

ECE 497NC: Unconventional Computer Architecture

Lecture 4: Processor-In-Memory Architectures 1: Rationale and Architectures

Why Integrate Processors and Memory?

- Growing gap between processor and memory speed
 - Processor speed growing at 50-60% per year
 - Memory speed growing at 10% per year historically
 - Fraction of “processor” chip transistors used to address memory latency greater than half and growing. (memory gap penalty)
- Memory capacity growing faster than needs of many systems
 - Approaching 128 MB/chip
 - Width of memory bus, not amount of memory needed, starting to determine number of memory chips in computer systems
- Becoming practical to build entire systems on one chip
 - Processor, 64MB of memory, etc. feasible on one chip

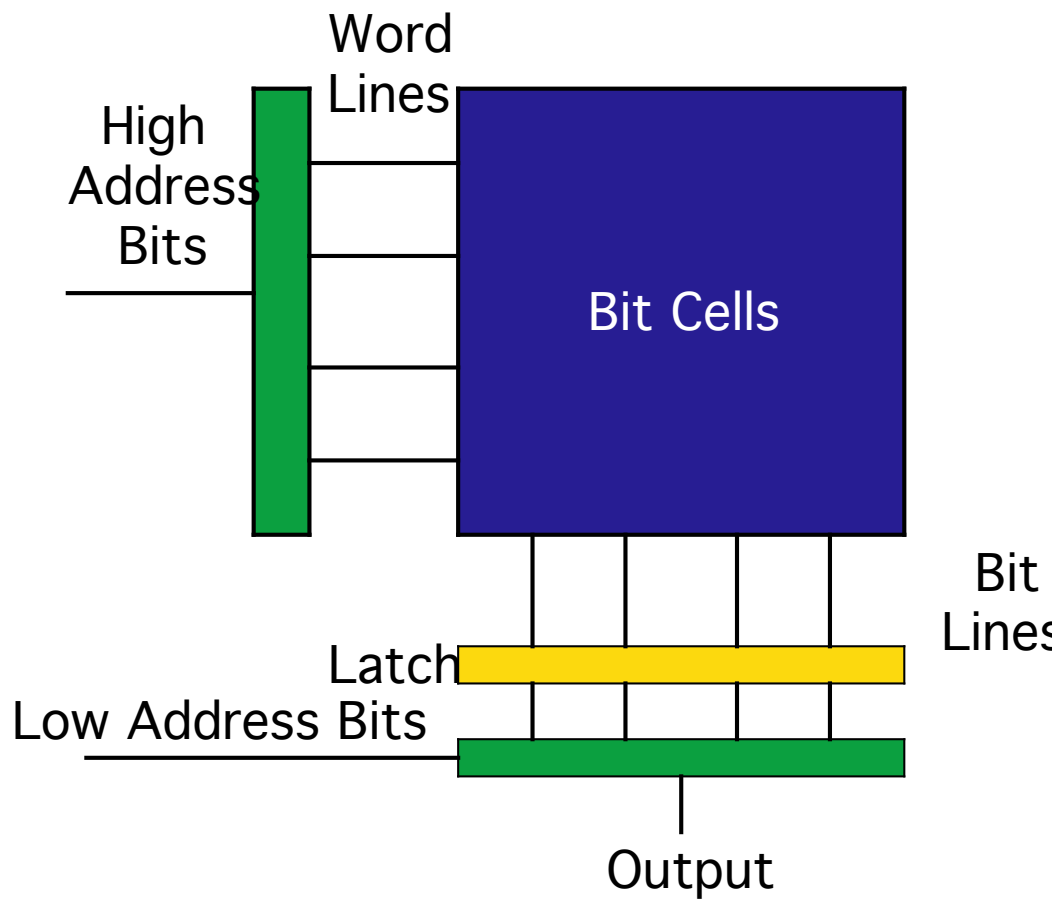
Why are DRAMs Slower than Processors?

- Capacity/Speed trade-offs in DRAMs
 - More capacitance = faster but fewer bits/chip
 - DRAM processes optimized for low leakage current at cost of speed
- Cost issues
 - I/O pins are expensive
 - Capacity of DRAM chips has grown by 64,000x, number of output bits has grown by 16x (from 1 to 16)
- One Metric -- time required to read/write the entire contents of of a memory chip
 - Has been growing at a tremendous rate in recent years.

Current DRAM State-of-the-Art

- 512 Mb chips, 4-, 8-, or 16-bit outputs
 - 1Gb chips reported in “engineering samples”
- Minimum clock period 3.75 ns (267 MHz)
 - Read/write on both clock edges (DDR) for maximum of 533 million accesses per second
 - Minimum latency of 15 ns from start of read until data available
 - Heavily pipelined, optimized for sequential accesses (burst mode)
 - Best case, requires just under half a second to read/write entire contents
 - 16-bit chip, pure sequential accesses at 533MHz
- Comparison:
 - 6502 (Apple II): DRAM read time = 1 cycle
 - 3 GHz Pentium 4: DRAM read time = 45 cycles minimum
 - In practice, cache miss time is much longer due to cache accesses, time for chip crossings, etc

DRAM Organization



On-chip bandwidth grows at at least the square root of the capacity.

Off-chip bandwidth grows much more slowly

Issues With Integrating Processors and Memory

- Fabrication Process -- DRAM optimized for density, logic for speed
 - Most efforts consider building logic in DRAM processes because of the special steps required for trench capacitors, etc.
 - Recently, IBM (and I assume, others) has begun offering mixed logic/DRAM processes
- Memory interface
 - Want to get as much bandwidth as you can, just using memory to back caches may not be most efficient technique
- Processor Architecture
 - Sharing chip with DRAM imposes different area/performance tradeoffs than separate processor and memory chips
 - Most designs aimed at imposing only limited area overhead on conventional DRAM

PIM #1 -- Active Pages

- Modern DRAMs are divided into blocks to improve access times and circuit characteristics
- Active Pages adds some reconfigurable logic to each block of bit cells -- goal is for reconfigurable logic to take up the same amount of chip area as memory.
 - RADram can be used as normal memory at small performance penalty, 50% capacity penalty
- From a computing standpoint, an Active Page is a block of data and a set of one or more computations that can be performed on the data
 - Model is that normal CPU controls program execution, with some functions offloaded into the RADram
 - To perform a computation, load data into the block, associate a function with it by configuring logic. Then can execute the function on the data.

Computing With Active Pages

- Two models
 - Memory-centric: Conventional execution of the program limited by memory bandwidth/latency, actual computational demands are low.
 - Functions of the program are offloaded into the RADram (ex: array search) , which performs a big chunk of the work
 - Processor-centric: Heavy computational demands. RADram used as co-processor to make life easier for the CPU. (ex: formatting data for sparse matrix multiply, MMX operations)
- Issues:
 - Are there programs that fit both categories simultaneously?
 - RaDram performance suffers when there's lots of computation that needs to access multiple pages
 - Interesting trade-offs between RaDram and processor utilization

Model #2 -- FlexRAM

- Divide memory into banks of 1MB (64 banks/chip in their model)
 - Assume that 70% of chip area goes to DRAM cells.
- Each bank has a P.array associated with it
 - Two-issue in-order processor
 - 8KB data cache
 - 4 P.arrays share a 8KB instruction cache
- Each chip also has a more powerful P.mem
 - Integer and floating-point operations
 - Two-issue superscalar
 - Inter-chip network allows each P.mem to communicate with the P.mems of other chips.
- Critical issue: minimizing power consumption.