Machine learning model for predicting threshold voltage by taper angle variation and word line position in 3D NAND flash memory

Dong Chan Lee, Jang Kyu Lee, and Hyungcheol Shin

Abstract
In this letter, a machine learning (ML) model is presented to predict the variation of the threshold voltage (V\text{th}) according to the taper angle and target word line (WLT) position in 3D NAND flash memory. Through Technology Computer-Aided Design (TCAD) simulation, V\text{th} is extracted according to taper angle and WLT position. TCAD data is used as the training data set required for learning by an artificial neural network algorithm (NNA). The completed ML model is then used to predict V\text{th} for each word line (WL). It was also confirmed that the ML model predicted well even for TCAD data that was not used as a training data set.

Keywords: 3D NAND flash memory, taper angle, artificial neural network algorithm, machine learning

1. Introduction

In NAND Flash memory, a 3D vertical structure was introduced to reduce bit cost and for high bit density [1, 2, 3]. In the vertical structure, a higher bit density was pursued by increasing the number of word line (WL) layers [4]. However, the threshold voltage (V\text{th}) variation becomes severe as the number of WL layers increased. As a result, WL loading variation occurs according to target word line (WLT) position, and the program time should be increased in order to set the same V\text{th} value for each WL in program operation [5]. In addition, taper angle, which occurs in the etch process for string hole fabrication, change the critical dimension according to the WLT position and caused variation in V\text{th} [6, 7]. So, even in the same WL position, the V\text{th} characteristics vary depending on the taper angle.

As the number of these variability sources increases, they must be modeled in consideration as various parameters. However, it is difficult and time-consuming to extract various parameters in highly scaled down devices [8, 9]. Machine learning (ML) is a good alternative to solving these problems. ML models are created through neural network algorithms (NNA) that analyze the relationship between inputs and outputs [10, 11, 12, 13, 14]. Its modeling process is less time consuming and shows high accuracy [15, 16]. Also, since modeling for multiple inputs is possible, ML is useful for predicting the variation issue caused by the variability sources [17].

In this paper, ML model is proposed to predicts the V\text{th} variation caused by the taper angle and WLT position in 3D NAND Flash memory. The results of the Technology Computer-Aided Design (TCAD) simulation are used as the training data set of NNA to implement the ML model. We also verify the accuracy of ML models with test data set, and show that ML models are completed in a less time.

2. Device details and simulation setup

Fig. 1 shows a schematic cross-section view of 3D NAND Flash memory string with a taper angle. In this work, the taper angle is defined by the angle between the blocking oxide and vertical axis. It is assumed that the taper angle follows a Gaussian distribution with a mean of 0.1° and a variance of 0.03°. When the taper angle was set, the radius of the top filler oxide was fixed at 15 nm. The thickness of the channel, tunneling oxide, nitride, and blocking oxide are 10, 5, 5, and 6 nm, respectively. The WL length and spacer length are 28 nm. The string consists of 96 WLs, drain selected line (DSL), and source selected line (SSL). The top cell (WL96) and bottom cell (WL1) are used as dummy cells. 2D-simulation is carried out reflecting the cylindrical coordinates. The voltage of the bit-line (BL) is set to 0.7 V, and that of the source-line (SL) is set to 0 V.
Fig. 2 I$_{BL}$-V$_{WL}$ curves according to the WLT position (WL2, WL48 and WL95) and taper angle.

Table I V$_{th}$ result according to taper angle and WLT position.

| Taper Angle | Threshold Voltage [mV] |
|-------------|------------------------|
| WL2 (bottom)| 475.74                 |
| WL48 (middle)| 465.6                  |
| WL95 (top)  | 449.8                  |

The voltage of DSL, SSL, and unselected WL are set to 6V. The threshold voltage (V$_{th}$) is defined as the voltage when a 10nA bit line current occurs by applying a constant current method [18]. The simulation is performed only for neutral state where electrons do not exist in the nitride layer.

3. Characteristics of taper angle

Fig. 2 shows bit-line currents (I$_{BL}$) in the neutral state as a function of WL voltage (V$_{WL}$) in the case of 0° and 0.13°. In tapered devices, the critical dimension of the channel decreases, resulting in a decrease in I$_{BL}$. Table I shows the V$_{th}$ according to the taper angle and WLT position. When the taper angle is 0°, the V$_{th}$ decreases as the WLT is located below. Because as the WLT is located below, the source resistance decreases [19]. On the other hand, if the taper angle exists, the lower cell has higher V$_{th}$. Because the radius of the channel decreases due to the taper angle and the V$_{th}$ roll-off is mitigated [20, 21, 22].

4. Machine learning model

Fig. 3(a) shows the input training data configuration extracted from TCAD simulation for the training ML. WLT numbers of multiples of 8 are selected except WL2 and WL95. Then, V$_{th}$ is extracted by setting 100 random taper angles for each WLT. Fig. 3(b) shows an NNA for ML. The algorithm consists of input layer, hidden layer, and output layer. The input layer contains two nodes: WLT number ($X_1$) and taper angle ($X_2$). There are three hidden layers and each layer has 30 nodes. The output layer has one node that is assigned the V$_{th}$ in this work. The activation function used in the algorithm is the Rectified Linear Unit (ReLU), and defined by the following equation [23, 24, 25]:

$$ f = \begin{cases} f(x) = 0 & (x < 0) \\ f(x) = x & (x \geq 0) \end{cases} $$

(1)

The output of each node (neuron), $e$, is defined by the following equation:

$$ e = f(W \cdot X + B) $$

(2)

Where $W$ and $B$ denote weights and bias. Weight and bias are initially assigned random values. The final output value of the NNA, $Y$, is determined by the operation of each node and is defined by the following equation:

$$ Y = W^O \cdot f_{i,o} \left( W^{H2} \times f_{o,i} \left( W^{H1} \times f_{a,i} \left( W^I \times X + B^I \right) + B^{hidden_1} \right) + B^{hidden_2} \right) + B^out $$

(3)

In training, the weights and biases are changed by the gradient descent method during the back-propagation process to reduce error rate [26, 27, 28, 29]. Fig. 4 shows the algorithm for how to complete the ML model. Training results are compared with the V$_{th}$ value extracted through TCAD simulation to find the error, and the training is repeated until the target error rate is reached. NNA was trained by Pytorch.
python library [30]. Fig. 5 shows the training process in which the error rate decreases as the epoch increases. As a result of setting the target error rate to 0.3%, about 50,000 repetitive learnings are performed and the final error rate is 0.28%.

5. Results

5.1 Modeling results

Fig. 6 shows the results of ML training for taper angles and WLT position. The data obtained through TCAD simulation is used as training data set for NNA. The ML model follows the trend of increasing $V_{th}$ in proportion to the taper angle. The effect of reducing the critical dimension by the taper angle differs according to the WLT position, and thus the $V_{th}$ variation range differs according to the WLT position. In the case of the ML model, even this trend is well followed.

5.2 Test results for taper angle

After modeling through NNA, the ML model is verified by 1000 TCAD test data generated in random taper angles. The data obtained through TCAD simulation is used as training data set for NNA. The ML model predicts the $V_{th}$ test data according to the taper angle. Fig. 7(b) shows $V_{th}$ distribution based on the results of (a). The histogram represents the number of cells within a certain $V_{th}$ interval, and the line is a Gaussian distribution using the mean and standard deviation of the $V_{th}$ of the target angle.

Table II Comparing $V_{th}$ distribution predicted through ML and TCAD simulation.

|               | Mean $V_{th}$ (mV) | Standard deviation $V_{th}$ (mV) | Error rate (%) |
|---------------|--------------------|----------------------------------|----------------|
| TCAD          | ML                 |                                  |                |
| WL2           | -314.20            | -314.52                          | 0.1            |
| WL48          | -385.93            | -386.71                          | 0.19           |
| WL95          | -439.29            | -439.02                          | 0.06           |

Fig. 7 Results of applying 1000 random test data (taper angle) to WL2, WL48 and WL95. (a) Comparison of predicted $V_{th}$ by TCAD simulation and ML model. (b) $V_{th}$ distribution based on the results of (a)

5.3 Test results for WLT position

Because WLT position is also used as input training data, the model test related to WLT position is also conducted. Previously, the WLT numbers of multiple of 8 were used as training data set. To use test data different from training data, we compared the results of the ML model and TCAD simulation using the WLT number of multiple of 5. Fig. 8 shows ML’s $V_{th}$ prediction as a function of the WLT number compared with the test data set of TCAD simulation. The test results prove that ML also has accurate predictions for different WLT position.

5.4 Time required for data extraction

Figure 9 shows a comparison of the computation time required to extract the test data set between a TCAD simulation and an ML model. The computation time of TCAD is the time taken to extract 3000 test data (WL2, WL48, WL95) through TCAD simulation. The computation time of ML is sum of the following time: the time taken to extract the
training data set by TCAD simulation, the training time of ML, and the time taken to predict a test data set through ML. It shows that modeling through ML is faster than TCAD simulation. In addition, after the ML model is completed, a large amount of data can be quickly derived without any further training process.

6. Conclusion

In this letter, the ML model was proposed to predict $V_{th}$ distribution by taper angle and WLT position in 3D NAND Flash memory. The ML model was completed by training NNA. In the algorithm, we used back-propagation to reduce the error rate. The results of the ML model have a good agreement with training data. In addition, as a result of verifying the model using the test data for each of the taper angle and WLT position, it was confirmed that a reasonable model was derived. The ML model was easily created by the training process, and the time required for modeling and data derivation was shorter than that of TCAD simulation. Once the model is completed, the desired output value can be predicted without additional training, and a large amount of data can be obtained quickly. In this view, ML can be a good methodology for studying variation.
[21] H. Abd El Hamid, et al.: “Analytical model of the threshold voltage and subthreshold swing of undoped cylindrical gate-all-around-based MOSFETs,” IEEE Trans. Electron Devices 54 (2007) 572 (DOI: 10.1109/TED.2006.890959).

[22] K. Miyaji, et al.: “Control gate length, spacing, channel hole diameter, and stacked layer number design for bit-cost scalable-type three-dimensional stackable NAND flash memory,” Japanese Journal of Applied Physics 53 (2014) 024201 (DOI: 10.7567/JJAP.53.024201).

[23] J. Schmidt-Hieber: “Nonparametric regression using deep neural networks with ReLU activation function,” Annals of Statistics 48 (2020) 1875 (DOI: 10.1214/19-AOS1875).

[24] B. Hanin: “Universal function approximation by deep neural nets with bounded width and relu activations,” Mathematics 7 (2019) 992 (DOI: 10.3390/math7100992).

[25] C. Zhang and P.C. Woodland: “DNN speaker adaptation using parameterised sigmoid and ReLU hidden activation functions,” 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2016) (DOI: 10.1109/ICASSP.2016.7472689).

[26] S.N. Endah, et al.: “Beyond back-propagation learning for diabetic detection: convergence comparison of gradient descent, momentum and adaptive learning rate,” 2017 IEEE 1st International Conference on Informatics and Computational Sciences (ICICoS) (2017) (DOI: 10.1109/ICICOS.2017.8276360).

[27] V. Janakiraman, et al.: “Voltage and temperature aware statistical leakage analysis framework using artificial neural networks,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst. 29 (2010) 1056 (DOI: 10.1109/TCAD.2010.2049059).

[28] K. Wang, et al.: “The method of predicting retention threshold voltage distribution for NAND flash memory based on back-propagation neural network,” 2019 IEEE 11th International Memory Workshop (IMW) (2019) (DOI: 10.1109/IMW.2019.8739277).

[29] S. Sahay, et al.: “Hardware security primitive exploiting intrinsic variability in analog behavior of 3-D NAND flash memory array,” IEEE Trans. Electron Devices 66 (2019) 2158 (DOI: 10.1109/TED.2019.2903786).

[30] V. Subramanian: Deep Learning with Pytorch (Packt, New York, 2019) 1st ed.