LEAST SQUARES BASED CELL-TO-CELL INTERFERENCE CANCELATION TECHNIQUE FOR MULTI-LEVEL CELL NAND FLASH MEMORY

Dong-hwan Lee and Wonyong Sung

Department of Electrical Engineering and Computer Science, Seoul National University
Gwanak-gu, Seoul, 151-744 Republic of Korea
Email: ldh@dsp.snu.ac.kr, wysung@snu.ac.kr

ABSTRACT

Cell-to-cell interference becomes a major source of bit errors in NAND flash memories as the semiconductor technology continuously shrinks down. Recently, signal processing approaches to mitigate the interference have been proposed, and the least mean square (LMS) adaptive filtering based method [1] offers a promising solution. In this research, we propose a least squares based cell-to-cell interference cancelation method, which is more suitable for NAND flash memory devices where one page of data is accessed at a time. With a simulation model, we show that this approach outperforms the LMS filtering based one whether the interference is severe or not. In order to simplify the algorithm, the input data to compute the channel characteristics is decimated, and as a result the arithmetic intensity of the proposed algorithm is comparable to the LMS based one.

Index Terms— Cell-to-cell interference, NAND flash memory, least squares, LMS adaptive filter

1. INTRODUCTION

NAND flash memory is widely used because of its high capacity, small access latency, and low power consumption [2]. During the last 10 years, the capacity of NAND flash memory has increased nearly 1,000 times by aggressive process scaling down and multi-level cell (MLC) technology. In MLC technology, two or more bits are stored in a cell to lower the cost per bit. However, because of small voltage gap between adjacent symbols, MLC NAND flash memory shows poor error performance when compared with SLC (single-level cell) one. The cell-to-cell interference (CCI), data retention, and excessive amount of program-erase cycles are the major error sources. Among them, the CCI is the most significant one in many cases [3, 4].

Previous studies to reduce the cell-to-cell interference can be classified into two; one is modifying the memory structure [5] and the programming scheme [6], and the other is using signal processing techniques [1, 7]. As an example of the former, the multi-page programming scheme that programs MSB (Most Significant Bit) and LSB (Least Significant Bit) pages separately was proposed in [6]. For the signal processing approach, a least mean square (LMS) adaptive filtering based interference canceler was recently proposed [1]. Even though LMS filtering requires only small arithmetic operations, this algorithm is a sequential process that utilizes the data sample one by one. In NAND flash memory, however, not just a single data sample but the entire samples are available when the read operation for a page is completed. Thus, a sequential processing, such as the LMS algorithm, does not utilize all the data efficiently.

In this research, we apply the least squares based algorithm for coupling cancelation in MLC NAND flash memory. In this method, sufficient amount of data samples in a page are used for finding the coupling coefficient. Also, a high degree of parallelism can be employed, which can greatly reduce the processing time when multiple processing elements are used. In order to verify the performance of the proposed interference cancelation techniques, we modeled a four-level NAND flash memory channel that includes the effect of CCI. For each CCI cancelation method, BER (Bit Error Rate) is measured and compared with that of uncompensated cells.

This paper is organized as follows. Section 2 explains the NAND flash memory channel model. In Section 3, we propose the least squares based CCI canceler. In Section 4, experimental results are shown. Finally, a discussion is given in Section 5.

2. MLC NAND FLASH MEMORY CHANNEL MODEL

In this paper, we use the MLC NAND flash memory channel that is modeled in [1, 7].

2.1. Program and erase processes

A cell block in NAND flash memory is a two-dimensional cell array that consists of multiple word- and bit-lines as shown in Fig. 1. In the even/odd bit-line structure, cells on the even bit-lines form even pages, while cells on the odd bit-lines become odd pages. Thus, there are four pages in a word-line: even MSB/LSB pages, and odd MSB/LSB pages. The program and read operations are carried out page based, while a block is used for erase process. Before programming the memory cells, the charges in each cell's floating gate need to be removed, which is called the erase operation. It is well-known that this process results in the Gaussian distribution [7].

To reduce the CCI between adjacent word-lines, MSB page programming for a selected word-line is performed after LSB page programming of its neighbor word-lines as illustrated in Fig. 1 [6]. Between the even and the odd pages, the former is programmed first. When the cells are programmed, the incremental stair pulse programming (ISPP) is used to achieve a tight threshold voltage bound [8]. In this programming scheme, the threshold voltage of each target cell increases as much as ΔV_{pp} and is compared with the target voltage at each iteration. If the voltage is higher than the target voltage, the programming operation stops. It is well known that ideal ISPP results in a uniform distribution as shown in Fig. 4(b). In this paper, the target voltages are denoted as V_1, V_2, and V_3 according to

This work was supported in part by the Brain Korea 21 Project grant funded by the Ministry of Education, Science and Technology (MEST), Republic of Korea.

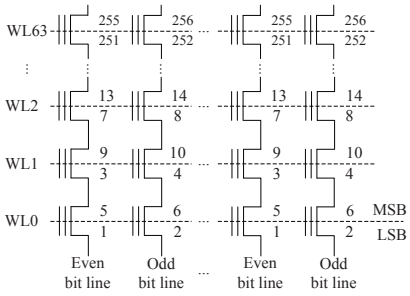

Fig. 1. NAND flash memory structure and programming order for a 4-level cell. The numbers represent the programming order.

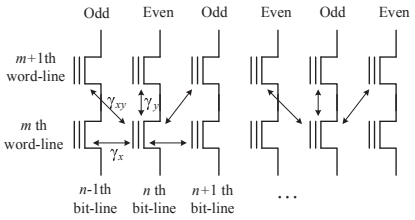

Fig. 2. Cell-to-cell interference model in the even/odd bit-line structure. The n-th cell is the victim cell, and n_{ul} to n_r represent the neighbor cells.

the cell's symbol. The distribution becomes distorted as the memory cells are exposed to the cell-to-cell interference.

2.2. Cell-to-cell interference

The threshold voltage shift of one cell changes those of neighbor cells due to the parasitic capacitor-coupling effect [3]. Thus, when one cell is programmed, not only the target cell's but also the surrounding cells' threshold voltages also increase. This is referred as the CCI [4].

The amount of interference that the victim cell receives, $V_I[m,n]$, can be represented as a linear combination of the neighbor cell's voltage shift.

$$\begin{aligned} V_I[m,n] = {} & \gamma_x \cdot (\Delta V[m,n-1] + \Delta V[m,n+1]) \\ & + \gamma_y \cdot \Delta V[m+1,n] \\ & + \gamma_{xy} \cdot (\Delta V[m+1,n-1] + \Delta V[m+1,n+1]), \end{aligned} \quad (1)$$

where $\Delta V[m,n-1]$ is the threshold voltage shift for the left neighbor of the victim cell, and so on. The coefficient, γ_i, is referred as the coupling ratio. In our cell model, we assume that the coupling ratios are Gaussian random variables, and $E[\gamma_x]$, $E[\gamma_y]$, and $E[\gamma_{xy}]$ are set to $0.05s$, $0.1s$, and $0.025s$, where the coupling coefficient factor, s, varies. As s increases, the CCI becomes severe. Note that $E[x]$ refers the expectation of random variable x. The coupling ratio parameters are set by referring [1].

According to the memory structure and the programming method, the number of interfering cells for a victim cell varies. Because of the verification process in the ISPP, the CCI due to the previously programmed cells does not exist. Only the surrounding cells that are programmed later than the victim cell cause the CCI. Hence, in the even/odd bit-line structure, three cells from the next word-line ($m+1$th word-line in Fig. 2) and two cells on the same word-line alter the threshold voltage of the even victim cell. Note that in Eq. (1) is for the even cells. An odd victim cell is affected only by three surrounding cells placed in the next word-line. Since the cells on the erase state are not programmed, they only receive the interference.

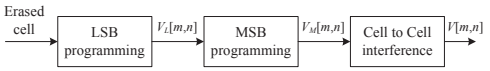

Fig. 3. 4-level NAND flash channel model.

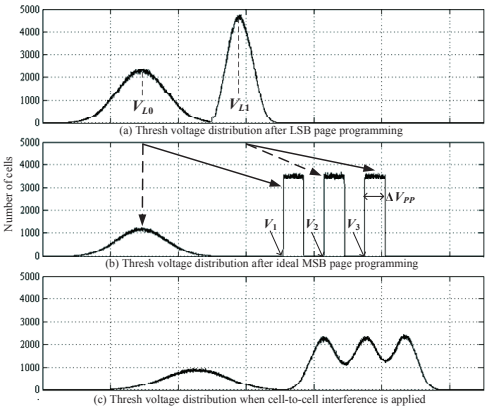

Fig. 4. Threshold distribution of a simulated 4-level NAND flash memory.

2.3. Channel modeling of four-level NAND flash memory

The hypothetical flash channel model is constructed in three stages: LSB programming, MSB programming, and CCI as illustrated in Fig. 3, and Fig. 4 shows the obtained threshold voltage distribution. After LSB page programming, the (m,n)-th cell has the threshold voltage of $V_L[m,n]$ that is sampled from a Gaussian distribution whose mean is V_{L0} when the LSB is '0', or V_{L1} otherwise. When the MSB pages are programmed, the threshold voltage becomes $V_M[m,n]$, and the voltage shift from $V_L[m,n]$ to $V_M[m,n]$ induces the CCI to the surrounding cells. As explained in the above, $V_M[m,n]$ is a random variable whose PDF (probability density function) is a uniform distribution with the mean of $V_1 + \frac{1}{2}\Delta V_{pp}$, $V_2 + \frac{1}{2}\Delta V_{pp}$, or $V_3 + \frac{1}{2}\Delta V_{pp}$ according to its symbol as shown in Fig. 4(b). Ideally, $\Delta V[m,n-1]$ in Eq. (1) is equal to $V_M[m,n-1] - V_L[m,n-1]$. Since these values cannot be measured in real situation, $E\{V_L[m,n-1]\}$ and $E\{V_M[m,n-1]\}$ are used instead. The (m,n)-th cell is affected by the CCI, and the resulting voltage is denoted as $V[n]$. In this work, we assume that $V[n]$ is measured in high precision during the read operation. The amount of CCI that the (m,n)-th cell receives is $V_I[m,n] = V[m,n] - V_M[m,n]$, and $V_M[n]$ needs to be replaced with its mean value as well.

2.4. Least mean square (LMS) filtering based coupling canceler

In order to derive the LMS solution for the CCI cancelation [1], let us define a vector as follows:

$$\mathbf{U}[n] = [\Delta V[m+1,n-1], \Delta V[m+1,n], \Delta V[m+1,n+1]]^T. \quad (2)$$

Note that n is 0, 1, $\cdots$, $N-1$, where N is the number of cells in a page. Eq. (2) is for the odd cells and can be simply expanded to the even cells. If we assume that the mean values of $V_M[m,n]$ and $V_L[m,n]$ are known in advance, $\mathbf{U}[n]$ is approximated to $E\{V_M[m,n]\} - E\{V_L[m,n]\}$. From this definition, we can rewrite the right hand side of Eq. (1) as follows:

$$y[n] = \mathbf{x}[n] \cdot \mathbf{U}[n], \tag{3}$$

where $\mathbf{x}[n]$ represents the coupling coefficient in a vector form. Again, we replace $V_I[m,n]$ in Eq. (1) as an estimator $V[m,n] - E\{V_M[m,n]\}$, and this value is defined as the desired signal of the LMS adaptive filter $d[n]$. Depending on the symbols of cells, $E\{V_M[m,n]\}$ and $E\{V_L[m,n]\}$ are varied, and the uncompensated hard-decisioned one (read symbol) can be used. $y[n]$ and $d[n]$ are two different estimators of the amount of CCI, and their difference is used to define a cost function.

$$J(\mathbf{x}) = \frac{1}{2}|e[n]|^2, \text{ where } e[n] = d[n] - y[n] \tag{4}$$

In the LMS filtering, $J(\mathbf{x})$ can be minimized by iteratively updating the weight vector.

$$\mathbf{x}[n+1] = \mathbf{x}[n] + \mu e[n] U[n] \tag{5}$$

Note that μ is a positive constant. The entire procedure of the LMS filtering is shown in Algorithm 1.

Algorithm 1 LMS filtering based cell-to-cell interference cancelation method

for ($n = 0$ to $N-1$)
LMS filtering:

$$y[n] = \mathbf{x}[n] \cdot \mathbf{U}[n]$$
$$e[n] = d[n] - y[n]$$
$$\mathbf{x}[n+1] = \mathbf{x}[n] + \mu e[n] U[n]$$

Cell-to-cell interference cancelation:

$$V[n] = V[n] - \mathbf{x}[n+1] \cdot \mathbf{U}[n]$$

end for

3. LEAST SQUARES BASED CELL-TO-CELL INTERFERENCE CANCELATION METHOD

In this section, we develop a least squares based cell-to-cell interference cancelation method. Since the entire data samples of a page are acquired at the same time in NAND flash memory, the LMS filtering based approach that uses only one data sample at a time is quite inefficient.

The least squares method is a batched process that requires data in advance, extra memory space for buffers, and time to gather them. Since NAND flash memory has page buffers, no extra delay and memory space are required when the least squares method is applied. In order to take advantage of this architectural feature, Eq. (4) can be redefined as follows.

$$J(\mathbf{x}) = \frac{1}{2}\sum_{n=0}^{N_s-1} |e[n]|^2. \tag{6}$$

Algorithm 2 Least squares based cell-to-cell interference cancelation method

Randomly choose N_s samples and compute:

$$\mathbf{x}_* = (A^T A)^{-1} A^T \mathbf{b}$$

Cell-to-cell interference cancelation:
for ($n = 0$ to $N-1$)

$$V[n] = V[n] - \mathbf{x}_* \cdot \mathbf{U}[n]$$

In this definition, N_s data samples are used to define the cost function. The cost function in the LMS coupling canceler uses only a single data point, thus the adaptation is slow. Since $\mathbf{U}[n]$ and $d[n]$ are not the exact but estimated ones as explained in the above, there can be many outliers that prohibit the adaptation process. Unlike the LMS one, the average error of N_s data samples are used in Eq. (6), thus we can expect a more reliable solution in the least squares based approach. Obviously, this method tends to be more reliable as more samples are used. In order to derive the least squares solution, the above equation can be rewritten in a matrix-vector form as follows.

$$J(\mathbf{x}) = \frac{1}{2}(A\mathbf{x} - \mathbf{b})^T \cdot (A\mathbf{x} - \mathbf{b}) = \frac{1}{2}\|A\mathbf{x} - \mathbf{b}\|_2^2, \tag{7}$$

$$\text{where } A = \begin{bmatrix} \mathbf{U}[0]^T \\ \mathbf{U}[1]^T \\ \vdots \\ \mathbf{U}[N_s-1]^T \end{bmatrix} \text{ and } \mathbf{b} = \begin{bmatrix} d[0] \\ d[1] \\ \vdots \\ d[N_s-1] \end{bmatrix}.$$

Note that A is an N_s by M matrix and $\mathbf{b}$ is an N_s dimensional vector. Eq. (7) is known as the least squares, and many algorithms have been developed in order to find $\mathbf{x}$ that minimizes the cost function. Since this is a linear system, the analytic solution of Eq. (7) can be derived simply.

$$\mathbf{x}_* = (A^T A)^{-1} A^T \mathbf{b} \tag{8}$$

Once the optimal solution $\mathbf{x}_*$ is computed, then we can move onto the interference cancelation process that is almost the same with the LMS filtering one. During this process, $\mathbf{x}_*$ remains the same, and the estimated interference is subtracted from the victim cell's threshold voltage. Algorithm 2 describes the least squares based approach.

The time complexity of the LMS algorithm is $O(NM)$, where M is either 5 (even cells) or 3 (odd cells). This is a sequential algorithm, thus the arithmetic overhead remains as $O(N)$ per processor even if we employ more than M processing units. If N_s data samples are used to obtain the coupling coefficients, the time complexity of the least squares approach is $O(N_s M^2)$ according to Algorithm 2, where $N_s M^2$ and M^3 arithmetic operations are required for computing $A^T A$ and the matrix inverse, respectively. However, the former is dominant because N_s is usually much larger than M. If N_s is smaller than $\frac{N}{M}$, the arithmetic overheads of both algorithms become comparable. In addition, when P (larger than M) processing units are applied in parallel, the number of arithmetic operations per processor in the least squares method becomes $\frac{M}{P}$ times of LMS one's.

4. EXPERIMENTAL RESULTS

We conducted Monte-Carlo simulations for verifying the performance of CCI cancelation techniques, and used a four-level MLC

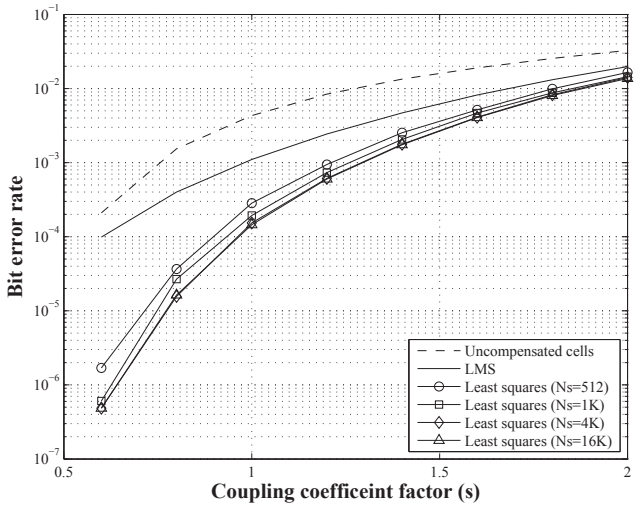

Fig. 5. Bit error rate performance of the even cells.

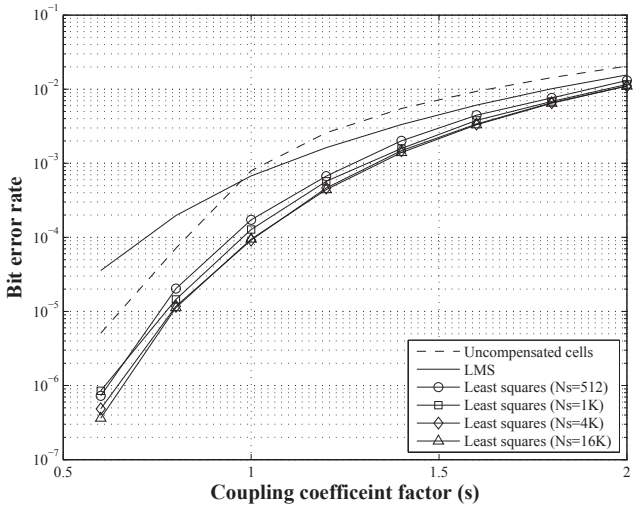

Fig. 6. Bit error rate performance of the odd cells.

NAND flash memory cell model described in Section 2. One memory block consists of 32K bit-lines and 64 word-lines. Since the memory block has the even/odd bit-line structure, it contains 256 pages ($= 64 \times 2 \times 2$), and each page can store 16K bits. The erase operation is regarded as a Gaussian random process, hence each cell's threshold voltage is sampled by a normal distribution whose mean and standard deviation are 0.0 V and 0.30 V, respectively. During programming, 2.55 V, 3.15 V, and 3.75 V are used for the target voltages (V_1, V_2, and V_3, respectively). $E[\gamma_x]$, $E[\gamma_y]$, and $E[\gamma_{xy}]$ are set to $0.05s$, $0.1s$, and $0.025s$, where the coupling coefficient factor, s, varies from 0.6 to 2.0. We measured BER for each method, and N_s is changed from 512 to 16K points. For comparison purpose, BER of the uncorrected cells is also measured.

Fig. 5 shows the BER performance of even pages. It is clear that the least squares method shows better BER performance than the LMS filtering approach even when only 512 cells are used to estimate the coupling coefficient $\mathbf{x}$. The BER performance gap between the two methods is orders of magnitude when the coupling coefficient factor is small, and it tends to be smaller as the CCI becomes larger. As N_s increases, the least squares method can correct more bit errors; however the performance is saturated when N_s is larger than 4K.

The BER performance of odd pages is shown in Fig. 6. The results are similar with those of the even pages; the least squares method outperforms the LMS filtering one. Actually, LMS filtering generates more errors rather than corrects them when s is below 1.0. Therefore, it is better not to use LMS filtering when the capacitance coupling is weak.

5. DISCUSSION

Even though both the LMS and the least squares based cell-to-cell interference cancelation techniques are quite effective, these methods are based on an assumption that the threshold voltages are acquired in a high precision. In order to sense the voltage precisely, many read operations are needed in conventional NAND flash memories, which can significantly increase the read latency. Hence, it is very needed to develop a fast way of conducting multiple reads. To find an optimal quantizer that minimizes the number of voltage sensing while showing reasonable BER performance remains as the future works.

6. REFERENCES

[1] D. Park and J. Lee, "Floating-gate coupling canceller for multi-level cell NAND flash," *IEEE Transactions on Magnetics*, vol. 47, no. 3, pp. 624–628, 2011.

[2] R. Bez, E. Camerlenghi, A. Modelli, and A. Visconti, "Introduction to flash memory," *Proceedings of the IEEE*, vol. 91, no. 4, pp. 489–502, 2003.

[3] J.D. Lee, S.H. Hur, and J.D. Choi, "Effects of floating-gate interference on NAND flash memory cell operation," *IEEE Electron Device Letters*, vol. 23, no. 5, pp. 264–266, 2002.

[4] K. Prall, "Scaling non-volatile memory below 30nm," in *Proceedings of 22nd IEEE Non-Volatile Semiconductor Memory Workshop*, 2007, pp. 5–10.

[5] R.A. Cernea, L. Pham, F. Moogat, S. Chan, B. Le, Y. Li, S. Tsao, T.Y. Tseng, K. Nguyen, J. Li, et al., "A 34 MB/s MLC write throughput 16 Gb NAND with all bit line architecture on 56 nm technology," *IEEE Journal of Solid-State Circuits*, vol. 44, no. 1, pp. 186–194, 2009.

[6] K.T. Park, M. Kang, D. Kim, S.W. Hwang, B.Y. Choi, Y.T. Lee, C. Kim, and K. Kim, "A zeroing cell-to-cell interference page architecture with temporary LSB storing and parallel MSB program scheme for MLC NAND flash memories," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 4, pp. 919–928, 2008.

[7] G. Dong, S. Li, and T. Zhang, "Using data postcompensation and predistortion to tolerate cell-to-cell interference in MLC NAND flash memory," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 10, pp. 2718–2728, 2010.

[8] K.D. Suh, B.H. Suh, Y.H. Lim, J.K. Kim, Y.J. Choi, Y.N. Koh, S.S. Lee, S.C. Kwon, B.S. Choi, J.S. Yum, et al., "A 3.3 V 32 Mb NAND flash memory with incremental step pulse programming scheme," *IEEE Journal of Solid-State Circuits*, vol. 30, no. 11, pp. 1149–1156, 1995.