



Endurance Enhancement of Flash-Memory Storage Systems: An Efficient Static Wear Leveling Design

Yuan-Hao Chang, Jen-Wei Hsieh, and Tei-Wei Kuo

Embedded Systems and Wireless Networking Laboratory Dept. of Computer Science and Information Engineering National Taiwan University Taipei, Taiwan

Outline

- Introduction
- System Architecture
- Motivation
- An Efficient Static Wear Leveling Mechanism
- Performance Evaluation
- Conclusion

Introduction - Why Flash Memory

Diversified Application Domains

- Portable Storage Devices
- Consumer Electronics
- Industrial Applications
- SoC and Hybrid Devices

 Critical System Components











Introduction - Characteristics of Flash Memory



System Architecture



*FTL: Flash Translation Layer, MTD: Memory Technology Device

Policies – FTL

• FTL adopts a page-level address translation mechanism.

- Main problem: Large memory space requirement



Policies – NFTL

- Each logical address under NFTL is divided into a virtual block address and a block offset.
 - e.g., LBA=1011 => virtual block address (VBA) = 1011 / 8 = 126 and block offset = 1011 % 8 = 3



Outline

- Introduction
- System Architecture

Motivation

- An Efficient Static Wear Leveling Mechanism
- Performance Evaluation
- Conclusion

Motivation

- A limited bound on erase cycles
 - SLC : 100,000
 - MLC_{x2}: 10,000

- Applications with frequent access
 - Disk cache (e.g., Robson)
 - Solid State Disks (SSD)





Solid State Disk

Motivation – Why Static Wear Leveling: No Wear Leveling

Number of Write/Erase Cycles per PBA (75% Static Data)



Motivation – Why Static Wear Leveling: Dynamic Wear Leveling

Number of Write/Erase Cycles per PBA (75% Static Data)



Motivation – Why Static Wear Leveling: Perfect Static Wear Leveling

Number of Write/Erase Cycles per PBA (75% Static Data)



Motivation – Why Static Wear Leveling is So Difficult?

- An intuitive SWL
- Problems:
 - Block erase
 - Live-page copying
 - -RAM space
 - -CPU time





Motivation – Comparison



Outline

- Introduction
- System Architecture
- Motivation

An Efficient Static Wear Leveling Mechanism

- Performance Evaluation
- Conclusion

Design Considerations in Static Wear Leveling

- Scalability Issues
 - The limitation on the RAM space
 - The limitation on computing power
- Compatibility Issue
 - Compatibility with
 FTL and NFTL



An Efficient Static Wear Leveling Mechanism

- A modular design for compatibility considerations
- A SWL mechanism
 - Block Erasing Table (BET)
 - bit flags
 - -SW Leveler
 - SWL-Procedure
 - SWL-Update



The Block Erasing Table (BET)

• A bit-array: Each bit is for 2^k consecutive blocks.

- Small k - in favor of hot-cold data separation



- e_{cnt}: the total number of block erases done since the BET is reset
- f_{cnt}: the number of 1's in the BET
 - : an index to a block that the Cleaner wants to erase
 - : a block that has been erased in the current resetting interval

The SW Leveler

- An unevenness level (e_{cnt} / f_{cnt}) >= T
 → Triggering of the SW Leveler
- Resetting of BET when all flags are set.



Main-Memory Requirements

	512MB	1GB	2GB	4GB	8GB
k=0	256B	512B	1024B	2048B	4096B
k=1	128B	256B	512B	1024B	2048B
k=2	64B	128B	256B	512B	1024B
k=3	32B	64B	128B	256B	512B

 MLC_{x2} (1 page = 2 KB, 1 block=128 pages)

Outline

- Introduction
- System Architecture
- Motivation
- An Efficient Static Wear Leveling Mechanism

Performance Evaluation

Conclusion

Performance Evaluation

Metrics

- Endurance Enhancement (First Failure Time)
- Additional Overheads (Extra Block Erases)

Experiment Setup

– The FTL Layer

- FTL (Flash Translation Layer protocol)
- NFTL (NAND Flash Translation Layer protocol)

- Traces:

- A mobile PC with a 20GB hard disk
- Duration: One month.
- Daily activities
- The average number of Operations: 1.82 writes/sec, 1.97 reads/sec
- The percentage of accessed LBA's: 36.62%

- Flash Memory: 1GB MLCx2

Endurance Improvement - The First Failure Time



Additional Overhead - Extra Block Erases



Conclusion

• An Efficient Static Wear Leveling Mechanism

- Small Memory Space Requirement
 - Less than 4KB for 8GB flash memory
- Efficient Implementation
 - An adjustable house-keeping data structure
 - An cyclic queue scanning algorithm
- Performance Evaluation
 - Improvement Ratio of the Endurance:
 - The ratio is more than 50% with proper settings to T and K
 - Extra Overhead:
 - It is less than 3% of extra block erase with proper settings
 - There is limited overhead in live-page copyings

Conclusion

Future work

- Endurance and Reliability Problems due to Manufacturing and Capacity Improvements
 - Read/Write Disturbance
 Problems

-System Components

- Memory Hierarchy, Devices, Adaptors, etc.

-Benchmark Designs

- Targets on Different Designs



~ Thank You ~