CS / EE 320
Computer Organization and Assembly Language

Lecture 25
Spring 2025

Shahid Masud

Topics: Cache/Memory Update Write Strategies, Cache Performance, Multi Level Caches, CD ROM Technology

Topics

- Cache Writes – Write Back and Write Through schemes
- Examples of calculating different parameters in Cache Design
- Some Examples of Cache Performance
- Introduction to Multi-level Caches
- Examples of Cache Performance
- ---
- Maybe CD ROM Technology

Quiz 5 Next Lecture

Dealing with CPU to Write to Memory and Caches

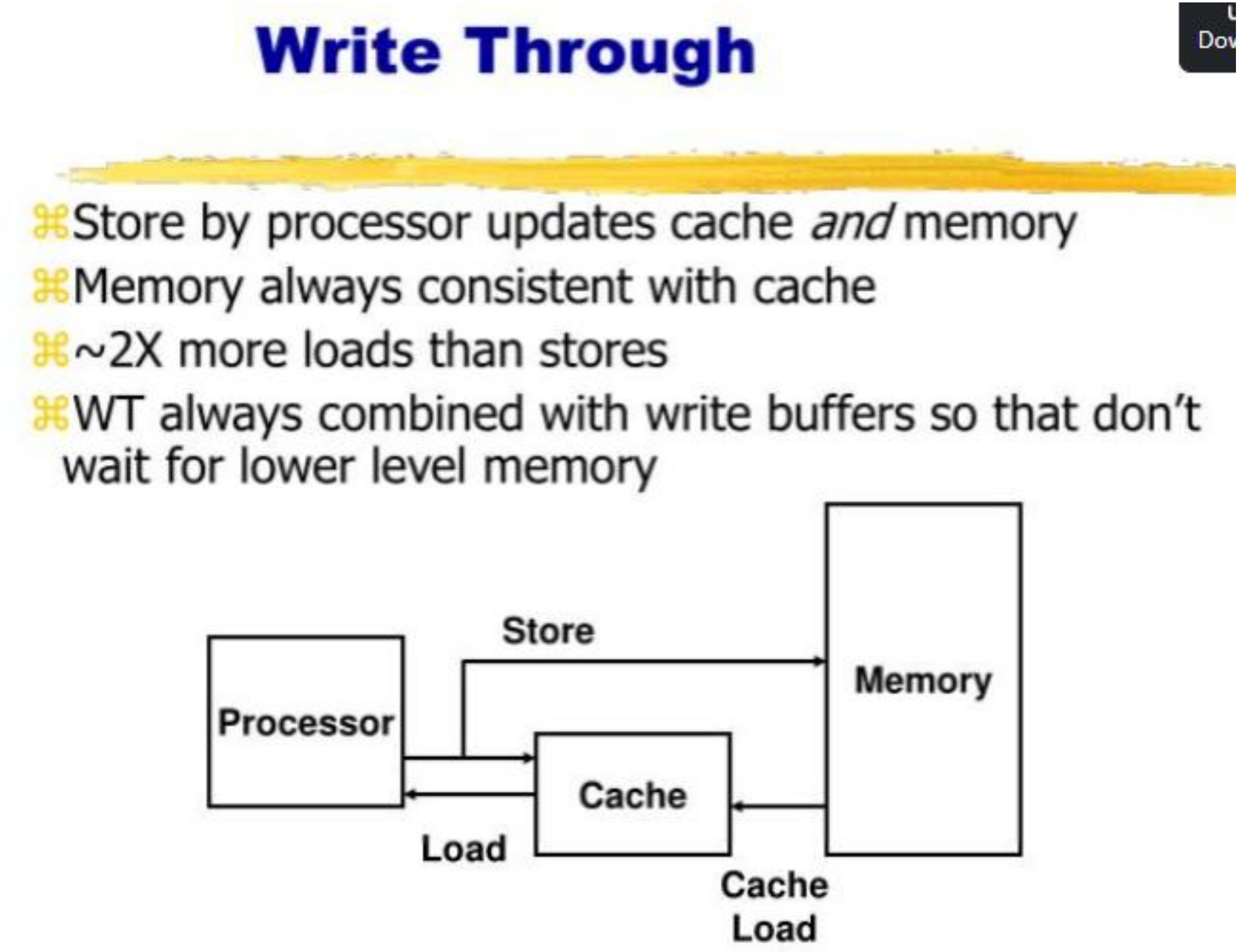
Write Policy

- Must not overwrite a cache block unless main memory is up to date
- Multiple CPUs may have individual caches
- I/O may address main memory directly

Write-through scheme

- All writes go to main memory as well as cache
- Multiple CPUs can monitor main memory traffic to keep local (to CPU) cache up to date
- Lots of traffic
- Slows down writes

Write Through - Diagram



Write Through

- Store by processor updates cache *and* memory
- Memory always consistent with cache
- ~2X more loads than stores
- WT always combined with write buffers so that don’t wait for lower level memory

Write-Through

- On data-write hit, could just update the block in cache
  - But then cache and memory would be inconsistent
- Write through: also update memory
- But makes writes take longer
  - e.g., if base CPI = 1, 10% of instructions are stores, write to memory takes 100 cycles
    - Effective CPI = 1 + 0.1×100 = 11
- Solution: write buffer
  - Holds data waiting to be written to memory
  - CPU continues immediately
    - Only stalls on write if write buffer is already full

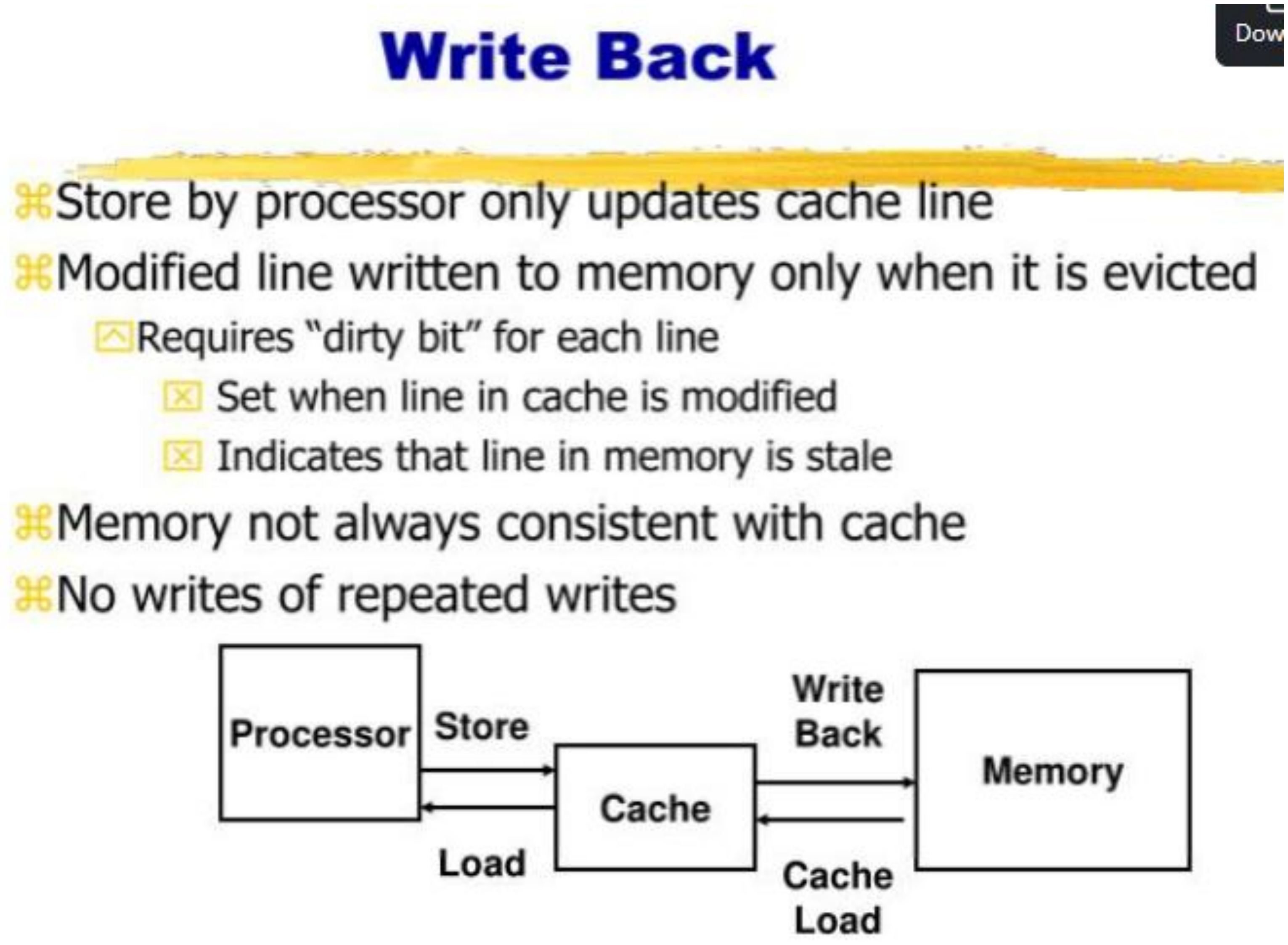
Write-back

- Updates initially made in cache only
- Update bit for cache slot is set when update occurs
- If block is to be replaced, write to main memory only if update bit is set
- Other caches get out of sync
- I/O must access main memory through cache
- N.B. 15% of memory references are writes

Write-Back

- Alternative: On data-write hit, just update the block in cache
  - Keep track of whether each block is dirty
- When a dirty block is replaced
  - Write it back to memory
  - Can use a write buffer to allow replacing block to be read first

Write Back Diagram



Write Back

- Store by processor only updates cache line
- Modified line written to memory only when it is evicted
  - Requires "dirty bit" for each line
    - Set when line in cache is modified
    - Indicates that line in memory is stale
- Memory not always consistent with cache
- No writes of repeated writes

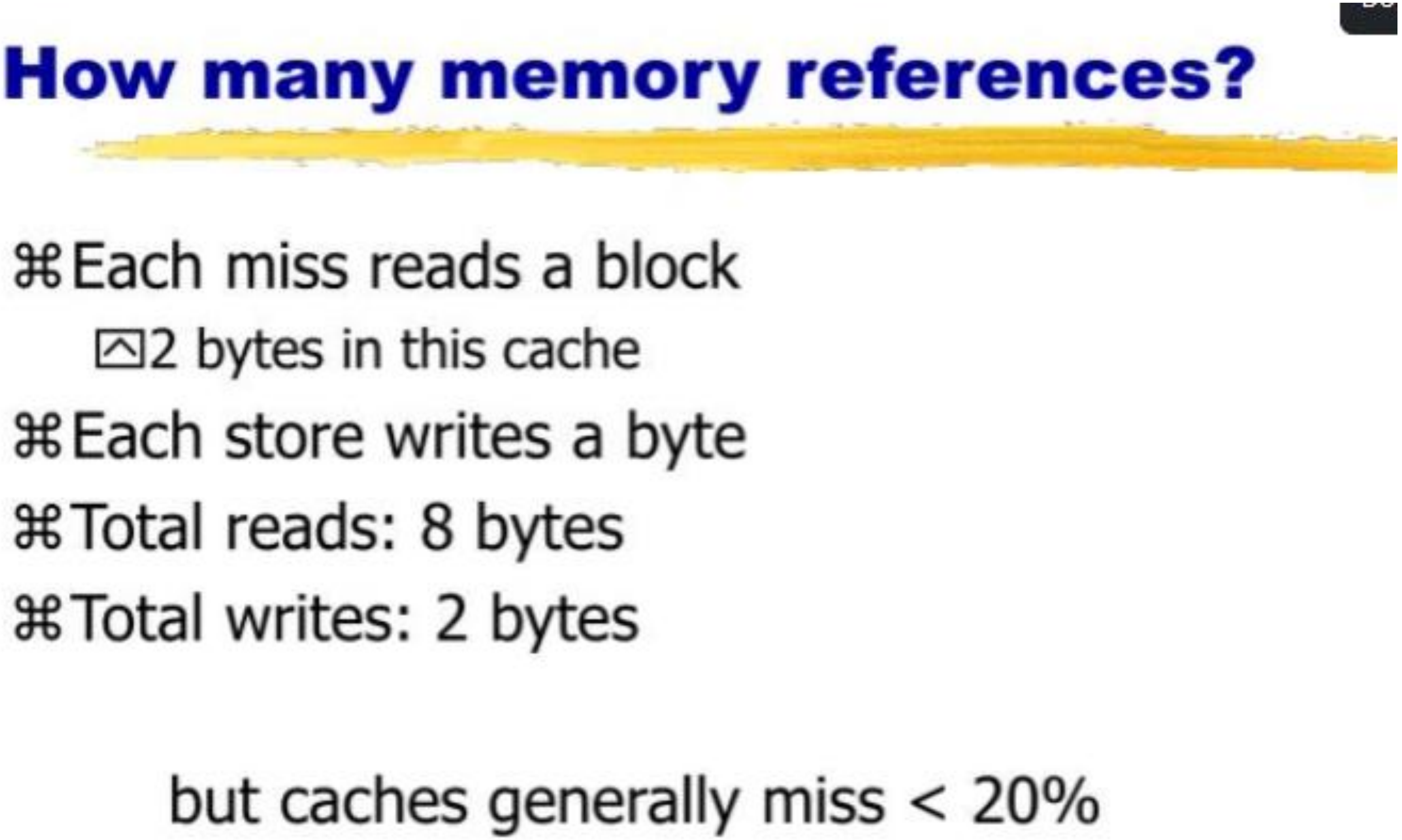
Write Miss

- What should happen on a write miss?
- Alternatives for write-through
  - Allocate on miss: fetch the block
  - Write around: don’t fetch the block
    - Since programs often write a whole block before reading it (e.g., initialization)
- For write-back
  - Usually fetch the block

Cache Design Trade-offs

Design change	Effect on miss rate	Negative performance effect
Increase cache size	Decrease capacity misses	May increase access time
Increase associativity	Decrease conflict misses	May increase access time
Increase block size	Decrease compulsory misses	Increases miss penalty. For very large block size, may increase miss rate due to pollution.

Number of Memory References



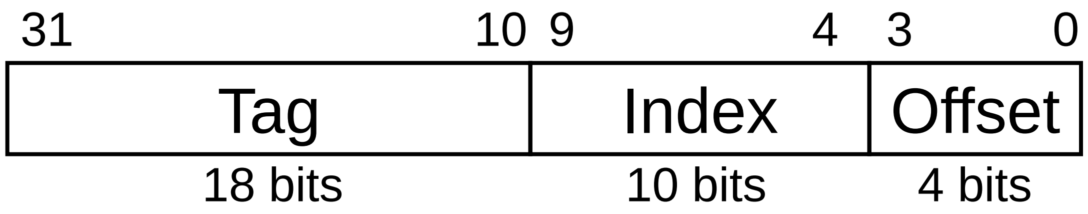
How many memory references?

- ⌘Each miss reads a block
    - ⍓2 bytes in this cache
- ⌘Each store writes a byte
- ⌘Total reads: 8 bytes
- ⌘Total writes: 2 bytes

but caches generally miss < 20%

Cache Controller

- Example cache characteristics
  - Direct-mapped, write-back, write allocate
  - Block size: 4 words (16 bytes)
  - Cache size: 16 KB (1024 blocks)
  - 32-bit byte addresses
  - Valid bit and dirty bit per block
  - Blocking cache
    - CPU waits until access is complete



31 … 10	9 … 4	3 … 0
Tag	Index	Offset
18 bits	10 bits	4 bits

- **Multi Level Caches**
- **Performance of Cache and Memory System**
- **Examples of Cache Performance**

Example 1 of Cache Performance

Question

Given that:

T1 is access time from Cache = 0.01 µsec

T2 is access time from Main Memory = 0.1 µsec

95% accesses are from Cache

5% accesses are from Main Memory

Find the Average Time seen by CPU to Access one word?

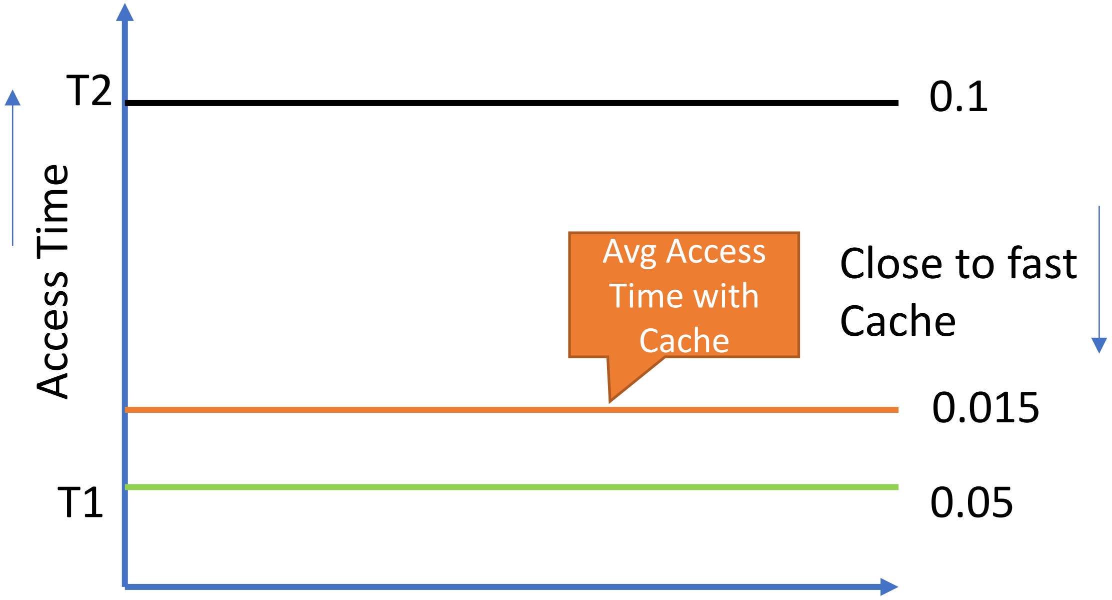
Solution

Average time to access one word:

Tavg= (0.95 x 0.01) + (0.5 x 0.1)

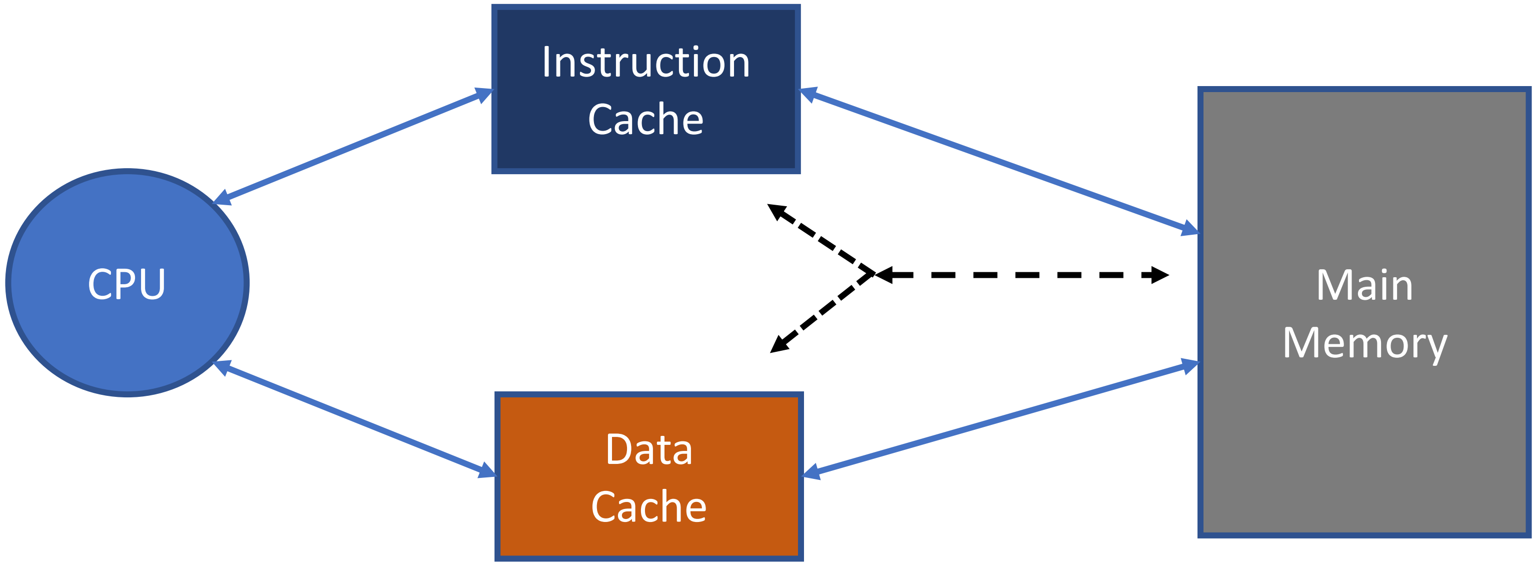
Tavg= 0.095 + 0.0055

Tavg= 0.015 µsec



Split Caches

Separate Cache for Instructions and Data to take advantage of pipelining
In case the program has too many localized accesses to data only or instructions only,
Then there may be no performance improvement

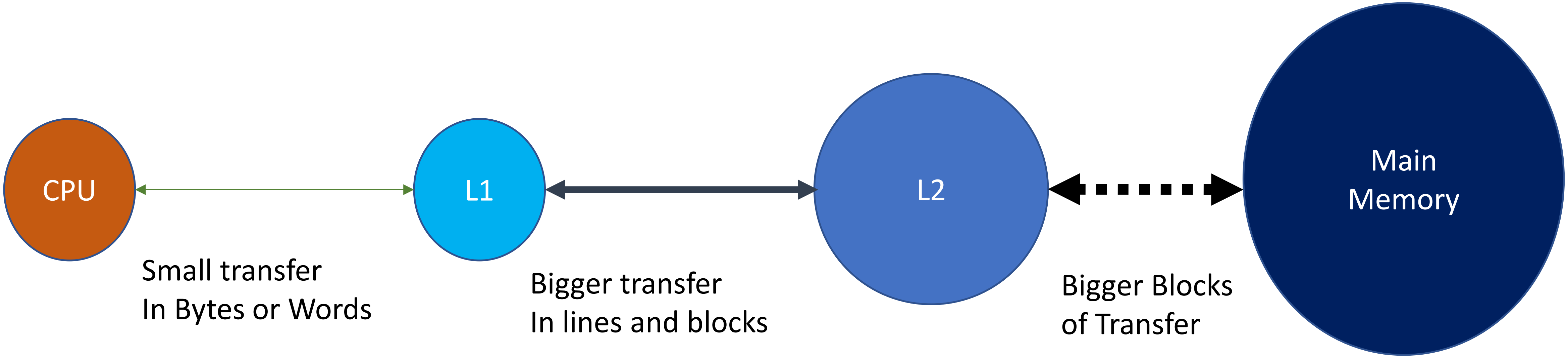


Multilevel Caches

- Primary cache attached to CPU
  - Small, but fast
- Level-2 cache services misses from primary cache
  - Larger, slower, but still faster than main memory
- Main memory services L-2 cache misses
- Some high-end systems include L-3 cache

Multi Level Caches

Main purpose is to reduce Miss Penalty

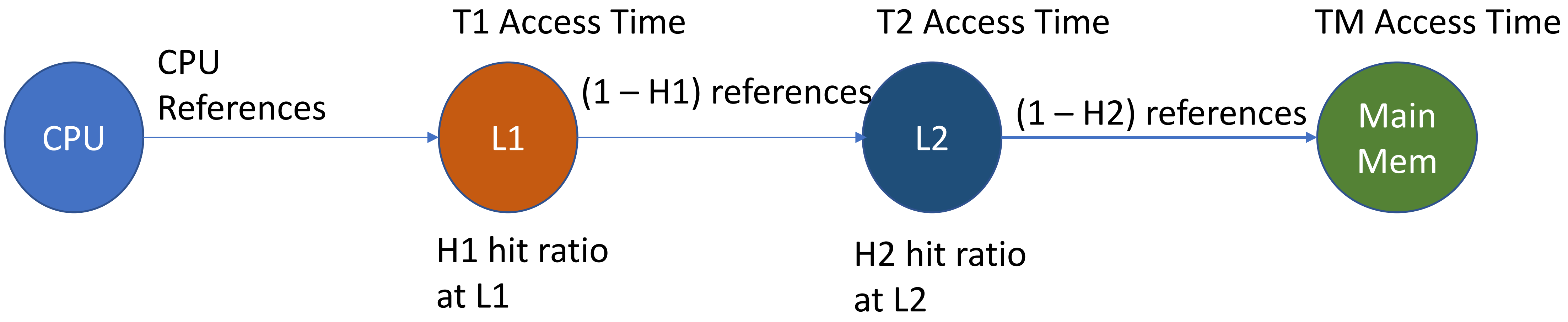


Small transfer
In Bytes or Words

Bigger transfer
In lines and blocks

Bigger Blocks
of Transfer

Multi Level Caches - Performance



Average Access Time in Multilevel Cache:

Tavg = T1 + (1 – H1) x T2 + (1 – H1) x (1 – H2) x TM

(1 – H1) = **miss rate at L1** = MRL1

Tavg = T1 + MRL1 x T2 + MRL1 x MRL2 x TM

MRL1 x MRL2 = Global Miss Rate at L2

Example 1, find Tavg in 2 level cache

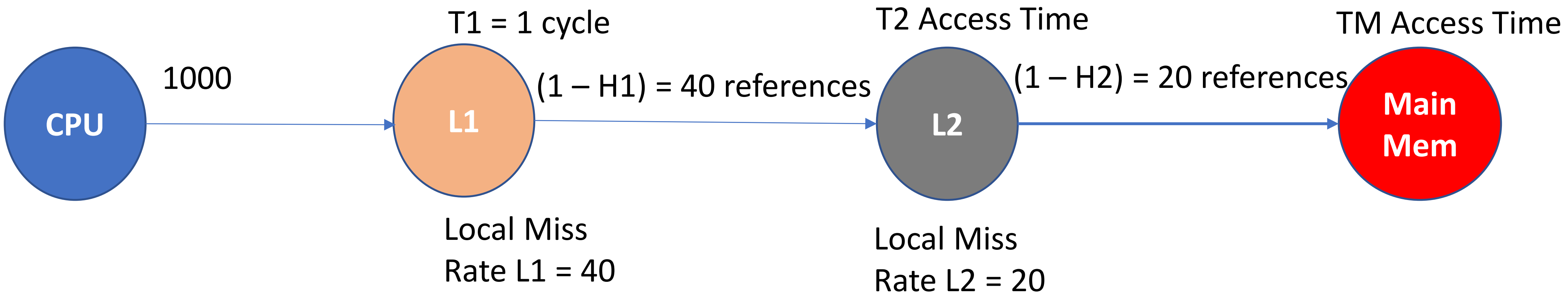
- CPU generates 100 memory references. 80 are hit in L1, 20 are miss in L1. 16 are hit in L2 and 4 are miss in L2. Access time of L1 is 1 cycle. Access time of L2 is 10 cycles. Access time of Main Mem is 100 cycle. Find average memory access time Tavg?

- Tavg = T1 + MRL1 x T2 + MRL1 x MRL2 x TM
- =1 + (20/100) x 10 + [(20/100)x(4/20)] x 100
- =1 + 2 + 4 = 6 cycles Avg Access time

Note:
MRL1 = Miss Rate at Cache L1

Example 2, find Tavg in 2 level cache

- CPU generates 1000 memory references. Number of Miss in L1 is 40. Number of miss in L2 is 20. Access time of L1 is 1 cycle. Access time of L2 is 10 cycle. Find Tavg when Miss Penalty of L2 is 100 cycle.



Tavg = T1 + MRL1 x T2 + MRL1 x MRL2 x TM

=1 + (40/1000) x 10 + [(40/1000) x (20/40)] x 100

=1 + 0.4 + 2 = 3.4 cycles Avg Access time

Measuring Cache Performance

- Components of CPU time
  - Program execution cycles
    - Includes cache hit time
  - Memory stall cycles
    - Mainly from cache misses
- With simplifying assumptions:

$$\text{Memory stall cycles}$$

$$= \frac{\text{Memory accesses}}{\text{Program}} \times \text{Miss rate} \times \text{Miss penalty}$$

$$= \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Misses}}{\text{Instruction}} \times \text{Miss penalty}$$

Cache Performance Example

- Given
  - I-cache miss rate = 2%
  - D-cache miss rate = 4%
  - Miss penalty = 100 cycles
  - Base CPI (ideal cache) = 2
  - Load & stores are 36% of instructions
- Miss cycles per instruction
  - I-cache: 0.02 × 100 = 2
  - D-cache: 0.36 × 0.04 × 100 = 1.44
- Actual CPI = 2 + 2 + 1.44 = 5.44
  - Ideal CPU is 5.44/2 =2.72 times faster

Average Memory Access Time

- Hit time is also important for performance
- Average memory access time (AMAT)
  - AMAT = Hit time + Miss rate × Miss penalty
- Example
  - CPU with 1ns clock, hit time = 1 cycle, miss penalty = 20 cycles, I-cache miss rate = 5%
  - AMAT = 1 + 0.05 × 20 = 2ns
    - 2 cycles per instruction

Performance Summary

- When CPU performance increased
  - Miss penalty becomes more significant
- Decreasing base CPI
  - Greater proportion of time spent on memory stalls
- Increasing clock rate
  - Memory stalls account for more CPU cycles
- Can’t neglect cache behavior when evaluating system performance

Multilevel Cache CPI Example

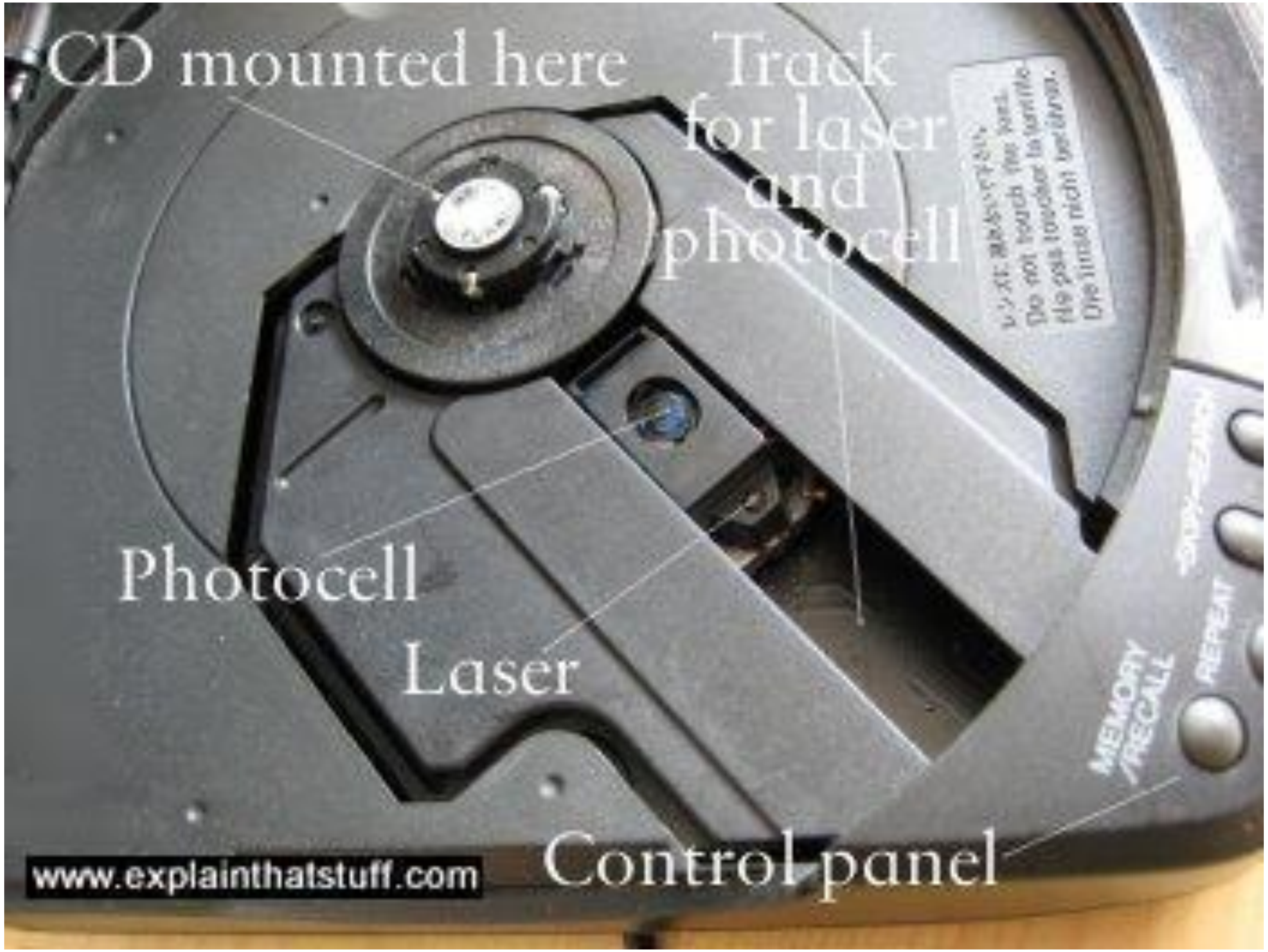
- Given
  - CPU base CPI = 1, clock rate = 4GHz
  - Miss rate/instruction = 2%
  - Main memory access time = 100ns
- With just primary cache
  - Miss penalty = 100ns/0.25ns = 400 cycles
  - Effective CPI = 1 + 0.02 × 400 = 9

Optical Drives CD / DVD

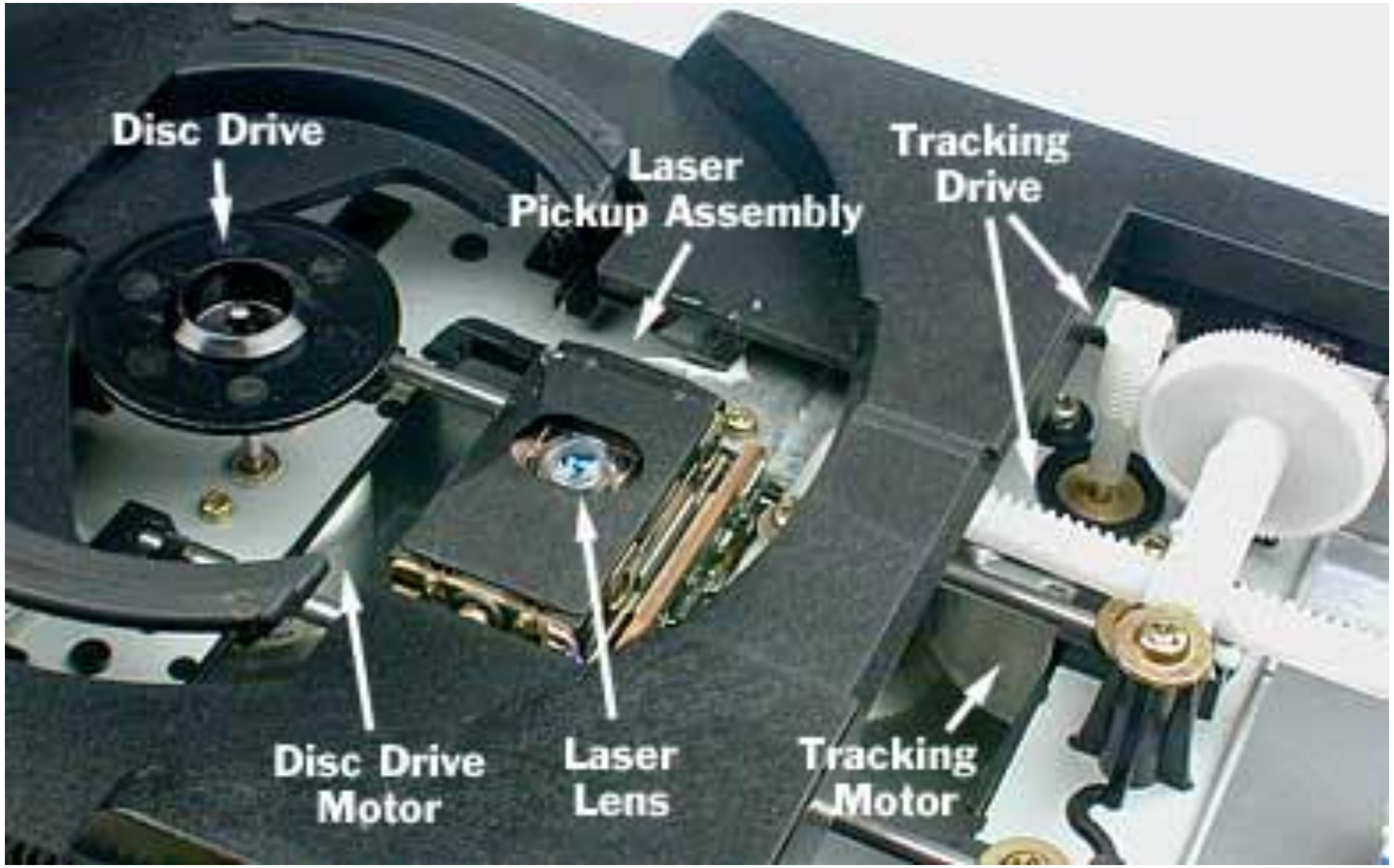
Optical Storage CD-ROM

- Originally for audio
- 650Mbytes giving over 70 minutes audio
- Polycarbonate coated with highly reflective coat, usually aluminum
- Data stored as pits
- Read by reflecting laser
- Constant packing density
- Constant linear velocity

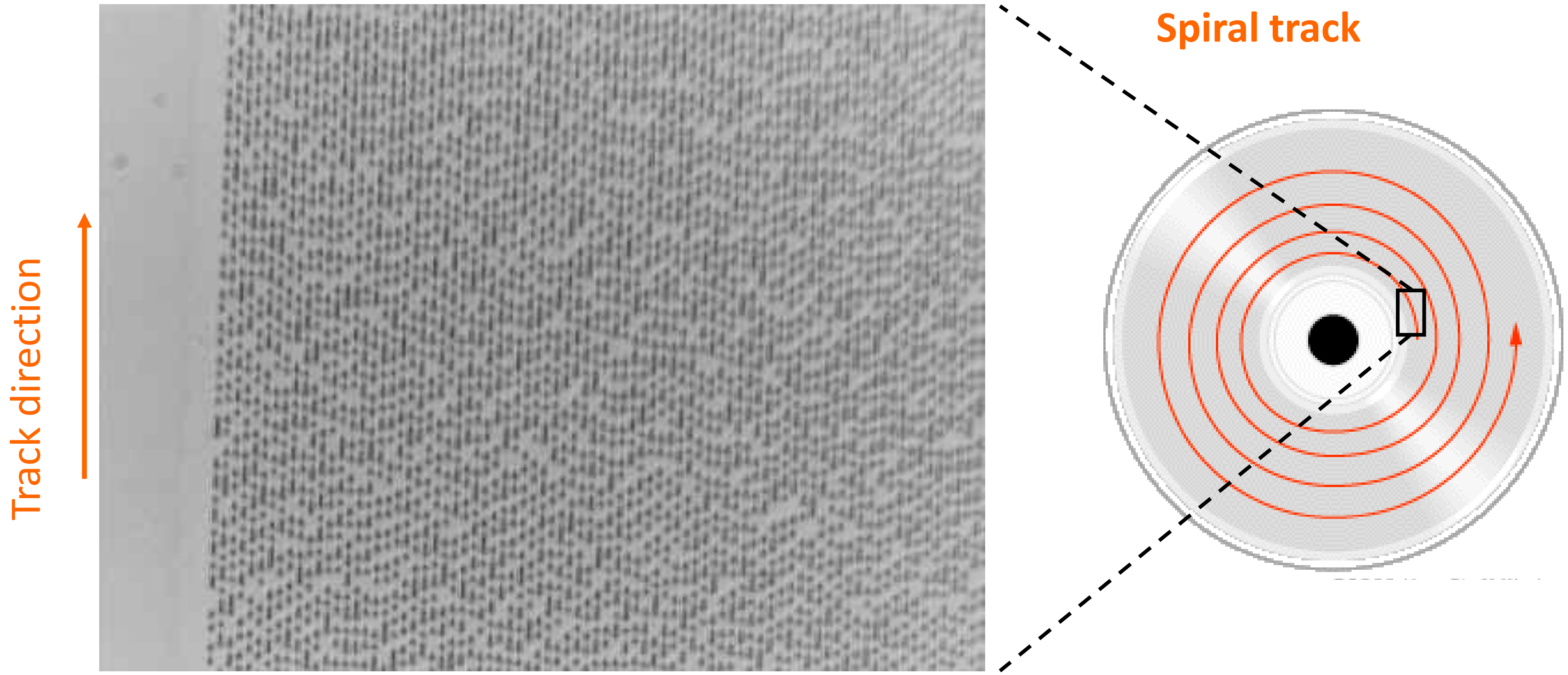
CD Components



CD Components

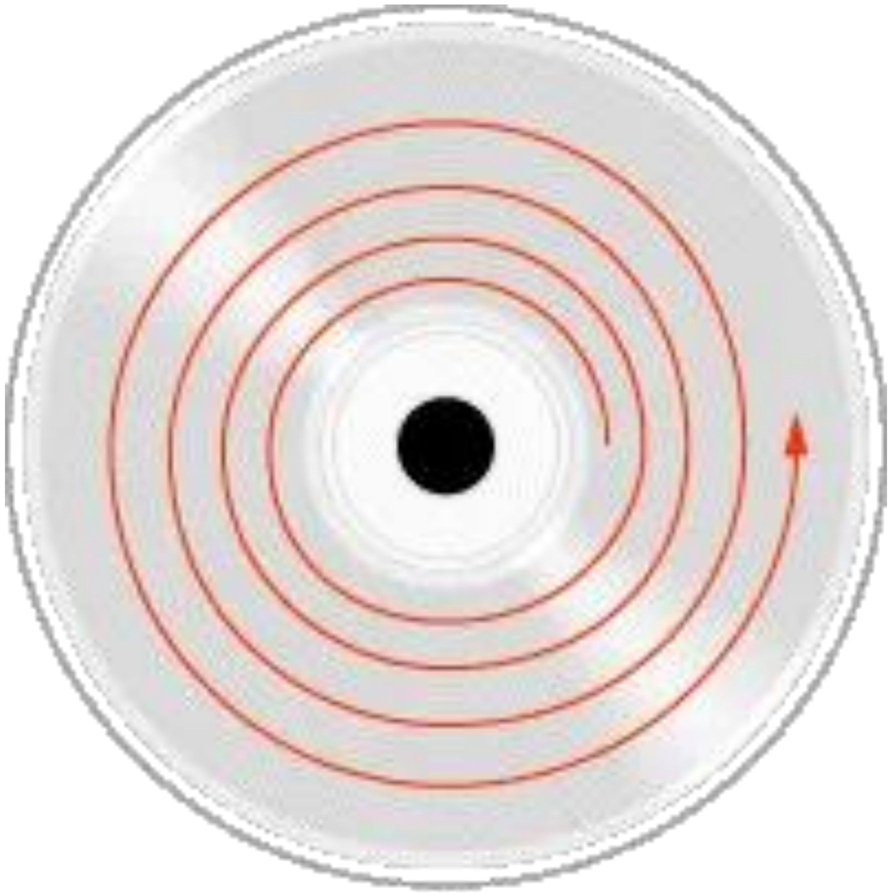


CD Surface



Low-magnification (×32) image of a CD showing an edge of the data zone.

Sprial Track of CD



©2000 How Stuff Works

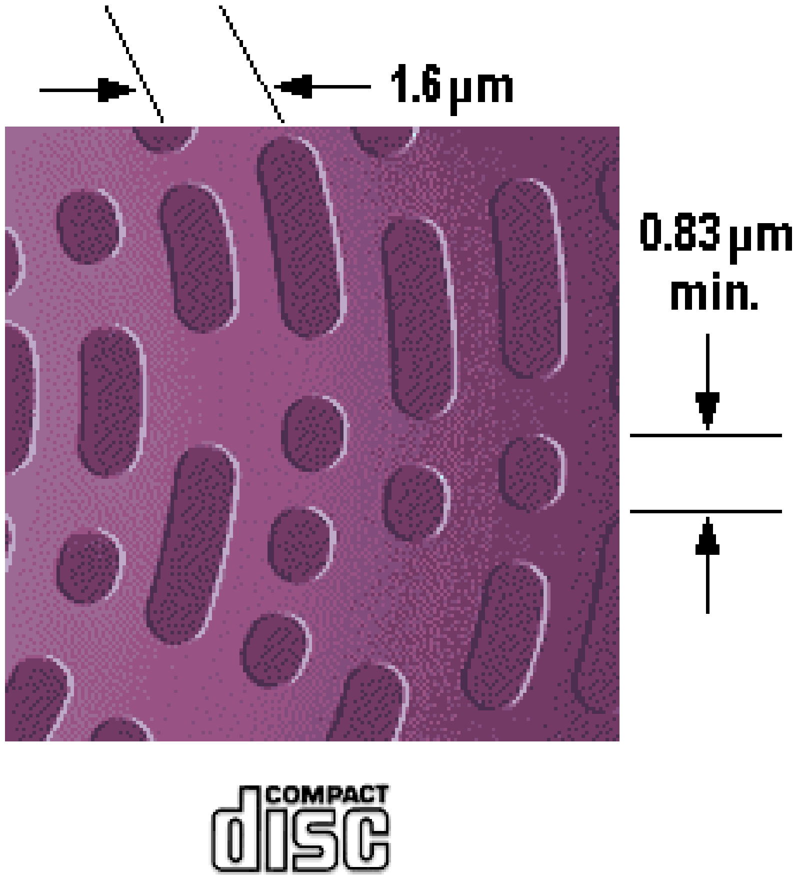
Track Density and Data Density

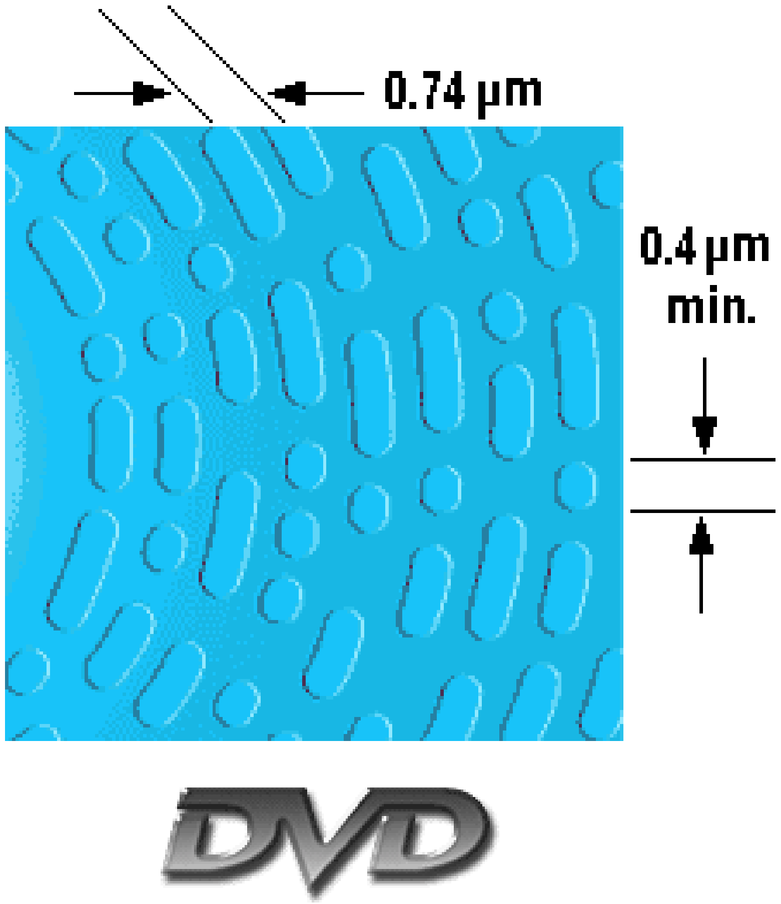
The CD is 12 cm in diameter, 1.2 mm thick, has a center hole 1.5 cm in diameter, and spins at a *constant linear velocity* (CLV) or *constant angular velocity* (CAV).

There is only one track on the optical disk and all data are stored in a spiral of about 2 billion small pits on the surface. There are about 30,000 windings on a CD - all part of the same track. This translates into about 16,000 tracks per inch and an areal density of 1 Mb/mm^2.

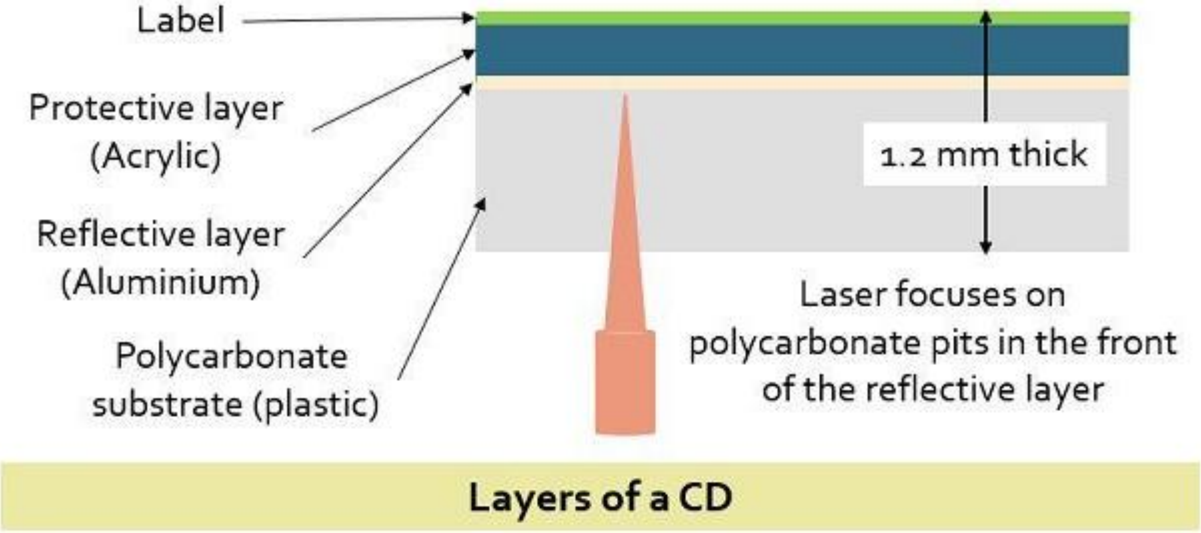
The total length of the track on a CD is almost 3 miles.

Track Density CD vs DVD





Surface Layers

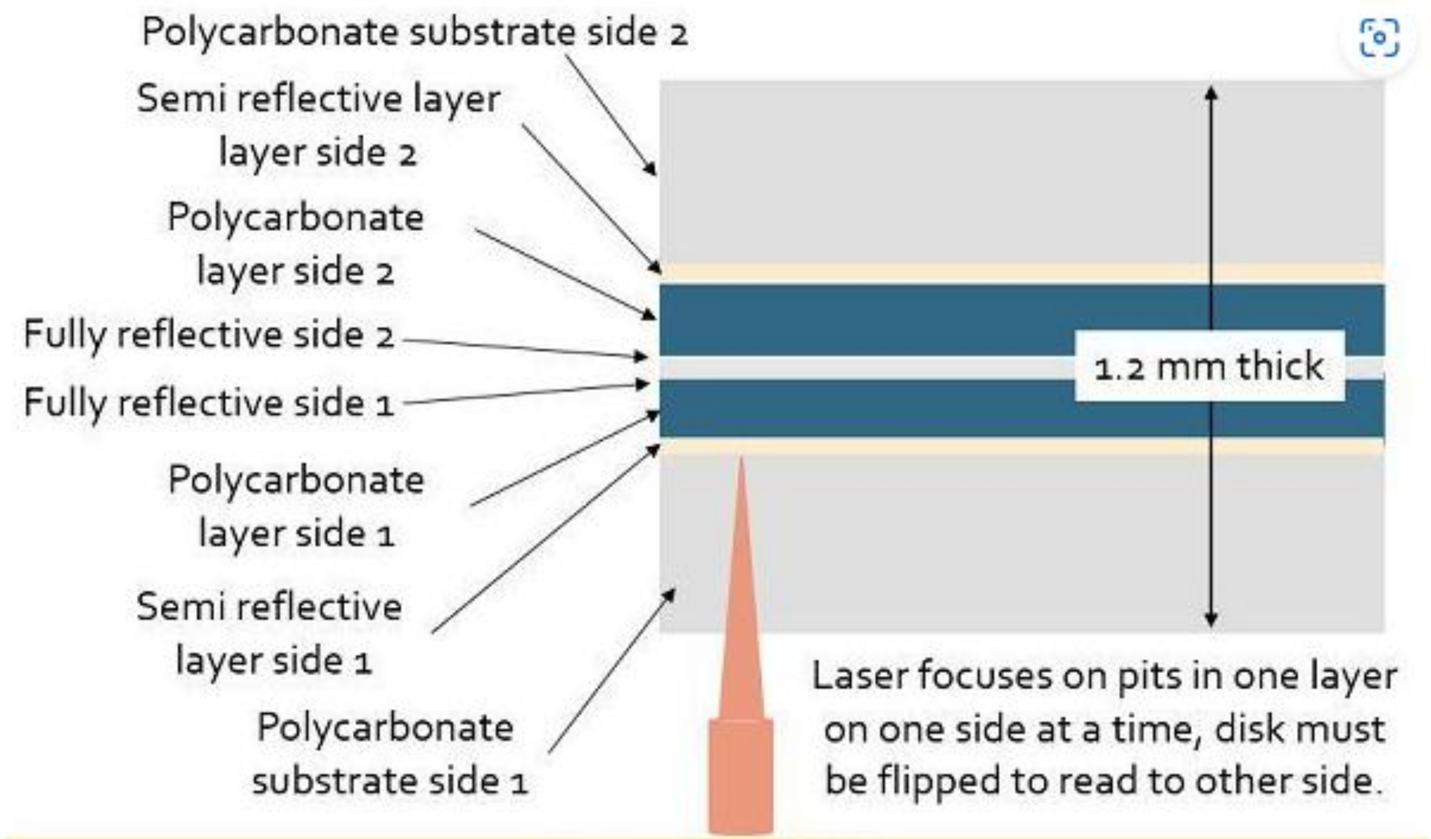

Label
Protective layer (Acrylic)
Reflective layer (Aluminium)
Polycarbonate substrate (plastic)
1.2 mm thick
Laser focuses on polycarbonate pits in the front of the reflective layer
Layers of a CD

A CD can store up to 74 minutes of music, so the total amount of digital data that must be stored on a CD is:

2 channels × 44,100 samples/channel/second × 2 bytes/sample × 74 minutes × 60 seconds/minute = 783,216,000 bytes

To fit more than 783 megabytes onto a disk only 12 cm in diameter requires that the individual bits be very small.

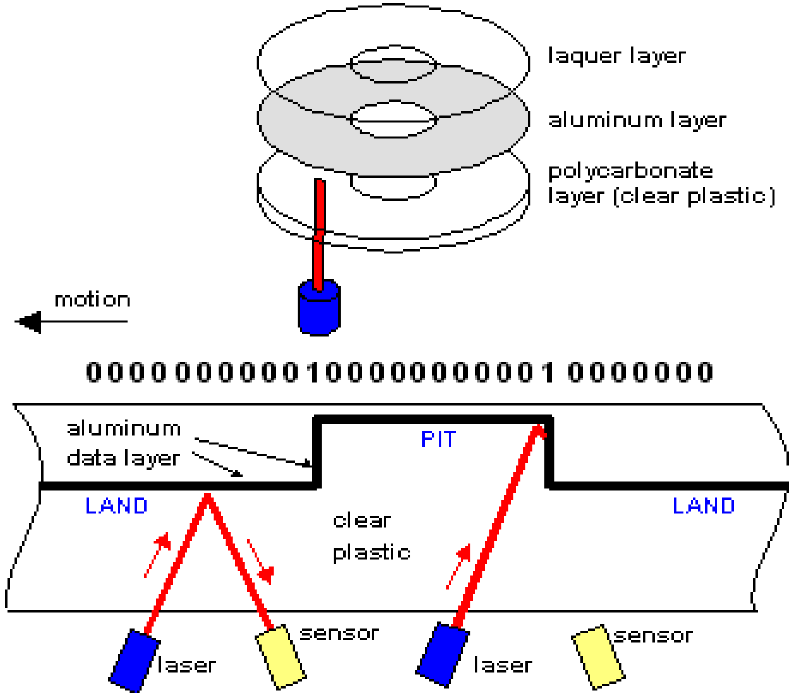
Layers of a DVD

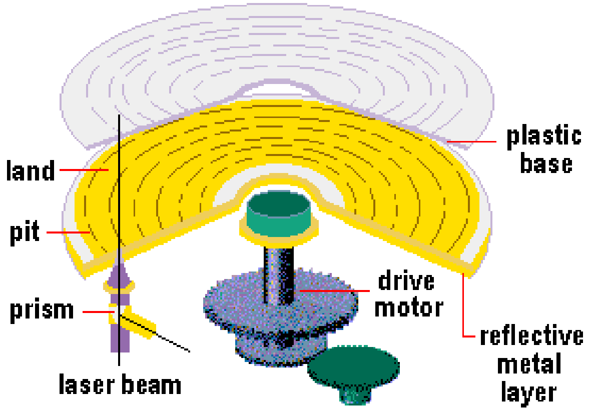

Polycarbonate substrate side 2
Semi reflective layer layer side 2
Polycarbonate layer side 2
Fully reflective side 2
Fully reflective side 1
Polycarbonate layer side 1
Semi reflective layer side 1
Polycarbonate substrate side 1
1.2 mm thick

Laser focuses on pits in one layer on one side at a time, disk must be flipped to read to other side.

Layers of a DVD

CD Mechanism





Digital data are carved into the disc as pits (low spots) and lands (high spots). As the laser shines into the moving pits and lands, a sensor detects a change in reflection when it encounters a transition from pit to land or land to pit.
Each transition is a 1. The lack of transitions are 0s. There is only one laser in a drive. Two are used here to illustrate the difference in reflection.

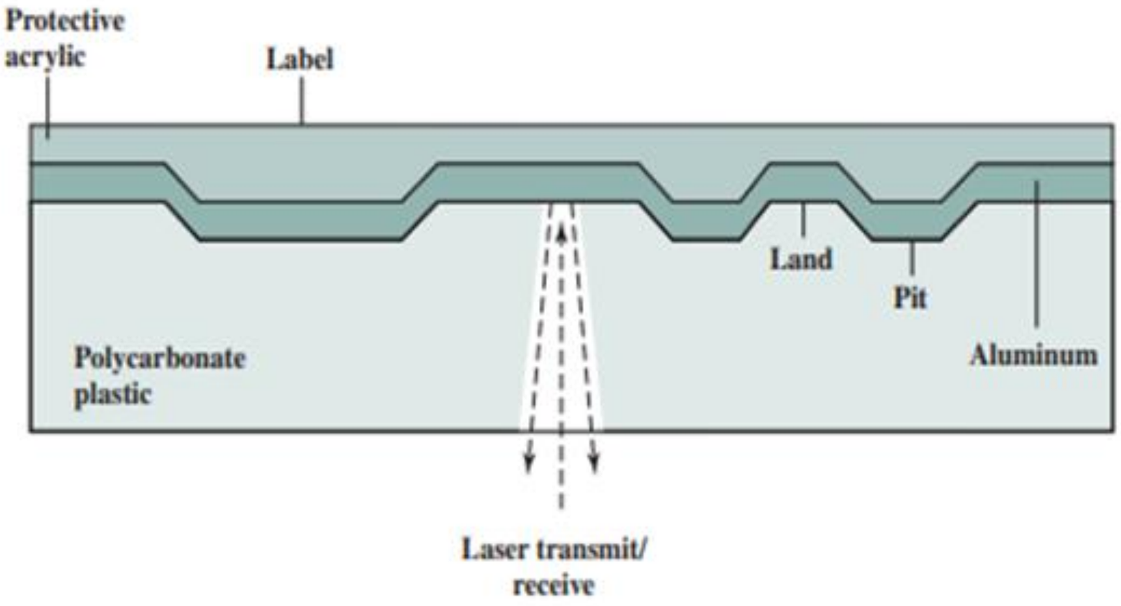
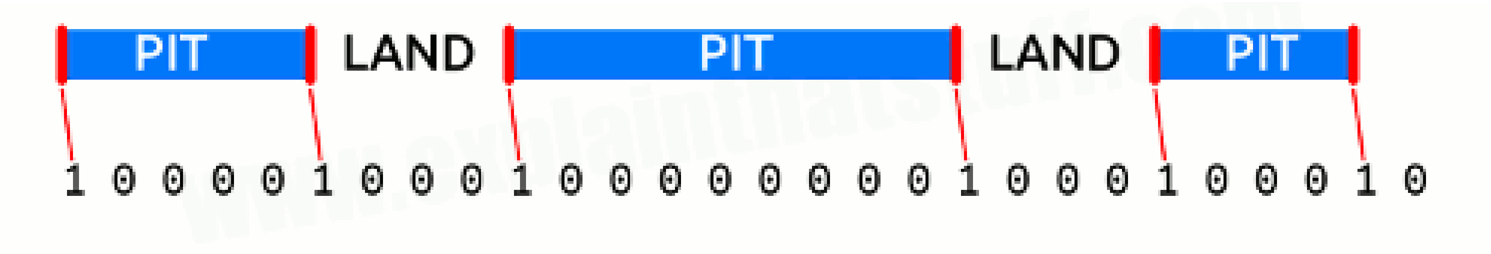
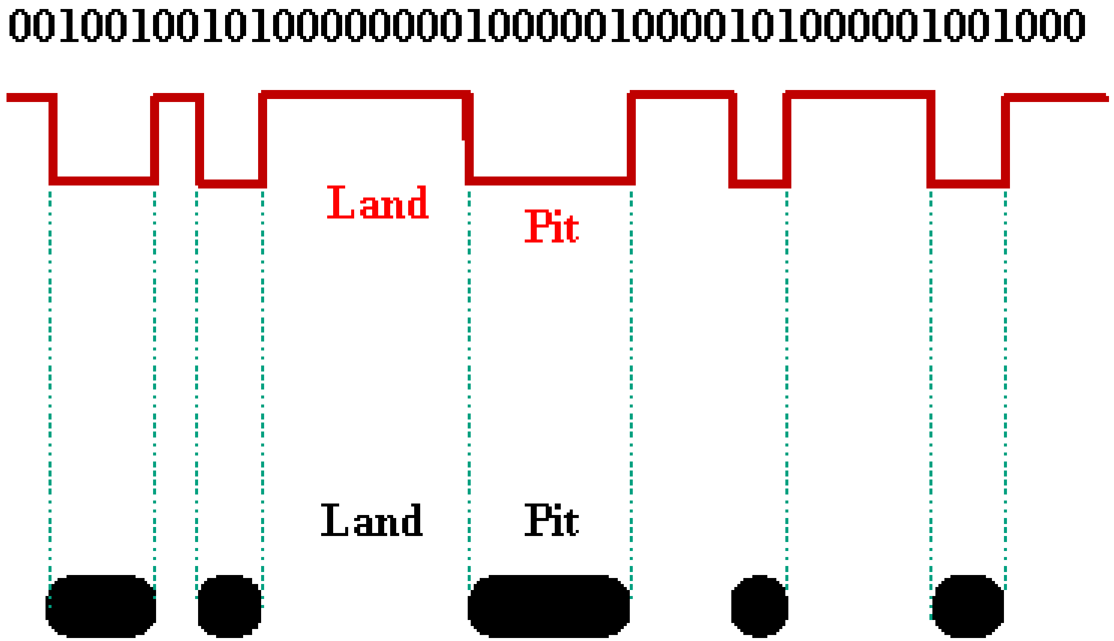


Figure 6.9 CD Operation

Data Format


PIT
LAND
PIT
LAND
PIT
1 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 1 0


001001001010000000001000000100001010000001001000
Land
Pit
Land
Pit

Storage Write and Read Principle

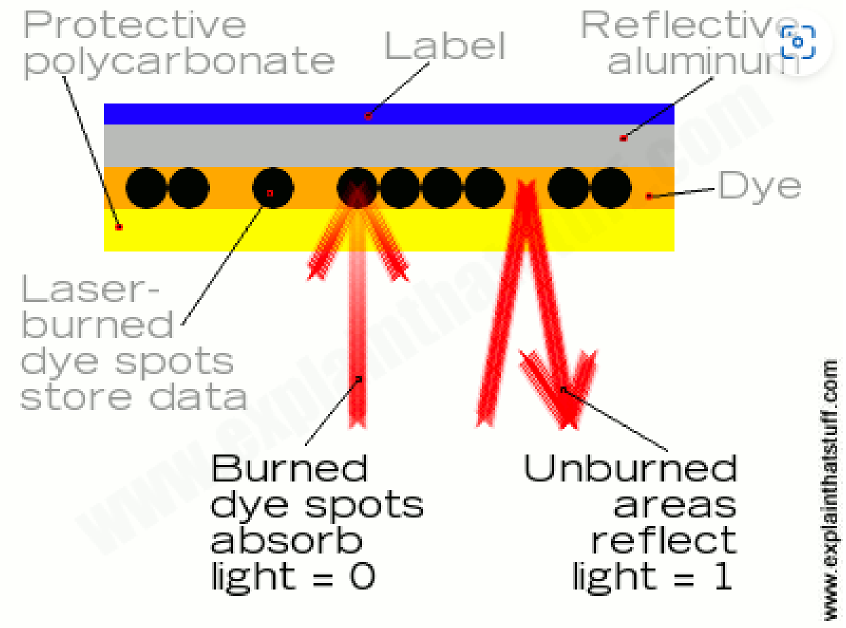
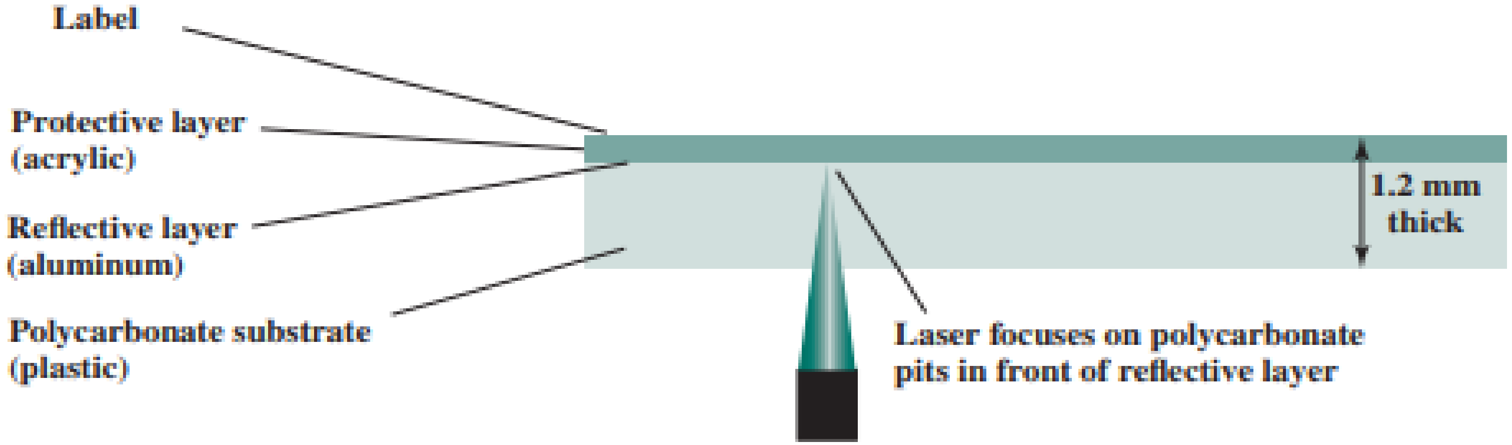
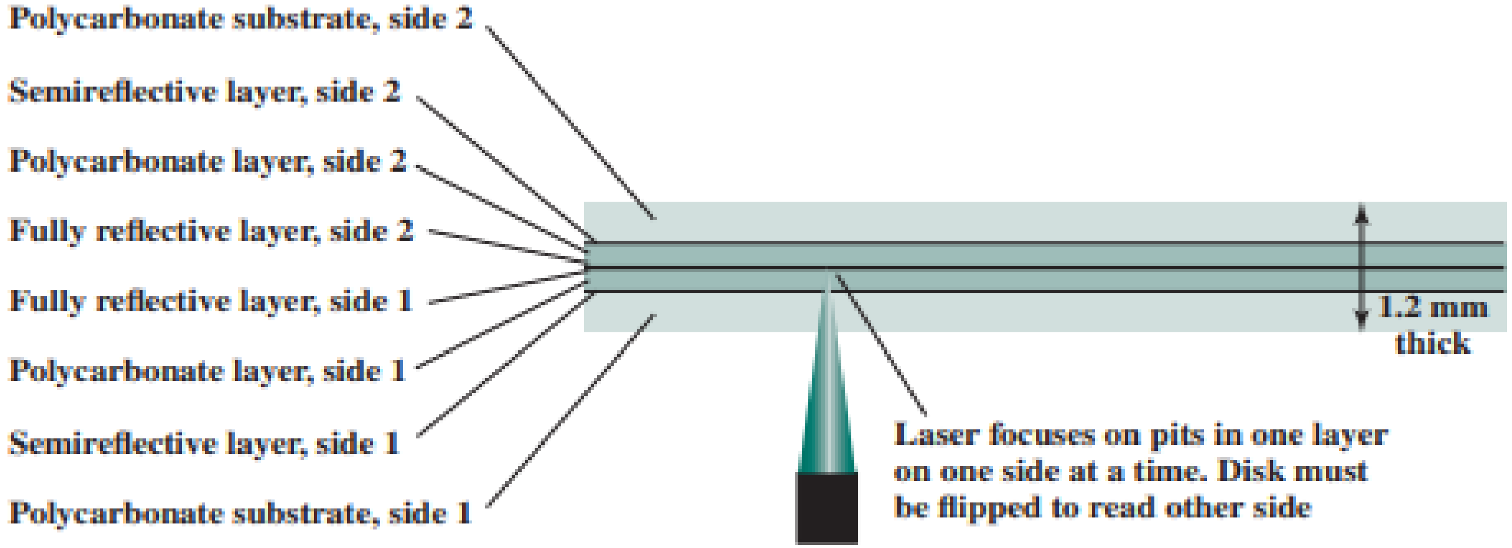

Protective polycarbonate
Label
Reflective aluminum
Dye
Laser-burned dye spots store data
Burned dye spots absorb light = 0
Unburned areas reflect light = 1
www.explainthatstuff.com

Illustration: With a CD-R, binary information is stored as "burned" areas (0) and unburned areas (1) in the dye layer sandwiched between the protective polycarbonate and the reflective aluminum.

CD and DVD Read Details


Label
Protective layer (acrylic)
Reflective layer (aluminum)
Polycarbonate substrate (plastic)
1.2 mm thick
Laser focuses on polycarbonate pits in front of reflective layer

(a) CD-ROM–Capacity 682 MB

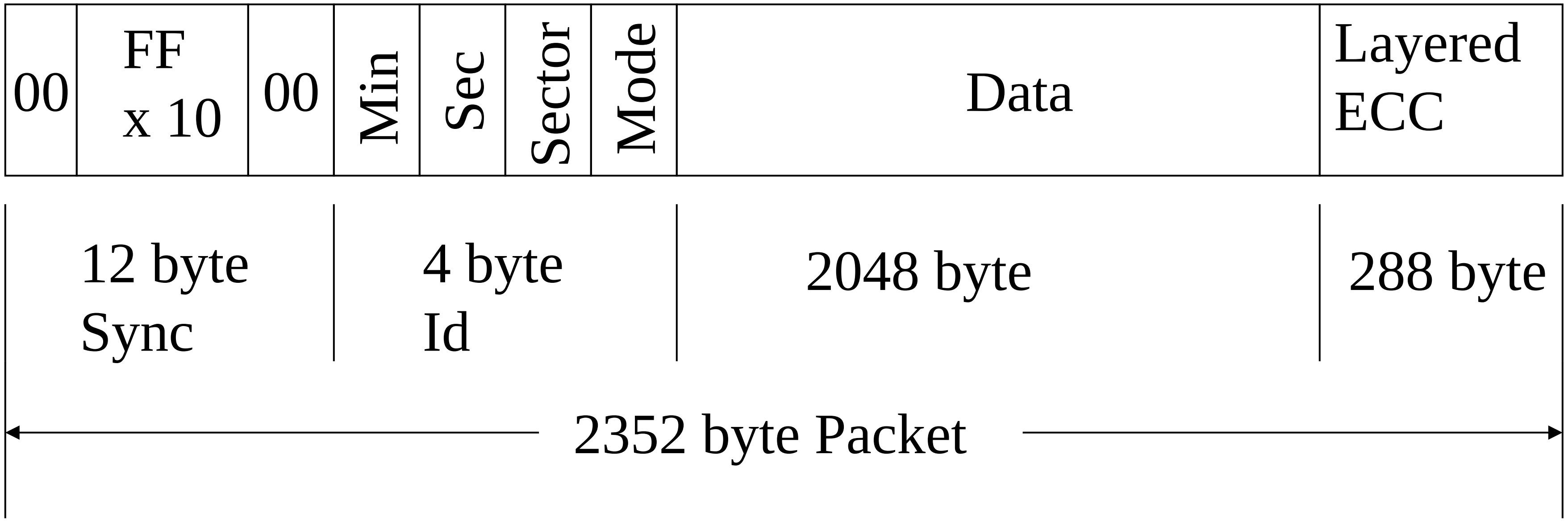

Polycarbonate substrate, side 2
Semireflective layer, side 2
Polycarbonate layer, side 2
Fully reflective layer, side 2
Fully reflective layer, side 1
Polycarbonate layer, side 1
Semireflective layer, side 1
Polycarbonate substrate, side 1
1.2 mm thick
Laser focuses on pits in one layer on one side at a time. Disk must be flipped to read other side

(b) DVD-ROM, double-sided, dual-layer–Capacity 17 GB

CD-ROM Drive Speeds

- Audio is single speed
  - Constant linear velocity
  - 1.2 ms^{-1}
  - Track (spiral) is 5.27km long
  - Gives 4391 seconds = 73.2 minutes
  - Date 176.4 K bytes/s total capacity 774.57 M Bytes
- Other speeds are quoted as multiples
- e.g. 24x ~= 4 M Bytes/s (data transfer rate)
- The quoted figure is the maximum the drive can achieve

CD-ROM Format Frame



00	FF x 10	00	Min	Sec	Sector	Mode	Data	Layered ECC
12 byte Sync			4 byte Id				2048 byte	288 byte

2352 byte Packet

- Mode 0=blank data field
- Mode 1=2048 byte data+error correction
- Mode 2=2336 byte data

Random Access on CD-ROM

- Difficult
- Move head to rough position
- Set correct speed
- Read address
- Adjust to required location

Readings

- Chap 5 of P&H Textbook