

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 879 806**

51 Int. Cl.:

G06F 9/44

(2008.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **17.11.2015 PCT/US2015/060957**

87 Fecha y número de publicación internacional: **02.06.2016 WO16085695**

96 Fecha de presentación y número de la solicitud europea: **17.11.2015 E 15802292 (1)**

97 Fecha y número de publicación de la concesión europea: **09.06.2021 EP 3224709**

54 Título: **Cambio de contexto virtual acelerado por hardware**

30 Prioridad:

25.11.2014 US 201462084412 P
25.03.2015 US 201514667879

45 Fecha de publicación y mención en BOPI de la
traducción de la patente:
23.11.2021

73 Titular/es:

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

72 Inventor/es:

HEPKIN, DAVID ALAN

74 Agente/Representante:

LINAGE GONZÁLEZ, Rafael

ES 2 879 806 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

Cambio de contexto virtual acelerado por hardware

5 ANTECEDENTES

Antecedentes y técnica relevante

10 Los ordenadores y los sistemas informáticos han afectado a casi todos los aspectos de la vida moderna. Los ordenadores generalmente están involucradas en el trabajo, la recreación, la atención médica, el transporte, el entretenimiento, la gestión del hogar, etc.

15 Los sistemas informáticos se pueden implementar de una manera en la que las máquinas virtuales se implementan utilizando máquinas físicas. En particular, una máquina física puede albergar una o más, y normalmente varias, máquinas virtuales. Normalmente, se mantiene un contexto de máquina virtual (VMC) para cada procesador virtual o máquina virtual. El VMC proporciona el contexto del procesador para una máquina virtual invitada. Normalmente, se puede mantener un único VMC para cada máquina virtual, o cada procesador de máquina virtual, de forma individualizada.

20 El objeto reivindicado en el presente documento no se limita a las realizaciones que resuelven cualquier desventaja o que operan solo en entornos como los descritos anteriormente. Más bien, estos antecedentes solo se proporcionan para ilustrar un área de tecnología ejemplar en la que se pueden poner en práctica algunas realizaciones descritas en el presente documento.

25 El documento US 2014/281465 describe una unidad de procesamiento central con capacidades de arranque dual, que comprende: una memoria de instrucciones que comprende una primera y una segunda área de memoria que están configuradas para ser programables individualmente, donde la primera y segunda área de memoria se pueden asignar a una memoria activa desde la cual se ejecutan las instrucciones y un memoria inactiva, respectivamente; en la que el conjunto de instrucciones para la unidad central de procesamiento comprende una instrucción dedicada que permite realizar un intercambio desde un área de memoria activa a un área de memoria inactiva, en la que el intercambio se realiza ejecutando la instrucción dedicada en la memoria activa seguida de un flujo de programa cambiar instrucción en la memoria activa, después de lo cual la memoria inactiva se convierte en la nueva memoria activa y la memoria activa se convierte en la nueva memoria inactiva y la ejecución de instrucciones continúa en la nueva memoria activa.

35 En el documento WO 03/046712 describe un método para distribuir instrucciones a una pluralidad de unidades de ejecución de un procesador, comprendiendo el método: detectar una instrucción dedicada para configurar dicho procesador; y enviar otra instrucción a una unidad de ejecución de la pluralidad de unidades de ejecución, caracterizado por seleccionar la unidad de ejecución de la pluralidad de unidades de ejecución tras la detección de la instrucción dedicada.

BREVE SUMARIO

45 Una realización incluye un método que se puede practicar en un entorno informático virtual. El método incluye actos para cambiar entre contextos virtuales aislados. El método incluye determinar que un procesador virtual debe cambiar de un primer contexto de máquina virtual a un segundo contexto de máquina virtual. Los contextos de la primera y segunda máquina virtual están aislados entre sí. Un procesador físico recibe una instrucción incluida en la arquitectura del conjunto de instrucciones para el procesador físico que indica que el procesador virtual debe cambiar directamente del contexto de la primera máquina virtual al contexto de la segunda máquina virtual. Como resultado, el método incluye realizar el cambio de contexto virtual mediante la ejecución de funciones nativas del procesador físico definido de forma nativa para la instrucción.

55 Otra realización se implementa en un entorno informático virtual. La realización incluye un sistema configurado para cambiar entre contextos virtuales aislados. El sistema incluye un procesador físico. El procesador físico incluye una arquitectura de conjunto de instrucciones. La arquitectura del conjunto de instrucciones incluye una instrucción incluida en la arquitectura del conjunto de instrucciones para el procesador físico que, cuando se invoca, indica que un procesador virtual implementado utilizando el procesador físico debe cambiar directamente de un primer contexto de máquina virtual a un segundo contexto de máquina virtual. Los contextos de la primera y segunda máquina virtual están aislados entre sí.

60 Este sumario se proporciona para presentar una selección de conceptos en una forma simplificada que se describen con más detalle a continuación en la descripción detallada. Este sumario no pretende identificar características clave o características esenciales de la materia patentable reivindicada, ni pretende ser utilizado como ayuda para determinar el alcance de la materia patentable reivindicada.

65 Las características y ventajas adicionales se expondrán en la descripción que sigue, y en parte serán obvias a partir

de la descripción, o pueden aprenderse mediante la práctica de las enseñanzas en el presente documento. Las características y ventajas de la invención pueden realizarse y obtenerse por medio de los instrumentos y combinaciones señalados particularmente en las reivindicaciones adjuntas. Las características de la presente invención resultarán más evidentes a partir de la siguiente descripción y las reivindicaciones adjuntas, o pueden aprenderse mediante la práctica de la invención como se expone a continuación.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

Con el fin de describir la manera en la que se pueden obtener las ventajas y características mencionadas anteriormente y otras, se presentará una descripción más particular de la materia patentable descrito con brevedad anteriormente con referencia a realizaciones específicas que se ilustran en los dibujos adjuntos. Entendiendo que estos dibujos representan solo realizaciones típicas y, por lo tanto, no deben considerarse de alcance limitativo, las realizaciones se describirán y explicarán con especificidad y detalle adicionales mediante el uso de los dibujos adjuntos en los que:

La figura 1A ilustra un sistema de ejemplo generalizado que tiene un procesador con una arquitectura de conjunto de instrucciones que incluye una instrucción para el cambio del contexto de la máquina virtual;

La figura 1B ilustra un sistema de ejemplo con niveles jerárquicos de confianza virtual, donde el sistema tiene un procesador con una arquitectura de conjunto de instrucciones que incluye una instrucción para el cambio de contexto de la máquina virtual;

La figura 1C ilustra un sistema de ejemplo con contextos de máquinas virtuales hermanas que tienen un procesador con una arquitectura de conjunto de instrucciones que incluye una instrucción para el cambio de contexto de la máquina virtual; y

La figura 2 ilustra un método para cambiar entre el contexto de una máquina virtual aislada.

DESCRIPCIÓN DETALLADA

Como se señaló en el contexto anterior, normalmente hay un único contexto de máquina virtual (VMC) por procesador virtual en una máquina virtual. Entonces, si una sola máquina virtual tiene un solo procesador virtual, habrá un VMC para la máquina virtual (para el único procesador virtual). Si la máquina virtual tiene dos procesadores virtuales, habrá dos contextos de máquina virtual para la máquina virtual, uno para cada procesador virtual. Sin embargo, puede haber escenarios en los que un solo procesador virtual utilice varios VMC. La figura 1A ilustra un ejemplo generalizado en el que un procesador virtual 112 puede cambiar entre varios VMC 110-A, 110-B y 110-C.

Un ejemplo en el que un procesador virtual puede cambiar entre varios VMC es cuando se desea cambiar entre niveles de confianza virtuales. Un ejemplo de esto se ilustra en la figura 1B, donde el cambio jerárquica desde un VMC 110-J de alto nivel de confianza y hasta VMC de nivel de confianza más bajo 110-K, 110-L y 110-M. Un ejemplo del mundo real de esto es una tecnología conocida como Modo Seguro Virtual (VSM). Con VSM, cada procesador virtual en una plataforma de virtualización, como Hyper-V disponible en Microsoft Corporation de Redmond Washington, tiene múltiples niveles de confianza virtual (VTL). Cada nivel de confianza virtual proporciona un límite de seguridad diferente y la plataforma de virtualización proporciona la capacidad de que un procesador virtual cambie entre diferentes niveles de confianza virtual. Hay un contexto de CPU independiente asociado con cada VTL.

En la mayoría de las arquitecturas informáticas modernas que admiten la virtualización de procesadores, el concepto de VMC proporciona el contexto del procesador para una máquina virtual invitada. El procesador proporciona capacidades para que un hipervisor cargue el contexto de un procesador desde un VMC y guarde el contexto de un procesador en el contexto de una máquina virtual.

Para el VSM, el hipervisor mantiene un VMC separado para cada nivel de confianza virtual. En algunos procesadores, como algunos disponibles en Intel Corporation de Santa Clara, California, esto significa tener una estructura de contexto de máquina virtual (VMCS) separada para cada nivel de confianza virtual. El VSM puede implementarse como un sistema jerárquico donde diferentes VMC tienen diferentes privilegios definidos jerárquicamente.

Cuando el SO de una máquina virtual invitada que se ejecuta en un procesador virtual desea cambiar su VTL activo, realiza una hiperllamada al hipervisor de la máquina física para solicitar un cambio. El hipervisor activa el cambio cambiando el VMC activo y luego regresando de la hiperllamada. Esto requiere la ejecución de varias instrucciones de máquina diferentes para provocar el cambio.

Cabe señalar que, si bien el VSM es un ejemplo de una tecnología que aprovecha múltiples VMC para una sola máquina virtual, existen otras tecnologías que también podrían usar múltiples VMC para una sola máquina virtual. Por ejemplo, en la virtualización anidada, como se ilustra en la figura 1C, puede haber situaciones en las que se implementen varias máquinas virtuales en las máquinas virtuales principales. Por ejemplo, la figura 1C ilustra un VMC 110-W principal y varios VMC hermanos 110-X, 110-Y y 110-Z que se ejecutan dentro del VMC 110-W

principal. Las máquinas virtuales que se ejecutan en la máquina virtual principal pueden implementarse como diferentes VMC (por ejemplo, VMC 110-X, 110-Y y 110-Z) de la máquina virtual principal (por ejemplo, una máquina virtual que tiene el VMC 110-W). En tales situaciones, puede ser deseable cambiar entre los diferentes VMC. Este es simplemente un ejemplo y debe tenerse en cuenta que se podrían implementar otros escenarios de cambio de VMC.

Estos dos ejemplos ilustran dos arreglos de VMC. El primer ejemplo (es decir, el ejemplo de VSM) es un ejemplo de VMC jerárquicos multinivel donde cada uno de los diferentes VMC se encuentra en un nivel diferente de la jerarquía entre sí. Por el contrario, en el segundo ejemplo (es decir, virtualización anidada) la mayoría de las máquinas virtuales (es decir, las máquinas virtuales que se ejecutan en la máquina virtual principal) tienen VMC hermanos donde cada VMC tiene el mismo nivel de privilegios o uno muy similar con respecto a otros VMC hermanos.

Si bien el cambio de VMC anterior dependía de un hipervisor para administrar y realizar el cambio, lo que resultaba en una mayor sobrecarga del procesador para ejecutar las diversas funciones asociadas con las comunicaciones del hipervisor, la funcionalidad se describe aquí para una instrucción de procesador incluida en la arquitectura del conjunto de instrucciones de una clase de procesadores que cuando se invoca aceleraría un cambio de contexto virtual. Ahora se hace referencia a la figura 1A, que ilustra un sistema 102, que incluye un procesador físico 104 que tiene una arquitectura de conjunto de instrucciones 106, que incluye una instrucción. La instrucción proporcionaría la capacidad para que una máquina virtual invitada 108 inicie un cambio de VMC entre VMC 110 para un procesador virtual 112 sin generar una intercepción en el hipervisor 114 (es decir, permitir que una máquina virtual invitada cambie un VMC por sí misma sin pasar por una salida de máquina virtual (VM)).

Al invocar una instrucción de procesador incluida como parte de la arquitectura de conjunto de instrucciones 106 de un procesador físico 104, en lugar de permitir que el hipervisor 114 administre cambios VMC, los sistemas pueden aumentar el rendimiento y la velocidad del sistema informático utilizando una cantidad reducida de procesamiento general para ser se realiza eliminando el procesamiento de sobrecarga requerido cuando el hipervisor 114 realiza el cambio de VMC.

La energía se puede conservar en los dispositivos informáticos móviles (o cualquier dispositivo informático para el caso) al reducir el número total de operaciones del procesador que deben realizarse para lograr un cambio de VMC.

Para acelerar un cambio de VMC, se implementa una nueva instrucción en una arquitectura de conjunto de instrucciones 106 para una clase de procesadores (o clases de procesadores) que permite que una máquina virtual invitada 108 inicie un cambio a un VMC diferente invocando la instrucción. En el resto en el presente documento, por conveniencia, esta instrucción se denomina instrucción VMCSWITCH. Sin embargo, debe tenerse en cuenta que esto no pretende ser una limitación, sino simplemente a modo de ejemplo.

Esta instrucción VMCSWITCH hace que las operaciones de guardado de estado y las operaciones de restauración de estado se realicen utilizando la funcionalidad nativa del procesador físico 104 en lugar de las operaciones de guardado de estado que ocurren en una salida de VM a un hipervisor por dirección del hipervisor y las operaciones de restauración de estado que ocurren en una entrada de VM en la máquina virtual invitada por dirección del hipervisor.

Por ejemplo, si una máquina virtual invitada 108 se estaba ejecutando con un VMC 110-A activo e inicia un VMCSWITCH para cambiar a VMC 110-B, ocurriría lo siguiente:

1. El estado actual de la máquina virtual invitada 108 se guarda en VMC 110-A mediante la funcionalidad del procesador físico nativo incluida en la arquitectura del conjunto de instrucciones 106 para la instrucción VMCSWITCH. El guardado de estado puede incluir operaciones como guardar registros de estado de la máquina (MSR) en un área de almacenamiento de MSR. Se pueden implementar varias alternativas diferentes para un área de almacenamiento de MSR. Por ejemplo, los MSR pueden simplemente almacenarse en la memoria del sistema 118. Alternativamente, los MSR pueden almacenarse en la memoria incorporada 122 para un acceso más rápido. Además, el procesador físico 104 puede incluir un almacenamiento de MSR 124 especial configurado especialmente para mantener el estado de MSR y para ser utilizado como registros de estado de modo que no se necesite una operación de guardado separada. De esta forma, cuando se cambia un VMC, puede que no sea necesario volver a cargar el estado de MSR en los MSR, sino que la porción apropiada del almacenamiento de MSR 124 podría usarse como MSR. Por ejemplo, la porción de almacenamiento de MSR 124-A puede ser esencialmente los MSR para VMC 110-A, la porción de almacenamiento de MSR 124-B puede ser esencialmente los MSR para VMC 110-B y la porción de almacenamiento de MSR 124-C puede ser esencialmente los MSR para VMC 110-C.

2. El nuevo estado de máquina virtual invitada se carga desde VMC 110-B en el procesador virtual 112 utilizando la funcionalidad del procesador físico nativo 104 incluida en la arquitectura del conjunto de instrucciones 106 para la instrucción VMCSWITCH. Esto puede incluir operaciones como restaurar los MSR desde un área de almacenamiento de MSR, restaurar el puntero de instrucción de la máquina virtual invitada, etc. En el ejemplo donde se usa un almacenamiento de MSR 124, las realizaciones pueden simplemente apuntar a la porción de almacenamiento de MSR apropiada para activar el estado apropiado.

Si hay un problema al cargar el estado desde VMC 110-B, se puede generar una intercepción en el hipervisor 114. De esta manera, el hipervisor puede integrarse en el cambio de VMC según sea necesario, pero excluido de la gestión de las operaciones de cambio de VMC de bajo nivel cuando dicha gestión puede ser realizada por el procesador físico utilizando la funcionalidad asociada con la instrucción VMCSWITCH especializada.

En algunas realizaciones, el procesador virtual 112 proporciona la capacidad para que el hipervisor 114 controle si una máquina virtual invitada 108 puede utilizar la instrucción VMCSWITCH. Por ejemplo, los VMC del procesador virtual podrían ampliarse para incluir un bit de control que indique si una máquina virtual invitada 108 puede utilizar la nueva instrucción VMCSWITCH. Si el bit de control está inhabilitado, un intento por parte de una máquina virtual invitada 108 de utilizar esta nueva instrucción, en algunas realizaciones, dará como resultado que el hipervisor se involucre más en el cambio. Por ejemplo, en algunas realizaciones, intentar utilizar la instrucción VMCSWITCH cuando el bit de control está inhabilitado provocará una excepción o generará una intercepción en el hipervisor 114. De esta manera, el hipervisor 114 puede controlar la cantidad de interacción que el hipervisor 114 tiene en el cambio de VMC permitiendo que una máquina virtual cambie de contexto automático, o evitando que la máquina virtual cambie de contexto automático, y en su lugar haciendo que el hipervisor 114 realice el cambio de contexto.

Además de poder controlar si una máquina virtual invitada puede utilizar la nueva instrucción VMCSWITCH, el procesador virtual también puede proporcionar la capacidad de permitir que el hipervisor 114 controle los VMC específicos a los que una máquina virtual invitada 108 puede cambiar. En una realización de ejemplo, esto puede lograrse mediante el hipervisor 114 manteniendo una lista de VMC 116 válidos a los que puede cambiar una máquina virtual invitada 108. Esto podría mantenerse, por ejemplo, de la siguiente manera:

- La lista 116 de VMC de "destino" válidos para un cambio se denomina "lista de VMC". En el ejemplo ilustrado, esto se organiza en una página de memoria 118. El hipervisor 114 proporciona una página con una lista de punteros VMC, y la máquina virtual invitada 108 selecciona un VMC especificando un índice en esta lista 116 como parámetro para la instrucción VMCSWITCH. El hipervisor 114 proporciona un puntero a esta lista a través de un campo en el VMC. Las realizaciones pueden incluir funcionalidad para indicar si una entrada particular en la lista 116 no es válida (por ejemplo, un valor de 0 en la lista podría significar que el puntero VMC no es válido). Los punteros de VMC pueden ser, por ejemplo, direcciones físicas. Si bien el ejemplo ilustra muestra la lista implementada en la memoria principal 118, en otras realizaciones, el propio procesador 104 puede tener una estructura de almacenamiento de lista 128 implementada en hardware, tal como registros de hardware especiales, en el procesador 104.
- Un operando de la instrucción VMCSWITCH especifica un VMC en la lista de VMC 116. En particular, el operando especifica un índice en la lista.

Con esta nueva capacidad VMCSWITCH, una máquina virtual invitada 108 podría emitir una instrucción VMCSWITCH para cambiar su VMC activo. Al emitir la instrucción VMCSWITCH, la máquina virtual invitada 108 especifica una entrada en la lista de VMC 116 (o estructura de almacenamiento de lista 128) a través de un registro de propósito general 120. Este registro de propósito general 120 contiene un índice en la lista de VMC 116 que indica el VMC de destino al que la máquina virtual invitada 108 le gustaría cambiar.

El procesador virtual 112 valida la entrada del puntero VMC en la lista 116 identificada por el índice de destino. Si el puntero de VMC no es válido, VMCSWITCH genera una intercepción en el hipervisor 114 con un nuevo código de salida para indicar el motivo de la intercepción.

Si la entrada del puntero de VMC es válida, el procesador físico 104 realiza el cambio de VMC como se describe anteriormente usando la funcionalidad del procesador nativo incluida como parte de la funcionalidad de la instrucción VMCSWITCH. Una vez completado el cambio de VMC, un nuevo VMC está activo y, por lo tanto, una nueva lista de punteros de VMC podría activarse como posibles destinos para la instrucción VMCSWITCH. El procesador virtual 112 se está ejecutando en la ubicación del puntero de instrucción especificada en el VMC recién activo.

En el ejemplo ilustrado anteriormente, los VMC pueden almacenarse en la memoria 118. Esta memoria 118 podría ser una memoria principal acoplada a un procesador físico. Sin embargo, en algunas realizaciones, los VMC podrían almacenarse en la memoria incorporada 122 (como la memoria caché) incluida en la misma matriz semiconductora con el procesador físico 104 o acoplada al procesador físico 104 de una manera que permita un acceso más rápido a los datos memoria principal. De esta manera, el cambio VCS podría ser más rápida y eficiente. Como se discutió anteriormente, en otras realizaciones alternativas, los VMC pueden almacenarse almacenando el estado de MSR en el almacenamiento de MSR 124.

La siguiente discusión ahora se refiere a una serie de métodos y actos de método que se pueden realizar. Aunque los actos del método pueden discutirse en un cierto orden o ilustrarse en un diagrama de flujo como ocurriendo en un orden particular, no se requiere un orden en particular a menos que se indique específicamente, o se requiera porque un acto depende de que otro acto se complete antes de que el acto que se realiza.

Con referencia ahora a la figura 2, se ilustra un método 200. El método 200 se puede practicar en un entorno informático virtual donde las máquinas virtuales se implementan en hardware físico. El método incluye actos para cambiar entre contextos de máquinas virtuales aisladas. El método incluye determinar que un procesador virtual debe cambiar de un primer contexto de máquina virtual a un segundo contexto de máquina virtual (acto 202). Los contextos de la primera y segunda máquina virtual están aislados entre sí. El aislamiento entre sí significa que los contextos de las máquinas virtuales no tienen acceso de lectura o escritura a la información de contexto de los demás.

El método incluye además un procesador físico que recibe una instrucción incluida en la arquitectura del conjunto de instrucciones para el procesador físico que indica que el procesador virtual debe cambiar directamente (es decir, sin que el hipervisor realice la función de cambio como resultado de una hiperllamada) desde el primer contexto de la máquina virtual a el segundo contexto de máquina virtual (acto 204).

Como resultado, el método 200 incluye además realizar el cambio ejecutando funciones nativas del procesador físico definidas de manera nativa para la instrucción (acto 206).

El método 200 se puede practicar donde el procesador físico recibe la instrucción del primer contexto virtual. Por ejemplo, el VMC 110-A puede llamar a una instrucción en la arquitectura del conjunto de instrucciones 106 para cambiar al VMC 110-B.

El método 200 se puede practicar donde el primer y segundo contextos se encuentran en diferentes niveles de privilegios jerárquicos entre sí. Un ejemplo de esto se ilustra arriba en el ejemplo de VSM donde los cambios se realizan a diferentes niveles de confianza virtual.

El método 200 puede practicarse donde el primer y segundo contextos son hermanos que tienen el mismo nivel de privilegios entre sí. Un ejemplo de esto se ilustra arriba en el caso de virtualización anidada donde las máquinas virtuales se ejecutan dentro de otras máquinas virtuales.

El método 200 puede incluir además determinar que el cambio del primer contexto al segundo contexto está incluido en una enumeración de los cambios de contexto permitidos. Esto se ilustra arriba en el contexto de la lista de VMC. En algunas realizaciones, la lista puede implementarse usando un índice de manera que los detalles específicos de dónde se oculta la información del contexto de la máquina virtual real de los diversos contextos de la máquina virtual. En algunas realizaciones, el hipervisor controla la información en el índice de la lista.

El método 200 puede incluir además determinar si el hardware permite el cambio de VMC nativo y, cuando no está permitido, realizar el cambio mediante el control del hipervisor del cambio de VMC. Se puede realizar una determinación, por ejemplo, determinando una versión de hardware. La versión de hardware puede tener arquitecturas de conjunto de instrucciones con soporte de cambio de VMC de hardware o puede que no. Por lo tanto, en algunos ejemplos, la versión de hardware se utilizará para determinar si se admite el cambio de VMC de hardware. En otros ejemplos, se puede realizar una determinación basándose en la información de capacidad del hardware almacenada en una descripción de firmware del hardware.

Sin embargo, en algunas realizaciones alternativas o adicionales, la funcionalidad de cambio de VMC soportada por hardware puede determinarse verificando que el cambio de VMC por hardware se ha habilitado en hardware que soporta el cambio de VMC en la arquitectura del conjunto de instrucciones para el hardware. Por tanto, aunque el hardware pueda soportar el cambio de VMC basada en hardware, esto puede habilitarse o deshabilitarse en algunas realizaciones. Cuando está deshabilitado, el cambio de VMC atrapa al hipervisor. En algunas realizaciones, el hipervisor puede habilitar o deshabilitar el cambio de VMC basado en hardware. Es posible que el hipervisor desee hacer esto en los casos en que el hipervisor desee forzar el cambio de VMC controlado por el hipervisor. Esto puede hacerse, por ejemplo, cuando un primer VMC solicita un cambio de VMC para obtener algunos datos o funcionalidad de un VMC de nivel superior y necesita datos devueltos desde el VMC de nivel superior al primer VMC. El hipervisor puede realizar el cambio de VMC y devolver los datos apropiados u otra información al primer VMC.

El método 200 puede implementarse donde el cambio es transparente para un VMC. Esto se puede hacer de tal manera que un VMC no pueda determinar de forma independiente si el cambio está controlada por el hipervisor o una instrucción de hardware.

Además, los métodos pueden ser practicados por un sistema informático que incluya uno o más procesadores y medios legibles por ordenador tales como memoria de ordenador. En particular, la memoria del ordenador puede almacenar instrucciones ejecutables por ordenador que cuando son ejecutadas por uno o más procesadores hacen que se realicen varias funciones, tales como los actos enumerados en las realizaciones.

Las realizaciones de la presente invención pueden comprender o utilizar un ordenador de propósito especial o de propósito general que incluye hardware de ordenador, como se describe con mayor detalle a continuación. Las realizaciones dentro del alcance de la presente invención también incluyen medios físicos y otros medios legibles por ordenador para transportar o almacenar instrucciones y/o estructuras de datos ejecutables por ordenador.

Dichos medios legibles por ordenador pueden ser cualquier medio disponible al que se pueda acceder mediante un sistema informático de propósito general o de propósito especial. Los medios legibles por ordenador que almacenan instrucciones ejecutables por ordenador son medios de almacenamiento físico. Los medios legibles por ordenador que llevan instrucciones ejecutables por ordenador son medios de transmisión. Así, a modo de ejemplo, y no de limitación, las realizaciones de la invención pueden comprender al menos dos tipos claramente diferentes de medios legibles por ordenador: medios de almacenamiento físicos legibles por ordenador y medios de transmisión legibles por ordenador.

Los medios de almacenamiento físicos legibles por ordenador incluyen RAM, ROM, EEPROM, CD-ROM u otro almacenamiento en disco óptico (como CD, DVD, etc.), almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético, o cualquier otro medio que pueda usarse para almacenar medios de código de programa deseados en forma de instrucciones o estructuras de datos ejecutables por ordenador y a los que se puede acceder mediante un ordenador de propósito general o de propósito especial.

Una "red" se define como uno o más enlaces de datos que permiten el transporte de datos electrónicos entre sistemas y/o módulos informáticos y/u otros dispositivos electrónicos. Cuando la información se transfiere o se proporciona a través de una red u otra conexión de comunicaciones (ya sea cableada, inalámbrica o una combinación de cableada o inalámbrica) a un ordenador, el ordenador ve correctamente la conexión como un medio de transmisión. Los medios de transmisión pueden incluir una red y/o enlaces de datos que se pueden usar para transportar o medios de código de programa deseados en forma de instrucciones o estructuras de datos ejecutables por ordenador y a las que se puede acceder mediante un ordenador de propósito general o de propósito especial. Las combinaciones de los anteriores también se incluyen dentro del alcance de los medios legibles por ordenador.

Además, al llegar a varios componentes del sistema informático, los medios de código de programa en forma de instrucciones ejecutables por ordenador o estructuras de datos pueden transferirse automáticamente desde medios de transmisión legibles por ordenador a medios de almacenamiento físicos legibles por ordenador (o viceversa). Por ejemplo, las instrucciones ejecutables por ordenador o las estructuras de datos recibidas a través de una red o un enlace de datos pueden almacenarse en la memoria RAM dentro de un módulo de interfaz de red (por ejemplo, un "NIC") y luego transferirse finalmente a la RAM del sistema informático y/o a una memoria menos volátil medios de almacenamiento físico legibles por ordenador en un sistema informático. Por tanto, los medios de almacenamiento físico legibles por ordenador pueden incluirse en los componentes del sistema informático que también (o incluso principalmente) utilizan medios de transmisión.

Las instrucciones ejecutables por ordenador comprenden, por ejemplo, instrucciones y datos que hacen que un ordenador de propósito general, un ordenador de propósito especial o un dispositivo de procesamiento de propósito especial realice una determinada función o grupo de funciones. Las instrucciones ejecutables por ordenador pueden ser, por ejemplo, binarios, instrucciones de formato intermedio como lenguaje ensamblador o incluso código fuente. Aunque la materia patentable se ha descrito en un lenguaje específico para características estructurales y/o actos metodológicos, debe entenderse que la materia patentable definida en las reivindicaciones adjuntas no está necesariamente limitada a las características o actos descritos anteriormente. Más bien, las características y actos descritos se describen como formas de ejemplo de implementación de las reivindicaciones.

Los expertos en la técnica apreciarán que la invención se puede poner en práctica en entornos informáticos de red con muchos tipos de configuraciones de sistemas informáticos, que incluyen ordenadores personales, ordenadores de escritorio, ordenadores portátiles, procesadores de mensajes, dispositivos portátiles, sistemas multiprocesador, microprocesadores, electrónica de consumo programable o basada en la red, PC de red, miniordenadores, ordenadores centrales, teléfonos móviles, PDA, buscapersonas, enrutadores, cambios y similares. La invención también se puede practicar en entornos de sistemas distribuidos donde los sistemas informáticos locales y remotos, que están vinculados (ya sea mediante enlaces de datos cableados, enlaces de datos inalámbricos o mediante una combinación de enlaces de datos cableados e inalámbricos) a través de una red, ambos realizan tareas. En un entorno de sistema distribuido, los módulos de programa pueden estar ubicados en dispositivos de almacenamiento de memoria tanto locales como remotos.

Alternativamente, o, además, la funcionalidad descrita en el presente documento se puede realizar, al menos en parte, mediante uno o más componentes lógicos de hardware. Por ejemplo, y sin limitación, los tipos ilustrativos de componentes lógicos de hardware que se pueden utilizar incluyen matrices de puertas programables en campo (FPGA), circuitos integrados específicos del programa (ASIC), productos estándar específicos del programa (ASSP), sistema en "sistemas en un chip" (SOC), dispositivos lógicos programables complejos (CPLD), etc.

REIVINDICACIONES

1. Un método implementado por ordenador (200) de cambio entre contextos virtuales aislados, siendo realizado el método implementado por ordenador por uno o más procesadores que ejecutan instrucciones ejecutables por ordenador para el método implementado por ordenador, y el método implementado por ordenador comprende:
5 determinar (202) que un procesador virtual debe cambiar de un primer contexto de máquina virtual a un segundo contexto de máquina virtual iniciado por una máquina virtual invitada, en el que los contextos de la primera y segunda máquina virtual están aislados entre sí;
10 un procesador físico que recibe (204) una instrucción incluida en una arquitectura de conjunto de instrucciones para el procesador físico que indica que el procesador virtual debe cambiar directamente por el procesador físico sin generar una intercepción en un hipervisor desde el contexto de la primera máquina virtual al contexto de la segunda máquina virtual; y
15 como resultado, realizar (206) el cambio de contexto virtual ejecutando funciones nativas del procesador físico definidas nativamente para la instrucción;

 en el que los contextos de la primera y segunda máquina virtual comprenden diferentes niveles de privilegios jerárquicos entre sí.
20 2. El método implementado por ordenador de la reivindicación 1, en el que el procesador físico recibe la instrucción del primer contexto de máquina virtual.

 3. El método implementado por ordenador de la reivindicación 1, que comprende además determinar que el cambio del contexto de la primera máquina virtual al contexto de la segunda máquina virtual está incluido en una
25 enumeración de los cambios de contexto virtual permitidos.

 4. El método implementado por ordenador de la reivindicación 1, que comprende además determinar si el hardware permite el cambio de contexto de la máquina virtual y, cuando no está permitido, realizar el cambio de contexto de la máquina virtual mediante el control del hipervisor del cambio de contexto de la máquina virtual.
30 5. El método implementado por ordenador de la reivindicación 4, en el que el cambio de contexto de máquina virtual es transparente para el contexto de una máquina virtual, de modo que el contexto de la máquina virtual no puede determinar de forma independiente si el cambio de contexto de la máquina virtual está controlada por el hipervisor o una instrucción de hardware de contexto de máquina virtual incluida en la arquitectura de conjunto de instrucciones de un procesador.
35 6. Un sistema informático (102) configurado para cambiar entre contextos virtuales aislados, comprendiendo el sistema informático:
40 un medio legible por ordenador que contiene instrucciones ejecutables por ordenador;
 uno o más procesadores (104) que, al ejecutar las instrucciones ejecutables por ordenador, hacen que el sistema informático se configure con una arquitectura para realizar un método implementado por ordenador para cambiar entre contextos virtuales aislados, y cuando se configura con la arquitectura, el sistema informático que realiza lo
45 siguiente:
 determinar que un procesador virtual debe cambiar de un primer contexto de máquina virtual a un segundo contexto de máquina virtual iniciado por una máquina virtual invitada, en el que los contextos de la primera y segunda máquina virtual están aislados entre sí;
50 un procesador físico que recibe una instrucción incluida en una arquitectura de conjunto de instrucciones (106) para el procesador físico (104) que indica que el procesador virtual debe cambiar directamente por el procesador físico sin generar una intercepción en un hipervisor desde el primer contexto de la máquina virtual al segundo contexto de la máquina virtual; y
 como resultado, realizar el cambio de contexto virtual ejecutando funciones nativas del procesador físico (104) definido de manera nativa para la instrucción;
55 en el que los contextos de la primera y segunda máquina virtual comprenden diferentes niveles de privilegios jerárquicos entre sí.

 7. El sistema informático de la reivindicación 6, que comprende, además:
60 un bit de control que se puede utilizar para indicar cuándo se puede invocar la instrucción; y
 un hipervisor (114) configurado para realizar el cambio de contexto virtual cuando el bit de control indica que la instrucción no puede invocarse.

 8. El sistema informático de la reivindicación 6, que comprende además un almacenamiento de datos configurado para mantener una lista de contextos de máquina virtual válidos a los que puede cambiar el procesador físico, y en
65 8

el que el almacenamiento de datos está configurado para mantener la lista como una lista indexada, de modo que la lista indexa las ubicaciones de la memoria donde se almacenan los contextos de la máquina virtual y la entrada a la lista se realiza mediante un índice tal que una máquina virtual que llama a la lista no tiene acceso a la información que identifica las ubicaciones de la memoria, sino que solo puede especificar un puntero a una entrada en la lista.

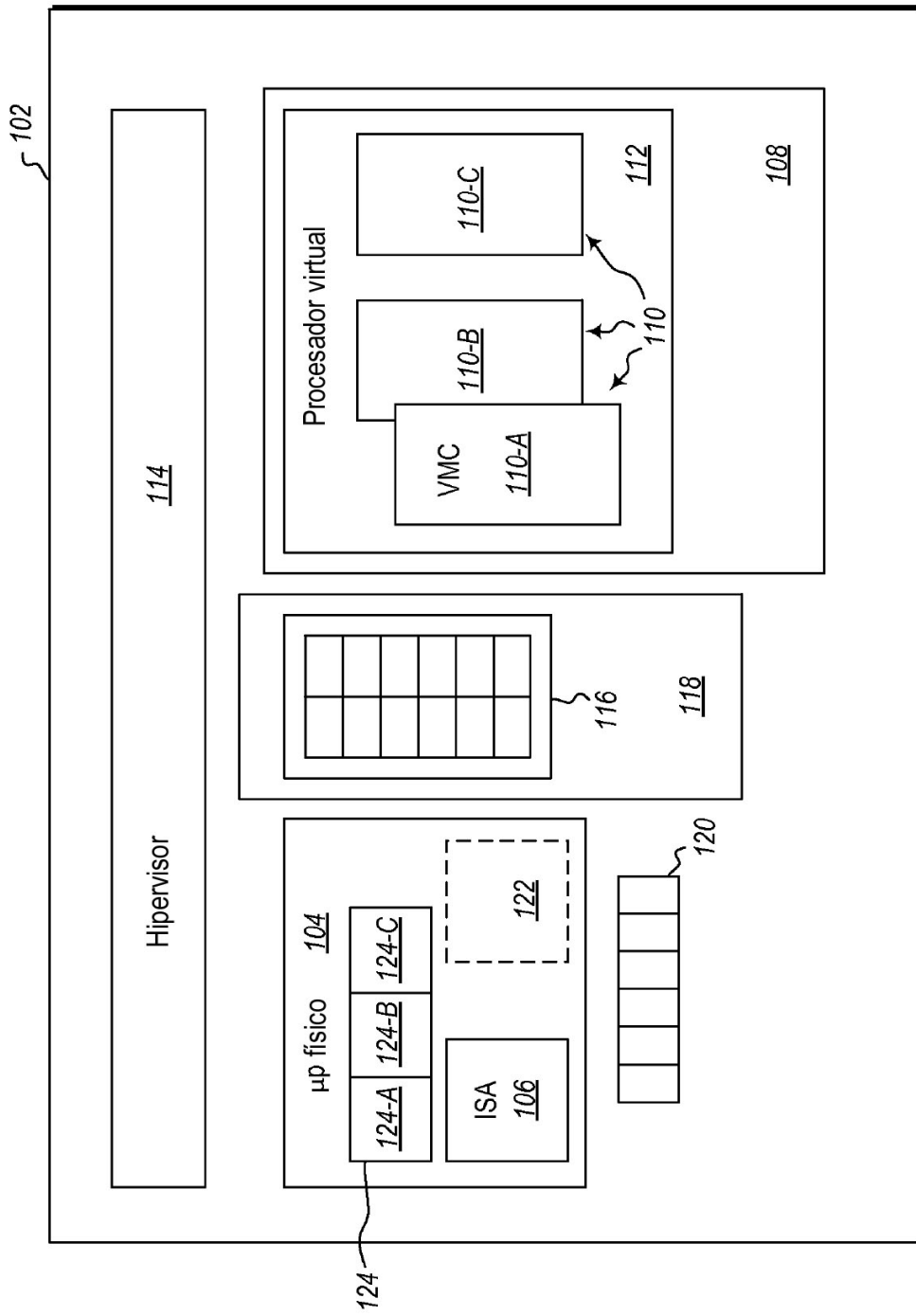


Figura 1A

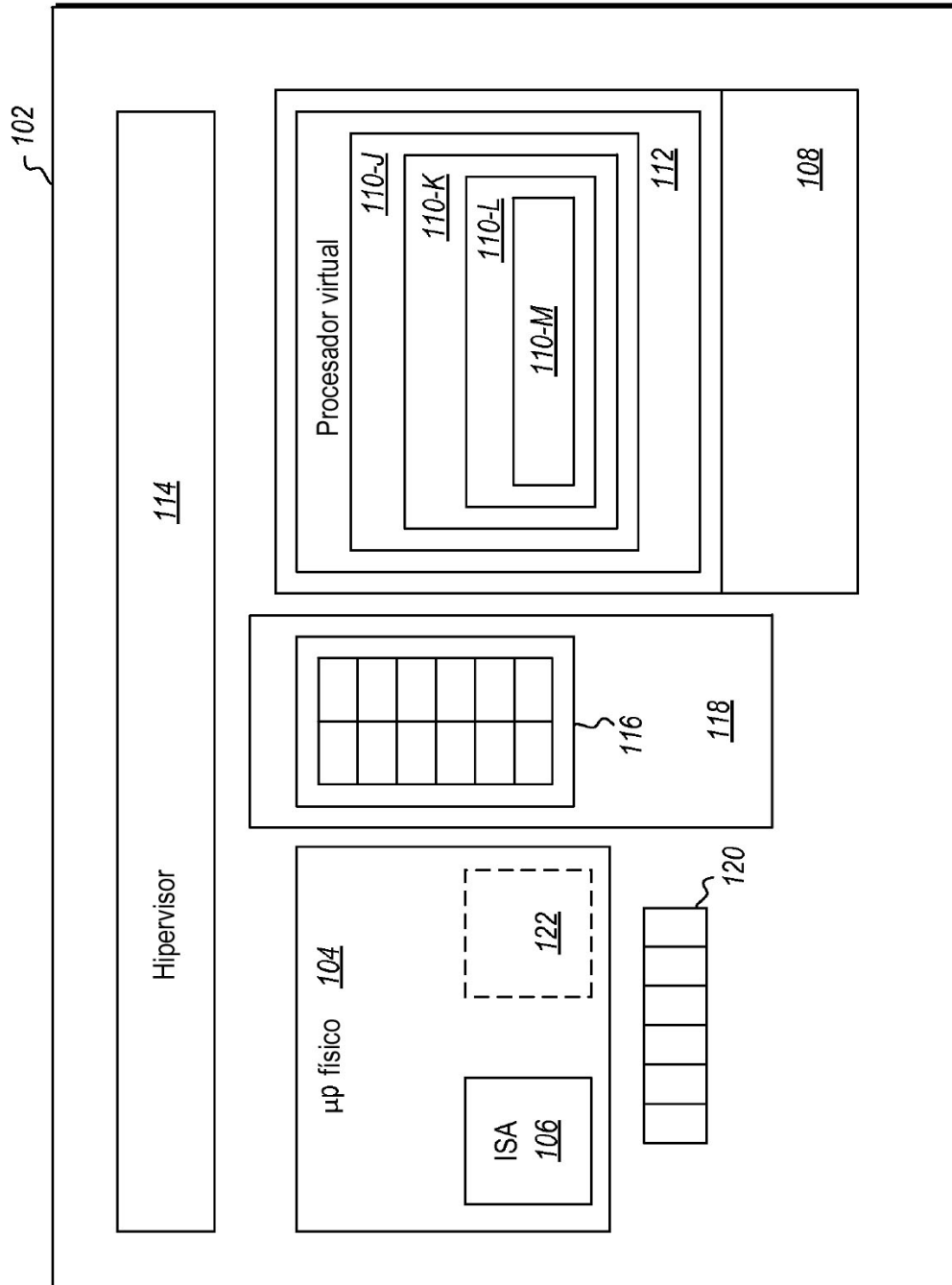


Figura 1B

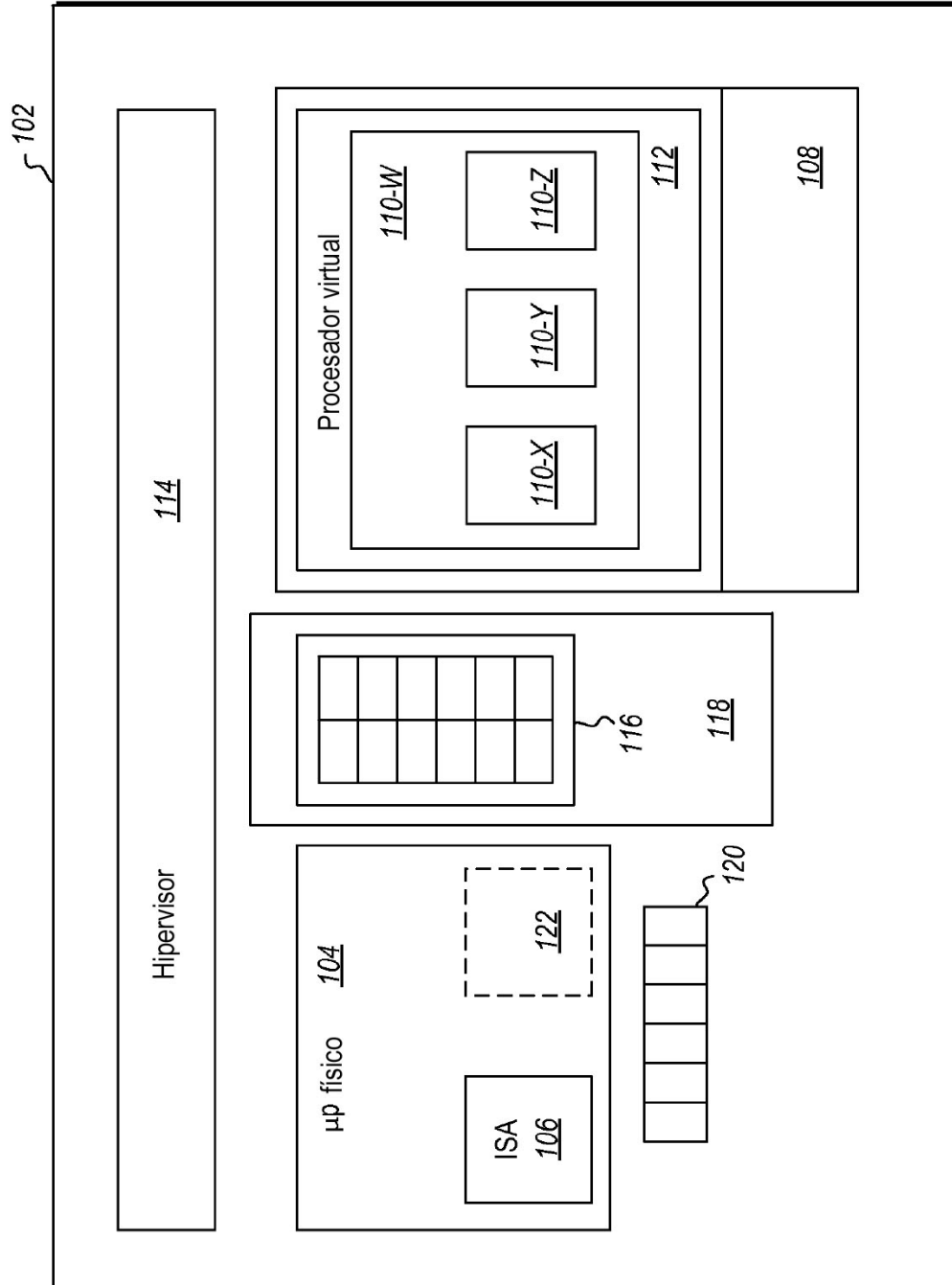


Figura 1C

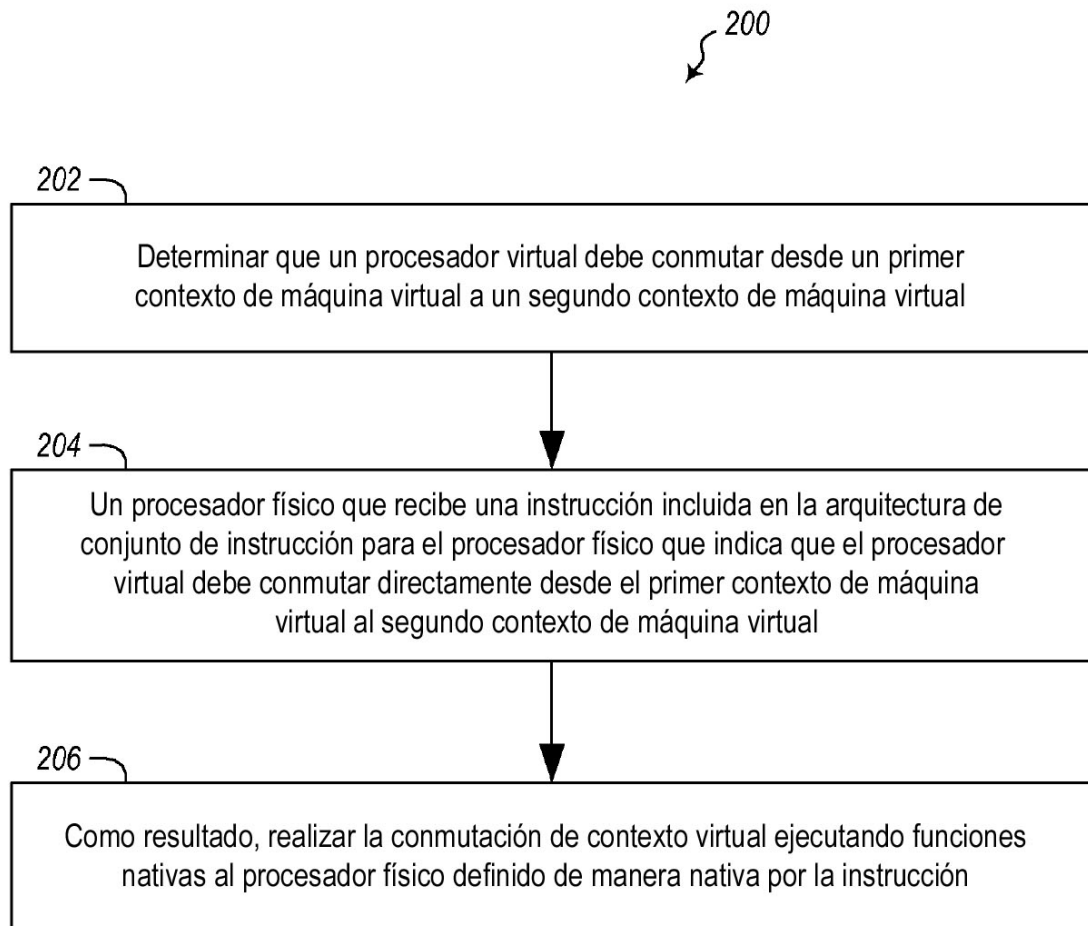


Figura 2