

[12] 发明专利申请公开说明书

[21] 申请号 02108306.1

[43]公开日 2002 年 11 月 13 日

[11]公开号 CN 1379334A

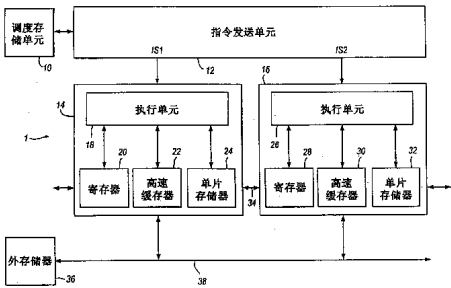
[22] 申请日 2002.3.28 [21] 申请号 02108306.1
[30] 优先权
[32]2001.3.30 [33]GB [31]0108106.6
[71] 申请人 斯罗扬有限公司
地址 英国伦敦
[72] 发明人 尼格尔·皮特·托法姆
林寿川

[74] 专利代理机构 中国国际贸易促进委员会专利商标事
务所
代理人 蒋世迅

权利要求书 3 页 说明书 18 页 附图 7 页

[54] 发明名称 地址转换
[57] 摘要

公开一种处理器,和一种访问该处理器中存储器的方法。该处理器为访问物理存储器而产生能变换为物理地址的虚地址,该物理存储器包括第一存储器部分(101)和第二存储器部分,第二存储器部分与第一存储器部分一样,是同一存储器级的一部分。当产生一虚地址时,该虚地址的一部分被变换为一局部的物理地址,并用该局部物理地址访问该第一存储器部分(101)内的存储器单元。与该存储器的访问并行,可以进行检验,以确定该局部物理地址是否正确。



I S S N 1 0 0 8 - 4 2 7 4

1. 一种为访问物理存储器而产生能变换为物理地址的虚地址的处理器，该物理存储器包括第一存储器部分和第二存储器部分，该两个存储器部分都是同一存储器级的一部分，该处理器包括：

产生一虚地址的产生装置；

把该虚地址的一部分变换为一局部的物理地址的变换装置；和
用该局部物理地址访问第一存储器部分中存储器的访问装置。

2. 按照权利要求1的处理器，其中，该局部物理地址足以唯一地识别第一存储器部分内的存储器单元，但不是整个物理存储器内的存储器单元。

3. 按照权利要求1或2的处理器，其中的地址包括帧号码和偏移，该偏移对相应的虚地址和物理地址是相同的，并且其中的局部物理地址包括一物理帧号码的一部分和偏移。

4. 按照权利要求3的处理器，其中的变换装置包括查阅表，查阅表中的表目包括一虚地址的一部分帧号码和该物理地址的一部分帧号码。

5. 按照前述权利要求中任一项的处理器，还包括检验装置，用于检验访问装置访问的存储器中，与该产生装置产生的虚地址对应的物理地址。

6. 按照权利要求5的处理器，还包括分配装置，用于响应检验装置的输出，把虚地址分配至第一存储器中的物理地址。

7. 按照权利要求6的处理器，其中的分配装置要能把虚存储器的一页分配至第一存储器部分中物理存储器的一页。

8. 按照权利要求7的处理器，其中的分配装置要能实现，只要两页虚页的地址中，地址比特的某一给定子集相同，则它们不能同时被映射到第一存储器部分。

9. 按照前述权利要求6至8中任一项的处理器，还包括当虚地址被分配至物理地址时，用于更新变换装置的装置。

10. 按照前述权利要求 6 至 9 中任一项的处理器，还包括当虚地址被分配至物理地址时，用于更新检验装置的装置。

11. 按照前述权利要求 5 至 10 中任一项的处理器，其中的检验装置包括用于把该局部物理地址变换为一虚地址的至少一部分的装置，和用于把该虚地址的至少一部分与产生装置产生的虚地址比较的装置。

12. 按照权利要求 11 的处理器，其中该变换局部物理地址的装置包括查阅表，查阅表中的表目包括一局部物理帧号码和一虚帧号码。

13. 按照前述权利要求 5 至 12 中任一项的处理器，其中的检验装置包括把产生装置产生的一虚地址变换为一物理地址的至少一部分的装置，和把该物理地址的至少一部分，与访问装置访问的存储器中物理地址比较的装置。

14. 按照权利要求 13 的处理器，其中变换虚地址的装置包括查阅表，查阅表中的表目包括一虚帧号码和对应的物理帧号码。

15. 按照前述权利要求 5 至 14 中任一项的处理器，其中，只要检验装置确定，从存储器读出的某一存储器物理地址，与产生装置产生的虚地址不对应，则该处理器要能忽略或废弃从存储器读出的数据。

16. 按照前述权利要求中任一项的处理器，还包括确定产生装置产生的虚地址，是否在第一存储器部分的地址范围内的装置，以及如果该虚地址在该范围外，则禁止访问第一存储器部分的装置。

17. 按照前述权利要求中任一项的处理器，其中该处理器按相继的时钟周期工作，而变换装置按与产生虚地址的产生装置相同的时钟周期，把一虚地址的一部分变换为该局部的物理地址。

18. 按照权利要求 5，或权利要求 6 至 17 中与权利要求 5 有关的任一项的处理器，其中，该处理器按相继的时钟周期工作，同时，检验装置检验访问装置访问的存储器物理地址，是否与产生装置产生的虚地址对应，该产生装置按与访问装置访问第一存储器部分的存储器相同的时钟周期产生虚地址。

19. 按照前述权利要求中任一项的处理器，其中该处理器是流水线

处理器，且访问存储器的装置是该处理器流水线的一部分。

20. 一种单片系统设备，包括：

按照前述权利要求任一项的处理器，所述第一存储器部分包括在该单片系统设备内；和

一种访问所述第二存储器部分的接口，所述第二存储器部分由该设备的外部提供。

21. 按照权利要求 20 的设备，其中第一存储器部分是单片 RAM，例如同步 RAM。

22. 一种访问处理器中存储器的方法，该处理器为访问物理存储器而产生能变换为物理地址的虚地址，该物理存储器包括第一存储器部分和第二存储器部分，该两个存储器部分都是同一存储器级的一部分，本方法包括的步骤有：

产生一虚地址；

把该虚地址的一部分变换为一局部的物理地址；和

用该局部物理地址访问第一存储器部分的存储器。

23. 按照权利要求 22 的方法还包括，在访问步骤中，检验访问的存储器的物理地址，是否与产生步骤中产生的虚地址对应。

地址转换

技术领域

本发明涉及结合处理器使用的地址转换技术，该处理器能访问各独立的存储器部分，如单片存储器和外存储器，且该处理器用虚存储器工作。本发明在“单片系统(system-on-chip)”设备中有特殊应用，在该“单片系统”设备中，在同一集成电路内提供一处理器和一定量的存储器。

背景技术

单片系统设备，通常包括由单个半导体集成电路或芯片提供的处理器和一定量的随机存取存储器(RAM)。因为单片存储器能够提供的存储量有限，同一存储器级的更多的存储器可以由外部向该芯片提供，该处理器也能访问外部提供的存储器。

如果该处理器利用虚存储器工作，那么由处理器产生的地址是虚地址，不直接对应于处理器能访问的存储器的物理地址。可以用转换过程，把处理器产生的虚地址变换为物理地址。

要把虚地址转换为物理地址，从产生虚地址到访问相应存储器单元之间可能引起延迟。该延迟导致访问存储器时出现不希望的长的访问时间，反过来可能降低处理器的整体性能。

因此，需要降低从产生虚地址到访问对应于该地址的存储器区域之间的时间。具体说，需要降低访问特定存储器部分，例如单片存储器所用的时间，该存储器部分是处理器可访问的整个存储器的一子集。

发明内容

按照本发明，提供了一种处理器，用于为访问物理存储器而产生能变换为物理地址的虚地址，该物理存储器包括第一存储器部分和第二存储器部分，该两个存储器部分都是同一存储器级的一部分，该处理器包括：

产生一虚地址的产生装置；

把该虚地址的一部分变换为一局部的物理地址的变换装置；和
用该局部物理地址访问第一存储器部分中存储器的访问装置。

由于只需把该虚地址的一部分变换为一局部的物理地址，并用该局部物理地址访问第一存储器部分，所以与完整的转换相比，能够降低地址转换所用时间。这本身又能够加速对第一存储器部分的访问，改善处理器的整体性能。

该处理器的存储器可以有简单的一级排列，或者，该处理器的存储器可以排列为包括多级的分级结构，每一级保持该级下一级的数据的一子集。两种情形中不论哪一种，都通过把特定存储器级的存储器分成两个或多个存储器部分，例如把该种存储器部分视同单片存储器，能够使一个这些存储器部分成为一个快速访问的存储器部分。这样做是适当的，因为例如，存在防止把该存储器级的全部存储器由单片提供的物理制约。

最好是（在局部的转换是正确的情形），该局部物理地址足以唯一地识别第一存储器部分内的存储器单元，但不是全部物理存储器内的存储器单元。这样，在假定需要的物理地址确实在第一存储器部分内的情形下，就能够确定，被访问的第一存储器部分中的地址是正确的。

在一个实施例中，对虚存储器和物理存储器，都用存储器分页，从而，一个地址可以包括帧号码及偏移，对相应的虚地址和物理地址，该偏移是相同的。于是，该局部物理地址，可以包括该物理帧号码的一部分（局部物理帧号码）和偏移。在本实施例中，变换装置可以包括查阅表，而查阅表的表目可以包括一个虚地址的部分帧号码和该物理地址的部分帧号码。

在某些情形中，例如事先知道，该虚地址映射到第一存储器部分，或该虚地址只映射到第一存储器部分，那么，局部的转换就足以识别正确的物理地址，在此情形中无需采取另外的动作。但是，在大多数的情形中，某一虚地址可能映射到物理存储器的任何地方，且事先不

知道该虚地址映射到存储器的哪一部分。此时，需要进行检验，以确认（或否定）该局部的转换是正确的。因此，处理器还可以包括检验装置，用于检验被访问装置访问的某一物理地址是否与产生装置产生的虚地址对应。

如果检验装置确定，该局部的转换是正确的，那么无需采取校正动作。但是，如果检验装置确定该局部的转换是不正确的，那么需要采取适当的校正动作。该动作可以包括废弃或忽略访问装置读出的任何数据，从而，如果检验装置确定该局部的转换是不正确的，则处理器可以用于废弃或用于忽略从存储器读出的数据。

如果检验装置确定该局部的转换是不正确的（因为被访问装置访问的物理存储器地址不对应于产生装置产生的虚地址），则检验装置最好能提供该虚地址映射到第一存储器部分中物理地址的新的映射。据此，如果该虚地址被再次使用（且该映射还没有被其他映射到相同物理地址的映射取代（superceded）），那么对应的物理地址应在第一存储器部分。因此，处理器还可以包括分配装置，用于响应检验装置的输出，把某一虚地址分配至第一存储器部分的某一物理地址。特别有利的是，此时产生的地址显示时间的局限性，就是说，同一地址在短的时间周期内趋向于被多次产生。

在使用存储器分页的情形下，分配装置可用于把一页虚存储器分配给第一存储器部分的一页物理存储器。此时，如果将来有任何虚地址在该虚页内产生（且该映射还没有被取代），则对应的物理地址将在第一存储器部分。特别有利的是，此时产生的地址显示空间的局限性，就是说，相继的虚地址趋向于彼此接近。

分配装置要能实现，只要两页虚页的地址中，地址比特的某一给定子集相同，则它们不能同时被映射到第一存储器部分。通过把本约束加于页的分配上，能够把执行局部地址转换时要求的逻辑简化。例如，如果第一存储器部分的物理页数是 2^n （或少于），则预设的比特数可以是 n ，同时有 2^n 表目的 n -比特查阅表可以用于局部地址转换。

为确保将来的局部的转换是正确的，处理器还可以包括对变换装

置进行更新的装置，用于在一虚地址被分配给一物理地址时，对变换装置进行更新。例如，如果变换装置包括查阅表，则处理器可以包括把新表目装入该查阅表的装置。此外，为确保将来对局部的转换的检验能正确地执行，处理器还可以包括对检验装置进行更新的装置，用于在一虚地址被分配给一物理地址时，对检验装置进行更新。例如，如果检验装置包括查阅表，则处理器可以包括把新表目装入该查阅表的装置。

如有必要，当一虚地址被分配给第一存储器部分中一物理地址时，先前在该物理地址保持的数据，可以存储到别的地方，而与该虚地址对应的数据可以装入该物理地址。如果用存储器分页，则保持在该物理页中的数据，可以存储到别的地方，而与该虚页对应的数据可以装入该物理页。

检验装置可以包括把该局部的物理地址变换为一虚地址的至少一部分的装置，和把该虚地址的至少一部分与产生装置产生的虚地址比较的装置。例如，变换该局部物理地址的装置，可以包括一查阅表，并且，如果用存储器分页，则在该查阅表中的表目可以包括一局部的物理帧号码和一局部的虚帧号码。在需要时，可以直接标引该查阅表。

另外，检验装置可以包括把产生装置产生的一虚地址变换为一物理地址的至少一部分的装置，和把该物理地址的至少一部分与访问装置访问的存储器物理地址比较的装置。变换虚地址的装置可以包括一查阅表，该查阅表中的表目可以包括一虚帧号码和对应的物理帧号码。在需要时，可以关联地标引该查阅表。

访问装置用于从存储器读出数据。在此情形下，由于只有读出操作，如果检验装置后来确定该局部转换的地址是不正确的，那么，没有数据被破坏，同时可以简单地废弃或忽略不正确读出的数据。但是，在某些情形中，还可能要求写入操作。例如，如果已知被访问的存储器区域不包含任何有效数据（例如因最近被复位），于是可能要求写入操作。

处理器还可以包括确定产生装置产生的虚地址是否在第一存储

器部分的地址范围内的装置，以及如果该虚地址在该范围之外时，则禁止访问该第一存储器部分的装置。这样可以阻止非必要的或不需要的存储器访问的发生，该类装置还能使该局部的转换机构既用写入操作工作也用读出操作工作，因为如果虚地址不是映射到第一存储器部分，则将执行写入操作。

处理器可以按相继的时钟周期工作，而变换装置可以按与产生虚地址的产生装置相同的时钟周期，把一虚地址的一部分变换为该局部的物理地址。这样，对第一存储器部分的访问，能够在产生地址之后的时钟周期内发生，无需为产生完整的地址转换而等待更多的时钟周期。

处理器可以按相继的时钟周期工作，而检验装置可以按与访问装置访问第一存储器部分的存储器相同的时钟周期，检验访问装置访问的物理地址是否对应于产生装置产生的虚地址。因此，地址检验操作可以与存储器的访问操作并行地进行，所以检验操作只要求很少的时间或不要求附加的时间。

最好是，处理器是流水线处理器，且访问存储器的装置是该处理器流水线的一部分。如果是这样，访问第一存储器部分的路径能够插入处理器的流水线，这样可以提高访问第一存储器部分的速度。

处理器可以作为单片系统设备的一部分，形成在一芯片上，而第一存储器部分可以是单片存储器，同时第二存储器部分可以是外存储器。因此，按照本发明的第二方面，是提供包括上述任何形式处理器的一种单片系统设备，该设备还包括作为单片存储器的所述第一存储器部分，以及访问作为外存储器的所述第二存储器部分的接口。第一存储器部分可以是单片 RAM，例如同步 RAM。

本文还提供本发明的类似方法特征，因此，按照本发明的第三方面，是提供一种在处理器内访问存储器的方法，该处理器为访问物理存储器而产生能变换为物理地址的虚地址，该物理存储器包括第一存储器部分和第二存储器部分，该两个存储器部分都是同一存储器级的一部分，本方法包括的步骤有：

产生一虚地址；

把该虚地址的一部分变换为一局部的物理地址；和

用该局部物理地址访问第一存储器部分的存储器。

最好是，本方法还包括检验步骤，以检验访问步骤中访问存储器的物理地址是否与产生步骤中产生的虚地址对应。本方法最好还能提供与上述说明的处理器特性对应的其他特性。

上述功能至少有一些可以用软件完成，例如部分操作系统，因而本发明还提供的操作程序包括，为完成任何本文所说明的方法的各程序部分，以及计算机可读介质，在其上存储的程序包括为完成任何本文所说明的方法的各程序部分。

附图说明

现在将参照附图，纯粹以举例的方式，说明本发明的优选特性，其中：

图 1 是方框图，画出本发明具体实施的处理器的各部分；

图 2 画出把虚存储器映射到物理存储器的一个例子；

图 3 画出页表的表目（PTE）的一个例子；

图 4 画出用于单片存储器访问的各流水线段的未优化顺序；

图 5 画出两个虚地址；

图 6 是方框图，画出本发明第一实施例处理器的各部分；

图 7 至 13 画出图 6 在各种情形下的查阅表内容的各个例子；和

图 14 是方框图，画出本发明第二实施例的处理器各部分。

画在各图中的功能块既可以用硬件，也可以用软件，或用两者的组合实现。

具体实施方式

图 1 是方框图，画出本发明具体实施的处理器。该处理器是单片系统设备的一部分，在该设备中的处理器，连同一定量的本地存储器，设在单一芯片（集成电路）上。在本例中，该处理器是一指令字非常长的处理器，用于执行可分为较小指令的长指令。

参考图 1，处理器 1 包括调度存储单元 10、指令发送单元 12、和

第一及第二计算单元 14、16。指令发送单元 12 有两个发送时隙 IS1、IS2，分别与计算单元 14、16 连接。计算单元 14 包括执行单元 18、一组寄存器 20、高速缓存器 22、和单片 RAM 24；计算单元 16 包括执行单元 26、一组寄存器 28、高速缓存器 30、和单片 RAM 32。计算单元彼此通过总线 34 连接。各计算单元还通过总线 38 与外存储器 36（如 RAM）连接。

运行时，待执行的指令分组，从调度存储单元 10 传送至指令发送单元 12。指令发送单元 12 把该指令分组分为构成它的各指令，并分别通过发送时隙 IS1 和 IS2，把两个指令送至计算单元 14、16。然后，计算单元 14、16 同时执行各自的指令。这样，长指令的不同部分被并行处理。

在另外的实施例中，每一计算单元内设有两个或更多的执行单元。此外，虽然图 1 画出两个计算单元 14、16，但也可以设置单个计算单元或多于两个计算单元。

计算单元 14、16 的每一个，使用流水线技术，把它处理指令的速率最大化。流水线操作是通过把多个指令执行状态的每一个，作为单个流水线段而实施的。各指令连同所有部分完成的指令，按生产线的方式，每处理器时钟周期向前移动一流水线段，相继地流过各流水线段。当它们到达流水线末端时，完成各指令的执行。

处理器的存储器按分级结构排列，以高速缓存器 22、30（小的快速存储器）为最高级。向下的一级由单片存储器 24、32 以及外存储器 36 构成。因此，单片存储器 24、32 构成与外存储器 36 同一存储器级（本例中是 RAM 存储器级）的一部分。可以设置更多的存储器级，例如在硬盘（未画出）的形式下。每一级保持它以下的级的数据的一子集，同时，相同级存储器的不同部分，一般保持不同的数据。

处理器被设计成用虚存储器工作。虚存储器是虚的存储器区域，不直接与处理器能访问的物理存储器对应。为了访问各存储器 24、32、和 36 中物理存储器单元，要把处理器产生的虚地址转换为物理地址。虚存储器的使用，能在处理器之间实现程序的可移植性，在用户与程

序之间提供保护，还产生大存储器区域的假象。

虚存储器，以及被它映射的物理存储器，两者都分为固定大小的页。对虚存储器和物理存储器，某地址在一页内的偏移是相同的。因此，虚地址可以分解为虚页号码（或帧号码）和偏移。从虚地址到物理地址的转换缩减为从虚帧号码到物理帧号码的转换。页表被用来把虚帧号码转换为物理帧号码。典型的系统可以对每种处理有一页表，并对操作系统有另外的页表。对被映射到物理存储器的每页虚存储器，在该页表中有表目，称为页表的表目（PTE），该表目中包含虚帧号码和虚帧号码映射的物理帧号码。为加速转换过程，把最近用过的 PTE 存储在 PTE 的高速缓存器内，亦称转换后备缓冲器（TLB）。作为改进，虚页可以大于物理页，且每一虚页可以映射到两页或更多的物理页上。

图 2 画出一个例子，说明如何把某些虚存储器页映射到单片存储器的物理页上。本例中，虚地址和物理地址都是 32 比特宽，并有 32 千字节的本地存储器。每一虚页大小是 8 千字节，映射到大小是 4 千字节的两页物理页上。虚页编址从 00000000 到 00007FFF，而物理页的编址从 10000000 到 10007FFF（所有地址均按十六进制记数法）。参考图 2，在地址 00006000 的虚页，其上半部映射到地址 10006000 的物理页，而其下半部映射到地址 10002000 的物理页；地址 00004000 的虚页没有被映射；地址 00002000 的虚页，其下半部映射到地址 10004000 的物理页，其上半部没有被映射；以及地址 00000000 的虚页没有被映射。

按照图 2 所示方案，虚地址的 12 个最低有效比特（比特 0 至 11），与物理地址的 12 个最低有效比特相同。从该页虚地址的 12 至 13 比特获得虚帧号码，又从该页物理地址的 12 至 13 比特获得物理帧号码。每一 PTE 包含虚帧号码和该虚页映射的两页物理页的帧号码。此外，还有与每一物理页号码相关的有效比特。对该种方案的一个 PTE 示于图 3。

为了把虚地址转换为物理地址，该地址的 PTE 或者在 TLB 中被

查阅，如果该 PTE 不在该 TLB 中，便是在页表中被查阅。理论上说，这一步必须在访问任何存储器前完成，但后面将说明，在某些情形下，有可能对正确的物理地址作初始假定，不必完成完整的地址转换。

在单片系统设备中，最好把访问单片高速缓存器的路径，引入处理器的流水线，使单片高速缓存器的访问能在一个流水线段中进行。如果提供了一定量的单片 RAM，那么，最好也把访问该单片 RAM 的路径，引入处理器的流水线。但是，通常有许多设计约束，加在处理器的设计上，并对实现单片存储器的流水线访问方式加以限制。特别有关的设计约束如下：

- 各 RAM 块常常是同步的，就是说，它们的输入寄存器，包含对当前周期的地址、写入数据、和读出/写入控制。
- 各系统被限制使用单一的时钟，所有寄存器都按该时钟的上升沿同步。

上述约束主要表明，对 RAM 块的流水线访问，必须在流水线时钟的上升沿开始，从而，地址、写入数据、和读出/写入控制，必须在前一流水线段结束之前确定。因此，当访问一块单片存储器时，需要的存储器单元的物理地址，必须在紧靠访问单片存储器的流水线段的前一段结束之前，可供使用。通常，要完成完整的地址转换，或者需要一完整的流水线段，或者需要流水线段的重要部分。在地址转换之前，要执行虚地址的计算操作，例如把寄存器的内容加在一文字值上或加在另一个寄存器的内容上。地址计算和完整的地址转换操作，通常超过流水线时钟的有效容限，从而不能在单一的流水线段中实现。单片存储器访问得到的各流水线段的（未优化）顺序，画在图 4。

与之相反，因为高速缓存器通常不受上述约束的限制，可以在地址计算流水线段后面的流水线段，访问单片高速缓存器，即比图 4 所示单片 RAM 段提前一段。例如，如果高速缓存器的大小比虚存储器的页还小，那么能够用虚地址较低位的比特，访问高速缓存器，因为那些比特对虚地址和物理地址都是相同的。

在本发明的各个实施例中，对单片 RAM 的地址转换所用时间降

低了。这一点是通过在计算虚地址的流水线段中附加的逻辑完成的，在将发生单片访问的假定下，该逻辑确定应当访问单片存储器的哪一页。这样可以减少单片存储器的读出时间。此外，在本发明的各个实施例中，使用带有单片高速缓存器和单片 RAM 的处理器，所以数据能够在流水线中同一点，按装入指令返回，不论该数据是从高速缓存器获得的还是从单片 RAM 获得的。

第一实施例

按照本发明的地址转换技术的第一实施例，为了简化从虚地址导出物理地址的逻辑，对从物理页到虚页的分配施加某种约束。该约束是，只要两页虚页的虚地址比特中某一小的或任选的子集相同，则它们不能同时被映射到单片存储器上。该子集通常包括 n 比特，而在该单片存储器中物理页的数目是 2^n ，于是，所有物理页至少可以被一页虚页映射。本文把这些比特称为局部虚帧号码。

作为一个例子，图 5 画出两个虚地址 X 和 Y。在高位的比特内，三个比特已经被标出（比特 a、b、和 c）。如果在 X 和 Y 中的这些比特是全同的，那么地址 X 和 Y 不能同时在转换的高速缓存器中被保持。

n 个虚地址比特的选出的子集，本例中 $n=3$ ，能够用来查阅单片存储器中一页使用该物理帧号码最低有效 n 比特（本例中是该物理地址的 12 至 14 比特）的物理页。本文把这些比特称为局部物理帧号码。在虚页被实际映射到单片存储器中物理页的假定下，在完整转换完成之前，能够开始访问单片存储器。

在第一实施例中，维持用于查阅局部物理帧号码的局部查阅表。该局部查阅表是一个 2^n 、 n -比特局部物理帧号码的表，该 n -比特局部物理帧号码供 n 虚地址比特选出的子集标引。然后由该局部帧号码，与该虚地址的偏移比特（在上述例子中是比特 0 至 11）一起，形成一局部物理地址。用该局部物理地址，访问含有该地址的单片存储器中的存储器单元。

如果要访问的只有单片存储器，只要局部查阅表中被选出的表目与原来的虚地址匹配，则该推测的地址将被确认。但是，如果虚地址

还要转换至芯片外存储器（在那里可以预料有数量大得多的物理页），那么许多虚页将映射到该局部查阅表中每一表目。因此，在该情形下，必须进行检验以确定推测的地址是否正确。

在本实施例中，为了检验推测的地址是否正确，维持该单片地址的 PTE 高速缓存器。该高速缓存器本文称为完整查阅表。该完整查阅表的访问是与访问单片存储器并行的，以便获得与正被访问的单片存储器页对应的（完整的）虚帧号码。把选出的 PTE 的虚帧号码，与虚地址比较，如果两者匹配，则推测的映射是正确的。

本技术根据的事实是，可以推测某一地址，然后，抱着该地址正确的期望，从单片存储器读出数据。完整转换与读出操作并行地发生，以检验该地址是否真正正确。如果该地址不正确，则废弃该数据。该种推测的访问能够在读出操作时进行，不必害怕破坏任何数据。但对写入操作，则宁可把写入推迟，直至确信该地址是正确的。然而，如果确信被写入的存储器区域不包含任何有效数据，那么也可以在完整的地址转换已经发生之前，开始写入操作。为简单起见，下面的说明是指读出操作，但可以预料，在适当的环境下，也可以执行写入操作。

第一实施例中，用于访问单片存储器的处理器各部分，示意地画在图 6。参照图 6，该处理器包括地址计算单元 100、局部查阅表 102、地址范围检验器 103（可选）、虚地址寄存器 106、流水线寄存器 107、同步单片 RAM 101、完整查阅表 104、地址比较器 105、命中/没命中寄存器 108、数据寄存器 109、和没命中处理程序 112。

工作时，地址计算单元 100 在给定的时钟周期中，产生一虚地址，该虚地址可以编址在整个 32 比特地址范围的任何单元。在同一时钟周期中，虚地址的 12、13、和 14（在本例中）比特，被用来从局部查阅表 102 选择八个 3-比特局部帧号码中的一个，在本例中，该局部查阅表 102 是 8-表目 3-比特的非同步 RAM。与之并行，地址范围检验器 103 可以任选地确定，该虚地址是否在单片存储器访问的合法范围。

在该时钟周期结束时，该虚地址在流水线寄存器 106 中被捕获，并且该局部物理帧号码在流水线寄存器 107 中被捕获。同时，把该虚

地址的最低有效 12 比特附于从局部查阅表 102 读出的 3-比特局部帧号码后面，形成该 15 比特的局部物理地址。单片范围检测逻辑 103 可以任选地用来取消单片 RAM 101 的读出启动输入。该 15 比特的局部物理地址和任选的读出启动输入，都寄存在单片 RAM 101 的输入上。

在每一时钟周期内，都按照前一时钟周期产生的局部物理地址，从单片 RAM 读出数据，或把数据写入单片 RAM。存储在寄存器 107 中的局部物理帧号码，被用来选择存储在完整查阅表 104 中八个 PTE 的一个。把保持在选出的 PTE 中的虚帧号码，与保持在寄存器 106 中虚地址的 12 至 31 比特进行比较，并检验该 PTE 中的有效比特。如果存储在寄存器 106 的虚地址与选出的 PTE 中的虚帧号码不对应，或如果选出的 PTE 是无效的，那么就发生没命中，并且从单片 RAM 读出的数据是无效的。命中/没命中寄存器 108 复位，以指示发生没命中。

否则，从单片 RAM 读出的数据是需要的数据，并可以正常使用。此时，令该命中/没命中寄存器 108 指示发生命中。

如果命中/没命中寄存器 108 指示发生没命中，则采取如下动作：

- 废弃或忽略寄存器 109 中保持的数据。
- 把单片存储器中的物理页分配给含有地址计算单元产生的虚地址的虚页。
 - 如果该新分配的物理页已经被虚存储器映射，则把该物理页的内容存储在存储器的其他地方。
 - 把该物理页（与该虚页对应）的新内容装入该物理页。
 - 更新局部查阅表 102 和完整查阅表 104。

新物理页的指配，以及随后存储器与查阅表的更新，都由没命中处理程序 112 完成，该没命中处理程序可以用硬件或软件实现，也可以两者兼用。当局部查阅表 102 中某一表目已被装入时，待映射的虚页的基本地址中 12、13、和 14 比特，被用来选择局部查阅表中的表目。用与物理页对应的地址的 12、13、和 14 比特，盖写该表目。与此同时，待映射的虚地址的基本地址中 12、13、和 14 比特，被用来选择完整查阅表 104 中的表目，并把完整的虚帧号码写入该完整查阅

表的该单元。这样能保证任何先前映射的、与新虚地址共享 12、13、和 14 比特的虚地址，将不再被映射。

现在参照图 7 至 13，说明第一实施例工作原理的一个例子。图 7 至 13 代表局部查阅表 102、完整查阅表 104、虚地址寄存器 106、和流水线寄存器 107 的内容。这些图还画出带有抽取的并用于标引局部查阅表的三个比特的虚地址。在各表中标记“-”的表目，代表“不必理会”的情形。这些表目可能包含不确定的值，但本方案仍能正确工作。

在每一时钟周期结束时，输入的虚地址被存储在虚地址寄存器 106 内，而局部物理帧号码被存储在流水线寄存器 107 内。下一时钟周期，用局部查阅表中的值，从包含虚页虚帧号码的完整查阅表 104 中选择一行，该虚页被映射到单片存储器中由局部查阅表内容给出的页号码上。这里还有一有效比特 V，该比特对有效映射置值 1，而对无效映射则置值 0。

图 7 画出局部查阅表 102 的状态和完整查阅表 104 在复位后的状态。复位时，在完整查阅表 104 中所有有效比特都置值 0。这样保证开始时没有单片存储器被映射。

图 8 画出接着复位后，提交一地址等待转换的情形。此时，没有单片存储器被映射，且查阅表中所有有效比特都被置值 0。图 8 中，提交地址 00006000（十六进制），等待转换。假定局部查阅表中表目 6 包含随机值 4。该值被读出并存储在流水线寄存器 107。下一时钟周期，完整查阅表的行 4 被读出，并发现其 V 比特是 0。这一结果导致没命中，因为在转换表中，对虚地址 00006000 没有有效转换。

响应该没命中，没命中处理程序 112 把转换表的表目装入局部和完整查阅表，以便需要的虚地址能够在随后被转换。该没命中处理程序必须知道虚地址和它被映射的物理地址。在本例中，没命中处理程序用虚地址的 12、13、和 14 比特，在局部查阅表中选择一行，并用对应的物理地址的 12、13、和 14 比特，在完整查阅表中选择一行。然后，把物理地址的 12、13、和 14 比特装入局部查阅表选出的行。

与此同时，把虚地址的 12 至 31 比特装入完整查阅表中选出行的 VFN 字段，并把相关的 V 比特置值 1。

图 9 画出转换表目装入之后的转换逻辑状态。在本例中，假定虚地址 00006000 在物理地址 10002000 上映射到单片存储器。因此，局部查阅表的行 6 装入值 2，而完整查阅表的行 2 装入值 00006。

图 10 画出页表的表目已经装入之后，如果重复原来图 8 所示的读出操作的情况。局部查阅表的输出（本例中是值 2）与虚地址的 0 至 11 比特连接，并用作单片存储器的读出地址。局部查阅表的输出还存储在流水线寄存器 107 中。

在下一个流水线周期中，读出完整查阅表 104 的行 2，并把存储在该行 VFN 字段的值（本例中是值 00006），与从流水线寄存器存储的虚地址获得的帧号码比较。因为这些值完全相同，且行 2 的 V 比特被置值 1，在命中/没命中寄存器 108 中登记命中，所以，与完整查阅表的访问并行地从单片存储器读出的数据，能够供处理器使用。

图 11 画出一个例子，表明开始时假定虚地址映射到单片存储器，但实际上映射到外存储器的情形。本例中，查阅表的内容假定与图 10 相同，但提交待转换的虚地址是 40006000。与图 10 所示的例子一样，选取局部查阅表 102 的行 6。存储在行 6 的值与虚地址的 0 至 11 比特连接，获得单片存储器的读出地址。还把存储在行 6 的值寄存在流水线寄存器 107 中。

在下一个流水线周期中，读出完整查阅表 104 的行 2，并把存储在该行 VFN 字段的值（本例中是值 00006），与从流水线寄存器存储的虚地址获得的帧号码（本例中是值 40006）比较。因为这些值不完全相同，在命中/没命中寄存器 108 中登记没命中，并把从单片存储器读出的数据忽略或废弃。没命中不会引起转换表的更新，因为没命中处理程序能确定该需要的物理地址不在单片存储器中。另一种转换机构，如常规的完全相关 PTE 高速缓存器，用来处理不映射到单片 RAM 的地址的转换。

图 12 画出一个例子，表明单片存储器的一虚页，作为已被映射

的一页，标引至转换表中相同单元的情形。在图 12 中，假定系统软件决定把虚地址 00016000 映射到物理地址 10005000。从原理上说，该映射可以与已有的从 00006000 到 10002000 的映射共存。但是，如果有两页虚地址比特中 12、13、和 14 比特相同，则该两页虚页不能同时映射，由于受此约束，不允许该两个映射共存。

当虚地址 00016000 被提交该转换机构时，局部查阅表的行 6 被选出。这一步产生 2 的局部物理帧。接着选出完整查阅表的行 2，这一步产生 00006 的有效 VFN。由于 00006 与流水线寄存器中的 VFN（它是 00016）不匹配，该转换机构记录 TLB 没命中。

然后，没命中处理程序 112 决定该 00016000 应该在物理地址 10005000 上映射到单片存储器。为演示当已有的转换被驱逐时将发生什么，画出表中给出的从 00013000 到 10005000 的另一种转换。这一情形对应于系统软件分享两页虚页间的物理存储器资源，在某一瞬间只有其中一页有效地映射到该物理页。没命中处理程序 112 把物理页 10005（对应于虚页 00013）的内容存储在存储器的其他地方，然后把页 10005（对应于虚页 00016）的新内容，从存储器曾经存储该新内容的某单元装入。在该过程中，对该页加上无效标记，避免该物理页的任何使用。过程完成后，把新的映射写入转换表。

图 13 画出新映射已经写入后转换表的状态。从图可见，如果提交的虚地址是 00016000，那么，局部查阅表的行 6 将被选出。这一步又导致完整查阅表的行 5 被选出，并产生 00016 的有效 VFN。从而将导致命中。

应当指出，局部查阅表的行 3 仍旧包含值 5，且完整查阅表的行 2 仍旧包含值 00006。但是，选择局部查阅表行 3 的任何虚地址的 12、13、和 14 比特将分别被置值 1、1、和 0，且该地址永远不能与完整查阅表的行 5 匹配，因为完整查阅表行 5 的比特当前值分别是 0、1、和 1，因而总记录没命中。类似地，完整查阅表的行 2 永远不能被访问，除非映射到物理帧 10002 的新转换装入转换表。写入该新转换的作用将使完整查阅表的行 2 变为可访问的，但也把行 2 的内容重新写

成该新转换的正确值。

第二实施例

在本发明的第二实施例中，按与第一实施例类似的方式，产生推测的地址转换，除此之外，不用正好覆盖单片存储器各页转换表目的直接标引表，而用结合命中检测器的关联标引 TLB，该命中检测器确定单片存储器各地址的假定是否正确。

图 14 按照第二实施例，画出访问单片存储器的处理器的各部分。参照图 14，该处理器包括地址计算单元 100、局部查阅表 102、地址范围检验器 103（任选）、虚地址寄存器 106、流水线寄存器 107、同步单片 RAM 101、命中/没命中寄存器 108、数据寄存器 109、TLB 查阅逻辑 111、命中检测器 110、和没命中处理程序 113。与第一实施例共通的处理器各部分，用相同的参考数字表示。

工作时，如同第一实施例，首先计算虚地址，然后向 8-表目 3-比特的 RAM 提交。任选地，该虚地址由单片范围逻辑 103 检验，以确定该请求是否能够访问单片存储器。如果答案是“否”，则没有必要执行单片存储器的访问。3-比特 RAM 102 的输出与虚地址的低位 12 比特连接，然后，如果适当，还加上启动信号，一并寄存在单片 RAM 101 的输入上。RAM 102 的 3-比特输出还要单独寄存，供随后的流水线段使用。

与访问单片存储器的数据并行，把虚地址提交给 TLB 查阅逻辑 111。该 TLB 查阅逻辑是 PTE 高速缓存器，可以用提交检验的虚地址进行关联搜索。如果提交的虚地址被该 TLB 中任何 PTE 所映射，那么，该 PTE 则从 TLB 中被读出，并供该 TLB 的输出使用。TLB 的输出包括虚帧号码和匹配转换的局部帧号码，还有有效标志 *tlb_hit*，如果发现不匹配，该标志为假。

命中检测逻辑 110 执行两种检验。第一，把 TLB 输出的局部帧号码的 12、13、和 14 比特，与存储在流水线寄存器 107 中预测的局部物理帧比较，产生 *prediction_hit* 信号。第二，且是并行地，对该局部帧号码解码，以确定该物理地址是否在分配给单片存储器的范围，

产生 *is_physical* 信号。转换的状态由以下规则确定。

- 如果 *tlb_hit* 为假，则该转换无效，并在流水线寄存器 108 中，报告 TLB 没命中。
- 如果 *tlb_hit* 为真，但 *is_physical* 为假，则报告 TLB 没命中。这一点对应于 TLB 转换的地址要么保持在外部，要么保持在独立的高速缓存器内。
- 如果 *tlb_hit* 为真，且 *is_physical* 亦真，但 *prediction_hit* 为假，则在流水线寄存器 108 中报告 TLB 没命中。这种情形通常不会发生，但可以按对付其他 TLB 没命中同样方式处理（即，重新装入一 TLB 表目）。
- 如果 *tlb_hit* 为真，*is_physical* 为真，且 *prediction_hit* 亦真，则在流水线寄存器 108 中报告 TLB 命中，从而推测的存储器访问是正确的。

如果在流水线寄存器 108 中报告 TLB 没命中，则废弃存储在寄存器 109 中的任何数据。然后，没命中处理程序 113 把单片存储器中的一页，分配给包含地址计算单元产生的虚地址的虚页。如有必要，把最新被分配的物理页的内容，存储在存储器的其他地方，而把物理页的新内容（与该虚页对应）装入该物理页。然后，更新局部查阅表 102 和 TLB 查阅表 111 各表目。

当装入一 TLB 表目时，没命中处理程序 113 对物理地址解码，以确定它是否在单片存储器的地址范围。如果是，则在虚地址的 12、13、和 14 比特所标引的单元上，把新 VFN 的 12、13、和 14 比特写入 8-表目 3-比特 RAM 102。

应当指出，以上对本发明的说明，纯粹以举例方式，因而在本发明的范围内，能够对细节作各种修改。

虽然上述以举例方式的说明中，涉及 VLIW 处理器，但显然，本发明同样可用于其他处理器，如简约指令系统处理器（RISC）、数字信号处理器（DSP）、或任何其他类型处理器。可以具体实现本发明的处理器，包括在多媒体应用、网络路由器、视频移动电话、智能汽

车、数字电视、语音识别、3D 游戏等等中使用的、作为高度集成“单片系统”(SOC)中处理器的“核心”。

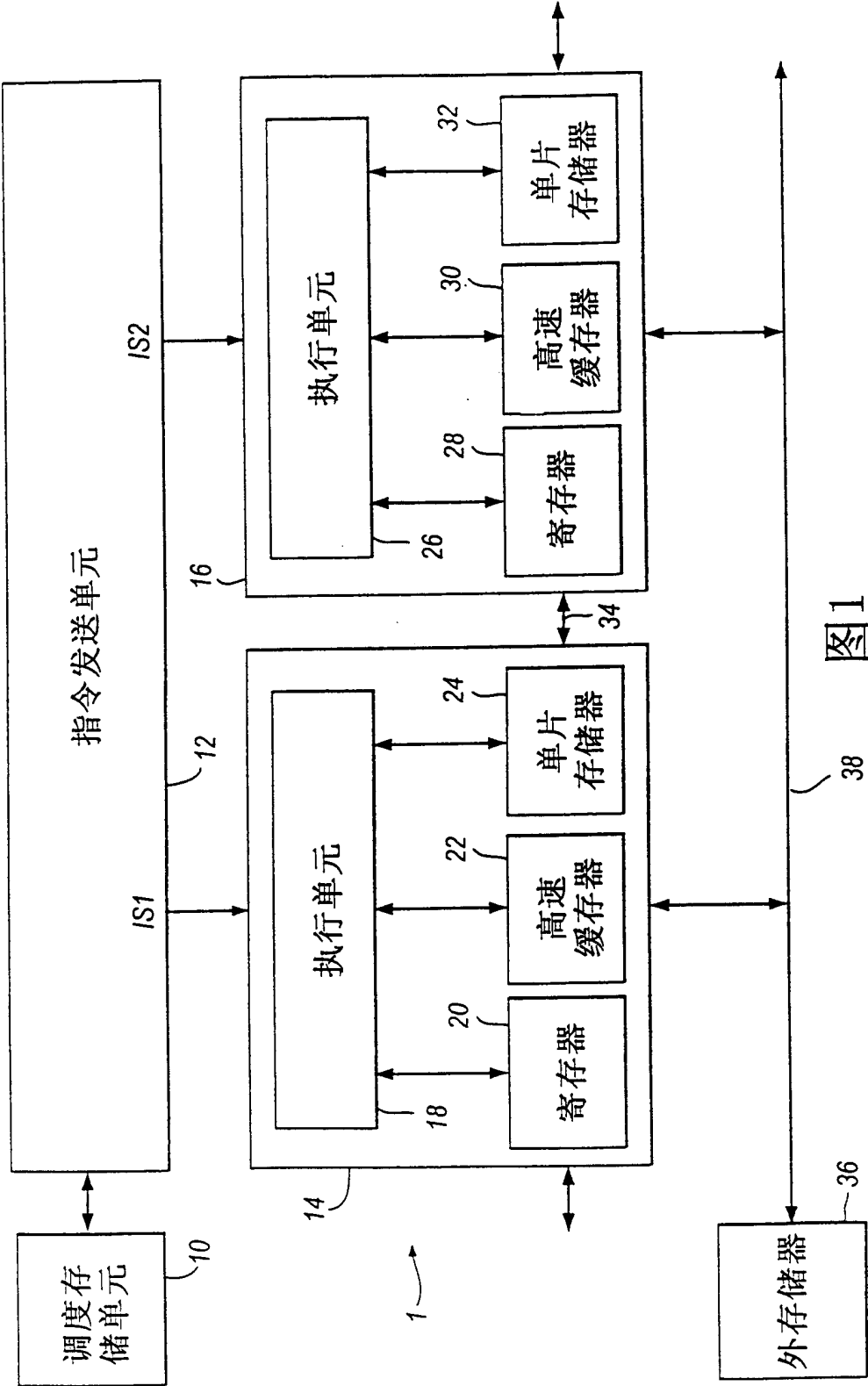


图1

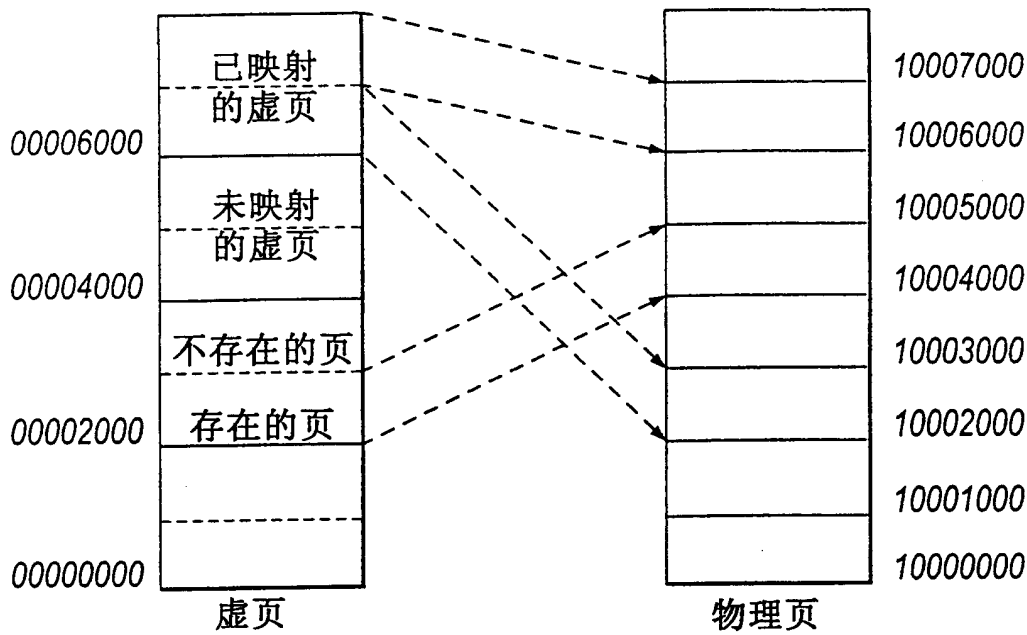


图2

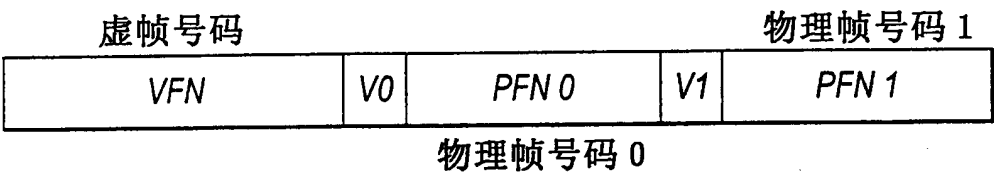


图3

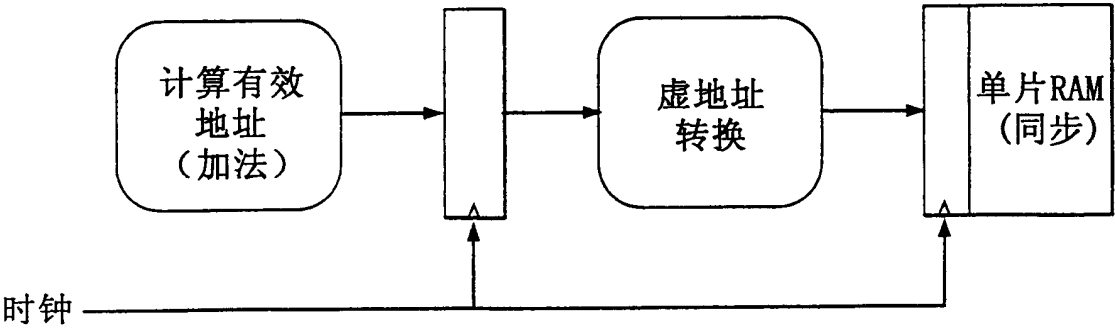


图4

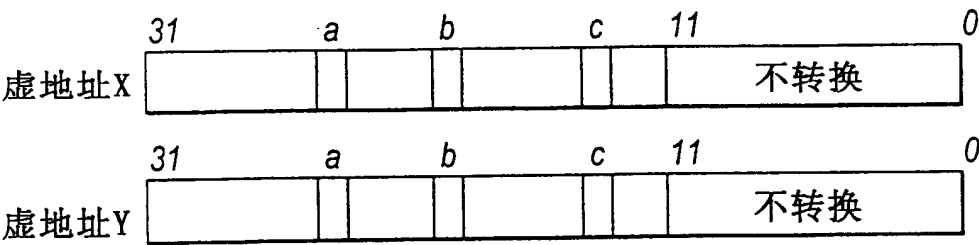


图5

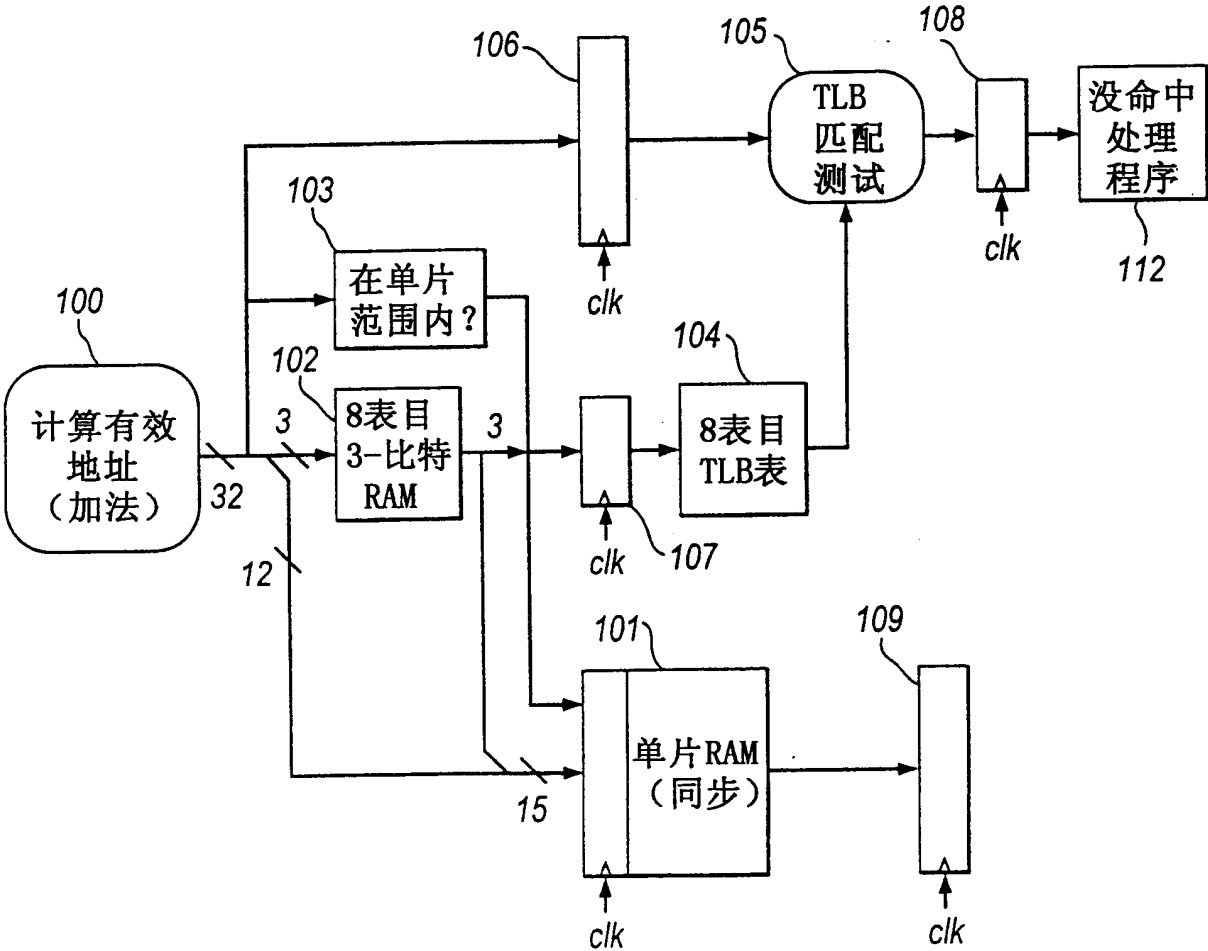


图6

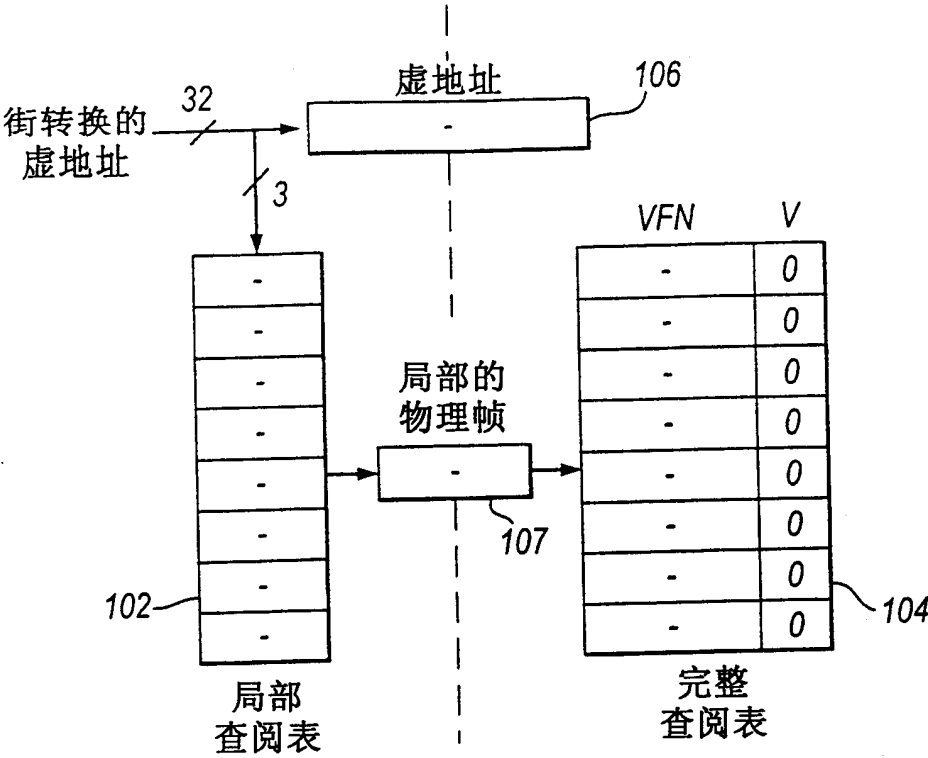


图7

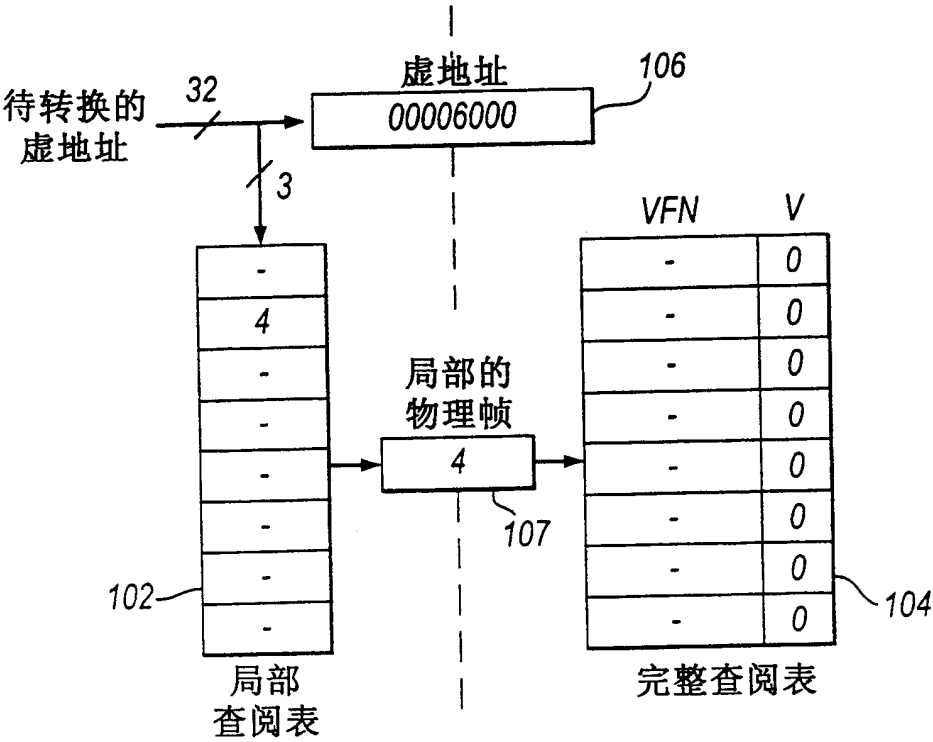


图8

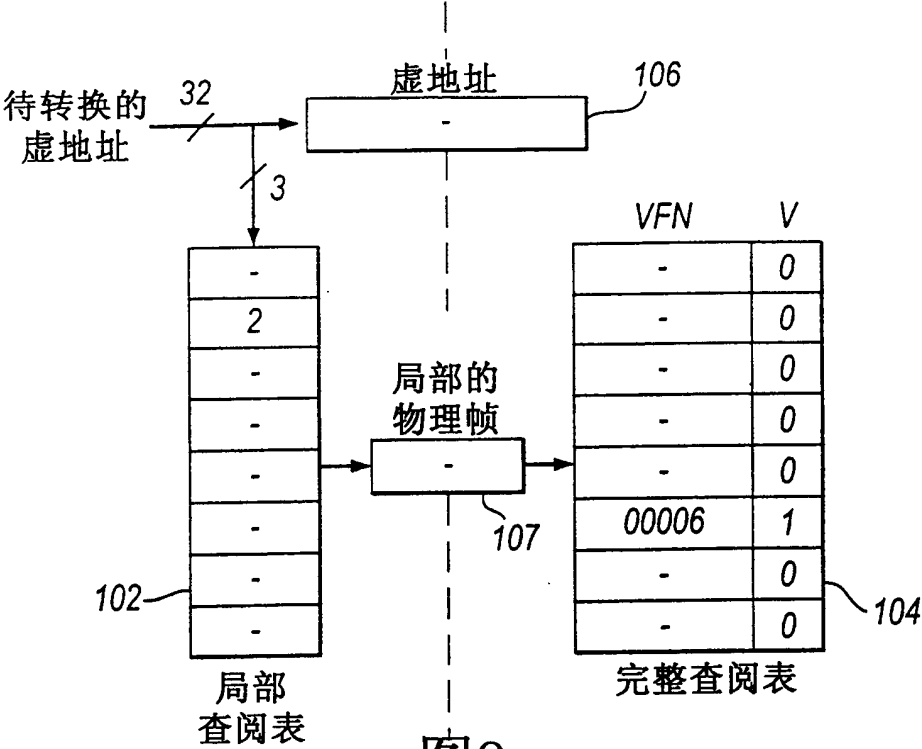


图9

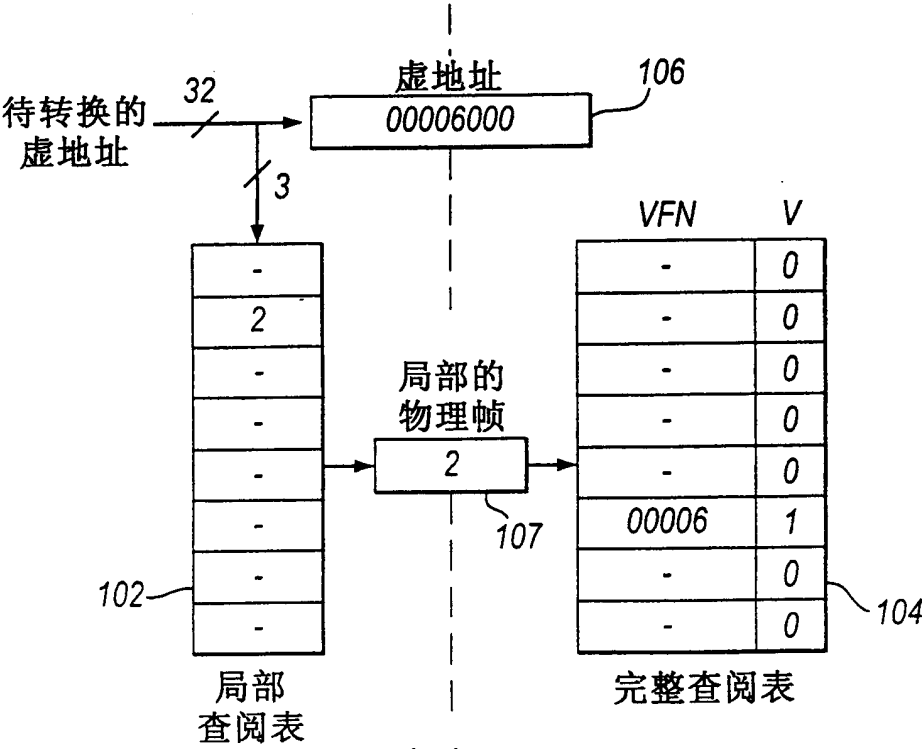


图10

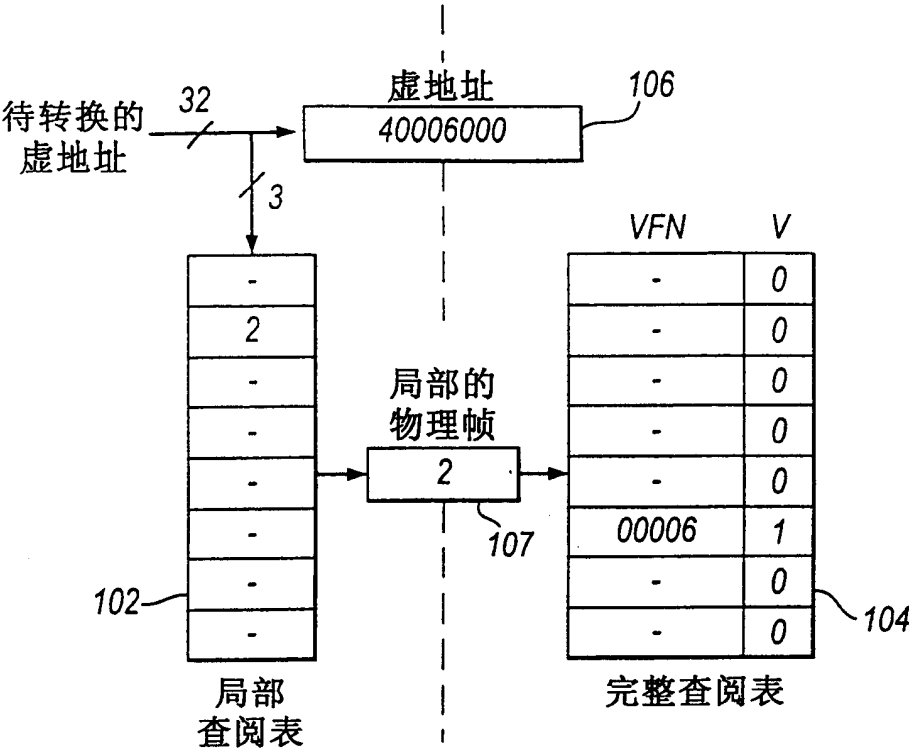


图11

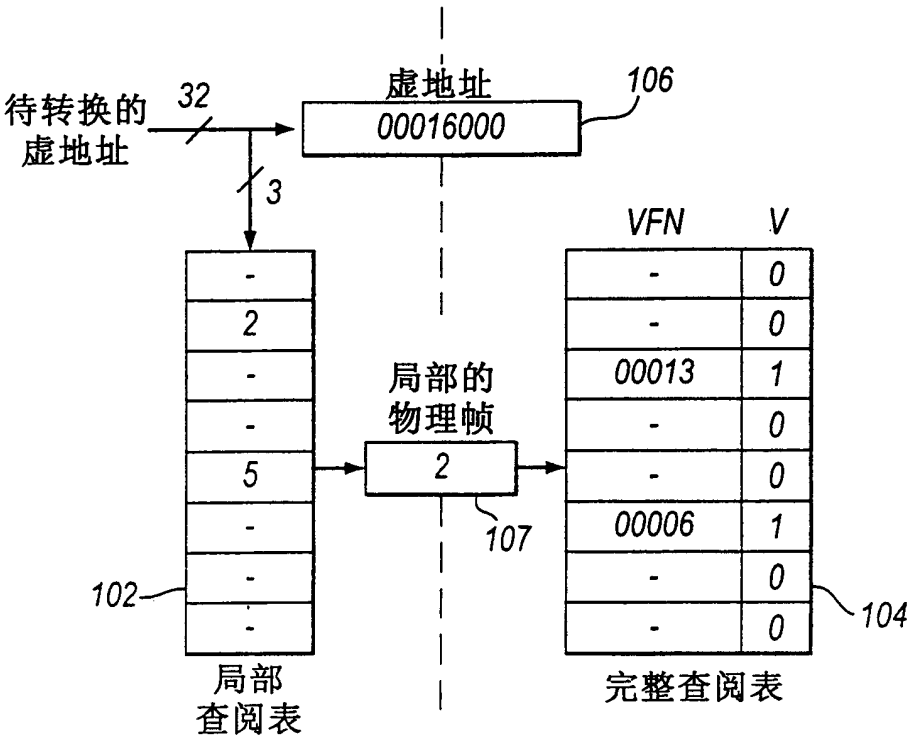


图12

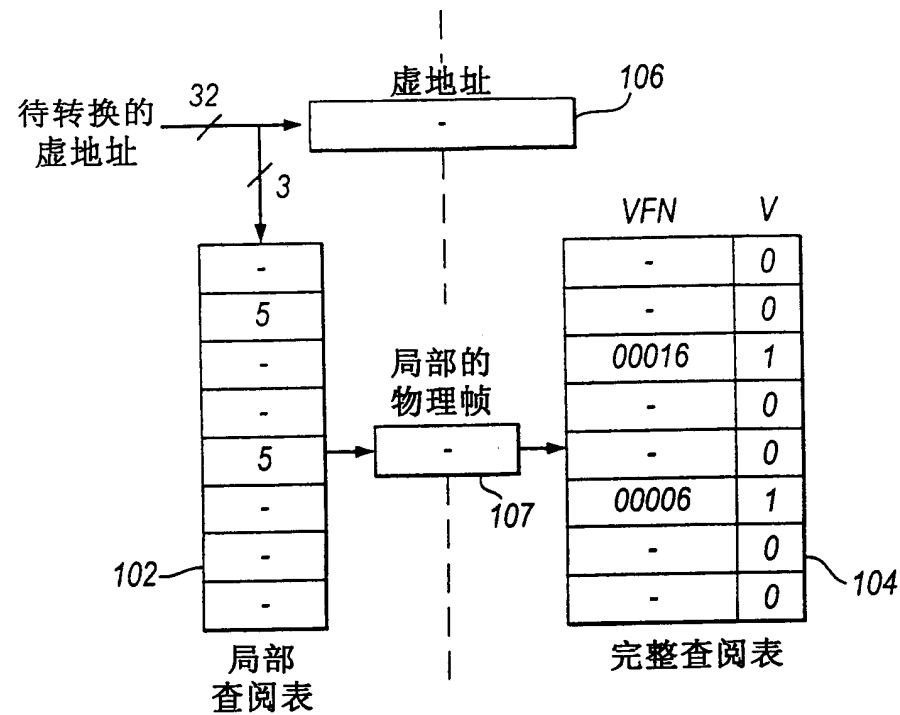


图13

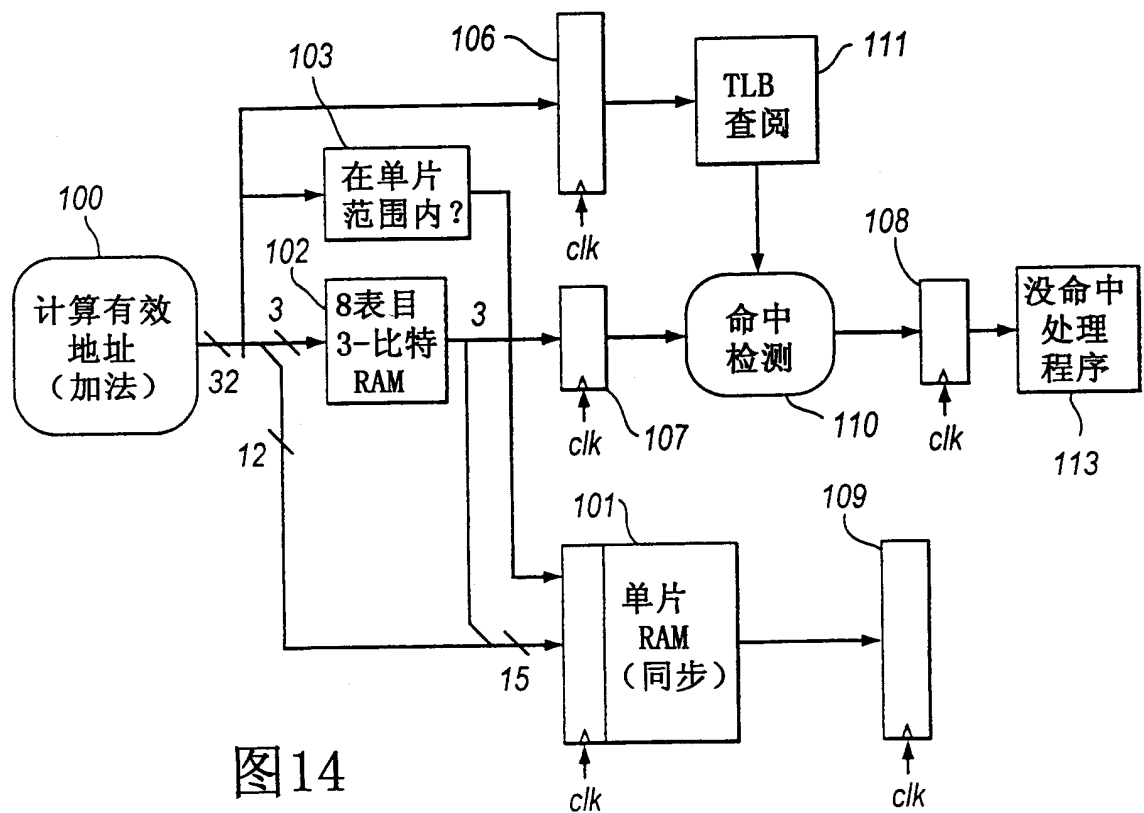


图14