CHAPTER 1

Introduction

Data communications and networking are changing the way we do business and the way we live. Business decisions have to be made ever more quickly, and the decision makers require immediate access to accurate information. Why wait a week for that report from Germany to arrive by mail when it could appear almost instantaneously through computer networks? Businesses today rely on computer networks and internetworks. But before we ask how quickly we can get hooked up, we need to know how networks operate, what types of technologies are available, and which design best fills which set of needs.

The development of the personal computer brought about tremendous changes for business, industry, science, and education. A similar revolution is occurring in data communications and networking. Technological advances are making it possible for communications links to carry more and faster signals. As a result, services are evolving to allow use of this expanded capacity. For example, established telephone services such as conference calling, call waiting, voice mail, and caller **ID** have been extended.

Research in data communications and networking has resulted in new technolo• gies. One goal is to be able to exchange data such as text, audio, and video from all points in the world. We want to access the Internet to download and upload information quickly and accurately and at any time.

This chapter addresses four issues: data communications, networks, the Internet, and protocols and standards. First we give a broad definition of data communications. Then we define networks as a highway on which data can travel. The Internet is dis• cussed as a good example of an internetwork (i.e., a network of networks). Finally, we discuss different types of protocols, the difference between protocols and standards, and the organizations that set those standards.

1.1 DATA COMMUNICATIONS

When we communicate, we are sharing information. This sharing can be local or remote. Between individuals, local communication usually occurs face to face, while remote communication takes place over distance. The term *telecommunication,* which

includes telephony, telegraphy, and television, means communication at a distance *(tele* is Greek for "far").

The word *data* refers to information presented in whatever form is agreed upon by the parties creating and using the data.

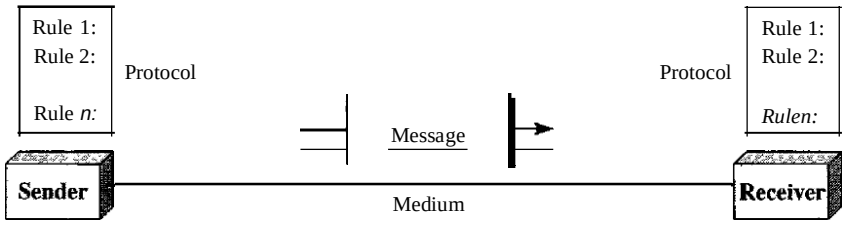
Data communications are the exchange of data between two devices via some form of transmission medium such as a wire cable. For data communications to occur, the communicating devices must be part of a communication system made up of a com• bination of hardware (physical equipment) and software (programs). The effectiveness of a data communications system depends on four fundamental characteristics: deliv• ery, accuracy, timeliness, and jitter.

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 case of video and audio, timely delivery means delivering data as they are produced, in the same order that they are produced, and without signifi• cant delay. 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. For example, let us assume that video packets are sent every 3D ms. If some of the packets arrive with 3D-ms delay and others with 4D-ms delay, an uneven quality in the video is the result.

COinponents

A data communications system has five components (see Figure 1.1).

Figure 1.1 *Five components ofdata communication*



I. 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 com• puter, workstation, telephone handset, video camera, and so on.
3. Receiver. The receiver is the device that receives the message. It can be a com• puter, 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 of transmission media include 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 repre• sents an agreement between the communicating devices. Without a protocol, two devices may be connected but not communicating, just as a person speaking French cannot be understood by a person who speaks only Japanese.

Data Representation

Information today comes in different forms such as text, numbers, images, audio, and video.

Text

In data communications, text is represented as a bit pattern, a sequence of bits (Os or Is). Different sets of bit patterns have been designed to represent text symbols. Each set is called a code, and the process of representing symbols is called coding. Today, the prevalent coding system is called Unicode, which uses 32 bits to represent a symbol or character used in any language in the world. The American Standard Code for Infor• mation Interchange (ASCII), developed some decades ago in the United States, now constitutes the first 127 characters in Unicode and is also referred to as Basic Latin. Appendix A includes part of the Unicode.

Numbers

Numbers are also represented by bit patterns. However, a code such as ASCII is not used to represent numbers; the number is directly converted to a binary number to simplify mathematical operations. Appendix B discusses several different numbering systems.

Images

Images are also represented by bit patterns. In its simplest form, an image is composed of a matrix of pixels (picture elements), where each pixel is a small dot. The size of the pixel depends on the *resolution.* For example, an image can be divided into 1000 pixels or 10,000 pixels. In the second case, there is a better representation of the image (better resolution), but more memory is needed to store the image.

After an image is divided into pixels, each pixel is assigned a bit pattern. The size and the value of the pattern depend on the image. For an image made of only black• and-white dots (e.g., a chessboard), a I-bit pattern is enough to represent a pixel.

If an image is not made of pure white and pure black pixels, you can increase the size of the bit pattern to include gray scale. For example, to show four levels of gray scale, you can use 2-bit patterns. A black pixel can be represented by 00, a dark gray pixel by 01, a light gray pixel by 10, and a white pixel by 11.

There are several methods to represent color images. One method is called RGB, so called because each color is made of a combination of three primary colors: *red,* green, and blue. The intensity of each color is measured, and a bit pattern is assigned to it. Another method is called YCM, in which a color is made of a combination of three other primary colors: yellow, cyan, and magenta.

Audio

Audio refers to the recording or broadcasting of sound or music. Audio is by nature different from text, numbers, or images. It is continuous, not discrete. Even when we

use a microphone to change voice or music to an electric signal, we create a continuous signal. In Chapters 4 and 5, we learn how to change sound or music to a digital or an analog signal.

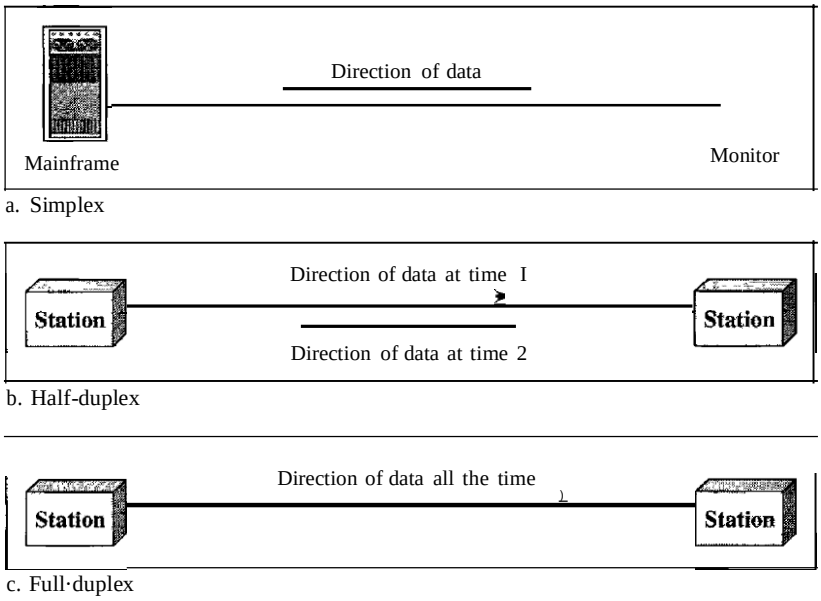
Video

Video refers to the recording or broadcasting of a picture or movie. Video can either be produced as a continuous entity (e.g., by a TV camera), or it can be a combination of images, each a discrete entity, arranged to convey the idea of motion. Again we can change video to a digital or an analog signal, as we will see in Chapters 4 and 5.

Data Flow

Communication between two devices can be simplex, half-duplex, or full-duplex as shown in Figure 1.2.

Figure 1.2 *Data flow (simplex, half-duplex, andfull-duplex)*



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 (see Figure 1.2a).

Keyboards and traditional monitors are examples of simplex devices. The key• board can only introduce input; the monitor can only accept output. The simplex mode can use the entire capacity of the channel to send data in one direction.

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 (see Figure 1.2b).

The half-duplex mode is like a one-lane road with traffic allowed in both direc• tions. When cars are traveling in one direction, cars going the other way must wait. In a half-duplex transmission, the entire capacity of a channel is taken over by whichever of the two devices is transmitting at the time. Walkie-talkies and CB (citizens band) radios are both half-duplex systems.

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.

Full-Duplex

In full-duplex mode (also called duplex), both stations can transmit and receive simul• taneously (see Figure 1.2c).

The full-duplex mode is like a two-way street with traffic flowing in both direc• tions at the same time. 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.

One common example of full-duplex communication is the telephone network. When two people are communicating by a telephone line, both can talk and listen at the same time.

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.

Distributed Processing

Most networks use distributed processing, in which 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.

Network Criteria

A network must be able to meet a certain number of criteria. The most important of these are performance, reliability, and security.

Performance

Performance can be measured in many ways, including transit time and response time. 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. The performance of a network depends on a number of factors, including the number of users, the type of transmission medium, the capabilities of the connected hardware, and the efficiency of the software.

Performance is often evaluated by two networking metrics: throughput and delay. We often need more throughput and less delay. However, these two criteria are often contradictory. If we try to send more data to the network, we may increase throughput but we increase the delay because of traffic congestion in the network.

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.

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.

Physical Structures

Before discussing networks, we need to define some network attributes.

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. For visualization purposes, it is simplest to imagine any link as a line drawn between two points. For communication to occur, two devices must be connected in some way to the same link at the same time. 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 (see Figure 1.3a). When you change television channels by infrared remote control, you are establishing a point-to-point connection between the remote control and the television's control system.

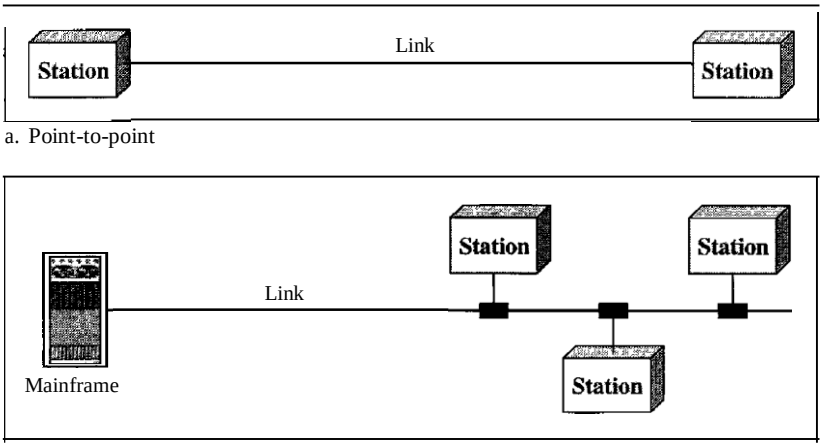
Multipoint A multipoint (also called multidrop) connection is one in which more than two specific devices share a single link (see Figure 1.3b).

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.

Physical Topology

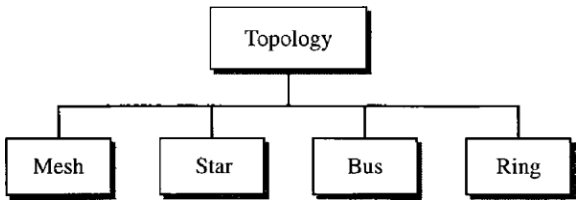
The term *physical topology* refers to the way in which a network is laid out physically.: Two or more devices connect to a link; two or more links form a topology. The topology

Figure 1.3 *Types ofconnections: point-to-point and multipoint*



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 (see Figure 1.4).

Figure **1.4** *Categories oftopology*



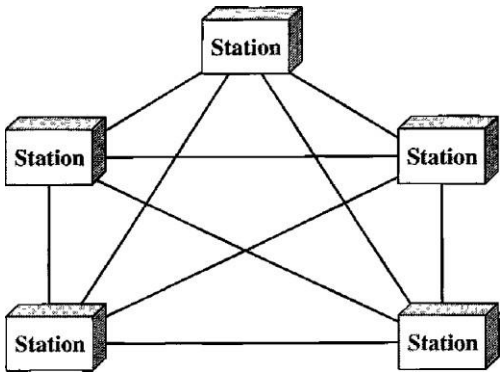
Mesh In a mesh topology, every device has a dedicated point-to-point link to every other device. The term *dedicated* means that the link carries traffic only between the two devices it connects. To find the number of physical links in a fully connected mesh network with *n* nodes, we first consider that each node must be connected to every other node. Node 1 must be connected to *n* - I nodes, node 2 must be connected to *n* - 1 nodes, and finally node *n* must be connected to *n* - 1 nodes. We need *n(n* - 1) physical links. However, if each physical link allows communication in both directions (duplex mode), we can divide the number of links by 2. In other words, we can say that in a mesh topology, we need

$$n(n-1)/2$$

duplex-mode links.

To accommodate that many links, every device on the network must have *n* - 1 input/output *(VO)* ports (see Figure 1.5) to be connected to the other *n* - 1 stations.

Figure 1.5 *A fully connected mesh topology (five devices)*



A mesh offers several advantages over other network topologies. First, the use of dedicated links guarantees that each connection can carry its own data load, thus elimi• nating the traffic problems that can occur when links must be shared by multiple devices. Second, a mesh topology is robust. If one link becomes unusable, it does not incapaci• tate the entire system. Third, there is the advantage of privacy or security. When every message travels along a dedicated line, only the intended recipient sees it. Physical boundaries prevent other users from gaining access to messages. Finally, point-to-point links make fault identification and fault isolation easy. Traffic can be routed to avoid links with suspected problems. This facility enables the network manager to discover the precise location of the fault and aids in finding its cause and solution.

The main disadvantages of a mesh are related to the amount of cabling and the number of I/O ports required. First, because every device must be connected to every other device, installation and reconnection are difficult. Second, the sheer bulk of the wiring can be greater than the available space (in walls, ceilings, or floors) can accom• modate. Finally, the hardware required to connect each link (I/O ports and cable) can be prohibitively expensive. For these reasons a mesh topology is usually implemented in a limited fashion, for example, as a backbone connecting the main computers of a hybrid network that can include several other topologies.

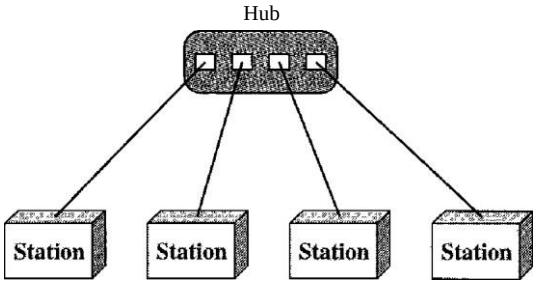
One practical example of a mesh topology is the connection of telephone regional offices in which each regional office needs to be connected to every other regional office.

Star Topology In a star topology, each device has a dedicated point-to-point link only to a central controller, usually called a hub. The devices are not directly linked to one another. Unlike a mesh topology, a star topology does not allow direct traffic between devices. The controller acts as an exchange: If one device wants to send data to another, it sends the data to the controller, which then relays the data to the other con• nected device (see Figure 1.6) .

A star topology is less expensive than a mesh topology. 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. Far less cabling needs to be housed, and addi• tions, moves, and deletions involve only one connection: between that device and the hub.

Other 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

Figure 1.6 *A star topology connecting four stations*



fault isolation. As long as the hub is working, it can be used to monitor link problems and bypass defective links.

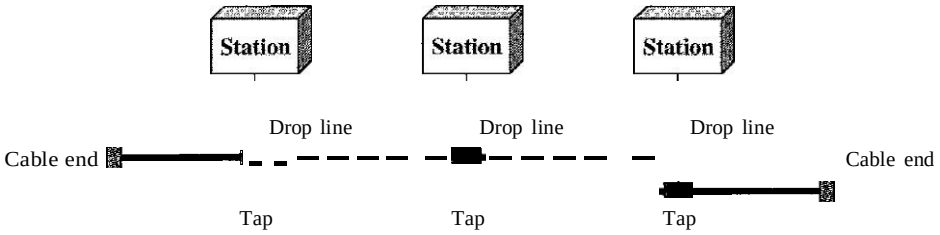
One big disadvantage of a star topology is the dependency of the whole topologyon one single point, the hub. If the hub goes down, the whole system is dead.

Although a star requires far less cable than a mesh, each node must be linked to a central hub. For this reason, often more cabling is required in a star than in some other topologies (such as ring or bus).

The star topology is used in local-area networks (LANs), as we will see in Chapter 13. High-speed LANs often use a star topology with a central hub.

Bus Topology The preceding examples all describe point-to-point connections. A **bus topology,** on the other hand, is multipoint. One long cable acts as a **backbone** to link all the devices in a network (see Figure 1.7).

Figure 1.7 *A bus topology connecting three stations*



Nodes are connected to the bus cable by drop lines and taps. A drop line is a con• nection running between the device and the main cable. 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. As a signal travels along the backbone, some of its energy is transformed into heat. Therefore, it becomes weaker and weaker as it travels farther and farther. For this reason there is a limit on the number of taps a bus can support and on the distance between those taps.

Advantages of a bus topology include ease of installation. Backbone cable can be laid along the most efficient path, then connected to the nodes by drop lines of various lengths. In this way, a bus uses less cabling than mesh or star topologies. In a star, for example, four network devices in the same room require four lengths of cable reaching

all the way to the hub. 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 near• est point on the backbone.

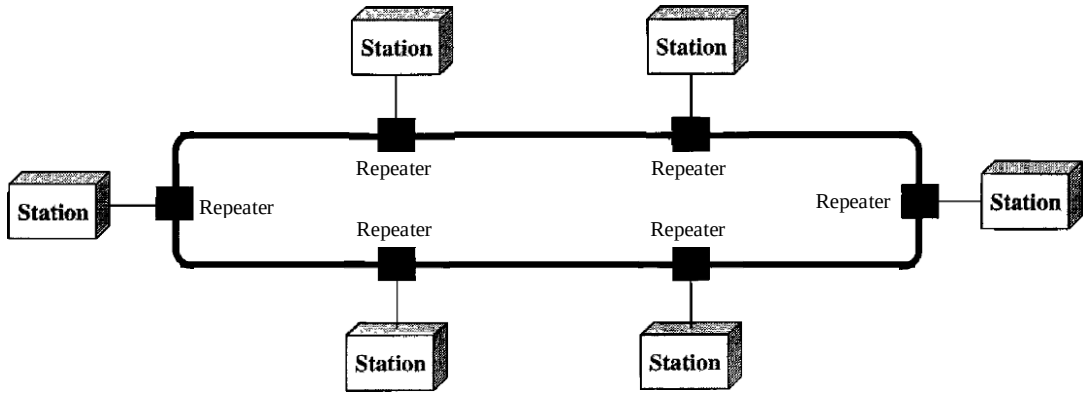
Disadvantages include difficult reconnection and fault isolation. A bus is usually designed to be optimally efficient at installation. It can therefore be difficult to add new devices. Signal reflection at the taps can cause degradation in quality. This degradation can be controlled by limiting the number and spacing of devices connected to a given length of cable. Adding new devices may therefore require modification or replacement of the backbone.

In addition, a fault or break in the bus cable stops all transmission, even between devices on the same side of the problem. The damaged area reflects signals back in the direction of origin, creating noise in both directions.

Bus topology was the one of the first topologies used in the design of early local• area networks. Ethernet LANs can use a bus topology, but they are less popular now for reasons we will discuss in Chapter 13.

Ring Topology In a ring topology, each device has a dedicated point-to-point con• nection with only the two devices on either side of it. A signal is passed along the ringin 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 (see Figure 1.8).

Figure 1.8 *A ring topology connecting six stations*



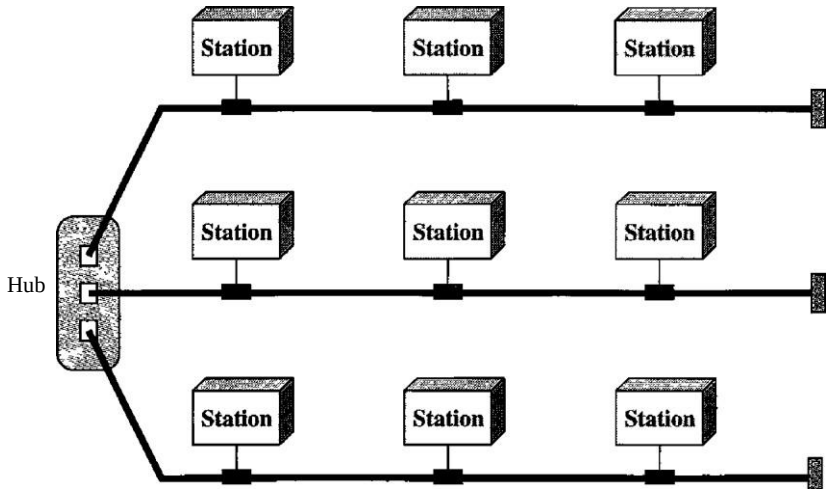
A ring is relatively easy to install and reconfigure. Each device is linked to only its immediate neighbors (either physically or logically). To add or delete a device requires changing only two connections. The only constraints are media and traffic consider• ations (maximum ring length and number of devices). In addition, fault isolation is sim• plified. Generally in a ring, a signal is circulating at all times. If one device does not receive a signal within a specified period, it can issue an alarm. The alarm alerts the network operator to the problem and its location.

However, unidirectional traffic can be a disadvantage. In a simple ring, a break in the ring (such as a disabled station) can disable the entire network. This weakness can be solved by using a dual ring or a switch capable of closing off the break.

Ring topology was prevalent when IBM introduced its local-area network Token Ring. Today, the need for higher-speed LANs has made this topology less popular.

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 as shown in Figure 1.9.

Figure 1.9 A *hybrid topology: a star backbone with three bus networks*



Network Models

Computer networks are created by different entities. Standards are needed so that these heterogeneous networks can communicate with one another. The two best-known standards are the OSI model and the Internet model. In Chapter 2 we discuss these two models. The OSI (Open Systems Interconnection) model defines a seven-layer network; the Internet model defines a five-layer network. This book is based on the Internet model with occasional references to the OSI model.

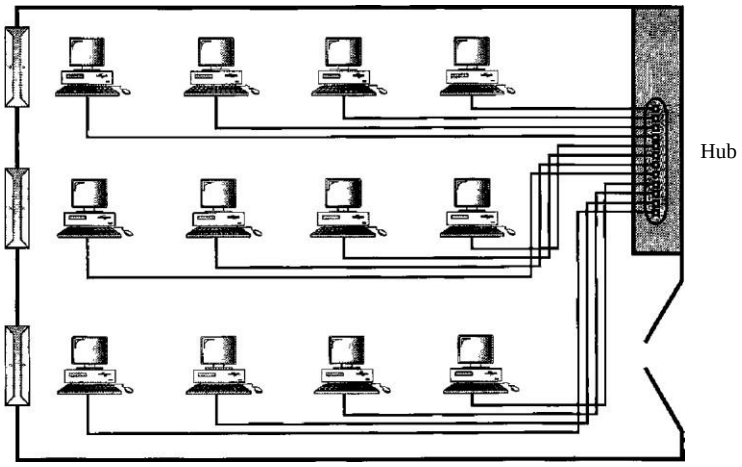
Categories of Networks

Today when we speak of networks, we are generally referring to two primary categories: local-area networks and wide-area networks. The category into which a network falls is determined by its size. A LAN normally covers an area less than 2 mi; a WAN canbe worldwide. Networks of a size in between are normally referred to as metropolitan area networks and span tens of miles.

Local Area Network

A local area network (LAN) is usually privately owned and links the devices in a single office, building, or campus (see Figure 1.10). Depending on the needs of an organization and the type of technology used, a LAN can be as simple as two PCs and a printer in someone's home office; or it can extend throughout a company and include audio and video peripherals. Currently, LAN size is limited to a few kilometers.

Figure 1.10 *An isolated IAN connecting* 12 *computers to a hub in a closet*



LANs are designed to allow resources to be shared between personal computers or workstations. The resources to be shared can include hardware (e.g., a printer), software (e.g., an application program), or data. A common example of a LAN, found in many business environments, links a workgroup of task-related computers, for example, engi• neering workstations or accounting PCs. One of the computers may be given a large• capacity disk drive and may become a server to clients. Software can be stored on this central server and used as needed by the whole group. In this example, the size of the LAN may be determined by licensing restrictions on the number of users per copy of soft• ware, or by restrictions on the number of users licensed to access the operating system.

In addition to size, LANs are distinguished from other types of networks by their transmission media and topology. In general, a given LAN will use only one type of transmission medium. The most common LAN topologies are bus, ring, and star.

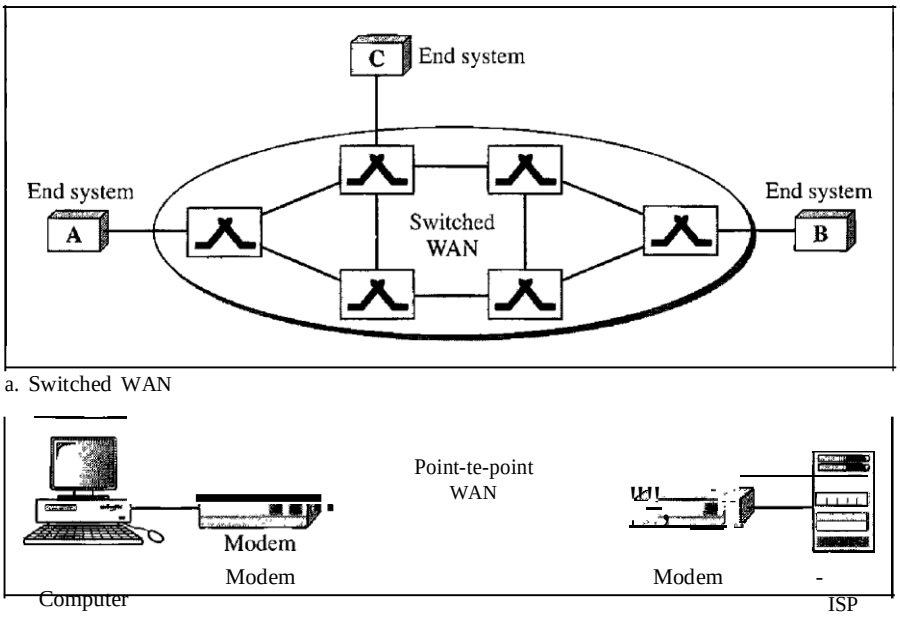
Early LANs had data rates in the 4 to 16 megabits per second (Mbps) range. Today, however, speeds are normally 100 or 1000 Mbps. LANs are discussed at length in Chapters 13, 14, and 15.

Wireless LANs are the newest evolution in LAN technology. We discuss wireless LANs in detail in Chapter 14.

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 conti• nent, or even the whole world. In Chapters 17 and 18 we discuss wide-area networks in greater detail. A WAN can be as complex as the backbones that connect the Internet or as simple as a dial-up line that connects a home computer to the Internet. We normally refer to the first as a switched WAN and to the second as a point-to-point WAN (Figure 1.11). The switched WAN connects the end systems, which usually comprise a router (internet• working connecting device) 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.

Figure 1.11 *WANs: a switched WAN and a point-to-point WAN*



An early example of a switched WAN is X.25, a network designed to provide connectivity between end users. As we will see in Chapter 18, 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. We will discuss ATM in Chapter 18. Another example ofWANs is the wireless WAN that is becoming more and more popular. We discuss wireless WANs and their evolution in Chapter 16.

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. We discuss DSL lines and cable TV networks in Chapter 9.

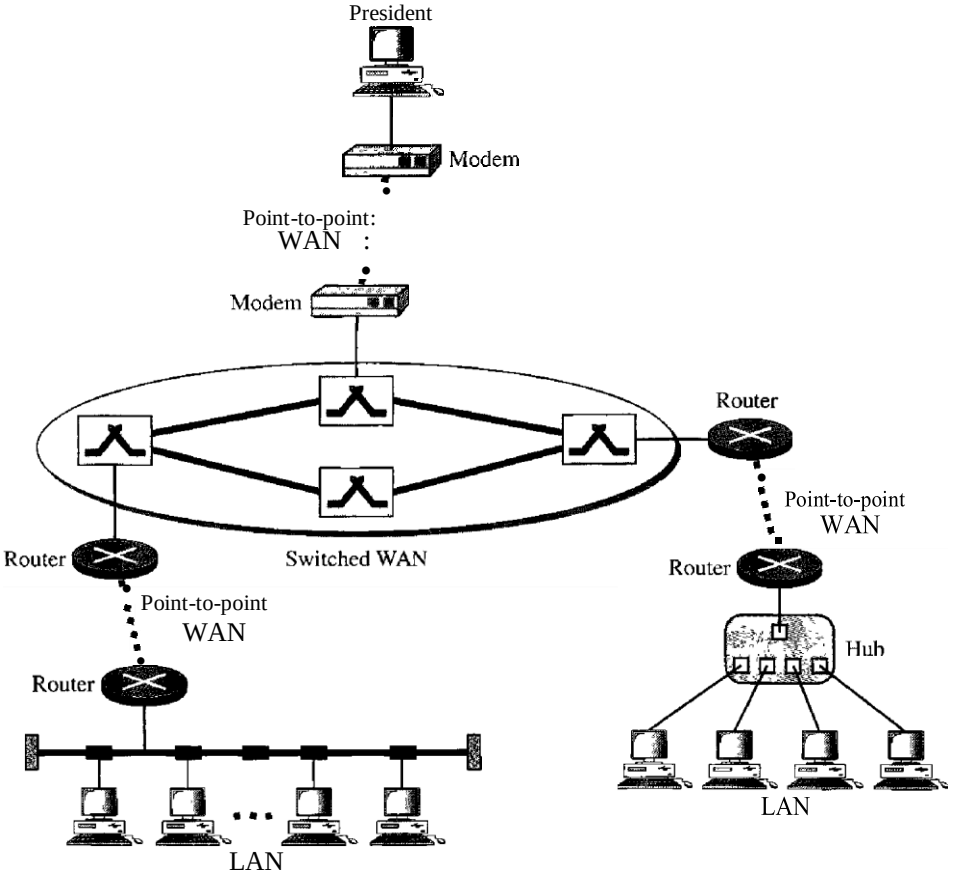
Interconnection of Networks: Internetwork

Today, it is very rare to see a LAN, a MAN, or a LAN in isolation; they are connected to one another. When two or more networks are connected, they become an internetwork, or internet.

As an example, assume that an organization has two offices, one on the east coast and the other on the west coast. The established office on the west coast has a bus topology LAN; the newly opened office on the east coast has a star topology LAN. The president of the company lives somewhere in the middle and needs to have control over the company

from her horne. To create a backbone WAN for connecting these three entities (two LANs and the president's computer), a switched WAN (operated by a service provider such as a telecom company) has been leased. To connect the LANs to this switched WAN, however, three point-to-point WANs are required. These point-to-point WANs can be a high-speed DSL line offered by a telephone company or a cable modern line offered by a cable TV provider as shown in Figure 1.12.

Figure 1.12 *A heterogeneous network made offour WANs and two LANs*



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. Count the ways you've used the Internet recently. Perhaps you've sent electronic mail (e-mail) to a business associate, paid a utility bill, read a newspaper from a distant city, or looked up a local movie schedule-all by using the Internet. Or maybe you researched a medical topic, booked a hotel reservation, chatted with a fellow Trekkie, or comparison-shopped for a car. 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 com• municate with each other. The most notable internet is called the Internet (uppercase letter I), a collaboration of more than hundreds of thousands of interconnected net• works. Private individuals as well as various organizations such as government agen• cies, schools, research facilities, corporations, and libraries in more than 100 countries use the Internet. Millions of people are users. Yet this extraordinary communication sys•tem only came into being in 1969.

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

In 1967, at an Association for Computing Machinery (ACM) meeting, ARPA pre• sented its ideas for ARPANET, a small network of connected computers. The idea was that each host computer (not necessarily from the same manufacturer) 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 commu• nicate with other IMPs as well as with its own attached host.

By 1969, ARPANET was a reality. Four nodes, at the University of California atLos 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 com• munication 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 land• mark 1973 paper outlined the protocols to achieve end-to-end delivery of packets. This paper on Transmission Control Protocol (TCP) included concepts such as encapsula• tion, the datagram, and the functions of a gateway.

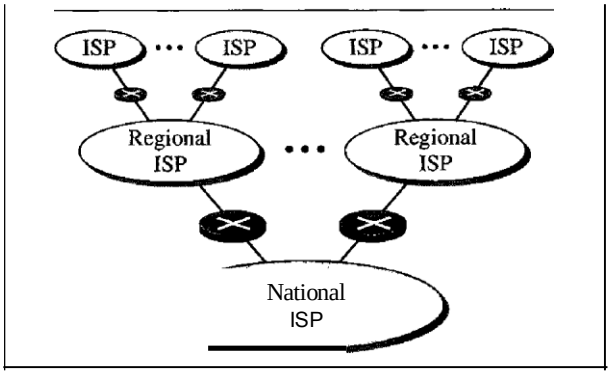
Shortly thereafter, authorities made a decision to split TCP into two protocols: Transmission Control Protocol (TCP) and Internetworking Protocol (lP). IP would handle datagram routing while TCP would be responsible for higher-level functions such as segmentation, reassembly, and error detection. The internetworking protocol became known as TCPIIP.

The Internet Today

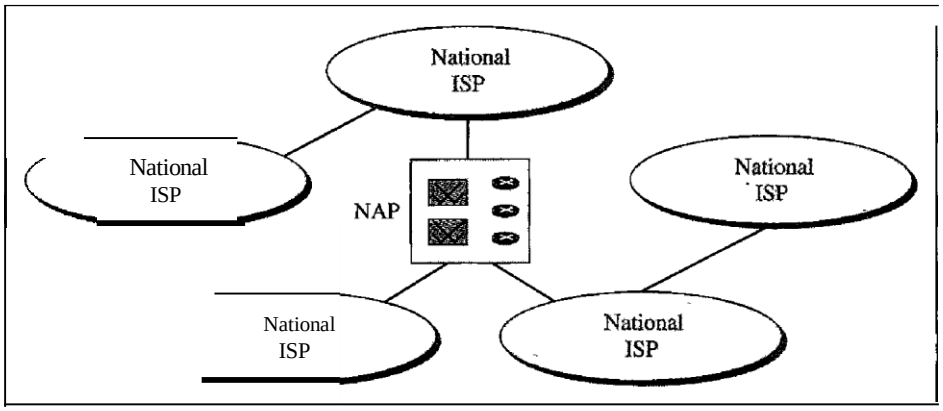
The Internet has come a long way since the 1960s. The Internet today is not a simple hierarchical structure. It is made up of many wide- and local-area networks joined by connecting devices and switching stations. It is difficult to give an accurate represen• tation of the Internet because it is continually changing-new networks are being added, existing networks are adding addresses, and networks of defunct companies are being removed. 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. The Internet today is run by private companies, not the government. Figure 1.13 shows a conceptual (not geographic) view of the Internet.

Figure 1.13 *Hierarchical organization of the Internet*



a. Structure of a national ISP



b. Interconnection of national ISPs

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 main• tained by specialized companies. There are many national ISPs operating in North America; some of the most well known are SprintLink, PSINet, UUNet Technology, AGIS, and internet Mel. To provide connectivity between the end users, these back• bone 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. Note that in this sense, a local ISP can be a company that just provides Internet services, a corporation with a network that supplies services to its own employees, or a nonprofit organization, such as a college or a university, that runs its own network. Each of these local ISPs can be connected to a regional or national service provider.

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

In computer networks, communication occurs between entities in different systems. An entity is anything capable of sending or receiving information. However, two entities can• not 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 it is communicated. The key elements of a protocol are syntax, semantics, and timing.

- 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.
- 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?
- 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 1 Mbps, 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").

- 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.
- De jure. Those standards that have been legislated by an officially recognized body are de jure standards.

Standards Organizations

Standards are developed through the cooperation of standards creation committees, forums, and government regulatory agencies.

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:

- 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.
- 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).
- American National Standards Institute (ANSI). Despite its name, the American National Standards Institute is a completely private, nonprofit 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.
- 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.
- Electronic Industries Association (EIA). Aligned with ANSI, the Electronic Industries Association is a nonprofit 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 informationtechnology, the EIA has made significant contributions by defining physical connec• tion interfaces and electronic signaling specifications for data communication.

Forums

Telecommunications technology development is moving faster than the ability of stan• dards 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 pur• pose of these agencies is to protect the public interest by regulating radio, television, and wire/cable communications. The FCC has authority over interstate and interna• tional 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 fol• lowed. There is a strict procedure by which a specification attains Internet standard status. A specification begins asan Internet draft. An **Internet draft** is a working docu• ment (a work in progress) with no official status and a 6-month lifetime. Upon recom• mendation 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.

1.5 SUMMARY

- Data communications are the transfer of data from one device to another via some form of transmission medium.
- A data communications system must transmit data to the correct destination in an accurate and timely manner.
- The five components that make up a data communications system are the message, sender, receiver, medium, and protocol.

- Text, numbers, images, audio, and video are different forms of information.
- Data flow between two devices can occur in one of three ways: simplex, half-duplex, or full-duplex.
- A network is a set of communication devices connected by media links.
- In a point-to-point connection, two and only two devices are connected by a dedicated link. In a multipoint connection, three or more devices share a link.
- Topology refers to the physical or logical arrangement of a network. Devices may be arranged in a mesh, star, bus, or ring topology.
- A network can be categorized as a local area network or a wide area network.
- A LAN is a data communication system within a building, plant, or campus, or between nearby buildings.
- A WAN is a data communication system spanning states, countries, or the whole world.
- An internet is a network of networks.
- The Internet is a collection of many separate networks.
- There are local, regional, national, and international Internet service providers.
- A protocol is a set of rules that govern data communication; the key elements of a protocol are syntax, semantics, and timing.

- Standards are necessary to ensure that products from different manufacturers can work together as expected.
- The ISO, ITD-T, ANSI, IEEE, and EIA are some of the organizations involved in standards creation.
- Forums are special-interest groups that quickly evaluate and standardize new technologies.
- A Request for Comment is an idea or concept that is a precursor to an Internet standard.

1.6 PRACTICE SET

Review Questions

1. Identify the five components of a data communications system.
2. What are the advantages of distributed processing?
3. What are the three criteria necessary for an effective and efficient network?
4. What are the advantages of a multipoint connection over a point-to-point connection?
5. What are the two types of line configuration?
6. Categorize the four basic topologies in terms of line configuration.
7. What is the difference between half-duplex and full-duplex transmission modes?
8. Name the four basic network topologies, and cite an advantage of each type.
9. For *n* devices in a network, what is the number of cable links required for a mesh, ring, bus, and star topology?
10. What are some of the factors that determine whether a communication system is a LAN or WAN?
11. What is an internet? What is the Internet?
12. Why are protocols needed?
13. Why are standards needed?

Exercises

14. What is the maximum number of characters or symbols that can be represented by Unicode?
15. A color image uses 16 bits to represent a pixel. What is the maximum number of different colors that can be represented?
16. Assume six devices are arranged in a mesh topology. How many cables are needed? How many ports are needed for each device?
17. For each of the following four networks, discuss the consequences if a connection fails.
    a. Five devices arranged in a mesh topology
    b. Five devices arranged in a star topology (not counting the hub)
    c. Five devices arranged in a bus topology
    d. Five devices arranged in a ring topology

18. You have two computers connected by an Ethernet hub at home. Is this a LAN, a MAN, or a WAN? Explain your reason.
19. In the ring topology in Figure 1.8, what happens if one of the stations is unplugged?
20. **In** the bus topology in Figure 1.7, what happens if one ofthe stations is unplugged?
21. Draw a hybrid topology with a star backbone and three ring networks.
22. Draw a hybrid topology with a ring backbone and two bus networks.
23. Performance is inversely related to delay. When you use the Internet, which of the following applications are more sensitive to delay?
    a. Sending an e-mail
    b. Copying a file
    c. Surfing the Internet
24. When a party makes a local telephone call to another party, is this a point-to-point or multipoint connection? Explain your answer.
25. Compare the telephone network and the Internet. What are the similarities? What are the differences?

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.

We can compare the task of networking to the task of solving a mathematics problem with a computer. The fundamental job of solving the problem with a computer is done by computer hardware. However, this is a very tedious task if only hardware is involved. We would need switches for every memory location to store and manipulate data. The task is much easier if software is available. At the highest level, a program can directthe problem-solving process; the details of how this is done by the actual hardware can be left to the layers of software that are called by the higher levels.

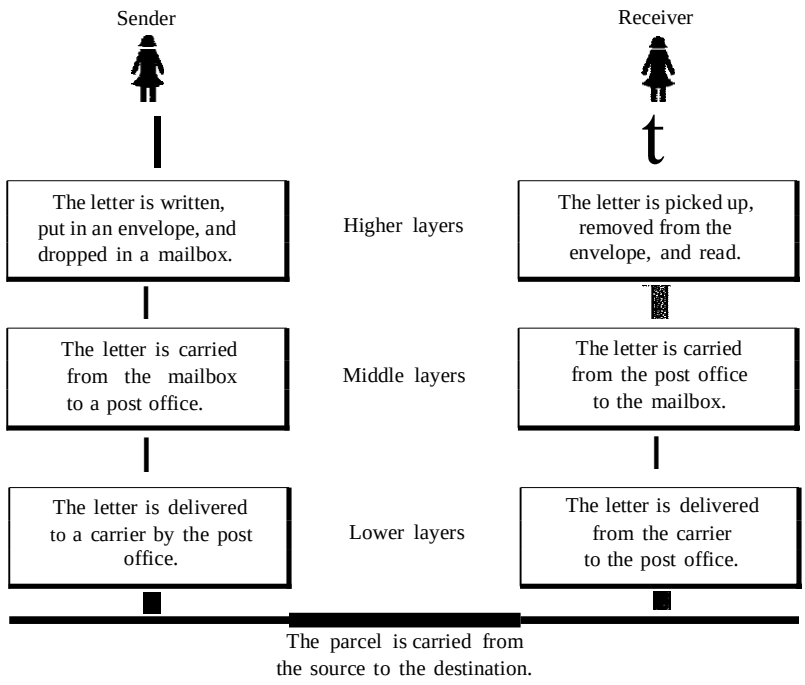
Compare this to a service provided by a computer network. For example, the task of sending an e-mail from one point in the world to another can be broken into several tasks, each performed by a separate software package. Each software package uses the services of another software package. At the lowest 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. Detailed descriptions of these layers follow in later chapters.

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. Fig•ure 2.1 shows the steps in this task.

Figure 2.1 *Tasks involved in sending a letter*



Sender, Receiver, and Carrier

In Figure 2.1 we have a sender, a receiver, and a carrier that transports the letter. There is a hierarchy of tasks.

At the Sellder Site

Let us first describe, in order, the activities that take place at the sender site.

- Higher layer. The sender writes the letter, inserts the letter in an envelope, writes the sender and receiver addresses, and drops the letter in a mailbox.
- Middle layer. The letter is picked up by a letter carrier and delivered to the post office.
- Lower layer. The letter is sorted at the post office; a carrier transports the letter.

011 the Way

The letter is then on its way to the recipient. On the way to the recipient's local post office, the letter may actually go through a central office. In addition, it may be trans• ported by truck, train, airplane, boat, or a combination of these.

At the Receiver Site

- Lower layer. The carrier transports the letter to the post office.
- Middle layer. The letter is sorted and delivered to the recipient's mailbox.
- Higher layer. The receiver picks up the letter, opens the envelope, and reads it.

Hierarchy

According to our analysis, there are three different activities at the sender site and another three activities at the receiver site. The task of transporting the letter between the sender and the receiver is done by the carrier. Something that is not obvious immediately is that the tasks must be done in the order given in the hierarchy. At the sender site, the letter must be written and dropped in the mailbox before being picked up by the letter carrier and delivered to the post office. At the receiver site, the letter must be dropped in the recipient mailbox before being picked up and read by the recipient.

Services

Each layer at the sending site uses the services of the layer immediately below it. The sender at the higher layer uses the services of the middle layer. The middle layer uses the services of the lower layer. The lower layer uses the services of the carrier.

The layered model that dominated data communications and networking literature before 1990 was the Open Systems Interconnection (OSI) model. Everyone believedthat the OSI model would become the ultimate standard for data communications, but this did not happen. The TCPIIP protocol suite became the dominant commercial archi• tecture 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 TCPIIP as a protocol suite.

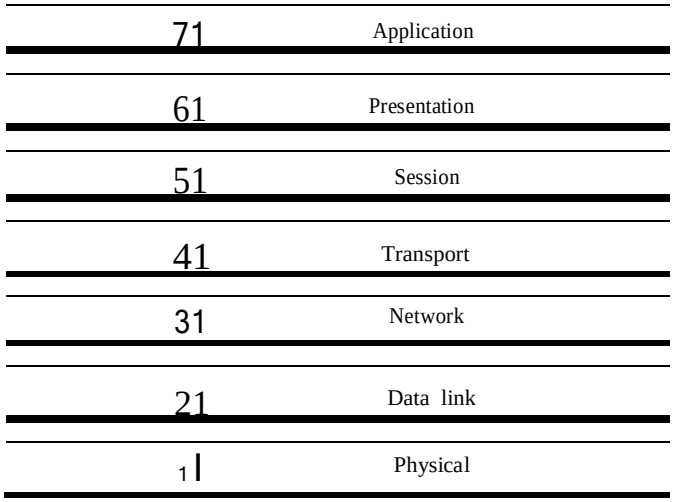
2.2 THE OSI MODEL

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 archi• tecture. The purpose of the OSI model is to show how to facilitate communication between different systems without requiring changes to the logic of the underlying hard• ware and software. The OSI model is not a protocol; it is a model for understanding and designing a network architecture that is flexible, robust, and interoperable.

ISO is the organization. OSI is the model.

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 sep• arate but related layers, each of which defines a part of the process of moving informationacross a network (see Figure 2.2). An understanding of the fundamentals of the OSI model provides a solid basis for exploring data communications.

Figure 2.2 *Seven layers of the OSI model*



Layered **Architecture**

The OSI model is composed of seven ordered layers: physical (layer 1), data link (layer 2), network (layer 3), transport (layer 4), session (layer 5), presentation (layer 6), and application (layer 7). Figure 2.3 shows the layers involved when a message is sent from device A to device B. As the message travels from A to B, it may pass through many intermediate nodes. These intermediate nodes usually involve only the first three layers of the OSI model.

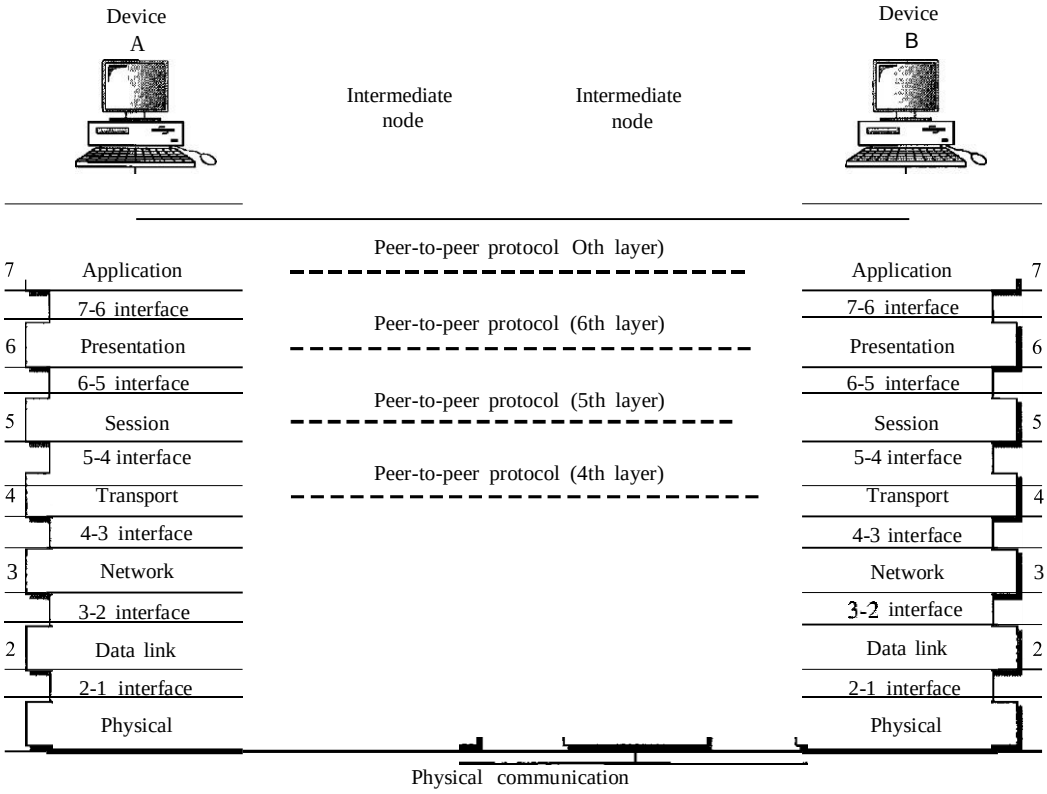
In *developing* the model, the designers distilled the process of transmitting data to its most fundamental elements. They identified which networking functions had related uses and collected those functions into discrete groups that became the layers. Each layer defines a family of functions distinct 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. Layer 3, for example, uses the services provided by layer 2 and provides services for layer 4. 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: In Figure 2.3, device A sends a stream of bits to device B (through intermediate nodes). At the higher layers, however, com• munication must move down through the layers on device A, over to device B, and then

Figure 2.3 *The interaction between layers in the OSI model*



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. For example, layer 2 removes the data meant for it, then passes the rest to layer 3. Layer 3 then removes the data meant for it and passes the rest to layer 4, and so on.

Interfaces Between Layers

The passing of the data and network information down through the layers of the send• ing device and back up through the layers of the receiving device is made possible byan interface between each pair of adjacent layers. Each interface defines the informa• tion 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.

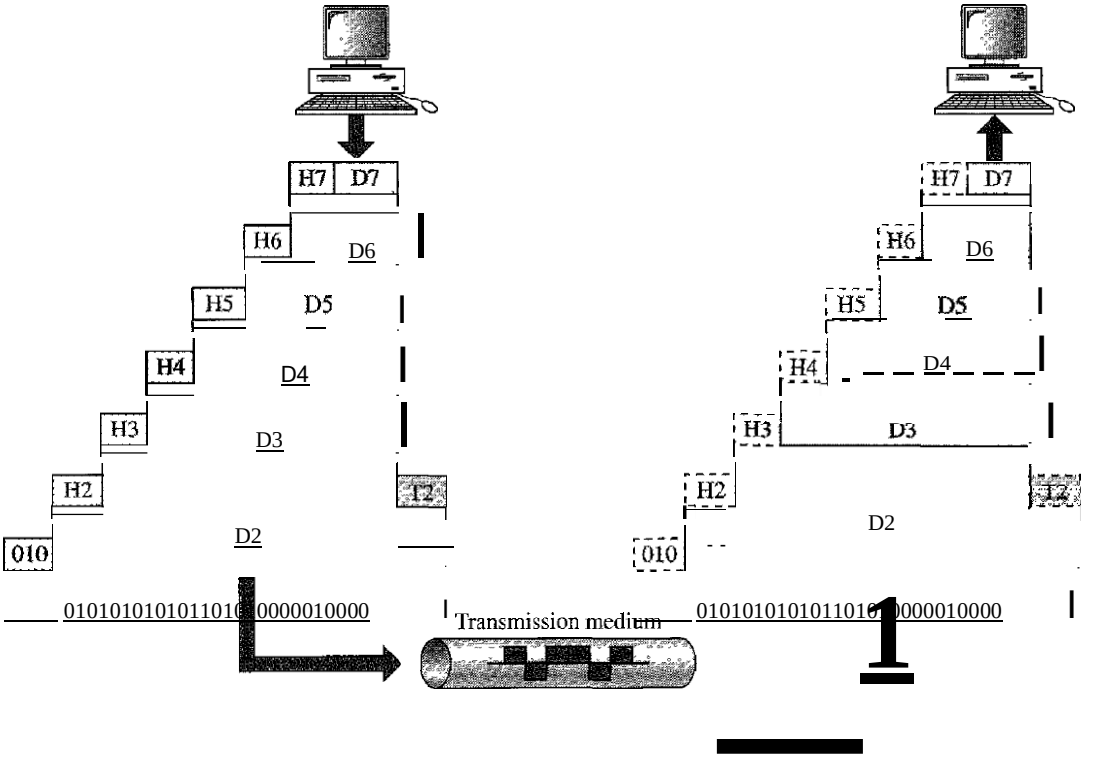
Organization ofthe Layers

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-can 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.

In Figure 2.4, which gives an overall view of the OSI layers, D7 means the data unit at layer 7, D6 means the data unit at layer 6, and so on. The process starts at layer7 (the application layer), then moves from layer to layer in descending, sequential order. 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. When the formatted data unit passes through the physical layer (layer 1), it is changed into an electromagnetic signal and transported along a physical link.

Figure 2.4 *An exchange using the OS! model*



Upon reaching its destination, the signal passes into layer 1 and is transformed back into digital form. The data units then move back up through the OSI layers. As each block of data reaches the next higher layer, the headers and trailers attached to it at the corresponding sending layer are removed, and actions appropriate to that layer are taken. By the time it reaches layer 7, the message is again in a form appropriate to the application and is made available to the recipient.

Encapsulation

Figure 2.3 reveals another aspect of data communications in the OSI model: encapsula• tion. 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.

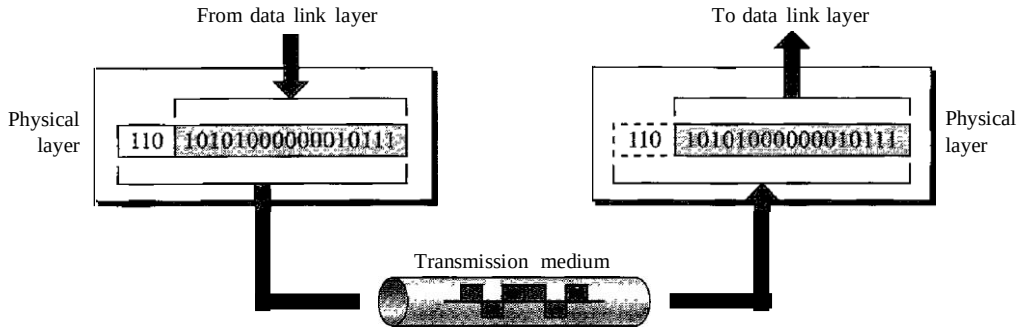
2.3 LAYERS IN THE OSI MODEL

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

Physical Layer

The physical layer coordinates the functions required to carry a bit stream over a physi• cal 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 2.5 shows the position of the physical layer with respect to the transmission medium and the data link layer.

Figure 2.5 *Physical 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:

- Physical characteristics of interfaces and medium. The physical layer defines the characteristics of the interface between the devices and the transmission medium. It also defines the type of transmission medium.
- Representation of bits. The physical layer data consists of a stream of bits (sequence of Os or 1s) with no interpretation. To be transmitted, bits must be

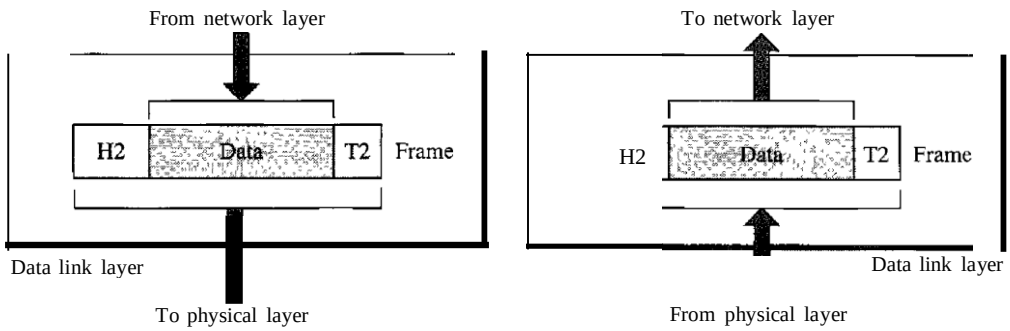
encoded into signals--electrical or optical. The physical layer defines the type of encoding (how Os and Is are changed to signals).

- Data rate. The transmission rate-the number of bits sent each second-is also defined by the physical layer. In other words, the physical layer defines the dura• tion of a bit, which is how long it lasts.
- Synchronization of bits. The sender and receiver not only must use the same bit rate but also must be synchronized at the bit level. In other words, the sender and the receiver clocks must be synchronized.
- Line configuration. The physical layer is concerned with the connection of devices to the media. In a point-to-point configuration, two devices are connected through a dedicated link. In a multipoint configuration, a link is shared among several devices.
- Physical topology. The physical topology defines how devices are connected to make a network. Devices can be connected by using a mesh topology (every device is connected to every other device), a star topology (devices are connected through a central device), a ring topology (each device is connected to the next, forming a ring), a bus topology (every device is on a common link), or a hybrid topology (this is a combination of two or more topologies).
- Transmission mode. The physical layer also defines the direction of transmission between two devices: simplex, half-duplex, or full-duplex. In simplex mode, only one device can send; the other can only receive. The simplex mode is a one-way communication. In the half-duplex mode, two devices can send and receive, but not at the same time. In a full-duplex (or simply duplex) mode, two devices can send and receive at the same time.

Data Link Layer

The data link layer transforms the physical layer, a raw transmission facility, to a reli• able link. It makes the physical layer appear error-free to the upper layer (network layer). Figure 2.6 shows the relationship of the data link layer to the network and phys• icallayers.

Figure 2.6 *Data link layer*



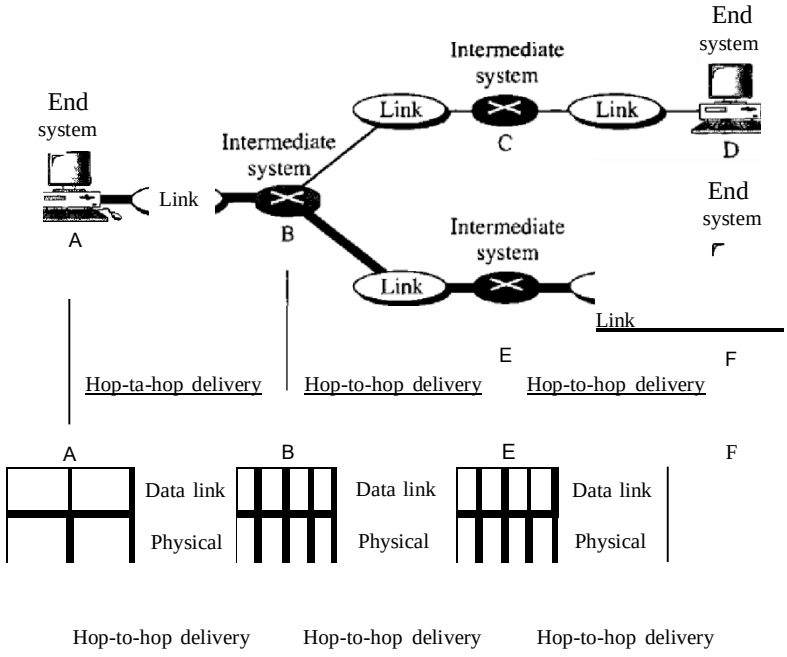
The data link layer is responsible for moving frames from one hop (node) to the next.

Other responsibilities of the data link layer include the following:

- Framing. The data link layer divides the stream of bits received from the network layer into manageable data units called frames.
- Physical addressing. If frames are to be distributed to different systems on the network, the data link layer adds a header to the frame to define the sender and/or receiver of the frame. If the frame is intended for a system outside the sender's network, the receiver address is the address of the device that connects the networkto the next one.
- Flow control. If the rate at which the data are absorbed by the receiver is less than the rate at which data are produced in the sender, the data link layer imposes a flow control mechanism to avoid overwhelming the receiver.
- Error control. The data link layer adds reliability to the physical layer by adding mechanisms to detect and retransmit damaged or lost frames. It also uses a mecha• nism to recognize duplicate frames. Error control is normally achieved through a trailer added to the end of the frame.
- 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.

Figure 2.7 illustrates hop-to-hop (node-to-node) delivery by the data link layer.

Figure 2.7 *Hop-to-hop delivery*



As the figure shows, communication at the data link layer occurs between two adjacent nodes. To send data from A to F, three partial deliveries are made. First, the data link layer at A sends a frame to the data link layer at B (a router). Second, the data

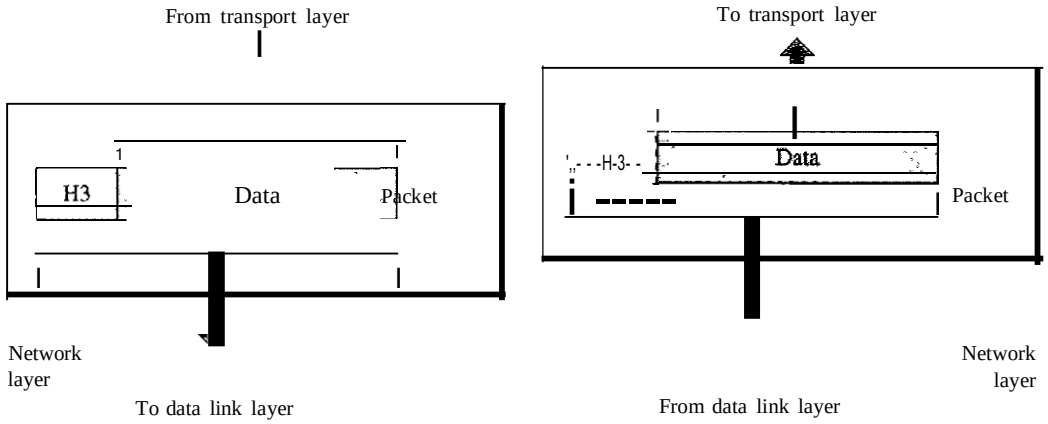
link layer at B sends a new frame to the data link layer at E. Finally, the data link layer at E sends a new frame to the data link layer at F. Note that the frames that are exchanged between the three nodes have different values in the headers. The frame from A to B has B as the destination address and A as the source address. The frame from B to E has E as the destination address and B as the source address. The frame from E to F has F as the destination address and E as the source address. The values of the trailers can also be different if error checking includes the header of the frame.

Network Layer

The network layer is responsible for the source-to-destination delivery of a packet, possibly across multiple networks (links). Whereas the data link layer oversees thedelivery of the packet between two systems on the same network (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 net• work layer. However, if the two systems are attached to different networks (links) with connecting devices between the networks (links), there is often a need for the network layer to accomplish source-to-destination delivery. Figure 2.8 shows the relationship of the network layer to the data link and transport layers.

Figure 2.8 *Network layer*



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:

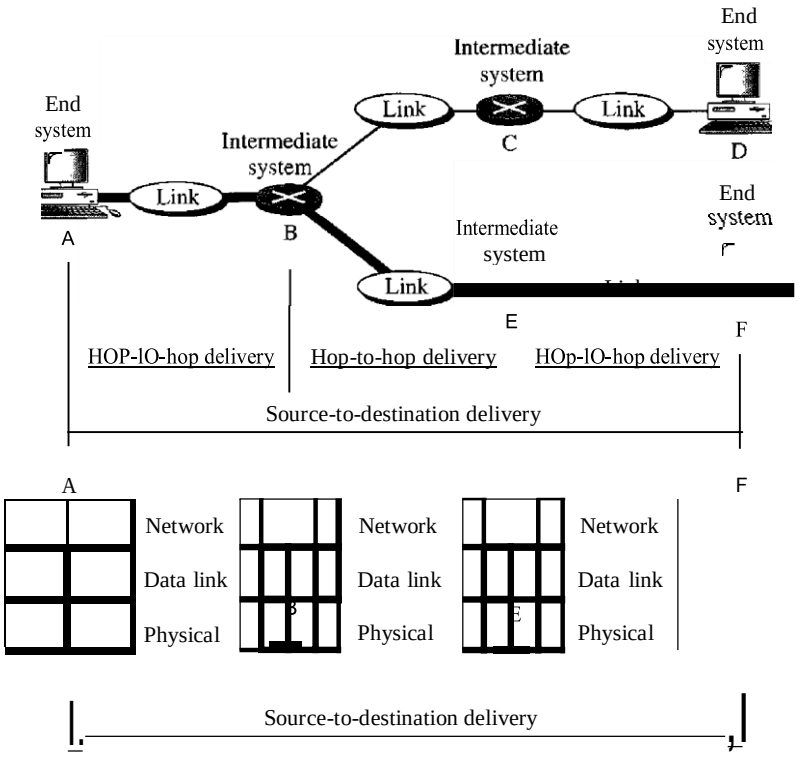
- Logical addressing. The physical addressing implemented by the data link layer handles the addressing problem locally. If a packet passes the network boundary, we need another addressing system to help distinguish the source and destination systems. The network layer adds a header to the packet coming from the upper layer that, among other things, includes the logical addresses of the sender and receiver. We discuss logical addresses later in this chapter.
- Routing. When independent networks or links are connected to create *intemetworks*

(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 func• tions of the network layer is to provide this mechanism.

Figure 2.9 illustrates end-to-end delivery by the network layer.

Figure 2.9 *Source-to-destination delivery*

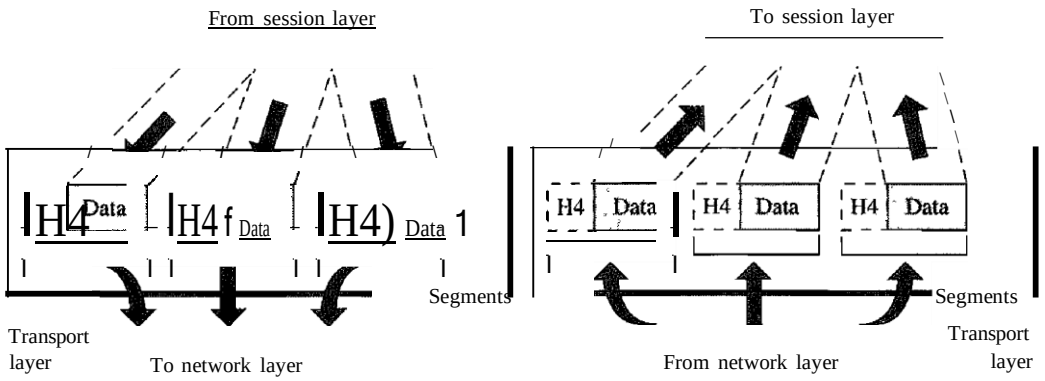


As the figure shows, now we need a source-to-destination delivery. The network layer at A sends the packet to the network layer at B. When the packet arrives at router B, the router makes a decision based on the final destination (F) of the packet. As we will see in later chapters, router B uses its routing table to find that the next hop is router E. The network layer at B, therefore, sends the packet to the network layer at E. The network layer at E, in tum, sends the packet to the network layer at F.

Transport Layer

The transport layer is responsible for process-to-process delivery of the entire mes• sage. 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 2.10 shows the relationship of the transport layer to the network and session layers.

Figure 2.10 *Transport layer*



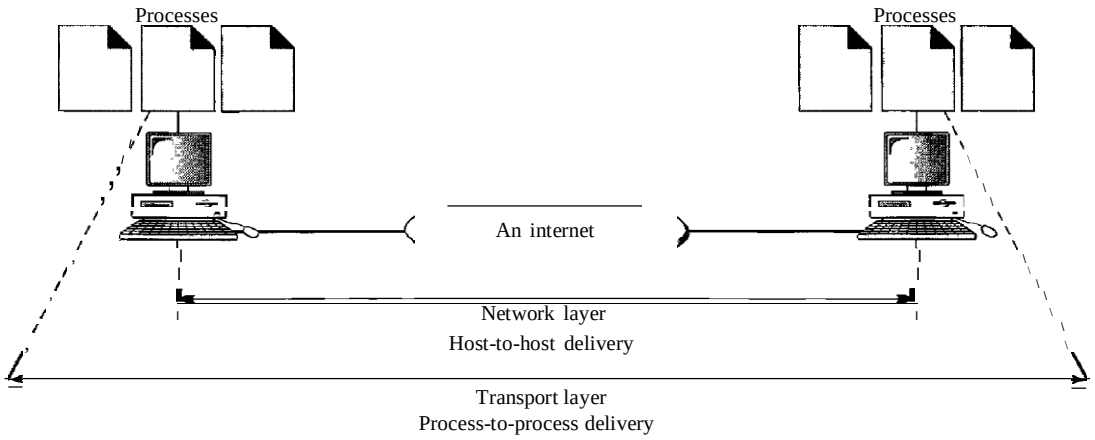
The transport layer is responsible for the delivery of a message from one process to another.

Other responsibilities of the transport layer include the following:

- Service-point addressing. Computers often run several programs at the same time. For this reason, source-to-destination delivery means delivery not only from one computer to the next but also from a specific process (running program) on one computer to a specific process (running program) on the other. The transport layer header must therefore 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.
- 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.
- 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.
- 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.
- 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.

Figure 2.11 illustrates process-to-process delivery by the transport layer.

Figure 2.11 *Reliable process-to-process delivery ofa message*



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 ofthe session layer include the following:

- 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.
- Synchronization. The session layer allows a process to add checkpoints, or syn• Chronization 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. In this case, if a crash happens during the transmission of page 523, the only pages that need to be resent after system recovery are pages 501 to 523. Pages previous to 501 need not be resent. Figure 2.12 illustrates the relationship of the session layer to the transport and presentation layers.

Presentation Layer

The presentation layer is concerned with the syntax and semantics of the information exchanged between two systems. Figure 2.13 shows the relationship between the pre• sentation layer and the application and session layers.

Figure 2.12 *Session layer*

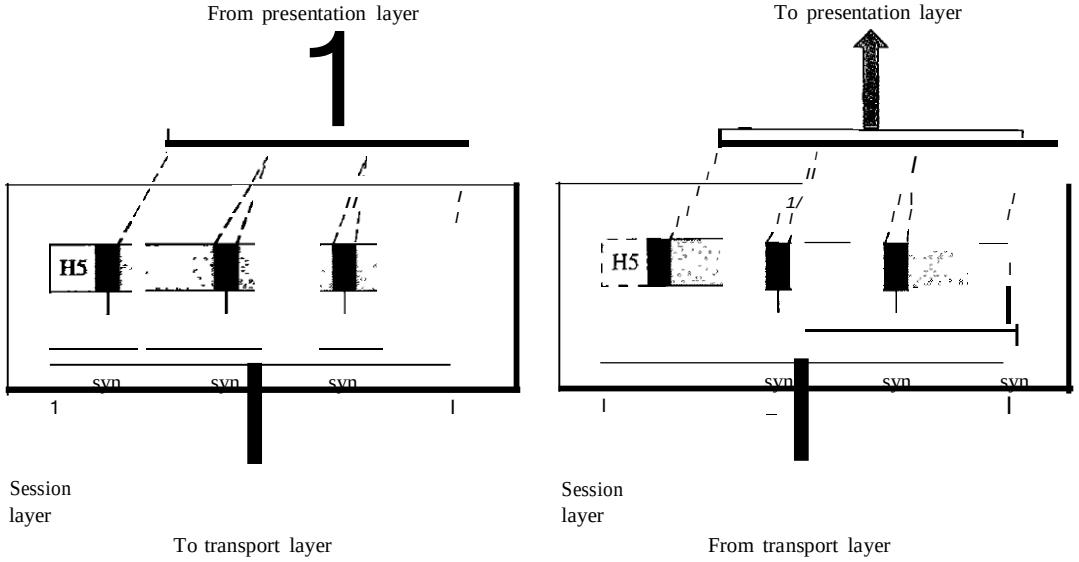
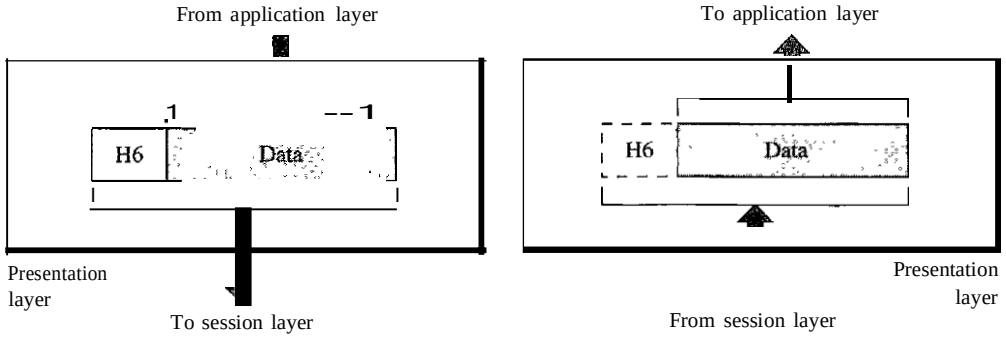


Figure 2.13 *Presentation layer*



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

Specific responsibilities of the presentation layer include the following:

- 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.
- 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.

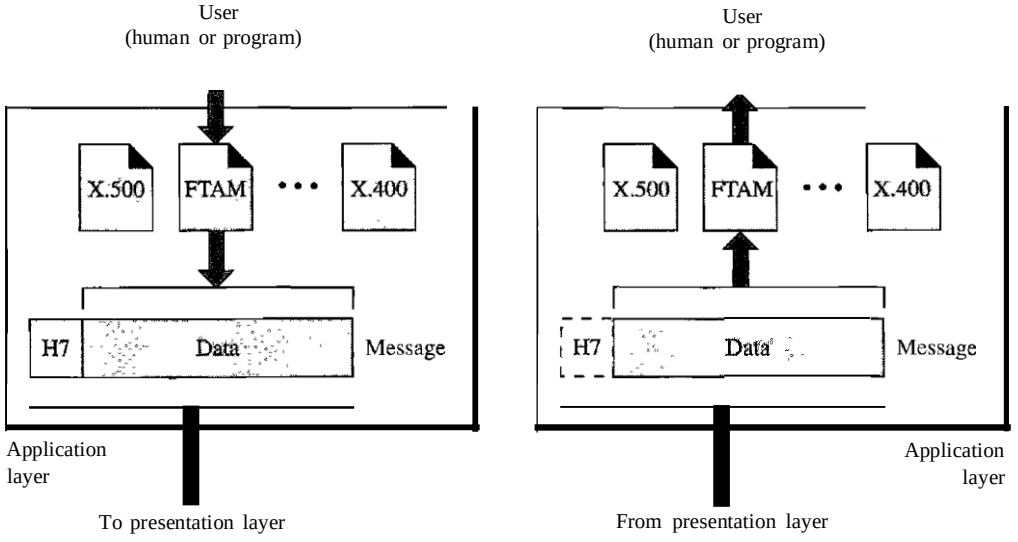
- Compression. Data compression reduces the number of bits contained in the information. Data compression becomes particularly important in the transmission of multimedia such as text, audio, and video.

Application Layer

The application layer enables the user, whether human or software, to access the net• work. It provides user interfaces and support for services such as electronic mail, remote file access and transfer, shared database management, and other types of distrib• uted information services.

Figure 2.14 shows the relationship of the application layer to the user and the pre• sentation layer. Of the many application services available, the figure shows only three: *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.

Figure 2.14 *Application layer*



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

Specific services provided by the application layer include the following:

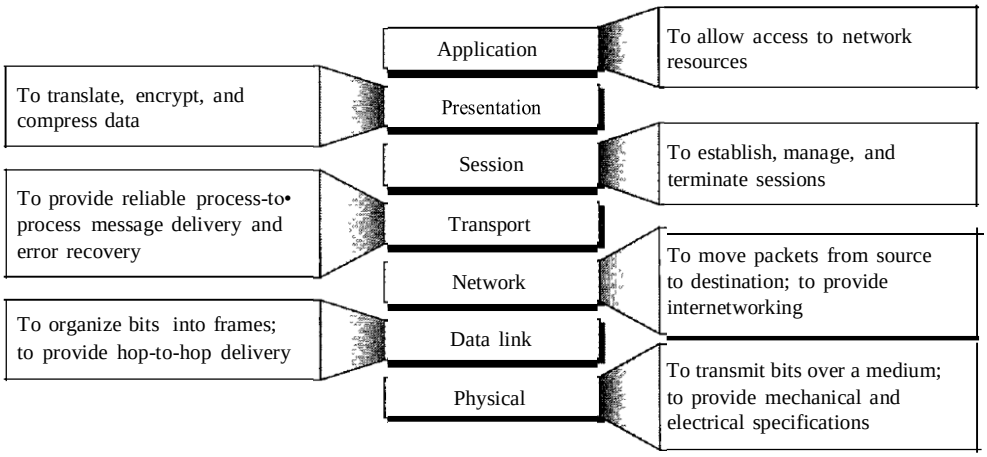
- Network virtual terminal. A network virtual terminal is a software version of a physical terminal, and it allows a user to log on to a remote host. To do so, the application creates a software emulation of a terminal at the remote host. The user's computer talks to the software terminal which, in turn, 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.

- 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.
- Mail services. This application provides the basis for e-mail forwarding and storage.
- Directory services. This application provides distributed database sources and access for global information about various objects and services.

Summary of Layers

Figure 2.15 shows a summary of duties for each layer.

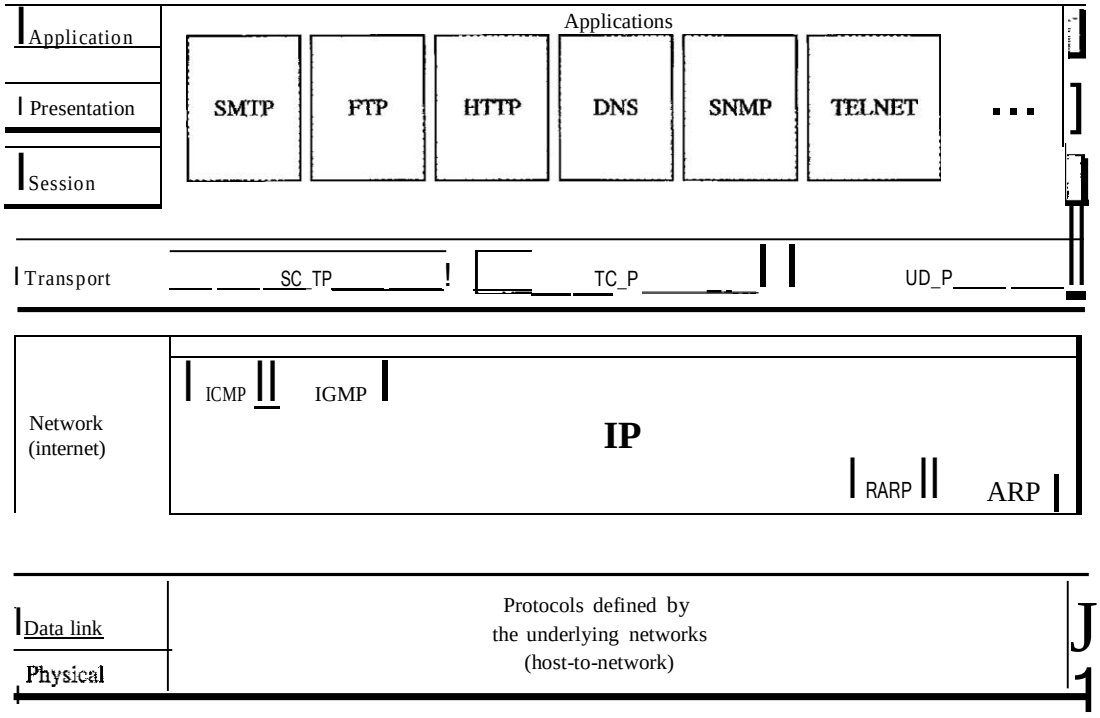
Figure 2.15 *Summary oflayers*



2.4 TCP/IP PROTOCOL SUITE

The TCPIIP protocol suite was developed prior to the OSI model. Therefore, the lay• ers 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: host-to-network, internet, transport, and application. However, when TCP/IP is compared to OSI, we can say that the host-to-network layer is equivalent to the combination of the physical and data link layers. The internet layer is equivalent to the network layer, and the applica• tion layer is roughly doing the job of the session, presentation, and application layers with the transport layer in TCPIIP taking care of part of the duties of the session layer. So in this book, we assume that the TCPIIP protocol suite is made of five layers: physi• cal, 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 TCPIIP by a single layer called the *application layer* (see Figure 2.16).

Figure 2.16 *TCPIIP and OSI model*



TCP/IP is a hierarchical protocol made up of interactive modules, each of which provides a specific functionality; however, the modules are not necessarily interdepen• dent. 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 *hierarchi• cal* means that each upper-level protocol is supported by one or more lower-level protocols.

At the transport layer, *TCP/IP* defines three protocols: Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and Stream Control Transmission Protocol (SCTP). At the network layer, the main protocol defined by TCP/IP is the Internetworking Protocol (IP); there are also some other protocols that support data movement in this layer.

Physical and Data Link Layers

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

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 chapters.

Internetworking Protocol (IP)

The 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 throughto its destination, but with no guarantees.

IP transports data in packets called *datagrams,* each of which is transported sepa• rately. 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 data• grams 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. IP is dis• cussed in Chapter 20.

Address Resolution Protocol

The 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 linkis identified by a physical or station address, usually imprinted on the network interfacecard (NIC). ARP is used to find the physical address of the node when its Internet address is known. ARP is discussed in Chapter 21.

Reverse Address Resolution Protocol

The Reverse Address Resolution Protocol (RARP) allows a host to discover its Inter• net address when it knows only its physical address. It is used when a computer is con• nected to a network for the first time or when a diskless computer is booted. We discuss RARP in Chapter 21.

Internet Control Message Protocol

The 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. We discuss ICMP in Chapter 21.

Internet Group Message Protocol

The Internet Group Message Protocol (IGMP) is used to facilitate the simultaneous transmission of a message to a group of recipients. We discuss IGMP in Chapter 22.

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.

User Datagram Protocol

The User Datagram Protocol (UDP) is the simpler of the two standard TCPIIP 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. UDP is discussed in Chapter 23.

Transmission Control Protocol

The Transmission Control Protocol (TCP) provides full transport-layer services to applications. TCP is a reliable stream transport protocol. The term *stream,* in this con• text, 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 col• lects each datagram as it comes in and reorders the transmission based on sequence numbers. TCP is discussed in Chapter 23.

Stream Control Transmission Protocol

The Stream Control Transmission Protocol (SCTP) provides support for newer applications such as voice over the Internet. It is a transport layer protocol that com• bines the best features of UDP and TCP. We discuss SCTP in Chapter 23.

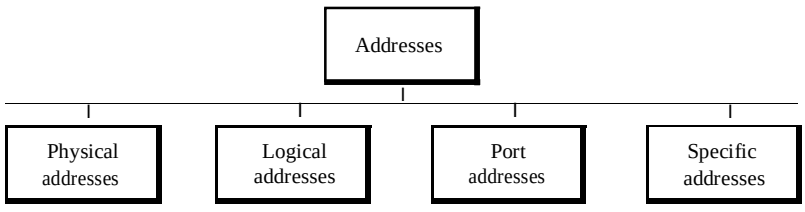
Application Layer

The *application layer* in TCPIIP is equivalent to the combined session, presentation, and application layers in the OSI modeL Many protocols are defined at this layer. We cover many of the standard protocols in later chapters.

2.5 ADDRESSING

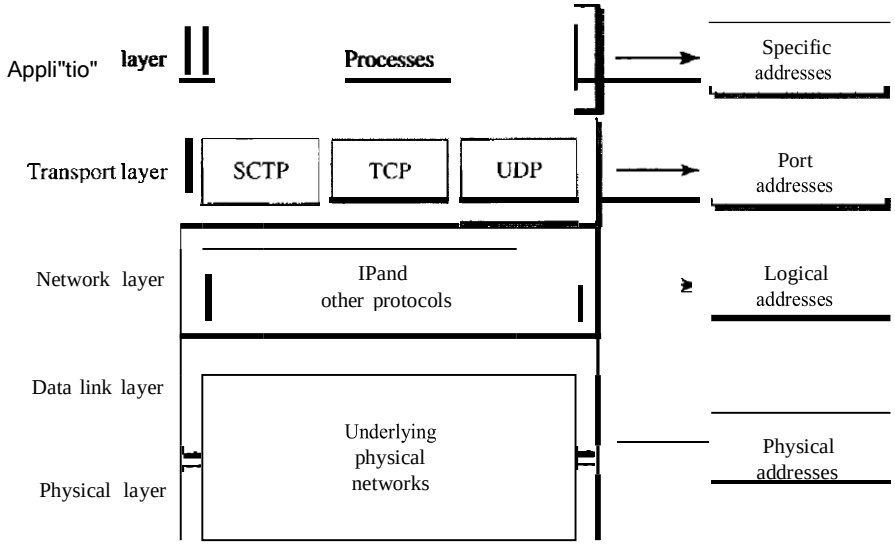
Four levels of addresses are used in an internet employing the *TCP/IP* protocols: physical (link) addresses, logical (IP) addresses, port addresses, and specific addresses (see Figure 2.17).

Figure 2.17 *Addresses in TCPIIP*



Each address is related to a specific layer in the TCPIIP architecture, as shown in Figure 2.18.

Figure 2.18 *Relationship oflayers and addresses in TCPIIP*



Physical Addresses

The physical address, also known as the link address, 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.

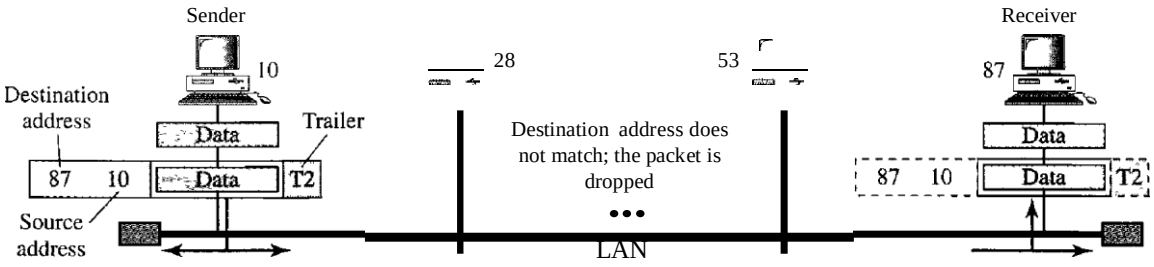
The physical addresses have authority over the network (LAN or WAN). The size and format of these addresses vary depending on the network. For example, Ethernet uses a 6-byte (48-bit) physical address that is imprinted on the network interface card (NIC). LocalTalk (Apple), however, has a I-byte dynamic address that changes each time the station comes up.

Example 2.1

In Figure 2.19 a node with physical address 10 sends a frame to a node with physical address 87. The two nodes are connected by a link (bus topology LAN). At the data link layer, this frame contains physical (link) addresses in the header. These are the only addresses needed. The rest of the header contains other information needed at this level. The trailer usually contains extra bits needed for error detection. As the figure shows, the computer with physical address lOis the sender, and the computer with physical address 87 is the receiver. The data link layer at the sender receives data from an upper layer. It encapsulates the data in a frame, adding a header and a trailer. The header, among other pieces of information, carries the receiver and the sender phys•ical (link) addresses. Note that in most data link protocols, the destination address, 87 in this case, comes before the source address (10 in this case).

We have shown a bus topology for an isolated LAN. In a bus topology, the frame is propa• gated in both directions (left and right). The frame propagated to the left dies when it reaches the end of the cable if the cable end is terminated appropriately. The frame propagated to the right is

Figure 2.19 *Physical addresses*



sent to every station on the network. Each station with a physical addresses other than 87 drops the frame because the destination address in the frame does not match its own physical address. The intended destination computer, however, finds a match between the destination address in the frame and its own physical address. The frame is checked, the header and trailer are dropped, and the data part is decapsulated and delivered to the upper layer.

Example 2.2

As we will see in Chapter 13, most local-area networks use a 48-bit (6-byte) physical address written as 12 hexadecimal digits; every byte (2 hexadecimal digits) is separated by a colon, as shown below:

07:01:02:01 :2C:4B
A 6-byte (12 hexadecimal digits) physical address

Logical Addresses

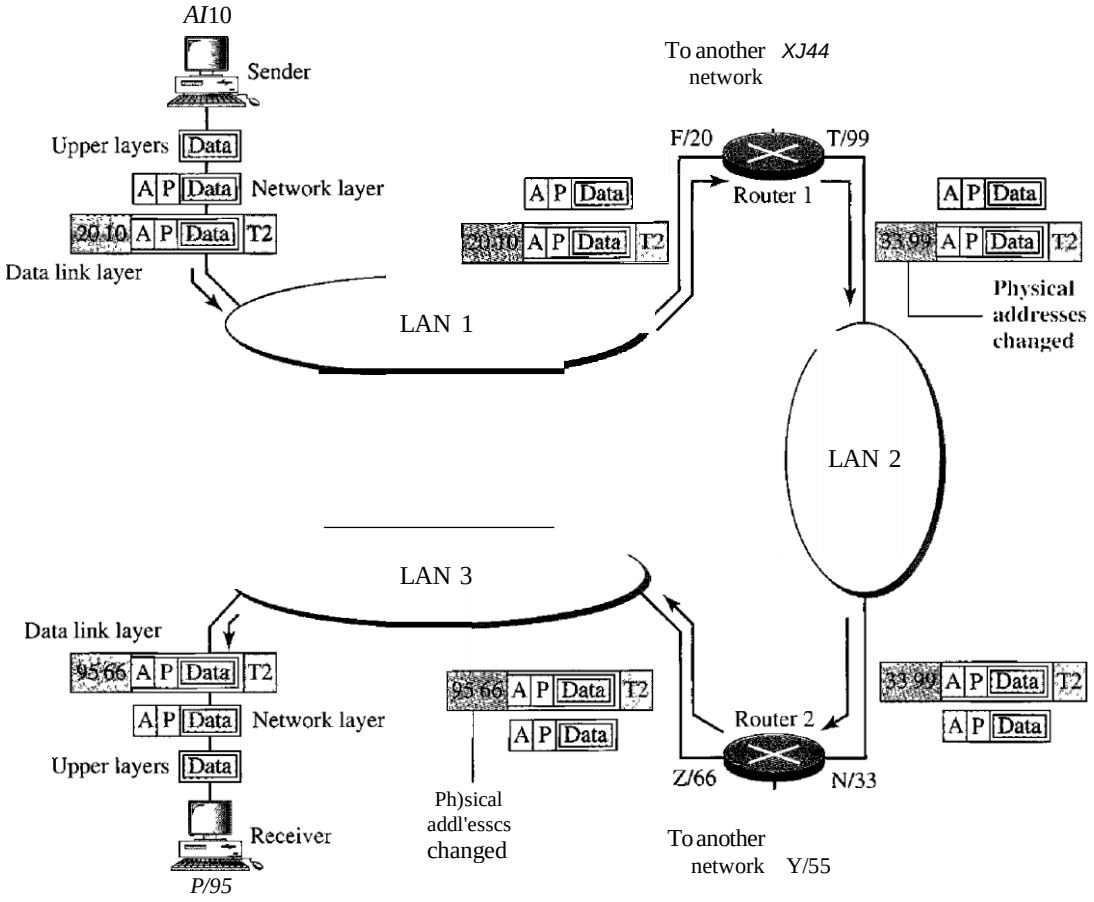
Logical addresses are necessary for universal communications that are independent of underlying physical networks. Physical addresses are not adequate in an internetwork environment where different networks can have different address formats. A universal addressing system is needed in which each host can be identified uniquely, regardless of the underlying physical network.

The logical addresses are designed for this purpose. A logical address in the Internet is currently a 32-bit address that can uniquely define a host connected to the Internet. No two publicly addressed and visible hosts on the Internet can have the same IP address.

Example 2.3

Figure 2.20 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. Although it may obvious that each router must have a separate physical address for each connection, it may not be obvious why it needs a logical address for each connection. We discuss these issues in Chapter 22 when we dis• cuss routing.

Figure 2.20 *IP addresses*



The computer with logical address A and physical address 10 needs to send a packet to the computer with logical address P and physical address 95. We use letters to show the logical addresses and numbers for physical addresses, but note that both are actually numbers, as we will see later in the chapter.

The sender encapsulates its data in a packet at the network layer and adds two logical addresses (A and P). Note that in most protocols, the logical source address comes before the logical destination address (contrary to the order of physical addresses). The network layer, however, needs to find the physical address of the next hop before the packet can be delivered. The network layer consults its routing table (see Chapter 22) and finds the logical address of the next hop (router I) to be F. The ARP discussed previously finds the physical address of router 1 that corresponds to the logical address of 20. Now the network layer passes this address to the data link layer, which in tum, encapsulates the packet with physical destination address 20 and physical source address 10.

The frame is received by every device on LAN 1, but is discarded by all except router 1, which finds that the destination physical address in the frame matches with its own physical address. The router decapsulates the packet from the frame to read the log• ical destination address P. Since the logical destination address does not match the router's logical address, the router knows that the packet needs to be forwarded. The

router consults its routing table and ARP to find the physical destination address of the next hop (router 2), creates a new frame, encapsulates the packet, and sends it to router 2.

Note the physical addresses in the frame. The source physical address changes from 10 to 99. The destination physical address changes from 20 (router 1 physical address) to 33 (router 2 physical address). The logical source and destination addresses must remain the same; otherwise the packet will be lost.

At router 2 we have a similar scenario. The physical addresses are changed, and a new frame is sent to the destination computer. When the frame reaches the destination, the packet is decapsulated. The destination logical address P matches the logical address of the computer. The data are decapsulated from the packet and delivered to the upper layer. Note that although physical addresses will change from hop to hop, logical addresses remain the same from the source to destination. There are some exceptions to this rule that we discover later in the book.

The physical addresses will change from hop to hop, but the logical addresses usually remain the same.

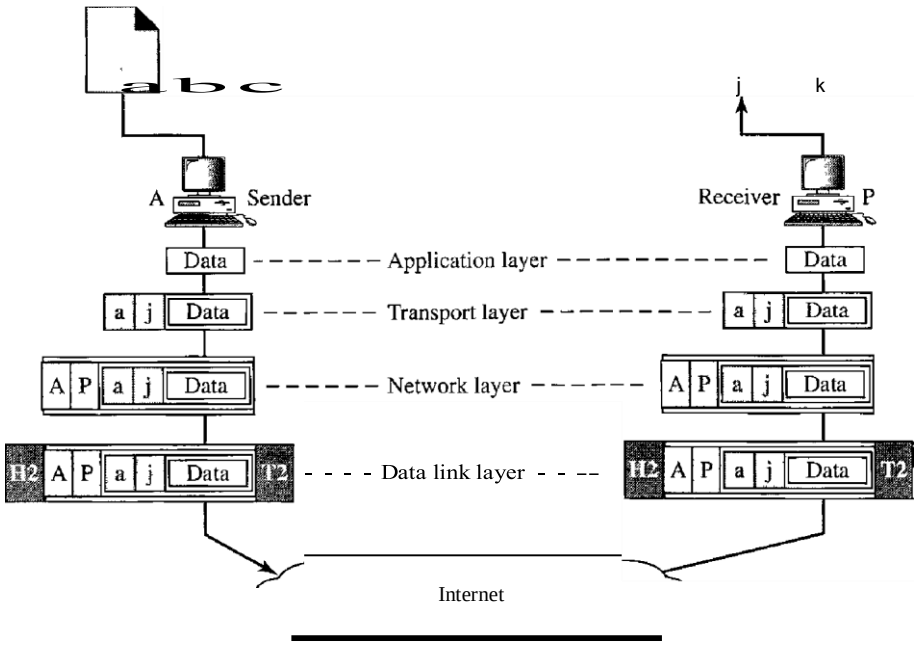
Port Addresses

The IP address and the physical address are necessary for a quantity of data to travel from a source to the destination host. However, arrival at the destination host is not the final objective of data communications on the Internet. A system that sends nothing but data from one computer to another is not complete. Today, computers are devices that can run multiple processes at the same time. The end objective of Internet communica• tion is a process communicating with another process. For example, computer A can communicate with computer C by using TELNET. At the same time, computer A com• municates with computer B by using the File Transfer Protocol (FTP). For these pro• cesses to receive data simultaneously, we need a method to label the different processes. In other words, they need addresses. In the TCPIIP architecture, the label assigned to a process is called a port address. A port address in TCPIIP is 16 bits in length.

Example 2.4

Figure 2.21 shows two computers communicating via the Internet. The sending computer is run• ning three processes at this time with port addresses a, b, and c. The receiving computer is running two processes at this time with port addresses j and k. Process a in the sending computer needs to communicate with process j in the receiving computer. Note that although both computers are using the same application, FTP, for example, the port addresses are different because one is a client program and the other is a server program, as we will see in Chapter 23. To show that data from process a need to be delivered to process j, and not k, the transport layer encapsulates data fromthe application layer in a packet and adds two port addresses (a and j), source and destination. The packet from the transport layer is then encapsulated in another packet at the network layer with logical source and destination addresses (A and P). Finally, this packet is encapsulated in a frame with the physical source and destination addresses of the next hop. We have not shown the physi•cal addresses because they change from hop to hop inside the cloud designated as the Internet. Note that although physical addresses change from hop to hop, logical and port addresses remain the same from the source to destination. There are some exceptions to this rule that we discuss later in the book.

Figure 2.21 *Port addresses*



The physical addresses change from hop to hop, but the logical and port addresses usually remain the same.

Example 2.5

As we will see in Chapter 23, a port address is a 16-bit address represented by one decimal number as shown.

753

A 16-bit port address represented as one single number

Specific Addresses

Some applications have user-friendly addresses that are designed for that specific address. Examples include the e-mail address (for example, forouzan@fhda.edu) and the Universal Resource Locator (URL) (for example, www.mhhe.com). The first defines the recipient of an e-mail (see Chapter 26); the second is used to find a document on the World Wide Web (see Chapter 27). These addresses, however, get changed to the corresponding port and logical addresses by the sending computer, as we will see in Chapter 25.

2.6 SUMMARY

- The International Standards Organization created a model called the Open Systems Interconnection, which allows diverse systems to communicate.
- The seven-layer OSI model provides guidelines for the development of universally compatible networking protocols.
- The physical, data link, and network layers are the network support layers.
- The session, presentation, and application layers are the user support layers.
- The transport layer links the network support layers and the user support layers.
- The physical layer coordinates the functions required to transmit a bit stream over a physical medium.
- The data link layer is responsible for delivering data units from one station to the next without errors.
- The network layer is responsible for the source-to-destination delivery of a packet across multiple network links.
- The transport layer is responsible for the process-to-process delivery of the entire message.
- The session layer establishes, maintains, and synchronizes the interactions between communicating devices.
- The presentation layer ensures interoperability between communicating devices through transformation of data into a mutually agreed upon format.
- The application layer enables the users to access the network.
- TCP/IP is a five-layer hierarchical protocol suite developed before the OSI model.
- The TCP/IP application layer is equivalent to the combined session, presentation, and application layers of the OSI model.
- Four levels of addresses are used in an internet following the TCP/IP protocols: physical (link) addresses, logical (IP) addresses, port addresses, and specific addresses.
- The physical address, also known as the link address, is the address of a node as defined by its LAN or WAN.
- The IP address uniquely defines a host on the Internet.
- The port address identifies a process on a host.
- A specific address is a user-friendly address.

2.7 PRACTICE SET

Review Questions

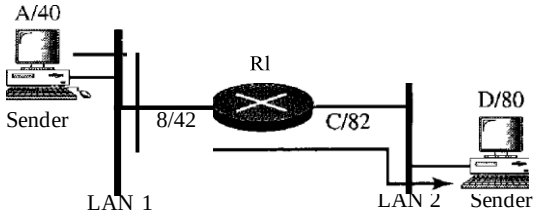
1. List the layers of the Internet model.
2. Which layers in the Internet model are the network support layers?
3. Which layer in the Internet model is the user support layer?
4. What is the difference between network layer delivery and transport layer delivery?

5. What is a peer-to-peer process?
6. How does information get passed from one layer to the next in the Internet model?
7. What are headers and trailers, and how do they get added and removed?
8. What are the concerns of the physical layer in the Internet model?
9. What are the responsibilities of the data link layer in the Internet model?
10. What are the responsibilities of the network layer in the Internet model?
11. What are the responsibilities of the transport layer in the Internet model?
12. What is the difference between a port address, a logical address, and a physical address?
13. Name some services provided by the application layer in the Internet model.
14. How do the layers of the Internet model correlate to the layers of the OSI model?

Exercises

15. How are OSI and ISO related to each other?
16. Match the following to one or more layers of the OSI model:
    a. Route determination
    b. Flow control
    c. Interface to transmission media
    d. Provides access for the end user
17. Match the following to one or more layers of the OSI model:
    a. Reliable process-to-process message delivery
    b. Route selection
    c. Defines frames
    d. Provides user services such as e-mail and file transfer
    e. Transmission of bit stream across physical medium
18. Match the following to one or more layers of the OSl model:
    a. Communicates directly with user's application program
    b. Error correction and retransmission
    c. Mechanical, electrical, and functional interface
    d. Responsibility for carrying frames between adjacent nodes
19. Match the following to one or more layers of the OSI model:
    a. Format and code conversion services
    b. Establishes, manages, and terminates sessions
    c. Ensures reliable transmission of data
    d. Log-in and log-out procedures
    e. Provides independence from differences in data representation
20. In Figure 2.22, computer A sends a message to computer D via LANl, router Rl, and LAN2. Show the contents of the packets and frames at the network and data link layer for each hop interface.

Figure 2.22 *Exercise 20*



21. In Figure 2.22, assume that the communication is between a process running at computer A with port address *i* and a process running at computer D with port address *j*. Show the contents of packets and frames at the network, data link, and transport layer for each hop.
22. Suppose a computer sends a frame to another computer on a bus topology LAN. The physical destination address of the frame is corrupted during the transmission. What happens to the frame? How can the sender be informed about the situation?
23. Suppose a computer sends a packet at the network layer to another computer somewhere in the Internet. The logical destination address of the packet is cor• rupted. What happens to the packet? How can the source computer be informed of the situation?
24. Suppose a computer sends a packet at the transport layer to another computer somewhere in the Internet. There is no process with the destination port address running at the destination computer. What will happen?
25. If the data link layer can detect errors between hops, why do you think we need another checking mechanism at the transport layer?

Research Activities

26. Give some advantages and disadvantages of combining the session, presentation, and application layer in the OSI model into one single application layer in the Internet model.
27. Dialog control and synchronization are two responsibilities of the session layer in the OSI model. Which layer do you think is responsible for these duties in the Internet model? Explain your answer.
28. Translation, encryption, and compression are some of the duties of the presentation layer in the OSI model. Which layer do you think is responsible for these duties in the Internet model? Explain your answer.
29. There are several transport layer models proposed in the OSI model. Find all of them. Explain the differences between them.
30. There are several network layer models proposed in the OSI model. Find all of them. Explain the differences between them.

PART

2

Physical Layer and Media

Objectives

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 Os and I s orga• nized into frames that are ready to be sent across the transmission medium. This stream of Os and I s must first be converted into another entity: signals. One of the services pro• vided 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.

In Part 2 of the book, we discuss issues related to the physical layer and the trans• mission medium that is controlled by the physical layer. In the last chapter of Part 2, we discuss the structure and the physical layers of the telephone network and the cable network.

Part 2 of the book is devoted to the physical layer

CHAPTER 3

Data and Signals

One of the major functions of the physical layer is to move data in the form of electro• magnetic 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 center, you are working with the transmis• sion 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 contin• uous; **digital data** refers to information that has discrete states. For example, an analog clock that has hour, minute, and second hands gives information in a continuous form; the movements of the hands are continuous. On the other hand, a digital clock that reports the hours and the minutes will change suddenly from 8:05 to 8:06.

Analog data, such as the sounds made by a human voice, take on continuous values. When someone speaks, an analog wave is created in the air. This can be captured by a microphone and converted to an analog signal or sampled and converted to a digital signal.

Digital data take on discrete values. For example, data are stored in computer memory in the form of Os and 1s. They can be converted to a digital signal or modu• lated into an analog signal for transmission across a medium.

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

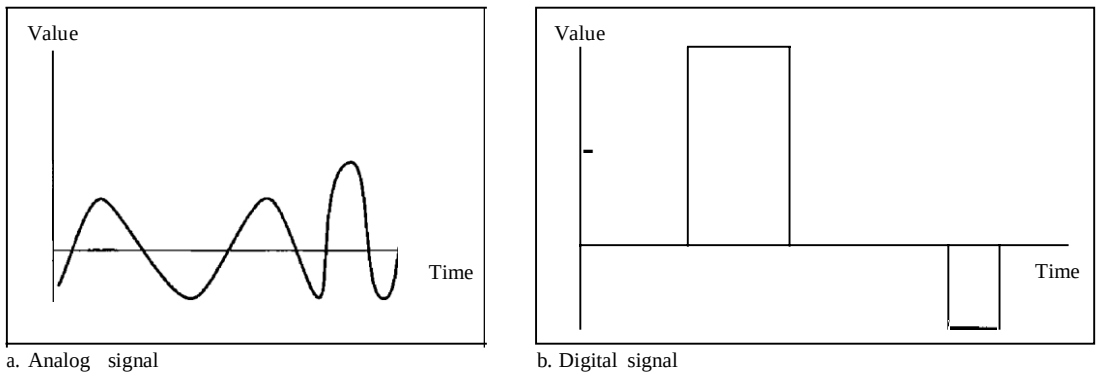
Analog and Digital Signals

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. As the wave moves from value *A* to value *B,* it passes through and includes an infinite number of values along its path. 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 O.

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. Figure 3.1 illustrates an analog signal and a digital signal. The curve representing the analog signal passes through an infinite number of points. The vertical lines of the digital signal, however, demonstrate the sudden jump that the signal makes from value to value.

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.

Figure 3.1 *Comparison ofanalog and digital signals*



a. Analog signal

b. Digital signal

Periodic and Nonperiodic Signals

Both analog and digital signals can take one of two forms: *periodic* or *nonperiodic* (sometimes refer to as *aperiodic*, because the prefix *a* in Greek means "non").

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 nonperiodic signal changes without exhibiting a pat• tern or cycle that repeats over time.

Both analog and digital signals can be periodic or nonperiodic. In data communi• cations, we commonly use periodic analog signals (because they need less bandwidth,

as we will see in Chapter 5) and nonperiodic digital signals (because they can represent variation in data, as we will see in Chapter 6).

In data communications, we commonly use periodic analog signals and nonperiodic digital signals.

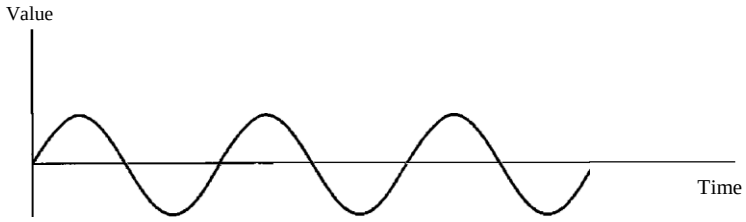
3.2 **PERIODIC** ANALOG SIGNALS

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 3.2 shows a sine wave. Each cycle consists of a single arc above the time axis followed by a single arc below it.

Figure 3.2 *A sine wave*



We discuss a mathematical approach to sine waves in Appendix C.

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

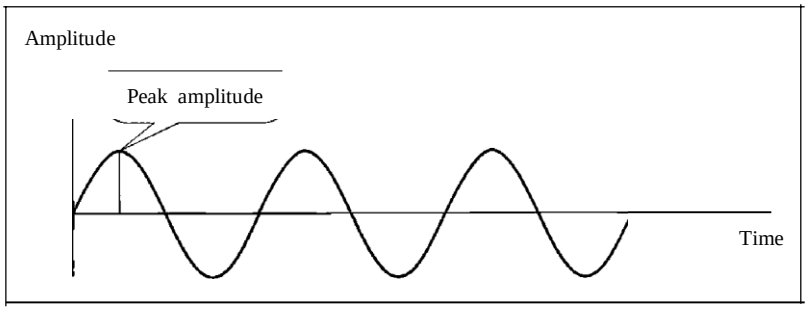
Peak Amplitude

The peak amplitude of a signal is the absolute value of its highest intensity, propor• tional to the energy it carries. For electric signals, peak amplitude is normally measured in *volts.* Figure 3.3 shows two signals and their peak amplitudes.

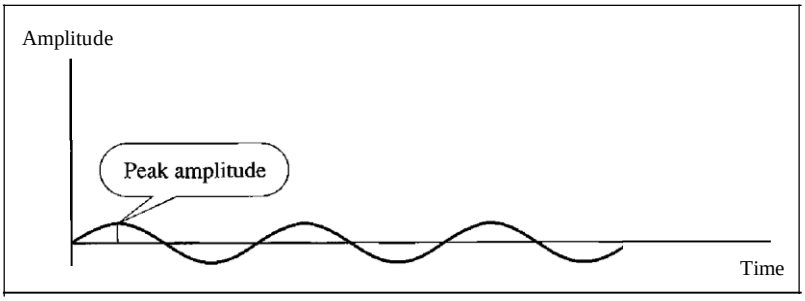
Example 3.1

The power in your house can be represented by a sine wave with a peak amplitude of 155 to 170 V. However, it is common knowledge that the voltage of the power in U.S. homes is 110 to 120 V.

Figure 3.3 *Two signals with the same phase andfrequency, but different amplitudes*



a. A signal with high peak amplitude



b. A signal with low peak amplitude

This discrepancy is due to the fact that these are root mean square (rms) values. The signal is squared and then the average amplitude is calculated. The peak value is equal to 2^{112} x rms value.

Example 3.2

The voltage of battery is a constant; this constant value can be considered a sine wave, as we will see later. For example, the peak value of an AA battery is normally 1.5 V.

Period and Frequency

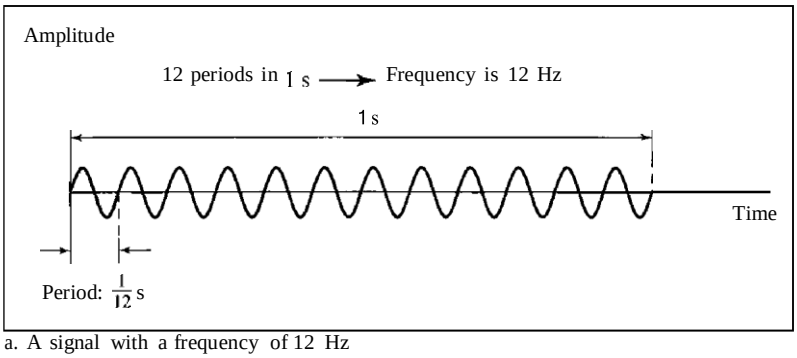
Period refers to the amount of time, in seconds, a signal needs to complete 1 cycle. Frequency refers to the number of periods in I s. Note that period and frequency are just one characteristic defined in two ways. Period is the inverse of frequency, and frequency is the inverse of period, as the following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

$$f = \frac{1}{T} \quad \text{and} \quad T = \frac{1}{f}$$

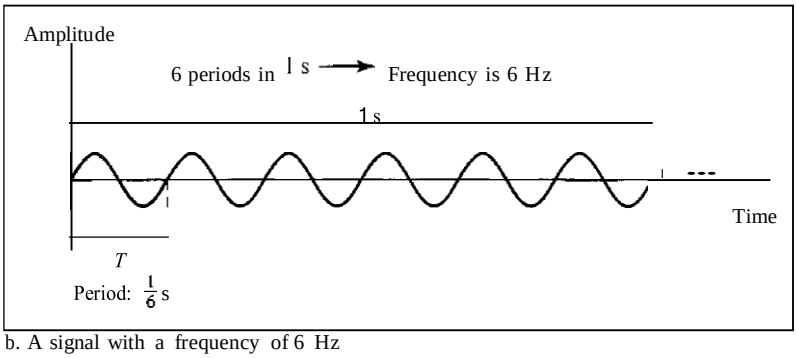
Frequency and period are the inverse of each other.

Figure 3.4 shows two signals and their frequencies.

Figure 3.4 *Two signals with the same amplitude and phase, but different frequencies*



a. A signal with a frequency of 12 Hz



b. A signal with a frequency of 6 Hz

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.

Table 3.1 *Units ofperiod andfrequency*

Unit	*Equivalent*	*Unit*	*Equivalent*
Seconds (s)	1 s	Hertz (Hz)	1 Hz
Milliseconds (ms)	10^{-3} s	Kilohertz (kHz)	10^{3} Hz
Microseconds (μs)	10-6 s	Megahertz (MHz)	10^{6} Hz
Nanoseconds (ns)	10^{-9} s	Gigahertz (GHz)	10^{9} Hz
Picoseconds (ps)	10^{-12} s	Terahertz (THz)	10^{12} Hz

Example 3.3

The power we use at home has a frequency of 60 Hz (50 Hz in Europe). The period of this sine wave can be determined as follows:

$$T = \frac{1}{f} = \frac{1}{60} = 0.0166 \text{ s} = 0.0166 \times 10^{3} \text{ ms} = 16.6 \text{ ms}$$

This means that the period of the power for our lights at home is 0.0116 s, or 16.6 ms. Our eyes are not sensitive enough to distinguish these rapid changes in amplitude.

Example 3.4

Express a period of 100 ms in microseconds.

Solution

From Table 3.1 we find the equivalents of 1 ms (1 ms is 10^{-3} s) and 1 s (1 sis 10^{6} μs). We make the following substitutions:

$$100 \text{ ms} = 100 \times 10^{-3} \text{ s} = 100 \times 10^{-3} \times 10^{6} \text{ μs} = 10^{2} \times 10^{-3} \times 10^{6} \text{ μs} = 10^{5} \text{ μs}$$

Example 3.5

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

Solution

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

$$100 \text{ ms} = 100 \times 10^{-3} \text{ s} = 10^{-1} \text{ s}$$

$$f = \frac{1}{T} = \frac{1}{10^{-1}} \text{ Hz} = 10 \text{ Hz} = 10 \times 10^{-3} \text{ kHz} = 10^{-2} \text{ kHz}$$

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 wave• forms; 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 sec• ond (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 sim• ple one. Ifa signal does not change at all, it never completes a cycle, so its frequency is aHz.

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.

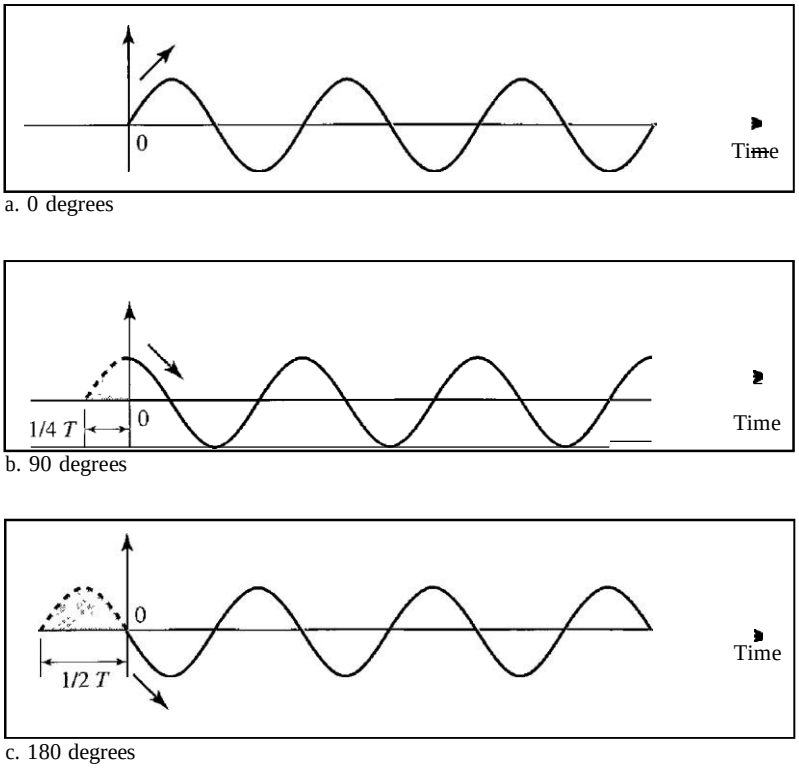
Phase

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° cor• responds to a shift of one-quarter of a period (see Figure 3.5).

Figure 3.5 *Three sine waves with the same amplitude andfrequency, but different phases*



a. 0 degrees

b. 90 degrees

c. 180 degrees

Looking at Figure 3.5, 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 $\frac{1}{4}$ cycle. However, note that the signal does not really exist before time O.
3. A sine wave with a phase of 180° is shifted to the left by $\frac{1}{2}$ cycle. However, note that the signal does not really exist before time O.

Example 3.6

A sine wave is offset $\frac{1}{6}$ cycle with respect to time O. What is its phase in degrees and radians?

Solution

We know that 1 complete cycle is 360°. Therefore, $\frac{1}{6}$ cycle is

$$\frac{1}{6} \times 360 = 60° \times \frac{2\pi}{360} \text{ rad} = \frac{\pi}{3} \text{ rad} = 1.046 \text{ rad}$$

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 (see Figure 3.6).

Figure 3.6 *Wavelength and period*

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 λ, propagation speed by c (speed of light), and frequency by *f*, we get

$$\text{Wavelength} = \text{propagation speed} \times \text{period} = \frac{\text{propagation speed}}{\text{frequency}}$$

The propagation speed of electromagnetic signals depends on the medium and on the frequency of the signal. For example, in a vacuum, light is propagated with a speed of 3 x 10^8 mls. That speed is lower in air and even lower in cable.

The wavelength is normally measured in micrometers (microns) instead of meters. For example, the wavelength of red light (frequency = 4 x 10^{14}) in air is

$$\lambda = \frac{c}{f} = \frac{3 \times 10^8}{4 \times 10^{14}} = 0.75 \times 10^{-6}\ \text{m} = 0.75\ \mu\text{m}$$

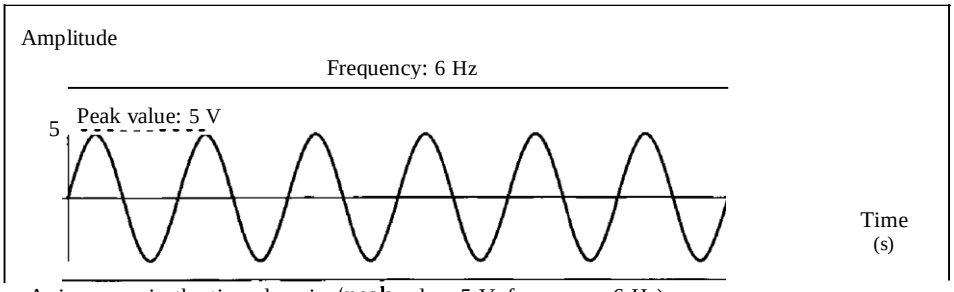
In a coaxial or fiber-optic cable, however, the wavelength is shorter (0.5 μm) because the propagation speed in the cable is decreased.

Time and Frequency Domains

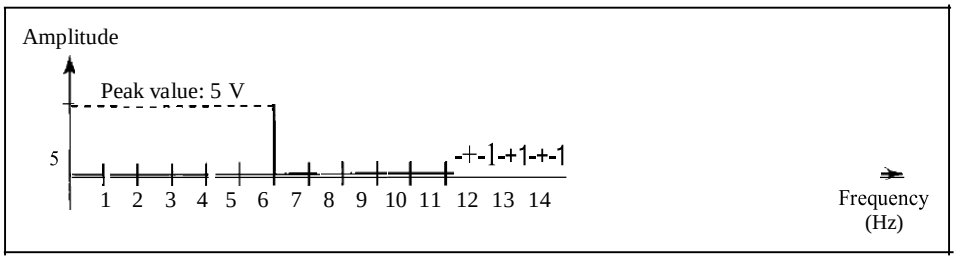
A sine wave is comprehensively defined by its amplitude, frequency, and phase. We have been showing a sine wave by using what is called a time-domain plot. The time- domain plot shows changes in signal amplitude with respect to time (it is an amplitude- versus-time plot). Phase is not explicitly shown 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.

Figure 3.7 *The time-domain andfrequency-domain plots ofa sine wave*



a. A sine wave in the time domain (**peak** value: 5 V, frequency: 6 Hz)



b. The same sine wave in the frequency domain (peak value: 5 V, frequency: 6 Hz)

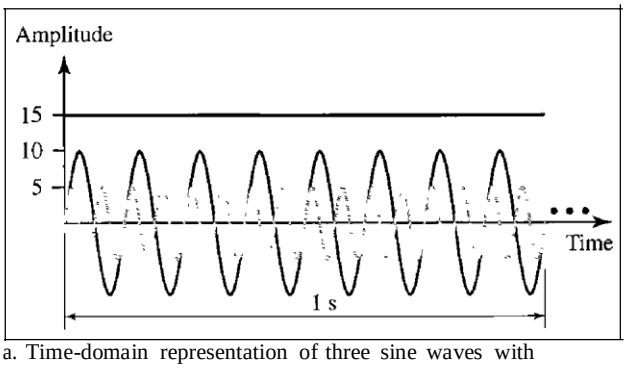
It is obvious that the frequency domain is easy to plot and conveys the information that one can find in a time domain plot. The advantage of the frequency domain is that we can immediately see the values of the frequency and peak amplitude. A complete sine wave is represented by one spike. The position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

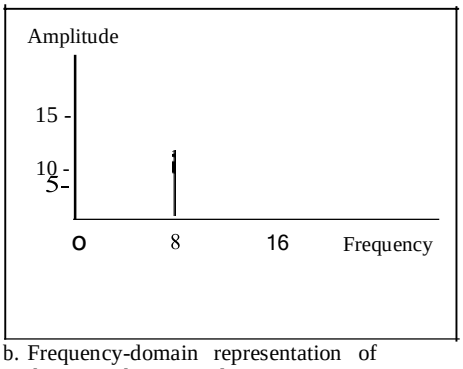
Example 3.7

The frequency domain is more compact and useful when we are dealing with more than one sine wave. For example, Figure 3.8 shows three sine waves, each with different amplitude and fre• quency. All can be represented by three spikes in the frequency domain.

Figure 3.8 *The time domain andfrequency domain ofthree sine waves*



a. Time-domain representation of three sine waves with frequencies 0, 8, and 16



b. Frequency-domain representation of the same three signals

Composite Signals

So far, we have focused on simple sine waves. Simple sine waves have many applica• tions in daily life. We can send a single sine wave to carry electric energy from one place to another. For example, the power company sends a single sine wave with a fre• quency of 60 Hz to distribute electric energy to houses and businesses. As another example, we can use a single sine wave to send an alarm to a security center when a burglar opens a door or window in the house. In the first case, the sine wave is carrying energy; in the second, the sine wave is a signal of danger.

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. As we will see in Chapters 4 and 5, 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. Fourier analysis is discussed in Appendix C; for our purposes, we just present the concept.

According to Fourier analysis, any composite signal is a combination of simple sine waves with different frequencies, amplitudes, and phases. Fourier analysis is discussed in Appendix C.

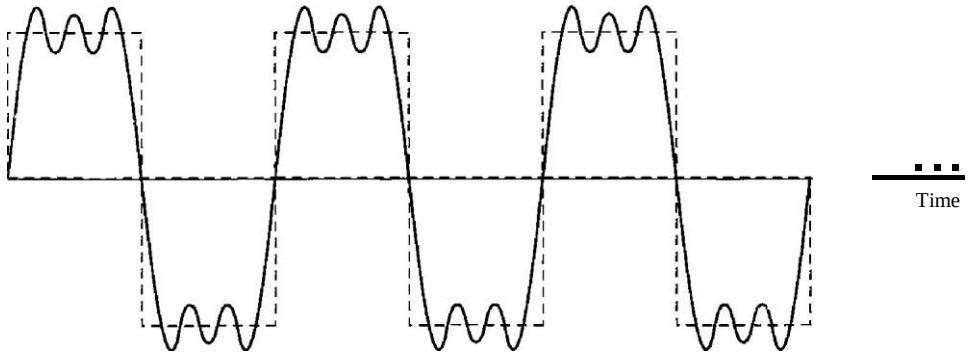
A composite signal can be periodic or nonperiodic. A periodic composite signal can be decomposed into a series of simple sine waves with discrete frequencies—frequencies that have integer values (1, 2, 3, and so on). A nonperiodic composite signal can be decomposed into a combination of an infinite number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

Example 3.8

Figure 3.9 shows a periodic composite signal with frequency f. This type of signal is not typical of those found in data communications. We can consider it to be three alarm systems, each with a different frequency. The analysis of this signal can give us a good understanding of how to decompose signals.

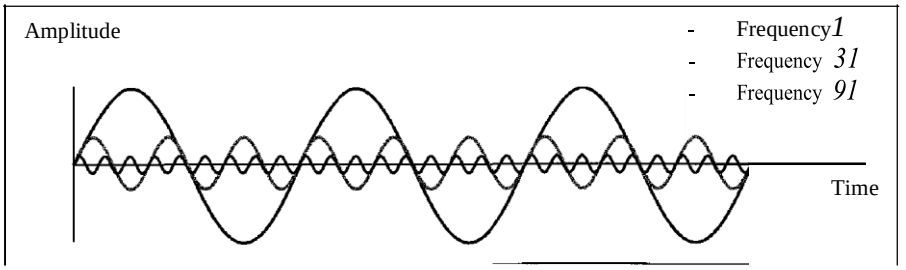
Figure 3.9 *A composite periodic signal*



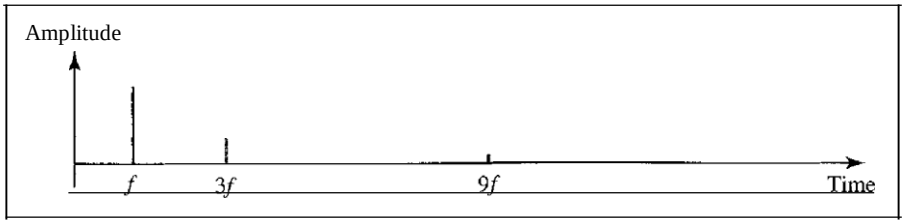
It is very difficult to manually decompose this signal into a series of simple sine waves. However, there are tools, both hardware and software, that can help us do the job. We are not concerned about how it is done; we are only interested in the result. Figure 3.10 shows the result of decomposing the 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

Figure 3.10 *Decomposition ofa composite periodic signal in the time andfrequency domains*



a. Time-domain decomposition of a composite signal



b. Frequency-domain decomposition of the composite signal

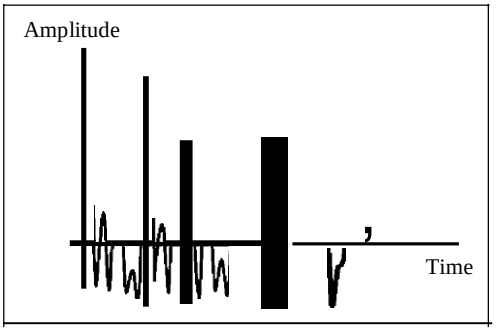
of the sine wave with frequency *f* is the same as the frequency of the composite signal; it is called the fundamental frequency, or first harmonic. The sine wave with frequency 3fhas a frequency of 3 times the fundamental frequency; it is called the third harmonic. The third sine wave with fre• quency *9fhas* a frequency of 9 times the fundamental frequency; it is called the ninth harmonic.

Note that the frequency decomposition of the signal is discrete; it has frequenciesf, *3f,* and *9f* Because *f* is an integral number, *3f* and *9f* are also integral numbers. There are no frequencies such as *1.2f* or *2.6f* The frequency domain of a periodic composite signal is always made of dis• crete spikes.

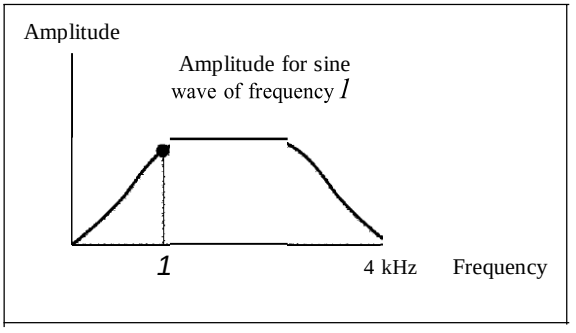
Example 3.9

Figure 3.11 shows a nonperiodic composite signal. It can be the signal created by a microphone or a telephone set when a word or two is pronounced. In this case, the composite signal cannot be peri• odic, because that implies that we are repeating the same word or words with exactly the same tone.

Figure 3.11 *The time andfrequency domains ofa nonperiodic signal*



a. Time domain



b. Frequency domain

In a time-domain representation of this composite signal, there are an infinite num• ber of simple sine frequencies. Although the number of frequencies in a human voice is infinite, the range is limited. A normal human being can create a continuous range of frequencies between 0 and 4 kHz.

Note that the frequency decomposition of the signal yields a continuous curve. There are an infinite number of frequencies between 0.0 and 4000.0 (real values). To find the amplitude related to frequency J, we draw a vertical line atfto intersect the envelope curve. The height of the vertical line is the amplitude of the corresponding frequency.

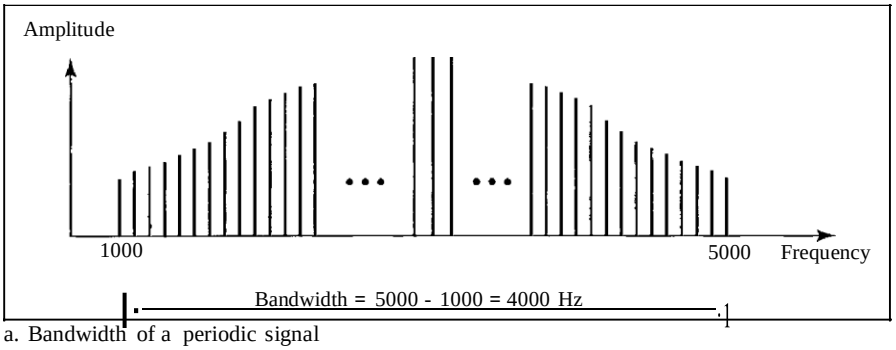
Bandwidth

The range of frequencies contained in a composite signal is its bandwidth. The band• width 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.

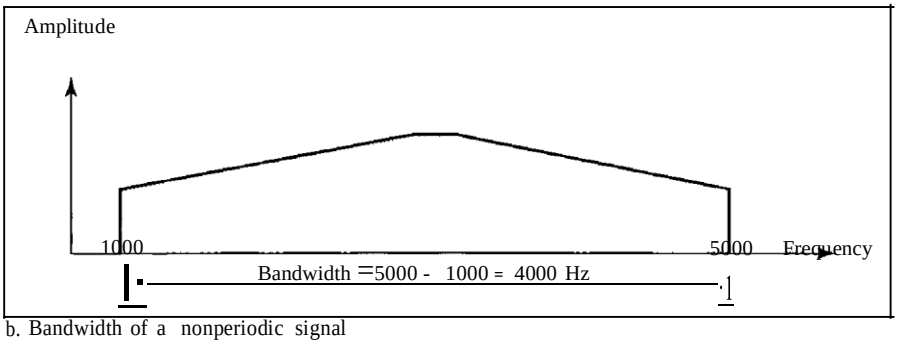
The bandwidth of a composite signal is the difference between the highest and the lowest frequencies contained in that signal.

Figure 3.12 shows the concept of bandwidth. The figure depicts two composite signals, one periodic and the other nonperiodic. The bandwidth of the periodic signal contains all integer frequencies between 1000 and 5000 (1000, 100 I, 1002, ...). The band- width of the nonperiodic signals has the same range, but the frequencies are continuous.

Figure 3.12 *The bandwidth ofperiodic and nonperiodic composite signals*



a. Bandwidth of a periodic signal



b. Bandwidth of a nonperiodic signal

Example 3.10

If a periodic signal is decomposed into five sine waves with frequencies of 100, 300, 500, 700, and 900 Hz, what is its bandwidth? Draw the spectrum, assuming all components have a maxi• mum amplitude of 10 V.

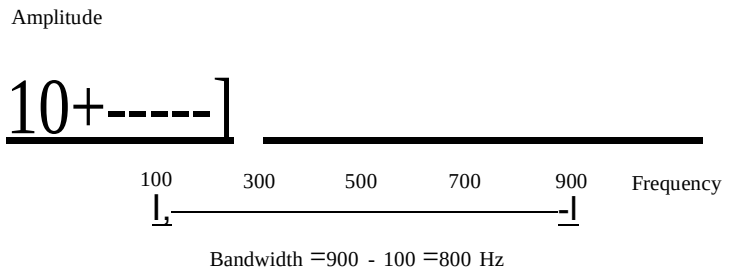
Solution

Letfh be the highest frequency, f_l the lowest frequency, and B the bandwidth. Then

$$B = fh - f_l = 900 - 100 = 800 \text{ Hz}$$

The spectrum has only five spikes, at 100, 300, 500, 700, and 900 Hz (see Figure 3.13).

Figure 3.13 *The bandwidthfor Example 3.10*



Example 3.11

A periodic signal has a bandwidth of 20 Hz. The highest frequency is 60 Hz. What is the lowest frequency? Draw the spectrum if the signal contains all frequencies of the same amplitude.

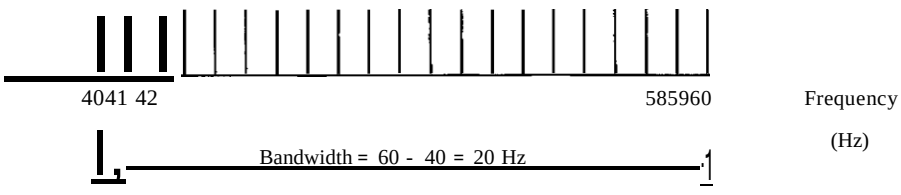
Solution

Letfh be the highest frequency,fz the lowest frequency, and B the bandwidth. Then

$$B = fh - f_l \Rightarrow 20 = 60 - f_l \Rightarrow f_l = 60 - 20 = 40 \text{ Hz}$$

The spectrum contains all integer frequencies. We show this by a series of spikes (see Figure 3.14).

Figure 3.14 *The bandwidth for Example 3.11*



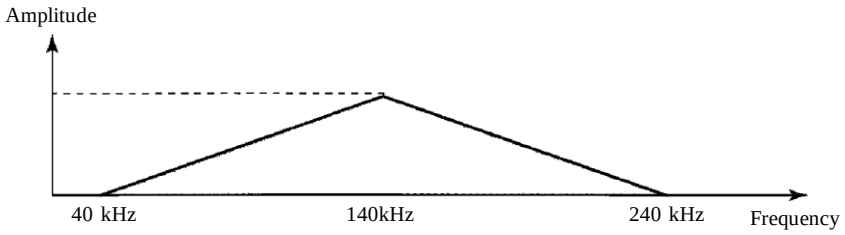
Example 3.12

A nonperiodic composite signal has a bandwidth of 200 kHz, with a middle frequency of 140 kHz and peak amplitude of 20 V. The two extreme frequencies have an amplitude of 0. Draw the frequency domain of the signal.

Solution

The lowest frequency must be at 40 kHz and the highest at 240 kHz. Figure 3.15 shows the fre• quency domain and the bandwidth.

Figure 3.15 *The bandwidth for Example 3.12*



Example 3. *J3*

An example of a nonperiodic composite signal is the signal propagated by an AM radio station, In the United States, each AM radio station is assigned a lO-kHz bandwidth. The total bandwidth ded• icated to AM radio ranges from 530 to 1700 kHz. We will show the rationale behind this lO-kHz bandwidth in Chapter 5.

Example 3. *J4*

Another example of a nonperiodic composite signal is the signal propagated by an FM radio sta• tion. In the United States, each FM radio station is assigned a 200-kHz bandwidth. The total bandwidth dedicated to FM radio ranges from 88 to 108 MHz. We will show the rationale behind this 200-kHz bandwidth in Chapter 5.

Example 3./5

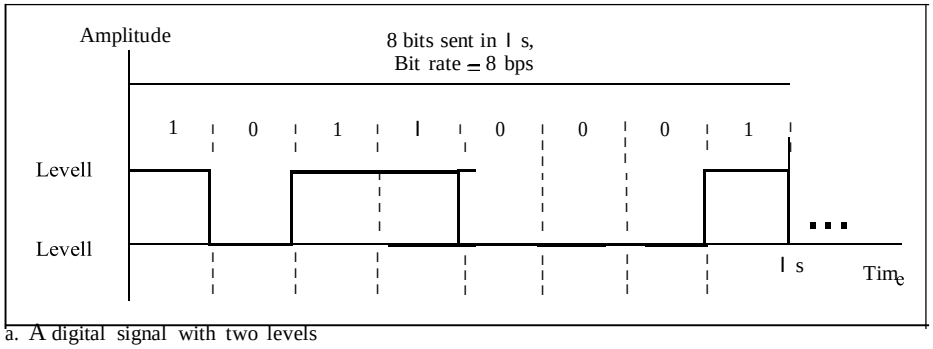
Another example of a nonperiodic composite signal is the signal received by an old-fashioned analog black-and-white TV. A TV screen is made up of pixels (picture elements) with each pixel being either white or black. The screen is scanned 30 times per second. (Scanning is actually 60 times per second, but odd lines are scanned in one round and even lines in the next and then interleaved.) If we assume a resolution of 525 x 700 (525 vertical lines and 700 horizontal lines), which is a ratio of 3:4, we have 367,500 pixels per screen. If we scan the screen 30 times per sec• ond, this is 367,500 x 30 = 11,025,000 pixels per second. The worst-case scenario is alternating black and white pixels. In this case, we need to represent one color by the minimum amplitude and the other color by the maximum amplitude. We can send 2 pixels per cycle. Therefore, we need 11,025,000/2 =5,512,500 cycles per second, or Hz. The bandwidth needed is 5.5124 MHz. This worst-case scenario has such a low probability of occurrence that the assumption is that we need only 70 percent of this bandwidth, which is 3.85 MHz. Since audio and synchronization sig• nals are also needed, a 4-MHz bandwidth has been set aside for each black and white TV chan• nel. An analog color TV channel has a 6-MHz bandwidth.

3.3 DIGITAL SIGNALS

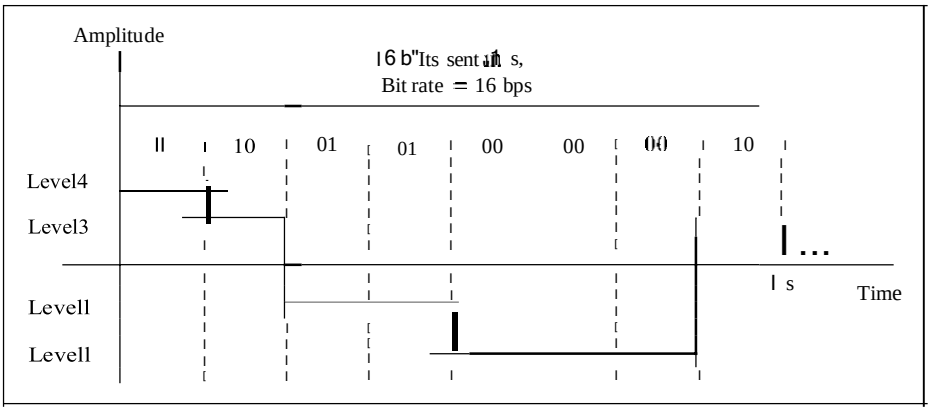
In addition to being represented by an analog signal, information can also be repre• sented by a digital signal. For example, a I can be encoded as a positive voltage and a 0 as zero voltage. A digital signal can have more than two levels. In this case, we can

send more than 1 bit for each level. Figure 3.16 shows two signals, one with two levels and the other with four.

Figure 3.16 *Two digital signals: one with two signal levels and the other with four signal levels*



a. A digital signal with two levels



b. A digital signal with four levels

We send 1 bit per level in part a 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.

Appendix C reviews information about exponential and logarithmic functions.

Example 3.16

A digital signal has eight levels. How many bits are needed per level? We calculate the number of bits from the formula

$$\text{Number of bits per level} = \log_2 8 = 3$$

Each signal level is represented by 3 bits.

Example 3.17

A digital signal has nine levels. How many bits are needed per level? We calculate the number of bits by using the formula. Each signal level is represented by 3.17 bits. However, this answer is not realistic. The number of bits sent per level needs to be an integer as well as a power of 2. For this example, 4 bits can represent one level.

Bit Rate

Most digital signals are nonperiodic, and thus period and frequency are not appropri• ate characteristics. Another *term-bit rate* (instead *ofjrequency)-is* used to describe digital signals. The bit rate is the number of bits sent in Is, expressed in bits per second (bps). Figure 3.16 shows the bit rate for two signals.

Example 3.18

Assume we need to download text documents at the rate of 100 pages per minute. What is the required bit rate of the channel?

Solution

A page is an average of 24 lines with 80 characters in each line. If we assume that one character requires 8 bits, the bit rate is

$$100 \times 24 \times 80 \times 8 = 1{,}636{,}000 \text{ bps} = 1.636 \text{ Mbps}$$

Example 3.19

A digitized voice channel, as we will see in Chapter 4, is made by digitizing a 4-kHz bandwidth analog voice signal. We need to sample the signal at twice the highest frequency (two samples per hertz). We assume that each sample requires 8 bits. What is the required bit rate?

Solution

The bit rate can be calculated as

$$2 \times 4000 \times 8 = 64{,}000 \text{ bps} = 64 \text{ kbps}$$

Example 3.20

What is the bit rate for high-definition TV (HDTV)?

Solution

HDTV uses digital signals to broadcast high quality video signals. The HDTV Screen is normally a ratio of 16 : 9 (in contrast to 4 : 3 for regular TV), which means the screen is wider. There are 1920 by 1080 pixels per screen, and the screen is renewed 30 times per second. Twenty-four bits represents one color pixel. We can calculate the bit rate as

$$1920 \times 1080 \times 30 \times 24 = 1{,}492{,}992{,}000 \text{ or } 1.5 \text{ Gbps}$$

The TV stations reduce this rate to 20 to 40 Mbps through compression.

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 transmis• sion medium.

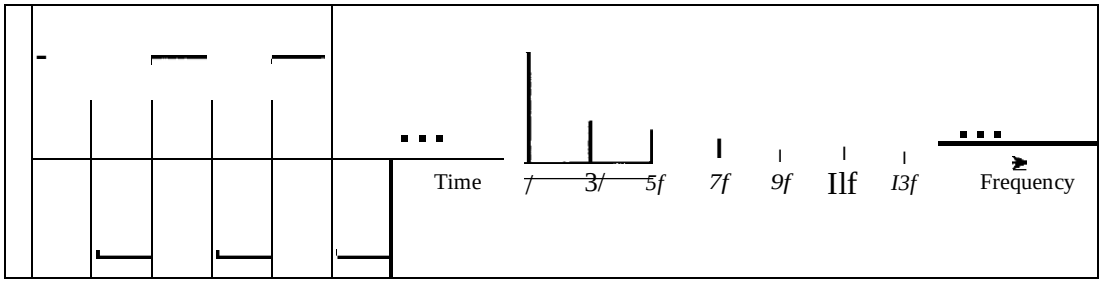
$$\text{Bit length} = \text{propagation speed} \times \text{bit duration}$$

Digital Signal as a Composite Analog Signal

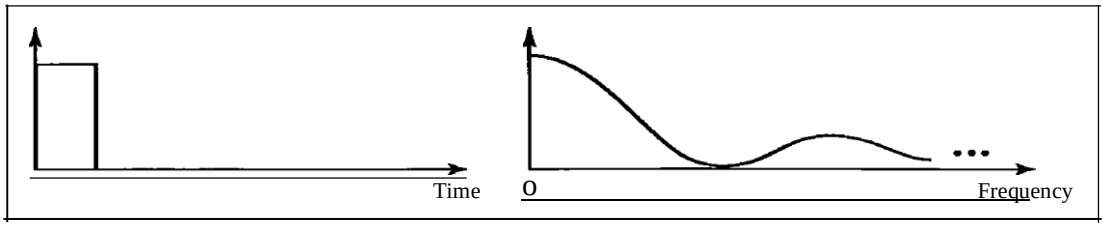
Based on Fourier analysis, a digital signal is a composite analog signal. The bandwidth is infinite, as you may have guessed. We can intuitively corne up with this concept when we consider a digital signal. A digital signal, in the time domain, comprises connected vertical and horizontal line segments. A vertical line in the time domain means a fre• quency 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 nonperiodic, the decomposed signal still has an infinite bandwidth, but the fre• quencies are continuous. Figure 3.17 shows a periodic and a nonperiodic digital signal and their bandwidths.

Figure 3.17 *The time andfrequency domains ofperiodic and nonperiodic digital signals*



a. Time and frequency domains of periodic digital signal



b. Time and frequency domains of nonperiodic digital signal

Note that both bandwidths are infinite, but the periodic signal has discrete frequen• cies while the nonperiodic signal has continuous frequencies.

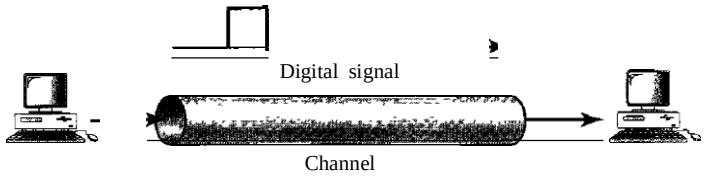
Transmission of Digital Signals

The previous discussion asserts that a digital signal, periodic or nonperiodic, is a com• posite analog signal with frequencies between zero and infinity. For the remainder of the discussion, let us consider the case of a nonperiodic 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. Figure 3.18 shows baseband transmission.

Figure 3.18 *Baseband transmission*

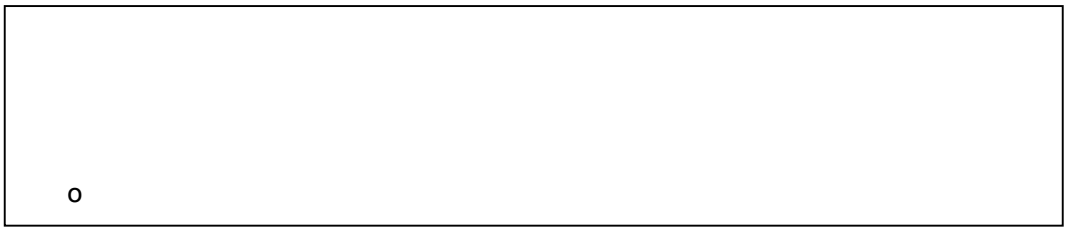


A digital signal is a composite analog signal with an infinite bandwidth.

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. For example, the entire bandwidth of a cable connecting two computers is one single channel. As another example, we may connect several computers to a bus, but not allow more than two stations to communicate at a time. Again we have a low-pass channel, and we can use it for baseband communication. Figure 3.19 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 band• width is ideal, but we cannot have such a channel in real life. However, we can get close.

Figure 3.19 *Bandwidths oftwo low-pass channels*

a. Low-pass channel, wide bandwidth



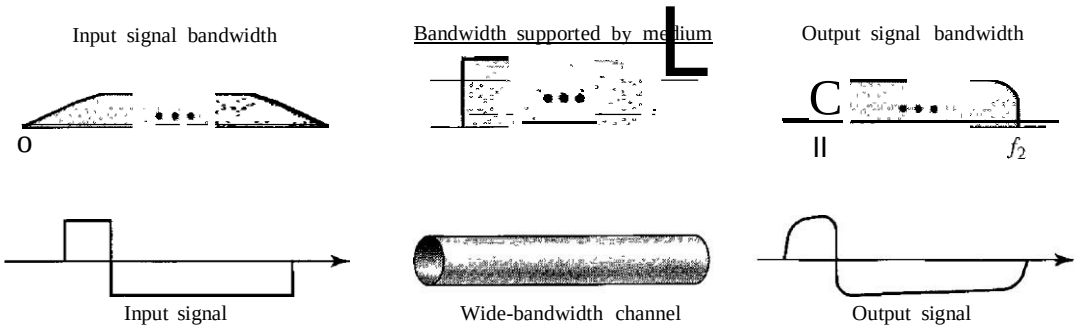
b. Low-pass channel, narrow bandwidth

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 nonperiodic 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), itis 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 3.20. Note that!i is close to zero, andh is very high.

Figure 3.20 *Baseband transmission using a dedicated medium*



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.

Example 3.21

An example of a dedicated channel where the entire bandwidth of the medium is used as one single channel is a LAN. Almost every wired LAN today uses a dedicated channel for two stations communicating with each other. In a bus topology LAN with multipoint connections, only two stations can communicate with each other at each moment in time (timesharing); the other stations need to refrain from sending data. In a star topology LAN, the entire channel between each station and the hub is used for communication between these two entities. We study LANs in Chapter 14.

Case 2: Low-Pass Channel with Limited Bandwidth

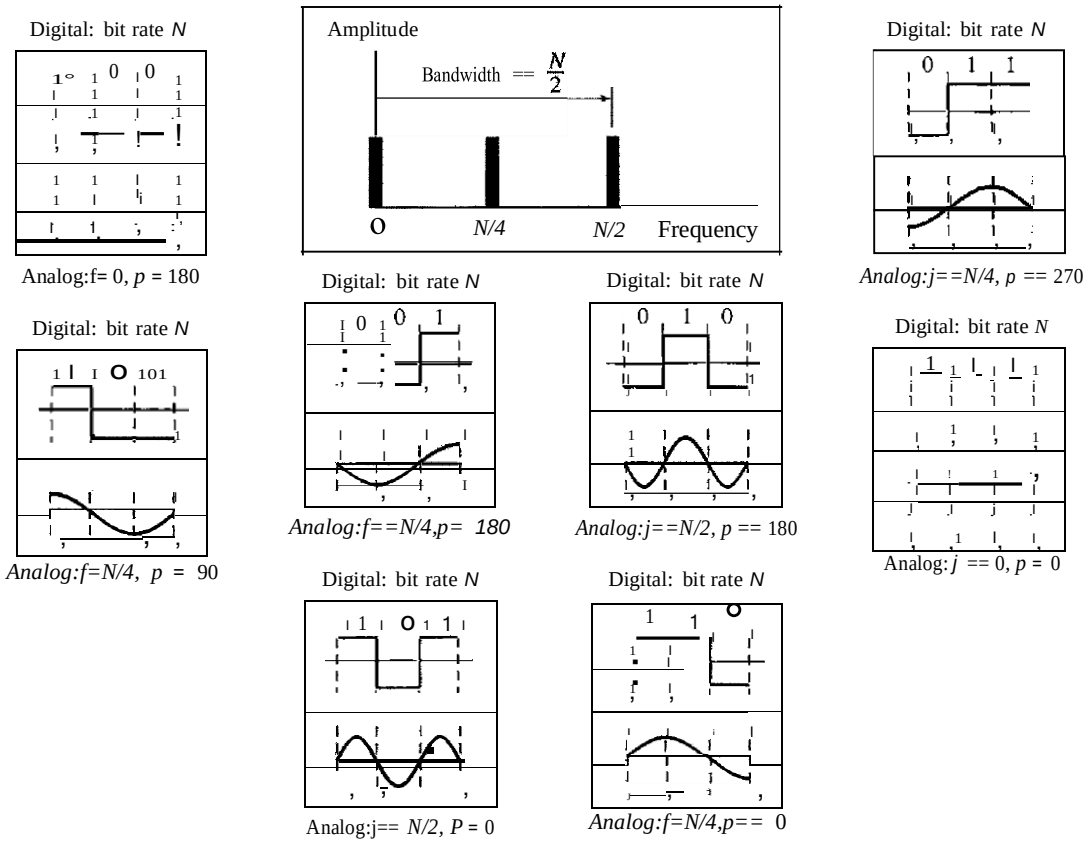
In a low-pass channel with limited bandwidth, we approximate the digital signal with an analog signal. The level of approximation depends on the bandwidth available.

Rough Approximation Let us assume that we have a digital signal of bit rate *N.* If we want to send analog signals to roughly simulate this signal, we need to consider the worstcase, a maximum number of changes in the digital signal. This happens when the signal

carries the sequence 01010101 ... or the sequence 10101010· ... To simulate these two cases, we need an analog signal of frequency $f = N12$. Let 1 be the positive peak value and **O**be the negative peak value. We send 2 bits in each cycle; the frequency of the analog signal is one-half of the bit rate, or *N12.* However, just this one frequency cannot make all patterns; we need more components. The maximum frequency is *NI2.* As an example of this concept, let us see how a digital signal with a 3-bit pattern can be simulated by using analog signals. Figure 3.21 shows the idea. The two similar cases (000 and 111) are simu• lated with a signal with frequency $f = 0$ and a phase of 180° for 000 and a phase of 0° for 111. The two worst cases (010 and 101) are simulated with an analog signal with fre• quency $f = NI2$ and phases of 180° and 0°. The other four cases can only be simulated with an analog signal with $f = NI4$ and phases of 180°, 270°, 90°, and 0°. In other words, we need a channel that can handle frequencies 0, *N14,* and *NI2.* This rough approximation is referred to as using the first harmonic *(NI2)* frequency. The required bandwidth is

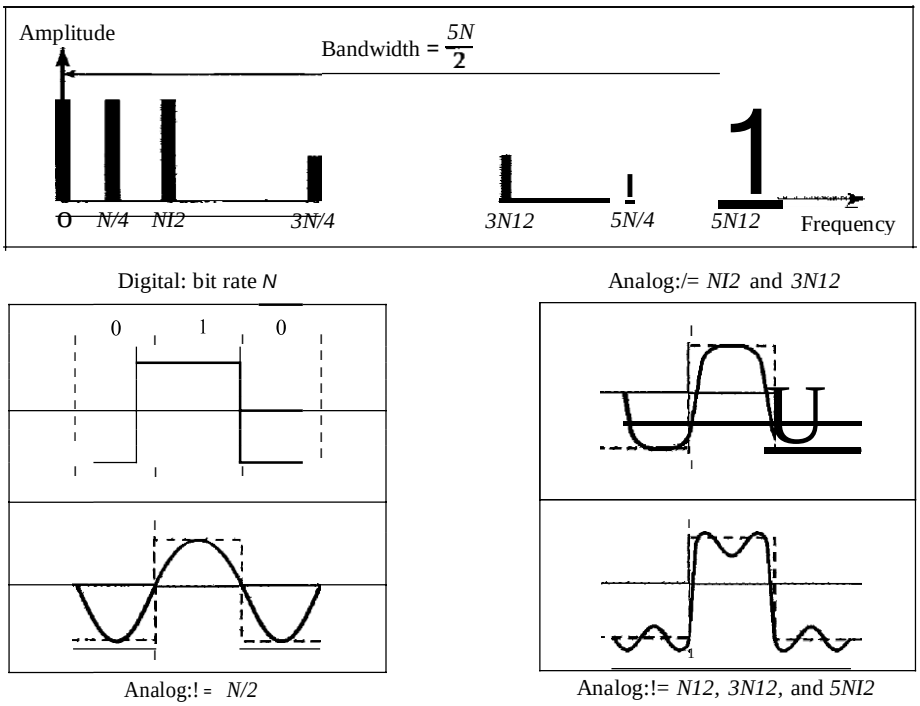
$$\text{Bandwidth} = \frac{N}{2} - 0 = \frac{N}{2}$$

Figure 3.21 *Rough approximation ofa digital signal using the first harmonic for worst case*



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, 7NI2,* and so on. Figure 3.22 shows the effect of this increase for one of the worst cases, the pattern 010.

Figure 3.22 *Simulating a digital signal with three first harmonics*



Note that we have shown only the highest frequency for each hannonic. We use the first, third, and fifth hannonics. The required bandwidth is now *5NJ2*, the difference between the lowest frequency 0 and the highest frequency *5NJ2*. As we emphasized before, we need to remember that the required bandwidth is proportional to the bit rate.

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

By using this method, Table 3.2 shows how much bandwidth we need to send data at different rates.

Table 3.2 *Bandwidth requirements*

Bit Rate	*Harmonic* 1	*Harmonics* 1,3	*Harmonics* 1,3,5
n = 1 kbps	*B=500Hz*	*B*= 1.5 kHz	*B=2.5* kHz
n = 10 kbps	*B=5* kHz	*B*= 15kHz	*B*= 25 kHz
n = 100 kbps	*B*= 50kHz	*B* = 150 kHz	*B*= 250 kHz

Example 3.22

What is the required bandwidth of a low-pass channel if we need to send 1 Mbps by using base• band transmission?

Solution

The answer depends on the accuracy desired.

a. The minimum bandwidth, a rough approximation, is B = bit rate /2, or 500 kHz. We need a low-pass channel with frequencies between 0 and 500 kHz.
b. A better result can be achieved by using the first and the third harmonics with the required bandwidth $B = 3 \times 500$ kHz = 1.5 MHz.
c. Still a better result can be achieved by using the first, third, and fifth harmonics with $B = 5 \times 500$ kHz = 2.5 MHz.

Example 3.23

We have a low-pass channel with bandwidth 100 kHz. What is the maximum bit rate of this channel?

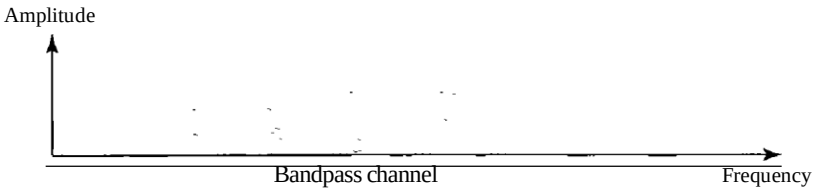
Solution

The maximum bit rate can be achieved if we use the first harmonic. The bit rate is 2 times the available bandwidth, or 200 kbps.

Broadband Transmission (Using Modulation)

Broadband transmission or modulation means changing the digital signal to an analog signal for transmission. Modulation allows us to use a bandpass channel-a channel with a bandwidth that does not start from zero. This type of channel is more available than a low-pass channel. Figure 3.23 shows a bandpass channel.

Figure 3.23 *Bandwidth ofa bandpass channel*

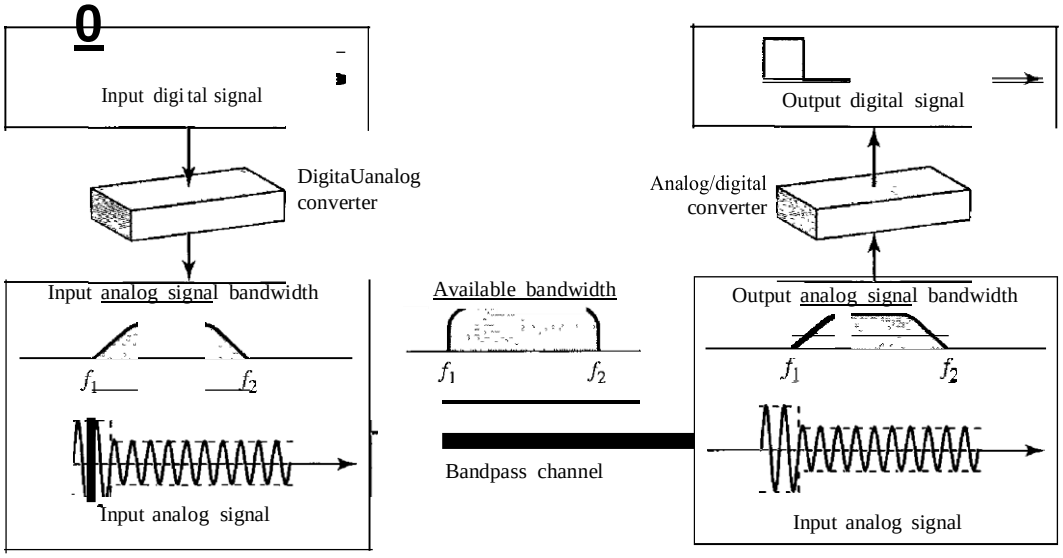


Note that a low-pass channel can be considered a bandpass channel with the lower frequency starting at zero.

Figure 3.24 shows the modulation of a digital signal. In the figure, a digital signal is converted to a composite analog signal. We have used a single-frequency analog signal (called a carrier); the amplitude of the carrier has been changed to look like the digital signal. The result, however, is not a single-frequency signal; it is a composite signal, as we will see in Chapter 5. At the receiver, the received analog signal is converted to digital, and the result is a replica of what has been sent.

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.

Figure 3.24 *Modulation ofa digital signal for transmission on a bandpass channel*



Example 3.24

An example of broadband transmission using modulation is the sending of computer data through a telephone subscriber line, the line connecting a resident to the central telephone office. These lines, installed many years ago, are designed to carry voice (analog signal) with a limited band• width (frequencies between 0 and 4 kHz). Although this channel can be used as a low-pass chan• nel, it is normally considered a bandpass channel. One reason is that the bandwidth is so narrow (4 kHz) that if we treat the channel as low-pass and use it for baseband transmission, the maximum bit rate can be only 8 kbps. The solution is to consider the channel a bandpass channel, convert the digital signal from the computer to an analog signal, and send the analog signal. We can install two converters to change the digital signal to analog and vice versa at the receiving end. The converter, in this case, is called a *modem* (modulator/demodulator), which we discuss in detail in Chapter 5.

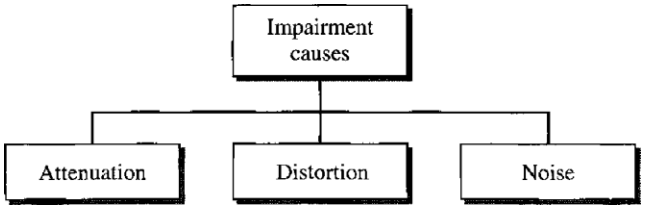
Example 3.25

A second example is the digital cellular telephone. For better reception, digital cellular phones convert the analog voice signal to a digital signal (see Chapter 16). Although the bandwidth allo• cated to a company providing digital cellular phone service is very wide, we still cannot send the digital signal without conversion. The reason is that we only have a bandpass channel available between caller and callee. For example, if the available bandwidth is Wand we allow lOOO cou• ples to talk simultaneously, this means the available channel is *WIlOOO,* just part of the entire bandwidth. We need to convert the digitized voice to a composite analog signal before sending. The digital cellular phones convert the analog audio signal to digital and then convert it again to analog for transmission over a bandpass channel.

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 (see Figure 3.25).

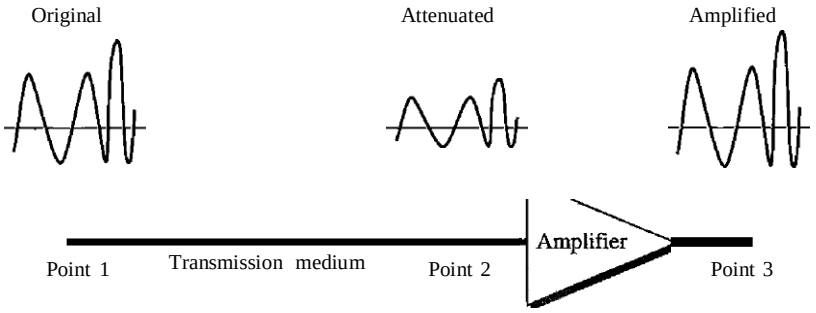
Figure 3.25 *Causes ofimpairment*



Attenuation

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. Figure 3.26 shows the effect of attenuation and amplification.

Figure 3.26 *Attenuation*



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 dif• ferent points. Note that the decibel is negative if a signal is attenuated and positive if a signal is amplified.

$$\mathbf{dB} = \mathbf{10} \log_{10} \frac{P_2}{P_1}$$

Variables *PI* and P_2 are the powers of a signal at points 1 and 2, respectively. Note that some engineering books define the decibel in terms of voltage instead of power. In this case, because power is proportional to the square of the voltage, the formula is dB = 20 log 10 (V_2/V_1). In this text, we express dB in terms of power.

Example 3.26

Suppose a signal travels through a transmission medium and its power is reduced to one-half. This means that $P_2 = \frac{1}{2}P_1$. In this case, the attenuation (loss of power) can be calculated as

$$10 \log_{10} \frac{P_2}{P_1} = 10 \log_{10} \frac{0.5P_1}{P_1} = 10 \log_{10} 0.5 = 10(-0.3) = -3 \text{ dB}$$

A loss of 3 dB (-3 dB) is equivalent to losing one-half the power.

Example 3.27

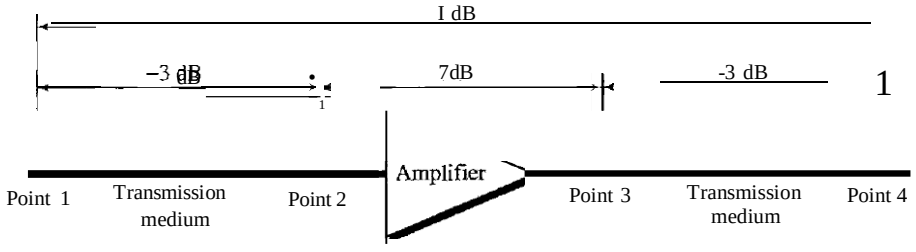
A signal travels through an amplifier, and its power is increased 10 times. This means that $P_2 = 10P_1$. In this case, the amplification (gain of power) can be calculated as

$$10 \log_{10} \frac{P_2}{P_1} = 10 \log_{10} \frac{10P_1}{P_1} = 10 \log_{10} 10 = 10(1) = 10 \text{ dB}$$

Example 3.28

One reason that engineers use the decibel to measure the changes in the strength of a signal is that decibel numbers can be added (or subtracted) when we are measuring several points (cascading) instead of just two. In Figure 3.27 a signal travels from point 1 to point 4. The signal is attenuated by the time it reaches point 2. Between points 2 and 3, the signal is amplified. Again, between points 3 and 4, the signal is attenuated. We can find the resultant decibel value for the signal just by adding the decibel measurements between each set of points.

Figure 3.27 *Decibels for Example 3.28*



In this case, the decibel value can be calculated as

$$dB = -3 + 7 - 3 = +1$$

The signal has gained in power.

Example 3.29

Sometimes the decibel is used to measure signal power in milliwatts. In this case, it is referred to as dB_m and is calculated as $dB_m = 10 \log_{10} P_m$ where P_m is the power in milliwatts. Calculate the power of a signal if its $dB_m = -30$.

Solution

We can calculate the power in the signal as

$$\mathrm{dB_m} = 10 \log_{10} Pm = -30$$
$$\log_{10} P_m = -3 \qquad Pm = 10^{-3}\ \mathrm{mW}$$

Example 3.30

The loss in a cable is usually defined in decibels per kilometer (dB/km). If the signal at the beginning of a cable with -0.3 dB/km has a power of 2 mW, what is the power of the signal at 5 km?

Solution

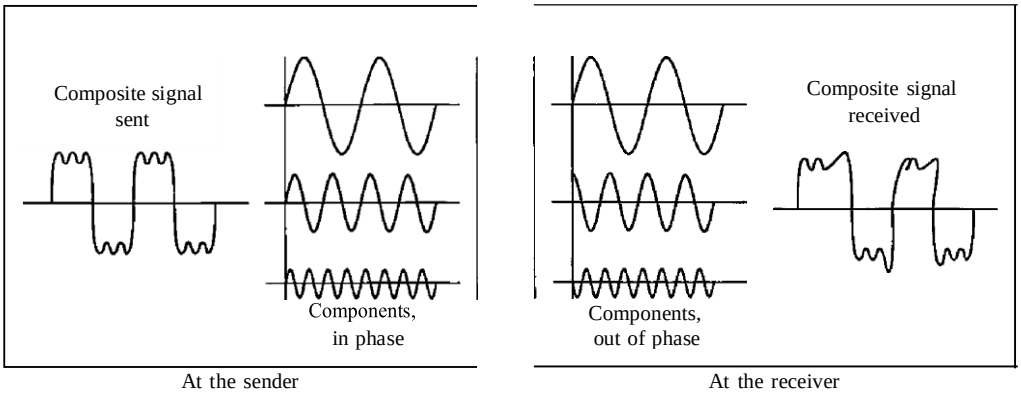
The loss in the cable in decibels is $5 \times (-0.3) = -1.5$ dB. We can calculate the power as

$$\mathrm{dB} = 10 \log_{10} \frac{P_2}{P_1} = -1.5$$
$$\frac{P_2}{P_1} = 10^{-0.15} = 0.71$$
$$P_2 = 0.71 P_1 = 0.7 \times 2 = 1.4\ \mathrm{mW}$$

Distortion

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. Figure 3.28 shows the effect of distortion on a composite signal.

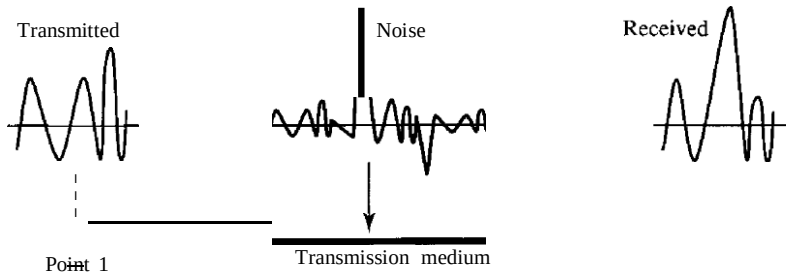
Figure 3.28 *Distortion*



Noise

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 appli• ances. 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 sig• nal 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. We discuss error in Chapter 10.

Figure 3.29 *Noise*



Signal-to-Noise Ratio (SNR)

As we will see later, to find the theoretical bit rate limit, we need to know the ratio of the signal power to the noise power. The signal-to-noise ratio is defined as

$$\text{SNR} = \frac{\text{average signal power}}{\text{average noise power}}$$

We need to consider the average signal power and the average noise power because these may change with time. Figure 3.30 shows the idea of SNR.

SNR is actually the ratio of what is wanted (signal) to what is not wanted (noise). A high SNR means the signal is less corrupted by noise; a low SNR means the signal is more corrupted by noise.

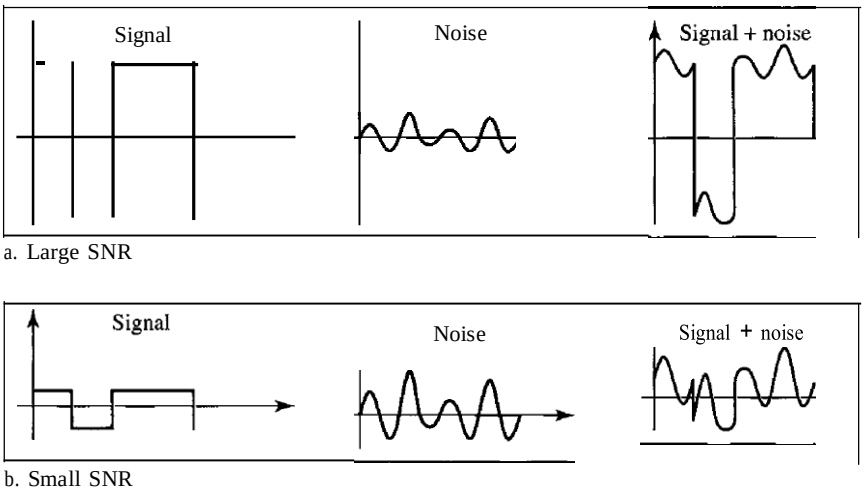
Because SNR is the ratio of two powers, it is often described in decibel units, SNR_{dB}, defined as

$$\text{SNR}_{\text{dB}} = 10 \log_{10} \text{SNR}$$

Example 3.31

The power of a signal is 10 mW and the power of the noise is 1 μW; what are the values of SNR and SNR_{dB}?

Figure 3.30 *Two cases of SNR: a high SNR and a low SNR*



Solution

The values of SNR and SNR_{dB} can be calculated as follows:

$$SNR = \frac{10.000\ \mu W}{1 mW} = 10\ 000$$

$$SNR_{dB} = 10 \log_{10} 10{,}000 = 10 \log_{10} 10^4 = 40$$

Example 3.32

The values of SNR and SNR_{dB} for a noiseless channel are

$$SNR = \frac{\text{signal power}}{0} = \infty$$

$$SNR_{dB} = 10 \log_{10} \infty = \infty$$

We can never achieve this ratio in real life; it is an ideal.

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

$$\text{BitRate} = 2 \times \text{bandwidth} \times \log_2 L$$

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 signa11eve1s. Although the idea is theoretically correct, practically there is a limit. When we increase the number of sig• nal1eve1s, we impose a burden on the receiver. Ifthe 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.

Example 3.33

Does the Nyquist theorem bit rate agree with the intuitive bit rate described in baseband transmission?

Solution

They match when we have only two levels. We said, in baseband transmission, the bit rate is 2 times the bandwidth if we use only the first harmonic in the worst case. However, the Nyquist formula is more general than what we derived intuitively; it can be applied to baseband transmis• sion and modulation. Also, it can be applied when we have two or more levels of signals.

Example 3.34

Consider a noiseless channel with a bandwidth of 3000 Hz transmitting a signal with two signal levels. The maximum bit rate can be calculated as

$$\text{BitRate} = 2 \times 3000 \times \log_2 2 = 6000 \text{ bps}$$

Example 3.35

Consider the same noiseless channel transmitting a signal with four signal levels (for each level, we send 2 bits). The maximum bit rate can be calculated as

$$\text{BitRate} = 2 \times 3000 \times \log_2 4 = 12{,}000 \text{ bps}$$

Example 3.36

We need to send 265 kbps over a noiseless channel with a bandwidth of 20 kHz. How many sig• nallevels do we need?

Solution

We can use the Nyquist formula as shown:

$$265{,}000 = 2 \times 20{,}000 \times \log_2 L$$
$$\log_2 L = 6.625 \qquad L = 2^{6.625} = 98.7 \text{ levels}$$

Since this result is not a power of 2, we need to either increase the number of levels or reduce the bit rate. If we have 128 levels, the bit rate is 280 kbps. If we have 64 levels, the bit rate is 240 kbps.

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:

$$\text{Capacity} = \text{bandwidth} \times \log_2 (1 + \text{SNR})$$

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 methodof transmission.

Example 3.37

Consider an extremely noisy channel in which the value of the signal-to-noise ratio is almost zero. In other words, the noise is so strong that the signal is faint. For this channel the capacity C is calculated as

$$C = B \log_2 (1 + \text{SNR}) = B \log_2 (1 + 0) = B \log_2 1 = B \times 0 = 0$$

This means that the capacity of this channel is zero regardless of the bandwidth. In other words, we cannot receive any data through this channel.

Example 3.38

We can calculate the theoretical highest bit rate of a regular telephone line. A telephone line nor• mally has a bandwidth of 3000 Hz (300 to 3300 Hz) assigned for data communications. The sig• nal- to-noise ratio is usually 3162. For this channel the capacity is calculated as

$$C = B \log_2 (1 + \text{SNR}) = 3000 \log_2 (1 + 3162) = 3000 \log_2 3163$$
$$= 3000 \times 11.62 = 34{,}860 \text{ bps}$$

This means that the highest bit rate for a telephone line is 34.860 kbps. If we want to send data faster than this, we can either increase the bandwidth of the line or improve the signal-to• noise ratio.

Example 3.39

The signal-to-noise ratio is often given in decibels. Assume that $SNR_{dB} = 36$ and the channel bandwidth is 2 MHz. The theoretical channel capacity can be calculated as

$$SNR_{dB} = 10 \log_{10} SNR \rightarrow SNR = 10^{SNR_{dB}/10} \rightarrow SNR = 10^{3.6} = 3981$$
$$C = B \log_2 (1 + SNR) = 2 \times 10^6 \times \log_2 3982 = 24 \text{ Mbps}$$

Example 3.40

For practical purposes, when the SNR is very high, we can assume that SNR + I is almost the same as SNR. In these cases, the theoretical channel capacity can be simplified to

$$C = B \times \frac{SNR_{dB}}{3}$$

For example, we can calculate the theoretical capacity of the previous example as

$$C = 2 \text{ MHz} \times \frac{36}{3} = 24 \text{ Mbps}$$

Using **Both** Limits

In practice, we need to use both methods to find the limits and signal levels. Let us show this with an example.

Example 3.41

We have a channel with a I-MHz bandwidth. The SNR for this channel is 63. What are the appro• priate bit rate and signal level?

Solution

First, we use the Shannon formula to find the upper limit.

$$C = B \log_2 (1 + SNR) = 10^6 \log_2 (1 + 63) = 10^6 \log_2 64 = 6 \text{ Mbps}$$

The Shannon formula gives us 6 Mbps, the upper limit. For better performance we choose something lower, 4 Mbps, for example. Then we use the Nyquist formula to find the number of signal levels.

$$4 \text{Mbps} = 2 \times 1 \text{ MHz} \times \log_2 L \rightarrow L = 4$$

The Shannon capacity gives us the upper limit;
the Nyquist formula tells us how many signal levels we need.

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, in greater detail in Chapter 24. In this section, we introduce terms that we need for future chapters.

Bandwidth

One characteristic that measures network performance is bandwidth. However, the termcan be used in two different contexts with two different measuring values: bandwidth in hertz and bandwidth in bits per second.

Bandwidth in Hertz

We have discussed this concept. Bandwidth in hertz is the range of frequencies con• tained in a composite signal or the range of frequencies a channel can pass. For exam•ple, we can say the bandwidth of a subscriber telephone line is 4 kHz.

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. For example, one can say the bandwidth of a Fast Ethernet network (or the links in this network) is a maximum of 100 Mbps. This means that this network can send 100 Mbps.

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. We discuss this relationship in Chapters 4 and 5.

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 transmis• sion in a channel or link.

Example 3.42

The bandwidth of a subscriber line is 4 kHz for voice or data. The bandwidth of this line for data trans• mission can be up to 56,000 bps using a sophisticated modem to change the digital signal to analog.

Example 3.43

If the telephone company improves the quality of the line and increases the bandwidth to 8 kHz,we can send 112,000 bps by using the same technology as mentioned in Example 3.42.

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. For example, we may have a link with a bandwidth of 1 Mbps, but the devices connected to the end of the link may handle only 200 kbps. This means that we cannot send more than 200 kbps through this link.

Imagine a highway designed to transmit 1000 cars per minute from one point to another. However, if there is congestion on the road, this figure may be reduced to 100 cars per minute. The bandwidth is 1000 cars per minute; the throughput is 100 carsper minute.

Example 3.44

A network with bandwidth of 10 Mbps can pass only an average of 12,000 frames per minute with each frame carrying an average of 10,000 bits. What is the throughput of this network?

Solution

We can calculate the throughput as

$$\text{Throughput} = \frac{12{,}000 \times 10{,}000}{60} = 2 \text{ Mbps}$$

The throughput is almost one-fifth of the bandwidth in this case.

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.

$$\text{Latency} = \text{propagation time} + \text{transmission time} + \text{queuing time} + \text{processing delay}$$

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 propaga• tion speed.

$$\text{Propagation time} = \frac{\text{Distance}}{\text{Propagation speed}}$$

The propagation speed of electromagnetic signals depends on the medium and on the frequency of the signaL For example, in a vacuum, light is propagated with a speed of 3 x 10^8 mfs. It is lower in air; it is much lower in cable.

Example 3.45

What is the propagation time if the distance between the two points is 12,000 km? Assume the propagation speed to be 2.4×10^8 mls in cable.

Solution

We can calculate the propagation time as

$$\text{Propagation time} = \frac{12000 \times 1000}{2.4 \times 10^8} = 50 \text{ ms}$$

The example shows that a bit can go over the Atlantic Ocean in only 50 ms if there is a direct cable between the source and the destination.

Tra/lsmissio/l 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 ear• lier; 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.

$$\text{Transmission time} = \frac{\text{Message size}}{\text{Bandwidth}}$$

Example 3.46

What are the propagation time and the transmission time for a 2.5-kbyte message (an e-mail) if the bandwidth of the network is 1 Gbps? Assume that the distance between the sender and the receiver is 12,000 km and that light travels at 2.4×10^8 *mls.*

Solution

We can calculate the propagation and transmission time as

$$\text{Propagation time} = \frac{12000 \times 1000}{2.4 \times 10^8} = 50 \text{ ms}$$

$$\text{Transmission time} = \frac{2500 \times 8}{10^9} = 0.020 \text{ ms}$$

Note that in this case, because the message is short and the bandwidth is high, the dominant factor is the propagation time, not the transmission time. The transmission time can be ignored.

Example 3.47

What are the propagation time and the transmission time for a 5-Mbyte message (an image) if the bandwidth of the network is 1 Mbps? Assume that the distance between the sender and the receiver is 12,000 km and that light travels at 2.4×10^8 mls.

Solution

We can calculate the propagation and transmission times as

$$\text{Propagation time} = \frac{12\,000 \times 1000}{2.4 \times 10^8} = 50 \text{ ms}$$

$$\text{Transmission time} = \frac{5{,}000{,}000 \times 8}{10^6} = 40 \text{ s}$$

Note that in this case, because the message is very long and the bandwidth is not very high, the dominant factor is the transmission time, not the propagation time. The propagation time can be ignored.

Queuing Time

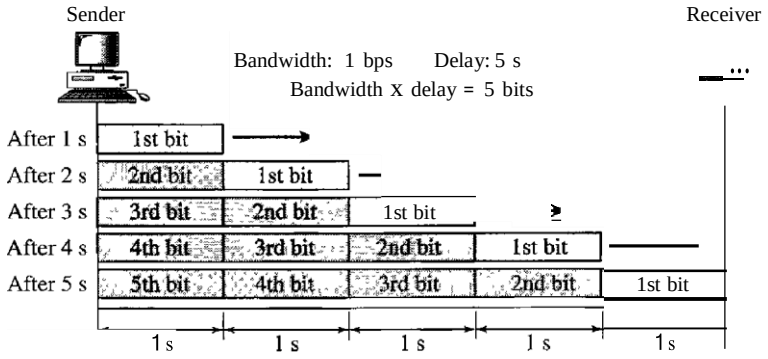
The third component in latency is the queuing time, the time needed for each intermediate or end device to hold the message before it can be processed. The queuing time is not a fixed factor; it changes with the load imposed on the network. When there is heavy traffic on the network, the queuing time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

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.

O Case 1. Figure 3.31 shows case 1.

Figure 3.31 *Filling the link with bits for case 1*

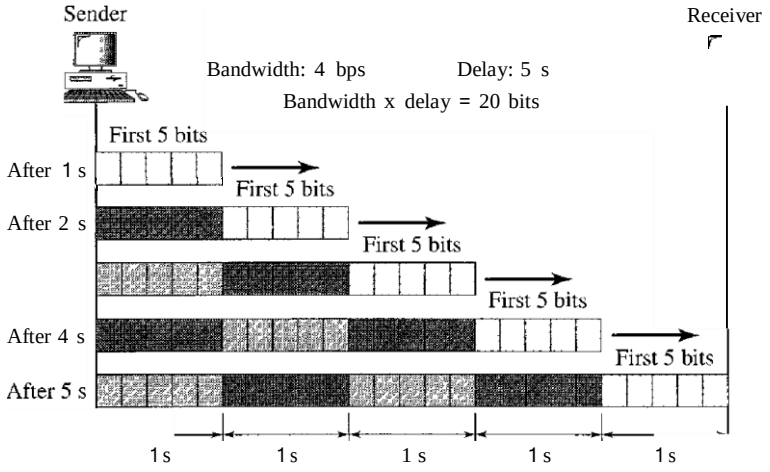


Let us assume that we have a link with a bandwidth of 1 bps (unrealistic, but good for demonstration purposes). We also assume that the delay of the link is 5 s (also unrealistic). We want to see what the bandwidth-delay product means in this case.

Looking at figure, we can say that this product 1 x 5 is the maximum number of bits that can fill the link. There can be no more than 5 bits at any time on the link.

- Case 2. Now assume we have a bandwidth of 4 bps. Figure 3.32 shows that there can be maximum 4 x 5 = 20 bits on the line. The reason is that, at each second, there are 4 bits on the line; the duration of each bit is 0.25 s.

Figure 3.32 *Filling the link with bits in case 2*



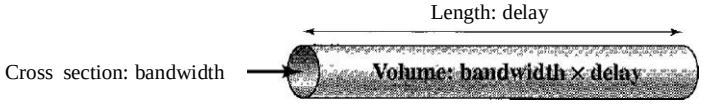
The above two cases show that the product of bandwidth and delay is the number of bits that can fill the link. This measurement is important if we need to send data in bursts and wait for the acknowledgment of each burst before sending the next one. To use the maximum capability of the link, we need to make the size of our burst 2 times the product of bandwidth and delay; we need to fill up the full-duplex channel (two directions). The sender should send a burst of data of (2 x bandwidth x delay) bits. The sender then waits for receiver acknowledgment for part of the burst before sending another burst. The amount 2 x bandwidth x delay is the number of bits that can be in transition at any time.

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

Example 3.48

We can think about the link between two points as a pipe. The cross section of the pipe represents the bandwidth, and the length of the pipe represents the delay. We can say the volume of the pipe defines the bandwidth-delay product, as shown in Figure 3.33.

Figure 3.33 *Concept ofbandwidth-delay product*



Jitter

Another performance issue that is related to delay is **jitter.** We can roughly say that jitteris 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 is40 ms, then the real-time application that uses the packets endures jitter. We discuss jitterin greater detail in Chapter 29.

3.7 SUMMARY

- Data must be transformed to electromagnetic signals to be transmitted.
- Data can be analog or digital. Analog data are continuous and take continuous values. Digital data have discrete states and take discrete values.
- 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.
- In data communications, we commonly use periodic analog signals and nonperiodic digital signals.
- Frequency and period are the inverse of each other.
- Frequency is the rate of change with respect to time.
- Phase describes the position of the waveform relative to time O.
- A complete sine wave in the time domain can be represented by one single spike in the frequency domain.
- 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.
- According to Fourier analysis, any composite signal is a combination of simple sine waves with different frequencies, amplitudes, and phases.
- The bandwidth of a composite signal is the difference between the highest and the lowest frequencies contained in that signal.
- A digital signal is a composite analog signal with an infinite bandwidth.
- 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.
- If the available channel is a bandpass channel, we cannot send a digital signal directly to the channel; we need to convert the digital signal to an analog signal before transmission.
- For a noiseless channel, the Nyquist bit rate formula defines the theoretical maximum bit rate. For a noisy channel, we need to use the Shannon capacity to find the maximum bit rate.
- Attenuation, distortion, and noise can impair a signal.
- Attenuation is the loss of a signal's energy due to the resistance of the medium.
- Distortion is the alteration of a signal due to the differing propagation speeds of each of the frequencies that make up a signal.
- Noise is the external energy that corrupts a signal.
- The bandwidth-delay product defines the number of bits that can fill the link.

3.8 PRACTICE SET

Review Questions

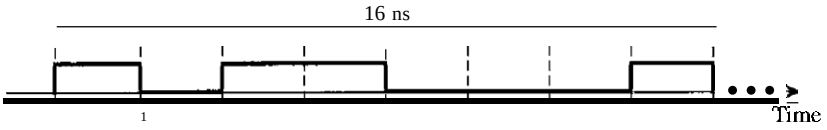
1. What is the relationship between period and frequency?
2. What does the amplitude of a signal measure? What does the frequency of a signal measure? What does the phase of a signal measure?
3. How can a composite signal be decomposed into its individual frequencies?
4. Name three types of transmission impairment.
5. Distinguish between baseband transmission and broadband transmission.
6. Distinguish between a low-pass channel and a band-pass channel.
7. What does the Nyquist theorem have to do with communications?
8. What does the Shannon capacity have to do with communications?
9. Why do optical signals used in fiber optic cables have a very short wave length?
10. Can we say if a signal is periodic or nonperiodic by just looking at its frequency domain plot? How?
11. Is the frequency domain plot of a voice signal discrete or continuous?
12. Is the frequency domain plot of an alarm system discrete or continuous?
13. We send a voice signal from a microphone to a recorder. Is this baseband or broad• band transmission?
14. We send a digital signal from one station on a LAN to another station. Is this base• band or broadband transmission?
15. We modulate several voice signals and send them through the air. Is this baseband or broadband transmission?

Exercises

16. Given the frequencies listed below, calculate the corresponding periods.
    a. 24Hz
    b. 8 MHz
    c. 140 KHz
17. Given the following periods, calculate the corresponding frequencies.
    a. 5 s
    b. 12 μs
    c. 220 ns
18. What is the phase shift for the foIlowing?
    a. A sine wave with the maximum amplitude at time zero
    b. A sine wave with maximum amplitude after 1/4 cycle
    c. A sine wave with zero amplitude after 3/4 cycle and increasing
19. What is the bandwidth of a signal that can be decomposed into five sine waves with frequencies at 0, 20, 50, 100, and 200 Hz? All peak amplitudes are the same. Draw the bandwidth.

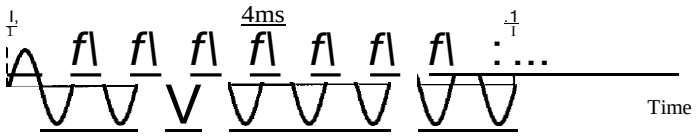
20. A periodic composite signal with a bandwidth of 2000 Hz is composed of two sine waves. The first one has a frequency of 100 Hz with a maximum amplitude of 20 V; the second one has a maximum amplitude of 5 V. Draw the bandwidth.
21. Which signal has a wider bandwidth, a sine wave with a frequency of 100 Hz or a sine wave with a frequency of 200 Hz?
22. What is the bit rate for each of the following signals?
    a. A signal in which 1 bit lasts 0.001 s
    b. A signal in which 1 bit lasts 2 ms
    c. A signal in which 10 bits last 20 μs
23. A device is sending out data at the rate of 1000 bps.
    a. How long does it take to send out 10 bits?
    b. How long does it take to send out a single character (8 bits)?
    c. How long does it take to send a file of 100,000 characters?
24. What is the bit rate for the signal in Figure 3.34?

Figure 3.34 *Exercise 24*



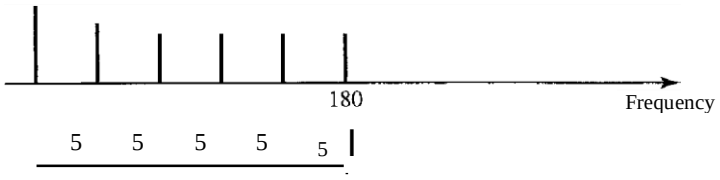
25. What is the frequency of the signal in Figure 3.35?

Figure 3.35 *Exercise 25*



26. What is the bandwidth of the composite signal shown in Figure 3.36.

Figure 3.36 *Exercise 26*



27. A periodic composite signal contains frequencies from 10 to 30 KHz, each with an amplitude of 10 V. Draw the frequency spectrum.
28. A non-periodic composite signal contains frequencies from 10 to 30 KHz. The peak amplitude is 10 V for the lowest and the highest signals and is 30 V for the 20-KHz signal. Assuming that the amplitudes change gradually from the minimum to the maximum, draw the frequency spectrum.
29. A TV channel has a bandwidth of 6 MHz. If we send a digital signal using one channel, what are the data rates if we use one harmonic, three harmonics, and five harmonics?
30. A signal travels from point A to point B. At point A, the signal power is 100 W. At point B, the power is 90 W. What is the attenuation in decibels?
31. The attenuation of a signal is -10 dB. What is the final signal power if it was originally 5 W?
32. A signal has passed through three cascaded amplifiers, each with a 4 dB gain. What is the total gain? How much is the signal amplified?
33. If the bandwidth of the channel is 5 Kbps, how long does it take to send a frame of 100,000 bits out of this device?
34. The light of the sun takes approximately eight minutes to reach the earth. What is the distance between the sun and the earth?
35. A signal has a wavelength of 1 μm in air. How far can the front of the wave travel during 1000 periods?
36. A line has a signal-to-noise ratio of 1000 and a bandwidth of 4000 KHz. What is the maximum data rate supported by this line?
37. We measure the performance of a telephone line (4 KHz of bandwidth). When the signal is 10 V, the noise is 5 mV. What is the maximum data rate supported by this telephone line?
38. A file contains 2 million bytes. How long does it take to download this file using a 56-Kbps channel? 1-Mbps channel?
39. A computer monitor has a resolution of 1200 by 1000 pixels. If each pixel uses 1024 colors, how many bits are needed to send the complete contents of a screen?
40. A signal with 200 milliwatts power passes through 10 devices, each with an average noise of 2 microwatts. What is the SNR? What is the SNR_{dB}?
41. If the peak voltage value of a signal is 20 times the peak voltage value of the noise, what is the SNR? What is the SNR_{dB}?
42. What is the theoretical capacity of a channel in each of the following cases:
    a. Bandwidth: 20 KHz $SNR_{dB} = 40$
    b. Bandwidth: 200 KHz $SNR_{dB} = 4$
    c. Bandwidth: 1 MHz $SNR_{dB} = 20$
43. We need to upgrade a channel to a higher bandwidth. Answer the following questions:
    a. How is the rate improved if we double the bandwidth?
    b. How is the rate improved if we double the SNR?

44. We have a channel with 4 KHz bandwidth. If we want to send data at 100 Kbps, what is the minimum SNR_{dB}? What is SNR?
45. What is the transmission time of a packet sent by a station if the length of the packet is 1 million bytes and the bandwidth of the channel is 200 Kbps?
46. What is the length of a bit in a channel with a propagation speed of 2×10^8 mls if the channel bandwidth is
    a. 1 Mbps?
    b. 10 Mbps?
    c. 100 Mbps?
47. How many bits can fit on a link with a 2 ms delay if the bandwidth of the link is
    a. 1 Mbps?
    b. 10 Mbps?
    c. 100 Mbps?
48. What is the total delay (latency) for a frame of size 5 million bits that is being sent on a link with 10 routers each having a queuing time of 2 μs and a processing time of 1 μs. The length of the link is 2000 Km. The speed of light inside the link is 2×10^8 *mls*. The link has a bandwidth of 5 Mbps. Which component of the total delay is dominant? Which one is negligible?

CHAPTER 4

Digital Transmission

A computer network is designed to send information from one point to another. This information needs to be converted to either a digital signal or an analog signal for trans• mission. In this chapter, we discuss the first choice, conversion to digital signals; in Chapter 5, we discuss the second choice, conversion to analog signals.

We discussed the advantages and disadvantages of digital transmission over analog transmission in Chapter 3. In this chapter, we show the schemes and techniques that we use to transmit data digitally. First, we discuss digital-to-digital conversion tech• niques, 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

In Chapter 3, we discussed data and signals. We said that data can be either digital or analog. We also said that signals that represent data can also be digital or analog. In this section, we see how we can represent digital data by using digital signals. The conver•sion involves three techniques: line coding, block coding, and scrambling. Line codingis always needed;block coding and scrambling mayor may not be needed.

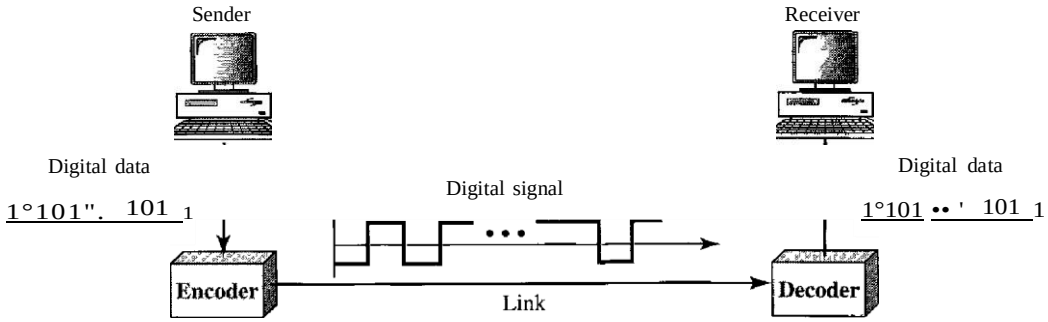
Line Coding

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 com• puter memory as sequences of bits (see Chapter 1). 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. Figure 4.1 shows the process.

Characteristics

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

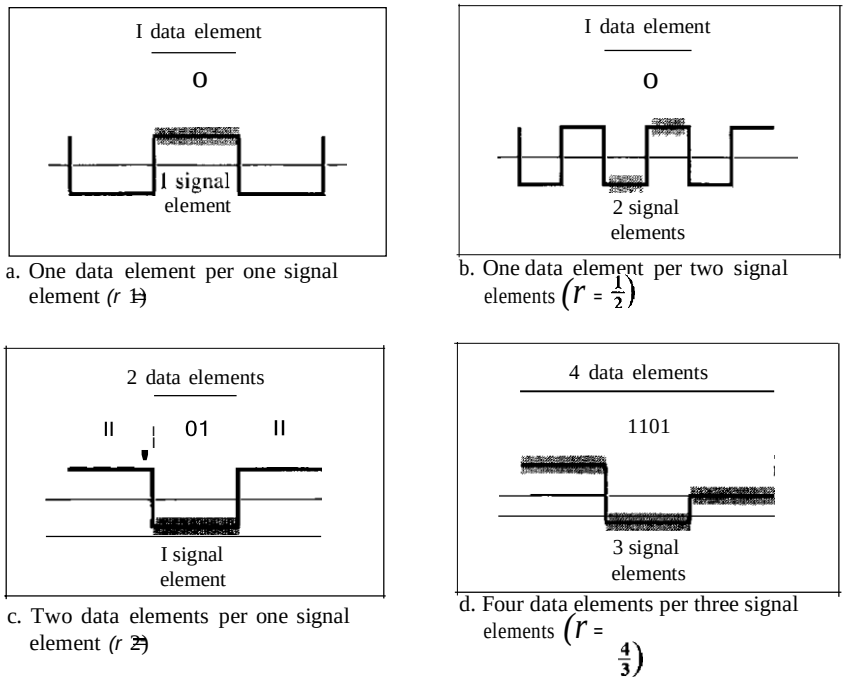
Figure 4.1 *Line coding and decoding*



Signal Element Versus Data Element Let us distinguish between a data element and a signal element. In data communications, our goal is to send data elements. A data element is the smallest entity that can represent a piece of information: this is the bit. In digital data communications, a signal element carries data elements. A signal element is the shortest unit (timewise) of a digital signal. In other words, data elements are what we need to send; signal elements are what we can send. Data elements are being carried; signal elements are the carriers.

We define a ratio *r* which is the number of data elements carried by each signal ele• ment. Figure 4.2 shows several situations with different values of *r.*

Figure 4.2 *Signal element versus data element*



a. One data element per one signal element ($r = 1$)

b. One data element per two signal elements ($r = \frac{1}{2}$)

c. Two data elements per one signal element ($r = 2$)

d. Four data elements per three signal elements ($r = \frac{4}{3}$)

In part a of the figure, one data element is carried by one signal element ($r = 1$). In part b of the figure, we need two signal elements (two transitions) to carry each data

element ($r = \frac{1}{2}$). We will see later that the extra signal element is needed to guarantee synchronization. In part c of the figure, a signal element carries two data elements ($r = 2$). Finally, in part d, a group of 4 bits is being carried by a group of three signal elements ($r = \frac{4}{3}$). For every line coding scheme we discuss, we will give the value of *r.*

An analogy may help here. Suppose each data element is a person who needs to be carried from one place to another. We can think of a signal element as a vehicle that can carry people. When $r = 1$, it means each person is driving a vehicle. When $r > 1$, it means more than one person is travelling in a vehicle (a carpool, for example). We can also have the case where one person is driving a car and a trailer ($r = \frac{1}{2}$).

Data Rate Versus Signal Rate The data rate defines the number of data elements (bits) sent in Is. The unit is bits per second (bps). The signal rate is the number of sig• nal elements sent in Is. The unit is the baud. There are several common terminologies used in the literature. 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. In our vehicle-people analogy, we need to carry more people in fewer vehicles to prevent traffic jams. We have a limited *bandwidth* in our transportation system.

We now need to consider the relationship between data rate and signal rate (bit rate and baud rate). This relationship, of course, depends on the value of *r.* It also depends on the data pattern. If we have a data pattern of all 1s or all Os, the signal rate may be different from a data pattern of alternating Os and Is. To derive a formula for the rela• tionship, we need to define 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. In data communications, we are usually interested in the average case. We can formu• late the relationship between data rate and signal rate as

$$S = c \times N \times \frac{1}{r} \quad \text{baud}$$

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.

Example 4.1

A signal is carrying data in which one data element is encoded as one signal element ($r = 1$). If the bit rate is 100 kbps, what is the average value of the baud rate if c is between 0 and l?

Solution

We assume that the average value of c is $\frac{1}{2}$. The baud rate is then

$$S = c \times N \times \frac{1}{r} = \frac{1}{2} \times 100{,}000 \times 1 = 50{,}000 = 50 \text{ kbaud}$$

Bandwidth We discussed in Chapter 3 that a digital signal that carries infonnation is nonperiodic. We also showed that the bandwidth of a nonperiodic signal is continuous

with an infinite range. However, most digital signals we encounter in real life have a

bandwidth with finite values. In other words, 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, when we talk about the bandwidth of a dig• ital signal, we need to remember that we are talking about this effective bandwidth.

Although the actual bandwidth ofa digital signal is infinite, the effective bandwidth is finite.

We can say that the baud rate, not the bit rate, determines the required bandwidthfor a digital signal. If we use the transpOltation analogy, the number of vehicles affects the traffic, not the number of people being carried. More changes in the signal mean injecting more frequencies into the signal. (Recall that frequency means change and change means frequency.) The bandwidth reflects the range of frequencies we need. There is a relationship between the baud rate (signal rate) and the bandwidth. Band• width is a complex idea. When we talk about the bandwidth, we normally define a range of frequencies. We need to know where this range is located as well as the valuesof the lowest and the highest frequencies. In addition, the amplitude (if not the phase) of each component is an impOltant issue. In other words, we need more information about the bandwidth than just its value; we need a diagram of the bandwidth. We will show the bandwidth for most schemes we discuss in the chapter. 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

$$B_{min} = c \times N \times \frac{1}{r}$$

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

$$N_{max} = \frac{1}{c} \times B \times r$$

Example 4.2

The maximum data rate of a channel (see Chapter 3) is $N_{max} = 2 \times B \times \log 2L$ (defined by the Nyquist formula). Does this agree with the previous formula for N_{max}?

Solution

A signal with L levels actually can carry $\log_2 L$ bits per level. If each level corresponds to one sig• nal element and we assume the average case ($c = \frac{1}{2}$), then we have

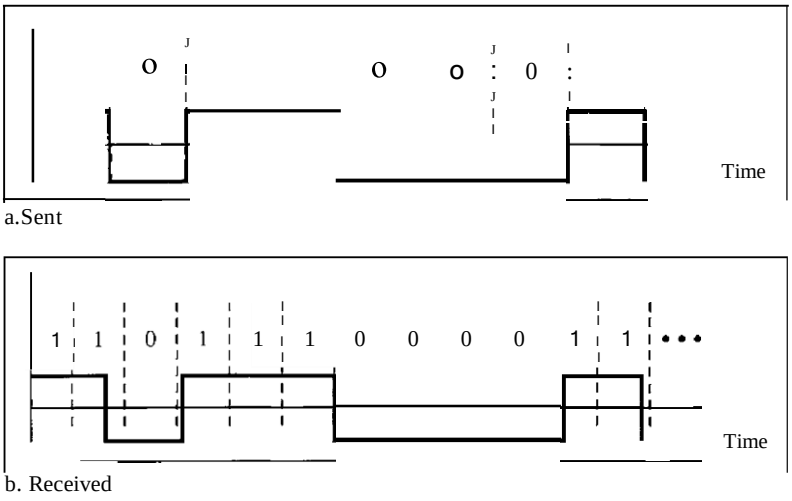
$$N_{max} = \frac{1}{c} \times B \times r = 2 \times B \times \log_2 L$$

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 ele• ment. A long string of Os 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.

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.

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.

Figure 4.3 *Effect of lack of synchronization*



A self-synchronizing digital signal includes timing information in the data being transmitted. This can be achieved if there are transitions in the signal that alert the receiver to the beginning, middle, or end of the pulse. If the receiver's clock is out of synchronization, these points can reset the clock.

Example 4.3

In a digital transmission, the receiver clock is 0.1 percent faster than the sender clock. How many extra bits per second does the receiver receive if the data rate is 1 kbps? How many if the data rate is 1 Mbps?

Solution

At 1 kbps, the receiver receives 1001 bps instead of 1000 bps.

1000 bits sent 1001 bits received 1 extra bps

At 1 Mbps, the receiver receives 1,001,000 bps instead of 1,000,000 bps.

1,000,000 bits sent 1,001,000 bits received 1000 extra bps

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 transmis• sion. Some encoding schemes that we will discuss have this capability to some extent.

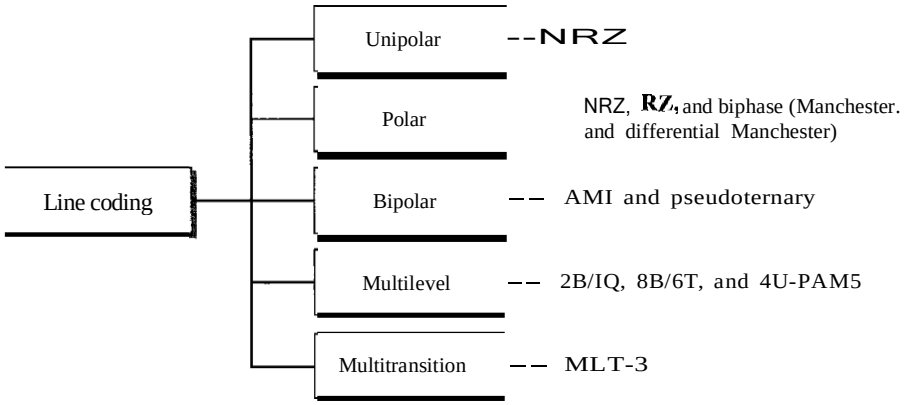
Immunity to Noise and Interference Another desirable code characteristic is a code that is immune to noise and other interferences. Some encoding schemes that we will discuss have this capability.

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 4.4.

Figure 4.4 *Line coding schemes*



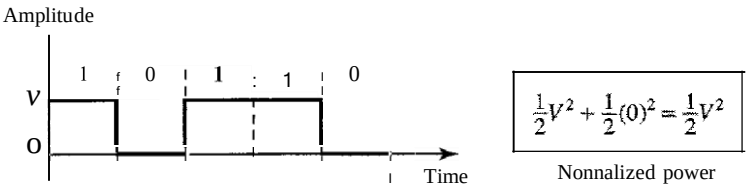
There are several schemes in each category. We need to be familiar with all schemes discussed in this section to understand the rest of the book. This section can be used as a reference for schemes encountered later.

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. It is called NRZ because the signal does not return to zero at the middle of the bit. Figure 4.5 show a unipolar NRZ scheme.

Figure 4.5 *Unipolar NRZ scheme*



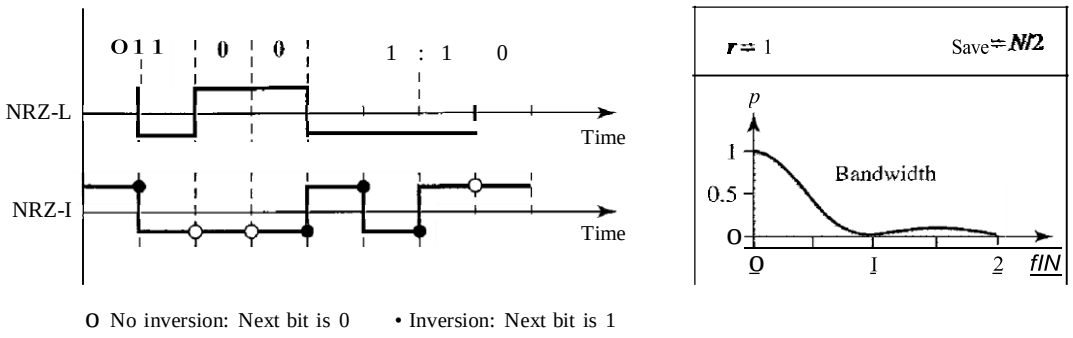
Compared with its polar counterpart (see the next section), this scheme is very costly. As we will see shortly, the normalized power (power needed to send 1 bit per unit line resistance) is double that for polar NRZ. For this reason, this scheme is nor• mally not used in data communications today.

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 band• width. 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.

Figure 4.6 *Polar NRZ-L and NRZ-I schemes*



O No inversion: Next bit is 0 • Inversion: Next 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 Os or Is 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 Is and all Is interpreted as as. NRZ-I does not have this problem. Both schemes have an average signal rate of *NI2* Bd.

NRZ-L and NRZ-J both have an average signal rate of *NI2* 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 frequen• cies between a and *NII.* 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.

Example 4.4

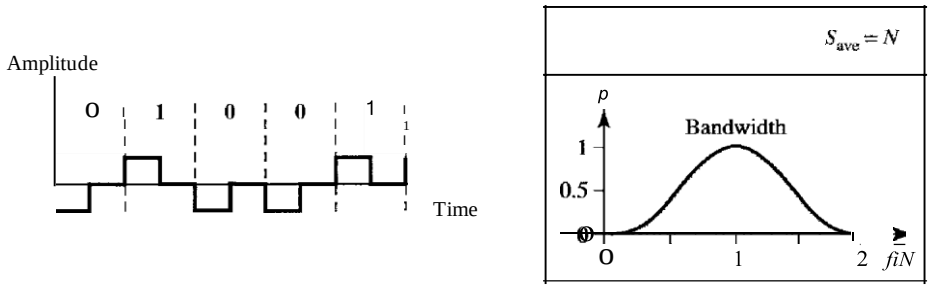
A system is using NRZ-I to transfer 10-Mbps data. What are the average signal rate and mini• mum bandwidth?

Solution

The average signal rate is S = *NI2* = 500 kbaud. The minimum bandwidth for this average baud rate is B_{min} = S = 500 kHz.

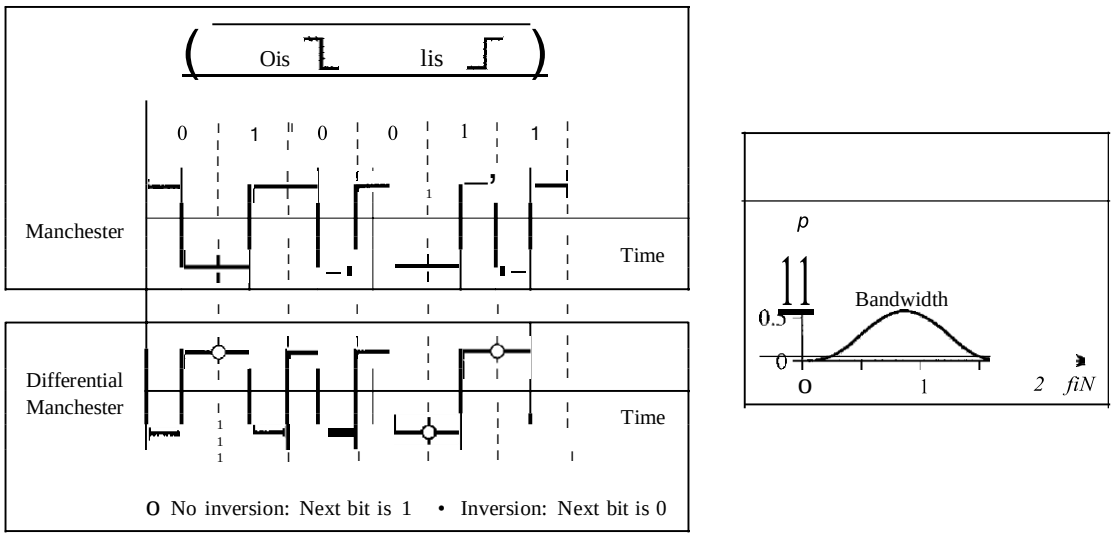
Return to Zero (RZ) The main problem with NRZ encoding occurs when the sender and receiver clocks are not synchronized. The receiver does not know when one bit has ended and the next bit is starting. One solution is the return-to-zero (RZ) scheme, which uses three values: positive, negative, and zero. 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 mid•dle of each bit. It remains there until the beginning of the next bit. The main disadvan•tage 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).

Figure 4.7 *Polar RZ scheme*



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.

Figure 4.8 *Polar biphase: Manchester and differential Manchester schemes*



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 of the bit and maybe one transition at the end of each 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.

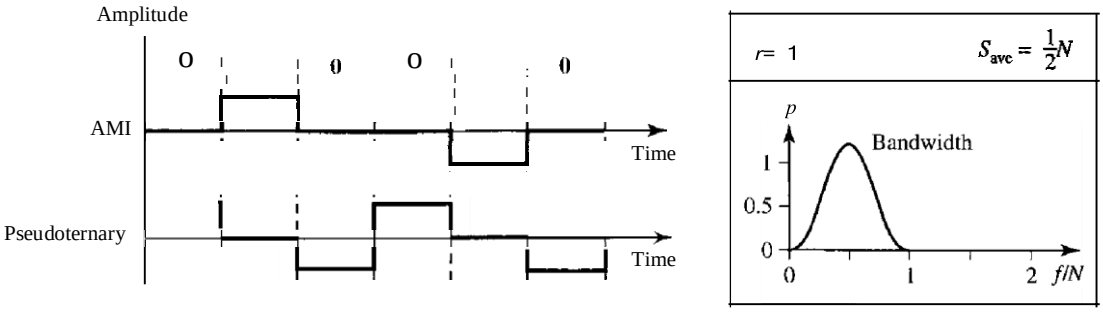
Bipolar Schemes

In bipolar encoding (sometimes called *multilevel binary),* there are three voltage lev• els: 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 of bipolar encoding: AMI and pseudoternary. A common bipolar encoding scheme 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 volt• age represents binary O. Binary Is are represented by alternating positive and negative voltages. A variation of AMI 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.

Figure 4.9 *Bipolar schemes: AMI and pseudoternary*



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 concen• tration of the 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 intu• itively. If we 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 synchroniza• tion problem when a long sequence of Os 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 (Os and Is), which means that a group of *m* data elements can produce a combination of 2^m 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 L^n combinations of signal patterns. If $2^m = L^n$, then each data pattern is encoded into one signal pattern. If $2^m < L^n$, data patterns occupy only a subset of signal patterns. The subset can be carefully designed to prevent baseline wandering, to pro• vide synchronization, and to detect errors that occurred during data transmission. Data encoding is not possible if $2^m > L^n$ 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.

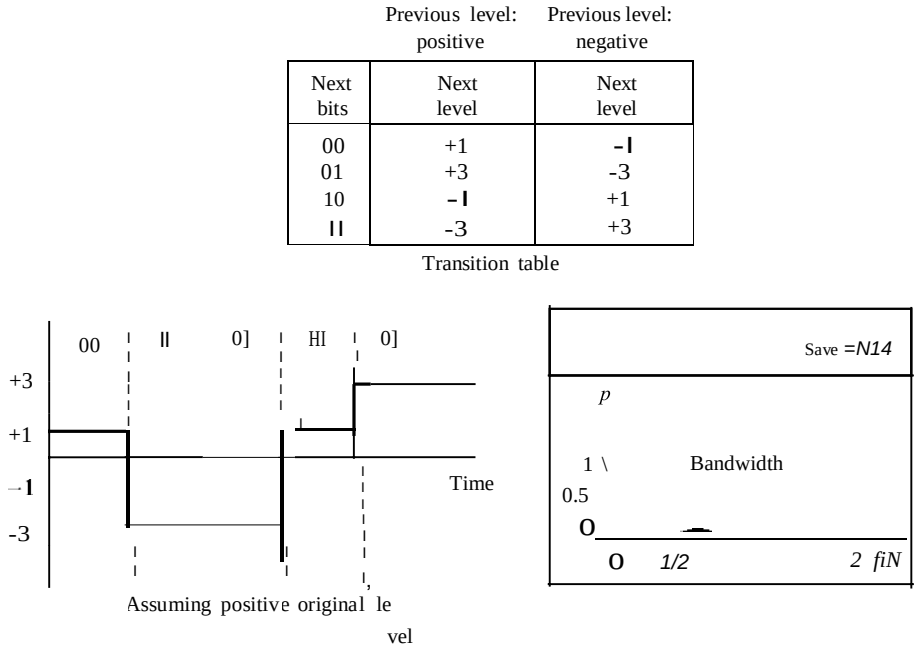
In *mBnL* schemes, a pattern of *m* data elements is encoded as a pattern of *n* signal elements in which $2^m \leq L^n$.

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 of encoding $m = 2$, $n = 1$, and $L = 4$ (quatemary). Fig• ure 4.10 shows an example of a 2B 1Q 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 sig• nal 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 $2^2 = 4^1$.

As we will see in Chapter 9, 2BIQ is used in DSL (Digital Subscriber Line) tech• nology to provide a high-speed connection to the Internet by using subscriber telephone lines.

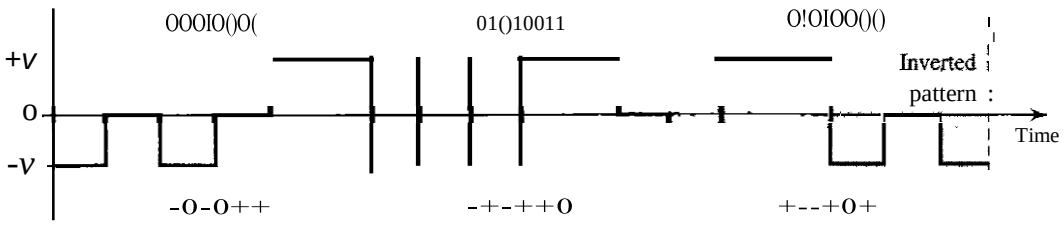
Figure 4.10 *Multilevel: 2B1Q scheme*



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 $2^8 = 256$ different data patterns and $3^6 = 478$ 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 of the redundancy is also used to provide DC balance. Each signal pattern has a weight of 0 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 010 10011 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.

Figure 4.11 *Multilevel: 8B6T scheme*

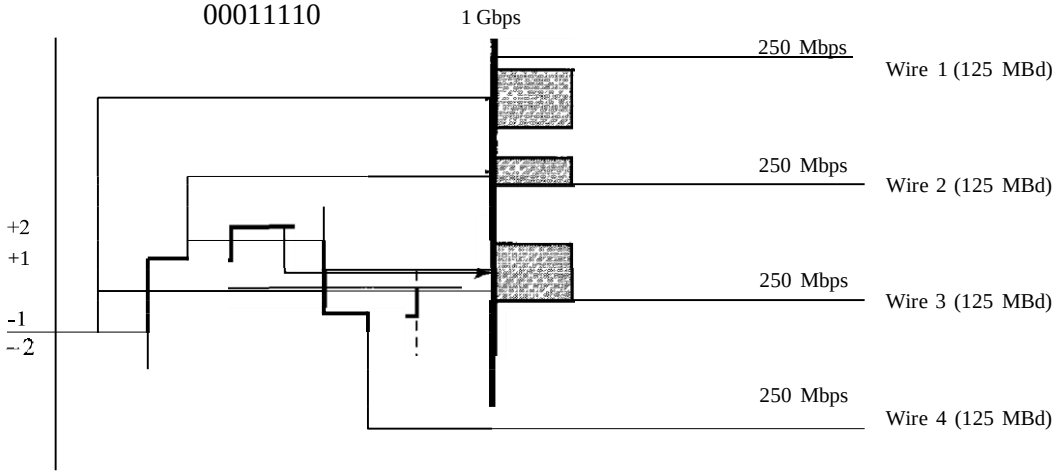


The average signal rate of the scheme is theoretically Save $= \frac{1}{2} \times N \times \frac{6}{8}$; in practice the minimum bandwidth is very close to *6N18.*

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 Chap• ter 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 of four differ•ent levels. The worst signal rate for this imaginary one-dimensional version is *N* x *4/8,* or *N12.*

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 2^8 data patterns are matched to $4^4 = 256$ signal patterns. The extra signal patterns can be used for other purposes such as error detection.

Figure 4.12 *Multilevel: 4D-PAM5 scheme*



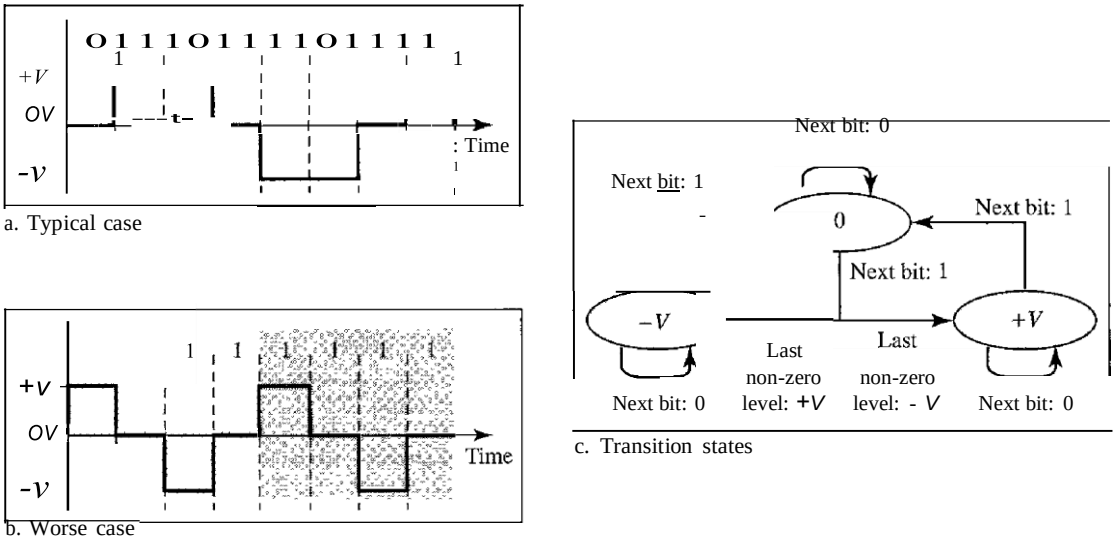
Multiline Transmission: MLT-3

NRZ-I and differential Manchester are classified as differential encoding but use two transi• tion rules to encode binary data (no inversion, inversion). If we 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 cutTent level is 0, the next level is the opposite of the last nonzero level.

The behavior 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 of an MLT-3 signal.

Figure 4.13 *Multitransition: MLT-3 scheme*



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 thebit duration. This worst-case situation can be simulated as an analog signal with a fre• quency 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 ofline coding schemes*

Category	*Scheme*	*Bandwidth (average)*	*Characteristics*
Unipolar	NRZ	*B=N/2*	Costly, no self-synchronization iflong Os or Is, DC
Unipolar	NRZ-L	*B=N/2*	No self-synchronization if long Os or 1s, DC
	NRZ-I	*B=N/2*	No self-synchronization for long aS, DC
	Biphase	*B=N*	Self-synchronization, no DC, high bandwidth

Table 4.1 *Summary ofline coding schemes (continued)*

Category	*Scheme*	*Bandwidth (average)*	*Characteristics*
Bipolar	AMI	*B=NI2*	No self-synchronization for long OS, 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 Os

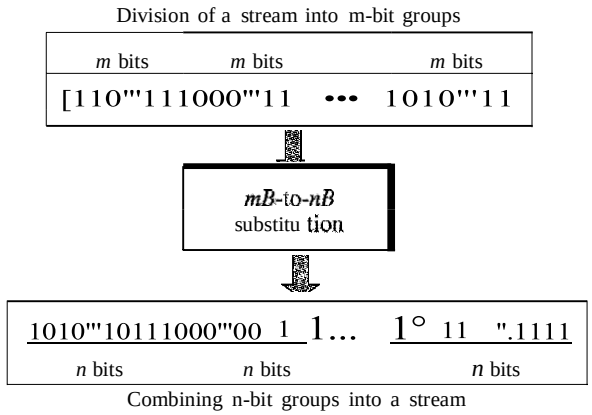
Block Coding

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 perfor• mance of line coding. In general, block coding changes a block of *m* bits into a blockof *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 cod• ing 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.

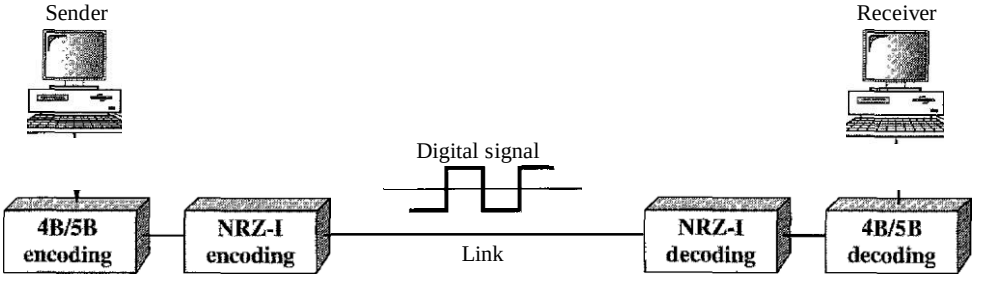
Figure 4.14 *Block coding concept*



4B/5B

The four binary/five binary (4B/5B) coding scheme was designed to be used in com• bination 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.

Figure 4.15 *Using block coding 4B/5B with NRZ-I line coding scheme*



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 corre• sponding 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 of 5 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 detec• tion. If a 5-bit group arrives that belongs to the unused portion of the table, the receiver knows that there is an error in the transmission.

Table 4.2 *4B/5B mapping codes*

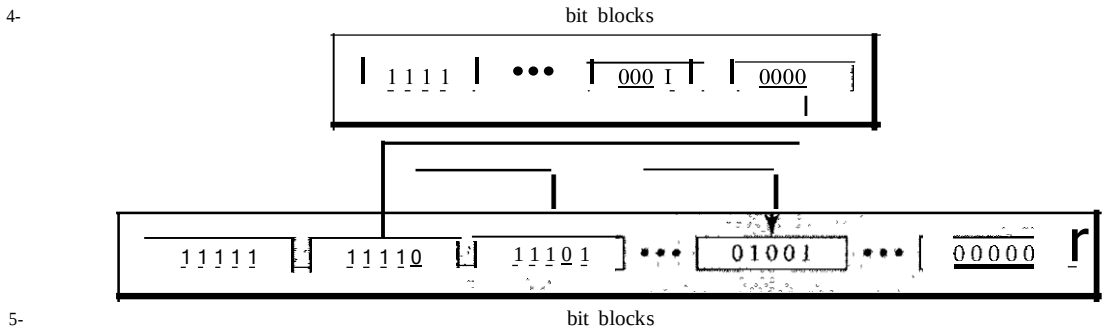
Data Sequence	*Encoded Sequence*	*Control Sequence*	*Encoded Sequence*
0000	11110	Q (Quiet)	00000
0001	01001	I (Idle)	11111
0010	10100	H (Halt)	00100
0011	10101	J (Start delimiter)	11000
0100	01010	K (Start delimiter)	10001
0101	01011	T (End delimiter)	01101

Table 4.2 *4B/5B mapping codes (continued)*

Data Sequence	*Encoded Sequence*	*Control Sequence*	*Encoded Sequence*
0110	01110	S (Set)	11001
0111	01111	R (Reset)	00111
1000	10010		
1001	10011		
1010	10110		
1011	10111		
1100	11 010		
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 of NRZ-1. However, we need to remember that it increases the signal rate of NRZ-1. The redun• dant bits add 20 percent more baud. Still, the result is less than the biphase scheme which has a signal rate of 2 times that of NRZ-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.

Figure 4.16 *Substitution in 48/5B block coding*



Example 4.5

We need to send data at a 1-Mbps rate. What is the minimum required bandwidth, using a combi• nation of *4B/5B* and NRZ-I or Manchester coding?

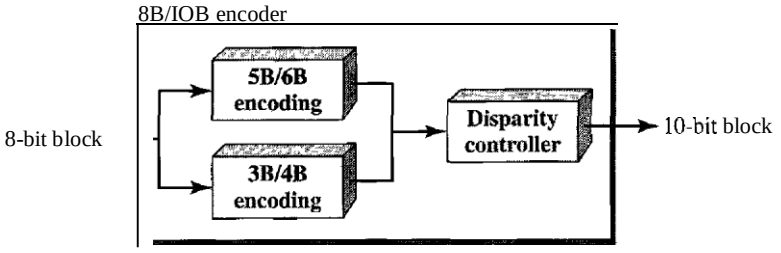
Solution

First *4B/5B* block coding increases the bit rate to 1.25 Mbps. The minimum bandwidth using NRZ-I is *NI2* or 625 kHz. The Manchester scheme needs a minimum bandwidth of 1 MHz. The first choice needs a lower bandwidth, but has a DC component problem; the second choice needsa higher bandwidth, but does not have a DC component problem.

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 combina•tion of 5B/6B and 3B/4B encoding, as shown in Figure 4.17.

Figure 4.17 *8B/lOB block encoding*



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 Os or Is, the code uses a disparity controller which keeps track of excess Os over Is (or Is over Os). If the bits in the cur• rent block create a disparity that contributes to the previous disparity (either direction), then each bit in the code is complemented (a 0 is changed to a 1 and a 1 is changed to a 0).

The coding has $2^{10} - 2^8 = 768$ 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.

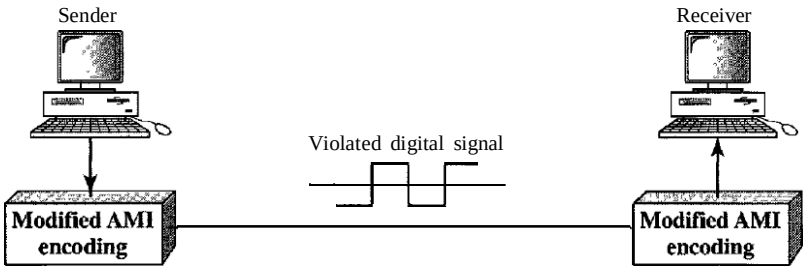
Scrambling

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 Os upsets the synchronization. If we can find a way to avoid a long sequence of Os 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 synchroniza• tion. We are looking for a solution that substitutes long zero-level pulses with a combi• nation 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

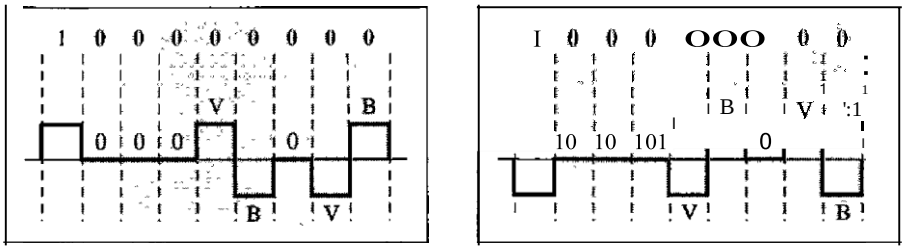
Bipolar with S-zero substitution (BSZS) is commonly used in North America. In this technique, eight consecutive zero-level voltages are replaced by the sequence

Figure 4.18 *AMI used with scrambling*



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.

Figure 4.19 *Two cases ofB8ZS scrambling technique*



a. Previous level is positive. b. Previous level is negative.

Note that the scrambling in this case does not change the bit rate. Also, the tech• nique balances the positive and negative voltage levels (two positives and two nega• tives), which means that the DC balance is maintained. Note that the substitution may change the polarity of a 1 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 of the previous nonzero pulse; B means the polarity opposite to the polarity of the previous nonzero pulse.

HDB3

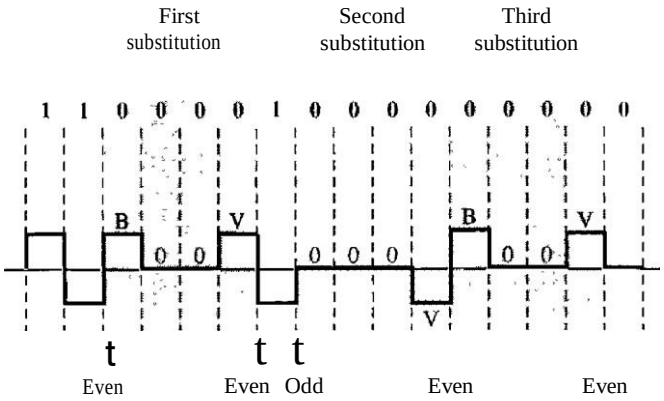
High-density bipolar 3-zero (HDB3) is commonly used outside of North America. In this technique, which is more conservative than B8ZS, four consecutive zero-level voltages are replaced with a sequence of OOOV or **B00V.** The reason for two different substitutions is to

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.

Figure 4.20 shows an example.

Figure 4.20 *Different situations in HDB3 scrambling technique*



There are several points we need to mention here. First, before the first substitu• tion, 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.

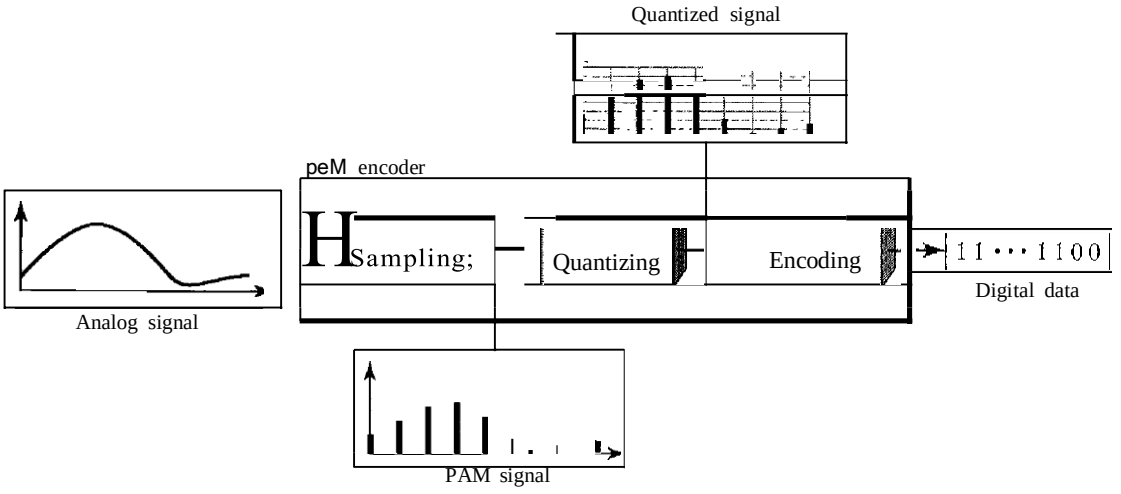
4.2 ANALOG-TO-DIGITAL CONVERSION

The techniques described in Section 4.1 convert digital data to digital signals. Some• times, however, we have an analog signal such as one created by a microphone or cam• era. 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 con• vert 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.

Figure 4.21 *Components of PCM encoder*



1. The analog signal is sampled.
2. The sampled signal is quantized.
3. The quantized values are encoded as streams of bits.

Sampling

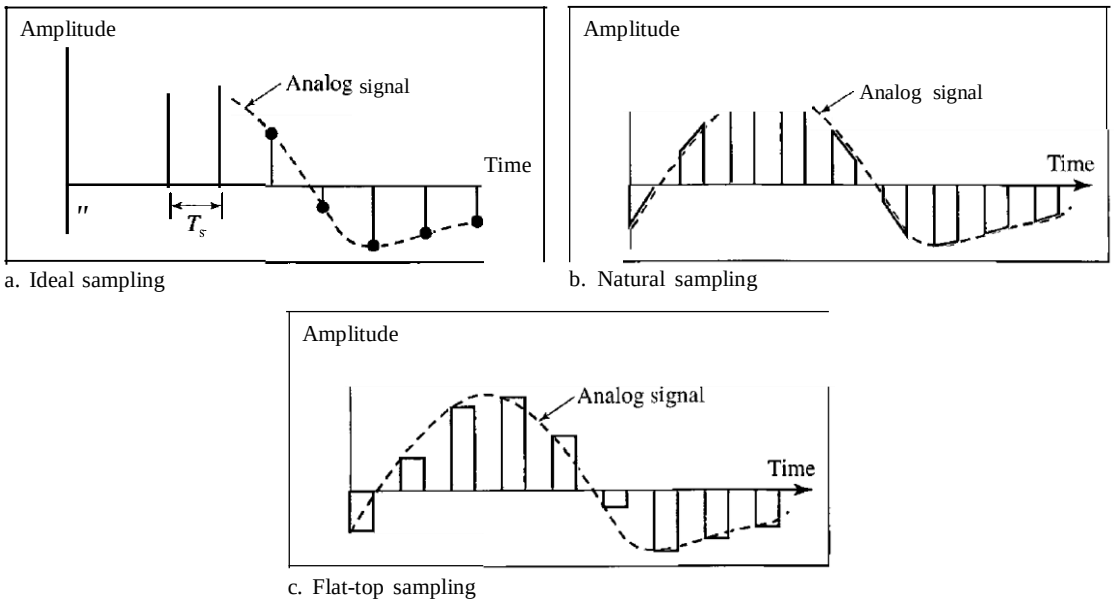
The first step in PCM is sampling. The analog signal is sampled every T_S s, where T_S is the sample interval or period. The inverse of the sampling interval is called the sam• pling rate or sampling frequency and denoted by is, where $is = IIT_S'$ There are three sampling methods-ideal, natural, and flat-top-as shown in Figure 4.22.

In ideal sampling, pulses from the analog signal are sampled. This is an ideal sam• pling 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 T_S? 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.

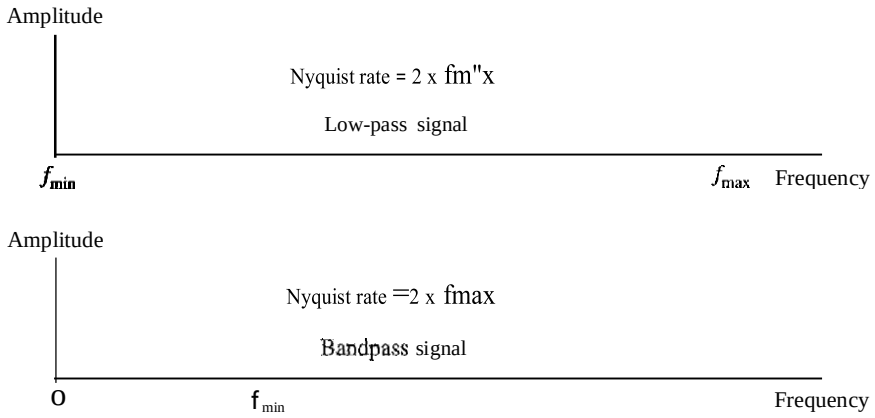
Figure 4.22 *Three different sampling methods for PCM*



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 fre• quency, 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.

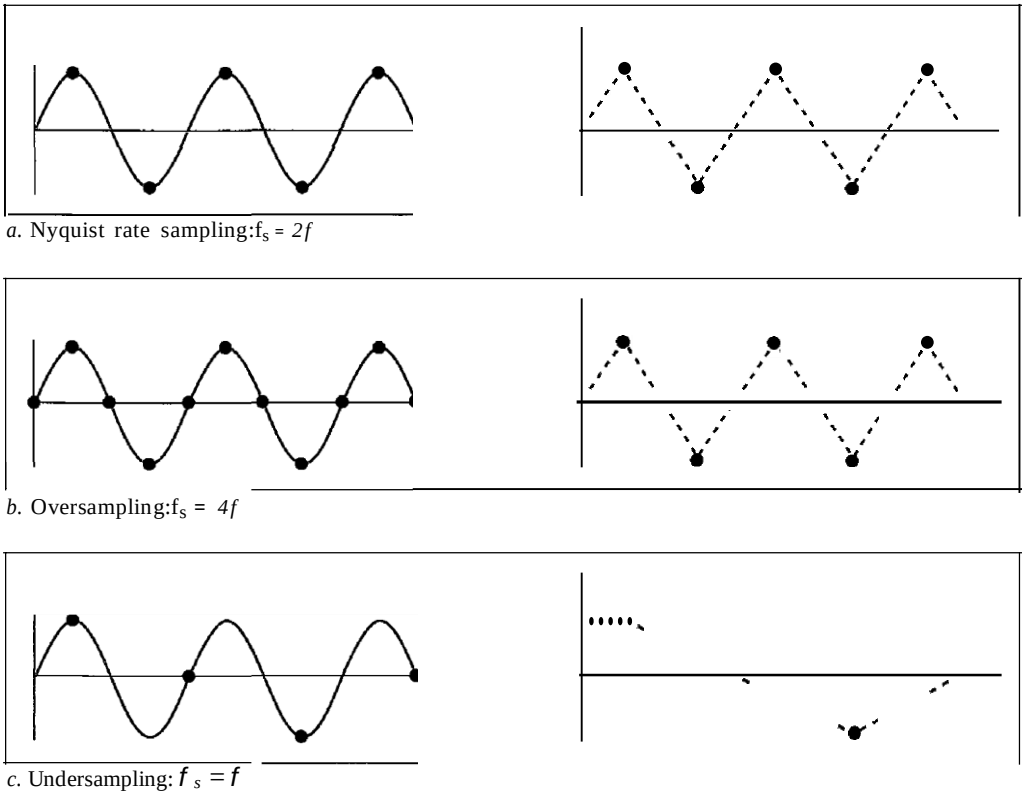
Figure 4.23 *Nyquist sampling rate for low-pass and bandpass signals*



Example 4.6

For an intuitive example of the Nyquist theorem, let us sample a simple sine wave at three sam• pling rates: $f_s = 4f$ (2 times the Nyquist rate $)'/s = 2f$ (Nyquist rate), and $f_S = f$ (one-half the Nyquist rate). Figure 4.24 shows the sampling and the subsequent recovery of the signal.

Figure 4.24 *Recovery ofa sampled sine wave for different sampling rates*



a. Nyquist rate sampling: $f_s = 2f$

b. Oversampling: $f_s = 4f$

c. Undersampling: $f_s = f$

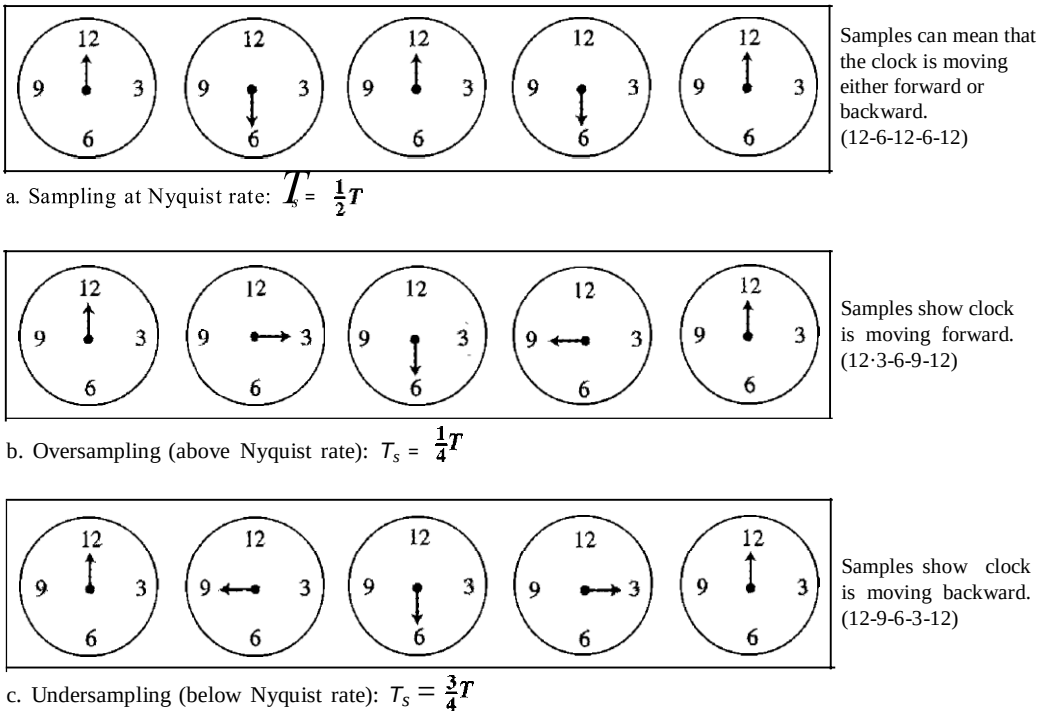
It can be seen that sampling at the Nyquist rate can create a good approximation of the orig• inal sine wave (part a). Oversampling in part b can also create the same approximation, but it is redundant and unnecessary. Sampling below the Nyquist rate (part c) does not produce a signal that looks like the original sine wave.

Example 4.7

As an interesting example, let us see what happens if we sample a periodic event such as the revolu• tion of a hand of a clock. The second hand of a clock has a period of 60 s. According to the Nyquist theorem, we need to sample the hand (take and send a picture) every 30 s ($T_S = \frac{1}{2}$ *Tor* $f_S = 2f$). In Figure 4.25a, the sample points, in order, are 12, 6, 12, 6, 12, and 6. The receiver of the samples cannot tell if the clock is moving forward or backward. In part b, we sample at double the Nyquist rate (every 15 s). The sample points, in order, are 12,3,6, 9, and 12. The clock is moving forward.

In part c, we sample below the Nyquist rate ($T_s = \frac{3}{4}T$ or $f_s = \frac{4}{3}f$). The sample points, in order, are 12, 9,6,3, and 12. Although the clock is moving forward, the receiver thinks that the clock is moving backward.

Figure 4.25 *Sampling ofa clock with only one hand*



Example 4.8

An example related to Example 4.7 is the seemingly backward rotation of the wheels of a forward• moving car in a movie. This can be explained by undersampling. A movie is filmed at 24 frames per second. If a wheel is rotating more than 12 times per second, the undersampling creates the impression of a backward rotation.

Example 4.9

Telephone companies digitize voice by assuming a maximum frequency of 4000 Hz. The sampling rate therefore is 8000 samples per second.

Example 4.10

A complex low-pass signal has a bandwidth of 200 kHz. What is the minimum sampling rate for this signal?

Solution

The bandwidth of a low-pass signal is between 0 andj, where f is the maximum frequency in the signal. Therefore, we can sample this signal at 2 times the highest frequency (200 kHz). The sam• pling rate is therefore 400,000 samples per second.

Example 4.1J

A complex bandpass signal has a bandwidth of 200 kHz. What is the minimum sampling rate for this signal?

Solution

We cannot find the minimum sampling rate in this case because we do not know where the band• width starts or ends. We do not know the maximum frequency in the signal.

Quantization

The result of sampling is a series of pulses with amplitude values between the maxi• mum and minimum amplitudes of the signal. The set of amplitudes can be infinite with nonintegral values between the two limits. These values cannot be used in the encoding process. The following are the steps in quantization:

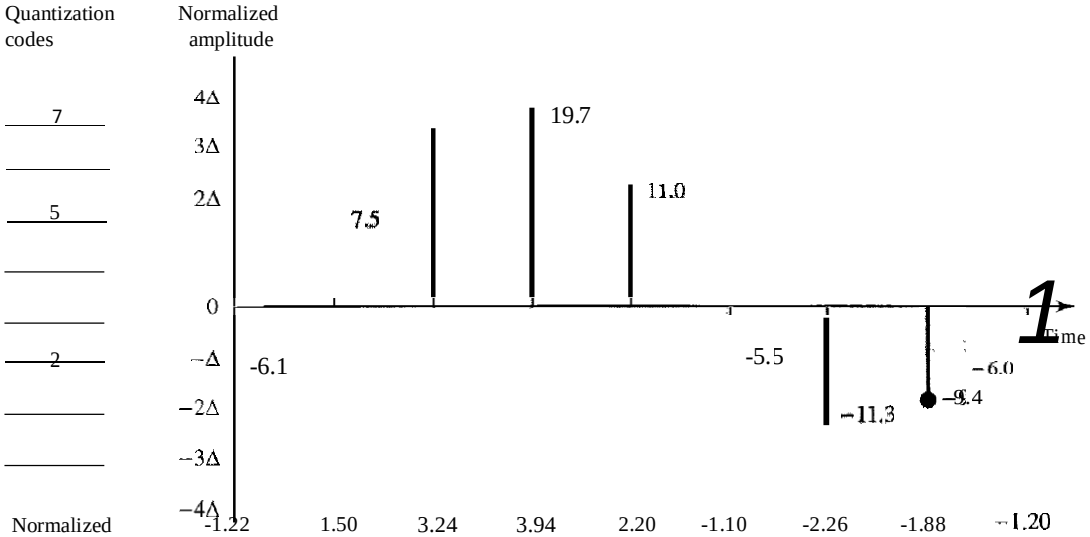
1. We assume that the original analog signal has instantaneous amplitudes between V_{min} and V_{max}.
2. We divide the range into L zones, each of height Δ (delta).

$$\Delta = \frac{V_{max} - V_{min}}{L}$$

3. We assign quantized values of 0 to L - I to the midpoint of each zone.
4. We approximate the value of the sample amplitude to the quantized values.

As a simple example, assume that we have a sampled signal and the sample ampli• tudes are between -20 and +20 V. We decide to have eight levels (L = 8). This means that Δ = 5 V. Figure 4.26 shows this example.

Figure 4.26 *Quantization and encoding ofa sampled signal*



Normalized PAM values	-1.22	1.50	3.24	3.94	2.20	-1.10	-2.26	-1.88	-1.20
Normalized quantized values	-1.50	1.50	3.50	3.50	2.50	-1.50	-2.50	-1.50	-1.50
Normalized error	-0.38	0	+0.26	-0.44	+0.30	-0.40	-0.24	+0.38	-0.30
Quantization code	2	5	7	7	6	2		2	2
Encoded words	010	101	111	111	110	010	001	010	OW

We have shown only nine samples using ideal sampling (for simplicity). The value at the top of each sample in the graph shows the actual amplitude. In the chart, the first row is the normalized value for each sample (actual amplitude/Δ). The quantization process selects the quantization value from the middle of each zone. This means that the normalized quantized values (second row) are different from the normalized amplitudes. The difference is called the *normalized error* (third row). The fourth row is the quantization code for each sample based on the quantization levelsat the left of the graph. The encoded words (fifth row) are the final products of the conversion.

Quantization Levels In the previous example, we showed eight quantization levels. The choice of L, the number of levels, depends on the range of the amplitudes of the analog signal and how accurately we need to recover the signal. If the amplitude of a signal fluctuates between two values only, we need only two levels; if the signal, like voice, has many amplitude values, we need more quantization levels. In audio digitizing, L is normally chosen to be 256; in video it is normally thousands. Choosinglower values of L increases the quantization error if there is a lot of fluctuation in the signal.

Quantization Error One important issue is the error created in the quantization process. (Later, we will see how this affects high-speed modems.) Quantization is an approximation process. The input values to the quantizer are the real values; the output values are the approximated values. The output values are chosen to be the middle value in the zone. If the input value is also at the middle of the zone, there is no quantization error; otherwise, there is an error. In the previous example, the normalized amplitude of the third sample is 3.24, but the normalized quantized value is 3.50. This means that there is an error of +0.26. The value of the error for any sample is less than $\Delta/2$. In other words, we have $-\Delta/2 \leq \text{error} \leq \Delta/2$.

The quantization error changes the signal-to-noise ratio of the signal, which in turn reduces the upper limit capacity according to Shannon.

It can be proven that the contribution of the quantization error to the SNR_{dB} of the signal depends on the number of quantization levels L, or the bits per sample nb, as shown in the following formula:

$$SNR_{dB} = 6.02nb + 1.76 \text{ dB}$$

Example 4.12

What is the SNR_{dB} in the example of Figure 4.26?

Solution

We can use the formula to find the quantization. We have eight levels and 3 bits per sample, so $SNR_{dB} = 6.02(3) + 1.76 = 19.82$ dB. Increasing the number of levels increases the SNR.

Example 4.13

A telephone subscriber line must have an SNR_{dB} above 40. What is the minimum number of bits per sample?

Solution

We can calculate the number of bits as

$$\text{SNR}_{\text{dB}} = 6.02nb + 1.76 \Rightarrow 40 \rightarrow n \Rightarrow 6.35$$

Telephone companies usually assign 7 or 8 bits per sample.

Uniform Versus Nonuniform Quantization For many applications, the distribu• tion of the instantaneous amplitudes in the analog signal is not uniform. Changes in amplitude often occur more frequently in the lower amplitudes than in the higher ones. For these types of applications it is better to use nonuniform zones. In other words, the height of is not fixed; it is greater near the lower amplitudes and less near the higher amplitudes. Nonuniform quantization can also be achieved by using aprocess called companding and expanding. The signal is companded at the sender before conversion; it is expanded at the receiver after conversion. Companding meansreducing the instantaneous voltage amplitude for large values; expanding is the oppo• site process. Companding gives greater weight to strong signals and less weight to weak ones. It has been proved that nonuniform quantization effectively reduces the SNR_{dB} of quantization.

Encoding

The last step in PCM is encoding. After each sample is quantized and the number of bits per sample is decided, each sample can be changed to an llb-bit code word. In Fig•ure 4.26 the encoded words are shown in the last row. A quantization code of 2 is encoded as 010; 5 is encoded as 101; and so on. Note that the number of bits for each sample is determined from the number of quantization levels. If the number of quanti• zation levels is L, the number of bits is $llb = \log_2 L$. In our example L is 8 and llb is therefore 3. The bit rate can be found from the formula

$$\text{Bit rate} = \text{sampling rate} \times \text{number of bits per sample} = f_s \times nb$$

Example 4.14

We want to digitize the human voice. What is the bit rate, assuming 8 bits per sample?

Solution

The human voice normally contains frequencies from 0 to 4000 Hz. So the sampling rate and bit rate are calculated as follows:

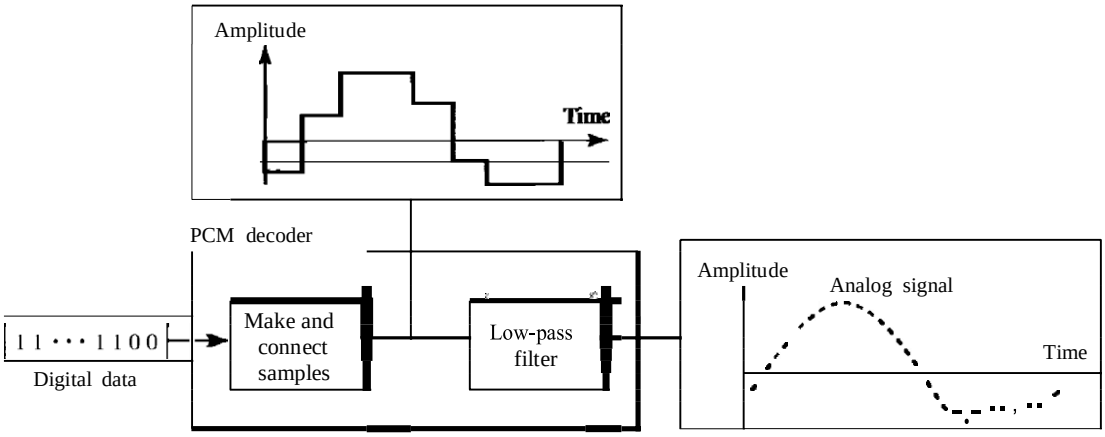
$$\text{Sampling rate} = 4000 \times 2 = 8000 \text{ samples/s}$$
$$\text{Bit rate} = 8000 \times 8 = 64{,}000 \text{ bps} = 64 \text{ kbps}$$

Original Signal Recovery

The recovery of the original signal requires the PCM decoder. The decoder first uses circuitry to convert the code words into a pulse that holds the amplitude until the next pulse. After the staircase signal is completed, it is passed through a low-pass filter to

smooth the staircase signal into an analog signal. The filter has the same cutoff fre• quency as the original signal at the sender. If the signal has been sampled at (or greater than) the Nyquist sampling rate and if there are enough quantization levels, the original signal will be recreated. Note that the maximum and minimum values of the original signal can be achieved by using amplification. Figure 4.27 shows the simplified process.

Figure 4.27 *Components ofa PCM decoder*



PCM Bandwidth

Suppose we are given the bandwidth of a low-pass analog signal. If we then digitize the signal, what is the new minimum bandwidth of the channel that can pass this digitized signal? We have said that the minimum bandwidth of a line-encoded signal is B_{min} =ex N x *(lIr).* We substitute the value of N in this formula:

$$B_{min} = c \times N \times \frac{1}{r} = c \times nb \times f_s \times \frac{1}{r} = c \times nb \times 2 \times B_{analog} \times \frac{1}{r}$$

When *lIr* = I (for a NRZ or bipolar signal) and c = (12) (the average situation), the minimum bandwidth is

$$B_{min} = n_b \times B_{analog}$$

This means the minimum bandwidth of the digital signal is *nb* times greater than the bandwidth of the analog signal. This is the price we pay for digitization.

Example 4.15

We have a low-pass analog signal of 4 kHz. If we send the analog signal, we need a channel with a minimum bandwidth of 4 kHz. If we digitize the signal and send 8 bits per sample, we need a channel with a minimum bandwidth of 8 x 4 kHz =32 kHz.

Maximum Data Rate ofa Channel

In Chapter 3, we discussed the Nyquist theorem which gives the data rate of a channel as $N_{max} = 2 \times B \times \log_2 L$. We can deduce this rate from the Nyquist sampling theorem by using the following arguments.

1. We assume that the available channel is low-pass with bandwidth *B.*
2. We assume that the digital signal we want to send has *L* levels, where each level is a signal element. This means $r = 1/10g_2 L$.
3. We first pass the digital signal through a low-pass filter to cut off the frequencies above *B* Hz.
4. We treat the resulting signal as an analog signal and sample it at 2 x *B* samples per second and quantize it using *L* levels. Additional quantization levels are useless because the signal originally had *L* levels.
5. The resulting bit rate is $N = fs \times nb = 2 \times B \times \log2 L$. This is the maximum band• width; if the case factor *c* increases, the data rate is reduced.

$$N_{max} = 2 \times B \times \log_2 L \quad \text{bps}$$

Minimum Required Bandwidth

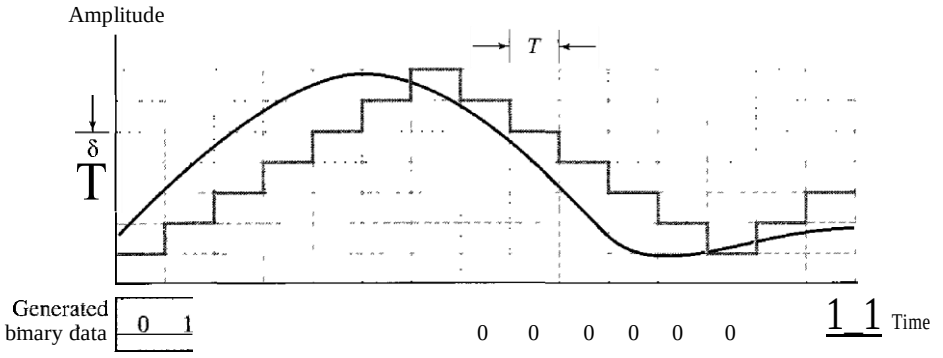
The previous argument can give us the minimum bandwidth if the data rate and the number of signal levels are fixed. We can say

$$B_{mm} = \frac{N}{2 \times \log_2 L} \quad \text{Hz}$$

Delta Modulation (DM)

PCM is a very complex technique. Other techniques have been developed to reduce the complexity of PCM. The simplest is *delta modulation.* PCM finds the value of the signal amplitude for each sample; DM finds the change from the previous sample. Fig• ure 4.28 shows the process. Note that there are no code words here; bits are sent one after another.

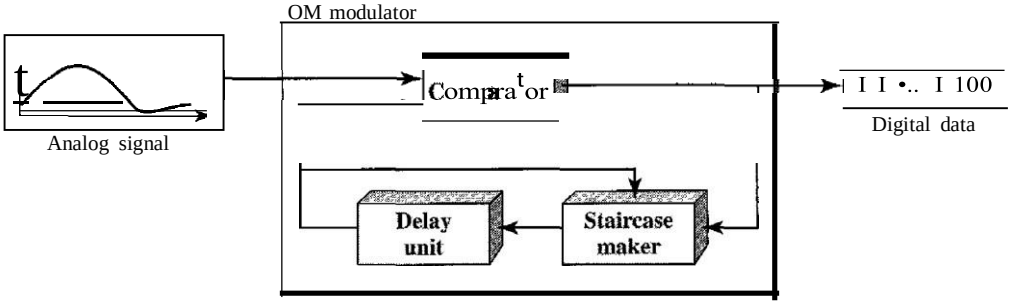
Figure 4.28 *The process ofdelta modulation*



Modulator

The modulator is used at the sender site to create a stream of bits from an analog signal.The process records the small positive or negative changes, called delta δ. If the delta is positive, the process records a I; if it is negative, the process records a O. However, the process needs a base against which the analog signal is compared. The modulator builds a second signal that resembles a staircase. Finding the change is then reduced to comparing the input signal with the gradually made staircase signal. Figure 4.29 shows a diagram of the process.

Figure 4.29 *Delta modulation components*

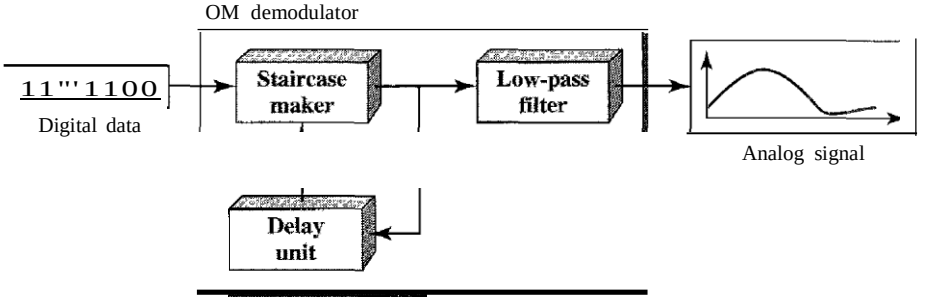


The modulator, at each sampling interval, compares the value of the analog signal with the last value of the staircase signal. If the amplitude of the analog signal is larger, the next bit in the digital data is 1; otherwise, it is O. The output of the comparator, how• ever, also makes the staircase itself. If the next bit is I, the staircase maker moves the last point of the staircase signal δ up; it the next bit is 0, it moves it δ down. Note that we need a delay unit to hold the staircase function for a period between two comparisons.

Demodulator

The demodulator takes the digital data and, using the staircase maker and the delay unit, creates the analog signal. The created analog signal, however, needs to pass through a low-pass filter for smoothing. Figure 4.30 shows the schematic diagram.

Figure 4.30 *Delta demodulation components*



Adaptive DM

A better performance can be achieved if the value of δ is not fixed. In adaptive delta modulation, the value of δ changes according to the amplitude of the analog signal.

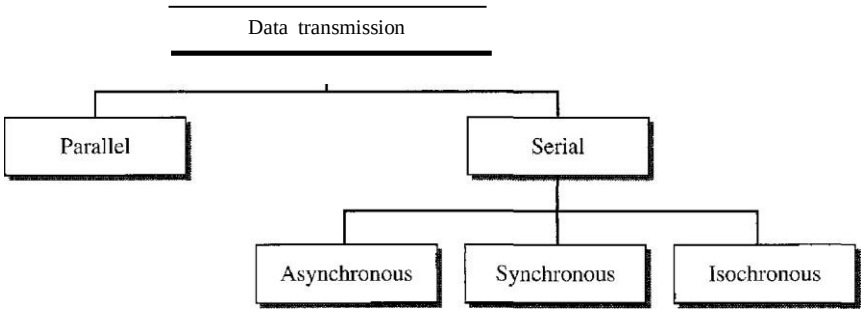
Quantization Error

It is obvious that DM is not perfect. Quantization error is always introduced in the process. The quantization error of DM, however, is much less than that for PCM.

4.3 TRANSMISSION MODES

Of primary concern when we are considering the transmission of data from one device to another is the wiring, and of primary concern when we are considering the wiring is the data stream. Do we send 1 bit at a time; or do we group bits into larger groups and, if so, how? The transmission of binary data across a link can be accomplished in either parallel or serial mode. In parallel mode, multiple bits are sent with each clock tick. In serial mode, 1 bit is sent with each clock tick. While there is only one way to send parallel data, there are three subclasses of serial transmission: asynchronous, synchronous, and isochronous (see Figure 4.31).

Figure 4.31 *Data transmission and modes*



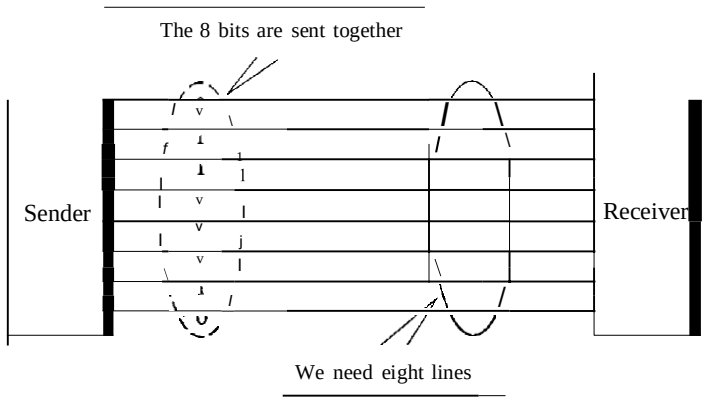
Parallel Transmission

Binary data, consisting of Is and Os, may be organized into groups of *n* bits each. Computers produce and consume data in groups of bits much as we conceive of and use spoken language in the form of words rather than letters. By grouping, we can send data *n* bits at a time instead of 1. This is called parallel transmission.

The mechanism for parallel transmission is a conceptually simple one: Use *n* wires to send *n* bits at one time. That way each bit has its own wire, and all *n* bits of one group can be transmitted with each clock tick from one device to another. Figure 4.32 shows how parallel transmission works for $n = 8$. Typically, the eight wires are bundled in a cable with a connector at each end.

The advantage of parallel transmission is speed. All else being equal, parallel transmission can increase the transfer speed by a factor of *n* over serial transmission.

Figure 4.32 *Parallel transmission*

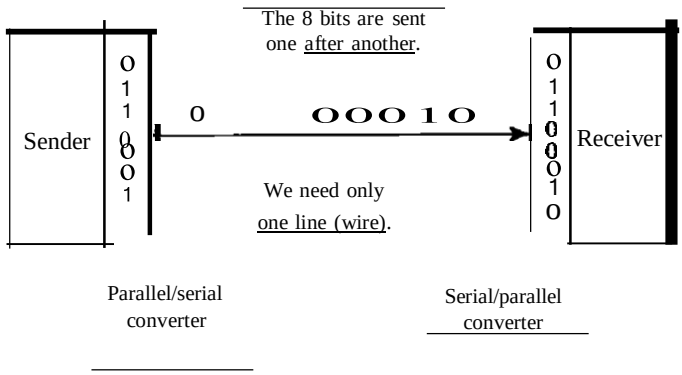


But there is a significant disadvantage: cost. Parallel transmission requires *n* communi• cation lines (wires in the example) just to transmit the data stream. Because this is expensive, parallel transmission is usually limited to short distances.

Serial Transmission

In serial transmission one bit follows another, so we need only one communica• tion channel rather than *n* to transmit data between two communicating devices (see Figure 4.33).

Figure 4.33 *Serial transmission*



The advantage of serial over parallel transmission is that with only one communi• cation channel, serial transmission reduces the cost of transmission over parallel by roughly a factor of *n*.

Since communication within devices is parallel, conversion devices are required at the interface between the sender and the line (parallel-to-serial) and between the line and the receiver (serial-to-parallel).

Serial transmission occurs in one of three ways: asynchronous, synchronous, and isochronous.

Asynchronous Transmission

Asynchronous transmission is so named because the timing of a signal is unimportant. Instead, information is received and translated by agreed upon patterns. As long as those patterns are followed, the receiving device can retrieve the information without regard to the rhythm in which it is sent. Patterns are based on grouping the bit stream into bytes. Each group, usually 8 bits, is sent along the link as a unit. The sending system handles each group independently, relaying it to the link whenever ready, without regard to a timer.

Without synchronization, the receiver cannot use timing to predict when the next group will arrive. To alert the receiver to the arrival of a new group, therefore, an extra bit is added to the beginning of each byte. This bit, usually a 0, is called the start bit.To let the receiver know that the byte is finished, 1 or more additional bits are appended to the end of the byte. These bits, usually I s, are called stop bits. By this method, each byte is increased in size to at least 10 bits, of which 8 bits is information and 2 bits or more are signals to the receiver. In addition, the transmission of each byte may then be followed by a gap of varying duration. This gap can be represented either by an idle channel or by a stream of additional stop bits.

In asynchronous transmission, we send 1 start bit (0) at the beginning and 1 or more stop bits (Is) at the end ofeach byte. There may be a gap between each byte.

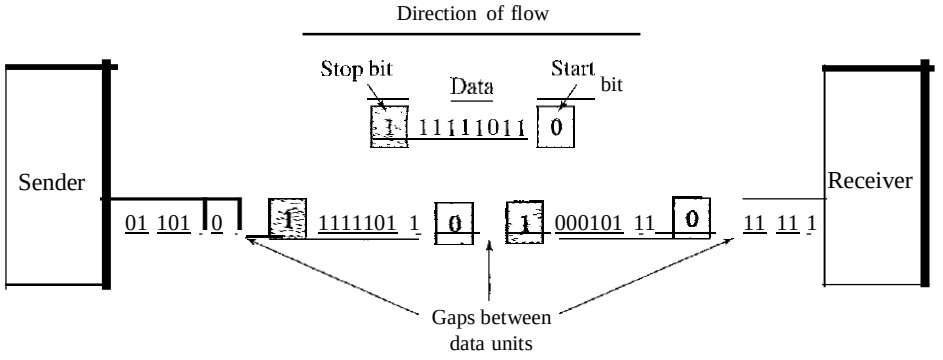
The start and stop bits and the gap alert the receiver to the beginning and end of each byte and allow it to synchronize with the data stream. This mechanism is called *asynchronous* because, at the byte level, the sender and receiver do not have to be synchronized. But within each byte, the receiver must still be synchronized with the incoming bit stream. That is, some synchronization is required, but only for the duration of a single byte. The receiving device resynchronizes at the onset of each new byte. When the receiver detects a start bit, it sets a timer and begins counting bits as they come in. After *n* bits, the receiver looks for a stop bit. As soon as it detects the stop bit, it waits until it detects the next start bit.

Asynchronous here means "asynchronous at the byte level," but the bits are still synchronized; their durations are the same.

Figure 4.34 is a schematic illustration of asynchronous transmission. In this example, the start bits are as, the stop bits are 1s, and the gap is represented by an idle line rather than by additional stop bits.

The addition of stop and start bits and the insertion of gaps into the bit stream make asynchronous transmission slower than forms of transmission that can operate without the addition of control information. But it is cheap and effective, two advantages that make it an attractive choice for situations such as low-speed communication. For example, the connection of a keyboard to a computer is a natural application for asynchronous transmission. A user types only one character at a time, types extremely slowly in data processing terms, and leaves unpredictable gaps of time between each character.

Figure 4.34 *Asynchronous transmission*



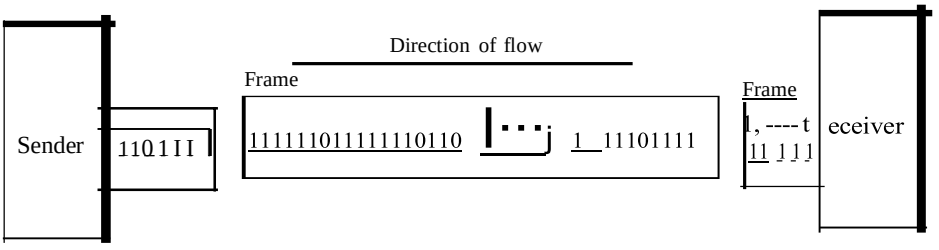
Synchronous Transmission

In synchronous transmission, the bit stream is combined into longer "frames," which may contain multiple bytes. Each byte, however, is introduced onto the transmission link without a gap between it and the next one. It is left to the receiver to separate the bit stream into bytes for decoding purposes. In other words, data are transmitted as an unbroken string of 1s and Os, and the receiver separates that string into the bytes, or characters, it needs to reconstruct the information.

In synchronous transmission, we send bits one after another without start or stop bits or gaps. It is the responsibility of the receiver to group the bits.

Figure 4.35 gives a schematic illustration of synchronous transmission. We have drawn in the divisions between bytes. In reality, those divisions do not exist; the sender puts its data onto the line as one long string. If the sender wishes to send data in separate bursts, the gaps between bursts must be filled with a special sequence of Os and Is that means *idle*. The receiver counts the bits as they arrive and groups them in 8-bit units.

Figure 4.35 *Synchronous transmission*



Without gaps and start and stop bits, there is no built-in mechanism to help the receiving device adjust its bit synchronization midstream. Timing becomes very impor• tant, therefore, because the accuracy of the received information is completely dependent on the ability of the receiving device to keep an accurate count of the bits as they come in.

The advantage of synchronous transmission is speed. With no extra bits or gaps to introduce at the sending end and remove at the receiving end, and, by extension, with fewer bits to move across the link, synchronous transmission is faster than asynchro- nous transmission. For this reason, it is more useful for high-speed applications such as the transmission of data from one computer to another. Byte synchronization is accom• plished in the data link layer.

We need to emphasize one point here. Although there is no gap between charactersin synchronous serial transmission, there may be uneven gaps between frames.

Isochronous

In real-time audio and video, in which uneven delays between frames are not accept• able, synchronous transmission fails. For example, TV images are broadcast at the rate of 30 images per second; they must be viewed at the same rate. If each image is sent by using one or more frames, there should be no delays between frames. For this type of application, synchronization between characters is not enough; the entire stream ofbits must be synchronized. The isochronous transmission guarantees that the data arrive at a fixed rate.

4.4 SUMMARY

- Digital-to-digital conversion involves three techniques: line coding, block coding, and scrambling.
- Line coding is the process of converting digital data to a digital signal.
- We can roughly divide line coding schemes into five broad categories: unipolar, polar, bipolar, multilevel, and multitransition.
- Block coding provides redundancy to ensure synchronization and inherent error detection. Block coding is normally referred to as *mB/nB* coding; it replaces each m- bit group with an n-bit group.
- Scrambling provides synchronization without increasing the number of bits. Two common scrambling techniques are B8ZS and HDB3.
- The most common technique to change an analog signal to digital data (digitiza• tion) is called pulse code modulation (PCM).
- 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 *fs*, where *fs* = *lITs*. There are three sampling methods-ideal, natural, and flat-top.
- 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.

- Other sampling techniques have been developed to reduce the complexity of PCM. The simplest is *delta modulation*. PCM finds the value of the signal amplitude for each sample; DM finds the change from the previous sample.
- While there is only one way to send parallel data, there are three subclasses of serial transmission: asynchronous, synchronous, and isochronous.
- In asynchronous transmission, we send 1 start bit (0) at the beginning and 1 or more stop bits (1 s) at the end of each byte.
- In synchronous transmission, we send bits one after another without start or stop bits or gaps. It is the responsibility of the receiver to group the bits.
- The isochronous mode provides synchronized for the entire stream of bits must. In other words, it guarantees that the data arrive at a fixed rate.

4.5 PRACTICE SET

Review Questions

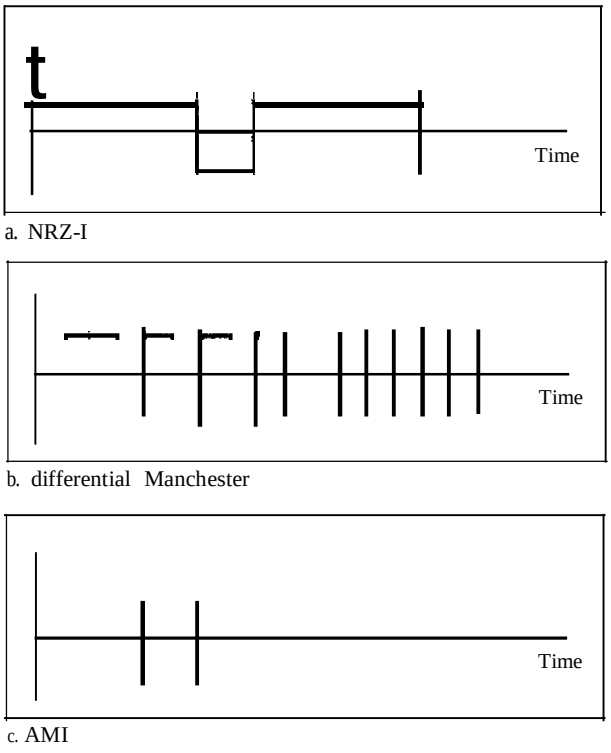
1. List three techniques of digital-to-digital conversion.
2. Distinguish between a signal element and a data element.
3. Distinguish between data rate and signal rate.
4. Define baseline wandering and its effect on digital transmission.
5. Define a DC component and its effect on digital transmission.
6. Define the characteristics of a self-synchronizing signal.
7. List five line coding schemes discussed in this book.
8. Define block coding and give its purpose.
9. Define scrambling and give its purpose.
10. Compare and contrast PCM and DM.
11. What are the differences between parallel and serial transmission?
12. List three different techniques in serial transmission and explain the differences.

Exercises

13. Calculate the value of the signal rate for each case in Figure 4.2 if the data rate is 1 Mbps and c = 1/2.
14. In a digital transmission, the sender clock is 0.2 percent faster than the receiver clock. How many extra bits per second does the sender send if the data rate is 1 Mbps?
15. Draw the graph of the NRZ-L scheme using each of the following data streams, assuming that the last signa11evel has been positive. From the graphs, guess the bandwidth for this scheme using the average number of changes in the signal level. Compare your guess with the corresp.onding entry in Table 4.1.
    a. 00000000
    b. 11111111
    c. 01010101
    d. 00110011

16. Repeat Exercise 15 for the NRZ-I scheme.
17. Repeat Exercise 15 for the Manchester scheme.
18. Repeat Exercise 15 for the differential Manchester scheme.
19. Repeat Exercise 15 for the 2B 1Q scheme, but use the following data streams.
    a. 0000000000000000
    b. 1111111111111111
    c. 0101010101010101
    d. 0011001100110011
20. Repeat Exercise 15 for the MLT-3 scheme, but use the following data streams.
    a. 00000000
    b. 11111111
    c. 01010101
    d. 00011000
21. Find the 8-bit data stream for each case depicted in Figure 4.36.

Figure 4.36 *Exercise 21*



a. NRZ-I

b. differential Manchester

c. AMI

22. An NRZ-I signal has a data rate of 100 Kbps. Using Figure 4.6, calculate the value of the normalized energy (P) for frequencies at 0 Hz, 50 KHz, and 100 KHz.
23. A Manchester signal has a data rate of 100 Kbps. Using Figure 4.8, calculate the value of the normalized energy (P) for frequencies at 0 Hz, 50 KHz, 100 KHz.

24. The input stream to a 4B/5B block encoder is 0100 0000 0000 0000 0000 OOOI. Answer the following questions:
    a. What is the output stream?
    b. What is the length of the longest consecutive sequence of Os in the input?
    c. What is the length of the longest consecutive sequence of Os in the output?
25. How many invalid (unused) code sequences can we have in 5B/6B encoding? How many in 3B/4B encoding?
26. What is the result of scrambling the sequence 11100000000000 using one of the following scrambling techniques? Assume that the last non-zero signal level has been positive.
    a. B8ZS
    b. HDB3 (The number of nonzero pules is odd after the last substitution)
27. What is the Nyquist sampling rate for each of the following signals?
    a. A low-pass signal with bandwidth of 200 KHz?
    b. A band-pass signal with bandwidth of 200 KHz if the lowest frequency is 100 KHz?
28. We have sampled a low-pass signal with a bandwidth of 200 KHz using 1024 levels of quantization.
    a. Calculate the bit rate of the digitized signal.
    b. Calculate the SNR_{dB} for this signal.
    c. Calculate the PCM bandwidth of this signal.
29. What is the maximum data rate of a channel with a bandwidth of 200 KHz if we use four levels of digital signaling.
30. An analog signal has a bandwidth of 20 KHz. If we sample this signal and send it through a 30 Kbps channel what is the SNR_{dB}?
31. We have a baseband channel with a I-MHz bandwidth. What is the data rate for this channel if we use one of the following line coding schemes?
    a. NRZ-L
    b. Manchester
    c. MLT-3
    d. 2B1Q
32. We want to transmit 1000 characters with each character encoded as 8 bits.
    a. Find the number of transmitted bits for synchronous transmission.
    b. Find the number of transmitted bits for asynchronous transmission.
    c. Find the redundancy percent in each case.

CHAPTERS

Analog Transmission

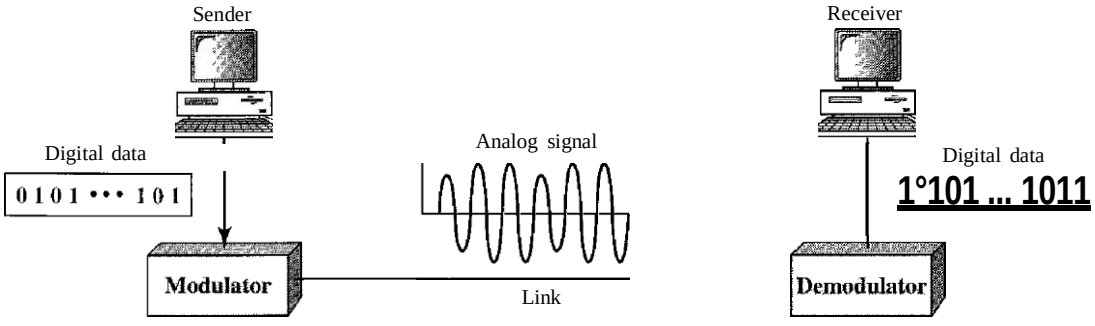
In Chapter 3, we discussed the advantages and disadvantages of digital and analog trans• mission. We saw that while digital transmission is very desirable, a low-pass channel is needed. We also saw that analog transmission is the only choice if we have a bandpass channel. Digital transmission was discussed in Chapter 4; we discuss analog transmis• sion in this chapter.

Converting digital data to a bandpass analog signal.is traditionally called digital• to-analog conversion. Converting a low-pass analog signal to a bandpass analog signal is traditionally called analog-to-analog conversion. In this chapter, we discuss these two types of conversions.

5.1 DIGITAL-TO-ANALOG CONVERSION

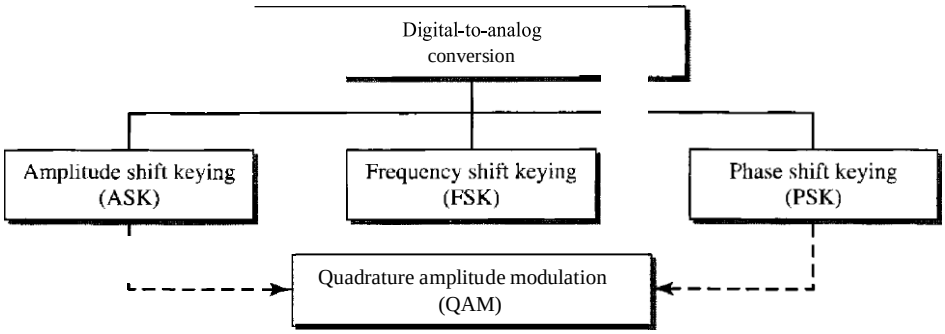
Digital-to-analog conversion is the process of changing one of the characteristics of an analog signal based on the information in digital data. Figure 5.1 shows the rela• tionship between the digital information, the digital-to-analog modulating process, and the resultant analog signal.

Figure 5.1 *Digital-to-analog conversion*



As discussed in Chapter 3, a sine wave is defined by three characteristics: amplitude, frequency, and phase. When we vary anyone of these characteristics, we create a differ• ent version of that wave. So, by changing one characteristic of a simple electric signal, we can use it to represent digital data. Any of the three characteristics can be altered in this way, giving us at least three mechanisms for modulating digital data into an analog signal: amplitude shift keying (ASK), frequency shift keying (FSK), and phase shift keying (PSK). In addition, there is a fourth (and better) mechanism that combines changing both the amplitude and phase, called quadrature amplitude modulation (QAM). QAM is the most efficient of these options and is the mechanism commonly used today (see Figure 5.2).

Figure 5.2 *Types ofdigital-to-analog conversion*



Aspects of Digital-to-Analog Conversion

Before we discuss specific methods of digital-to-analog modulation, two basic issues must be reviewed: bit and baud rates and the carrier signal.

Data Element Versus Signal Element

In Chapter 4, we discussed the concept of the data element versus the signal element. We defined a data element as the smallest piece of information to be exchanged, the bit. We also defined a signal element as the smallest unit of a signal that is constant. Although we continue to use the same terms in this chapter, we will see that the nature of the signal element is a little bit different in analog transmission.

Data Rate Versus Signal Rate

We can define the data rate (bit rate) and the signal rate (baud rate) as we did for digital transmission. The relationship between them is

$$S = N \times \frac{1}{r} \quad \text{baud}$$

where N is the data rate (bps) and r is the number of data elements carried in one signal element. The value of r in analog transmission is $r = \log_2 L$, where L is the type of sig• nal element, not the level. The same nomenclature is used to simplify the comparisons.

Bit rate is the number of bits per second. Baud rate is the number of signal elements per second. In the analog transmission of digital data, the baud rate is less than or equal to the bit rate.

The same analogy we used in Chapter 4 for bit rate and baud rate applies here. In transportation, a baud is analogous to a vehicle, and a bit is analogous to a passenger. We need to maximize the number of people per car to reduce the traffic.

Example 5.1

An analog signal carries 4 bits per signal element. If 1000 signal elements are sent per second, find the bit rate.

Solution

In this case, $r = 4$, $S = 1000$, and N is unknown. We can find the value of N from

$$S = N \times \frac{1}{r} \quad \text{or} \quad N = S \times r = 1000 \times 4 = 4000 \text{ bps}$$

Example 5.2

An analog signal has a bit rate of 8000 bps and a baud rate of 1000 baud. How many data elements are carried by each signal element? How many signal elements do we need?

Solution

In this example, $S = 1000$, $N = 8000$, and rand L are unknown. We find first the value of rand then the value of L.

$$S = N \times \frac{1}{r} \qquad r = \frac{N}{S} = \frac{8000}{1000} = 8 \text{ bltslbaud}$$

$$r = \log_2 L \qquad L = 2^r = 2^8 = 256$$

Bandwidth

The required bandwidth for analog transmission of digital data is proportional to the signal rate except for FSK, in which the difference between the carrier signals needs tobe added. We discuss the bandwidth for each technique.

Carrier Signal

In analog transmission, the sending device produces a high-frequency signal that acts as a base for the information signal. This base signal is called the carrier signal or car• rier frequency. The receiving device is tuned to the frequency of the carrier signal that it expects from the sender. Digital information then changes the carrier signal by modify• ing one or more of its characteristics (amplitude, frequency, or phase). This kind of modification is called modulation (shift keying).

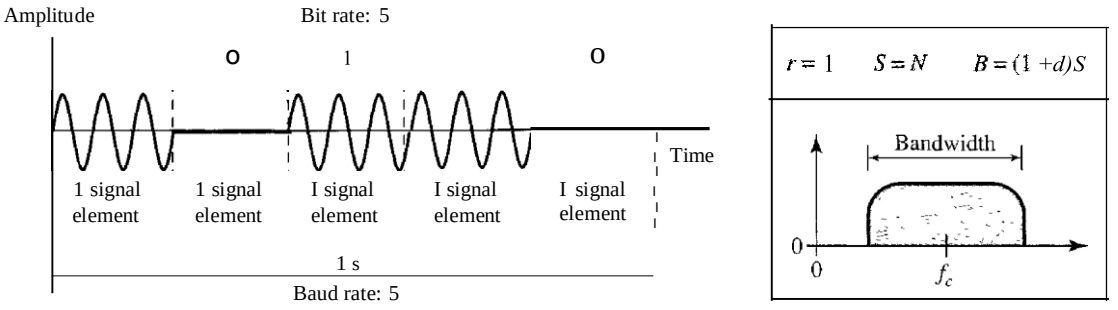
Amplitude Shift Keying

In amplitude shift keying, the amplitude of the carrier signal is varied to create signal elements. Both frequency and phase remain constant while the amplitude changes.

Binary ASK (BASK)

Although we can have several levels (kinds) of signal elements, each with a different amplitude, ASK is normally implemented using only two levels. This is referred to as binary amplitude shift keying or *on-off keying* (OOK). The peak amplitude of one sig• nallevel is 0; the other is the same as the amplitude of the carrier frequency. Figure 5.3 gives a conceptual view of binary ASK.

Figure 5.3 *Binary amplitude shift keying*



Bandwidth for ASK Figure 5.3 also shows the bandwidth for ASK. Although the carrier signal is only one simple sine wave, the process of modulation produces anonperiodic composite signal. This signal, as was discussed in Chapter 3, has a contin• uous set of frequencies. As we expect, the bandwidth is proportional to the signal rate (baud rate). However, there is normally another factor involved, called *d,* which depends on the modulation and filtering process. The value of *d* is between 0 and 1. This means that the bandwidth can be expressed as shown, where 5 is the signal rate and the *B* is the bandwidth.

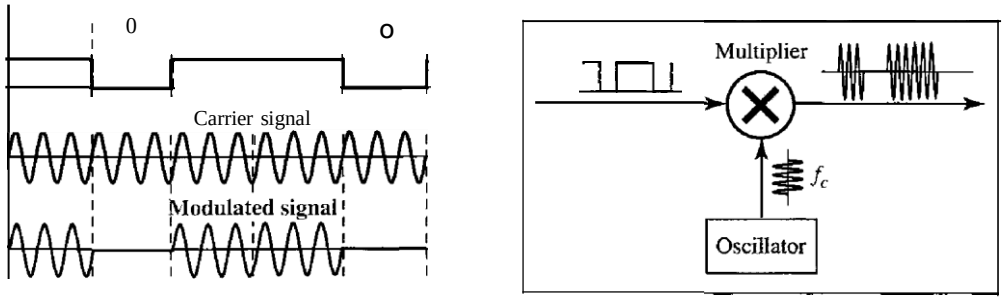
$$B = (1 + d) \times S$$

The formula shows that the required bandwidth has a minimum value of 5 and a maximum value of 25. The most important point here is the location of the bandwidth. The middle of the bandwidth is where *Ie* the carrier frequency, is located. This means if we have a bandpass channel available, we can choose our *Ie* so that the modulated signal occupies that bandwidth. This is in fact the most important advantage of digital• to-analog conversion. We can shift the resulting bandwidth to match what is available.

Implementation The complete discussion of ASK implementation is beyond the scope of this book. However, the simple ideas behind the implementation may help us to better understand the concept itself. Figure 5.4 shows how we can simply implement binary ASK.

If digital data are presented as a unipolar NRZ (see Chapter 4) digital signal with a high voltage of I V and a low voltage of 0 V, the implementation can achieved by multiplying the NRZ digital signal by the carrier signal coming from an oscillator. When the amplitude of the NRZ signal is 1, the amplitude of the carrier frequency is

Figure 5.4 *Implementation ofbinary ASK*



held; when the amplitude of the NRZ signal is 0, the amplitude of the carrier frequency IS zero.

Example 5.3

We have an available bandwidth of 100 kHz which spans from 200 to 300 kHz. What are the car• rier frequency and the bit rate if we modulated our data by using ASK with $d = 1$?

Solution

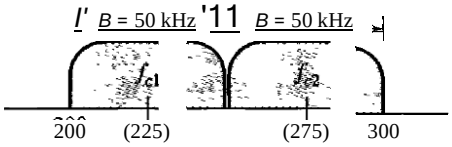
The middle of the bandwidth is located at 250 kHz. This means that our carrier frequency can be *atfe* =250 kHz. We can use the formula for bandwidth to find the bit rate (with $d = 1$ and $r = 1$).

$$B = (1 + d) \times S = 2 \times N \times \frac{1}{r} = 2 \times N = 100 \text{ kHz} \ldots\ldots\ldots\ldots N = 50 \text{ kbps}$$

Example 5.4

In data communications, we normally use full-duplex links with communication in both direc• tions. We need to divide the bandwidth into two with two carrier frequencies, as shown in Fig• ure 5.5. The figure shows the positions of two carrier frequencies and the bandwidths. The available bandwidth for each direction is now 50 kHz, which leaves us with a data rate of 25 kbpsin each direction.

Figure 5.5 *Bandwidth offull-duplex ASK used in Example 5.4*



Multilevel ASK

The above discussion uses only two amplitude levels. We can have multilevel ASK in which there are more than two levels. We can use 4,8, 16, or more different amplitudes for the signal and modulate the data using 2, 3, 4, or more bits at a time. In these cases,

$r = 2$, $r = 3$, $r = 4$, and so on. Although this is not implemented with pure ASK, it is implemented with QAM (as we will see later).

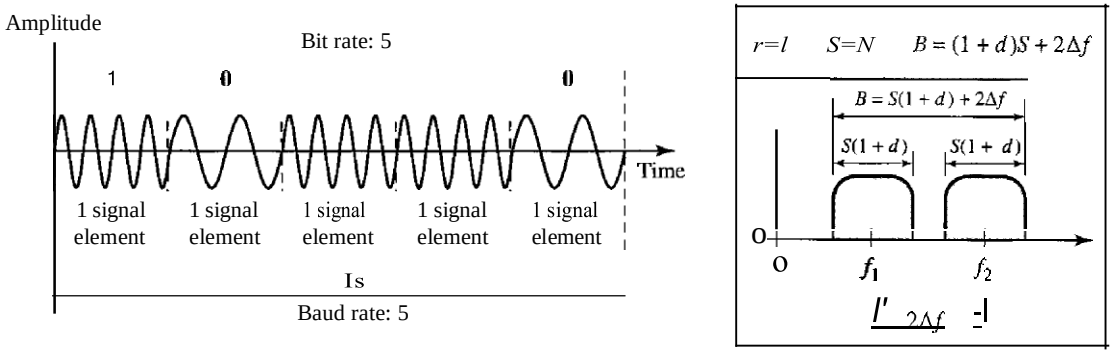
Frequency Shift Keying

In frequency shift keying, the frequency of the carrier signal is varied to represent data. The frequency of the modulated signal is constant for the duration of one signal ele• ment, but changes for the next signal element if the data element changes. Both peak amplitude and phase remain constant for all signal elements.

Binary FSK (BFSK)

One way to think about binary FSK (or BFSK) is to consider two carrier frequencies. In Figure 5.6, we have selected two carrier frequencies,f} and f_2. We use the first carrier if the data element is 0; we use the second if the data element is 1. However, note that this is an unrealistic example used only for demonstration purposes. Normally the carrier frequencies are very high, and the difference between them is very small.

Figure 5.6 *Binary frequency shift keying*



As Figure 5.6 shows, the middle of one bandwidth is f_1 and the middle of the other is f_2. Both f_1 and f_2 are Δf apart from the midpoint between the two bands. The differ• ence between the two frequencies is $2\Delta f$.

Bandwidth for BFSK Figure 5.6 also shows the bandwidth of FSK. Again the car• rier signals are only simple sine waves, but the modulation creates a nonperiodic com• posite signal with continuous frequencies. We can think of FSK as two ASK signals, each with its own carrier frequency Cil or f_2). If the difference between the two fre• quencies is $2\Delta f$,then the required bandwidth is

$$B = (1 + d) \times S + 2\Delta f$$

What should be the minimum value of $2\Delta f$? In Figure 5.6, we have chosen a value greater than $(1 + d)S$. It can be shown that the minimum value should be at least S for the proper operation of modulation and demodulation.

Example 5.5

We have an available bandwidth of 100 kHz which spans from 200 to 300 kHz. What should be the carrier frequency and the bit rate if we modulated our data by using FSK with $d = 1$?

Solution

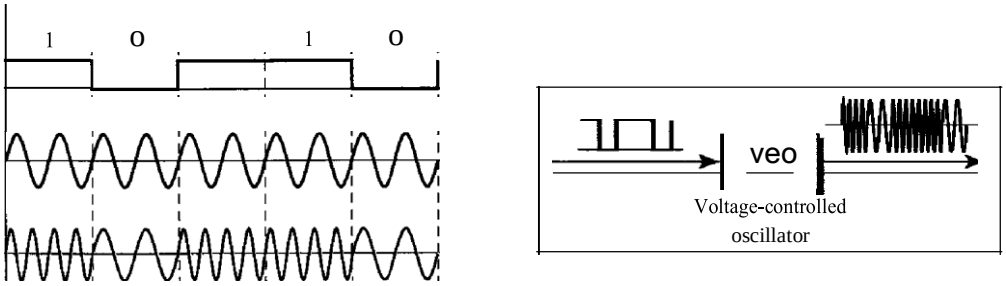
This problem is similar to Example 5.3, but we are modulating by using FSK. The midpoint of the band is at 250 kHz. We choose $2\Delta f$ to be 50 kHz; this means

$$B = (1 + d) \times S + 2\Delta f = 100 \quad \Rightarrow \quad 2S = 50 \text{ kHz} \quad S = 25 \text{ kbaud} \quad N = 25 \text{ kbps}$$

Compared to Example 5.3, we can see the bit rate for ASK is 50 kbps while the bit rate for FSK is 25 kbps.

Implementation There are two implementations of BFSK: noncoherent and coherent. In noncoherent BFSK, there may be discontinuity in the phase when one signal element ends and the next begins. In coherent BFSK, the phase continues through the boundary of two signal elements. Noncoherent BFSK can be implemented by treating BFSK as two ASK modulations and using two carrier frequencies. Coherent BFSK can be implemented by using one *voltage-controlled oscillator* (VCO) that changes its frequency according to the input voltage. Figure 5.7 shows the simplified idea behind the second implementation. The input to the oscillator is the unipolar NRZ signal. When the amplitude of NRZ is zero, the oscillator keeps its regular frequency; when the amplitude is positive, the frequency is increased.

Figure 5.7 *Implementation of BFSK*



Multilevel FSK

Multilevel modulation (MFSK) is not uncommon with the FSK method. We can use more than two frequencies. For example, we can use four different frequencies f_1, f_2, f_3, and 14 to send 2 bits at a time. To send 3 bits at a time, we can use eight frequencies. And so on. However, we need to remember that the frequencies need to be $2\Delta f$ apart. For the proper operation of the modulator and demodulator, it can be shown that the minimum value of $2\Delta f$ needs to be S. We can show that the bandwidth with $d = 0$ is

$$B = (1 + d) \times S + (L - 1)2\Delta f \quad \Rightarrow \quad B = L \times S$$

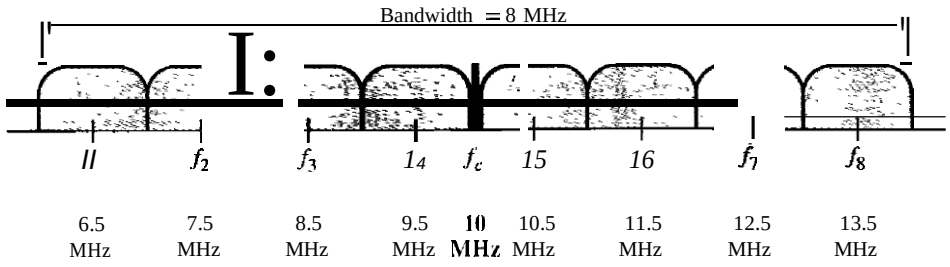
Example 5.6

We need to send data 3 bits at a time at a bit rate of 3 Mbps. The carrier frequency is 10 MHz. Calculate the number of levels (different frequencies), the baud rate, and the bandwidth.

Solution

We can have $L = 2^3 = 8$. The baud rate is S = 3 MHz/3 = 1000 Mbaud. This means that the carrier frequencies must be 1 MHz apart ($2\Delta f = 1$ MHz). The bandwidth is $B = 8 \times 1000 = 8000$. Figure 5.8 shows the allocation of frequencies and bandwidth.

Figure 5.8 *Bandwidth ofMFSK used in Example 5.6*



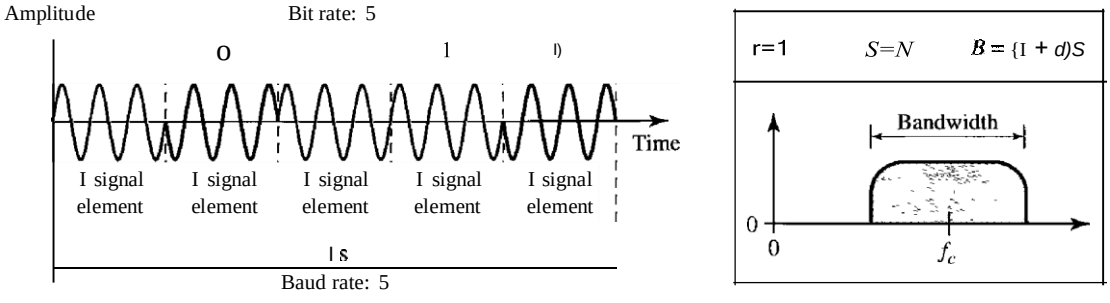
Phase Shift Keying

In phase shift keying, the phase of the carrier is varied to represent two or more differ• ent signal elements. Both peak amplitude and frequency remain constant as the phase changes. Today, PSK is more common than ASK or FSK. However, we will see Sh0l1ly that QAM, which combines ASK and PSK, is the dominant method of digital• to-analog modulation.

Binary PSK (BPSK)

The simplest PSK is binary PSK, in which we have only two signal elements, one witha phase of 0°, and the other with a phase of 180°. Figure 5.9 gives a conceptual viewof PSK. Binary PSK is as simple as binary ASK with one big advantage-it is less

Figure 5.9 *Binary phase shift keying*

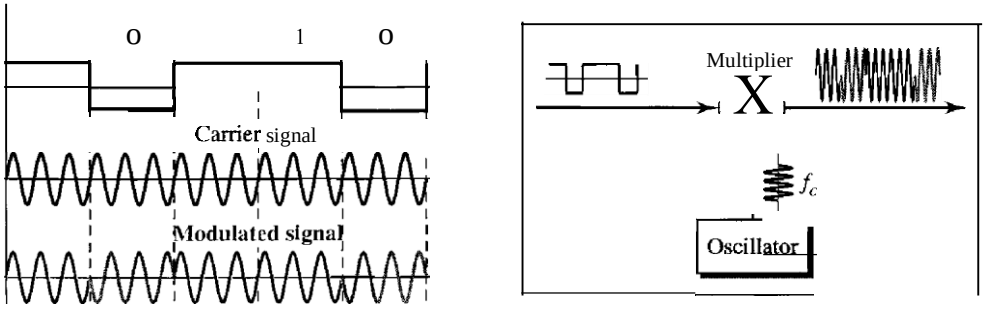


susceptible to noise. In ASK, the criterion for bit detection is the amplitude of the signal; in PSK, it is the phase. Noise can change the amplitude easier than it can changethe phase. In other words, PSK is less susceptible to noise than ASK. PSK is superior to FSK because we do not need two carrier signals.

Bandwidth Figure 5.9 also shows the bandwidth for BPSK. The bandwidth is the same as that for binary ASK, but less than that for BFSK. No bandwidth is wasted for separating two carrier signals.

Implementation The implementation of BPSK is as simple as that for ASK. The reason is that the signal element with phase 180° can be seen as the complement of the signal element with phase 0°. This gives us a clue on how to implement BPSK. We use the same idea we used for ASK but with a polar NRZ signal instead of a unipolar NRZ signal, as shown in Figure 5.10. The polar NRZ signal is multiplied by the carrier frequency; the 1 bit (positive voltage) is represented by a phase starting at 0°; the 0 bit (negative voltage) is represented by a phase starting at 180°.

Figure 5.10 *Implementation of BASK*

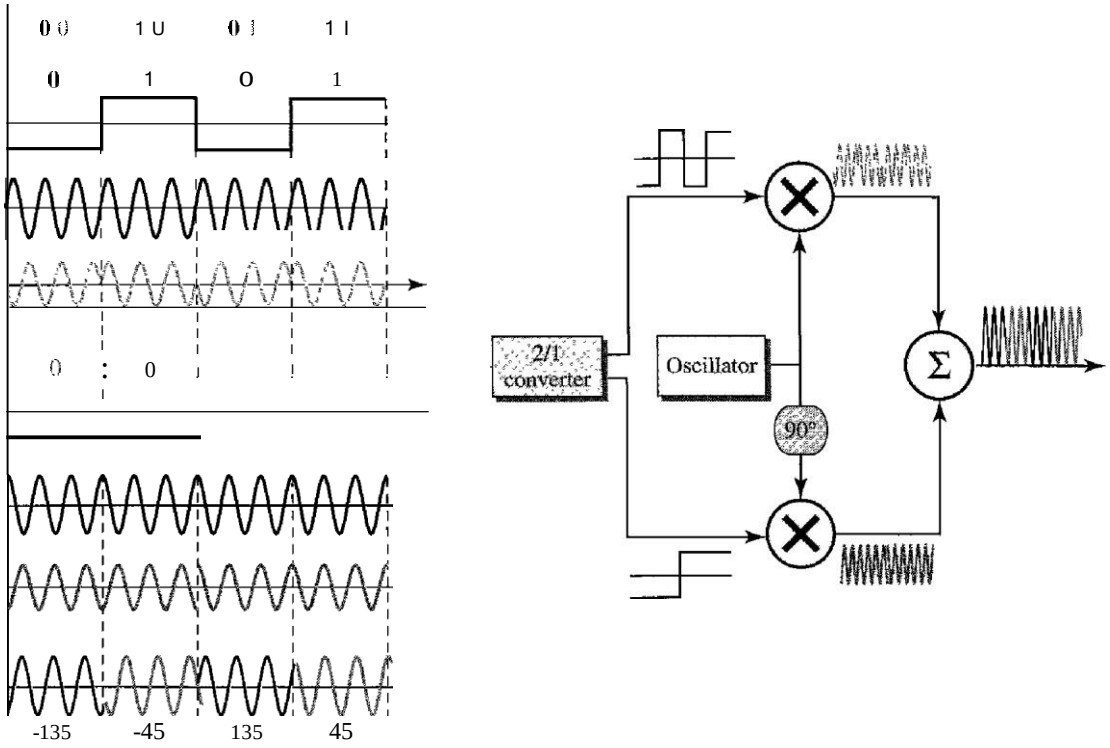


Quadrature PSK (QPSK)

The simplicity of BPSK enticed designers to use 2 bits at a time in each signal element, thereby decreasing the baud rate and eventually the required bandwidth. The scheme is called quadrature PSK or QPSK because it uses two separate BPSK modulations; one is in-phase, the other quadrature (out-of-phase). The incoming bits are first passed through a serial-to-parallel conversion that sends one bit to one modulator and the next bit to the other modulator. If the duration of each bit in the incoming signal is T, the duration of each bit sent to the corresponding BPSK signal is $2T$. This means that the bit to each BPSK signal has one-half the frequency of the original signal. Figure 5.11 shows the idea.

The two composite signals created by each multiplier are sine waves with the same frequency, but different phases. When they are added, the result is another sine wave, with one of four possible phases: $45°$, $-45°$, $135°$, and $-135°$. There are four kinds of signal elements in the output signal ($L = 4$), so we can send 2 bits per signal element ($r = 2$).

Figure 5.11 *QPSK and its implementation*



Example 5.7

Find the bandwidth for a signal transmitting at 12 Mbps for QPSK. The value of $d = 0$.

Solution

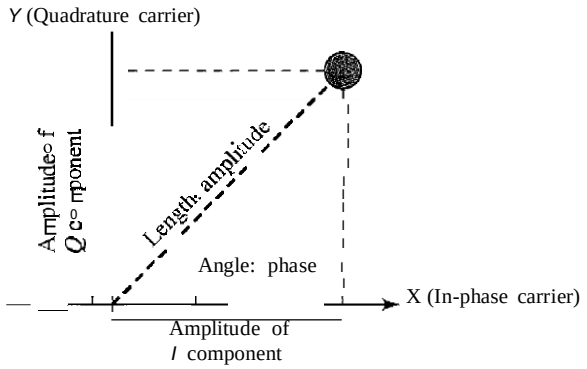
For QPSK, 2 bits is carried by one signal element. This means that $r = 2$. So the signal rate (baud rate) is $S = N \times (1/r) = 6$ Mbaud. With a value of $d = 0$, we have $B = S = 6$ MHz.

Constellation Diagram

A **constellation diagram** can help us define the amplitude and phase of a signal element, particularly when we are using two carriers (one in-phase and one quadrature), The diagram is useful when we are dealing with multilevel ASK, PSK, or QAM (see next section). In a constellation diagram, a signal element type is represented as a dot. The bit or combination of bits it can carry is often written next to it.

The diagram has two axes. The horizontal X axis is related to the in-phase carrier; the vertical *Y* axis is related to the quadrature carrier. For each point on the diagram, four pieces of information can be deduced. The projection of the point on the X axis defines the peak amplitude of the in-phase component; the projection of the point on the *Y* axis defines the peak amplitude of the quadrature component. The length of the line (vector) that connects the point to the origin is the peak amplitude of the signal element (combination of the X and *Y* components); the angle the line makes with the X axis is the phase of the signal element. All the information we need, can easily be found on a constellation diagram. Figure 5.12 shows a constellation diagram.

Figure 5.12 *Concept ofa constellation diagram*



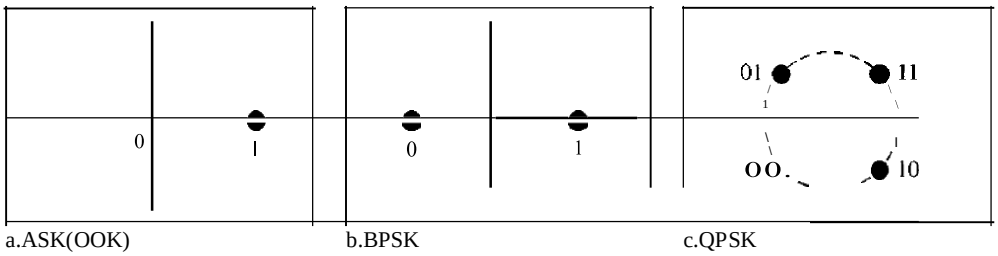
Example 5.8

Show the constellation diagrams for an ASK (OOK), BPSK, and QPSK signals.

Solution

Figure 5.13 shows the three constellation diagrams.

Figure 5.13 *Three constellation diagrams*



Let us analyze each case separately:

a. For ASK, we are using only an in-phase carrier. Therefore, the two points should be on the X axis. Binary 0 has an amplitude of 0 V; binary 1 has an amplitude of 1 V (for example). The points are located at the origin and at 1 unit.

b. BPSK also uses only an in-phase carrier. However, we use a polar NRZ signal for modu• lation. It creates two types of signal elements, one with amplitude 1 and the other with amplitude -1. This can be stated in other words: BPSK creates two different signal elements, one with amplitude I V and in phase and the other with amplitude 1 V and out of phase 180°

c. QPSK uses two carriers, one in-phase and the other quadrature. The point representing 11 is made of two combined signal elements, both with an amplitude of 1 V. One element is rep• resented by an in-phase carrier, the other element by a quadrature carrier. The amplitude of the final signal element sent for this 2-bit data element is 2^{112}, and the phase is *45°*. The argument is similar for the other three points. All signal elements have an amplitude of 2^{112}, but their phases are different *(45°*, 135°, -135°, and -45°). Of course, we could have chosen the amplitude of the carrier to be 1/(21/2) to make the final amplitudes 1 V.

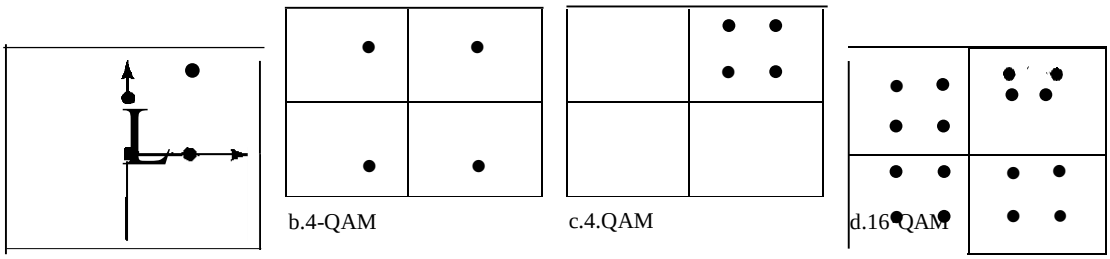
Quadrature Amplitude Modulation

PSK is limited by the ability of the equipment to distinguish small differences in phase. This factor limits its potential bit rate. So far, we have been altering only one of the three characteristics of a sine wave at a time; but what if we alter two? Why not com• bine ASK and PSK? The idea of using two carriers, one in-phase and the other quadra• ture, with different amplitude levels for each carrier is the concept behind quadrature amplitude modulation (QAM).

Quadrature amplitude modulation is a combination of ASK and PSK.

The possible variations of QAM are numerous. Figure 5.14 shows some of these schemes. Figure 5.14a shows the simplest 4-QAM scheme (four different signal ele• ment types) using a unipolar NRZ signal to modulate each carrier. This is the same mechanism we used for ASK (OOK). Part b shows another 4-QAM using polar NRZ, but this is exactly the same as QPSK. Part c shows another QAM-4 in which we used a signal with two positive levels to modulate each of the two carriers. Finally, Figure 5.14dshows a 16-QAM constellation of a signal with eight levels, four positive and four negative.

Figure 5.14 *Constellation diagrams for some QAMs*



Bandwidth for QAM

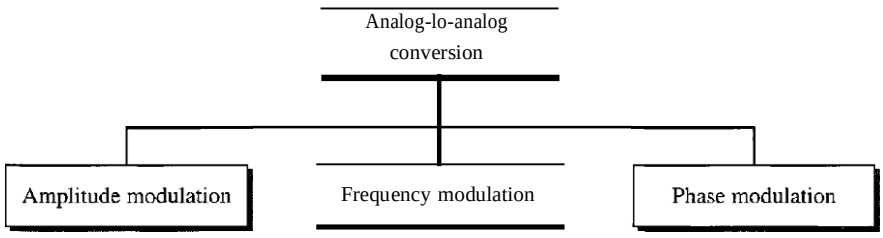
The minimum bandwidth required for QAM transmission is the same as that required for ASK and PSK transmission. QAM has the same advantages as PSK over ASK.

5.2 ANALOG-TO-ANALOG CONVERSION

Analog-to-analog conversion, or analog modulation, is the representation of analog information by an analog signal. One may ask why we need to modulate an analog sig• nal; it is already analog. Modulation is needed if the medium is bandpass in nature or if only a bandpass channel is available to us. An example is radio. The government assigns a narrow bandwidth to each radio station. The analog signal produced by each station is a low-pass signal, all in the same range. To be able to listen to different stations, the low-pass signals need to be shifted, each to a different range.

Analog-to-analog conversion can be accomplished in three ways: amplitude modulation (AM), frequency modulation (FM), and phase modulation (PM). FM and PM are usually categorized together. See Figure 5.15.

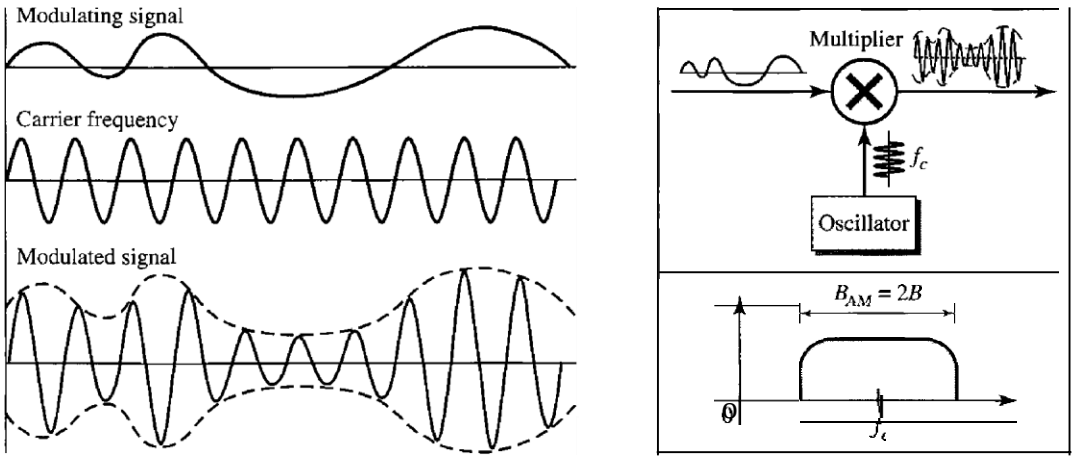
Figure 5.15 *Types ofanalog-to-analog modulation*



Amplitude Modulation

In AM transmission, the carrier signal is modulated so that its amplitude varies with the changing amplitudes of the modulating signal. The frequency and phase of the carrier remain the same; only the amplitude changes to follow variations in the information. Fig• ure 5.16 shows how this concept works. The modulating signal is the envelope of the carrier.

Figure 5.16 *Amplitude modulation*



As Figure 5.16 shows, AM is normally implemented by using a simple multiplier because the amplitude of the carrier signal needs to be changed according to the ampli• tude of the modulating signal.

AM Bandwidth

Figure 5.16 also shows the bandwidth of an AM signal. The modulation creates a band• width that is twice the bandwidth of the modulating signal and covers a range centered on the carrier frequency. However, the signal components above and below the carrier

frequency carry exactly the same information. For this reason, some implementations discard one-half of the signals and cut the bandwidth in half.

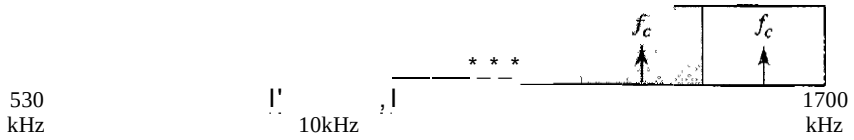
The total bandwIdth required for AM can be determined from the bandwidth of the audio signal: $B_{AM} = 2B$.

Stalldard Balldwidth Allocatioll for AM Radio

The bandwidth of an audio signal (speech and music) is usually 5 kHz. Therefore, an AM radio station needs a bandwidth of 10kHz. In fact, the Federal Communications Commission (FCC) allows 10 kHz for each AM station.

AM stations are allowed carrier frequencies anywhere between 530 and 1700 kHz (1.7 MHz). However, each station's carrier frequency must be separated from those on either side of it by at least 10 kHz (one AM bandwidth) to avoid interference. If one station uses a carrier frequency of 1100 kHz, the next station's carrier frequency cannot be lower than 1110 kHz (see Figure 5.17).

Figure 5.17 *AM band allocation*



Frequency Modulation

In FM transmission, the frequency of the carrier signal is modulated to follow the chang• ing voltage level (amplitude) of the modulating signal. The peak amplitude and phase of the carrier signal remain constant, but as the amplitude of the information signal changes, the frequency of the carrier changes correspondingly. Figure 5.18 shows the relationships of the modulating signal, the carrier signal, and the resultant FM signal.

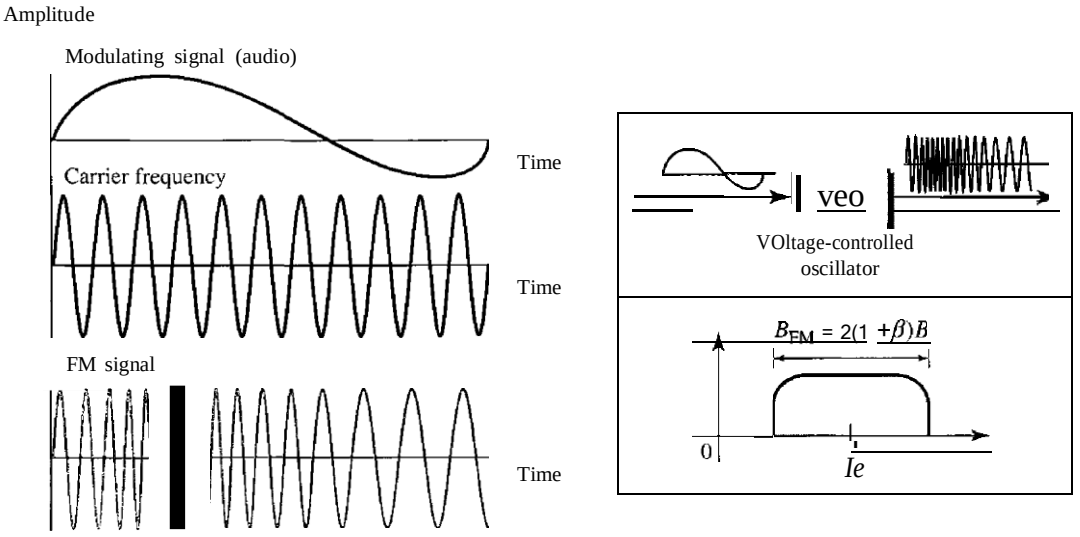
As Figure 5.18 shows, FM is nOimalIy implemented by using a voltage-controlled oscillator as with FSK. The frequency of the oscillator changes according to the input voltage which is the amplitude of the modulating signal.

FM Balldwidth

Figure 5.18 also shows the bandwidth of an FM signal. The actual bandwidth is diffi• cult to determine exactly, but it can be shown empirically that it is several times that of the analog signal or $2(1 + \beta)B$ where β is a factor depends on modulation technique with a common value of 4.

The total bandwidth required for FM can be determined from the bandwidth of the audio signal: $B_{FM} = 2(1 + \beta)B$.

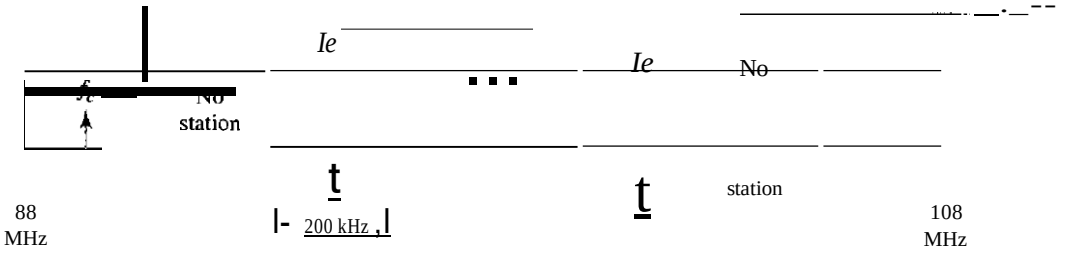
Figure 5.18 *Frequency modulation*



Standard Bandwidth Allocation for FM Radio

The bandwidth of an audio signal (speech and music) broadcast in stereo is almost 15 kHz. The FCC allows 200 kHz (0.2 MHz) for each station. This mean $\beta = 4$ with some extra guard band. FM stations are allowed carrier frequencies anywhere between 88 and 108 MHz. Stations must be separated by at least 200 kHz to keep their band• widths from overlapping. To create even more privacy, the FCC requires that in a given area, only alternate bandwidth allocations may be used. The others remain unused to pre• vent any possibility of two stations interfering with each other. Given 88 to 108 MHz asa range, there are 100 potential PM bandwidths in an area, of which 50 can operate at anyone time. Figure 5.19 illustrates this concept.

Figure 5.19 *FM band allocation*



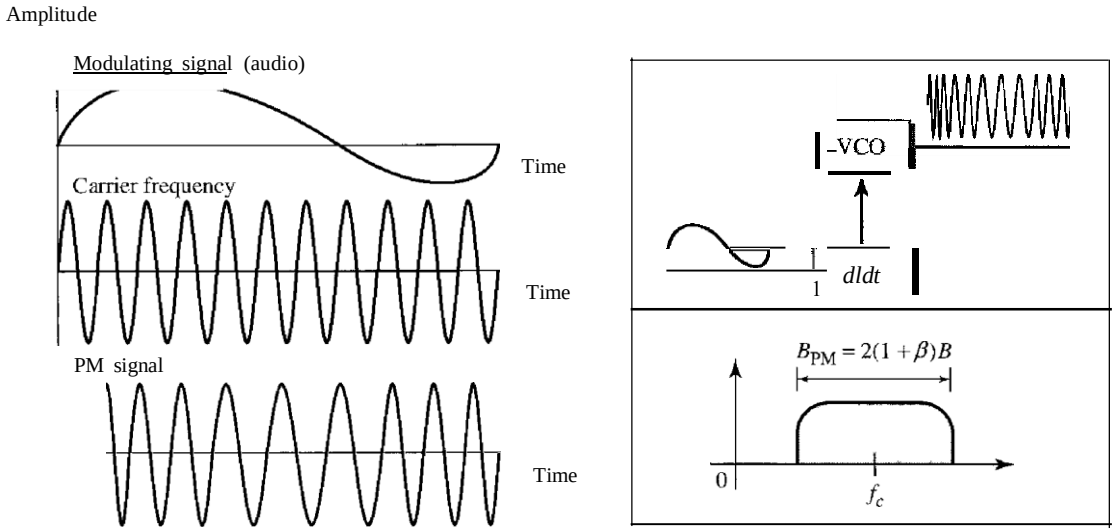
Phase Modulation

In PM transmission, the phase of the carrier signal is modulated to follow the changing voltage level (amplitude) of the modulating signal. The peak amplitude and frequency of the carrier signal remain constant, but as the amplitude of the information signal changes, the

phase of the carrier changes correspondingly. It can proved mathemati• cally (see Appendix C) that PM is the same as FM with one difference. In FM, the instantaneous change in the carrier frequency is proportional to the amplitude of the

modulating signal; in PM the instantaneous change in the carrier frequency is proportional to the derivative of the amplitude of the modulating signal. Figure 5.20 shows the relationships of the modulating signal, the carrier signal, and the resultant PM signal.

Figure 5.20 *Phase modulation*



As Figure 5.20 shows, PM is normally implemented by using a voltage-controlled oscillator along with a derivative. The frequency of the oscillator changes according to the derivative of the input voltage which is the amplitude of the modulating signal.

PM Bandwidth

Figure 5.20 also shows the bandwidth of a PM signal. The actual bandwidth is difficultto determine exactly, but it can be shown empirically that it is several times that of the analog signal. Although, the formula shows the same bandwidth for FM and PM, thevalue of β is lower in the case of PM (around 1 for narrowband and 3 for wideband).

The total bandwidth required for PM can be determined from the bandwidth and maximum amplitude of the modulating signal: $B_{PM} = 2(1 + \beta)B$.

5.3 SUMMARY

- Digital-to-analog conversion is the process of changing one of the characteristics of an analog signal based on the information in the digital data.
- Digital-to-analog conversion can be accomplished in several ways: amplitude shift keying (ASK), frequency shift keying (FSK), and phase shift keying (PSK). Quadrature amplitude modulation (QAM) combines ASK and PSK.
- In amplitude shift keying, the amplitude of the carrier signal is varied to create signal elements. Both frequency and phase remain constant while the amplitude changes.

- In frequency shift keying, the frequency of the carrier signal is varied to represent data. The frequency of the modulated signal is constant for the duration of one signal element, but changes for the next signal element if the data element changes. Both peak amplitude and phase remain constant for all signal elements.
- In phase shift keying, the phase of the carrier is varied to represent two or more different signal elements. Both peak amplitude and frequency remain constant as the phase changes.
- A constellation diagram shows us the amplitude and phase of a signal element, particularly when we are using two carriers (one in-phase and one quadrature).
- Quadrature amplitude modulation (QAM) is a combination of ASK and PSK. QAM uses two carriers, one in-phase and the other quadrature, with different amplitude levels for each carrier.
- Analog-to-analog conversion is the representation of analog information by an analog signal. Conversion is needed if the medium is bandpass in nature or if only a bandpass bandwidth is available to us.
- Analog-to-analog conversion can be accomplished in three ways: amplitude modulation (AM), frequency modulation (FM), and phase modulation (PM).
- In AM transmission, the carrier signal is modulated so that its amplitude varies with the changing amplitudes of the modulating signal. The frequency and phase of the carrier remain the same; only the amplitude changes to follow variations in the information.
- In PM transmission, the frequency of the carrier signal is modulated to follow the changing voltage level (amplitude) of the modulating signal. The peak amplitude

and phase of the carrier signal remain constant, but as the amplitude of the infor• mation signal changes, the frequency of the carrier changes correspondingly.

- In PM transmission, the phase of the carrier signal is modulated to follow the changing voltage level (amplitude) of the modulating signal. The peak amplitude and frequency of the carrier signal remain constant, but as the amplitude of the information signal changes, the phase of the carrier changes correspondingly.

5.4 PRACTICE SET

Review Questions

1. Define analog transmission.
2. Define carrier signal and its role in analog transmission.
3. Define digital-to-analog conversion.
4. Which characteristics of an analog signal are changed to represent the digital signal in each of the following digital-to-analog conversion?
   a. ASK
   b. FSK
   c. PSK
   d. QAM
5. Which of the four digital-to-analog conversion techniques (ASK, FSK, PSK or QAM) is the most susceptible to noise? Defend your answer.
6. Define constellation diagram and its role in analog transmission.
7. What are the two components of a signal when the signal is represented on a con- . stellation diagram? Which component is shown on the horizontal axis? Which is shown on the vertical axis?
8. Define analog-to-analog conversion?
9. Which characteristics of an analog signal are changed to represent the lowpass analog signal in each of the following analog-to-analog conversions?
   a. AM
   b. FM
   c. PM
10. Which of the three analog-to-analog conversion techniques (AM, FM, or PM) is the most susceptible to noise? Defend your answer.

Exercises

11. Calculate the baud rate for the given bit rate and type of modulation.
    a. 2000 bps, FSK
    b. 4000 bps, ASK
    c. 6000 bps, QPSK
    d. 36,000 bps, 64-QAM

12. Calculate the bit rate for the given baud rate and type of modulation.
    a. 1000 baud, FSK
    b. 1000 baud, ASK
    c. 1000 baud, BPSK
    d. 1000 baud, 16-QAM
13. What is the number of bits per baud for the following techniques?
    a. ASK with four different amplitudes
    b. FSK with 8 different frequencies
    c. PSK with four different phases
    d. QAM with a constellation of 128 points.
14. Draw the constellation diagram for the following:
    a. ASK, with peak amplitude values of 1 and 3
    b. BPSK, with a peak amplitude value of 2
    c. QPSK, with a peak amplitude value of 3
    d. 8-QAM with two different peak amplitude values, I and 3, and four different phases.
15. Draw the constellation diagram for the following cases. Find the peak amplitude value for each case and define the type of modulation (ASK, FSK, PSK, or QAM). The numbers in parentheses define the values of I and Q respectively.
    a. Two points at (2, 0) and (3, 0).
    b. Two points at (3, 0) and (-3, 0).
    c. Four points at (2, 2), (-2, 2), (-2, -2), and (2, -2).
    d. Two points at (0 , 2) and (0, -2).
16. How many bits per baud can we send in each of the following cases if the signal constellation has one of the following number of points?
    a. 2
    b. 4
    c. 16
    d. 1024
17. What is the required bandwidth for the following cases if we need to send 4000 bps? Let d = 1.
    a. ASK
    b. FSK with $2\Delta f = 4$ KHz
    c. QPSK
    d. 16-QAM
18. The telephone line has 4 KHz bandwidth. What is the maximum number of bits we can send using each of the following techniques? Let d = O.
    a. ASK
    b. QPSK
    c. 16-QAM
    d. 64-QAM

19. A corporation has a medium with a I-MHz bandwidth (lowpass). The corporation needs to create 10 separate independent channels each capable of sending at least 10 Mbps. The company has decided to use QAM technology. What is the mini• mum number of bits per baud for each channel? What is the number of points in the constellation diagram for each channel? Let d = O.
20. A cable company uses one of the cable TV channels (with a bandwidth of 6 MHz) to provide digital communication for each resident. What is the available data rate for each resident if the company uses a 64-QAM technique?
21. Find the bandwidth for the following situations if we need to modulate a 5-KHz voice.
    a. AM
    b. PM (set $\beta = 5$)
    c. PM (set $\beta = 1$)
22. Find the total number of channels in the corresponding band allocated by FCC.
    a. AM
    b. FM