Modeling of Threshold Voltage Distribution in 3D NAND Flash Memory

Weihua Liu[†], Fei Wu[†*], Jian Zhou[†], Meng Zhang[†], Chengmo Yang[§], Zhonghai Lu[‡], Yu Wang[†], Changsheng Xie[†]
[†]Wuhan National Laboratory for Optoelectronics, Key Laboratory of Information Storage System,
Engineering Research Center of Data Storage Systems and Technology, Ministry of Education of China,
School of Computer Science and Technology, Huazhong University of Science and Technology, China
[§]University of Delawa, USA
[‡]KTH Royal Institute of Technology, Sweden
[*]Corresponding author: {Fei Wu, wufei@hust.edu.cn}

***Abstract*—3D NAND flash memory faces unprecedented complicated interference than planar NAND flash memory, resulting in more concern regarding reliability and performance. Stronger error correction code (ECC) and adaptive reading strategies are proposed to improve the reliability and performance taking a threshold voltage (V_{th}) distribution model as the backbone. However, the existing modeling methods are challenged to develop such a V_{th} distribution model for 3D NAND flash memory. To facilitate it, in this paper, we propose a machine learning-based modeling method. It employs a neural network taking advantage of the existing modeling methods and fully considers multiple interferences and variations in 3D NAND flash memory. Compared with state-of-the-art models, evaluations demonstrate it is more accurate and efficient for predicting V_{th} distribution.**

***Index Terms*—Modeling, Threshold Voltage Distribution, 3D NAND flash memory, ECC**

I. Introduction

To fulfill the increasing performance and capacity requirements of storage, three-dimension (3D) NAND flash memory-based solid-state drives (SSDs) are becoming the prevailing choice [1]. Meanwhile, the reliability and performance of 3D NAND flash memory are receiving more concern [2] with the continuous scaling of process technology, stacked layers, and increasing storage density. A lot of techniques are proposed to alleviate the reliability and performance problems by optimizing the error correction code (ECC) [3]. However, such techniques need to understand the V_{th} distribution in NAND flash memory deeply [4]. Unfortunately, there is no existing synergistic V_{th} distribution model of 3D NAND flash memory for the following reasons.

On the one hand, the V_{th} distribution is the synergistic result of multiple interferences and variations, such as Program/Erase cycle, data retention time, and read disturbance. They are getting complicated with the evolution of 3D stacking technology. However, most prior studies only give the V_{th} distribution model with a single interference factor [5, 6]. Hence, the existing models are insufficient to understand the V_{th} distribution and optimize the reliability and performance of 3D NAND flash memory. On the other hand, the existing V_{th} distribution modeling methods show several disadvantages in terms of **accuracy** and **universality**. We classify them into two categories: (1) the result-oriented Shape-based Modeling method (**SBM**) and (2) the cause-oriented Variable-based Modeling method (**VBM**). SBM [5, 7, 8] uses the well-known distribution models to fit the real V_{th} distribution shape. It keeps a fantastic modeling universality but fails to bridge the gap between the well-known distribution model and the real V_{th} distribution. VBM [6, 9, 10] constructs the new dedicated V_{th} distribution models taking the interference-related factors as the variables. VBM has the potentiality to obtain excellent accuracy but shows poor modeling universality, which has to renew the model when considering new interference factors. In conclusion, the existing modeling methods have a gap regarding the modeling accuracy and universality.

To overcome the aforementioned challenges in 3D NAND flash memory and bridge the gap between VBM and SBM, in this paper, we propose a machine learning-based V_{th} distribution modeling method. We collect the V_{th} distribution data from the real-world 64-layer 3D NAND flash memory and take advantage of SBM and VBM by employing a neural network taking various interferences as the input features (Variable) to estimate the V_{th} distribution (Shape). The results compared with state-of-the-art models show our model achieves the more accurate prediction result efficiently. The key contributions of this paper are as follows:

- We provide a universal methodology for modeling the V_{th} distribution. It fully considers the synergy of multiple interferences and variations in 3D NAND flash memory, which has not been discussed in prior literature.
- The proposed modeling methodology bridges the gap between SBM and VBM in terms of accuracy and universality. The evaluation results show it efficiently achieves high prediction accuracy.

We give the background of NAND flash memory and discuss the related work in the next section. The modeling methodology is clarified in Section III. Section IV evaluates the proposed model with state-of-the-art models. We conclude this paper in Section V.

978-3-9819263-5-4/DATE21/©2021 EDAA

II. Background and Related Work

A. Basics of NAND Flash Memory

A cell is the minimal storage unit of NAND flash memory. The data can be represented by the number of electrons (voltage) stashed in a cell. For example, a TLC cell stores 3 bits by quantifying the number of electrons into 8 states from "000" to "111", mapped to the logical state from "0" to "7". Without loss of generality, we use the logical state to demonstrate the methodology in this paper. For each state, the number of electrons (voltage) presents a specific distribution within a certain range [11].

There are 3 basic operations to a cell: Erase, Program (Write), and Read. The erase operation sweeps out the electrons from a cell, and the program operation injects electrons into a cell. A wordline (WL) consists of several series-connected cells in a row, and a block contains several WLs stacked in different layers. Any operation will disturb the target WL or its adjacent WLs, resulting in the electrons injection or loss. Hence, the V_{th} distribution shifts dynamically under the synergy of ceaseless multiple interferences. The major interferences include Program/Erase (P/E) cycle, data retention (DR) time, and read disturbance.

B. Interference and Variation in 3D NAND Flash Memory

1) P/E cycle: When executing an erase or write operation, a high-level voltage will be applied to a cell to wipe out or inject electrons. High-level voltage and electron tunneling behavior gradually wear out the oxide layer of a cell [12].

2) Data Retention (DR) Time: Charge loss indicates the electrons diffusing out of the cell as time goes [12]. When the program operation is done, the charge loss rate is getting slow with the passing of DR time.

3) Read Disturbance (RD): When a read operation is issued, all WLs in the same block excluding the target WL will be applied a low-level voltage, which is equivalent to a weak program operation [11].

4) Layer Variation (LV): 3D NAND flash memory introduces the exclusive LV dominated by the process variation among different layers [12]. It extremely depends on the manufacturing technique and stacking architecture and is the main challenging factor in modeling the V_{th} distribution of 3D NAND flash memory.

5) State Variation (SV) and Intra-state Variation (ISV): SV and ISV are mainly caused by the cell which distributes in the high states (or the right end of a state) and the low states (or the left end of a state) having different voltages. ISV leads to a skewed V_{th} distribution.

C. Related Work

SBM models the V_{th} distribution using the well-known distribution functions. Y. Cai et al. [5] fitted the V_{th} distribution using the Gaussian model. The distribution is manipulated by the parameter (μ, σ). To correlate the interference and V_{th} distributions, it maintains a mapping table [2], such as [P/E cycle, (μ, σ)]. SBM achieves a higher universality but sacrifices the accuracy for the skewed real V_{th} distribution. To compensate for the accuracy loss, T. Parnell et al. [7] proposed the shape-based mixture modeling method (SBMM).

On the opposite, VBM models V_{th} distribution starting from the causes [6, 9, 10]. It analyzes and takes the interference, physical mechanisms, and other factors as the variables of a new dedicated model. Theoretically, it could eliminate the gap between the real V_{th} distribution and the proposed model, but it may be failed to unify the modeling method for heterogeneous chip models, structures, and process technology. Note that albeit some VBMs gave the new V_{th} distribution model, they also fitted the V_{th} distribution with the well-known distribution model [10] due to the implementation challenges of newly proposed models. K. Wang et al. modeled the Vth distribution with data retention noise [6]. H. Li [9] gave a VBM taking the real V_{th} distribution as a baseline, and proposed 3 individual models to estimate V_{th} distribution: the program noise and P/E cycle model, the read noise and P/E cycle model, and the data retention noise model. Unfortunately, the inconsistent shifting directions of different stacking layers and the increasing complicated interference make it hard to be applied in 3D NAND flash memory. To our knowledge, a synergistic model for 3D NAND flash memory has not been discussed.

III. Methodology

We select a 3D charge-trap (CT) TLC NAND flash memory to demonstrate the modeling methodology, it is also adaptive to other chip models based on our experiments. The chip has a 64-stack architecture. Each block has 256 WLs (768 pages, 4-WL/layer). Each state has 256 V_{th} steps normalized from -128 to 127, and each step indicates 10 mV, as Fig. 1 exhibits. The chip parameters and test configurations are the same as [11]. We spend more than 2 months to test and collect V_{th} probability density (PD) suffered from various interference permutations. The WL number is used to characterize the stacked-layer [11].

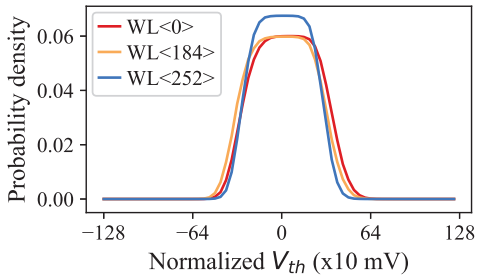

Fig. 1: Layer V_{th} distribution variation.

Fig. 1 exhibits the V_{th} distributions of state 1 in different WLs. A noticeable layer V_{th} distribution variation is observed. As the WL number increases, V_{th} distribution shifts to the left. However, it shrinks in some layers. For example, the V_{th} distribution of WL<252> narrows to the middle and breaks the predictable shift direction from WL<0> to WL<184>. The inconsistent shift directions depend on the evolving stacking structure and the process

technology. Thus, LV becomes the most challenging and indistinct factor in modeling.

To overcome the challenges in 3D NAND flash memory and bridge the gap between SBM and VBM, we propose a new modeling method taking advantage of SBM and VBM. We denote pe, dr, rd, wl as the interference features. To take advantage of accuracy (VBM), we select these interference features as the model variables. It can be expressed as follow:

$$x = (pe, dr, rd, wl)^T$$

To maintain a high modeling universality, we shape the V_{th} distribution via a V_{th} distribution PD vector:

$$y = \left(pd^{(1)}, pd^{(2)}, \ldots, pd^{(M)}\right)^T$$

$pd^{(i)}$ is the i-th feature of y. The goal of the modeling method is to find a function f such that $y = f(x)$.

SBM fits y directly to approximate f (from y to f), while VBM formulates a new f to fit y (from f to y). However, they sacrifice the accuracy and universality in the fitting or approximating process. In this paper, we probe to approach such function f automatically. To achieve this, we employ a neural network (NN) to correlate y and x. We set the individual NN model for each state to avoid the large-scale output vector and reduce the complexity.

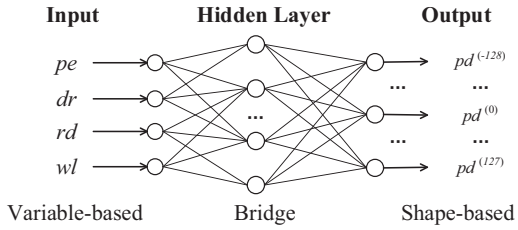

Fig. 2: The neural network model

The individual NN model is delineated in Fig. 2. It consists of 3 neuron layers: the input layer (x), one hidden layer (h), and the output layer (y). The input layer accepts the fed x from a scrambled training data stream. If the model does not consider someone interference, such as RD, the training data can feed 0 to rd. If there are some new features in future NAND flash architecture, it could also extend the model to accept new features by adding the new input neurons. The full connection is deployed among 3 neuron layers. The hidden layer implements a nonlinear activation function, Rectified Linear Unit (ReLU), and the output layer implements the linear function. The quadratic loss function and the random weight initiation are selected in model training empirically.

The max number of pd neurons relates to the V_{th} scan density. Theoretically, the more output neurons are implemented, the more accurate the model is. However, more neurons will increase the complexity. The neuron number in the hidden layer depends on the input and the output layer. There is no theoretical reference to guide how many neurons are appropriate. We tune it empirically and test to find the proper neuron number. We also evaluate the computation complexity of pd neurons number in the next section.

IV. Evaluation

In this section, we evaluate the accuracy and overhead of the proposed NN model. We use $D(y, f(x))$ as the metric of the average V_{th} distribution deviation between predicted and measured data. The definition of $D(y, f(x))$ is:

$$D(y, f(x)) = |y - f(x)| = \frac{1}{M}\sum_{j=1}^{M}\left|y^{(j)} - \hat{y}^{(j)}\right|$$

A. The Generality of NN Model

We randomly select 70% of collected data as the training set, and the rest 30% is the test set. In this evaluation, the NN model is implemented with 96 neurons in the hidden layer and 64 output neurons because the interval of collected V_{th} PD is 4 from -128 to 124. We train it and predict the V_{th} distribution with arbitrary interference combinations within the trained range where $pe <= 7000$, $dr <= 90$, and $rd <= 4000$. The predicted results of all states are compared with the measured data. The average $D(y, f(x))$ is only $\mathbf{2.6 \times 10^{-4}}$. Taking $(pe, dr, rd, wl) = (4000, 7, 1000, 154)$ as an example, the evaluation results are depicted in Fig. 3. Moreover, we also probe to validate its generalization ability of the worst case in predicting the uncovered training area where $pe > 7000$, $dr > 90$, and $rd > 4000$. The average $D(y, f(x))$ of predicting the uncovered area is $\mathbf{1.6 \times 10^{-3}}$.

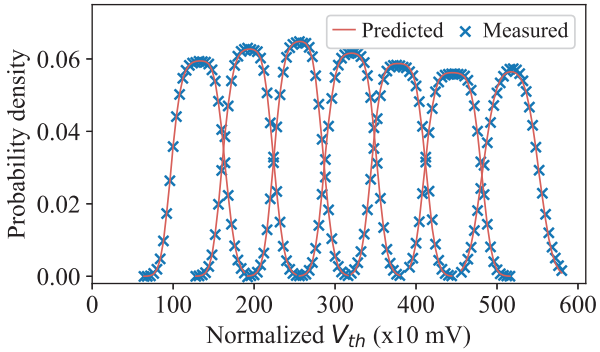

Fig. 3: Prediction exemplar, $x = (4000, 7, 1000, 154)^T$.

B. Comparison with State-of-the-art Model

B.1. Prediction Accuracy

To further the evaluation, we make a detailed comparison between the proposed model and state-of-the-art models. The Monte Carlo modeling method [9] is selected as the representative of VBM. We evaluate the prediction accuracy of various models: Gaussian (SBM), Gaussian Mixture (SBMM), Monte Carlo (VBM), and NN model. The evaluation runs on a Ubuntu 18.04 server equipping an i5-8400@2.4GHz CPU and 16GiB-DDR4 DRAM. The main software in evaluation includes Python 3.7.3, Scipy 1.4.1, and Tensorflow 1.14.0. We implement 96 neurons in the hidden layer and 64 output neurons in the NN model, and all data in the training set with various features' permutations are trained apart from pe in {8000, 9000, 10000}.

Considering most existing models are proposed with only a single disturbance, to fairly compare, we evaluate the

prediction accuracy of V_{th} distribution with P/E cycle. The Gaussian model, Gaussian Mixture model, and NN model are trained with the data from pe = 1 to pe = 7000. Besides, to predict the out-of-trained-range V_{th} distribution, we model the parameters of the Gaussian and Gaussian Mixture model as [4] did.

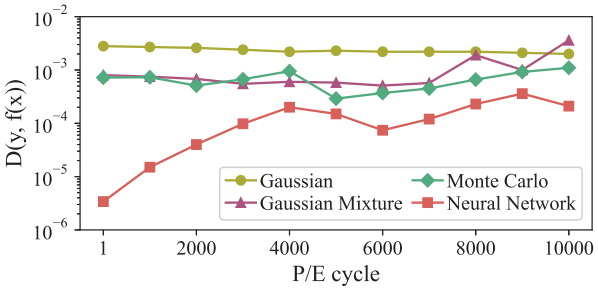

Fig. 4: Prediction accuracy

We count the average $D(y, f(x))$ of 4 models and exhibit the results in Fig. 4. The vertical axis is plotted on the logarithmic scale. Evaluation results demonstrate that the NN model achieves the highest prediction accuracy in all cases. The Monte Carlo model also shows excellent prediction results even for the out-of-trained-range data. The Gaussian Mixture model has a considerable prediction accuracy within the trained range, but it declines with the increase of out-of-trained-range P/E cycles. The Gaussian model has the worst prediction accuracy.

B.2. Overhead

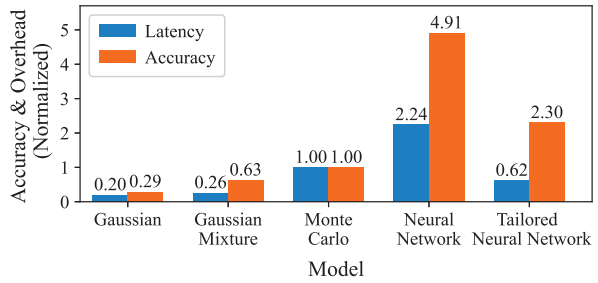

Fig. 5: Overhead and accuracy evaluations

We then evaluate the efficiency and overhead of them. We run all models with only CPU computation and count the CPU time (I/O and sleep time excluded) for predicting. To evaluate the overhead of the NN scale, we tailor a NN to a 48 neurons' hidden layer and 32 output neurons. Fig. 5 gives the results taking the Monte Carlo model as a baseline. The Gaussian model and Gaussian Mixture model show the lower latency with low prediction accuracy. Albeit the NN model spends 2.24x computing time, it enables the model to achieve 4.91x prediction accuracy. The higher accuracy-latency ratio also demonstrates the NN is more efficient than the Monte Carlo model. Moreover, the tailored NN only consumes 0.62x computation latency while achieving 2.3x prediction accuracy. Note that the model training process is offline in a server or dedicated machine. The model we proposed in this paper is a shallow NN, and it also will not introduce extra training overhead for SSD.

V. Conclusion

In this paper, we discussed the advantages and disadvantages of the two prevailing V_{th} distribution modeling methods. We then proposed a universal method to model the V_{th} distribution in 3D NAND flash memory considering multiple interferences and variations. Our approach bridged the gap between two prevailing modeling methods, and it did not depend on specific flash structures and nanotechnology. The evaluation results showed it enables accurate V_{th} distribution prediction and strong generalization ability. It can guide to model the V_{th} distribution with various NAND flash models and help to optimize the ECC in the future.

Acknowledgment

This work was supported in part by Key-Area Research and Development Program of Guangdong Province No.2019B010107001, in part by the NSFC under Grant No.61821003, No.61872413, No.U1709220, No.61902137, in part by National Key Research and Development Program of China No.2018YFB1003305, No.2018YFA0701800, in part by the 111 Project (No.B07038), in part by the Project funded by China Postdoctoral Science Foundation.

References

[1] W. Liu, F. Wu, M. Zhang, C. Yang, Z. Lu, J. Wan, and C. Xie, "Deps: Exploiting a dynamic error prechecking scheme to improve the read performance of ssd," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2020.

[2] Y. Cai, S. Ghose, E. F. Haratsch, Y. Luo, and O. Mutlu, "Error characterization, mitigation, and recovery in flash-memory-based solid-state drives," *Proceedings of the IEEE*, vol. 105, no. 9, pp. 1666–1704, 2017.

[3] M. Zhang, F. Wu, Q. Yu, W. Liu, Y. Wang, and C. Xie, "Exploiting error characteristic to optimize read voltage for 3-d nand flash memory," *IEEE Transactions on Electron Devices*, vol. 67, no. 12, pp. 5490–5496, 2020.

[4] Y. Luo, S. Ghose, Y. Cai, E. F. Haratsch, and O. Mutlu, "Enabling accurate and practical online flash channel modeling for modern mlc nand flash memory," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 9, pp. 2294–2311, 2016.

[5] Y. Cai, E. F. Haratsch, O. Mutlu, and K. Mai, "Threshold voltage distribution in mlc nand flash memory: Characterization, analysis, and modeling," in *Proceedings of the Conference on Design, Automation and Test in Europe*, pp. 1285–1290, 2013.

[6] K. Wang, G. Du, Z. Lun, and X. Liu, "Investigation of retention noise for 3-d tlc nand flash memory," *IEEE Journal of the Electron Devices Society*, vol. 7, pp. 150–157, 2018.

[7] T. Parnell, N. Papandreou, T. Mittelholzer, and H. Pozidis, "Modelling of the threshold voltage distributions of sub-20nm nand flash memory," in *2014 IEEE Global Communications Conference*, pp. 2351–2356, IEEE, 2014.

[8] S. Suzuki, Y. Deguchi, T. Nakamura, and K. Takeuchi, "Endurance-based dynamic v th distribution shaping of 3d-tlc nand flash memories to suppress both lateral charge migration and vertical charge de-trap and increase data-retention time by 2.7 x," in *2018 48th European Solid-State Device Research Conference (ESSDERC)*, pp. 150–153, IEEE, 2018.

[9] H. Li, "Modeling of threshold voltage distribution in nand flash memory: A monte carlo method," *IEEE Transactions on Electron Devices*, vol. 63, no. 9, pp. 3527–3532, 2016.

[10] H. Yassine, J. Coon, M. Ismail, and H. Fletcher, "Towards an analytical model of nand flash memory and the impact on channel decoding," in *2016 IEEE International Conference on Communications (ICC)*, pp. 1–6, IEEE, 2016.

[11] W. Liu, F. Wu, M. Zhang, Y. Wang, Z. Lu, X. Lu, and C. Xie, "Characterizing the reliability and threshold voltage shifting of 3d charge trap nand flash," in *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 312–315, IEEE, 2019.

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