(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 101952807 B
(45)授权公告日 2017.02.15

(21)申请号 200980106262.3

(22)申请日 2009.02.16

(65)同一申请的已公布的文献号
申请公布号 CN 101952807 A

(43)申请公布日 2011.01.19

(30)优先权数据
61/031,160 2008.02.25 US
12/052,675 2008.03.20 US
12/052,683 2008.03.20 US

(85)PCT国际申请进入国家阶段日
2010.08.24

(86)PCT国际申请的申请数据
PCT/EP2009/051799 2009.02.16

(87)PCT国际申请的公布数据
WO2009/106451 EN 2009.09.03

(73)专利权人 国际商业机器公司
地址 美国纽约

(72)发明人 P·K·什韦德 D·奥西谢克
L·海勒 M·法雷尔 C·小盖尼
D·格雷纳

(74)专利代理机构 北京市中咨律师事务所
11247
代理人 于静 杨晓光

(51)Int.Cl.
G06F 11/07(2006.01)

(56)对比文件
US 2004/0064618 A1,2004.04.01,

审查员 张雯

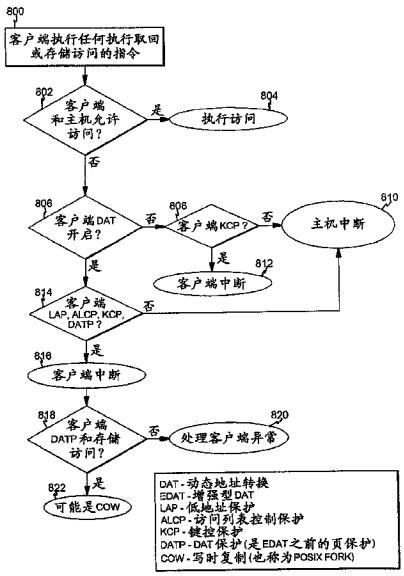
权利要求书2页 说明书30页 附图10页

(54)发明名称

管理计算环境的多个可分页客户端对存储装置的使用

(57)摘要

促进了由计算环境的可分页客户端使用的存储装置的管理。提供了一种增强的保护时抑制工具,其允许响应于所尝试的存储访问而确定何种保护级别(主机或客户端)导致错误条件。



CN 101952807 B

1.一种用于促进支持可分页客户端的计算环境的存储管理的方法,包括:

在尝试访问受主机级别保护和客户端级别保护的存储区域时检测存储保护错误,其中所述尝试访问由可分页客户端执行;

判定所检测的存储保护错误是源于所述主机级别保护还是所述客户端级别保护;以及

响应于检测到的存储保护错误是源于所述主机级别保护允许写时复制。

2.如权利要求1中所述的方法,其中在尝试访问所述存储区域的可分页客户端的执行期间检测所述存储保护错误。

3.如权利要求1或2中所述的方法,其中所尝试的访问为写访问并且所述存储区域被写保护。

4.如权利要求1或2中所述的方法,还包括:获取有关所述存储保护错误的信息,所述信息分别包括主机地址或客户端地址,具体取决于所述判定指示所述存储保护错误是源于所述主机级别保护还是所述客户端级别保护。

5.如权利要求1或2中所述的方法,其中所述尝试访问包括由可分页客户端发出尝试对所述存储区域执行存储的指令。

6.如权利要求1或2中所述的方法,其中所述方法还包括:

响应于所述判定指示所检测的存储保护错误是源于所述客户端级别保护,执行客户端中断;以及

响应于所述判定指示所检测的存储保护错误不是源于所述客户端级别保护,执行主机中断。

7.如任一上述权利要求中所述的方法,其中所述客户端级别保护为客户端DAT保护,以及所述主机级别保护为主机DAT保护。

8.一种用于支持可分页客户端的计算环境的存储的管理系统,所述管理系统包括:

用于在尝试访问受主机级别保护和客户端级别保护的存储区域时检测存储保护错误的装置;

用于判定所检测的存储保护错误是源于所述主机级别保护还是所述客户端级别保护的装置,其中所述尝试访问由可分页客户端执行;以及

源于响应于检测到的存储保护错误是源于所述主机级别保护允许写时复制的装置。

9.如权利要求8中所述的管理系统,其中在尝试访问所述存储区域的可分页客户端的执行期间检测所述存储保护错误。

10.如权利要求8或9中所述的管理系统,进一步包括:用于获取有关所述存储保护错误的信息的装置,所述信息分别包括主机地址或客户端地址,具体取决于所述判定指示所述存储保护错误是源于所述主机级别保护还是所述客户端级别保护。

11.如权利要求8或9中所述的管理系统,其中所述尝试访问包括由可分页客户端发出尝试对所述存储区域执行存储的指令。

12.如权利要求8或9中所述的管理系统,进一步包括:

用于响应于所述判定指示所检测的存储保护错误是源于所述客户端级别保护,执行客户端中断的装置;以及

用于响应于所述判定指示所检测的存储保护错误不是源于所述客户端级别保护,执行主机中断的装置。

13.一种用于执行指令的方法，包括：

获取测试保护机器指令，所述测试保护机器指令包括操作码字段、第一操作数和第二操作数；以及

由可分页客户端执行所获取的测试保护机器指令，所述执行包括：

测试由所述第一操作数指定的位置以查找一个或多个保护异常，所述测试使用所述第二操作数的键，其中保护异常指示包括所述保护异常是源于主机级别保护还是客户端级别保护的指示；以及

响应于所述测试判定所述位置的保护异常条件是源于主机级别保护而执行一个操作，以及响应于所述测试判定所述位置的保护异常是源于客户端级别保护而执行不同于该操作的另一操作。

14.如权利要求13所述的方法，其中所述一个操作包括响应于所述保护异常条件是源于主机级别保护而拦截所述测试保护机器指令的执行。

15.如权利要求14所述的方法，还包括：响应于源于主机级别保护的拦截，由分配所述可分页客户端的主机模拟所述测试保护机器指令的执行。

16.如权利要求13至15中的任一权利要求所述的方法，其中所述另一操作包括响应于所述保护异常条件是源于客户端级别保护而返回条件码。

17.如权利要求13至15中的任一权利要求所述的方法，其中所述测试保护机器指令进一步包括第二操作数，并且所述测试进一步包括使用所述第二操作数的数据，所述数据包括键。

18.一种用于执行指令的系统，包括：

用于获取测试保护机器指令的装置，所述测试保护机器指令包括操作码字段、第一操作数和第二操作数；以及

用于由可分页客户端执行所获取的测试保护机器指令的装置，所述执行包括：

测试由所述第一操作数指定的位置以查找一个或多个保护异常，所述测试使用所述第二操作数的键，其中保护异常指示包括所述保护异常是源于主机级别保护还是客户端级别保护的指示；以及

响应于所述测试判定所述位置的保护异常条件是源于主机级别保护而执行一个操作，以及响应于所述测试判定所述位置的保护异常是源于客户端级别保护而执行不同于该操作的另一操作。

19.如权利要求18所述的系统，其中所述一个操作包括响应于所述保护异常条件是源于主机级别保护而拦截所述测试保护机器指令的执行。

20.如权利要求19所述的系统，还包括：用于响应于源于主机级别保护的拦截，由分配所述可分页客户端的主机模拟所述测试保护机器指令的执行的装置。

21.如权利要求18至20中的任一权利要求所述的系统，其中所述另一操作包括响应于所述保护异常条件是源于客户端级别保护而返回条件码。

22.如权利要求18至20中的任一权利要求所述的系统，其中所述测试保护机器指令进一步包括第二操作数，并且所述测试进一步包括使用所述第二操作数的数据，所述数据包括键。

管理计算环境的多个可分页客户端对存储装置的使用

技术领域

[0001] 本发明一般地涉及支持可分页客户端的计算环境，具体地说，涉及管理计算环境的多个可分页客户端对存储装置的使用。

背景技术

[0002] 计算环境的至少一部分存储装置(在此也称为存储器)通常由该环境内执行的多个过程所共享。这种对存储装置的共享会增加危害存储装置中存储的数据的完整性的风险。因此，为了提供数据完整性，将应用特定的存储保护。这些保护用于管理存储装置，进而保护数据。

[0003] 所述保护可以应用在不同类型的计算环境中，包括那些支持解释执行和可分页客户端的环境。例如，在可分页客户端环境中，多个客户端共享同一存储装置，并且再次地，使用保护提供数据完整性并保护存储装置。但是在此类环境中，有关保护的信息对于客户端和/或执行客户端的主机而言变得非常复杂。

发明内容

[0004] 基于上述内容，需要促进多个可分页客户端所使用的存储装置的管理的能力。在一个实例中，需要指示存储保护错误是源于主机级别保护还是源于客户端级别保护的能力。

[0005] 通过提供一种包括至少一个具有用于促进支持可分页客户端的计算环境的存储管理的计算机可读程序代码逻辑的计算机可用介质的制品，克服了现有技术的缺点并提供了其他优点。所述计算机可读程序代码逻辑由计算机执行以便执行方法，所述方法例如包括：尝试访问受主机级别保护和客户端级别保护所保护的存储区域；以及判定响应于所尝试的访问而收到的错误是源于所述主机级别保护还是所述客户端级别保护。

[0006] 通过提供一种用于执行指令的计算机程序产品也克服了现有技术的缺点并提供了其他优点。所述计算机程序产品包括存储介质，其可由处理电路读取并存储由所述处理电路执行以执行方法的指令。所述方法例如包括：获取测试保护机器指令，所述测试保护机器指令包括操作码字段、第一操作数和第二操作数；以及由可分页客户端执行所获取的测试保护机器指令，所述执行例如包括：测试由所述第一操作数指定的位置以查找一个或多个保护异常，所述测试使用所述第二操作数的键(key)，其中保护异常指示包括所述保护异常是源于主机级别保护还是客户端级别保护的指示。

[0007] 此处还描述并要求保护与本发明的一个或多个方面相关的方法、系统和计算机程序。

[0008] 通过本发明的技术实现了其他特性和优点。在此将详细说明本发明的其他实施例和方面并它们被视为所要求保护的发明的一部分。

附图说明

[0009] 现在将仅通过实例的方式参考附图描述本发明的优选实施例，这些附图是：

[0010] 图1示出了结合和使用本发明的一个或多个方面的计算环境的一个实施例；

[0011] 图2示出了结合和使用本发明的一个或多个方面的仿真的计算环境的一个实施例；

[0012] 图3示出了根据本发明的一个方面使用的帧描述符的一个实例；

[0013] 图4A-4C示出了根据本发明的一个方面使用的区表表项的实例；

[0014] 图5A-5B示出了根据本发明的一个方面使用的段表表项的实例；

[0015] 图6示出了根据本发明的一个方面使用的页表表项的一个实例；

[0016] 图7示出了根据本发明的一个方面使用的页状态表表项的一个实例；

[0017] 图8A-8B示出了根据本发明的一个方面的支持保护时抑制和增强的保护时抑制工具的环境中的客户端处理逻辑的一个实施例；

[0018] 图9示出了根据本发明的一个方面的测试保护指令的格式的一个实例；

[0019] 图10A-10B示出了根据本发明的一个方面的测试保护指令的逻辑的一个实施例；以及

[0020] 图11示出了结合本发明的一个或多个方面的计算机程序产品的一个实施例。

具体实施方式

[0021] 根据本发明的一个方面，提供了一种促进计算环境的多个可分页客户端所使用的存储装置的管理的能力。作为一个实例，提供了增强的保护时抑制工具，其允许确定哪种保护级别(主机还是客户端)导致错误条件。作为另一实例，提供了查询指令(例如，Test Protection(TPROT)指令)以详细说明与查询中指示的存储区域相关的信息。

[0022] 参考图1描述了结合和使用本发明的一个或多个方面的计算环境的一个实施例。计算环境100例如基于**IBM® z/Architecture®**。在标题为“z/Architecture Principles of Operation(z/Architecture操作原理，出版号：SA22-7832-05，2007年4月)”的IBM出版物中描述了z/Architecture。在一个实例中，基于z/Architecture的计算环境包括IBM eServer™。(IBM、z/Architecture、eServer和zSeries是国际商业机器公司在美国和/或其他国家/地区的商标)。此处所用的其他名称可能是国际商业机器公司或其他公司的注册商标、商标或产品名称。

[0023] 作为一个实例，计算环境100包括提供虚拟机(VM)支持的中央处理器复合体(CPC)102。CPC 102包括例如一个或多个虚拟机104、一个或多个中央处理器106、至少一个主机108(例如，诸如系统管理程序之类的控制程序)，以及输入/输出子系统110，下面将逐一进行介绍。在该实例中，虚拟机和主机包括在存储器内。

[0024] CPC的虚拟机支持提供了运行大量虚拟机的能力，每个虚拟机都能托管诸如Linux之类的客户端操作系统112(Linux是Linux Torvalds在美国和/或其他国家/地区的注册商标)。每个虚拟机104都能作为单独的系统运行。也就是说，每个虚拟机都能被独立地重置，托管客户端操作系统和使用不同的程序运行。在虚拟机中运行的操作系统或应用程序看起来可以访问整个和完整的系统，但实际上只能访问其中的一部分。

[0025] 在该特定实例中，虚拟机的模型为V＝V模型，其中虚拟机的绝对存储器或实存储器由主机虚拟存储器(而非主机实存储器或绝对存储器)支持。每个虚拟机都具有虚拟线性

存储器空间。物理资源由主机108所拥有，主机根据需要将共享的物理资源分配给客户端操作系统以满足其处理需求。该V＝V虚拟机(即，可分页客户端)模型假设客户端操作系统与物理共享机器资源间的交互由主机控制，由于存在大量客户端，因此主机一般不是简单地执行分区并将硬件资源分配给已配置的客户端。在标题为“z/VM:Running Guest Operating Systems(z/VM:运行客户端操作系统，IBM出版号:SC24-5997-02，2001年10月)”的IBM出版物中进一步描述了V＝V模型的一个或多个方面。

[0026] 中央处理器106是可分配给虚拟机的物理处理器资源。例如，虚拟机104包括一个或多个逻辑处理器，每个逻辑处理器表示可以动态地分配给虚拟机的全部或部分物理处理器资源106。虚拟机104由主机108管理。作为实例，主机可以通过在处理器106上运行的微代码实现或作为在机器上执行的主操作系统的一部分。在一个实例中，主机108为VM系统管理程序，例如由位于纽约阿蒙克的国际商业机器公司提供的**z/VM®**(z/VM是国际商业机器公司在美国和/或其他国家/地区的商标)。在标题为“z/VM:General Information Manual(z/VM:总体信息手册，IBM出版号:GC24-5991-04，2001年10月)”的IBM出版物中描述了z/VM的一个实施例。

[0027] 输入/输出子系统110在设备和主存储装置之间引导信息流。它与中央处理复合体耦合，因为其可以是中央处理复合体的一部分或独立于中央处理复合体。I/O子系统减轻了中央处理器与耦合到CPC的I/O设备直接通信的任务并允许数据处理与I/O处理同时进行。

[0028] 在一个实施例中，主机(例如，z/VM)和处理器(例如，System**z®**)硬件/固件以可控的协作方式彼此交互，以便在无需在客户端操作系统和主机之间转移控制的情况下处理V＝V客户端操作系统操作。(System z是国际商业机器公司在美国和/或其他国家/地区的商标)。客户端操作可以在没有主机介入的情况下，通过允许针对可分页存储模式客户端解释执行指令的工具而直接执行。此工具提供了一个可由主机发出的指令-Start Interpretive Execution(SIE)，该指令指定称为状态描述的控制块，所述控制块保存客户端(虚拟机)状态和控件。所述指令将机器置于解释执行模式，在此模式中，直接处理客户端指令和中断，直到出现需要主机介入的条件。当出现此类条件时，解释执行终止，并且或者提供主机中断，或者SIE指令完成存储所遇到的条件的详细信息；此后一个操作称为拦截。在“System/370 Extended Architecture/Interpretive Execution(System/370扩展架构/解释执行，IBM出版号:SC22-7095-01，1985年9月)”中描述了解释执行的一个实例。

[0029] 在一个实例中，解释执行工具是IBM推出的Processor Resource/Systems Manager™(PR/SM)的一个元素。它允许使用单个指令(Start Interpretive Execution(SIE))在处理器上运行虚拟服务器指令流。服务器的逻辑分区(LPAR)支持使用所述SIE指令，以便例如将zSeries或**S/390®**处理器复合体分为已被证明为安全的逻辑分区。(Processor Resource/Systems Manager和S/390是国际商业机器公司在美国和/或其他国家/地区的商标)。(要指出的是，处理器PR/SM工具的IBM S/390 CMOSG6系列获得了ITSEC E4安全级别证书)。

[0030] SIE指令运行由控制程序分配的虚拟服务器，直到耗尽服务器的时间片，或直到服务器需要执行硬件无法虚拟化的操作或执行控制程序需要重新获得控制的操作。此时，SIE指令终止并将控制返回给控制程序，控制程序或者模拟所述指令，或者将虚拟服务器置于非自愿等待状态。当完成时，控制程序再次调度虚拟服务器运行，并且再次开始循环。通过

这种方式，虚拟服务器可以使用中央处理单元(CPU)的全部能力和速度。仅拦截需要控制程序的帮助或验证的那些特权指令。控制程序还使用这些SIE拦截(如已知的那样)对虚拟服务器可能针对实际设备执行的操作施加限制。

[0031] 此外，此机制使得控制程序能够限制许多类硬件或软件故障的范围。如果可以将错误隔离到特定虚拟服务器，则只是该虚拟服务器发生故障并且可以重试操作或在不影响其他虚拟服务器中运行的任何测试或生产工作的情况下重新初始化(重新引导)该虚拟服务器。所述控制程序被设计为虚拟服务器中发生的故障不会对其或其他虚拟服务器产生影响。

[0032] 图2示出了结合本发明的一个或多个方面的计算环境的另一实例。在该实例中，提供了仿真具有主架构的主计算机202的仿真的主计算机系统200。在仿真的主计算机系统200中，主处理器(CPU)204是仿真的主处理器(或虚拟主处理器)并且通过仿真处理器206(其本机指令集架构不同于主计算机202的处理器使用的本机指令集架构)实现。仿真的主计算机系统200具有可由仿真处理器206访问的存储器208。在所述实例实施例中，存储器208被分为主计算机存储器210部分和仿真例程212部分。主计算机存储器210可用于与主计算机架构一致的仿真的主计算机202的程序并且可以包括主机或系统管理程序214以及一个或多个运行客户端操作系统218的虚拟机216，它们与图1中具有相似名称的元素类似。

[0033] 仿真处理器206执行架构的指令集(其架构不同于仿真的处理器204的架构)的本机指令。所述本机指令从例如仿真例程存储器212获取。仿真处理器206可以通过使用从序列与访问/解码例程获取的一个或多个指令来访问主计算机存储器210中的程序内要执行的主机指令，所述序列与访问/解码例程可解码所访问的主机指令(多个)以确定用于仿真所访问的主机指令的功能的本机指令执行例程。一个此类主机指令可以是例如Start Interpretative Execution(SIE)指令，主机可以通过该指令在虚拟机中执行客户端程序。仿真例程212可以包括对该指令的支持，以及包括对根据该SIE指令的定义在虚拟机216中执行一系列客户端指令的支持。

[0034] 针对主计算机系统202架构定义的其他工具可以通过架构的工具例程仿真，所述架构的工具例程例如包括通用寄存器、控制寄存器、动态地址转换和I/O子系统支持以及处理器高速缓存之类的工具。仿真例程还可以利用仿真处理器206中可用的功能(例如通用寄存器和虚拟地址的动态转换)提高仿真例程的性能。还可以提供专用硬件和卸载引擎来帮助处理器206仿真主计算机202的功能。

[0035] 包括上述环境的许多类型的计算环境、通用计算机、数据处理系统以及其他环境都采用使用虚拟存储器方案组织的存储装置。虚拟存储器系统使用称为块(或页)的单位组织存储装置。这些块在速度较快的主存储器和一个或多个更大并且通常较慢的第二、第三等存储单元之间移动。块的移动(通常称为交换)对于计算环境中执行的应用或过程而言是透明的，使得应用或程序表现为就好像它们具有无限的存储量一样。

[0036] 尽管一般的虚拟存储器允许计算环境中执行的应用和/或过程表现为就好像它们可根据需要具有无限的存储量一样，但实际上，特定应用或过程可用的存储量受计算环境中存储量的限制并且进一步受共享存储装置的并行执行程序数目的限制。虚拟存储器方案对应用程序隐藏存储器的实际物理地址。应用程序使用逻辑地址(例如，虚拟地址)访问其存储器空间，所述逻辑地址然后由计算环境转换为物理地址。

[0037] 在存储引用期间将虚拟地址转换为相应的实地址或绝对地址的过程称为动态地址转换(DAT)。所述虚拟地址可以是主虚拟地址、第二虚拟地址、存取寄存器(AR)指定的虚拟地址或归属(home)虚拟地址。这些地址分别通过主地址空间控制元素、第二地址空间控制元素、AR指定的地址空间控制元素或内部地址空间控制元素进行转换。选择相应的地址空间控制元素之后,对于所有四种虚拟地址类型而言,转换过程都是相同的。DAT可以使用五到二个级别的表(区第一表、区第二表、区第三表、段表和页表)作为转换参数。增强的动态地址转换(EDAT)过程可以使用五到一个级别的表,其中针对某些或全部转换省略了页表。特定地址空间的最高级别表的指定(起始地址和长度)被称为地址空间控制元素,在控制寄存器中找到所述元素或由存取寄存器指定所述元素以供DAT使用。备选地,地址空间的地址空间控制元素可以是实空间指定,它指示DAT仅通过将虚拟地址视为实地址来转换虚拟地址而不使用任何表。

[0038] DAT在不同的时间使用不同的控制寄存器中的地址空间控制元素或使用由存取寄存器指定的地址空间控制元素。该选择由当前PSW(程序状态字)中指定的由程序指定的转换模式确定。存在四种转换模式:主空间模式、第二空间模式、存取寄存器模式以及归属空间模式。可根据转换模式寻址不同的地址空间。

[0039] 对虚拟地址执行增强的DAT可以得到实地址或绝对地址。如果是实地址,则执行添加前缀操作以获取相应的可用于引用存储器的绝对地址。添加前缀提供了针对每个CPU将实地址范围0-8191(举例说明)分配给绝对存储装置中的不同区域的能力,从而允许一个以上共享主存储装置的CPU在干扰最小的情况下并行运行,尤其是在处理中断的过程中。添加前缀可使范围0-8191中的实地址按照一对一的关系对应于通过CPU的前缀寄存器的位位置0-50中的值标识的8k绝对地址区域(前缀区域),并且可使通过前缀寄存器中的该值标识的实地址区域按照一对一的关系对应到绝对地址0-8191。其余的实地址与对应的绝对地址相同。此转换使得每个CPU都能够访问整个主存储装置,其中包括第一8KB以及由其他CPU的前缀寄存器指定的位置。

[0040] 在Gainey等人的标题为“Enhanced Dynamic Address Translation with Frame Management Function(使用帧管理功能的增强的动态地址转换,IBM案号:POU920070313US1,申请日:2008年1月11日)”的美国申请第11/972,725号中更详细地描述了动态地址转换、添加前缀和增强的DAT。

[0041] 为了便于理解本发明的一个或多个方面,将引用下面所述的各种术语和数据结构(例如,表、列表)。

[0042] 帧描述符

[0043] 帧描述符描述了主机页帧;即,能够保存虚拟存储器的一部分(页)的实存储器区域(帧)。由主机分配、取消分配和初始化帧描述符并且可以由主机页管理帮助功能(例如,如Blandy等人的标题为“Facilitating Management of Storage of a Pageable Mode Virtual Environment Absent Intervention of a Host of the Environment(促进无环境主机干预下的可分页模式虚拟环境的存储管理,申请日:2004年5月27日)”的美国申请第10/854,990中所述)更新帧描述符。

[0044] 在一个实例中,帧描述符300(图3)例如是位于32字节边界上的主机归属空间虚拟存储装置中的32字节块并包括以下字段,例如:

[0045] (a)下一帧描述符地址302：在一个实例中，该字段的内容(右侧附加5个0)指定列表中下一帧描述符的主机归属空间虚拟地址。0值指示帧描述符位于列表的末尾。

[0046] 该字段由主机进行初始化并且可以由主机或主机页管理帮助功能更改。

[0047] (b)页帧实地址或页表表项(PTE)副本304：当帧描述符位于可用帧描述符列表(AFDL)中时，该字段的内容(右侧附加12个0)指定可供分配以提供主机存储的主机帧的第一字节(字节0)的主机实地址。

[0048] 当帧描述符位于已处理帧描述符列表(PFDL)中时，该字段包括由页表表项地址字段指定的PTE的副本，好像其在解析主机页之前出现那样。

[0049] 该字段由主机进行初始化并且可以由主机或主机页管理帮助功能更改。

[0050] (c)页表表项地址306：当帧描述符位于已处理帧描述符列表中时，该字段的内容(右侧附加3个0)指定主机虚拟页的页表表项的主机实地址或主机绝对地址。

[0051] 该字段由主机进行初始化并且可以由主机或主机页管理帮助功能更改。

[0052] 多个帧描述符可以相互链接形成一个列表，例如可用帧描述符列表(AFDL)或已处理帧描述符列表(PFDL)。帧描述符存在于这两个列表之一中。为每个CPU提供由这些列表构成的单独列表对。通过CPU的前缀区域中的字段指定CPU的AFDL和PFDL的起始地址。

[0053] 可用帧描述符列表(AFDL)是描述主机已清除并可供分配给主机页的主机帧的帧描述符列表。由指定的主机实地址处的AFDL起始地址(AFDLO)指定AFDL。

[0054] AFDLO的内容(右侧附加5个0)指定AFDL中的第一帧描述符的主机归属空间虚拟地址。0值指示该列表为空。

[0055] AFDLO由主机进行初始化并且可以由主机或主机页管理帮助功能更改。在一个实施例中，通过非互锁更新操作更改AFDLO。

[0056] 已处理帧描述符列表(PFDL)是描述已用于在客户端解释期间解析主机页无效条件的主机帧的帧描述符列表。由PFDL描述的主机帧已被分配给为客户端提供存储的主机页。由指定的主机实地址处的PFDL起始地址(PFDLO)指定PFDL。PFDLO的内容(右侧附加5个0)指定PFDL中的第一帧描述符的主机归属空间虚拟地址。0值指示该列表为空。

[0057] PFDLO由主机进行初始化并且可以由主机或主机页管理帮助功能更改。在一个实施例中，通过维护列表完整性的双字并发互锁更新操作更改PFDLO。

[0058] 区表表项

[0059] 术语“区表表项”指示区第一表表项、区第二表表项或区第三表表项。包含表项的表的级别(第一、第二或第三)由表项中的表类型(TT)位进行标识。图4A-4C中示出了从区第一表、区第二表和区第三表取回的表项的格式实例。具体地说，图4A示出了区第一表表项400的格式的一个实施例；图4B示出了区第二表表项430的格式的一个实施例；以及图4C示出了区第三表表项460的格式的一个实施例。

[0060] 作为实例，三个级别的区表表项中的字段按如下方式进行分配：

[0061] 区第二表起始地址402、区第三表起始地址432和段表起始地址462：区第一表表项包括区第二表起始地址。区第二表表项包括区第三表起始地址。区第三表表项包括段表起始地址。以下说明适用于全部三种起始地址。在一个实例中，表项的位0-51(右侧附加12个0)形成指定下一低级表的开头的64位地址。

[0062] DAT保护位(P)406、436、466：当应用增强的DAT时，位54被视为与转换中使用的每

个后续区表表项、段表表项以及页表表项(当适用时)中的DAT保护位进行“或”。因此，当该位为1时，将DAT保护应用于由区表表项指定的整个区。当未安装增强的DAT工具，或者当已安装此工具但是增强的DAT启用控制为0时，将忽略区表表项的位54。

[0063] 区第二表偏移408、区第三表偏移438和段表偏移(TF)468：区第一表表项包括区第二表偏移。区第二表表项包括区第三表偏移。区第三表表项包括段表偏移。以下说明适用于全部三种偏移。表项的位56和57指定在表开头处缺失的下一低级表部分的长度；也就是说，所述位指定下一低级表中实际存在的第一表项的位置。所述位以4,096字节为单位指定缺失部分的长度，从而使缺失部分的长度以512个表项的倍数变化。以4,096字节为单位表示的缺失部分的长度等于TL值。偏移字段的内容与长度字段(位62和63)一起用于确定要通过下一低级表转换的虚拟地址部分(RSX、RTX或SX)是否指定表中实际存在的表项。

[0064] 区无效位(I)410、440、470：区是例如虚拟地址的2GB的连续范围。区第一表表项或区第二表表项中的位58控制与所述表项关联的一组区是否可用。区第三表表项中的位58控制与所述表项关联的单个区是否可用。当位58为0时，通过使用区表表项继续地址转换。当所述位为1时，表项不能用于转换。

[0065] 表类型位(TT)412、442、472：区第一表表项、区第二表表项和区第三表表项的位60和61标识包含表项的表的级别，如下所示：位60和61标识正确的表级别，考虑了是转换中使用的地址空间控制元素的表指定类型和目前已用的表级别数；否则，将识别到转换指定异常。作为一个实例，下表示出了表类型位：

[0066]

位 60 和 61	区表级别
11	第一
10	第二
01	第三

[0067] 区表表项的表类型位

[0068] 区第二表长度414、区第三表长度444和段表长度474(TL)：区第一表表项包括区第二表长度。区第二表表项包括区第三表长度。区第三表表项包括段表长度。以下说明适用于全部三种长度。表项的位62和63以4,096字节为单位指定下一低级表的长度，从而使表的长度以512个表项的倍数变化。以4,096字节为单位表示的下一低级表的长度比TL值大1。长度字段的内容与偏移字段(位56和57)一起用于确定要通过下一低级表转换的虚拟地址部分(RSX、RTX或SX)是否指定表中实际存在的表项。

[0069] 将保留区表表项的所有其他位位置以供可能的未来扩展并且这些位位置应包含0；否则，程序在将来可能不能兼容地运行。当应用增强的DAT时，即使表表项无效，区表表项的保留位位置也应包含0。

[0070] 段表表项

[0071] 当未应用增强的DAT，或当应用增强的DAT并且段表表项(STE)格式控制(段表表项的位53)为0时，从段表取回的表项具有图5A中所示的格式(例如，格式0)。当应用增强的DAT并且STE格式控制为1时，从段表取回的表项具有例如图5B中所示的格式(例如，格式1)。

[0072] 作为一个实例，格式0段表表项500(图5A)包括以下字段：

[0073] 页表起始地址502：当未应用增强的DAT，或当应用增强的DAT但是STE格式控制(段

表表项的位53)为0时，位0-52(右侧附加11个0)形成指定页表的开头的64位地址。无法预测此地址是实地址还是绝对地址。

[0074] STE格式控制(FC)506：当应用增强的DAT时，位53为段表表项的格式控制，如下所示：

[0075] ●当FC位为0时，表项的位0-52形成页表起始地址，并且保留位55。

[0076] ●当FC位为1时，表项的位0-43形成段帧绝对地址，位47为ACCF有效性控制，位48-51为访问控制位，位52为取回保护位，以及位55为更改记录忽略(override)。当未应用增强的DAT时，将忽略位53。

[0077] DAT保护位(P)508：位54，当为1时，指示将DAT保护应用于整个段。

[0078] ●当未应用增强的DAT时，位54被视为与转换中使用的页表表项中的DAT保护位进行“或”。

[0079] ●当应用增强的DAT时，转换中使用的任何及所有区表表项中的DAT保护位被视为与段表表项中的DAT保护位进行“或”；当STE格式控制为0时，STE中的DAT保护位还被视为与页表表项中的DAT保护位进行“或”。

[0080] 段无效位(I)510：位58控制与段表表项关联的段是否可用。

[0081] ●当所述位为0时，通过使用段表表项继续地址转换。

[0082] ●当所述位为1时，段表表项不能用于转换。

[0083] 公共段位(C)512：位59控制对段表表项的转换后备缓冲区(TLB)副本的使用。当未应用增强的DAT或当应用增强的DAT但是格式控制为0时，位59还控制对段表表项指定的页表的TLB副本的使用。

[0084] ●0标识专用段；在此情况下，段表表项及其指定的任何页表只能与指定段表表项所在的段表的段表起始地址联合使用。

[0085] ●1标识公共段；在此情况下，即使已指定其他段表，段表表项及其指定的任何页表也可以继续用于转换与段索引对应的地址。

[0086] 但是，如果转换中使用的地址空间控制元素中的专用空间控制(位55)为1或者如果该地址空间控制元素为实空间指定时，则不可以使用公共段的段表表项和任何页表的TLB副本。如果所使用的地址空间控制元素中的专用空间控制为1，并且在转换期间从存储装置取回段表表项，则公共段位将为0；否则，将识别到转换指定异常。

[0087] 表类型位(TT)514：段表表项的位60和61为二进制00以标识包含表项的表的级别。区表表项或段表表项中的位60和61的可能值的含义如下：

[0088]

位60和61	表级别
11	区第一
10	区第二
01	区第三
00	段

[0089] 表类型位60、61

[0090] 位60和61标识正确的表级别，其中考虑了是转换中使用的地址空间控制元素的表指定的类型和目前已用的表级别数；否则，将识别到转换指定异常。

[0091] 将保留段表表项的所有其他位位置以便将来可能进行扩展并且这些位位置应包含0；否则，程序在将来可能不能兼容地运行。当应用增强的DAT时，即使表项无效，段表表项的保留位位置也应包含0。

[0092] 作为一个实例，格式1段表表项550(图5B)包括以下字段：

[0093] 段帧绝对地址(SFAA)552：当应用增强的DAT并且STE格式控制为1时，表项的位0-43(右侧附加20个0)形成段的64位绝对地址。

[0094] ACCF有效性控制(AV)556：当应用增强的DAT并且STE格式控制为1时，位47为访问控制位和取回保护位(ACCF)有效性控制。当AV控制为0时，将忽略段表表项的位48-52。当AV控制为1时，按照下面所述使用位48-52。

[0095] 访问控制位(ACC)558：当应用增强的DAT，STE格式控制为1，并且AV控制为1时，段表表项的位48-51包括可用于应用于地址的任何键控访问检查的访问控制位。

[0096] 取回保护位(F)560：当应用增强的DAT，STE格式控制为1，并且AV控制为1时，段表表项的位52包括可用于应用于地址的任何键控访问检查的取回保护位。

[0097] STE格式控制(FC)562：当应用增强的DAT时，位53为段表表项的格式控制，如下所示：

[0098] ●当FC位为0时，表项的位0-52形成页表起始地址，并且保留位55。

[0099] ●当FC位为1时，表项的位0-43形成段帧绝对地址，位47为访问控制位和取回保护位ACCF有效性控制，位48-51为访问控制位，位52为取回保护位，以及位55为更改记录忽略。当未应用增强的DAT时，将忽略位53。

[0100] DAT保护位(P)564：位54，当为1时，指示将DAT保护应用于整个段。

[0101] ●当未应用增强的DAT时，位54被视为与转换中使用的页表表项中的DAT保护位进行“或”。

[0102] ●当应用增强的DAT时，转换中使用的任何及所有区表表项中的DAT保护位被视为与段表表项中的DAT保护位进行“或”；当STE格式控制为0时，STE中的DAT保护位还被视为与页表表项中的DAT保护位进行“或”。

[0103] 更改记录忽略(CO)556：当应用增强的DAT并且STE格式控制为1时，段表表项的位55为段的更改记录忽略。当未应用增强的DAT，或当应用增强的DAT但是STE格式控制为0时，将忽略段表表项的位55。

[0104] 段无效位(I)568：位58控制与段表表项关联的段是否可用。

[0105] ●当所述位为0时，通过使用段表表项继续地址转换。

[0106] ●当所述位为1时，段表表项不能用于转换。

[0107] 公共段位(C)570：位59控制对段表表项的转换后备缓冲区(TLB)副本的使用。当未应用增强的DAT或者当应用增强的DAT但是格式控制为0时，位59还控制对段表表项指定的页表的TLB副本的使用。

[0108] ●0标识专用段；在此情况下，段表表项及其指定的任何页表只能与指定段表表项所在的段表的段表起始地址联合使用。

[0109] ●1标识公共段；在此情况下，即使已指定其他段表，段表表项及其指定的任何页

表也可以继续用于转换与段索引对应的地址。

[0110] 但是，如果转换中使用的地址空间控制元素中的专用空间控制(位55)为1或者如果该地址空间控制元素为实空间指定时，则不可以使用公共段的段表表项和任何页表的TLB副本。如果所使用的地址空间控制元素中的专用空间控制为1，并且在转换期间从存储装置取回段表表项，则公共段位将为0；否则，将识别到转换指定异常。

[0111] 表类型位(TT)572：段表表项的位60和61为二进制00以标识包含表项的表的级别。区表表项或段表表项中的位60和61的可能值的含义如下：

[0112]

位60和61	表级别
11	区第一
10	区第二
01	区第三
00	段

[0113] 表类型位60、61

[0114] 位60和61标识正确的表级别，其中考虑了是转换中使用的地址空间控制元素的表指定的类型和目前已用的表级别数；否则，将识别到转换指定异常。

[0115] 将保留段表表项的所有其他位位置以便将来可能进行扩展并且这些位位置应包含0；否则，程序在将来可能不能兼容地运行。当应用增强的DAT时，即使表项无效，段表表项的保留位位置也应包含0。

[0116] 页表表项

[0117] 客户端块(例如，绝对存储器中具有关联的单个存储键和协作存储管理(CMM)状态的区域(例如，4KB))的状态信息在例如描述客户端存储器的主机页表(PT)和页状态表(PGST)中进行维护。这些表分别例如包括一个或多个页表表项(PTE)和一个或多个页状态表表项(PGSTE)，下面将进一步详细说明。

[0118] 参考图6描述了页表表项600的一个实例。在一个实施例中，页表表项中的字段按如下方式进行分配：

[0119] 页帧实地址(PFRA)602：位0-51提供实(在该情况中为主机实)存储地址的最左侧的位。当这些位的右侧与虚拟地址的12位字节索引字段连接时，将得到64位实地址。

[0120] 页无效位(I)604：位53控制与页表表项关联的页是否可用。当所述位为0时，通过使用页表表项继续地址转换。此外，对于主机和客户端之间的协作存储管理(CMM)，主机状态为r(常驻；即，客户端块位于主机帧内)。当所述位为1时，页表表项不能用于转换，并且CMM主机状态为p(保存；即，客户端块不在主机帧内，但是已被主机保存在某个辅助存储装置中)或z(逻辑为0；即，客户端块不在主机帧内而且客户端块的内容已知为0)，上述状态由PGSTE.Z确定。

[0121] DAT保护位(P)606：位54控制是否可以在页内进行存储访问。该保护机制作为键控保护和低地址保护机制的补充。所述位对取回访问没有影响。如果所述位为0，则允许在页内进行存储，但是需要受到以下附加约束：

[0122] ●转换中使用的段表表项中的DAT保护位为0。

[0123] ●当应用增强的DAT时，转换中使用的所有区表表项中的DAT保护位为0。

[0124] 如果DAT保护位为1，则不允许存储。当不存在更高优先级的异常条件时，在DAT保护位为1时尝试存储会导致识别到保护异常。当判定是否将DAT保护应用于页时，段表表项中的DAT保护位被视为与位54进行“或”。当应用增强的DAT并且判定是否应用DAT保护位时，转换中使用的任何区表表项中的DAT保护位也被视为与位54进行“或”。

[0125] 诸如键控保护、低地址保护和访问列表控制保护之类的其他保护机制可以独立于DAT保护而应用并且也可禁止访问。

[0126] 更改记录忽略(CO)608：当未应用增强的DAT时，页表表项的位55包含0；否则，将在执行使用该表项进行地址转换的指令时识别到转换指定异常。当应用增强的DAT并且STE格式控制为0时，页表表项的位55为该页的更改记录忽略。

[0127] 除了上述内容之外，在一个实例中，表项的位位置52包含0；否则，将在执行使用该表项进行地址转换的指令时识别到转换指定异常。不分配位位置56-63并忽略它们。

[0128] 参考图7描述了页状态表表项的一个实例。页状态表表项700例如包括以下内容：

[0129] (a)Acc 702：访问控制键；

[0130] (b)FP 704：取回保护指示符；

[0131] (c)页控制互锁(PCL)706：这是用于串行化对页表表项(PTE)和相应PGSTE(除了被标记为保留的PGSTE状态区域和PGSTE位以外)的更新的互锁控制。

[0132] (d)HR 708：主机引用备份指示符；

[0133] (e)HC 710：主机更改备份指示符；

[0134] (f)GR 712：客户端引用备份指示符；

[0135] (g)GC 714：客户端更改备份指示符；

[0136] (h)状态716：旨在供主机程序使用。

[0137] (i)页内容逻辑0指示符(Z)718：当相应的PTE页无效(PTE.I)位为1时，该位才有意义。

[0138] 当Z为1时，由该PGSTE和相应的PTE描述的页内容被视为0。不必保留该页的任何先前内容并且可以被包含0的页所替换。

[0139] 当Z为0时，由PGSTE和相应的PTE描述的页内容不被视为0。主机保存该页的内容。

[0140] 当Z位为1并且相应的PTE.I位为1时，CMM主机状态为z(逻辑0)。这表示主机或主机页管理帮助工具的功能可以替换页内容。

[0141] 当Z位为1并且相应的PTE.I位为1，以及页内容被替换时，应通过将该页与已被设为0的帧相关联来替换该页。

[0142] 当Z位为0并且PTE无效位为1时，CMM主机状态为p(保存)。

[0143] (j)页类别(PC)720：当为0时，由PGSTE和相应的PTE描述的页为0类页并且0类页的增量钉扎(pin)页计数数组(DPPCA)用于对页的钉扎和解除钉扎操作进行计数。当为1时，由PGSTE和相应的PTE描述的页为1类页并且1类页的DPPCA用于对页的钉扎和解除钉扎操作进行计数。

[0144] (k)钉扎计数溢出(PCO)722：当为1时，钉扎计数字段处于溢出状态。在此情况下，总钉扎计数由主机保存在机器不访问的另一数据结构中。当为0时，钉扎计数字段不处于溢

出状态。

[0145] (l)已处理帧描述符列表中的帧描述符(FPL)724:当为1时,由PGSTE和相应的PTE描述的页的帧描述符位于已处理帧描述符列表中。所述帧描述符标识由该页的HPMA解析主机页功能使用的主机帧。

[0146] (m)请求的页内容替换(PCR)726:当为1时,在针对由PGSTE和相应的PTE表示的页调用HPMA解析主机页功能时请求页内容替换。

[0147] (n)使用状态(US)728:对于主机和客户端之间的协作存储管理,该字段指示客户端状态是否为S(稳定;即,稳定块的内容保持与客户端所设的内容相同);U(未使用;即,未使用块的内容对于客户端而言无意义);V(易失;即,易失性块的内容对于客户端而言有意义,但是主机可随时丢弃块内容和回收后备帧);或P(潜在易失;即,潜在易失性块的内容对于客户端而言有意义,但是根据客户端更改历史,主机可以丢弃块内容,或应保存块内容)。

[0148] (o)状态730:旨在供主程序使用。

[0149] (p)钉扎计数732:无符号二进制整数计数,用于指示由PGSTE和相应的PTE表示的主机虚拟页的内容是否被钉扎在PTE的页帧实地址字段所指定的实主机帧中。当该字段的值大于0或者页计数溢出(PCO)位为1时,相应的页视为被钉扎。当该字段的值为0并且PCO位为0时,不将相应的页视为被钉扎。

[0150] 通过主机或CPU钉扎页时,该字段应递增1。通过主机或CPU解除钉扎页时,该字段应递减1。

[0151] 当钉扎计数字段的值大于0或者PCO位为1时,相应的PTE.I(页无效)位为0。否则,可能出现无法预测的结果。

[0152] 在页被钉扎时,主机程序不应更改PTE或段表表项(STE)中的PTE页帧实地址(PFRA)字段的内容、PTE页无效(I)位的设置或页保护(P)位的设置。否则可能出现无法预测的结果。

[0153] 在标题为“z/Architecture Principles Of Operation(z/Architecture操作原理,IBM出版号:SA22-7832-05,2007年4月)”的IBM出版物中描述了有关页表表项和页表以及段表表项的其他详细信息。此外,在Blandy等人的标题为“Facilitating Management of Storage of a Pageable Mode Virtual Environment Absent Intervention of a Host of the Environment(促进无环境主机干预下的可分页模式虚拟环境的存储管理,申请日:2004年5月27日)”的美国申请第10/854,990和Adlung等人的标题为“Facilitating Processing Within Computing Environment Supporting Pageable Guests(促进支持可分页客户端的计算环境内的处理,公开日:2007年1月18日)”的美国专利申请公开US 2007/0016904A1中描述了有关PGSTE的其他详细信息。

[0154] 在一个实施例中,针对每个页表存在一个页状态表,页状态表的大小与页表的大小相同,页状态表表项的大小等于页表表项的大小,并且页状态表(在主机实存储器中)位于距页表固定的位移处。因此,每个页表表项和页状态表表项之间存在一一对应关系。给定主机的页虚拟地址,机器和主机均可轻松找到与客户端块的页表表项对应的页状态表表项。

[0155] 写时复制

[0156] 有时需要复制部分存储器。该复制可以由用户发起,也可以由操作系统发起。传统

的系统通常使用懒惰复制技术进行快速复制，其中为要复制的存储装置指定只读状态，但是实际复制推迟到以后进行。如果尝试写入原存储或副本，则此时复制存储器并且为原存储和副本指定读写输入/输出(I/O)状态。通过这种方式，看上去可以立即生成副本，但是实际复制推迟到以后进行(可能的最晚时间)。如果未执行任何写入，则不进行复制。为此，此方法称为写时复制(COW)或虚拟复制。

[0157] 一般而言，写时复制操作是计算密集的，因为单次写入会导致两次写入操作。也就是说，首先将现有的数据块从旧物理块复制到新物理块，然后对新物理块执行实际的更新/写入操作。

[0158] 指令执行

[0159] 在z/Architecture中，例如，通过以下五种方式之一结束指令执行：完成、废除、抑制、终止和部分完成，下面将逐一进行介绍。

[0160] 指令执行的完成提供了在定义指令时要求的结果。在完成指令执行之后发生中断时，旧PSW中的指令地址指定下一顺序指令。

[0161] 指令执行的废除与下面所述的抑制的效果相同，只是在废除指令执行之后发生中断时，旧PSW中的指令地址指定其执行被废除的指令(或相应地为Execute指令)，而不指定下一顺序指令。

[0162] 指令执行的抑制使得指令的执行效果相当于被指定了“无任何操作”。包括条件码在内的任何结果字段的内容都不变。在抑制之后发生中断时，旧PSW中的指令地址指定下一顺序指令。

[0163] 指令执行的终止导致将由指令更改的任何字段的内容变得无法预测。所述操作可能替换指定结果字段的所有内容、部分内容，也可能不替换的任何内容并且可以在指令需要更改条件码时做出此更改。除非中断由机器检查条件造成，否则PSW中指令地址的有效性、中断码和指令长度码不受影响，而且状态或机器操作不受其他任何方面的影响。在终止之后发生中断时，旧PSW中的指令地址指定下一顺序指令。

[0164] 指令执行的部分完成针对可中断指令发生。

[0165] 尽管当发生抑制或废除时，指令的执行被视为不执行任何操作，但是作为隐式跟踪与某些指令关联的操作的结果，可以执行存储操作。

[0166] 保护

[0167] 提供了四种保护工具来保护主存储装置的内容不被包含错误的程序或未授权的程序毁坏或误用：键控保护、访问列表控制保护、页保护和低地址保护。所述保护工具被独立地应用；在所有工具都不禁止访问的情况下，允许对主存储装置进行访问。下面将介绍每种保护工具。

[0168] 键控保护：键控保护提供了防止不正确的存储或同时防止不正确的存储和取回，但是不单独防止不正确的取回。

[0169] 对存储访问实施键控保护时，当存储键匹配与存储访问请求关联的访问键时，允许执行存储；当上述两个键匹配或当存储键的取回保护位为0时，允许执行取回。当存储键的四个访问控制位等于访问键，或者当访问键为0时，所述键被称为匹配。下表总结了保护操作：

[0170]

条件		是否允许对存储装置的访问	
存储键的取回保护位	键关系	取回	存储
0	匹配	是	是
0	不匹配	是	否
1	匹配	是	是
1	不匹配	否	否

说明

匹配：存储键的四个访问控制位等于访问键，或访问键为0。

是 允许访问。

否 不允许访问。取回时，信息对于程序而言不可用；存储时，存储位置的内容不变。

[0171] 当对存储装置的访问由CPU发起并且应用了键控保护时，PSW键为访问键，除了针对诸如Move To Secondary和Move With Destination Key之类的特定指令的第一操作数；针对诸如Move To Primary、Move With Key和Move With Source Key之类的指令的第二操作数；以及针对例如Move Page指令的第一操作数或第二操作数在通用寄存器中指定访问键以外。PSW键占据例如当前PSW的位位置8-11。当访问存储装置是为了执行通道程序时，与该通道程序关联的子通道键为访问键。在操作请求块(ORB)中指定通道程序的子通道键。当为了监视通道子系统而访问测量块时，测量块键为访问键。测量块键例如由Set Channel Monitor指令指定。

[0172] 当由于键控保护而禁止CPU访问时，将终止指令执行，并且发生保护异常程序中断。但是，可能抑制操作单元或指令执行。当禁止通道程序访问时，将终止启动功能，并且在关联的中断-响应块(IRB)中指示保护检查条件。当禁止测量块访问时，将指示I/O测量块保护检查条件。当由于键控保护而禁止存储访问时，受保护位置的内容保持不变。当禁止取回访问时，不会将受保护信息加载到寄存器、移动到另一存储位置，或提供给I/O设备。对于禁止的指令取回，将抑制指令，并且指示任意指令长度码。

[0173] 键控保护与CPU是否出现问题或处于监控状态无关，并且在一些例外的情况下，不依赖正在执行的CPU指令或通道命令字的类型。除非另外指出，否则对程序显式指定的存储位置和CPU用于存储或取回信息的存储位置的访问将受键控保护限制。当存储保护忽略控制为1并且存储键的四个访问控制位的值例如为9时，不应用键控保护。当取回保护忽略控制为1时，可以应用也可以不应用针对取回的键控保护，具体取决于有效地址和专用空间控制。存储保护忽略控制和取回保护忽略控制不影响通道子系统所做的存储引用。对例如Test Block指令的第二操作数的访问不受键控保护限制。通道子系统为了访问I/O测量块而做出的存储访问，或通道程序为了取回CCW、IDAW(间接数据地址字)或MIDAW(修改后的间接数据地址字)或为了访问在执行CCW期间指定的数据区域而做出的存储访问受键控保护限制。但是，如果预取CCW、IDAW、MIDAW或输出数据，则直到CCW、IDAW或MIDAW要取得控制或直到要写入数据时才指示保护检查。键控保护不应用于针对以下任一序列隐式地做出的访问：

[0174] ●中断

[0175] ●CPU注销

[0176] ●取回表表项以进行存取寄存器转换、动态地址转换、PCnumber转换、ASN转换或ASN授权

[0177] ●跟踪

[0178] ●存储状态功能

[0179] ●当Test Pending Interruption(测试暂挂中断)操作数地址为0时在实位置184-191中存储

[0180] ●初始程序加载。

[0181] 类似地,保护不会应用于通过用于修改或显示信息的操作员工具启动的访问。但是,当程序显式指定这些位置时,它们将受保护限制。

[0182] 受到键控保护限制的访问可能受存储保护忽略控制的影响。例如,控制寄存器0的位39为存储保护忽略控制。当该位为1时,存储保护忽略是活动的。当该位为0时,存储保护忽略是不活动的。当存储保护忽略活动时,针对关联存储键值例如为9的存储位置忽略键控存储保护。当存储保护忽略不活动时,不针对为9的存储键值执行任何特殊操作。存储保护忽略应用于指令取回以及其操作数地址为逻辑地址、虚拟地址或实地址的指令的取回访问和存储访问。它不应用于为了执行通道程序或为了监视通道子系统而进行的访问。存储保护忽略对不受键控保护限制的访问没有影响。

[0183] 当可能错误的应用程序与可靠的子系统一起执行时,可以使用存储保护忽略来提高可靠性,前提是该应用程序只需访问该子系统所访问的存储装置的一部分。下面介绍执行此操作的技术。由该应用程序访问的存储装置被赋予存储键9。仅由该子系统访问的存储装置被赋予其他某个非零存储键(例如,键8)。该应用通过PSW键9执行。该子系统通过PSW键8执行(在此实例中)。结果是,该子系统可以访问键8和键9存储装置两者,而该应用程序只能访问键9存储装置。

[0184] 存储保护忽略影响CPU对存储装置的访问,同时也影响Test Protection设置的结果。但是,这些在问题状态中测试PSW键掩码以判定是否可以使用特定键值的指令不受存储保护忽略是否活动的影响。这些指令包括Move With Key和Set PSW Key From Address等。为了允许这些指令在问题状态中使用为9的访问键,PSW键掩码的位9应为1。

[0185] 除了存储保护忽略控制之外,受到键控保护限制的访问还可能受取回保护忽略控制的影响。作为一个实例,控制寄存器0的位38为取回保护忽略控制。当该位为1时,针对例如有效地址0-2047处的位置忽略取回保护。有效地址是通过动态地址转换或添加前缀进行任何转换之前存在的地址。但是,如果有效地址需要进行动态地址转换并且转换中使用的地址空间控制元素中的专用空间控制(例如,位55)为1,则不忽略取回保护。取回保护忽略应用于指令取回以及其操作数地址为逻辑地址、虚拟地址或实地址的指令的取回访问。它不应用于为了执行通道程序或为了监视通道子系统而进行的取回访问。当该位被设为0时,对有效地址0-2047处的位置的取回保护由与这些位置关联的存储键的取回保护位的状态确定。取回保护忽略对不受键控保护限制的访问没有任何影响。

[0186] 取回保护忽略控制允许对例如地址2048-4095处的位置进行取回保护,同时不对地址0-2047处的位置进行任何取回保护。

[0187] 访问列表控制保护

[0188] 在存取寄存器模式中，访问列表表项的位6(仅取回位)控制允许对访问列表表项所指定的地址空间进行哪些类型的操作数引用。当在引用的存取寄存器转换部分中使用表项并且位6为0时，同时允许取回型和存储型引用；当位6为1时，允许取回型引用，并且存储尝试导致识别到保护异常以及指令执行被抑制。仅取回位包含在ALB访问列表表项中。更改主存储装置内的访问列表表项中的仅取回位不一定立即(如果有)影响是否识别到保护异常。但是，例如，执行清除ALB的Purge ALB或Compare And Swap And Purge指令之后，此位更改会立即产生影响。下面将介绍的Test Protection指令考虑了在CPU处于存取寄存器模式下的访问列表控制保护。访问列表控制保护的违反将导致设置例如条件码1，只是在满足例如条件码2或3的条件下，其不会阻止设置这些条件码。

[0189] 访问列表控制保护的违反将导致出现抑制操作。任何其他保护类型的违反都可导致终止操作。

[0190] 页保护

[0191] 页保护工具通过使用每个页表表项和段表表项中的页保护位来控制对虚拟存储装置的访问。其提供了对不正确存储的保护。页保护位(页表表项的位54)控制是否允许存储到相应的4KB页。当该位为0时，同时允许取回和存储；当该位为1时，仅允许取回。当尝试存储到受保护页时，所述页的内容保持不变，将抑制操作单元或指令执行，并且发生针对保护的程序中断。页保护位(段表表项的位54)被视为与段表表项所指定的页表中的每个表项的页保护位位置进行“或”。因此，当段表表项页保护位为1时，其效果就像页保护位在指定的页表中的每个表项内为1。页保护应用于使用虚拟地址的存储型引用。

[0192] 低地址保护

[0193] 低地址保护工具提供了对中断处理期间毁坏CPU所使用的主存储信息的保护。通过禁止指令使用位于例如范围0到511和4096到4607中的有效地址(每个第一和第二4KB有效地址块的第一512字节)进行存储来实现此目的。在通过动态地址转换或添加前缀进行地址的地址转换(如果有)之前应用所述范围准则。但是，如果有效地址需要进行动态地址转换并且转换中使用的地址空间控制元素中的专用空间控制(例如，位55)为1，则不应用范围准则，从而也不应用低地址保护。如果要使用的地址空间控制元素由于另一类型的异常而不可用，则不会应用低地址保护。低地址保护受控制寄存器0的位35(低地址保护控制位)控制。当该位为0时，低地址保护关闭；当该位为1时，低地址保护开启。

[0194] 如果由于低地址保护而禁止访问，则受保护位置的内容保持不变，指令执行终止，并且发生针对保护异常的程序中断。但是，可抑制操作单元或指令执行。程序使用位于例如范围0到511和4096到4607中的有效地址尝试进行存储受低地址保护限制。低地址保护应用于其操作数地址为逻辑地址、虚拟地址或实地址的指令的存储访问。低地址保护还应用于跟踪表。低地址保护不应用于CPU或通道子系统针对诸如中断、CPU注销、通过Test Pending Interruption将I/O中断码存储到例如实位置184-191中以及初始程序加载和存储状态功能之类的序列做出的访问，也不在I/O数据传输期间应用于数据存储。但是，程序在这些位置中的任一位置处进行的显式存储受低地址保护限制。

[0195] 低地址保护和键控保护应用于同一存储访问，只是：低地址保护不应用于由通道子系统执行的存储，而键控保护应用于此类存储；以及键控保护不应用于跟踪、Test Block

的第二操作数或专门针对链接堆栈操作的指令，而低地址保护应用于上述各项。

[0196] 由于取回保护忽略和低地址保护不应用于专用空间控制在地址空间控制元素中为1的地址空间，因此，可像所述地址空间中的其他位置一样使用该空间中的位置0-2047和4096-4607。

[0197] 保护时抑制

[0198] 有些指令定义指定如果由于任何类型的保护而识别到保护异常时，将抑制操作。如果未进行该指定，则当由于访问列表控制保护或DAT保护(又称为页保护)识别到保护异常时，将抑制指令执行，如果由于低地址保护或键控保护而识别到保护异常，则可以抑制也可以终止指令执行。

[0199] 保护时抑制功能允许控制程序定位在导致保护异常的虚拟地址转换中使用的段表表项、页表表项以及区表表项(当应用增强的DAT时)，以便确定所述异常是否源于DAT保护。这用于例如实现POSIX fork功能(或写时复制功能)。所述功能还允许控制程序在地址不是虚拟地址或异常源于访问列表控制保护的情况下避免定位段表表项和页表表项。当应用增强的DAT时，如果地址不是虚拟地址或异常源于访问列表控制保护，则控制程序还可以避免定位区表表项。

[0200] 在源于保护异常的程序中断期间，1或0存储在例如实位置168-175的位位置61。例如，在z/Architecture中，将1存储到位位置61中指示：

[0201] ●抑制其间识别到异常的操作单元或指令执行。

[0202] ●如果开启动态地址转换(DAT)，如程序旧PSW中的DAT模式位所指示的，则导致异常的有效地址是要通过DAT转换的地址。(有效地址是通过DAT或添加前缀进行任何转换之前存在的地址)。如果开启DAT，则位61设为0，但是有效地址不通过DAT进行转换，因为它是实地址。如果关闭DAT，则保护异常不可能由DAT保护或访问列表控制保护造成。

[0203] ●实位置168-175的位位置0-51包含导致异常的有效地址的位0-51。如果开启DAT(指示要通过DAT转换有效地址)，则实位置168-175和实位置160的位位置62和63包含的信息与由页转换异常导致的程序中断期间存储的信息相同-此信息标识包含受保护地址的地址空间。此外，如果保护异常不是源于访问列表控制保护，则实位置168-175的位60为0，如果所述异常是源于访问列表控制保护，则实位置168-175的位60为1。位位置60中的1指示异常不是源于DAT保护。如果关闭DAT，则无法预测实位置168-175的位位置60、62和63的内容以及实位置160的内容。实位置168-175的位位置52-59的内容无法预测。

[0204] 位61为0指示操作被抑制或终止并且实位置168-175的其余位位置的内容以及实位置160的内容无法预测。

[0205] 如果保护异常是源于访问列表控制保护或DAT保护，则将位61设为1。如果保护异常是源于低地址保护或键控保护，则可以将位61设为1。

[0206] 如果由于访问列表控制保护或DAT保护而存在保护异常，并且还由于低地址保护或键控保护而存在保护异常，则无法预测位61被设为0还是1。

[0207] 保护时抑制功能对于执行POSIX fork功能非常有用，fork功能可以导致创建重复的地址空间。下面的说明涉及何时不应用增强的DAT，或何时应用增强的DAT，但是段表表项中的格式控制(FC)位为0。当发生分叉(forking)时，控制程序使不同地址空间的同一页映射到实存储装置的单个页帧，前提是未尝试在该页内进行存储。然后，当在特定地址空间中

尝试存储时，控制程序为该地址空间中的页分配唯一的页帧并将该页的内容复制到新的页帧。后一个操作有时称为写时复制功能。控制程序在页的页表表项中将DAT保护位设为1以便检测在该页中进行存储的尝试。控制程序可以初始地在段表表项中将DAT保护位设为1以检测在指定段中的任意位置进行存储的尝试。当应用增强的DAT并且段表表项中的格式控制(FC)位为1时，可以使用类似的技术映射绝对存储装置的单个段帧。

[0208] 当开启DAT时，实位置168-175中的位61为1指示导致保护异常的地址为虚拟地址。与低地址保护或键控保护相比，该指示允许执行编程形式的存取寄存器转换和动态地址转换，以确定异常是否源于DAT保护。

[0209] 下表总结了保护时抑制的结果：

[0210]

异常条件				提供的字段		
LA 保护或键控保护	DAT	ALC 保护或页保护	有效地址	位 61	如果位 61 为 1	
					位 62、63 和位置 160	位 60
否	开启	是	Log.	1	P	1A
是	开启	是	Log.	U1	P	1A
是	关闭	否	Log.	U2	U3	U3
是	关闭	否	实	U2	U3	U3
是	开启	否	Log.	U2	P	0
是	开启	否	实	OR	-	-

说明：

- 无关紧要或不适用。
OR 0，因为有效地址为实地址
1A 如果位 61 因访问列表控制保护被设为 1，则为 1；否则为 0。
ALC 访问列表控制。
LA 低地址。
Log. 逻辑
P 可预测

[0211]

U1 由于可识别到低地址保护或键控保护而非访问列表控制保护或页保护而无法预测。
U2 由于只需针对访问列表控制保护或页保护将位 61 设为 1，因此无法预测。
U3 由于关闭了 DAT 而无法预测。

[0212] 增强的保护时抑制

[0213] 当安装了增强的保护时抑制功能时，对于在保护异常期间可出现的操作存在以下附加约束。这些约束的优先级高于原始保护时抑制功能中定义的任何约束。

[0214] 在由保护异常导致程序中断期间，实位置168-175的位61指示识别到的保护异常类型。如果保护异常是源于访问列表控制保护或DAT保护，则位61被设为1。如果保护异常是源于低地址保护或键控保护，则位61被设为0。

[0215] 如果保护异常条件的出现源于访问列表控制保护或DAT保护，但同时还源于低地址保护或键控保护，则无法预测将识别到哪个异常以及是将位61设为0还是1。但是，尽管无法预测将识别到哪个异常，但是识别到的异常会产生一致的行为，如下表所示：

[0216]

异常类型	DAT	位 61	位 60	位 62、63 和位置 160	位 0-51
LAP	任意	0	-	-	-
KCP	任意	0	-	-	-
ALCP	开启	1	1	AS	A
DATP	开启	1	0	AS	A
说明:					
A	导致异常的有效地址的位 0-51。				
ALCP	访问列表控制保护。				

[0217]

AS	标识包含导致异常的有效地址的地址空间
DATP	DAT 保护。
KCP	键控保护。
LAP	低地址保护。
-	未定义。

[0218] 针对本机架构以及在解释执行模式中提供了SOP和ESOP工具。

[0219] 在z/Architecture中，解释执行模式中的保护时抑制用于在异常控制(例如，ECA.18)为1时处理保护异常；解释执行模式中的增强的保护时抑制用于在异常控制(例如，ECA.18)为1并且已安装增强的保护时抑制工具之后处理异常。在该实例中，ECA.18是保护拦截控制，一般而言，如果ECA中的位为0，则关联的条件导致拦截，如果ECA中的位为1，则由主机尝试关联功能的解释执行。

[0220] 此外，在一个实例中，执行控制(例如，ECB.6)是主机保护中断控制。一般而言，如果ECB中的位为0，则关联的条件导致拦截，如果ECB中的位为1，则尝试关联功能的解释执行。解释执行模式中的增强的保护时抑制用于在ECB.6为1并且已安装增强的保护时抑制工具之后处理主机页保护异常。

[0221] 根据ECA.18、ECB.6和IC.2(拦截控制)的值，保护异常可能导致程序中断拦截。否则，程序中断导致使用例如拦截码08进行拦截，前提是相应的拦截控制(位1或位2)针对所述程序中断的类型被设为1。在一个实例中，通过拦截码08强制执行拦截。针对由例如以下

异常导致的客户端程序中断存储拦截码08：寻址、指定和特殊操作。

[0222] 客户端寻址异常可能源于无效的客户端或主机地址。

[0223] 客户端保护异常可能源于客户端保护条件、主机页保护条件或在给定模式(例如，MCDS-多控制数据空间)中，源于主机访问列表控制保护条件。

[0224] 中断参数被置于状态描述中。

[0225] 在各种解释执行模式中，客户端对客户端存储装置进行访问受在本机使用实机器存储键定义的键控存储保护的限制。客户端访问还受客户端低地址保护、客户端页保护、客户端取回保护忽略以及客户端访问列表控制保护的限制。主机低地址保护不应用于对客户端存储的客户端引用。

[0226] 此外，主机页保护应用于可分页存储模式客户端引用，如同MCDS模式中的主机访问列表控制保护一样。

[0227] 一般而言，不允许的存储会导致识别到保护异常，所述保护异常将导致客户端或主机程序中断。但是，在进入解释执行模式时，将检查对客户端前缀区域的第一4KB块的存储访问，并且在此测试期间遇到的任何访问异常都会导致有效性拦截(例如，VIR码0037(十六进制))。接下来，在解释执行期间，访问客户端前缀区域的第一4KB块时遇到的访问异常条件可以正常呈现，或者可能导致有效性拦截。如果为了执行客户端中断而对客户端前缀区域进行的引用所遇到的访问异常条件导致有效性拦截，则可能丢失以下信息：

[0228] ●要存储在客户端前缀区域中的中断信息。(但是，中断之前的客户端PSW存储在状态描述中)。

[0229] ●中断请求本身，如果适用。

[0230] 在客户端Test Protection指令的解释执行期间设置条件码时，将考虑主机页保护和访问列表控制保护机制，这在下面介绍。

[0231] 有关解释执行模式中SOP和ESOP的其他详细信息将在下面介绍。

[0232] 解释执行模式中的保护时抑制

[0233] 针对可分页模式DAT关闭客户端，可以例如由z/VM通过主机访问列表控制保护和主机页保护使用保护时抑制功能。当DAT模式位(例如，程序旧PSW的位5)为0时，客户端的DAT关闭；当DAT模式位为1时，客户端的DAT开启。主机访问列表控制保护应用于MCDS客户端(它是DAT关闭客户端)。

[0234] 在解释执行模式中，出于此保护时抑制定义的目的，实位置168-175中存储的有效地址为客户端有效地址，即使当保护异常源于主机访问列表控制保护或主机页保护时也是如此。

[0235] 当保护异常由主机访问列表控制保护或主机页保护造成并且客户端为DAT关闭客户端时，即使当客户端有效地址是被定义为实(与逻辑相对)地址的地址，位61也被设为1，并且存储客户端有效地址的位0-51。在此情况下，实位置168-175的位62和63以及实位置160的内容正确地指示在客户端为MCDS模式客户端的情况下应用异常的主机地址空间，或者如果客户端为非MCDS模式DAT关闭客户端，则这些字段的内容无法预测。(在后一种情况下，已知异常应用于主机主地址空间)。

[0236] 没有ESOP的情况下，当保护异常由主机页保护造成并且客户端为DAT开启客户端时，无法预测位61。由于无法区分DAT打开客户端页保护异常和主机页保护异常，因此不能

将主机页保护用于DAT打开客户端。

[0237] 一般而言，上述对实位置的引用在发生客户端中断时引用客户端实位置。如果发生拦截而非客户端中断，则在状态描述中的相应位置处存储信息。

[0238] 如上所述，ECA.18(在状态描述中)是保护拦截控制。当ECA.18为0时，保护异常导致强制拦截。当ECA.18为1时，将阻止拦截并且发生客户端中断。IC.2优先于ECA.18，当IC.2为1时，指定在任何程序中断时发生拦截。SIE状态保留帮助可保持ECA.18。

[0239] 解释执行模式中的增强的保护时抑制

[0240] 针对例如在z/Architecture模式中运行的主机和在z/Architecture模式中的主机控制下的任意架构模式中运行的客户端定义解释执行模式中的增强的保护时抑制。(但是在其他实施例中，主机可以在其他架构模式中运行)。

[0241] ECA.18和ECB.6(在状态描述中)分别为保护拦截和主机保护中断控制。当ECA.18为1以及IC.2为0时，针对客户端识别到的任何保护异常阻止拦截，并且发生客户端中断。当ECA.18为0时，保护异常(除了ECB.6为1时的主机页保护异常)导致强制拦截。

[0242] 如果ECB.6为1，则源于主机页保护的异常表现为主机程序中断。如果ECB.6为0，则主机页保护异常表现为受ECA.18和IC.2控制的客户端程序中断或拦截。在源于主机页保护的主机中断的情况下，将发生以下情况：

[0243] ●识别到异常期间的客户端操作单元或客户端指令执行被废除。

[0244] ●抑制Start Interpretive Execute指令。

[0245] ●将信息存储在主机存储装置中，就好像其是主机中的正常异常。转换异常ID中存储的地址为主机虚拟地址，并且ASCE ID和异常访问ID标识主机地址空间。(对于非MCDS客户端，这是指主机主空间)。

[0246] IC.2优先于源于ECA.18的拦截阻止，当IC.2为1时，指定发生拦截，而非客户端程序中断。但是，如果ECB.6和IC.2均为1，则主机页保护异常仍然呈现为主机中断(也就是说，ECB.6＝1处理的优先级高于IC.2＝1处理)。SIE状态保留帮助可保持ECA.18和ECB.6。

[0247] 当客户端保护异常由主机访问列表控制保护或主机页保护造成并且客户端处于MCDS模式中时，即使当客户端有效地址为被定义为实(与逻辑相对)地址，客户端实位置168-175的位61也被设为1，并且存储客户端有效地址的位0-51，位62和63以及实位置160的内容正确地指示应用异常的主机地址空间。当在非MCDS客户端中识别到源于主机页保护的保护异常并且禁用主机保护中断(也就是说，ECB.6为0)时，客户端实位置168-175的位61被设为0，并且未定义其余字段。

[0248] 将参考图8A-8B描述有关SOP和ESOP的其他详细信息，在图8A-8B中描述了根据本发明的一个方面的SOP/ESOP客户端处理的一个实施例。

[0249] 初始地参考图8A，进入该逻辑基于在环境中执行的可分页客户端所发出的客户端指令，所述可分页客户端尝试在存储位置处执行取回或存储访问(步骤800)。响应于存储访问请求，判定客户端和主机是否允许所请求访问(例如，取回或存储)(询问802)。在该实例中，尝试进行访问，如果成功，则执行访问并且指令完成(步骤804)。但是，如果尝试访问失败，则处理继续判定是主机级别保护还是客户端级别保护导致访问失败。

[0250] 如果不允许访问，则判定对于客户端而言DAT处理是否打开(又称为DAT打开客户端)(询问806)。如果客户端不进行转换(即，它是DAT关闭客户端)，则进一步查询是否针对

客户端设置了键控保护(询问808)。在该实例中,对于DAT关闭客户端,仍可存在实存储块的客户端键控保护。如果没有客户端键控保护,则它不是拒绝访问的客户端级别保护,而是主机级别保护。因此,提供如下所述的主机中断(步骤810)。

[0251] 返回询问808,如果存在客户端键控保护,则提供客户端中断(步骤812)。这向客户端指示它尝试访问受保护的位置。

[0252] 再次参考询问806,如果对于客户端而言DAT处于打开状态,则进一步查询是否存在其他也可应用的保护形式,包括例如低地址保护、访问列表控制保护、键控保护或DAT保护(在增强的DAT之前,也称为页保护)。如果不应用这些其他保护形式中的任何形式,则其是主机问题(即,导致访问失败的主机级别保护),因此呈现主机中断(步骤810)。但是,如果应用这些其他保护形式之一,则提供客户端中断(步骤816)。

[0253] 此外,如果应用客户端保护方案,则判定是否是适用的客户端DAT保护(DATP)以及请求是否针对存储访问(询问818)。如果否,则处理客户端异常(步骤820)。在一个实例中,当客户端收到这些保护条件之一时,系统管理程序(例如,主机)执行策略所确定的任何所需或希望的处理。

[0254] 返回到询问818,如果是客户端DATP和存储访问,则客户端操作系统理解客户端级别的写时复制(步骤822)。如果没有写时复制意图,则可执行客户端异常。

[0255] 参考图8B介绍了有关处理主机中断的其他详细信息。初始地,判定保护中断是否源于存储访问时的主机DATP(询问840)。如果否,则将条件反映给客户端(步骤842)。在一个实例中,主机执行任何所需的操作以向客户端指示导致错误的特定条件。

[0256] 再次参考询问840,如果是存储访问时的主机DATP,则判定是否为DAT开启客户端(询问844)。如果是DAT开启客户端,则进一步询问是SOP还是ESOP处理正在控制(询问846)。如果是SOP处理,则将条件反映给客户端(步骤848)。但是,如果是ESOP处理,则对于遇到DATP条件的存储访问尝试,转换异常标识(TEID)中的位61是可预测的。没有ESOP的情况下,客户端DAT开启将导致终止向客户端反映。

[0257] 如果客户端DAT处于关闭状态(询问844),或者如果客户端DAT处于开启状态并且是ESOP(询问846),则出现以下情况。判定要访问的存储区域(例如,页)是否为主机写时复制方案的一部分(询问850)。如果否,则将条件反映给客户端(步骤852)。但是,如果是主机写时复制方案的一部分,则执行主机写时复制(步骤854)。

[0258] 有利地,ESOP提供了主机和客户端了解存储(例如,页)错误情况以及谁将对此负责的手段。ESOP工具提供了在解释操作期间区分主机页保护和客户端页保护的机器(例如,CPU)的能力。之前,不能辨别页保护是因为客户端页错误还是主机页错误。也就是说,页保护异常可能源于客户端,也可能源于主机以及某种已存在的不可预测性。因此,主机不能对此做出更多操作。通过ESOP,消除了先前的不可预测性并且主机页保护导致主机程序中断,并带有有关主机坏页的信息(例如,失效的主机地址)。类似地,客户端页保护产生客户端程序中断,并带有有关客户端页的信息(例如,失效的客户端地址)。

[0259] 先前,仅可针对主机程序视为DAT关闭客户端(例如,CMS、MCDS)的客户端执行主机COW,因为对于DAT关闭客户端,页保护一定是主机页保护。但是,对于DAT开启客户端,无法确定页保护是主机页保护还是客户端页保护,因此,不能使用COW。

[0260] 根据本发明的一个方面,这种情况已经改变。增强的保护时抑制(ESOP)工具允许

区分主机级别保护和客户端级别保护，因此，允许在适当时使用COW。在使用ESOP时，附加功能允许执行区分操作以便即使针对DAT开启客户端进行拦截(可能受阻于其自己的受保护页)，主机也有足够的信息来了解属于哪种异常，当所述异常属于主机异常时，仍能执行写时复制。

[0261] 测试保护

[0262] 根据本发明的一个方面，提供了允许测试保护异常的能力。作为一个实例，Test Protection(TPROT)指令用于为执行TPROT指令的程序提供有关所述程序要使用的存储装置的信息。例如，它提供有关存储保护的信息并允许区分主机级别保护和客户端级别保护。根据本发明的一个或多个方面，TPROT利用ESOP。

[0263] 为了进一步说明，当程序分配存储帧时，通常会执行查询命令以了解帧是否具有允许程序访问的属性(即，它是常驻的，没有写保护)。作为实例，在IBM Syestem z处理器中，TEST Protection指令用于此目的，在Sun Microsystems中，使用mincore()系统调用。通过发出指令或调用，程序通过前者中的条件码以及后者中的返回码获知帧的属性。当发出存储查询命令的程序在主机系统管理程序的控制下作为客户端解释执行时，从客户端地址转换表以及主机地址转换表的角度，返回码可反映帧的属性。此行为将揭示使用特定主机存储控制方案(例如，到客户端的写时复制)的效果，因此弱化了系统管理程序提供的隔离。

[0264] 如本文所述，写时复制(COW)方案用于减轻操作系统给存储器造成的压力。随着处理器性能的增加，在单个操作系统的控制下运行多个程序以及在单个系统管理程序的控制下运行多个操作系统的能力也相应地增加。为了适应大量并行程序，采取了专门的措施来管理系统存储需求。

[0265] 这些措施之一是使用COW。在操作系统启动程序时，最初分配的存储帧被标记为写保护，并在同一操作系统的控制下与其他程序物理共享所述存储帧。然后，当尝试对其中一个帧执行存储操作时，将出现程序异常并且操作系统分配唯一的存储帧并复制原始共享帧的内容。

[0266] 有些程序可以使用返回客户端级别和主机级别处的帧属性的综合属性内容的存储查询命令成功地运行。但是，如果程序被编写为避免分配写保护存储，则当存储查询命令返回所有被查询帧的写保护指示时(没有任何资源可进行客户端角度的帧修改)，其可能失败。如果系统管理程序未实施写时复制方案，则会出现这种情况。

[0267] 有些系统可以允许客户端程序被编写为使用半虚拟化(para-virtualization)；通过定义的接口将客户端要求传递给主机。在此情况下，客户端可以直接向主机指出需要使用存储并且主机可以按照上面所述通过分配复制的帧做出响应。在此情况下，客户端程序需要进行修改以适合到它在其上执行的每个系统管理程序主机的接口。此外，半虚拟化调用通常具有较高的开销，并给系统管理程序造成执行压力。

[0268] 当解释执行的客户端程序针对存储帧发出查询命令时，如果存在命令的条件码或返回码将指示所述帧为写保护(仅因为所述帧在主机地址转换表中是写保护的)的条件，则会发生程序拦截，而非根据反映主机地址转换表状态的条件码或返回码指示完成执行。

[0269] 根据本发明的一个方面，诸如写时复制之类的特定主机存储管理技术对客户端程序透明地运行。它允许程序逻辑依赖于存储查询命令的结果以便仅反映客户端角度的帧存

储属性。如果页仅在客户端地址转换表中是写保护的，则客户端无需进行系统调用或与系统管理程序进行通信便可获知此情况。

[0270] 在一个实例中，Test Protection(TPROT)指令将帧地址作为输入操作数，并使用以下四种返回码之一完成：

[0271] 0允许取回；允许存储

[0272] 1允许取回，不允许存储

[0273] 2不允许取回，不允许存储

[0274] 3转换不可用。

[0275] 根据本发明的一个方面，提供了一种用于执行Test Protection(TPROT)指令以提供特定益处的工具，所述益处包括但不限于区分主机级别保护和客户端级别保护；以及提供指示执行成功完成的伪返回码，尽管不更新转换表以适应COW。在该实施例中，系统管理程序设置逻辑分区的客户端映像的状态值(ECB.6)；所述客户端映像的客户端程序取回要执行的TPROT指令，所述TPROT指令指定要测试的存储位置；所述TPROT指令测试要保护的存储位置(存储区)；响应于指示无抑制的状态值，执行根据测试设置条件码或执行异常操作中的任意一项；或者响应于指示抑制的状态值，执行响应于指示应设置条件码的测试，抑制指令的执行；以及执行指令拦截。

[0276] 在一个实例中，当以下所有条件为真时，可分页模式客户端对Test Protection指令的执行受到抑制并且识别到主机中断(而非设置条件码1)：

[0277] ●已在主机中安装增强的保护时抑制工具。

[0278] ●ECB.6为1。

[0279] ●存在要求设置条件码1的条件。

[0280] ●存在主机页保护异常条件。

[0281] ●既不存在访问列表控制保护异常条件，也不存在客户端页保护异常条件。

[0282] 在存储引用期间，当遇到客户端页保护条件时，除非由于键控保护设置条件码2，否则由客户端设置条件码1，并且主机不执行任何操作。对于主机页保护，如果上述指定的条件为真，将抑制客户端的执行并识别到指令拦截，并且主机不执行任何操作。

[0283] 参考图9描述了TPROT指令格式的一个实例。如图所示，TPROT指令900例如包括标识TPROT指令的操作码902(例如，E501)；第一寄存器字段(B_1)904和第一位移字段(D_1)906；以及第二寄存器字段(B_2)908和第二位移字段(D_2)910。通过将D_1字段的内容与由B_1字段标识的寄存器的内容相加来确定第一操作数地址。同样，通过将D_2字段的内容与由B_2字段指定的寄存器的内容相加来形成第二操作数地址。

[0284] 在执行中，通过使用由例如第二操作数的位56-59指定的存储键测试由第一操作数地址指定的位置以查找保护异常。

[0285] 所述第二操作数地址不用于地址数据；相反，所述地址的位56-59形成在测试中使用的访问键。在该实例中，忽略第二操作数地址的位0-55和60-63。

[0286] 所述第一操作数地址为逻辑地址。当CPU处于存取寄存器模式(当DAT开启并且PSW位16和17为二进制01)时，第一操作数地址通过存取寄存器转换(ART)和动态地址转换(DAT)过程进行转换。ART应用于由B1字段指定的存取寄存器，并且获取要由DAT使用的地址空间控制元素。当DAT开启，但是CPU未处于存取寄存器模式时，将通过DAT转换第一操作数

地址。在此情况下，当CPU分别处于主空间模式、第二空间模式或归属空间模式时，DAT使用控制寄存器1、7或13中包含的地址空间控制元素。当DAT关闭时，第一操作数地址为实地址，不通过ART或DAT进行转换。

[0287] 当CPU处于存取寄存器模式并且由于通常导致下表中示出的异常之一的条件而使ART无法获取地址空间控制元素时，通过设置条件码3完成指令。

[0288]

异常名称	原因
ALET 指定	访问列表表项令牌（ALET）位 0-6 不全为 0
ALEN 转换	访问列表表项（ALE）位于列表之外或无效（位 0 为 1）
ALE 序列	ALET 中的 ALE 序号（ALESN）不等于 ALE 中的 ALESN
ASTE 有效性	ASN 第二表表项（ASTE）无效（位 0 为 1）
ASTE 序列	ALE 中的 ASTE 序号（ASTESN）不等于 ASTE 中的 ASTESN
扩展授权	ALE 专用位不为 0，ALE 授权索引（ALEAX）不等于扩展授权索引（EAX），并且 EAX 选择的第二位位于授权表之外或为 0

[0289] 当存取寄存器包含十六进制00000000或00000001时，ART分别从控制寄存器1或7获取地址空间控制元素而无需存取访问列表。当B_1字段指定存取寄存器0时，ART将存取寄存器视为包含十六进制00000000并且不检查存取寄存器的实际内容。

[0290] 当ART成功地完成时，继续通过执行DAT来执行操作。

[0291] 当DAT开启并且由于通常导致下表中示出的异常之一的条件而无法转换第一操作数地址时，通过设置条件码3完成指令。

[0292]

异常名称	原因
ASCE 类型	正在使用的地址空间控制元素（ASCE）为区第二表指定，并且第一操作数地址的位 0-10 不全为 0；ASCE 为区第三表指定，并且第一操作数地址的位 0-21 不全为 0；或者 ASCE 为段表指定，并且第一操作数地址的位 0-32 不全为 0。
区第一转换	区第一表表项位于表之外或无效。
区第二转换	区第二表表项位于表之外或无效。
区第三转换	区第三表表项位于表之外或无效。
段转换	段表表项位于表之外或无效。
页转换	页表表项无效。

[0293] 当第一操作数地址转换可以完成，或者当DAT关闭时，针对第二操作数地址的位位置56-59中指定的访问键测试由第一操作数地址指定的块的存储键，并且将条件码设为指示是否允许存储和取回访问，其中考虑适用的保护机制。因此，例如，如果低地址保护处于活动状态并且第一操作数有效地址位于范围0-511或4096-4607中，则不允许存储访问。另外还考虑了访问列表控制保护、页保护、存储保护忽略和取回保护忽略。

[0294] 存储内容(包括更改位)不受影响。根据模型的不同，可以将第一操作数地址的引用位设为1，即使对位置实施取回保护也是如此。

[0295] 当CPU处于存取寄存器模式时，如果ART用于取回有效访问列表指定或ALE、ASTE或授权表表项的地址指定的位置不在配置中，将识别到寻址异常。

[0296] 当DAT开启时，如果区表表项、段表表项或页表表项的地址或转换后的操作数实地址指定的位置不在配置中，将识别到寻址异常。此外，如果区表表项或段表表项或页表表项具有格式错误，将识别到转换指定异常。在该实例中，当DAT关闭时，仅识别到由于应用操作数实地址导致的寻址异常。

[0297] 对于上述所有情况，将抑制操作。

[0298] 结果条件码：

[0299] 0允许取回；允许存储

[0300] 1允许取回；不允许存储

[0301] 2不允许取回；不允许存储

[0302] 3转换不可用。

[0303] 程序异常：

[0304] ●寻址(有效访问列表指定、访问列表表项、ASN第二表表项、授权表表项、区表表项、段表表项、页表表项或操作数1)

[0305] ●特权操作。

[0306] ●转换指定。

[0307] Test Protection允许程序确定通过调用程序在不出现程序异常的情况下传递的地址的保护属性。指令设置条件码以指示由指令的第一操作数地址指定的位置处是否允许取回或存储。所述指令考虑了机器中的保护机制:例如,访问列表控制保护、页保护、键控保护、低地址保护、存储保护忽略和取回保护忽略。另外,由于ASCE类型、区转换、段转换和页转换异常条件可以是保护违反的程序表现,因此,这些条件用于设置条件码而非导致程序异常。当CPU处于存取寄存器模式时,Test Protection还允许程序检查存取寄存器中的访问列表表项令牌(ALET)的可用性而不会导致程序异常。将检查ALET的有效性(无ALET指定、ALEN转换和ALE序列异常条件)及其是否被授权由程序使用(无ASTE有效性、ASTE序列和扩展授权异常条件)。

[0308] 使用Test Protection的方法具有测试的优势,所述测试不会导致中断;但是,所述测试和使用在时间上是分开的并且如果所讨论位置的存储键可能在测试时间和使用时间之间发生变化,则所述测试和使用可能是不准确的。

[0309] 在处理动态地址转换时,Test Protection类似于Load Real Address,因为这两个指令不会导致ASCE类型、区转换、段转换和页转换异常。相反,这些异常条件通过条件码设置来指示。类似地,在执行这两个指令之一期间执行存取寄存器转换时,所述转换将设置特定异常条件的条件码。导致Load Real Address的条件码1、2和3的条件将导致Test Protection的条件码3。这两个指令在其他一些方面也有差别。Test Protection的第一操作数地址为逻辑地址,因此当DAT关闭时不进行动态地址转换。Load Real Address的第二操作数地址是转换后的的虚拟地址。

[0310] 当CPU处于存取寄存器模式(DAT开启)时,存取寄存器转换应用于Test Protection,而当PSW位16和17为二进制01时,不管DAT是开启还是关闭,该转换都应用于Load Real Address。当由于存取寄存器转换中的异常条件而设置条件码3时,Load Real Address(而非Test Protection)在通用寄存器中返回分配给异常的程序中断码。

[0311] 条件码3不一定指示程序始终无法访问第一操作数位置;它仅指示当前条件阻止指令确定操作数的保护属性。例如,在虚拟存储环境中,如果操作系统已将存储位置换出,则可以设置条件码3。如果程序尝试访问所述位置,则操作系统可以解决页转换异常并随后使得所述位置可以被程序访问。类似地,条件码1不一定指示地址永远无法被存储。在实现POSIXfork功能的操作系统中,页保护用于警告操作系统写时复制事件。在操作系统解决写时复制事件之后,可以授予程序对所述位置的存储访问权限。

[0312] 参考图10A-10B描述了根据本发明的一个方面的与执行Test Protection指令关联的逻辑的一个实施例。初始地,客户端执行Test Protection指令以了解有关正在通过Test Protection指令查询的存储的信息(步骤1000)。在执行期间,判定客户端和主机是否允许存储(询问1002)。在一个实例中,通过测试是否可以发生存储来做出此判定,尽管存储实际上未发生。例如,通过存储地址检查各种权限和测试,但是不执行存储。如果存储在执行之后可以成功,则认为客户端和主机都允许存储。因此,以成功条件码(例如,0)完成查询指令(步骤1004)。

[0313] 但是,如果客户端和/或主机不允许存储,则进一步查询客户端地址转换是否可用(询问1006)。作为实例,如果客户端为DAT关闭客户端,则转换可用,在这个意义上,标识为

转换后的地址。但是，如果客户端为DAT打开客户端，则搜索一个或多个客户端转换表以查看相应的表项是否有效。如果有效，则客户端地址转换可用于DAT打开客户端。如果客户端地址转换不可用，则以指示此情况的条件码(例如，CC＝3)完成Test Protection指令(步骤1008)。

[0314] 如果客户端地址转换可用，则进一步判定主机地址转换是否可用(询问1010)。此判定的做出类似于判定客户端地址转换是否可用。但是，对于DAT开启主机，使用DAT转换表而非客户端转换表。如果主机地址转换不可用，则执行主机中断(步骤1012)。在一个实例中，存在客户端并且存在一个被调用以管理此条件的主机异常例程。

[0315] 如果客户端地址转换和主机地址转换均可用，则进一步判定客户端和主机是否允许取回(询问1014)。在一个实例中，此判定基于DAT表中存储的权限。如果客户端和主机均不允许取回，则以选定条件码(例如，CC＝2)完成指令(步骤1016)。如果客户端和主机均允许取回，则进一步判定是否已启用主机保护异常条件陷阱(询问1018)。在一个实例中，当启用了ESOP并且ECB.6开启时，作为本发明的一个方面的主机保护异常条件处于启用状态。如果未启用主机保护异常条件陷阱，则以选定条件码(例如，CC＝1)完成指令(步骤1020)。如果不启用陷阱，不会专门区分客户端和主机，也不会区分DAT和其他类型的存储保护。

[0316] 如果已启用主机保护异常条件陷阱，则进一步判定客户端设置是否允许存储(询问1022)。在一个实例中，通过检查例如转换表(例如，客户端DAT表)和/或其他数据结构(例如包括客户端所设的保护)中由客户端所设的设置来做出此判定。如果允许存储，则以选定条件码(例如，CC＝1)完成指令(步骤1024)。但是，如果客户端允许存储，则进一步判定主机写保护是否不同于DAT(询问1026)。如果是，则以指定条件码(例如，CC＝1)完成指令(步骤1028)。但是，如果仅存在主机DAT保护，则发生主机拦截(步骤1030)。

[0317] 此时，主机仿真TPROT指令(步骤1032)。也就是说，当指令的客户端执行导致主机拦截，则主机程序修改客户端环境，以便当(由主机)重新启动客户端程序时，对客户端而言似乎已执行拦截指令。在仿真期间，判定要被访问的存储区域(例如，页)是否为主机COW方案的一部分(询问1040)。如果所述页不是主机COW方案的一部分，从而指示不能对其进行写入操作，则提供选定完成码(例如，CC＝1)(步骤1042)。但是，如果所述页是主机COW方案的一部分，则发出带有成功完成码(例如，CC＝0)的TPROT(步骤1044)。

[0318] 根据本发明的一个方面，通过此成功条件码，主机不更改其DAT表的状态。帧保持写保护状态。在主机COW方案中，当客户端实际对帧执行存储操作时，将发生主机中断(例如，通过ESOP)并且主机异常例程将执行COW。该“伪”返回码是本发明的优选实施例的一个方面的优势，因为当针对永远不进行存储的块发出TPROT时，可避免执行COW。

[0319] 在一个实施例中，提供了保护查询指令(例如，TPROT)，其中当主机表中的保护设置为取回保护时，将发生主机中断。但是，当客户端表中的保护设置为取回保护时，将返回条件码。

[0320] 作为一个实例，如果解释地执行TPROT指令，并且从主机角度(即，在主机地址转换表中)而非客户端角度(即，在客户端地址转换表中)而言，仅因为帧处于写保护而使得以CC＝1完成该指令，则会发生主机拦截。在处理所述拦截时，如果主机正在实施写时复制(COW)存储控制方案并且如果帧确实可由客户端程序进行修改，则会生成所述帧的副本，更新主机地址转换表，并且通过以CC＝0完成TPROT指令而继续客户端执行。

[0321] 本发明的一个或多个方面可以包括在具有例如计算机可用介质的制品(例如,一个或多个计算机程序产品)内。所述介质包含例如计算机可读程序代码装置或逻辑(例如,指令、代码、命令等)以提供和促进本发明的能力。所述制品可以作为计算机系统的一部分,也可以单独出售。

[0322] 参考图11描述了结合本发明的一个或多个方面的制品或计算机程序产品的一个实例。计算机程序产品1100例如包括一个或多个计算机可用介质1102以存储计算机可读程序代码装置或逻辑1104以便提供和促进本发明的一个或多个方面。所述介质可以是电、磁、光、电磁、红外线或半导体系统(或装置或设备)或传播介质。计算机可读介质的实例包括半导体或固态存储器、磁带、可移动计算机盘、随机存取存储器(RAM)、只读存储器(ROM)、硬磁盘和光盘。光盘的当前实例包括光盘-只读存储器(CD-ROM)、光盘-读/写(CR-R/W)和数字多用光盘(DVD)。

[0323] 一个或多个计算机可读程序代码装置或逻辑定义的一系列程序指令或一个或多个相关模块的逻辑组合引导执行本发明的一个或多个方面。

[0324] 商业实现

[0325] 尽管此处提及了由IBM的z/Architecture,但是本发明的一个或多个方面同样适用于其他采用可分页实体或类似结构的机器架构和/或计算环境。

[0326] 此处披露的TPROT指令、工具以及其他格式、指令和属性的商业实现可以通过硬件实现,也可以由例如使用汇编语言的编程人员(如操作系统编程人员)实现。此类编程指令可存储在旨在在诸如z/Architecture IBMserver之类的计算环境中本机地执行,或备选地在执行其他架构的机器中本机地执行的存储介质中。所述指令可以在现有和未来的IBM服务器中以及在其他机器或主机上进行仿真。它们可以在其中总体执行处于仿真模式的机器中执行。

[0327] 在仿真模式中,解码被仿真的特定指令,然后构建子例程以实现位于所述子例程或驱动程序中的各个指令,或者使用其他某种技术提供特定硬件的驱动程序,本领域的技术人员在理解了有关说明之后便可理解此技术。各种软件和硬件仿真技术在多个美国专利中进行了描述,这些专利包括:5,551,013、5,574,873、5,790,825、6,009,261、6,308,255和6,463,582。许多其他教导进一步示出了各种仿真具有目标机器架构的指令格式的方法。

[0328] 除了上述内容之外,在Farrell等人的标题为“Method and Apparatus for Managing the Execution of a Broadcast Instruction on a Guest Processor(用于管理广播指令在客户端处理器上的执行的方法和装置,2007年3月27日公告)”的美国专利第7,197,585号中描述了有关客户端处理器和相关处理的其他详细信息。

[0329] 其他变型和架构

[0330] 尽管此处描述了各种实例和实施例,但是它们仅作为实例,许多变型也包括在本发明的范围之中。例如,此处所述的计算环境仅是一个实例。许多其他环境,包括其他类型的通信环境,可以包括本发明的一个或多个方面。例如,可以采用不同类型的处理器、客户端和/或主机。此外,可分页主机以及可分页客户端可以使用本发明的一个或多个方面。此外,其他类型的架构可以采用本发明的一个或多个方面。

[0331] 本发明的优选实施例的各个方面对于许多类型的环境是有利的,其中包括具有多个区的其他环境以及非分区环境。进而,可以没有中央处理器复合体,但是仍存在多个耦合

在一起的处理器。此处所述的各个方面可适用于单处理器环境。

[0332] 此外，在此处提供的数据结构和流的实例中，创建和/或使用不同的字段可以包括许多变型，例如不同数量的位；不同顺序的位；与此处所述相比更多、更少或不同的位；更多、更少或不同的字段；不同顺序的字段；不同大小的字段等。此外，提供这些字段仅作为实例，还可以包括许多变型。此外，此处所述的指示符和/或控制可以具有不同的形式。例如，它们可以通过“位”以外的方式进行表示。此外，尽管此处使用了术语“地址”，但是可以使用任意指定。

[0333] 如此处使用的，术语“页”用于指固定大小或预定大小的虚拟存储(即，虚拟存储器)区。作为一个实例，主机页是主机虚拟存储区域。尽管在此处提供的实例中，页大小为4KB，但是页大小是可变的。此外，“帧”用于指固定大小或预定大小的实存储或绝对存储(即，存储器)区域。作为实例，主机帧是主机实存储或绝对存储区域，并且客户端帧是客户端实存储或绝对存储区域。在可分页客户端中，客户端实存储或绝对存储通过主机虚拟存储进行映射。一般而言，主机虚拟存储页根据需要由主机实存储或绝对存储进行支持。尽管在此处提供的实例中，帧大小为4KB或1MB，但是帧大小是可变的。然而在其他实施例中，可以存在不同大小的页、帧、段、区、存储块等。此外，在其他架构中，术语“页”和“段”可互换使用或者术语“页”可应用于多个虚拟存储大小单位。术语“获取”(例如获取指令)包括但不限于取回、具有、接收、被提供、创建、形成、发出等。指令可引用其他寄存器或者可引用寄存器以外的元素，例如操作数、字段、位置等。上述元素的许多其他备选物也是可能的。此外，尽管此处使用了诸如列表、表之类的术语，但是可以使用任何类型的数据结构。例如，表同样可以包括其他数据结构。另外，此处所述的元素仅作为实例。

[0334] 此外，适合于存储和/或执行程序代码的数据处理系统是可用的，并且包括至少一个直接或通过系统总线间接连接到存储器元件的处理器。所述存储器元件例如包括在程序代码的实际执行期间采用的本地存储器、大容量存储装置以及提供至少某些程序代码的临时存储以减少必须在执行期间从大容量存储装置检索代码的次数的高速缓冲存储器。

[0335] 输入/输出或I/O设备(包括但不限于键盘、显示器、指点设备、DASD、磁带、CD、DVD、闪盘和其他存储介质等)可以直接或通过中间I/O控制器与系统相连。网络适配器也可以被连接到系统以使所述数据处理系统能够通过中间专用或公共网络变得与其他数据处理系统或远程打印机或存储设备相连。调制解调器、电缆调制解调器和以太网卡只是可用的网络适配器类型中的少数几种。

[0336] 本发明的一个或多个方面的能力可以通过软件、固件、硬件或它们的某种组合实现。可以提供至少一个可由机器读取的程序存储设备，所述程序存储设备承载至少一个可由机器执行以执行本发明的能力的指令程序。

[0337] 此处描述的流程图仅作为实例。所述的这些图形或步骤(或操作)可以具有多种不偏离本发明的优选实施例的精神的变型。例如，所述步骤可以按照其他顺序执行，也可以添加、删除或修改步骤。所有这些变型均被视为要求保护的发明的一部分。

[0338] 尽管此处详细地描述和说明了各实施例，但是对于相关领域的技术人员将显而易见的是，可以在不偏离本发明的优选实施例的精神的情况下做出各种修改、添加、替换等，因此它们被视为处于下面的权利要求中限定的本发明的范围之内。

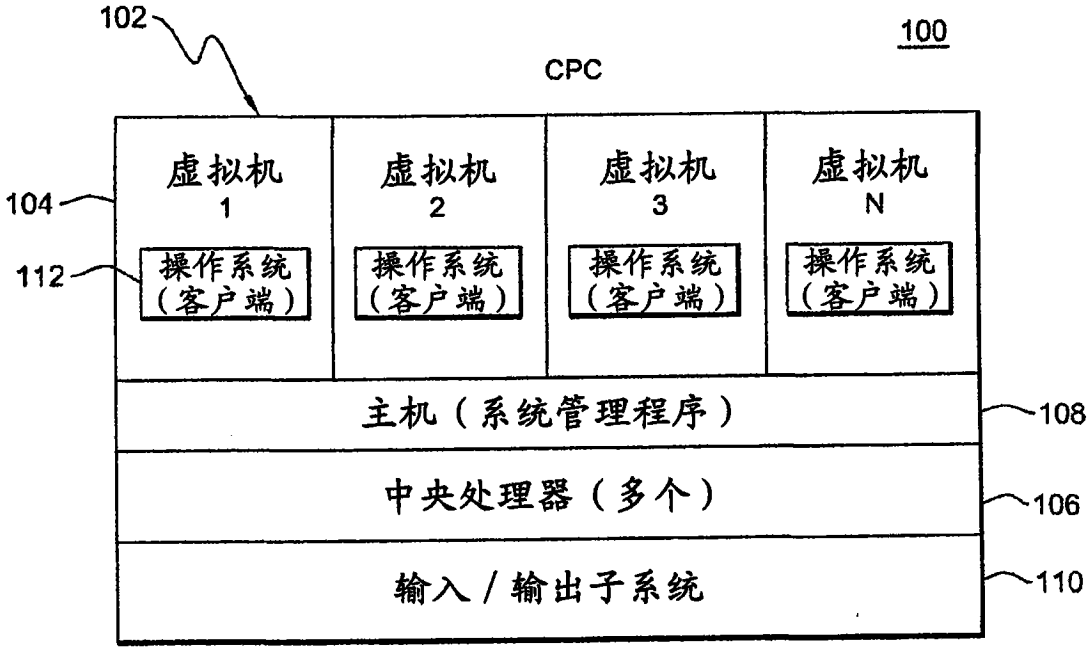

102
100
CPC
104
112
虚拟机 1
虚拟机 2
虚拟机 3
虚拟机 N
操作系统（客户端）
操作系统（客户端）
操作系统（客户端）
操作系统（客户端）
主机（系统管理程序）
108
中央处理器（多个）
106
输入/输出子系统
110

图1

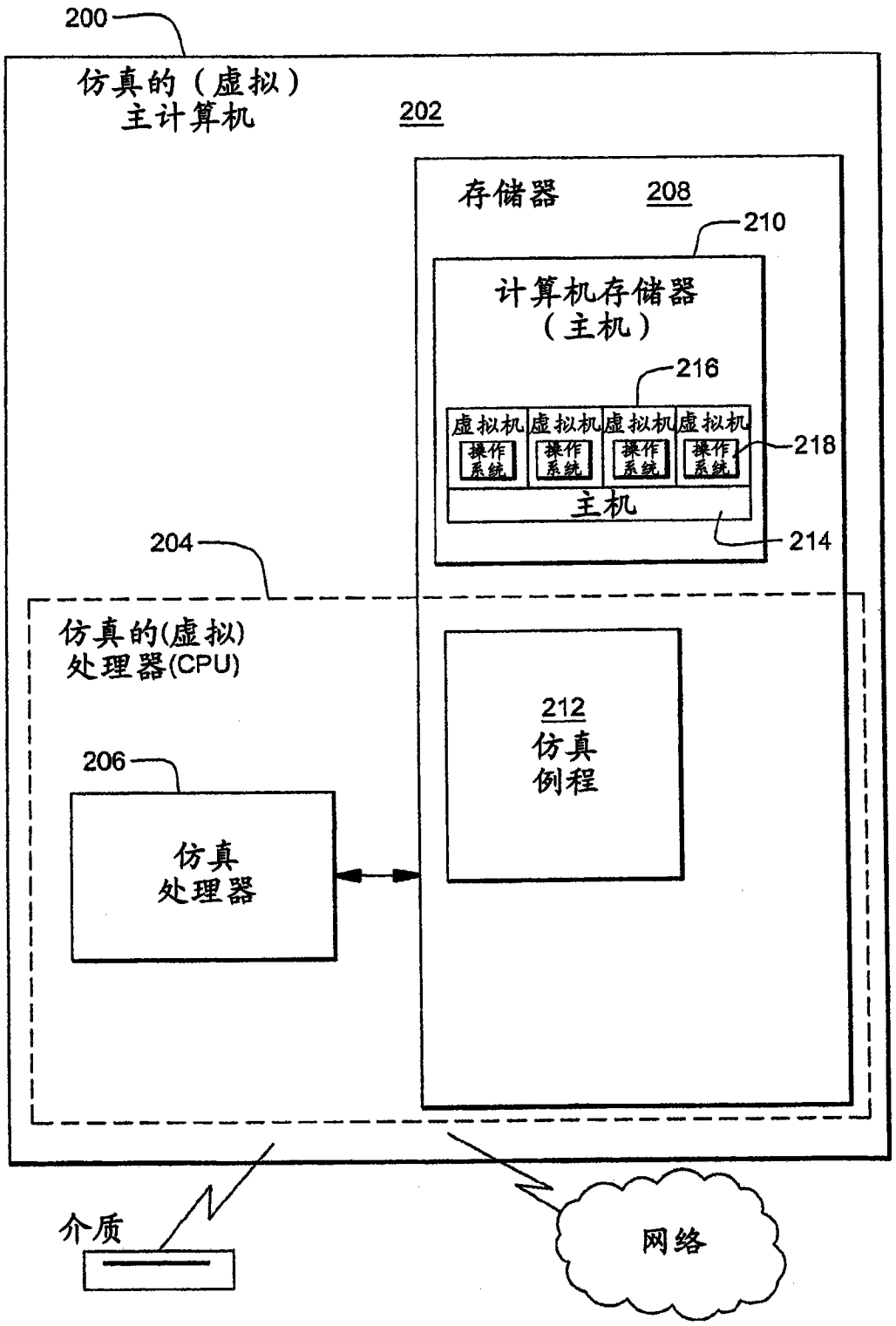

200
仿真的（虚拟）
主计算机
202
存储器
208
210
计算机存储器
（主机）
216
虚拟机
虚拟机
虚拟机
虚拟机
操作系统
操作系统
操作系统
操作系统
218
主机
214
204
仿真的(虚拟)
处理器(CPU)
212
仿真
例程
206
仿真
处理器
介质
网络

图2

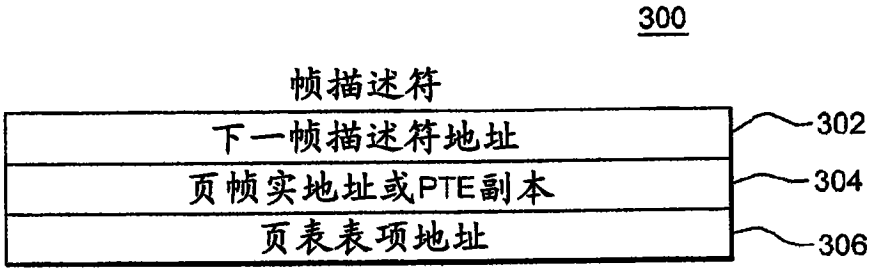


图3

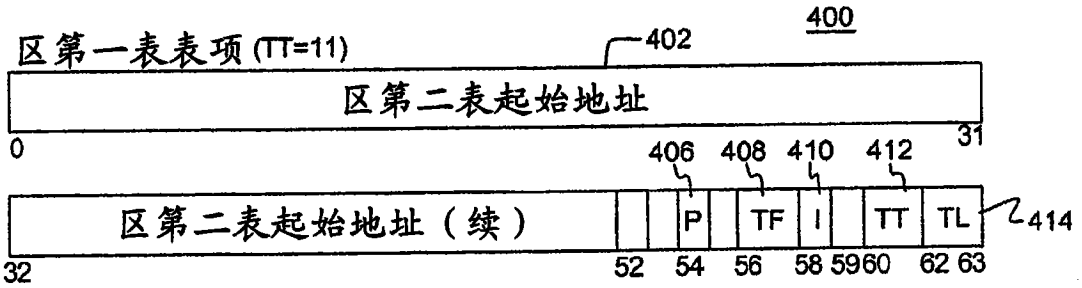


图4A

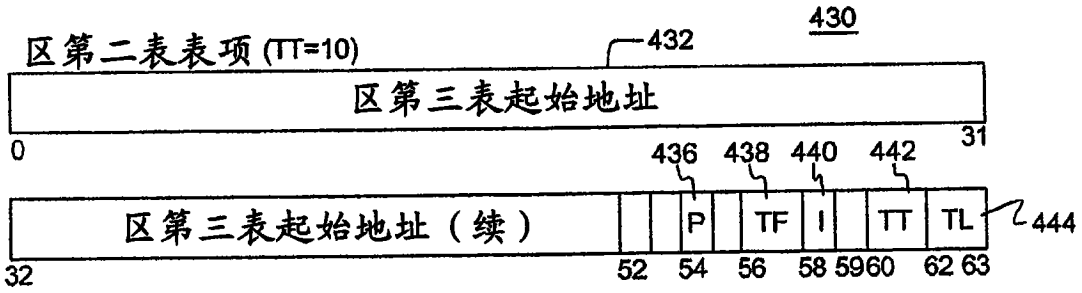


图4B

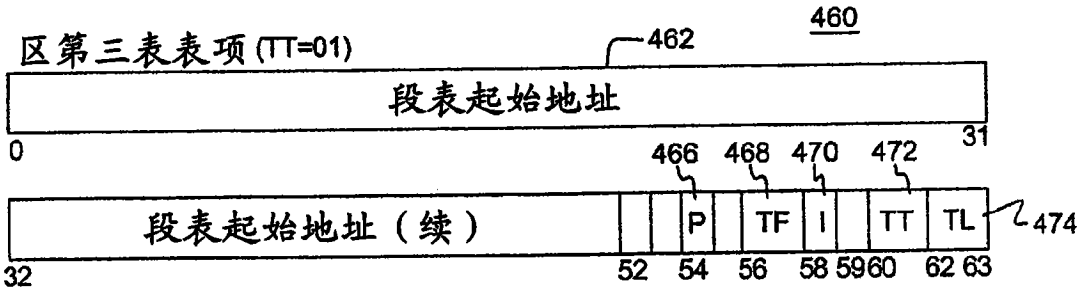
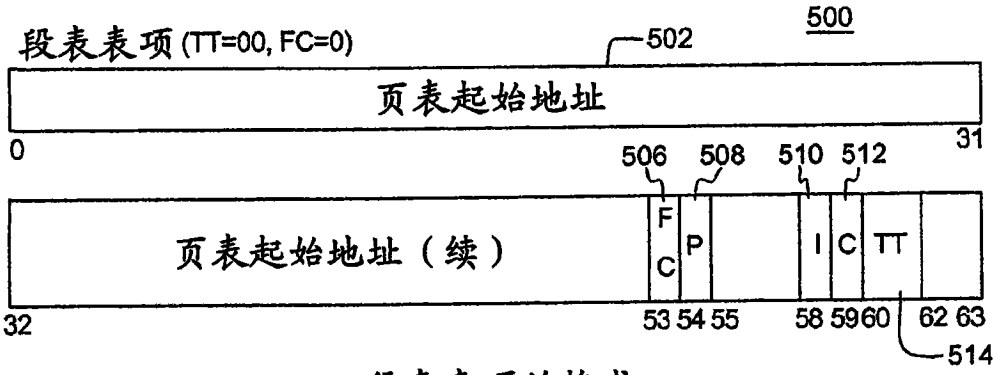
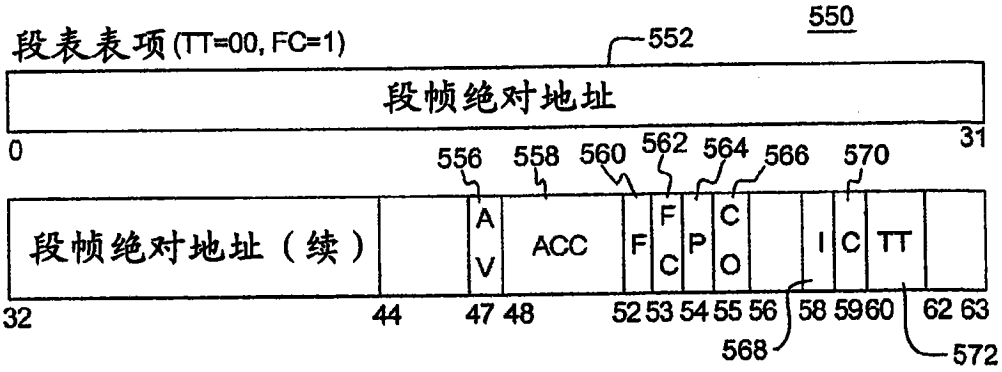


图4C



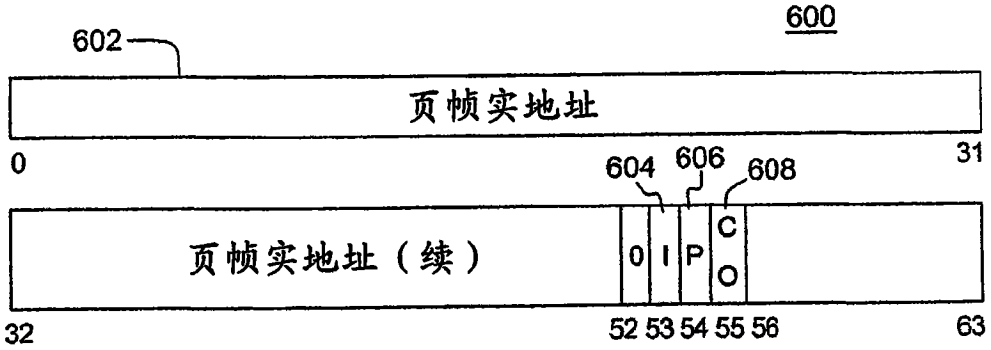
段表表项的格式0

图5A



段表表项的格式1

图5B



页表表项的格式

图6

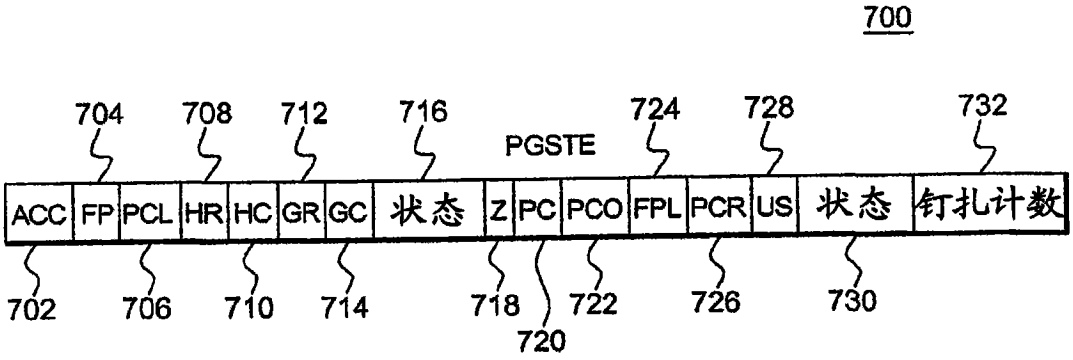

700
704
708
712
716
PGSTE
724
728
732
ACC
FP
PCL
HR
HC
GR
GC
状态
Z
PC
PCO
FPL
PCR
US
状态
钉扎计数
702
706
710
714
718
720
722
726
730

图7

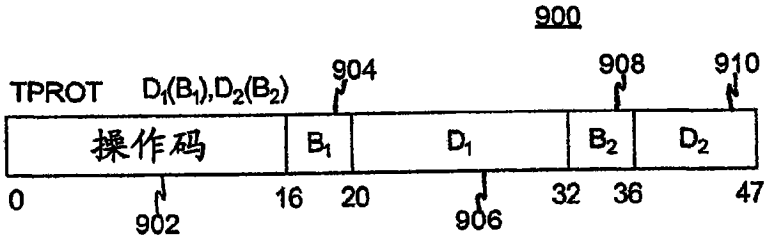

900
TPROT
D1(B1),D2(B2)
904
908
910
操作码
B1
D1
B2
D2
0
902
16
20
906
32
36
47

图9

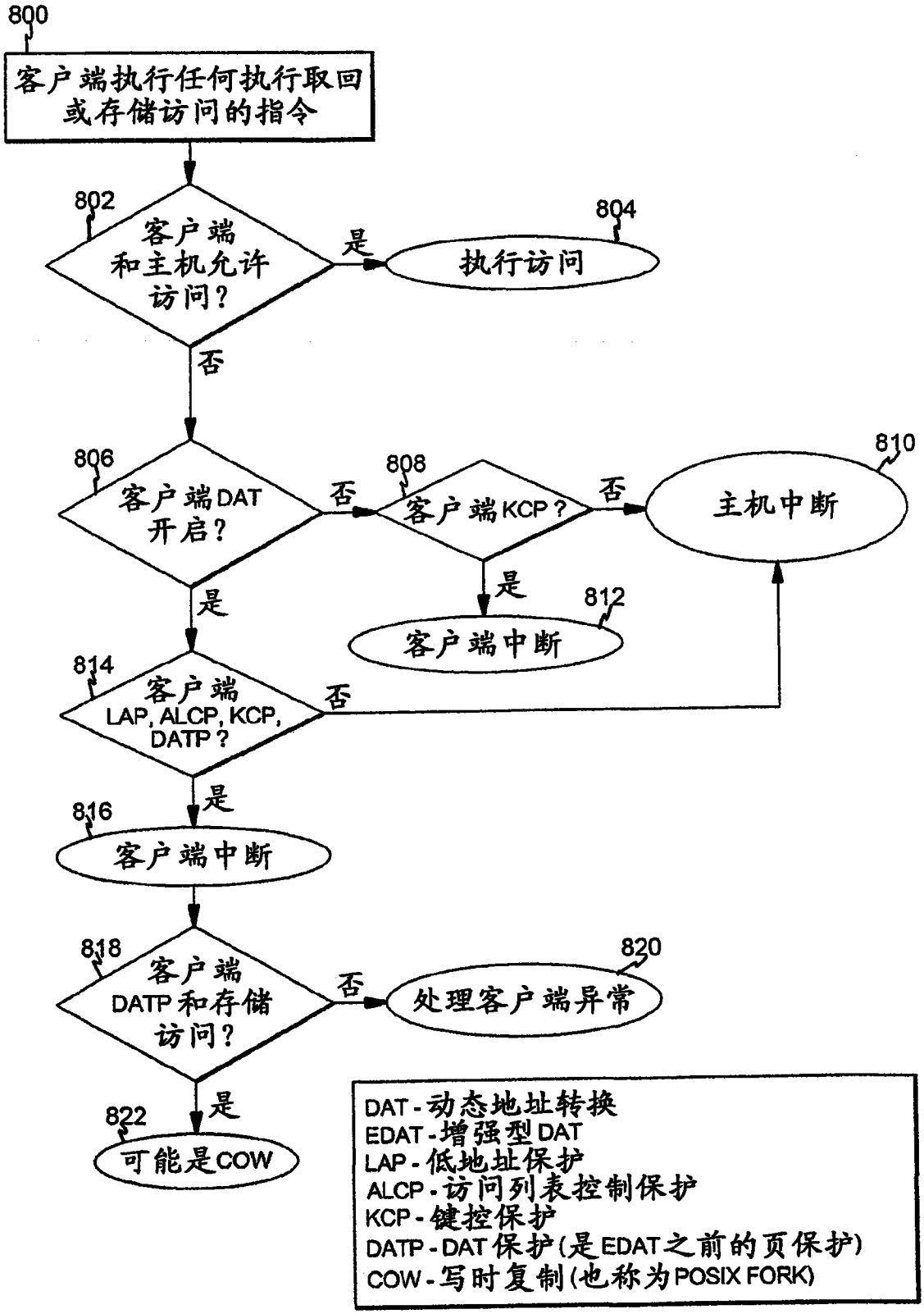

800
客户端执行任何执行取回或存储访问的指令
802
客户端和主机允许访问？
是
804
执行访问
否
806
客户端DAT开启？
否
808
客户端KCP？
否
810
主机中断
是
812
客户端中断
是
814
客户端LAP, ALCP, KCP, DATP？
否
是
816
客户端中断
818
客户端DATP和存储访问？
否
820
处理客户端异常
是
822
可能是COW
DAT - 动态地址转换
EDAT - 增强型DAT
LAP - 低地址保护
ALCP - 访问列表控制保护
KCP - 键控保护
DATP - DAT保护(是EDAT之前的页保护)
COW - 写时复制(也称为POSIX FORK)

图8A

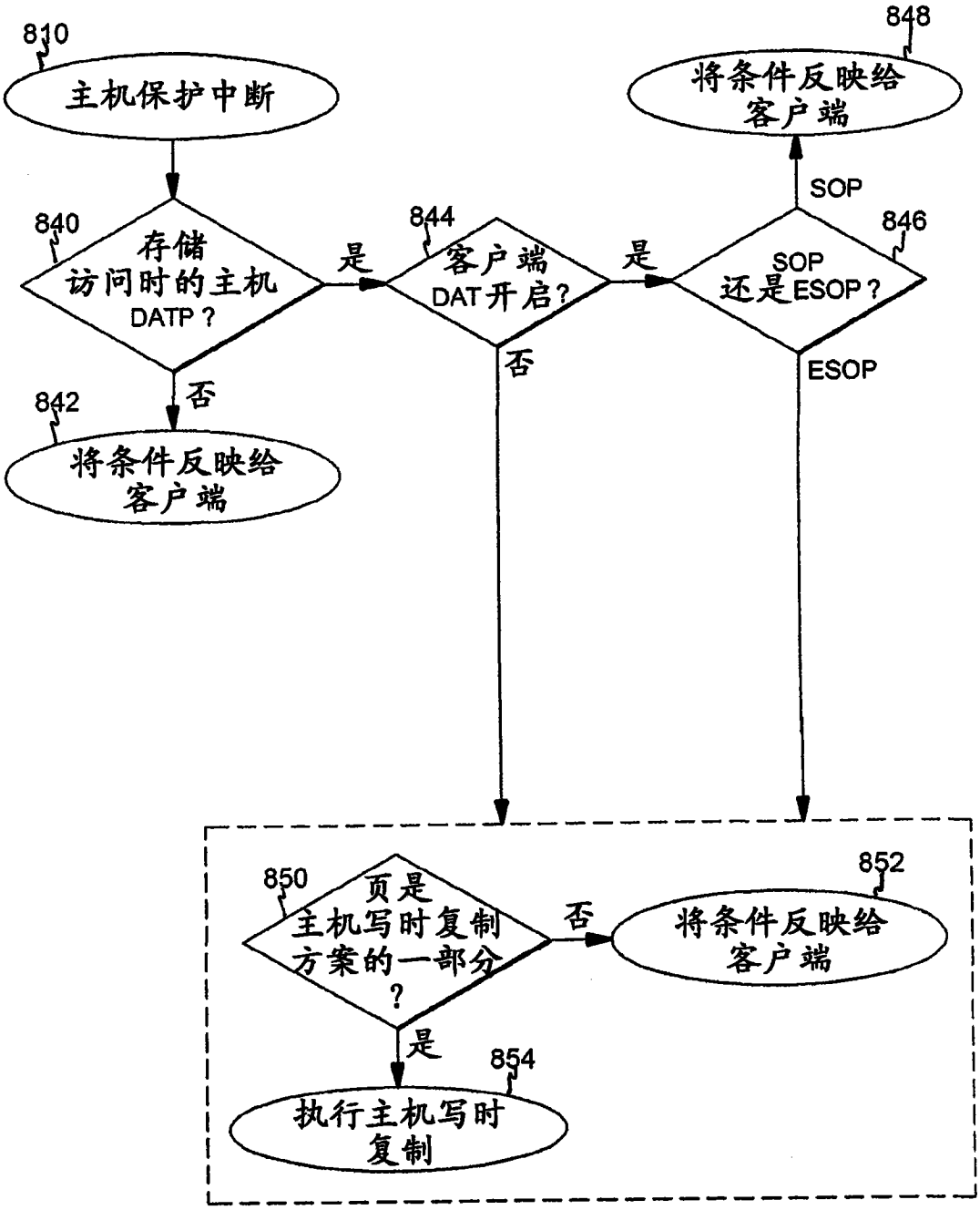

810
主机保护中断
840
存储
访问时的主机
DATP？
是
否
842
将条件反映给
客户端
844
客户端
DAT开启？
是
否
846
SOP
还是ESOP？
SOP
ESOP
848
将条件反映给
客户端
850
页是
主机写时复制
方案的一部分
？
否
852
将条件反映给
客户端
是
854
执行主机写时
复制

图8B

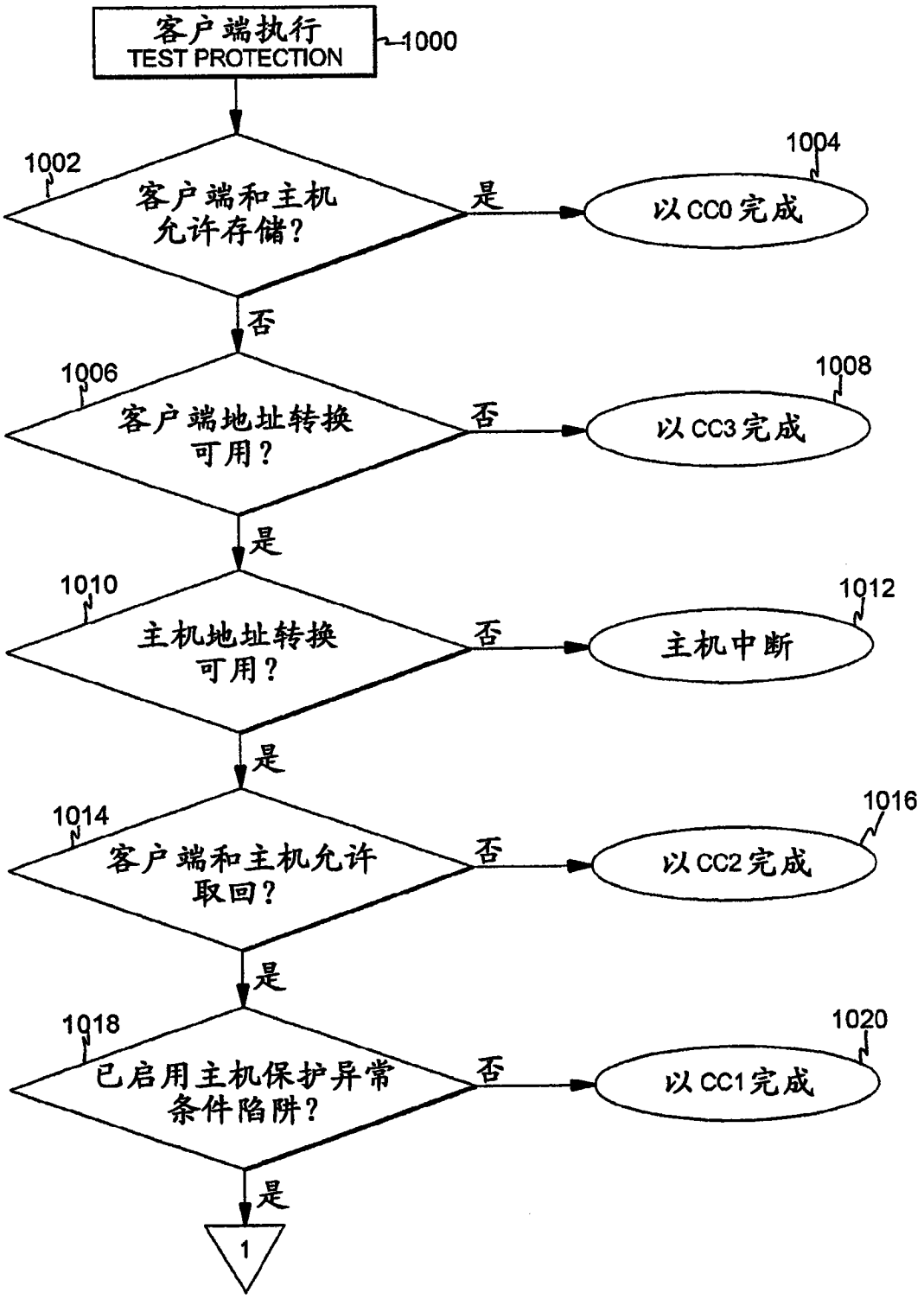

客户端执行
TEST PROTECTION
1000
1002
客户端和主机
允许存储？
是
1004
以CC0完成
否
1006
客户端地址转换
可用？
否
1008
以CC3完成
是
1010
主机地址转换
可用？
否
1012
主机中断
是
1014
客户端和主机允许
取回？
否
1016
以CC2完成
是
1018
已启用主机保护异常
条件陷阱？
否
1020
以CC1完成
是
1

图10A

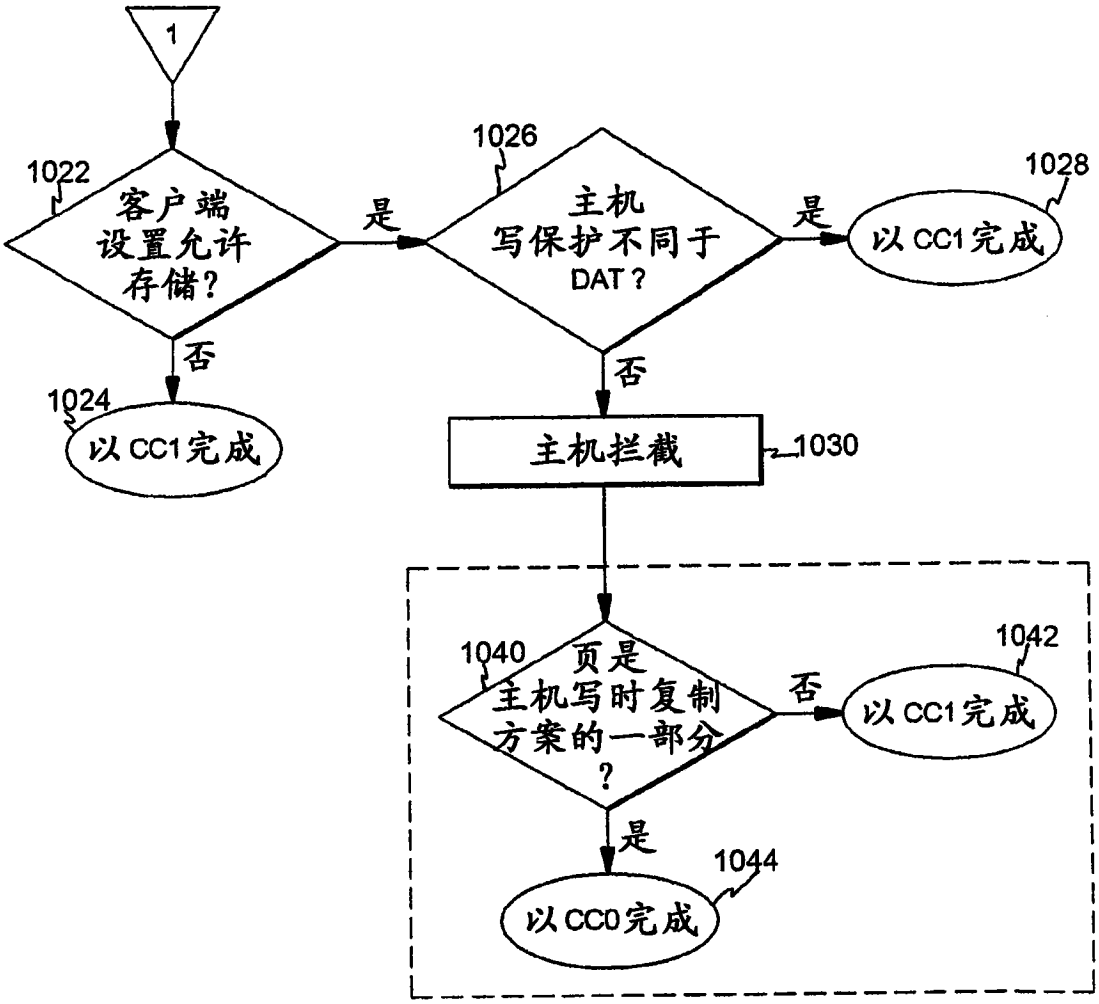

1
1022
客户端设置允许存储?
是
1026
主机写保护不同于DAT?
是
1028
以CC1完成
否
1024
以CC1完成
否
主机拦截
1030
1040
页是主机写时复制方案的一部分?
否
1042
以CC1完成
是
1044
以CC0完成

图10B

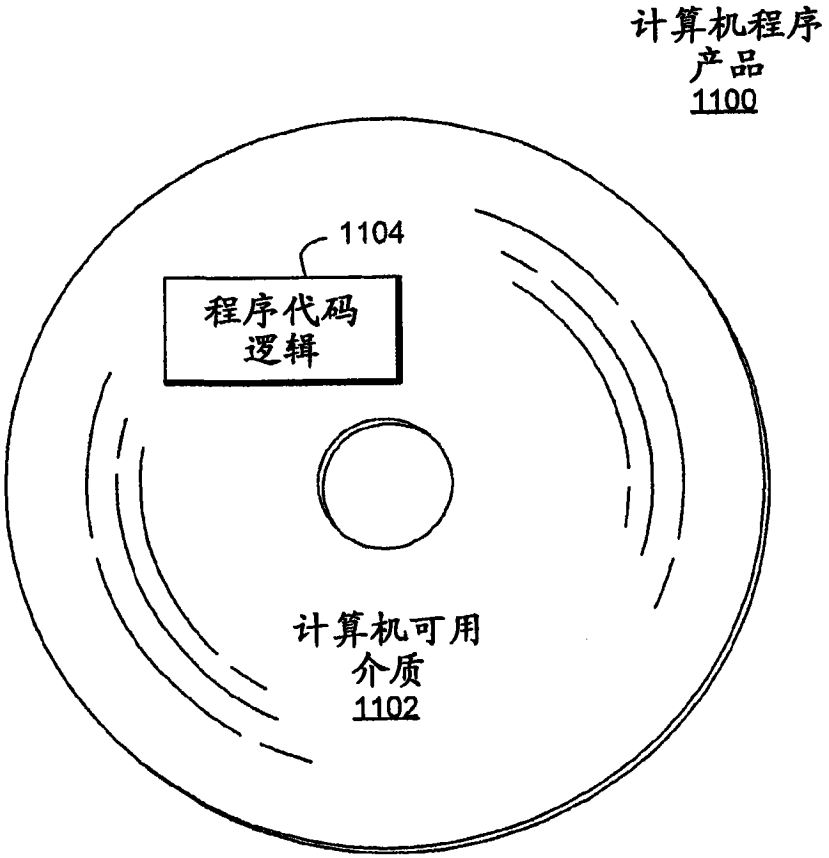


图11