Legacy Disk Interfaces

- **ATA - “AT Attachment”**
  - 16 bits of data in parallel
  - 40 or 80-conductor “Ribbon cables”
  - Peak of 133MB/s
  - Two drives per cable

- **SCSI -- Small Computer System Interface**
  - Synonymous with high-end IO
  - Fast bus speeds: up to 160Mhz QDR (four data transfers per clock)
  - Many variants up to SCSI Ultra-640: 640MB/s
  - Scalable: up to 16 devices per SCSI bus.
  - Expensive.
The Serial Revolution

- Wider busses are an obvious way to increased bandwidth
  - But “jitter” and “clock skew” becomes a problem
  - If you have 32 lines in a bus, you need to wait for the slowest one.
  - All devices must use the same clock.
  - This limits bus speeds.
- Lately, high speed serial lines have been replacing wide buses.
High speed serial

- Two wires, but not power and ground
- “low voltage differential signaling”
  - If signal 1 is higher than signal 2, it’s a 1
  - If signal 2 is higher, it’s a 0
  - Detecting the difference is possible at lower voltages, which further increases speed
- Max bandwidth per pair: currently 10Gb/s
- Cables are much cheaper and can be longer and cheaper -- External hard drives.
  - SCSI cables can cost $100s -- and they fail a lot.
Serial interfaces

- **SATA -- Serial ATA**
  - Replaces ATA
  - The logical protocol is the same, but the “transport layer” is serial instead of parallel.
  - Max performance: 600MB/s -- much less in practice.
- **SAS -- Serial attached SCSI**
  - Replace SCSI, Same logical protocol.
PCle -- Peripheral Component Interconnect (express)

- The fastest general-purpose expansion option
  - Graphics cards
  - Network cards
  - High-performance disk controllers (RAID)
- PCle
  - Replace PCI and PCIX
  - PCle busses are actually point-to-point
  - Between 1 and 32 lanes, each of which is a differential pair.
  - Latest version: 1GB/s per lane
  - Max of 32GB/s per card -- I don’t know of any 32 lane cards, but 16 is common.
Hard Disks

- Hard disks are amazing pieces of engineering
  - Cheap
  - Reliable
  - Huge.
Disk Density

1 Tb/square inch
Hard drive Cost

Today at newegg.com: $0.04 GB ($0.00004/MB)
Desktop, 2 TB
The Problem With Disk: It’s Slooooooowww

- on-chip cache
  - KBs
  - Cost: 0.000008 $/MB
  - Access time: < 1ns

- off-chip cache
  - MBs
  - Cost: 0.07 $/MB
  - Access time: 60ns

- main memory
  - GBs
  - Cost: 2.5 $/MB
  - Access time: 5ns

- Disk
  - TBs
  - Cost: 0.000008 $/MB
  - Access time: 10,000,000ns
Why Are Disks Slow?

- They have moving parts :-(
  - The disk itself and the a head/arm
- The head can only read at one spot.
- High end disks spin at 15,000 RPM
  - Data is, on average, 1/2 an revolution away: 2ms
  - Power consumption limits spindle speed
  - Why not run it in a vacuum?
- The head has to position itself over the right “track”
  - Currently about 150,000 tracks per inch.
  - Positioning must be accurate with about 175nm
  - Takes 3-13ms
Making Disks Faster

- Caching
  - Everyone tries to cache disk accesses!
  - The OS
  - The disk controller
  - The disk itself.

- Access scheduling
  - Reordering accesses can reduce both rotational and seek latencies
RAID!

• Redundant Array of Independent (Inexpensive) Disks
• If one disk is not fast enough, use many
  • Multiplicative increase in bandwidth
  • Multiplicative increase in Ops/Sec
  • Not much help for latency.
• If one disk is not reliable enough, use many.
  • Replicate data across the disks
  • If one of the disks dies, use the replica data to continue running and re-populate a new drive.
• Historical foot note: RAID was invented by one of the text book authors (Patterson)
There are several ways of ganging together a bunch of disks to form a RAID array. They are called “levels.”

Regardless of the RAID level, the array appears to the system as a sequence of disk blocks.

The levels differ in how the logical blocks are arranged physically and how the replication occurs.
RAID 0

• Double the bandwidth.
• For an n-disk array, the n-th block lives on the n-th disk.
• Worse for reliability
  • If one of your drives dies, all your data is corrupt-- you have lost every nth block.
RAID 1

- Mirror your data
- 1/2 the capacity
- But, you can tolerate a disk failure.
- Double the bandwidth for reads
- Same bandwidth for writes.
• Stripe your data across a bunch of disks
• Use one bit to hold parity information
  • The number of 1’s at corresponding locations across the drives is always even.
• If you lose on drive, you can reconstruct it from the others.
• Read and write all the disks in parallel.
Solid-state disks (SSDs)

- Use NAND flash memory instead of a spinning disk
- They are everywhere
  - iPods,
  - Laptops,
  - USB keys
  - Embedded systems
  - Digital cameras.
  - Data centers (sometimes)
Flash’s Internal Structure

- Flash stores bit on a “floating gate” in a floating gate transistor.
- The gate is electrically isolated, so charge stays put.
- Charge can be pulled on and off the gate using large voltages on the terminals of the transistor.
- The charge on the gate affects the transistors switching characteristics, which allows us to read the bits out.

Transistor

Floating gate transistor

Floating gate

Transistor
The Flash Chain

- Floating gate transistors are arranged in series as “chains”
- This allows for very high density: \(4F^2/\text{bit}\)
- DRAM is \(17F^2/\text{bit}\)
- Makes reading and writing slow -- all the other gates are in the way
Flash Blocks

- Many parallel chains form a block.
- A slice across the chains is a page.
- Read/Program operations affect one bit in each chain
- Erases effect all the bits in a chain.
NAND Flash

- Three operations
  - Erase a block (very slow)
  - Program a page (slower)
  - Read a page (fast)
- SLC – one bit per xtr
  - Fast, less dense
- MLC – two bits per xtr
  - Denser, slower, cheaper
- Reliability decreases with program/erase cycles
Individual Flash Chip Performance

• Flash is very slow for a memory.
• Transfer on and off the chip: 40MHz by 8 bits
  • Silly historical reasons. Currently a move is underway to 133MHz by 16 bits
  • DRAM is currently ~1GHz
• Operation latencies
  • 25-35us for reads
  • 200-2000us for programs
  • 1-3ms for erase
Reliability

- Flash wears out with use
  - Break down in the insulation around the floating gates lets charge leak off the gate.
- For MLC devices -- 10k program/erase cycles
- For SLC devices -- 100k program/erase cycles
- You can “burn a hole” in a flash chip in about 12 hours.
Wear Leveling

- SSDs must spread out program/erase operations evenly across the flash chips.
- They maintain an table that maps “Logical block addresses” (i.e., disk block addresses) to flash pages/blocks
- This “Flash translation layer” reduces performance and adds complexity.
  - SSD performance can be erratic.
- FTLs also provide error correction to recover from bit errors (which can be frequent, esp. for MLC)
- This is the key differentiator between SSDs
SSDs vs HDD

- **Expensive**
  - SSD - $3/Gig (80GB Intel SSD)
  - Disk - $0.08/Gig (2TB seagate drive)

- **Fast**
  - IOPS
    - Random IO operations per second (IOPS)
    - SSD -- 3000/s for writes, 35,000 for reads (says Intel)
    - Disk -- 1/15ms = 66/s
  - BW MB/s
    - SSD -- 170MB/s write; 250MB/s read (max)
    - Disk -- 125MB/s (max)
  - Latency
    - SSD -- 75 microseconds for reads; intel won’t say for writes(!) probably 100s-1000s of microseconds
    - Disk -- 4-8ms

- **Low power**
  - SSD -- 2.4W max 0.06W idle
  - Disk -- 6.56W active; 5W idle
  - How often is your disk idle? **They are idle a lot**