Error Characterization and Comparison of ECCs on MLC and TLC Flash Memories

Veeresh Taranalli, Eitan Yaakobi, Paul H. Siegel

Center for Magnetic Recording Research
University of California, San Diego
Outline

- Flash Memory Basics
- Error Characterization
- Performance comparison of BCH, LDPC codes
  - Various decoding techniques
- Polar Codes
  - LP Decoding of Polar codes
Acknowledgment

CMMR – STAR Group Members
Brian Butler
Scott Kayser
Xiaojie Zhang
Aman Bhatia
Minghai Qin
Flash Memory Basics

- Arrays ("blocks") of floating-gate transistors ("cells")

- A cell can support $q$ voltage levels e.g., $q = 2, 4, 8$.

- Increasing the voltage level (program) of a cell is easy to do.

- To decrease a cell level, we must first erase its entire block, then re-program all cells.

- A block erase is costly in time, power and cell wear.
SLC, MLC, TLC Flash

- **SLC Flash**
  - 1 Bit Per Cell
  - 2 States
  - High Voltage
    - 0
  - Low Voltage
    - 1

- **MLC Flash**
  - 2 Bits Per Cell
  - 4 States
  - High Voltage
    - 00
    - 01
  - Low Voltage
    - 10
    - 11

- **TLC Flash**
  - 3 Bits Per Cell
  - 8 States
  - High Voltage
    - 000
    - 001
    - 010
    - 011
    - 100
    - 101
    - 110
    - 111
  - Low Voltage
Flash Memory Structure - SLC

- Group of cells → Page
- Group of Pages → Block

Typical SLC Block Layout

<table>
<thead>
<tr>
<th>page 0</th>
<th>page 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>page 2</td>
<td>page 3</td>
</tr>
<tr>
<td>page 4</td>
<td>page 5</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>page 62</td>
<td>page 63</td>
</tr>
</tbody>
</table>
### Flash Memory Structure - MLC

- **2 bits/cell**: MSB and LSB pages

<table>
<thead>
<tr>
<th>Row index</th>
<th>MSB of first (2^{14}) cells</th>
<th>LSB of first (2^{14}) cells</th>
<th>MSB of last (2^{14}) cells</th>
<th>LSB of last (2^{14}) cells</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>page 0</td>
<td>page 4</td>
<td>page 1</td>
<td>page 5</td>
</tr>
<tr>
<td>1</td>
<td>page 2</td>
<td>page 8</td>
<td>page 3</td>
<td>page 9</td>
</tr>
<tr>
<td>2</td>
<td>page 6</td>
<td>page 12</td>
<td>page 7</td>
<td>page 13</td>
</tr>
<tr>
<td>3</td>
<td>page 10</td>
<td>page 16</td>
<td>page 11</td>
<td>page 17</td>
</tr>
<tr>
<td></td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
</tr>
<tr>
<td>30</td>
<td>page 118</td>
<td>page 124</td>
<td>page 119</td>
<td>page 125</td>
</tr>
<tr>
<td>31</td>
<td>page 122</td>
<td>page 126</td>
<td>page 123</td>
<td>page 127</td>
</tr>
</tbody>
</table>

**MSB/LSB**

- 01
- 00
- 10
- 11
Flash Memory Structure - TLC

- 3 bits/cell $\rightarrow$ MSB, CSB and LSB pages

<table>
<thead>
<tr>
<th>Row index</th>
<th>MSB of first $2^{16}$ cells</th>
<th>CSB of first $2^{16}$ cells</th>
<th>LSB of first $2^{16}$ cells</th>
<th>MSB of last $2^{16}$ cells</th>
<th>CSB of last $2^{16}$ cells</th>
<th>LSB of last $2^{16}$ cells</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>page 0</td>
<td></td>
<td></td>
<td>page 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>page 2</td>
<td>page 6</td>
<td>page 12</td>
<td>page 3</td>
<td>page 7</td>
<td>page 13</td>
</tr>
<tr>
<td>2</td>
<td>page 4</td>
<td>page 10</td>
<td>page 18</td>
<td>page 5</td>
<td>page 11</td>
<td>page 19</td>
</tr>
<tr>
<td>3</td>
<td>page 8</td>
<td>page 16</td>
<td>page 24</td>
<td>page 9</td>
<td>page 17</td>
<td>page 25</td>
</tr>
<tr>
<td>4</td>
<td>page 14</td>
<td>page 22</td>
<td>page 30</td>
<td>page 15</td>
<td>page 23</td>
<td>page 31</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>62</td>
<td>page 362</td>
<td>page 370</td>
<td>page 378</td>
<td>page 363</td>
<td>page 371</td>
<td>page 379</td>
</tr>
<tr>
<td>63</td>
<td>page 368</td>
<td>page 376</td>
<td></td>
<td>page 369</td>
<td>page 377</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>page 374</td>
<td>page 382</td>
<td></td>
<td>page 375</td>
<td>page 383</td>
<td></td>
</tr>
<tr>
<td>65</td>
<td>page 380</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>page 381</td>
</tr>
</tbody>
</table>
Error Characterization

- Program/Erase (P/E) cycling of many blocks on MLC and TLC flash memories
- For each block the following steps were repeated:
  - The block is erased.
  - Pseudo-random data are programmed to the block.
  - The data are read and errors are identified.

**Disclaimers:**
- We measured many more P/E cycles than the manufacturer’s guaranteed lifetime of the device.
- The experiments were done in laboratory conditions and related factors such as temperature change, intervals between erasures, or multiple readings before erasures were not considered.
Experiment Setup
Flash Chip Interface (Ming I Board)
Experiment Setup
FPGA Controller
Results – BER, MLC

MLC BER

BER

Program/Erase Cycle

x10^{-3}

x10^5
Results – BER per page, MLC

MLC BER per page (averaged on every 250 iterations)

Pages, colored the same, behave similarly

Flash Memory Summit 2014
Santa Clara, CA
Results – BER, TLC
Results – BER per page, TLC

Average BER for Every Page

- Left MSB Page
- Right MSB Page
- Left CSB Page
- Right CSB Page
- Left LSB Page
- Right LSB Page

M/C/L

011
010
000
001
101
100
110
111
ECC Comparison for TLC Flash

- BCH Codes
  - Length $2^{16}$

- LDPC Codes
  - Gallager codes (3,k)-regular, $R=0.8$, 0.9, 0.925, length $2^{16}$
  - AR4JA protograph-based codes, $R=0.8$, lengths 1280, 5120, 20480
  - MacKay codes variable-regular degree (3 or 4); no 4-cycles, $R=0.82$, 0.87, 0.93; lengths 4095, 16383, 32000.
  - IEEE 802.3an* (10Gb/s Ethernet), $R \approx 0.84$, length 2048.

ECC Comparison for TLC Flash

- BER computed for the first 100 iterations, then every 25\textsuperscript{th} iteration from then on.
- Data averaged over 6 TLC blocks.
- BCH decoder: corrects error patterns with up to \( t \) errors; detects and leaves unchanged more than \( t \) errors.
- LDPC decoders: assume binary symmetric channel model \( \text{BSC}(p) \), with empirical error probability \( p \).
LDPC Decoders

- Sum-product algorithm (SPA)
  - Floating-point, max iterations 200
  - (5+1)-bit quasi-uniform quantization
- Min-sum algorithm (MSA)
  - No LLR limits, max iterations 200
- Linear programming (LP) decoding
  - Alternating Direction Method of Multipliers (ADMM)*
    with new fast “projection step”

Rate ≈ 0.8, LDPC codes with SPA Decoding

BER of Different Codes of Rate ≈ 0.8
Rate ≈ 0.82, LDPC codes with SPA Decoding

BER of Different Codes of Rate ≈ 0.8

RAW BER
- BCH 65536 (R=0.8)
- 802.3an (R=0.84)
- DJCM-3 (R=0.82)
- DJCM-4 (R=0.82)
- AR4JA 1280 (R=0.8)
Rate $\approx 0.9$, LDPC codes with SPA Decoding

Rate \approx 0.925, LDPC codes with SPA Decoding

Rate ≈ 0.8, MSA v/s SPA Decoding

BER of Different Codes of Rate ≈ 0.8

- RAW BER
- DJCM-3 MSA (R=0.82)
- DJCM-4 MSA (R=0.82)
- 802.3an MSA (R=0.84)
- DJCM-3 SPA (R=0.82)
- DJCM-4 SPA (R=0.82)
- 802.3an SPA (R=0.84)
Other Decoding Techniques

- Linear Programming (LP) decoding of LDPC codes
  - LP decoding with Alternating Direction Method of Multipliers (ADMM) proposed by Barman, et al., in 2011 to speed up LP decoding
  - A more efficient scheme based upon Adaptive LP decoding (ALP) with fast Cut-Search Algorithm (CSA) to further speed up the key “projection step” in LP-ADMM.

- SPA with \((q+1)\)-bit Quasi-uniform Quantization


Legend Notation

- **M4376**: MacKay code, length 4376 and rate 0.9356
- **DJCM-4**: MacKay code, length 3200 and rate 0.93
- **LP**: ADMM-based LP decoder, max iterations 200
- **ft-SPA**: floating-point SPA
- **Quantized SPA**: (5+1)-bit quasi-uniform quantized SPA
Rate $\approx 0.925$

LP vs. SPA Decoding on TLC

BER of Different Codes of Rate $\sim 0.925$

- RAW BER
- M4376 LP (R=0.94)
- M4376 quantized SPA (R=0.94)
- Gallager (R=0.925)
- BCH (R=0.925)
- DJCM-4 LP (R=0.93)
- DJCM-4 ft-SPA (R=0.93)
- DJCM-4 quantized SPA (R=0.93)
Summary of Observations

- Best LDPC performance surpasses BCH at all code rates $R \approx 0.8, 0.9, 0.925$.
- $R \approx 0.8$ LDPC codes at 15k cycles has BER comparable to $R \approx 0.9$ LDPC codes at 10k cycles.
- MSA was inferior to SPA decoding at $R \approx 0.8$.
- LP-ADMM was comparable to SPA decoding at $R \approx 0.925$, with slightly steeper slope.
- $(5+1)$-bit quasi-uniform quantized SPA (not optimized) matches floating-point SPA.
Polar Codes

- Based on the phenomenon of channel polarization
- Achieve the capacity of symmetric binary input discrete memoryless channels under successive cancellation (SC) decoding *
- CRC concatenated Polar codes with SC-List decoding can beat LDPC code performance (Tal and Vardy, 2011)

LP Decoding of Polar Codes

- High density parity check codes
- Goela et al. in 2010 proposed a sparse LP polytope representation which works for a binary erasure channel but doesn’t work well for a binary AWGN channel.
- We present results using a modified adaptive LP decoder for polar codes and a lower complexity polytope representation based on reduction of the polar code sparse factor graph.*

(128, 64) Polar Code over a BAWGN channel
Future Directions

- Characterization of inter-cell interference (ICI) and design of codes to combat it, e.g., ECC + constrained codes.

- Optimization of ECC schemes for better flash memory channel models e.g., asymmetric channels, non-binary channels, time-varying channels.

- Adaptation of new ECC techniques (polar codes, spatially-coupled LDPC codes) to flash memory applications.
Thank You!