L1 Data Cache Decomposition for Energy Efficiency

Michael Huang, Joe Renau, Seung-Moon Yoo, Josep Torrellas
University of Illinois at Urbana-Champaign
http://iacoma.cs.uiuc.edu/flexram
Objective

- Reduce L1 data cache energy consumption
- No performance degradation

- Partition the cache in multiple ways
- Specialization for stack accesses
Outline

- L1 D-Cache decomposition
- Specialized Stack Cache
- Pseudo Set-Associative Cache
- Simulation Environment
- Evaluation
- Conclusions
L1 D-Cache Decomposition

- A Specialized Stack Cache (SSC)
- A Pseudo Set-Associative Cache (PSAC)
Selection

- Selection done in decode stage to speed up
  - Based on instruction address and opcode
- 2Kbit table to predict the PSAC way

Diagram:

- Opcode
- Address
- PSAC
- SSC
- L1 Cache
- Pseudo Set-Associative Cache
- Specialized Stack Cache
- From Processor
- To L2 Cache
Stack Cache

- Small, direct-mapped cache
- Virtually tagged
- Software optimizations:
  - Very important to reduce stack cache size
  - Avoid trashing: allocate large structs in heap
  - Easy to implement
SSC: Specialized Stack Cache

Pointers to reduce traffic:

- **TOS**: reduce number write-backs
- **SRB (safe-region-bottom)**: reduce unnecessary line-fills for write miss
  - Region between TOS & SRB is “safe” (missing lines are non initialized)

Infrequent access
Pseudo Set-Associative Cache

- Partition the cache in 4 ways

- Evaluated activation policies: Sequential, FallBackReg, Phased Cache, FallBackPha, PredictPha
Sequential (Calder ‘96)
Fallback-regular (Inoue ‘99)
Phased Cache (Hasegawa ‘95)
Fallback-phased (ours)

- Emphasis in energy reduction

Diagram:

- Cycle 1
- Cycle 2
- Cycle 3
Predictive Phased (ours)

- Emphasis in performance
Simulation Environment

Baseline configuration:

- Processor: 1GHz R10000 like
- L1: 32 KB 2-way
- L2: 512KB 8-way phased cache
- Memory: 1 Rambus Channel
- Energy model: extended CACTI
- Energy is for data memory hierarchy only
Applications

- **Multimedia**
  - Mp3dec: MP3 decoder
  - Mp3enc: MP3 encoder
  - Gzip: Data compression
- **SPECint**
  - Crafty: Chess game
  - MCF: Traffic model
  - Bsom: data mining
- **Scientific**
  - Blast: protein matching
  - Treeadd: Olden tree search
Adding a Stack Cache

For the same size the Specialized Stack Cache is always better
Pseudo Set-Associative Cache

PredictPha has the best delay and energy-delay product
PSAC: 2-way vs. 4-way

For $E^*D$, 4-way PSAC is better than 2-way
Pseudo Set-Associative + Specialized Stack Cache

Combining PSAC and SSC reduces E*D by 44% on average
Area Constrained: small PSAC+SSC

SSC + small PSAC delivers cost effective $E^*D$ design
Energy Breakdown

[Bar chart showing energy breakdown for different benchmarks and models, normalized to baseline. The chart includes categories for BLAST, MCF, and MP3D, with subcategories for Baseline, 4-way PSAC, SSC512B, and Comb.](chart)
Conclusions

- Stack cache: important for energy-efficiency
- SW optimization required for stack caches
- Effective Specialized Stack Cache extensions
- Pseudo Set-Associative Cache:
  - 4-way more effective than 2-way
  - Predictive Phased PSAC has the lowest E*D
- Effective to combine PASC and SSC
  - E*D reduced by 44% on average
Backup Slides
Cache Energy

![Graph showing cache energy vs cache size for 4-way, 2-way, and 1-way caches. The x-axis represents cache size in KiB (4K, 8K, 16K, 32K, 64K), and the y-axis represents energy in pJ (picojoules). The graph shows a clear trend of increasing energy with cache size for all configurations.]
Extended CACTI

- New sense amplifier
  - 15% bit-line swing for reads
- Full bit-line swing for writes
- Different energy for reads, writes, line-fills, and write backs
- Multiple optimization parameters
SSC Energy Overhead

- Small energy consumption required to use TOS and SRB
- Registers updated at function call and return
- Registers check on cache miss
Overview