Temperature-Aware Data Allocation Strategy for 3D Charge-Trap Flash Memory

Yi Wang¹, Mingxu Zhang¹, Jing Yang²

¹ Shenzhen University
² Harbin Institute of Technology

yiwang@szu.edu.cn
Flash Memory Properties

- Faster access performance
- Lower power consumption
- Smaller size
- Lighter weight
- Shock resistance
2D (Planar) Flash Memory

- Floating Gate (FG)

The number of electrons on the floating gate affects the threshold voltage of the cell.
Both SLC and MLC use the threshold voltage to manipulate the state of the flash.

<table>
<thead>
<tr>
<th>Value</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Programmed</td>
</tr>
<tr>
<td>1</td>
<td>Erased</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Value</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Fully Programmed</td>
</tr>
<tr>
<td>01</td>
<td>Partially Programmed</td>
</tr>
<tr>
<td>10</td>
<td>Partially Erased</td>
</tr>
<tr>
<td>11</td>
<td>Fully Erased</td>
</tr>
</tbody>
</table>
2D Flash $\rightarrow$ 3D Flash

- Process Improvement
2D Flash → 3D Flash

- Process Improvement
- SLC → MLC → TLC → QLC
2D Flash $\rightarrow$ 3D Flash

- Floating Gate (2D) $\rightarrow$ Charge-Trap (3D)
## 3D Flash Memory

<table>
<thead>
<tr>
<th></th>
<th>TOSHIBA / Sandisk</th>
<th>Intel / Micron</th>
<th>SAMSUNG</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Technology</strong></td>
<td>BiCS</td>
<td>FG</td>
<td>V-NAND</td>
</tr>
<tr>
<td><strong>Layers</strong></td>
<td>48</td>
<td>32</td>
<td>48</td>
</tr>
<tr>
<td><strong>Release</strong></td>
<td>2015</td>
<td>2015</td>
<td>2013</td>
</tr>
<tr>
<td><strong>Capacity</strong></td>
<td>128Gb (MLC)</td>
<td>256Gb (MLC)</td>
<td>128Gb (MLC)</td>
</tr>
<tr>
<td></td>
<td>256Gb (TLC)</td>
<td>384Gb (TLC)</td>
<td>256Gb (TLC)</td>
</tr>
</tbody>
</table>
Thermal is a critical issue for 3D flash

- The variation of $I_{BL}$ waveform is strongly activated by the temperature [1-2].
- High temperature introduces charge loss and worse retention behavior.


TempLoad: A Temperature-Aware Data Allocation Strategy for 3D Flash Memory

- TempLoad allocates physical space based on the temperature status.
- TempLoad does not require prior knowledge of temperature.
- TempLoad does not need to maintain the temperature status of each physical block.
- **Objective**: reduce peak temperature and enhance data integrity
System Architecture of TempLoad

Application 1 -> Application 2 -> ... -> Application n

File Systems
(e.g., NTFS, FAT32, ext3/4)

Device Drivers
Translation Module (FTL)

Low-Level Driver Interface
- Error Handling
- Block Management
- Platform Dependent Layer

TempLoad

3D Charge-Trap Flash Memory
Reliability and Thermal Models

- We use lifetime reliability (LR) or instantaneous mean time to failure (MTTF) to model the impact on flash temperature [3].

\[ LR \propto \left[ \ln \left( \frac{a}{1 + 2e^{b/\kappa T}} \right) - \ln \left( \frac{a}{1 + 2e^{b/\kappa T}} - c \right) \right] \times \frac{T}{e^{-d/\kappa T}} \right]^{\frac{1}{\delta}} \]

- The temperature differences can be modelled as [4-5]

\[ C \frac{dT}{dt} = R^{-1} T(t) - pU(t) \]


Block Selection with Temperature Mining

- Using this multiple-level quad tree, TempLoad only needs to compare the temperatures of a fixed number of sampling points at each stage.
Data Allocation with Set-Associative Block

- Solve the large-capacity block issue in 3D flash.
- Reduce the unnecessary garbage collection.
Evaluation

- TempLoad was implemented on the built-in NAND flash memory driver in Linux kernel 4.0.2.
- 3D-CBM [6] is selected as the baseline scheme.
- A 256Gb NAND flash memory is configured based on the specifications of a 3D flash memory test chip PF29F32B2ALCMG2 from Intel.

<table>
<thead>
<tr>
<th>Trace</th>
<th># of write operations</th>
<th># of read operations</th>
<th>% of write</th>
<th>% of read</th>
</tr>
</thead>
<tbody>
<tr>
<td>Financial</td>
<td>4,099,354</td>
<td>1,235,633</td>
<td>76.84</td>
<td>23.16</td>
</tr>
<tr>
<td>Webserver</td>
<td>1,260</td>
<td>4,260,449</td>
<td>0.03</td>
<td>99.97</td>
</tr>
<tr>
<td>onlineGames</td>
<td>653,133</td>
<td>817,887</td>
<td>44.40</td>
<td>55.60</td>
</tr>
<tr>
<td>KV-store</td>
<td>2,087,310</td>
<td>247,489</td>
<td>89.40</td>
<td>10.60</td>
</tr>
</tbody>
</table>

The thermal image for standard benchmarks *Financial* (Figures (a) and (b)) and *onlineGames* (Figures (c) and (d)).
Peak and Average Temperature

(a) Financial

(b) onlineGames

(c) Webserver

(d) KV-store
Data Integrity

(a) *Financial*

(b) *onlineGames*

(c) *Webserver*

(d) *KV-store*
# Response Time & Block Erase Counts

- **Response time**

<table>
<thead>
<tr>
<th></th>
<th>3D-CBM (s)</th>
<th>TempLoad (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Financial</td>
<td>1.2469</td>
<td>0.9601</td>
</tr>
<tr>
<td>Webserver</td>
<td>1.0632</td>
<td>0.9620</td>
</tr>
<tr>
<td>onlineGames</td>
<td>2.5931</td>
<td>1.7078</td>
</tr>
<tr>
<td>KV-Store</td>
<td>3.3024</td>
<td>3.2281</td>
</tr>
</tbody>
</table>

- **Block erase counts**

<table>
<thead>
<tr>
<th></th>
<th>3D-CBM</th>
<th>TempLoad</th>
</tr>
</thead>
<tbody>
<tr>
<td>Financial</td>
<td>1,944,522</td>
<td>1,470,692</td>
</tr>
<tr>
<td>Webserver</td>
<td>1,096,940</td>
<td>955,315</td>
</tr>
<tr>
<td>onlineGames</td>
<td>1,174,240</td>
<td>787,942</td>
</tr>
<tr>
<td>KV-Store</td>
<td>2,284,932</td>
<td>2,172,934</td>
</tr>
</tbody>
</table>
Conclusion

- We present TempLoad, the first system-level temperature-aware data allocation strategy for 3D charge-trap flash memory.
- TempLoad allocates critical data to relatively low temperature physical blocks and improves the physical space utilization ratio.
- Experimental results show that TempLoad can significantly reduce the peak temperature and effectively enhance the data integrity.
Thank you!