Hybrid Cache Architecture (HCA) with Disparate Memory Technologies

Xiaoxia Wu†, Jian Li‡, Lixin Zhang‡,
Evan Speight‡, Ram Rajamony‡, Yuan Xie†

† Pennsylvania State University
‡ IBM Austin Research Laboratory

Acknowledgement: Elmootazbellah (Mootaz) Elnozahy, Hung Le,
Balaram Sinharoy, William (Bill) J. Starke, and Chung-Lung Kevin Shum
Motivation and Introduction
Methodology
Level based Hybrid Cache Architecture
Region based Hybrid Cache Architecture
3D Hybrid Cache Stacking
Conclusion
Introduction

Traditional SRAM-based Cache Architecture
- Limited size with CMP: cache-core balance
- Leakage power
- More cache levels: design overhead, coherence
- Non-Uniform Cache Architecture (wire delay)

Improve cache power-performance with Emerging Memory Technologies, under the same chip area/footprint
- Embedded DRAM
- Magnetic RAM
- Phase Change RAM
- Three-dimensional space
Different Memory Technologies

- **SRAM**: 6T structure
- **DRAM**: 1T1C structure
- **Magnetic RAM**: 1T1J structure
- **Phase Change RAM**: 1T1J structure

MTJ (Magnetic Tunnel Junction)
## Comparisons

<table>
<thead>
<tr>
<th></th>
<th>SRAM</th>
<th>eDRAM</th>
<th>MRAM</th>
<th>PRAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Density (ratio)</td>
<td>Low (1)</td>
<td>High (4)</td>
<td>High (4)</td>
<td>High(16)</td>
</tr>
<tr>
<td>Dynamic Power</td>
<td>Low</td>
<td>Medium</td>
<td>Low for read; High for write</td>
<td>Medium for read; High for write</td>
</tr>
<tr>
<td>Reduce Cache miss rate</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Increase hit latency</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Leakage Power</td>
<td>High</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Speed</td>
<td>Very Fast</td>
<td>Fast</td>
<td>Fast for read; Slow for write</td>
<td>Slow for read; Very slow for write</td>
</tr>
<tr>
<td>Non-volatility</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Scalability</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Endurance</td>
<td>$10^{16}$</td>
<td>$10^{16}$</td>
<td>$&gt;10^{15}$</td>
<td>$10^8$</td>
</tr>
</tbody>
</table>
Motivation

No single memory technology has the best power-performance.

Hybrid Cache may outperform its counterpart of single technology.
Outline

- Introduction and Motivation
- **Methodology**
- Level based Hybrid Cache Architecture
- Region based Hybrid Cache Architecture
- 3D Hybrid Cache Stacking
- Conclusions
Evaluation Methodology

Baseline: a 2D 8-core CMP (3-level SRAM Caches)

3DHCA

Flattening L3 and L4 with hybrid cache

3DHCA

Flattening L2, L3 and L4 with hybrid cache

3DHCA

A cache design scenario with 3D chip integration
## Evaluation Setup

<table>
<thead>
<tr>
<th>Cache</th>
<th>Density</th>
<th>Latency (cycles)</th>
<th>Dyn. eng (nJ)</th>
<th>Static power (W)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRAM(1MB)</td>
<td>1</td>
<td>8</td>
<td>0.388</td>
<td>1.36</td>
</tr>
<tr>
<td>eDRAM(4MB)</td>
<td>4</td>
<td>24</td>
<td>0.72</td>
<td>0.4</td>
</tr>
<tr>
<td>MRAM(4MB)</td>
<td>4</td>
<td>Read:20, write:60</td>
<td>Read:0.4</td>
<td>0.15</td>
</tr>
<tr>
<td>PRAM(16MB)</td>
<td>16</td>
<td>Read:40, write:200</td>
<td>Read:0.8</td>
<td>0.3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Processor</th>
<th>8-way issue out-of-order, 8-core, 4GHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1</td>
<td>32KB DL1, 32KB IL1, 128B, 4-way, 1 R/W port</td>
</tr>
<tr>
<td>L2/L3/L4</td>
<td>See corresponding design cases</td>
</tr>
<tr>
<td>Memory</td>
<td>400 cycles latency</td>
</tr>
</tbody>
</table>

- **Benchmarks**: SpecInt06, Specjbb, NAS, Bioperf, Parsec
- **Simulator**: SystemSim full system simulator
Outline

- Introduction and Motivation
- Methodology
- **Level based Hybrid Cache Architecture**
- Intra-Level Hybrid Cache Architecture
- 3D Hybrid Cache Stacking
- Conclusions
LHCA: Performance and Power

Simple LHCA can provide Performance and Power benefits

Parameters from the methodology

Core w/ L1s
L2 (SRAM)

Normalized IPC

Normalized Power

Core w/ L1s
L2 (SRAM)

L1 32KB DL1, 32KB IL1, 128B, 4-way, 1 R/W port
L2 256KB, 128B, 4-way, 1 R/W port
L3 32KB DL1, 32KB IL1, 128B, 4-way, 1 R/W port
Outline

- Introduction and Motivation
- Methodology
- Level based Hybrid Cache Architecture
- Region based Hybrid Cache Architecture
- 3D Hybrid Cache Stacking
- Conclusions
RHCA Features

- Fast and slow regions in one cache level
- Intra-cache data movement policy: Move frequently used data to the fast region
- Mutually Exclusive Regions
- Parallel search
  Unified LRU
- Drowsy RHCA: Keep slow region in drowsy mode, wake up latency (details in paper)
RHCA Policy

- Cache line migration policy

Access cache line

- Hit?
  - Y
    - Provide data; increment saturation counter
    - MSB set (hot) in slow region?
      - Y
        - Select an LRU cache line of the same set in the fast region
          - Sticky bit set?
            - Y
              - Clear sticky bit
            - N
              - Swap; reset saturation counter; set sticky bit
      - N
        - No action
  - N
    - Allocate the new line and replace an LRU line if needed, across all regions
      - Reset 2-bit saturation counter; set sticky bit
    - No action
● Hardware support
  ○ Saturating counter in slow and sticky bit in fast, swap buffer
  ○ Minimum hardware support: 1-bit sticky bit in fast region
RHCA Configuration

<table>
<thead>
<tr>
<th>RHCA (fast+slow)</th>
<th>Fast region</th>
<th>L2 total size (latency)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRAM+eDRAM</td>
<td>Core w/ L1s</td>
<td>4MB (24 cycles)</td>
</tr>
<tr>
<td>SRAM+MRAM</td>
<td>L2 Fast</td>
<td>4MB (r:20, w:60)</td>
</tr>
<tr>
<td>SRAM+PRAM</td>
<td>L2 Slow</td>
<td>16MB (r:40, w:200)</td>
</tr>
</tbody>
</table>

- Slow region: 256KB/bank, 1 r/w port, block size 128B, associativity 16, 64
- RHCA is 256KB less size than corresponding LHCA
  - Avoid odd-sized cache
- DNUCA policy: more fine grained, move a line to a closer bank on each hit, bank-based, same size
RHCA Result

RHCA achieves better performance than SRAM baseline and LHCA

PRAM is not promising for Cache
Outline

- Introduction and Motivation
- Methodology
- Level based Hybrid Cache Architecture
- Region based Hybrid Cache Architecture
- 3D Hybrid Cache Stacking
- Conclusions
3DHCA-configuration

- **3DHCA-C** (3D LHCA): 256KB L2 SRAM, 4M L3 eDRAM, 32M L4 PRAM
- **3DHCA-D**: 32M L2 fast, middle, slow region (3D RHCA)
  - Data in slow region can be moved to fast and middle regions
- **3DHCA-E**: 4M L2 fast+slow region, 32M L3 PRAM (LHCA+RHCA)
3DHCA-result

Normalized IPC

Normalized Power
Conclusion

- Hybrid cache architecture is promising to improve cache power-performance under same chip area/footprint

- RHCA and LHCA achieve better power-performance than SRAM-based design

- RHCA outperforms LHCA with minimal hardware support

- 3DHCA achieves better performance than LHCA and RHCA, while still maintains lower power than 2D SRAM baseline