

# CS 33

## Memory Hierarchy I

# Random-Access Memory (RAM)

- **Key features**
  - **RAM** is traditionally packaged as a chip
  - basic storage unit is normally a **cell** (one bit per cell)
  - multiple RAM chips form a memory
- **Static RAM (SRAM)**
  - each cell stores a bit with a four- or six-transistor circuit
  - retains value indefinitely, as long as it is kept powered
  - relatively insensitive to electrical noise (EMI), radiation, etc.
  - faster and more expensive than DRAM
- **Dynamic RAM (DRAM)**
  - each cell stores bit with a capacitor; transistor is used for access
  - value must be refreshed every 10-100 ms
  - more sensitive to disturbances (EMI, radiation,...) than SRAM
  - slower and cheaper than SRAM

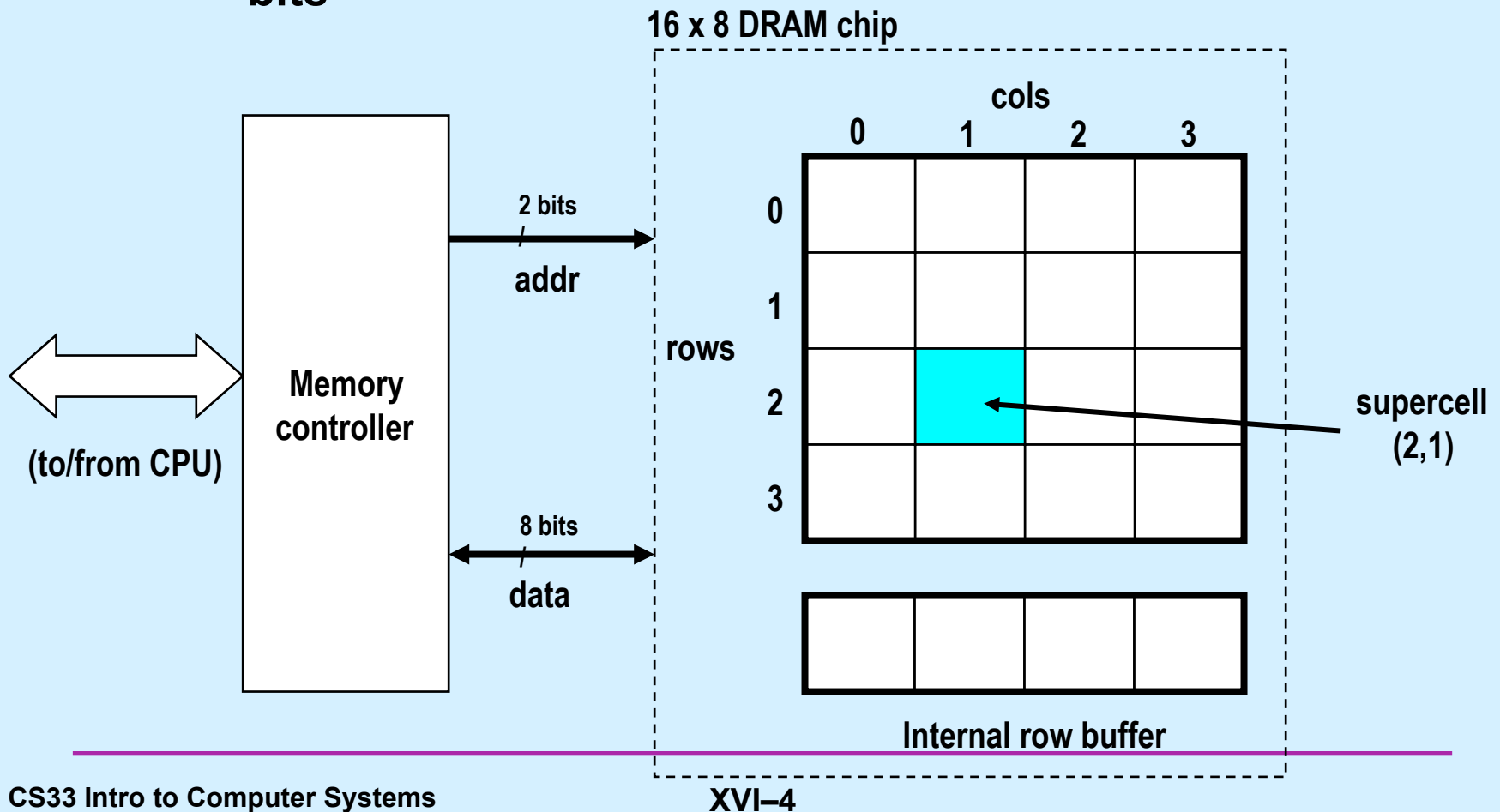
# SRAM vs DRAM Summary

	Trans. per bit	Access time	Needs refresh?	Needs EDC?	Cost	Applications
SRAM	4 or 6	1X	No	Maybe	100x	Cache memories
DRAM	1	10X	Yes	Yes	1X	Main memories, frame buffers

- **EDC = error detection and correction**
  - to cope with noise, etc.

# Conventional DRAM Organization

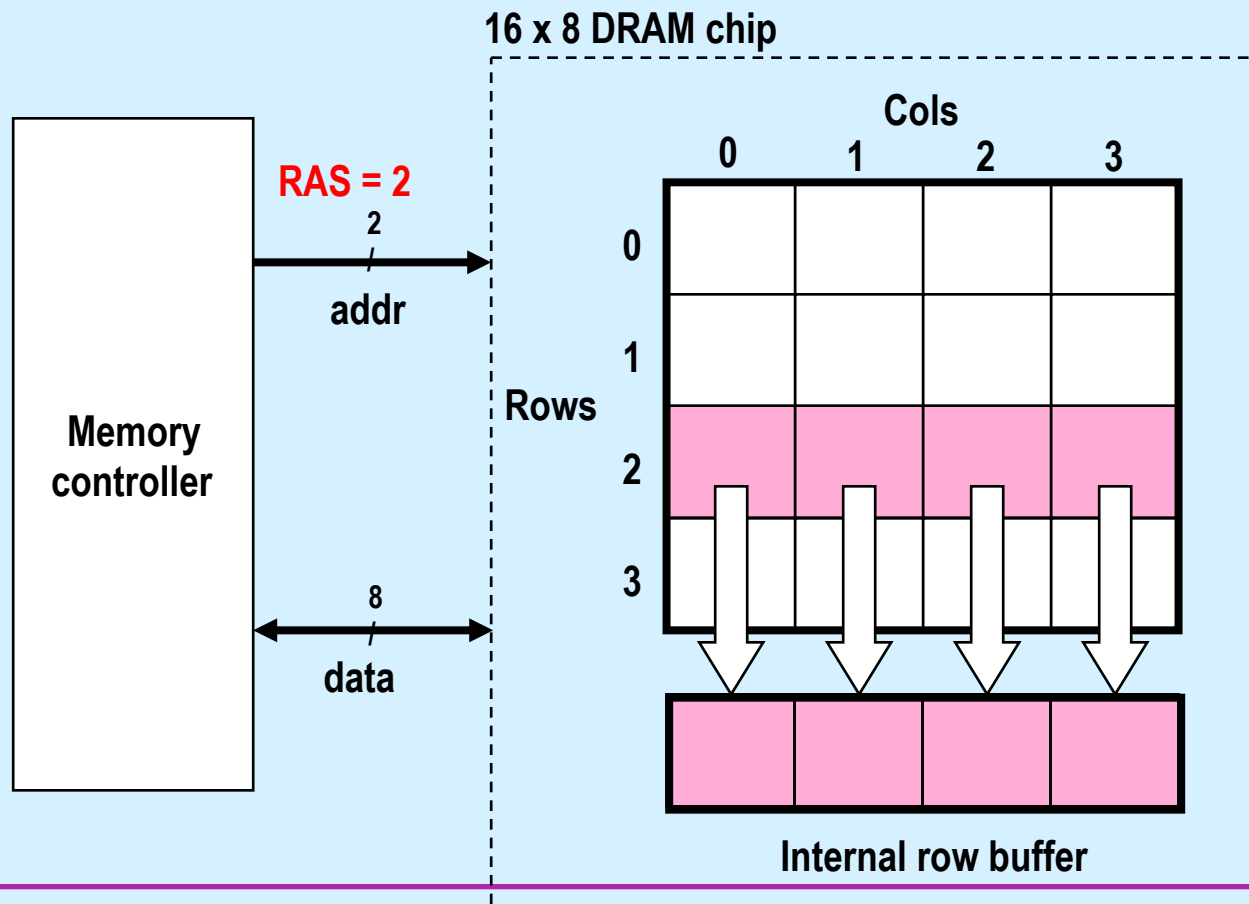
- $d \times w$  DRAM:
  - $dw$  total bits organized as  $d$  **supercells** of size  $w$  bits



# Reading DRAM Supercell (2,1)

Step 1(a): row access strobe (**RAS**) selects row 2

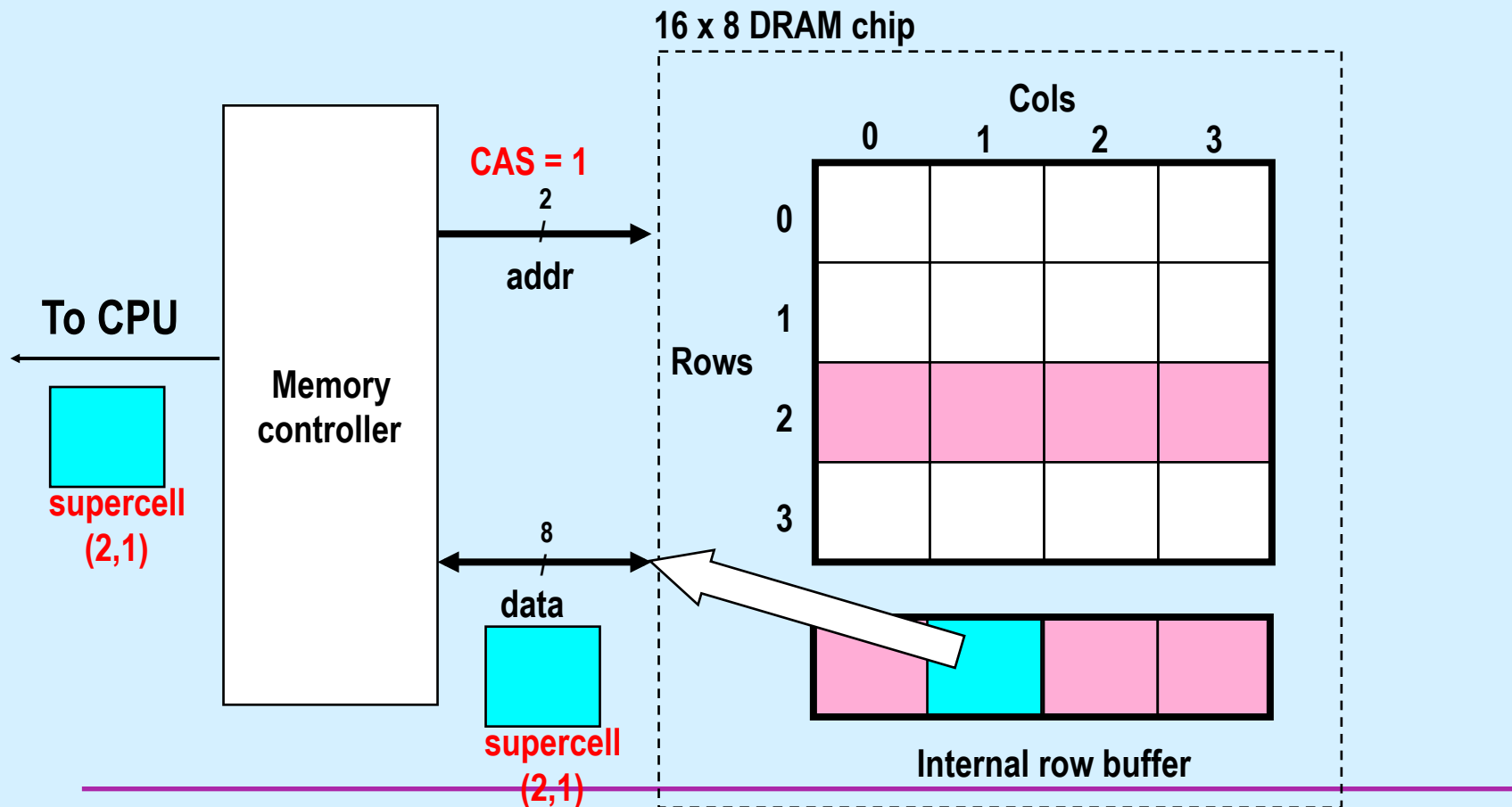
Step 1(b): row 2 copied from DRAM array to row buffer



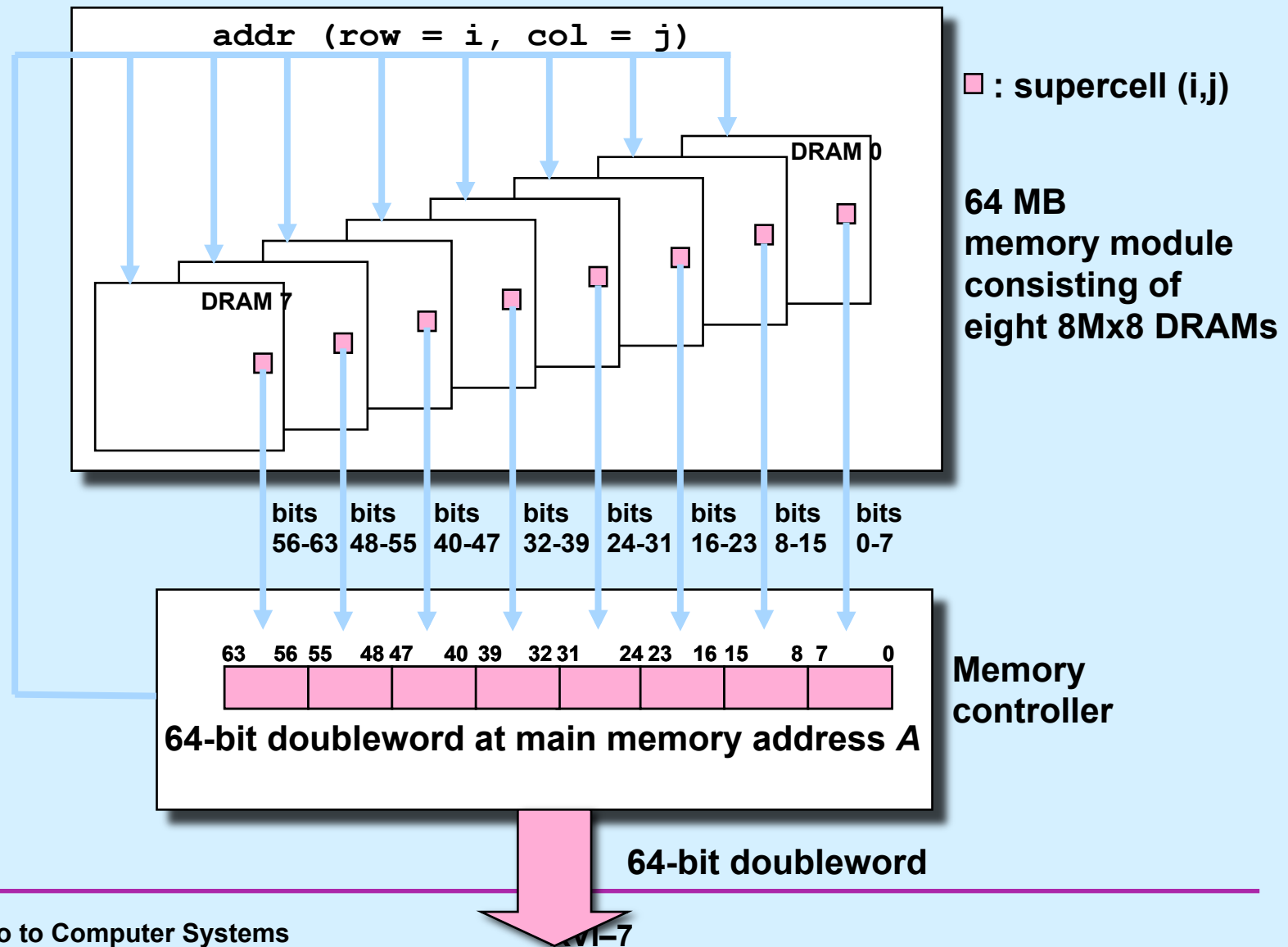
# Reading DRAM Supercell (2,1)

Step 2(a): column access strobe (**CAS**) selects column 1

Step 2(b): supercell (2,1) copied from buffer to data lines, and eventually back to the CPU



# Memory Modules

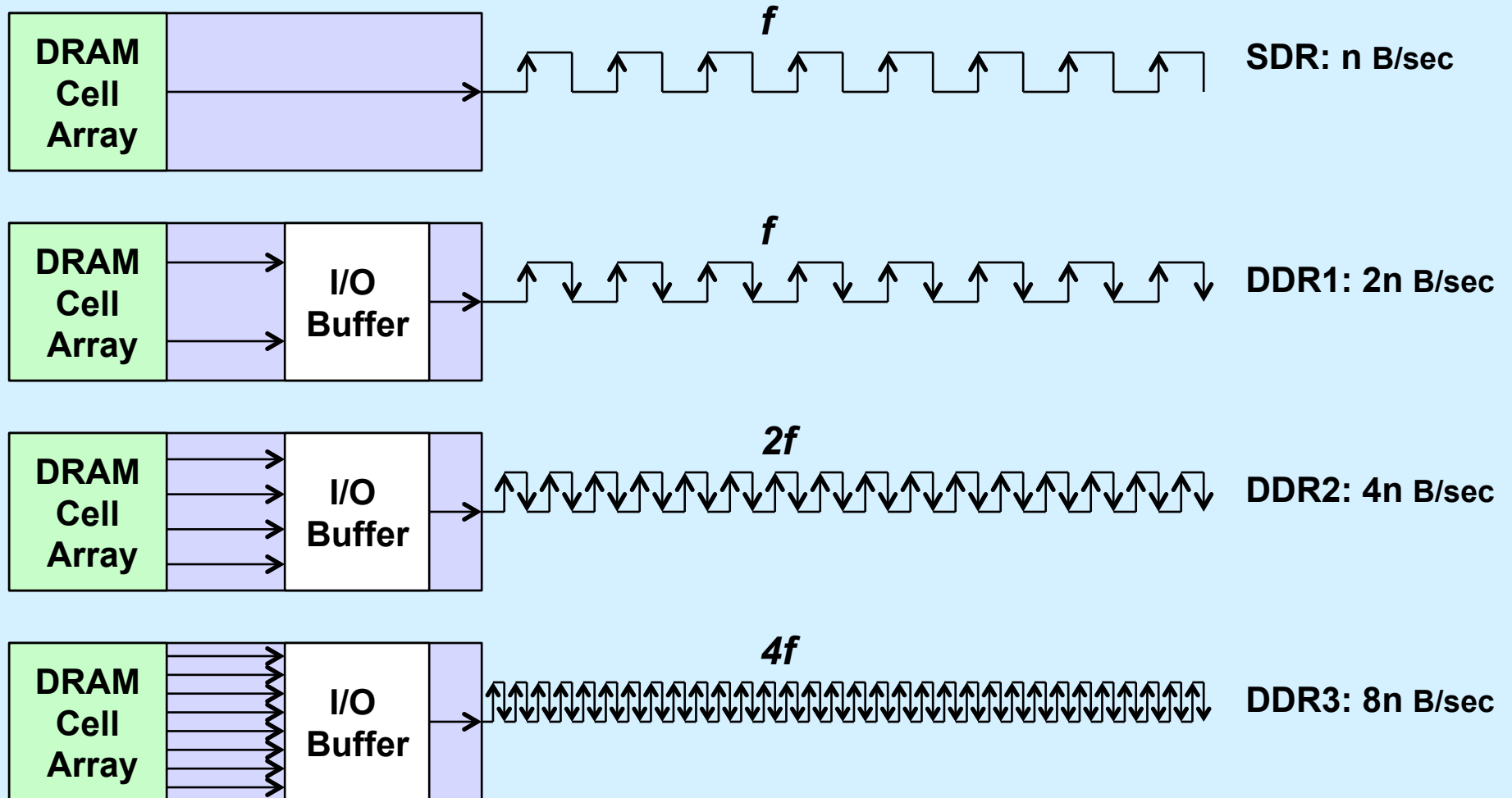


# Enhanced DRAMs

- **Basic DRAM cell has not changed since its invention in 1966**
  - commercialized by Intel in 1970
- **DRAM cores with better interface logic and faster I/O:**
  - **synchronous DRAM (SDRAM)**
    - » uses a conventional clock signal instead of asynchronous control
    - » allows reuse of the row addresses (e.g., RAS, CAS, CAS, CAS)
  - **double data-rate synchronous DRAM (DDR SDRAM)**
    - » **DDR1**
      - twice as fast
    - » **DDR2**
      - four times as fast
    - » **DDR3**
      - eight times as fast



# Enhanced DRAMs



# Quiz 1

**A program is loading randomly selected bytes from memory. These bytes will be delivered to the processor on a DDR3 system  $n$  times faster than on an SDR system, where  $n$  is:**

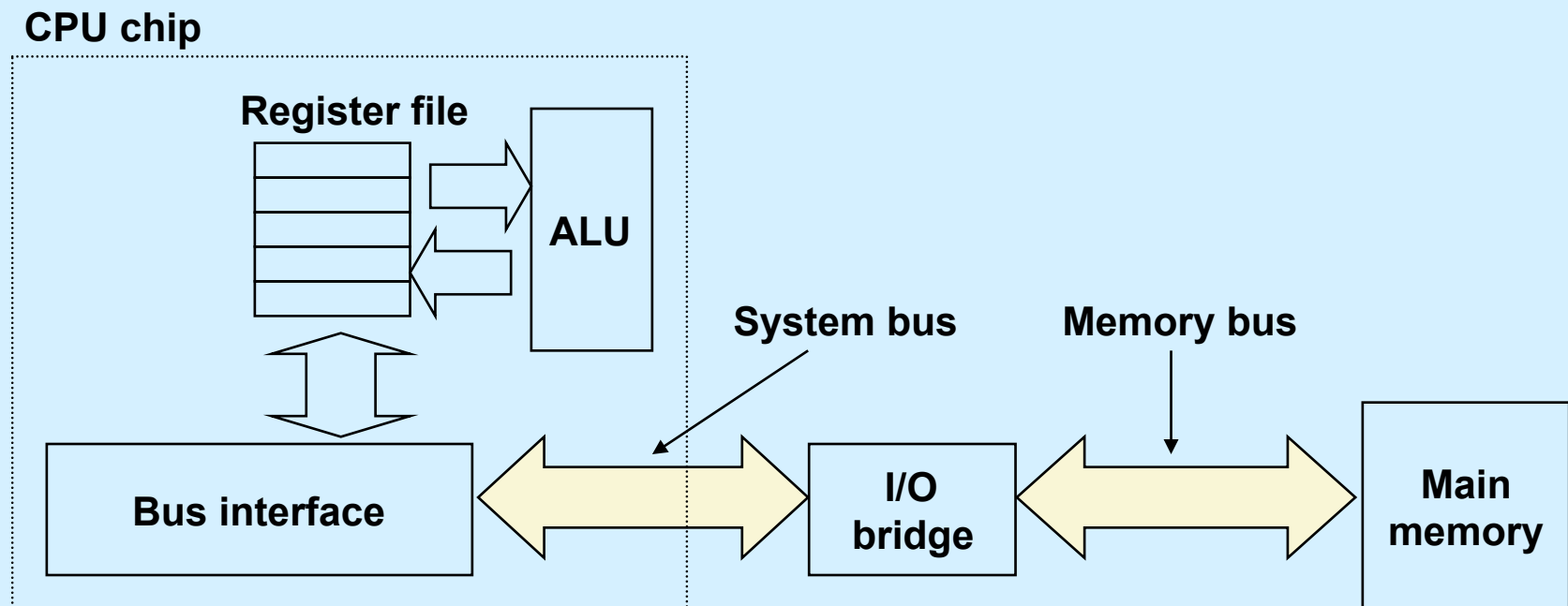
- a) 1**
- b) 2**
- c) 4**
- d) 8**

# Nonvolatile Memories

- **DRAM and SRAM are volatile memories**
  - lose information if powered off
- **Nonvolatile memories retain value even if powered off**
  - read-only memory (**ROM**): programmed during production
  - programmable ROM (**PROM**): can be programmed once
  - eraseable PROM (**EPROM**): can be bulk erased (UV, X-Ray)
  - electrically eraseable PROM (**EEPROM**): electronic erase capability
  - flash memory: EEPROMs with partial (sector) erase capability
    - » wears out after about 100,000 erasings
- **Uses for nonvolatile memories**
  - firmware programs stored in a ROM (BIOS, controllers for disks, network cards, graphics accelerators, security subsystems,...)
  - solid state disks (replace rotating disks in thumb drives, smart phones, mp3 players, tablets, laptops,...)
  - disk caches

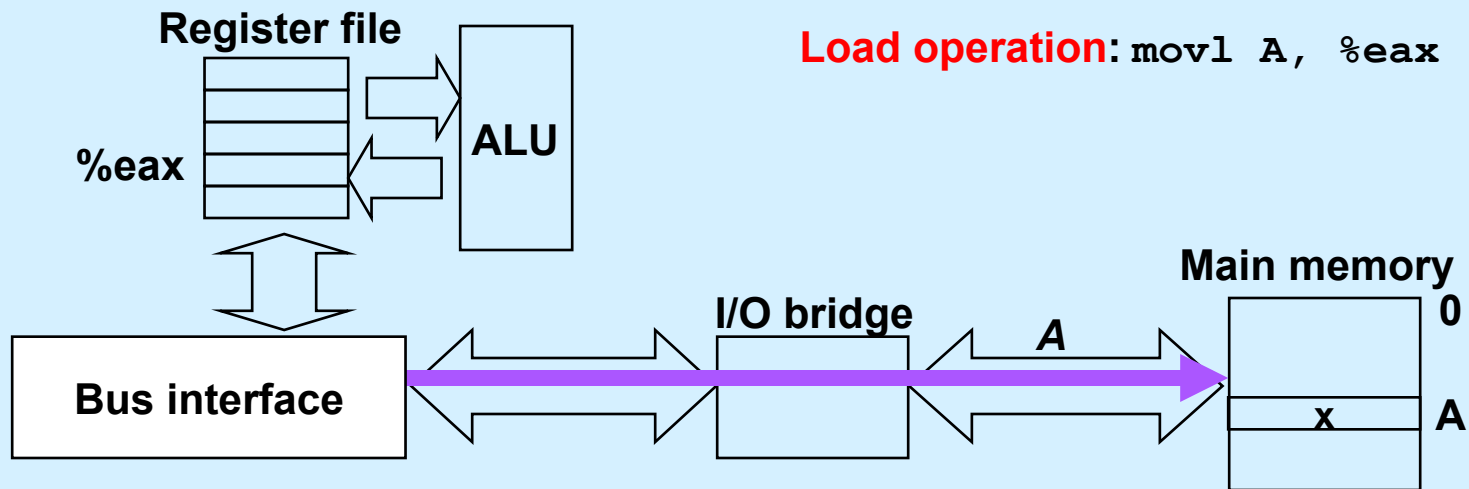
# Traditional Bus Structure Connecting CPU and Memory

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



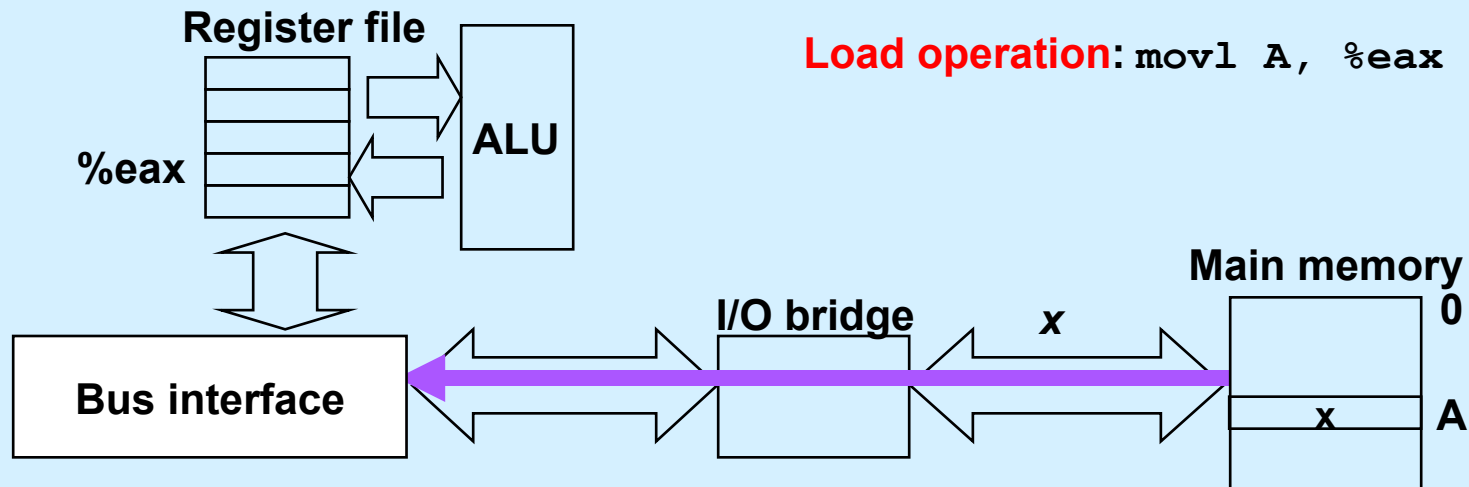
# Memory Read Transaction (1)

- CPU places address A on the memory bus



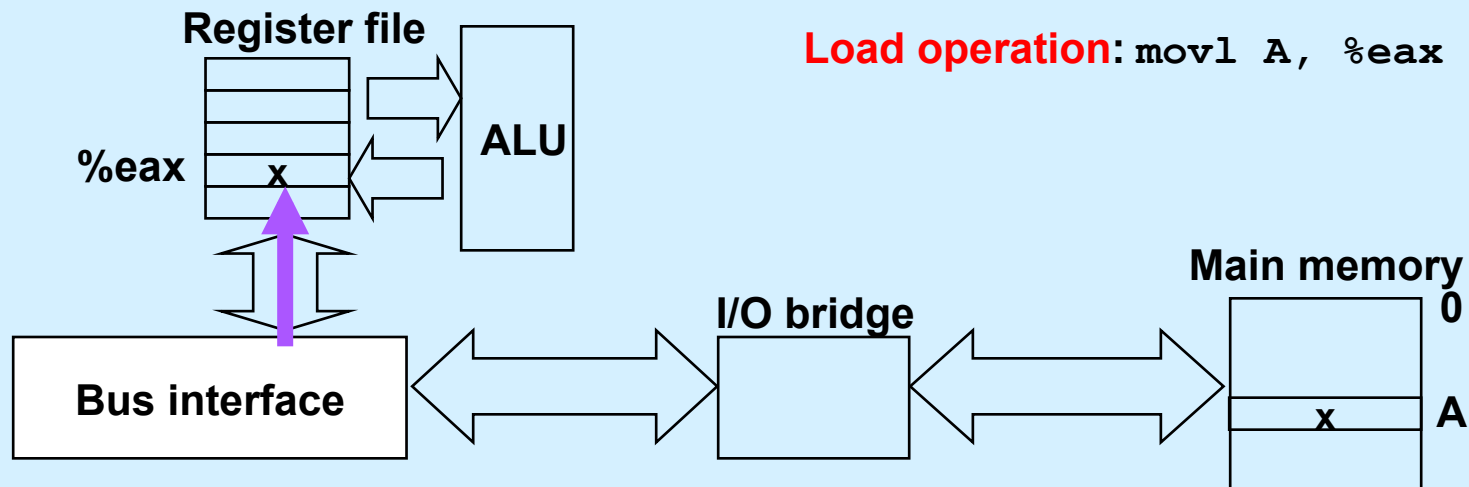
# Memory Read Transaction (2)

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



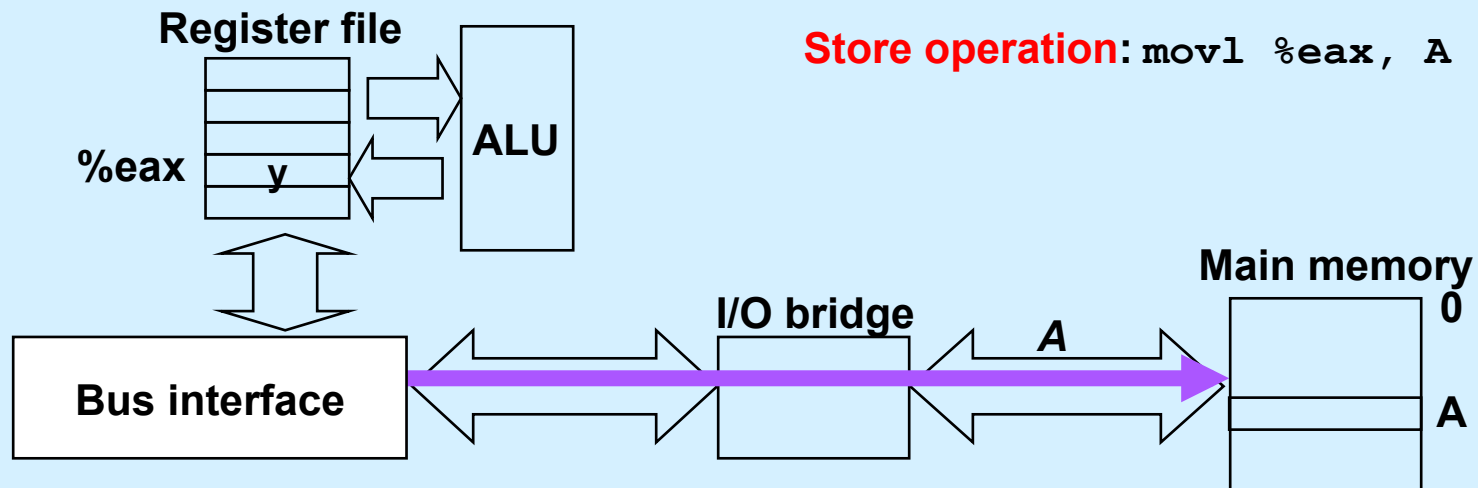
# Memory Read Transaction (3)

- CPU reads word  $x$  from the bus and copies it into register `%eax`



# Memory Write Transaction (1)

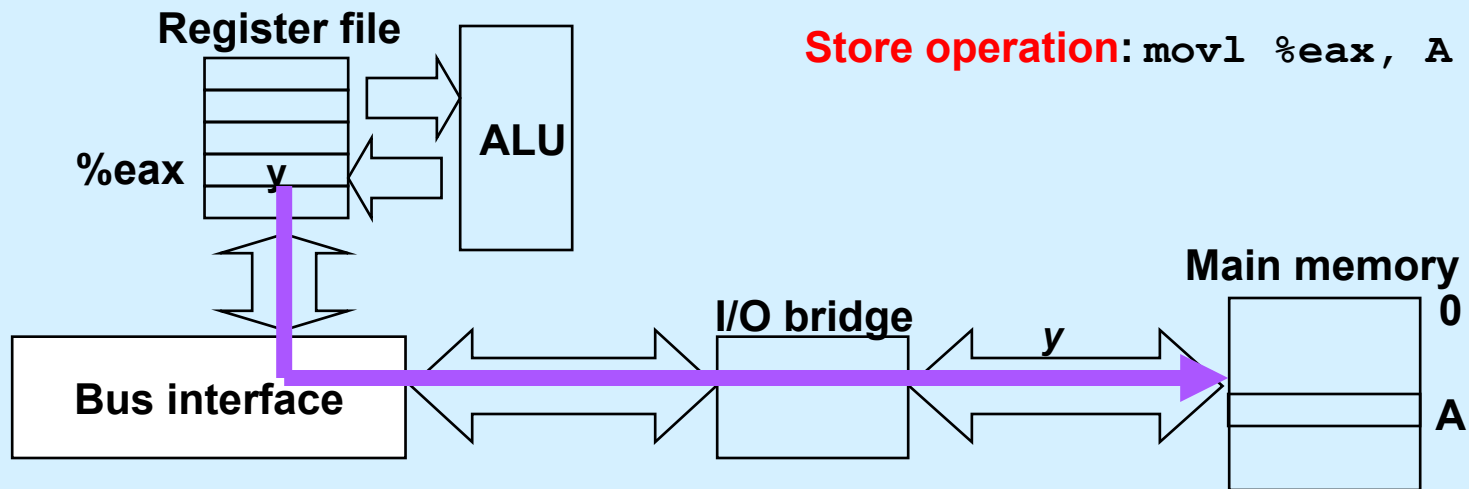
- CPU places address A on bus. Main memory reads it and waits for the corresponding data word to arrive





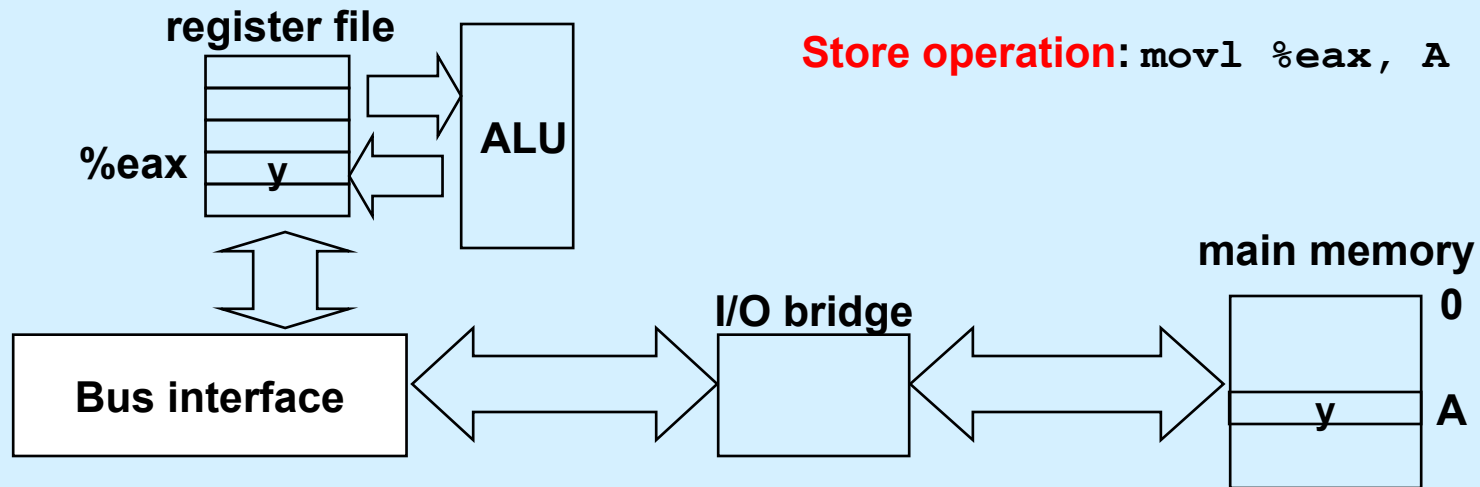
# Memory Write Transaction (2)

- CPU places data word  $y$  on the bus

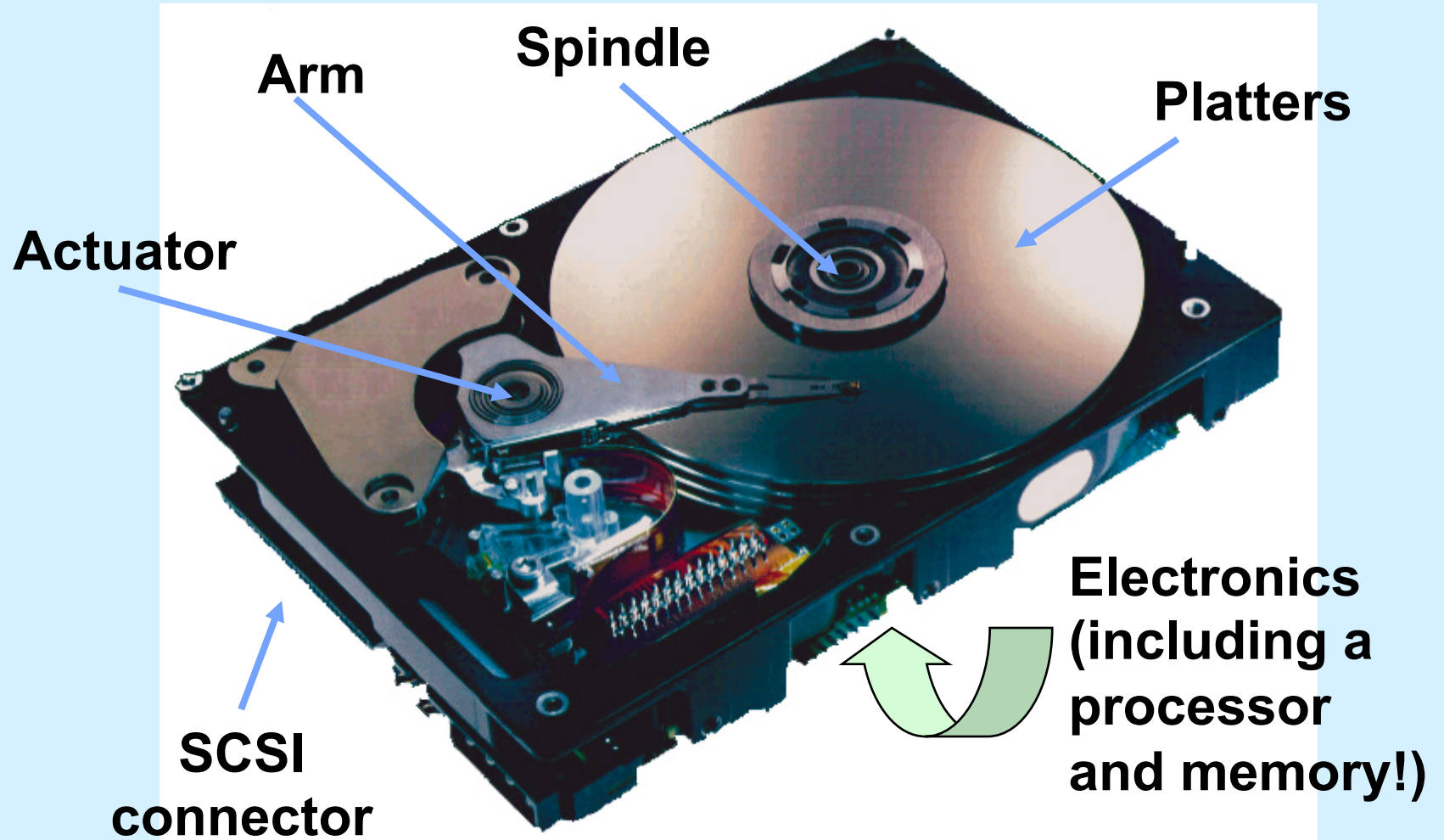


# Memory Write Transaction (3)

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



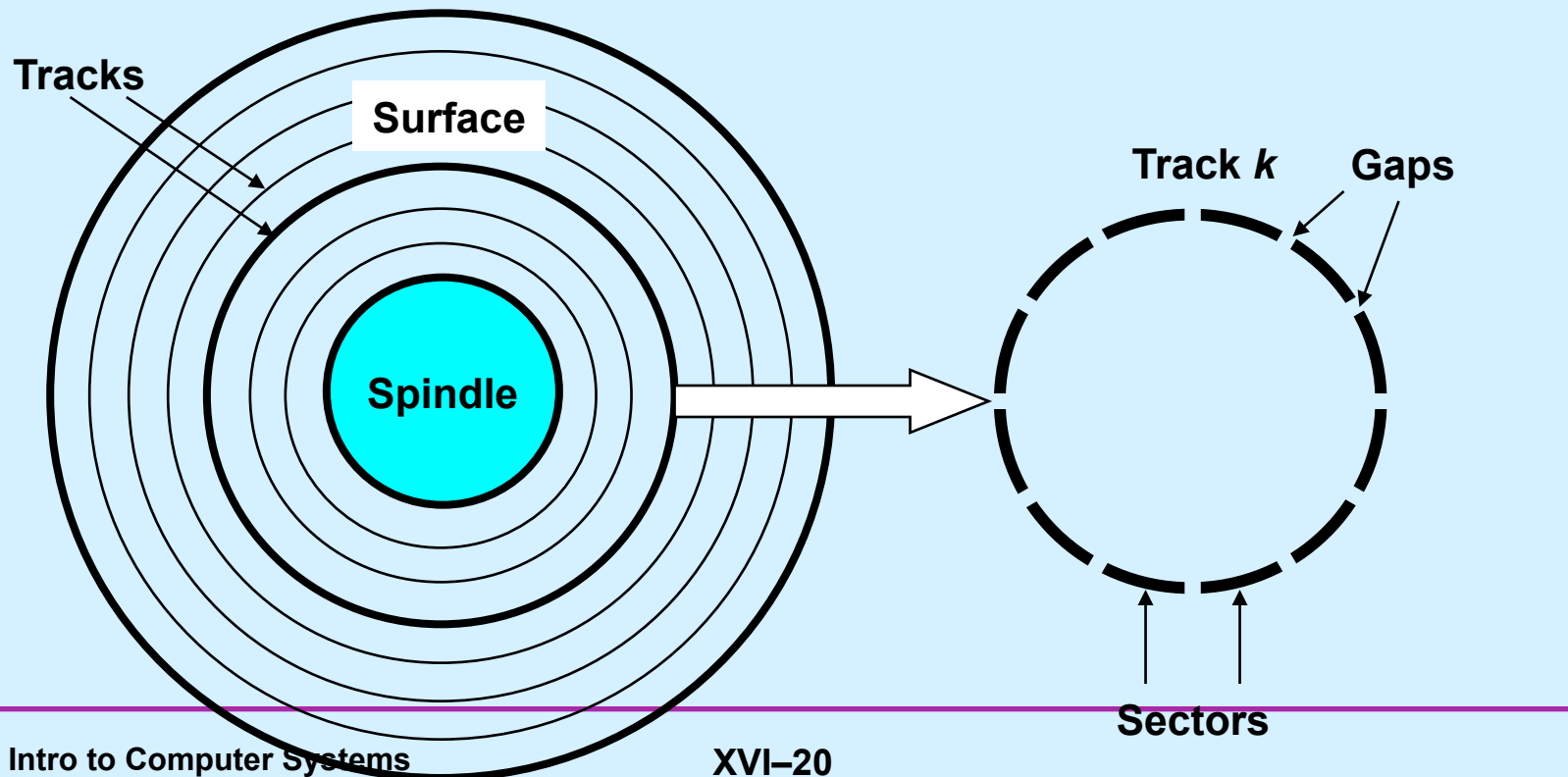
# What's Inside A Disk Drive?



*Image courtesy of Seagate Technology*

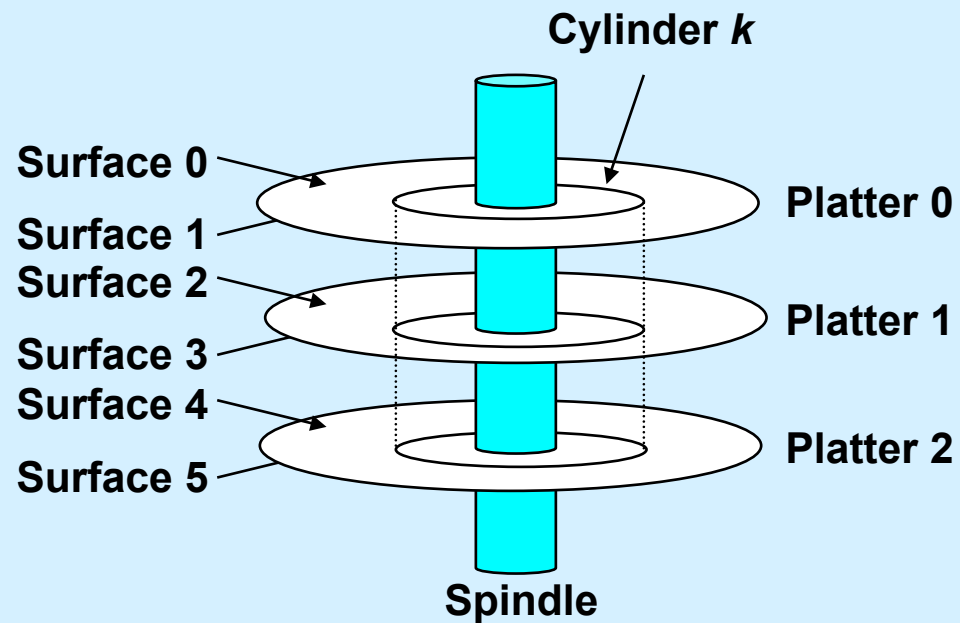
# Disk Geometry

- 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 (Multiple-Platter View)

- Aligned tracks form a cylinder



# Disk Capacity

- **Capacity**: maximum number of bits that can be stored
  - capacity expressed in units of gigabytes (GB), where  $1 \text{ GB} = 2^{30} \text{ Bytes} \approx 10^9 \text{ Bytes}$
- Capacity is determined by these technology factors:
  - **recording density** (bits/in): number of bits that can be squeezed into a 1 inch segment of a track
  - **track density** (tracks/in): number of tracks that can be squeezed into a 1 inch radial segment
  - **areal density** (bits/in<sup>2</sup>): product of recording and track density
- Modern disks partition tracks into disjoint subsets called **recording zones**
  - each track in a zone has the same number of sectors, determined by the circumference of innermost track
  - each zone has a different number of sectors/track

# Computing Disk Capacity

$$\text{Capacity} = (\# \text{ bytes/sector}) \times (\text{avg. } \# \text{ sectors/track}) \times (\# \text{ tracks/surface}) \times (\# \text{ surfaces/platter}) \times (\# \text{ platters/disk})$$

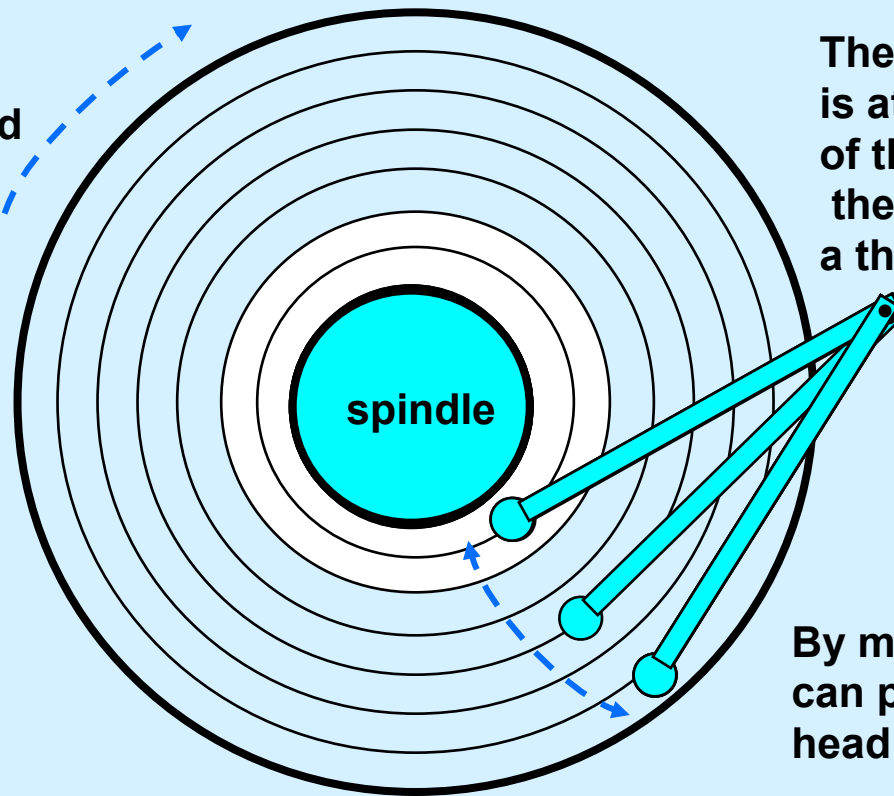
**Example:**

- 512 bytes/sector
- 600 sectors/track (on average)
- 40,000 tracks/surface
- 2 surfaces/platter
- 5 platters/disk

$$\begin{aligned}\text{Capacity} &= 512 \times 600 \times 40000 \times 2 \times 5 \\ &= 122,280,000,000 \\ &= 113.88 \text{ GB}\end{aligned}$$

# Disk Operation (Single-Platter View)

The disk surface spins at a fixed rotational rate

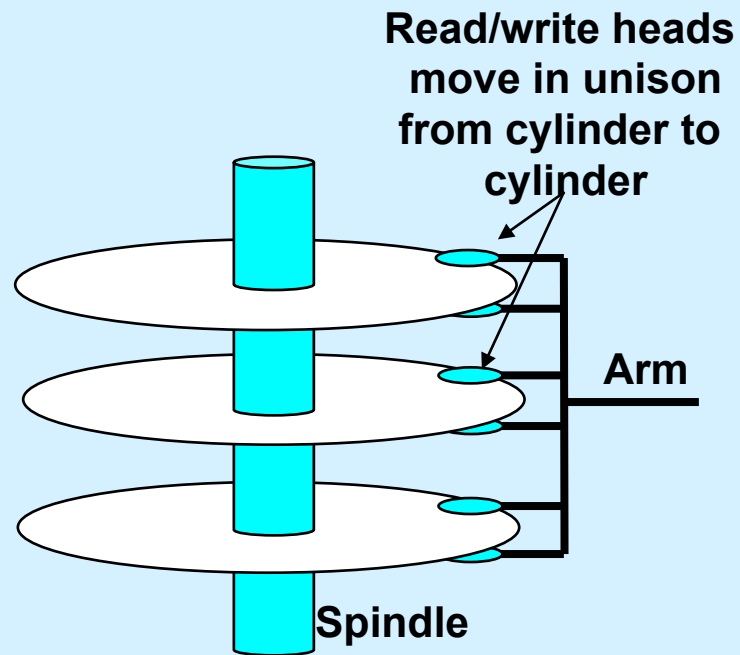


The read/write *head* is attached to the end of the *arm* and flies over the disk surface on a thin cushion of air

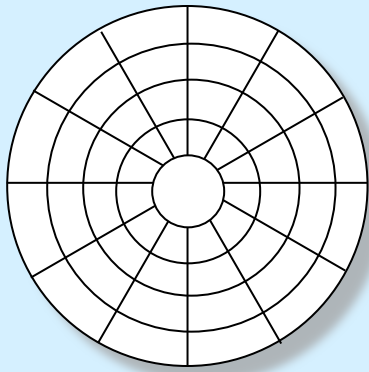
By moving radially, the arm can position the read/write head over any track



# Disk Operation (Multi-Platter View)



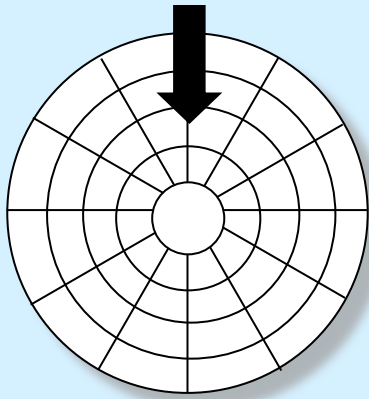
# Disk Structure: Top View of Single Platter



**Surface organized into tracks**

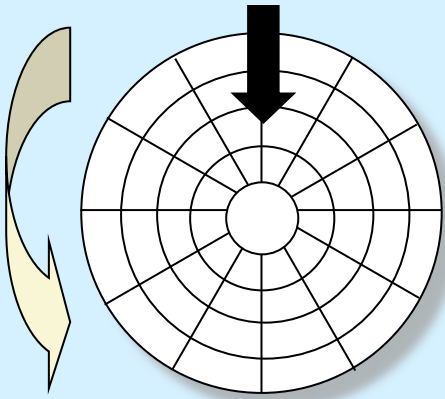
**Tracks divided into sectors**

# Disk Access



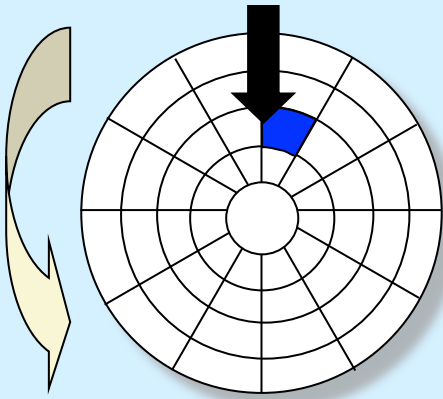
**Head in position above a track**

# Disk Access



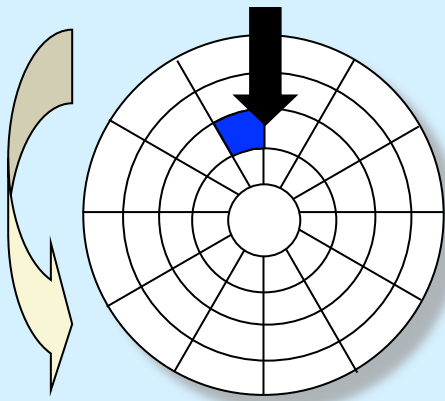
**Rotation is counter-clockwise**

# Disk Access – Read



**About to read blue sector**

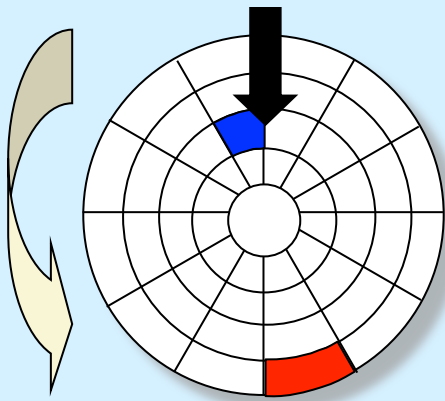
# Disk Access – Read



After **BLUE**  
read

After reading blue sector

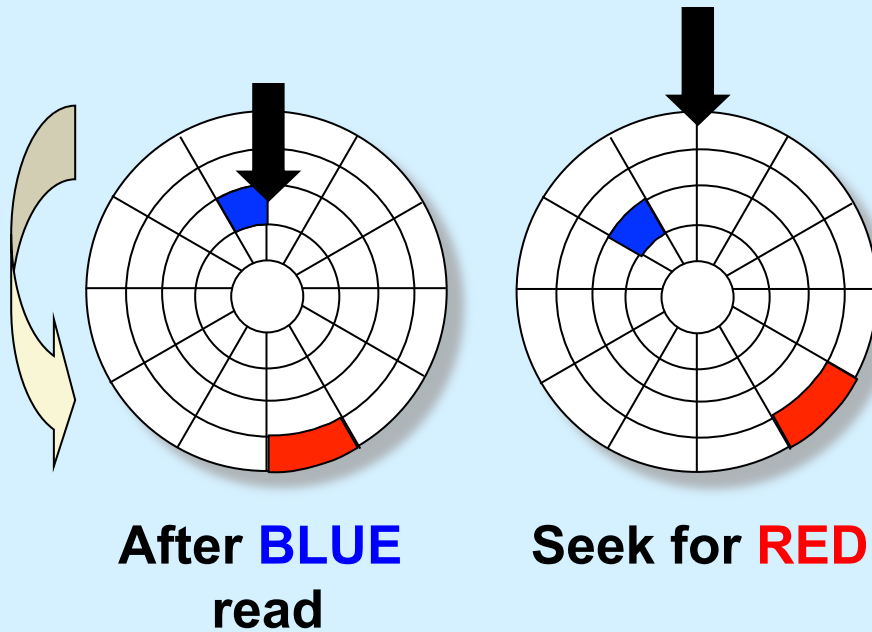
# Disk Access – Read



After **BLUE**  
read

**Red request scheduled next**

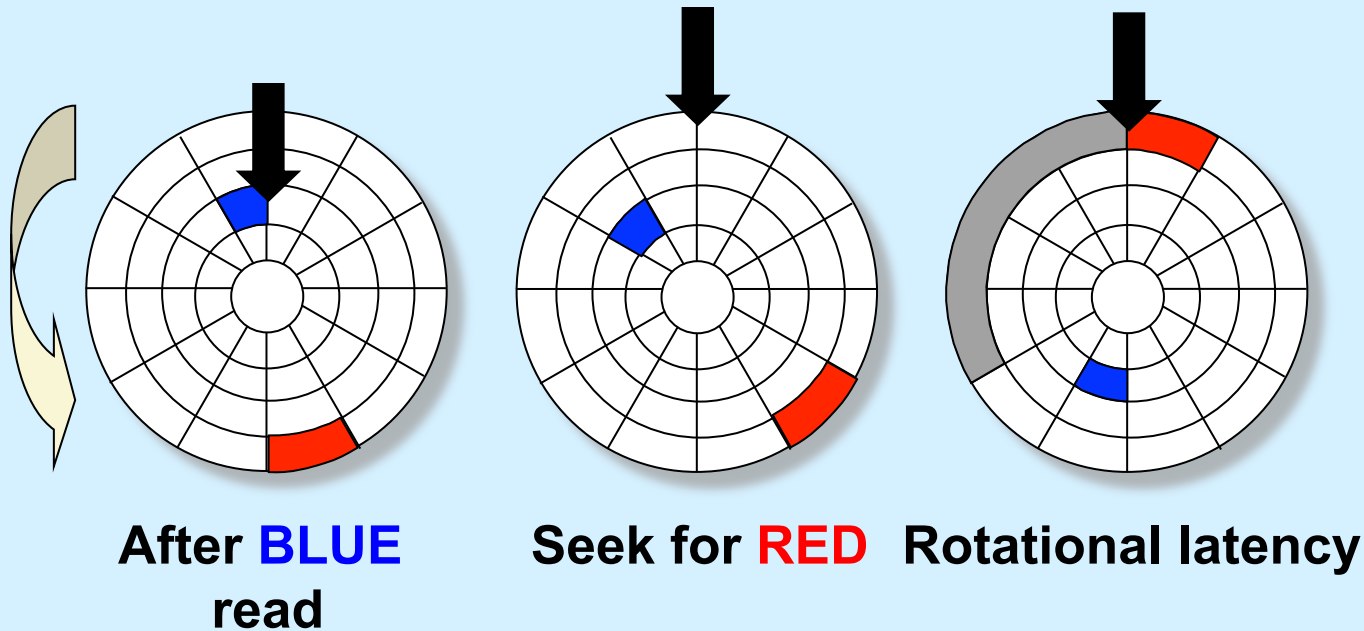
# Disk Access – Seek



**Seek to red's track**

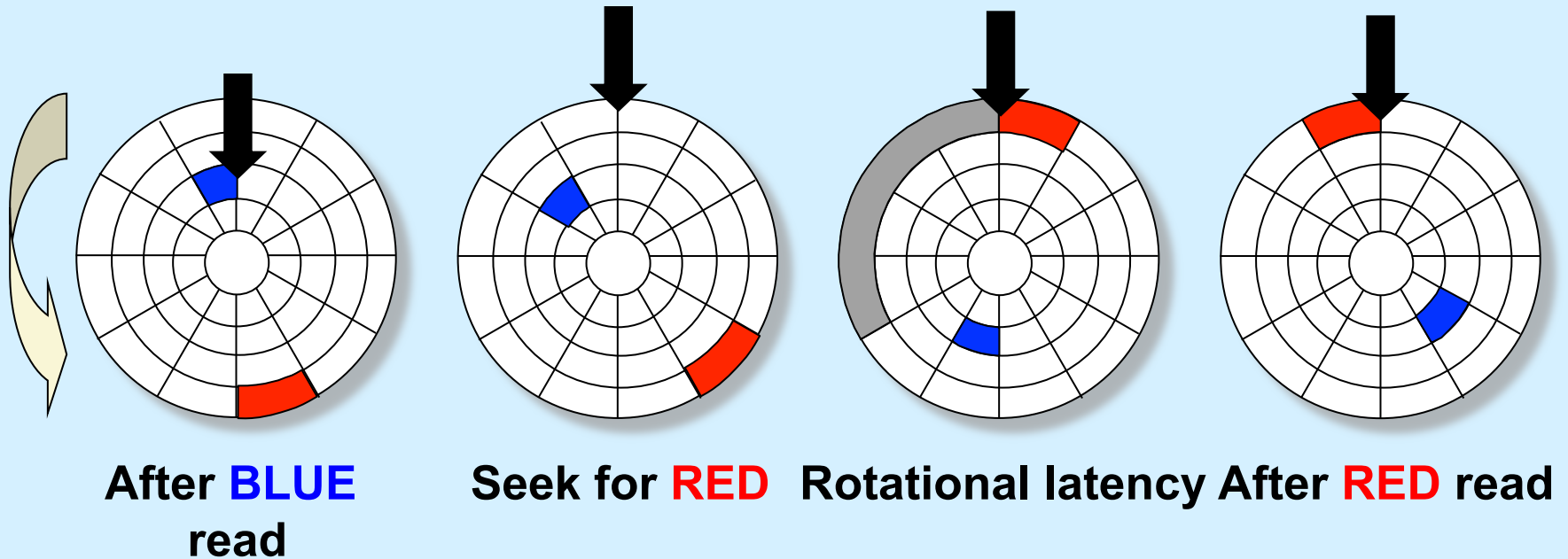


# Disk Access – Rotational Latency



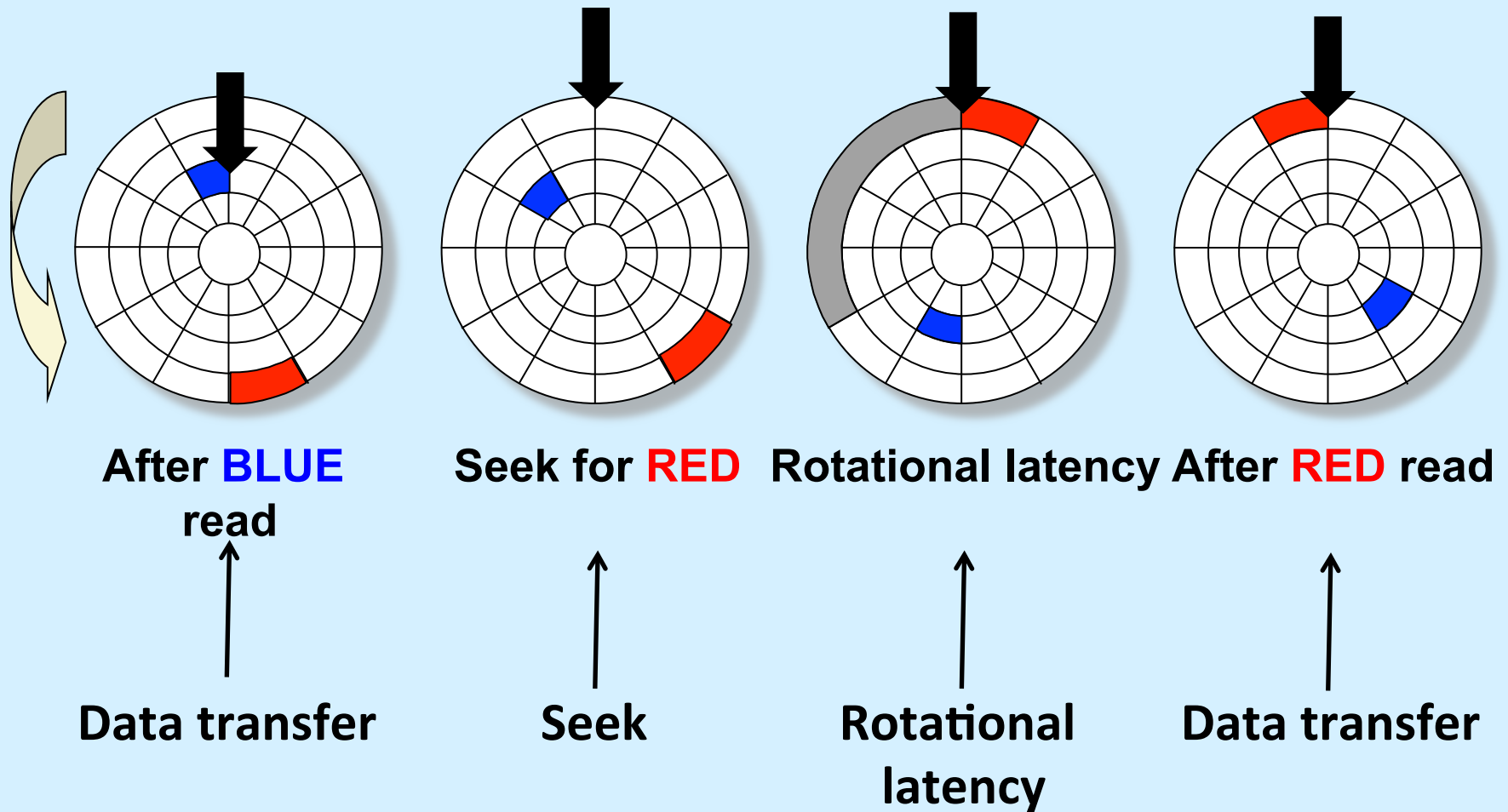
**Wait for red sector to rotate around**

# Disk Access – Read



**Complete read of red**

# Disk Access – Service Time Components



# Disk Access Time

- Average time to access some target sector approximated by :
  - $T_{\text{access}} = T_{\text{avg seek}} + T_{\text{avg rotation}} + T_{\text{avg transfer}}$
- **Seek time** ( $T_{\text{avg seek}}$ )
  - time to position heads over cylinder containing target sector
  - typical  $T_{\text{avg seek}}$  is 3–9 ms
- **Rotational latency** ( $T_{\text{avg rotation}}$ )
  - time waiting for first bit of target sector to pass under r/w head
  - typical rotation speed  $R = 7200$  RPM
  - $T_{\text{avg rotation}} = \frac{1}{2} \times \frac{1}{R} \times 60 \text{ sec/1 min}$
- **Transfer time** ( $T_{\text{avg transfer}}$ )
  - time to read the bits in the target sector
  - $T_{\text{avg transfer}} = \frac{1}{R} \times \frac{1}{(\text{avg \# sectors/track})} \times 60 \text{ secs/1 min}$

# Disk Access Time Example

- **Given:**
  - rotational rate = 7,200 RPM
  - average seek time = 9 ms
  - avg # sectors/track = 600
- **Derived:**
  - $T_{\text{avg rotation}} = 1/2 \times (60 \text{ secs}/7200 \text{ RPM}) \times 1000 \text{ ms/sec} = 4 \text{ ms}$
  - $T_{\text{avg transfer}} = 60/7200 \text{ RPM} \times 1/600 \text{ sects/track} \times 1000 \text{ ms/sec} = 0.014 \text{ ms}$
  - $T_{\text{access}} = 9 \text{ ms} + 4 \text{ ms} + 0.014 \text{ ms}$
- **Important points:**
  - access time dominated by seek time and rotational latency
  - first bit in a sector is the most expensive, the rest are free
  - SRAM access time is about 4 ns/doubleword, DRAM about 60 ns
    - » disk is about 40,000 times slower than SRAM
    - » 2,500 times slower than DRAM

## Quiz 2

**Assuming a 5-inch diameter disk spinning at 10,000 RPM, what is the approximate speed at which the outermost track is moving?**

- a) faster than a speeding bullet (i.e., supersonic)**
- b) roughly the speed of a pretty fast car (250 kph/155 mph)**
- c) roughly the speed of a pretty slow car (50 mph)**
- d) roughly the speed of a world-class marathoner (13.1 mph)**