Boosting the Performance of 3D Charge Trap NAND Flash with Asymmetric Feature Process Size Characteristic

Shuo-Han Chen[1], Yen-Ting Chen[1], Hsin-Wen Wei[2], Wei-Kuan Shih[1]
[1]National Tsing Hua University, Department of Computer Science, Hsinchu, Taiwan
[2]Tamkang University, Department of Electrical and Computer Engineering, Taipei, Taiwan

Abstract

The growing demands of large capacity flash-based storages have facilitated the down-scaling process of NAND flash memory. Among NAND flash technologies, 3D charge trap flash is regarded as one of the most promising candidates. Owing to the cylindrical geometry of vertical channels, the access performance of each page in one block is distinctive, and this situation is exaggerated in the 3D charge trap flash with the fast-growing number of layers. In this study, a progressive performance boosting strategy is proposed to boost the performance of 3D charge trap flash by utilizing its asymmetric page access speed feature. A series of experiments was conducted to demonstrate the capability of the proposed strategy on improving access performance of 3D charge trap flash.

Keywords 3D NAND flash, flash storage, hot/cold identification

1 Introduction

3D charge trap flash memory has been considered as an optimistic alternative to providing a larger scale flash memory with low cost. Constructing 3D charge trap flash involves stacking multiple gate stack layers and punching vertical channels through the stacked layer so as to create cylinder-shape charge traps for storing bits. Owing to the erosion process of creating vertical channels, the feature process size is different throughout vertical channels. Due to the asymmetric feature process size, the access speed of each page in one block is distinctive. As the number of gate stack layers grows, the last page of one block could be much faster than the first page of the same block. To exploit this unique asymmetric page access speed feature, this study proposes a progressive performance boosting strategy to progressively store data of different hotness to the pages with suitable access speed so as to increase the overall access performance. However, since one block has both fast and slow pages, residing both hot/cold data to fast/slow pages of a single block could hinder the performance of garbage collection process. Thus, the technical difficulty lays in *how to boost the performance of 3D charge trap flash by exploiting the asymmetric page access speed without affecting the efficiency of garbage collection process.*

Flash memory has many attractive features, such as high access speed and low power consumption. However, it also has several constraints, including the erase-before-write property and the limited number of program/erase (P/E) cycles. Due to the erase-before-write property, flash memory cell cannot be overwritten unless it is erased first. Therefore, flash translation layer (FTL), such as FTL [3] and FAST [10], are mainly designed to conduct out-of-place update to redirect write requests to an free area and to avoid the long latency owing to the erase-before-write property. On the other hand, a flash memory chip contains a lot of blocks, it units of read/write and erase operations are different. A block, which is the basic unit of erase operations, contains a fixed number of pages and pages are the unit of read/write operations. Due to the unit size difference, the garbage collection (GC) mechanisms, designed to reclaim out-of-date pages, must be carefully planned to minimize costs of copying valid (or live) pages while erasing blocks.

To minimize costs, various data hot/cold identification mechanisms were proposed to collect hot data into the same blocks as much as possible because hot data tends to be frequently updated and become invalid soon while cold data are rarely changed. Thus, the number of copying live pages during GC operations can be lowered. Researchers have proposed numbers of excellent methods for effectively identifying hot/cold data, such as request size-based prediction strategy [1], two-level LRU scheme [2], table-based hot/cold history management scheme [5], and compression-based identification scheme [7]. However, these hot/cold data identification mechanisms only focus on either improving the efficiency of GC operations or wear-leveling mechanisms. To the best of our knowledge, none or few FTL design or hot/cold data identification solution has exploited the page access speed difference feature of 3D charge trap flash to boost the storage access performance.

To fully exploit the potential benefits of the asymmetric page access speed feature of 3D charge trap flash, this study proposes the first systemic solution, called progressive performance booster (PPB) strategy, to progressively boost the access performance of 3D charge trap flash by storing data of different hotness in pages with appropriate access speed. To exploit benefits of the asymmetric page access speed feature, the proposed PPB strategy introduces (1) a new four-level data hot/cold identification, (2) a virtual block concept, and (3) a hot/cold data area design. The four-level data hot/cold identification progressively identifies data hotness based on the re-access frequency. Thus, both hot data can cold data can be further divided into two categories to form the four-level data hotness. After that, the virtual block concept will group the pages that have similar access speed to split a physical block into virtual blocks of different speeds. The PPB strategy then progressively move data to the pages with suitable speed based on the hotness identification results. Finally, the PPB strategy tracks the usage of virtual blocks by the mechanism maintained in the hot/cold area. With the proposed PPB strategy, the read performance of 3D charge trap based flash storage can be improved by 18.56% in the experiments, compared to the conventional FTL design without the PPB strategy.

The rest of paper is organized as follows. Section 2 shows the background of 3D charge trap flash and research motivation, and Section 3 describes the design and mechanism of the proposed progressive performance boosting strategy. The performance of the proposed strategy is then evaluated with trace-driven experiments, which are presented in Section 4. Section 5 concludes this study and the research remarks.

2 Background and Motivation

2.1 Background of 3D Charge Trap NAND Flash

As the demand of high-density and low-cost non-volatile memories continues to grow in the storage market, the 3D NAND technology is gathering increasing attention as a future high-density memory technology to increase bit density and cost effectiveness. However, the scaling of traditional planar floating-gate flash memory faces several challenges such as poor endurance, power consumption, insufficient programming/erasing efficiency, and interference coupling issue. In addition, according to the previous investigation [18], the tunnel oxide thickness of floating-gate cell must be more than 6-nm to prevent charge leakage and assure enough retention time. Therefore, scaling down traditional floating-gate cell without affecting the retention and endurance characteristics becomes much more challenging. To overcome these issues, charge trap memories, such as SONOS and

This work is supported by the Ministry of Science and Technology, R.O.C., under 105-2622-8-009-008, 105-2634-F-007-001 and 105-1221-E-032-031-MY2.

DAC '17, Austin, TX, USA

DOI: http://dx.doi.org/10.1145/3061639.3062209

TANOS, have been proposed [6, 11, 14, 16] and regarded as the mainstream candidate for the next generation 3D NAND flash memory technology.

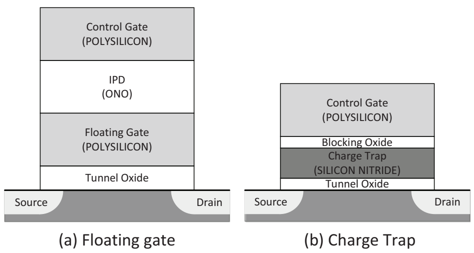

Figure 1. Comparison of floating-gate and charge trap transistor structures [12].

As the comparison shown in Figure 1, the concept of charge trap memory cell is composed of a metal oxide semiconductor device where the floating gate is replaced by a charge trap device, which is typically made of silicon nitride. The charge trap device can hold charges and prevent charges from moving freely; therefore, charge trap devices have better endurance than the conventional floating-gate devices. The 3D charge trap flash can be achieved in two different ways. The simplest method is to stack multiple 2D planar arrays to construct a 3D structure. However, this stacking approach does not improve P/E cycle or retention, compared to the planar charge trap cells. Another method is known as "vertical channel", which builds charge trap cells with cylindrical channels. Based on vertical channel, several 3D charge trap flash architectures have been proposed, such as BiCS [16] and TCAT [6]. As illustrated in Figure 2(a), the structure of vertical channel 3D charge trap flash involves several gate stack layers and vertical cylindrical channels. To create vertical cylindrical channels, manufacturers apply liquid chemicals to erode the gate stack layers. Due to the physical characteristic of liquid, the eroded cylindrical channel will have a larger opening at the top layer and a smaller opening at the bottom layer. This physical phenomenon results in asymmetric feature process size across the gate stack layers. After the erosion process, each cylindrical channel is filled with charge trap materials, as shown in Figure 2(b) to store bits at each gate stack layer.

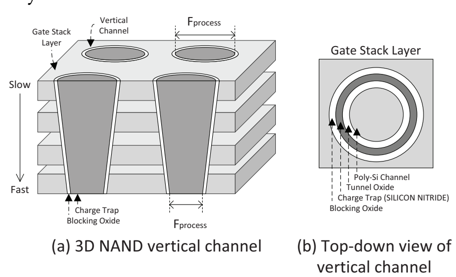

Figure 2. Vertical channels of 3D charge trap flash [4, 8].

The asymmetric feature process size of vertical channels results in the different strength of the electric field at each gate stack layer. The smaller the opening is, the stronger the electric field will be [9]. Therefore, accessing bits stored at the bottom layer will be faster than the top layer. As the number of gate stacked layers grows, the access speed at the bottom layer could be multiple times [1] faster than the top layer. In the design of Flash Translation Layer (FTL), vertical channels are mapped as blocks and channel sections located at each gate stack layer are mapped as pages. Thus, pages within the same block have inconsistent access speed due to the unique cylindrical shape of vertical channels.

The irregular page access speed feature of 3D charge trap flash can be exploited to enhance the performance of flash-based storage. One intuitive idea is to store hot data in pages with faster access speed and cold data in pages with slower access speed because hot data tends to be frequently updated or accessed while cold data is rarely changed. Serving hot data requests with faster pages can indeed improve the overall performance of the flash-based storage. However, *placing both hot data and cold data within a block will lead to tremendous overhead when recycling invalid space during garbage collection process.* Thus, the major technical problem in exploiting the inconsistent access speed feature is how to maximize the system performance without sacrificing the garbage collection efficiency.

2.2 Motivation

Due to the unique cylindrical shape of 3D charge trap flash, pages within a single block have different access speed. The access speed difference could be used to boost the performance of flash-based storage devices. Unfortunately, current FTL designs, such as FTL [3] and FAST [10], assume all pages have the same access speed. Consequently, current FTL designs cannot fully exploit the benefit of irregular page access speed feature in 3D charge trap flash. To boost the performance of 3D charge trap flash, applying existing hot/cold data identification mechanisms to store hot and cold data in the fast and slow pages respectively becomes a viable option. However, placing both hot data and cold data within a single block could be harmful to the efficiency of garbage collection because hot data tends to be frequently updated and becomes invalid while cold data is rarely updated and usually remains valid. Eventually, when the garbage collection is triggered to reclaim invalid space, most of the blocks become half-valid and half-invalid, as shown in Figure 3. This situation prevents the garbage collection mechanism from selecting an appropriate block to minimize the overhead of copying live pages.

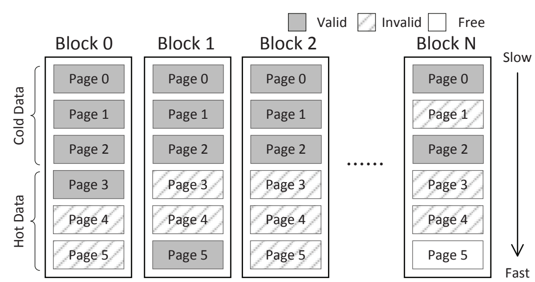

Figure 3. Issue of applying conventional hot/cold identification.

With the concern of degrading the garbage collection efficiency, applying existing hot/cold data identification to exploit the page access speed difference is not applicable. *Therefore, the technical difficulty lays on how to take page access speed difference into the design of flash management to boost the storage system performance without affecting the garbage collection performance.* To resolve this issue, we propose a progressive performance boosting (PPB) strategy to identify and place data with different hotness to appropriate locations to improve the access performance of flash-based storage systems. The details of the proposed strategy are described in Section 3.

3 Progressive Performance Boosting Strategy

3.1 Overview

To exploit the potential benefits brought by the page access speed difference, this study presents the progressive performance boosting (PPB) strategy to increase the access performance of 3D charge trap flash devices. *To the best of our knowledge, this is the first system design on investigating the asymmetric page access speed phenomenon.* The goal of the proposed strategy is to gradually place data of different hotness to pages with suitable access speed without affecting the performance of garbage collection process. Therefore, the overall access performance can be enhanced by serving frequent-accessed hot data with fast pages. To achieve this goal, the proposed PPB strategy introduces the four levels hot/cold data identification, the concept of virtual block, and a hot/cold data area design. Figure 4 shows the system architecture of the proposed strategy.

[1]The access speed of the bottom layer is typically 2x to 5x faster than the top layer. Currently, the access speed difference of 64 layers 3D charge trap flash is within 2x.

As shown in Figure 4, the proposed PPB strategy focuses on placing hot/cold data to pages with suitable access speed so as to improve the overall access performance of 3D charge trap flash memory. Since placing hot/cold data with a block could hinder the performance of flash-based storage, the PPB strategy further classifies storage data into four hotness level, including *iron-hot*, *hot*, *cold* and *icy-cold* (see Section 3.2) and progressively store each type of data to pages with suitable access speed. Therefore, the PPB strategy could put hot/iron-hot data or cold/icy-cold data to fast/slow pages of one block without hinder the garbage collection performance. Furthermore, instead of proposing a new hot/cold data identification mechanisms, the four level hot/cold identification is achieved based on existing identification mechanisms to preserve the decades worth of work on data hotness identification. Therefore, the proposed strategy is compatible with any hot/cold data identification mechanisms. In the following sections, we refer to the size check method as a case study.

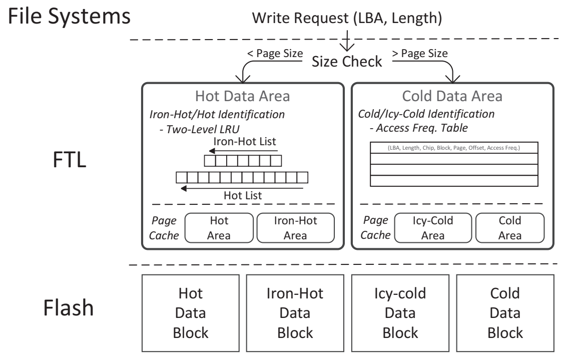

Figure 4. The architecture of PPB Strategy.

On the other hand, to achieve the elaborate data placement design for the four different data hotness, the proposed PPB strategy introduce the concept of *virtual block* to group pages with similar access speed. With the introduction of virtual block, the original block lifecycle and allocation mechanism needs to be changed to consider the speed difference of virtual blocks (see Sections 3.3). Finally, after initially identifying the data hotness, hot/cold data are diverted to the *hot/cold data area* respectively for recording the data access pattern. In the hot data area, a two-level LRU list is used to track to further identify data hotness and gradually move data to pages with suitable access speed. On the other hand, an access frequency table is also included in the cold data area to record and serve data with appropriate pages (see Section 3.4). With the designed components, the PPB strategy is the first solution that investigates and exploits the benefit of the asymmetric page access speed feature of 3D charge trap flash memory.

3.2 Four-level Hot/Cold data Identification

To exploit the page access speed difference of 3D charge trap flash, the most simple solution is to reside hot/cold data directly in the fast/slow pages respectively. However, due to the fact blocks have both fast and slow pages, putting hot/cold data in one single block will degrade the performance of garbage collection process. In the end, the overall performance might be worse than original data placement without the identification process.

To resolve the described issue, the concept of four level hot/cold data identification is introduced to further categorize hot/cold data into four hotness level so as to exploit the benefit of the page access speed difference. Based on the frequency of read operations, the hot data is further classified into *iron-hot data* and *hot data*. Iron-hot data refers to those data that are frequently read/updated, such as file system metadata. On the other hand, hot data are those data that are frequently updated but receive less read operation, such as temporary cache files. On the other hand, cold data are divided into *cold data* and *icy-cold data*. Cold data are those write-once-read-many data, including videos and pictures, while icy-cold are those write-once-read-few data, such as backup files. With the four level hot/cold data identification, the PPB strategy could put iron-hot/hot data or cold/icy-cold data to the fast/slow pages of one block without hinder the garbage collection performance. For better understandings of the data placement with different hot level, Figure 5 presents an example of placing data in pages with different access speed.

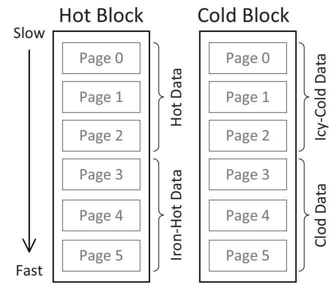

Figure 5. Four levels of hot/cold data.

As Figure 5 shows, data blocks are classified into hot blocks and cold blocks to avoid putting hot/cold data in one single block. The slow pages of both hot/cold blocks are used to store data with less-frequent read operations, which are hot data and icy-cold data. On the other hand, fast pages are used to serve iron-hot and cold data for boosting the access performance. *In addition, instead of trying to store data to the suitable pages at first place, the PPB strategy aims to progressively move data to the suitable pages.* Therefore, after the first stage hot/cold data identification, hot/cold data are firstly stored at hot data area and icy-cold data area respectively. Next, as shown in Figure 6, the PPB strategy gradually move data to the pages with suitable access speed during garbage collection or data update process.

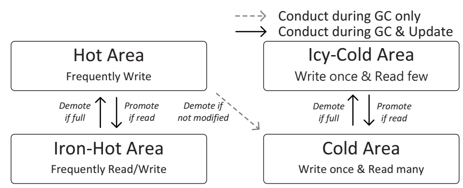

Figure 6. Data movement between four hot/cold data level.

To facilitate the process of storing data to pages with suitable speed, the proposed PPB strategy also introduces the concept of virtual block to group pages with adjacent access speed for storing data of different hot level (see Section 3.3.1).

3.3 The Concept of Virtual Block

To support the proposed PPB strategy and four level hot/cold data identification, the virtual block (see Sections 3.3.1) are introduced to store data of different hot level in pages with suitable access speed. In addition, the allocation mechanism (see Sections 3.3.2) and lifecycle (see Sections 3.3.3) of physical blocks are also redefined by the concept of virtual block.

3.3.1 Virtual Blocks with Different Access Speed

Since traditional FTL designs do not consider the page access speed difference of 3D charge trap flash, pages of different access performance cannot be assigned effectively to store data of different hotness level. To resolve this issue, the PPB strategy includes the concept of virtual block to divide physical blocks into virtual blocks. The concept of virtual block is to group pages with adjacent access speed for storing the data of different hot level. Figure 7 shows an example on how to group pages to form virtual blocks.

As Figure 7 shows, pages are grouped together based on their access speed. In this example, each physical block is divided into two virtual blocks to store data of different hot level. For instance, the physical block N is divided into virtual block N and virtual block 2N+1, each of which consists of slow and fast pages respectively. Therefore, after a block N is allocated as a hot block, the virtual block 2N can be used to store hot data with less read operation while the virtual block 2N+1 can be used to serve frequently-read iron-hot data to boost the performance of access iron-hot data. On the other hand, when a block N is assigned as a cold block, the virtual blocks 2N and

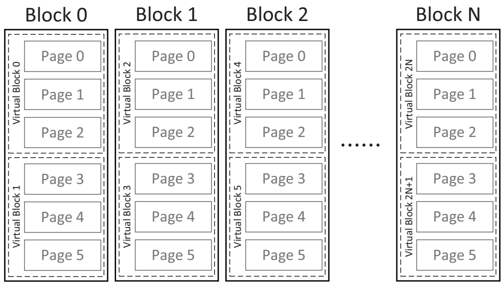

Figure 7. The Concept of Virtual Block.

2N+1 are used to store icy-cold and cold data respectively for improving the performance of accessing cold data. Note that a physical block can be divided into multiple virtual blocks rather than two; however, the performance enhancement and the overhead of maintaining the virtual blocks should be balanced. With the introduction of virtual block concept, a new allocation mechanism is needed to support allocating pages of different access speed under the constraints of the original physical block allocation mechanism (see Section 3.3.2).

3.3.2 Virtual Block Allocation

To enable allocating pages of different speed, the PPB strategy divides physical blocks into virtual blocks based on the page access speed. To facilitate the management of virtual blocks, five virtual block (VB) lists are introduced, including the free, hot, iron-hot, icy-cold and cold VB lists. The free list is used to track free virtual blocks, and virtual blocks are arranged according to their original physical block number. To avoid putting hot and cold data within a single physical block, virtual blocks of the same physical block can only be allocated to either hot or cold area. For instance, as shown in the steps 1 and 2 of Figure 8, when the virtual block 0 with slow access speed is assigned to the hot VB list, the virtual block 1 with fast access speed can only be allocated by the iron-hot VB list. On the other hand, since pages of a physical block can only be written from the beginning of a physical block, the latter virtual block cannot be written until the former virtual block is written. As shown in the steps 3 and 4 of Figure 8, the virtual block 3 cannot be assigned to the iron-hot list until the virtual block 2 is fully used.

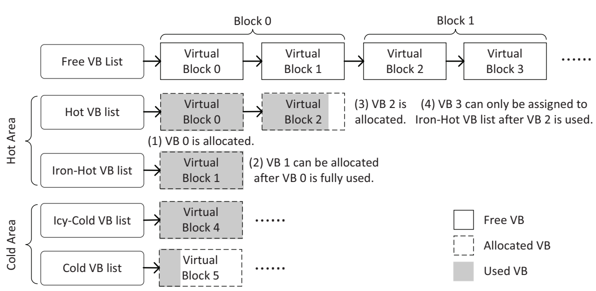

Figure 8. Virtual Block Allocation List.

3.3.3 Lifecycle of Virtual Blocks

Due to page writing order constraint, virtual blocks cannot be randomly allocated. Therefore, virtual blocks need to follow a predefined lifecycle to comply with the writing order constraint. The lifecycle of virtual blocks is illustrated in Figure 9.

As shown in Figure 9, the physical block n is divided into virtual block 2n with slow access speed and virtual block 2n+1 with fast access speed based on the concept of virtual block. When both virtual blocks are free, the virtual block 2n can be allocated to either hot or icy-cold list to store data with less re-access frequency. After virtual block 2n is allocated, the virtual block 2n+1 can only be allocated after virtual block 2n is fully used due to the writing order constraint. Then, the virtual block 2n+1 can be allocated to the iron-hot or cold list for storing frequently re-access data. However, to avoid coexisting hot and cold data in one physical block, both virtual blocks must

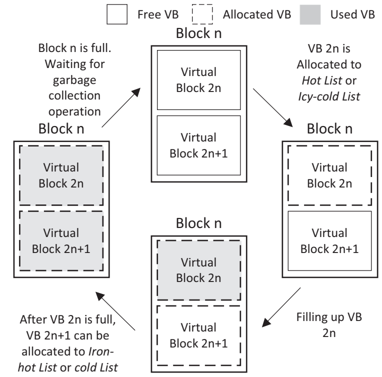

Figure 9. Virtual Block Lifecycle.

be allocated by the same area. Finally, when both virtual blocks are fully used, the block n waits for garbage collection operation to reclaim invalid space.

3.4 Operations of Hot/Cold Area

To exploit the page access speed difference and avoid residing hot/cold data within a single block, the PPB strategy further categorizes hot/cold data into four different hotness level. The PPB strategy first identifies hot/cold data based on previous excellent hot/cold identification mechanisms. Next, entries are diverted to the hot area or cold area of PPB strategy for recording the re-access frequency.

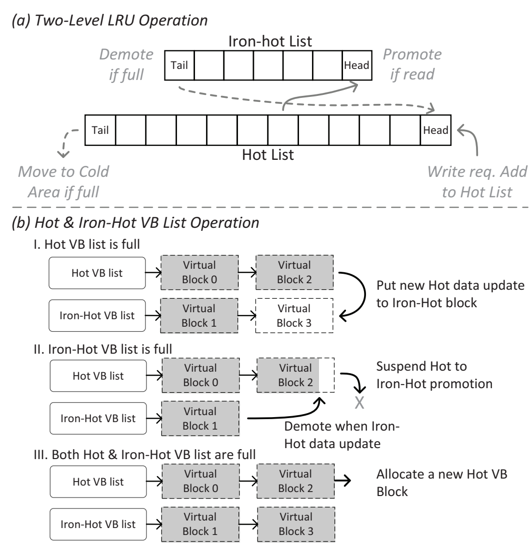

Figure 10. Hot Area Operation.

To further categorize hot data based on the four-level identification, the PPB strategy includes a two-level LRU to track the re-access frequency of iron-hot and hot data chunks. The two-level LRU list is used for its simplicity because hot data is typically re-accessed frequently. Therefore, a complex mechanism could hinder the access performance. As shown in Figure 10 (a), when a new write request is diverted into the hot area, the PPB strategy put the new data chunk to the head of the hot list. If a data chunk in the hot list receives read requests, the data chunk is then promoted to the iron-hot list. However, the corresponding data chunk is not immediately moved from hot VB to iron-hot VB. Instead, *the PPB strategy progressive moves data to its new location when updating or conducting garbage operations.* Therefore, the PPB strategy could boost the access performance without inducing additional garbage collection overhead. On the other hand, when either the iron-hot list is full or when there is no free space left in the iron-hot VB list, the least-recently-used entry is demoted to the head of the hot list. Similar operations also apply to the hot list for demoting entries to the cold area.

In addition to tracking the re-access frequency of hot data, the write operations to virtual blocks also needs to comply with the page writing order constraints of flash devices. In addition, since fast and slow virtual blocks belong to a single physical block, if either fast or slow virtual blocks are allocated excessively, the physical blocks could become half-full and half-empty. Therefore, the space utilization of physical block could be degraded. To resolve above issues, special allocation mechanisms are included. In the hot area, iron-hot and hot VB list are used to track the usage of the two virtual blocks with different access speed, as shown in Figure 10 (b). To prevent excessively allocation of either fast or slow virtual blocks, new virtual blocks can only be allocated by hot area when both hot and iron-hot lists are full, as shown in Figure 10 (b) III. On the other hand, as shown in Figure 10 (b) I and II, if one of the lists is full while the other still has some free space, write or update requests are diverted to the other list to prevent degrading space utilization. The operations are briefly summarized in Algorithm 1.

Algorithm 1: Hot/Iron-hot VB List Operation.

```
Input: LBA, length, requestVBType, areaType
Output:
if areaType is Hot Area then
    Check Iron-hot and Hot list for duplicated LBA;
    if LBA is found then
        Invalidate original data chunk;
        Remove the duplicated LBA entry;
    end
    if requestVBType is Iron-hot VB then
        if Both Iron-hot and Hot VB list has no free space then
            Allocate new VB to Hot VB list;
            Divert write request to Hot VB list;
        else if Iron-hot list has no free space then
            Divert write request to Hot VB list;
        end
        Store the write request to free space of Iron-hot VB list;
        Insert new LBA entry to Iron-hot list;
    else if requestVBType is Hot VB then
        if Hot list has no free space then
            Divert write request to Iron-Hot VB list;
        else if Both Iron-hot and Hot VB list has no free space then
            Allocate new VB to Hot VB list;
        end
        Store the write request to free space of Hot VB list;
        Insert new LBA entry to Iron-hot list;
    end
end
return;
```

Similar to the hot area, the PPB strategy also separates cold data into two categories based on the re-access frequency. Cold data are stored in virtual blocks of fast access speed to increase re-access performance, while icy-cold data are stored in virtual blocks of slower speed since icy-cold are rarely accessed. As shown in Figure 11 (a), an access frequency table is included in the cold area to log the access frequency for each chunk of data to record the re-access frequency of cold and icy-cold data. Then, the table is sorted based on the logged access frequency. Finally, the PPB strategy moves the data to its new location with suitable access speed. On the other hand, due to the constraints of page writing order and avoiding degrading space utilization, allocation constraints are required for allocating virtual blocks to the cold and icy-cold VB lists, as illustrated Figure 11 (b). Similar to operations of hot and iron-hot VB lists, new virtual blocks can only be allocated when both cold and icy-cold list are full. If either list still has free space, the free space should be used to serve write or update requests regardless of the data hotness level. The operation is similar to the process summarized for the hot area in Algorithm 1.

4 Performance Evaluation

4.1 Experiment Setup

In this session, experiments are conducted to evaluate the capability of the proposed PPB strategy regarding the read/write access latency and block erase count. Note that this study mainly focuses on boosting the access performance of 3D charge trap flash memory with its

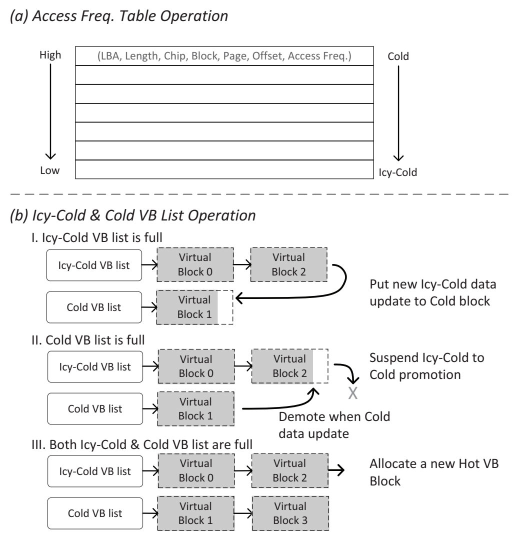

Figure 11. Cold Area Operation.

asymmetric page access speed features. Therefore, this session focuses on evaluating the enhanced performance of the proposed strategy instead of endurance because many excellent wear-leveling designs can be easily integrated into the flash architecture to extend its lifetime. The proposed PPB strategy is integrated into the conventional FTL design, and all experiments were conducted on a flash simulator with the two traces collected by Microsoft Research Cambridge [13, 17] from enterprise servers. The experimental settings of I/O latency are summarized in Table 1, which are set according to the specification of 3D NAND manufactured by Samsung [15]. To investigate the capability of PPB strategy under different page access speed difference, experiments are conducted on pages with different access speed difference, ranging from 2x to 5x. In addition, due to the trend of growing page size, experiments are also conducted on pages with different size, including 8KB and 16KB, to understand the impact of growing page size on the proposed PPB strategy. Furthermore, since PPB strategy progressively moves data to a new location to improve the access performance, the number of erased blocks could be increased because of this data movement. Therefore, additional experiments are also conducted to track the number of erased blocks for both traces.

Table 1. Experimental Paramaters [15].

Item	Specification
Flash size	64GBs
Page size	16KBs
Number of pages per block	384
Page write latency (μs)	600
Page read latency (μs)	49
Data transfer rate	533Mbps
Block erase time (ms)	4

4.2 Experimental Results on I/O Performance

For evaluating the capability of the proposed PPB strategy on improving the access performance of the 3D charge trap flash memory, real-world traces are used to conduct the experiments. Figures 12 and 15 show the read and write performance enhancement, compared to conventional FTL design, of two different page sizes respectively after running the media server and web server traces. As the results show, the enhancement achieved by PPB strategy could reach 18.56% with the 16KB page size after running the web server trace. From the result, we can see that the proposed PPB strategy can hugely improve the read performance while maintaining almost identical write performance. This is because the PPB strategy moves frequently reaccessed data to pages with faster access speed, In addition, the PPB strategy only moves data to its new location during update or garbage collection operations. Therefore, the write performance is not degraded due to the data movement. On the other hand, the results

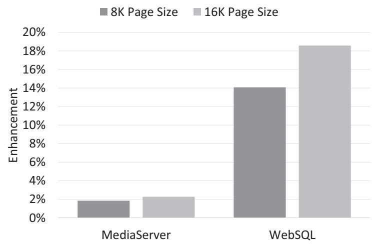

Figure 12. Read Performance Enhancement.

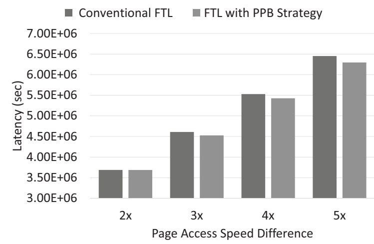

Figure 13. Media Server Trace : Read Latency Comparison.

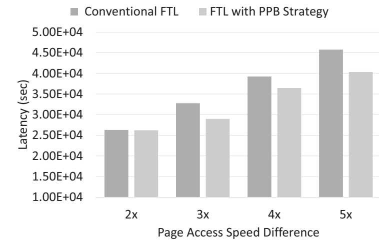

Figure 14. Web Server Trace : Read Latency Comparison.

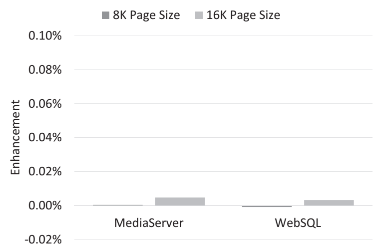

Figure 15. Write Performance Enhancement.

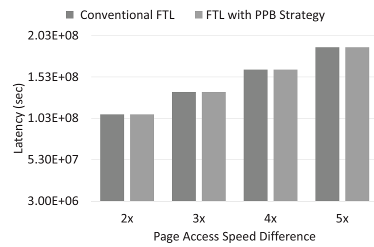

Figure 16. Media Server Trace : Write Latency Comparison.

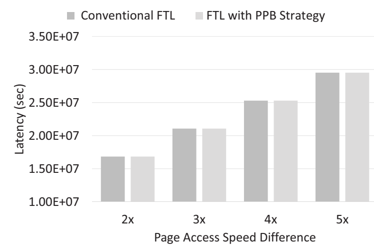

Figure 17. Web Server Trace : Write Latency Comparison.

show that the PPB strategy performs better when the page size grows because the flash memory could provide larger space with fast access speed when the page size grows. Therefore, the PPB strategy can achieve better performance enhancement.

To investigate the effectiveness of the proposed strategy on pages with different access speed difference, experiments are conducted based on 2x, 3x, 4x, and 5x page speed difference. The results after running the media server and web server trace are illustrated in Figures 13, 14, 16, and 17. Results show that the read latency of the PPB strategy is smaller than the conventional FTL design by 10% on average for the four access speed difference while maintaining almost identical write latency and the difference is only 0.0001%. Besides, the number of erase block count is not increased excessively by the proposed PPB strategy, as shown in Figure 18. Therefore, the garbage collection performance of original FTL design is retained.

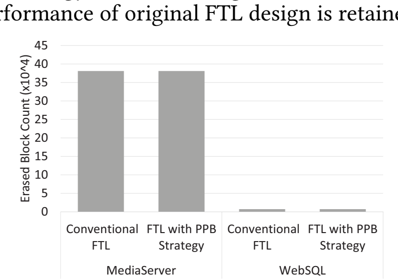

Figure 18. Erased Block Count Comparison.

5 Conclusion

To exploit the potential benefit bought by asymmetric page access speed feature, this study presents the first systemic design, called the progressive performance boosting (PPB) strategy to boost access performance of 3D charge trap flash memory. The proposed PPB strategy introducing the four-level hot/cold data identification to further categorize hot/cold data into iron-hot/hot data and cold/icy-cold data respectively based on the data re-access frequency. In addition, to store data of different hot level at pages with suitable access speed, the concept of virtual block is included to divide blocks into virtual blocks by grouping pages of adjacent access speed. Therefore, pages of different access speed can be allocated effectively under PPB strategy. To maximize the effectiveness of storing data in suitable pages, a new allocation mechanism and a hot/cold data area design are included in the PPB strategy to manage virtual block allocation and to log the re-access frequency of data. The read performance of 3D charge trap flash memory with the proposed strategy is improved 18.56% without additional overhead on garbage collection or write operation.

References

[1] L.-P. Chang. Hybrid solid-state disks: Combining heterogeneous nand flash in large ssds. In *2008 Asia and South Pacific Design Automation Conference*, March 2008.

[2] L.-P. Chang and T.-W. Kuo. An adaptive striping architecture for flash memory storage systems of embedded systems. In *Real-Time and Embedded Technology and Applications Symposium, 2002. Proceedings. Eighth IEEE*, 2002.

[3] P. C. EHLINGER, JR. Flash file system, 4 1995.

[4] A. Goda and K. Parat. Scaling directions for 2d and 3d nand cells. In *Electron Devices Meeting (IEDM), 2012 IEEE International*, Dec 2012.

[5] J.-W. Hsieh, L.-P. Chang, and T.-W. Kuo. Efficient on-line identification of hot data for flash-memory management. In *Proceedings of the 2005 ACM Symposium on Applied Computing*, pages 838–842, 2005.

[6] J. Jang, H. S. Kim, W. Cho, H. Cho, J. Kim, S. I. Shim, Younggoan, J. H. Jeong, B. K. Son, D. W. Kim, Kihyun, J. J. Shim, J. S. Lim, K. H. Kim, S. Y. Yi, J. Y. Lim, D. Chung, H. C. Moon, S. Hwang, J. W. Lee, Y. H. Son, U. I. Chung, and W. S. Lee. Vertical cell array using tcat(terabit cell array transistor) technology for ultra high density nand flash memory. In *2009 Symposium on VLSI Technology*, June 2009.

[7] K. Kim, S. Jung, and Y. H. Song. Compression ratio based hot/cold data identification for flash memory. In *Consumer Electronics (ICCE), 2011 IEEE International Conference on*, Jan 2011.

[8] Y. Kim, R. Mateescu, S.-H. Song, Z. Bandic, and B. V. K. V. Kumar. Coding scheme for 3d vertical flash memory. In *IEEE International Conference on Communications (ICC)*, 2015.

[9] J. H. Lee, G. S. Lee, S. Cho, J.-G. Yun, and B.-G. Park. Investigation of field concentration effects in arch gate silicon–oxide–nitride–oxide–silicon flash memory. In *Japanese Journal of Applied Physics*, 2010.

[10] S.-W. Lee, D.-J. Park, T.-S. Chung, D.-H. Lee, S. Park, and H.-J. Song. A log buffer-based flash translation layer using fully-associative sector translation. In *ACM Transactions on Embedded Computing Systems (TECS)*, 2007.

[11] F. R. Libsch and M. H. White. Charge transport and storage of low programming voltage sonos/monos memory devices. In *Solid-State Electron*, 1990.

[12] R. Micheloni. *3D Flash Memories*. Springer Nature, 2016.

[13] D. Narayanan, A. Donnelly, and A. Rowstron. Write off-loading: Practical power management for enterprise storage. In *ACM Transactions on Storage (TOS)*, 2008.

[14] K. T. Park, J. m. Han, D. Kim, S. Nam, K. Choi, M. S. Kim, P. Kwak, D. Lee, Y. H. Choi, K. M. Kang, M. H. Choi, D. H. Kwak, H. w. Park, S. w. Shim, H. J. Yoon, D. Kim, S. w. Park, K. Lee, K. Ko, D. K. Shim, Y. L. Ahn, J. Park, J. Ryu, D. Kim, K. Yun, J. Kwon, S. Shin, D. Youn, W. T. Kim, T. Kim, S. J. Kim, S. Seo, H. G. Kim, D. S. Byeon, H. J. Yang, M. Kim, M. S. Kim, J. Yeon, J. Jang, H. S. Kim, W. Lee, D. Song, S. Lee, K. H. Kyung, and J. H. Choi. 19.5 three-dimensional 128gb mlc vertical nand flash-memory with 24-wl stacked layers and 50mb/s high-speed programming. In *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, Feb 2014.

[15] Samsung. Samsung v-nand@ONLINE, http://www.samsung.com/semiconductor/products/flash-storage/v-nand/, 2015.

[16] H. Tanaka, M. Kido, K. Yahashi, M. Oomura, R. Katsumata, M. Kito, Y. Fukuzumi, M. Sato, Y. Nagata, Y. Matsuoka, Y. Iwata, H. Aochi, and A. Nitayama. Bit cost scalable technology with punch and plug process for ultra high density flash memory. In *2007 IEEE Symposium on VLSI Technology*, June 2007.

[17] A. Traeger, E. Zadok, N. Joukov, and C. P. Wright. A nine year study of file system and storage benchmarking. In *ACM Transactions on Storage (TOS)*, 2008.

[18] C. Zhao, C. Z. Zhao, S. Taylor, and P. R. Chalker. Review on non-volatile memory with high-k dielectrics: Flash for generation beyond 32 nm. In *Materials*, 2014.