Bit-error rate improvement of TLC NAND Flash using state re-ordering

Ik Joon Chang¹ and Joon-Sung Yang²a)
¹ Department of Electronic and Radio Engineering, Kyunghee University, Yongin-si, Gyeonggi-Do, 446–701, Korea
² Department of Semiconductor Systems Engineering, SungKyunKwan University, Suwon-si, Gyeonggi-Do, 440–746, Korea
a) js.yang@skku.edu

Abstract: In scaled technologies, large cell-to-cell interference and F-N tunneling disturbance degrade threshold voltage (V_t) window which we can place program states. Moreover, in Triple Layer Cell (TLC) NAND Flash we should place seven program states (P₁ ~ P₇) in the narrow V_t window, incurring large bit-error rate (BER). In this paper, we propose a state re-ordering technique to increase the efficiency of V_t window utilization in TLC NAND Flash memories. Our simulation results show that under equivalent V_t window sizes, the proposed technique provides 12.5~18.4% smaller BER compared to conventional Gray-code mapping.

Keywords: TLC NAND Flash Design, code-mapping
Classification: Storage technology

References

1 Introduction

During programming operations of NAND Flash memories, cell-to-cell interference and F-N tunneling disturbance vary threshold voltages of the
E-state cells [1, 2]. This narrows the window where threshold voltages ($V_t$’s) of program-state cells can be placed. The cell-to-cell interference and the F-N tunneling disturbance become larger as the NAND Flash technology is scaled down, making worse such a phenomenon. Moreover, the cell-to-cell interference also degrades $V_t$ distributions of program-state cells. Consequently, in sub-30 nm Triple-Level Cell (TLC) NAND Flash it is challenging to place seven program states in the narrow $V_t$ window.

2 State re-ordering

Due to the narrow $V_t$ window, we cannot avoid overlapped regions between adjacent states in TLC NAND Flash memories, as shown in Fig. 1. At these regions, we have bit-errors during read operations and they are recovered using an error correcting code (ECC) technique in the off-chip controller [2]. In scaled NAND Flash memories, BCH code is widely used for the ECC technique [3]. Typically, 1 KB word is the basic processing unit of the BCH code and extra parity bits are added to this unit for error correction. A group of 1 KB word and corresponding parity bits is called as one code-word. The maximum number of bit-errors which the BCH code can correct in single code-word, namely maximum correction number (MCN), is decided by the number of parity bits (i.e. MCN: 40-bit for '1 KB word + 560-bit parity') [4].

![Fig. 1. The mapping between states and page information under Gray-code](image)

To read the information of three pages stored in TLC NAND Flash, we need seven read levels (RL’s) as shown in Fig. 1. Here, we should note that in TLC NAND Flash, Gray-code has been used for the mapping between each state and page information [5]. For example, ‘110’ (the 1st page: 0, the 2nd and the 3rd pages: 1) is mapped to $P_7$-state while $P_6$-state represents ‘010’ (the 1st and the 3rd pages: 0, the 2nd page: 1). Under such an environment, one, two and four RL’s should be assigned for the 1st, the 2nd, and the 3rd page readings respectively, as shown in Fig. 2. As addressed above, some bit-errors occur during read operation of each read level (RL), which are corrected by using BCH code. For successful error corrections, the inequality conditions of Fig. 2 should be satisfied.

To deliver these requirement, we need to control the number of bit-errors occurring at the overlapped region of adjacent states by varying the distances between $VL_i$’s ($VL_i$: the verify level of $P_i$ state). Fig. 3 shows the relation between $\Delta VL_2$ ($\Delta VL_i$: when $i > 1$, $VL_i - VL_{i-1}$). When $i=1$, $VL_1$ –
minimum \( V_t \) of E-state cells) and the corresponding number of error-bits in \( P_1 \) and \( P_2 \) states. For simplicity, we assume that \( V_t \)'s of \( P_1 \) and \( P_2 \) states have a Gaussian distribution with 150 mV standard deviation. As shown in Fig. 2, four RL’s (RL1, RL3, RL5, and RL7) are necessary for the 3\textsuperscript{rd} page reading due to Gray-code mapping. Under the assumption that the MCN is 40-bits, the average number of bit-errors per each RL in the 3\textsuperscript{rd} page reading is 10-bits. In Fig. 3, we can observe that as \( \Delta V_{L2} \) increases, the corresponding reduction of bit-errors becomes exponentially reduced. This implies that large \( \Delta V_{L1} \), \( \Delta V_{L3} \), \( \Delta V_{L5} \) and \( \Delta V_{L7} \) are required to regulate the number of the corresponding bit-errors below 10-bits, degrading the efficiency of \( V_t \) window utilization. This makes it difficult to satisfy the 40-bit MCN requirement under the narrow \( V_t \) window in sub-30 nm technologies.

We alleviate this problem by proposing a state re-ordering technique. Fig. 4 shows the proposed technique. Here, '110' is mapped to E-state while \( P_7 \)-state represents '110' under Gray-code mapping. Then, the information of E and \( P_1 \) ~ \( P_6 \) states under the Gray-code mapping is mapped to \( P_1 \) ~ \( P_7 \) states. In the proposed scheme, two, two and three RL’s are assigned for the 1\textsuperscript{st}, the 2\textsuperscript{nd}, and the 3\textsuperscript{rd} page, as shown in Fig. 5. Since the number of RL’s for each page varies, the inequality conditions for error correction also change. Compared to Gray-code mapping, the number of RL’s for the 1\textsuperscript{st} page reading increases in the proposed method. This degrades the efficiency of \( V_t \) window utilization to a certain degree. However, in the 3rd page reading,
the number of RL’s decreases, improving the utilization efficiency more significantly.

In order to validate the proposed technique, we simulate and compare an occupied $V_t$ window size between Gray-code mapping and the proposed technique. For simplicity, we assume the following conditions in these simulations: all erase or program states ($E, P_1 \sim P_7$) have a Gaussian distribution and all program states have the same standard deviation value as $\sigma_{V_{\text{P-states}}}$ while they have different average values. For two mapping scenarios, the average and standard deviation of E-state cells are assumed to be equivalent: 0 V and 350 mV, respectively. We also assume that the MCN is 40-bits and each state has equivalent cell numbers for every code-word. We consider that the minimum grid of verify level variation is 50 mV due to a circuit limitation. Based on the above assumptions, we can achieve proper $\Delta V_L$’s to satisfy the inequality conditions of Fig. 2 and Fig. 5 for 1536 code-words, which is one-block size (=64WL per one block $\times$ 24 code-words per one WL) in TLC NAND Flash. Using these $\Delta V_L$’s, we are able to generate $V_t$ distributions for two mapping scenarios, as shown in Fig. 6. When $\sigma_{V_{\text{L-state}}}$ = 150 mV, our proposed technique shows 150 mV smaller $V_t$ window size compared to Gray-code mapping.

We made the same comparisons for $\sigma_{V_{\text{L-state}}}$ = 140 mV and $\sigma_{V_{\text{L-state}}}$ = 130 mV, whose results are shown in Fig. 7 (a). Here, the proposed state ordering shows at least 150 mV smaller window size than the Gray-code mapping. This implies that under the same $V_L$ window, the proposed technique will show better bit-error rate (BER) compared to the Gray-code mapping. In order to prove this, we compared maximum number of bit-errors for 1536 code-words under the equivalent $V_t$ window condition. For each $\sigma_{V_{\text{L-state}}}$ we used the $V_t$ window size of Gray-code mapping obtained from Fig. 7 (a). The results are shown in Fig. 7 (b), where the
The proposed technique shows 12.5~18.4% smaller number of bit-errors. This proves the aforementioned conjecture well.

3 Conclusions

We propose the state-reordering technique to ameliorate the efficiency of $V_t$ window utilization. Our simulations prove that the proposed technique occupies smaller $V_t$ window in TLC NAND Flash, compared to Gray-code mapping. Hence, this implies that we can improve BER of TLC NAND Flash memories by using the proposed technique.