



(12) 发明专利申请

(10) 申请公布号 CN 114072777 A

(43) 申请公布日 2022. 02. 18

(21) 申请号 201980098081.4

(51) Int.Cl.

(22) 申请日 2019.07.02

G06F 12/06 (2006.01)

(85) PCT国际申请进入国家阶段日  
2021.12.30

(86) PCT国际申请的申请数据  
PCT/CN2019/094419 2019.07.02

(87) PCT国际申请的公布数据  
W02021/000263 EN 2021.01.07

(71) 申请人 微软技术许可有限责任公司  
地址 美国华盛顿州

(72) 发明人 张霖涛 J·G·班尼特 李博杰

(74) 专利代理机构 北京市金杜律师事务所  
11256

代理人 张宁

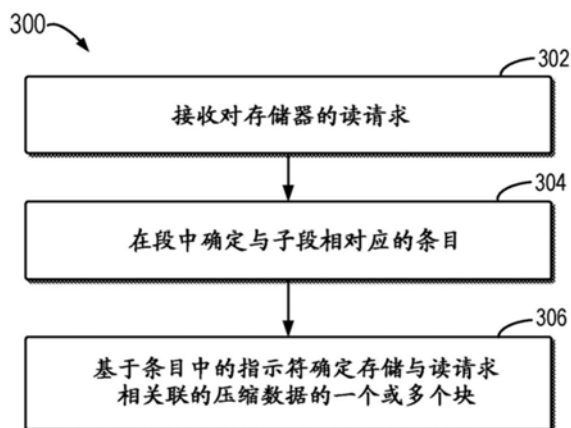
权利要求书2页 说明书11页 附图4页

(54) 发明名称

基于硬件的存储器压缩

(57) 摘要

压缩存储器被划分为多个段,每个段被划分为多个子段,并且未压缩数据空间中的每个子段被压缩为压缩数据空间中的块。当接收到对压缩存储器中的段中的子段的读请求时,首先基于子段与条目之间的第一级地址映射确定对应条目,然后基于条目与块之间的第二级地址映射确定对应块。通过使用两级地址映射,可以减小条目的大小,从而实现较低的元数据开销。此外,提出了用于压缩存储器的数据布局。



1. 一种用于存储器压缩的方法,包括:

接收对存储器的读请求,所述存储器被划分为多个段,所述多个段中的一个段被划分为多个子段,所述多个子段中的一个子段被压缩为一个或多个块,并且所述读请求与所述存储器中的段的子段相关;

在所述段中确定与所述子段相对应的条目,所述条目包括指示符;以及

基于所述条目中的所述指示符确定存储与所述读请求相关联的压缩数据的块。

2. 根据权利要求1所述的方法,其中在所述段中确定与所述子段相对应的条目包括:

基于映射规则确定与所述子段相对应的所述条目,所述映射规则包括子段与条目之间的在所述段内的静态映射。

3. 根据权利要求1所述的方法,其中一个子段的大小被预设为一个块的大小的至少八倍。

4. 根据权利要求1所述的方法,其中基于所述条目中的所述指示符确定存储与所述读请求相关联的压缩数据的块包括:

基于所述条目中的两个或更多个指针确定与所述读请求相关联的两个或更多个块,所述条目中的指针的最大数目基于一个子段的大小与一个块的大小之间的第一比率来确定,并且所述条目中的一个指针的大小基于一个段的大小与一个块的大小之间的第二比率来确定。

5. 根据权利要求4所述的方法,还包括:

响应于确定与所述读请求相关联的两个或更多个块,从所述两个或更多个块并行读取所述压缩数据;以及

解压缩所述压缩数据以生成解压缩数据。

6. 根据权利要求1所述的方法,还包括:

接收对所述存储器的写请求;

响应于接收到所述写请求,检查段中的位图,所述段包括多个条目、多个块和所述位图,并且所述位图中的每个位指示所述段中的每个块是否被分配;

响应于所述位图指示所述段中有足够的块可用于所述写请求,基于所述位图将与所述写请求相关联的压缩数据写入所述段中的一个或多个块中;以及

更新所述位图和所述段中的一个或多个条目。

7. 根据权利要求6所述的方法,还包括:

响应于所述位图指示所述段中没有足够的块可用于所述写请求,确定所述存储器中具有足够的可用块的另外的段;以及

将与所述写请求相关联的压缩数据的至少一部分写入所述另外的段中的一个或多个块中。

8. 根据权利要求6所述的方法,其中不被频繁访问的数据被迁移到压缩存储器,而被频繁访问的数据被迁移到未压缩存储器,并且所述未压缩存储器和所述压缩存储器共享从处理器可访问的同一地址空间。

9. 根据权利要求1所述的方法,其中所述方法在所述存储器中被实现,并且对所述存储器中的所述段的所述子段的所述读请求经由存储器控制器而被接收。

10. 一种存储器,包括:

存储器芯片;以及

控制器,被配置为执行动作,所述动作包括:

接收对所述存储器芯片的读请求,所述存储器芯片被划分为多个段,所述多个段中的一个段被划分为多个子段,所述多个子段中的一个子段被压缩为一个或多个块,并且所述读请求与所述存储器中的段的子段相关;

在所述段中确定与所述子段相对应的条目,所述条目包括指示符;以及

基于所述条目中的所述指示符确定存储与所述读请求相关联的压缩数据的块。

11.根据权利要求10所述的存储器,其中在所述段中确定与所述子段相对应的条目包括:

基于映射规则确定与所述子段相对应的所述条目,所述映射规则包括子段与条目之间的在所述段内的静态映射。

12.根据权利要求10所述的存储器,其中一个子段的大小被预设为一个块的大小的至少八倍。

13.根据权利要求10所述的存储器,其中基于所述条目中的所述指示符确定存储与所述读请求相关联的压缩数据的块包括:

基于所述条目中的两个或更多个指针确定与所述读请求相关联的两个或更多个块,所述条目中的指针的最大数目基于一个子段的大小与一个块的大小之间的第一比率来确定,并且所述条目中的一个指针的大小基于一个段的大小与一个块的大小之间的第二比率来确定。

14.根据权利要求13所述的存储器,所述动作还包括:

响应于确定与所述读请求相关联的两个或更多个块,从所述两个或更多个块并行读取所述压缩数据;以及

解压缩所述压缩数据以生成解压缩数据。

15.一种包括可执行指令的程序产品,所述可执行指令当在设备上被执行时使所述设备执行动作,所述动作包括:

接收对存储器的读请求,所述存储器被划分为多个段,所述多个段中的一个段被划分为多个子段,所述多个子段中的一个子段被压缩为一个或多个块,并且所述读请求与所述存储器中的段的子段相关;

在所述段中确定与所述子段相对应的条目,所述条目包括指示符;以及

基于所述条目中的所述指示符确定存储与所述读请求相关联的压缩数据的块。

## 基于硬件的存储器压缩

### 背景技术

[0001] 存储器是指计算设备中存储用于在计算系统中即刻使用的信息的组件,并且存储器也称为主存储器。诸如随机存取存储器 (RAM) 等存储器高速运行,其不同于提供慢速访问信息但提供更高容量的存储装置。通常,存储器主要有两种,诸如易失性存储器和非易失性存储器,并且易失性存储器的示例包括动态随机存取存储器 (DRAM)、静态随机存取存储器 (SRAM) 等。

[0002] 存储器压缩是一种存储器管理技术,它利用数据压缩来减少存储在存储器中的数据的大小,该技术可以包括基于软件的存储器压缩和基于硬件的存储器压缩。基于软件的存储器压缩已经在操作系统中使用,其中压缩和解压缩过程通常由处理器或处理单元来实现。相反,基于硬件的存储器压缩通过硬件来实现,并且压缩和解压过程可以由硬件组件来处理。

### 发明内容

[0003] 在本公开的实施例中,提供了一种用于基于硬件的存储器压缩的方法。根据本公开的实施例的压缩存储器被划分为多个段,每个段被划分为多个子段,并且未压缩数据空间中的每个子段被压缩为数据空间中的一个或多个块。在接收到对压缩存储器中的段中的子段的读请求时,首先基于子段与条目之间的第一级地址映射确定对应条目,然后基于条目与一个或多个块之间的第二级地址映射确定一个或多个对应块。根据本公开的实施例,通过使用两级地址映射,可以减小条目的大小,从而实现较低的元数据开销。此外,所提出的压缩存储器的数据布局可以在压缩存储器中实现稳定快速的块分配。

[0004] 提供本“发明内容”以便以简化的形式介绍在下面的“具体实施方式”中进一步描述的概念的选择。本“发明内容”不旨在确定所要求保护的主题的关键特征或基本特征,也不旨在用于限制所要求保护的主题的范围。

### 附图说明

[0005] 通过参考附图更详细地描述本公开,本公开的实施例的上述和其他特征、优点和方面将变得更加明显。在附图中,相同或相似的附图标记代表相同或相似的元素,在附图中

[0006] 图1示出了可以在其中实现本公开的一个或多个实施例的计算设备/服务器的框图;

[0007] 图2示出了根据本公开的实施例的具有基于图块的压缩存储器的示例架构;

[0008] 图3示出了根据本公开的实施例的用于基于硬件的存储器压缩的方法的流程图;

[0009] 图4示出了根据本公开的实施例的针对图块的两级地址映射的示意图;

[0010] 图5示出了根据本公开的实施例的从段落到一个或多个块的示例数据压缩的示意图;

[0011] 图6示出了根据本公开的实施例的图块中的示例数据布局;

[0012] 图7示出了根据本公开的实施例的用于将压缩数据写入图块的方法的流程图;以

及

[0013] 图8示出了根据本公开的实施例的用于实现存储器压缩的示例硬件架构。

### 具体实施方式

[0014] 下面将参考附图更详细地描述本公开的实施例。尽管附图示出了本公开的一些实施例,但是应当理解,本公开可以以多种形式实现,并且本公开不应当被理解为限于本文中示出的实施例。相反,本文中提供这些实施例是为了能够更透彻和完整地理解本公开。应当理解,本发明的附图和实施例仅用于示例性目的,并不用于限制本发明的保护范围。

[0015] 如本文中使用的,术语“包括”及其变体应当理解为意指“包括但不限于”的开放术语。术语“基于”应当理解为“至少部分基于”。术语“一个实施例”应当理解为“至少一个实施例”。术语“另一实施例”应当理解为“至少一个其他实施例”。术语“一些实施例”应当理解为“至少一些实施例”。其他术语的定义将在下文中给出。

[0016] 通常,存储器消耗诸如计算机或服务器等计算设备的大部分硬件成本。例如,DRAM可能会消耗云服务器的高达40%或甚至更高的硬件成本,并且因此存储器压缩可以用于降低存储器成本或进一步扩展服务器。存储器压缩技术主要包括基于软件的存储器压缩和基于硬件的存储器压缩。基于软件的存储器压缩通过软件来实现,并且消耗大量处理器资源并且导致过多延迟,从而导致操作系统和/或应用显著变慢。传统的基于硬件的存储器压缩具有较差的数据布局和块分配机制,导致较高元数据开销和内部碎片化,从而导致压缩存储器利用率较低。因此,传统的存储器压缩方式可能会导致系统变慢或元数据开销较高。

[0017] 为此,提出了一种用于基于硬件的存储器压缩的新方法。在本公开的实施例中,压缩存储器被划分为多个段,每个段被划分为多个子段,并且未压缩数据空间中的每个子段被压缩为数据空间中的一个或多个块。在接收到对压缩存储器中的段中的子段的读请求时,首先基于子段与条目之间的第一级地址映射确定对应条目,然后基于条目与一个或多个块之间的第二级地址映射确定一个或多个对应块。

[0018] 根据本公开的实施例,通过使用两级地址映射,可以减小条目的大小,从而实现较低的元数据开销。此外,根据本公开的实施例的所提出的用于压缩存储器的数据布局能够在压缩存储器中实现稳定快速的块分配。此外,由于本公开的实施例使用基于硬件的存储器压缩,因此与基于软件的存储器压缩相比,可以避免系统和应用变慢。

[0019] 另外,在本公开的一些实施例中,通过增大每个子段(其是压缩存储器中的压缩单元)的大小,可以提高存储器压缩的压缩率,从而进一步提高存储器的利用率。本公开的实施例可以在每个条目中配置多个指针,可以减小存储器的存储单位(即,块大小),从而减少内部碎片。下面将参考示例实现描述本公开的实施例的其他优点。下面参考图1至图8来说明本文中的本公开的基本原理和若干示例实施例。

[0020] 图1示出了可以在其中实现本公开的一个或多个实施例的计算设备/服务器100的框图。应当理解,图1中描述的计算设备/服务器100仅用于说明,而不以任何方式限制本公开的功能和范围。计算设备/服务器100可以是计算机或服务器。

[0021] 如图1所示,计算设备/服务器100是通用计算设备的形式。计算设备/服务器100的组件可以包括但不限于一个或多个处理器或处理单元110、存储器120、存储装置130、一个或多个通信单元140、一个或多个输入设备150和一个或多个输出设备160。处理单元110可

以是物理或虚拟处理器,并且基于存储在存储器120中的程序执行各种处理。在多处理器系统中,多个处理单元可以并行执行计算机可执行指令以提高计算设备/服务器100的并行处理能力。

[0022] 计算设备/服务器100通常包括各种计算机存储介质。计算机存储介质可以是计算设备/服务器100可访问的任何介质,包括但不限于易失性和非易失性介质,或者可移动和不可移动介质。存储器120可以是易失性存储器(例如,寄存器、高速缓存、随机存取存储器(RAM))、非易失性存储器(例如,只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、闪存)或其任何组合。

[0023] 如图1所示,存储器120可以包括用于实现根据本公开的实施例的存储器压缩和块分配的程序125,程序125可以具有被配置为执行本文中描述的各种实施例的方法和功能的一组或多组程序模块。在一些实施例中,存储器120可以包括传统的常规存储器(诸如未压缩存储器)和根据本公开的实施例的所提出的压缩存储器,压缩存储器可以具有新的数据布局和块分配机制,这可以用于存储可能不频繁或不常访问的压缩数据。

[0024] 存储装置130可以是任何可移动或不可移动介质,并且可以包括机器可读介质,诸如闪存驱动器、磁盘和任何其他介质,机器可读介质可以用于存储信息和/或数据并且在计算设备/服务器100内访问。例如,存储装置130可以是硬盘驱动器(HDD)或固态驱动器(SSD)。

[0025] 计算设备/服务器100还可以包括附加的可移除/不可移除或易失性/非易失性存储介质。尽管图1中未示出,但是提供了磁盘驱动器以用于从可移动的非易失性磁盘(例如,“软盘”)读取/向其写入,并且可以提供光盘驱动器以用于从可移动的非易失性光盘读取/向其写入。在这种情况下,每个驱动器经由一个或多个数据媒体接口连接到总线(未示出)。

[0026] 通信单元140经由通信介质与另一计算设备通信。此外,计算设备/服务器100中的组件的功能可以在单个计算集群或经由通信连接彼此通信的多个计算机器中实现。因此,计算设备/服务器100可以使用到一个或多个其他服务器、网络个人计算机(PC)或另一网络节点的逻辑连接在网络环境中操作。

[0027] 输入设备150可以包括一个或多个输入设备,诸如鼠标、键盘、跟踪球等。输出设备160可以包括一个或多个输出设备,诸如显示器、扬声器、打印机等。计算设备/服务器100还可以经由通信单元140与一个或多个外部设备(未示出)通信,诸如存储设备或显示设备、使得用户能够与计算设备/服务器100交互的一个或多个设备、或使得计算设备/服务器100能够与一个或多个其他计算设备(例如,网卡、调制解调器等)通信的任何设备。这种通信可以经由输入/输出(I/O)接口(未示出)来执行。接下来,参考图2至图8来详细说明本公开的示例实施例。

[0028] 图2示出了根据本公开的实施例的具有基于图块的压缩存储器的示例架构200。如图2所示,架构200包括CPU 210、存储器控制器220、未压缩存储器231、压缩存储器232和软件应用240。CPU 210可以是用于执行程序指令的处理单元,诸如如图1所示的处理单元110。存储器控制器220是管理去往和来自未压缩存储器231和压缩存储器232的数据流的数字电路。存储器控制器可以是单独的芯片或集成到另一芯片中,诸如放置在同一管芯上或作为CPU 210的组成部分。存储器控制器可以管理对存储器的读取和写入操作,同时通过向存储器供应电流来保持存储器活动。

[0029] 未压缩存储器231是常规存储器,诸如传统的DRAM,并且未压缩存储器231可以用于存储频繁访问的数据,因为频繁访问的数据被频繁访问并且不适合压缩。压缩存储器232是根据本公开的实施例的新提出的存储器,并且可以用于存储不太频繁访问的数据以增加可以存储在存储器中的数据量。软件应用240包括一个或多个程序,该程序可以包括应用、库、驱动程序、操作系统、管理程序和/或其他组件。软件应用240中的一个或多个程序可以区分频繁访问的数据和不频繁访问的数据,并且可以控制哪些数据应当存储在未压缩存储器231中和哪些数据应当存储在压缩存储器232中。在本公开的实施例中,未压缩存储器231和压缩存储器232可以共享从CPU 210可访问的同一地址空间。

[0030] 在一些实施例中,数据的压缩和解压缩操作可以在压缩存储器232中实现,在这种情况下,本公开的实施例可以实现为存储器模块中的数据布局和块分配的新设计。替代地,压缩存储器232的压缩操作和解压缩操作可以在存储器控制器220中实现。

[0031] 根据示例架构200,CPU 210可以基于软件应用240的确定经由存储器控制器220向未压缩存储器231或压缩存储器232传输数据。本公开的实施例提出了用于压缩存储器232的新的数据布局和块分配算法,该算法可以减少压缩存储器的元数据开销并且实现压缩存储器中的快速块分配。

[0032] 图3示出了根据本公开的实施例的用于基于硬件的存储器压缩的方法300的流程图。应当理解,方法300可以由参考图1描述的计算设备/服务器100的存储器120或参考图2描述的压缩存储器232或存储器控制器220来执行。

[0033] 在302处,接收对存储器的读请求,并且该读请求与存储器中的段的子段相关。在本文中的本公开的实施例中,压缩存储器被划分为多个段,并且每个段可以具有第一预定义大小(诸如32KiB)并且可以称为“图块”(tile),其中存储器例如可以具有4GiB的容量。假定压缩存储器的压缩比为2:1,4GiB的压缩存储器可以存储8GiB的未压缩数据。

[0034] 在本文中的本公开的实施例中,每个段被划分为多个子段,并且每个子段可以具有第二预定义大小(诸如512字节)并且可以称为“段落”(paragraph)。未压缩数据空间中的每个子段在压缩数据空间中被压缩为一个或多个块,并且每个块可以具有第三预定义大小(诸如64字节)。应当理解,虽然在一些实施例中描述了存储器、图块、段落和块的示例大小,但是其他大小也是可能的,这取决于开发者和/或管理员的手动配置。图块是固定大小的自包含的数据结构,其包括根条目、块和位图。应当理解,虽然本公开的实施例使用术语“图块”和“段落”来表示概念“段”和“子段”,但是其他术语或词语也可以表示概念“段”和“子段”。

[0035] 在304处,确定段中与子段相对应的条目,并且该条目包括指示符(诸如一个或多个指针)。在本公开的一些实施例中,条目可以位于图块的根中,来作为图块的自包含数据结构的一部分,并且条目也可以称为“根条目”。例如,首先基于段落与条目之间的第一级地址映射来确定对应的根条目。在一些实施例中,第一级地址映射可以是固定映射规则,并且该映射规则可以包括段落与图块内的根条目之间的静态映射。一旦知道段落,对应的根条目就可以被定位于要访问的特定图块内。以这种方式,与动态映射相比,可以减少根条目的存储开销。在一些实施例中,固定映射规则可以以交错方式配置。例如,段落地址的最后几位用作根条目的索引,使得连续的段落广泛分布。以这种方式,可以减少或更好地处理图块中的溢出事件。替代地,固定映射规则可以是线性映射(例如,连续段落具有相似的压缩比)

或随机映射(诸如CRC32算法)。

[0036] 在306处,基于条目中的指示符确定存储与读请求相关联的压缩数据的一个或多个块。例如,可以基于根条目与一个或多个块之间的第二级地址映射来确定一个或多个对应块。第二级地址映射可以是包括根条目与一个或多个块之间的动态映射的动态映射规则。以这种方式,可以实现快速块分配。根据本公开的方法300,通过使用两级地址映射,可以减小条目的大小,从而实现较低元数据开销并且节省存储器空间。此外,由于压缩存储器可以比未压缩存储器存储更多数据,因此本公开的实施例可以降低存储器成本和/或增大存储器容量。

[0037] 在一些实施例中,如果所确定的根条目包括两个或更多个指针,则将确定两个或更多个对应块,并且可以从两个或更多个块中并行读取压缩数据。以这种方式,可以并行取回数据块,从而提高压缩存储器的读取速度。即,在读取根条目之后,可以确定第二级的块,并且可以并行发出多个块读取以减少延迟。

[0038] 此外,由于一个根条目可以包括多个指针,因此可以从一个条目确定多个块。即,一个段落可以被压缩为很多块,例如,1个块、2个块、4个块、8个块等。以这种方式,通过使得段落大小大于块大小,可以配置多个块以存储从一个段落压缩而来的压缩数据,并且从而可以减少内部碎片。

[0039] 图4示出了根据本公开的实施例的针对图块的两级地址映射的示意图400。如图4所示,压缩存储器包括多个图块,诸如图块410、420等,并且图块中的段落到块的地址映射通过两级地址映射来实现,其中根条目用作两级地址映射的间接(indirection)。

[0040] 参考图4,图块410可以包括多个根条目(诸如根条目411)、多个块(诸如块412、413)和位图(诸如位图414),根条目用于定位对应块,块用于存储压缩数据,并且位图用于指示图块410中块的分配状态。例如,位图414中的每一位可以指示图块410中的每个块是否被分配。在图4的示例中,首先,基于固定映射将图块中的一个段落映射到根条目(诸如根条目411),如箭头416所示,其中每个段落对应于一个根条目,并且根条目411包括一个或多个指针,以用于寻找对应块。其次,基于动态映射将根条目411映射到块412和413,如箭头417和418所示。通过将动态映射限制到图块,压缩存储器中块的分配可以是稳定快速的。

[0041] 在一些实施例中,根条目411中的指针的最大数目可以基于段落大小与块大小之间的第一比率来确定,而根条目411中的每个指针的大小可以基于图块大小与块大小之间的第二比率来确定。例如,在其中每个图块的大小为32KiB,每个段落的大小为512个字节,并且每个块的大小为64个字节的以上示例中,每个根条目的大小为8\*9位,即9个字节,其中8表示一个根条目中的指针的最大数目,并且9位表示每个指针的大小。以这种方式,根据本公开的实施例的根条目(即,元数据)只占用很小的存储器空间,从而节省了压缩存储器中的元数据开销。

[0042] 图5示出了根据本公开的实施例的从段落到一个或多个块的示例数据压缩的示意图500。在本公开的实施例中,存储器分配的单位是大小为例如32KiB的图块,未压缩数据空间中的压缩单位是大小为例如512字节的段落,并且压缩数据空间中的存储器访问的单位是大小为64字节的块。在该示例中,一个段落的大小被预设为例如一个块的八倍。以这种方式,通过增大每个段落的大小,可以提高存储器压缩的压缩率,因为较大的压缩单元通常比较小的压缩单元具有更大的压缩率。

[0043] 参考图5,每个图块可以被划分为多个段落,诸如段落510和520,并且段落可以由处理器和应用查看的数据对象。如图5所示,未压缩数据空间中的段落510可以被压缩为压缩数据空间中的四个块,诸如块511、512、513和514,而未压缩数据空间中的段落520可以被压缩为压缩数据空间中的三个块,诸如块521、522和523。如图5所示,不同数据可以有不同压缩率,这取决于数据的类型和数据中的实际内容。换言之,块的实际数量取决于个体段落的压缩率。

[0044] 图6示出了根据本公开的实施例的图块610中的示例数据布局600。如图所示,图块610包括多个根条目620(诸如根条目621和622)、多个数据块630和位图640。在示例数据布局600中,图块610的大小可以为例如32KiB,每个段落的大小为512字节,每个块的大小为64字节。

[0045] 在以上示例中,图块610总共具有512个块。假定压缩比被预设2:1,图块610可以具有128个段落和128个根条目。每个根条目可以包括多达8个指针,并且每个指针的大小为9位。这样,每个段落可以对应于大小高达9字节的根条目,128个段落将占用128\*9个字节的存储器空间(对应于18个块),每个段落可以压缩为多达8个块。位图640用于控制压缩存储器中块的分配,图块610中的512个块将需要512位,512位对应于1个块。因此,图块610具有用于存储根条目的18个块和用于存储位图的1个块,总共占用19个块用于存储元数据,因此将剩余493个块用于存储压缩数据,诸如图6中ID为“18”-“510”的块。以这种方式,可以提高存储器的利用率。

[0046] 参考图6,位图640指示一些块被分配而其他块未被分配,其中位“1”表示对应块已经被分配。如果接收到对图块610的写请求,将首先检查位图640以分配一个或多个块来存储压缩数据,然后,用所分配的块的ID来写对应的根条目,位图640也将被修改。如果接收到对图块610中的段落的读请求,则首先基于第一级固定映射确定对应根条目(诸如根条目621),然后将基于第二级动态映射确定相关块(诸如ID为“18”、“19”、“20”和“21”的块)。以这种方式,可以快速分配和标识块,从而提高压缩存储器的存储效率。

[0047] 图7示出了根据本公开的实施例的用于写压缩数据的方法700的流程图。与图3的方法300中对图块的读请求的处理相比,方法700描述了对存储器的写请求的处理方法。应当理解,方法700可以由参考图1描述的计算设备/服务器100的存储器120或由参考图2描述的压缩存储器232或存储器控制器220来执行。

[0048] 在702处,在接收到对压缩存储器中的存储器的写请求时,执行从图块到根条目的地址转换。例如,地址转换器可以根据未压缩存储器空间中的物理地址来计算压缩存储器空间中的根条目地址。

[0049] 在704处,检查图块的位图,并且位图中的每个位指示图块中的每个块是否被分配。在706处,根据位图,确定图块中是否有足够的块可用于写请求。如果位图指示图块中有足够块可用于写请求,则在708处,基于位图将与写请求相关联的压缩数据写入图块中的一个或多个块中。然后,在710处,位图和根条目也被写和更新,并且压缩数据被成功地存储在压缩存储器中。

[0050] 如果位图指示图块中没有足够块可用于写请求,则在708处,需要找到存储器中的另一图块,该另一图块具有足够的可用于写请求的块。在这种情况下,这表示该图块中的块已经被消费并且发生溢出事件。然后,在714处,将与写请求相关联的压缩数据的至少一部

分写入另一图块中的一个或多个块中。以这种方式,可以正确处理溢出事件,以避免压缩存储器中的数据丢失。在一些实施例中,常规根条目中的特殊值(例如,511)可以用于表示溢出根条目,并且该特殊值后跟指针,以引用溢出图块中的另一块。溢出图块中的块可以用作根条目,以引用溢出图块中的压缩数据块。在一些实施例中,可以使用另一特殊值(例如,0)来指示空的根条目,这表示该段落未被分配或者存储器内容的所有位为零。

[0051] 图8示出了根据本公开的实施例的用于实现存储器压缩的示例硬件架构800。图8示出了可以在存储器控制器中实现的本公开的示例,然而,本公开的实施例也可以在压缩存储器中实现。示例硬件架构800可以是经由外围组件互连快速(PCIe)总线附接到主机的现场可编程门阵列(FPGA)板。虚拟机管理程序将FPGA的PCIe存储器映射I/O(MMIO)空间映射到软件,因此CPU可以直接访问压缩存储器。此外,CPU也可以直接访问未压缩存储器。

[0052] 如图8所示,示例硬件架构800包括平台包装器810、PCIe/UPI820、包括高速缓存/缓冲器831的存储器控制器830、写控制器832、读控制器833和DRAM 840。存储器控制器830可以控制来自或去往DRAM 840的数据,并且可以经由PCIe/UPI 820与平台包装器810通信。包装器810可以用于在读请求期间读取根条目和块中的压缩数据,并且包装器810可以用于在写请求期间读取位图。

[0053] 高速缓存/缓冲器831存储正在被读/写或最近被访问的段落,并且它可以有两个目的。首先,高速缓存/缓冲器831在同一段落上找出冲突的读/写操作,并且将它们排队到每个段落的FIFO中,使得这些操作可以按顺序执行。高速缓存/缓冲器831只对存储器控制器830的剩余部分进行独立的读/写操作,因此可以避免由并发操作造成的不一致。其次,高速缓存/缓冲器831高速缓存最近访问的段落以减轻访问放大问题。由于CPU访问粒度(高速缓存行大小或指令字大小)通常小于段落大小,因此CPU在顺序扫描段落时会发出多个读/写操作。通过使用高速缓存,段落的压缩块不需要从DRAM中被多次加载和解压,从而减少了开销。

[0054] 读控制器833接收来自高速缓存/缓冲器831的读请求,并且根据读请求中的地址计算压缩存储器空间中的根条目地址。然后,读控制器833访问根条目,并且确定图块中的块偏移并且计算压缩存储器空间中的块地址。然后,它访问块并且将数据馈送到解压缩引擎,并且解压缩数据被存储在高速缓存/缓冲器831中并且被发送到CPU。

[0055] 写控制器832接收来自高速缓存/缓冲器831的写请求,并且根据写请求中的地址计算压缩存储器空间中的根条目地址。然后,写控制器832访问根条目和位图,确定图块中的块偏移,并且计算压缩存储器空间中的块地址。写控制器832根据写请求修改数据,然后将数据馈送到压缩引擎。在压缩之后,根据压缩数据大小的变化,从块分配器中分配或取消分配一个或多个块。最后,压缩数据被写入图块中的块。

[0056] 压缩引擎可以执行数据的压缩和解压缩。为了加速并行请求,它可以在内部具有并行运行的多个压缩引擎。根据本公开的实施例,并行请求的顺序并不重要,因为高速缓存/缓冲器831已经过滤了冲突操作。

[0057] 块分配器可以管理每个图块中的位图。为了加快分配速度,块分配器在内部维护位图高速缓存,以缓存最近访问的图块和具有大量可用块的若干图块的位图。对于解除分配,对应位可以设置为零。对于分配,它首先加载对应图块的位图。如果有足够的可用块,则设置对应位。否则,它检查位图高速缓存以找到具有足够可用块的溢出图块。如果未找到这

样的图块,则它扫描多个图块中的位图,并且如果未找到溢出图块,则报告分配失败。

[0058] 本文中描述的功能可以至少部分由一个或多个硬件逻辑组件来执行。例如而非限制,可以使用的说明性类型的硬件逻辑组件包括现场可编程门阵列 (FPGA)、专用集成电路 (ASIC)、专用标准产品 (ASSP)、系统级芯片系统 (SOC)、复杂可编程逻辑器件 (CPLD) 等。

[0059] 用于执行本公开的方法的程序代码可以用一种或多种编程语言的任何组合来编写。这些程序代码可以被提供给通用计算机、专用计算机或其他可编程数据处理装置的处理器的控制器,使得该程序代码在由处理器或控制器执行时引起在流程图和/或框图中指定的功能/操作被实现。程序代码可以完全在机器上、部分在机器上、作为独立软件包、部分在机器上并且部分在远程机器上、或者完全在远程机器或服务服务器上执行。

[0060] 在本公开的上下文中,机器可读介质可以是包含或存储由指令执行系统、装置或设备使用或与其结合使用的程序的任何有形介质。机器可读介质可以是机器可读信号介质或机器可读存储介质。机器可读介质可以包括但不限于电子、磁、光、电磁、红外或半导体系统、装置或设备、或前述各项的任何合适的组合。机器可读存储介质的更具体示例将包括具有一根或多根电线的电连接、便携式计算机软盘、硬盘、随机存取存储器 (RAM)、只读存储器 (ROM)、可擦除可编程只读存储器 (EPROM或闪存)、光纤、便携式光盘只读存储器 (CD-ROM)、光存储设备、磁存储设备、或前述各项的任何合适的组合。

[0061] 此外,虽然以特定顺序描述了操作,但这不应当被理解为要求这样的操作以所示出的特定顺序或按顺序执行,或者要求执行所有所示出的操作,以实现期望结果。在某些情况下,多任务和并行处理可能是有利的。同样地,虽然上述讨论中包含了若干具体实现细节,但这些不应当被解释为对本公开的范围的限制,而是对可以特定于特定实施例的特征的描述。在单独实施例的上下文中描述的某些特征也可以在单个实现中组合实现。相反,在单个实现的上下文中描述的各种特征也可以单独地或以任何合适的子组合在多个实施例中实现。

[0062] 下面列出了本公开的一些示例实施例。

[0063] 在一个方面,提供了一种用于存储器压缩的方法。所述方法包括:接收对存储器的读请求,其中所述存储器被划分为多个段,所述多个段中的一个段被划分为多个子段,并且所述多个子段中的一个子段被压缩为一个或多个块,所述读请求与所述存储器中的段的子段相关;在所述段中确定与所述子段相对应的条目,其中所述条目包括指示符;以及基于所述条目中的所述指示符确定存储与所述读请求相关联的压缩数据的块。

[0064] 在一些实施例中,其中在所述段中确定与所述子段相对应的条目包括:基于映射规则确定与所述子段相对应的所述条目,其中所述映射规则包括子段与条目之间的在所述段内的静态映射。

[0065] 在一些实施例中,其中一个子段的大小被预设为一个块的大小的至少八倍。

[0066] 在一些实施例中,其中基于所述条目中的所述指示符确定存储与所述读请求相关联的压缩数据的块包括:基于所述条目中的两个或更多个指针确定与所述读请求相关联的两个或更多个块,其中所述条目中的指针的最大数目基于一个子段的大小与一个块的大小之间的第一比率来确定,并且所述条目中的一个指针的大小基于一个段的大小与一个块的大小之间的第二比率来确定。

[0067] 在一些实施例中,所述方法还可以包括:响应于确定与所述读请求相关联的两个

或更多个块,从所述两个或更多个块并行读取所述压缩数据;以及解压缩所述压缩数据以生成解压缩数据。

[0068] 在一些实施例中,所述方法还可以包括:接收对所述存储器的写请求;响应于接收到所述写请求,检查段中的位图,其中所述段包括多个条目、多个块和所述位图,并且所述位图中的每个位指示所述段中的每个块是否被分配;响应于所述位图指示所述段中有足够的块可用于所述写请求,基于所述位图将与所述写请求相关联的压缩数据写入所述段中的一个或多个块中;以及更新所述位图和所述段中的一个或多个条目。

[0069] 在一些实施例中,所述方法还可以包括:响应于所述位图指示所述段中没有足够的块可用于所述写请求,确定所述存储器中具有足够可用块的另外的段;以及将与所述写请求相关联的压缩数据的至少一部分写入所述另外的段中的一个或多个块中。

[0070] 在一些实施例中,其中不被频繁访问的数据被迁移到压缩存储器,而被频繁访问的数据被迁移到未压缩存储器,并且所述未压缩存储器和所述压缩存储器共享从处理器可访问的同一地址空间。

[0071] 在一些实施例中,其中所述方法在所述存储器中实现,并且对所述存储器中的所述段的所述子段的所述读请求经由存储器控制器而被接收。

[0072] 在另一方面,提供了一种存储器。所述存储器包括存储器芯片和被配置为执行动作的控制器。所述动作包括:接收对所述存储器芯片的读请求,其中所述存储器芯片被划分为多个段,所述多个段中的一个段被划分为多个子段,并且所述多个子段中的一个子段被压缩为一个或多个块,所述读请求与所述存储器中的段的子段相关;在所述段中确定与所述子段相对应的条目,所述条目包括指示符;以及基于所述条目中的所述指示符确定存储与所述读请求相关联的压缩数据的块。

[0073] 在一些实施例中,其中在所述段中确定与所述子段相对应的条目包括:基于映射规则确定与所述子段相对应的所述条目,其中所述映射规则包括子段与条目之间的在所述段内的静态映射。

[0074] 在一些实施例中,其中一个子段的大小被预设为一个块的大小的至少八倍。

[0075] 在一些实施例中,其中基于所述条目中的所述指示符确定存储与所述读请求相关联的压缩数据的块包括:基于所述条目中的两个或更多个指针确定与所述读请求相关联的两个或更多个块,其中所述条目中的指针的最大数目基于一个子段的大小与一个块的大小之间的第一比率来确定,并且所述条目中的一个指针的大小基于一个段的大小与一个块的大小之间的第二比率来确定。

[0076] 在一些实施例中,所述动作还可以包括:响应于确定与所述读请求相关联的两个或更多个块,从所述两个或更多个块并行读取所述压缩数据;以及解压缩所述压缩数据以生成解压缩数据。

[0077] 在一些实施例中,所述动作还可以包括:接收对所述存储器的写请求;响应于接收到所述写请求,检查段中的位图,其中所述段包括多个条目、多个块和所述位图,并且所述位图中的每个位指示所述段中的每个块是否被分配;响应于所述位图指示所述段中有足够的块可用于所述写请求,基于所述位图将与所述写请求相关联的压缩数据写入所述段中的一个或多个块中;以及更新所述位图和所述段中的一个或多个条目。

[0078] 在一些实施例中,所述动作还可以包括:响应于所述位图指示所述段中没有足够

的块可用于所述写请求,确定所述存储器中具有足够可用块的另外的段;以及将与所述写请求相关联的压缩数据的至少一部分写入所述另外的段中的一个或多个块中。

[0079] 在一些实施例中,其中不被频繁访问的数据被迁移到压缩存储器,而被频繁访问的数据被迁移到未压缩存储器,并且所述未压缩存储器和所述压缩存储器共享从处理器可访问的同一地址空间。

[0080] 在另一方面,提供了一种程序产品。所述程序产品包括可执行指令,所述可执行指令当在设备上执行时使所述设备执行动作。所述动作包括:接收对存储器的读请求,其中所述存储器被划分为多个段,所述多个段中的一个段被划分为多个子段,并且所述多个子段中的一个子段被压缩为一个或多个块,所述读请求与所述存储器中的段的子段相关;在所述段中确定与所述子段相对应的条目,所述条目包括指示符;以及基于所述条目中的所述指示符确定存储与所述读请求相关联的压缩数据的块。

[0081] 在一些实施例中,其中在所述段中确定与所述子段相对应的条目包括:基于映射规则确定与所述子段相对应的所述条目,其中所述映射规则包括子段与条目之间的在所述段内的静态映射。

[0082] 在一些实施例中,其中一个子段的大小被预设为一个块的大小的至少八倍。

[0083] 在一些实施例中,其中基于所述条目中的所述指示符确定存储与所述读请求相关联的压缩数据的块包括:基于所述条目中的两个或更多个指针确定与所述读请求相关联的两个或更多个块,其中所述条目中的指针的最大数目基于一个子段的大小与一个块的大小之间的第一比率来确定,并且所述条目中的一个指针的大小基于一个段的大小与一个块的大小之间的第二比率来确定。

[0084] 在一些实施例中,所述动作还可以包括:响应于确定与所述读请求相关联的两个或更多个块,从所述两个或更多个块并行读取所述压缩数据;以及解压缩所述压缩数据以生成解压缩数据。

[0085] 在一些实施例中,所述动作还可以包括:接收对所述存储器的写请求;响应于接收到所述写请求,其中检查段中的位图,所述段包括多个条目、多个块和所述位图,并且所述位图中的每个位指示所述段中的每个块是否被分配;响应于所述位图指示所述段中有足够的块可用于所述写请求,基于所述位图将与所述写请求相关联的压缩数据写入所述段中的一个或多个块中;以及更新所述位图和所述段中的一个或多个条目。

[0086] 在一些实施例中,所述动作还可以包括:响应于所述位图指示所述段中没有足够的块可用于所述写请求,确定所述存储器中具有足够可用块的另外的段;以及将与所述写请求相关联的压缩数据的至少一部分写入所述另外的段中的一个或多个块中。

[0087] 在一些实施例中,其中不被频繁访问的数据被迁移到压缩存储器,而被频繁访问的数据被迁移到未压缩存储器,并且所述未压缩存储器和所述压缩存储器共享从处理器可访问的同一地址空间。

[0088] 虽然本公开已经以特定于结构特征和/或方法动作的语言进行了描述,但是应当理解,在所附权利要求中指定的主题不一定限于上述特定特征或动作。相反,上述特定特征和动作被公开作为实现权利要求的示例形式。

[0089] 本公开的各种实施例的描述是出于说明的目的而呈现的,但并不旨在穷举或限于所公开的实施例。在不脱离所描述的实施例的范围和精神的情况下,很多修改和变化对于

本领域普通技术人员来说将是很清楚的。

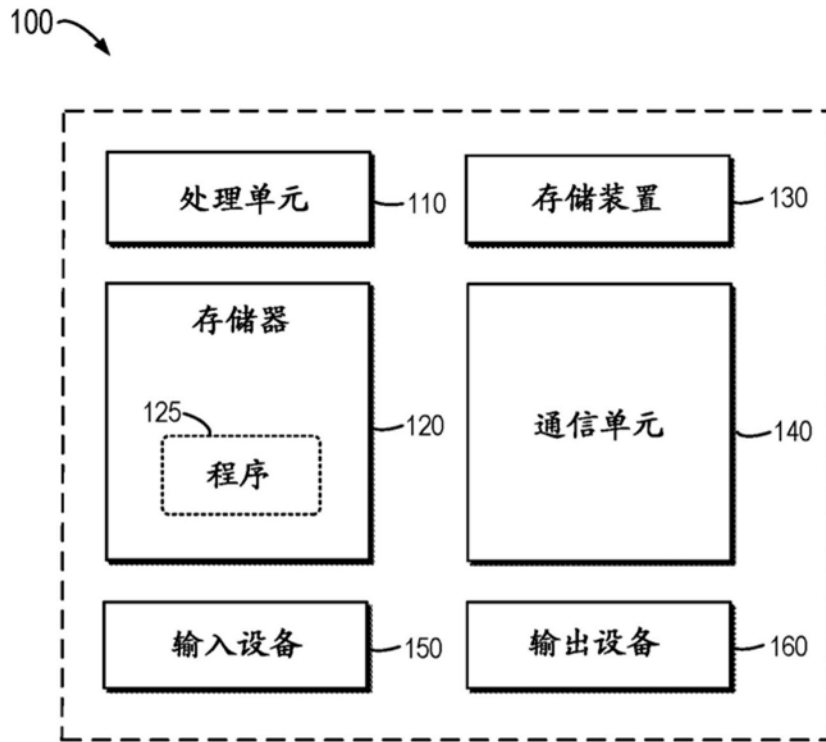


图1

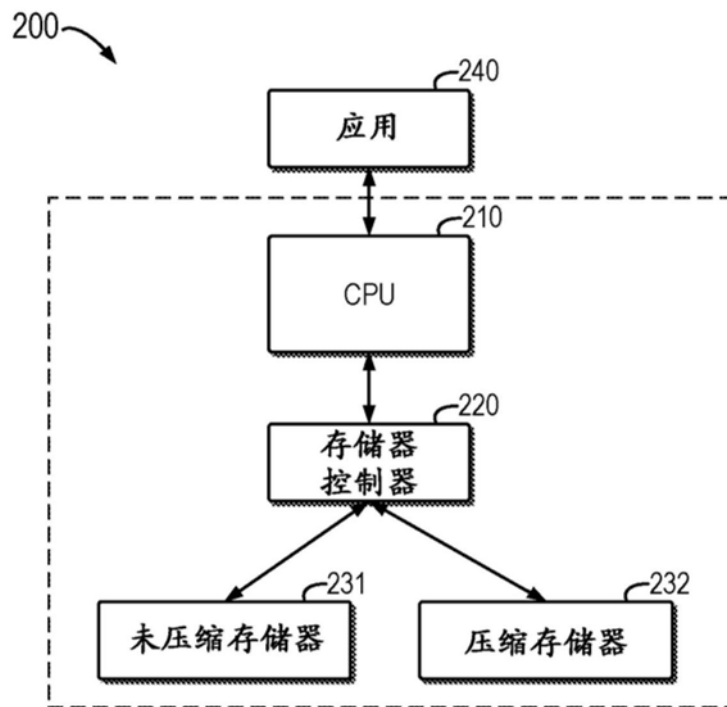


图2

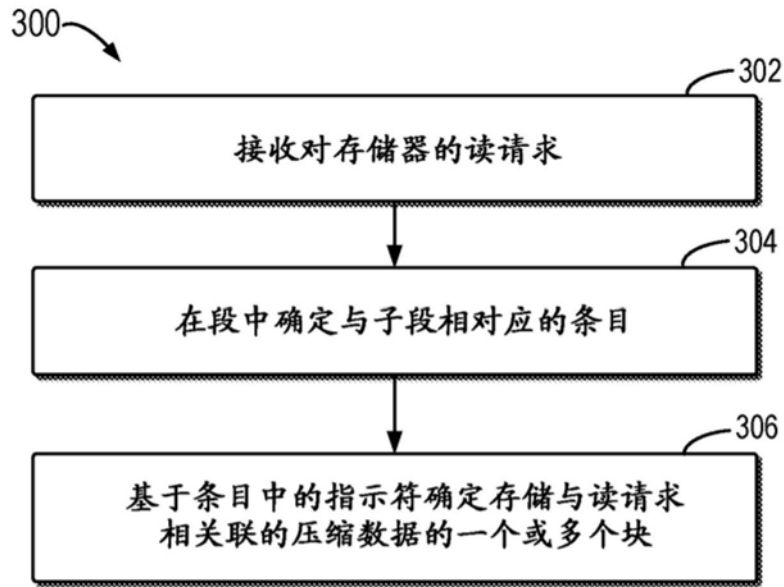


图3

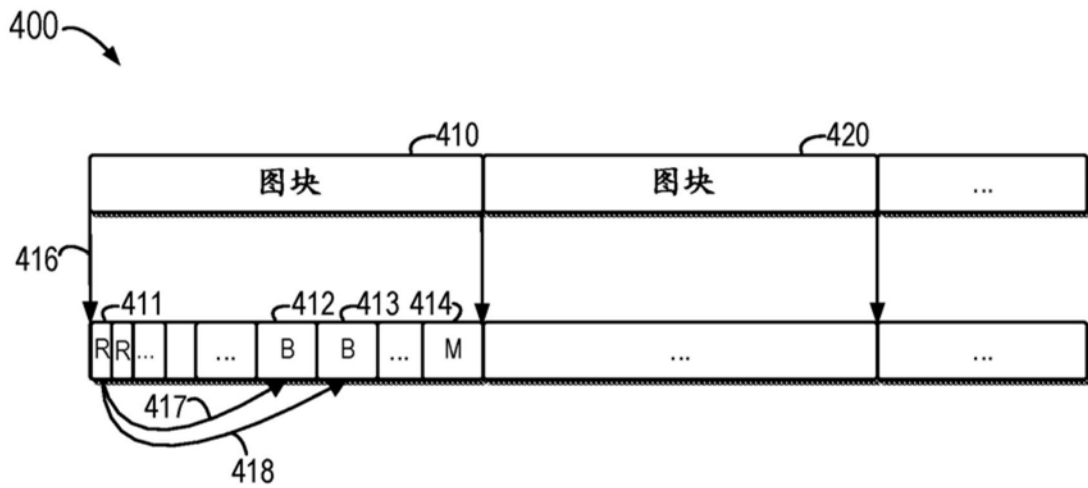


图4

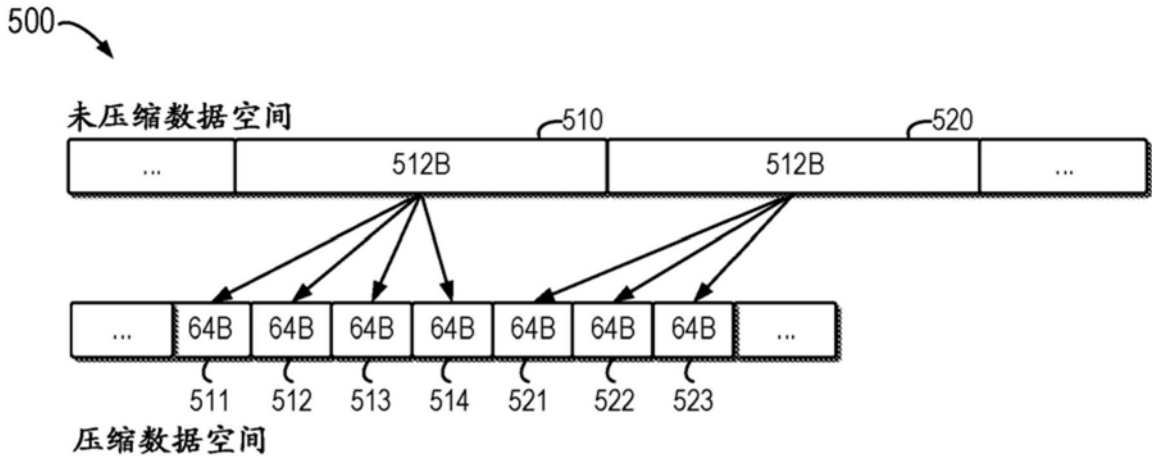


图5

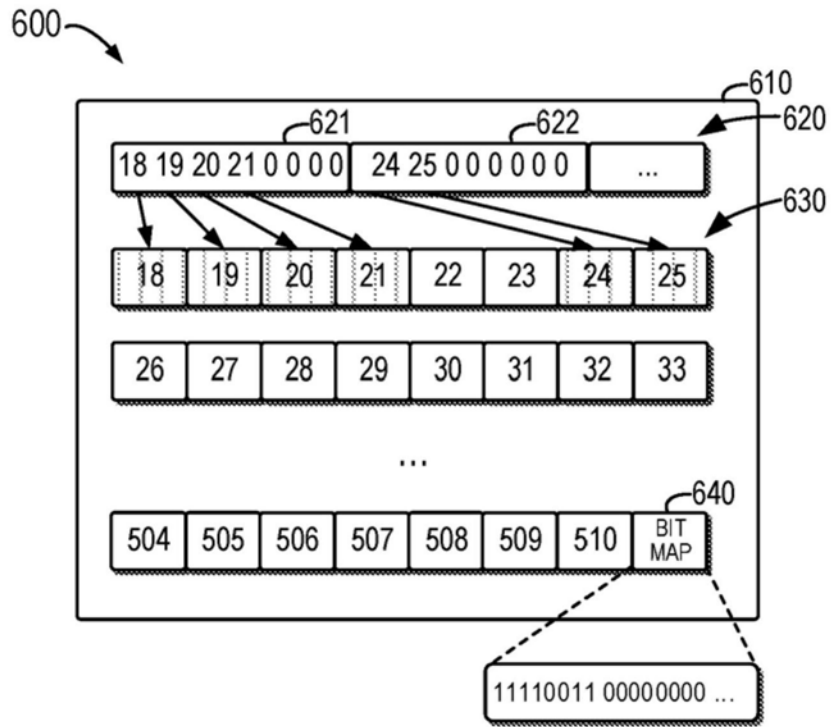


图6

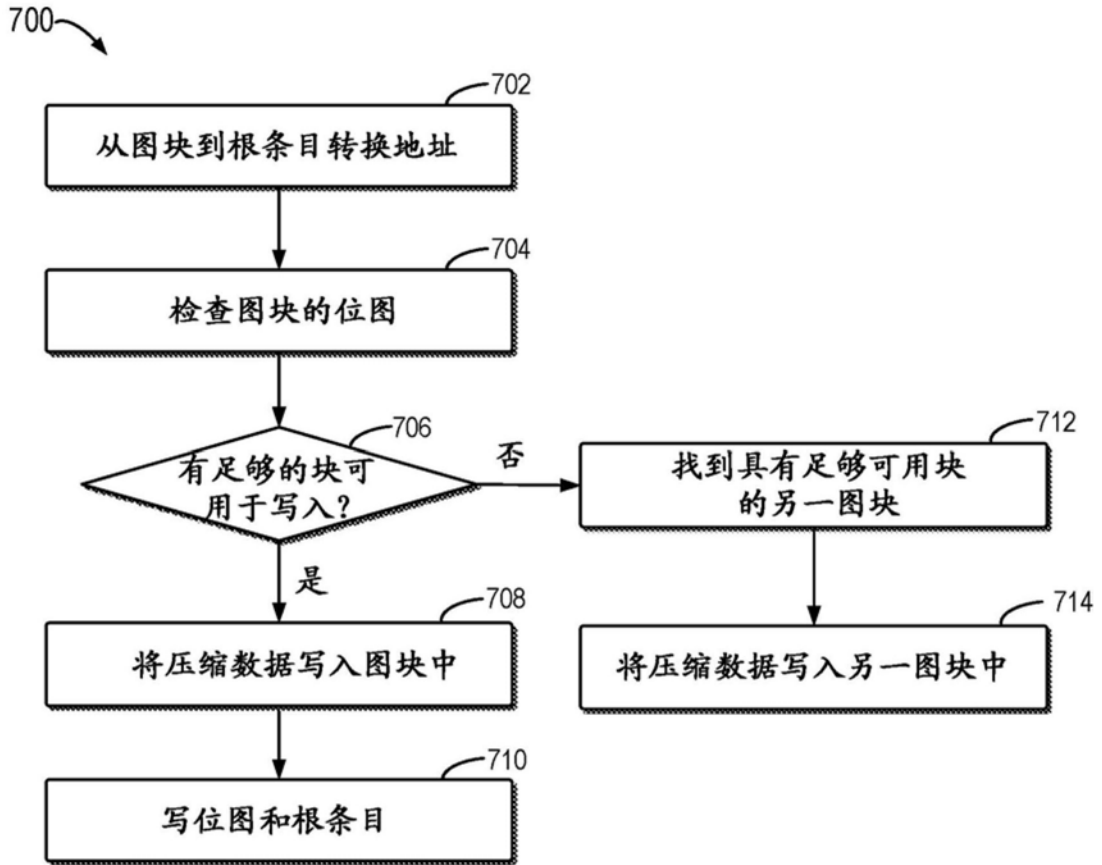


图7

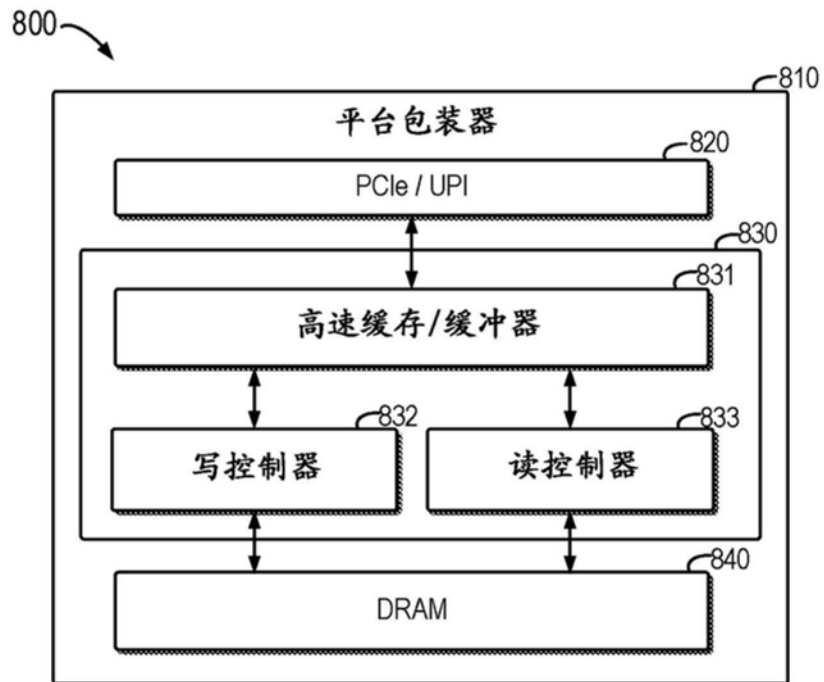


图8