10/15: Lecture Topics

- Input/Output
  - Types of I/O Devices
  - How devices communicate with the rest of the system
    - communicating with the processor
    - communicating with memory
  - I/O Performance

Why I/O?

- Without input, the only data available to programs: random numbers
- For that matter, where will the programs come from?
- Without output, there's no way to determine the result of computations

Types of I/O

- **Behavior:**
  - input only
  - output only
  - input and output
  - storage
- **Partner:** human or machine
- **Data rate:** from .01 KB/s or less to 60,000 KB/s or more

Three Characteristic Devices

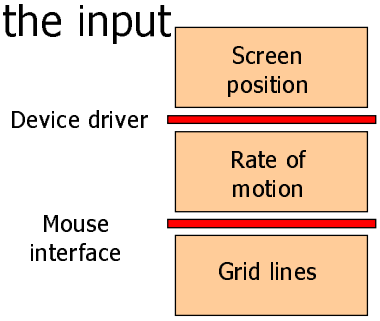
- Mouse: input only; human; .01-.02 KB/s
- Magnetic disks: storage; machine; 100-10,000 KB/s
- Networks: input and output; machine; 500-6000 KB/s

Detecting Mouse Motion

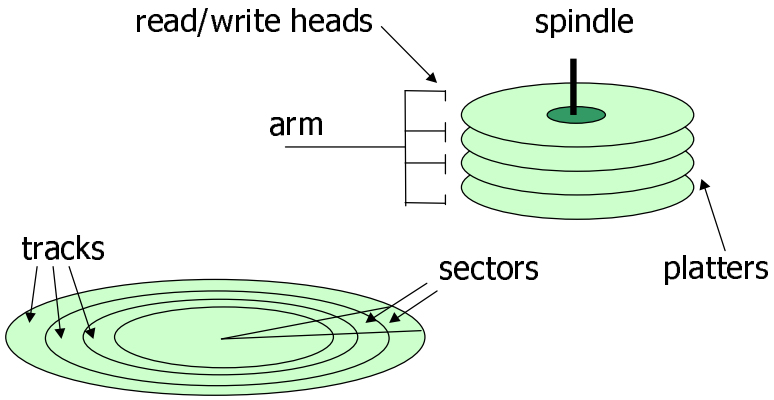
- Mechanical
  - Two perpendictular wheels on the inside connected to potentiometers
- Optical
  - LED and a photodetector on the bottom; requires a special gridded mouse pad
- Optomechanical
  - The wheels have slots; an LED shines through them
- Lots of others

Mouse Interfaces

- Pointing device must provide:
  - Status of each button
  - Rate of motion in X and Y directions
- Software must interpret the input
  - Double clicks
  - Limits to motion, speed



Anatomy of a Magnetic Disk



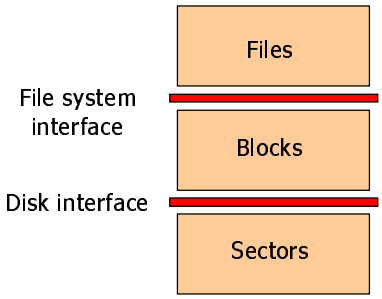
cylinder: all tracks at the same radius, two per platter

Accessing Data from a Disk

- First, the disk arm moves over the right cylinder: *seek time*
- Next, wait for the data to rotate under the disk arm: *rotational delay*
- Finally, read the data from the disk: *transfer time*
- At this point, the data is in the *disk controller* and can be transferred to memory

Disk Interface

- Only two operations on a disk:
  - read block, write block
- Hidden behind the interface:
  - block ↔ sector mappings
- Unknown to the disk:
  - contents of the blocks/sectors

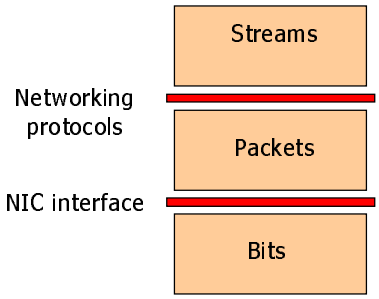


Networks

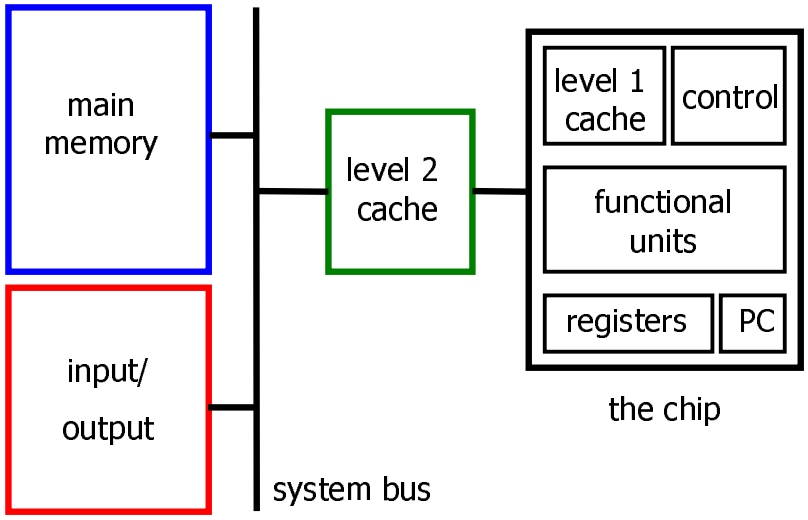
- The device that lets a system connect to a network: *network interface card*
- Listens for data on the network important to this system
- Bundles the bits into packets and signals the OS when a packet is complete
- Also takes packets from OS and sends them as bits on the wire

Networking Interfaces

- OS puts extra packets in to define where stream begins and ends
- NIC puts extra bits in to define where packets begin and end



Communicating with Devices



System Bus

- Lots of variations:
  - How many bits can be sent simultaneously (bus width)?
  - How will access to the bus be controlled?
  - Synchronous (clocked) vs. asynchronous (handshake protocol)

Commands to I/O Devices

- Memory-mapped I/O
  - A special region of memory is set aside for a device
  - Loads and stores to addresses in this region are interpreted as commands to the device
  - Provides easy access control via the memory system
- Special I/O instructions

Device to Processor Comm.

- The device has some information for the processor. Two ways to convey it:
  - Issue an interrupt
  - Wait for the processor to ask for it - *polling*
- Which is better, interrupt-driven I/O or polling? Depends on:
  - time sensitivity of data
  - whether data is expected

Device to Memory Comm.

- *Direct Memory Access (DMA)* allows devices and memory to communicate without involvement of processor
- Processor sets up the transaction
- Device and memory transfer the data
- Device interrupts processor to signal completion
- The processor gets a lot of other work done while transfer is happening

Performance Issues in I/O

- Processors double in speed every 18 months
- Networks double in speed more slowly, perhaps every 3 years
- Disks improve very slowly, because they are limited by mechanical factors

The I/O Bottleneck

- System A: processor speed = 100 MHz; disk transfer takes 10 ms
  - How many clock cycles elapse while disk transfer takes place?
- System B: processor speed = 400 MHz; disk transfer *still* takes 10 ms
  - How many clock cycles now?