ECE 152
Introduction to Computer Architecture

Input/Output (I/O)

Copyright 2008 Daniel J. Sorin

Duke University

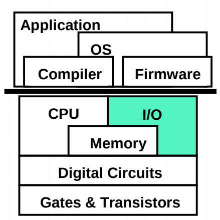
Slides are derived from work by
Amir Roth (Penn)

Spring 2008

Where We Are in This Course Right Now

- So far:
  - We know how to design a processor that can fetch, decode, and execute the instructions in an ISA
  - We understand how to design caches and memory
- Now:
  - We learn about the lowest level of storage (disks)
  - We learn about input/output in general
- Next:
  - Evaluating performance
  - Improving performance

This Unit: I/O



- I/O system structure
  - Devices, controllers, and buses
- Device characteristics
  - Disks
- Bus characteristics
- I/O control
  - Polling and interrupts
  - DMA

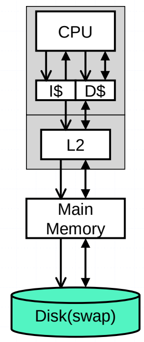
Readings

- Patterson and Hennessy
  - Chapter 8

Computers Interact with Outside World

- **Input/output (I/O)**
  - Otherwise, how will we ever tell a computer what to do…
  - …or exploit the results of its work?
- Computers without I/O are not useful
- ICQ: What kinds of I/O do computers have?

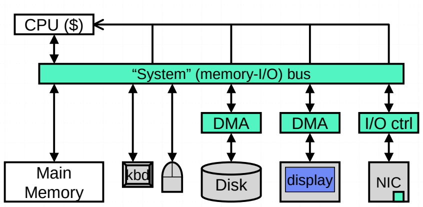
One Instance of I/O



- Have briefly seen one instance of I/O
  - **Disk**: bottom of memory hierarchy
  - Holds whatever can't fit in memory
  - ICQ: What else do disks hold?

A More General/Realistic I/O System

- A computer system
  - CPU, including cache(s)
  - Memory (DRAM)
  - **I/O peripherals**: disks, input devices, displays, network cards, ...
    - With built-in or separate I/O (or DMA) controllers
  - All connected by a **system bus**



I/O: Control + Data Transfer

- I/O devices have two ports
  - **Control**: commands and status reports
    - How we tell I/O what to do
    - How I/O tells us about itself
    - Control is the tricky part (especially status reports)
  - **Data**
    - Labor-intensive part
    - "Interesting" I/O devices do data transfers (to/from memory)
      - Display: video memory → monitor
      - Disk: memory ↔ disk
      - Network interface: memory ↔ network card

Operating System (OS) Plays a Big Role

- I/O interface is typically under OS control
  - User applications access I/O devices indirectly (e.g., SYSCALL)
  - Why?
  - Device **drivers** are “programs” that OS uses to manage devices
- **Virtualization**: same argument as for memory
  - Physical devices shared among multiple programs
  - Direct access could lead to conflicts – example?
- **Synchronization**
  - Most have asynchronous interfaces, require unbounded waiting
  - OS handles asynchrony internally, presents synchronous interface
- **Standardization**
  - Devices of a certain type (disks) can/will have different interfaces
  - OS handles differences (via drivers), presents uniform interface

I/O Device Characteristics

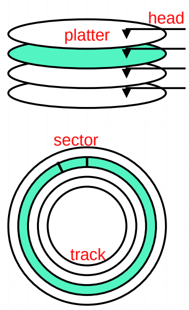
- Primary characteristic
  - **Data rate** (aka **bandwidth**)
- Contributing factors
  - **Partner**: humans have slower output data rates than machines
  - **Input or output or both (input/output)**

Device	Partner	I? O?	Data Rate (KB/s)
Keyboard	Human	Input	0.01
Mouse	Human	Input	0.02
Speaker	Human	Output	0.60
Printer	Human	Output	200.00
Display	Human	Output	240,000.00
Modem	Machine	I/O	7.00
Ethernet card	Machine	I/O	10,000.00
Disk	Machine	I/O	10,000.00

I/O Device Bandwidth: Some Examples

- Keyboard
  - 1 B/key * 10 keys/s = 10 B/s
- Mouse
  - 2 B/transfer * 10 transfers/s = 20 B/s
- Display
  - 4 B/pixel * 1M pixel/display * 60 displays/s = 240 MB/s

I/O Device: Disk



- **Disk**: like stack of record players
- Collection of **platters**
  - Each with read/write head
- Platters divided into concentric **tracks**
  - Head seeks (forward/backward) to track
  - All heads move in unison
- Each track divided into **sectors**
  - ZBR (zone bit recording)
    - More sectors on outer tracks
  - Sectors rotate under head
- **Controller**
  - Seeks heads, waits for sectors
  - Turns heads on/off
  - May have its own cache (made w/DRAM)

Disk Parameters

	Seagate ST3200	Seagate Savvio	Toshiba MK1003
Diameter	3.5"	2.5"	1.8"
Capacity	200 GB	73 GB	10 GB
RPM	7200 RPM	10000 RPM	4200 RPM
Cache	8 MB	?	512 KB
Disks/Heads	2/4	2/4	1/2
Average Seek	8 ms	4.5 ms	7 ms
Peak Data Rate	150 MB/s	200 MB/s	200 MB/s
Sustained Data Rate	58 MB/s	94 MB/s	16 MB/s
Interface	ATA	SCSI	ATA
Use	Desktop	Laptop	iPod

- Slightly newer disk from Toshiba
  - 0.85", 4 GB drives, used in iPod-mini

Disk Read/Write Latency

- Disk read/write latency has four components
  - **Seek delay (t_{seek})**: head seeks to right track
  - **Rotational delay ($t_{rotation}$)**: right sector rotates under head
    - On average: time to go halfway around disk
  - **Transfer time ($t_{transfer}$)**: data actually being transferred
  - **Controller delay ($t_{controller}$)**: controller overhead (on either side)
- Example: time to read a 4KB page assuming…
  - 128 sectors/track, 512 B/sector, 6000 RPM, 10 ms t_{seek}, 1 ms $t_{controller}$
  - 6000 RPM → 100 R/s → 10 ms/R → $t_{rotation}$ = 10 ms / 2 = 5 ms
  - 4 KB page → 8 sectors → $t_{transfer}$ = 10 ms * 8/128 = 0.6 ms
  - $t_{disk} = t_{seek} + t_{rotation} + t_{transfer} + t_{controller}$
    = 10 + 5 + **0.6** + 1 = 16.6 ms

Disk Bandwidth

- Disk is bandwidth-inefficient for page-sized transfers
  - Actual data transfer ($t_{transfer}$) a small part of disk access (and cycle)
- Increase bandwidth: **stripe data across multiple disks**
  - Striping strategy depends on disk usage model
  - "File System" or "web server": many small files
    - Map entire files to disks
  - "Supercomputer" or "database": several large files
    - Stripe single file across multiple disks
- Both bandwidth and individual transaction latency important

Error Correction: RAID

- **Error correction**: more important for disk than for memory
  - Mechanical disk failures (entire disk lost) is common failure mode
  - Entire file system can be lost if files striped across multiple disks
- **RAID (redundant array of inexpensive disks)**
  - Similar to DRAM error correction, but…
  - Major difference: which disk failed is known
    - Even parity can be used to recover from single failures
    - Parity disk can be used to reconstruct data faulty disk
  - RAID design balances bandwidth and fault-tolerance
  - Many flavors of RAID exist
    - Tradeoff: extra disks (cost) vs. performance vs. reliability
    - Deeper discussion of RAID in ECE 252 and ECE 254
  - RAID doesn't solve all problems → can you think of any examples?