

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06F 13/24 (2006.01)

G06F 9/455 (2006.01)



[12] 发明专利申请公布说明书

[21] 申请号 200610064133.0

[43] 公开日 2007 年 7 月 25 日

[11] 公开号 CN 101004726A

[22] 申请日 2006. 12. 30

[21] 申请号 200610064133.0

[30] 优先权

[32] 2005. 12. 30 [33] US [31] 11/323114

[71] 申请人 英特尔公司

地址 美国加利福尼亚州

[72] 发明人 G·奈格尔

R·S·马杜卡鲁穆库马纳

R·A·乌利希 U·施泰因伯格

S·谢恩伯格 S·穆思拉萨纳卢尔

S·M·本尼特 A·V·安德森

E·C·科塔-罗布尔斯

[74] 专利代理机构 中国专利代理(香港)有限公司

代理人 王小衡 刘杰

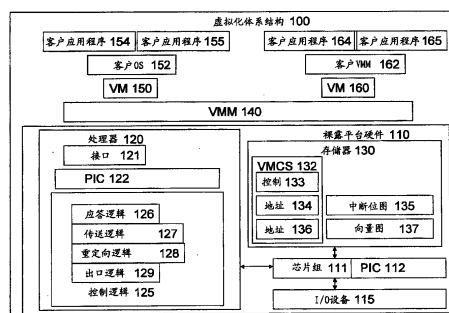
权利要求书 2 页 说明书 9 页 附图 3 页

[54] 发明名称

直接传送中断到虚拟处理器

[57] 摘要

公开了将中断传送到虚拟处理器的设备，方法和系统的实施例。在一个实施例中，设备包括接收中断请求的接口，传送逻辑，和出口逻辑。该传送逻辑基于该中断请求的属性确定该中断请求是否将被传送到虚拟处理器。如果该传送逻辑确定该中断请求不将被传送到该虚拟处理器，该出口逻辑电路传送控制到主机。



1. 一种设备包括:

接口, 该接口用于当客户在一个虚拟处理器上运行时接收一个中断请求;
传送逻辑, 用于基于该中断请求的属性确定该中断请求是否将被传送到该虚拟处理器; 以及

出口逻辑, 如果该传送逻辑确定该中断请求不将被传送到虚拟处理器, 该出口逻辑将设备的控制从客户传送到主机。

2. 如权利要求 1 所述的设备, 其中如果该传送逻辑确定该中断请求将被传送到虚拟处理器, 该传送逻辑还将该中断请求传送到该虚拟处理器。

3. 如权利要求 1 所述的设备, 其中该中断请求的属性包括物理中断向量, 传送模式和触发模式中的一个。

4. 如权利要求 1 所述的设备, 其中

该接口还用于接收一个物理中断向量; 以及

该传送逻辑基于该物理中断向量确定该中断请求是否将被传送到该虚拟处理器。

5. 如权利要求 4 所述的设备, 其中该传送逻辑基于中断传送数据结构中用于物理中断向量的入口确定该中断请求是否将被传送到该虚拟处理器。

6. 如权利要求 4 所述的设备, 还包括应答逻辑以应答中断请求以促使传送物理中断向量。

7. 如权利要求 4 所述的设备, 还包括重定向逻辑以将该物理中断向量翻译成虚拟中断向量。

8. 如权利要求 7 所述的设备, 其中重定向逻辑基于中断重定向数据结构中的入口将该物理中断向量翻译成虚拟中断向量。

9. 一种方法包括:

当客户在一个虚拟处理器上运行时接收一个中断请求;

基于该中断请求的属性确定该中断请求是否将被传送到该虚拟处理器; 以及

如果该中断请求不将被传送到该虚拟处理器, 将控制从客户传送到主机。

10. 如权利要求 9 所述的方法, 其中该中断请求的属性包括物理中断向量, 传送模式和触发模式中的一个。

11. 如权利要求9所述的方法,还包括接收一个物理中断向量。

12. 如权利要求11所述的方法,其中基于该物理中断向量确定该中断请求是否将被传送到该虚拟处理器。

13. 如权利要求12所述的方法,其中基于中断传送数据结构中用于物理中断向量的入口确定该中断请求是否将被传送到该虚拟处理器。

14. 如权利要求11所述的方法,还包括应答该中断请求以促使传送该物理中断向量。

15. 如权利要求11所述的方法,还包括将该物理中断向量翻译成虚拟中断向量。

16. 如权利要求15所述的方法,其中基于中断重定向数据结构中的入口将该物理中断向量翻译成虚拟中断向量。

17. 一种系统包括:

存储器,用于存储一数据结构以控制具有虚拟处理器的虚拟机;以及
物理处理器包括:

接口,用于接收一个中断请求;

传送逻辑,基于该中断请求的属性确定该中断请求是否将被传送到该虚拟处理器;以及

出口逻辑,如果该传送逻辑确定该中断请求不将被传送到该虚拟处理器,该出口逻辑将该处理器的控制从客户传送到主机。

18. 如权利要求17所述的系统,其中

该接口还接收一个物理中断向量;以及

该传送逻辑基于该物理中断向量确定该中断请求是否将被传送到该虚拟处理器。

19. 如权利要求18所述的系统,其中该传送逻辑基于数据结构中用于物理中断向量的入口确定该中断请求是否将被传送到该虚拟处理器。

20. 如权利要求18所述的系统,还包括重定向逻辑以将该物理中断向量翻译成虚拟中断向量。

21. 如权利要求20所述的系统,其中该重定向逻辑基于数据结构中的中断重定向图将该物理中断向量翻译成虚拟中断向量。

直接传送中断到虚拟处理器

技术领域

本发明的公开涉及信息处理领域，并且更特别地，涉及在虚拟化环境中处理中断的领域。

背景技术

通常，信息处理系统中虚拟化的概念允许在一个信息处理系统上运行一个或者多个操作系统（每个为一个“OS”）的多个实例，即使是每个 OS 被设计为在系统和其资源上具有完全的、直接的控制。虚拟化典型的通过使用软件（例如虚拟机监视器，或“VMM”）来实现以向每个 OS 展示一个具有虚拟资源的“虚拟机”（“VM”），其包括一个或多个虚拟处理器，这样该 OS 可以完全的并且直接的控制，而该 VMM 保持一个用于执行虚拟化策略的系统环境，诸如在 VMs（虚拟化环境）中共享和/或分配该物理资源。在 VM 上运行的每个 OS 和任何其他软件被称为一个“客户”或者“客户软件”，而“主机”或者“主机软件”是在虚拟化环境外运行的并且可以知道或不知道该虚拟化环境的软件，诸如 VMM。

信息处理系统中的物理处理器可以支持虚拟化，例如，通过支持一指令来进入虚拟化环境以在 VM 中的虚拟处理器（也就是说，在一个由 VMM 强加约束下的物理处理器）上运行客户。在虚拟化环境中，某些事件，操作和情况，例如外部中断或者试图访问特许寄存器或者资源，可以被“截取（intercepted）”，也就是说，导致该处理器退出虚拟化环境从而 VMM 可以操作，例如执行虚拟化策略。一个物理处理器也可以支持用于保持虚拟化环境的其他指令，并且可以包括存储器或者寄存器位，其指示或者控制物理处理器的虚拟化能力。

当客户在虚拟化环境内的虚拟处理器上运行时，支持虚拟化环境的物理处理器可以接受一个中断请求。典型的，该中断请求将被截取并且控制被传输到一个 VMM 来确定如何处理该中断。例如，一个中断服务例程（interrupt service routine）可以通过 VMM 被调用，或者该 VMM 可以创建一个虚拟中断并且将其插入到一个 VM 中来允许客户调用该中断服务例程。在很多情况下，被输入以调用中断服务请求的 VM 可以与被退出以允许该 VMM 来截取中断请求的 VM 相同。例如，

中断请求可以已经由被分配给被退出的该相同的 VM 的输入/输出 (“I/O”) 设备产生，或者中断请求可以为在相同 VM 中的两个虚拟处理器之间的处理器间 (inter-processor) 中断。

附图说明

本发明通过举例的形式阐明并且不局限于所伴随的附图中。

图 1 示出了一个虚拟化环境，其中本发明的一个实施例可以在该环境中操作。

图 2 以一个方法示出了本发明的一个实施例用于 VMM 初始化支持以直接传送中断到虚拟处理器。

图 3 以一个方法示出了本发明的一个实施例用于直接传送中断到虚拟处理器。

具体实施方式

用于直接传送中断到虚拟处理器的设备，方法和系统的实施例在下文描述。在描述中，大量的特定细节，例如元件和系统配置，可以被列出来提供对本发明一个更彻底的理解。然而，本领域的技术人员应当意识到，本发明可以在缺少这种特定细节的情况下被实施。另外，一些共知的结构，电路和类似的东西并未详细地示出，以避免本发明不必要的不清楚。

如果截取事件的频率被最小化，则虚拟化环境的性能可以被改进。本发明的实施例可以被用于在不需要通过 VMM 截取的情况下支持向虚拟处理器传送中断请求 (“直接传送”)。这样，可以改进其中全部中断请求均由 VMM 截取的虚拟化环境的性能。

图 1 示出了虚拟化结构 100，本发明的一个实施例可以在其中操作。在图 1 中，裸露平台硬件 110 可以是任何的能够执行任何 OS 或者 VMM 软件的数据处理设备。例如，裸露平台硬件可以是个人计算机，大型计算机，便携计算机，手持设备，机顶盒，服务器，或者任何其他计算系统。裸露平台硬件 110 包括处理器 120，存储器 130，输入/输出 (“I/O”) 设备 115，和芯片组 111。

处理器 120 可以为任何类型的处理器，包括通用微处理器，例如 Pentium® 处理器系列，Itanium® 处理器系列，或者其他来自 Inter 公司的处理器系列中的处理

器, 或者来自另外公司的另外的处理器, 或者数字信号处理器或者微控制器。虽然附图 1 只示出一个这种处理器 120, 但是裸露平台硬件 110 可以包括任何数量的处理器, 包括任何数量的多核处理器, 每个处理器具有任何数量的执行核, 和任何数量的支持多线程的处理器, 每个处理器具有任何数量的线程。

存储器 130 可以是静态或者动态随机访问存储器, 基于半导体的只读或者闪存存储器, 磁或者光盘存储器, 处理器 120 可读的任何其他类型的介质, 或者任何这种介质的组合。I/O 设备 115 可以表示任何数量的外围设备或者 I/O 设备, 例如监视器, 键盘, 鼠标, 打印机, 网络接口, 信息存储设备等等。芯片组 111 可以为包括任何数量的执行任何数量任务例如系统逻辑、总线控制、总线接口、总线桥接、存储控制、外围设备控制、外围设备功能、系统配置等等的组件。

处理器 120, 存储器 130, I/O 设备 115, 和芯片组 111 可以根据任何已知的方法诸如通过一个或者多个总线, 点到点, 或者其他有线或者无线连接直接或间接地彼此耦合或通信。裸露平台硬件 110 也可以包括任何数量的附加设备或者连接。

除裸露平台硬件 100 外, 图 1 示出了 VMM 140, VM 150 和 160, 以及客户 152, 154, 155, 162, 164 和 165。

VMM 140 可以是任意的安装在裸露平台硬件 110 上的或者可访问裸露平台硬件 11 的软件, 固件, 或者硬件主机, 以将 VMs, 也就是说裸露平台硬件 110 的抽象 (abstractions) 呈现给客户, 或者创建 VMs, 管理 VMs, 并且在虚拟化环境 100 中执行虚拟化策略。在其他的实施例中, 主机可以为任何 VMM、管理程序 (hypervisor)、OS、或者其它能够控制裸露平台硬件 110 的软件、固件、或者硬件。客户可以是任何 OS、任何 VMM, 包括另外的 VMM 140 的例子, 任何管理程序, 或者任何应用程序或者其它软件。

每个客户期望依照处理器的体系结构和出现在 VM 中的平台访问物理资源, 例如裸露平台硬件 110 的处理器和平台寄存器、存储器以及输入/输出设备。图 1 示出两个 VMs, 150 和 160, 其具有客户 OS 152 和安装在 VM 150 上的客户应用程序 154 和 155 以及客户 OS 162 和安装在 VM 160 上的客户应用程序 164 和 165。虽然图 1 仅示出两个 VMs 和每个 VM 的两个应用程序, 在本发明的范围内可以创建任何数量的 VMs 并且可以在每个 VM 上运行任何数量的应用程序。

可以由客户访问的资源可以被分类为“特许 (privileged)”或“非特许

(non-privileged)”资源。对于一个特许资源，VMM 140 促进了 (facilitating) 客户期望的功能性同时保留在资源上的最终控制。非特许资源无需被 VMM 140 控制并且可以被客户直接访问。

并且，每个客户 OS 期望处理不同的事件，诸如异常（例如页错误，和一般保护错误），中断（例如硬件中断和软件中断），以及平台事件（例如初始化和系统管理中断）。在这里这些异常，中断和平台事件被统称和单独地称作“事件”。这些事件中的一些为“特许的”，因为他们必须通过 VMM 140 被处理以确保 VMs 150 和 160 的适当操作，来自客户的 VMM 140 的保护以及客户彼此间的保护。

在任何给定的时间，处理器 120 可以执行来 VMM 140 或者任意客户的指令，这样 VMM 140 或者客户可以在处理器 120 上运行或者受处理器 120 控制。当一个特许事件发生或者客户试图访问一个特许资源时，控制可以从客户传输到 VMM 140。控制从客户到 VMM 140 的传送在这里被称作一个“VM 出口”。在处理事件或者适当地促进对资源的访问后，VMM 140 可以返回控制到客户。控制从 VMM 140 到客户的传送在这里被称作“VM 入口”。

在图 1 的实施例中，处理器 120 依照存储在虚拟机控制结构（“VMCS”）132 内的数据控制 VMs 150 和 160 的操作。VMCS 132 为这样一种结构，该结构可以包含一个客户或者多个客户的状态、VMM 140 的状态、指示 VMM 140 如何控制一个客户或者多个客户操作的执行控制信息，关于 VM 出口和 VM 入口的信息，任何其他这种信息。处理器 120 从 VMCS 132 读取信息来确定 VM 的执行环境并且约束它的行为。在这个实施例中，VMCS 132 被存储在存储器 130 中。在一些实施例中，多个 VMCSs 被用于支持多个 VMs。虽然图 1 示出 VMCS 132 存储在存储器 130 中，本发明无需在存储器中存储 VMCS。

处理器 120 包括中断控制器 122 来接收，产生，区分优先次序，传送，保持挂起，或者其它控制或管理中中断请求。例如，中断控制器 122 可以是一个依照 Pentium®处理器系列的体系结构的本地高级可编程中断控制器（“APIC”）。芯片组 111 除了与中断控制器 112 连接或者替代中断控制器 112 外，也可以包括中断控制器 112 来接收，产生，区分优先次序，传送，保持挂起，或者其它控制或管理中中断请求。例如，中断控制器 112 可以为一个 I/O APIC。处理器 120 和/或芯片组 111 可以包括任何其他中断控制器，和/或任何其他处理器，芯片组，或者图 1 未示出的包括中断控制器的元件，和/或可以依照任何其他途径控制或管理中中断请

求。

处理器 120 还包括接口 121，该接口 121 可以为一个总线单元或者任何其他单元，端口，或者接口以允许处理器 120 通过任何类型的总线，点到点，或者其他连接，直接地或者通过诸如芯片组 111 的任何其它元件接收中断请求和中断向量。接口 121 可以为内部接口，例如，从本地 APIC 接收中断请求，和/或外部接口，例如从外部资源接收中断请求。

另外，处理器 120 包括控制逻辑 125 来支持虚拟化，包括传送中断到虚拟处理器。控制逻辑 125 可以是微代码，可编程逻辑，硬编码逻辑，或者任何其他形式的处理器 120 中的控制逻辑。在其他实施例中，控制逻辑 125 可以以任何形式的硬件，软件或固件诸如处理器内或任何处理器可访问的元件或可读取的介质诸如存储器内的处理器抽象层来实现。

控制逻辑 125 促使处理器 120 执行本发明的方法实施例，例如下面图 2 和 3 中描述的方法实施例，例如，通过使处理器 120 包括一个或者多个微操作的执行，例如，在它响应于虚拟化指令中或者来自主机或客户的其他指令中支持虚拟化。

控制逻辑 125 包括中断应答逻辑 126，中断传送逻辑 127，中断重定向逻辑 128，和出口逻辑 129。中断应答逻辑 126 用于应答中断请求，其中某些实施例可能引起一个中断向量被传送到处理器 120。中断传送逻辑 127 是用来确定中断请求是否将被传送到虚拟处理器。中断重定向逻辑 128 是用来重新定向中断以传送到虚拟处理器而不是物理处理器 120，例如通过把物理中断向量翻译成虚拟中断向量。出口逻辑 129 是用来做准备并且如果中断请求不将被传送到虚拟处理器则使得 VM 退出。这些逻辑单元中的每个也可以执行其它功能，包括那些所描述的由另外的逻辑单元执行的功能，并且任何或者全部的这些逻辑单元可以被集成到一个单独的逻辑单元中。

VMCS 132 可以包括字段，控制位，或者其他数据结构来支持虚拟化，包括传送中断到虚拟处理器。这些数据结构可以被检查或否则由控制逻辑 125 参考来确定如何管理一个 VM 环境。例如，象下面描述的，中断分配控制位 133 可以被设置成允许向虚拟处理器直接传送中断请求。在这个实施例的描述中，控制位被置位以允许或者引起一个期望的效果，这里置位意思是写入一个逻辑 1 到该位，但是在本发明的范围内可以使用任何逻辑约定法或者命名。

同样在 VMCS 132 中，地址字段 134 可以被用来存储一个存储器位置的地

址, 在这个存储器位置上可以存储指示中断是否将被传送到虚拟处理器的数据结构。例如, 该地址可以是一个 256 位数组中的 64 位地址, 其包括一个用于直到 256 个物理中断向量的每个的配置位。类似地, 地址字段 136 可以用来存储一个存储位置的地址, 在该存储位置上可以存储提供虚拟中断向量的数据结构。例如, 该地址可以是一个 256 字节数组的 64 位地址, 其包括一个用于直到 256 个物理中断向量的每个的 1 字节虚拟中断向量。在一个可替代实施例中, VMCS 132 可以包括直接存储这种指示或者虚拟中断向量的数据结构。

图 2 和 3 阐明了本发明的方法实施例。虽然方法实施例不局限于这个方面, 但参考图 1 的虚拟化环境 100 来描述图 2 和 3 的方法实施例。

图 2 以一个方法示出了本发明的一个实施例用于 VMM 初始化支持以直接传送中断到虚拟处理器。

在图 2 的框 210 中, 图 1 的 VMM 140 创建了一个用于 VM 的 VMCS (例如, VMCS 132)。在框 212 到 217 中, VMM 140 配置 VMCS 132 来实现传送中断到虚拟处理器的支持。在框 212 中, 象下文描述的, 中断分配控制位 133 被置位以允许直接传送中断到虚拟处理器。

在框 214 中, 用于存储数据结构的存储器位置的地址被写入到地址字段 134, 该数据结构指示中断是否将被传送到虚拟处理器。确定数量 (例如 256 个) 潜在中断的每个可以通过一个物理中断向量被识别。该数据结构 (例如, 中断位图 135) 可以包括用于每个物理中断向量的入口。每个入口可以包括一个配置位来指示相应于每个该物理中断向量的物理中断是否将被传送到虚拟处理器。在框 215 中, 例如, 通过存储物理中断向量和为每个物理中断向量存储一个期望的配置位来初始化数据结构。

在框 216 中, 用于存储数据结构来将物理中断向量映射到虚拟中断向量的存储位置的地址被写入到地址字段 136。该数据结构 (例如向量重定向图 137) 可以包括用于每个物理中断向量的入口。对于每个将被传送到虚拟处理器的中断, 每个入口可以包括一个相应于物理中断向量的虚拟中断向量。这种再映射 (remapping) 可以被用于例如用来说明 (account for) 再映射系统存储器 130, 或者系统存储器 130 的一部分到分配给包括虚拟处理器的 VM 的存储器。在框 217 中, 例如, 通过存储物理中断向量并且为每个物理中断向量存储一个期望的虚拟中断向量来初始化该数据结构。

图3 以一个方法示出了本发明的一个实施例用于直接传送中断到虚拟处理器。

在图3的框310中，一个VM入口被执行并且控制被传输到一个客户。在框320中，该客户开始或者继续在VM中的虚拟处理器上运行。在框330中，接口121接收一个中断请求。该中断请求可以来自一个中断控制器，例如中断控制器122或112，或任何其他设备。在一些实施例中，该中断可以是一个由另外的物理处理器产生的处理器间中断，或者由一个客户向它自己的虚拟处理器产生。

在框340中，控制逻辑125检查中断分配控制位133来确定到虚拟处理器的中断的直接传送是否被允许。如果到虚拟处理器的中断的直接传送不被允许，那么，在框390中，出口逻辑129使得VM出口出现来允许该VMM处理该中断。在一些实施例中，出口逻辑129通过查阅该VMCS中的一个或多个控制位例如中断退出控制位来确定在这种情况下是否需要VM出口。在一些实施例中，出口逻辑129应答该中断请求并且提取(fetch)该中断向量。在一些实施例中，由于硬件中断可以通过VMCS中的一个或者多个控制位例如，在出口控制位上的应答中断来控制作为VM出口的一部分的中断应答。

然而，如果到虚拟处理器的中断的直接传送是允许的，那么，在框350中，中断应答逻辑126应答该中断请求，例如，通过发送一个中断应答信息。在处理器包括一个本地APIC的实施例中，该中断请求在本地APIC被应答。在框352中，处理器120接收物理中断向量。在处理器包括本地APIC的实施例中，中断请求的应答也可以使得该中断请求寄存器(“IRR”)，服务中寄存器(“ISR”)，和处理器优先寄存器(“PPR”)被更新。

在框306中，中断传送逻辑127确定该中断是否被传送到该虚拟处理器，例如，通过检查中断位图135。该确定也可以或者代替地基于其他中断请求的属性，诸如传送模式(例如，固定的或者用户定义的，SMI(系统管理中断)，NMI(不可掩蔽的中断)，INT(软复位)，或外部的)和触发模式(例如，边沿或者电平)。如果该中断不将被传送到虚拟处理器，则，在框390中，出口逻辑129使得VM出口出现来允许该VMM处理该中断。该物理中断向量可以通过将其存储在VMCS中的一个VM出口字段中被提供给VMM。

然而，若该中断将被传送到虚拟处理器，则，在框362中，中断重定向逻辑128重定向中断以传送到虚拟处理器而不是物理处理器，例如，通过把物理中断

向量翻译成虚拟中断向量。在这个实施例中，可以通过在中断向量重定向图 137 中查找物理中断向量来发现相应的虚拟中断向量的方式来执行翻译。

在一个本地 APIC 的实施例中，框 362 也可以包括另外的操作来适当地管理中断请求。例如，控制逻辑 125 可以使得本地 APIC 中的中断结束（“EOI”）寄存器被清除，这样该 ISR 和 PPR 被更新。

在框 364 中，该重定向中断被传送到虚拟处理器，例如，通过使用该虚拟中断向量作为到与该虚拟处理器相关的中断描述符表（“IDT”）的索引来寻找一个中断处理器的入口点。在一些实施例中，依赖于该中断的优先权，该重定向中断可以优先于到虚拟处理器的传送被保持挂起。优先化可以是在将被传送到虚拟存储器的中断之间，或者可也是包括将被传送到 VMM 的中断。

在具有共享的中断请求线路的实施例中，可以使用多种技术的任何一种来防止与指向虚拟处理器的第一中断请求相关的第一中断处理器自动地链接到与不指向虚拟处理器的第二中断请求相关的第二中断处理器。在一个实施例中，该本地 APIC 的 EQI 寄存器可以使用一个被不广播到 I/O APICs 的特定消息被清除。在另外一个实施例中，可以基于触发模式（例如，边沿或者电平）或其他指示共享能力（shareability）的属性来确定中断请求是否将被传送到虚拟处理器。

并且，在一些实施例中，来自调用的中断处理器的 EOI 信息由于直接传送的中断可以被截取，而在其他实施例中，处理器可以包括控制逻辑以在没有 VM 出口的情况下对其进行处理。

在本发明的范围内，图 2 和 3 阐明的方法在具有省略的示出的框，在具有增加另外的框，或者在具有重新排序的、省略的或者附加框的组合的情况下以不同的顺序被执行。例如，VMCS 可以以任意的顺序配置 VMCS，例如，框 212 到 217 可以以任意顺序重新排列。

根据本发明实施例设计的处理器 120，或者任意其他元件或者元件的一部分可以在从创建到仿真到制造的不同阶段被设计。表示设计的数据可以以多种方式表示设计。首先，由于在仿真中是有用的，可以使用硬件描述语言或者另外的功能性描述语言表示硬件。另外或者可选的，具有逻辑电路和/或晶体管门电路的电路级模型可以在设计过程的某些阶段被制造。并且，大多数设计，在某些阶段，达到一个它们可以以表示各种设备的物理布局的数据被模拟的水平。在使用常规的半导体制造技术的情况下，该表示设备布局模型的数据可以是规定在用于制造

集成电路的掩模的不同的掩模层上存在或不存在各种特征的数据。

在设计中的任意表示中，数据可以被存储在任意形式的可读机械介质中。光或者电波调制或者另外产生来传输此类信息、存储器、或者磁或者光存储介质，诸如盘，可以是可读机械介质。这些介质中的任意一个可以“携带(carry)”或“指示(indicate)”该设计，或者使用在本发明的实施例中的其它信息，诸如在误差恢复程序中的指令。当指示或者携带信息的电载波被传输时，进行新的拷贝到执行复制、缓冲、或者重传输电信号的程度。这样，通信提供者的活动或者网络提供者的活动可以构成制作包含本发明的技术的产品例如载波的拷贝。

因此，已经描述了用于直接传送中断到虚拟处理器的设备，方法和系统。虽然已经描述了特定的实施例，并且在附图中示出，但是应当理解这些实施例只是示意性的并且没有对宽的发明进行限制，并且本发明不局限于所示出和描述的特定结构和安排，因为那些本领域普通技术人员来说在研究本发明的基础上可以想出各种其它的修改。在本技术领域，发展是快速的并且不容易预见更大的进步，在不背离本发明公开的主旨和所伴随的权利要求的范围内，通过使用技术上的进步的可以便于在结构和细节对所公开的实施例容易地进行修改。

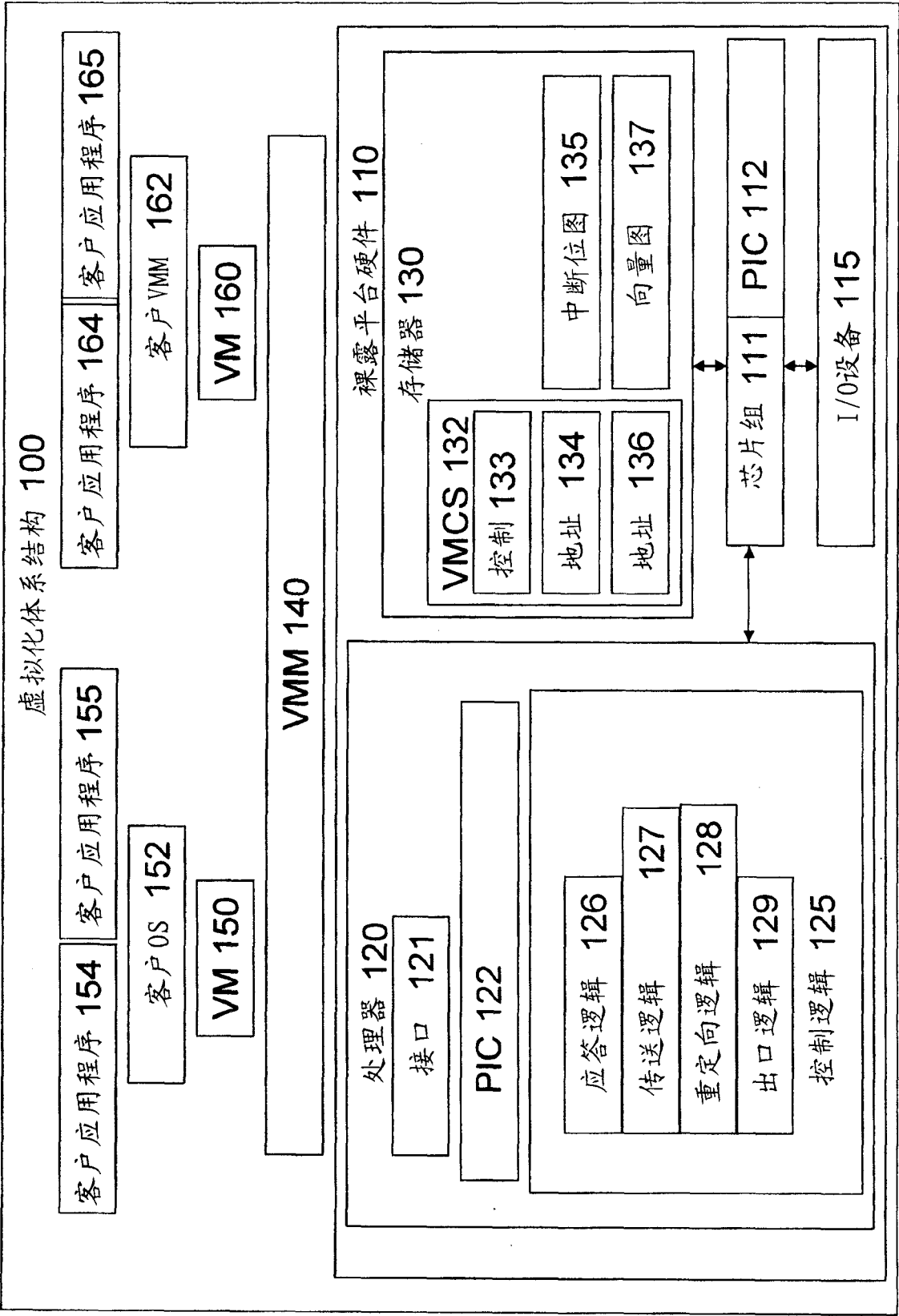


图 1

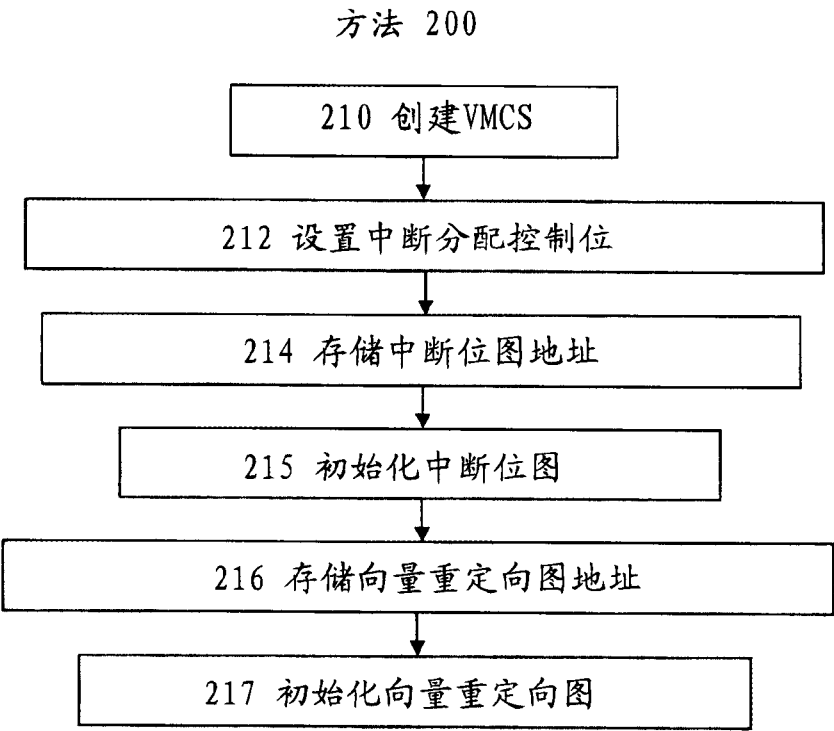


图 2

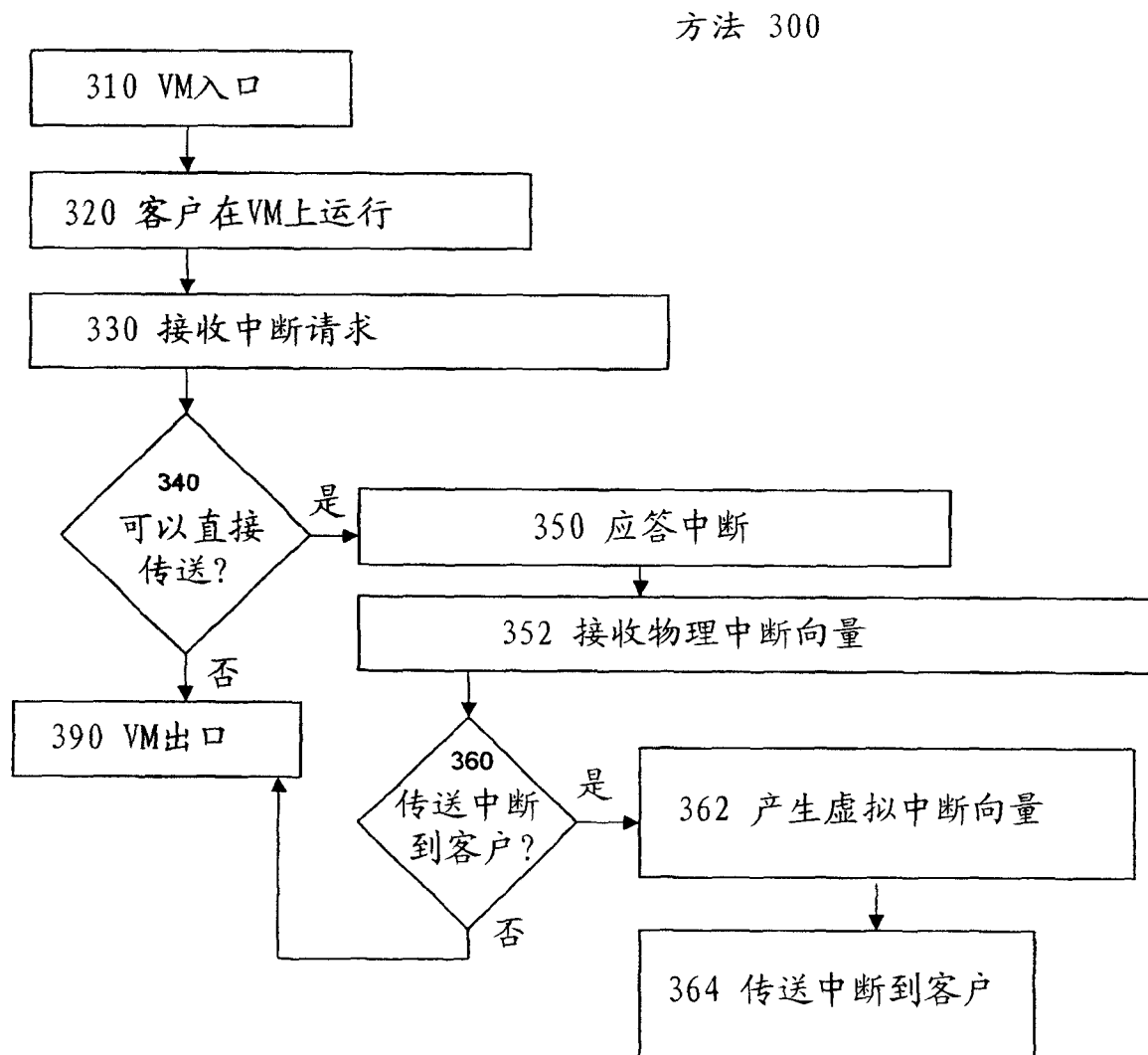


图 3