Chapter 08: The Memory System

Lesson 17:

Auxiliary memory Peripheral Devices — Secondary Storage Magnetic Ferrite Core Memories

Objective

- Understand how several disks can share an IO bus
- Understand hard disk organisation as multiple platters with each platter separate head and each platter number of tracks and sectors
- Learn three request scheduling mechanisms for read or write: First-come-first-serve (FCFS), Shortest-seek-time-first (SSTF) and LOOK scheduling

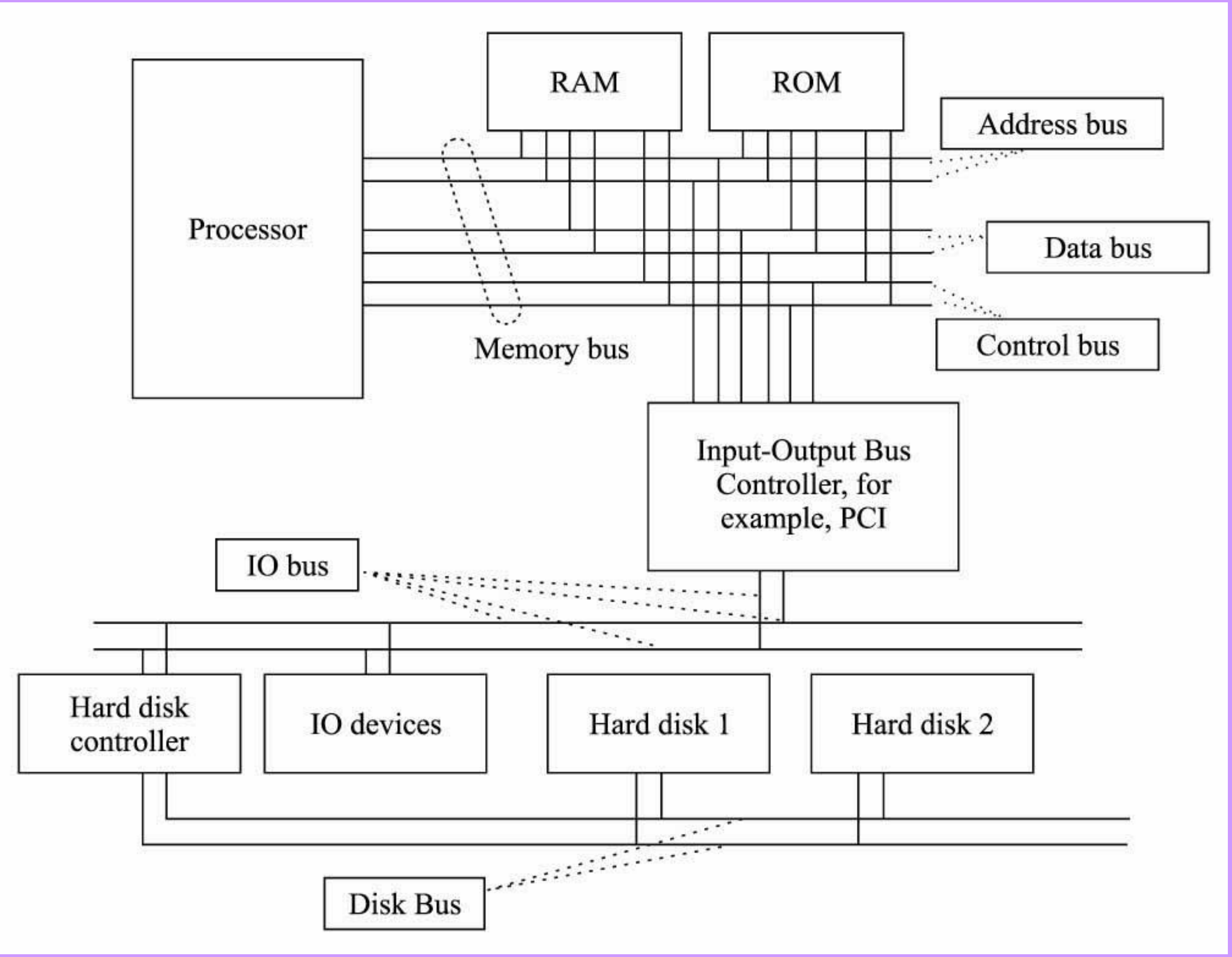
Memory systems

Secondary storage

- One of the most critical components of memory
- Supports virtual memory
- Supports the large permanent storage
- A typical disk subsystem attaches to I/O buses
- Several disks can be shared by the buses

Disk Memory in systems

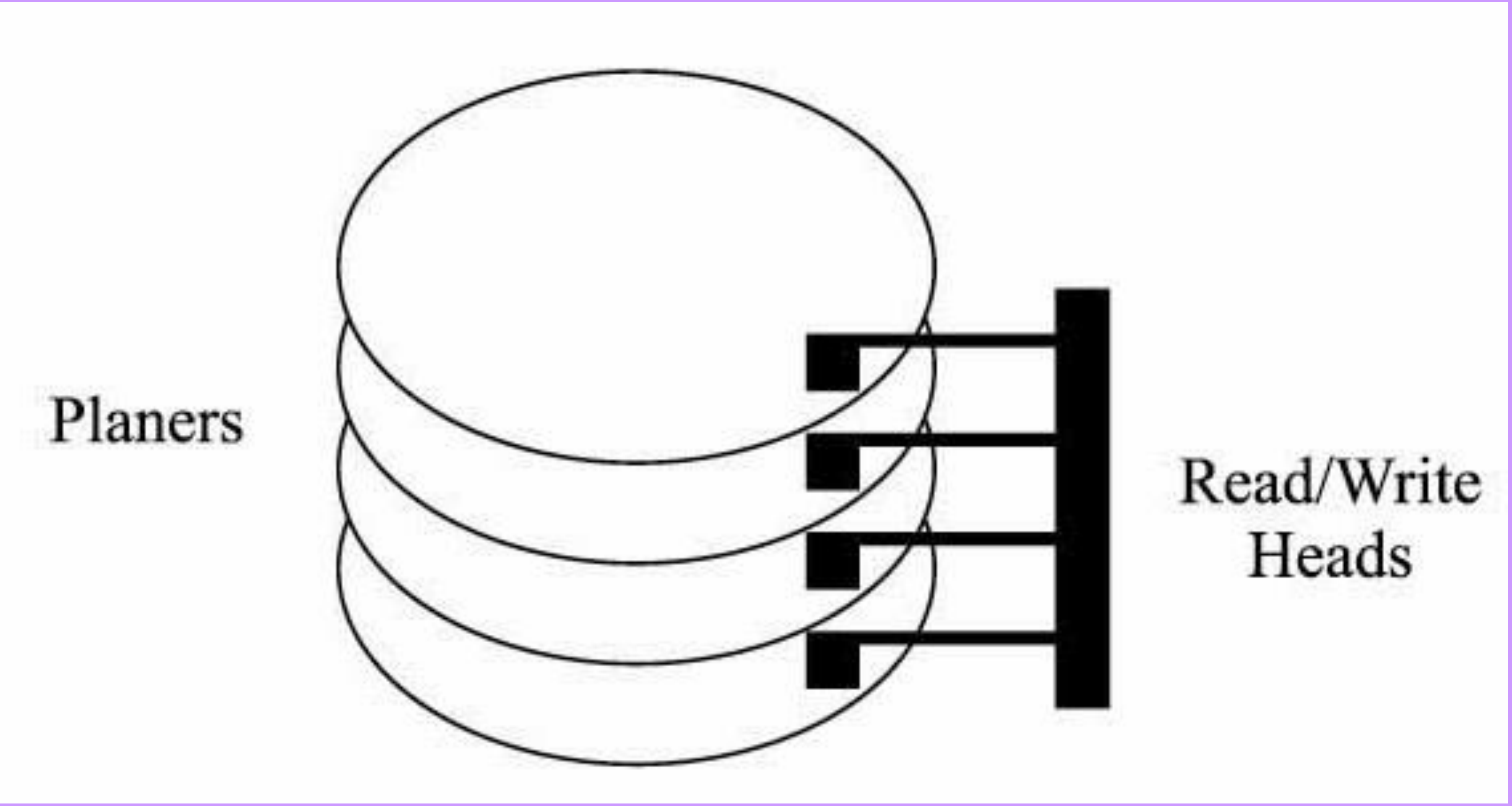
Two or more hard disks interconnect to the memory system bus through the hard disk controller and I/O bus



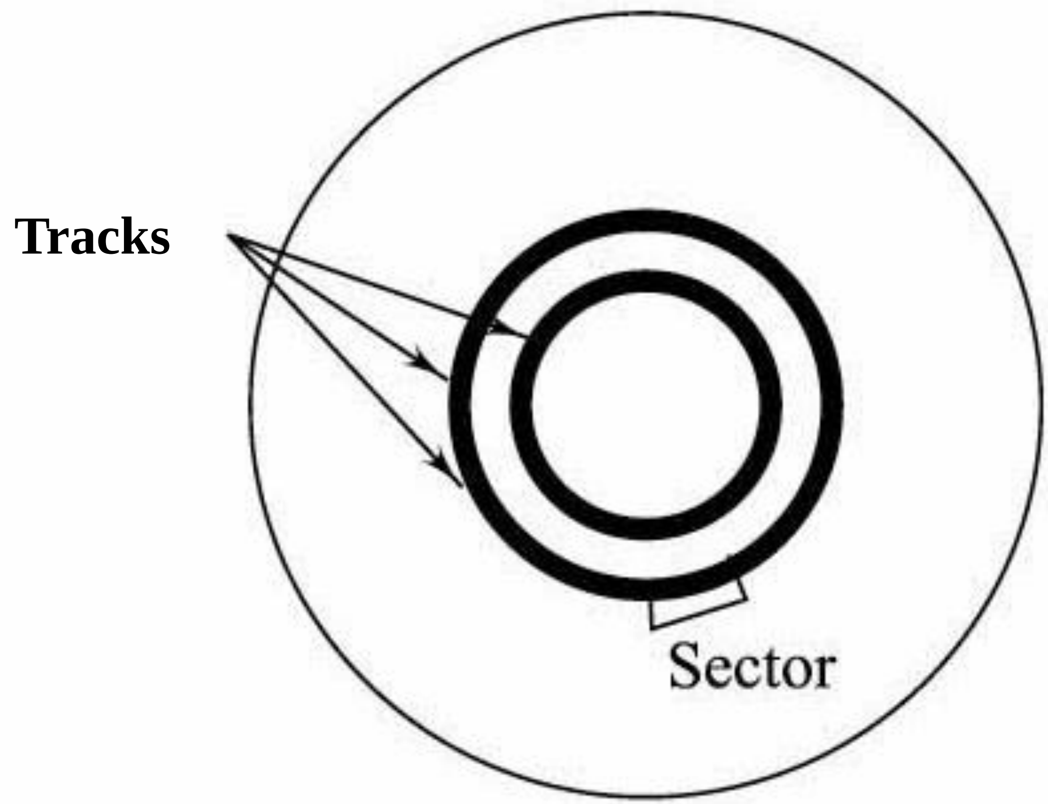
Organisation

- Each platter Its own read/write head
- Organisation allows accesses to data on different platters to proceed in parallel
- Within each platter, data is organized into tracks (concentric rings) and sectors (fractions of a ring)
- On a given hard disk, each sector contains the same amount of data, often 512 bytes

Hard disk organisation



Tracks and sectors



Disks that have a constant number of sectors per track

- The number of sectors per track is set based on the number of bits that the drive can fit on the innermost (smallest) track, which is a function of both the circumference of the innermost track and the technology used to construct the disk
- Other tracks contain the same number of bits, but they are written less densely, so that each sector takes up the same fraction of the track's circumference as on the innermost track

Recent disks dense storage of data

- By keeping the density at which bits are written closer to constant and
- By varying the number of sectors stored on each track so that tracks further out from the center of the disk contain more data

Request scheduling

The seek time

1. The seek time t_s — amount of time that it takes to move the read/write head from the track it is currently accessing to the track that contains the requested data

The rotation time

2. The rotational latency t_s — time that the head has to wait once it reaches the correct track before the requested sector arrives under the read/write head

The transfer time and total time to complete a read or write operation

3. The transfer time t_{tr} — time that it takes to read or write the sector once it reaches the read/write head, which is basically the time that it takes the sector to pass under the read/write head

- Total Time = $t_s + t_s + t_{tr}$

Example

- Find the average rotational latency time of a request
- Find the transfer time for a sector
- Assume— A disk spins at 10,000 r/min (revolution per minute)
- Assume— A given track on the disk has 1024 sectors

Solution

- At 10,000 r/mm, it takes 6 ms for a complete rotation of the disk
- On the average the read/write head will have to wait for half a rotation before the needed sector reaches it, so the average rotational latency will be 3 ms
- Since there are 1024 sectors on the track, the transfer time will be equal to the rotation time of the disk divided by 1024, or approximately 6 microseconds

Request scheduling

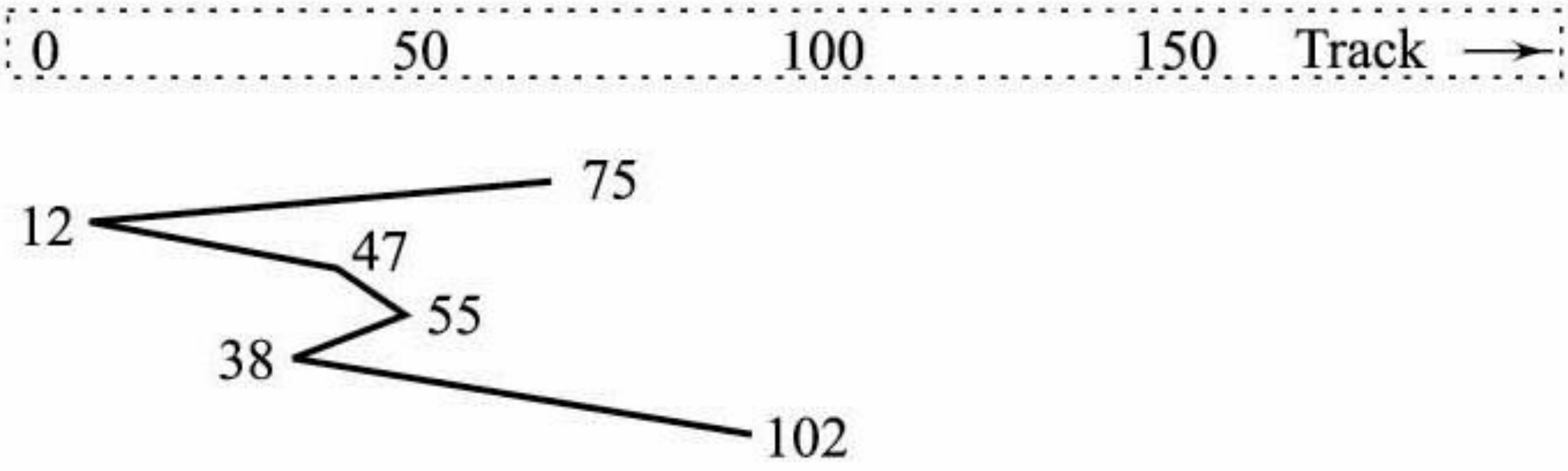
Scheduling of request for a read or write operation

- Three commonly used policies

1. First-come-first-serve (FCFS)
2. Shortest-seek-time-first (SSTF)
3. LOOK scheduling

First-come-first-serve Scheduling

Sequence of requests: 12, 47, 55, 38, and 102, Head starts at

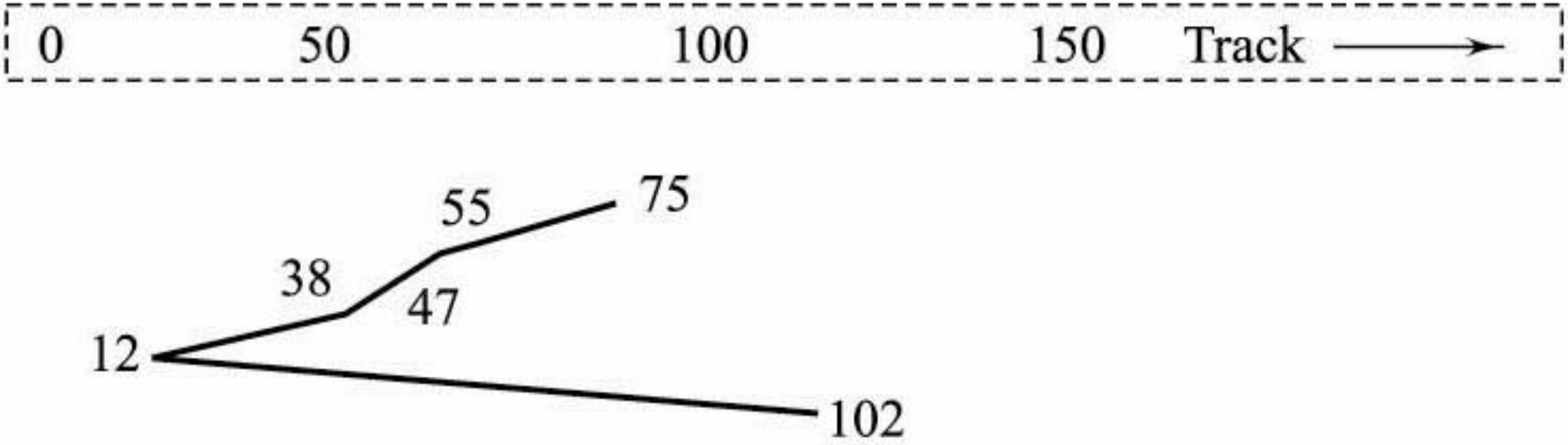


FCFS scheduling

- Advantage— that it is easy to implement
- Disadvantage— require much more motion of the disk head than other scheduling policies
- Increase in disk motion leads to longer average seek times and lower performance

Shortest-seek-time-first Scheduling

Sequence of requests: 12, 47, 55, 38, and 102, Head starts at



SSTF scheduling

- Advantage— Significantly reduce the average seek time of a disk
- Disadvantage— a sequence of requests to tracks near each other can prevent a request to a further-away track from ever being completed (starvation)

Example

- Consider a program that starts out by making requests to tracks 1, 2, and 100, and then makes a sequence of requests to tracks 1 and 2
- If the request to track 1 is handled first, the request to track 2 will be handled second, followed by the next request to track 1, and so on
- Until the sequence of requests to tracks 1 and 2 ends, the request to track 100 will never be satisfied

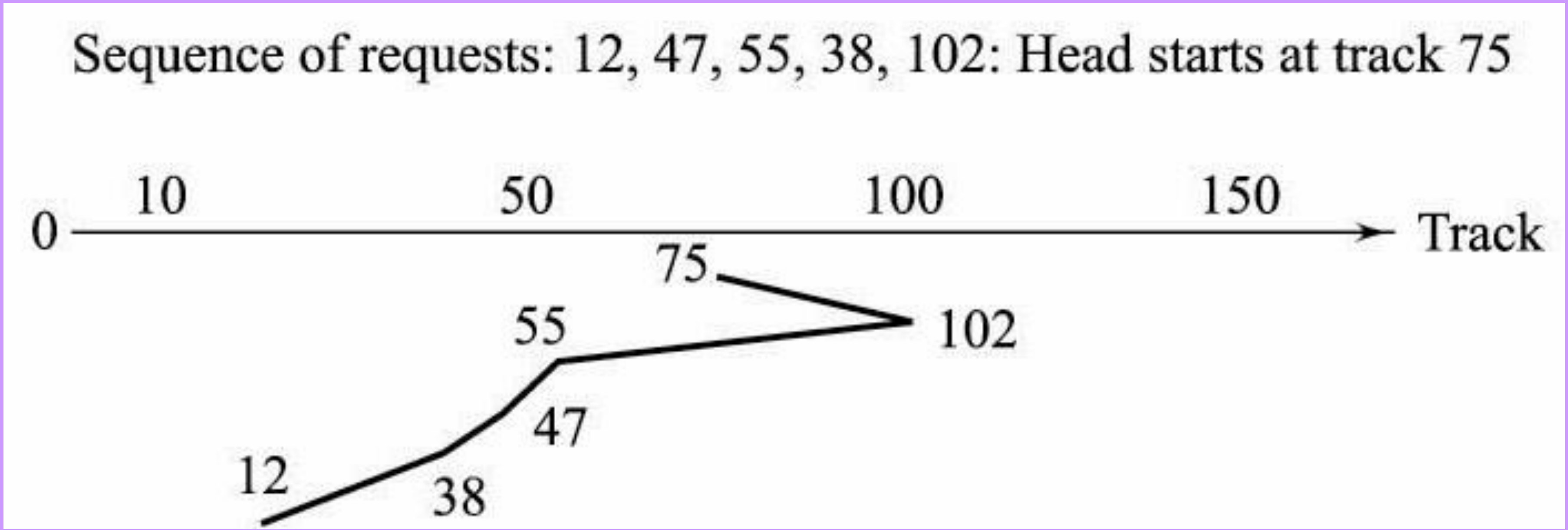
LOOK scheduling

- The head starts moving either inward or outward on the platter, satisfying all of the requests for tracks that it passes
- When it reaches the track of the innermost or outermost request, it reverses direction handling pending requests as it reaches their track

LOOK Scheduling

- Elevator modeled scheduling
- Compromise between FCFS and SSTF
- Generally delivers better performance than FCFS without the possibility of starvation

Look Scheduling



Summary

We learnt

- Several disks can share an IO bus
- Hard disk organised as multiple platters with each platter separate head and each platter number of tracks and sectors
- Three request scheduling mechanisms for read or write: First-come-first-serve (FCFS), Shortest-seek-time-first (SSTF) and LOOK scheduling

End of Lesson 17 on

Auxiliary memory Peripheral Devices – Secondary Storage Magnetic Ferrite Core Memories