Exploiting Asymmetric Errors for LDPC Decoding Optimization on 3D NAND Flash Memory

Qiao Li, Liang Shi, Yufei Cui, and Chun Jason Xue

Abstract—By stacking layers vertically, the adoption of 3D NAND has significantly increased the capacity for storage systems. The complex structure of 3D NAND introduces more errors than planer flash. To address the reliability issue, low-density parity-check (LDPC) code with a strong error correction capability is now widely applied on 3D NAND flash memory. However, LDPC has long decoding latency when the raw bit error rates (RBER) are high. This is because it needs fine-grained soft sensing between voltage states to iteratively decode the raw data. Multiple sensing voltages are applied on flash cell array to gain necessary information for decoding. In this article, a new sensing level placement scheme with reduced number of sensing levels is proposed. The basic idea for the placement scheme is motivated by three asymmetric error characteristics of flash memory: the asymmetric errors between different states, the asymmetric errors caused by voltage left-shifts and right-shifts and asymmetric errors among layers in a 3D NAND flash block. With awareness of these three types of error characteristics, reduced number of sensing levels are placed to achieve reduced read latency for LDPC decoding while maintaining the error correction capability of LDPC. Experiment analysis shows that the proposed scheme achieves significant performance improvement.

Index Terms—Asymmetric errors, LDPC, sensing level placement, NAND flash memory

1 Introduction

To continuously increase the capacity and decrease the cost per bit of NAND flash memory, flash vendors have been aggressively increasing the bit density and scaling flash cells to smaller process nodes. Due to the challenges in technology scaling, 3D NAND flash memory has been introduced. Though 3D NAND has been deployed at larger scale, there exists a rising reliability issue for NAND flash memory [2], [3], [4]. Strong error correction code (ECC) schemes are indispensable in ensuring the data integrity [5], [6], [7], [8], [9]. With a strong error correction capability, low-density parity-check (LDPC) code has been widely used for the state-of-the-art flash memory during the last decade [5], [6]. Different from hard-decision code schemes, such as RS-codes and BCH, LDPC uses soft-decision log-likelihood ratio (LLR) information to achieve a high error correction capability. However, the challenge for LDPC is that its decoding latency is significantly increased for data with high raw bit error rates (RBER). The correction capability of LDPC is highly correlated with its input reliability information. More bit errors can be corrected if the input reliability information is more accurate. However, for data with high RBERs, acquiring high-accuracy input reliability information is time consuming. This is because two adjacent voltage states of memory cells have little noise margin or are even overlapped for data with high RBERs. In this case, LDPC needs to use fine-grained soft sensing scheme to iteratively acquire the reliability information. For example, for flash memory with 2 bits per cell, the maximal number of sensing levels is 21, 7 for each pair of adjacent voltage states. In this work, a new approach is proposed to reduce the decoding latency of LDPC through reducing the number of sensing levels without sacrificing its error correction capability.

Several previous works have been proposed to reduce LDPC sensing levels on flash memory [5], [10], [11]. Dong *et al.* [10] proposed a nonuniform memory sensing strategy. Their design is motivated by the observation that the most overlaps between two adjacent cell states occur closely to the corresponding hard-decision reference voltage (i.e., the boundary of two adjacent states). With this observation, they proposed to sense the overlap regions with a much higher accuracy by placing more sensing levels. Zhao *et al.* [5] proposed to perform read-retry steps by starting from the lowest-level precision, i.e., hard sensing, and then progressively adding extra sensing levels until sucessful decoding. This method outperforms the two-step decoding scheme [10] by not conducting the highest level but the suitable levels when it fails at hard-decision decoding. REAL [11] is designed to improve the input reliability information by taking the bit patterns into account. For example, for a 2-bit cell, if the first bit is 1, the second bit can be 1 with a higher probability. With these information, the accuracy of the input reliability information can be increased. All of these works apply *symmetric* sensing levels between adjacent states, without taking the asymmetric state error characteristics of flash memory cells into consideration.

This work proposes a new sensing level placement scheme to reduce LDPC decoding latency through exploiting error characteristics on flash memory. Specifically, errors of flash memory have three distinctive characteristics on cell states: inter-state *asymmetric*, intra-state *asymmetric* and cross-layer

- *Q. Li, Y. Cui, and C.J. Xue are with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong. E-mail: {qiaoli045, yufeicui92}@gmail.com, jasonxue@cityu.edu.hk.*
- *L. Shi is with the School of Computer Science and Technology, East China Normal University, Shanghai 200062, China. E-mail: shi.liang.hk@gmail.com.*

Manuscript received 20 Feb. 2019; revised 21 Oct. 2019; accepted 11 Nov. 2019. Date of publication 18 Dec. 2019; date of current version 10 Mar. 2020. (Corresponding author: Liang Shi.)
Recommended for acceptance by B. Childers.
Digital Object Identifier no. 10.1109/TC.2019.2959318

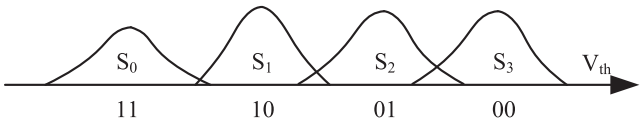


Fig. 1. Illustration of four voltage states for MLC NAND flash memory.

asymmetric. First, inter-state asymmetric errors result from the asymmetric threshold voltage distributions for different flash cell states. For example, for 2-bit per cell flash memory with 4 states, the RBERs of the 4 states present more than 10 times difference [12]. Second, intra-state asymmetric errors result from the asymmetric voltage shifts for each cell state. The voltage of cell states may shift in two directions: left-shift and right-shift, which are induced by different error sources. For example, the dominant error source of left shift is charge leakage over retention time [13], and right shift mainly comes from P/E cycling induced charge trapping [12] and cell-to-cell interference [8]. These two asymmetric errors are common on both planar and 3D NAND flash memory. Third, cross-layer *asymmetric* errors come from the specific structure of 3D flash blocks, where some wordlines have more errors than other wordlines. All the blocks have common cross-layer variation, which means that the indexes of wordlines with more errors are the same in different blocks. These three asymmetric error characteristics are exploited for sensing level placement. For inter-state asymmetric errors, more sensing levels are placed to the pair of adjacent states under more errors and fewer sensing levels to the pair under fewer errors. For intra-state asymmetric errors, more sensing levels will be placed on the left if left-shift is greater than right-shift, and vice versa. For cross-layer asymmetric errors, read operations start with a larger number of sensing levels on the wordlines with more errors and vice versa. In this way, unnecessary sensing levels can be reduced while maintaining the same error correction capability. The major contributions are as follows.

- Proposed an inter-state asymmetric sensing level placement scheme to reduce the sensing levels with awareness of the error characteristics of different states;
- Proposed an intra-state asymmetric sensing level placement scheme to reduce the sensing levels with awareness of the error characteristics of left-shift and right-shift;
- Proposed a cross-layer asymmetric sensing level placement scheme considering the error characteristics of 3D NAND flash blocks;
- Conducted experiments to verify the effectiveness of the proposed techniques. Experimental results show that the proposed approach can achieve great reduction on the decoding latency of LDPC.

The remainder of this paper is organized as follows. Section 2 presents the basics of flash memory and LDPC, as well as the related works. Section 3 presents the problem definition and motivation. The proposed sensing voltage placement scheme is discussed in Section 4. Experiments and analysis are presented in Section 5. Section 6 concludes this work.

2 Background and Related Work

2.1 Basics of NAND Flash Memory

In NAND flash memory, a flash cell uses a transistor to store certain amount of charges. The threshold voltage (V_{th}) of the transistor determines the data values stored in the cell. For the flash memory with n bits per cell, the threshold voltage will be divided into 2^n different voltage states. Fig. 1 shows an example of four voltage states for multi-level cell (MLC) NAND flash memory, which stores two bits per cell, least significant bit (LSB) and most significant bit (MSB). Each one of the four states represents two-bit data and the threshold voltage in one voltage state follows a Gaussian-like distribution.

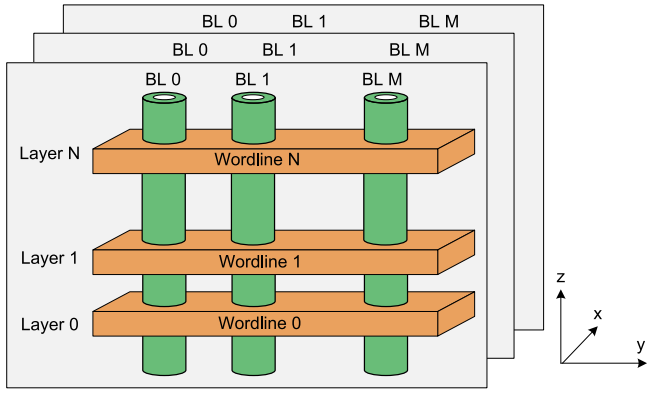


Fig. 2. Cell array of 3D NAND flash memory.

Flash write operations are the process to inject charges into the transistors, to move the threshold voltage of each cell in one wordline into one of the voltage states that are apart from each other with certain noise margin. A high programming voltage is applied on the cells, where the programming voltage will increase monotonically by a step size to increase the threshold voltage of the flash cells. The process will stop when the cells reach the target voltage. Flash read operations are conducted to detect the voltage states of the flash cells in the wordline. One or several reference voltages V_{ref} will be applied on the read wordline to differentiate the voltage states. If the threshold voltage V_{th} of one read cell is larger than the V_{ref}, the sense amplifier in the corresponding bitline will sense that the cell is turned off. To make sure the data is read from the destination cells, a pass-through voltage V_{pass} which is higher than threshold voltage of all the cells is applied on all the unread wordlines in the flash block.

2.2 3D NAND Flash Memory

There are increasing demands for higher density and larger capacity for NAND flash memory. To achieve the goal, planer NAND usually increases the bit density by storing more bits in each flash cell and shrinking the cell to a smaller size, which leads to degraded reliability with more errors. 3D NAND flash memory emerges as a promising solution for the capacity problem by stacking flash cells in the vertical direction.

Fig. 2 shows the structure of a 3D flash block, where a flash block is a three-dimensional array of flash cells. The bitlines of a block are organized vertically, along the z-axis, due to the charge trap transistor design, which is widely adopted in 3D NAND flash memory to replace floating gate cells. In the design, cells from different layers are connected to its neighbors. As a result, all the cells along the z-axis share the same charge trap insulator, which introduces new

error characteristics in 3D NAND [3]. For example, due to the charge trap design, 3D NAND flash memory suffers from early retention loss, where a large amount of charge losses shortly after programming, usually within one hour. Another major difference is the significant reliability variation among layers inside one flash block, which stems from the block architecture.

2.3 NAND Flash Errors

NAND flash memory is prone to errors for several factors, which are presented in the following.

P/E Cycling. Before being programmed, a flash cell has to be erased, by removing all the charges from the floating gate or charge trap. During erase and program operations, a high voltage is performed across the tunnel oxide, which will degrade the strength of charge trapping over time. As a result, charges can go through the tunnel oxide more easily, which causes both charge leakage over retention time and over-programming during programming operations. The flash memory program/erase (P/E) cycling causes damage to the tunnel oxide of floating gate transistors, which directly results in threshold voltage shift and fluctuation [5], [8], [14], [15]. This will gradually introduce errors to flash cells since it gets easier to be affected by different error sources. Cell-to-cell interference, retention time and process variation are three major error sources in NAND flash memory.

Cell-to-Cell Interference. Cell-to-cell interference is the result of coupling effect that programming on one flash cell can increase the V_{th} of neighboring cells [8], [10]. Thus, the threshold voltage state distribution tends to shift to the right. At the early age of flash memory, the insulation strength of the tunnel oxide is strong enough to avoid the V_{th} rising to the overlap region. Only after flash memory has endured certain amount of P/E cycles, cell-to-cell errors become significant. As a result, P/E cycle is used to indicate the errors caused by cell-to-cell interference.

Retention Time Errors. The errors accumulated during retention time are another dominant error source in flash memory [8], [12], [13], [14]. It is caused by charge leakage of programmed flash cells over retention time. Retention errors lead to left-shift errors because the threshold voltage will decline after charge leakage. Retention errors will increase with the increasing of retention time. There exists a new phenomenon in 3D NAND called early retention loss, and the charge leakage rate is very high in the first two hours. Besides, the leakage rate is related to the amount of charges in the transistors. If there are more charges, the leakage is faster, which introduces more errors. As a result, the state being programmed with more charges will suffer more from retention errors.

Process Variation. Process variation is a natural characteristic of semiconductors in NAND flash memory [16], [17], [18], [19]. During the manufacturing process, several memory cell device parameters, like oxide thickness, gate length and width, vary greatly, which results in significant variations of flash cell reliability [20]. Therefore, inherent variation exists among flash cells, where cells will have different RBERs under the same error conditions, for example, under the same P/E cycle and the same retention time. In planer flash, there are no obvious rules that which cells have higher RBERs than others. Exploiting process variation is difficult with high overheads, because it takes great efforts to check the whole flash chip for variation detection. In addition, the information about process variation is supposed to be maintained for management, which requires high overheads in flash systems.

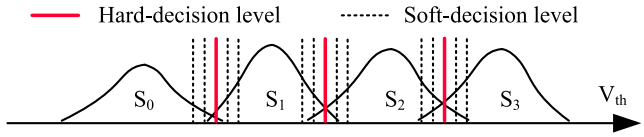


Fig. 3. Illustration of memory sensing for LDPC soft-decision with symmetric sensing voltage placement.

However, in 3D NAND, errors with layer-to-layer variation have some specific characteristics [4]. This is because the variation is the result of the 3D block structure. The electric field of each layer gate has different strengths, which comes from the asymmetric feature process size of vertical channels, i.e., the vertical bitlines in Fig. 2 [21]. Therefore, the layer-to-layer variation is common in different blocks. The WLs with high RBERs in one block have the same index in another block. All the blocks will share the information that which WLs have higher RBER than other wordlines, which can be detected from one block and be maintained for one block.

2.4 Basics of LDPC on Flash Memory

The increasing bit density of multi-level cell and advanced technology scaling on flash memory have raised the requirement for strong ECC, such as BCH and LDPC. The coding schemes for ECC can be grouped into two types: hard-decision codes and soft-decision codes. Hard-decision applies single sensing level between adjacent voltage states in data decoding. The sensing level directly determines the voltage states of the read cells. In contrast, soft-decision applies more sensing levels for obtaining more accurate probability information for decoding input, i.e., the soft information about the threshold voltages of read cells. Hard-decision is always much faster than soft-decision, but has a much weaker error correction capability. In order to achieve a strong error correction capability, LDPC, a soft-decision code scheme, has been recommended for the state-of-the-art flash memories [5], [6].

LDPC decoding is performed by aggressively increasing sensing levels between each pair of adjacent states. It starts with one sensing level between each pair of adjacent states. If the decoding fails, more sensing levels will be iteratively added to each pair. This process stops until the decoding succeeds or the number of sensing levels reaches the maximal value. In each decoding iteration with a certain number of sensing levels (denoted as N), there are two steps: sensing and transferring. First, N sensing levels are symmetrically placed on flash cells to obtain the input probability information. The error correction capability of LDPC highly depends on the accuracy of the input information, where more sensing levels lead to more accurate information. Fig. 3 shows an example on the symmetric sensing level placement for LDPC in flash memory. In the example, five sensing levels are placed between each pair of adjacent states. Second, the sensed information is transferred from flash to LDPC decoder in the controller. Since N sensing levels divide the voltage threshold of the flash cells into

$N+1$ regions, $\lceil log(N+1) \rceil$ bits are needed to represent the information. In Fig. 3, 4 bits will be used to represent the information. As a result, decoding data with high RBER is costly for LDPC, which includes two parts: a long sensing latency with many sensing levels, and a long flash-to-controller information transferring latency with increased information bits.

2.5 Related Work

2.5.1 Flash Memory Sensing Optimization

Several techniques were proposed to optimize memory sensing voltages to improve reliability and read performance. The key is to find the optimal hard sensing voltage to reduce the raw bit error rate seen by ECC. Cai *et al.* [13] and Luo *et al.* [4] observed that the optimal sensing voltages for different wordlines inside one block are close. They proposed to record the optimal sensing voltages of the last wordline, which are updated every day by repeatedly reading the wordline with tuned voltage in fine-grained steps. Peleato *et al.* [22] proposed to dynamically adapt the voltage thresholds using read-back data from progressive reads. Ho *et al.* [23] proposed to count and store the number of cells in each voltage state. During a read operation, the number of cells will be counted and the near-optimal sensing voltages can be found when the measured and the stored numbers are close enough.

2.5.2 LDPC Optimization on Flash Memory

Many studies proposed approaches to optimize LDPC performance on NAND flash memory, which can be categorized into three groups. The first group aims to reduce the RBER of data, thus reducing the read latency. Guo *et al.* [24] proposed to reduce RBER by enlarging noise margins via V_{th} reduction of the data with high LDPC overhead. Liu *et al.* [25] first observed that the majority of errors in reads of the same NAND flash page appear in the same positions until the page is erased. They proposed to store the data which will always get errors in cache and reduce the overall RBER on the data page. AGCR [9] proposed to take advantage of the characteristics that slow-programmed data present low error rates, which requires less sensing levels to gain reliable input information for LDPC decoding and thus leads to shorter read latency. Xie *et al.* [26] were motivated by the fact that lossless compression on flash memory can provide more redundancy for ECC with stronger LDPC decoding strength. They proposed to apply compression on user data and increase the length of ECC parity.

The second group exploits flash error characteristics to optimize LDPC decoding process. REAL [11] incorporated the numerical-correlation characteristic of retention errors into the process of LDPC decoding, to decode data with extra numerical information from the other bit of the same flash cell. Aware of intra-cell data dependence, Sun *et al.* [27] proposed to put the upper page bit and the lower page bit that belong to the same flash cell into one LDPC codeword. The decoding process will be faster by exploiting the additional belief information during the decoding procedure. Aslam *et al.* [28] proposed a novel retention-aware belief-propagation decoding scheme for improving the error rate performance of the flash channel, especially when the memory cells are severely influenced by the retention noise.

The third group strives to conduct the smallest number of sensing levels for successful LDPC decoding. Observing that most overlaps between two adjacent states occur at the boundary of two adjacent states, Dong *et al.* [10] proposed a nonuniform memory sensing strategy to reduce sensing levels on flash memory. LDPC-in-SSD [5] was designed with a read-retry of progressive sensing method to progressively increase the number of sensing levels for stronger decoding capability. Du *et al.* [29] improved the progressive sensing method by starting from the number of sensing levels of last read operation on the data, because the error rate on data will only be increased over retention time. Peleato *et al.* [30] proposed to place the sensing levels based on the results of previous reads, following an optimal policy derived through a dynamic programming backward recursion. However, all of the above works assume that the errors for each state are symmetric, which is asymmetric in fact. Different from all of these works, this work proposes to take the specific asymmetric error characteristics of flash memory into consideration to reduce sensing levels.

2.5.3 Optimization for 3D NAND

Based on the different error characteristics from planer flash, lots of studies have proposed methods to exploit the reliability characteristics for performance and lifetime improvement on 3D NAND flash memory. This subsection mainly introduces the work leveraging layer-to-layer variation for optimization.

Zhu *et al.* [31] observed cross-layer variation of read disturb errors on 3D NAND flash memory. They built a read disturb error model and proposed a location-aware redistribution method to redistribute read-hot pages to locations with less read disturb errors to improve its reliability. Chen *et al.* [32] and Wang *et al.* [33] exploited the asymmetric page access speed feature to boost the access performance by storing data of different hotness in pages with appropriate access speed. By testing on real 3D NAND flash chips, Luo *et al.* [4] found that the average error rate of each 3D-stacked layer in a chip is significantly different. They proposed layer-to-layer variation aware reading scheme by applying different read reference voltages. They further proposed a new RAID structure by grouping pages from different chips and different layers, such that the low-reliability pages are distributed to different RAID groups. Hung *et al.* [34] examined the circuit-level layer variation and proposed to apply different program voltages for each layer to reduce the number of program cycles.

3 Problem Statement and Motivation

In this section, three types of asymmetric errors are presented on 3D NAND flash memory, including inter-state, intra-state and cross-layer asymmetric errors.

3.1 Inter-State Asymmetric Errors

Inter-state asymmetric errors mean that the errors between different pairs of adjacent states are different. Fig. 4 presents a description on the inter-state asymmetric errors. This figure presents the threshold voltage distributions for flash memory with 2 bits per cell. As described in Section 2, all of the cell states can be impacted by several error sources, such as charge leakage, P/E cycling, and cell-to-cell interference [8], [10], [12],

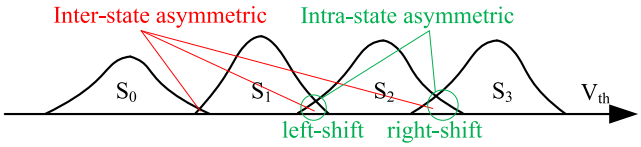


Fig. 4. Inter-state asymmetric errors for different states and intra-state asymmetric errors for one states.

[13]. The reason of inter-state asymmetric errors is that different states contain varied charges and are impacted by the error sources in different degrees. States with more charges tend to lose charges over retention time but they are less likely to increase charges. Therefore, high-voltage states (such as S_2 and S_3) are more vulnerable to left-shift/retention errors but less vulnerable to right-shift errors than low-voltage states (such as S_0 and S_1). Overall, retention errors are the dominant force. Therefore, states S_2 and S_3 are significantly impacted by errors, while S_0 and S_1 are slightly impacted. This kind of asymmetric characteristics has also been identified and tested in previous work [8], [12], [14], [15].

3.2 Intra-State Asymmetric Errors

Intra-state asymmetric errors mean that the errors of a single state are asymmetric in different error directions. There are two error directions for a cell state, voltage left-shift and voltage right-shift. For the voltage left-shift induced errors, the dominant impact factor is retention time [13]. For the voltage right-shift induced errors, the dominant impact factors include the P/E cycling induced charge trapping [8], and cell-to-cell interference induced cell charging [8]. Fig. 4 presents a description for intra-state asymmetric errors. Take state S_2 as an example. Its left-shift and right-shift errors are different and non-deterministic. When retention time is short, left shift could be small and when the P/E cycle is large, right shift could be increased. This kind of asymmetric characteristic has also been identified and tested in previous work [8], [12], [13], [15].

3.3 Cross-Layer Asymmetric Errors

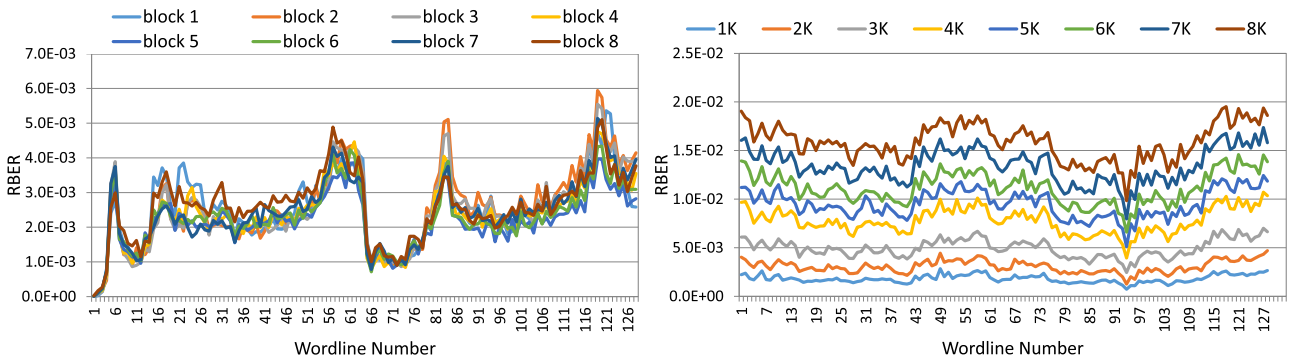
The above two asymmetric errors are common in both planer and 3D NAND flash memories. Cross-layer asymmetric errors only exist in 3D NAND flash memory, which comes from physical structure. Take the vertical channel 3D charge trap NAND flash as an example. There are several gate stack layers and vertical cylindrical channels. Liquid chemicals are applied to erode the gate stack layers and thus the eroded cylindrical channel will have different size openings at different layers. As a result, the RBERs of different layers present variation [4], [32]. The RBERs are collected on two flash chips, a Micron B16A 3D TLC NAND flash chip and a SK Hynix 3D-V4 TLC NAND flash chip. First, Fig. 5a shows the RBER variation across layers of 8 blocks in the Micron flash chip, which has endured 3K P/E cycles. The wordline number in x-axis is normalized to the range from 1 to 128. There are 792 wordlines in each block, and adjacent 12 wordlines share one layer. The page size is 16 KB plus 2,208 bytes spare area. As shown in the figure, the RBER of each layer in one block is significantly different. The RBERs of some layers could be one order of magnitude higher than others. The number of sensing levels required by different layers will vary significantly. Based on existing studies, process variation will increase with the increasing of P/E cycles. Therefore, the requirement for sensing levels can be different in different life stages, which will be studied in the experiment.

Another important observation is that, there are eight randomly selected blocks in the flash chip and they all have the same error variation across layers. The layers with high RBERs and with low RBERs appear in the same positions of different blocks. Therefore, cross-layer asymmetric errors caused from process variation on 3D NAND are consistent among blocks.

Second, we further collect the RBERs on a Hynix flash. There are 576 wordlines in each block, and page size is 8 KB plus 1,024 bytes spare area. Fig. 5b shows the RBERs of different wordlines in block 100 with different number of P/E cycles. Cross-layer asymmetry is less significant compared to the Micron flash chip. However, with the increasing of P/E cycles, the variation increases and when the number of P/E cycle is large, the cross-layer variation is significant.

3.4 Motivation

As described in Section 2, the strong error correction capability of LDPC is realized at the sacrifice of read performance. This is because LDPC has to perform fine-grained sensing with a large number of sensing levels to provide accurate input probability information. The increased number of sensing levels and information bits result in a bad performance. In order to improve LDPC decoding, it is highly desirable to reduce the number of sensing levels.



(a) Different blocks (b) The same block

Fig. 5. Cross-layer asymmetric errors.

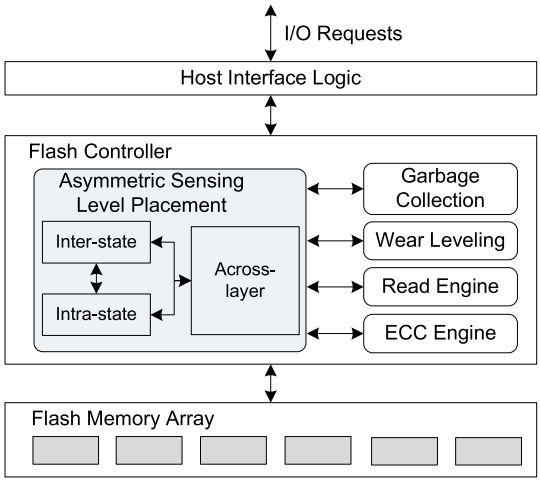


Fig. 6. Overview of asymmetric sensing level placement.

However, it is a challenge that reduced number of sensing levels always induce reduced error correction capability. Existing approaches [5], [10], [11] proposed sensing level placement schemes to reduce them. They proposed to symmetrically place sensing levels to each pair of adjacent states. The number of sensing levels should be able to guarantee the decoding success of the states with the highest error rate. Undoubtedly, the number of sensing levels for the not-worst states can be reduced. On the other hand, existing approaches placed the same number of sensing levels to both the left and right sides of hard-decision in the middle of two adjacent states. This may lead to unnecessary sensing levels on the left for the data with little or no charge leakage, which can be reduced as well, and similarly for the right side. Finally, the layer-to-layer error asymmetry has not been taken into consideration for LDPC decoding optimization. In the following section, three approaches are proposed to reduce unnecessary sensing levels in current design and maintain the same error correction capability as before.

4 LDPC Optimization via Asymmetric Sensing Level Placement

The above three asymmetric error characteristics motivate us to smartly place sensing levels, so as to reduce the number of sensing levels for LDPC decoding. In this section, asymmetric sensing level placement schemes are proposed through exploiting asymmetric error characteristics for LDPC decoding.

4.1 Overview

Fig. 6 shows the overview structure for the proposed schemes. Asymmetric sensing level placement module with three kinds of asymmetric strategies is added in the flash controller. For each read request, the module decides how to place sensing levels on the flash cells, and read engine conducts the read operation based on the placement strategy. Then, the sensed information from the read operation will be transferred to ECC engine for ECC decoding. Besides read requests from the host, read operations are also required during garbage collection and wear leveling. Basically, garbage collection is designed to reclaim space by moving valid pages from victim block to free pages in a clean block and wear leveling is designed to wear blocks evenly by moving pages from hot blocks to cold blocks. For these two components, there are read operations, which are also conducted based on asymmetric sensing level placement.

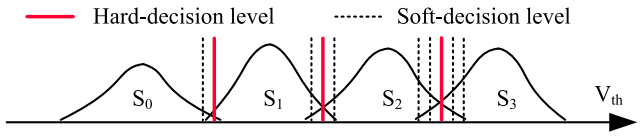


Fig. 7. Illustration of asymmetric sensing level placement with reduced sensing voltages.

Based on previous work [5], which adopts fine-grained progressive sensing, the number of memory sensing levels between each pair of voltages states will be increased by one, when LDPC decoding fails. The process continues until the decoding succeeds, which indicates successful read operations, or until the maximal number of sensing levels, which indicates a read failure. In this case, the read operation starts with a number of sensing levels between two adjacent voltage states, denoted as n_0, to conduct memory sensing. If LDPC decoding fails to correct the errors, one more sensing level will be added to further obtain soft information. Two submodules, inter-state and intra-state asymmetric placement, decide where to place this extra sensing level. While the third submodule, cross-layer asymmetric placement decides the starting number of sensing levels n_0 for each wordline.

In the following, the detailed implementations of these three submodules are presented.

4.2 Inter-State Asymmetric Errors Aware Sensing Level Placement

As stated in Section 3, the errors between each pair of adjacent states differ from each other. In this section, an asymmetric sensing level placement is proposed for different states. For the LDPC in flash memory, assume that N sensing levels are required to decode the data with RBER e. Based on the fact that data with higher RBERs require a stronger error correction capability, which is realized by conducting more sensing levels. In this case, the following relationship is valid.

$$\begin{aligned} &if\ e_k < e_{k+1} \\ &then\ N_k \leq N_{k+1}, \end{aligned} \tag{1}$$

where e_k and N_k respectively are the RBER and the number of sensing levels between states S_k and S_{k+1}. Based on the above relationship, the basic idea of the inter-state asymmetric errors aware sensing level placement scheme is to place sensing levels based on the error rates between two states. If the error rates are high, then more sensing levels are placed, and vice versa. Take Fig. 7 as an example. Assume that the RBER between adjacent states are e_0, e_1, and e_2, which has the following relationship: $e_0 < e_1 < e_2$ [8]. In this case, the numbers of sensing levels for these RBERs satisfy this relationship: $N_0 \leq N_1 \leq N_2$.

Implementation. The implementation of the proposed sensing level placement scheme is similar to previous work [5], [29], where the sensing levels are added iteratively. The sensing levels of last read on the data will first be conducted for reading. If the decoding fails, one more sensing level is

TABLE 1
Existing Sensing Level Adding [5] to Read LSB and MSB Pages

Steps	1	2	3	4	5	6	7
N_0	1	2	3	4	5	6	7
N_1	1	2	3	4	5	6	7
N_2	1	2	3	4	5	6	7

added for more accurate soft information as shown in Table 1. The process ends when the decoding succeeds or the largest number of sensing levels is reached, which is 7 between each pair of states in current flash systems.

We consider two cases to explain the implementation of inter-state asymmetric errors aware placement scheme. First, if LSB and MSB pages are accessed at the same time, for example, sequential read, both pages will be read through placing sensing levels between all three pairs of adjacent voltage states. The sensing levels are preferentially added to the region between S_2 and S_3 if decoding fails. During each iteration, the relationship that $N_0 \leq N_1 \leq N_2$ is maintained. Based on this principle, the following sequences are proposed for sensing level adding. Table 2 shows the detailed steps for the sensing level adding. In this table, the maximal number of sensing levels between each adjacent states is 7, which is the same as previous works [5], [6]. When decoding fails, sensing levels are iteratively increased, until reaching 7 levels between two adjacent states [6]. The difference between previous works [5], [6] and the proposed method lies in the increasing strategy. In case of decoding failure, they will add one more sensing level between every pair of adjacent voltage states, while the proposed asymmetric placement method takes inter-state asymmetric errors into consideration and adds sensing levels preferentially to the pair with high RBER. As shown in Table 2, N_2 is first added up to 7, then N_1 takes the priority for continual increasing, and finally N_0 is added to 7 iteratively.

Second, if only one of LSB and MSB pages is accessed, inter-state asymmetric placement is only applicable for MSB page, because sensing levels are placed only between states S_1 and S_2 for LSB page. To read one MSB page, sensing levels are placed in two regions, the region between S_0 and S_1 and the region between S_2 and S_3. Similarly, the region between S_2 and S_3 has the priority to increase the number of sensing levels in case of decoding failure. In the increasing steps, we maintain the relationship that $N_0 \leq N_2$. Table 3 shows the detailed steps to add sensing levels.

4.3 Intra-State Asymmetric Errors Aware Sensing Level Placement

In the following, intra-state asymmetric aware sensing level placement is proposed. The basic idea is to place more sensing levels to the side with more errors, while fewer sensing levels to the side under fewer errors. N_k sensing levels between states S_k and S_{k+1}, consists of 1 hard-decision sensing level in the middle, N_k^l sensing levels on the left of the hard-sensing level, and N_k^r sensing levels on the right.

TABLE 2
Sensing Level Adding for Inter-State Asymmetric Error Aware Design to Read LSB and MSB Pages

Steps	1	2	3	4	5	6	7	8	9	10	11	12
N_0	1			2			3	4		5	6	7
N_1	1	2		3	4		5	6	7			
N_2	1	2	3	4	5	6	7					

TABLE 3
Sensing Level Adding for Inter-State Asymmetric Error Aware Design to Read MSB Page

Steps	1	2	3	4	5	6	7	8	9	10
N_0	1		2		3		4	5	6	7
N_2	1	2	3	4	5	6	7			

$$N_k = N_k^l + 1 + N_k^r, \tag{2}$$

e_k^l and e_k^r are used to represent the left-shift and right-shift error rates. In the following, two situations are discussed to describe the approach: left-shift aware sensing level placement and right-shift aware sensing level placement.

Left-Shift Aware. When the data have a long retention time on flash memory with light P/E cycling, i.e., at the beginning of flash usage, the data suffers a lot from retention errors and few from cell-to-cell interference errors. The following relationship is satisfied.

$$e_k^l > e_k^r, N_k^l \geq N_k^r. \tag{3}$$

In this case, the threshold voltage distributions mainly shift to left. The left-shift aware scheme is designed to place more sensing levels to the left. Figs. 8a and 8c shows a comparison between left-shift unaware and aware design. As shown in Fig. 8a, left-shift unaware placement places soft-sensing levels in the middle of the two original states before left-shift. As a result, some of the sensing levels on the right are unnecessary because there are fewer errors, while the error region on the left may require more sensing levels for a successful decoding. On the contrary, with the same number of sensing levels, left-shift aware placement in Fig. 8c places more sensing levels on the left. In this case, left-shift aware placement presents a stronger error correction capability over left-shift unaware placement with the same number of sensing levels. Therefore, the number of sensing levels can be reduced for LDPC decoding.

Right-Shift Aware. When the data has a short retention time with heavy P/E cycling, for example, hot data updated frequently, the data mainly suffers from cell-to-cell interference. In this case, the voltage state distributions mainly shift to the right. The following relationship is satisfied.

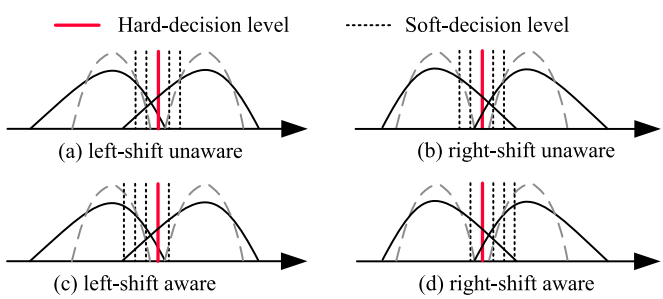


Fig. 8. Comparison between shift unaware sensing level placement and shift error aware voltage placement.

TABLE 4
Sensing Level Adding for Intra-State Asymmetric Errors Aware Design

	P/E Cycles $> T_{PE}$						P/E Cycles $\leq T_{PE}$					
Steps	1	2	3	4	5	6	1	2	3	4	5	6
N_k^l			1		2	3	1	2		3		
N_k^r	1	2		3					1		2	3

$$e_k^r > e_k^l, N_k^r \geq N_k^l. \quad (4)$$

Similar to left-shift aware level placement, the right-shift aware scheme is designed to place more sensing levels to the right. Figs. 8b and 8d show a comparison between right-shift unaware and aware design. Right-shift unaware placement scheme in Fig. 8b places soft-sensing levels in the middle of the two original states before right-shift. On the other hand, right-shift aware placement scheme in Fig. 8d places more sensing levels on the right to provide a stronger error correction capability over right-shift unaware placement scheme with the same number of sensing levels. Therefore, the number of sensing levels can be reduced for the same RBERs.

Implementation. For intra-state asymmetric errors aware placement, the decision has to be made to place sensing levels either on the left or right, 3 sensing levels at most on each side. To achieve this goal, error sources causing left-shift and right-shift are supposed to be known. P/E cycles of flash memory can be easily achieved since wear leveling is implemented in current flash memory controllers. This information is used to indict whether the right-shift errors are significant. A threshold T_{PE} is set for P/E cycles. If P/E cycles are greater than T_{PE}, the right-shift errors are significant. In this case, right side takes the priority over left side to get more sensing levels once decoding fails. This is because no matter the left-shift errors caused by retention time are significant or not, this strategy can reach the objective. On one hand, if left-shift errors are not significant, it is justifiable to give priority to the right side. On the other hand, if left-shift errors are significant, we need to add sensing levels on both sides. Based on the strategy, sensing levels are first added on the right, then sensing levels will be added on the left. Therefore, we still follow the principle to add sensing levels to the regions with high RBERs. Similarly, if P/E cycles are smaller than T_{PE}, left side takes the priority over right side to get more sensing levels once decoding fails. This is justifiable for the similar reason above. Table 4 shows the sensing level adding process.

4.4 Cross-Layer Asymmetric Errors Aware Sensing Level Placement

The above two asymmetric sensing level placement schemes decide how to add sensing levels one by one in case that LDPC decoding fails. This subsection presents the scheme exploiting cross-layer asymmetric errors inside a 3D flash block, which decides the starting number of sensing levels for each wordline.

The two main error sources, P/E cycle and retention time, in a block are similar for different layers. On the one hand, different layers in one block have endured the same number of P/E cycles because the erase unit is a block. On the other hand, the retention time of one layer is very close to another since data will be programmed to one block page by page. Thus, state-of-the-art studies usually start with the same number of sensing levels, n_0, to read data from different layers of a block. Due to cross-layer variation, n_0 can be either larger or smaller than the required number of sensing levels for some layers. The ideal case is to conduct just enough number of sensing levels for each read operation to minimize read latency. Aware of layer-to-layer variation on 3D NAND, this work proposes to start with different numbers of sensing levels on different layers.

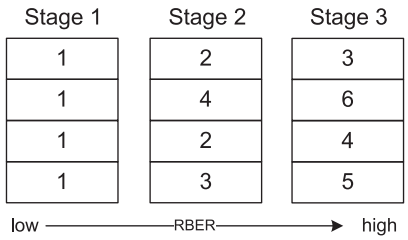


Fig. 9. Illustration of cross-layer asymmetric number of sensing levels due to asymmetric errors. The number means the step number in Table 2, which indicates the number of sensing levels required by each layer to correctly decode data in three error stages. The stage with larger numbers represents the cases with high RBERs, for example, large P/E cycles and long retention time.

Considering inter-state asymmetric errors, we record the step number in Table 2, while Table 3 can be converted into Table 2, for example, Step 8 in Table 3 is step 9 in Table 2. Fig. 9 illustrates an example of cross-layer asymmetric numbers of sensing levels with three stages. The number means the step number in Table 2, which indicates the number of sensing levels required by each layer to correctly decode data in three error stages. In stage 1 with small P/E cycles and short retention time, the RBERs of all the layers are low. Therefore the data on all the layers can be correctly read with only one sensing level between each adjacent state pair, i.e., step 1 in Table 2. By setting n_0 as 1, read operations will be completed with one sensing level in this stage.

However, with the increasing of P/E cycles and retention time, the RBERs increase and the variation will be more significant among layers. The number of sensing levels required by each layer presents variation, as illustrated in stages 2 and 3. With the existence of such variation, if all the wordlines from different layers start with the same number of sensing levels, there are two cases that read latency will be increased. First, the RBER of some layers may require larger number of sensing levels than n_0, and read retry operations will be conducted to obtain more accurate soft information by adding sensing levels one by one. This means multiple read operations with prolonged read latency. For example, in stage 3 of Fig. 9, if n_0 is set as the number in step 3, no read retry is needed to read the data in the first layer. However, to read the data in the second layer, four reads are required to increase the number of sensing levels from step 3 to 6. Second, some layers may have lower RBERs and the sensing levels with smaller number than n_0 can obtain the soft information for successful read. We can correctly read the data with a smaller number of sensing levels. For example, in stage 3 of Fig. 9, if n_0 is set as the number in step 4, the read operation will

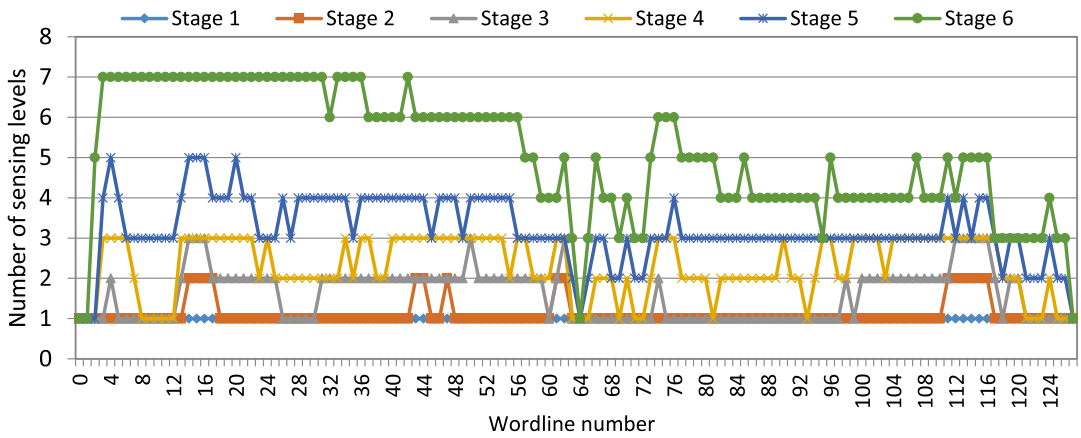


Fig. 10. The number of sensing levels of all the wordlines in 6 different stages.

be processed with sensing levels in step 4 (total 9 sensing levels) while step 3 (total 6 sensing levels) is enough to correctly read the data out.

Therefore, this work proposes to set the starting number of sensing levels according to the required numbers on different layers. Several stages with different ranges of RBERs are maintained. In each stage, the starting number is set as the required number of each layer to avoid above two cases with long read latency.

Implementation. Based on the results in Fig. 5, different blocks have the same cross-layer variation, which comes from the structure characteristics of 3D NAND flash memory. Thus, the variation collected on one block can represent the variation of all the blocks. During the chip manufacturing process, several blocks can be chosen to collect the variation information by programming random data on the selected blocks. The errors will be collected in several error stages under different P/E cycles and different retention time to achieve the variation in different error ranges. For implementation, the errors of all the wordlines inside the chosen blocks will be collected with different error sources. Instead of the accurate RBER values, the corresponding step number, which corresponds to a starting number of sensing levels, will be recorded for reference.

Based on the recorded cross-layer variation table, the starting number of sensing levels for all the wordlines in a block can be achieved if the step number of one wordline inside the block is known. Therefore, when any one wordline of a block is first read, to look up the variation table with the step number and the index of the wordline for the read operation, the stage for the block is achieved, which will be recorded. For the following read operations on the block, the corresponding starting number of sensing levels for any wordline can be achieved according to the stage. If any read operation fails with current stage, more sensing levels will be added. The recorded stage is then updated for future read requests.

4.5 Joint Implementation of Three Asymmetric Placement Methods

Three asymmetric placement methods are introduced separately above. In the following, we present how to combine these three methods together to conduct a read operation.

As shown in Fig. 9, a block will stay in one of the stages based on the RBER of data, for example, 6 stages are maintained on the tested NAND flash as shown in Fig. 10. The stage of each block will be recorded and reset to 1 after erase operations. For each read operation, the stage number will be checked to determine the starting number of sensing level n_0 for each wordline. When one page or one wordline of the block is read, the starting number of sensing level n_0 is looked up. Take Fig. 9 as an example. If one wordline in the third layer is read and the block is in stage 3, the step number is 4. Therefore, based on Table 2, we first place 2 sensing levels between S_0 and S_1, 3 between S_1 and S_2 and 4 between S_2 and S_3. In the meantime, we need to compare the number of P/E cycles with the threshold. Then, the sensing levels will be placed on the left and right based on Table 4. After that, the read operation is conducted on the wordline. If LDPC decoding succeeds with the sensed soft information, the read operation finishes and there is no need to update the stage number recorded for the block. Otherwise, one more sensing level will be added based on Tables 2 and 4 until read successes and the stage number will be updated for the block.

The main overhead of the proposed method comes from the storage cost of all the information. First, the common information which will never be modified for all the blocks includes two parts. The first part is shown in Tables 2, 3, and 4, which requires less than 16 bytes to store. The second part is the variation table for cross-layer asymmetry, where we need to store the step numbers of each layer in each stage. Suppose there are s stages and l layers inside one block and it takes t steps to reach the highest sensing level, which needs $\lceil log(t) \rceil$ bits to represent. The variation information requires $s * l * \lceil log(t) \rceil$ bits to store. As presented in Table 2, there are total 12 steps, which need 4 bits. Use the tested Micron flash chip as an example, there are 6 stages for the tested flash chip and 66 layers. The total size of the variation information will be $6 * 66 * 4$ bits, i.e., 198 bytes. Additionally, for each block, we need to maintain the stage number, which is 3 bits.

The above analysis is based on MLC NAND flash memory, which is used as one example to exploit asymmetric errors in this work. On NAND flash memories with higher density, like TLC, these three types of asymmetric errors also exist. Since there are more voltage states and more sensing levels,

TABLE 5
Asymmetric Errors Aware Sensing Level Placement

Symmetric			Left-shift aware (P/E = 5K)				Right-shift aware (Retention = 1 day)			
N_{total}	bits	level placement	Retention	N_{total}	bits	level placement	P/E	N_{total}	bits	level placement
21	5	(3,3)(3,3)(3,3)	3 years	14	4	(2,0)(3,1)(3,2)	25K	15	4	(1,1)(2,3)(2,3)
18	5	(3,2)(3,2)(3,2)	1 year	12	4	(2,0)(2,1)(3,1)	24K	13	4	(1,1)(2,2)(2,2)
15	4	(2,2)(2,2)(2,2)	6 months	9	4	(1,0)(2,0)(2,1)	22K	9	4	(0,1)(1,1)(1,2)
12	4	(2,1)(2,1)(2,1)	3 months	7	3	(1,0)(1,0)(2,0)	19K	6	3	(0,0)(0,1)(1,1)
9	4	(1,1)(1,1)(1,1)	1 month	5	3	(0,0)(1,0)(1,0)	15K	5	3	(0,0)(0,1)(0,1)
6	3	(1,0)(1,0)(1,0)	1 week	4	3	(0,0)(0,0)(1,0)	10K	4	3	(0,0)(0,0)(0,1)
3	2	(0,0)(0,0)(0,0)	1 day	3	2	(0,0)(0,0)(0,0)	5K	3	2	(0,0)(0,0)(0,0)

the sensing level addition is more complex. Thus, there will be more steps and the overhead will be larger. Nevertheless, the performance improvement will be more significant because asymmetric errors are also more significant.

5 Experimental Analysis

In this section, experiments are conducted to verify the effectiveness of the proposed approach. First, the asymmetric errors aware sensing level placement is presented, which maintains the error correction capability of LDPC with reduced number of sensing levels. Second, the evaluation on simulator is conducted to show the performance improvement due to read latency reduction.

5.1 Asymmetric Sensing Level Placement

Bit error rates are computed from the widely used flash memory error model [15] through probability computation. The model takes three main error sources into consideration, including P/E cycle, retention time and random telegragh noise (RTN). The program operation on flash memory is processed with an iterative program-verify algorithm. In each iteration, the program voltage increases by a step size of ΔV_{pp}. Denote the verify voltage of the kth programmed state as V_p. The threshold voltage of the kth programmed state tends to have a uniform distribution over $[V_p, V_p + \Delta V_{pp}]$.

$$p_p^{(k)}(x) = \begin{cases} \frac{1}{\Delta V_{pp}}, & if v_p^{(k)} \le x \le v_p^{(k)} + \Delta V_{pp} \\ 0, & else \end{cases}, \tag{5}$$

where $p_p^{(k)}(x)$ is the voltage distribution of kth programmed state. This is the ideal voltage distribution for programmed cells without errors. With the impacts of several errors sources, the distribution will fluctuate. The threshold voltage fluctuation $p_{tr}(x)$ introduced by retention time can be formed as a Gaussian distribution.

$$p_{tr}(x) = N(\mu, \delta^2). \tag{6}$$

While it is a symmetric exponential function for RTN-induced fluctuation $p_r(x)$.

$$p_r(x) = \frac{1}{2\lambda_r} e^{-\frac{|x|}{\lambda_r}}. \tag{7}$$

Interface state trap generation grows with the P/E cycling number N_{cyc} in a power law fashion, with the exponent as 0.62, thus $\lambda_r = \alpha N_{cyc}^{0.62}$. Therefore the probability density function of the kth programmed state $p^{(k)}(x)$ can be modeled as

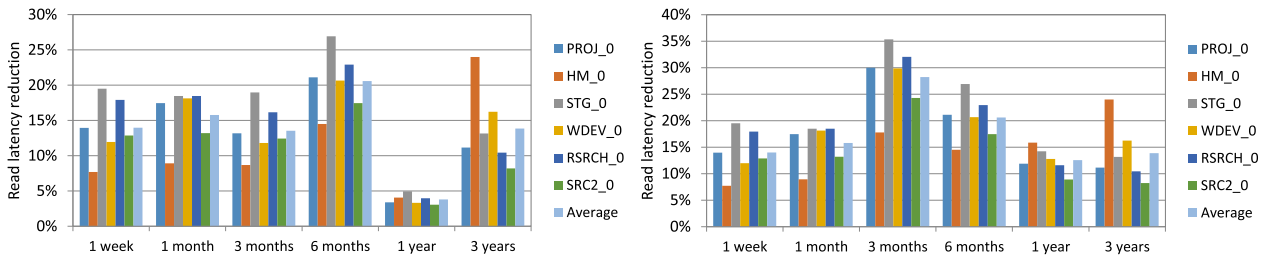
$$p^{(k)}(x) = p_p^{(k)}(x) \otimes p_{tr}(x) \otimes p_r(x). \tag{8}$$

The errors occur when a voltage state overlaps with another.

To correct data with certain RBERs, corresponding number of sensing levels are placed. Table 5 shows the related information, which is produced by the error model [15]. The relationship between the number of sensing levels and RBER has been studied in previous work [5]. In real cases, the RBER of data may vary from the computation value, and therefore we cannot achieve the exact value of RBER. Nevertheless, for implementation inside NAND flash memory, RBER will not be tracked, we only need to record the number of sensing levels, which will be increased iteratively if LDPC decoding fails with current number of sensing levels.

In Table 5, N_{total} represents the total sensing levels of the 3 regions, bits represent the required information bits, and the 3-pair array in the table stands for the placement details $(N_0^l, N_0^r)(N_1^l, N_1^r)(N_2^l, N_2^r)$. For example, when P/E cycle is 5K and retention time is 3 years, the traditional symmetric scheme places 7 sensing levels between each pair of adjacent states, with 3 levels on either side. The proposed approach reduces sensing level to 3 between S_0 and S_1, 5 between S_1 and S_2, and 6 between S_2 and S_3. Thus, the total number of sensing levels N_{total} is reduced from 21 to 14, and the information bit, which represents the soft information of flash cell, is reduced from 5 bits to 4 bits.

We further analyze the performance comparison based on read latency [29]. For read operations with hard-decision level, the data sensing latency is 25 μs and 50 μs for an LSB page and MSB page respectively, while the transfer latency is 20 μs. By adding one more soft sensing level, the sensing latency will be increased by 14 μs, while the transfer latency depends on whether the number of information bits is increased or not. In this case, when reading an MSB page with 5K P/E cycles and 3-year retention time, symmetric scheme places a total of 14 sensing levels (2 hard-decision levels and 12 extra soft-decision levels), and 4 bits are required to differentiate 15 regions divided by 14 sensing levels. The sensing latency is 218 μs, resulting from 50 μs plus 12*14 μs. The transfer latency is 4*20 μs, i.e., 80 μs. Therefore, the total latency is 298 μs. While the proposed asymmetric scheme requires 11 sensing levels (2 hard-decision levels and 9 extra soft-decision levels) and also 4 bits to differentiate 12 regions. The sensing latency is 50 μs plus 9*14 μs, i.e., 176 μs, and the transfer latency is also 80 μs. The total read latency is 256 μs, compared to 298 μs of symmetric method.



(a) Inter-state asymmetric (b) Intra-state asymmetric

Fig. 11. The read latency reduction of inter-state and intra-state asymmetric sensing level placement, under 5K P/E cycles.

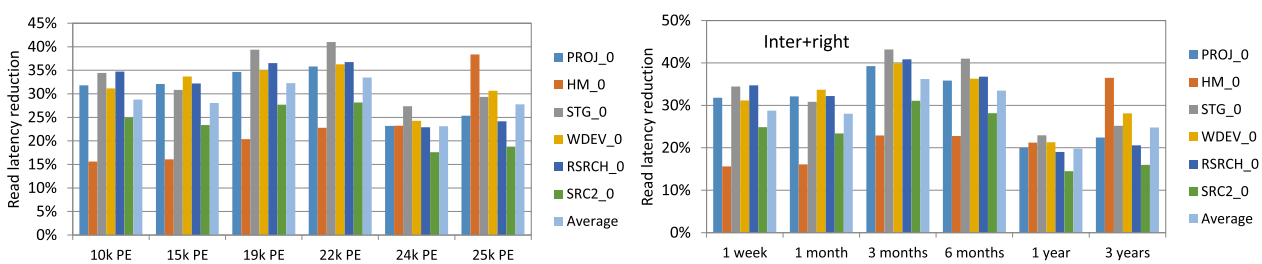
Based on the LDPC decoding capability [5], we separate the RBERs into 6 stages. Since different blocks in a specific NAND flash chip have the same structure and thus have similar cross-layer variation. We studied the RBER increment of different wordlines inside one block. Increasing both P/E cycles and retention time, the number of sensing levels of each wordline is increased by different degrees due to variation. During this process, we record the change of the number of sensing levels, from which 6 stages are achieved. Fig. 10 shows the number of sensing levels of different wordlines in 6 different stages tested on Micron NAND flash chip. The number of wordlines is normalized to 128, where the index is from 0 to 127. y-axis in Fig. 10 is the number of sensing levels required between S_2 and S_3 for successful decoding, which is used to indicate the RBER range. As the figure shows, the variation among layers is significant. In Stage 1, all the wordlines can be accessed with only one sensing levels. With the increasing of stages, which is controlled via changing the setting of P/E cycle and retention time, there are more errors. In the meanwhile, the variation also increases. In Stage 6, the number of sensing levels required for different layers varies from 1 to 7. The figure also presents that the wordlines that need more sensing levels for decoding than others in one stage will also need more sensing levels in all other stages. This further confirms that the reliability variation comes from the structure characteristics, which can be exploited in the whole lifetime of NAND flash memory.

5.2 Performance Improvement

To further quantitatively evaluate the influence on flash access performance, trace-driven simulation is conducted on the SSD module in DiskSim [35] with different real-world workloads from MSR traces [36]. The simulated SSD has 8 channels and each channel has 8 chips, each of which has 2,048 blocks. One block has 64 pages, whose page size is 4 KB. The settings for read latency are the same as Section 5.1. The program latency for a page is 700 μs, and the erase latency of a block is 3,000 μs. In the following, four sets of experiments are conducted to evaluate the effectiveness of the proposed three types of asymmetric errors aware sensing level placement, as well as the combination of the three methods.

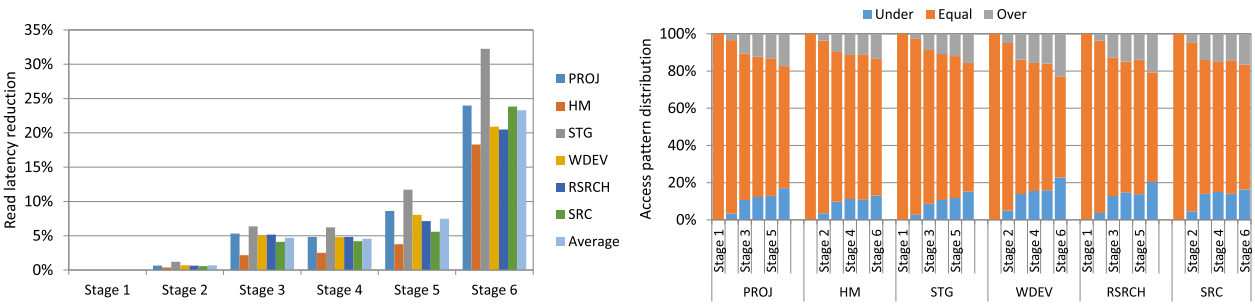
Fig. 11 shows the read latency reduction of inter-state and intra-state asymmetric placement respectively, over current symmetric sensing level placement scheme [5]. The P/E cycle is set as 5K and the retention time varies from 1 week to 3 years to present the cases with different errors. Results show that inter-state and intra-state asymmetric errors aware sensing level placement scheme can gain great read latency reduction by 13.6 and 17.5 percent respectively, on average. The effectiveness is more significant for those read-intensive traces such as WDEV and HM. The latency reduction comes from both reduced sensing latency and reduced transferring latency since the number of sensing levels is reduced.

Fig. 12 shows the read latency reduction of the combinations of inter-state and intra-state asymmetric placement, and intra-state asymmetric awareness includes left-shift and right-shift, by changing the retention time and P/E cycles respectively. The reduction of the combinations is much more significant than Fig. 11. When the retention time is longer for left shift and the P/E cycle is larger for right shift, the improvement of read performance tends to be more significant. This is because more sensing levels are needed to decode data with longer retention time and larger P/E cycle, which presents more potentials for improvement. For left shift, a special case is that when retention time is 1 year, the improvement decreases. Looking back on Table 5, the level reduction from 15 to 9 does not lead to the reduction of information bits, which is still 4 bits. This means only sensing latency is reduced in this case, while both sensing latency and transferring latency are reduced in other long-retention cases. Similarly, this phenomenon also exists in right shift for the case when P/E cycle is 24K.



(a) Inter-state and left-shift asymmetric (b) Inter-state and right-shift asymmetric

Fig. 12. The read latency reduction of inter-state asymmetric sensing level placement with left-shift aware and right-shift aware, respectively.



(a) Cross-layer asymmetric (b) Access pattern distribution

Fig. 13. The read latency reduction of cross-layer asymmetric sensing level placement and the access pattern distribution of original symmetric placement.

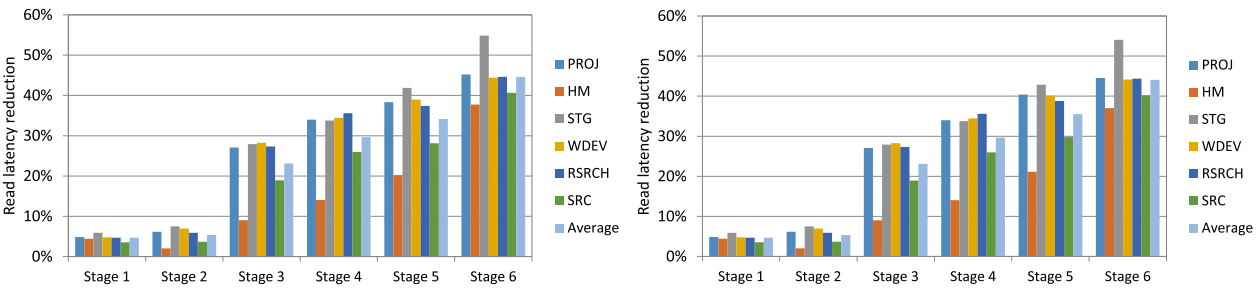
The results for cross-layer asymmetric placement are shown in Fig. 13, where Fig. 13a shows the read latency reduction and Fig. 13b shows the access pattern distribution of traditional symmetric placement. In traditional scheme, the number of sensing levels will be increased iteratively until the data can be correctly decoded. The number of sensing levels of last read that correctly decodes data on the block will be recorded, which will be used as the starting number of sensing levels of current read [29]. There are two cases that the read requests may suffer from longer read latency than the just long enough latency. Due to the existence of cross-layer asymmetric errors, some pages will have more errors than the errors that can be corrected by the recorded number of sensing levels (represented as Under in Fig. 13b). In this case, read retry is required and the number of sensing levels will be increased. Therefore, read latency will be increased. For the second case, some pages may have less errors than the errors that can be corrected by the recorded number of sensing levels, and thus it is not necessary to conduct so many sensing levels (represented as Over in Fig. 13b). These data can be correctly decoded by a smaller number of sensing levels, which incurs shorter read latency. The read latency reduction of the proposed cross-layer comes from the elimination of the prolonged read latency in these two cases.

As shown in Fig. 13a, there is no latency benefit in Stage 1. The read latency reduction increases with the increasing of stage and higher stages present more benefits. Specifically, cross-layer asymmetric placement gains 23.3 percent read latency reduction in Stage 6. The read latency reduction corresponds to the access pattern distribution in Fig. 13b. There are three access patterns: Under, Equal and Over. Under means the recorded number of sensing levels cannot decode the data of current read, equal means the recorded number of sensing level is exactly the one needed by current read, and Over means the recorded number of sensing levels is larger than required. In Stage 1, cross-layer variation is small, all the reads can be satisfied by 1 sensing level, therefore all the patterns are Equal. With the increasing of stage, the variation also increases, which leads to larger variation among pages. As a result, Equal becomes less and the read latency reduction is larger.

Fig. 14 presents the results for combinations of the three asymmetric placement methods. Fig. 14a shows the read latency reduction of combinations of inter-state, left-shift and cross-layer asymmetric placement and Fig. 14b shows that of combinations of inter-state, right-shift and cross-layer asymmetric placement. The read latency reduction also increases with the increasing of stages, from 4.7 to 44.1 percent on average. Overall, the combination of the three schemes presents the most read latency reduction.

6 Summary and Conclusions

The adoption of LDPC on NAND flash memory deals with the degraded reliability, which comes from increased bit density, technology scaling and emerging of 3D NAND. The weakness of LDPC is that long read latency will be introduced to correct data with high RBERs. There have been lots of works to optimize the LDPC decoding strategy based on specific flash characteristics. In this work, three asymmetric error characteristics are studied on flash memory, among which inter-state and intra-state asymmetric errors are common on both planar and 3D NAND flash memory, while cross-layer asymmetric errors are new characteristics on 3D NAND. Motivated by these characteristics, three asymmetric sensing



(a) Inter-state, left-shift and cross-layer asymmetric (b) Inter-state, right-shift and cross-layer asymmetric

Fig. 14. The read latency reduction of the combination of all three asymmetric sensing level placement methods, which is separated into left shift and right shift.

level placement schemes are proposed to place just enough number of sensing levels between the adjacent voltage state pairs. The approaches successfully reduce the unnecessary sensing levels in original symmetric placement method, thus reduce read latency significantly. The proposed approaches are especially effective when RBERs are high. The simulation results show that the proposed approach is effective in reducing read latency by up to 45 percent.

Acknowledgments

This article was supported in part by a Grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CityU 11204718). An earlier version of this article has been accepted in the Proceedings of the 22nd Asia and South Pacific Design Automation Conference (ASP-DAC '17) [1]. The conference paper covers only the reliability improvement on planer flash memory, whereas the new manuscript presents the improvement approach based on the unique error characteristics on 3D NAND.

References

[1] Q. Li, L. Shi, C. J. Xue, Q. Zhuge, and E. H.-M. Sha, "Improving LDPC performance via asymmetric sensing level placement on flash memory," in *Proc. 22nd Asia South Pacific Design Autom. Conf.*, 2017, pp. 560–565.

[2] M. Huang, Z. Liu, L. Qiao, Y. Wang, and Z. Shao, "An endurance-aware metadata allocation strategy for MLC NAND flash memory storage systems," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 35, no. 4, pp. 691–694, Apr. 2016.

[3] Q. Xiong *et al.*, "Characterizing 3D floating gate NAND flash: Observations, analyses, and implications," *ACM Trans. Storage*, vol. 14, no. 2, 2018, Art. no. 16.

[4] Y. Luo, S. Ghose, Y. Cai, E. F. Haratsch, and O. Mutlu, "Improving 3D NAND flash memory lifetime by tolerating early retention loss and process variation," in *Proc. Abstracts ACM Int. Conf. Meas. Model. Comput. Syst.*, 2018, pp. 106–106.

[5] 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.

[6] K.-C. Ho, P.-C. Fang, H.-P. Li, C.-Y. Wang, and H.-C. Chang, "A 45nm 6b/cell charge-trapping flash memory using LDPC-based ECC and drift-immune soft-sensing engine," in *Proc. IEEE Int. Solid-State Circuits Conf. Digest Tech. Papers*, 2013, pp. 222–223.

[7] 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: Regular Papers*, vol. 60, no. 9, pp. 2412–2421, Sep. 2013.

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

[9] Q. Li *et al.*,"Access characteristic guided read and write cost regulation for performance improvement on flash memory," in *Proc. 14th USENIX Conf. File Storage Technol.*, 2016, pp. 125–132.

[10] G. Dong, N. Xie, and T. Zhang, "On the use of soft-decision error-correction codes in nand flash memory," *IEEE Trans. Circuits Syst. I, Regular Papers*, vol. 58, no. 2, pp. 429–439, Feb. 2011.

[11] 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. 32nd Symp. Mass Storage Syst. Technol.*, 2016, pp. 1–13.

[12] C. Gao, L. Shi, K. Wu, C. J. Xue, and E. H.-M. Sha, "Exploit asymmetric error rates of cell states to improve the performance of flash memory storage systems," in *Proc. 32nd IEEE Int. Conf. Comput. Design*, 2014, pp. 202–207.

[13] Y. Cai, Y. Luo, E. F. Haratsch, K. Mai, and O. Mutlu, "Data retention in MLC NAND flash memory: Characterization, optimization, and recovery," in *Proc. IEEE 21st Int. Symp. High Perform. Comput. Architecture*, 2015, pp. 551–563.

[14] R.-S. Liu, C.-L. Yang, and W. Wu, "Optimizing NAND flash-based SSDs via retention relaxation," in *Proc. USENIX Conf. File Storage Technol.*, 2012, pp. 125–138.

[15] Y. Pan, G. Dong, Q. Wu, and T. Zhang, "Quasi-nonvolatile SSD: Trading flash memory nonvolatility to improve storage system performance for enterprise applications," in *Proc. IEEE 18th Int. Symp. High Perform. Comput. Architecture*, 2012, pp. 1–10.

[16] 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. Design*, 2017, pp. 681–684.

[17] 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. 50th ACM/EDAC/IEEE Design Autom. Conf.*, 2013, pp. 1–6.

[18] 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.

[19] 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.

[20] 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.

[21] 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," *Japanese J. Appl. Phys.*, vol. 49, no. 11R, 2010, Art. no. 114202.

[22] B. Peleato, R. Agarwal, J. Cioffi, M. Qin, and P. H. Siegel, "Towards minimizing read time for NAND flash," in *Proc. IEEE Global Commun. Conf.*, 2012, pp. 3219–3224.

[23] K.-C. Ho, P.-C. Fang, H.-P. Li, C.-Y. M. Wang, and H.-C. Chang, "A 45nm 6b/cell charge-trapping flash memory using LDPC-based ECC and drift-immune soft-sensing engine," in *Proc. IEEE Int. Solid-State Circuits Conf. Digest Tech. Papers*, 2013, pp. 222–223.

[24] 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. 52nd ACM/EDAC/IEEE Design Autom. Conf.*, 2015, pp. 1–6.

[25] R.-S. Liu, M.-Y. Chuang, C.-L. Yang, C.-H. Li, K.-C. Ho, and H.-P. Li, "EC-cache: Exploiting error locality to optimize LDPC in NAND flash-based SSDs," in *Proc. 51st ACM/EDAC/IEEE Design Autom. Conf.*, 2014, pp. 1–6.

[26] 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.

[27] H. Sun, W. Zhao, M. Lv, G. Dong, N. Zheng, and T. Zhang, "Exploiting intracell bit-error characteristics to improve min-sum LDPC decoding for MLC NAND flash-based storage in mobile device," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 8, pp. 2654–2664, Aug. 2016.

[28] C. A. Aslam, Y. L. Guan, and K. Cai, "Retention-aware belief-propagation decoding for NAND flash memory," *IEEE Trans. Circuits Syst. II, Express Briefs*, vol. 64, no. 6, pp. 725–729, Jun. 2017.

[29] Y. Du, D. Zou, Q. Li, L. Shi, H. Jin, and C. J. Xue, "LaLDPC: Latency-aware LDPC for read performance improvement of solid state drives," in *Proc. 33th Int. Conf. Massive Storage Syst. Technol.*, 2017, pp. 1–13.

[30] B. Peleato, R. Agarwal, J. M. Cioffi, M. Qin, and P. H. Siegel, "Adaptive read thresholds for NAND flash," *IEEE Trans. Commun.*, vol. 63, no. 9, pp. 3069–3081, Sep. 2015.

[31] Y. Zhu, F. Wu, Q. Xiong, Z. Lu, and C. Xie, "ALARM: A location-aware redistribution method to improve 3D FG NAND flash reliability," in *Proc. Int. Conf. Netw. Architecture Storage*, 2017, pp. 1–10.

[32] S.-H. Chen, Y.-T. Chen, H.-W. Wei, and W.-K. Shih, "Boosting the performance of 3D charge trap NAND flash with asymmetric feature process size characteristic," in *Proc. 54th Annu. Design Autom. Conf.*, 2017, Art. no. 83.

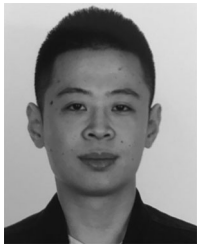
[33] Y. Wang, L. Dong, and R. Mao, "P-alloc: Process-variation tolerant reliability management for 3D charge-trapping flash memory," *ACM Trans. Embedded Comput. Syst.*, vol. 16, no. 5s, 2017, Art. no. 142.

[34] C.-H. Hung *et al.*, "Layer-aware program-and-read schemes for 3D stackable vertical-gate BE-SONOS NAND flash against cross-layer process variations," *IEEE J. Solid-State Circuits*, vol. 50, no. 6, pp. 1491–1501, Jun. 2015.

[35] J. S. Bucy *et al.*, "The DiskSim simulation environment version 3.0 reference manual," Carnegie Mellon Univ., Pittsburgh, PA, Tech. Rep. CMU-PDL-08–101, 2003.
[36] D. Narayanan, E. Thereska, A. Donnelly, S. Elnikety, and A. Rowstron, "Migrating server storage to SSDs: Analysis of tradeoffs," in *Proc. 4th ACM Eur. Conf. Comput. Syst.*, 2009, pp. 145–158.

Qiao Li received the BS and MS degrees in computer science and technology from Chongqing University in China, in 2014 and 2017 respectively. She is currently working toward the PhD degree in the Department of Computer Science, City University of Hong Kong. Her research interests include NAND flash memory, embedded systems, and computer architecture.

Liang Shi received the BS degrees in computer science from the Xi'an University of Post & Telecommunication, Xi'an, Shanxi, China, in July 2008, and the PhD degree from the University of Science and Technology of China, Hefei, China, in June 2013. He is currently a full-time professor with the School of Computer Science and Technology, East China Normal University. His research interests include flash memory, embedded systems, and emerging non-volatile memory technology.



Yufei Cui received the BE degree in telecommunication from Shandong University, Shandong, China, in 2015, and the ME degree in telecommunication from the Hong Kong University of Science and Technology, in 2016. He is currently working toward the PhD degree in the Department of Computer Science, City University of Hong Kong. His research interests include machine learning and embedded system.

Chun Jason Xue received the BS degree in computer science and engineering from the University of Texas at Arlington, Arlington, TX, in 1997, and the MS and PhD degrees in computer science from the University of Texas at Dallas, Richardson, TX, in 2002 and 2007, respectively. He is currently an associate professor 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 co-design, real-time systems, and computer security.