Systems I

The Memory Hierarchy

Topics
- Storage technologies
- Capacity and latency trends
- The hierarchy
Random-Access Memory (RAM)

Key features
- RAM is packaged as a chip.
- Basic storage unit is a cell (one bit per cell).
- Multiple RAM chips form a memory.

Static RAM (SRAM)
- Each cell stores bit with a six-transistor circuit.
- Retains value indefinitely, as long as it is kept powered.
- Relatively insensitive to disturbances such as electrical noise.
- Faster and more expensive than DRAM.

Dynamic RAM (DRAM)
- Each cell stores bit with a capacitor and transistor.
- Value must be refreshed every 10-100 ms.
- Sensitive to disturbances, slower and cheaper than SRAM.

Flash RAM - it’s in your ipod and cell phone
- Each cell stores 1 or more bits on a “floating-gate” capacitor
- Keeps state even when power is off
- As cheap as DRAM, but much slower
# RAM Summary

<table>
<thead>
<tr>
<th></th>
<th>Tran. per bit</th>
<th>Access time</th>
<th>Persist?</th>
<th>Sensitive?</th>
<th>Cost</th>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRAM</td>
<td>6</td>
<td>1X</td>
<td>Yes</td>
<td>No</td>
<td>100x</td>
<td>cache memories</td>
</tr>
<tr>
<td>DRAM</td>
<td>1</td>
<td>10X</td>
<td>No</td>
<td>Yes</td>
<td>1X</td>
<td>Main memories, frame buffers</td>
</tr>
<tr>
<td>Flash</td>
<td>1/2-1</td>
<td>10000X</td>
<td>Yes</td>
<td>No</td>
<td>1X</td>
<td>Disk substitute</td>
</tr>
</tbody>
</table>
Conventional DRAM Organization

**d x w DRAM:**
- dw total bits organized as d supercells of size w bits

![Diagram of DRAM organization](image-url)

- Memory controller
- Internal row buffer
- 16 x 8 DRAM chip
- Supercell (2,1)
Reading DRAM Supercell (2,1)

Step 1(a): Row access strobe (RAS) selects row 2.
Step 1(b): Row 2 copied from DRAM array to row buffer.
Reading DRAM Supercell (2,1)

Step 2(a): Column access strobe (CAS) selects column 1.

Step 2(b): Supercell (2,1) copied from buffer to data lines, and eventually back to the CPU.
Memory Modules

64 MB memory module consisting of eight 8Mx8 DRAMs

addr (row = i, col = j)

: supercell (i,j)

64-bit doubleword at main memory address A

Memory controller

64-bit doubleword
Enhanced DRAMs

All enhanced DRAMs are built around the conventional DRAM core.

- **Fast page mode DRAM (FPM DRAM)**
  - Access contents of row with [RAS, CAS, CAS, CAS, CAS] instead of [(RAS,CAS), (RAS,CAS), (RAS,CAS), (RAS,CAS)].

- **Extended data out DRAM (EDO DRAM)**
  - Enhanced FPM DRAM with more closely spaced CAS signals.

- **Synchronous DRAM (SDRAM)**
  - Driven with rising clock edge instead of asynchronous control signals.

- **Double data-rate synchronous DRAM (DDR SDRAM)**
  - Enhancement of SDRAM that uses both clock edges as control signals.

- **Video RAM (VRAM)**
  - Like FPM DRAM, but output is produced by shifting row buffer
  - Dual ported (allows concurrent reads and writes)
Nonvolatile Memories

DRAM and SRAM are volatile memories
- Lose information if powered off.

Nonvolatile memories retain value even if powered off.
- Generic name is read-only memory (ROM).
- Misleading because some ROMs can be read and modified.

Types of ROMs
- Programmable ROM (PROM)
- Eraseable programmable ROM (EPROM)
- Electrically eraseable PROM (EEPROM)
- Flash memory

Firmware
- Program stored in a ROM
  - Boot time code, BIOS (basic input/output system)
  - graphics cards, disk controllers.
Typical Bus Structure Connecting CPU and Memory

A **bus** is a collection of parallel wires that carry address, data, and control signals.

Buses are typically shared by multiple devices.
Memory Read Transaction (1)

CPU places address A on the memory bus.

Load operation: $\text{movl} \ A, \ %\text{eax}$
Memory Read Transaction (2)

Main memory reads A from the memory bus, retrieves word x, and places it on the bus.

Load operation: `movl A, %eax`
Memory Read Transaction (3)

CPU read word $x$ from the bus and copies it into register %eax.

Load operation: \texttt{movl $A$, %eax}
Memory Write Transaction (1)

CPU places address A on bus. Main memory reads it and waits for the corresponding data word to arrive.

Store operation: \texttt{movl \%eax, A}
Memory Write Transaction (2)

CPU places data word \( y \) on the bus.

Store operation: \texttt{movl} \%eax, A
Main memory read data word $y$ from the bus and stores it at address $A$.

Store operation: \texttt{movl} $\%eax$, $A$

```
%eax
\rightarrow
ALU
\rightarrow
I/O bridge
\rightarrow
main memory
<table>
<thead>
<tr>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y$</td>
</tr>
<tr>
<td>$A$</td>
</tr>
</tbody>
</table>
```
Disks consist of **platters**, each with two **surfaces**.

Each surface consists of concentric rings called **tracks**.

Each track consists of **sectors** separated by **gaps**.
Disk Geometry (Multiple-Platter View)

Aligned tracks form a cylinder.
Disk Capacity

Capacity: maximum number of bits that can be stored.
- Vendors express capacity in units of gigabytes (GB), where 1 GB = 10^9.

Capacity is determined by these technology factors:
- Recording density (bits/in): number of bits that can be squeezed into a 1 inch segment of a track.
- Track density (tracks/in): number of tracks that can be squeezed into a 1 inch radial segment.
- Areal density (bits/in^2): product of recording and track density.

Modern disks partition tracks into disjoint subsets called recording zones
- Each track in a zone has the same number of sectors, determined by the circumference of innermost track.
- Each zone has a different number of sectors/track
Computing Disk Capacity

Capacity = (# bytes/sector) \times (\text{avg. # sectors/track}) \times (\text{# tracks/surface}) \times (\text{# surfaces/platter}) \times (\text{# platters/disk})

Example:
- 512 bytes/sector
- 300 sectors/track (on average)
- 20,000 tracks/surface
- 2 surfaces/platter
- 5 platters/disk

Capacity = 512 \times 300 \times 20000 \times 2 \times 5 = 30,720,000,000 = 30.72 \text{ GB}
Disk Operation (Single-Platter View)

The disk surface spins at a fixed rotational rate.

The read/write head is attached to the end of the arm and flies over the disk surface on a thin cushion of air.

By moving radially, the arm can position the read/write head over any track.
Disk Operation (Multi-Platter View)

read/write heads move in unison from cylinder to cylinder
Disk Access Time

Average time to access some target sector approximated by:

- $T_{access} = T_{avg \ seek} + T_{avg \ rotation} + T_{avg \ transfer}$

Seek time ($T_{avg \ seek}$)

- Time to position heads over cylinder containing target sector.
- Typical $T_{avg \ seek} = 9$ ms

Rotational latency ($T_{avg \ rotation}$)

- Time waiting for first bit of target sector to pass under r/w head.
- $T_{avg \ rotation} = 1/2 \times 1/\text{RPMs} \times 60 \text{ sec/1 min}$

Transfer time ($T_{avg \ transfer}$)

- Time to read the bits in the target sector.
- $T_{avg \ transfer} = 1/\text{RPM} \times 1/(\text{avg \ # \ sectors/track}) \times 60 \text{ secs/1 min}$
Disk Access Time Example

Given:
- Rotational rate = 7,200 RPM
- Average seek time = 9 ms.
- Avg # sectors/track = 400.

Derived:
- $T_{avg\ rotation} = \frac{1}{2} \times \left( \frac{60 \text{ secs}}{7200 \text{ RPM}} \right) \times 1000 \text{ ms/sec} = 4 \text{ ms.}$
- $T_{avg\ transfer} = \frac{60}{7200} \text{ RPM} \times \frac{1}{400} \text{ secs/track} \times 1000 \text{ ms/sec} = 0.02 \text{ ms}$
- $T_{access} = 9 \text{ ms} + 4 \text{ ms} + 0.02 \text{ ms}$

Important points:
- Access time dominated by seek time and rotational latency.
- First bit in a sector is the most expensive, the rest are free.
- SRAM access time is about 4 ns/doubleword, DRAM about 60 ns
  - Disk is about 40,000 times slower than SRAM,
  - 2,500 times slower then DRAM.
Logical Disk Blocks

Modern disks present a simpler abstract view of the complex sector geometry:

- The set of available sectors is modeled as a sequence of b-sized logical blocks (0, 1, 2, ...)

Mapping between logical blocks and actual (physical) sectors

- Maintained by hardware/firmware device called disk controller.
- Converts requests for logical blocks into (surface, track, sector) triples.

Allows controller to set aside spare cylinders for each zone.

- Accounts for the difference in “formatted capacity” and “maximum capacity”.
I/O Bus

Expansion slots for other devices such as network adapters.
CPU initiates a disk read by writing a command, logical block number, and destination memory address to a port (address) associated with disk controller.
Disk controller reads the sector and performs a direct memory access (DMA) transfer into main memory.
When the DMA transfer completes, the disk controller notifies the CPU with an interrupt (i.e., asserts a special “interrupt” pin on the CPU)
## Storage Trends

### SRAM

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$/MB</td>
<td>19,200</td>
<td>2,900</td>
<td>320</td>
<td>256</td>
<td>100</td>
<td>190</td>
</tr>
<tr>
<td>Access (ns)</td>
<td>300</td>
<td>150</td>
<td>35</td>
<td>15</td>
<td>2</td>
<td>100</td>
</tr>
</tbody>
</table>

### DRAM

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$/MB</td>
<td>8,000</td>
<td>880</td>
<td>100</td>
<td>30</td>
<td>1</td>
<td>8,000</td>
</tr>
<tr>
<td>Access (ns)</td>
<td>375</td>
<td>200</td>
<td>100</td>
<td>70</td>
<td>60</td>
<td>6</td>
</tr>
<tr>
<td>Typical size(MB)</td>
<td>0.064</td>
<td>0.256</td>
<td>4</td>
<td>16</td>
<td>64</td>
<td>1,000</td>
</tr>
</tbody>
</table>

### Disk

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$/MB</td>
<td>500</td>
<td>100</td>
<td>8</td>
<td>0.30</td>
<td>0.05</td>
<td>10,000</td>
</tr>
<tr>
<td>Access (ms)</td>
<td>87</td>
<td>75</td>
<td>28</td>
<td>10</td>
<td>8</td>
<td>11</td>
</tr>
<tr>
<td>Typical size(MB)</td>
<td>1</td>
<td>10</td>
<td>160</td>
<td>1,000</td>
<td>9,000</td>
<td>9,000</td>
</tr>
</tbody>
</table>

*(Culled from back issues of Byte and PC Magazine)*
## CPU Clock Rates

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>processor</td>
<td>8080</td>
<td>286</td>
<td>386</td>
<td>Pent</td>
<td>P-III</td>
<td></td>
</tr>
<tr>
<td>clock rate(MHz)</td>
<td>1</td>
<td>6</td>
<td>20</td>
<td>150</td>
<td>750</td>
<td>750</td>
</tr>
<tr>
<td>cycle time(ns)</td>
<td>1,000</td>
<td>166</td>
<td>50</td>
<td>6</td>
<td>1.6</td>
<td>750</td>
</tr>
</tbody>
</table>
The CPU-Memory Gap

The increasing gap between DRAM, disk, and CPU speeds.
An Example Memory Hierarchy

L0: registers

L1: on-chip L1 cache (SRAM)

L2: off-chip L2 cache (SRAM)

L3: main memory (DRAM)

L4: local secondary storage (local disks)

L5: remote secondary storage (distributed file systems, Web servers)

CPU registers hold words retrieved from L1 cache.

L1 cache holds cache lines retrieved from the L2 cache memory.

L2 cache holds cache lines retrieved from main memory.

Main memory holds disk blocks retrieved from local disks.

Local disks hold files retrieved from disks on remote network servers.

Smaller, faster, and costlier (per byte) storage devices

Larger, slower, and cheaper (per byte) storage devices
Summary

Today

- Memory and storage technologies
- Trends
- Hierarchy of capacity and latency

Next time

- Principles of locality
- Cache architectures