Section: Mass Storage
Tapes, Disks, CDs, DVDs, RAID etc.

Deniz Altinbuken

CS 3410, Spring 2010

Computer Science

Cornell University

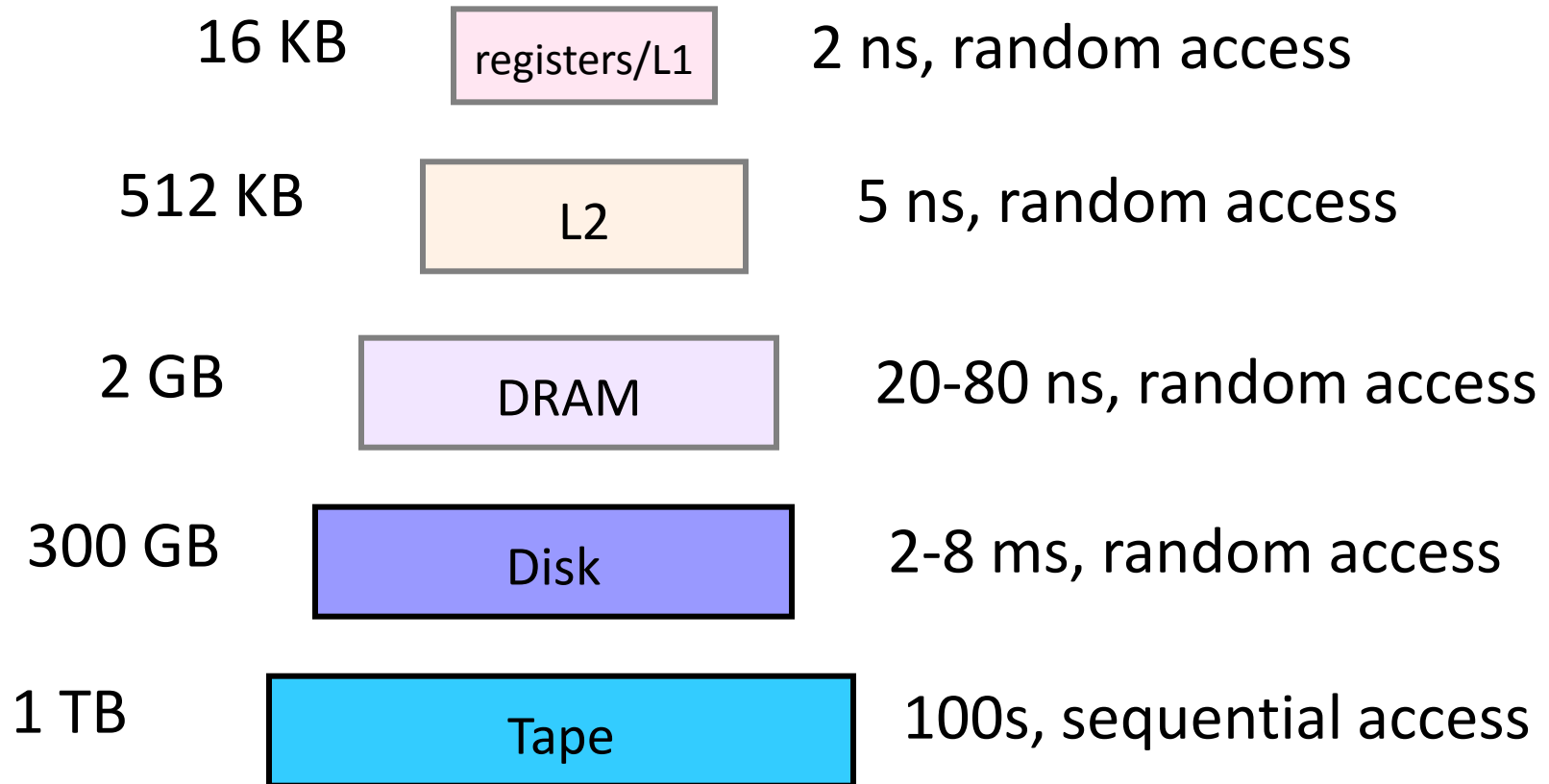adapted from slides of Kevin Walsh and Emin Gun Sirer

Challenge

- How do we store lots of data for a long time
  - Disk (Hard disk, floppy disk, …)
  - Tape (cassettes, backup, VHS, …)
  - CDs/DVDs

I/O System Characteristics

- Dependability is important
  - Particularly for storage devices
- Performance measures
  - Latency (response time)
  - Throughput (bandwidth)
  - Desktops & embedded systems
    - Mainly interested in response time & diversity of devices
  - Servers
   - Mainly interested in throughput & expandability of devices

Memory Hierarchy

|		
16 KB	registers/L1	2 ns, random access
512 KB	L2	5 ns, random access
2 GB	DRAM	20-80 ns, random access
300 GB	Disk	2-8 ms, random access
1 TB	Tape	100s, sequential access

Tapes

- Same basic principle for 8-tracks, cassettes, VHS, …

  - Ferric Oxide Powder: ferromagnetic material

  - During recording, the audio signal is sent through the coil of wire to create a magnetic field in the core.

  - During playback, the motion of the tape creates a varying magnetic field in the core and therefore a signal in the coil.
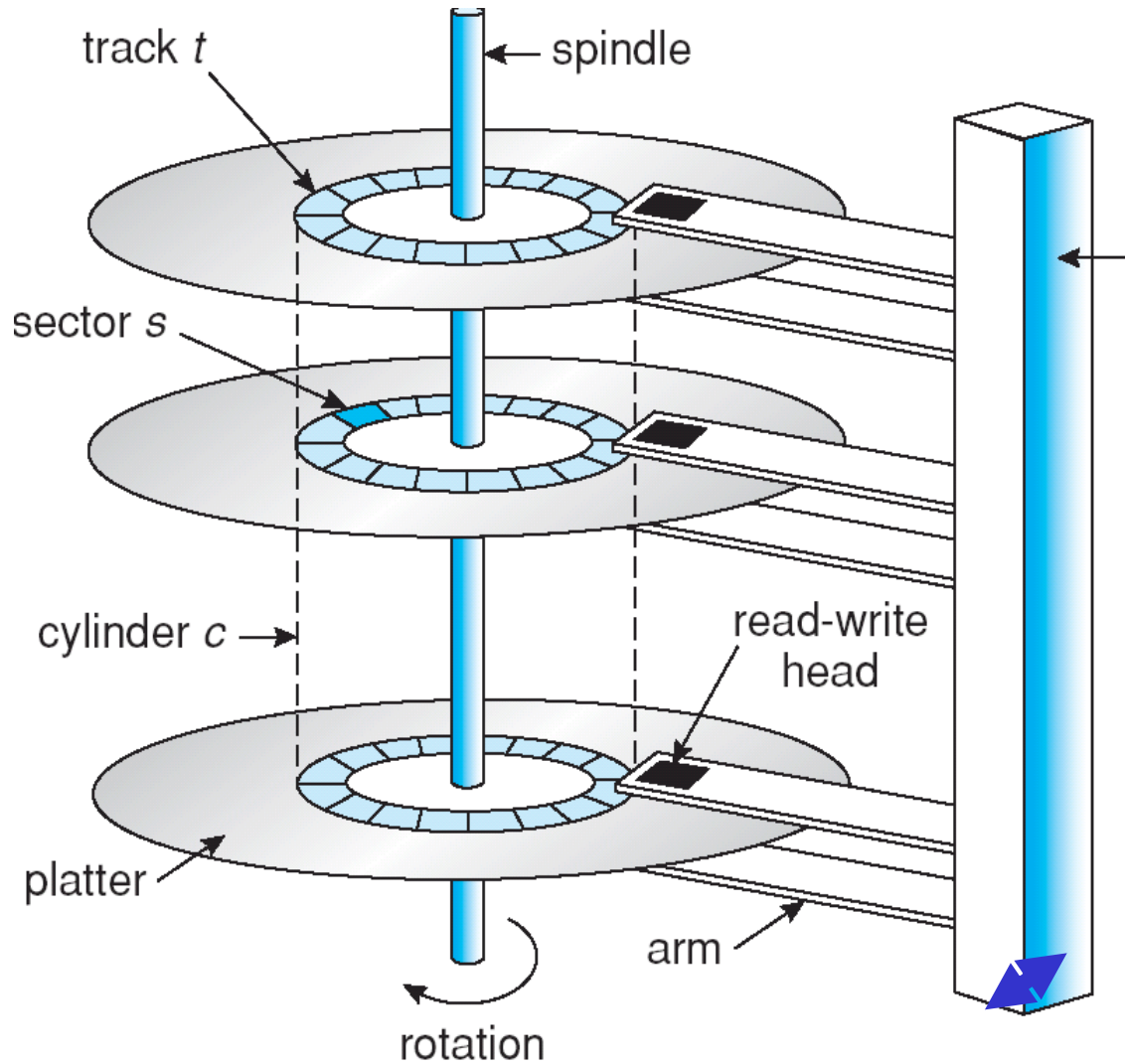
0   0   1   0   1   0   1   0   1
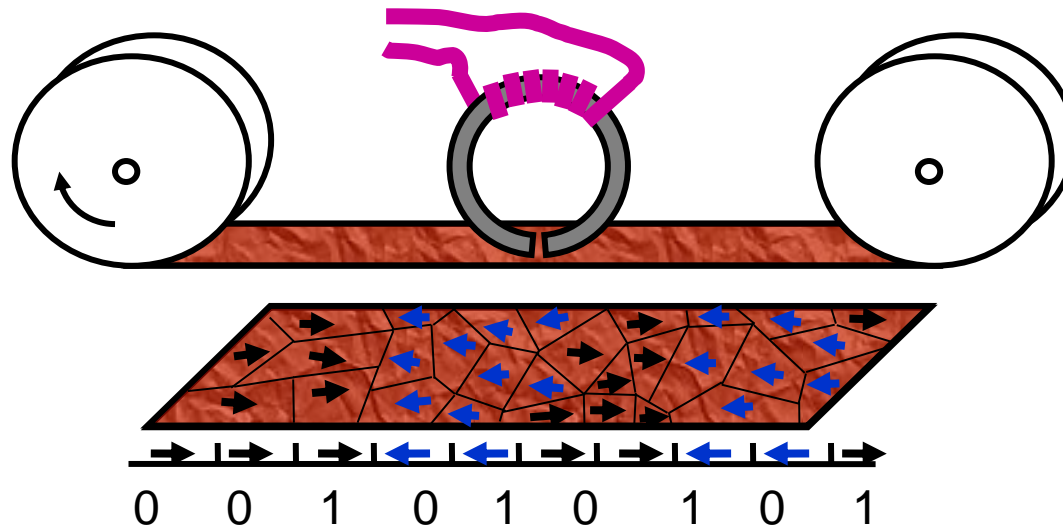
Disks & CDs

- Disks use same magnetic medium as tapes
  - concentric rings (not a spiral)
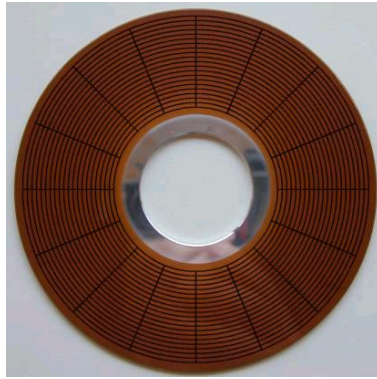
- CDs & DVDs use optics and a single spiral track

Disk Physics

track *t* ← spindle

sector *s*

cylinder *c* →

read-write head

platter

arm

rotation

Typical parameters :
- 1 spindle
- 1 arm assembly
- 1-4 platters
- 1-2 sides/platter
- 1 head per side
  (but only 1 active head at a time)
- 700-20480 tracks/surface
- 16-1600 sectors/track

Disk Accesses

- Accessing a disk requires:
  - specify sector: C (cylinder), H (head), and S (sector)
  - specify size: number of sectors to read or write
  - specify memory address

- Performance:
  - seek time: move the arm assembly to track
  - Rotational delay: wait for sector to come around
  - transfer time: get the bits off the disk
  - Controller time: time for setup

Track

Sector

Seek Time
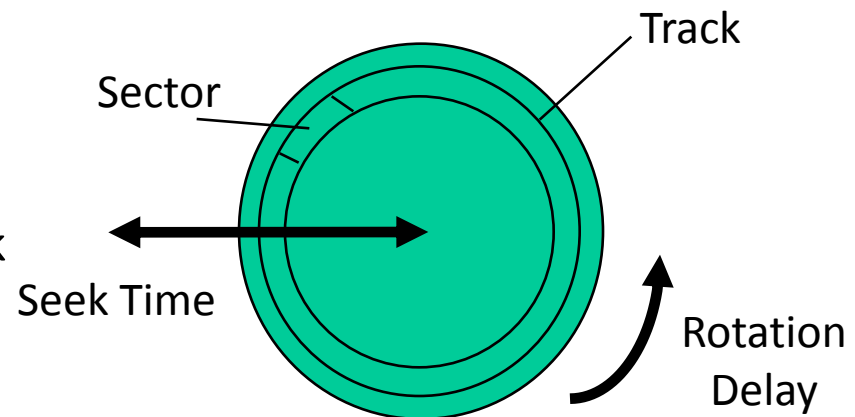
Rotation Delay

Example

- Average time to read/write 512-byte sector
  - Disk rotation at 10,000 RPM
  - Seek time: 6ms
  - Transfer rate: 50 MB/sec
  - Controller overhead: 0.2 ms
- Average time:
  - Seek time + rotational delay + transfer time + controller overhead
  - 6ms + 0.5 rotation/(10,000 RPM) + 0.5KB/(50 MB/sec) + 0.2ms
  - 6.0 + 3.0 + 0.01 + 0.2 = 9.2ms

Disk Access Example

- If actual average seek time is 2ms
 - Average read time = 5.2ms

Disk Scheduling

- Goal: minimize seek time
  - secondary goal: minimize rotational latency
- FCFS (First come first served)
- Shortest seek time
- SCAN/Elevator
  - First service all requests in one direction
  - Then reverse and serve in opposite direction
- Circular SCAN
  - Go off the edge and come to the beginning and start all over again

FCFS

queue = 98, 183, 37, 122, 14, 124, 65, 67
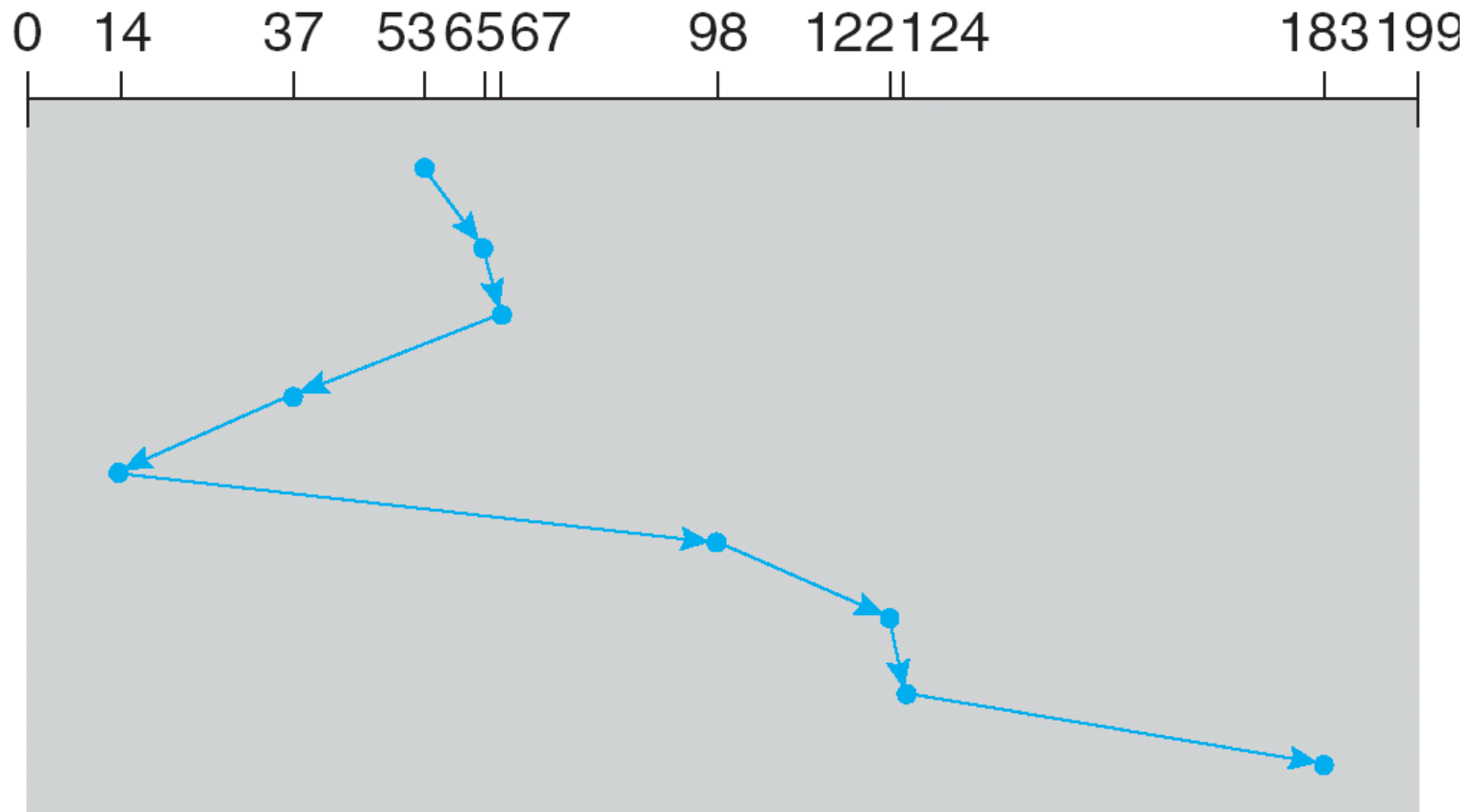head starts at 53
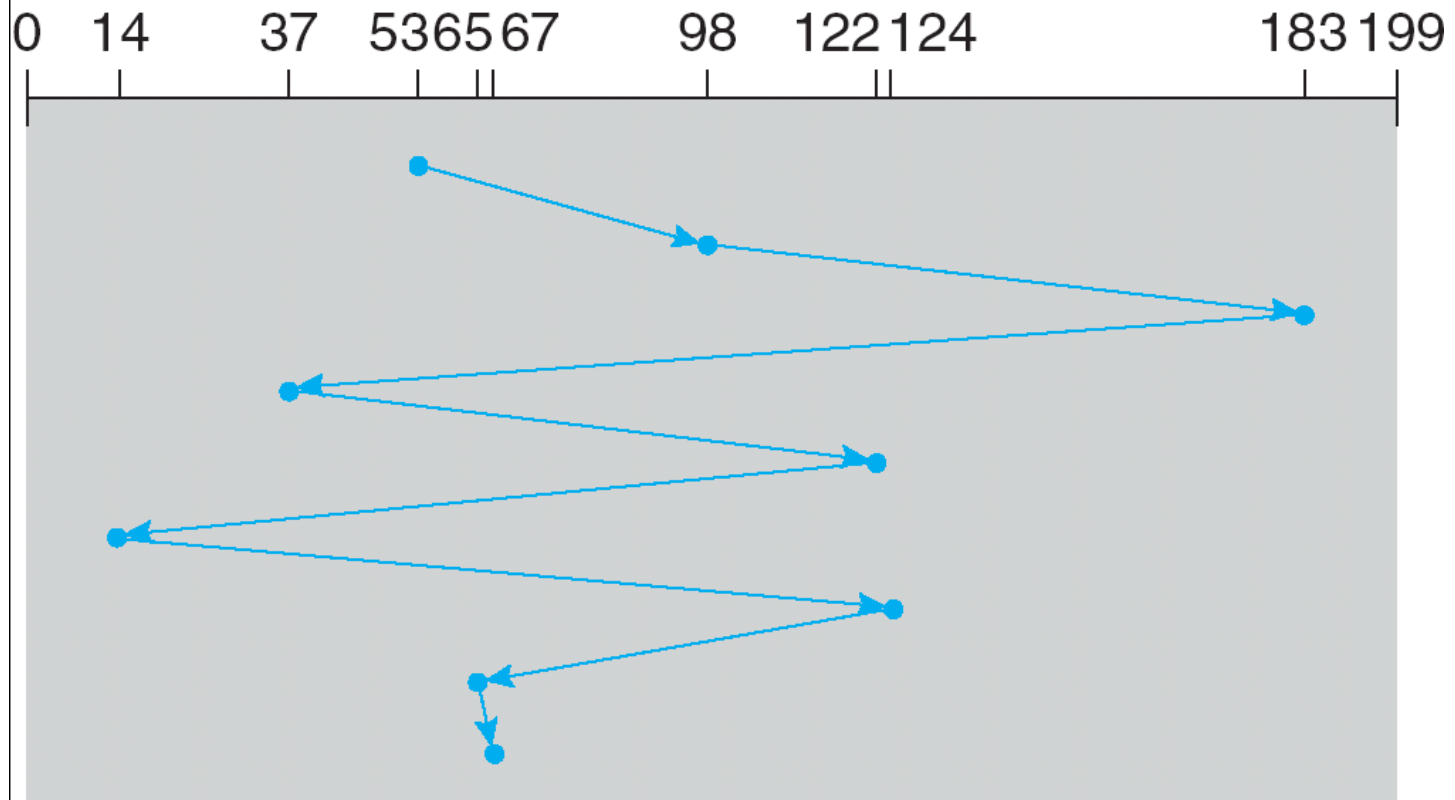
SSTF

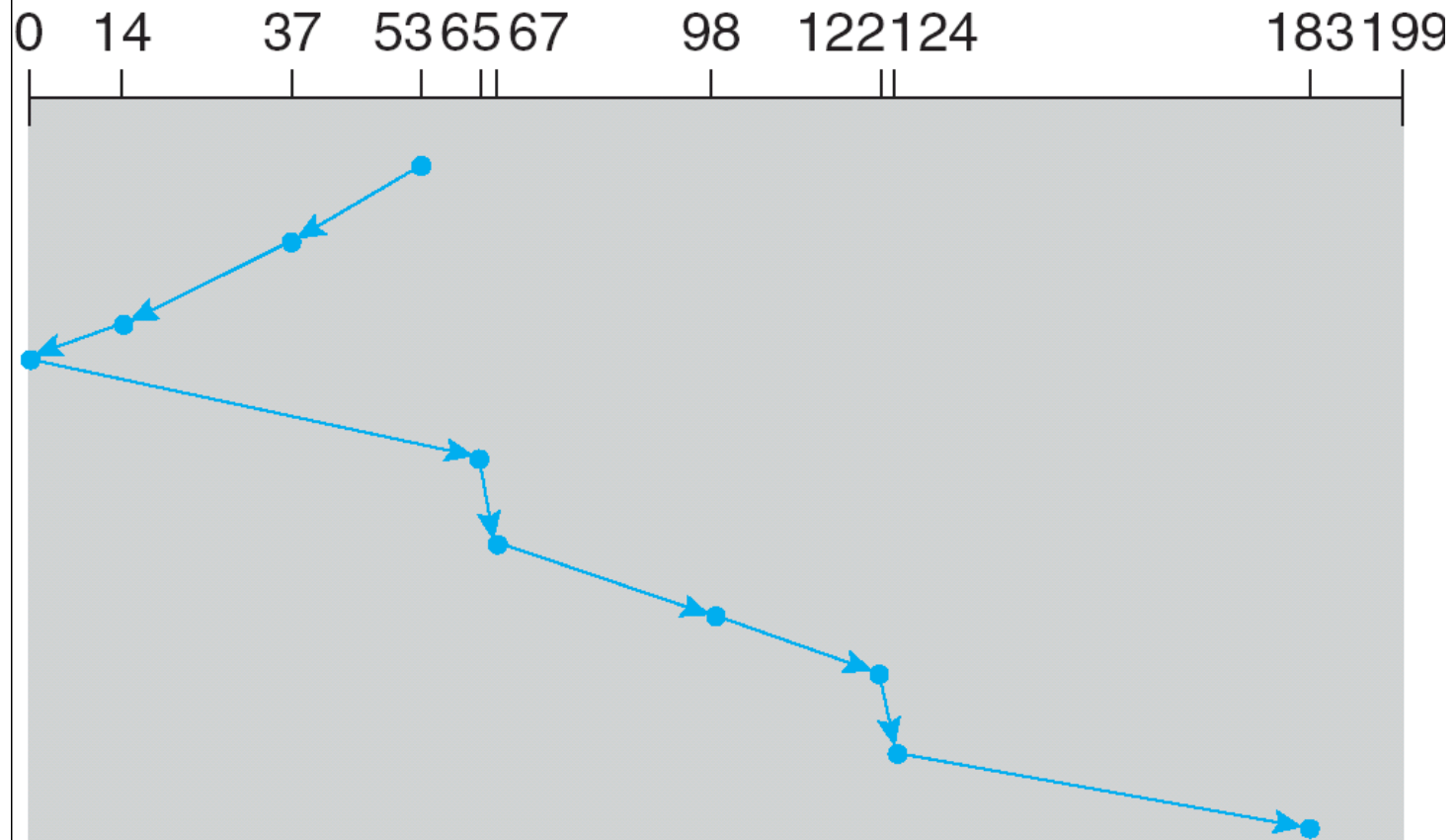queue = 98, 183, 37, 122, 14, 124, 65, 67

head starts at 53

SCAN

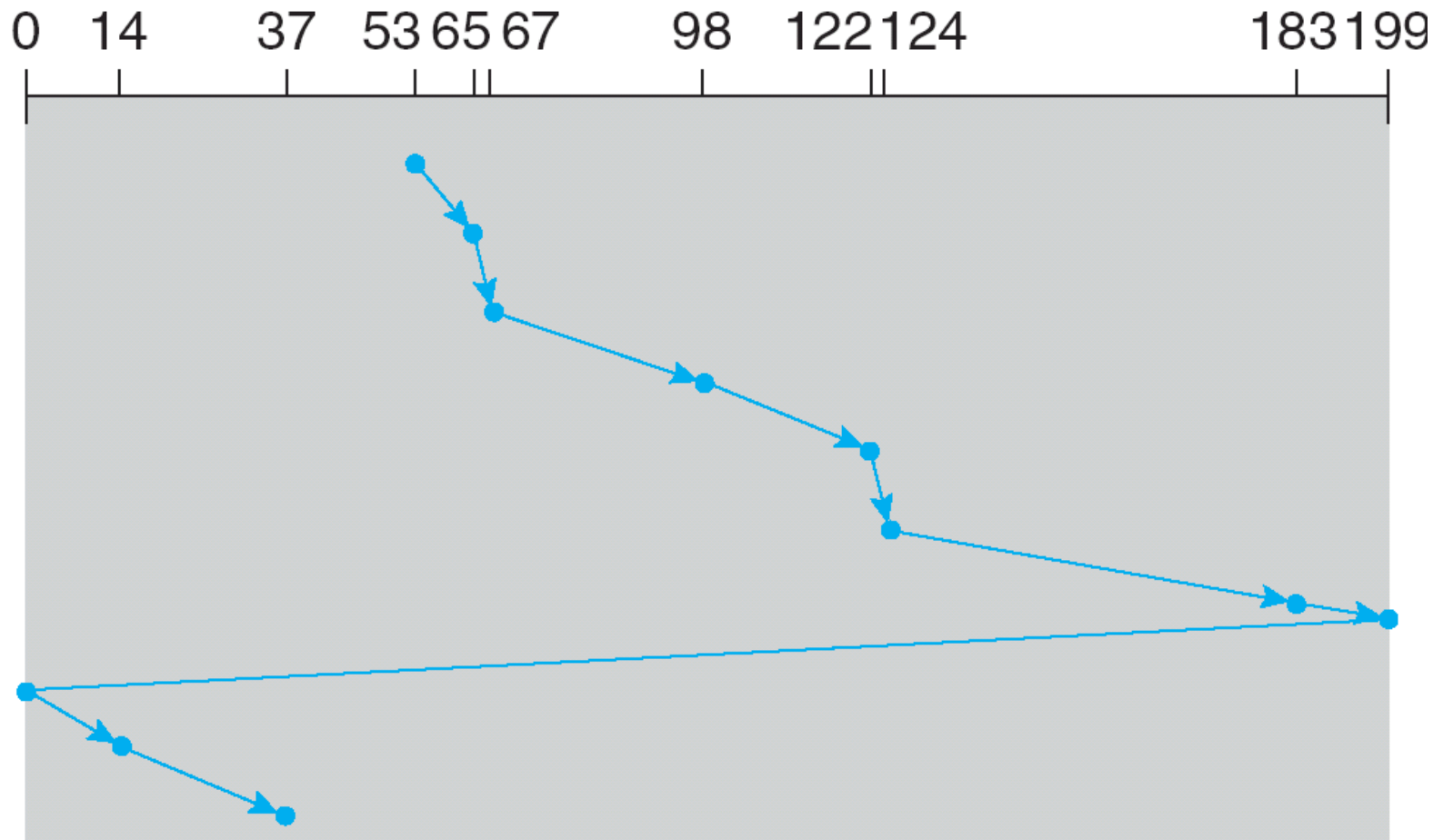queue = 98, 183, 37, 122, 14, 124, 65, 67

head starts at 53

0   14   37   53 65 67   98   122 124   183 199

C-SCAN

queue = 98, 183, 37, 122, 14, 124, 65, 67

head starts at 53

Disk Geometry: LBA

- New machines use *logical block addressing* instead of CHS

  - machine presents illusion of an array of blocks, numbered 0 to N

- Modern disks…

  - have varying number of sectors per track

    - roughly constant data density over disk

    - varying throughput over disk

  - remap and reorder blocks (to avoid defects)

  - completely obscure their actual physical geometry

  - have built-in caches to hide latencies when possible (but being careful of persistence requirements)

  - have internal software running on an embedded CPU

Flash Storage

- Nonvolatile semiconductor storage
  - 100× – 1000× faster than disk
  - Smaller, lower power
  - But more $/GB (between disk and DRAM)
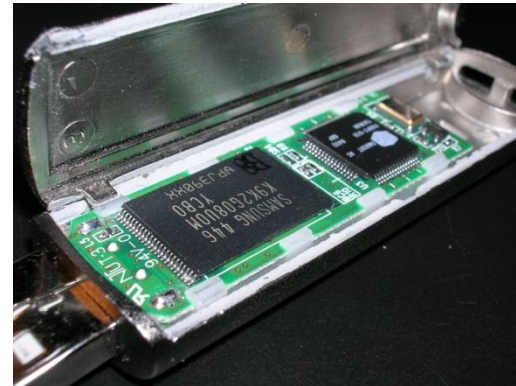
Flash Types

- NOR flash: bit cell like a NOR gate

  - Random read/write access

  - Used for instruction memory in embedded systems

- NAND flash: bit cell like a NAND gate

  - Denser (bits/area), but block-at-a-time access

  - Cheaper per GB

  - Used for USB keys, media storage, …

- Flash bits wears out after 1000's of accesses

  - Not suitable for direct RAM or disk replacement

- Flash has unusual interface

  - can only "reset" bits in large blocks

I/O vs. CPU Performance

- Amdahl's Law
  - Don't neglect I/O performance as parallelism increases compute performance
- Example
  - Benchmark takes 90s CPU time, 10s I/O time
  - Double the number of CPUs/2 years
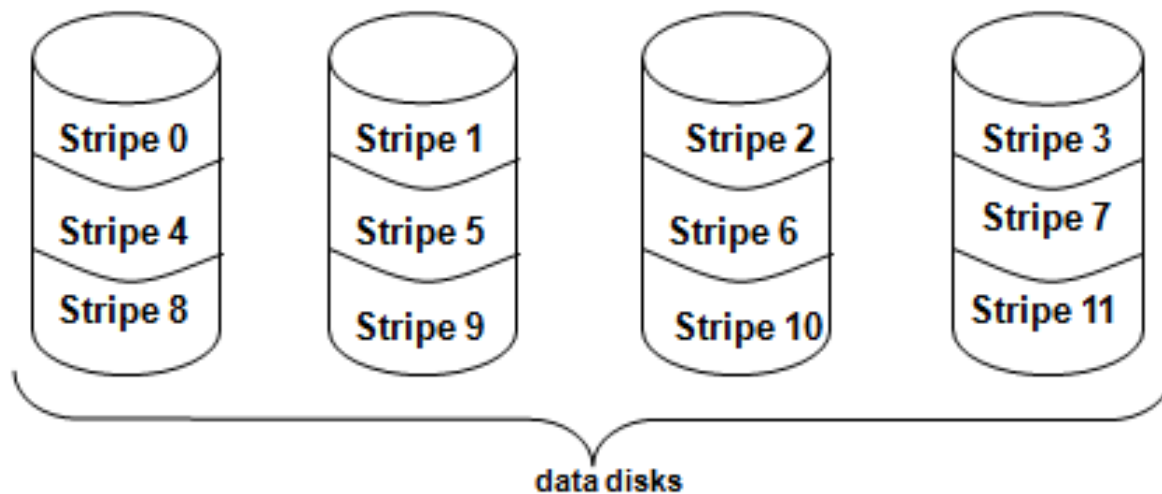   - I/O unchanged

Year	CPU time	I/O time	Elapsed time	% I/O time
now	90s	10s	100s	10%
+2	45s	10s	55s	18%
+4	23s	10s	33s	31%
+6	11s	10s	21s	47%

RAID

- Redundant Arrays of Inexpensive Disks

- Big idea:

  - Parallelism to gain performance
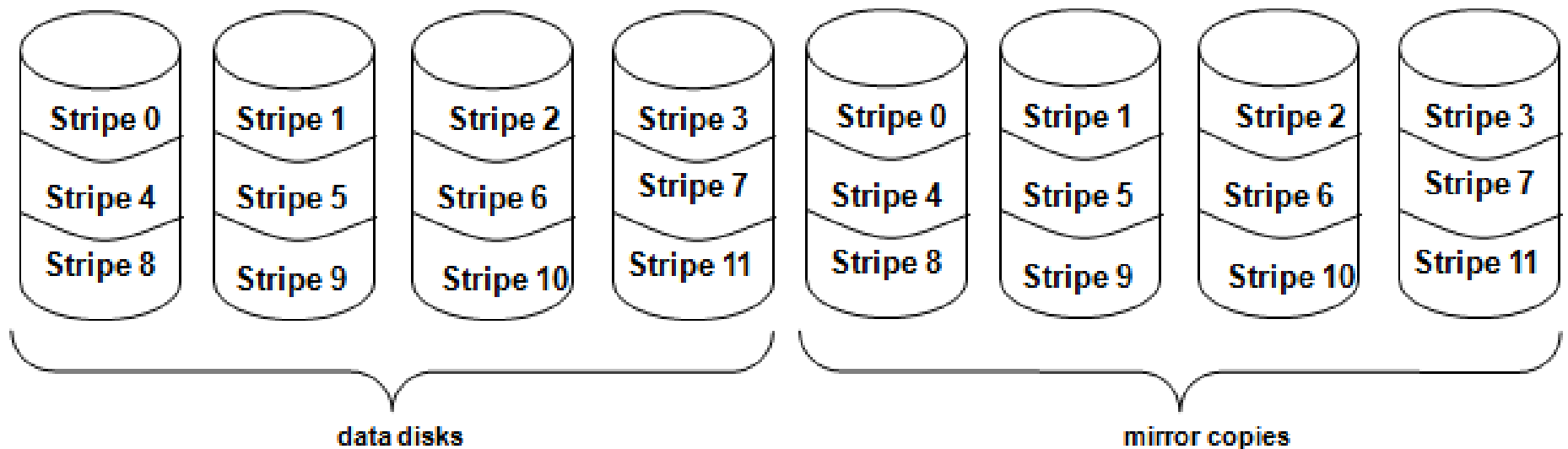
  - Redundancy to gain reliability

Raid 0

- Striping
  - Non-redundant disk array!

Raid 1

- Mirrored Disks!
 - More expensive
  - On failure use the extra copy

Stripe 0	Stripe 1	Stripe 2	Stripe 3	Stripe 0	Stripe 1	Stripe 2	Stripe 3
Stripe 4	Stripe 5	Stripe 6	Stripe 7	Stripe 4	Stripe 5	Stripe 6	Stripe 7
Stripe 8	Stripe 9	Stripe 10	Stripe 11	Stripe 8	Stripe 9	Stripe 10	Stripe 11

data disks   mirror copies

Raid 2-3-4-5-6

- ## Bit Level Striping and Parity Checks!

  - ### As level increases:

    - #### More guarantee against failure, more reliability
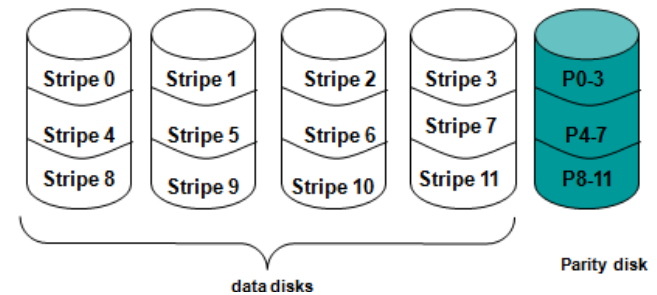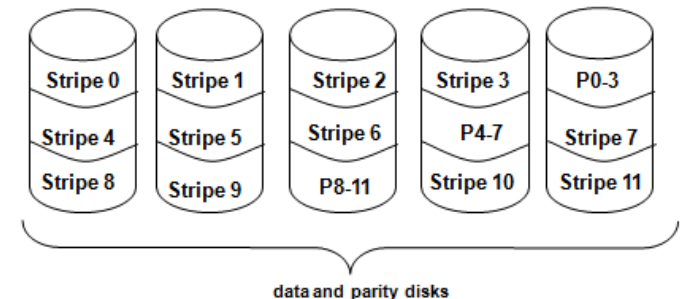
    - #### Better read/write performance
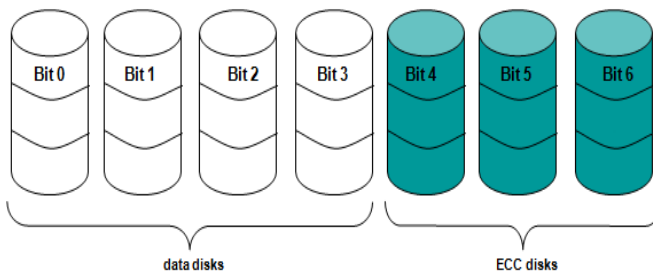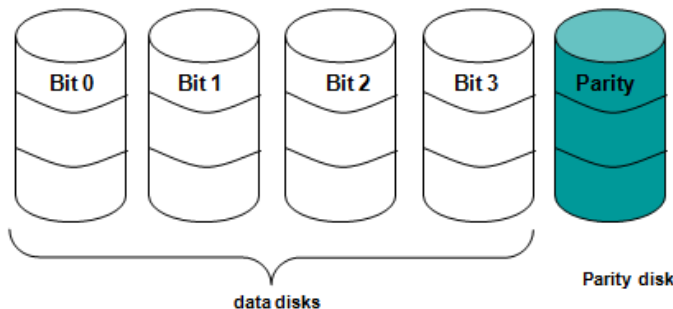
Summary

- Disks provide nonvolatile memory
- I/O performance measures
  - Throughput, response time
  - Dependability and cost very important
- RAID
  - Redundancy for fault tolerance and speed