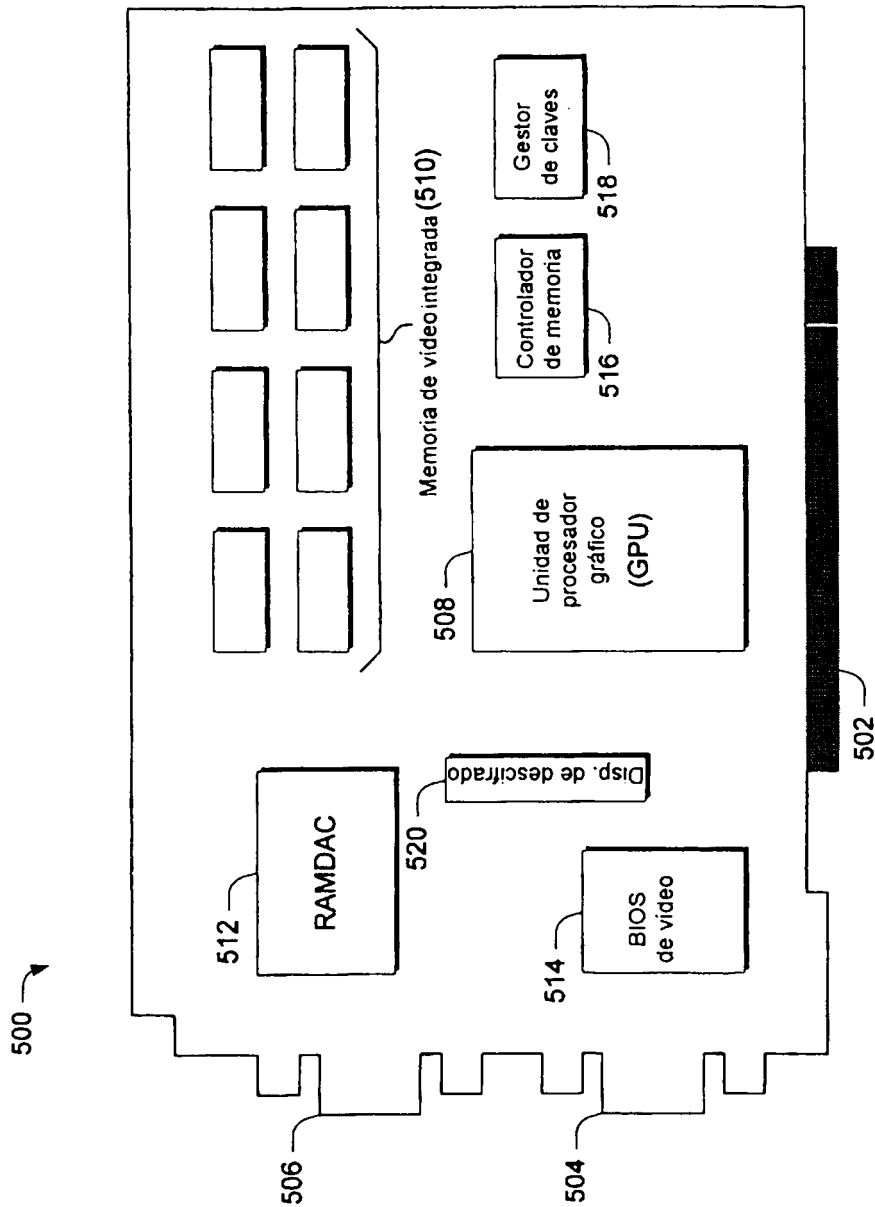


Fig. 5

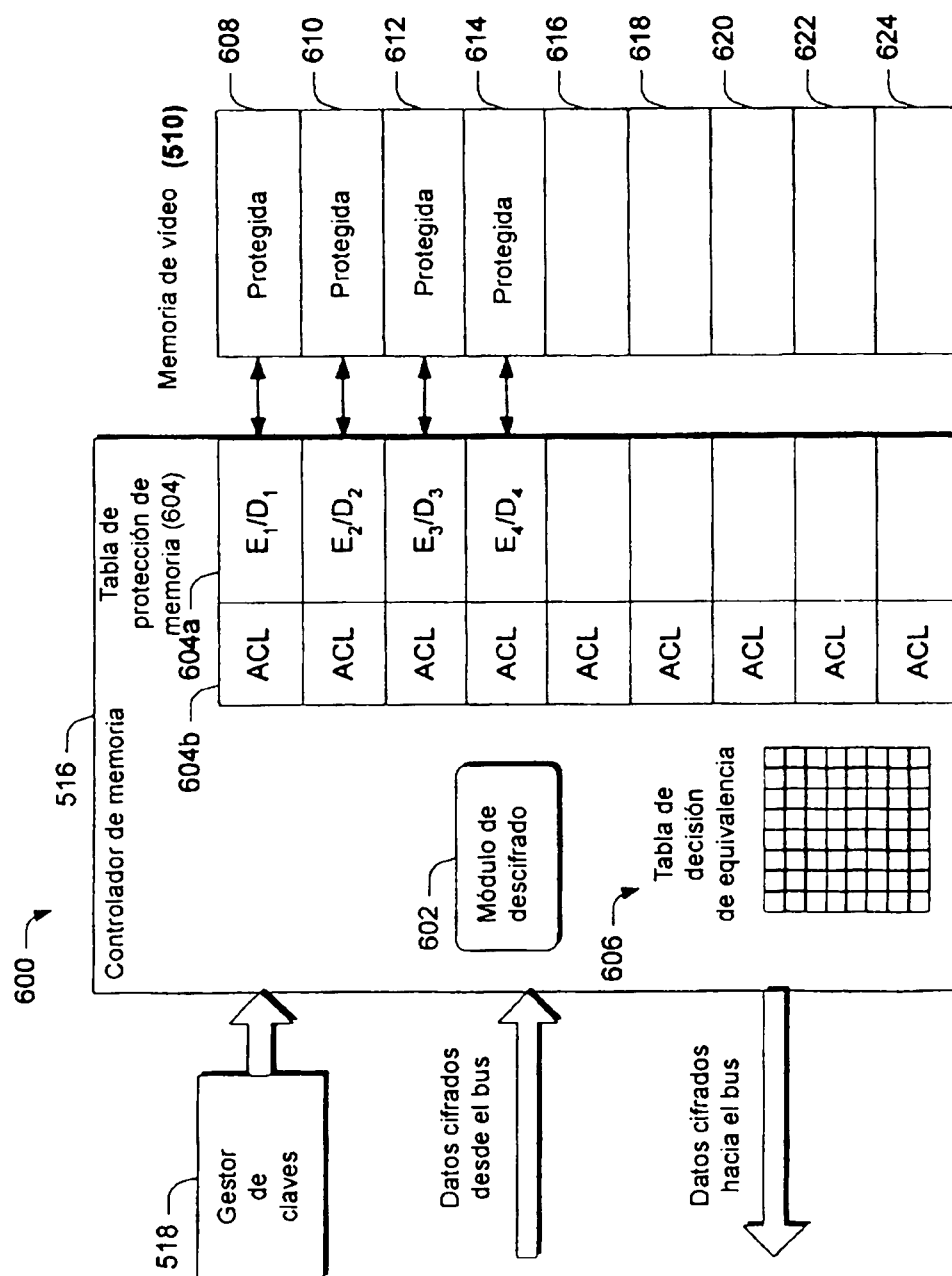


Fig. 6

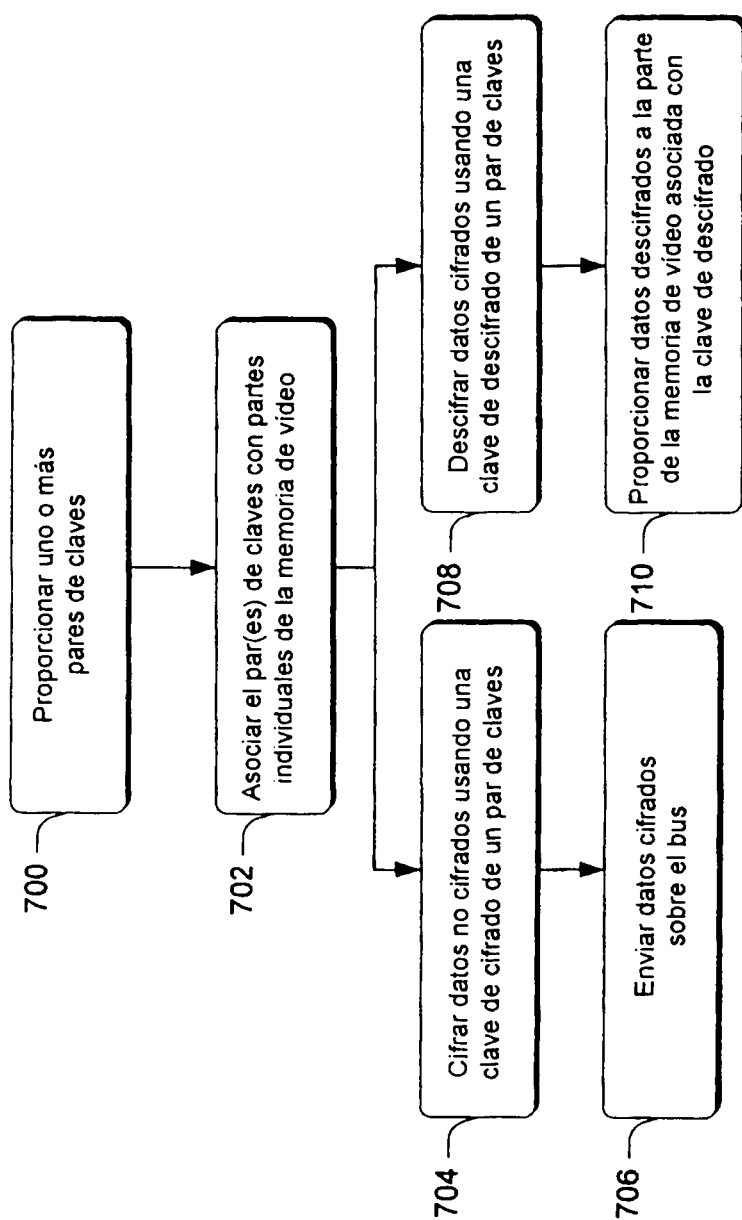


Fig. 7

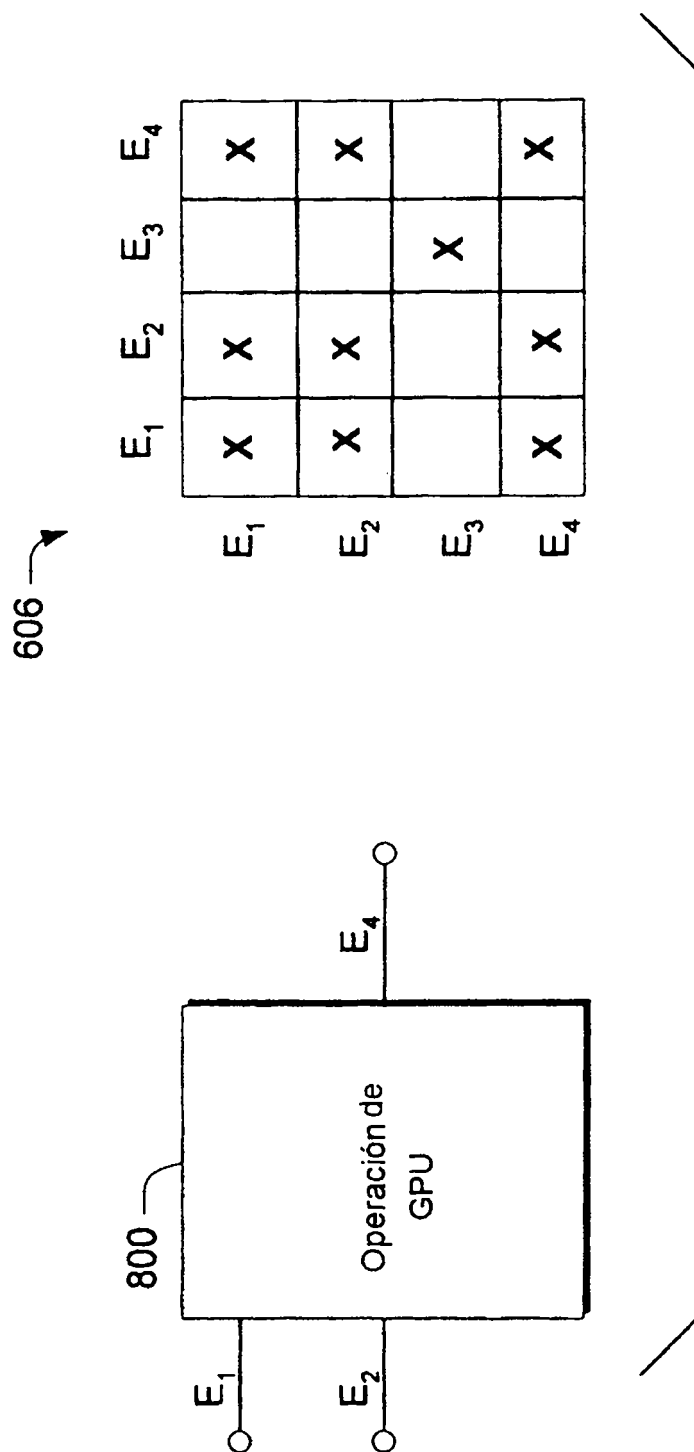


Fig. 8

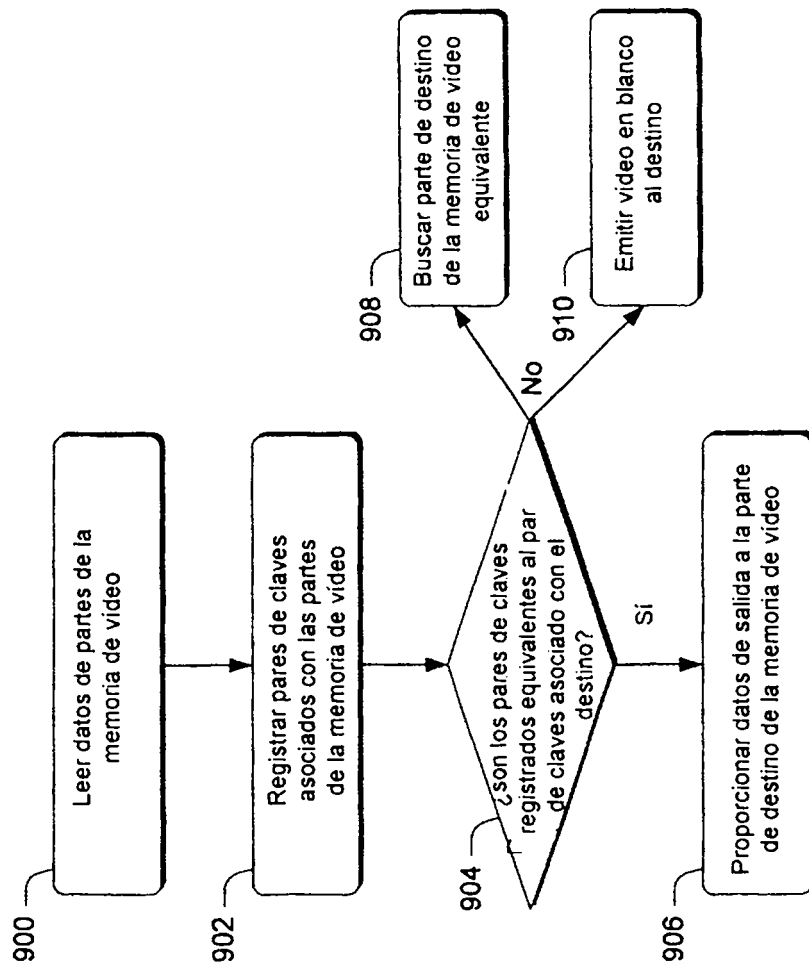


Fig. 9

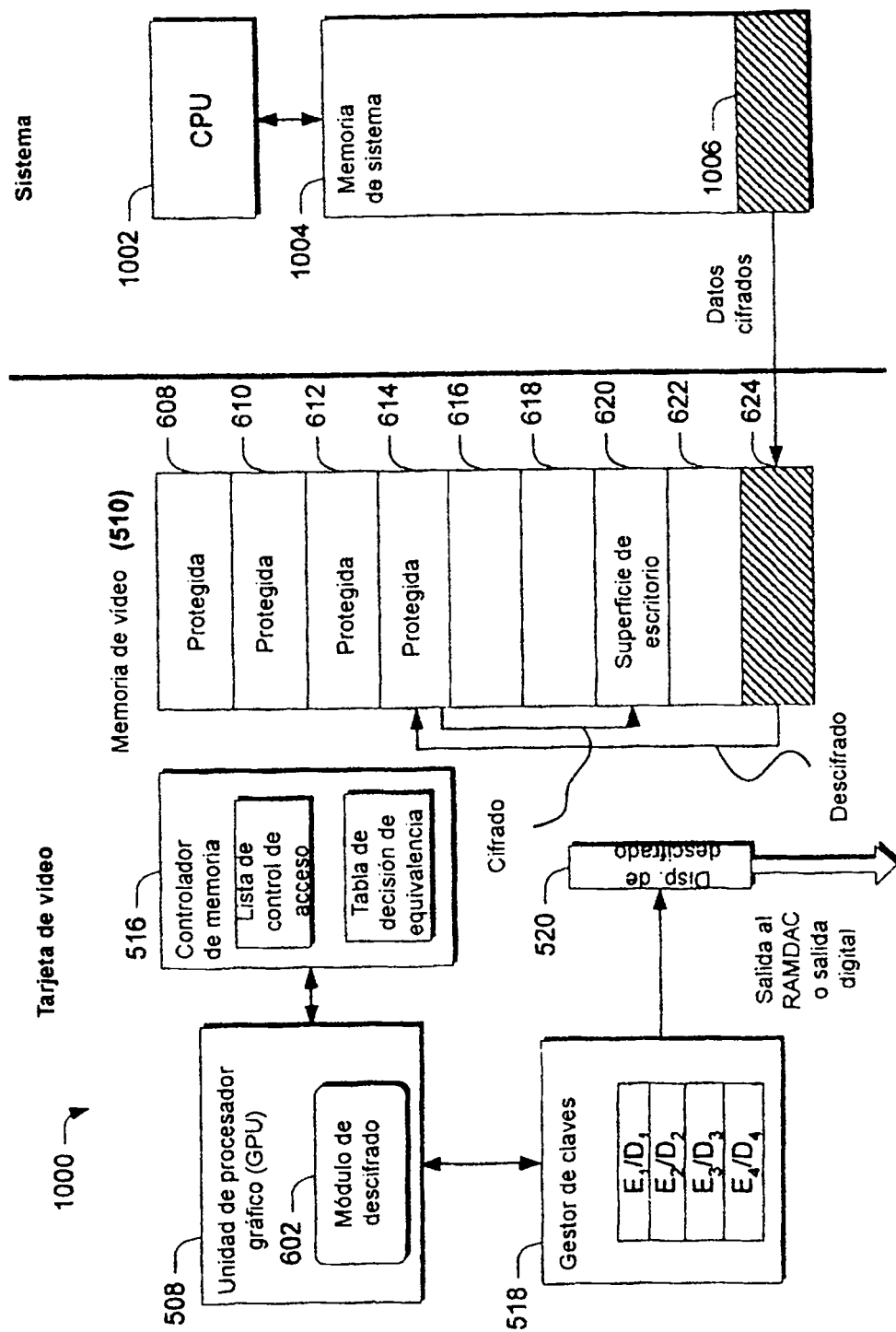


Fig. 10