**PART 1 Overview**

**Chapter 1** Introduction

**Chapter 2** Network Models

**PART 2 Physical Layer and Media**

**Chapter 3** Data and Signals

**Chapter 4** Digital Transmission

**Chapter 5** Analog Transmission

**Chapter 6** Bandwidth Utilization: Multiplexing and Spreading

**Chapter 7** Transmission Media

**Chapter 8** Switching

**Chapter 9** Using Telephone and Cable Networks for Data Transmission

**PART 3 Data Link Layer**

**Chapter 10** Error Detection and Correction

**Chapter 11** Data Link Control

**Chapter 12** Multiple Access

**Chapter 13** Wired LANs: Ethernet

**Chapter 14** Wireless LANs

**Chapter 15** Connecting LANs, Backbone Networks, and Virtual LANs

**Chapter 16** Wireless WANs: Cellular Telephone and Satellite Networks

**Chapter 17** SONETISDH

**Chapter 18** Virtual-Circuit Networks: Frame Relay and ATM

**PART 4 Network Layer**

**Chapter 19** Network Layer: Logical Addressing

**Chapter 20** Network Layer: Internet Protocol

**Chapter 21** Network Layer: Address Mapping, Error Reporting, and Multicasting

**Chapter 22** Network Layer: Delivery, Fonvarding, and Routing

**PARTS 5 Transport Layer**

**Chapter 23** Process-to-Process Delivery: UDP, TCP, and SCTP

**Chapter 24** Congestion Control and Quality of Service

**PART 6 Application Layer**

**Chapter 25** Domain Name System

**Chapter 26** Remote Logging, Electronic Mail, and File Transfer

**Chapter 27** WWW and HTTP

**Chapter 28** Network Management: SNMP

**Chapter 29** Multimedia

**PART 7 Security**

**Chapter 30** Cryptography

**Chapter 31** Network Security

**Chapter 32** Security in the Internet: IPSec, SSLlTLS, PCp, VPN, and Firewalls

**Appendix A** Unicode

**Appendix B** Numbering Systems

**Appendix C** Mathematical Review

**Appendix D** 8B/6T Code

**Appendix E** Telephone History

**Appendix F** Co!1tact Addresses

**Appendix G** RFCs

**Appendix H** UDP and TCP Ports

**Part 1**

**Overview**

Four major concepts are discussed: **data communications, networking, protocols and standards, and networking models.**

Networks exist so that data may be sent from one place to another-the basic concept of **data communications**. Data communications between remote parties can be achieved through a process called **networking**,

**Protocols and standards** are vital to the implementation of data communications and networking. **Protocols** refer to the rules; a **standard** is a protocol that has been adopted by vendors and manufacturers.

**Network models** serve to organize, unify, and control the hardware and software components of data communications and networking. Although the term "network model" suggests a relationship to networking, the model also encompasses data communications.

**Chapter 1**

In Chapter 1, we introduce the concepts of data communications and networking. We discuss data communications components, data representation, and data flow. We then move to the structure of networks that carry data. We discuss network topologies, categories of networks, and the general idea behind the Internet. The section on protocols and standards gives a quick overview of the organizations that set standards in data communications and networking.

**Chapter 2**

The two dominant networking models are the Open Systems Interconnection (OSI) and the Internet model (TCP/IP). The first is a theoretical framework; the second is the actual model used in today's data communications. In Chapter 2, we first discuss the OSI model to give a general background. We then concentrate on the Internet model, which is the foundation for the rest of the book.

**Chapter 1**

**INTRODUCTION**

**1.1 DATA COMMUNICATIONS**

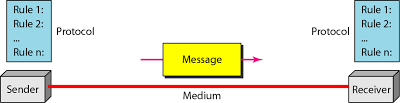
The term telecommunication, which includes telephony, telegraphy, and television, means communication at a distance (tele is Greek word for "far"). The word data refers to information presented in whatever form is agreed upon by the parties creating and using the data.

Data communications are the exchange of data between two devices via some form of transmission medium such as a wire cable.

Data communications system depends on four fundamental characteristics:

1. **Delivery**. The system must deliver data to the correct destination. Data must be received by the intended device or user and only by that device or user.
2. **Accuracy**. The system must deliver the data accurately. Data that have been altered in transmission and left uncorrected are unusable.
3. **Timeliness**. The system must deliver data in a timely manner. Data delivered late are useless. In the same order that data are produced, and without significant delay deliver it. This kind of delivery is called real-time transmission.
4. **Jitter**. Jitter refers to the variation in the packet arrival time. It is the uneven delay in the delivery of audio or video packets.

**1.1.1 COMPONENTS**



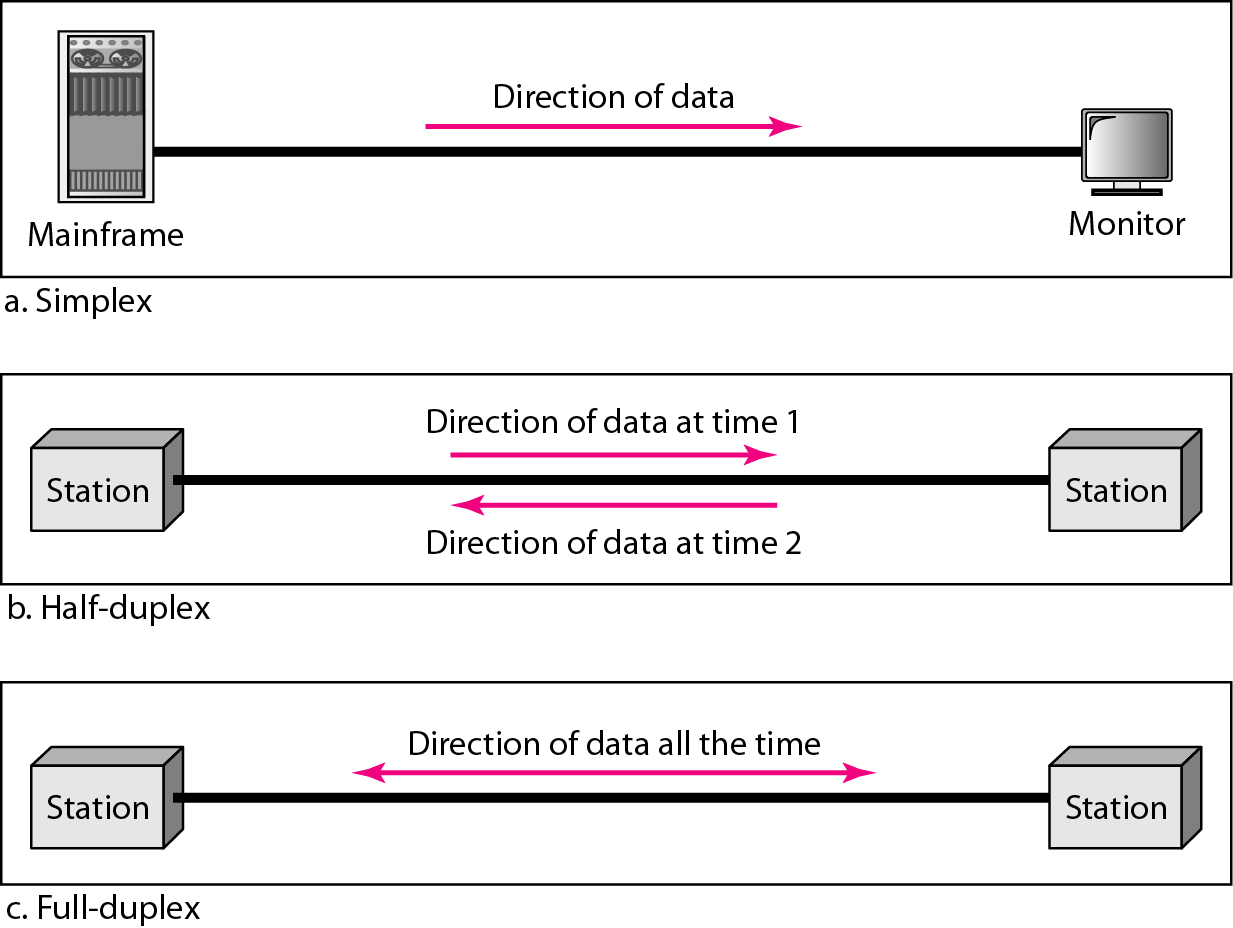
1. **Message**. The message is the information (data) to be communicated. Popular forms of information include text, numbers, pictures, audio, and video.
2. **Sender**. The sender is the device that sends the data message. It can be a computer, workstation, telephone handset, video camera, and so on.
3. **Receiver**. The receiver is the device that receives the message. It can be a computer, workstation, telephone handset, television, and so on.
4. **Transmission medium**. The transmission medium is the physical path by which a message travels from sender to receiver. Some examples are twisted-pair wire, coaxial cable, fiber-optic cable, and radio waves.
5. **Protocol**. A protocol is a set of rules that govern data communications. It represents an agreement between the communicating devices. Without a protocol, is like just as a person speaking French cannot be understood by a person who speaks only Japanese.

**1.1.2 DATA REPRESENTATION**

1. **Text**: - Text is represented as a bit pattern. Bit patterns have been designed to represent text symbols. The process of representing symbols is called coding. Today’s coding system is called Unicode, which uses 32 bits to represent a symbol, A decades ago ASCII was developed which used (7bit ASCII or 8bit ASCII).
2. **Number**: - Also represented by bit patterns. ASCII is not used to represent numbers; The number is directly converted to a binary number to simplify mathematical operations
3. **Images**: - also represented by bit patterns. In its simplest form, an image is composed of a matrix of pixels. The size of the pixel depends on the resolution. an image made of only black and white dots (e.g., a chessboard), a 1-bit pattern is enough to represent a pixel. To represent colour images. One method is called RGB (red, green, and blue). The intensity of each colour is measured, and a bit pattern is assigned to it. Another method is called YCM (yellow, cyan, and magenta).
4. **Audio**: - refers to the recording or broadcasting of sound or music.
5. **Video**: - refers to the recording or broadcasting of a picture or movie.

**1.1.3 DATA FLOW**

Communication between two devices can be simplex, half-duplex, or full-duplex:



**I. Simplex**

In simplex mode, the communication is unidirectional, as on a one-way street. Only one of the two devices on a link can transmit; the other can only receive. The simplex mode can use the entire capacity of the channel to send data in one direction.

**II. Half-Duplex**

In half-duplex mode, each station can both transmit and receive, but not at the same time. When one device is sending, the other can only receive, and vice versa. The half-duplex mode is used in cases where there is no need for communication in both directions at the same time; the entire capacity of the channel can be utilized for each direction.

**III. Full-Duplex**

In full-duplex mode (also called duplex), both stations can transmit and receive simultaneously. In full-duplex mode, signals going in one direction share the capacity of the link with signals going in the other direction. This sharing can occur in two ways: Either the link must contain two physically separate transmission paths, one for sending and the other for receiving; or the capacity of the channel is divided between signals traveling in both directions. The full-duplex mode is used when communication in both directions is required all the time. The capacity of the channel, however, must be divided between the two directions.

**1.2 NETWORKS**

A network is a set of devices (often referred to as nodes) connected by communication links. A node can be a computer, printer, or any other device capable of sending and/or receiving data generated by other nodes on the network.

**1.2.1 DISTRIBUTED PROCESSING**

in this a task is divided among multiple computers. Instead of one single large machine being responsible for all aspects of a process, separate computers (usually a personal computer or workstation) handle a subset of process.

**1.2.2 NETWORK CRITERIA**

**I. Performance**

Transit time is the amount of time required for a message to travel from one device to another. Response time is the elapsed time between an inquiry and a response. Performance is often evaluated by two networking metrics: throughput and delay. We often need more throughput and less delay.

**II. Reliability**

In addition to accuracy of delivery, network reliability is measured by the frequency of failure, the time it takes a link to recover from a failure, and the network's robustness in a catastrophe.

**III. Security**

Network security issues include protecting data from unauthorized access, protecting data from damage and development, and implementing policies and procedures for recovery from breaches and data losses.

**1.2.3 PHYSICAL STRUCTURES**

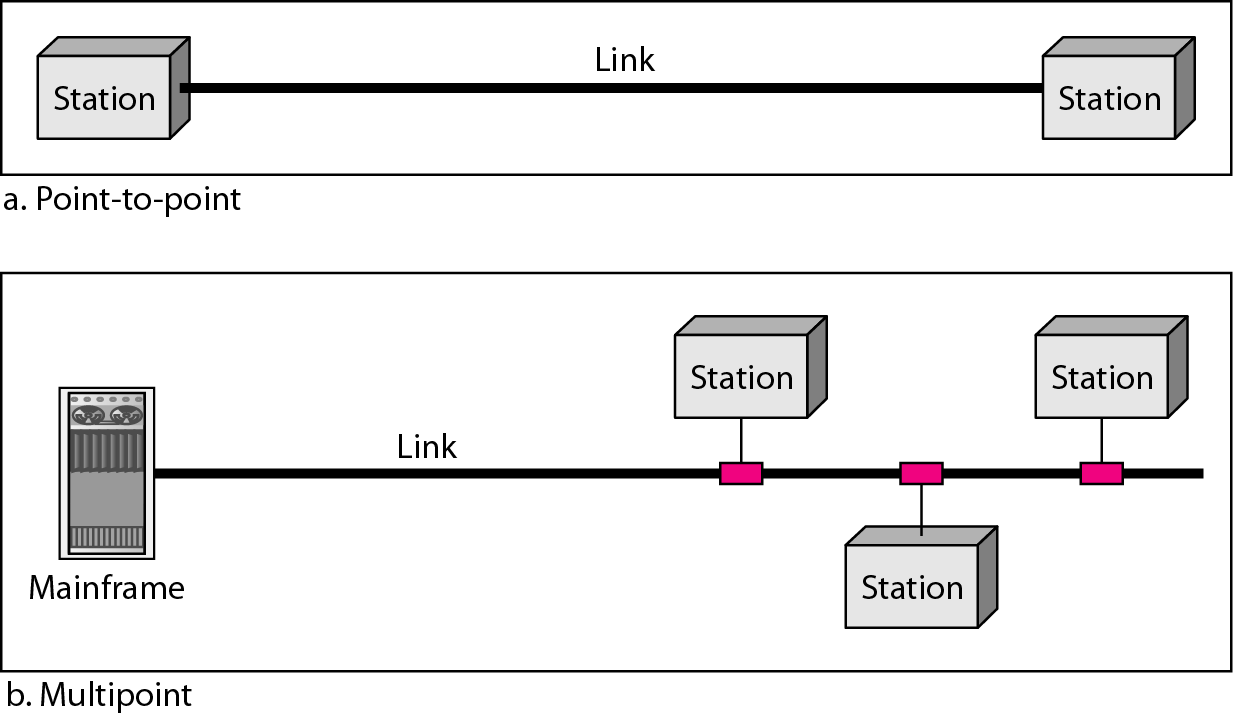
some network attributes:

**1.2.3.1 Type of Connection**

A network is two or more devices connected through links. A link is a communications pathway that transfers data from one device to another. There are two possible types of connections: point-to-point and multipoint.

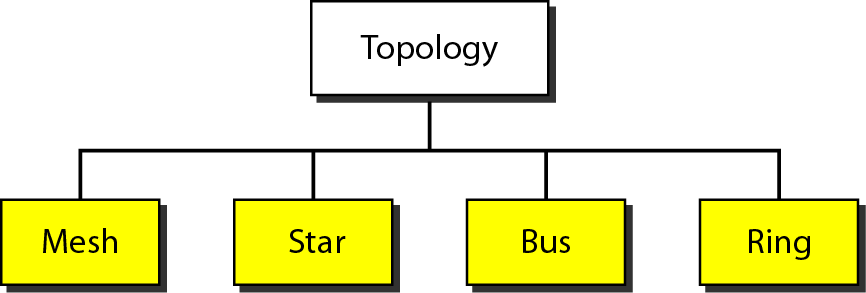
**Point-to-Point:** A point-to-point connection provides a dedicated link between two devices. The entire capacity of the link is reserved for transmission between those two devices. Most point-to-point connections use an actual length of wire or cable to connect the two ends, but other options, such as microwave or satellite links, are also possible.

**Multipoint:** A multipoint (also called multidrop) connection is one in which more than two specific devices share a single link. In a multipoint environment, the capacity of the channel is shared, either spatially or temporally. If several devices can use the link simultaneously, it is a spatially shared connection. If users must take turns, it is a timeshared connection.

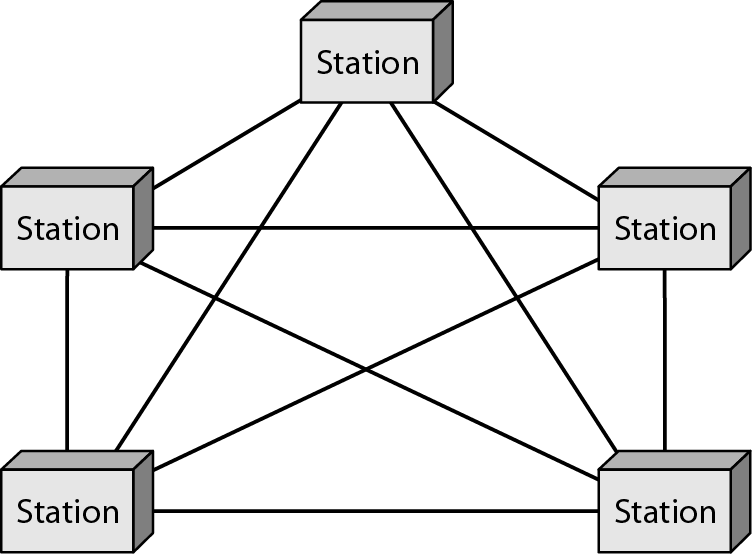


**1.2.3.2 Physical Topology**

The term physical topology refers to the way in which a network is laid out physically.: Two or more devices connect to a link; two or more links form a topology. The topology of a network is the geometric representation of the relationship of all the links and linking devices (usually called nodes) to one another. There are four basic topologies possible: mesh, star, bus, and ring.



**Mesh:** In a mesh topology, every device has a dedicated point-to-point link to every other device. The term dedicated means that the link carries traffic only between the two devices it connects. In a mesh topology, we need {N(N-1)}/2 duplex-mode links.



several advantages over other network topologies:

1ST The use of dedicated links guarantees that each connection.

2ND A mesh topology is robust (If one link becomes unusable, it does not incapacitate the entire system).

3RD There is the advantage of privacy or security.

4TH Point-to-Point links make fault identification and fault isolation easy.

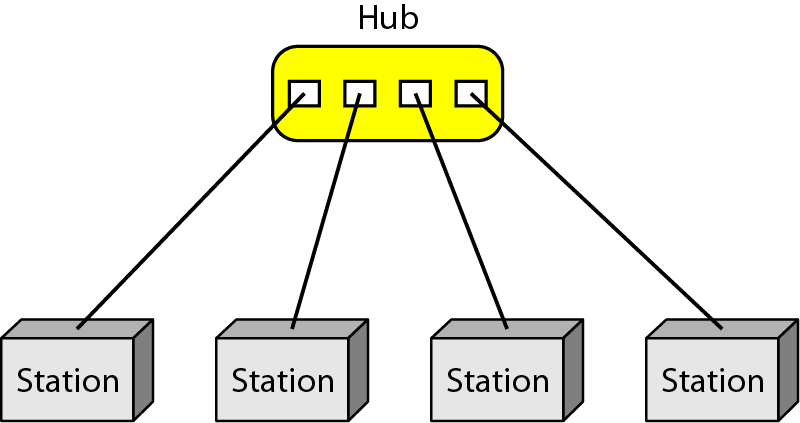
disadvantages of a mesh

1ST Installation and reconnection are difficult.

2ND The sheer bulk of the wiring can be greater than the available space (in walls, ceilings, or floors) can accommodate.

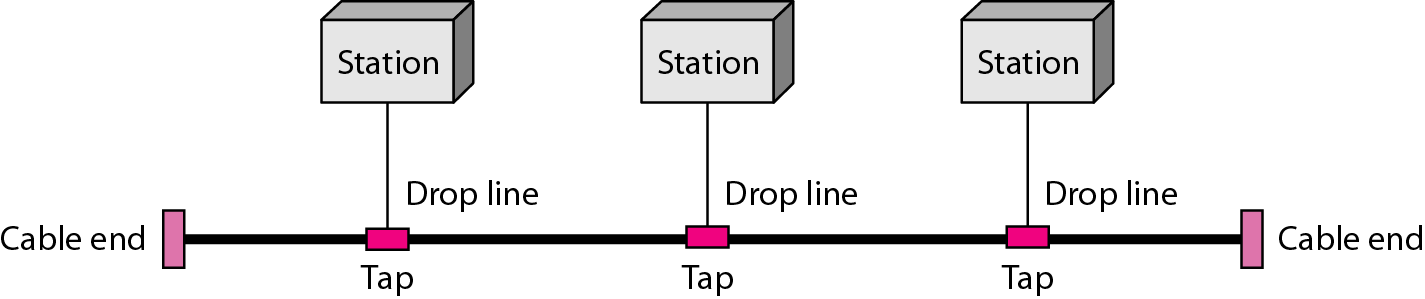
3RD The hardware required to connect each link can be prohibitively expensive.

**Star Topology:** In a star topology, each device has a dedicated point-to-point link only to a central controller, usually called a hub. In a star, each device needs only one link and one I/O port to connect it to any number of others. This factor also makes it easy to install and reconfigure. advantages include robustness (If one link fails, only that link is affected. All other links remain active). This factor also lends itself to easy fault identification and fault isolation. As long as the hub is working, it can be used to monitor link problems and bypass defective links.



One big disadvantage of a star topology is If the hub goes down, the whole system is dead.

**Bus Topology:** A bus topology, is multipoint. One long cable acts as a backbone to link all the devices in a network. Nodes are connected to the bus cable by drop lines and taps. A tap is a connector that either splices into the main cable or punctures the sheathing of a cable to create a contact with the metallic core

. 

Advantages of a bus topology

1ST Ease of installation.

2ND A bus uses less cabling than mesh or star topologies.

3RD In a bus, this redundancy is eliminated. Only the backbone cable stretches through the entire facility. Each drop line has to reach only as far as the nearest point on the backbone.

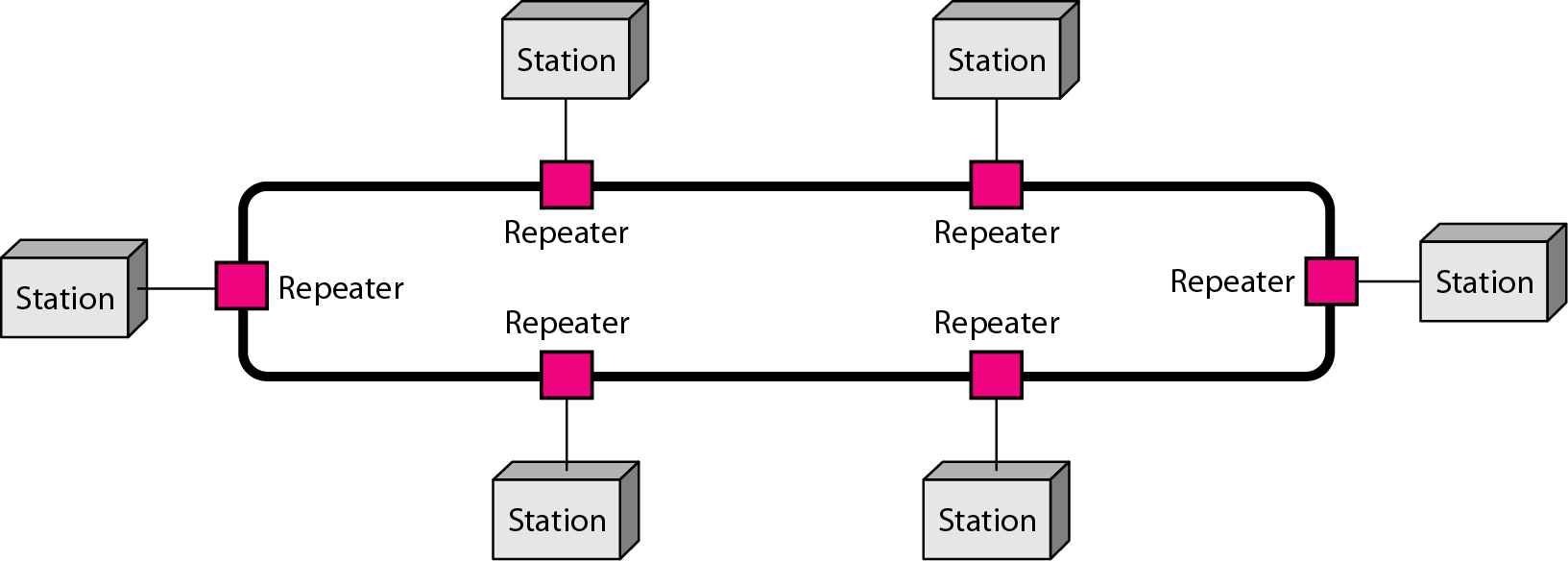
Disadvantages

1ST Difficult reconnection and fault isolation.

2ND A bus is usually designed to be optimally efficient at installation.

3RD In addition, a fault or break in the bus cable stops all transmission

**Ring Topology: -** In a ring topology, each device has a dedicated point-to-point connection with only the two devices on either side of it. A signal is passed along the ring in one direction, from device to device, until it reaches its destination. Each device in the ring incorporates a repeater. When a device receives a signal intended for another device, its repeater regenerates the bits and passes them along.



Advantages of a ring

1ST relatively easy to install and reconfigure.

2ND Add or delete a device requires changing only two connections.

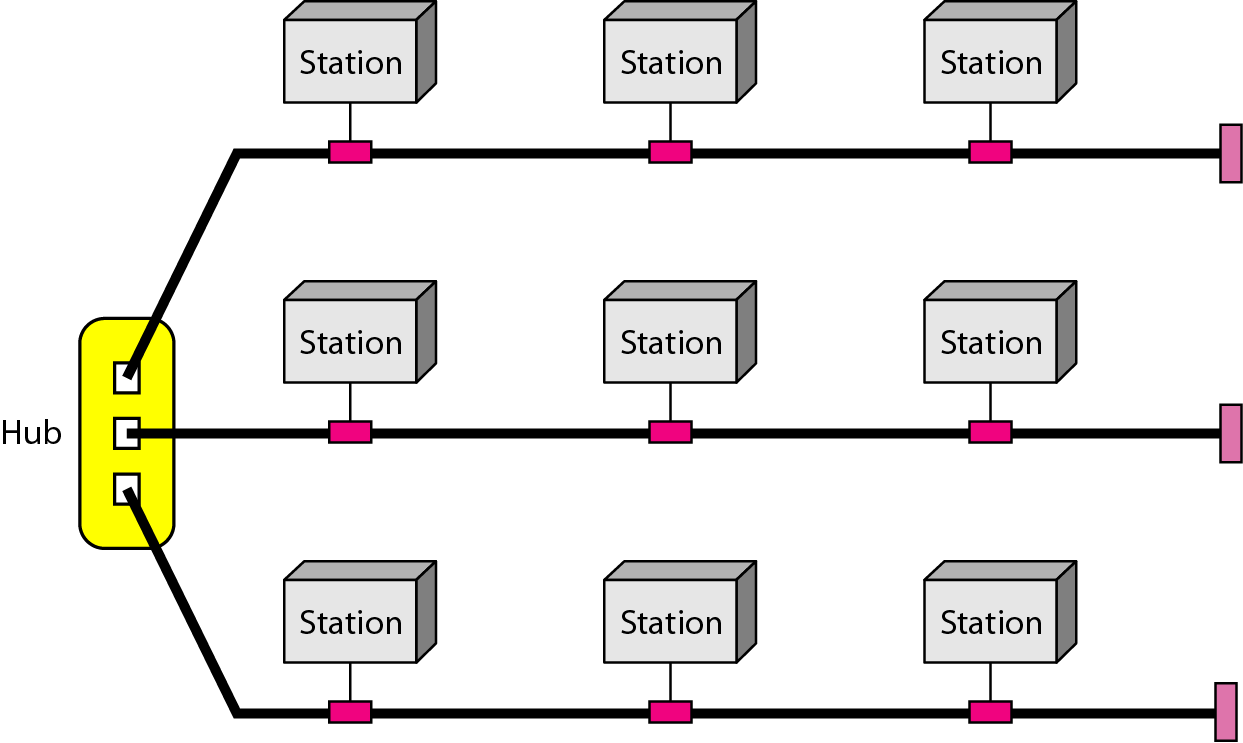
3RD Fault isolation is simplified.

Disadvantages of a ring

1ST unidirectional traffic can be a disadvantage.

2ND A break in the ring (such as a disabled station) can disable the entire network.

**Hybrid Topology:** A network can be hybrid. For example, we can have a main star topology with each branch connecting several stations in a bus topology

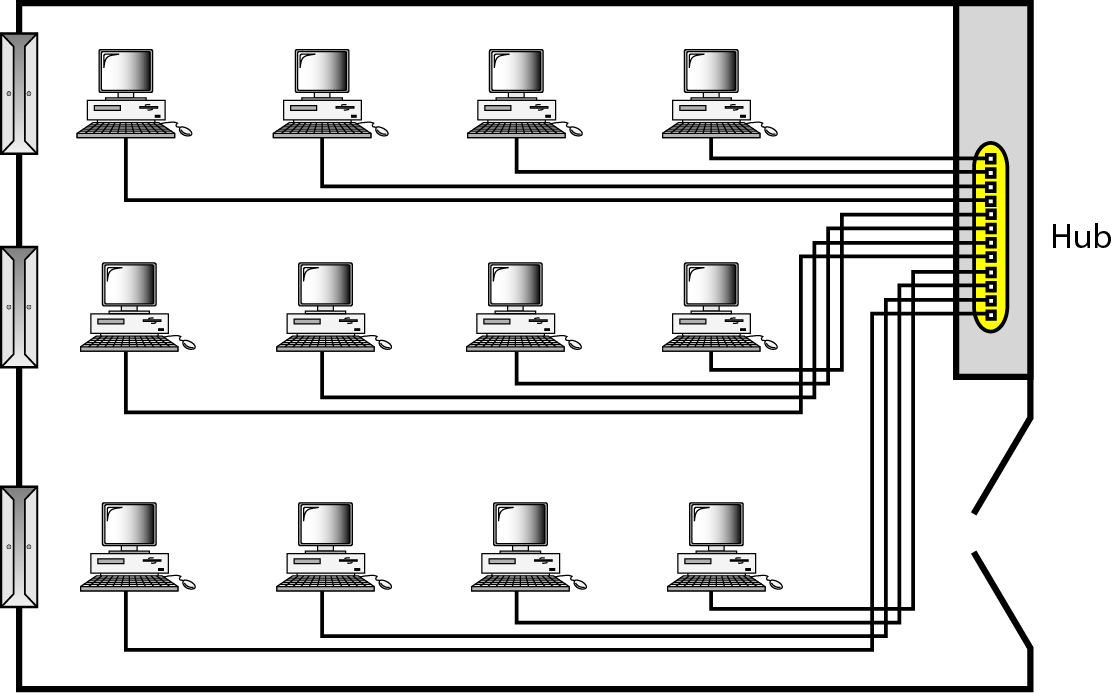


**Categories of Networks**

Generally referring to two primary categories: local-area networks and wide-area networks.

**Local Area Network**

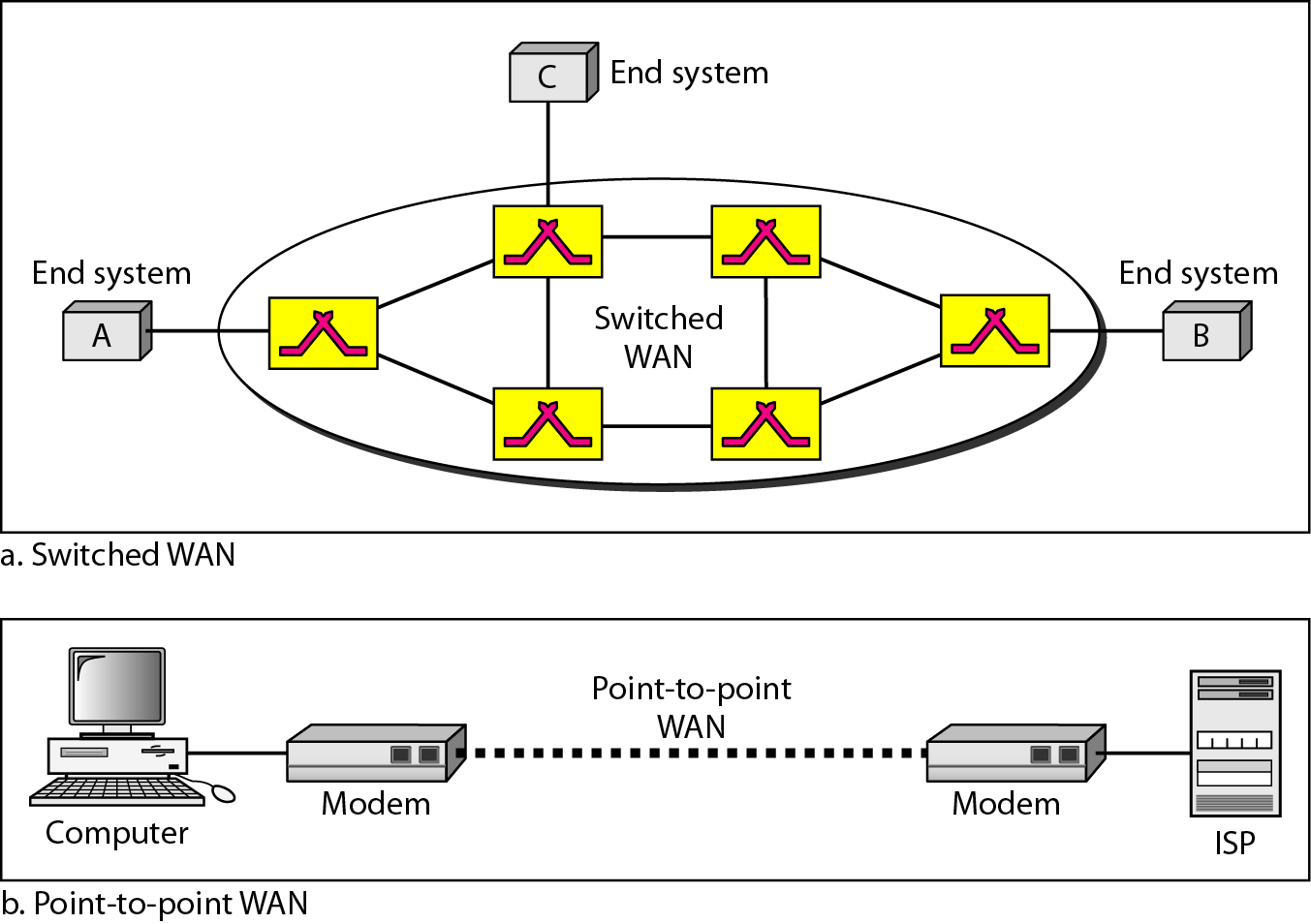
A local area network (LAN) is usually privately owned and links the devices in a single office, building, or campus. Currently, LAN size is limited to a few metres.



The resources to be shared can include hardware (e.g., a printer), software (e.g., an application program), or data. LANs are distinguished from other types of networks by their transmission media and topology. The most common LAN topologies are bus, ring, and star.

**Wide Area Network**

A wide area network (WAN) provides long-distance transmission of data, image, audio, and video information over large geographic areas that may comprise a country, a continent, or even the whole world. A WAN can be as complex as the backbones that connect the Internet (also refer as switched WAN) or as simple as a dial-up line that connects a home computer to the Internet (also refer as point-to-point WAN). The switched WAN connects the end systems, which usually comprise a router that connects to another LAN or WAN. The point-to-point WAN is normally a line leased from a telephone or cable TV provider that connects a home computer or a small LAN to an Internet service provider (lSP). This type of WAN is often used to provide Internet access.



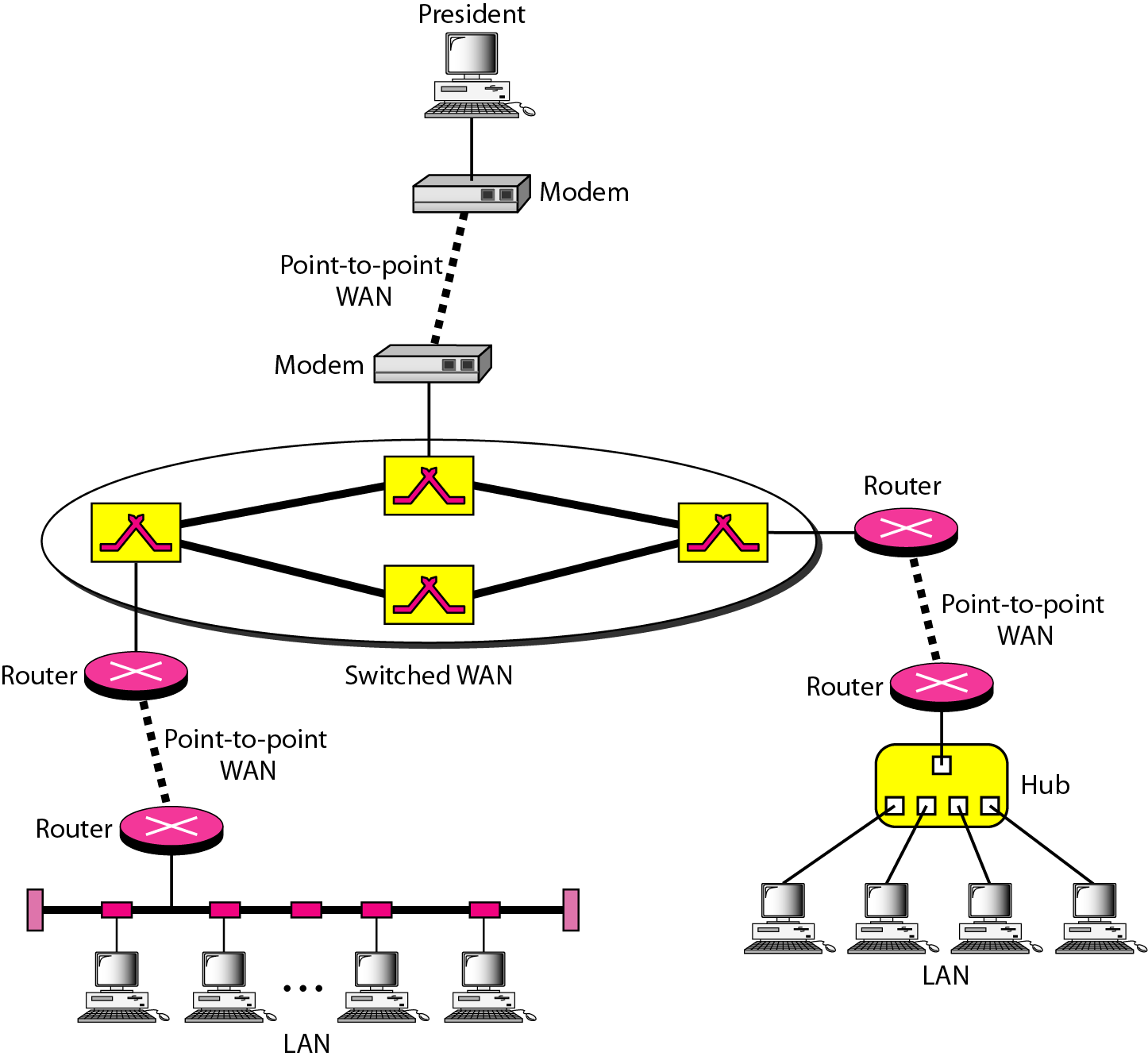
An early example of a switched WAN is X.25, a network designed to provide connectivity between end users. X.25 is being gradually replaced by a high-speed, more efficient network called Frame Relay. A good example of a switched WAN is the asynchronous transfer mode (ATM) network, which is a network with fixed-size data unit packets called cells. Another example of WANs is the wireless WAN that is becoming more and more popular.

**Metropolitan Area Networks**

A metropolitan area network (MAN) is a network with a size between a LAN and a WAN. It normally covers the area inside a town or a city. It is designed for customers who need a high-speed connectivity, normally to the Internet, and have endpoints spread over a city or part of city. A good example of a MAN is the part of the telephone company network that can provide a high-speed DSL line to the customer. Another example is the cable TV network that originally was designed for cable TV, but today can also be used for high-speed data connection to the Internet.

**Interconnection of Networks: Internetwork**

When two or more networks are connected, they become an internetwork, or internet.



**1.3 THE INTERNET**

The Internet has revolutionized many aspects of our daily lives. It has affected the way we do business as well as the way we spend our leisure time. The Internet is a communication system that has brought a wealth of information to our fingertips and organized it for our use. The Internet is a structured, organized system. We begin with a brief history of the Internet. We follow with a description of the Internet today.

**A Brief History**

A network is a group of connected communicating devices such as computers and printers. An internet (note the lowercase letter i) is two or more networks that can communicate with each other. The most notable internet is called the Internet (uppercase letter I), a collaboration of more than hundreds of thousands of interconnected networks. Private individuals as well as various organizations such as government agencies, schools, research facilities, corporations, and libraries in more than 100 countries use the Internet. Millions of people are users. Yet this extraordinary communication system only came into being in 1969.

In the mid-1960s, mainframe computers in research organizations were standalone devices. Computers from different manufacturers were unable to communicate with one another. The Advanced Research Projects Agency (ARPA) in the Department of Defence (DoD) was interested in finding a way to connect computers so that the researchers they funded could share their findings, thereby reducing costs and eliminating duplication of effort.

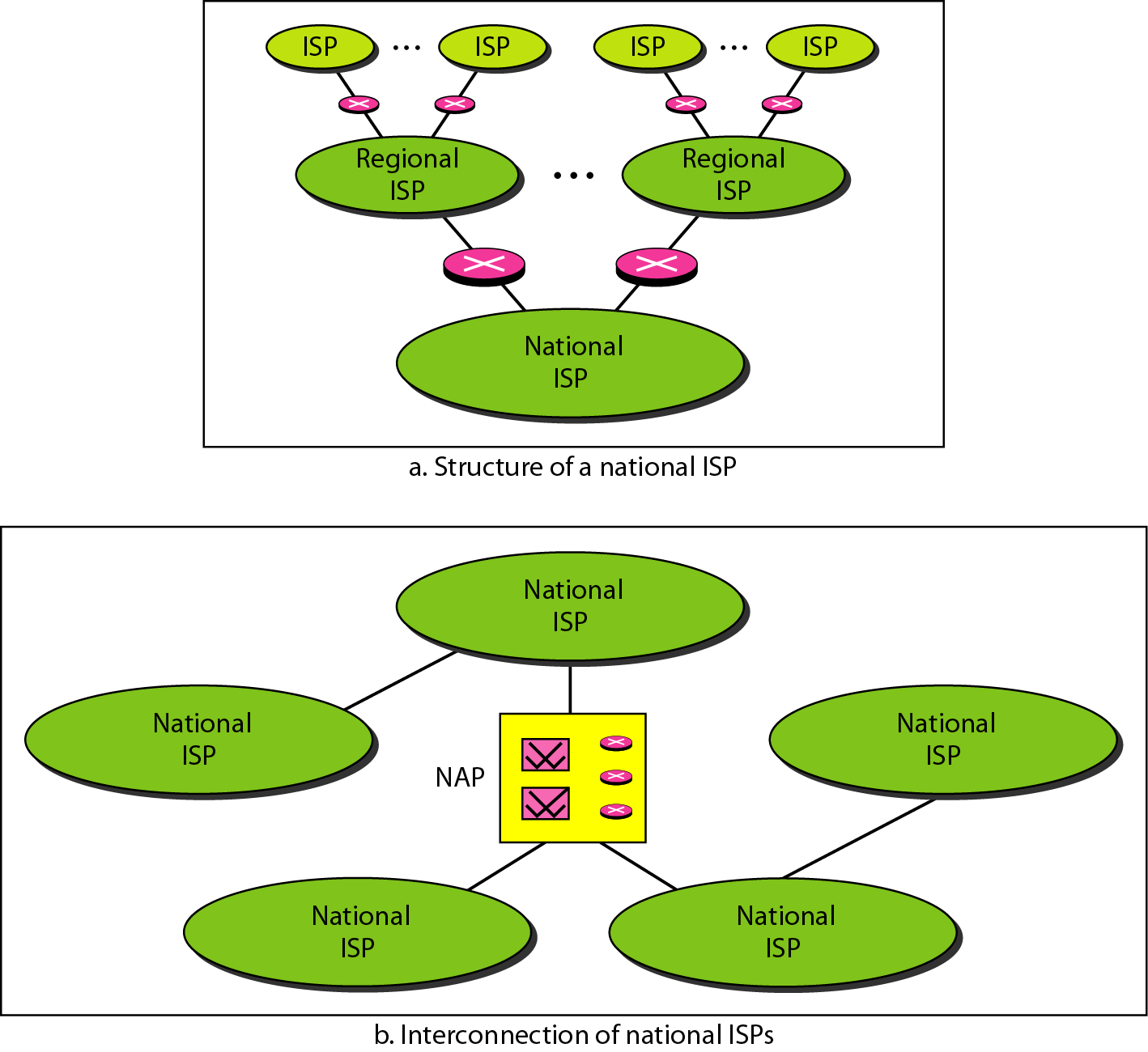
In 1967, at an Association for Computing Machinery (ACM) meeting, ARPA presented its ideas for ARPANET, a small network of connected computers. The idea was that each host computer would be attached to a specialized computer, called an interface message processor (IMP). The IMPs, in tum, would be connected to one another. Each IMP had to be able to communicate with other IMPs as well as with its own attached host.

By 1969, ARPANET was a reality. Four nodes, at the University of California at Los Angeles (UCLA), the University of California at Santa Barbara (UCSB), Stanford Research Institute (SRI), and the University of Utah, were connected via the IMPs to form a network. Software called the Network Control Protocol (NCP) provided communication between the hosts.

In 1972, Vint Cerf and Bob Kahn, both of whom were part of the core ARPANET group, collaborated on what they called the Internetting Project. Cerf and Kahn's landmark 1973 paper outlined the protocols to achieve end-to-end delivery of packets. This paper on Transmission Control Protocol (TCP) included concepts such as encapsulation, the datagram, and the functions of a gateway. Shortly thereafter, authorities made a decision to split TCP into two protocols: Transmission Control Protocol (TCP) and Internetworking Protocol (lP). IP would handle datagram routing while TCP would be responsible for higher-level functions such as segmentation, reassembly, and error detection. The internetworking protocol became known as TCP/IP.

**The Internet Today**

The Internet has come a long way since the 1960s. The Internet today is not a simple hierarchical structure. It is made up of many wide- and local-area networks joined by connecting devices and switching stations. Today most end users who want Internet connection use the services of Internet service providers (lSPs). There are international service providers, national service providers, regional service providers, and local service providers.



**International Internet Service Providers**

At the top of the hierarchy are the international service providers that connect nations together.

**National Internet Service Providers**

The national Internet service providers are backbone networks created and maintained by specialized companies. To provide connectivity between the end users, these backbone networks are connected by complex switching stations (normally run by a third party) called network access points (NAPs). Some national ISP networks are also connected to one another by private switching stations called peering points. These normally operate at a high data rate (up to 600 Mbps).

**Regional Internet Service Providers**

Regional internet service providers or regional ISPs are smaller ISPs that are connected to one or more national ISPs. They are at the third level of the hierarchy with a smaller data rate.

**Local Internet Service Providers**

Local Internet service providers provide direct service to the end users. The local ISPs can be connected to regional ISPs or directly to national ISPs. Most end users are connected to the local ISPs.

**1.4 PROTOCOLS AND STANDARDS**

In this section, we define two widely used terms: protocols and standards. First, we define protocol, which is synonymous with rule. Then we discuss standards, which are agreed-upon rules.

**Protocols**

Two entities (nodes) cannot simply send bit streams to each other and expect to be understood. For communication to occur, the entities must agree on a protocol. A protocol is a set of rules that govern data communications. A protocol defines what is communicated, how it is communicated, and when itis communicated. The key elements of a protocol are syntax, semantics, and timing.

1. **Syntax.** The term syntax refers to the structure or format of the data, meaning the order in which they are presented. For example, a simple protocol might expect the first 8 bits of data to be the address of the sender, the second 8 bits to be the address of the receiver, and the rest of the stream to be the message itself.
2. **Semantics.** The word semantics refers to the meaning of each section of bits. How is a particular pattern to be interpreted, and what action is to be taken based on that interpretation? For example, does an address identify the route to be taken or the final destination of the message?
3. **Timing.** The term timing refers to two characteristics: when data should be sent and how fast they can be sent. For example, if a sender produces data at 100 Mbps but the receiver can process data at only 1Mbps, the transmission will overload the receiver and some data will be lost.

**Standards**

Standards are essential in creating and maintaining an open and competitive market for equipment manufacturers and in guaranteeing national and international interoperability of data and telecommunications technology and processes. Standards provide guidelines to manufacturers, vendors, government agencies, and other service providers to ensure the kind of interconnectivity necessary in today's marketplace and in international communications. Data communication standards fall into two categories: de facto (meaning "by fact" or "by convention") and de jure (meaning "by law" or "by regulation").

1. **De facto.** Standards that have not been approved by an organized body but have been adopted as standards through widespread use are de facto standards. De facto standards are often established originally by manufacturers who seek to define the functionality of a new product or technology.
2. **De jure.** Those standards that have been legislated by an officially recognized body are de jure standards.

**Standards Creation Committees**

While many organizations are dedicated to the establishment of standards, data telecommunications in North America rely primarily on those published by the following:

1. **International Organization for Standardization (ISO).** The ISO is a multinational body whose membership is drawn mainly from the standards creation committees of various governments throughout the world. The ISO is active in developing cooperation in the realms of scientific, technological, and economic activity.
2. **International Telecommunication Union-Telecommunication Standards Sector (ITU-T).** By the early 1970s, a number of countries were defining national standards for telecommunications, but there was still little international compatibility. The United Nations responded by forming, as part of its International Telecommunication Union (ITU), a committee, the Consultative Committee for International Telegraphy and Telephony (CCITT). This committee was devoted to the research and establishment of standards for telecommunications in general and for phone and data systems in particular. On March 1, 1993, the name of this committee was changed to the International Telecommunication Union Telecommunication Standards Sector (ITU-T).
3. **American National Standards Institute (ANSI).** Despite its name, the American National Standards Institute is a completely private, non-profit corporation not affiliated with the U.S. federal government. However, all ANSI activities are undertaken with the welfare of the United States and its citizens occupying primary importance.
4. **Institute of Electrical and Electronics Engineers (IEEE).** The Institute of Electrical and Electronics Engineers is the largest professional engineering society in the world. International in scope, it aims to advance theory, creativity, and product quality in the fields of electrical engineering, electronics, and radio as well as in all related branches of engineering. As one of its goals, the IEEE oversees the development and adoption of international standards for computing and communications.
5. **Electronic Industries Association (EIA).** Aligned with ANSI, the Electronic Industries Association is a non-profit organization devoted to the promotion of electronics manufacturing concerns. Its activities include public awareness education and lobbying efforts in addition to standards development. In the field of information technology, the EIA has made significant contributions by defining physical connection interfaces and electronic signaling specifications for data communication.

**Forums**

Telecommunications technology development is moving faster than the ability of standards committees to ratify standards. Standards committees are procedural bodies and by nature slow-moving. To accommodate the need for working models and agreements and to facilitate the standardization process, many special-interest groups have developed forums made up of representatives from interested corporations. The forums work with universities and users to test, evaluate, and standardize new technologies. By concentrating their efforts on a particular technology, the forums are able to speed acceptance and use of those technologies in the telecommunications community. The forums present their conclusions to the standards bodies.

**Regulatory Agencies**

All communications technology is subject to regulation by government agencies such as the Federal Communications Commission (FCC) in the United States. The purpose of these agencies is to protect the public interest by regulating radio, television, and wire/cable communications. The FCC has authority over interstate and international commerce as it relates to communications.

**Internet Standards**

An Internet standard is a thoroughly tested specification that is useful to and adhered to by those who work with the Internet. It is a formalized regulation that must be followed. There is a strict procedure by which a specification attains Internet standard status. A specification begins as an Internet draft. An Internet draft is a working document (a work in progress) with no official status and a 6-month lifetime. Upon recommendation from the Internet authorities, a draft may be published as a Request for Comment (RFC). Each RFC is edited, assigned a number, and made available to all interested parties. RFCs go through maturity levels and are categorized according to their requirement level.

**Chapter 2**

**NETWORK MODELS**

A network is a combination of hardware and software that sends data from one location to another. The hardware consists of the physical equipment that carries signals from one point of the network to another. The software consists of instruction sets that make possible the services that we expect from a network. For example, the task of sending an e-mail from one point in the world to another can be broken into several tasks, each performed by a separate software package. Each software package uses the services of another software package. At the lowest layer, a signal, or a set of signals, is sent from the source computer to the destination computer. In this chapter, we give a general idea of the layers of a network and discuss the functions of each.

**2.1 LAYERED TASKS**

We use the concept of layers in our daily life. As an example, let us consider two friends who communicate through postal mail. The process of sending a letter to a friend would be complex if there were no services available from the post office.

The layered model that dominated data communications and networking literature before 1990 was the Open Systems Interconnection (OSI) model. Everyone believed that the OSI model would become the ultimate standard for data communications, but this did not happen. The TCP/IP protocol suite became the dominant commercial architecture because it was used and tested extensively in the Internet; the OSI model was never fully implemented. In this chapter, first we briefly discuss the OSI model, and then we concentrate on TCP/IP as a protocol suite.

**2.2 THE OSI MODEL**

Established in 1947, the International Standards Organization (ISO) is a multinational body dedicated to worldwide agreement on international standards. An ISO standard that covers all aspects of network communications is the **Open Systems Interconnection model**. It was first introduced in the late 1970s. An open system is a set of protocols that allows any two different systems to communicate regardless of their underlying architecture. The OSI model is not a protocol; it is a model for understanding and designing a network architecture that is flexible, robust, and interoperable.

The OSI model is a layered framework for the design of network systems that allows communication between all types of computer systems. It consists of seven separate but related layers.

**Layered Architecture**

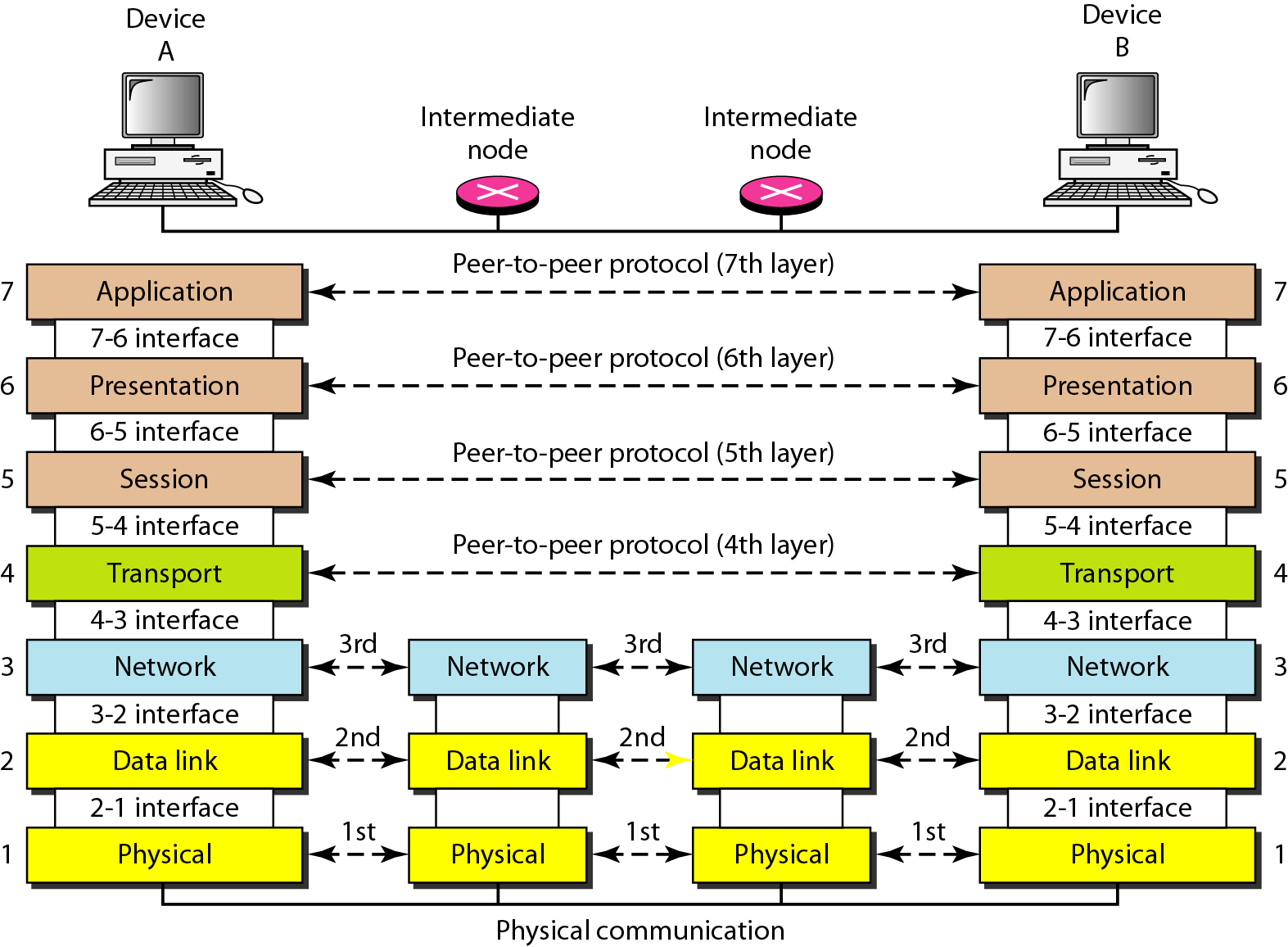
The OSI model is composed of seven ordered layers:

1. Physical (layer 1),
2. Data link (layer 2),
3. Network (layer 3),
4. Transport (layer 4),
5. Session (layer 5),
6. Presentation (layer 6), and
7. Application (layer 7).

As the message travels it may pass through many intermediate nodes. These intermediate nodes usually involve only the first three layers of the OSI model. They identified which networking functions had related uses and collected those functions into discrete groups that became the layers. Each layer defines a family of functions distinct from those of the other layers. By defining and localizing functionality in this fashion, the designers created an architecture that is both comprehensive and flexible. Most importantly, the OSI model allows complete interoperability between otherwise incompatible systems. Within a single machine, each layer calls upon the services of the layer just below it. Between machines, layer x on one machine communicates with layer x on another machine. This communication is governed by an agreed-upon series of rules and conventions called protocols. The processes on each machine that communicate at a given layer are called peer-to-peer processes. Communication between machines is therefore a peer-to-peer process using the protocols appropriate to a given layer.

**Peer-to-Peer Processes**

At the physical layer, communication is direct: device A sends a stream of bits to device B (through intermediate nodes). Communication must move down through the layers on device A, over to device B, and then back up through the layers. Each layer in the sending device adds its own information to the message it receives from the layer just above it and passes the whole package to the layer just below it. At layer I the entire package is converted to a form that can be transmitted to the receiving device. At the receiving machine, the message is unwrapped layer by layer, with each process receiving and removing the data meant for it.

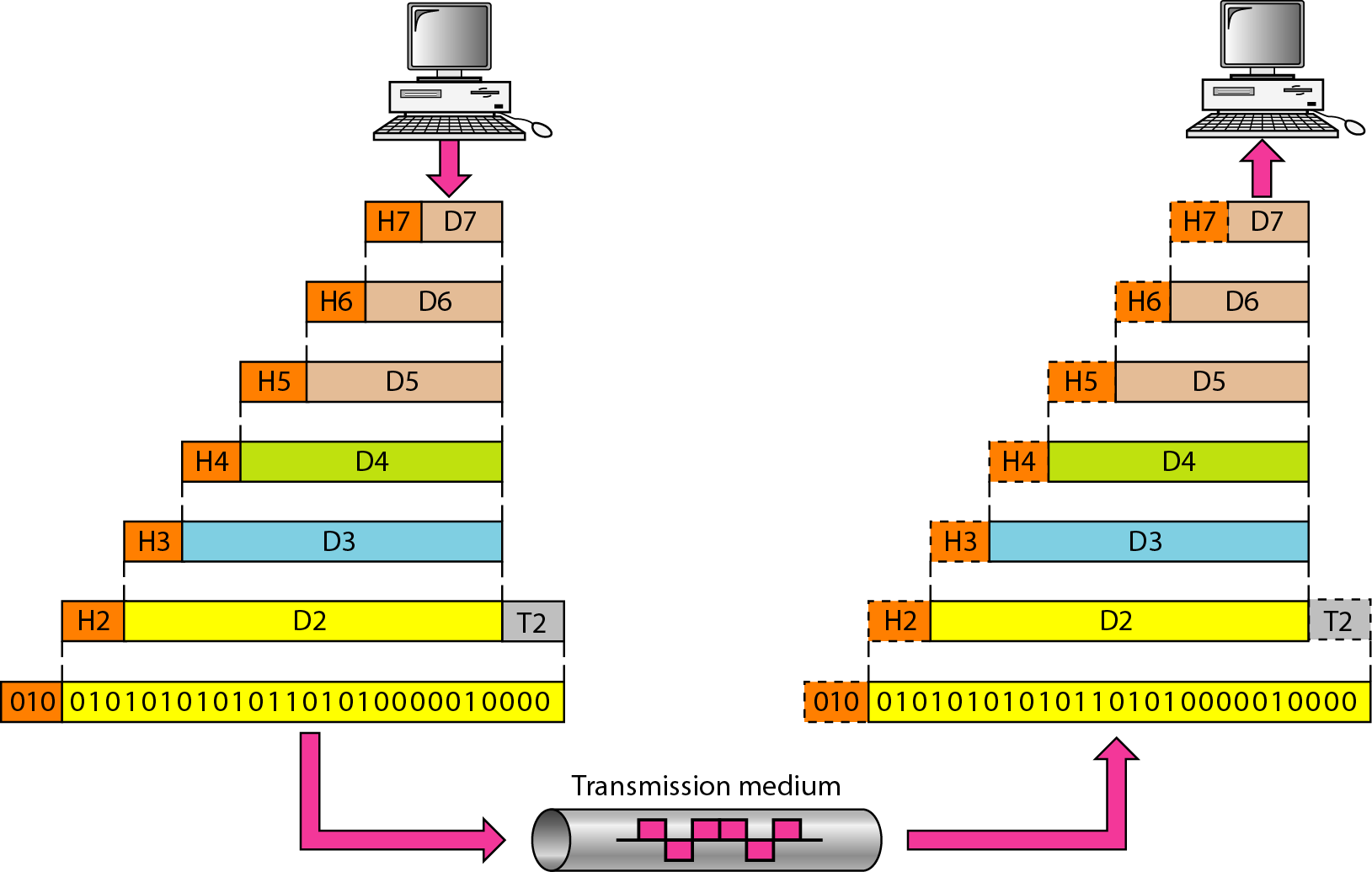


**Interfaces Between Layers**

The passing of the data and network information down through the layers of the sending device and back up through the layers of the receiving device is made possible by an interface between each pair of adjacent layers. Each interface defines the information and services a layer must provide for the layer above it. Well-defined interfaces and layer functions provide modularity to a network. As long as a layer provides the expected services to the layer above it, the specific implementation of its functions can be modified or replaced without requiring changes to the surrounding layers.

**Organization of the Layers**

The seven layers can be thought of as belonging to **three subgroups**. Layers I, 2, and 3-**physical, data link, and network-are** the **network support layers**; they deal with the physical aspects of moving data from one device to another (such as electrical specifications, physical connections, physical addressing, and transport timing and reliability). Layers 5, 6, and 7-**session, presentation, and application-are** be thought of as the **user support layers**; they allow interoperability among unrelated software systems. Layer 4, the transport layer, links the two subgroups and ensures that what the lower layers have transmitted is in a form that the upper layers can use. The upper OSI layers are almost always implemented in software; lower layers are a combination of hardware and software, except for the physical layer, which is mostly hardware. At each layer, a header, or possibly a trailer, can be added to the data unit. Commonly, the trailer is added only at layer 2.



**Encapsulation**

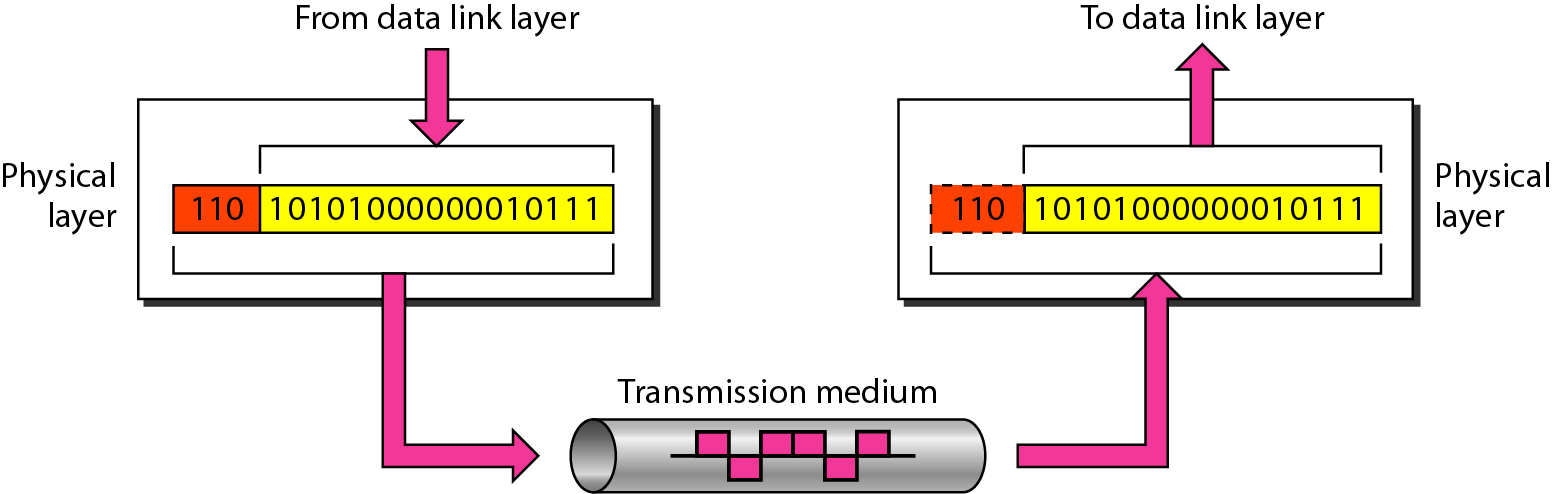
Figure 2.3 reveals another aspect of data communications in the OSI model: encapsulation. A packet (header and data) at level 7 is encapsulated in a packet at level 6. The whole packet at level 6 is encapsulated in a packet at level 5, and so on. In other words, the data portion of a packet at level N - 1 carries the whole packet (data and header and maybe trailer) from level N. The concept is called encapsulation; level N - 1 is not aware of which part of the encapsulated packet is data and which part is the header or trailer. For level N - 1, the whole packet coming from level N is treated as one integral unit.

**2.3 LAYERS IN THE OSI MODEL**

In this section we briefly describe the functions of each layer in the OSI model.

**I. Physical Layer**

The physical layer coordinates the functions required to carry a bit stream over a physical medium. It deals with the mechanical and electrical specifications of the interface and transmission medium. It also defines the procedures and functions that physical devices and interfaces have to perform for transmission to Occur. Figure shows the position of the physical layer with respect to the transmission medium and the data link layer.



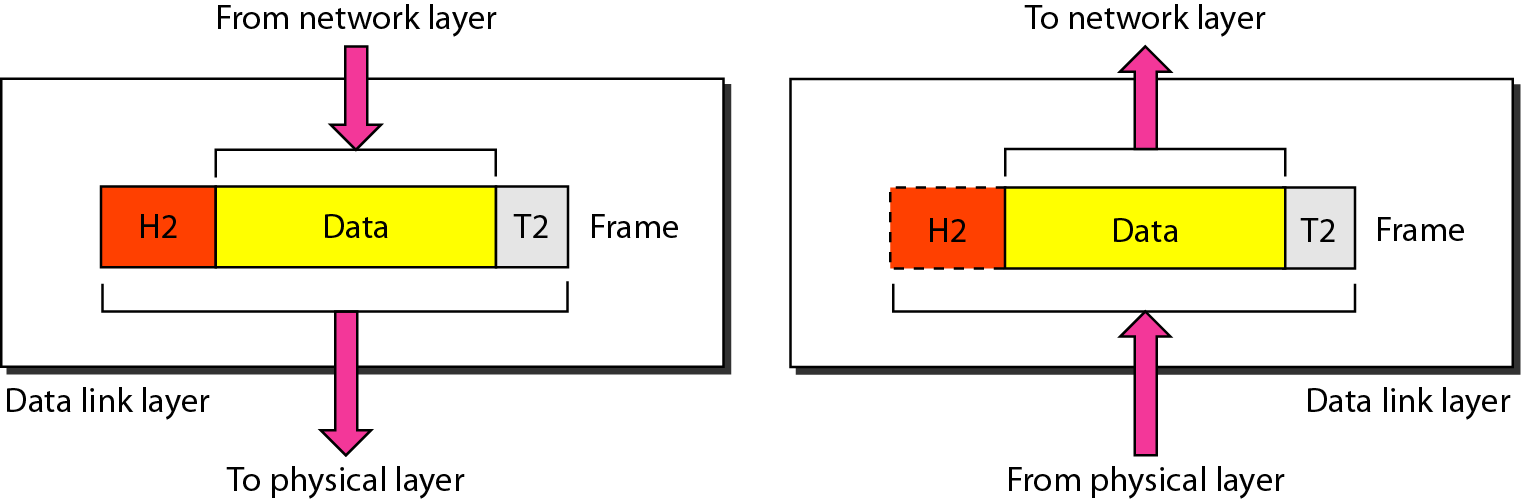
**The physical layer is responsible for movements of individual bits from one hop (node) to the next.**

The physical layer is also concerned with the following:

1. **Physical characteristics of interfaces and medium.**
2. **Representation of bits.**
3. **Data rate.**
4. **Synchronization of bits.**
5. **Line configuration.**
6. **Physical topology.**
7. **Transmission mode**.

**II. Data Link Layer**

The data link layer transforms the physical layer, a raw transmission facility, to a reliable link. It makes the physical layer appear error-free to the upper layer (network layer). Figure shows the relationship of the data link layer to the network and physical layers.

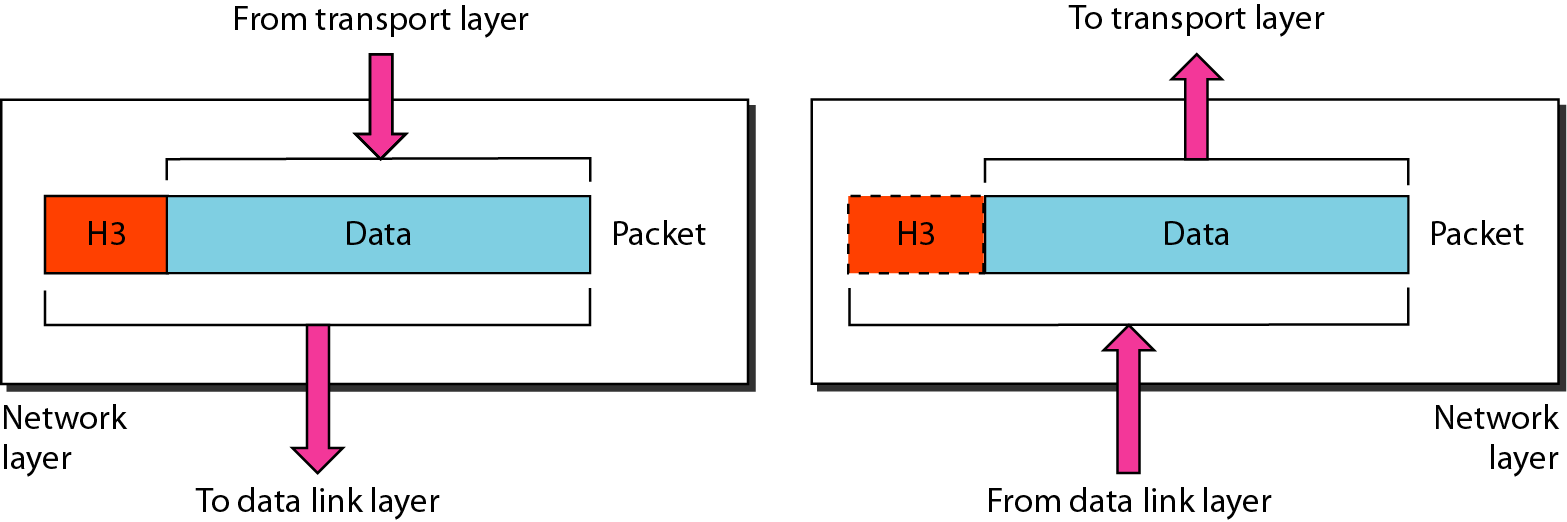


**The datalink layer is responsible for moving frames from one hop (node) to the next.**

Other responsibilities of the data link layer include the following:

1. **Framing**. The data link layer divides the stream of bits received from the network layer into manageable data units called frames.
2. **Physical addressing**. the data link layer adds a header to the frame to define the sender and/or receiver of the frame.
3. **Flow control.** The data link layer imposes a flow control mechanism to avoid overwhelming the receiver.
4. **Error control.** The data link layer adds reliability to the physical layer by adding mechanisms to detect and retransmit damaged or lost frames and it also uses to recognize duplicate frames. Error control is normally achieved through a trailer added to the end of the frame.
5. **Access control.** When two or more devices are connected to the same link, data link layer protocols are necessary to determine which device has control over the link at any given time.

**III. Network Layer**



The network layer is responsible for the source-to-destination delivery of a packet, possibly across multiple networks (links). The network layer ensures that each packet gets from its point of origin to its final destination. If two systems are connected to the same link, there is usually no need for a network layer. However, if the two systems are attached to different networks (links), there is often a need for the network layer to accomplish source-to-destination delivery. Figure shows the relationship of the network layer to the data link and transport layers.

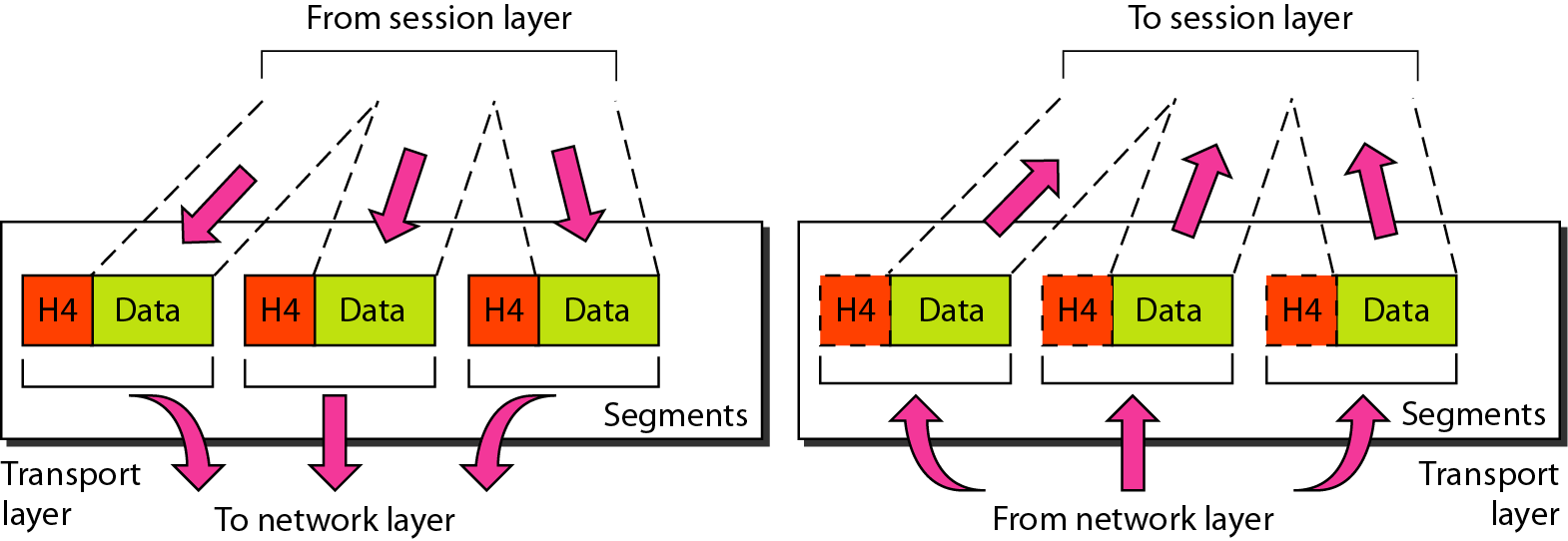
**The network layer is responsible for the delivery of individual packets from the source host to the destination host.**

Other responsibilities of the network layer include the following:

1. **Logical addressing.** The physical addressing implemented by the data link layer handles the addressing problem locally. If a packet passes the network boundary, we need another addressing system to help distinguish the source and destination systems. The network layer adds a header to the packet coming from the upper layer that, among other things, includes the logical addresses of the sender and receiver.
2. **Routing.** When independent networks or links are connected to create internetworks (network of networks) or a large network, the connecting devices (called routers or switches) route or switch the packets to their final destination. One of the functions of the network layer is to provide this mechanism.

**IV. Transport Layer**

The transport layer is responsible for process-to-process delivery of the entire message. A process is an application program running on a host. Whereas the network layer oversees source-to-destination delivery of individual packets, it does not recognize any relationship between those packets. It treats each one independently, as though each piece belonged to a separate message, whether or not it does. The transport layer, on the other hand, ensures that the whole message arrives intact and in order, overseeing both error control and flow control at the source-to-destination level. Figure shows the relationship of the transport layer to the network and session layers.



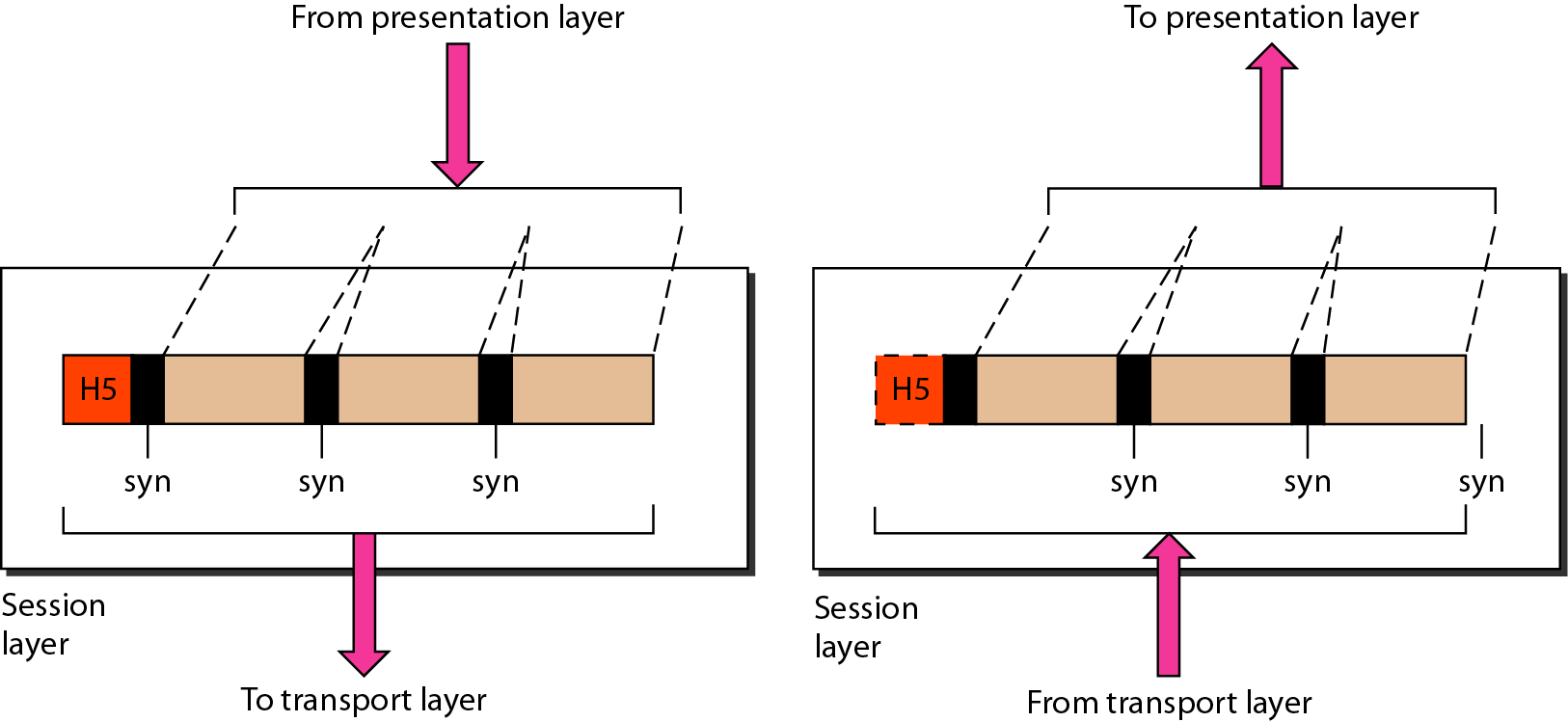
**The transport layer is responsible for the delivery of a message from one process to another.**

Other responsibilities of the transport layer include the following:

1. **Service-point addressing.** Computers often run several programs at the same time. The transport layer header must include a type of address called a service-point address (or port address). The network layer gets each packet to the correct computer; the transport layer gets the entire message to the correct process on that computer.
2. **Segmentation and reassembly.** A message is divided into transmittable segments, with each segment containing a sequence number. These numbers enable the transport layer to reassemble the message correctly upon arriving at the destination and to identify and replace packets that were lost in transmission.
3. **Connection control.** The transport layer can be either connectionless or connection oriented. A connectionless transport layer treats **each segment as an independent packet** and delivers it to the transport layer at the destination machine. A connection oriented transport layer **makes a connection with the transport layer** at the destination machine first before delivering the packets. After all the data are transferred, the connection is terminated.
4. **Flow control.** Like the data link layer, the transport layer is responsible for flow control. However, flow control at this layer is performed end to end rather than across a single link.
5. **Error control.** Like the data link layer, the transport layer is responsible for error control. However, error control at this layer is performed process-to-process rather than across a single link. The sending transport layer makes sure that the entire message arrives at the receiving transport layer without error (damage, loss, or duplication). Error correction is usually achieved through retransmission.

**V. Session Layer**

The services provided by the first three layers (physical, data link, and network) are not sufficient for some processes. The session layer is the network dialog controller. It establishes, maintains, and synchronizes the interaction among communicating systems.



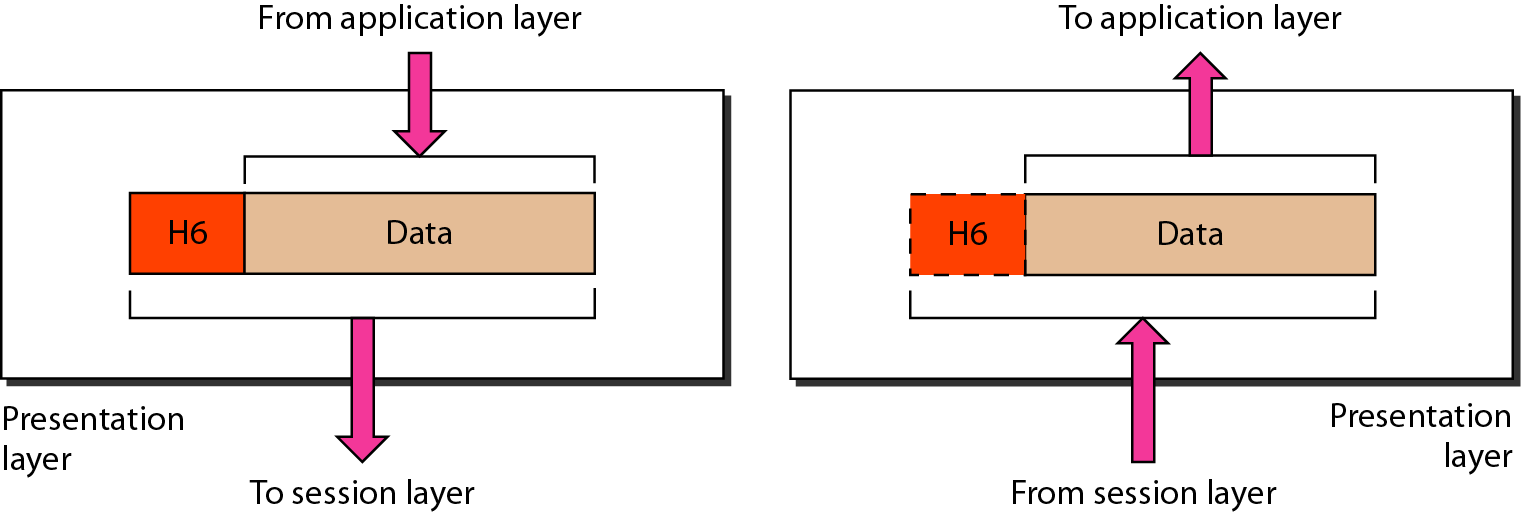
**The session layer is responsible for dialog control and synchronization.**

Specific responsibilities of the session layer include the following:

1. **Dialog control.** The session layer allows two systems to enter into a dialog. It allows the communication between two processes to take place in either half-duplex (one way at a time) or full-duplex (two ways at a time) mode.
2. **Synchronization.** The session layer allows a process to add checkpoints, or synchronization points, to a stream of data. For example, if a system is sending a file of 2000 pages, it is advisable to insert checkpoints after every 100 pages to ensure that each 100-page unit is received and acknowledged independently.

**VI. Presentation Layer**

The presentation layer is concerned with the syntax and semantics of the information exchanged between two systems.



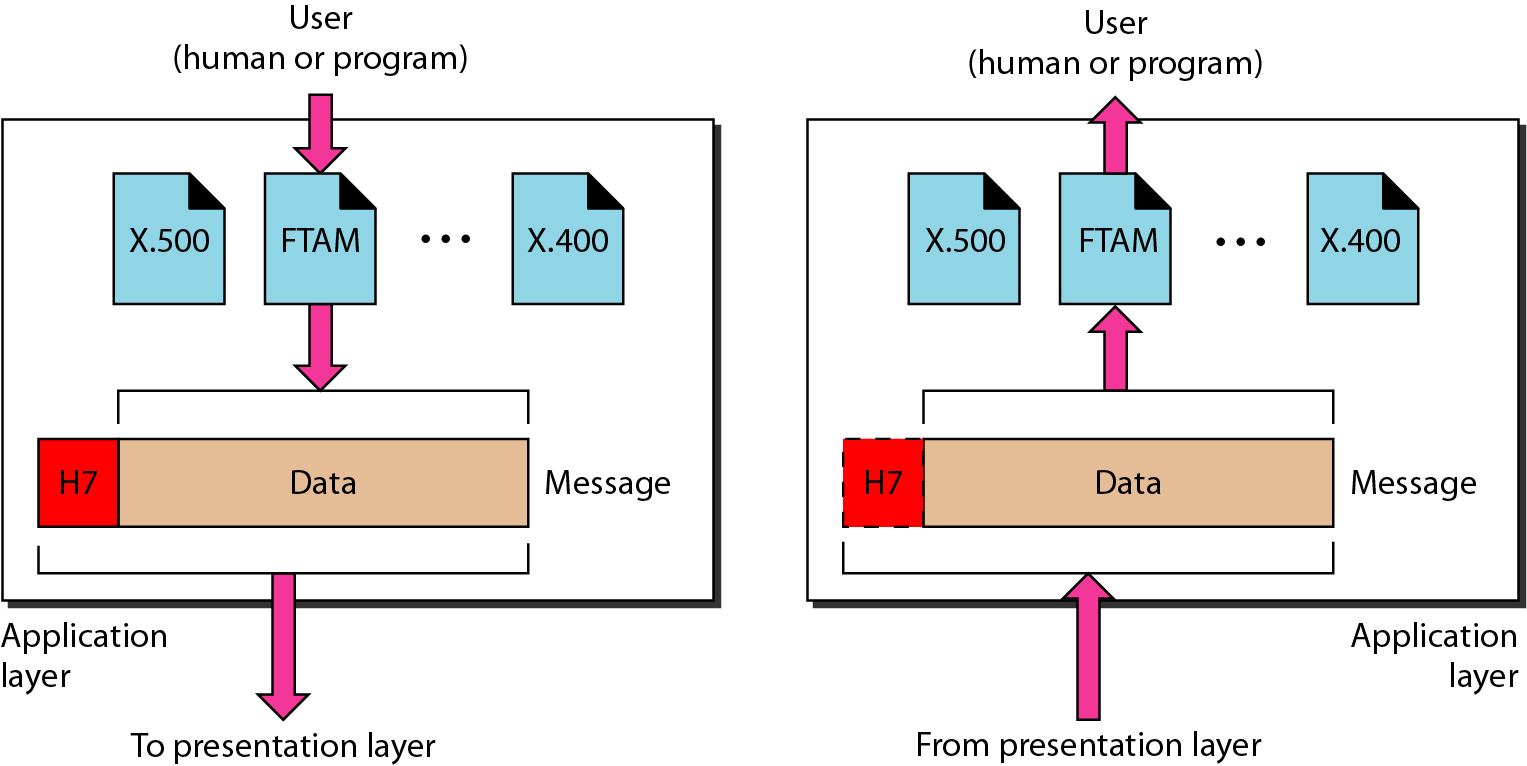
**The presentation layer is responsible for translation, compression, and encryption.**

Specific responsibilities of the presentation layer include the following:

1. **Translation.** The processes (running programs) in two systems are usually exchanging information in the form of character strings, numbers, and so on. The information must be changed to bit streams before being transmitted. Because different computers use different encoding systems, the presentation layer is responsible for interoperability between these different encoding methods. The presentation layer at the sender changes the information from its sender-dependent format into a common format. The presentation layer at the receiving machine changes the common format into its receiver-dependent format.
2. **Encryption.** To carry sensitive information, a system must be able to ensure privacy. **Encryption** means that the sender transforms the original information to another form and sends the resulting message out over the network. **Decryption** reverses the original process to transform the message back to its original form.
3. **Compression.** Data compression reduces the number of bits contained in the information. Data compression becomes particularly important in the transmission of multimedia such as text, audio, and video.

**VII. Application Layer**

The application layer enables the user, whether human or software, to access the network. It provides user interfaces and support for services such as electronic mail, remote file access and transfer, shared database management, and other types of distributed information services. XAOO (message-handling services), X.500 (directory services), and file transfer, access, and management (FTAM). The user in this example employs XAOO to send an e-mail message.

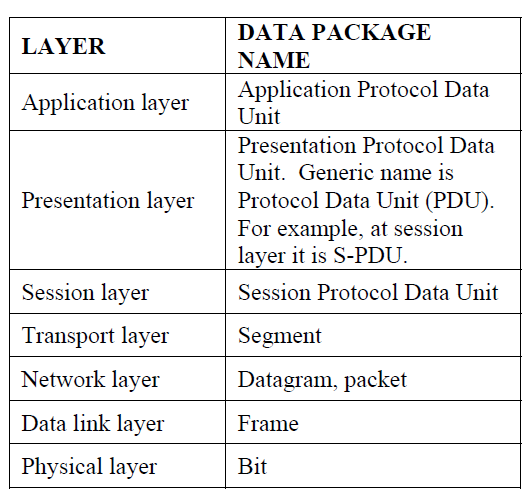


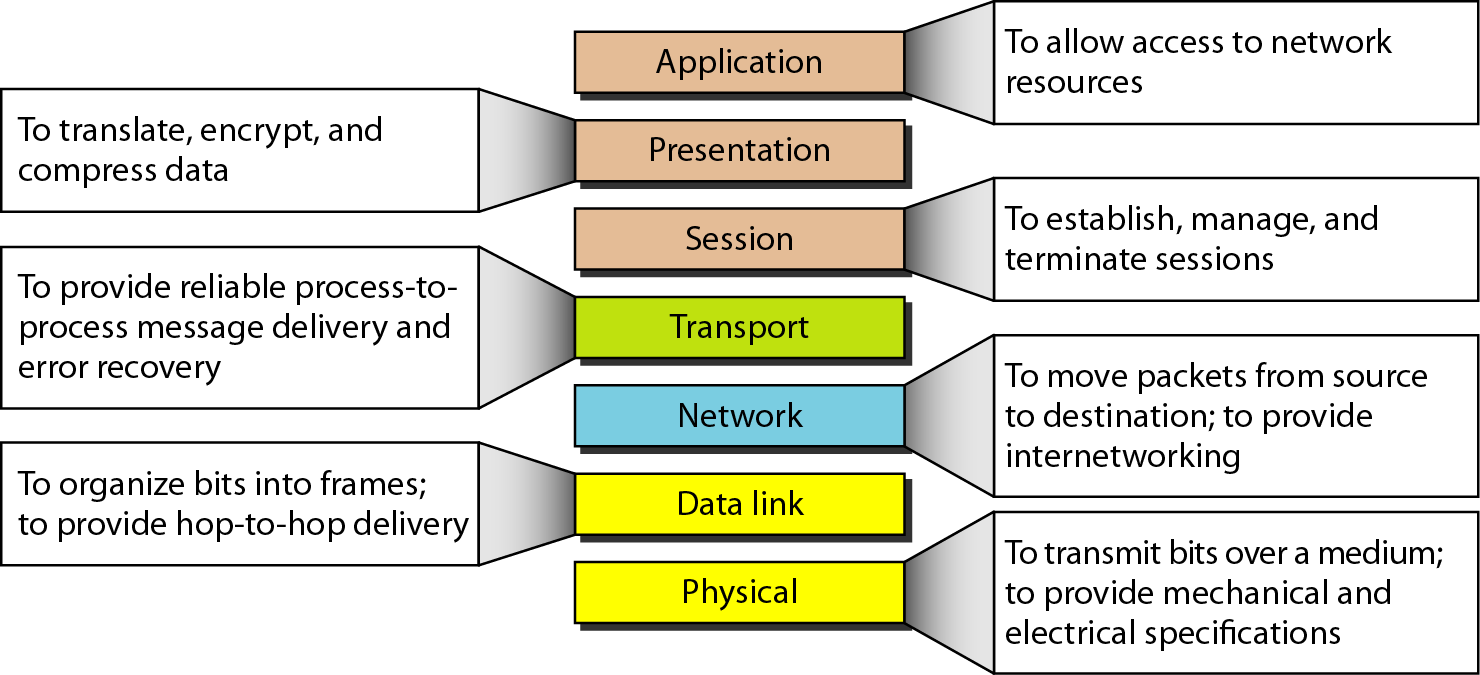
**The application layer is responsible for providing services to the user.**

Specific services provided by the application layer include the following:

1. **Network virtual terminal.** A network virtual terminal is a software version of a physical terminal, and it allows a user to log on to a remote host. To do so, the application creates a software emulation of a terminal at the remote host. The user's computer talks to the software terminal which, in turn, talks to the host, and vice versa. The remote host believes it is communicating with one of its own terminals and allows the user to log on.
2. **File transfer, access, and management.** This application allows a user to access files in a remote host (to make changes or read data), to retrieve files from a remote computer for use in the local computer, and to manage or control files in a remote computer locally.
3. **Mail services.** This application provides the basis for e-mail forwarding and storage.
4. **Directory services.** This application provides distributed database sources and access for global information about various objects and services.

**Summary of OSI Layers**

****

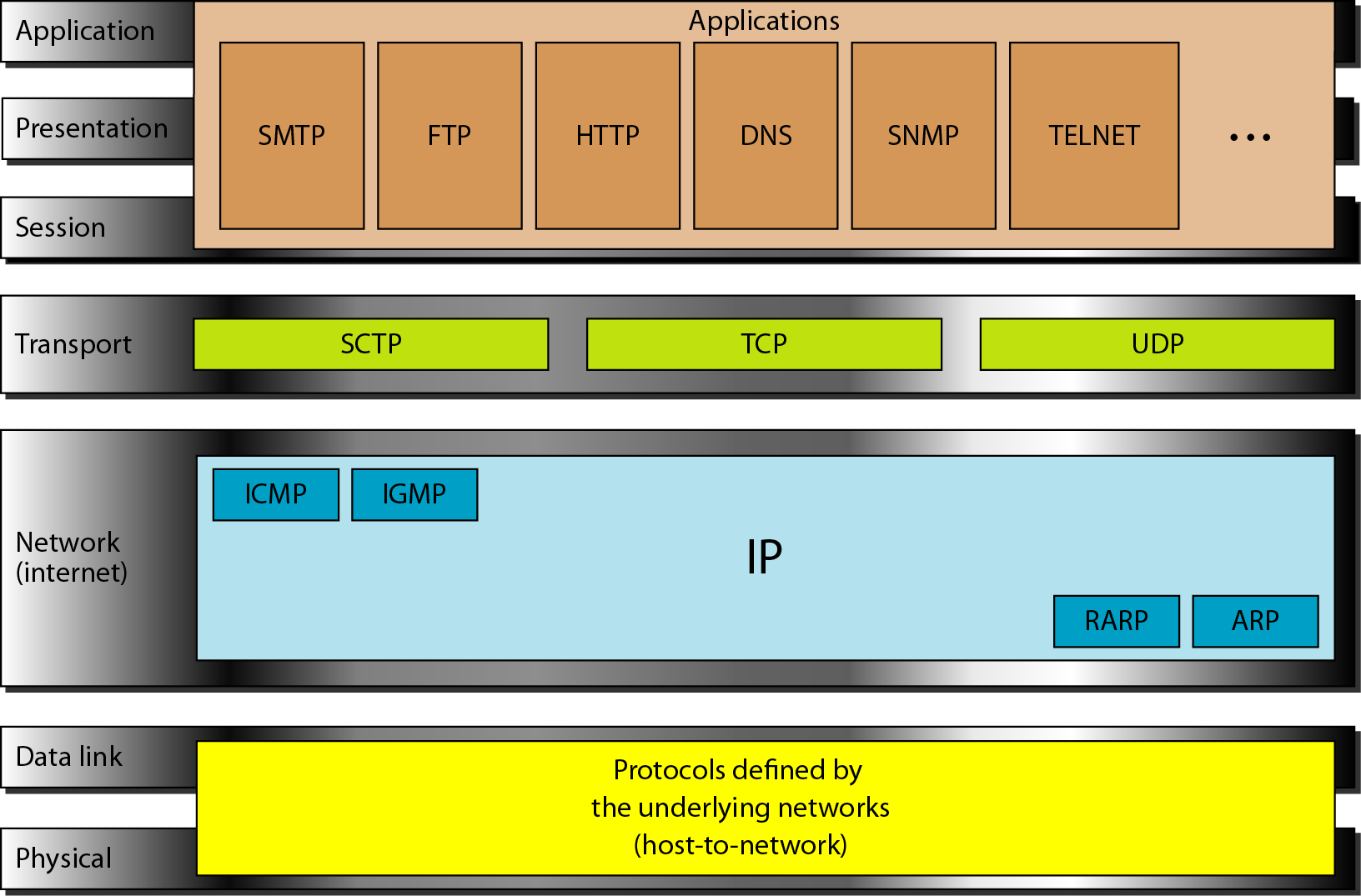


**2.4 TCP/IP PROTOCOL SUITE**

The TCPIIP protocol suite was developed prior to the OSI model. Therefore, the layers in the TCP/IP protocol suite do not exactly match those in the OSI model. The original TCP/IP protocol suite was defined as having **four layers:**

1. **Host-to-Network, (**Physical Layer & data link Layers in OSI**)**
2. **Internet, (**Network Layer in OSI**)**
3. **Transport, (**Transport Layerin OSI**) and**
4. **Application (**Session Layer, Presentation Layer, and Application Layers**)**

So in this book, we assume that the TCP/IP protocol suite is made of five layers: physical, data link, network, transport, and application. The first four layers provide physical standards, network interfaces, internetworking, and transport functions that correspond to the first four layers of the OSI model. The three topmost layers in the OSI model, however, are represented in TCP/IP by a single layer called the application layer.



TCP/IP is a hierarchical protocol made up of interactive modules, each of which provides a specific functionality; however, the modules are not necessarily interdependent. Whereas the OSI model specifies which functions belong to each of its layers, the layers of the TCP/IP protocol suite contain relatively independent protocols that can be mixed and matched depending on the needs of the system. The term hierarchical means that each upper-level protocol is supported by one or more lower-level protocols.

**I. Physical and Data Link Layers**

At the physical and data link layers, TCP/IP does **not define any specific protocol**. It supports all the standard and proprietary protocols. A network in a TCP/IP internetwork can be a local-area network or a wide-area network.

**II. Network Layer**

At the network layer (or, more accurately, the internetwork layer), TCP/IP supports the Internetworking Protocol. IP, in turn, uses four supporting protocols: ARP, RARP, ICMP, and IGMP. Each of these protocols is described in greater detail in later.

1. **Internetworking Protocol (IP)**: - is the transmission mechanism used by the TCP/IP protocols. It is an unreliable and connectionless protocol, a best-effort delivery service. The term best effort means that IP provides no error checking or tracking. IP assumes the unreliability of the underlying layers and does its best to get a transmission through to its destination, but with no guarantees. IP transports data in packets called datagrams, each of which is transported separately. Datagrams can travel along different routes and can arrive out of sequence or be duplicated. IP does not keep track of the routes and has no facility for reordering datagrams once they arrive at their destination. The limited functionality of IP should not be considered a weakness, however. IP provides bare-bones transmission functions that free the user to add only those facilities necessary for a given application and thereby allows for maximum efficiency.
2. **Address Resolution Protocol(ARP): -** is used to associate a logical address with a physical address. On a typical physical network, such as a LAN, each device on a link is identified by a physical or station address, usually imprinted on the network interface card (NIC). ARP is used to find the physical address of the node when its Internet address is known.
3. **Reverse Address Resolution Protocol(RARP): -** allows a host to discover its Internet address when it knows only its physical address. It is used when a computer is connected to a network for the first time or when a diskless computer is booted.
4. **Internet Control Message Protocol(ICMP)**: - is a mechanism used by hosts and gateways to send notification of datagram problems back to the sender. ICMP sends query and error reporting messages.
5. **Internet Group Message Protocol(IGMP): -** is used to facilitate the simultaneous transmission of a message to a group of recipients.

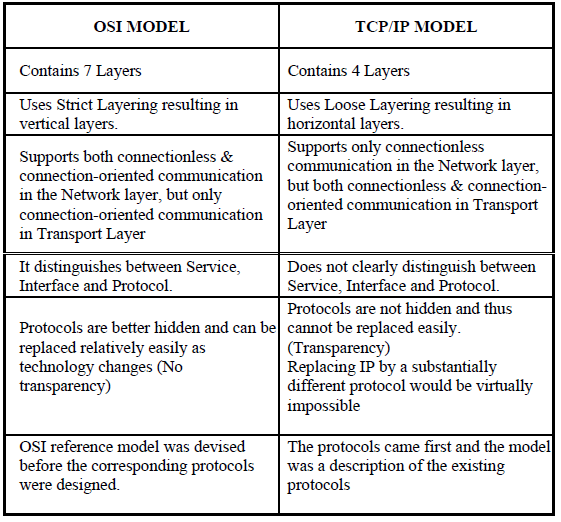
**III. Transport Layer**

Traditionally the transport layer was represented in TCP/IP by two protocols: TCP and UDP. IP is a host-to-host protocol, meaning that it can deliver a packet from one physical device to another. UDP and TCP are transport level protocols responsible for delivery of a message from a process (running program) to another process. A new transport layer protocol, SCTP, has been devised to meet the needs of some newer applications.

1. **User Datagram Protocol(UDP): -** is the simpler of the two standard TCP/IP transport protocols. It is a process-to-process protocol that adds only port addresses, checksum error control, and length information to the data from the upper layer.
2. **Transmission Control Protocol(TCP): -** provides full transport-layer services to applications. TCP is a reliable stream transport protocol. The term stream, in this context, means connection-oriented: A connection must be established between both ends of a transmission before either can transmit data. At the sending end of each transmission, TCP divides a stream of data into smaller units called segments. Each segment includes a sequence number for reordering after receipt, together with an acknowledgment number for the segments received. Segments are carried across the internet inside of IP datagrams. At the receiving end, TCP collects each datagram as it comes in and reorders the transmission based on sequence numbers.
3. **Stream Control Transmission Protocol(SCTP): -** provides support for newer applications such as voice over the Internet. It is a transport layer protocol that combines the best features of UDP and TCP.

**IV. Application Layer**

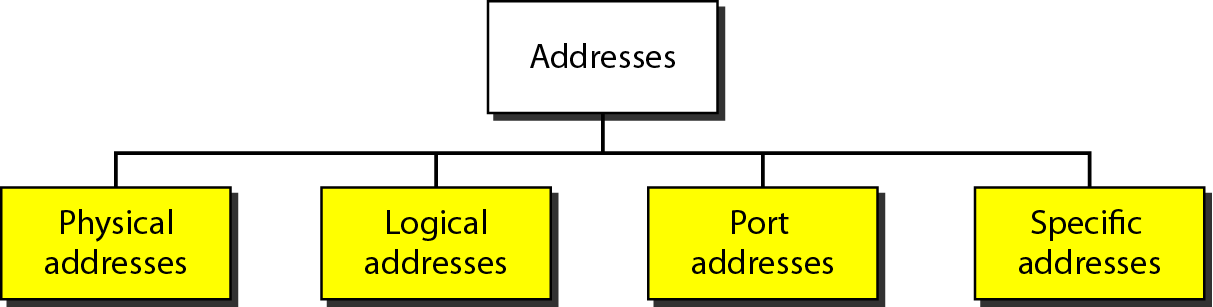
The application layer in TCP/IP is equivalent to the combined session, presentation, and application layers in the OSI model. Many protocols are defined at this layer.



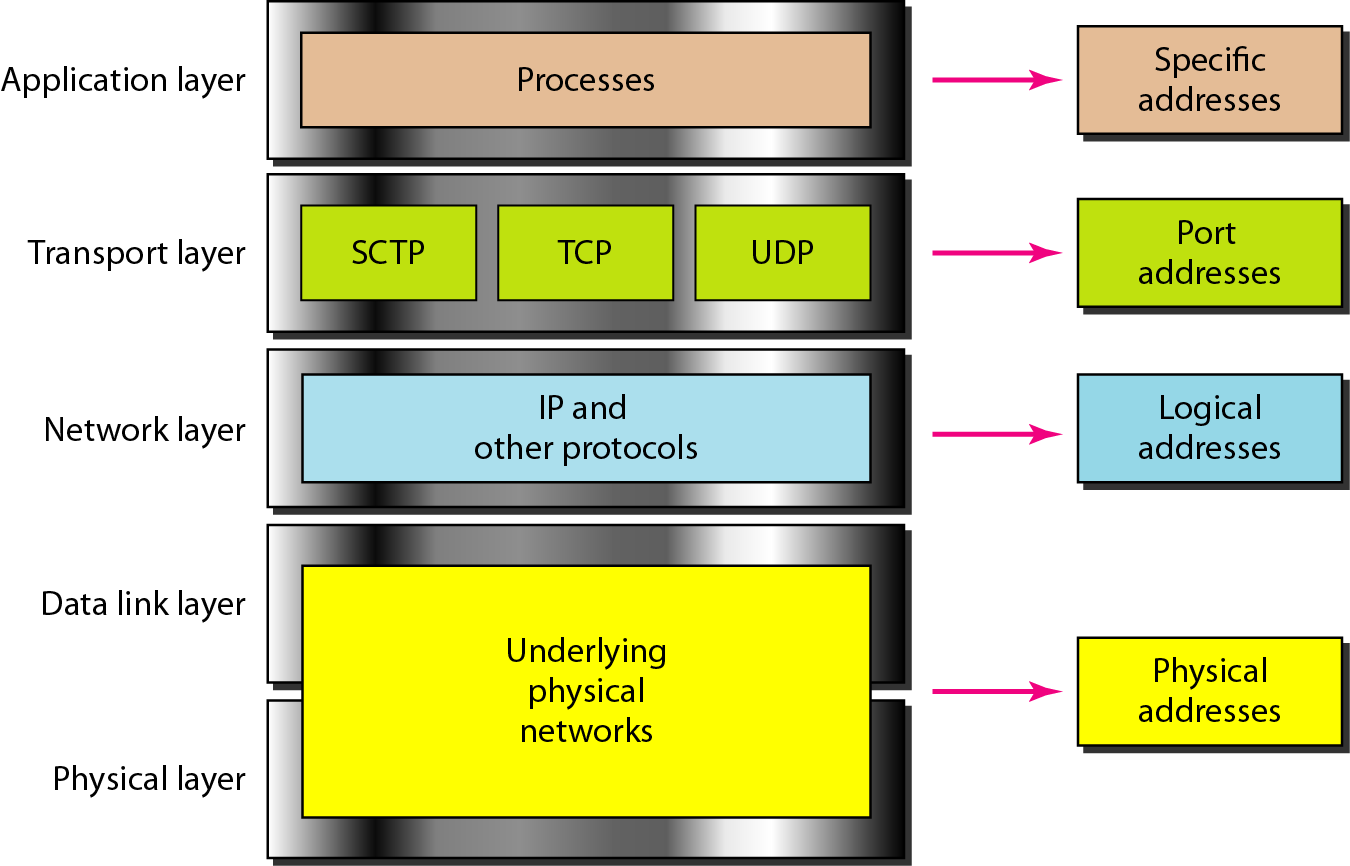
**2.5 ADDRESSING**

Four levels of addresses are used in an internet employing the TCP/IP protocols:

1. physical (link) addresses,
2. logical (IP) addresses,
3. port addresses, and
4. specific addresses

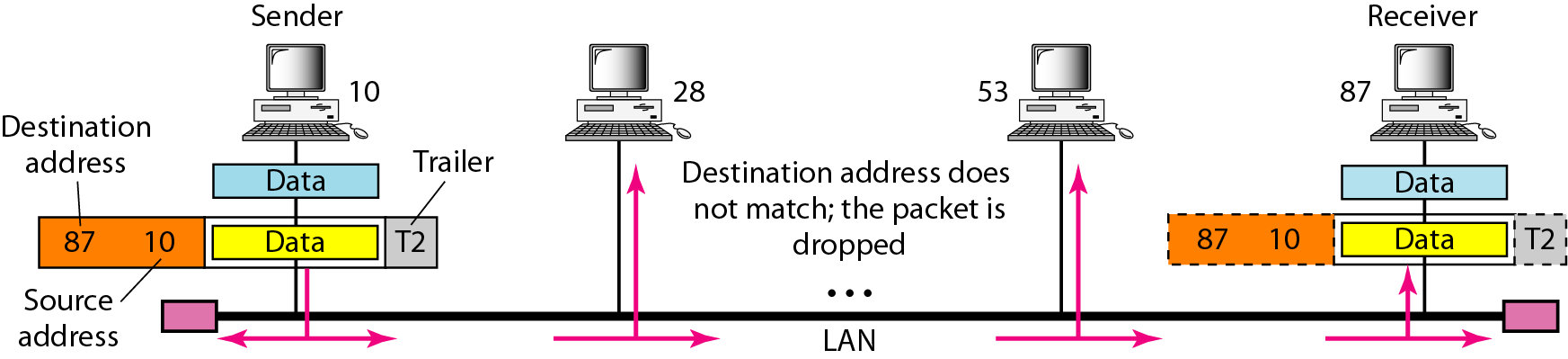


Each address is related to a specific layer in the TCPIIP architecture, as shown in Figure



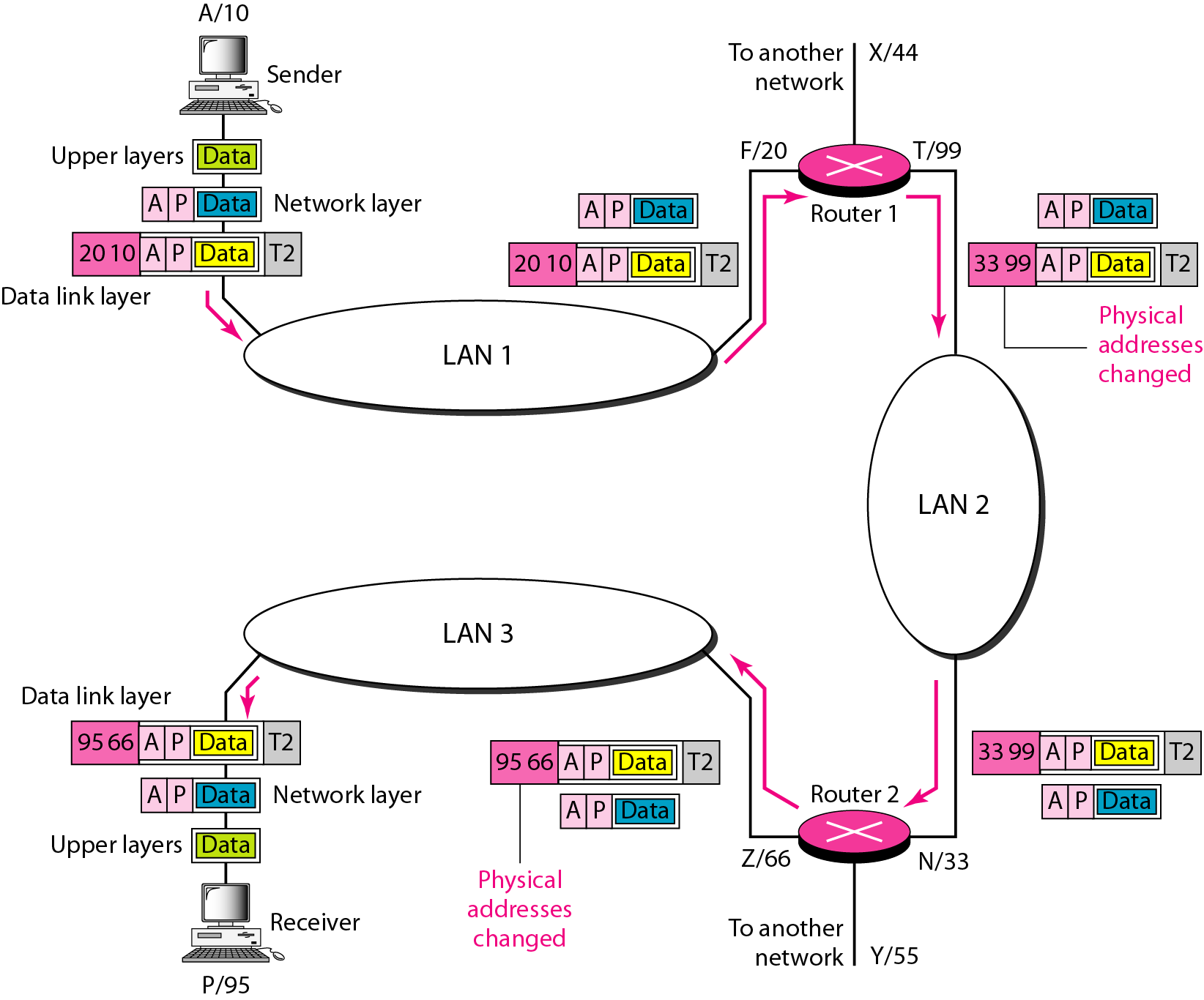
**I. Physical Addresses**

The physical address, also known as the link address or media access control address (**MAC**), is the address of a node as defined by its LAN or WAN. It is included in the frame used by the data link layer. It is the lowest-level address. The physical addresses have authority over the network (LAN or WAN). The size and format of these addresses vary depending on the network. For example, Ethernet uses a 6-byte (48-bit) physical address that is imprinted on the network interface card (NIC). LocalTalk (Apple), however, has a I-byte dynamic address that changes each time the station comes up.

****

**II. Logical Addresses**

Logical addresses (IP) are necessary for universal communications that are independent of underlying physical networks. Physical addresses are not adequate in an internetwork environment where different networks can have different address formats. A universal addressing system is needed in which each host can be identified uniquely, regardless of the underlying physical network. The logical addresses are designed for this purpose. No two publicly addressed and visible hosts on the Internet can have the same IP address.

****

*Figure shows a part of an internet with two routers connecting three LANs. Each device (computer or router) has a pair of addresses (logical and physical) for each connection. In this case, each computer is connected to only one link and therefore has only one pair of addresses. Each router, however, is connected to three networks (only two are shown in the figure). So each router has three pairs of addresses, one for each connection.*

**III. Port Addresses**

Arrival at the destination host is not the final objective of data communications on the Internet. The end objective of Internet communication is a process communicating with another process. Multiple processes are run at the same time, to receive data simultaneously, we need a method to label the different processes. In other words, they need addresses. In the TCP/IP architecture, the label assigned to a process is called a port address. A port address in TCP/IP is 16 bits in length.

**IV. Specific Addresses**

Some applications have user-friendly addresses that are designed for that specific address. Examples include the e-mail address and the Universal Resource Locator (URL). The first defines the recipient of an e-mail; the second is used to find a document on the World Wide Web.

**Part 2**

**Physical Layer and Media**

We start the discussion of the Internet model with the bottom-most layer, the physical layer. It is the layer that actually interacts with the transmission media, the physical part of the network that connects network components together. This layer is involved in physically carrying information from one node in the network to the next.

The physical layer has complex tasks to perform. One **major task** is to **provide services for the data link** layer. The data in the data link layer consists of 0s and 1s organized into frames that are ready to be sent across the transmission medium. This stream of 0s and 1s must first be converted into another entity: **signals**. One of the services provided by the physical layer is to create a signal that represents this stream of bits.

The physical layer must also take care of the physical network, the transmission medium. The transmission medium is a passive entity; it has no internal program or logic for control like other layers. The transmission medium must be controlled by the physical layer. The physical layer decides on the directions of data flow. The physical layer decides on the number of logical channels for transporting data coming from different sources.

**Part 2 of the book is devoted to the physical layer and the transmission media.**

**Chapter 3** discusses the relationship between data, which are created by a device, and electromagnetic signals, which are transmitted over a medium.

**Chapter 4** deals with digital transmission. We discuss how we can covert digital or analog data to digital signals.

**Chapter 5** deals with analog transmission. We discuss how we can covert digital or analog data to analog signals.

**Chapter 6** shows how we can use the available bandwidth efficiently. We discuss two separate, but related topics, multiplexing and spreading.

**Chapter 7** After explaining some ideas about data and signals and how we can use them efficiently, we discuss the characteristics of transmission media, both guided and unguided, in this chapter. Although transmission media operates under the physical layer, they are controlled by the physical layer.

**Chapter 8** Although the previous chapters in this part are issues related to the physical layer or transmission media, Chapter 8 discusses switching, a topic that can be related to several layers. We have included this topic in this part of the book to avoid repeating the discussion for each layer.

**Chapter 9** shows how the issues discussed in the previous chapters can be used in actual networks. In this chapter, we first discuss the telephone network as designed to carry voice. We then show how it can be used to carry data. Second, we discuss the cable network as a television network. We then show how it can also be used to carry data.

**CHAPTER 3**

**DATA AND SIGNALS**

One of the major functions of the physical layer is to move data in the form of electromagnetic signals across a transmission medium. Whether you are collecting numerical statistics from another computer, sending animated pictures from a design workstation, or causing a bell to ring at a distant control canter, you are working with the transmission of data across network connections. Generally, the data usable to a person or application are not in a form that can be transmitted over a network. For example, a photograph must first be changed to a form that transmission media can accept. Transmission media work by conducting energy along a physical path.

**To be transmitted, data must be transformed to electromagnetic signals.**

**3.1 ANALOG AND DIGITAL**

Both data and the signals that represent them can be either analog or digital in form.

**Analog and Digital Data**

Data can be analog or digital. The term **analog data** refers to information that is continuous; **digital data** refers to information that has discrete states.

Data can be analog or digital. Analog data are continuous and take continuous values. Digital data have discrete states and take discrete values (0 or 1).

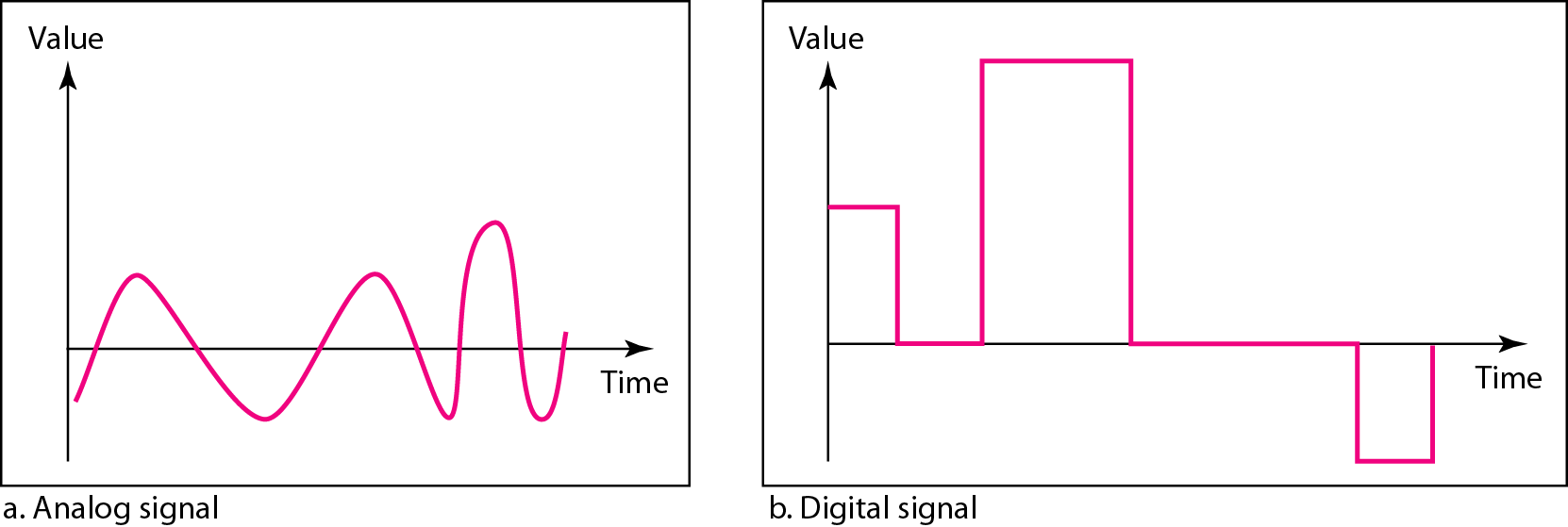
**“Data can be analog or digital. Analog data are continuous and take continuous values. Digital data have discrete states and take discrete values”.**

**Analog and Digital Signals**

Like the data they represent, signals can be either analog or digital. An **analog signal** has infinitely many levels of intensity over a period of time. A **digital signal**, on the other hand, can have only a limited number of defined values. Although each value can be any number, it is often as simple as 1 and 0.

The simplest way to show signals is by plotting them on a pair of perpendicular axes. The vertical axis represents the value or strength of a signal. The horizontal axis represents time. The curve representing the analog signal passes through an infinite number of points. The vertical lines of the digital signal,

**“Signals can be analog or digital. Analog signals can have an infinite number of values in a range; digital signals can have only a limited number of values”.**

****

**Periodic and Non-periodic Signals**

A **periodic signal** completes a pattern within a measurable time frame, called a period, and repeats that pattern over subsequent identical periods. The completion of one full pattern is called a **cycle**.

A **non-periodic signal** changes without exhibiting a pattern or cycle that repeats over time. Both analog and digital signals can be periodic or non-periodic.

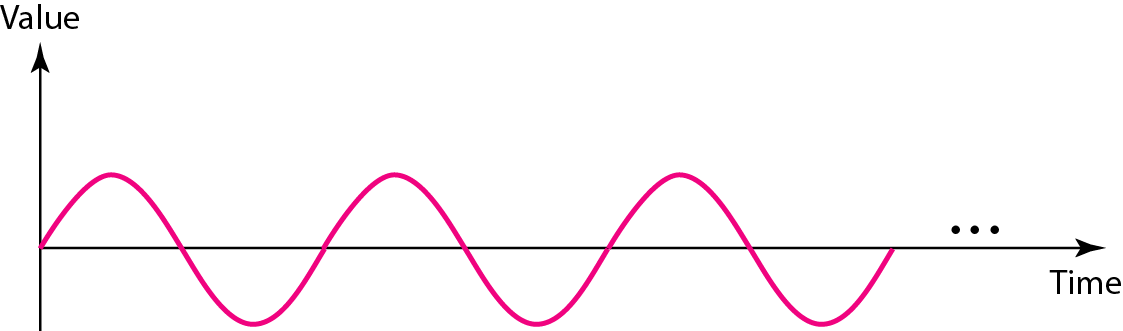
**“In data communications, we commonly use periodic analog signals and non-periodic digital signals”.**

**PERIODIC ANALOG SIGNALS**

Periodic analog signals can be classified as simple or composite. A simple periodic analog signal, a sine wave, cannot be decomposed into simpler signals. A composite periodic analog signal is composed of multiple sine waves.

**Sine Wave**

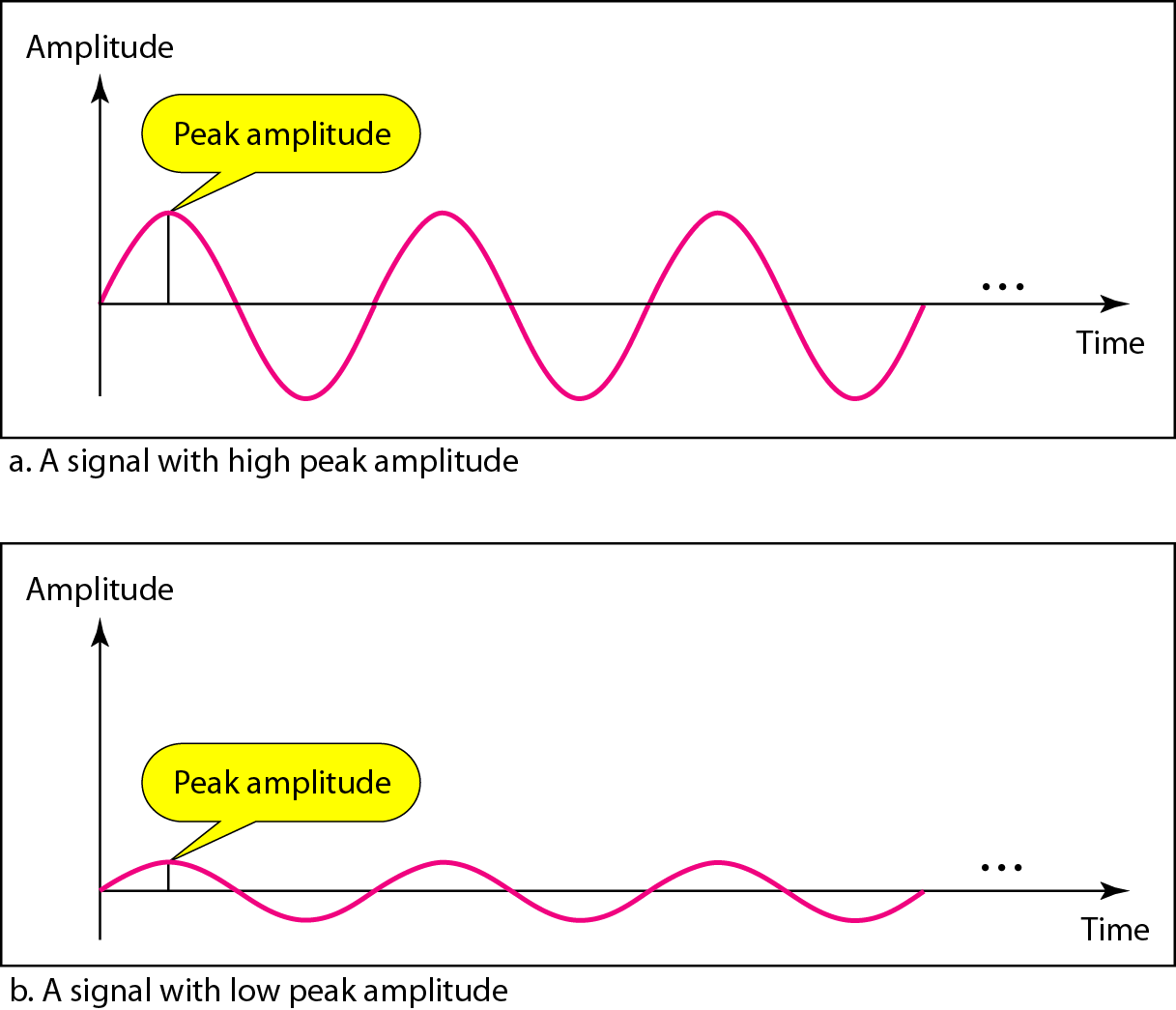
The sine wave is the most fundamental form of a periodic analog signal. When we visualize it as a simple oscillating curve, its change over the course of a cycle is smooth and consistent, a continuous, rolling flow. Figure shows a sine wave. Each cycle consists of a single arc above the time axis followed by a single arc below it.



A sine wave can be represented by three parameters: the *peak amplitude,* the *frequency,* and the *phase.* These three parameters fully describe a sine wave.

***Peak Amplitude***

The peak amplitude of a signal is the absolute value of its highest intensity, proportional to the energy it carries. For electric signals, peak amplitude is normally measured in *volts.* Figure shows two signals and their peak amplitudes.

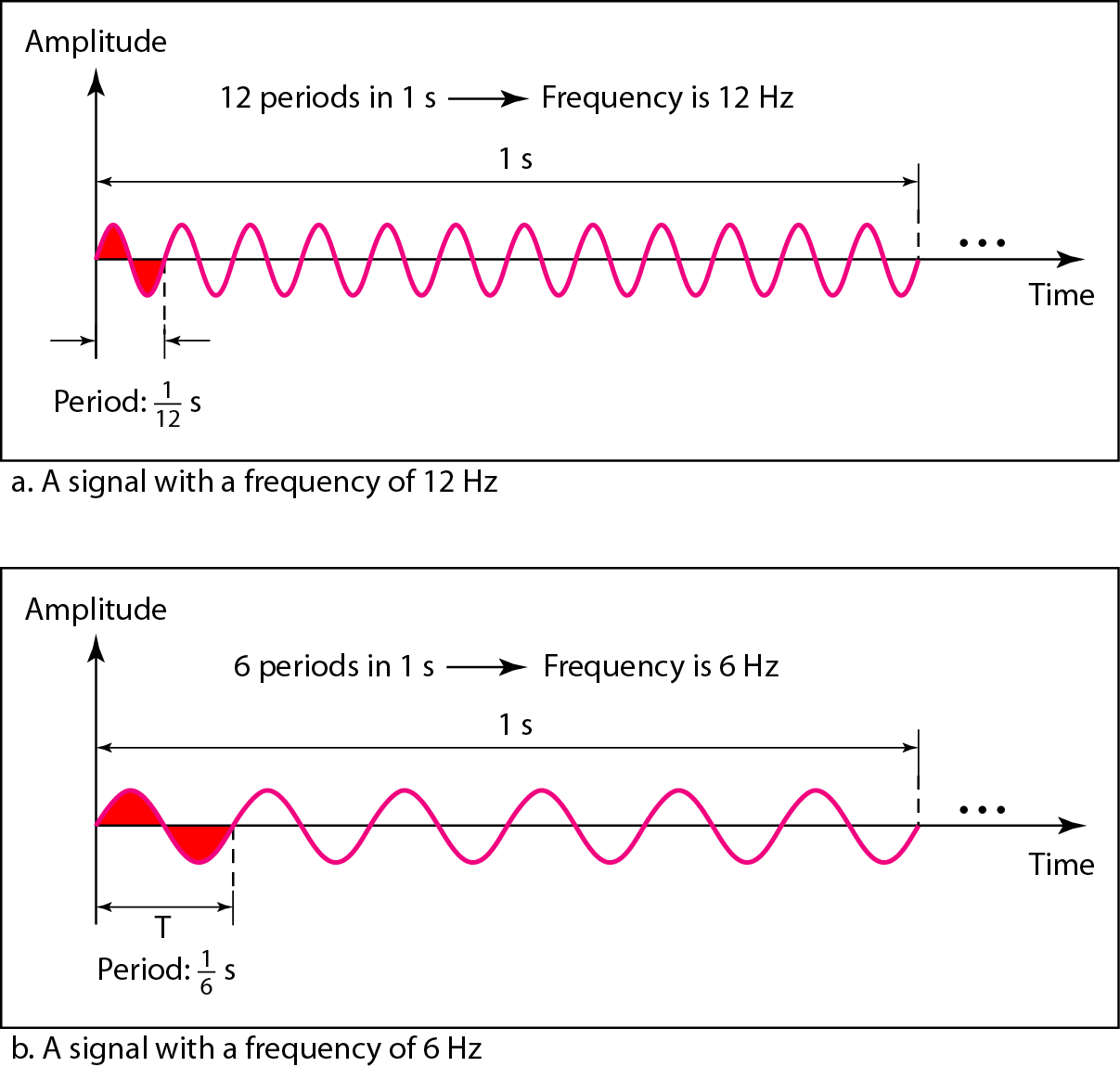
****

*Period and Frequency*

Period refers to the amount of time, in seconds, a signal needs to complete 1 cycle. Frequency refers to the number of periods in I s. Note that period and frequency are just one characteristic defined in two ways. Period is the inverse of frequency, and frequency is the inverse of period, as the following formulas show.

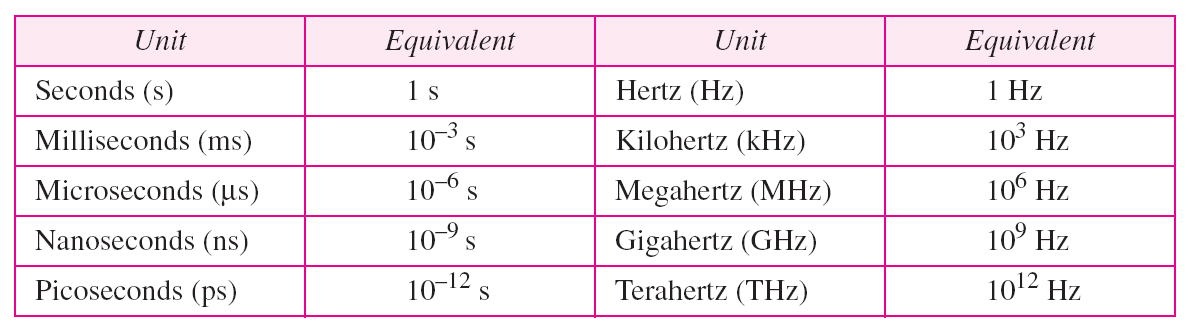


***Two signals with the same amplitude and phase, but different frequencies***



**Period is formally expressed in seconds. Frequency is formally expressed in Hertz (Hz), which is cycle per second.**

Units of period and frequency are shown in Table 3.1.

****

***Example***

***The period of a signal is 100 ms. What is its frequency in kilohertz?***

Solution

***First we change 100 ms to seconds, and then we calculate the frequency from the period (1 Hz = 10−3 kHz).***



***More About Frequency***

We already know that frequency is the relationship of a signal to time and that the frequency of a wave is the number of cycles it completes in 1 s. But another way to look at frequency is as a measurement of the rate of change. Electromagnetic signals are oscillating waveforms; that is, they fluctuate continuously and predictably above and below a mean energy level. A 40-Hz signal has one-half the frequency of an 80-Hz signal; it completes 1 cycle in twice the time of the 80-Hz signal, so each cycle also takes twice as long to change from its lowest to its highest voltage levels. Frequency, therefore, though described in cycles per second (hertz), is a general measurement of the rate of change of a signal with respect to time.

**Frequency is the rate of change with respect to time. Change in a short span of time means high frequency. Change over a long span of time means low frequency.**

If the value of a signal changes over a very short span of time, its frequency is high. If it changes over a long span of time, its frequency is low.

**Two Extremes**

What if a signal does not change at all? What if it maintains a constant voltage level for the entire time it is active? In **such a case, its frequency is zero**. Conceptually, this idea is a simple one. If a signal does not change at all, it never completes a cycle, so its frequency is a Hz. But what if a signal changes instantaneously? What if it jumps from one level to another in no time? **Then its frequency is infinite**. In other words, when a signal changes instantaneously, its period is zero; since frequency is the inverse of period, in this case, the frequency is 1/0, or infinite (unbounded).

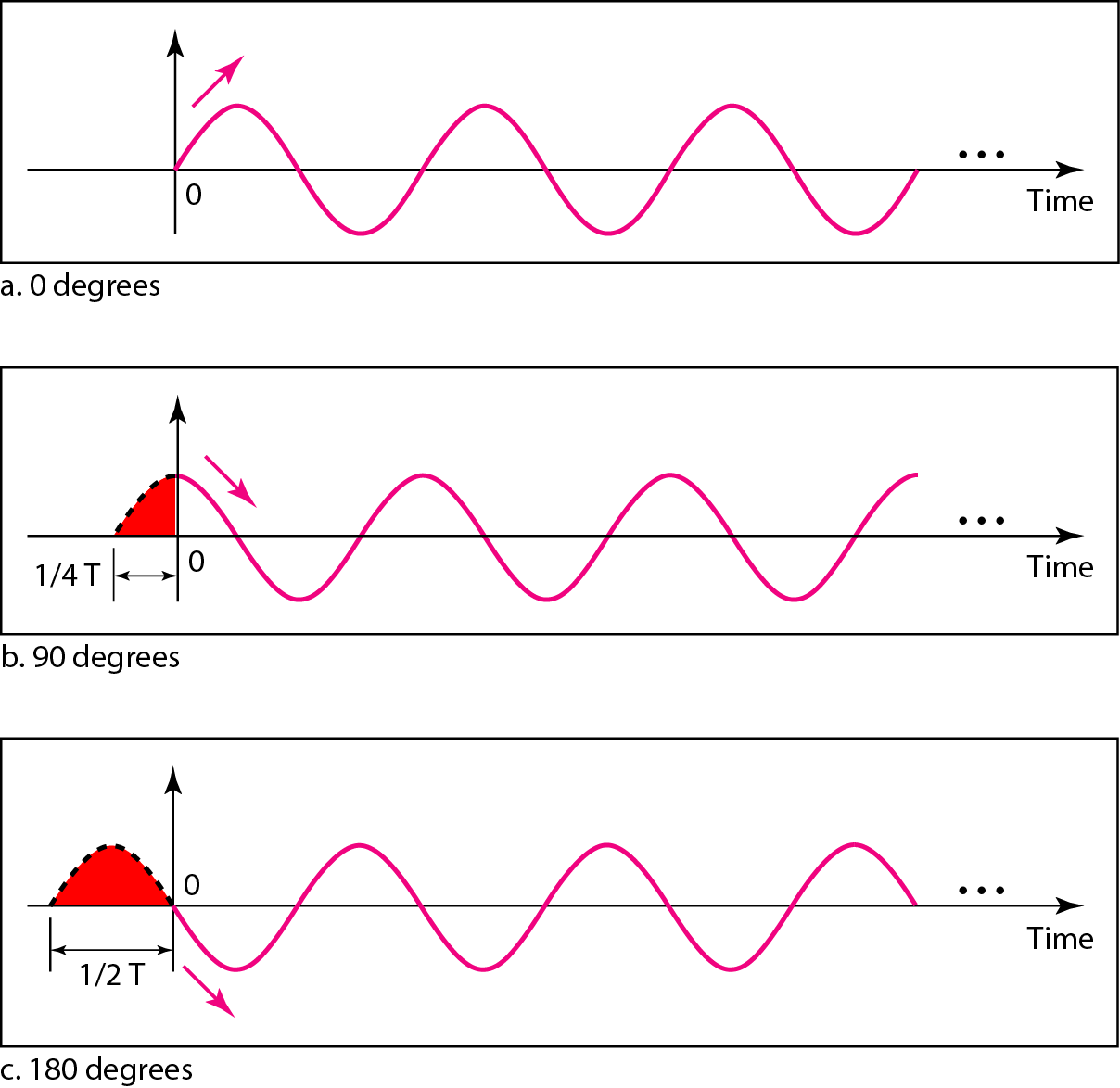
**If a signal does not change at all, its frequency is zero. If a signal changes instantaneously, its frequency is infinite.**

**Phase**

The term phase describes the position of the waveform relative to time O. If we think of the wave as something that can be shifted backward or forward along the time axis, phase describes the amount of that shift. It indicates the status of the first cycle.

**Phase describes the position of the waveform relative to time O.**

Phase is measured in degrees or radians [360° is 2n rad; 1° is 2n/360 rad, and 1 rad is 360/(2n)]. A phase shift of 360° corresponds to a shift of a complete period; a phase shift of 180° corresponds to a shift of one-half of a period; and a phase shift of 90° corresponds to a shift of one-quarter of a period.



we can say that

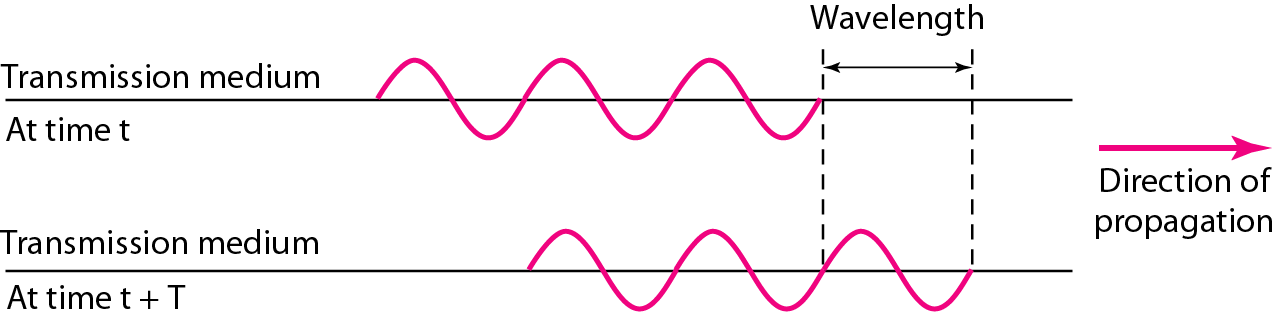
1. A sine wave with a phase of 0° starts at time 0 with a zero amplitude. The amplitude is increasing.
2. A sine wave with a phase of 90° starts at time 0 with a peak amplitude. The amplitude is decreasing.
3. A sine wave with a phase of 180° starts at time 0 with a zero amplitude. The amplitude is decreasing.

Another way to look at the phase is in terms of shift or offset. We can say that:

1. A sine wave with a phase of 0° is not shifted.
2. A sine wave with a phase of 90° is shifted to the left by ¼ cycle. However, note that the signal does not really exist before time 0.
3. A sine wave with a phase of 180° is shifted to the left by ½ cycle. However, note that the signal does not really exist before time 0.

**Wavelength**

Wavelength is another characteristic of a signal traveling through a transmission medium. Wavelength binds the period or the frequency of a simple sine wave to the propagation speed of the medium.

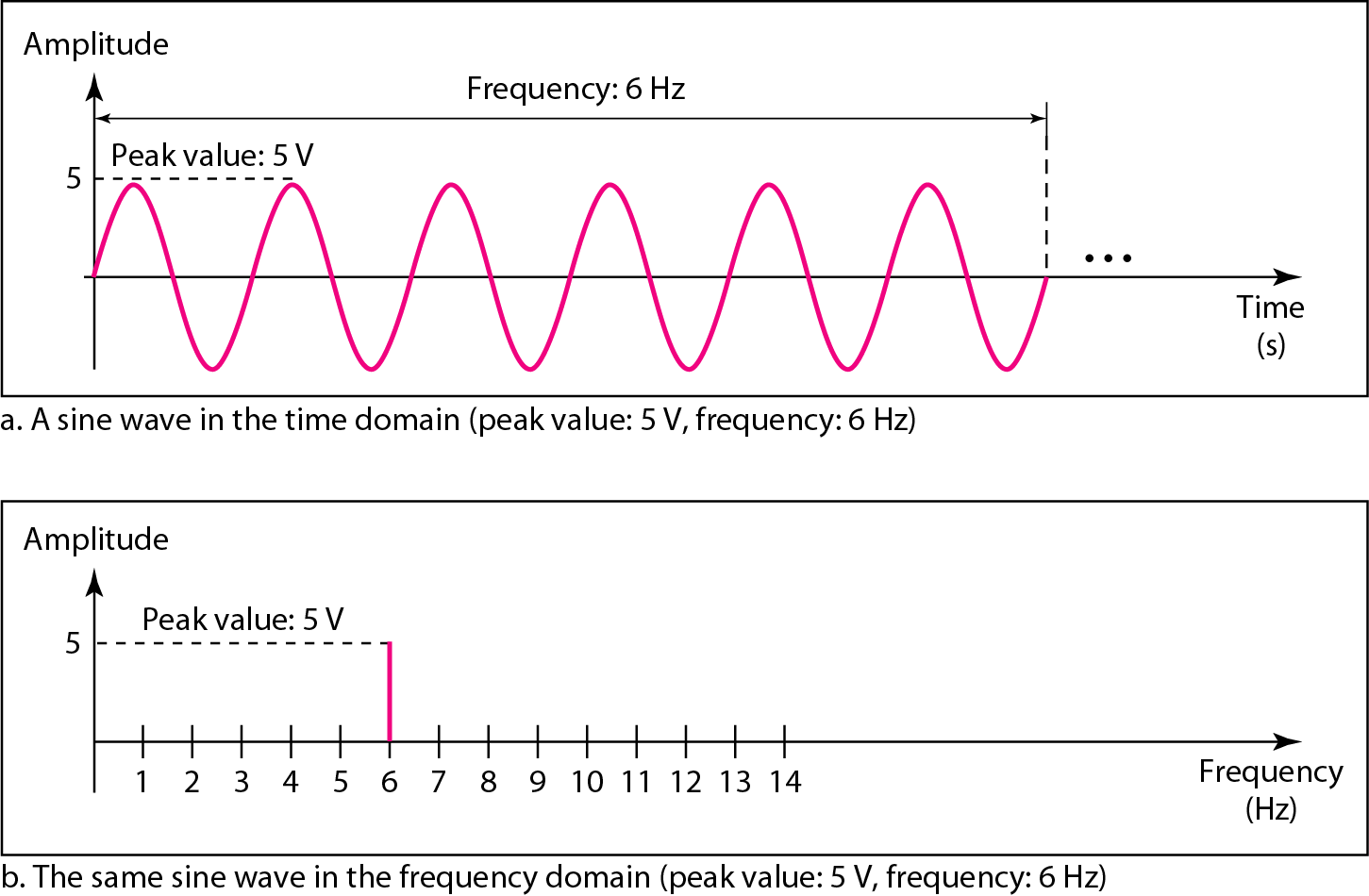


While the frequency of a signal is independent of the medium, the wavelength depends on both the frequency and the medium. Wavelength is a property of any type of signal. In data communications, we often use wavelength to describe the transmission of light in an optical fiber. The wavelength is the distance a simple signal can travel in one period.

Wavelength can be calculated if one is given the propagation speed (the speed of light) and the period of the signal. However, since period and frequency are related to each other, if we represent wavelength by A, propagation speed by S (speed of light), and frequency by F, we get A=C/F.

**Time and Frequency Domains**

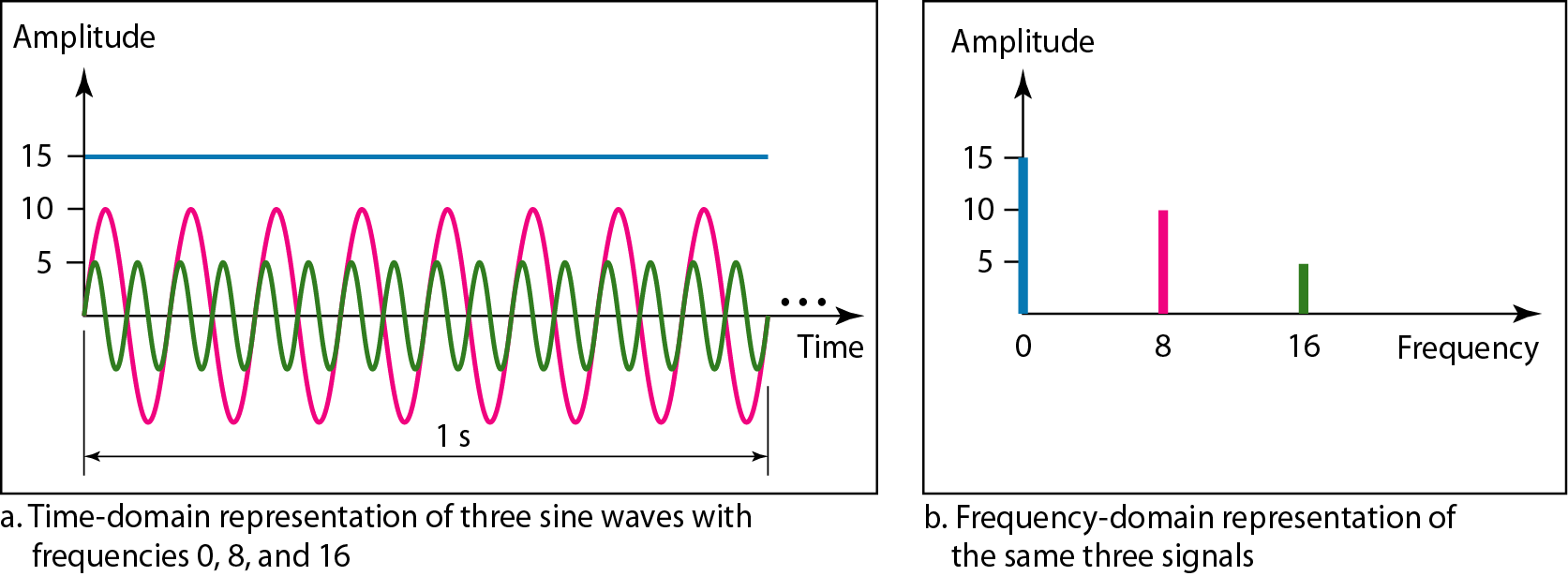
A sine wave is comprehensively defined by its amplitude, frequency, and phase. We have been showing a sine wave by using what is called a time-domain plot. The time-domain plot shows changes in signal amplitude with respect to time (it is an amplitude-versus-time plot). Phase is not explicitly shown on a time-domain plot. To show the relationship between amplitude and frequency, we can use what is called a frequency-domain plot. A frequency-domain plot is concerned with only the peak value and the frequency. Changes of amplitude during one period are not shown. Figure 3.7 shows a signal in both the time and frequency domains.



It is obvious that the frequency domain is easy to plot and conveys the information that one can find in a time domain plot. The advantage of the frequency domain is that we can immediately see the values of the frequency and peak amplitude. A complete sine wave is represented by one spike. The position of the spike shows the frequency; its height shows the peak amplitude.

**A complete sine wave in the time domain can be represented by one single spike in the frequency domain.**

The frequency domain is more compact and useful when we are dealing with more than one sine wave. For example, Figure shows three sine waves, each with different amplitude and frequency. All can be represented by three spikes in the frequency domain.



**Composite Signals**

So far, we have focused on simple sine waves. Simple sine waves have many applications in daily life. We can send a single sine wave to carry electric energy from one place to another. If we had only one single sine wave to convey a conversation over the phone, it would make no sense and carry no information. We would just hear a buzz. we need to send a composite signal to communicate data. A composite signal is made of many simple sine waves.

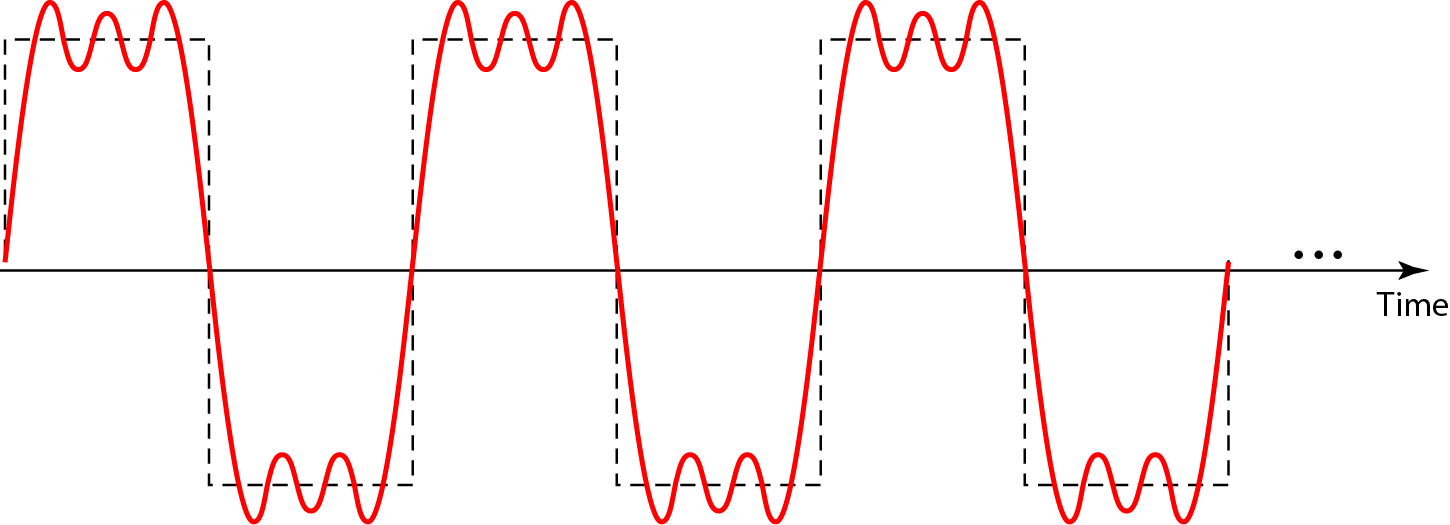
**A single-frequency sine wave is not useful in data communications; we need to send a composite signal, a signal made of many simple sine waves.**

In the early 1900s, the French mathematician Jean-Baptiste Fourier showed that any composite signal is actually a combination of simple sine waves with different frequencies, amplitudes, and phases.

A composite signal can be periodic or non-periodic. A periodic composite signal can be decomposed into a series of simple sine waves with discrete frequencies-frequencies that have integer values (1, 2, 3, and so on). A non-periodic composite signal can be decomposed into a combination of an infinite number of simple sine waves with continuous frequencies, frequencies that have real values.

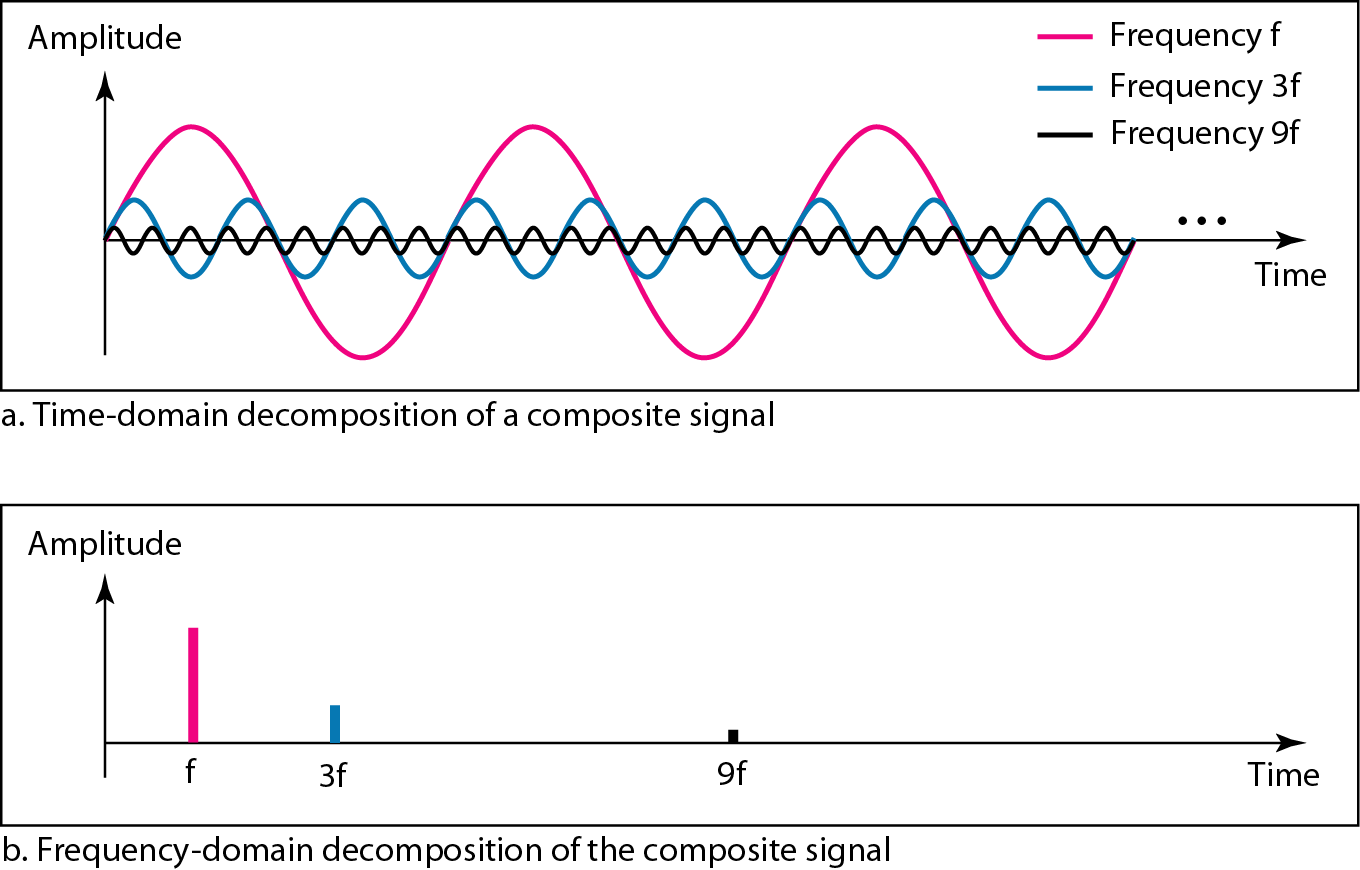
**If the composite signal is periodic, the decomposition gives a series of signals with discrete frequencies; if the composite signal is non-periodic, the decomposition gives a combination of sine waves with continuous frequencies.**

**Figure: *A composite periodic signal***

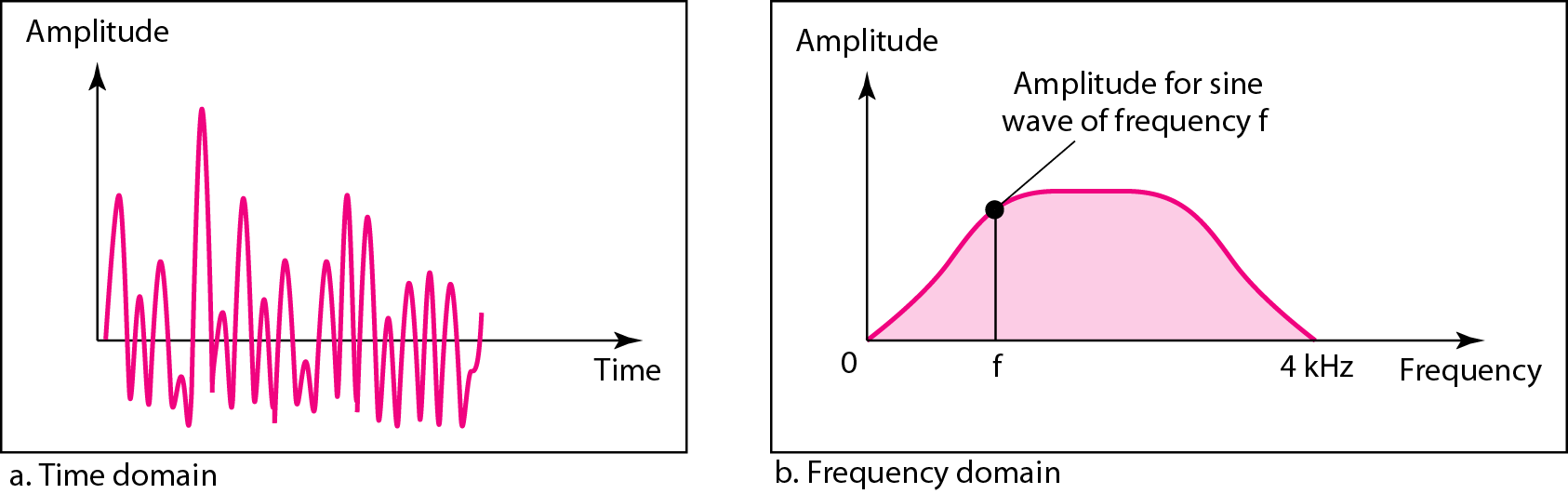


It is very difficult to manually decompose this signal into a series of simple sine waves. However, there are tools, both hardware and software, that can help us do the job. We are not concerned about how it is done; we are only interested in the result. Figure shows the result of decomposing the above signal in both the time and frequency domains.

The amplitude of the sine wave with frequency *f* is almost the same as the peak amplitude of the composite signal. The amplitude of the sine wave with frequency *3f*is one-third of that of the first, and the amplitude of the sine wave with frequency *9f* is one-ninth of the first. The frequency.

****

*The time andfrequency domains ofa nonperiodic signal*

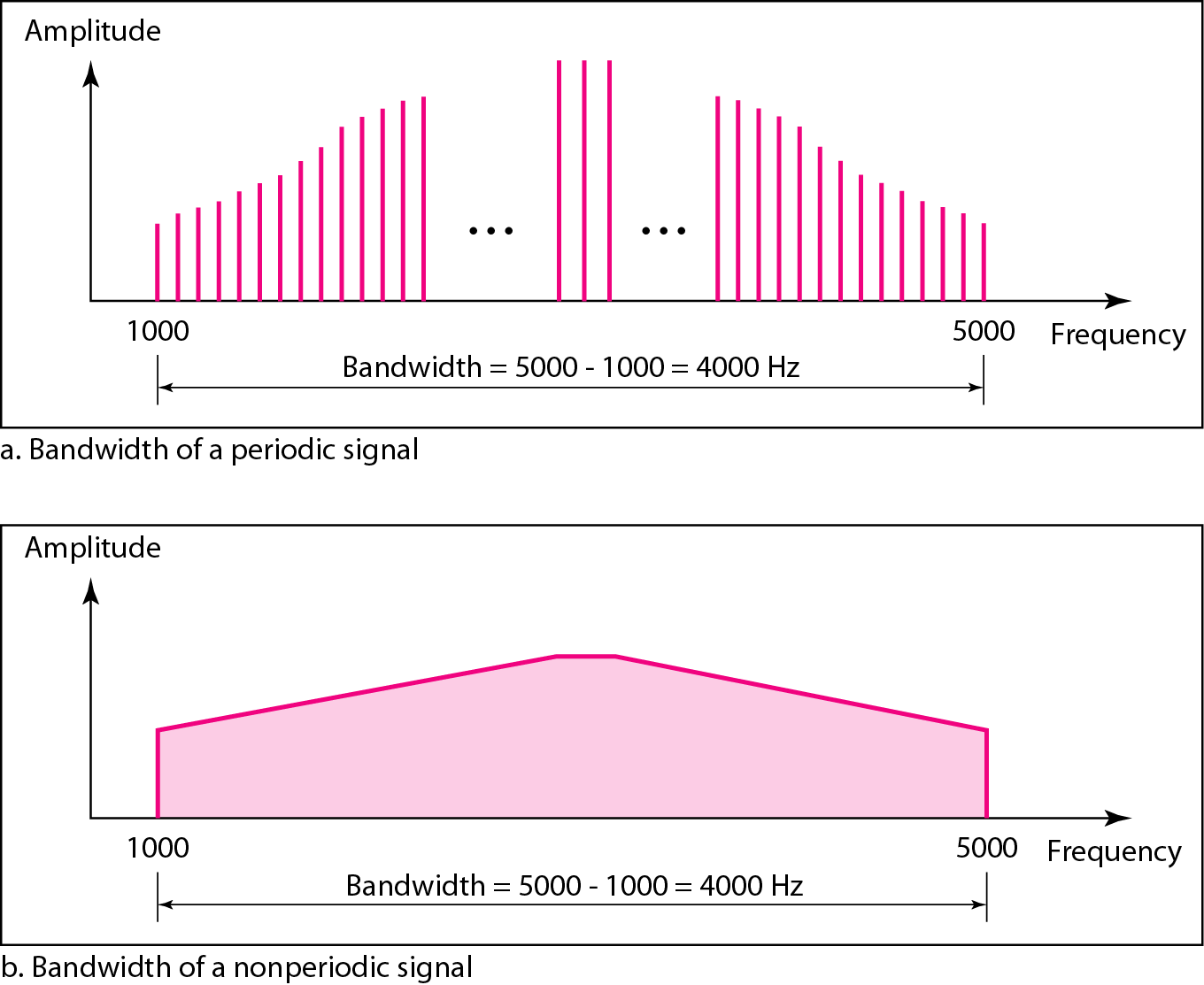
****

**Bandwidth**

The range of frequencies contained in a composite signal is its bandwidth. The bandwidth is normally a difference between two numbers. For example, if a composite signal contains frequencies between 1000 and 5000, its bandwidth is 5000 - 1000, or 4000.

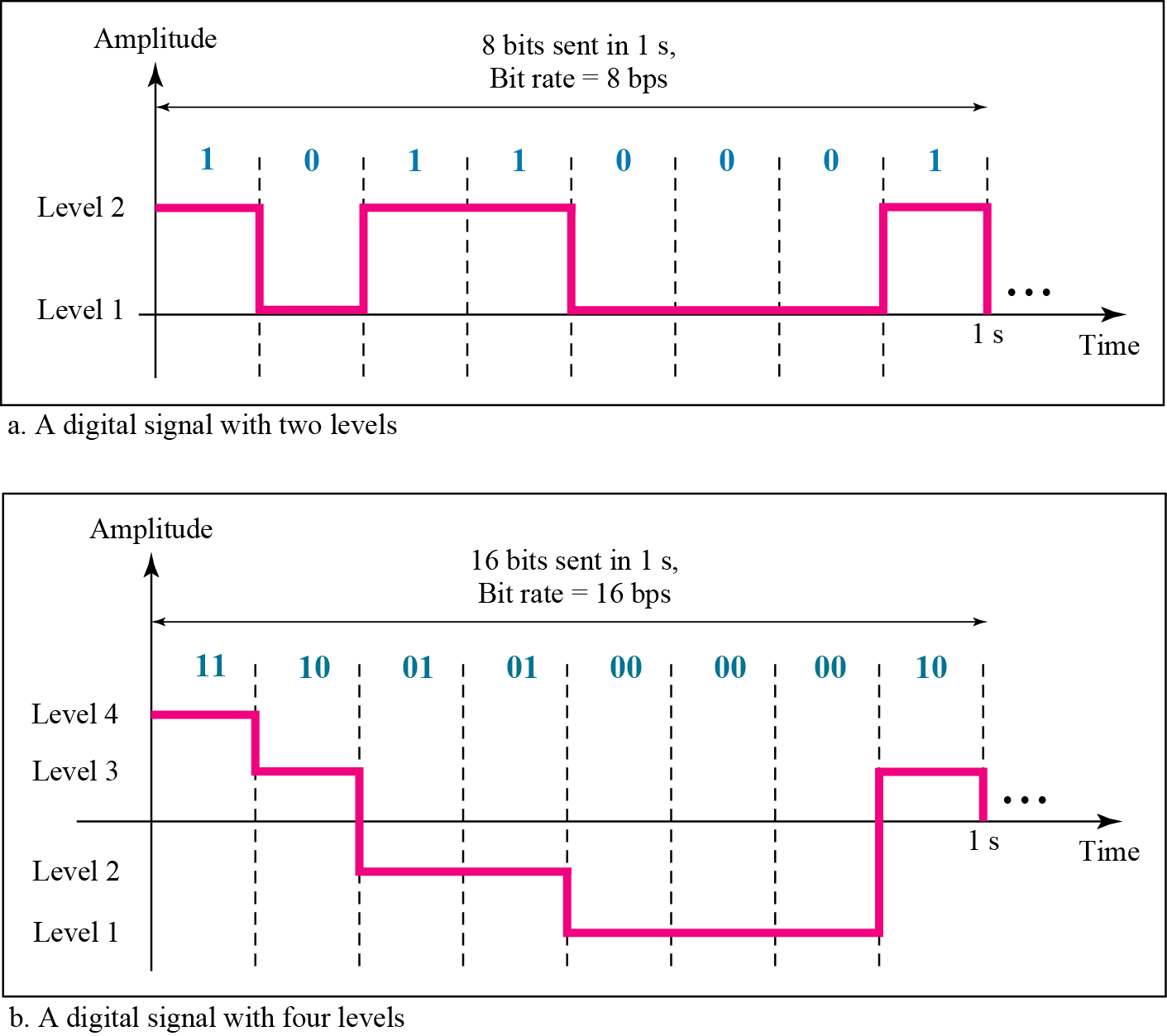
**The bandwidth of a composite signal is the difference between the highest and the lowest frequencies contained in that signal.**

*The bandwidth ofperiodic and nonperiodic composite signals*

****

**3.3 DIGITAL SIGNALS**

In addition to being represented by an analog signal, information can also be represented by a digital signal. For example, a I can be encoded as a positive voltage and a 0 as zero voltage. A digital signal can have more than two levels. In this case, we can send more than 1 bit for each level. Figure shows two signals, one with two levels and the other with four.



We send 1 bit per level in part an of the figure and 2 bits per level in part b of the figure. In general, if a signal has L levels, each level needs log2L bits.

**Bit Rate**

Most digital signals are non-periodic, and thus period and frequency are not appropriate characteristics. Another term-bit rate (instead of Frequency)-is used to describe digital signals. The bit rate is the number of bits sent in 1s, expressed in bits per second (bps).

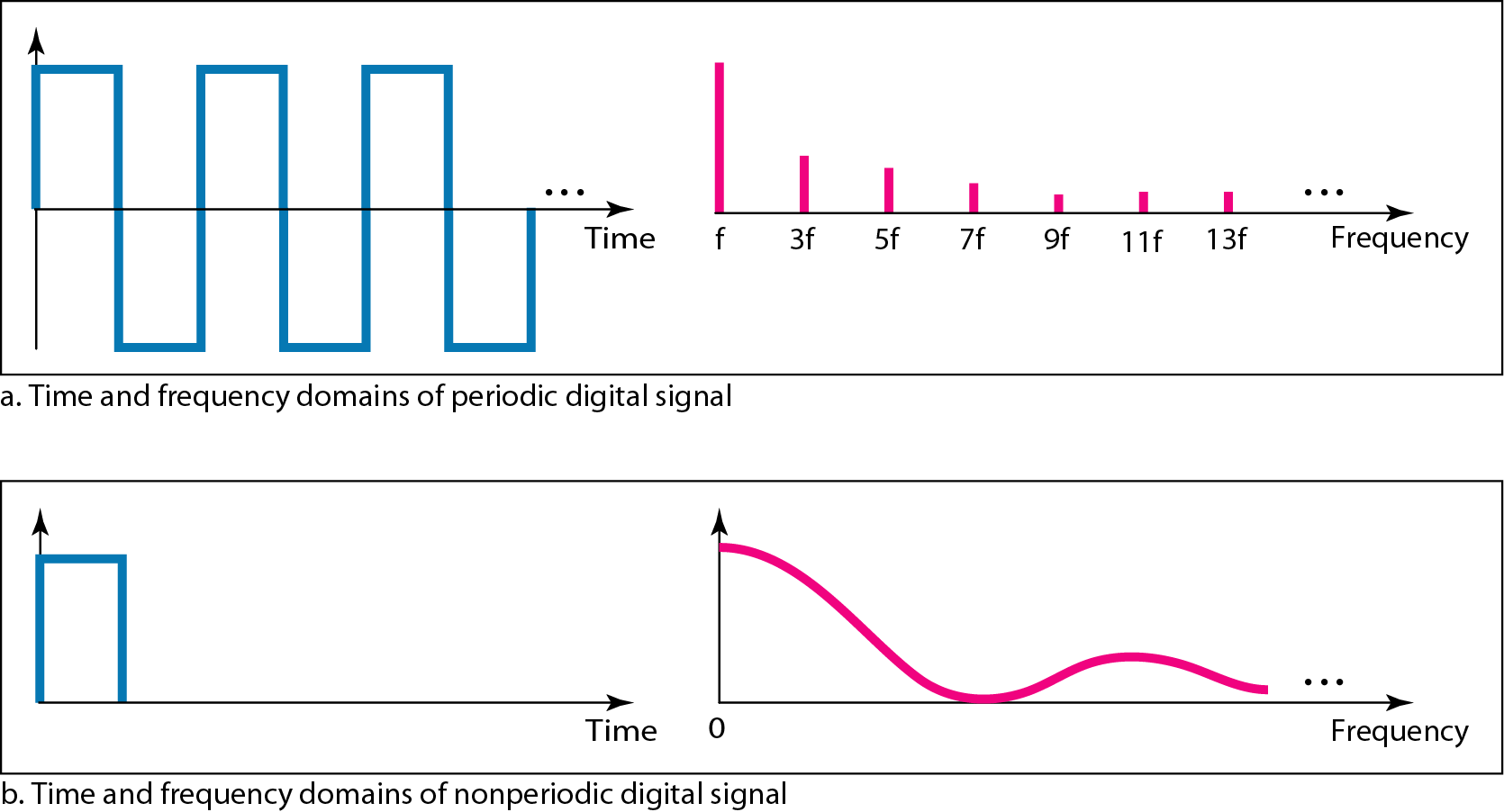
**Bit Length**

We discussed the concept of the wavelength for an analog signal: the distance one cycle occupies on the transmission medium. We can define something similar for a digital signal: the bit length. The bit length is the distance one bit occupies on the transmission medium.

**Bit length=propagation speed x bit duration**

**Digital Signal as a Composite Analog Signal**

Based on Fourier analysis, a digital signal is a composite analog signal. The bandwidth is infinite, as you may have guessed. We can intuitively come up with this concept when we consider a digital signal. A digital signal, in the time domain, comprises connected vertical and horizontal line segments. A vertical line in the time domain means a frequency of infinity (sudden change in time); a horizontal line in the time domain means a frequency of zero (no change in time). Going from a frequency of zero to a frequency of infinity (and vice versa) implies all frequencies in between are part of the domain. Fourier analysis can be used to decompose a digital signal. If the digital signal is periodic, which is rare in data communications, the decomposed signal has a frequency domain representation with an infinite bandwidth and discrete frequencies. If the digital signal is non-periodic, the decomposed signal still has an infinite bandwidth, but the frequencies are continuous. Figure shows a periodic and a non-periodic digital signal and their bandwidths.

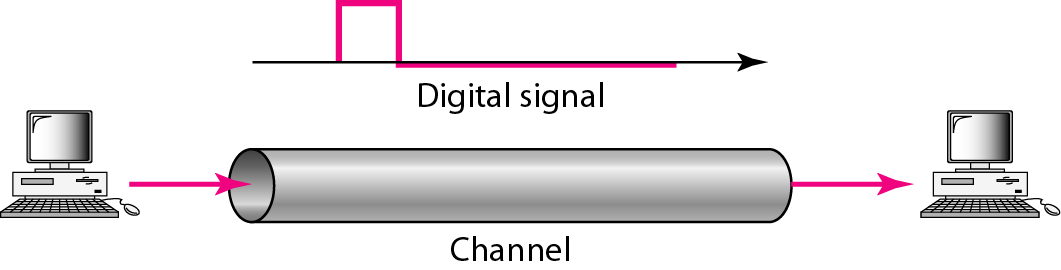


**Transmission of Digital Signals**

The previous discussion asserts that a digital signal, periodic or non-periodic, is a composite analog signal with frequencies between zero and infinity. For the remainder of the discussion, let us consider the case of a non-periodic digital signal, similar to the ones we encounter in data communications. The fundamental question is, how can we send a digital signal from point A to point B? We can transmit a digital signal by using one of two different approaches: baseband transmission or broadband transmission (using modulation).

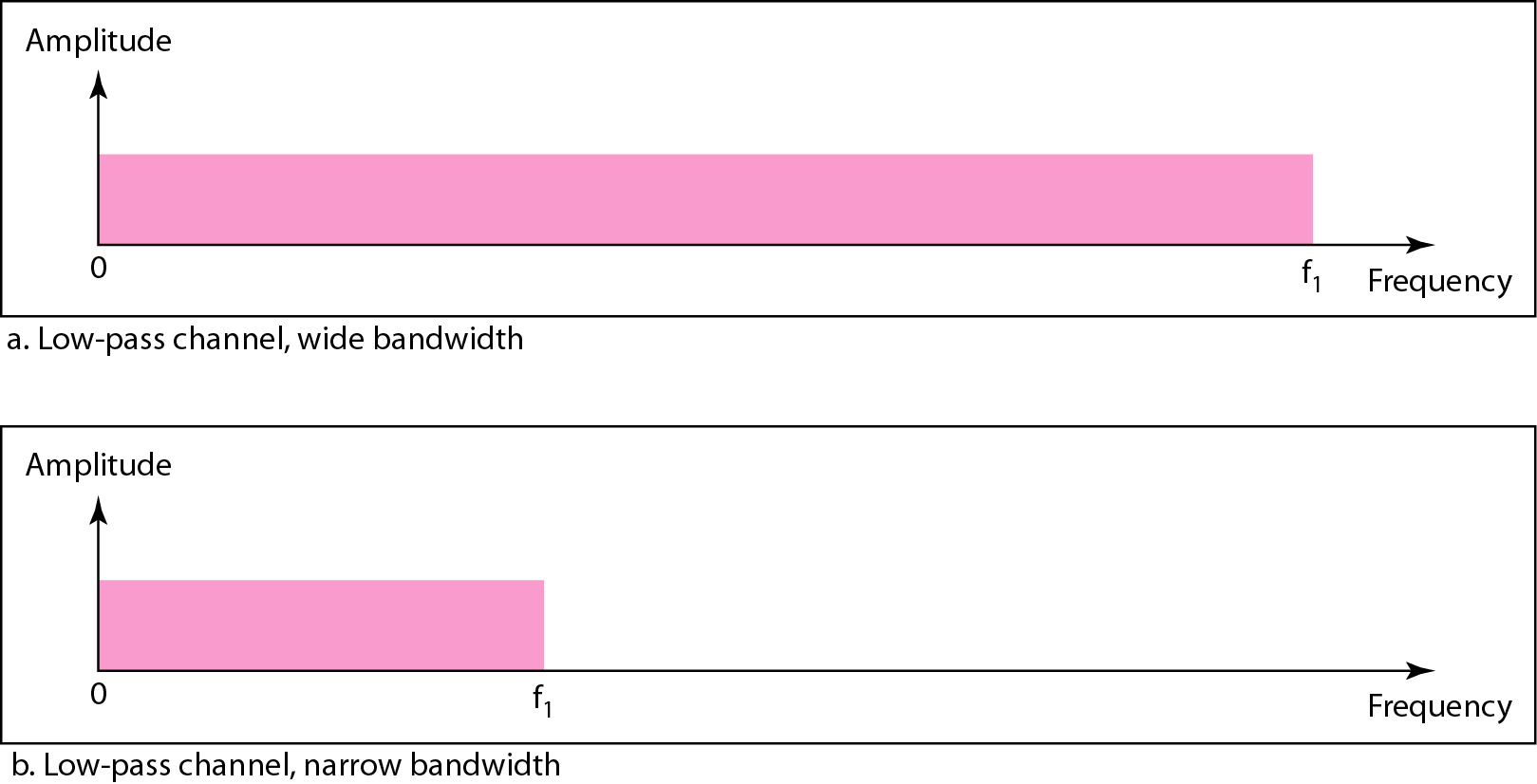
**Baseband Transmission**

Baseband transmission means sending a digital signal over a channel without changing the digital signal to an analog signal.



**A digital signal is a composite analog signal with an infinite bandwidth.**

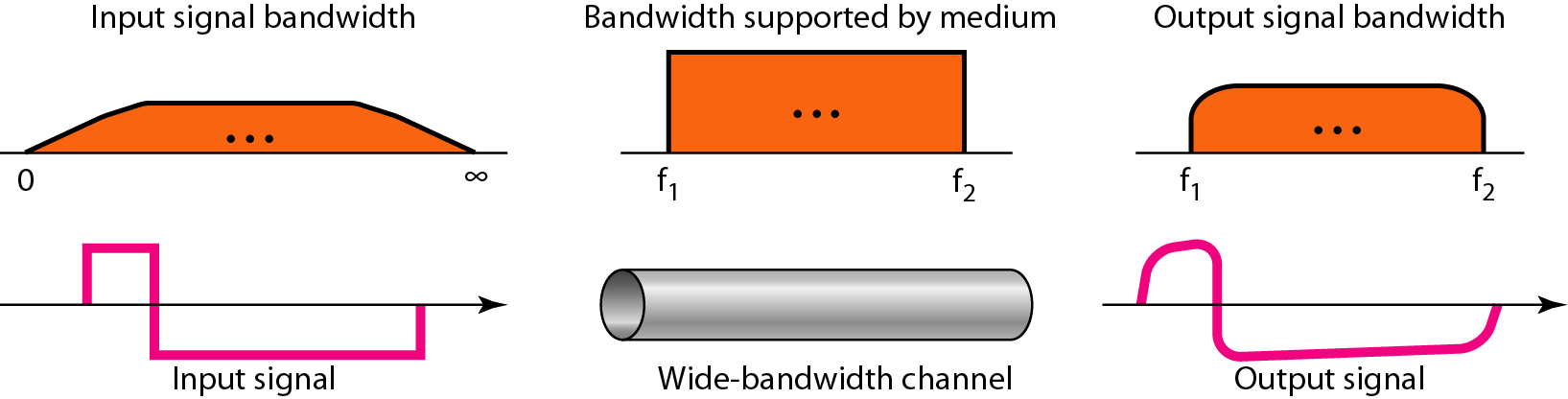
Baseband transmission requires that we have a low-pass channel, a channel with a bandwidth that starts from zero. This is the case if we have a dedicated medium with a bandwidth constituting only one channel. Again we have a low-pass channel, and we can use it for baseband communication. Figure shows two low-pass channels: one with a narrow bandwidth and the other with a wide bandwidth. We need to remember that a low-pass channel with infinite bandwidth is ideal, but we cannot have such a channel in real life. However, we can get close.



Let us study two cases of a baseband communication: a low-pass channel with a wide bandwidth and one with a limited bandwidth.

**Case 1: Low-Pass Channel with Wide Bandwidth**

If we want to preserve the exact form of a non-periodic digital signal with vertical segments vertical and horizontal segments horizontal, we need to send the entire spectrum, the continuous range of frequencies between zero and infinity. This is possible if we have a dedicated medium with an infinite bandwidth between the sender and receiver that preserves the exact amplitude of each component of the composite signal. Although this may be possible inside a computer (e.g., between CPU and memory), it is not possible between two devices. Fortunately, the amplitudes of the frequencies at the border of the bandwidth are so small that they can be ignored. This means that if we have a medium, such as a coaxial cable or fiber optic, with a very wide bandwidth, two stations can communicate by using digital signals with very good accuracy, as shown in Figure. Note that i is close to zero, and h is very high.



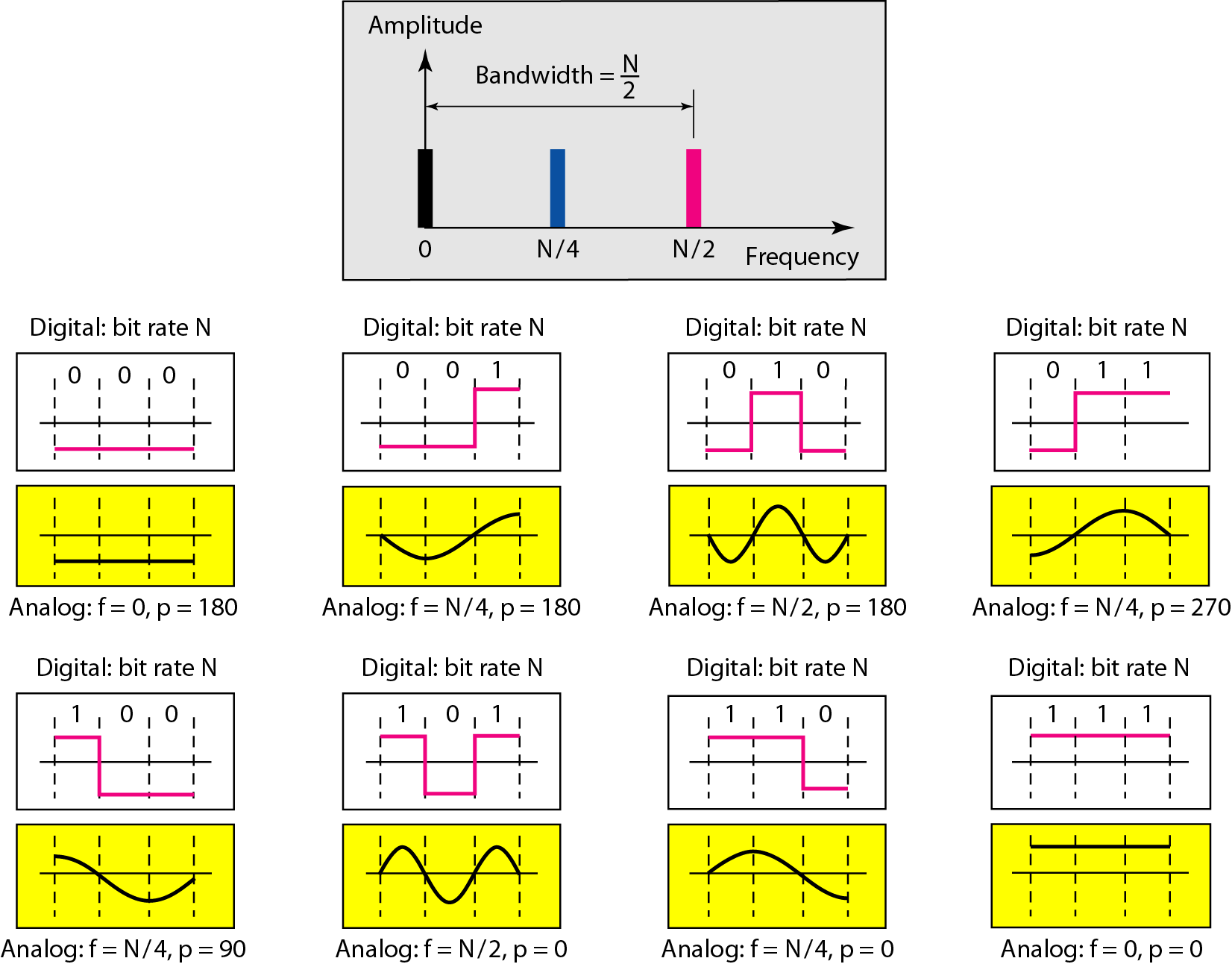
Although the output signal is not an exact replica of the original signal, the data can still be deduced from the received signal. Note that although some of the frequencies are blocked by the medium, they are not critical.

**Baseband transmission of a digital signal that preserves the shape of the digital signal is possible only if we have a low-pass channel with an infinite or very wide bandwidth.**

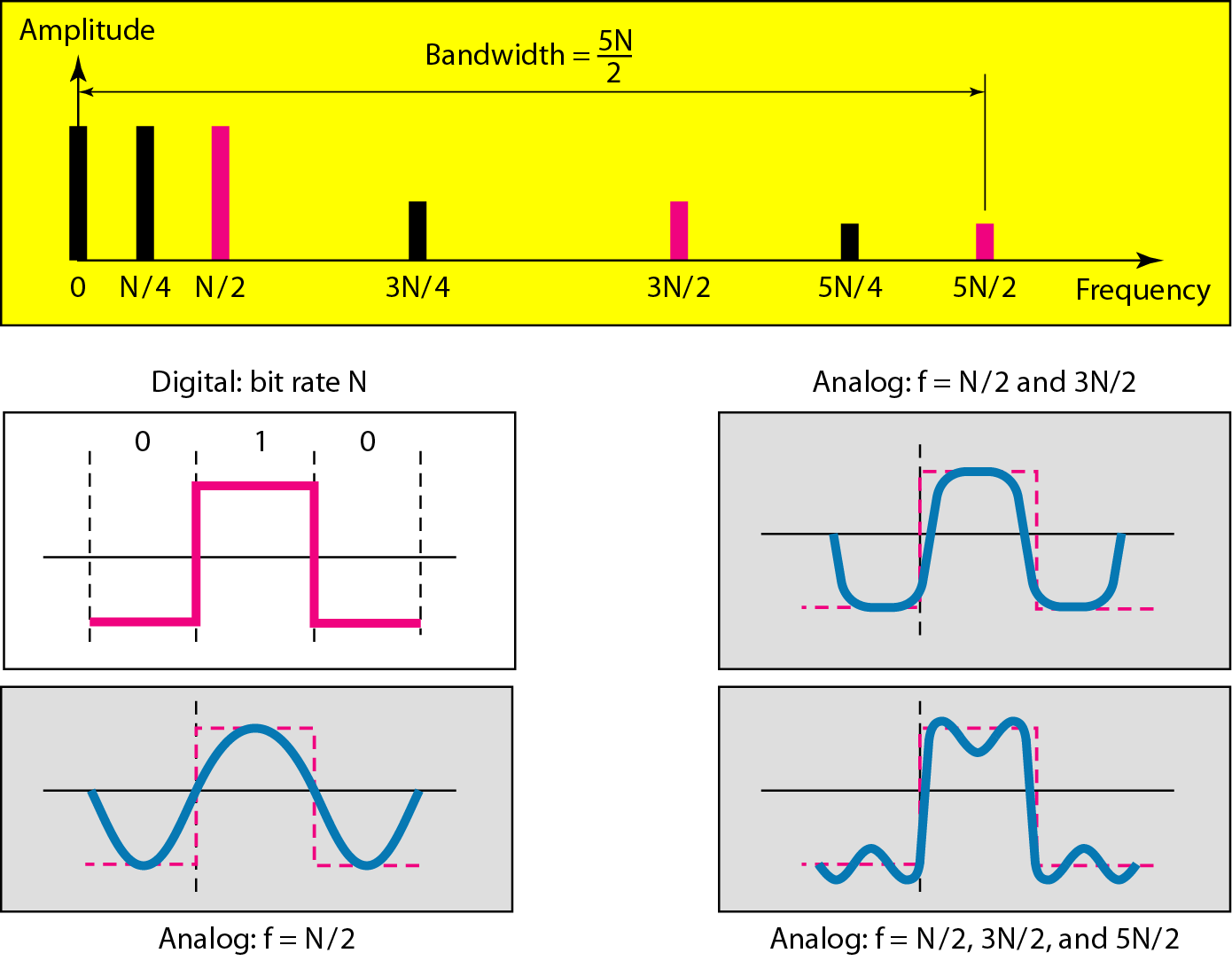
**Case 2: Low-Pass Channel with Limited Bandwidth**

In a low-pass channel with limited bandwidth, we approximate the digital signal with an analog signal. The level of approximation depends on the bandwidth available.

**In baseband transmission, the required bandwidth is proportional to the bit rate; if we need to send bits faster, we need more bandwidth.**

****

**Better Approximation** To make the shape of the analog signal look more like that of a digital signal, we need to add more harmonics of the frequencies. We need to increase the bandwidth. We can increase the bandwidth to *3N12, 5N12,* 7*NI2,* and so on. Figure shows the effect of this increase for one of the worst cases, the pattern 010.

****

**If the available channel is a bandpass channel, we cannot send the digital signal directly to the channel; we need to convert the digital signal to an analog signal before transmission.**

**3.4 TRANSMISSION IMPAIRMENT**

Signals travel through transmission media, which are not petfect. The impetfection causes signal impairment. This means that the signal at the beginning of the medium is not the same as the signal at the end of the medium. What is sent is not what is received. Three causes of impairment are attenuation, distortion, and noise.

**Attenuation: -** means **a loss of energy**. When a signal, simple or composite, travels through a medium, it loses some of its energy in overcoming the resistance of the medium. That is why a wire carrying electric signals gets warm, if not hot, after a while. Some of the electrical energy in the signal is converted to heat. To compensate for this loss, amplifiers are used to amplify the signal.

**Decibel: -** To show that a signal has lost or gained strength, engineers use the unit of the decibel. The decibel (dB) measures the relative strengths of two signals or one signal at two different points. Note that the decibel is negative if a signal is attenuated (loss of power) and positive if a signal is amplified (gain of power).

**Distortion: -** means that the **signal changes its form or shape**. Distortion can occur in a composite signal made of different frequencies. Each signal component has its own propagation speed (see the next section) through a medium and, therefore, its own delay in arriving at the final destination. Differences in delay may create a difference in phase if the delay is not exactly the same as the period duration. In other words, signal components at the receiver have phases different from what they had at the sender. The shape of the composite signal is therefore not the same.

**Noise: -** is another cause of impairment. Several types of noise, such as thermal noise, induced noise, crosstalk, and impulse noise, may corrupt the signal. Thermal noise is the random motion of electrons in a wire which creates an extra signal not originally sent by the transmitter. Induced noise comes from sources such as motors and appliances. These devices act as a sending antenna, and the transmission medium acts as the receiving antenna. Crosstalk is the effect of one wire on the other. One wire acts as a sending antenna and the other as the receiving antenna. Impulse noise is a spike (a signal with high energy in a very short time) that comes from power lines, lightning, and so on. Figure 3.29 shows the effect of noise on a signal.

**3.5 DATA RATE LIMITS**

A very important consideration in data communications is how fast we can send data, in bits per second, over a channel. Data rate depends on three factors:

1. The bandwidth available
2. The level of the signals we use
3. The quality of the channel (the level of noise)

Two theoretical formulas were developed to calculate the data rate: one by Nyquist for a noiseless channel. another by Shannon for a noisy channel.

**Noiseless Channel: Nyquist Bit Rate**

For a noiseless channel, the Nyquist bit rate formula defines the theoretical maximum bit rate

**BitRate= 2 x bandwidth x 10g2 L**

In this formula, bandwidth is the bandwidth of the channel, L is the number of signal levels used to represent data, and BitRate is the bit rate in bits per second.

According to the formula, we might think that, given a specific bandwidth, we can have any bit rate we want by increasing the number of signal levels. Although the idea is theoretically correct, practically there is a limit. When we increase the number of signal levels, we impose a burden on the receiver. If the number of levels in a signal is just 2, the receiver can easily distinguish between a 0 and a 1. If the level of a signal is 64, the receiver must be very sophisticated to distinguish between 64 different levels. In other words, increasing the levels of a signal reduces the reliability of the system.

**Increasing the levels of a signal may reduce the reliability of the system.**

**Noisy Channel: Shannon Capacity**

In reality, we cannot have a noiseless channel; the channel is always noisy. In 1944, Claude Shannon introduced a formula, called the Shannon capacity, to determine the theoretical highest data rate for a noisy channel:

**Capacity =bandwidth X log2 (1 +SNR)**

In this formula, bandwidth is the bandwidth of the channel, SNR is the signal-to-noise ratio, and capacity is the capacity of the channel in bits per second. Note that in the Shannon formula there is no indication of the signal level, which means that no matter how many levels we have, we cannot achieve a data rate higher than the capacity of the channel. In other words, the formula defines a characteristic of the channel, not the method of transmission.

**3.6 PERFORMANCE**

Up to now, we have discussed the tools of transmitting data (signals) over a network and how the data behave. One important issue in networking is the performance of the network-how good is it? We discuss quality of service, an overall measurement of network performance,

**Bandwidth**

One characteristic that measures network performance is bandwidth. However, the term can be used in two different contexts with two different measuring values:

**I. Bandwidth in Hertz**

We have discussed this concept. Bandwidth in hertz is the range of frequencies contained in a composite signal or the range of frequencies a channel can pass.

**II. Bandwidth in Bits per Seconds**

The term bandwidth can also refer to the number of bits per second that a channel, a link, or even a network can transmit.

**Relationship**

There is an explicit relationship between the bandwidth in hertz and bandwidth in bits per seconds. Basically, an increase in bandwidth in hertz means an increase in bandwidth in bits per second. The relationship depends on whether we have baseband transmission or transmission with modulation.

**In networking, we use the term bandwidth in two contexts.**

* The first, bandwidth in hertz, refers to the range of frequencies in a composite signal or the range of frequencies that a channel can pass.
* The second, bandwidth in bits per second, refers to the speed of bit transmission in a channel or link.

**Throughput**

The throughput is a measure of how fast we can actually send data through a network. Although, at first glance, bandwidth in bits per second and throughput seem the same, they are different. A link may have a bandwidth of B bps, but we can only send T bps through this link with T always less than B. In other words, the bandwidth is a potential measurement of a link; the throughput is an actual measurement of how fast we can send data.

**Latency (Delay)**

The latency or delay defines how long it takes for an entire message to completely arrive at the destination from the time the first bit is sent out from the source. We can say that latency is made of four components: propagation time, transmission time, queuing time and processing delay.

**Latency = propagation time + transmission time + queuing time + processing delay**

**I. Propagation Time**

Propagation time measures the time required for a bit to travel from the source to the destination. The propagation time is calculated by dividing the distance by the propagation speed.

**Propagation time = Distance / Propagation speed**

The propagation speed of electromagnetic signals depends on the medium and on the frequency of the signal.

**II. Transmission Time**

In data communications we don't send just 1 bit, we send a message. The first bit may take a time equal to the propagation time to reach its destination; the last bit also may take the same amount of time. However, there is a time between the first bit leaving the sender and the last bit arriving at the receiver. The first bit leaves earlier and arrives earlier; the last bit leaves later and arrives later. The time required for transmission of a message depends on the size of the message and the bandwidth of the channel.

**Transmission time=Message size / Bandwidth**

**III. Queuing Time**

The third component in latency is the queuing time, the time needed for each intermediate or end device to hold the message before it can be processed. The queuing time is not a fixed factor; it changes with the load imposed on the network. When there is heavy traffic on the network, the queuing time increases. An intermediate device, such as a router, queues the arrived messages and processes them one by one. If there are many messages, each message will have to wait.

**IV. Bandwidth-Delay Product**

Bandwidth and delay are two performance metrics of a link. However, as we will see in this chapter and future chapters, what is very important in data communications is the product of the two, the bandwidth-delay product. Let us elaborate on this issue, using two hypothetical cases as examples.

**The bandwidth-delay product defines the number of bits that can fill the link.**

**Jitter**

Another performance issue that is related to delay is jitter. We can roughly say that jitter is a problem if different packets of data encounter different delays and the application using the data at the receiver site is time-sensitive (audio and video data, for example). If the delay for the first packet is 20 ms, for the second is 45 ms, and for the third is 40 ms, then the real-time application that uses the packets endures jitter.

**CHAPTER 4**

**DIGITAL TRANSMISSION**

we show the schemes and techniques that we use to transmit data digitally. First, we discuss digital-to-digital conversion techniques, methods which convert digital data to digital signals. Second, we discuss analog-to-digital conversion techniques, methods which change an analog signal to a digital signal. Finally, we discuss transmission modes.

**4.1 DIGITAL-TO-DIGITAL CONVERSION**

We see how we can represent digital data by using digital signals. The conversion involves three techniques: line coding, block coding, and scrambling. Line coding is always needed~ block coding and scrambling mayor may not be needed.

**Line Coding**

Line coding is the process of converting **digital data to digital signals**. We assume that data, in the form of text, numbers, graphical images, audio, or video, are stored in computer memory as sequences of bits. Line coding *converts a sequence of bits to a digital signal*. At the sender, digital data are encoded into a digital signal; at the receiver, the digital data are recreated by decoding the digital signal.

**Characteristics**

Before discussing different line coding schemes, we address their common characteristics.

**1. Signal Element Versus Data Element: -** A **data element** is the smallest entity that can represent a piece of information. A signal element carries data elements. A **signal element** is the shortest unit (time wise) of a digital signal.

**2. Data Rate Versus Signal Rate: -** The **data rate** defines the number of data elements (bits) sent in 1s. The unit is bits per second (bps). The **signal rate** is the number of signal elements sent in 1s. The unit is the baud. The **data rate** is sometimes called the **bit rate**; the **signal rate** is sometimes called the **pulse rate**, the **modulation rate**, or the **baud rate**. One goal in data communications is to increase the data rate while decreasing the signal rate. *Increasing the data rate increases the speed of transmission; decreasing the signal rate decreases the bandwidth requirement*. There are three cases: the worst, best, and average. The worst case is when we need the maximum signal rate; the best case is when we need the minimum. we are usually interested in the average case.

baud

where N is the data rate (bps); c is the case factor, which varies for each case; S is the number of signal elements; and r is the previously defined factor.

**3. Bandwidth**: - most digital signals we encounter in real life have a bandwidth with finite values. the bandwidth is theoretically infinite, but many of the components have such a small amplitude that they can be ignored. The **effective** **bandwidth** is finite. From now on we are talking about this effective bandwidth. ***Although the actual bandwidth of a digital signal is infinite, the effective bandwidth is finite*.** More changes in the signal mean injecting more frequencies into the signal. The bandwidth reflects the range of frequencies we need. There is a relationship between the baud rate (signal rate) and the bandwidth. When we talk about the bandwidth, we normally define a range of frequencies. For the moment, we can say that the bandwidth (range of frequencies) is proportional to the signal rate (baud rate). The minimum bandwidth can be given as

We can solve for the maximum data rate if the bandwidth of the channel is given.

**4. Baseline Wandering:** **-** In decoding a digital signal, the receiver calculates a running average of the received signal power. This average is called the baseline. The incoming signal power is evaluated against this baseline to determine the value of the data element. A long string of 0s or 1s can cause a drift in the baseline (baseline wandering) and make it difficult for the receiver to decode correctly. A good line coding scheme needs to prevent baseline wandering.

**5. DC Components: -** When the voltage level in a digital signal is constant for a while, the spectrum creates very low frequencies (results of Fourier analysis). These frequencies around zero, called DC (direct-current) components, present problems for a system that cannot pass low frequencies or a system that uses electrical coupling (via a transformer). For example, a telephone line cannot pass frequencies below 200 Hz. Also a long-distance link may use one or more transformers to isolate different parts of the line electrically. For these systems, we need a scheme with no DC component.

**6. Self-synchronization: -** To correctly interpret the signals received from the sender, the receiver's bit intervals must correspond exactly to the sender's bit intervals. If the receiver clock is faster or slower, the bit intervals are not matched and the receiver might misinterpret the signals. Figure 4.3 shows a situation in which the receiver has a shorter bit duration. The sender sends 10110001, while the receiver receives 110111000011.

**7. Built-in Error Detection: -** It is desirable to have a built-in error-detecting capability in the generated code to detect some of or all the errors that occurred during transmission. Some encoding schemes that we will discuss have this capability to some extent.

**8. Immunity to Noise and Interference: -** Another desirable code characteristic is a code I that is immune to noise and other interferences. Some encoding schemes that we will discuss have this capability.

**9. Complexity: -** A complex scheme is more costly to implement than a simple one. For example, a scheme that uses four signal levels is more difficult to interpret than one that uses only two levels.

**Line Coding Schemes**

We can roughly divide line coding schemes into five broad categories, as shown in Figure

**Unipolar Scheme**

In a unipolar scheme, all the signal levels are on one side of the time axis, either above or below.

NRZ (Non-Return-to-Zero) Traditionally, a unipolar scheme was designed as a non-return-to-zero (NRZ) scheme in which the positive voltage defines bit I and the zero voltage defines bit O. Itis called NRZ because the signal does not return to zero at the middle of the bit. Figure show a unipolar NRZ scheme.

Compared with its polar counterpart (see the next section), this scheme is very costly. As we will see shortly, the normalized power (power needed to send 1 bit per unit line resistance) is double that for polar NRZ. For this reason, this scheme is normally not used in data communications today.

**Polar Schemes**

In polar schemes, the voltages are on the both sides of the time axis. For example, the voltage level for 0 can be positive and the voltage level for I can be negative.

Non-Return-to-Zero (NRZ) In polar NRZ encoding, we use two levels of voltage amplitude. We can have two versions of polar NRZ: NRZ-Land NRZ-I, as shown in Figure 4.6. The figure also shows the value of r, the average baud rate, and the bandwidth. In the first variation, NRZ-L (NRZ-Level), the level of the voltage determines the value of the bit. In the second variation, NRZ-I (NRZ-Invert), the change or lack of change in the level of the voltage determines the value of the bit. If there is no change, the bit is 0; if there is a change, the bit is 1.

In NRZ-L the level of the voltage determines the value of the bit. In NRZ-I the inversion or the lack of inversion determines the value of the bit.

Let us compare these two schemes based on the criteria we previously defined. Although baseline wandering is a problem for both variations, it is twice as severe in NRZ-L. If there is a long sequence of 0s or 1s in NRZ-L, the average signal power becomes skewed. The receiver might have difficulty discerning the bit value. In NRZ-I this problem occurs only for a long sequence of as. If somehow we can eliminate the long sequence of as, we can avoid baseline wandering. We will see shortly how this can be done.

The synchronization problem (sender and receiver clocks are not synchronized) also exists in both schemes. Again, this problem is more serious in NRZ-L than in NRZ-I. While a long sequence of as can cause a problem in both schemes, a long sequence of 1s affects only NRZ-L.

Another problem with NRZ-L occurs when there is a sudden change of polarity in the system. For example, if twisted-pair cable is the medium, a change in the polarity of the wire results in all as interpreted as 1s and all 1s interpreted as as. NRZ-I does not have this problem. Both schemes have an average signal rate of N/2 Bd.

**NRZ-L and NRZ-J both have an average signal rate ofNI2 Bd.**

Let us discuss the bandwidth. Figure 4.6 also shows the normalized bandwidth for both variations. The vertical axis shows the power density (the power for each I Hz of bandwidth); the horizontal axis shows the frequency. The bandwidth reveals a very serious problem for this type of encoding. The value of the power density is velY high around frequencies close to zero. This means that there are DC components that carry a high level of energy. As a matter of fact, most of the energy is concentrated in frequencies between a and NIl. This means that although the average of the signal rate is N12, the energy is not distributed evenly between the two halves.

**NRZ-L and NRZ-J both have a DC component problem.**

Return to Zero (RZ) The main problem with NRZ encoding occurs when the sender and receiver clocks are not synchronized. The receiver does not know when one bit has ended and the next bit is starting. One solution is the return-to-zero (RZ) scheme, which uses three values: positive, negative, and zero. In RZ, the signal changes not between bits but during the bit. In Figure 4.7 we see that the signal goes to 0 in the middle of each bit. It remains there until the beginning of the next bit. The main disadvantage of RZ encoding is that it requires two signal changes to encode a bit and therefore occupies greater bandwidth. The same problem we mentioned, a sudden change of polarity resulting in all as interpreted as 1s and all 1s interpreted as as, still exist here, but there is no DC component problem. Another problem is the complexity: RZ uses three levels of voltage, which is more complex to create and discern. As a result of all these deficiencies, the scheme is not used today. Instead, it has been replaced by the better-performing Manchester and differential Manchester schemes (discussed next).

**Biphase:** Manchester and Differential Manchester The idea of RZ (transition at the middle of the bit) and the idea of NRZ-L are combined into the Manchester scheme. In Manchester encoding, the duration of the bit is divided into two halves. The voltage remains at one level during the first half and moves to the other level in the second half. The transition at the middle of the bit provides synchronization. Differential Manchester, on the other hand, combines the ideas of RZ and NRZ-I. There is always a transition at the middle of the bit, but the bit values are determined at the beginning of the bit. If the next bit is 0, there is a transition; if the next bit is 1, there is none. Figure 4.8 shows both Manchester and differential Manchester encoding.

**In Manchester and differential Manchester encoding, the transition at the middle of the bit is used for synchronization.**

The Manchester scheme overcomes several problems associated with NRZ-L, and differential Manchester overcomes several problems associated with NRZ-I. First, there is no baseline wandering. There is no DC component because each bit has a positive and negative voltage contribution. The only drawback is the signal rate. The signal rate for Manchester and differential Manchester is double that for NRZ. The reason is that there is always one transition at the middle ofthe bit and maybe one transition at the end ofeach bit. Figure 4.8 shows both Manchester and differential Manchester encoding schemes. Note that Manchester and differential Manchester schemes are also called biphase schemes.

**The minimum bandwidth of Manchester and differential Manchester is 2 times that of NRZ**.

**Bipolar Schemes**

In bipolar encoding (sometimes called multilevel binary), there are three voltage levels: positive, negative, and zero. The voltage level for one data element is at zero, while the voltage level for the other element alternates between positive and negative.

**In bipolar encoding, we use three levels: positive, zero, and negative**.

**AMI and Pseudoternary** Figure 4.9 shows two variations ofbipolar encoding: AMI and pseudoternary. A common bipolar encoding scheme is called bipolar alternate mark inversion (AMI). In the term alternate mark inversion, the word mark comes from telegraphy and means 1. So AMI means alternate I inversion. A neutral zero voltage represents binary O. Binary Is are represented by alternating positive and negative voltages. A variation ofAMI encoding is called pseudoternary in which the 1 bit is encoded as a zero voltage and the 0 bit is encoded as alternating positive and negative voltages.

The bipolar scheme was developed as an alternative to NRZ. The bipolar scheme has the same signal rate as NRZ, but there is no DC component. The NRZ scheme has most of its energy concentrated near zero frequency, which makes it unsuitable for transmission over channels with poor performance around this frequency. The concentration ofthe energy in bipolar encoding is around frequency N12. Figure 4.9 shows the typical energy concentration for a bipolar scheme.

One may ask why we do not have DC component in bipolar encoding. We can answer this question by using the Fourier transform, but we can also think about it intuitively. Ifwe have a long sequence of 1s, the voltage level alternates between positive and negative; it is not constant. Therefore, there is no DC component. For a long sequence of Os, the voltage remains constant, but its amplitude is zero, which is the same as having no DC component. In other words, a sequence that creates a constant zero voltage does not have a DC component. AMI is commonly used for long-distance communication, but it has a synchronization problem when a long sequence ofOs is present in the data. Later in the chapter, we will see how a scrambling technique can solve this problem.

**Multilevel Schemes**

The desire to increase the data speed or decrease the required bandwidth has resulted in the creation of many schemes. The goal is to increase the number of bits per baud by encoding a pattern of m data elements into a pattern of n signal elements. We only have two types of data elements (0s and 1s), which means that a group of m data elements can produce a combination of 2m data patterns. We can have different types of signal elements by allowing different signal levels. If we have L different levels, then we can produce Ln combinations of signal patterns. If 2m =Ln, then each data pattern is encoded into one signal pattern. If 2m < Ln, data patterns occupy only a subset of signal patterns. The subset can be carefully designed to prevent baseline wandering, to provide synchronization, and to detect errors that occurred during data transmission. Data encoding is not possible if 2m > Ln because some of the data patterns cannot be encoded. The code designers have classified these types of coding as mBnL, where m is the length of the binary pattern, B means binary data, n is the length of the signal pattern, and L is the number of levels in the signaling. A letter is often used in place of L: B (binary) for L =2, T (ternary) for L =3, and Q (quaternary) for L =4. Note that the first two letters define the data pattern, and the second two define the signal pattern.

**In mBnL schemes, a pattern of m data elements is encoded as a pattern of n signal elements in which 2m≤Ln.**

**2BIQ** The first mBnL scheme we discuss, two binary, one quaternary (2BIQ), uses data patterns of size 2 and encodes the 2-bit patterns as one signal element belonging to a four-level signal. In this type ofencoding m =2, n =1, and L =4 (quatemary). Figure 4.10 shows an example ofa 2B1Q signal. The average signal rate of 2BlQ is S =N/4. This means that using 2BIQ, we can send data 2 times faster than by using NRZ-L. However, 2B lQ uses four different signal levels, which means the receiver has to discern four different thresholds. The reduced bandwidth comes with a price. There are no redundant signal patterns in this scheme because 22 =41. As we will see in Chapter 9, 2BIQ is used in DSL (Digital Subscriber Line) technology to provide a high-speed connection to the Internet by using subscriber telephone lines.

**8B6T** A very interesting scheme is eight binary, six ternary (8B6T). This code is used with 100BASE-4T cable, as we will see in Chapter 13. The idea is to encode a pattern of 8 bits as a pattern of 6 signal elements, where the signal has three levels (ternary). In this type of scheme, we can have 28 =256 different data patterns and 36 =478 different signal patterns. The mapping table is shown in Appendix D. There are 478 - 256 =222 redundant signal elements that provide synchronization and error detection. Part ofthe redundancy is also used to provide DC balance. Each signal pattern has a weight of0 or +1 DC values. This means that there is no pattern with the weight -1. To make the whole stream Dc-balanced, the sender keeps track of the weight. If two groups of weight 1 are encountered one after another, the first one is sent as is, while the next one is totally inverted to give a weight of -1. Figure 4.11 shows an example of three data patterns encoded as three signal patterns. The three possible signal levels are represented as -,0, and +. The first 8-bit pattern 00010001 is encoded as the signal pattern -0-0++ with weight 0; the second 8-bit pattern 01010011 is encoded as - + - + + 0 with weight +1. The third bit pattern should be encoded as + - - + 0 + with weight +1. To create DC balance, the sender inverts the actual signal. The receiver can easily recognize that this is an inverted pattern because the weight is -1. The pattern is inverted before decoding.

The average signal rate of the scheme is theoretically Save = ! X N X §; in practice the minimum bandwidth is very close to 6N18. 2 8

**4D-PAMS** The last signaling scheme we discuss in this category is called four dimensional five-level pulse amplitude modulation (4D-PAM5). The 4D means that data is sent over four wires at the same time. It uses five voltage levels, such as -2, -1, 0, 1, and 2. However, one level, level 0, is used only for forward error detection (discussed in Chapter 10). If we assume that the code is just one-dimensional, the four levels create something similar to 8B4Q. In other words, an 8-bit word is translated to a signal element off our different levels. The worst signal rate for this imaginary one-dimensional version is N X 4/8, orN12. The technique is designed to send data over four channels (four wires). This means the signal rate can be reduced to N18, a significant achievement. All 8 bits can be fed into a wire simultaneously and sent by using one signal element. The point here is that the four signal elements comprising one signal group are sent simultaneously in a four-dimensional setting. Figure 4.12 shows the imaginary one-dimensional and the actual four-dimensional implementation. Gigabit LANs (see Chapter 13) use this technique to send 1-Gbps data over four copper cables that can handle 125 Mbaud. This scheme has a lot of redundancy in the signal pattern because 28 data patterns are matched to 44 = 256 signal patterns. The extra signal patterns can be used for other purposes such as error detection.

**Multiline Transmission: MLT-3**

NRZ-I and differential Manchester are classified as differential encoding but use two transition rules to encode binary data (no inversion, inversion). Ifwe have a signal with more than two levels, we can design a differential encoding scheme with more than two transition rules. MLT-3 is one of them. The multiline transmission, three level (MLT-3) scheme uses three levels (+v, 0, and - V) and three transition rules to move between the levels.

1. If the next bit is 0, there is no transition.

2. If the next bit is 1 and the current level is not 0, the next level is 0.

3. If the next bit is 1 and the cut Tent level is 0, the next level is the opposite of the last nonzero level.

The behaviour of MLT-3 can best be described by the state diagram shown in Figure 4.13. The three voltage levels (-V, 0, and +V) are shown by three states (ovals). The transition from one state (level) to another is shown by the connecting lines. Figure 4.13 also shows two examples ofan MLT-3 signal.

One might wonder why we need to use MLT-3, a scheme that maps one bit to one signal element. The signal rate is the same as that for NRZ-I, but with greater complexity (three levels and complex transition rules). It turns out that the shape of the signal in this scheme helps to reduce the required bandwidth. Let us look at the worst-case scenario, a sequence of Is. In this case, the signal element pattern +VO - VO is repeated every 4 bits. A nonperiodic signal has changed to a periodic signal with the period equal to 4 times the bit duration. This worst-case situation can be simulated as an analog signal with a frequency one-fourth of the bit rate. In other words, the signal rate for MLT-3 is one-fourth the bit rate. This makes MLT-3 a suitable choice when we need to send 100 Mbps on a copper wire that cannot support more than 32 MHz (frequencies above this level create electromagnetic emissions). MLT-3 and LANs are discussed in Chapter 13.

Summary ofLine Coding Schemes

We summarize in Table 4.1 the characteristics of the different schemes discussed.

Table 4.1 Summary of line coding schemes

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Scheme** | **Bandwidth (average)** | **Characteristics** |
| **Unipolar** | NRZ | B=N/2 | Costly, no self-synchronization if long 0s or 1s, DC |
| **polar** | NRZ-L | B=N/2 | No self-synchronization if long 0s or 1s, DC |
| NRZ-I | B=N/2 | No self-synchronization for long aS, DC |
| Biphase | B=N | Self-synchronization, no DC, high bandwidth |
| **Bipolar** | AMI | B=NI2 | No self-synchronization for long 0S, DC |
| **Multilevel** | 2BIQ | B=N/4 | No self-synchronization for long same double bits |
| 8B6T | B =3N/4 | Self-synchronization, no DC |
| 4D-PAM5 | B=N/8 | Self-synchronization, no DC |
| **Multiline** | MLT-3 | B=N/3 | No self-synchronization for long 0s |

**Block Coding**

We need redundancy to ensure synchronization and to provide some kind of inherent error detecting. Block coding can give us this redundancy and improve the performance of line coding. In general, block coding changes a block of m bits into a block of n bits, where n is larger than m. Block coding is referred to as an mB/nB encoding technique.

**Block coding is normally referred to as mBlnB coding; it replaces each m~bit group with an n~bit group.**

The slash in block encoding (for example, 4B/5B) distinguishes block encoding from multilevel encoding (for example, 8B6T), which is written without a slash. Block coding normally involves three steps: division, substitution, and combination. In the division step, a sequence of bits is divided into groups of m bits. For example, in 4B/5B encoding, the original bit sequence is divided into 4-bit groups. The heart of block coding is the substitution step. In this step, we substitute an m-bit group for an n-bit group. For example, in 4B/5B encoding we substitute a 4-bit code for a 5-bit group. Finally, the n-bit groups are combined together to form a stream. The new stream has more bits than the original bits. Figure 4.14 shows the procedure.

**4B/5B**

The four binary/five binary (4B/5B) coding scheme was designed to be used in combination with NRZ-I. Recall that NRZ-I has a good signal rate, one-half that of the biphase, but it has a synchronization problem. A long sequence of as can make the receiver clock lose synchronization. One solution is to change the bit stream, prior to encoding with NRZ-I, so that it does not have a long stream of as. The 4B/5B scheme achieves this goal. The block-coded stream does not have more that three consecutive as, as we will see later. At the receiver, the NRZ-I encoded digital signal is first decoded into a stream of bits and then decoded to remove the redundancy. Figure 4.15 shows the idea.

In 4B/5B, the 5-bit output that replaces the 4-bit input has no more than one leading zero (left bit) and no more than two trailing zeros (right bits). So when different groups are combined to make a new sequence, there are never more than three consecutive as. (Note that NRZ-I has no problem with sequences of Is.) Table 4.2 shows the corresponding pairs used in 4B/5B encoding. Note that the first two columns pair a 4-bit group with a 5-bit group. A group of 4 bits can have only 16 different combinations while a group of5 bits can have 32 different combinations. This means that there are 16 groups that are not used for 4B/5B encoding. Some of these unused groups are used for control purposes; the others are not used at all. The latter provide a kind of error detection. Ifa 5-bit group arrives that belongs to the unused portion of the table, the receiver knows that there is an error in the transmission.

|  |  |  |  |
| --- | --- | --- | --- |
| Data Sequence | Encoded Sequence | Control Sequence | Encoded Sequence |
| 0000 | 11110 | Q (Quiet) | 00000 |
| 0001 | 01001 | I (Idle) | 11111 |
| 0010 | 10100 | J (Start delimiter) | 00100 |
| 0011 | 10101 | K (Start delimiter) | 11000 |
| 0100 | 01010 | T (End delimiter) | 10001 |
| 0101 | 01011 | S (Set) | 11001 |
| 0110 | 01110 | R (Reset) | 00111 |
| 0111 | 01111 |  |  |
| 1000 | 10010 |  |  |
| 1001 | 10011 |  |  |
| 1010 | 10110 |  |  |
| 1011 | 10111 |  |  |
| 1100 | 11010 |  |  |
| 1101 | 11011 |  |  |
| 1110 | 11100 |  |  |
| 1111 | 11101 |  |  |

Figure 4.16 shows an example of substitution in 4B/5B coding. 4B/5B encoding solves the problem of synchronization and overcomes one of the deficiencies ofNRZ-1. However, we need to remember that it increases the signal rate of NRZ-1. The redundant bits add 20 percent more baud. Still, the result is less than the biphase scheme which has a signal rate of 2 times that ofNRZ-1. However, 4B/5B block encoding does not solve the DC component problem of NRZ-1. If a DC component is unacceptable, we need to use biphase or bipolar encoding.

**8RIlOR**

The eight binary/ten binary (SBIlOB) encoding is similar to 4B/5B encoding except that a group of 8 bits of data is now substituted by a lO-bit code. It provides greater error detection capability than 4B/5B. The 8BIlOB block coding is actually a combination of5B/6B and 3B/4B encoding, as shown in Figure 4.17.

The most five significant bits of a 10-bit block is fed into the 5B/6B encoder; the least 3 significant bits is fed into a 3B/4B encoder. The split is done to simplify the mapping table. To prevent a long run of consecutive 0s or Is, the code uses a disparity controller which keeps track of excess 0s over Is (or Is over 0s). If the bits in the current block create a disparity that contributes to the previous disparity (either direction), then each bit in the code is complemented (a0 is changed to a 1and a 1is changed to a 0). The coding has 210 - 28 =768 redundant groups that can be used for disparity checking and error detection. In general, the technique is superior to 4B/5B because of better built-in error-checking capability and better synchronization.

Scrambling

Biphase schemes that are suitable for dedicated links between stations in a LAN are not suitable for long-distance communication because of their wide bandwidth requirement. The combination of block coding and NRZ line coding is not suitable for long-distance encoding either, because of the DC component. Bipolar AMI encoding, on the other hand, has a narrow bandwidth and does not create a DC component. However, a long sequence of 0s upsets the synchronization. If we can find a way to avoid a long sequence of 0s in the original stream, we can use bipolar AMI for long distances. We are looking for a technique that does not increase the number of bits and does provide synchronization. We are looking for a solution that substitutes long zero-level pulses with a combination of other levels to provide synchronization. One solution is called scrambling. We modify part of the AMI rule to include scrambling, as shown in Figure 4.18. Note that scrambling, as opposed to block coding, is done at the same time as encoding. The system needs to insert the required pulses based on the defined scrambling rules. Two common scrambling techniques are B8ZS and HDB3.

**R8ZS**

Bipolar with S-zero substitution (BSZS) is commonly used in North America. In this technique, eight consecutive zero-level voltages are replaced by the sequence

OOOVBOVB. The V in the sequence denotes violation; this is a nonzero voltage that breaks an AMI rule of encoding (opposite polarity from the previous). The B in the sequence denotes bipolm; which means a nonzero level voltage in accordance with the AMI rule. There are two cases, as shown in Figure 4.19.

Note that the scrambling in this case does not change the bit rate. Also, the technique balances the positive and negative voltage levels (two positives and two negatives), which means that the DC balance is maintained. Note that the substitution may change the polarity ofa1 because, after the substitution, AMI needs to follow its rules.

B8ZS substitutes eight consecutive zeros with OOOVBOVB.

One more point is worth mentioning. The letter V (violation) or B (bipolar) here is relative. The V means the same polarity as the polarity ofthe previous nonzero pulse; B means the polarity opposite to the polarity ofthe previous nonzero pulse.

HDB3

High-density bipolar3-zero (HDB3) is commonly used outside of North America. In this technique, which is more conservative than B8ZS, four consecutive zero-level voltages are replaced with a sequence of OOOV or BOO\: The reason for two different substitutions is to maintain the even number of nonzero pulses after each substitution. The two rules can be stated as follows:

1. If the number of nonzero pulses after the last substitution is odd, the substitution pattern will be OOOV, which makes the total number of nonzero pulses even.

2. If the number of nonzero pulses after the last substitution is even, the substitution pattern will be BOOV, which makes the total number of nonzero pulses even.

There are several points we need to mention here. First, before the first substitution, the number of nonzero pulses is even, so the first substitution is BODY. After this substitution, the polarity of the 1 bit is changed because the AMI scheme, after each substitution, must follow its own rule. After this bit, we need another substitution, which is OOOV because we have only one nonzero pulse (odd) after the last substitution. The third substitution is BOOV because there are no nonzero pulses after the second substitution (even).

**HDB3 substitutes four consecutive zeros with OOOV or BOOV depending on the number of nonzero pulses after the last substitution.**

4.2 ANALOG-TO-DIGITAL CONVERSION

The techniques described in Section 4.1 convert digital data to digital signals. Sometimes, however, we have an analog signal such as one created by a microphone or camera. We have seen in Chapter 3 that a digital signal is superior to an analog signal. The tendency today is to change an analog signal to digital data. In this section we describe two techniques, pulse code modulation and delta modulation. After the digital data are created (digitization), we can use one of the techniques described in Section 4.1 to convert the digital data to a digital signal.

Pulse Code Modulation (PCM)

The most common technique to change an analog signal to digital data (digitization) is called pulse code modulation (PCM). A PCM encoder has three processes, as shown in Figure 4.21.

1. The analog signal is sampled.

2. The sampled signal is quantized.

3. The quantized values are encoded as streams of bits.

Sampling

The first step in PCM is sampling. The analog signal is sampled every Ts s, where Ts is the sample interval or period. The inverse of the sampling interval is called the sampling rate or sampling frequency and denoted by is, where is = IITs ' There are three sampling methods-ideal, natural, and flat-top-as shown in Figure 4.22. In ideal sampling, pulses from the analog signal are sampled. This is an ideal sampling method and cannot be easily implemented. In natural sampling, a high-speed switch is turned on for only the small period of time when the sampling occurs. The result is a sequence of samples that retains the shape of the analog signal. The most common sampling method, called sample and hold, however, creates flat-top samples by using a circuit. The sampling process is sometimes referred to as pulse amplitude modulation (PAM). We need to remember, however, that the result is still an analog signal with nonintegral values.

Sampling Rate One important consideration is the sampling rate or frequency. What are the restrictions on Ts? This question was elegantly answered by Nyquist. According to the Nyquist theorem, to reproduce the original analog signal, one necessary condition is that the sampling rate be at least twice the highest frequency in the original signal.

**According to the Nyquist theorem, the sampling rate must be at least 2 times the highest frequency contained in the signal.**

We need to elaborate on the theorem at this point. First, we can sample a signal only if the signal is band-limited. In other words, a signal with an infinite bandwidth cannot be sampled. Second, the sampling rate must be at least 2 times the highest frequency, not the bandwidth. If the analog signal is low-pass, the bandwidth and the highest frequency are the same value. If the analog signal is bandpass, the bandwidth value is lower than the value of the maximum frequency. Figure 4.23 shows the value of the sampling rate for two types of signals.