Density Tradeoffs of Non-Volatile Memory as a Replacement for SRAM based Last Level Cache

Kunal Korgaonkar, Ishwar Bhati, Huichu Liu, Jayesh Gaur, Sasikanth Manipatruni, Sreenivas Subramoney, Tanay Karnik, Steven Swanson, Ian Young and Hong Wang
Novel Non-volatile Memory based Last Level Cache (NVMLLC) Architecture

• Next generation NVM has potential for bigger and energy efficient LLC
  ✓ Potential ~3x capacity gain over state-of-art SRAM with logic compatible process, non-volatility
  ○ Write error rate (WER) target for industry LLC adoption increases write latency in practice

• Our proposed solutions show good performance gains and can help make NVM as viable replacement of SRAM for LLC
Agenda

- Motivation & problem
- Current solutions
- Our proposals
- Results
NVMs offer capacity advantages over SRAMs for LLC

- NVMs promise high density
- Spin Torque Transfer (STT) RAM, Spin Hall Effect (SHE) MRAM, etc..
- Can build large LLCs
- Significant power/density benefits over SRAM LLC
Advantage of increasing LLC capacity

Perf. normalized to 4MB SRAM LLC

- SRAM 4MB
- SRAM 8MB
- SRAM 16MB

1.25
1.20
1.15
1.10
1.05
1.00
0.95
0.90
But, high write latency negates the capacity gains

Performance normalized to 4MB SRAM LLC

- SRAM 4MB
- STTRAM 8MB WR +0ns
- STTRAM 8MB WR +5ns
- STTRAM 8MB WR +10ns
- STTRAM 8MB WR +20ns
None of the current techniques reduce the write latency enough

- Architectural Techniques
  - Dead block predictor for bypassing
  - LAP
  - Hybrid Cache

- Circuit and Device Techniques
  - Increase bit-cell transistor size, trade-off latency with retention/higher WER, new devices, etc
Our Proposal:

1. Reduce Write Interference
2. Eliminate Redundant Writes
Reduce Write Interference

- Many programs exhibit long high read/write phases
- Usual Dead Block Predictor based bypassing not sufficient
- Need more aggressive write bypassing to reduce write interference
Write Congestion Aware Bypassing (WCAB)

If any read ready

Send read

Send write

NO

Request queue is full && pending writes > write_th

Don’t bypass

Get average write occupancy calculated in intervals (int_write_occ)

Refer Lookup Table to find bypass score threshold (byp_score_th) for int_write_occ

Lookup Table (Tuned)

<table>
<thead>
<tr>
<th>Interval write occupancy (int_write_occ)</th>
<th>Bypass score threshold (byp_score_th)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/4th of request queue</td>
<td>20%</td>
</tr>
<tr>
<td>Half of request queue</td>
<td>50%</td>
</tr>
<tr>
<td>3/4th of request queue</td>
<td>70%</td>
</tr>
<tr>
<td>Equal to request queue</td>
<td>100%</td>
</tr>
</tbody>
</table>

write_th 75% of request queue

Find pending write with lowest live score (min_score)

min_score <= byp_score_th

Don’t bypass

Bypass write with min_score
Eliminates Redundant Writes

- Significant percentage of frequent clean and dirty fills in LLC
  - Dirty fills generate writes in both Exclusive and Inclusive LLC
  - Clean fills create writes in Exclusive LLC
Virtual Hybrid Cache (VHC)

- Write Merging in L2
  - Frequent dirty lines stay in L2 for longer
  - Used existing technique to classify frequent dirty lines
  - Many writes merge in L2 reducing fills in LLC
- Relaxed Exclusivity (duplicate lines b/w L2 and LLC)
  - Enhancement over LAP for Exclusive Cache
  - Retain the duplicate lines near LRU to reduce hit rate loss
  - Dirty lines (whenever found) not duplicated in LLC
Simulation Methodology & Results
Simulation Methodology

- Used modified version Multi2Sim simulating 4 x86 cores
  - Core parameters similar to Intel Skylake
  - SRAM baseline: 4MB, 4 banks, 16 ways with round trip delay of 20 cycles
  - STTRAM baseline: 8MB, 8 banks, additional write latency of 20ns
- Workloads:
  - Selected 20 workloads from SPEC 2006 and HPCG
  - With High L2 MPKI and a range of LLC MPKIs (Table 1 in the paper)
  - 20 homogeneous and 44 heterogeneous (by randomly mixing the 20 workloads)
Our proposals provide 26% performance gain over the baseline
Our proposals provide up to 18% performance gain over the SRAM of same area
Performance vs Prior Art

Our proposals perform significantly better than the prior art
Conclusions

• Next generation NVM has potential for bigger and energy efficient LLC
• Require architectural solutions to absorb high write latency and obtain capacity benefits
• Our proposed solutions show good performance gains and can help make NVM as viable replacement of SRAM for LLC

THANK YOU!😊!