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, 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 \( \Delta V_{pp} \) 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...
In order to derive the LMS solution for the CCI cancelation [1], let us define a vector as follows:

\[ U[n] = [\Delta V[m + 1, n - 1], \Delta V[m + 1, n], \Delta V[m + 1, n + 1]]^T. \]  

(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, $U[n]$ 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:

$$y[n] = x[n] \cdot U[n],$$

where $x[n]$ represents the coupling coefficient in a vector form. Again, we replace $V_L[m, n]$ 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.

$$J(x) = \frac{1}{2} |e[n]|^2,$$


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

$$x[n + 1] = x[n] + \mu e[n] U[n]$$

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

**Algorithm 1** Least squares based cell-to-cell interference cancelation method

```plaintext```
for ($n = 0$ to $N - 1$)
    LMS filtering:
    $y[n] = x[n] \cdot U[n]$
    $e[n] = d[n] - y[n]$
    $x[n + 1] = x[n] + \mu e[n] U[n]$

Cell-to-cell interference cancelation:
$V[n] = V[n] - x[n + 1] \cdot U[n]$
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.

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

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 $U[n]$ 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.

$$J(x) = \frac{1}{2} (Ax - b)^T (Ax - b) = \frac{1}{2} \|Ax - b\|^2,$$

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

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

$$x_* = (A^T A)^{-1} A^T b$$

Once the optimal solution $x_*$ is computed, then we can move onto the interference cancelation process that is almost the same with the LMS filtering one. During this process, $x_*$ 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 $O(N M)$, 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 $N_s M^2$ 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 $\frac{M}{2}$, 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
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 × 2 × 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 \(N_x\) 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 factor. 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 \(N_x\) increases, the least squares method can correct more bit errors; however, the performance is saturated when \(N_x\) 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