Trends in Memory Systems

Prof. Bruce Jacob
Keystone Professor & Director of Computer Engineering Program
Electrical & Computer Engineering
University of Maryland at College Park

Maryland Memory-Systems Research

- Cagdas Dirik, Ph.D. 2009. *Performance Analysis of NAND* ***Flash Memory Solid-State Disks***. (SanDisk)
- Sadagopan Srinivasan, Ph.D. 2007. ***Prefetching vs. the Memory System****: Optimizations for Multi-core Server Platforms*. (Intel)
- Brinda Ganesh, Ph.D. 2007. *Understanding and Optimizing* ***High-Speed Serial Memory-System*** *Protocols*. (Intel)
- Ankush Varma, Ph.D. 2007. *High-Speed* ***Performance, Power, and Thermal Co-Simulation*** *for SoC Design*. (Intel)
- Nuengwong (Ohm) Tuaycharoen, Ph.D. 2006. ***Disk Design-Space Exploration*** *in Terms of System-Level Performance, Power, and Energy Consumption*. (Dhurakijpundit University, Thailand)
- Samuel Rodriguez, Ph.D. 2006. *myCACTI: A New* ***Cache-Design Tool*** *for Pipelined Nanometer Caches*. (AMD)
- Aamer Jaleel, Ph.D. 2005. *The Effects of Out-of-Order Execution on the* ***Memory System***. (Intel)
- David Tawei Wang, Ph.D. 2005. *Modern* ***DRAM Memory Systems****: Performance Analysis and a High Performance, Power-Constrained DRAM-Scheduling Algorithm*. (MetaRAM, RIP)

Yesterday's high-performance technologies
are today's embedded technologies,
but yesterday's embedded-systems ***issues***
are today's high-performance ***issues***

Ankush Varma, U. Maryland PhD 2007 (Intel)

What's the point, exactly?

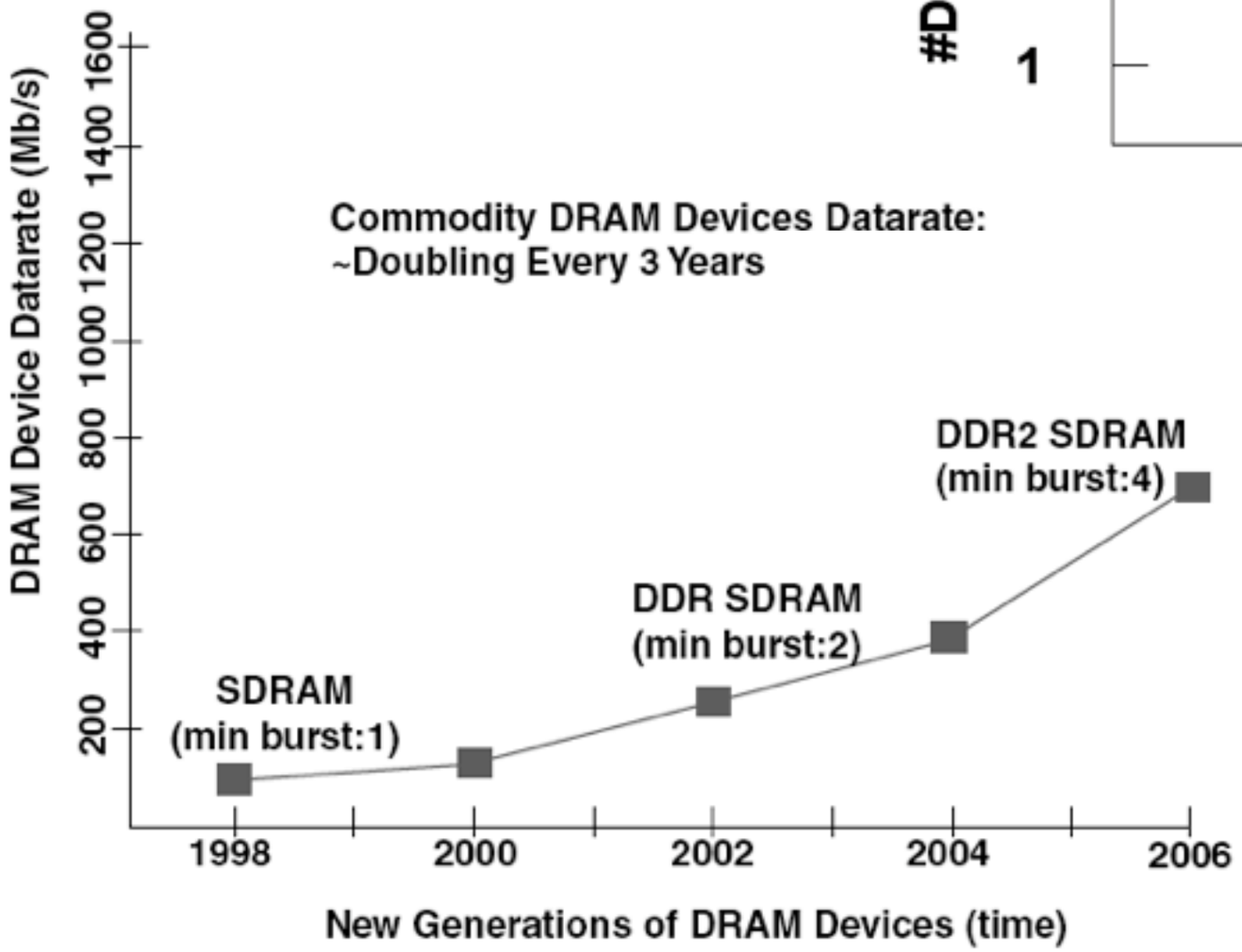
- Embedded systems designers care deeply about power & heat dissipation, cost, physical size, and correctness of design.

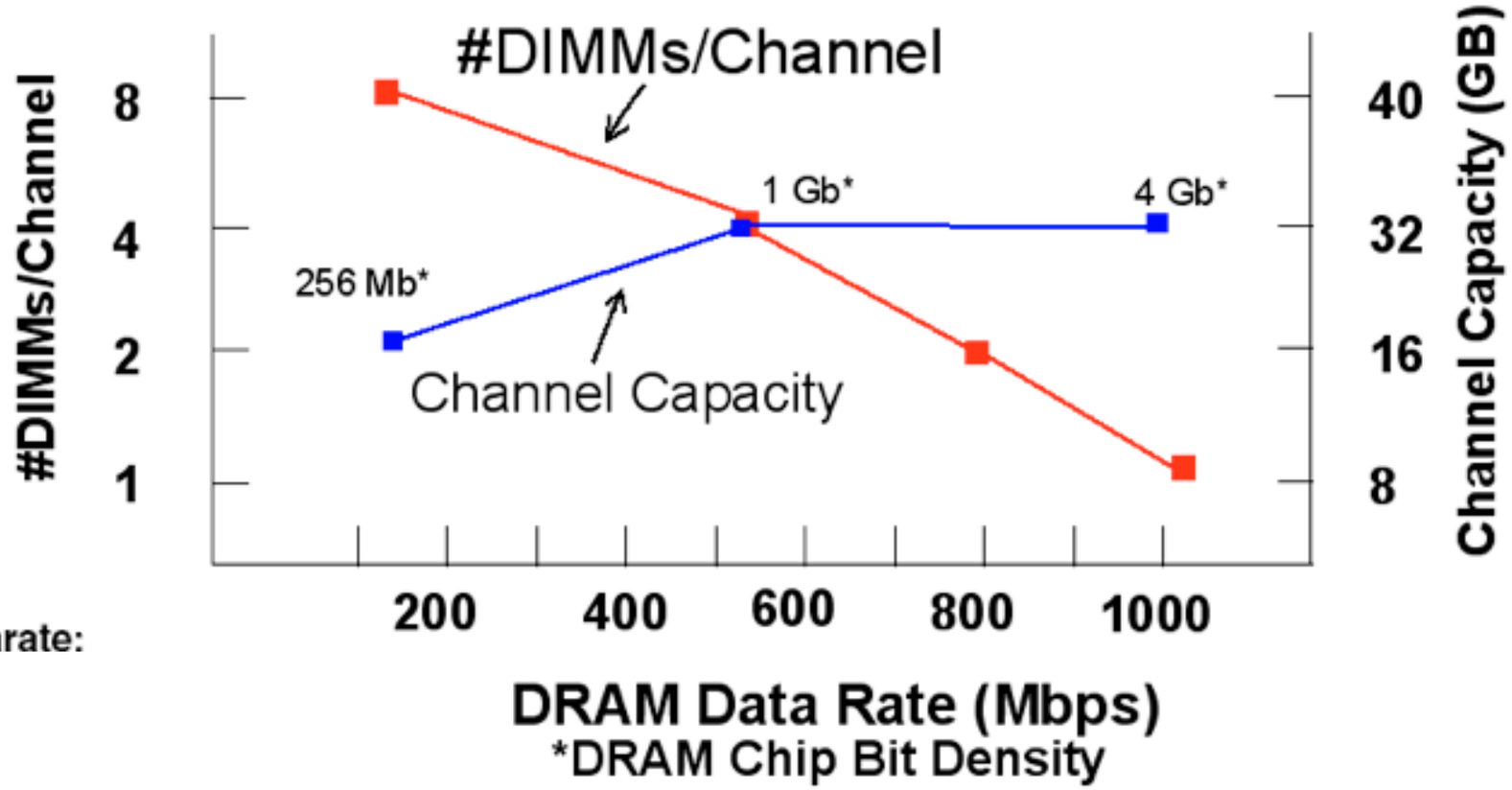
- The memory system has become the dominant concern in performance, and it is rapidly becoming a/the dominant concern in power. Our ability to deliver main-memory capacity is nonexistent, limited by bandwidth and power. Bandwidth is also a problem, limited by power/heat, as well as physical size. In larger systems, correctness is critical. The list goes on.

- The embedded-systems community has SOLVED (or at least ADDRESSED more-or-less successfully) issues of correctness, power/space, etc. … in particular the very issues that now confront the general-purpose community.

- Pretty obvious where to look if you want to predict the near-term future …

Problem: Capacity

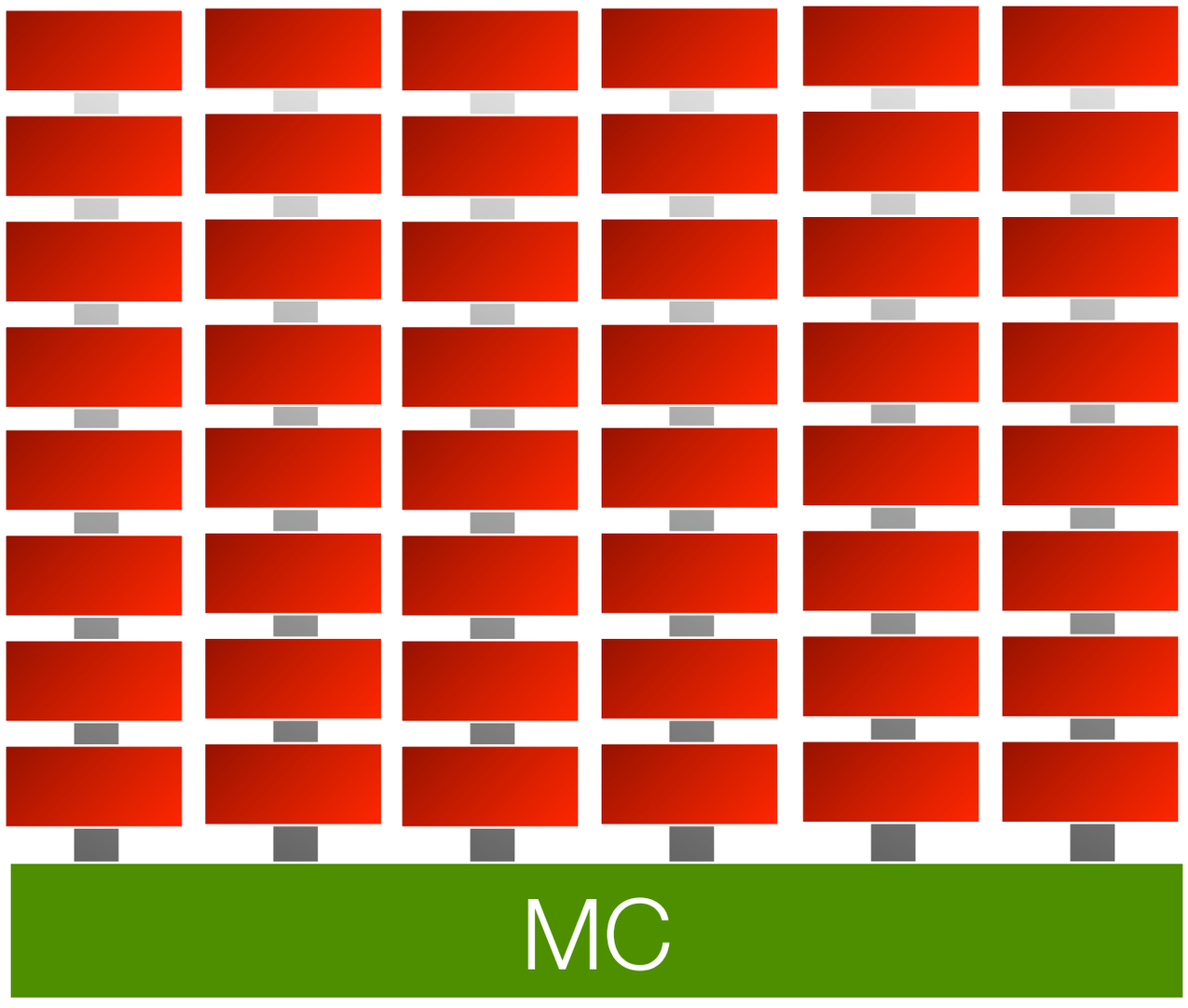




Problem: Capacity

JEDEC DDRx

~10W/DIMM, ~20W total



FB-DIMM

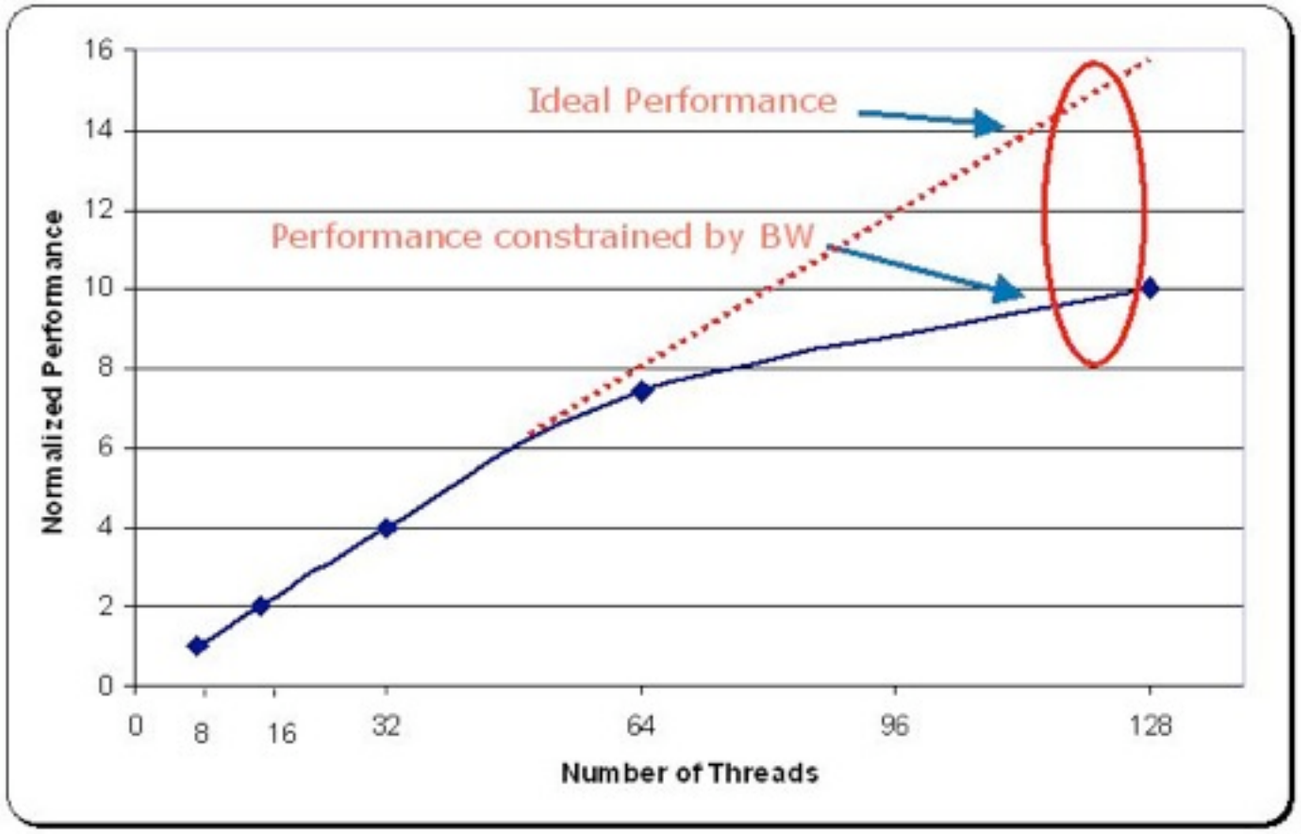
~10W/DIMM, ~300W total

Problem: Bandwidth

- Like capacity, primarily a power and heat issue: can get more BW by adding busses, but they need to be narrow & thus fast. Fast = hot.

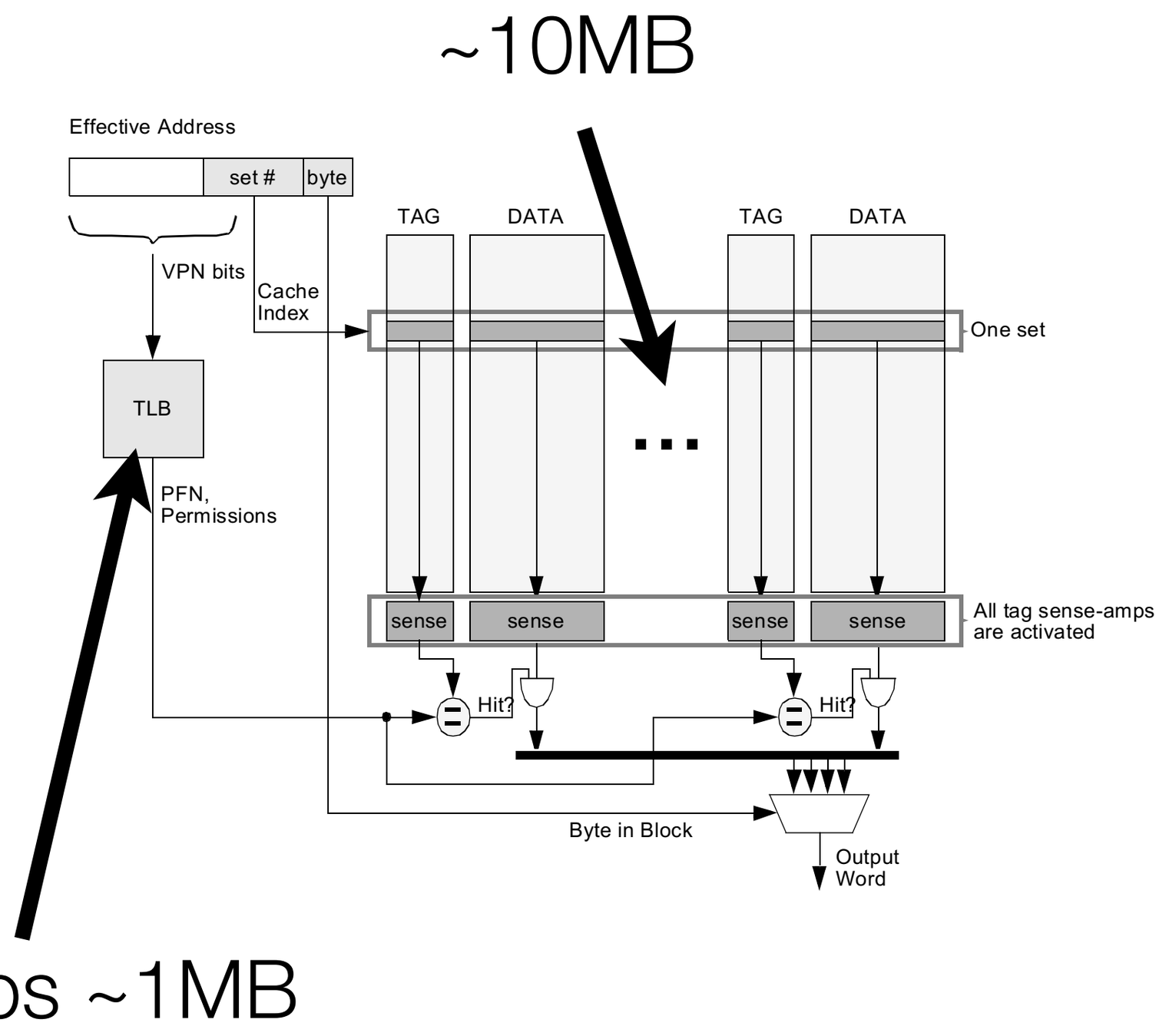
- Required BW per core is roughly 1 GB/s, and cores per chip is increasing

- Graph: Thread-based load (SPECjbb), memory set to 52GB/s sustained
  … cf. 32-core Sun Niagara: saturates at 25.6 GB/s

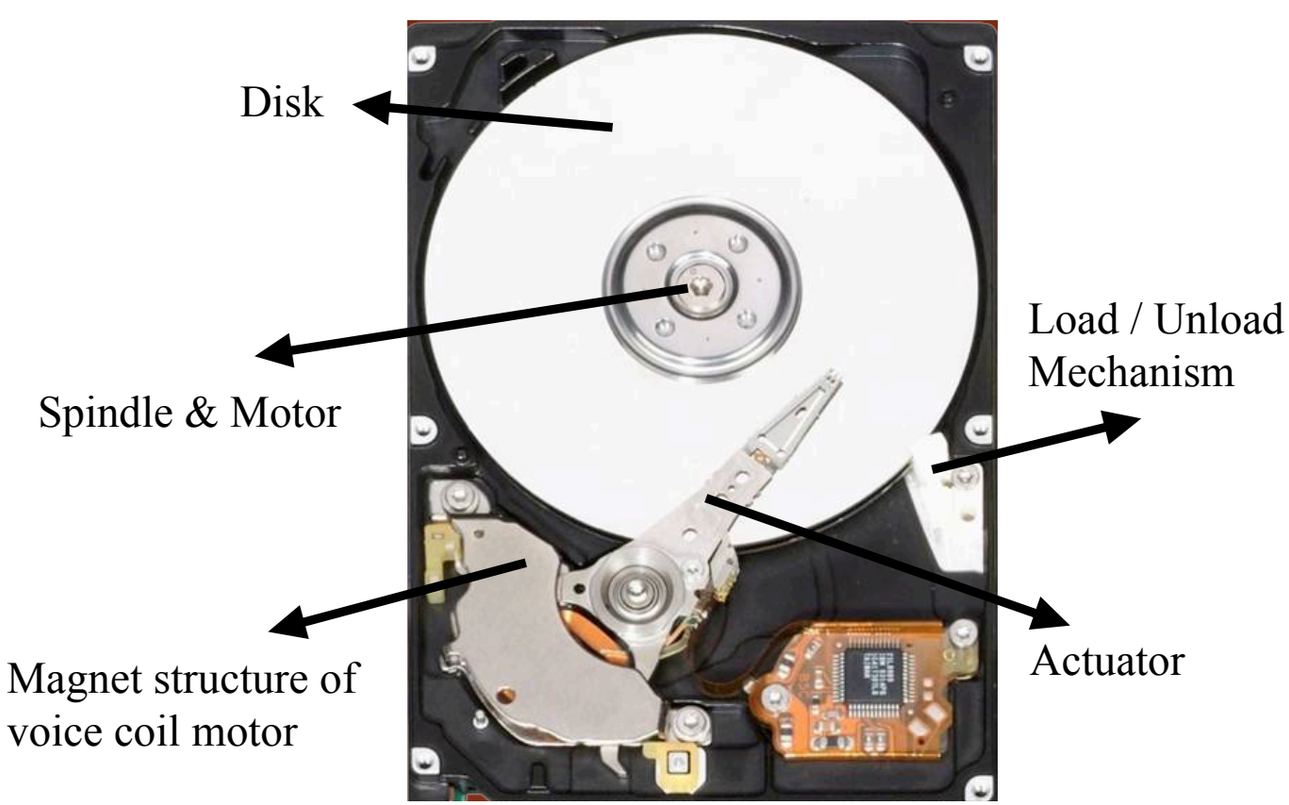


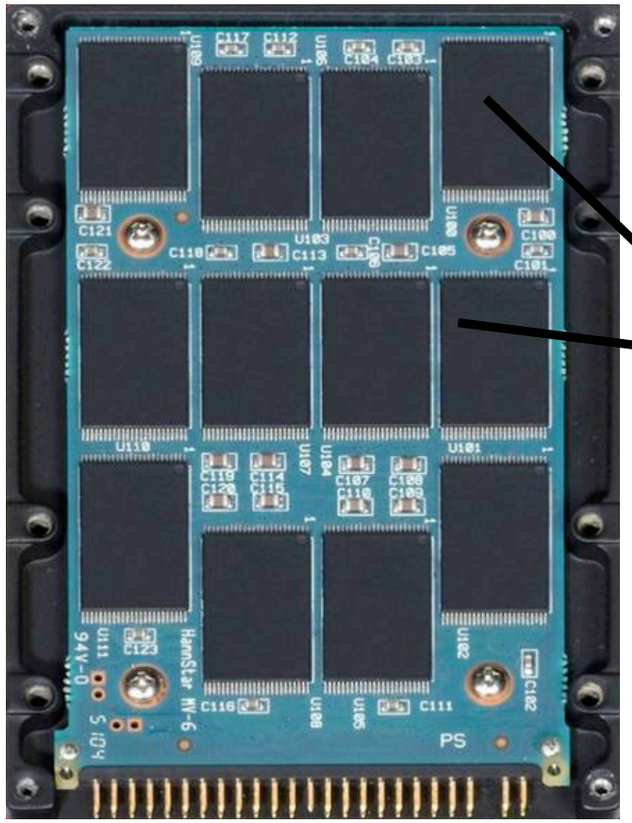
Problem: TLB Reach

- Doesn't scale at all (still small and not upgradeable)
- Currently accounts for **20+%** of system overhead
- Higher associativity (which offsets the TLB's small size) can create a power issue
- The TLB's "reach" is actually much worse than it looks, because of different access granularities



Trend: Disk, Flash, and other NV

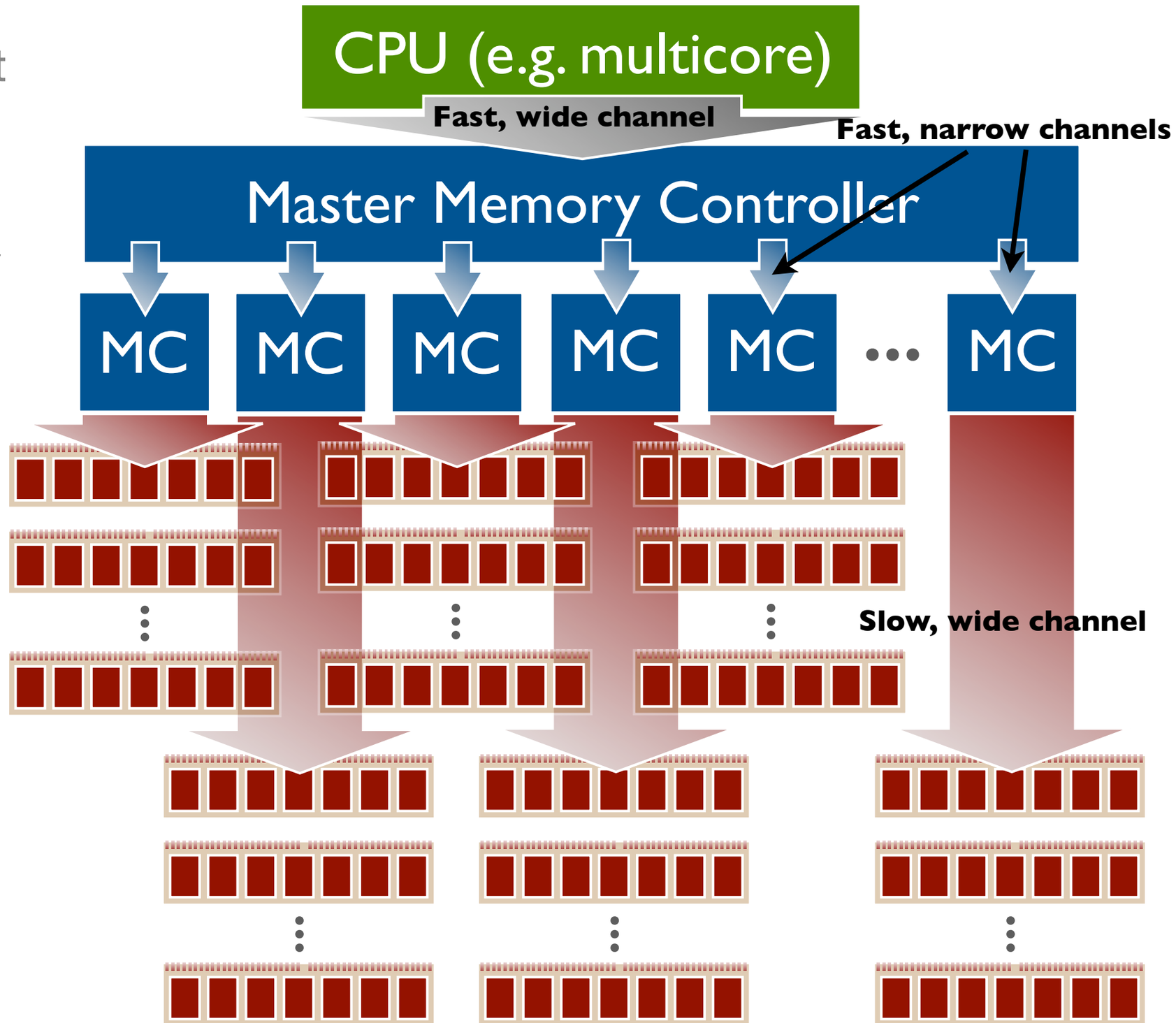




- Flash is currently eating Disk's lunch
- PCM is expected to eat Flash's lunch

Obvious Conclusions I

- Want capacity without sacrificing bandwidth
- **Need a new memory system architecture**
- This is coming (details will change, of course)



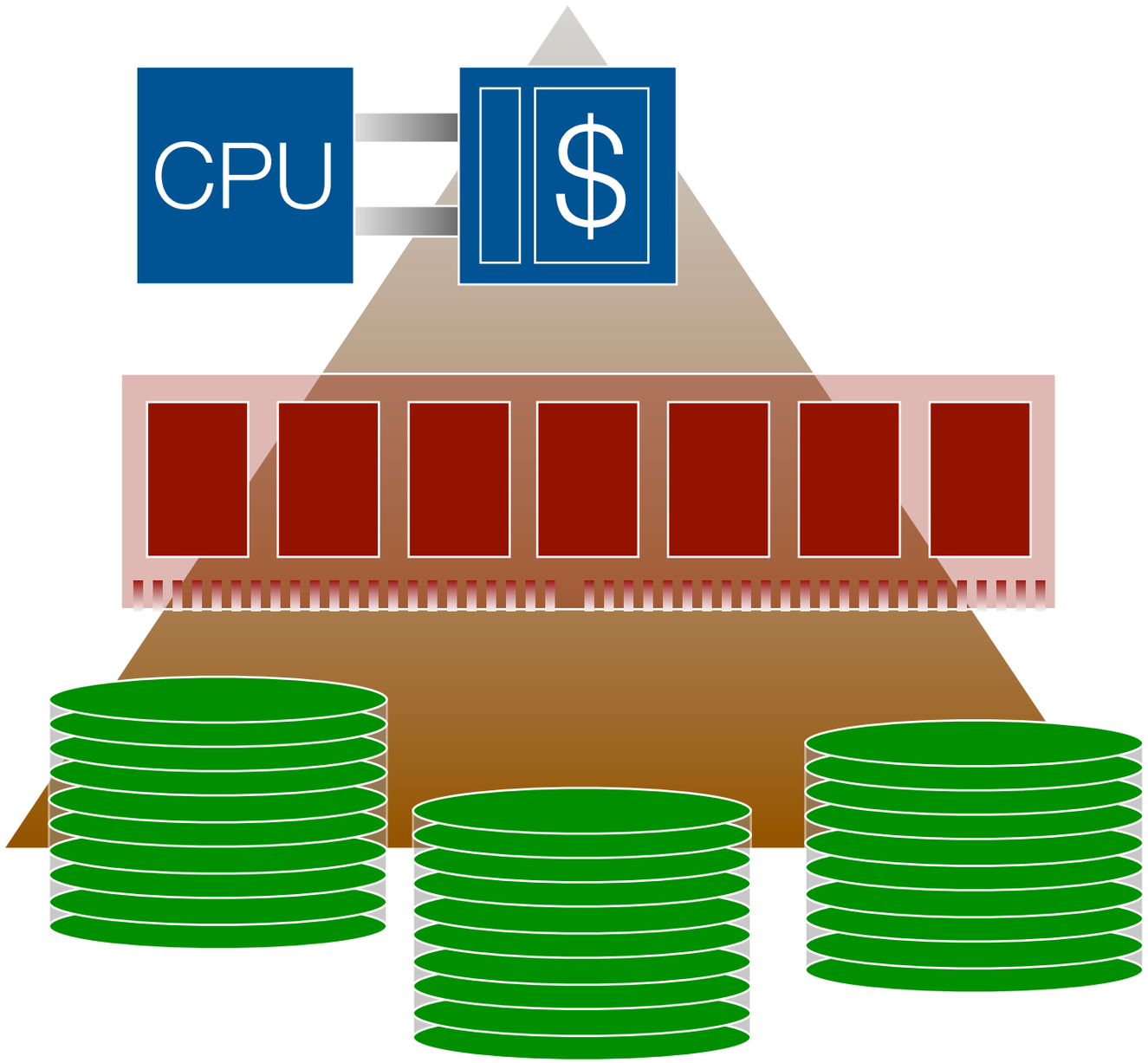
Obvious Conclusions II

- Flash/NV is inexpensive, is fast (rel. to disk), and has better capacity roadmap than DRAM

- **Make it a first-class citizen in the memory hierarchy**

- Access it via load/store interface, use DRAM to buffer writes, software management

- Probably reduces capacity pressure on DRAM system



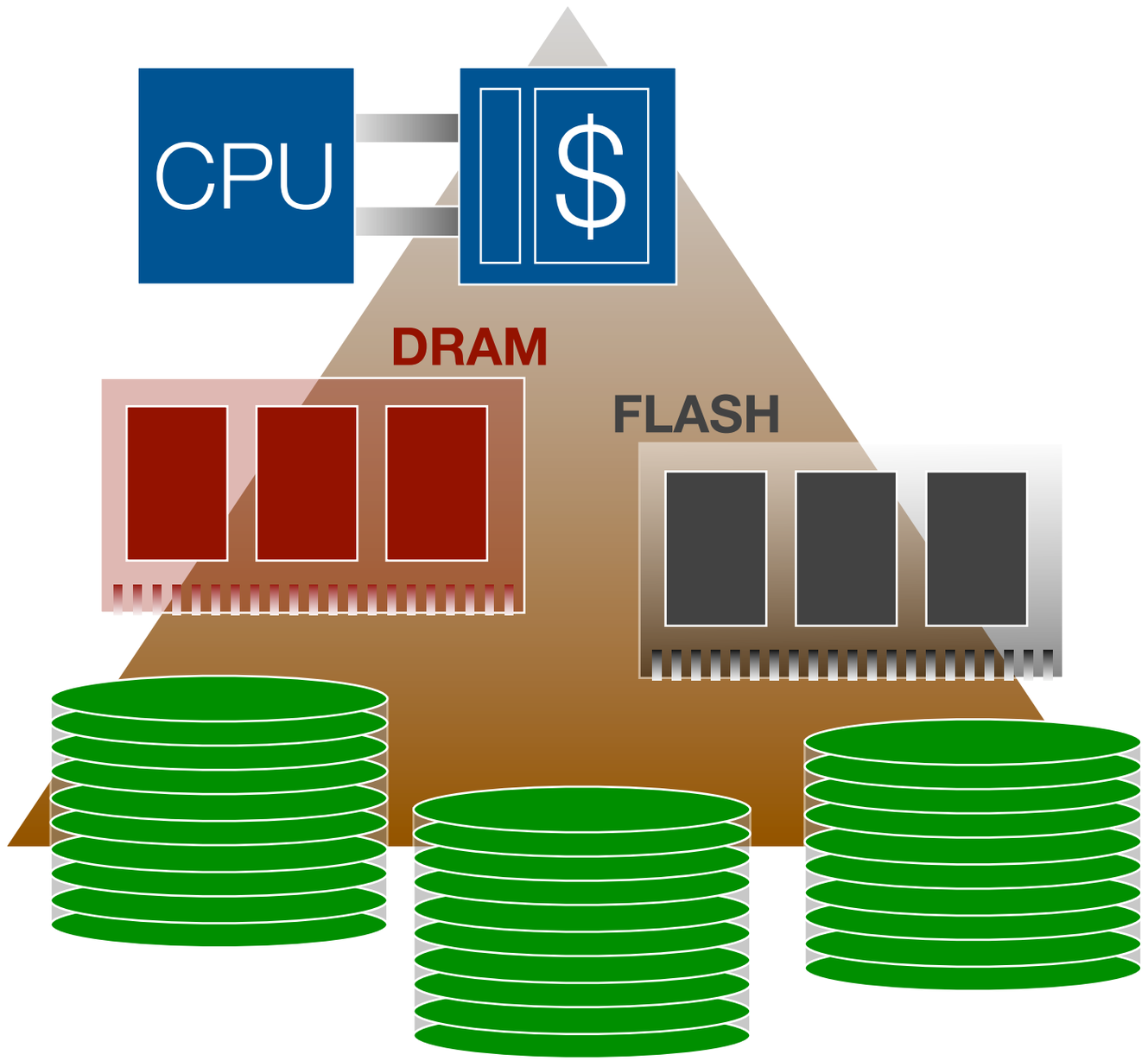
Obvious Conclusions II

- Flash/NV is inexpensive, is fast (rel. to disk), and has better capacity roadmap than DRAM

- **Make it a first-class citizen in the memory hierarchy**

- Access it via load/store interface, use DRAM to buffer writes, software management

- Probably reduces capacity pressure on DRAM system



Obvious Conclusions III

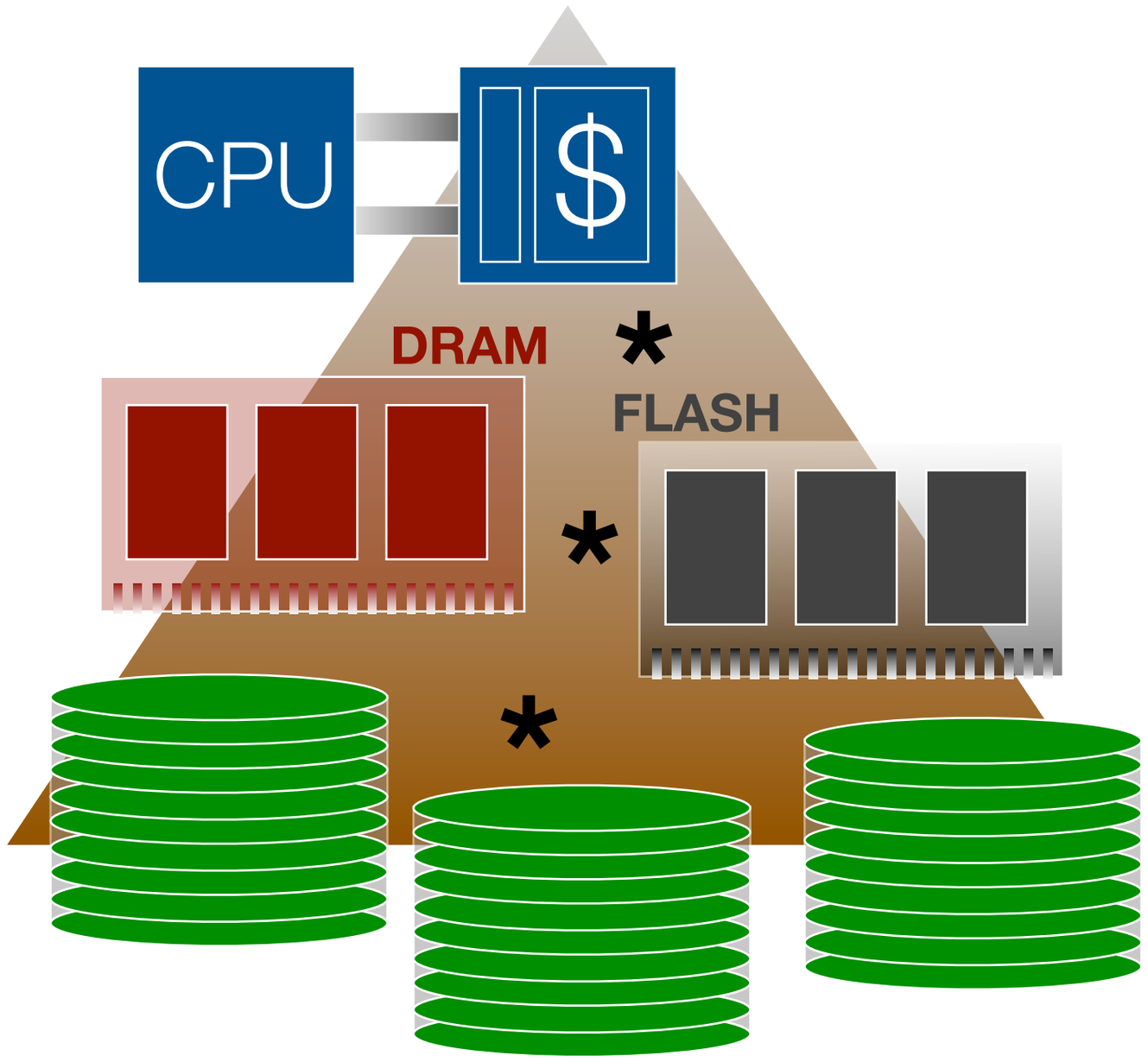
- Reduce translation overhead (both in performance and in power)

- **Need an OS/arch redesign**

- Revisit superpages, multi-level TLBs

- Revisit SASOS concepts, *location of translation point/s*

- Probably most suited for the high-end, at least initially



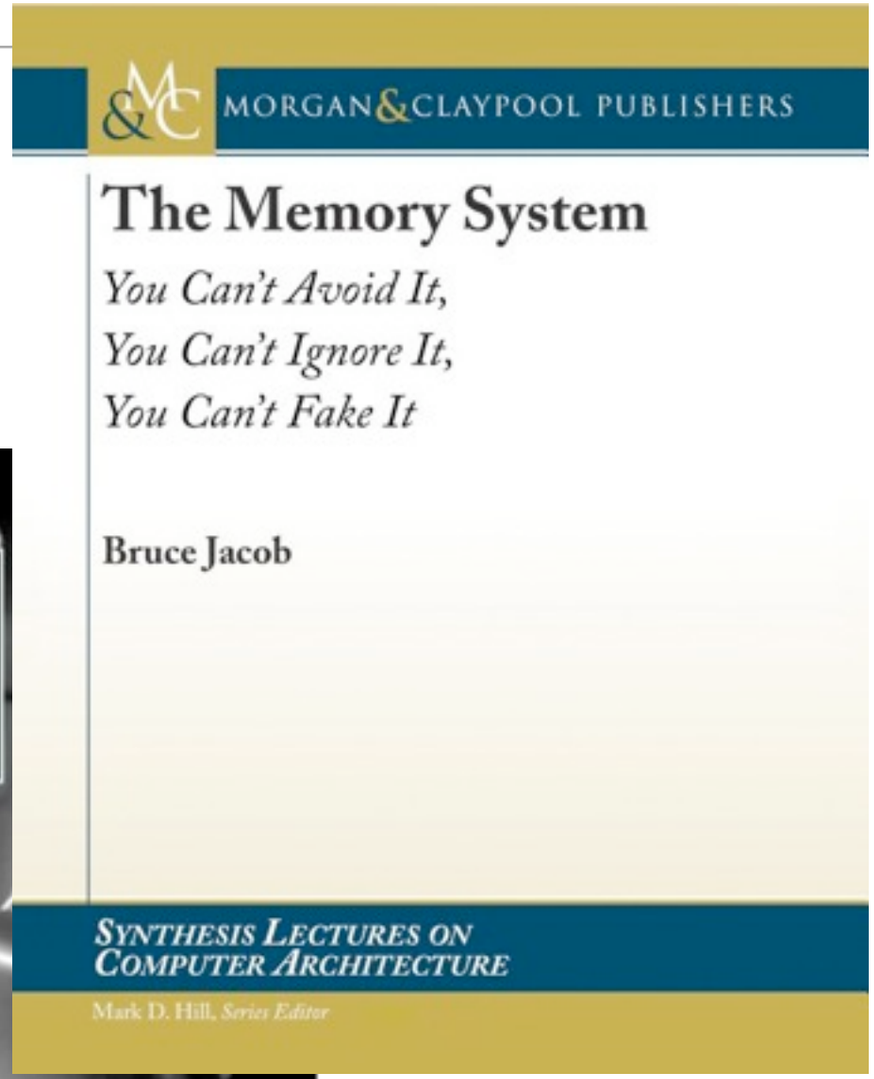
… and while we're on the topic of high-end …

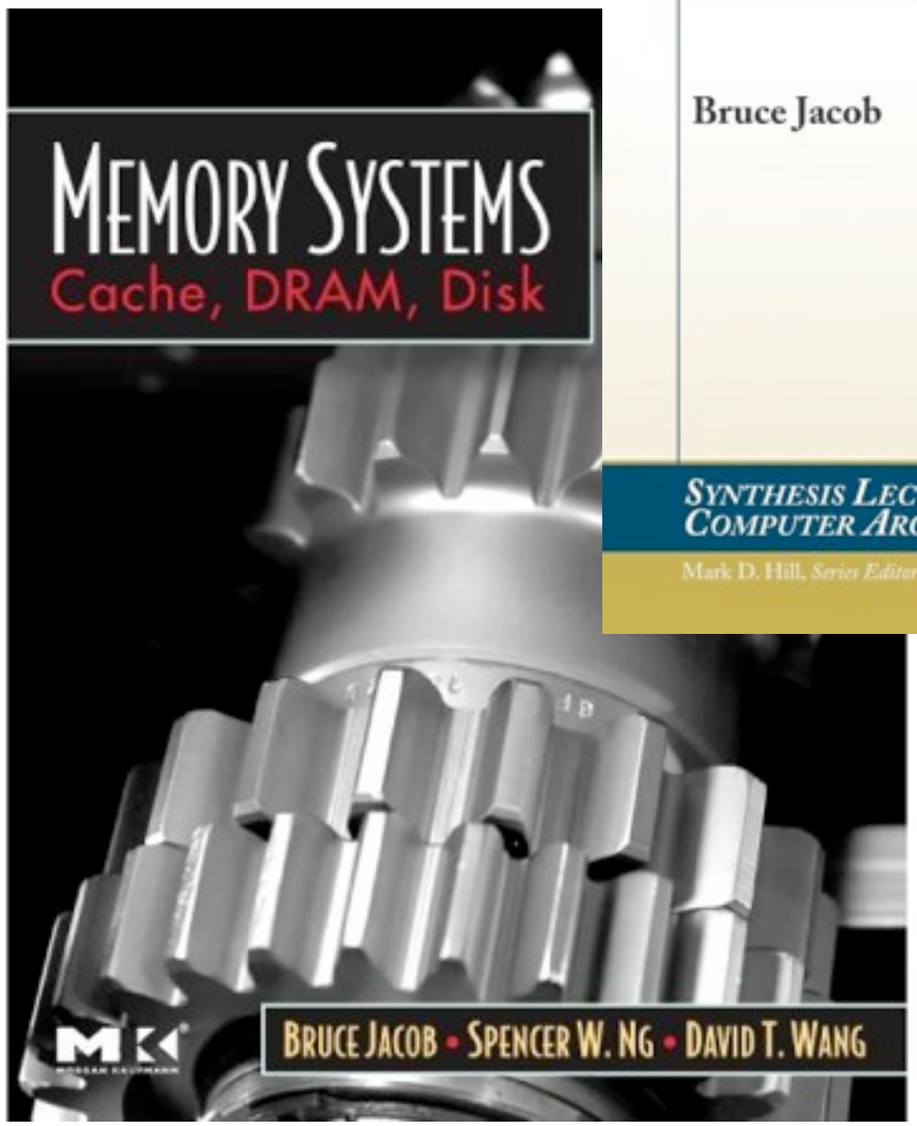
Enterprise & Super- Computing

- Run same app (set of apps) 24x7
- Developers spend significant time/energy optimizing apps
- Frequently run a custom (or at least fine-tune the existing) OS
- Have significant, pressing correctness/failure/dependability issues
  => *not intrinsic to application area, but because of large-scale multipliers*
- Care very deeply about energy consumption and heat dissipation
  => *not intrinsic to application area, but because of large-scale multipliers*
- Sounds a lot like embedded systems, no?

Acknowledgements & Shameless Plugs

- Much of this has appeared previously in our books, papers, etc.
  - The Memory System (You Can't Avoid It; You Can't Ignore It; You Can't Fake It). B. Jacob, with contributions by S. Srinivasan and D. T. Wang. ISBN 978-1598295870. Morgan & Claypool Publishers: San Rafael CA, 2009.
  - Memory Systems: Cache, DRAM, Disk. B. Jacob, S. Ng, and D. Wang, with contributions by S. Rodriguez. ISBN 978-0123797513. Morgan Kaufmann: San Francisco CA, 2007.
- Support from Intel, DoD, DOE, Sandia National Lab, Micron, Cypress Semiconductor





Questions?

(thank you for your kind indulgence)

Prof. Bruce Jacob
Keystone Professor & Director of Computer Engineering Program
Electrical & Computer Engineering
University of Maryland at College Park

blj@umd.edu
www.ece.umd.edu/~blj

… or just google “bruce jacob”