Modeling and Testing of Interference Faults in the Nano NAND Flash Memory

Jin Zha*, Xiaole Cui*, and Chung Len Lee
School of Computer & Information Engineering,
Peking University Shenzhen Graduate School, Guangdong Province, China
E-mail: paulazha05@gmail.com; szbigendian@gmail.com

***Abstract*—Advance of the fabrication technology has enhanced the size and density for the NAND Flash memory but also brought new types of defects which need to be tested for the quality consideration. This work analyzes three types of physical defects for the deep nano-meter NAND Flash memory based on the circuit level simulation and proposes new categories of interference faults (IFs). Testing algorithm is also proposed to test the faults under the worst case condition. The algorithm, in addition to test IFs, can also detect the conventional address faults, disturbance faults and other RAM-like faults for the NAND Flash.**

Keywords—NAND Flash; Fault Model; Interference Fault

I. INTRODUCTION

The Flash memory, due to its high density and silicon technology compatibility, is a good candidate to replace hard disk as the secondary mass storage device for the portable systems. Similar to all other semiconductor memories, many fault models and various testing mechanisms were proposed to test to guarantee its quality [1]~[5]. For examples: several fault models were proposed in [1], where three types of disturbance faults were described and a modified March algorithm was proposed to test them; and the fault models were further expanded and March-like algorithms were proposed to test those faults [2]~[3]. Also, efforts were done to diagnose the memory when faults occur [4]~[5].

As the Flash memory is continuously scaled down, the reliability becomes a more serious problem due to the extremely critical process control. For the scale of 90nm, the main concerned problem is disturbance between cells, caused by defects in the tunnel oxide and the ONO layer [1]. However, for the memory fabricated in the sub-40nm range, other effects, caused by process defects [7]~[9], will cause additional interferences among adjacent cells. These effects will affect the operations of the NAND Flash memory in a way very similar to the coupling faults (CFs) of DRAM. They cause a cell of the NAND Flash to output a wrong value when its neighboring cells have certain logic states or make certain operations. This work explains these effects and tries to model the faults caused by these effects through simulation. Three types of faults, which are caused by three different kinds of defects, are analyzed. They are: (1) Cell-to-Cell Coupling [7]~[8], (2) Sidewall Isolation Leakage [9], and (3) Direct Electric Field Effect [10]. In the end, simple test algorithms will be proposed and demonstrate to test these faults.

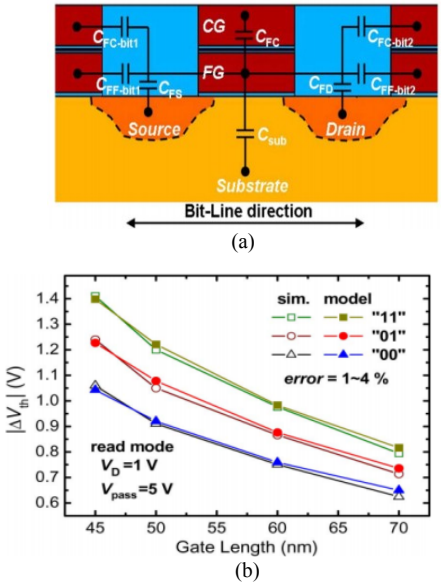


Figure 1. (a) The cross-section of an EEPROM cell with two adjacent cells; (b) The Vth shift of the programmed cell (Vth =2 V) versus the gate length of the memory.[7]

II. PHYSICAL MECHANISM OF INTERFERENCE FAULTS

A. Cell-to-Cell Coupling(CCC)

For the NAND type of Flash memory, the adjacent bits of the same bit line (BL) share the same source/drain. The distance between the floating gates (FGs) of each adjacent cell is determined by the size of the source/drain region (Fig. 1(a)). As the size of the cell transistor keeps shrinking, the capacitance coupling between the FGs will becomes significant. According to [7], the ΔV_{th} (transistor V_{th} shift) caused by the FG of the adjacent cells could be as high as 1.4V for a V_{th} = 2V programmed cell as the gate length reaches 45 nm (Fig. 1(b)). This results in a wrong output when the cell is read.

B. Sidewall Isolation Leakage (SIL)

The leakage caused by the electric field between two adjacent cells on a BL is another important issue for the deca-nanometer NAND flash memory. Fig. 2 shows the simulated potential distributions of three neighboring cells when one of

978-3-9810801-8-6/DATE12/ ©2012 EDAA

the cell, WL(n), is CG (control gate) and the other two neighboring cells, WL(n+1) and WL(n-1) on the same BLs are applied a Vpass. High electric fields (up to 9.0e+6V/cm [8]) exist between the two edge regions of the CG of WL(n) with respect to the FGs of the two neighboring cells WL(n+1) and WL(n-1). Such high electric fields could induce leakage current to leak the charges from the cells WL(n+1) and WL(n-1). Hence the neighboring cells undergo extra "erase" operation, causing erase faults.

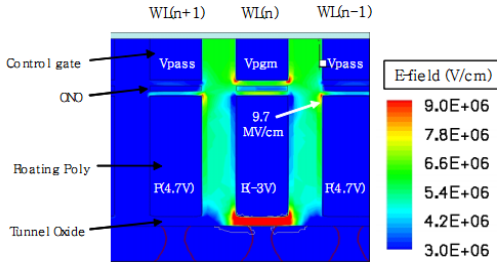


Figure 2. The simulated electric potentials of three neighboring cells with the cell WL(n) applied a programmed voltage and two neighboring cells applied a pass voltage.[8]

C. *Direct Electric Field Interference(DEFI)*

For the deca-nanometer NAND Flash memory, due to the extremely closeness of devices, the charges in the FG of a cell do not only affect the potential of the FGs in its neighboring cells, they also affect the field distribution of the conduction channel of its neighboring cells[9]. Fig. 3 shows the simulated electric potential of two cells on two different neighboring BLs. The voltage on the FG of the victim cell is -1V and that of the aggressor cell is changed from -2V to 2V. Due to the closeness of two BLs, the electric potential on the channel edge of the victim cell is raised, which becomes much higher than the potential of the channel center. In Fig. 3, about 0.23V (16%) difference exists between them. This phenomenon becomes more serious when there is a field recess, which induces an extra control path from the CG in the shallow isolating trench (STI) area to the channel (see Fig. 3). This is caused by the imprecise process control during manufacturing on the STI between two BLs [9]. The ΔV_{th} caused could be as high as 0.9V. That means when a cell is read, its output is "1" instead of "0" due to the lower value of V_{th} which causes conduction at the edge of the channel width of the cell.

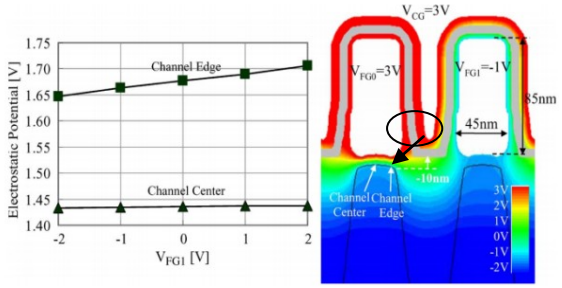


Figure 3. Simulation results depicting the potential distribution of the channel of a selected cell at the center and the edge due to the voltage of the FG of its neighboring bit-line cell.[9]

III. FAULTY EFFECTS ANALYZED THROUGH SIMULATION

To study the effects caused by the above defects through circuit level SPICE simulations, Fig. 4 is shown, in which an EEprom cell is modeled by a combination of one capacitor and one MOS transistor, proposed in [6].

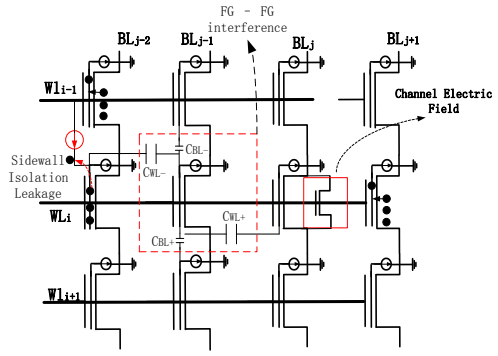


Figure 4. SPICE model and functional behavior of the interference defects.

A. *SPICE Models*

First, for the CCC FG-FG interference, the defect is denoted by capacitors in the circuit since it is caused by the capacitive coupling between the FGs of cells. In Fig. 4, $C_{i,j-1}$is the victim cell and its four neighboring cells on the BL and WL are aggressors. C_{BL+}, C_{BL-}, C_{WL+} and C_{WL-} represent the defects causing the CCC interference. Next, the SIL defect is modeled as a Fowler-Nordheim [11] tunneling current source between cell $C_{i,j-2}$ and cell $C_{i-1,j-2}$ since the sidewall isolation leakage is induced by a large parasitic electric field in the sidewall isolation region between two cells. When a high programming voltage is applied to the control gate $CG_{i-1,j-2}$ of the aggressor cell $C_{i-1,j-2}$ and a low voltage exists on $FG_{i,j-2}$ of the victim cell $C_{i,j-2}$, an erroneous erasure can happen to cell $C_{i,j-2}$. As for the DEFI, it is modeled by a parasitic transistor between cell $C_{i,j}$ and cell $C_{i,j+1}$. As described in Section II, the extra electric fields from the victim cell's own CG in the STI area are able to raise the voltage of the channel, resulting in an unexpected "ON" of the victim cell itself. This effect on the channel on the cell transistor is just the same as that of a conventional MOS transistor, hence a parasitic MOS transistor is used to model this defect.

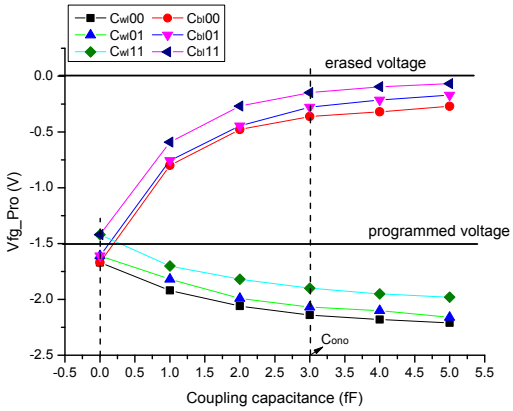


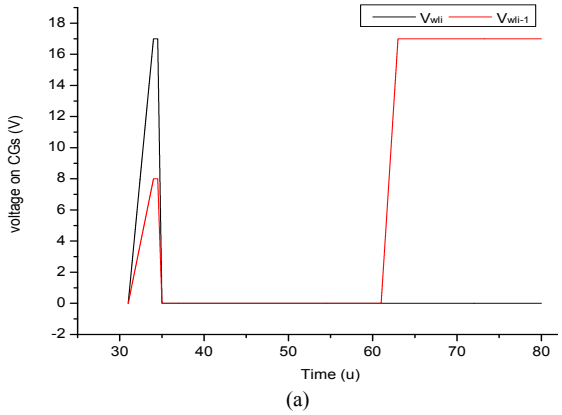
Figure 5. Vth_shift induced by the coupling capacitors.

B. *Simulation Results*

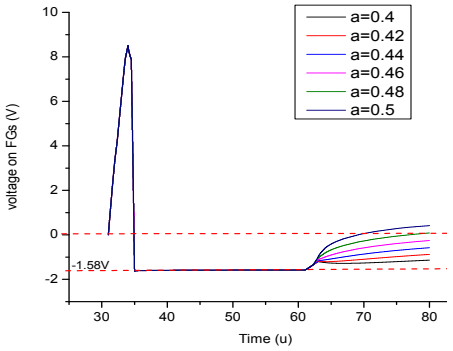
The above circuit models were used to simulate the faulty behavior of the Flash for a 3x3 NAND array. The results are shown and explained below.

Fig.5 is the simulated voltages on the FG of a programmed victim cell due to the CCC effect when its adjacent aggressor cells are at certain logic states (00, 01, or 11). The curves are plotted w.r.t. the coupling capacitance values for various cases that different patterns were applied to the adjacent aggressor cells. In the figure, $C_{bl}00$ means that the two adjacent aggressor cells on the same BL of the victim cell were at states (0,0) respectively and $C_{wl}11$, the two aggressor cells on the WL at states (1,1), etc. -1.5V and 0V on FGs were used for the logic levels 0 and 1 in the simulation. It is seen that, when there are no defects, i.e., coupling capacitance is 0, the V_{fg} difference of the victim cell related with the background is less than 0.2V. When the capacitance exists and increases, V_{fg} shifts. The C_{wl} induces negative voltage shift, which means the interference from the WL enhances programming of the victim cell. On the contrary, the BL interference, inducing positive shift, inhibits programming. Both the interferences on the two directions depend on the background states of adjacent cells. Overall, the V_{fg} is higher for the aggressor cells at (1,1) state than that at (0,0) state, indicating that (1,1) is good for activating a BL interference fault while (0,0) is useful for activating a WL interference fault for testing. Besides, when the capacitance exceeds 3fF (C_{ono}), which is the value of the original ONO layer (floating gate insulator of the cell transistor), C_{bl} causes the shift to reach 1V. This is an "un-programmed" fault. Similarly, with a coupling on WLs, the V_{fg} is programmed to a lower voltage, resulting in an "overwrite" fault.

Fig.6 is the simulation result of SIL effect, where (a) shows the waveforms of the voltage respectively applied on the selected CG on WL_{i-1} and the unselected CG on WL_i; and (b) is that of the voltage of the FG of the victim cell. For this simulation, the victim cell $C_{i,j-2}$ was firstly programmed with a V_{th}= 1.58V, which is proportional to the absolute value of V_{fg}. Then a high program voltage was applied on its CG of its neighboring aggressor $C_{i-1,j-2}$ at t = 63u as shown in Fig. 6(a). A large electric field between $CG_{i-1,j-2}$ and $FG_{i,j-2}$ is induced in the isolation region. Under the influence of this electric field, charges on $FG_{i,j-2}$ are leaked through FN tunneling, and the voltage on this FG rises gradually from -1.58V to near 0. This results in an erase fault. Also, the larger the defect size, the quicker the rise. The parameter 'a' in Fig. 6(b) represents the influence of the defect size, which is multiplied with the electric field.



(a)



(b)

Figure 6. (a) The voltage applied on the CGs; (b) the simulated voltage on the FG of the victim cell.

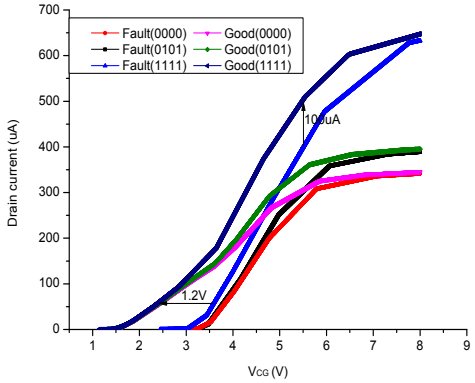


Figure 7. The simulated current flowing through the victim cell w.r.t. the its CG for the faulty case and the good case respectively for different aggressor background patterns.

For the DEFI effect, the simulation result is shown in Fig. 7, where the current flowing through the victim cell is plotted w.r.t. the voltage of its CG for both the faulty case and the good case with various aggressor background patterns. In this simulation, the victim cell was originally at 1 state, and then programmed to 0 state. As shown in Fig. 7, the V_{th} of the faulty cell is much lower than that of the good cell after the program operation. The drain current in the faulty cell increases much slowly than does the good cell. However, as time goes on and the voltage on the CG rises, two currents become nearly the same. As a result, when the reading voltage on the CG is low, a 1 will is read in the programmed cell due to this parasitic transistor problem. Hence, the cell is harder to be programmed to the 0 state. Besides, the defect could be inhibited through increasing the reading voltage.

IV. FAULT MODELS

As the results of the above study show, three functional fault models are given in this section. A 3x3 array as shown in Fig. 9(a) is taken as an example for explanation. Only the interference caused by the four adjacent cells on the WL and BL are considered.

As discussed previously, the interference faults (IFs) depend much on the background patterns. This is very similar to the neighborhood pattern sensitive faults (NPSFs) in DRAM [12]. The notation to express the pattern for testing those IFs is

described in the form: {(0, 1, a, 2, 3); O/F}, where 'a' is the initialization of the victim cell, (0, 1, 2, 3,) are patterns of the four adjacent cells (aggressors), which could affect the content of the victim, "O" stands for the operation performed on the victim cell, and "F" is the faulty value.

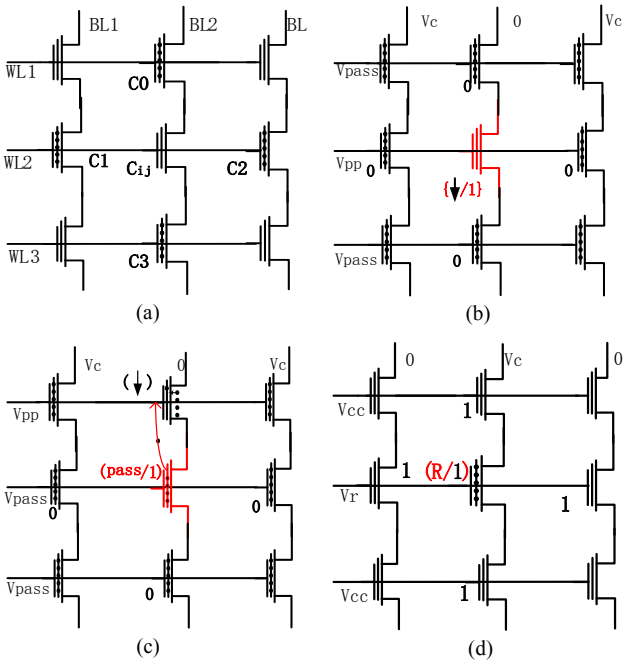


Figure 8. (a) a typical 3x3 NAND Flash array; (b) the test condition for the PRIF fault; (c) the test condition for the SFIF fault; and (d) the test condition for CFIF fault.

The three fault models are described as follows:

PRIF (Pattern Related Interference Fault), is caused by the CCC defect. It has been simulated in Section III, that all '1" is the worst background pattern for an un-programmed fault, caused by the BL interference defect; and all '0' is the worst case for an overwrite fault, induced by the WL interference defect. It can be noted as {1,1,1,1,1; ↓/1} and {0,0,1,0,0;(↓,↑)/0} respectively, where ↑ and ↓ represent program operation and erase operation respectively. Fig. 8(b) demonstrates the testing bias and the behavior of this fault.

SFIF (Side Field Interference Fault), is caused by the SIL effect. The leakage current induced by the high electric field produces an erasure fault on the victim cell. To test this fault, the center cell Cij is programmed firstly. Then three of the neighboring cells are set to "0" and the last is selected to set a high voltage on its CG, which is used to induce a high electric field in the isolate region. If there exist SFIFs, the cell should be read as "1" due to leaking of charges from its FG. The notation is { ↓, 0, 0, 0, 0; R/1} or {0, 0, 0, 0,↓; R/1}. Fig. 8(b) demonstrates the test condition and behavior for this fault.

CFIF (Channel Field Interference Fault), is caused by the DEFI effect. Due to the electric field caused by the neighboring FG and the deep CG, the victim cell is partially on. As a result, it is read as "1" even though a "0" is stored on it. To test it, the pattern is {1,1,0,1,1; R/1}, as shown in Fig. 8 (c). Table I compiles all the test conditions for the interference faults and their corresponding DRAM-like faults. As most IFs are related with the background patterns of the aggressor cells, it is hard to take all combinations of background patterns to test an IF since it is too expensive to applying all combinations of patterns. Hence for the practical purpose, to test an IF, only the worst case background combination is introduced.

TABLE I. COMPILATION OF IFS AND THEIR TEST CONDITIONS

Fault	Notation	Excitation Location	RAM- Like Fault	Right value
PRIF1	{1, 1, 1, 1, 1; ↓ /1}	BL	PNPSF	0
PRIF0	{0, 0, 1, 0, 0; (↓,↑)/0}	WL	PNPSF	1
SFIF	{↓, 0, 0, 0, 0; -/1}, {0, 0, 0, 0,↓; -/1}	BL	ANPSF	0
CFIF	{1,1,0,1,1; R/1}	WL	CFst	0

V. TESTING ALGORITHM

As shown in Table I, the worst case background combination patterns of aggressor cells on the same BL or WL are always uniform patterns, i.e. (0,0) or (1,1), and two faults with the same background pattern have different activation conditions in order to be tested, the activation patterns never overlap with the background pattern at the same time. A Diagonal Checkboard Algorithm (DCA), which is able to gain the almost worst case and to detect two faults at the same time, is proposed, which is shown in Fig. 9. In the figure, the numbers behind cells means the sequence of programming. The cells on the diagonal lines are programmed by every other diagonal line. It can be seen that, before programming, the cells on a certain diagonal line have the background pattern {1111}. After each diagonal programmed, the cells on the diagonal line will have the background pattern {0000}. The detail algorithm is given in Table II for an array with m columns and n rows. Some steps of the algorithm are explained as follows:

Step2: Initialize all cells the value '1' and read "1", all cells have background pattern {1111}.

Step3: Program the odd diagonals with pattern (R, P, R), where the first R detects the WPD (word line program disturbance) and BPD (bit line program disturbance) faults. The next "P" activates PRIF1 and CFIF of the selected cell, while also activates those PDs on the neighboring diagonals. Accordingly, the following R can detect the PRIF1and CFIF. Besides, the SA1 and TF1 faults on the cell could also be tested through this read operation. After that, two R operations are added to the two neighborhoods of the programmed cell on the previous diagonal. They can check out PDs on these cells. After the whole diagonal is programmed, the cells on the previous diagonal will have the background pattern {0000}.

Step4: Program the even diagonals with the shorter pattern (P, R), since two single R operations in Step3 are enough to test the target faults. At this time, the background pattern is {0000}. The program operations in this step are able to activate PRIF0 on the selected cell and the SFIFs, WED (word line erase disturbance) and BED (bit line erase disturbance) on the odd diagonals. The R operation in the pattern is used to test the SA0 and TF0 on the programmed cells. Another two R operations on the neighboring cells in this step are used to detect the activated faults previously.

Step5: Reset the values stored in the cell. After this flash operation, the following read operations are able to test PRIF0, and in the following P and R operations can also detect the OE (over erase) fault.

Step6: Change the program operation sequence from odd to even and detect the complementary faults.

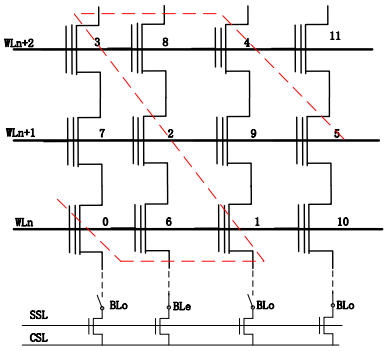


Figure 9. Testing sequence for checkboard pattern.

TABLE II. DIAGONAL CHECKBOARD ALGORITHM

Step 1. Regroup all cells in diagonals. // n+m-1 columns
Step 2. Flash all and Read all; // cells are set to '1'
Step 3. For i=1; i+2; i<n+m-1 //program the odd diagonals
For j=0; j+1; j<k_i
//k is the number of the cells on the i_{th} diagonal
Add pattern (R, P, R) to C(i,j);
Read C(i-1, j-1)
Read C(i-1, j)
//read the adjacent cells on the previous even diagonal
Step 4. For i=0; i+2; i<n+m-1
For j=0; j+1; j<k_i
//program the even diagonals in the same way
Add patter (P, R) to C(i,j);
Read C(i-1, j-1)
Read C(i-1, j)
// read the adjacent cells on the previous odd diagonal
Step 5. Flash all and read all;
Step 6. Repeat 3~4, from even diagonals to odd diagonals.
Step 7. Analyze the results

The complexity of the above algorithm is $2F + 2mn \cdot P + 9mn \cdot R$, in which F, P and R are the time of the flash operation, program operation and R operation, which is acceptable in the manufacturing stage. Besides, compared to those algorithm, like TLSNPSF1G [12], which needs 43.5n operations (n means number of the cell in the array), it could be very efficient for testing NPSFs. Moreover, since the algorithm is based on the pattern (R, P, R), which has been proved to be able to test PDs, EDs and some of RAM-like faults in [2], it can also be used to detect those kinds of faults.

VI. CONCLUSION

This paper analyzes three types of physical defects which may occur in the deep nano-meter NAND Flash memory due to the vulnerable extreme fabrication process control with circuit level device defect models. Based on the simulation results, it proposes three types of interference faults and elucidates the test conditions for them. Finally, it also proposes the testing algorithm to test these faults under the worst case condition. The algorithm is moderated in its computation complexity, which is acceptable in the manufacturing stage. It is proved to be able to detect the conventional faults such as address faults, disturbance faults and other RAM-like faults.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (Grant No.61006032), the Natural Science Foundation of Guangdong Province, China (Grant No.S2011010001234) and R&D project of Shenzhen Government, China (Grant No. CXB201005260062A)

REFERENCES

[1] M. G. Mohammad and K. K. Saluja, "Flash memory disturbances: Modeling and test," in Proc. IEEE VTS, Marina Del Rey, CA, Apr. 2001, pp. 218–224

[2] Yeh, J.-C., Kuo-Liang Cheng, Yung-Fa Chou; Cheng-Wen Wu; , "Flash Memory Testing and Built-In Self-Diagnosis With March-Like Test Algorithms," Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on , vol.26, no.6, pp.1101-1113, June 2007

[3] Kuo-Liang Cheng, Jen-Chieh Yeh, Chih-Wea Wang, Chih-Tsun Huang, Cheng-Wen Wu, "RAMSES-FT: a fault simulator for Flash memory testing and diagnostics," VLSI Test Symposium, 2002. (VTS 2002). Proceedings 20th IEEE , vol., no., pp. 281- 286, 2002

[4] Ginez, O., Portal, J.-M., Aziza, H., "A High-Speed Structural Method for Testing Address Decoder Faults in Flash Memories," ITC 2008. IEEE International , vol., no., pp.1-10, 28-30 Oct. 2008

[5] Mauroux, P.-D., Virazel, A., Bosio, A., Dilillo, L., Girard, P., Pravossoudovitch, S.; Godard, B. , "NAND Flash testing: A preliminary study on actual defects," ITC 2009. International , vol., no., pp.1, 1-6 Nov. 2009

[6] L. Larcher, I.R.A. Padovani, P. Pavan, G.M.A. Calderoni, F. Gattel, and P. Fantini. "Modeling NAND Flash Memories for Circuit Simulations". In Simulation of Semiconductor Processes and Devices (SISPAD), pages 293--296, Italy, 2007. Springer Vienna.

[7] Postel-Pellerin, J., Canet, P., Lalande, F., Bouchakour, R., Jeuland, F., Bertello, B., Villard, B., , "A Full TCAD simulation and 3D parasitic capacitances extraction in 90nm NAND Flash memories," NVMTS 2008. 9th Annual , vol., no., pp.1-4, 11-14 Nov. 2008

[8] Sang-Goo Jung, Keun-Woo Lee, Ki-Seog Kim, Seung-Woo Shin, Seaung-Suk Lee, Jae-Chul Om, Gi-Hyun Bae, Jong-Ho Lee, "Modeling of Vth Shift in NAND Flash-Memory Cell Device Considering Crosstalk and Short-Channel Effects," Electron Devices, IEEE Transactions on , vol.55, no.4, pp.1020-1026, April 2008

[9] Yong Seok Kim, Dong Jun Lee, Chi Kyoung Lee, Hyun Ki Choi, Seong Soo Kim, Jai Hyuk Song, Du Heon Song, Jeong-Hyuk Choi, Kang-Deog Suh, Chilhee Chung, "New scaling limitation of the floating gate cell in NAND Flash Memory," Reliability Physics Symposium (IRPS), 2010 IEEE International , vol., no., pp.599-603, 2-6 May 2010

[10] Mincheol Park, Keonsoo Kim, Jong-Ho Park, Jeong-Hyuck Choi, "Direct Field Effect of Neighboring Cell Transistor on Cell-to-Cell Interference of NAND Flash Cell Arrays," Electron Device Letters, IEEE , vol.30, no.2, pp.174-177, Feb. 2009

[11] Zous N.-K., Wang T., Yeh C.-C., Tsai C.-W., Huang C., "A comparative study of SILC transient characteristics and mechanisms in FN stressed and hot hole stressed tunnel oxides," Reliability Physics Symposium Proceedings, pp.405-409, 1999

[12] A.J.van.de Goor, Testing Semiconductor Memories: Theory and Practice. Chichester, UK: John Wiley & Sons, Inc., 1991