



(12) 发明专利申请

(10) 申请公布号 CN 116860170 A

(43) 申请公布日 2023. 10. 10

(21) 申请号 202310849544.4

(22) 申请日 2023.07.11

(71) 申请人 深圳大普微电子科技有限公司

地址 518000 广东省深圳市龙岗区龙城街道黄阁坑社区腾飞路9号创投大厦3501

(72) 发明人 黄朋 孟鹏涛 杨亚飞

(74) 专利代理机构 深圳市深佳知识产权代理事务

所(普通合伙) 44285

专利代理师 王曙聘

(51) Int. Cl.

G06F 3/06 (2006.01)

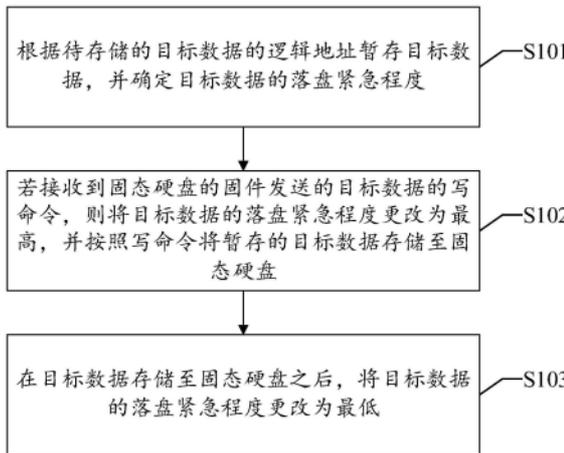
权利要求书2页 说明书10页 附图4页

(54) 发明名称

一种数据处理方法、装置、设备、介质及固态硬盘

(57) 摘要

本申请公开了计算机技术领域内的一种数据处理方法、装置、设备、介质及固态硬盘。本申请在固态硬盘的控制器中设置硬件模块来为固件分担写流程中的数据管理工作,减少固件与控制器的交互。具体的,硬件模块根据待存储的目标数据的逻辑地址暂存目标数据,并确定目标数据的落盘紧急程度;若接收到固态硬盘的固件发送的目标数据的写命令,则将目标数据的落盘紧急程度更改为最高,并按照写命令将暂存的目标数据存储至固态硬盘;在目标数据存储至固态硬盘之后,将目标数据的落盘紧急程度更改为最低。该方案中,硬件模块能够按照数据落盘的紧急程度合理管理数据,由此将落盘紧急程度最高的数据先落盘,能够提升写性能。



1. 一种数据处理方法,其特征在于,应用于固态硬盘的控制器中的硬件模块,包括:
根据待存储的目标数据的逻辑地址暂存所述目标数据,并确定所述目标数据的落盘紧急程度;
若接收到所述固态硬盘的固件发送的所述目标数据的写命令,则将所述目标数据的落盘紧急程度更改为最高,并按照所述写命令将暂存的所述目标数据存储至所述固态硬盘;
在所述目标数据存储至所述固态硬盘之后,将所述目标数据的落盘紧急程度更改为最低。
2. 根据权利要求1所述的方法,其特征在于,所述根据待存储的目标数据的逻辑地址暂存所述目标数据,包括:
以所述逻辑地址为key值,计算相应的value值;
基于所述value值确定所述目标数据对应的内存地址;
若目标缓存区的剩余空间不小于预设第一阈值,则构建所述内存地址与所述目标缓存区中的缓存地址之间的映射关系,将所述目标数据存储至所述缓存地址,并记录所述映射关系。
3. 根据权利要求2所述的方法,其特征在于,还包括:
若所述目标缓存区的剩余空间小于预设第一阈值,则将所述目标数据存储至目标内存区中的所述内存地址。
4. 根据权利要求2所述的方法,其特征在于,还包括:
存储所述逻辑地址与所述内存地址的对应关系,并在哈希位图中插入相应标志信息;所述标志信息用于表示所述内存地址对应有所述目标数据。
5. 根据权利要求4所述的方法,其特征在于,还包括:
若接收到主机端发送的所述目标数据的读请求,则根据所述逻辑地址查询所述哈希位图;
若查询到所述标志信息,则根据所述内存地址获取所述目标数据,并返回所述目标数据至所述主机端;
若未查询到所述标志信息,则从所述固态硬盘获取所述目标数据,并返回所述目标数据至所述主机端。
6. 根据权利要求1所述的方法,其特征在于,所述确定所述目标数据的落盘紧急程度,包括:
按照所述目标数据在当前写操作中的写入顺序确定所述目标数据的落盘紧急程度。
7. 根据权利要求1至6任一项所述的方法,其特征在于,所述确定所述目标数据的落盘紧急程度之后,还包括:
确定所述目标数据的落盘紧急程度的等级标识和所述目标数据的暂存区;所述暂存区为目标缓存区或目标内存区;
在所述暂存区对应的具有所述等级标识的链表中记录所述目标数据在所述暂存区中的存储地址的前一存储地址和后一存储地址,并更新所述链表的总长度和结束地址。
8. 根据权利要求7所述的方法,其特征在于,
若目标缓存区的已用空间大于预设第二阈值,则按照落盘紧急程度由高到低的顺序,将目标内存区中存储的数据迁移至目标缓存区,并相应更新目标缓存区和目标内存区对应

的链表、以及当前被迁移数据在哈希位图中的标志信息；

若目标缓存区的已用空间不大于预设第二阈值，则按照落盘紧急程度由低到高的顺序，将目标缓存区中存储的数据迁移至目标内存区，并相应更新目标缓存区和目标内存区对应的链表、以及当前被迁移数据在哈希位图中的标志信息。

9. 根据权利要求7所述的方法，其特征在于，在所述目标数据存储至所述固态硬盘之后，还包括：

若所述暂存区的剩余空间小于预设第三阈值，则删除所述暂存区中的所述目标数据，并相应更新所述暂存区对应的链表、以及所述目标数据在哈希位图中的标志信息。

10. 根据权利要求1至6任一项所述的方法，其特征在于，所述根据待存储的目标数据的逻辑地址暂存所述目标数据之前，还包括：

申请目标缓存区和目标内存区。

11. 一种数据处理装置，其特征在于，应用于固态硬盘的控制器中的硬件模块，包括：

暂存单元，用于根据待存储的目标数据的逻辑地址暂存所述目标数据，并确定所述目标数据的落盘紧急程度；

存储单元，用于若接收到所述固态硬盘的固件发送的所述目标数据的写命令，则将所述目标数据的落盘紧急程度更改为最高，并按照所述写命令将暂存的所述目标数据存储至所述固态硬盘；

更改单元，用于在所述目标数据存储至所述固态硬盘之后，将所述目标数据的落盘紧急程度更改为最低。

12. 一种电子设备，其特征在于，包括：

存储器，用于存储计算机程序；

处理器，用于执行所述计算机程序，以实现如权利要求1至10任一项所述的方法。

13. 一种可读存储介质，其特征在于，用于保存计算机程序，其中，所述计算机程序被处理器执行时实现如权利要求1至10任一项所述的方法。

14. 一种固态硬盘，其特征在于，包括：控制器，所述控制器中设有硬件模块，所述硬件模块用于实现如权利要求1至10任一项所述的方法。

一种数据处理方法、装置、设备、介质及固态硬盘

技术领域

[0001] 本申请涉及计算机技术领域,特别涉及一种数据处理方法、装置、设备、介质及固态硬盘。

背景技术

[0002] 在固态硬盘上进行数据读写时,依赖固态硬盘的控制器的固件进行数据管理和命令调度。例如:在进行写操作时,固件先从主机端把数据提前存储到DDR,再通过固件FTL处理,生成闪存端的物理信息,并将数据通过DDR送给主控,最后送给闪存端。由固件控制数据读写过程时,固件不仅需要和各端进行通信,还需要管理数据,频繁参与了各种操作,会导致数据读写降低。

[0003] 因此,如何简化控制器固件在数据读写过程中的操作,是本领域技术人员需要解决的问题。

发明内容

[0004] 有鉴于此,本申请的目的在于提供一种数据处理方法、装置、设备、介质及固态硬盘,以简化控制器固件在数据读写过程中的操作。其具体方案如下:

[0005] 第一方面,本申请提供了一种数据处理方法,应用于固态硬盘的控制器中的硬件模块,包括:

[0006] 根据待存储的目标数据的逻辑地址暂存所述目标数据,并确定所述目标数据的落盘紧急程度;

[0007] 若接收到所述固态硬盘的固件发送的所述目标数据的写命令,则将所述目标数据的落盘紧急程度更改为最高,并按照所述写命令将暂存的所述目标数据存储至所述固态硬盘;

[0008] 在所述目标数据存储至所述固态硬盘之后,将所述目标数据的落盘紧急程度更改为最低。

[0009] 可选地,所述根据待存储的目标数据的逻辑地址暂存所述目标数据,包括:

[0010] 以所述逻辑地址为key值,计算相应的value值;

[0011] 基于所述value值确定所述目标数据对应的内存地址;

[0012] 若目标缓存区的剩余空间不小于预设第一阈值,则构建所述内存地址与所述目标缓存区中的缓存地址之间的映射关系,将所述目标数据存储至所述缓存地址,并记录所述映射关系。

[0013] 可选地,还包括:

[0014] 若所述目标缓存区的剩余空间小于预设第一阈值,则将所述目标数据存储至目标内存区中的所述内存地址。

[0015] 可选地,还包括:

[0016] 存储所述逻辑地址与所述内存地址的对应关系,并在哈希位图中插入相应标志信

息;所述标志信息用于表示所述内存地址对应有所述目标数据。

[0017] 可选地,还包括:

[0018] 若接收到主机端发送的所述目标数据的读请求,则根据所述逻辑地址查询所述哈希位图;

[0019] 若查询到所述标志信息,则根据所述内存地址获取所述目标数据,并返回所述目标数据至所述主机端;

[0020] 若未查询到所述标志信息,则从所述固态硬盘获取所述目标数据,并返回所述目标数据至所述主机端。

[0021] 可选地,所述确定所述目标数据的落盘紧急程度,包括:

[0022] 按照所述目标数据在当前写操作中的写入顺序确定所述目标数据的落盘紧急程度。

[0023] 可选地,所述确定所述目标数据的落盘紧急程度之后,还包括:

[0024] 确定所述目标数据的落盘紧急程度的等级标识和所述目标数据的暂存区;所述暂存区为目标缓存区或目标内存区;

[0025] 在所述暂存区对应的具有所述等级标识的链表中记录所述目标数据在所述暂存区中的存储地址的前一存储地址和后一存储地址,并更新所述链表的总长度和结束地址。

[0026] 可选地,若目标缓存区的已用空间大于预设第二阈值,则按照落盘紧急程度由高到低的顺序,将目标内存区中存储的数据迁移至目标缓存区,并相应更新目标缓存区和目标内存区对应的链表、以及当前被迁移数据在哈希位图中的标志信息;

[0027] 若目标缓存区的已用空间不大于预设第二阈值,则按照落盘紧急程度由低到高的顺序,将目标缓存区中存储的数据迁移至目标内存区,并相应更新目标缓存区和目标内存区对应的链表、以及当前被迁移数据在哈希位图中的标志信息。

[0028] 可选地,在所述目标数据存储至所述固态硬盘之后,还包括:

[0029] 若所述暂存区的剩余空间小于预设第三阈值,则删除所述暂存区中的所述目标数据,并相应更新所述暂存区对应的链表、以及所述目标数据在哈希位图中的标志信息。

[0030] 可选地,所述根据待存储的目标数据的逻辑地址暂存所述目标数据之前,还包括:

[0031] 申请目标缓存区和目标内存区。

[0032] 第二方面,本申请提供了一种数据处理装置,应用于固态硬盘的控制器中的硬件模块,包括:

[0033] 暂存单元,用于根据待存储的目标数据的逻辑地址暂存所述目标数据,并确定所述目标数据的落盘紧急程度;

[0034] 存储单元,用于若接收到所述固态硬盘的固件发送的所述目标数据的写命令,则将所述目标数据的落盘紧急程度更改为最高,并按照所述写命令将暂存的所述目标数据存储至所述固态硬盘;

[0035] 更改单元,用于在所述目标数据存储至所述固态硬盘之后,将所述目标数据的落盘紧急程度更改为最低。

[0036] 第三方面,本申请提供了一种电子设备,包括:

[0037] 存储器,用于存储计算机程序;

[0038] 处理器,用于执行所述计算机程序,以实现前述公开的数据处理方法。

[0039] 第四方面,本申请提供了一种可读存储介质,用于保存计算机程序,其中,所述计算机程序被处理器执行时实现前述公开的数据处理方法。

[0040] 通过以上方案可知,本申请提供了一种数据处理方法,应用于固态硬盘的控制器中的硬件模块,包括:根据待存储的目标数据的逻辑地址暂存所述目标数据,并确定所述目标数据的落盘紧急程度;若接收到所述固态硬盘的固件发送的所述目标数据的写命令,则将所述目标数据的落盘紧急程度更改为最高,并按照所述写命令将暂存的所述目标数据存储至所述固态硬盘;在所述目标数据存储至所述固态硬盘之后,将所述目标数据的落盘紧急程度更改为最低。

[0041] 可见,本申请在固态硬盘的控制器中设置硬件模块来为固件分担写流程中的数据管理工作,减少固件与SSD控制器的交互。具体的,硬件模块根据待存储的目标数据的逻辑地址暂存目标数据,并确定目标数据的落盘紧急程度;若接收到固态硬盘的固件发送的目标数据的写命令,则将目标数据的落盘紧急程度更改为最高,并按照写命令将暂存的目标数据存储至固态硬盘;在目标数据存储至固态硬盘之后,将目标数据的落盘紧急程度更改为最低。该方案中,硬件模块能够按照数据落盘的紧急程度划分数据等级,从而按照紧急程度合理管理数据,由此将落盘紧急程度最高的数据先落盘,能够提升写性能。

[0042] 相应地,本申请提供的一种数据处理装置、设备、介质及固态硬盘,也同样具有上述技术效果。

附图说明

[0043] 为了更清楚地说明本申请实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本申请的实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据提供的附图获得其他的附图。

[0044] 图1为本申请公开的一种数据处理方法流程图;

[0045] 图2为本申请公开的一种写流程示意图;

[0046] 图3为本申请公开的一种读流程示意图;

[0047] 图4为本申请公开的一种有关哈希位图的操作及信息示意图;

[0048] 图5为本申请公开的一种优先级确定方式示意图;

[0049] 图6为本申请公开的一种链表示意图;

[0050] 图7为本申请公开的一种硬件模块示意图;

[0051] 图8为本申请公开的一种数据处理装置示意图;

[0052] 图9为本申请公开的一种电子设备示意图。

具体实施方式

[0053] 下面将结合本申请实施例中的附图,对本申请实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本申请一部分实施例,而不是全部的实施例。基于本申请中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本申请保护的范围。

[0054] 目前,在固态硬盘上进行数据读写时,依赖固态硬盘的控制器的固件进行数据管

理和命令调度。例如：在进行写操作时，固件先从主机端把数据提前存储到DDR，再通过固件FTL处理，生成闪存端的物理信息，并将数据通过DDR送给主控，最后送给闪存端。由固件控制数据读写过程时，固件不仅需要和各端进行通信，还需要管理数据，频繁参与了各种操作，会导致数据读写降低。为此，本申请提供了一种数据处理方案，能够简化控制器固件在数据读写过程中的操作，减少固件与SSD控制器的交互，提升写性能。

[0055] 参见图1所示，本申请实施例公开了一种数据处理方法，应用于固态硬盘的控制器中的硬件模块，包括：

[0056] S101、根据待存储的目标数据的逻辑地址暂存目标数据，并确定目标数据的落盘紧急程度。

[0057] 需要说明的是，硬件模块为集成电路实现的电路模块，集成电路可以通过FPGA等实现，其为固态硬盘控制器的一部分。

[0058] 在本实施例中，待存储的目标数据为固态硬盘连接的主机发送的写操作要存储的数据，本实施例设有目标缓存区（即缓存）和目标内存区（即内存）两种暂存区，目标缓存区和目标内存区能够互相减轻对方的读写压力，避免因暂存区响应过慢而影响读写性能。

[0059] 在一种实施方式中，根据待存储的目标数据的逻辑地址暂存目标数据，包括：以逻辑地址为key值，计算相应的value值；基于value值确定目标数据对应的内存地址；若目标缓存区的剩余空间不小于预设第一阈值，则构建内存地址与目标缓存区中的缓存地址之间的映射关系，将目标数据存储至缓存地址，并记录映射关系。若目标缓存区的剩余空间小于预设第一阈值，则将目标数据存储至目标内存区中的内存地址。可见，本实施例以待写数据（即待存储的目标数据）的逻辑地址为key值计算相应的value值，并以此value值确定对应的内存地址，待写数据一定对应有内存地址，但待写数据并非一定存储在内存中，还可能存储在缓存中。当缓存的剩余空间足够多，则构建前述确定的内存地址与某一缓存地址之间的映射关系，并将待写数据存储至该缓存地址，同时记录前述确定的内存地址与数据所暂存的缓存地址之间的映射关系。当缓存的剩余空间不足，则将待写数据直接存储至前述确定的内存地址。由此将待写数据优先暂存至缓存，以提升读写效率，同时减轻内存的压力。

[0060] 在一种实施方式中，还包括：存储逻辑地址与内存地址的对应关系，并在哈希位图中插入相应标志信息；标志信息用于表示内存地址对应有目标数据。如：在位图中记录标志信息“1”，标志信息“1”与内存地址相对应，用于表示该内存地址上存有目标数据或该内存地址所对应的缓存地址上存有目标数据。相应地，如果在位图中，某一内存地址对应有空标志信息“0”，则表示该内存地址上未存数据或该内存地址与任何缓存地址没有映射关系。

[0061] 相应地，借助哈希位图还可以进行数据读操作和数据删除操作，数据读操作用于读取前述标志信息“1”，以读取相应内存地址上的数据；数据删除操作用于删除前述标志信息“1”，也就是使其变更为空标志信息“0”。

[0062] 针对数据读操作的处理包括：若接收到主机端发送的目标数据的读请求，则根据逻辑地址查询哈希位图；若查询到标志信息，则根据内存地址获取目标数据，并返回目标数据至主机端；由于数据暂存于内存或缓存，因此根据内存地址从缓存或内存中读取数据。若未查询到标志信息，则从固态硬盘获取目标数据，并返回目标数据至主机端，具体包括：把固态硬盘里的目标数据放入缓存，再从缓存返回目标数据给主机，之后删除缓存里的目标数据。

[0063] 在一种实施方式中,确定目标数据的落盘紧急程度,包括:按照目标数据在当前写操作中的写入顺序确定目标数据的落盘紧急程度。一个写操作通常写批量数据,目标数据是批量数据中的其中一个,因此按照目标数据在批量数据中的位置(也就是目标数据在当前写操作中的写入顺序)确定其落盘紧急程度。本实施例设定:一个写操作中的先被写的数据的落盘紧急程度高,后被写的数据的落盘紧急程度低;据此,将一个写操作要写的批量数据按照写入顺序排列后,那么此批量数据的落盘紧急程度就按照由高至低排列。其中,暂存至缓存或内存的数据才对应落盘紧急程度。

[0064] S102、若接收到固态硬盘的固件发送的目标数据的写命令,则将目标数据的落盘紧急程度更改为最高,并按照写命令将暂存的目标数据存储至固态硬盘。

[0065] S103、在目标数据存储至固态硬盘之后,将目标数据的落盘紧急程度更改为最低。

[0066] 其中,在目标数据存储至固态硬盘之后,将目标数据写入缓存或内存对应的释放链,以在缓存或内存空间不足时,基于释放链释放缓存或内存的空间。其中,优先释放最早记录至释放链的数据。

[0067] 需要说明的是,固态硬盘的固件能够将主机发送的目标数据的逻辑地址转换为相应地物理地址,具体包括通道信息等,固件将这些信息以写命令的形式发送至硬件模块,硬件模块据此写命令对目标数据进行落盘存储。由于目标数据即将要落盘存储,因此将其落盘紧急程度更改为最高。在目标数据落盘存储后,将目标数据的落盘紧急程度更改为最低。

[0068] 本实施例中,由于数据暂存于缓存或内存,因此缓存和内存里的数据都有自己的落盘紧急程度,当落盘紧急程度划分有多个等级时,每一等级有自己的等级标识;且每一等级的数据在暂存区的地址在专属的链表中进行记录。例如:对于内存里的数据,不同数据可能对应不同落盘紧急程度,如果落盘紧急程度共有3个等级:中、高、底(此即为等级标识),那么中等级对应一个链表A-中,用于记录中等级的数据在内存里的位置;高等级对应一个链表B-高,用于记录高等级的数据在内存里的位置;低等级对应一个链表C-低,用于记录低等级的数据在内存里的位置(用数据所在地址的前一地址和后一地址表示)。在一种实施方式中,确定目标数据的落盘紧急程度之后,还包括:确定目标数据的落盘紧急程度的等级标识和目标数据的暂存区;暂存区为目标缓存区或目标内存区;在暂存区对应的具有等级标识的链表中记录目标数据在暂存区中的存储地址的前一存储地址和后一存储地址,并更新链表的总长度和结束地址。

[0069] 进一步地,缓存和内存里的数据可以相互迁移,若目标缓存区的已用空间大于预设第二阈值(即缓存空间不足),则按照落盘紧急程度由高到低的顺序,将目标内存区中存储的数据迁移至目标缓存区,以将内存里的高落盘紧急程度的数据迁移至缓存,使得此类数据尽早落盘存储,并相应更新目标缓存区和目标内存区对应的链表、以及当前被迁移数据在哈希位图中的标志信息;若目标缓存区的已用空间不大于预设第二阈值(即缓存空间充足),则按照落盘紧急程度由低到高的顺序,将目标缓存区中存储的数据迁移至目标内存区,以将缓存里的低落盘紧急程度的数据迁移至内存,避免此类数据占用缓存资源,并相应更新目标缓存区和目标内存区对应的链表、以及当前被迁移数据在哈希位图中的标志信息。

[0070] 在一种实施方式中,在目标数据存储至固态硬盘之后,还包括:若暂存区的剩余空间小于预设第三阈值(即缓存或内存空间不足),则删除暂存区中的目标数据,以清理缓存

或内存,并相应更新暂存区对应的链表、以及目标数据在哈希位图中的标志信息。

[0071] 为了避免硬件模块申请缓存和内存占用额外时间,在一种实施方式中,根据待存储的目标数据的逻辑地址暂存目标数据之前,还包括:申请目标缓存区和目标内存区。也就是:在读写请求到达固态硬盘之前,预先申请缓存和内存。

[0072] 可见,本实施例在固态硬盘的控制器中设置硬件模块来为固件分担写流程中的数据管理工作,减少固件与SSD控制器的交互。具体的,硬件模块根据待存储的目标数据的逻辑地址暂存目标数据,并确定目标数据的落盘紧急程度;若接收到固态硬盘的固件发送的目标数据的写命令,则将目标数据的落盘紧急程度更改为最高,并按照写命令将暂存的目标数据存储至固态硬盘;在目标数据存储至固态硬盘之后,将目标数据的落盘紧急程度更改为最低。该方案中,硬件模块能够按照数据落盘的紧急程度划分数据等级,从而按照紧急程度合理管理数据,由此将落盘紧急程度最高的数据先落盘,能够提升写性能。

[0073] 下面对读写流程进行详细介绍。

[0074] 如图2所示,主机发送要写的携带LBA(Logical Block Address,逻辑地址)的数据过来,数据和LBA被传输至硬件模块,控制信息交给FTL。硬件模块将LBA当作Cuckoo hash的key值,对哈希位图执行Hash插入操作,对数据进行暂存;当FTL组建写编程命令后,将携带LBA的命令送给硬件模块,将PHA(Physical Block Address)送入后端;硬件模块通过LBA确定数据的暂存位置,然后通过S4和S5,最终将数据送给FLASH。

[0075] 如图3所示,主机发送一个读命令时,首先会通过S1搜索Cache(缓存)和DDR(内存),具体的:通过Cuckoo hash搜索确定哈希位图中是否存在相同LBA的标志信息;若存在,则直接通过S2把数据返回给主机;若不存在,则通过S3把LBA信息送给FTL,以获取到闪存颗粒的物理信息,然后通过S4读取闪存颗粒,最后通过S5、S6、S7把数据送给主机。

[0076] 在读写过程中,最多与FTL软件进行了2次交互,大大降低了软件参与读写的次数,能够降低读写延迟。并且,采用Cache和DDR的方式可以充分利用暂存空间,并且将数据优先暂存至缓存,以减少写操作的延迟。同时,利用链表对Cache和DDR的内部空间进行管理,具体为:利用链表对Cache和DDR里的不同落盘紧急程度的数据进行记录,并将内存里的高落盘紧急程度的数据尽可能迁移至缓存,将缓存里的低落盘紧急程度的数据尽可能迁移至内存。当数据从Cache或DDR中被删除,该数据脱离链表,由此增大数据在Cache或DDR的停留时间,以提高读操作命中率。

[0077] 本实施例通过cuckoo hash算法以较少的计算换取较大的空间,并在哈希位图中进行LBA插入、LBA查找和LBA删除。通过哈希位图记录相应LBA是否对应有有效数据,可以减少对Cache或DDR的访问次数。当插入完成后,将哈希位图中对应的比特位Flag置1;LBA删除后,将哈希位图中对应的比特位Flag置0;LBA查找时,只需查询位图对应的比特位Flag是1还是0;若是1,则直接通过图3的S2把数据返回给主机;若不存在,则通过图3的S3、S4、S5、S6、S7把数据送给主机。其中,与哈希位图相关的操作及信息记录如图4所示。

[0078] 需要说明的是,Cache和DDR里存的数据按照落盘紧急程度划分有优先级,同一优先级的数据的位置记录在一个链表里。请参见图5,如果主机要写入的数据共有2个128K的数据包:数据包S和数据包F,数据包S按照4K划分可得到32个4K,标注为S0~S31;数据包F按照4K划分可得到32个4K,标注为F0~F31。FTL组建写编程命令时先从数据包S中拿S0和S1、从数据包F中拿F0和F1,然后将S0、S1、F0、F1组为第一个写命令,此时S0、S1、F0、F1即为落盘

紧急程度最高的数据。相应地,第二个写命令包括的数据S2、S3、F2、F3为落盘紧急程度第二高的数据,以此类推,第15个写命令包括的数据S30、S31、F30、F31为落盘紧急程度最低的数据。由此按照落盘紧急程度划分的优先级由高至低可以表示为:15、14、13……0。其中,FTL组建一个写编程命令时,从几个数据包里拿数据,从每一数据包中拿几个多大的数据,由固态硬盘的读写机制确定。

[0079] 数据说明的是,需写入的数据会在系统中经历如下几个阶段:

[0080] S1:主机输入数据,携带内存地址和优先级第一次输入;

[0081] S2:FTL构建写命令,该系统锁定当前数据;

[0082] S3:数据被送给后端;

[0083] S4:数据被成功写入闪存。

[0084] 在这4个阶段中,硬件模块对数据的优先级会进行自动调整。在第一阶段,数据优先级按照图5的方式进行确定;在第二个阶段,当数据即将被写入到闪存时(还未写),把该数据优先级置为15(15表示优先级最高,表示数据被锁定,如果数据在缓存,那么数据不会被搬移到DDR);第三阶段,数据被送给后端后,将其优先级置为0(0表示优先级最低)。由于数据的优先级有调整,因此该数据对应的链表也需相应调整。

[0085] 同一优先级的数据用一个链表管理时,可参照如下过程进行。如图6所示,比如陆续到来了优先级0的4个数据,其地址分别为`ddr_id0`、`ddr_id2`、`ddr_id3`和`ddr_id5`。当`ddr_id0`到来时,记录优先级0对应的链表的`first_ddr_id=ddr_id0`,`new_ddr_id=ddr_id0`,`total_cnt=1`。当`ddr_id2`到来时,记录优先级0对应的链表的`first_ddr_id=ddr_id0`,`new_ddr_id=ddr_id2`,`total_cnt=2`,并更新`ddr_id0`的`nxt_addr=ddr_id2`,`ddr_id2`的`pre_addr=ddr_id0`。当`ddr_id3`到来时,记录优先级0对应的链表的`first_ddr_id=ddr_id0`,`new_ddr_id=ddr_id3`,`total_cnt=3`,并更新`ddr_id2`的`nxt_addr=ddr_id3`,`ddr_id3`的`pre_addr=ddr_id2`。当`ddr_id5`到来时,同理也一样。`first_ddr_id`表示链表的首地址,`total_cnt`表示链表的长度,`new_ddr_id`表示链表的尾地址,同时,链表中每一数据对应应有`nxt_addr`(后一地址)和`pre_addr`(前一地址)。

[0086] 本实施例支持Cache和DDR里数据的相互搬移。为Cache设置两个阈值:`move2ddr_threthold`和`move2cache_threthold`。当cache的使用空间超过`move2ddr_threthold`,硬件模块会把cache中最晚存入的低优先级数据自动迁移到DDR的对应位置;当cache的使用空间小于`move2cache_threthold`,硬件模块会自动把DDR里的最早存入的高优先级数据自动迁移到Cache。可见Cache里的数据搬移到DDR原则为:优先去查看低优先级的`total_cnt`是否为非零,若为非零,会把当前优先级的`new_ddr_id`搬移到DDR中。同理,DDR搬移到Cache的原则为:会优先去查看高优先级的`total_cnt`是否为非零,若为非零,会把当前优先级`first_ddr_id`搬移到Cache中。

[0087] 对于优先级为15的数据,硬件模块会锁存数据,即:当数据的优先级为15后,当前优先级数据不会move到DDR,这样的好处是:给外部一个确定状态,用于外部对该数据进行快速操作处理。

[0088] 如前所述,数据在不同阶段优先级存在相应调整,通过合理的优先级调整以及数据搬移,让Cache中的数据一直进行动态更新,让更高优先级的数据保存在Cache中,从而在有限的Cache空间及较大的DDR空间下,不仅能存储较多数据,并能动态管理数据,提升系统

整体性能。经实验验证,当时钟为600MHz,数据位宽为32B时,以此方式可使数据吞吐性能基本稳定在18.75GB/s。

[0089] 当DDR或Cache空间快满时,删除DDR或Cache里的某些数据,并对其所对应链表进行相应内容的删除,由此保证DDR或Cache充满需操作的数据,提高DDR或Cache的命中率。

[0090] 请参见图7,硬件模块中的各个功能模块对数据读写的操作流程如下:

[0091] S1:通过DDR预取模块和Cache预取模块申请DDR空间和Cache空间。

[0092] S2:接收前端操作,并把数据放置到DDR或Cache的对应位置。

[0093] S3:当需要读取当前LBA的内容,通过S3将当前LBA的优先级锁定为15,锁定当前的数据。

[0094] S4:通过S4发送读消息。

[0095] S5:系统通过S5把数据传递给后端。

[0096] S6:当写成功后,将数据记录在释放链中,以在DDR或Cache空间快满时,通过S6对释放链中的数据进行删除。

[0097] 本实施例采用硬件封闭式管理,主要分为两部分:前后端数据匹配和空间管理。对于前端数据和后端数据匹配,本实施例采用了cuckoo hash算法,以主机的LBA作为key,能够完成大数量的地址搜索匹配。在数据管理方面采用了传统的Cache和DDR组合的管理方式,在Cache和DDR中对数据进行了优先级划分。每个主机过来的数据采用优先级管理,需要尽快落盘的数据划分为高优先级,不需要尽快落盘的数据划分为相对低优先级。通过合理的优先级调整以及数据搬移,让Cache中的数据一直进行动态更新,将高优先级的后到达的数据优先导入到Cache中,保证数据尽可能锁定到Cache中,提升数据的传输性能。当写数据执行成功后,数据并不会在Cache或DDR进行即时释放,而是构建了一个释放链,当Cache或DDR空间不够时,会自动在释放链中释放相应数据,从而提高了数据在Cache和DDR中保存的时间,提高了读命中概率。

[0098] 在写流程中,固件只需要保存好主机过来的LBA,通过FTL构建的写指令,系统自动将数据送到存储颗粒;在读流程中,硬件模块可自动完成读搜索。可见,本实施例解放了FTL管理数据的操作,还简化了读写流程,减少了FTL和SSD主控之间的频繁操作,能够提升系统性能,保证读写稳定性。由此可提高固态硬盘的性能稳定性。

[0099] 下面对本申请实施例提供的一种数据处理装置进行介绍,下文描述的一种数据处理装置与本文描述的其他实施例可以相互参照。

[0100] 参见图8所示,本申请实施例公开了一种数据处理装置,应用于固态硬盘的控制器中的硬件模块,包括:

[0101] 暂存单元801,用于根据待存储的目标数据的逻辑地址暂存目标数据,并确定目标数据的落盘紧急程度;

[0102] 存储单元802,用于若接收到固态硬盘的固件发送的目标数据的写命令,则将目标数据的落盘紧急程度更改为最高,并按照写命令将暂存的目标数据存储至固态硬盘;

[0103] 更改单元803,用于在目标数据存储至固态硬盘之后,将目标数据的落盘紧急程度更改为最低。

[0104] 在一种实施方式中,暂存单元具体用于:

[0105] 以逻辑地址为key值,计算相应的value值;

- [0106] 基于value值确定目标数据对应的内存地址；
- [0107] 若目标缓存区的剩余空间不小于预设第一阈值，则构建内存地址与目标缓存区中的缓存地址之间的映射关系，将目标数据存储至缓存地址，并记录映射关系。
- [0108] 在一种实施方式中，暂存单元具体用于：
- [0109] 若目标缓存区的剩余空间小于预设第一阈值，则将目标数据存储至目标内存区中的内存地址。
- [0110] 在一种实施方式中，还包括：
- [0111] 位图更新单元，用于存储逻辑地址与内存地址的对应关系，并在哈希位图中插入相应标志信息；标志信息用于表示内存地址对应有目标数据。
- [0112] 在一种实施方式中，还包括：
- [0113] 读操作单元，用于若接收到主机端发送的目标数据的读请求，则根据逻辑地址查询哈希位图；若查询到标志信息，则根据内存地址获取目标数据，并返回目标数据至主机端；若未查询到标志信息，则从固态硬盘获取目标数据，并返回目标数据至主机端。
- [0114] 在一种实施方式中，暂存单元具体用于：
- [0115] 按照目标数据在当前写操作中的写入顺序确定目标数据的落盘紧急程度。
- [0116] 在一种实施方式中，还包括：
- [0117] 链表更新单元，用于确定目标数据的落盘紧急程度的等级标识和目标数据的暂存区；暂存区为目标缓存区或目标内存区；在暂存区对应的具有等级标识的链表中记录目标数据在暂存区中的存储地址的前一存储地址和后一存储地址，并更新链表的总长度和结束地址。
- [0118] 在一种实施方式中，还包括：
- [0119] 迁移单元，用于若目标缓存区的已用空间大于预设第二阈值，则按照落盘紧急程度由高到低的顺序，将目标内存区中存储的数据迁移至目标缓存区，并相应更新目标缓存区和目标内存区对应的链表、以及当前被迁移数据在哈希位图中的标志信息；若目标缓存区的已用空间不大于预设第二阈值，则按照落盘紧急程度由低到高的顺序，将目标缓存区中存储的数据迁移至目标内存区，并相应更新目标缓存区和目标内存区对应的链表、以及当前被迁移数据在哈希位图中的标志信息。
- [0120] 在一种实施方式中，还包括：
- [0121] 删除操作单元，用于若暂存区的剩余空间小于预设第三阈值，则删除暂存区中的目标数据，并相应更新暂存区对应的链表、以及目标数据在哈希位图中的标志信息。
- [0122] 在一种实施方式中，还包括：
- [0123] 申请单元，用于根据待存储的目标数据的逻辑地址暂存目标数据之前，申请目标缓存区和目标内存区。
- [0124] 其中，关于本实施例中各个模块、单元更加具体的工作过程可以参考前述实施例中公开的相应内容，在此不再进行赘述。
- [0125] 可见，本实施例提供了一种数据处理装置，能够简化控制器固件在数据读写过程中的操作，减少固件与SSD控制器的交互，提升读写性能。
- [0126] 下面对本申请实施例提供的一种电子设备进行介绍，下文描述的一种电子设备与本文描述的其他实施例可以相互参照。

[0127] 参见图9所示,本申请实施例公开了一种电子设备,包括:

[0128] 存储器901,用于保存计算机程序;

[0129] 处理器902,用于执行所述计算机程序,以实现上述任意实施例公开的方法。

[0130] 下面对本申请实施例提供的一种可读存储介质进行介绍,下文描述的一种可读存储介质与本文描述的其他实施例可以相互参照。

[0131] 一种可读存储介质,用于保存计算机程序,其中,所述计算机程序被处理器执行时实现前述实施例公开的数据处理方法。关于该方法的具体步骤可以参考前述实施例中公开的相应内容,在此不再进行赘述。

[0132] 本申请涉及的“第一”、“第二”、“第三”、“第四”等(如果存在)是用于区别类似的对象,而不必用于描述特定的顺序或先后次序。应该理解这样使用的数据在适当情况下可以互换,以便这里描述的实施例能够以除了在这里图示或描述的内容以外的顺序实施。此外,术语“包括”和“具有”以及他们的任何变形,意图在于覆盖不排他的包含,例如,包含了一系列步骤或单元的过程、方法或设备不必限于清楚地列出的那些步骤或单元,而是可包括没有清楚地列出的或对于这些过程、方法或设备固有的其它步骤或单元。

[0133] 需要说明的是,在本申请中涉及“第一”、“第二”等的描述仅用于描述目的,而不能理解为指示或暗示其相对重要性或者隐含指明所指示的技术特征的数量。由此,限定有“第一”、“第二”的特征可以明示或者隐含地包括至少一个该特征。另外,各个实施例之间的技术方案可以相互结合,但是必须是以本领域普通技术人员能够实现为基础,当技术方案的结合出现相互矛盾或无法实现时应当认为这种技术方案的结合不存在,也不在本申请要求的保护范围之内。

[0134] 本说明书中各个实施例采用递进的方式描述,每个实施例重点说明的都是与其它实施例的不同之处,各个实施例之间相同或相似部分互相参见即可。

[0135] 结合本文中所公开的实施例描述的方法或算法的步骤可以直接用硬件、处理器执行的软件模块,或者二者的结合来实施。软件模块可以置于随机存储器(RAM)、内存、只读存储器(ROM)、电可编程ROM、电可擦除可编程ROM、寄存器、硬盘、可移动磁盘、CD-ROM、或技术领域内所公知的任意其它形式的可读存储介质中。

[0136] 本文中应用了具体个例对本申请的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本申请的方法及其核心思想;同时,对于本领域的一般技术人员,依据本申请的思想,在具体实施方式及应用范围上均会有改变之处,综上所述,本说明书内容不应理解为对本申请的限制。

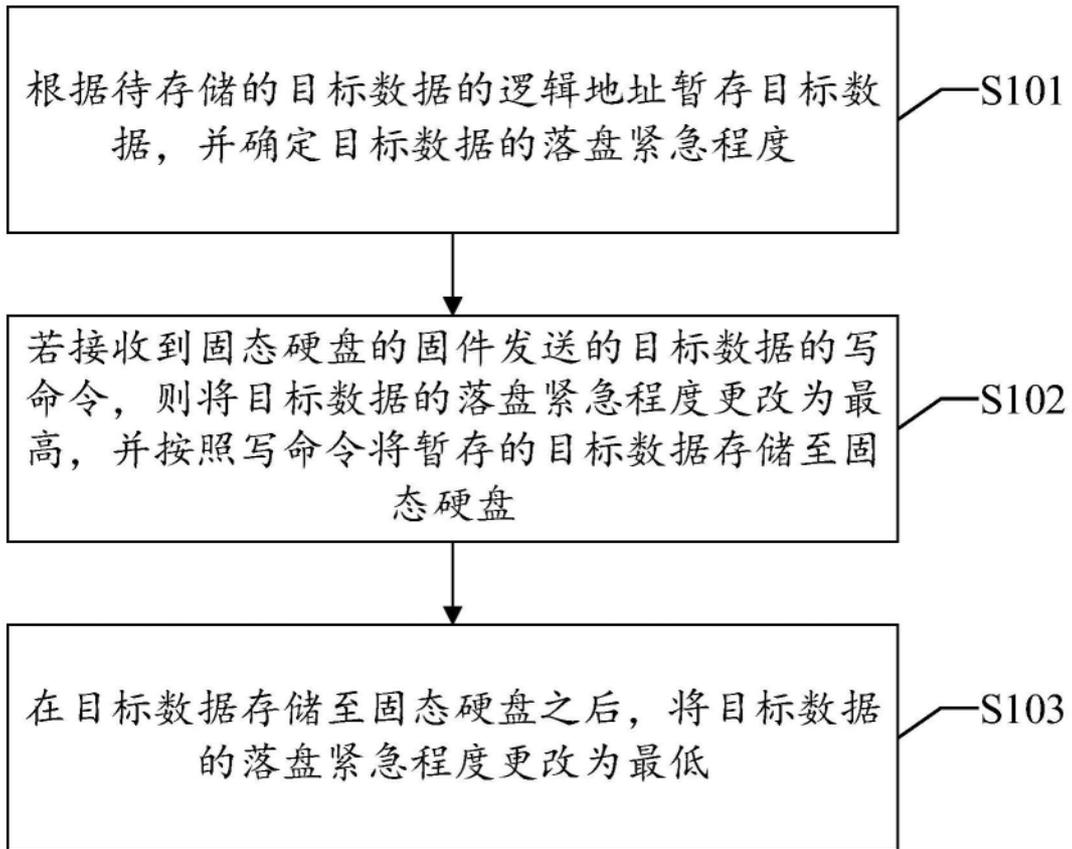


图1

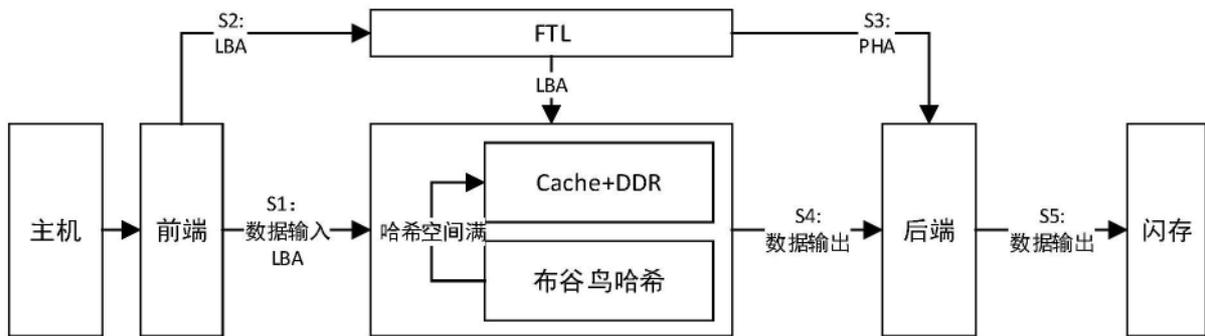


图2

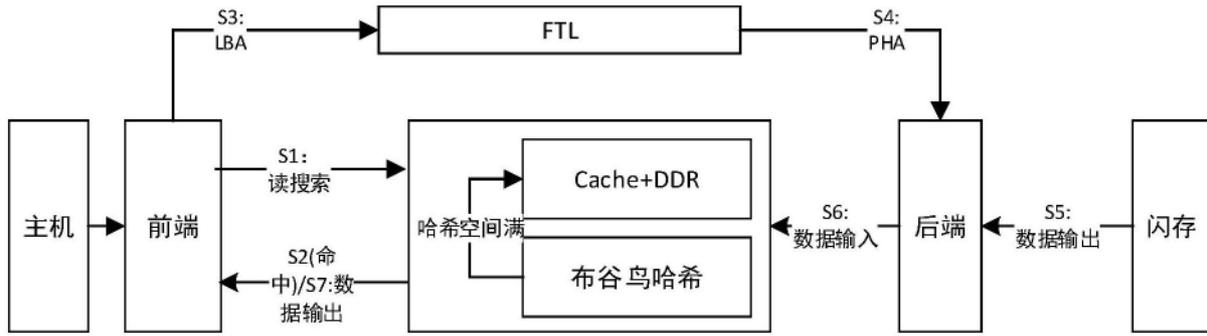


图3

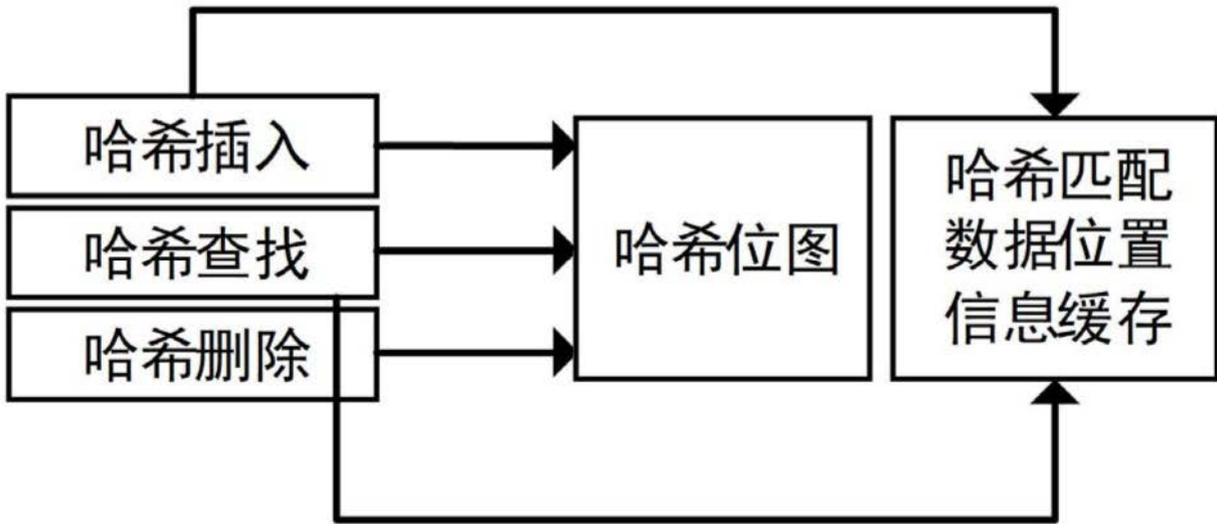


图4

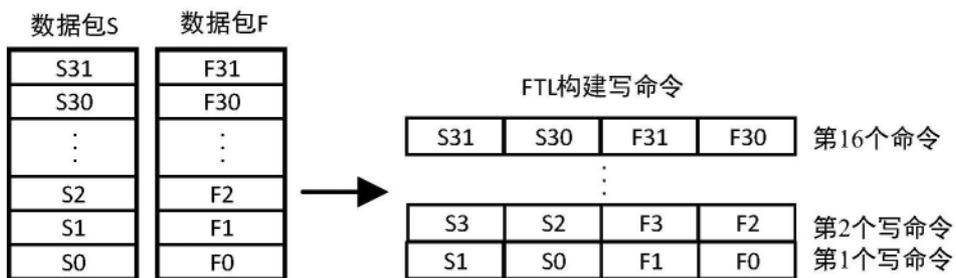


图5



图6

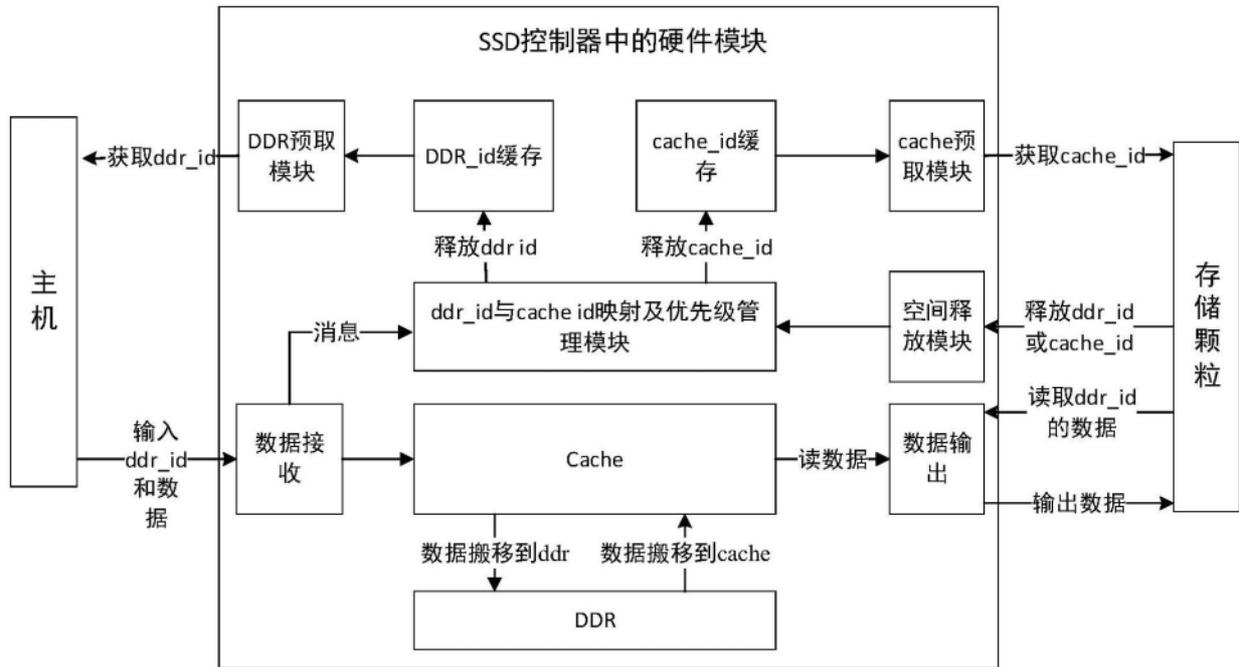


图7

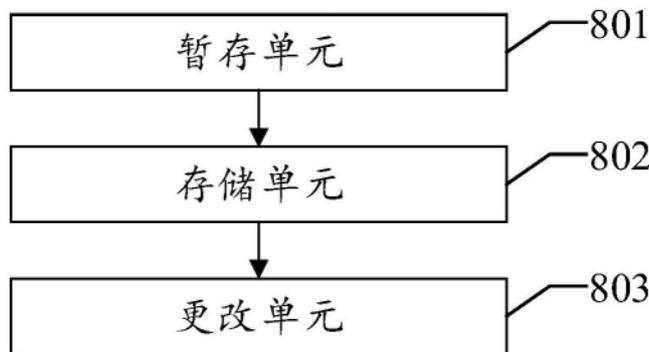


图8

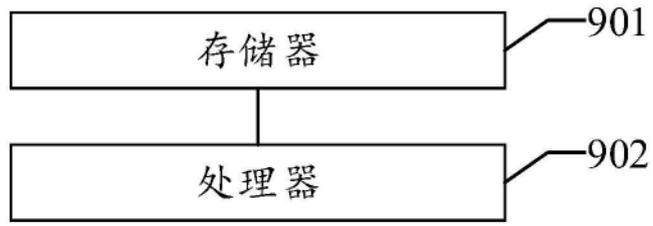


图9