

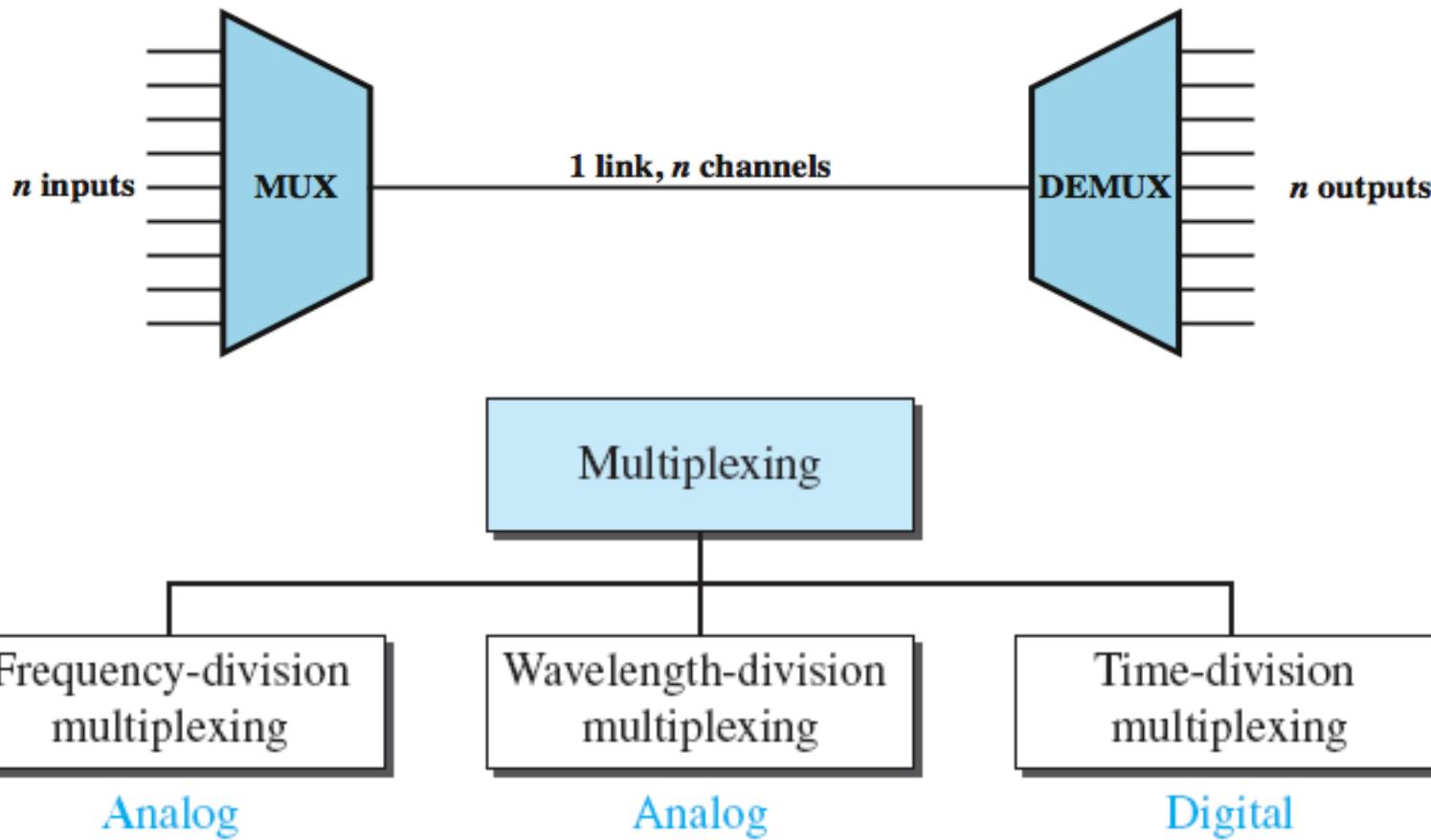
Module IV Multiplexing, Circuit Switching and Packet Switching

Multiplexing

- To make efficient use of high-speed telecommunications lines, some form of multiplexing is used. Multiplexing allows several transmission sources to share a larger transmission capacity. A common application of multiplexing is in long-haul communications. Trunks on long-haul networks are high-capacity fiber, coaxial, or microwave links. These links can carry large numbers of voice and data transmissions simultaneously using multiplexing. Common forms of multiplexing are frequency division multiplexing (FDM), time division multiplexing (TDM), and statistical TDM (STDM).
- There are n inputs to a multiplexer. The multiplexer is connected by a single data link to a demultiplexer. The link is able to carry n separate channels of data. The multiplexer combines data from the n input lines and transmits over a higher-capacity data link. The demultiplexer accepts the multiplexed data stream, separates (demultiplexes) the data according to channel, and delivers data to the appropriate output lines.

Multiplexing

- multiple links on 1 physical line
- common on long-haul, high capacity, links
- have FDM, WDM and TDM.



Frequency-division multiplexing (FDM): is an analog technique that can be applied when the bandwidth of a link (in hertz) is greater than the combined bandwidths of the signals to be transmitted.

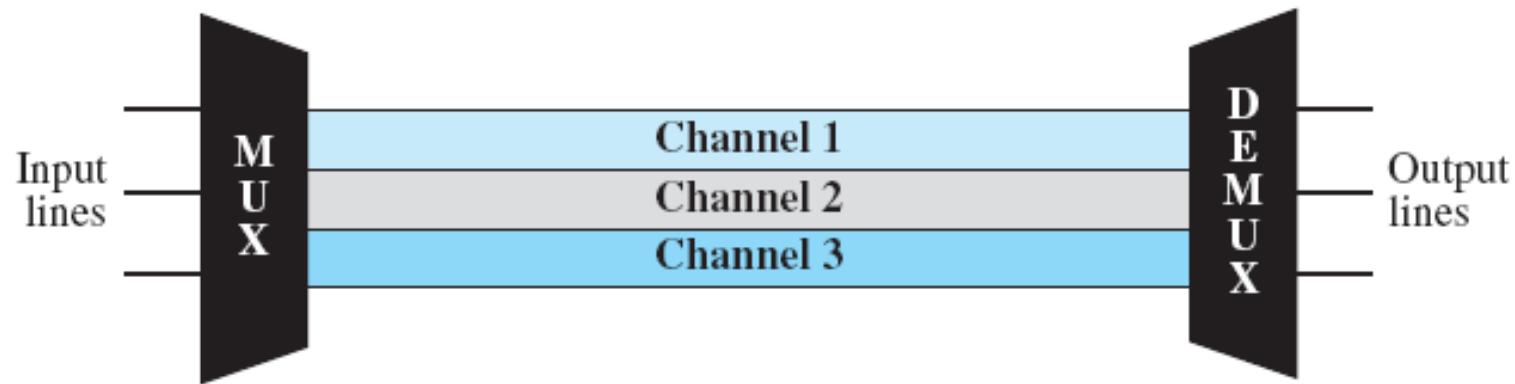
In FDM, signals generated by each sending device modulate different carrier frequencies. These modulated signals are then combined into a single composite signal that can be transported by the link. Carrier frequencies are separated by sufficient bandwidth to accommodate the modulated signal. These bandwidth ranges are the channels through which the various signals travel.

Channels can be separated by strips of unused bandwidth—**guard bands**—to prevent signals from overlapping.

In addition, carrier frequencies must not interfere with the original data frequencies.

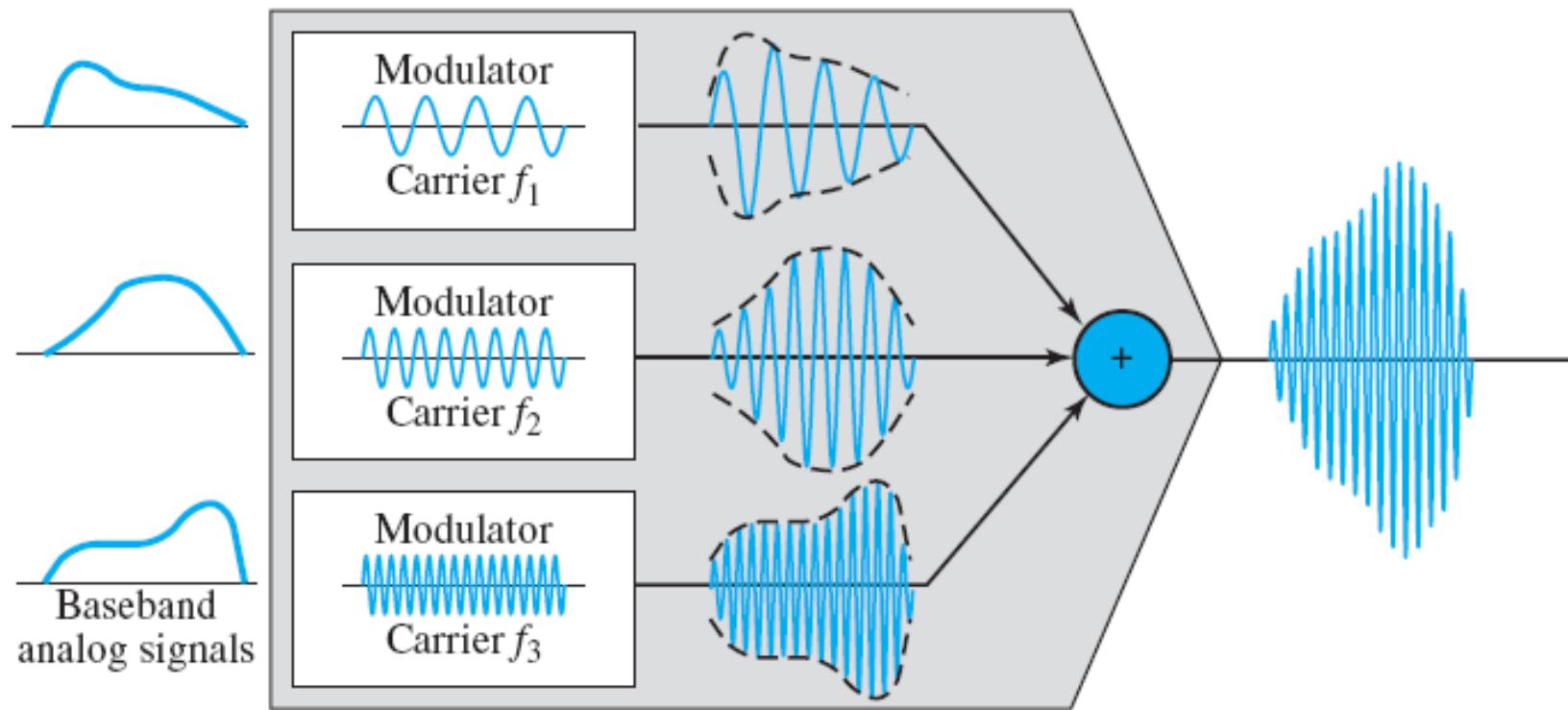
Figure 6.3 gives a conceptual view of FDM. In this illustration, the transmission path is divided into three parts, each representing a channel that carries one transmission.

Figure 6.3 Frequency-division multiplexing



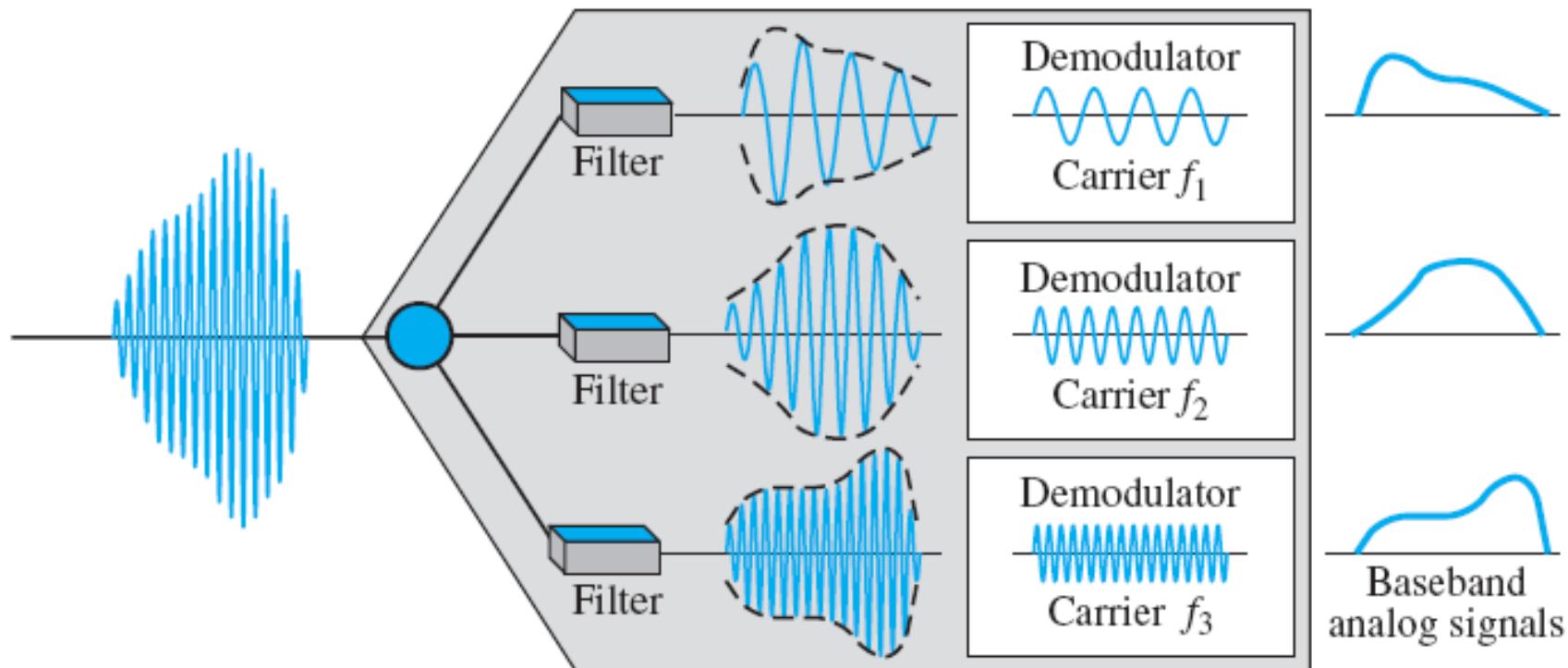
Multiplexing Process: Figure shows a conceptual illustration of the multiplexing process. Each source generates a signal of a similar frequency range. Inside the multiplexer, these similar signals modulate different carrier frequencies (f_1 , f_2 , and f_3). *The resulting modulated signals are then combined into a single composite signal that is sent out over a media link that has enough bandwidth to accommodate it.*

Figure 6.4 FDM process

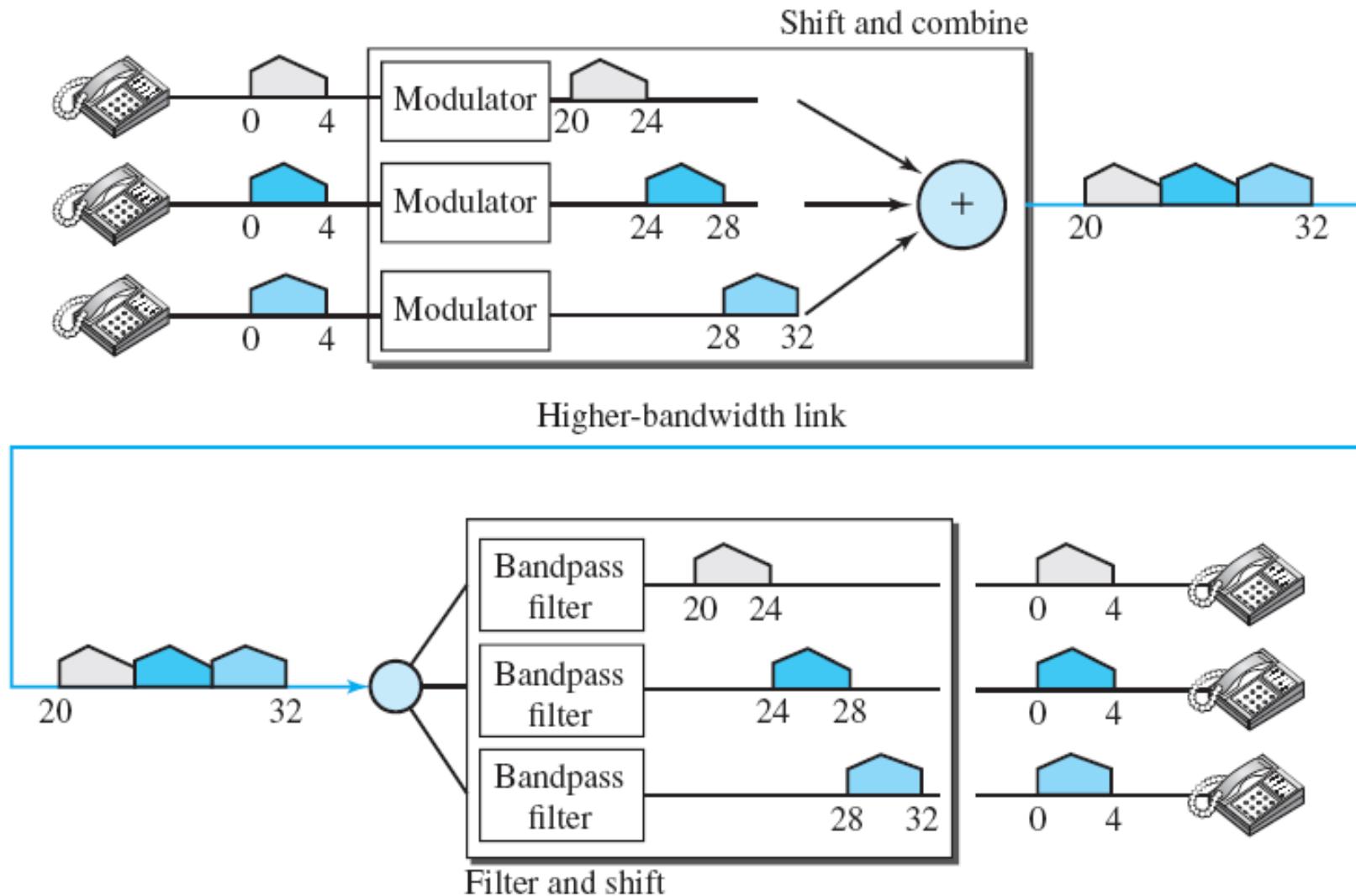


Demultiplexing Process: The demultiplexer uses a series of filters to decompose the multiplexed signal into its constituent component signals. The individual signals are then passed to a demodulator that separates them from their carriers and passes them to the output lines. Figure shows a conceptual illustration of demultiplexing process.

Figure 6.5 FDM demultiplexing example



Example 6.1: Assume that a voice channel occupies a bandwidth of 4 kHz. We need to combine three voice channels into a link with a bandwidth of 12 kHz, from 20 to 32 kHz. Show the configuration, using the frequency domain. Assume there are no guard bands.

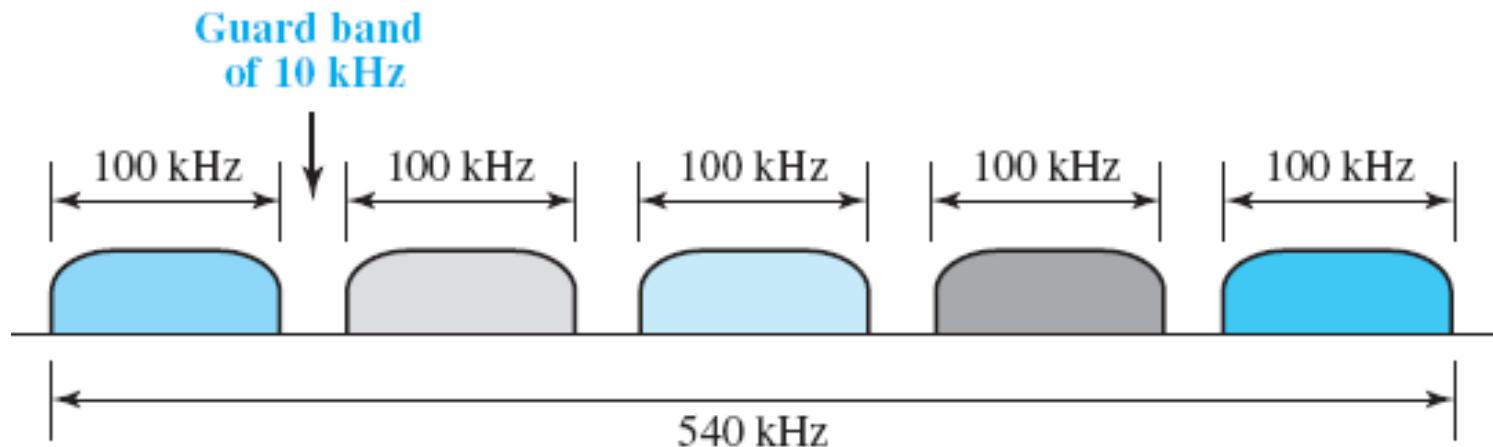


Example 6.2

Five channels, each with a 100-kHz bandwidth, are to be multiplexed together. What is the minimum bandwidth of the link if there is a need for a guard band of 10 kHz between the channels to prevent interference?

Solution

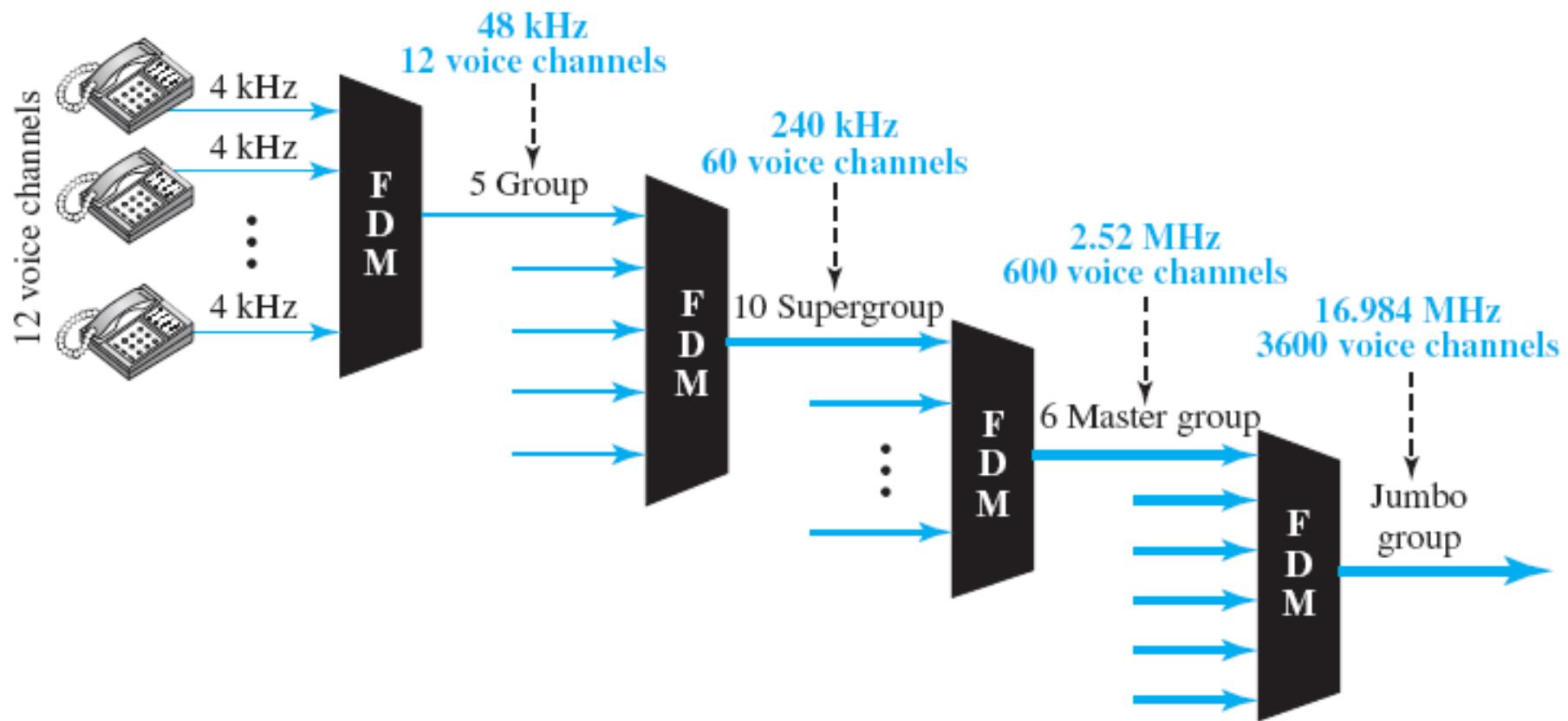
For five channels, we need at least four guard bands. This means that the required bandwidth is at least $5 \times 100 + 4 \times 10 = 540$ kHz, as shown in Figure 6.7.



Analog Hierarchy

- long-distance links use an FDM hierarchy
- AT&T (USA) and ITU-T (International) variants
- Group
 - 12 voice channels (4kHz each) = 48kHz
 - in range 60kHz to 108kHz
- Supergroup
 - FDM of 5 group signals supports 60 channels
 - on carriers between 420kHz and 612 kHz
- Mastergroup
 - FDM of 10 supergroups supports 600 channels
- so original signal can be modulated many times

Figure 6.9 Analog hierarchy



Other Applications of FDM: Common application of FDM is AM and FM radio broadcasting. Radio uses the air as the transmission medium. A special band from 530 to 1700 kHz is assigned to AM radio. All radio stations need to share this band.

Each station uses a different carrier frequency, which means it is shifting its signal and multiplexing. The signal that goes to the air is a combination of signals. A receiver receives all these signals, but filters (by tuning) only the one which is desired.

Without multiplexing, only one AM station could broadcast to the common link, the air. However, we need to know that there is no physical multiplexer or demultiplexer here. As we will see in Chapter 12, multiplexing is done at the data-link layer.

The situation is similar in FM broadcasting. However, FM has a wider band of 88 to 108 MHz because each station needs a bandwidth of 200 kHz.

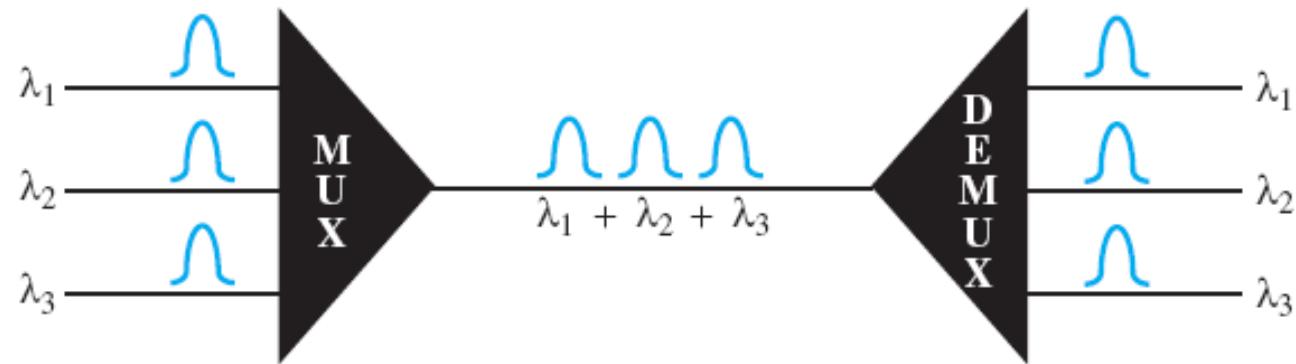
Another common use of FDM is in television broadcasting. Each TV channel has its own bandwidth of 6 MHz.

Wavelength Division Multiplexing

Wavelength-division multiplexing (WDM) is designed to use the high-data-rate capability of fiber-optic cable. The optical fiber data rate is higher than the data rate of metallic transmission cable, but using a fiber-optic cable for a single line wastes the available bandwidth. Multiplexing allows us to combine several lines into one.

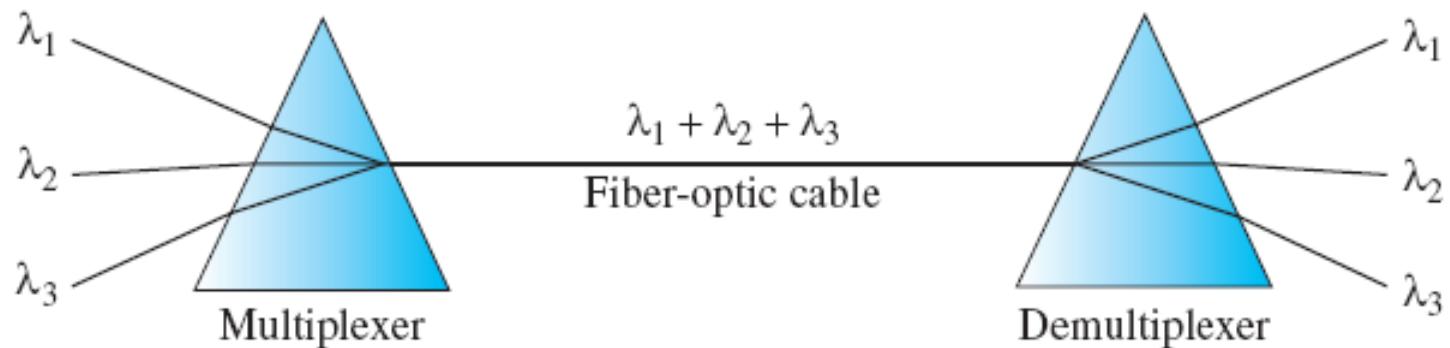
WDM is conceptually the same as FDM, except that the multiplexing and demultiplexing involve optical signals transmitted through fiber-optic channels. The idea is the same: We are combining different signals of different frequencies. The difference is that the frequencies are very high.

Figure 6.10 Wavelength-division multiplexing



WDM is an analog multiplexing technique to combine optical signals.

Figure 6.11 Prisms in wavelength-division multiplexing and demultiplexing



Although WDM technology is very complex, the basic idea is very simple. We want to combine multiple light sources into one single light at the multiplexer and do the reverse at the demultiplexer.

The combining and splitting of light sources are easily handled by a prism. Recall from basic physics that a prism bends a beam of light based on the angle of incidence and the frequency.

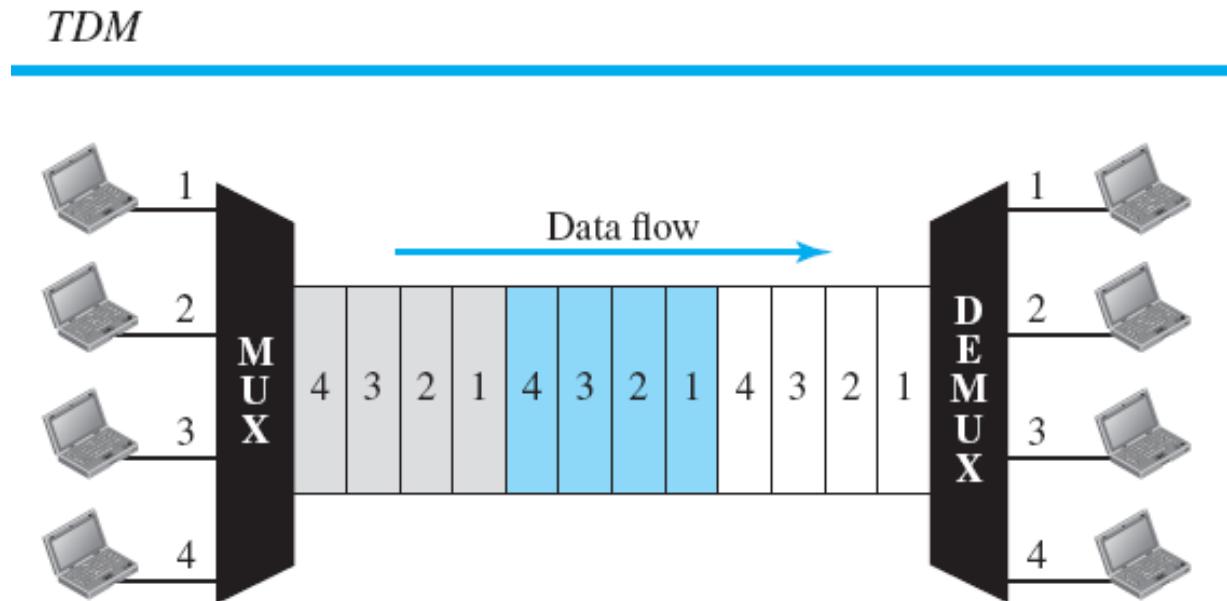
Using this technique, a multiplexer can be made to combine several input beams of light, each containing a narrow band of frequencies, into one output beam of a wider band of frequencies.

A demultiplexer can also be made to reverse the process. Figure 6.11 shows the concept.

Time Division Multiplexing

Time-division multiplexing (TDM) is a digital process that allows several connections to share the high bandwidth of a link. Each connection occupies a portion of time in the link.

Figure 6.12 gives a conceptual view of TDM. Note that the same link is used as in FDM; here, however, the link is shown sectioned by time rather than by frequency. In the figure, portions of signals 1, 2, 3, and 4 occupy the link sequentially.



We can divide TDM into two different schemes: synchronous and statistical. We first discuss **synchronous TDM** and then show how **statistical TDM differs**.

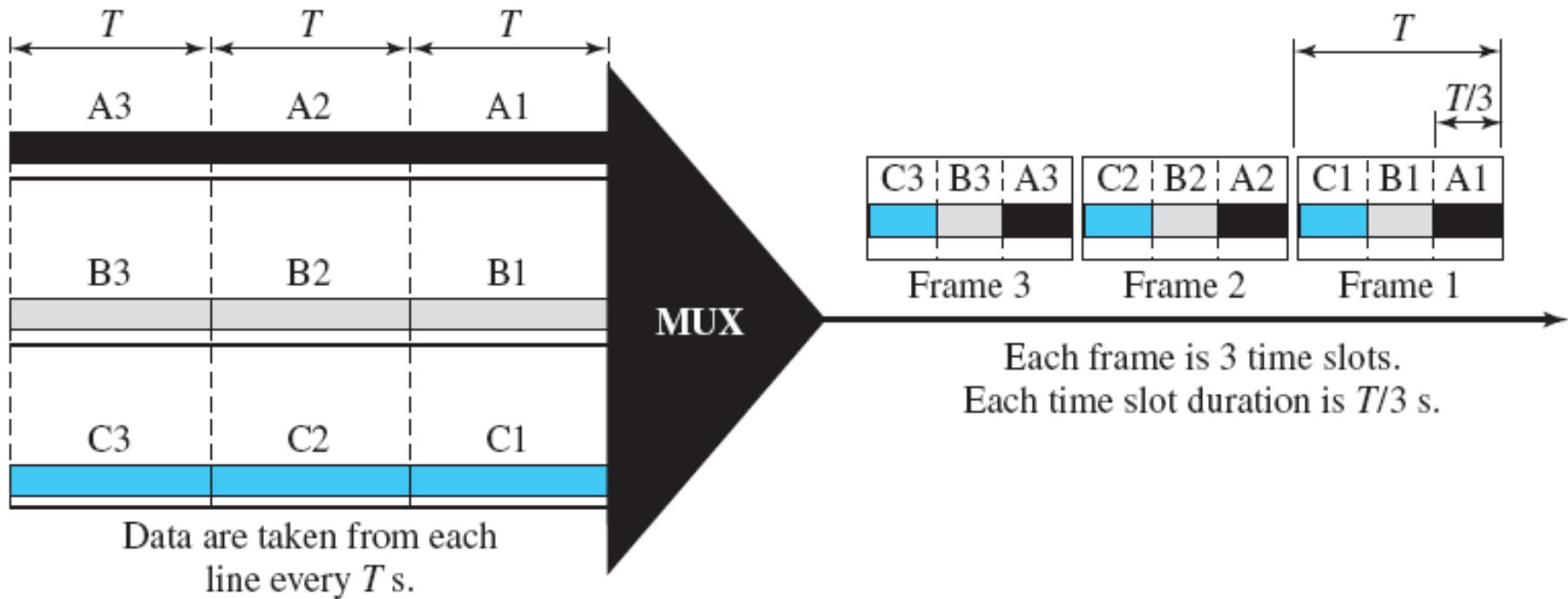
Synchronous TDM: In synchronous TDM, each input connection has an allotment in the output even if it is not sending data.

Time Slots and Frames

In synchronous TDM, the data flow of each input connection is divided into units, where each input occupies one input time slot. A unit can be 1 bit, one character, or one block of data. Each input unit becomes one output unit and occupies one output time slot. However, the duration of an output time slot is *n times shorter than the duration of* an input time slot.

If an input time slot is T sec, the output time slot is T/n sec, where n is the number of connections. In other words, a unit in the output connection has a shorter duration; it travels faster. Figure 6.13 shows an example of synchronous TDM where n is 3.

Figure 6.13 Synchronous time-division multiplexing



In synchronous TDM, a round of data units from each input connection is collected into a frame (we will see the reason for this shortly). If we have n connections, a frame is divided into n time slots and one slot is allocated for each unit, one for each input line. If the duration of the input unit is T , the duration of each slot is T/n and the duration of each frame is T .

The data rate of the output link must be *n times the data rate of a connection* to guarantee the flow of data. In Figure 6.13, the data rate of the link is 3 times the data rate of a connection; likewise, the duration of a unit on a connection is 3 times that of the time slot (duration of a unit on the link). In the figure we represent the data prior to multiplexing as 3 times the size of the data after multiplexing. This is just to convey the idea that each unit is 3 times longer in duration before multiplexing than after.

Example 6.5

In Figure 6.13, the data rate for each input connection is 1 kbps. If 1 bit at a time is multiplexed (a unit is 1 bit), what is the duration of

- 1. each input slot,**
- 2. each output slot, and**
- 3. each frame?**

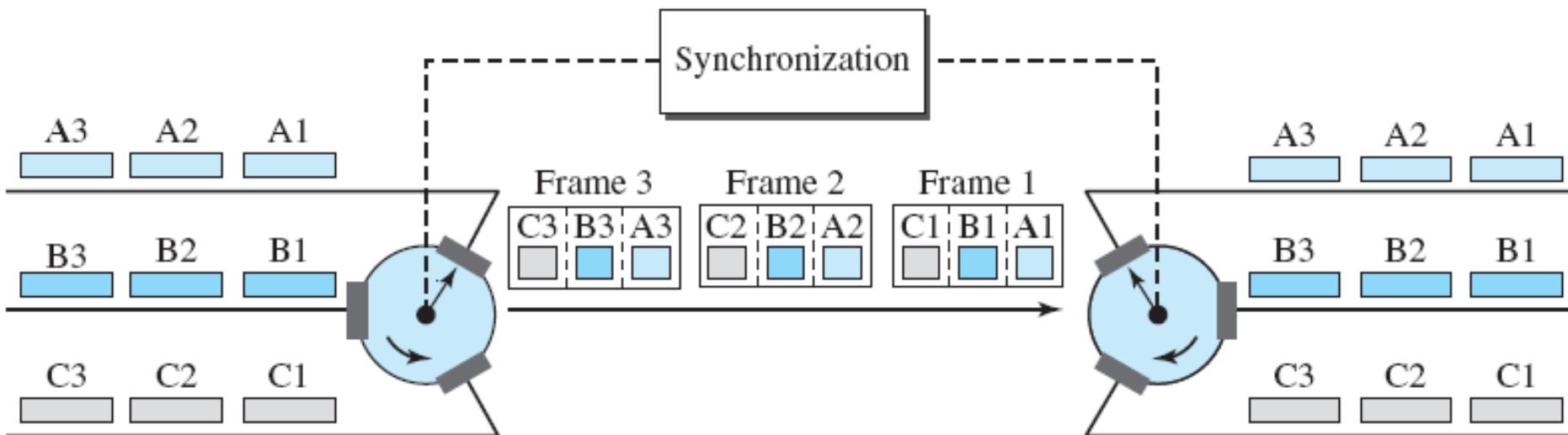
Solution

We can answer the questions as follows:

- 1. The data rate of each input connection is 1 kbps. This means that the bit duration is $1/1000$ s or 1 ms. The duration of the input time slot is 1 ms (same as bit duration).**
- 2. The duration of each output time slot is one-third of the input time slot. This means that the duration of the output time slot is $1/3$ ms.**
- 3. Each frame carries three output time slots. So the duration of a frame is $3 \times 1/3$ ms, or 1 ms. The duration of a frame is the same as the duration of an input unit.**

Interleaving: TDM can be visualized as two fast-rotating switches, one on the multiplexing side and the other on the demultiplexing side. The switches are synchronized and rotate at the same speed, but in opposite directions. On the multiplexing side, as the switch opens in front of a connection, that connection has the opportunity to send a unit onto the path. This process is called **interleaving**. On the **demultiplexing side**, as the switch opens in front of a connection, that connection has the opportunity to receive a unit from the path.

Figure 6.15 Interleaving

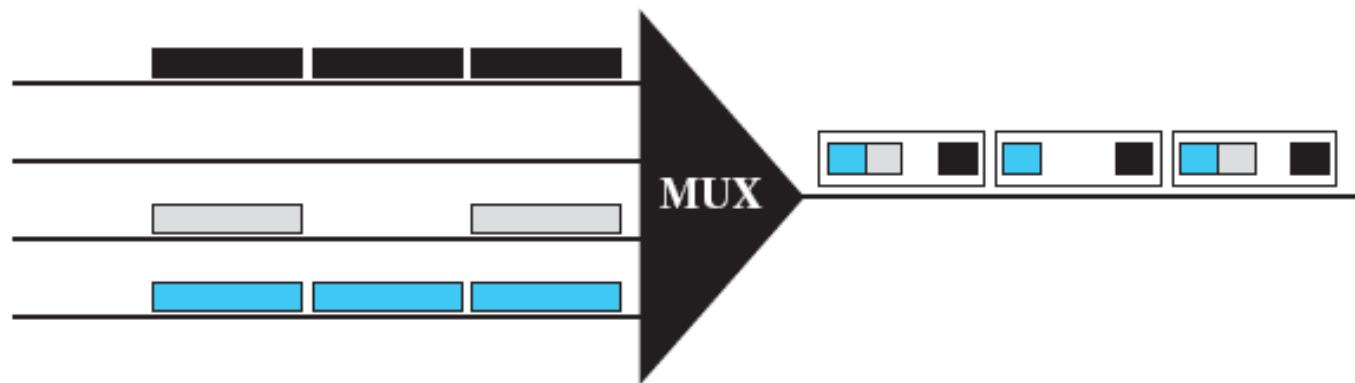


Empty Slots: Synchronous TDM is not as efficient as it could be. If a source does not have data to send, the corresponding slot in the output frame is empty. Figure 6.18 shows a case in which one of the input lines has no data to send and one slot in another input line has discontinuous data.

The first output frame has three slots filled, the second frame has two slots filled, and the third frame has three slots filled. No frame is full.

We learn in the next section that statistical TDM can improve the efficiency by removing the empty slots from the frame.

Figure 6.18 *Empty slots*



Statistical TDM

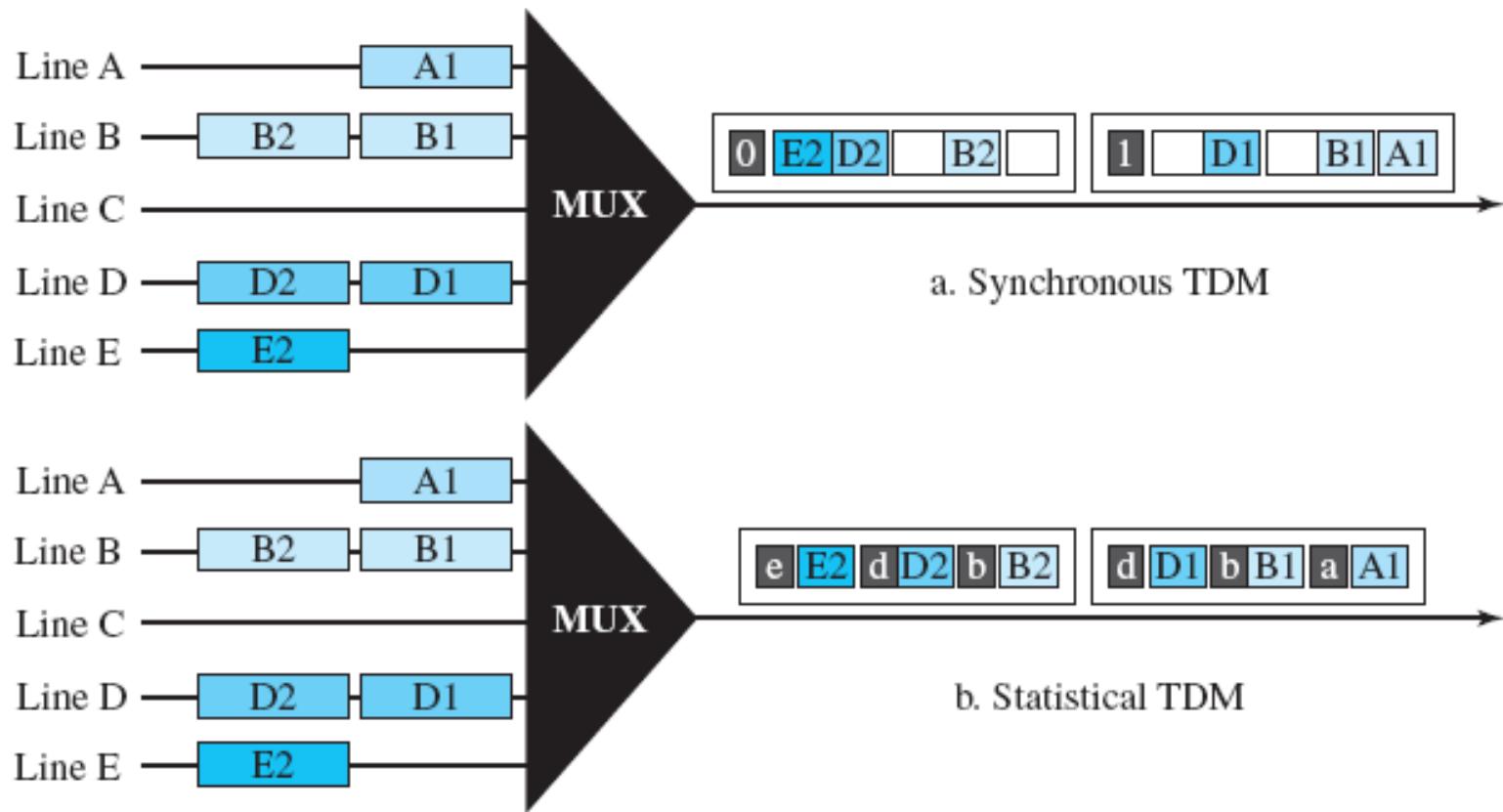
Statistical Time-Division Multiplexing

As we saw in the previous section, in synchronous TDM, each input has a reserved slot in the output frame. This can be inefficient if some input lines have no data to send.

In statistical time-division multiplexing, slots are dynamically allocated to improve bandwidth efficiency. Only when an input line has a slot's worth of data to send is it given a slot in the output frame. In statistical multiplexing, the number of slots in each frame is less than the number of input lines. The multiplexer checks each input line in roundrobin fashion; it allocates a slot for an input line if the line has data to send; otherwise, it skips the line and checks the next line.

Figure 6.26 shows a synchronous and a statistical TDM example. In the former, some slots are empty because the corresponding line does not have data to send. In the latter, however, no slot is left empty as long as there are data to be sent by any input line.

Figure 6.26 *TDM slot comparison*



Statistical TDM

- in Synch TDM many slots are wasted
- Statistical TDM allocates time slots dynamically based on demand
- multiplexer scans input lines and collects data until frame full
- line data rate lower than aggregate input line rates
- may have problems during peak periods
 - must buffer inputs

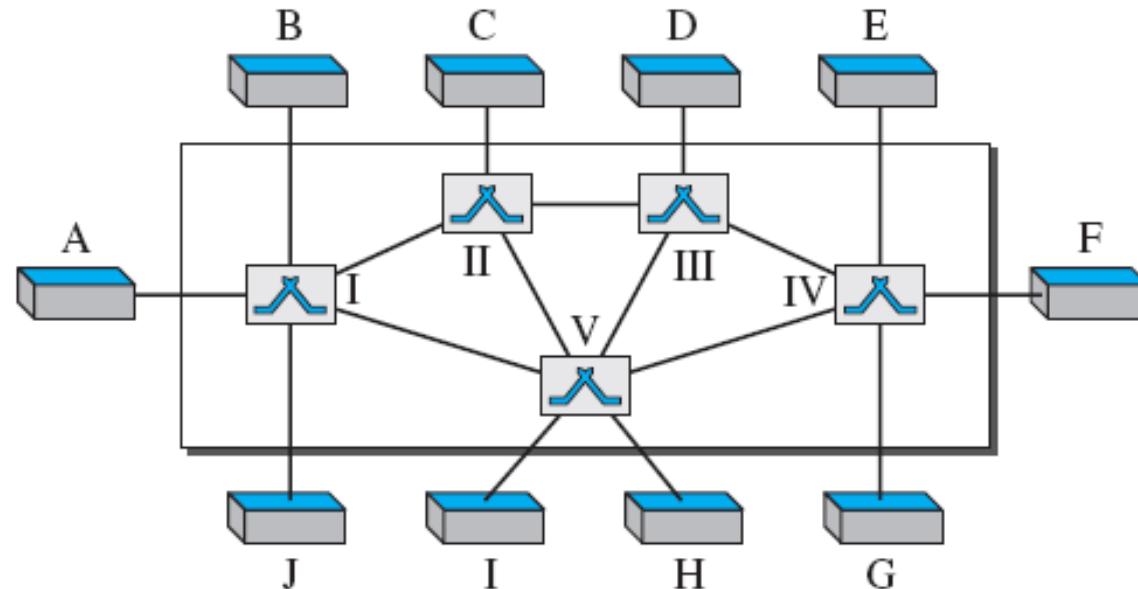
Why switching is required? A network is a set of connected devices. Whenever we have multiple devices, we have the problem of how to connect them to make one-to-one communication possible.

One solution is to make a point-to-point connection between each pair of devices (a mesh topology) or between a central device and every other device (a star topology). These methods, however, are impractical and wasteful when applied to very large networks.

The number and length of the links require too much infrastructure to be cost-efficient, and the majority of those links would be idle most of the time. Other topologies employing multipoint connections, such as a bus, are ruled out because the distances between devices and the total number of devices increase beyond the capacities of the media and equipment.

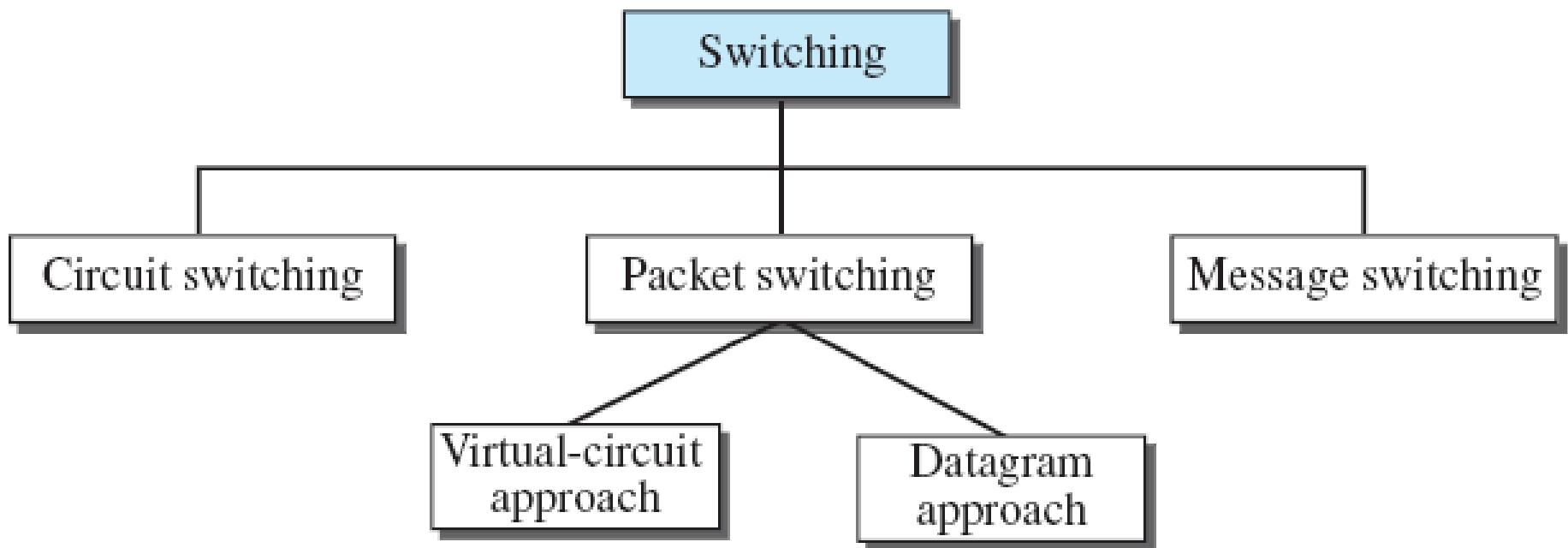
A better solution is **switching**. A switched network consists of a series of interlinked nodes, called *switches*. *Switches are devices capable of creating temporary connections* between two or more devices linked to the switch.

Figure 8.1 Switched network



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.

Methods of Switching



Switching can happen at several layers of the TCP/IP protocol suite.

Switching at Physical Layer

At the physical layer, we can have only circuit switching. There are no packets exchanged at the physical layer. The switches at the physical layer allow signals to travel in one path or another.

Switching at Data-Link Layer

At the data-link layer, we can have packet switching. However, the term *packet* in this case means *frames* or *cells*. Packet switching at the data-link layer is normally done using a virtual-circuit approach.

Switching at Network Layer

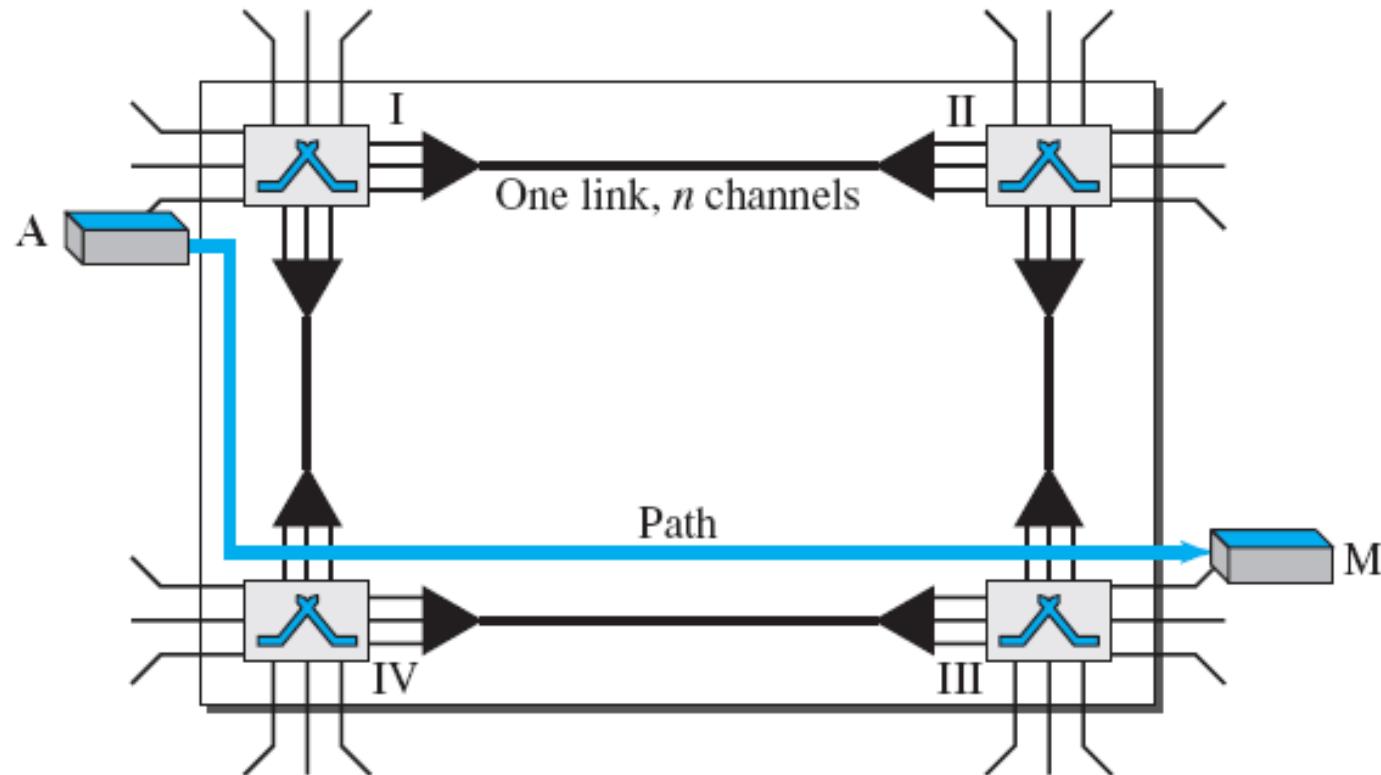
At the network layer, we can have packet switching. In this case, either a virtual-circuit approach or a datagram approach can be used. Currently the Internet uses a datagram approach, as we see in Chapter 18, but the tendency is to move to a virtual-circuit approach.

Switching at Application Layer

At the application layer, we can have only message switching. The communication at the application layer occurs by exchanging messages. Conceptually, we can say that communication using e-mail is a kind of message-switched communication, but we do not see any network that actually can be called a message-switched network.

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, as discussed previously.

Figure 8.3 A trivial circuit-switched network



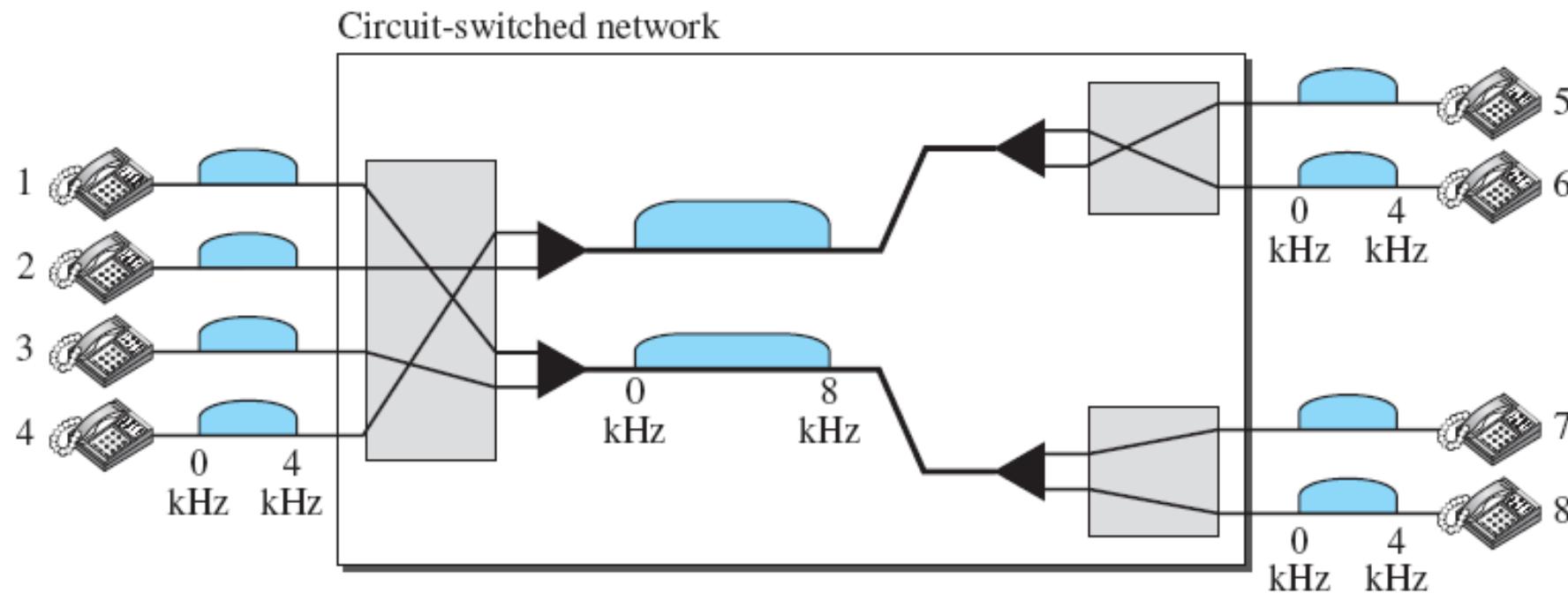
We need to emphasize several points here:

- ❑ 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.

Figure 8.4 Circuit-switched network used in Example 8.1



Three Phases: The actual communication in a circuit-switched network requires three phases:
Connection setup,
data transfer, and
connection teardown.

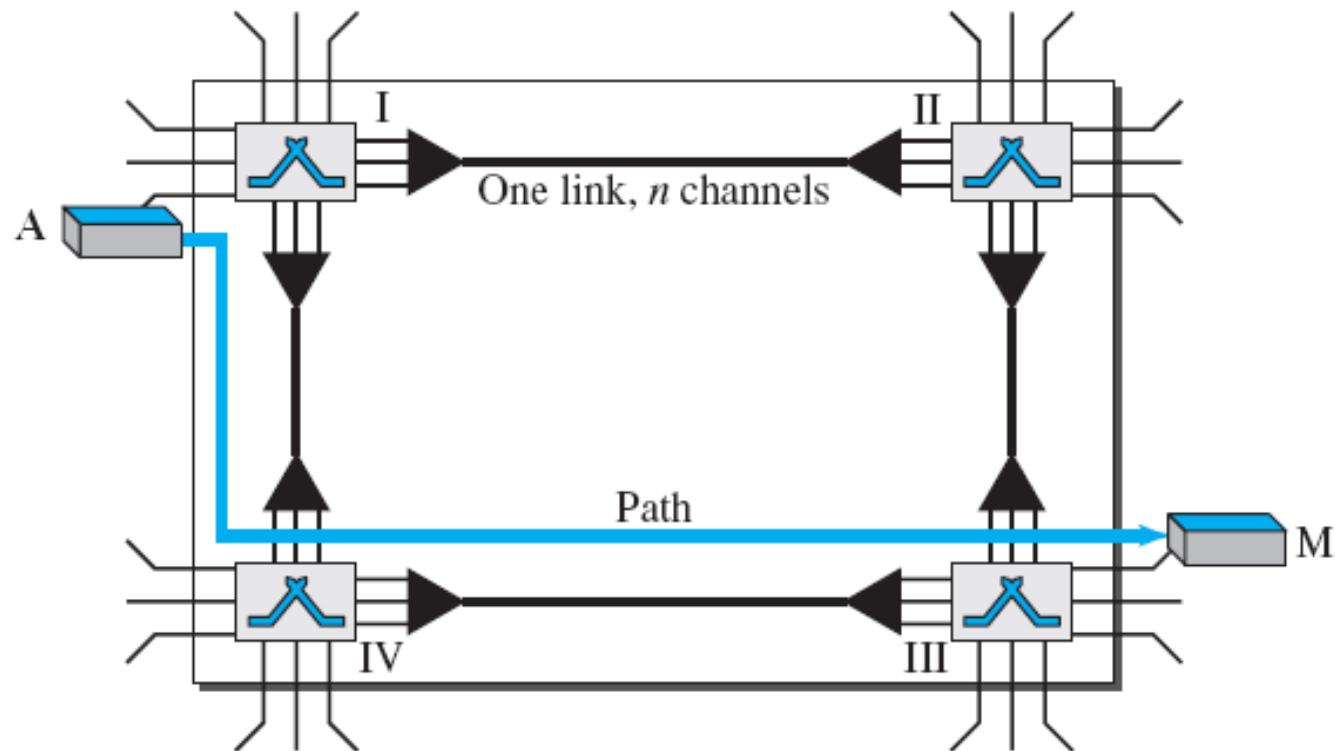
Setup Phase

Before the two parties (or multiple parties in a conference call) 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.

For example, in Figure 8.3, when system A needs to connect to system M, it sends a setup request that includes the address of system M, to switch I. Switch I finds a channel between itself and switch IV that can be dedicated for this purpose.

Switch I then sends the request to switch IV, which finds a dedicated channel between itself and switch III. Switch III informs system M of system A's intention at this time.

In the next step to making a connection, an acknowledgment from system M needs to be sent in the opposite direction to system A. Only after system A receives this acknowledgment is the connection established. Note that 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.

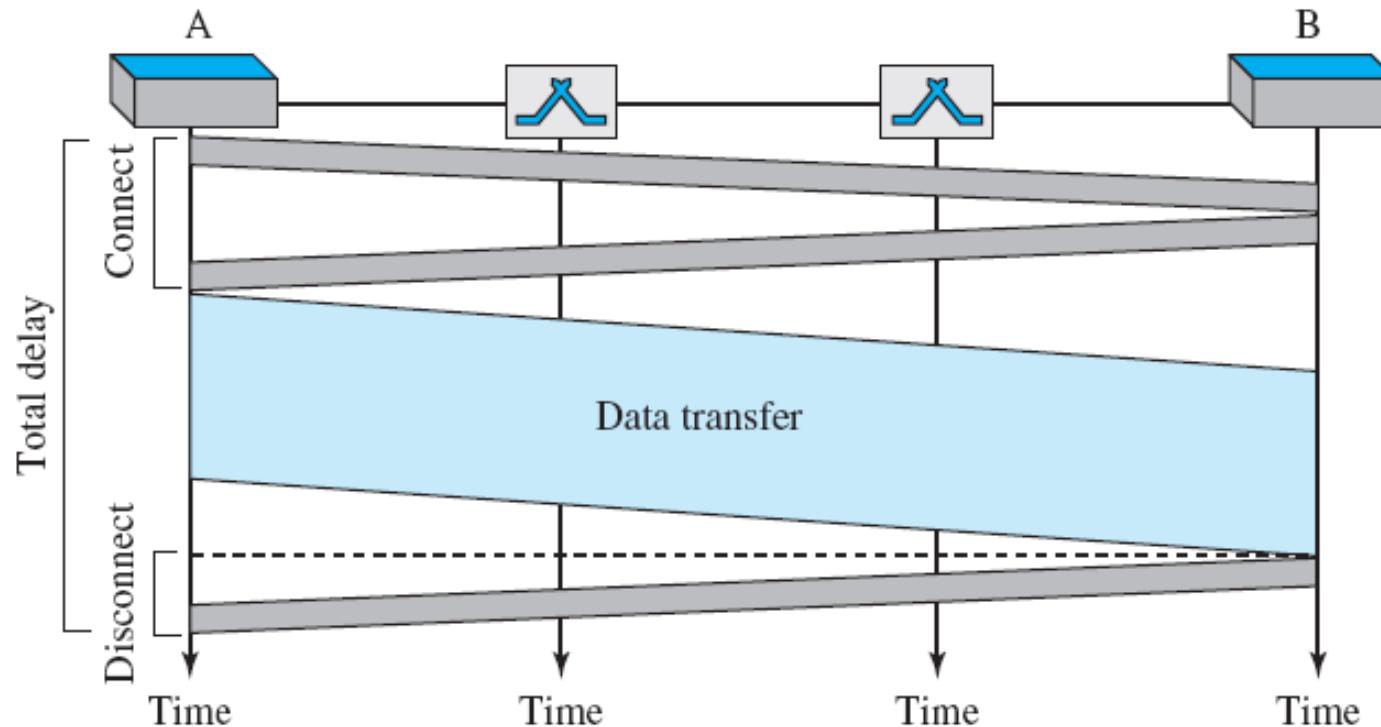
Teardown Phase

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

8.2.3 Delay

Although a circuit-switched network normally has low efficiency, the delay in the network is minimal. During data transfer the data are not delayed at each switch. Resources are allocated for the duration of the connection. Figure 8.6 shows the delay in a circuit-switched network when only two switches are involved.

Figure 8.6 Delay in a circuit-switched network



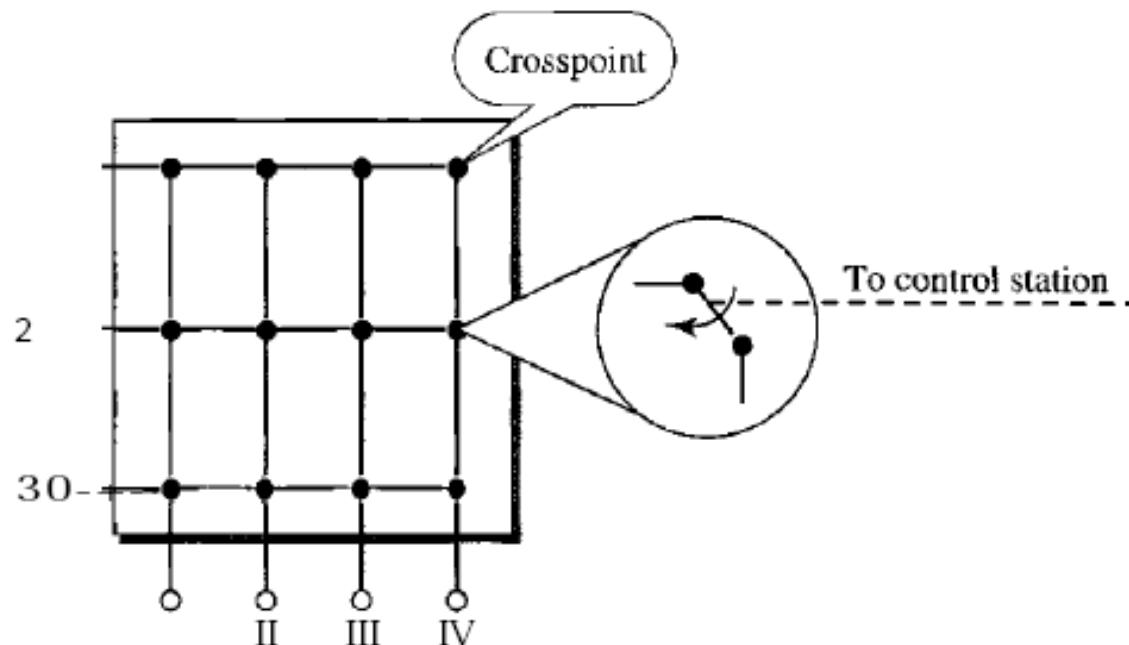
As Figure 8.6 shows, there is no waiting time at each switch. The total delay is due to the time needed to create the connection, transfer data, and disconnect the circuit. The delay caused by the setup is the sum of four parts: the propagation time of the source computer request (slope of the first 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.

Space Division Switching

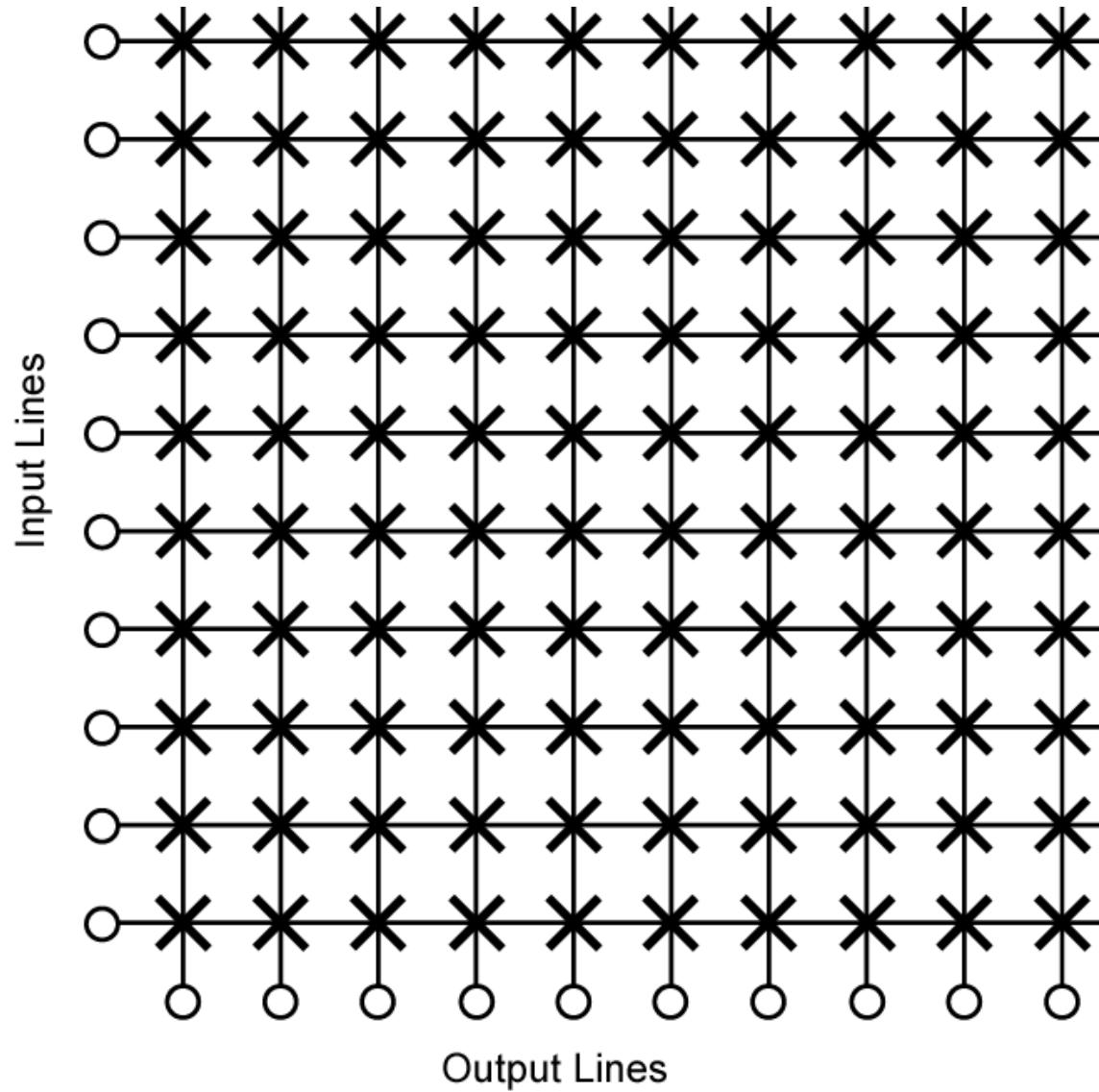
- Developed for analog environment
- Separate physical paths
- Crossbar switch
 - Number of crosspoints grows as square of number of stations
 - Loss of crosspoint prevents connection
 - Inefficient use of crosspoints
 - All stations connected, only a few crosspoints in use
 - Non-blocking

Crossbar Switch A crossbar switch connects n inputs to m outputs in a grid, using electronic microswitches (transistors) at each crosspoint (see Figure 8.17). The major limitation of this design is the number of crosspoints required. To connect n inputs to m outputs using a crossbar switch requires $n \times m$ 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.

Figure 8.17 *Crossbar switch with three inputs and four outputs*



Space Division Switch: Crossbar

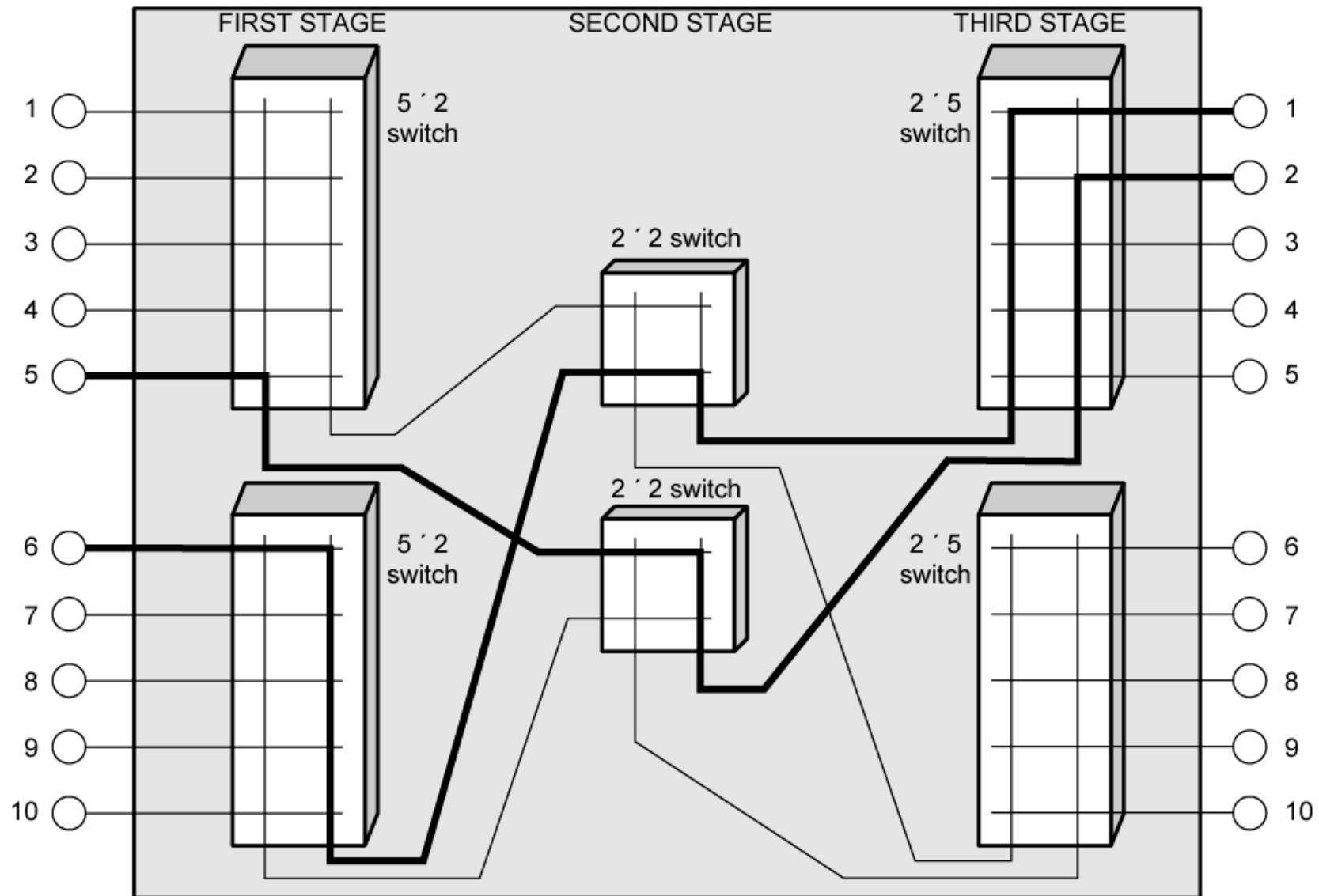


Multistage Switch

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

- Reduced number of cross points
- More than one path through network
 - Increased reliability
- More complex control
- May be blocking

Three Stage Space Division Switch



Time Division Switching

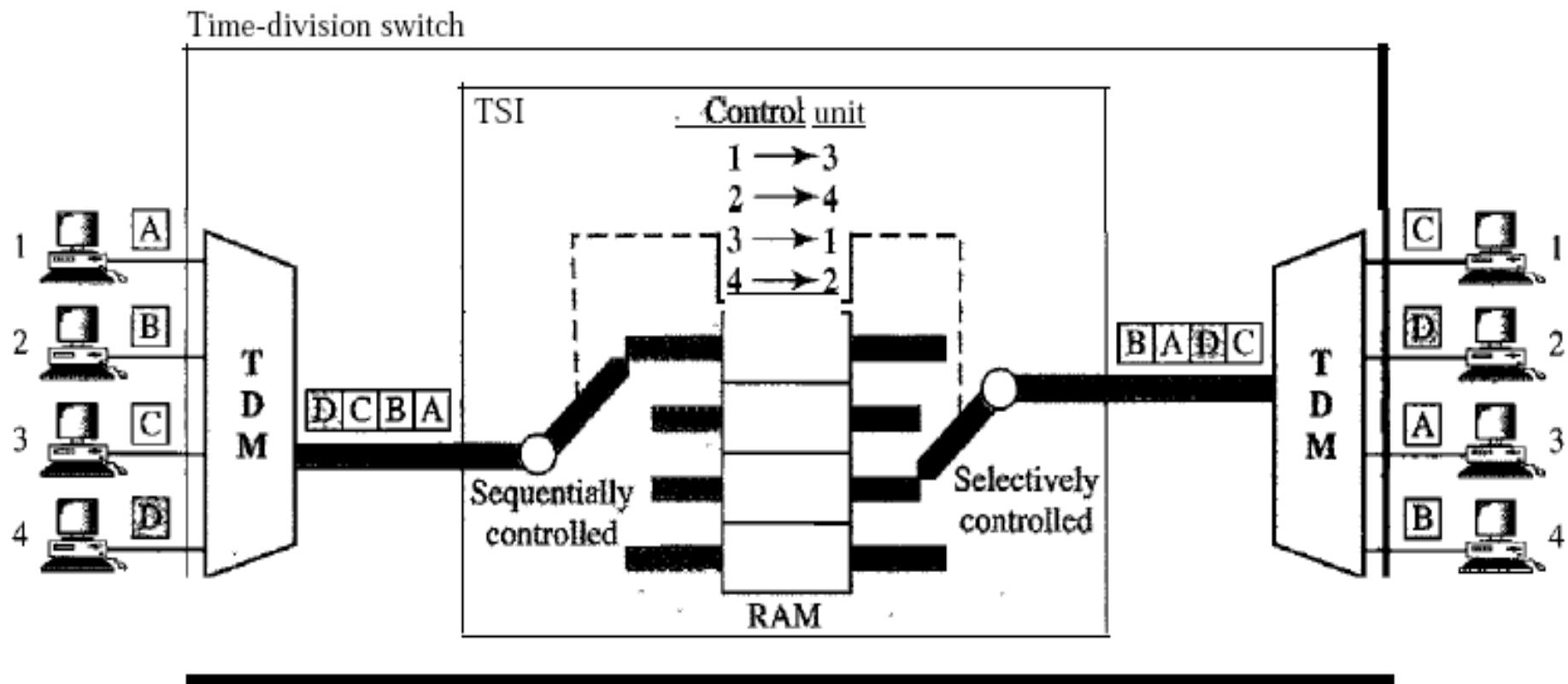
Time-Division Switch

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 Figure 8.19 shows a system connecting four input lines to four output lines. Imagine that each input line wants to send data to an output line according to the following pattern:

$$1 \rightarrow 3 \quad 2 \rightarrow 4 \quad 3 \rightarrow 1 \quad 4 \rightarrow 2$$

Figure 8.19 *Time-slot interchange*



The figure combines a TDM multiplexer, a TDM demultiplexer, and a TSI consisting of random access memory (RAM) with several memory locations. 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

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.

Figure 8.20 *Time-space-time switch*

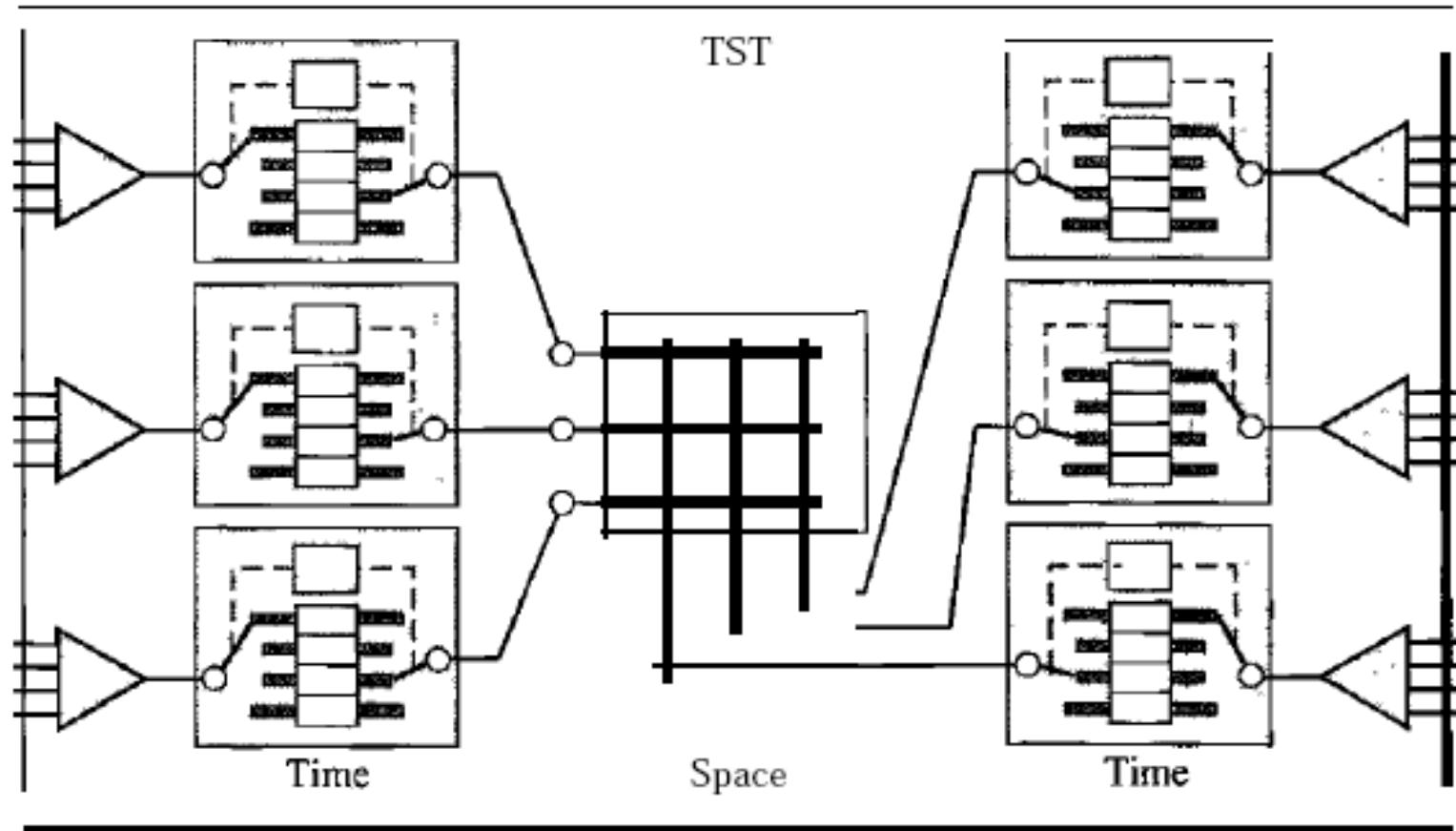


Figure 8.20 shows a simple TST switch that consists of two time stages and one space stage and has 12 inputs and 12 outputs. Instead of one time-division switch, it divides the inputs into three groups (of four inputs each) and directs them to three time-slot interchanges. The result is that the average delay is one-third of what would result from using one time-slot interchange to handle all 12 inputs.

Location of Signaling

- Subscriber to network
 - Depends on subscriber device and switch
- Within network
 - Management of subscriber calls and network
 - more complex

In Channel Signaling

- Use same channel for signaling and call
 - Requires no additional transmission facilities
- Inband
 - Uses same frequencies as voice signal
 - Can go anywhere a voice signal can
 - Impossible to set up a call on a faulty speech path
- Out of band
 - Voice signals do not use full 4kHz bandwidth
 - Narrow signal band within 4kHz used for control
 - Can be sent whether or not voice signals are present
 - Need extra electronics
 - Slower signal rate (narrow bandwidth)

Drawbacks of In Channel Signaling

- Limited transfer rate
- Delay between entering address (dialing) and connection
- Overcome by use of common channel signaling

Common Channel Signaling

- Control signals carried over paths independent of voice channel
- One control signal channel can carry signals for a number of subscriber channels
- Common control channel for these subscriber lines
- Associated Mode
 - Common channel closely tracks interswitch trunks
- Disassociated Mode
 - Additional nodes (signal transfer points)
 - Effectively two separate networks

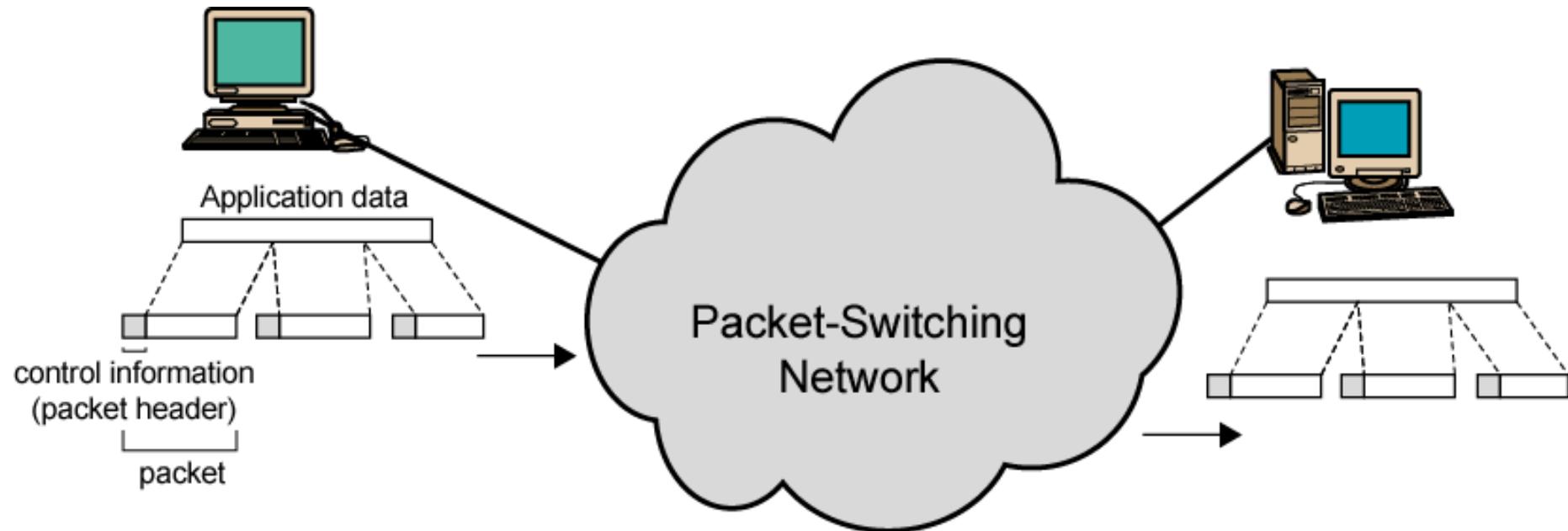
Packet Switching Principles

- Circuit switching designed for voice
 - Resources dedicated to a particular call
 - Much of the time a data connection is idle
 - Data rate is fixed
 - Both ends must operate at the same rate

Basic Operation

- Data transmitted in small packets
 - Typically 1000 octets
 - Longer messages split into series of packets
 - Each packet contains a portion of user data plus some control info
- Control info
 - Routing (addressing) info
- Packets are received, stored briefly (buffered) and passed on to the next node
 - Store and forward

Use of Packets



Advantages

- Line efficiency
 - Single node to node link can be shared by many packets over time
 - Packets queued and transmitted as fast as possible
- Data rate conversion
 - Each station connects to the local node at its own speed
 - Nodes buffer data if required to equalize rates
- Packets are accepted even when network is busy
 - Delivery may slow down
- Priorities can be used

Switching Technique

- Station breaks long message into packets
- Packets sent one at a time to the network
- Packets handled in two ways
 - Datagram
 - Virtual circuit

Datagram

- Each packet treated independently
- Packets can take any practical route
- Packets may arrive out of order
- Packets may go missing
- Up to receiver to re-order packets and recover from missing packets

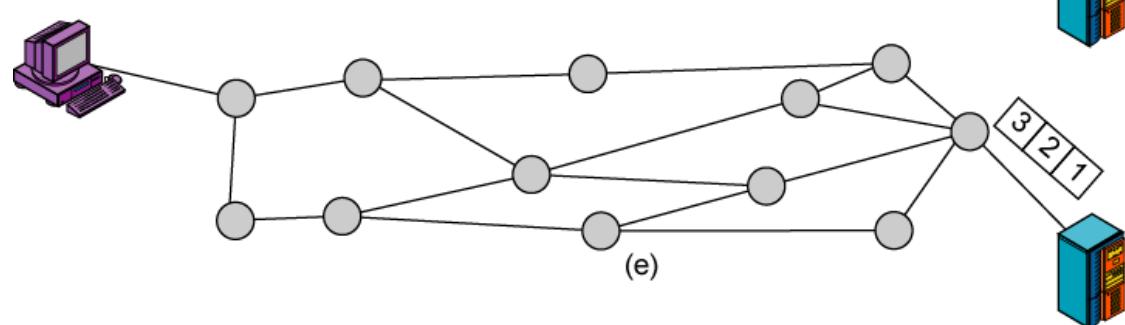
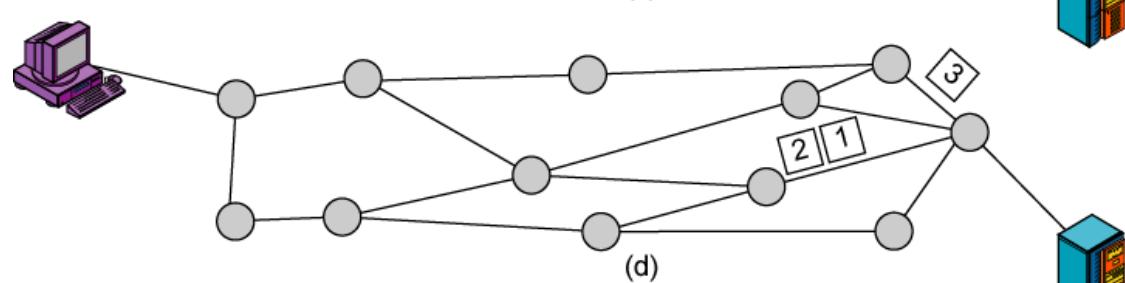
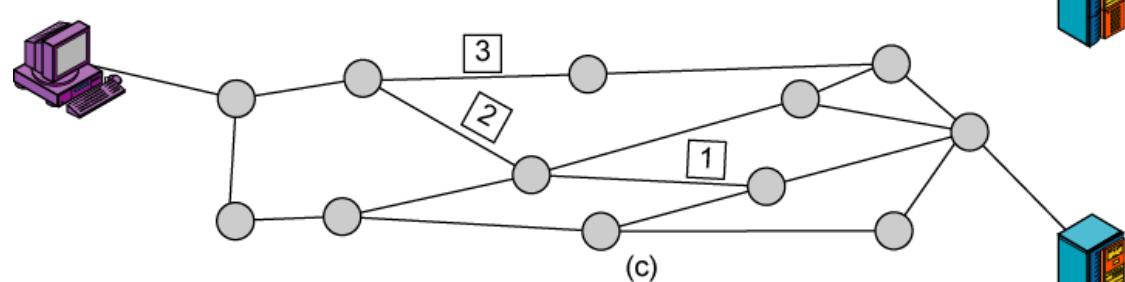
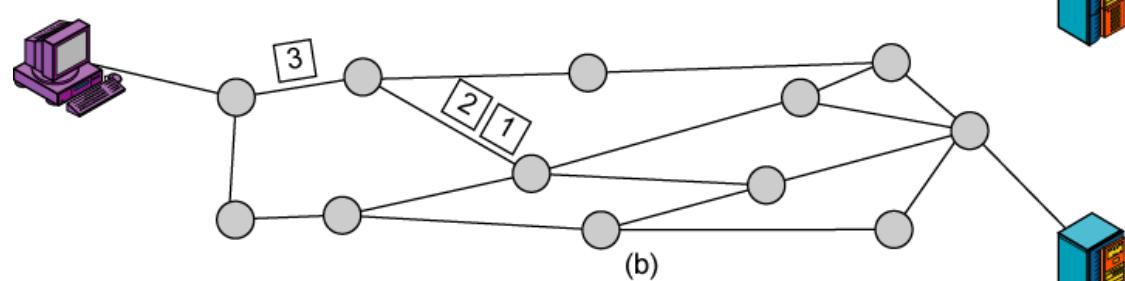
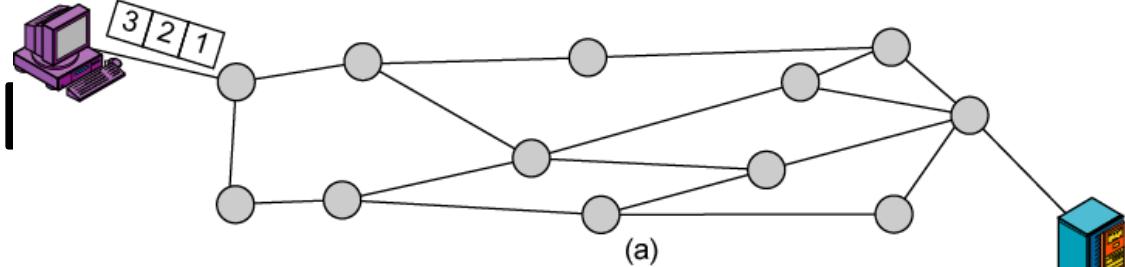
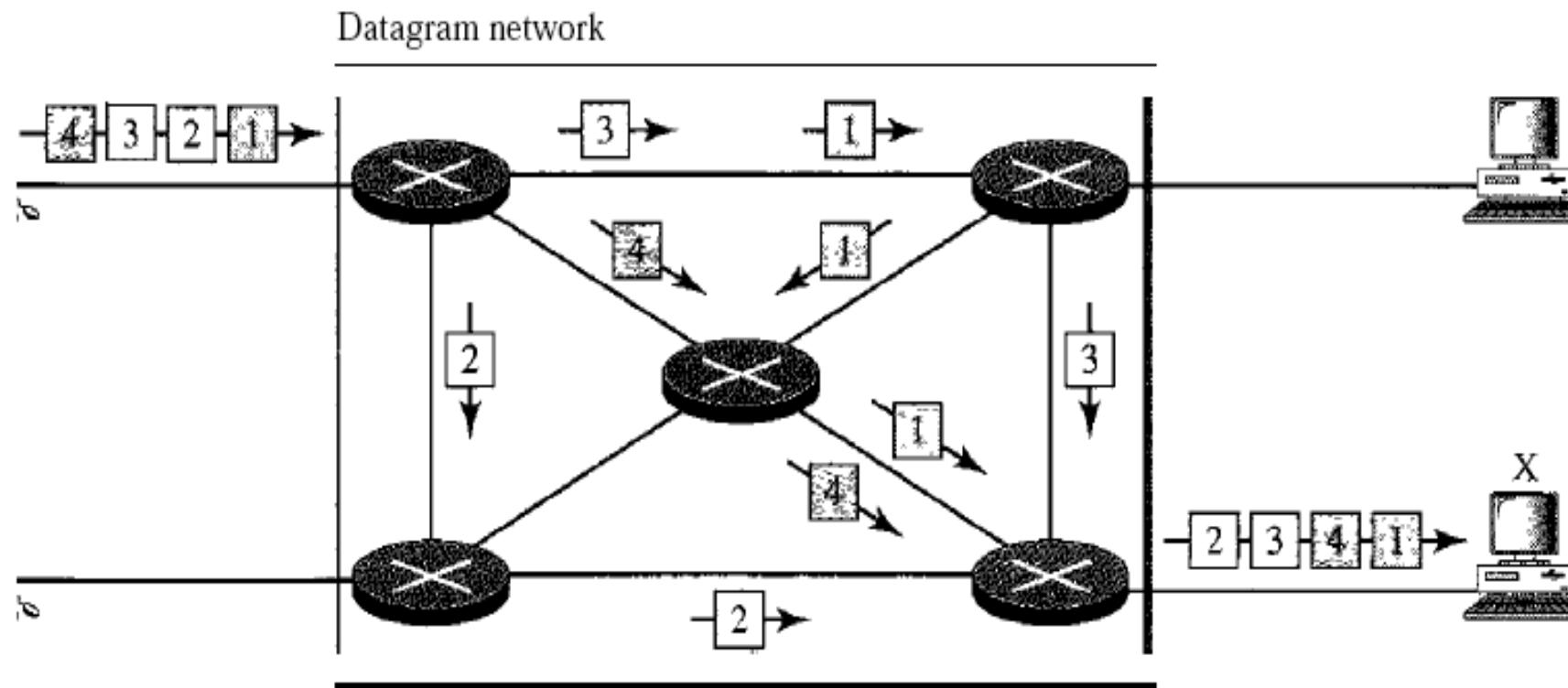


Figure 8.7 A datagram network with four switches (routers)



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.

Virtual Circuit

- Preplanned route established before any packets sent
- Call request and call accept packets establish connection (handshake)
- Each packet contains a virtual circuit identifier instead of destination address
- No routing decisions required for each packet
- Clear request to drop circuit
- Not a dedicated path

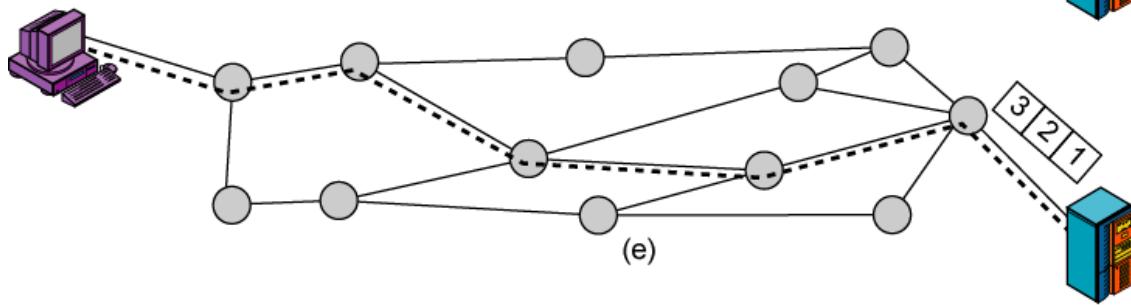
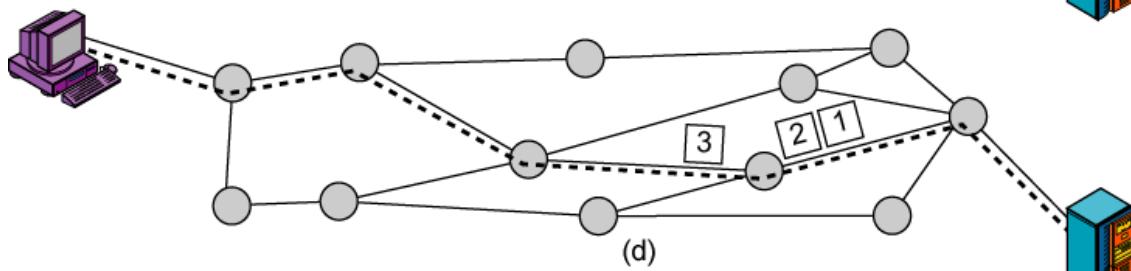
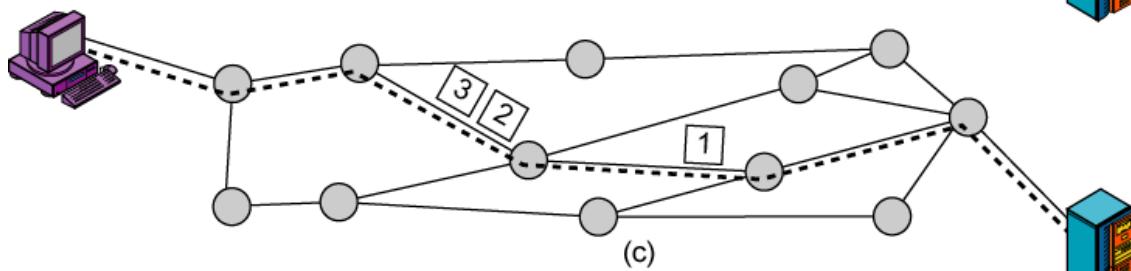
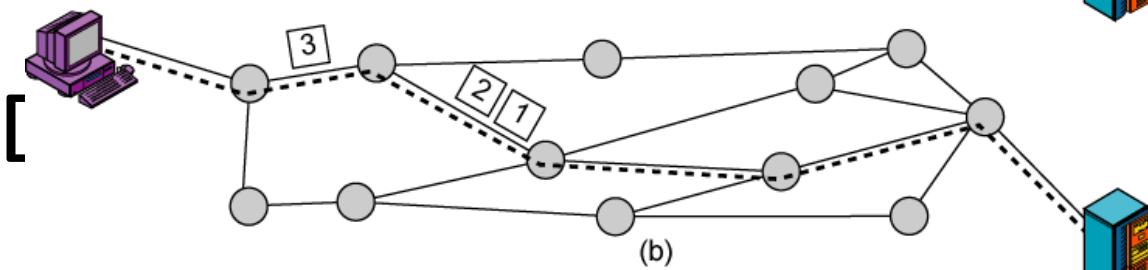
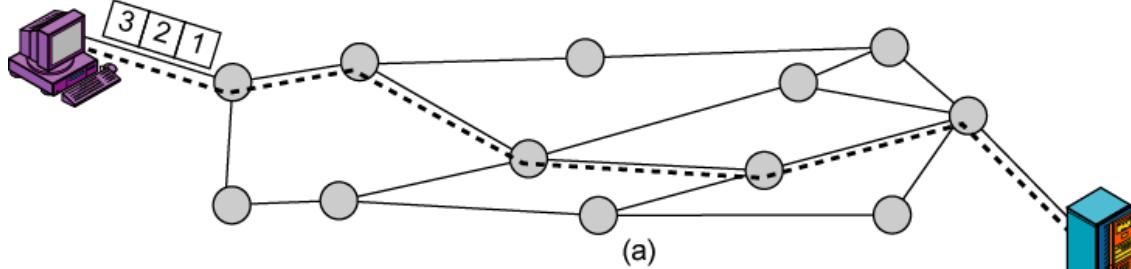


Figure 8.13 Source-to-destination data transfer in a virtual-circuit network

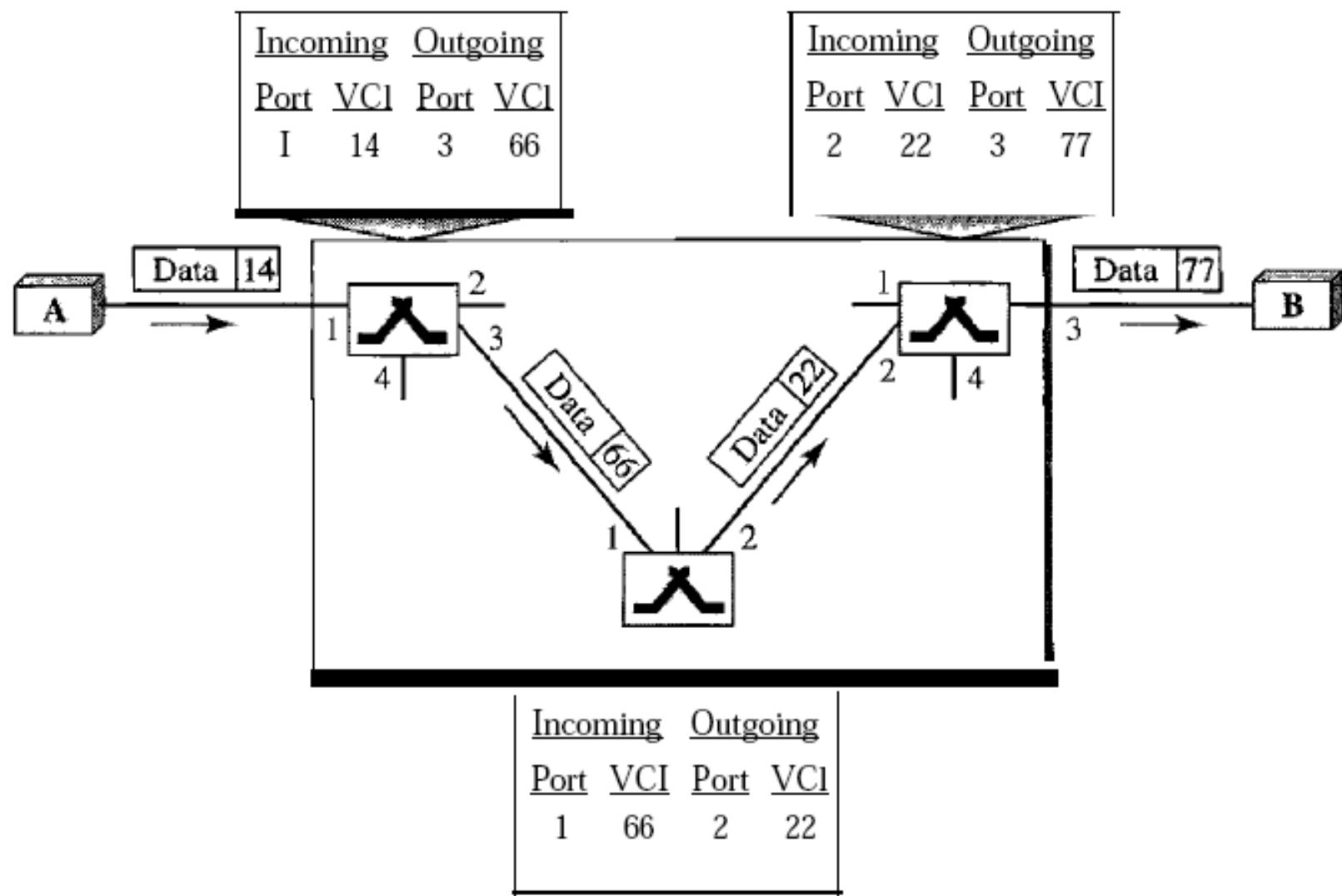


Figure 8.14 Setup request in a virtual-circuit network

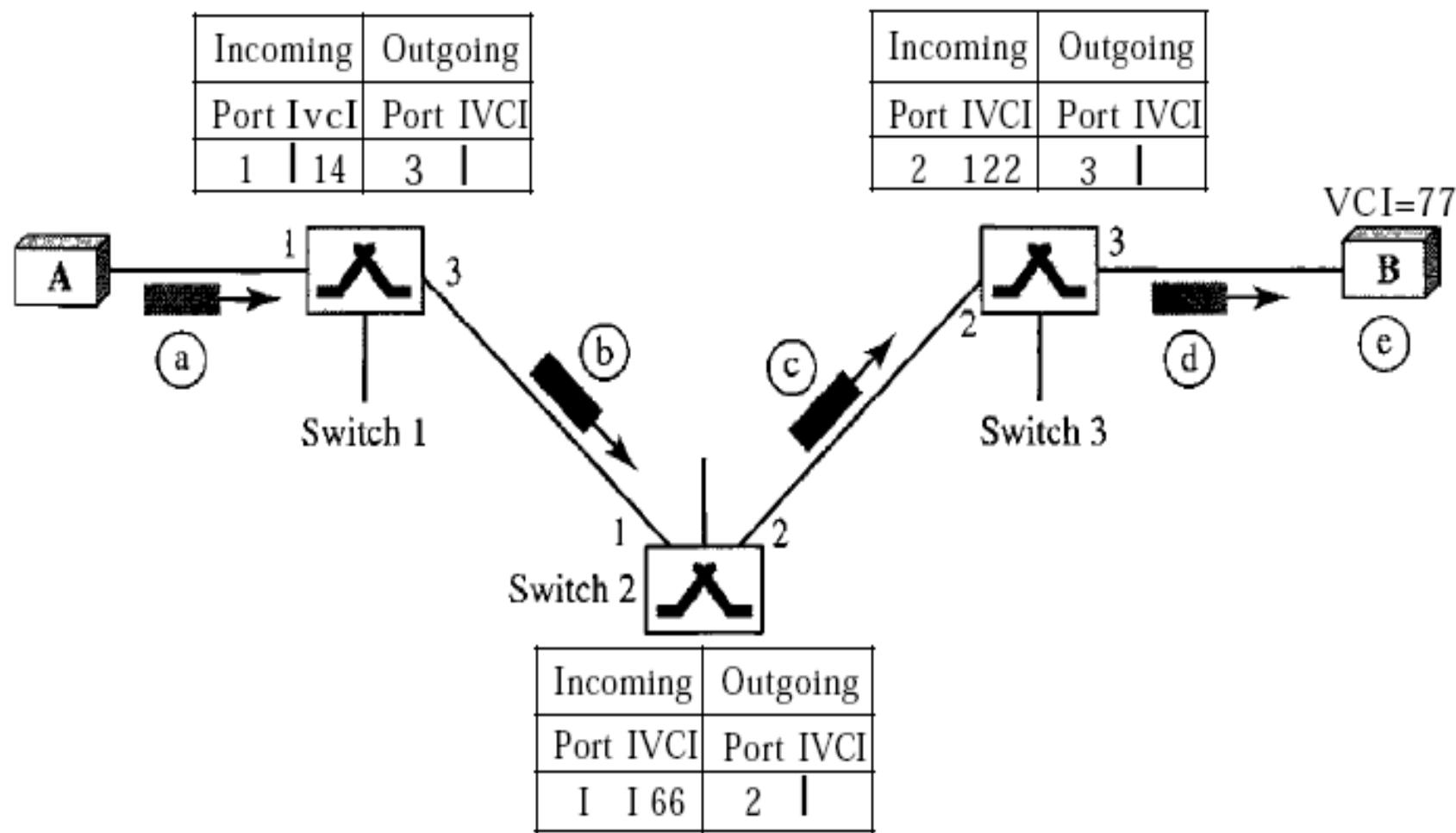
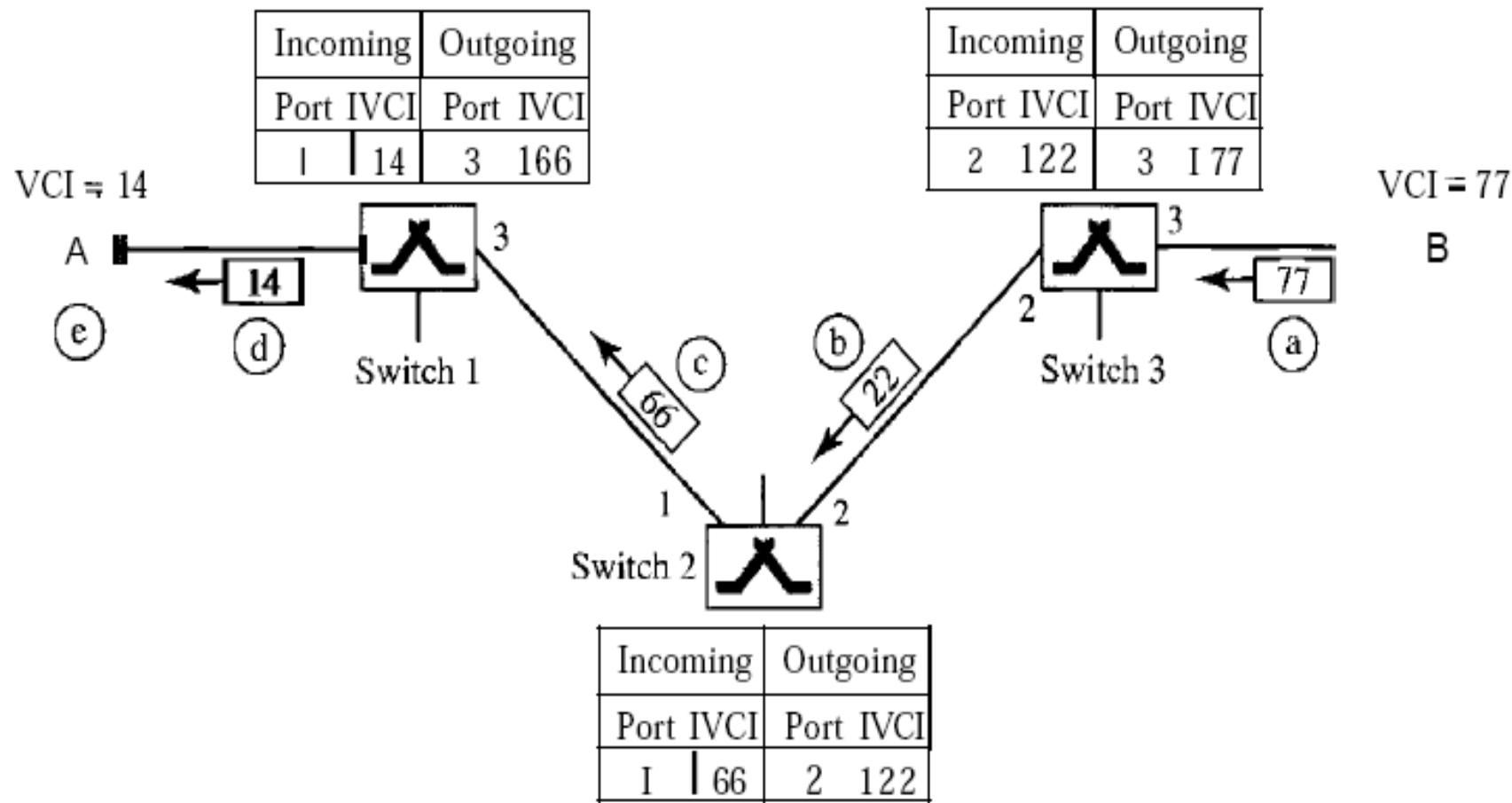


Figure 8.15 Setup acknowledgment in a virtual-circuit network



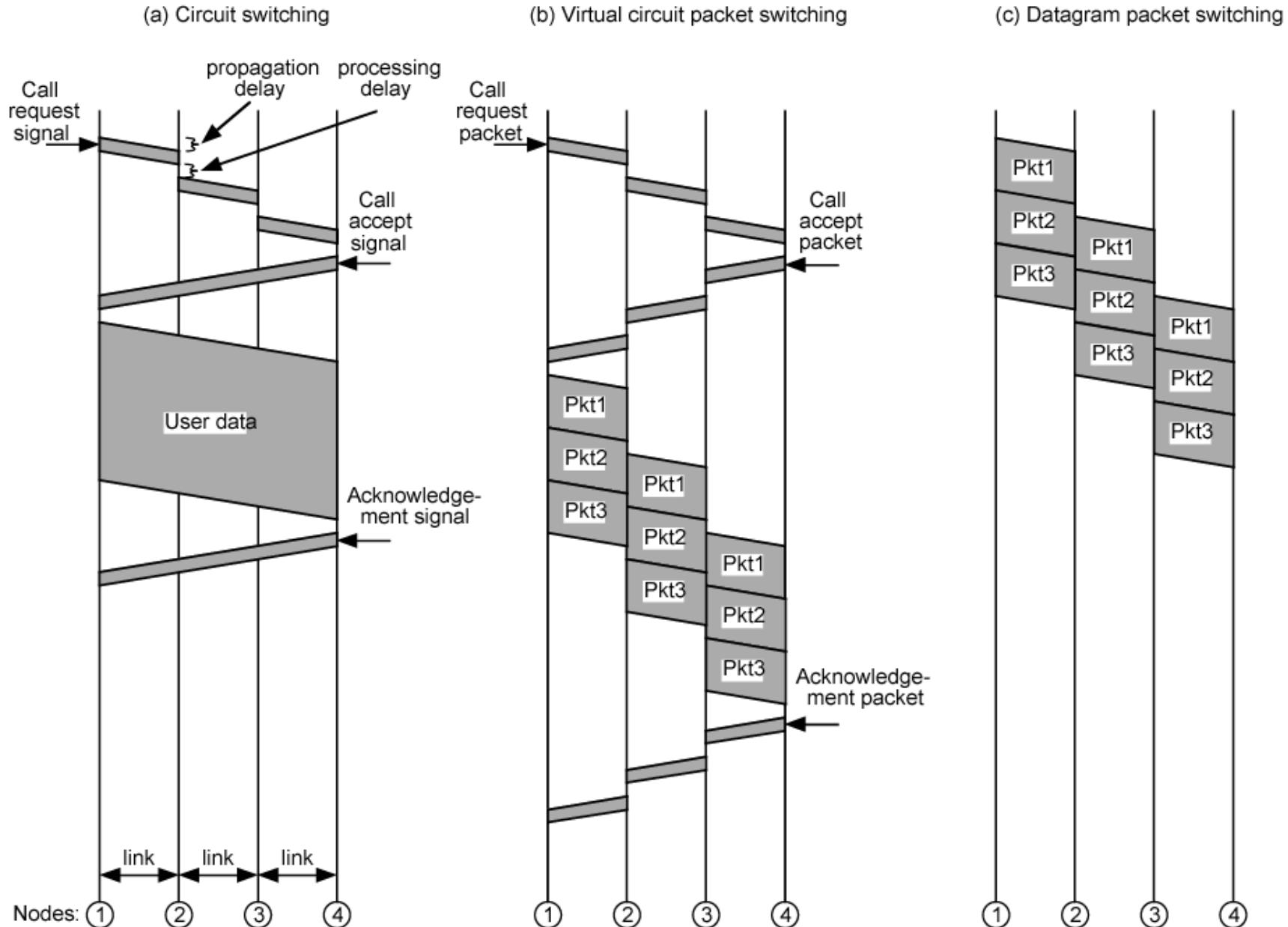
Virtual Circuits v Datagram

- Virtual circuits
 - Network can provide sequencing and error control
 - Packets are forwarded more quickly
 - No routing decisions to make
 - Less reliable
 - Loss of a node loses all circuits through that node
- Datagram
 - No call setup phase
 - Better if few packets
 - More flexible
 - Routing can be used to avoid congested parts of the network

Circuit v Packet Switching

- Performance
 - Propagation delay
 - Transmission time
 - Node delay

Event Timing



X.25

- 1976
- Interface between host and packet switched network
- Almost universal on packet switched networks and packet switching in ISDN
- Defines three layers
 - Physical
 - Link
 - Packet

X.25 - Physical

- Interface between attached station and link to node
- Data terminal equipment DTE (user equipment)
- Data circuit terminating equipment DCE (node)
- Uses physical layer specification X.21
- Reliable transfer across physical link
- Sequence of frames

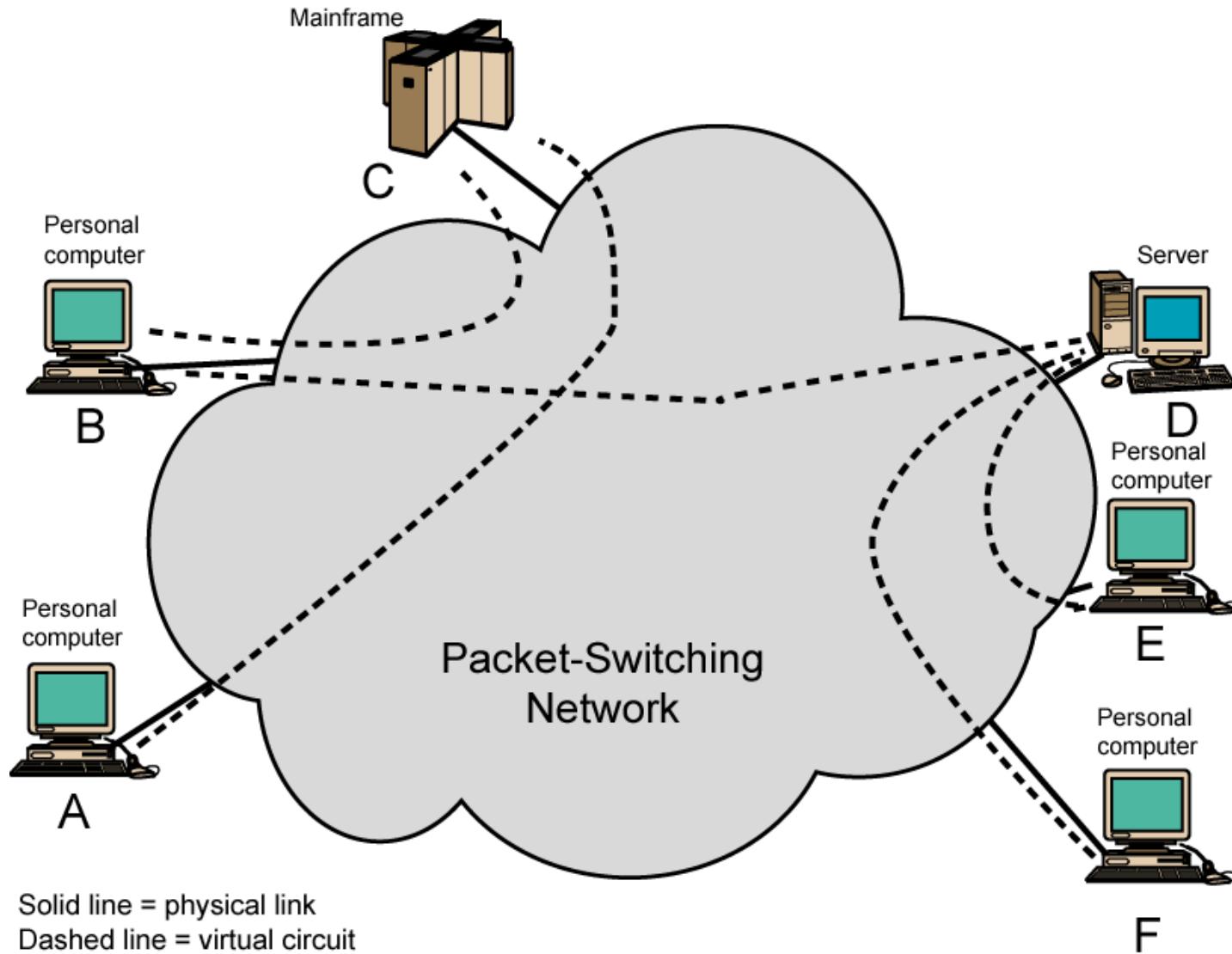
X.25 - Link

- Link Access Protocol Balanced (LAPB)
 - Subset of HDLC
 - see chapter 7

X.25 - Packet

- External virtual circuits
- Logical connections (virtual circuits) between subscribers

X.25 Use of Virtual Circuits



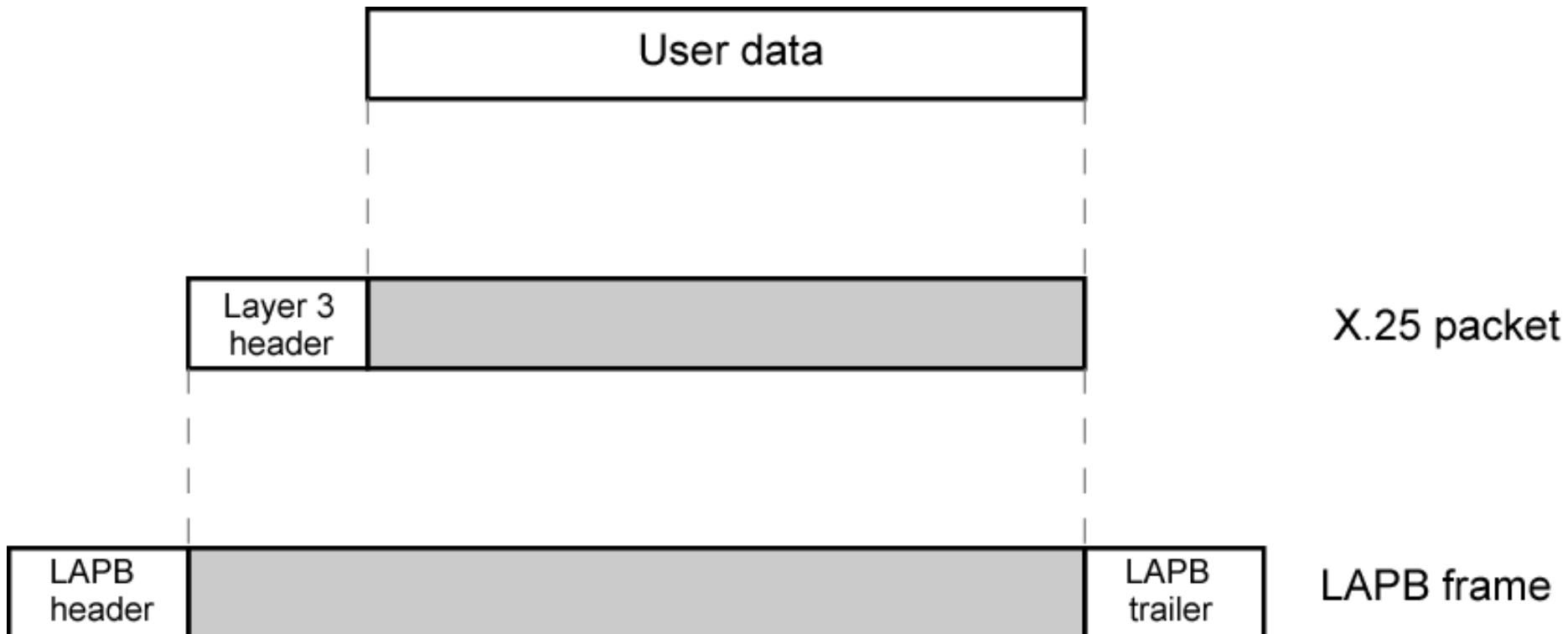
Virtual Circuit Service

- Logical connection between two stations
 - External virtual circuit
- Specific preplanned route through network
 - Internal virtual circuit
- Typically one to one relationship between external and internal virtual circuits
- Can employ X.25 with datagram style network
- External virtual circuits require logical channel
 - All data considered part of stream

X.25 Levels

- User data passes to X.25 level 3
- X.25 appends control information
 - Header
 - Identifies virtual circuit
 - Provides sequence numbers for flow and error control
- X.25 packet passed down to LAPB entity
- LAPB appends further control information

User Data and X.25 Protocol Control Information



Frame Relay

- Designed to be more efficient than X.25
- Developed before ATM
- Larger installed base than ATM
- ATM now of more interest on high speed networks

Frame Relay Background - X.25

- Call control packets, in band signaling
- Multiplexing of virtual circuits at layer 3
- Layer 2 and 3 include flow and error control
- Considerable overhead
- Not appropriate for modern digital systems with high reliability

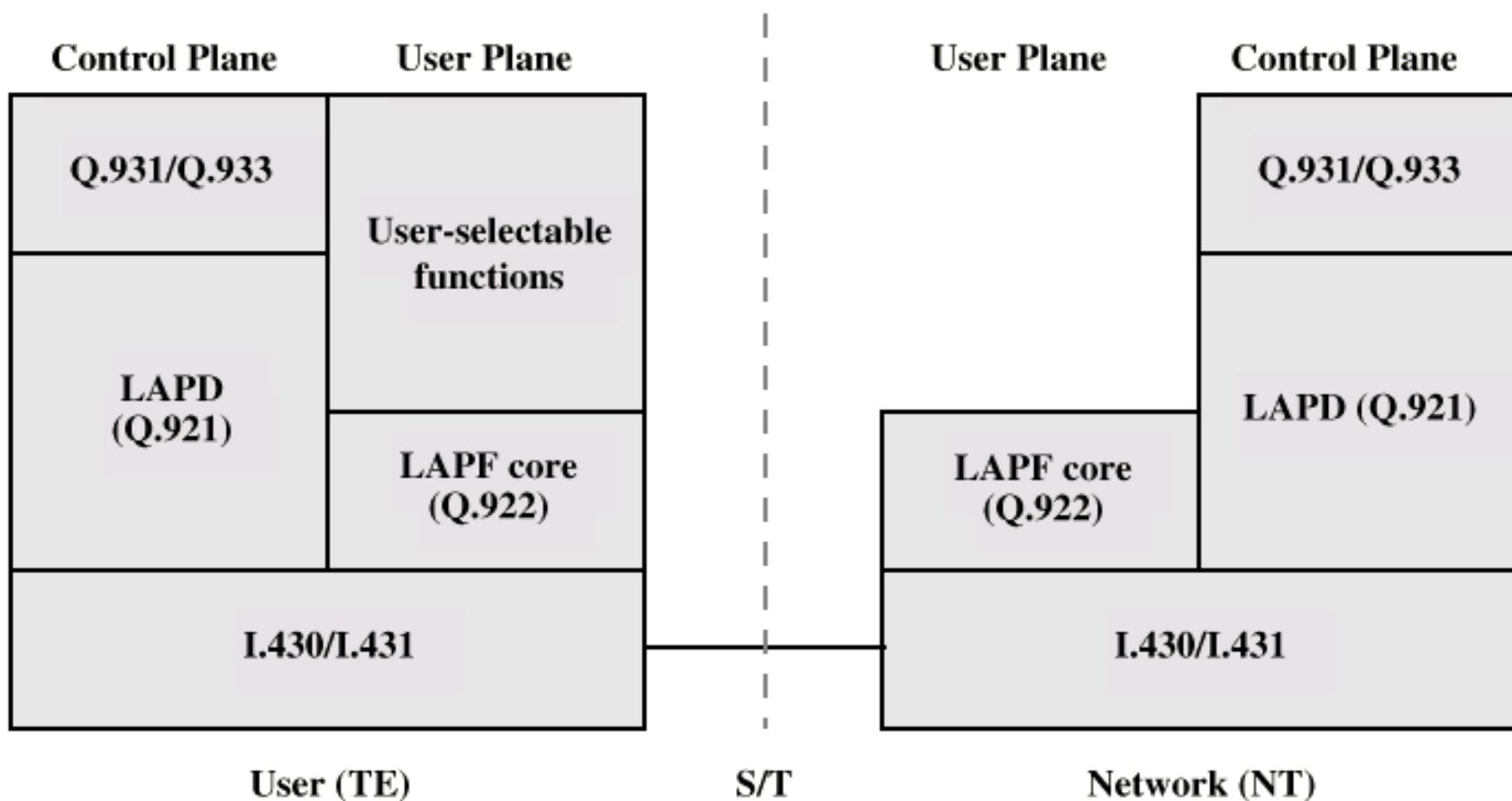
Frame Relay - Differences

- Call control carried in separate logical connection
- Multiplexing and switching at layer 2
 - Eliminates one layer of processing
- No hop by hop error or flow control
- End to end flow and error control (if used) are done by higher layer
- Single user data frame sent from source to destination and ACK (from higher layer) sent back

Advantages and Disadvantages

- Lost link by link error and flow control
 - Increased reliability makes this less of a problem
- Streamlined communications process
 - Lower delay
 - Higher throughput
- ITU-T recommend frame relay above 2Mbps

Protocol Architecture



Control Plane

- Between subscriber and network
- Separate logical channel used
 - Similar to common channel signaling for circuit switching services
- Data link layer
 - LAPD (Q.921)
 - Reliable data link control
 - Error and flow control
 - Between user (TE) and network (NT)
 - Used for exchange of Q.933 control signal messages

User Plane

- End to end functionality
- Transfer of info between ends
- LAPF (Link Access Procedure for Frame Mode Bearer Services) Q.922
 - Frame delimiting, alignment and transparency
 - Frame mux and demux using addressing field
 - Ensure frame is integral number of octets (zero bit insertion/extraction)
 - Ensure frame is neither too long nor short
 - Detection of transmission errors
 - Congestion control functions