

基于区域页级映射的闪存转换层设计

蔡晓乐^{1,2}, 张亚棣^{1,2}, 李亚晖^{1,2}, 海钰琳^{1,2}

(1 中航工业 西安航空计算技术研究所, 陕西 西安 710068;

2 机载弹载计算机航空科技重点实验室, 陕西 西安 710068)

摘 要: 针对固态硬盘中闪存不能进行本地更新操作以及擦写次数有限的特性, 基于多通道并行结构的固态盘, 提出一种基于区域的页级地址映射方案, 将连续逻辑页分配到不同的通道中并利用三级地址映射方法, 利用动态和静态损耗均衡算法, 提出了分层的损耗均衡算法和垃圾回收方案. 分析表明该算法能释放固态硬盘内存空间, 延长固态硬盘的寿命, 提高整体性能, 映射表内存占用相对传统页级映射减少近 70%.

关键词: 固态硬盘; 多通道结构; 闪存转换层; 地址映射; 损耗均衡

中图分类号: TP301

文献标识码: A

文章编号: 1000—7180(2016)04—0032—05

Design of a Region-Based Page Mapping Flash Translation Layer

CAI Xiao-le^{1,2}, ZHANG Ya-di^{1,2}, LI Ya-hui^{1,2}, HAI Yu-lin^{1,2}

(1 Xi'an Aeronautics Computing Technique Research Institute, AVIC, Xi'an 710068, China;

2 Aviation Key Laboratory of Science and Technology on Airborne and Missile-borne Computer, Xi'an 710068, China)

Abstract: In allusion to the shortcoming of lacking in-place write and the limited erasing operations of flash memory in SSD (Solid-State Drive), this paper proposes a region-based page level address mapping algorithm considering the multi-channel architecture of SSD. The algorithm assigns continuous logical pages to different channels and imposes three-level address mapping method, besides, a scheme of hierarchical wear-leveling and garbage collection method is proposed in the basis of dynamic and static wear-leveling algorithm. Theoretical analysis shows that the algorithm we proposed can release memory space in SSD, prolong the life-span of SSD and improve the whole performance, the memory usage of mapping table can be decreased by nearly 70% compared with original page-level mapping method.

Key words: solid-state drive; multi-channel architecture; flash translation layer; address mapping; wear leveling

1 引言

随着现代航空电子系统高速发展, 对新一代机载存储设备提出大容量、高带宽、可扩展的新要求. 基于闪存的固态硬盘(Solid-State Drive)是近年应用比较广泛的一种新型存储设备, 相比于传统硬盘, 其在性能、功耗、可靠性和存储密度方面均具有显著优势, 可实现具有高带宽的存储访问速度、高可靠的数据容错能力和智能存储管理功能的大容量存储组件, 从而成为航空数据存储系统中的首选.

目前被广泛采用的闪存 NAND 和 NOR 两

种类型. 相对于 NOR, NAND 闪存的主要优点是存储密度更高, 写入速度更快, 因而广泛应用于大容量的存储系统中. 但 NAND 型闪存也存在“异地更新”、有限的块擦写次数等问题. 为了解决这些不足, 使闪存(本文中闪存均指 NAND 型闪存)像传统块设备一样工作, 固态硬盘系统中出现了一种称为闪存转换层(Flash Translation Layer, FTL)^[1]的嵌入式软件, 将逻辑地址映射为物理地址. 如何有效地执行地址映射并管理映射信息, 成为一个关键问题. 本文讨论了常见 FTL 算法, 提出一种适用于多通道固态硬盘的基于区域映射的算法, 分析对比其特点和性能.

2 相关 FTL 地址映射算法

典型的基于页的地址映射算法有 DFTL^[2], TPFTL^[3]等. 由于粒度小, 基于页的地址映射算法具有灵活性高、性能好的优点, 但是所需要的 RAM 空间较大. 此外, 由于地址映射表经常会更新, 机载系统环境恶劣, 映射表太大会导致更新的地址映射表在掉电等情况下无法恢复到闪存中去, 导致文件系统无法恢复到正确的状态, 严重影响系统的可靠性.

NFTL^[4]算法是基于块的地址映射算法. 块映射表的大小仅为页映射表大小的 $1/P$, P 为每个块内包含的页数. 因此, 块级地址映射只需要很小的 RAM 空间存储映射信息. 但是, 这种映射方法不灵活, 逻辑页只能被映射到同一个物理页.

基于混合的地址映射的算法同时利用页级和块级映射, 有 BAST^[5], FAST^[6], Super Block^[7]等. Kim 等人提出了 BAST 算法, 该算法针对替换算法做了改进, 一个数据块对应一个日志块, 同一个数据块的更新操作会顺序写入对应的日志块中. 然而在随机写操作比较频繁的情况下, 日志块的利用率非常低. Lee 等人针对这个问题改进了 BAST 算法, 提出了 FAST 算法, 它使得一个日志块不仅仅对应一个数据块, 而是让一个日志块可以对应所有的数据块, 这增加了映射的灵活性, 但也增加了一些存储开销. Superblock 算法提出了多个块组成一个超级块的方法, 在超级块上采用粗粒度映射, 而在超级块内部采用以页为单位的映射. 它将存储细粒度的映射信息存储到闪存的冗余空间, 引入了大量的对冗余空间的读写操作.

上述提到的所有映射算法, 包括页级映射、块级映射、混合映射等, 都是针对闪存本身的读写特性提出的映射算法, 而并没有考虑到多层次并行固态硬盘的特点. 本文在已有算法的基础上, 提出一种适用于多通道并行固态硬盘结构的映射算法——区域页级映射算法.

3 基于区域页级映射的 FTL

在本文所描述的区域映射关系中, 将逻辑页分为几个区域, 并且逻辑页可以映射到所在区域内相应物理块的物理页上, 当数据需要更新时, 新的数据页会映射到区域内空闲物理页上. 下面首先介绍区域页级映射中适用于多通道并行结构固态硬盘的逻辑页和物理页的分配方法, 接着分析区域映射的映射

机制及其损耗均衡和垃圾回收等问题.

3.1 适用多通道结构固态硬盘的分配算法

在现在的高性能固态硬盘中, 存在多个通道, 每个通道上拥有多个闪存芯片, 每个芯片中存在多个晶圆, 每个晶圆中存在多个面, 也就是说, 固态硬盘中存在四个层次的并行性, 文献[8]讨论了并行性顺序的优先级, 并得出按照<通道级, 晶圆级, 面级, 芯片级>这个顺序分配能使固态硬盘发挥到最佳性能的结论. 本文讨论的固态硬盘都假设有 8 个通道, 每个通道有 2 个芯片, 每个芯片上 2 个晶圆, 每个晶圆上 2 个面, 每个面上 2 048 个块, 每个块有 128 个物理页, 每个物理页大小为 4 kB^[9], 固态硬盘总大小为 64 GB.

在多通道结构的固态硬盘中^[10], 一个物理页由通道号、芯片号、晶圆号、面号、块号和块内页号组成, 如表 1 所示. 对于上面提到的固态硬盘, 每个号所对应的位数如表 1 中第二行. 那么一个物理页的表项大小为 $3+1+1+1+11+7=24$ bit, 根据<通道级, 晶圆级, 面级, 芯片级>这个分配顺序, 按照下述公式可以将逻辑页分配到每个物理页上, 分配公式如下:

表 1 物理页参数确定

通道号	芯片号	晶圆号	面号	块号	块内页号
3	1	1	1	11	7
通道号 = $\text{lpn} \% \text{chssd}$ (1)					
晶圆号 = $(\text{lpn} / \text{chssd}) \% \text{diechip}$ (2)					
面号 = $(\text{lpn} / (\text{chssd} \times \text{diechip})) \% \text{planedie}$ (3)					
芯片号 = $(\text{lpn} / (\text{chssd} \times \text{diechip} \times \text{planedie})) \% \text{chipch}$ (4)					
页号 = $(\text{lpn} / (\text{chssd} \times \text{diechip} \times \text{planedie} \times \text{chipch})) \% \text{pageblk}$ (5)					
块号 = $(\text{lpn} / (\text{chssd} \times \text{diechip} \times \text{planedie} \times \text{chipch} \times \text{pageblk})) \% \text{blkplane}$ (6)					

式中, lpn 为逻辑页号; chssd 为固态硬盘中通道数; diechip 为每个芯片上晶圆数; planedie 为每个晶圆上面数; chipch 为每个通道上芯片数; pageblk 为每个块上的物理页数; blkplane 为每个面上的块数.

按照式(1)~(6), 可以将连续的逻辑页均匀地分布到固态硬盘的各个通道各个芯片上去, 如图 1 所示, 连续的逻辑页被分成 8 个区域, 每一个分别对应一个区域, 每个区域中的逻辑页按照并行顺序的分配方式, 分别分配到各个通道的各个芯片晶圆当中去. 图 1 中右边数字即为文中固态硬盘每个区域所对应的逻辑地址范围.

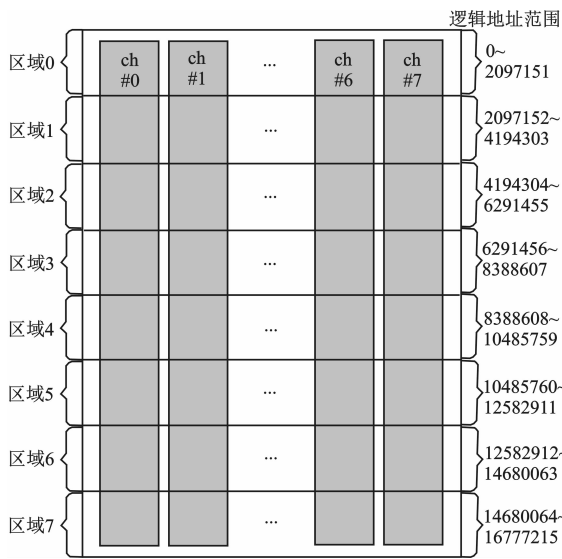


图 1 逻辑页对应区域分配示意图

在逻辑页分配的方法中,由于存在并行的优先级顺序,所以其分配方式是首先通道号递增,其次晶圆号,然后是面号,芯片号,页号,最后才是块号变化,因此,对于每个物理页,它的区域号取式(6)计算出来的块号的最高3位,逻辑块号由块号低8位、芯片号、面号、晶圆号、通道号组成,逻辑块内的偏移就等于式(5)中计算出的页号,如图2所示。这样,就把连续的逻辑页均匀的分配到固态盘的各个通道上去,并且每个逻辑页都有相应的物理页与之对应。

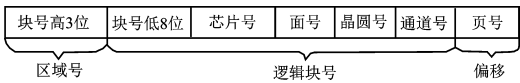


图 2 物理页对应的区域号、逻辑块号以及偏移的确定方式

3.2 地址映射方法

为完成逻辑地址到物理地址的映射,需要在内存和闪存内维持两种映射表:区域块映射表和逻辑页映射表。假设每个区域有16个逻辑块,则每个区域都有一张块映射表和16个逻辑页映射表。区域块映射表用于记载这个区域内逻辑块与物理块之间的对应关系,表内的偏移代表逻辑块号,表项内容为与之对应的物理块号。块内逻辑页映射表用于指示所分配逻辑块中各页在物理块内存储的位置。

区域页级映射确定物理页的方法流程如下:

- (1)根据逻辑页号计算出区域号、逻辑块号和偏移号;
- (2)查找对应区域块映射表,根据逻辑块号查出物理块号;
- (3)根据第(1)步中的偏移号和逻辑块号找到逻辑页映射表,找到对应的物理页号。

通过以上三个步骤的两次查表,就可以找到一个逻辑页地址所对应的物理页地址,完成一次地址映射。

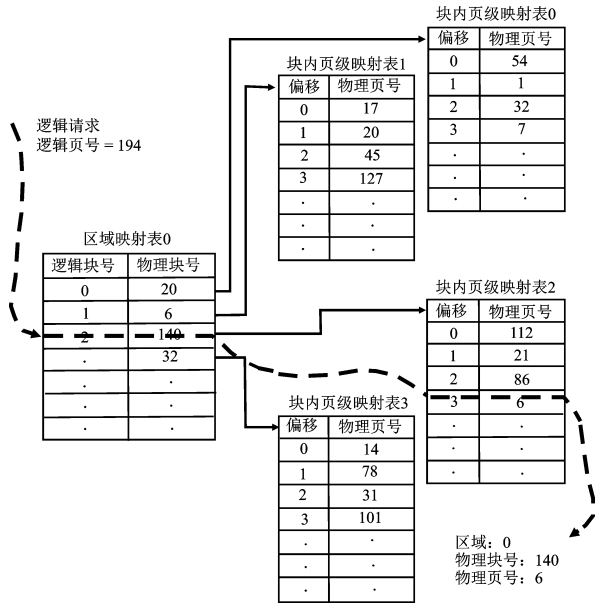


图 3 区域映射示意图

以本文所述固态硬盘为例,区域映射算法将连续的逻辑页分成8个区域,每个区域有16 384个逻辑块。每个物理块包含128个物理页。假设现在请求逻辑页194,首先得到区域号为0,则在内存中查找区域0的区域块映射表,其次根据图2中的计算方法得到其逻辑块号为2,则查找区域映射表0的第3个表项(也即逻辑块号为2的表项),如图3。查表得知逻辑块2对应的物理块号为140,进一步查找这个逻辑块的局部块内页级映射表2,根据式(5)计算出逻辑页194的偏移为3,查找局部映射表查到这个逻辑页对应的物理页号为6。这样,就完成了了一次地址映射,映射的过程中进行了两次查找映射表的操作。

3.3 损耗均衡和垃圾回收

3.3.1 损耗均衡

本算法从区域内部和区域之间这两方面,解决擦写均衡和垃圾回收的问题。

在区域内部,区域内的每一个通道都维护一个通道内部空闲块的链表,链表按照物理块被擦写的次数从小到大排列,当区域内某个物理块需要进行垃圾回收操作时,则取这个块对应区域对应通道的空闲块链表头所对应的空闲块,将被回收块的有效页数据复制到新分配的空闲块中。然后擦除被回收

块,被回收块对应的擦除次数加 1,按照最新的擦写次数,将其插入到空闲块链表中. 这样,垃圾回收时每次都选择所在通道内擦写次数最少的块来替换旧块,实现了区域内动态的损耗均衡.

在区域之间,对于固态硬盘中每个区域,都计算一个平均擦写次数 AET(Average Erase Time),计算方法即取这个区域内所有块的擦写次数的平均值,如式(7),其中 ϵ_j 为这个区域中块 j 的擦除次数, n 为区域中块的个数. 将这些区域对应的 AET 的值从大到小排序插入到链表,当表头的值与表尾 i 的值的差超过设定的值 t 时,则交换这对应的两个区域的号,使物理块相对较热的区域映射到物理块相对较冷的数据区域上. 这样,就能实现区域间的擦写均衡. 在进行冷热区域数据交换的时候,会引发大量的读写操作,影响固态硬盘的性能,因此,必须限定只有在固态硬盘处于空闲状态的时候才触发区域之间的静态损耗均衡算法. 这样通过两级擦写均衡,解决了整个固态硬盘的损耗均衡问题.

$$AET_i = (\epsilon_1 + \epsilon_2 + \cdots + \epsilon_n)/n$$

(7)

3.3.2 垃圾回收

垃圾回收需要考虑两个问题:一是什么时候进行垃圾回收,另一个是对哪些块进行垃圾回收. 对每个区域内的每个通道上维护的空闲块的链表,当链表的表项小于某一设定值,即空闲页数太少时,需要触发垃圾回收操作. 这就解决了什么时候进行垃圾回收的问题. 另一方面,垃圾回收时,将块按照无效页的数量排序,选择无效页数最多的块进行回收. 这样,在进行垃圾回收时,既能保证系统中拥有充足的空闲块,又能保证在回收时所消耗的时间和损耗最少.

4 算法的特性分析

区域映射算法采用三级映射机制,针对多通道结构的固态硬盘进行地址映射,具有以下三个特点.

(1)映射表占用小,由于采用了<区域,块,页>三级映射机制,通过查找两次映射表,区域映射表和块内页映射来得到逻辑地址,将单个映射表结构中大量重复的页表项通过分级的方式使其降到最少. 腾出了宝贵的内存资源.

(2)适用多通道的并行结构,由于区域映射是按照分散公式将连续的逻辑页分配到各个通道的不同芯片中去的,因此,对于同一个读写请求,连续的逻辑页被分配到不同通道中,这样,固态硬盘控制器就可以同时并行的将这些连续的逻辑页写入或读出,大

大减少读写介质的时间.

(3)两级损耗均衡,一方面,在区域内部,针对多通道结构,在每个通道内部进行动态的损耗均衡操作. 这样对于整个固态硬盘,其动态损耗均衡操作可以在通道间并行的执行,另一方面,考虑冷热数据的问题,在固态硬盘空闲时候进行区域之间的静态损耗均衡算法. 保证整个固态硬盘的擦写均衡. 这种既通过动态损耗均衡,又利用静态损耗均衡的两级损耗均衡算法能够保证多通道固态硬盘的寿命更加长久.

5 性能对比分析

在空间占用,性能,寿命这三个方面对页级映射,块级映射、BAST 和基于区域的页级映射做分析比较.

如表 2 所示,对于页级映射,其映射是以页为粒度,具有灵活性高,性能高的特点,固态硬盘的寿命也比较长,但是映射表占用的空间却非常大,以文中固态硬盘为例,其映射表大小为 64 MB.

表 2 不同映射算法对比

映射算法	映射表大小	响应速度	粒度	寿命
页级映射	64 MB	快	页	长
块级映射	384 kB	慢	块	短
BAST	8.375 MB	中	部分块,部分页	长
区域映射	16.25 MB	快	页	长

对于块级映射,其映射是以块为粒度,占用 RAM 空间大小仅仅只有 384 kB(每个表项 3 个字节),但粗粒度的映射使得在操作过程中存在大量的擦写、重定位数据的操作,严重影响固态硬盘的性能和寿命.

对于采用混合映射机制的 BAST,其粒度是部分块,部分页,因此其映射表占用 RAM 空间大小介于页级和块级映射之间,但是在随机写操作比较频繁的情况下,日志块的利用率非常低. 因此其相应速度也是介于它俩之间.

对于区域映射,由于采用了区域、块、页这三级映射,使得映射表的大小显著减少,同样的固态硬盘,页级映射需要 64 MB 的空间存储映射表,而区域映射只需 16.25 MB 的大小,比页级映射表的大小减少了近 70%. 而且同时,由于采用细粒度的页级映射,其性能和响应速度相对页级映射并没有减少多少,却比块级映射大大提高,同时,考虑到多通道固态硬盘的结构,区域映射对固态硬盘的读写性能也大幅

提高.

可以看到,区域页级映射算法相对于传统页级映射算法明显降低了映射表所占用的内存空间,这是区域页级映射算法的最大优势.另外,由于区域映射也是基于页级映射的,其最小的映射粒度是页级的,因此它的性能要高于以块为粒度的块级映射.由于在区域页级映射算法中直接利用了通道间的并行,其读操作性能会优于传统页级映射.

6 结束语

本文介绍了一种基于区域页级映射的闪存转换层映射机制.针对多通道并行固态硬盘的结构,提出适用并行结构的物理页分配算法,通过区域、块、页这三级映射关系完成地址映射,大幅减少了页级映射表占用内存空间太大的问题,由于考虑到固态硬盘多通道的并行结构,对固态硬盘的性能、寿命、功耗等都有很大提升.对固态硬盘技术中的损耗均衡和垃圾回收方法进行了研究,提出了两级损耗均衡的方法,通过分析表明,区域映射机制在内存空间占用、性能、寿命等方面占有优势.

参考文献:

- [1] Intel Corporation. Understanding the flash translation layer (FTL) specification[R]. Santa Clara: Intel Corporation, 1998.
- [2] Gupta A, Kim Y, Ugaonkar B. DFTL: A flash translation layer employing demand-based selective of page-level address mapping [J]. SIGPLAN Not., 2009, 44(3):229-240.
- [3] Zhou You, Wu Fei, Huang Ping, et al. An efficient page-level FTL to optimize address translation in flash memory[C]//Proceedings of the Tenth European Conference on Computer Systems. New York: ACM, 2015: 12-27.
- [4] Ban A. Flash file system optimized for page-mode flash technologies; USA, 5937425 [P]. 1999-08-10.

- [5] Kim J, Kim J M, Noh S H, et al. A space-efficient flash translation layer for compact-flash systems [J]. IEEE Transactions on Consumer Electronics, 2002, 48(2): 366-375.
- [6] Lee S W, Park D J, Chung T S, et al. A log buffer-based flash translation layer using fully-associative sector translation [J]. ACM Transactions on Embedded Computing Systems, 2007, 6(3):1-27.
- [7] Kang J U, Jo H, Kim J S, et al. A superblock-based flash translation layer for nand flash memory [C]//Proceedings of the 6th ACM & IEEE International Conference on Embedded Software. New York: ACM, 2006: 161-170.
- [8] Hu Yang, Jiang Hong, Feng Dan, et al. Exploring and exploiting the multi-level parallelism inside SSDs for improved performance and endurance [J]. IEEE Transactions on Computers, 2013, 62(6):1141-1155.
- [9] Micro. Micro MT29F64G08AKCBBH2 NAND flash memory datasheet [EB/OL]. [2015-03-18]. http://www.micron.com/-/media/Documents/Products/DataSheet/NANDFlash/60Series/M62B_16Gb_32Gb_64Gb_128Gb_AsyncSync_NAND.pdf, 2009.
- [10] Kang J, Kim J, Park C, et al. A mufti-channel architecture for high-performance and flash-based storage system[J]. Journal of Systems Architecture, 2007, 53: 644-658.

作者简介:

蔡晓乐 男,(1991-),硕士研究生,助理工程师.研究方向为计算机应用. E-mail:cxle_avic@163.com.

张亚棣 男,(1968-),博士,研究员.研究方向为计算机体系结构、嵌入式软件.

李亚晖 男,(1976-),博士,高级工程师.研究方向为嵌入式操作系统、计算机网络、信息安全.

海钰琳 女,(1991-),硕士研究生,助理工程师.研究方向为计算机应用.

(上接第 31 页)

- [6] 左森,郭晓松,万敬,等. Graph cuts 及其在立体匹配中的应用[C]//立体图象技术及其应用研讨会论文集.天津:中国图像图形学会,2005.
- [7] 李惠光,翁良宝,姜洪磊.改进的基于扩展移动的立体匹配算法[J].计算机工程与设计,2013,34(9):3180-3183.

作者简介:

马东东 男,(1990-),硕士研究生.研究方向为图像处理算法. E-mail:m104104@163.com.

杨 靓 男,(1975-),研究员,博士生导师.研究方向为空间计算机体系结构设计.

刘向增 男,(1981-),博士,工程师.研究方向计算机视觉.