Topologically Defined Flash Memory

Yanjun Ma
Interlake Research, Bellevue, WA 98005, Email: yanjun.ma@interlakeresearch.com

***Abstract*—** We discuss a topological method of storing and retrieving information from flash memory. We first present a sensing method where the threshold voltage level, as represented by a sensing time, is extracted in one sensing cycle. The sense time distribution from one set of flash cells, e.g. one physical row, is then processed in software to decode the digital state of each cell. The decoding method uses topological constraints but no rigid or predetermined voltage thresholds to digitize the distribution. The software defined nature of the topologically defined flash (TDF) allows greater flexibility for allocating cells to arbitrary number of digital states.

Keywords – flash memory; topological flash memory; sensing; error correction

I. Introduction

Present flash memory uses an analogue quantity, the threshold voltage of a transistor, to store digital data by dividing the available range of the threshold voltage into two (for SLC), four (MLC), eight (TLC), or even 16 (QLC) levels. Hardwired voltage thresholds are usually assigned to define the boundaries between levels. Cell level sensing method comprises of multiple sensing operations where successive read voltages are applied to the control gate of the flash memory transistor. A sense amplifier is used to determine if the memory cell is conducting. Each sensing operations determines for each cell whether the cell threshold voltage is below (conducting) or above (cell not conducting) the level threshold and assign a digital value. More recently, fixed thresholds have been replaced by more adaptive and adjustable read threshold and with the option for multiple read using the read-retry (RR) feature. The read retry method can be used to obtain the actual threshold voltage, but due to the long time needed for the multiple RR it has been used only in a limited fashion or in characterization only.

We discuss an alternate way of storing and retrieving information from flash memory cells without the use of voltage thresholds [1]. We first present a sensing method where the threshold voltage level, as represented by a sensing time, is extracted in one sensing cycle. The sense time distribution from one set of flash cells, e.g. one physical row, is then processed in software to decode the digital state of each cell. The decoding method uses topological constraints but no rigid or predetermined thresholds to digitize the distribution. Since this is done in software this method has the advantage of tight integration with ECC methods, e.g. those described in [2,3]. The software defined nature can allow greater flexibility for allocating cells to almost arbitrary digital states. Additionally, no precision voltage standard and associated design complexity is necessary to implement our scheme. Finally, we discussed several methods for programming such topological flash memories.

The analogue nature of the threshold voltage has been explored in the rank modulation (RM) scheme as an alternative representation of data in flash memory arrays [4]. RM encodes data using the relative threshold voltage of memory cells and can be regarded as a topological based method. In this sense, the method discussed in this paper can be considered an extension of the topological paradigm.

II. Threshold Sensing Methods

In this section, we present a method that achieves sensing of the threshold voltage for all cells in a physical row, as defined by a word line, in one sensing operation. The method keeps track of the timing (through counting the pulses of a reference clock) the sensing (tripping of sense amplifier or comparator) is done on each cell. The preferred sensing process is illustrated with the flash array in Figure 1.

The array and major circuit blocks are similar to the current NAND flash chips. Here the sensing result of each column is the output of a clock counter at the trip point of the respective sense amplifier. The counter data are logged to the associated column multi-bit register. The clock is also used as the clock controlling the ramping of the selected wordline voltage. Alternatively, an ADC converter can be used to convert the wordline voltage to digital output at the time of sense amplifier tripping [5].

The associated timing diagram is shown in Figure 2. Here optional three reference cells are shown which can be used to determine the actual threshold voltage value. When the threshold voltage of the reference cells is set at the read verify levels of the MLC cell the method recovers to the conventional outcome and senses digital state of the cell directly.

The precharge process is similar to the present practice. The Vpass is set to allow the transistors on the unselected rows to turn on. At the start of the evaluation period, the counter starts to counter the clock pulses as Vread starts to ramp. At this point some cells, e.g. erased cells with negative Vt, may be turned on and start to discharge the respective bitline (or the dedicated capacitor in the all-bit line architecture). An example is Col0 in Figure 2. The Vread continues to ramp, more bitlines will start to discharge. As the bit line voltages decrease to the trip point (set at an appropriate value, Vref) of the sense amplifier, the output of the sense amplifier changes and triggers the data latch on the corresponding column to latch the counter output for each column. The latched sense time data maybe streamed out during or at the end of the evaluation cycle. The output from the sensing block is the sensing time of each cell that is measured in terms of clock cycle of the reference clock.

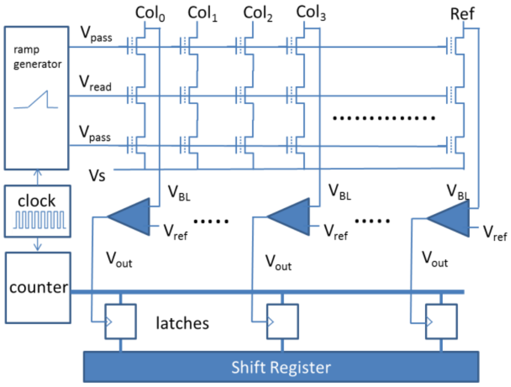

Figure 1. Flash memory array architecture

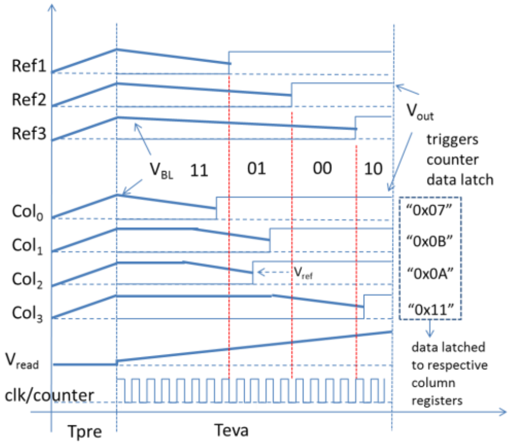

Figure 2 Timing diagram of a read operation

III. Data Processing

The sensing time from the memory cells are used for further analysis, including the assignment of digital state for each cell. This is illustrated with a MLC array in Figure 3, where the sense time distribution histogram of all the cells of a given physical row is plotted. The distributions are broad to simulate effects of transistor mismatch due to process variation or random telegraph noise; programming/erase cycle induced oxide damage, and cell to cell (C2C) interference.

From this type of figure, several methods can be applied to assign a digital state to each cell. Figure 3 illustrates three methods. The first is to assign the boundaries between the digital states at the minimum of the distributions, as shown by the vertical arrows, that are used to separate different levels. This method is similar to current practice, except that here the thresholds between the level are not voltages and are not defined *a priori*, and that the thresholds can be defined in software. Hence this method is much more flexible and can be more easily adapted to work with ECC methods, such as LDPC.

The second method requires no threshold between different states. But topological constraints in the form of additional information (metadata) that are stored when the rows are programmed. For example, during programming, the number of cells with "11", "01", "00", and "10" states (e.g., N11, N01, N00, N10) are counted and stored [4]. During analysis we start to count the number of cells from the lowest Vt (normally the erased state). The first N11 cells are assigned to the "11" states, the next N01 cells are assigned "01" states, and so on.

In the third method, we fit the overall distribution by known or a model cell distribution, e.g. Gaussian distributions including a retention model, and assign cells fall into different distributions to different digital states. The metadata, number of cells in each level, can be used as the constraints in the fit.

For the cells at the level boundary, maximum likelihood method can be used for the assignment in all three methods. All three methods are topological in nature in the sense that the distribution may expand or shrink, as indicated by the red arrow in Figure 3, or shift, as indicated by the green arrow in Figure 3, assuming they do not move cross each other without affecting the outcome of the analysis. Only the relative positions of the distributions are used in the digital state assignment.

Further processing can be done with optional reference cells, or through characterization of threshold voltage and sense time. With the additional calibration, the actual threshold voltage of each cell can be obtained. Once the threshold voltage values for a cell and the threshold voltage values of its surrounding cells are known, the shift of the threshold voltage due to the cell to cell interference can be determined. De-convolution of the C2C interference effect removes its broadening effect and helps with the accurate assignment of digital states.

IV. Programming for TDF

With the new sensing scheme and the topological methods for defining the cell state, the programming of the cells is more flexible in the topologically defined flash (TDF) memory. Ignoring the broadening effect due to retention and other disturbing mechanisms, to maintain data integrity, one has only to make sure that the programming keeps consistent topological order for the programmed cells.

Note that the sensing method discussed in Section II dictates that we program by a whole physical row, rather the devide the row into pages as in the conventional methods. There are several methods that can be applied to program the row:

1. Conventional method: divide the voltage memory window by the number of state to be programmed into each row, and use reference voltages to program the cells. The conventional program-verify process and the program verify threshold can be used. In this method, the windows are divided according to the occupation of the level. For example, the window size of a particular level may be inversely proportional to the number of cells that are to be programmed into that level. So that the probability of the

distribution overlapping with adjecent levels are maintained to be the same across all levels. In an extremely example, suppose that there are no "01" cells in a particular row, the storage window of the whole row can then be decided into only three levels, instead of the rigid four levels in the conventional MLC memory. The topological constraint of $N_{01} = 0$ is used during memory read to make sure that only three distributions are used to decode the content of this row. Because of the larger window for the occupied levels, there memory will intrinsically be more reliable.

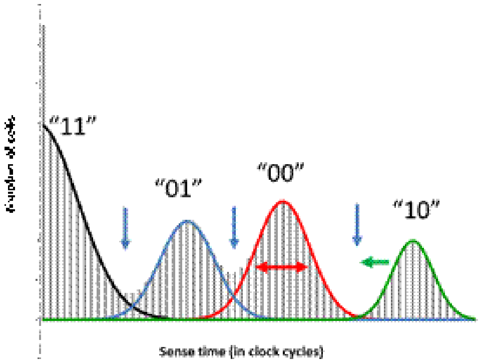

Figure 3. Simulated sense time distribution (bars) of a MLC row. Vertical bars indicate the likely level boundaries that are used to digitize this distribution. Because of the topological nature of the assignment methods discussed in the text, expand/shrink or shift of the level distribution do not affect the outcome of the cell assignment. Only the relative position of the distribution matters.

2. Programming by relative ranking, similar to that for programming an RM memory [7,8]. For the MLC example, starting with an erased distribution of "11" cells, we program all the "01" cells using incremental pulse until have cells have a longer sensing time than that of the top most "11" cells; we then program all the "00" cells so that all have longer than sensing time than all the "01" cells, etc. A margin may be added to increase the reliability of the memory.

For the method #2, there are no fixed thresholds for the whole memory, and the sensing time comparison are localized to each row. In principle, this should yield better reliability to the whole memory.

V. Discussion and Conclusions

In this paper we discussed a new scheme to storing and sensing information in flash memories. The presented sensing method can be accomplished in one sensing cycle and provide a quasi-analogue quantity (the sensing time) for each cell in a memory row that represent the threshold voltage of the memory transistor. The sensing time can be used to define/assign the digital level/state of the memory cell by using topological constraints. Because the assignment uses topological constraints and is performed on each physical row, we believe the TDF is inherently more reliable than conventional flash memory, where fixed thresholds for the whole memory arrays are used.

Since the digital level assignment in TDF are done in software, this greatly increases the flexibility. The sensing time data (or Vt data if reference cells are used) from a flash row can be manipulated and interpreted in software. The threholds for distinguishing each digital levels are adaptable for each row and can, in fact, be for partial rows. For example, each physical row of a flash array can be software definable as SLC, MLC, TLC, etc. For example, we can choose one row to be a MLC row and next row to be a TLC row. A row can be a MLC today and redefined to be a TLC row tomorrow. In fact, each row can even contains sections of SLC, MLC, or TLC cells. For example the first half of the row can have SLC cells while the second half can be TLC cells.

To implement such new scheme, as can be seen in Figure 1, the memory architecture is the same as the conventional flash memory. Only sensing method is slightly modified. On the other hand, since no precision voltage references are needed in the TDF, including the requisite temperature compensate circuit, the sensing circuit is easier to implement and more reliable.

We should caution that broadening of the distribution due to leakage, in particular the tail bits, can greatly complicate the analysis and may render some of the advantages mute. During programming meta data may be stored, e.g. number of cells in each state and the cell to cell interference parameters. The pre-interference data may be stored and used to extract the interference parameters after all the cells are programmed.

In conclusion, we presented a topological and software defined scheme to store information in flash memory. This scheme has the advantage of increased reliability, simplicity, and potentially higher capacity than the conventional flash memory.

Acknowledgment

The author would like to thank Yue Li, A. Jiang, E.C. Kan, and J. Bruck for valuable discussions.

References

[1] Y. Ma, US Patent No: 9,824,750

[2] H. Zhou, A. Jiang, and J. Bruck, "Error-correcting schemes with dynamic thresholds in non-volatile memories," IEEE ISIT , St. Petersburg, 2011.

[3] F. Sala, R. Gabrys, and L. Dolecek, "Dynamic threshold schemes for multi-level non-volatile memories," IEEE Trans. on Communications, vol 61, 2624 – 2634, 2013.

[4] A. Jiang, R. Mateescu, M. Schwartz, and J. Bruck, "Rank modulation for flash memories," *IEEE Trans. on Inform. Theory*, vol. 55, no. 6, pp. 2659–2673, 2009.

[5] G. Naso, et al, Micron 20 nm NAND, *ISSCC* 2013; Sarin et. al., "Sensing memory cells," US Patent 7,948,802.

[6] Yu Cai, et al, "Threshold Voltage Distribution in MLC NAND Flash Memory: Characterization, Analysis and Modeling" Proceedings of the Design, Automation, and Test in Europe Conference (DATE), Grenoble, 2013.

[7] Y. Ma, U.S. Patent No. 9,202,558;

[8] Y. Ma, Y. Li, E. C. Kan and J. Bruck, "Reliability and hardware implementation of rank modulation flash memory,"*15th Non-Volatile Memory Technology Symposium (NVMTS)*, Beijing, 2015.