(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 107480075 B
(45)授权公告日 2020.10.27

(21)申请号 201611191432.0

(22)申请日 2016.12.21

(65)同一申请的已公布的文献号
申请公布号 CN 107480075 A

(43)申请公布日 2017.12.15

(30)优先权数据
62/347,495 2016.06.08 US
15/293,627 2016.10.14 US

(73)专利权人 谷歌有限责任公司
地址 美国加利福尼亚州

(72)发明人 埃里克·诺瑟普
本杰明·查尔斯·塞利布林

(74)专利代理机构 中原信达知识产权代理有限责任公司 11219
代理人 高伟 周亚荣

(51)Int.Cl.
G06F 12/1027(2016.01)
G06F 9/455(2006.01)

(56)对比文件
CN 1728113 A,2006.02.01
CN 104919417 A,2015.09.16
US 2006026359 A1,2006.02.02
US 2014189285 A1,2014.07.03
CN 101398768 A,2009.04.01
CN 102262557 A,2011.11.30

审查员 白利敏

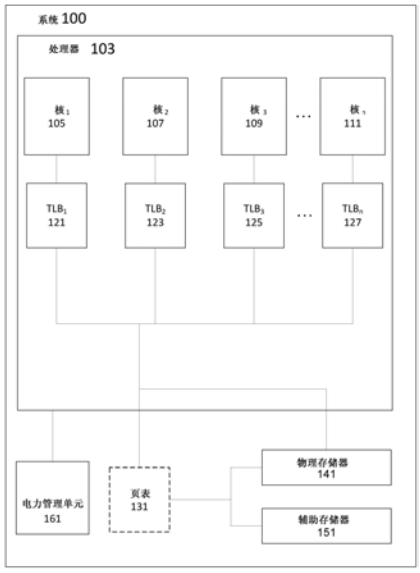
权利要求书3页 说明书8页 附图5页

(54)发明名称

低开销的转换后备缓冲器下拉

(57)摘要

公开了低开销的转换后备缓冲器下拉。本公开的各方面涉及在硬件内引导和跟踪转换后备缓冲器(TLB)下拉。包括一个或多个处理器核的一个或多个处理器可以确定在处理器核上执行的进程导致一个或多个虚拟存储器页面与一个或多个先前关联的物理存储器地址解除关联。正在执行引起解除关联的处理的处理器核可以生成TLB下拉请求。处理器核可以将TLB下拉请求发送到其它核。TLB下拉请求可以包括标识信息、指示需要从其它核的相应TLB刷新的解除关联的虚拟存储器页面的下拉地址以及指示其它核可以在何处确认TLB下拉请求的完成的通知地址。



CN 107480075 B

1.一种用于引导和跟踪硬件内的转换后备缓冲器(TLB)下拉的方法,包括:

由一个或多个处理器确定在第一处理器核上执行的进程使一个或多个虚拟存储器页面与一个或多个先前关联的物理存储器地址解除关联,每个处理器包括一个或多个处理器核;

由所述第一处理器核产生转换后备缓冲器下拉请求;和

由所述第一处理器核将所述转换后备缓冲器下拉请求传送到所述一个或多个处理器核中的其它处理器核,所述转换后备缓冲器下拉请求包括:

下拉地址,所述下拉地址指示待从所述其它处理器核的相应转换后备缓冲器刷新的所解除关联的一个或多个虚拟存储器页面;

通知地址,所述通知地址指示所述其它处理器核能够在何处确认所述转换后备缓冲器下拉请求的完成;以及

标识信息,其中,所述标识信息包括用于部件的标识符,所述标识信息包含一个或多个高级可编程中断控制器ID、一个或多个虚拟处理器ID和一个或多个进程上下文标识符,其中,所述部件包括以下中的一个或多个:虚拟机、所述虚拟机内的虚拟计算机处理器、处理器核和在虚拟计算机处理器或处理器核内执行的进程。

2.根据权利要求1所述的方法,还包括:在所述其它处理器核完成所述转换后备缓冲器下拉请求时,从所述其它处理器核接收确认,其中,在所述通知地址处接收所述确认。

3.根据权利要求1所述的方法,其中,所述一个或多个处理器执行一个或多个虚拟机,其中所述一个或多个虚拟机包括一个或多个虚拟计算机处理器(VCPU),并且所述进程能够是在所述一个或多个虚拟计算机处理器中执行的第一进程。

4.根据权利要求1所述的方法,还包括:

由电力管理单元确定接收到所述转换后备缓冲器下拉请求的第二处理器核是否处于低电力状态;以及

在所述第二处理器核处于低电力状态的情况下,由所述电力管理单元确认所述转换后备缓冲器下拉请求。

5.根据权利要求1所述的方法,还包括:

通过高级可编程中断控制器(APIC)确定接收到所述转换后备缓冲器下拉请求的第二处理器核是否处于低电力状态,并且在所述第二核处于低电力状态的情况下,通过所述高级可编程中断控制器确定所述转换后备缓冲器下拉请求。

6.根据权利要求1所述的方法,还包括:

跟踪从由所述标识信息标识的所述一个或多个部件接收的确认的数量,直到所有的所述一个或多个部件确认所述转换后备缓冲器下拉请求;以及

在从由所述标识信息标识的所有的所述一个或多个部件接收到确认时,由所述第一处理器核完成所述转换后备缓冲器下拉请求。

7.一种用于在硬件内引导和跟踪转换后备缓冲器(TLB)下拉的系统,所述系统包括:

一个或多个处理器,每个处理器包括一个或多个处理器核,其中所述一个或多个处理器被配置为:

确定在第一处理器核上执行的进程导致一个或多个虚拟存储器页面与一个或多个先前关联的物理存储器地址解除关联;

生成转换后备缓冲器下拉请求;和

将所述转换后备缓冲器下拉请求传送到所述一个或多个处理器核中的其它处理器核,所述转换后备缓冲器下拉请求包括:

下拉地址,所述下拉地址指示待从所述其它处理器核的相应转换后备缓冲器刷新的所解除关联的一个或多个虚拟存储器页面;

通知地址,所述通知地址指示所述其它处理器核能够在何处确认所述转换后备缓冲器下拉请求的完成;以及

标识信息,其中,所述标识信息包括用于部件的标识符,所述标识信息包含一个或多个高级可编程中断控制器ID、一个或多个虚拟处理器ID和一个或多个进程上下文标识符,其中,所述部件包括以下中的一个或多个:虚拟机、所述虚拟机内的虚拟计算机处理器、处理器核和在虚拟计算机处理器或处理器核内执行的进程。

8.根据权利要求7所述的系统,其中,所述一个或多个处理器被配置成在其完成所述转换后备缓冲器下拉请求时接收来自所述其它处理器核的确认,其中在所述通知地址处接收所述确认。

9.根据权利要求7所述的系统,其中,所述一个或多个处理器被配置为执行一个或多个虚拟机,其中所述虚拟机包括一个或多个虚拟计算机处理器(VCPU),并且所述进程是在所述一个或更多虚拟计算机处理器中执行的第一进程。

10.根据权利要求7所述的系统,还包括电力管理单元,其中所述电力管理单元被配置为确定接收到所述转换后备缓冲器下拉请求的第二处理器核是否处于低电力状态,并且在所述第二处理器核处于低电力状态时,由所述电力管理单元确认所述转换后备缓冲器下拉请求。

11.根据权利要求7所述的系统,还包括高级可编程中断控制器(APIC),其中所述高级可编程中断控制器被配置为确定接收到所述转换后备缓冲器下拉请求的第二处理器核是否处于低电力状态,并且在所述第二处理器核处于低电力状态时,由所述高级可编程中断控制器确认所述转换后备缓冲器下拉请求。

12.根据权利要求7所述的系统,其中,所述一个或多个处理器被配置为跟踪从由所述标识信息标识的部件接收的确认的数量,直到所有的所述一个或多个部件确认所述转换后备缓冲器下拉请求为止;以及

在从由所述标识信息标识的所有的所述一个或多个部件接收到确认时完成所述转换后备缓冲器下拉请求。

13.一种非暂时性计算机可读介质,在所述非暂时性计算机可读介质上存储有指令,所述指令在被一个或多个处理器执行时使得所述一个或多个处理器执行用于在硬件内引导和跟踪转换后备缓冲器(TLB)下拉的方法,每个处理器包括一个或多个处理器核,所述方法包括:

确定在第一处理器核上执行的进程导致一个或多个虚拟存储器页面与一个或多个先前关联的物理存储器地址解除关联;

生成转换后备缓冲器下拉请求;

将所述转换后备缓冲器下拉请求传送到所述一个或多个处理器核中的其它处理器核,所述转换后备缓冲器下拉请求包括:

下拉地址，所述下拉地址指示待从所述其它处理器核的相应转换后备缓冲器刷新的所解除关联的一个或多个虚拟存储器页面；

通知地址，所述通知地址指示所述其它处理器核能够在何处确认所述转换后备缓冲器下拉请求的完成；以及

标识信息，其中，所述标识信息包括用于部件的标识符，所述标识信息包含一个或多个高级可编程中断控制器ID、一个或多个虚拟处理器ID和一个或多个进程上下文标识符，其中，所述部件包括以下中的一个或多个：虚拟机、所述虚拟机内的虚拟计算机处理器、处理器核和在虚拟计算机处理器或处理器核内执行的进程。

14.根据权利要求13所述的非暂时性计算机可读介质，其中，所述方法还包括在所述其它处理器核完成所述转换后备缓冲器下拉请求时接收来自所述其它处理器核的确认，其中在所述通知地址处接收所述确认。

15.根据权利要求13所述的非暂时性计算机可读介质，其中，所述方法还包括：在第二处理器核处于低电力状态的情况下，从电力管理单元接收所述转换后备缓冲器下拉请求的确认。

16.根据权利要求13所述的非暂时性计算机可读介质，其中，所述方法还包括：在第二处理器核处于低电力状态的情况下，从高级可编程中断控制器(APIC)接收所述转换后备缓冲器下拉请求的确认。

17.根据权利要求13所述的非暂时性计算机可读介质，其中，所述方法还包括执行一个或多个虚拟机，其中所述一个或多个虚拟机包括一个或多个虚拟计算机处理器(VCPU)，并且所述进程是在一个或多个虚拟计算机处理器中执行的第一进程。

18.根据权利要求13所述的非暂时性计算机可读介质，其中，所述方法还包括跟踪从由所述标识信息标识的部件接收的确认的数量，直到所有的所标识的部件确认所述转换后备缓冲器下拉请求为止；以及

在由所述标识信息标识的所有部件接收到确认时完成所述转换后备缓冲器下拉请求。

低开销的转换后备缓冲器下拉

[0001] 相关申请的交叉引用

[0002] 本申请要求于2016年6月8日提交的美国临时专利申请No.62/347,495和2016年10月14日提交的美国专利申请No.15/293,627的申请日的权益,其公开内容通过引用并入本文。

背景技术

[0003] 多核处理器和虚拟处理器的激增导致虚拟存储器的使用的增加。虚拟存储器向在系统的处理器上操作的每个进程提供存储器的连续部分的错觉,但是每个进程使用的实际物理存储器可以散布在系统的物理存储器上。在这点上,虚拟存储器通常被分割成页面,其中每个页面被映射到系统的物理存储器的位置。页表可以用于将一段数据的虚拟存储器页面映射到存储数据的对应物理地址。为了提高将虚拟存储器页面转换为相应物理地址的速度,处理器的每个核可以实现存储虚拟存储器页面到物理存储器地址的最近转换的转换后备缓冲器(translation lookaside buffer,TLB)。

[0004] 当在页表中修改虚拟存储器页面映射时,或者当管理程序从虚拟机的虚拟存储器去映射或以其它方式修改客户页面时,需要相应地更新每个处理器核的转换后备缓冲器(TLB)。在一些情况下,这是通过发送称为TLB下拉(shootdown)中断的中断来实现的,该中断指示每个目标处理器核查看未映射的虚拟存储器页面条目的软件定义的列表,并且从它们各自的TLB中移除这些条目。目标处理器核可以从它们各自的TLB表中移除未映射的条目,并且向TLB下拉的发起处理器发信号通知它们完成以上步骤。在发起处理器上的OS软件必须等到所有响应都返回,然后可以重新开始进一步处理。类似地,接收处理器核必须在恢复进一步处理之前完成TLB下拉请求。

[0005] 在虚拟化环境中,发送和接收中断(例如TLB下拉请求)可能是耗时的,因为物理处理器和虚拟处理器之间的通信需要管理程序的介入。此外,虚拟处理器可以离线(例如,由管理程序进行预排序或者物理CPU可以被停止),从而导致来自那些虚拟处理器的TLB下拉确认被延迟。在具有大量虚拟处理器的系统中,这些延迟可能增加,有时是超线性地增加,导致显著的性能降低。

发明内容

[0006] 本公开内的各种实施例基本涉及处理TLB下拉请求。一个方面包括一种用于在硬件内引导和跟踪TLB下拉的方法。在这点上,包括一个或多个处理器核的一个或多个处理器可以确定在处理器核上执行的进程导致一个或多个虚拟存储器页面与一个或多个先前关联的物理存储器地址解除关联。正在执行引起解除关联的进程的处理器核可以生成TLB下拉请求。处理器核可以将TLB下拉请求发送到其它核。TLB下拉请求可以包括标识信息、指示需要从其它核的相应TLB刷新的解除关联的虚拟存储器页的页面的下拉地址以及指示其它核可以在何处确认TLB下拉请求完成的通知地址。

[0007] 在一些示例中,该方法还可以包括在处理器核完成TLB下拉请求时从其它核接收

确认。可以在通知地址处接收确认。所述一个或多个处理器可以执行一个或多个虚拟机，其中所述虚拟机包括一个或多个虚拟计算机处理器。该进程可以是在一个或多个虚拟机中执行的第一进程。

[0008] 在一些示例中，该方法还可以包括：由电力管理单元确定接收到TLB下拉请求的第一核是否处于低电力状态，以及在第一核处于低电力状态的情况下通过电力管理单元确认TLB下拉请求。

[0009] 另一方面包括一种用于在硬件内引导和跟踪TLB下拉的系统。该系统可以包括一个或多个计算设备，其具有包括一个或多个核的一个或多个处理器。在这点上，一个或多个处理器可以被配置为确定在一个或多个处理器核之一上执行的进程导致一个或多个虚拟存储器页面与一个或多个先前关联的物理存储器地址解除关联。正在执行导致解除关联的进程的处理器核可以被配置为生成TLB下拉请求。处理器核可以进一步被配置为将TLB下拉请求发送到其它核。TLB下拉请求可以包括标识信息、指示需要从其它核的相应TLB刷新的解除关联的虚拟存储器页的页面的下拉地址以及指示其它核可以在何处确认TLB下拉请求的完成的通知地址。

附图说明

[0010] 在附图中通过示例而非限制的方式示出了本技术，其中相同的附图标记表示相同的元件，包括：

[0011] 图1示出了根据本公开的方面的系统。

[0012] 图2示出了根据本公开的方面的执行多个虚拟机的系统。

[0013] 图3根据本公开的方面的在多个核上执行虚拟机的系统。

[0014] 图4是根据本公开的实施例的流程图。

[0015] 图5是根据本公开的实施例的流程图。

具体实施方式

[0016] 综述

[0017] 本技术涉及用于有效地处理转换后备缓冲器(TLB)下拉的系统、方法和计算机可读介质。在常规多核处理器和虚拟机(VM)环境中，从处理器上的第一核发送并由其它每个核(或核的子集)接收TLB下拉。然后，每个接收核需要检查其相应的TLB以确定TLB是否包含TLB下拉指示需要去除的一个或多个虚拟页面。在确定TLB下拉是成功的或不是必需的时，接收核需要单独地确认TLB下拉到发送核的完成。这可能是时间和资源密集型进程，特别是当TLB下拉由虚拟机中的进程实例发起时，由此通过多层软件的通信可能显著延迟所有核的进一步处理。

[0018] 根据本公开，可以通过使用完全硬件实施方案来将TLB下拉指引和跟踪到仅与可能需要移除虚拟存储器页面的TLB相关联的那些核，来减少完成TLB下拉所需的处理资源或页面。例如，当在一个或多个处理器核上执行的进程导致一个或多个虚拟存储器页面与一个或多个先前关联的物理存储器地址解除关联时，可能由引起解除关联的核产生TLB下拉请求。TLB下拉请求可以包括标识信息，该标识信息允许每个接收核快速确定是否需要下拉，指示需要从相应TLB刷新的虚拟存储器页面或多个页面的下拉地址，以及指示接收核可

以确认TLB下拉请求的完成的通知地址。

[0019] 标识信息可以包括标识符，每个VM、在VM内的每个VCPU每个处理器核以及在VCPU和/或处理器核内执行的每个进程可以具有标识符。在某些实施例中，例如在x86系统内，标识信息可以包括一个或多个高级可编程中断控制器ID(APICID)、虚拟处理器ID(VPID)和进程上下文标识符(PCID)。APICID可以由诸如操作系统的软件应用跟踪和提供。例如，系统的每个核可以由操作系统或由硬件本身分配APICID。操作系统可以跟踪哪些核正在执行特定进程并确定这些核的APICID。在进程在虚拟机(VM)上操作的情况下，操作系统可以使用表格来将正在执行特定进程的VM的虚拟处理器(VCPU)映射到正在执行VM的物理核。APICID可以是正在执行VM的那些物理核。

[0020] 虚拟处理器ID可以被分配给VM的两个逻辑处理器和物理核。例如，当进程在VM外部执行时，该物理核可以被分配为VPID值是0。在该进程在VM中操作的情况下，控制VM的管理程序可以分配VPID值。

[0021] 可以将进程上下文标识符(PCID)分配给所执行的每个进程。在这点上，每个进程的PCID可以被分配给与相应进程相关联的特定TLB条目。对于请求TLB下拉请求的进程，可以确定PCID。与请求TLB下拉的进程相关联的APICID、VPID和PCID一起可以被认为是标识信息。

[0022] TLB下拉请求可以被发送到在系统上操作的所有核。如本文所描述的，TLB下拉请求可以包括标识信息、下拉地址和通知地址。

[0023] 对于接收到TLB下拉请求的每个核，可以在接收核的标识信息和包含在TLB下拉请求中的标识信息之间进行比较。在这点上，如果接收核的标识信息与包含在TLB下拉请求中的标识信息不匹配，则可以通过硬件机制而不影响CPU的指令流的执行来忽略TLB下拉请求。否则，接收核可以检查其相应的TLB以确定其是否包含下拉地址，并且如果是，则它将使该地址无效。在无效完成时，或者如果下拉地址不在接收核的TLB内，或者如果忽略无效，则接收核可以向通知地址发送无效确认，并恢复其正常功能，诸如继续处理应用。

[0024] 发送核可跟踪从接收核接收的确认的数量。在这点上，发送核可以跟踪在通知地址处接收的确认的数量，直到它期望确认TLB下拉请求的所有接收核这样做。在一些实施例中，发送核可以一发送TLB下拉请求就继续操作。为了监视TLB下拉请求的状态，诸如操作系统的软件可以确保已经接收到所有确认。

[0025] 这里描述的特征可以允许系统更有效地处理TLB下拉请求。在这方面，TLB下拉请求可以被没有执行引起TLB下拉进程的处理器核忽略。另外，发送TLB下拉的处理器核可以接收TLB下拉完成或根本不需要跟踪确认的确认，从而允许发送TLB下拉的处理器核更快地继续其它操作。

[0026] 示例性系统

[0027] 图1示出了可以执行TLB下拉的系统100。在这点上，系统包括至少一个处理器103。系统100还包括页表131、物理存储器141、辅助存储器151和电力管理单元161。

[0028] 物理存储器141和辅助存储器151可以存储可由一个或多个处理器103访问的信息，包括可以由处理器103执行的指令(未示出)。物理和辅助存储器还可以包括可以由处理器103检索、操纵或存储的数据(未示出)。物理和辅助存储器可以是能够存储可由处理器访问的信息的任何非暂时类型，诸如硬盘驱动器、存储卡、ROM、RAM、DVD、CD-ROM、可写的和只

读的存储器。

[0029] 存储在物理和辅助存储器内的指令可以是由处理器直接执行的任何指令集，诸如机器代码，或者由处理器间接执行的任何指令集，诸如脚本。在这方面，术语“指令”、“应用”、“步骤”和“程序”在本文中可以互换使用。指令可以以目标代码格式存储以由处理器直接处理，或者以任何其它计算设备语言包括根据需要解释或预先编译的独立源代码模块的脚本或集合。下面更详细地解释指令的功能、方法和例程。

[0030] 存储在物理和辅助存储器内的数据可以由处理器103根据指令检索、存储和修改。例如，尽管本文描述的主题不受任何特定数据结构限制，但是数据可以存储在计算机寄存器中，在关系数据库中存储为具有许多不同字段和记录的表，或XML文档。数据还可以以任何计算设备可读格式格式化，例如但不限于二进制值、ASCII或Unicode。此外，数据可以包括足以标识相关信息的任何信息，例如数字、描述性文本、专有代码、指针、对存储在其它存储器中的数据的引用，例如在其它网络位置，或由函数使用的信息计算相关数据。

[0031] 处理器103可以是任何常规处理器，诸如来自Intel公司或Advanced Micro Devices的处理器。可替代的，处理器可以是诸如专用集成电路(ASIC)、现场可编程门阵列(FPGA)等的专用控制器。另外，处理器103可以包括多个处理器、多核处理器或其组合。虽然在图1中仅示出了一个处理器103，但是本领域普通技术人员将认识到，系统100中可以存在若干处理器。因此，对处理器的引用将被理解为包括对可以并行操作或可以不并行操作的处理器或专用逻辑的集合的引用。

[0032] 处理器103可以包括一个或多个核105-111。每个核可以独立于其它核执行程序指令，从而实现多处理。虽然未示出，但是每个核可以包含高速缓存(cache)，或者单个高速缓存可以与所有核相关联。

[0033] 系统还包括至少一个页表131。页表131可以用于将在处理器103内缓存的一段数据的虚拟存储器页面映射到存储数据的相应物理地址，比如在物理存储器141或辅助存储器151中。尽管页表131被示出在处理器103的外部，但是页表131可以存储在处理器103的高速缓存内或者存储在物理存储器141或辅助存储器151中。

[0034] 为了提高将虚拟存储器页面转换为相应物理地址的速度，处理器的每个核105-111可以包括转换后备缓冲器(TLB)121-127，其存储虚拟存储器页面到物理存储器的最近转换地址。

[0035] 系统100可以包括电力管理单元161。电力管理单元可以跟踪处理器103内的每个核的电力状态。这种状态可以包括“开”状态、“关”状态或者在“开”和“关”状态之间的电力状态范围。如下面详细描述的，电力管理单元可以被编程为代表每个核进行通信。

[0036] 如图2所示，系统100可以被配置为操作一个或多个虚拟机(VM)201-205。在这方面，管理程序221可以安装于在处理器103上执行的主机231上。在操作中，主机231可以运行包括管理程序(诸如管理程序231)或虚拟机管理器(VMM)的操作系统。在一些实施例中，管理程序221可以在没有主机231的情况下直接在处理器103上操作。为了本申请的目的，管理程序和VMM可以互换使用。此外，本领域普通技术人员将认识到，主机231的操作系统可以是Linux、Windows™或能够支持虚拟机的任何其它合适的操作系统。

[0037] 管理程序可以管理每个VM，使得VM看起来彼此隔离。也就是说，每个VM 201、203和205可以认为自己是具有其自己的硬件资源的独立机器。在这点上，管理程序221可以控制

VM访问系统100的资源(即,存储器、网络接口控制器等)。管理程序221可以实现根据需要向VM分配硬件资源的硬件虚拟化方案。根据一些示例,处理器103是VM 201、203和205经由管理程序221与其交互的硬件资源之一。

[0038] VM 201、203和205是计算机的软件实施方案。也就是说,VM 201、203和205执行操作系统。虽然在附图中仅示出了三个VM,但是本领域的普通技术人员将认识到系统100可以支持任何数量的VM。各个VM 201、203和205的操作系统可以是相同的操作系统作为主机231,但不一定必须是。此外,每个VM的操作系统可以不同于其它VM。例如,主机231可以运行基于Linux的操作系统,而VM 201可以运行Windows™操作系统,并且VM 203可以运行Solaris™操作系统。操作系统的各种组合对于本领域技术人员将是显而易见的,并且在本文中不再更详细地讨论。在一些实施例中,VM可以嵌套在其它VM内。在这点上,VM 201可以向另一个客户VM播放主机。

[0039] 每个VM可以包括其自己的虚拟中央处理单元(VCPU)211-215。尽管图2仅示出了具有单个VCPU的VM,但是VM可以包括任何数量的VCPU。VCPU是被分配给物理处理器(例如处理器103的处理器核)的虚拟化处理器。每个VCPU可以具有其自己的唯一标识,称为虚拟处理器ID(VPID)。

[0040] 示例性方法

[0041] 为了有效地处理TLB下拉请求,可以通过将TLB下拉引导和跟踪到仅与可能需要去除虚拟存储器页面的TLB相关联的那些核来减少TLB下拉。例如,如图4所示,当在一个或多个处理器核上执行的进程导致一个或多个虚拟存储器页面与一个或多个先前关联的物理存储器地址解除关联时,可以由导致关联解除的核产生TLB下拉请求,如框图401所示。如前所述,TLB下拉请求可以包括允许每个接收核快速确定是否需要下拉的标识信息、指示需要从相应TLB刷新的虚拟存储器页面的下拉地址以及指示接收核可以在何处确认TLB下拉请求的完成的通知地址。

[0042] 标识信息可以包括标识符,每个VM、VM内的每个VCPU、每个处理器核以及在VCPU和/或处理器核内执行的每个进程可以具有标识符。在某些实施例中,例如在x86系统内,标识信息可以包括一个或多个高级可编程中断控制器ID(APICID)、虚拟处理器ID(VPID)和进程上下文标识符(PCID)。APICID可以由诸如主机231操作系统之类的软件应用来跟踪和提供,如框图403所示。例如,系统的每个核可以由操作系统分配APICID,诸如在主机上执行的操作系统。操作系统可以跟踪哪些核正在执行特定进程并且确定那些核的APICID。

[0043] 为了将APICID分配给TLB下拉请求,可以确定发起TLB下拉请求的进程是否正在VM(客户)或主机(实际)上执行,如框图404所示。在进程在VM上操作的情况下,操作系统可以使用表来将正在执行特定进程的VM的虚拟处理器(VCPU)映射到正在执行该进程的物理核VM,如框图405所示。正在执行VM的那些物理核的APICID可以被分配为TLB下拉请求的APICID。在进程未在VM中执行的情况下,APICID将是正在执行进程的真实APICID,如框图409所示。

[0044] 在图3中示出了向VCPU分配APICID的示例。在这点上,包括VCPU 211的VM 201可以由管理程序221控制,管理程序221又安装在主机操作系统231上。VCPU 211可以在第一时间分配给处理器的核105,如框301所示。主机操作系统231可以包含将VCPU 211的APICID映射到处理器核105和107的APICID的表(未示出)。这样,当在VM 201上执行的第一进程请求TLB

下拉时，主机操作系统可以分配与处理器核105和107相关联的两个APICID。

[0045] 返回到图4，虚拟处理器ID可以被分配给VM的两个逻辑处理器和分配给物理核。例如，当在物理核上执行该进程时，可以向该物理核分配VPID值为0，如框图411所示。在进程在VM上操作的情况下，控制VM的管理程序可以分配VPID值。在这点上，可以忽略在VM内操作的VCPU的TLB下拉请求中的VPID值，因为该VPID值可随着从系统100添加或移除VM而改变。

[0046] 可以将进程上下文标识符(PCID)分配给所执行的每个进程。进程上下文标识符为由处理器执行的每个进程提供唯一的标签，包括VCPU。在这点上，每个进程的PCID可以被分配给与相应进程相关联的特定TLB条目。对于请求TLB下拉请求的进程，PCID可以由处理器确定。与请求TLB下拉的进程相关联的APICID、VPID和PCID可以一起被认为是标识信息。

[0047] TLB下拉的标识信息、下拉地址和通知地址可以作为元组被写入存储器，例如物理存储器141，或者可以被分配给处理器103中的寄存器(未示出)。例如，在x86处理器架构的情况下，TLB下拉的标识信息、下拉地址和通知地址可以被输入到各种处理器注册表中，诸如RDX、RAX、RCX和RDI寄存器。在一个实例中，APICID可以被输入到RDX寄存器中，并且VPID和PCID可以经由写入模型专用寄存器命令(WRMSR)被输入到RDI寄存器中。

[0048] RCX寄存器可以被编码以指示将要发送中断命令(例如，TLB下拉请求)。例如，在x86系统中，现有的RCX寄存器值可以是0x830，RDX可以是目的字段(例如，APICID)。现有RAX值可以如下：目的地简写：00；触发模式：0；电平：1级；目的地模式：0或1表示物理或逻辑。可以通过向“传送模式”和“向量”添加新的编码来修改这些值。例如，传递模式＝010(对于SMI)，其中向量是一个新的非零值。或传递模式＝100(NMI)，其中向量是一个新的非零值。这个新编码可以指示正在执行TLB下拉。此外，其它新值可能在其它寄存器中。例如，RDI可以包含扩展目的地字段(例如，VPID和PCID)，RSI可以包含通知地址，并且RBX包含下拉地址编码。其它编码也可以用于指示正在执行TLB下拉。这些值可以通过写入模型专用寄存器命令(WRMSR)输入到RAX寄存器以及其它寄存器。

[0049] APICID可以被编码以指示TLB下拉请求是否将以逻辑或物理模式发送。在这方面，物理模式将需要对系统中每个核单独的TLB下拉请求。例如，处理器103包含四个核，其中之一是发送TLB请求。这样，三个单独的下拉请求将被发送到其它三个核中的每一个。当APICID处于逻辑模式时，其它三个核都将发送相同的TLB下拉请求。

[0050] 下拉地址可以包括在元组或注册表中。在这点上，下拉地址可以被编码到元组或注册表中。例如，下拉地址可以被编码为4kb页面的一页地址(例如，与INVLPG指令编码匹配的地址)、开始地址和页数(例如，被编码为log(页数)或编码为(页数))或作为无效地址的列表的一个或多个。

[0051] TLB下拉请求可以被发送到在系统上操作的所有核，如框图417所示。如所描述的，TLB下拉请求可以包括标识信息、下拉地址和通知地址。在一些实施例中，操作系统可以将TLB下拉仅引导到与可能需要去除虚拟存储器页面的TLB相关联的那些核。这样，OS可以将TLB下拉引导到核的子集。

[0052] 现在转到图5，在接收到TLB下拉请求的每个核中，可以在接收核的标识信息和包含在TLB下拉请求中的标识信息之间进行比较，如框图501所示。在这一点上，如果接收核的标识信息与在TLB下拉请求中包含的标识信息不匹配，则可以忽略TLB下拉请求，如框图503所示。否则，接收核可以检查其相应的TLB以确定其是否包含下拉地址，如果是，它将使该地

址无效，如框图505所示。

[0053] 在无效完成时，或者如果下拉地址不在接收核的TLB内，则接收核可以向通知地址发送无效确认，如框图507所示。还可以在接收核的标识信息与包含在TLB下拉请求中的标识信息不匹配时发送确认。接收核可以恢复正常功能，例如在完成TLB下拉时继续处理应用。在一些示例中，确认可以由接收TLB下拉的所有核发送，甚至从忽略TLB下拉的接收核发送。在其它示例中，只有服从下拉的核才会发送确认。

[0054] 在一些示例中，处理器的电力管理单元可以确认TLB下拉的完成。在这点上，核可以在切换到低于特定阈值的电力状态时刷新其TLB，使得对该核的TLB无效没有进一步的影响。如果核在低于阈值的低电力状态下接收到TLB下拉请求，则电力管理单元可代表接收核确认TLB下拉请求的完成，因为处于低电力状态的接收核先前已经释放了其整个TLB。通过允许电力管理单元确认TLB下拉请求的完成，与如果发送核需要等待直到接收核返回到更高电力状态相比，发送核将接收对请求的更快响应。此外，接收核的处理可以不被TLB下拉请求中断。在一些示例中，系统100内的其它部件(诸如高级可编程中断控制器(APIC)(未示出))可以用于代表每个核进行通信。例如，APIC可以确认TLB下拉的完成。通常，可以扩展或创建其它总是被供电的硬件代理以代表核执行确认。例如，当前未运行的VCPU将不具有与其相关联的TLB条目，因此总是供电的代理确认可以允许完成TLB下拉请求而没有来自VCPU的输入。

[0055] 发送核可跟踪从接收核接收的确认的数目。在这点上，发送核可以跟踪在通知地址处接收的确认的数量，直到它期望确认TLB下拉请求的所有接收核这样做。例如，基于包含在TLB下拉请求中的APICID，传输核可以知道预期有多少确认。这样，发送核可以监视通知地址，例如位向量的特定虚拟或物理地址，直到接收到这些确认数目。在一些示例中，发送核可以一旦发送TLB下拉请求就继续操作。为了监视TLB下拉请求的状态，诸如主机操作系统231的软件可以确保已经接收到所有确认。

[0056] 当在单个系统内发现多个处理器时，套接字管理器(socket manager)可以被分配给每个相应的处理器。然后，每个相应的套接字管理器可以监视来自其处理器内的所有核的确认。套接字管理器然后可以将接收到的确认报告回通知地址。在一些实施例中，套接字管理器可以被分配给多于单个处理器。在这点上，套接字管理器可以被分配到插座的子区域、电路板、机箱等。

[0057] 可以通过写入高速缓存行来触发TLB下拉。在这点上，每个VCPU可以在每个VM、VCPU对而言唯一的共享状态中保持高速缓存行。为了启动TLB下拉，正在发起TLB下拉的VCPU可以写入这个高速缓存行，导致高速缓存行从其相应的高速缓存释放并且触发指示核应该启动TLB下拉进程的硬件或微代码机制。

[0058] 此外，TLB下拉可以仅对准特定的核。在这点上，软件可以维持所有VCPU/VM到核的映射。基于该映射，TLB下拉可以仅针对处理触发TLB下拉的进程的那些核。

[0059] 大多数前述替代示例不是相互排斥的，而是可以以各种组合来实施以实现独特的优点。由于在不脱离由权利要求限定的主题的情况下可以利用上述特征的这些和其它变化和组合，因此应当通过说明而不是限制由权利要求限定的主题来对实施例进行前述描述。作为示例，前面的操作不必以上述精确的顺序执行。相反，可以以不同的顺序处理各种步骤，诸如颠倒或同时处理。除非另有说明，步骤也可以省略。此外，本文描述的示例以及措辞

被称为“例如”、“包括”等的术语的提供不应被解释为将权利要求的主题限制为具体示例；而是，这些示例旨在仅示出许多可能的实施例中的一个。此外，在不同附图中相同的附图标记可以标识相同或相似的元件。

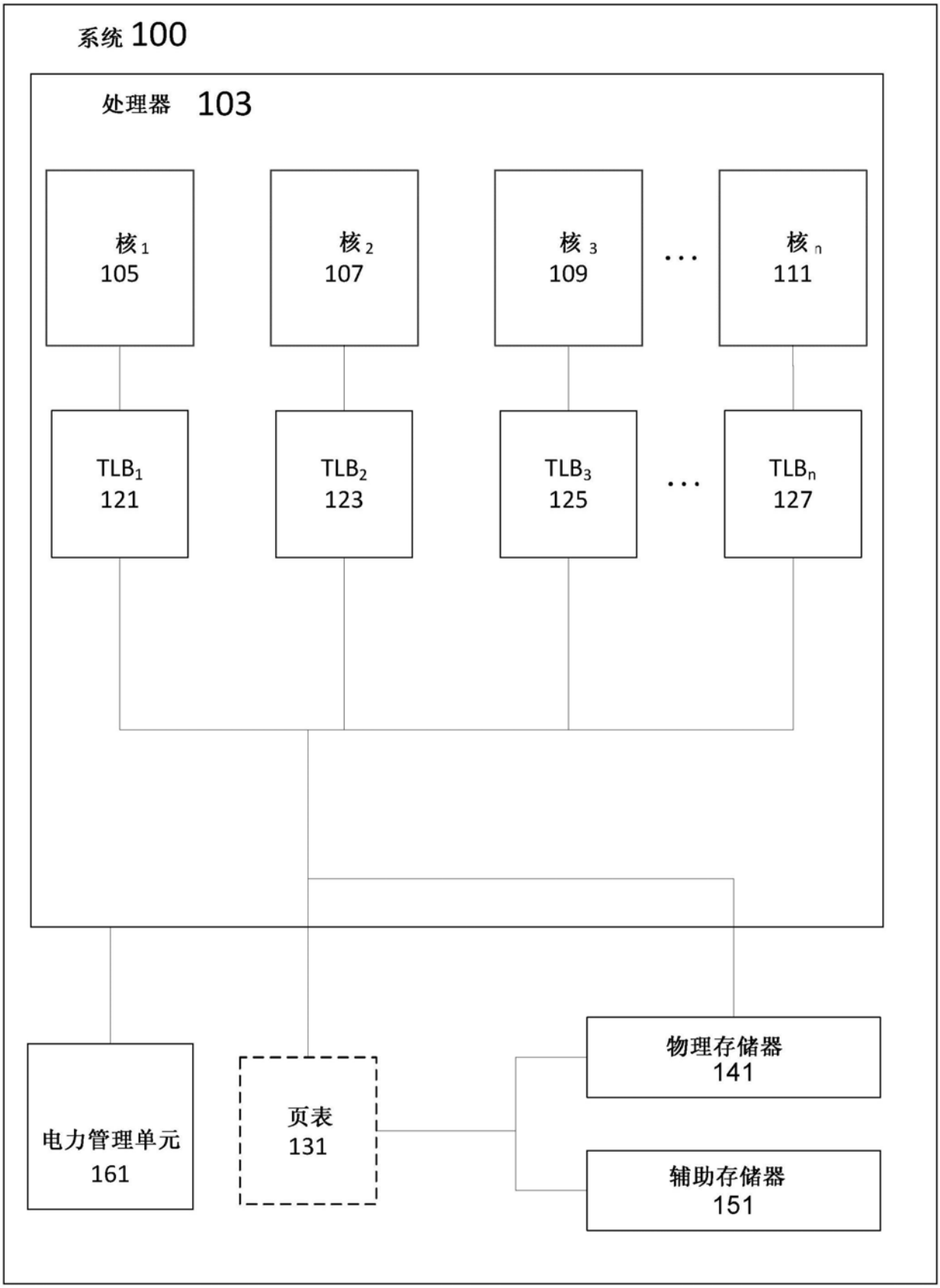

系统 100
处理器 103
核 1
105
核 2
107
核 3
109
核 n
111
TLB1
121
TLB2
123
TLB3
125
TLBn
127
电力管理单元
161
页表
131
物理存储器
141
辅助存储器
151

图1

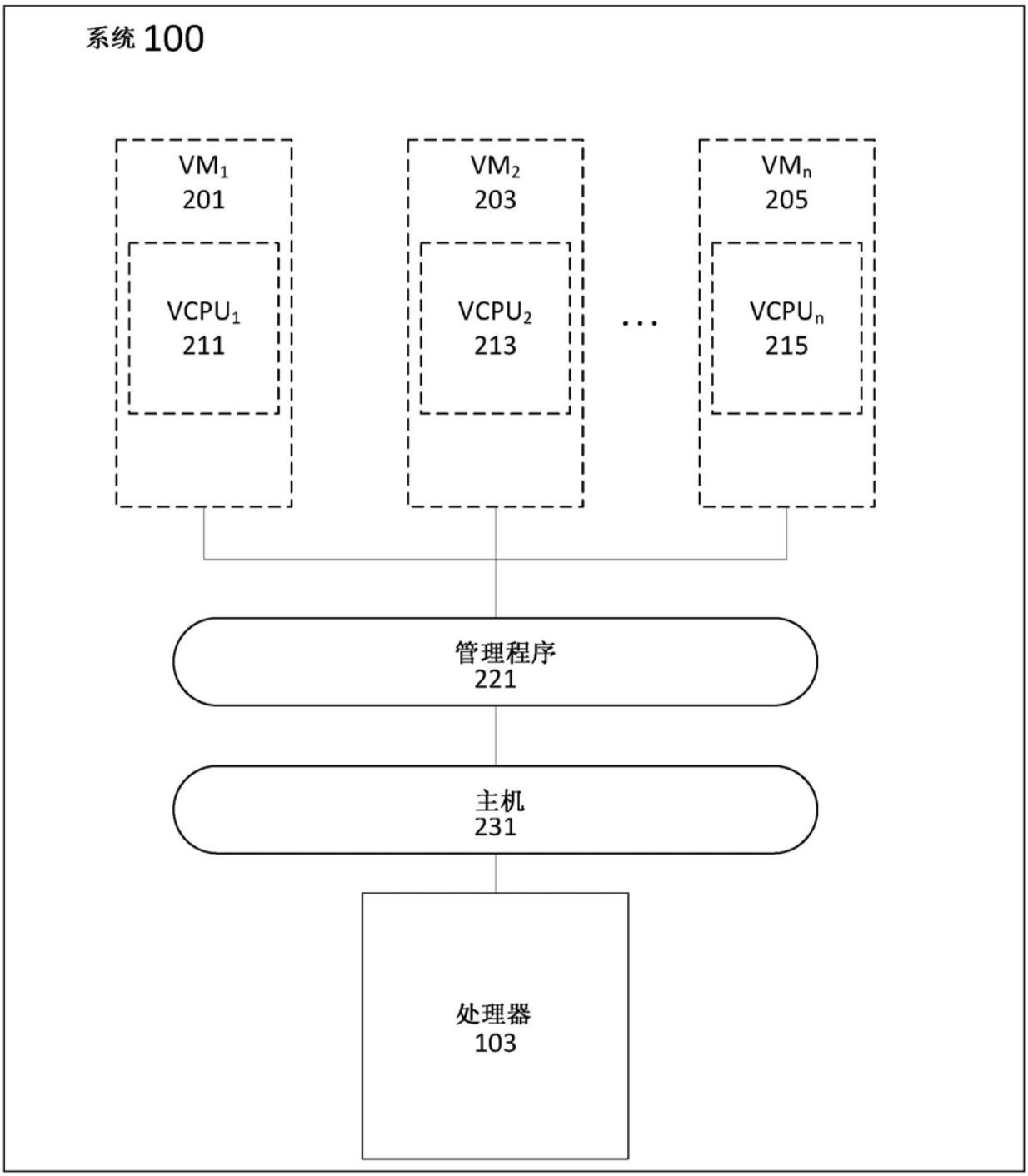

系统 100
VM1
201
VCPU1
211
VM2
203
VCPU2
213
...
VMn
205
VCPUn
215
管理程序
221
主机
231
处理器
103

图2

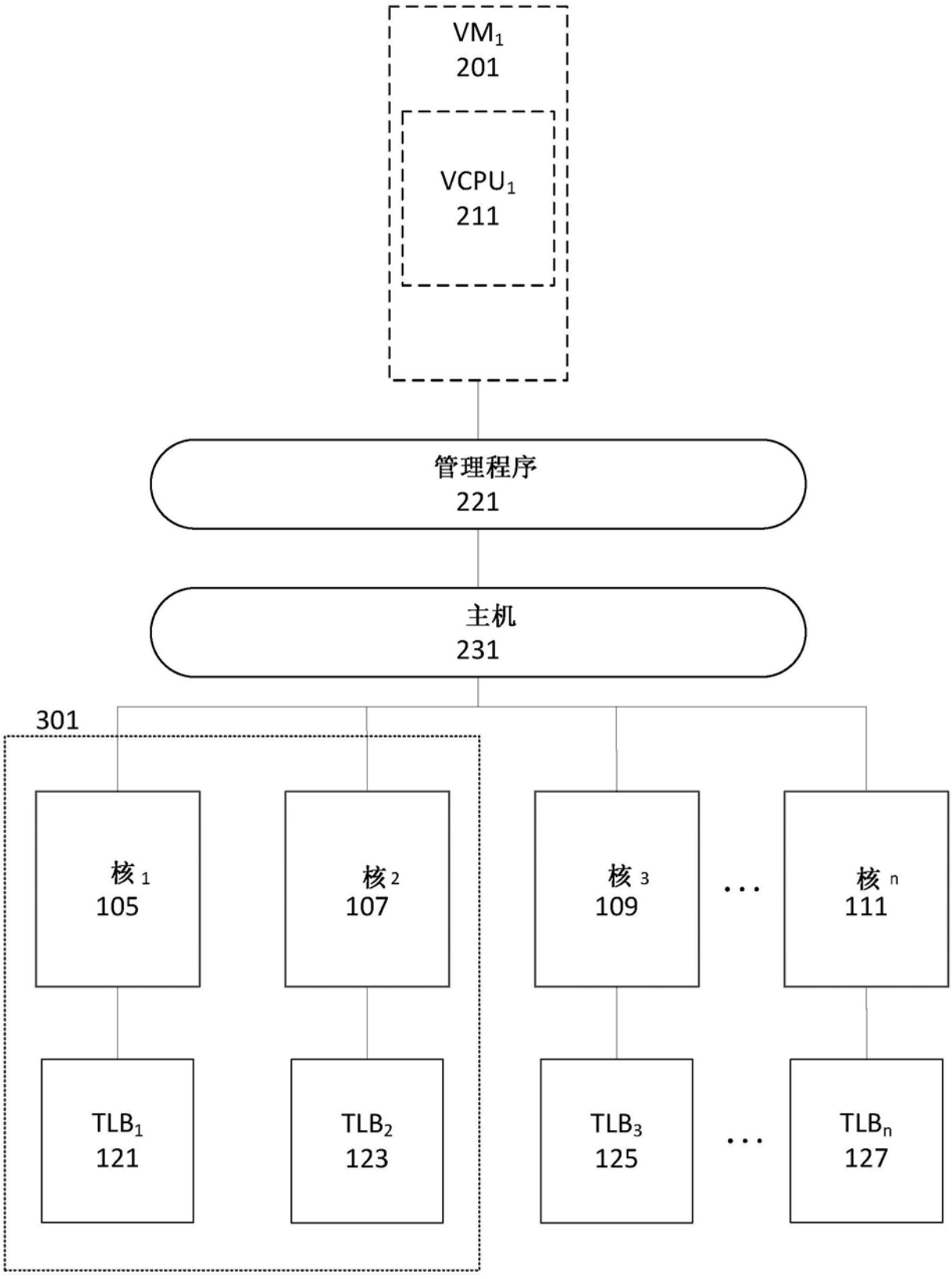

VM1
201
VCPU1
211
管理程序
221
主机
231
301
核 1
105
核 2
107
核 3
109
...
核 n
111
TLB1
121
TLB2
123
TLB3
125
...
TLBn
127

图3

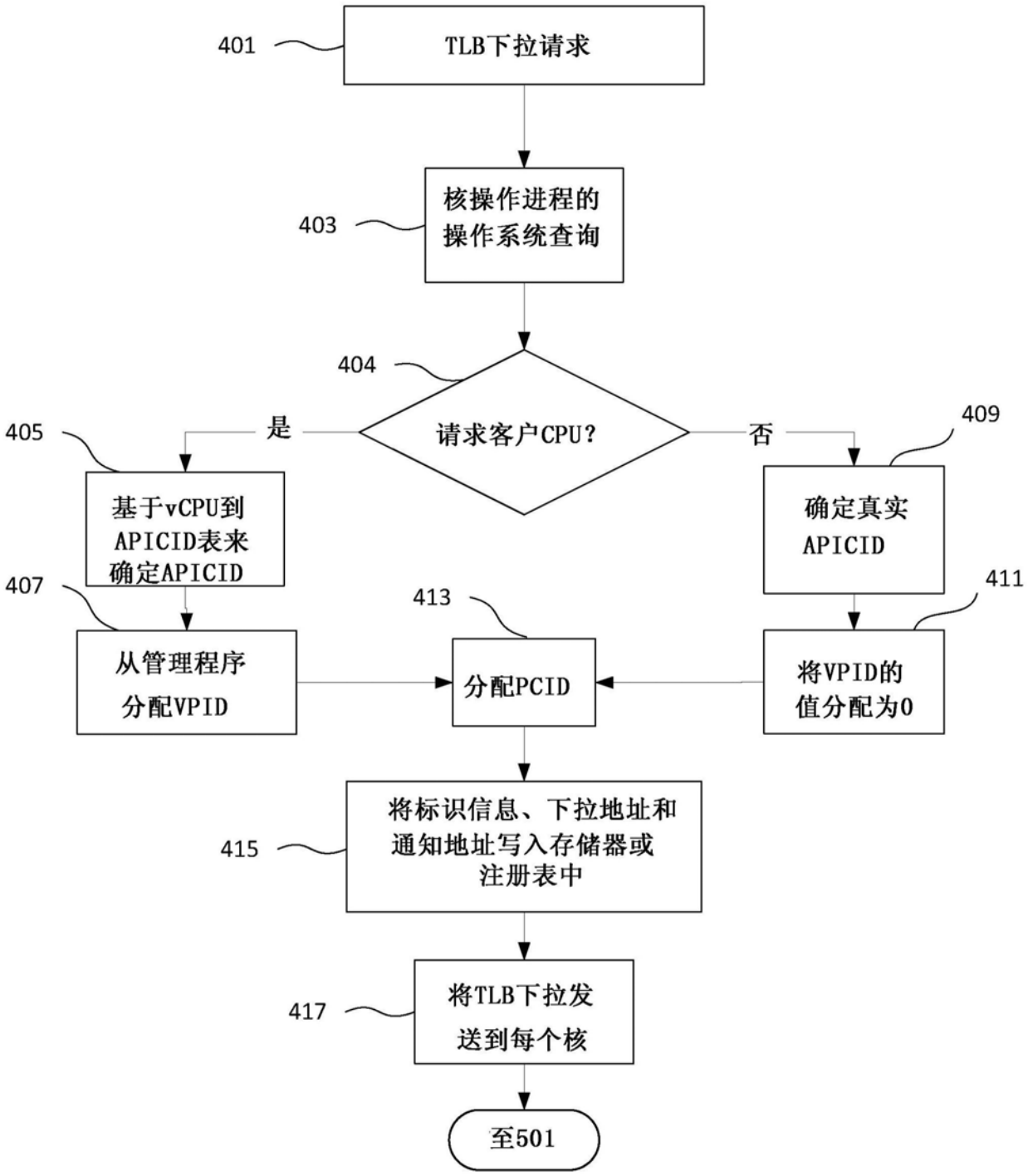

401
TLB下拉请求
403
核操作进程的操作系统查询
404
请求客户CPU?
是
否
405
基于vCPU到APICID表来确定APICID
409
确定真实APICID
407
从管理程序分配VPID
413
分配PCID
411
将VPID的值分配为0
415
将标识信息、下拉地址和通知地址写入存储器或注册表中
417
将TLB下拉发送到每个核
至501

图4

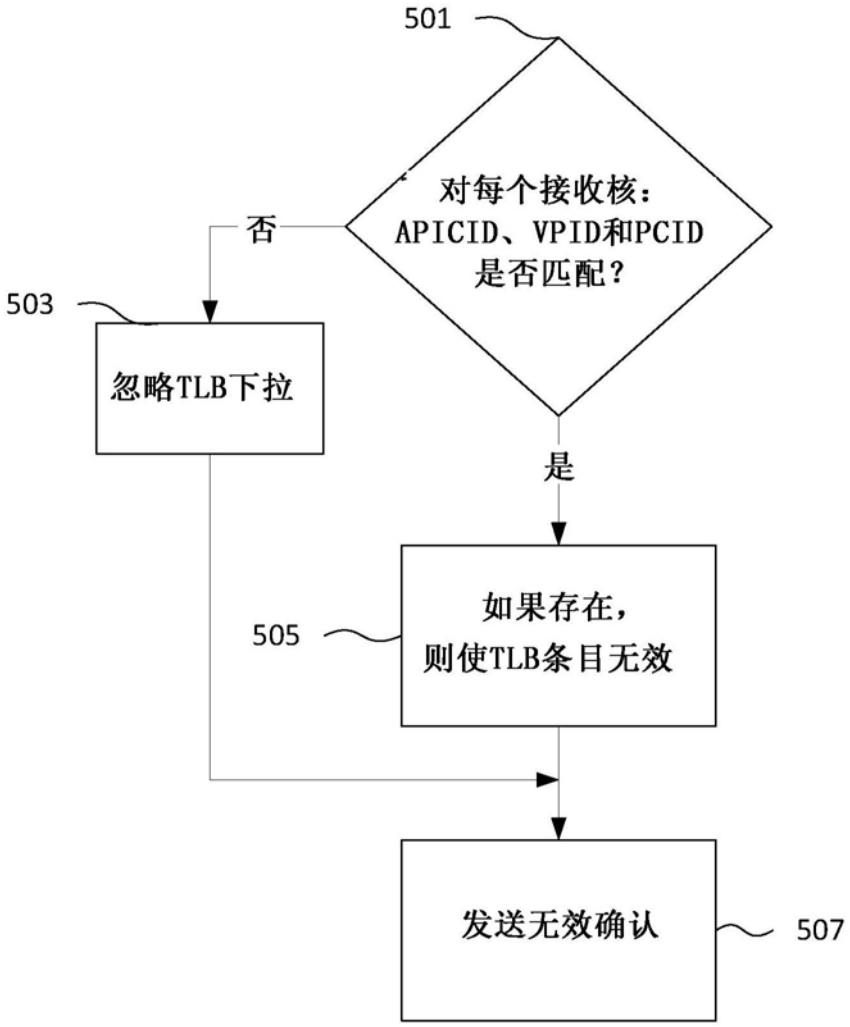

501
对每个接收核：
APICID、VPID和PCID
是否匹配？
否
503
忽略TLB下拉
是
505
如果存在，
则使TLB条目无效
发送无效确认
507

图5