SEESAW: 
Set Enhanced Superpage Aware caching

Mayank Parasar$^\Sigma$, Abhishek Bhattacharjee$^\Omega$, Tushar Krishna$^\Sigma$

$^\Sigma$School of Electrical and Computer Engineering
Georgia Institute of Technology
$^\Omega$Department of Computer Science
Rutgers University

mparasar3@gatech.edu
Outline

- Motivation
- SEESAW: Concept
- SEESAW: Micro-architecture
- Evaluation Methodology
- Results
- Conclusion
# L1 Cache Characteristics

<table>
<thead>
<tr>
<th>Feature</th>
<th>Ideal-Cache</th>
<th>VIPT-Cache</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fast lookup</td>
<td>![Checkmark]</td>
<td></td>
</tr>
<tr>
<td>High hit-rate</td>
<td>![Checkmark]</td>
<td></td>
</tr>
<tr>
<td>Energy Efficiency</td>
<td>![Checkmark]</td>
<td></td>
</tr>
</tbody>
</table>
Virtually Indexed Physically Tagged [VIPT] Cache

- VA
- VPN
- Page Offset
- TLB
- PA
- PPN
- Page Offset
- Data block
- v
- tag
- set
- way

Mayank Parasar, School of Electrical and Computer Engineering, Georgia Tech
Virtually Indexed Physically Tagged [VIPT] Cache

VIPT Caches necessitate: 
(set-index + block-offset) <= Page-offset
Impact of Associativity on Access Latency and Energy of cache

Cache Access Latency

Cache Access Energy

Mayank Parasar, School of Electrical and Computer Engineering, Georgia Tech

6/26/18
High Associativity hurts latency and energy without commensurately improving hit rate
Revisiting L1 Cache Characteristics for VIPT Cache

<table>
<thead>
<tr>
<th>Feature</th>
<th>Ideal-Cache</th>
<th>VIPT-Cache</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fast lookup</td>
<td>✔</td>
<td>✗</td>
</tr>
<tr>
<td>High hit-rate</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Energy Efficiency</td>
<td>✔</td>
<td>✗</td>
</tr>
</tbody>
</table>

Virtual memory!
Opportunity: Superpage

Is it possible to relax constrains of Traditional VIPT cache? **Yes**

How?

Offset-bits: 12
Baseline Page 4-KB

Offset-bits: 21
Super Page 2-MB

Offset-bits: 30
Super Page 1-GB

More page-offset bits for superpage!

HW and OS Support for Superpages in modern processors
Prevalence of superpages in modern OSes under memory fragmentation

Ran on 32-core; Sandybridge; 32 GB RAM

Memhog causes memory fragmentation; higher percentage indicates higher fragmentation
Outline

- Motivation
- SEESAW: Concept
- SEESAW: Micro-architecture
- Evaluation Methodology
- Results
- Conclusion
SEESAW: Concept

**Less-sets**
Less-associativity

**More-sets**
More-associativity

---

**Faster Energy-Efficient**

---

**Set:**
- **1:**
  - Way-1
  - Way-2
  - Way-3

- **2:**
  - Way-1
  - Way-2
  - Way-3

- **3:**
  - Way-1
  - Way-2
  - Way-3

---

**Super-page**

---

**Base-page**

---

**Dash:**
- **1:**
  - Way-1
  - Way-2
  - Way-3

- **2:**
  - Way-1
  - Way-2
  - Way-3

- **3:**
  - Way-1
  - Way-2
  - Way-3

---

**Dash:**
- **1:**
  - Way-1
  - Way-2
  - Way-3

- **2:**
  - Way-1
  - Way-2
  - Way-3

- **3:**
  - Way-1
  - Way-2
  - Way-3

---

**Faster Energy-Efficient**

---

**Set:**
- **1:**
  - Way-1
  - Way-2
  - Way-3

- **2:**
  - Way-1
  - Way-2
  - Way-3

- **3:**
  - Way-1
  - Way-2
  - Way-3

---

**Dash:**
- **1:**
  - Way-1
  - Way-2
  - Way-3

- **2:**
  - Way-1
  - Way-2
  - Way-3

- **3:**
  - Way-1
  - Way-2
  - Way-3

---

**Faster Energy-Efficient**

---

**Set:**
- **1:**
  - Way-1
  - Way-2
  - Way-3

- **2:**
  - Way-1
  - Way-2
  - Way-3

- **3:**
  - Way-1
  - Way-2
  - Way-3

---

**Dash:**
- **1:**
  - Way-1
  - Way-2
  - Way-3

- **2:**
  - Way-1
  - Way-2
  - Way-3

- **3:**
  - Way-1
  - Way-2
  - Way-3

---

**Faster Energy-Efficient**
Outline

- Motivation
- SEESAW: Concept
- SEESAW: Micro-architecture
- Evaluation Methodology
- Results
- Conclusion
SEESAW: Micro-architecture

Translation Filter Table (TFT)

Predicts whether page is superpage

Superpage offset

Decodes partition index from partition bit

VPN

Set index

block offset

Partition bit

Basepage Offset

VA

PA

PPN

Cache

VPN

V tag

Data block

VPN

V tag

Data block

V tag

Data block

V tag

Data block

Set index

Way-1

Way-2

Way-3

Way-4

Way-1

Way-2

Way-3

Way-4

Way-1

Way-2

Way-3

Way-4

Way-1

Way-2

Way-3

Way-4

Way-1

Way-2

Way-3

Way-4

Way-1

Way-2

Way-3

Way-4

Way-1

Way-2

Way-3

Way-4

Set-N

Partition decoder

Decodes partition index from partition bit

Partition

bit

Translation Filter Table (TFT)

Predicts whether page is superpage

Superpage offset

Decodes partition index from partition bit

VPN

Set index

block offset

Partition bit

Basepage Offset

VA

PA

PPN

Basepage Offset

Partition-0

Cache

Partition-1

Set-N

Set-1
SEESAW: Micro-architecture
SEESAW: Superpage access

<table>
<thead>
<tr>
<th>VA</th>
<th>VPN</th>
<th>Basepage Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>TLB</td>
<td>Translation Filter Table (TFT)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PA</td>
<td>PPN</td>
</tr>
</tbody>
</table>

Superpage offset
Partition bit
Set index
Block offset

Translation Filter Table (TFT)

Super Page

Partition decoder

Partition-0
Cache

HIT/MISS

Mayank Parasar, School of Electrical and Computer Engineering, Georgia Tech

6/26/18
SEESAW: Basepage access

Translation Filter Table (TFT)

VPN

Basepage Offset

VA

TLB

PA

PPN

Basepage Offset

Not a Super Page

HIT/MISS

Cache
SEESAW: TFT and Partition Decoder

**Translation Filter Table (TFT)**

- **TFT Lookup**
  - Direct mapped
  - False negative due to size

- **TFT Update**
  - VA misprediction
  - 2MB L1-TLB fill
  - 2MB L1-TLB Invalidation

**Partition Decoder**

- For 32kB Cache
- For 64kB Cache

---

Mayank Parasar, School of Electrical and Computer Engineering, Georgia Tech
SEESAW: Cache line insertion policy

Which partition should cache-line be inserted?
SEESAW: Cache line insertion policy

- **4way-8way**
  - Superpage miss: victim within the partition
  - Basepage miss: victim within the set

- **4way**
  - Uses LRU within the associated partition
  - Avoid installing the same line twice
  - Saves energy
SEESAW: System Level Optimization

- Cache coherence
  - Cache coherence lookups use physical address
  - Snoopy provide higher energy benefits over Directory based coherence

- Page table modifications
  - Superpage splintered into multiple basepages
  - Multiple basepages promoted to superpages
Outline

- Motivation
- SEESAW: Concept
- SEESAW: Micro-architecture
- Evaluation Methodology
- Results
- Conclusion
SEESAW: Simulated system

<table>
<thead>
<tr>
<th>CPU Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>Out-of-Order</td>
</tr>
<tr>
<td>In-order</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Memory System</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1 Cache</td>
</tr>
<tr>
<td>TLB (Atom)</td>
</tr>
<tr>
<td>TLB (Sbridge)</td>
</tr>
<tr>
<td>LLC</td>
</tr>
<tr>
<td>DRAM</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>System Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology</td>
</tr>
<tr>
<td>Frequency</td>
</tr>
<tr>
<td>Cores</td>
</tr>
<tr>
<td>Coherence</td>
</tr>
</tbody>
</table>
SEESAW: Workloads

- Spec
- Parsec
- Cloudsuite
  - Tunkrank
- Biobench
  - Mummer
  - Tiger
- MongoDB

- Server Workload
  - graph500
  - Nutch Hadoop
- Social-event web service
  - Olia
- Key value store
  - Redis
Outline

- Motivation
- SEESAW: Concept
- SEESAW: Micro-architecture
- Evaluation Methodology
- Results
- Conclusion
SEESAW observes 3-10% better runtime over baseline
SEESAW: Performance improvement

Out-of-order CPU
~10% performance improvement for 64kB cache in OoO CPUs
10-20% more energy savings over CPUs using baseline VIPT caches!
Approx. one-third of energy savings from coherence
SEESAW: TFT analysis and Way-Prediction

TFT Analysis

16-entry TFT drives miss-rate under 10%

SEESAW+WP shows symbiotic behavior
Outline

- Motivation
- SEESAW: Concept
- SEESAW: Micro-architecture
- Evaluation Methodology
- Results
- Conclusion
# Revisiting L1 Cache Characteristic

<table>
<thead>
<tr>
<th>Feature</th>
<th>Ideal-Cache</th>
<th>VIPT-Cache</th>
<th>SEESAW Cache</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fast lookup</td>
<td>✔</td>
<td>✗</td>
<td>✔</td>
</tr>
<tr>
<td>High hit-rate</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Energy Efficiency</td>
<td>✔</td>
<td>✗</td>
<td>✔</td>
</tr>
</tbody>
</table>
SEESAW: Conclusion

- L1 caches are optimized for latency
  - VIPT imposes indirect restriction on number of sets in a L1 cache, increasing associativity
  - There is non-linear relation between associativity and access latency/energy of the L1 cache

- Superpages are often used in modern OSes
  - SEESAW provides low-associative access to superpages, providing both latency and energy benefits
  - Up to 10% performance improvement and 20% energy reduction in modern workloads

- SEESAW has extremely low-overhead and is readily implementable