Accelerating Flash Memory Access by Speculative Early Sensing Decision

Amit Berman and Yitzhak Birk
Electrical Engineering Department, Technion – Israel Institute of Technology
{bermanam@tx, birk@ee}.technion.ac.il

As Flash technology shrinks, storage capacity increases. However, cell-size variability increases, making it harder to accurately inject the desired amount of charge. Multi-level cells (MLC), whose reading and programming require multiple reference-voltage comparisons, therefore exhibit increased write, read and erase latencies.

There are various approaches for reducing Flash access time. NOR Flash enables high speed read access at the cost of a twofold density reduction than NAND and a lower program speed [1]. Throughput improvements include cache read and program [2], but those are efficient for sequential access whereas access patterns can be random [3]. Device approaches include program pulse magnitude optimization [4], but its efficiency decreases with an increase in cell-dimension variability. Architectural approaches include multi-page programming [5], inter-cell interference encoding [6], multiple plains on a chip [7], multi-channel and multi-chip architectures.

Our approach, speculative early sensing, is based on the variability of the achievable read time of non-proximal memory cells due to process variation as well as its variation along the lifetime of a cell, along with the fact that specified fixed read times are based on worst-case cell characteristics. The idea is to speculatively read prematurely, and positively detect errors if any and retry (with a longer read time), thereby reducing average access time while ensuring correctness.

NAND flash read comprises two stages: 1) data transfer from memory array to on-chip volatile buffer (SRAM); 2) data transfer from this buffer to I/O pins. Our focus is on stage 1, which is done simultaneously for an entire page, and typically lasts 25-50µs. Our technique is next explained using SLC NAND Flash, but is also applicable MLC and to other memory types, including NOR Flash.

Data transfer from the array to the buffer comprises two phases: precharge and evaluation. Reading is performed simultaneously for a row of cells (page). Each cell in a row is associated with a unique column (NAND string). Fig. 1 depicts the Read data path of a single cell. A string is modeled as a current source. Prior to reading, the RESET transistor is on, setting a default value ‘1’ (‘0’) in the DATA (DATA_N) node.

During precharge (~5µs), I_{CELL}=0. The SEL transistor is biased to V_t, and the PRE transistor is kept grounded. Consequently, C_{out} is charged to V_{DD}, and the bitline parasitic capacitor C_{BL} is charged to V_{th}-V_{th,sel} (where V_{th,sel} is the threshold voltage of the SEL transistor). At the end of precharge, PRE and SEL transistors are switched off, while the OUT and BL nodes are floating, precharged at their initial voltage values.

During the evaluation phase (duration T_{EVAL}), either I_{CELL}>0 (erased cell) and it discharges C_{BL}, or else I_{CELL}=0 (programmed cell) and C_{BL} retains its initial charge. After T_{EVAL} time, the gate of the SEL transistors is biased to V_2 (V_2>V_1). If C_{BL} was discharged then V_2>V_{BL}>V_{th,sel}, so the SEL transistor is switched on and the voltage of the output node will match the bitline voltage V_{OUT}=V_{BL}. Else, C_{BL} was not discharged, and the output node will keep its initial value V_{OUT}=V_{DD}. Next, READ transistor is switched on, and the value of OUT node is converted to compatible logic levels and latched. If the value of OUT node is V_2>V_{th,sel} or less, the output DATA is 1. Else, DATA is 0. T_{EVAL} is given by

\[ T_{EVAL} = C_{BL} \left( V_1 - V_2 \right)/I_{CELL}. \]

Fig. 1. NAND Flash read data path. NAND string is modeled as a current source, bitline capacitance is marked with C_{BL}. Read is divided into two stages: precharge, where PRE is at ground and SEL is open with V_1, so that C_{BL} and C_{OUT} are charged. Evaluation phase where PRE is off, and SEL is biased with V_2>V_1, such that if the cell is erased, i.e. I_{CELL}>0, C_{BL} is discharged. In this phase, if V_{BL}<V_2-V_{th,sel}, then V_{OUT}=V_{BL} and the sensed data is 1. Else, the sensed data is 0.
Since default read value is 1, the errors resulting from shortened evaluation are uni-directional (0→1), as depicted in Fig. 2. We add a \( \log_2(N+1) \) bit signature whose value is the number of 0’s in the N data bits (Berger code [12]). Since the number of 0’s in the data can only increase and the signature’s value can only decrease, any number of 0→1 errors is detected.

![Fig. 2. Flash read channel reflecting shortened read attempts.](image1)

Our proposed algorithm is depicted in Fig. 3.

![Fig. 3. Speculative early sensing read algorithm. Prior to using the algorithm, data is programmed with additional Berger code check bits. In case of detected errors, sensing time is increased and re-read is applied. The process is repeated until no errors detected or when full sense time is used.](image2)

We simulated page evaluation, assuming all cells in the same page are with small variations. We used conservative cell variations [9,10,11]. \( C_{BL} \) is normally distributed (2.5pF, 0.05pF), and clipped to [2pF, 3pF]. \( I_{CELL} \) is assumed to be normally distributed (500nA, 150nA), and clipped to [100nA, 1µA]. We calculate the probability of page to have at least one error (and thus requiring a second read attempt) as a function of shortened read time (Fig. 4).

Consider, for example \( T_{read}=20\mu s \), of which \( T_{eval}=15\mu s \) and no errors. Instead (Fig. 4), we use \( T_{eval}=5\mu s \) \( (P_s=0.05) \) and revert to 15µs upon failure. The average read time is \( T_{read}=5+5(1-P_s)+(5+15)P_s=10.75\mu s \), a 46% improvement.

The accelerated read can also serve for program acceleration by using shortened data verify. Here, no Berger code check is necessary. Read verify accounts for nearly 40% of program time. Because of the (only) possible error direction, only an erased cell can be mistaken to be programmed. Therefore, errors during program are not irreversible, as a cell is never overcharged. At the apparent end of program, full sensing time must be applied. Considering the previous example, program time is improved by 0.46-0.4=18%.

![Fig. 4. Simulated page error probability vs. evaluation time. \( C_{BL} \) is normally distributed with mean 2.5pF, \( \sigma=0.05pF \). If generated values is out of\([2pF,3pF]\) interval, they are changed to closest interval's bound. \( I_{CELL} \) is assumed to be normally distributed with mean 500nA, \( \sigma=150nA \), and limit the bounds to be \([100nA,1\mu s]\). Speculative early sensing can be used to tailor performance to the capabilities of different areas on a given chip as well as to its age. Cost is negligible. Future work includes optimization of speculative read algorithm and use of uni-directional error correction code.

![REFERENCES](image3)