[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G06F 12/10 (2006.01)

[12] 发明专利说明书

专利号 ZL 200410088261.X

[45] 授权公告日 2008 年 10 月 8 日

[11] 授权公告号 CN 100424657C

[22] 申请日 2004.10.18
[21] 申请号 200410088261.X
[30] 优先权
[32] 2003.11.26 [33] US [31] 10/723,823
[73] 专利权人 微软公司
地址 美国华盛顿州
[72] 发明人 E·S·科恩
[56] 参考文献
US2003/0200402A1 2003.10.23
US6510508B1 2003.1.21
CN1361887A 2002.7.31
Translation – Lookaside Buffer Consistency. Patricia J. Teller. COMPUTER, Vol. 23 No. 6. 1990
审查员 俞立文

[74] 专利代理机构 上海专利商标事务所有限公司
代理人 陈 斌

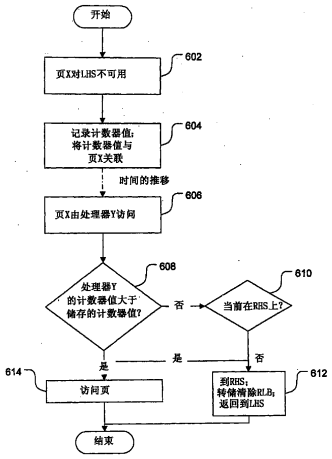
权利要求书 4 页 说明书 15 页 附图 6 页

[54] 发明名称

转换后备缓冲器的惰性转储清除

[57] 摘要

地址转换控制(ATC)限制了虚拟和物理地址之间的映射，以实现存储器访问政策。多处理器系统中的每一处理器维护高速缓存映射的转换后备缓冲器(TLB)，以加速虚拟地址的转换。每一处理器也维护一计数器。每次当处理器的 TLB 被转储清除时，递增该处理器的计数器。当从地址转换映像移除到页的链接时，记录所有处理器的计数器值。当该页由处理器访问时，将记录的计数器值与该处理器的当前计数器值比较，以确定自从到该页的链接被从映射中移除以来该处理器的 TLB 是否已被转储清除。昂贵的 TLB 转储清除操作被延迟直到需要，但是仍足够早地出现以防止无效的 TLB 条目被用于违反访问政策。



1．一种从多个映射高速缓存的第一个映射高速缓存中清除过时条目的方法，所述多个映射高速缓存的每一个与计算装置的多个处理单元的对应的一个关联，每一所述高速缓存用于将虚拟地址转换成物理地址并基于地址转换映像储存映射，其特征在于，所述方法包括：

维护一计数器；

每次当转储清除所述多个映射高速缓存的第一个时更新所述计数器；

响应于所述地址转换映像中的变化记录所述计数器的值；

基于所述计数器的当前值与所记录的计数器的值的比较确定自从所述地址转换映像中的所述变化出现以来所述多个映射高速缓存的第一个一定未被转储清除；以及

转储清除所述多个映射高速缓存的第一个。

2．如权利要求1所述的方法，其特征在于，所述计数器是多个计数器之一，每一计数器与所述映射高速缓存的对应的一个关联。

3．如权利要求2所述的方法，其特征在于，当计数器的对应的映射高速缓存被转储清除时，更新所述多个计数器的每一个。

4．如权利要求1所述的方法，其特征在于，所述地址转换映像包括到具有第一页的存储器页的链接，并且其中，所述变化包括将所述地址转换映像置于所述地址转换映像不包含到所述第一页的任一链接的状态。

5．如权利要求1所述的方法，其特征在于，所述地址转换映像定义可由实体读取的存储器的部分，所述存储器的部分包括第一部分，并且其中，所述变化包括将所述地址转换映像置于所述第一部分不可由所述实体读取的状态。

6．如权利要求1所述的方法，其特征在于，所述地址转换映像定义可由实体写入的存储器的部分，所述存储器的部分包括第一部分，并且其中，所述变化包括将所述地址转换映像置于所述第一部分不可由所述实体写入的状态。

7．如权利要求1所述的方法，其特征在于，一政策定义对存储器的许可访问，并且其中，所述方法还包括：

控制所述地址转换映像的内容，使得所述地址转换映像不向实体展现准许所述实体在违反所述政策的情况下访问所述存储器的虚拟地址映射；

其中，所述变化包括：使所述映射适应所述政策或维持所述映射与所述政策的适应性的对所述映射的修改，或限制所述实体对所述映射的访问的对所述映射的修改。

8．如权利要求 7 所述的方法，其特征在于，所述地址转换映像包括到所述存储器的一部分的链接，其中，控制所述地址转换映像的内容包括令所述存储器的部分对所述实体不可访问，并且其中，所述变化包括从所述地址转换映像中移除到所述存储器的部分的所有链接。

9．如权利要求 1 所述的方法，其特征在于，它还包括：

确定可能使用所述地址转换映像在所述变化之前的状态的事件已出现；

其中，所述多个映射高速缓存的第一个的转储清除响应于确定所述事件已出现而执行。

10．如权利要求 9 所述的方法，其特征在于，所述事件包括将虚拟地址转换成包含在由所述变化从所述地址转换映像中解除链接的存储器部分内的物理地址。

11．一种管理地址映射高速缓存的应用的系统，其特征在于，所述系统包括：

多个处理器，每一所述处理器具有一映射高速缓存以及一与其关联的计数器；

储存地址转换映像的存储器，每一所述映射高速缓存基于所述地址转换映像储存映射，有一支配对所述存储器的访问的政策，控制所述地址转换映像的内容来防止在违反所述政策的情况下准许对所述存储器的访问的映射的展现；

第一逻辑模块，所述第一逻辑模块转储清除所述映射高速缓存的第一个，并且当所述映射高速缓存的第一个被转储清除时，递增所述计数器的第一个；

第二逻辑模块，所述第二逻辑模块响应于所述地址转换映像中或关于所述地址转换映像的特性中的变化，记录所述第一计数器的当前值，所记录的计数器的值与所述变化关联地储存；

第三逻辑模块，所述第三逻辑模块将所记录的计数器的值与所述第一计数器的当前值相比较，并且如果所述比较指示自从变化以来所述映射高速缓存的第一个尚未被转储清除，则促使所述映射高速缓存的第一个被转储清除。

12．如权利要求 11 所述的系统，其特征在于，每一所述处理器与所述计数器的第一个关联，并且其中，所述第一逻辑模块在任一所述映射高速缓存被转储清除时递增所述计数器的第一个。

13．如权利要求 11 所述的系统，其特征在于，所述映射高速缓存的每一个与

多个计数器的不同的一个关联，并且其中，所述的第一逻辑模块在一给定的映射高速缓存被转储清除时递增对应于所述给定映射高速缓存的计数器。

14．如权利要求 11 所述的系统，其特征在于，所述变化包括将所述地址转换映像置于在其中到所述存储器的第一页的所有链接都被从所述地址转换映像移除的状态。

15．如权利要求 11 所述的系统，其特征在于，所述变化包括将所述地址转换映像置于其中不存在到所述存储器的第一页的可写链接的状态。

16．如权利要求 11 所述的系统，其特征在于，所述第三逻辑模块响应于检测到要使用所述变化的结果而被调用。

17．如权利要求 16 所述的系统，其特征在于，所述变化包括将所述地址转换映像置于其中到所述存储器的第一页的所有链接都被从所述地址转换映像中移除的状态，并且其中，所述检测是基于被转换成所述第一页上的位置的虚拟地址。

18．一种用于管理高速缓存地址映射的转换后备缓冲器的转储清除的方法，其特征在于，所述方法包括：

接收由虚拟地址指示目标位置的访问请求；

转换所述虚拟地址来获取所述目标位置的物理地址；

将（1）与包括所述目标地址的页关联的储存的计数器的值与（2）当前计数器的值相比较；

基于所述储存的计数器的值和所述当前计数器的值之间的比较，确定自从影响所述页的映射的事件被修改以来，所述转换后备缓冲器尚未被转储清除；以及

转储清除所述转换后备缓冲器。

19．如权利要求 18 所述的方法，其特征在于，所述地址映射由储存在存储器中的地址转换映像来定义，所述转换后备缓冲器高速缓存基于所述地址转换映像的映射。

20．如权利要求 18 所述的方法，其特征在于，一政策定义存储器对软件实体的可访问性，并且其中，所述事件包括从所述地址映射所基于的地址转换映像中移除到所述存储器的页的链接，所述页在所述政策下对所述软件实体不可访问。

21．如权利要求 18 所述的方法，其特征在于，一政策定义存储器对软件实体的可访问性，并且其中，所述事件包括调整地址转换映像以使对页的映射不可写，其中，或者（1）所述政策将所述页定义为不可由所述软件实体写，或者（2）所述

页储存所述地址转换映像的一部分。

22．如权利要求 18 所述的方法，其特征在于，一地址转换控制机制控制可用于储存地址转换映像的一部分的存储器页组的成员资格，并且其中，所述事件包括所述页组的成员资格中的变化。

23．一种从多个映射高速缓存的第一个清除过时条目的方法，所述多个映射高速缓存的每一个与计算装置的多个处理单元的对应的一个关联，每一所述高速缓存用于将虚拟地址转换到物理地址，并基于地址转换映像储存映射，其特征在于，所述方法包括：

维护基于其可确定事件的顺序前进的对象；

响应于对从页中移除（1）读访问，或（2）读/写访问中的一个或多个的所述地址转换映像的变化，记录所述对象的值；

响应于对所述页的访问请求，基于所述对象的当前值与所记录的值的比较，确定自从所述变化出现以来所述多个映射高速缓存的第一个已被转储清除；以及

如果无法基于所述比较毫无疑问地确定自从所述变化出现以来所述多个映射高速缓存的第一个已被转储清除，则转储清除所述多个映射高速缓存的第一个。

24．如权利要求 23 所述的方法，其特征在于，所述对象包括时钟。

25．如权利要求 23 所述的方法，其特征在于，所述对象包括在每次所述多个映射高速缓存的第一个被转储清除时被递增的计数器。

26．如权利要求 23 所述的方法，其特征在于，所述对象包括多个计数器，每一所述计数器对应于所述多个映射高速缓存之一，其中，当所述计数器的对应的映射高速缓存被转储清除时，递增每一所述计数器。

27．如权利要求 23 所述的方法，其特征在于，所述变化包括移除对所述页的读访问，并且其中，所述访问请求包括读取所述页的请求。

28．如权利要求 23 所述的方法，其特征在于，所述变化包括移除对所述页的读/写访问，并且其中，所述访问请求包括写所述页的请求。

转换后备缓冲器的惰性转储清除

技术领域

本发明一般涉及存储器管理领域，尤其涉及地址转换高速缓存的转储清除。

背景技术

大多数计算机系统提供虚拟地址机制，藉此将虚拟地址映射到物理地址。当使用虚拟地址作出访问存储器位置的请求时，将该虚拟地址转换成向其寻找访问的目标存储器位置的对应的物理地址。一组地址转换表定义了虚拟地址和物理地址之间的映射。该转换表通常储存于存储器内，因此地址的转换需要存储器访问以读取该表。读该表所需要的存储器访问是在目标位置上执行的访问操作之外的访问。由此，当使用虚拟寻址时，相对于如果所有访问请求由物理地址作出将产生的访问数而言，有系统执行的存储器访问数将加倍。一些虚拟地址就其需要映射分阶段进行间接引用而言是多级的，这意味着可能花费两次或更多次存储器访问来执行地址转换（由此令实现一个基础访问请求所需要的存储器访问数变为三倍或更多）。

为减少转换地址所必须发生的存储器访问数，许多虚拟地址系统采用一种称为转换后备缓冲器（TLB：translation look-aside buffer）的高速缓存类型。由于最近被访问的存储器页可能在不久的将来再次被访问，一旦地址转换表被用于将虚拟页描述符转换成物理页位置，则将虚拟页和物理页之间的对应高速缓存在 TLB 中。每次需要执行地址转换时，检查 TLB 来确定 TLB 是否包含了所请求的存储器单元所处的页的高速缓存的映射。如果相关的映射被高速缓存在 TLB 中，则使用高速缓存的副本；否则，从转换表来转换地址。由于访问 TLB 要比访问存储器中的转换表快，因此当连续的存储器访问位于同一组页上时－事情往往如此－使用 TLB 将加快执行。

当用虚拟存储器提供存储器保护时，TLB 产生了其它问题。存储器保护寻求实施支配哪些软件组件可执行对哪些物理页的哪种访问（如，读、写）的安全政策；该保护可由虚拟存储器通过控制对虚拟－物理地址转换的控制来实施。（该控制可由创建映射的操作系统或将变化过滤到这一映射的地址转换控制（ATC）来行使。）

然而，当修改地址转换时，旧映射仍可在 TLB 中存在。由此，当为某一软件组件修改地址转换表来撤回对页的某一访问权限时，组件可保留对页的访问直到从TLB转储清除（flush）这些旧映射。完成这一过程的普通方法是作为操作的一部分强制转储清除所有相关的 TLB 。

然而，转储清除 TLB 是高代价的，尤其是在共享存储器的多处理器上。必须发信号通知包含违反新安全政策的过时映射的每一处理器转储清除其 TLB；发这种信号通常需要相对较慢的处理器间 中断（IPI）。另外，转储清除本身的开销大。

鉴于以上原因，需要一种克服现有技术的缺点的机制。

发明内容

本发明提供了一种支持 TLB 的惰性转储清除（lazy flushing）的机制。对每一 TLB，维护一计数器，并且每次当转储清除 TLB 时，递增该计数器。在每一处理器维护其自己的 TLB 的多处理器系统中，每一处理器可有其单独的计数器，其中，当一特定处理器转储清除其 TLB 时，递增该处理器的计数器。当触发事件出现时，在触发事件完成之后记录任一相关计数器的值。“触发事件”是以过时的 TLB 条目会导致政策的违反的方式影响地址转换映像或约束映射的存储器访问政策的事件；相关计数器是可包含这一过时条目的 TLB 的计数器。例如，在某一政策下，可将存储器的给定页声明为在界限外，并且从地址转换映像中移除到该页的所有映射；这种从映射对该页解除链接（de-link）是触发事件的一个示例，并且包含到该页的映射的任一 TLB 的计数器都是相关的。情况可以是执行了触发事件，并且在触发事件的效果或结果实际在地址转换过程中使用之前已经过了充分的时间。（例如，到存储器的给定页的映射可在给定的时间点上发生改变，但是在作出尝试来重新使用解除链接的页之前可能发生了上百万个操作。）当要使用触发事件的结果时，将储存的计数器值与当前计数器值向比较，以确定哪些（如果有的话）相关 TLB 可能自从触发事件以来尚未被转储清除。如果任一相关 TLB 具有匹配其记录值的计数器值，则以常规方式转储清除所有这样的 TLB。

应当注意，如果计数器值改变，则 TLB 是安全的－即，一定没有可导致违反适用的访问政策的过时 TLB 条目，而并且如果计数器值未改变，则 TLB 是不安全的。由此，在转储清除时，可以安全地使用任意大小的计数器，并以任意方式改变计数器（或甚至根本就不改变）。另一可能性是使用实时计数器来为最后一次转储

清除的时间加时间戳；如果时钟被同步到某一时滞（time skew）内，则转储清除的启动方可仅记录修改转换的时间，并假定其时间戳至少超出转换修改时间该时滞的任何转储清除已完成。

在高保证和非高保证环境（分别为“左侧”和“右侧”）并排存在，并且每一 TLB 转储清除需要到右侧的行程的系统中，本发明的机制可用于避免到右侧的不必要的行程。例如，在右侧和左侧之间的每一次变化可导致 TLB 转储清除，并且在每次处理器从右侧移到左侧时，递增每一处理器的计数器。换言之，由于右侧可靠地执行（或核实）TLB 的转储清除，计数器反映出 TLB 包含将破坏存储器访问控制模式的条目的最后时间。每次令页变得对左侧不可访问（或不可写）时，本质上使用计数器值来对该页“加时间戳”。当由处理器访问该页时，确定处理器是否：（1）当前在右侧，或（2）自从页状态改变以来可靠地转储清除了其 TLB。后一确定可通过确保处理器的计数器值大于该页的保存值来执行。

下文描述了本发明的其它特征。

附图说明

当结合附图阅读时，可以更好地理解上述概述以及以下较佳实施例的详细描述。为说明本发明的目的，附图中示出了本发明的示例性构造；然而，本发明不限于所揭示的具体方法和手段。附图中：

图 1 是可在其中实现本发明的各方面的示例计算环境的框图；

图 2 是示例虚拟地址系统的框图；

图 3 是在一个机器上共存的两个环境的框图，并且其存储器部分被遮蔽（curtained）；

图 4 所示是在地址转换过程中使用的转换后备缓冲器（TLB）的框图；

图 5 是包括多个处理器的系统的框图，其中，每一处理器与一 TLB 和一计数器关联；以及

图 6 是依照本发明的 TLB 惰性转储清除的过程的流程图。

具体实施方式

综述

地址转换控制（ATC）可用于通过动态地控制用于将虚拟地址转换成物理地

址的映射来实现存储器访问政策。当转换后备缓冲器（TLB）用于高速缓存映射时，与访问政策的当前状态不一致的旧映射可保留在 TLB 中，由此向政策展现了运行计数器要使用的存储器。TLB 可以被转储清除，但是转储清除 TLB 是昂贵的操作，优选尽可能少地执行该操作。本发明提供了一种惰性 TLB 转储清除的机制，允许推迟 TLB 的转储清除直到需要转储清除。该机制较佳地在每一处理器维护其自己的 TLB 的多处理器系统中使用，可独立于其它处理器的 TLB 来转储清除。

示例性计算布置方案

图 1 示出了适合在其中实现本发明的各方面的一个示例计算环境。计算系统环境 100 仅为合适的计算环境的一个示例，并非建议对本发明的使用或功能的范围的局限。也不应将计算环境 100 解释为对示例性操作环境 100 中示出的任一组件或其组合具有依赖或需求。

本发明可以使用众多其它通用或专用计算系统环境或配置来操作。适合使用本发明的已知的计算系统、环境和/或配置包括但不限于：个人计算机、服务器计算机、手持式或膝上设备、多处理器系统、基于微处理器的系统、机顶盒、可编程消费者电子设备、网络 PC、小型机、大型机、嵌入式系统、包括任一上述系统或设备的分布式计算环境等等。

本发明可在计算机可执行指令的一般上下文环境中描述，计算机可执行指令如由计算机执行的程序模块。一般而言，程序模块包括例程、程序、对象、组件、数据结构等等，执行特定的任务或实现特定的抽象数据类型。本发明也可以在分布式计算环境中实践，其中，任务由通过通信网络或其它数据传输媒质连接的远程处理设备来执行。在分布式计算环境中，程序模块可以位于本地和远程计算机存储媒质中，如存储器存储设备。

参考图 1，用于实现本发明的示例系统包括以计算机 110 形式的通用计算装置。计算机 110 的组件可包括但不限于，处理单元 120、系统存储器 130 以及将包括系统存储器的各类系统组件耦合至处理单元 120 的系统总线 121。处理单元 120 可代表多个逻辑处理单元，如多线程处理器上所支持的。系统总线 121 可以是若干种总线结构类型的任一种，包括存储器总线或存储器控制器、外围总线以及使用各类总线结构的局部总线。作为示例而非局限，这类结构包括工业标准体系结构（ISA）总线、微通道体系结构（MCA）总线、增强 ISA（EISA）总线、视频电子

技术标准协会（VESA）局部总线以及外围部件互连（PCI）总线（也称为 Mezzanine 总线）。系统总线 121 也可被实现为点对点连接、交换光纤等通信设备。

计算机 110 通常包括各种计算机可读媒质。计算机可读媒质可以是可由计算机 110 访问的任一可用媒质，包括易失和非易失媒质、可移动和不可移动媒质。作为示例而非局限，计算机可读媒质包括计算机存储媒质和通信媒质。计算机存储媒质包括以用于储存信息的任一方法或技术实现的易失和非易失，可移动和不可移动媒质，信息如计算机可读指令、数据结构、程序模块或其它数据。计算机存储媒质包括但不限于，RAM、ROM、EEPROM、闪存或其它存储器技术、CD-ROM、数字多功能盘（DVD）或其它光盘存储、磁盒、磁带、磁盘存储或其它磁存储设备、或可以用来储存所期望的信息并可由计算机 110 访问的任一其它媒质。通信媒质通常在诸如载波或其它传输机制的已调制数据信号中包含计算机可读指令、数据结构、程序模块或其它数据，并包括任一信息传送媒质。术语“已调制数据信号”指以对信号中的信息进行编码的方式设置或改变其一个或多个特征的信号。作为示例而非局限，通信媒质包括有线媒质，如有线网络或直接连线连接，以及无线媒质，如声学、RF、红外和其它无线媒质。上述任一的组合也应当包括在计算机可读媒质的范围之内。

系统存储器 130 包括以易失和/或非易失存储器形式的计算机存储媒质，如只读存储器（ROM）131 和随机存取存储器（RAM）132。基本输入/输出系统 133（BIOS）包括如在启动时帮助在计算机 110 内的元件之间传输信息的基本例程，通常储存在 ROM 131 中。RAM 132 通常包含处理单元 120 立即可访问或者当前正在操作的数据和/或程序模块。作为示例而非局限，图 1 示出了操作系统 134、应用程序 135、其它程序模块 136 和程序数据 137。

计算机 110 也可包括其它可移动/不可移动、易失/非易失计算机存储媒质。仅作示例，图 1 示出了对不可移动、非易失磁媒质进行读写的硬盘驱动器 141、对可移动、非易失磁盘 152 进行读写的磁盘驱动器 151 以及对可移动、非易失光盘 156，如 CD ROM 或其它光媒质进行读写的光盘驱动器 155。可以在示例性操作环境中使用的其它可移动/不可移动、易失/非易失计算机存储媒质包括但不限于，磁带盒、闪存卡、数字多功能盘、数字视频带、固态 RAM、固态 ROM 等等。硬盘驱动器 141 通常通过不可移动存储器接口，如接口 140 连接到系统总线 121，磁盘驱动器 151 和光盘驱动器 155 通常通过可移动存储器接口，如接口 150 连接到系统总线

121。

图 1 讨论并示出的驱动器及其关联的计算机存储媒质为计算机 110 提供了计算机可读指令、数据结构、程序模块和其它数据的存储。例如，在图 1 中，示出硬盘驱动器 141 储存操作系统 144、应用程序 145、其它程序模块 146 和程序数据 147。注意，这些组件可以与操作系统 134、应用程序 135、其它程序模块 136 和程序数据 137 相同，也可以与它们不同。这里对操作系统 144、应用程序 145、其它程序模块 146 和程序数据 147 给予不同的标号来说明至少它们是不同的副本。用户可以通过输入设备，如键盘 162 和定位设备 161（通常指鼠标、跟踪球或触摸板）向计算机 110 输入命令和信息。其它输入设备（未示出）可包括麦克风、操纵杆、游戏垫、圆盘式卫星天线、扫描仪等等。这些和其它输入设备通常通过耦合至系统总线的用户输入接口 160 连接至处理单元 120，但是也可以通过其它接口和总线结构连接，如并行端口、游戏端口或通用串行总线（USB）。监视器 191 或其它类型的显示设备也通过接口，如视频接口 190 连接至系统总线 121。除监视器之外，计算机也包括其它外围输出设备，如扬声器 197 和打印机 196，通过输出外围接口 195 连接。

计算机 110 可以在使用到一个或多个远程计算机，如远程计算机 180 的逻辑连接的网络化环境中操作。远程计算机 180 可以是个人计算机、服务器、路由器、网络 PC、对等设备或其它公用网络节点，并通常包括许多或所有上述与计算机 110 相关的元件，尽管在图 1 中仅示出了存储器存储设备 181。图 1 描述的逻辑连接包括局域网（LAN）171 和广域网（WAN）173，但也可包括其它网络。这类网络环境常见于办公室、企业范围计算机网络、内联网以及因特网。

当在 LAN 网络环境中使用时，计算机 110 通过网络接口或适配器 170 连接至 LAN 171。当在 WAN 网络环境中使用时，计算机 110 通常包括调制解调器 172 或其它装置，用于通过 WAN 173，如因特网建立通信。调制解调器 172 可以是内置或外置的，通过用户输入接口 160 或其它合适的机制连接至系统总线 121。在网络化环境中，描述的与计算机 110 相关的程序模块或其部分可储存在远程存储器存储设备中。作为示例而非局限，图 1 示出了远程应用程序 185 驻留在存储器设备 181 中。可以理解，示出的网络连接是示例性的，也可以使用在计算机之间建立通信链路的其它装置。

示例性虚拟地址模式

图 2 示出了虚拟地址系统的一个示例。图 2 描述的示例是一分页类型虚拟地址模式，尽管可以理解，虚拟寻址可基于其它模型，如分段（segmentation)。图 2 所示的模式是二级地址模式，如 INTEL x86 处理器上可用的虚拟寻址模式之一。该模式就其必须使用两级间接以将虚拟页标识符转换成物理页而言是“两级”的，如后文所描述的。

在该分页模式中，页目录 202 包含一组条目。条目的一个示例结构在下文结合图 3 更详细地描述，但是本质上，每一条目标识了具体页表，如页表 204(1)、204(2)或 204(3)的物理位置（如，页帧号或“PFN”）。每一页表进而包含一组条目，每一条目标识具体数据页，如页 206(1)、206(2)、206(3)或 206(4)的物理位置（还是页帧号）。数据页是确定长度的 RAM 132 的邻近部分。数据页可储存任一类型的数据，并且应当注意，除储存普通数据之外，数据页也可用于储存页目录 202 和页 204(1)到 204(3)的内容。由此，给定的页可以是目录、表/数据页或扮演作为这三个结构的组合的多重角色。

图 2 描述的虚拟地址模式是两级虚拟地址模式，因为需要通过页目录（级 1）和页表（级 2）以查找具体页。本领域的技术人员可以理解，可以用任意数量级来设计虚拟地址系统，并且本发明的原理可应用到所有这类虚拟地址模式。如本领域所已知的，IINTEL x86 处理器支持具有一、二或三级的虚拟地址，并且通常采用一种“混合”模式，其中“小”页（即，长度为 4 千字节的页）使用两级虚拟地址，“大”页（即，长度为 4 兆字节的页）使用一级虚拟地址。

在图 2 的分页模式中，页上的一个字节可由虚拟地址 210 来标识，包括页目录偏移 211、页表偏移 212 和页偏移 213。由此，为查找一物理地址，执行地址转换的存储器管理单元（MMU）220 使用页目录偏移 211 来查找页目录 202 中具体条目。例如，偏移 211 可等于零，指示应当参考页目录 202 中的第零个条目。该条目包含储存页表的 PFN，因此 MMU 20 使用该 PFN 来查找页表之一（如，页表 204(1)）。MMU 220 然后使用页表偏移 212 作为到标识的页表的索引，并检索在该偏移处找到的条目。该条目包含数据页（如页 206(1)）的 PFN，因此 MMU 220 将页偏移 213 添加到标识的页的基础地址，以查找物理存储器的具体字节。除纯粹地址转换之外，MMU 220 也可适用于执行各种其它功能：如，如果表中的页条目被标记为“不存在”，则 MMU 220 可从盘上加载该页；如果该页被标记为“只读”，

则 MMU 可不允许写访问，等等。

在图 2 的虚拟地址模式中，页目录本身的位置（即，PFN）储存在存储位置 201。当开始转换虚拟地址 210 时，MMU 220 使用存储位置的内容来查找页目录 202。由此，可以存在多个页映射，并且可通过设置存储位置 201 的内容包含给定映射的页目录的 PFN 来选择具体的映射用于当前使用。在 INTEL x86 处理器的示例中，存储位置 201 对应于名为 CR3 的寄存器。

高保证环境和通过地址转换控制的遮蔽存储器

可以理解，上文描述的虚拟地址系统的一个特征是可能有在给定地址转换映像下未由任意虚拟地址映射的物理地址。由此，在几乎所有的访问请求都由虚拟地址作出的诸如 INTEL x86 处理器等系统中，有可能通过确保给定源的地址转换映像不指向界限外的存储器来令物理存储器的一部分在该源的界限外。尽力在地址转换映像上进行控制来实现这一界限外存储器被称为地址转换控制（ATC）。存储器的界限外部分有时被称为“遮蔽存储器（curtained memory）”，并且 ATC 是实现遮蔽存储器的一种方法。

遮蔽存储器的一种用途是在高保证环境和非高保证环境在同一机器上共存时。高保证环境可具有不能被非高保证环境访问的遮蔽存储器。高保证环境的概念—及其对遮蔽存储器的关系—在下文参考图 3 描述。

图 3 示出了在同一机器上共存的两个环境：高保证环境 360 和非高保证环境 350。为方便起见，这些环境分别被称为右侧（RHS）和左侧（LHS）；RHS 360 是高保证的，而 LHS 350 是非高保证的。对于一个环境是“高保证的”意味着能够高度确保在该环境中执行的功能将被正确地执行。由此，LHS 350 执行各种功能 306(1)、306(2)、…、306(n)，并且 RHS 执行各种功能 308(1)、308(2)、…、308(m)。LHS 350 与描述功能 306(1)到 306(n)如何表现的规范 302 关联。同样，RHS360 与描述功能 308(1)到 308(m)如何表现的规范 304 关联。可以书写或不书写规范 302 和 304。例如，LHS 350 可以是与解释其各种服务、驱动器、系统调用等如何表现的书面手册一起供给的商用操作系统。或者，可以仅有对给定环境如何表现的一般理解，并且该理解可构成规范。

不论规范采用什么形式，可以理解，几乎所有的软件包含程序错误、非法途径（backdoor）（已知的或未发现的）、逻辑错误等等，它们可导致软件以非预期

的方式表现。然而，可以评估给定的软件将以期望的方式表现的保证度－即，软件的行为实际上将与其规范中所描述的匹配的保证度。在图 3 的示例中，RHS 360 在具有设计 RHS 360 来执行的功能 308(1)到 308(m)实际上将依照规范 304 执行的相对高保证度的意义上是“高保证的”。同样，LHS 350 在具有 LHS 350 将依照规范 302 执行功能 306(1)到 306(n)的相对低保证度的意义上是“非高保证的”。在该上下文中，“相对高”和“相对低”意味着 RHS 360 将依照规范 304 表现的保证度高于 LHS 350 将依照规范 302 表现的保证度。

一般的情况是，诸如 MICROSOFT WINDOWS 操作系统等具有全特征的商用操作系统是非高保证环境。这类环境支持可自由添加设备驱动器、插件和其它类型的扩充的开放体系结构，这令所有可想像的情况下这一操作系统的行为变得很难核实。因此，当期望某一类型的安全性时，与高保证操作环境并排地运行这一具有完整特征的操作系统是有用的。高保证操作系统是提供少量特征的小型操作系统。由于高保证操作系统提供的功能是较小的，有较少的变量影响其操作，因此在高度确定性上，其行为更容易核实。

在一个较佳的实施例中，高保证环境包括遮蔽存储器－即，对非高保证环境不可访问的存储器的一部分。由此，RHS 360 可在遮蔽存储器中储存机密数据（如，密码密钥），而不会有被运行在 LHS 350 中的进程读或写的危险。例如，规范 304 可规定 RHS 360 能够保护信息免遭外部干预，并且遮蔽存储器允许 RHS 360 执行这一功能。此外，RHS 360 用于执行其各种功能的代码可储存在遮蔽存储器中，以防止运行在 LHS 350 中的进程用不同的代码来重写该代码（这将导致 RHS 360 的行为超出其规范）。图 3 示出了可由 RHS 360 使用的遮蔽存储器 312。物理地址空间 310 包括给定计算装置上可用的所有物理存储器位置。物理遮蔽存储器 312 是这些位置的子集。如图 3 所示，RHS 360 可对所有物理地址空间 310 进行访问，但是 LHS 350 缺乏对构成遮蔽存储器 312 的物理地址空间 310 部分的访问。（应当注意，尽管图 3 示出地址空间的遮蔽和非遮蔽部分的分别是连续的，但对这一连续性没有需求。此外，RHS 360 不必具有对整个物理地址空间的访问，或者 LHS 350 不必具有对处于遮蔽存储器 312 之外的物理地址空间的每一部分的访问。）

如上所述，在某些系统中，几乎所有的存储器访问请求都由物理地址作出。在这一系统上实现遮蔽存储器 312 的一种方法是以不向 LHS 350 展现遮蔽存储器 312 的任何虚拟地址的方式控制地址转换映像的内容。（当某一类型的访问请求有

可能按照其物理地址标识其目标时，可由某一辅助协作机制来限制对遮蔽存储器的访问，如过滤来自直接存储器访问设备或来自其它源的访问请求的排除矢量。）有若干算法可用来确保 LHS 350 无法使用将通向遮蔽存储器 312 的地址转换映像，但是这些算法背后的中心思想是：（1）当处理器在 LHS 350 中操作时，确保加载到 CR3（图 2 所示的存储位置 201）中的任一映射不通向包含在遮蔽存储器 312 内的页；以及（2）对于编辑 LHS 350 中的活动映射的任一尝试，对所提出的编辑进行评估以确保该编辑将不产生到遮蔽存储器 312 的页的链接。

以下是用于通过 ATC 实现遮蔽存储器的示例算法：

设 D1 是可用作页目录的页组。设 D2 是可用作页表的页组。设 D 是 D1 和 D2 的并集（即，D = D1∪D2）。页目录或页表中被标记为“存在”（即，其存在位被设置）的每一条目被称为“链接”。如果有从 D1 中的某一页到所讨论的 D2 的小读写链接，则 D2 中的页是“写活动”的。（“小”链接是从目录到表的链接－即，目录中最终将通向小页的链接。“大”链接是目录中指向大页的链接。）假定有定义某一实体被准许对其读和/或写访问的页的政策。

维护以下不变式（invariant）：

－CR3 在 D1 内；

－在相关政策下所有 D1 和 D2 页是可读的；

－每一小链接从 D1 页指向 D2 页；

－D2 页的链接指向政策下可读的页；

－写活动 D2 页的每一读写链接指向相关政策下可写但不在 D 中的页；

－包含在 D1 页的大链接的大页目标中的每一小页在政策下是可读的；如果该链接是读写，则该小页在政策下也是可写的，并且不在 D 中。

例如，如果政策将遮蔽存储器 312 定义为不可访问，则为可由 LHS 350 使用的所有地址转换表维护上述不变式将确保没有安全虚拟地址标识其目标并在 LHS 350 中出现的访问请求会到达遮蔽存储器 312。

结合 ATC 使用转换后备缓冲器

转换后备缓冲器（TLB）本质上是储存了映射数据的高速缓存。TLB 背后的基本思想是最近被访问的页可能再次被访问，因此最近使用过的从给定虚拟页到对应的物理页的映射被储存在 TLB 中。从地址转换表计算映射（至少以 INTEL x86

处理器上采用的最常见二级虚拟地址模式）需要两次存储器访问：一次访问页目录来找出相关页表的位置，然后第二次访问页表来找出相关数据页。仅可在执行了这两次（其它）存储器访问来找出目标数据的物理位置之后才可找寻到要访问的实际目标位置。TLB 通过提供维护最近使用过的映射信息的快速存储器减少了这一额外的存储器访问的需求。由此，对于需要执行的每一地址转换，处理器首先参考其 TLB 来确定给定虚拟页的映射数据是否高速缓存在 TLB 中。如果映射数据不在 TLB 中，则参考地址转换表。

图 4 示出了处理器如何使用 TLB 402。处理器 102 执行代码片断，并且在执行过程中，引发读或写存储器 400 的目标位置 406 的指令。该指令不标识目标位置 406 的物理地址，而是指定了目标位置 406 的虚拟地址。由此，处理器 120（或，在许多情况下，与处理器 120 关联的存储器管理单元）必须将虚拟地址转换成物理地址。处理器 120 首先在 TLB 402 中查找以确定是否高速缓存了可在虚拟地址转换中使用的映射（圈 1）。如果存在这一映射，则使用该映射来转换地址。如果不存在这样的映射，则从存储器 400 读取地址转换映像的相关部分（圈 2），并且使用这些映射部分来转换虚拟地址。不论地址转换是如何进行的，使用转换的地址来访问物理目标位置 406（圈 3）。TLB 能够提供效率的原因是在 TLB 402 中查找信息比在存储器 400 中读地址转换映像 404 要快。另外，在一些设计中，储存在 TLB 402 中的数据可将两步转换过程压缩成一步：即，使用地址转换映像需要在以不同的步骤来读页目录和页表；然而，单个的页目录（PD）偏移/页表（PT）偏移组合可以高速缓存在 TLB 402 中，由此允许在单个步骤中由 PD/PT 偏移组合来标识虚拟页。

当在 ATC 系统中使用 TLB 时，对 ATC 不变式进行修改以把通过 TLB 可用的转换考虑进来。例如，在先前描述的示例 ATC 算法中，链接的定义可被修改如下：如果从某一页到另一页的链接在物理存储器中存在或者高速缓存在某一 TLB 中，则该链接存在。尽管在大多数体系结构中（如，x86 处理器）TLB 的内容不是直接可观察的，然而可绑定其内容，因为转换仅通过存储器进入 TLB。使用示例算法中的不变式，则必须在从 D1 或 D2 移除一页时、放弃对页的读或读写访问时以及在对 D2 作出将页的状态从写活动改为写非活动或从写非活动改为写活动的写或添加时，转储清除 TLB。

然而，有充分的激励来进一步减少 TLB 转储清除，因为转储清除 TLB 是减慢

性能的昂贵操作。首先，TLB 对效率改进的程度仅止于用有用的映射数据来填充它、且转储清除 TLB 从 TLB 中消除了这一映射数据而已。第二，由于以下原因，实际上转储清除本身可能是昂贵的：在图 3 描述的体系结构中，遮蔽存储器 312 的维护取决于计数器政策映射的缺乏。在一个较佳的实施例中，由于计数器政策映射的缺乏是提供高保证环境的元素，因此，对映射的控制本身必须在高保证环境中执行。由此，不仅地址转换映像的控制本身必须在高保证环境中执行，而且 TLB 的转储清除也必须在高保证环境中执行（或至少被核实）。因此，作为 ATC 系统的一部分被依赖的每一 TLB 转储清除需要当前环境从 LHS 350 改为 RHS 360。将上下文从当前环境改变到另一环境本身是昂贵的，它增加了转储清除的开支。因此，期望尽可能地避免转储清除 TLB。

本发明提供了一种延迟并尽可能避免某些此类转储清除的机制。在一个实施例中，本发明基于两种子机制。第一种子机制规定那些不修改实际存储器映射的操作（如，向 D2 添加页，或令页对写不可访问）可在概念上被延迟，直到需要其完成来准许对存储器的某一修改。例如，放弃对页的读访问（在安全政策中）的行动可被延迟，直到该页可能被修改（当另一计算实体创建对该页的读写映射时）。第二种子机制使用时间戳来更精确地跟踪到页的特定映射是否在特定的 TLB 中存在。

在先前所呈现的示例算法中，需要转储清除的可延迟操作可被划分成两个类：需要从 TLB 移除允许读页（放弃对页的读访问或从 D1 或 D2 移除页）的任一转换的那些操作，以及需要从 TLB 移除允许修改页（移除对页的写访问）的任一转换的那些操作。调用前一“读转储清除操作”和后一“写转储清除操作”，这些操作可被如下延迟：

- 读转储清除操作可被延迟直到某一计算实体示图创建到该页的读写映射，或直到某一设备可能通过直接存储器访问修改该页，或直到由 ATC 算法本身修改该页；
- 写转储清除操作可被延迟直到某一计算实体试图创建到该页的读或读写映射，或直到某一设备可能通过直接存储器访问修改该页，或直到由 ATC 算法本身修改该页。

当执行读转储清除操作时，仅需在 TLB 仍潜在地包含到所讨论的页的读映射时转储清除该 TLB。唯一的这种情况是自从最后一次有对该页的读映射以来 TLB

未被转储清除过。类似地，当执行写转储清除操作时，仅需在自从最后一次有对该页的读写映射以来 TLB 未被转储清除过时转储清除 TLB。

为确定自从有对页的读/读写映射以来 TLB 是否被转储清除，为每一页维护两个时间戳（其读时间戳和写时间戳），并且对每一 TLB 维护一个时间戳（其转储清除时间戳）。只要移除了到页的最后一个读/读写映射，在其读/读写时间戳中记录当前时间。只要转储清除了 TLB，在其转储清除时间戳中记录当前时间。

可以若干种方式测量时间。在第一种示例方法中，用于 TLB 的时钟仅是转储清除时间戳。由此，从 TLB 的转储清除时间戳复制页的读和读写时间戳。当转储清除 TLB 时，转储清除时间戳改变（如何改变并不重要）。如果页的读/读写时间戳与转储清除时间戳不同，则确保读/读写映射不在 TLB 中。如果碰巧转储清除时间戳改变（可能改变了若干次），而恰巧与读/读写时间戳符合（如，由于时钟的绕回），不产生任何损害；只是会有一次不必要的转储清除。

在第二种示例方法中，用实时或虚拟时钟来测量时间。如果确保用于记录读/读写时间戳和转储清除时间戳的时钟在总是某一时滞内一致，则如果读/读写时间戳超前转储清除时间戳至少该时滞，则确保读/读写映射不再 TLB 中。完全避免时滞的一个方法是使用全局时钟，一个当转储清除时被递增的共享计数器；然而，这类时钟可能变成一个热点（hotspot）。

第一种方法具有不需要同步时钟的优点。然而，如果要考虑大量的 TLB，则第一种方法对每一 TLB 需要一个单独的读/读写时间戳，而第二种方法仅需要读单个时钟。此外，它消除了时钟本身的存储器争用。也可组合这两种方法，并且也可采用其它机制（如，矢量时钟）。

本发明提供了一种允许 TLB 的惰性转储清除的机制，使得 TLB 的显式转储清除被延迟直到有 LHS 350 实际上使用了不再符合地址转换映像的旧 TLB 条目访问页的潜在可能。本质上，当移除了对页的最后一次映射时，该页被“加时间戳”，（如下文所讨论的，“时间戳”可能实际上不包含时间，而是顺序计数器的值。）当随后访问页时，根据自从页被加时间戳以来 TLB 是否被转储清除来确定是否要转储清除 TLB。

在图 5 所示的多处理器系统中，处理器 120(1)到 120(n)的每一个维护其子集的 TLB 402(1)到 402(n)。换言之，处理器 120(1)到 120(n)的每一个可访问同一基础地址转换表，但是它们可以单独从这些表高速缓存映射。另外，在一个实施例中，每

一处理器可在 LHS 350 或 RHS 360 中操作。依照本发明，计数器 502(1)到 502(n)与处理器 120(1)到 120(n)的每一个关联。每一次处理器上的环境从 RHS 360 改变到 LHS 350 时，递增与该处理器关联的计数器。

每当令页对 LHS 350 不可用（即，当在相关访问政策下令页对 LHS 350 不可用、并将其从对 LHS 350 可用的所有映射解除链接），则有效地用计数器值为该页“加时间戳”。（计数器不在物理意义上计算“时间”，而可以被视为一种类型的计时，因为计数器响应于某一事件的出现而前移。）由此，通过将页的储存的计数器值与试图访问该页的处理器的当前计数器值相比较，可确定计数器是否在页对 LHS 350 之后进入了 RHS 360。（如上文所解释的，进入 RHS 360 促使 TLB 被可靠地转储清除。）如果处理器在页被解除链接之后未进入 RHS 360，则处理器进入 RHS 360 并转储清除其 TLB。如果在 LHS 350 中出现访问该页的尝试，则处理器返回到 LHS 350 并再次试图访问该页。如果处理器在页被解除链接之后进入了 RHS 360（即，如果处理器当前在 RHS 360 中，或在页被解除链接之后的某一点进入 RHS），则已知自从该页最后一次对 LHS 350 可用以来已可靠地转储清除了 TLB，并且由此 TLB 不能包含到该页的任何映射，因此在那一时刻不需要转储清除 TLB。

图 6 所示是在令页变得对 LHS 350 不可用之后转储清除 TLB 的过程的流程图。在图 6 所示的过程的开始，假定了在一种适用访问政策下可从 LHS 350 访问给定页（页 X）。在某一点，在该访问政策下令页 X 对 LHS 350 不可用，并且根据需要调整对 LHS 350 可用的地址转换映像以确保页 X 从那些映射中解除链接（602）。在页 X 被解除链接时，记录机器上所有处理器的计数器值，并且该记录与页 X 关联用于稍后的检索（604）。该记录较佳地包含机器上每一处理器的当前计数器值。

在从对 LHS 350 可用的映射解除链接页 X 之后，可以在过去某些时间后再实际访问页 X 。然而，在某一点上，对页 X 的访问请求将始发于一特定处理器（处理器 Y）（606）。然后检索与该页关联的记录的计数器值，并将处理器 Y 的储存值与处理器 Y 的当前计数器值相比较。（该记录还包含其它处理器的储存的计数器值，但是在确定处理器 Y 是否需要转储清除其 TLB 时丢弃这些其它值）。如果处理器 Y 的当前计数器值大于储存的计数器值，则准许对该页的访问（614）而不需要进一步的询问，因为处理器 Y 肯定在页 X 对 LHS 350 不可用之后已进入了 RHS 360 并转储清除了其 TLB。另一方面，如果处理器 Y 储存的计数器值与处理器 Y 的当前计数器值相等，则在页 X 解除链接之后尚未进入 RHS 360，这意味着处理

器 Y 的 TLB 可能仍包含到页 X 的映射。

如果处理器 Y 运行在 RHS 360 中（610），则 TLB 可在不违反访问政策的情况下包含到页 X 的映射，因此准许对页 X 的访问（614）。然而，如果处理器 Y 运行在 LHS 350 中（并且从 608 已知处理器 Y 在页 X 被解除链接之后尚未转储清除其 TLB），则有处理器 Y 的 TLB 的内容将向 LHS 350 展现到页 X 的映射的可能。由此，处理器 Y 进入 RHS 360 并转储清除其 TLB（612）。在 TLB 被转储清除之后，处理器 Y 返回到 LHS 350，并重新执行访问请求（614）（由此需要用新的空 TLB 重新转换请求地址。）

应当注意，令页对 LHS 350 不可访问不是可触发转储清除 TLB 的需求的唯一事件，并且本发明的机制也可用于触发由于其它原因而引起的 TLB 转储清除。例如，从组 D1 或 D2（如上文所定义的）之一移除页，或放弃对页的读访问（或写访问），也可引发 TLB 转储清除的需求，因为这些事件会影响什么映射在相关政策下是合法的。在这一情况下，本发明的机制可以类似于上文描述的方式来使用一如，当从 D1 或 D2 移除页时，或当放弃读（或写）访问时，可记录计数器值，并且处理器访问页的任一随后的尝试可触发处理器的当前计数器值和该页的储存的计数器值之间的比较。

还应当注意，本发明的机制不限于转储清除需要到 RHS 360 的行程的情况一或甚至是 LHS 350 和 RHS 360 都可用的情况。更一般而言，本发明的机制可在有过时条目在不再反映映射的状态的 TLB 中存在的可能的任一情况下使用一如，在由操作系统执行的普通进程隔离的情况下。

注意，以上示例仅为解释目的提供，并且不应被解释为对本发明的局限。尽管参考各种实施例描述了本发明，可以理解，此处所使用的词语是描述和说明的词语，而非限制的词语。此外，尽管此处参考具体的方法、材料和实施例描述了本发明，本发明不意味着对此处所揭示的特性的限制；相反，本发明延及所有功能等效的结构、方法和使用，如处于所附权利要求书的范围之内的。从本说明书的教导获益的本领域的技术人员可在不脱离本发明各方面的范围和精神的情况下作出各种修改和变化。

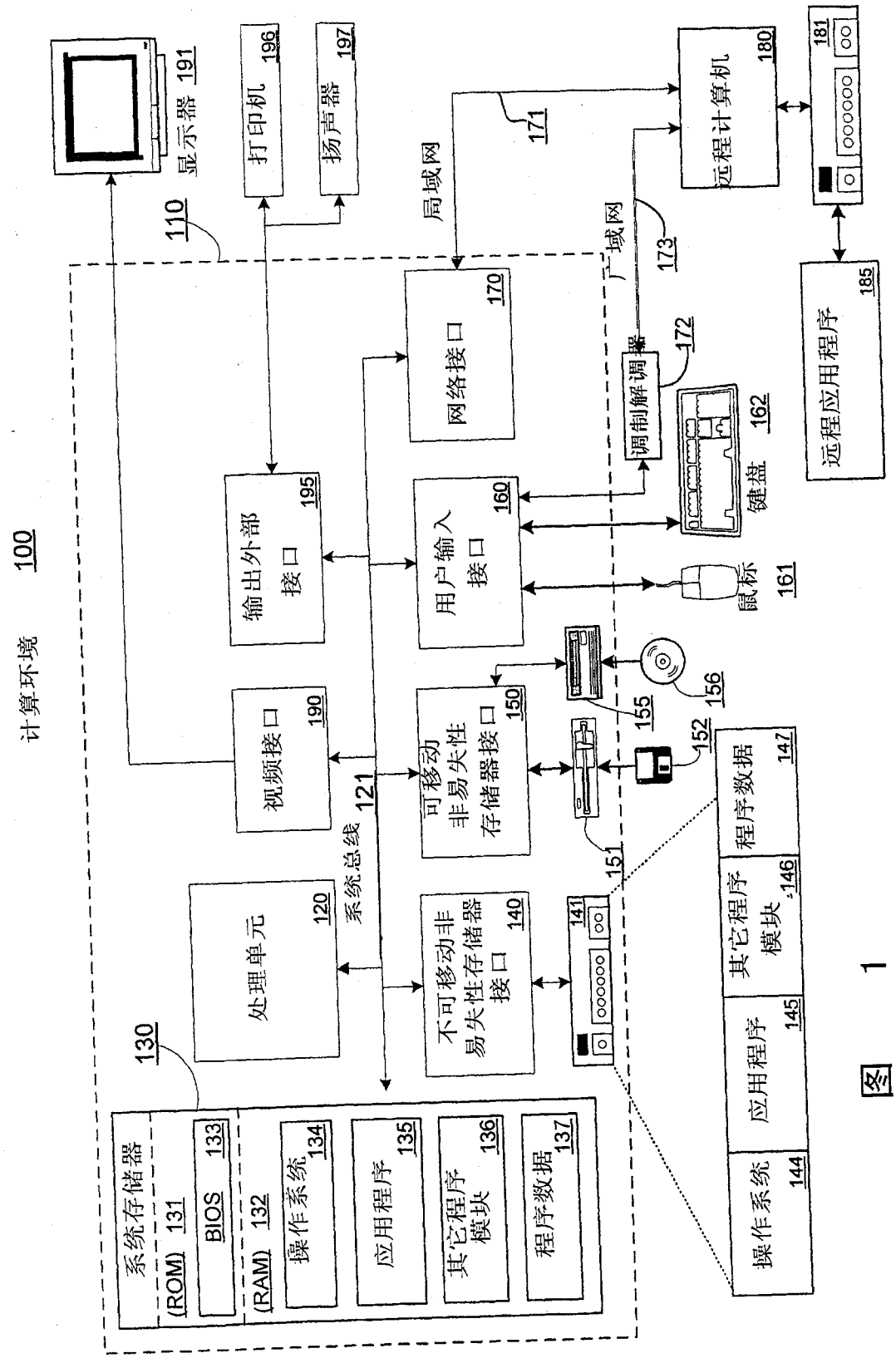

计算环境 100
110
显示器 191
打印机 196
扬声器 197
局域网 171
广域网 173
远程计算机 180
181
远程应用程序 185
网络接口 170
调制解调器 172
输出外部接口 195
用户输入接口 160
键盘 162
鼠标 161
视频接口 190
可移动非易失性存储器接口 150
处理单元 120
系统总线 121
不可移动非易失性存储器接口 140
130
系统存储器
(ROM) 131
BIOS 133
(RAM) 132
操作系统 134
应用程序 135
其它程序模块 136
程序数据 137
141
151
152
155
156
操作系统 144
应用程序 145
其它程序模块 146
程序数据 147

图 1

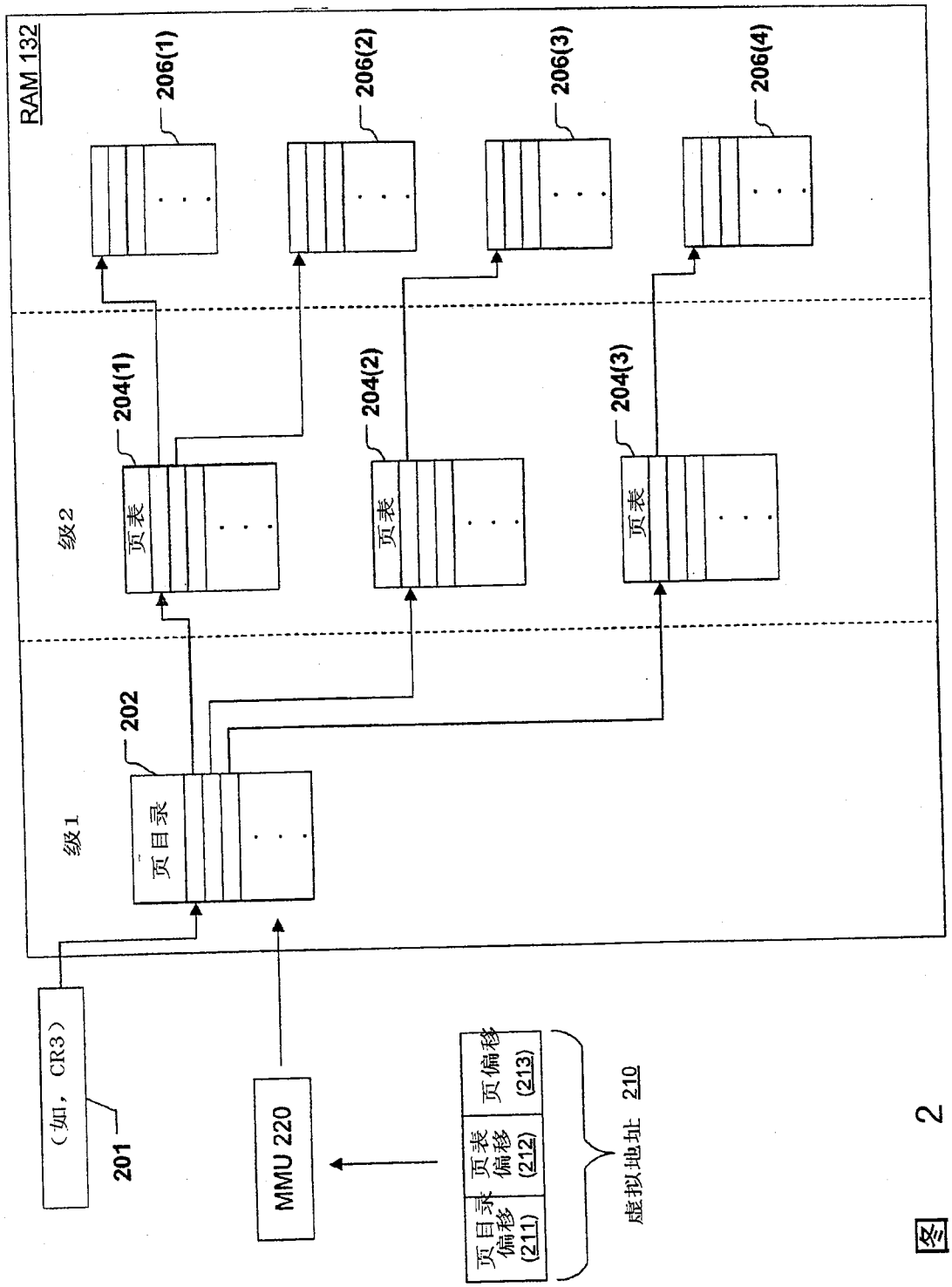

RAM 132
206(1)
206(2)
206(3)
206(4)
204(1)
204(2)
204(3)
级2
页表
202
级1
页目录
（如，CR3）
201
MMU 220
页目录偏移 (211)
页表偏移 (212)
页偏移 (213)
虚拟地址 210

图 2

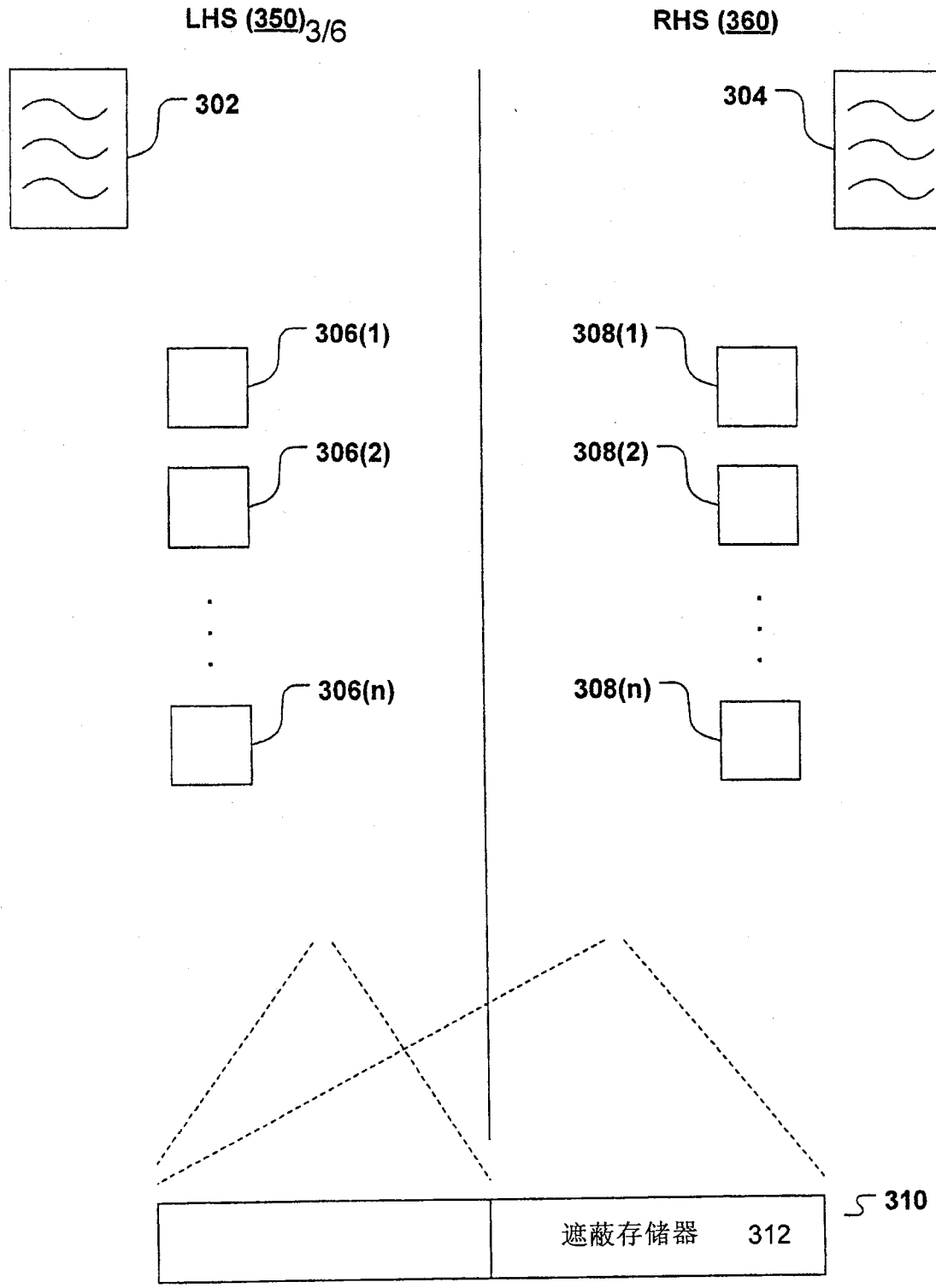

LHS (350)3/6
RHS (360)
302
304
306(1)
306(2)
306(n)
308(1)
308(2)
308(n)
遮蔽存储器 312
310

图 3

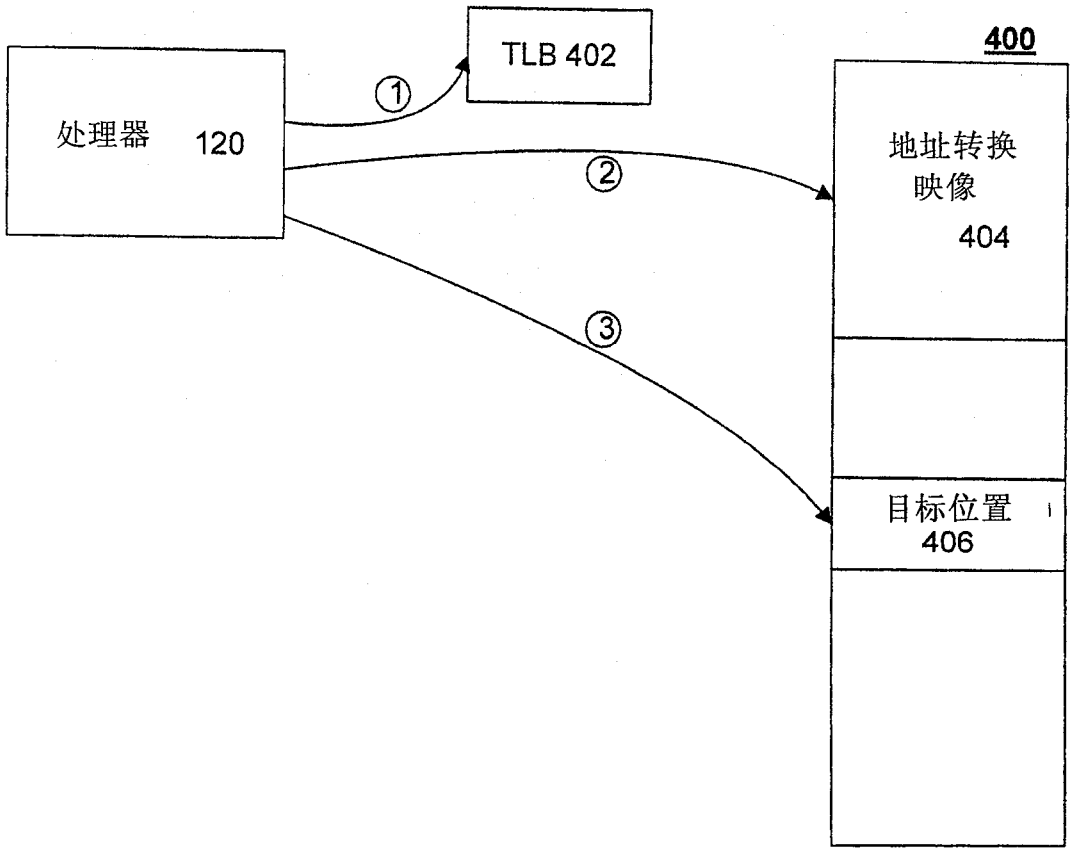


图 4

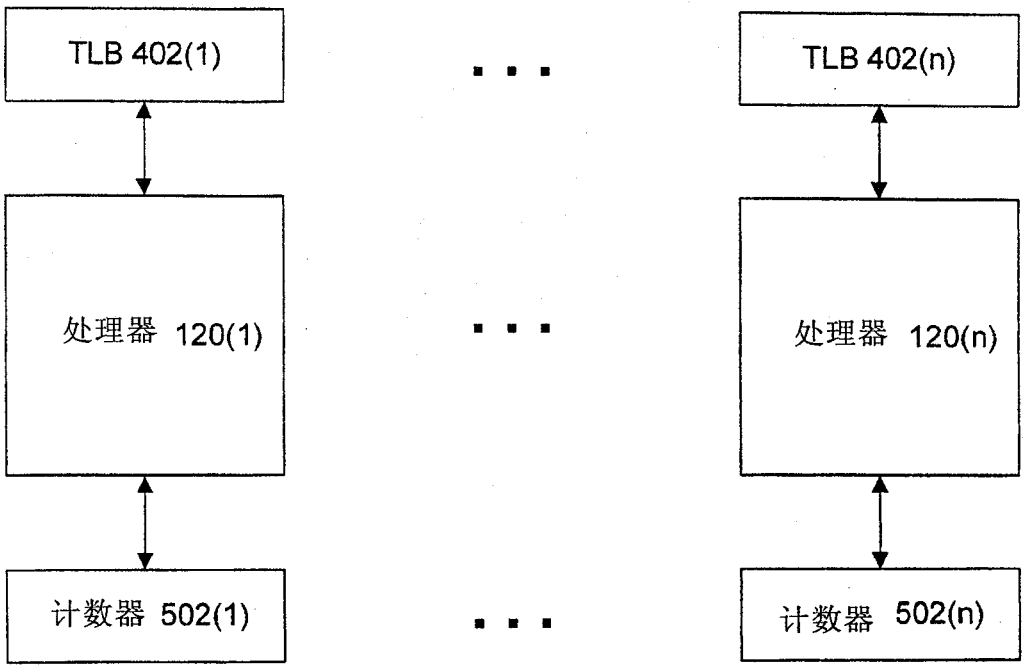


图 5

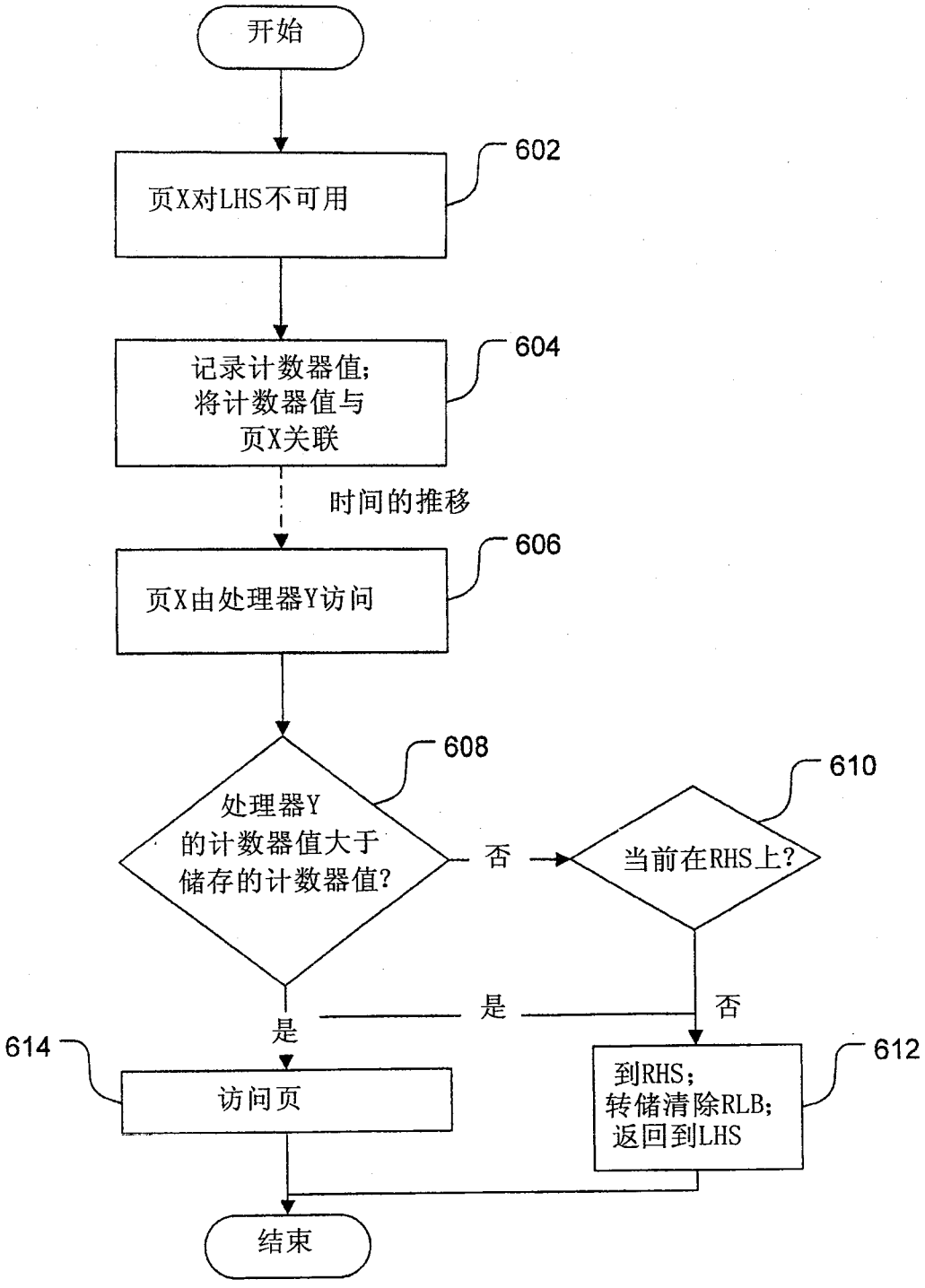

开始
602
页X对LHS不可用
604
记录计数器值;
将计数器值与
页X关联
时间的推移
606
页X由处理器Y访问
608
处理器Y
的计数器值大于
储存的计数器值?
否
610
当前在RHS上?
是
否
是
614
访问页
到RHS;
转储清除RLB;
返回到LHS
612
结束

图 6