Short Background on NAND Flash Memory Operation
NAND Flash Memory Background

Flash Memory

Block 0
Read
Pass
Pass
Pass
...  

......

Block N

Flash Controller

SAFARI
Flash Cell Array

Block X
Page Y
Sense Amplifiers
Row
Column
Sense Amplifiers
Flash Cell

Floating Gate Transistor (Flash Cell)

$V_{th} = 2.5$ V
Threshold Voltage ($V_{th}$)

Flash cell

Normalized $V_{th}$

Flash cell

1

0
Flash Read

\[ V_{\text{read}} = 2.5 \text{ V} \]

\[ V_{\text{th}} = 2 \text{ V} \]

\[ V_{\text{th}} = 3 \text{ V} \]

Gate

1

0
Flash Pass-Through

\[ V_{pass} = 5 \text{ V} \]

\[ V_{th} = 2 \text{ V} \]

\[ V_{pass} = 5 \text{ V} \]

\[ V_{th} = 3 \text{ V} \]

Gate
Read from Flash Cell Array

Correct values for page 2:

<table>
<thead>
<tr>
<th>Page 1</th>
<th>Page 2</th>
<th>Page 3</th>
<th>Page 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>V\text{pass} = 5.0</td>
<td>V\text{read} = 2.5</td>
<td>V\text{pass} = 5.0</td>
<td>V\text{pass} = 5.0</td>
</tr>
<tr>
<td>Pass (5V)</td>
<td>Read (2.5V)</td>
<td>Pass (5V)</td>
<td>Pass (5V)</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

V\text{pass} = 5.0

V\text{read} = 2.5

V\text{pass} = 5.0

V\text{pass} = 5.0
Aside: NAND vs. NOR Flash Memory

**NAND**

**NOR**
Threshold Voltage ($V_{th}$)

Flash cell

Normalized $V_{th}$

1

0

SAFARI
Threshold Voltage ($V_{th}$) Distribution

Probability Density Function (PDF)

Normalized $V_{th}$
Read Reference Voltage ($V_{\text{ref}}$)
Multi-Level Cell (MLC)

Erased (11)

P1 (10)

P2 (00)

P3 (01)

Normalized $V_{th}$

PDF

ER-P1 $V_{ref}$

P1-P2 $V_{ref}$

P2-P3 $V_{ref}$
Threshold Voltage Reduces Over Time

After some retention loss:

- **P1 (10)**
- **P2 (00)**
- **P3 (01)**
Fixed Read Reference Voltage Becomes Suboptimal

After some retention loss:

PDF

P1
(10)

P1-P2 V_{ref}

P2
(00)

P2-P3 V_{ref}

P3
(01)

Normalized V_{th}

Raw bit errors
Optimal Read Reference Voltage (OPT)

After some retention loss:

PDF

P1 (10)

P1-P2 OPT

P1-P2 $V_{\text{ref}}$

P2 (00)

P2-P3 OPT

P2-P3 $V_{\text{ref}}$

P3 (01)

Minimal raw bit errors

Normalized $V_{\text{th}}$
How Current Flash Cells are Programmed

- Programming 2-bit MLC NAND flash memory in two steps

```
  1 0 0 1
ER (11) Temp (0x) ER (11) P1 (10) P2 (00) P3 (01)
  1 0 0 1
```

**LSB Program**

**MSB Program**
MLC Architecture

LSB-Even Page Sets
LSB-Odd Page Sets
MSB-Even Page Sets
MSB-Odd Page Sets
Planar vs. 3D NAND Flash Memory

**Scaling**
- **Planar NAND Flash Memory**: Reduce flash cell size, Reduce distance b/w cells
- **3D NAND Flash Memory**: Increase # of layers

**Reliability**
- **Planar NAND Flash Memory**: Scaling hurts reliability
- **3D NAND Flash Memory**: Not well studied!
3D NAND Flash Memory Structure

Flash Cell

Layer M
Layer 1
Layer 0

BL N  ...  BL 1  BL 0

SAFARI
Charge Trap Based 3D Flash Cell

- Cross-section of a charge trap transistor
Fig. 43. Organization of flash cells in an \(M\)-layer 3D charge trap NAND flash memory chip, where each block consists of \(M\) wordlines and \(N\) bitlines.
More Background and State-of-the-Art

Error Characterization, Mitigation, and Recovery in Flash-Memory-Based Solid-State Drives

This paper reviews the most recent advances in solid-state drive (SSD) error characterization, mitigation, and data recovery techniques to improve both SSD’s reliability and lifetime.

By Yu Cai, Saugata Ghose, Erich F. Haratsch, Yixin Luo, and Onur Mutlu

https://arxiv.org/pdf/1706.08642
More Up-to-date Version


Errors in Flash-Memory-Based Solid-State Drives: Analysis, Mitigation, and Recovery

YU CAI, SAUGATA GHOSE
Carnegie Mellon University

ERICH F. HARATSCH
Seagate Technology

YIXIN LUO
Carnegie Mellon University

ONUR MUTLU
ETH Zürich and Carnegie Mellon University
Flash Memory
Reliability and Security
Error Analysis and Management of NAND Flash Memory
Limits of Charge Memory

- Difficult charge placement and control
  - Flash: floating gate charge
  - DRAM: capacitor charge, transistor leakage

- Reliable sensing becomes difficult as charge storage unit size reduces
Executive Summary

- Problem: MLC NAND flash memory reliability/endurance is a key challenge for satisfying future storage systems’ requirements

- Our Goals: (1) Build reliable error models for NAND flash memory via experimental characterization, (2) Develop efficient techniques to improve reliability and endurance

- This lecture provides a “flash” summary of our recent results published in the past 8 years:
  - Experimental error and threshold voltage characterization [DATE’12&13]
  - Retention-aware error management [ICCD’12]
  - Program interference analysis and read reference V prediction [ICCD’13]
  - Neighbor-assisted error correction [SIGMETRICS’14]
  - Read disturb error handling [DSN’15]
  - Data retention error handling [HPCA’15]
Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
  - Neighbor-Assisted Error Correction
  - Read Disturb Error Handling
  - Retention Error Handling
- 3D NAND Flash Memory Reliability
- Summary
Flash memory is widening its range of applications
- Portable consumer devices, laptop PCs and enterprise servers

Seaung Suk Lee, “Emerging Challenges in NAND Flash Technology”, Flash Summit 2011 (Hynix)
Flash Challenges: Reliability and Endurance

E. Grochowski et al., “Future technology challenges for NAND flash and HDD products”, Flash Memory Summit 2012

- P/E cycles (required)
  - A few thousand
  - Writing the full capacity of the drive 10 times per day for 5 years (STEC)
  - > 50k P/E cycles

- P/E cycles (provided)

NAND Flash Memory Endurance Properties

Program/Erase Cycles

100000
10000
1000
100

SLC MLC TLC

Lithography nm


130 90 64 51 40 32 20 18 16 14

E. Grochowski et al., “Future technology challenges for NAND flash and HDD products”, Flash Memory Summit 2012
Decreasing Endurance with Flash Scaling

- Endurance of flash memory decreasing with scaling and multi-level cells
- Error correction capability required to guarantee storage-class reliability (UBER < 10^{-15}) is increasing exponentially to reach less endurance

UBER: Uncorrectable bit error rate. Fraction of erroneous bits after error correction.

Ariel Maislos, “A New Era in Embedded Flash Memory”, Flash Summit 2011 (Anobit)
NAND Flash Memory is Increasingly Noisy

Write → Noisy NAND → Read
Our Goals:
Build reliable error models for NAND flash memory
Design efficient reliability mechanisms based on the model
NAND Flash Error Model

Experimentally characterize and model dominant errors

Luo et al., “Enabling Accurate and Practical Online Flash Channel Modeling for Modern MLC NAND Flash Memory”, JSAC 2016

Write
- Erase block
- Program page

Noisy NAND
- Neighbor page prog/read (c-to-c interference)

Read
- Retention

Cai et al., “Neighbor-Cell Assisted Error Correction in MLC NAND Flash Memories”, SIGMETRICS 2014
Cai et al., “Read Disturb Errors in MLC NAND Flash Memory: Characterization and Mitigation”, DSN 2015
Cai et al., “Error Analysis and Retention-Aware Error Management for NAND Flash Memory”, ITJ 2013
Cai et al., “Data Retention in MLC NAND Flash Memory: Characterization, Optimization and Recovery”, HPCA 2015

SAFARI
Our Goals and Approach

Goals:
- Understand error mechanisms and develop reliable predictive models for MLC NAND flash memory errors
- Develop efficient error management techniques to mitigate errors and improve flash reliability and endurance

Approach:
- Solid experimental analyses of errors in real MLC NAND flash memory → drive the understanding and models
- Understanding, models, and creativity → drive the new techniques
Many Errors and Their Mitigation [PIEEE’17]

Table 3: List of Different Types of Errors Mitigated by NAND Flash Error Mitigation Mechanisms

<table>
<thead>
<tr>
<th>Mitigation Mechanism</th>
<th>Error Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P/E Cycling [32,33,42] ([IV-A])</td>
</tr>
<tr>
<td></td>
<td>Program [40,42,53] ([IV-B])</td>
</tr>
<tr>
<td></td>
<td>Cell-to-Cell Interference [32,36,55] ([IV-C])</td>
</tr>
<tr>
<td></td>
<td>Data Retention [20,32,34,37,39] ([IV-D])</td>
</tr>
<tr>
<td></td>
<td>Read Disturb [20,32,38,62] ([IV-E])</td>
</tr>
</tbody>
</table>

| Shadow Program Sequencing [35,40] (Section V-A) | X |
| Neighbor-Cell Assisted Error Correction [36] (Section V-B) | X |
| Refresh [34,39,67,68] (Section V-C) | X X |
| Read-Retry [33,72,107] (Section V-D) | X X X |
| Voltage Optimization [37,38,74] (Section V-E) | X X X |
| Hot Data Management [41,63,70] (Section V-F) | X X X X X |
| Adaptive Error Mitigation [43,65,77,78,82] (Section V-G) | X X X X X |

Many Errors and Their Mitigation [PIEEE’17]

Proceedings of the IEEE, Sept. 2017

Error Characterization, Mitigation, and Recovery in Flash-Memory-Based Solid-State Drives

This paper reviews the most recent advances in solid-state drive (SSD) error characterization, mitigation, and data recovery techniques to improve both SSD’s reliability and lifetime.

By Yu Cai, Saugata Ghose, Erich F. Haratsch, Yixin Luo, and Onur Mutlu

https://arxiv.org/pdf/1706.08642

Errors in Flash-Memory-Based Solid-State Drives: Analysis, Mitigation, and Recovery

YU CAI, SAUGATA GHOSE
Carnegie Mellon University

ERICH F. HARATSCH
Seagate Technology

YIXIN LUO
Carnegie Mellon University

ONUR MUTLU
ETH Zürich and Carnegie Mellon University
Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
  - Neighbor-Assisted Error Correction
  - Read Disturb Error Handling
  - Retention Error Handling
  - 3D NAND Flash Memory Reliability
- Summary
Experimental Testing Platform

HAPS-52 Mother Board

Virtex-V FPGA (NAND Controller)

USB Daughter Board

USB Jack

Virtex-II Pro (USB controller)

1x-nm NAND Flash

NAND Daughter Board


NAND Flash Error Types

- Four types of errors [Cai+, DATE 2012]

- Caused by common flash operations
  - Read errors
  - Erase errors
  - Program (interference) errors

- Caused by flash cell losing charge over time
  - Retention errors
    - Whether an error happens depends on required retention time
    - Especially problematic in MLC flash because threshold voltage window to determine stored value is smaller
NAND Flash Usage and Error Model

Program Errors

Erase Errors

Retention Errors

Read Errors

Start

P/E cycle 0

P/E cycle i

P/E cycle n

End of life

Erase Block

Program Page

Retention1
(t1 days)

Retention j
(tj days)

Program Page

Retention Errors

Read Errors

Read Errors

Retention Errors
Methodology: Error and ECC Analysis

- Characterized errors and error rates of 3x and 2y-nm MLC NAND flash using an experimental FPGA-based platform
  - [Cai+, DATE’12, ICCD’12, DATE’13, ITJ’13, ICCD’13, SIGMETRICS’14]

- Quantified Raw Bit Error Rate (RBER) at a given P/E cycle
  - Raw Bit Error Rate: Fraction of erroneous bits without any correction

- Quantified error correction capability (and area and power consumption) of various BCH-code implementations
  - Identified how much RBER each code can tolerate
    - how many P/E cycles (flash lifetime) each code can sustain
Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
  - Neighbor-Assisted Error Correction
  - Read Disturb Error Handling
  - Retention Error Handling
  - 3D NAND Flash Memory Reliability
- Summary
Error Types and Testing Methodology

- **Erase errors**
  - Count the number of cells that fail to be erased to “11” state

- **Program interference errors**
  - Compare the data immediately after page programming and the data after the whole block being programmed

- **Read errors**
  - Continuously read a given block and compare the data between consecutive read sequences

- **Retention errors**
  - Compare the data read after an amount of time to data written
    - Characterize short term retention errors under room temperature
    - Characterize long term retention errors by baking in the oven under 125°C
Retention errors:

- Raw bit error rate increases exponentially with P/E cycles.
- Retention errors are dominant (>99% for 1-year retention time).
- Retention errors increase with retention time requirement.

Cai et al., Error Patterns in MLC NAND Flash Memory, DATE 2012.
Retention Error Mechanism

- Electron loss from the floating gate causes retention errors
  - Cells with more programmed electrons suffer more from retention errors
  - Threshold voltage is more likely to shift by one window than by multiple
Cells with more programmed electrons tend to suffer more from retention noise (i.e. 00 and 01)
More on Flash Error Analysis


Error Patterns in MLC NAND Flash Memory: Measurement, Characterization, and Analysis

Yu Cai¹, Erich F. Haratsch², Onur Mutlu¹ and Ken Mai¹
¹Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA
²LSI Corporation, 1110 American Parkway NE, Allentown, PA
¹{yucai, onur, kenmai}@andrew.cmu.edu, ²erich.haratsch@lsi.com
Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
  - Neighbor-Assisted Error Correction
  - Read Disturb Error Handling
  - Retention Error Handling
  - 3D NAND Flash Memory Reliability
- Summary
Solution to Retention Errors

- Refresh periodically

- Change the period based on P/E cycle wearout
  - Refresh more often at higher P/E cycles

- Use a combination of in-place and remapping-based refresh

Flash Correct-and-Refresh (FCR)

- Key Observations:
  - Retention errors are the dominant source of errors in flash memory \[\text{Cai+ DATE 2012}\][\text{Tanakamaru+ ISSCC 2011}] → limit flash lifetime as they increase over time
  - Retention errors can be corrected by “refreshing” each flash page periodically

- Key Idea:
  - Periodically read each flash page,
  - Correct its errors using “weak” ECC, and
  - Either remap it to a new physical page or reprogram it in-place,
  - Before the page accumulates more errors than ECC-correctable
  - Optimization: Adapt refresh rate to endured P/E cycles

Cai et al., Flash Correct and Refresh, ICCD 2012.
FCR: Two Key Questions

- **How to refresh?**
  - **Remap** a page to another one
  - **Reprogram** a page (in-place)
  - **Hybrid** of remap and reprogram

- **When to refresh?**
  - **Fixed period**
  - **Adapt the period** to retention error severity
In-Place Reprogramming of Flash Cells

- Pro: No remapping needed → no additional erase operations
- Con: Increases the occurrence of program errors

Retention errors are caused by cell voltage shifting to the left.

ISPP moves cell voltage to the right; fixes retention errors.
Adaptive-rate FCR provides the highest lifetime
Lifetime of FCR much higher than lifetime of stronger ECC
Energy Overhead

- Adaptive-rate refresh: <1.8% energy increase until daily refresh is triggered
Flash Correct-and-Refresh [ICCD’12]

- Yu Cai, Gulay Yalcin, Onur Mutlu, Erich F. Haratsch, Adrian Cristal, Osman Unsal, and Ken Mai,

"Flash Correct-and-Refresh: Retention-Aware Error Management for Increased Flash Memory Lifetime"

Proceedings of the 30th IEEE International Conference on Computer Design (ICCD), Montreal, Quebec, Canada, September 2012. Slides (ppt)(pdf)

Flash Correct-and-Refresh: Retention-Aware Error Management for Increased Flash Memory Lifetime

Yu Cai\textsuperscript{1}, Gulay Yalcin\textsuperscript{2}, Onur Mutlu\textsuperscript{1}, Erich F. Haratsch\textsuperscript{3}, Adrian Cristal\textsuperscript{2}, Osman S. Unsal\textsuperscript{2} and Ken Mai\textsuperscript{1}

\textsuperscript{1}DSSC, Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA
\textsuperscript{2}Barcelona Supercomputing Center, C/Jordi Girona 29, Barcelona, Spain
\textsuperscript{3}LSI Corporation, 1110 American Parkway NE, Allentown, PA
More Detail on Flash Error Analysis

Yu Cai, Gulay Yalcin, Onur Mutlu, Erich F. Haratsch, Adrian Cristal, Osman Unsal, and Ken Mai,
"Error Analysis and Retention-Aware Error Management for NAND Flash Memory"
Table 3 List of Different Types of Errors Mitigated by NAND Flash Error Mitigation Mechanisms

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Shadow Program Sequencing [35,40] (Section V-A)</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neighbor-Cell Assisted Error Correction [36] (Section V-B)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Refresh [34,39,67,68] (Section V-C)</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Read-Retry [33,72,107] (Section V-D)</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Voltage Optimization [37,38,74] (Section V-E)</td>
<td></td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Hot Data Management [41,63,70] (Section V-F)</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Adaptive Error Mitigation [43,65,77,78,82] (Section V-G)</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Error Characterization, Mitigation, and Recovery in Flash-Memory-Based Solid-State Drives

This paper reviews the most recent advances in solid-state drive (SSD) error characterization, mitigation, and data recovery techniques to improve both SSD’s reliability and lifetime.

By Yu Cai, Saugata Ghose, Erich F. Haratsch, Yixin Luo, and Onur Mutlu

https://arxiv.org/pdf/1706.08642
More Up-to-date Version

Yu Cai, Saugata Ghose, Erich F. Haratsch, Yixin Luo, and Onur Mutlu, "Errors in Flash-Memory-Based Solid-State Drives: Analysis, Mitigation, and Recovery"

Errors in Flash-Memory-Based Solid-State Drives: Analysis, Mitigation, and Recovery

YU CAI, SAUGATA GHOSE
Carnegie Mellon University

ERICH F. HARATSCH
Seagate Technology

YIXIN LUO
Carnegie Mellon University

ONUR MUTLU
ETH Zürich and Carnegie Mellon University
Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
  - Neighbor-Assisted Error Correction
  - Read Disturb Error Handling
  - Retention Error Handling
  - 3D NAND Flash Memory Reliability
- Summary
Key Questions

- How does threshold voltage (Vth) distribution of different programmed states change over flash lifetime?

- Can we model it accurately and predict the Vth changes?

- Can we build mechanisms that can correct for Vth changes? (thereby reducing read error rates)
Threshold Voltage Distribution Model

Gaussian distribution with additive white noise
As P/E cycles increase ...
- Distribution shifts to the right
- Distribution becomes wider

Characterized on 2Y-nm chips using the read-retry feature

Cai et al., Threshold Voltage Distribution in MLC NAND Flash Memory, DATE 2013.
Threshold Voltage Distribution Model

- **Vth distribution** can be modeled with ~95% accuracy as a Gaussian distribution with additive white noise.

- **Distortion in Vth** over P/E cycles can be modeled and predicted as an exponential function of P/E cycles:
  - With more than 95% accuracy.
More Detail on Threshold Voltage Model


Threshold Voltage Distribution in MLC NAND Flash Memory: Characterization, Analysis, and Modeling

Yu Cai\textsuperscript{1}, Erich F. Haratsch\textsuperscript{2}, Onur Mutlu\textsuperscript{1} and Ken Mai\textsuperscript{1}
\textsuperscript{1}DSSC, Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA
\textsuperscript{2}LSI Corporation, 1110 American Parkway NE, Allentown, PA
\textsuperscript{1}\{yucai, onur, kenmai\}@andrew.cmu.edu, \textsuperscript{2}erich.haratsch@lsi.com
More Accurate and Online Channel Modeling


Enabling Accurate and Practical Online Flash Channel Modeling for Modern MLC NAND Flash Memory

Yixin Luo, Saugata Ghose, Yu Cai, Erich F. Haratsch, Onur Mutlu
Non-Gaussian Vth Distributions (1X-nm)

Fig. 4: Gaussian-based model (solid/dashed lines) vs. data measured from real NAND flash chips (markers) under different P/E cycle counts.
Better Modeling of Vth Distributions (I)

Fig. 6: Our new Student’s $t$-based model (solid/dashed lines) vs. data measured from real NAND flash chips (markers) under different P/E cycle counts.
## Better Modeling of Vth Distributions (II)

<table>
<thead>
<tr>
<th>P/E Cycles</th>
<th>0</th>
<th>2.5K</th>
<th>5K</th>
<th>7.5K</th>
<th>10K</th>
<th>12K</th>
<th>14K</th>
<th>16K</th>
<th>18K</th>
<th>20K</th>
<th>AVG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian</td>
<td>.99%</td>
<td>1.8%</td>
<td>1.6%</td>
<td>1.8%</td>
<td>1.9%</td>
<td>2.4%</td>
<td>3.1%</td>
<td>8.7%</td>
<td>2.1%</td>
<td>2.3%</td>
<td><strong>2.6%</strong></td>
</tr>
<tr>
<td>Normal-Laplace</td>
<td>.34%</td>
<td>.46%</td>
<td>.55%</td>
<td>.61%</td>
<td>.63%</td>
<td>.67%</td>
<td>.68%</td>
<td>.70%</td>
<td>.67%</td>
<td>.67%</td>
<td><strong>.61%</strong></td>
</tr>
<tr>
<td>Student’s t</td>
<td>.37%</td>
<td>.51%</td>
<td>.61%</td>
<td>.68%</td>
<td>.70%</td>
<td>.76%</td>
<td>.76%</td>
<td>.78%</td>
<td>.76%</td>
<td>.78%</td>
<td><strong>.68%</strong></td>
</tr>
</tbody>
</table>

**TABLE 1:** Modeling error of the evaluated threshold voltage distribution models, at various P/E cycle counts.
Fig. 13: Threshold voltage distribution as predicted by our dynamic model for 20K P/E cycles, using characterization data from 2.5K, 5K, 7.5K, and 10K P/E cycles, shown as solid/dashed lines. Markers represent data measured from real NAND flash chips at 20K P/E cycles.
Program Interference Errors

- When a cell is being programmed, **voltage level of a neighboring cell changes** (unintentionally) due to parasitic capacitance coupling
  - can change the data value stored

- Also called program interference error

- Causes neighboring cell voltage to increase (shift right)

- Once retention errors are minimized, these errors can become dominant
How Current Flash Cells are Programmed

- Programming 2-bit MLC NAND flash memory in two steps

![Diagram showing the programming process of 2-bit MLC NAND flash memory](image.png)
Basics of Program Interference

(n-1,j-1)  (n-1,j)  (n-1,j+1)  (n+1,j-1)  (n+1,j)  (n+1,j+1)

WL<0>  WL<1>  WL<2>
Traditional model for victim cell threshold voltage change

$$\Delta V_{\text{victim}} = \frac{(2C_x \Delta V_x + C_y \Delta V_y + 2C_{xy} \Delta V_{xy})}{C_{\text{total}}}$$

Not accurate and requires knowledge of coupling caps!
Our Goal and Idea

- Develop a new, more accurate and easier to implement model for program interference.

Idea:
- Empirically characterize and model the effect of neighbor cell Vth changes on the Vth of the victim cell.
- Fit neighbor Vth change to a linear regression model and find the coefficients of the model via empirical measurement.

\[
\Delta V_{\text{victim}}(n, j) = \sum_{y=j-K}^{j+K} \sum_{x=n+1}^{n+M} \alpha(x, y) \Delta V_{\text{neighbor}}(x, y) + \alpha_0 V_{\text{before \, victim}}(n, j)
\]

Can be measured
Feature extraction for $V_{th}$ changes based on characterization

- Threshold voltage changes on aggressor cell
- Original state of victim cell

Enhanced linear regression model

$$\Delta V_{\text{victim}}(n, j) = \sum_{j+K}^{n+M} \sum_{y=j-K}^{x=n+1} \alpha(x, y) \Delta V_{\text{neighbor}}(x, y) + \alpha_0 V_{\text{before}}^{\text{victim}}(n, j)$$

$$Y = X\alpha + \varepsilon \quad \text{(vector expression)}$$

Maximum likelihood estimation of the model coefficients

$$\arg\min_{\alpha} \left( \|X \times \alpha - Y\|^2_2 + \lambda \|\alpha\|_1 \right)$$
Effect of Neighbor Voltages on the Victim

- Immediately-above cell interference is dominant
- Immediately-diagonal neighbor is the second dominant
- Far neighbor cell interference exists
- Victim cell’s Vth has negative effect on interference
New Model for Program Interference

\[ \Delta V_{\text{victim}}(n, j) = \sum_{y=j-K}^{j+M} \sum_{x=n+1}^{n+M} \alpha(x, y) \Delta V_{\text{neighbor}}(x, y) + \alpha_0 V_{\text{before}}^{\text{victim}}(n, j) \]

Cai et al., Program Interference in MLC NAND Flash Memory, ICCD 2013
Model Accuracy

Characterized on 2Y-nm chips using the read-retry feature

Interference causes systematic Vth shift

Ideal if prediction is 100% accurate

Model corrects for the Vth shift: 96.8% acc.
Many Other Results in the Paper


Program Interference in MLC NAND Flash Memory: Characterization, Modeling, and Mitigation

Yu Cai¹, Onur Mutlu¹, Erich F. Haratsch² and Ken Mai¹

1. Data Storage Systems Center, Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA
2. LSI Corporation, San Jose, CA
yucaicai@gmail.com, {omutlu, kenmai}@andrew.cmu.edu
Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- **Error Analysis and Management**
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - **Read Reference Voltage Prediction**
  - Neighbor-Assisted Error Correction
  - Read Disturb Error Handling
  - Retention Error Handling
  - 3D NAND Flash Memory Reliability
- Summary
Mitigation: Applying the Model

- So, what can we do with the model?
- Goal: Mitigate the effects of program interference caused voltage shifts
Read reference voltage affects the raw bit error rate

There exists an optimal read reference voltage

- Predictable if the statistics (i.e. mean, variance) of threshold voltage distributions are characterized and modeled

\[ \text{BER1} = \int_{v_{\text{ref}}}^{+\infty} f(x) \, dx + \int_{-\infty}^{v_{\text{ref}}} g(x) \, dx \]

\[ \text{BER2} = \int_{v'_{\text{ref}}}^{+\infty} f(x) \, dx + \int_{-\infty}^{v'_{\text{ref}}} g(x) \, dx \]
Vth shift learning (done every ~1k P/E cycles)
- Program sample cells with known data pattern and test Vth
- Program aggressor neighbor cells and test victim Vth after interference
- Characterize the mean shift in Vth (i.e., program interference noise)

Optimum read reference voltage prediction
- Default read reference voltage + Predicted mean Vth shift by model
Effect of Read Reference Voltage Prediction

- Read reference voltage prediction reduces raw BER (by 64%) and increases the P/E cycle lifetime (by 30%).

32k-bit BCH Code
(acceptable BER = 2x10⁻³)

30% lifetime improvement

- No read reference voltage prediction
- With read reference voltage prediction

Raw bit error rate

Number of P/E Cycles

- Read reference voltage prediction reduces raw BER (by 64%) and increases the P/E cycle lifetime (by 30%).
More on Read Reference Voltage Prediction

Yu Cai, Onur Mutlu, Erich F. Haratsch, and Ken Mai,
"Program Interference in MLC NAND Flash Memory: Characterization, Modeling, and Mitigation"
Slides (pptx) (pdf) Lightning Session Slides (pdf)

Program Interference in MLC NAND Flash Memory: Characterization, Modeling, and Mitigation

Yu Cai¹, Onur Mutlu¹, Erich F. Haratsch² and Ken Mai¹
1. Data Storage Systems Center, Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA
2. LSI Corporation, San Jose, CA
yuicaicai@gmail.com, {omutlu, kenmai}@andrew.cmu.edu
More Accurate and Online Channel Modeling

Readings on Flash Memory
Error Characterization, Mitigation, and Recovery in Flash-Memory-Based Solid-State Drives

This paper reviews the most recent advances in solid-state drive (SSD) error characterization, mitigation, and data recovery techniques to improve both SSD's reliability and lifetime.

By Yu Cai, Saugata Ghose, Erich F. Haratsch, Yixin Luo, and Onur Mutlu

https://arxiv.org/pdf/1706.08642
More Up-to-date Version

Yu Cai, Saugata Ghose, Erich F. Haratsch, Yixin Luo, and Onur Mutlu, "Errors in Flash-Memory-Based Solid-State Drives: Analysis, Mitigation, and Recovery"
[Preliminary arxiv.org version]

Errors in Flash-Memory-Based Solid-State Drives: Analysis, Mitigation, and Recovery

YU CAI, SAUGATA GHOSE
Carnegie Mellon University

ERICH F. HARATSCH
Seagate Technology

YIXIN LUO
Carnegie Mellon University

ONUR MUTLU
ETH Zürich and Carnegie Mellon University
Flash Memory Reliability
Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- **Error Analysis and Management**
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - **Read Reference Voltage Prediction**
  - Neighbor-Assisted Error Correction
  - Read Disturb Error Handling
  - Retention Error Handling
  - 3D NAND Flash Memory Reliability
- Summary
So, what can we do with the model?

Goal: Mitigate the effects of program interference caused voltage shifts
Optimum Read Reference for Flash Memory

- Read reference voltage affects the raw bit error rate

\[
\text{State-A} \quad f(x) \quad g(x) \quad \text{State-B}
\]

\[
\text{State-A} \quad f(x) \quad g(x) \quad \text{State-B}
\]

\[
\begin{align*}
\text{BER1} &= \int_{v_{\text{ref}}}^{+\infty} f(x) \, dx + \int_{-\infty}^{v_{\text{ref}}} g(x) \, dx \\
\text{BER2} &= \int_{v'_{\text{ref}}}^{+\infty} f(x) \, dx + \int_{-\infty}^{v'_{\text{ref}}} g(x) \, dx
\end{align*}
\]

- There exists an optimal read reference voltage
  - Predictable if the statistics (i.e. mean, variance) of threshold voltage distributions are characterized and modeled
Vth shift learning (done every ~1k P/E cycles)
- Program sample cells with known data pattern and test Vth
- Program aggressor neighbor cells and test victim Vth after interference
- Characterize the mean shift in Vth (i.e., program interference noise)

Optimum read reference voltage prediction
- Default read reference voltage + Predicted mean Vth shift by model
Effect of Read Reference Voltage Prediction

Read reference voltage prediction reduces raw BER (by 64%) and increases the P/E cycle lifetime (by 30%).

32k-bit BCH Code (acceptable BER = $2 \times 10^{-3}$)

30% lifetime improvement

Read reference voltage prediction reduces raw BER (by 64%) and increases the P/E cycle lifetime (by 30%).
More on Read Reference Voltage Prediction

- Yu Cai, Onur Mutlu, Erich F. Haratsch, and Ken Mai,
"Program Interference in MLC NAND Flash Memory: Characterization, Modeling, and Mitigation"
Slides (pptx) (pdf) Lightning Session Slides (pdf)

Program Interference in MLC NAND Flash Memory: Characterization, Modeling, and Mitigation

Yu Cai\textsuperscript{1}, Onur Mutlu\textsuperscript{1}, Erich F. Haratsch\textsuperscript{2} and Ken Mai\textsuperscript{1}
1. Data Storage Systems Center, Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA
2. LSI Corporation, San Jose, CA
yucaicai@gmail.com, \{omutlu, kenmai\}@andrew.cmu.edu
Non-Gaussian Vth Distributions (1X-nm)

Fig. 4: Gaussian-based model (solid/dashed lines) vs. data measured from real NAND flash chips (markers) under different P/E cycle counts.
Better Modeling of Vth Distributions (I)

Fig. 6: Our new Student’s t-based model (solid/dashed lines) vs. data measured from real NAND flash chips (markers) under different P/E cycle counts.

### TABLE 1: Modeling error of the evaluated threshold voltage distribution models, at various P/E cycle counts.

<table>
<thead>
<tr>
<th>P/E Cycles</th>
<th>0</th>
<th>2.5K</th>
<th>5K</th>
<th>7.5K</th>
<th>10K</th>
<th>12K</th>
<th>14K</th>
<th>16K</th>
<th>18K</th>
<th>20K</th>
<th>AVG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian</td>
<td>.99%</td>
<td>1.8%</td>
<td>1.6%</td>
<td>1.8%</td>
<td>1.9%</td>
<td>2.4%</td>
<td>3.1%</td>
<td>8.7%</td>
<td>2.1%</td>
<td>2.3%</td>
<td><strong>2.6%</strong></td>
</tr>
<tr>
<td>Normal-Laplace</td>
<td>.34%</td>
<td>.46%</td>
<td>.55%</td>
<td>.61%</td>
<td>.63%</td>
<td>.67%</td>
<td>.68%</td>
<td>.70%</td>
<td>.67%</td>
<td>.67%</td>
<td><strong>.61%</strong></td>
</tr>
<tr>
<td>Student’s t</td>
<td>.37%</td>
<td>.51%</td>
<td>.61%</td>
<td>.68%</td>
<td>.70%</td>
<td>.76%</td>
<td>.76%</td>
<td>.78%</td>
<td>.76%</td>
<td>.78%</td>
<td><strong>.68%</strong></td>
</tr>
</tbody>
</table>

![Characterization Latency vs Online Computation Latency](image)

**Fig. 8:** Overall latency breakdown of the three evaluated threshold voltage distribution models for static modeling.
Vth Prediction vs. Reality with Better Modeling

Fig. 13: Threshold voltage distribution as predicted by our dynamic model for 20K P/E cycles, using characterization data from 2.5K, 5K, 7.5K, and 10K P/E cycles, shown as solid/dashed lines. Markers represent data measured from real NAND flash chips at 20K P/E cycles.
Online Read Reference Voltage Prediction

Fig. 16: Actual and modeled optimal read reference voltages \( (V_{opt}) \) using the three evaluated threshold voltage distribution models at different P/E cycle counts.
Effect on RBER of Read Ref V Prediction

Fig. 17: RBER achieved by actual and modeled\textit{ optimal} read reference voltages ($V_{opt}$) using the three evaluated threshold voltage distribution models at different P/E cycle counts.

Enabling Accurate and Practical Online Flash Channel Modeling for Modern MLC NAND Flash Memory

Yixin Luo, Saugata Ghose, Yu Cai, Erich F. Haratsch, Onur Mutlu
Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
  - Neighbor-Assisted Error Correction
  - Read Disturb Error Handling
  - Retention Error Handling
  - 3D NAND Flash Memory Reliability
- Summary
Goal

- Develop a better error correction mechanism for cases where ECC fails to correct a page
Observations So Far

- Immediate neighbor cell has the most effect on the victim cell when programmed.

- A single set of read reference voltages is used to determine the value of the (victim) cell.

- The set of read reference voltages is determined based on the overall threshold voltage distribution of all cells in flash memory.
New Observations [Cai+ SIGMETRICS’14]

- Vth distributions of **cells with different-valued immediate-neighbor cells** are significantly different
  - Because neighbor value affects the amount of Vth shift

- **Corollary:** If we know the value of the immediate-neighbor, we can find a **more accurate set of read reference voltages** based on the “conditional” threshold voltage distribution

Cai et al., **Neighbor-Cell Assisted Error Correction for MLC NAND Flash Memories**, SIGMETRICS 2014.
Victim WL before MSB page of aggressor WL are programmed

Victim WL after MSB page of aggressor WL are programmed
Then, we could choose a different read reference voltage to more accurately read the “victim” cell.
Overall vs Conditional Reading

- Using the optimum read reference voltage based on the overall distribution leads to more errors.

- Better to use the optimum read reference voltage based on the conditional distribution (i.e., value of the neighbor).
  - Conditional distributions of two states are farther apart from each other.
Real NAND Flash Chip Measurement Results

Raw BER of conditional reading is much smaller than overall reading
Idea: Neighbor Assisted Correction (NAC)

- Read a page with the read reference voltages based on overall Vth distribution (same as today) and buffer it

- If ECC fails:
  - Read the immediate-neighbor page
  - Re-read the page using the read reference voltages corresponding to the voltage distribution assuming a particular immediate-neighbor value
  - Replace the buffered values of the cells with that particular immediate-neighbor cell value
  - Apply ECC again

118
Neighbor Assisted Correction Flow

- Trigger neighbor-assisted reading only when ECC fails
- Read neighbor values and use corresponding read reference voltages in a prioritized order until ECC passes

How to select next local optimum read reference voltage?
Lifetime Extension with NAC

Stage-0 Stage-1 Stage-2 Stage-3

ECC capable of correcting 40 bits per 1k-Byte

33% lifetime improvement at no performance loss
Performance Analysis of NAC

No performance loss within nominal lifetime and with reasonable (1%) ECC fail rates
More on Neighbor-Assisted Correction

- Yu Cai, Gulay Yalcin, Onur Mutlu, Eric Haratsch, Osman Unsal, Adrian Cristal, and Ken Mai,

"Neighbor-Cell Assisted Error Correction for MLC NAND Flash Memories"


Neighbor-Cell Assisted Error Correction for MLC NAND Flash Memories

Yu Cai¹, Gulay Yalcin², Onur Mutlu¹, Erich F. Haratsch⁴, Osman Unsal², Adrian Cristal²,³, and Ken Mai¹

¹Electrical and Computer Engineering Department, Carnegie Mellon University
²Barcelona Supercomputing Center, Spain
³IIIa – CSIC – Spain National Research Council
⁴LSI Corporation

yucaicai@gmail.com, {omutlu, kenmai}@ece.cmu.edu, {gulay.yalcin, adrian.cristal, osman.unsal}@bsc.es
Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- **Error Analysis and Management**
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
  - Neighbor-Assisted Error Correction
  - **Read Disturb Error Handling**
  - Retention Error Handling
  - 3D NAND Flash Memory Reliability
- Summary
Read Disturb Errors in Flash Memory
One Issue: Read Disturb in Flash Memory

- All scaled memories are prone to read disturb errors
- DRAM
- SRAM
- Hard Disks: Adjacent Track Interference
- NAND Flash
NAND Flash Memory Background

Flash Memory

Block 0
- Read
- Pass
- Pass
- ...
- Pass

......

Block N

Flash Controller

SAFARI
Flash Cell

Floating Gate Transistor
(Flash Cell)

$V_{th} = 2.5 \text{ V}$
Flash Read

\[ V_{\text{read}} = 2.5 \text{ V} \]

\[ V_{\text{th}} = 2 \text{ V} \]

\[ V_{\text{read}} = 2.5 \text{ V} \]

\[ V_{\text{th}} = 3 \text{ V} \]

Gate
Flash Pass-Through

\[ V_{\text{pass}} = 5 \, \text{V} \]

\[ V_{\text{th}} = 2 \, \text{V} \]

\[ V_{\text{pass}} = 5 \, \text{V} \]

\[ V_{\text{th}} = 3 \, \text{V} \]
Read from Flash Cell Array

$V_{\text{pass}} = 5.0$ V

Page 1: Pass (5V)

Page 2: Read (2.5V)

Page 3: Pass (5V)

Page 4: Pass (5V)

Correct values for page 2:

| 0 | 0 | 1 | 1 |

V$\text{read} = 2.5$ V

V$\text{pass} = 5.0$ V
Read Disturb Problem: “Weak Programming” Effect

Repeatedly read page 3 (or any page other than page 2)
Read Disturb Problem: “Weak Programming” Effect

High pass-through voltage induces "weak-programming" effect

Incorrect values from page 2:
Executive Summary [DSN’15]

• **Read disturb errors** limit flash memory lifetime today
  – Apply a *high pass-through voltage* ($V_{\text{pass}}$) to multiple pages on a read
  – Repeated application of $V_{\text{pass}}$ can alter stored values in unread pages

• We **characterize read disturb** on real NAND flash chips
  – Slightly lowering $V_{\text{pass}}$ greatly reduces read disturb errors
  – Some flash cells are more prone to read disturb

• **Technique 1:** *Mitigate* read disturb errors online
  – $V_{\text{pass}}$ *Tuning* dynamically finds and applies a lowered $V_{\text{pass}}$ per block
  – Flash memory lifetime improves by 21%

• **Technique 2:** *Recover* after failure to prevent data loss
  – *Read Disturb Oriented Error Recovery* (RDR) selectively corrects cells more susceptible to read disturb errors
  – Reduces raw bit error rate (RBER) by up to 36%
Key Observation 1: Slightly lowering $V_{\text{pass}}$ greatly reduces read disturb errors

Fig. 11. Raw bit error rate vs. read disturb count for different $V_{\text{pass}}$ values, for flash memory under 8K P/E cycles of wear.
Outline

• Background (Problem and Goal)
• Key Experimental Observations
• Mitigation: $V_{\text{pass}}$ Tuning
• Recovery: Read Disturb Oriented Error Recovery
• Conclusion
Read Disturb Mitigation: $V_{\text{pass}}$ Tuning

• Key Idea: Dynamically find and apply a lowered $V_{\text{pass}}$

• Trade-off for lowering $V_{\text{pass}}$
  + Allows more read disturbs
  – Induces more read errors
Read Errors Induced by $V_{\text{pass}}$ Reduction

Reducing $V_{\text{pass}}$ to 4.9V

$V_{\text{pass}} = 4.9$ V

<table>
<thead>
<tr>
<th>Page 1</th>
<th>Page 2</th>
<th>Page 3</th>
<th>Page 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.0V</td>
<td>3.5V</td>
<td>2.2V</td>
<td>3.5V</td>
</tr>
<tr>
<td>3.8V</td>
<td>2.9V</td>
<td>4.3V</td>
<td>2.3V</td>
</tr>
<tr>
<td>3.9V</td>
<td>2.4V</td>
<td>4.6V</td>
<td>1.9V</td>
</tr>
<tr>
<td>4.8V</td>
<td>2.1V</td>
<td>1.8V</td>
<td>4.3V</td>
</tr>
</tbody>
</table>

Reducing $V_{\text{pass}}$ to 4.9V

$V_{\text{read}} = 2.5$ V

<table>
<thead>
<tr>
<th>Page 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.5V</td>
</tr>
<tr>
<td>2.9V</td>
</tr>
<tr>
<td>2.4V</td>
</tr>
<tr>
<td>2.1V</td>
</tr>
</tbody>
</table>

Reducing $V_{\text{pass}}$ to 4.9V

<table>
<thead>
<tr>
<th>Page 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.2V</td>
</tr>
<tr>
<td>4.3V</td>
</tr>
<tr>
<td>4.6V</td>
</tr>
<tr>
<td>1.8V</td>
</tr>
</tbody>
</table>

Reducing $V_{\text{pass}}$ to 4.9V

<table>
<thead>
<tr>
<th>Page 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.5V</td>
</tr>
<tr>
<td>2.3V</td>
</tr>
<tr>
<td>1.9V</td>
</tr>
<tr>
<td>4.3V</td>
</tr>
</tbody>
</table>

SAFARI
Read Errors Induced by $V_{\text{pass}}$ Reduction

Reducing $V_{\text{pass}}$ to 4.7V

$V_{\text{pass}} = 4.7 \text{ V}$

$V_{\text{read}} = 2.5 \text{ V}$

Incorrect values from page 2:

0 0 1 0
Utilizing the Unused ECC Capability

1. ECC provisioned for high retention “age”
2. Unused ECC capability can be used to fix read errors
3. Unused ECC capability decreases over retention age
   Dynamically adjust $V_{\text{pass}}$ so that read errors fully utilize the unused ECC capability
V_{pass} Reduction Trade-Off Summary

• Today: Conservatively set V_{pass} to a high voltage
  – Accumulates more read disturb errors at the end of each refresh interval
  + No read errors

• Idea: **Dynamically adjust V_{pass} to unused ECC capability**
  + Minimize read disturb errors
    o Control read errors to be tolerable by ECC
    o If read errors exceed ECC capability, read again with a higher V_{pass} to correct read errors
$V_{\text{pass}}$ Tuning Steps

• Perform once for each block every day:

1. Estimate *unused ECC capability (using retention age)*
2. Aggressively reduce $V_{\text{pass}}$ until *read errors exceeds ECC capability*
3. Gradually increase $V_{\text{pass}}$ until read error becomes just less than ECC capability
Evaluation of $V_{\text{pass}}$ Tuning

- 19 real workload I/O traces
- Assume 7-day refresh period
- Similar methodology as before to determine acceptable $V_{\text{pass}}$ reduction

- **Overhead** for a 512 GB flash drive:
  - 128 KB storage overhead for per-block $V_{\text{pass}}$ setting and worst-case page
  - 24.34 sec/day average $V_{\text{pass}}$ Tuning overhead
Average lifetime improvement: 21.0%
Read Disturb Prone vs. Resistant Cells

PDF

Disturb-Resistant

Disturb-Prone

Normalized $V_{th}$

N read disturbs

N read disturbs
Observation 2: Some Flash Cells Are More Prone to Read Disturb

After 250K read disturbs:

- Disturb-prone cells have higher threshold voltages
- Disturb-resistant cells have lower threshold voltages

Disturb-prone cells transition to an ER state, while disturb-resistant cells transition to a P1 state.
Read Disturb Oriented Error Recovery (RDR)

• Triggered by an uncorrectable flash error
  – Back up all valid data in the faulty block
  – Disturb the faulty page 100K times (more)
  – Compare \( V_{th} \)'s before and after read disturb
  – Select cells susceptible to flash errors \( (V_{ref} - \sigma < V_{th} < V_{ref} - \sigma) \)
  – Predict among these susceptible cells
    • Cells with more \( V_{th} \) shifts are disturb-prone \( \rightarrow \) Lower \( V_{th} \) state
    • Cells with less \( V_{th} \) shifts are disturb-resistant \( \rightarrow \) Higher \( V_{th} \) state

Reduces total error count by up to 36% @ 1M read disturbs
ECC can be used to correct the remaining errors
RDR Evaluation

Reduces total error counts by up to 36% @ 1M read disturbs
ECC can be used to correct the remaining errors
More on Flash Read Disturb Errors [DSN’15]

- Yu Cai, Yixin Luo, Saugata Ghose, Erich F. Haratsch, Ken Mai, and Onur Mutlu,

"Read Disturb Errors in MLC NAND Flash Memory: Characterization and Mitigation"

Proceedings of the 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Rio de Janeiro, Brazil, June 2015.

Read Disturb Errors in MLC NAND Flash Memory: Characterization, Mitigation, and Recovery

Yu Cai, Yixin Luo, Saugata Ghose, Erich F. Haratsch*, Ken Mai, Onur Mutlu
Carnegie Mellon University, *Seagate Technology
yucaicai@gmail.com, {yixinluo, ghose, kenmai, onur}@cmu.edu
Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
  - Neighbor-Assisted Error Correction
  - Read Disturb Error Handling
  - Retention Error Handling
  - 3D NAND Flash Memory Reliability
- Summary
Data Retention in Flash Memory
Characterize: retention loss in real NAND chip

Optimize: read performance for old data

Recover: old data after failure
An unfortunate tale about Samsung's SSD 840 read performance degradation

An avalanche of reports emerged last September, when owners of the usually speedy Samsung SSD 840 and SSD 840 EVO detected the drives were no longer performing as they used to.

The issue has to do with older blocks of data: reading old files consistently slower than normal as slow as 30MB/s whereas newly-written files ones used in benchmarks, perform as fast as new – around 500 MB/s for the well regarded SSD 840 EVO. The reason no one had noticed (we reviewed the drive back in September 2013) is that data has to be several weeks old to show the problem. Samsung promptly admitted the issue and proposed a fix.

Why is old data slower?

Retention loss!
Retention loss

Charge leakage over time

One dominant source of flash memory errors \[\text{DATE '12, ICCD '12}\]

Side effect: Longer read latency
Multi-Level Cell (MLC) threshold voltage distribution

Erased (11)  
P1 (10)  
P2 (00)  
P3 (01)  

Normalized $V_{th}$
Experimental Testing Platform


Cai et al., FPGA-based Solid-State Drive prototyping platform, FCCM 2011.
Finding: Cell’s threshold voltage decreases over time.
Threshold voltage reduces over time

Old data

Less charge

PDF

More charge

Normalized $V_{th}$

P1 (10)

P2 (00)

P3 (01)

New data

Old data

Less charge

PDF

More charge

Normalized $V_{th}$

P1 (10)

P2 (00)

P3 (01)
First read attempt fails

Old data

PDF

Less charge

More charge

Raw bit errors > ECC correctable errors
Read-retry

Old data

PDF

Increase read latency

Fewer raw bit errors

PDF

Normalized $V_{th}$

Old data

$V_b'$ $V_b$

$V_c'$ $V_c$

P1 (10)

P2 (00)

P3 (01)

Fewer raw bit errors
Why is old data slower?

Retention loss

- Leak charge over time
  - Generate retention errors
    - Require read-retry
      - Longer read latency
Characterize retention loss in real NAND chip

Optimize read performance for old data

Recover old data after failure
The ideal read voltage

OPT: Optimal read reference voltage  ➔ minimal read latency

Minimal raw bit errors
In reality

• **OPT changes over time due to retention loss**

• *Luckily, OPT change is:*
  - Gradual
  - Uni-directional (decreases over time)
Retention Optimized Reading (ROR)

Components:

1. Online pre-optimization algorithm
   - Learns and records OPT
   - Performs in the background once every day

2. Simpler read-retry technique
   - If recorded OPT is out-of-date, read-retry with lower voltage
1. Online Pre-Optimization Algorithm

- Triggered periodically (e.g., per day)
- Find and record an OPT as per-block $V_{pred}$
- Performed in background
- Small storage overhead

![Diagram showing PDF and normalized $V_{th}$, with new and old $V_{pred}$ values indicated.](image)
2. Improved Read-Retry Technique

• Performed as normal read
• $V_{\text{pred}}$ already close to actual OPT
• Decrease $V_{\text{ref}}$ if $V_{\text{pred}}$ fails, and retry
ROR result

- **Read-retry count**: Baseline 90%, ROR 60% (30% improvement)
- **BCH decoding latency**: Baseline 100%, ROR 90% (10% improvement)
- **Total read latency**: Baseline 100%, ROR 80% (20% improvement)
Retention optimized reading

Retention loss $\rightarrow$ longer read latency

Optimal read reference voltage (OPT)
$\rightarrow$ Shortest read latency
$\rightarrow$ Decreases gradually over time (retention)
$\rightarrow$ Learn OPT periodically
$\rightarrow$ Minimize read-retry & RBER
$\rightarrow$ Shorter read latency
**Characterize** retention loss in real NAND chip

**Optimize** read performance for old data

**Recover** old data after failure
Retention failure

Very old data

PDF

P1 (10) OPTb

P2 (00) OPTc

P3 (01)

Uncorrectable errors

Normalized $V_{th}$
Leakage speed variation

PDF

N-day retention

low-leaking cell

N-day retention

fast-leaking cell

Normalized $V_{th}$
A simplified example
Reading very old data

Very old data

Fast-leaking cells have lower $V_{th}$

Slow-leaking cells have higher $V_{th}$

PDF

Normalized $V_{th}$

175

Fast-leaking cells have lower $V_{th}$

Slow-leaking cells have higher $V_{th}$

SAFARI
“Risky” cells

PDF

Normalized $V_{th}$

Risky cells

$+ S = \square \text{ P2}$

$+ F = \bigcirc \text{ P3}$

Key Formula

Uncorrectable errors

SAFARI
Retention Failure Recovery (RFR)

**Key idea:** Guess original state of the cell from its leakage speed property

**Three steps**

1. Identify risky cells
2. Identify fast-/slow-leaking cells
3. Guess original states
RFR Evaluation

- Expect to eliminate 50% of raw bit errors
- ECC can correct remaining errors

Program with random data

Detect failure, backup data

Recover data

28 days

12 addt’l. days
Characterize retention loss in real NAND chip

Optimize read performance for old data

Recover old data after failure
Conclusion

Retention loss → Longer read latency

Retention optimized reading (ROR)
→ Learns OPT periodically
→ 71% shorter read latency

Retention failure recovery (RFR)
→ Use leakage property to guess correct state
→ 50% error reduction before ECC correction
→ Recover data after failure
More on Flash Read Disturb Errors

Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
  - Neighbor-Assisted Error Correction
  - Read Disturb Error Handling
  - Retention Error Handling
  - Large Scale Field Analysis
- Summary
Large Scale Field Analysis of Flash Memory Errors
SSD Error Analysis of Facebook Systems

- First large-scale field study of flash memory errors
  [Slides (pptx) (pdf)] [Coverage at ZDNet] [Coverage on The Register] [Coverage on TechSpot] [Coverage on The Tech Report]

A Large-Scale Study of Flash Memory Failures in the Field

Justin Meza  
Carnegie Mellon University  
meza@cmu.edu

Qiang Wu  
Facebook, Inc.  
qwu@fb.com

Sanjeev Kumar  
Facebook, Inc.  
skumar@fb.com

Onur Mutlu  
Carnegie Mellon University  
onur@cmu.edu
A few SSDs cause most errors
A few SSDs cause most errors
Summary

SSD lifecycle

Access pattern dependence

New reliability trends

Read disturbance

Temperature
Early detection lifecycle period distinct from hard disk drive lifecycle.
SSD lifecycle

New reliability trends

Access pattern dependence
Read disturbance
Temperature
Storage lifecycle background: the bathtub curve for disk drives

Failure rate

Usage

[Schroeder+, FAST'07]
Storage lifecycle background: the bathtub curve for disk drives

[Schroeder+, FAST'07]
Storage lifecycle background: the bathtub curve for disk drives

Do SSDs display similar lifecycle periods?

[Schroeder+, FAST'07]
Use data written to flash to examine SSD lifecycle

(time-independent utilization metric)
720GB, 1 SSD  
720GB, 2 SSDs

SSD failure rate

Data written (TB)
720GB, 1 SSD    720GB, 2 SSDs

SSD failure rate

Data written (TB)

Early failure period

Useful life period

Wearout period
Early detection lifecycle period distinct from hard disk drive lifecycle.
SSD lifecycle

Access pattern dependence

New reliability trends

Read disturbance

Temperature
Temperature sensor
720GB, 1 SSD   720GB, 2 SSDs
High temperature: may **throttle** or **shut down**
1.2TB, 1 SSD  3.2TB, 1 SSD

Throttling
Throttling SSD usage helps mitigate temperature-induced errors.
Summary

We do not observe the effects of *read disturbance* errors in the field.
Summary

Throttling SSD usage helps mitigate temperature-induced errors.
Summary

We quantify the effects of the page cache and write amplification in the field.

Access pattern dependence

SSD lifecycle

Temperature
First large-scale field study of flash memory errors

[Slides (pptx) (pdf)] [Coverage at ZDNet] [Coverage on The Register] [Coverage on TechSpot] [Coverage on The Tech Report]
Other Works on NAND Flash Memory Modeling & Issues
Flash Memory Programming Vulnerabilities

- Yu Cai, Saugata Ghose, Yixin Luo, Ken Mai, Onur Mutlu, and Erich F. Haratsch,
  "Vulnerabilities in MLC NAND Flash Memory Programming: Experimental Analysis, Exploits, and Mitigation Techniques"
  [Slides (pptx) (pdf)] [Lightning Session Slides (pptx) (pdf)]

Vulnerabilities in MLC NAND Flash Memory Programming: Experimental Analysis, Exploits, and Mitigation Techniques

Yu Cai†, Saugata Ghose†, Yixin Luo††, Ken Mai†, Onur Mutlu§†, Erich F. Haratsch‡
†Carnegie Mellon University ††Seagate Technology §ETH Zürich
Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
  - Neighbor-Assisted Error Correction
  - Read Disturb Error Handling
  - Retention Error Handling
  - Large Scale Field Analysis
  - 3D NAND Flash Memory Reliability
- Summary
3D NAND Flash Memory
3D NAND Flash Reliability I [HPCA’18]


[Lightning Talk Video]
[Slides (pptx) (pdf)] [Lightning Session Slides (pptx) (pdf)]

HeatWatch: Improving 3D NAND Flash Memory Device Reliability by Exploiting Self-Recovery and Temperature Awareness

Yixin Luo†  Saugata Ghose†  Yu Cai‡  Erich F. Haratsch‡  Onur Mutlu§†

† Carnegie Mellon University  ‡ Seagate Technology  § ETH Zürich
[Abstract]
NAND Flash Memory Lifetime Problem

Flash lifetime decreases in each generation despite increased ECC strength.
Planar vs. 3D NAND Flash Memory

**Planar NAND Flash Memory**
- **Scaling**: Reduce flash cell size, Reduce distance b/w cells
- **Reliability**: Scaling hurts reliability

**3D NAND Flash Memory**
- **Scaling**: Increase # of layers
- **Reliability**: Not well studied!
Charge Trap Based 3D Flash Cell

- Cross-section of a charge trap transistor
2D vs. 3D Flash Cell Design

2D Floating-Gate Cell

3D Charge-Trap Cell
3D NAND Flash Memory Organization

Fig. 43. Organization of flash cells in an $M$-layer 3D charge trap NAND flash memory chip, where each block consists of $M$ wordlines and $N$ bitlines.
More Background and State-of-the-Art


**Errors in Flash-Memory-Based Solid-State Drives: Analysis, Mitigation, and Recovery**

YU CAI, SAUGATA GHOSE
Carnegie Mellon University

ERICH F. HARATSCH
Seagate Technology

YIXIN LUO
Carnegie Mellon University

ONUR MUTLU
ETH Zürich and Carnegie Mellon University
Table 4. Changes in behavior of different types of errors in 3D NAND flash memory, compared to planar (i.e., two-dimensional) NAND flash memory. See Section 6.2 for a detailed discussion.

<table>
<thead>
<tr>
<th>Error Type</th>
<th>Change in 3D vs. Planar</th>
</tr>
</thead>
<tbody>
<tr>
<td>P/E Cycling (Section 3.1)</td>
<td>3D is <em>less susceptible</em>, due to current use of charge trap transistors for flash cells</td>
</tr>
<tr>
<td>Program (Section 3.2)</td>
<td>3D is <em>less susceptible for now</em>, due to use of one-shot programming (see Section 2.4)</td>
</tr>
<tr>
<td>Cell-to-Cell Interference (Section 3.3)</td>
<td>3D is <em>less susceptible for now</em>, due to larger manufacturing process technology</td>
</tr>
<tr>
<td>Data Retention (Section 3.4)</td>
<td>3D is <em>more susceptible</em>, due to early retention loss</td>
</tr>
<tr>
<td>Read Disturb (Section 3.5)</td>
<td>3D is <em>less susceptible for now</em>, due to larger manufacturing process technology</td>
</tr>
</tbody>
</table>
Improving 3D NAND Flash Memory Lifetime by Tolerating Early Retention Loss and Process Variation

Yixin Luo  Saugata Ghose  Yu Cai  Erich F. Haratsch  Onur Mutlu
Executive Summary

- Problem: 3D NAND error characteristics are **not well studied**
- Goal: _Understand_ & _mitigate_ 3D NAND errors to improve lifetime

- Contribution 1: **Characterize** real 3D NAND flash chips
  - **Process variation:** $21 \times$ error rate difference across layers
  - **Early retention loss:** Error rate increases by $10 \times$ after 3 hours
  - **Retention interference:** Not observed before in planar NAND

- Contribution 2: **Model** RBER and threshold voltage
  - **RBER (raw bit error rate) variation model**
  - **Retention loss model**

- Contribution 3: **Mitigate** 3D NAND flash errors
  - **LaVAR:** Layer Variation Aware Reading
  - **LI-RAID:** Layer-Interleaved RAID
  - **ReMAR:** Retention Model Aware Reading
  - Improve flash lifetime by $1.85 \times$ or reduce ECC overhead by 78.9%
Agenda

• Background & Introduction

• Contribution 1: Characterize real 3D NAND flash chips

• Contribution 2: Model RBER and threshold voltage

• Contribution 3: Mitigate 3D NAND flash errors

• Conclusion
Agenda

• Background & Introduction

• Contribution 1: Characterize real 3D NAND flash chips
  • Process variation
  • Early retention loss
  • Retention interference

• Contribution 2: Model RBER and threshold voltage

• Contribution 3: Mitigate 3D NAND flash errors

• Conclusion
Process Variation Across Layers

Flash cells on different layers may have different error characteristics.
Characterization Methodology

- Modified firmware version in the flash controller
  - Controls the read reference voltage of the flash chip
  - Bypasses ECC to get raw data (with raw bit errors)
- Analysis and post-processing of the data on the server
Layer-to-Layer Process Variation

Max RBER

21 × 2.4 ×
Layer-to-Layer Process Variation

Large RBER variation across layers and LSB-MSB pages
Retention Loss Phenomenon

Planar NAND Cell

Control Gate
Gate Oxide
Floating Gate (Conductor)
Tunnel Oxide

Source
Substrate
Drain

3D NAND Cell

Charge Trap (Insulator)
Control Gate
Gate Oxide
Tunnel Oxide

Most dominant type of error in planar NAND. Is this true for 3D NAND as well?
Early Retention Loss

Retention errors increase quickly immediately after programming
Characterization Summary

• **Layer-to-layer process variation**
  • Large RBER variation across layers and LSB-MSB pages
  • → Need new mechanisms to tolerate RBER variation!

• **Early retention loss**
  • RBER increases quickly after programming
  • → Need new mechanisms to tolerate retention errors!

• **Retention interference**
  • Amount of retention loss correlated with neighbor cells’ states
  • → Need new mechanisms to tolerate retention interference!

• **More threshold voltage and RBER results in the paper:**
  3D NAND P/E cycling, program interference, read disturb, read variation, bitline-to-bitline process variation

• **Our approach** based on insights developed via our experimental characterization: Develop error models, and build online error mitigation mechanisms using the models
Agenda

• Background & Introduction

• Contribution 1: Characterize real 3D NAND flash chips

• Contribution 2: Model RBER and threshold voltage
  • Retention loss model
  • RBER variation model

• Contribution 3: Mitigate 3D NAND flash errors

• Conclusion
What Do We Model?

Read Reference Voltages

Threshold Voltage Distribution

Raw Bit Errors

Probability

MSB

LSB

11

01

00

10

Threshold Voltage (V_{th})
Optimal Read Reference Voltage

Probability

Threshold Voltage ($V_{th}$)

Raw Bit Errors

$V_a$, $V_b$, $V_c$
Retention Loss Model

Early retention loss can be modeled as a simple linear function of \( \log(\text{retention time}) \)
Retention Loss Model

• Goal: Develop a simple linear model that can be used online

• Models
  • Optimal read reference voltage ($V_b$ and $V_c$)
  • Raw bit error rate ($\log(RBER)$)
  • Mean and standard deviation of threshold voltage distribution ($\mu$ and $\sigma$)

• As a function of
  • Retention time ($\log(t)$)
  • P/E cycle count ($PEC$)

• e.g., $V_{opt} = (\alpha \times PEC + \beta) \times \log(t) + \gamma \times PEC + \delta$

• Model error <1 step for $V_b$ and $V_c$
• Adjusted $R^2 > 89\%$
RBER Variation Model

Variation-agnostic $V_{opt}$
- Same $V_{ref}$ for all layers optimized for the entire block

RBER distribution follows gamma distribution

KL-divergence error = 0.09
Agenda

• Background & Introduction

• Contribution 1: Characterize real 3D NAND flash chips

• Contribution 2: Model RBER and threshold voltage

• Contribution 3: Mitigate 3D NAND flash errors
  • LaVAR: Layer Variation Aware Reading
  • LI-RAID: Layer-Interleaved RAID
  • ReMAR: Retention Model Aware Reading

• Conclusion
LaVAR: Layer Variation Aware Reading

- **Layer-to-layer process variation**
  - Error characteristics are different in each layer
- **Goal:** Adjust read reference voltage **for each layer**

- **Key Idea:** Learn a **voltage offset (Offset)** for each layer
  - \[ V_{\text{opt}}^{\text{Layer aware}} = V_{\text{opt}}^{\text{Layer agnostic}} + Offset \]

- **Mechanism**
  - **Offset:** Learned once for each chip & stored in a table
    - Uses \((2 \times \text{Layers})\) Bytes memory per chip
  - \( V_{\text{opt}}^{\text{Layer agnostic}} \): Predicted by any existing \( V_{\text{opt}} \) model
    - E.g., ReMAR [Luo+Sigmetrics’18], HeatWatch [Luo+HPCA’18], OFCM [Luo+JSAC’16], ARVT [Papandreou+GLSVLSI’14]

- Reduces RBER on average by **43%** (based on our characterization data)
LI-RAID: Layer-Interleaved RAID

• **Layer-to-layer process variation**
  • Worst-case RBER much higher than average RBER

• **Goal:** Significantly reduce worst-case RBER

• **Key Idea**
  • Group flash pages on *less reliable layers* with pages on *more reliable layers*
  • Group **MSB pages** with **LSB pages**

• **Mechanism**
  • Reorganize RAID layout to eliminate worst-case RBER
  • **<0.8%** storage overhead
## Conventional RAID

<table>
<thead>
<tr>
<th>Wordline #</th>
<th>Layer #</th>
<th>Page</th>
<th>Chip 0</th>
<th>Chip 1</th>
<th>Chip 2</th>
<th>Chip 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>MSB</td>
<td>Group 0</td>
<td>Group 0</td>
<td>Group 0</td>
<td>Group 0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>LSB</td>
<td>Group 1</td>
<td>Group 1</td>
<td>Group 1</td>
<td>Group 1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>MSB</td>
<td>Group 2</td>
<td>Group 2</td>
<td>Group 2</td>
<td>Group 2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>LSB</td>
<td>Group 3</td>
<td>Group 3</td>
<td>Group 3</td>
<td>Group 3</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>MSB</td>
<td>Group 4</td>
<td>Group 4</td>
<td>Group 4</td>
<td>Group 4</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>LSB</td>
<td>Group 5</td>
<td>Group 5</td>
<td>Group 5</td>
<td>Group 5</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>MSB</td>
<td>Group 6</td>
<td>Group 6</td>
<td>Group 6</td>
<td>Group 6</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>LSB</td>
<td>Group 7</td>
<td>Group 7</td>
<td>Group 7</td>
<td>Group 7</td>
</tr>
</tbody>
</table>

**Worst-case RBER in any layer limits the lifetime of conventional RAID**
LI-RAID: Layer-Interleaved RAID

<table>
<thead>
<tr>
<th>Wordline #</th>
<th>Layer #</th>
<th>Page</th>
<th>Chip 0</th>
<th>Chip 1</th>
<th>Chip 2</th>
<th>Chip 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>MSB</td>
<td>Group 0</td>
<td>Blank</td>
<td>Group 4</td>
<td>Group 3</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>LSB</td>
<td>Group 1</td>
<td>Blank</td>
<td>Group 5</td>
<td>Group 2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>MSB</td>
<td>Group 2</td>
<td>Group 1</td>
<td>Blank</td>
<td>Group 5</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>LSB</td>
<td>Group 3</td>
<td>Group 0</td>
<td>Blank</td>
<td>Group 4</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>MSB</td>
<td>Group 4</td>
<td>Group 3</td>
<td>Group 0</td>
<td>Blank</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>LSB</td>
<td>Group 5</td>
<td>Group 2</td>
<td>Group 1</td>
<td>Blank</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>MSB</td>
<td>Blank</td>
<td>Group 5</td>
<td>Group 2</td>
<td>Group 1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>LSB</td>
<td>Blank</td>
<td>Group 4</td>
<td>Group 3</td>
<td>Group 0</td>
</tr>
</tbody>
</table>

Any page with worst-case RBER can be corrected by other reliable pages in the RAID group.
LI-RAID: Layer-Interleaved RAID

- **Layer-to-layer process variation**
  - Worst-case RBER much higher than average RBER
- **Goal**: Significantly reduce worst-case RBER

- **Key Idea**
  - Group flash pages on *less reliable layers* with pages on *more reliable layers*
  - Group *MSB pages* with *LSB pages*

- **Mechanism**
  - Reorganize RAID layout to eliminate worst-case RBER
  - *<0.8%* storage overhead

- Reduces worst-case RBER by **66.9%** (based on our characterization data)
ReMAR: Retention Model Aware Reading

• Early retention loss
  • Threshold voltage shifts quickly after programming

• Goal: Adjust read reference voltages based on retention loss

• Key Idea: Learn and use a retention loss model online

• Mechanism
  • Periodically characterize and learn retention loss model online
  • Retention time = Read timestamp - Write timestamp
    • Uses 800 KB memory to store program time of each block
  • Predict retention-aware $V_{opt}$ using the model

• Reduces RBER on average by 51.9%
  (based on our characterization data)
Impact on System Reliability

LaVAR, LI-RAID, and ReMAR improve flash lifetime or reduce ECC overhead significantly.
Error Mitigation Techniques Summary

- **LaVAR: Layer Variation Aware Reading**
  - Learn a $V_{opt}$ offset for each layer and apply *layer-aware $V_{opt}$*

- **LI-RAID: Layer-Interleaved RAID**
  - Group flash pages on *less reliable layers* with pages on *more reliable layers*
  - Group *MSB pages* with *LSB pages*

- **ReMAR: Retention Model Aware Reading**
  - Learn retention loss model and apply *retention-aware $V_{opt}$*

- **Benefits:**
  - Improve flash lifetime by $1.85\times$ or reduce ECC overhead by $78.9\%$

- **ReNAC (in paper):** Reread a failed page using $V_{opt}$ based on the *retention interference* induced by neighbor cell
Agenda

• Background & Introduction

• Contribution 1: Characterize real 3D NAND flash chips

• Contribution 2: Model RBER and threshold voltage

• Contribution 3: Mitigate 3D NAND flash errors

• Conclusion
Conclusion

• Problem: 3D NAND error characteristics are not well studied
• Goal: Understand & mitigate 3D NAND errors to improve lifetime

• Contribution 1: Characterize real 3D NAND flash chips
  • Process variation: $21 \times$ error rate difference across layers
  • Early retention loss: Error rate increases by $10 \times$ after 3 hours
  • Retention interference: Not observed before in planar NAND

• Contribution 2: Model RBER and threshold voltage
  • RBER (raw bit error rate) variation model
  • Retention loss model

• Contribution 3: Mitigate 3D NAND flash errors
  • LaVAR: Layer Variation Aware Reading
  • LI-RAID: Layer-Interleaved RAID
  • ReMAR: Retention Model Aware Reading
  • Improve flash lifetime by $1.85 \times$ or reduce ECC overhead by 78.9%
Improving 3D NAND Flash Memory Lifetime by Tolerating Early Retention Loss and Process Variation

Yixin Luo  Saugata Ghose  Yu Cai  Erich F. Haratsch  Onur Mutlu

Carnegie Mellon  SK hynix  ETH Zürich  SAFARI  SEAGATE
Yixin Luo, Saugata Ghose, Yu Cai, Erich F. Haratsch, and Onur Mutlu, "Improving 3D NAND Flash Memory Lifetime by Tolerating Early Retention Loss and Process Variation"
[Abstract]
One More Idea
Improving NAND Flash Memory Lifetime with Write-hotness Aware Retention Management

Yixin Luo, Yu Cai, Saugata Ghose, Jongmoo Choi*, Onur Mutlu
Carnegie Mellon University, *Dankook University
Executive Summary

• Flash memory can achieve **50x endurance improvement by relaxing retention time using refresh** [Cai+ ICCD ’12]

• **Problem:** Frequent refresh consumes the majority of endurance improvement

• **Goal:** Reduce refresh overhead to increase flash memory lifetime

• **Key Observation:** Refresh is unnecessary for **write-hot data**

• **Key Ideas of Write-hotness Aware Retention Management (WARM)**
  - Physically partition write-hot pages and write-cold pages within the flash drive
  - **Apply different policies** (garbage collection, wear-leveling, refresh) to each group

• **Key Results**
  - WARM w/o refresh **improves lifetime by 3.24x**
  - WARM w/ adaptive refresh **improves lifetime by 12.9x** (1.21x over refresh only)
Conventional Write-Hotness Oblivious Management

<table>
<thead>
<tr>
<th>Flash Memory</th>
<th></th>
<th></th>
<th>Erase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hot Page 1</td>
<td>Hot Page 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cold Page 2</td>
<td>Hot Page 4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hot Page 1</td>
<td>Cold Page 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cold Page 3</td>
<td>Cold Page 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hot Page 4</td>
<td>Cold Page 4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cold Page 5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hot Page 4</td>
<td>Page 511</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Unable to relax retention time for blocks with write-hot and cold pages
Key Idea: Write-Hotness Aware Management

Can relax retention time for blocks with write-hot pages only
Write-Hotness Aware Retention Management

Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
  - Neighbor-Assisted Error Correction
  - Read Disturb Error Handling
  - Retention Error Handling
  - Large Scale Field Analysis
  - 3D NAND Flash Memory Reliability
- Summary
Summary of Key Works


Cai+, "Read Disturb Errors in MLC NAND Flash Memory: Characterization and Mitigation," DSN 2015.

Meza+, "A Large-Scale Study of Flash Memory Errors in the Field," SIGMETRICS 2015.


Vulnerabilities in MLC NAND Flash Memory Programming: Experimental Analysis, Exploits, and Mitigation Techniques

Yu Cai†, Saugata Ghose†, Yixin Luo‡†, Ken Mai†, Onur Mutlu§†, Erich F. Haratsch‡
†Carnegie Mellon University ‡Seagate Technology §ETH Zürich

Modern NAND flash memory chips provide high density by storing two bits of data in each flash cell, called a multi-level cell (MLC). An MLC partitions the threshold voltage range of a flash cell into four voltage states. When a flash cell is programmed, a high voltage is applied to the cell. Due to parasitic capacitance coupling between flash cells that are physically close to each other, flash cell programming can lead to cell-to-cell program interference, which introduces errors into neighboring flash cells. In order to reduce the impact of cell-to-cell interference on the reliability of MLC NAND flash memory, flash manufacturers adopt a two-step programming method, which programs the MLC in two separate steps. First, the flash memory partially programs the least significant bit of the MLC to some intermediate threshold voltage. Second, it programs the most significant bit to bring the MLC up to its full voltage state.

In this paper, we demonstrate that two-step programming exposes new reliability and security vulnerabilities. We expect

NAND Flash Errors: A Modern Survey

Proceedings of the IEEE, Sept. 2017

Error Characterization, Mitigation, and Recovery in Flash-Memory-Based Solid-State Drives

This paper reviews the most recent advances in solid-state drive (SSD) error characterization, mitigation, and data recovery techniques to improve both SSD’s reliability and lifetime.

By Yu Cai, Saugata Ghose, Erich F. Haratsch, Yixin Luo, and Onur Mutlu

https://arxiv.org/pdf/1706.08642
Errors in Flash-Memory-Based Solid-State Drives: Analysis, Mitigation, and Recovery

YU CAI, SAUGATA GHOSE
Carnegie Mellon University

ERICH F. HARATSCH
Seagate Technology

YIXIN LUO
Carnegie Mellon University

ONUR MUTLU
ETH Zürich and Carnegie Mellon University
Computer Architecture
Lecture 26: Flash Memory and Solid-State Drives

Prof. Onur Mutlu
ETH Zürich
Fall 2020
31 December 2020
Other Works on Flash Memory
HeatWatch

Improving 3D NAND Flash Memory Device Reliability by Exploiting Self-Recovery and Temperature Awareness

Yixin Luo    Saugata Ghose    Yu Cai    Erich F. Haratsch    Onur Mutlu

Carnegie Mellon    SK hynix    ETH Zürich

SAFARI    SEAGATE
Storage Technology Drivers - 2018

3D NAND Flash Memory

Stacked layers

Store large amounts of data reliably for months to years

3D NAND Flash Memory

DNA
Executive Summary

- 3D NAND flash memory susceptible to retention errors
  - Charge leaks out of flash cell
  - Two unreported factors: self-recovery and temperature

- We study self-recovery and temperature effects

- Experimental characterization of real 3D NAND chips

- Unified Self-Recovery and Temperature (URT) Model
  - Predicts impact of retention loss, wearout, self-recovery, temperature on flash cell voltage
  - Low prediction error rate: 4.9%

- We develop a new technique to improve flash reliability

- HeatWatch
  - Uses URT model to find optimal read voltages for 3D NAND flash
  - Improves flash lifetime by 3.85x
Outline

• Executive Summary

• **Background on NAND Flash Reliability**

• Characterization of Self-Recovery and Temperature Effect on Real 3D NAND Flash Memory Chips

• URT: Unified Self-Recovery and Temperature Model

• HeatWatch Mechanism

• Conclusion
3D NAND Flash Memory Background

3D NAND Flash Memory

Flash Cell

Higher Voltage State
Data Value = 0

Charge = Threshold Voltage

Read Reference Voltage

Lower Voltage State
Data Value = 1
Flash Wearout

Program/Erase (P/E) $\rightarrow$ Wearout

Wearout Effects:

1. Retention Loss
   (voltage shift over time)

2. Program Variation
   (init. voltage difference b/w states)

Wearout Introduces Errors
Errors introduced by wearout limit flash lifetime (measured in P/E cycles)

Two Ways to Improve Flash Lifetime

Exploiting the Self-Recovery Effect
Exploiting the Temperature Effect
Exploiting the Self-Recovery Effect

Partially repairs damage due to wearout

Dwell Time: Idle Time Between P/E Cycles

Longer Dwell Time: More Self-Recovery

Reduces Retention Loss
Exploiting the Temperature Effect

- **High Program Temperature**
- **High Storage Temperature**

Increases Program Variation

Accelerates Retention Loss
Prior Studies of Self-Recovery/Temperature

<table>
<thead>
<tr>
<th></th>
<th>Planar (2D) NAND</th>
<th>3D NAND</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Self-Recovery Effect</strong></td>
<td>✔</td>
<td>✗</td>
</tr>
<tr>
<td></td>
<td>Mielke 2006</td>
<td></td>
</tr>
<tr>
<td><strong>Temperature Effect</strong></td>
<td>✔</td>
<td>✗</td>
</tr>
<tr>
<td></td>
<td>JEDEC 2010</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(no characterization)</td>
<td></td>
</tr>
</tbody>
</table>
Outline

• Executive Summary

• Background on NAND Flash Reliability

• Characterization of Self-Recovery and Temperature Effect on Real 3D NAND Flash Memory Chips

• URT: Unified Self-Recovery and Temperature Model

• HeatWatch Mechanism

• Conclusion
Characterization Methodology

• Modified firmware version in the flash controller
  • Control the read reference voltage of the flash chip
  • Bypass ECC to get raw NAND data (with raw bit errors)
• Control temperature with a heat chamber
Characterized Devices

Real 30-39 Layer 3D MLC NAND Flash Chips

2-bit MLC

30- to 39-layer
MLC Threshold Voltage Distribution Background

Lowest Voltage State

Read Reference Voltage

11

10

00

01

Highest Voltage State

Threshold Voltage Distribution

Probability
Characterization Goal

Characterized Metrics

Retention Loss Speed
(how fast voltage shifts over time)

Program Variation
(initial voltage difference between states)

Characterized Phenomena

Self-Recovery Effect

Temperature Effect
Increasing dwell time from 1 minute to 2.3 hours slows down retention loss speed by 40%
Increasing program temperature from 0°C to 70°C improves program variation by 21%
Lowering storage temperature from 70°C to 0°C slows down retention loss speed by 58%.
Characterization Summary

Major Results:
• *Self-recovery* affects retention loss speed
• Program *temperature* affects program variation
• *Storage temperature* affects retention loss speed

Other Characterizations Methods in the Paper:
• More detailed results on self-recovery and temperature
  • Effects on error rate
  • Effects on threshold voltage distribution
• Effects of recovery cycle (P/E cycles with long dwell time) on retention loss speed
Outline

• Executive Summary

• Background on NAND Flash Reliability

• Characterization of Self-Recovery and Temperature Effect on Real 3D NAND Flash Memory Chips

• **URT: Unified Self-Recovery and Temperature Model**

• HeatWatch Mechanism

• Conclusion
Minimizing 3D NAND Errors

Optimal read reference voltage minimizes 3D NAND errors
Predicting the Mean Threshold Voltage

Our URT Model:

\[ V = V_0 + \Delta V \]

- **Mean Threshold Voltage**
- **Initial Voltage Before Retention (Program Variation)**
- **Voltage Shift Due to Retention Loss**
URT Model Overview

1. Program Variation Component
   - PEC
   - $T_p$

2. Self-Recovery and Retention Component
   - $t_r$
   - $T_r$
   - $t_d$
   - $T_d$
   - $t_{r, eff}$
   - PEC
   - $t_{d, eff}$

3. Temperature Scaling Component

$V = V_0 + \Delta V$

- Initial Voltage Before Retention
- Voltage Shift Due to Retention Loss
1. Program Variation Component

P/E Cycle

Program Temperature

PEC

$T_p$

$V_0$

Initial Voltage

$V_0 = A \cdot T_p \cdot PEC + B \cdot T_p + C \cdot PEC + D$

Validation: $R^2 = 91.7\%$
2. Self-Recovery and Retention Component

- Retention Time: $t_r$
- P/E Cycle: PEC
- Dwell Time: $T_d$

Retention Shift: \[ \Delta V (t_{er}, t_{ed}, PEC) = b \cdot (PEC + c) \cdot \ln \left( 1 + \frac{t_{er}}{t_0 + a \cdot t_{ed}} \right) \]

Validation: 3x more accurate than state-of-the-art model
3. Temperature Scaling Component

Validation: Adjust an important parameter, $E_a$, from 1.1 eV to 1.04 eV

Arrhenius Equation: $AF = \frac{t_{\text{real}}}{t_{\text{room}}} = \exp \left( \frac{E_a}{k_B} \cdot \left( \frac{1}{T_{\text{real}}} - \frac{1}{T_{\text{room}}} \right) \right)$
### URT Model Summary

1. **Program Variation Component**
   - $P_E C$
   - $T_p$

2. **Self-Recovery and Retention Component**
   - $t_r$
   - $T_r$
   - $t_d$
   - $T_d$
   - $t_{r,eff}$
   - $t_{d,eff}$
   - PEC

3. **Temperature Scaling Component**
   - $t_{r,eff}$
   - $t_{d,eff}$

**Validation:**
- Prediction Error Rate = 4.9%

Mathematical Equation:
$$V = V_0 + \Delta V$$
Outline

• Executive Summary
• Background on NAND Flash Reliability
• Characterization of Self-Recovery and Temperature Effect on Real 3D NAND Flash Memory Chips
• URT: Unified Self-Recovery and Temperature Model
• **HeatWatch Mechanism**
• Conclusion
HeatWatch Mechanism

• Key Idea

  • **Predict change in threshold voltage distribution** by using the URT model

  • **Adapt read reference voltage to near-optimal** \(V_{\text{opt}}\) based on predicted change in voltage distribution
HeatWatch Mechanism Overview

Tracking Components

SSD Temperature
Dwell Time
P/E Cycles & Retention Time

Prediction Components

$V_{\text{opt}}$ Prediction
Fine-Tuning URT Parameters
Tracking SSD Temperature

Tracking Components

- SSD Temperature
- Dwell Time
- P/E Cycles & Retention Time

• Use existing sensors in the SSD
• **Precompute** temperature scaling factor at logarithmic time intervals

Prediction Components

- $V_{opt}$ Prediction
- Fine-Tuning URT Parameters
Tracking Dwell Time

Tracking Components

SSD Temperature  Dwell Time  P/E Cycles & Retention Time

Prediction Components

- Only need to log the timestamps of last 20 full drive writes
- Self-recovery effect diminishes after 20 P/E cycles

V_{opt} Prediction  Fine-Tuning URT Parameters
Tracking P/E Cycles and Retention Time

**Tracking Components**

- **SSD Temperature**
- **Dwell Time**
- **P/E Cycles & Retention Time**

**Prediction Components**

- $V_{opt}$ Prediction
- Fine-Tuning URT Parameters

- P/E cycle count **already recorded** by SSD
- **Log write timestamp** for each block
- Retention time = read timestamp – write timestamp
Predicting Optimal Read Reference Voltage

**Tracking Components**

- SSD Temperature
- Dwell Time
- P/E Cycles & Retention Time

**Prediction Components**

- Calculate URT using tracked information
- Modeling error: 4.9%

$V_{opt}$ Prediction

Fine-Tuning URT Parameters
Fine-Tuning URT Parameters Online

### Tracking Components
- SSD Temperature
- Dwell Time
- P/E Cycles & Retention Time

### Prediction Components
- $V_{opt}$ Prediction
- Fine-Tuning URT Parameters

- Accommodates **chip-to-chip variation**
- Uses **periodic sampling**
HeatWatch Mechanism Summary

Tracking Components

- SSD Temperature
- Dwell Time
- P/E Cycles & Retention Time

Storage Overhead: 0.16% of DRAM in 1TB SSD

URT

Prediction Components

- $V_{opt}$ Prediction
- Fine-Tuning URT Parameters

Latency Overhead: < 1% of flash read latency
HeatWatch Evaluation Methodology

• 28 real workload storage traces
  • MSR-Cambridge
  • We use real dwell time, retention time values obtained from traces

• Temperature Model:
  Trigonometric function + Gaussian noise
  • Represents periodic temperature variation in each day
  • Includes small transient temperature variation
HeatWatch Greatly Improves Flash Lifetime

HeatWatch improves lifetime by capturing the effect of retention, wearout, self-recovery, temperature

Graph showing the comparison of various Flash memory benchmarks:

- **Fixed $V_{\text{ref}}$**
- **State-of-the-art**
- **ECC limit**
- **Oracle**

The graph indicates:
- **3.85x over Fixed $V_{\text{ref}}$**
- **24% over state-of-the-art**
Outline

• Executive Summary

• Background on NAND Flash Reliability

• Characterization of Self-Recovery and Temperature Effect on Real 3D NAND Flash Memory Chips

• URT: Unified Self-Recovery and Temperature Model

• HeatWatch Mechanism

• Conclusion
Conclusion

• 3D NAND flash memory susceptible to retention errors
  • Charge leaks out of flash cell
  • Two unreported factors: self-recovery and temperature

• We study self-recovery and temperature effects

  • Experimental characterization of real 3D NAND chips

• Unified Self-Recovery and Temperature (URT) Model
  • Predicts impact of retention loss, wearout, self-recovery, temperature on flash cell voltage
  • Low prediction error rate: 4.9%

• We develop a new technique to improve flash reliability

  • HeatWatch
    • Uses URT model to find optimal read voltages for 3D NAND flash
    • Improves flash lifetime by 3.85x
HeatWatch

Improving 3D NAND Flash Memory Device Reliability by Exploiting Self-Recovery and Temperature Awareness

Yixin Luo   Saugata Ghose   Yu Cai   Erich F. Haratsch   Onur Mutlu

Carnegie Mellon   SK hynix   ETH Zürich   SAFARI   SEAGATE