Memory: The Unturned Stone

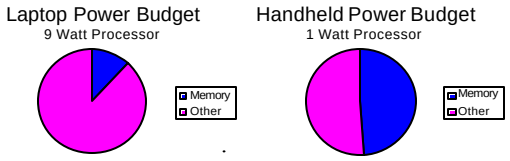
Previous Architecture/OS Energy Studies:

- Disk spindown policies [Douglis, Krishnan, Helmbold, Li]
- Processor voltage and clock scaling [Weiser, Pering, Lorch, Farkas et al]
- Network Interface [Stemm, Kravets]
- Mems-based storage [Nagle et al]
- Application-aware adaptation & API [Flinn&Satya]

- But where is main memory management?

Power Aware Page Allocation [ASPLOS00]

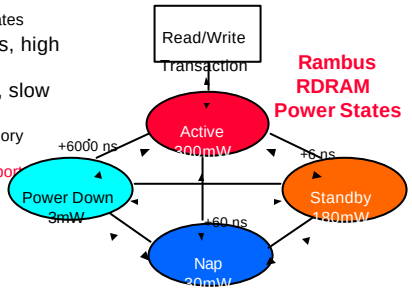
Memory System Power Consumption



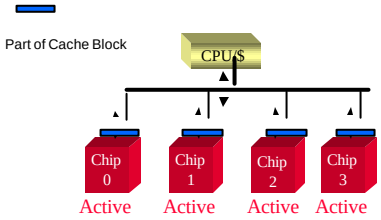
- Laptop: memory is small percentage of total power budget
- Handheld: low power processor, memory is more important

Opportunity: Power Aware DRAM

- Multiple power states
  - Fast access, high power
  - Low power, slow access
- New take on memory hierarchy
- How to exploit opport

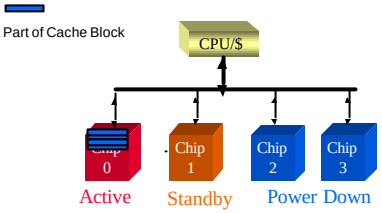


Conventional Main Memory Design



- 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

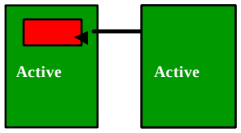
RAMBUS RDRAM Main Memory Design



- 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: Activate only one chip to perform access at same high bandwidth as conventional design

RDRAM as a Memory Hierarchy

- Each chip can be independently put into appropriate power mode
- Number of chips at each "level" of the hierarchy can vary dynamically.



Policy choices

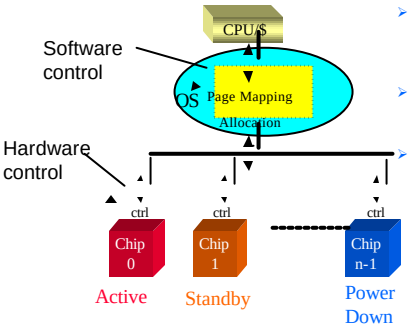
- initial page placement in an "appropriate" chip
- dynamic movement of page from one chip to another
- transitioning of power state of chip containing page

Exploiting the Opportunity

Interaction between power state model and access locality

- How to manage the power state transitions?
  - Memory controller policies
  - Quantify benefits of power states
- What role does software have?
  - Energy impact of allocation of data/text to memory.

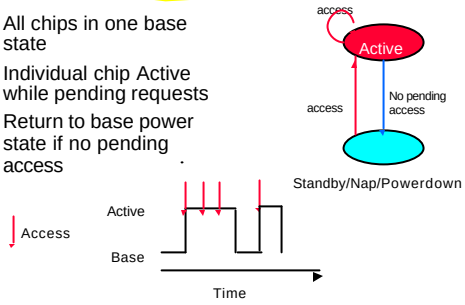
Power-Aware DRAM Main Memory Design



- Properties of RDRAM allow us to access and control each chip individually
- 2 dimensions to affect energy policy: HW controller / OS
- Energy strategy:
  - Cluster accesses to already powered up chips
  - Interaction between power state transitions and data locality

Dual-state HW Power State Policies

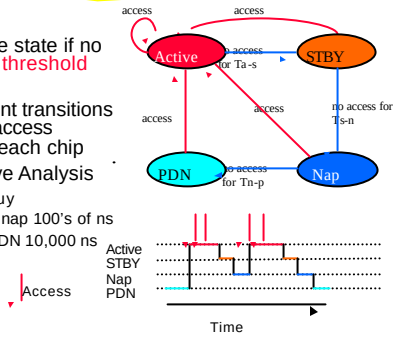
- All chips in one base state
- Individual chip Active while pending requests
- Return to base power state if no pending access



18

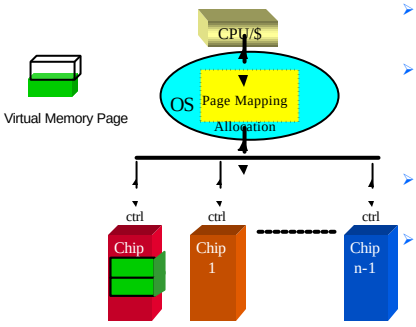
Quad-state HW Policies

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



19

Page Allocation and Power-Aware DRAM



- 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

20

Page Allocation Polices

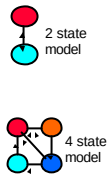
Virtual to Physical Page Mapping

- Random Allocation – baseline policy
  - Pages spread across chips
- Sequential First-Touch Allocation
  - Consolidate pages into minimal number of chips
  - One shot
- Frequency-based Allocation
  - First-touch not always best
  - Allow (limited) movement after first-touch

21

The Design Space

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



22

Methodology

- Metric: Energy*Delay Product
  - Avoid very slow solutions
- Energy Consumption (DRAM only)
  - Processor & Cache affect runtime
  - Runtime doesn't change much in most cases
- 8KB page size
- L1/L2 non-blocking caches
  - 256KB direct-mapped L2
  - Qualitatively similar to 4-way associative L2
- Average power for transition from lower to higher state
- Trace-driven and Execution-driven simulators

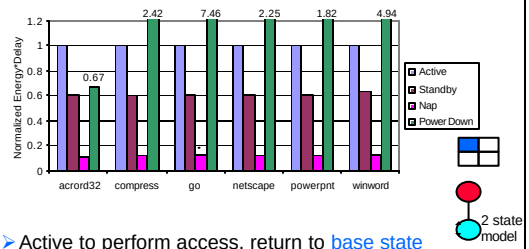
23

Methodology Continued

- Trace-Driven Simulation
  - Windows NT personal productivity applications (Etch at Washington)
  - Simplified processor and memory model
  - Eight outstanding cache misses
  - Eight 32Mb chips, total 32MB, non-interleaved
- Execution-Driven Simulation
  - SPEC benchmarks (subset of integer)
  - SimpleScalar w/ detailed RDRAM timing and power models
  - Sixteen outstanding cache misses
  - Eight 256Mb chips, total 256MB, non-interleaved

24

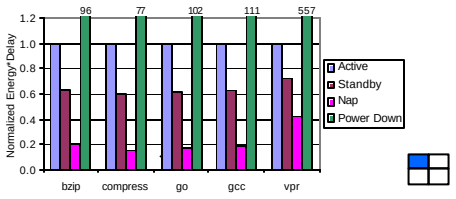
Dual-state + Random Allocation (NT Traces)



- Active to perform access, return to base state
- Nap is best ~85% reduction in E*D over full power
- Little change in run-time, most gains in energy/power

25

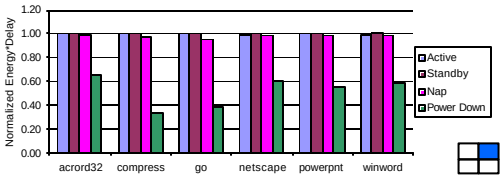
Dual-state + Random Allocation (SPEC)



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

26

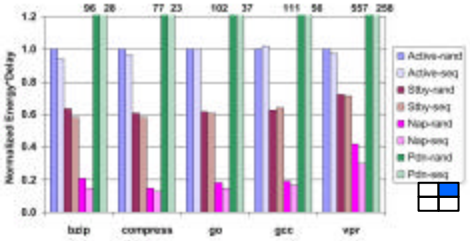
Benefits of Sequential Allocation (NT Traces)



- Sequential normalized to random for same dual-state policy
- Very little benefit for most modes
  - Helps PowerDown, which is still really bad

27

Benefits of Sequential Allocation (SPEC)

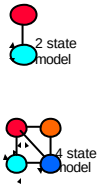


- 10% to 30% additional improvement for dual-state nap
- Some benefits due to cache effects

29

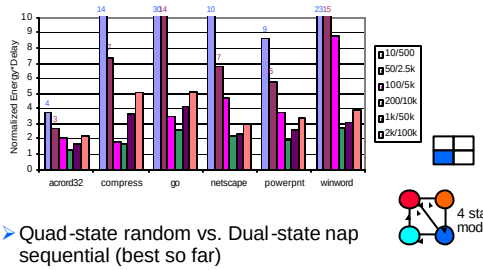
Results (Energy*Delay product)

	Random Allocation	Sequential Allocation
Dual-state Hardware	Nap is best 60%-85% improvement	10% to 30% improvement for nap. Base for future results
Quad-state Hardware	What about smarter HW?	Smart HW and OS support?



30

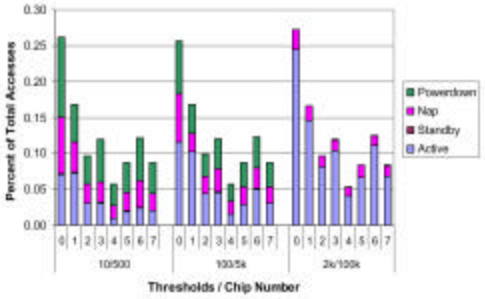
Quad-state HW + Random Allocation (NT)



- Quad-state random vs. Dual-state nap sequential (best so far)
- With these thresholds, sophisticated HW is not enough.

31

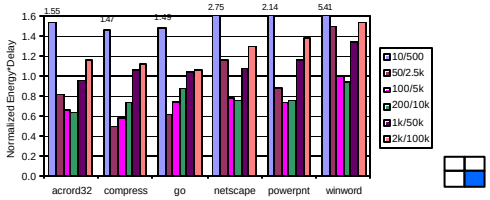
Access Distribution: Netscape



- Quad-state Random with different thresholds

32

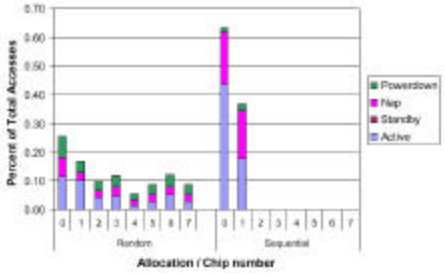
Quad-state HW + Sequential Allocation (NT)



- Quad-state vs. Dual-state nap sequential
- Additional 6% to 50% improvement over best dual-state

33

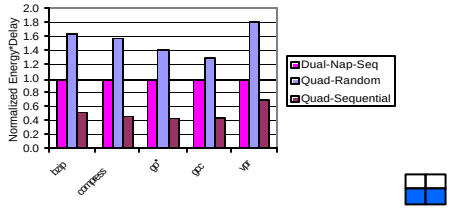
Allocation and Access Distribution: Netscape



- Based on Quad-state threshold 100/5K

34

Quad-state HW (SPEC)



- 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
- HW / SW Cooperation is important

35

Results (Energy*Delay product)

	Random Allocation	Sequential Allocation
Dual-state Hardware	Nap is best dual-state policy 60%-85% .	Additional 10% to 30% over Nap
Quad-state Hardware	Improvement not obvious, Could be equal to dual-state	Best Approach: 5% to 55% over dual-nap-seq, 30% to 99% over all active.

2 state model

4 state model

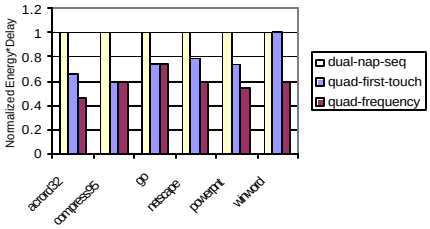
36

Better Page Allocation Policies?

- Intuitively, first-touch will not always be best
- Allow movement after first-touch as "corrections"
- Frequency-based allocation
- Preliminary results
  - Offline algorithm: sort by page count
  - Allocate sequentially in decreasing order
  - Packs most frequently accessed pages into first chip
  - Provides insight into potential benefits (if any) of page movement and motivate an on-line algorithm

37

Frequency vs. First-Touch (NT)



- Base: dual-state nap sequential
- Thresholds: 100 A->N; 5,000 N->PDN
- Opportunity for further improvements beyond first-touch

38

Hardware Support for Page Movement

- Data collection hardware
  - Reserve n pages in chip 0 (n=128)
  - 10-bit saturating counter per physical page
- On-line Algorithm
  - Warmup for 100ms, sample accesses for 2ms
  - Sort counts, move 128 most frequent pages to reserved pages in hot chip, repack others into minimum number of chips
- Preliminary experiments and results
  - Use 0.011ms and 0.008mJ for page move
  - 10% improvement for winword
  - Need to consider in execution-driven simulator

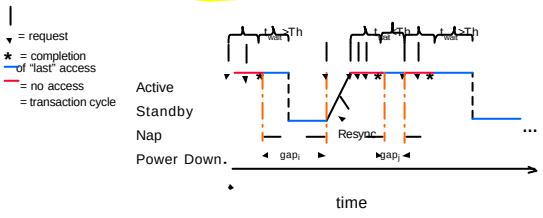
39

Hardware Policies

- Without OS support
  - Thresholds not obvious
- With OS support
  - Idle DRAM chips power down
  - Can we decouple thresholds for Nap & Power down?
- Recent studies advocate "smarter" polices [Delaluz HPCA '01]
  - No caches, no virtual memory
- Are "smarter" policies required for cache-based systems?
- Use analytic modeling to evaluate space
- Validate with simulation

40

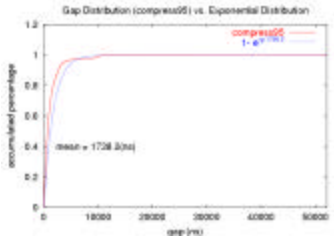
Key Parameters for Model



- Gap: time between clusters of accesses
- Threshold (Th): time to remain in high power before transition
- How does gap affect threshold selection?

41

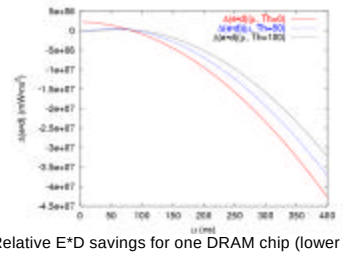
Distribution of Accesses



- Can approximate gaps using exponential
  - Same mean (μ)
  - Use analytic instead of many many simulations

42

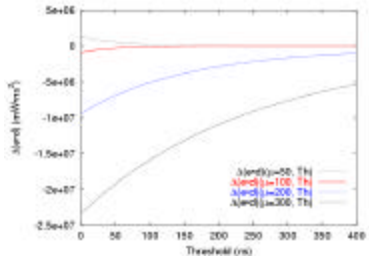
Change in E*D vs. Gap



- Relative E*D savings for one DRAM chip (lower better)
- Transition immediately to lower power state (Th = 0) is best for reasonably large average gap
  - Resynch cost for small gaps

43

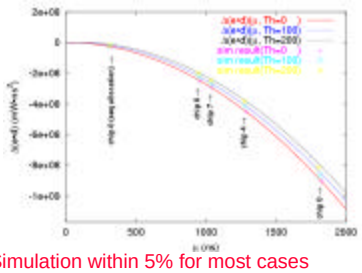
Change in E*D vs. Threshold



- For fixed average gap, increasing threshold reduces amount of time in lower power state

44

Model Validation



- Simulation within 5% for most cases
  - 20%-50% for small average gap (small values)
- Long wait time for PDN, 0 threshold for Active -> Nap

45

Conclusion

- Energy is an important metric for Post-PC computing
- Memory is unexplored, but increasingly important
- New DRAM technologies provide opportunity
  - Multiple power states
- Hardware power mode management
- Effects of operating system page allocation
- Demonstrated an effective cooperative hardware / software solution for energy management of main memory
- What *else* can we do within the OS (with appropriate architectural support)?

46