

Power Aware Page Allocation

Alvin R. Lebeck, Xiaobo Fan, Heng Zeng, Carla Ellis



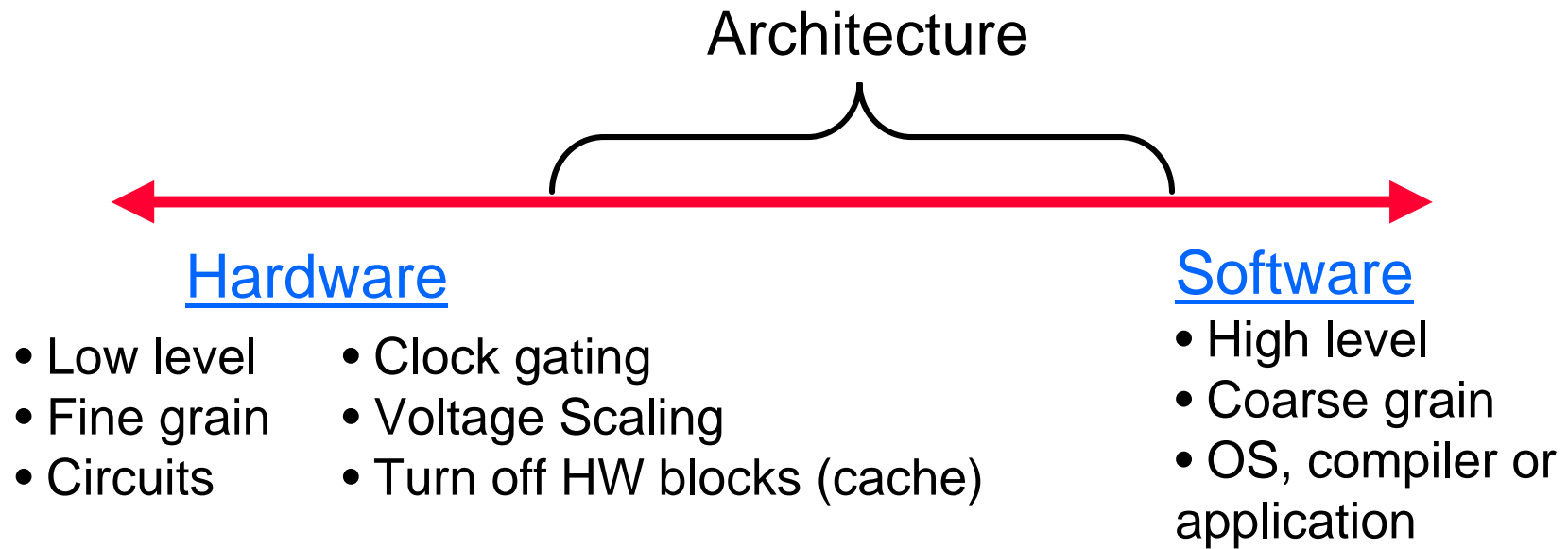
Milly Watt

Department of Computer Science
Duke University
alvy@cs.duke.edu
<http://www.cs.duke.edu/~alvy>

Power Aware System Design

- ✍ Traditional system design targets increased performance
- ✍ Post-PC world has many battery powered devices
 - Energy optimization becomes increasingly important
- ✍ Today, little understanding of energy ramifications of application/system design
- ✍ **Goal: Power Aware System Design**
 - Need: Revisit all aspects of system design [SIGOPS EW '00]

The Energy Saving Spectrum



 Re-examine interactions between HW and SW, particularly the Operating System

The Unturned Stone

Traditional OS resources

- Processor, Memory, Disk, Network

Power Studies

 Disk spin down [[Baker, Douglass, Li](#)]

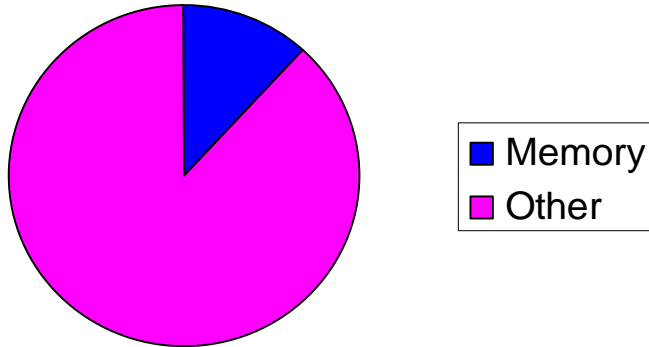
 Processor voltage/clock [[Weiser, Pering, Lorch, Grunwald](#)]

 Network interface [[Stemm, Kravets, Imielinski](#)]

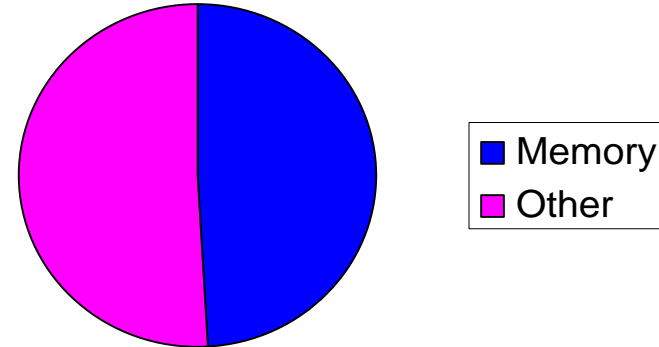
 **Where is main memory?**

Memory Power Budget

Laptop Power Budget
9 Watt Processor



Handheld Power Budget
1 Watt Processor



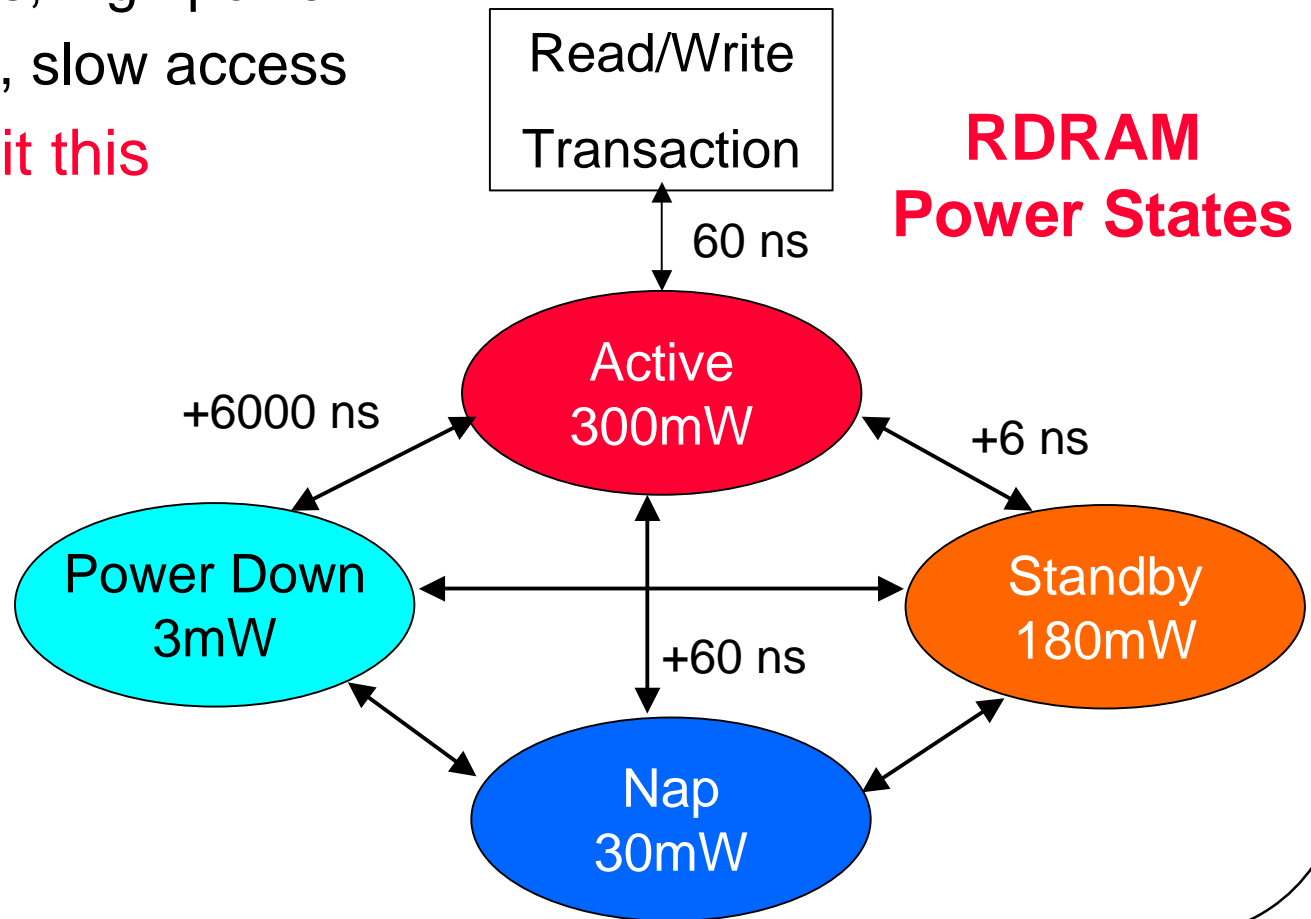
- ✍ Laptop: memory is small percentage of total power budget
- ✍ Handheld: low power processor, memory is more important

Opportunity: Power Aware DRAM

✍ Multiple power states

- Fast access, high power
- Low power, slow access

✍ How to exploit this opportunity?



Outline

✍ Motivation

✍ The opportunity

✍ Hardware power management policies

✍ Operating system page allocation policies

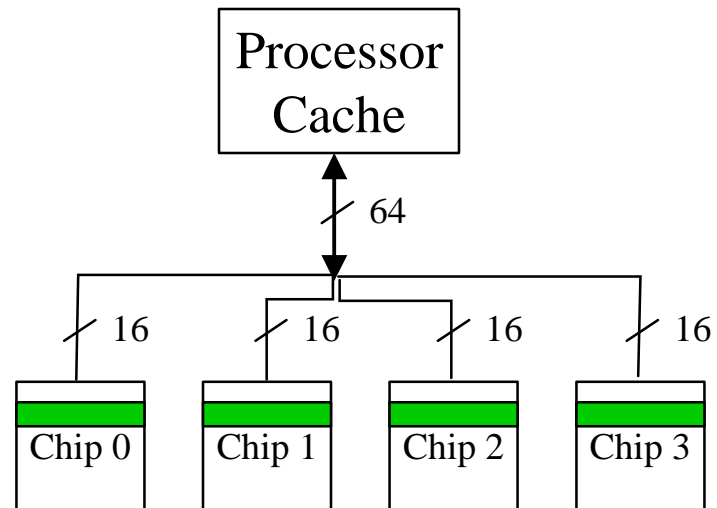
✍ Results

- Simple HW does very well
- **Need both HW and OS support to maximize benefits**
- New HW policy results

✍ Conclusion

Conventional Main Memory Design

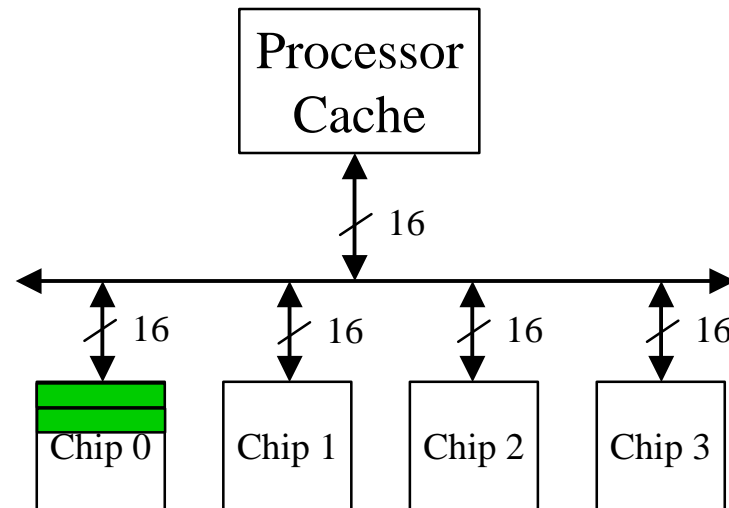
Cache Block



- ✍ Multiple DRAM chips provide high bandwidth per access
 - Wide bus to processor
 - Few internal banks
- ✍ Energy implication: **Must activate all those chips to perform access at high bandwidth**

RAMBUS Main Memory Design

Cache Block



- ✍ Single RDRAM chip provides high bandwidth per access
 - Novel signaling scheme transfers multiple bits on one wire
 - Many internal banks: many requests to one chip
- ✍ Energy implication: **Must activate only one chip to perform access at same high bandwidth as conventional design**
 - **Cluster accesses to already powered up chip**

Exploiting the opportunity

 Interaction between state transitions and access locality

 Q1: **How do we manage the power state transitions?**

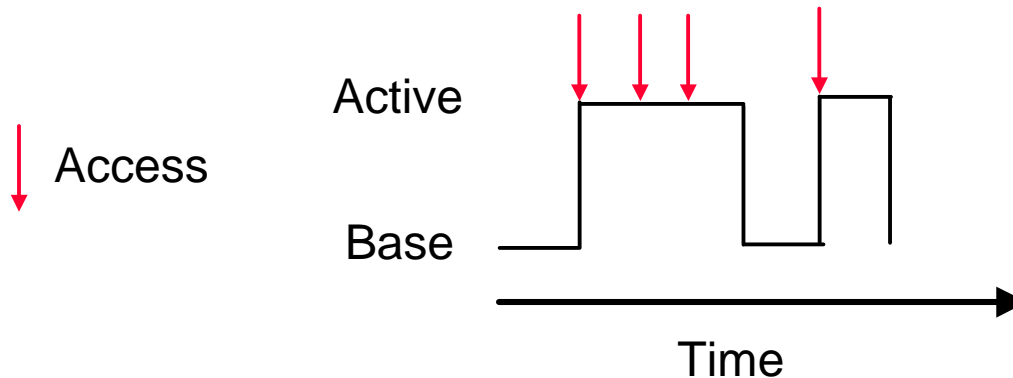
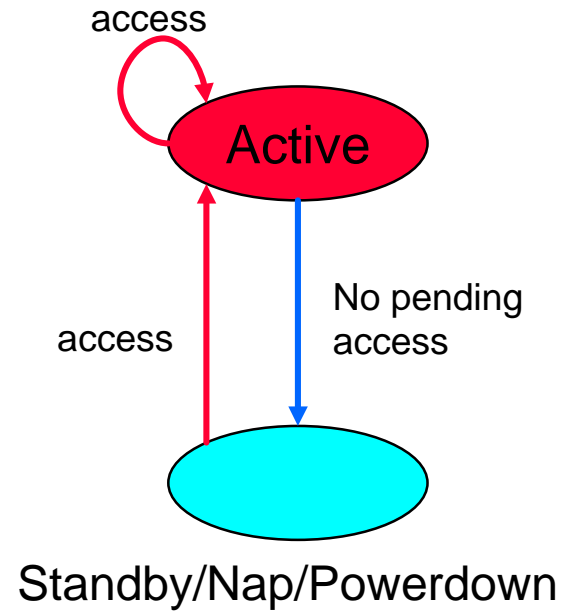
- Memory controller policies
- Quantify benefits of power states

 Q2: **What role does software have?**

- Does allocation of data/text to memory affect energy?

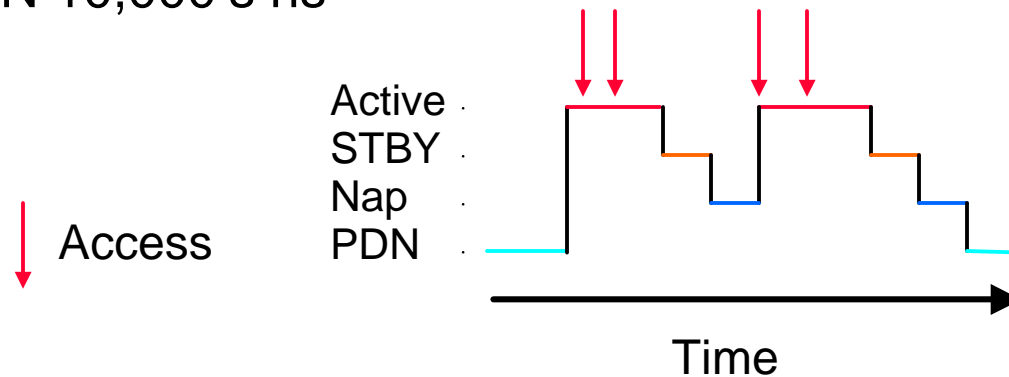
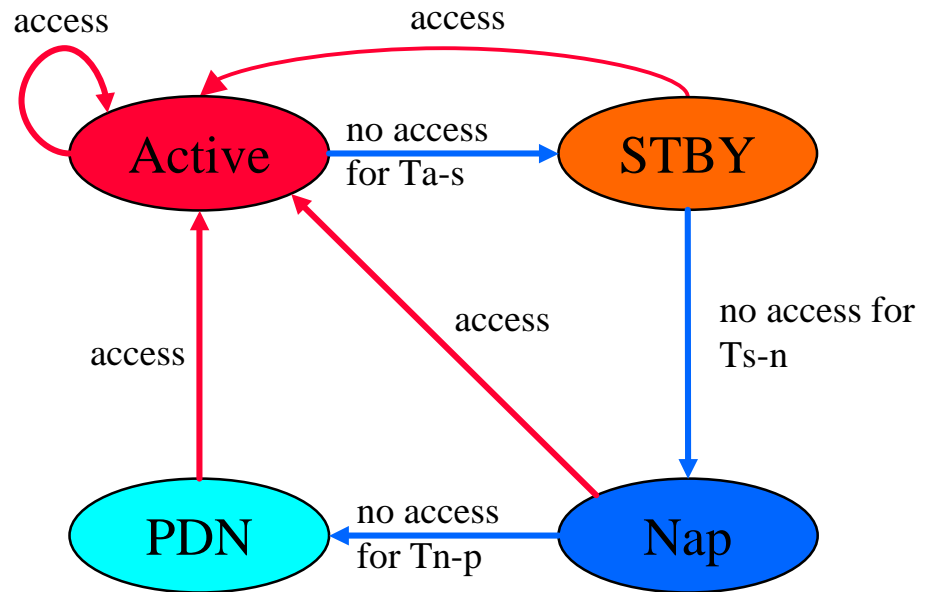
Dual-state (Static) HW Power State Policies

- ✍ All chips in same base state
- ✍ Individual chip Active while pending requests
- ✍ Return to base power state if no pending access



Quad-state (Dynamic) HW Policies

- ✍ Downgrade state if no access for **threshold** time
- ✍ Independent transitions based on access pattern to each chip
- ✍ Analysis
 - Active/Stby to nap 100's of ns
 - Nap to PDN 10,000's ns



Exploiting PDRAM Power States

Hardware power management policies

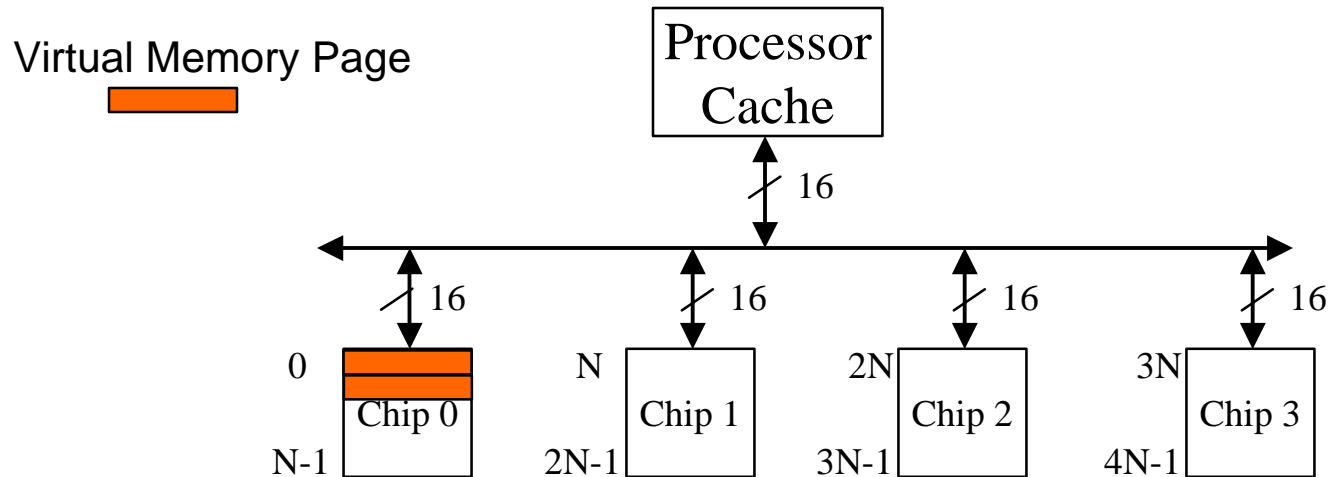
- Exploit locality to reduce energy consumption
- Dual-state model
- Quad-state model

Want to increase DRAM chip-level locality

Performance penalty with conventional DRAM

- Can ignore BW tradeoff w/ RDRAM properties

Page Allocation and PDRAM



- ✍ Physical address determines which chip is accessed
- ✍ Assume non-interleaved memory
 - Addresses 0 to N-1 to chip 0, N to 2N-1 to chip 1, etc.
- ✍ Entire virtual memory page in one chip
- ✍ Virtual memory page allocation influences chip-level locality

Page Allocation Policies

Random Allocation

- Pages spread across chips

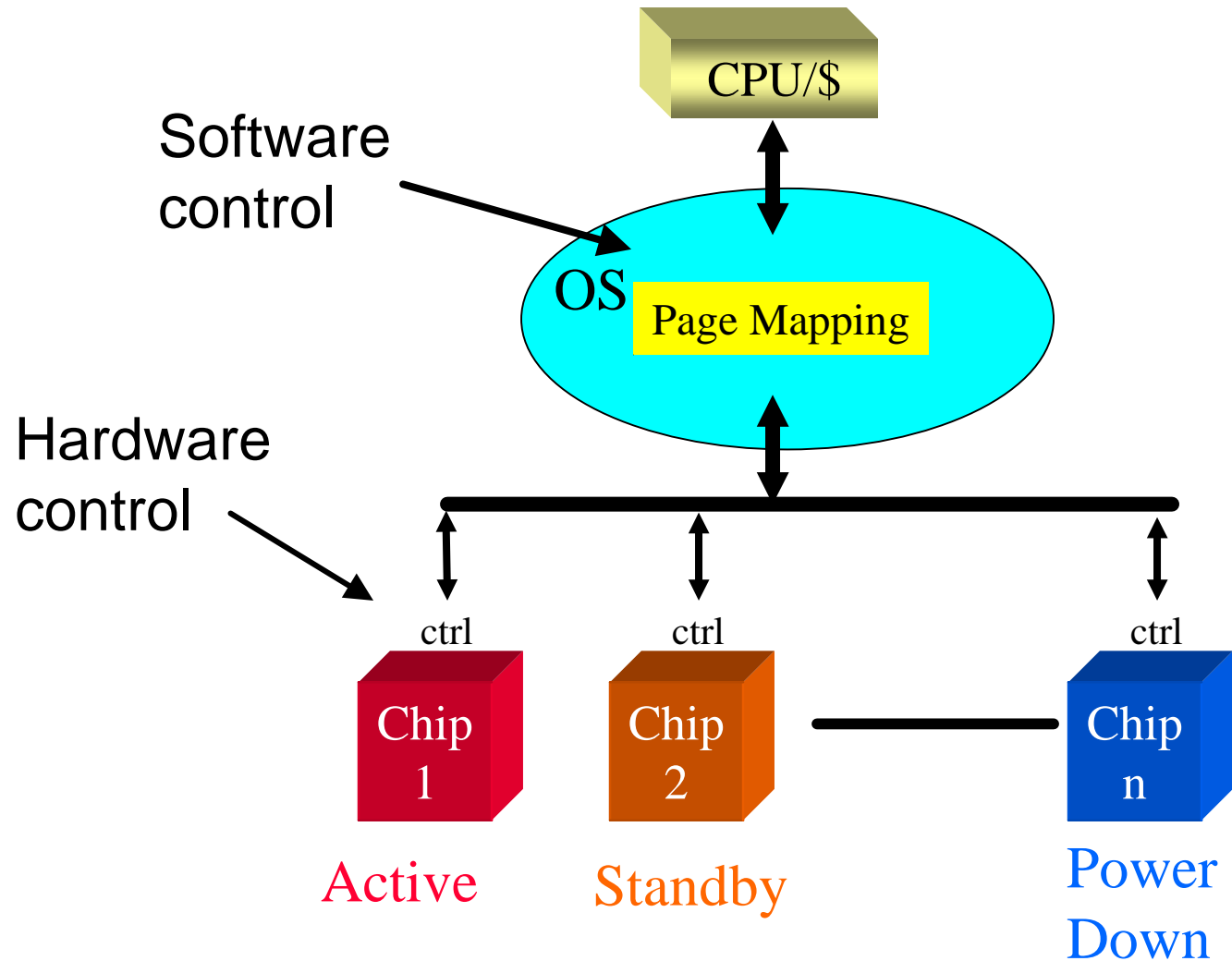
Sequential First-Touch Allocation

- Consolidate pages into minimal number of chips
- One shot

Frequency-based Allocation

- Preliminary results in paper

Two Dimensions to Control Energy



Outline

 Motivation

 The opportunity: Power Aware DRAM

 Hardware power management policies

 Operating system page allocation policies

 **Results**

- Methodology
- Hardware and Software Policies
- New hardware policies

 Conclusion

Methodology

✍ Metric: Energy*Delay Product

- Avoid very slow solutions

✍ Energy Consumption (DRAM only)

- Processor & Cache affect runtime

✍ 8KB page size

✍ L1/L2 non-blocking caches

- 256KB direct-mapped L2
- Qualitatively similar to 4-way

✍ Average power for transition from lower to higher state

Methodology Continued

Trace-Driven Simulation

- NT applications from U. of Washington Etch group
- Simple processor
- Eight 32Mb chips, total 32MB, non-interleaved
- See paper for full results

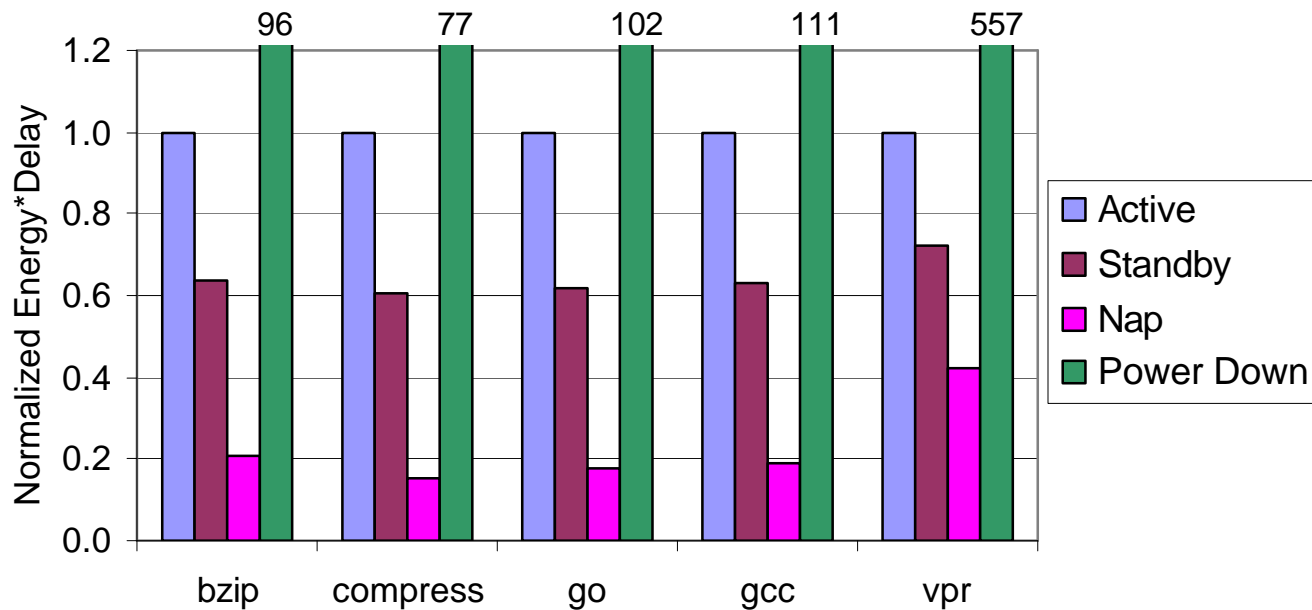
Execution-Driven Simulation

- SPEC benchmarks (subset of integer)
- SimpleScalar w/ **detailed RDRAM timing and power models**
- Eight 256Mb chips, total 256MB, non-interleaved

The Design Space

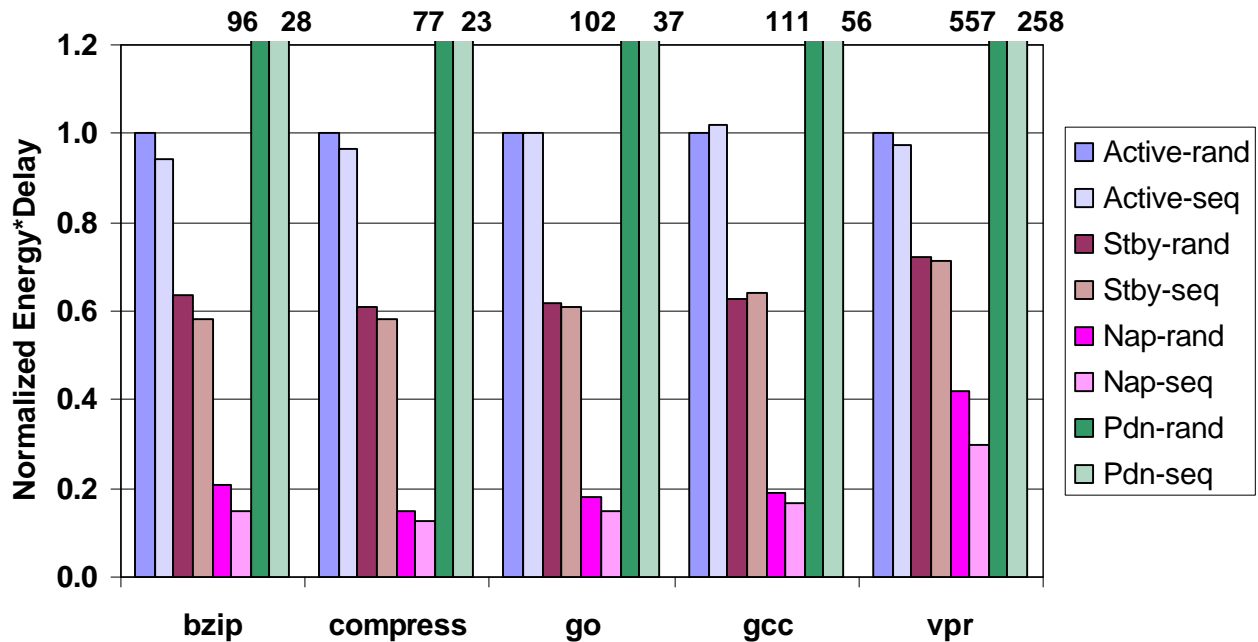
	Random Allocation	Sequential Allocation
Dual-state Hardware (static)	1 Simple HW	2 Can the OS help?
Quad-state Hardware (dynamic)	3 Sophisticated HW	4 Cooperative HW & SW

Dual-state + Random Allocation



- ✍ All chips use same base state
- ✍ Nap is best 60% to 85% reduction in $E \cdot D$ over full power
- ✍ Simple HW provides good improvement

Benefits of Sequential Allocation



✍ 10% to 30% relative improvement for dual-state nap

✍ Some benefits due to cache effects

The Design Space

Random
Allocation

Sequential
Allocation

Dual-state
Hardware
(static)

Nap is best
60%-85%
improvement

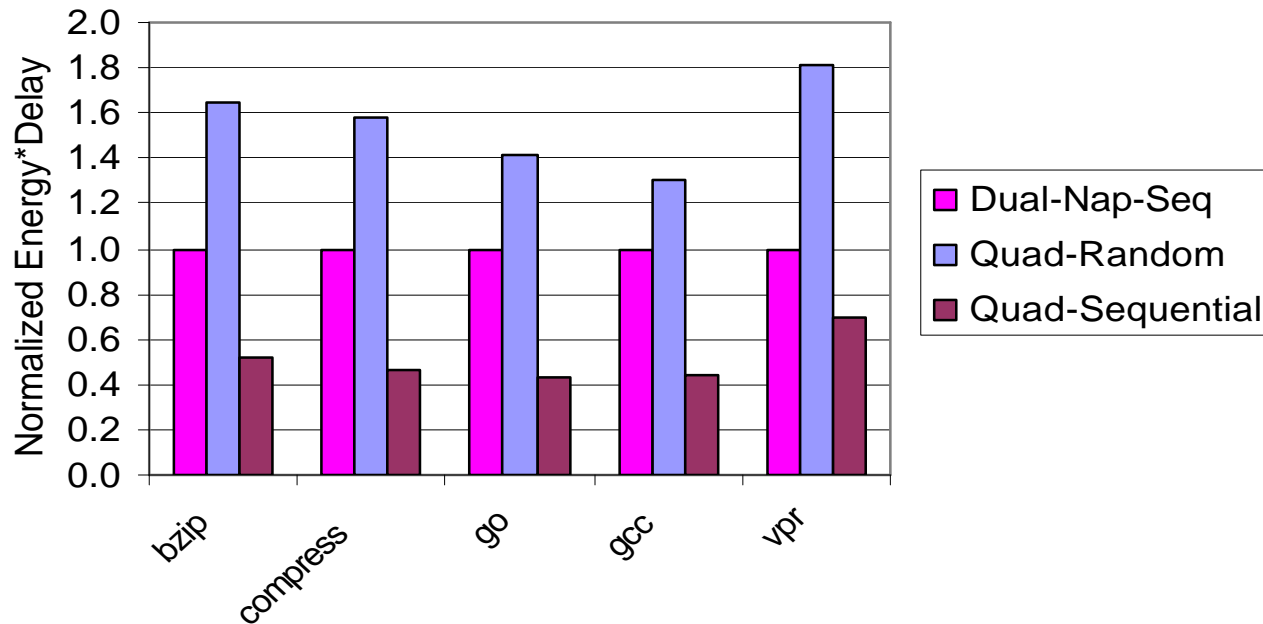
10% to 30%
improvement for
nap. Base for
future results

Quad-state
Hardware
(dynamic)

What about
smarter HW?

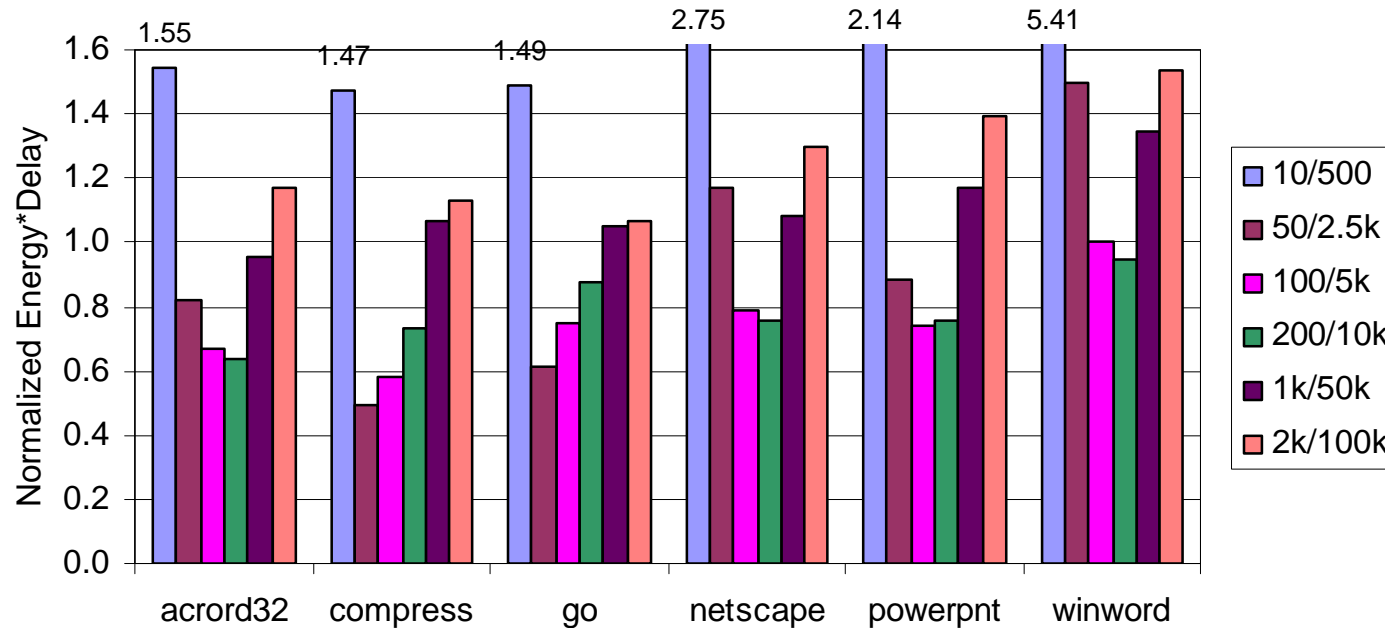
Smart HW and
OS support?

Quad-state Hardware



- ✍ Base: Dual-state Nap Sequential Allocation
- ✍ Thresholds: 0ns A->S; 750ns S->N; 375,000 N->P
- ✍ Quad-state + Sequential 30% to 55% additional improvement over dual-state nap sequential
- ✍ Sophisticated HW must get thresholds correct

Threshold Sensitivity (NT Traces)



✍ Quad-state seq. vs. Dual-state nap seq.

✍ Bars: Active to Nap / Nap to PDN threshold values

- Best thresholds match general results of analysis

✍ 6% to 50% improvement over best dual-state

The Design Space

Random
Allocation

Sequential
Allocation

Dual-state
Hardware
(static)

Nap is best
dual-state policy
60%-85%
improvement

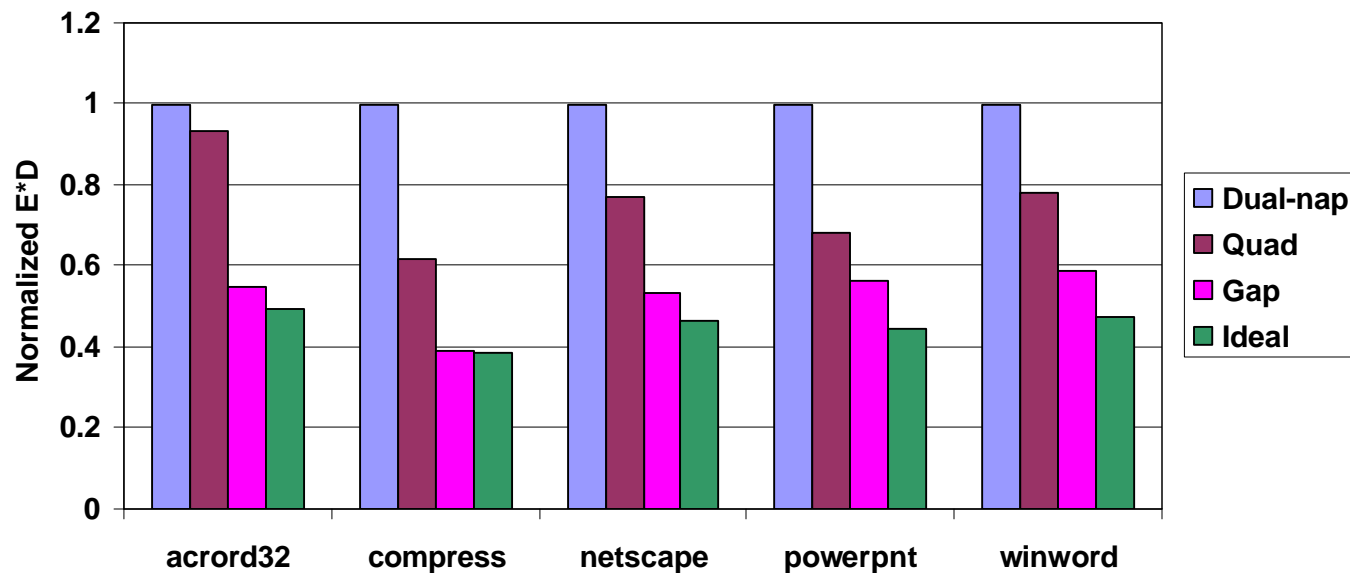
Additional
10% to 30%
improvement
over Nap

Quad-state
Hardware
(dynamic)

Thresholds not
obvious,
Could be equal
to dual-state

Best Approach:
6% to 55% over
dual-nap-seq,
80% to 99% over
all active.

Locality Aware Memory Controller Policy



✍ Sequential Page Allocation, NT Traces

✍ Ideal: Offline, Delay = all Active, minimize Power

✍ **Gap: History-based prediction for next access time**

- If gap > benefit boundary, immediately transition
- Stable for random allocation

Conclusion

- ✍ Energy is an important metric for Post-PC computing
- ✍ Memory is unexplored, but important
- ✍ New DRAM technologies provide opportunity
 - Multiple power states
- ✍ **Cooperative hardware / software solution is best**
- ✍ **New memory controller policies even better**