Memory Technology 15-213

Oct. 20, 1998

Topics

- Memory Hierarchy Basics
- Static RAM
- Dynamic RAM
- Magnetic Disks
- Access Time Gap
Computer System

Processor

Cache

Memory-I/O bus

Memory

I/O controller

Disk

I/O controller

Disk

I/O controller

Display

I/O controller

Network
Levels in Memory Hierarchy

- **CPU**
  - **Register**: size: 200 B, speed: 3 ns, $/Mbyte: $100/MB, block size: 4 B

- **Cache**
  - **Register**: size: 32 KB / 4MB, speed: 6 ns, $/Mbyte: $100/MB, block size: 8 B

- **Memory**
  - **Cache**: size: 128 MB, speed: 100 ns, $/Mbyte: $1.50/MB, block size: 4 KB

- **Disk Memory**
  - **Memory**: size: 20 GB, speed: 10 ms, $/Mbyte: $0.06/MB

Directions:
- Larger, slower, cheaper
Scaling to 0.1µm

- **Semiconductor Industry Association, 1992 Technology Workshop**
  - Projected future technology based on past trends

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature size</td>
<td>0.5</td>
<td>0.35</td>
<td>0.25</td>
<td>0.18</td>
<td>0.12</td>
<td>0.10</td>
</tr>
</tbody>
</table>
  - Industry is slightly ahead of projection

- **DRAM cap**
  16M, 64M, 256M, 1G, 4G, 16G
  - Doubles every 1.5 years
  - Prediction on track

- **Chip cm²**
  2.5, 4.0, 6.0, 8.0, 10.0, 12.5
  - Way off! Chips staying small
Static RAM (SRAM)

Fast
- ~6 ns [1998]

Persistent
- as long as power is supplied
- no refresh required

Expensive
- ~$100/MByte [1995]
- 6 transistors/bit

Stable
- High immunity to noise and environmental disturbances

Technology for caches
Anatomy of an SRAM Cell

Write:
- set bit lines to opposite values
- set word line
- Flip cell to new state

Read:
- set bit lines high
- set word line high
- see which bit line goes low

Stable Configurations

0 1 1 0
SRAM Cell Principle

Inverter Amplifies
- Negative gain
- Slope < –1 in middle
- Saturates at ends

Inverter Pair Amplifies
- Positive gain
- Slope > 1 in middle
- Saturates at ends

![Graph showing gain characteristics of inverters and inverter pairs](image-url)
Bistable Element

Stability
- Require $V_{in} = V_2$
- Stable at endpoints
  - recover from perturbation
- Metastable in middle
  - Fall out when perturbed

Ball on Ramp Analogy
Example SRAM Configuration (16 x 8)

Address decoder

A0
A1
A2
A3

W0
W1
W15

b7
b1
b0

b7'
b1'
b0'

memory cells

sense/write amps

Input/output lines
d7
d1
d0

R/W
Dynamic RAM (DRAM)

Slower than SRAM
  • access time \( \sim 70 \text{ ns} \) [1995]

Nonpersistence
  • every row must be accessed every \( \sim 1 \text{ ms} \) (refreshed)

Cheaper than SRAM
  • \( \sim \$1.50 \text{ / MByte} \) [1998]
  • 1 transistor/bit

Fragile
  • electrical noise, light, radiation

Workhorse memory technology
Anatomy of a DRAM Cell

Writing
Word Line
Bit Line
Storage Node

Reading
Word Line
Bit Line

\[ \Delta V \sim \frac{C_{\text{node}}}{C_{\text{BL}}} \]
Addressing Arrays with Bits

Array Size
- R rows, \( R = 2^r \)
- C columns, \( C = 2^c \)
- \( N = R \times C \) bits of memory

Addressing
- Addresses are \( n \) bits, where \( N = 2^n \)
- \( \text{row(address)} = \text{address} / C \)
  - leftmost \( r \) bits of address
- \( \text{col(address)} = \text{address} \% C \)
  - rightmost bits of address

Example
- \( R = 2 \)
- \( C = 4 \)
- \( \text{address} = 6 \)
Example 2-Level Decode DRAM (64Kx1)

- Row address latch
- Column address latch
- Row decoder
- 256x256 cell array
- Column sense/write amps
- Column latch and decoder
- RAS
- CAS
- A7-A0
- Provide 16-bit address in two 8-bit chunks
- 8
- 256 Rows
- 256 Columns
- R/W'
- Dout Din

 exhält

DRAM Operation

Row Address (~50ns)
  • Set Row address on address lines & strobe RAS
  • Entire row read & stored in column latches
  • Contents of row of memory cells destroyed

Column Address (~10ns)
  • Set Column address on address lines & strobe CAS
  • Access selected bit
    – READ: transfer from selected column latch to Dout
    – WRITE: Set selected column latch to Din

Rewrite (~30ns)
  • Write back entire row
Observations About DRAMs

Timing

• Access time = 60ns < cycle time = 90ns
• Need to rewrite row

Must Refresh Periodically

• Perform complete memory cycle for each row
• Approx. every 1ms
• Sqrt(n) cycles
• Handled in background by memory controller

Inefficient Way to Get Single Bit

• Effectively read entire row of Sqrt(n) bits
Enhanced Performance DRAMs

Conventional Access
- Row + Col
- RAS CAS RAS CAS ...

Page Mode
- Row + Series of columns
- RAS CAS CAS CAS ...
- Gives successive bits

Other Acronyms
- EDORAM
  - “Extended data output”
- SDRAM
  - “Synchronous DRAM”

Typical Performance

<table>
<thead>
<tr>
<th>row access time</th>
<th>col access time</th>
<th>cycle time</th>
<th>page mode cycle time</th>
</tr>
</thead>
<tbody>
<tr>
<td>50ns</td>
<td>10ns</td>
<td>90ns</td>
<td>25ns</td>
</tr>
</tbody>
</table>

Entire row buffered here
Video RAM

Performance Enhanced for Video / Graphics Operations

- Frame buffer to hold graphics image

Writing

- Random access of bits
- Also supports rectangle fill operations
  - Set all bits in region to 0 or 1

Reading

- Load entire row into shift register
- Shift out at video rates

Performance Example

- 1200 X 1800 pixels / frame
- 24 bits / pixel
- 60 frames / second
- 2.8 GBits / second
DRAM Driving Forces

Capacity

- 4X per generation
  - Square array of cells

- Typical scaling
  - Lithography dimensions 0.7X
    - Areal density 2X
  - Cell function packing 1.5X
  - Chip area 1.33X

- Scaling challenge
  - Typically $C_{\text{node}} / C_{\text{BL}} = 0.1-0.2$
  - Must keep $C_{\text{node}}$ high as shrink cell size

Retention Time

- Typically 16–256 ms
- Want higher for low-power applications
DRAM Storage Capacitor

Planar Capacitor
- Up to 1Mb
- \( C \) decreases linearly with feature size

Trench Capacitor
- 4–256 Mb
- Lining of hole in substrate

Stacked Cell
- > 1Gb
- On top of substrate
- Use high \( \varepsilon \) dielectric
Trench Capacitor

Process

- Etch deep hole in substrate
  - Becomes reference plate
- Grow oxide on walls
  - Dielectric
- Fill with polysilicon plug
  - Tied to storage node

![Diagram of Trench Capacitor](image)
IBM DRAM Evolution

- IBM J. R&D, Jan/Mar '95
- Evolution from 4 – 256 Mb
- 256 Mb uses cell with area 0.6 µm

Cell Layouts

4 Mb Cell Structure

- 4Mb
- 16Mb
- 64Mb
- 256Mb
Mitsubishi Stacked Cell DRAM

- IEDM '95
- Claim suitable for 1 – 4 Gb

Technology
- 0.14 µm process
  - Synchrotron X-ray source
- 8 nm gate oxide
- 0.29 µm² cell

Storage Capacitor
- Fabricated on top of everything else
- Rubidium electrodes
- High dielectric insulator
  - 50X higher than SiO₂
  - 25 nm thick
- Cell capacitance 25 femtofarads
Mitsubishi DRAM Pictures

Fig. 3. SEM cross-sectional photomicrograph of the fabricated 0.29-μm³ memory cell with RuBST/Ru stacked capacitor. The facet was fabricated by focused ion beam etching.

Fig. 8. SEM photomicrograph of a Ru-metal storage node array with a projection to a height of 0.2 μm.

Fig. 10. SEM cross-sectional view of a Ru/BST/Ru capacitor cell. The facet shown is a cleaved facet.
Magnetic Disks

Disk surface spins at 3600–7200 RPM

The surface consists of a set of concentric magnetized rings called tracks

Each track is divided into sectors

The read/write head floats over the disk surface and moves back and forth on an arm from track to track.
## Disk Capacity

<table>
<thead>
<tr>
<th>Parameter</th>
<th>540MB Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number Platters</td>
<td>8</td>
</tr>
<tr>
<td>Surfaces / Platter</td>
<td>2</td>
</tr>
<tr>
<td>Number of tracks</td>
<td>1046</td>
</tr>
<tr>
<td>Number sectors / track</td>
<td>63</td>
</tr>
<tr>
<td>Bytes / sector</td>
<td>512</td>
</tr>
</tbody>
</table>

**Total Bytes** 539,836,416
Disk Operation

Operation
  • Read or write complete sector

Seek
  • Position head over proper track
  • Typically 10ms

Rotational Latency
  • Wait until desired sector passes under head
  • Worst case: complete rotation
    – 3600RPM: 16.7 ms

Read or Write Bits
  • Transfer rate depends on # bits per track and rotational speed
  • E.g., 63 * 512 bytes @3600RPM = 1.9 MB/sec.
  • Modern disks up to 80 MB / second
Disk Performance

Getting First Byte
  • Seek + Rotational latency 10,000 – 27,000 microseconds

Getting Successive Bytes
  • ~ 0.5 microseconds each

Optimizing
  • Large block transfers more efficient
  • Try to do other things while waiting for first byte
    – Switch context to other computing task
    – Interrupts processor when transfer completed
Disk / System Interface

Processor Signals Controller

- Read sector X and store starting at memory address Y

Read Occurs

- Direct Memory Access
- Under control of I/O controller

I / O Controller Signals Completion

- Interrupt processor
- Can resume suspended process
Magnetic Disk Technology

Seagate ST-12550N Barracuda 2 Disk

- Linear density: 52,187. bits per inch (BPI)
  - Bit spacing: 0.5 microns
- Track density: 3,047. tracks per inch (TPI)
  - Track spacing: 8.3 microns
- Total tracks: 2,707. tracks
- Rotational Speed: 7200. RPM
- Avg Linear Speed: 86.4 kilometers / hour
- Head Floating Height: 0.13 microns

Analogy
- Put Sears Tower on side
- Fly around world 2.5 cm off ground
- 8 seconds per orbit
CD Read Only Memory (CDROM)

Basis
- Optical recording technology developed for audio CDs
  - 74 minutes playing time
  - 44,100 samples / second
  - 2 X 16-bits / sample (Stereo)
  - Raw bit rate = 172 KB / second
- Add extra 288 bytes of error correction for every 2048 bytes of data
  - Cannot tolerate any errors in digital data, whereas OK for audio

Bit Rate
- \[ \frac{172 \times 2048}{288 + 2048} = 150 \text{ KB / second} \]
  - For 1X CDROM
  - \( N \times \text{CDROM} \) gives bit rate of \( N \times 150 \)
  - E.g., 12X CDROM gives 1.76 MB / second

Capacity
- 74 Minutes \times 150 \text{ KB / second} \times 60 \text{ seconds / minute} = 650 \text{ MB}
## Storage Trends

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SRAM</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$/MB</td>
<td>19,200</td>
<td>2,900</td>
<td>320</td>
<td>256</td>
<td>75</td>
</tr>
<tr>
<td>access (ns)</td>
<td>300</td>
<td>150</td>
<td>35</td>
<td>15</td>
<td>20</td>
</tr>
<tr>
<td><strong>DRAM</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$/MB</td>
<td>8,000</td>
<td>880</td>
<td>100</td>
<td>30</td>
<td>266</td>
</tr>
<tr>
<td>access (ns)</td>
<td>375</td>
<td>200</td>
<td>100</td>
<td>70</td>
<td>5</td>
</tr>
<tr>
<td>typical size(MB)</td>
<td>0.064</td>
<td>0.256</td>
<td>4</td>
<td>16</td>
<td>250</td>
</tr>
<tr>
<td><strong>Disk</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$/MB</td>
<td>500</td>
<td>100</td>
<td>8</td>
<td>0.30</td>
<td>1,600</td>
</tr>
<tr>
<td>access (ms)</td>
<td>87</td>
<td>75</td>
<td>28</td>
<td>10</td>
<td>9</td>
</tr>
<tr>
<td>typical size(MB)</td>
<td>1</td>
<td>10</td>
<td>160</td>
<td>1,000</td>
<td>1,000</td>
</tr>
</tbody>
</table>

Culled from back issues of Byte and PC Magazine
Storage price/MByte

$\$/Mbyte

Year


SRAM
DRAM
disk
Storage access times

![Graph showing storage access times for different types of storage: disk, DRAM, and SRAM over years 1980 to 1995. The access times are measured in nanoseconds (ns) and show a decreasing trend over time.]
### Processor clock rates

#### Processors

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>typical clock (MHz)</td>
<td>1</td>
<td>6</td>
<td>20</td>
<td>150</td>
<td>150</td>
</tr>
<tr>
<td>processor</td>
<td>8080</td>
<td>286</td>
<td>386</td>
<td>150</td>
<td>pentium</td>
</tr>
</tbody>
</table>

Culled from back issues of Byte and PC Magazine.
The widening processor/memory gap

Access time (ns)

Year


microprocessor clock periods  SRAM access time  DRAM access time

1000

100

10

1

The graph shows the comparisons of different access times for memory and microprocessors from 1980 to 1995.
Memory Technology Summary

Cost and Density Improving at Enormous Rates

Speed Lagging Processor Performance

Memory Hierarchies Help Narrow the Gap:
  - Small fast SRAMS (cache) at upper levels
  - Large slow DRAMS (main memory) at lower levels
  - Incredibly large & slow disks to back it all up

Locality of Reference Makes It All Work
  - Keep most frequently accessed data in fastest memory