



(12)发明专利

(10)授权公告号 CN 103403681 B

(45)授权公告日 2018.09.18

(21)申请号 201180066712.8

(72)发明人 李智刚 S-H·F·奥 朴正日
申炫锡

(22)申请日 2011.12.19

(74)专利代理机构 北京市金杜律师事务所
11256

(65)同一申请的已公布的文献号
申请公布号 CN 103403681 A

代理人 鄢迅 辛鸣

(43)申请公布日 2013.11.20

(51)Int.Cl.
G06F 9/48(2006.01)

(30)优先权数据
61/425,098 2010.12.20 US

(56)对比文件

(85)PCT国际申请进入国家阶段日
2013.08.01

US 2010058003 A1,2010.03.04,
US 2009077305 A1,2009.03.19,
US 2010043009 A1,2010.02.18,
US 2009089492 A1,2009.04.02,
CN 101512496 A,2009.08.19,

(86)PCT国际申请的申请数据
PCT/US2011/065913 2011.12.19

审查员 郭妍捷

(87)PCT国际申请的公布数据
W02012/087971 EN 2012.06.28

(73)专利权人 马维尔国际贸易有限公司
地址 巴巴多斯圣米加勒

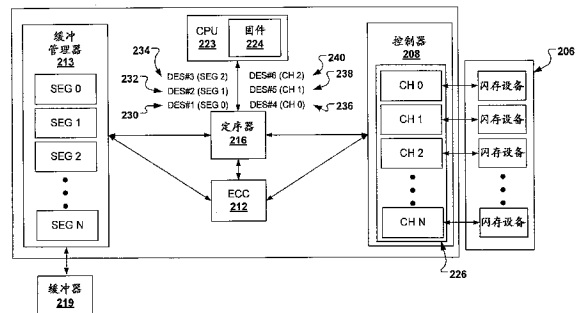
权利要求书3页 说明书18页 附图4页

(54)发明名称

描述符调度器

(57)摘要

描述符用于提供调度作业描述符的定序器的方法、系统和计算机程序产品。定序器可以基于作业描述符的相应分段和通道的可用性管理调度作业描述符以用于执行。例如,定序器可以检查分段的状态,并且标识在忙碌或者满状态中的一个或者多个分段或者在非忙碌或者空状态中的一个或者多个分段。基于状态检查,定序器可以乱序执行作业描述符,并且特别地向关联分段可用的作业描述符给予超过关联分段在忙碌或者满状态中的作业描述符的优先级。在这样做时,可以更快处理未决作业描述符并且可以避免不必要延时。



1. 一种用于存储器管理的设备,包括:

缓冲管理器,其具有多个分段,每个分段存储与多个作业描述符中的不同作业描述符关联的数据,以用于暂时存储与所述不同作业描述符关联的数据,所述数据包括待写入到多个闪存设备中的闪存设备的数据或者从所述闪存设备待读取的数据;

控制器,其控制所述多个闪存设备,每个闪存设备操作地耦合到对应通道,所述控制器被配置用于控制通过所述对应通道向或者从所述多个闪存设备传达与相应作业描述符关联的数据;以及

定序器,其:

以第一顺序接收所述多个作业描述符,其中每个作业描述符指定来自所述多个分段的服务于与所述作业描述符关联的数据的目标分段,和用于从来自所述多个闪存设备的闪存设备取回数据或者向来自所述多个闪存设备的闪存设备写入数据的目标通道;

针对以所述第一顺序接收的所述作业描述符中的每一个,基于来自所述缓冲管理器的分段可用性的更新和来自所述控制器的通道可用性的更新来确定所述目标分段和所述目标通道的可用性;和

基于所述多个作业描述符的相应的所述目标分段和所述目标通道的可用性,以第二顺序来调度所述多个作业描述符的执行,其中所述第二顺序不同于所述第一顺序。

2. 根据权利要求1所述的设备,其中所述多个作业描述符包括第一作业描述符和第二作业描述符,所述第一作业描述符指定服务于与所述第一作业描述符关联的数据的第一分段,所述第二作业描述符指定服务于与所述第二作业描述符关联的数据的第二分段。

3. 根据权利要求2所述的设备,其中如果所述第二分段可用并且所述第一分段不可用,则所述定序器调度在所述第一作业描述符之前执行所述第二作业描述符。

4. 根据权利要求2所述的设备,其中如果所述第一分段可用并且所述第二分段不可用,则所述定序器调度在所述第二作业描述符之前执行所述第一作业描述符。

5. 根据权利要求2所述的设备,其中:

所述多个作业描述符包括第三作业描述符,所述第三作业描述符指定服务于与所述第三作业描述符关联的数据的第三分段;并且

如果所述第一分段和所述第二分段不可用并且所述第三分段可用,则所述定序器调度在所述第一作业描述符和所述第二作业描述符之前执行所述第三作业描述符。

6. 根据权利要求2所述的设备,其中:

所述多个作业描述符包括指定第三分段的第三作业描述符;并且

如果所述第一分段和所述第三分段可用并且所述第二分段不可用,则所述定序器调度在所述第一作业描述符之后、但是在所述第二作业描述符之前执行所述第三作业描述符。

7. 根据权利要求1所述的设备,其中所述多个作业描述符包括第一作业描述符和第二作业描述符,所述第一作业描述符指定将用于向第一闪存设备传输数据或者从所述第一闪存设备接收数据的第一通道,所述第二作业描述符指定与所述第一通道不同的第二通道,所述第二通道将用于向与所述第一闪存设备不同的第二闪存设备传输数据或者从所述第二闪存设备接收数据。

8. 根据权利要求7所述的设备,其中如果所述第二通道可用并且所述第一通道不可用,则所述定序器调度在所述第一作业描述符之前执行所述第二作业描述符。

9. 一种用于存储器管理的方法,包括:

通过缓冲管理器控制多个分段,每个分段与多个作业描述符中的不同作业描述符关联,以用于暂时存储与所述不同作业描述符关联的数据;

通过定序器以第一顺序接收所述多个作业描述符,其中每个作业描述符指定来自所述多个分段的服务于与所述作业描述符关联的数据的目标分段和用于从来自多个闪存设备的第一闪存设备取回数据或者向来自多个闪存设备的第一闪存设备写入数据的目标通道;

通过所述定序器,针对以所述第一顺序接收的每个作业描述符,基于来自所述缓冲管理器的分段可用性的更新和来自控制器的通道可用性的更新来确定所述目标分段的和所述目标通道的可用性;以及

基于所述多个作业描述符的相应的所述目标分段和所述目标通道的可用性以第二顺序来调度所述多个作业描述符的执行,其中所述第二顺序不同于所述第一顺序。

10. 根据权利要求9所述的方法,其中:

所接收的作业描述符被包括在命令中,并且指定分段或指令中的至少一项。

11. 根据权利要求10所述的方法,还包括确定第一通道和第二通道的通道状态,

其中调度第一命令和第二命令的处理包括如果所述通道状态指示所述第二通道可用并且所述第一通道不可用,则调度在第一作业描述符之前处理第二作业描述符。

12. 根据权利要求10所述的方法,还包括确定第一通道和第二通道的通道状态,

其中调度第一命令和第二命令的处理包括如果所述通道状态指示所述第一通道可用并且所述第二通道不可用,则调度在第二作业描述符之前处理第一作业描述符。

13. 根据权利要求10所述的方法,还包括:

接收包括第三作业描述符的第三命令,所述第三作业描述符指定用于通过第三通道从第三闪存设备取回数据或者向所述第三闪存设备写入数据的第三指令。

14. 根据权利要求13所述的方法,还包括确定第一通道、第二通道和所述第三通道的通道状态,

其中调度第一作业描述符、第二作业描述符和所述第三作业描述符的处理包括如果所述第一通道和所述第二通道不可用并且所述第三通道可用,则调度在所述第一作业描述符和所述第二作业描述符之前处理所述第三作业描述符。

15. 根据权利要求13所述的方法,还包括确定第一通道、第二通道和所述第三通道的通道状态,

其中调度第一作业描述符、第二作业描述符和所述第三作业描述符的处理包括如果所述通道状态指示所述第一通道和所述第三通道可用并且所述第二通道不可用,则调度在所述第一作业描述符之后、但是在所述第二作业描述符之前处理所述第三作业描述符。

16. 根据权利要求10所述的方法,还包括确定第一分段和第二分段的分段状态,

其中调度第一命令和第二命令的处理包括如果所述第二分段可用并且所述第一分段不可用,则调度在第一作业描述符之前处理第二作业描述符。

17. 根据权利要求10所述的方法,还包括确定第一分段和第二分段的分段状态,

其中调度第一命令和第二命令的处理包括如果所述第一分段可用并且所述第二分段不可用,则调度在第二作业描述符之前处理第一作业描述符。

18. 根据权利要求10所述的方法,还包括接收包括第三作业描述符的第三命令,所述第

三作业描述符指定服务于与所述第三作业描述符关联的数据的第三分段,以及
确定第一分段、第二分段和所述第三分段的分段状态,

其中调度第一命令和第二命令的处理包括如果所述第一分段和所述第二分段不可用并且所述第三分段可用,则调度在第一作业描述符和第二作业描述符之前处理所述第三作业描述符。

19. 根据权利要求10所述的方法,还包括接收包括第三作业描述符的第三命令,所述第三作业描述符指定服务于与所述第三作业描述符关联的数据的第三分段,以及

确定第一分段、第二分段和所述第三分段的分段状态,

其中调度第一命令和第二命令的处理包括如果所述第一分段和所述第三分段可用并且所述第二分段不可用,则调度在第一作业描述符之后、但是在第二作业描述符之前处理所述第三作业描述符。

20. 根据权利要求9所述的方法,还包括确定第一分段和第二分段的分段状态以及第一通道和第二通道的通道状态,

其中调度第一命令和第二命令的处理基于所述分段状态和所述通道状态二者被执行。

21. 一种用于存储器管理的系统,包括:

主机,其发出多个作业描述符,每个作业描述符指定目标分段和目标通道;

缓冲管理器,其控制多个分段,每个分段与所述多个作业描述符中的不同作业描述符关联以用于暂时存储与所述不同作业描述符关联的数据,所述数据包括将向多个闪存设备中的闪存设备写入的数据或者从所述闪存设备读取的数据;

所述多个闪存设备,每个闪存设备操作地耦合到用于在所述主机与相应闪存设备之间传达与所述不同作业描述符关联的所述数据的多个通道中的不同通道;以及

定序器,其:

以第一顺序接收所述多个作业描述符;

针对以所述第一顺序接收的所述多个作业描述符中的每一个,基于来自所述缓冲管理器的分段状态的更新和来自所述多个通道的通道状态的更新来确定所述目标分段和所述目标通道的可用性;和

基于相应的所述目标分段和所述目标通道的可用性,以第二顺序来调度作业描述符相对于其他作业描述符的执行,所述第二顺序不同于所述第一顺序。

描述符调度器

[0001] 有关申请的交叉引用

[0002] 本申请要求对通过完全引用将其公开内容结合于此、于2010年12月20日提交、名称为“DESCRIPTOR SCHEDULER”的第61/425,098号美国临时申请的优先权。本申请也与2009年9月29日提交、名称为“FLEXIBLE SEQUENCE DESIGN ARCHITECTURE FOR SOLID STATE MEMORY CONTROLLER”的第12/569,089号美国申请有关并且要求对2009年1月8日提交的第61/143,386号美国临时申请的优先权。

技术领域

[0003] 本申请的主题内容总体上涉及存储器管理。

背景技术

[0004] 许多电子设备包括用于控制底层设备的操作并且用于提供增强的功能和操作灵活性的嵌入式系统。通常,包括非易失性存储器作为嵌入式系统的一部分以用于存储用于操作嵌入式系统的操作系统程序代码、发出的命令和关键数据。近来,嵌入式系统已经开始使用固态存储器作为非易失性存储器(例如,电可擦除只读存储器(EEPROM))。固态存储器可以用来加速应用以增加数据访问时间并且可以被有利地编程或者再编程以用于提供信息的非易失性存储。

发明内容

[0005] 描述用于提供调度作业描述符的定序器的方法、系统和计算机程序产品。定序器可以基于作业描述符的相应分段和通道的可用性管理调度作业描述符以用于执行。例如,定序器可以检查分段的状态,并且标识在忙碌或者满状态中的一个或者多个分段或者在非忙碌或者空状态中的一个或者多个分段。基于状态检查,定序器可以乱序执行作业描述符并且特别地向关联分段可用的作业描述符给予超过关联分段在忙碌或者满状态中的作业描述符的优先级。在这样做时,可以更快地处理未决作业描述符并且可以避免不必要延时。

[0006] 在一些实现方式中,提供一种设备,该设备包括:缓冲管理器,具有多个分段,每个分段存储与多个作业描述符中的不同作业描述符关联的数据;控制器,控制多个闪存设备,每个闪存设备操作地耦合到对应通道,控制器被配置用于控制通过对应通道向或者从多个闪存设备传达与相应作业描述符关联的数据;以及定序器,基于对应分段或者通道的可用性调度多个作业描述符中的每个作业描述符的执行。

[0007] 在一些实现方式中,提供一种方法,该方法包括:接收用于访问存储器的第一命令,第一命令包括服务于与第一命令关联的数据的第一分段或者用于通过第一通道从第一闪存设备取回数据或者向第一闪存设备写入数据的第一指令中的至少一项;接收第二命令,第二命令包括服务于与第二命令关联的数据的第二分段或者用于通过第二通道从第二闪存设备取回数据或者向第二闪存设备写入数据的第二指令中的至少一项;确定第一分段和第二分段的分段状态或者第一通道和第二通道的通道状态;以及基于分段状态或者通道

状态调度第一命令和第二命令的处理。

[0008] 在一些实现方式中,提供一种系统,该系统包括:主机,发出多个作业描述符,每个作业描述符指定目标分段或者目标通道中的至少一项;缓冲管理器,控制包括目标分段的多个分段,每个分段与不同作业描述符关联以用于暂时存储与不同作业描述符关联的数据,数据包括将向闪存设备写入的数据或者从闪存设备读取的数据;多个闪存设备,每个闪存设备操作地耦合到用于在主机与相应闪存设备之间传达与不同作业描述符关联的数据的不同通道;以及定序器,基于目标分段或者目标通道的可用性调度作业描述符相对于其他作业描述符的执行。

[0009] 可以实现以下优点中的一个或者多个优点。定序器可以检查分段的状态,并且标识在忙碌或者满状态中的一个或者多个分段或者在非忙碌或者空状态中的一个或者多个分段。基于状态检查,定序器可以乱序执行作业描述符,并且特别地向关联分段可用的作业描述符给予超过关联分段在忙碌或者满状态中的作业描述符的优先级。在这样做时,可以更快地处理未决作业描述符并且可以避免延时。

[0010] 在附图和以下描述中阐述本发明的一个或者多个实施例的细节。将从描述和附图中以及从权利要求中清楚本发明的其他特征、目的和优点。

附图说明

[0011] 图1示出闪存设备的一个实施例的框图。

[0012] 图2A示出包括一个或者多个图1中所示闪存设备的固态驱动系统的一个实施例。

[0013] 图2B示出如下框图,该框图图示实施调度方案的定序器,该调度方案调度作业描述符的执行。

[0014] 图3示出具有多个未决命令的内部命令队列的一个实施例,每个未决命令包含T位信息。

[0015] 图4是示出用于调度作业描述符的过程的示例的流程图示例。

[0016] 各种附图中的相似标号指示相似单元。

具体实施方式

[0017] 描述用于提供调度作业描述符的定序器的系统、方法和计算机程序产品。传统定序器按照由固件指定的或者从固件接收的顺序执行作业描述符。每个作业描述符然后按照该顺序由关联分段服务。例如,更早发出的作业描述符“DES#1”由分段“0”服务;更晚发出的作业描述符“DES#2”在第一作业描述符“DES#1”已经由分段“0”服务之后由分段“1”服务;并且最后发出的作业描述符“DES#3”在更晚发出的作业描述符“DES#2”已经由分段“1”服务之后由分段“2”服务。

[0018] 在指明作业描述符以用于由它的关联分段处理时,如果该分段可用或者为“空”,则常规定序器在该分段中存储作业描述符。在已经处理分段中的数据时该分段可用或者为“空”。在分段仍然正处理旧数据或者来自先前周期的数据时该分段不可用或者为“满”。在分段不可用时,在常规定序器处保持它的关联作业描述符。由于按照具体顺序处理作业描述符,所以也在常规定序器处保持更晚发出的作业描述符,直至已经执行更早发出的作业描述符。

[0019] 例如,在更早发出的作业描述符“DES#1”由于它的分配分段“0”的忙碌状态而处于保持之时,即使分配的用于存储更晚发出的作业描述符“DES#2”的分段(例如,分段“1”)可用,更晚发出的作业描述符“DES#2”仍然也被保持于常规定序器处并且等待处理。在已经处理分段“0”中的所有数据并且分段“0”再次变成可用时,向分段“0”转发更早发出的作业描述符“DES#1”以用于存储。然后在已经执行第一作业描述符“DES#1”并且可以释放第一作业描述符“DES#1”之后发送并且在分段“1”中存储更晚发出的作业描述符“DES#2”。

[0020] 由于缺乏调度方案,所以常规定序器可能遭受延时问题,因为仅可以在完成先前作业描述符(例如,作业描述符“DES#1”)时存储和执行更晚发出的作业描述符(例如,作业描述符“DES#2”)。在这一“等待”时段期间,更晚发出的作业描述符即使它们的相应分段可用于存储却仍然“陷入”流水线中。在用于传输与更早发出的作业描述符关联的数据的存储器通道不可用而与更晚发出的作业描述符关联的其他存储器通道可用的情况下也存在前述性能问题。直至已经处理忙碌通道中的更早发出的作业描述符,更晚发出的作业描述符才可以访问可用通道。

[0021] 为了克服在常规定序器中指出的各种性能问题,如这里描述的定序器可以被配置用于基于作业描述符的相应分段和通道的可用性来管理调度作业描述符以用于执行。例如,定序器可以检查分段的状态,并且标识在忙碌或者满状态中的一个或者多个分段或者在非忙碌或者空状态中的一个或者多个分段。基于状态检查,定序器可以乱序执行作业描述符,并且特别地向关联分段可用的作业描述符给予超过关联分段在忙碌或者满状态中的作业描述符的优先级。例如,使用上文给出的示例,如果更早发出的作业描述符“DES#1”当前未决并且等待访问分段“0”,则定序器可以调度更晚发出的作业描述符“DES#2”以用于存储于分段“1”中并且先执行。在这样做时,可以更快地处理未决作业描述符,并且可以避免延时。

[0022] 类似地,定序器可以检查存储器通道中的每个存储器通道及其相应可用性状态,并且基于这些通道状态调度未决作业描述符。例如,如果向作业描述符“DES#1”分配的通道CH“0”忙碌、但是向作业描述符“DES#2”分配的通道CH“1”空闲或者可用,则定序器可以对作业描述符区分优先顺序并且调度在执行作业描述符“DES#1”之前先经由通道CH“1”处理作业描述符“DES#2”。

[0023] 通过基于分段或者通道的可用性乱序调度作业描述符并且对作业描述符区分优先顺序,定序器允许更快速的闪存操作(比如存储器读取、写入和擦除操作)并且增强对闪存中的数据进行管理的固态控制器的总性能。

[0024] 闪存概述

[0025] 在与相机、视频游戏、计算机和其他电子设备关联的跳跃驱动、内存卡和其他非易失性存储器应用中使用闪存设备,比如快闪电可擦除可编程只读存储器(EEPROM)。图1示出闪存设备100的框图。

[0026] 如图1中所示,可以按位组织闪存设备100。例如,存储器设备100可以是深度包括8位的阵列这一形式108。也可以按字节组织闪存设备100。例如,闪存设备100可以包括含2K字节的部分104和包含64字节的部分106。还可以将闪存设备100组织成页面。例如,闪存设备100可以包括512K页面110。可以将单个页面112组织成可以包括部分104和部分106作为第一部分114和第二部分116的两个部分。第二部分116可以对应于八位宽数据输入/输出

(I/O) 路径 (例如, I/O[0]-I/O[7])。更进一步, 可以按块布置闪存设备100。例如, 闪存设备100可以包括等于64个页面的块102。块可以在尺寸上被设定用于存储一个扇区的用户数据和开销数据, 该开销数据包括信息, 比如而不仅限于与用户数据关联的纠错码 (ECC)、块的使用历史、缺陷和存储器单元的其他物理信息。可以使用前述位、字节、页面和块来形成闪存设备100。

[0027] 在一些实现方式中, 闪存设备可以是“NAND”型闪存。NAND闪存一般具有比例如NOR型闪存更快的擦除和编程时间、更高的密度、更低的每位成本和更多的耐用性。NAND闪存可以利用多于两个 (比如16或者32个) 存储器单元的串联串, 这些存储器单元与一个或者多个选择晶体管一起连接于个别位线与参考电势之间以形成单元列。字线可以跨越这样的列内的单元延伸。可以在编程期间通过使串中的剩余单元接通以使得流过串的电势可以依赖于列内的个别单元中存储的电荷电平来读取和验证寻址的单元。

[0028] NAND闪存I/O接口可以用于与NAND存储器通信, 并且可以允许对NAND闪存中存储的数据的依次访问。NAND闪存I/O接口可以包括各自与具体功能对应的多个管脚。尽管仅已经描述了NAND闪存, 但是应当注意也可以使用其他存储器类型, 比如而不仅限于NOR闪存EEPROM、AND闪存EEPROM、DiNOR闪存EEPROM、串行闪存EEPROM、DRAM、SRAM、ROM、EPROM、FRAM、MRAM和PCRAM。

[0029] 固态驱动系统

[0030] 图2A示出包括一个或者多个闪存设备206 (比如图1中所示的闪存设备) 的固态驱动系统200的一个实施例。如图2A中所示, 系统200包括通过主机接口210与固态驱动204通信的主机202。主机接口210在一些实现方式中可以包括串行高级技术附件 (SATA) 接口或者并行高级技术附件 (PATA) 接口。SATA接口或者PATA接口可以分别用来将串行或者并行数据转换成并行或者串行数据。例如, 如果主机接口210包括SATA接口, 则SATA接口可以通过总线203 (例如, SATA总线) 接收从主机202传送的串行数据, 并且将接收的串行数据转换成并行数据。在其他实现方式中, 主机接口210可以包括混合接口。在这些实现方式中, 混合接口可以例如与串行接口结合使用。

[0031] 主机接口210在一些实现方式中可以包括其中可以暂时存储来自主机202的操作命令和地址的一个或者多个寄存器。主机接口210可以响应于寄存器中的存储的信息向固态控制器208传达写入或者读取操作。

[0032] 在一些实现方式中, 如下文将更具体讨论的那样, 固态驱动204可以支持一个或者多个通道 (例如, 八个通道), 并且每个通道可以被配置用于接收一个或者多个控制信号 (例如, 四个芯片使能信号) 以容纳内部命令队列217中未决的多于一个命令。

[0033] 闪存设备

[0034] 每个闪存设备206在一些实现方式中可以包括非易失性存储器 (例如, 单电平闪存或者多电平闪存)。在一些实现方式中, 非易失性存储器可以包括NAND型闪存模块。NAND型闪存模块可以包括命令/地址/数据复用接口, 从而使得可以通过对应输入/输出管脚提供命令、数据和地址。

[0035] 在一些实现方式中, 每个闪存设备可以连接到通道226。每个通道可以例如支持一个或者多个输入和输出线、芯片选择信号线、芯片使能信号线等。通道也可以支持其他信号线, 比如写入使能、读取使能、读取/忙碌输出和重置信号线。在一些实现方式中, 闪存设备

206a-206d可以共享共同通道。在其他实现方式中,为了增加并行性的程度,每个闪存设备可以让它自己的通道连接到固态驱动204。例如,闪存设备206a可以使用通道226a来连接到固态驱动204;闪存设备206b可以使用通道226b来连接到固态驱动204;闪存设备206c可以使用通道226c来连接到固态驱动204;并且,闪存设备206d可以使用通道226d来连接到固态驱动204。

[0036] 在一些实现方式中,闪存设备206a-206d可以可拆卸。在一些实现方式中,闪存设备206a-206d可以使用标准连接器来连接到固态驱动204。标准连接器的示例可以包括而不仅限于SATA、USB(通用串行总线)、SCSI(小型计算机系统接口)、PCMCIA(个人计算机存储卡国际协会)和IEEE-1394(火线)。

[0037] 在一些实现方式中,每个闪存设备206可以包括在存储体(bank)中布置的一个或者多个固态存储元件。可以将固态存储元件分割为页面。在一些实现方式中,固态存储元件可以具有2000字节(即一个页面)的容量。固态存储元件在一些实现方式中可以包括用于提供4000-4096字节(即4kB)的总容量的两个寄存器。

[0038] 在一些实现方式中,每个闪存设备206也可以包括各自使用芯片使能信号或者芯片选择信号来选择的一个或者多个存储体。芯片使能或者芯片选择信号可以响应于主机命令选择一个或者多个固态存储元件。

[0039] 在一些实现方式中,每个固态存储元件可以包括一个或者多个单电平单元(“SIC”)设备。在一些实现方式中,每个固态存储元件可以包括一个或者多个多电平单元(“MLC”)设备。可以使用芯片使能信号或者芯片选择信号来选择SLC或者MLC设备,控制器208可以使用从主机202接收的控制和地址信息的组合来生成该芯片使能信号或者芯片选择信号。

[0040] 在使用多个存储体时,在一些实现方式中,固态驱动208可以在相同时间访问相同闪存设备中的多于一个存储体。在一些实现方式中,固态驱动208可以在相同时间访问不同闪存设备中的不同存储体。用于访问多于一个存储体的能力允许固态设备204完全利用可用资源和通道226a-226d以增加固态驱动204的总性能。另外,在闪存设备206a-206d共享相同存储器输入/输出线和控制信号(例如,芯片使能信号)时,可以减少控制器208的管脚数目以进一步最小化用于制造固态驱动204的成本。

[0041] 固态控制器

[0042] 作为闪存控制器来工作的固态控制器208接收一个或者多个服务请求或者命令(例如,读取和编程请求)。控制器208可以被配置用于处置用于访问闪存设备206a-206d的任何命令、状态或者控制请求。例如,控制器208可以被配置用于管理和控制在闪存设备206a-206d中存储和取回数据。

[0043] 在一些实现方式中,控制器208可以是在微处理器(未示出)的控制之下的微型计算机系统的一部分。控制器208可以控制在主机202与固态驱动204之间的命令和数据流。在一些实现方式中,控制器208可以包括只读存储器(ROM)、随机存取存储器(RAM)和其他内部电路。控制器208在一些实现方式中可以被配置用于支持与闪存设备206关联的各种功能,比如而不仅限于诊断闪存设备206、向闪存设备206发送命令(例如,激活、读取、编程、擦除、预充电和刷新命令)和从闪存设备206接收状态。控制器208可以形成于与闪存设备206a-206d不同的芯片上(例如,形成于与固态驱动204相同的芯片上)或者相同芯片上。

[0044] 如图2A中所示,控制器208可以包括检错码(error checking code,ECC) 212、接口逻辑214、定序器216和先入先出(FIFO) 218。在一些实现方式中,固态驱动204可以包括中央处理器单元(CPU) 223,该CPU具有可以用来控制控制器208的嵌入式固件224。CPU223可以包括微处理器、信号处理器(例如,数字信号处理器)或者微控制器。在一些实现方式中,具有嵌入式固件224的CPU223可以驻留于固态驱动204以外。

[0045] 在一些实现方式中,固态驱动204也可以连接到用于控制一个或者多个硬盘222的硬盘驱动(HDD) 控制器220。HDD控制器220可以是用于海量存储系统的控制器、光盘驱动控制器或者用于管理数据存储装置的其他相似控制器。在一些实现方式中,HDD控制器220可以与控制器208结合用来管理和控制闪存设备和硬盘222以最大化固态系统200的存储容量。

[0046] 接口逻辑模块

[0047] 接口逻辑214可以向闪存设备206a-206d发送来自控制器208的命令、地址和控制信息。接口逻辑214也可以向控制器208转送来自闪存设备206a-206d的命令状态。

[0048] 在一些实现方式中,接口逻辑214可以被配置用于支持NAND型闪存和NOR型闪存。接口逻辑214可以通过通道226a-226d与闪存设备206a-206d通信。在使用不同类型的闪存设备(例如,NAND存储器和NOR存储器)时,通道226a-226d可以连接到相同或者不同类型的存储器。在一些实现方式中,在对不同通道执行读取或者编程操作时,接口逻辑214可以有选择地例如执行软件和硬件交织操作。在其他实现方式中,接口逻辑214也可以并发处置用于不同通道的读取和编程操作。

[0049] 在一些实现方式中,接口逻辑214可以包括而限于一个或者多个数据端口、信号检测器、编码器、解码器、交织器、解交织器、比较器等。接口逻辑214可以用作在控制器208与闪存设备206a-206d之间的用于时钟、数据掩码、选通、等级选择、存储器地址、存储器命令以及读取和编程数据的接口。当在闪存设备206a-206d中包括SLC或者MLC设备(例如,用于存储数据的块和页面缓冲器)时,接口逻辑214可以提供用于与SLC/MLC设备通信的更低级存储器接口。

[0050] FIFO

[0051] FIFO218可以使用先入先出方案来操作,并且可以被配置用于按照与接收的顺序相同的顺序输出数据(例如,输出先接收的数据)。在一些实现方式中,FIFO218可以包括各自具有比数据分段虚拟页面更多的容量的一个或者多个寄存器。FIFO218可以用来暂时存储在例如向定序器216和闪存设备206传送以用于执行之时尚未处理的数据和命令。在一些实现方式中,ECC212和/或接口逻辑214可以利用FIFO218以缓冲在主机202与闪存设备206之间的数据传送,以及为从闪存设备206接收或者向闪存设备206传输的数据提供纠错。在一些实现方式中,FIFO218也可以包括内部命令队列(例如,从内部命令队列217分离),定序器216可以通过该内部命令队列存储用于命令或者控制闪存设备206的命令或者控制信号。相同或者不同内部命令队列也可以用来存储在嵌入式固件(例如,固件224)与闪存设备206之间的首部(header)和/或内部数据。

[0052] 在一些实现方式中,FIFO218可以生成用于例如通过定序器216的内部命令队列217(如下文将更具体讨论的那样)向控制器208传送数据和从控制器208接收数据的数据和控制信号。FIFO218可以例如基于来自主机、定序器204或者闪存设备206a-206d的命令确定

内部命令队列217中存储的数据的传送方向。

[0053] 在一些实现方式中,可以在FIFO218中存储由ECC212处理的数据。在一些实现方式中,ECC212可以释放在FIFO218中存储的每个数据块以用于代码纠正。具体而言,在ECC212标识数据中的错误时,可以向数据施加纠正。然后可以在完成对数据块的ECC过程之后基于从主机202接收的命令从FIFO218向由定序器216选择的闪存设备传送纠正的数据块。

[0054] 纠错码

[0055] 闪存设备206a-206d易受缺陷和故障的影响。多个因素可能引起这样的错误,例如包括存储器状态的阈值电平由于环境条件的逐渐移位、来自包括编程、擦除、读取操作等的正常操作的应力等。其他错误可以例如由闪存设备中的一个或者多个故障存储器位、向和闪存设备的错误传输等造成。因此,为了防止在操作期间的错误,在一些实现方式中,可以在控制器208中包括ECC212以保证固态驱动系统200的数据完整性。ECC212可以被配置用于检测和纠正例如由在读取、擦除或者编程时间的软错误和硬错误引起的数据故障。

[0056] 在一些实现方式中,ECC212可以用于在向闪存设备206发送之前检查和纠正与在定序器216的内部命令队列217中存储的数据和命令关联的错误。此外,ECC212可以用于检查和纠正与向闪存设备206传输和从闪存设备206接收的数据关联的错误。例如,在从闪存设备向控制器208传送数据(例如,状态命令)时,ECC212可以执行错误检测操作以保证传送的数据的完整性。如果在传送的数据之中检测到错误,则ECC212可以迅速请求来自控制器208的检错并且随后执行数据纠正。然后可以向定序器216发送纠正的数据以用于进一步处理。

[0057] 作为示例操作,读取请求可以被定序器216排队(如下文将更具体讨论的那样)并且向闪存设备206发送。然后可以从闪存设备206向ECC212返回数据。ECC212可以针对错误检查返回的数据。如果检测到错误、但是纠错被停用,则可以向定序器216发送错误通知以取代预计的数据。错误通知可以包括指示数据错误已经出现的错误报告。然后可以通过将出故障的位置的地址移向出故障映射来停用该位置。例程可以恢复并且等待下一请求。如果ECC212检测到错误并且纠正功能被启用,则可以纠正、验证并且向定序器216转发出故障的数据。

[0058] 在一些实现方式中,控制器204可以生成可以在编程操作期间向数据扇区追加的冗余位(例如,奇偶校验位)。例如,512字节数据扇区可以包括追加的16字节的ECC数据,从而产生528字节页面。在读取操作期间,在16字节的ECC数据中代表的冗余数据可以用来检测和纠正从闪存设备206读出的数据中的错误。

[0059] 在一些实现方式中,ECC212也可以被配置用于检测和纠正主机202与固态设备204之间流动的数据中的错误。ECC212可以在检查和纠正错误时实施包括循环冗余校验(CRC)以及奇偶校验和前向纠错(FEC)的纠错方案。

[0060] 定序器

[0061] 在一些实现方式中,定序器216可以用来代表固件224执行命令。定序器216可以例如控制向闪存设备206a-206d的一个或者多个数据扇区写入和从该一个或者多个数据扇区读取中的一项或者两项的访问过程。

[0062] 在一些实现方式中,定序器216可以在内部命令队列217中存储从固件224(或者主机202)接收的命令。内部命令队列217允许控制器208执行多块数据访问。在一些实现方式

中,闪存设备(例如,闪存设备206a-206d)包括内置的512字节页面寄存器以使得在执行向闪存设备的数据写入时,在向闪存单元写入之前先向页面寄存器写入数据。常规闪存控制器通过向闪存设备的页面寄存器一次传送一个单块(512字节)的数据来控制闪存访问周期。因此,一旦填充了512字节页面寄存器就不允许对闪存设备的其他访问。因而,使用单块数据访问方法的常规闪存控制器限制向和从闪存设备存储和取回数据的性能。

[0063] 为了增强数据并行性并且扩展控制器208的用于接收对任何给定的闪存设备的多于一个命令的能力,内部命令队列217可以被提供并且工作于存储未决命令,并且在一些实现方式中,内部命令队列217可以具有适合于存储各自代表一个页面的信息的三十二个命令的容量。主机202可以向控制器208同时发送各自包含唯一数据的多个命令,并且控制器208可以在内部命令队列217中存储这样的命令用于向闪存设备206a-206d并发或者依次传输。不同于一次传送单个数据块的常规单块数据传送控制器,利用定序器216和内部命令队列217的控制器208允许向闪存设备206a-206d发送多于一个命令,因此显著提高数据传输性能。

[0064] 作为示例,主机202可以发出读取命令并且向定序器216发送读取命令。在定序器216接收读取命令时,定序器216可以将命令放置于内部命令队列217中。定序器216然后可以标识命令被发送到的闪存设备和标识的闪存设备的其中数据将被读取的地址。在验证可以在该地址访问标识的闪存设备之后,定序器216可以通过使用芯片使能(或者芯片选择)信号选择标识的闪存设备并且向选择的闪存设备发送出地址和命令来向标识的闪存设备门控地址和命令。在从定序器216接收读取命令(或者指示读取命令的控制信号)时,标识的闪存设备可以处理读取命令并且随后向定序器216发送回串行数据或者命令状态信息。这一过程可以继续,直至寻址的数据的最后一位已经被标识的闪存设备传输并且被定序器216接收。

[0065] 在一些实现方式中,可以在FIFO218中先存储从标识的闪存设备接收的数据。如果由标识的闪存设备传输的数据包含坏位,则可以在向定序器216发送数据之前将一个或者多个坏位替换为一个或者多个纠正的位。备选地,可以向定序器216发送从标识的闪存设备接收的数据,此后向FIFO218转发数据以向主机202发送出。在数据在FIFO218中等待之时,可以对数据执行ECC处理。在已经执行命令或者命令集合之后,定序器216可以加载内部命令定序器217中的下一命令或者命令集合以用于处理。

[0066] 类似地,在主机202发出编程命令时,定序器216可以从固件224(或者从主机202)取读写入数据。固件224可以被嵌入于固态驱动204的用于控制控制器208的CPU223中。在接收数据时,定序器216可以请求ECC212执行ECC纠错过程。这允许在向标识的闪存设备中写入之前检测和纠正正在传输期间出现的错误。写入数据可以在执行ECC纠错过程之时存储于FIFO218中,并且随后在纠正之后向指定的闪存设备传递。

[0067] 在一些实现方式中,可以在向闪存设备206a-206d发送之前将数据(例如,该数据可以包括命令、地址和控制信息)格式化串行数据。流水线架构也可以用来在从接口逻辑214向标识的闪存设备门控数据时提供高效吞吐量。在一些实现方式中,可以在向标识的闪存设备发送之前先验证从接口逻辑214门控出的数据。在已经向标识的闪存设备中加载数据之后,闪存设备可以处理包括命令的数据。

[0068] 在一些实现方式中,定序器216可以在滚动基础上执行每个命令(例如,在接收每

个命令时执行它)。在其他实现方式中,定序器216可以确定内部命令队列217是否为满或者接近为满(例如,包括预定数目的命令)。在一些实现方式中,在内部命令队列217为满或者接近为满时,定序器216可以逐个或者并发执行命令(例如,一起执行当前存储于命令队列217中的所有未决命令)。在希望处理多个命令时,接口逻辑214可以被配置用于并行执行多个命令。可选地,序列216可以包括具有多个核的多核系统,其中每个核被配置用于处理一个命令以便允许一次执行多个命令。

[0069] 例如,假设内部命令序列217可以容纳三十二个命令,则定序器216可以监视内部命令序列217以确定是否存在存储于内部命令队列217中的三十二个未决命令。如果存在三十二个未决命令,则定序器216可以在相同时间执行所有三十二个命令,并且在满足了与命令关联的请求时向固件224报告每个执行的命令的状态。在一些实现方式中,定序器216可以向接口逻辑214发出一个或者多个命令或者控制信号,从而使得通道226a-226d可以并发开始数据传送过程。

[0070] 在从定序器216接收一个或者多个命令时,选择的闪存设备可以执行与命令关联的请求的操作(例如,读取或者编程操作)。在一些实现方式中,固件224无需在执行命令时轮询命令的状态。在这些实现方式中,闪存设备206a-206d可以通过接口逻辑214向定序器216转发命令状态信息,这可以触发中断例程。在触发中断例程时,可以向固件224发送中断(或者消息的其他形式)以向固件224通知命令状态信息。

[0071] 在一些实现方式中,定序器216可以在从对应闪存设备发送状态时实时向固件224报告命令的状态。在一些实现方式中,可以配置定序器216以使得定序器216仅在满足具体条件时报告命令状态。具体条件可以包括检错、纠错、编程延迟等。在一些实现方式中,具体条件可以依赖于由命令中的T位提供的信息。图3图示具有多个未决命令的示例内部命令队列300,每个未决命令包含T位信息。

[0072] 如图3中所示,命令302当前存储于内部命令队列300中待执行。可以例如从主机202接收命令302。每个命令可以包括与命令编号字段308、命令类型字段310、通道字段318、芯片使能字段312、T位字段314和存储器字段316关联的信息。

[0073] 命令编号字段308可以用来指示当前存储于内部命令队列300中的命令的编号。命令类型字段310可以指示与每个命令关联的命令类型。例如,命令302具有命令类型“读取”,该命令类型指示命令302是读取命令。作为另一示例,命令304具有命令类型“编程”,该命令类型指示命令304是编程命令。

[0074] 通道字段318可以定义将用来发送对应命令的通道(例如,通道226a-226d之一)。作为示例,将通过通道“1”向闪存设备阵列中的第3闪存设备(如存储器字段316中所示)发送命令302。作为另一示例,将通过通道“2”向第6闪存设备发送命令304。作为又一示例,将通过通道“3”向第7闪存设备发送命令306。

[0075] 在一些实现方式中,固件224可以定义用于每个字段的T位字段314的T位信息。在一些实现方式中,T位信息用作关于是否将触发用于向固件224通知命令状态信息的中断例程的指示符。例如,如图3中所示,固件224可以指明命令306的T位字段314为“1”,这指示定序器216将在与命令306关联的状态从第7闪存设备可用时向固件224发送中断。作为另一示例,固件224可以指明命令302的T位字段314为“0”,这指示定序器216无需在与命令302关联的状态从第3闪存设备可用时向固件224发送中断。类似地,固件224可以指明命令304的T位

字段314为“0”，这指示定序器216无需在与命令304关联的状态从第6闪存设备可用时向固件224发送中断。T位字段314一般可以向固件224提供用于定义将何时和多么经常调用固件224的灵活性，从而使得可以将不必要计算资源的使用和与接收状态信息关联的延时减少至最小。

[0076] 在一些实现方式中，固件224可以将T位信息与从主机202接收的命令一起向定序器216提供。在其他实现方式中，固件224可以向定序器216单独提供T位信息和从主机202接收的命令。

[0077] 定序器216可以有助于未决命令的全部处理，包括但不限于存储从固件224接收的一个或者多个未决命令、向对应闪存设备传送未决命令、协调命令在由闪存设备处理命令时的状态轮询以及向固件224报告命令的状态。

[0078] 在一些实现方式中，固态驱动204 (和/或主机202) 可以装配于片上系统 (SOC) 上。在这些实现方式中，可以例如使用半导体工艺来制作SOC。SOC可以包括从固态驱动204中的嵌入式过程系统 (例如，嵌入式CPU) 分离的嵌入式过程系统。SOC也可以包括用于处理程序代码和数据的SRAM、系统逻辑、高速缓存存储器和高速缓存控制器。可以在闪存设备206中存储并且例如通过接口向SOC传达与嵌入式过程系统关联的程序代码和数据。这一接口可以由转译器 (未示出) 用于转译在接口与SOC的内部总线结构之间流动的信息。例如，在读取操作期间控制信号可以从SOC流向闪存设备206a-206d，而指令和数据可以从闪存设备206a-206d流向SOC。当然，指令和数据也可以例如在擦除操作期间流向闪存设备206a-206d。

[0079] 作业描述符调度

[0080] 图2B是图示作业描述符调度过程的框图。在一些实现方式中，固件224可以采取作业描述符的形式向控制器208生成一个或者多个命令。作为示例，固件224可以生成各自代表处理操作 (比如读取、写入或者擦除操作) 的作业描述符230 (DES#1) 和232 (DES#2)。在固件224向闪存设备206分配作业时，发出具体作业描述符，该作业描述符包含信息，如例如目标分段、目标通道 (例如，通道226a-226d中的任何通道)、目标闪存设备 (例如，闪存设备206a-206d中的任何闪存设备)、数据的类型和数量、操作类型 (例如，读取、写入、擦除或者其组合) 以及ECC功率校正信息。

[0081] 定序器216可以接收从固件224发出的作业描述符，并且基于一个或者多个调度条件调度作业描述符中的每个作业描述符以用于存储和执行 (如下文将更具体讨论的那样)。调度发出的作业描述符的存储和执行通过允许在调度分配的通道当前不可用或者在“忙碌状态”中的作业描述符的处理之前先调度分配的分段或者通道“可用”、“自由”或者不“忙碌”的作业描述符以用于存储和处理来增强数据并行性并且最大化固态驱动204的数据处理速度 (如下文将更具体讨论的那样)。忙碌分段在已经执行指定使用该段的作业描述符并且不再需要该段 (例如，已经从闪存设备226取回或者向闪存设备226写入固件224请求的数据) 时变成可用。

[0082] 在定序器216对作业描述符区分优先顺序并且调度其执行时，可以执行作业描述符而未引起由于当在处理流水线中“俘获”更早发出的作业描述符并且更晚发出的作业描述符直至已经处理更早发出的作业描述符才可以继续时出现的“陷入”条件所造成的任何延迟。更早发出的作业描述符可能在它们的分配的通道或者分段暂时不可用时在处理流水

线中被“俘获”。例如,在向分段“0”分配作业描述符以用于暂时存储(例如,在错误检查和纠正之前)、但是段“0”不可用时,控制器208直至已经执行更早发出的作业描述符才可以继续执行更晚发出的作业描述符。

[0083] 为了有效管理所有可用带宽,定序器216可以按照具体调度顺序调度作业描述符以用于存储和执行。在一些实现方式中,定序器216可以检查在作业描述符中指定的目标分段、目标通道和/或目标闪存设备的可用性,并且一旦目标分段、目标通道和/或目标闪存设备变成可用就调度作业描述符以用于暂时存储和执行。不同于需要按照主机发出的顺序执行作业描述符(这在更早发出的作业描述符等待执行时引起“陷入”条件)的常规方案,定序器216可以实施其中在处理更晚发出的作业描述符之后处理更早发出的作业描述符的调度方案。如果多个作业描述符的相应目标设备、分段或者通道可用,则定序器216也可以立刻调度它们的执行。通过恰当排序和调度作业描述符的执行,可以增强数据并行性并且可以减少或者消除延时问题。

[0084] 在一些实现方式中,固件224无需保持对每个作业描述符的状态的跟踪。取而代之,固件224可以通过设置作业描述符的特定位来将若干作业描述符放入单个跟踪中。例如,如果固件224想要将十六(16)个描述符放入单个跟踪中,则可以设置最后描述符的T位(例如,T位314)。T位允许将多个作业描述符一起分组并且在完成时向固件224报告,从而使得固件无需花费太多时间来服务于每个作业描述符。定序器216然后可以在完成跟踪时(例如,当在跟踪中的作业描述符中定义的所有作业完成时)向固件224生成中断信号。固件224可以每个轨道边界地对作业描述符编程。在一些实现方式中,无论跟踪如何,只要目标闪存设备可用,就可以并行发出每个作业描述符。

[0085] 为了有效执行作业描述符,定序器216可以与缓冲管理器213通信,并且在缓冲器219中暂时存储每个描述符,直至ECC212已经标识和纠正与作业描述符关联的所有错误(例如,在执行作业描述符或者响应于作业描述符从目标闪存设备读取数据或者向目标闪存设备写入数据之时出现的错误)。缓冲器219可以是DDR SDRAM(双倍数据速率同步动态随机存取存储器)缓冲器、SDR SDRAM(单倍数据速率同步动态随机存取存储器)、RDRAM(直接Rambus随机存取存储器)或者是具有固定或者可变分区大小的其他相似类型的存储器。一般而言,缓冲器219可以是作为用于存储与作业描述符关联的数据或者作业描述符本身的暂时存储设施来工作的本地存储器。

[0086] 可以将缓冲器219划分成相等固定大小的分区。例如,可以将缓冲器219划分成各自具有六十四(64)字节的多个分区。缓冲器219中的分区数目可以等于支持的流和闪存设备的数目,并且分区可以是相同大小。例如,可以将缓冲器219分割成各自与四个闪存设备206a-206d之一对应的四个相等大小的分区。

[0087] 也可以将分区分组多个虚拟或者逻辑组。例如,可以将分区划分成两组,其中一组将用于数据写入并且另一组将用于数据读取。尽管分区是相等大小(例如,每组中四个分区),但是分区也可以是不同大小(例如,一组中四个分区而另一组中十二个分区)。分区也无需都在连续地址。

[0088] 虚拟分区是数据记录的分组,其中基于预定义的规则或者标准形成分组。每个虚拟分区可以在任何瞬间与缓冲器219中的分段关联并且被映射到该分段,而且可以在创建附加分段时被重新映射到不同分段。分段是缓冲器219的已经被保留用于存储数据块(或者

其部分)的物理区域。分段可以是用于一个存储高速缓存访问的数据大小单位(访问单位)。分段的大小可以从高速缓存分区区域到高速缓存分区区域变化。可以在虚拟分区被映射到的分段中存储虚拟分区内的所有数据记录。分段可以存储属于被映射到该分段的一个或者多个虚拟分区的数据记录。可以在分段中存储的其他数据包括而限于将主机202的地址映射到目标闪存设备206的地址的映射表以及支持固件操作的其他数据。

[0089] 在服务于作业描述符时,可以向缓冲器219中的分段发送关联数据或者其块。通常,可以向缓冲器219中的相同逻辑分段发送在一个操作中读取的部分。逻辑分段可以是连续地址空间并且指示从第一分区的第一存储器位置到最后分区的最后存储器位置的连续存储器位置的单个集合。例如,如果在缓冲器219中存在“X”个分区并且每个分区是“Y”个存储器位置宽,则第一分段对应于前X*Y个存储器位置并且第二分段对应于接下来的X*Y个存储器位置。缓冲管理器213和/或控制器208也可以物理地分散、链接和管理逻辑分段。

[0090] 缓冲器219的总大小可以是一(1)千兆字节(Gb)或者两(2)千兆字节(Gb),并且可以将缓冲器219划分成256K的分区。一般而言,缓冲器大小依赖于应用和设计,并且也设想其他缓冲器大小。256K的分区中的每个分区也可以在统计上被分配到闪存设备206并且对应于不同分段(例如,分段“1”指代第一分区),从而使得每个流和闪存设备与一个分区和分段关联。

[0091] 为了执行读取操作,固件224可以发出作业描述符,该作业描述符请求闪存设备206的读取访问。在接收作业描述符时,定序器216可以从作业描述符的内容标识作业描述符被分配到的分段。例如,定序器216可以从作业描述符230的内容标识已经分配分段“0”以用于存储从目标闪存设备取回或者访问的数据。作为另一示例,定序器216可以标识分段“1”为已经被分配用于存储将响应于作业描述符232向目标闪存设备写入的数据。如上文讨论的那样,固件224可以分配用于每个作业描述符的分段。可以在每个作业描述符的内容中包括分配的通道。每个作业描述符的内容也可以包括逻辑块地址(例如,定义如何按照主机(比如CPU223)的使用顺序存储数据)、缓冲器地址(例如,定义目标分段)以及例如标识目标通道和目标闪存设备的物理块地址。

[0092] 图4是示出用于调度作业描述符的过程400的示例的流程图。参照图4,在402处,接收用于访问存储器的第一命令,其中将在缓冲器的第一分段中存储与第一命令关联的数据。例如,在调度作业描述符以用于执行时,定序器216可以确定是否存在更早发出的作业描述符,并且如果存在,则确定更早发出的作业描述符是否仍然未决。更早发出的作业描述符例如在服务于更早发出的作业描述符的分段为“满”、“不可用”或者“忙碌”时未决。“满”、“不可用”或者“忙碌”分段意味着分段包含数据并且不含可用空间。反言之,分段在分段不含数据时为“空”或者“可用”。

[0093] 例如,定序器216可以确定作业描述符230是第一个发出的作业描述符,并且因此没有在发出作业描述符230之前的更早发出的作业描述符未决。定序器216也可以确定作业描述符230指定缓冲器219的分段“0”以用于暂时存储与作业描述符230关联的数据(例如,将向目标闪存设备写入或者从目标存储设备读取的数据)。

[0094] 在404处,接收第二命令,其中将在缓冲器的第二分段中存储与第二命令关联的数据。作为示例,定序器216可以接收作业描述符232。在接收作业描述符232时,定序器216可以确定作业描述符232是第二个发出的作业描述符,并且固件224已经在发出作业描述符

232之前发出作业描述符230。

[0095] 在406处,可以确定第一分段和第二分段的状态。例如,在确定第一分段和第二分段的状态时,定序器216可以向缓冲管理器213请求第一分段和第二分段的更新。例如,缓冲管理器213可以用分段“0”忙碌(或者为满)并且分段“1”可用(或者为空)这样的状态更新定序器216。作为另一示例,缓冲管理器213可以用两个分段“0”和“1”忙碌或者两个分段“0”和“1”可用这样的状态更新定序器216。

[0096] 在408处,可以基于状态调度第一命令和第二命令的处理。例如,为了调度作业描述符230和作业描述符232的执行,定序器216可以先考虑分段“0”(服务于作业描述符230)是否忙碌或者可用。定序器216也可以考虑分段“1”(服务于作业描述符232)是否忙碌或者可用。作为示例,在分段“0”和分段“1”均可用时,定序器216可以按照任何顺序或者按照由固件224分配的顺序执行作业描述符230和作业描述符232二者。作为另一示例,在分段“0”不可用、但是分段“1”可用时,定序器216可以调度其数据将先被存储和执行的作业描述符232。一旦已经暂时存储与作业描述符232关联的数据并且已经随后执行作业描述符232,则可以调度暂时存储与作业描述符230关联的数据并且接下来执行作业描述符230。作为又一示例,在分段“0”可用、但是分段“1”不可用时,定序器216可以调度先执行作业描述符230、继而为作业描述符232。

[0097] 在一些实现方式中,如果分段“0”可用,则定序器216无需考虑分段“1”是否忙碌或者可用,因为作业描述符230是更早发出的描述符,除非更早发出的描述符被分配到的分段不可用,否则可以向更早发出的描述符默认给予超过更晚发出的作业描述符的优先级。

[0098] 在存在附加作业描述符(例如,在作业描述符230和作业描述符232之后发出的作业描述符)时,可以重复前述调度过程以确定这些新作业描述符被分配到的分段中的每个分段是否也可用。然后可以向关联分段可用的作业描述符给予优先级,即使这些作业描述符是更晚发出的描述符。

[0099] 作为示例,在存在分配有个别化的分段(例如,分段“2”)的更晚发出的作业描述符234(DES#3)时,定序器216可以通过进一步检查分段“2”是否可用在作业描述符230、作业描述符232和作业描述符234之间区分优先顺序。如果分段“2”可用(但是分段“0”和分段“1”不可用),则定序器216可以先调度与作业描述符234关联的数据的暂时存储和作业描述符234的执行。如在作业描述符230与作业描述符232之间可以应用如先前讨论的调度过程以确定它们各自的优先级和调度(例如,如果分段“0”不可用并且分段“1”可用则先调度作业描述符232或者如果分段“0”可用并且分段“1”不可用则先调度作业描述符230)。

[0100] 缓冲管理器213可以存储从或者向目标闪存设备读取/写入的数据。例如,缓冲管理器213可以在缓冲器219中存储控制器209取回的数据以用于暂时存储并且随后向CPU223传输。作为另一示例,缓冲器213可以在缓冲器219中暂时存储将向目标闪存设备写入的(例如,如由CPU223或者另一主机接收的)数据,直至控制器208已经向目标闪存设备写入所有数据。可选地,ECC212可以检查取回的数据或者将由控制器208写入的数据以确定取回的数据或者待写入的数据是否无错误,并且如果取回的数据或者待写入的数据包含一个或者多个错误,则执行纠错以去除这些错误。

[0101] ECC212可以与定序器216和CPU223连续通信以向定序器216和CPU223更新在与作业描述符关联的数据中存在的任何错误。例如,定序器216可以暂时存储取回的数据,直至

取回的数据已经被ECC212审查并且向缓冲器219的对应分段发送以用于暂时存储。作为另一示例,定序器可以暂时存储控制器208将向目标闪存设备写入的数据,直至ECC212已经审查待写入的数据。缓冲器219中的每个分段可以对应于待服务于的命令并且存储已检错的数据。如上文讨论的那样,CPU223和固件224可以在每个命令中指定取回的数据或者待写入的数据将被暂时存储到的特定分段。例如,由CPU223发出的读取命令可以指定在分段“0”中存储响应于作业描述符230访问的数据。

[0102] 在一些实现方式中,在处理作业描述符时,定序器216可以向作业描述符添加描述符ID以生成新描述符。缓冲管理器213可以在它的对应分段中存储与新描述符关联的数据,直至已经从目标闪存设备取回并且向缓冲器219或者ECC212传输所有相关数据。

[0103] 在ECC212检查取回的数据时,ECC212可以在从控制器208接收取回的数据的每个块时逐个处理取回的数据。在一些实现方式中,作业描述符可以包括缓冲器地址字段,并且ECC212可以使用作业描述符中的缓冲器地址字段以向缓冲器219中的正确字段发送取回的数据的每个块。作业描述符也可以包括ID字段,并且ECC212可以使用作业描述符中的“ID”字段以向定序器216传输信号,该信号指示已经完成针对取回的数据的对应块的检错。

[0104] 定序器216可以收集如从ECC212传输的“ID”字段中的数据。在一些实现方式中,定序器216可以被配置用于存储列表,该列表记录在用于每个作业描述符的ID字段中标识的数据。以这一方式,定序器216可以跟踪控制器208已经响应于发出的作业描述符中的每个作业描述符取回的请求的数据的每个部分或者块。在定序器216确定ECC212已经传输所有数据部分或者块时,定序器216可以向缓冲管理器213传输用于向CPU223传输请求的数据的所有部分或者块的请求。

[0105] 尽管关于响应于读取命令取回(或者读取)数据来描述前述实现方式,但是相同过程也可以用于写入或者擦除数据。类似地,前述实现方式也适用于基于通道226的可用性调度作业描述符。

[0106] 例如,也可以向每个作业描述符分配通道,并且如果更早发出的作业描述符的分配的通道不可用,则定序器216可以先调度更晚发出的作业描述符以用于执行。定序器216可以向控制器208查询通道中的每个通道的更新或者特定通道的状态。

[0107] 在接收作业描述符中的每个作业描述符的相应通道状态时,定序器216可以通过确定作业描述符的优先级来调度作业描述符的执行。作为示例,定序器216可以接收作业描述符236(DES#4)、继而为另一作业描述符238(DES#5)。定序器216可以从相应作业描述符236和238的内容确定向通道CH“0”分配作业描述符236并且向通道CH“1”分配作业描述符238。

[0108] 在确定通道CH“1”和通道CH“0”的状态时,定序器216可以向控制器208请求通道226的更新。例如,控制器208可以用通道CH“0”忙碌(或者满)并且通道CH“1”可用(或者为空)这样的状态更新定序器216。作为另一示例,缓冲管理器213可以用两个分段“0”和“1”忙碌或者两个分段“0”和“1”可用这样的状态更新定序器216。

[0109] 如果通道被连接到的目标闪存设备通道空闲,则该通道可用。目标闪存设备例如在目标闪存设备未响应于任何作业描述符或者命令访问它的数据时空闲。例如,如果闪存设备206a空闲并且可用于访问(或者当在通道226a中没有命令/数据传送时),则通道226可用。在向多个闪存设备分配通道时,如果至少一个分配的闪存设备空闲,则通道可以被视为

空闲。作为另一示例,如果闪存设备206b忙于处置数据写入或者数据读取,则通道226b不可用。

[0110] 为了调度作业描述符236和作业描述符238的执行,定序器216可以考虑通道CH“0”(作业描述符236被分配到该通道)是否忙碌或者可用。定序器216也可以考虑通道CH“1”(作业描述符238被分配到该通道)是否忙碌或者空闲。例如,在通道CH“0”和通道CH“1”均可用时,定序器216可以按照任何顺序或者按照固件224发出的顺序执行作业描述符236和作业描述符238二者。作为另一示例,在通道CH“0”不可用、但是通道CH“1”可用时,定序器216可以调度先执行作业描述符238。一旦已经执行作业描述符238,则可以调度作业描述符236被控制器208接下来执行。作为又一示例,在通道CH“0”可用、但是通道CH“1”不可用时,定序器216可以调度控制器208先执行作业描述符236、继而为作业描述符238。

[0111] 与用于评估分段可用性的过程相似,如果通道CH“0”可用,则定序器216无需考虑通道CH“1”(作业描述符238被分配到该通道)是否忙碌或者可用,因为作业描述符236是更早发出的描述符,除非更早发出的描述符被分配到的通道不可用,否则可以默认向更早发出的描述符给予超过更晚发出的作业描述符的优先级。

[0112] 在存在多个作业描述符(例如,在作业描述符236和作业描述符238之后发出的作业描述符)时,可以重复前述调度过程以确定这些新作业描述符被分配到的分段中的每个分段是否可用。然后可以向关联通道可用的作业描述符给予优先级,即使这些作业描述符是更晚发出的描述符。

[0113] 作为示例,在存在向通道CH“2”分配的更晚发出的作业描述符240(DES#6)时,定序器216可以通过进一步检查通道CH“2”是否可用在作业描述符236、作业描述符238和作业描述符240之间区分优先顺序。如果通道CH“2”可用(但是通道CH“0”和通道CH“1”不可用),则定序器216可以先调度关联数据的暂时存储和作业描述符240的执行。如在作业描述符236与作业描述符238之间可以应用如先前讨论的调度过程以确定它们各自的优先级和调度(例如,如果通道CH“0”不可用并且通道“1”可用则先调度作业描述符238以用于执行,或者如果通道CH“0”可用并且通道CH“1”不可用则先调度作业描述符236以用于执行)。

[0114] 概括而言,定序器216可以作为在将执行作业描述符时调度的调度器来操作。固件224可以在每个作业描述符中指定定序器216可以用于调度作业描述符的暂时存储和执行的所有分段和通道信息。

[0115] 如上文讨论的那样,每个作业描述符可以作为用于向/从目标闪存设备写入、读取或者擦除数据的命令来工作。可以向连续闪存设备写入或者从连续闪存设备读取或者擦除数据块。备选地,可以向相同闪存设备写入或者从相同闪存设备读取或者擦除数据块。在另一备选地,可以跨越多个闪存设备分布数据块,并且每个闪存设备可以包括各自物理地存储数据的一个或者多个块的一个或者多个扇区。可以发出作业描述符以从多个闪存设备中包括的多个扇区读取数据块或者向该多个扇区写入数据块。

[0116] 调度器216可以在发出每个作业描述符时调度作业描述符的执行(例如,一旦固件224向调度器216发出作业描述符就可以出现调度)。备选地,一旦已经接收所有作业描述符或者其子集,则定序器216可以调度作业描述符的暂时存储和执行。

[0117] 尽管关于基于分段或者通道的可用性调度作业描述符来描述前述实现方式,但是定序器216也可以基于分段和通道二者的可用性执行调度。例如,在向作业描述符分配的分

段和通道均可用时,可以比分段或者通道忙碌的更早发出的作业描述符先调度作业描述符以用于执行。作为另一示例,在向作业描述符分配的分段可用、但是它的通道不可用时,即使向分段不可用的更早发出的作业描述符分配的通道可用,仍然可以比更早发出的作业描述符先调度作业描述符以用于执行。作为又一示例,在向作业描述符分配的分段不可用、但是它的通道可用时,即使向通道不可用的更早发出的作业描述符分配的分段可用,仍然可以比更早发出的作业描述符先调度作业描述符以用于执行。

[0118] 在一些实现方式中,在更早发出的作业描述符和更晚发出的作业描述符二者的相应分段可用时,如果向更晚发出的作业描述符分配的通道可用、但是向更早发出的作业描述符分配的通道不可用,则定序器216可以先调度更晚发出的作业描述符的执行。在存在附加的更晚发出的作业描述符时,如果更晚发出的作业描述符的分段或者通道都可用或者不可用,则可以基于接收的顺序调度更晚发出的作业描述符。

[0119] 如上文讨论的那样,定序器216也可以分别向缓冲管理器213和控制器208查询缓冲器219的分段和通道226的状态。在一些实现方式中,定序器216可以向缓冲管理器213查询特定分段将需要多久变成可用于数据存储的估计或者预测。类似地,定序器216可以向控制器208查询特定通道将需要多久变成可用的估计或者预测。这样的估计可以基于分段或者通道的先前访问或者当前被执行的操作的类型。这一估计然后可以用来在分段可用、但是通道不可用的作业描述符与分段不可用、但是通道可用的作业描述符之间调度。

[0120] 作为示例,如果估计表明更晚发出的作业描述符(例如,具有可用通道和不可用分段)的不可用分段很可能在更早发出的作业描述符(例如,具有可用分段和不可用通道)的不可用通道变成可用之前变成可用,则定序器216可以比更早发出的作业描述符先调度更晚发出的作业描述符以用于执行。

[0121] 估计也可以用来调度具有同等可用分段和/或通道的竞争作业描述符。例如,如果更早发出的作业描述符和更晚发出的作业描述符的分段(或者通道)二者均不可用,那么如果估计表明更晚发出的作业描述符的分段(或者通道)很可能在更早发出的作业描述符的分段(或者通道)变成可用之前先变成可用,则定序器216可以比更晚发出的作业描述符先调度更晚发出的作业描述符以用于执行。

[0122] 除了用于特定分段或者通道变成可用的时间长度之外,估计也可以依赖于其他标准,比如操作的重要性(例如,定序器216可以向必须在很严格时间表内完成的关键作业描述符给予优先级)、作业描述符的条件(例如,必须在可以执行作业描述符之前满足的条件)和依赖约束(例如,基于其他作业描述符的成功完成)。

[0123] 如上文讨论的那样,作业描述符的优先级和调度也可以依赖于被执行的操作的类型。例如,在存在指定写入操作(例如,从分段“6”向闪存设备传送数据)的一个作业描述符(例如,DES#6)和指定读取操作(例如,从闪存设备向分段“7”传送由作业描述符DES#6写入的数据)的另一作业描述符(例如,DES#7)时,定序器216可以调度并且向DES#6给予超过DES#7的优先级。如果向DES#7给予超过DES#6的优先级,则作业描述符DES#7将已经向缓冲器219传送旧数据而不是如由DES#6写入的新数据。

[0124] 可以在数字电路装置中或者在计算机、硬件、固件、软件中或者在它们的组合中实施描述的特征。可以在机器可读存储设备中有形地体现的用于由可编程处理器执行的计算机程序产品中实施装置;并且方法步骤可以由可编程处理器执行,该可编程处理器执行指

令程序以通过对输入数据操作并且生成输出来执行描述的实现方式的功能。可以在可编程系统上可执行的一个或者多个计算机程序中有利地实施描述的特征,该可编程系统包括至少一个可编程处理器、至少一个输入设备和至少一个输出设备,该至少一个可编程处理器被耦合用于从数据存储系统接收数据和指令并且向数据存储系统传输数据和指令。计算机程序是可以在计算机中直接或者间接用来执行某个活动或者带来某个结果的指令集。可以用包括编译或者解译语言的任何形式的编程语言编写计算机程序,并且可以用任何形式部署它,该形式包括作为独立程序或者作为适合用于在计算环境中使用的模块、部件、子例程或者其他单元。

[0125] 用于执行指令程序的适当处理器例如包括通用和专用微处理器二者 and 任何种类的计算机的唯一处理器或者多个处理器之一。一般而言,处理器将从只读存储器或者随机存取存储器或者二者接收指令和数据。计算机的基本单元是用于执行指令的处理器以及用于存储指令和数据的一个或者多个存储器。一般而言,计算机也将包括用于存储数据文件的一个或者多个海量存储设备或者被操作地耦合以用于与该一个或者多个海量存储设备通信;这样的设备包括磁盘,比如内部硬盘和可去除盘;磁光盘;以及光盘。适合用于有形地体现计算机程序指令和数据的存储设备包括所有形式的非易失性存储器,这些形式例如包括半导体存储器设备,比如EPROM、EEPROM和闪存设备;磁盘,比如内部硬盘和可去除盘;磁光盘;以及CD-ROM和DVD-ROM盘。处理器和存储器可以由ASIC(专用集成电路)补充或者并入于ASIC中。

[0126] 此外,图中描绘的逻辑流程无需所示特定顺序或者依次顺序以实现所需结果。此外,可以提供其他步骤或者可以从描述的流程消除步骤并且可以向描述的系统添加或者从描述的系统去除其他部件。因而,其他实现方式在所附权利要求的范围内。

[0127] 已经描述了本发明的多个实施例。然而,将理解,可以进行各种修改而不脱离本发明的精神实质和范围。因而,其他实施例在所附权利要求的范围内。

[0128] 上文已经具体描述了数个实施例,并且各种修改是可能的。可以在电子电路装置、计算机硬件、固件、软件中或者在它们的组合(比如在本说明书中公开的结构装置及其结构等效物)中实施公开的主题内容,该结构装置潜在地包括可操作用于使一个或者多个数据处理装置执行描述的操作的程序(比如在计算机可读介质中编码的程序,该计算机可读介质可以是存储器设备、存储设备、机器可读存储基板或者其他物理机器可读介质,或者它们中的一项或者多项的组合),该主题内容包括在本说明书中描述的功能操作。

[0129] 可以用任何形式的编程语言编写程序(也被称为计算机程序、软件、软件应用、脚本或者代码),该形式包括作为独立程序或者作为适合用于在计算环境中使用的模块、部件、子例程或者其他单元。程序代码未必对应于文件系统中的文件。程序可以存储于保持其他程序或者数据的文件(例如,在标记语言文档中存储的一个或者多个脚本)的一部分中、专用于讨论的程序的单个文件中或者多个协同文件(例如,存储一个或者多个模块、子程序或者代码部分的文件)中。程序可以被部署用于在一个计算机上或者在位于一个地点或者分布于多个地点并且被通信网络互连的多个计算机上执行。

[0130] 尽管本说明书包含许多细节,但是这些不应被解释为限制可以要求保护的内容的范围,而实际上是描述具体实施例特有的特征。在本说明书中在不同的实施例的背景中描述的某些特征也可以在单个实施例中组合实施。换言之,在单个实施例的背景中描述的各

种特征也可以在多个实施例分离地或者在任何适当子组合中实施。另外,虽然上文可以描述特征为在某些组合中动作并且甚至起初这样要求保护,但是来自要求保护的组合的一个或者多个特征可以在一些情况下从该组合中被删除,并且要求保护的组合可以涉及子组合或者子组合的变化。

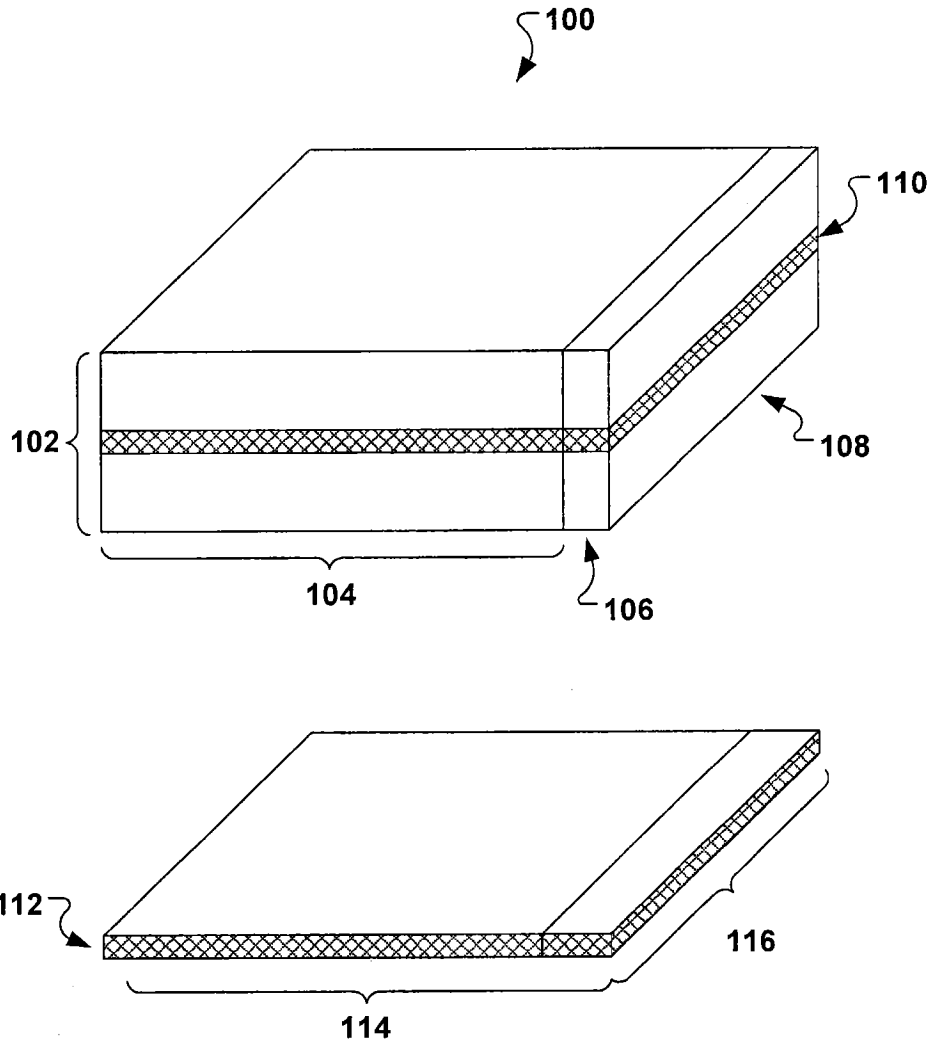


图1

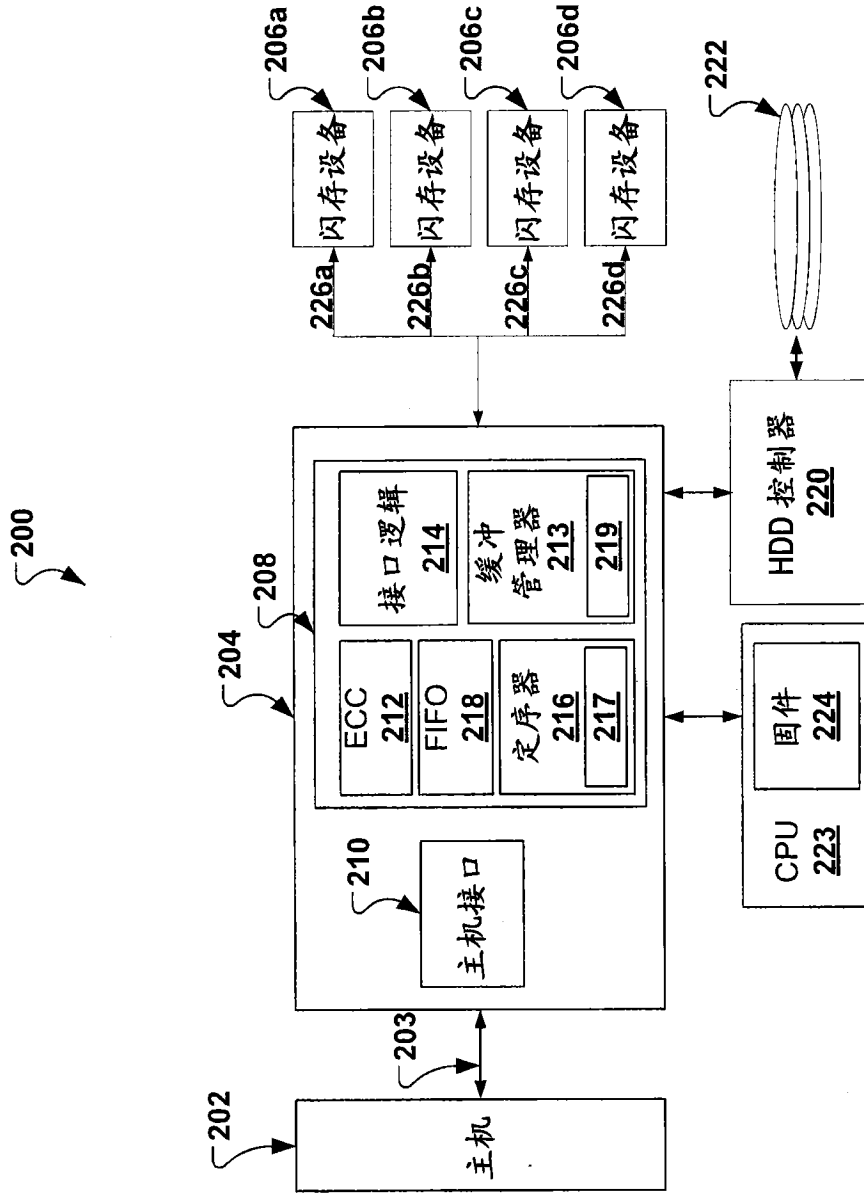


图2A

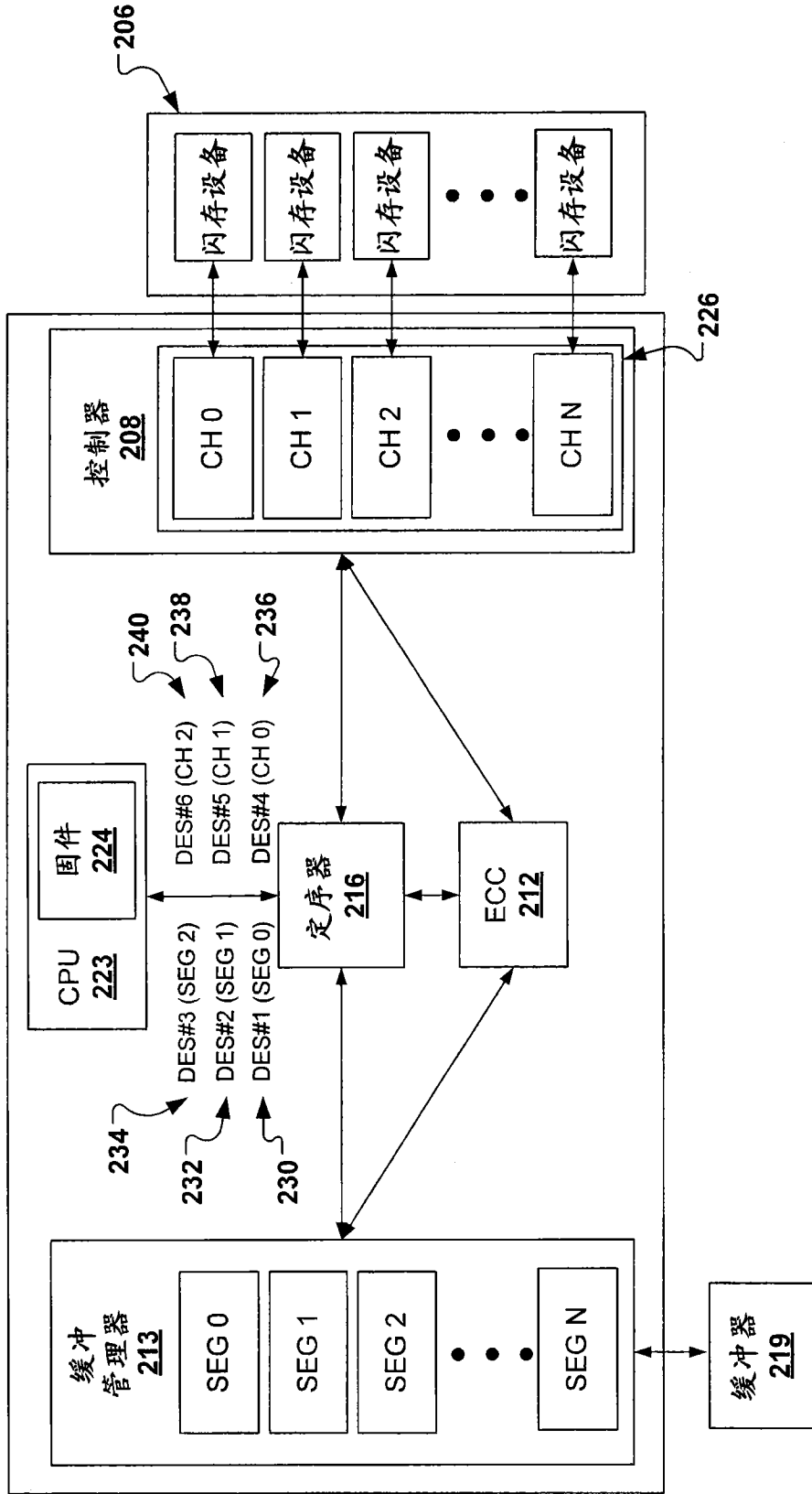


图2B

300

	308 命令编号	310 命令类型	318 通道	312 芯片使能	314 T位	316 存储器
302	1	读取	1	CE_	0	第3
304	2	编程	2	CE_	0	第6
306	3	读取	3	CE_	1	第7

图3

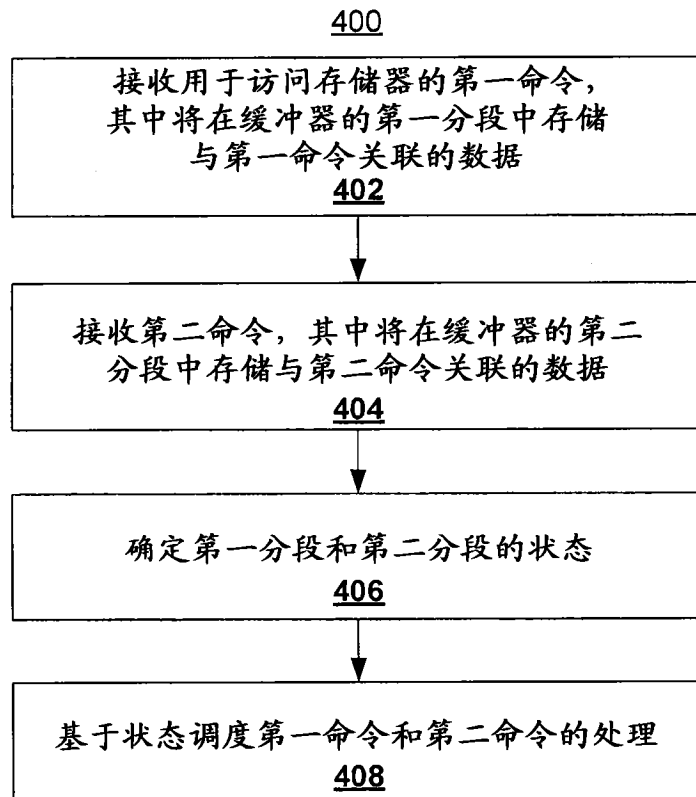


图4