Switching

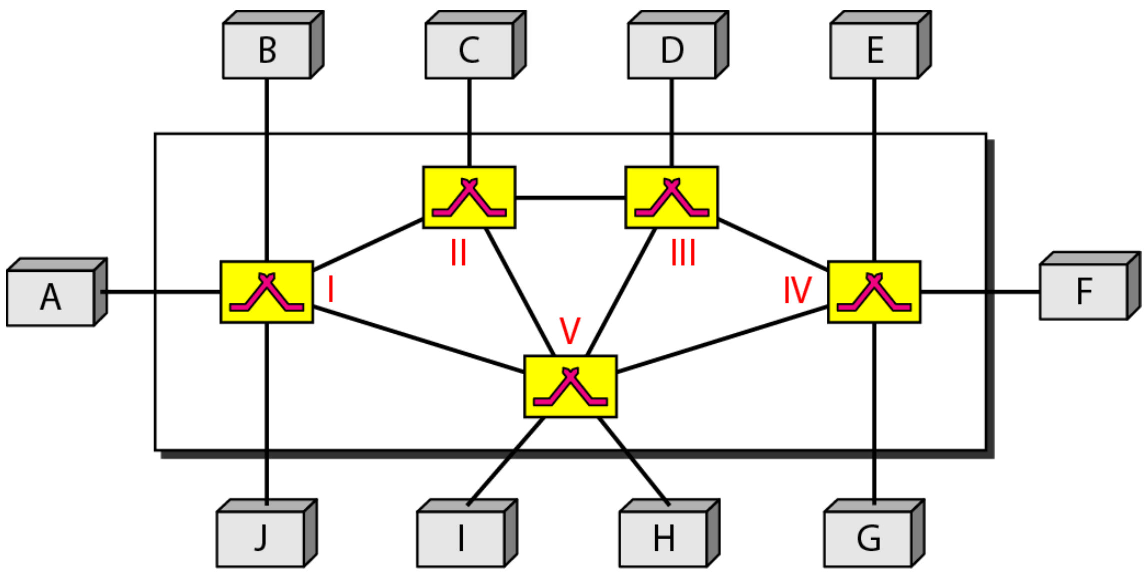
8 Connecting devices

A network is a set of connected devices. How to connect them to make one-to-one communication possible?
Point-to-point connection (mesh, star)? Multipoint connection (Bus)?

A better solution is Switching

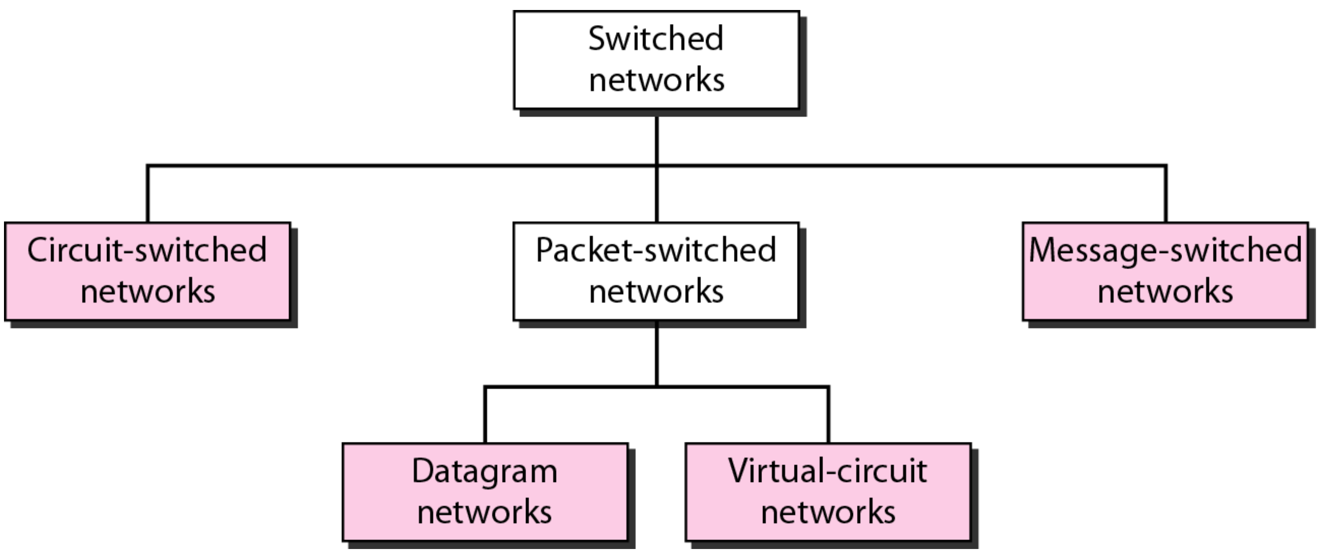
Switching

A switched network consists of a series of interlinked nodes, called switches. Switches are devices capable of creating temporary connections between two or more devices linked to the switch. In a switched network, some of these nodes are connected to the end systems (computers or telephones, for example). Others are used only for routing. The end systems (communicating devices) are labeled A, B, C, D, and so on, and the switches are labeled I, II, III, IV, and V. Each switch is connected to multiple links.



Type of Switched Networks

Traditionally, three methods of switching have been important: circuit switching, packet switching, and message switching. The first two are commonly used today. The third has been phased out in general communications but still has networking applications. In message switching, each switch stores the whole message and forwards it to the next switch. Although, we don't see message switching at lower layers, it is still used in some applications like electronic mail (e-mail).



8-1 CIRCUIT-SWITCHED NETWORKS

A circuit-switched network consists of a set of switches connected by physical links. A connection between two stations is a dedicated path made of one or more links. However, each connection uses only one dedicated channel on each link. Each link is normally divided into n channels by using FDM or TDM.

Topics discussed in this section:

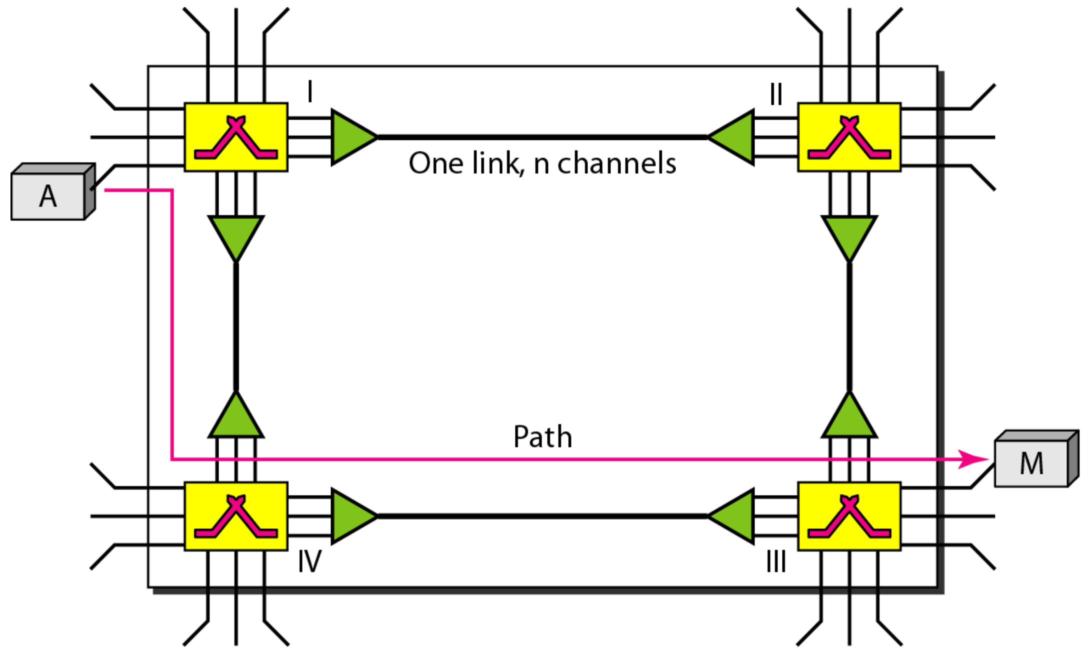
Three Phases
Efficiency
Delay
Circuit-Switched Technology in Telephone Networks

Note

A circuit-switched network is made of a set of switches connected by physical links, in which each link is divided into *n* channels.

Figure 8.3 *A trivial circuit-switched network*

The end systems, such as computers or telephones, are directly connected to a switch. We have shown only two end systems for simplicity. When end system A needs to communicate with end system M, system A needs to request a connection to M that must be accepted by all switches as well as by M itself. This is called the setup phase; a circuit (channel) is reserved on each link, and the combination of circuits or channels defines the dedicated path. After the dedicated path made of connected circuits (channels) is established, data transfer can take place. After all data have been transferred, the circuits are torn down (teardown phase).



Few points about Circuit Switching

•Circuit switching takes place at the physical layer.

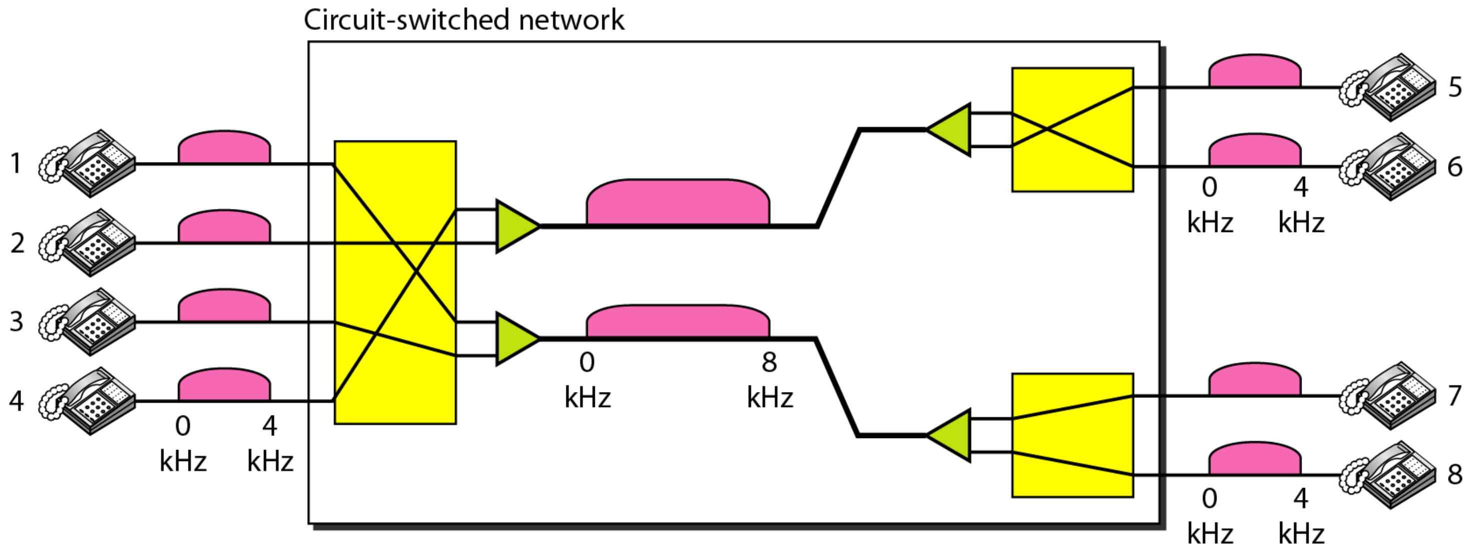
•Before starting communication, the stations must make a reservation for the resources to be used during the communication. These resources, such as channels (bandwidth in FDM and time slots in TDM), switch buffers, switch processing time, and switch input/output ports, must remain dedicated during the entire duration of data transfer until the teardown phase.

•Data transferred between the two stations are not packetized (physical layer transfer of the signal). The data are a continuous flow sent by the source station and received by the destination station, although there may be periods of silence.

•There is no addressing involved during data transfer. The switches route the data based on their occupied band (FDM) or time slot (TDM). Of course, there is end-to-end addressing used during the setup phase, as we will see shortly.

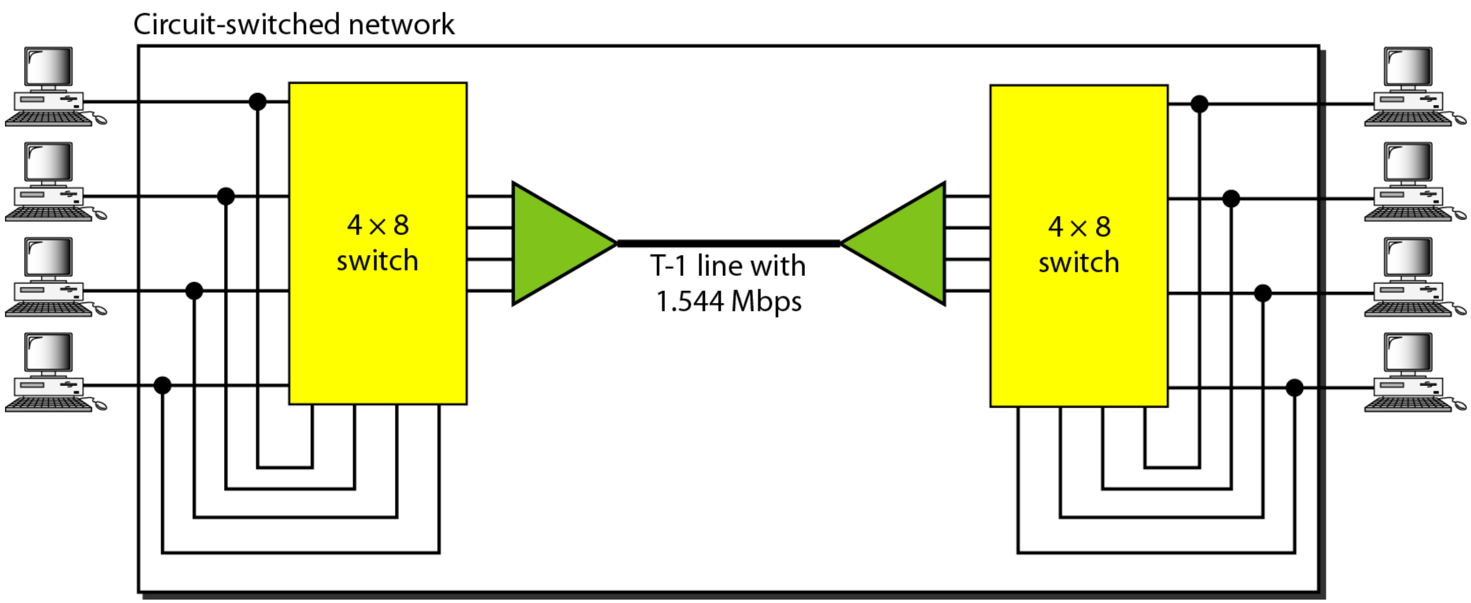
Example 8.1

As a trivial example, let us use a circuit-switched network to connect eight telephones in a small area. Communication is through 4-kHz voice channels. We assume that each link uses FDM to connect a maximum of two voice channels. The bandwidth of each link is then 8 kHz. Figure 8.4 shows the situation. Telephone 1 is connected to telephone 7; 2 to 5; 3 to 8; and 4 to 6. Of course the situation may change when new connections are made. The switch controls the connections.



Example 8.2

As another example, consider a circuit-switched network that connects computers in two remote offices of a private company. The offices are connected using a T-1 line leased from a communication service provider. There are two 4 × 8 (4 inputs and 8 outputs) switches in this network. For each switch, four output ports are folded into the input ports to allow communication between computers in the same office. Four other output ports allow communication between the two offices. Figure 8.5 shows the situation.



Three Phases

The actual communication in a circuit-switched network requires three phases:

*•**Set-up Phase:** Before the two parties can communicate, a dedicated circuit (combination of channels in links) needs to be established. The end systems are normally connected through dedicated lines to the switches, so connection setup means creating dedicated channels between the switches. An end-to-end addressing is required for creating a connection between the two end systems.*

__Data Transfer Phase:__ After the establishment of the dedicated circuit (channels), the two parties can transfer data.

__Teardown Phase:__ When one of the parties needs to disconnect, a signal is sent to each switch to release the resources.

Efficiency

It can be argued that circuit-switched networks are not efficient because resources are allocated during the entire duration of the connection.

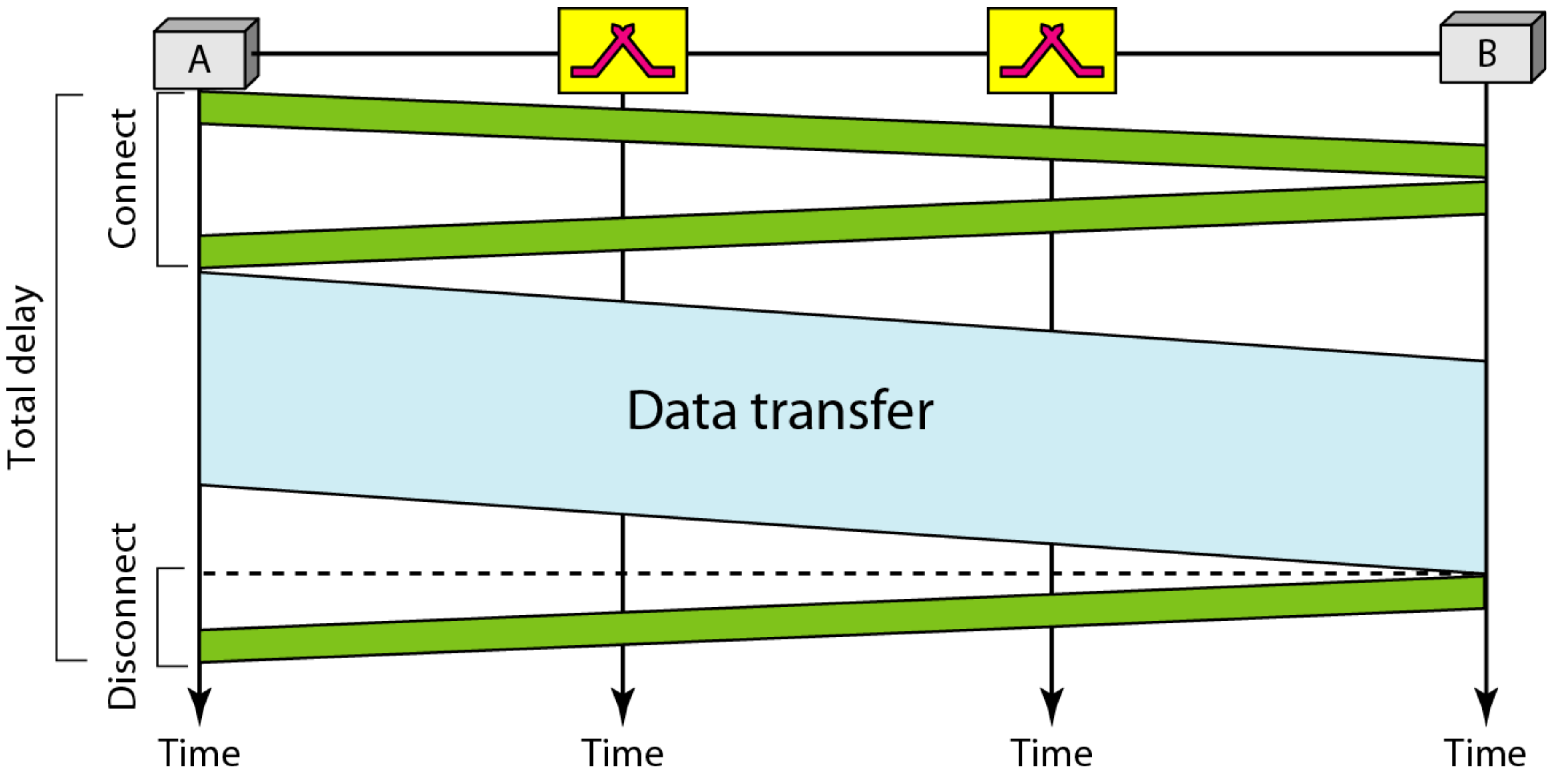
These resources are unavailable to other connections. In a telephone network, people normally terminate the communication when they have finished their conversation.

However, in computer networks, a computer can be connected to another computer even if there is no activity for a long time. In this case, allowing resources to be dedicated means that other connections are deprived.

Delay

Although a circuit-switched network normally has low efficiency, the delay in this type of network is minimal. During data transfer the data are not delayed at each switch; the resources are allocated for the duration of the connection. Figure 8.6 shows the idea of delay in a circuit-switched network when only two switches are involved. As Figure 8.6 shows, there is no waiting time at each switch. The total delay is due to the time needed to create the connection, transfer data, and disconnect the circuit. The delay caused by the setup is the sum of four parts: the propagation time of the source computer request (slope of the first gray box), the request signal transfer time (height of the first gray box), the propagation time of the acknowledgment from the destination computer (slope of the second gray box), and the signal transfer time of the acknowledgment (height of the second gray box). The delay due to data transfer is the sum of two parts: the propagation time (slope of the colored box) and data transfer time (height of the colored box), which can be very long. The third box shows the time needed to tear down the circuit. We have shown the case in which the receiver requests disconnection, which creates the maximum delay.

Figure 8.6 *Delay in a circuit-switched network*



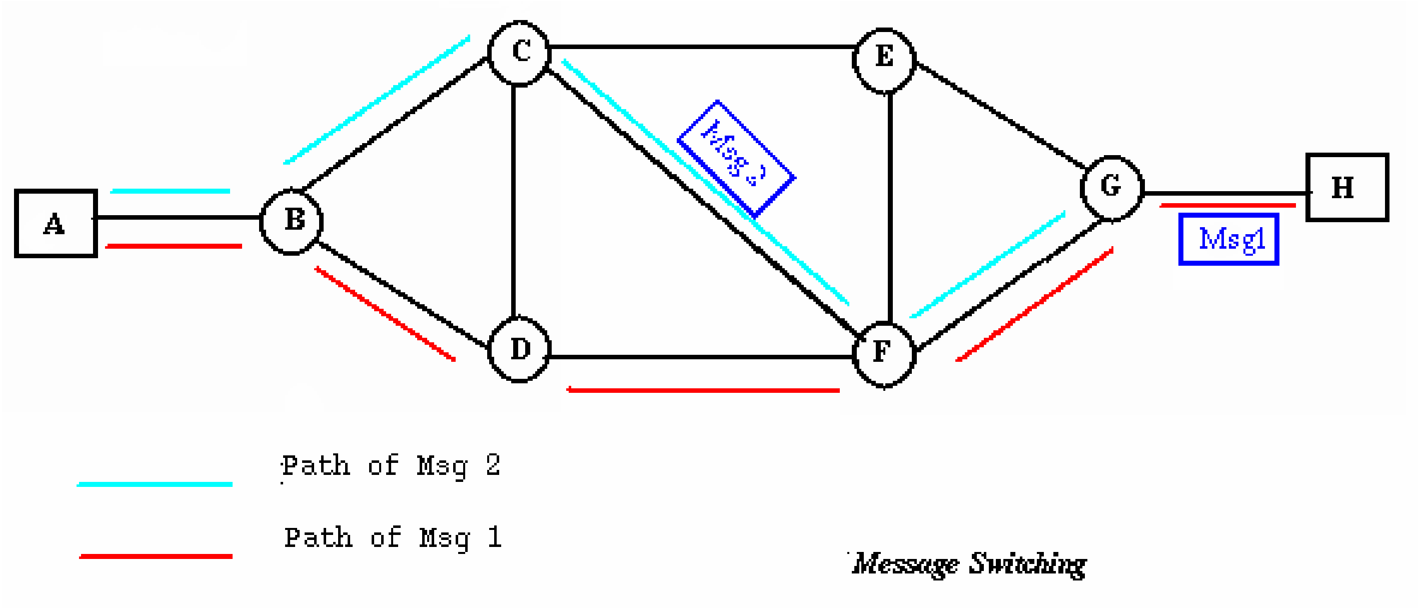
Note

Switching at the physical layer in the traditional telephone network uses the circuit-switching approach.

Message Switching

- With message switching there is no need to establish a dedicated path between two stations.
- When a station sends a message, the destination address is appended to the message.
- The message is then transmitted through the network, in its entirety, from node to node.
- Each node receives the entire message, stores it in its entirety on disk, and then transmits the message to the next node.
- This type of network is called a store-and-forward network.

Message Switching



A message-switching node is typically a general-purpose computer. The device needs sufficient secondary-storage capacity to store the incoming messages, which could be long. A time delay is introduced using this type of scheme due to store- and-forward time, plus the time required to find the next node in the transmission path.

Message Switching

Advantages:

- Channel efficiency can be greater compared to circuit-switched systems, because more devices are sharing the channel.
- Traffic congestion can be reduced, because messages may be temporarily stored in route.
- Message priorities can be established due to store-and-forward technique.
- Message broadcasting can be achieved with the use of broadcast address appended in the message.

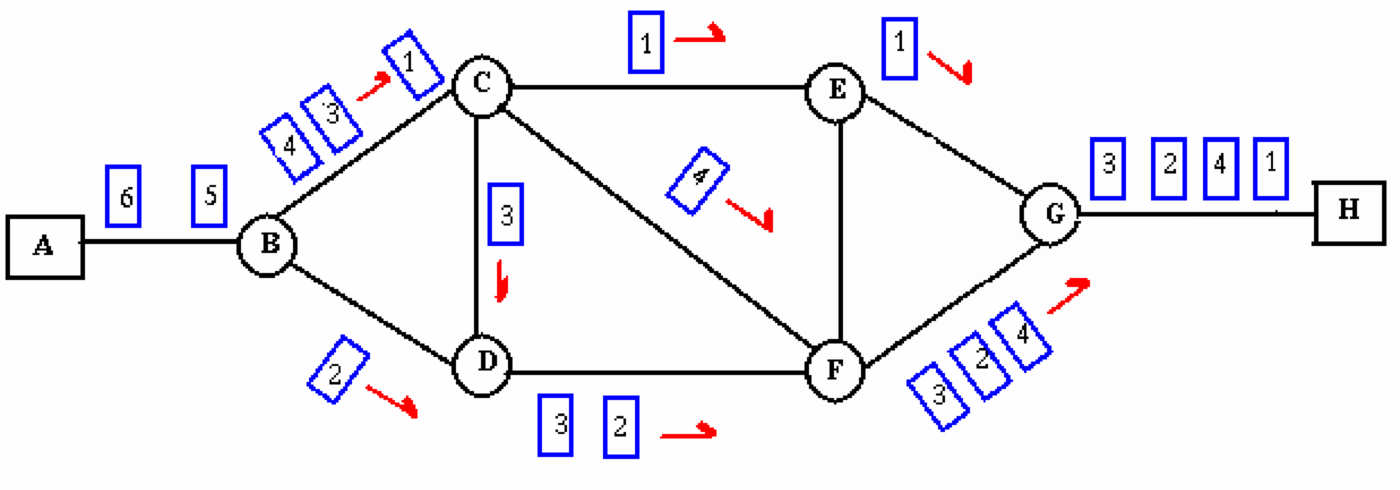
Message Switching

Disadvantages

- Message switching is not compatible with interactive applications.
- Store-and-forward devices are expensive, because they must have large disks to hold potentially long messages.

Packet Switching

- *Packet switching* can be seen as a solution that tries to combine the advantages of message and circuit switching and to minimize the disadvantages of both.
- There are two methods of packet switching: Datagram and virtual circuit.



Packet Switching

Packet Switching

- In both packet switching methods, a message is broken into small parts, called packets.
- Each packet is tagged with appropriate source and destination addresses.
- Since packets have a strictly defined maximum length, they can be stored in main memory instead of disk, therefore access delay and cost are minimized.
- Also the transmission speeds, between nodes, are optimized.
- With current technology, packets are generally accepted onto the network on a first-come, first-served basis. If the network becomes overloaded, packets are delayed or discarded (``dropped'').

Packet size

- The size of the packet can vary from 180 bits, the size for the Datakit® virtual circuit switch designed by Bell Labs for communications and business applications; to 1,024 or 2,048 bits for the 1PSS® switch, also designed by Bell Labs for public data networking; to 53 bytes for ATM switching, such as Lucent Technologies' packet switches.

Packet switching

- In packet switching, the analog signal from your phone is converted into a digital data stream. That series of digital bits is then divided into relatively tiny clusters of bits, called packets. Each packet has at its beginning the digital address -- a long number -- to which it is being sent. The system blasts out all those tiny packets, as fast as it can, and they travel across the nation's digital backbone systems to their destination: the telephone, or rather the telephone system, of the person you're calling.
- They do not necessarily travel together; they do not travel sequentially. They don't even all travel via the same route. But eventually they arrive at the right point -- that digital address added to the front of each string of digital data -- and at their destination are reassembled into the correct order, then converted to analog form, so your friend can understand what you're saying.

Packet Switching: Datagram

- Datagram packet switching is similar to message switching in that each packet is a self-contained unit with complete addressing information attached.
- This fact allows packets to take a variety of possible paths through the network.
- So the packets, each with the same destination address, do not follow the same route, and they may arrive out of sequence at the exit point node (or the destination).
- Reordering is done at the destination point based on the sequence number of the packets.
- It is possible for a packet to be destroyed if one of the nodes on its way is crashed momentarily. Thus all its queued packets may be lost.

Packet Switching:Virtual Circuit

- In the virtual circuit approach, a preplanned route is established before any data packets are sent.
- A logical connection is established when
  - ➢ a sender send a "call request packet" to the receiver and
  - ➢ the receiver send back an acknowledge packet "call accepted packet" to the sender if the receiver agrees on conversational parameters.
- The conversational parameters can be maximum packet sizes, path to be taken, and other variables necessary to establish and maintain the conversation.
- Virtual circuits imply acknowledgements, flow control, and error control, so virtual circuits are reliable.
- That is, they have the capability to inform upper-protocol layers if a transmission problem occurs.

Packet Switching:Virtual Circuit

- In virtual circuit, the route between stations does not mean that this is a dedicated path, as in circuit switching.
- A packet is still buffered at each node and queued for output over a line.
- The difference between virtual circuit and datagram approaches:

➢ With virtual circuit, the node does not need to make a routing decision for each packet.

➢ It is made only once for all packets using that virtual circuit.

Packet Switching: Virtual Circuit

VC's offer guarantees that

- the packets sent arrive in the order sent
- with no duplicates or omissions
- with no errors (with high probability)

regardless of how they are implemented internally.

Advantages of packet switching

Advantages:

- Packet switching is cost effective, because switching devices do not need massive amount of secondary storage.
- Packet switching offers improved delay characteristics, because there are no long messages in the queue (maximum packet size is fixed).
- Packet can be rerouted if there is any problem, such as, busy or disabled links.
- The advantage of packet switching is that many network users can share the same channel at the same time. Packet switching can maximize link efficiency by making optimal use of link bandwidth.

Disadvantages of packet switching

Disadvantages:

- Protocols for packet switching are typically more complex.
- It can add some initial costs in implementation.
- If packet is lost, sender needs to retransmit the data.
- Another disadvantage is that packet-switched systems still can’t deliver the same quality as dedicated circuits in applications requiring very little delay - like voice conversations or moving images.

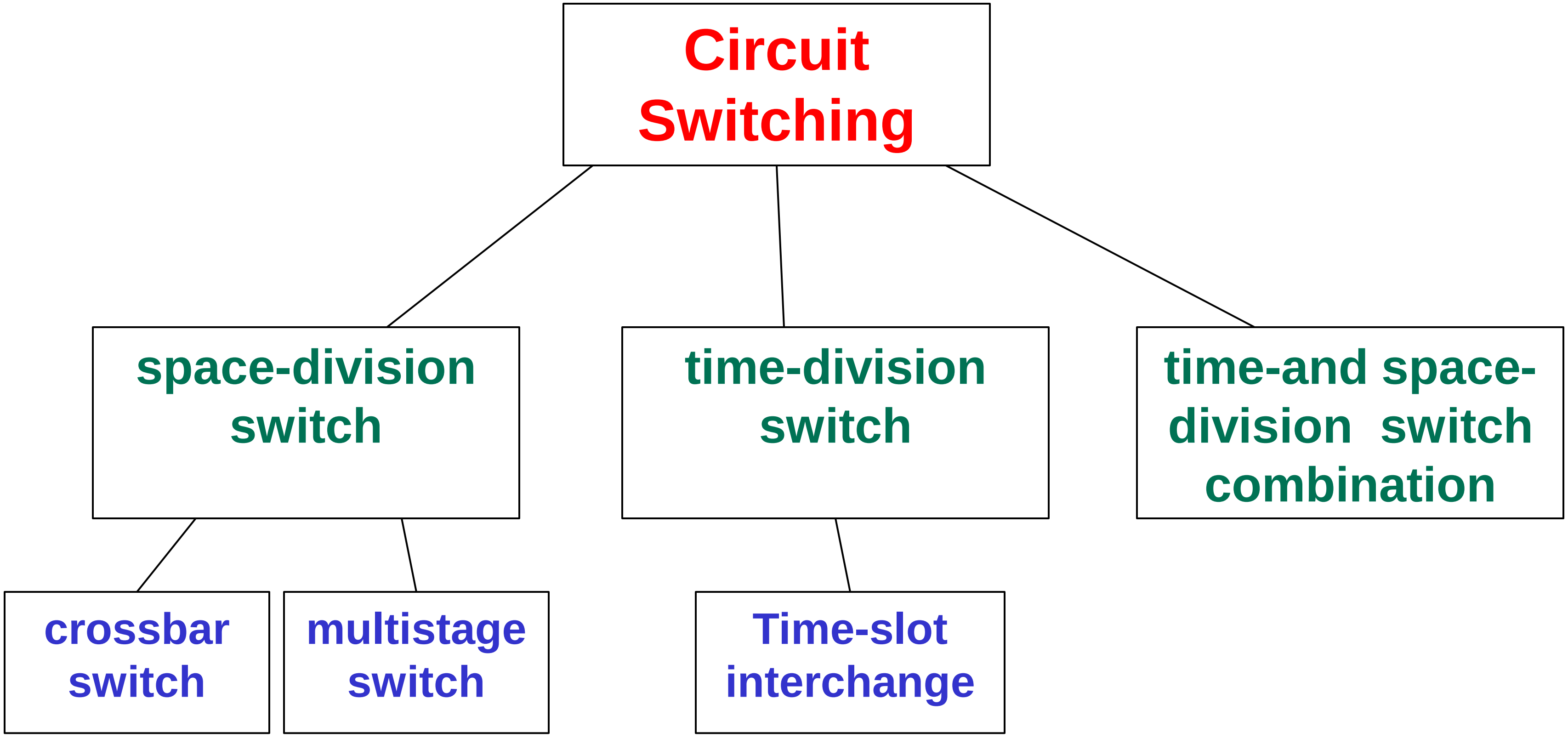
8-4 STRUCTURE OF A SWITCH

We use switches in circuit-switched and packet-switched networks. In this section, we discuss the structures of the switches used in each type of network.

Topics discussed in this section:

Structure of Circuit Switches
Structure of Packet Switches

Circuit Switches



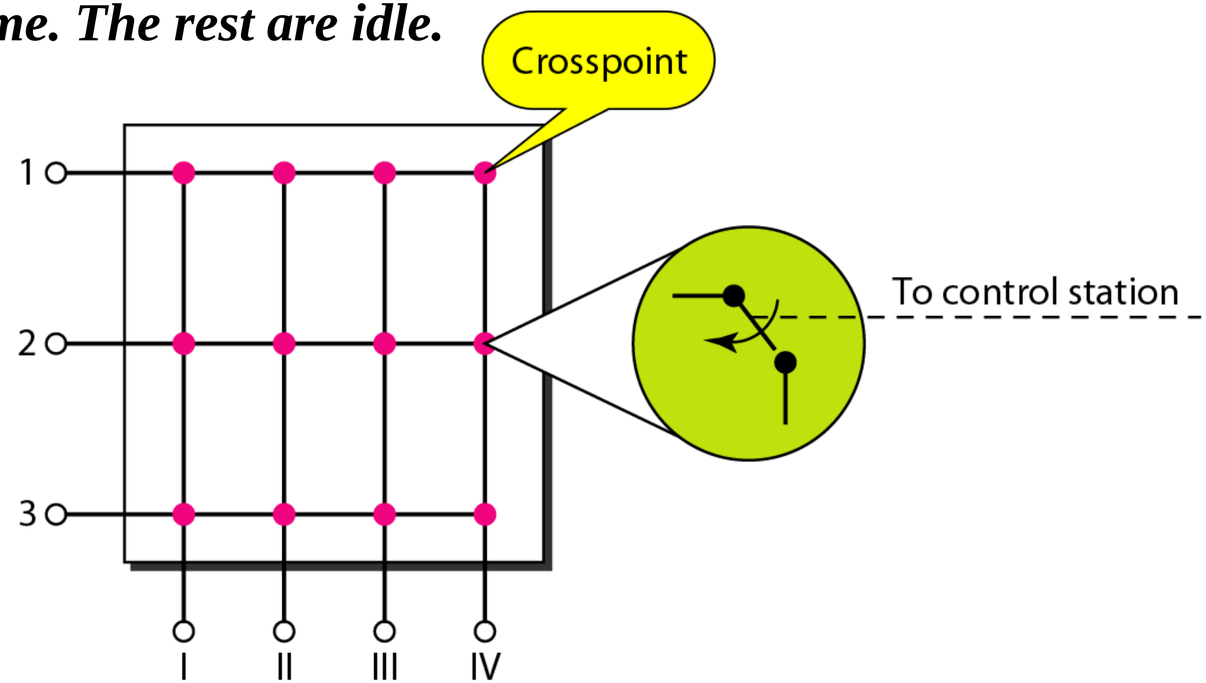
Space-Division Switch

In space-division switching, the paths in the circuit are separated from one another spatially. This technology was originally designed for use in analog networks but is used currently in both analog and digital networks. It has evolved through a long history of many designs.

- ***Crossbar Switches***
- ***Multistage Switches***

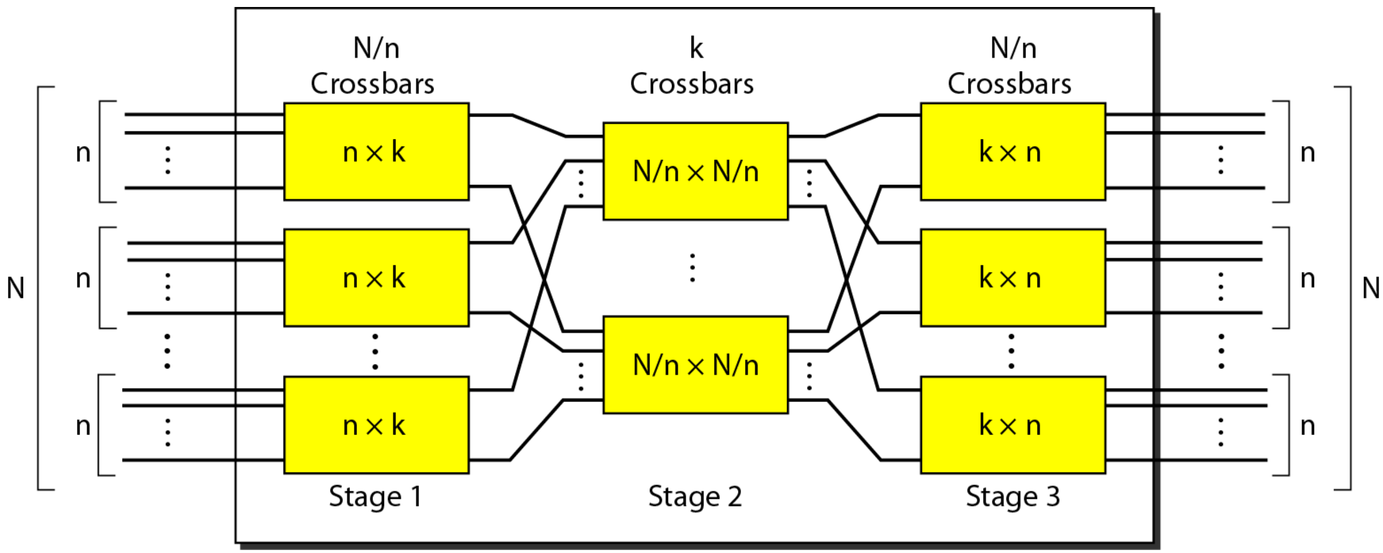
Crossbar Switch

A crossbar switch connects n inputs to m outputs in a grid, using electronic microswitches (transistors) at each crosspoint. 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. For example, to connect 1000 inputs to 1000 outputs requires a switch with 1,000,000 crosspoints. A crossbar with this 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.



Multistage switch

The solution to the limitations of the crossbar switch is the multistage switch, which combines crossbar switches in several (normally three) stages, as shown in 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.



Design of a multistage switch

To design a three-stage switch, we follow these steps:
1. 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.
2. We use k crossbars, each of size (N/n) x (N/n) in the middle stage.
3. 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:

$$(N/n)(n \times k)+k\,(N/n \times N/n) + (N/n)(k \times n)=2kN+k(N/n)^2$$

Note

In a three-stage switch, the total number of crosspoints is

$2kN + k(N/n)^2$

which is much smaller than the number of crosspoints in a single-stage switch (N^2).

Example 8.3

Design a three-stage, 200 × 200 switch ($N = 200$) with $k = 4$ and $n = 20$.

Example 8.3 - solution

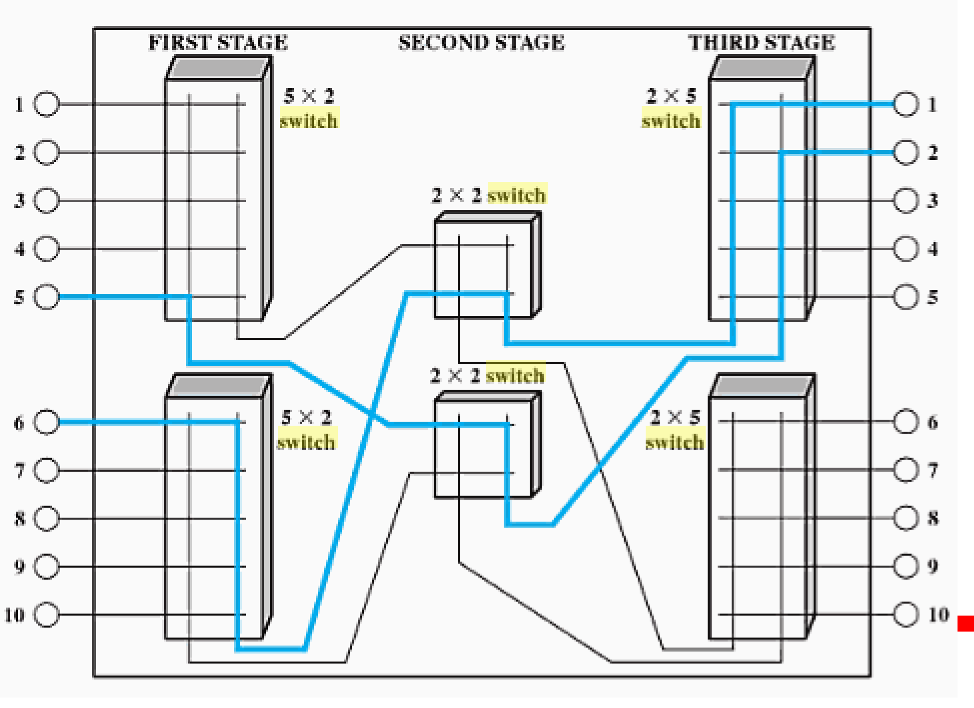
Design a three-stage, 200 × 200 switch (N = 200) with k = 4 and n = 20.

Solution

In the first stage we have N/n or 10 crossbars, each of size 20 × 4. In the second stage, we have 4 crossbars, each of size 10 × 10. In the third stage, we have 10 crossbars, each of size 4 × 20. The total number of crosspoints is $2kN + k(N/n)^2$, or 2000 crosspoints. This is 5 percent of the number of crosspoints in a single-stage switch (200 × 200 = 40,000).

Blocking

The multistage switch in Example 8.3 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. In a single-stage switch, blocking does not occur because every combination of input and output has its own crosspoint; there is always a path. In large systems, such as those having 10,000 inputs and outputs, the number of stages can be increased to cut down on the number of crosspoints required. As the number of stages increases, however, possible blocking increases as well.



Clos criteria for non blocking switching

- ***Number of users, N is given***
- ***n, k, number of stages are design choices***
- ***Clos criteria: condition of nonblocking***

1. $n = (N/2)^{1/2}$

2. $k \geq 2n - 1$.

3. Crosspoints $\geq 4N\,[(2N)^{1/2} - 1]$

Note

According to the Clos criterion:

$n = (N/2)^{1/2}$

$k > 2n - 1$

Crosspoints $\geq$ $4N\,[(2N)^{1/2} - 1]$

Example 8.4

Redesign the previous three-stage, 200 × 200 switch, using the Clos criteria with a minimum number of crosspoints.

Example 8.4 - solution

Redesign the previous three-stage, 200 × 200 switch, using the Clos criteria with a minimum number of crosspoints.

Solution

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 × 19 crosspoints. In the second stage, we have 19 crossbars, each with 10 × 10 crosspoints. In the third stage, we have 20 crossbars each with 19 × 10 crosspoints. The total number of crosspoints is 20(10 × 19) + 19(10 × 10) + 20(19 ×10) = 9500.

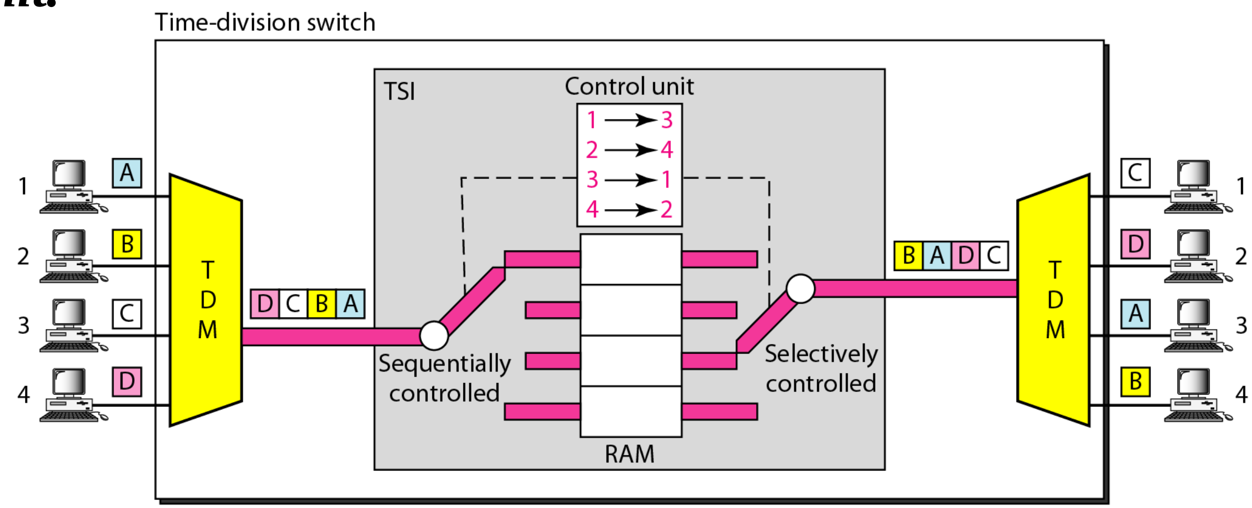
Time-Division Switch

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

Time-slot interchange

Imagine that each input line wants to send data to an output line according to the following pattern: 1 ➡ 3 2 ➡ 4 3 ➡ 1 4 ➡ 2

The figure combines a TDM multiplexer, a TDM demultiplexer, and a TSI consisting of random access memory (RAM) with several memory locations. The size of each location is the same as the size of a single time slot. The number of locations is the same as the number of inputs (in most cases, the numbers of inputs and outputs are equal). 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.



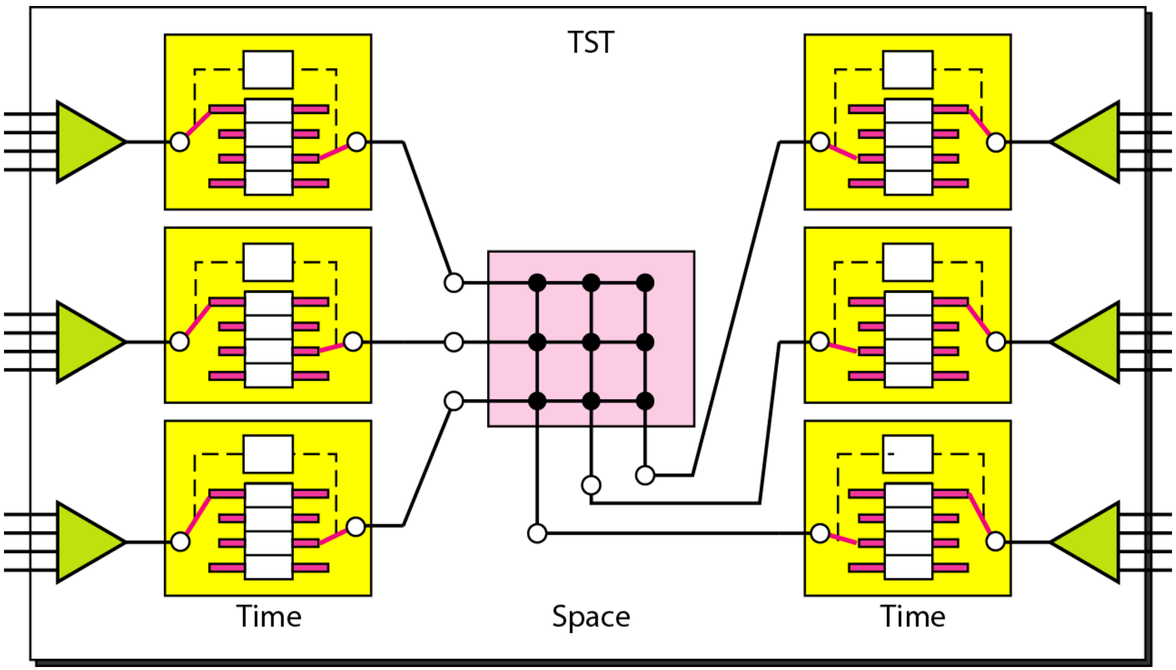
Time- and Space-Division Switch Combinations

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.

We can 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.

Figure 8.20 *Time-space-time switch*

Figure 8.20 shows a simple TST switch that consists of two time stages and one space stage and has 12 inputs and 12 outputs. Instead of one time-division switch, it divides the inputs into three groups (of four inputs each) and directs them to three timeslot interchanges. The result is that the average delay is one-third of what would result from using one time-slot interchange to handle all 12 inputs. The last stage is a mirror image of the first stage. The middle stage is a space division switch (crossbar) that connects the TSI groups to allow connectivity between all possible input and output pairs (e.g., to connect input 3 of the first group to output 7 of the second group).



8-2 DATAGRAM NETWORKS

In data communications, we need to send messages from one end system to another. If the message is going to pass through a packet-switched network, it needs to be divided into packets of fixed or variable size. The size of the packet is determined by the network and the governing protocol.

Topics discussed in this section:

Routing Table
Efficiency
Delay
Datagram Networks in the Internet

Note

In a packet-switched network, there is no resource reservation; resources are allocated on demand on a first-come first-served basis.

Characteristics of Datagram Switching

•*Each packet is treated independently of all others. Even if a packet is part of a multipacket transmission, the network treats it as though it existed alone. Packets in this approach are referred to as datagrams.*

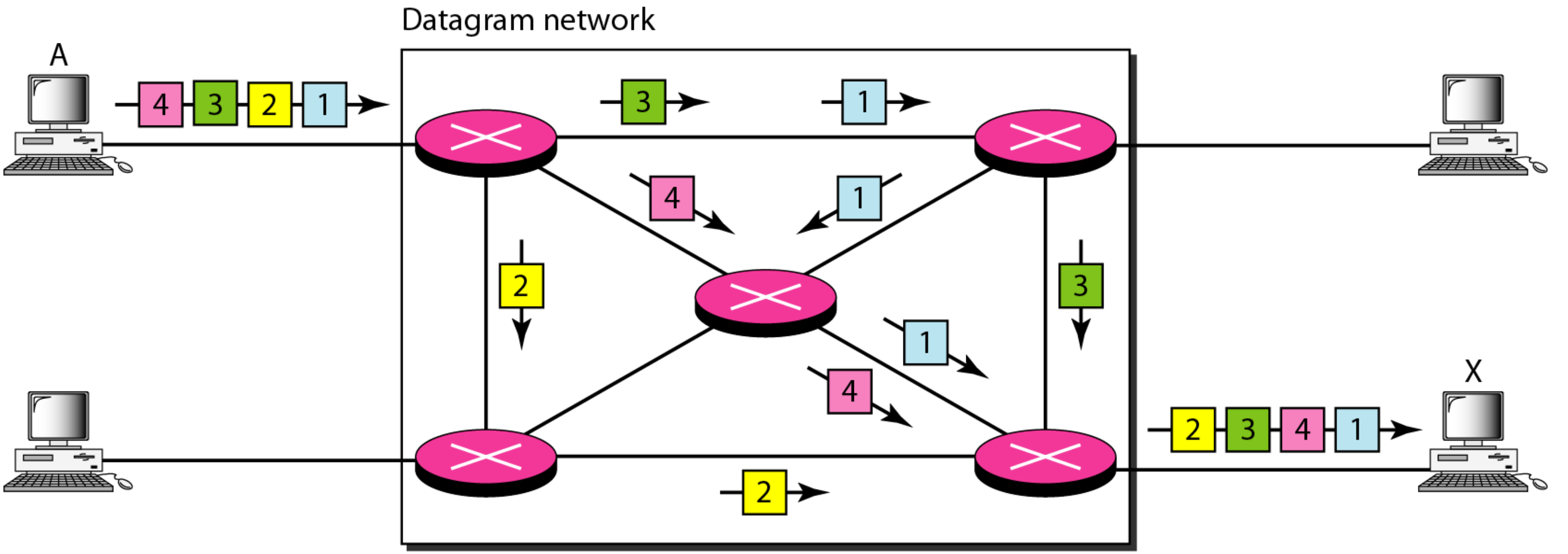
•*Datagram switching is normally done at the network layer.*

•*The switches in a datagram network are traditionally referred to as routers.*

•*Datagrams of a transmission can arrive at their destination out of order with different delays between the packets. Packets may also be lost or dropped because of a lack of resources. In most protocols, it is the responsibility of an upper-layer protocol to reorder the datagrams or ask for lost datagrams before passing them on to the application.*

•*The datagram networks are sometimes referred to as connectionless networks. The term connectionless here means that the switch (packet switch) does not keep information about the connection state. There are no setup or teardown phases. Each packet is treated the same by a switch regardless of its source or destination.*

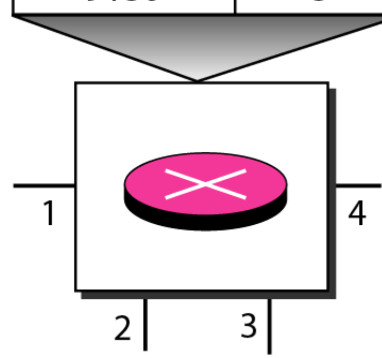
Figure 8.7 *A datagram network with four switches (routers)*



Routing Table

In this type of network, each switch (or packet switch) has a routing table which is based on the destination address. The routing tables are dynamic and are updated periodically. The destination addresses and the corresponding forwarding output ports are recorded in the tables.

Destination address	Output port
1232	1
4150	2
⋮	⋮
9130	3



Note

The destination address in the header of a packet in a datagram network remains the same during the entire journey of the packet.

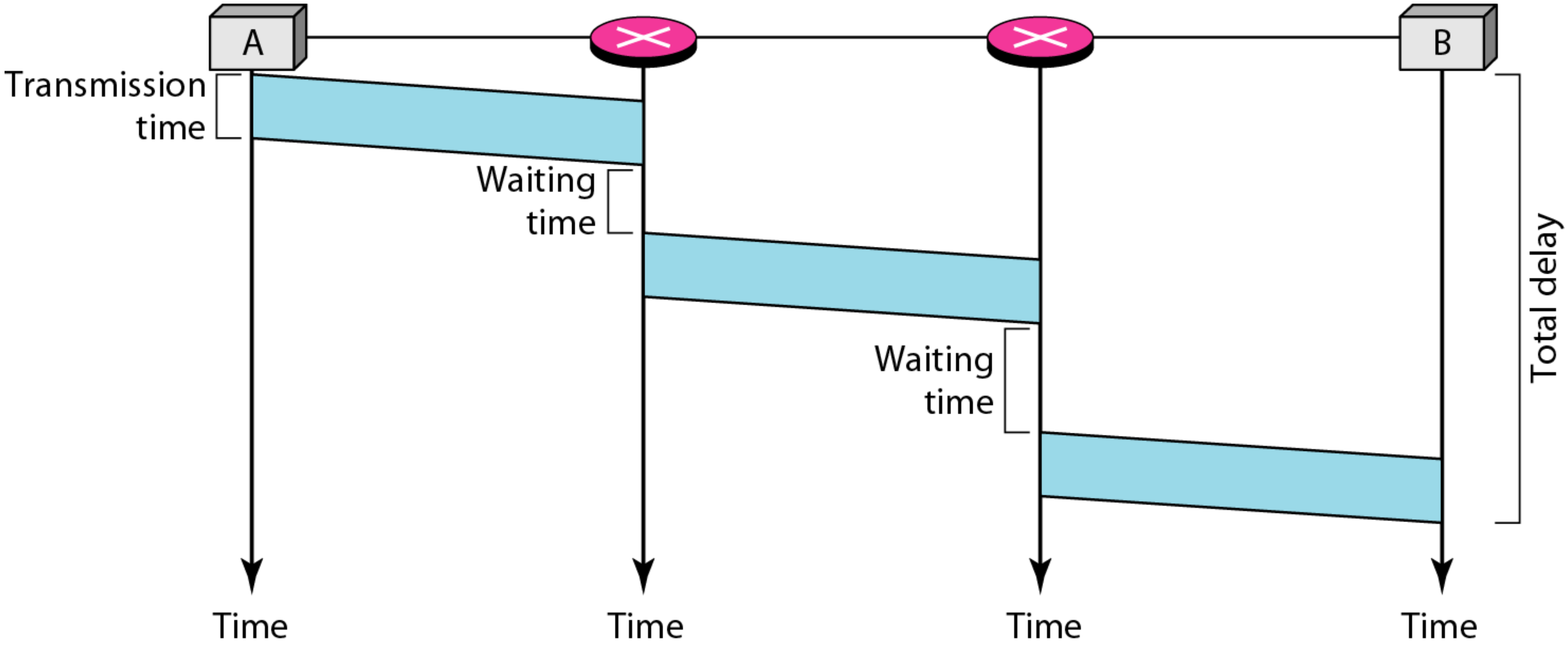
Efficiency

The efficiency of a datagram network is better than that of a circuit-switched network; resources are allocated only when there are packets to be transferred. If a source sends a packet and there is a delay of a few minutes before another packet can be sent, the resources can be reallocated during these minutes for other packets from other sources.

Delay

There may be greater delay in a datagram network than in a virtual-circuit network.
Although there are no setup and teardown phases, each packet may experience a wait at a switch before it is forwarded. In addition, since not all packets in a message necessarily travel through the same switches, the delay is not uniform for the packets of a message.

Figure 8.9 *Delay in a datagram network*



Note

Internet has chosen the datagram approach to switching at the network layer. It uses the universal addresses defined in the network layer to route packets from the source to the destination.

8-3 VIRTUAL-CIRCUIT NETWORKS

A virtual-circuit network is a cross between a circuit-switched network and a datagram network. It has some characteristics of both.

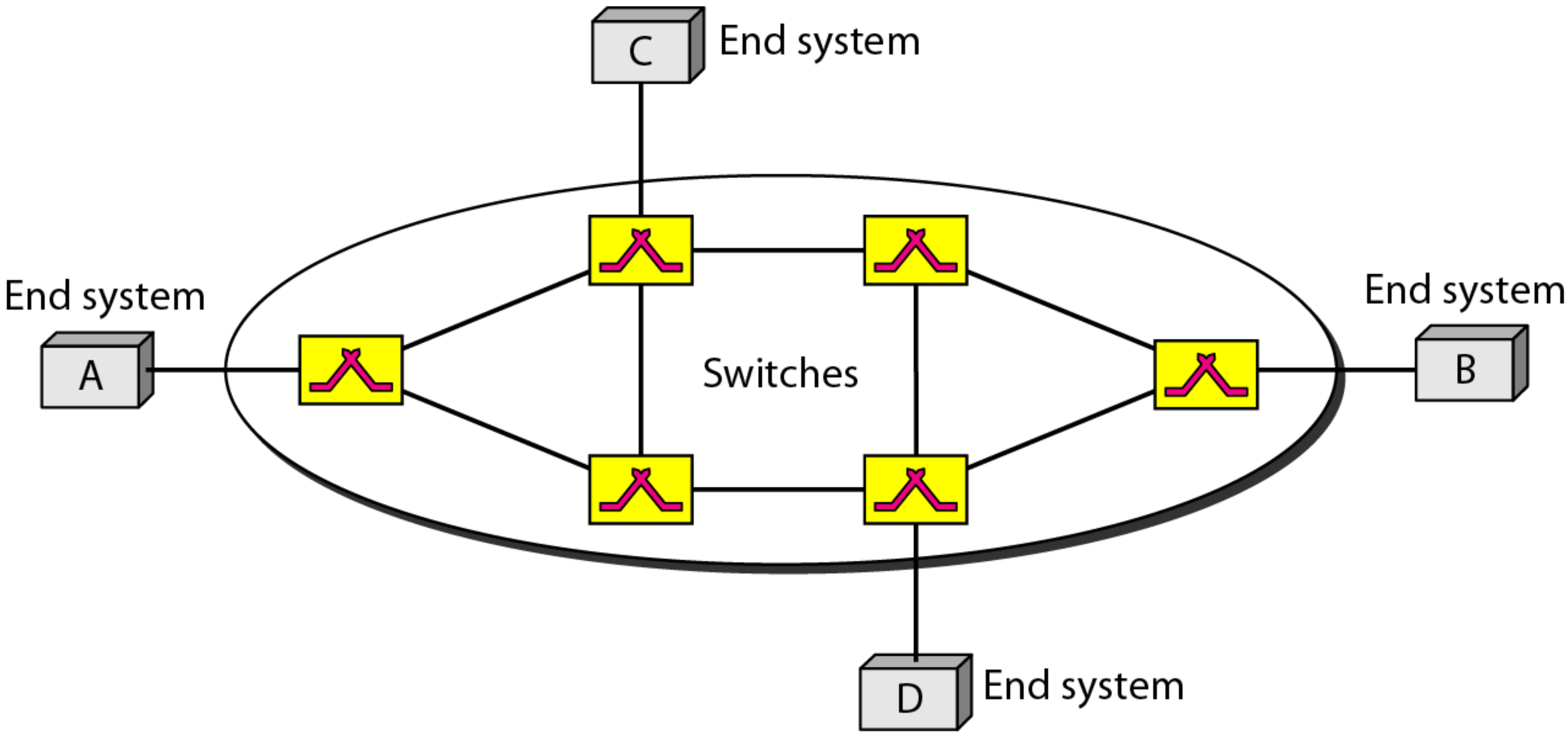
Topics discussed in this section:

Addressing
Three Phases
Efficiency
Delay
Circuit-Switched Technology in WANs

Characteristics of Virtual-Circuit Switching

- *As in a circuit-switched network, there are setup and teardown phases in addition to the data transfer phase.*
- *Resources can be allocated during the setup phase, as in a circuit-switched network, or on demand, as in a datagram network.*
- *As in a datagram network, data are packetized and each packet carries an address in the header. However, the address in the header has local jurisdiction (it defines what should be the next switch and the channel on which the packet is being carried), not end-to-end jurisdiction*.
- *As in a circuit-switched network, all packets follow the same path established during the connection.*
- *A virtual-circuit network is normally implemented in the data link layer, while a circuit-switched network is implemented in the physical layer and a datagram network in the network layer.*

Figure 8.10 *Virtual-circuit network*



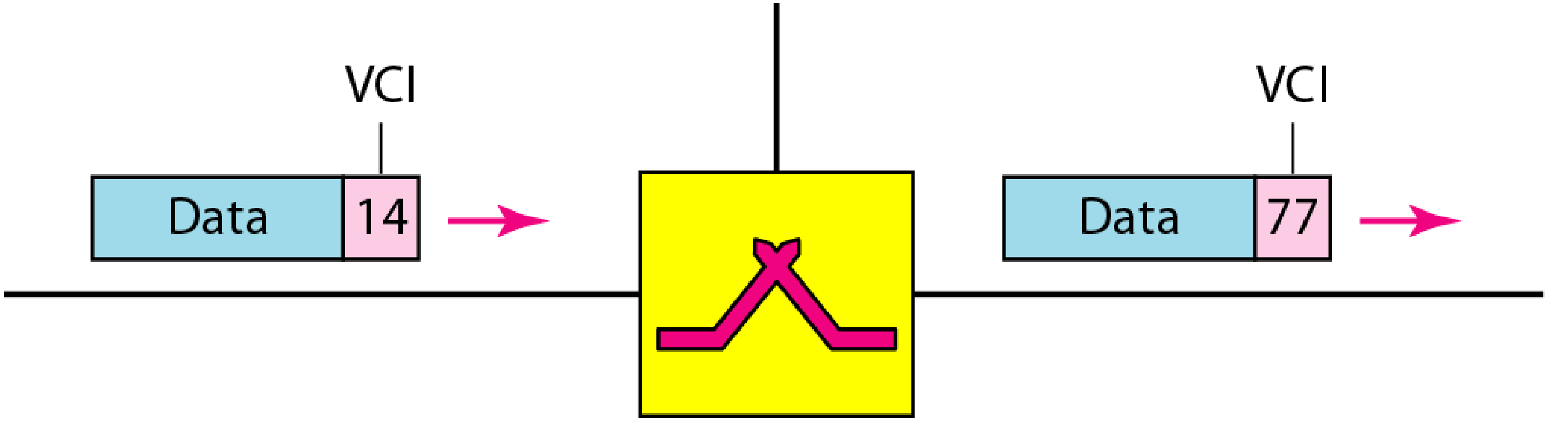
Addressing

In a virtual-circuit network, two types of addressing are involved:

Global Addressing: ***A source or a destination needs to have a global address-an address that can be unique in the scope of the network or internationally if the network is part of an international network. However, we will see that a global address in virtual-circuit networks is used only to create a virtual-circuit identifier.***

Local (Virtual-Circuit Identifier): ***The identifier that is actually used for data transfer is called the virtual-circuit identifier (VCI). A VCI, unlike a global address, is a small number that has only switch scope; it is used by a frame between two switches. When a frame arrives at a switch, it has a VCI; when it leaves, it has a different VCI. Note that a VCI does not need to be a large number since each switch can use its own unique set of VCIs.***

Figure 8.11 *Virtual-circuit identifier*



Three Phases

As in a circuit-switched network, a source and destination need to go through three phases in a virtual-circuit network: setup, data transfer, and teardown.

Setup Phase

In the setup phase, a switch creates an entry for a virtual circuit. For example, suppose source A needs to create a virtual circuit to B. Two steps are required: the setup request and the acknowledgment.

Setup Request: ***A setup request frame is sent from the source to the destination. Figure 8.14 shows the process.***

a)Source A sends a setup frame to switch 1.

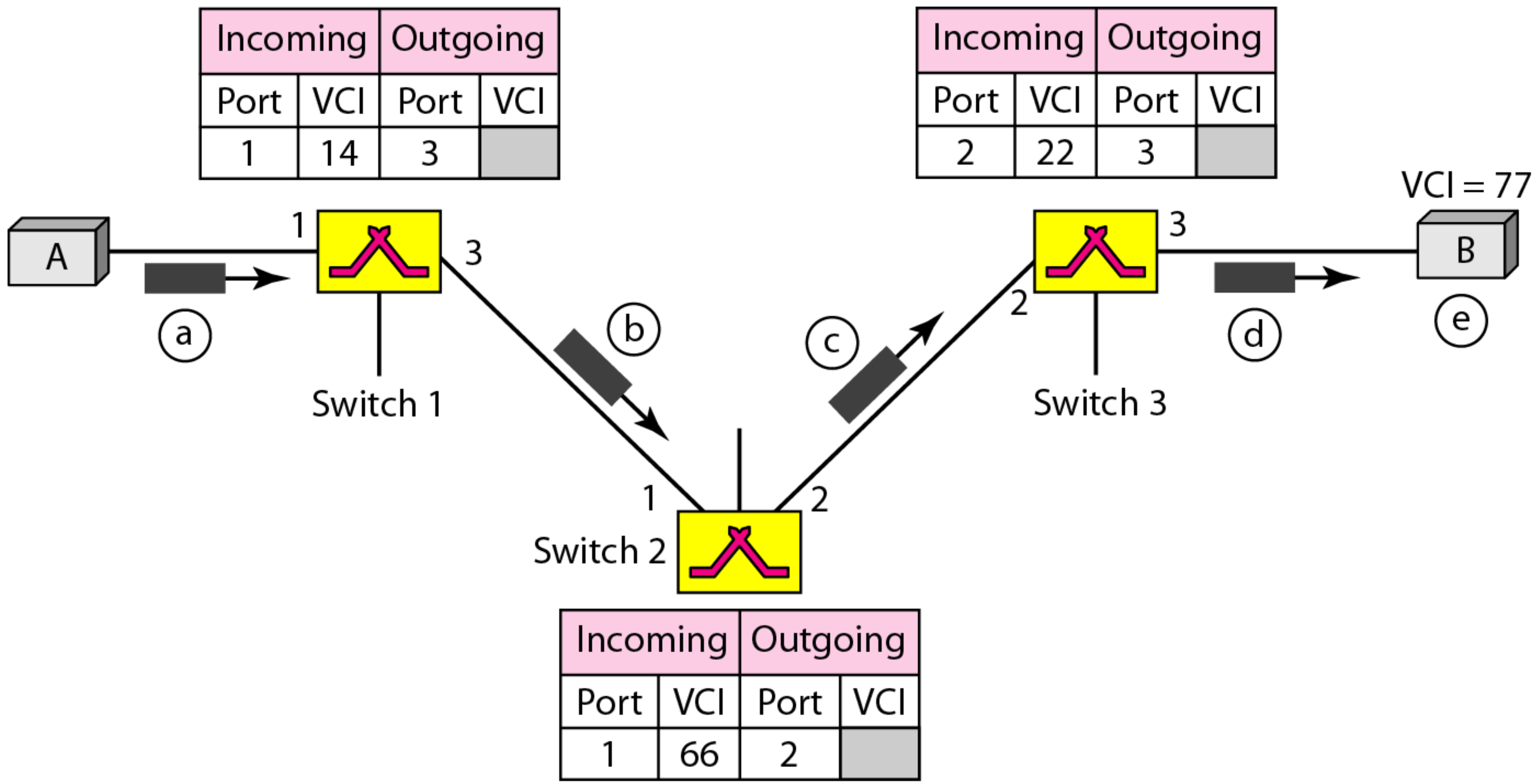
b)Switch 1 receives the setup request frame. It knows that a frame going from A to B goes out through port 3. The switch, in the setup phase, acts as a packet switch; it has a routing table which is different from the switching table. For the moment, assume that it knows the output port. The switch creates an entry in its table for this virtual circuit, but it is only able to fill three of the four columns. The switch assigns the incoming port (1) and chooses an available incoming VCI (14) and the outgoing port (3). It does not yet know the outgoing VCI, which will be found during the acknowledgment step. The switch then forwards the frame through port 3 to switch 2.

c)Switch 2 receives the setup request frame. The same events happen here as at switch 1; three columns of the table are completed: in this case, incoming port (l), incoming VCI (66), and outgoing port (2).

d)Switch 3 receives the setup request frame. Again, three columns are completed: incoming port (2), incoming VCI (22), and outgoing port (3).

e)Destination B receives the setup frame, and if it is ready to receive frames from A, it assigns a VCI to the incoming frames that come from A, in this case 77. This VCI lets the destination know that the frames come from A, and not other sources.

Figure 8.14 *Setup request in a virtual-circuit network*



Setup Phase (cont.)

Acknowledgment: ***A special frame, called the acknowledgment frame, completes the entries in the switching tables. Figure 8.15 shows the process.***

a)The destination sends an acknowledgment to switch 3. The acknowledgment carries the global source and destination addresses so the switch knows which entry in the table is to be completed. The frame also carries VCI 77, chosen by the destination as the incoming VCI for frames from A. Switch 3 uses this VCI to complete the outgoing VCI column for this entry. Note that 77 is the incoming VCI for destination B, but the outgoing VCI for switch 3.

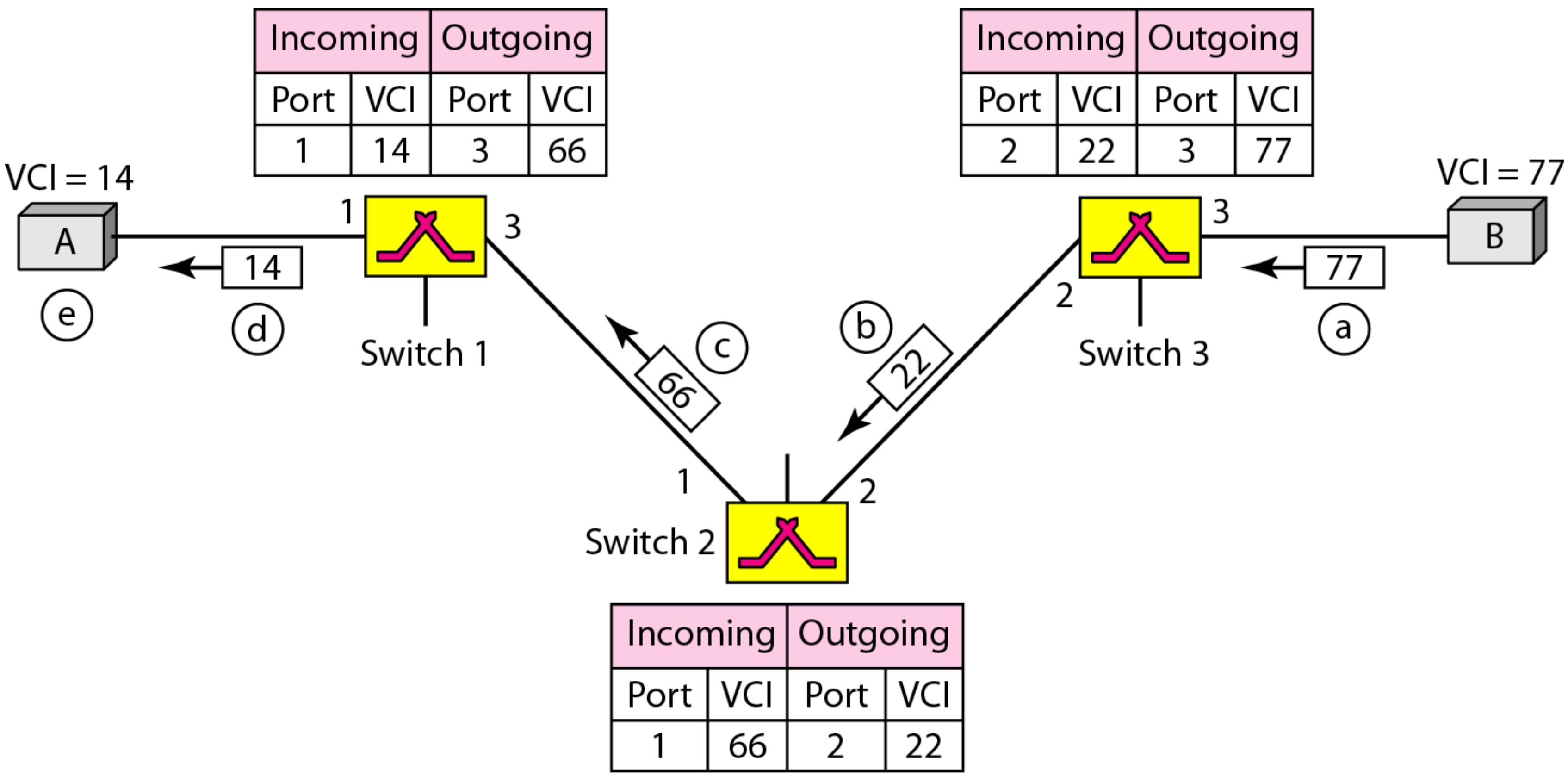
b) Switch 3 sends an acknowledgment to switch 2 that contains its incoming VCI in the table, chosen in the previous step. Switch 2 uses this as the outgoing VCI in the table.

c)Switch 2 sends an acknowledgment to switch 1 that contains its incoming VCI in the table, chosen in the previous step. Switch 1 uses this as the outgoing VCI in the table.

d)Finally switch 1 sends an acknowledgment to source A that contains its incoming VCI in the table, chosen in the previous step.

e)The source uses this as the outgoing VCI for the data frames to be sent to destination B.

Figure 8.15 *Setup acknowledgment in a virtual-circuit network*



Data Transfer Phase

To transfer a frame from a source to its destination, all switches need to have a table entry for this virtual circuit. The table, in its simplest form, has four columns. This means that the switch holds four pieces of information for each virtual circuit that is already set up. We show later how the switches make their table entries, but for the moment we assume that each switch has a table with entries for all active virtual circuits. Figure 8.12 shows such a switch and its corresponding table.

Figure 8.12 shows a frame arriving at port 1 with a VCI of 14. When the frame arrives, the switch looks in its table to find port 1 and a VCI of 14. When it is found, the switch knows to change the VCI to 22 and send out the frame from port 3.

Figure 8.13 shows how a frame from source A reaches destination B and how its VCI changes during the trip. Each switch changes the VCI and routes the frame.

The data transfer phase is active until the source sends all its frames to the destination. The procedure at the switch is the same for each frame of a message. The process creates a virtual circuit, not a real circuit, between the source and destination.

Figure 8.12 *Switch and tables in a virtual-circuit network*

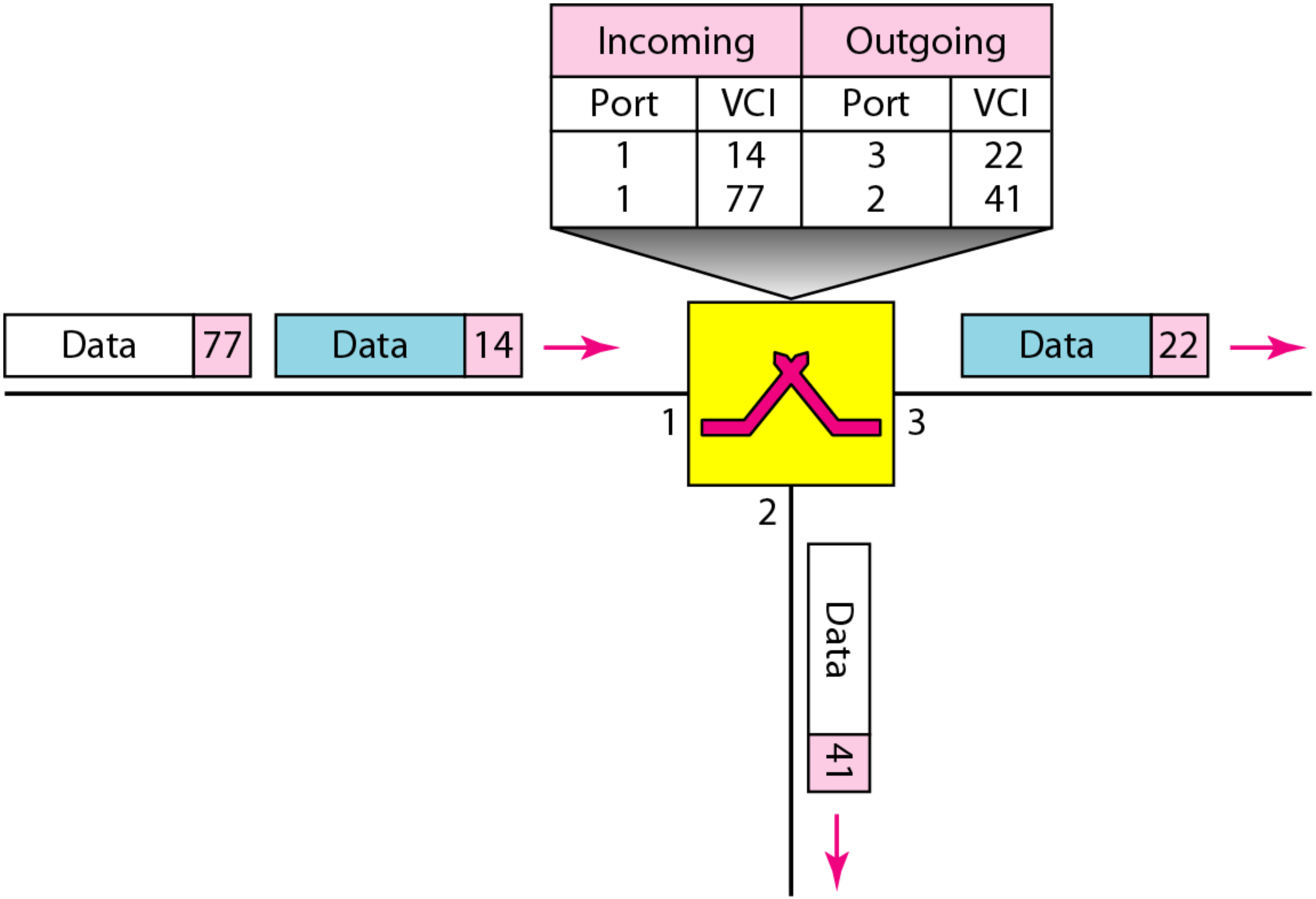
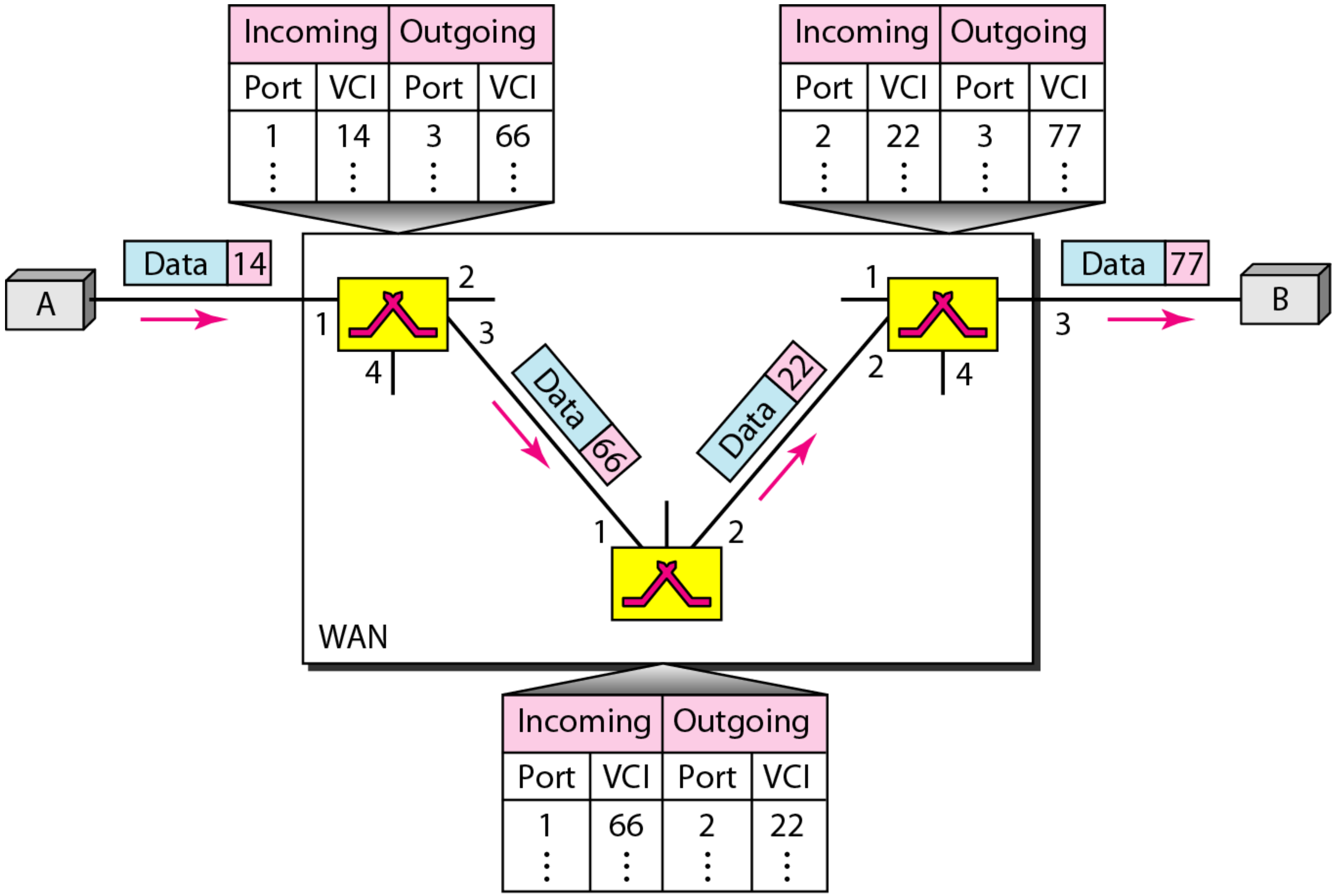


Figure 8.13 *Source-to-destination data transfer in a virtual-circuit network*



Teardown Phase

In this phase, source A, after sending all frames to B, sends a special frame called a teardown request. Destination B responds with a teardown confirmation frame. All switches delete the corresponding entry from their tables.

Efficiency

As we said before, resource reservation in a virtual-circuit network can be made during the setup or can be on demand during the data transfer phase. In the first case, the delay for each packet is the same; in the second case, each packet may encounter different delays. There is one big advantage in a virtual-circuit network even if resource allocation is on demand. The source can check the availability of the resources, without actually reserving it.

Note

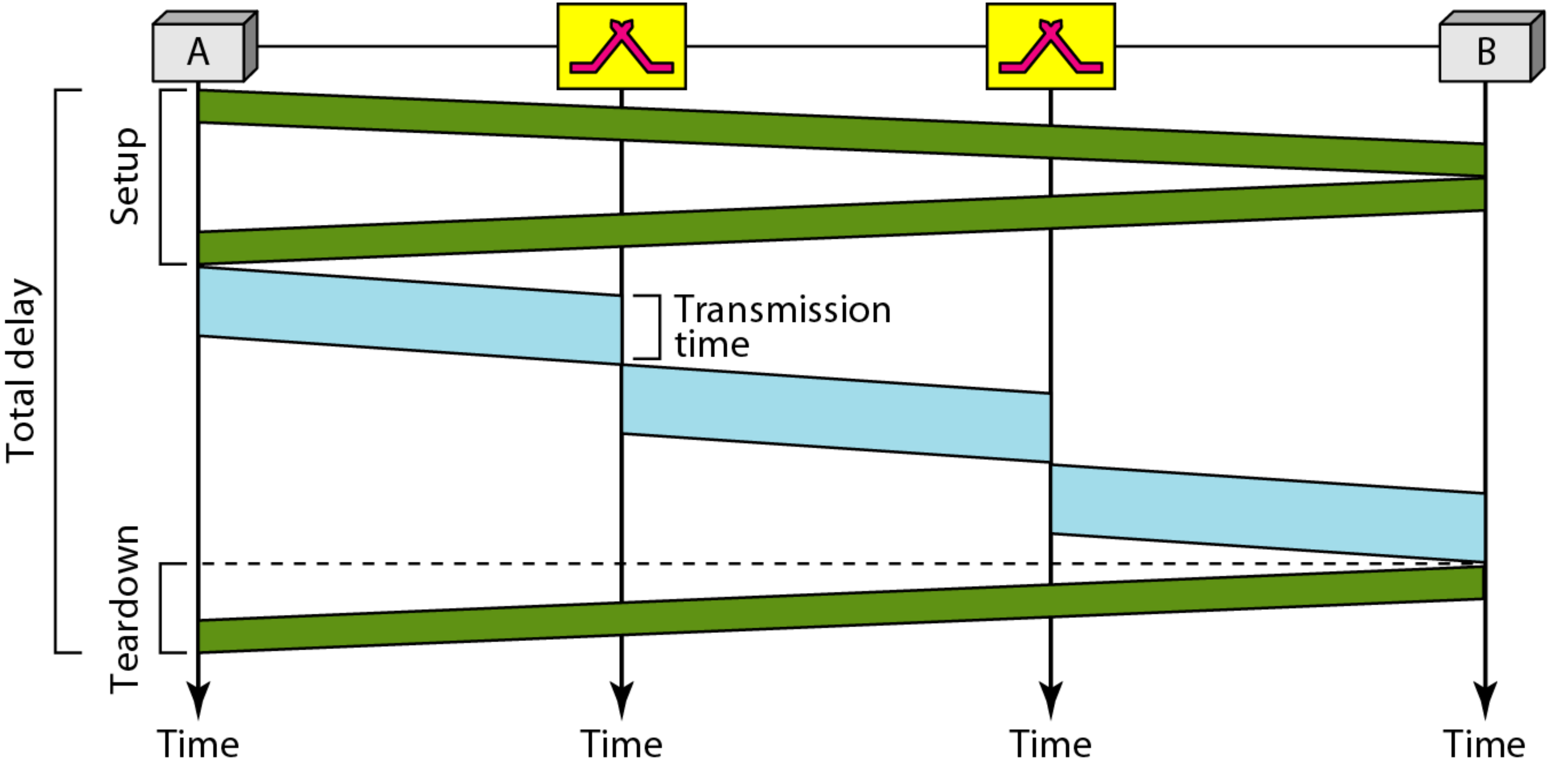
In virtual-circuit switching, all packets belonging to the same source and destination travel the same path; but the packets may arrive at the destination with different delays if resource allocation is on demand.

Delay

In a virtual-circuit network, there is a one-time delay for setup and a one-time delay for teardown. If resources are allocated during the setup phase, there is no wait time for individual packets. Figure 8.16 shows the delay for a packet traveling through two switches in a virtual-circuit network.
The packet is traveling through two switches (routers). There are three transmission times (3T), three propagation times (3τ), data transfer depicted by the sloping lines, a setup delay (which includes transmission and propagation in two directions), and a teardown delay (which includes transmission and propagation in one direction).
We ignore the processing time in each switch. The total delay time is

Total delay = *3T+ 3τ + setup delay + teardown delay*

Figure 8.16 *Delay in a virtual-circuit network*

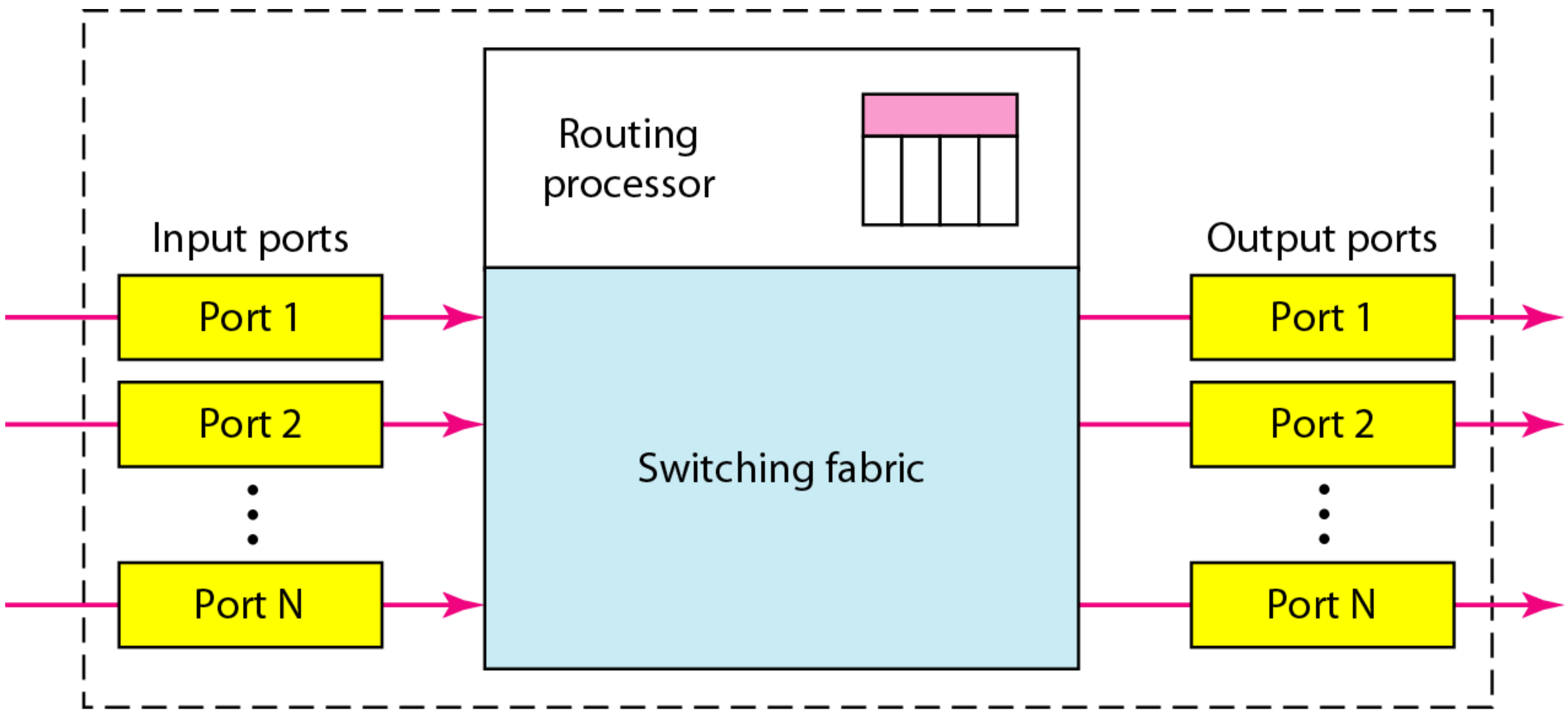


Note

Switching at the data link layer in a switched WAN is normally implemented by using virtual-circuit techniques.

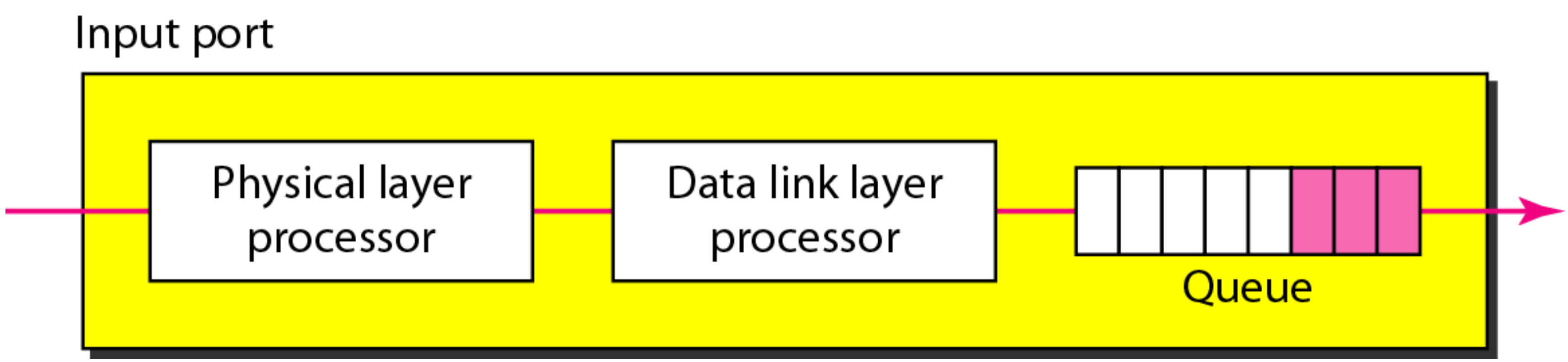
Structure of Packet Switches

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 Figure 8.21.



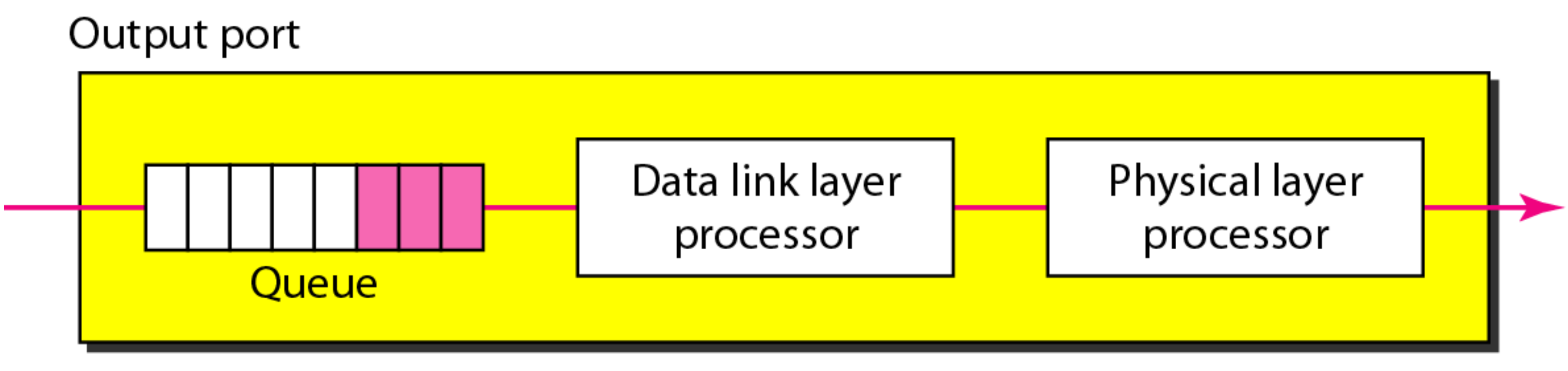
Input port

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 packet is now ready to be routed by the network layer. In addition to a physical layer processor and a data link processor, the input port has buffers (queues) to hold the packet before it is directed to the switching fabric.
Figure 8.22 shows a schematic diagram of an input port.



Output port

The output port performs the same functions as the input port, but in the reverse order.
First the outgoing packets are queued, then the packet is encapsulated in a frame, and finally the physical layer functions are applied to the frame to create the signal to be sent on the line. Figure 8.23 shows a schematic diagram of an output port..



Routing Processor

The routing processor performs the functions of the network layer. The destination address is used to find the address of the next hop and, at the same time, the output port number from which the packet is sent out. This activity is sometimes referred to as table lookup because the routing processor searches the routing table. In the newer packet switches, this function of the routing processor is being moved to the input ports to facilitate and expedite the process.

Switching Fabrics

The most difficult task in a packet switch is to move the packet from the input queue to the output queue. The speed with which this is done affects the size of the input/output queue and the overall delay in packet delivery. In the past, when a packet switch was actually a dedicated computer, the memory of the computer or a bus was used as the switching fabric. The input port stored the packet in memory; the output port retrieved the packet from memory. Today, 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 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 $\log_2 n$ 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. Figure 8.24 shows a banyan switch with eight inputs and eight outputs. The number of stages is $\log_2(8) = 3$.

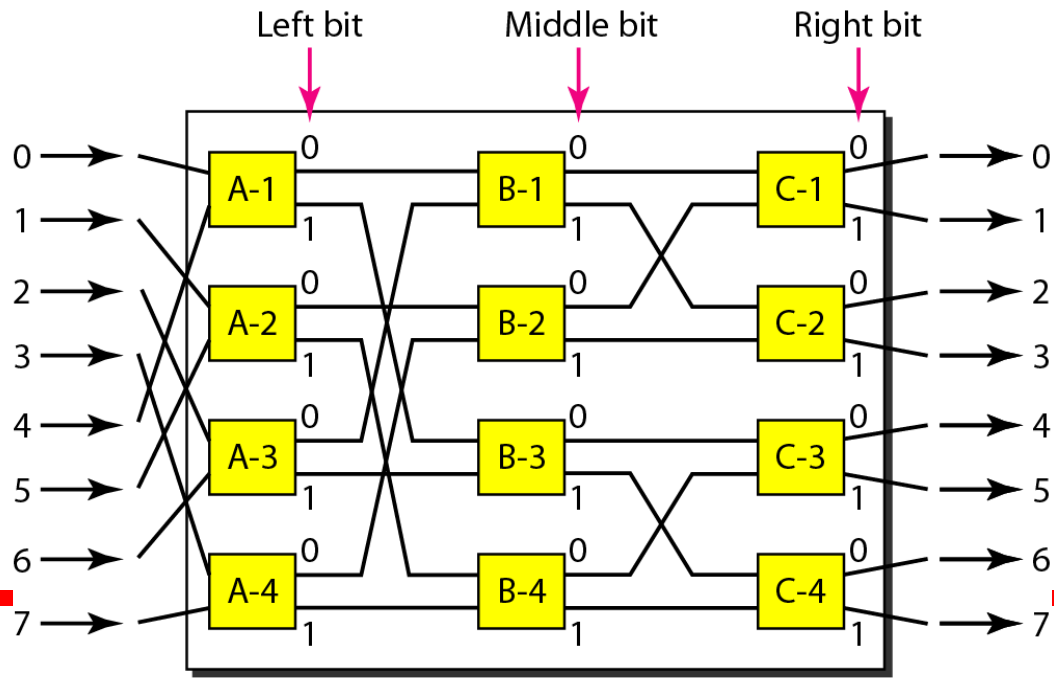
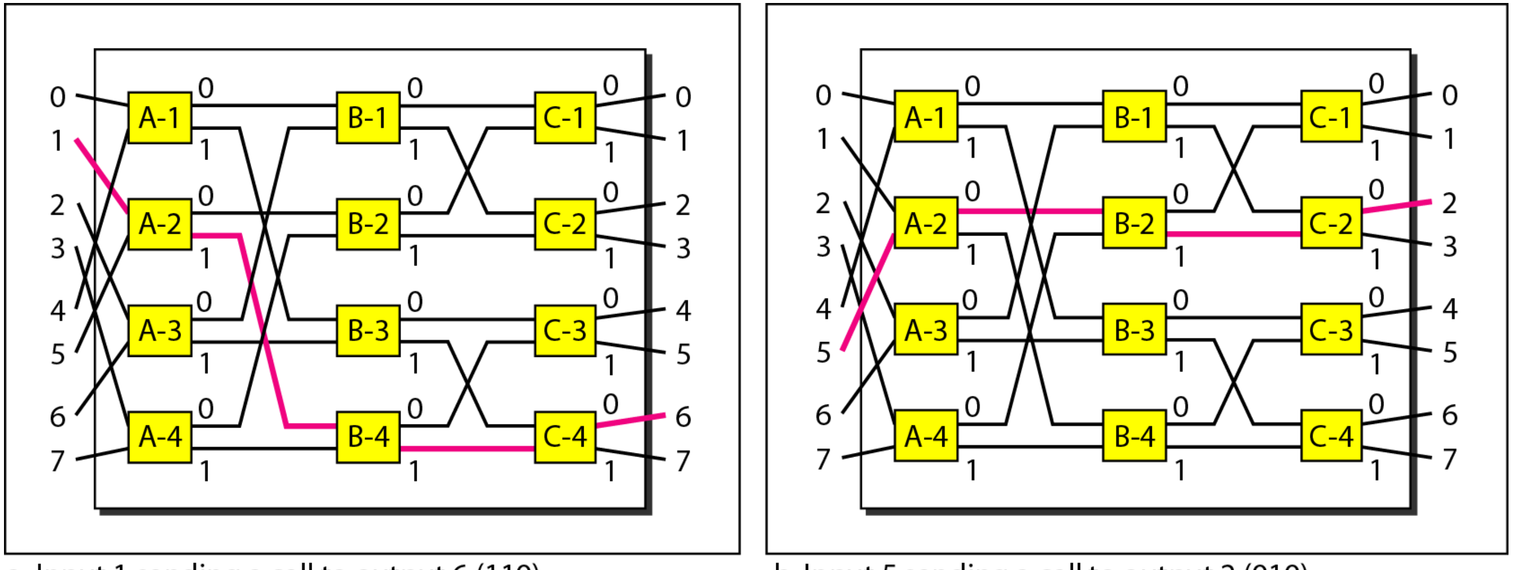


Figure 8.25 *Examples of routing in a banyan switch*

In part a, a packet has arrived at input port 1 and must go to output port 6 (110 in binary). The first microswitch (A-2) routes the packet based on the first bit (1), the second microswitch (B-4) routes the packet based on the second bit (1), and the third microswitch (C-4) routes the packet based on the third bit (0). In part b, a packet has arrived at input port 5 and must go to output port 2 (010 in binary). The first microswitch (A-2) routes the packet based on the first bit (0), the second microswitch (B-2) routes the packet based on the second bit (l), and the third microswitch (C-2) routes the packet based on the third bit (0).



a. Input 1 sending a cell to output 6 (110)

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

Batcher-banyan switch

The problem with the banyan switch is the possibility of internal collision even when two packets are not heading for the same output port. We can solve this problem by sorting the arriving packets based on their destination port. K. E. Batcher designed a switch that comes before the banyan switch and sorts the incoming packets according to their final destinations. Normally, another hardware module called a trap is added between the Batcher switch and the banyan switch. The trap module prevents duplicate packets (packets with the same output destination) frompassing 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.