CS533: Processor in memory (PIM)

Josep Torrellas

University of Illinois in Urbana-Champaign

March 31, 2015

Today's Paper

A Case For Intelligent RAM

D. Patterson, T. Anderson, N. Cardwell, R. Fromm, K. Keeton, C. Kozyrakis, R. Thomas, K. Yelick

University of California Berkeley

Main Idea

- memory system is the greatest inhibitor of performance (low bandwidth, high latency)
- therefore: integrate a high-performance processor & DRAM main memory on a chip
  - low latency = 20-30ns instead of 300ns
  - high bandwidth = 128 bytes instead of 16 bytes

Tradeoffs

Advantages

- high bandwidth to memory
- low latency to memory
- energy consumption in memory system decreases several times
    - reduction of off-chip accesses (high capacitance)
- fewer pins necessary in chip (currently, most pins used for wide mem. interface)
    - smaller packages thanks to fewer pins
    - regular chp layout, more dense

Tradeoffs (Continued)

Advantage

- can be combined with any processor organization

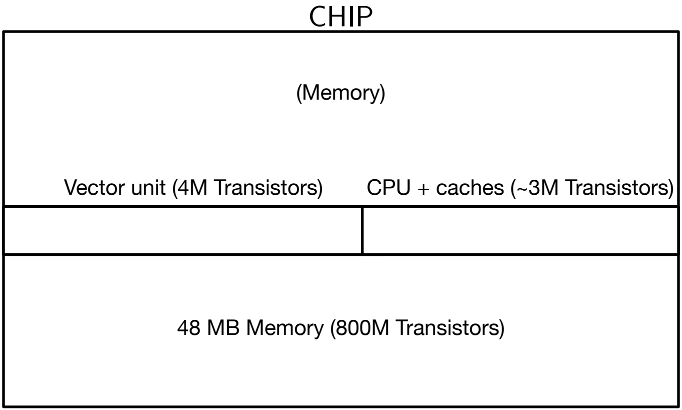
Disadvantages

- conventional processor design gains little from IRAM because they were designed with assumption of slow memory system

  → need to open up the memory

- if the programming model is too revolutionary, old apps will not run

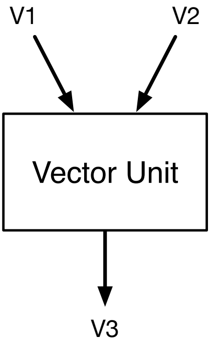
Layout


CHIP
(Memory)
Vector unit (4M Transistors)
CPU + caches (~3M Transistors)
48 MB Memory (800M Transistors)

- DRAM technology (memory) is much more dense than SRAM technology (caches)
  - 16 to 32 times more $\Longrightarrow$ more storage on chip

Vector Processors

- ↑ capacity ↑ bandwidth $\Longrightarrow$ vector processing would work well



MULTV V1,V2,V3

– Need explicit compilation into vector code

+ it is claimed that many multimedia apps will be vectorizable
    - e.g. MMX can be considered modest vector unit

Advantages of IRAM

- Higher bandwidth
- Lower latency
- Energy efficiency
- Memory size and width → free organization
- Board space (small)

Disadvantages of IRAM

- Area and speed of logic in a DRAM process technology
  - Area: 30% - 70% Larger, Speed: 30% - 70% Less
- Area and power impact of increasing bandwidth to DRAM core
- Retention time of DRAM when operating at high Temperature
  - retention rate halved for every 10° C
  - refresh rates must increase for high temperature
- Scaling system beyond single IRAM
- Matching IRAM to commodity focus of DRAM industry
- Testing (single processor, everyone?)

Challenges to IRAM

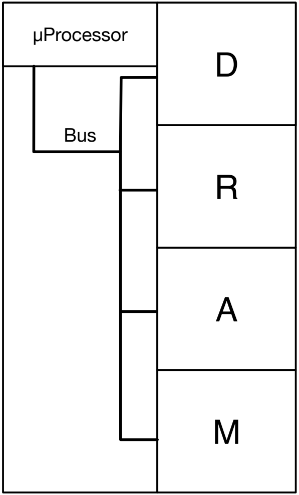
1. Fabrication process is tough
   - DRAM fabrication technology is different than logic (microprocessor) technology
     - you get slower microprocessors
     - you need to complicate design to avoid noise of switching logic on memory array
     - refresh rates increase as temperature increases
2. Bounded amount of DRAM: soon 96 MBytes
   - OK for portable computers
   - not OK for workstations
     - what about multiple IRAMs?

Evaluation

Evaluation of Existing Architectures in IRAM Systems
Christoforos Kozrakis, Ngeci Bowman, Neal Cardwell, Cynthia Rommer and Helen Wang
Workshop on "Mixing Logic and DRAM", ISCA 1997

Motivation

- Intelligent IRAM promises:
  - high memory bandwidth (100x)
  - low memory latency (0.1x)
  - high energy efficiency (4x)
  - higher system integration
- Which microprocessor architecture can turn these advantages into significant application performance benefits?



Evolutionary IRAM Approach

- Use an existing processor architecture:
  - simple RISC micro, superscalar or out-of-order execution organization
- Advantages:
  - Good knowledge of how to design and implement them
  - Performance trade-offs are well understood
  - "Out of the box" solutions borth for system software and applications – software compatibility
  - Higher performance by tuning programs and compilers to new memory hierarchy characteristics
- **This work:** evaluate potential performance benefits of this approach

IRAM Architectural Considerations

- IRAM systems using existing DRAM technology:
  - 256Mbit DRAM 0.25μm CMOS process
  - 1/4 of die area for microprocessor
  - Up to 24MBytes of on-chip DRAM
- Memory access latency can be as low as 21ns
- Logic speed potentially 10% to 50% slower compared to conventional processors for initial implementations
- No L2 cache necessary since on-chip DRAM can have comparable latency
- Memory bus as wide as cache line

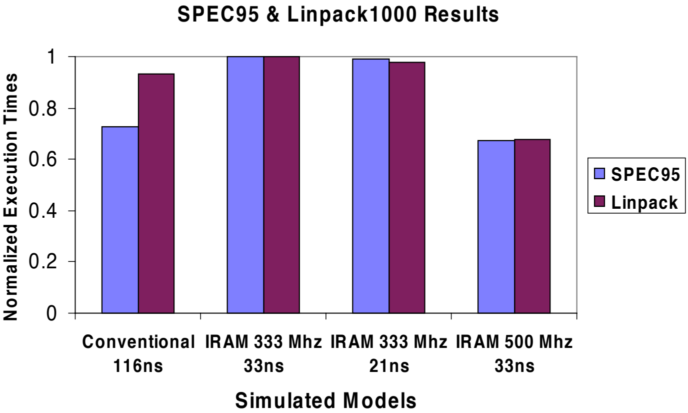
Method II: Detailed System Simulations

- Used SimOS to simulate simple MIPS R4000-based IRAM and conventional architectures
- Equal die size comparison:
  - Area for on-chip DRAM in IRAM systems same as area for L2 cache in conventional system
- Wide memory bus for IRAM systems
- Main simulation parameters:
  - On-chip DRAM access latency
  - Logic speed (CPU frequency)
- Benchmarks: SPEC95Int (compress, li, ijpeg, perl, gcc), SPEC95Fp (tomcatv, su2cor, wave5), Linpack1000

Simulated Models

	IRAM	Conventional
Pipeline	Simple in-order	Simple in-order
CPU Frequency	333 or 500 MHz	500 MHz
Technology	0.25μm DRAM	0.25μm logic
L1 Configuration	64KB I + 64KB D	64KB I + 64KB D
L1 Associativity	2-way	2-way
L1 Block Size	128B	64B I + 32B D
L1 Type	On-chip SRAM	On-chip SRAM
L1 Access Time	1 CPU cycle	1 CPU cycle
L2 Configuration	N/A	2MB unified
L2 Associativity	N/A	2-way
L2 Block Size	N/A	128B
L2 Type	N/A	On-chip SRAM
L2 Access Time	N/A	12 CPU cycles
Memory Configuration	24MB DRAM on-chip	24MB 166MHz SDRAM off-chip
Memory Bus Width	128B	16B
Total Latency	21 or 33ns	116ns

Method II: Results



- Execution times normalized to basic IRAM model (333MHz, 33ns memory latency)
- IRAM models up to 40% faster than conventional

Conclusions

- IRAM systems with existing processors provide only moderate performance benefits
- High bandwidth/low latency used to speed up memory accesses but not computation
- *Reason:* existing architectures developed under the assumption of a low bandwidth memory system
- Still attractive for portable/embedded domain
    - up to 4 times more energy efficient
    - higher system integration

Towards a Revolutionary Approach

- To provide significant performance benefits, IRAM systems need microprocessor architectures that turn memory bandwidth into application performance
- *Candidates:*
  - Vector microprocessor
  - Multithreading architectures
  - Multiprocessor on a chip
  - Some hybrid combination?
  - Some new idea?

What People are Looking at?

- Nearest neighbor database searching
- IStore (Intelligent Storage)
- Multimedia apps
- SIMD computation
- Distributed vector
- ATM switch controllers
- Scalable DSMs
- Single chip MP + DRAM
- Synchronization & special support
- Petaflop: large scale
- ...