

(12) **FASCÍCULO DE PATENTE DE INVENÇÃO**

(22) Data de pedido: 2009.02.17	(73) Titular(es): INTERNATIONAL BUSINESS MACHINES CORPORATION NEW ORCHARD ROAD ARMONK, NY 10504 US
(30) Prioridade(s): 2008.02.26 US 37268	
(43) Data de publicação do pedido: 2010.11.10	(72) Inventor(es): LISA HELLER US DAMIAN OSISEK US DAN GREINER US ERWIN PFEFFER DE
(45) Data e BPI da concessão: 2012.03.28 090/2012	(74) Mandatário: LUÍS MANUEL DE ALMADA DA SILVA CARVALHO RUA VÍCTOR CORDON, 14 1249-103 LISBOA PT

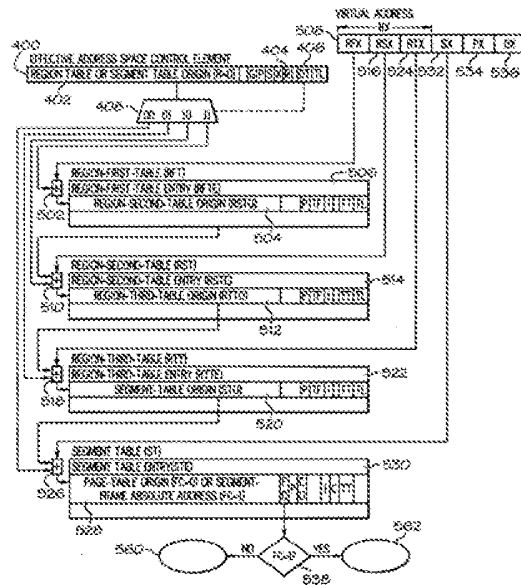
(54) Epígrafe: **CONVERSÃO DINÂMICA DE ENDEREÇO COM QUALIFICADOR DE EXCEPÇÃO DE CONVERSÃO**

(57) Resumo:

O QUE É FORNECIDO É UMA FACILIDADE MELHORADA DE CONVERSÃO DINÂMICA DE ENDEREÇOS. NUMA PRÓPRIA MODALIDADE DE REALIZAÇÃO, SÃO OBTIDOS UM ENDEREÇO VIRTUAL A SER CONVERTIDO E UM ENDEREÇO DE ORIGEM INICIAL DE UMA TABELA DECONVERSÃO DA HIERARQUIA DE TABELAS DE CONVERSÃO. RESULTA A CONVERSÃO DINÂMICA DE ENDEREÇOS VIRTUAIS. EM RESPOSTA A UMA INTERRUPÇÃO DE CONVERSÃO QUE TENHA OCORRIDO DURANTE A CONVERSÃO DINÂMICA DE ENDEREÇOS, OS BITS SÃO ARMAZENADOS NUM CAMPO QUALIFICADOR DE EXCEPÇÃO DE CONVERSÃO (TXQ) PARA INDICAR QUE A EXCEPÇÃO OU FOI UMA EXCEPÇÃO DA DAT DE HOSPEDEIRO A TER OCORRIDO DURANTE A EXECUÇÃO DE UM PROGRAMA HOSPEDEIRO OU UMA EXCEPÇÃO DA DAT DE HOSPEDEIRO A TER OCORRIDO DURANTE A EXECUÇÃO DE UM PROGRAMA DE CONVIDADO. O TXQ TEM AINDA CAPACIDADE DE INDICAR QUE A EXCEPÇÃO ESTAVA ASSOCIADA COM UM ENDEREÇO VIRTUAL DE HOSPEDEIRO DERIVADO DE UM ENDEREÇO REAL DE TRAMA DE PÁGINA DE CONVIDADO OU DE UM ENDEREÇO ABSOLUTO DE TRAMA DE SEGMENTO DE CONVIDADO. O TXQ TEM AINDA CAPACIDADE DE INDICAR QUE UM MAIOR OU MENOR TAMANHO DE TRAMA DE HOSPEDEIRO É PREFERIDO PARA SUPORTAR UMA TRAMA DE CONVIDADO.

RESUMO**"CONVERSÃO DINÂMICA DE ENDEREÇO COM QUALIFICADOR DE
EXCEPÇÃO DE CONVERSÃO"**

O que é fornecido é uma facilidade melhorada de conversão dinâmica de endereços. Numa própria modalidade de realização, são obtidos um endereço virtual a ser convertido e um endereço de origem inicial de uma tabela de conversão da hierarquia de tabelas de conversão. Resulta a conversão dinâmica de endereços virtuais. Em resposta a uma interrupção de conversão que tenha ocorrido durante a conversão dinâmica de endereços, os bits são armazenados num campo qualificador de exceção de conversão (TXQ) para indicar que a exceção ou foi uma exceção da DAT de hospedeiro a ter ocorrido durante a execução de um programa hospedeiro ou uma exceção da DAT de hospedeiro a ter ocorrido durante a execução de um programa de convidado. O TXQ tem ainda capacidade de indicar que a exceção estava associada com um endereço virtual de hospedeiro derivado de um endereço real de trama de página de convidado ou de um endereço absoluto de trama de segmento de convidado. O TXQ tem ainda capacidade de indicar que um maior ou menor tamanho de trama de hospedeiro é preferido para suportar uma trama de convidado.



DESCRIÇÃO

"CONVERSÃO DINÂMICA DE ENDEREÇO COM QUALIFICADOR DE EXCEPÇÃO DE CONVERSÃO"

CAMPO DA INVENÇÃO

A presente invenção diz genericamente respeito a sistemas e métodos para converter um endereço virtual no seio de um sistema de computador e, mais particularmente, refere-se aos sistemas e métodos de conversão de um endereço virtual num endereço real ou absoluto dum bloco de dados num sistema de computador com uma facilidade de conversão dinâmica de endereços em que a conversão de endereço virtual ocorre através de uma hierarquia de tabelas de conversão.

ANTECEDENTES DA INVENÇÃO

A Conversão Dinâmica de Endereços (DAT) fornece a capacidade de interromper a execução de um programa num momento arbitrário, gravá-lo e mais os seus dados numa memória auxiliar, tal como um dispositivo de armazenamento de acesso directo, e numa altura posterior fazer retornar o programa e os dados para diferentes localizações de armazenamento principal (memória) para o reatamento da execução. A transferência do programa e dos seus dados entre a memória principal e a auxiliar pode ser realizada

de forma fragmentada, e o retorno da informação para a memória principal pode ter lugar em resposta a uma tentativa pela CPU para lhe aceder no momento em que é necessária para a execução. Estas funções podem ser realizadas sem alteração ou fiscalização do programa e dos seus dados, não requerem qualquer convenção de programação explícita no programa realocado, e não perturbam a execução do programa excepto para o atraso de tempo envolvido.

Com o apoio apropriado de um sistema operativo, a facilidade de conversão dinâmica de endereços pode ser usada para fornecer a um utilizador um sistema em que a memória aparenta ser maior do que a memória principal, que está disponível na configuração. Esta memória principal aparente é muitas vezes referida como memória virtual, e os endereços usados para designar locais na memória virtual são muitas vezes referidos como endereços virtuais. A memória virtual de um utilizador pode exceder em muito o tamanho da memória principal, que está disponível na configuração e normalmente é mantido na memória auxiliar. A memória virtual é considerada para ser composta de blocos de dados, vulgarmente chamados de páginas (também referido como segmentos e regiões). Apenas os mais recentemente referidos como páginas da memória virtual são designados para ocupar blocos da memória principal física. Como os utilizadores se referem a páginas de memória virtual que não aparecem na memória principal, eles são trazidos para substituir passos no seio da memória principal que são menos prováveis de serem necessários. Em alguns casos, a

memória virtual é designada para memória principal, por um período longo de tempo (ou permanentemente), independentemente de se a memória é referenciada. A troca de páginas de armazenagem pode ser realizada pelo sistema operativo sem o conhecimento do utilizador.

Os programas usam endereços (ou endereços virtuais) para aceder à memória virtual. O programa pode ir buscar instruções de memória virtual ou de dados de carga ou de dados de memória virtual usando endereços virtuais. Os endereços virtuais associados com um intervalo de memória virtual definem um espaço de endereço. Com o suporte apropriado de um sistema operativo, a facilidade de conversão dinâmica de endereços pode ser usada para fornecer um certo número de espaços de endereços. Estes espaços de endereços podem ser usados para fornecer graus de isolamento entre utilizadores. Tal suporte pode consistir em espaço de endereço completamente diferente para cada utilizador, proporcionando assim um isolamento completo, ou pode ser fornecida uma área partilhada através do mapeamento de uma fracção de cada espaço de endereço para uma única área de memória comum. Para além disso, são fornecidas instruções que permitem um programa semi-privilegiado para aceder a mais do que um tal espaço de endereço.

A conversão dinâmica de endereço proporciona a conversão de endereços virtuais a partir de múltiplos diferentes espaços de endereços. Num processador do Sistema

z IBM®, por exemplo, esses espaços de endereços são chamados de espaço de endereço principal, espaço de endereço secundário e de espaços de endereços de especificado Registador de Acesso. Um programa privilegiado também pode causar o espaço de endereço de página principal para ser acedido. A conversão dinâmica de endereço pode ser especificada para instruções e endereços de dados gerados pela CPU.

Como é comum na tecnologia, DAT é realizada pela utilização de fracções sucessivas do endereço virtual como índices para seleccionar as entradas numa série de tabelas de conversão (por exemplo, primeira região, segunda região, terceira região, segmento e tabelas de páginas). Cada entrada na tabela intermédia, se marcada como válida, contém a origem, o deslocamento e o comprimento da tabela seguinte de nível inferior, a qual é então indexada pela fracção seguinte do endereço virtual, até que seja atingida a entrada de uma "folha", contendo um endereço de trama real ou absoluto. A fracção remanescente do endereço virtual é em seguida, utilizada como um índice de byte no seio daquela trama para completar o resultado da conversão.

A virtualização é usada para melhorar a eficiência e flexibilidade em ambientes de computação. Prioritariamente à virtualização, um sistema operativo único normalmente correu numa máquina. Num ambiente virtualizado, um programa hipervisor ou de hospedeiro está no controlo de recursos da máquina. Este programa

hospedeiro cria múltiplas máquinas virtuais, contentores que em separado, casos independentes do sistema operativo, chamados de convidados, podem correr, partilhando recursos, tais como processadores e memória sob o controlo do programa hospedeiro.

Num ambiente paginável de convidado (máquina virtual), por exemplo, processador do Sistema z IBM®, a conversão dinâmica de endereços ocorre em dois níveis: i) uma página virtual de convidado que é apoiada por tramas reais do hospedeiro, e ii) essas tramas de convidados por sua vez serem representadas como memória virtual do hospedeiro, divididas em páginas virtuais do hospedeiro que são apoiadas por tramas reais de hospedeiro. Uma vez que a conversão de endereço é gerida de forma independente pelo convidado e pelo hospedeiro, uma trama de convidado de qualquer tamanho pode ser mapeada no seio da área virtual de hospedeiro composta de páginas de qualquer tamanho. Assim, uma trama de convidado pode consistir de uma página de hospedeiro, em muitas páginas de hospedeiro (trama de convidado ampla em páginas de hospedeiro pequenas), ou uma fracção de uma página de hospedeiro (tramas de convidados pequenas numa página de hospedeiro grande). A memória pode ser gerida de modo mais eficiente, e a conversão por tabelas numa memória (TLB) ("Translation Lookaside Buffer (TLB)" - "Memória tampão com tabelas para a Conversão" (TLB)) na máquina pode ser usada de modo mais eficiente, quando o hospedeiro utiliza o mesmo tamanho de página que a trama de convidado que ela apoia. Assim, por exemplo, uma

trama de convidado com 1Megabyte é tratada como uma unidade pelo convidado, e deve ser apoiado por um hospedeiro com uma trama de 1Megabyte, ao invés de 4 tramas separadamente paginadas de 256 Kilobyte. Isto permite que uma única entrada TLB mapeie o megabyte inteiro do endereço virtual de convidado para fazer corresponder ao endereço absoluto de hospedeiro.

A fim de que o tamanho da página de hospedeiro esteja de acordo com o tamanho da trama de convidado, o hospedeiro deve ser capaz de determinar qual o tamanho das tramas que o convidado tem a intenção de usar nas diferentes áreas da memória de convidado. Em alguns casos, o convidado pode utilizar uma instrução de gestão de trama que indica o tamanho de trama de convidado pretendido, e manipular de que instrução do firmware ou do hospedeiro pode então proporcionar uma trama de hospedeiro em conformidade de tamanho para apoiar a trama de convidado. No entanto, se o convidado não usar essa instrução no momento do desdobramento, ou se posteriormente muda aquele tamanho da trama, os tamanhos de hospedeiro e o de convidado deixam de estar em conformidade. Em particular, se o hospedeiro tem paginada uma fracção de memória do convidado e o convidado em seguida, faz referência a ela, segue-se uma excepção de conversão do hospedeiro, de modo que o hospedeiro tem a oportunidade de proporcionar memória de apoio com o conteúdo de trama de convidado preferido. Esta interrupção oferece uma oportunidade adicional para o

hospedeiro atribuir uma trama em conformidade de tamanho para a trama de convidado.

O que é necessário é uma facilidade melhorada de conversão dinâmica de endereços que proporcione funcionalidade adicional, até agora desconhecida para esta tecnologia, a qual de modo efectivo e eficiente informa o processador do hospedeiro para alocar uma trama de tamanho adequado com a qual apoie em resposta a trama de convidado para saber se a interrupção foi causada por execução numa configuração do hospedeiro ou de convidado e, se numa configuração de convidado, saber se a interrupção diz respeito a uma trama de convidado grande ou pequena identificada por uma entrada de folha de tabela de DAT de convidado, ou a uma trama de convidado referenciada por algum outro caminho.

A US 2004/024953 A1 descreve um método para a utilização de uma pluralidade de espaços de memória virtual para proporcionar a compatibilidade binária eficiente entre uma pluralidade de arquitecturas de fonte e uma arquitectura de destino único.

SUMÁRIO DA INVENÇÃO

O que é proporcionado é um método como o reivindicado na reivindicação 1 e o correspondente sistema e produto de programa de computador, proporcionando um qualificador de excepção de conversão para uma facilidade de conversão dinâmica de endereços a qual converte um

endereço virtual num endereço real ou absoluto de um bloco de dados na memória principal de um sistema de computador que tem uma arquitectura de máquina.

Num exemplo próprio de modelo de realização, é obtido um endereço virtual a ser convertido. É obtido um endereço de origem de uma primeira tabela de conversão de uma hierarquia de tabelas de conversão utilizadas na conversão. A hierarquia de tabelas de conversão consiste em uma ou mais de uma tabela de primeira região, uma tabela de segunda região, uma tabela de terceira região, uma tabela de segmento e, opcionalmente, uma tabela de página. A conversão dinâmica de endereços do endereço virtual num endereço real ou absoluto de um bloco de dados, residente na memória principal. Se a conversão não puder ser completada, por exemplo, se uma das entradas de tabela necessárias para a conversão é marcada como inválida, em seguida ocorre uma interrupção com excepção de conversão. Em resposta a um evento de interrupção com excepção de conversão tendo ocorrido durante a conversão do endereço virtual, os bits são armazenados num qualificador de excepção de conversão para indicar que a excepção de conversão foi uma de uma excepção de DAT de hospedeiro a ter ocorrido enquanto se executava um programa do hospedeiro, e uma excepção de DAT de hospedeiro a ter ocorrido durante a execução de um programa de convidado. O campo TXQ também é capaz de indicar que a excepção de conversão foi uma excepção de DAT de convidado a ter ocorrido durante a execução de um programa de convidado;

que a exceção de DAT de hospedeiro dizia respeito a um endereço derivado de uma entrada da tabela de folha de convidado; que a exceção de DAT de hospedeiro dizia respeito a um endereço derivado de um endereço real de trama de página de convidado, e que a exceção de DAT de hospedeiro dizia respeito a um endereço derivado de um endereço absoluto de trama de segmento de convidado. O campo TXQ é para além disso capaz de indicar um tamanho de uma trama de convidado para a qual a exceção de DAT de convidado diz respeito e que um tamanho de trama maior ou menor é necessário, de modo a que o hospedeiro ou o firmware possa fornecer um tamanho adequado de trama de hospedeiro com a qual apoia a trama de convidado. São proporcionados outros modelos de realização.

A invenção será em seguida descrita em ligação com determinados modelos de realização ilustrados.

BREVE DESCRIÇÃO DOS DESENHOS

As figuras que acompanham onde números de referência semelhantes dizem respeito a elementos idênticos ou funcionalmente similares ao longo de todas as vistas independentes e as quais, juntamente com a descrição detalhada que se segue são incorporadas nela e formam parte da especificação, e servem para melhor ilustrar os vários modelos de realização e para explicar os vários princípios e vantagens, todos em concordância com a presente invenção.

A **FIG. 1** ilustra um modelo de realização de um sistema de computador hospedeiro da anterior tecnologia em que será realizada a conversão dinâmica melhorada de endereços de convidado e de hospedeiro;

A **FIG. 2** proporciona um exemplo de um sistema de computador hospedeiro emulado da anterior tecnologia que emula o sistema de computador hospedeiro de uma arquitectura de hospedeiro;

A **FIG. 3** ilustra um modelo de realização da anterior tecnologia de como a palavra de estado do programa é usada para determinar a ASCE efectiva para a conversão dinâmica de endereço do endereço virtual;

A **FIG. 4** ilustra um modelo de realização da anterior tecnologia em que a ASCE efectiva determinada na **FIG. 3** é usada para determinar a tabela mais elevada de conversão na hierarquia de tabelas de conversão utilizadas na conversão do endereço virtual;

A **FIG. 5A** ilustra um modelo de realização do processo de conversão dinâmica de endereço de um endereço virtual utilizando uma hierarquia de tabelas de conversão para o nível de tabela de segmento;

A **FIG. 5B** ilustra uma continuação da conversão dinâmica de endereço da **FIG. 5A**, em que o controlo de formato (FC) da Entrada de Tabela de Segmento (STE) é zero;

A **FIG. 5C** ilustra uma continuação da conversão dinâmica de endereço da **FIG. 5A**, em que o controlo de formato (FC) da Entrada da Tabela de Segmento (STE) é um;

A **FIG. 6** ilustra um diagrama de fluxo de um modelo de realização da conversão dinâmica de endereços melhorada (EDAT) no nível de convidado para obter um campo de controlo de formato numa entrada da tabela de segmento;

A **FIG. 7** ilustra uma continuação do diagrama de fluxo do nodo 630 da **FIG. 6**, quando o controlo de formato STE de convidado é zero;

A **FIG. 8** ilustra uma continuação do diagrama de fluxo do nodo 632 da **FIG. 6**, quando o controlo de formato STE de convidado é um;

A **FIG. 9** ilustra um diagrama de fluxo de um modelo de realização de EDAT no nível de hospedeiro, o qual pode ser chamado a partir do processo de EDAT de convidado, para se obter um campo de controlo de formato numa entrada de tabela de segmento de hospedeiro;

A **FIG. 10** ilustra uma continuação do diagrama de fluxo a partir do nodo 928 da **FIG. 9**, quando o controlo de formato STE de hospedeiro é zero;

A **FIG. 11** ilustra uma continuação do diagrama de fluxo a partir do nodo 930 da **FIG. 9**, quando o controlo de formato STE de hospedeiro é um;

A **FIG. 12** ilustra o conteúdo do registador de controlo 1;

A **FIG. 13** ilustra o conteúdo de registador de controlo 7;

A **FIG. 14** ilustra o conteúdo de registador de controlo 13;

A **FIG. 15** ilustra o conteúdo da palavra de estado do programa;

A **FIG. 16** ilustra o conteúdo do prefixo de registador;

A **FIG. 17** ilustra a relação entre os endereços real e absoluto;

A **FIG. 18** ilustra o formato do endereço virtual;

A **FIG. 19** ilustra o formato do endereço virtual;

A **FIG. 20** ilustra os formatos das entradas da tabela de região;

A **FIG. 21** ilustra o formato de uma entrada conseguida a partir de uma tabela de segmento;

A **FIG. 22** ilustra o formato de uma entrada da tabela de segmento, e

A **FIG. 23** ilustra o formato de uma entrada conseguida a partir de uma tabela de página.

DESCRIÇÃO DETALHADA

Deve ser entendido que as alegações na especificação do presente pedido não fazem necessariamente limitar qualquer uma das várias invenções reivindicadas. Para além disso, algumas alegações podem aplicar-se a algumas realizações inventivas, mas não para outras. Salvo indicação em contrário, elementos singulares podem estar no plural e vice-versa, sem perda de generalidade.

Um vulgar especialista na tecnologia estará prontamente familiarizado com armazenagem de endereçamentos num ambiente de computação e utilizando bits num registador ou campo de endereço para indicar os estados diferenciados e actuando sobre esses estados. Para além disso, um vulgar especialista na tecnologia será conhecedor das tecnologias de software de computador e conhecedor do funcionamento e das inter-relações entre os componentes dos sistemas de computadores e suficiente para implementar os ensinamentos do presente documento nos seus próprios ambientes de computação sem excessiva experimentação.

VISÃO GERAL

O que é fornecido é um exemplo de um modelo de realização de uma facilidade de uma melhorada conversão dinâmica de endereços (DAT) na instalação. Quando a facilidade de melhorada DAT é instalada e activada, a conversão de DAT pode produzir quer um endereço real de trama de página ou um endereço absoluto de trama de segmento, determinado pelo controlo de formato de entrada de tabela de segmento (STE) na entrada da tabela de segmento. Como é aqui usado, o termo "DAT melhorada aplicável" ("enhanced DAT applies") significa que todas as seguintes condições são verdadeiras: 1) A facilidade de EDAT está instalada; 2) A facilidade de EDAT é activada através de registador de controlo 0 (CR0) bit 40, e, 3) O endereço é convertido por meio de entradas de tabelas de DAT.

Quando a DAT melhorada é aplicável, está disponível a seguinte função adicional no processo de DAT:

- Um bit de protecção de DAT é adicionado nas entradas da tabela de região, proporcionando função similar aos bits de protecção de DAT nas entradas das tabelas de segmento e de página.
- Um controlo de formato STE é adicionado à entrada da tabela de segmento. Quando o controlo de formato STE é zero, o procedimento de DAT como está actualmente definido, excepto que uma mudança de gravação de substituição na entrada da tabela de página indicar se o bit de fixação de mudança pode ser ignorado para a página.

- Quando o controlo de formato STE é um, a entrada da tabela de segmento também contém o seguinte:
- Um endereço absoluto de trama de segmento (em vez de uma origem de tabela de página), especificando o local de armazenamento absoluta do bloco de 1 Megabyte.
- Os bits de controlo de acesso e um bit de protecção de busca que opcionalmente pode ser utilizado em vez de os bits correspondentes nas chaves de memória individual de segmento.
- Um bit que determina a validade dos bits de controlo de acesso e o bit de protecção de busca na entrada da tabela de segmento.
- Uma mudança de gravação de substituição que indica se o bit de fixação de mudança pode ser contornado nas chaves de memória individual de segmento.
- Um qualificador de excepção de conversão (TXQ) é armazenado quando ocorre uma interrupção de excepção de DAT, para proporcionar os detalhes quanto ao ambiente de execução (de hospedeiro ou de convidado) no qual a excepção ocorreu, e a fonte do endereço que estava a ser convertida.

Sistema de Computador Hospedeiro

Um sistema emulado inclui um programa emulador que emula um sistema de computador que pode proporcionar tanto a arquitectura de um hospedeiro e uma capacidade interpretativa de execução. A memória do sistema emulado pode conter o hospedeiro e os convidados pagináveis. Um programa de hospedeiro em execução numa arquitectura de hospedeiro emulado pode Iniciar a execução de um programa

Interpretativo de convidado, que em seguida executa sob emulação da facilidade de execução interpretativa. Um próprio exemplo de uma facilidade de execução interpretativa e a instrução de Início de Execução Interpretativa (SIE) para a execução de convidados pagináveis é descrita na "Arquitetura Estendida do System/370 IBM[®]", IBM[®] Pub. No. SA22-7095 (1985).

A referência está agora a ser feita para a **FIG. 1** que ilustra um próprio modelo de realização de um computador hospedeiro do sistema 100 em que no convidado e no hospedeiro será realizada a conversão dinâmica de endereço melhorada. Ambiente de computação de hospedeiro 100 é preferencialmente baseado na z/Arquitetura[®] oferecido pela International Business Machines Corporation (IBM), Armonk, Nova York. A z/Architecture[®] é de modo mais completo descrito em: "z/Architecture[®] Principles of Operation" IBM[®] Pub. No. SA22-7832-05, 6^a Edição, (Abril 2007). Ambientes de computação baseados na z/Architecture[®] incluem, por exemplo, eServer e zSeries[®], ambos da IBM[®].

O ambiente de computação **100** inclui um complexo processador central (CPC) **102** proporcionando um suporte de da máquina virtual. O CPC **102** inclui, por exemplo, uma ou mais máquinas virtuais **104**, um ou mais processadores **106**, pelo menos, um hospedeiro **108** (por exemplo, um programa de controlo, tal como um hipervisor), e um subsistema de input/output **110**. O suporte de máquina virtual do CPC proporciona a capacidade de operar um grande número de

máquinas virtuais, cada uma capaz de acolher um sistema operacional convidado **112**, tal como o Linux.

Cada máquina virtual é capaz de funcionar como um sistema separado. Isto é, cada máquina virtual pode ser reiniciada de modo independente, o hospedeiro contém um sistema operacional convidado, e opera com programas diferentes. Um sistema operacional ou programa aplicativo em execução numa máquina virtual parece ter acesso a um sistema total e completo, mas na realidade, apenas uma parte dele está disponível.

Neste exemplo particular, o modelo de máquinas virtuais é um modelo $V = V$, no qual a memória de uma máquina virtual é apoiada pela memória virtual, em vez da memória real. Cada máquina virtual tem um espaço de memória linear virtual. Os recursos físicos são propriedade do hospedeiro, e os recursos físicos partilhados são enviados pelo hospedeiro para os sistemas operacionais convidados, conforme necessário, para atender às procuras de processamento. Este modelo de máquina virtual $V = V$ assume que as interacções entre os sistemas operacionais convidados e os recursos físicos partilhados de máquina são controlados pelo hospedeiro, uma vez que o grande número de convidados impede tipicamente o hospedeiro de simplesmente de fazer a partição e a atribuição dos recursos de hardware para os convidados configurados. Um próprio ou mais aspectos do modelo $V = V$ são de modo mais completo

descritos em: "Z/VM: Running Guest Operating Systems", IBM® Pub No. SC24-5997-02, (2001).

Os processadores centrais **106** são recursos do processador físico que são atribuíveis a uma máquina virtual. Por exemplo, a máquina virtual **104** inclui um ou mais processadores virtuais, cada um dos quais representa a totalidade ou parte de um recurso do processador físico que pode ser dinamicamente alocado na máquina virtual. As máquinas virtuais são geridas pelo hospedeiro. O hospedeiro pode ser implementado em execução de microcódigo no próprio ou em mais processadores ou ser parte de um sistema operativo de hospedeiro de execução na máquina. Num próprio exemplo, o hospedeiro é um hipervisor VM, tal como Z/VM® disponibilizado pela IBM®. Um próprio modelo de realização de Z/VM® é descrito de modo mais completo em: "Z/VM: General Information Manual", IBM® Pub No. CG24-5991-04, (2001).

O sistema Input/output **110** direcciona o fluxo de informação entre os dispositivos e a memória principal. Ele é acoplado ao complexo processamento central. Pode ser parte do complexo processamento central ou separado dele. O subsistema I/O alivia os processadores centrais da tarefa de comunicação em modo directo com os dispositivos I/O acoplados ao CPC e permite o processamento de dados para proceder de modo concorrente com o processamento de I/O.

Os processadores centrais **106** podem ter uma facilidade de Conversão Dinâmica de Endereço (DAT) (função ou unidade) para transformar endereços de programa (endereços virtuais) dentro de endereço real de memória. Uma facilidade de DAT geralmente inclui uma memória com uso de tabelas para conversão para o armazenamento em cache de conversões de modo a que mais tarde no acesso ao bloco de memória do computador não exige o atraso de conversão de endereço. Tipicamente uma cache é empregue entre a memória do computador e o processador. A cache pode ser hierárquica tendo uma grande cache disponível para mais do que uma CPU e menores, caches mais rápidas (nível inferior) entre a grande cache e cada CPU. Em algumas implementações as caches de nível mais baixo são divididas para proporcionar caches separadas de nível baixo para a busca de instruções e acessos de dados. Num modelo de realização, uma instrução é obtida a partir da memória por uma instrução unidade de busca através de uma cache. A instrução é descodificada numa unidade de descodificação de instrução e enviada (com outras instruções em alguns modelos de realização) para unidades de execução de instrução. Tipicamente várias unidades de execução são empregues, por exemplo uma unidade de execução aritmética, uma unidade de execução de ponto flutuante e uma unidade de execução de instrução de ramo. A instrução é executada pela unidade de execução, acedendo a operandes de registos especificados de instrução ou a memória conforme necessário. Se um operante é para ser acedido (carregado ou armazenado) a partir da memória, uma

unidade de memória de carga manipula geralmente o acesso sob o controlo da instrução a ser executada.

Num modelo de realização, a invenção pode ser levada à prática por software (por vezes referido como Código Interno Licenciado (LIC), firmware, micro-código, mili-código, código de pico e outros semelhantes, qualquer dos quais deverá ser consistente com a presente invenção). O Código de programa de software que incorpora a presente invenção é normalmente acedido pelo processador também conhecido como uma CPU (Unidade Central de Processamento) do sistema de computador a partir de meios de armazenamento de longo prazo, tais como uma unidade de CD-ROM unidade de fita ou disco rígido. O código do programa de software pode ser incorporado em qualquer um de uma variedade de meios conhecidos para utilização com um sistema de processamento de dados, tal como uma disquete, um disco rígido, ou um CD-ROM. O código pode ser distribuído em tais meios, ou pode ser distribuído para os utilizadores a partir da memória do computador ou armazenamento do próprio sistema de computador através de uma rede para outros sistemas de computador para utilização dos utilizadores de tais sistemas.

De modo alternativo, o código do programa pode ser incorporado na memória e acedido pelo processador utilizando o bus de processador. Tal código de programa inclui um sistema operativo que controla a função e interacção dos vários componentes do computador e um ou

mais programas de aplicação. O código do programa é normalmente paginado a partir dos meios de armazenamento densos onde está disponível uma elevada velocidade de memória para processamento pelo processador. As técnicas e métodos, para incorporar na memória o código de programa de software, quanto aos meios físicos, e/ou distribuição de código de software através de redes, são bem conhecidos e não serão ainda discutidos neste documento. O código do programa, quando criado e armazenado num meio tangível (com inclusão, mas não limitado a módulos de memória eletrónicos (RAM), a memória flash, a discos compactos (CDs), a DVDs, a fita magnética e a semelhantes) é muitas vezes referido como um "produto de programa de computador". O produto médio de programa de computador é normalmente legível por um circuito de processamento a ser executado pelo circuito de processamento de preferência no seio de um sistema de computador.

Um outro exemplo de um ambiente de computação para incorporar um ou mais aspectos da presente invenção é representado na **FIG. 2**. Neste exemplo, é disponibilizado um sistema de computador hospedeiro emulado **200** que emula um sistema de computador hospedeiro **202**, numa arquitectura de hospedeiro. O processador hospedeiro emulado **204** (ou processador hospedeiro virtual) é realizado através de um processador de emulação **206** tendo uma arquitectura conjunto de instruções nativas diferente da que a utilizada pelos processadores do computador hospedeiro. O sistema de computador hospedeiro emulado tem a memória **208** acessível

ao processador de emulação **206**. No exemplo de modelo de realização, a memória **208** é subdividida numa fracção de memória do computador hospedeiro **210** e numa fracção de rotinas de emulação **212**. A memória do computador hospedeiro **210** está disponível para os programas de computador hospedeiro emulado **202** e pode conter tanto um hospedeiro ou um hipervisor **214** e uma ou mais máquinas virtuais **216** executando sistemas operacionais convidados **218**, análogos aos elementos igualmente denominados na **FIG. 1**. O processador de emulação **206** executa instruções nativas de um conjunto de instruções arquitectadas de uma outra arquitectura daquela do processador emulado; as instruções nativas são obtidas, por exemplo, a partir da memória das rotinas de emulação **212**. O processador de emulação **206** pode aceder a instruções de hospedeiro para a execução a partir de um programa na memória do computador hospedeiro **210** através do emprego de uma ou mais instrução (ões) obtida numa sequência & rotina acesso/descodificação a qual pode descodificar a instrução (ões) de hospedeiro acedida para determinar uma rotina de execução de instrução nativa para emular a função da instrução de hospedeiro acedida. Uma tal instrução de hospedeiro pode ser, por exemplo, uma instrução de Início de Execução Interpretativa (SIE), pela qual o hospedeiro procura executar um programa numa máquina virtual. As rotinas de emulação **212** podem incluir suporte para esta instrução, e para a execução de uma sequência de instruções de convidado numa máquina virtual de acordo com a definição desta instrução SIE.

Outras facilidades que são definidos para a arquitectura do sistema do computador hospedeiro podem ser emuladas pelas rotinas de facilidades architectadas, incluindo facilidades, tais como por exemplo registadores de objectivo geral, controlo de registos, conversão de endereço dinâmica, e subsistema de suporte I/O e cache de processador. As rotinas de emulação podem também tirar proveito das funções disponíveis no processador de emulação **206** (tais como registos gerais e conversão dinâmica de endereços virtuais) para melhorar o desempenho das rotinas de emulação. Hardware especial e motores fora de carga podem também ser previstos para auxiliar o processador na emulação da função do computador hospedeiro. O computador hospedeiro, num modelo de realização, está em comunicação com uma variedade de meios de armazenamento **220** conhecidos, tais como, por exemplo, uma disquete, um disco rígido, ou um CD-ROM. O código de programa de software pode ser distribuído em tais meios ou podem ser distribuídos aos utilizadores através de uma rede **222**.

Processador de Computador e Registadores

Num modelo de realização, a funcionalidade de instruções do programa de uma CPU comunica com uma pluralidade de registadores sobre um bus de comunicação. O bus de comunicação pode ser interno ou externo relativamente à CPU. Alguns registadores podem ser somente de leitura. Outro hardware e/ou software pode também ler/escrever para um ou mais dos registadores acessíveis pela CPU. Um código de operação (opcódigo) de instruções

determina qual o tipo de registador que é para ser utilizado em qualquer operação de instrução de uma máquina particular.

Registadores Gerais

As instruções podem designar informação num ou mais dos 16 registadores gerais. Os registadores gerais podem ser usados como registadores de endereços de base e registadores de índice em endereço aritmético e como acumuladores quando em aritmética geral e operações lógicas. Cada registador contém 64 posições de bits. Os registadores gerais são identificados pelos números de 0 a 15 e numa instrução são designados por um campo R de quatro bits. Algumas instruções fornecem registadores gerais de múltiplo endereçamento que têm vários campos R. Para algumas instruções, está implicado o uso de um registador geral específico em vez do designado explicitamente por um campo R da instrução. Para algumas operações, ou os bits de 32 a 63 ou os bits de 0 a 63 de dois registadores gerais adjacentes são acoplados, proporcionando respectivamente um formato de 64-bits ou de 128-bits. Nestas operações, o programa deve designar um registador de numeração par, que contém o mais à esquerda (de ordem alta) 32 ou 64 bits. O próximo registador de maior numeração contém o mais à direita (baixa ordem) 32 ou 64 bits. Em adição com a sua utilização como acumuladores em aritmética geral e operações lógicas, 15 dos 16 registadores gerais também são usados como endereço de base e registadores de índice na geração de endereço. Nestes casos, os registadores são

designados por um campo B de quatro bits ou por campo X numa instrução. Um valor de zero nos campos B ou X especifica que nenhuma base ou índice é para ser aplicada, e, assim, o registador geral 0 não pode ser designado como contendo um endereço de base ou índice.

Registadores de Controlo

Os registadores de controlo fornecem para a manutenção e o manuseamento de informação de controlo externo a palavra de status do programa. A CPU tem 16 registadores de controlo, cada um com 64 posições de bit. As posições de bit nos registadores são atribuídas às facilidades particulares no sistema, tais como programa de gravação de evento, e são usadas tanto para especificar ou que uma operação pode ter lugar ou para fornecer informação especial requerida pela facilidade. Os registadores de controlo são identificados pelos números de 0 a 15 e são designados pelos campos R de quatro bits nas instruções CONTROLO DE CARGA (LOAD CONTROL) e CONTROLO DE ARMAZENAGEM (STORE CONTROL). Registadores de controlo múltiplo podem ser endereçados por essas instruções.

Registador de Controlo 1

O registador de controlo 1 contém o Elemento de Controlo do Espaço de Endereço Primário (PASCE) (Primary Address Space Control Element). Num modelo próprio de realização, o registador de controlo 1 tem um dos dois formatos como mostrado na fig. 12, dependendo do bit de controlo de espaço real (R) no registador:

Campos seleccionados no Elemento de Controlo do Espaço de Endereço Primário (PASCE) são alocados como se segue:

Tabela de Região Primária ou Tabela de Origem de Segmento: Bits 0 a 51 da tabela de região primária ou designação de tabela de segmento no registador de controlo 1, com 12 zeros acrescentados à direita, formam um endereço de 64 bits que designa o início da tabela de região primária ou tabela de segmento. É imprevisível se o endereço é real ou absoluto. Esta tabela é denominada a tabela de região primária ou tabela de segmento, uma vez que é usada para converter os endereços virtuais no espaço de endereço primário.

O Controlo do Espaço Real Primário (R): Se o bit 58 do registador de controlo 1 é zero, o registador contém uma designação de tabela de região ou tabela de segmento. Se o bit 58 é um, o registador contém uma designação de espaço real. Quando o bit 58 é um, um valor um do bit do segmento comum numa representação de uma entrada da tabela de segmento da memória tampão com tabelas para conversão (TLB) impede a entrada de tabela de página e a memória tampão com tabelas para conversão (TLB) faz a sua cópia que designa para ser usada na conversão de referências para o espaço de endereço primário, mesmo com uma correspondência entre a origem de testemunho de verificação no registador de controlo 1 e a entrada da origem de tabela na memória tampão com tabelas para conversão (TLB).

O Controlo de Tipo de Designação Primário (DT): Quando R é zero, o tipo de designação da tabela no registador de controlo 1 é especificado pelos bits 60 e 61 no registador , como se segue:

Bits 60 e 61	Tipo de Designação
11	Tabela-primeira-Região
10	Tabela-segunda-Região
01	Tabela-terceira-Região
00	Tabela-Segmento

Bits de controlo de Tipo Designação primária (DT)

Quando R é zero, os bits 60 e 61 devem ser 11 binário quando é feita uma tentativa de usar a PASCE para converter um endereço virtual em que o bit mais à esquerda está nas posições de bit 0 a 10 do endereço. Do mesmo modo, os bits 60 e 61 deve ser 11 ou 10 binários quando o bit mais à esquerda é um nas posições de bit 11 a 21 do endereço, e eles devem ser 11, 10 ou 01 binários, quando o bit mais à esquerda está nas posições de bit 22 a 32 do endereço. Caso contrário, é reconhecido uma excepção de tipo ASCE.

Tabela de Região Primária ou Comprimento de Tabela de Segmento (TL): Os bits 62 e 63 da designação de tabela de região primária ou designação da tabela de segmento no registador de controlo 1 especificam o comprimento da tabela de região primária ou tabela de segmento em unidades de 4.096 bytes, tornando assim o comprimento da tabela de região ou da tabela de segmento variável em múltiplos de 512 entradas. O comprimento da

tabela de região primária ou tabela de segmento, em unidades de 4.096 bytes, é mais um do que o valor TL. Os conteúdos do campo de comprimento são utilizados para determinar se a fracção do endereço virtual (RFX, RSX, RTX, ou SX) a ser convertida por meio da tabela que designa uma entrada que cai dentro da tabela.

Origem de Testemunho de Verificação de Espaço Real Primário: os bits 0 a 51 da designação de espaço real primário, no registador de controlo 1, com 12 zeros acrescentados à direita, formam um endereço de 64 bits que pode ser utilizado na formação e na utilização das entradas de memória tampão com tabelas para conversão (TLB) que proporcionam uma conversão real que iguala a virtual nas referências ao espaço de endereço primário. Embora este endereço seja utilizado apenas como um testemunho de verificação e não é usado para efectuar uma referência de armazenamento, ele ainda será um endereço válido; caso contrário, uma entrada incorrecta de memória tampão com tabelas para conversão (TLB) pode ser utilizada quando são utilizados os conteúdos do registador de controlo 1.

Os seguintes bits do registador de controlo 1 não são atribuídos e são ignorados: os bits 52, 53, e 59 se o registador contém uma designação de tabela de região ou designação de tabela de segmento, e os bits 52, 53 e 59 a 63 se o registador contém a designação de espaço real.

Registador de Controle 7

O registrador de controle 7 contém o Elemento de Controle de Espaço de Endereço Secundário (SASCE) (Secondary Address Space Control Element). Num modelo de realização, o registrador de controle 7 tem um dos dois formatos ilustrados na Fig. 13, dependendo do bit de controle de espaço real (R) no registrador.

Registador de Controle 13

O registrador de controle 13 contém o Elemento de Controle do Espaço de Endereço de Página Principal (HASCE) (Home Address Space Control Element). Num modelo de realização, o registrador de controle 13 tem um dos dois formatos como ilustrado na Fig. 14, dependendo do bit de controle de espaço real (R) no registrador.

Registadores de o Acesso

A CPU tem 16 registradores de acesso numerados de 0 a 15. Um registrador de acesso consiste em 32 bits de posição, contendo uma especificação indireta de um ASCE. Um ASCE é um parâmetro utilizado pelo mecanismo de conversão dinâmica de endereços (DAT) para converter referências a um correspondente espaço de endereço. Quando a CPU está num modo chamado de modo de acesso de registrador (controlado por bits na palavra de status do programa), um campo B de instrução, usado para especificar um endereço lógico para uma referência de operante de armazenagem, que designa um registrador de acesso, e o ASCE especificado pelo registrador de acesso é utilizado pela DAT para a referência a ser

efectuada. Para algumas instruções, é utilizado um campo R em vez de um campo B. As instruções são previstas para o carregamento e o armazenamento de conteúdos dos registadores de acesso e para a movimentação de conteúdos de um registador de acesso para um outro.

Cada um dos registadores de acesso 1 a 15 pode designar qualquer espaço de endereço, incluindo o espaço de instrução corrente (o espaço de endereço primário). O registador de acesso 0 designa o espaço de instrução primário. Quando um dos registadores de acesso 1 a 15 é utilizado para designar um espaço de endereço, a CPU determina qual o espaço de endereço que é designado pela conversão dos conteúdos do registador de acesso. Quando o registador de acesso 0 é usado para designar um espaço de endereço, a CPU trata o registador de acesso como designação do espaço de instrução primário, e não tem em consideração o conteúdo corrente do registador de acesso. Portanto, os 16 registadores de acesso podem designar, em qualquer uma momento, o espaço de instrução primário e um máximo de 15 outros espaços.

Palavra de Status do Programa (PSW) (Programa Word Status)

A palavra de status do programa inclui o endereço de instrução, o código de condição, e outra informação utilizada para controlar o sequenciamento de instruções e para determinar o estado da CPU. A palavra de status de programa activa ou em controlo é denominada de palavra de

status corrente do programa. Ela governa o programa que está a ser executado no momento corrente.

A CPU tem uma capacidade de interrupção, a qual permite que a CPU comute rapidamente para um outro programa em resposta a condições excepcionais e a estímulos externos. Quando uma interrupção ocorre, a CPU coloca a palavra status corrente do programa numa localização de de armazenagem atribuída, denominada de localização de antiga palavra de status de programa, para a classe especial de interrupção. A CPU busca uma nova palavra de status de programa a partir de uma segunda localização de armazenagem atribuída. Esta nova palavra de status de programa determina o programa seguinte a ser executado. Quando ele tiver terminado o processamento da interrupção, o programa de manuseamento da interrupção pode recarregar a antiga palavra de status de programa, tornando-a novamente como a corrente palavra de status de programa, de modo que o programa interrompido pode continuar.

Existem seis classes de interrupção: externa, I/O, máquina de verificação, programa, re-início, e chamada supervisora. Cada classe tem um par de localizações distintas de antiga palavra de status de programa e de nova palavra de status de programa atribuídas de forma permanente no armazenamento real.

Palavra de Status de Programa Corrente

A palavra de status de programa corrente na CPU contém a informação necessária para a execução de programa activo no momento. A palavra de status de programa é de 128 bits de comprimento e inclui o endereço de instrução, o código de condição, e outros campos de controlo. Em geral, a palavra de status de programa é usada para controlar o sequenciamento de instruções e de manter e indicar a maior parte do status da CPU em relação ao programa correntemente a ser executado. O Controlo adicional e o status de informação estão contidos em registadores de controlo e permanentemente atribuídas localizações de armazenagem. O status da CPU pode ser alterado pelo carregamento de uma nova palavra de status de programa ou parte de uma palavra de status de programa.

O controlo é comutado durante uma interrupção da CPU por meio do armazenamento da palavra status do programa corrente, de modo a preservar o estado da CPU, e em seguida o carregamento de uma nova palavra de status de programa. A Execução de LOAD PSW (carregar PWS) ou LOAD PSW EXTENDED (carrega PSW alargada), ou a conclusão bem sucedida da sequência de carregamento do programa inicial, introduz uma nova palavra de status de programa. O endereço de instrução é actualizado pela execução da instrução sequencial e substituído por ramos bem sucedidos. São previstas outras instruções que operam numa fracção da palavra de status de programa.

A nova ou modificada palavra de status de programa torna-se activa (ou seja, a informação introduzida no seio da palavra de status de programa corrente assume o controlo sobre a CPU) quando a interrupção ou a execução de uma instrução que muda a palavra de status de programa está concluída. A interrupção para Gravação de Evento de Programa (PER) (Program Event Recording) associada a uma instrução que muda a palavra do status de programa ocorre sob o controlo de máscara PER que é efectiva no início da operação. Os bits 0 a 7 da palavra de status de programa são referidos colectivamente como a máscara do sistema. Num modelo de realização, a palavra de status de programa tem o formato ilustrado na fig. 15.

O que se segue é um breve resumo das funções de campos seleccionados de palavras de status de programa.

Modo DAT (T): o bit 5 controla se ocorre a conversão dinâmica de endereço implícita de endereços lógicos e de instrução usados para aceder ao armazenamento. Quando o bit 5 é igual a zero, a DAT está desligada e endereços lógicos e de instrução são tratados como endereços reais. Quando o bit 5 é um, a DAT está ligada, e o mecanismo de conversão dinâmica de endereço é chamado.

Chave PSW: os bits 8 a 11 formam a chave de acesso para as referências de armazenagem pela CPU. Se a referência está sujeita à protecção controlada chave, a chave PSW é combinada com uma chave de armazenagem quando a

informação é armazenada ou quando a informação é obtida a partir de uma localização que é protegida contra buscas. No entanto, para um dos operandos de cada um dos MOVE TO PRIMARY (mover para o primário), MOVE TO SECONDARY (mover para o secundário), MOVE WITH KEY (mover com a chave), MOVE WITH SOURCE KEY (mover com a chave de fonte), e MOVE WITH DESTINATION KEY (mover com a chave de destino), é usada uma chave de acesso especificada como um operando em vez da chave PSW.

Controlo de Espaço de Endereço (AS): os bits 16 e 17, em conjunção com a Palavra de Status de Programa bit 5, controlam o modo de conversão.

Código de condição (CC): os bits 18 e 19 são os dois bits do código de condição. O código de condição é ajustado para 0, 1, 2, ou 3, dependendo do resultado obtido na execução de determinadas instruções. Mais operações aritméticas e lógicas, bem como algumas outras operações, ajustam o código de condição. A instrução BRANCH ON CONDITION (ramo na condição) pode especificar qualquer selecção de valores de código de condição como um critério para a ramificação.

Endereço de instrução: os bits 64 a 127 da palavra de status do programa são o endereço de instrução. Este endereço designa a localização do byte mais à esquerda da próxima instrução a ser executada, a menos que a CPU

esteja no estado de espera (bit 14 da palavra de status de programa é um).

Formatos & Tipos de Endereço

Para fins de armazenamento principal de endereçamento, três tipos básicos de endereços são reconhecidos: absoluto, real e virtual. Os endereços são distinguidos com base em transformações que são aplicadas ao endereço durante um acesso de armazenamento. A Conversão de endereços converte um endereço virtual num endereço real. A Prefixação converte um endereço real num endereço absoluto. Para além dos três tipos de endereços de base, são definidos tipos adicionais os quais são tratados como um ou outro dos três tipos básicos, dependendo da instrução e do modo corrente.

Endereço Absoluto

Um endereço absoluto é o endereço atribuído a uma localização de armazenamento principal. Um endereço absoluto é utilizado para um acesso de armazenamento sem quaisquer transformações executadas sobre ele. O subsistema de canal e todas as CPUs na configuração referem-se a uma localização de armazenamento principal partilhada pelo uso do mesmo endereço absoluto. A memória principal disponível é geralmente atribuída a contíguos endereços absolutos começando em 0, e os endereços são atribuídos em blocos completos de 4Kilobyte sobre limites integrais. Uma excepção é reconhecida quando é feita uma tentativa para usar um endereço absoluto num bloco que não tenha sido

atribuído a localizações físicas. Em alguns modelos, podem ser proporcionados os controles de reconfiguração de armazenamento os quais permitem que o operador altere a correspondência entre endereços absolutos e localizações físicas. No entanto, em qualquer um momento, uma localização física não está associada com mais do que um endereço absoluto. O armazenamento é referido como armazenamento absoluto consistindo de localizações de byte sequenciadas de acordo com os seus endereços absolutos.

Endereço Real

Um endereço real identifica uma localização no armazenamento real. Quando um endereço real é usado para um acesso para o armazenamento principal, é convertido, por meio de prefixação, para formar um endereço absoluto. Em qualquer instante há um endereço real para o mapeamento de endereço absoluto para cada CPU na configuração. Quando um endereço real é usado por uma CPU para aceder ao armazenamento principal, pode ser convertido para um endereço absoluto pela prefixação. A transformação específica é definida pelo valor no registrador prefixo para a CPU. O armazenamento consistindo em localizações de byte sequenciadas de acordo com os seus endereços reais é referido como o armazenamento real.

Endereço Virtual

Um endereço virtual identifica uma localização no armazenamento virtual. Quando um endereço virtual é usado para um acesso à memória principal, é convertido por meio

de conversão dinâmica de endereços, quer para um endereço real que pode estar sujeito a prefixação para formar um endereço absoluto, ou directamente para um endereço absoluto.

Endereço Virtual Primário

Um endereço virtual primário é um endereço virtual o qual é para ser convertido por meio do Elemento de Controlo do Espaço de Endereço Primário (PASCE). Os endereços lógicos são tratados como endereços virtuais primários quando no modo de espaço primário. Os endereços de instrução são tratados como endereços virtuais primários quando no modo de espaço primário, no modo espaço secundário, ou no modo registador de acesso. O endereço do primeiro operando MOVE TO PRIMARY e o endereço do segundo operando MOVE TO SECONDARY são tratados como endereços virtuais primários. Para além disso, quando um convidado paginável está em execução, a armazenagem principal (memória) que o convidado vê como armazenamento absoluto é representado no espaço de endereço primário de hospedeiro, ou seja, endereços absolutos de convidado são tratados como endereços virtuais primários de hospedeiro.

Endereço Virtual Secundário

Um endereço virtual secundário é um endereço virtual que é para ser convertido por meio do Elemento de Controlo do Espaço de Endereço Secundário (SASCE). Os endereços lógicos são tratados como endereços virtuais secundários quando no modo de espaço secundário. O endereço

de segundo operando MOVE TO PRIMARY e o endereço do primeiro operando MOVE TO SECONDARY são tratados como endereços virtuais secundários.

Endereço Virtual Especificado AR

Um endereço virtual especificado AR é um endereço virtual que é para ser convertido por meio de um Elemento de Controlo do Espaço de Endereço especificado - Registador de Acesso (Access Register-specified Address Space Control Element). Os endereços lógicos são tratados como endereços especificados AR quando no modo de registador de acesso.

Endereço Virtual de Página principal

Um endereço virtual de página principal é um endereço virtual o qual é para ser convertido por meio do Elemento de Controlo do Espaço de Endereço de Página principal (HASCE) (Home Address Space Control Element). Os endereços lógicos e os endereços de instrução são tratados como endereços virtuais de página principal quando no modo de espaço de página principal.

Endereço de Instrução

Os endereços utilizados para buscar instruções da armazenagem são denominados de endereços de instrução. Endereços de instrução são tratados como endereços reais no modo real, como endereços virtuais primários no modo espaço primário, no modo espaço secundário, ou no modo registador de acesso, e como endereços virtuais de página principal no modo de espaço página principal. O endereço de instrução na

palavra status de programa corrente e o endereço de destino de EXECUTE são endereços de instrução.

Endereço Efectivo

Em algumas situações, é conveniente o uso do termo "endereço efectivo." Um endereço efectivo é o endereço que existe antes de qualquer transformação por conversão dinâmica de endereços ou de ser executada a prefixação. Um endereço efectivo pode ser especificado directamente num registador ou pode resultar de aritmética de endereço. A aritmética de endereço é a adição da base e deslocamento ou da base, do índice e deslocamento.

Prefixação

A prefixação proporciona a possibilidade de atribuir o intervalo de endereços reais 0 a 8191 para um bloco diferente em armazenamento absoluto para cada CPU, permitindo assim mais do que uma CPU partilhar o armazenamento principal para operar simultaneamente com um mínimo de interferência, especialmente no processamento de interrupções. A prefixação dá origem a endereços reais na faixa de 0 a 8191 a corresponder um-para-um para o bloco de 8K byte de endereços absolutos (a área de prefixo), identificado pelo valor das posições de bit de 0 a 50 do registador de prefixo para a CPU, e o bloco de endereços reais identificados por aquele valor no registador de prefixo para corresponder um-para-um para endereços absolutos 0 a 8191. Os endereços reais remanescentes são os mesmos que os endereços absolutos correspondentes. Esta

transformação permite a cada CPU aceder a todo o armazenamento principal, incluindo os primeiros 8K bytes e as localizações designadas pelos registadores de prefixo de outras CPUs.

O prefixo é uma quantidade de 51-bit contido nas posições de bits 0 a 50 do registador de prefixo. Num modelo de realização, o registador de prefixo tem o formato tal como ilustrado na fig. 16.

Formato do Registador de Prefixo

Quando a prefixação é aplicada, o endereço real é transformado num endereço absoluto pela utilização de uma das seguintes regras, dependendo dos bits 0 a 50 do endereço real:

1. Bits 0 a 50 do endereço, se todos zeros, são substituídos com bits 0 a 50 do prefixo.
2. Bits 0 a 50 do endereço, se for igual aos bits 0 a 50 do prefixo, são substituídos por zeros.
3. Bits 0 a 50 do endereço, se não todos zeros e não for igual aos bits 0 a 50 do prefixo, permanece inalterado.

Somente o endereço apresentado para o armazenamento é convertido por prefixação. O conteúdo da fonte do endereço permanece inalterado.

A distinção entre os endereços reais e absoluto é feita mesmo quando o registador de prefixo contém todos

zeros, caso em que um endereço real e do seu endereço correspondente absoluto são idênticos.

Relacionamento entre Endereços Reais e Absoluto

A relação entre os endereços reais e absoluto é representada graficamente na fig. 17.

Espaços de Endereço

Um espaço de endereço é uma sequência consecutiva de números inteiros (endereços virtuais); juntos com os parâmetros de transformação específicos que permitem que cada número seja associado com uma localização de byte no armazenamento. A sequência começa em zero e procede-se da esquerda para a direita.

Quando um endereço virtual é usado por uma CPU para aceder ao armazenamento principal, é primeiro convertido, por meio de conversão dinâmica de endereços (DAT), para um endereço real ou absoluto. Um endereço real pode ainda ser submetido a prefixação para formar um endereço absoluto. A DAT pode utilizar como parâmetros de transformação uma tabela de primeira região, uma tabela de segunda região, uma tabela de terceira região, uma tabela de segmento, e uma tabela de página. A designação (origem e comprimento) da tabela de nível mais elevado para um espaço de endereço específico é denominado um Elemento Controlo do Espaço de Endereço (ASCE), e é encontrado pela DAT para uso num registador de controlo ou conforme especificado por um registador de acesso. De modo alternativo, o ASCE para um

espaço de endereço pode ser uma designação de espaço real, o qual indica que a DAT é para simplesmente converter o endereço virtual tratando-o como um endereço real e sem utilizar qualquer tabela.

A DAT utiliza, em momentos diferentes, a ASCE em diferentes registadores de controlo, ou especificado pelos registadores de acesso. A escolha é determinada pelo modo de conversão especificado na palavra de status de programa corrente. Quatro modos de conversão estão disponíveis: modo de espaço primário, modo de espaço secundário, modo de registador de acesso, e modo de espaço de página principal. Os diferentes espaços de endereços são endereçáveis dependendo do modo de conversão.

Em qualquer momento, quando a CPU está no modo de espaço primário ou modo de espaço secundário, a CPU pode converter endereços virtuais pertencentes a dois espaços de endereços - o espaço de endereço primário e o espaço de endereço secundário. Em qualquer momento, quando a CPU está no modo de registador de acesso, pode converter os endereços virtuais até 16 espaços de endereços - o espaço de endereço primário e até 15 espaços de endereços especificados AR. A qualquer momento em que a CPU está no modo de espaço de página principal, pode converter os endereços virtuais do espaço de endereço de página principal.

O espaço de endereço primário é identificado como tal porque consiste de endereços virtuais primários, que são convertidos por meio do Elemento de Controlo de Espaço de Endereços Primário (PASCE). Do mesmo modo, o espaço de endereço secundário é constituído por endereços virtuais secundários convertidos por meio do Elemento de Controlo de Espaço de Endereços Secundário (SASCE). Os espaços de endereços especificados AR consistem em endereços virtuais especificados AR convertidos por meio de Elemento de Controlo de Espaço de Endereço especificado-Registador de Acesso (ASCE especificado AR), e o espaço de endereço de página principal é constituído por endereços virtuais de página principal convertidos por meio do Elemento de Controlo de Espaço de Endereço de Página principal (HASCE). Os ASCEs primários e secundários estão nos registadores de controlo 1 e 7, respectivamente. Os ASCEs especificados AR podem estar nos registadores de controlo 1 e 7, ou em entradas de tabela de entradas da tabela denominadas entradas de tabela secundária ASN. O HASCE está no registador de controlo 13.

Conversão dinâmica de endereços

A conversão dinâmica é o processo de conversão de um endereço virtual (durante uma referência de armazenamento, por exemplo) para dentro do endereço de memória principal correspondente (endereço real ou endereço absoluto no modelo de realização). O endereço virtual pode ser um endereço virtual primário, um endereço virtual secundário, um endereço virtual especificado de Registador

de Acesso, ou um endereço virtual de página principal. Estes endereços são convertidos por meio do PASCE, SASCE, AR-especificado ASCE, ou o HASCE, respectivamente. Após a selecção do ASCE apropriado, o processo de conversão é o mesmo para todos os quatro tipos de endereço virtual.

Modo de Conversão de Endereçamento

Um endereço efectivo é o endereço (endereço virtual) que existe antes de qualquer transformação por conversão dinâmica de endereços ou é efectuada a prefixação. Os três bits na palavra de status de programa que controla a conversão dinâmica de endereços são o bit 5, bit de modo DAT, e os bits 16 e 17, os bits de controlo de espaço de endereço. Quando o bit de modo DAT é zero, então a DAT está desligada, e a CPU está no modo de real. Quando o bit de modo DAT é um, então a DAT está ligada, e a CPU está no modo de conversão designado pelos bits de controlo de espaço de endereço: 00 binário designa o modo de espaço primário, 01 binário designa o modo de registador de acesso, 10 binário designa o modo de espaço secundário, e 11 binário designa o modo de espaço de página principal. Os vários modos são mostrados em baixo, juntamente com o manuseamento de endereços em cada modo.

PSW Bit			DAT	Modo	Tratamento de Endereços	
5	16	17			Endereços de Instrução	Endereços Lógicos
0	0	0	Off	Modo Real	Real	Real
0	0	1	Off	Modo Real	Real	Real
0	1	0	Off	Modo Real	Real	Real
0	1	1	Off	Modo Real	Real	Real
1	0	0	On	Modo Espaço-Primário	Virtual Primário	Virtual Primário
1	0	1	On	Modo Registrador-Acesso	Virtual Primário	Virtual AR-especificado
1	1	0	On	Modo Espaço-Secundário	Virtual Primário	Virtual Secundário
1	1	1	On	Modo Espaço-Página Principal	Página Principal virtual	Página Principal virtual

Modos de Conversão

A Palavra de Status de Programa é uma palavra de 128 bits que, em parte, dispõe de 2 bits que indicam o modo de endereçamento. Num modelo de realização, o bit 31 é o bit de Modo de Endereçamento Estendido (EA) e o bit 32 é o bit de Modo de Endereçamento de Base (BA). Estes dois bits indicam o tamanho de endereços. O estado de cada um destes dois bits é (1 ou 0) binário. Se o bit de EA é 0 e o bit de BA é 0 então é indicado o endereçamento de 24 bits. Se é indicado endereçamento de 24 bits, os bits 40 a 63 de palavra de 64 bits (uma entidade de 64-bit é comumente denominada de palavra-dupla (doubleword)) é onde o endereço está localizado. Sempre que o endereço de instrução ocupa os segundos 64 bits de uma entidade de 128 bits (uma palavra-quádrupla (quadword)), e as posições de bits na palavra do status de programa são como se segue. No modo de 24 bits, o endereço de instrução está nos bits 104 a 127 da palavra de status de programa. No modo de 31 bits, o endereço de instrução está nos bits 97 a 127 da palavra de

status de programa. No modo de 64 bits, o endereço de instrução está nos bits 64 a 127 da palavra de status de programa. Se o bit de EA é 0 e o bit de BA é 1, então é indicado o endereçamento de 31 bits. A palavra apropriada de 64 bits contém um endereço de 31 bits, localizado nas posições de bit 33 a 63. Se o bit EA é 1 e o bit de BA é 1, então os bits 0 a 63 contêm o endereço, que são os 64-bits completo, de uma palavra de 64-bits. Caso contrário, é indicada uma condição de exceção. Uma vez que o modo de endereçamento tenha sido obtido, o ASCE necessita de ser determinado.

Elemento de Controlo do Espaço de Endereço (ASCE)

A referência está agora a ser feita na **FIG. 3** a qual ilustra um modelo de realização da forma como a Palavra de Status de Programa é usada para determinar o efectivo Elemento de Controlo do Espaço de Endereços (ASCE) para conversão dinâmica de endereços do endereço virtual. O ASCE pode especificar, por exemplo, um espaço de endereço de 2 Gigabytes (Giga = 2^{30}). Ou, ele pode especificar, por exemplo, 4 Terabytes (Tera = 2^{40}), 8 Petabytes (Peta = 2^{50}) ou a 16 Exabytes (Exa = 2^{60}) como espaço de endereço. Ou, ele pode especificar uma designação de espaço real. Uma designação espaço real faz com que o endereço virtual venha a ser tratado como uma endereço real no armazenamento sem fazer referência a uma ou mais tabelas de conversão de endereços.

A Palavra de Status de Programa **300** contém uma conversão de bits (T) (translation) **302** e Espaço de Endereço de bits (AS) (Address Space) **304**. Na **306**, se o bit de conversão (T) é zero, então o endereço é um endereço real **326**. Se, em **308**, o espaço de endereço (AS) é igual a zero (00 binário), então, o ASCE efectivo para este endereço virtual é o Elemento de Controlo de Espaço de Endereço Primário (PASCE) **310**. Se, na **312**, o Espaço de Endereço (AS) é igual a um (01 binário), então, o ASCE efectivo é o Elemento Controlo de Espaço de Endereço especificado de Registador **314**. Se, na **316**, um Espaço de Endereço (AS) é igual a dois (10 binário), então, o ASCE efectivo é o Elemento de Controlo de Espaço de Endereço Secundário (SASCE) **318**. Caso contrário, o espaço de endereço (AS) é igual a três (11 binário) e o ASCE efectivo é o Elemento Controlo do Espaço de Endereço de Página principal (HASCE) **322**.

Após a selecção do ASCE efectivo, o processo de conversão dinâmica de endereço é, de preferência o mesmo para todos os quatro tipos de endereços virtuais.

A designação da tabela segmento ou designação da tabela de região faz com que a conversão a ser executada por meio de tabelas estabelecidas pelo sistema operativo em armazenamento real ou absoluto. Uma designação de espaço real faz com que o endereço virtual venha a ser simplesmente tratado como um endereço real, sem o uso de tabelas no armazenamento.

No processo de conversão quando se utiliza uma designação da tabela de segmento ou uma designação de tabela de região, são reconhecidas três tipos de unidades de informação - as regiões, os segmentos e as páginas. A região é um bloco de endereços virtuais sequenciais abrangendo 2Gigabytes e começando num limite de 2Gigabyte. Um segmento é um bloco de endereços virtuais sequenciais abrangendo 1Megabytes e começando num limite de 1 Megabyte. A página é um bloco de endereços virtuais sequenciais abrangendo 4Kilobytes e começando num limite de 4Kilobyte.

Formato de Endereço Virtual

A conversão de um endereço virtual pode envolver referência a uma pluralidade de tabelas de conversão de entre uma hierarquia de tabelas de conversão para obter um endereço real ou absoluto. O endereço real pode ainda ser sujeito a uma operação de prefixação para formar um endereço absoluto. O endereço virtual contém índices para as entradas das tabelas de conversão na hierarquia de tabelas de conversão. O endereço virtual, por conseguinte, é dividido em quatro campos principais. Os bits 0 a 32 são denominados de índice de região (RX), os bits 33 a 43 são denominados de índice de segmento (SX), os bits 44 a 51 são denominados de índice de página (PX), e os bits 52 a 63 são denominados de índice de byte (BX). Num modelo de realização, o endereço virtual tem o formato tal como ilustrado na **FIG. 18**.

Conforme determinado pelo seu ASCE, um espaço de endereço virtual pode ser um espaço de 2 Gigabyte consistindo numa região, ou pode ser um espaço de até 16 Exabyte que consiste de regiões até 8 Gigabyte. A parte RX de um endereço virtual da aplicação dum espaço de endereço de 2 Gigabyte deve ser toda com zeros; caso contrário, é reconhecida uma excepção. A parte RX de um endereço virtual é dividida em si mesmo em três campos. Os bits 0 a 10 são denominados de índice de primeira região (RFX), os bits 11 a 21 são denominados de índice de segunda região (RSX), e os bits 22 a 32 são denominados de índice de terceira região (RTX). Num modelo de realização, os bits 0 a 32 do endereço virtual têm o formato tal como ilustrado de novo na **FIG. 19**.

Um endereço virtual no qual o RTX é a parte significativa mais à esquerda (um endereço de 42 bits) é capaz de endereçar 4 Terabytes (2048 regiões), um em que o RSX é a parte significativa mais à esquerda (um endereço de 53-bit) é capaz de endereçar 8 Petabytes (4.194.304 regiões), e um em que o RFX é a parte significativa mais à esquerda (um endereço de 64 bits) é capaz de endereçar 16 Exabytes (8.589.934.592 regiões).

Um endereço virtual no qual o RX é zero pode ser convertido num endereço real por meio de duas tabelas de conversão: uma tabela de segmento e uma tabela de página. Com a facilidade de EDAT activada, a conversão pode ser completada apenas com a tabela de segmento. O RFX pode ser

não-zero, em cujo caso são necessárias, uma tabela de primeira região, uma tabela de segunda região, e uma tabela de terceira região. Se o RFX é zero, mas o RSX possa ser não-zero, são exigidas uma tabela de segunda região e uma tabela de terceira região. Se o RFX e o RSX são zero, mas o RTX possa ser não-zero, é exigida uma tabela terceira região.

É reconhecida uma exceção se o ASCE para um espaço de endereço não designar o mais alto nível de tabela (começando com a tabela de primeira região e continuando no sentido descendente, para a tabela de segmento) necessária para converter uma referência para o espaço de endereço.

Conversão Dinâmica de Endereço Virtual

A referência está agora a ser feita na **FIG. 4** que ilustra um modelo de realização em que o ASCE efectivo determinado na **FIG. 3** é usado para determinar a primeira tabela de conversão a utilizar na conversão do endereço virtual de entre a hierarquia de tabelas de conversão.

Num modelo de realização, o registador de controlo 1 (CR1) contém o PASCE. O registador de controlo 7 (CR7) contém o SASCE. O registador de controlo 13 (CR13) contém o HASCE, e uma Entrada de Segunda tabela de Espaço de Endereço (ASTE), que é determinada pelo processo de Conversão de Registador de Acesso (ART) (Access-Register-Translation) contém um Elemento Controlo de Espaço de Endereço especificado de Registador de Acesso. Um ASCE

efectivo **400** é seleccionado de entre uma destas localizações.

Uma primeira fracção do ASCE **400** efectivo contém uma origem de tabela **402** que contém um endereço de origem com a designação ou de uma tabela de primeira região, uma tabela de segunda região, uma tabela de terceira região, ou uma tabela de segmento. A origem de tabela (bits 0...51) é anexada com 12 zeros binários para formar um endereço de origem de 64 bits da tabela de conversão mais elevada na hierarquia de tabelas de conversão a serem usadas na conversão do endereço virtual. O ASCE **400** efectivo também contém um bit de controlo de espaço real (R) **404** e bits de DT **406**. Se o bit de controlo de espaço real (R) é zero, então os bits DT são descodificados pelo selector **408** para determinar de qual endereço de origem particular é a origem da tabela **402**. Se os bits de DT igualam três (11 binário), então a origem de tabela **402** designa uma tabela de primeira região **410**. Se os bits de DT igualam dois (10 binário), então a origem de tabela **402** designa uma tabela de segunda região **412**. Se os bits de DT igualam um (01 binário), então a origem de tabela **402** designa uma tabela de terceira região **414**. Caso contrário, se os bits de DT iguais a zero (00 binário), então a origem de tabela **402** designa uma tabela de segmento **416**.

Uma tabela de primeira região, uma tabela de segunda região, ou uma tabela de terceira região é por vezes simplesmente referido como uma tabela de região. Do

mesmo modo, uma designação de tabela de primeira região, uma designação de tabela de segunda região, ou uma designação de tabela de terceira região é por vezes referido como uma designação de tabela de região. As tabelas de região, de segmento, e de página reflectem a atribuição corrente de armazenamento real. A página é um termo utilizado para a atribuição de armazenamento virtual. O armazenamento real é alocado em blocos fixos. As páginas não precisam de ser adjacentes em armazenamento real apesar de atribuídas a um conjunto sequencial de endereços virtuais.

Quando o ASCE utilizado numa conversão é uma designação de tabela de primeira região, o processo de conversão consiste na utilização de uma pesquisa de multi-nível, por exemplo, uma tabela de primeira região, uma tabela de segunda região, uma tabela de terceira região, uma tabela de segmento e, opcionalmente, uma tabela de página. Estas tabelas encontram-se em armazenamento real ou absoluto. Quando o ASCE é uma designação de tabela de segunda região, uma designação de tabela de terceira região, ou designação de tabela de segmento, as pesquisas nos níveis de tabelas acima do nível designado são omitidas, e as próprias tabelas de nível mais elevado são omitidas.

A referência está agora a ser feita na **FIG. 5A** que ilustra um modelo de realização da conversão de endereço dinâmica de um endereço virtual usando uma

hierarquia de tabelas de conversão. O ASCE **400** efectivo da **FIG. 4** contém os bits de Tipo de Designação (DT) **406**. Se o bit o controlo de espaço real (R) **404** do ASCE é zero, então os bits DT são descodificados pelo selector **408** para determinar qual a tabela de endereços de origem **402** designa a origem. Se o bit de controlo de espaço real (R) é um então tem lugar a conversão dinâmica de endereços, como mostrado no nodo D **564** na **FIG. 5B**.

Se os bits de DT igualam três (11 binário) no selector **408**, então, a tabela designada em primeiro lugar na hierarquia de tabelas de conversão é uma tabela de primeira região. A origem da tabela **402** é de modo aritmético adicionada, em **502**, com um Índice de Primeira Região (RFX) **508** fracção do endereço virtual para referenciar a entrada da tabela de primeira região **506** na tabela de primeira região. A origem da tabela (quer com 12 zeros anexados à direita, ou multiplicado por 4096) é adicionada ao produto do índice multiplicado por 8 (ou o índice com três zeros acrescentados à direita). A entrada da tabela de primeira região contém uma origem de tabela de segunda região **504** para uma tabela seguinte mais baixa na hierarquia das tabelas de conversão utilizadas na conversão. A tabela seguinte mais baixa para a tabela de primeira região é a tabela de segunda região. Se o bit inválido (I) de entrada da tabela de primeira região é igual a um, então, a entrada de tabela de primeira região é inválida e não pode ser usada na conversão. É indicada uma condição de excepção.

Se os bits DT igualam dois (10 binário) no selector **408**, então, a tabela designada em primeiro lugar da hierarquia de tabelas de conversão é uma tabela de segunda região. A origem da tabela **402** é de modo aritmético adicionada, em **510**, com um Índice de Segunda Região (RSX) **516** fracção do endereço virtual para referenciar a entrada da tabela de segunda região **514** na tabela de segunda região. A origem de tabela (quer com 12 zeros anexados à direita, ou multiplicado por 4096) é adicionada ao produto do índice multiplicado por 8 (ou o índice com três zeros acrescentados à direita). A entrada da tabela de segunda região contém uma origem de tabela de terceira região **512** para uma tabela seguinte mais baixa na hierarquia das tabelas de conversão utilizadas na conversão. A tabela seguinte mais baixa para a tabela de segunda região é a tabela de terceira região. Se o bit inválido (I) de entrada da tabela de segunda região é igual a um, então, a entrada de tabela de segunda região é inválida e é indicada uma condição de excepção.

Se os bits DT igualam um (01 binário) no selector **408**, então, a tabela designada em primeiro lugar da hierarquia de tabelas de conversão é uma tabela de terceira região. A origem da tabela **402** é de modo aritmético adicionada, em **518**, com um Índice de Terceira Região (RTX) **524** fracção do endereço virtual para referenciar a entrada da tabela de terceira região **522** na tabela de terceira região. A origem de tabela (quer com 12 zeros anexados à direita, ou multiplicado por 4096) é adicionada ao produto

do índice multiplicado por 8 (ou o índice com três zeros acrescentados à direita). A entrada da tabela de terceira região contém uma origem de tabela de segmento **520** para uma tabela seguinte mais baixa na hierarquia das tabelas de conversão utilizadas na conversão. A tabela seguinte mais baixa para a tabela de terceira região é a tabela de segmento. Se o bit inválido (I) de entrada da tabela de terceira região é igual a um, então, a entrada de tabela de terceira região é inválida e é indicada uma condição de exceção.

Se os bits DT igualam zero (00 binário) no selector **408**, então, a tabela designada em primeiro lugar da hierarquia de tabelas de conversão é uma tabela de segmento. A origem da tabela **402** é de modo aritmético adicionada, em **526**, com um Índice de Segmento (SX) **532** fracção do endereço virtual para referenciar a entrada da tabela de segmento **530** na tabela de segmento. A origem de tabela (quer com 12 zeros anexados à direita, ou multiplicado por 4096) é adicionada ao produto do índice multiplicado por 8 (ou o índice com três zeros acrescentados à direita). A entrada da tabela de segmento contém ou um endereço de origem para uma tabela de página ou um endereço absoluto de trama de segmento (SFAA), seja mostrado em **528**. Se o bit inválido (I) de entrada da tabela de segmento é igual a um, então, a entrada de tabela de segmento é inválida e é indicada uma condição de exceção.

Em **538**, o bit de controlo formato de STE (FC) da tabela de segmento é examinado. Se o controlo de formato de STE é um, então, a entrada da tabela de segmento **530** contém um endereço absoluto de trama de segmento (SFAA) **552** e a conversão dinâmica de endereço continua com referência ao **562** na **FIG. 5C**. Caso contrário, a entrada da tabela de segmento obtida para formar a tabela de segmento contém um endereço de origem de tabela de página e a conversão dinâmica de endereços continua com referência ao **560** na **FIG. 5B**.

Com referência agora a ser feita na **FIG. 5B**. Se o controlo de formato STE na entrada de tabela de segmento é zero, então a entrada da tabela segmento obtida a partir da tabela de segmento contém um endereço de origem para a tabela inferior seguinte na hierarquia de tabelas de conversão. A tabela inferior seguinte para a tabela de segmento é uma tabela de página. A origem da tabela de página **528**, obtida a partir da entrada da tabela de segmento **530** da **FIG. 5A**, é de modo aritmético adicionado, em **538**, com um Índice de Página (PX) **534** fracção do endereço virtual para referenciar a entrada de tabela de página **542** numa tabela de página. A origem da tabela (quer com 11 zeros adicionados à direita, ou multiplicado por 2048) é adicionada ao produto do índice multiplicado por 8 (ou o índice com três zeros acrescentados à direita). A entrada de tabela de página contém um endereço real de trama de página (PFRA) **546**. Quando os bits mais à esquerda do endereço real de trama de página são concatenados, em

548, com um índice de byte (BX) **536** é obtida uma fracção do endereço virtual, um endereço real de 64 bits **550**. O endereço real de 64 bits pode ser ainda submetido a mais uma operação de prefixação para formar um endereço absoluto. O endereço virtual convertido faz referência a um desejado bloco de 4Kilobyte (4096 bytes) de dados na armazenagem principal ou memória.

De preferência, a informação utilizada na conversão dinâmica de um endereço virtual para um endereço de memória é armazenada numa entrada com etiqueta de memória tampão com tabelas para conversão (TLB) juntamente com o endereço do bloco de memória associado com o endereço virtual. O subsequente acesso de armazenamento pode rapidamente converter um endereço virtual, comparando a informação ASCE e a informação de endereço virtual com etiquetas de memória tampão com tabelas para conversão (TLB). Se uma etiqueta é encontrada para ser a de endereço virtual, o endereço de memória tampão com tabelas para conversão (TLB) do bloco de memória pode ser usado em vez de executar o acesso sequencial lento de cada tabela de conversão envolvida. Num modelo de realização, o endereço real de trama de página (PFRA) juntamente com uma etiqueta que consiste em, por exemplo, o ASCE e o RX, SX, e PX fracções do endereço virtual são armazenados numa entrada de memória tampão com tabelas para conversão (TLB) **544**. A conversão subsequente deste endereço virtual é depois derivada a partir de informação armazenada na memória tampão com tabelas para conversão (TLB).

Com referência agora a ser feita na **FIG. 5C**. Se o controlo de formato STE na entrada da tabela de segmento **530** é um, então a entrada da tabela de segmento contém um endereço absoluto de trama de segmento (SFAA) (segment frame absolute address) **552**. Quando os bits mais à esquerda do endereço absoluto de trama de segmento são concatenados, em **554**, com uma fracção do índice de página **534** e uma fracção do índice de byte **536** do endereço virtual, é obtido um endereço absoluto de 64 bits **556**. O endereço virtual convertido referencia um desejado grande bloco de dados na armazenagem principal ou memória. O grande bloco de dados é de pelo menos de 1 megabyte (1.048.576 bytes) de tamanho.

Num modelo de realização, o endereço absoluto de trama de segmento (SFAA), juntamente com o RX e SX fracções do endereço virtual são armazenadas numa memória tampão com tabelas para conversão (TLB) **544**. A conversão subsequente deste endereço virtual é depois derivada a partir da informação armazenada na memória tampão com tabelas para conversão (TLB).

Formatos de Entrada de Tabelas de Conversão

Os modelos das várias entradas de tabela de conversão na hierarquia de tabelas de conversão utilizadas na conversão são os seguintes.

Entradas de Tabela Região

O termo "entrada da tabela de região", significa uma entrada de tabela de primeira região, entrada de tabela

de segunda região, ou entrada de tabela de terceira região. As entradas obtidas a partir da tabela de primeira região, da tabela de segunda região e da tabela de terceira região têm os seguintes formatos. O nível (primeiro, segundo, ou terceiro) da tabela contendo uma entrada é identificado na entrada pelos bits de tipo de tabela (TT).

Num modelo de realização, os formatos da entrada da tabela de primeira região, da entrada da tabela de segunda região, e da entrada da tabela de terceira região são tal como é ilustrado na **FIG. 20**.

Origem de Tabela de Segunda Região, Origem de Tabela de Terceira Região, e Origem de Tabela de Segmento: A entrada da tabela de primeira região contém uma origem da tabela de segunda região. A entrada da tabela de segunda região contém uma origem de tabela de terceira região. Uma entrada da tabela de terceira região contém uma origem de tabela de segmento. A seguinte descrição aplica-se a cada uma das três origens de tabela. Os bits 0 a 51 da entrada, com 12 zeros anexados à direita, formam um endereço de 64 bits que determina o início da tabela seguinte de nível imediatamente inferior.

O Bit de Protecção de **DAT** (P): Quando a DAT melhorada se aplica, o bit 54 é tratado como sendo OR'ed com o bit de protecção de DAT em cada subsequente entrada da tabela de região, a entrada da tabela de segmento, e, quando aplicável, a entrada de tabela de página usada na

conversão. Assim, quando o bit for um, a protecção de DAT aplica-se a toda a região ou a regiões especificadas pela entrada da tabela de região. Quando a facilidade de melhorada de DAT não está instalada, ou quando a facilidade está instalada, mas o controlo de activação de melhorada DAT é zero, o bit 54 de entrada da tabela de região é ignorado.

Compensação de Tabela de Segunda Região, Compensação de Tabela de Terceira Região, Compensação de Tabela de Segmento (TF): A entrada da tabela de primeira região contém uma compensação da tabela de segunda região. A entrada da tabela de segunda região contém uma compensação da tabela de terceira região. A entrada da tabela de terceira região contém uma compensação da tabela de segmento. A seguinte descrição aplica-se a cada uma das três compensações das tabelas. Os bits 56 e 57 da entrada especifica o comprimento de uma fracção da tabela de nível inferior seguinte que está ausente no início da tabela, isto é, os bits especificam a localização da primeira entrada actualmente existente na tabela de nível inferior seguinte. Os bits especificam o comprimento da fracção em falta em unidades de 4.096 bytes, tornando assim o comprimento da fracção ausente variável em múltiplos de 512 entradas. O comprimento da fracção em falta, em unidades de 4.096 bytes, é igual ao valor de TF. Os conteúdos do campo de compensação, em conjunto com o campo de comprimento, os bits 62 e 63, são utilizados para estabelecer se a fracção do endereço virtual (RSX, RTX, ou SX) a ser convertida por

meio da tabela de nível inferior seguinte designa uma entrada que realmente existe na tabela.

Bit de Região Inválida (I): o bit 58 numa entrada da tabela de primeira região ou entrada da tabela de segunda região controla se o conjunto de regiões associadas com a entrada está disponível. O bit 58 numa entrada da tabela de terceira região controla se a única região associada com a entrada está disponível. Quando o bit 58 é zero, procede endereço de conversão usando a entrada da tabela de região. Quando o bit for um, a entrada não pode ser usada para conversão.

Bits de Tipo de Tabela (TT): os bits 60 e 61 da entrada da tabela de primeira região, de entrada da tabela de segunda região, e entrada de tabela de terceira região identificam o nível da tabela que contém a entrada, como se segue: os bits 60 e 61 devem identificar o nível correcto de tabela, considerando o tipo de designação de tabela que é o ASCE sendo usado na conversão e no número de níveis da tabela que têm sido usados até ao momento, caso contrário, é reconhecida uma excepção de especificação de conversão. A tabela a seguir mostra os bits de tipo de tabela:

Bits 60 e 61	Nível de Região Tabela
11	Primeiro
10	Segundo
01	Terceiro

Bits de Tipo de Tabela para Entradas de Tabela de região

Comprimento de Tabela de Segunda Região, Comprimento de Tabela de Terceira Região, e Comprimento de Tabela de segmento (TL): Uma entrada da tabela de primeira região contém um comprimento da tabela de segunda região. A entrada da tabela de segunda região contém um comprimento da tabela de terceira região. Uma entrada da tabela de terceira região contém um comprimento da tabela de segmento. A seguinte descrição aplica-se a cada um dos três comprimentos de tabela. Os bits 62 e 63 da entrada especificam o comprimento da tabela de nível inferior seguinte em unidades de 4.096 bytes, tornando assim variável o comprimento da tabela em múltiplos de 512 entradas. O comprimento do nível inferior seguinte da tabela, em unidades de 4.096 bytes, é mais um do que o valor de TL. Os conteúdos do campo de comprimento, em conjunção com o campo de compensação, bits 56 e 57, são utilizados para determinar se a fracção de endereço virtual (RSX, RTX, ou SX) a ser convertida por meio da tabela de nível inferior seguinte designa uma entrada que realmente existe na tabela. Todas as outras posições de bits de entrada da tabela de região são reservadas para futuras extensões possíveis e deverão conter zeros; caso contrário, o programa não poderá operar compativelmente no futuro. Quando a melhorada DAT se aplica, a posições de bits reservadas da entrada da tabela de região deverão conter zeros, mesmo que a entrada da tabela seja inválida.

Entradas da Tabela de Segmento

Quando a melhorada DAT não se aplica, ou quando a melhorada DAT se aplica e o controlo de formato STE, o bit 53 da entrada da tabela de segmento é zero, a entrada obtida a partir da tabela de segmento, num modelo próprio de realização, tem o formato tal como o ilustrado na **FIG. 21**.

Quando a melhorada DAT se aplica e o controlo de formato STE é um, a entrada obtida a partir da tabela de segmento, num modelo próprio de realização, tem o formato tal como o ilustrado na **FIG. 22**.

Campos seleccionados na entrada da tabela de segmento são distribuídos da seguinte forma:

Origem da Tabela de página: Quando a melhorada DAT não se aplica, ou quando a melhorada DAT se aplica, mas o controlo de formato STE, bit 53 da entrada da tabela de segmento, é zero, os bits 0 a 52, com 11 zeros anexados à direita, formam um endereço de 64 bits que determina o início de uma tabela de página. É imprevisível se o endereço é real ou absoluto.

Endereço Absoluto de Trama de Segmento (SFAA): Quando se aplica a melhorada DAT e o controlo de formato STE é um, os bits 0 a 43 da entrada, com 20 zeros acrescentados à direita, formam o endereço absoluto de 64-bit do segmento.

ACCF Controlo de Validade (AV): Quando a melhorada DAT se aplica e o controlo de formato STE é um, bit 47 é o bit de controlo de acesso e bit de protecção de busca (ACCF) controlo de validade. Quando o controlo AV é zero, os bits 48 a 52 da entrada da tabela de segmento são ignorados. Quando o AV de controlo é um, os bits 48 a 52 são utilizados como descrito em baixo.

Bits de Controlo de Acesso (ACC): Quando a melhorada DAT se aplica, o controlo de formato STE é um, e o controlo AV é um, os bits 48 a 51 da entrada da tabela de segmento contém os bits de controlo de acesso que podem ser usados para qualquer chave de verificação de acesso controlado que se aplica ao endereço.

Bit de Protecção de Busca (F): Quando a melhorada DAT se aplica, o controlo de formato STE é um, e o controlo de AV é um, o bit 52 da entrada da tabela de segmento contém o bit de protecção de busca que pode ser utilizado para qualquer chave de verificação de acesso controlado que se aplica ao endereço.

Controlo de Formato STE (FC): Quando a melhorada DAT se aplica, bit 53 é o controlo de formato para a entrada da tabela de segmento, como se segue:

- Quando o bit FC é zero, os bits 0 a 52 formam a entrada da origem da tabela de página, e o bit 55 é reservado.

- Quando o bit FC é um, os bits 0 a 43 formam a entrada do endereço absoluto da trama de segmento, o bit 47 é o controlo de validade de ACCF, os bits 48 a 51 são os bits de controlo de acesso, o bit 52 é o bit de protecção de busca, e o bit 55 é a alteração da substituição de gravação. Quando a melhorada DAT não se aplica, o bit 53 é ignorado.

Bit de Protecção de DAT (P): Bit 54, quando é um, indica que a protecção DAT aplica-se a todo o segmento.

- Quando a melhorada DAT não se aplica, bit 54 é tratado como sendo OR'ed com o bit de protecção de DAT na entrada da tabela de página usada na conversão.
- Quando a melhorada DAT se aplica, o bit de protecção de DAT em toda e qualquer das entradas das tabelas de região utilizadas na conversão que são tratados como sendo OR'ed com o bit de protecção de DAT na entrada da tabela de segmento; quando o controlo de formato STE é zero, o bit de protecção de DAT no STE é ainda tratado como sendo OR'ed com o bit de protecção de DAT na entrada da tabela de página.

Substituição de Gravação de Mudança (CO): Quando a melhorada DAT se aplica e o controlo de formato STE é um, o bit 55 da entrada da tabela de segmento é a substituição de gravação da alteração para o segmento. Quando a melhorada DAT não se aplica, ou quando a melhorada DAT se aplica mas o controlo de formato STE é igual a zero, o bit 55 da entrada da tabela de segmento é ignorado.

Bit inválido de Segmento (I): Bit 58 controla quanto ao segmento associado com a entrada da tabela de segmento estar disponível.

- Quando o bit for zero, procede a conversão de endereço pelo uso da entrada da tabela de segmento.
- Quando o bit for um, a entrada da tabela de segmento não pode ser usada para conversão.

Bit Segmento Comum (C): Bit 59 controla o uso de cópias da memória tampão com tabelas para conversão (TLB) da entrada da tabela de segmento. Quando a melhorada DAT não se aplica ou quando se aplica a melhorada DAT mas o controlo de formato é zero, o bit 59 também controla a utilização de cópias da memória tampão com tabelas para conversão (TLB) da tabela de página designada pela entrada da tabela de segmento.

- Um zero identifica um segmento particular; neste caso, designa a entrada da tabela de segmento e qualquer tabela de página só pode ser utilizada em associação com a origem da tabela de segmento que designa a tabela de segmento na qual reside a entrada da tabela de segmento.
- Um um identifica um segmento comum; neste caso, designa a entrada da tabela de segmento e qualquer tabela de página pode continuar a ser utilizada para converter os correspondentes endereços ao índice de segmento, embora uma tabela de segmento diferente seja especificada.

No entanto, cópias da memória tampão com tabelas para conversão (TLB) de entrada da tabela de segmento e

qualquer tabela de página para um segmento comum não são utilizáveis se o controlo de espaço particular, o bit 55, é um no ASCE utilizado na conversão ou se aquele ASCE é uma designação espaço real. O bit de segmento comum deve ser zero se a entrada da tabela de segmento é obtida a partir do armazenamento durante uma conversão quando o controlo de espaço particular é um no ASCE a ser utilizado. Caso contrário, uma excepção de especificação de conversão é reconhecida.

Bits de Tipo de Tabela (TT): Bits 60 e 61 da entrada da tabela de segmento são 00 binários para identificar o nível da tabela que contém a entrada. Os significados de todos os valores possíveis de bits 60 e 61 numa entrada da tabela de região ou entrada da tabela de segmento são como se segue:

Bits 60 e 61	Nível de Tabela
11	Primeira-Região
10	Segunda-Região
01	Terceira-Região
00	Segmento

Bits de Tipo de Tabela 60,61

Os bits 60 e 61 devem identificar o correcto nível de tabela, considerando o tipo de designação da tabela que é o ASCE a ser usado na conversão e o número de níveis da tabela que até agora têm sido utilizados, caso contrário, é reconhecida uma excepção de especificação de conversão. Todos os outros bits posições da entrada da

tabela de segmento são reservadas para futuras possíveis extensões e devem conter zeros, caso contrário, o programa pode não funcionar no futuro de forma compatível. Quando a melhorada DAT se aplica, as posições de bits reservadas da entrada da tabela de segmento deve conter zeros, mesmo que seja inválida a entrada da tabela.

Entradas da Tabela de Página

Num modelo de realização, a entrada obtida a partir da tabela de página tem o formato tal como ilustrado na **Fig.23**.

Os campos seleccionados na entrada da tabela de página são distribuídos da seguinte forma:

Endereço Real da Trama de Página (PFRA): Bits 0 a 51 fornecem os bits mais à esquerda de um endereço de armazenagem real. Quando estes bits são concatenados com o campo de índice de 12-bit byte do endereço virtual à direita, é obtido um endereço real de 64 bits.

Bit Inválido de Página (I): Bit 53 controla sobre se a página associada com a entrada da tabela de páginas está disponível. Quando o bit é zero, procede a conversão de endereço usando a entrada da tabela de página. Quando o bit for um, a entrada da tabela de página não pode ser usada para a conversão.

Bit Protecção de DAT (P): Bit 54 controla sobre se os acessos de armazenagem podem ser efectuados na

página. Este mecanismo de protecção está em acréscimo com a protecção controlada da chave e os mecanismos de protecção de endereço baixo. O bit não tem algum efeito sobre a busca acessos. Se o bit for zero, os armazenamentos são permitidos na página, sujeito às seguintes restrições adicionais:

- O bit de protecção de DAT sendo zero na entrada da tabela de segmento utilizado na conversão.
- Quando a melhorada DAT se aplica, o bit de protecção de DAT sendo zero em toda as entradas da tabela de região utilizadas na conversão.
- Outros mecanismos de protecção.

Se o bit é um, os armazenamentos não são permitidos. Quando não existem condições de excepção de mais elevada de prioridade, uma tentativa de armazenamento conduz a que seja reconhecida uma excepção de protecção quando o bit de protecção de DAT for um,. O bit de protecção de DAT na entrada da tabela de segmento é tratado como sendo OR'ed com o bit 54 para a determinação se a protecção de DAT se aplica à página. Quando a melhorada DAT se aplica, o bit de protecção de DAT em qualquer das entradas da tabela de região usadas no seio da conversão também é tratado como sendo OR'ed com o bit 54 para determinar se a protecção de DAT se aplica.

Substituição de Gravação de Mudança (CO): Quando a melhorada DAT não se aplica, o bit 55 da entrada da tabela de página deve conter zero; caso contrário, é

reconhecida uma excepção de especificação de conversão como parte da execução de uma instrução usando aquela entrada para a conversão de endereços. Quando a melhorada DAT se aplica e o controlo de formato STE é zero, o bit 55 de entrada da tabela de página é a substituição de gravação de mudança para a página.

A posição de bit 52 da entrada deve conter zero; caso contrário, uma excepção de especificação de conversão é reconhecida como parte da execução de uma instrução usando aquela entrada para a conversão de endereços. As posições de bits 56 a 63 não são atribuídas e são ignorados.

Outro Modelo de Realização da Conversão Dinâmica

Esta secção descreve o processo de conversão, tal como é implicitamente executado antes que um endereço virtual seja usado para aceder ao armazenamento principal.

A conversão de um endereço virtual é controlada pelo bit de modo de DAT e os bits de controlo de espaço de endereço na palavra de status do programa e pelos ASCEs nos registadores de controlo 1, 7, e 13 e como especificado pelos registadores de acesso. Quando o ASCE utilizado numa conversão é uma designação de tabela de primeira região, a conversão é efectuada por meio de uma tabela de primeira região, uma tabela de segunda região, uma tabela de terceira região, uma tabela de segmento, e uma tabela de página, todas as quais encontram-se em armazenamento real

ou absoluto. Quando o ASCE é um tipo de nível mais baixo de designação da tabela (designação de tabela de segunda região, designação de tabela de terceira região, ou designação de tabela de segmento) a conversão é efectuada apenas por meio dos níveis de tabela começando com o nível designado, e os bits do endereço virtual que não são zero, requererão o uso de um nível superior, ou os níveis de tabela devem ser todos zeros; caso contrário, é reconhecida uma excepção de tipo ASCE. Quando o ASCE é uma designação do espaço real, o endereço virtual é tratado como um endereço real, e as entradas da tabela em armazenamento real ou absoluto não são utilizadas.

O ASCE utilizado para uma particular conversão de endereços é denominado de ASCE efectivo.

Consequentemente, quando um endereço virtual primário é convertido, os conteúdos do registador de controlo 1 são utilizados como o ASCE efectivo. Do mesmo modo, para um endereço virtual secundário, são utilizados os conteúdos do registador de controlo 7; para um endereço virtual especificado AR, o ASCE especificado é utilizado pelo registador de acesso, e para um endereço virtual de página principal são utilizados os conteúdos de registador de controlo 13.

Quando o controlo do espaço real no ASCE efectivo é zero, o tipo de designação no ASCE especifica o tipo de designação da tabela: tabela de primeira região, tabela de

segunda região, tabela de terceira região, ou segmento de tabela. A fracção correspondente do endereço virtual (índice de primeira região região, índice de segunda região, índice de terceira região, ou índice de segmento) é verificado na designação em confronto com o campo de comprimento de tabela, e ele é adicionado à origem na designação para seleccionar uma entrada na tabela designada. Se a entrada seleccionada está fora da sua tabela, como determinado na designação pelo campo de comprimento da tabela, ou se o bit I é um na entrada seleccionada, é reconhecido uma excepção de conversão de primeira região, conversão de segunda região, conversão de terceira região, ou conversão de segmento, dependendo do nível da tabela especificado pela designação. Se os bits de tipo de tabela na entrada seleccionada não indicam o nível esperado de tabela, é reconhecido uma excepção de especificação de conversão.

A entrada da tabela seleccionada por meio do ASCE efectivo designa o nível inferior seguinte de tabela a ser usada. Se a tabela corrente é uma tabela de primeira região, tabela de segunda região, ou tabela de terceira região, a fracção seguinte do endereço virtual (índice de segunda região, índice de terceira região, ou índice de segmento, respectivamente) é verificado em confronto com o deslocamento de tabela e campos de comprimento de tabela na entrada corrente da tabela, e ele é adicionado à origem da entrada para seleccionar uma entrada do nível inferior seguinte da tabela. Se a entrada seleccionada da tabela

seguinte está fora da sua tabela, como determinado pelo deslocamento de tabela e campos de comprimento de tabela na entrada da tabela corrente, ou se o bit I é um na entrada seleccionada, uma conversão de segunda região, conversão de terceira região, ou uma excepção de conversão de segmento é reconhecida, dependendo do nível da tabela seguinte. Se os bits de tipo de tabela na entrada seleccionada não indicam o nível esperado de tabela, é reconhecida uma excepção de especificação de conversão.

O processamento de fracções do endereço virtual, por meio de níveis de tabela sucessivos continua até que uma entrada da tabela de segmento tiver sido seleccionada. A entrada da tabela de segmento contém um bit de protecção de página que se aplica a todas as páginas do segmento especificado.

A fracção de índice de página do endereço virtual é adicionado à origem da tabela de página na entrada da tabela de segmento para seleccionar uma entrada na tabela de página. Se o bit I é um na entrada da tabela de página, uma excepção de conversão página é reconhecida. A entrada da tabela de página contém os bits mais à esquerda do endereço real que representam a conversão do endereço virtual, e contém um bit de protecção de página que se aplica apenas à página especificada pela entrada da tabela de página.

O campo de índice de byte do endereço virtual é usado sem modificações assim como as posições de bit mais à direita do endereço real.

Com vista a eliminar o atraso associado com as referências às tabelas de conversão no armazenamento real ou absoluto, a informação obtida normalmente a partir das tabelas também é colocada numa memória tampão especial, a memória tampão com tabelas para conversão (TLB), e subseqüentes conversões que envolvam as mesmas entradas de tabela podem ser efectuadas com base na informação registada na memória tampão com tabelas para conversão (TLB). A memória tampão com tabelas para conversão (TLB) pode também gravar conversões virtuais iguais às reais relacionadas com uma designação espaço real.

Sempre que o acesso ao armazenamento real ou absoluto é feito durante o processo de conversão de endereços para o propósito de buscar uma entrada de uma tabela de região, tabela de segmento, ou tabela de página, a protecção controlada chave não se aplica.

Pesquisa numa Tabela Designada por um ASCE

O controlo de DT, os bits 60 a 61 do ASCE efectivo, tanto especificam ambos, o tipo de designação da tabela do ASCE e a fracção do endereço virtual que é para ser convertido por meio da tabela designada, como se segue:

Bits 60 e 61	Tipo de Designação	Endereço-Virtual Fracção Transferida pela Tabela
11	Tabela-Primeira-Região	Índice Primeira-Região (bits 0-10)
10	Tabela-Segunda-Região	Índice Segunda-Região Bits (11-21)
01	Tabela-Terceira-Região	Índice Terceira-Região (bits 22-32)
00	Tabela-Segmento	Índice Segmento (bits 33-48)

Conversão por meio de Tabela Designada

Quando os bits 60 e 61 têm o valor 11 binário, o índice de primeira região fracção do endereço virtual, em conjunção com a origem de tabela de primeira região contida no ASCE, é usado para seleccionar uma entrada da tabela de primeira região. O endereço de 64-bit da entrada da tabela de primeira região no seio do armazenamento real ou absoluto é obtido acrescentando 12 zeros à direita dos bits 0 a 51 da designação da tabela de primeira região e adicionando o índice de primeira região com três zeros mais à direita e 50 zeros mais à esquerda anexados. Como parte do processo de pesquisa da tabela de primeira região, os bits 0 e 1 do endereço virtual (que são os bits 0 e 1 do índice de primeira região) são comparados em confronto com o comprimento de tabela, os bits 62 e 63 da designação da tabela de primeira região, para estabelecer se a entrada endereçada está dentro da tabela primeira região. Se o valor no campo de comprimento de tabela é menor do que o valor nas posições de bits correspondentes do endereço virtual, é reconhecida uma excepção de conversão de primeira região. A comparação com o comprimento da tabela

pode ser omissa se o equivalente da entrada da tabela de primeira região na memória tampão com tabelas para conversão é usado na conversão. A entrada obtida a partir da tabela de primeira região designa o início e especifica o deslocamento e comprimento da correspondente tabela de segunda região.

Quando os bits 60 e 61 da ASCE tem o valor 10 binário, o índice de segunda região fracção do endereço virtual, em conjunção com a origem da tabela de segunda região contida no ASCE, é usado para seleccionar uma entrada a partir da tabela de segunda região. Os bits 11 e 12 do endereço virtual (que são os bits 0 e 1 do índice de segunda região) são comparados em confronto com a tabela de comprimento no ASCE. Se o valor no campo de comprimento da tabela é menor do que o valor nas posições de bits correspondentes do endereço virtual, é reconhecida uma excepção de conversão de segunda região. A comparação com o comprimento da tabela pode ser omissa se o equivalente da entrada da tabela de segunda região na memória tampão com tabelas para conversão é usado na conversão. O processo de pesquisa de tabela de segunda região é de outro modo o mesmo que o processo de pesquisa da tabela de primeira região; a entrada obtida a partir da tabela de segunda região designa o início e especifica o deslocamento e o comprimento da tabela de terceira região correspondente.

Quando os bits 60 e 61 da ASCE têm o valor 01 binário, o índice de terceira região fracção do endereço

virtual, em conjunção com a origem da tabela de terceira região contida no ASCE, é usado para seleccionar uma entrada a partir da tabela de terceira região. Os bits 22 e 23 do endereço virtual (que são os bits 0 e 1 do índice de terceira região) são comparados com o comprimento da tabela no ASCE. Se o valor no campo de comprimento da tabela é menor do que o valor nas posições de bit correspondentes do endereço virtual, é reconhecida uma excepção de conversão de terceira região. O processo de pesquisa de tabela de terceira região é de outro modo o mesmo que o processo de pesquisa da tabela de primeira região, incluindo a verificação dos bits de tipo de tabela da entrada da tabela de terceira região. A entrada obtida a partir da tabela de terceira região designa o início e especifica o deslocamento e o comprimento da tabela de segmento correspondente.

Quando os bits 60 e 61 do ASCE têm o valor 00 binário, o índice de segmento fracção do endereço virtual, em conjunção com a origem da tabela de segmento contida no ASCE, é usado para seleccionar uma entrada a partir da tabela de segmento. Os bits 33 e 34 do endereço virtual (que são os bits 0 e 1 do índice do segmento) são comparados com o comprimento da tabela no ASCE. Se o valor no campo de comprimento da tabela é menor do que o valor nas posições de bits correspondentes do virtual endereço, é reconhecida uma excepção de conversão de segmento. A comparação com o comprimento de tabela pode ser omissa se o equivalente de uma entrada da tabela de segmento na memória

tampão com tabelas para conversão é usado na conversão. O processo de pesquisa da tabela de segmento é de outro modo o mesmo que o processo de pesquisa da tabela de primeira região, incluindo a verificação dos bits de tipo de tabela da entrada da tabela de segmento. O processamento é como se segue:

- Quando a melhorada DAT não se aplica, ou quando a melhorada DAT se aplica mas o controlo de formato STE é zero, a entrada obtida a partir da tabela de segmentos designa o início da correspondente tabela de página, e o processamento continua como descrito na "Pesquisa da Tabela de Página", em baixo.
- Quando a melhorada DAT se aplica e o controlo de formato STE é um, a entrada obtida a partir da tabela de segmento contém os bits mais à esquerda do endereço absoluto de trama de segmento. Se o bit de protecção de DAT é um, quer em qualquer entrada da tabela de região usada na conversão ou na entrada da tabela de segmento, e a referência de armazenamento para a qual a conversão está a ser efectuada é um armazenamento, é reconhecida uma excepção de protecção.

Pesquisa numa Tabela Designada por uma Entrada da Tabela de Região

Quando o ASCE efectivo é uma designação da tabela de região, uma entrada da tabela de região é seleccionada como descrito na secção anterior. Então os conteúdos da entrada seleccionada e a fracção índice seguinte do endereço virtual são usadas para seleccionar uma entrada da

tabela de nível mais baixo seguinte, que pode ser uma outra tabela de região ou uma tabela de segmento. Quando a entrada da tabela seleccionada por meio do ASCE é uma entrada da tabela de primeira região, a fracção do índice de segunda região do endereço virtual, em conjunção com a origem da tabela de segunda região contida na entrada da tabela de primeira região, é usada para seleccionar uma entrada da tabela de segunda região. O endereço de 64-bit da entrada da tabela de segunda região no seio do armazenamento real ou absoluto é obtido anexando 12 zeros para a direita dos bits 0 a 51 da entrada da tabela de primeira região e adicionando o índice de segunda região com três zeros mais à direita e 50 zeros mais à esquerda anexados.

Quando da formação do endereço de uma de entrada da tabela de segunda região, de terceira região, ou de segmento, é imprevisível saber se prefixando, se for caso disso, é aplicada à origem da tabela respectiva contida na entrada da tabela de nível mais elevado anterior à adição do valor do índice da tabela, ou a prefixação é aplicada ao endereço de entrada da tabela que é formada pela adição da origem de tabela e o valor índice da tabela.

Como parte do processo de pesquisa da tabela de segunda região, os bits 11 e 12 do endereço virtual (que são os bits 0 e 1 do índice de segunda região) são comparados com o deslocamento de tabela, os bits 56 e 57 da entrada da tabela de primeira região, e confronto com o

comprimento da tabela, os bits 62 e 63 da entrada da tabela de primeira região, para estabelecer se a entrada endereçada está dentro da tabela de segunda região. Se o valor no campo de deslocamento da tabela é maior do que o valor nas posições de bits correspondentes do endereço virtual, ou se o valor no campo de comprimento da tabela é menor do que o valor nas posições de bits correspondentes do endereço virtual, é reconhecida uma exceção de conversão de segunda região.

A tabela de segunda região designa o início e especifica o deslocamento e comprimento da tabela de terceira região correspondente.

Quando a entrada da tabela seleccionada por meio do ASCE é uma entrada de tabela de segunda região, ou se uma entrada de tabela de segunda região foi seleccionada por meio dos conteúdos de uma entrada de tabela de primeira região, a fracção do índice de terceira região do endereço virtual, em conjunção com a origem da tabela de terceira região contida na entrada da tabela de segunda região, é usada para seleccionar uma entrada da tabela de terceira região. Os bits 22 e 23 do endereço virtual (que são os bits 0 e 1 do índice de terceira região) são comparados com o deslocamento de tabela e com o comprimento de tabela da entrada de tabela de segunda região. É reconhecida uma exceção de conversão de terceira região se o deslocamento de tabela é maior do que os bits de 22 e 23 ou se o comprimento da tabela é menor do que os bits 22 e 23. O

processo de pesquisa de tabela de terceira região é de outro modo o mesmo que o processo de pesquisa de tabela de segunda região. A entrada obtida a partir da tabela de terceira região designa o início e especifica o deslocamento e o comprimento da tabela de segmento correspondente.

Quando a entrada da tabela seleccionada por meio do ASCE é uma entrada de tabela de terceira região, ou se uma entrada de tabela de terceira região foi seleccionada por meio dos conteúdos de uma entrada de tabela de segunda região, a fracção de índice de segmento do endereço virtual, em conjunção com a origem de tabela de segmento contida na entrada da tabela de terceira região, é utilizada para seleccionar uma entrada a partir da tabela de segmento. Os bits 33 e 34 do endereço virtual (que são os bits 0 e 1 do índice do segmento) são em comparação com o deslocamento da tabela e o comprimento da tabela na entrada da tabela de terceira região. É reconhecida uma excepção de conversão de segmento se o deslocamento de tabela é maior do que os bits 33 e 34 ou se o comprimento da tabela é menor do que os bits 33 e 34. É reconhecida uma excepção de especificação de conversão se (1) o controlo de espaço particular, o bit 55, no ASCE é um e (2) o bit de segmento comum, bit 59, na entrada obtida a partir da tabela de segmento é um. O processo de pesquisa de tabela de segmento é de outro modo o mesmo que o processo de pesquisa de tabela de segunda região. O processamento é como se segue:

- Quando a melhorada DAT não se aplica, ou quando a melhorada DAT se aplica mas o controlo de formato STE é zero, a entrada obtida a partir da tabela de segmento designa o início da tabela de página correspondente, e o processamento continua como descrito na "Pesquisa de Tabela de Página", em baixo.
- Quando a melhorada DAT se aplica e o controlo formato STE é um, a entrada obtida a partir da tabela de segmento contém os bits mais à esquerda do endereço absoluto de trama de segmento. Se o bit de protecção de DAT é um, quer em qualquer entrada da tabela de região usada na conversão ou na entrada da tabela de segmento, e a referência de armazenamento para a qual a conversão está a ser executada é um armazenamento, é reconhecida uma excepção de protecção.

Pesquisa de Tabela de Página

Quando a melhorada DAT não se aplica, ou quando a melhorada DAT se aplica mas o controlo formato STE é igual a zero, a fracção de índice de página do endereço virtual, em conjunção com a origem da tabela de página contida na entrada da tabela de segmento, é usada para seleccionar uma entrada da tabela de página.

O endereço de 64-bit da entrada da tabela de página na memória real ou absoluta é obtida anexando 11 zeros à direita da origem da tabela de página e adicionando o índice de página, com três zeros mais à direita e 53

zeros mais à esquerda anexados. A execução da posição de bit 0 não pode ocorrer.

A entrada obtida a partir da tabela de página indica a disponibilidade da página e contém os bits mais à esquerda do endereço real da trama de página. O bit inválido de página, bit 53, é examinado para estabelecer se a página correspondente está disponível. Se este bit for um, é reconhecida uma excepção de conversão da página. Se a posição de bit 52 contém um um, é reconhecida uma excepção de especificação de conversão. Quando a melhorada DAT não se aplica, ou a melhorada DAT se aplica e o controlo de formato STE é zero, uma excepção de especificação de conversão também é reconhecida se a posição de bit 55 contém um um. Se o bit de protecção de DAT é um quer na entrada da tabela de segmento utilizado na conversão, na entrada da tabela de página, ou, quando a melhorada DAT se aplica, em qualquer entrada da tabela de região utilizada durante a conversão, e a referência de armazenamento para o qual a conversão está a ser efectuada é um armazenamento, é reconhecida uma excepção de protecção.

Formação dos endereços reais e Absoluto

Quando o ASCE efectivo é uma designação de espaço real, os bits 0 a 63 do endereço virtual são utilizados directamente como o endereço de armazenamento real. O endereço real pode ser submetido a prefixação para formar um endereço absoluto. Quando o ASCE efectivo não é uma designação de espaço real e nenhuma excepções no processo

de conversão foram encontradas, aplicam-se as seguintes condições:

- Quando a melhorada DAT não se aplica, ou quando a melhorada DAT se aplica mas o controlo de formato STE é zero, o endereço real de trama de página é obtido a partir da entrada da tabela de página. O endereço real da trama de página e a fracção de índice de byte do endereço virtual são concatenados, com o endereço real de trama de página formando a parte mais à esquerda. O resultado é o endereço de armazenamento real o qual corresponde ao endereço virtual. O endereço real pode ser ainda submetido a prefixação para formar um endereço absoluto.
- Quando a melhorada DAT se aplica e o controlo de formato STE é um, o endereço absoluto de trama de segmento e as fracções de índice de página e de de índice byte do endereço virtual são concatenados, da esquerda para a direita, respectivamente, para formar o endereço absoluto, o qual corresponde ao endereço virtual.

Reconhecimento de Excepções durante a Conversão

As entradas de tabela marcadas como inválidas, ou contendo endereços inválidos ou formatos inválidos podem dar origem a excepções a serem reconhecidas durante o processo de conversão. As excepções são reconhecidas quando informação contida nas entradas de tabela é usada para a conversão e é considerada incorrecta.

Para além disso, uma excepção, por exemplo, uma excepção tipo-ASCE poderia ocorrer se o endereço virtual a

ser convertido está para além da gama que pode ser representada pela tabela de nível superior designada pelo ASCE. Num próprio exemplo, uma exceção tipo-ASCE poderia ocorrer como um resultado do seguinte pseudocódigo:

```
If ((DT < 3 AND RFX != 0) OR (DT < 2 AND (RFX || RSX) != 0) OR (DT < 1 AND RX != 0))  
then asce_type_exception();
```

Os bits de DT são o tipo de Designação (bits 60 a 61 do ASCE). O RFX é o Índice de Primeira Região (bits 0 a 10 do endereço virtual). O RSX é o Índice de Segunda Região (bits 11 a 21 do endereço virtual). O RX é o índice de região inteira (bits 0 a 32 do endereço virtual). Um especialista nesta tecnologia prontamente compreenderá as operações lógicas e os operandos comparativos usados no pseudo código.

Qualificador de Exceção de Conversão (TXQ)

Com respeito ao Qualificador de Exceção de Conversão, deve ser entendido que o termo "quando a melhorada DAT se aplica" refere-se ao nível (hospedeiro ou convidado) para o qual a exceção de conversão é apresentada. Por exemplo, um TXQ é apresentado numa exceção de conversão de hospedeiro sempre que a melhorada DAT aplica o nível de hospedeiro, quer se aplique ou não no nível de convidado.

Durante uma interrupção do programa DAT-relacionada, uma identificação exceção de conversão (TEID) é armazenada. Quando a facilidade DAT melhorada está

instalada e activada, o qualificador de excepção de uma conversão (TXQ) é armazenado em três bits da TEID. O TXQ apoia uma configuração de hospedeiro na determinação se a interrupção foi causada pelo hospedeiro ou pelo convidado. Se a excepção foi causada pelo convidado, o TXQ indica se a interrupção foi causada ou não pelo endereço contido numa entrada de tabela de folha de convidado e se assim for, quer a entrada de folha era uma Entrada da Tabela de Segmento ou uma Entrada de Tabela de Página. Isto permite que o hospedeiro possa reconhecer referências a áreas exibidas de convidado como tramas de 4 K-bytes versus 1 M-byte. O TXQ é concebido de tal modo que pode ser expandido se for necessário um tamanho maior de trama, por exemplo, 2 Gigabyte.

Durante uma interrupção do programa devido a um tipo ASCE, conversão-primeira-região, conversão-segunda-região, conversão-terceira-região, conversão-segmento, ou excepção conversão-página, bits 0 a 51 do endereço virtual causando que a excepção seja armazenada nas posições de bits 0 a 51 das localizações 168 a 175. Este endereço é por vezes referido como o endereço de excepção de conversão. Os bits 52 a 56 das localizações 168 a 175 são imprevisíveis.

Quando a facilidade de DAT reforçada está instalada e activada na configuração de hospedeiro e um tipo ASCE hospedeiro, conversão-região, conversão-segmento, ou excepção de conversão-página é reconhecido, os bits 57 a

59 contêm o qualificador de excepção-conversão (TXQ), como segue:

- 0 - Excepção foi causada pela configuração actualmente em execução (isto é, uma Excepção DAT convidado é apresentada durante a execução de convidado, ou uma excepção DAT hospedeiro durante a execução de hospedeiro).
- 1 - Excepção foi associada com um endereço virtual de hospedeiro derivado de um endereço de convidado que não os em baixo listados.
- 2 - Excepção foi associada com um endereço virtual de hospedeiro derivado do endereço real da trama-página de convidado.
- 3 - Excepção foi associada com um endereço virtual de hospedeiro derivado do endereço absoluto de trama-segmento de convidado.
- 4-7 reservados.

Os valores TXQ 1 a 3 são apresentados apenas em excepções de hospedeiro que ocorram durante a execução de convidado.

Dois Nível DAT

Como foi explicado acima, durante a execução de um convidado paginável, dois níveis de DAT são executados: endereços virtuais convidados são traduzidos através de DAT de convidado e, quando aplicável, com prefixação de endereços absolutos convidados, os quais são então tratados como endereços virtuais de hospedeiro no espaço de endereço primário hospedeiro, e convertido através de DAT de

hospedeiro e, quando aplicável, com prefixação para endereços absolutos hospedeiros. Durante o processo de DAT convidado, as referências são feitas para entradas da tabela de conversão de convidados, localizadas através de endereços absolutos convidados os quais devem também ser convertidos por meio de DAT de hospedeiro e, quando aplicável, com prefixação. O processo pode resultar em exceções DAT convidado com base em conteúdos e especificações de tabela de convidados, e cada referência a um endereço absoluto convidado, para uma entrada da tabela de convidado ou o destino final da conversão de convidado, pode dar origem a exceções DAT de hospedeiro. O TXQ apropriado deve ser gerado para cada uma dessas exceções, através das especificações acima. As **FIGS. 6 a 9** ilustram em detalhe o processo DAT de convidado, e as **FIGS. 9 a 11** ilustram o processo DAT de hospedeiro.

DAT de Convidado

A referência é agora feita sobre a **FIG. 6**, que ilustra um diagrama de fluxo de um modelo de realização da conversão de endereço dinâmica convidado no ponto de obtenção de um campo de controlo de formato a partir uma entrada da tabela de segmento convidado.

É obtido em **602**, um endereço virtual de convidado a ser convertido. São obtidos em **604**, o endereço de origem, o nível da tabela, e o comprimento da tabela de entre a mais elevada tabela de conversão utilizada na conversão do endereço virtual. O endereço de origem da tabela de

conversão utilizada pela primeira vez na conversão é retirado o campo de origem de tabela no ASCE; o nível de tabela e comprimento da tabela dependem respectivamente dos bits de DT e de TL no ASCE. O deslocamento de tabela para a tabela de maior conversão utilizada é sempre tratada como zero. Em **606**, uma fracção de índice do endereço virtual convidado a ser utilizada para referenciar a entrada da tabela apropriada na tabela de conversão é inicialmente verificada em confronto com o deslocamento de tabela e com comprimento da tabela, os quais determinam os valores de índice mínimo e máximo representados na tabela. Se, em **606**, o índice está fora desse intervalo, então a conversão do endereço virtual não pode prosseguir para diante. Num modelo de realização, em **608**, é apresentada ao nível de tabela a ser indexado uma apropriada excepção de conversão de convidado (por exemplo, uma excepção de conversão de primeira-região, de segunda-região, de terceira-região, ou de segmento). Porque isto é uma excepção de DAT convidado encontrada durante a execução do convidado, o TXQ é definido como zero para significar uma excepção no nível de configuração corrente. A conversão deste endereço virtual é, então, parada, em **610**. Se, em **606**, o índice está dentro do intervalo válido, então, em **612**, o índice multiplicado pelo comprimento de entrada da tabela é adicionado à origem de tabela para gerar o endereço absoluto convidado de uma entrada da tabela. Em **614**, este endereço absoluto convidado é tratado como um endereço virtual de hospedeiro, e DAT de hospedeiro e, quando aplicável, a prefixação é chamada para a obtenção do correspondente endereço absoluto de

hospedeiro. Um parâmetro adicional para a DAT hospedeiro, para ser usado se necessário na geração de um TXQ, indica que a fonte do endereço a ser convertido é um endereço convidado diferente de um endereço real de trama-página ou de endereço absoluto de trama-segmento. Em baixo explicado, as **FIGS 9 a 11**, detalham o processo de DAT hospedeiro. Se, em **616**, o processo DAT hospedeiro falhou, então, em **618**, o processo de conversão de convidado está encerrado, o processo DAT hospedeiro gerará uma exceção. O TXQ para esta exceção de hospedeiro é determinada como é descrito na **FIG. 9**. Se, em **616**, o processo DAT hospedeiro foi conseguido, então em **620**, a entrada da tabela-conversão de convidado é obtida utilizando a resultate endereço absoluto de hospedeiro. Em **622**, o bit inválido (I) da entrada da tabela de convidado é examinado. Se ele estiver definido, então a conversão ulterior do endereço virtual não pode prosseguir usando a entrada da tabela uma vez que foi marcada como sendo inválida; em **608**, uma exceção de conversão de convidado é apresentada, com um valor de TXQ indicando a configuração corrente, como acima, e a conversão ulterior deste endereço virtual que utiliza esta entrada da tabela de segmento pára em **610**. Caso contrário, em **624**, se a entrada obtida a partir da tabela de conversão de convidado não é uma entrada da tabela de segmento, então a tabela de segmento na hierarquia das tabelas de conversão não foi ainda referenciada. Neste caso, em **626**, é obtida a partir da entrada de tabela, a origem, o deslocamento e o comprimento de uma tabela seguinte, inferior na hierarquia de tabelas de conversão de convidado. O controlo regressa à

etapa **606**, no qual o campo de índice seguinte dentro do endereço virtual de convidado é verificado em confronto com o novo deslocamento e comprimento, e se dentro do intervalo, este índice é usado para referenciar a entrada da tabela correspondente na seguinte inferior tabela usada na conversão.

Por exemplo, se o endereço de origem de tabela da primeira tabela de conversão de convidado a ser usada na conversão é o de uma tabela de primeira região, então, a fracção de RFX do endereço virtual convidado é usada para fazer referência a uma entrada de tabela de primeira região com a tabela de primeira região. Se o endereço de origem de tabela é o de uma tabela de segunda região, então, a fracção RSX do endereço virtual é usada para fazer referência a uma entrada de tabela de segunda região dentro da tabela de segunda região. Se o endereço de origem de tabela é o de uma tabela de terceira região, então, a fracção RTX do endereço virtual é usada para fazer referência a uma entrada de tabela de terceira região dentro da tabela de terceira região. Se o endereço de origem de tabela é o de uma tabela de segmento, então a fracção SX do endereço virtual é usada para fazer referência a uma entrada de tabela de segmento dentro da tabela de segmento. As tabelas sucessivas são referenciadas até que a entrada da tabela de segmento tenha sido alcançada.

Uma vez que a entrada da tabela de segmento de convidado tenha sido alcançada, em **628**, o bit de controlo de formato (STE) da entrada da tabela de segmento é examinado para determinar se o controlo de formato está definido para este particular endereço virtual. Se o controlo de formato STE é zero, então ocorre a conversão dinâmica de endereço respeitante ao nodo **630**. Se o controlo de formato STE é um, então ocorre conversão dinâmica de endereço respeitante ao nodo **632**.

**Conversão de Endereço Dinâmica de Convidado
(Controlo de Formato STE é zero)**

A referência é agora feita sobre a **FIG. 7**, a qual ilustra uma continuação do diagrama de fluxo a partir do nodo **630** da **FIG. 6**, quando o controlo de formato STE de convidado é zero.

Em **710**, um endereço de origem para uma tabela de página de convidado é obtida a partir da entrada de tabela de segmento de convidado. Em **712**, uma fracção de índice de página (PX) do endereço virtual convidado é multiplicado pelo comprimento da entrada de tabela e adicionado à origem de tabela de página de convidado para gerar o endereço absoluto de convidado de uma entrada da tabela de página de convidado. Em **714**, este endereço absoluto de convidado é tratado como um endereço virtual de hospedeiro, e a DAT de hospedeiro e, quando aplicável, a prefixação é chamada para obtenção do correspondente endereço absoluto de hospedeiro. Um parâmetro adicional para a DAT de hospedeiro, para ser

usado na geração de um TXQ se necessário, indica que a fonte do endereço a ser convertido é um endereço de convidado que não seja um do endereço real de trama-página ou endereço absoluto de trama-segmento. Se, em **716**, falhou o processo de DAT de hospedeiro, então em **718**, está terminado o processo de conversão de convidado; o processo de DAT de hospedeiro terá gerado uma exceção. Se, em **716**, o processo de DAT hospedeiro é conseguido, então, em **720**, é obtida a entrada da tabela de página de convidado utilizando o endereço absoluto de hospedeiro, resultante. Um bit de inválido (I) é obtido a partir da entrada de tabela de página de convidado. Se, em **722**, o bit inválido (I) é um então, a conversão do endereço virtual não pode prosseguir utilizando esta entrada da tabela de página porque a entrada foi assinalada como sendo inválida; em **724**, uma exceção de conversão da página de convidado é apresentada, com um valor de TXQ que indica o surgimento duma exceção a partir da conversão no nível de configuração corrente. A conversão para diante do endereço virtual utilizando esta entrada da tabela de página pára, em **726**. Se, em **722**, o bit inválido (I) é zero, então, em **728**, o endereço real de trama de página de convidado (PFRA) a partir da entrada da tabela de página de convidado é combinada com uma fracção de índice do byte (BX) do endereço virtual de convidado para gerar o endereço real de convidado correspondente à entrada do endereço virtual de convidado. Na **730**, este endereço real de convidado é ainda sujeito a uma operação de prefixação de convidado para formar um endereço absoluto de convidado. Em **732**, este

endereço absoluto convidado é tratado como um endereço virtual de hospedeiro, e a DAT de hospedeiro e, quando aplicável, a prefixação é chamada para obter o correspondente endereço absoluto de hospedeiro. Um parâmetro adicional para a DAT de hospedeiro, para ser usado na geração de um TXQ se necessário, indica que a fonte do endereço a ser convertido é um endereço real de trama-página de convidado. Se, em **734**, o processo de DAT de hospedeiro falhou, então, em **718**, o processo de conversão convidado está terminado; o processo de DAT de hospedeiro gerará uma exceção. Se, em **734**, o processo de DAT de hospedeiro é conseguido, então, em **736**, o endereço absoluto de hospedeiro resultante é usado para aceder a um bloco de dados endereçados pelo endereço virtual de convidado convertido.

Conversão de Endereço Dinâmica de Convidado (Controlo de Formato STE é um)

A referência é agora feita sobre a **FIG. 8**, a qual ilustra uma continuação do diagrama de fluxo a partir do nodo **632** da **FIG. 6**, quando o controlo de formato STE de convidado é um.

Em **810**, um endereço absoluto de trama de segmento de convidado (SFAA) é obtido a partir de uma fracção da entrada da tabela de segmento de convidado. Em **812**, o endereço absoluto de trama de segmento de convidado (SFAA) é combinado com as fracções de PX e BX do endereço virtual de convidado para gerar o endereço absoluto de convidado de

um desejado bloco de dados no armazenamento principal ou na memória. Em **814**, este endereço absoluto de convidado é tratado como um endereço virtual de hospedeiro, e a DAT de hospedeiro e, quando aplicável, a prefixação é chamada para obter o correspondente endereço absoluto de hospedeiro. Um parâmetro adicional para a DAT de hospedeiro, para ser usado na geração de um TXQ se necessário, indica que a fonte do endereço a ser convertido é um endereço absoluto de trama-segmento de convidado. Se, em **816**, o processo de DAT de hospedeiro falhou, então em **818**, o processo de conversão de convidado está terminado, o processo de DAT hospedeiro gerará uma exceção. Se, em **816**, o processo de DAT hospedeiro foi conseguido, então, em **720**, o endereço absoluto do hospedeiro resultante é usado para aceder ao desejado bloco de dados endereçados pelo endereço virtual de convidado convertido.

DAT de Hospedeiro

A referência é agora feita sobre a **FIG. 8**, a qual ilustra um diagrama de fluxo de um modelo de realização da conversão dinâmica de endereços do hospedeiro no nível do hospedeiro, o qual pode ser chamado a partir do processo EDAT de convidado, para obter um campo de controlo de formato de uma entrada da tabela de segmento de hospedeiro.

Em **902**, é obtido um endereço virtual de hospedeiro a ser convertido. Em **904**, são obtidos o endereço de origem, o nível de tabela, e o comprimento da tabela mais elevada da tabela de conversão de hospedeiro,

utilizada na conversão do endereço virtual de hospedeiro. O endereço de origem da primeira tabela de conversão de hospedeiro usada na conversão é tomado a partir do campo de origem de tabela no ASCE; o nível de tabela e comprimento de tabela dependem respectivamente dos bits de DT e de TL no ASCE. O deslocamento de tabela para a mais elevada tabela de conversão usada é sempre tratado como zero. Em **906**, uma fracção de índice do endereço virtual de hospedeiro a ser utilizado para referenciar a entrada da tabela apropriada na tabela de conversão é primeiro verificado no confronto com o deslocamento de tabela e o comprimento de tabela, que determinam o mínimo e o máximo valores de índice representados na tabela. Se, em **906**, o índice está fora deste intervalo, então a conversão para diante do endereço virtual não pode prosseguir, uma excepção de conversão de hospedeiro deve ser apresentada, apropriado para o nível de tabela que está a ser indexado (por exemplo, excepção de conversão de uma primeira-região, de segunda-região, de terceira-região, ou de segmento). Num modelo de realização, em **908**, é feito um teste para saber se um programa de convidado está actualmente em execução. Se não, isto é, se o hospedeiro está em execução, então, em **910**, uma excepção de conversão de hospedeiro é apresentado com o conjunto de TXQ, por exemplo, para zero, para significar uma excepção no nível de configuração corrente. Se em vez de um programa convidado está em execução, então, em **912**, uma excepção de conversão de hospedeiro é apresentado com o conjunto de TXQ para o valor de origem de endereço passado a partir da operação de conversão de

convidado. Em qualquer caso, o processo de conversão é então parado, em **914**.

Se, em **906**, o índice está dentro do intervalo válido, então, em **916**, o índice multiplicado pelo comprimento de entrada da tabela é adicionado à origem da tabela de hospedeiro para gerar o endereço absoluto de hospedeiro de uma entrada da tabela de hospedeiro. Em **918**, a entrada da tabela de conversão de hospedeiro é obtida usando esse endereço absoluto de hospedeiro. Em **920**, o bit inválido (I) na entrada da tabela de hospedeiro é examinado. Se ele estiver definido, então a conversão para diante do endereço virtual não pode prosseguir usando a entrada da tabela, porque foi assinalado como sendo inválido; controlo é transferido para **908** para apresentar uma exceção de conversão de hospedeiro com o adequado valor de TXQ, como acima, e a conversão para diante deste endereço virtual usando esta entrada da tabela de segmento é parada. Caso contrário, a **922**, se a entrada obtida a partir da conversão tabela não é uma entrada da tabela de segmento, então, a tabela de segmento na hierarquia das tabelas de conversão ainda não foi referenciada. Neste caso, em **924**, a origem, o deslocamento, e o comprimento de uma seguinte menor tabela na hierarquia de tabelas de conversão de hospedeiro são obtidos a partir da entrada da tabela. O controlo retorna para a etapa **906**, em que o campo de índice seguinte dentro do endereço virtual de hospedeiro é verificado em confronto com o novo deslocamento e comprimento, e se estiver dentro do intervalo, este índice

é usado para referenciar a entrada da tabela de hospedeiro correspondente na tabela imediatamente inferior usada na conversão.

Por exemplo, se o endereço de origem de tabela da primeira tabela de conversão de hospedeiro a ser utilizado em conversão é uma tabela de primeira região, então, a fracção RFX do endereço virtual de hospedeiro é usada para fazer referência a uma entrada de tabela de primeira região com a tabela de primeira região. Se o endereço de origem de tabela é uma tabela de segunda região, então, a fracção RSX do endereço virtual é usada para fazer referência a uma entrada da tabela de segunda região dentro da tabela segunda região. Se o endereço de origem de tabela é uma tabela de terceira região, então, a fracção RTX do endereço virtual é usada para fazer referência a uma entrada de tabela de terceira região dentro da tabela de terceira região. Se o endereço de origem de tabela é uma tabela de segmento, então a fracção de SX do endereço virtual é usada para referenciar uma entrada de tabela de segmento dentro da tabela de segmento. Tabelas sucessivas são referenciadas até que a entrada da tabela de segmento seja obtida.

Uma vez que a entrada da tabela de segmento de hospedeiro foi alcançada, em **926**, a entrada da tabela de segmento o bit de controlo de formato (STE) é examinado para determinar se o controlo de formato está definido para este endereço virtual particular. Se o controlo de formato STE é zero, então ocorre a conversão dinâmica de endereço

com respeito ao nodo **928**. Se o controlo de formato STE é um então ocorre a conversão dinâmica de endereço com respeito ao nodo **930**.

**Conversão dinâmica de endereços de Hospedeiro
(Controlo de Formato STE é zero)**

A referência é agora feita sobre a **FIG. 10** a qual ilustra uma continuação do diagrama de fluxo a partir do nodo **928** da **FIG. 9**, quando o controlo de formato de hospedeiro STE é zero.

Em **1002**, um endereço de origem para uma tabela de página de hospedeiro é obtido a partir da entrada da tabela de segmento de hospedeiro. Em **1004**, uma fracção do índice de página (PX) do endereço virtual de hospedeiro é multiplicado pelo comprimento da entrada da tabela e adicionado à origem da tabela de página para gerar o endereço absoluto de hospedeiro de uma entrada da tabela de página de hospedeiro. Em **1006**, a entrada da tabela de página de hospedeiro é obtida usando este endereço de absoluto de hospedeiro. Um bit inválido (I) é obtido a partir da entrada da tabela de página de hospedeiro. Se, em **1008**, o bit inválido (I) é um então, a conversão do endereço virtual não pode continuar com a utilização desta entrada da tabela de página porque a entrada foi assinalada como sendo inválida; o controlo é transferido para **940** na **FIG. 9**, para apresentar uma excepção de conversão de hospedeiro com o adequado valor TXQ, como acima, e a conversão para diante deste endereço virtual utilizando

esta entrada da tabela de página é parada. Se, em **1008**, o bit inválido (I) é zero então, em **1010**, o endereço real de trama de página (PFRA) de hospedeiro a partir da entrada da tabela de página é combinado com a fracção de índice de byte (BX) do endereço virtual de hospedeiro para gerar o endereço real de hospedeiro correspondente à entrada do endereço virtual de hospedeiro. Em **1012**, este endereço de real de hospedeiro é ainda sujeito a uma operação de prefixação para formar um endereço absoluto de hospedeiro. Em **1014**, este endereço absoluto de hospedeiro retorna como resultado da conversão de endereço de hospedeiro, para ser usado, por exemplo, no processo de DAT de convidado, que o chamou, ou se no caso de nenhum, então para aceder aos dados de destino durante a execução do hospedeiro.

Conversão dinâmica de endereços de Hospedeiro (Controlo de Formato STE é um)

A referência é agora feita sobre a **FIG. 11** a qual ilustra uma continuação do diagrama de fluxo a partir do nodo **930** da **FIG. 9**, quando o controlo de formato STE de hospedeiro é um.

Em **1102**, um endereço absoluto de trama de segmento (SFAA) de hospedeiro é obtido a partir de uma fracção da entrada da tabela de segmento de hospedeiro. Em **1104**, o endereço absoluto de trama de segmento de hospedeiro (SFAA) é combinado com as fracções de PX e de BX do endereço virtual do hospedeiro para gerar o endereço absoluto de hospedeiro de um desejado bloco de dados no

armazenamento principal ou na memória. Em **1106**, este endereço absoluto de hospedeiro retorna como resultado da conversão de endereço de hospedeiro, para ser usado, por exemplo, no processo de DAT de convidado que o chamou, ou se no caso de nenhum, então para aceder aos dados de destino durante a execução de hospedeiro.

O campo de qualificador de exceção de conversão (TXQ) é, assim, capaz de indicar que a interrupção foi causada por um endereço contido numa entrada de tabela de folha de convidado; e nesse caso, se a entrada de folha era uma entrada da tabela de segmento ou uma entrada da tabela de página, isto é, se a exceção estava associada com um endereço virtual de hospedeiro derivado a partir de um endereço real de trama de página de convidado ou estava associada a um endereço virtual de hospedeiro derivado a partir de um endereço absoluto de trama de segmento de convidado. O campo TXQ é assim capaz de indicar o tamanho adequado da trama de hospedeiro com o qual se apoia uma trama de convidado. É ainda capaz de indicar que um maior tamanho de trama é necessário, por exemplo 2 Gigabyte ou maior.

Implementação comercial

Embora a z/Arquitectura® IBM® é aqui mencionada, um ou mais aspectos da presente invenção são igualmente aplicáveis a outras arquitecturas de máquina and/or ambientes de computação que empregam entidades pagináveis ou construções semelhantes.

Implementações comerciais da facilidade EDAT, o TXQ, e outros formatos, instruções, e atributos aqui divulgados podem ser implementados em hardware ou por programadores, tais como programadores de sistema operativo, escrito em, por exemplo, a linguagem assembly. Tais instruções de programação podem ser armazenadas num meio de armazenamento pretende-se que sejam ser executadas nativamente num ambiente de computação, tais como o servidor IBM[®] System z, ou em alternativamente em máquinas que executam outras arquitecturas. As instruções podem ser emuladas em servidores existentes e em servidores futuros e em outras máquinas ou mainframes. Eles podem ser executados em máquinas onde geralmente a execução é num modo de emulação.

Em modo de emulação, a instrução específica que está a ser emulada é descodificada, e uma sub-rotina é executada para implementar a instrução individual, tal como numa sub-rotina ou controlador, ou alguma outra técnica é utilizada para proporcionar um controlador para o hardware específico, tal como está no íntimo daqueles que têm perícia na tecnologia após a compreensão da descrição do presente documento. Várias técnicas de software e hardware de emulação são descritos em numerosas patentes dos Estados Unidos, que se incluem: 5.551.013, 5.574.873, 5.790.825, 6.009.261, 6.308.255, e 6.463.582. Muitos outros ensinamentos para além destes ilustram uma variedade de maneiras para alcançar a emulação de um conjunto de instruções arquitectadas para uma dada máquina de destino.

Outras variações e Architecturas

Os vários modelos de realização aqui descritos são precisamente exemplos. Pode haver muitas variações a estes modelos de realização.

Uma ou mais das capacidades da presente invenção podem ser implementadas no software, hardware, firmware, ou alguma combinação destes. Aspectos da invenção são benéficos para muitos tipos de ambientes, incluindo outros ambientes que têm uma pluralidade de zonas, e ambientes não-particionados. Para além disso, pode não existir algum complexo processador central, mas ainda, vários processadores acoplados em conjunto. Vários aspectos deste documento são aplicáveis a ambientes de um único processador.

Se o ambiente é logicamente particionado, em seguida, mais ou menos partições lógicas podem ser incluídas no ambiente. Para além disso, podem existir múltiplos complexos de processamento central acoplados conjuntamente. Para além disso, outras variações são possíveis.

Embora o termo 'página' seja usado para se referir a um tamanho fixo ou uma área de tamanho pré-definido de armazenamento, o tamanho de uma página pode variar. Da mesma forma, o tamanho de um bloco pode variar. Pode haver diferentes tamanhos de blocos e/ou páginas. Uma página pode ser equivalente a um bloco. Outras estruturas

podem ser alternativamente utilizadas ou de outro modo implementadas através de software e/ou hardware. Para além disso, nos exemplos aqui descritos, podem existir muitas variações, incluindo, mas não limitado a diferentes tamanhos de palavras ou endereços; um número diferente de bits; bits numa ordem diferente; mais, menos ou diferentes bits; mais, menos ou diferentes campos; campos numa ordem diferente; diferentes tamanhos de campos, etc.. Novamente, estes só são fornecidos como um exemplo. Muitas variações são possíveis.

Uma variação possível no processo de conversão aqui descrito é para tratar endereços da entrada da tabela de de DAT como real ao invés de endereços absolutos. Uma outra variação possível é a realização de uma transformação entre endereço absoluto de convidado e endereço virtual de hospedeiro, tal como a adição de origem de armazenamento-principal de convidado ou testes com um limite de armazenamento-principal de convidado, de modo a restringir o armazenamento absoluto de convidado a uma fracção desejada do espaço que contém o endereço do hospedeiro. Ainda uma outra variação poderia permitir endereços absolutos de convidado a serem referenciados em múltiplos espaços do endereço de hospedeiro, seleccionados, por exemplo, através dos conteúdos de registadores de acesso.

Uma unidade de processamento inclui entidades pagináveis, tais como convidados, hospedeiros, outros processadores, outros emuladores, máquinas virtuais e/ou

outras construções semelhantes. Uma memória tampão inclui uma área de armazenamento e/ou memória, bem como tipos diferentes de estruturas de dados, incluindo, mas não se limitando para, matrizes ou entidades pagináveis. Uma tabela pode incluir também outras estruturas de dados. Uma instrução pode referenciar outros registadores. Além disso, uma página, um segmento, e/ou uma região pode ser de variados tamanhos diferentes do que aqueles aqui descritos.

Um ou mais aspectos da presente invenção pode ser incluído num artigo de fabrico (por exemplo, um ou mais produtos de programa de computador) tendo, por exemplo, computador utilizável ou suportes legíveis por máquina. Os media têm neles incorporado, por exemplo, computador legível de meios de programa de código ou lógica (por exemplo, instruções, código, comandos, etc) para proporcionar e facilitar as capacidades da presente invenção. O artigo de fabrico pode ser incluído como uma parte de um sistema de computador ou vendido separadamente. Para além disso, pode ser fornecido pelo menos, um dispositivo de armazenamento de programas legível por uma máquina que integra, pelo menos, um programa de instruções

executável pela máquina para executar as capacidades da presente invenção.

Lisboa, 26 de Abril de 2012

REIVINDICAÇÕES

1. Um método para a qualificação de uma exceção de conversão numa facilidade de conversão dinâmica de endereços com a capacidade de conversão dum endereço virtual num endereço convertido a partir de um bloco de dados na memória principal (208) num sistema de computador (200), o método compreendendo:

obtenção de um endereço virtual para ser convertido;

a conversão de modo dinâmico do endereço virtual num endereço real ou absoluto de um desejado bloco de dados da memória principal; e

em resposta a um evento de interrupção de exceção de conversão a ter ocorrido durante a conversão dinâmica de endereços do endereço virtual;

armazenamento de bits num qualificador de exceção de conversão, para indicar que a referida exceção de conversão foi própria de uma exceção de DAT de hospedeiro a ter ocorrido durante a execução de um programa convidado, e uma exceção de DAT de hospedeiro a ter ocorrido durante a execução de um programa convidado; e caracterizado por

armazenamento de bits no referido qualificador de exceção de conversão para indicar qualquer um que seja o tamanho de uma trama de convidado à qual a referida exceção de DAT pertence, e um tamanho de uma trama de

hospedeiro a ser alocada para o retorno à referida trama de convidado.

2. O método de acordo com a reivindicação 1, compreendendo ainda o armazenamento de bits no referido qualificador de exceção de conversão para indicar que a referida exceção de conversão foi uma exceção de DAT de convidado a ter ocorrido durante a execução do referido programa convidado.

3. O método de acordo com a reivindicação 1 ou 2, compreendendo ainda o armazenamento de bits no referido qualificador de exceção de conversão para indicar que a referida exceção de DAT de hospedeiro pertencia a um endereço derivado de uma entrada da tabela de folha de convidado.

4. O método de acordo com a reivindicação 1, 2 ou 3, compreendendo ainda o armazenamento de bits no referido qualificador de exceção de conversão para indicar que a referida exceção de DAT de hospedeiro pertencia a um endereço derivado de um endereço real de trama de página de convidado.

5. O método de acordo com a reivindicação 1, 2, 3 ou 4, compreendendo ainda o armazenamento de bits no referido qualificador de exceção de conversão para indicar que a referida exceção de DAT de hospedeiro pertencia a um

endereço derivado de um endereço absoluto de uma trama de segmento de convidado.

6. Um sistema que compreende meios adaptados para a realização de todos as etapass do método de acordo com qualquer reivindicação do método precedente.

7. Um programa de computador que compreende instruções para a realização de todas as etapas do método de acordo com qualquer reivindicação do método precedente, quando o referido programa de computador é executado num sistema de computador.

Lisboa, 26 de Abril de 2012

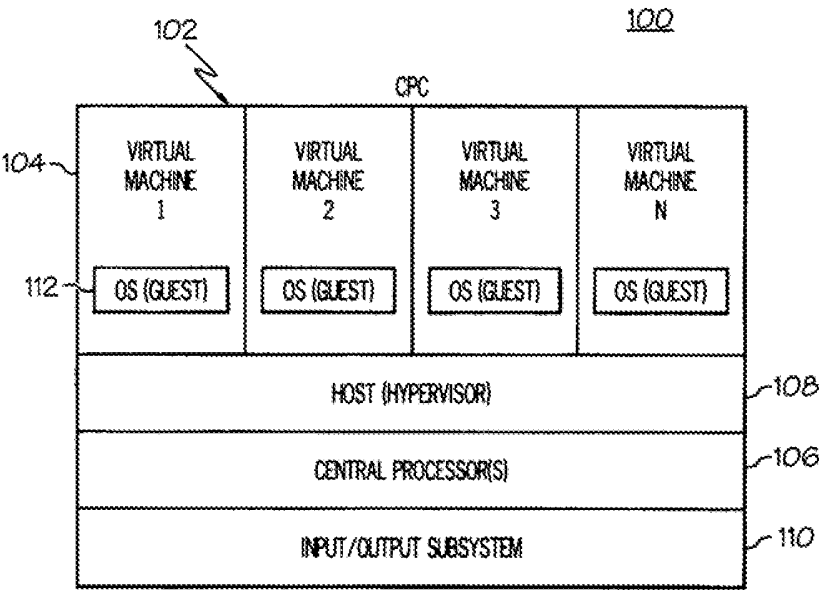


FIG. 1
(PRIOR ART)

Legenda da Figura 1:

	(PRIOR ART)	(TECNOLOGIA ANTECEDENTE)
100	Computing environment	Ambiente de computação
102	Central processor	Processador central
104	Virtual machines	Máquinas virtuais
106	Central processor(s)	Processador(es) central(is)
108	Host (hypervisor)	Hospedeiro (hipervisor)
110	input/output subsystem	subsistema de input/output
112	OS (GUEST)	SISTEMA OPERACIONAL (DE CONVIDADO)

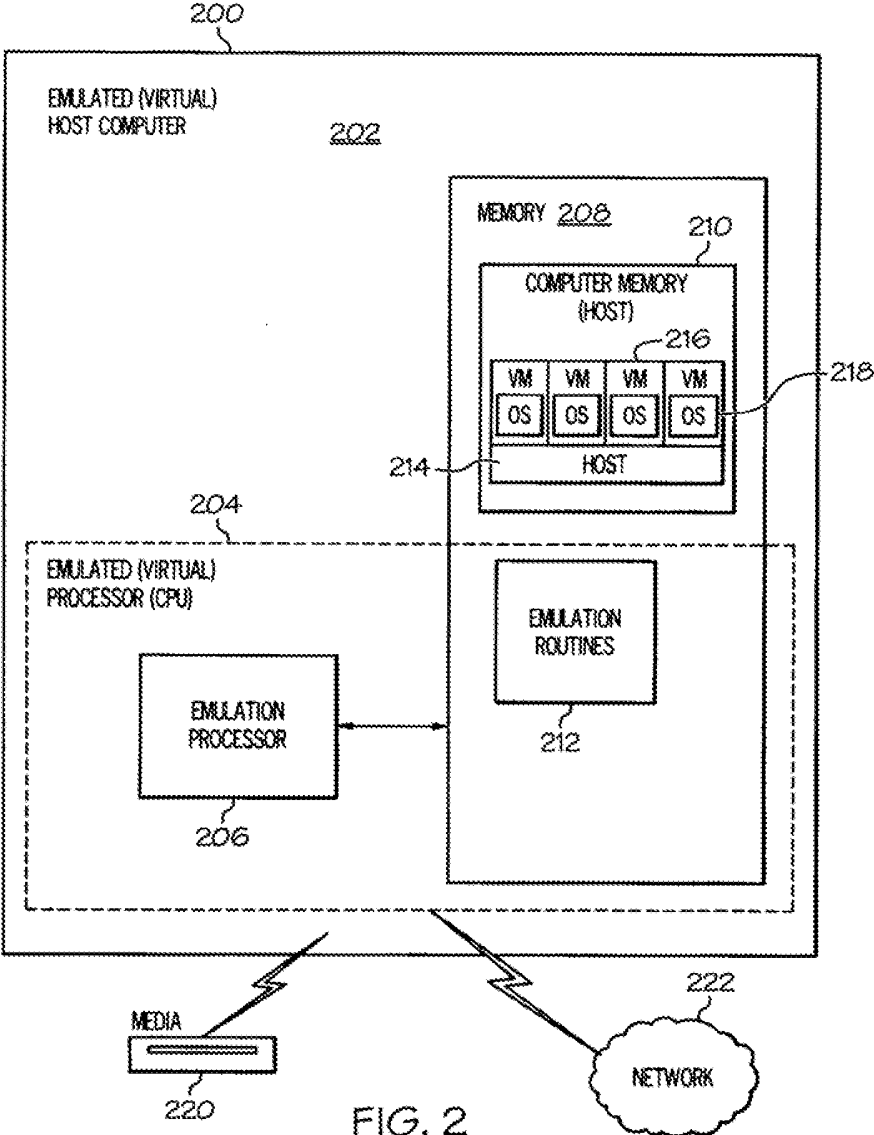


FIG. 2
(PRIOR ART)

Legenda da Figura 2:

	(PRIOR ART)	(TECNOLOGIA ANTECEDENTE)
200C	EMULATED HOST COMPUTER SYSTEM	SISTEMA DE COMPUTADOR HOSPEDEIRO EMULADO
202	EMULATED (VIRTUAL) HOST COMPUTER	COMPUTADOR HOSPEDEIRO EMULADO (VIRTUAL)
204	EMULATED (VIRTUAL) PROCESSOR (CPU)	EMULADO (VIRTUAL) PROCESSADOR (CPU)

206	EMULATION PROCESSOR	PROCESSADOR DE EMULAÇÃO
208	MEMORY	MEMÓRIA
210	COMPUTER MEMORY (HOST)	MEMÓRIA DE COMPUTADOR (HOSPEDEIRO)
212	EMULATION ROUTINES	ROTINAS DE EMULAÇÃO
214	HOST	HOSPEDEIRO
216	VM	Máquinas Virtuais
218	OS	Sistema Operacional
220	MEDIA	Meios de Armazenamento
222	NETWORK	REDE

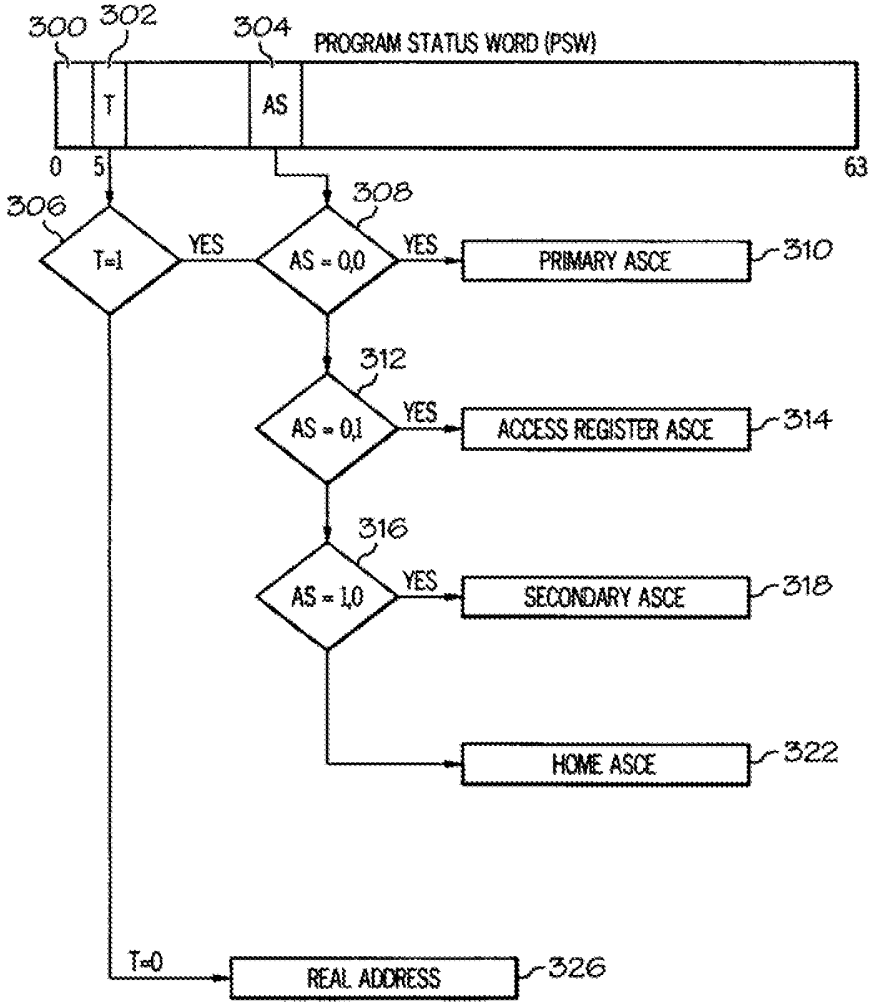


FIG. 3
(PRIOR ART)

Legenda da Figura 3:

	(PRIOR ART)	(TECNOLOGIA ANTECEDENTE)
300	PROGRAM STATUS WORD (PSW)	Palavra de Status de Programa
302	(T) (translation)	conversão de bits
304	(AS) (Address Space)	Espaço de Endereço de bits

306	(T) (translation)= 1	(T) = 1 (efectou-se conversação)
308, 312, 316	AS = 0,0; AS = 0,1; AS = 1,0	Diferentes situações para o Espaço de Endereço de bits
310	PRIMARY ASCE	ASCE PRIMÁRIO
314	ACCESS REGISTER ASCE	REGISTADOR DE ACESSO DE ASCE
318	SECONDARY ASCE	ASCE SECUNDÁRIO
322	HOME ASCE	PÁGINA DE ROSTO DE ASCE
326	REAL ADDRESS	ENDEREÇO REAL

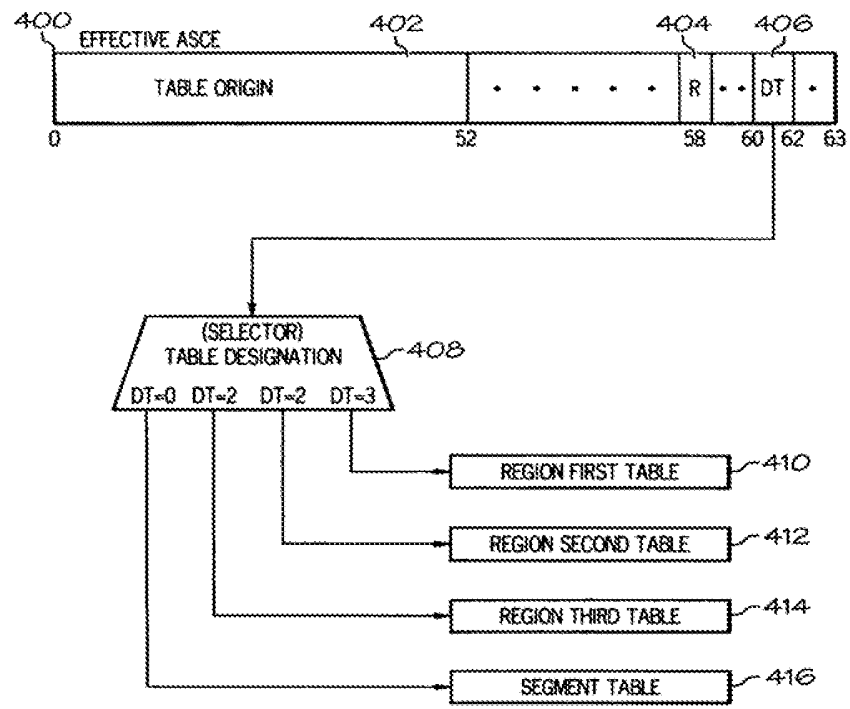


FIG. 4
(PRIOR ART)

Legenda da Figura 4:

	(PRIOR ART)	(TECNOLOGIA ANTECEDENTE)
400	EFFECTIVE ASCE	ASCE EFECTIVO
402	TABLE ORIGIN	ORIGEM DE TABELA
404	(R) - Real	Bit de controlo de espaço Real.
406	(DT) - Designation type = 1	Bit de Tipo de Designação = 1
408	(SELECTOR) TABLE DESIGNATION	(SELECTOR) DESIGNAÇÃO DE TABELA
410	REGION FIRST TABLE	TABELA DE PRIMEIRA REGIÃO
412	REGION SECOND TABLE	TABELA DE SEGUNDA REGIÃO
414	REGION THIRD TABLE	TABELA DE TERCEIRA REGIÃO
416	SEGMENT TABLE	TABELA DE SEGMENTO

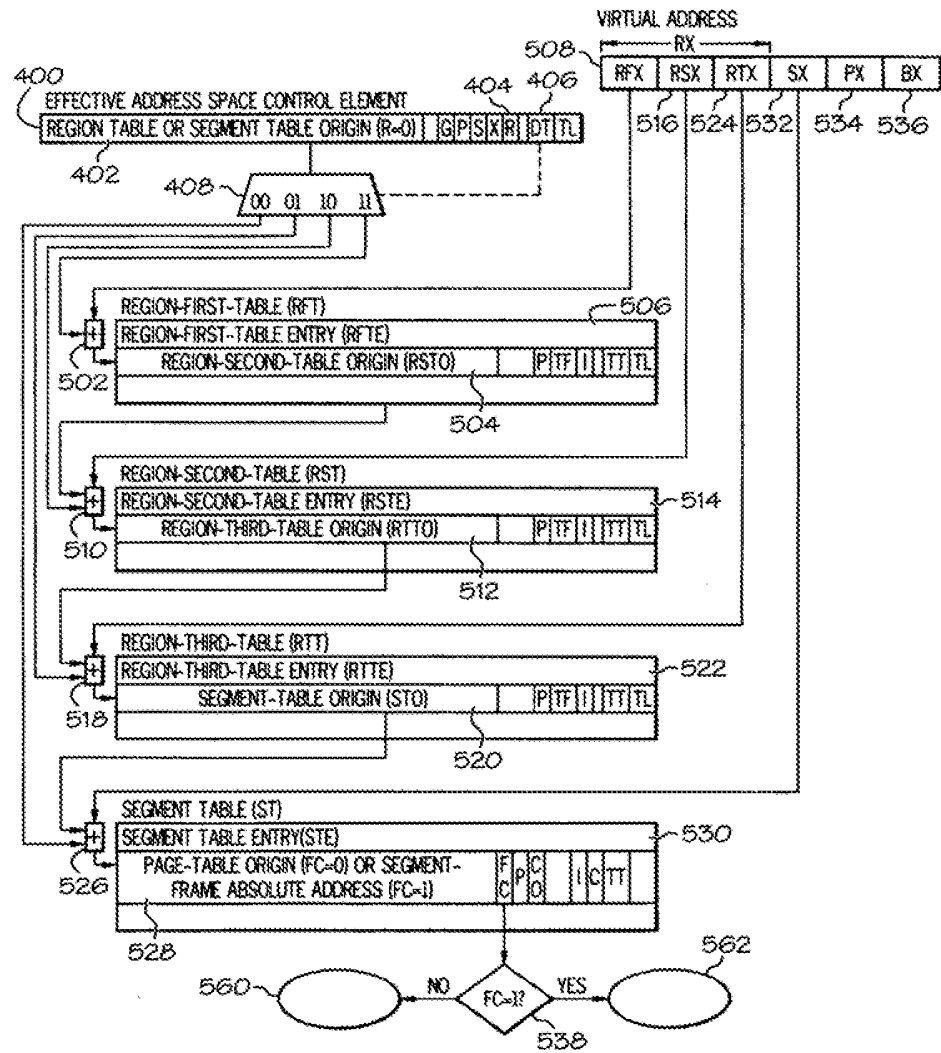


FIG. 5A

Legenda da Figura 5A:

400	EFFECTIVE ADDRESS SPACE CONTROL ELEMENT (ASCE)	ELEMENTO DE CONTROLO DE ESPAÇO DE ENDEREÇO EFECTIVO (ASCE EFECTIVO)
402	REGION TABLE OR SEGMENT TABLE ORIGIN (R=0)	ORIGEM DE TABELA DE REGIÃO OU DE TABELA DE SEGMENTO (R=0)
404	(R) - Real	Bit de controlo de espaço Real.
406	(DT) - Designation type	Bit de Tipo de Designação
408	(SELECTOR)	(SELECTOR)
	TABLE DESIGNATION	DESIGNAÇÃO DE TABELA

502, 510, 518, 526,	(arithmetical addition)	(adição aritmética)
	REGION-FIRST-TABLE (RFT)	TABELA DE PRIMEIRA REGIÃO (RFT)
504	REGION-SECOND-TABLE ORIGIN (RSTO)	ORIGEM DE TABELA DE SEGUNDA REGIÃO (RSTO)
506	REGION-FIRST-TABLE ENTRY (RFTE)	ENTRADA DE TABELA DE PRIMEIRA REGIÃO (RFTE)
(508)	VIRTUAL ADDRESS	ENDEREÇO VIRTUAL
	REGION-SECOND-TABLE (RST)	TABELA DE SEGUNDA REGIÃO (RST)
512	REGION-THIRD-TABLE ORIGIN (RTTO)	ORIGEM DE TABELA DE TERCEIRA REGIÃO (RTTO)
514	REGION-SECOND-TABLE ENTRY (RSTE)	ENTRADA DE TABELA DE SEGUNDA REGIÃO (RSTE)
	REGION-THIRD-TABLE (RTT)	TABELA DE TERCEIRA REGIÃO (RTT)
520	SEGMENT-TABLE ORIGIN (STO)	ORIGEM DE TABELA DE SEGMENTO (STO)
522	REGION-THIRD-TABLE ENTRY (RTTE)	ENTRADA DE TABELA DE TERCEIRA REGIÃO (RTTE)
	SEGMENT TABLE (ST)	TABELA DE SEGMENTO (ST)
528	PAGE-TABLE ORIGIN (FC=0) OR SEGMENT - FRAME ABSOLUTE ADDRESS (FC=1)	ORIGEM DE TABELA DE PÁGINA (FC=0) OU ENDEREÇO ABSOLUTO DE TRAMA DE SEGMENTO (FC=1)
530	SEGMENT TABLE ENTRY (STE)	ENTRADA DE TABELA DE SEGMENTO (STE)
538	FC = 1 ?	FC = 1 ? (Controlo de formato = 1 ?)

PE2248025

9/30

NO

NÃO

YES

SIM

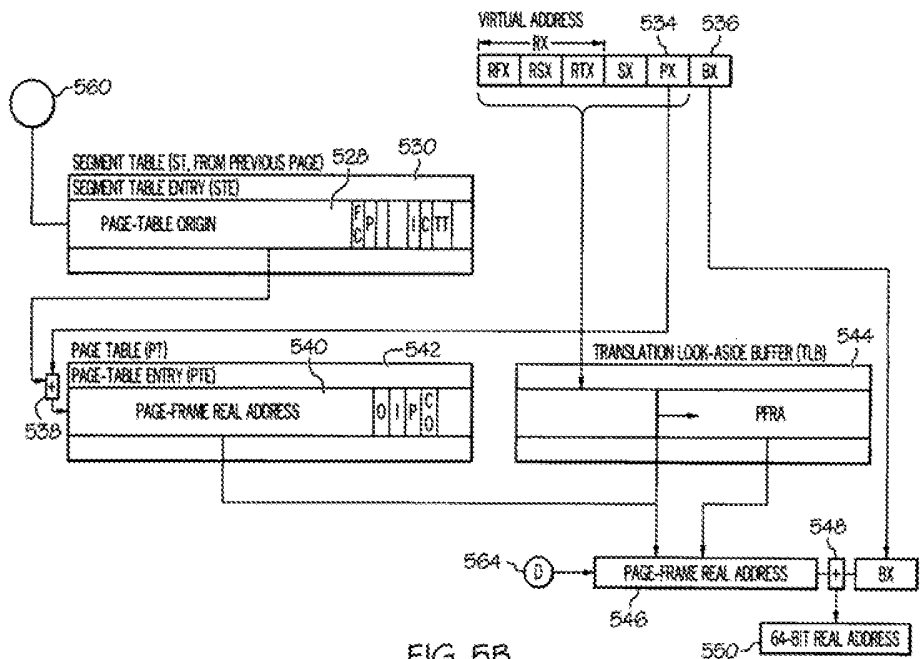


FIG. 5B

Legenda da Figura 5B:

	SEGMENT TABLE (ST, FROM PREVIOUS PAGE)	TABELA DE SEGMENTO (ST, DA PÁGINA ANTERIOR (prévia))
528	PAGE-TABLE ORIGIN	ORIGEM DE TABELA DE PÁGINA
530	SEGMENT TABLE ENTRY (STE)	ENTRADA DE TABELA DE SEGMENTO (STE)
	VIRTUAL ADDRESS	ENDEREÇO VIRTUAL
538, 548	(arithmetical addition)	(adição aritmética)
	PAGE TABLE (PT)	TABELA DE PÁGINA (PT)
540	PAGE-FRAME REAL ADDRESS	ENDEREÇO REAL DE TRAMA DE PÁGINA
542	PAGE TABLE ENTRY (PTE)	ENTRADA DE TABELA DE PÁGINA (PTE)
(544)	TRANSLATION LOOK-ASIDE BUFFER (TLB)	MEMÓRIA TAMPÃO COM TABELAS PARA CONVERSÃO (TLB)
546	PAGE-FRAME REAL ADDRESS	ENDEREÇO REAL DE TRAMA DE PÁGINA
550	64-BIT REAL ADDRESS	ENDEREÇO REAL DE 64 BITS

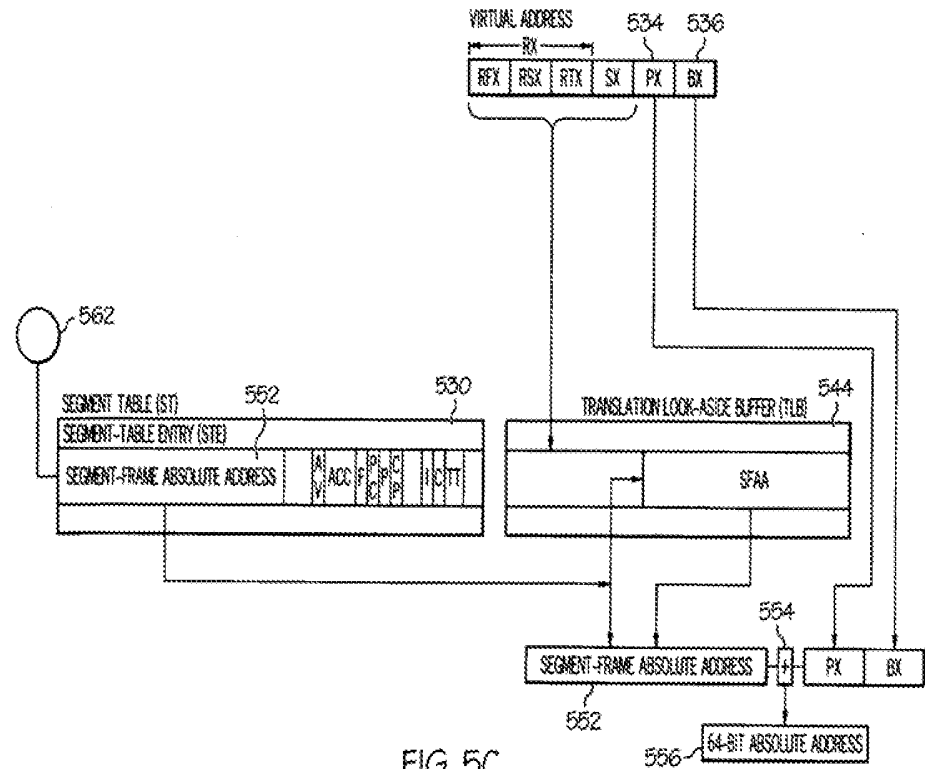


FIG. 5C

Legenda da Figura 5C:

	SEGMENT TABLE (ST)	TABELA DE SEGMENTO (ST)
530	SEGMENT TABLE ENTRY (STE)	ENTRADA DE TABELA DE SEGMENTO (STE)
552	SEGMENT-FRAME ABSOLUTE ADDRESS	ENDEREÇO ABSOLUTO DE TRAMA DE SEGMENTO
	VIRTUAL ADDRESS	ENDEREÇO VIRTUAL
(544)	TRANSLATION LOOK-ASIDE BUFFER (TLB)	MEMÓRIA TAMPÃO COM TABELAS PARA CONVERSÃO (TLB)
554	(arithmetical addition)	(adição aritmética)
552	SEGMENT-FRAME ABSOLUTE ADDRESS	ENDEREÇO ABSOLUTO DE TRAMA DE SEGMENTO
556	64-BIT REAL ADDRESS	ENDEREÇO REAL DE 64 BITS

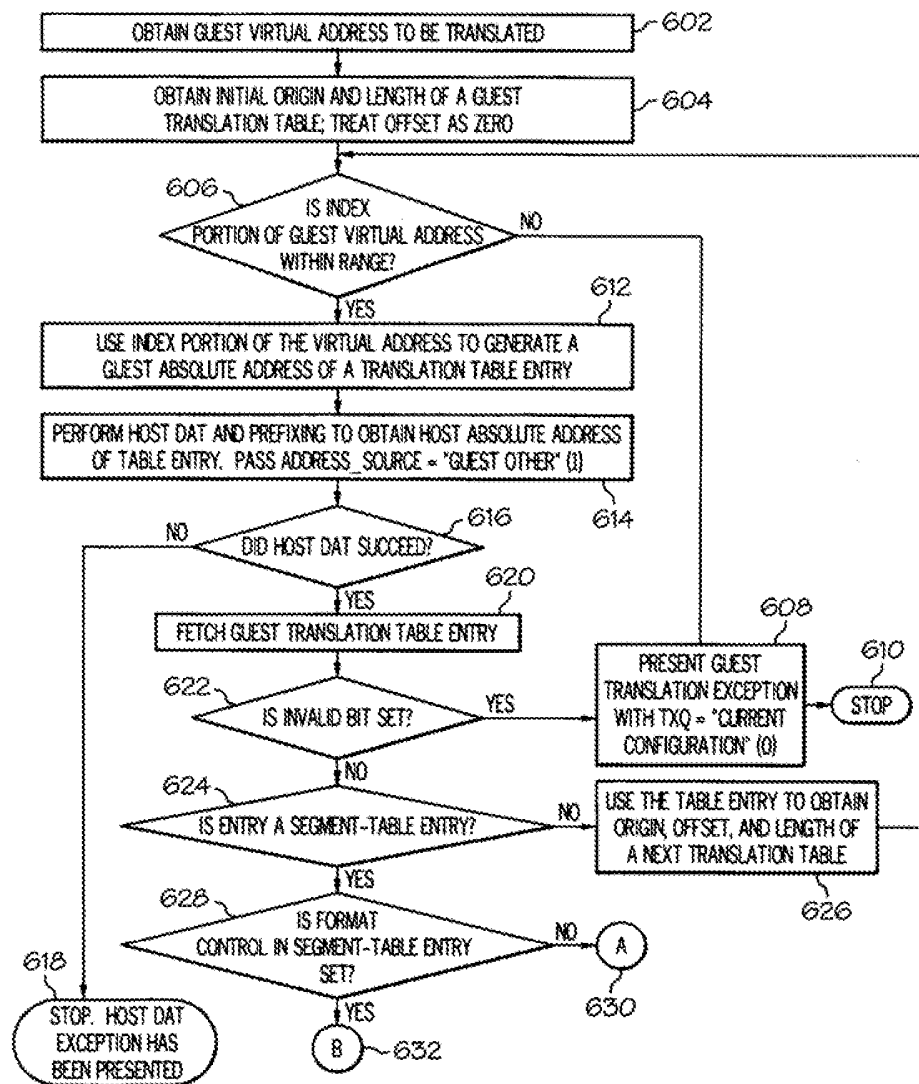


FIG. 6

Legenda da Figura 6:

602	OBTAIN GUEST VIRTUAL ADDRESS TO BE TRANSLATED	OBTER ENDEREÇO VIRTUAL DE CONVIDADO A SER CONVERTIDO
604	OBTAIN INITIAL ORIGIN AND LENGTH OF A GUEST TRANSLATION TABLE; TREAT OFFSET AS ZERO	OBTER ORIGEM E COMPRIMENTO INICIAL DE TABELA DE CONVERSÃO DE CONVIDADO; TRATAR DESLOCAMENTO COMO ZERO
606	IS INDEX PORTION OF GUEST VIRTUAL ADDRESS WITHIN RANGE?	ESTÁ A FRACÇÃO DE ÍNDICE DE ENDEREÇO DE CONVIDADO DENTRO DO INTERVALO?

608	PRESENT GUEST TRANSLATION EXCEPTION WITH TXQ = "CURRENT CONFIGURATION" (0)	PRESENÇA DE EXCEPÇÃO DE CONVERSÃO DE CONVIDADO COM TXQ = "CONFIGURAÇÃO CORRENTE" (0)
610	STOP YES NO	PARAR SIM NÃO
612	USE INDEX PORTION OF A VIRTUAL ADDRESS TO GENERATE A GUEST ABSOLUTE ADDRESS OF A TRANSLATION TABLE ENTRY	UTILIZA FRACÇÃO DE ÍNDICE DE UM ENDEREÇO VIRTUAL PARA GERAR UM ENDEREÇO DE ABSOLUTO DE UMA ENTRADA DE TABELA DE CONVERSÃO
614	PERFORM HOST DAT AND PREFIXING TO OBTAIN HOST ABSOLUTE ADDRESS OF TABLE ENTRY. PASS ADDRESS_SOURCE = "GUEST OTHER" (1)	EFECTUAR DAT E PREFIXAÇÃO DE HOSPEDEIRO PARA OBTER ENDEREÇO ABSOLUTO DE HOSPEDEIRO DE ENTRADA DE TABELA. PASSAR FONTE DE ENDEREÇO = "OUTRO CONVIDADO" (1)
616	DID HOST DAT SUCCEED?	A DAT DE CONVIDADO FOI BEM SUCEDIDA?
618	STOP. HOST DAT EXCEPTION HAS BEEN PRESENTED	PARAR.EXCEPÇÃO DE DAT DE CONVIDADO FOI APRESENTADA
620	FETCH GUEST TRANSLATION TABLE ENTRY	BUSCA DE ENTRADA DE TABELA DE CONVIDADO
622	IS INVALID BIT SET?	O BIT DE INVÁLIDO ESTÁ DEFINIDO?
624	IS ENTRY A SEGMENT-TABLE ENTRY SET?	ESTÁ DEFINIDA A ENTRADA DE UMA ENTRADA DE TABELA DE SEGMENTO?
626	USE THE ENTRY TO OBTAIN ORIGIN, OFFSET AND LENGTH OF A NEXT TRANSLATION TABLE	UTILIZAR A ENTRADA PARA OBTER A ORIGEM, O DESLOCAMENTO E O COMPRIMENTO DE UMA TABELA DE CONVERS~SO SEGUINTE
628	IS FORMAT CONTROL IN SEGMENT-TABLE ENTRY SET?	O CONTROLO DE FORMATO NA ENTRADA DA TABELA DE SEGMENTO ESTÁ DEFINIDO?

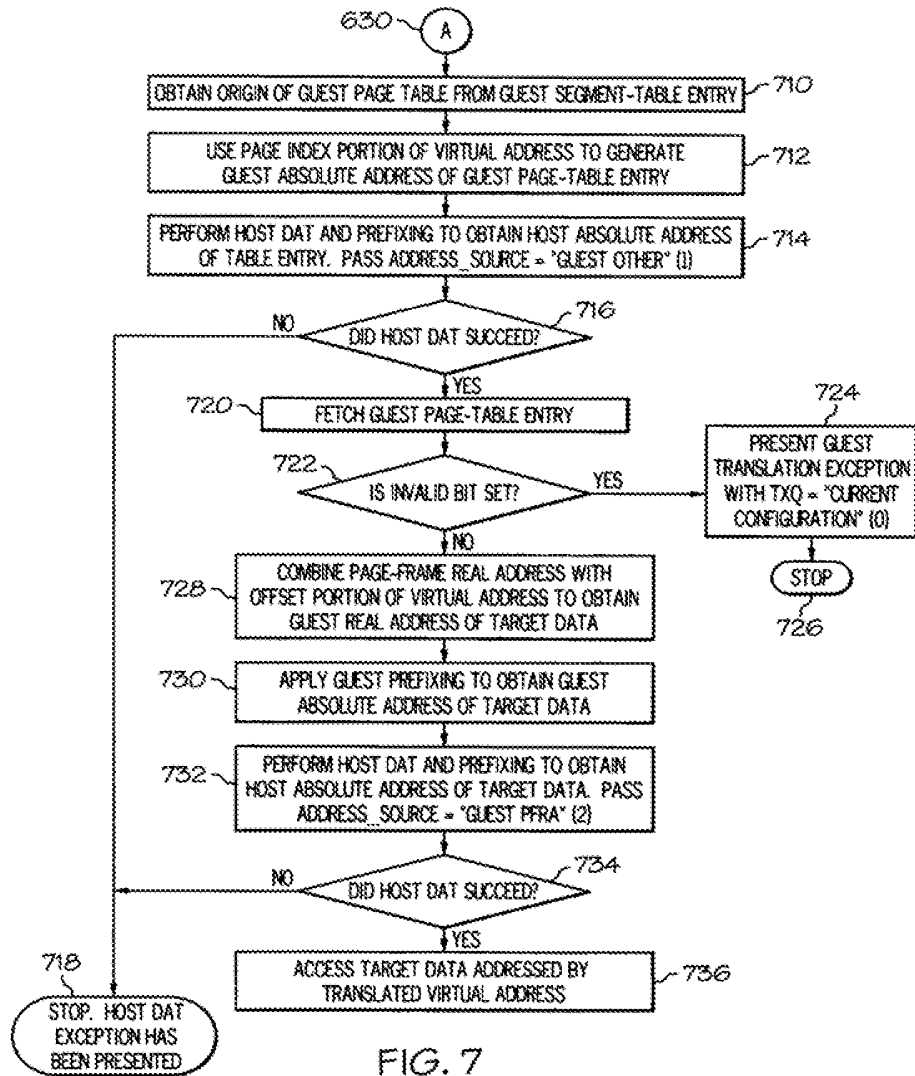


FIG. 7

Legenda da Figura 7:

710	OBTAIN ORIGIN OF GUEST PAGE TABLE FROM GUEST SEGMENT-TABLE ENTRY	OBTENIR ORIGEM ENDEREÇO VIRTUAL DE CONVIDADO A SER CONVERTIDO
712	USE PAGE INDEX PORTION OF VIRTUAL ADDRESS TO GENERATE GUEST ABSOLUTE ADDRESS OF GUEST PAGE-TABLE ENTRY	USAR FRACÇÃO DE ÍNDICE DE ENDEREÇO VIRTUAL PARA GERAR ENDEREÇO ABSOLUTO DE CONVIDADO DE ENTRADA DE TABELA DE PÁGINA DE CONVIDADO
714	PERFORM HOST DAT AND PREFIXING TO OBTAIN HOST ABSOLUTE ADDRESS OF TABLE	EFECTUAR DAT E PREFIXAÇÃO DE HOSPEDEIRO PARA OBTENIR ENDEREÇO ABSOLUTO DE HOSPEDEIRO DE ENTRADA

	ENTRY. PASS ADDRESS_SOURCE = "GUEST OTHER" (1)	DE TABELA. PASSAR FONTE DE ENDEREÇO = "OUTRO CONVIDADO" (1)
716	DID HOST DAT SUCCEED?	A DAT DE CONVIDADO FOI BEM
734		SUCEDIDA?
718	STOP. HOST DAT EXCEPTION HAS BEEN PRESENTED	PARAR.EXCEPÇÃO DE DAT DE CONVIDADO FOI APRESENTADA
	YES	SIM
	NO	NÃO
720	FETCH GUEST PAGE-TABLE ENTRY	OBTER ENTRADA DE TABELA DE PÁGINA DE CONVIDADO.
722	IS INVALID BIT SET?	O BIT DE INVÁLIDO ESTÁ DEFINIDO?
724	PRESENT GUEST TRANSLATION EXCEPTION WITH TXQ = "CURRENT CONFIGURATION" (0)	PRESENÇA DE EXCEPÇÃO DE CONVERSÃO DE CONVIDADO COM TXQ = "CONFIGURAÇÃO CORRENTE" (0)
726	STOP	PARAR
728	COMBINE PAGE-FRAME REAL ADDRESS WITH OFFSET PORTION OF VIRTUAL ADDRESS TO OBTAIN GUEST REAL ADDRESS OF TARGET DATA	COMBINAR ENDEREÇO REAL DE TRAMA DE PÁGINA COM FRACÇÃO DE DESLOCAMENTO DO ENDEREÇO VIRTUAL PARA OBTER ENDEREÇO REAL DE CONVIDADO DE DADOS DE DESTINO
730	APPLY GUEST PREFIXING TO OBTAIN GUEST ABSOLUTE ADDRESS OF TARGET DATA	APLICAR PREFIXAÇÃO DE CONVIDADO PARA OBTER ENDEREÇO ABSOLUTO DE CONVIDADO DE DADOS DE DESTINO
732	PERFORM HOST DAT AND PREFIXING TO OBTAIN HOST ABSOLUTE ADDRESS OF TABLE DATA. PASS ADDRESS_SOURCE = "GUEST PFRA" (2)	EFECTUAR DAT E PREFIXAÇÃO DE HOSPEDEIRO PARA OBTER ENDEREÇO ABSOLUTO DE HOSPEDEIRO DE ENTRADA DE TABELA. PASSAR FONTE DE ENDEREÇO = "PFRA DE CONVIDADO" (2)
736	ACCESS TARGET DATA ADDRESS BY TRANSLATED VIRTUAL ADDRESS	ENDEREÇO DE DADOS DE DESTINO DE ACESSO PELO ENDEREÇO VIRTUAL CONVERTIDO

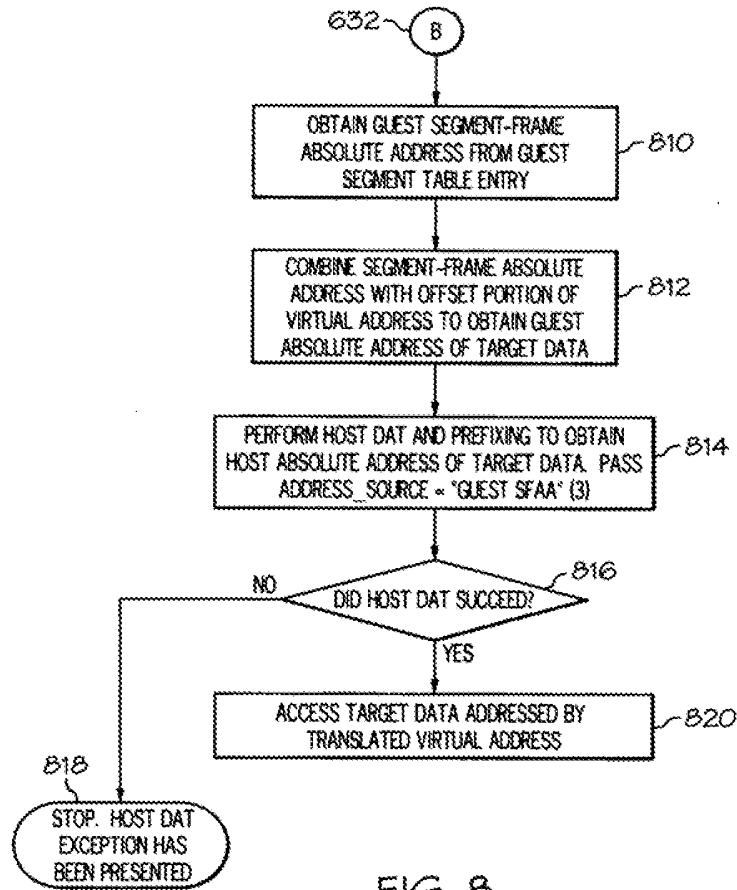


FIG. 8

Legenda da Figura 8:

810	OBTAIN GUEST SEGMENT-FRAME ABSOLUTE ADDRESS FROM GUEST SEGMENT TABLE ENTRY	OBTER ORIGEM ENDEREÇO ABSOLUTO DE TRAMA DE SEGMENTO DE CONVIDADO
812	COMBINE SEGMENT-FRAME ABSOLUTE ADDRESS WITH OFFSET PORTION OF VIRTUAL ADDRESS TO OBTAIN GUEST ABSOLUTE ADDRESS OF TARGET DATA	COMBINAR ENDEREÇO ABSOLUTO DE TRAMA DE SEGMENTO COM FRACÇÃO DE DESLOCAMENTO DO ENDEREÇO VIRTUAL PARA OBTER ENDEREÇO ABSOLUTO DE CONVIDADO DE DADOS DE DESTINO
814	PERFORM HOST DAT AND PREFIXING TO OBTAIN HOST ABSOLUTE ADDRESS OF TARGET DATA. PASS ADDRESS_SOURCE = "GUEST SFAA" (3)	EFECTUAR DAT E PREFIXAÇÃO DE HOSPEDEIRO PARA OBTER ENDEREÇO ABSOLUTO DE HOSPEDEIRO DE DADOS DE DESTINO. PASSAR FONTE DE ENDEREÇO = "SFAA DE CONVIDADO" (3)

816	DID HOST DAT SUCCEED?	A DAT DE CONVIDADO FOI BEM ' SUCEDIDA?
818	STOP. HOST DAT EXCEPTION HAS BEEN PRESENTED	PARAR.EXCEPÇÃO DE DAT DE CONVIDADO FOI APRESENTADA
	YES	SIM
	NO	NÃO
820	ACCESS TARGET DATA ADDRESS BY TRANSLATED VIRTUAL ADDRESS	ENDEREÇO DE DADOS DE DESTINO DE ACESSO PELO ENDEREÇO VIRTUAL CONVERTIDO

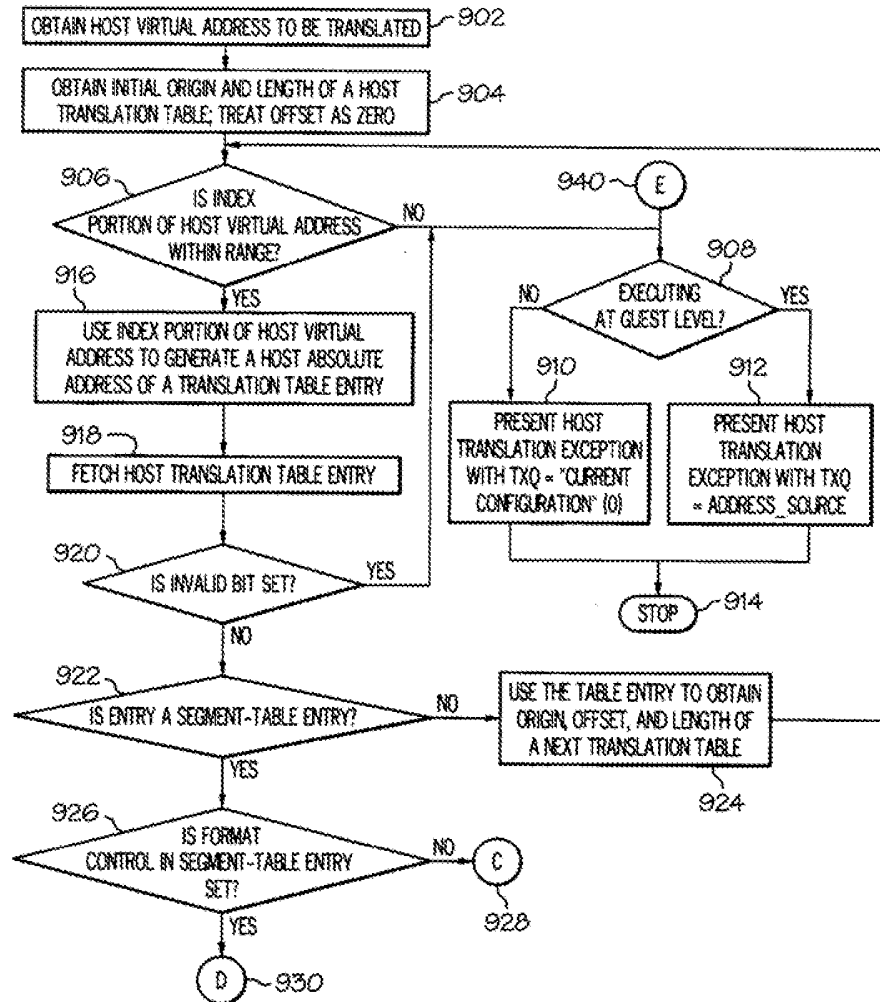


FIG. 9

Legenda da Figura 9:

902	OBTAIN HOST VIRTUAL ADDRESS TO BE TRANSLATED	OBTER ENDEREÇO VIRTUAL DE HOSPEDEIRO A SER CONVERTIDO
904	OBTAIN INITIAL ORIGIN AND LENGTH OF A HOST TRANSLATION TABLE; TREAT OFFSET AS ZERO	OBTER ORIGEM E COMPRIMENTO INICIAL DE TABELA DE CONVERSÃO DE HOSPEDEIRO; TRATAR DESLOCAMENTO COMO ZERO
906	IS INDEX PORTION OF HOST VIRTUAL ADDRESS WITHIN RANGE?	ESTÁ A FRACÇÃO DE ÍNDICE DE ENDEREÇO DE HOSPEDEIRO DENTRO DO INTERVALO?

908	EXECUTING AT GUEST LEVEL?	EXECUÇÃO PARA NÍVEL DE CONVIDADO?
910	PRESENT HOST TRANSLATION EXCEPTION WITH TXQ = "CURRENT CONFIGURATION" (0)	APRESENTAR EXCEPÇÃO DE CONVERSÃO DE HOSPEDEIRO COM TXQ = "CONFIGURAÇÃO CORRENTE" (0)
	YES	SIM
	NO	NÃO
912	PRESENT HOST TRANSLATION EXCEPTION WITH TXQ = ADDRESS_SOURCE	APRESENTAR EXCEPÇÃO DE CONVERSÃO DE HOSPEDEIRO COM TXQ = FONTE DE ENDEREÇO
914	STOP	PARAR
916	USE INDEX PORTION OF HOST VIRTUAL ADDRESS TO GENERATE A HOST ABSOLUTE ADDRESS OF A TRANSLATION TABLE ENTRY	USAR A FRACÇÃO DE ÍNDE ENDEREÇO VIRTUAL DE HOSPEDEIRO PARA GERAR UM ENDEREÇO ABSOLUTO DE HOSPEDEIRO DE UMA ENTRADA DE TABELA DE CONVERSÃO
918	FETCH HOST TRANSLATION TABLE ENTRY	OBTER ENTRADA DE TABELA DE CONVERSÃO DE HOSPEDEIRO
920	IS INVALID BIT SET?	O BIT DE INVÁLIDO ESTÁ DEFINIDO?
922	IS ENTRY A SEGMENT-TABLE ENTRY?	É ENTRADA A ENTRADA DE UMA DE TABELA DE SEGMENTO?
924	USE THE TABLE ENTRY TO OBTAIN ORIGIN, OFFSET AND LENGTH OF A NEXT TRANSLATION TABLE	UTILIZAR A TABELA DE ENTRADA PARA OBTER A ORIGEM, O DESLOCAMENTO E O COMPRIMENTO DE UMA TABELA DE CONVERS~SO SEGUINTE
926	IS FORMAT CONTROL IN SEGMENT-TABLE ENTRY SET?	ESTÁ DEFINIDO O CONTROLO DE FORMATO NA ENTRADA DA TABELA DE SEGMENTO?

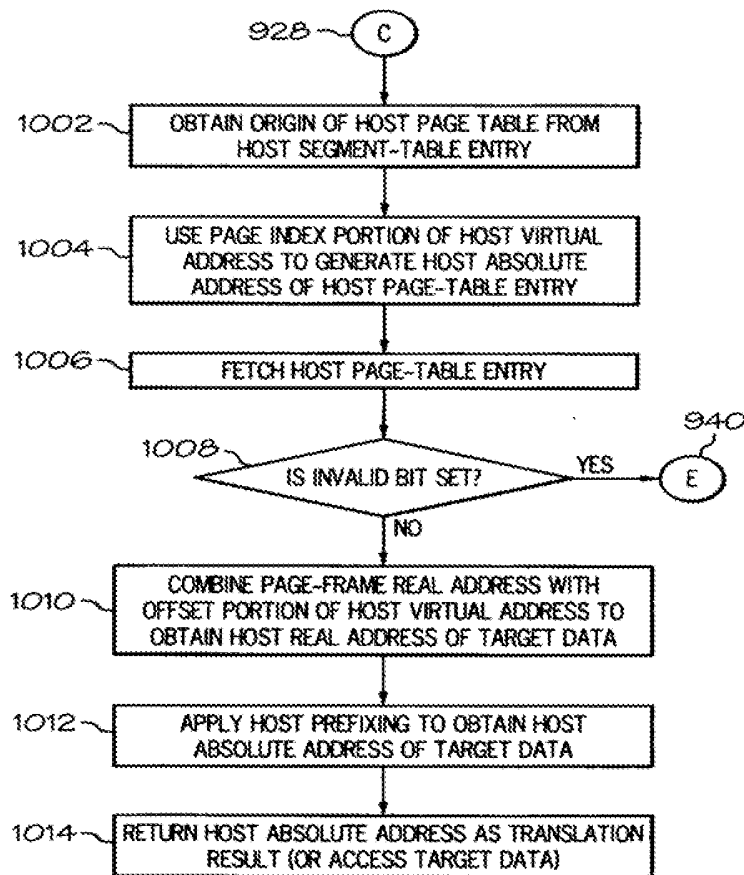


FIG. 10

Legenda da Figura 10:

1002	OBTAIN ORIGIN OF HOST PAGE TABLE FROM HOST SEGMENT-TABLE ENTRY	OBTER ORIGEM DA TABELA DE PÁGINA DE HOSPEDEIRO A PARTIR DA ENTRADA DA TABELA DE SEGMENTO DE HOSPEDEIRO
1004	USE PAGE INDEX PORTION OF HOST VIRTUAL ADDRESS TO GENERATE A HOST ABSOLUTE ADDRESS OF HOST PAGE-TABLE ENTRY	USAR FRACÇÃO DO ÍNDICE DE PÁGINA DO ENDEREÇO VIRTUAL DE HOSPEDEIRO PARA GERAR UM ENDEREÇO ABSOLUTO DE HOSPEDEIRO A PARTIR DA ENTRADA DA TABELA DE PÁGINA DE HOSPEDEIRO
1006	FETCH HOST PAGE-TABLE ENTRY	OBTER ENTRADA DE TABELA DE PÁGINA DE HOSPEDEIRO

1008	IS INVALID BIT SET?	O BIT DE INVÁLIDO ESTÁ DEFINIDO?
	YES	SIM
	NO	NÃO
1010	COMBINE PAGE-FRAME REAL ADDRESS WITH OFFSET PORTION OF HOST VIRTUAL ADDRESS TO OBTAIN HOST REAL ADDRESS OF TARGET DATA	COMBINAR O ENDEREÇO REAL DE TRAMA DE PÁGINA COM A FRACÇÃO DE DESLOCAMENTO DO ENDEREÇO VIRTUAL PARA OBTER O ENDEREÇO REAL DE HOSPEDEIRO DOS DADOS DE DESTINO
1012	APPLY HOST PREFIXING TO OBTAIN HOST ABSOLUTE ADDRESS OF TARGET DATA	APLICAR A PREFIXAÇÃO DE HOSPEDEIRO PARA OBTER O ENDEREÇO ABSOLUTO DE HOSPEDEIRO DOS DADOS DE DESTINO
1014	RETURN HOST ABSOLUTE ADDRESS AS TRANSLATION RESULT (OR ACCESS TARGET DATA)	RETORNO DO ENDEREÇO ABSOLUTO DE HOSPEDEIRO COMO RESULTADO DA CONVERSÃO (OU DADOS DE DESTINO DE ACESSO)

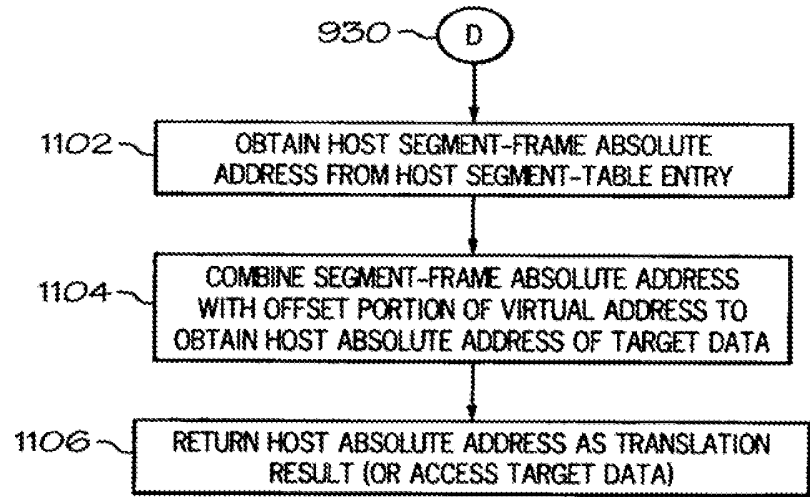
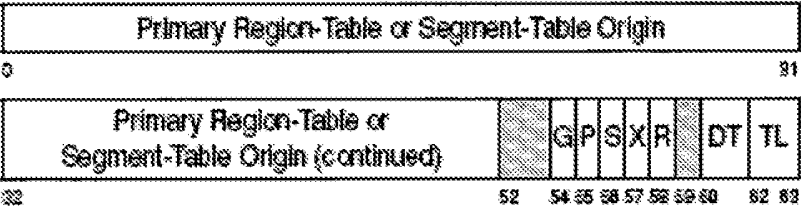


FIG. 11

Legenda da Figura 11:

1102	OBTAIN HOST SEGMENT-FRAME ABSOLUTE ADDRESS FROM HOST SEGMENT-TABLE ENTRY	OBTER ENDEREÇO ABSOLUTO DE TRAMA DE SEGMENTO A PARTIR DA ENTRADA DA TABELA DE SEGMENTO DE HOSPEDEIRO
1104	COMBINE SEGMENT-FRAME ABSOLUTE ADDRESS WITH OFFSET PORTION OF VIRTUAL ADDRESS TO OBTAIN HOST ABSOLUTE ADDRESS OF TARGET DATA	COMBINAR O ENDEREÇO ABSOLUTO DE TRAMA DE SEGMENTO COM A FRACÇÃO DE DESLOCAMENTO DO ENDEREÇO VIRTUAL PARA OBTER O ENDEREÇO ABSOLUTO DE HOSPEDEIRO DOS DADOS DE DESTINO
1106	RETURN HOST ABSOLUTE ADDRESS AS TRANSLATION RESULT (OR ACCESS TARGET DATA)	RETORNO DO ENDEREÇO ABSOLUTO DE HOSPEDEIRO COMO RESULTADO DA CONVERSÃO (OU DADOS DE DESTINO DE ACESSO)

**Primary Region-Table or
Segment-Table Designation (R=0)**



Primary Real-Space Designation (R=1)

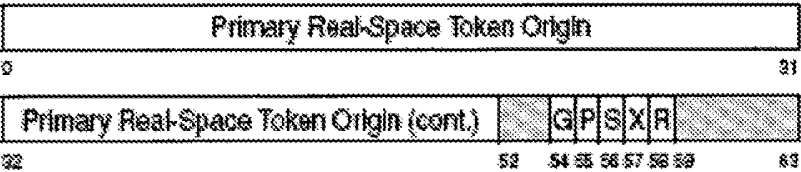
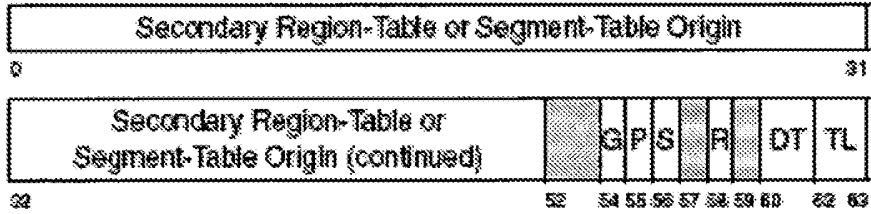


FIG. 12

Legenda da Figura 12:

Primary Region-Table or Segment-Table Designation (R = 0)	Designação de Tabela de Primeira Região ou tabela de Segmento (R = 0)
Primary Region-Table or Segment-Table Origin	Origem de Tabela de Primeira Região ou tabela de Segmento
Primary Region-Table or Segment-Table Origin (continued)	Origem de Tabela de Primeira Região ou tabela de Segmento (continuada)
Primary Real-Space Designation (R = 1)	Designação de Espaço Real Primário (R = 1)
Primary Real-Space Token Origin	Origem de testemunho de verificação de Espaço Real Primário
Primary Real-Space Token Origin (cont.)	Origem de testemunho de verificação de Espaço Real Primário (continuada)

**Secondary Region-Table or
Segment-Table Designation (R=0)**



Secondary Real-Space Designation (R=1)

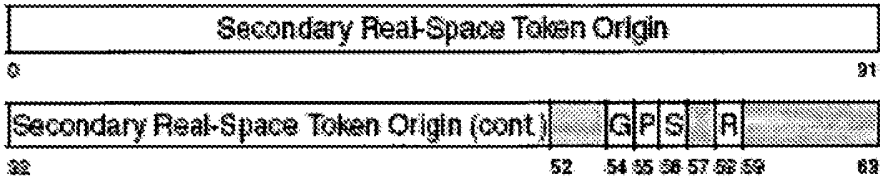
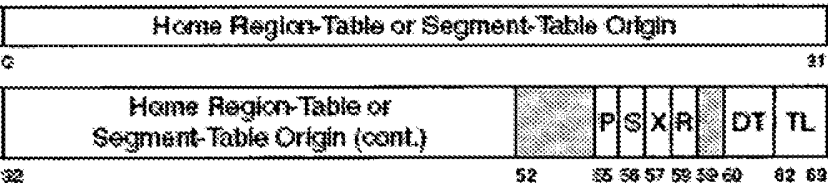


FIG. 13

Legenda da Figura 13:

Secondary Region-Table or Segment-Table Designation (R = 0)	Designação de Tabela de Segunda Região ou tabela de Segmento (R = 0)
Secondary Region-Table or Segment-Table Origin	Origem de Tabela de Segunda Região ou tabela de Segmento
Secondary Region-Table or Segment-Table Origin (continued)	Origem de Tabela de Segunda Região ou tabela de Segmento (continuada)
Secondary Real-Space Designation (R = 1)	Designação de Espaço Real Secundário (R = 1)
Secondary Real-Space Token Origin	Origem de Testemunho de Verificação de Espaço Real Secundário
Secondary Real-Space Token Origin (cont.)	Origem de Testemunho de Verificação de Espaço Real Secundário (continuada)

**Home Region-Table or
Segment-Table Designation (R=0)**



Home Real-Space Designation (R=1)

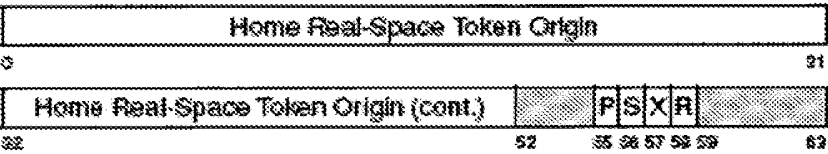


FIG. 14

Legenda da Figura 14:

Home Region-Table or Segment-Table Designation (R = 0)	Designação de Região de Tabela de Página Principal ou Tabela de Segmento (R = 0)
Home Region-Table or Segment-Table Origin	Origem de Tabela de Região de Página Principal ou Tabela de Segmento
Home Region-Table or Segment-Table Origin (continued)	Origem de Tabela de Região de Página Principal ou Tabela de Segmento (continuada)
Home Real-Space Designation (R = 1)	Designação de Espaço Real de Página Principal (R = 1)
Home Real-Space Token Origin	Origem de Testemunho de Verificação de Espaço Real de Página Principal
Home Real-Space Token Origin (cont.)	Origem de Testemunho de Verificação de Espaço Real de Página Principal (continuada)

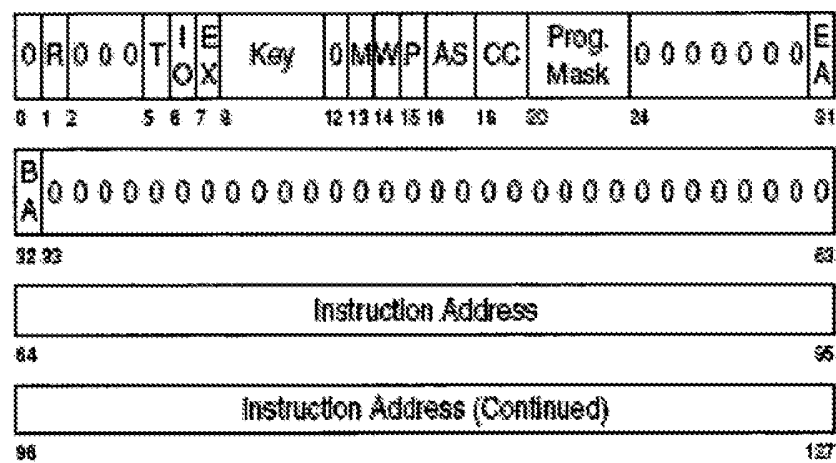


FIG. 15

Legenda da Figura 15:

Instruction Address	Endereço de Instrução
Instruction Address (Continued)	Endereço de Instrução (continuado)

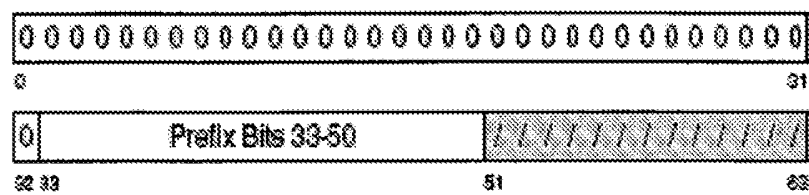
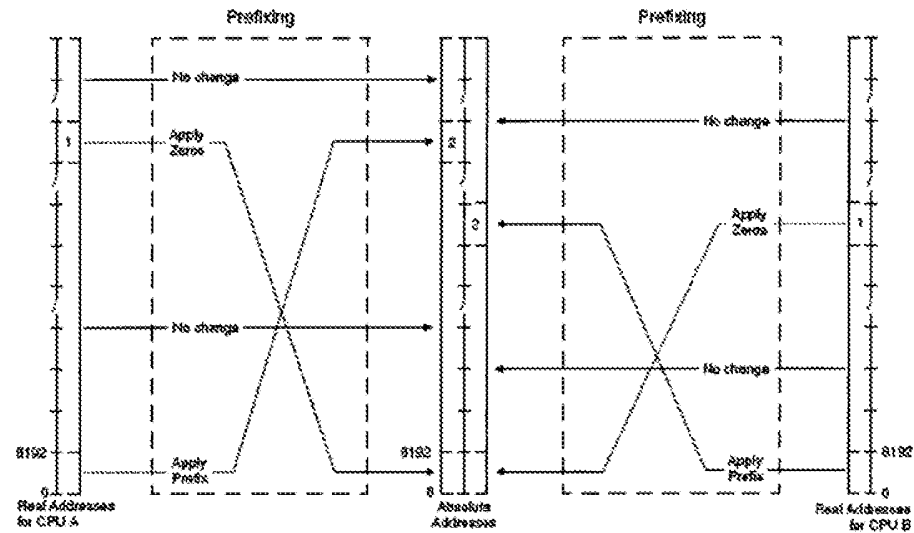


FIG. 16

Legenda da Figura 16:

Prefix Bits 33-50	Bits de Prefixo 33-50
-------------------	-----------------------



- (1) Real addresses in which bits 0-50 are equal to bits 0-50 of the prefix for this CPU (A or B).
- (2) Absolute addresses of the block that contains for this CPU (A or B) the real locations 0-8191.

FIG. 17

Legenda da Figura 17:

Prefixing	Prefixação
No change	Não Mudar
Apply zeros	Aplicar zeros
Apply Prefix	Aplicar Prefixo
Real Addresses for CPU A	Endereços Reais para CPU A
Absolute Addresses	Endereços Absoluto
Real Addresses for CPU B	Endereços Reais para CPU B
(1)Real addresses in which bits 0-50 are equal to bits 0-50 of the prefix for this CPU (A or B).	(1)Endereços reais nos quais os bits 0-50 são iguais aos bits do prefixo para esta CPU (A ou B).
(2)Absolute addresses of the block that contains for this CPU (A or B) the real locations 0-8191.	(2)Endereços absolutos do bloco que contém para esta CPU (A ou B) as reais localizações 0-8191.

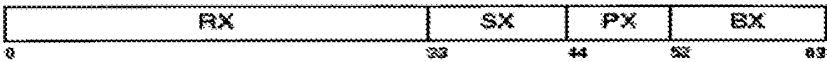


FIG. 18

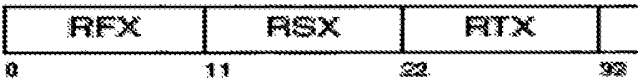
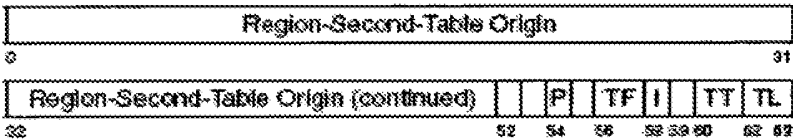
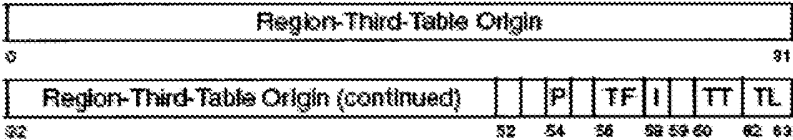


FIG. 19

Region-First-Table Entry (TT=11)



Region-Second-Table Entry (TT=10)



Region-Third-Table Entry (TT=01)

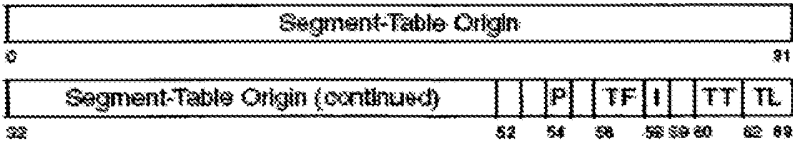


FIG. 20

Legenda da Figura 20:

Region-First-Table Entry (TT = 11)	Entrada de Tabela de Primeira Região (TT = 11)
Region-Second-Table Origin	Origem de Tabela de Segunda Região
Region-Second-Table Origin (continued)	Origem de Tabela de Segunda Região (continuada)

Region-Second-Table Entry (TT = 10)	Entrada de Tabela de Segunda Região (TT = 10)
Region-Third-Table Origin	Origem de Tabela de Terceira Região
Region-Third-Table Origin (continued)	Origem de Tabela de Terceira Região (continuada)
Region-Third-Table Entry (TT = 01)	Entrada de Tabela de Terceira Região (TT = 01)
Segment-Table Origin	Origem de Tabela de Segmento
Segment-Table Origin (continued)	Origem de Tabela de Segmento (continuada)

Segment-Table Entry (TT=00, FC=0)

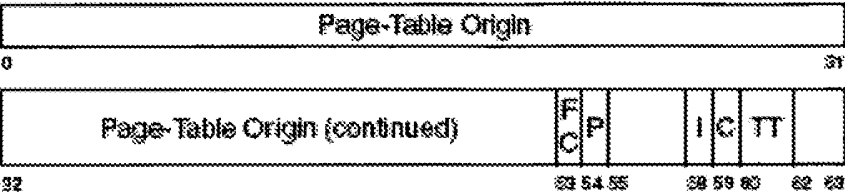


FIG. 21

Legenda da Figura 21:

Segment-Table Entry (TT = 00, FC = 0)	Entrada de Tabela de Segmento (TT = 00, FC = 0)
Page-Table Origin	Origem de Tabela de Página
Page-Table Origin (continued)	Origem de Tabela de Página (continuada)

Segment-Table Entry (TT=00, FC=1)

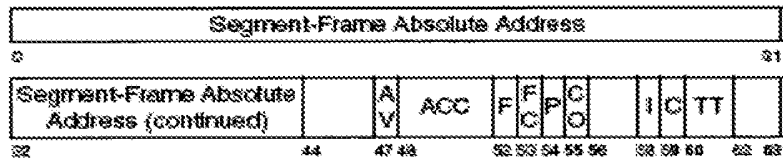


FIG. 22

Legenda da Figura 22:

Segment-Table Entry (TT = 00, FC = 1)	Entrada de Tabela de Segmento (TT = 00, FC = 1)
Segment-Frame Absolute Address	Endereço Absoluto de Trama de Segmento
Segment-Frame Absolute Address (continued)	Endereço Absoluto de Trama de Segmento (continuado)

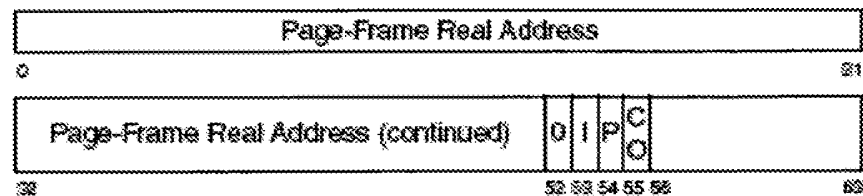


FIG. 23

Legenda da Figura 23:

Page-Frame Real Address	Endereço Real de Trama de Página
Page-Frame Real Address (continued)	Endereço Real de Trama de Página (continuado)

REFERÊNCIAS CITADAS NA DESCRIÇÃO

A lista de referências citadas pelo requerente é apresentada somente para conveniência do leitor. Ela não faz parte do documento de patente Europeia. Embora tendo havido um grande cuidado na compilação das referências, os erros e omissões não estarão completamente excluídos, e o European Patent Office - EPO descarta qualquer responsabilidade a este respeito.

Documentos de Patente citados na descrição:

• US 2004/024953 A1	• "Arquitectura Estendida do System/370 IBM [®] ", IBM [®] Pub. No. SA22-7095 (1985)
• z/Arquitectura [®] oferecido pela International Business Machines Corporation (IBM), Armonk, Nova York. A z/Architecture [®]	• "z/Architecture [®] Principles of Operation" IBM [®] Pub. No. SA22-7832-05, 6 ^a Edição, (Abril 2007)
• "Z/VM: Running Guest Operating Systems", IBM [®] Pub No. SC24-5997-02, (2001)	• Patentes dos Estados Unidos, incluídas: 5.551.013, 5.574.873, 5.790.825, 6.009.261, 6.308.255, e 6.463.582

