JASHWANTH REDDY GANGULA

Advanced Computer Architecture

Assignment – 3

SUID : 646254141

B.1 [10/10/10/15] <B.1> You are trying to appreciate how important the principle of

locality is in justifying the use of a cache memory, so you experiment with a

computer having an L1 data cache and a main memory (you exclusively focus on

data accesses). The latencies (in CPU cycles) of the different kinds of accesses

are as follows: cache hit, 1 cycle; cache miss, 105 cycles; main memory access

with cache disabled, 100 cycles.

a. [10] <B.1> When you run a program with an overall miss rate of 5%, what

will the average memory access time (in CPU cycles) be?

**Solution:** Average memory access time = Access time for cache is hit + Miss rate \* Miss penalty

= ((1-MissRate) \* Hit time ) + (Miss Rate) \* Miss penalty

= (0.95 \* 1) + (0.05 \* 105)  
 = 6.2 cycles

b. [10] <B.1> Next, you run a program specifically designed to produce completely

random data addresses with no locality. Toward that end, you use an

array of size 256 MB (all of it fits in the main memory). Accesses to random

elements of this array are continuously made (using a uniform random number

generator to generate the elements indices). If your data cache size is 64 KB,

what will the average memory access time be?

**Solution:** Number of Times RAM is bigger than L1 Cache = 256MB/64KB ~ 4\*1000 = 4000

When a random element is accessed it could be from one of the 4000 blocks. The block might be present in L1 or not because of pseudo random index access. Hence   
Average memory access time = ((1-MissRate) \* Hit time ) + (Miss Rate) \* Miss penalty  
 = (1/4000)\*1 + ((3999/4000)\*105)

~ 104.974 cycles

c. [10] <B.1> If you compare the result obtained in part (b) with the main memory

access time when the cache is disabled, what can you conclude about the

role of the principle of locality in justifying the use of cache memory?

**Solution:**  From the above problem we observe that

average main memory access with cache disabled = 100 cycles

average memory access with L1 cache and pseudo random memory access = 104.9 cycles

If there is no locality references made in the above example, the access time with Cache is higher than direct RAM access time. Hence without locality property in the sequences of data access patterns cache will be useless and becomes unnecessary overhead for data access.

d. [15] <B.1> You observed that a cache hit produces a gain of 99 cycles (1 cycle

vs. 100), but it produces a loss of 5 cycles in the case of a miss (105 cycles vs.

100). In the general case, we can express these two quantities as G (gain) and

L (loss). Using these two quantities (G and L), identify the highest miss rate

after which the cache use would be disadvantageous.

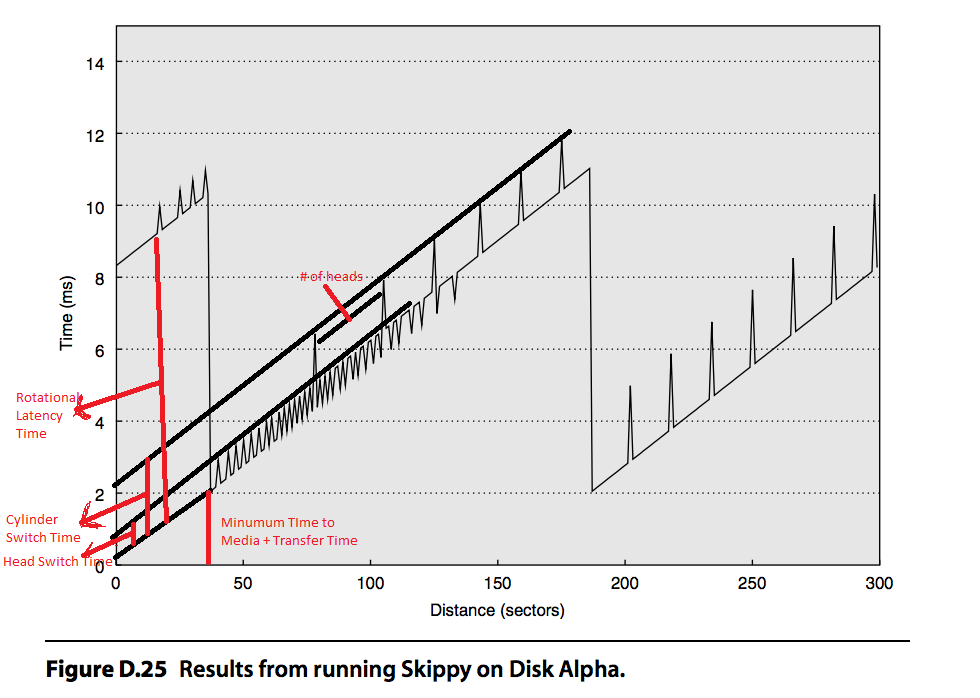
**Solution:** let miss rate be x(fraction) Average Memory access time = (1-x) + x\*105 = 1+104x

The above memory access time should be less than or equal to 100 cycles to improve the performance with L1 cache. Hence

1+104x <= 100 => x <= 99/104 = 0.95192

So when the miss rate exceeds 95.19% the cache would be disadvantageous.

**D.1 [10/10/10/10/10] The results of running Skippy are shown for a mock disk (Disk Alpha) in Figure D.25.**



a. [10] What is the minimal transfer time?

Minimum Transfer time is when there is no seek or rotational Latency. From the figure

Minimum transfer time = 2 ms

b. [10] What is the rotational latency?

Time for the required sector to rotate under the head is referred to as rotational latency. In the figure rotational Latency = 8.3 ms

c. [10] What is the head switch time?

Head Switch time is the process of switching the data channel from one surface to the next in the same cylinder. In this figure head switch time ~ 0.8 ms

d. [10] What is the cylinder switch time?

 Cylinder switch time is the time that elapses when the drive finishes reading (or writing) all the data on a given cylinder and needs to switch to the next one. In this figure cylinder switch time ~ 21.ms

e. [10] What is the number of disk heads?

15 disk heads

**D.2 [25] Draw an approximation of the graph that would result from running Skippy on Disk Beta, a disk with the following parameters:**

■ Minimal transfer time, 2.0 ms

■ Rotational latency, 6.0 ms

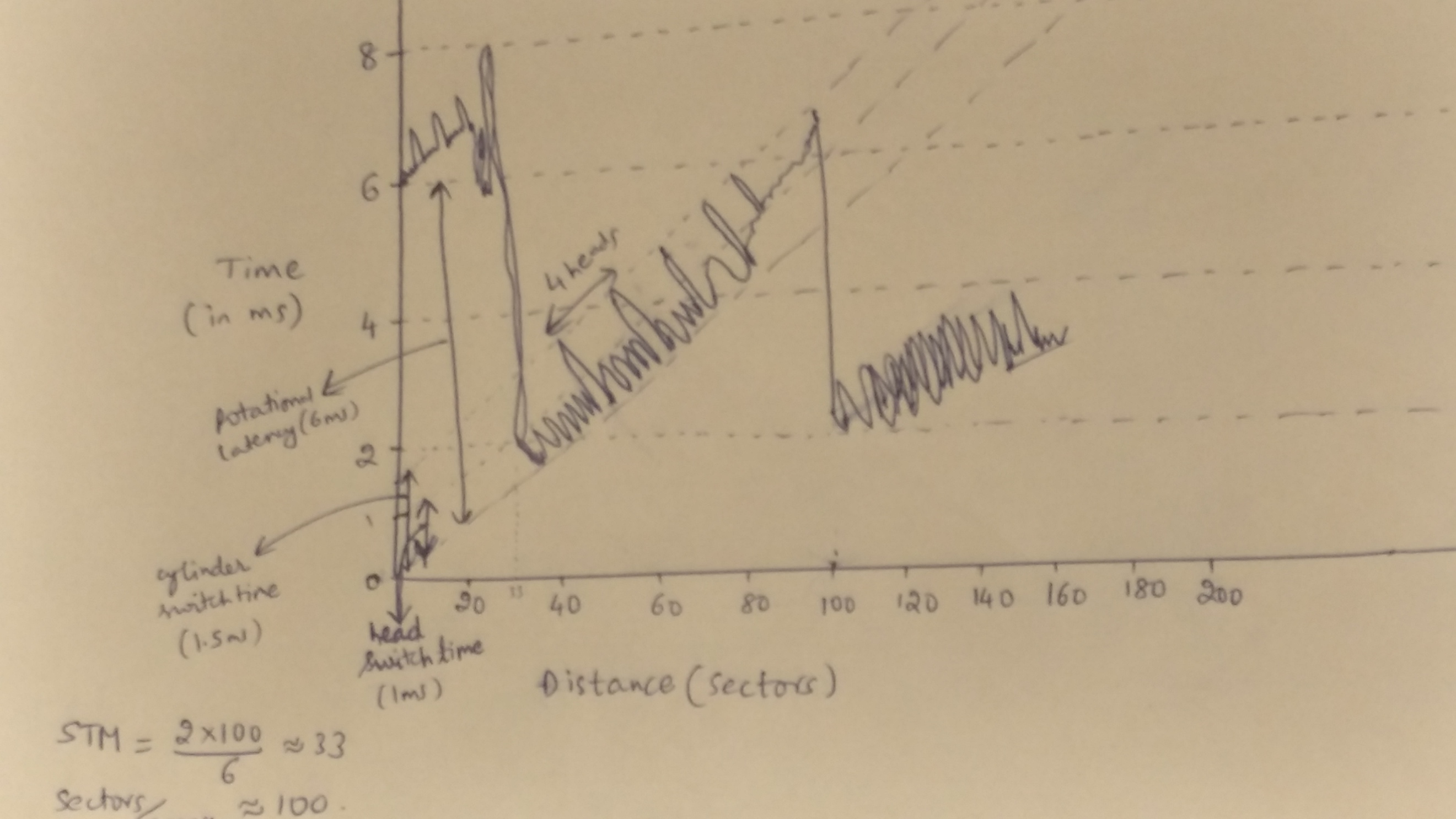
■ Head switch time, 1.0 ms

■ Cylinder switch time, 1.5 ms

■ Number of disk heads, 4

■ Sectors per track, 100

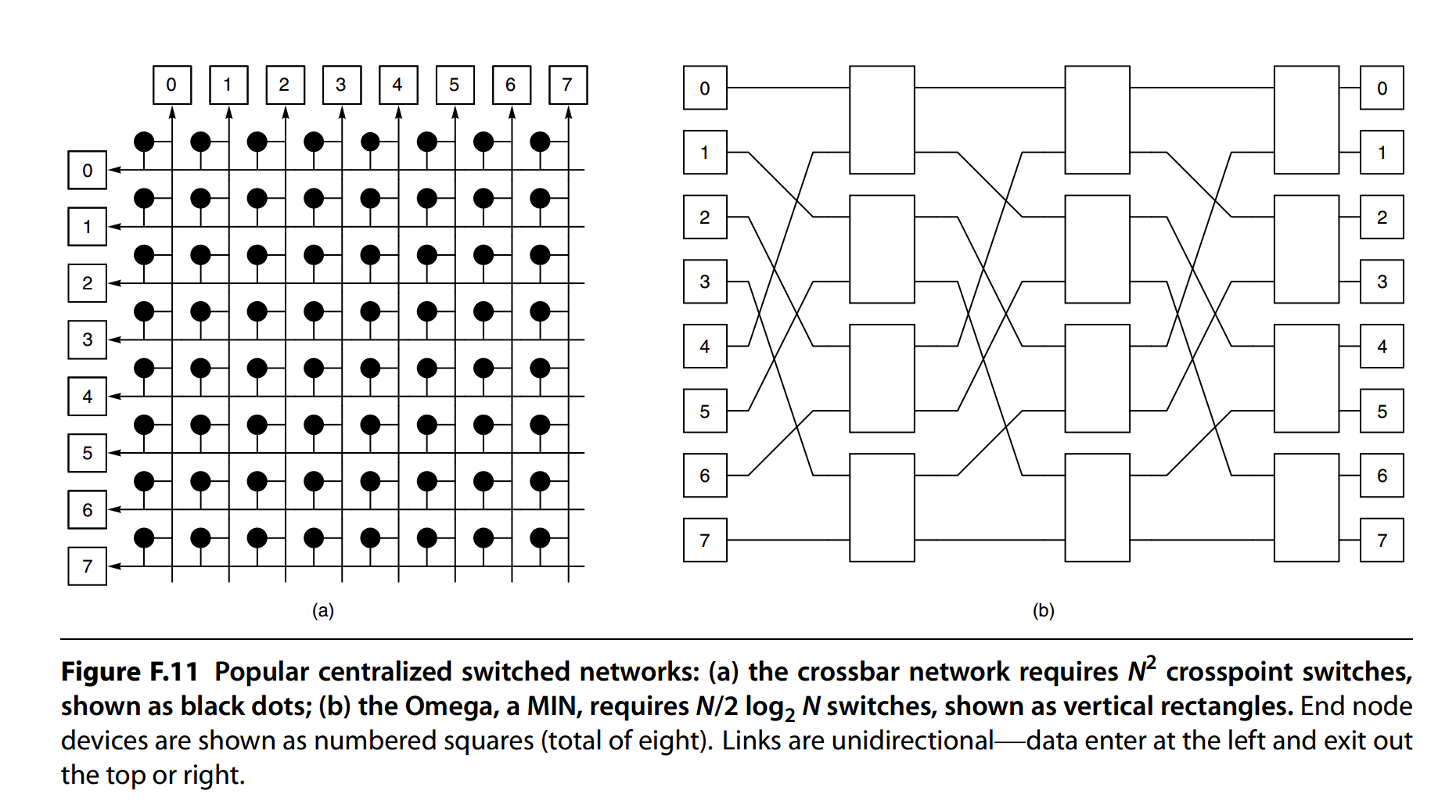
As per the above definitions for the previous problem the rough graph for the following parameters looks like this:



**D.19 [10] You will begin by designing an I/O subsystem that is optimized only for capacity and performance (and not reliability), specifically IOPS. Discuss the RAID level and block size that will deliver the best performance.**

**RAID 0** splits data evenly across two or more disks, without parity information, redundancy, or fault tolerance. RAID 0 can be used for applications that require high performance and can accept lower tolerance rate. Block is the minimal unit of data that the file system reads or writes from the storage system; The block size as per the given building block information will be 16Kb because the workload will require concurrent, random IOs with an average size of 16kb. Block size less than this will result wastage of CPU cycles

**F.15 [15] Compare the interconnection latency of a crossbar, Omega network, and fat tree with eight nodes. Use Figure F.11 on page F-31, Figure F.12 on page F-33, and Figure F.14 on page F-37. Assume that the fat tree is built entirely from two-input, two-output switches so that its hardware resources are more comparable to that of the Omega network. Assume that each switch costs a unit time delay. Assume that the fat tree randomly picks a path, so give the best case and worst case for each example. How long will it take to send a message from node 0 to node 6? How long will it take node 1 and node 7 to communicate?**



**Solution:**

**I ) cross bar network Figure F.11-a:**

Every node is connected to every other node in the network.

Time taken for node 1 to communicate with node 7 is 1-unit time.

**Time taken for path 0 to 6:**

Best case: 1-unit time + constant time.

**Time taken for path 1 to 7:**

This depends on whether 1 and 7 nodes communicate as unidirectional or bidirectional mode.

When Unidirectional communication: 1-unit time + constant time

Bidirectional communication: 2-unit time + constant time

II) **Omega Network Figure F.11-b:**

**Time taken for path 0 to 6:**

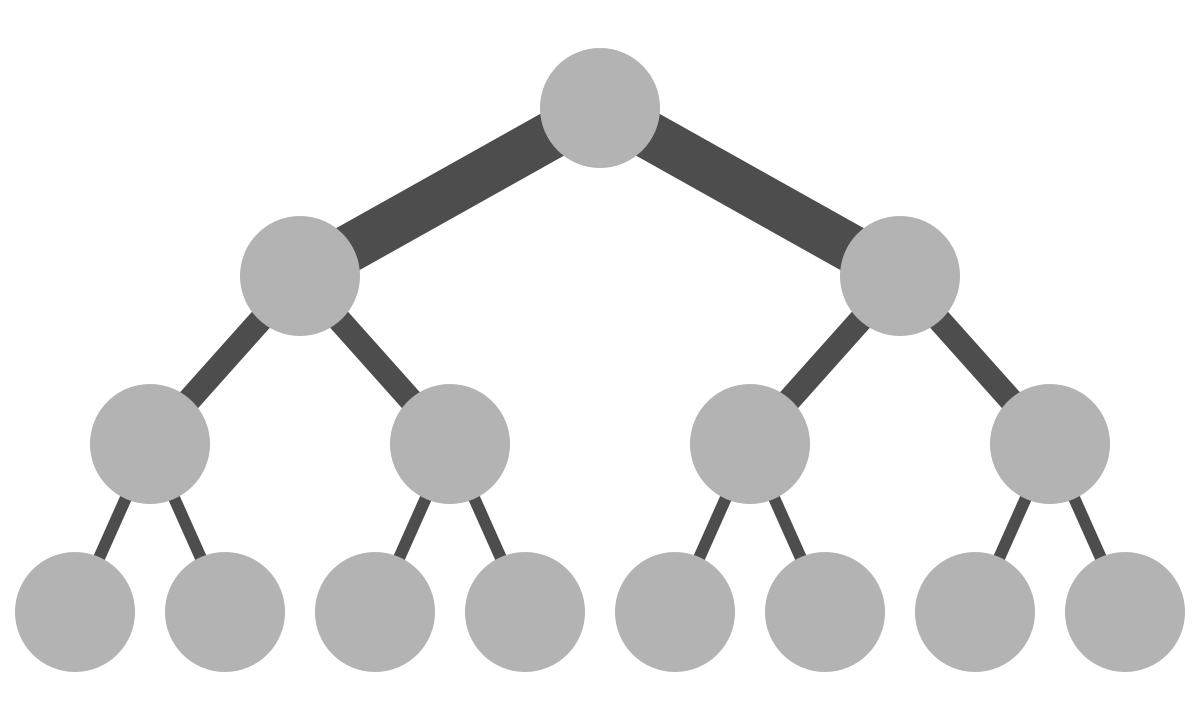
Best case: 3-unit time + constant time.

**Time taken for path 1 to 7:**

Again consider the case of unidirectional and bidirectional:

Time taken : 6-unit time + constant time

**III) Fat Tree Network:**

****

Fat tree network is a network of bidirectional multistage interconnection network with non-blocking properties.

**Time taken for path 0 to 6:**

Best case: Node 0 and 6 will be in the same switch. Time taken = 1-unit time + constant time

Worst case: Node 0 and 6 are in different places. Time taken = 5-unit time + constant time

**Time taken for path 1 to 7:**

Best case: Node 1 to Node 7 had unidirectional communication,

Time taken = 2-unit time + constant time

Worst case: Time Taken =10 unit time + constant time