Bit Error Generation for NAND Flash Memory

Kijin Kim and Seung-Ho Lim
Division of Computer and Electronic Systems Engineering,
Hankuk University of Foreign Studies
do131104@naver.com

Abstract. As NAND flash memory becomes popular from embedded systems to high performance computing, reliability becomes highly important due to its limited endurance. In this paper, we provide an error generation model based on Bit Error Rate (BER) of real NAND flash devices which can be used for on data corruption recovery of NAND Flash Memory.

Keywords: NAND flash memory, BER, NANDSim, PE

1 Introduction

Rapid increase in preference of NAND flash memory in IT systems results in high requirement on data reliability. A typical data storage device built with NAND flash memory comes with reliability problems due to its limited endurance[1]. A flash memory chip uses Single-Level-Cell (SLC), Multi-Level-Cell (MLC) or Triple-Level-Cell (TLC) technology which can endure up to 100,000, 10,000, 1,000 PE cycles respectively, and repeated read, program and erase operations gradually damage flash cells resulting in increased BER. For better endurance to bit errors, memory controllers included in the storage devices generally use Error Correcting Codes to detect these errors and attempt to correct data. This error correction is one of the many ways to expand lifetime of NAND flash device.

This paper introduces a bit error generation for NAND flash to test software-simulated NAND flash devices. The need of this error model rises from current error reporting scheme in NANDSim that it reports error only when a block has reached its PE cycle limit where it is crucial to monitor BER under PE cycle limit for thorough analysis of NAND flash device. The proposed bit error generation model replaces current scheme and enables monitoring BER over increase in PE cycle.

2 Background

NAND flash storage devices provide superior performance to IT systems over hard disk drives. In this storage unit, cells in NAND flash memory are organized in serial lines. The basic operation units used in flash memories are page and block. A page is a group of cells in neighboring serial lines, and a block is a group of these pages. The
device undergoes three primitive operations, read, program and erase. The basic read
and program unit is a page whereas erase operations are done in blocks. This complex

Organizing cells in a serial manner enables manufacturers to build large storage
devices with high cell density[2]. Despite the increase in storage size, as cells are
close to one another, they are exposed to interference with neighbor cells when it is
programmed, erased or read which results in bit errors. Aside from the bit error
caused by neighboring cells, the most dominant physical error pattern in NAND flash
is the retention error that is caused by charge leakage in NAND flash cells over time.
As to ensure data reliability on storage devices, it is essential to monitor bit error
patterns of NAND flash memory.

One of the main concerns of NAND flash memory based storage devices is dealing
with bit errors. It is crucial for manufacturers to detect and recover from these bit
errors. The ECC is the most common mechanism used in data storage to detect and
correct data corruption [3]. As to ensure data reliability on storage devices, it is essential to monitor bit error patterns of NAND flash memory and apply suitable ECC
accordingly. This paper provides the error model and log for bit error that can be
further used to test ECC performance.

The NANDSim is a simulator included in Linux kernel that is used to generate
software-simulated memory devices that behave like NAND flash devices. The error
report scheme in NANDsim works as follows. It keeps a threshold erase limit
according to the NAND flash type and counts number of erase on every block in the
simulated device. When the erase count on a block reaches the threshold, the
simulator categorizes the block as bad block and reports error if any further erase
attempt are done. This scheme is used to generate and report error in a block when it
reaches PE cycle limit, and not for bit errors that occur during increase in PE cycle.
The absence of generating and reporting bit errors throughout PE cycle is not suitable
to be an ideal simulated NAND flash device. This paper supplies a scheme that
generates error as in real devices.

3 Error Generation Model

The model is built and applied according to the following two steps, BER sampling
and introducing new error generation scheme.

First, in process of building up an error model, samples from BER versus PE cycle
data are taken from the real NAND flash devices [4][5]. As storage devices based on
NAND flash use SLC, MLC or TLC types, three separate data groups are collected.
Sample groups are taken as set of bit error probabilities at given PE cycles according
to the flash type.

Before every program operation, erase operation is performed, and every block
involved is assigned with an entry which is managed in a linked list to keep track of
erase counts. According to flash type configuration in NANDSim, it also keeps
threshold erase counts. This linked list of erase counts is used to report error when a
block reaches its erase count limit. This scheme allows error report only when erase
operation is done over erase count threshold.
In new scheme, we have added a routine in program operation which looks up each entry of the block involved in the operation for their erase counts and apply bit error probability accordingly and keep track of bit error generated at given erase count for further uses.

4 Experimental Results

We have taken thirteen and twenty-four and twenty-one BER samples from real BER data of SLC and MLC devices respectively, and applied bit error probability in NAND flash device simulated with NANDSim.

In this section, the BER of the simulated device is compared with sampled BER. Figure 1 shows the sampled BER from real SLC and MLC devices in blue line and BER in simulated NAND flash devices in red respectively. In each graph, BER of the simulated device shows the set of sampled bit error probabilities are indeed applied.

Fig. 1.

Figure 1 Simulation Result for SLC and MLC Devices

5 Conclusion

Storage devices based on NAND flash memory experience data corruption problems. To ensure data reliability, it is essential to detect bit error and introduce suitable recovery scheme. In this paper, we focus on NANDSim in Linux kernel which is used to evaluate NAND flash devices. NANDSim, in its current state, has no scheme to generate bit error before reaching its PE cycle threshold. With proposed bit error model scheme, NANDSim is improved to simulate NAND flash devices that can be used in BER and ECC research.