Sentinel Cells Enabled Fast Read for NAND Flash

Qiao Li[§], Min Ye[‡], Yufei Cui[§], Liang Shi[*], Xiaoqiang Li[‡], Chun Jason Xue[§]
[§]*Department of Computer Science, City University of Hong Kong*
[‡]*YEESTOR Microelectronics Co., Ltd*
[*]*School of Computer Science and Software Engineering, East China Normal University*

Abstract

With the shrinking technology size and increasing bits per cell, NAND flash is experiencing increasing number of errors. Error correction code (ECC) is now a key component of flash memory systems. In flash memory, data can be read with different voltages hence with different errors. The read reference voltages are the key factor for RBER seen by ECC. A read failure followed by a read retry happens when errors exceed ECC capacity. Finding the right read voltage with the smallest number of read failures is the key to read performance. The main challenge is that voltages in different cells may drift to different directions by different degrees due to different error sources. Previous methods are designed to either progressively tune the voltage value or empirically predict a read voltage based on error models. This paper proposes a novel approach, by reserving a small set of cells as sentinels, from which the optimal voltage can be inferred, as drifting caused errors exhibit strong locality. Experiments demonstrate the proposed technique is both efficient and effective in finding the optimal read voltage for fast read operations.

1 Introduction

NAND flash memory has been widely adopted as storage devices in mobile devices, embedded systems, and data centers. With the increasing bit density, decreasing technology size, and the introduction of 3D NAND flash memory to stack cells in layers [5, 14, 15], raw bit error rates (RBER) are now ranging from 10^{-3} to 10^{-2}, for today's NAND flash memory [7].

To read data correctly, error correction codes (ECC) are implemented in NAND flash memory [2, 6]. The user data will be stored together with the parity generated from ECC encoding and ECC decoding will be performed for read operations. However, the number of bit errors that can be corrected by ECC is limited. When the number of bit errors exceeds the error correction capability of ECC, a read failure occurs. In this case, read retry is applied together with ECC to improve reliability. After a read failure, the read reference voltages will be tuned followed by a read retry. This is because the voltage states gradually shift over time due to charge leakage or disturbance and tuned read reference voltages can counteract the shift. Though read retry can reduce RBER, it is detrimental to the read performance because the repeated read operations increases read latency significantly.

The performance of read retry highly depends on the voltage tuning strategy. Due to the existence of various error sources on flash memory, the voltage drift of flash cells is difficult to predict, which is the key challenge for voltage tuning. In the literature, there are two main approaches. The first one is to try all the read reference voltages until the read succeeds. For example, Cai et al. [3] proposed to obtain the optimal read reference voltage by repeatedly reading the wordline with tuned voltage in fine-grained step. This method may generate a huge number of read retry operations before a read success. The second is to estimate the state shifts based on prior assumptions and additional information like P/E cycles and retention time, to guide the voltage tuning [8]. However, an accurate estimation of voltage states is challenging, which needs large overheads to record additional information, especially on 3D NAND where there are more types of errors.

This work proposes an efficient and effective voltage tuning approach. The basic idea is to find the optimal read reference voltages on a wordline through reserving a small set of cells as sentinels in the wordline. Through our preliminary study, the voltage state shift for the whole wordline is nearly consistent with that of sentinels when a small set of cells are reserved. Therefore, the errors on a wordline can be predicted from sentinels. By programming known data on sentinel cells, the exact errors are known by comparing the original data and the readout data by default read reference voltage. The optimal read reference voltages can be computed based on the errors for the wordline. Then the read voltages are tuned to the optimal ones for read retry. The tests on real flash chips in the experiments verify that the proposed approach can find the optimal read reference voltage with one read retry operation for most (more than 95%) cases.

2 Background and Related Work

2.1 NAND Flash Memory

A flash cell is a floating gate or a charge trap transistor to represent data by storing certain amount of charges, which determines the threshold voltage (V_{th}) of the cell. Current flash memory stores two or three bits in each cell for density improvement. Figure 1 shows an example of the V_{th} distribution of triple-level cell (TLC) flash. Each TLC cell stores three bits, most significant bit (MSB), center significant bit (CSB) and least significant bit (LSB). The LSB (CSB/MSB) of all cells in one wordline form the LSB (CSB/MSB) page of that wordline.

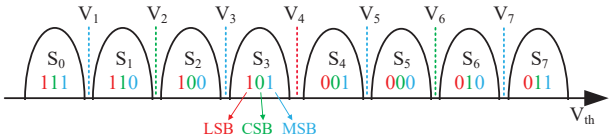


Figure 1: Probability distributions of V_{th} for TLC flash cells.

To read a particular page from a flash block, the controller applies one or several read reference voltages to the control gates of all the cells on the wordline containing the page. As shown in Figure 1, to read LSB page, V_4 will be applied. For all the cells, if $V_{th} < V_4$, the reading bit is 1, otherwise, it is 0. To read CSB page, two voltages, V_2 and V_6 will be applied. If $V_{th} < V_2$ or $V_{th} > V_6$, the reading bit is 1, otherwise, it is 0. To read MSB page, the remaining four read reference voltages are applied to differentiate 1 and 0.

The V_{th} of flash cells can be biased owing to different error sources [8, 11, 12]. For example, program and erase operations wear flash cells and damage the strength of charge trapping over time. Charge leakage over retention time and read disturb will become more severe with the increasing of program/erase (P/E) cycles. Thus, the V_{th} in one state may shift to other states, which introduces errors on the stored data.

To guarantee reliability, error correction codes (ECC) are adopted on NAND flash memory. The user data will be stored together with the parity generated from ECC encoding, where the combination of user data and parity is called one ECC frame. The parity is stored in the out-of-band (OOB) area, which are additional spare bytes within a page. The parity size determines the error correction capability of ECC, which is fixed at design time. When the RBER of data exceeds the error correction capability, read retry is used to reduce RBER.

2.2 Read retry

Through special commands, read retry dynamically adjusts the read reference voltages. Figure 2 presents the read process with read retry. If ECC fails to decode the data, the data read will be retried with the adjusted read voltages. Figure 3 shows an example, using the default read voltage $V_{default}$, the

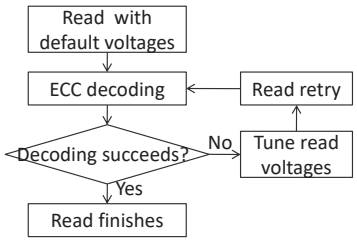


Figure 2: The read process with read retry.

consequent number of errors might exceed error correction capability of ECC. By tuning the read voltage and using V_{opt}, the number of errors can be reduced significantly [3, 6, 8].

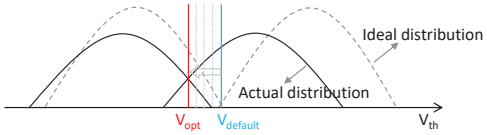


Figure 3: The illustration of read retry with tuned read reference voltages.

There are two main methods to tune the read voltage for read retry. The first method is to try different read reference voltages. Peleato et al. [9] proposed to dynamically adapt the voltage thresholds using read-back data from progressive reads. It requires the assumption that the underlying statistics are Gaussian. Cai et al. [3] observed that the optimal read voltages in the same block are close. They proposed to record the optimal read voltages of the last wordline, which are updated every day by repeatedly reading the wordline with tuned voltage in fine-grained step. The number of trials highly depends on the search strategy, including the tuning direction and step. The read times can be amplified and unpredictable.

The second method is to record some prior information to assist the tuning process. Ho et al. [4] 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 read voltages will be found when the measured and the stored numbers are close enough. Similar to this work, balanced coding is proposed in [10] to encode the programming data such that all the voltage states have the same or comparable number. Luo et al. [8] performed the detailed experimental characterization of major impacting factors on 3D NAND flash reliability, inlcuding P/E cycle, retention time, temperature and dwell time, and developed a model to predict the optimal read reference voltage. However, previous characterization on NAND flash memory suggests that V_{th} distribution of flash cells is complex [2]. It is difficult to accurately predict the optimal read reference voltages based on the impacting factors and it requires lots of storage overheads.

We therefore propose a new read reference voltage tuning approach. The key of the approach is to obtain the accurate estimation on the V_{th} distribution of the read data with reasonable overheads. This work proposes to reserve a subset of

cells on each wordline for this purpose.

3 Preliminary Study

This section studies if it is possible to use the reserved cells to predict the errors on all the cells of one wordline. As shown in Figure 4, $V_i (1 \leq i \leq 7)$ is applied between two neighboring states, S_{i-1} and S_i, to read data. The errors resulting from cells in S_{i-1} being misread as S_i are called *up errors* while those resulting from cells in S_i being misread as S_{i-1} are called *down errors*. Note that only the errors caused by cells shifting to the neighboring states are considered because shifting to faraway states seldom happens. The difference between the number of up errors and the number of down errors indicates the V_{th} transformation. For example, in Figure 4, after left shift of S_{i-1} and S_i, there are more down errors (instances of S_i below the threshold V_i) than up errors (S_{i-1} above V_i).

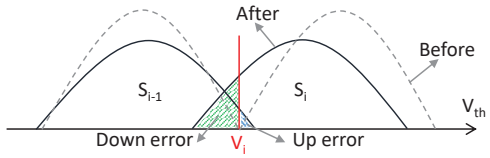


Figure 4: The illustration of up errors and down errors.

We conducted tests on real flash chips to collect the difference between up errors and down errors. For each read reference voltage, a_i is calculated by subtracting down errors from up errors when counting all the cells on the wordline and similarly p_i is the subtraction result when counting the reserved cells (from 0.5% to 3%). Figure 5 shows the percentages of the error difference of reserved cells to all cells, which are obtained through dividing p_i by a_i. Instead of all 576 wordlines inside one block, the data in the figure only covers the first 64 wordlines for a clear presentation and the remaining wordlines have the similar results.

There are two observations from the results. First, the ratio p_i/a_i is very close to the reserving ratio for most wordlines. When 1% cells are reserved, approximately 1% of the error bits on the wordline come from the reserved 1% cells. This means a small part of cells can be used to predict the errors of all the cells on the wordline. One exception is V_1, where the percentages of error difference of reserved cells have a large variance. The reason is that the state on the left of V_1 is the erase state, which has a wider distribution. Since the errors happen near V_1 are the least significant part, the deviation regarding this read voltage is trivial. Second, reserving more cells provides better prediction on the errors of all the cells, but the additional gain diminishes after 1%. On the one hand, the voltage state drift due to error sources including P/E and retention time, is global for all the cells on the wordline. On the other hand, there is local drift for each cell due to random factors. Since global drift is much more significant, a small set of cells can predict the errors on the wordline.

4 Sentinel Cells Enabled Fast Read

4.1 Overview

Based on the observation, this work proposes to reserve a small set of cells, as sentinel cells, for each wordline to find the optimal read reference voltages. During write operations, known data patterns will be programmed on sentinel cells. We program equal numbers of cells to each of the 8 voltage states. In this case, after a read operation, the original voltage state and the readout data of each sentinel cell are known information. The error difference p_i between up errors and down errors of reserved cells can be computed to estimate the errors of all the cells on the wordline. The optimal V_i will be calculated based on p_i and be set on the wordline for read retry. As a result, there is almost always one read retry when a read operation fails.

4.2 Error Calculation

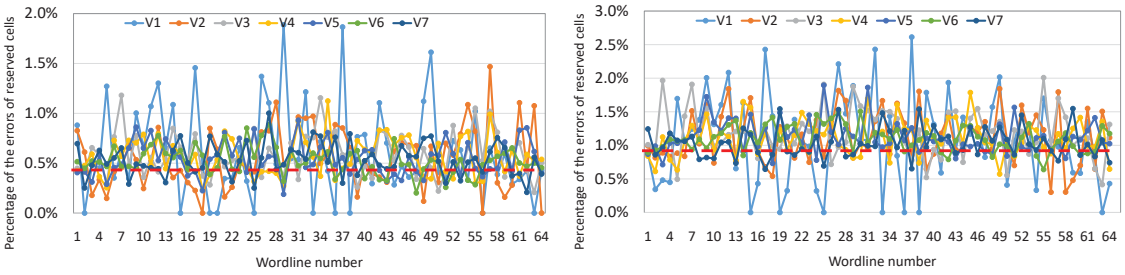
To serve a read request, a set of default read reference voltages is applied on the wordline. The voltage sets are $\{V_4\}$, $\{V_2, V_6\}$ and $\{V_1, V_3, V_5, V_7\}$ for LSB, CSB and MSB page read respectively. If a read fails, we can calculate the value of $p_i (1 \leq i \leq 7)$ based on the original programmed data and readout data. p_i equals to the number of up errors minus the number of down errors regarding V_i for all the sentinel cells.

Table 1 shows how to count the numbers of up errors and down errors. For LSB page, there is only one read referecen voltage V_4. Therefore, the up errors are the cells originally programmed with '1' on LSB page but being misread as '0', similarly for down errors. While for CSB and MSB pages, there are multiple read reference voltages. The calculation is based on the original states and the read bits. A practical assumption is that cells only shift to one of its neighbors. Based on previous studies [2, 13] and our testing on flash chips, flash errors mainly occur in the situation that the V_{th} of cells fall into its neighbor voltage states, either to the left or the right. For example, to calculate p_2, the up errors are S_1 being misread as S_2, and the down errors are S_2 being misread as S_1. Therefore, if the original state is S_1 while the read bit is '0', it is an up error. If the original state is S_2 while the read bit is '1', it is a down error.

4.3 Optimal Voltage Tuning

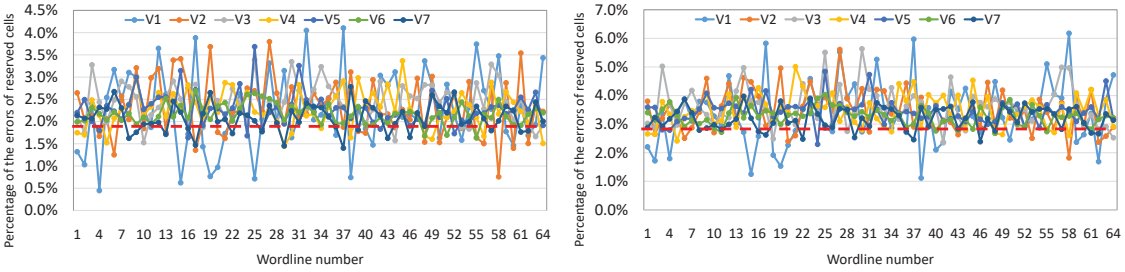
With p_is, we then study the optimal read reference voltages for sentinel cells. By collecting hundreds of pairs of p_i and the offset to the optimal read voltages of sentinels, we fit the relationship between the two, denoted as f_i. The result is presented in detail in the experiments, which is fitted as a polynomial function of degree 5.

Since the optimal read voltages for sentinels and for all cells are very close to each other, V_i is tuned to $V_i + f_i(p_i)$,



(a) Reserving ratio = 0.5%
Average: 0.59%, 0.58%, 0.56%, 0.56%, 0.54%, 0.52%, 0.52%
SD: 0.004, 0.003, 0.002, 0.002, 0.002, 0.001, 0.002

(b) Reserving ratio = 1%
Average: 1.1%, 1.2%, 1.1%, 1.1%, 1.1%, 1.1%, 1.0%
SD: 0.006, 0.004, 0.003, 0.003, 0.003, 0.003, 0.002



(c) Reserving ratio = 2%
Average: 2.3%,2.3%, 2.3%, 2.3%, 2.2%, 2.1%, 2.1%
SD: 0.008, 0.006, 0.005, 0.005, 0.004, 0.003, 0.003

(d) Reserving ratio = 3%
Average: 3.3%, 3.4%, 3.5%, 3.4%, 3.3%, 3.2%, 3.2%
SD: 0.010, 0.007, 0.006, 0.006, 0.005, 0.003, 0.004

Figure 5: The percentage of errors of the reserved cells. The data are calculated through dividing the error difference of all the cells by that of reserved cells. The average and standard deviation (SD) for V_1 to V_7 are given under the sub-captions.

Table 1: Error calculation.

Page	LSB		CSB				MSB							
p_i	p_4		p_2		p_6		p_1		p_3		p_5		p_7	
Original state	1	0	S_1	S_2	S_5	S_6	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7
Read bit	0	1	0	1	1	0	0	1	1	0	0	1	1	0
Error	up	down	up	down	up	down	up	down	up	down	up	down	up	down

which is the predicted optimal read reference voltage for all cells, for read retry. The ECC decoding for this read retry operation has a very high possibility to succeed, based on our testing, which is 95%. However, if the read still fails due to the imperfect fitting of the function, the tuned V_i is close to the optimal read reference voltage. It won't take many steps to further tune V_i to the optimal value using existing methods.

5 Experiments

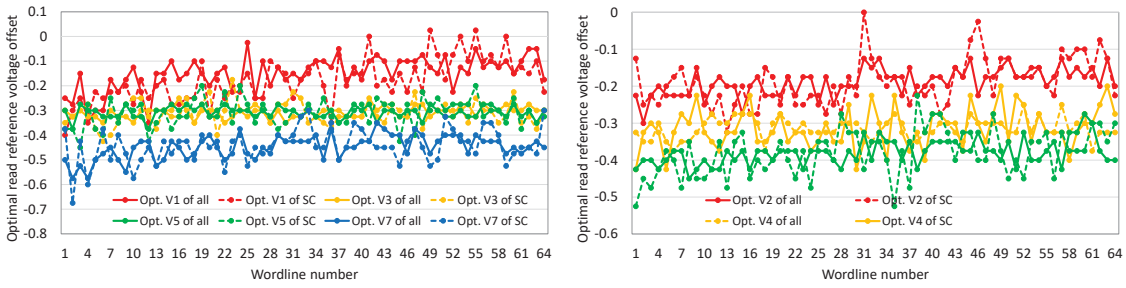
The proposed approach is evaluated on YEESTOR 9083 SSD testing platform [16] with SK Hynix 3D-V4 TLC NAND flash [1]. There are 576 wordlines in each block, and page size is 8KB plus 1024-byte spare area. Random data are programmed on several blocks.The voltage tuning space for each read reference voltage is normalized to the range $[-1.0, 1.0]$. The percentage of sentinel cells is set as 1%, and other settings for this parameter are not discussed in this work due to the page limitation.

5.1 Optimal Read Reference Voltage

Figure 6 shows the offset between the optimal read reference voltages and the default read voltages of all the cells (OPT V_i of all) and of the sentinel cells (OPT V_i of SC). As the results show, the optimal voltages of all the cells and the reserved 1% sentinel cells are very close to each other. We normalize the difference by dividing the difference between the optimal read reference voltages for sentinel cells and for all cells by the tuning scope. From V1 to V7, the normalized differences are 0.0347, 0.0223, 0.0204, 0.0233, 0.0220, 0.0268 and 0.0260. Thus, the optimal read voltage of sentinels are the optimal or near-optimal of all cells.

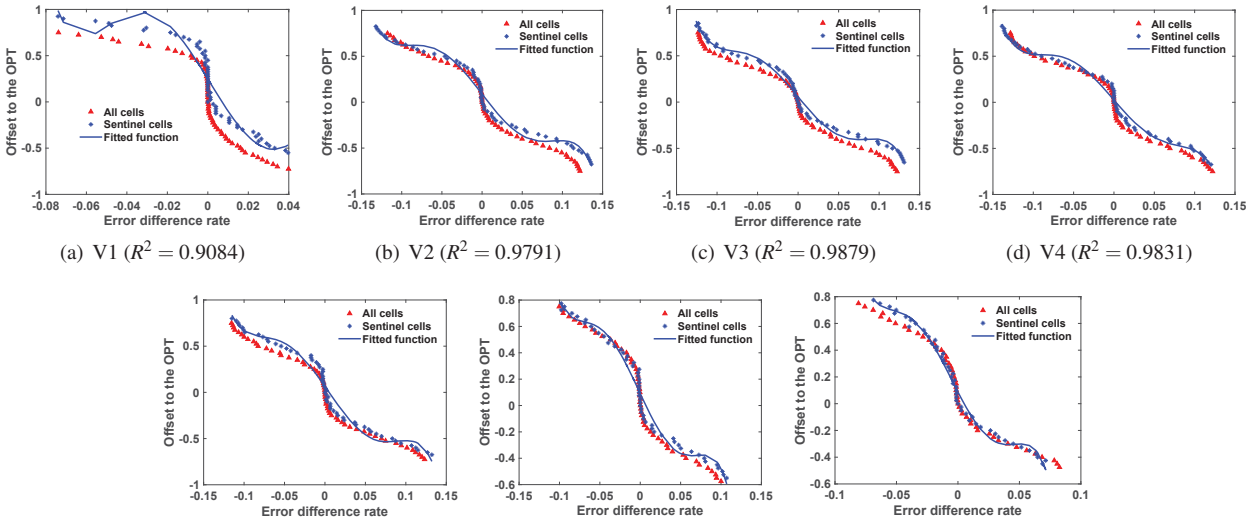
5.2 Offset to the Optimal Voltage

This subsection presents how to predict the tuning voltage offset based on the error difference of sentinel cells p_i. Figure 7 illustrates the relationship between the offset to the optimal read reference voltages and the error difference rate. The error



(a) Optimal offset for V1, V3, V5 and V7 (b) Optimal offset for V2, V4 and V6

Figure 6: The comparison of optimal offset to the default read reference voltages.



(a) V1 ($R^2 = 0.9084$) (b) V2 ($R^2 = 0.9791$) (c) V3 ($R^2 = 0.9879$) (d) V4 ($R^2 = 0.9831$)

(e) V5 ($R^2 = 0.9682$) (f) V6 ($R^2 = 0.9702$) (g) V7 ($R^2 = 0.9865$)

Figure 7: The relationship between the offset to the optimal read reference voltages and the error difference rate. R^2 represents the goodness of fit.

difference rate (x-axis) is calculated by dividing p_i by the total number of sentinel cells. Similarly, the data on x-axis for all the cells in the wordline are calculated in the same way. As we can see in the figure, the offset predicted based on the reserved cells is very close to the actual offset of all the cells. The results are good for most voltages except V_1. We do not suppose this to be a problem because state shifts between S_0 and S_1 are very rare in practice.

By regression analysis, the offset to OPT is fitted as a polynomial function of degree 5, which performs well as the figure shows. After one read operation with the default read voltages, the optimal read reference voltages for a wordline can be predicted based on the error difference ratio of sentinels. Then, one read retry operation with OPT voltages can be conducted. As a result, the number of read retry can be reduced from many to one in most cases, which indicates significant read latency reduction.

6 Conclusion

Due to the increased RBER on NAND flash memory, read retry operations may be required to issue a read request. In the literature, without accurate estimation of the errors, a successful read may need multiple trails to tune the read reference voltages. In this work, we reserve a subset of cells as sentinel cells on each wordline. Pre-determined data patterns will be programmed on sentinels to identify optimal read voltages for the wordline instantly. Real-flash tests show that the approach can achieve the optimal read voltages with a single read retry for more than 95% read requests.

Acknowledgement

We thank our shepherd Prof. Peter Desnoyers and the anonymous reviewers for their comments and suggestions. This work is supported by NSFC 61772092 and 61572411.

7 Discussion

This manuscript presents one simple implementation of the proposed idea. There are several issues and more potentials left to be studied in the future.

First, setting a proper value for the ratio of sentinels is vital to the performance. Sentinel cells come from the ECC parity area, leading to reduced error correction capability of ECC. For the flash tested in this work, the page size is the sum of 8192 bytes for user data and 1024 bytes for spare area, which is used to store ECC parity. If 1% cells are reserved for each wordline, the number of sentinel cells is 96 bytes. Therefore, the ECC parity is reduced to 928 bytes, and the error correction capability is reduced by nearly 10%. Since the proposed work can reduce read retry times significantly, the read disturb errors can be reduced a lot. Therefore, the reduction on the error correction capability can be compensated by the reduced read disturb errors in some degree.

Second, sentinel cells greatly benefit the read operations with high RBER. However, it is not necessary to enable sentinels when the RBER of data is low, like in the early lifetime. Read failures will appear sooner when enabling sentinels because of the reduced error correction capability.

Third, other than each wordline, the sentinels can be set for other granularity, like a flash block. One or more wordlines can be reserved as sentinels for each block. On one hand, all the wordlines inside one block have the same P/E cycles and similar retention time. On the other hand, the sentinel wordlines will not be accessed until a read failure and the layout for other wordlines will not be changed.

Fourth, we only focus on hard-decision error correction in this paper, while the idea can also be applied to soft-decision correction. Since sentinels can provide accurate estimation on the errors, it will be much easier to achieve accurate soft information with small number of read voltages. Therefore, sentinels could greatly benefit the soft decoding performance.

References

[1] Sk hynix 3d-v4 256gb tlc nand flash. https://www.skhynix.com/eng/product/nandUFS.jsp, 2018.

[2] CAI, Y., HARATSCH, E. F., MUTLU, O., AND MAI, K. Threshold voltage distribution in MLC NAND flash memory: Characterization, analysis, and modeling. In *Proceedings of the Conference on Design, Automation and Test in Europe (DATE)* (2013), EDA Consortium, pp. 1285–1290.

[3] CAI, Y., LUO, Y., HARATSCH, E. F., MAI, K., AND MUTLU, O. Data retention in MLC NAND flash memory: Characterization, optimization, and recovery. In *2015 IEEE 21st International Symposium on High Performance Computer Architecture (HPCA)* (2015), IEEE, pp. 551–563.

[4] HO, K.-C., FANG, P.-C., LI, H.-P., WANG, C.-Y. M., AND CHANG, H.-C. A 45nm 6b/cell charge-trapping flash memory using LDPC-based ECC and drift-immune soft-sensing engine. In *IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)* (2013), IEEE, pp. 222–223.

[5] KANG, D., JEONG, W., KIM, C., KIM, D.-H., CHO, Y. S., KANG, K.-T., RYU, J., KANG, K.-M., LEE, S., KIM, W., ET AL. 256 Gb 3 b/cell V-NAND flash memory with 48 stacked wl layers. *IEEE Journal of Solid-State Circuits 52*, 1 (2017), 210–217.

[6] KIM, B. S., CHOI, J., AND MIN, S. L. Design tradeoffs for SSD reliability. In *17th USENIX Conference on File and Storage Technologies (FAST)* (Boston, MA, 2019), USENIX Association, pp. 281–294.

[7] LI, Q., SHI, L., XUE, C. J., WU, K., JI, C., ZHUGE, Q., AND SHA, E. H.-M. Access characteristic guided read and write cost regulation for performance improvement on flash memory. In *14th USENIX Conference on File and Storage Technologies (FAST 16)* (2016), pp. 125–132.

[8] LUO, Y., GHOSE, S., CAI, Y., HARATSCH, E. F., AND MUTLU, O. Heatwatch: Improving 3D NAND flash memory device reliability by exploiting self-recovery and temperature awareness. In *2018 IEEE International Symposium on High Performance Computer Architecture (HPCA)* (2018), IEEE, pp. 504–517.

[9] PELEATO, B., AGARWAL, R., CIOFFI, J., QIN, M., AND SIEGEL, P. H. Towards minimizing read time for NAND flash. In *2012 IEEE Global Communications Conference (GLOBECOM)* (2012), IEEE, pp. 3219–3224.

[10] SALA, F., GABRYS, R., AND DOLECEK, L. Dynamic threshold schemes for multi-level non-volatile memories. *IEEE Transactions on Communications 61*, 7 (2013), 2624–2634.

[11] SCHROEDER, BIANCA, R. L., AND MERCHANT., A. Flash reliability in production: The expected and the unexpected. In *14th USENIX Conference on File and Storage Technologies (FAST 16)* (2016), pp. 67–80.

[12] WILSON, E. H., JUNG, M., AND KANDEMIR, M. T. ZombieNAND: Resurrecting dead NAND flash for improved SSD longevity. In *2014 IEEE 22nd International Symposium on Modelling, Analysis & Simulation of Computer and Telecommunication Systems* (2014), IEEE, pp. 229–238.

[13] WU, F., ZHANG, M., DU, Y., LIU, W., LU, Z., WAN, J., TAN, Z., AND XIE, C. Using error modes aware

LDPC to improve decoding performance of 3D TLC NAND flash. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2019).

[14] WU, F., ZHU, Y., XIONG, Q., LU, Z., ZHOU, Y., KONG, W., AND XIE, C. Characterizing 3D charge trap NAND flash: Observations, analyses and applications. In *2018 IEEE 36th International Conference on Computer Design (ICCD)* (2018), IEEE, pp. 381–388.

[15] XIONG, Q., WU, F., LU, Z., ZHU, Y., ZHOU, Y., CHU, Y., XIE, C., AND HUANG, P. Characterizing 3D floating gate NAND flash. In *2017 ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS)* (2017), ACM, pp. 31–32.

[16] YEESTOR. Ys9083xt/ys9081xt ssd platform. http://www.yeestor.com/index-product-info-id-946-cid-33-pid-3-infopid-33.html, 2018.