República Federativa do Brasil
Ministério da Economia
Instituto Nacional da Propriedade Industrial

(11) BR 112015022865-8 B1

(22) Data do Depósito: 10/03/2014

(45) Data de Concessão: 22/03/2022

(54) Título: MÉTODO E APARELHO PARA ATIVAR SELETIVAMENTE AS OPERAÇÕES DE UM MONITOR DE MÁQUINA VIRTUAL SOB DEMANDA

(51) Int.Cl.: G06F 9/455; G06F 21/53.

(30) Prioridade Unionista: 12/03/2013 US 13/796,442.

(73) Titular(es): QUALCOMM INCORPORATED.

(72) Inventor(es): THOMAS ZENG; AZZEDINE TOUZNI; PHILIP JR. MUELLER; PIYUSH PATEL.

(86) Pedido PCT: PCT US2014022731 de 10/03/2014

(87) Publicação PCT: WO 2014/164536 de 09/10/2014

(85) Data do Início da Fase Nacional: 11/09/2015

(57) Resumo: MÉTODO E APARELHO PARA ATIVAR SELETIVAMENTE AS OPERAÇÕES DE UM MONITOR DE MÁQUINA VIRTUAL SOB DEMANDA Nos vários aspectos, técnicas de virtualização podem ser utilizadas para aperfeiçoar o desempenho e reduzir a quantidade de energia consumida pela ativação seletiva de um hipervisor operando em um dispositivo de computação durante as sessões de sandbox. Nos vários aspectos, um sistema operacional de alto nível pode alocar memória de modo que seus endereços físicos intermediários sejam iguais aos endereços físicos. Quando o supervisor é desativado, o hipervisor pode suspender das traduções de segundo estágio dos endereços físicos intermediários para endereços físicos. Durante uma sessão de sandbox, o hipervisor pode ser ativado e retomar a realização das traduções de segundo estágio.

"MÉTODO E APARELHO PARA ATIVAR SELETIVAMENTE AS OPERAÇÕES DE UM MONITOR DE MÁQUINA VIRTUAL SOB DEMANDA"

Fundamentos

[0001] Geralmente, a tecnologia de virtualização permite a abstração (ou virtualização) dos recursos de computação pela colocação de um programa de controle de software (por exemplo, um Monitor de Máquina Virtual "VMM" ou hipervisor) entre o sistema operacional e o hardware. O hipervisor executa no modo privilegiado e pode hospedar múltiplos sistemas operacionais (chamados de sistemas operacionais convidados). Cada sistema operacional convidado comunica com o hipervisor da mesma forma que se comunicaria com o hardware físico, visualizando a combinação de hipervisor e hardware como uma única máquina virtual. Isso permite que cada sistema operacional convidado opere sob a ilusão de ter acesso exclusivo aos processadores, periféricos, memória e I/O.

[0002] Os sistemas operacionais são responsáveis pela divisão da memória física através de múltiplos processos. Nos sistemas que incluem um sistema operacional convidado rodando em cima de uma máquina virtual, a memória alocada pelo sistema operacional convidado não é a memória física verdadeira, mas uma memória física intermediária. Em tais sistemas, o hipervisor é responsável pela alocação real da memória física.

[0003] A maior parte dos processadores suporta apenas um estágio da translação do espaço de endereço de memória, e o hipervisor gerencia a relação entre os endereços virtuais (VA), os endereços físicos intermediários (IPA), e os endereços físicos (PA). Isso é geralmente alcançado pelo hipervisor mantendo suas próprias tabelas de translação (chamadas tabelas de translação de sombra), que são derivadas pela interpretação de cada uma

das tabelas de translação do sistema operacional convidado. Especificamente, o hipervisor garante que todas as mudanças nas tabelas de translação do sistema operacional convidado sejam refletidas nas estruturas de sombra, além de reforçar a proteção e redirecionar as falhas de acesso ao estágio adequado.

[0004] Diferentemente dos processadores de estágio único discutidos acima, os sistemas processadores ARM fornecem assistência de hardware para ambos os estágios da translação de memória (por exemplo, através das Extensões de Virtualização ARM tal como a Unidade de Gerenciamento de Memória de Sistema "SMMU"). Por exemplo, os processadores ARM incluem Extensões de Virtualização que permitem uma translação de dois estágios na qual os endereços virtuais (VA) são transladados para endereços físicos intermediários (IPA) no primeiro estágio (isso é, uma translação de primeiro estágio) e os endereços físicos intermediários (IPA) são transladados para os endereços físicos no segundo estágio (isso é, uma translação de segundo estágio). Isso reduz os overheads associados com o hipervisor.

Sumário

[0005] Os vários aspectos incluem dispositivo de computação e métodos para implementação seletiva de um hipervisor para reforçar de forma eficiente o controle de acesso para proteger os dados e/ou software quando necessário. Em vários aspectos, o hipervisor é normalmente desativado e é ativado quando uma condição é detectada que exige que o hipervisor implemente o controle de acesso (isso é, "sessão de caixa de areia"). Um sistema operacional de alto nível (HLOS) pode operar em uma máquina virtual gerenciada pelo hipervisor. HLOS pode manter uma tabela de página de endereço físico intermediário para uso

na alocação de endereços virtuais para vários processos ou aplicativos rodando no HLOS. O HLOS pode alocar memória diretamente do espaço de endereço de memória física, garantindo, assim, que os endereços de memória física intermediários sejam sempre iguais aos endereços de memória física. Garantindo-se que HLOS possa alocar memória de modo que os endereços físicos intermediários sejam iguais aos endereços físicos durante todo o tempo, o hipervisor pode ser seletivamente ativado quando existe a necessidade de uma sessão sandboxed, e desativado quando não houver uma sessão sandboxed atual. Enquanto está desativado, o hipervisor pode não realizar as translações de Estágio 2 a partir dos endereços físicos intermediários para os endereços físicos. Além disso, enquanto o hipervisor está desativado, HLOS pode alocar a memória a partir de todo o espaço de endereço de memória física.

[0006] Nos vários aspectos, o hipervisor pode ser ativado pela duração de uma sessão sandbox. Enquanto ativado, o hipervisor pode retomar a realização das translações de Estágio 2 a partir dos endereços físicos intermediários para os endereços físicos. Além disso, enquanto ativado, o hipervisor pode restringir o acesso do HLOS ao espaço de endereço de memória física, permitindo, assim, que o HLOS aloque memória a partir de apenas uma parte do espaço de endereço de memória física, e em alguns aspectos, restringe o acesso do HLOS a interrupções de hardware e/ou temporizadores de hardware. Pela configuração do hipervisor de modo a não realizar as translações de Estágio 2, entre outras coisas, enquanto sandboxing não é necessário (isso é, enquanto o hipervisor está desativado), os vários aspectos podem aperfeiçoar o desempenho geral do dispositivo de computação enquanto fornecem a segurança necessária como adequado.

[0007] Os vários aspectos incluem um método de gerenciamento de memória em um dispositivo de computação pela inicialização de um hipervisor, um monitor de segurança, e um sistema operacional de alto nível (HLOS); desativando o hipervisor depois da inicialização; monitorando um sinal a partir do monitor de segurança para iniciar uma sessão de sandbox; ativando o hipervisor quando o sinal for recebido para iniciar a sessão de sandbox; e implementando o controle de acesso enquanto o hipervisor é ativado. Em um aspecto, o monitor de segurança pode ser um ARM TrustZone®. Em outro aspecto, o hipervisor pode ser desativado ou ativado através de pelo menos um dentre um limite de circuito integrado e um limite de chip. Em outro aspecto, a inicialização do hipervisor pode incluir a configuração de HLOS para alocar espaço de memória de modo que cada endereço físico intermediário no espaço de endereço físico intermediário do HLOS é igual a um endereço físico correspondente em um espaço de endereço físico. Em outro aspecto adicional, a inicialização do hipervisor também pode incluir a autenticação do código de hipervisor e dados com o monitor de segurança. Em outro aspecto, o método pode incluir a configuração do código do hipervisor e dados para serem inacessíveis para pelo menos um dentre um processador de sinal digital e uma CPU incluída no processador de sinal digital enquanto o hipervisor é ativado.

[0008] Em outro aspecto, a desativação do hipervisor pode incluir a configuração de todos os bancos de contexto das unidades de gerenciamento de memória de sistema (SMMU) para ultrapassar a translação de segundo estágio e o desligamento das translações de segundo estágio para HLOS. Em um aspecto, a desativação do hipervisor pode incluir pelo menos um dentre a suspensão de restrição dos

acessos HLOS para interrupções de hardware, suspensão de restrição de acessos HLOS para os temporizadores de hardware, e suspensão de restrição dos acessos I/O do HLOS.

[0009] Em outro aspecto, o método pode incluir a determinação de se a sessão sandbox terminou, a realização de um procedimento de desligamento de sessão sandbox quando for determinado que a sessão de sandbox terminou, e a desativação do hipervisor depois da realização do procedimento de desligamento de sessão sandbox. Em outro aspecto, a determinação de se a sessão sandbox terminou pode incluir o recebimento de outro sinal indicando que a sessão sandbox terminou. Em outro aspecto, a realização do procedimento de desligamento de sessão sandbox pode incluir a liberação de todos os armazenadores temporários para um componente sandboxed e a restauração das tabelas de página de segunda translação para remover todos os fragmentos.

[0010] Em um aspecto, a ativação do hipervisor pode incluir a ativação das unidades de gerenciamento de memória de segundo estágio PL0 e PL1, a configuração das solicitações de interrupção a serem tomadas no modo de hipervisor, e a solicitação de acionadores SMMU para colocar todos os bancos de contexto SMMU ativos nas translações de primeiro estágio aninhados dentro das translações de segundo estágio. Em outro aspecto, o método também pode incluir a inicialização das comunicações interprocessador com um processador de sinal digital. Em outro aspecto, o método pode incluir adicionalmente o manuseio de falhas SMMU.

[0011] Em outro aspecto, a implementação do controle de acesso pode incluir a implementação das translações de segundo estágio. Em outro aspecto, a implementação do controle de acesso pode incluir pelo menos um dentre a retomada de restrição dos acessos do HLOS às

interrupções de hardware, a retomada de restrição dos acessos de HLOS aos temporizadores de hardware, e a retomada de restrição dos acessos I/O do HLOS. Em um aspecto, a implementação das translações de segundo estágio pode incluir o monitoramento de uma tentativa por parte do HLOS em alocar memória e fornecimento para o HLOS de um ou mais endereços físicos em um espaço de endereço físico acessível ao HLOS quando o HLOS tenta alocar memória.

[0012] Em outro aspecto, o método pode incluir a determinação de se um componente sandboxed está tentando alocar memória quando HLOS tenta alocar memória e fornecimento de endereços físicos para o componente sandboxed a partir de endereços físicos em um espaço de endereço físico quando for determinado que o componente sandboxed está tentando alocar memória. Em um aspecto, o fornecimento de endereços físicos para o componente sandboxed pode incluir a remoção dos endereços físicos a serem fornecidos para o componente sandboxed a partir dos endereços físicos no espaço de endereço físico que são acessíveis ao HLOS e fornecimento de endereços físicos para o componente sandboxed a partir de endereços físicos disponíveis no espaço de endereço físico.

[0013] Aspectos adicionais incluem um dispositivo de computação que pode incluir uma memória, e um processador acoplado à memória, onde o processador é configurado com instruções executáveis por processador para realizar as operações que podem incluir a inicialização de um hipervisor, um monitor de segurança, e um sistema operacional de alto nível (HLOS), desativação do hipervisor após a inicialização, monitoramento de use um sinal é recebido para iniciar a sessão sandbox, e implementação do controle de acesso enquanto o hipervisor está ativado. Em outro aspecto, o monitor de segurança pode ser um ARM

TrustZone®. Em outro aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações de modo que o hipervisor possa ser desativado ou ativado através de pelo menos um dentre um limite de circuito integrado e um limite de chip.

[0014] Em outro aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações de modo que a inicialização do hipervisor inclua a configuração do HLOS para alocar o espaço de memória de modo que cada endereço físico intermediário no espaço de endereço físico intermediário do HLOS seja igual a um endereço físico correspondente em um espaço de endereço físico. Em um aspecto, o processador pode ser configurado com instruções executáveis por hipervisor autenticando o código do hipervisor e os dados com o monitor de segurança. Em outro aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações que incluem a configuração do código do hipervisor e dados para que sejam inacessíveis a pelo menos um dentre um processador de sinal digital e uma CPU incluída no processador de sinal digital enquanto o hipervisor é ativado.

[0015] Em um aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações de modo que a desativação do hipervisor inclua a configuração de todas os bancos de contexto das unidades de gerenciamento de memória do sistema (SMMU) para ultrapassar a translação de segundo estágio, e desligar as translações de segundo estágio para o HLOS. Em outro aspecto, o processador pode ser configurado com as instruções executáveis por processador para realizar as operações de modo que a desativação do hipervisor inclua pelo menos um dentre suspensão de

restrição dos acessos do HLOS a interrupções de hardware, suspensão de restrição de acesso do HLOS a temporizadores de hardware, e suspensão de restrição dos acessos I/O do HLOS.

[0016] Em outro aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações que incluem a determinação de se a sessão sandbox já terminou, realizando um procedimento de desligamento de sessão sandbox quando for determinado que a sessão sandbox já terminou, e desativando o hipervisor depois da realização do procedimento de desligamento de sessão sandbox. Em outro aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações de modo que a determinação de se a sessão sandbox já terminou inclua o recebimento de outro sinal indicando que a sessão sandbox já terminou. Em outro aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações de modo que a realização do procedimento de desligamento da sessão sandbox inclua a liberação de todos os armazenadores temporários para um componente sandboxed e a restauração das tabelas de página de translação de segundo estágio para remoção de todos os fragmentos.

[0017] Em um aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações de modo que a ativação do hipervisor inclua a ativação das unidades de gerenciamento de memória de segundo estágio PL0 e PL1, a configuração das solicitações de interrupção para serem realizadas no modo de hipervisor, e a ativação de acionadores SMMU para colocar todos os bancos de contexto SMMU ativos nas translações de primeiro estágio aninhadas dentro das translações de segundo estágio. Em outro aspecto, o

processador pode ser configurado com instruções executáveis por processador para realizar operações que incluem a inicialização das comunicações de interprocessador com um processador de sinal digital. Em outro aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações que incluem o manuseio de falhas SMMU.

[0018] Em um aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações de modo que a implementação de controle de acesso inclua a implementação de translações de segundo estágio. Em outro aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações de modo que a implementação de controle de acesso inclua adicionalmente pelo menos um dentre a retomada de restrição de acessos do HLOS às interrupções de hardware, retomada de restrição dos acessos do HLOS aos temporizadores de hardware, e restrição de retomada dos acessos I/O do HLOS.

[0019] Em um aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações de modo que a implementação das translações de segundo estágio incluam o monitoramento de uma tentativa pelo HLOS de alocar memória e fornecer para o HLOS um ou mais endereços físicos em um espaço de endereço físico acessível ao HLOS quando o HLOS tenta alocar memória. Em outro aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações que incluem adicionalmente a determinação de se um componente sandboxed está tentando alocar a memória quando o HLOS tenta alocar a memória e fornecendo endereços físicos para o componente sandboxed a partir dos endereços físicos em um espaço de endereço

físico quando for determinado que o componente sandboxed está tentando alocar a memória. Em outro aspecto, o processador pode ser configurado com instruções executáveis por processador para realizar as operações de modo que o fornecimento dos endereços físicos para o componente sandboxed inclua a remoção de endereços físicos a serem fornecidos para o componente sandboxed a partir dos endereços físicos no espaço de endereço físico que são acessíveis para HLOS e fornecendo endereços físicos para o componente sandboxed a partir dos endereços físicos disponíveis no espaço de endereço físico.

[0020] Aspectos adicionais incluem um dispositivo de computação incluindo meios para inicializar um hipervisor, um monitor de segurança, e um sistema operacional de alto nível (HLOS); meios para desativar o hipervisor depois da inicialização; meios para monitorar um sinal a partir do monitor de segurança para inicializar uma sessão de sandbox; meios para ativar o hipervisor quando o sinal é recebido para inicializar a sessão sandbox; e meios para implementar o controle de acesso enquanto o hipervisor é ativado. Em outro aspecto, o monitor de segurança pode ser um ARM TrustZone®. Em outro aspecto, o hipervisor pode ser desativado ou ativado através de pelo menos um dentre um limite de circuito integrado e um limite de chip. Em outro aspecto, meios para inicialização do hipervisor podem incluir meios de configuração de HLOS para alocar o espaço de memória de modo que cada endereço físico intermediário no espaço de endereço físico intermediário do HLOS seja igual a um endereço físico correspondente em um espaço de endereço físico. Em outro aspecto, meios para inicialização do hipervisor podem incluir adicionalmente meios para autenticar o código de hipervisor e dados com o monitor de segurança. Em outro aspecto, o dispositivo de computação

pode incluir meios para configurar o código de hipervisor e dados para que sejam inacessíveis a pelo menos um dentre um processador de sinal digital e uma CPU incluída no processador de sinal digital enquanto o hipervisor é ativado.

[0021] Em um aspecto, meios para desativar o hipervisor podem incluir meios para configurar todas os bancos de contexto das unidades de gerenciamento de memória de sistema (SMMU) para ultrapassar a translação de segundo estágio e meios para desligar as translações de segundo estágio para HLOS. Em outro aspecto, meios para desativar o hipervisor podem incluir adicionalmente pelo menos um dos meios para suspender a restrição aos acessos do HLOS às interrupções de hardware, meios para suspender a restrição dos acessos do HLOS aos temporizadores de hardware, e meios para suspender a restrição aos acessos I/O pelo HLOS.

[0022] Em um aspecto, o dispositivo de computação pode incluir adicionalmente meios para determinar se a sessão de sandbox já terminou, meios para realizar um procedimento de desligamento de sessão sandbox quando for determinado que a sessão sandbox já terminou, e meios para desativar o hipervisor depois da realização do procedimento de desligamento de sessão sandbox. Em outro aspecto, meios para determinar se a sessão sandbox já terminou podem incluir meios para o recebimento de outro sinal indicando que a sessão de sandbox já terminou. Em outro aspecto, meios de realização do procedimento de desligamento de sessão de sandbox podem incluir meios para liberar todos os armazenadores temporários para um componente sandboxed e meios para restaurar as tabelas de página de translação de segundo estágio para remover todos os fragmentos

[0023] Em um aspecto, meios para ativar o hipervisor podem incluir meios para ativar as unidades de

gerenciamento de memória de segundo estágio PL0 e PL1, meios para configurar as solicitações de interrupção a serem realizadas no modo de hipervisor, e meios para chamar os acionadores SMMU para colocar todos os bancos de contexto SMMU ativos nas translações de primeiro estágio aninhadas dentro das translações de segundo estágio. Em outro aspecto, o dispositivo de computação pode incluir meios para inicializar as comunicações de interprocessador com um processador de sinal digital. Em outro aspecto, o dispositivo de computação pode incluir meios para manusear as falhas SMMU.

[0024] Em um aspecto, meios para implementação do controle de acesso podem incluir meios para implementar as translações de segundo estágio. Em outro aspecto, meios para implementação do controle de acesso podem incluir pelo menos um dentre os meios para retomar a restrição dos acessos HLOS às interrupções de hardware, meios para retomar a restrição dos acessos HLOS aos temporizadores de hardware, e meios para retomar a restrição dos acessos I/O de HLOS. Em outro aspecto, os meios de implementação das translações de segundo estágio podem incluir meios para o monitoramento de uma tentativa realizada pelo HLOS em alocar memória e meios para fornecer para o HLOS um ou mais endereços físicos em um espaço de endereço físico acessível ao HLOS quando o HLOS tenta alocar memória. Em outro aspecto, o dispositivo de computação também pode incluir meios para determinar se um componente sandboxed está tentando alocar memória quando HLOS tenta alocar memória e meios par fornecer endereços físicos para o componente sandboxed a partir de endereços físicos em um espaço de endereço físico quando for determinado que o componente sandboxed está tentando alocar memória. Em outro aspecto, os meios para fornecimento de endereços físicos para o

componente sandboxed podem incluir meios de remoção dos endereços físicos a serem fornecidos para o componente sandboxed a partir dos endereços físicos no espaço de endereço físico que são acessíveis ao HLOS e meios de fornecimento dos endereços físicos para o componente sandboxed a partir de endereços físicos disponíveis no espaço de endereço físico.

[0025] Em aspectos adicionas, um meio de armazenamento legível por processador não transitório pode ter armazenado no mesmo instruções de software executáveis por processador configuradas para fazer com que um processador realize as operações para o gerenciamento de memória em um dispositivo de computação, as operações incluindo a inicialização de um hipervisor, um monitor de segurança, e um sistema operacional de alto nível (HLOS); a desativação do hipervisor após a inicialização; o monitoramento de um sinal do monitor de segurança para iniciar uma sessão sandbox; a ativação do hipervisor quando o sinal é recebido para iniciar a sessão sandbox; e a implementação de controle de acesso enquanto o hipervisor é ativado. Em outro aspecto, o monitor de segurança pode ser um ARM TrustZone®. Em outro aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações de modo que o supervisor possa ser desativado ou ativado através de pelo menos um dentre um limite de circuito integrado e um limite de chip.

[0026] Em um aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações de modo que a inicialização do hipervisor inclua a configuração de HLOS para alocar espaço de memória de modo que cada endereço físico intermediário no espaço de

endereço físico intermediário do HLOS seja igual a um endereço físico correspondente em um espaço de endereço físico. Em outro aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações de modo que a inicialização do hipervisor inclua a autenticação do código do hipervisor e dados com o monitor de segurança. Em outro aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações que incluem a configuração do código do hipervisor e dados a serem inacessíveis para pelo menos um dentre um processador de sinal digital e uma CPU incluída no processador de sinal digital enquanto o hipervisor é ativado.

[0027] Em um aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações de modo que a desativação do hipervisor inclua a configuração de todos os bancos de contexto das unidades de gerenciamento de memória de sistema (SMMU) para ultrapassar a translação de segundo estágio e desligar as translações de segundo estágio para o HLOS. Em outro aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize operações de modo que a desativação do hipervisor inclua adicionalmente pelo menos uma dentre a suspensão da restrição dos acessos do HLOS às interrupções de hardware, a suspensão da restrição dos acessos do HLOS aos temporizadores de hardware, e a suspensão da restrição dos acessos a I/O por parte do HLOS.

[0028] Em um aspecto, as instruções de software executáveis por processador armazenadas podem ser

configuradas para fazer com que um processador realize as operações que incluem a determinação de se a sessão de sandbox já terminou, a realização de um procedimento de desligamento de sessão sandbox quando for determinado que a sessão sandbox já terminou, e a desativação do hipervisor pós a realização do procedimento de desligamento de sessão sandbox. Em um aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações de modo que a determinação de se a sessão sandbox já terminou inclua o recebimento de outro sinal indicando que a sessão sandbox já terminou. Em outro aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações de modo que a realização do procedimento de desligamento de sessão sandbox inclua a liberação de todos os armazenadores para um componente sandboxed e restaurando as tabelas de página de segunda translação para remover todos os fragmentos.

[0029] Em um aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações de modo que a ativação do hipervisor inclua a ativação das unidades de gerenciamento de memória de segundo estágio PL0 e PL1, a configuração de solicitações de interrupções a serem realizadas no modo de hipervisor, e a ativação dos acionadores SMMU para colocar todos os bancos de contexto SMMU ativos nas translações de primeiro estágio aninhadas dentro das translações de segundo estágio. Em outro aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações que incluem a inicialização das comunicações

interprocessador com um processador de sinal digital. Em outro aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações que incluem o manuseio de falhas SMMU.

[0030] Em um aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações de modo que a implementação do controle de acesso inclua a implementação da translações de segundo estágio. Em outro aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações de modo que a implementação do controle de acesso inclua adicionalmente pelo menos um dentre retomada da restrição dos acessos do HLOS às interrupções de hardware, a retomada de restrição dos acessos HLOS aos temporizadores de hardware, e a retomada da restrição dos acessos I/O por parte do HLOS. Em outro aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações de modo que a implementação de translações de segundo estagio inclua o monitoramento de uma tentativa por parte do HLOS em alocar memória e fornecendo para o HLOS um ou mais endereços físicos em um espaço de endereço físico acessível ao HLOS quando o HLOS tenta alocar memória. Em outro aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações que incluem a determinação de se o componente sandboxed está tentando alocar memória quando o HLOS tenta alocar memória e fornecendo endereços físicos para o componente sandboxed a partir de endereços físicos em um espaço de endereço físico quando for

determinado que o componente sandboxed está tentando alocar memória. Em outro aspecto, as instruções de software executáveis por processador armazenadas podem ser configuradas para fazer com que um processador realize as operações de modo que o fornecimento dos endereços físicos para o componente sandboxed inclua a remoção dos endereços físicos a serem fornecidos para o componente sandboxed dos endereços físicos no espaço de endereço físico que são acessíveis ao HLOS e fornecendo os endereços físicos para o componente sandboxed a partir dos endereços físicos disponíveis no espaço de endereço físico.

Breve Descrição dos Desenhos

[0031] Os desenhos em anexo, que são incorporados aqui e constituem parte dessa especificação, ilustram aspectos ilustrativos da invenção, e juntamente com a descrição geral fornecida acima e com a descrição detalhada fornecida abaixo, servem para explicar as características da invenção.

[0032] A figura 1 é um diagrama em bloco de componente de um dispositivo de computação de aspecto;

[0033] A figura 2 é um diagrama em bloco funcional de módulos de um dispositivo de computação;

[0034] A figura 3 é um diagrama arquitetônico de computador em camadas de um sistema de computação de aspecto;

[0035] As figuras 4 e 5 são diagramas arquitetônicos de computador em camadas de componentes lógicos de aspecto em máquinas virtuais;

[0036] A figura 6 é um bloco funcional e diagrama de mapa de memória ilustrando mapeamentos de endereço de memória de dois estágios em um dispositivo de computação implementando uma máquina virtual de sistema;

[0037] A figura 7 é um diagrama de mapa de memória ilustrando mapeamentos de endereço de memória de dois estágios em um dispositivo de computação implementando uma máquina virtual de sistema enquanto o hipervisor é desativado;

[0038] A figura 8 é um diagrama de mapa de memória ilustrando mapeamentos de endereço de memória de dois estágios em um dispositivo de computação implementando uma máquina virtual de sistema enquanto o hipervisor é ativado durante uma sessão sandbox;

[0039] A figura 9 é um diagrama de mapa de memória ilustrando os mapeamentos de endereço de memória de dois estágios em um dispositivo de computação implementando uma máquina virtual de sistema com a memória virtual compartilhada enquanto o hipervisor é ativado durante uma sessão sandbox;

[0040] A figura 10 é um fluxograma de processo ilustrando um método de aspecto para ativar e desativar seletivamente um hipervisor;

[0041] A figura 11 é um fluxograma de chamada ilustrando a sinalização envolvida na inicialização de um ambiente de memória virtual compartilhado em um dispositivo de computação;

[0042] A figura 12 é um fluxograma de processo ilustrando um método de aspecto de configuração de uma sessão de memória virtual compartilhada;

[0043] A figura 13 é um fluxograma de processo ilustrando um método de aspecto de desativação de um hipervisor;

[0044] A figura 14 é um fluxograma de chamada ilustrando a sinalização envolvida na inicialização de uma sessão sandbox de acordo com um dispositivo de computação;

[0045] A figura 15 é um fluxograma de processo ilustrando um método de aspecto de ativação de um hipervisor;

[0046] As figuras 16A e 16B são fluxogramas de processo ilustrando métodos de aspecto de implementação de translações de segundo estágio;

[0047] A figura 17 é um fluxograma de processo ilustrando um método de aspecto de realização de um desligamento de sessão sandbox;

[0048] A figura 18 é um diagrama em bloco de componente de um dispositivo de computação adequado para implementação de vários aspectos;

[0049] A figura 19 é um diagrama em bloco de componente de outro dispositivo de computação adequado para implementação de vários aspectos;

Descrição Detalhada

[0050] Os vários aspectos serão descritos em detalhes com referência aos desenhos em anexo. Sempre que possível, as mesmas referências numéricas serão utilizadas por todos os desenhos para fazer referência a partes iguais ou similares. Referências feitas a exemplos e implementações em particular servem à finalidade ilustrativa, e não devem limitar o escopo da invenção ou das reivindicações.

[0051] Em uma visão geral, os vários aspectos incluem um dispositivo de computação e métodos para a implementação seletiva de um hipervisor no dispositivo de computação para garantir de forma eficiente o controle de acesso com um hipervisor. Em vários aspectos, o hipervisor é normalmente desativado e é ativado quando uma condição é detectada que exigiria que o hipervisor implementasse o controle de acesso (isso é, uma "sessão de sandbox"). Enquanto desativado, o hipervisor pode retomar as

translações de segundo estágio. Em alguns aspectos, enquanto ativado, o hipervisor também pode retomar outras atividades que são suspensas enquanto o hipervisor está desativado, incluindo a restrição de acessos de entrada/saída (I/O), acessos de interrupção de hardware, e/ou acessos a temporizador de hardware. Dessa forma, pela ativação seletiva do hipervisor apenas enquanto sandboxing é necessário, o dispositivo de computação pode aperfeiçoar o desempenho geral e experiência de usuário enquanto mantém um ambiente operacional seguro.

[0052] Nos vários aspectos, o dispositivo de computação pode incluir memória e um processador acoplado à memória que é configurado com um hipervisor, implementado em hardware (isso é, um hipervisor de metal bare), em software (isso é, um hipervisor hospedado operando dentro de um ambiente de sistema operacional convencional), ou em uma combinação de hardware e software. O hipervisor pode criar e gerenciar adicionalmente uma ou mais máquinas virtuais para vários aplicativos, sistemas operacionais, processos, sinais, etc.

[0053] O termo "dispositivo de computação" é utilizado aqui para fazer referência a qualquer um ou todos os telefones celulares, smartphones, aparelhos de multimídia pessoal ou móvel, assistentes de dados pessoais (PDAs), computadores laptop, computadores tablet, smartbooks, computadores palm-top, receptores de correio eletrônico sem fio, telefones celulares ativados para Internet e multimídia, controladores de jogos sem fio, e dispositivos eletrônicos pessoais similares que incluem um processador programável e uma memória, e operam sob energia de bateria de modo que os métodos de conservação de energia sejam benéficos. Enquanto os vários aspectos são particularmente úteis nos dispositivos móveis, tal como

telefones celulares, que possuem energia de processamento limitada e capacidade de bateria limitada, os aspectos são geralmente úteis em qualquer dispositivo de computação que podem beneficiar do desempenho de processador aperfeiçoado e consumo de energia reduzido.

[0054] Os termos "monitor de máquina virtual", "VMM", e "hipervisor" são utilizados de forma intercambiável aqui para fazer referência a um gerenciador de máquina virtual. O termo "sistema operacional de alto nível" (HLOS) é utilizado aqui para fazer referência a um sistema operacional convidado operando em uma máquina virtual que o hipervisor gerencia. Em um aspecto, o hipervisor pode segregar HLOS durante uma sessão sandbox.

[0055] Os termos "translação de Estágio 1" e "translação de primeiro estágio" são utilizados de forma intercambiável aqui para fazer referência a uma translação ou mapeamento de um endereço de memória virtual (um "VA") para um endereço de memória intermediário (um "IPA"). Em um exemplo, HLOS pode realizar uma translação de primeiro estágio a partir dos endereços virtuais alocados para um processo operando no HLOS para os endereços físicos intermediários mantidos no espaço de endereço físico intermediário HLOS.

[0056] Os termos "translação de estágio 2" e "translação de segundo estágio" são utilizados de forma intercambiável aqui para fazer referência a uma translação ou mapeamento a partir de um endereço físico intermediário para um endereço de memória física (um "PA"). Em um exemplo, o supervisor ou uma unidade de gerenciamento de memória de sistema (uma "SMMU") pode realizar uma translação de segundo estágio a partir dos endereços físicos intermediários alocados ao HLOS para endereços

físicos mantidos pelo hipervisor no espaço de endereço físico.

[0057] O termo "sessão sandbox" é utilizado aqui para fazer referência a um período de tempo no qual o hipervisor está realizando o controle de acesso entre duas ou mais entidades. Em um aspecto, uma sessão sandbox pode começar quando o hipervisor é alertado sobre o conteúdo protegido (por exemplo, conteúdo protegido pelas técnicas de gerenciamento de direitos digitais (DRM)) que devem ser processados separadamente no dispositivo de computação (por exemplo, um dispositivo de vídeo seguro para reprodução de mídia DRM) e podem terminar quando essa separação não é mais necessária, tal como quando o conteúdo protegido acabou de ser processado ou reproduzido e o serviço ou aplicativo foi liberado.

[0058] O termo "componente sandboxed" é utilizado aqui para fazer referência a um componente, aplicativo, processo, etc. que o hipervisor separa (isso é, sandboxes) do HLOS. Em um aspecto, o hipervisor pode alocar endereços físicos no espaço de endereço de memória física para o componente sandboxed que não se sobrepõem aos endereços físicos alocados ao HLOS.

[0059] O termo "entidade de compartilhamento" é utilizado aqui para se referir a um componente, aplicativo, processo, etc. que compartilha a memória virtual com HLOS. Em um aspecto, a entidade de compartilhamento e o HLOS pode compartilhar acesso a um ou mais endereços físicos no espaço de endereço de memória física.

[0060] Pela criação e gerenciamento de máquinas virtuais, o hipervisor pode criar um "sandbox" ou separação segura em torno de várias operações ou dados, incluindo sistemas operacionais, aplicativos, processos, etc. O hipervisor pode utilizar uma sandbox para limitar o acesso

a várias características, fornecendo, assim, a segurança para operações ou dados. Por exemplo, um HLOS pode operar como um sistema operacional convidado dentro de uma máquina virtual que o hipervisor gerencia, e o hipervisor pode gerenciar um sinal de vídeo processado fora da máquina virtual do HLOS de modo que o HLOS pode ser esquecido (isso é, incapaz de detectar ou acessar) o sinal de vídeo.

[0061] No entanto, existe um custo de desempenho associado com a utilização do hipervisor para garantir o controle de acesso. O teste em pequena escala mostrou que utilizando o hipervisor para garantir o controle de acesso pode causar uma queda de 5% a 30% no desempenho dependendo da referência. Os vários aspectos superam essa penalidade de desempenho pela implementação do hipervisor apenas quando existe a necessidade de dados e/ou segurança de software permitidos pelo sandbox.

[0062] Em vários aspectos, um HLOS pode operar em uma máquina virtual gerenciada pelo hipervisor. HLOS pode manter uma tabela de página de endereço físico intermediária para uso na alocação de endereço virtual para vários processo ou aplicativos rodando no HLOS. HLOS pode alocar a memória diretamente a partir do espaço de endereço de memória física, garantindo, assim, que os endereços físicos intermediários sejam sempre iguais que os endereços físicos. Em outras palavras, os endereços físicos intermediários no espaço de endereço físico intermediário do HLOS são iguais aos endereços físicos no espaço de endereço físico durante todo o tempo. Pela garantia de que HLOS possa alocar memória de modo que os endereços físicos intermediários sejam iguais aos endereços físicos durante todo o tempo, os vários aspectos permitem que o hipervisor seja seletivamente ativado e desativado, aperfeiçoando, dessa forma, o desempenho total visto que o desempenho do

hipervisor seja incorrido apenas quando uma sessão sandbox é necessária.

[0063] Nos vários aspectos, HLOS pode alocar memória diferentemente do espaço de endereço de memória física. No entanto, enquanto o hipervisor está ativado, o hipervisor pode restringir o acesso HLOS para o espaço de endereço de memória física, permitindo, assim, que HLOS aloque memória a partir de apenas uma parte do espaço de endereço de memória física.

[0064] Em um aspecto, o processador do dispositivo de computação pode monitorar situações nas quais sandboxing é necessário (isso é, dados e/ou software que deve ser processado em uma sessão de sandbox). Em um aspecto, uma sessão de sandbox pode ser uma situação ou período de tempo no qual sandboxing é necessário para garantir o controle de acesso entre processos, aplicativos separados, etc. Por exemplo, o dispositivo de computação pode detectar que um sinal seguro tenha sido recebido (por exemplo, um sinal de vídeo sujeito ao gerenciamento de direitos digitais) ou que um segundo sistema operacional esteja sendo iniciado que deve ser mantido isolado do HLOS. Quando a necessidade de uma sessão sandbox é detectada, o dispositivo de computação pode ativar o hipervisor. Depois de ser ativado, o hipervisor pode implementar translações de segundo estágio e limitar o HLOS para alocação de memória a partir de apenas uma parte do espaço de endereço de memória física, estabelecendo, assim, a sessão sandbox. O hipervisor, enquanto ativado, também pode retomar outras operações de controle de acesso, tal como restrição de um ou mais acessos I/O, acessos interrompidos de hardware, e acessos de temporizador de hardware.

[0065] Em outro aspecto, o hipervisor pode monitorar o final de uma sessão sandbox, por exemplo, o

hipervisor pode determinar se o sinal de vídeo seguro não está mais sendo recebido. Quando a sessão sandbox é encerrada, o hipervisor pode realizar um desligamento de sessão. No processo de desligamento de sessão, o hipervisor pode liberar os recursos alocados para o componente sandboxed. Por exemplo, o hipervisor pode liberar os recursos alocados para um processador de sinal digital para processar o sinal de vídeo seguro, permitindo, assim, que HLOS aloque memória a partir de todo o espaço de endereço de memória física. Em um aspecto adicional, o hipervisor pode ser desativado até que a próxima situação de sandboxing comece.

[0066] Em outro aspecto, o hipervisor pode permitir translações de segundo estágio para HLOS e uma entidade de compartilhamento. Nesse aspecto, a entidade de compartilhamento e HLOS podem compartilhar memória virtual e compartilhar o acesso aos endereços físicos no espaço de endereço de memória física. Em um aspecto adicional, o hipervisor pode realizar o controle de acesso entre HLOS e a entidade de compartilhamento para os endereços de memória que não são compartilhados.

[0067] Os vários aspectos podem ser implementados em uma ampla variedade de arquiteturas de computador de processador único e múltiplos processadores, um exemplo das quais é ilustrado na figura 1. Um dispositivo de computação 100 pode incluir vários processadores heterogêneos, tal como o processador de sinal digital ilustrado (DSP) 102, o processador de modem 104, o processador de gráficos 106, e o processador de aplicativo 108. O dispositivo de computação 100 também pode incluir um ou mais coprocessadores de vetor 110 conectados a um ou mais dos processadores 102-108. Cada processador 102-110 pode incluir um ou mais núcleos, e cada processador/núcleo pode

realizar operações independentes de outros processadores/núcleos. Cada processador 102-110 também pode incluir um a memória (não ilustrada) e/ou um controlador de sistema de gerenciamento de memória. Em um aspecto, os componentes do dispositivo de computação 100 podem ser localizados em um único substrato e/ou acoplados de perto juntos como um sistema em chip (SOC) 125.

[0068] O dispositivo de computação 100 pode incluir um conjunto de circuitos analógicos e um conjunto de circuitos personalizado 114 para o gerenciamento de dados de sensor, conversões de analógico para digital, transmissões de dados sem fio, e realização de outras operações especializadas, tal como processamento de sinais de áudio codificados para jogos e filmes. O dispositivo de computação 100 pode incluir adicionalmente componentes e recursos de sistema 116, tal como reguladores de voltagem, osciladores, circuitos travados em fase, pontes periféricas, controladores de dados, controladores de memória, controladores de sistema, portas de acesso, temporizadores e outros componentes similares utilizados para suportar os processadores, memórias e clientes rodando em um dispositivo de computação. Cada componente ou recurso de sistema 116 pode incluir adicionalmente uma memória (não ilustrada) e/ou um controlador de sistema de gerenciamento de memória.

[0069] Em vários aspectos, o processador de aplicativos 108 pode ser uma unidade de processamento central (CPU), um componente de uma CPU, ou uma unidade de processamento acoplada a uma CPU. Em um aspecto, a CPU pode ser configurada para ler e escrever informação para e de várias memórias dos processadores 102-110, componentes e recursos de sistema 116 e/ou periféricos, que podem ser

alcançados através dos controladores de sistema de gerenciamento de memória dos respectivos processadores 102-110, recursos 116 e/ou periféricos.

[0070] O dispositivo de computação 100 pode incluir adicionalmente um módulo de entrada/saída (não ilustrado) para comunicação entre os componentes e recursos, tal como um relógio 118 e um regulador de voltagem 120. Os processadores 102-108 podem ser interconectados a um ou mais elementos de memória 112, recursos 116, conjunto de circuitos personalizado 114, e vários outros componentes de sistema através de um módulo de interconexão/barramento 122.

[0071] Como mencionado acima, o dispositivo de computação 100 pode incluir um ou mais coprocessadores de vetor 110 conectados a um ou mais dos processadores 102-108. Tais coprocessadores de vetor 110 podem ser particularmente úteis para processamento de aplicativos que exigem execução rápida e paralela, tal como aplicativos de multimídia e sequenciamento de vídeo. Em um aspecto, o coprocessador de vedor 110 pode implementar uma arquitetura de conjunto de instruções (ISA) de múltiplos dados de instrução única (SIMD) que inclui registros de hardware, memória e/ou hardware de execução independentes. O coprocessador do vetor SIMD pode ser uma parte de, ou pode ser acoplado de forma próxima ao processador principal do dispositivo de computação 100 (por exemplo, processador de aplicativos 108, CPU, etc.)

[0072] A figura 2 ilustra um diagrama de componente de um dispositivo de computação de aspecto 206 capaz de manter um ambiente virtual seguro (isso é, um sandbox). O sistema operacional inseguro 208 (isso é, HLOS) pode estar em comunicação com o hipervisor 212. O

hipervisor pode estar em comunicação com a memória física 216. Em um aspecto, o hipervisor pode agir como um intermediário entre o sistema operacional inseguro 208 e a memória física 216 ou outro hardware (não ilustrado). Em outro aspecto, o hipervisor 212 pode facilitar o mapeamento de endereços físicos intermediários (IPA) mantidos pelo sistema operacional inseguro 208 para endereços físicos (PA) na memória física 216.

[0073] Em um aspecto, o hipervisor 212 também pode estar em comunicação com um monitor de segurança 214 (por exemplo, um ARM TrustZone®). O monitor de segurança 214 pode agir como um guardião, garantindo que apenas dados seguros entrem e saiam do ambiente virtual seguro 210. O ambiente virtual seguro 210 pode, por sua vez, estar em comunicação com uma rede segura 204. O ambiente virtual seguro 210 pode transmitir ou receber dados da rede segura 204. Em um exemplo, o ambiente virtual seguro 210 pode incluir um processador de sinal digital (isso é, um DSP) que pode receber dados sensíveis da rede segura 204. Nesse exemplo, os dados sensíveis podem ser um sinal contendo dados de vídeo regulados pelas limitações de gerenciamento de direitos digitais. Em um aspecto, o monitor de segurança 214 pode comunicar com o hipervisor 212 para garantir que os dados sensíveis sejam armazenados em uma parte da memória física 216 que é inacessível ao sistema operacional inseguro (ou outros sistemas ou processos). Em um aspecto adicional, esses dados sensíveis podem ser armazenados na memória criptografada (não ilustrada) dentro da memória física 216.

[0074] A figura 3 ilustra uma arquitetura em camadas de um processador ilustrando componentes lógicos e interfaces em um sistema de computação típico. A arquitetura do sistema de computação ilustrado 300 inclui

ambos os componentes de hardware 322 e os componentes de software 320. Os componentes de software 320 podem incluir um sistema operacional 302, um módulo de biblioteca 304, e um ou mais programas de aplicativo (A_0 a A_n) 306. Os componentes de hardware 322 podem incluir periféricos 308 (por exemplo, aceleradores de hardware, dispositivos de entrada e saída, etc.), uma unidade de processamento central (CPU) 310, uma unidade de gerenciamento de memória de unidade de processamento central (MMU CPU) 316, uma ou mais unidades de gerenciamento de memória de sistema (aqui, "MMU de sistema" ou "SMMU") 312, e uma ou mais memórias 314.

[0075] Software de aplicativo escrito para dispositivos de computação móveis podem ser compilados em código executável, que é o que é comumente referido como "aplicativos", "apps" ou programas de aplicativo 306. Cada programa de aplicativo 306 pode ser um processo único ou sequência, e pode incluir uma pluralidade de processos ou sequências.

[0076] Os programas de aplicativo 306 podem emitir chamadas de biblioteca de linguagem de nível alto (HLL) para o módulo de biblioteca 304 através de uma interface de programa de aplicativo (API). O módulo de biblioteca 304 pode invocar serviços (Por exemplo, através de chamadas do sistema operacional) no sistema operacional 302 através de uma interface binária de aplicativo (ABI). O sistema operacional 302 pode comunicar com os componentes de hardware utilizando uma arquitetura de conjunto de instrução específica (ISA), que é uma listagem dos códigos de operação específicos (opcode) e comandos nativos implementados pelo hardware 322. Dessa forma, a arquitetura de conjunto de instruções define o hardware 322 como observado pelo sistema operacional 302.

[0077] O sistema operacional 302 pode ser responsável pela coordenação e controle da alocação e uso de várias memórias 314 entre os programas de aplicativo 306, que pode incluir a divisão da memória física através de múltiplos programas de aplicativo (A0-An) 306. Em um aspecto, o sistema operacional 302 pode incluir um ou mais sistemas de gerenciamento de memória (por exemplo, um gerenciador de memória virtual, etc.) para o gerenciamento da alocação e uso da memória do sistema por vários programas de aplicativo (A_0 a A_n) 306. Os sistemas de gerenciamento de memória podem funcionar para garantir que a memória utilizada por um processo não interfira com a memória já em uso por outro processo.

[0078] Em um aspecto, o sistema operacional 302 pode incluir um gerenciador de memória virtual (VMM OS) configurado para realizar as operações de "endereçamento virtual" que permitem que o sistema operacional 302 faça com que um endereço físico em particular apareça para outro endereço (isso é, um endereço virtual). As operações de endereçamento virtual podem incluir a alocação de endereço de memória virtual para os programas de aplicativo (A_0-A_n) 306. Incluindo um gerenciador de memória virtual dentro do sistema de operação 302 é possível se simplificar a coordenação e controle da memória do sistema entre os múltiplos processos ou programas de aplicativo (A_0-A_n) 306.

[0079] Em adição aos sistemas de gerenciamento de memória com base em software (por exemplo, VMM OS, etc.) discutidos acima, o sistema pode incluir um ou mais sistemas de gerenciamento de memória com base em hardware, tal como a unidade de gerenciamento de memória (MMU) da unidade de processamento central (CPU) 316 e a MMU de sistema 312 ilustrada na figura 3. A MMU CPU 316 e a MMU de sistema 312 podem, cada uma, incluir um ou mais componentes

de hardware responsáveis pela realização de várias operações relacionadas com memória, tal como a translação dos endereços virtuais para endereços físicos, controle de memória temporária, arbitragem de barramento e proteção de memória. Em um aspecto, a MMU CPU 316 pode ser responsável pelo fornecimento de serviços de tradução de endereço e funcionalidades de proteção para a CPU principal 310, e a MMU do sistema 312 pode ser responsável pelo fornecimento de serviços de tradução de endereço e funcionalidades de proteção para outros componentes de hardware (por exemplo, processador de sinal digital, processador de modem, processador de gráficos, etc.).

[0080] Em vários aspectos, um ou mais dos sistemas de gerenciamento de memória (por exemplo, MMU de sistema 312, MMU CPU 316, etc.) podem incluir um armazenador temporário look-aside de tradução (TLB), que é uma memória temporária que pode ser utilizada para traduções de endereço de memória (por exemplo, traduzindo endereços virtuais em endereços físicos, etc.). Em um aspecto, o armazenador look-aside de tradução (TLB) pode ser uma memória endereçável por conteúdo (CAM), que pode ser uma memória de conjunto associativo de hardware na qual a informação armazenada é organizado em formato de valor chave (por exemplo, tabela hash). As chaves podem ser endereços virtuais e os valores podem ser endereços físicos. Em vários aspectos, o armazenador look-aside de tradução pode ser gerenciado por hardware, gerenciado por software, ou gerenciado por uma combinação de hardware e software. Com um armazenador look-aside de tradução gerenciado por hardware, o formato dos registros do armazenador look-aside de tradução pode não ser visível para o software, e, dessa forma, pode ser diferente para diferentes tipos de unidades de processador central.

[0081] Geralmente, como parte de um processo de tradução de endereço de memória, um sistema de gerenciamento de memória (por exemplo, VMM OS, MMU de sistema 312, MMU de CPU 316, etc.) pode realizar uma busca de memória endereçável por conteúdo para solicitar um endereço físico a partir do armazenador look-aside de tradução pelo envio para o armazenador look-aside de tradução de um endereço virtual como a chave. Se uma chave de endereço virtual possui um valor de endereço físico correspondente no armazenador look-aside de tradução (isso é, o "hit TLB" ocorre), a busca de memória endereçável por conteúdo pode recuperar e retornar o endereço físico correspondente. Se o endereço solicitado não estiver no armazenador look-aside de tradução (isso é, "perda TLB" ocorre), o processo de tradução de endereço de memória pode realizar um walk de página (por exemplo, um walk de página de software, um walk de página de hardware, etc.) pela leitura do conteúdo de múltiplos locais de memória e computando o endereço físico. Depois que o endereço físico é determinado pelo walk de página, um mapeamento de endereço virtual para endereço físico pode ser armazenado no armazenador look-aside de tradução.

[0082] Em aspectos que incluem um armazenador look-aside de tradução gerenciado por software, uma perda TLB pode fazer com que o sistema operacional walk as tabelas de página e realize a tradução em software. Em aspectos que incluem um armazenador look-aside de tradução gerenciado por hardware, o sistema de gerenciamento de memória pode realizar um walk de tabela de hardware para determinar se um registro de tabela de página válido existe para uma chave de endereço virtual especificada.

[0083] Os vários aspectos fornecem sistemas de gerenciamento de memória que utilizam técnicas de

virtualização. As tecnologias de virtualização permitem a abstração (ou virtualização) dos recursos de computação, que podem ser alcançados pela colocação de um programa de controle (por exemplo, um Monitor de Máquina Virtual "VMM" ou hipervisor) entre o sistema operacional e o hardware. As técnicas de virtualização são comumente implementadas em uma máquina virtual (VM), que pode ser um aplicativo de software que executa programas de aplicativo como uma máquina de hardware físico. A máquina virtual fornece uma interface entre programas de aplicativo e o hardware de execução, permitindo que os programas de aplicativo amarrados a uma arquitetura de conjunto de instruções específica sejam executados em hardware implementando uma arquitetura de conjunto de instruções diferente.

[0084] As figuras 4 e 5 ilustram componentes lógicos em um sistema de computador típico implementando uma máquina virtual. As máquinas virtuais podem ser categorizadas em duas categorias gerais: máquinas virtuais de sistema, e máquinas virtuais de processo. As máquinas virtuais de sistema permitem o compartilhamento do hardware físico subjacente entre diferentes processos ou aplicativos. As máquinas virtuais de processo, por outro lado, suportam um processo ou aplicativo único.

[0085] A figura 4 é um diagrama arquitetônico em camadas ilustrando camadas lógicas de um dispositivo de computação 400 implementando uma máquina virtual de processo 410. O sistema de computador 400 pode incluir componentes de hardware 408 e software que incluem um módulo de processo de aplicativo 402, um módulo de virtualização 404, e um sistema operacional 406.

[0086] Como discutido acima com referência à figura 3, os componentes de hardware são visíveis apenas para os programas de aplicativo 306 através do sistema

operacional 302, e ABI e API definem efetivamente as características de hardware disponíveis pra os programas de aplicativo 306. O módulo de software de virtualização 404 pode realizar operações lógicas no nível de ABI/API e/ou emular chamadas de sistema operacional ou chamadas de biblioteca de modo que o processo de aplicativo 402 comunique com o módulo de software de virtualização 404 da mesma forma que se comunicaria com os componentes de hardware (isso é, através de chamadas de sistema ou biblioteca). Dessa forma, o processo de aplicativo 402 visualiza a combinação de módulo de virtualização 404 sistema operacional 406, e hardware 408 como uma máquina única, tal como a máquina virtual de processo 410 ilustrada na figura 4. Isso simplifica a tarefa do projetista de aplicativo visto que o software de aplicativo não precisa se preocupar com a arquitetura real dos dispositivos de computação nos quais o aplicativo será executado por fim.

[0087] A máquina virtual de processo 410 existe apenas para suportar um processo de aplicativo único 402, e portanto, é criada com o processo 402 e encerrada quando o processo 402 encerra a execução. O processo 402 que roda na máquina virtual 410 é chamado de "convidado" e a plataforma subjacente é chamada de "hospedeira". O software de virtualização 404 que implementa a máquina virtual de processo é tipicamente chamada de software de runtime (ou simplesmente "runtime").

[0088] A figura 5 é um diagrama arquitetônico em camadas ilustrando as camadas lógicas em um dispositivo de computação 500 implementando uma máquina virtual do sistema 510. O sistema de computador pode incluir componentes de hardware (por exemplo, hardware de execução, memória, dispositivos I/O, etc.)508 e componentes de software que incluem um módulo de programas de aplicativo 502, um

sistema operacional 504, e um módulo de virtualização 506. O software que roda em cima do módulo de virtualização 506 é referido como software "convidado" e a plataforma subjacente que suporta o módulo de virtualização é referida como hardware "hospedeiro".

[0089] Diferentemente das máquinas virtuais de processo, uma máquina virtual de sistema 510 fornece um ambiente completo no qual múltiplos sistemas operacionais (chamados de "sistemas operacionais convidados") podem coexistir. Da mesma forma, a plataforma de hardware hospedeiro pode ser configurada para suportar simultaneamente vários ambientes de sistema operacional convidados isolados. O isolamento entre os sistemas operacionais sendo executados simultaneamente adiciona um nível de segurança ao sistema. Por exemplo, se a segurança em um sistema operacional convidado sofre uma brecha, ou se um sistema operacional convidado sofrer uma falha, o software rodando em outros sistemas convidados não é afetado pela brecha/falha. A plataforma de hardware hospedeira também simplifica a tarefa do projetista de aplicativo visto que o software de aplicativo não precisa se preocupar com a arquitetura real dos dispositivos de computação nos quais o aplicativo será executado por fim.

[0090] O módulo de software de virtualização 506 pode ser logicamente situado entre o hardware hospedeiro e o software convidado. O software de virtualização pode rodar no hardware real (nativo) ou em cima de um sistema operacional (hospedado), e é tipicamente referido como um "hipervisor" ou monitor de máquina virtual (VMM). Em configurações nativas, o software de virtualização roda no hardware real no modo de privilegio mais alto disponível, e os sistemas operacionais convidados rodam com privilégios reduzidos de modo que o software de virtualização possa

interceptar e emular todas as ações do sistema operacional convidado que normalmente acessariam ou manipulariam os recursos de hardware. Nas configurações hospedadas, o software de virtualização roda em cima de um sistema operacional hospedeiro existente, e pode se basear no sistema operacional hospedeiro para fornecer acionadores de dispositivo e outros serviços de nível mais baixo. Em qualquer caso, cada um dos sistemas operacionais convidados (por exemplo, sistema operacional 504) se comunica com o módulo de software de virtualização 506 da mesma forma que se comunicariam com o hardware físico 508, visualizando a combinação do módulo de virtualização 506 e hardware 508 como uma única máquina virtual 510. Isso permite que cada sistema operacional convidado (por exemplo, sistema operacional 504) opere sob a ilusão de possuir acesso exclusivo aos processadores, periféricos, I/O, MMUs e memórias no hardware 508.

[0091] Como discutido acima com referência à figura 3, um sistema operacional pode ser responsável pela divisão da memória física através de múltiplos processos. Isso pode ser alcançado através de um processo de tradução de espaço de endereço de memória. Em um processo de tradução de espaço de endereço de memória o sistema operacional designa endereços virtuais para cada programa de aplicativo, e então aloca os endereços físicos com base nos endereços virtuais antes da execução do programa. No entanto, em sistemas que incluem um sistema operacional convidado rodando em cima de uma máquina virtual, os endereços de memória alocados pelo sistema operacional convidado não são os endereços físicos verdadeiros, mas endereços físicos intermediários. Em tais sistemas, a alocação real da memória física é geralmente realizada pelo hipervisor, que pode ser necessário para manter as relações

entre os endereços virtuais, os endereços físicos intermediários, e os endereços físicos.

[0092] A maior parte dos sistemas de processador suportam apenas um único estágio de processo de tradução e endereço de memória, e exigem que o hipervisor gerencie a relação entre os endereços virtuais, endereços físicos intermediários, e endereços físicos. Isso é geralmente alcançado pelo hipervisor mantendo suas próprias tabelas de tradução (chamadas de tabelas de tradução sombreadas), que podem ser derivadas pela interpretação de cada uma das tabelas de tradução do sistema operacional convidado. Em tais sistemas, o hipervisor garante que todas as mudanças nas tabelas de tradução do sistema operacional convidado sejam refletidas nas estruturas de sombra, além de garantir proteções e redirecionamento de falhas de acesso para o estágio adequado. Como discutido acima, essas operações aumentam a complexidade do hipervisor, e adicionam overheads significativos à execução, manutenção e/ou gerenciamento do hipervisor.

[0093] Diferentemente dos processadores de estágio único discutidos acima, alguns sistemas de processador (por exemplo, ARM v7-A) fornecem assistência de hardware para ambos os estágios de tradução de memória. Por exemplo, os processadores ARM podem incluir Extensões de Virtualização que permitem que o sistema operacional convidado traduza os endereços virtuais em endereços físicos intermediários em um primeiro estágio (isso e, traduções de primeiro estágio), e para o hardware traduzir os endereços físicos intermediários em endereços físicos em um segundo estágio (isso é, traduções de segundo estágio). Tais Extensões de Virtualização reduzem os overheads associados com a execução, manutenção e/ou gerenciamento de

hipervisor, e aperfeiçoam o desempenho do dispositivo de computação.

[0094] A figura 6 ilustra componentes lógicos ilustrativos e traduções de endereço associadas com a alocação da memória em dois estágios em um dispositivo de computação 600 implementando uma máquina virtual de sistema. Um sistema de gerenciamento de memória de um sistema operacional convidado 610 (por exemplo, HLOS) pode designar um espaço de endereço virtual 602, 604 para cada um dos programas de aplicativo ou processos (A_0, A_n). Por exemplo, os espaços de endereço virtual 602, 604 podem ser designados por um gerenciador de memória virtual (por exemplo, VMM OS Convidado). Cada programa de aplicativo ou processo (A_0, A_n) pode receber seu próprio espaço de endereço virtual 602, 604 e cada espaço de endereço virtual 602, 602 pode incluir um ou mais endereços virtuais VA_0 616, VA_n 618.

[0095] No exemplo ilustrado na figura 6, os endereços de memória são traduzidos em dois estágios. Em uma tradução de primeiro estágio 612, o gerenciador de memória virtual do sistema operacional convidado 610 (VMM OS Convidado) pode mapear os endereços virtuais VA_0 616, VA_n 618 em endereços físicos intermediários IPA_0 626, IPA_n 628 em um espaço de endereço físico intermediário 606. Em uma tradução de segundo estágio 614, as extensões de hipervisor e/ou virtualização podem mapear os endereços físicos intermediários IPA_0 626, IPA_n 628 em endereços físicos PA_0 636, PA_n 638 em um espaço de endereço físico 608. O primeiro estágio de tradução 612 pode ser realizado independentemente do segundo estágio de tradução 614, e nos sistemas existentes, os componentes que realizam o segundo estágio de tradução 614 não alocam os endereços físicos com base nas características da memória.

[0096] A figura 7 ilustra componentes lógicos e traduções de endereço ilustrativos associados com a alocação de memória em dois estágios em um dispositivo de computação 700 implementando uma máquina virtual de sistema enquanto o hipervisor é desativado. Em um aspecto, enquanto desativado, o hipervisor pode não engajar em traduções de segundo estágio 708 até que uma sessão sandbox seja iniciada. Por exemplo, sandboxing pode não ser necessário quando apenas um sistema operacional convidado (por exemplo, um HLOS) está sendo executado no dispositivo de computação. Portanto, em vários aspectos, o hipervisor pode ter um desempenho mais eficiente por não engajar nas traduções de segundo estágio quando sandboxing é determinado como desnecessário.

[0097] Em um aspecto, enquanto o hipervisor está desativado, em uma tradução de primeiro estágio 706, HLOS pode mapear os endereços virtuais no espaço de endereço virtual 710 em endereços físicos intermediários no espaço de endereço físico intermediário do HLOS 720 como discutido acima com referência à figura 6. Por exemplo, o HLOS pode traduzir/mapear os endereços virtuais VA_0 712 e VA_n 714 em endereços físicos intermediários IPA_0 722 e IPA_n 724, respectivamente. Em outro exemplo, o HLOS (ou uma MMU operando no HLOS) pode alocar blocos de memória virtual para uso pelos aplicativos A_0 a A_n pela realização da tradução de primeiro estágio 706 entre o espaço de endereço virtual 710 e o espaço de endereço físico intermediário 720.

[0098] Em um aspecto adicional, enquanto desativado, o hipervisor pode não realizar traduções a partir do espaço de endereço físico intermediário 720 para espaço de endereço físico 730 através de uma tradução de segundo estágio 708. Nesse aspecto, HLOS pode ultrapassar

as traduções de segundo estágio 708. Dessa forma, visto que HLOS pode ultrapassar as traduções de segundo estágio 708, HLOS pode alocar a memória diretamente a partir do espaço de endereço físico 730. Pela ativação de HLOS para ultrapassar as traduções de segundo estágio 708, o hipervisor garante que os endereços físicos intermediários sejam iguais aos endereços físicos. Dessa forma, em um exemplo, IPA_n 724 no espaço de endereço físico intermediário 720 é equivalente a PA_n 734 no espaço de endereço físico 730. De forma similar, IPA_0 722 no espaço de endereço físico intermediário 720 é igual a PA_0 732 no espaço de endereço físico 730.

[0099] A figura 8 ilustra componentes lógicos e traduções de endereço ilustrativos associados com a alocação de memória em dois estágios em um dispositivo de computação 800 durante uma sessão sandbox. Nos vários aspectos, o hipervisor pode ser ativado em resposta à detecção do início de uma sessão sandbox.

[0100] Em um aspecto, uma sessão sandbox pode ser uma situação na qual HLOS deve ser isolado do conteúdo protegido. O conteúdo protegido pode incluir um aplicativo seguro, um segundo sistema operacional rodando no dispositivo de computação, ou qualquer outra coisa que possa precisar ser processada ou armazenada separadamente. Por exemplo, um processador de sina digital (isso é, um "DSP") pode receber um sinal de vídeo seguro (isso é, conteúdo protegido) para processamento. Nesse exemplo, o sinal de vídeo seguro pode precisar ser processado separadamente do HLOS para manter a integridade e/ou segurança do sinal de vídeo. Em um aspecto adicional, em resposta à determinação de que o sinal de vídeo seguro precisa ser processado separadamente, um monitor de segurança, tal como ARM TrustZone® pode alertar o

hipervisor que uma sessão sandbox foi iniciada. Em resposta ao recebimento do alerta, o hipervisor pode ser ativado e pode retomar a implementação das traduções de segundo estágio a partir do espaço de endereço físico intermediário para o espaço de endereço físico.

[0101] Como ilustrado na figura 8, uma vez que uma sessão sandbox foi iniciada e o hipervisor está ativado, o hipervisor pode retomar as traduções de segundo estágio 808 e 846. Em um aspecto, o conteúdo protegido pode ser processado utilizando ambas uma tradução de primeiro estágio 844 e uma tradução de segundo estágio 846, similar a como HLOS aloca a memória como descrito acima com referência à figura 6. Por exemplo, um ambiente seguro (por exemplo, um DSP operando dentro de uma máquina virtual segura) pode receber um sinal de vídeo seguro através de uma conexão com uma rede segura como discutido com referência à figura 2. Nesse exemplo, o DSP pode alocar um ou mais blocos de 4 kb de memória (por exemplo, VA_{CP} 852) a partir de um espaço de endereço virtual 860 para um aplicativo de processamento de vídeo rodando no DSP para armazenamento de um sinal de vídeo seguro recebido. O DSP também pode manter um mapeamento do VA_{CP} 852 no espaço de endereço virtual 850 em um endereço físico intermediário IPA_{CP} 862 no espaço de endereço físico intermediário 860 pela realização de uma tradução de primeiro estágio 844.

[0102] Durante uma sessão de sandbox (isso é, durante as alocações de memória de ambiente seguro a partir do espaço de endereço físico intermediário 860), o HLOS pode realizar também alocações de memória. No entanto, visto que o hipervisor é ativado e retomou as traduções de segundo estágio 808, HLOS não precisa mais de uma capacidade irrestrita de alocar a memória diretamente a partir de todo o espaço de endereço físico 830.

[0103] Dessa forma, em um aspecto, o hipervisor pode restringir os endereços físicos no espaço de endereço físico 830 disponível par alocação pelo HLOS. Em outras palavras, HLOS ainda pode realizar alocações de memória diretamente para o espaço de endereço físico 830, mas o hipervisor pode limitar a capacidade do HLOS em acessar algumas partes do espaço de endereço físico 830. Por exemplo, HLOS pode alocar memórias virtuais VA_0 712 e VA_n 714, que mapeiam em IPA_0 722 e IPA_n 724 depois de uma tradução de primeiro estágio 706, respectivamente, no espaço de endereço físico intermediário 720. Adicionalmente, IPA_0 722 e IPA_n 724 ainda podem ser mapeados em PA_0 732 e PA_n 734 como ilustrado na figura 7, mas, ao passo que HLOS pode alocar memória a partir de todo o espaço de endereço físico 830 como descrito com referência à figura 7, enquanto o hipervisor está desativado (isso é, enquanto HLOS é capaz de ultrapassar as traduções de segundo estágio 808), HLOS pode ter acesso a um conjunto menor de endereços físicos enquanto o hipervisor é ativado.

[0104] Enquanto o hipervisor é ativado e está realizando as traduções de segundo estágio 808, 846, o hipervisor pode alocar memória ao componente sandboxed a partir de endereços físicos anteriormente disponíveis para HLOS. Por exemplo, o hipervisor pode mapear IPA_{CP} 862 em PA_{CP} 872, que agora não está mais disponível para HLOS. Dessa forma, enquanto ativado, o hipervisor pode "perfurar" os endereços físicos disponíveis para HLOS enquanto o hipervisor está desativado pela alocação da memória para, por exemplo, a entidade sandboxed. Na realização da tradução de segundo estágio 808 a partir do espaço de endereço físico intermediário do HLOS 720 para o espaço de endereço físico 830, o hipervisor pode obscurecer os

endereços físicos "perfurados" no espaço de endereço físico 830, impedindo, assim, que HLOS acesse os endereços físicos "perfurados". Dessa forma, por exemplo, depois que o hipervisor aloca PA_{CP} 872 para o componente sandboxed, o HLOS pode não mais ter acesso ao endereço físico.

[0105] Não permitindo mais que HLOS ultrapasse a tradução de estágio 2 808, o hipervisor pode novamente gerenciar os endereços físicos para os quais HLOS (e, dessa forma, o componente sandboxed) tem acesso finalmente. Dessa forma, o hipervisor pode instituir o sandboxing pela retomada, entre outras coisas, da tradução de estágio 2 (isso é, pelo gerenciamento direto do acesso à memória física) quando existem mais múltiplos aplicativos, processos, sistemas operacionais, etc. que devem ser segregados.

[0106] A figura 9 ilustra os componentes lógicos ilustrativos e traduções de endereço associados com a alocação de memória em dois estágios em um dispositivo de computação 900 durante uma sessão de sandbox e um processo de memória virtual compartilhado. Em vários aspectos, o hipervisor pode ser ativado em resposta à detecção do início de uma sessão de sandbox com uma entidade de compartilhamento (por exemplo, um DSP), e HLOS e a entidade de compartilhamento podem compartilhar alguns endereços físicos no espaço de endereço físico 930.

[0107] Em um aspecto descrito abaixo com referência às figuras 11 e 12, o dispositivo de computação pode iniciar uma sessão de memória virtual compartilhada, tal como entre um HLOS e um processador de sinal digital (DSP). Em um aspecto adicional, uma sessão de memória virtual compartilhada entre HLOS e DSP pode incluir a configuração de HLOS e DSP para compartilhar acesso a um conjunto de endereços físicos no espaço de endereço físico

930. Por exemplo, HLOS e DSP podem sofrer uma sessão de memória virtual compartilhada quando HLOS e DSP precisam compartilhar estruturas de dados, rotinas, etc. Pelo compartilhamento de acessos de memória direta, HLOS e DSP podem compartilhar de forma eficiente a informação sem precisar copiar e transmitir informação armazenada no espaço de endereço físico 930.

[0108] Como ilustrado na figura 9, HLOS e DSP podem ter memórias virtuais alocadas (isso é, VA_{SVM1} 914 e VA_{SVMn} 912) em espaços de endereço virtual respectivos 910, 950 para aplicativos, processos, etc. operando no HLOS e DSP, respectivamente. Por exemplo, os aplicativos operando no HLOS e um DSP podem compartilhar determinadas estruturas de dados, funções ou bibliotecas. HLOS e DSP podem realizar traduções de primeiro estágio 906, 944, respectivamente, para mapear os endereços virtuais VA_{SVM1} 914 a VA_{SVMn} 912 em IPA_{SVM1} 924 e IPA_{SVMn} 922 em cada um dentre o espaço de endereço físico intermediário da entidade de compartilhamento 960 e o espaço de endereço físico intermediário HLOS 920.

[0109] Em outro aspecto, visto que o hipervisor é ativado em resposta à inicialização da sessão sandbox, o hipervisor pode ativar as traduções de segundo estágio 908 que mapeiam os endereços físicos intermediários a partir do espaço de endereço físico intermediário HLOS 920 para o espaço de endereço físico 930. O hipervisor também pode ativar as traduções de segundo estágio 946 que mapeiam os endereços físicos intermediários no espaço de endereço físico intermediário da entidade de compartilhamento 960 em espaço de endereço físico 930.

[0110] Em um aspecto, como descrito com referência à figura 8, permitindo-se as traduções de segundo estágio 908 a partir do espaço de endereço físico

intermediário do HLOS 920 e do espaço de endereço físico 930, o hipervisor pode limitar os endereços físicos no espaço de endereço físico 930 ao qual HLOS tem acesso (isso é, removendo os mapeamentos para alguns endereços físicos de modo que HLOS não posas acessar esses endereços físicos no espaço de endereço físico 930). Como ilustrado na figura 9, o hipervisor pode manter os mapeamentos 940 de IPA_{HLOS} 926, IPA_{SVM1} 924 e IPA_{SVMn} 922 em endereços físicos PA_{HLOS} 936, PA_{SVM1} 934, PA_{SVMn} 932, respectivamente, de modo que os mapeamentos 940 do espaço de endereço físico intermediário do HLOS 920 para o espaço de endereço físico 930 garanta que os endereços físicos intermediários sejam iguais aos endereços físicos. De forma similar, o hipervisor pode manter, entre outros mapeamentos, os mapeamentos compartilhados 941 de IPA_{SVM1} 924 e IPA_{SVMn} 922 no espaço de endereço físico intermediário da entidade de compartilhamento 960 para PA_{SVM1} 934, e PA_{SVMn} 932, respectivamente, no espaço de endereço físico 930.

[0111] Em outro aspecto, o hipervisor pode implementar o sandboxing "parcial" da entidade de compartilhamento. Nesse aspecto, o hipervisor pode remover os mapeamentos do HLOS para endereços físicos alocados para a entidade de compartilhamento que não são compartilhados com HLOS (por exemplo, $PA_{non-SVM}$ 962). A entidade de compartilhamento e HLOS podem, cada um, manter os mapeamentos para memória no espaço de endereço físico que não são compartilhados. Por exemplo, DSP pode manter a informação associada com um $IPA_{non-SVM}$ 923 que pode ser, por exemplo, o núcleo do DSP. Em outro exemplo, HLOS pode manter a memória em IPA_{HLOS} 926 no espaço de endereço físico intermediário do HLOS 920 que é mapeado para PA_{HLOS} 936, que não é compartilhado com a entidade de compartilhamento.

[0112] No entanto, o hipervisor pode não remover os mapeamentos do HLOS para endereços físicos que são alocados para a entidade de compartilhamento, mas são compartilhados com HLOS (por exemplo, IPA_{SVM1} 924, e IPA_{SVMn} 922). Pela não remoção dos mapeamentos, o hipervisor pode permitir que o HLOS e a entidade de compartilhamento compartilhem a informação armazenada nesses endereços físicos, tal como apontadores para estruturas de dados, bibliotecas, rotinas, etc.

[0113] Dessa forma, pelo gerenciamento de mapeamentos para o espaço de endereço físico 930 que são removidos das traduções de segundo estágio 908 do espaço de endereço físico intermediário do HLOS 920, o hipervisor pode permitir que HLOS e a entidade de compartilhamento compartilhem informação armazenada em determinados endereços físicos e ainda pode garantir o controle de acesso dos endereços que não são compartilhados (por exemplo, $PA_{non-SVM}$ 962 e PA_{HLOS} 936).

[0114] A figura 10 ilustra um método de aspecto 1000 que pode ser implementado em um processador de dispositivo de computação (por exemplo, uma CPU) para ativar seletivamente um hipervisor durante uma sessão de sandbox. No bloco 1002, o processador do dispositivo de computação pode inicializar o hipervisor, monitor de segurança, e HLOS. Em um aspecto, o processador do dispositivo de computação pode inicializar o hipervisor, monitor de segurança, e HLOS pelo booting no hipervisor, monitor de segurança, e HLOS utilizando um fluxo de boot seguro Linaro ARMv8 e a tabela de concessão estilo Xen. Em outro aspecto, o monitor de segurança pode ser um ARM TrustZone®.

[0115] Em um aspecto adicional, o código de hipervisor e dados podem ser autenticados e/ou assinados

pelo monitor de segurança durante a inicialização. Durante a inicialização, o hipervisor também pode ser configurado de modo que seu código e dados sejam inacessíveis para processadores externos, tal como um processador de sinal digital (DSP) ou uma CPU que é incluída em um DSP. Em outro aspecto, a autenticação do hipervisor e/ou evitar que processadores externos acessem o código de hipervisor e dados enquanto o hipervisor é ativado podem garantir que sessões de sandbox futuras estejam seguras.

[0116] No bloco de determinação opcional 1004, o processador do dispositivo de computação pode determinar se existe uma sessão de computação heterogênea simultânea com memória virtual compartilhada. Em um aspecto, HLOS e uma entidade de compartilhamento (por exemplo, um DSP) podem compartilhar estruturas de dados complexas contendo apontador. Se o processador de dispositivo de computação determinar que uma sessão de computação heterogênea simultânea com a situação de memória virtual compartilhada está presente (isso é, o bloco de determinação 1004 = "Sim"), o processador pode configurar a sessão de computação heterogênea simultânea com memória virtual compartilhada no bloco 1006. Em um aspecto, HLOS e um DSP, por exemplo, podem ser configurados para compartilhar a mesma tabela de página de primeiro estágio. A configuração da sessão de computação heterogênea simultânea com memória virtual compartilhada é discutida em detalhes abaixo com referência às figuras 11 e 12. O processador do dispositivo de computação pode continuar a operar no bloco 1008. Se o processador do dispositivo de computação determinar que não existe sessão de computação heterogênea simultânea com memória virtual compartilhada (isso é, o boco de determinação opcional 1004 = "Não"), o processador pode continuar a operação no bloco 1008.

[0117] No bloco 1108, o processador do dispositivo de computação pode desativar o hipervisor. Em um aspecto, a condição padrão do hipervisor pode ser desativada. Em outro aspecto, a desativação do hipervisor pode desativar as traduções de segundo estágio dos espaços de endereço físico intermediários para o espaço de endereço de memória física. Outras consequências da desativação do hipervisor podem incluir a cessação da restrição dos acessos HLOS às interrupções de hardware, temporizadores de hardware, e entrada/saída. A desativação do hipervisor é adicionalmente discutida abaixo com referência à figura 13.

[0118] No bloco 1009, o processador do dispositivo de computação pode monitorar um sinal recebido no hipervisor para começar uma sessão sandbox. Em um aspecto, o monitor de segurança (por exemplo, um ARM TrustZone®) pode receber ou detectar conteúdo protegido e enviar um sinal de acordar para o hipervisor para iniciar uma sessão de sandbox. Por exemplo, um DSP operando dentro de um ambiente virtual seguro pode receber um sinal de vídeo seguro para um processamento seguro. Nesse exemplo, o DSP pode ser configurado para armazenar sinal de vídeo em uma parte do espaço de endereço de memória física inacessível ao HLOS, por exemplo.

[0119] No bloco de determinação 1010, o processador de dispositivo de computação pode determinar se o hipervisor recebeu um sinal para iniciar uma sessão sandbox. Se o processador de dispositivo de computação determinar que o hipervisor não recebeu um sinal para iniciar uma sessão de sandbox (isso é, o bloco de determinação 1010 = "Não"), o processador pode continuar a operar no bloco 1009. Em um aspecto, o processador do dispositivo de computação pode continuar a monitorar um sinal para o hipervisor para iniciar uma sessão de sandbox.

[0120] Se o processador do dispositivo de computação determinar que o hipervisor recebeu um sinal para iniciar uma sessão de sandbox (isso é, o bloco de determinação 1010 = "Sim"), o processador pode ativar o hipervisor no bloco 1012. Em um aspecto, a ativação do hipervisor pode incluir a retomada das traduções de segundo estágio. A ativação do hipervisor é discutida em maiores detalhes abaixo com referência às figuras 14 e 15.

[0121] O hipervisor pode então implementar o controle de acesso no bloco 1014. Em um aspecto, o hipervisor pode implementar o controle de acesso pela realização das traduções de segundo estágio a partir dos endereços físicos intermediários em endereços físicos. Em um aspecto adicional, o hipervisor pode implementar adicionalmente o controle de acesso pela retomada da restrição de acessos ao I/O, interrupções de hardware, e temporizadores de hardware. O processo de implementação de controle de acesso é descrito em maiores detalhes abaixo com referência às figuras 16A e 16B.

[0122] No bloco de determinação 1016, o hipervisor pode determinar se a sessão de sandbox terminou. Por exemplo, uma sessão de sandbox pode ter terminado quando não houver mais conteúdo que deva ser garantido ou isolado do HLOS ou outros processos, aplicativos ou componentes. Por exemplo, no exemplo fornecido abaixo, a sessão de sandbox iniciada quando o DSP recebeu um sinal de vídeo seguro pode terminar depois que o DSP processou o sinal de vídeo seguro e não precisa mais armazenar os armazenadores de vídeo de sinal de vídeo na memória física. Em outro aspecto, o monitor de segurança ou outro componente no ambiente virtual seguro pode sinalizar o hipervisor que a sessão de sandbox terminou.

[0123] Se a sessão de sandbox não tiver terminado (isso é, o bloco de determinação 1016 = "Não"), o hipervisor pode continuar a realizar as operações no bloco 1014. Do contrário (isso é, o bloco de determinação 1016 = "Sim"), o hipervisor pode desligar a sessão de sandbox no bloco 1018. Em um aspecto, o hipervisor pode retornar o HLOS e vários outros componentes do dispositivo de computação para um estado "padrão" ou configuração como resultado da realização do procedimento de desligamento de sessão sandbox. Os desligamentos de sessão de sandboxing são descritos em maiores detalhes abaixo com referência à figura 17. O dispositivo de computação pode continuar a realizar as operações no bloco 1008.

[0124] A figura 11 ilustra um sinal de aspecto e um fluxo de chamada dentre vários componentes de um dispositivo de computação para iniciar uma sessão de computação heterogênea simultânea. Em um aspecto, na operação 1112, HLOS 1102 pode criar uma tabela de tradução de primeiro estágio. O HLOS 1102 também pode alocar um identificador específico de aplicativo (ASID) no identificador de máquina virtual HLOS (isso é, "HLOS_VMID") na operação 1114. Em outro aspecto, o hipervisor 1104 pode enviar um sinal 1116 para uma unidade de gerenciamento de memória de sistema (SMMU) de segundo estágio do processador de sinal digital (DSP) 1106 para criar uma tabela de tradução de segundo estágio utilizando HLOS_VMID do HLOS. Em um aspecto, a SMMU de segundo estágio do DSP pode utilizar a tabela de tradução de segundo estágio para realizar as traduções de segundo estágio do espaço de endereço físico intermediário do HLOS para o espaço de endereço físico.

[0125] Em um aspecto, HLOS pode enviar um sinal 1118 para o hipervisor do DSP 1108, solicitando a criação

de um processo de memória virtual compartilhada (SVM) utilizando o HLOS_VMID do HLOS, o identificador específico de aplicativo selecionado, e a tabela de tradução de primeiro estágio. O hipervisor do DSP 1108 pode iniciar o processo de memória virtual compartilhada do DSP 1110 com um sinal 1120 que programa a tabela de tradução de primeiro estágio para a unidade de gerenciamento de memória do DSP (MMU), e então inicia o processo de memória virtual compartilhada do DSP 1110.

[0126] A figura 12 ilustra um método de aspecto 1006a que pode ser implementado em um dispositivo de computação para iniciar um processo de memória virtual compartilhado entre um HLOS e um DSP. O processador de dispositivo de computação pode iniciar o método 1006a quando o processador determina que existe uma sessão de computação heterogênea simultânea com memória virtual compartilhada (isso é, bloco de determinação 1004 = "Sim"). No bloco 1204, o dispositivo de computação pode configurar o HLOS para criar uma tabela de tradução de primeiro estágio. Em um aspecto, HLOS pode utilizar a tabela de tradução de primeiro estágio para mapear o endereço virtual para endereços físicos intermediários. O dispositivo de computação também pode configurar o HLOS para alocar um identificador específico de aplicativo no identificador de máquina virtual do HLOS (isso é, HLOS_VMID) no bloco 1206.

[0127] No bloco 1208, o dispositivo de computação pode configurar o hipervisor para enviar uma configuração de segundo estágio com base em HLOS_VMID para a SMMU de segundo estágio do DSP. Em um aspecto, a SMMU de segundo estágio do DSP pode criar uma tabela de tradução de segundo estágio para HLOS com base em HLOS_VMID. A SMMU (ou hipervisor) pode utilizar a tabela de tradução de segundo estágio para realizar as traduções de segundo estágio a

partir do espaço de endereço físico intermediário do HLOS para o espaço de endereço de memória física do dispositivo de computação.

[0128] No bloco 1210, o dispositivo de computação pode configurar o HLOS para solicitar que o hipervisor do DSP crie um processo de memória virtual compartilhado utilizando HLOS_VMID, o ASID selecionado, e a tabela de tradução de primeiro estágio do HLOS.

[0129] O dispositivo de computação também pode configurar o hipervisor do DSP para programar uma unidade de gerenciamento de memória de primeiro estágio (MMU) e iniciar o processo de memória virtual compartilhado no bloco 1212. Em um aspecto, a MMU de primeiro estágio do DSP pode iniciar uma tabela de tradução de primeiro estágio que é igual à tabela de traduções de primeiro estágio do HLOS. Dessa forma, nesse aspecto, o HLOS e o DSP podem compartilhar a memória virtual visto que compartilham a mesma tabela de transmissão de primeiro estágio.

[0130] Com os processos de memória virtual compartilhada completados no bloco 1212, o processador do dispositivo de computação pode desativar o hipervisor no bloco 1008 como descrito acima com referência à figura 10 quando não existe mais necessidade de uma sessão de sandbox.

[0131] A figura 13 ilustra um método de aspecto 1008a para desativar o hipervisor no dispositivo de computação.

[0132] No bloco 1304, o hipervisor pode configurar todos os bancos de contexto do primeiro estágio da SMMU para ultrapassar as traduções de segundo estágio. O hipervisor pode desligar as traduções de segundo estágio do HLOS no bloco 1306.

[0133] Em alguns aspectos, o hipervisor pode suspender várias outras atividades quando desativado. Por exemplo, o hipervisor pode suspender opcionalmente a restrição aos acessos I/O no bloco opcional 1308. O hipervisor também pode suspender a restrição de acessos à interrupção de hardware no bloco opcional 1310. Adicionalmente, no bloco opcional 1312, o hipervisor pode suspender a restrição aos acessos de temporizador de hardware.

[0134] O hipervisor pode determinar se o hipervisor recebeu um sinal para iniciar uma sessão de sandbox no bloco de determinação 1010 como descrito acima com referência à figura 10.

[0135] Em alguns aspectos, as várias funções de hipervisor (por exemplo, controle de acesso, memória de sandboxing, etc.) podem ser desativadas através de um limite de circuito integrado e/ou limite de chip. Em um aspecto, em uma combinação de conjunto de chip tipo fusão que inclui conjuntos de chip discretos (por exemplo, um chip de modem e um chip de processador de aplicativo), um conjunto de chips principal (isso é, um hipervisor principal) pode suspender a memória de sandboxing ou outros controles de acesso em outros conjuntos de chip quando a funcionalidade do hipervisor foi desativada. Por exemplo, o hipervisor principal em um conjunto de chips de processador de aplicativo pode suspender as traduções de segundo estágio dos endereços físicos intermediários para endereços físicos em um conjunto de chip de modem ou DSP. Dessa forma, quando as funções de hipervisor foram desativadas, o hipervisor principal pode suspender essas funções em vários chips discretos.

[0136] A figura 14 ilustra fluxos de chamada 1400 entre múltiplos componentes operando no dispositivo de

computação enquanto configura uma sessão de sandbox para um sinal de vídeo de conteúdo protegido. Em vários aspectos, páginas para armazenador de vídeo podem ser páginas de 4 kb e podem fragmentar as tabelas de página de tradução de segundo estágio do HLOS.

[0137] Em um aspecto, um sinal 1402 pode ser enviado para vários componentes, tal como uma estrutura de trabalho MM Android 1450, um componente OpenMax (OMX) 1452, um acionador de vídeo V4L2 1454, um alocador de página de núcleo 1456, um hipervisor 1458, uma SMMU 1460, e um núcleo 1462, para inicializar o firmware com autenticação de segurança. Em um aspecto, a autenticação de segurança pode ser um ARM TrustZone®. Em outro aspecto, se o vídeo estiver em um DSP, o dispositivo de computação pode carregar o aplicativo de vídeo DSP e codec.

[0138] Em um aspecto, uma estrutura de trabalho MM Android 1450 pode enviar um sinal 1404 para o componente OMX para inicialização no futuro. O componente OMX 1452 pode enviar um sinal 1406 para configurar um codec no acionador de vídeo V4L2 1454. O componente OMX 1452 também pode enviar um sinal de pesquisa de dimensionamento de armazenador 1408 para o acionador de vídeo v4L2 1454. A estrutura de trabalho MM Android 1450 também pode enviar um sinal de inatividade 1410 para o componente OMX 1452.

[0139] Em outro aspecto, o componente OMX 1452 pode enviar um sinal de entrada LIGADO sequencial 1412 para o acionador de vídeo V4L2 1454. O acionador de vídeo V4L2 1454 pode enviar um sinal de inicialização de sessão de interface de firmware hospedeiro (HFI) 1414 para o núcleo 1462. O núcleo 1462 pode responder ao sinal de inicialização de sessão HFI 1414 enviando um sinal de "sessão HFI realizada" 1415 para o acionador de vídeo V4L2 1454.

[0140] A estrutura de trabalho MM Android pode enviar um sinal de alocação de memória 1418 (isso é, "loctl ION_IOC_ALLOC") para o componente OMX 1452. Em outro aspecto, a estrutura de trabalho MM Android pode enviar o sinal de alocação de memória 1418 para gerenciar o conjunto de páginas. O acionador de vídeo OMX V42L 1454 pode enviar um sinal "Ion_alloc (ION cp heap)" 1420 para o alocador de página núcleo 1456.

[0141] O alocador de página núcleo 1456 pode então enviar um sinal VMM_CALL 1422 para o hipervisor 1458 para remover os mapeamentos de segundo estágio do processador. Em um aspecto, esse sinal 1422 pode notificar o hipervisor que uma sessão sandbox foi iniciada e que determinadas localizações de memória física devem ser removidas desse endereço físico acessível a HLOS. Em um aspecto adicional, o hipervisor 1458 pode enviar um sinal de mapeamento de tradução de segundo estágio 1424 para a SMMU 1460. Em um aspecto, a SMMU pode implementar essas traduções de segundo estágio (isso é, SMMU pode retomar as traduções de segundo estágio). Em um aspecto adicional, a SMMU pode manter os mapeamentos do HLOS para endereços físicos acessíveis (isso é, mapeamentos de endereços físicos que não foram ocultados do HLOS). Em um aspecto adicional, se o vídeo estiver em um DSP, o alocador de página núcleo 1456 pode sinalizar adicionalmente o hipervisor 1458 para mapear as páginas para um mapeamento de tradução de segundo estágio do DSP.

[0142] Em outro aspecto, o acionador de vídeo V4L2 1454 pode enviar armazenadores de conjunto de interface de firmware hospedeiros 1426 para o núcleo 1462 e pode enviar um sinal de inatividade 1428 indicando para a estrutura de trabalho MM Android que todos os armazenadores estão prontos. A estrutura de trabalho MM Android 1450 pode

então sinalizar 1430 para que o componente OMX transite para execução. A estrutura de trabalho MM Android 1450 também pode enviar para o componente OMX 1452 um sinal 1432 para enfileirar o primeiro armazenador com um cabeçalho. O componente OMX 1452 também pode enviar um sinal 1434 para o acionador de vídeo V4L2 1454 para enfileirar o primeiro armazenador com um cabeçalho.

[0143] O acionador de vídeo V4L2 1454 pode enviar para o alocador de página de núcleo 1456 um sinal 1436 que mapeia o primeiro armazenador para o Núcleo 1462. O alocador de página de núcleo 1456 pode sinalizar 1438 para a SMMU 1460 para mapear o primeiro armazenador para os bancos de contexto de entrada de tradução de primeiro estágio. O acionador de vídeo V4L2 1454 também pode enviar um sinal 1440 enfileirando um primeiro armazenador com um cabeçalho para o Núcleo 1462.

[0144] A figura 15 ilustra um método de aspecto 1012a que pode ser implementado em um processador do dispositivo de computação para permitir que o hipervisor (isso é, depois que o hipervisor é desativado no bloco 1010 descrito acima com referência à figura 10). No bloco 1504, o dispositivo de computação pode ativar a MMU de segundo estágio PL0 e PL1 (isso é, o nível de privilegio 1 e o nível de privilegio 2) através da configuração de HCR.VM para "1". No bloco 1506, o dispositivo de computação pode configurar as solicitações de interrupção ("IRQ") para que sejam realizadas no modo de hipervisor. Em um aspecto, o dispositivo de computação pode realizar isso pela configuração de SCR.IMO para "1".

[0145] O dispositivo de computação também pode chamar os acionadores de SMMU no bloco 1508 para colocar todos os bancos de contexto SMMU ativos em um estado no qual as traduções de primeiro estágio são aninhadas com as

traduções de segundo estágio. O dispositivo de computação pode configurar esse estado pela configuração do elemento SMMU_CBARn.type para "0b11".

[0146] Em um aspecto, o hipervisor pode ser ativado no bloco 1012 como descrito acima com referência à figura 10 uma vez que essas etapas sejam realizadas. Depois de ter sido ativado no bloco 1012, o hipervisor pode, opcionalmente, iniciar as comunicações interprocessador (IPC) com um DSP no bloco opcional 1510. O hipervisor também pode manusear, de forma opcional, as falhas SMMU no bloco opcional 1512.

[0147] Em alguns aspectos, o hipervisor pode retomar várias outras atividades quando ativado. Por exemplo, o hipervisor pode retomar a restrição aos acessos I/O no bloco opcional 1514. O hipervisor também pode retomar a restrição de acesso de interrupção de hardware no bloco 1516. No bloco 1518, o hipervisor pode retomar adicionalmente a restrição aos acessos de temporizador de hardware.

[0148] O processador do dispositivo de computação pode então implementar o controle ao acesso no bloco 1014 como descrito acima com referência à figura 10.

[0149] Em alguns aspectos, as várias funções do hipervisor (por exemplo, controle ao acesso, memória de sandboxing, etc.) podem ser ativadas através de um limite de circuito integrado e/ou limite de chip. Como discutido acima com relação à figura 13, em um aspecto, um conjunto de chip principal (isso é, um hipervisor principal) pode controlar a memória de sandboxing em outros conjuntos de chip através, por exemplo, de uma interface expressa de interconexão de componente periférico. Por exemplo, o hipervisor principal em um conjunto de chips de processador de aplicativo pode controlar as traduções dos endereços

físicos intermediários em endereços físicos em um modem ou conjunto de chip DSP. Dessa forma, quando as funções do hipervisor são ativadas, o hipervisor principal pode realizar essas funções através de vários chips discretos.

[0150] As figuras 16A e 16B ilustram métodos de aspecto que podem ser implementados em um processador do dispositivo de computação para realização das traduções de segundo estágio durante as sessões de sandbox. Em vários aspectos, o hipervisor pode gerenciar as localizações no espaço de endereço de memória física no qual vários componentes (por exemplo, HLOS ou um DSP) podem acessar.

[0151] A figura 16A ilustra um método de aspecto 1014a para a memória de alocação de hipervisor durante uma sessão de sandbox. Quando o hipervisor é ativado no bloco 1012 como descrito acima com referência à figura 10.

[0152] O hipervisor pode determinar no bloco de determinação opcional 1604 se uma situação de memória virtual compartilhada existe no momento. Em um aspecto, uma situação de memória virtual compartilhada pode existir quando, por exemplo, o HLOS está compartilhando uma memória virtual com outro componente. Se houver uma situação de memória virtual compartilhada (isso é, o bloco de determinação 1604 = "Sim"), o hipervisor pode executar o método 1014b descrito abaixo com referência à figura 16B. Do contrário (isso é, o bloco de determinação 1604 = "Não"), o hipervisor pode monitorar uma tentativa de alocar memória no bloco 1608. O hipervisor pode determinar se o HLOS está tentando alocar memória no bloco de determinação 1610. Em um aspecto, HLOS pode tentar alocar memória para aplicativos ou processos atualmente operando no HLOS. Por exemplo, o HLOS pode alocar memória para um aplicativo pela criação de um espaço de endereço virtual acessado por esse aplicativo. Se o processador determinar que HLOS está

tentando alocar memória (isso é, bloco de determinação 1610 = "Sim"), o hipervisor pode fornecer endereços físicos para o HLOS a partir do espaço de endereço de memória física que é acessível ao HLOS no bloco 1612. Em um aspecto, HLOS pode não ter acesso arbitrário ao espaço de endereço físico visto que alguns endereços físicos foram removidos dos mapeamentos de tradução de segundo estágio do HLOS e podem ser alocados para conteúdo protegido. Por exemplo, HLOS pode não ter um mapeamento de um endereço físico que está alocado a um DSP para armazenamento de armazenadores de vídeo de 4 kb. O hipervisor pode então determinar se a sessão de sandbox está encerrada no bloco de determinação 1016 descrito acima com referência à figura 10.

[0153] Se HLOS não estiver tentando alocar memória (isso é, o bloco de determinação 1610 = "Não"), o hipervisor pode determinar se o componente sandboxed está tentando alocar memória no bloco de determinação 1611. Por exemplo, o hipervisor pode determinar se um DSP processando um sinal de vídeo seguro está tentando armazenar armazenadores de vídeo de 4 kb na memória física. Se o componente sandboxed não estiver tentando alocar memória (isso é, o bloco de determinação 1611 = "Não"), o hipervisor pode determinar se a sessão de sandbox foi encerrada no bloco de determinação 1016 descrito acima com referência à figura 10.

[0154] Se o hipervisor determinar que o componente sandboxed está tentando alocar memória (isso é, bloco de determinação 1611 = "Sim"), o hipervisor pode remover os endereços físicos que serão fornecidos para o componente sandboxed a partir dos endereços físicos no espaço de endereço físico que são acessíveis ao HLOS no bloco 1614. O hipervisor também pode fornecer endereços físicos para o componente sandboxed a partir dos endereços

físicos disponíveis no espaço de endereço físico no bloco 1616. Em um aspecto, os endereços físicos disponíveis podem ser endereços físicos no espaço de endereço físico que não foram alocados para o HLOS. Em outras palavras, os endereços físicos disponíveis estão "livres" de endereços de memória. Em um aspecto, uma vez que o hipervisor aloca memória no espaço de endereço físico para uso pelo componente sandboxed, o HLOS pode não ter mais acesso a essa memória física durante a sessão de sandbox. Por exemplo, uma vez que o armazenador de vídeo de 4 kb para um vídeo seguro é armazenado em um endereço físico em particular, o HLOS pode não ter mais um mapeamento para esse endereço físico (isso é, HLOS pode não ser mais capaz de "ver" esses endereços físicos para alocar). O hipervisor pode então determinar se a sessão de sandbox foi encerrada no bloco de determinação 1016 descrito acima com referência à figura 10.

[0155] A figura 16B ilustra um método de aspecto 1014b que pode ser implementado em um hipervisor para alocação da memória durante uma sessão de sandbox enquanto HLOS está compartilhando memória virtual com outro componente (isso é, quando o bloco de determinação 1604 = "Sim"). No bloco de determinação 1624, o hipervisor pode determinar se HLOS está tentando alocar os endereços físicos. Por exemplo, o HLOS pode estar tentando alocar determinados endereços físicos no espaço de endereço físico para uso pelos aplicativos rodando no HLOS. Se o hipervisor determinar que o HLOS está tentando alocar endereços físicos (isso é, bloco de determinação 1624 = "Sim"), o hipervisor pode fornecer endereços físicos para o HLOS a partir de endereços físicos no espaço de endereço de memória física que são acessíveis ao HLOS no bloco 1612. No aspecto descrito acima com referência à figura 16A, os

endereços físicos podem estar acessíveis ao HLOS quando, por exemplo, o hipervisor permite que o HLOS aloque esses endereços físicos. Em outras palavras, os endereços físicos acessíveis ao HLOS pode não ser alocados ao componente sandboxed e ocultados do HLOS. Depois da alocação dos endereços físicos para o HLOS, o hipervisor pode determinar se a sessão de sandbox está encerrada no bloco de determinação 1016 descrito acima com referência à figura 10.

[0156] Se o hipervisor determinar que o HLOS não está tentando alocar endereços físicos (isso é, o bloco de determinação 1624 = "Não"), o hipervisor pode determinar se o componente sandboxed está tentando alocar endereços físicos no bloco de determinação 1626. Por exemplo, um DSP operando dentro do componente sandboxed pode tentar acessar determinado endereço físico para armazenar armazenadores de vídeo de 4 kb. Se o hipervisor determinar que o componente sandboxed está tentando alocar os endereços físicos (isso é, o bloco de determinação 1626 = "sim"), o hipervisor pode remover os endereços físicos que serão fornecidos para o componente sandboxed dos PAs no espaço de endereço de memória física acessível ao HLOS no bloco 1628. Em um aspecto, os endereços físicos alocados para o componente sandboxed podem ser ocultados do HLOS. Em outras palavras, o hipervisor pode remover os mapeamentos de segundo estágio do HLOS para os endereços físicos alocados para o componente sandboxed. O hipervisor também pode fornecer endereços físicos para o componente sandboxed a partir de endereços físicos disponíveis no espaço de endereço físico no bloco 1632. Em um aspecto, os endereços físicos disponíveis podem incluir os endereços físicos para os quais o hipervisor não separou para uso pelo HLOS (isso é, endereços físicos "livres" no espaço de endereço físico).

Depois da alocação de endereços físicos para o componente sandboxed, o hipervisor pode determinar se a sessão de sandbox terminou no bloco de determinação 1016 descrito acima com referência à figura 10.

[0157] Se o hipervisor determinar que o componente sandboxed não está tentando alocar endereços físicos (isso é, o bloco de determinação 1626 = "Não"), o hipervisor pode determinar se a entidade de compartilhamento está tentando alocar endereços físicos no bloco de determinação 1630. Em um aspecto, a entidade de compartilhamento pode ser um componente operando no dispositivo de computação que está compartilhando a memória virtual com o HLOS. Em outro aspecto, o HLOS e outra entidade podem compartilhar memória virtual pelo compartilhamento de apontadores para endereços físicos. Em outras palavras, HLOS e a entidade de compartilhamento podem ser capazes de acessar ou alocar os mesmos endereços físicos no espaço de endereço de memória física.

[0158] Se o hipervisor determinar que a entidade de compartilhamento não está tentando alocar endereços físicos (isso é, bloco de determinação 1630 = "Não"), o hipervisor pode determinar se a sessão de sandbox terminou no bloco de determinação 1016 descrito acima com referência à figura 10. Do contrário (isso é, bloco de determinação 1630 = "Sim"), o hipervisor pode determinar se a entidade de compartilhamento está tentando alocar endereços físicos compartilhados no bloco de determinação 1632. Em outras palavras, o hipervisor pode determinar se a entidade de compartilhamento está tentando utilizar, mudar, acessar, alocar ou de outra forma ler ou escrever em um endereço físico compartilhado com HLOS.

[0159] Se o hipervisor determinar que a entidade de compartilhamento está tentando alocar endereços físicos

compartilhados (isso é, bloco de determinação 1632 = "Sim"), o hipervisor pode fornecer endereços físicos compartilhados para a entidade de compartilhamento no bloco 1636. Em um aspecto, HLOS também pode acessar e alocar os endereços físicos compartilhados. Em outras palavras, o hipervisor pode não ocultar os endereços físicos compartilhados alocados para a entidade de compartilhamento a partir do HLOS. O hipervisor pode determinar se a sessão de sandbox terminou no bloco de determinação 1016 descrito acima com referência à figura 10.

[0161] A figura 17 ilustra um método de aspecto 1018a que pode ser implementado em um hipervisor para realizar um desligamento de sessão. Quando o hipervisor determina que a sessão de sandbox terminou (isso é, o bloco de determinação 1016 = "sim"), o hipervisor pode liberar todos os armazenadores do componente sandboxed no bloco 1704. Por exemplo, o hipervisor pode liberar os armazenadores de vídeo de 4 kb armazenados em vários endereços físicos no espaço de endereço de memória física. Em um aspecto, pela liberação desses armazenadores, o hipervisor pode preparar esses endereços físicos para serem acessíveis ao HLOS.

[0162] No bloco 1706, o hipervisor pode restaurar as tabelas de página de tradução de segundo estágio para remover todas as fragmentações. Em um aspecto, o hipervisor pode restaurar os endereços físicos no espaço de endereço físico que podem ser perfurados pelas alocações de memória para o componente sandboxed. Em outro aspecto, o hipervisor pode adicionar mapeamentos de segundo estágio que podem ativar o HLOS para acessar endereços físicos que o hipervisor ocultou depois de alocar esses endereços físicos para o componente sandboxed. O hipervisor também pode ser desativado no bloco 1008 como descrito acima com referência

à figura 10. Dessa forma, em um aspecto, depois da realização do procedimento de desligamento de sessão, o hipervisor pode colocar o HLOS de volta em uma posição na qual pode alocar a memória diretamente a partir de todo o espaço de endereço de memória física, e então ser desativado.

[0163] Dispositivos de computação típicos 1800 adequados para uso com os vários aspectos terão em comum os componentes ilustrados na figura 18. Por exemplo, um dispositivo de computação típico 1800 pode incluir um processador 1802 acoplado à memória interna 1801, um monitor 1803, e a um alto falante 1864. Adicionalmente, o dispositivo de computação pode ter uma antena 1804 para envio e recebimento de radiação eletromagnética acoplada ao processador 1802. Em alguns aspectos, o dispositivo de computação 1800 pode incluir um ou mais processadores de finalidade especial ou geral 1805, 1824, que podem incluir sistemas em chips. Os dispositivos de computação também incluem tipicamente um teclado ou teclado em miniatura (não ilustrado) e botões de seleção de menu 1808a, 1808b para o recebimento de registros de usuário. Os dispositivos de computação também podem incluir um botão de energia 1834 para ligar e desligar os dispositivos de computação.

[0164] Outras formas de dispositivos de computação, tal como computador laptop 1900 ilustrado na figura 19, também podem implementar e se beneficiar de vários aspectos. Os dispositivos de computação como um computador laptop 1900 incluem tipicamente um processador 1902 acoplado à memória interna 1901 e uma memória não volátil de grande capacidade, tal como um acionador de disco 1905 ou memória flash, e um monitor 1909. Os dispositivos de computação também podem incluir um teclado

1908 e botões de seleção 1907 para receber registros de usuário.

[0165] Os processadores 1802.1805, 1824, 1902 utilizados nos dispositivos de computação implementando os vários aspectos podem ser qualquer microprocessador programável, microcomputador ou chip ou chips de múltiplos processadores que podem ser configurados pelas instruções de software executáveis por processador (aplicativos) para realizar uma variedade de funções, incluindo as funções dos vários aspectos descritos aqui. Tipicamente aplicativos de software e instruções executáveis por processador podem ser armazenados na memória interna 1801, 1901 antes de serem acessadas e carregadas nos processadores 1802, 1805, 1824, 1902. Em alguns dispositivos de computação, os processadores 1802, 1805, 1824, 1902 podem incluir memória interna suficiente para armazenar as instruções de software de aplicativo.

[0166] Em alguns dispositivos de computação, a memória segura pode estar em um chip de memória separado acoplado ao processador 1802, 1805, 1824, 1902. Em muitos dispositivos de computação, a memória interna 1801, 1901 pode ser uma memória volátil ou não volátil, tal como a memória flash, ou uma mistura de ambas. A memória pode incluir qualquer número de tipos diferentes de tecnologias de memória, incluindo memória de mudança de fase (PCM), memória de acesso randômico dinâmica (DRAM), memória de acesso randômica estativa (SRAM), memória de acesos randômico não volátil (NVRAM), memória de acesso randômico pseudostática (PSRAM), memória de acesso randômico dinâmica, sincronizada, de taxa de dados dupla (SDRAM DDR) e outras tecnologias de memória de acesso randômico (RAM) e memória de leitura apenas (ROM) conhecidas da técnica. Para fins dessa descrição, uma referência geral à memória se

refere a toda a memória acessível pelos processadores 1802, 1805, 1824, 1902 incluindo a memória interna, a memória removível conectada ao dispositivo de computação e memória dentro dos processadores.

[0167] As descrições de método acima e os fluxogramas de processo são fornecidos meramente como exemplos ilustrativos e não devem exigir ou implicar que as etapas de vários aspectos devam ser realizadas na ordem apresentada. Como será apreciado pelos versados na técnica a ordem das etapas nos aspectos acima pode ser realizada em qualquer ordem. As palavras tal como "doravante", "então", "a seguir", etc. não devem limitar a ordem das etapas; essas palavras são simplesmente utilizadas para orientar o leitor através da descrição dos métodos. Adicionalmente, qualquer referência aos elementos de reivindicação no singular, por exemplo, utilizando os artigos "um", "uma", ou "o", "a" não deve ser considerada como limitada ao elemento no singular.

[0168] Os vários blocos lógicos, circuitos e etapas de algoritmo ilustrativos descritos com relação aos aspectos descritos aqui podem ser implementados como hardware eletrônico, software de computador, ou combinações de ambos. Para se ilustrar claramente essa capacidade de intercambio de hardware e software, vários componentes ilustrativos, blocos, módulos, circuitos e etapas foram descritos acima geralmente em termos de sua funcionalidade. Se tal funcionalidade é implementada como hardware ou software depende da aplicação em particular e das restrições de desenho impostas ao sistema como um todo. Os versados na técnica podem implementar a funcionalidade descrita de várias formas para cada aplicativo em particular, mas tais decisões de implementação não devem

ser interpretadas como responsáveis pelo distanciamento do escopo da presente invenção.

[0169] O hardware utilizado para implementar as várias lógicas ilustrativas, blocos lógicos, módulos e circuitos descritos com relação aos aspectos descritos aqui podem ser implementados ou realizados com um processador de finalidade geral, um processador de sinal digital (DSP), um DSP dentro de um chip receptor de difusão de multimídia, um circuito integrado específico de aplicativo (ASIC), um conjunto de porta programável em campo (FPGA) ou outro dispositivo lógico programável, porta discreta ou lógica de transistor, componentes de hardware discretos, ou qualquer combinação dos mesmos projetada para realizar as funções descritas aqui. Um processador de finalidade geral pode ser um microprocessador, mas, na alternativa, o processador pode ser qualquer processador convencional, controlador, micro controlador ou máquina de estado. Um processador também pode ser implementado como uma combinação de dispositivos de computação, por exemplo, uma combinação de um DSP e um microprocessador, uma pluralidade de microprocessadores, um ou mais microprocessadores em conjunto com um núcleo DSP, ou qualquer outra configuração dessas. Alternativamente, algumas etapas ou métodos podem ser realizados pelo conjunto de circuitos que é específico de uma determinada função.

[0170] Em um ou mais aspectos ilustrativos, as funções descritas podem ser implementadas em hardware, software, firmware, ou qualquer combinação dos mesmos. Se implementadas em software, as funções podem ser armazenadas como uma ou mais instruções ou código em um meio legível por computador não transitório, ou um meio legível por processador não transitório. As etapas de um método ou algoritmo descritas aqui podem ser consubstanciadas em um

módulo de software executável por processador que pode residir em um meio de armazenamento legível por processador não transitório ou um meio de armazenamento legível por computador não transitório. O meio de armazenamento legível por computador ou processador não transitório pode ser qualquer meio de armazenamento que possa ser acessado por um computador ou um processador. Por meio de exemplo, mas não de limitação, tal meio legível por computador ou processador não transitório pode incluir memória RAM, ROM, EEPROM, FLASH, CD-ROM ou outro armazenamento em disco ótico, armazenamento em disco magnético ou outros dispositivos de armazenamento magnéticos, ou qualquer outro meio que possa ser utilizado para armazenar o código de programa desejado na forma de instruções ou estruturas de dados e que possa ser acessado por um computador. Disquete e disco, como utilizados aqui, incluem disco compacto (CD), disco a laser, disco ótico, disco versátil digital (DVD), disquete e disco blu-ray onde disquetes normalmente reproduzem os dados magneticamente, enquanto discos reproduzem os dados oticamente com lasers. As combinações do acima também são incluídas no escopo de meio legível por computador e processador não transitório. Adicionalmente, as operações de um método ou algoritmo podem residir como um ou qualquer combinação ou conjunto de códigos e/ou instruções em um meio legível por processador não transitório e/ou meio legível por computador, que pode ser incorporado em um produto de programa de computador.

[0171] A descrição anterior dos aspectos descritos é fornecida para permitir que os versados na técnica criem ou façam uso da presente invenção. Várias modificações a esses aspectos serão prontamente aparentes aos versados na técnica, e os princípios genéricos definidos aqui podem ser aplicados a outros aspectos sem se

distanciar do espirito ou escopo da invenção. Dessa forma, a presente invenção não deve ser limitada aos aspectos ilustrados aqui, mas deve ser acordado o escopo mais amplo consistente com as reivindicações a seguir e princípios e características de novidade descritos aqui.

REIVINDICAÇÕES

1. Método (1000) de gerenciamento de memória em um dispositivo de computação, caracterizado pelo fato de que compreende:

inicializar (1002) um hipervisor (212), um monitor de segurança (214), e um sistema operacional de alto nível, HLOS, (208);

desabilitar (1008) o hipervisor (212) após a inicialização;

monitorar (1009) um sinal do monitor de segurança (214) para iniciar uma sessão de sandbox;

habilitar (1012) o hipervisor (212) quando o sinal é recebido para iniciar a sessão de sandbox;

implementar (1014) controle de acesso enquanto o hipervisor está habilitado; e

alocar memória pelo HLOS (208) durante a sessão de sandbox de modo que um ou mais endereços físicos intermediários em um espaço de endereço físico intermediário (720) sejam mapeados para um ou mais endereços físicos correspondentes em um espaço de endereço físico (830).

2. Método, de acordo com a reivindicação 1, caracterizado pelo fato de que desabilitar (1008) o hipervisor (212) compreende:

configurar todos os bancos de contexto de unidades de gerenciamento de memória de sistema, SMMU, para contornar tradução de segundo estágio; e

desligar traduções de segundo estágio para o HLOS (208).

3. Método (1000), de acordo com a reivindicação 1, caracterizado pelo fato de que compreende adicionalmente:

determinar se a sessão de sandbox está terminada;

realizar (1018) um procedimento de interrupção de sessão de sandbox quando for determinado que a sessão de sandbox está terminada; e

desabilitar o hipervisor (212) após realizar o procedimento de interrupção de sessão de sandbox.

4. Método (1000), de acordo com a reivindicação 3, caracterizado pelo fato de que realizar o procedimento de interrupção de sessão de sandbox compreende:

liberar todos os armazenadores para um componente que sofreu sandbox; e

restaurar tabelas de página de tradução de segundo estágio para remover todas as fragmentações.

5. Método (1000), de acordo com a reivindicação 1, caracterizado pelo fato de que habilitar o hipervisor (212) compreende:

habilitar unidades de gerenciamento de memória de segundo estágio PL0 e PL1;

configurar solicitações de interrupção a serem realizadas no modo de hipervisor; e

chamar acionadores SMMU para colocar todos os bancos de contexto SMMU em traduções de primeiro estágio encaixados dentro das traduções de segundo estágio.

6. Método (1000), de acordo com a reivindicação 5, caracterizado pelo fato de que compreende adicionalmente tratar falhas SMMU.

7. Método (1000), de acode com a reivindicação 1, caracterizado pelo fato de que implementar (1014) controle de acesso compreende implementar traduções de segundo estágio.

8. Dispositivo de computação (200), caracterizado pelo fato de que compreende:

meios para inicializar um hipervisor (212), um monitor de segurança (214), e um sistema operacional de alto nível (HLOS) (208);

meios para desabilitar o hipervisor (212) após a inicialização;

meios para monitorar um sinal de um monitor de segurança (214) para iniciar uma sessão de sandbox;

meios para habilitar o hipervisor (212) quando o sinal é recebido para iniciar a sessão de sandbox;

meios para implementar controle de acesso enquanto o hipervisor (212) está habilitado; e

meios para alocar memória pelo HLOS (214) durante a sessão de sandbox de modo que um ou mais endereços físicos intermediários em um espaço de endereço físico intermediário (720) sejam mapeados para um ou mais endereços físicos correspondentes em um espaço de endereço físico (830).

9. Dispositivo de computação (200), de acordo com a reivindicação 8, caracterizado pelo fato de que os meios para desabilitar o hipervisor (212) compreendem:

meios para configurar todos os bancos de contexto de unidades de gerenciamento de memória de sistema, SMMU, para contornar tradução de segundo estágio; e

meios para desligar traduções de segundo estágio para o HLOS (214).

10. Dispositivo de computação (200), de acordo com a reivindicação 8, caracterizado pelo fato de que compreende adicionalmente:

meios para determinar se a sessão de sandbox está terminada;

meios para realizar um procedimento de interrupção de sessão de sandbox quando for determinado que a sessão de sandbox está terminada; e

meios para desabilitar o hipervisor após realizar o procedimento de interrupção de sessão de sandbox.

11. Dispositivo de computação (200), de acordo com a reivindicação 10, caracterizado pelo fato de que os meios para realizar o procedimento de interrupção de sessão de sandbox compreendem:

meios para liberar todos os armazenadores para um componente que sofreu sandbox; e

meios para restaurar tabelas de página de tradução de segundo estágio para remover todas fragmentações.

12. Dispositivo de computação (200), de acordo com a reivindicação 8, caracterizado pelo fato de que os meios para habilitar o hipervisor compreendem:

meios para habilitar unidades de gerenciamento de memória de segundo estágio PL0 e PL1;

meios para configurar solicitações de interrupção a serem realizadas em modo de hipervisor; e

meios para chamar acionadores SMMU para colocar todos os bancos de contexto SMMU ativos em traduções de primeiro estágio encaixados nas traduções de segundo estágio.

13. Dispositivo de computação (200), de acordo com a reivindicação 12, caracterizado pelo fato de que compreende adicionalmente meios para tratar falhas SMMU.

14. Dispositivo de computação (200), de acordo com a reivindicação 8, caracterizado pelo fato de que os meios para implementar controle de acesso compreendem meios para implementar traduções de segundo estágio.

15. Memória caracterizada pelo fato de que compreende instruções para realizar o método conforme definido em qualquer uma das reivindicações 1 a 7.

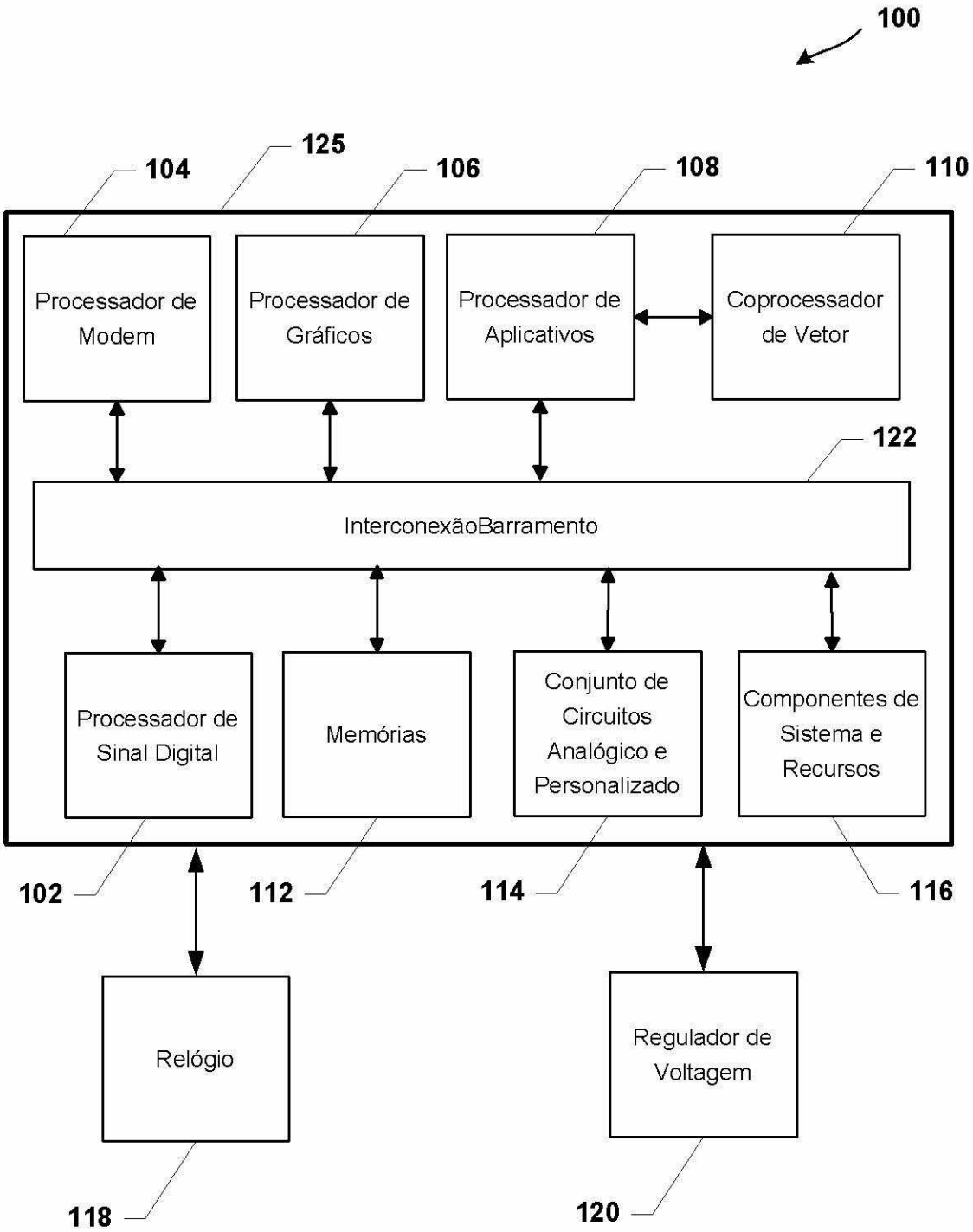
100
104
125
106
108
110
Processador de Modem
Processador de Gráficos
Processador de Aplicativos
Coprocessador de Vetor
122
InterconexãoBarramento
Processador de Sinal Digital
Memórias
Conjunto de Circuitos Analógico e Personalizado
Componentes de Sistema e Recursos
102
112
114
116
Relógio
Regulador de Voltagem
118
120

FIG. 1

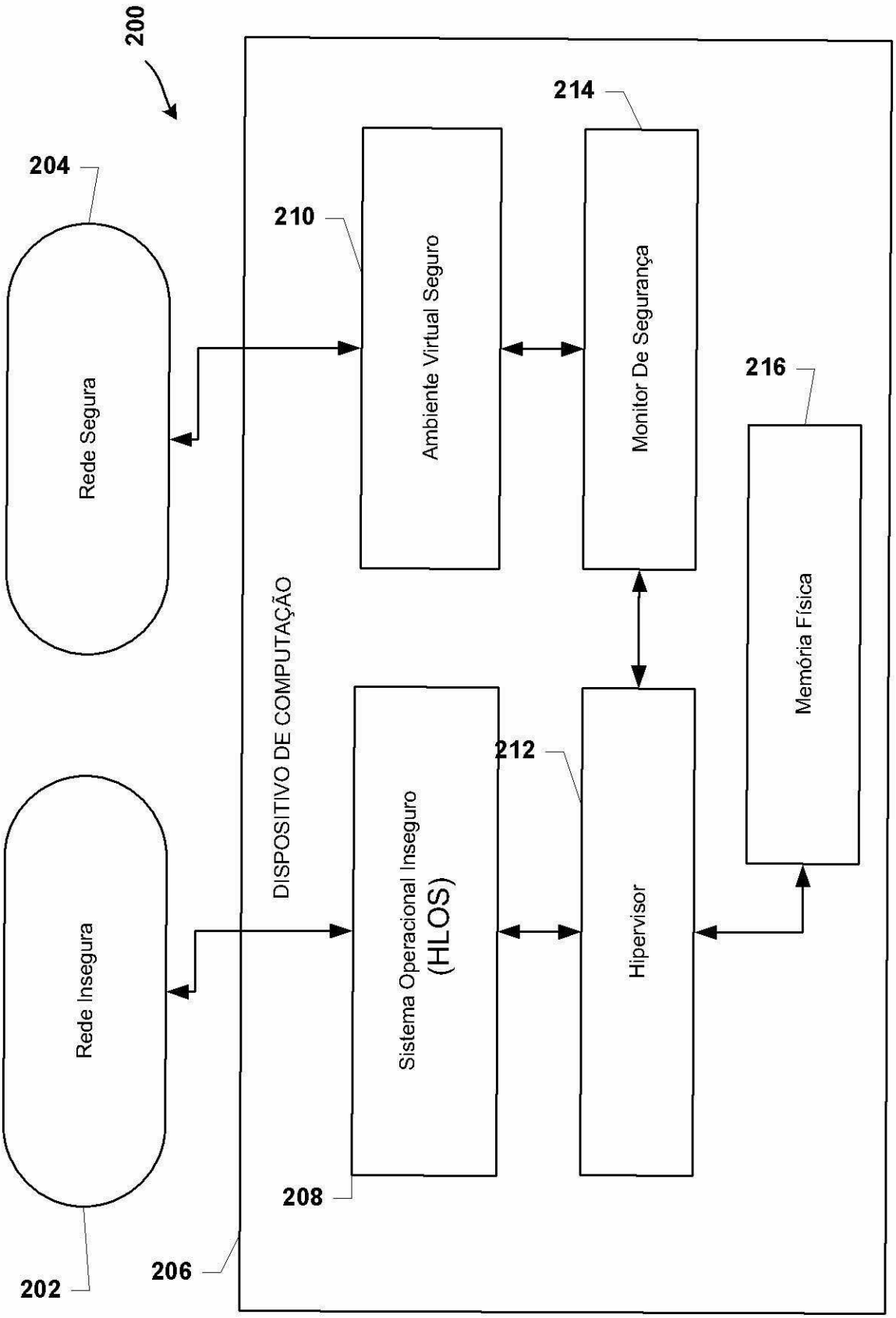
200
204
Rede Segura
210
Ambiente Virtual Seguro
214
Monitor De Segurança
216
Memória Física
DISPOSITIVO DE COMPUTAÇÃO
212
Hipervisor
Sistema Operacional Inseguro
(HLOS)
208
Rede Insegura
202
206

FIG. 2

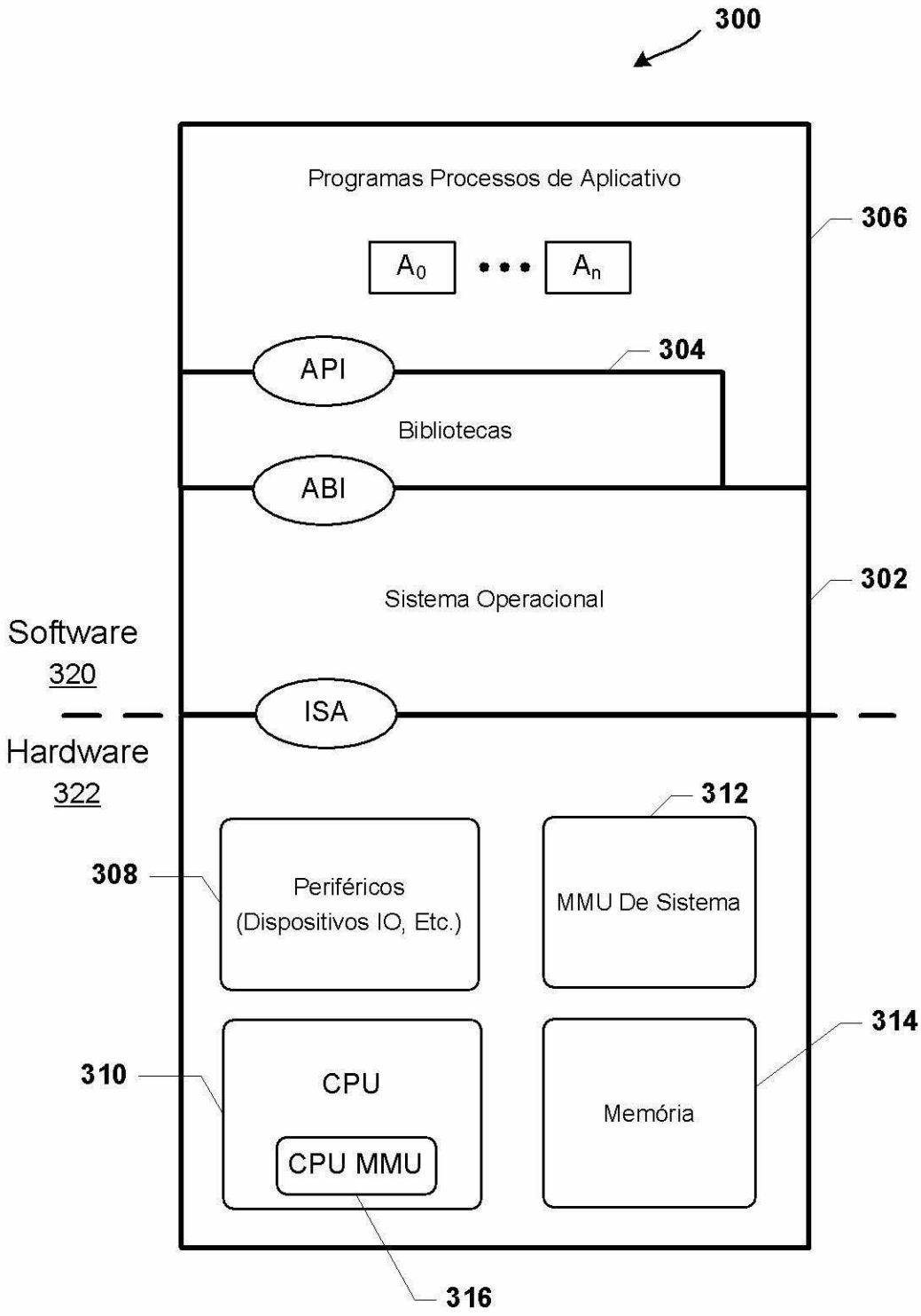
300
Programas Processos de Aplicativo
306
A0
An
API
304
Bibliotecas
ABI
302
Sistema Operacional
Software
320
ISA
Hardware
322
312
308
Periféricos
(Dispositivos IO, Etc.)
MMU De Sistema
314
310
CPU
Memória
CPU MMU
316

FIG. 3

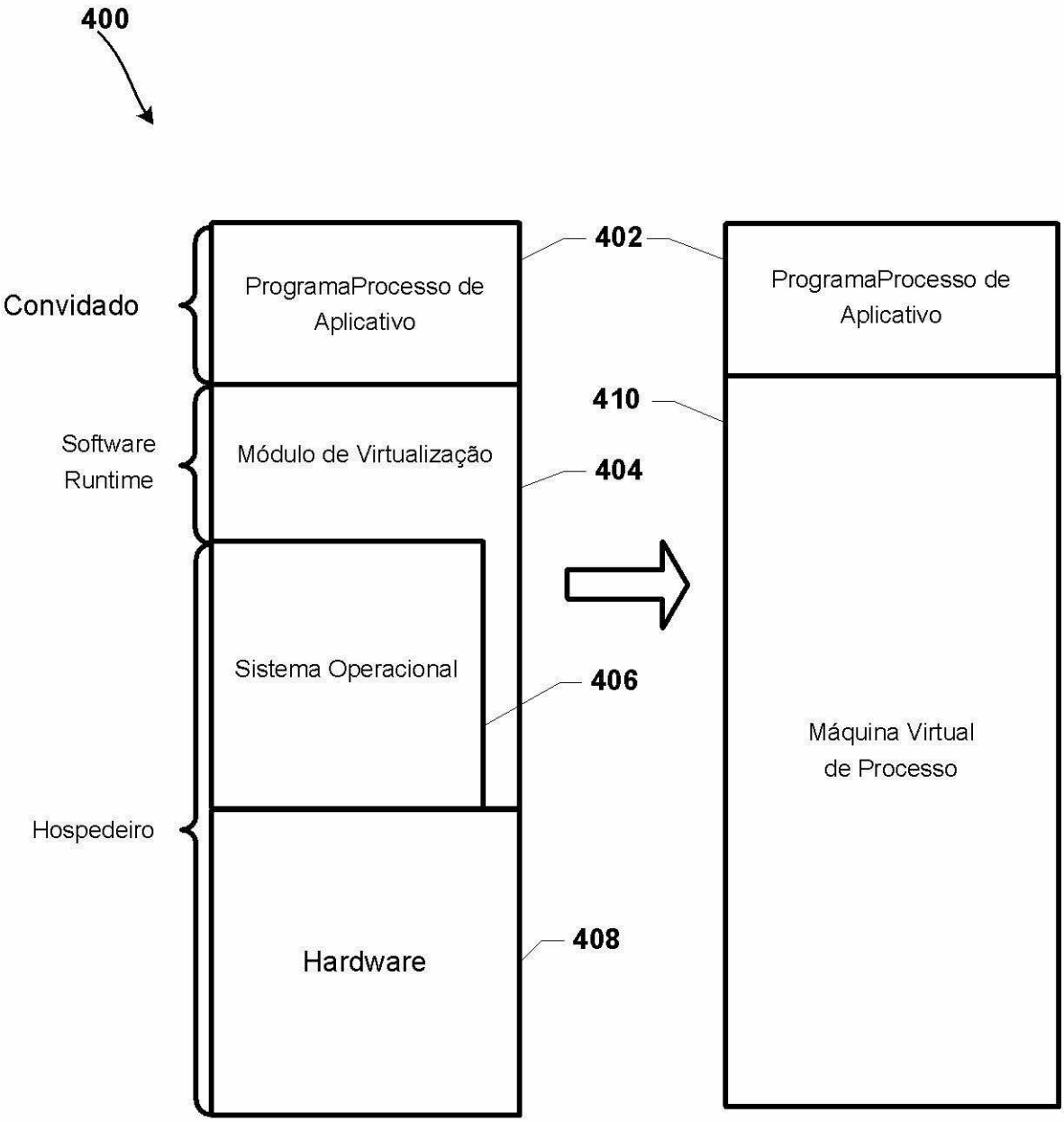
400
Convidado
ProgramaProcesso de Aplicativo
402
ProgramaProcesso de Aplicativo
410
Software Runtime
Módulo de Virtualização
404
Sistema Operacional
406
Máquina Virtual de Processo
Hospedeiro
Hardware
408

FIG. 4

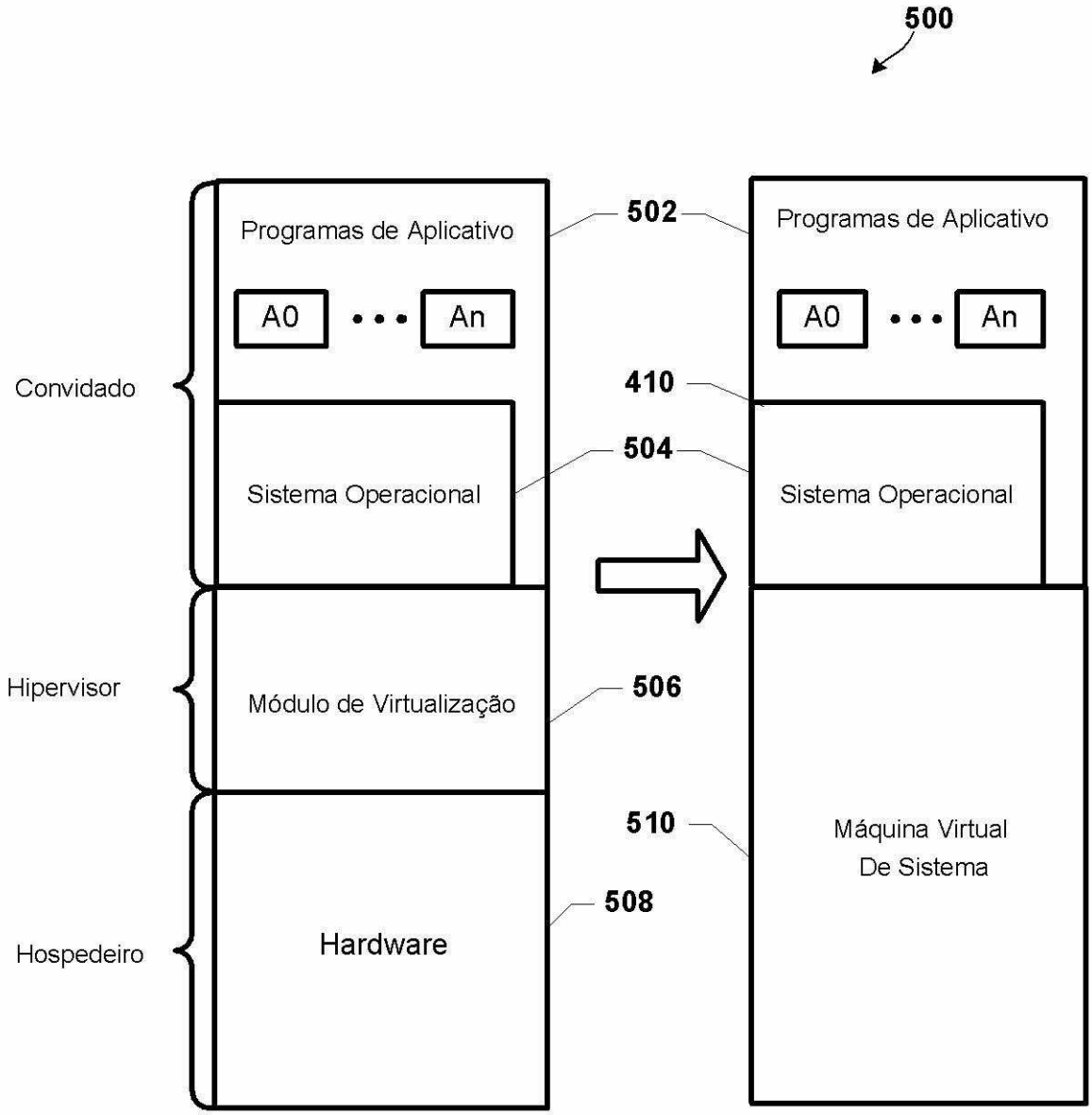
500
Programas de Aplicativo
A0
An
502
Programas de Aplicativo
A0
An
Convidado
410
Sistema Operacional
504
Sistema Operacional
Hipervisor
Módulo de Virtualização
506
510
Máquina Virtual
De Sistema
508
Hospedeiro
Hardware

FIG. 5

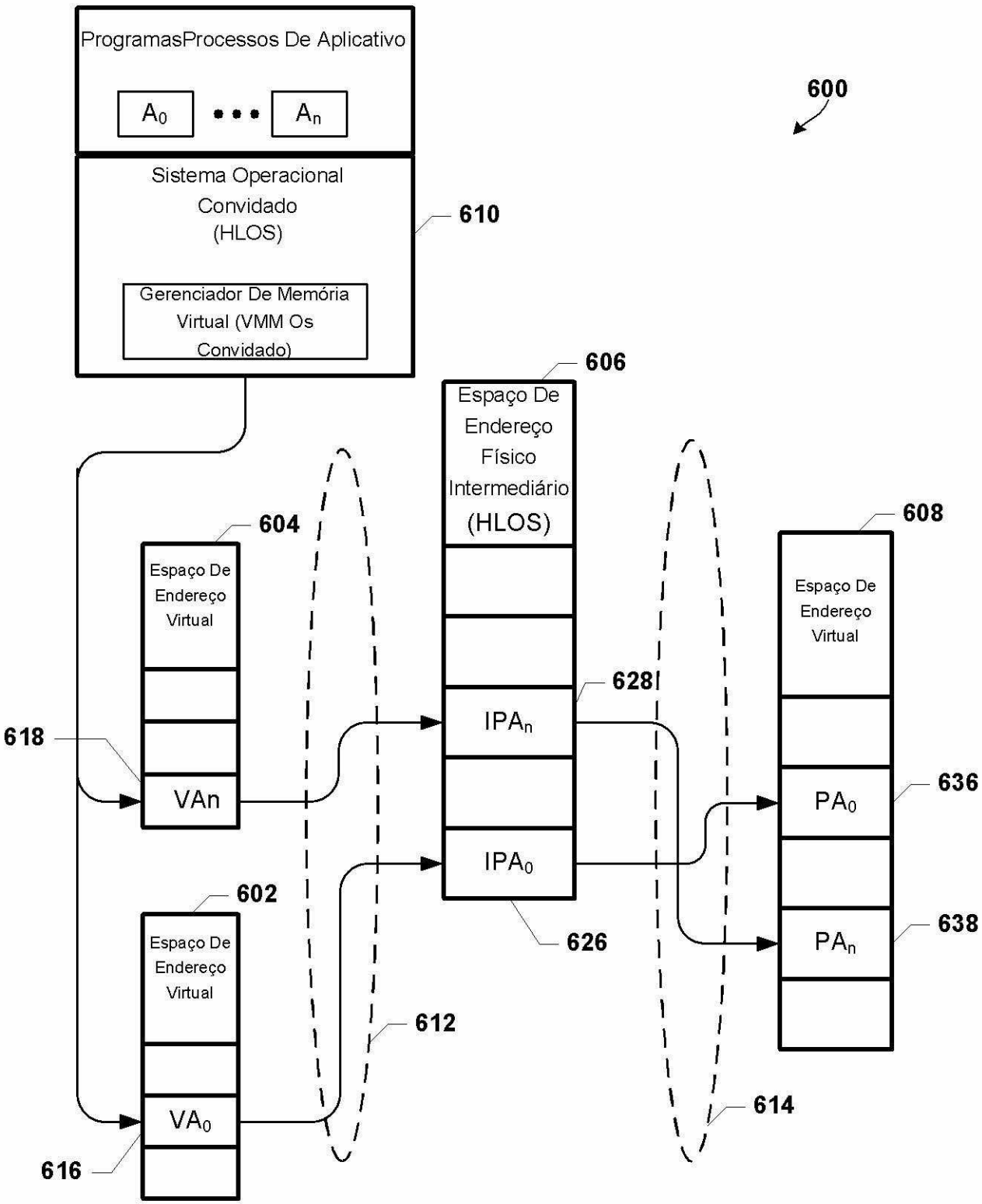
ProgramasProcessos De Aplicativo
A0
An
Sistema Operacional Convidado (HLOS)
610
Gerenciador De Memória Virtual (VMM Os Convidado)
600
606
Espaço De Endereço Físico Intermediário (HLOS)
604
Espaço De Endereço Virtual
608
Espaço De Endereço Virtual
618
VAn
IPAn
628
PA0
636
IPA0
602
Espaço De Endereço Virtual
PAn
638
626
612
VA0
614
616

FIG. 6

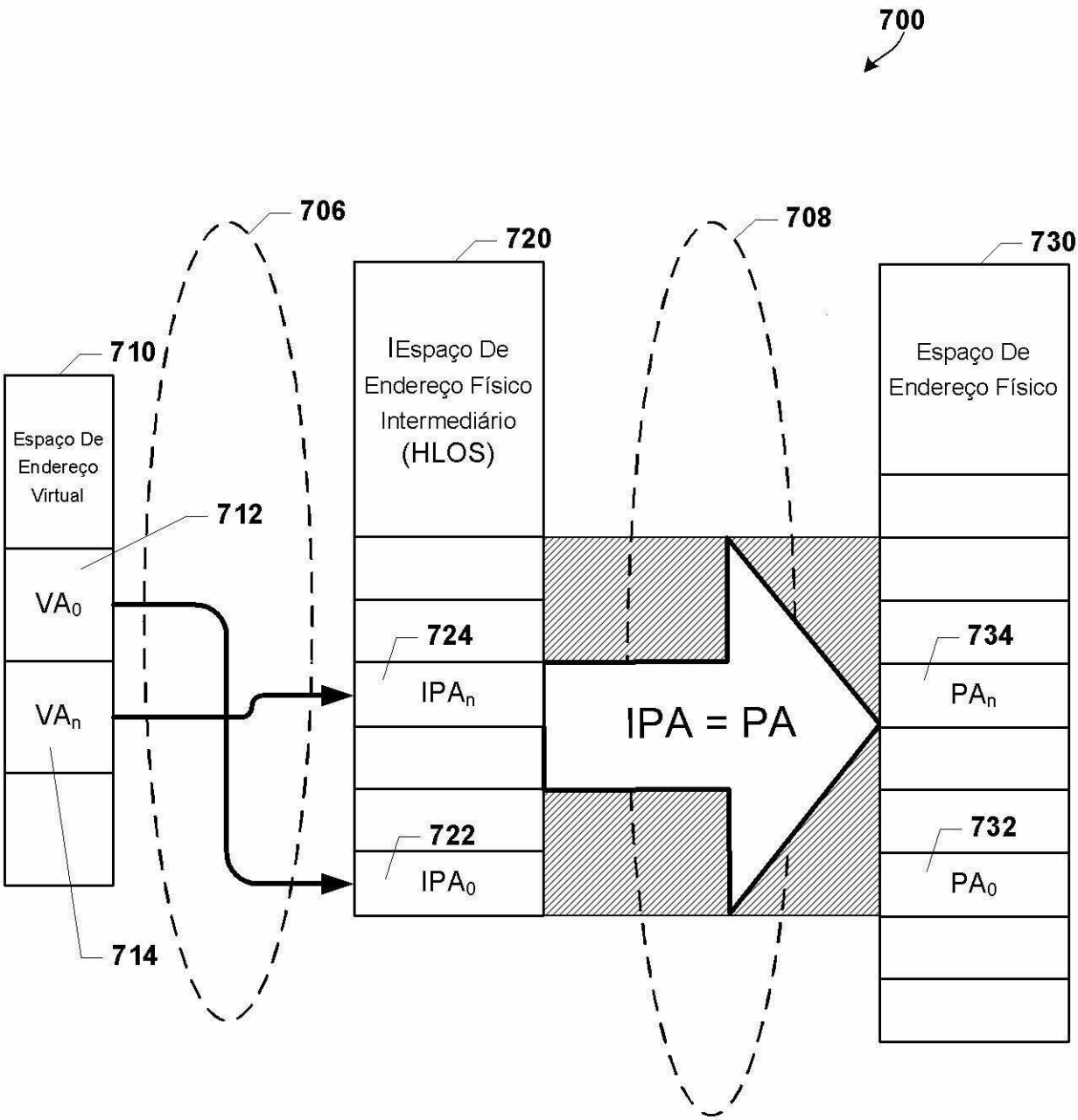
700
706
708
720
730
710
IEspaço De Endereço Físico Intermediário (HLOS)
Espaço De Endereço Físico
Espaço De Endereço Virtual
712
VA0
VAn
724
IPAn
734
PAn
IPA = PA
722
IPA0
732
PA0
714

FIG. 7

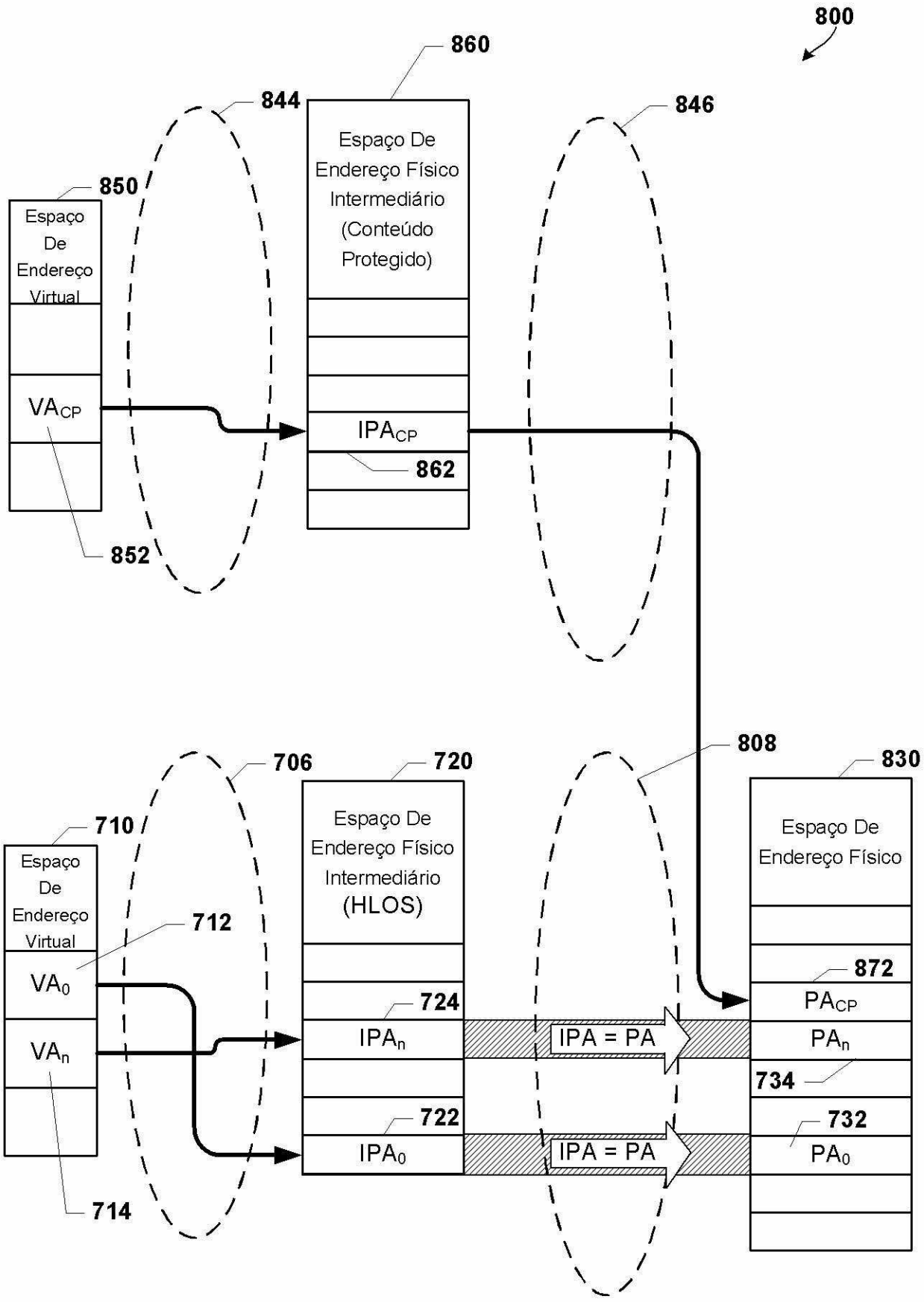
800
860
844
846
850
Espaço De Endereço Virtual
Espaço De Endereço Físico Intermediário (Conteúdo Protegido)
VA_CP
IPA_CP
862
852
808
706
720
830
710
Espaço De Endereço Virtual
Espaço De Endereço Físico Intermediário (HLOS)
Espaço De Endereço Físico
712
872
VA_0
724
IPA_n
IPA = PA
PA_CP
PA_n
VA_n
734
722
732
IPA_0
IPA = PA
PA_0
714

FIG. 8

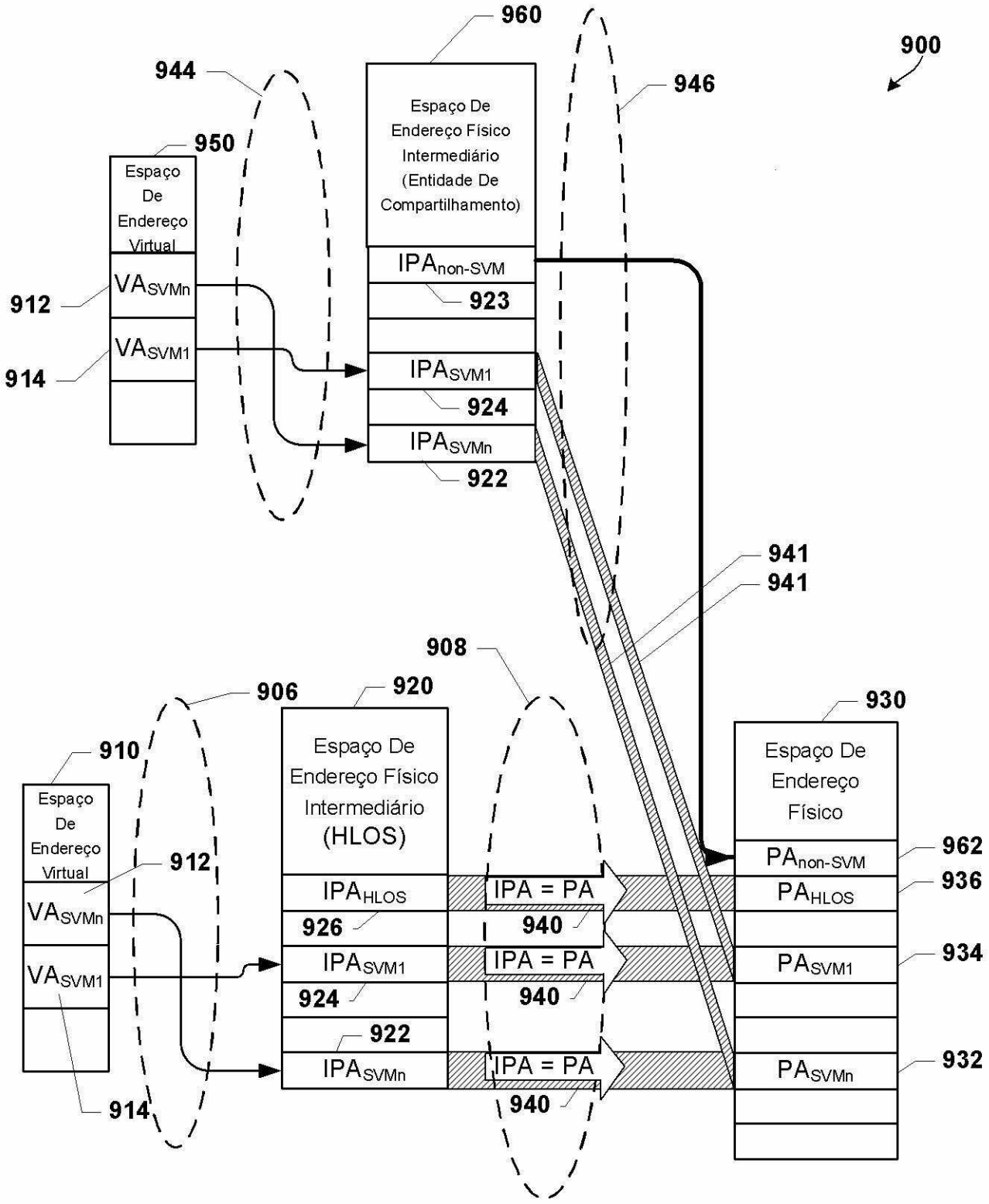
900
960
944
946
950
Espaço De Endereço Virtual
Espaço De Endereço Físico Intermediário (Entidade De Compartilhamento)
912
914
VASVMn
VASVM1
IPAnon-SVM
923
IPASVM1
924
IPASVMn
922
941
941
908
906
920
930
910
Espaço De Endereço Virtual
Espaço De Endereço Físico Intermediário (HLOS)
Espaço De Endereço Físico
912
VASVMn
VASVM1
IPAHLOS
926
IPASVM1
924
922
IPASVMn
IPA = PA
940
IPA = PA
940
IPA = PA
940
PAnon-SVM
PAHLOS
PASVM1
PASVMn
962
936
934
932
914

FIG. 9

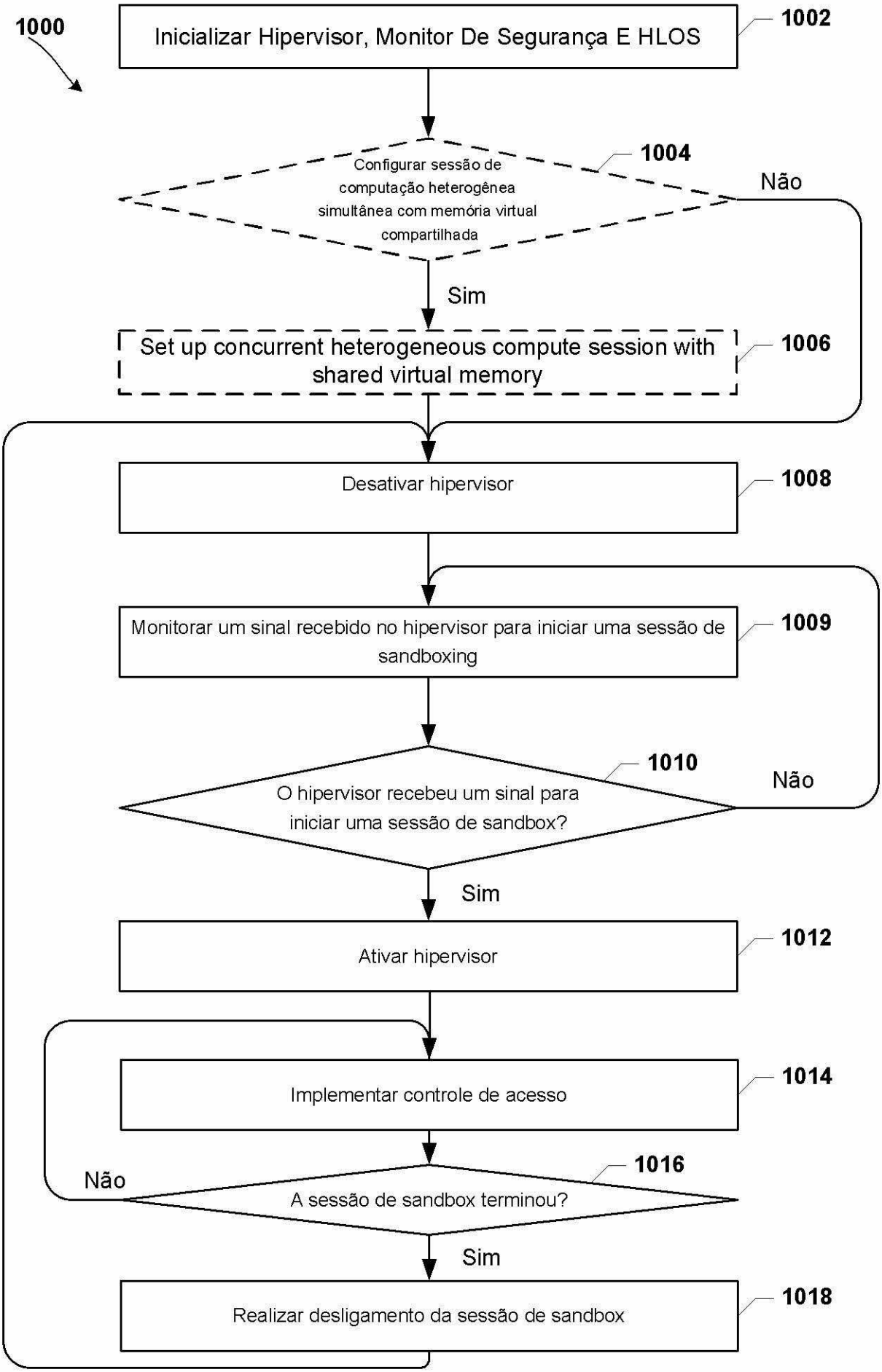
1000
Inicializar Hipervisor, Monitor De Segurança E HLOS
1002
Configurar sessão de computação heterogênea simultânea com memória virtual compartilhada
1004
Não
Sim
Set up concurrent heterogeneous compute session with shared virtual memory
1006
Desativar hipervisor
1008
Monitorar um sinal recebido no hipervisor para iniciar uma sessão de sandboxing
1009
O hipervisor recebeu um sinal para iniciar uma sessão de sandbox?
1010
Não
Sim
Ativar hipervisor
1012
Implementar controle de acesso
1014
A sessão de sandbox terminou?
1016
Não
Sim
Realizar desligamento da sessão de sandbox
1018

FIG. 10

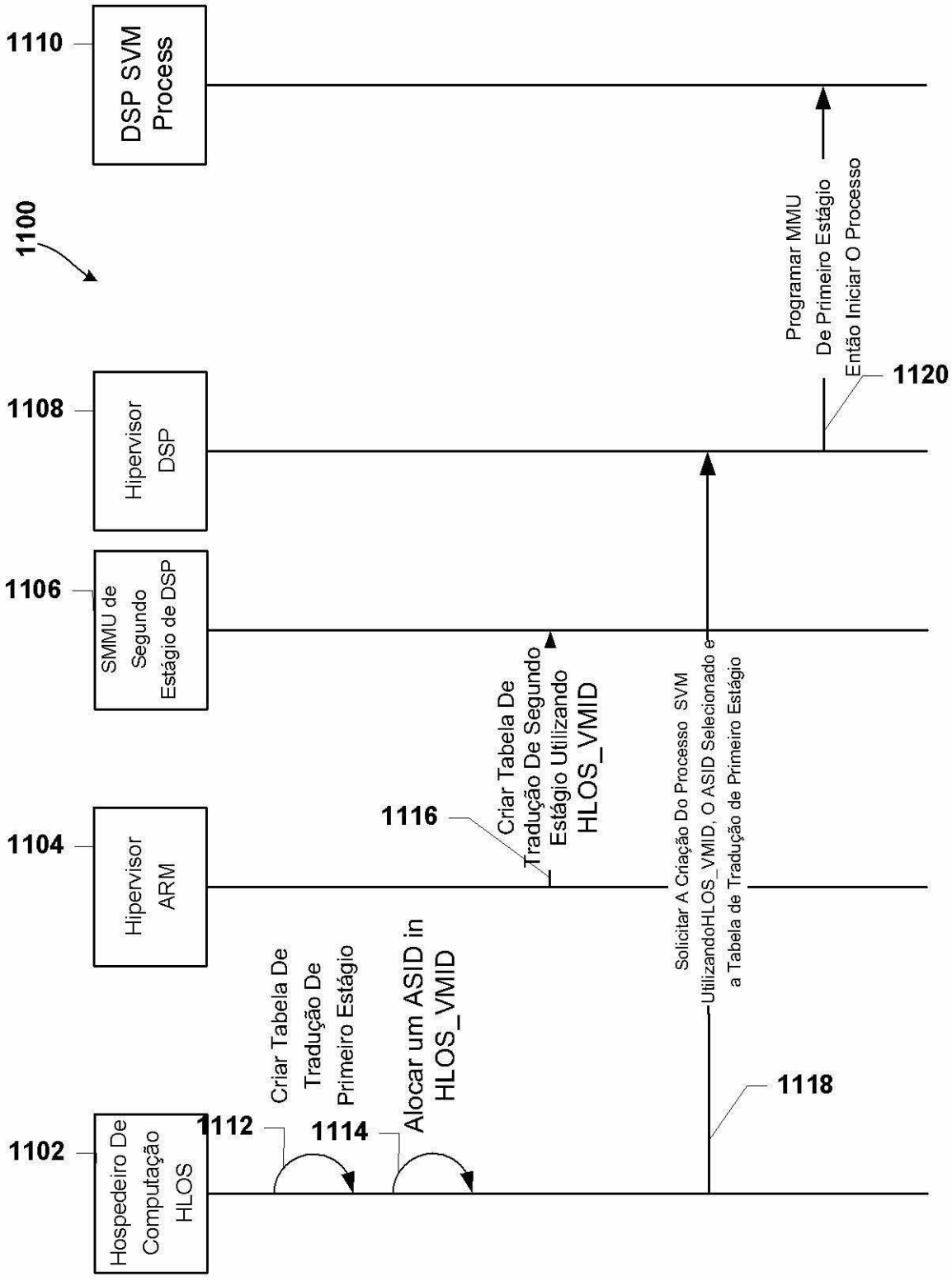
1100
1102
Hospedeiro De Computação HLOS
1104
Hipervisor ARM
1106
SMMU de Segundo Estágio de DSP
1108
Hipervisor DSP
1110
DSP SVM Process
1112
Criar Tabela De Tradução De Primeiro Estágio
1114
Alocar um ASID in HLOS_VMID
1116
Criar Tabela De Tradução De Segundo Estágio Utilizando HLOS_VMID
1118
Solicitar A Criação Do Processo SVM UtilizandoHLOS_VMID, O ASID Selecionado e a Tabela de Tradução de Primeiro Estágio
1120
Programar MMU De Primeiro Estágio Então Iniciar O Processo

FIG. 11

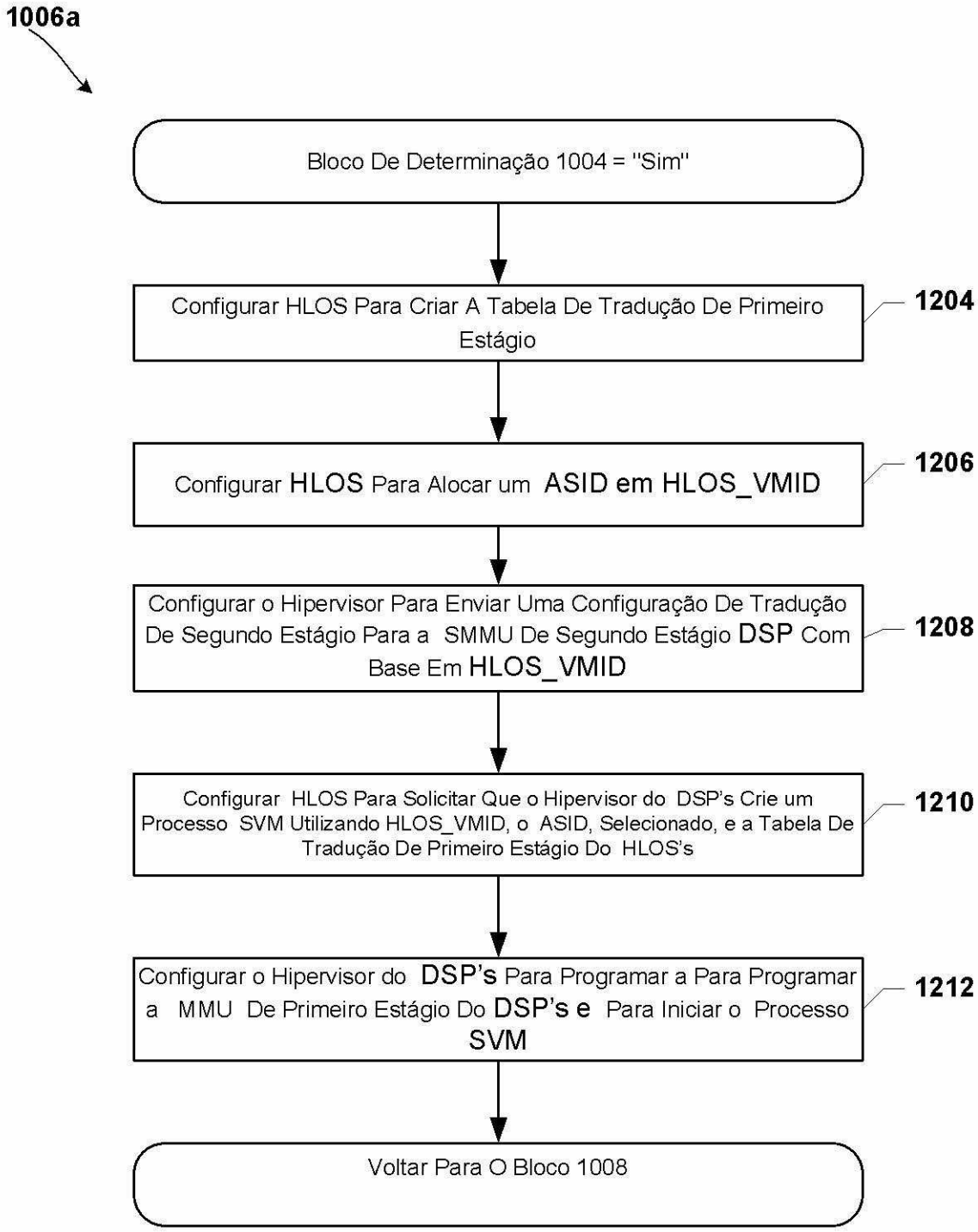
1006a
Bloco De Determinação 1004 = "Sim"
Configurar HLOS Para Criar A Tabela De Tradução De Primeiro Estágio
1204
Configurar HLOS Para Alocar um ASID em HLOS_VMID
1206
Configurar o Hipervisor Para Enviar Uma Configuração De Tradução De Segundo Estágio Para a SMMU De Segundo Estágio DSP Com Base Em HLOS_VMID
1208
Configurar HLOS Para Solicitar Que o Hipervisor do DSP's Crie um Processo SVM Utilizando HLOS_VMID, o ASID, Selecionado, e a Tabela De Tradução De Primeiro Estágio Do HLOS's
1210
Configurar o Hipervisor do DSP's Para Programar a Para Programar a MMU De Primeiro Estágio Do DSP's e Para Iniciar o Processo SVM
1212
Voltar Para O Bloco 1008

FIG. 12

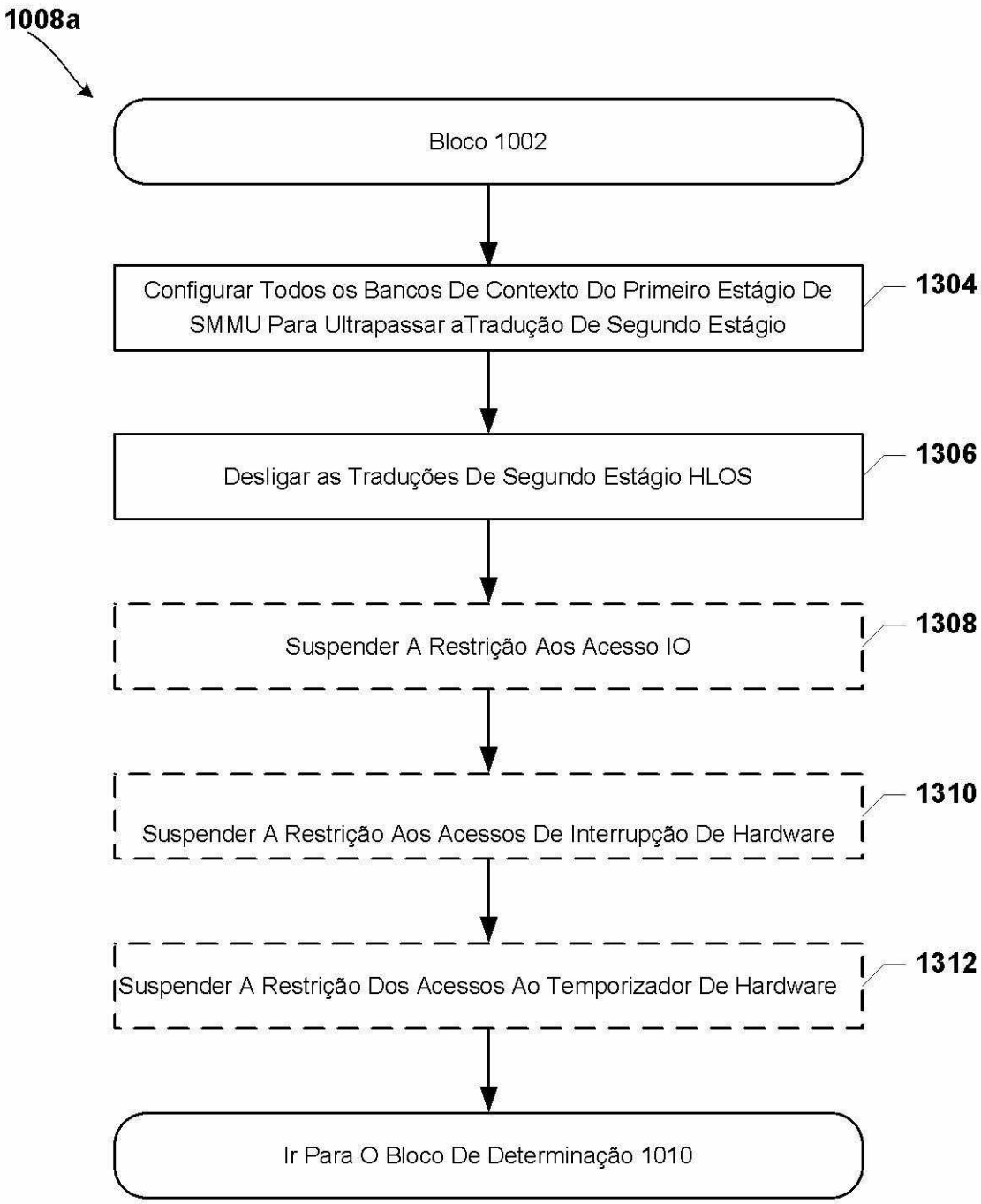
1008a
Bloco 1002
Configurar Todos os Bancos De Contexto Do Primeiro Estágio De SMMU Para Ultrapassar aTradução De Segundo Estágio
1304
Desligar as Traduções De Segundo Estágio HLOS
1306
Suspender A Restrição Aos Acesso IO
1308
Suspender A Restrição Aos Acessos De Interrupção De Hardware
1310
Suspender A Restrição Dos Acessos Ao Temporizador De Hardware
1312
Ir Para O Bloco De Determinação 1010

FIG. 13

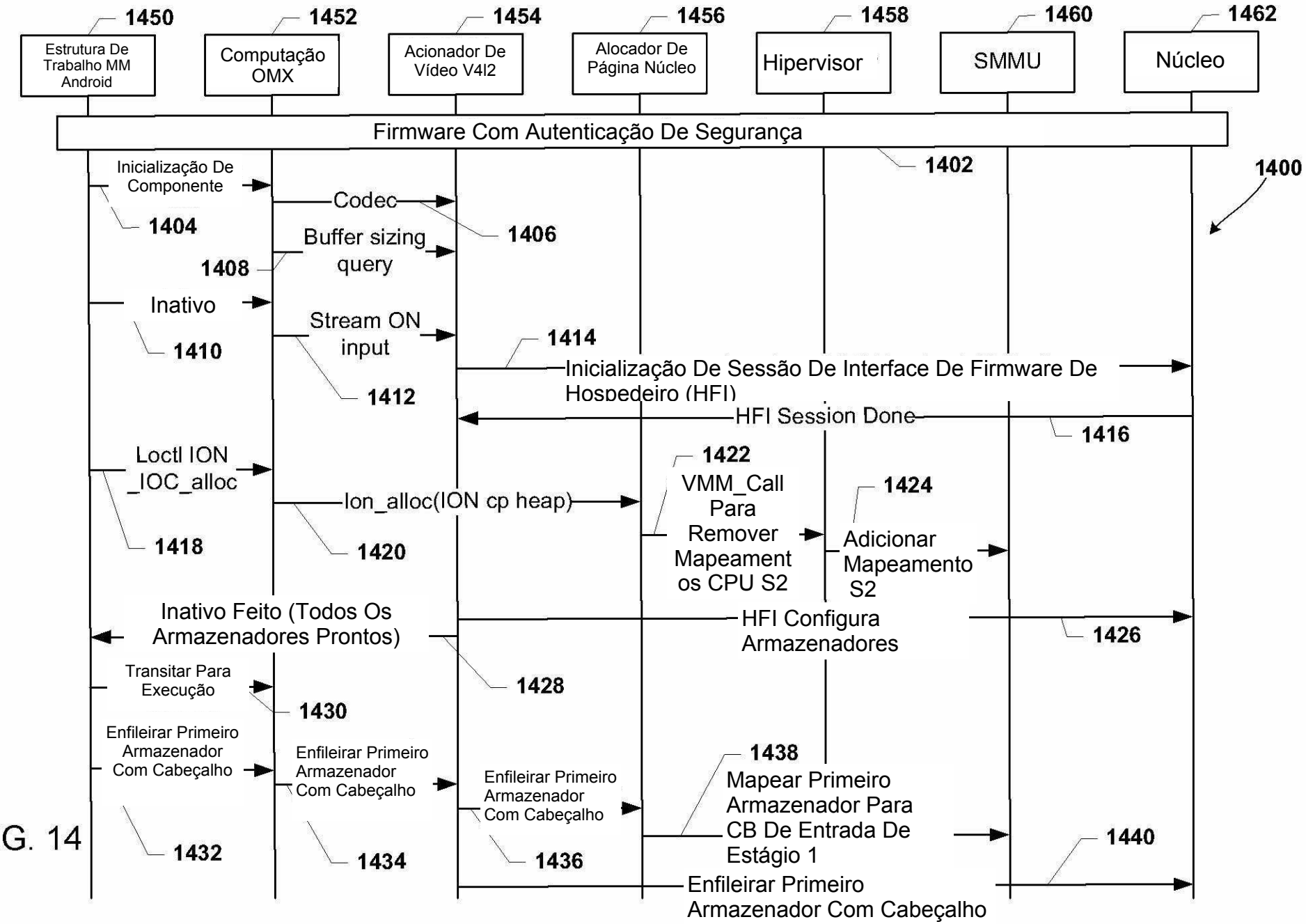
1450
Estrutura De Trabalho MM Android
1452
Computação OMX
1454
Acionador De Vídeo V4l2
1456
Alocador De Página Núcleo
1458
Hipervisor
1460
SMMU
1462
Núcleo
Firmware Com Autenticação De Segurança
1402
1400
Inicialização De Componente
1404
Codec
1406
Buffer sizing query
1408
Inativo
1410
Stream ON input
1412
1414
Inicialização De Sessão De Interface De Firmware De Hospedeiro (HFI)
HFI Session Done
1416
Loctl ION _IOC_alloc
1418
Ion_alloc(ION cp heap)
1420
1422
VMM_Call Para Remover Mapeamentos CPU S2
1424
Adicionar Mapeamento S2
Inativo Feito (Todos Os Armazenadores Prontos)
1428
HFI Configura Armazenadores
1426
Transitar Para Execução
1430
Enfileirar Primeiro Armazenador Com Cabeçalho
1432
Enfileirar Primeiro Armazenador Com Cabeçalho
1434
Enfileirar Primeiro Armazenador Com Cabeçalho
1436
1438
Mapear Primeiro Armazenador Para CB De Entrada De Estágio 1
1440
Enfileirar Primeiro Armazenador Com Cabeçalho

FIG. 14

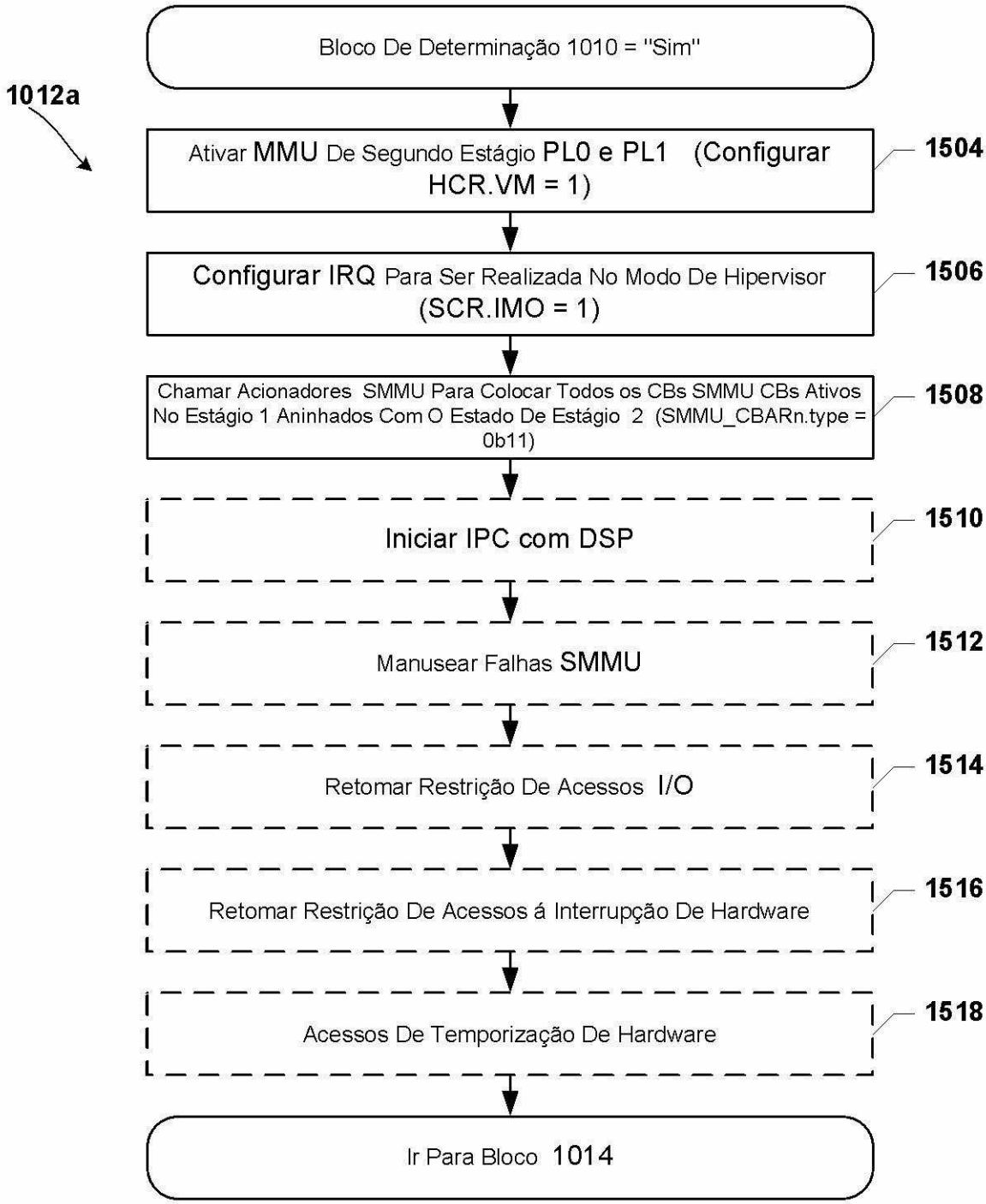
Bloco De Determinação 1010 = "Sim"
1012a
Ativar MMU De Segundo Estágio PL0 e PL1 (Configurar HCR.VM = 1)
1504
Configurar IRQ Para Ser Realizada No Modo De Hipervisor (SCR.IMO = 1)
1506
Chamar Acionadores SMMU Para Colocar Todos os CBs SMMU CBs Ativos No Estágio 1 Aninhados Com O Estado De Estágio 2 (SMMU_CBARn.type = 0b11)
1508
Iniciar IPC com DSP
1510
Manusear Falhas SMMU
1512
Retomar Restrição De Acessos I/O
1514
Retomar Restrição De Acessos á Interrupção De Hardware
1516
Acessos De Temporização De Hardware
1518
Ir Para Bloco 1014

FIG. 15

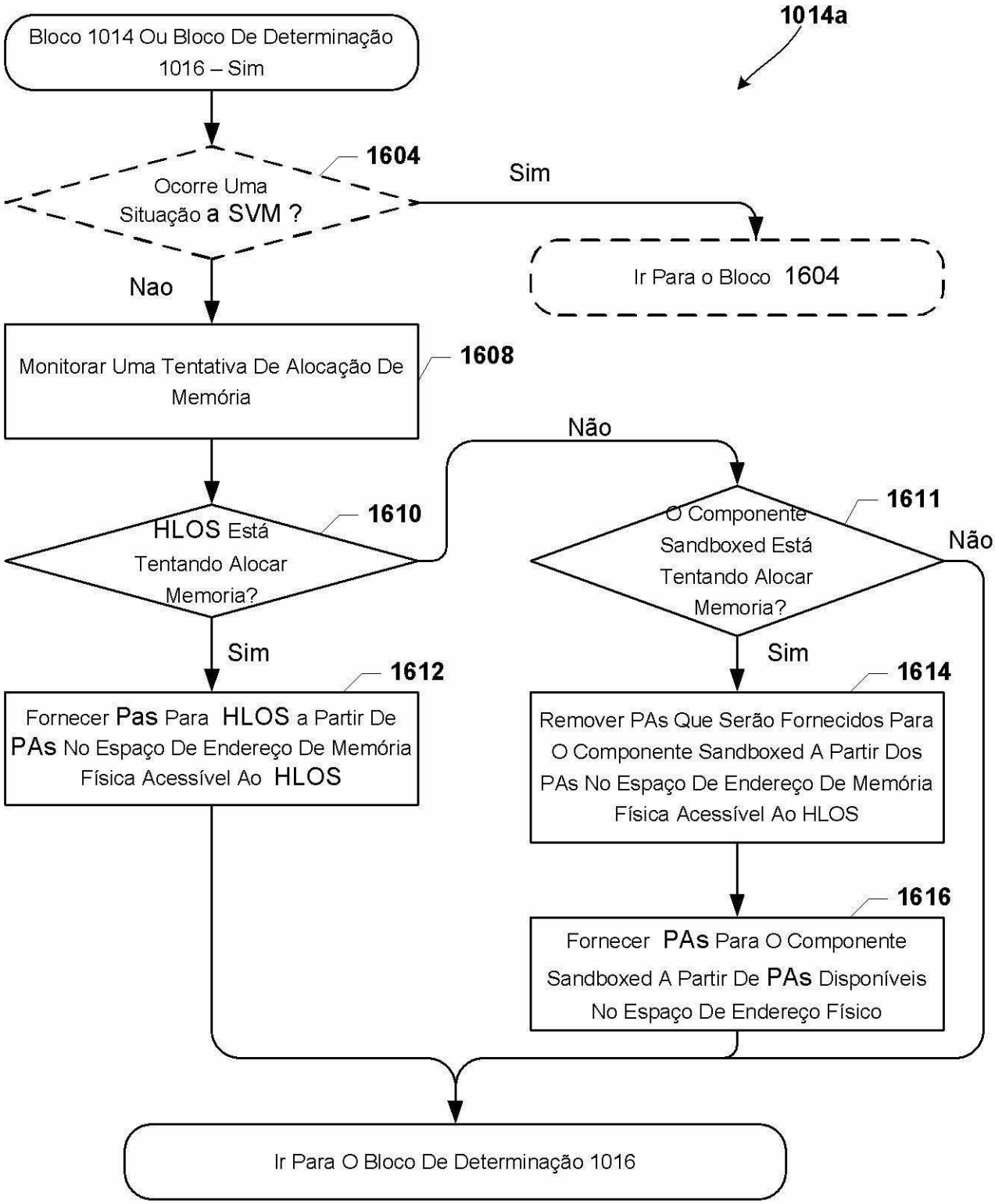
1014a
Bloco 1014 Ou Bloco De Determinação 1016 – Sim
1604
Ocorre Uma Situação a SVM ?
Sim
Ir Para o Bloco 1604
Nao
Monitorar Uma Tentativa De Alocação De Memória
1608
Não
1610
HLOS Está Tentando Alocar Memoria?
1611
O Componente Sandboxed Está Tentando Alocar Memoria?
Não
Sim
Sim
1612
Fornecer Pas Para HLOS a Partir De PAs No Espaço De Endereço De Memória Física Acessível Ao HLOS
1614
Remover PAs Que Serão Fornecidos Para O Componente Sandboxed A Partir Dos PAs No Espaço De Endereço De Memória Física Acessível Ao HLOS
1616
Fornecer PAs Para O Componente Sandboxed A Partir De PAs Disponíveis No Espaço De Endereço Físico
Ir Para O Bloco De Determinação 1016

FIG. 16A

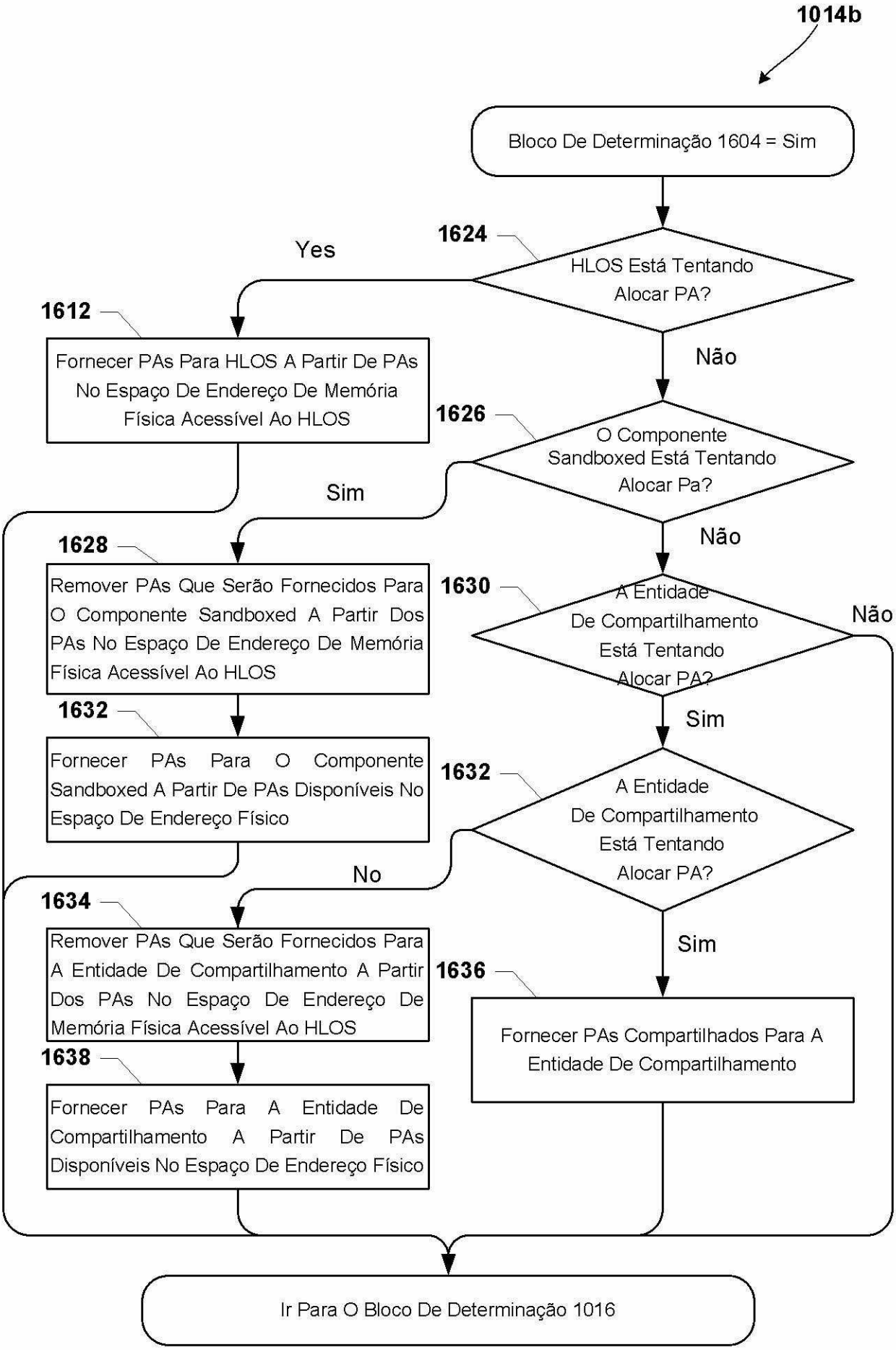
1014b
Bloco De Determinação 1604 = Sim
1624
HLOS Está Tentando Alocar PA?
Yes
Não
1612
Fornecer PAs Para HLOS A Partir De PAs No Espaço De Endereço De Memória Física Acessível Ao HLOS
1626
O Componente Sandboxed Está Tentando Alocar Pa?
Sim
Não
1628
Remover PAs Que Serão Fornecidos Para O Componente Sandboxed A Partir Dos PAs No Espaço De Endereço De Memória Física Acessível Ao HLOS
1630
A Entidade De Compartilhamento Está Tentando Alocar PA?
Não
Sim
1632
Fornecer PAs Para O Componente Sandboxed A Partir De PAs Disponíveis No Espaço De Endereço Físico
1632
A Entidade De Compartilhamento Está Tentando Alocar PA?
No
Sim
1634
Remover PAs Que Serão Fornecidos Para A Entidade De Compartilhamento A Partir Dos PAs No Espaço De Endereço De Memória Física Acessível Ao HLOS
1636
Fornecer PAs Compartilhados Para A Entidade De Compartilhamento
1638
Fornecer PAs Para A Entidade De Compartilhamento A Partir De PAs Disponíveis No Espaço De Endereço Físico
Ir Para O Bloco De Determinação 1016

FIG. 16B

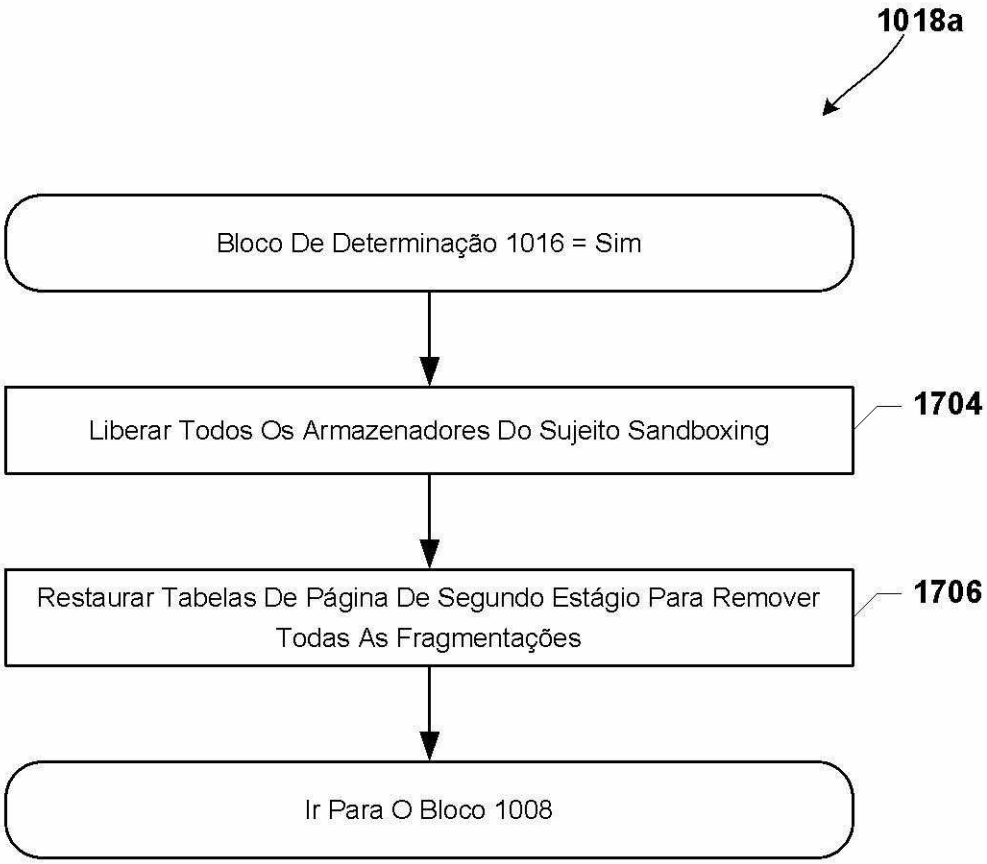
1018a
Bloco De Determinação 1016 = Sim
Liberar Todos Os Armazenadores Do Sujeito Sandboxing
1704
Restaurar Tabelas De Página De Segundo Estágio Para Remover Todas As Fragmentações
1706
Ir Para O Bloco 1008

FIG. 17

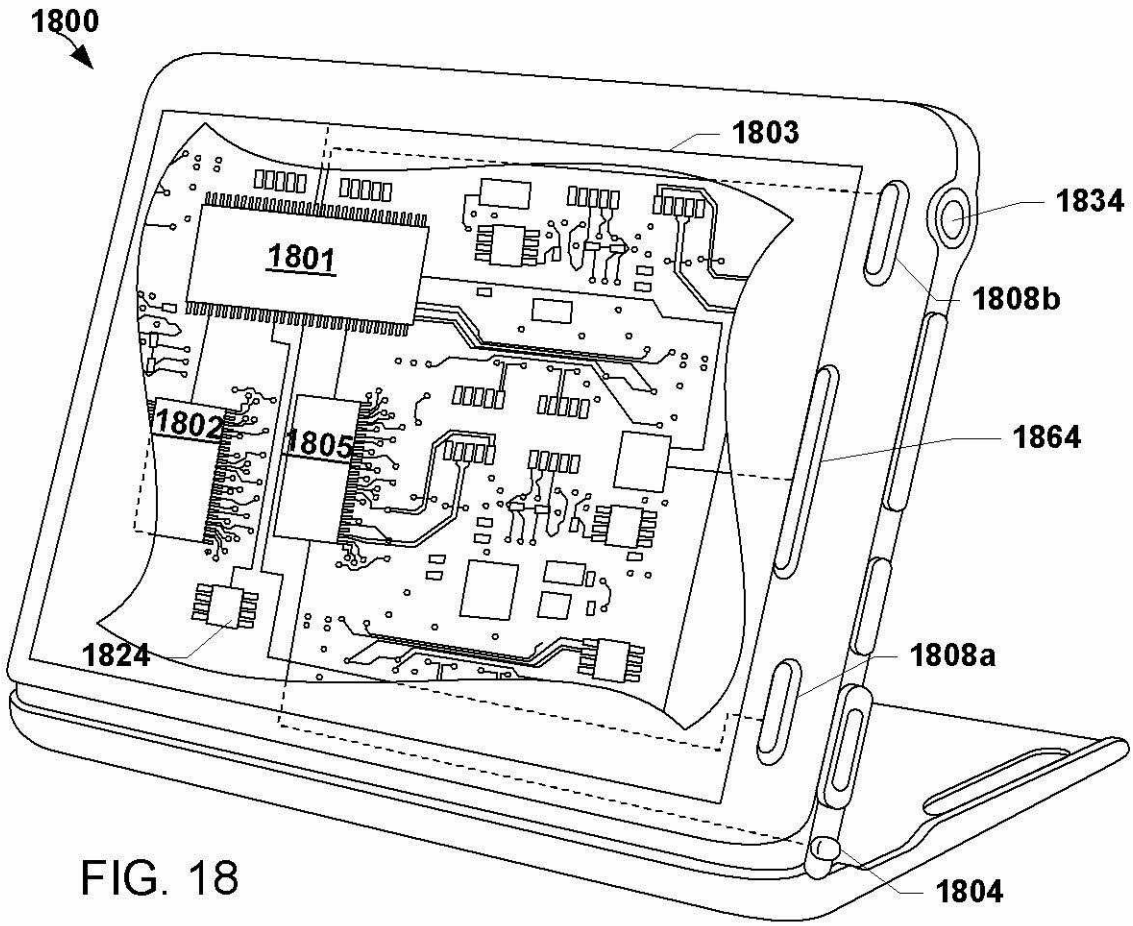
1800
1803
1834
1801
1808b
1802
1805
1864
1824
1808a
1804

FIG. 18

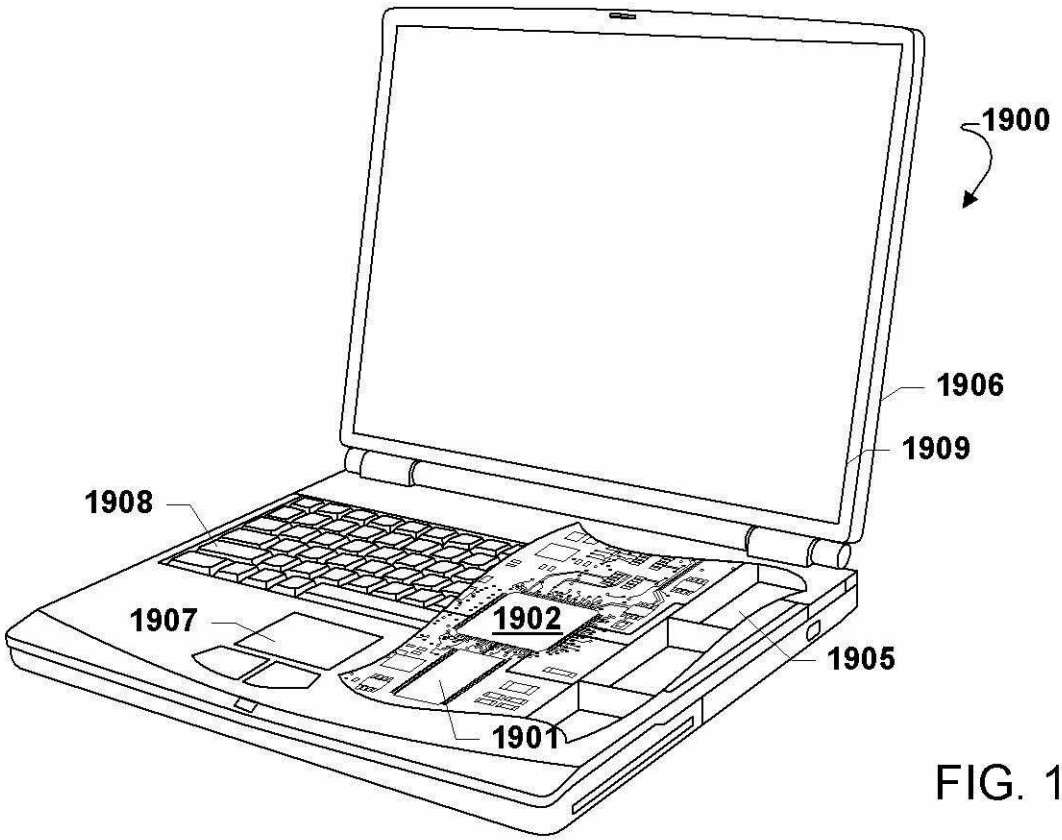
1900
1906
1909
1908
1907
1902
1905
1901

FIG. 19