CIS 2107
Memory Hierarchy

thank you, CMU

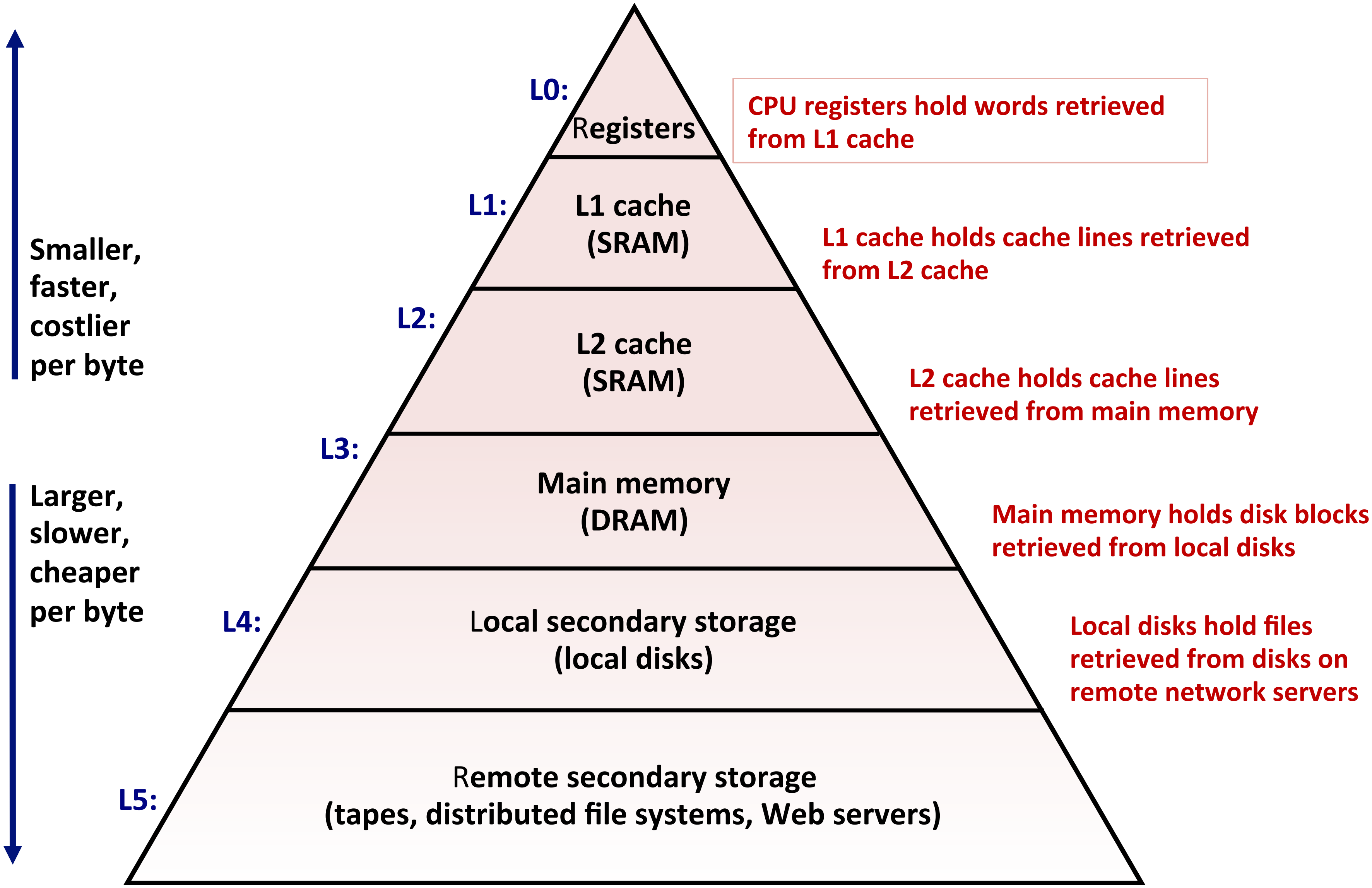
Memory Hierarchy

- Overview. Asymmetry.
- The details:
  - RAM
  - Disks

Memory Hierarchy

- **Overview. Asymmetry.**
- The details:
  - RAM
  - Disks

An Example Memory Hierarchy



Numbers Everyone Should Know

Jeff Dean talk at Stanford

L1 cache reference	0.5 ns
Branch mispredict	5 ns
L2 cache reference	7 ns
Mutex lock/unlock	100 ns
Main memory reference	100 ns
Compress 1K bytes with Zippy	10,000 ns
Send 2K bytes over 1 Gbps network	20,000 ns
Read 1 MB sequentially from memory	250,000 ns
Round trip within same datacenter	500,000 ns
Disk seek	10,000,000 ns
Read 1 MB sequentially from network	10,000,000 ns
Read 1 MB sequentially from disk	30,000,000 ns
Send packet CA $\rightarrow$ Netherlands$\rightarrow$CA	150,000,000 ns

Fake Problem

location	**access time**
L1 cache	0.5 ns
L2 cache	7 ns
RAM	100 ns
hard drive	10 ms
DVD	140 ms

Pretend for a minute

location	**access time**
L1 cache	0.5 ns
L2 cache	7 ns
RAM	100 ns
hard drive	10 ms
DVD	140 ms

location	***fake access time***
L1 cache	1 sec
L2 cache	
RAM	
hard drive	
DVD	

If we keep the ratios the same as on the LHS, what are the remaining numbers on the RHS?

Pretend for a minute

location	**access time**
L1 cache	0.5 ns
L2 cache	7 ns
RAM	100 ns
hard drive	10 ms
DVD	140 ms

location	***fake access time***
L1 cache	1 sec
L2 cache	14 sec
RAM	200 sec or 3 min, 20 sec
hard drive	20,000,000 sec or $\approx$ 231.5 days
DVD	280,000,000 sec or $\approx$ 8.9 years

Pretend again. Let’s bake a cake.

if the counter is the CPU and the L1 cache is the cabinet two feet above …

location	***fake distance***
L1 cache	2 feet
L2 cache	28 feet
RAM	400 feet
hard drive	40,000,000 feet or about 7,500 miles about twice the distance from Philly to Nome, AK
DVD	560,000,000 feet or about 106,000 miles or about the distance around the earth 4.25 times

The moral of the story

- If you’re baking a cake and you have to walk to Nome, AK to get the flour:
  1. get what you need for the rest of the recipe
  2. pick up some sugar

Locality

- Temporal locality
- Spatial locality

Memory Hierarchy

- Overview. Asymmetry.
- The details:
  - **RAM**
  - Disks

Random Access Memory. RAM.

- Random Access?
- SRAM – *static* RAM
- DRAM – *dynamic* RAM

SRAM

- Fast
- Expensive
- Used in cache memory (on and off chip)
- How much? A few MB
- More transistors/circuit
- Stable. Retain value as long as there's power
- No need for refresh

DRAM

- Used in main memory, graphics framebuffer
- “Stores each bit as a charge on a capacitor”
- Sensitive to disturbance (elec. noise, radiation)
  – Use as digital cam sensor
- Leakage:
  – Needs to be refreshed every 10-100 ms
  – How? Just read value and write it again
- Slower than SRAM
- Cheaper than SRAM

SRAM vs DRAM

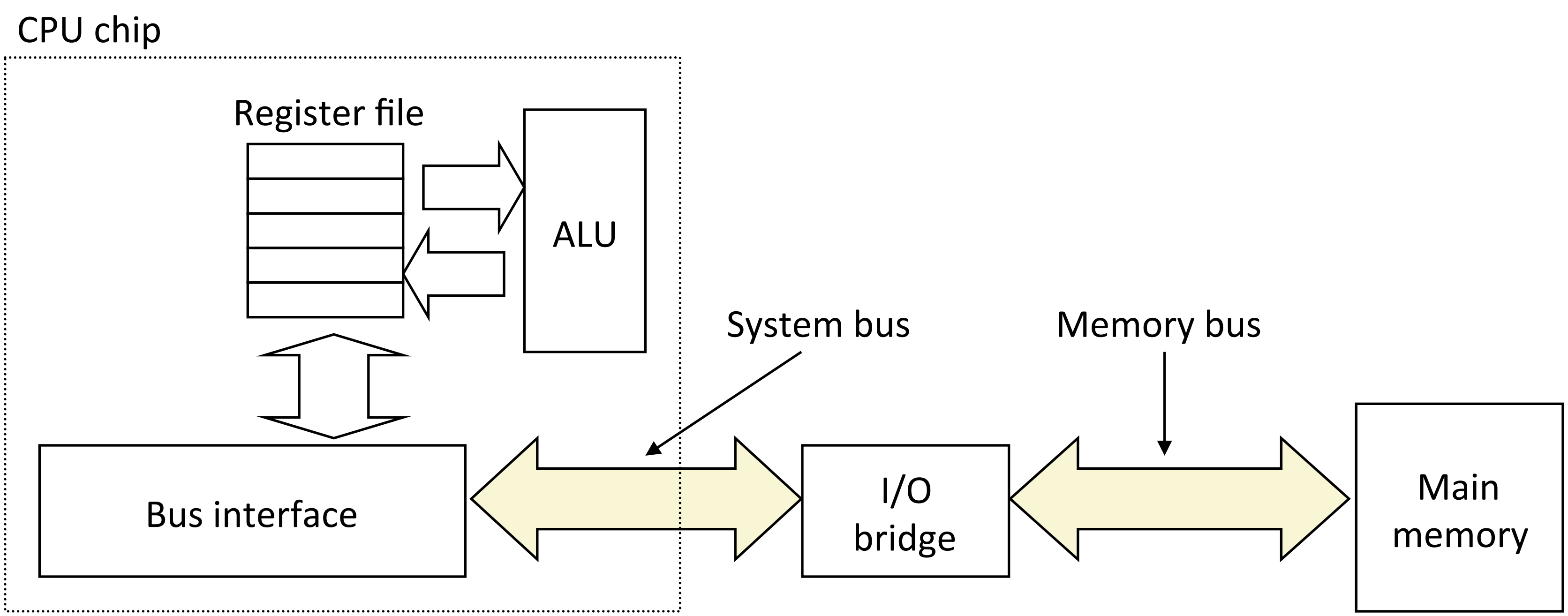
	transistors per bit	access time	needs refresh?	cost	where?
SRAM	6	1x	no	100x	cache memory (on or off chip)
DRAM	1	10x	yes	1x	main memory, graphics framebuffers

volatile vs. non-volatile storage

- volatile – value lost on power off
- non-volatile –
  - Hard disk
  - “ROMs”, “PROMs”, “firmware”
  - Solid state disks

Traditional Bus Structure Connecting CPU and Memory

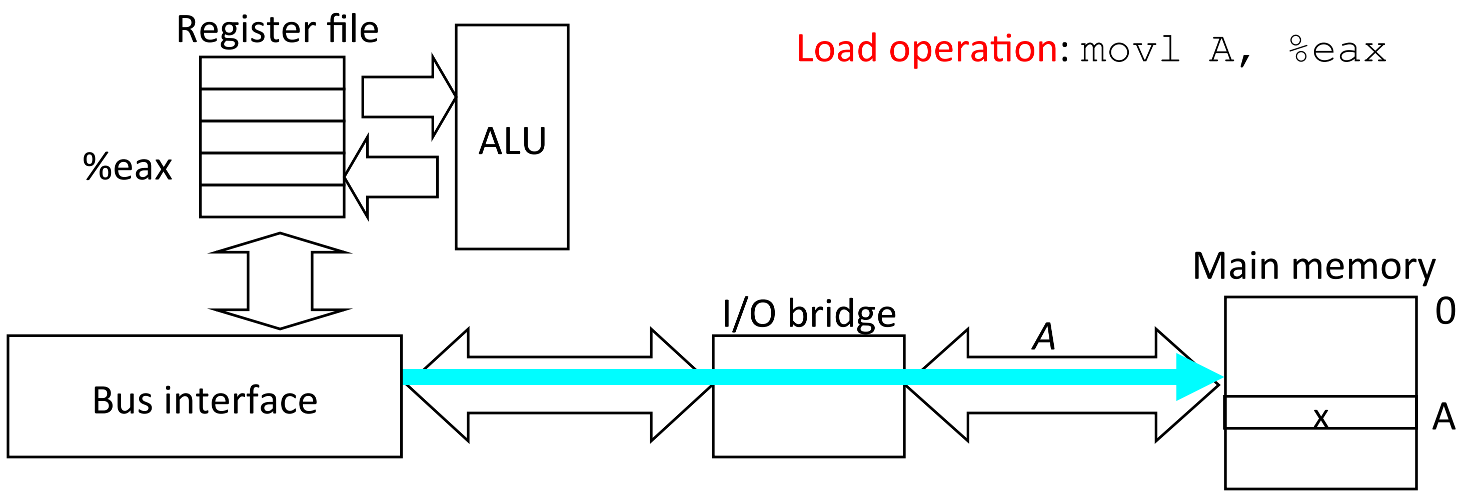
- A bus is a collection of parallel wires that carry address, data, and control signals.
- Buses are typically shared by multiple devices.



Memory Read Transaction (1)

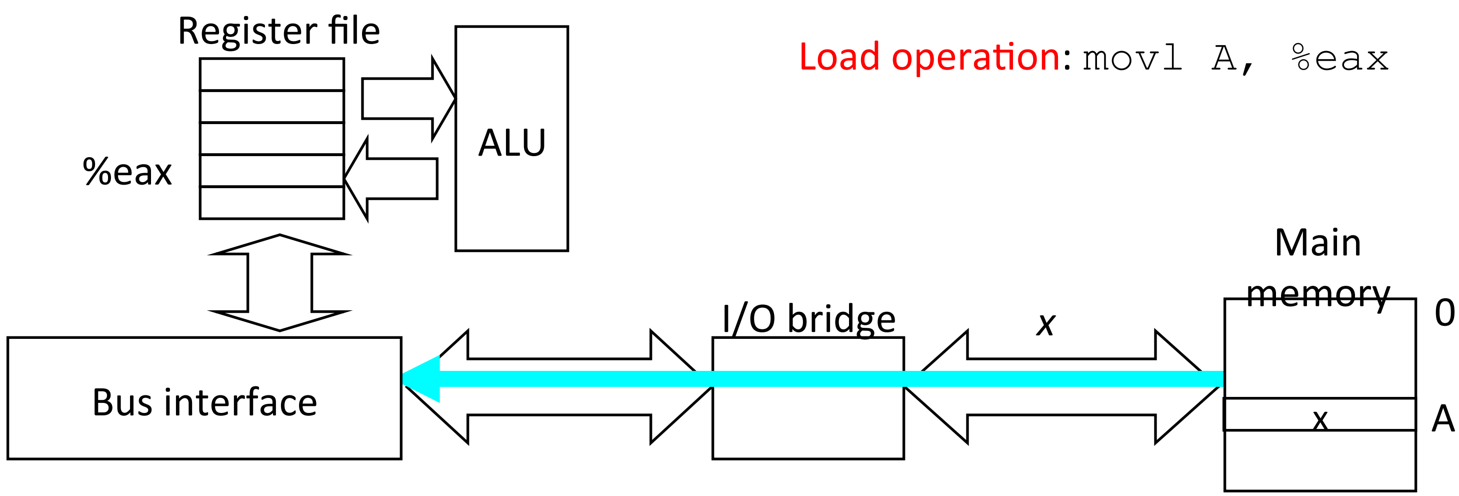
- CPU places address A on the memory bus.

Load operation: `movl A, %eax`



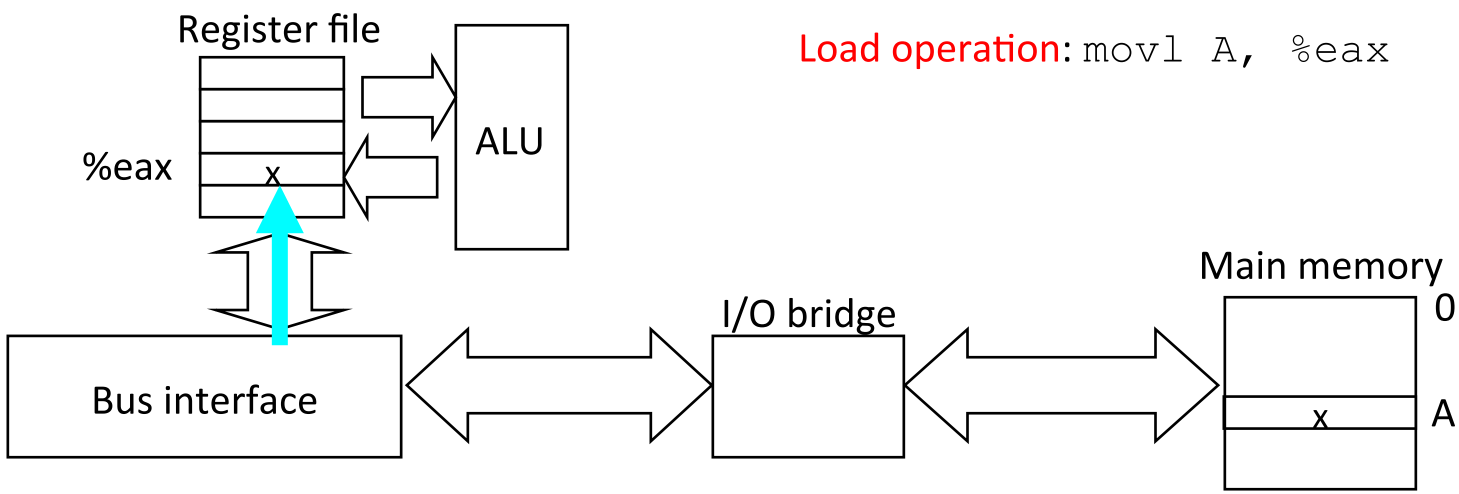
Memory Read Transaction (2)

- Main memory reads A from the memory bus, retrieves word x, and places it on the bus.



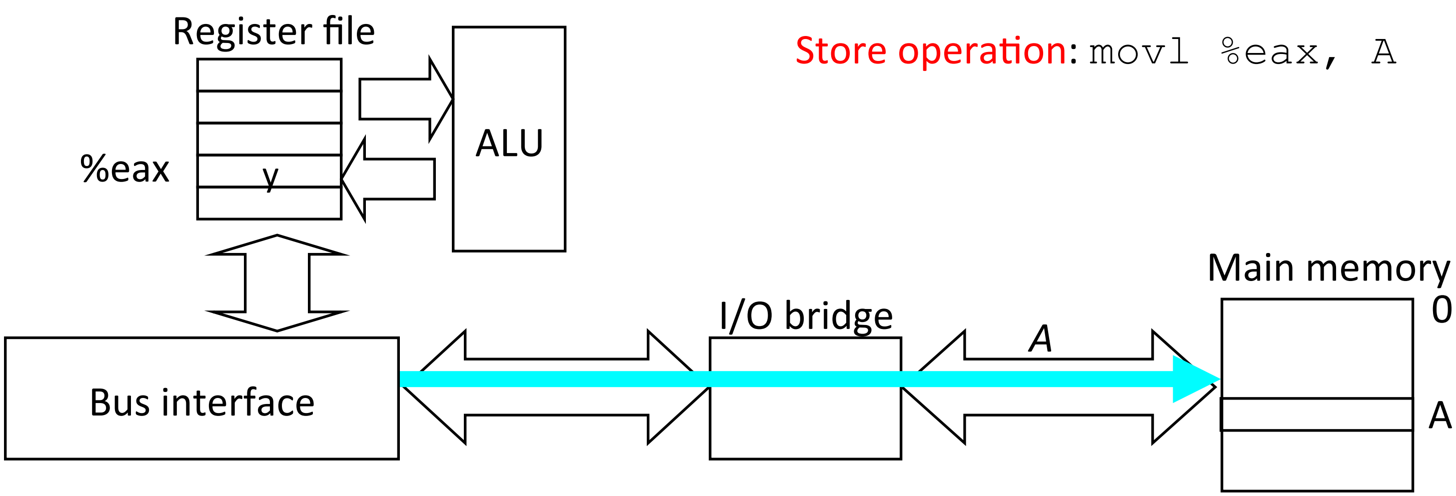
Memory Read Transaction (3)

- CPU read word x from the bus and copies it into register %eax.



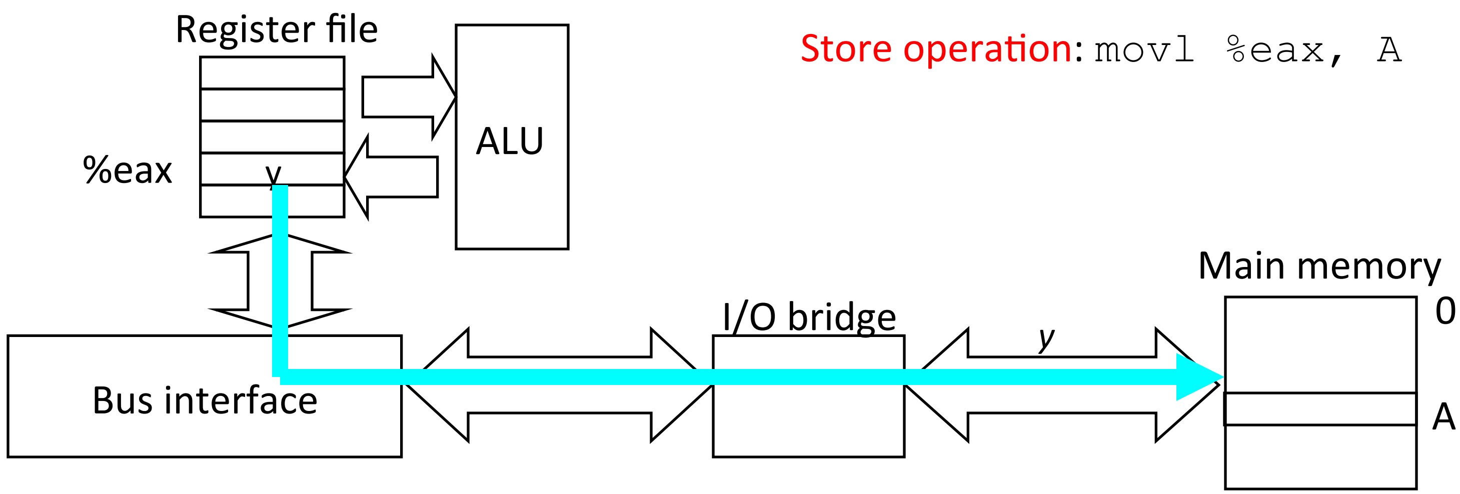
Memory Write Transaction (1)

- CPU places address A on bus. Main memory reads it and waits for the corresponding data word to arrive.



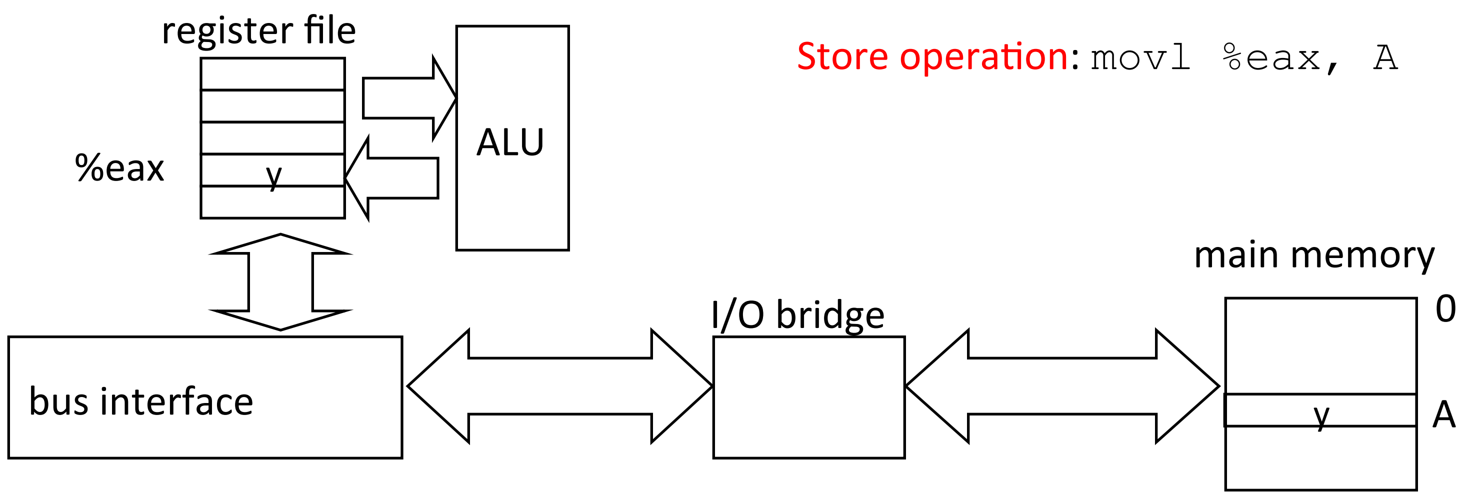
Memory Write Transaction (2)

- CPU places data word y on the bus.



Memory Write Transaction (3)

- Main memory reads data word y from the bus and stores it at address A.



Memory Hierarchy

- Overview. Asymmetry.
- The details:
  - RAM
  - **Disks**

What's Inside A Disk Drive?

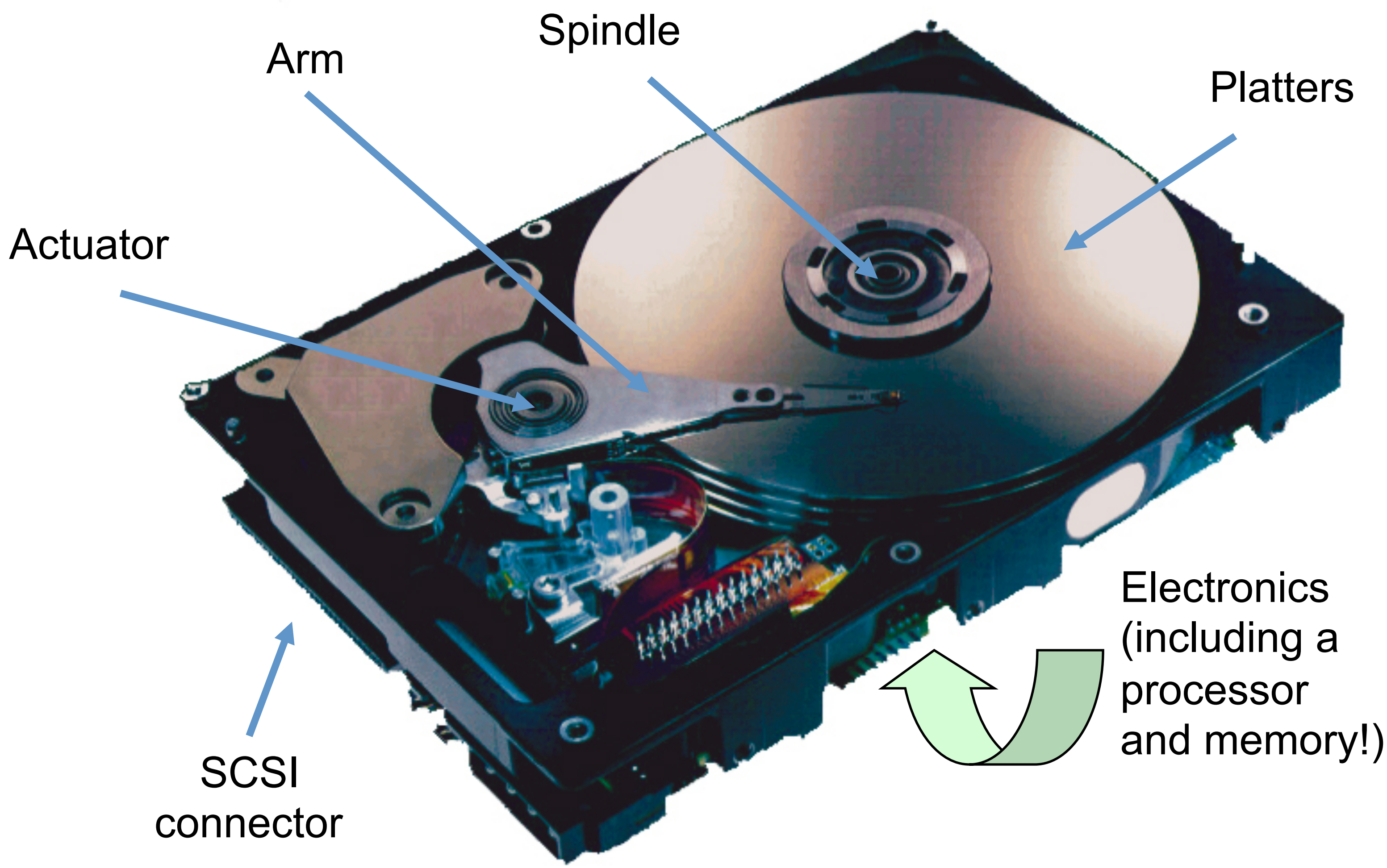
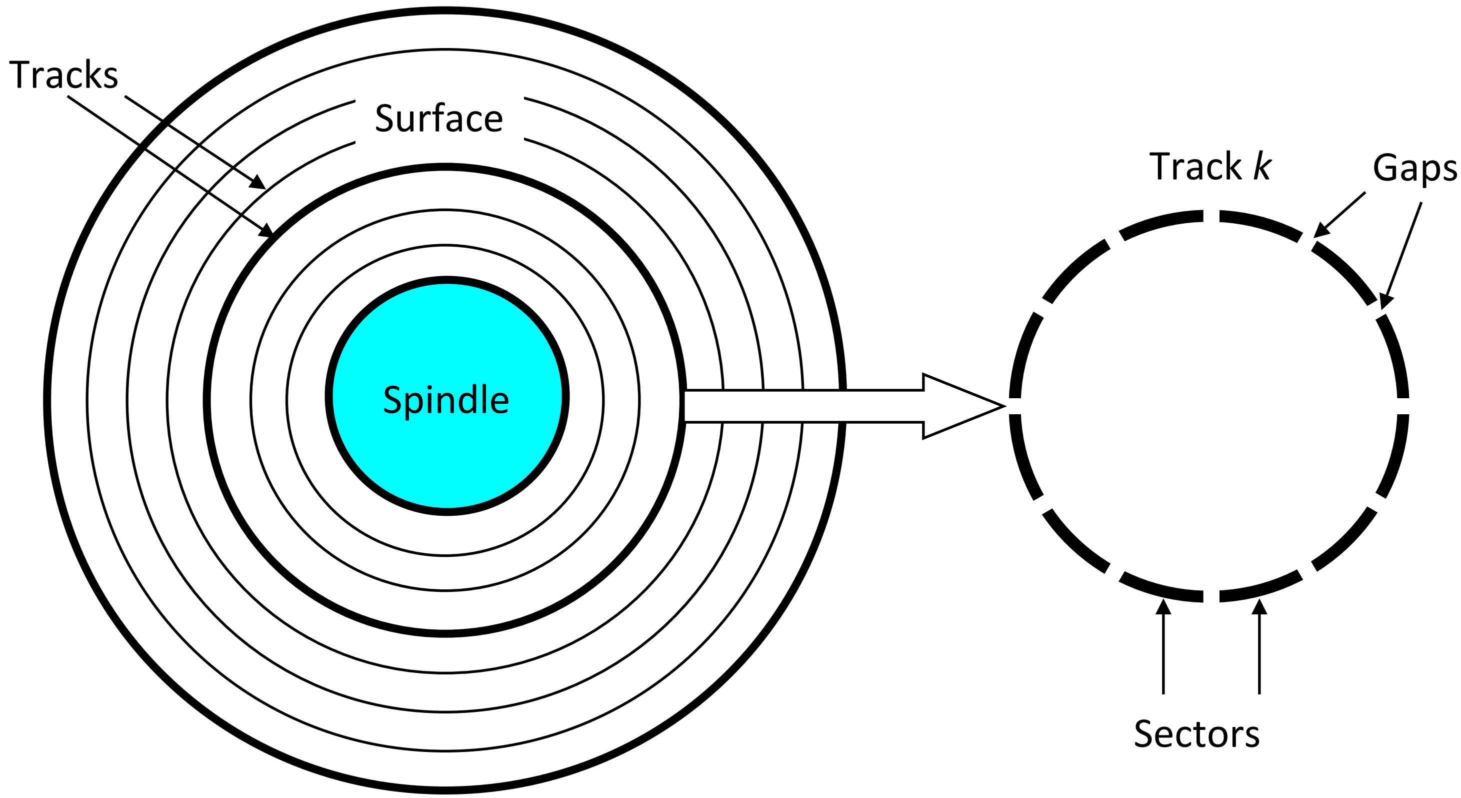


Image courtesy of Seagate Technology

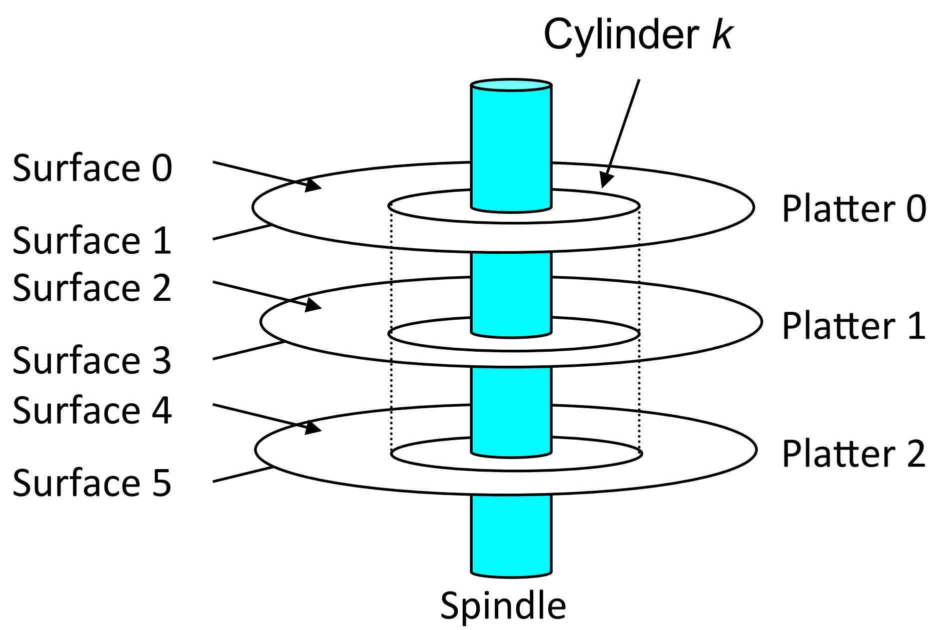
Disk Geometry

- Disks consist of platters, each with two surfaces.
- Each surface consists of concentric rings called tracks.
- Each track consists of sectors separated by gaps.



Disk Geometry (Muliple-Platter View)

- Aligned tracks form a cylinder.

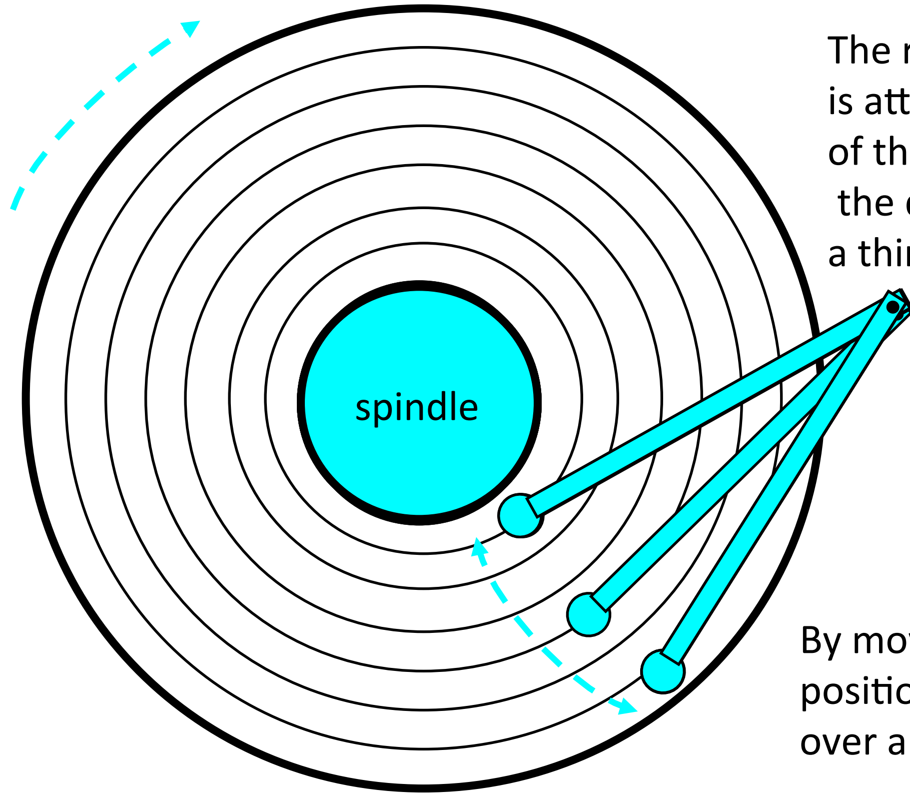


disk terms

- Capacity – #bits that can be recorded
- Density
  - **recording density** #bits per 1-inch seg of track.
  - **track density** #tracks per 1-inch radial segment.
  - **areal density** (bits/in^2) recording density * track density
- Recording **zones** – partition of recording space
  - Each track in a zone – same number of sectors
  - Each zone – different number of sectors/track

Disk Operation (Single-Platter View)

The disk surface spins at a fixed rotational rate



The read/write *head* is attached to the end of the *arm* and flies over the disk surface on a thin cushion of air.

By moving radially, the arm can position the read/write head over any track.

spin

Rotation rate these days:

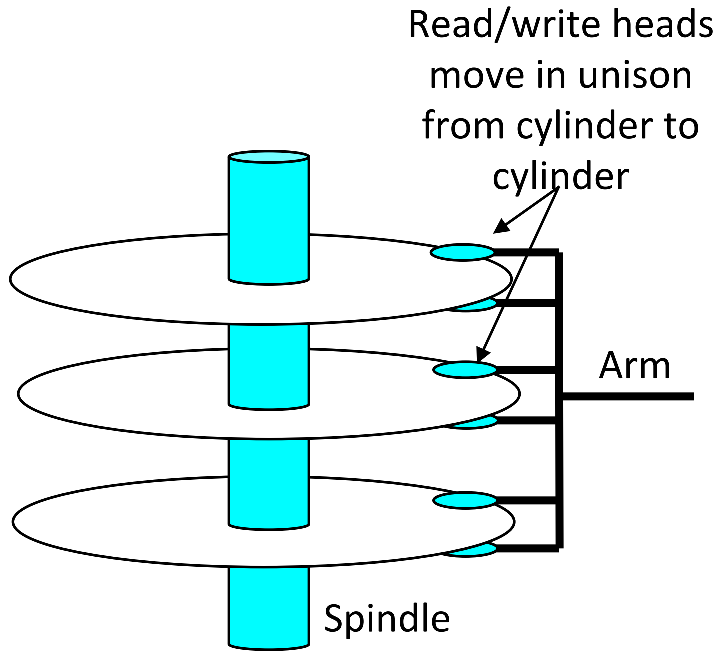
5400, 7200 RPM

from the book …

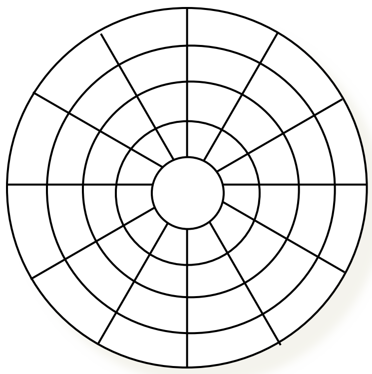
- equivalent of Sears Tower on its side:
  - 1 inch above the surface of the Earth
  - each orbit takes 8 seconds

- Speck of dust equivalent to a boulder
- Head crash

Disk Operation (Multi-Platter View)



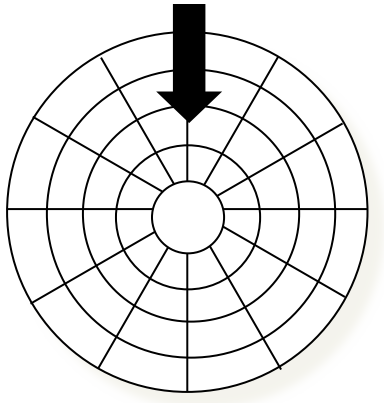
Disk Structure - top view of single platter



Surface organized into tracks

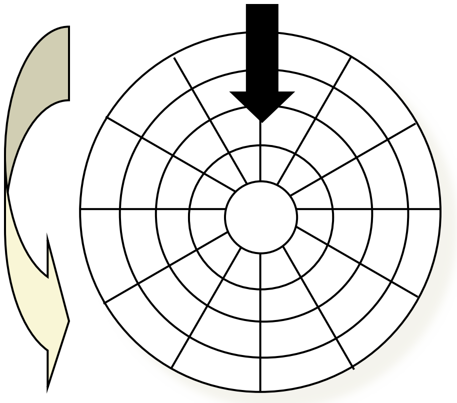
Tracks divided into sectors

Disk Access



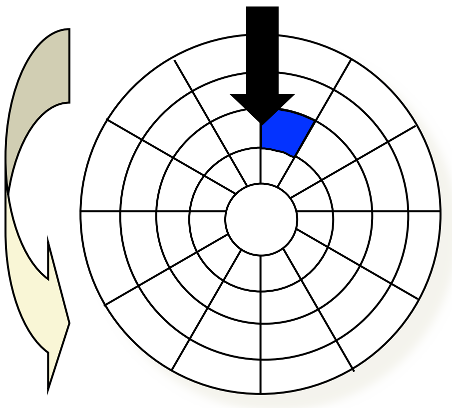
Head in position above a track

Disk Access



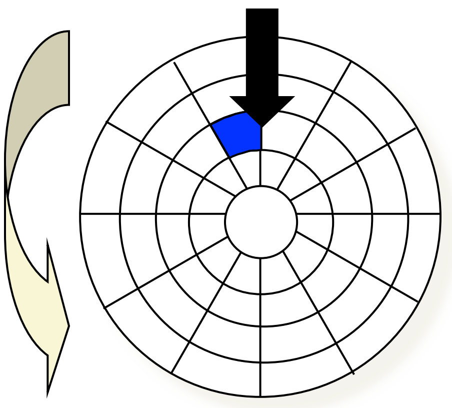
Rotation is counter-clockwise

Disk Access – Read



About to read blue sector

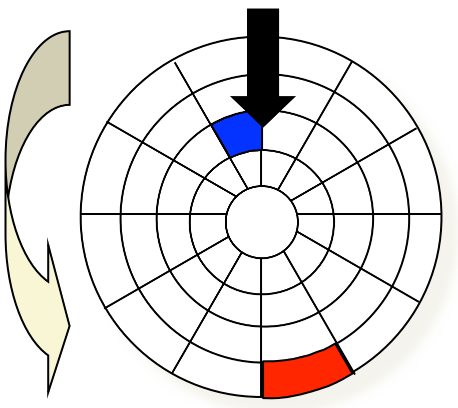
Disk Access – Read



After BLUE read

After reading blue sector

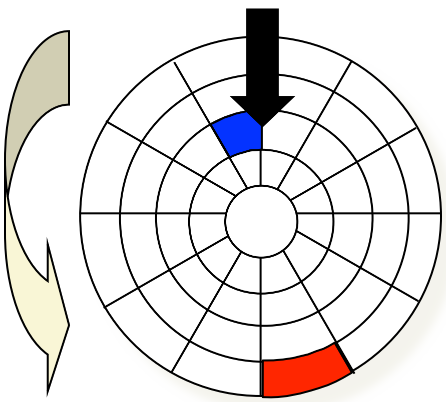
Disk Access – Read



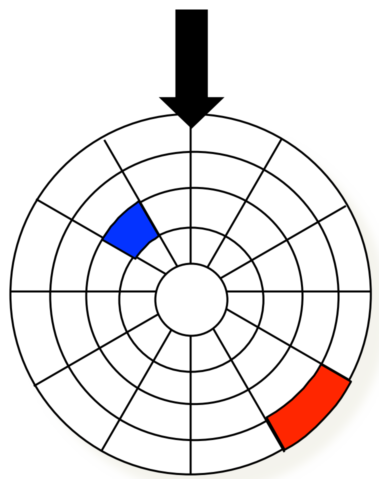
After BLUE read

Red request scheduled next

Disk Access – Seek



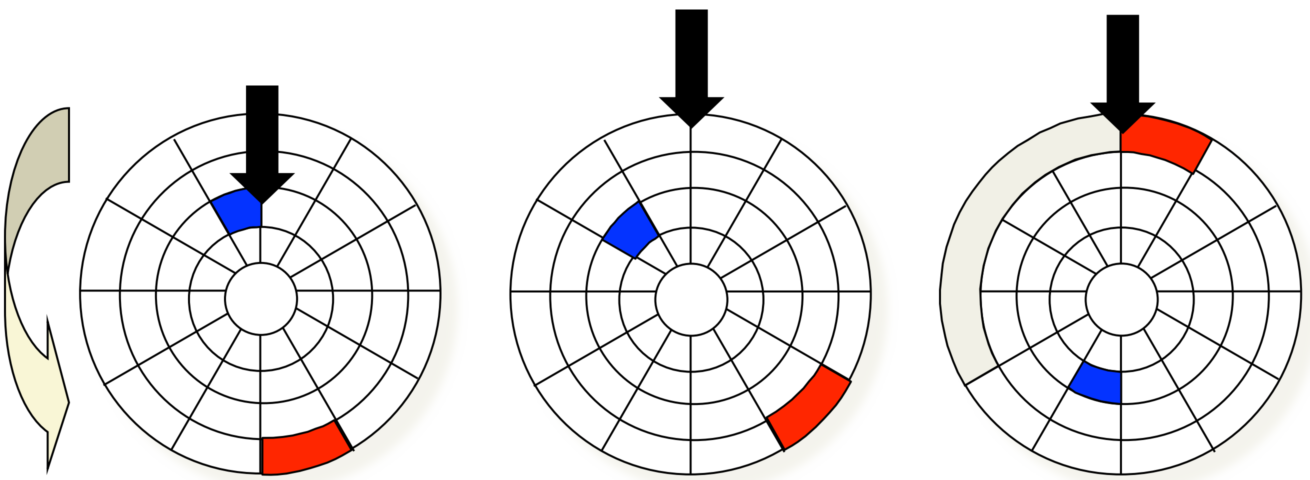
After BLUE read



Seek for RED

Seek to red's track

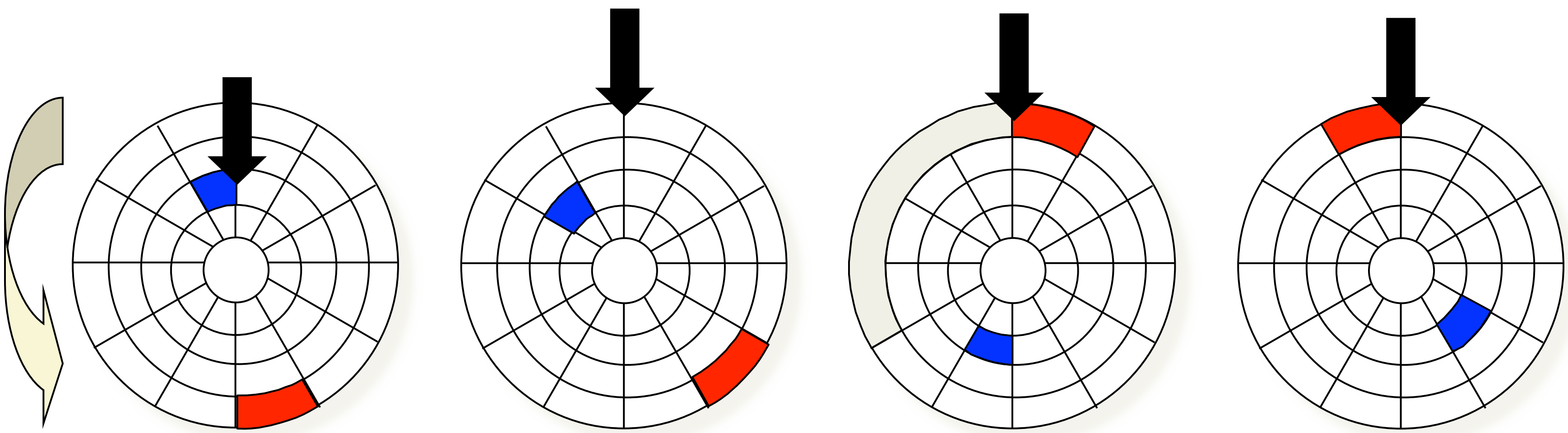
Disk Access – Rotational Latency



After BLUE read Seek for RED Rotational latency

Wait for red sector to rotate around

Disk Access – Read



After BLUE read

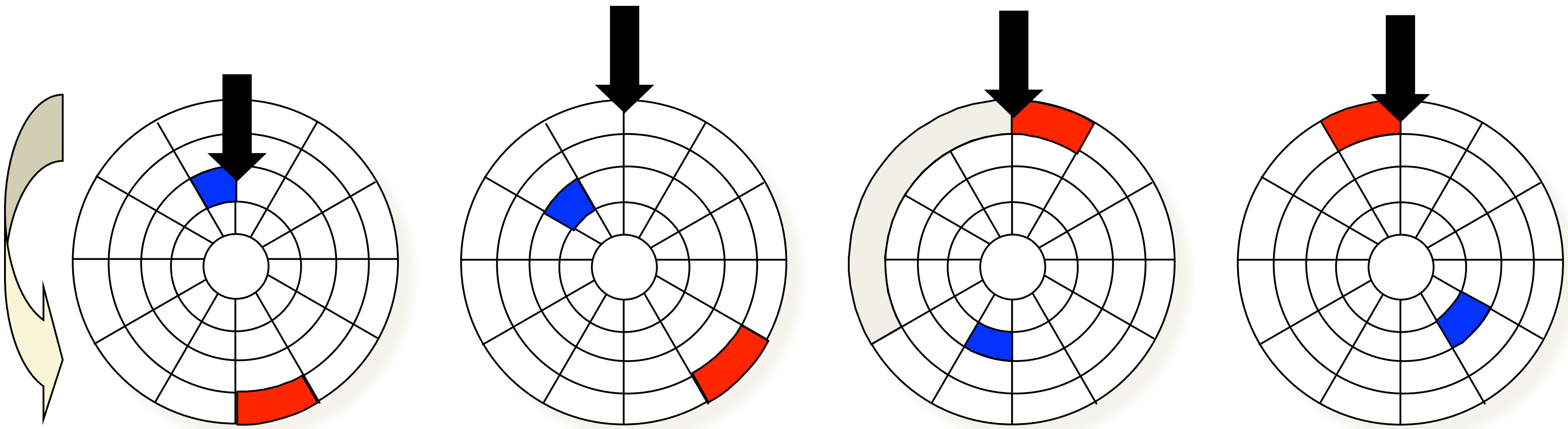
Seek for RED

Rotational latency

After RED read

Complete read of red

Disk Access – Service Time Components



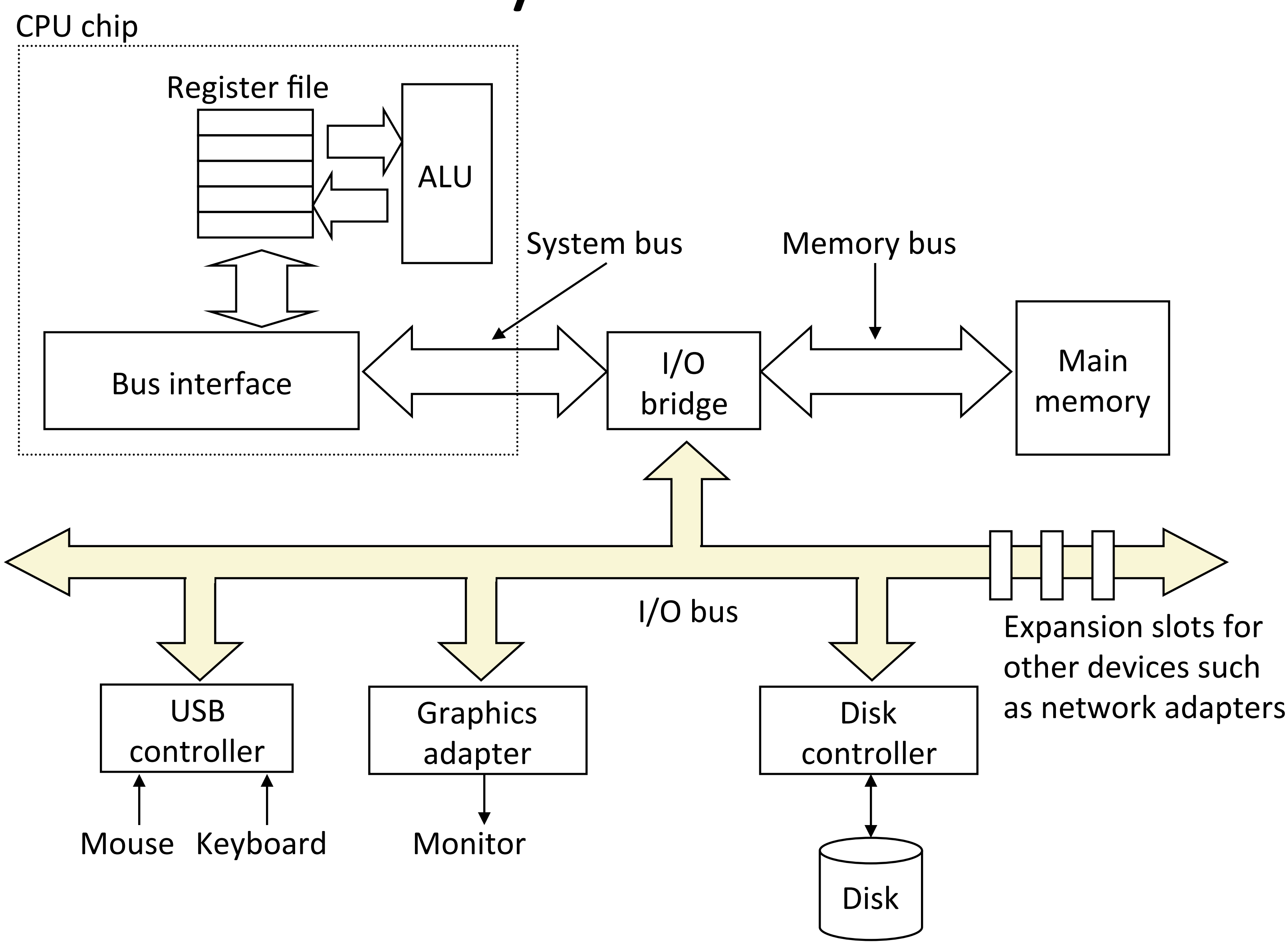
After BLUE read | Seek for RED | Rotational latency | After RED read

Data transfer | Seek | Rotational latency | Data transfer

Logical Disk Blocks

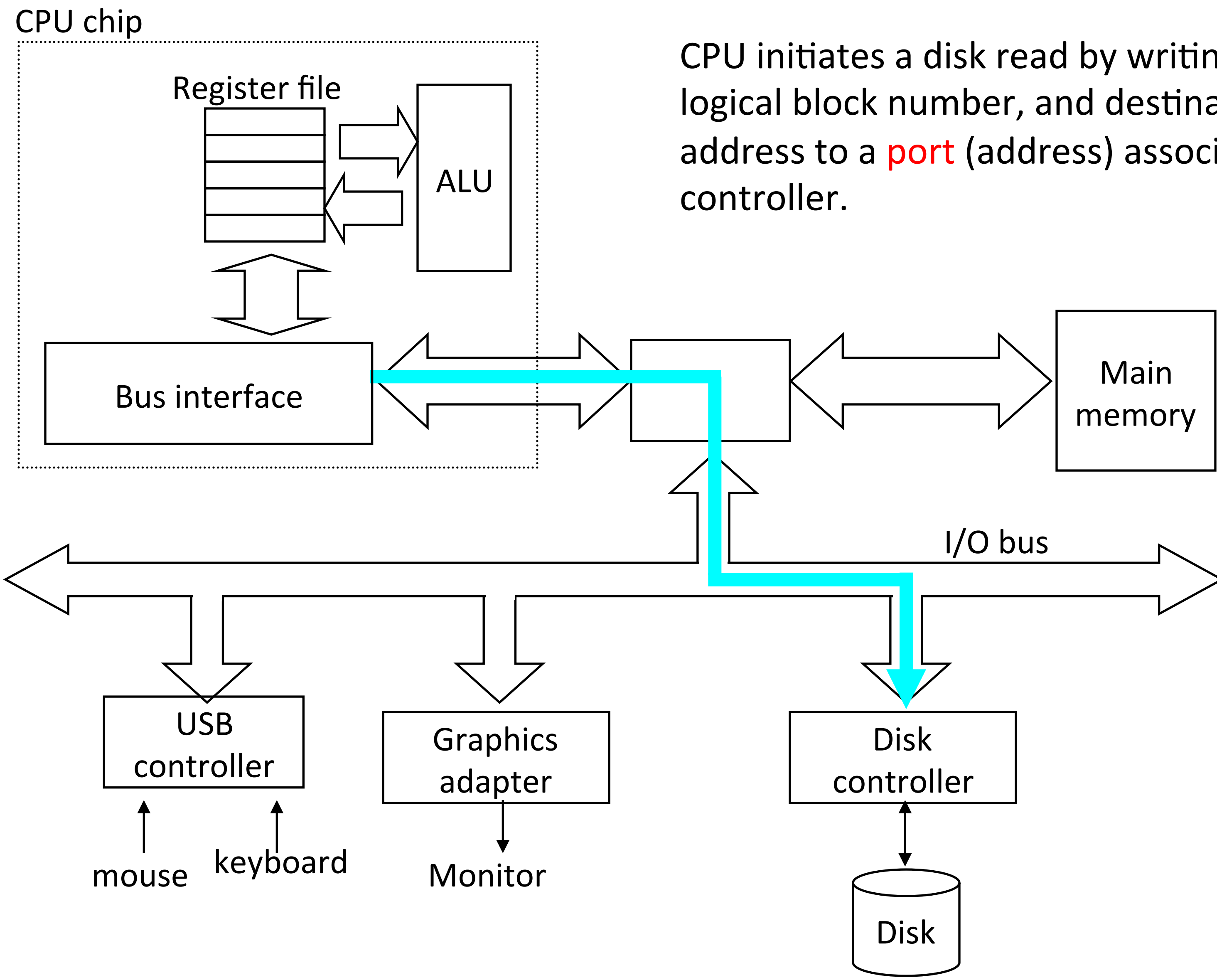
- Modern disks present a simpler abstract view of the complex sector geometry:
  - The set of available sectors is modeled as a sequence of b-sized logical blocks (0, 1, 2, ...)
- Mapping between logical blocks and actual (physical) sectors
  - Maintained by hardware/firmware device called disk controller.
  - Converts requests for logical blocks into (surface,track,sector) triples.
- Allows controller to set aside spare cylinders for each zone.
  - Accounts for the difference in "formatted capacity" and "maximum capacity".

I/O Bus


CPU chip
Register file
ALU
System bus
Memory bus
Bus interface
I/O bridge
Main memory
I/O bus
Expansion slots for other devices such as network adapters.
USB controller
Graphics adapter
Disk controller
Mouse
Keyboard
Monitor
Disk

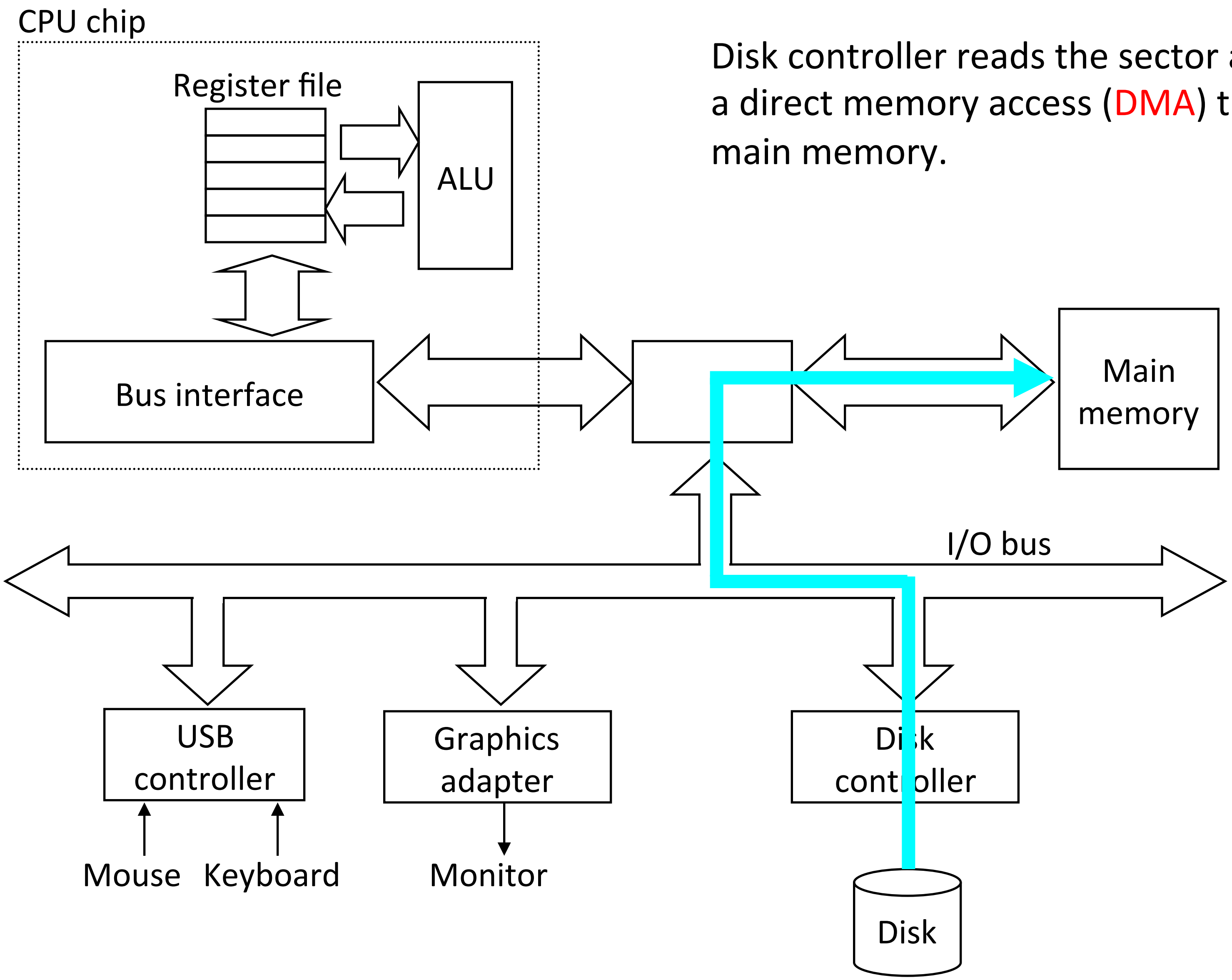
Reading a Disk Sector (1)

CPU initiates a disk read by writing a command, logical block number, and destination memory address to a port (address) associated with disk controller.



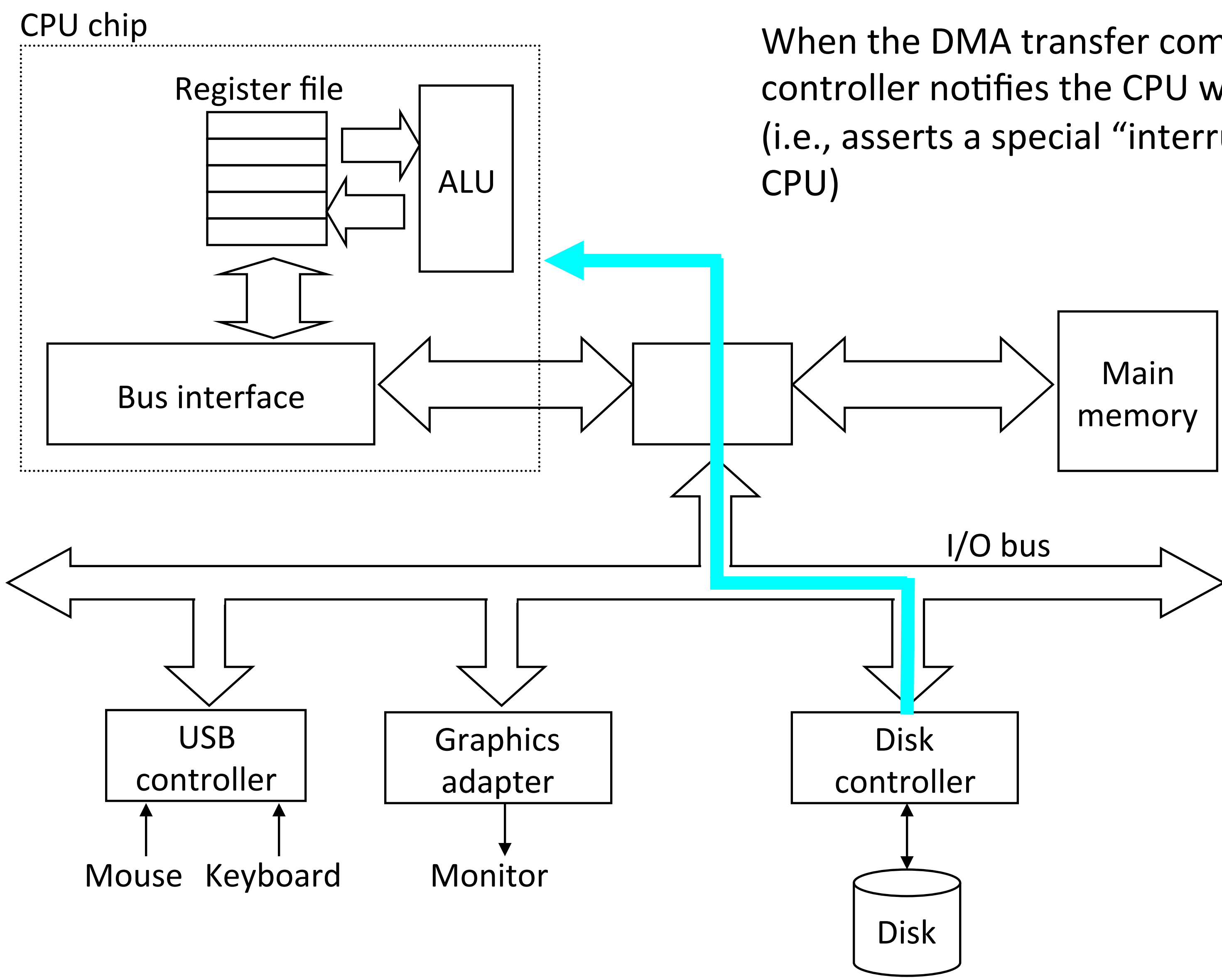
Reading a Disk Sector (2)

Disk controller reads the sector and performs a direct memory access (DMA) transfer into main memory.

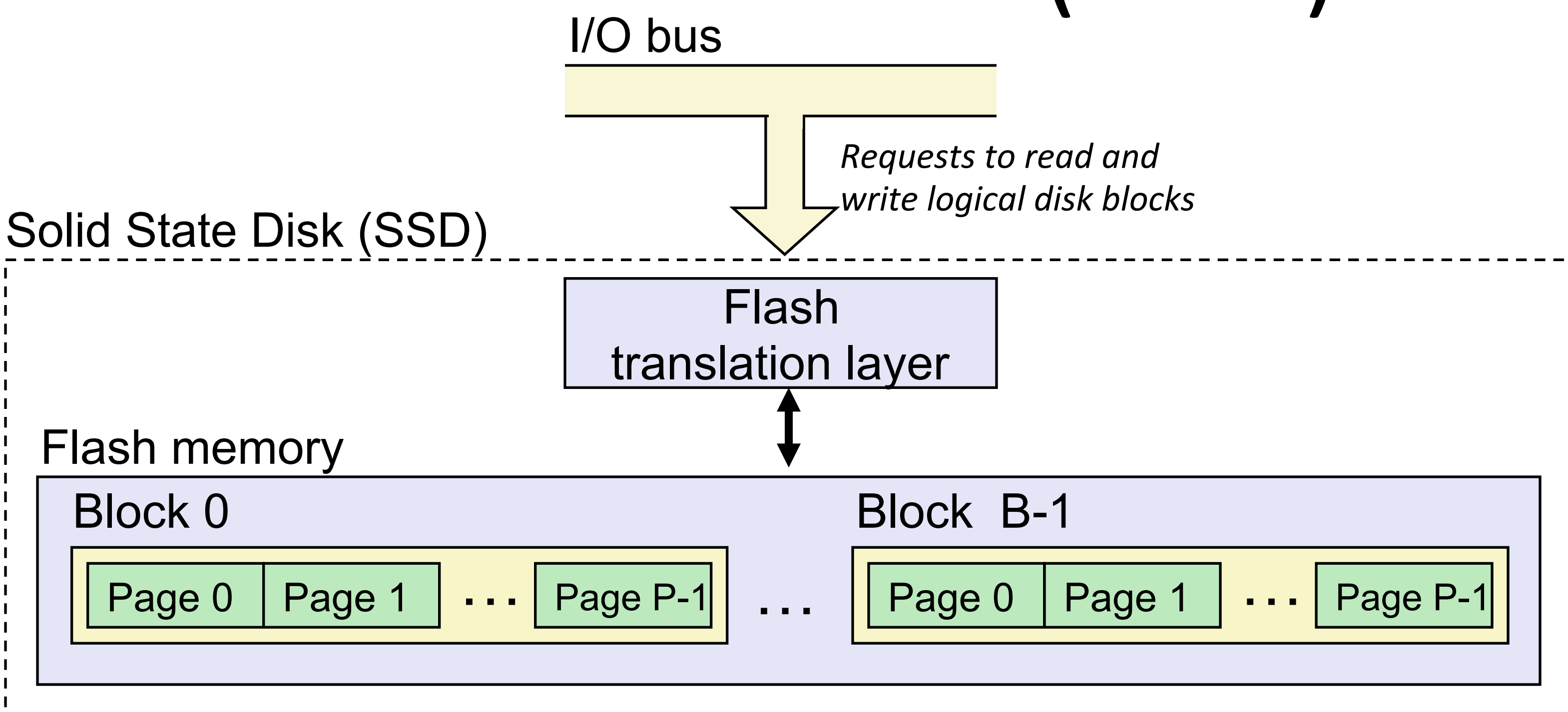


Reading a Disk Sector (3)

When the DMA transfer completes, the disk controller notifies the CPU with an *interrupt* (i.e., asserts a special "interrupt" pin on the CPU)



Solid State Disks (SSDs)



- Pages: 512KB to 4KB, Blocks: 32 to 128 pages
- Data read/written in units of pages.
- Page can be written only after its block has been erased
- A block wears out after 100,000 repeated writes.

SSD Performance Characteristics

Sequential read tput	250 MB/s	Sequential write tput	170 MB/s
Random read tput	140 MB/s	Random write tput	14 MB/s
Rand read access	30 us	Random write access	300 us

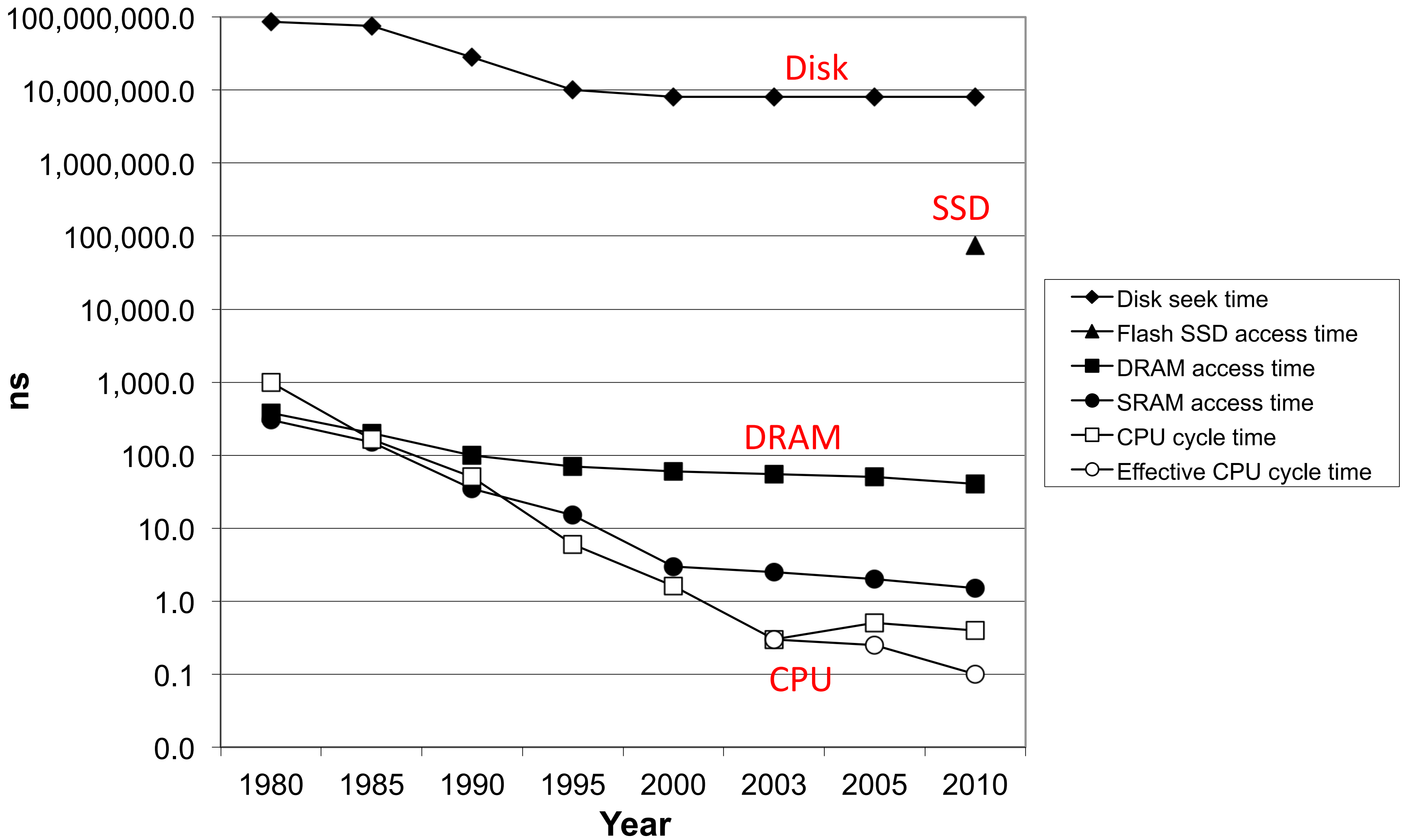
- Why are random writes so slow?
  - Erasing a block is slow (around 1 ms)
  - Write to a page triggers a copy of all useful pages in the block
    - Find an used block (new block) and erase it
    - Write the page into the new block
    - Copy other pages from old block to the new block

SSD Tradeoffs vs Rotating Disks

- Advantages
  - No moving parts → faster, less power, more rugged
- Disadvantages
  - Have the potential to wear out
    - Mitigated by “wear leveling logic” in flash translation layer
    - E.g. Intel X25 guarantees 1 petabyte (1015 bytes) of random writes before they wear out
  - In 2010, about 100 times more expensive per byte
- Applications
  - MP3 players, smart phones, laptops
  - Beginning to appear in desktops and servers

The CPU-Memory Gap

The gap widens between DRAM, disk, and CPU speeds.



What to do about it