

# Microscopic Bit-Level Wear-Leveling for NAND Flash Memory

Yong Song<sup>1</sup>, Woomin Hwang<sup>1</sup>, Ki-Woong Park<sup>2</sup>, and Kyu Ho Park<sup>1</sup>

<sup>1</sup> CORE Lab., Dept. of Electrical Engineering, KAIST, Korea

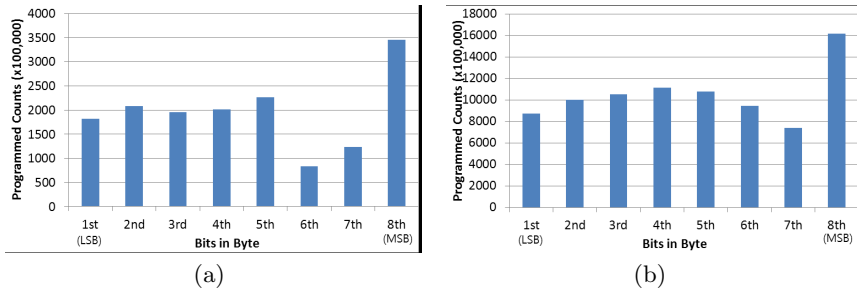
<sup>2</sup> Dept. of Computer Hacking and Information Security, Daejeon University, Korea  
{ysong,wmhwang,kpark}@core.kaist.ac.kr, woongbak@dju.kr

**Abstract.** By microscopically observing widely used data files, we identified the considerable room for life time improvement in NAND flash memory, which is due to the discovery of a non-uniformity in bit-level data patterns. In an attempt to exploit the discovery, we propose a novel bit-level wear-leveling scheme. Instead of considering only the view of page-level or block-level, we incorporate the non-uniformity in data encoding patterns into wear-leveling scheme. Because of its orthogonality to the existing block-level wear-leveling approaches, our solution can be adopted over the existing solutions without considerable overhead and extend NAND flash's life span up to 36% in case of SLC.

## 1 Introduction

During the last decade, NAND flash memory has become a de facto solid state storage technology. In the NAND flash memory-based storage systems, it is essential to reliably store data for years, however, NAND flash has endurance problem that each NAND flash cell is worn-out by program/erase cycling and eventually loses its capability to store data. Thus, the endurance problem of NAND flash has been considered in many previous researches and they proposed wear-leveling algorithms, which even out the wearing of different blocks of NAND flash memory.

To enlarge NAND flash's life span, many researches proposed their algorithms for wear-leveling which tries to make wear-out of different blocks of flash memory even [3][4][5][6][7]. To expand lifetime of NAND flash blocks, commonly, the previous algorithms take page-granularity approaches and use block erase count as a wear-out indicator. This is based on an implicit assumption all cells in same page or same block have the same probability to be worn-out by P/E cycle. To identify whether the assumption is right or not, we obtained data patterns of widely used file data frequently written to disk or accounting for large part of storage, for example, linux source codes, database workloads, encoded video files. After analyzing the data, we found out a case that the bit-level data pattern is not uniform. Base on our observation, we build a simple statistical wear-out model of NAND flash cell and we propose bit-level wear-leveling scheme. The evaluation shows that it expands life span of NAND flash up to 30%.



**Fig. 1.** (a) Bitwise counts of programmed state of Linux source codes, (b)Bitwise count of programmed state of DB workloads

## 2 Motivation

In case of SLC NAND flash, a cell stores a bit and a page is composed of a series of NAND cells and a block contains multiple pages. To store data on them, a block erase operation should be performed prior to write data, which makes all cells in the block changed to the erased state, representing logic ‘1’. If logical value of ‘0’ is written to a cell when page writing operation of NAND flash, a state transition of the cell occurs from the erase state to the programmed state, indicating ‘0’. On the other hands, if logical value of ‘1’ is written to a cell, there is no state transition of the cell and it remains in the erased state. Because a state transition from erase state to programmed state makes a cell worn out[2], each cell can experience different wear-out progress according to the number of its actual programs.

A problem arises from that the previous block-based wear-leveling algorithms treat all cells comprising a block together as if they experience the same number of state transitions between the erase state and the programmed state, thus having the same degree of wear-out. If we assume that the physical endurance of each cell is same, among all cells comprising a block, the cell experiencing the highest number of programs is to be completely worn out most rapidly. If the number of completely worn-out cells is bigger than the number of correctable bit errors by bit error correction algorithm such as ECC or DHC, it is regarded that the life time of the page is ended. Even though other cells still have their life time left and they can operate normally, the page is not used any more. This leads to the underestimation of the life time of the target page. It therefore makes the NAND flash memory block lose chances to prolong lifetime of the block.

Outstanding frequency of programs in a specific bit position appears significantly in text-based workloads. We selected Linux source codes as a representative text-based workloads and obtained their bit-level data patterns by traversing Linux source code tree. Figure 1(a) shows the accumulated counts of the case of logical ‘0’ at each bit position per byte, which requires the state transition of the cell from erase state to the programmed state. We can see that the most significant bit (MSB) position in a byte(character) experiences the highest number