Structure of a Switch

<u>Objective of Lesson 7:</u>

On Completion of this lesson, the students will be able to:

Define the types of switch structure.

Explain the types of Space-Division Switch.

Discuss Structure of Packet Switches.

Describe the components of Packet Switches.

Understand how the Banyan Switch is working.

There are switches in circuit-switched and packet-switched networks.

1. Structure of Circuit Switches

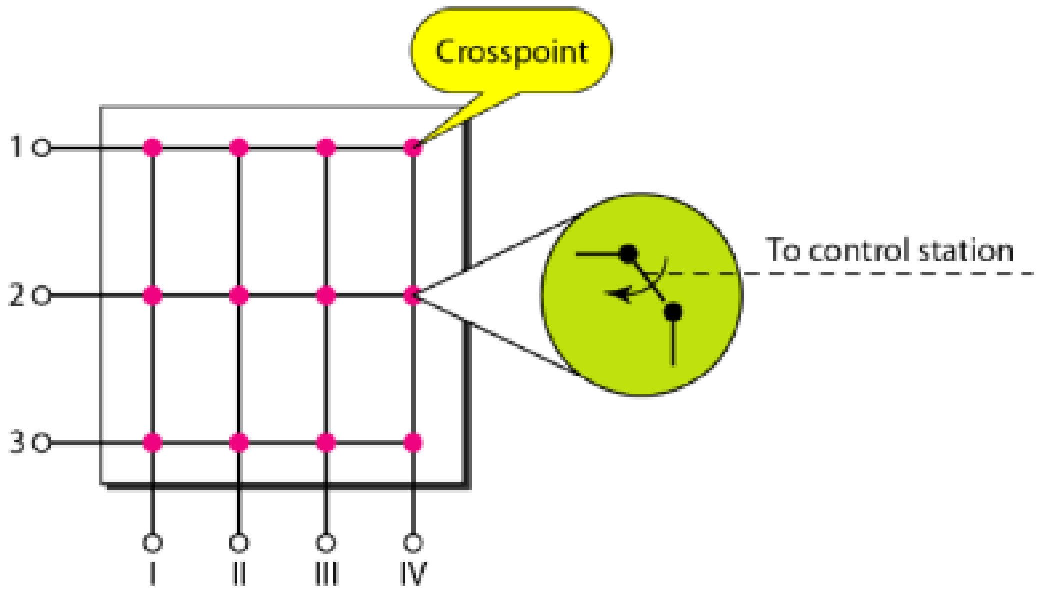
Circuit switching today can use either of two technologies: the space-division switch or the time-division switch.

Space-Division Switch:

In space-division switching, the paths in the circuit are separated from one another spatially. This technology is used in both analog and digital networks.

2.1. Crossbar Switch:

A crossbar switch connects n inputs to m outputs in a grid, using electronic micro-switches (transistors) at each crosspoint as shown in the following figure:

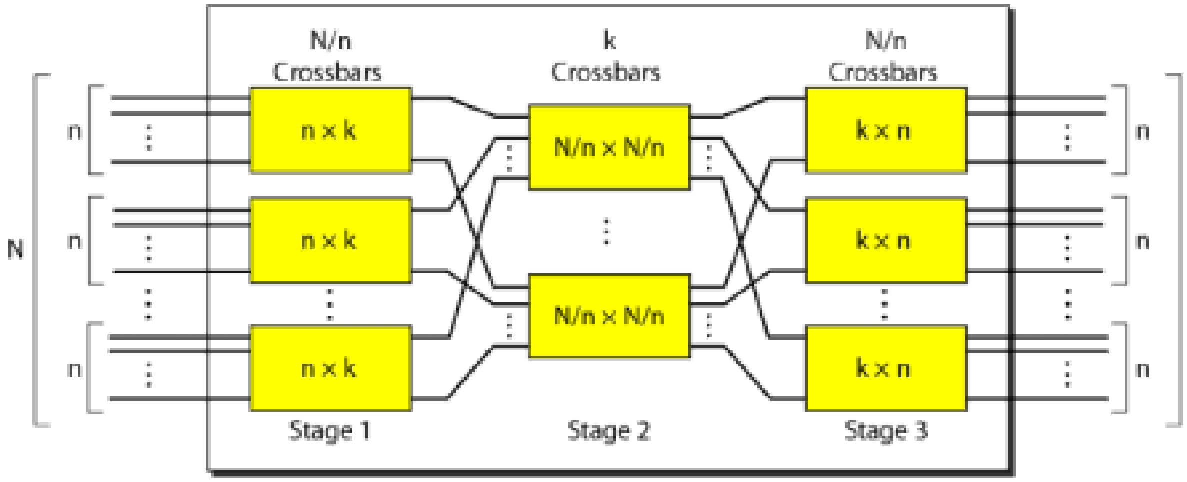


The major limitation of this design is the number of crosspoints required.

- To connect **n** inputs to **m** outputs using a crossbar switch requires **n x m** crosspoints (The resulting number of crosspoints is impractical).
- Such a switch is also inefficient because statistics show that, in practice, fewer than 25 percent of the crosspoints are in use at any given time. The rest are idle.

2.2. Multistage Switch

It is the solution to the limitations of the crossbar switch. It combines crossbar switches in several (normally three) stages, as shown in the following figure:



In a single crossbar switch, only one row or column (one path) is active for any connection. So we need N x N crosspoints. If we can allow multiple paths inside the switch, we can decrease the number of crosspoints. Each crosspoint in the middle stage can be accessed by multiple crosspoints in the first or third stage. To design a three-stage switch, we follow these steps:

We divide the **N** input lines into groups, each of **n** lines. For each group, we use one crossbar of size **n x k**, where **k** is the number of crossbars in the middle stage. In other words, the first stage has **N/n** crossbars of **n x k** crosspoints.

We use **k** crossbars, each of size **(N/n) x (N/n)** in the middle stage.

We use **N/n** crossbars, each of size **k x n** at the third stage.

We can calculate the total number of crosspoints as follows:

$$\frac{N}{n}(n \times k) + k\left(\frac{N}{n} \times \frac{N}{n}\right) + \frac{N}{n}(k \times n) = 2kN + k\left(\frac{N}{n}\right)^2$$

For example: Design a three-stage, 200 x 200 switch with k=4 and n = 20.

The multistage switch in the previous example has one drawback--blocking during periods of heavy traffic. The whole idea of multistage switching is to share the crosspoints in the middle-stage crossbars. Sharing can cause a lack of availability if

the resources are limited and all users want a connection at the same time. Blocking refers to times when one input cannot be connected to an output because there is no path available between them--all the possible intermediate switches are occupied. The small number of crossbars at the middle stage creates blocking. If the number of stages is increased to cut down on the number of crosspoints required, however as the number of stages increases, the possible blocking increases as well. Clos investigated the condition of nonblocking in multistage switches and came up with the following formula. In a nonblocking switch, the number of middle-stage switches must be at least 2n – 1(it means k≥2n-1).

It is found that to minimize the number of crosspoints with a fixed N by using the Clos criteria the n must be equal to or greater than $(N/2)^{1/2}$.

In this case, the total number of crosspoints is greater than or equal to

$$4N [(2N)^{1/2} - 1]$$

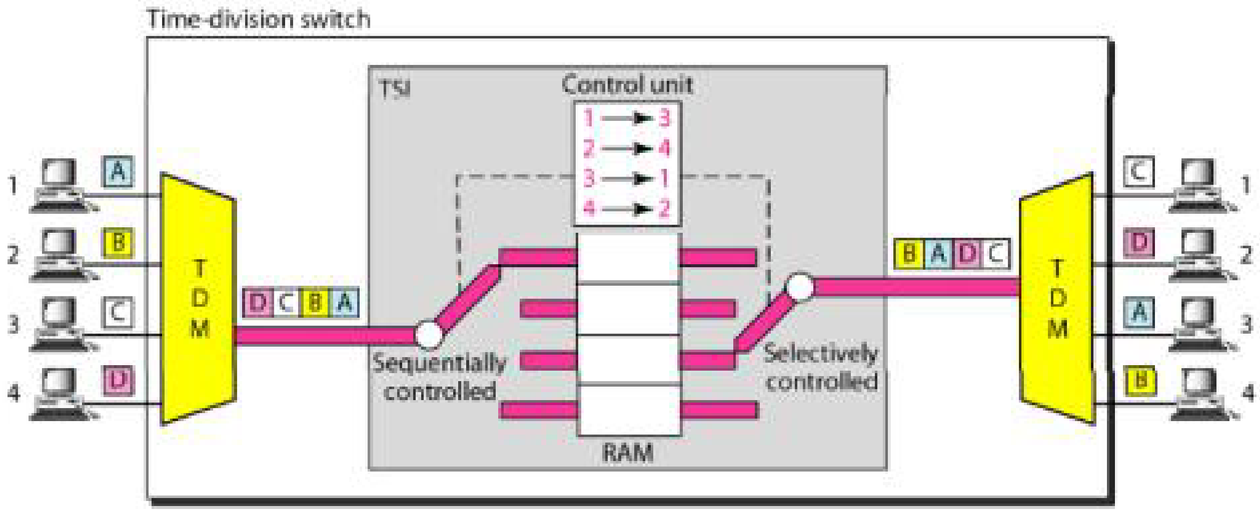
Design a three-stage, 200 x 200 switch (N = 200) with k -- 4 and n = 20. Use the Clos criteria.

We let n = $(200/2)^{1/2}$, or n = 10. We calculate k = 2n – 1=19. In the first stage, we have 200/10, or 20, crossbars, each with 10 x 19 crosspoints. In the second stage, we have 19 crossbars, each with 10 x 10 crosspoints. In the third stage, we have 20 crossbars each with 19 x 10 crosspoints. The total number of crosspoints is 20(10 x 19) + 19(10 x 10) + 20(19 x10) =9500.

A multistage switch that uses the Clos criteria and a minimum number of crosspoints still requires a huge number of crosspoints. The number can be reduced if we accept blocking. Today, telephone companies use time-division switching or a combination of space- and time-division switches.

3. Time-Division Switch

Time-division switching uses time-division multiplexing (TDM) inside a switch. The most popular technology is called the time-slot interchange (TSI). See the following figure:



It consists of:

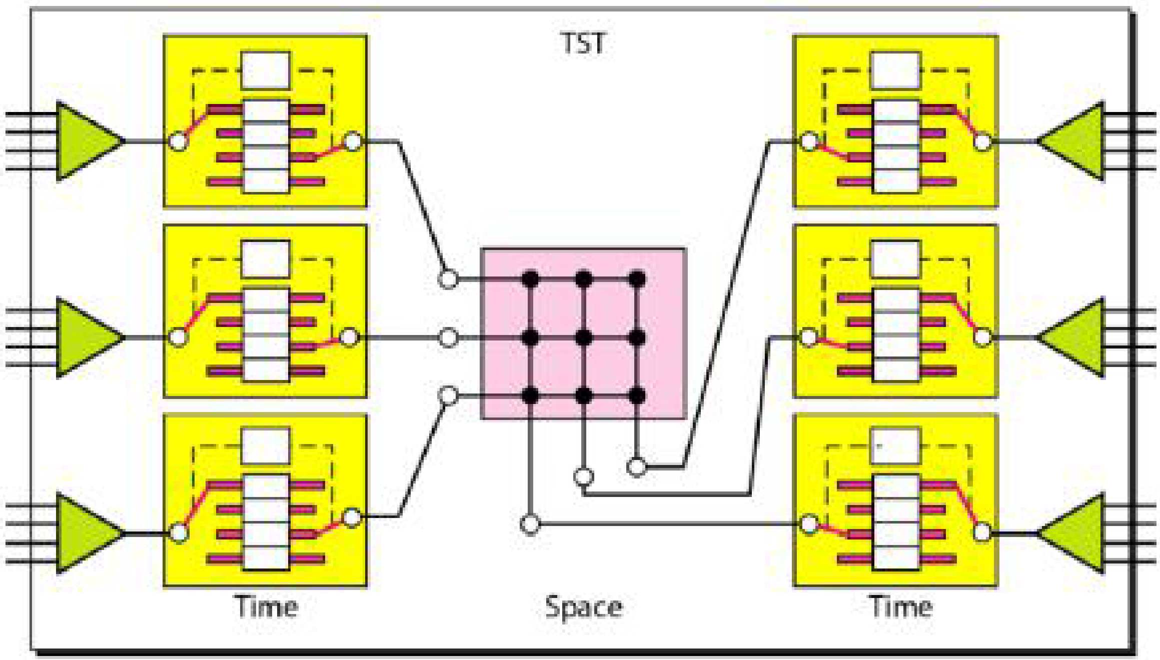
- TDM multiplexer.
- TDM demultiplexer.
- Random access memory (RAM) with several memory locations.
- Control unit.

The RAM fills up with incoming data from time slots in the order received. Slots are then sent out in an order based on the decisions of a control unit.

3.1. Time- and Space-Division Switch Combinations:

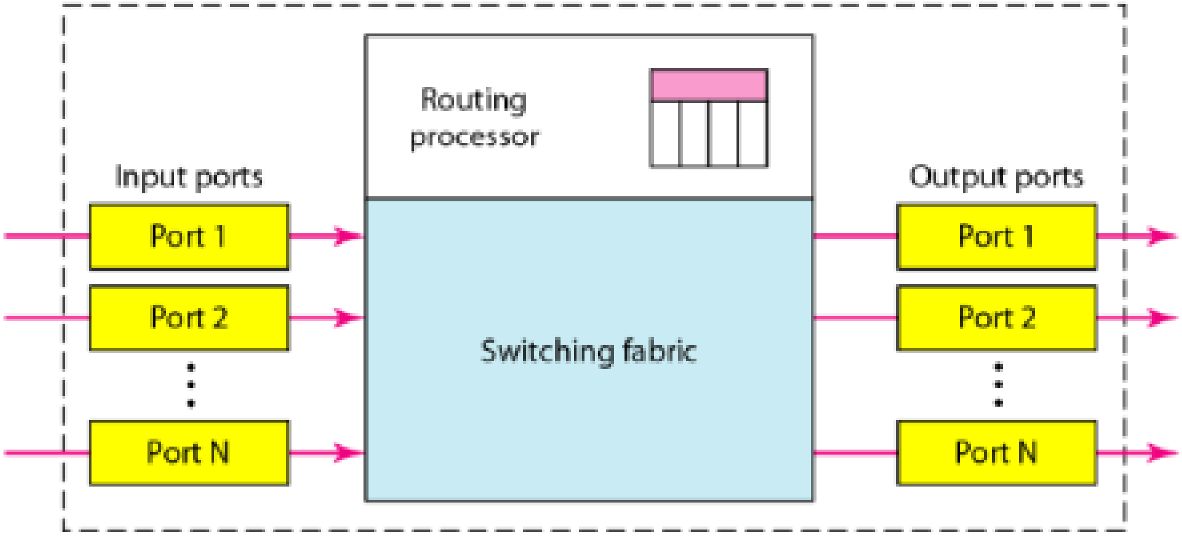
When we compare space-division and time-division switching, some interesting facts emerge. The advantage of space-division switching is that it is instantaneous. Its disadvantage is the number of crosspoints required to make space-division switching acceptable in terms of blocking. The advantage of time-division switching is that it needs no crosspoints. Its disadvantage, in the case of TSI, is that processing each connection creates delays. Each time slot must be stored by the RAM, then retrieved and passed on.

In a third option, we combine space-division and time-division technologies to take advantage of the best of both. Combining the two results in switches that are optimized both physically (the number of crosspoints) and temporally (the amount of delay). Multistage switches of this sort can be designed as time-space-time (TST) switch.



4. Structure of Packet Switches:

A switch used in a packet-switched network has a different structure from a switch used in a circuit-switched network. We can say that a packet switch has four components: input ports, output ports, the routing processor, and the switching fabric, as shown in the figure:



4.1. Input Ports

An input port performs the physical and data link functions of the packet switch.

The bits are constructed from the received signal.

The packet is decapsulated from the frame.

Errors are detected and corrected.

The input port has buffers (queues) to hold the packet before it is directed to the switching fabric. The following figure shows a schematic diagram of an input port.

4.2. Routing Processor

The routing processor performs the functions of the network layer.

Find the address of the next hop.
Find the output port number from which the packet is sent out.

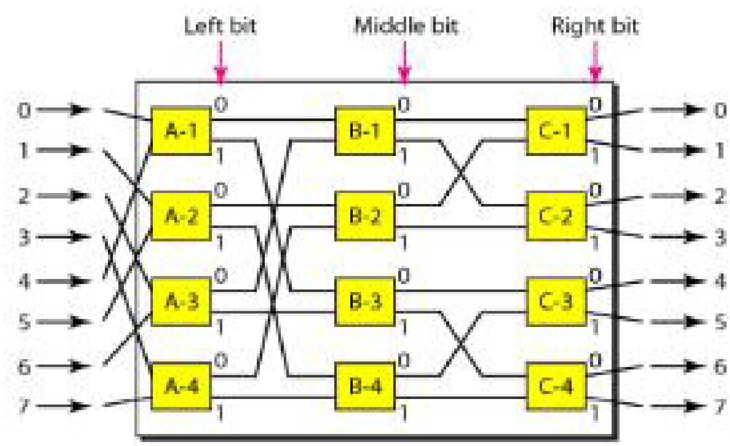
4.3. Switching Fabrics

The most difficult task in a packet switch is to move the packet from the input queue to the output queue. The packet switches are specialized mechanisms that use a variety of switching fabrics. Some of these fabrics are:

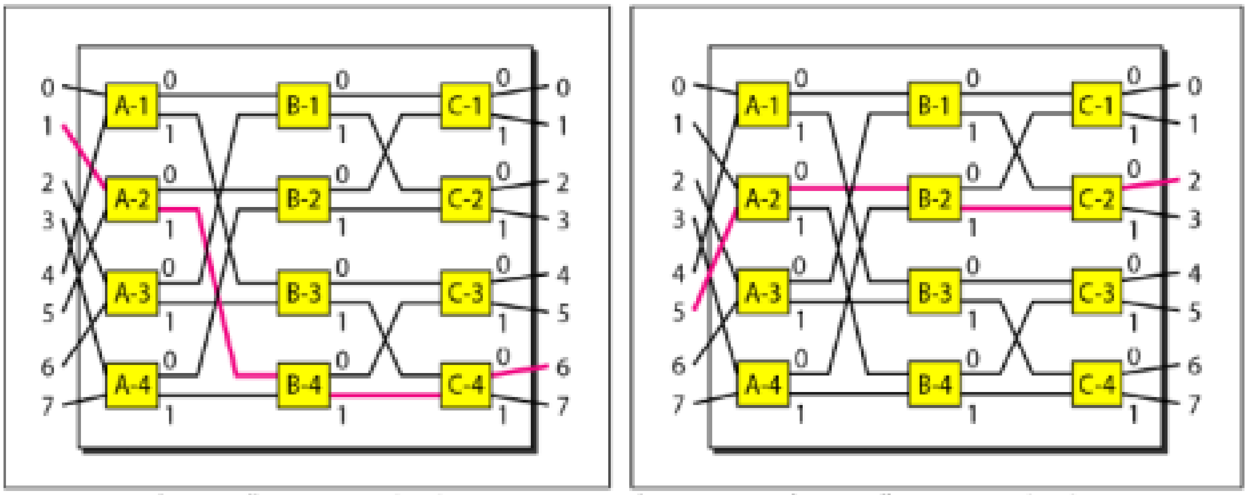
Crossbar Switch: the simplest type of switching fabric is the crossbar switch, discussed in the previous section.

Banyan Switch: a more realistic approach than the crossbar switch is the banyan switch (named after the banyan tree). A banyan switch is a multistage switch

with microswitches at each stage that route the packets based on the output port represented as a binary string. For **n** inputs and **n** outputs, we have **log2n** stages with **n/2** microswitches at each stage. The first stage routes the packet based on the high-order bit of the binary string. The second stage routes the packet based on the second high-order bit, and so on. The figure shows a banyan switch with eight inputs and eight outputs. The number of stages is log2(8) = 3.

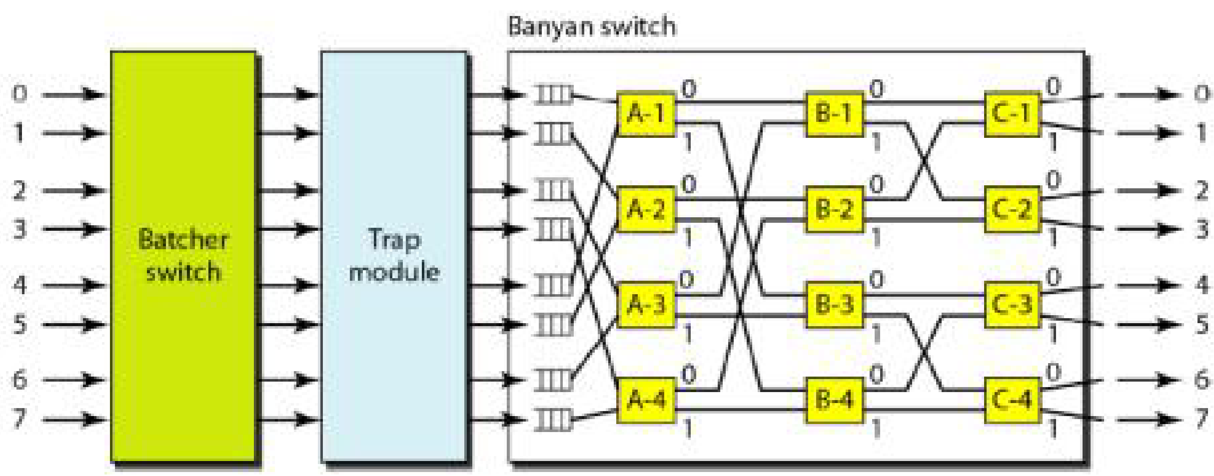


The following two examples for banyan switch:



a. Input 1 sending a cell to output 6 (110) b. Input 5 sending a cell to output 2 (010)

One of the drawbacks of banyan switch is the possibility of internal collision. This problem can be solved by sorting the arriving packets based on their destination port. For that purpose another hardware module called a trap is added between the Batcher switch and the banyan switch, see the following figure:



The trap module prevents duplicate packets (packets with the same output destination) from passing to the banyan switch simultaneously. Only one packet for each destination is allowed at each tick; if there is more than one, they wait for the next tick.