Exploring Latency-Power Tradeoffs in Deep Nonvolatile Memory Hierarchies

Doe Hyun Yoon, Tobin Gonzalez, Parthasarathy Ranganathan, and Robert S. Schreiber

Intelligent Infrastructure Lab (IIL), Hewlett-Packard Labs

Computing Frontiers – May 15 2012
Deeper and Deeper Memory Hierarchy

Cache memory for reducing average memory latency
- Initially, a small L1 cache
- Then, L2, L3, ...
- DRAM cache for NVM main memory
- SLC / MLC NVM memory

Is a DEEP hierarchy power efficient?

We develop a model to explore latency-power tradeoffs

Flattened hierarchies with a large NVM cache are power efficient
1. Latency-Power Model

- Configuration:
  - # levels
  - Size of each level
  - Array type: SRAM, DRAM, ...

Change config for exhaustive search

Tech parameters

Performance & Power model

Optimizer

App characteristics

Pareto-optimal frontier

Estimate power and performance of an application
Exhaustive search to find Pareto-optimal frontier
Application Characteristics

Program

Binary Instrumentation

Load/store instr

32kB cache

Miss

Traffic below a 32kB cache

32kB cache

Miss

Traffic below a combined 64kB cache

64kB cache

Miss

Traffic below a combined 128kB cache

An exclusive hierarchy

Profile MPKI vs. cache sizes

MPKI

Proc  128kB  1MB  8MB  64MB  512MB  4GB

1000

100

10

1
Example) SPEC CPU 2006

Graph showing MPKI (log scale) vs. Cache Size for different benchmarks:
- mcf
- bzip2
- hmmer
- lbm
- astar
- milc
- omnetpp

The graph displays the performance trend of these benchmarks with varying cache sizes from Proc to 4GB.
Example) MiniFE, MiniMD, and Graph 500

MPKI (log scale)

Cache Size

MiniFE

MiniMD

Graph 500

Proc

128kB

1MB

8MB

64MB

512MB

4GB
Technology Parameters

Latency, static power, and energy per access of different caches

SRAM and DRAM $\rightarrow$ CACTI 6

Phase-Change RAM (PCRAM) $\rightarrow$ NVSim
Performance Model

Use AMAT (Average Memory Access Time)

\[ \text{AMAT} = L(1) + \sum_{i=0}^{N-1} \left( \frac{M(i)}{M(0)} \times L(i+1) \right) \]

Total N cache levels

L(i): Latency of cache level i (tech parameter)

M(i): MPKI of cache level i (app characteristic)
Power Model

\[ P = P_{\text{static}} + P_{\text{dyn}} \]

\[ P_{\text{static}} = \sum P_s(i) \]

\[ E_{\text{dyn}} = M(0) \times E(1) + \sum [ M(i) \times (E_r(i+1) + E_w(i)) ] \]

\[ P_{\text{dyn}} = \frac{E_{\text{dyn}}}{[(1000 - M(0)) \times T_{\text{cyc}} + \text{AMAT} \times M(0)]} \]

\( E(i) \): Energy per access (tech parameter)
\( T_{\text{cyc}} \): cycle time (1 ns in our study)
The Model, Again

Configuration:
- # levels
- Size of each level
- Array type: SRAM, DRAM, ...

Change config for exhaustive search

Pareto-optimal frontier

Technology parameters

CACTI NVSim

Application

PIN cache model

Performance & Power model

Optimizer

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Application

PIN cache model

Technology parameters

CACTI NVSim

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power

AMAT

Change config

Performance & Power model

Optimizer

Pareto-optimal frontier

Power
Exhaustive Search Example

Power (mW) (log scale)

AMAT (ns)
Pareto-Optimal Frontier

Power (mW)

Pareto-Optimal Frontier

AMAT (ns)

Power (mW) (log scale)
Optimum Configurations

Power (mW) vs. AMAT (ns)

- Minimum latency configuration
- Power-efficient configuration

Pareto-Optimal Frontier

Minimum power configuration
2. Depth of a Cache Hierarchy

Use the latency-power model

SRAM caches, ranging from 32kB to 2GB
2 to 6-level hierarchies
Graph 500

- Power (mW) (log scale)
  - $10^2$
  - $10^3$
  - $10^4$
  - $10^5$
  - $10^6$

- AMAT (ns)
  - 8
  - 9
  - 10
  - 11
  - 12
  - 13
  - 14

- 2-level is better
- 3-level is better
- 4-level
- 5-level
- 6-level

Log scale
MiniFE

AMAT reduces only when cache is bigger than 1.5GB

Deep hierarchies are not effective
Lessons Learned

Cache hierarchies deeper than 3 levels
→ Increase AMAT
→ Use more power

Large SRAM caches
→ Increase static power a lot
→ Only a small improvement in AMAT
3. Cache Hierarchies with NVM

Use the latency-power model

Replace DRAM main memory with NVM main memory

Caches with heterogeneous technologies

• SRAM caches, ranging 32kB to 32MB
• DRAM caches, ranging 4MB to 64MB
• PCRAM caches, ranging 16MB to 1GB
Graph 500

- **Power (mW)**
  - 10^2
  - 10^3
  - 10^4

- **AMAT (ns)**
  - 0
  - 20
  - 40
  - 60
  - 80
  - 100
  - 120

**Graph Description**
- **SRAM cache + DRAM main memory**
- **SRAM cache + PCRAM main memory**
- **SRAM/DRAM cache + PCRAM main memory**
- **SRAM/DRAM/PCRAM cache + PCRAM main memory**

- **Penalty due to slow NVM main memory**
- **Mitigating NVM penalty with DRAM caches**
- **Further reducing power with low-leakage NVM caches**
Power (mW) (log scale)

- SRAM cache + DRAM main memory
- SRAM cache + PCRAM main memory
- SRAM/DRAM cache + PCRAM main memory
- SRAM/DRAM/PCRAM cache + PCRAM main memory

MCF

NVM cache

DRAM cache

NVM main memory
4. Streaming Patterns

Working set is simply larger than the cache

LRU policy thrashes data

Large caches can’t improve AMAT but waste power unless the cache is larger than the working set
Even the largest NVM cache can’t hold the working set.
Alternative Replacement Policy

Inserted lines are never replaced

Mimicking more recent techniques such as DIP [Qureshi+ ISCA’07] or RRIP [Jaleel+ ISCA’10]

Can preserve a fraction of working set in the cache
MiniFE: MPKI vs. Cache Size

Caches smaller than 2GB can reduce MPKI.
MiniFE with the Alternative Policy

1GB NVM cache can reduce power and AMAT.
**Conclusions**

Developed a latency-power model

Deep hierarchies are less power efficient than flat hierarchies

Large NVM caches can be power efficient

Advanced insertion/replacement policy should be combined to handle streaming patterns.
Limitations

Assumed a blocking in-order core
  Ignored memory level parallelism, prefetching, ...

Write endurance in NVM caches

Aggressive power control in SRAM caches
Acknowledgement & Disclaimer

This material is based upon work supported by the Department of Energy under Award Number DE-SC0005026.

See http://www.hpl.hp.com/DoE-Disclaimer.html for additional information
Exploring Latency-Power Tradeoffs in Deep Nonvolatile Memory Hierarchies

Doe Hyun Yoon, Tobin Gonzalez, Parthasarathy Ranganathan, and Robert S. Schreiber

Intelligent Infrastructure Lab (IIL), Hewlett-Packard Labs