Memory & Devices
Memory

• Random Access Memory (vs. Serial Access Memory)

• Different flavors at different levels
  – Physical Makeup (CMOS, DRAM)
  – Low Level Architectures (FPM, EDO, BEDO, SDRAM, DDR)

• Cache uses **SRAM**: Static Random Access Memory
  – No refresh (6 transistors/bit vs. 1 transistor)

• Main Memory is **DRAM**: Dynamic Random Access Memory
  – Dynamic since needs to be refreshed periodically (1% time)
  – Addresses divided into 2 halves (Memory as a 2D matrix):
    • **RAS** or *Row Access Strobe*
    • **CAS** or *Column Access Strobe*
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.
## Semiconductor Memory Types

<table>
<thead>
<tr>
<th>Memory Type</th>
<th>Category</th>
<th>Erasure</th>
<th>Write Mechanism</th>
<th>Volatility</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random-access memory (RAM)</td>
<td>Read-write memory</td>
<td>Electrically, byte-level</td>
<td>Electrically</td>
<td>Volatile</td>
</tr>
<tr>
<td>Read-only memory (ROM)</td>
<td>Read-only memory</td>
<td>Not possible</td>
<td>Masks</td>
<td></td>
</tr>
<tr>
<td>Programmable ROM (PROM)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Erasable PROM (EPROM)</td>
<td></td>
<td>UV light, chip-level</td>
<td>Electrically</td>
<td>Nonvolatile</td>
</tr>
<tr>
<td>Electrically Erasable PROM (EEPROM)</td>
<td>Read-mostly memory</td>
<td>Electrically, byte-level</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Flash memory</td>
<td></td>
<td>Electrically, block-level</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Static RAM

- Bits stored in transistor “latches” → no capacitors!
  - no charge leak, no refresh needed
- Pro: no refresh circuits, faster
- Con: more complex construction, larger per bit
- Transistors “switch” faster than capacitors charge!
Static RAM Structure

six transistors per bit ("flip flop")

0/1 = example
Static RAM Operation

• Transistor arrangement (flip flop) has 2 stable logic states

• Write
  1. signal bit line: High $\rightarrow$ 1    Low $\rightarrow$ 0
  2. address line active $\rightarrow$ “switch” flip flop to stable state matching bit line

• Read
  1. address line active
  2. drive bit line to same state as flip flop

no need for refresh!
• $t_{AA}$ (access time for address): how long it takes to get stable output after a change in address.
• $t_{ACS}$ (access time for chip select): how long it takes to get stable output after CS is asserted.
• $t_{OE}$ (output enable time): how long it takes for the three-state output buffers to leave the high-impedance state when OE and CS are both asserted.
• $t_{OZ}$ (output-disable time): how long it takes for the three-state output buffers to enter high-impedance state after OE or CS are negated.
• $t_{OH}$ (output-hold time): how long the output data remains valid after a change to the address inputs.
SRAM Read Timing (typical)

ADDR: stable ≥ t_{AA} \rightarrow \text{Max}(t_{AA}, t_{ACS}) \rightarrow t_{OH}

CS_L:

OE_L:

DOUT: valid \rightarrow t_{OZ} \rightarrow t_{OE} \rightarrow t_{OZ} \rightarrow t_{OE} \rightarrow \text{valid}

WE_L = \text{HIGH}
Dynamic RAM

• Bits stored as charge in capacitors → charge “leaks”
  – level of charge determines value
  – need refreshing even when powered
• Pro: simple, small, inexpensive
• Con: need refresh circuits, slow
• Main memory
Dynamic RAM Structure

‘High’ Voltage at Y allows current to flow from X to Z or Z to X

one transistor and one capacitor per bit
DRAM Operation

- **Address line active**
  - Transistor switch closed and current flows

- **Write**
  1. Data signal to bit line: High $\rightarrow$ 1, Low $\rightarrow$ 0
  2. Address line active $\rightarrow$ Transfers charge from bit line to capacitor

- **Read**
  1. Address line active
  2. Transfer charge from capacitor to bit line (then to amplifier)
  3. Capacitor charge must be restored!
Static RAM

• Bits stored in transistor “latches” $\rightarrow$ no capacitors!
  – no charge leak, no refresh needed
• Pro: no refresh circuits, faster
• Con: more complex construction, larger per bit

transistors “switch” faster than capacitors charge!
1-Transistor DRAM Cell

- Write: $C_s$ is charged or discharged by asserting WL and BL.
- Read: Charge redistribution takes place between bit line and storage capacitance.
- Voltage swing is small; typically around 250 mV.
DRAM Cell Observations

- 1T DRAM requires a sense amplifier for each bit line, due to charge redistribution read-out.
- DRAM memory cells are single ended in contrast to SRAM cells.
- The read-out of the 1T DRAM cell is destructive; read and refresh operations are necessary for correct operation.
- Unlike 3T cell, 1T cell requires presence of an extra capacitance that must be explicitly included in the design.
- When writing a “1” into a DRAM cell, a threshold voltage is lost. This charge loss can be circumvented by bootstrapping the word lines to a higher value than $V_{DD}$. 
1-T DRAM Cell

Cross-section

Uses Polysilicon-Diffusion Capacitance
Expensive in Area
DRAM logical organization
(4 Mbit)

- Square root of bits per RAS/CAS
Advanced DRAM Cells

Advanced 1T DRAM Cells

- Trench Cell
- Stacked-capacitor Cell
DRAM logical organization

Redundancy

- Redundant rows
- Redundant columns
- Memory Array
- Row Address
  - Row Decoder
- Column Decoder
- Column Address
- Fuse Bank
4 Key DRAM Timing Parameters

- $t_{RAC}$: minimum time from RAS line falling to the valid data output.
  - Quoted as the speed of a DRAM when buy
  - A typical 4Mb DRAM $t_{RAC} = 60$ ns
  - Speed of DRAM since on purchase sheet?

- $t_{RC}$: minimum time from the start of one row access to the start of the next.
  - $t_{RC} = 110$ ns for a 4Mbit DRAM with a $t_{RAC}$ of 60 ns

- $t_{CAC}$: minimum time from CAS line falling to valid data output.
  - 15 ns for a 4Mbit DRAM with a $t_{RAC}$ of 60 ns

- $t_{PC}$: minimum time from the start of one column access to the start of the next.
  - 35 ns for a 4Mbit DRAM with a $t_{RAC}$ of 60 ns
DRAM Read Timing

- Every DRAM access begins at:
  - The assertion of the RAS_L
  - 2 ways to read: early or late v. CAS

DRAM Read Cycle Time

Early Read Cycle: OE_L asserted before CAS_L
Late Read Cycle: OE_L asserted after CAS_L
DRAM Performance

• A 60 ns ($t_{\text{RAC}}$) DRAM can
  – perform a row access only every 110 ns ($t_{\text{RC}}$)
  – perform column access ($t_{\text{CAC}}$) in 15 ns, but time between column accesses is at least 35 ns ($t_{\text{PC}}$).
    • In practice, external address delays and turning around buses make it 40 to 50 ns

• These times do not include the time to drive the addresses off the microprocessor nor the memory controller overhead!
Typical 16 Mb DRAM (4M x 4)

RAS = Row Addr. Select
CAS = Column Addr. Select

WE = Write Enable
OE = Output Enable

$2^k \times 2^k = 4 \text{ M}$

nibble
Need for Error Correction!

- Motivation:
  - Failures/time proportional to number of bits!
  - As DRAM cells shrink, more vulnerable
- Went through period in which failure rate was low enough without error correction that people didn’t do correction
  - DRAM banks too large now
  - Servers always corrected memory systems
- Basic idea: add redundancy through parity bits
  - Simple but wasteful version:
    - Keep three copies of everything, vote to find right value
    - 200% overhead, so not good!
  - Common configuration: Random error correction
    - SEC-DED (single error correct, double error detect)
    - One example: 64 data bits + 8 parity bits (11% overhead)
    - Papers up on reading list from last term tell you how to do these types of codes
  - Really want to handle failures of physical components as well
    - Organization is multiple DRAMs/SIMM, multiple SIMMs
    - Want to recover from failed DRAM and failed SIMM!
    - Requires more redundancy to do this
    - All major vendors thinking about this in high-end machines
Architecture in practice

Figure 1. PlayStation 2000 employs an unprecedented level of parallelism to achieve workstation-class 3D performance.

Figure 2. PlayStation 2000 screenshot. (Source: Namco)

(as reported in Microprocessor Report, Vol 13, No. 5)

- Emotion Engine: 6.2 GFLOPS, 75 million polygons per second
- Graphics Synthesizer: 2.4 Billion pixels per second
- Claim: Toy Story realism brought to games!
Error Correction

- **Hard Failure**
  - Permanent defect

- **Soft Error**
  - Random, non-destructive
  - No permanent damage to memory

- Detected using Hamming error correcting code
Fast Page Mode DRAM

• Page: All bits on the same ROW (Spatial Locality)
  – Don’t need to wait for wordline to recharge
  – Toggle CAS with new column address
Extended Data Out (EDO)

- Overlap Data output w/ CAS toggle
  - Later brother: Burst EDO (CAS toggle used to get next addr)

**NOTE 1:** Implementing WE at RAS time during a Read or Write Cycle is optional. Doing so will facilitate compatibility with future EDO DRAMs.
Synchronous DRAM (SDRAM)

- Access is synchronized with an external clock
- Address is presented to RAM
- RAM finds data (CPU waits in conventional DRAM)
- Since SDRAM moves data in time with system clock, CPU knows when data will be ready
- CPU does not have to wait, it can do something else
- Burst mode allows SDRAM to set up stream of data and fire it out in block
- DDR-SDRAM sends data twice per clock cycle (leading & trailing edge)
IBM 64Mb SDRAM Operation

mode register value determines burst length and latency

Figure 5.13 SDRAM Read Timing (Burst Length = 4, CAS latency = 2)
Rambus Diagram

To processor

“Smart”
RAMBUS (RDRAM)

- Protocol based RAM w/ narrow (16-bit) bus
  - High clock rate (400 Mhz), but long latency
  - Pipelined operation
- Multiple arrays w/ data transferred on both edges of clock
RDRAM Timing

Direct RDRAM Protocol (32 byte Xfer)

- Separate Row and Column control
  - Enables pipelining, enhances performance
- Smaller Write-Read bubble
  - Increases bandwidth
- Bank conflicts still possible
  - High bank counts reduce probability of conflicts
RAMBUS DRAM (RDRAM)

- Adopted by Intel for Pentium & Itanium
- Main competitor to SDRAM
- Vertical package – all pins on one side
- Data exchange over 28 wires < 12 cm long
- Bus addresses up to 320 RDRAM chips at 1.6Gbps
- Asynchronous block protocol
  - 480ns initial access time
  - Then 1.6 Gbps data rate
The cache-line-fill operation comprises address-transport, data-access, and data-transport time.
A 66-MHz EDO DRAM system that uses a 64-bit-wide data bus with blocks of four 1M×16-bit EDO DRAMs in parallel has a peak data-transfer rate of 266 Mbytes/sec.
A 66-MHz SDRAM system that uses a 64-bit-wide data bus with blocks of four 1M×16-bit SDRAMs provides a 533-Mbyte/sec peak data-transfer rate.
A 533-MHz RDRAM system uses a two-channel configuration that comprises blocks of two 2M×8-bit RDRAMs, yielding a 16-bit datapath.
Intel 875P Chipset

- **Intel® Pentium® 4 Processor**
  - 6.4, 4.2 or 3.2 GB/s
  - DDR400/333 SDRAM

- **82875P MCH**
  - 2.0 GB/s
  - **Intel® Hub Architecture**

- **AGP8X**
  - Communication Streaming Architecture/GbE

- **Dual Independent Serial ATA Ports**
  - 150 MB/s

- **10/100 LAN Connect Interface**

- **Legacy ATA 100**

- **ICH5/ICH5R**
  - BIOS Supports HT Technology
  - Hi-Speed USB 2.0 8 Ports
  - 6 Channel Audio
  - 133 MB/s

- **DDR**
  - 6.4 GB/s

- **PCI**

- **Intel® RAID Technology (ICH5R only)**
Multiple Clock Domains II

Figure 14. Intel® 875P Chipset System Clock Block Diagram

- Most clock networks in system use spread spectrum
Intel® X58 Express Chipset Block Diagram
Intel z68 chipset diagram