RISC Design:
Pipeline Hazards

Virendra Singh
Associate Professor
Computer Architecture and Dependable Systems Lab
Department of Electrical Engineering
Indian Institute of Technology Bombay
http://www.ee.iitb.ac.in/~viren/
E-mail: viren@ee.iitb.ac.in

CP-226: Computer Architecture

Lecture 15 (15 March 2013)
Big Picture

• Memory
  – Just an “ocean of bits”
  – Many technologies are available

• Key issues
  – Technology (how bits are stored)
  – Placement (where bits are stored)
  – Identification (finding the right bits)
  – Replacement (finding space for new bits)
  – Write policy (propagating changes to bits)

• Must answer these regardless of memory type
Memory Performance Gap

The graph illustrates the increasing performance gap between processor and memory over the years from 1980 to 2010. The x-axis represents the year, and the y-axis shows performance on a logarithmic scale.

- **Processor Performance**:
  - Linear increase from 1 in 1980 to 100,000 in 2010.

- **Memory Performance**:
  - Linear increase from 1 in 1980 to 100,000 in 2010.

This indicates a significant disparity in performance growth between the processor and memory, emphasizing the need for advancements in memory technology to keep pace with processor performance.
# Types of Memory

<table>
<thead>
<tr>
<th>Type</th>
<th>Size</th>
<th>Speed</th>
<th>Cost/bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Register</td>
<td>&lt; 1KB</td>
<td>&lt; 1ns</td>
<td>$$$</td>
</tr>
<tr>
<td>On-chip SRAM</td>
<td>8KB-6MB</td>
<td>&lt; 10ns</td>
<td>$$</td>
</tr>
<tr>
<td>Off-chip SRAM</td>
<td>1Mb – 16Mb</td>
<td>&lt; 20ns</td>
<td>$$</td>
</tr>
<tr>
<td>DRAM</td>
<td>64MB – 1TB</td>
<td>&lt; 100ns</td>
<td>$</td>
</tr>
<tr>
<td>Disk</td>
<td>40GB – 1PB</td>
<td>&lt; 20ms</td>
<td>~0</td>
</tr>
</tbody>
</table>
Technology - Registers

Diagram showing a circuit with labels for C_Adx, A_Adx, and B_Adx, connected to DFFs and Data_C(i), Data_A(i), and Data_B(i).
Technology – SRAM

“Word” Lines
-select a row

“Bit” Lines
-carry data in/out

Data_C(i)
A_Adx select
B_Adx select
C_Adx select

Register Bit Slice

C = Write Port
A,B = Read Ports

Data_A(i)  Data_B(i)
Technology – DRAM

• Logically similar to SRAM
• Commodity DRAM chips
  – E.g. 1Gb
  – Standardized address/data/control interfaces
• Very dense
  – 1T per cell (bit)
  – Data stored in capacitor – decays over time
    • Must rewrite on read, refresh
• Density improving vastly over time
• Latency barely improving
Memory Timing – Read

- Latch-based SRAM or asynchronous DRAM (FPM/EDO)
  - Multiple chips/banks share address bus and tristate data bus
  - Enables are decoded from address to select bank
    - E.g. bbbbbbbbb0 is bank 0, bbbbbbbbb1 is bank 1
- Timing constraints: straightforward
  - $t_{EN}$ setup time from Addr stable to EN active (often zero)
  - $t_D$ delay from EN to valid data (10ns typical for SRAM)
  - $t_O$ delay from EN disable to data tristate off (nonzero)
Memory Timing - Write

- WR & EN triggers write of Data to ADDR
- Timing constraints: not so easy
  - $t_S$ setup time from Data & Addr stable to WR pulse
  - $t_P$ minimum write pulse duration
  - $t_H$ hold time for data/addr beyond write pulse end
- Challenge: WR pulse must start late, end early
  - $>t_S$ after Addr/Data, $>t_H$ before end of cycle
  - Requires multicycle control or glitch-free clock divider
Technology – Disk

• Bits stored as magnetic charge
• Still mechanical!
  – Disk rotates (3600-15000 RPM)
  – Head seeks to track, waits for sector to rotate to it
  – Solid-state replacements are becoming popular

• Glacially slow compared to DRAM (10-20ms)
• Density improvements astounding (100%/year)
Memory System

• Programmers want unlimited amounts of memory with low latency
• Fast memory technology is more expensive per bit than slower memory
• Solution: organize memory system into a hierarchy
  – Entire addressable memory space available in largest, slowest memory
  – Incrementally smaller and faster memories, each containing a subset of the memory below it, proceed in steps up toward the processor
• Temporal and spatial locality insures that nearly all references can be found in smaller memories
  – Gives the allusion of a large, fast memory being presented to the processor
Memory Hierarchy

- Registers
- On-Chip SRAM
- Off-Chip SRAM
- DRAM
- Disk

CAPACITY

SPEED and COST

15 Mar 2013
Computer Architecture@MNIT
Memory Hierarchy

(a) Memory hierarchy for a server

- CPU
  - Registers
  - Register reference
  - Size: 1000 bytes
  - Speed: 300 ps
- L1 Cache
  - Level 1 Cache reference
  - Size: 64 KB
  - Speed: 1 ns
- L2 Cache
  - Level 2 Cache reference
  - Size: 256 KB
  - Speed: 3–10 ns
- L3 Cache
  - Level 3 Cache reference
  - Size: 2–4 MB
  - Speed: 10–20 ns
- Memory
  - Memory reference
  - Size: 4–16 GB
  - Speed: 50–100 ns
- I/O bus
- Disk storage
  - Disk memory reference
  - Size: 4–16 TB
  - Speed: 5–10 ms

(b) Memory hierarchy for a personal mobile device

- CPU
  - Registers
  - Register reference
  - Size: 500 bytes
  - Speed: 500 ps
- L1 Cache
  - Level 1 Cache reference
  - Size: 64 KB
  - Speed: 2 ns
- L2 Cache
  - Level 2 Cache reference
  - Size: 256 KB
  - Speed: 10–20 ns
- Memory
  - Memory reference
  - Size: 256–512 MB
  - Speed: 50–100 ns
- Storage
  - FLASH memory reference
  - Size: 4–8 GB
  - Speed: 25–50 us

15 Mar 2013

Computer Architecture@MNIT
Why Memory Hierarchy?

- Need lots of bandwidth
  \[
  BW = \frac{1.0\text{inst}}{\text{cycle}} \times \left[ \frac{1\text{fetch}}{\text{inst}} \times 4B + \frac{0.4\text{Dref}}{\text{inst}} \times 4B \right] \times \frac{1\text{Gcycles}}{\text{sec}}
  \]

  = \frac{5.6\text{GB}}{\text{sec}}

- Need lots of storage
  - 64MB (minimum) to multiple TB

- Must be cheap per bit
  - (TB x anything) is a lot of money!

- These requirements seem incompatible
Memory Hierarchy Design

• Memory hierarchy design becomes more crucial with recent multi-core processors:
  – Aggregate peak bandwidth grows with # cores:
    • Intel Core i7 can generate two references per core per clock
    • Four cores and 3.2 GHz clock
      – 25.6 billion 64-bit data references/second +
      – 12.8 billion 128-bit instruction references
      – = 409.6 GB/s!
    • DRAM bandwidth is only 6% of this (25 GB/s)
  • Requires:
    – Multi-port, pipelined caches
    – Two levels of cache per core
    – Shared third-level cache on chip
Performance and Power

• High-end microprocessors have >10 MB on-chip cache
  – Consumes large amount of area and power budget
Why Memory Hierarchy?

• Fast and small memories
  – Enable quick access (fast cycle time)
  – Enable lots of bandwidth (1+ L/S/I-fetch/cycle)

• Slower larger memories
  – Capture larger share of memory
  – Still relatively fast

• Slow huge memories
  – Hold rarely-needed state
  – Needed for correctness

• All together: provide appearance of large, fast memory with cost of cheap, slow memory
Thank You