Announcements

• HW 4: Malloc
  • If you haven’t yet, please fill in the partner form
    • http://tinyurl.com/CS61-Fa11-malloc-groups
  • Design checkpoint on Thursday Oct 13

• Mid-course evaluation
  • http://tinyurl.com/CS61-fa11-midcourse-eval
  • Opportunity to tell us how the course is going, and how to improve it
  • Responses are anonymous, and do not affect CUE guide scores.
Today

• Memory
• Disk drives
• I/O and memory buses
• Solid-state disks
• Storage technology trends
What's inside a computer anyway?
PCI slots

PCle (graphics)

“Southbridge”

“Northbridge”

CPU

IDE

SATA

Memory slots

Floppy

USB

Ethernet

Audio

Power
What's inside a computer anyway?
Random-Access Memory (RAM)

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

- Static RAM (**SRAM**)
  - Each cell stores a bit with a four or six-transistor circuit.
  - Retains value indefinitely, as long as it is kept powered.
  - *Relatively* insensitive to electrical noise (EMI), radiation, etc.
  - Faster and more expensive than DRAM – can be 100x more expensive!

- Dynamic RAM (**DRAM**)
  - Each cell stores bit with a capacitor. One transistor is used for access
  - Stored electric charge decays over time -- must be refreshed every 10-100 ms.
  - More sensitive to disturbances (EMI, radiation,…) than SRAM.
  - Slower, but cheaper than SRAM.
Conventional DRAM Organization

$d \times w$ DRAM:
- $d \times w$ total bits organized as $d$ supercells of size $w$ bits
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 CPU
Memory Modules

64 MB memory module consisting of eight 8Mx8 DRAMs

addr (row = i, col = j)

64-bit doubleword at main memory address A

Memory controller

64-bit doubleword
DRAM packaging

- Dual In-Line Package (DIP)
- Single In-Line Pin Package (SIPP)
- Single In-Line Memory Module (SIMM) 30-pin 72-pin
- Double In-Line Memory Module (DIMM) 168-pin
- Double Data Rate (DDR) DIMM (184-pin)

Enhanced DRAMs

- DRAM Cores with better interface logic and faster I/O:
  - Synchronous DRAM (SDRAM)
    - Uses a conventional clock signal instead of asynchronous control
  - Double data-rate synchronous DRAM (DDR SDRAM)
    - Double edge clocking sends two bits per cycle per pin
  - RamBus™ DRAM (RDRAM)
    - Uses faster signaling over fewer wires with a transaction oriented interface protocol

- Obsolete Technologies:
  - Fast page mode DRAM (FPM DRAM)
    - Allowed re-use of row-addresses
  - Extended data out DRAM (EDO DRAM)
    - Enhanced FPM DRAM with more closely spaced CAS signals.
  - Video RAM (VRAM)
    - Dual ported FPM DRAM with a second, concurrent, serial interface
  - Extra functionality DRAMS (CDRAM, GDRAM)
    - Added SRAM (CDRAM) and support for graphics operations (GDRAM)
Nonvolatile Memories

- DRAM and SRAM are volatile memories
  - Lose information if powered off.
- Nonvolatile memories retain value even if powered off
  - Read-only memory (ROM): programmed during production
  - Magnetic RAM (MRAM): stores bit magnetically (in development)
  - Ferro-electric RAM (FeRAM): uses a ferro-electric dielectric
  - Programmable ROM (PROM): can be programmed once
  - Erasable PROM (EPROM): can be bulk erased (UV, X-Ray)
  - Electrically erasable PROM (EEPROM): electronic erase capability
  - Flash memory: EEPROMs with partial (sector) erase capability
Nonvolatile Memories

- Uses for Nonvolatile Memories
  - Firmware programs stored in ROM
    - BIOS, controllers for disks, network cards, graphics accelerators, security subsystems,…
  - Solid state disks (flash cards, memory sticks, etc.)
  - Smart cards, embedded systems, appliances
  - Disk caches
A **bus** is a collection of parallel wires that carry address, data, and control signals.

Buses are typically shared by multiple devices.
• 1. CPU places address A on memory bus

Load operation: movl A, %eax
• 2. Main memory reads $A$ from memory bus, retrieves word $x$, and places it on bus.
3. CPU reads word x from bus, copies it to register %eax

Load operation: `movl A, %eax`
Memory Write Transaction

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

```
Load operation: movl %eax, A
```

![Diagram showing memory write transaction](image)
2. CPU places data word $y$ on memory bus.

Load operation: `movl %eax, A`
3. Main memory reads data word $y$ from bus and stores it at address A.

Load operation: `movl %eax, A`
Errors happen!

- Electrical or magnetic interference can cause bits to flip from “1” to “0” (or vice versa)
  - Can be caused by cosmic rays passing through the memory chips
  - Error rates go up as densities increase: Up to one bit error per GB per HOUR

- One solution: Error-Correcting Code (ECC) RAM
  - Contains redundant information used to detect and correct bit errors: Hamming code.
  - Can detect and correct single bit errors; can detect (but not correct) 2-bit errors
  - Costs more: need more memory chips to store ECC information
  - Slower: Requires time to check and correct bit errors
Forced errors

• Around $80^\circ$ - $100^\circ$C, memory starts to have more frequent failures

*Using Memory Errors to Attack a Virtual Machine*
Govindavajhala and Appel
2003 IEEE Symposium on Security and Privacy
What happens to DRAM when you turn off the power?

- The contents are erased... right? right?
- Not so fast.

Lest We Remember: Cold Boot Attacks on Encryption Keys
Halderman et al.
USENIX Security Symposium 2008

http://citp.princeton.edu/memory/
What happens to DRAM when you turn off the power?

• The contents are erased... right? right?
• Not so fast.

Lest We Remember: Cold Boot Attacks on Encryption Keys
Halderman et al.
USENIX Security Symposium 2008

http://www.youtube.com/watch?v=JDaicPlgn9U
Today

• Memory
• Disk drives
• I/O and memory buses
• Solid-state disks
• Storage technology trends
Disk storage

- Storage devices that hold enormous amounts of data
  - Hundreds to thousands of \textit{gigabytes} \((=10^9\text{ bytes})\)
  - RAM: hundreds to thousands of megabytes
A Disk Primer

• (Rotating) Disks consist of one or more platters divided into tracks
  • Each **platter** may have one or two **heads** that perform read/write operations
  • Each **track** consists of multiple **sectors**
  • The set of sectors across all platters is a **cylinder**

![Diagram of disk components: platter, head, track, sector](image.png)
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 actuator arm and flies about 10 nanometers above 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
**Capacity** of disk is maximum number of bits that can be stored on disk

- Determined by:
  - Recording density (bits/inch)
    - Number of bits than can be squeezed into a 1-inch segment of track
  - Track density (tracks/inch)
    - Number of tracks per 1-inch segment of radius
  - Areal density (bits/inch$^2$)
    - Recording density $\times$ track density
Hard-drive teardown

Hard Disk Evolution

• IBM 305 RAMAC (1956)
  • First commercially produced hard drive
  • 5 MB capacity, 50 platters each 24” in diameter!
Hard Disk Evolution
Disk access time

• Command overhead:
  • Time to issue I/O, get the HDD to start responding, select appropriate head

• Seek time:
  • Time to move disk arm to the appropriate track
  • Depends on how fast you can physically move the disk arm
    • These times are not improving rapidly!

• Rotational latency:
  • Time for the appropriate sector to move under the disk arm
  • Depends on the rotation speed of the disk (e.g., 7200 RPM)

• Transfer time
  • Time to transfer a sector to/from the disk controller
  • Depends on density of bits on disk and RPM of disk rotation
  • Faster for tracks near the outer edge of the disk – why?
    • Modern drives have more sectors on the outer tracks!
Disk access time

- Access time dominated by seek time and rotational latency.
  - First bit in a sector is the most expensive, the rest are free.
- Disk is slooooodow...
  - SRAM access time is about 4 ns, DRAM about 60 ns
  - Disk is about 40,000 times slower than SRAM,
  - 2,500 times slower then DRAM.
- Requires careful scheduling of I/O requests
Example disk characteristics

- Seagate Constellation ES SAS 6Gb/s 1-TB Hard Drive
  - Form factor: 3.5"
  - Capacity: 1.0 TB
  - Rotation rate: 7,200 RPM
    - Avg. rotational latency: 4.16 ms
  - Platters: 2 (4 surfaces)
  - Cylinders: 248,600
  - Cache: 32MB
  - Average read time: 8.3ms
  - Average write time: 9.3ms
  - Transfer rate: Buffer to/from disk: 95-212 MB/s
    Host to/from drive (sustained): 60-150 MB/s
Disks are messy

• Disks provide a low level interface for reading and writing sectors
  • Generally read/write an entire sector at a time
    • So, what do you do if you need to write a single byte to a file?
  • No notion of “files” or “directories”, just raw sectors
  • Disk may have numerous bad sectors that need to be avoided
• Difficult to use the low level interface
Logical Disk Blocks

• Modern disks present a simpler abstract view of the complex sector geometry:
  • Disk presented as sequence of B logical blocks, each of fixed size.

• 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.

• Benefits of abstraction:
  • Controller can transparently avoid bad sectors
    • Just change mapping
  • Controller can set aside spare cylinders
    • Accounts for the difference in “formatted capacity” and “maximum capacity”
Today

- Memory
- Disk drives
- I/O and memory buses
- Solid-state disks
- Storage technology trends
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.
Reading a disk sector

Disk controller reads the sector and performs a direct memory access (DMA) transfer into main memory.
Reading a disk sector

When the DMA transfer completes, the disk controller notifies the CPU with an interrupt (i.e., asserts a special “interrupt” pin on the CPU)
Today

- Memory
- Disk drives
- I/O and memory buses
- Solid-state disks
- Storage technology trends
Solid state disks (flash memory)

- Non-volatile, solid state storage
  - No moving parts!
  - Fast access times (about 0.1 msec!!)
  - Lower power (no moving parts)
- Expensive: about $2.50/GB versus ~$0.25/GB for HDD.
Solid state disks (flash memory)

- Accessed through logical blocks
- Presents same interface as rotational disks
- Sequence of B blocks, each with P pages
  - Page typically 512B – 4KB, P is typically 32-128
Solid state access times

- **Reads**
  - Sequential read throughput: 250 MB/s
  - Random read throughput: 140 MB/s
  - Random read access time: 30µs

- **Writes**
  - Sequential write throughput: 170 MB/s
  - Random write throughput: 14 MB/s
  - Random write access time: 300µs

- **Writes significantly slower!**
  - To write a page, entire block must first be erased
  - Each block can be erased about 100,000 times before wearing out
Solid state vs. rotating

- **Pros:**
  - faster (no spinup, no seeking)
  - less power
  - more rugged (no moving parts, handle wider temperature range)
  - quieter
  - fewer errors

- **Cons**
  - wear out (wear leveling to reduce this)
  - wear leveling increases fragmentation
  - more expensive
    - But cost coming down

- Stay tuned for innovative uses of flash memory...
Today

• Memory
• Disk drives
• I/O and memory buses
• Solid-state disks
• Storage technology trends
Access times in perspective

- 2.26 GHz processor $\Rightarrow$ 1 cycle = 0.44 ns
- Use physical distance as analogy

<table>
<thead>
<tr>
<th>Data</th>
<th>Distance analogy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage technology</td>
<td>Time</td>
</tr>
<tr>
<td>Access register</td>
<td></td>
</tr>
<tr>
<td>SRAM access</td>
<td></td>
</tr>
<tr>
<td>DRAM access</td>
<td></td>
</tr>
<tr>
<td>Flash read access</td>
<td></td>
</tr>
<tr>
<td>Flash write access</td>
<td></td>
</tr>
<tr>
<td>Disk seek</td>
<td></td>
</tr>
</tbody>
</table>
Access times in perspective

- 2.26 GHz processor ⇒ 1 cycle = 0.44 ns
- Use physical distance as analogy

<table>
<thead>
<tr>
<th>Data</th>
<th>Storage technology</th>
<th>Time</th>
<th>Distance</th>
<th>Distance analogy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access register</td>
<td></td>
<td>0.5 ns</td>
<td>0.5 m</td>
<td>Within arms reach</td>
</tr>
<tr>
<td>SRAM access</td>
<td></td>
<td>10 ns</td>
<td>10 m</td>
<td>Office next door to you</td>
</tr>
<tr>
<td>DRAM access</td>
<td></td>
<td>50 ns</td>
<td>50 m</td>
<td>Office one floor away from you</td>
</tr>
<tr>
<td>Flash read access</td>
<td></td>
<td>30 μs</td>
<td>30 km</td>
<td>1.5 times length of Manhattan island</td>
</tr>
<tr>
<td>Flash write access</td>
<td></td>
<td>300 μs</td>
<td>300 km</td>
<td>approx. Boston to New York City</td>
</tr>
<tr>
<td>Disk seek</td>
<td></td>
<td>9 ms</td>
<td>9,000 km</td>
<td>approx. Boston to LA and back</td>
</tr>
</tbody>
</table>
Storage technology trends

- Different storage technologies have different price and performance trade-offs
- Price and performance properties are changing at different rates
# Storage technology trends

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<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>
<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>100</td>
<td>75</td>
<td>60</td>
<td>320 x</td>
</tr>
<tr>
<td>access (ns)</td>
<td>300</td>
<td>150</td>
<td>35</td>
<td>15</td>
<td>3</td>
<td>2</td>
<td>1.5</td>
<td>200 x</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>2,000</td>
<td>8,000</td>
<td>1,500,000 x</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DRAM</strong></td>
<td></td>
<td></td>
<td></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>1</td>
<td>0.1</td>
<td>0.06</td>
<td>130,000 x</td>
</tr>
<tr>
<td>access (ns)</td>
<td>375</td>
<td>200</td>
<td>100</td>
<td>70</td>
<td>60</td>
<td>50</td>
<td>40</td>
<td>9 x</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>2,000</td>
<td>8,000</td>
<td>125,000 x</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Disk</strong></td>
<td></td>
<td></td>
<td></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>0.01</td>
<td>0.005</td>
<td>0.0003</td>
<td>1,600,000 x</td>
</tr>
<tr>
<td>access (ms)</td>
<td>87</td>
<td>75</td>
<td>28</td>
<td>10</td>
<td>8</td>
<td>5</td>
<td>3</td>
<td>29 x</td>
</tr>
<tr>
<td>typical size(MB)</td>
<td>1</td>
<td>10</td>
<td>160</td>
<td>1,000</td>
<td>20,000</td>
<td>160,000</td>
<td>1,500,000</td>
<td>1,500,000 x</td>
</tr>
</tbody>
</table>
Storage technology trends

- Gap between CPU and memory increasing!
Next lecture

• Caching!