Process Variation Aware Read Performance Improvement for LDPC-Based NAND Flash Memory

Qiao Li, Liang Shi, Yejia Di, Congming Gao, Cheng Ji, Yu Liang, and Chun Jason Xue

***Abstract*—With the rapid development of technology scaling and cell density improvement for capacity increase and cost reduction, NAND flash memory is confronted with degraded reliability. On one hand, while low-density parity-check (LDPC) codes have been deployed in today's NAND flash memories to enhance reliability, flash read latency has still been a performance bottleneck with the increased raw bit error rates (RBER). On the other hand, significant process variations (PV) have been found on existing NAND flash memories, which introduce great reliability variations among different flash blocks. Recent studies have proposed to exploit PV to improve endurance by better wear leveling or to improve write performance. These approaches are prone to allocate read data to blocks with low reliability, which further degrades read performance. This paper proposes to enhance read performance of LDPC-equipped NAND flash memory by exploiting the reliability variations from PV. The paper consists of three parts. First, a block grouping approach is presented to categorize flash blocks according to their reliability. Second, according to the grouping scheme, a data placement scheme is proposed, which allocates read-hot data to flash blocks with high reliability. At the same time, the read-cold data is moved to blocks with low reliability. As a result, the read performance is enhanced. However, allocating high reliable blocks for read-hot data collides with previous PV-based wear leveling methods. To address the issue, the third part is a grouping partition scheme which limits the amount of high reliable blocks occupied by read-hot data. Therefore, read performance enhancement can be achieved and the wear leveling schemes will be impacted slightly. Experiment results present that, the proposed approach can provide significant read performance improvement on LDPC-equipped NAND flash memory and is compatible with the previous PV-based wear leveling.**

***Index Terms*—Low-density parity-check (LDPC) codes, NAND flash memory, process variation (PV), read performance.**

I. Introduction

NAND flash memory has been widely adopted as storage devices in mobile devices, embedded systems, and data centers [2], [3]. The rapid development in cell density improvement and technology size scaling has lowered its cost. Besides, the adoption of three-dimensional (3-D) NAND flash memory has further developed the storage capacity by stacking flash cells vertically [2], [4], [5]. However, the down sides of the development of NAND flash memory are reliability degradation and great process variations (PV) [6]–[10]. To guarantee the correctness of read data, error correction codes (ECC) with a strong capability to correct data with high raw bit error rates (RBER) are required. With strong error correction capability through soft decoding, low-density parity-check (LDPC) codes have been applied for the state-of-the-art NAND flash memories [7], [8], [11]–[13]. LDPC is designed to decode data by iteratively sensing the data until the data is correctly decoded. However, on one hand, LDPC needs long read latency and high sensing power consumptions to correct data. These drawbacks are much severe especially when the data to be accessed have high RBER [7], [11], [12]. On the other hand, these drawbacks are further amplified by recent works on exploiting PV for write performance improvement or lifetime improvement on NAND flash memory [9], [10], [14]. PV has been acknowledged in recent NAND flash memories, which presents great reliability variations among flash blocks. Recent PV-aware wear leveling and write performance improvement schemes introduced allocate write-hot data to high reliable blocks. As a result, read-hot data are prone to be allocated to low reliable blocks, which leads to degraded read performance.

Improving read performance is critical for storage systems because reads are always on the critical path of the systems. To boost the read performance of NAND flash memory, several approaches have been introduced [7], [8], [11], [12], [15]. Read latency on LDPC-based NAND flash memory is closely related with the RBER of read data. For data with high RBER, the data are sensed with more reference voltages to obtain the soft information for stronger error correction capability. In this case, the read latency will be increased. Previous work introduced attempts to decrease read latency by reducing the number

Manuscript received July 2, 2018; accepted November 22, 2018. This work was supported by the NSFC under Grant 61772092 and Grant 61572411. This paper was presented in part in the Proceedings of the 35th IEEE International Conference on Computer Design (ICCD'17)[1]. The conference paper covers only the data placement with awareness of read hotness, whereas the new manuscript addresses the conflicts between the read performance and lifetime improvement. Associate Editor: T. Dohi. *(Corresponding Author: Liang Shi.)*

Q. Li, C. Ji, Y. Liang, and C. J. Xue are with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong (e-mail: qiaoli045@gmail.com; chengji4-c@my.cityu.edu.hk; yliang22-c@my.cityu.edu.hk; jasonxue@cityu.edu.hk).

L. Shi is with the School of Computer Science and Software Engineering, East China Normal University, Shanghai 200062, China (e-mail: shi.liang.hk@gmail.com).

Y. Di and C. Gao are with the College of Computer Science, Chongqing University, Chongqing 400044, China (e-mail: yejia.di.cqu@gmail.com; albertgaocm@gmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TR.2019.2893287

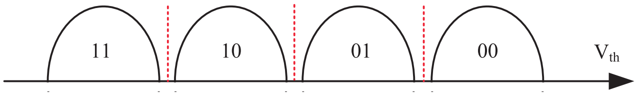


Fig. 1. Voltage distribution of flash memory cells that store two bits per cell with four voltage states.

of reference voltages while maintaining the same error correction capability. For example, the work from [8], [11], and [12] proposed to decrease reference voltages by exploiting error characteristics on NAND flash memory. A different strategy is proposed in [7] and [15] to utilize the access characteristics of workloads to reduce RBER or lossless compression in order to increase the redundancy for ECC. As a basic characteristic on NAND flash memory, PV presents significant influence on reliability, which has not been considered for read performance improvement. Great PV has been identified on existing NAND flash memory [9], [16]. Several recent work has well studied the characteristics of PV, which presents significant reliability variations among flash blocks [9]. PV has been exploited to optimize wear leveling strategies or improve write performance [9], [14], [16], [17]. However, these strategies ignored the PV influence on read access performance for NAND flash memory.

For LDPC-equipped NAND flash memory, the read latency is greatly dependant on the reliability characteristics of flash blocks. It takes longer latency to read data on blocks with lower reliability (higher RBER) than to read data on blocks with higher reliability (lower RBER). Hence in contrast to recent PV-based wear leveling methods, this paper exploits PV to reduce read latency on LDPC-based NAND flash memory. The goal is achieved from three parts. First, a flash block grouping scheme is presented, which categorizes the flash blocks according to their reliability characteristics. Since the latency of read requests is greatly dependant on the reliability of each block, flash blocks can be easily classified based on their read latency. Second, a PV-aware read data placement approach is proposed. The fundamental principal is to place read data according to the hotness. Read-hot data are placed on high reliable block groups, while read-cold data are placed on low reliable groups. This scheme is able to reduce the read latency to access read data. Nevertheless, it collides with state-of-the-art wear leveling methods with awareness of PV, because they proposed to allocate high-reliable blocks for write-hot data. To solve this issue, the third part of the paper presents a group partition scheme, which is designed to limit the amount of high reliable blocks allocated for read-hot data. As a result, the majority of high reliable blocks are still used in wear leveling for lifetime improvement. With these three parts, read performance improvement can be achieved and the impacts on wear leveling are minimized. The contributions of this paper are listed in the following:

1) present a flash block grouping method to categorize flash blocks according to the reliability;
2) propose a read data placement scheme to place read-hot data on high reliable blocks and read-cold data on low reliable blocks;
3) propose a block group partition scheme to resolve the conflict between the proposed approaches and previous PV-aware wear leveling approaches;
4) present an efficient implementation with negligible overhead. Experimental results present that the proposed approaches achieve great read performance improvement with little impact on lifetime improvement.

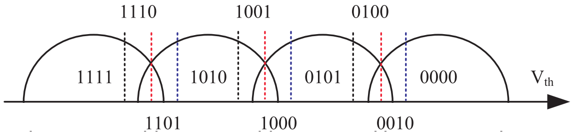


Fig. 2. Voltage distribution and more reference voltages for LDPC to decode data with higher RBER.

In the following, Section II states the background and related work. Section III shows the motivation of this paper. The proposed approaches are presented in Section IV. Section V presents the experiments and analysis. Section VI concludes the paper.

II. Background and Related Work

A. *Background*

NAND flash memory stores specific quantity of charges in floating gates or charge to represent the data stored. There will be 2^n different voltage states to store n bits in a flash cell. Fig. 1 shows an example for multilevel cell (MLC) NAND flash memory storing two bits in a cell, where four voltage states are defined. Each flash cell will be one of the four voltage states by injecting charges into the floating gates in programming operations. The reliability of stored data depends on the voltage distribution of each state and the noise margins between neighboring voltage states. As more bits are stored in a flash cell and smaller charge is used, the noise margins are narrowed, which leads to a higher RBER and thus, reliability degradation [6], [18]. To guarantee the correctness of read data, LDPC codes [7], [8], [12] have been adopted as ECC due to stronger error correction capability.

LDPC has a stronger error correction capability than traditional ECC, such as BCH. Basically, LDPC is defined with a sparse parity-check matrix that uses an iterative belief-propagation decoding algorithm to recover data. LDPC uses probability information as input for decoding, and the accuracy of input information directly impacts the error correction capability. Probability information with higher accuracy is achieved by performing more reference voltages to offer a stronger error correction capability. This can be explained by Fig. 2. When the threshold voltage distribution is overlapped, there will be more errors. To correctly decode the data, more reference voltages are performed to distinguish the states. In Fig. 2, nine reference voltages are used (three between every pair of neighboring states, compared to one in Fig. 1) to decode the data. As a result, the voltage distribution is divided into more partitions. There will be longer information bits to differentiate different regions, where four bits are needed in Fig. 2 contrasted to two bits in Fig. 1. The above process shows that in contrast to decoding data with a lower RBER, decoding data with a higher RBER requires higher time cost when using LDPC to recover data from NAND flash memory. The increased time costs include two aspects—an increased sensing time with many reference voltages, and an increased flash-to-controller information transfer time with long information bits [11]. This paper will exploit PV to reduce read latency on LDPC based NAND flash memory.

PV is a natural characteristic of semiconductors in NAND flash memory [14], [19]–[21]. During the manufacturing process, several memory cell device parameters, like oxide thickness, gate length and width, vary greatly, which result in significant variations of flash cell reliability [22]. The variation is especially severe with the continuous decrease of technology size and increase of bit density on NAND flash memory. In current 19-nm-cell NAND flash memory, the reliability variation between different blocks can reach ten times and more [14], [19]. Because LDPC decoding latency depends on the RBER of read data, flash read performance experiences great variations among different blocks as well. Read requests on blocks with higher reliability (lower RBER) require shorter latency than those on blocks with lower reliability (higher RBER). This paper will exploit this variation to improve read performance.

B. Related Work

In this section, the related works are presented from three aspects. First, there are many prior works for LDPC read performance improvement on NAND flash memory. The works in [8], [11], and [12] focused on optimizing LDPC decoding mechanism by exploiting the relationship between flash errors and LDPC decoding algorithm, where the reliability characteristics of flash memory are exploited to optimize the decoding process of LDPC. To avoid high time costs in decoding data with high RBER, Guo *et al.* [23] proposed to decrease RBER by enlarging noise margins between neighboring voltage states which have more errors, and Xie *et al.* [15] proposed to apply lossless compression to provide more redundancy for LDPC with stronger decoding capability. Li *et al.* [7] exploited read and write latency tradeoff to speed up read requests of read-only data for read performance improvement. Du *et al.* [24] proposed to correct errors in read-hot pages by refreshing more often. All these works are completely orthogonal to our paper because they mainly focus on the optimization of LDPC decoding.

Second, read-hotness of data is explored to improve flash read performance. For example, the works in [25] and [26] proposed techniques to migrate data based on read-hotness. As the read disturb accumulated from read operations causes voltage states shifting to a higher value, the migration technique in [25] is used to separate read-hot and read-cold data, where optimal read reference voltages can be applied differently on different blocks. Conversely, Ha *et al.* [26] mainly migrated read-hot data to different blocks to distribute read requests and avoid read disturb accumulation in one block. Nevertheless, these two approaches focused on read disturb issues, while our migration is based on the PV of NAND flash memory.

Finally, PV has been widely studied in previous works. State-of-the-art works mainly exploited PV for write performance [21] or lifetime improvement [9], [14], [16], [19]. Shi *et al.* [21] attempted to exploit the tradeoff between write speed and RBER of data, by applying higher programming speed on more reliable blocks to improve write performance. Cui *et al.* [27] proposed an I/O scheduling scheme based on the hotness and retention age of read data for conflict reduction. Others [9], [14], [16], [19], proposed PV-aware wear leveling schemes. Because high reliable blocks can endure more P/E cycles, they proposed to allocate more write requests to them for lifetime improvement. Di *et al.* [17] further studied the retention time variation among blocks and exploited it to reduce the frequency of refresh operations on NAND flash memory. However, all these PV-aware approaches cannot be applied for read performance improvement. In fact, these works introduced bad read performance, which will be discussed in the following section. This paper presents the first attempt to exploit PV for improving the read performance of NAND flash memory.

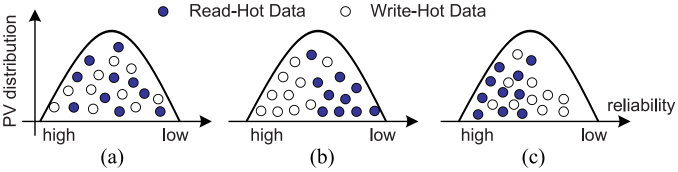


Fig. 3. Three data placement schemes: (a) PV-unaware scheme; (b) PV-aware wear leveling scheme; and (c) PV-aware data placement scheme for read performance improvement.

III. Motivation

In this section, a study on data placement schemes is presented for NAND flash memory with significant PV and degraded reliability. Recent works show that PV presents characteristics of reliability variation, which can be formulated as a Gaussian distribution [14], [16], [17]. Fig. 3 presents three data placement schemes on this distribution—traditional PV-unaware scheme, PV-aware wear leveling scheme for lifetime improvement, and PV-aware data placement approach for read performance improvement. In Fig. 3, hollow circle is used to represent write-hot data and solid circle to represent read-hot data. The *X*-axis represents the reliability of blocks and *Y*-axis represents the distribution of blocks with different reliability, which follows a Gaussian distribution. As shown in the figure, PV-unaware scheme does not consider the reliability variations among flash blocks, where the data are placed based on the default block allocation method in the flash memory controller [28]. The PV-aware wear leveling scheme represents the most recent work, which uses more reliable blocks for write-hot data for better wear leveling, and the PV-aware data placement scheme is an ideal scheme, which places read-hot data on high reliable blocks for better read performance.

A preliminary study focusing on read performance of the above three schemes is carried out on a set of widely studied workloads [29]. Table II shows the statistical information for the evaluated workloads. A well-configured SSD based on DiskSim simulator with SSD model [30] is simulated to conduct experiments where LDPC is configured as the default ECC. It is assumed that the PV of all the blocks and the hotness of data are both identified off-line, and can be used directly in the controller. The reliability characteristics among blocks are formulated by a Gaussian distribution, where the parameters for the Gaussian distribution comply with previous works [14], [17]. The details of the experiment settings are presented in the experimental section.

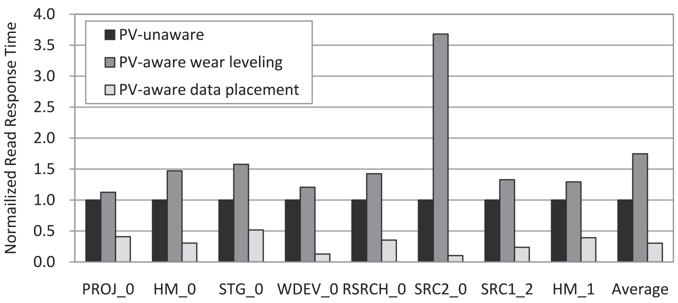


Fig. 4. Read response time comparison among the three data placement schemes.

Fig. 4 shows the experiment results. As shown in the figure, the PV-aware wear leveling scheme degrades read performance compared to the PV-unaware scheme by 70% on average. This is because this scheme prefers to allocate more reliable blocks for write-hot data. As a result, read-hot data are more likely to be allocated to less reliable blocks. On the contrary, the PV-aware data placement scheme improves read performance over the PV-unaware scheme by 80% averagely. The reason is that the read-hot data allocated to more reliable blocks are accessed with a smaller latency. Based on the results, the following observations can be made, which motivate this paper. First, data placement is critical for read performance. Second, recent data placement scheme for wear leveling largely deteriorates read performance.

IV. PV-Aware Data Placement

The above-mentioned analysis and results present that exploiting PV for read performance improvement is critical for NAND flash memory. In the following, a process variation aware data placement scheme is introduced to improve read performance of NAND flash memory. The basic idea of the scheme is to place read-hot data on blocks with high reliability. In this case, these data can be read with smaller latency. However, there are several challenges for this scheme. First, read-hot data should be identified ahead before the data placement. Second, as read-hot data are placed on blocks with high reliability, the PV-aware wear leveling will be impacted. In order to solve the above challenges, the following methods are presented. First, the read access characteristics of several workloads are studied to support the identification of read-hot data. Second, based on the observations from the study, a data placement approach is proposed. Third, a block group partition scheme is introduced to integrate the proposed data placement scheme with recent wear leveling schemes. Finally, the implementation and overhead analysis are presented.

A. Read Access Characteristics

The read access characteristics for real workloads are studied in this section. Data pages are first partitioned into N groups based on their read frequencies, which is achieved by counting the number of reads on each data page for all workloads. Fig. 5 presents the statistical results of cumulative distribution function (CDF) of the footprints and read requests for each group. The X-axis represents the count of read operations on a data page. There are seven groups using access count ranges, where requests or pages falling in the ranges are grouped in the same group.

Fig. 5(a) is the CDF of read pages. Based on the results, data pages are distinguished into three types, as well as three observations listed as follows.

1) *Read-cold:* A data page is read-cold if it is seldom read. As shown in the results, if we define a read-cold page as a page which is read only once or twice during its lifetime, we find that more than 73% of read pages are read-cold on average;
2) *Read-hot:* A data page is read-hot if it is frequently read. If we define a read-hot page as a page read more than ten times, we find that no more than 10% of pages are read-hot on average;
3) *Read-warm:* A data page is read-warm if it does not belong to the above-mentioned two cases. As shown in the results, few pages are read-warm.

With the above differentiation and the results for read requests in Fig. 5(b), we can make the following three observations:

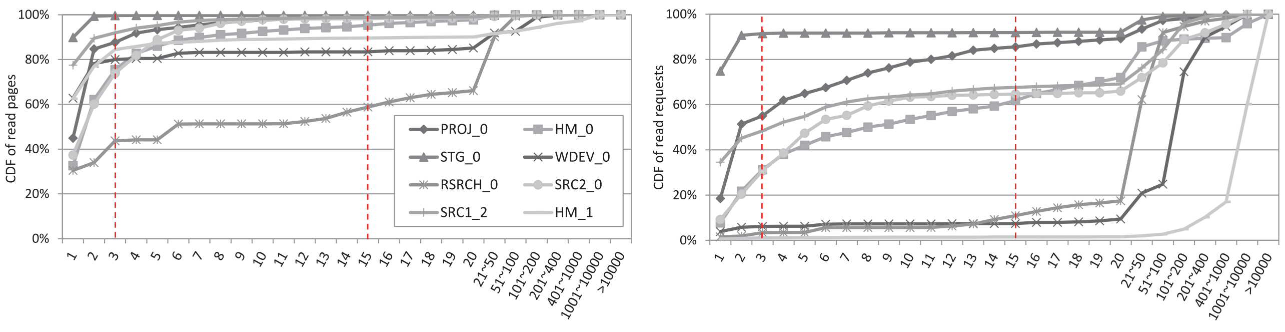
1) Observation 1—For some workloads, most read requests access read-cold pages. For example, for PROJ_0 and STG_0, more than 60% of read requests access cold data pages;
2) Observation 2—For some workloads, most read requests access read-hot pages. For example, for WDEV_0 and RSRCH_0, more than 80% of read requests access hot data pages;
3) Observation 3—For all the workloads, merely a small percentage of read requests access warm data pages.

These observations show that the read-hot and read-cold pages can be easily classified, where most read requests access either hot or cold data pages, only a few on warm data pages. It can be derived that if a data page has received several reads, for example, three reads, most likely it will be a hot page, because the data pages being read more than three times take a really small part. With this in mind and combining the characteristics of PV, a data placement scheme is proposed, which is implemented in NAND flash controller. The basic idea of the scheme is to migrate the data which have been read for a predefined number of times to high reliable blocks.

B. PV-Aware Data Placement

To realize the data placement scheme, two things need to be determined ahead. First, a classification method is required to classify the flash blocks into high reliable blocks and low reliable blocks. Second, the hotness of read data should be identified. After that, the identified data can be allocated or moved to the flash blocks with corresponding reliability. In the following, the flash block grouping scheme is first discussed. Then, according to the grouping, read data placement scheme is presented. Fig. 6 shows an overview for the PV-aware data placement scheme.

1) PV-Aware Block Grouping: Previously, there were several works proposed to identify the reliability characteristics of PV. For example, Woo *et al.* [16] proposed to use write latency of blocks as the metric for block grouping. The method is based on the fact that high reliable blocks always have longer programming latency. Different from previous work, the decoding latency of LDPC is used as the metric for block



(a)

(b)

Fig. 5. CDF of footprints and read requests. The X-axis represents the count of read operations on a data page. (a) CDF of read pages. (b) CDF of read requests.

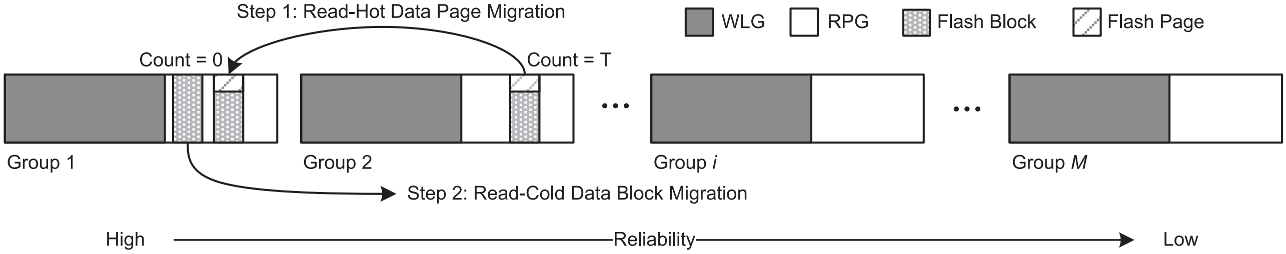


Fig. 6. Overview of the proposed schemes.

grouping. Two reasons account for the use of this new metric. First, LDPC can distinguish blocks with diverse reliability. Recall in Section II, the read latency using LDPC is greatly dependent on the RBER of the read data. In this case, the decoding latency of LDPC enables the easy classification of blocks. Second, the proposed approach is for read performance improvement on LDPC-equipped flash memory. Adopting access latency of LDPC for block differentiation is more direct for the online grouping of blocks.

Based on the above analysis, the process of PV-aware block grouping is discussed as follows. Current LDPC in NAND flash memory can support up to seven levels of voltage sensing, and larger number of reference voltages need longer read latency. Therefore, the flash blocks can be easily differentiated by counting the number of reference voltages applied to correctly read data. In this paper, flash block is the basic unit of grouping for simplicity. In each block, the page with the longest read latency is adopted to measure its reliability level for grouping. Assume that M block groups are developed in the proposed approach. Then, based on a specific grouping scheme, blocks are added to the corresponding group. In the experiment, there are five group schemes presented based on the supported number of sensing levels of LDPC.

Take Fig. 6 for an illustrative example. Among the M groups, group 1 is the most reliable group consisting of the blocks with the highest reliability, while group M is the least reliable group. The NAND flash controller maintains the group information, i.e., group ID. The information is updated online to accord with the block wearing. As a small number of P/E cycles may not cause appreciable reliability degradation on a flash block, reliability remeasuring of the flash block can be conducted after every specific number of P/E cycles, for example, N_{ITV}. If the reliability of a flash block is different from the reliability obtained from the last measurement, the group ID of the block may be updated based on the grouping scheme. Suppose the specified maximally endured number of P/E cycles is N_{PE}. Thus, the grouping scheme will be conducted N_{PE}/N_{ITV} times during the whole lifetime of NAND flash memory. For implementation, several bits are needed for each block to maintain the group information, which will be discussed in the implementation section. These information can also be stored in the out-of-band area of flash page for persistence. Then, they can be reconstructed when the block is read. For the existing solid state drives that contain several flash chips, the grouping scheme will be performed on each NAND flash chip.

2) PV-Aware Read Data Placement: In this section, we present the data placement scheme, which includes the following two steps.

Step 1: Migrating Read-Hot Data to High Reliable Group. Hotness of read data can only be identified after it has been read several times. Therefore, when it is written for the first time, default address allocation scheme [28] in the flash memory controller is applied. After its hotness is identified, it can be migrated to the corresponding group.

Based on the observations in Section IV-A, data hotness can be identified by counting read requests that have occurred on data pages. A threshold T is introduced for read hotness identification. Data pages that are read T times will be identified as read-hot. As shown in Section V, T can be simply set to three. The simple setting has two advantages. First, if a data page is read three times, it is likely to be accessed later and it has a good potential to be read-hot based on Fig. 5. Second, using a small number for hotness identification introduces less costs. For the setting with T equal to three, only a 2-bit counter is required for each flash page. During the implementation, the little cost can be easily realized in the NAND flash memory controller. As shown in Fig. 6, migration happens when a read data of group i ($i > 1$) has been identified as read-hot, which will be migrated to group $i-1$. The read count of the page will be reset to zero after migration.

During a specific time period, most data are cold data on NAND flash memory and will not be read. In addition, current hot data may become cold later. Therefore, we only record the read counters of the data pages which have their mapping information cached in the controller. To read a data page, the mapping information will first be loaded to the mapping cache. Therefore, the recorded read counter is the read count in a period, which can be regarded as read frequency. The hotness identification based on access frequency is also a widely used scheme in previous work [31], [32].

Step 2: Migrating Read-Cold Data Out of High Reliable Group. There is no free space in a high reliable group for read-hot data being migrated to it, read-cold data has to be elected and moved out of the group. Several strategies can be used to identify the read-cold data in the high reliable group, such as the least recently used cache management policy [33], [34]. For the design, two issues need to be considered in this situation. First, the granularity of the migration out approach is supposed to be flash block because a block is the basic unit for erase operations. Only after erase operations, the flash block will be in a free state to be used by new read-hot data. Second, the identification should be cost negligible. This is because the hotness information needs to be maintained for all blocks in the group. To meet the above-mentioned two requirements, a CLOCK algorithm [35]-based read-cold block migration out scheme is presented..

The CLOCK algorithm works as follows. First, except for group M, the least reliable group, all the blocks in every group is organized in a CLOCK structure. Second, a reference bit is associated with each block. The reference bit will be set when a read happens on the associated block. The block whose reference bit is unset will be selected to be migrated out of the group if there is no space available for new read-hot data. For migration operations, the valid pages in the victim block are copied to a block in a lower reliable group, followed by an erase operation to reclaim the space.

C. Group Partition for Read Performance Improvement and Wear Leveling

The aforesaid PV-aware data placement method can improve read performance via placing read-hot data on high reliable block groups. However, the approach collides with previous PV-aware wear leveling methods, such as RBER-based aware leveling and health-binning [9], [16], because they proposed to allocate write-hot data to high reliable blocks.

To mitigate the conflict between wear leveling schemes and the proposed PV-aware data placement scheme, a group partition scheme is introduced to minimize their impacts in this part. The basic idea of the approach is to further divide each group into two sub-groups, one for wear leveling (WLG) and the other for read performance improvement (RPG), as shown in Fig. 6. To reduce the influence on wear leveling, we limit the size of RPG, while all the blocks in the larger sub-group, WLG, are enrolled in wear leveling. For the blocks in WLG, previous schemes [9], [36] suggested to allocate them according to write-hotness. With the above scheme, most blocks of high reliable groups will still be prioritized for wear leveling. The blocks in RPG are used for read-hot data to improve read performance, following the above-mentioned schemes.

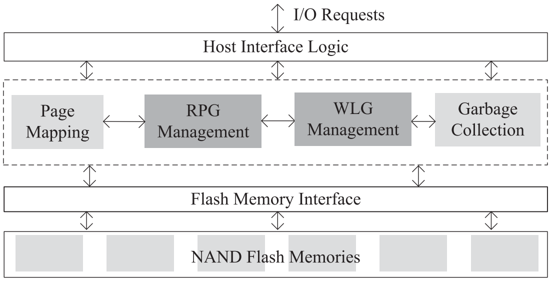


Fig. 7. Implementation of the proposed schemes.

In the experiment, the percentage of RPG is varied to study the impact on the wear leveling scheme. The experimental results show that a small-size RPG is able to support a significant read performance improvement. Please note that the partition sizes of these two sub-groups can be adaptively varied based on the access patterns of the workloads. For example, if the workload is read-intensive, the size of RPG can be enlarged and vice versa. In the experiments, the ratio is simply fixed for these two sub-groups to verify the little lifetime impact from the proposed scheme. Adaptive group partition scheme will be studied as a future work.

D. Implementation and Analysis

The proposed approach is implemented in NAND flash controller, as shown in Fig. 7. There are two components required in the controller: RPG management and WLG management. The RPG management component is used to support the read data placement schemes for read performance improvement, and WLG management component is used to support the wear leveling schemes presented in previous works [9], [14].

Initially, based on the read access latency with LDPC decoding, flash blocks are classified into M groups. During the classification, the grouping information are recorded in the flash controller. For each block, $\log M$ bits are used as the group ID. In this work, five grouping schemes, varying from 1 to 7 groups, will be evaluated, which are presented in the experimental setup. In the experiment, the three-group scheme is set as the default one. In this case, 2 bits will be needed for a block as the group ID. For the blocks in the high reliable groups, they are further partitioned into two sub-groups, RPG and WLG. If a block belongs to RPG, a 2-bit counter will be used for each page. The counter is increased when the data page is read. When the counter reaches the threshold T, the data is migrated and its counter is reset to zero. In order to minimize the cost for the read counters, we only maintain the counter information for the pages which have their mapping entries cached in the NAND flash memory controller. In the controller, there is a memory which is used to cache the mapping entries of flash pages. The data for the cache mapping entries are always the hot data. Thus, read counters are needed only for these data. In this case, the cost for the read counters can be significantly reduced. CLOCK algorithm is also implemented in the controller, where each CLOCK is constructed with an array of block number and a reference bit. When a block is accessed, the reference bit is set. A block is migrated if the reference bit is unset by the CLOCK algorithm, when a block has to be migrated to a lower reliable group. Finally, for each group, a free tag is required. Data from blocks migrated out of high reliable groups should be moved to a group with enough free pages. If a block belongs to WLG, it is implemented with recent PV-aware wear leveling schemes [9], [36].

There are two types of overheads for the implementation—storage overhead and firmware overhead. There are three parts that contribute to the storage overhead—the block grouping information, a 2-bit counter for each mapping entry in the flash controller, and a CLOCK structure for each group. First, if there are three groups, each block needs 2 bits to indicate its grouping information. For a 128 GB NAND flash memory whose page size is 4 KB and whose block contains 64 pages, 128 KB is required to store the grouping information. Second, if 10% mapping entries can be loaded in the cache of controller, the storage of the 2-bit counters needs at most 800 KB. Third, for the CLOCK algorithm, the overhead highly depends on the percentages of RPG. From the experiment, the result shows that 5% of high reliable group is enough for read performance improvement. As a result, the overhead for CLOCK algorithm will be no more than 200 KB. In all, the storage overhead is around 1 MB to implement the proposed approach. The firmware overhead comes from the computation required for read counter increase, CLOCK algorithm, and data migration [37]. The overhead is small as presented in the experiment.

V. Experiment and Analysis

In this section, the experimental methodology is first presented. Then the results will be presented together with the detailed analysis of the performance improvement, lifetime impact, and sensitivity studies.

A. Experimental Setup

The experiments are conducted on a widely used simulator, DiskSim [30] with SSD model [28]. We adopt MLC NAND flash memory to simulate the experiment storage system, whose capacity is configured as 128 GB. The simulated solid state drive has 8 channels, each of which has 4 chips with 4 planes in each chip. There are 2048 blocks in a plane and a block consists of 64 4 KB pages.

Block Grouping: Considering that LDPC supports up to seven reference voltages between each pair of neighboring voltage states, blocks can be grouped into seven groups based on the sensing level applied to successfully decode the data. For the experiment evaluations, five grouping schemes are implemented in this section. The five group schemes are illustrated in Table I. The first row of the table is the number of reference voltages supported by LDPC. The second to sixth rows are the five group schemes used in the experiment. The 1-group scheme means all flash blocks belong to one group, which is the traditional PV-unaware scheme. For the other group schemes, they can be applied in this paper. Take the 3-group scheme as an explanation example. Blocks that store data being correctly decoded by LDPC with one reference voltage are classified into group 1. Blocks that can be corrected by LDPC with two, three, or four reference voltages are classified into group 2, and other blocks are classified into group 3. Furthermore, due to the dynamic changes in the reliability of flash blocks during flash wearing, the distributions of blocks in each group are varied correspondingly. PV among blocks is formulated as a Gaussian distribution based on existing works [14], [21]. The "distributions" row of Table I presents the simulated PV distribution, which is derived from previous work [14]. The above five block grouping schemes will be evaluated, where the 3-group scheme is configured as the default one. In the implementation, the distributions of blocks can be initialized during system start-up.

TABLE I
Block Grouping Schemes, Reference Voltages, and Access Latencies

Sensing Level	1	2	3	4	5	6	7
1-Group	1						
2-Group	1	2					
3-Group	1	2			3		
4-Group	1	2		3		4	
7-Group	1	2	3	4	5	6	7
Distributions (%)	9	12.5	21.3	19.6	17.6	12.9	7
Sensing latency (μs)	25	50	75	100	125	150	175
Transfer latency (μs)	20	30	40	40	40	50	50

TABLE II
Statistical Information for Workloads

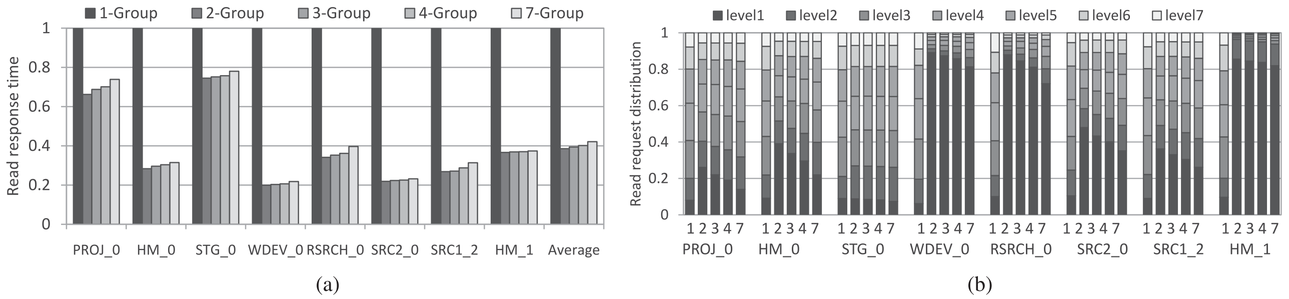
Workload	Footprint(GB)	Read(GB)	Write(GB)	R.Ratio	Read-Hot
PROJ_0	1.47	2.11	7.48	22%	0.7%
HM_0	0.56	1.40	4.08	25%	14.6%
STG_0	3.20	3.76	4.73	44%	0.3%
WDEV_0	0.26	1.62	4.69	26%	7.6%
RSRCH_0	0.17	0.69	5.60	11%	11.7%
SRC2_0	0.36	0.83	4.62	15%	14.9%
SRC1_2	0.82	1.26	18.30	6%	5.6%
HM_1	0.10	4.34	3.47	56%	11.3%

Read Latency Model: Read latencies applied in the experiment, presented at the last two rows of Table I, is derived based on the latency model in [7] and [13]. There are two main steps for a read operation before LDPC decoding—sensing the data stored in the flash page and transferring the data to the controller. In this case, read latency is the sum of the sensing latency, which is proportional to the number of reference voltages, and transfer latency, which is proportional to the length of the transferred information [11]

$$\begin{aligned} &\mathrm{RC}(N) = \alpha \times N + \beta \times \lceil \log(N+1) \rceil \\ \text{where} \quad &\mathrm{RBER} < \mathrm{CBER}_{\mathrm{LDPC}}(N) \end{aligned} \tag{1}$$

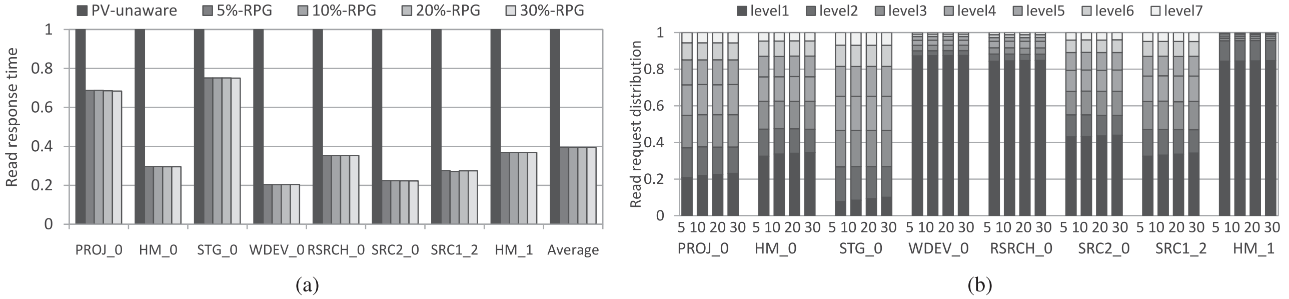
where $\mathrm{RC}(N)$ denotes the read latency with N reference voltages in the applied LDPC code, α and β are two variables. The condition is that the RBER of read data should not exceed the error correction capability of the deployed LDPC code, $\mathrm{CBER}_{\mathrm{LDPC}}(N)$, with N sensing levels.

Workloads: Several real-world traces of Microsoft Research (MSR) Cambridge [29] which are collected from enterprise servers are used to evaluate the proposed schemes. Table II presents statistical information for the evaluated workloads. The footprint, the read size, and write size of each workload are listed. For each workload, several gigabit of read or write data are evaluated. The R.Ratio in Table II presents the percentages of read operations. Read-Hot in the table shows the ratio of read-



(a) (b)

Fig. 8. Read performance evaluation of different block grouping schemes. (a) Normalized read response time. (b) Read request distributions.



(a) (b)

Fig. 9. Evaluation of group partition schemes with different percentages of read performance improvement sub-group. (a) Normalized read response time. (b) Read request distributions.

hot data pages among footprints of workloads. The threshold T is set as 3 for the identification of read-hot data in default.

B. Experimental Results

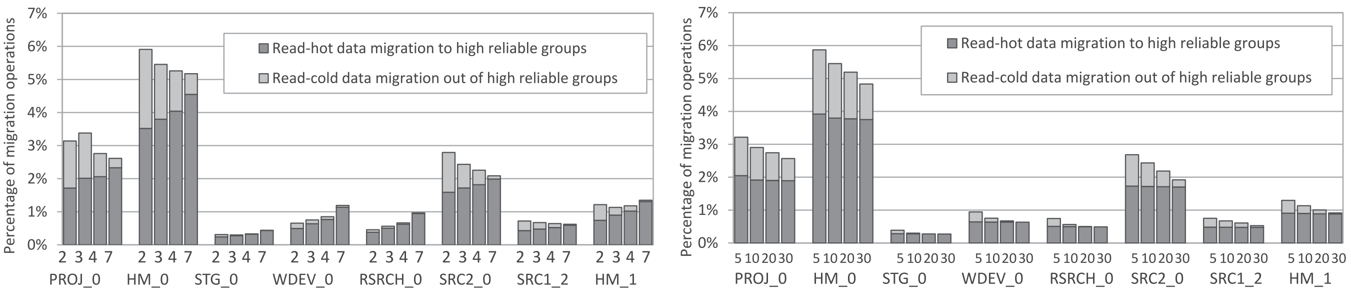
In this section, the read performance improvement is first evaluated by varying the number of groups and the size of RPG. Then, the lifetime impact from the proposed scheme is discussed. Finally, several sensitivity studies are presented.

Read Performance Improvement: Two sets of experiments are performed for read performance evaluation. The first set varies the number of groups and the second varies the size of RPG.

For the first set, the number of groups is varied from 1 to 7 based on Table I. The size of RPG is set as 10% of its group. Fig. 8 shows the evaluated results. Fig. 8(a) is the normalized read response time of the five block grouping schemes. The 1-group scheme is the traditional PV-unaware scheme. The proposed approach works for the other four group schemes. As shown in Fig. 8(a), we can make the following two conclusions. First, the proposed scheme can significantly reduce read response time compared to the PV-unaware method by more than 60% averagely. Second, there are little variations among the read response time for the different grouping schemes. The grouping scheme that having lower number of groups presents more read response time reduction. This is because a read-hot data page will be migrated to the most reliable group with less steps when the number of groups is small. To illustrate the details of the read performance improvement, the sensing level distributions of LDPC for all the read requests are also collected for each workload. The results are presented in Fig. 8(b). For each column, the percentages of read requests processed by different number of sensing levels are listed. For example, if the read request is decoded successfully by two sensing levels, the request will be grouped to the level2 group. As shown in the figure, compared with the 1-group scheme, the other group schemes have more read requests processed by less number of sensing levels. In addition, comparing the four group schemes, for most workloads, more read requests are processed by smaller number of sensing levels, especially for workloads, such as PROJ_0, HM_0, and SRC2_0. This is the reason why the grouping scheme with less number of groups presents better performance compared with the larger ones, which corresponds to the results for read response time.

For the second set of experiments, the size of RPG is varied from 5% to 30%. Fig. 9 shows the evaluated results. In the experiments, the number of groups is set to 3 in default. The results in Fig. 9(a) show that there will be little difference in read response time by varying the size of RPG. These results can be explained by the information presented in Table II. As illustrated in Table II, only a small percentage of data within the working set are read-hot based on the hotness identification scheme presented in this paper. In addition, for the read-hot data pages that are also write-hot, they will be allocated to the WLG, while for the read-hot data pages which are write-cold, they will be seldom updated and need only small space. As a result, only a small number of high reliable flash blocks are needed to speed up the read access of read-hot data. Fig. 9(b) shows the read request distributions among the seven sensing levels. The results also confirm that with the increasing size of the RPG, the distributions are similar.

Lifetime Impact: The proposed work is designed to improve read performance by exploiting the PV of NAND flash memory. However, migrating read-hot data to blocks with high reliability collides with current PV-based wear leveling methods. In the following, the lifetime impact of the proposed work is discussed. First, only the minority of the high reliable blocks are applied to improve read performance. The majority of blocks with high reliability are still used for lifetime improvement. From the above



(a)

(b)

Fig. 10. Migration costs. (a) Migration costs of different block grouping schemes. (b) Migration costs of different percentages for RPG.

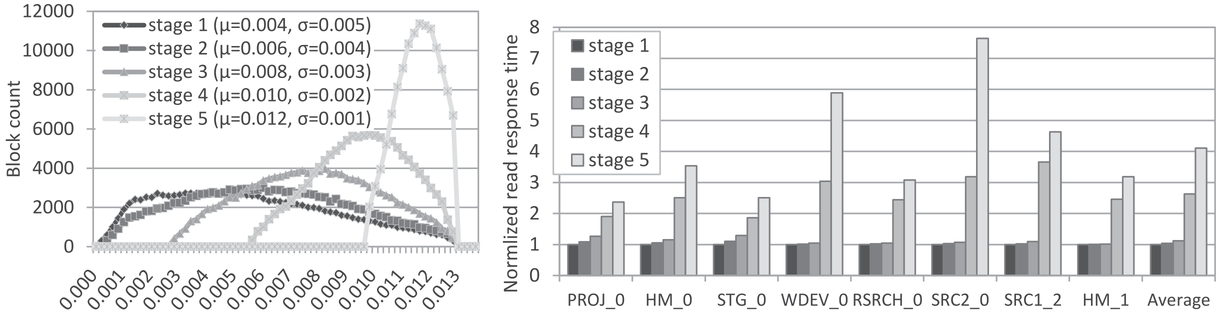
experimental results, great performance improvement can be achieved by a small size of RPG, i.e., 5%. The majority blocks with high reliability are still used for wear leveling, and thus, the proposed scheme presents a small influence on the lifetime improvement of NAND flash memory. Another important effect on the lifetime of flash memory exists in the newly generated write operations on NAND flash memory by the proposed work. A data page will be migrated if it is identified as read-hot and a read-cold block will be migrated to make room for newly identified hot data, where write operations are introduced. This is the additional wearing to the flash memory compared with the traditional scheme. To understand the additional wearing characteristics, we collect the percentages of write operations caused by data migration through varying the number of groups and size of RPG. Fig. 10 presents the migration costs, where the migration operation is normalized to the total requests, including migration of read-hot data to block groups with high reliability and migration of read-cold data out of block groups with low reliability. Fig. 10(a) presents the migration costs of different group schemes. The figure shows that there are much read-hot data migration than read-cold data migration. This is because most of the flash blocks are grouped as low reliable blocks, and read-cold data does not need to be migrated. Second, the grouping schemes with larger number of groups incur lower costs for most traces. Third, the overall costs are little, 2% on average. Fig. 10(b) presents the migration costs with different sizes of RPG. The cases with higher percentages generally introduce less cost for read-cold data migration, which comes from the reduction of migration operations to move read-cold data out of high reliable groups. Overall, the migration cost is small.

To conclude, the lifetime degradation of the proposed approach comes from the following two parts. First, as Fig. 9(a) shows, significant read performance improvement can be achieved by a small size of RPG, i.e., 5% blocks with high reliability. The majority blocks with high reliability are still used for wear leveling. If all the 5% blocks are never used for wear leveling, the lifetime will be influenced by 5%. However, these blocks will also endure some program and erase cycles; therefore, the lifetime will be impacted less than 5%. Second, the migration operations introduce extra wearing to flash blocks. The migration costs are shown in Fig. 10. The figure shows the ratio of migration operations to the total requests, including the migration operations that move read-hot data to block groups with high reliability and those to move read-cold data out of high reliable groups. The ratio is low, less than 2% on average. Overall, the proposed approach will result in less than 7% degradation on the lifetime of NAND flash memory.

From the above evaluations, the conclusion can be made that the proposed approach is able to provide significant read performance improvement while incurring a small impact on the lifetime of NAND flash memory. In addition, the proposed approach is also compatible with the current PV-aware wear leveling methods.

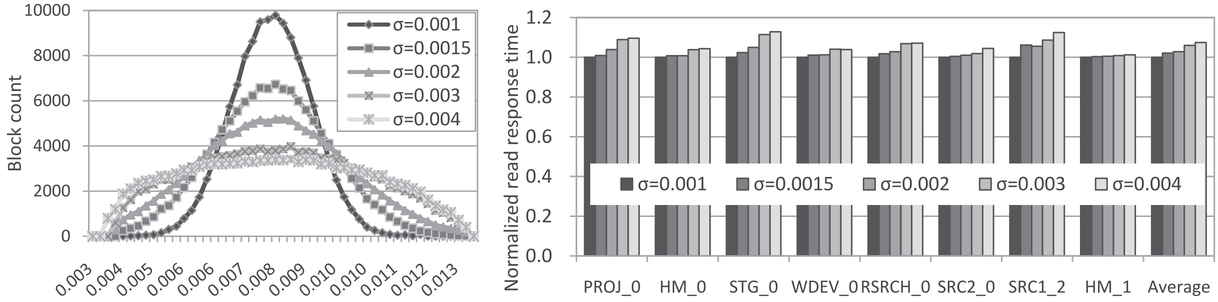
Sensitivity Studies: In the following, a set of sensitivity studies are presented on read performance along with different lifetime periods, different PV distributions, and different thresholds for the read hotness identification.

1) Read Performance During Different Lifetimes: The PV characteristics are generated during the fabrication of NAND flash chip. With the wearing increase of NAND flash memory, the reliability characteristics of different blocks will be changed. As presented above, the varied reliability characteristics can be identified online based on the sensing level required by LDPC during read operations. With the wearing of flash blocks, their reliability is degraded and blocks are grouped online when they are read after specific P/E cycles. In this section, the read performance variation during the lifetime of NAND flash memory is evaluated. To simulate the different lifetime stages, we assume that there is an ideal wear leveling scheme that is able to wear the flash blocks according to the reliability characteristics of flash blocks, such as the RBER-based wear leveling scheme [14]. Based on this idea, the reliability characteristics are simulated by varying the parameters of the Gaussian function $N(\mu, \sigma)$ as follows: μ is varied from 0.004 to 0.012 and σ from 0.005 to 0.001 for five stages. Fig. 11(a) shows the distributions of the five stages, where *X*-axis represents the RBER of blocks and *Y*-axis is the block distributions. The simulation results present that with the wearing of flash blocks under an ideal wear leveling scheme, the reliability distribution presents two features. First, reliability of blocks are gracefully degraded. Second, the distributions of blocks regarding reliability have become more uniform with the life progress. Based on the above-mentioned lifetime stage simulation, the read performance for different life stages is presented in Fig. 11(b). As shown in the figure, the read performance is degraded with the wearing of NAND flash memory. From the results, we believe that if there are reliability variations among blocks, the proposed approach can improve read performance. The improvement will be more when there exists more severe PV.



(a) (b)

Fig. 11. Evaluation of different lifetime stages. (a) PV distributions. (b) Normalized read response time.



(a) (b)

Fig. 12. Sensitivity studies on different PV. (a) PV distributions. (b) Normalized read response time.

2) Performance With Different PV Distributions: PV characteristics highly depend on several factors of flash memory, such as technology sizes, bit densities, and organizations. In this section, different types of PV distributions are evaluated to show the advantages of the proposed work. Using RBER as the metric, the parameter σ of the Gaussian function is varied from 0.01 to 0.04 to represent different PV distributions. Varying σ is to change the distributions of blocks, which are presented in Fig. 12(a). As shown in the figure, there are five PV distributions with different ranges of reliability characteristics to represent various types of PV on different chips. Fig. 12(b) shows the normalized read response time for these five PV distributions. There are two observations from the results in the figure. First, for a wider PV distribution, the read performance is slightly reduced. The reason is that a wider PV distribution introduces more low reliable blocks, which leads to increased access latency. Second, varying the block distributions with similar reliability characteristics has a small impact on the read performance improvement. The results further confirm that the proposed approach is able to improve read performance for different types of NAND flash chips with varied PV characteristics.

3) Performance With Different Threshold: The last group of sensitivity study focuses on the threshold for the read-hotness identification. The observations in Section III indicate that a small threshold is able to differentiate the read-hot and read-cold data. In the above experiment, the threshold with three is used as the default one. In this section, to understand the characteristics of this parameter, experiments are conducted to show the impact of threshold on the read performance and migration cost. Basically, the threshold T is important for the design. Because for the small T, the read-cold data may be identified as read-hot, which would introduce large amount of data migration with little benefit. However, for a large T, the read-hot data may be identified as read-cold, which would induce little read performance improvement. In addition, a large T would also introduce storage overhead for the hotness identification counter. In this section, the threshold is varied from 2 to 5 to show its impact on the performance and migration cost. Fig. 13(a) is the result for the read response time compared to the PV-unaware scheme. The results show that a small threshold is enough for read performance improvement. Fig. 13(b) is the result for the data migration cost. The results show that a small threshold introduces more migration cost, especially for threshold two. In this paper, to satisfy both the performance and implementation cost, threshold is set to three as the default one, which only needs 2 bits as the read counter and introduces great read performance improvement and reasonable migration costs.

TABLE III
COMPARISON BETWEEN LDPC AND BCH

Rate-8/9 Length-4KB	BCH	LDPC-hard	LDPC-soft (7-level)
Error correction capability	0.0041	0.0044	0.0132
Read latency (μs)	51	53	233

Comparison Between LDPC and BCH: We present the comparison between LDPC and BCH on NAND flash memory. Table III shows the error correction capability and read latency of BCH and LDPC, including LDPC hard decision with one sensing level between two neighboring states and LDPC soft decision with seven sensing levels. Both BCH and LDPC codes are configured as 8/9 code rate and 4 KB length. The error correction capability is for the RBER with the decoding failure probability 10^{-15}. On one hand, LDPC can correct data

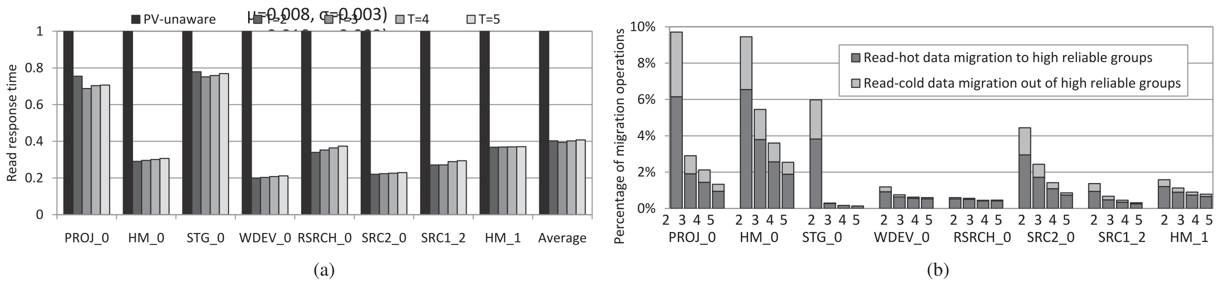


Fig. 13. Results of different threshold for read hotness identification. (a) Normalized read response time. (b) Migration cost.

with much higher RBER. With the development of NAND flash memory, the required error correction capability is increasing. For example, current 3-D NAND designs should be able to correct data with RBER higher than 0.0078 [38], which cannot be satisfied by BCH. If using BCH as ECC on such 3-D NAND flash memory, the P/E cycles will be reduced from 4000 to 1500. On the other hand, LDPC soft decoding with more sensing levels shows longer read latency to correct data with higher RBER, which will present bad read performance. LDPC hard decoding presents comparable read latency to BCH. With the proposed approach, most read requests will happen on blocks with low RBER where read-hot data are placed on. Therefore, the probability to trigger LDPC soft decoding is low. The proposed approach with LDPC on NAND flash memory can achieve a good read performance.

VI. Conclusion

Degraded reliability and significant PV are two critical issues on current NAND flash memory. LDPC is adopted to address the reliability issue by providing strong correction capability, but it significantly degrades read performance. Furthermore, the problem is amplified by existing wear leveling methods based on PV. To improve read performance, this paper proposed a read data placement approach by exploiting the reliability characteristics of PV. First, a flash block grouping method was introduced to classify blocks into several groups according to their reliability. Second, a data placement scheme was proposed to allocate read-hot data to blocks with high reliability to improve read performance. Third, a block group partition scheme was proposed to relax the conflicts between the proposed data placement scheme and the recent wear leveling schemes by limiting the number of blocks for read performance improvement. The experimental results presented that our approach introduced significant performance improvement and had small impacts on the lifetime of NAND flash memory.

Recently, 3-D NAND flash memory has been widely developed for its high density and better technology scaling capability. However, recent works show that PV is also a key issue for 3-D NAND. In further work, we will extend the proposed work to 3-D NAND by exploiting the specific PV characteristics for performance improvement.

References

[1] Q. Li, L. Shi, Y. Di, Y. Du, C. J. Xue, and H. Edwin, "Exploiting process variation for read performance improvement on LDPC based flash memory storage systems," in *Proc. IEEE Int. Conf. Comput. Des.*, 2017, pp. 681–684.

[2] D. Kang, W. Jeong, and C. Kim, "256Gb 3b/cell V-NAND flash memory with 48 stacked WL layers," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2016, pp. 130–131.

[3] S. Lee, J. Y. Lee, I. H. Park, and J. Park, "A 128Gb 2b/cell NAND flash memory in 14nm technology with tPROG=640ms and 800MB/s I/O rate," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2016, pp. 138–139.

[4] J. W. Im, W. P. Jeong, and D. H. Kim, "A 128Gb 3b/cell V-NAND flash memory with 1 Gb/s I/O rate," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2015, pp. 1–3.

[5] K. T. Park *et al.*, "Three-dimensional 128 Gb MLC vertical NAND flash memory with 24-WL stacked layers and 50 MB/s high-speed programming," *IEEE J. Solid-State Circuits*, vol. 50, no. 1, pp. 204–213, Jan. 2015.

[6] Y. Cai, E. F. Haratsch, O. Mutlu, and K. Mai, "Error patterns in MLC NAND flash memory: Measurement, characterization, and analysis," in *Proc. IEEE Des. Autom. Test Eur.*, 2012, pp. 521–526.

[7] Q. Li *et al.*, "Access characteristic guided read and write cost regulation for performance improvement on flash memory," in *Proc. USENIX Conf. File Storage Technol.*, 2016, pp. 125–132.

[8] M. Zhang, F. Wu, X. He, P. Huang, S. Wang, and C. Xie, "REAL: A retention error aware LDPC decoding scheme to improve NAND flash read performance," in *Proc. IEEE Symp. Mass Storage Syst. Technol.*, 2016, pp. 1–13.

[9] R. A. Pletka and S. Tomić, "Health-binning: Maximizing the performance and the endurance of consumer-level NAND flash," in *Proc. ACM Int. Syst. Storage Conf.*, 2016, pp. 31–40.

[10] X. Jimenez, D. Novo, and P. Ienne, "Wear unleveling: Improving NAND flash lifetime by balancing page endurance," in *Proc. USENIX Conf. File Storage Technol.*, 2014, pp. 47–59.

[11] G. Dong, N. Xie, and T. Zhang, "Enabling NAND flash memory use soft-decision error correction codes at minimal read latency overhead," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 60, no. 9, pp. 2412–2421, Sep. 2013.

[12] K. Zhao, W. Zhao, H. Sun, T. Zhang, X. Zhang, and N. Zheng, "LDPC-in-SSD: Making advanced error correction codes work effectively in solid state drives," in *Proc. USENIX Conf. File Storage Technol.*, 2013, pp. 244–256.

[13] Q. Li *et al.*, "Maximizing IO performance via conflict reduction for flash memory storage systems," in *Proc. IEEE Des. Autom. Test Eur.*, 2015, pp. 904–907.

[14] Y. Pan, G. Dong, and T. Zhang, "Error rate-based wear-leveling for NAND flash memory at highly scaled technology nodes," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 21, no. 7, pp. 1350–1354, Jul. 2013.

[15] N. Xie, G. Dong, and T. Zhang, "Using lossless data compression in data storage systems: Not for saving space," *IEEE Trans. Comput.*, vol. 60, no. 3, pp. 335–345, Mar. 2011.

[16] Y.-J. Woo and J.-S. Kim, "Diversifying wear index for MLC NAND flash memory to extend the lifetime of SSDs," in *Proc. ACM Int. Conf. Embedded Softw.*, 2013, pp. 1–10.

[17] Y. Di, L. Shi, K. Wu, and C. J. Xue, "Exploiting process variation for retention induced refresh minimization on flash memory," in *Proc. IEEE Des. Autom. Test Eur.*, 2016, pp. 391–396.

[18] P. Huang, P. Subedi, X. He, S. He, and K. Zhou, "FlexECC: Partially relaxing ECC of MLC SSD for better cache performance," in *Proc. USENIX Annu. Tech. Conf.*, 2014, pp. 489–500.

[19] M. C. Yang, Y. H. Chang, C. W. Tsao, and P. C. Huang, "New ERA: New efficient reliability-aware wear leveling for endurance enhancement of flash storage devices," in *Proc. ACM/EDAC/IEEE Des. Autom. Conf.*, 2013, pp. 1–6.

[20] D. Wei, L. Deng, L. Qiao, and P. Zhang, "PEVA: A page endurance variance aware strategy for the lifetime extension of NAND flash," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 24, no. 5, pp. 1749–1760, May 2016.

[21] L. Shi, Y. Di, M. Zhao, C. J. Xue, K. Wu, and E. H.-M. Sha, "Exploiting process variation for write performance improvement on NAND flash memory storage systems," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 24, no. 1, pp. 334–337, Jan. 2016.

[22] A. Spessot *et al.*, "Variability effects on the VT distribution of nanoscale NAND flash memories," in *Proc. IEEE Int. Rel. Phys. Symp.*, 2010, pp. 970–974.

[23] J. Guo, W. Wen, J. Hu, D. Wang, H. Li, and Y. Chen, "Flexlevel: A novel NAND flash storage system design for LDPC latency reduction," in *Proc. ACM/EDAC/IEEE Des. Autom. Conf.*, 2015, pp. 1–6.

[24] Y. Du, Q. Li, L. Shi, D. Zou, H. Jin, and C. J. Xue, "Reducing LDPC soft sensing latency by lightweight data refresh for flash read performance improvement," in *Proc. ACM/EDAC/IEEE Des. Autom. Conf.*, 2017, pp. 1–6.

[25] A. Kobayashi, T. Tokutomi, and K. Takeuchi, "Versatile TLC NAND flash memory control to reduce read disturb errors by 85% and extend read cycles by 6.7-times of read-hot and cold data for cloud data centers," in *Proc. IEEE Symp. VLSI Circuits*, 2016, pp. 1–2.

[26] K. Ha, J. Jeong, and J. Kim, "A read-disturb management technique for high-density NAND flash memory," in *Proc. ACM Asia-Pacific Workshop Syst.*, 2013, pp. 13–18.

[27] J. Cui, W. Wu, X. Zhang, J. Huang, and Y. Wang, "Exploiting latency variation for access conflict reduction of NAND flash memory," in *Proc. IEEE Symp. Mass Storage Syst. Technol.*, 2016, pp. 1–7.

[28] N. Agrawal, V. Prabhakaran, T. Wobber, J. D. Davis, M. S. Manasse, and R. Panigrahy, "Design tradeoffs for SSD performance," in *Proc. USENIX Annu. Tech. Conf.*, 2008, pp. 57–70.

[29] D. Narayanan, E. Thereska, A. Donnelly, S. Elnikety, and A. Rowstron, "Migrating server storage to SSDS: Analysis of tradeoffs," in *Proc. ACM Eur. Conf. Comput. Syst.*, 2009, pp. 145–158.

[30] J. S. Bucy, J. Schindler, S. W. Schlosser, G. R. Ganger, and contributors, "The DiskSim simulation environment version 4.0 reference manual," Parallel Data Lab., Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CMU-PDL-08-101, 2008.

[31] J.-W. Hsieh, T.-W. Kuo, and L.-P. Chang, "Efficient identification of hot data for flash memory storage systems," *ACM Trans. Storage*, vol. 2, no. 1, pp. 22–40, 2006.

[32] D. Park and D. H. Du, "Hot data identification for flash-based storage systems using multiple bloom filters," in *Proc. IEEE Symp. Mass Storage Syst. Technol.*, 2011, pp. 1–11.

[33] H. Kim and S. Ahn, "BPLRU: A buffer management scheme for improving random writes in flash storage," in *Proc. USENIX Conf. File Storage Technol.*, 2008, pp. 239–252.

[34] C. Yang, P. Jin, L. Yue, and P. Yang, "Efficient buffer management for tree indexes on solid state drives," *Int. J. Parallel Program.*, vol. 44, no. 1, pp. 5–25, 2016.

[35] W. Stallings and G. K. Paul, *Operating Systems: Internals and Design Principles*, vol. 3, Upper Saddle River, NJ, USA: Prentice Hall, 1998.

[36] X. Hu, R. Haas, and E. Evangelos, "Container marking: Combining data placement, garbage collection and wear levelling for flash," in *Proc. IEEE Model., Anal., Simul. Comput. Telecommun. Syst.*, 2011, pp. 237–247.

[37] S. Lee, K. Ha, K. Zhang, J. Kim, and J. Kim, "FlexFS: A flexible flash file system for MLC NAND flash memory," in *Proc. USENIX Annu. Tech. Conf.*, 2009, pp. 1–14.

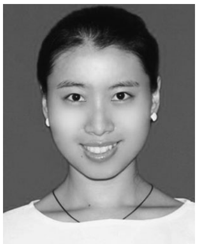
[38] M. M. Shihab, J. Zhang, M. Jung, and M. Kandemir, "ReveNAND: A fast-drift-aware resilient 3D NAND flash design," *ACM Trans. Archit. Code Optim.*, vol. 15, no. 2, 2018, Art. no. 17.

Qiao Li received the B.S. and M.S. degrees in computer science and technology from Chongqing University, Chongqing, China, in 2014 and 2017, respectively. She is currently working toward the Ph.D. degree in computer science with the Department of Computer Science, City University of Hong Kong, Hong Kong.

Her research interests include NAND flash memory, embedded systems, and computer architecture.

Liang Shi received the B.S. degree in computer science from the Xi'an University of Post & Telecommunication, Xi'an, China, in 2008 and the Ph.D. degree in computer science from the University of Science and Technology of China, Hefei, China, in 2013.

He is now a Full-Time Professor with the School of Computer Science and Software Engineering, East China Normal University, Shanghai, China. His research interests include flash memory, embedded systems, and emerging nonvolatile memory technology.



Yejia Di received the B.S. degree in computer science and technology from Chongqing University, Chongqing, China, in 2014. She is currently working toward the Ph.D. degree in computer science with the College of Computer Science, Chongqing University.

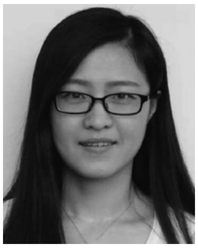
Her research interests include embedded and real-time systems, flash memory, and system optimizations.

Congming Gao received the B.S. degree in computer science and technology from Chongqing University, Chongqing, China, in 2014, where he is currently working toward the Ph.D. degree in computer science with the College of Computer Science.

His current research interests include embedded and real-time systems, nonvolatile memory, and architecture optimizations.

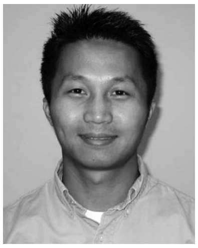
Cheng Ji received the B.S. degree from the School of Computer Science and Communication Engineering, Jiangsu University, China, in 2011, and the M.E. degree from the School of Computer Science and Technology, University of Science and Technology of China, Hefei, China, in 2014, both in computer science. He is currently working toward the Ph.D degree in computer science with the Department of Computer Science, City University of Hong Kong, Hong Kong.

His research interests include embedded systems, nonvolatile memory, and hardware/software co-design.



Yu Liang received the B.E. and M.E. degrees from the Department of Computer Science and Technology, Shandong University, Jinan, China, in 2010 and 2013, respectively, both in computer science. She is currently working toward the Ph.D. degree in computer science at the Department of Computer Science, City University of Hong Kong, Hong Kong.

Her research interests include file systems, memory management, and Android systems.



Chun Jason Xue received the B.S. degree in computer science and engineering from the University of Texas at Arlington, Arlington, TX, USA, in 1997, and the M.S. and Ph.D. degrees in computer science from the University of Texas at Dallas, Richardson, TX, USA, in 2002 and 2007, respectively.

He is currently an Associate Professor of Computer Science with the Department of Computer Science, City University of Hong Kong, Hong Kong. His current research interests include memory and parallelism optimization for embedded systems, software/hardware codesign, real-time systems, and computer security.