GUDLAVALLERU ENGINEERING COLLEGE

(An Autonomous Institute with Permanent Affiliation to JNTUK, Kakinada)

Seshadri Rao Knowledge Village, Gudlavalleru – 521 356.

Department of Computer Science and Engineering

HANDOUT

on

COMPUTER NETWORKS-I

Vision of the Department

To be a Centre of Excellence in computer science and engineering education and training to meet the challenging needs of the industry and society

Mission of the Department

- To impart quality education through well-designed curriculum in tune with the growing software needs of the industry.
- To be a Centre of Excellence in computer science and engineering education and training to meet the challenging needs of the industry and society.
- To serve our students by inculcating in them problem solving, leadership, teamwork skills and the value of commitment to quality, ethical behavior & respect for others.
- To foster industry-academia relationship for mutual benefit and growth.

Program Educational Objectives

PEO1: Identify, analyze, formulate and solve Computer Science and Engineering problems both independently and in a team environment by using the appropriate modern tools.

PEO2: Manage software projects with significant technical, legal, ethical, social, environmental and economic considerations.

PEO3: Demonstrate commitment and progress in lifelong learning, professional development, leadership and Communicate effectively with professional clients and the public.

HANDOUT ON COMPUTER NETWORKS-I

Class & Sem. :III B.Tech – I Semester Year :2018-19

Branch : CSE Credits: 3

==========

1. Brief History and Scope of the Subject

A **computer network** or **data network** is a telecommunications network which allows computers to exchange data. In computer networks, networked computing devices exchange data with each other using a data link. The connections between nodes are established using either cable media or wireless media. The best-known computer network is the Internet. Network computer devices that originate, route and terminate the data are called network nodes Nodes can include hosts such as personal computers, phones, servers as well as networking hardware. Two such devices can be said to be networked together when one device is able to exchange information with the other device, whether or not they have a direct connection to each other. Computer networks differ in the transmission medium used to carry their signals, the communications protocols to organize network traffic, the network's size, topology and organizational intent. Computer networks support an enormous number of applications such as access to the World Wide Web, video, digital audio, shared use of application and storage servers, printers, and fax machines, and use of email and instant messaging applications as well as many others.

2. Pre-Requisites

Basic knowledge on computer hardware and software components.

3. Course Objectives:

• To familiarize with the basics of data communication, various types of computer networks and topologies.
• To get exposed to the OSI and TCP/IP protocol suite.

4. Course Outcomes:

Students will be able to:

CO1: defining the concept of local area networks, their topologies, protocols and applications.
CO2: analyze the requirements for a given organizational structure and select the most appropriate networking architecture and technologies.
CO3: have a working knowledge of DDL and MAC protocols.
CO4: specify and identify deficiencies in existing protocols, and then go onto formulate new and better protocols

5. Program Outcomes:

Graduates of the Computer Science and Engineering Program will have

a. An ability to apply knowledge of computing, mathematics, science and engineering fundamentals to solve complex engineering problems.

b. An ability to formulate and analyze a problem, and define the computing requirements appropriate to its solution using basic principles of mathematics, science and computer engineering.

c. An ability to design, implement, and evaluate a computer based system, process, component, or software to meet the desired needs.

d. An ability to design and conduct experiments, perform analysis and interpretation of data and provide valid conclusions.

e. An ability to use current techniques, skills, and tools necessary for computing practice.

f. An ability to understand legal, health, security and social issues in Professional Engineering practice.

g. An ability to understand the impact of professional engineering solutions on environmental context and the need for sustainable development.

h. An ability to understand the professional and ethical responsibilities of an engineer.

i. An ability to function effectively as an individual, and as a team member/ leader in accomplishing a common goal.

j. An ability to communicate effectively, make effective presentations and write and comprehend technical reports and publications.

k. An ability to learn and adopt new technologies, and use them effectively towards continued professional development throughout the life.

l. An ability to understand engineering and management principles and their application to manage projects in the software industry.

6. Mapping of Course Outcomes with Program Outcomes:

	a	b	c	d	e	f	g	h	i	j	k
CO1		H								H	
CO2			M								
CO3	M	M			H						
CO4	H	L		M				L			

7. Prescribed Text Books

a)Behrouz A Fourzan, “Data communications and networking”, TMH,4th edition.

b) Andrew S Tanenbaum, “Computer Networks”, Pearson, 4th edition.

c) Mayank Dave, “Computer Networks” , Cengage

Reference Text Books

a)Larry L Peterson and Bruce S Davie, Computer networks, A system Approach, Elsevier, 5th edition

URLs and Other E-Learning Resources

a. Data Communication introduction : www.cne.gmu.edu

b. Protocol Standards: www.ietf.org

8. Digital Learning Materials:

- http://nptel.ac.in/courses/106105081
- http://nptel.ac.in/courses/106105080/
- http://nptel.ac.in/courses/106106091/
- https://www.youtube.com/watch?v=tP9y0bVUYCA
- https://www.youtube.com/watch?v=UXMIxCYZu8o
- http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-033-computer-system- engineering-spring-2009/video-lectures/lecture-11/

- http://freevideolectures.com/Course/2276/Computer-Networks#
- http://homepages.herts.ac.uk/~comqrgd/docs/network-notes/network-notes.pdf
- http://intronetworks.cs.luc.edu/current/ComputerNetworks.pdf
- http://www.pucomp.org/2008/07/data-communications-ppt-from-forouzan.html
- http://www.technolamp.co.in/2010/08/computer-networks-tanenbaum-powerpoint.html

9. Lecture Schedule / Lesson Plan

Topic	No. of Periods	
	Theory	Tutorial
UNIT –1: Introduction		
Data Communications: Components, data representation, data flow	2	1
Networks: distributed processing, network criteria, physical structures	1	
Network models, categories of network, inter connection of networks	1	
Protocols and Standards: protocols, standards, standard organization, Internet standards	1	
OSI models: layered architecture, peer to peer process, Encapsulation	1	1
Layers in OSI model	2	
TCP/IP protocol suite	1	
Addressing: physical address, Logical address, port address	1	
UNIT – 2:		
Multiplexing: Frequency division multiplexing	1	1
Wave length division multiplexing	1	
Synchronous time division multiplexing	2	
Statistical time division multiplexing	1	1
Introduction to switching: Circuit Switched Networks	1	
Datagram Networks	1	
Virtual Circuit Networks	1	
UNIT – 3:		
Framing: fixed size framing, variable size framing	1	2
Flow control, Error control,	1	
Error detection, Error correction : block coding, linear block codes	2	
Cyclic codes - cyclic redundancy check	2	

Checksum - idea, one's complement internet check sum	1	1
Services provided to Network Layer, Elementary Data link Layer Protocols - Unrestricted Simplex protocol	2	
Simplex Stop-and-Wait Protocol, Simplex protocol for Noisy Channel	2	
UNIT – 4:		
Sliding Window Protocol: One bit, Go back N	2	1
Selective Repeat	2	
HDLC: configuration and transfer modes	2	1
Frames, control field	2	
point to point protocol(PPP): framing, transition phase	1	
UNIT – 5:		
Random Access: ALOHA	2	1
career sense multiple access (CSMA)	2	
Career sense multiple access with collision detection	1	
Career sense multiple access with collision avoidance	1	1
Controlled Access – Reservation, polling, Token passing	2	
UNIT – 6:		
IEEE Standards: Data link layer, physical layer	1	2
Manchester encoding	2	
Fast Ethernet- MAC Sub Layer, physical layer	1	
IEEE - 802.11- Architecture, MAC sub layer, frame structure.	2	
Data Link Layer Switching- Bridges, Local internetworking.	1	1
Spanning tree bridges,Remote bridges	2	
switch virtual LANs	1	
Total No.of Periods:	**56**	**14**

10. Seminar Topics

- OSI Reference Model
- TCP/IP protocol suite
- Differences between OSI and TCP/IP models

Learning Material

UNIT-I

Syllabus:
Introduction-Data Communication, components, data representation, data flow; **Networks**-distributed processing, network criteria, physical structures, network models, categories of network, inter connection of networks; **Protocols & standards**-protocols, standards, standard organization, internet standards; **The OSI models**-layered architecture, peer to peer process, encapsulation, Layers in OSI model: physical layer, data link layer, Network layer, transport layer, session layer , presentation layer , application layer ; **TCP/IP protocol suite**-physical and data link layers, network layer, transport layer, application layer; **Addressing**- physical address, logical address, port address

INTRODUCTION

1.1 Data Communications:

- In recent years, the network that is making significant impact in our day-to-day life is the **Computer network**. By computer network we mean an interconnected set of autonomous computers. The term autonomous implies that the computers can function independent of others. However, these computers can exchange information with each other through the communication network system.

- Computer networks have emerged as a result of the convergence of two technologies of this century- Computer and Communication as shown in Fig. 1.1. The consequence of this revolutionary merger is the emergence of a integrated system that transmit all types of data and information.

- There is no fundamental difference between data communications and data processing and there are no fundamental differences among data, voice and video communications.

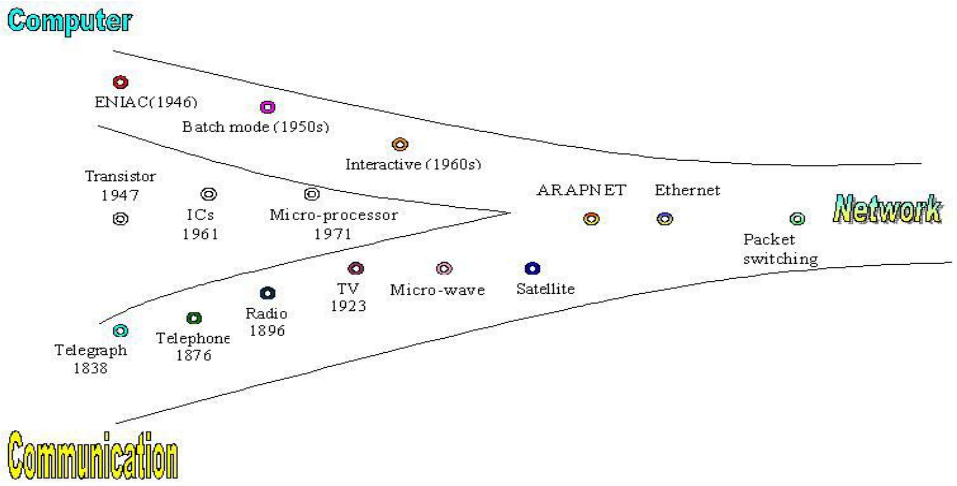


Fig: Evolution of computer networks

- 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 combination of hardware (physical equipment) and software (programs).

- The effectiveness of a data communications system depends on four fundamental characteristics: delivery, 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 significant 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 30 ms. If some of the packets arrive with 30-ms delay and others with 40-ms delay, an uneven quality in the video is the result.

1.2 Components:

A data communications system has five components:

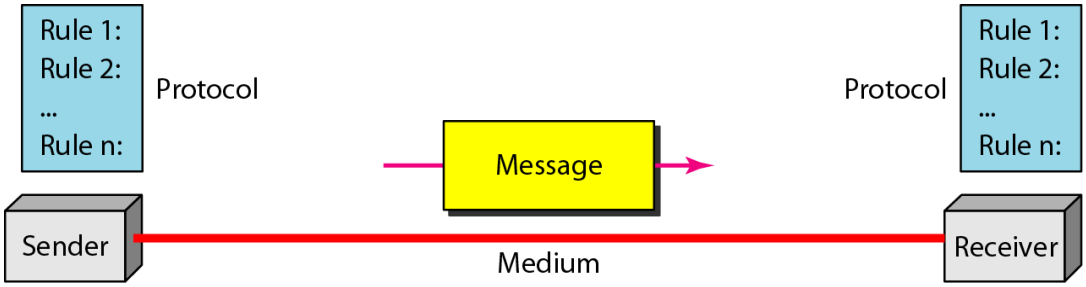


Fig: Components of data communication system

1. Message: The message is the information (data) to be communicated. Popular forms of information include text, numbers, pictures, audio, and video.
2. Sender: The sender is the device that sends the data message. It can be a computer, workstation, telephone handset, video camera, and so on.
3. Receiver: The receiver is the device that receives the message. It can be a computer, workstation, telephone handset, television, and so on.
4. Transmission medium: The transmission medium is the physical path by which a message travels from sender to receiver. Some examples 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 represents 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.

1.3 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 (0's or 1's). 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 Information 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.

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.

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.

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.

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.

1.4 Data Flow:

Communication between two devices can be simplex, half-duplex, or full-duplex as shown in the below figure.

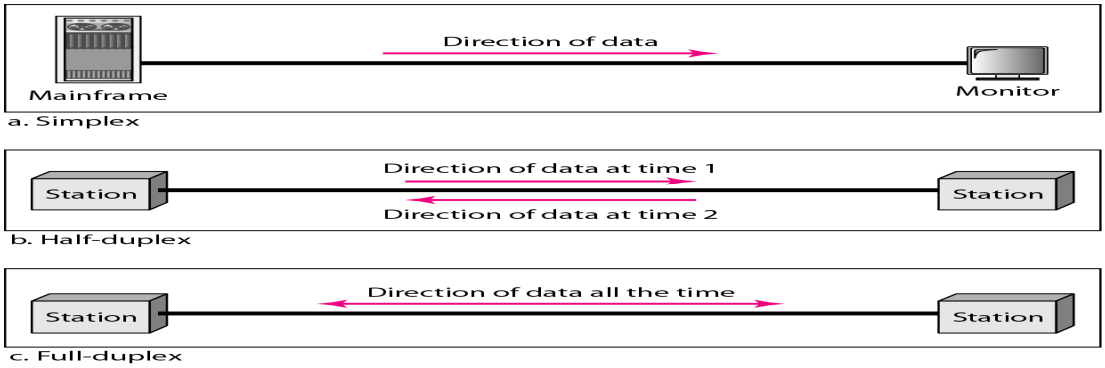


Fig: Transmission modes

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 (Figure a). Keyboards and traditional monitors are examples of simplex devices.
- The keyboard 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 (Figure b). The half-duplex mode is like a one-lane road with traffic allowed in both directions.
- 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 simultaneously(figure c).
- The full-duplex mode is like a two-way street with traffic flowing in both directions 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.

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.

2.1Distributed 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.

2.2 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.

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.

2.3 Physical Structures

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

2.3.1Type *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 a).

- 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 b). 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.

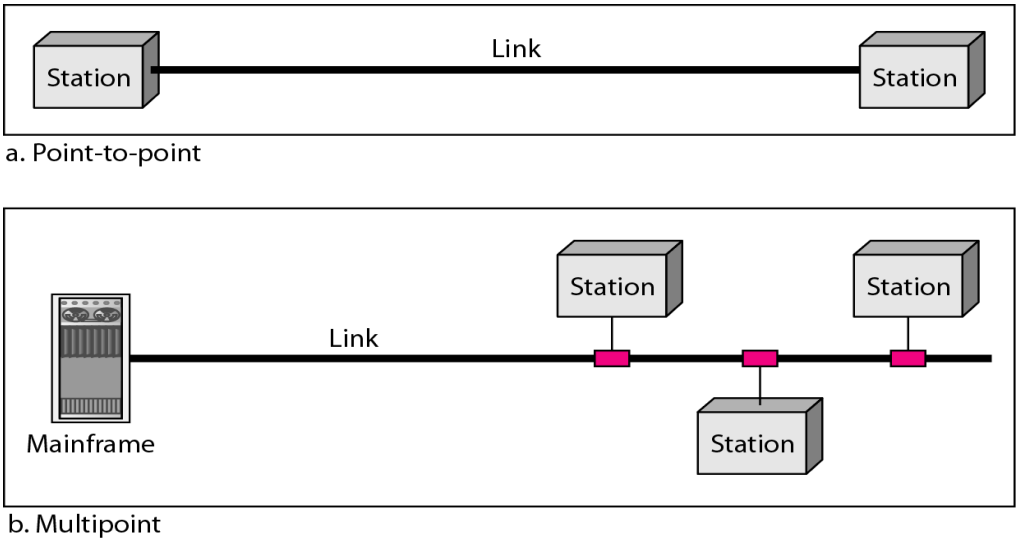


Fig: point-to-point and multipoint connection

2.3.2 Physical Topology

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

linking devices (usually called nodes) to one another. There are four basic topologies
possible: mesh, star, bus, and ring

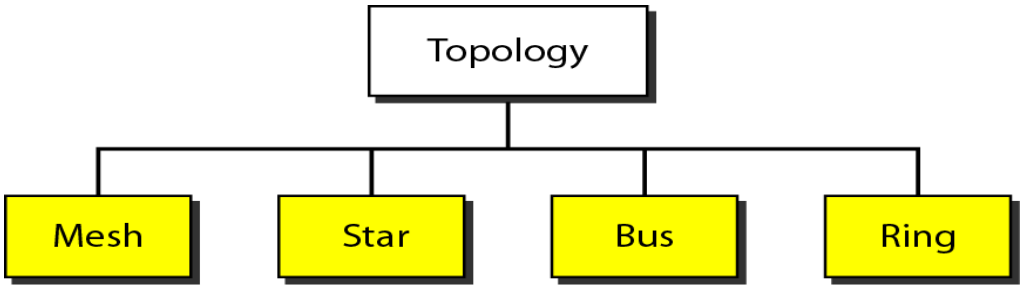


Fig: Types of topologies

Mesh Topology: 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 below figure) to be connected to the other *n* - 1 stations.

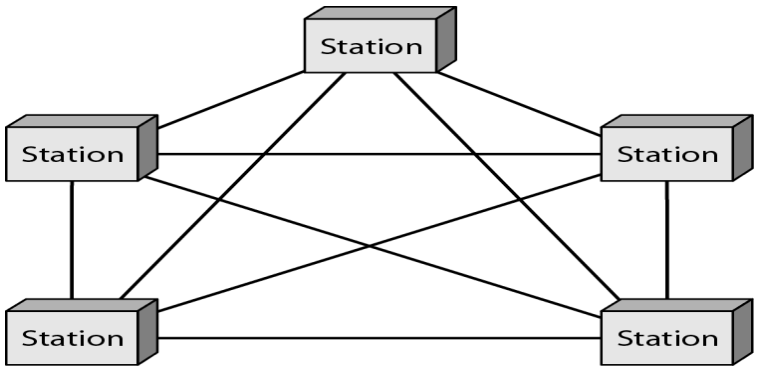


Fig: mesh topology

Advantages:

1. The use of dedicated links guarantees that each connection can carry its own data load, thus eliminating the traffic problems that can occur when links must be shared by multiple devices.

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

3. 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.

Disadvantages:

1. Disadvantage of a mesh are related to the amount of cabling because every device must be connected to every other device, installation and reconnection are difficult.

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

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 connected device (see figure).

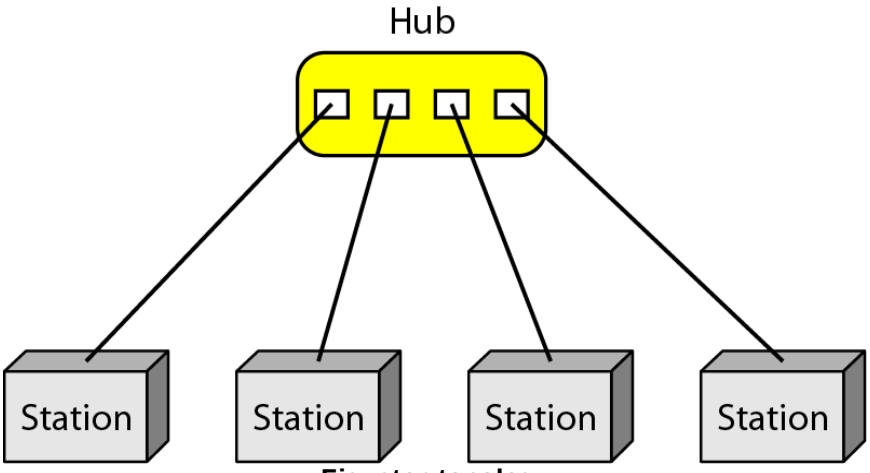


Fig: star topology

- 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.
- Other advantages include robustness. If one link fails, only that link is affected. All other links remain active.
- One big disadvantage of a star topology is the dependency of the whole topology on 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).

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).

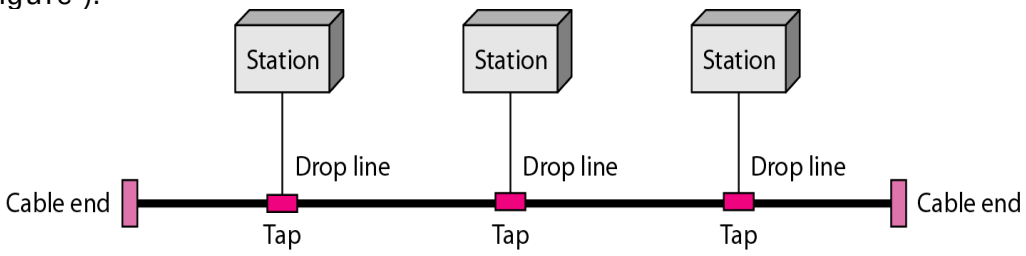


Fig: bus topology

- Nodes are connected to the bus cable by drop lines and taps.
- A drop line is a connection 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.

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

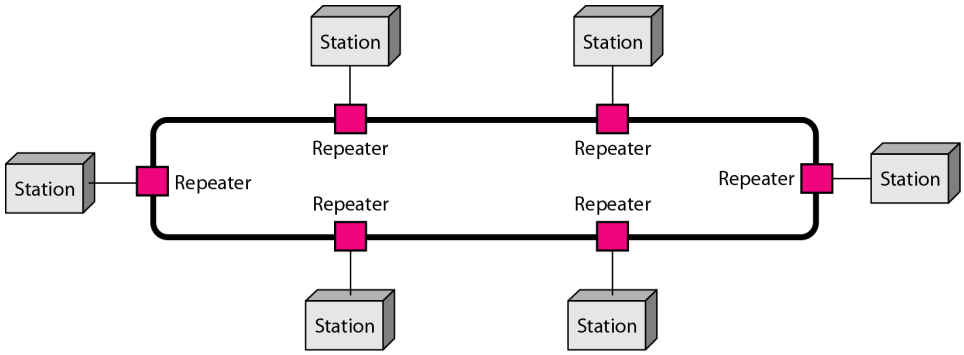


Fig: ring topology

A ring is relatively easy to install and reconfigure. Each device is linked to only its immediate neighbours (either physically or logically). To add or delete a device requires changing only two connections. The only constraints are media and traffic considerations (maximum ring length and number of devices). In addition, fault isolation is simplified.

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 below figure

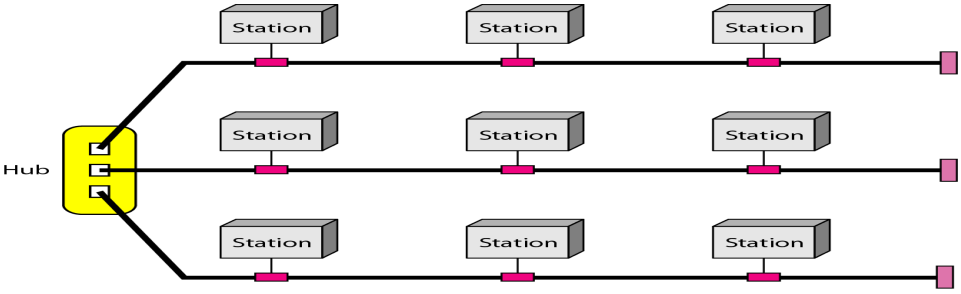


Fig: hybrid topology

2.4 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.

2.5 Categories of Networks :

They are divided into Local Area (LAN), Metropolitan Area Network (MAN) and Wide Area Networks (WAN) based on their size, transmission media and topology.

Local Area Network

- LAN is usually privately owned and links the devices in a single office, building or campus of up to few kilometers in size.
- These are used to share resources (may be hardware or software resources) and to exchange information.
- LANs are distinguished from other kinds of networks by three categories: their size, transmission technology and topology.
- LANs are restricted in size, which means that their worst-case transmission time is bounded and known in advance.
- Hence this is more reliable as compared to MAN and WAN. Knowing this bound makes it possible to use certain kinds of design that would not otherwise be possible.
- It also simplifies network management.

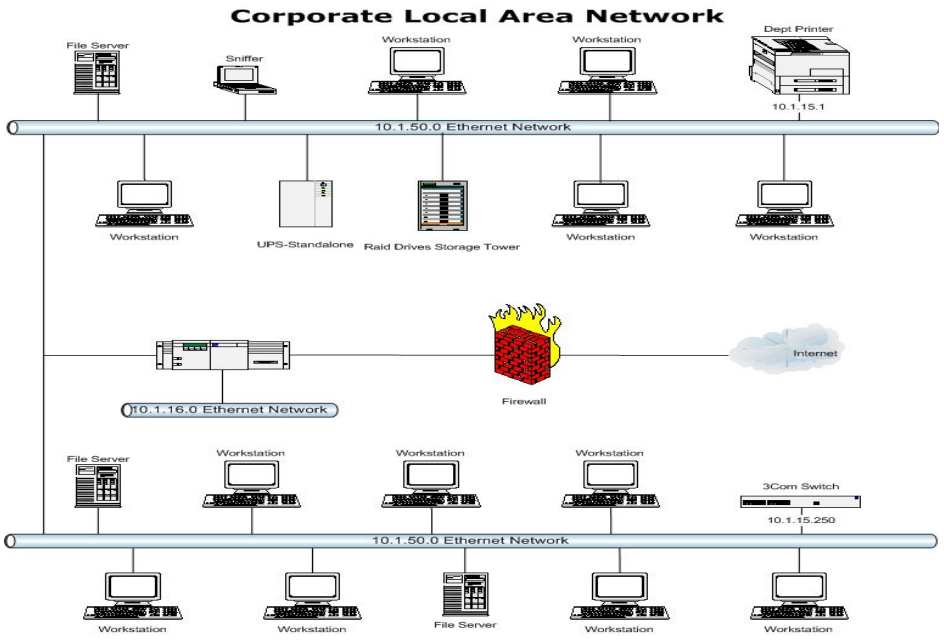
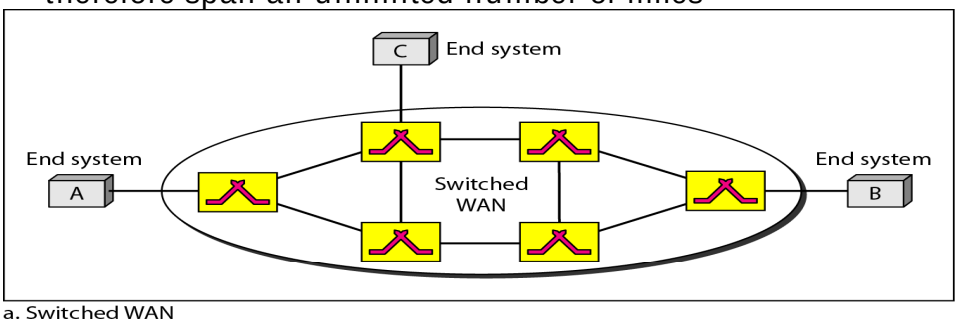


Fig: LAN

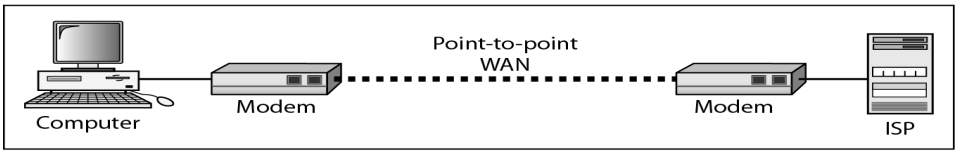
Wide Area Network

A wide area network (WAN) provides long-distance transmission of data, image, audio, and video information over large geographic areas that may comprise a country, a continent, or even the whole world.

- WAN provides long-distance transmission of data, voice, image and information over large geographical areas that may comprise a country, continent or even the whole world.
- In contrast to LANs, WANs may utilize public, leased or private communication devices, usually in combinations, and can therefore span an unlimited number of miles



a. Switched WAN



b. Point-to-point WAN

Fig: WAN

Metropolitan Area Networks

A metropolitan area network (MAN) is a network with a size between a LAN and a WAN. It normally covers the area inside a town or a city. It is designed for customers who need a high-speed connectivity, normally to the Internet, and have endpoints spread over a city or part of city.

- A good example of a MAN is the part of the telephone company network that can provide a high-speed DSL line to the customer. Another example is the cable TV network that originally was designed for cable TV, but today can also be used for high-speed data connection to the Internet.

Interconnection of Networks: Internetwork

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 home. 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 below figure

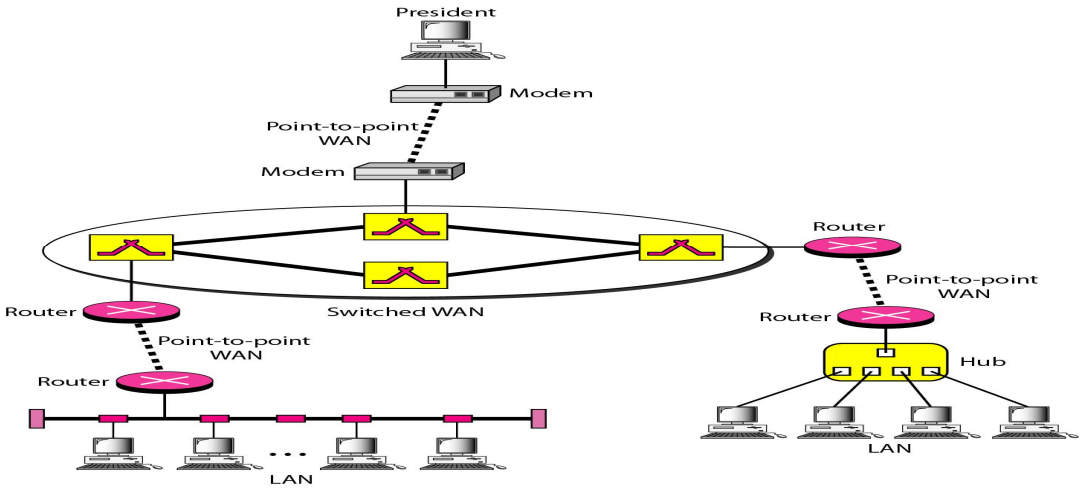


Fig: Example for WAN

3.PROTOCOLS AND STANDARDS

The term **protocol**, which is synonymous with rule and the term **standards**, which are agreed-upon rules.

3.1 Protocols

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.

3.2 Standards

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.

3.2.1 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:

a. **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.

b. 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).

c. **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.

d. **Institute of Electrical and Electronics Engineers (IEEE):** The Institute of Electrical and Electronics Engineers is the largest professional engineering society in the world.

- 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.

e. **Electronic Industries Association (EIA):** Aligned with ANSI, the Electronic Industries Association is a non-profit organization devoted to the promotion of electronics manufacturing concerns.

- Its activities include public awareness education and lobbying efforts in addition to standards development. In the field of information technology, the EIA has made significant contributions by defining physical connection interfaces and electronic signaling specifications for data communication.

3.3 Internet Standards

- An **Internet standard** is a thoroughly tested specification that is useful to and adhered to by those who work with the Internet. It is a formalized regulation that must be followed.

- There is a strict procedure by which a specification attains Internet standard status. A specification begins as an Internet draft.

- An **Internet draft** is a working document (a work in progress) with no official status and a 6-month lifetime. Upon recommendation from the Internet authorities, a draft may be published as a **Request for Comment** (RFC). Each RFC is edited, assigned a number, and made available to all interested parties.

- RFCs go through maturity levels and are categorized according to their requirement level.

4. THE OSI MODEL

The Open System Interconnection (OSI) reference model describes how information from a software application in one computer moves through a network medium to a software application in another computer.

The OSI reference model is a conceptual model composed of seven layers, each specifying particular network functions. The model was developed by the International Organization for Standardization (ISO) in 1984, and it is now considered the primary architectural model for inter-computer communications.

The OSI model divides the tasks involved with moving information between networked computers into seven smaller, more manageable task groups. A task or group of tasks is then assigned to each of the seven OSI layers.

Each layer is reasonably self-contained so that the tasks assigned to each layer can be implemented independently. This enables the solutions offered by one layer to be updated without adversely affecting the other layers.

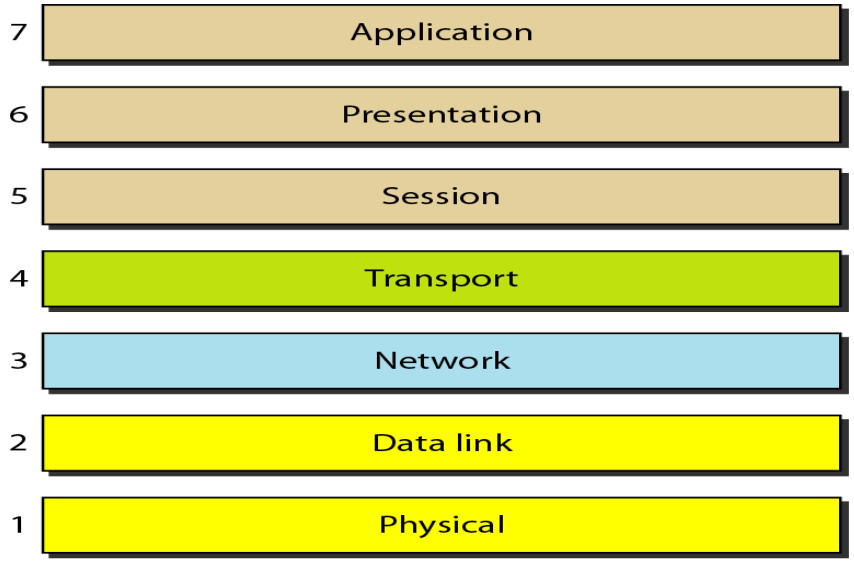


Fig: Seven Layers of OSI model

4.1 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). Below figure 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.

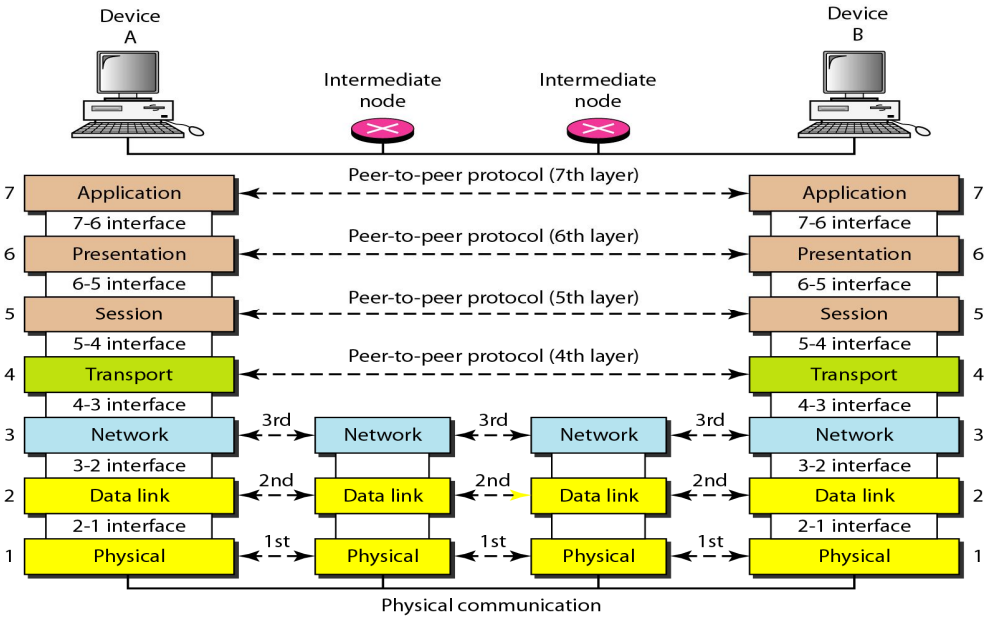


Fig: Layered Architecture of OSI model

4.2 Organization of the Layers

The seven layers can be thought of as belonging to three subgroups. Layers 1, 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 below figure , 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 layer 7 (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.

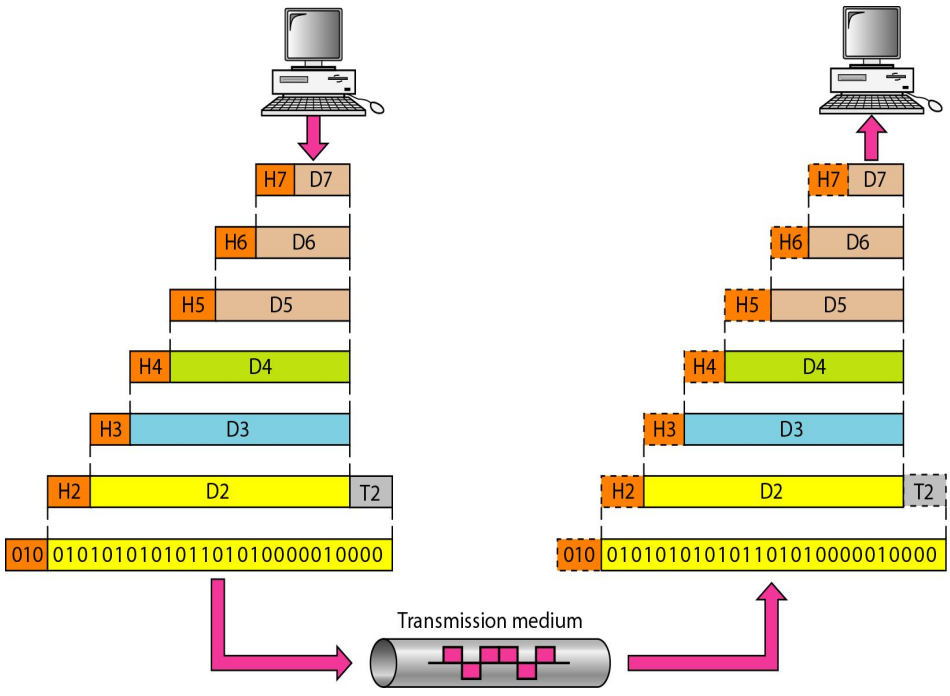


Fig: Encapsulation and decapsulation process

- 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

4.3 . LAYERS IN THE OSI MODEL

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

a. Physical Layer

- The physical layer coordinates the functions required to carry a bit stream over a physical

medium. It deals with the mechanical and electrical specifications of the interface and transmission medium.

- It also defines the procedures and functions that physical devices and interfaces have to perform for transmission to Occur. Below figure shows the position of the physical layer with respect to the transmission medium and the data link layer.

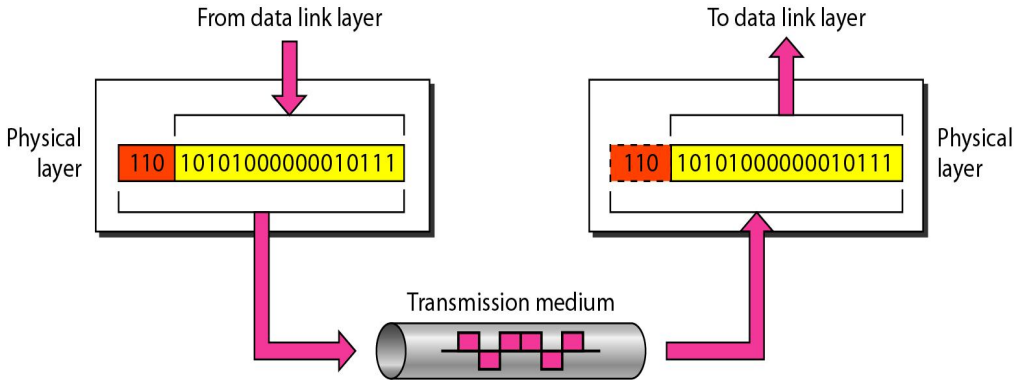


Fig: Physical Layer

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 0's or 1's) with no interpretation. To be transmitted, bits must be encoded into signals--electrical or optical. The physical layer defines the type of encoding (how 0's and 1's 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 duration 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.

b. Data Link Layer

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

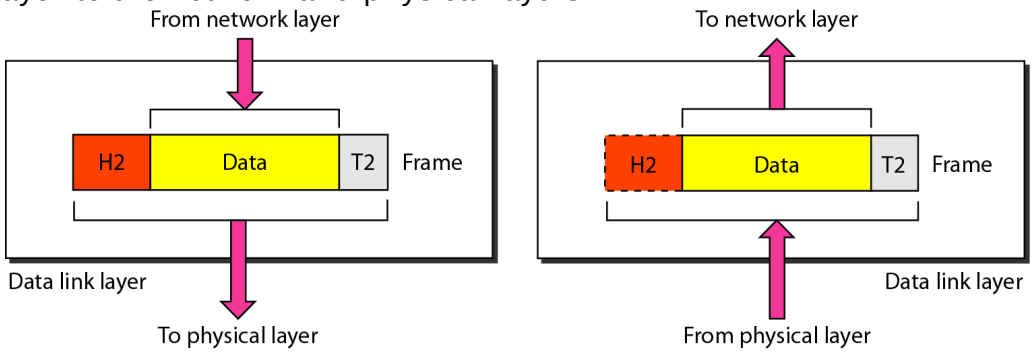


Fig: Data Link Layer

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 network to 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 mechanism 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.

Below figure illustrates hop-to-hop (node-to-node) delivery by the data link layer.

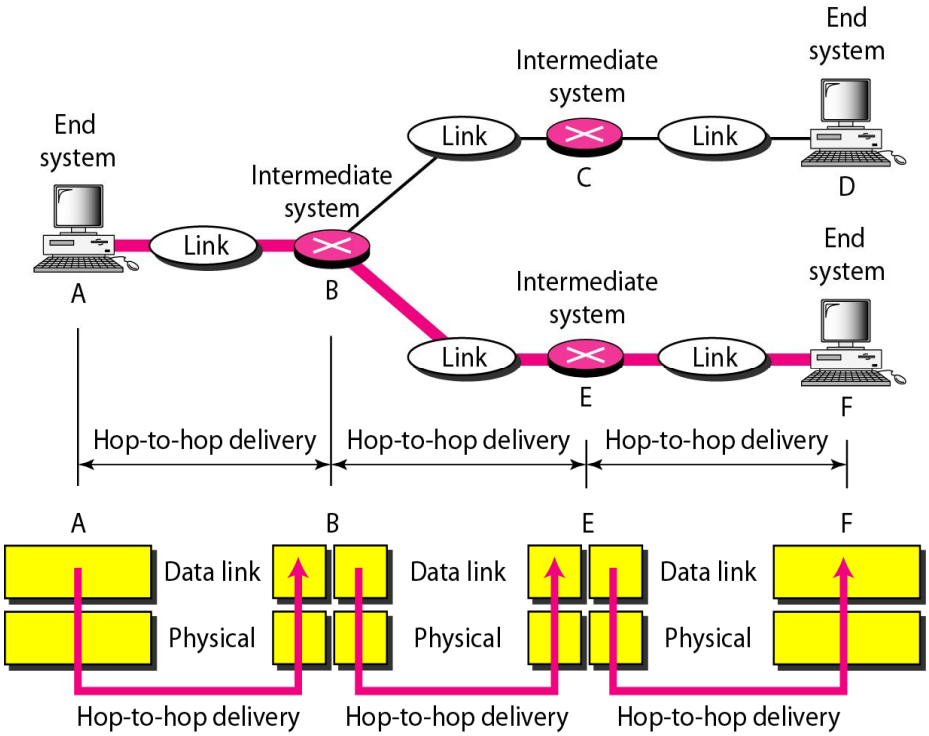


Fig: Delivery of data in the data link layer

- 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.

c. 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 the delivery 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 network 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. Below figure shows the relationship of the network layer to the data link and transport layers.

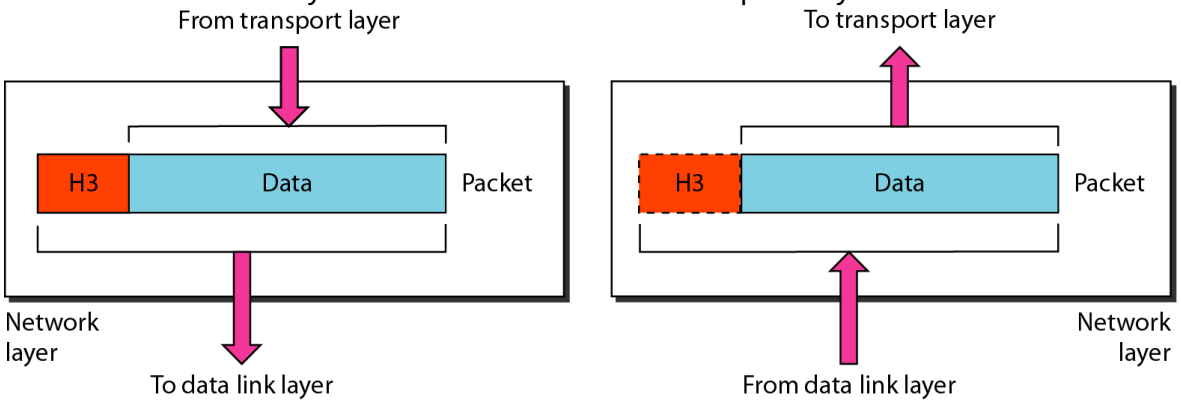


Fig: Network Layer

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.
- **Routing:** When independent networks or links are connected to create *inter networks* (network of networks) or a large network, the connecting devices (called *routers* or *switches)* route or switch the packets to their final destination. Below figure illustrates end-to-end delivery by the network layer

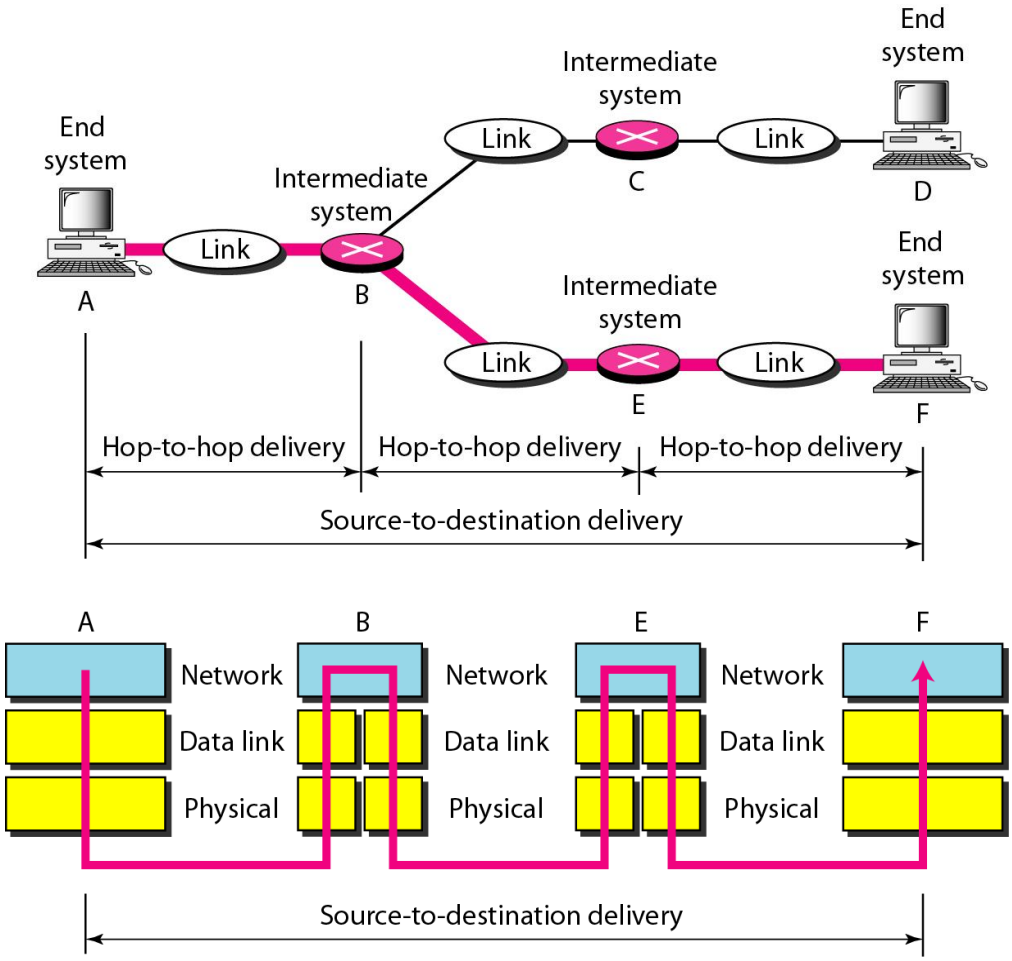


Fig: Source-to-destination delivery

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.

d. Transport Layer

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

figure shows the relationship of the transport layer to the network and session layers.

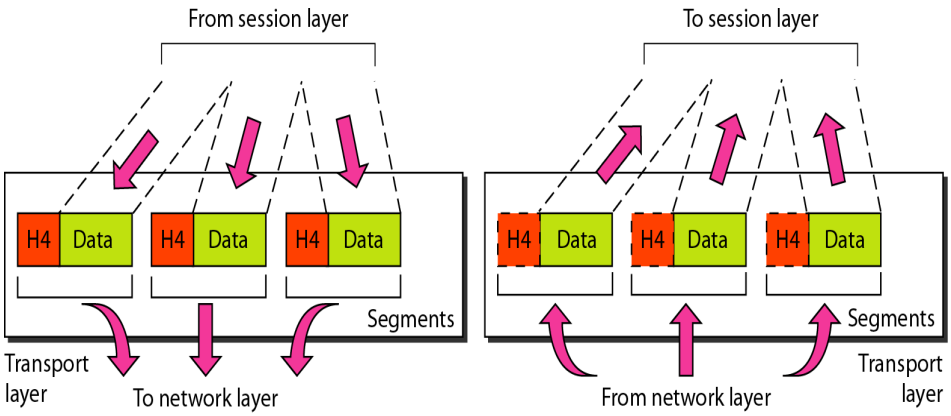


Fig: Transport Layer

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 connectionoriented. A connectionless transport layer treats each segment as an independent packet and delivers it to the transport layer at the destination machine. A connectionoriented 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.

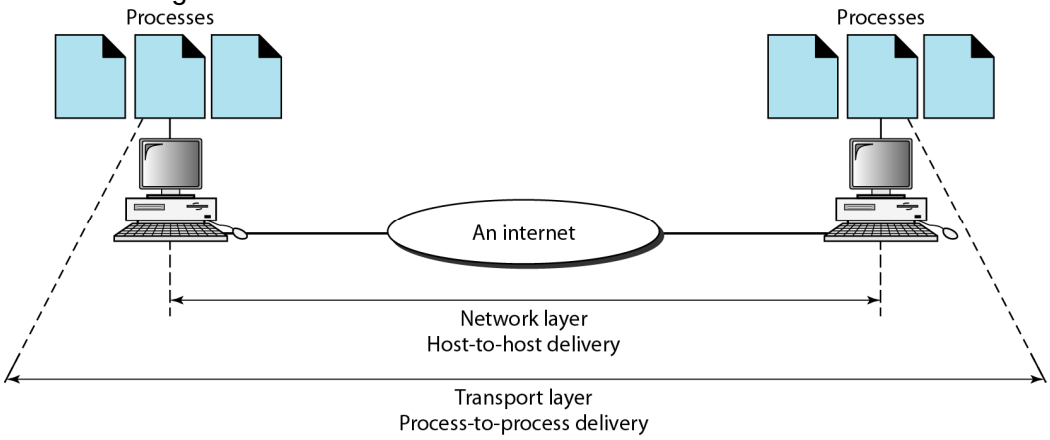


Fig: Reliable process-to-process delivery of a message

e. 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.

Specific responsibilities of the 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 halfduplex (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 synchronization points, to a stream of data. For example, if a system is sending a file of 2000 pages, it is advisable to insert checkpoints after every 100 pages to ensure that each 100-page unit is received and acknowledged independently. 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. Below figure illustrates the relationship of the session layer to the transport and presentation layers.

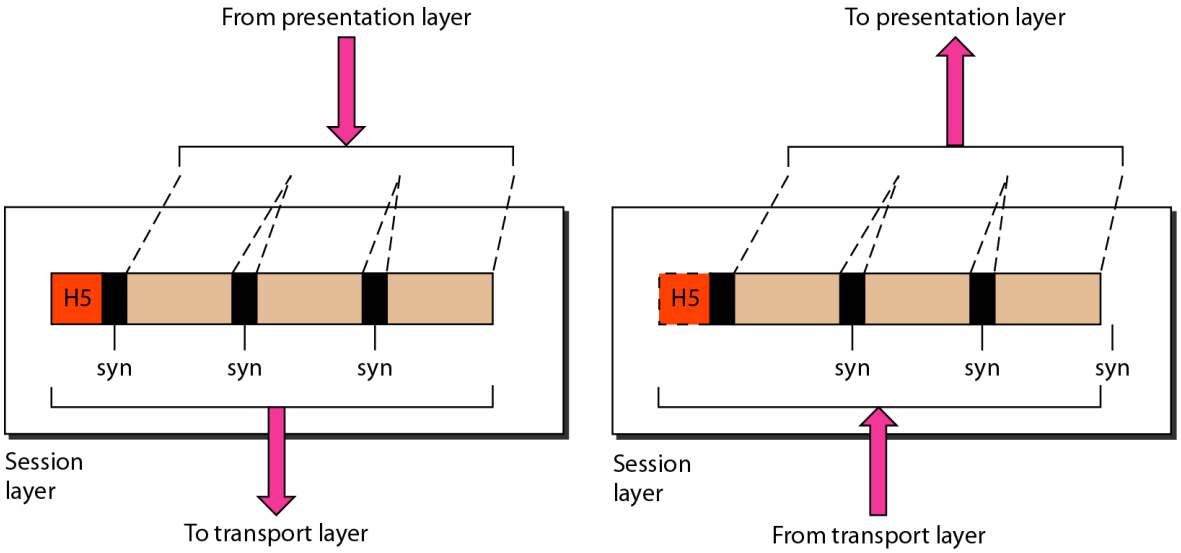


Fig: Session Layer

f. Presentation Layer

The presentation layer is concerned with the syntax and semantics of the information exchanged between two systems. Below figure shows the relationship between the presentation layer and the application and session layers.

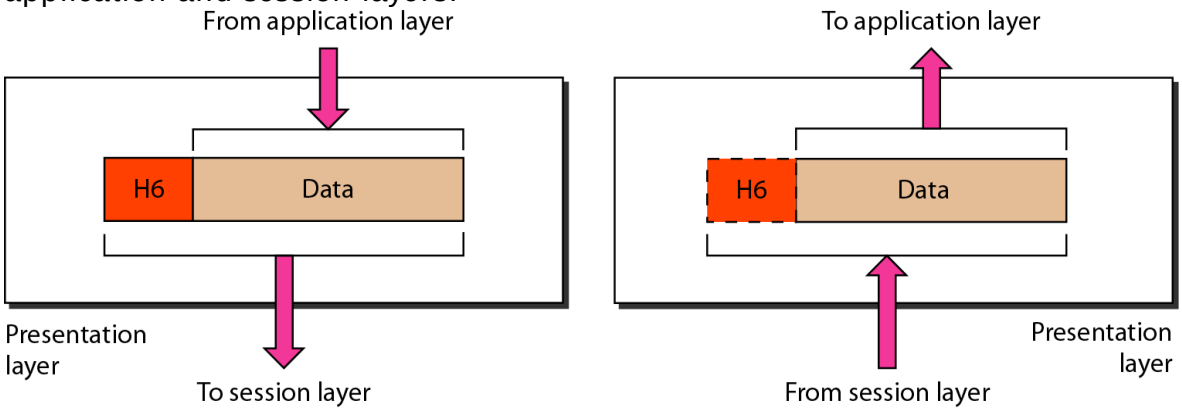


Fig: Presentation Layer

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.

g. Application Layer

- The application layer enables the user, whether human or software, to access the network.

- It provides user interfaces and support for services such as electronic mail, remote file access and transfer, shared database management, and other types of distributed information services.

- Below figure shows the relationship of the application layer to the user and the presentation 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.

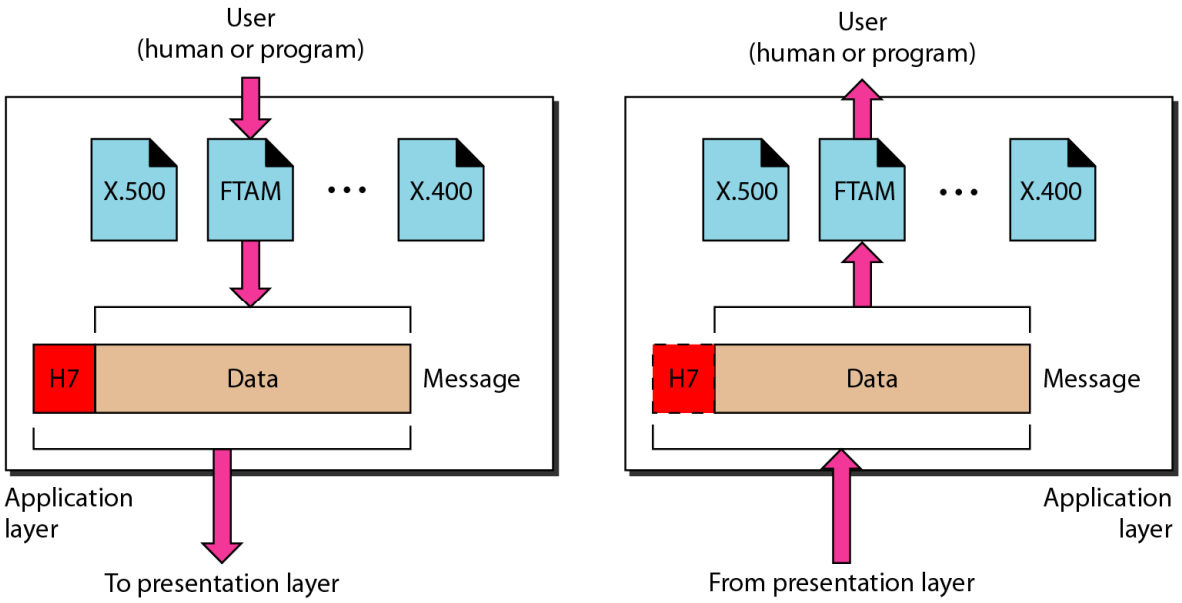


Fig: Application Layer

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 for forwarding and storage.
- **Directory services:** This application provides distributed database sources and access for global information about various objects and services.

5. TCP/IP PROTOCOL SUITE

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

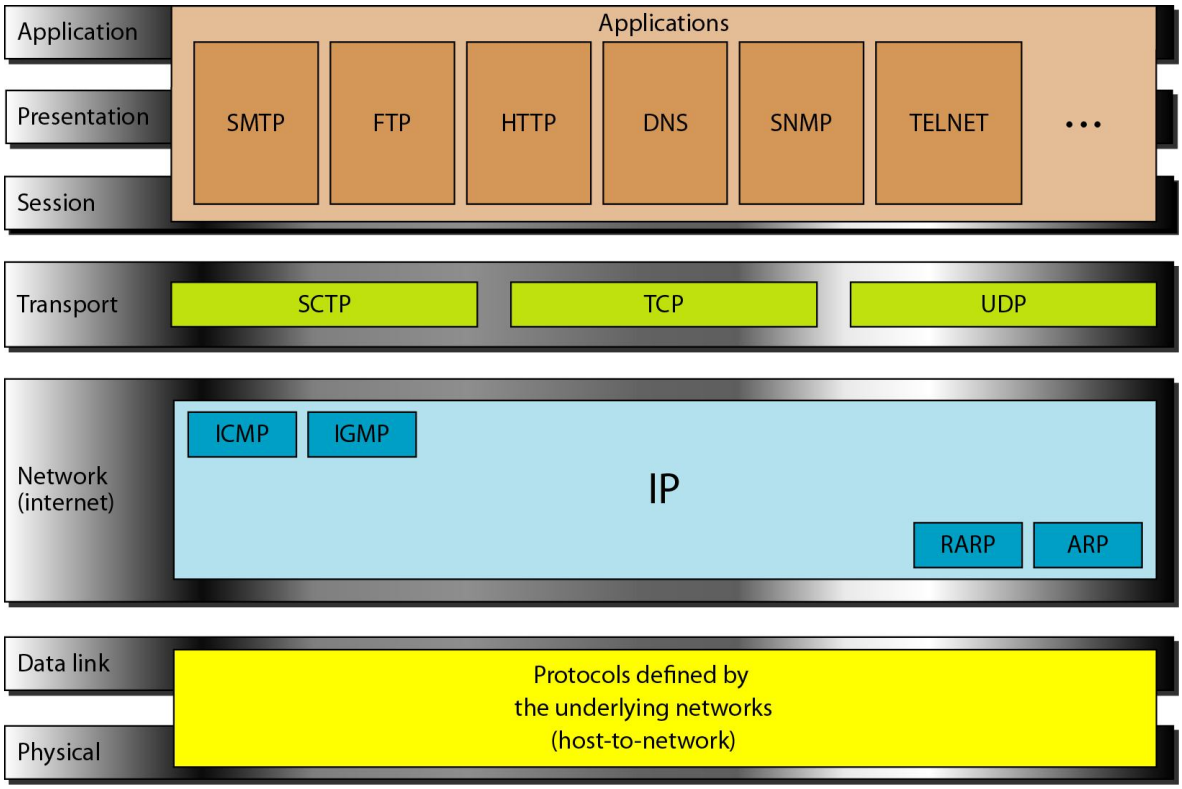


Fig: TCP/IP and OSI model

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

- The term *hierarchical* means that each upper-level protocol is supported by one or more lower-level protocols.
- 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.

5.1 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.

5.2 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.

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 through to its destination, but with no guarantees.

- IP transports data in packets called *datagrams,* each of which is transported separately.
- Datagrams can travel along different routes and can arrive out of sequence or be duplicated. IP does not keep track of the routes and has no facility for reordering datagrams once they arrive at their destination.
- The limited functionality of IP should not be considered a weakness, however. IP provides bare-bones transmission functions that free the user to add only those facilities necessary for a given application and thereby allows for maximum efficiency

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 link is identified by a physical or station address, usually imprinted on the network interface card (NIC).
- ARP is used to find the physical address of the node when its Internet address is known.

Reverse Address Resolution Protocol

- The Reverse Address Resolution Protocol (RARP) allows a host to discover its Internet address when it knows only its physical address.
- It is used when a computer is connected to a network for the first time or when a diskless computer is booted.

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.

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.

5.3 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.

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.

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 context,

means connection-oriented: A connection must be established between both ends of a transmission before either can transmit data.

- At the sending end of each transmission, TCP divides a stream of data into smaller units called *segments.* Each segment includes a sequence number for reordering after receipt, together with an acknowledgment number for the segments received.
- Segments are carried across the internet inside of IP datagrams. At the receiving end, TCP collects each datagram as it comes in and reorders the transmission based on sequence numbers.

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 combines the best features of UDP and TCP.

5.4 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.

---

6. 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 below figure).

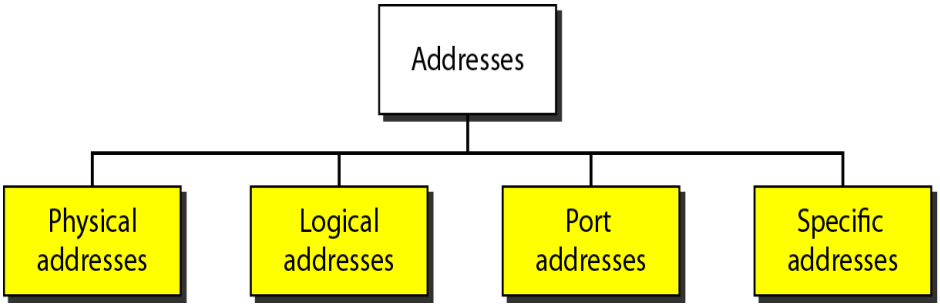


Fig: Types of addresses in TCP/IP

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

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.

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.

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 communication 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 communicates with computer B by using the File Transfer Protocol (FTP). For these processes 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.

Assignment-Cum-Tutorial Questions

SECTION-A

I. Objective Questions

1. The topology with highest reliability is [ ]
a)Bus topology b)Star topology
c) Ring Topology d) Mesh Topology

2. Protocols are [ ]
a) Agreements on how communication components and DTE's are to communicate
b) Logical communication channels for transferring data
c) Physical communication channels sued for transferring data
d) None of above

3. In OSI model dialogue control and token management are responsibilities of [ ]
a)Session Layer b)Network layer
c)Transport layer d)Data link layer

4. The method of communication in which transaction takes place in both directions, vice versa called [ ]
a) Simplex b) Four wire circuit
c) Full duplex d) half-duplex

5. Choose the best matching between Group 1 and Group 2. [ ]

Group-1	Group-2
P. Data link layer	1. Ensures reliable transport of data over a physical point-to-point link
Q. Network layer	2. Encoder/decodes data for physical transmission
R. Transport layer	3. Allows end-to-end communication between two processes
	4. Routes data from one network node to the next

a. P-1, Q-4, R-3 b. P-2, Q-4, R-1
c. P-2, Q-3, R-1 d. P-1, Q-3, R-2

6. In Internet protocol stack, when data is sent from device A to device B, the 5th layer to receive data at B is [ ]
a) Application layer b) Transport layer
c) Link layer d) Session layer

7. Which address is used in an internet employing the TCP/IP protocols[ ]
a) physical address and logical address b) port address
c) specific address d) all of the mentioned

8. Which of the following technique is used for encapsulation[ ]
a. a technique used in best-effort delivery systems to avoid endlessly looping packets

b. a technique used by protocols in which a lower level protocol accepts a message from a higher level protocol and places it in the data portion of the low level frame.
c. One of the pieces that results when an IP gateway divides an IP datagram into smaller pieces for transmission across a network that cannot handle the original datagram size
d. All of the above

9. Match the following : [ ]

List - I	List - II
(a) Data link layer	(i) Encryption
(b) Network layer	(ii) Connection control
(c) Transport layer	(iii) Routing
(d) Presentation layer	(iv) Framing

Code :

	(a)	(b)	(c)	(d)
(1)	(iv)	(iii)	(i)	(ii)
(2)	(iii)	(iv)	(ii)	(i)
(3)	(iv)	(ii)	(iii)	(i)
(4)	(iv)	(iii)	(ii)	(i)

10. Which of the following layer of OSI Reference model is also called end-to-end layer? [ ]
(1) Network layer (2) Datalink layer (3) Session layer (4) Transport layer

11. In TCP/IP Reference model, the job of ..... layer is to permit hosts to inject packets into any network and travel them independently to the destination. [ ]
(A) Physical (B) Transport (C) Application (D) Host-to-network

12. Match the following: [ ]

List – I	**List – II**
a. Session layer	i. Virtual terminal software
b. Application layer	ii. Semantics of the information transmitted
c. Presentation layer	iii. Flow control
d. Transport layer	iv. Manage dialogue control

Codes :

	a	b	c	d
(A)	iv	i	ii	iii
(B)	i	iv	ii	iii
(C)	iv	i	iii	ii
(D)	iv	ii	i	iii

13. Which of the following is not associated with the session layer [ ]
    (A) Dialog control (B) Token management
    (C) Semantics of the information transmitted
    (D) Synchronization
14. For n devices in a network, ... number of duplex-mode links are required for a mesh topology. [ ]
    (A) n(n + 1) (B) n (n – 1) (C) n(n + 1)/2 (D) n(n – 1)/2

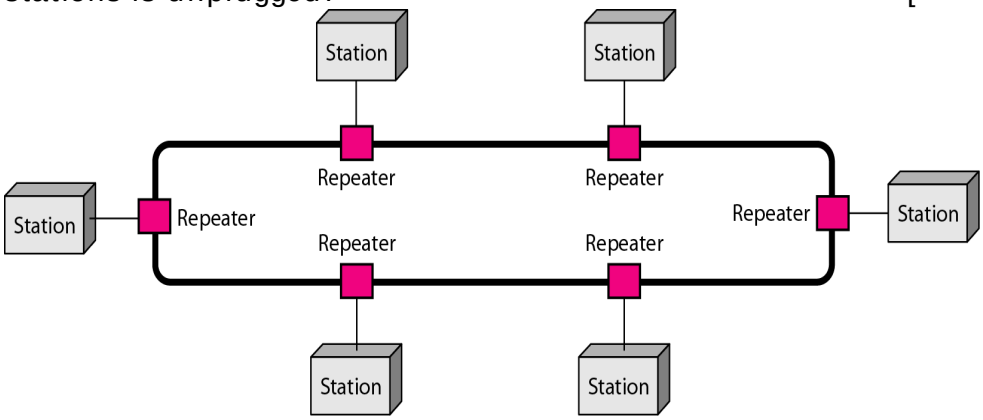
15. Match the following: [ ]

List – I	List - II
a. Physical Layer	i. Allow resources to network access
b. Datalink Layer	ii. Move packets from one destination to other
c. Network Layer	iii. Process to process message delivery
d. Transport Layer	iv. Transmission of bit stream
e. Application Layer	v. Formation of frames

Codes:

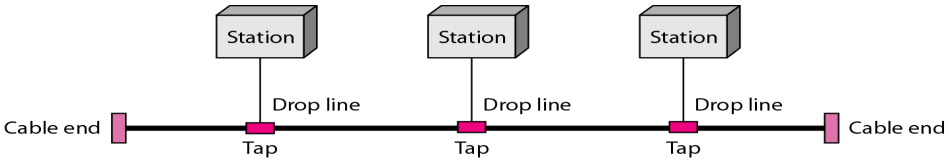
	a	b	c	d	e
(A)	iv	v	ii	iii	i
(B)	v	iv	i	ii	iii
(C)	i	iii	ii	v	iv
(D)	i	ii	iv	iii	v

16. In the ring topology in the given figure, what happens if one of the stations is unplugged? [ ]



a) entire network disturbed b) only that station communication effects
c) network performance improves d)neighbouring stations effects

17. In the bus topology in the given figure, what happens if one of the stations is unplugged? [ ]



a) entire network disturbed b) only that station communication effects
c) network performance improves d)neighbouring stations effects

18. Your company has a LAN in its downtown office and has set up a ain the manufacturing plant in the suburbs. To enable everyone to share data and resources between the two LANS, What type of device(s) are needed to connect them? [ ]

a) switch b)hub c)router d)modem

II.Descriptive Questions

1. Write about the characteristics of a data communications system.
2. Identify the five components of a data communications system.
3. Explain about network criteria in data communications system.
4. What is the difference between full duplex and half duplex transmission mode?
5. Name the four basic network topologies, and cite an advantage of each type.
6. Discuss the classification of Networks according to their size.
7. Describe the layered architecture of OSI model.
8. Draw and explain in detail about ISO-OSI reference model.
9. Draw and explain in detail about TCP/IP model.
10. Explain different standard organizations.
11. Compare and contrast OSI and TCP/IP models.
12. Differentiate between LAN, MAN and WAN.
13. You have two computers connected by an Ethernet hub at home? Is this a LAN, MAN or WAN? Explain with reasons.

C. GATE Questions

1. The protocol data unit(PDU) for the application layer in the Internet stack is [ ]

a. Segment b. Datagram c. Message d. Frame **[GATE 2012]**

2. In the following pairs of OSI protocol layer/sub-layer and its functionality, the INCORRECT pair is [ ]

a. Network layer and Routing

b. Data Link Layer and Bit synchronization

c. Transport layer and End-to-end process communication.

d. Medium Access Control sub-layer and Channel sharing **[GATE 2014].**

Physical Layer and Overview of PL Switching

Objectives:

Build an understanding of the fundamental concepts of multiplexing and switching.

Syllabus:

Physical layer and overview of PL Switching Multiplexing - Frequency division multiplexing, wave length division multiplexing, synchronous time division multiplexing, statistical time division multiplexing**, introduction to switching** - Circuit Switched Networks, Datagram Networks, Virtual Circuit Networks

Outcomes:

Students will be able to

- Identify the different types of multiplexing and their functions within a network
- Compare and contrast different multiplexing techniques
- Enumerate the techniques of switching
- Compare different switching techniques
- Explain the concept of switching, and identify and analyze the different types of delay in switched networks

Learning Material

2.1 MULTIPLEXING

Whenever the bandwidth of a medium linking two devices is greater than the bandwidth needs of the devices, the link can be shared. **Multiplexing is the set of techniques that allows the simultaneous transmission of multiple signals across a single data link**.

In a multiplexed system, *n* lines share the bandwidth of one link. Figure shows the basic format of a multiplexed system. The lines on the left direct their transmission streams to a multiplexer (MUX), which combines them into a single stream (many-to one).

At the receiving end, that stream is fed into a demultiplexer(DEMUX), which separates the stream back into its component transmissions (one-to-many) and directs them to their corresponding lines. In the figure, the word link refers to the physical path. The word channel refers to the portion of a link that carries a transmission between a given pair of lines. One link can have many *(n)* channels.

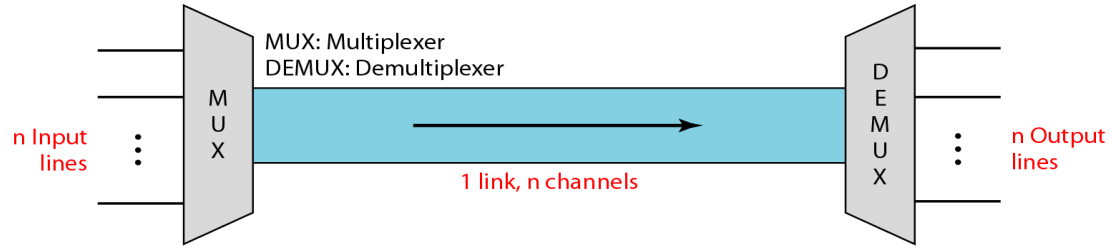


Figure: Dividing a link into channels

There are three basic multiplexing techniques: frequency-division multiplexing, wavelength-division multiplexing, and time-division multiplexing. The first two are techniques designed for analog signals, the third, for digital signals.

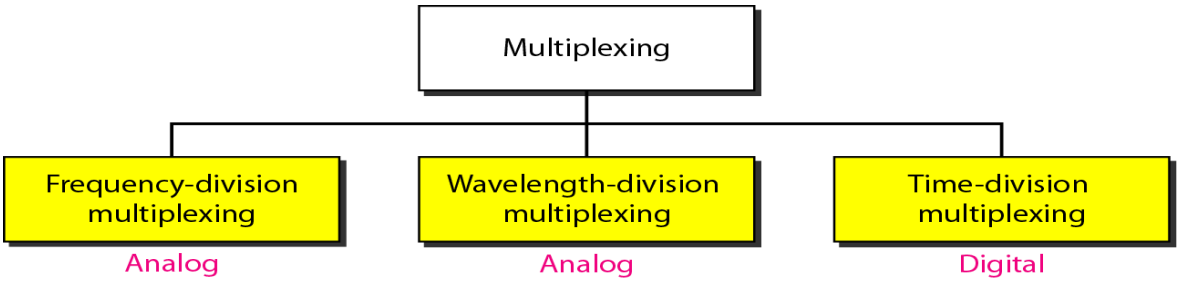


Figure:Categories of multiplexing

2.1.1 Frequency-Division Multiplexing

Frequency-division multiplexing (FDM) is an analog technique that can be applied when the bandwidth of a link (in hertz) is greater than the combined bandwidths of the signals to be transmitted. In FDM, signals generated by each sending device modulate different carrier frequencies.

- These modulated signals are then combined into a single composite signal that can be transported by the link. Carrier frequencies are separated by sufficient bandwidth to accommodate the modulated signal. These bandwidth ranges are the channels through which the various signals travel. Channels can be separated by strips of unused bandwidth-guard bands-to prevent signals from overlapping. In addition, carrier frequencies must not interfere with the original data frequencies.

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

Figure:FDM

- We consider FDM to be an analog multiplexing technique; however, this does not mean that FDM cannot be used to combine sources sending digital signals. A digital signal can be converted to an analog signal before FDM is used to multiplex them.

Multiplexing Process

Figure is a conceptual illustration of the multiplexing process. Each source generates a signal of a similar frequency range. Inside the multiplexer, these similar signals modulates different carrier frequencies *(f1,f2,* and *f3).* The resulting modulated signals are then combined into a single composite signal that is sent out over a media link that has enough bandwidth to accommodate it.

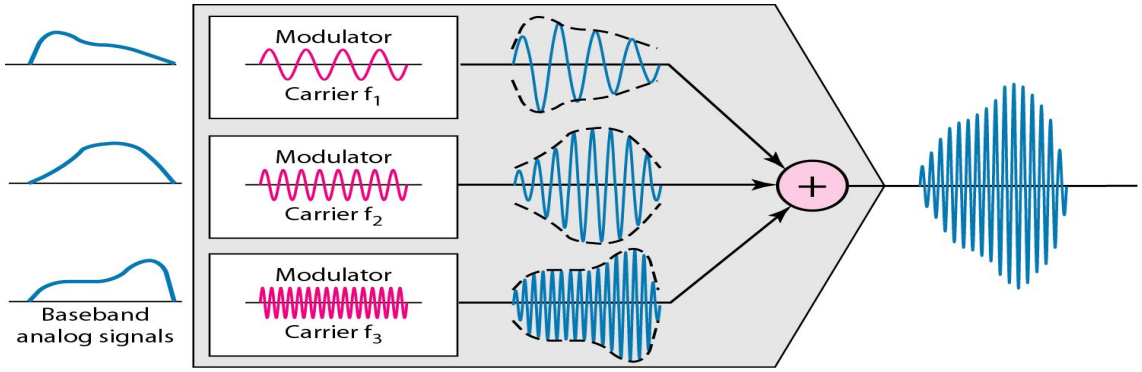


Figure : multiplexing process

Demultiplexing Process

The demultiplexer uses a series of filters to decompose the multiplexed signal into its constituent component signals. The individual signals are then passed to a demodulator that separates them from their carriers and passes them to the output lines.

- .

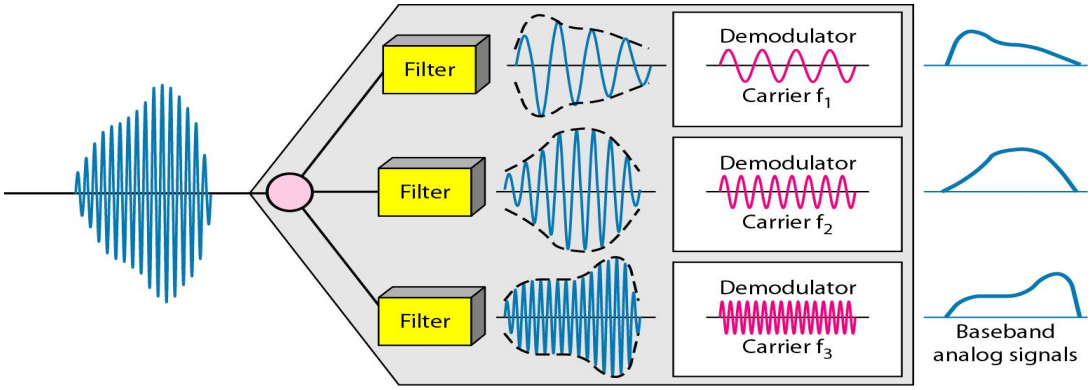


Figure : demultiplexing process

The Analog Carrier System

To maximize the efficiency of their infrastructure, telephone companies have traditionally multiplexed signals from lower-bandwidth lines onto higher-bandwidth lines. In this way, many switched or leased lines can be combined into fewer but bigger channels. For analog lines, FDM is used.

One of these hierarchical systems used by AT&T is made up of groups, supergroups, master groups, and jumbo groups.

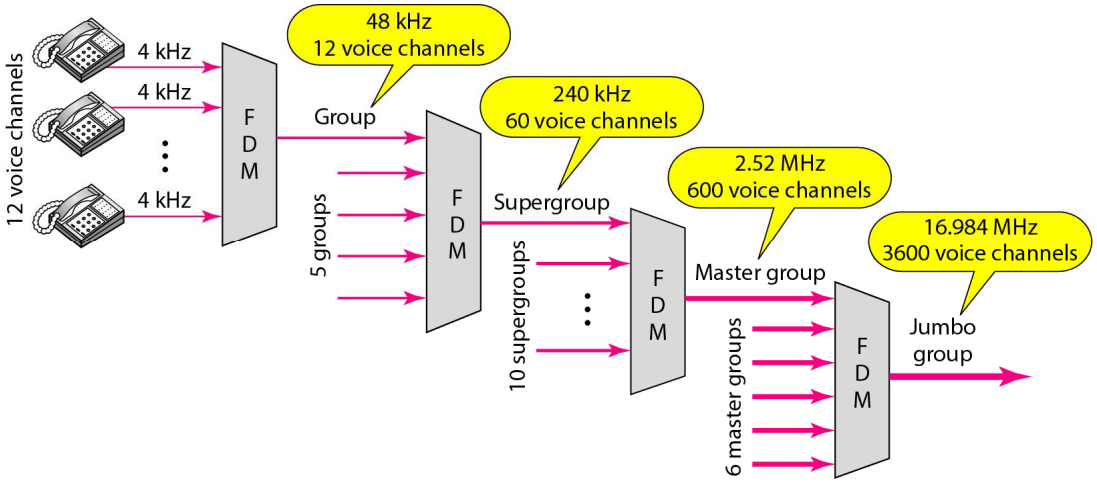


Figure: Analog hierarchy

In this analog hierarchy, 12 voice channels are multiplexed onto a higher-bandwidth line to create a group.

- A group has 48 kHz of bandwidth and supports 12 voice channels. At the next level, up to five groups can be multiplexed to create a composite signal called a supergroup.

- A supergroup has a bandwidth of 240 kHz and supports up to 60 voice channels. Supergroups can be made up of either five groups or 60 independent voice channels.

- At the next level, 10 supergroups are multiplexed to create a master group. A master group must have 2.40 MHz of bandwidth, but the need for guard bands between the supergroups increases the necessary bandwidth to 2.52 MHz. Master groups support up to 600 voice channels.

- Finally, six master groups can be combined into a jumbo group. A jumbo group must have 15.12 MHz (6 x 2.52 MHz) but is augmented to 16.984 MHz to allow for guard bands between the master groups.

Other Applications of FDM

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

- Another common use of FDM is in television broadcasting. Each TV channel has its own bandwidth of 6 MHz.
- The first generation of cellular telephones (still in operation) also uses FDM. Each user is assigned two 30-kHz channels, one for sending voice and the other for receiving.

2.1.2 Wavelength-Division Multiplexing

Multiplexing Wavelength-division multiplexing (WDM) is designed to use the high-data-rate capability of fiber-optic cable. The optical fiber data rate is higher than the data rate of metallic transmission cable.

- Using a fiber-optic cable for one single line wastes the available bandwidth. Multiplexing allows us to combine several lines into one.
- WDM is conceptually the same as FDM, except that the multiplexing and demultiplexing involve optical signals transmitted through fiber-optic channels. The idea is the same: We are combining different signals of different frequencies. The difference is that the frequencies are very high.
- Figure gives a conceptual view of a WDM multiplexer and demultiplexer. Very narrow bands of light from different sources are combined to make a wider band of light. At the receiver, the signals are separated by the demultiplexer.

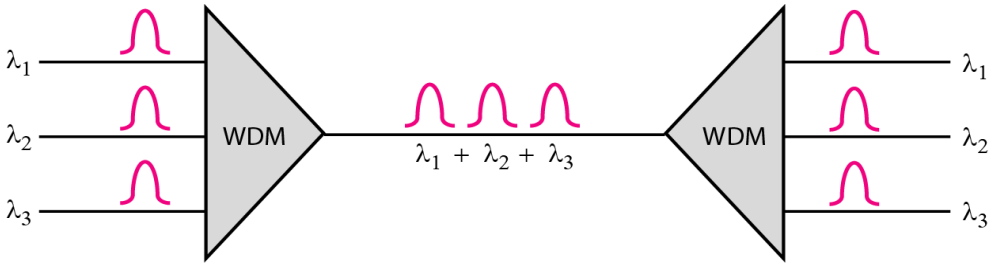


Figure: Wavelength-division multiplexing

- Although WDM technology is very complex, the basic idea is very simple. We want to combine multiple light sources into one single light at the multiplexer and do the reverse at the demultiplexer.
- The combining and splitting of light sources are easily handled by a prism. Recall from basic physics that a prism bends a beam of light based on the angle of incidence and the frequency.
- Using this technique, a multiplexer can be made to combine several input beams of light, each containing a narrow band of frequencies, into one output beam of a wider band of frequencies. A demultiplexer can also be made to reverse the process.

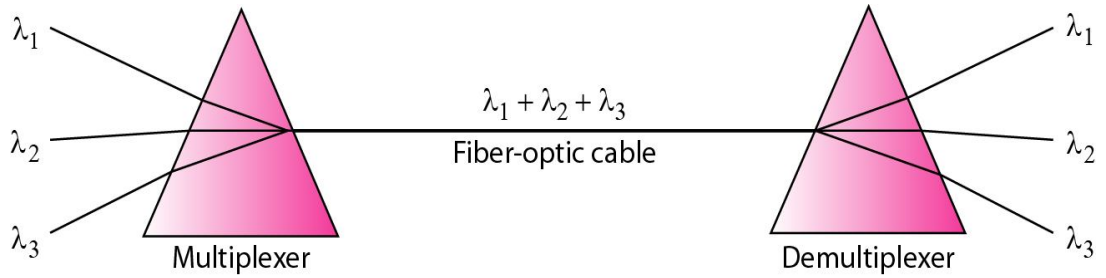


Figure: Prisms in wavelength-division multiplexing and demultiplexing

Applications of WDM:

One application of WDM is the SONET network in which multiple optical fiber lines are multiplexed and demultiplexed.

- A new method, called dense WDM (DWDM), can multiplex a very large number of channels by spacing channels very close to one another. It achieves even greater efficiency.

2.1.3 Synchronous Time-Division Multiplexing

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

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

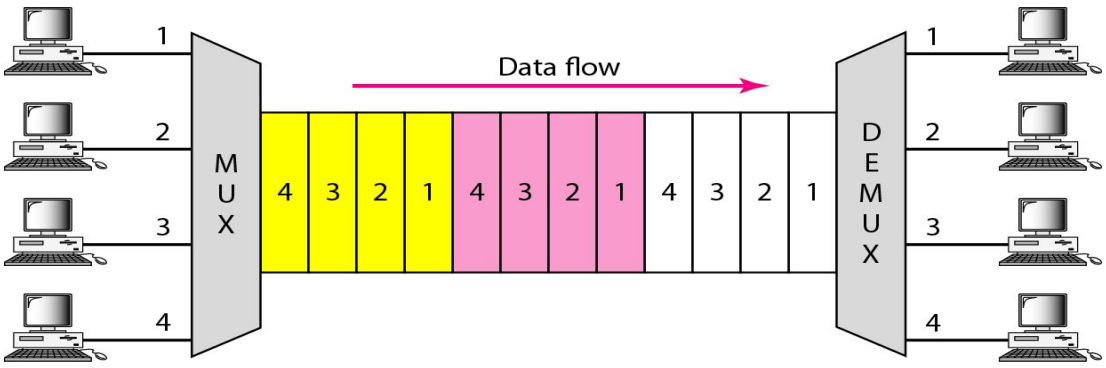


Figure:TDM

Note that, in Figure we are concerned with only multiplexing, not switching. This means that all the data in a message from source 1 always go to one specific destination, be it 1, 2, 3, or 4. The delivery is fixed and unvarying,

unlike switching. We also need to remember that TDM is, in principle, a digital multiplexing technique.

TDM is, in principle, a digital multiplexing technique. Digital data from different sources are combined into one timeshared link. However, this does not mean that the sources cannot produce analog data; analog data can be sampled, changed to digital data, and then multiplexed by using TDM.

We can divide TDM into two different schemes: synchronous and statistical. In synchronous TDM, each input connection has an allotment in the output even if it is not sending data.

Time Slots and Frames

In synchronous TDM, the data flow of each input connection is divided into units, where each input occupies one input time slot. A unit can be 1 bit, one character, or one block of data. Each input unit becomes one output unit and occupies one output time slot. However, the duration of an output time slot is *n* times shorter than the duration of an input time slot. If an input time slot is *T* s, the output time slot is *T*/*n* s, where *n* is the number of connections. In other words, a unit in the output connection has a shorter duration; it travels faster. Figure shows an example of synchronous TDM where *n* is 3.

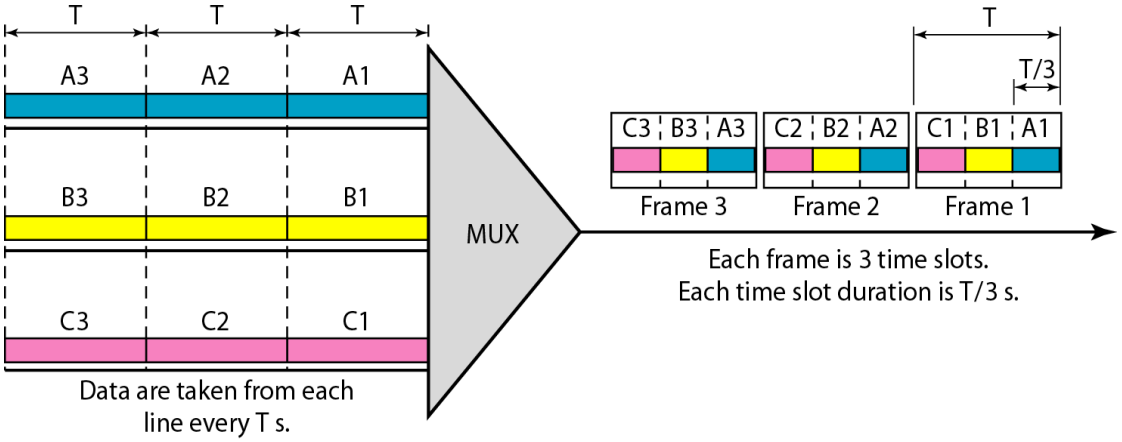


Figure: Synchronous TDM

- In synchronous TDM, each input connection has an allotment in the output even if it is not sending data. Time Slots and Frames In synchronous TDM, the data flow of each input connection is divided into units, where each input occupies one input time slot. A unit can be 1 bit, one character, or one block of data. Each input unit becomes one output unit and occupies one output time slot.
- However, the duration of an output time slot is n times shorter than the duration of an input time slot. If an input time slot is T s, the

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

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

- The data rate of the output link must be n times the data rate of a connection to guarantee the flow of data. In above figure, the data rate of the link is 3 times the data rate of a connection; likewise, the duration of a unit on a connection is 3 times that of the time slot (duration of a unit on the link). In the figure, we represent the data prior to multiplexing as 3 times the size of the data after multiplexing. This is just to convey the idea that each unit is 3 times longer in duration before multiplexing than after. Time slots are grouped into frames. A frame consists of one complete cycle of time slots, with one slot dedicated to each sending device. In a system with n input lines, each frame has n slots, with each slot allocated to carrying data from a specific input line.

Interleaving:

TDM can be visualized as two fast-rotating switches, one on the multiplexing side and the other on the demultiplexing side. The switches are synchronized and rotate at the same speed, but in opposite directions.

- On the multiplexing side, as the switch opens in front of a connection, that connection has the opportunity to send a unit onto the path. This process is called interleaving. On the demultiplexing side, as the switch opens in front of a connection, that connection has the opportunity to receive a unit from the path.

- Below Figure 8 shows the interleaving process for the connection shown in above figure. In this figure, we assume that no switching is involved and that the data from the first connection at the multiplexer site go to the first connection at the demultiplexer

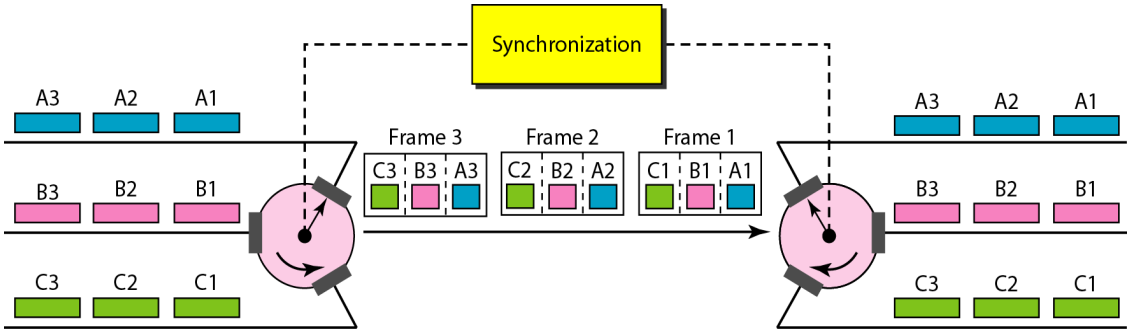


Figure: Interleaving

Empty Slots

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

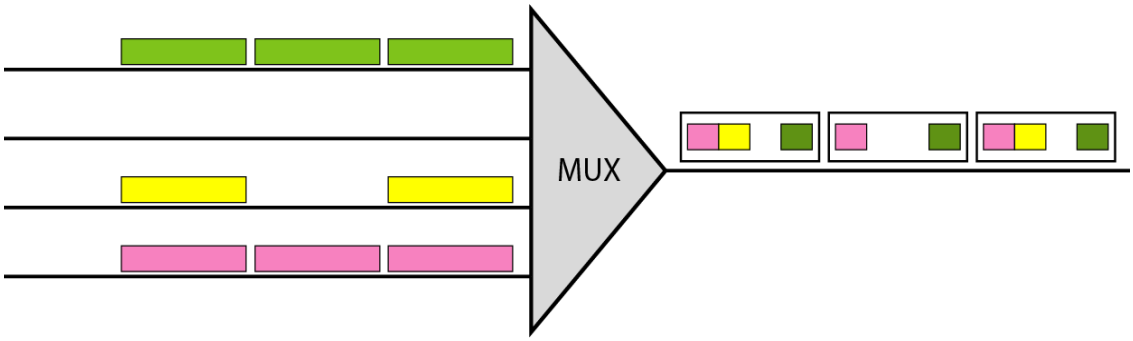


Figure: Empty Slots

Data Rate Management

One problem with TDM is how to handle a disparity in the input data rates. In all our discussion so far, we assumed that the data rates of all input lines were the same. However, if data rates are not the same, three strategies, or a combination of them, can be used. We call these three strategies **multilevel multiplexing, multiple-slot allocation,** and **pulse stuffing.**

Multilevel Multiplexing Multilevel multiplexing is a technique used when the data rate of an input line is a multiple of others. For example, in below figure ,we have two inputs of 20 kbps and three inputs of 40 kbps. The first two input lines can be multiplexed together to provide a data rate equal to the last three. A second level of multiplexing can create an output of 160 kbps.

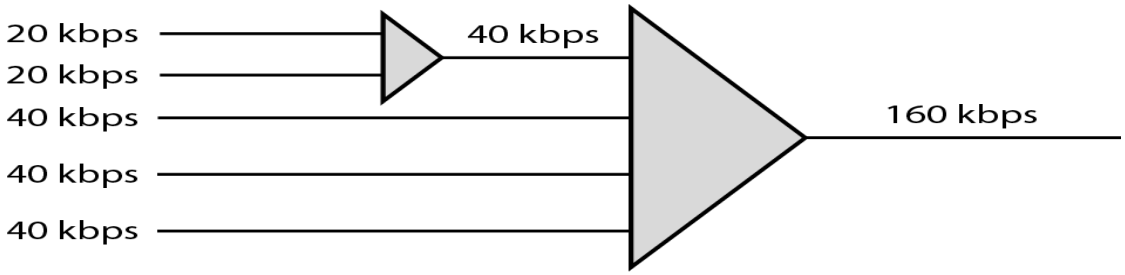


Figure: Multilevel Multiplexing

Multiple-Slot Allocation Sometimes it is more efficient to allot more than one slot in a frame to a single input line. For example, we might have an input line that has a data rate that is a multiple of another input. In below figure, the input line with a 50-kbps data rate can be given two slots in the output. We insert a serial-to-parallel converter in the line to make two inputs out of one.

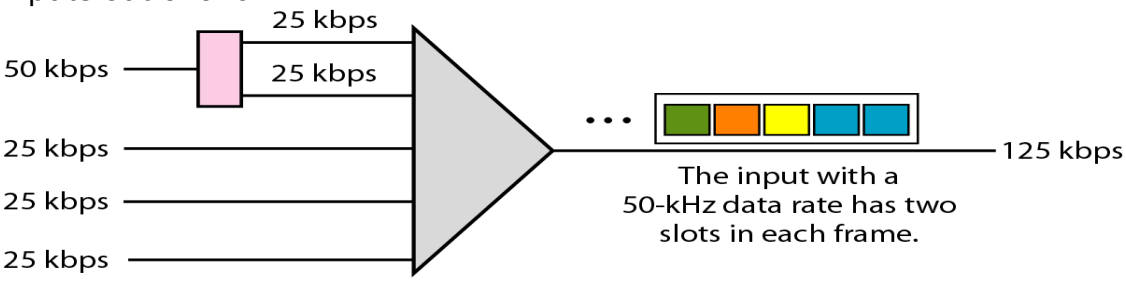


Figure: Multiple-Slot Allocation

Pulse Stuffing Sometimes the bit rates of sources are not multiple integers of each other. Therefore, neither of the above two techniques can be applied. One solution is to make the highest input data rate the dominant data rate and then add dummy bits to the input lines with lower rates. This will increase their rates. This technique is called pulse stuffing, bit padding, or bit stuffing. The idea is shown in below figure. The input with a data rate of 46 is pulse-stuffed to increase the rate to 50 kbps. Now multiplexing can take place.

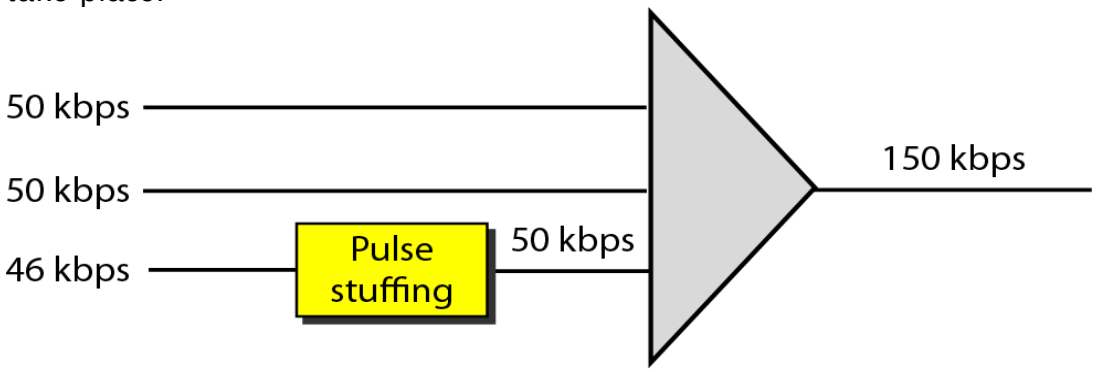


Figure: pulse stuffing

Frame Synchronizing

The implementation ofTDM is not as simple as that of FDM. Synchronization between the multiplexer and demultiplexer is a major issue. If the multiplexer and the demultiplexer are not synchronized, a bit belonging to one channel may be received by the wrong channel.

For this reason, one or more synchronization bits are usually added to the beginning of each frame. These bits, called framing bits, follow a pattern, frame to frame, that allows the demultiplexer to synchronize with the incoming stream so that it can separate the time slots accurately.

In most cases, this synchronization information consists of 1 bit per frame, alternating between 0 and 1, as shown in figure.

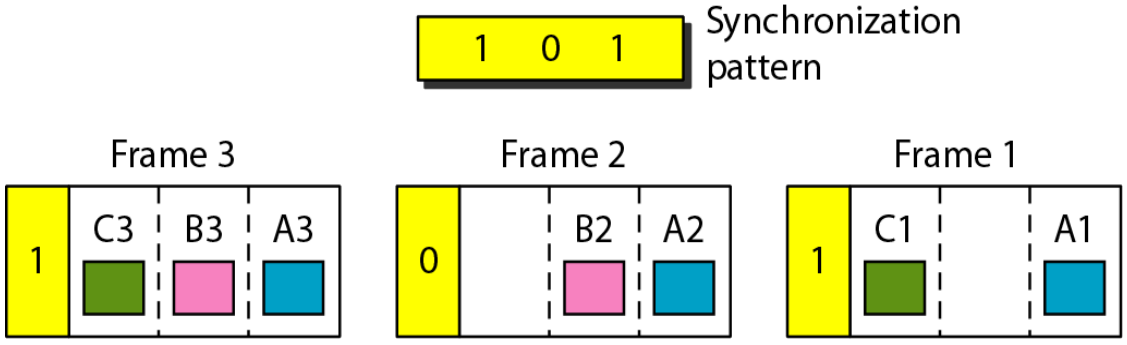


Figure: Framing bits

Digital Signal Service

Telephone companies implement TDM through a hierarchy of digital signals, called digital signal (DS) service or digital hierarchy. Figure shows the data rates supported by each level.

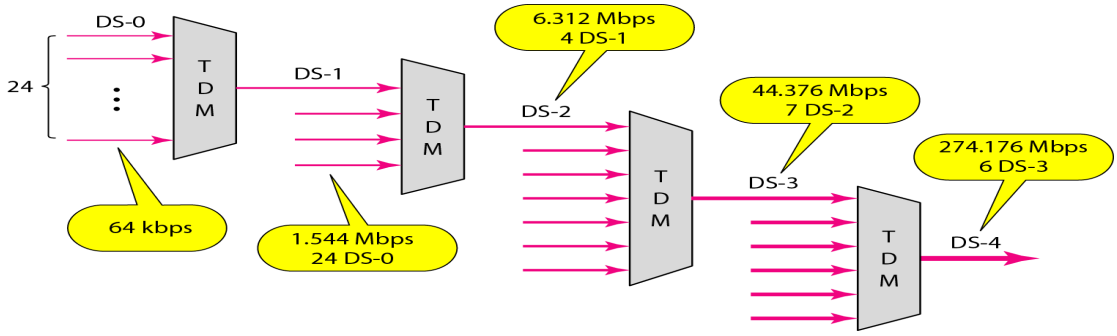


Figure: Digital hierarchy

- A **DS-0** service is a single digital channel of 64 kbps.
- **DS-**1 is a 1.544-Mbps service; 1.544 Mbps is 24 times 64 kbps plus 8 kbps of overhead. It can be used as a single service for 1.544-Mbps transmissions, or it can be used to multiplex 24 DS-0 channels or to carry any other combination desired by the user that can fit within its 1.544-Mbps capacity.

- **DS-2** is a 6.312-Mbps service; 6.312 Mbps is 96 times 64 kbps plus 168 kbps of overhead. It can be used as a single service for 6.312-Mbps transmissions; or it can be used to multiplex 4 DS-l channels, 96 DS-0 channels, or a combination of these service types.
- **DS-3** is a 44.376-Mbps service; 44.376 Mbps is 672 times 64 kbps plus 1.368 Mbps of overhead. It can be used as a single service for 44.376-Mbps transmissions; or it can be used to multiplex 7 DS-2 channels, 28 DS-l channels, 672 DS-0 channels, or a combination of these service types.
- **DS-4** is a 274. 176-Mbps service; 274.176 is 4032 times 64 kbps plus 16.128 Mbps of overhead. It can be used to multiplex 6 DS-3 channels, 42 DS-2 channels, 168 DS-l channels, 4032 DS-0 channels, or a combination of these service types.

Synchronous TDM Applications:

- Some second-generation cellular telephone companies use synchronous TDM.

2.1.4 Statistical Time-Division Multiplexing

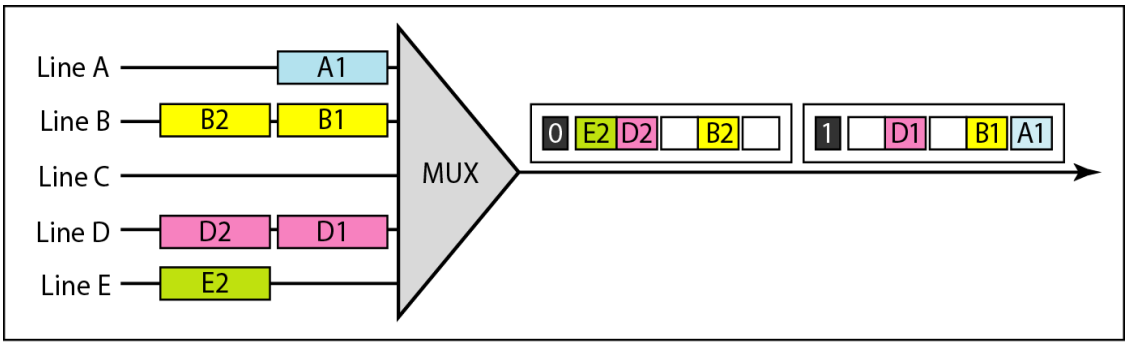
In synchronous TDM, each input has a reserved slot in the output frame. This can be inefficient if some input lines have no data to send. In statistical time-division multiplexing, slots are dynamically allocated to improve bandwidth efficiency. Only when an input line has a slot's worth of data to send is it given a slot in the output frame. In statistical multiplexing, the number of slots in each frame is less than the number of input lines. The multiplexer checks each input line in roundrobin fashion; it allocates a slot for an input line if the line has data to send; otherwise, it skips the line and checks the next line.
Below figure hows a synchronous and a statistical TDM example. In the former, some slots are empty because the corresponding line does not have data to send. In the latter, however, no slot is left empty as long as there are data to be sent by any input line.

Addressing:

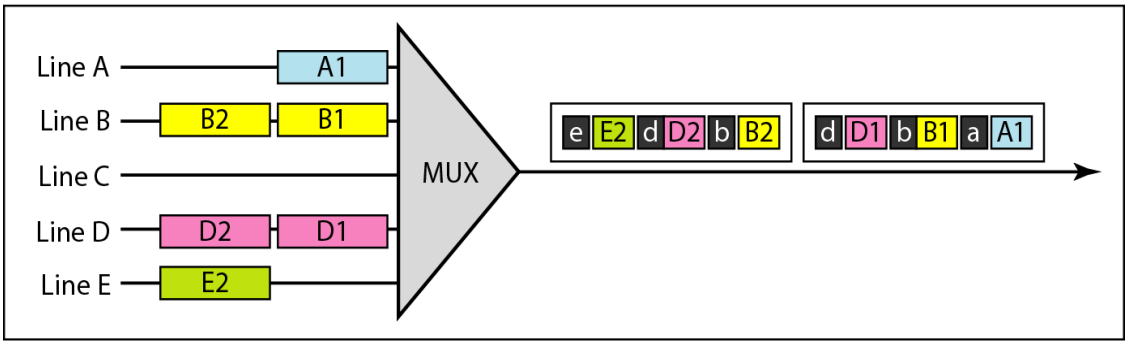
Figure a also shows a major difference between slots in synchronous TDM and statistical TDM. An output slot in synchronous TDM is totally occupied by data; in statistical TDM, a slot needs to carry data as well as the address of the destination. In synchronous TDM, there is no need for addressing; synchronization and preassigned relationships between the inputs and outputs serve as an address.

- We know, for example, that input 1 always goes to input 2. If the multiplexer and the demultiplexer are synchronized, this is guaranteed. In statistical multiplexing, there is no fixed relationship between the inputs and outputs because there are no preassigned or reserved slots.

- We need to include the address of the 35 receiver inside each slot to show where it is to be delivered. The addressing in its simplest form can be n bits to define N different output lines with n =10g$_2$ N. For example, for eight different output lines, we need a 3-bit address.



a. Synchronous TDM



b. Statistical TDM

Figure: TDM slot comparison

Slot Size

Since a slot carries both data and an address in statistical TDM, the ratio of the data size to address size must be reasonable to make transmission efficient. For example, it would be inefficient to send 1 bit per slot as data when the address is 3 bits.

- This would mean an overhead of 300 percent. In statistical TDM, a block of data is usually many bytes while the address is just a few bytes.

No Synchronization Bit

There is another difference between synchronous and statistical TDM, but this time it is at the frame level. The frames in statistical TDM need not be synchronized, so we do not need synchronization bits.

Bandwidth

In statistical TDM, the capacity of the link is normally less than the sum of the capacities of each channel. The designers of statistical TDM define the capacity of the link based on the statistics of the load for each channel.

- If on average only *x* percent of the input slots are filled, the capacity of the link reflects this. Of course, during peak times, some slots need to wait.

2.2 SWITCHING

A network is a set of connected devices. Whenever we have multiple devices, we have the problem of how to connect them to make one-to-one communication possible. One solution is to make a point-to-point connection between each pair of devices (a mesh topology) or between a central device and every other device (a star topology). These methods, however, are impractical and wasteful when applied to very large networks. The number and length of the links require too much infrastructure to be cost-efficient, and the majority of those links would be idle most of the time.

- Other topologies employing multipoint connections, such as a bus, are ruled out because the distances between devices and the total number of devices increase beyond the capacities of the media and equipment.
- A better solution is switching. A switched network consists of a series of interlinked nodes, called switches. Switches are devices capable of creating temporary connections between two or more devices linked to the switch.

- In a switched network, some of these nodes are connected to the end systems (computers or telephones, for example). Others are used only for routing. Figure shows a switched network.

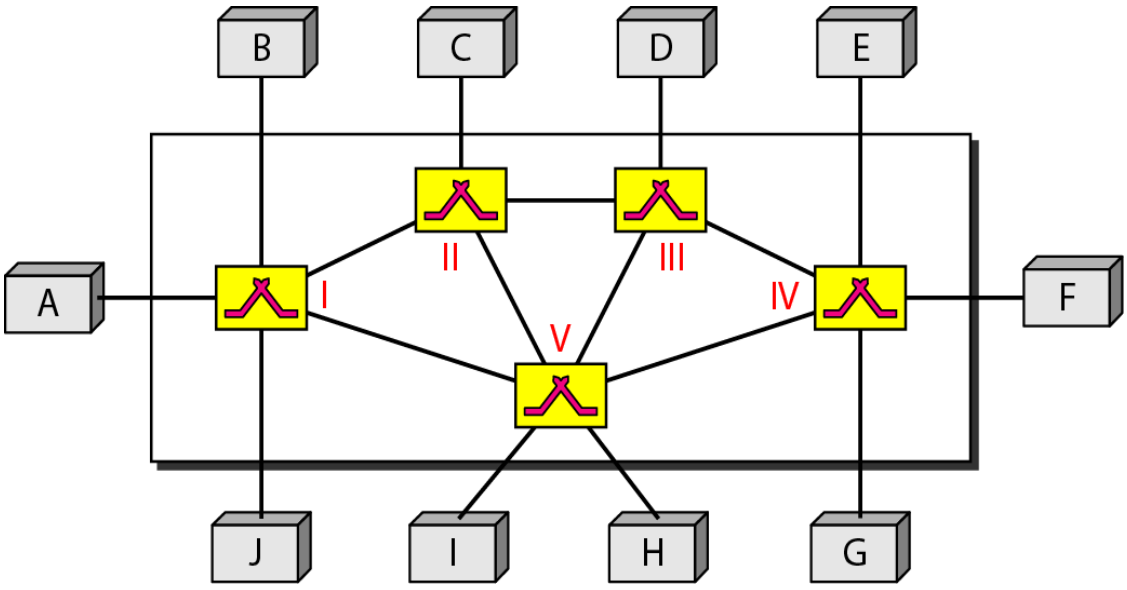


Figure: Switched network

- The end systems (communicating devices) are labeled A, B, C, D, and so on, and the switches are labeled I, II, III, IV, and V. Each switch is connected to multiple links.

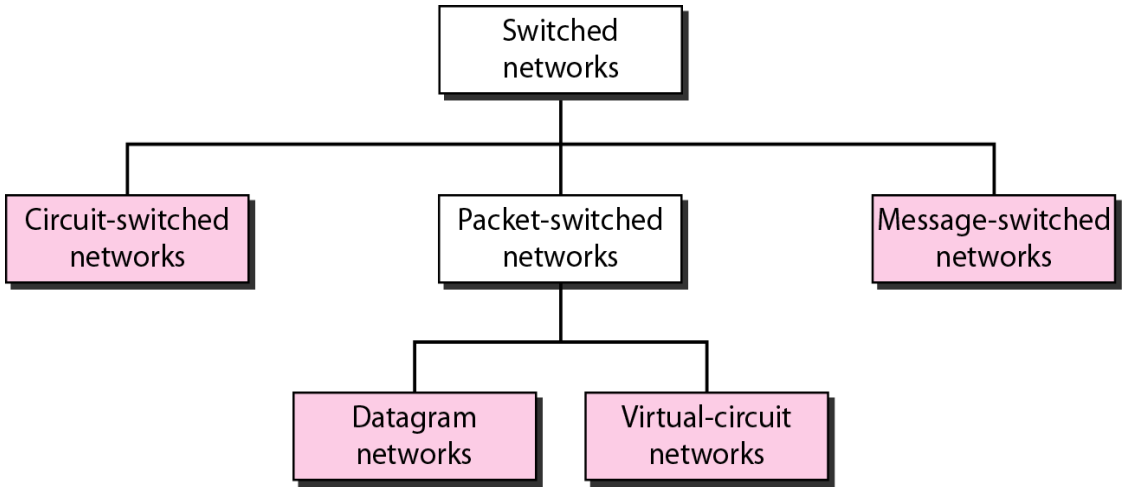


Figure: Taxonomy of switched networks

2.2.1 CIRCUIT-SWITCHED NETWORKS

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

- Figure shows a trivial circuit-switched network with four switches and four links. Each link is divided into *n (n* is 3 in the figure) channels by using FDM or TDM.

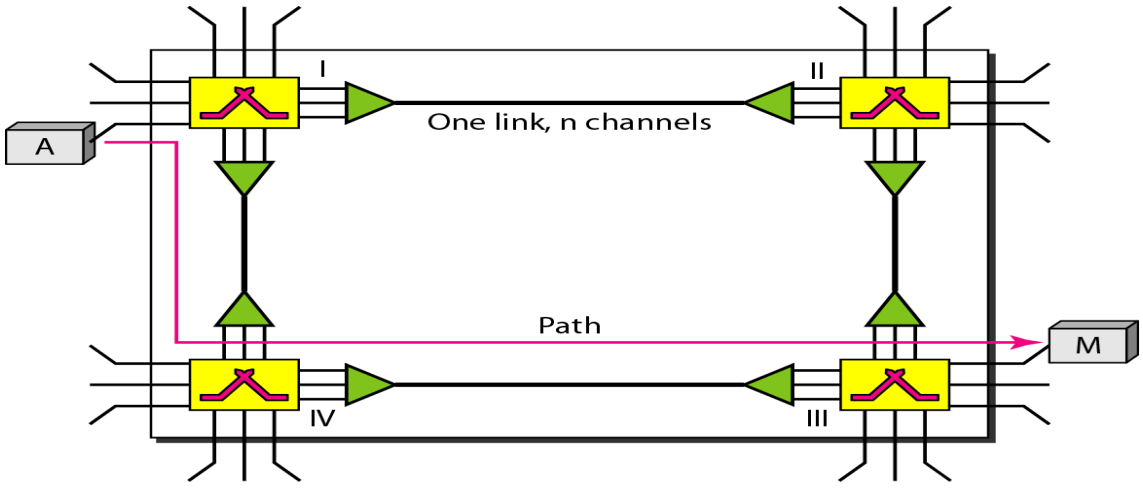


Figure: circuit-switched network

Three Phases

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

***Setup Phase**:*

- Before the two parties (or multiple parties in a conference call) can communicate, a dedicated circuit (combination of channels in links) needs to be established. The end systems are normally connected through dedicated lines to the switches, so connection setup means creating dedicated channels between the switches.
- For example, in Figure, when system A needs to connect to system M, it sends a setup request that includes the address of system M, to switch I. Switch I finds a channel between itself and switch IV that can be dedicated for this purpose. Switch I then sends the request to switch IV, which finds a dedicated channel between itself and switch III. Switch III informs system M of system A's intention at this time.
- In the next step to making a connection, an acknowledgment from system M needs to be sent in the opposite direction to system A. Only after system A receives this acknowledgment is the connection established. Note that end-to-end addressing is required for creating a connection between the two end systems. These can be, for example, the addresses of the computers assigned by the administrator in a TDM network, or telephone numbers in an FDM network.

***Data Transfer Phase**:*

- After the establishment of the dedicated circuit (channels), the two parties can transfer data.

Teardown Phase:

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

Efficiency:

It can be argued that circuit-switched networks are not as efficient as the other two types of networks because resources are allocated during the entire duration of the connection. These resources are unavailable to other connections.

- In a telephone network, people normally terminate the communication when they have finished their conversation. However, in computer networks, a computer can be connected to another computer even if there is no activity for a long time. In this case, allowing resources to be dedicated means that other connections are deprived.

Delay :

Although a circuit-switched network normally has low efficiency, the delay in this type of network is minimal. During data transfer the data are not delayed at each switch; the resources are allocated for the duration of the connection.

- Figure 8.6 shows the idea of delay in a circuit-switched network when only two switches are involved. As Figure shows, there is no waiting time at each switch. The total delay is due to the time needed to create the connection, transfer data, and disconnect the circuit.

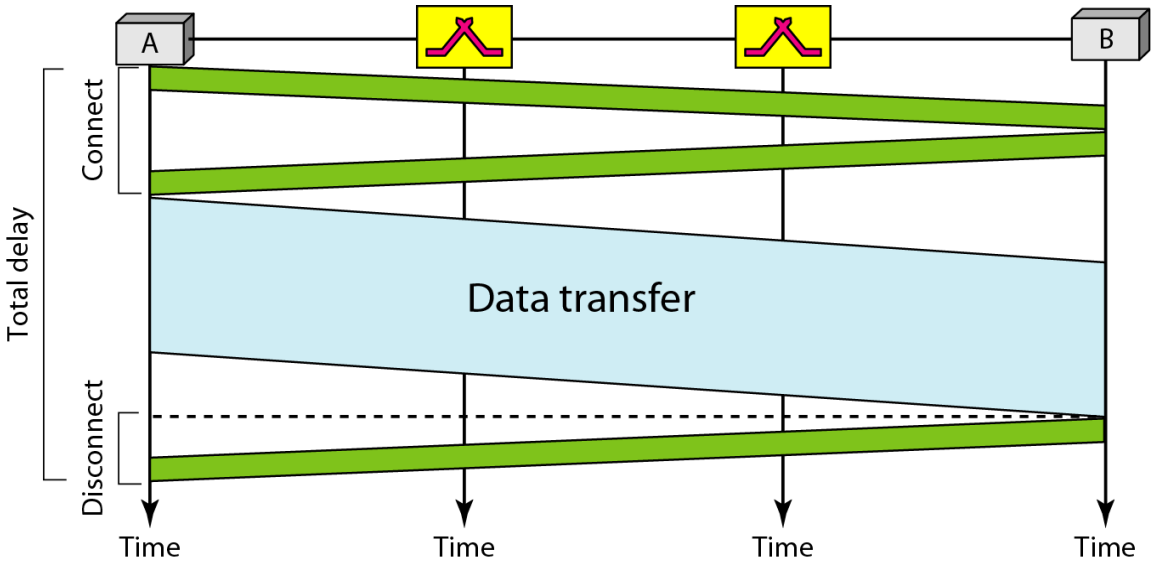


Figure: Delay in a circuit-switched network

- The delay caused by the setup is the sum of four parts: the propagation time of the source computer request (slope of the first gray box), the request signal transfer time (height of the first gray box), the propagation time of the acknowledgment from the destination computer (slope of the second gray box), and the signal transfer time of the acknowledgment (height of the second gray box). The delay due to data transfer is the sum of two parts: the propagation time (slope of the colored box) and data transfer time (height of the colored box), which can be very long.

- The third box shows the time needed to tear down the circuit. We have shown the case in which the receiver requests disconnection, which creates the maximum delay.

2.2.2 DATAGRAM NETWORKS

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

- Datagram switching is normally done at the network layer. We briefly discuss datagram networks here as a comparison with circuit-switched and virtual-circuit switched networks
- Figure shows how the datagram approach is used to deliver four packets from station A to station X. The switches in a datagram network are traditionally referred to as routers. That is why we use a different symbol for the switches in the figure.

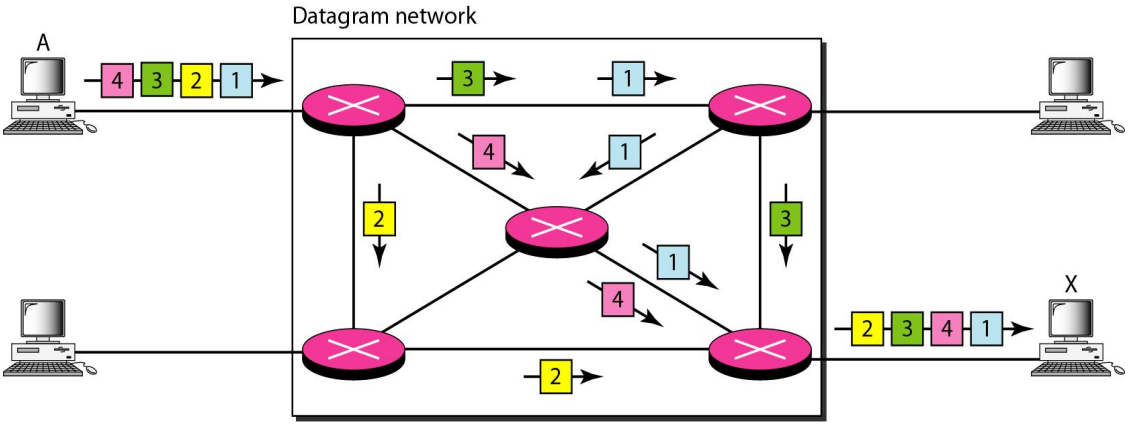


Figure : A datagram network with four switches

- In this example, all four packets (or datagrams) belong to the same message, but may travel different paths to reach their destination. This is so because the links may be involved in carrying packets from other sources and do not have the necessary bandwidth available to carry all the packets from A to X.
- This approach can cause the datagrams of a transmission to arrive at their destination out of order with different delays between the packets. Packets may also be lost or dropped because of a lack of resources. In most protocols, it is the responsibility of an upper-layer protocol to reorder the datagrams or ask for lost datagrams before passing them on to the application.
- The datagram networks are sometimes referred to as connectionless networks. The term *connectionless* here means that the switch (packet switch) does not keep information about the connection state. There are no setup or teardown phases. Each packet is treated the same by a switch regardless of its source or destination.

Routing Table

- If there are no setup or teardown phases, how are the packets routed to their destinations in a datagram network? In this type of network, each switch (or packet switch) has a routing table which is based on the destination address. The routing tables are dynamic and are updated periodically.

- The destination addresses and the corresponding forwarding output ports are recorded in the tables. This is different from the table of a circuit switched network in which each entry is created when the setup phase is completed and deleted when the teardown phase is over. Figure shows the routing table for a switch.

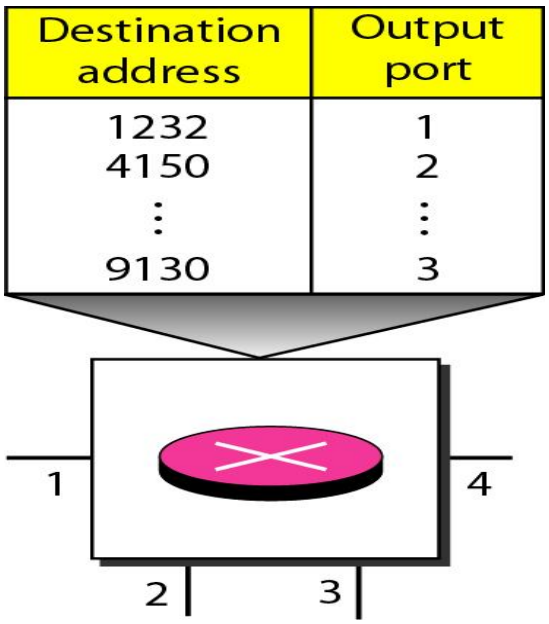


Figure: Routing table in a datagram network

Destination Address

- Every packet in a datagram network carries a header that contains, among other information, the destination address of the packet. When the switch receives the packet, this destination address is examined; the routing table is consulted to find the corresponding port through which the packet should be forwarded.
- This address, unlike the address in a virtual-circuit-switched network, remains the same during the entire journey of the packet.

Efficiency

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

Delay

There may be greater delay in a datagram network than in a virtual-circuit network. Although there are no setup and teardown phases, each packet

may experience a wait at a switch before it is forwarded. In addition, since not all packets in a message necessarily travel through the same switches, the delay is not uniform for the packets of a message.

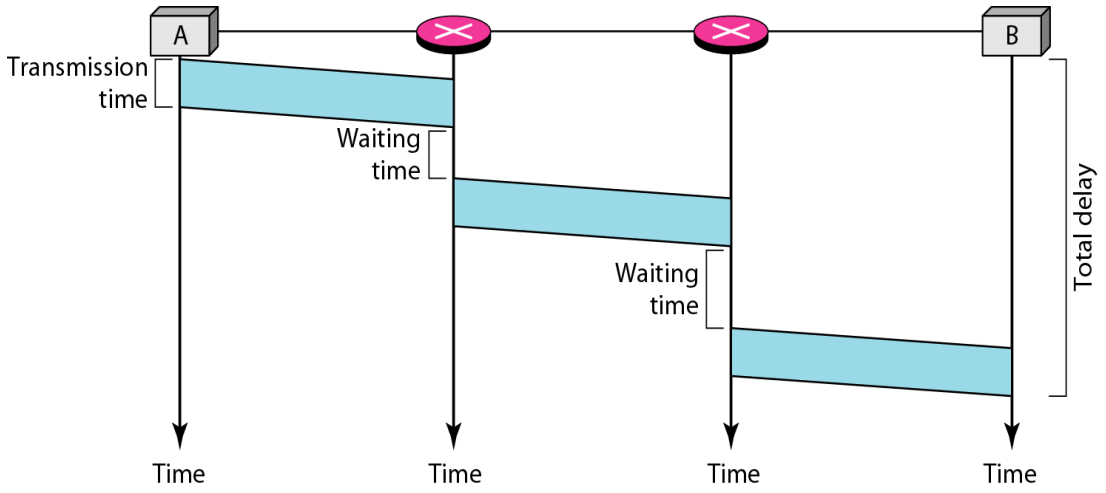


Figure: Delay in a datagram network

The packet travels through two switches. There are three transmission times *(3T),* three propagation delays (slopes 3't of the lines), and two waiting times (W 1 + W*2).* We ignore the processing time in each switch. The total delay is Total delay =*3T* + 3t + W1 + W2

2.2.3 VIRTUAL-CIRCUIT NETWORKS:

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

1. As in a circuit-switched network, there are setup and teardown phases in addition to the data transfer phase.
2. Resources can be allocated during the setup phase, as in a circuit-switched network, or on demand, as in a datagram network.
3. As in a datagram network, data are packetized and each packet carries an address in the header. However, the address in the header has local jurisdiction (it defines what should be the next switch and the channel on which the packet is being carried), not end-to-end jurisdiction. The reader may ask how the intermediate switches know where to send the packet if there is no final destination address carried by a packet. The answer will be clear when we discuss virtual-circuit identifiers in the next section.
4. As in a circuit-switched network, all packets follow the same path established during the connection.
5. A virtual-circuit network is normally implemented in the data link layer, while a circuit-switched network is implemented in the physical layer and a datagram network in the network layer. But this may change in the future.

Figure is an example of a virtual-circuit network. The network has switches that allow traffic from sources to destinations. A source or destination can be a computer, packet switch, bridge, or any other device that connects other networks.

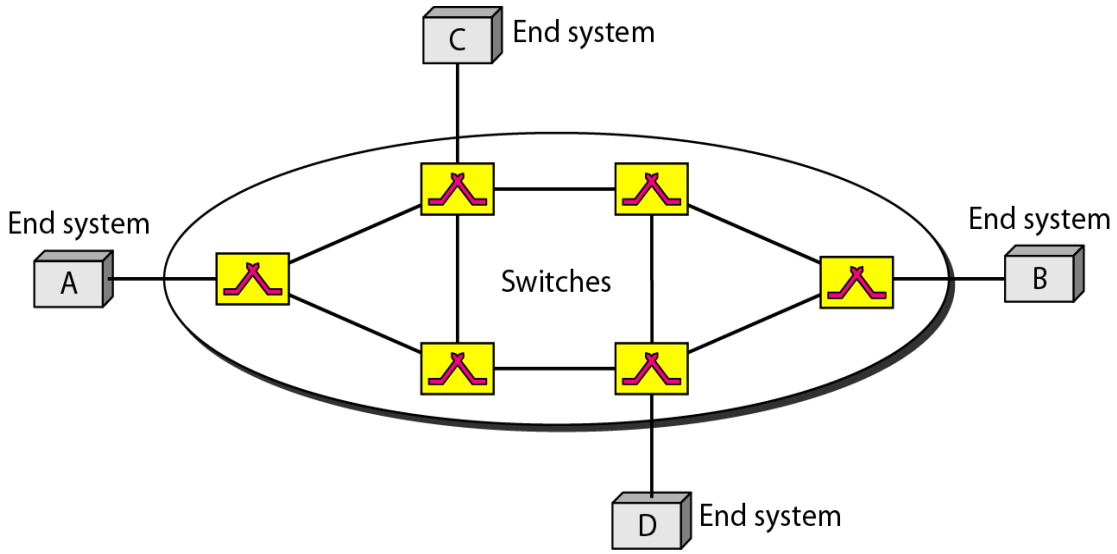


Figure: virtual-circuit network

Addressing

In a virtual-circuit network, two types of addressing are involved: global and local (virtual-circuit identifier).

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

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

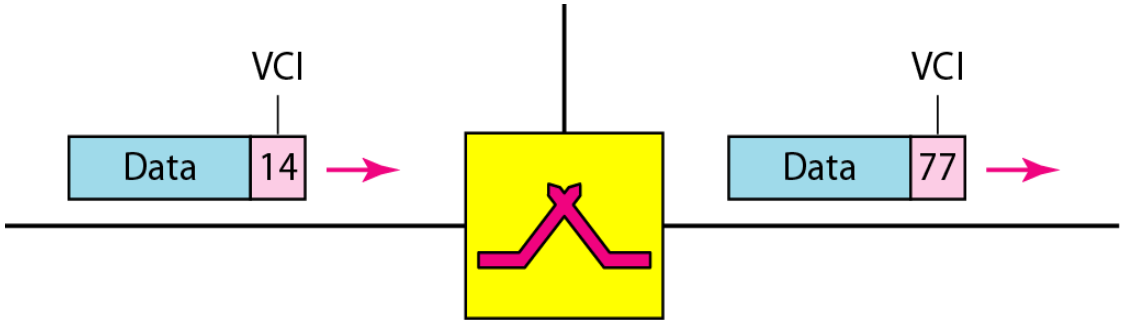


Figure: virtual-circuit identifier

Three Phases

As in a circuit-switched network, a source and destination need to go through three phases in a virtual-circuit network: setup, data transfer, and teardown. In the setup phase, the source and destination use their global addresses to help switches make table entries for the connection. In the teardown phase, the source and destination inform the switches to delete the corresponding entry. Data transfer occurs between these two phases.

- We first discuss the data transfer phase, which is more straightforward; we then talk about the setup and teardown phases.

Data Transfer Phase :

To transfer a frame from a source to its destination, all switches need to have a table entry for this virtual circuit. The table, in its simplest form, has four columns. This means that the switch holds four pieces of information for each virtual circuit that is already set up.

- Figure 2 shows such a switch and its corresponding table. And also shows a frame arriving at port 1 with a VCI of 14. When the frame arrives, the switch looks in its table to find port 1 and a VCI of 14. When it is found, the switch knows to change the VCI to 22 and send out the frame from port 3.
- Figure 3 shows how a frame from source A reaches destination B and how its VCI changes during the trip. Each switch changes the VCI and routes the frame. The data transfer phase is active until the source sends all its frames to the destination. The procedure at the switch is the same for each frame of a message. The process creates a virtual circuit, not a real circuit, between the source and destination.

Setup Phase:

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

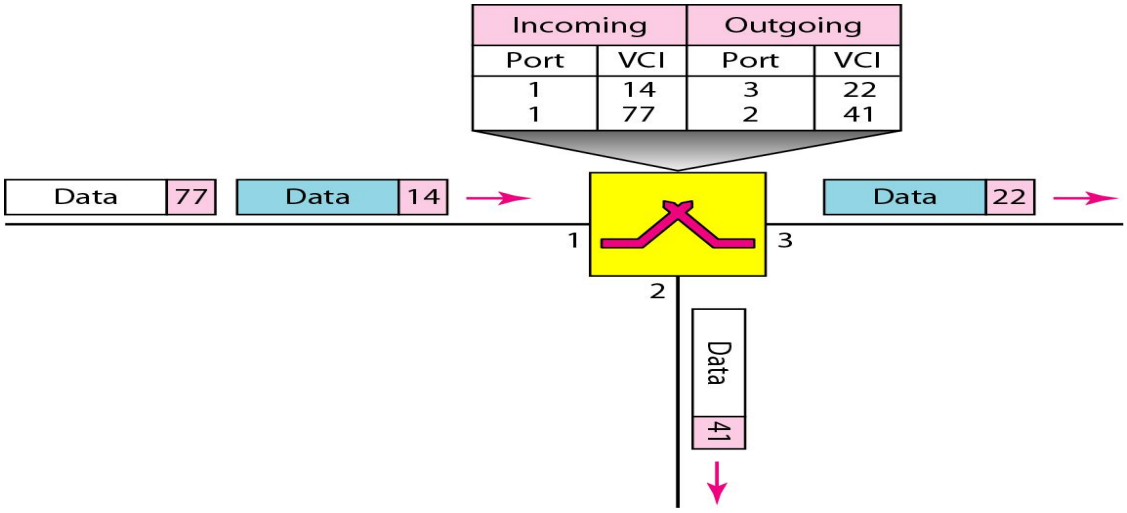


Figure: switch and tables in a virtual-circuit network

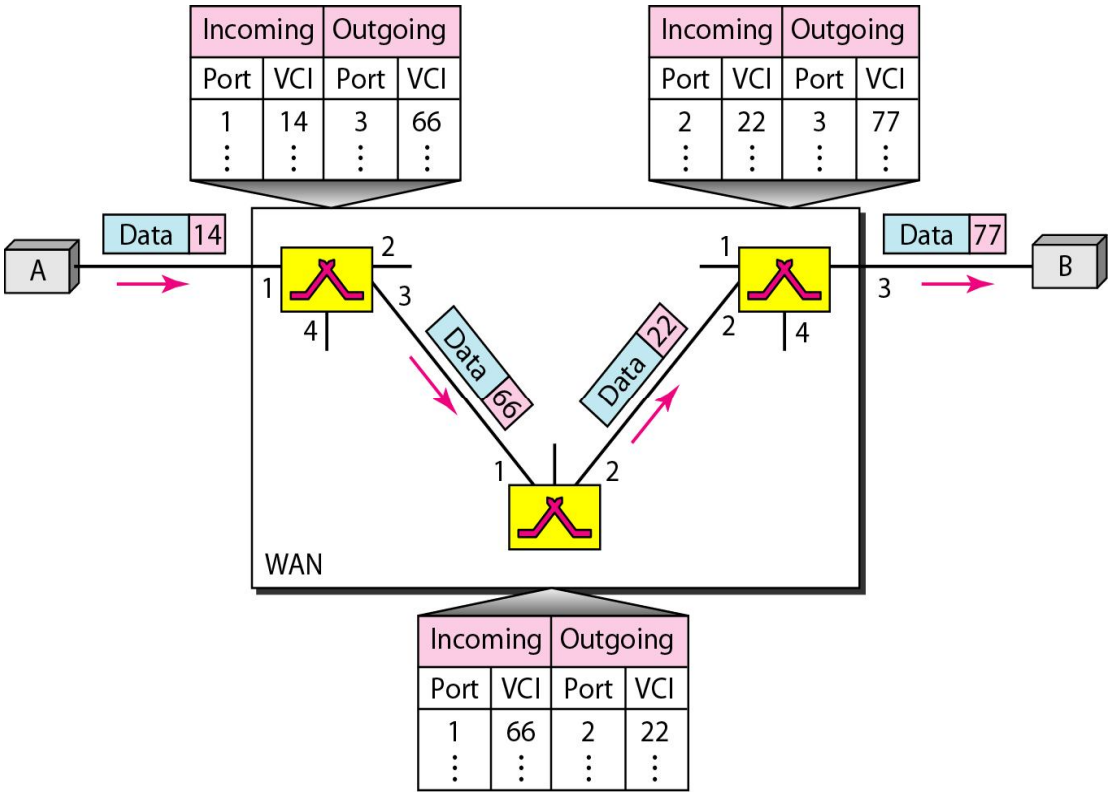


Figure: ***Source-to-destination data transfer in a virtual-circuit network***

Setup Request :

A setup request frame is sent from the source to the destination.
Figure 4 shows the process.

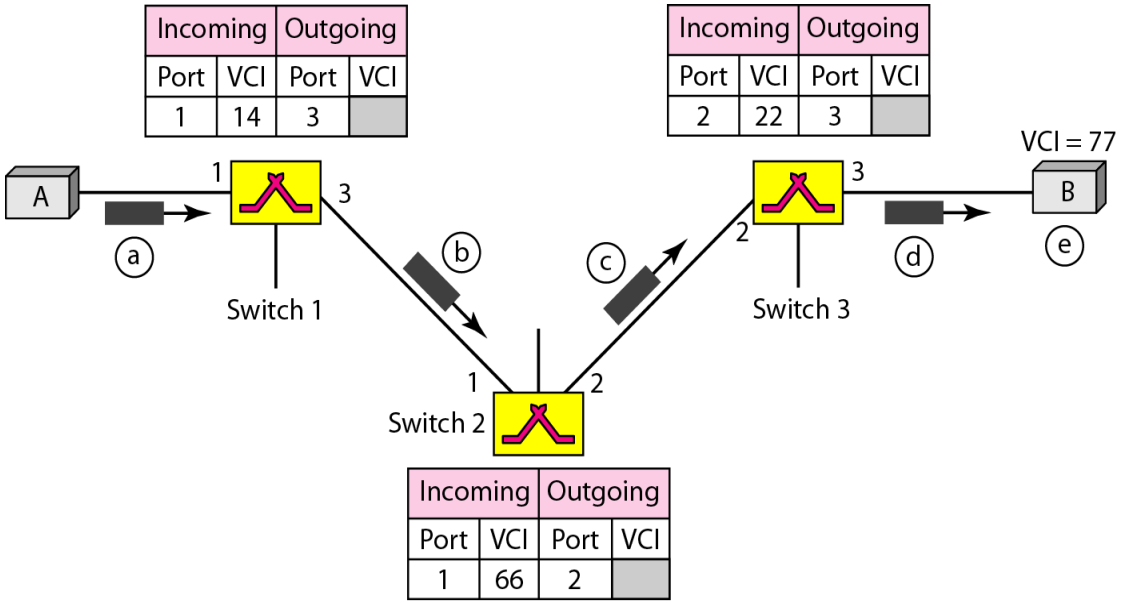


Figure: ***Setup request in a virtual-circuit network***

a. Source A sends a setup frame to switch 1.
b. Switch 1 receives the setup request frame. It knows that a frame going from A to B goes out through port 3.
The switch, in the setup phase, acts as a packet switch; it has a routing table which is different from the switching table. For the moment, assume that it knows the output port. The switch creates an entry in its table for this virtual circuit, but it is only able to fill three of the four columns. The switch assigns the incoming port (1) and chooses an available incoming VCI (14) and the outgoing port (3). It does not yet know the outgoing VCI, which will be found during
the acknowledgment step. The switch then forwards the frame through port 3 to switch 2.
c. Switch 2 receives the setup request frame. The same events happen here as at switch 1; three columns of the table are completed: in this case, incoming port (l), incoming VCI (66), and outgoing port (2).
d. Switch 3 receives the setup request frame. Again, three columns are completed: incoming port (2), incoming VCI (22), and outgoing port (3).
e. Destination B receives the setup frame, and if it is ready to receive frames from A, it assigns a VCI to the incoming frames that come from A, in this case 77. This VCI lets the destination know that the frames come from A, and not other sources.
Acknowledgment A special frame, called the acknowledgment frame, completes the entries in the switching tables. Figure 8.15 shows the process.
a. The destination sends an acknowledgment to switch 3. The acknowledgment carries the global source and destination addresses so the switch knows which entry in the table is to be completed. The frame also

carries VCI 77, chosen by the destination as the incoming VCI for frames from A. Switch 3 uses this VCI to complete the outgoing VCI column for this entry. Note that 77 is the incoming VCI for destination B, but the outgoing VCI for switch 3.

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

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

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

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

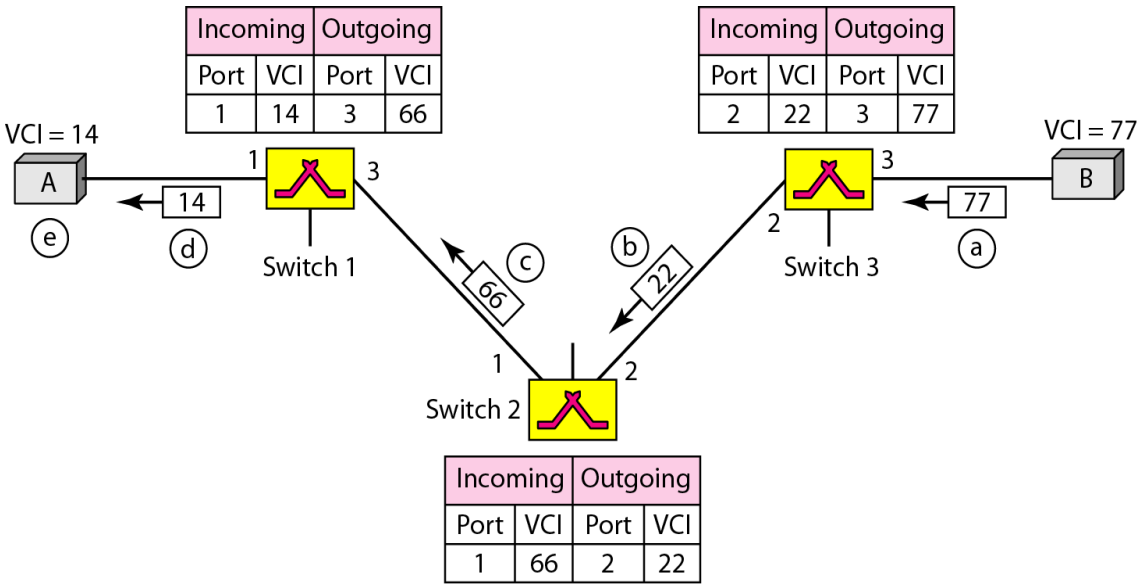


Figure :*Setup acknowledgment in a virtual-circuit network*

Teardown Phase :

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

Efficiency

As we said before, resource reservation in a virtual-circuit network can be made during the setup or can be on demand during the data transfer phase. In the first case, the delay for each packet is the same; in the second case, each packet may encounter different delays.

- There is one big advantage in a virtual-circuit network even if resource allocation is on demand. The source can check the availability of the resources, without actually reserving it. Consider a family that wants to dine at a restaurant. Although the restaurant may not accept reservations (allocation of the tables is on demand), the family can call and find out the waiting time. This can save the family time and effort.

Delay in Virtual-Circuit Networks

In a virtual-circuit network, there is a one-time delay for setup and a one-time delay for teardown. If resources are allocated during the setup phase, there is no wait time for individual packets. Below Figure shows the delay for a packet traveling through two switches in a virtual-circuit network.

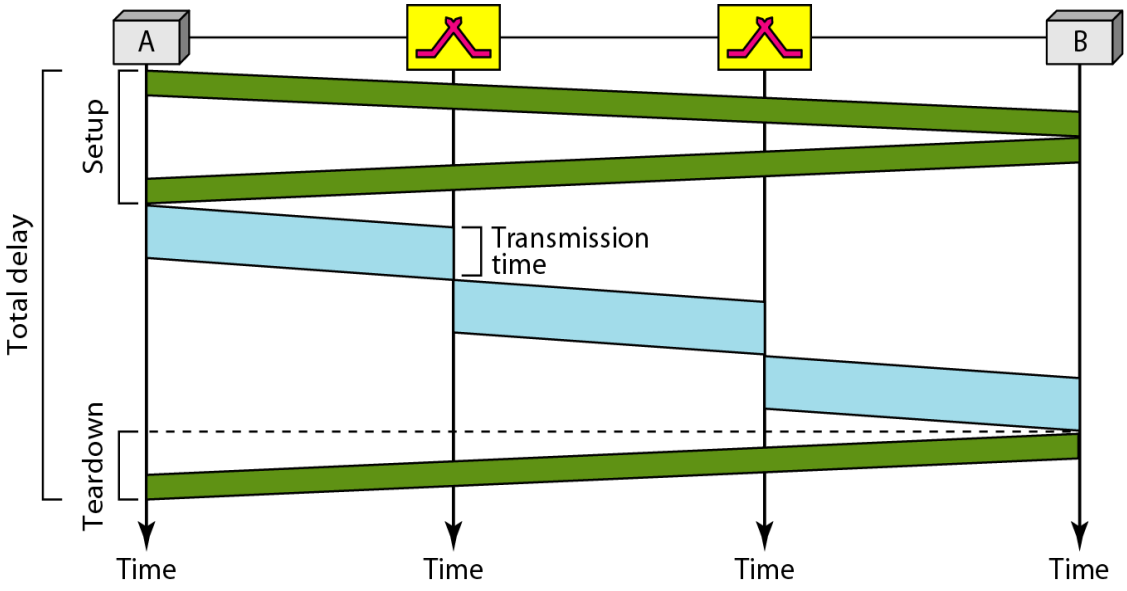


Figure : *Delay in a virtual-circuit network*

The packet is traveling through two switches (routers). There are three transmission times *(3T),* three propagation times (3't), data transfer depicted by the sloping lines, a setup delay (which includes transmission and propagation in two directions), and a teardown delay (which includes transmission and propagation in one direction). We ignore the processing time in each switch. The total delay time is

Total delay = *3T*+ 3't + setup delay + teardown delay

UNIT-II

Assignment-Cum-Tutorial Questions

SECTION-A

Objective Questions

1) 1. The sharing of a medium and its link by two or more devices is called _____ [ ]

a)Fully duplexing b) Multiplexing

c) Both a and b d) None of the mentioned

2) Multiplexing is used in _____ [ ]

a) Packet switching b) Circuit switching

c) Data switching d) None of the mentioned

3)Which multiplexing technique transmits digital signals ? [ ]

a) FDM b) TDM c) WDM d) None of the mentioned

4)Multiplexing can provide [ ]

a) Efficiency b) Privacy

c) Anti jamming d) Both a and b

5)The state when dedicated signals are idle are called [ ]

a) Death period b) Poison period

c) Silent period d) None of the mentioned

6)In TDM, slots are further divided into [ ]

a) Seconds b) Frames

c) Packets d) None of the mentioned

7)In TDM Data rate management is done by which of these strategies

a) Multilevel multiplexing b) Multi-slot allocation [ ]

c) Pulse stuffing d) all of the above

8) Method(s) to move data through a network of links and switches

a)Packet switching b)Circuit switching [ ]

c)Line switching d) Both a and b

9._____ is an analog multiplexing technique that combines analog signals

10._____ is an analog multiplexing technique to combine optical signals.

11. A _____ service is a single digital channel of 64kbps

12) If there are n signal sources of same data rate than TDM link has _____slots [ ]

a)n b)n/2 c)n*2 d) 2^n

13) if link transmits 4000frames per second, and each slot has 8 bits,the transmission rate of circuit this TDM is [ ]

a)32kbps b)500bps c)500kbps d) None of the mentioned

14)In TDM, the transmission rate of the multiplexed path is usually _____ the sum of the transmission rates of the signal sources.

a) Greater than b) Lesser than [ ]

c)Equal to d) Equal to or greater than

15)What are the phases in circuit switching? [ ]

a) Setup, data transfer, teardown

b) request-connect, data sending-acknowledgment, request-disconnect

c) send-connect, data transfer, request-disconnect

d) none of above

16)Which of these statements is true about packet switching networks? [ ]

a)Resource allocation is done for a packet beforehand

b) Bandwidth is reserved on the links

c) Scheduled processing for a packet

d) Resource allocation is done on demand

17)What are the components of a packet switch? [ ]

a) input ports, output ports, a router processor, a switching fabric

b) input ports, output ports, a router processor.

c) input ports, output ports, a switching fabric

d) input ports, output ports, a router processing, a switching fabric, a memory chip

18)How switching is performed in the internet? [ ]

a) data gram approach to circuit switching at datalink layer

b) Virtual circuit approach to message switching at network layer

c) datagram approach to message switching at datalink layer

d) datagram approach to packet switching at network layer

19)Which of these is correct for synchronous Time Division Multiplexing [ ]

a) Data rate of link is n times faster and the unit duration is n times longer

b) Data rate of link is n times slower and the unit duration is n times shorter

c) Data rate of link is n times slower and the unit duration is n times longer

d) Data rate of link is n times faster and the unit duration is n times shorter

20) Multiplexing technique that shifts each signal to a different carrierfrequency [ ]

a) FDM b)TDM c)Either a or b d) Both a and b

21)Which of these multiplexing techniques is digital for combining several low- rate channels into high-rate one [ ]

a)FDM b)WDM c)TDM d) None of the above

SECTION-B

SUBJECTIVE QUESTIONS

1. Describe the functioning of FDM

2. Discuss the various approaches to packet-switching

3. Compare and contrast a circuit-switched network and a packet switched network

4. Draw the diagram of a datagram network with four switches. And explain how will it work

5. Explain the process of TDM with an example

6. Write a short note on interleaving

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

8. Assume that a voice channel occupies a bandwidth of 4 kHz. We need to multiplex 10 voice channels with guard bands of 500 Hz using FDM. Calculate the required bandwidth.

9. Explain the addressing mechanism in virtual circuit networks.

10. What is multiplexing? Why is it necessary? Compare synchronous TDM with statistical TDM.

SECTION-C
QUESTIONS AT THE LEVEL OF GATE

1.Let us consider a statistical time division multiplexing of packets. The number of sources is 10. In a time unit, a source transmits a packet of 1000 bits. The number of sources sending data for the first 20 time units is 6, 9, 3, 7, 2, 2, 2, 3, 4, 6, 1, 10, 7, 5, 8, 3, 6, 2, 9, 5 respectively. The output capacity of multiplexer is 5000 bits per time unit. Then the average number of backlogged of packets per time unit during the given period is **(GATE-2007)** **[ ]**

(A) 5 (B) 4.45 (C) 3.45 (D) 0

2. A link carries data from n connections. In TDM, the data rate of link is _____ time faster and uni-duration is _____ time shorter than a single link connection. **(GATE-2015)** **[ ]**

a)n,n b)(n+1),(n+1) c)n, n+1 d)(n+1),n

UNIT-III

Objectives:

- Understanding of the How the Data link Layer Perform Flow and Error control methods.
- Basic Knowledge of Elementary Protocols Supported by Data link Layer

Syllabus:

Data Link Layer Framing - fixed size framing, variable size framing, , Flow control, Error control, Error detection, Error correction - block coding, linear block codes, cyclic codes - cyclic redundancy check, Checksum - idea, one's complement internet check sum, services provided to Network Layer, Elementary Data link Layer protocols - Unrestricted Simplex protocol, Simplex Stop-and-Wait Protocol, Simplex protocol for Noisy Channel.

Outcomes:

Students will be able to

- Different Framing Methods used by data Link Layer
- Error Correction by Codes like Linear, Block, CRC
- What type of Services provided to network layer
- Types of Noisy channels protocols

Learning Material

3.1 FRAMING:

- Data transmission in the physical layer means moving bits in the form of a signal from the source to the destination.
- The physical layer provides bit synchronization to ensure that the sender and receiver use the same bit durations and timing.
- The data link layer, on the other hand, needs to pack bits into frames, so that each frame is distinguishable from another. Our postal system practices a type of framing.
- The simple act of inserting a letter into an envelope separates one piece of information from another; the envelope serves as the delimiter. In addition, each envelope defines the sender and receiver addresses since the postal system is a many-to-many carrier facility.
- Framing in the data link layer separates a message from one source to a destination, or from other messages to other destinations, by adding a sender address and a destination address. The destination address defines where the packet is to go; the sender address helps the recipient acknowledge the receipt.

3.1.1 Fixed-Size Framing: Frames can be of fixed or variable size. In fixed-size framing, there is no need for defining the boundaries of the frames; the size itself can be used as a delimiter. An example of this type of framing is the ATM wide-area network, which uses frames of fixed size called cells.

3.1.2 Variable-Size Framing : In variable-size framing, we need a way to define the end of the frame and the beginning of the next. Historically, two approaches were used for this purpose: a character-oriented approach and a bit-oriented approach

3.1.2.1 *Character-Oriented Protocols*

- In a character-oriented protocol, data to be carried are 8-bit characters from a coding system such as The header, which normally carries the source and destination addresses and other control information, and the trailer, which carries error detection or error correction redundant bits, are also multiples of 8 bits.

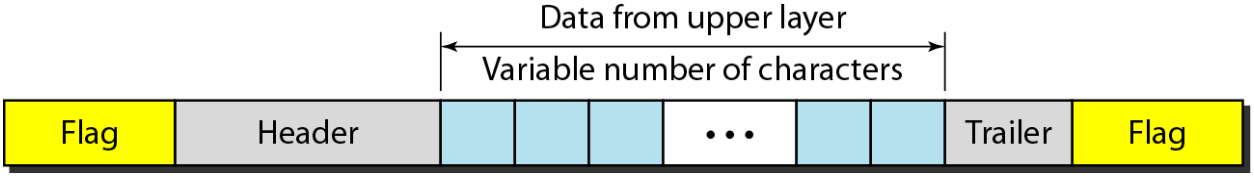


Figure: *A frame in a character-oriented protocol*

- To separate one frame from the next, an 8-bit (1-byte) flag is added at the beginning and the end of a frame. The flag, composed of protocol-dependent special characters, signals the start or end of a frame. Below figure shows the format of a frame in a character-oriented protocol.
- Character-oriented framing was popular when only text was exchanged by the data link layers. The flag could be selected to be any character not used for text communication. Now, however, we send other types of information such as graphs, audio, and video. Any pattern used for the flag could also be part of the information. If this happens, the receiver, when it encounters this pattern in the middle of the data, thinks it has reached the end of the frame. To fix this problem, a byte-stuffing strategy was added to character-oriented framing. In byte stuffing (or character stuffing), a special byte is added to the data section of the frame when there is a character with the same pattern as the flag. The data section is stuffed with an extra byte. This byte is usually called the escape character (ESC), which has a predefined bit pattern. Whenever the receiver encounters the ESC character, it removes it from the data section and treats the next character as data, not a delimiting flag.

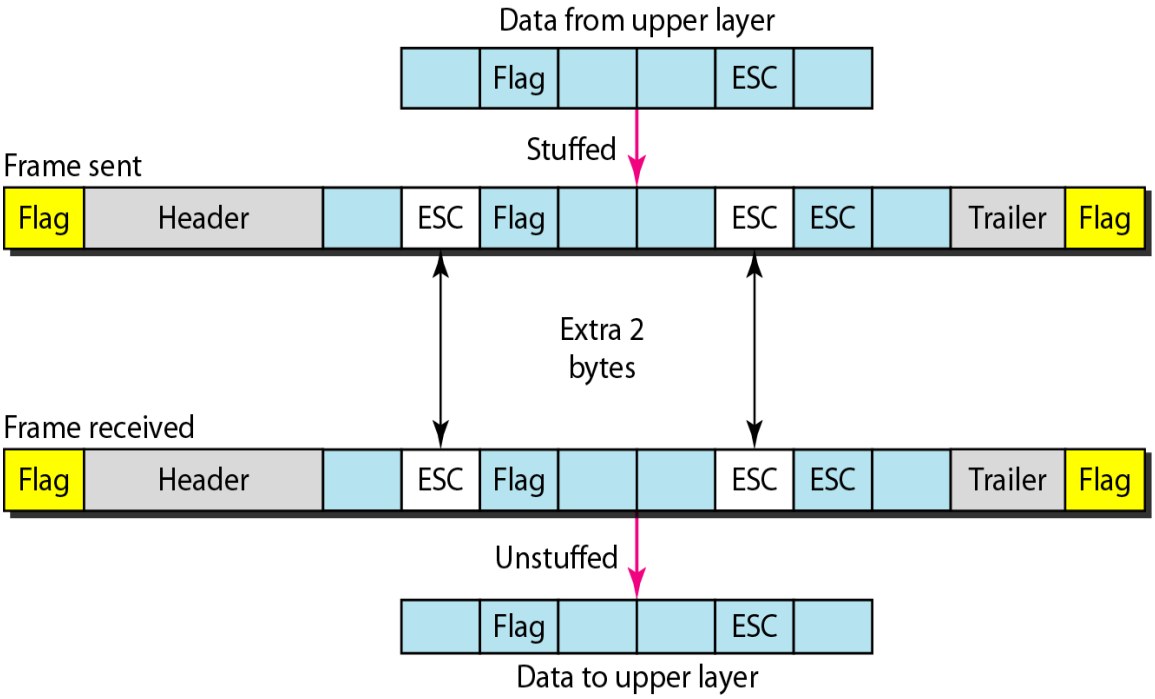


Figure : *Byte stuffing and unstuffing*

- Byte stuffing by the escape character allows the presence of the flag in the data section of the frame, but it creates another problem. What happens if the text contains one or more escape characters followed by a flag? The receiver removes the escape character, but keeps the flag,

which is incorrectly interpreted as the end of the frame. To solve this problem, the escape characters that are part of the text must also be marked by another escape character. In other words, if the escape character is part of the text, an extra one is added to show that the second one is part of the text. Below figure shows the situation.

3.1.2.2Bit-Oriented Protocols

- In a bit-oriented protocol, the data section of a frame is a sequence of bits to be interpreted by the upper layer as text, graphic, audio, video, and so on. However, in addition to headers (and possible trailers), we still need a delimiter to separate one frame from the other. Most protocols use a special 8-bit pattern flag 01111110 as the delimiter to define the beginning and the end of the frame, as shown in figure.

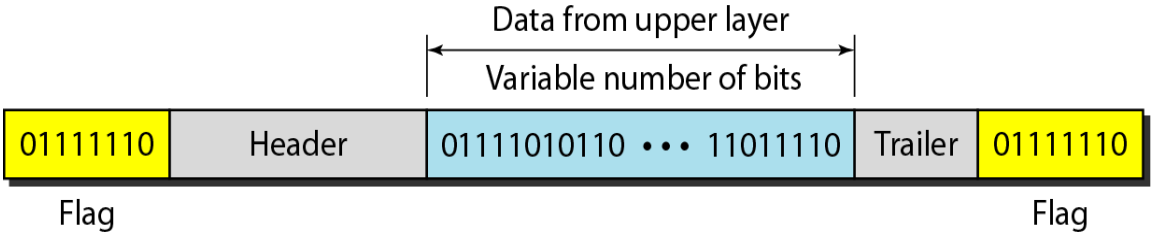


Figure: *A frame in a bit-oriented protocol*

- This flag can create the same type of problem we saw in the byte-oriented protocols.
- That is, if the flag pattern appears in the data, we need to somehow inform the receiver that this is not the end of the frame. We do this by stuffing 1 single bit (instead of 1 byte) to prevent the pattern from looking like a flag. The strategy is called bit stuffing. In bit stuffing, if a 0 and five consecutive 1 bits are encountered, an extra 0 is added.
- This extra stuffed bit is eventually removed from the data by the receiver. Note that the extra bit is added after one 0 followed by five 1s regardless of the value of the next bit. This guarantees that the flag field sequence does not inadvertently appear in the frame.
- Below figure shows bit stuffing at the sender and bit removal at the receiver. Note that even if we have a 0 after five 1s, we still stuff a 0. The 0 will be removed by the receiver.

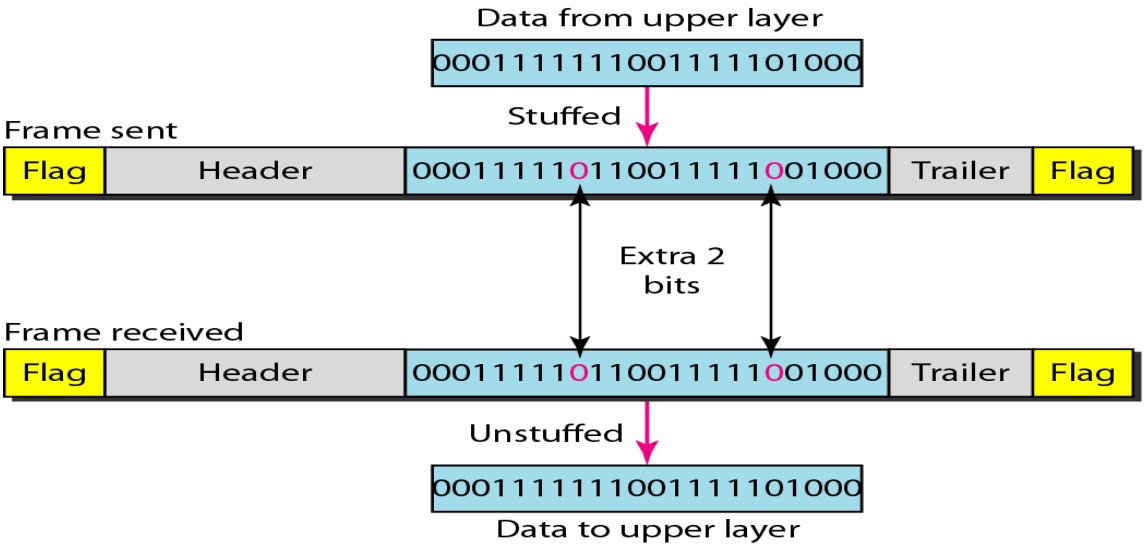


Figure:*Bit stuffing and unstuffing*

- This means that if the flaglike pattern 01111110 appears in the data, it will change to 011111010 (stuffed) and is not mistaken as a flag by the receiver. The real flag 01111110 is not stuffed by the sender and is recognized by the receiver.

3.2FLOW AND ERROR CONTROL

Data communication requires at least two devices working together, one to send and the other to receive. Even such a basic arrangement requires a great deal of coordination for an intelligible exchange to occur. The most important responsibilities of the data link layer are flow control and error control. Collectively, these functions are known as data link control.

3.2.1Flow Control

- Flow control coordinates the amount of data that can be sent before receiving an acknowledgment and is one of the most important duties of the data link layer. In most protocols,
- Flow control is a set of procedures that tells the sender how much data it can transmit before it must wait for an acknowledgment from the receiver. The flow of data must not be allowed to overwhelm the receiver.
- Any receiving device has a limited speed at which it can process incoming data and a limited amount of memory in which to store incoming data. The receiving device must be able to inform the sending device before those limits are reached and to request that the transmitting device send fewer frames or stop temporarily.
- Incoming data must be checked and processed before they can be used. The rate of such processing is often slower than the rate of transmission.
- For this reason, each receiving device has a block of memory, called a *buffer,* reserved for storing incoming data until they are processed.

- If the buffer begins to fill up, the receiver must be able to tell the sender to halt transmission until it is once again able to receive.

3.2.2 Error Control

- Error control is both error detection and error correction. It allows the receiver to inform the sender of any frames lost or damaged in transmission and coordinates the retransmission of those frames by the sender.
- In the data link layer, the term *error control* refers primarily to methods of error detection and retransmission.
- Error control in the data link layer is often implemented simply: Any time an error is detected in an exchange, specified frames are retransmitted. This process is called automatic repeat request (ARQ).

3.3 *Error Detection and Correction*

3.3.1 Introduction:

3.3.1.1Types of Errors

Whenever bits flow from one point to another, they are subject to unpredictable changes because of interference. This interference can change the shape of the signal. In a single-bit error, a 0 is changed to a 1 or a 1 to a O. In a burst error, multiple bits are changed.

Single-Bit Error

The term *single-bit error* means that only 1 bit of a given data unit (such as a byte, character, or packet) is changed from 1 to 0 or from 0 to 1.

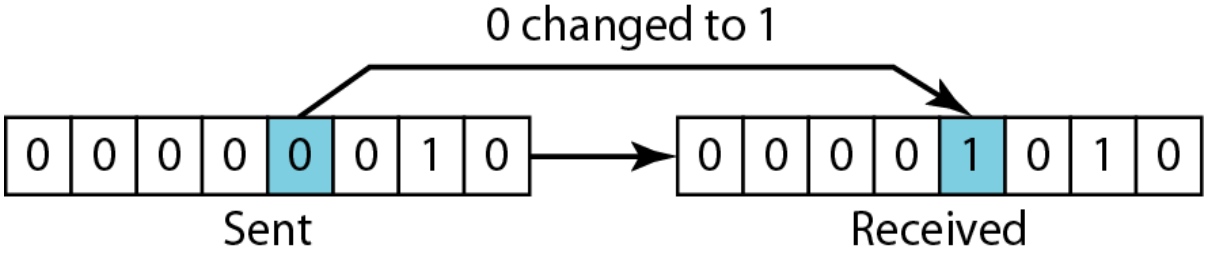


Figure :*Single-bit error*

Burst Error

- The term *burst error* means that 2 or more bits in the data unit have changed from 1 to 0 or from 0 to 1.
- Figure shows the effect of a burst error on a data unit. In this case, 0100010001000011 was sent, but 0101110101100011 was received. Note that a burst error does not necessarily mean that the errors occur in consecutive bits. The length of the burst is measured from the first corrupted bit to the last corrupted bit.

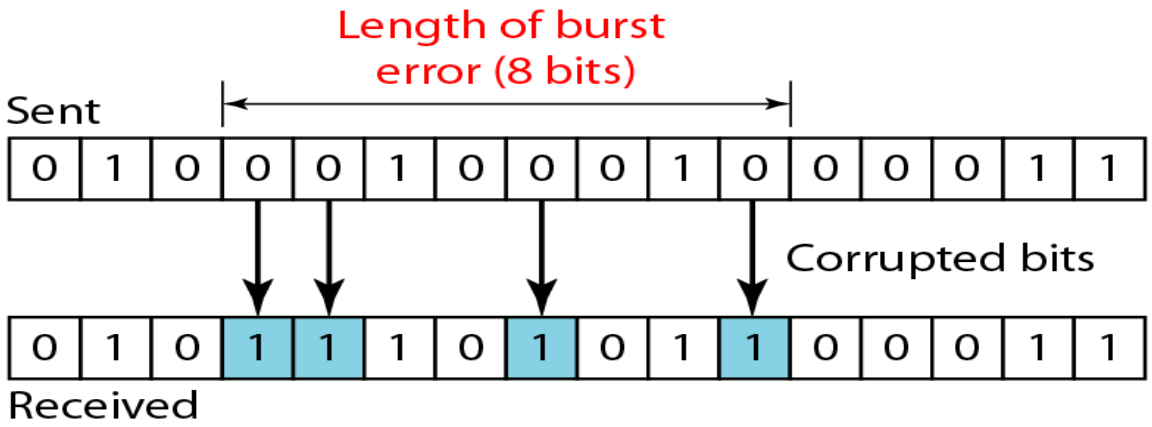


Figure:Burst error of length 8

3.3.1.2Redundancy

- The central concept in detecting or correcting errors is redundancy. To be able to detect or correct errors, we need to send some extra bits with our data. These redundant bits are added by the sender and removed by the receiver. Their presence allows the receiver to detect or correct corrupted bits.

3.3.1.3 Coding

- Redundancy is achieved through various coding schemes. The sender adds redundant bits through a process that creates a relationship between the redundant bits and the actual data bits. The receiver checks the relationships between the two sets of bits to detect or correct the errors. The ratio of redundant bits to the data bits and the robustness of the process are important factors in any coding scheme. Figure shows the general idea of coding.

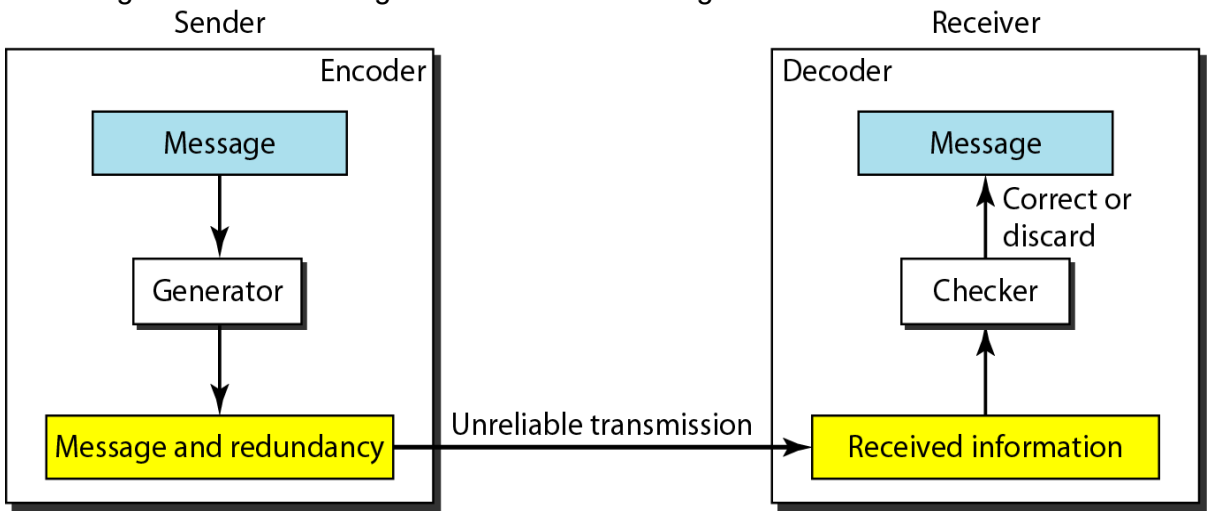


Figure :*The structure of encoder and decoder*

Coding schemes are classified into two broad categories: block coding and convolution coding.

3.3.2 Block coding

- In block coding, we divide our message into blocks, each of *k* bits, called datawords. We add *r* redundant bits to each block to make the length *n* = *k* + *r.* The resulting *n-bit* blocks are called codewords.
- How the extra *r* bits is chosen or calculated is something we will discuss later. For the moment, it is important to know that we have a set of datawords, each of size *k,* and a set of codewords, each of size of *n.*
- With *k* bits, we can create a combination of *2k* datawords; with *n* bits, we can create a combination of *2n* codewords. Since *n* > *k,* the number of possible codewords is larger than the number of possible datawords. The block coding process is one-to-one; the same dataword is always encoded as the same codeword.
- This means that we have $2^n - 2^k$ codewords that are not used. We call these codewords invalid or illegal. Figure shows the situation.

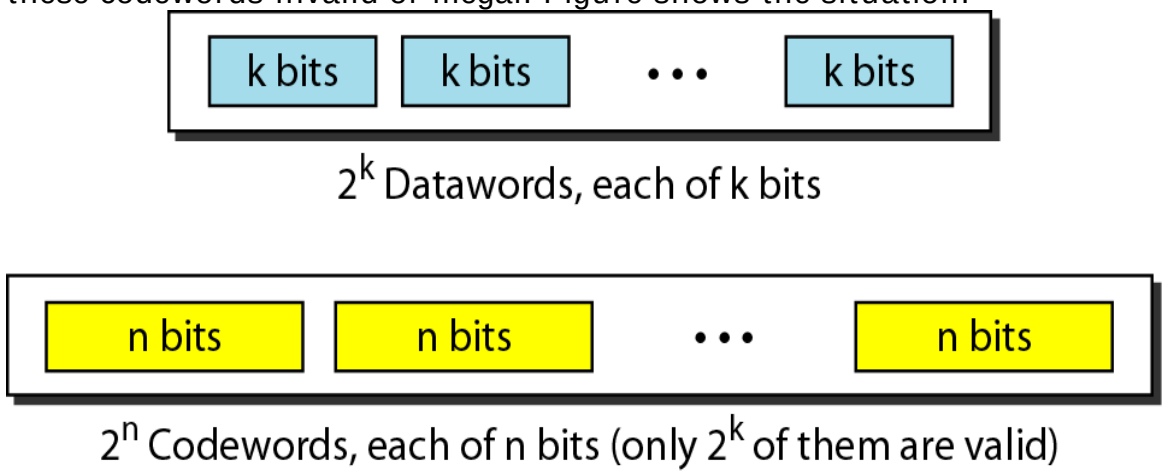


Figure: Datawords and Codewords in Block Coding

3.3.2.1 Error Detection:

How can errors be detected by using block coding? If the following two conditions are met, the receiver can detect a change in the original codeword.

i. The receiver has (or can find) a list of valid codewords.

ii. The original codeword has changed to an invalid one.

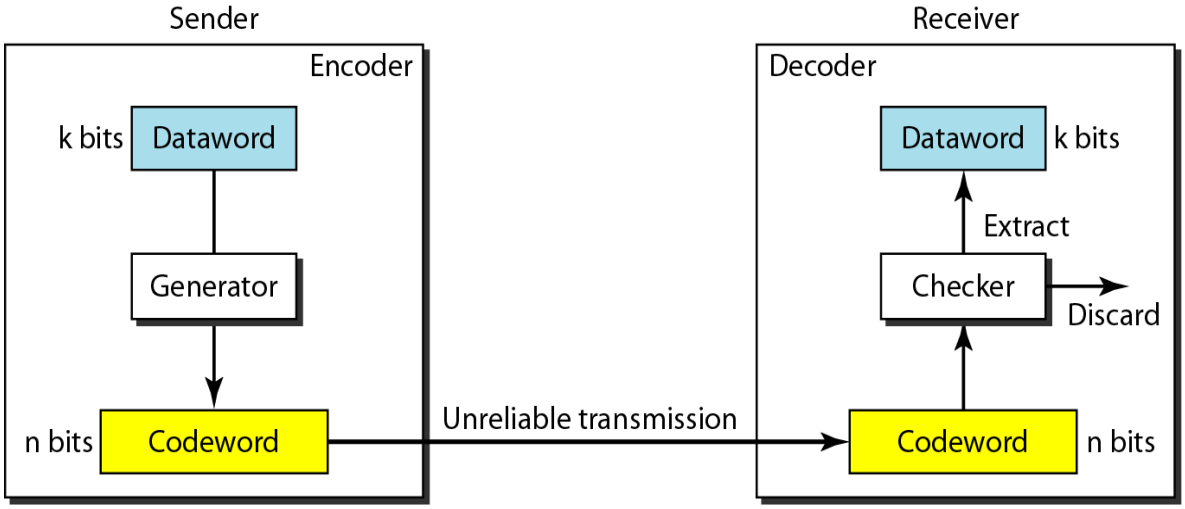


Figure :*Process of error detection in block coding*

- The sender creates codewords out of datawords by using a generator that applies the rules and procedures of encoding .Each codeword sent to the receiver may change during transmission.
- If the received codeword is the same as one of the valid codewords, the word is accepted; the corresponding dataword is extracted for use.
- If the received codeword is not valid, it is discarded. However, if the codeword is corrupted during transmission but the received word still matches a valid codeword, the error remains undetected.
- This type of coding can detect only single errors. Two or more errors may remain undetected.

3.3.2.2 Error Correction

- In error detection, the receiver needs to know only that the received codeword is invalid; in error correction the receiver needs to find (or guess) the original codeword sent. We can say that we need more redundant bits for error correction than for error detection. Figure shows the role of block coding in error correction.

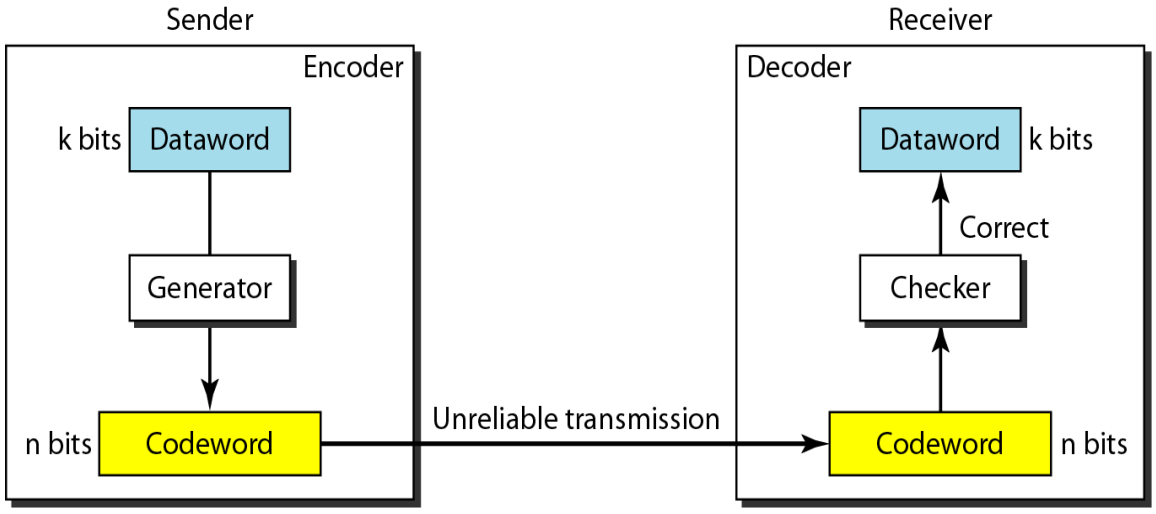


Figure :*Structure of encoder and decoder in error correction*

3.3.3 Hamming Distance

- One of the central concepts in coding for error control is the idea of the Hamming distance.
- The Hamming distance between two words (of the same size) is the number of differences between the corresponding bits. We show the Hamming distance between two words *x* and *y* as *d(x, y).*
- The Hamming distance can easily be found if we apply the XOR operation on the two words and count the number of Is in the result.
- Note that the Hamming distance is a value greater than zero.
- The **minimum Hamming distance** is the smallest Hamming distance between all possible pairs in a set of words

---

3.4 LINEAR BLOCK CODES

A linear block code is a code in which the exclusive OR (addition modulo-2) of two valid codewords creates another valid codeword.

3.4.1 Minimum Distance for Linear Block Codes

It is simple to find the minimum Hamming distance for a linear block code. The minimum

Hamming distance is the number of Is in the nonzero valid codeword with the smallest number of 1s.

3.4.2 Simple Parity-Check Code

- The most familiar error-detecting code is the simple parity-check code. In this code, a *k-bit* dataword is changed to an n-bit codeword where $n = k + 1$.
- The extra bit, called the parity bit, is selected to make the total number of 1s in the codeword even. Although some implementations specify an odd number of 1s, we discuss the even case. The minimum Hamming

distance for this category is d_{min} =2, which means that the code is a single-bit error-detecting code; it cannot correct any error.

- Our first code (Table) is a parity-check code with k = 2 and n =3. The code in table is also a parity-check code with k =4 and n =5.

Simple parity-check code C(5, 4)

Datawords	Codewords	Datawords	Codewords
0000	00000	1000	10001
0001	00011	1001	10010
0010	00101	1010	10100
0011	00110	1011	10111
0100	01001	1100	11000
0101	01010	1101	11011
0110	01100	1110	11101
0111	01111	1111	11110

- Figure shows a possible structure of an encoder (at the sender) and a decoder (at the receiver).

Encoder and decoder for simple parity-check code

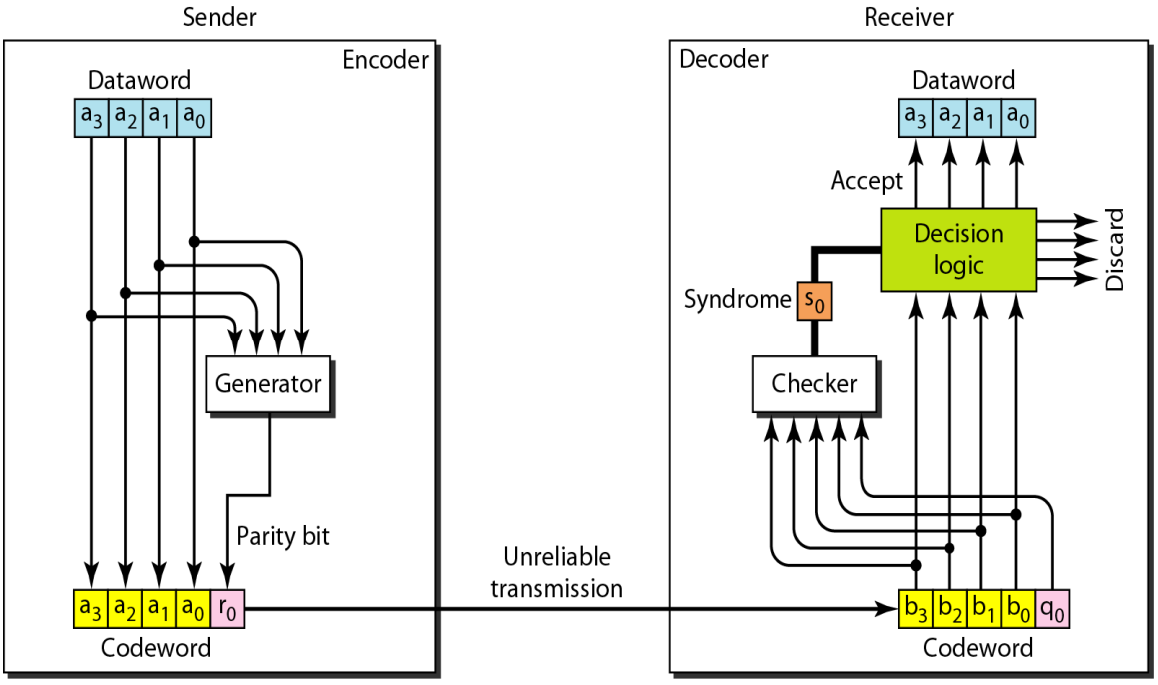


Figure :*Encoder and decoder for simple parity-check code*

- The encoder uses a generator that takes a copy of a 4-bit dataword *(a_0, a_1, a_2,* and *a_3)* and generates a parity bit *r_0* .The dataword bits and the parity bit create the 5-bit codeword. The parity bit that is added makes the number of 1s in the codeword even.
- This is normally done by adding the 4 bits of the dataword (modulo-2); the result is the parity bit. In other words,

$$r_0=a_3+a_2+a_1+a_0 \text{ (modulo-2)}$$

- If the number of 1s is even, the result is 0; if the number of 1s is odd, the result is 1. In both cases, the total number of 1s in the codeword is even. The sender sends the codeword which may be corrupted during transmission. The receiver receives a 5-bit word. The checker at the receiver does the same thing as the generator in the sender with one exception: The addition is done over all 5 bits. The result, which is called the syndrome, is just 1 bit. The syndrome is 0 when the number of 1s in the received codeword is even; otherwise, it is 1.

$$S_0=b_3+b_2+b_1+b_0+q_0 \text{ (modulo-2)}$$

- The syndrome is passed to the decision logic analyzer. If the syndrome is 0, there is no error in the received codeword; the data portion of the received codeword is accepted as the dataword; if the syndrome is 1, the data portion of the received codeword is discarded. The dataword is not created.

3.4.3 Hamming Codes

- Now let us discuss a category of error-correcting codes called Hamming codes. These codes were originally designed with $d_{min} = 3$, which means that they can detect up to two errors or correct one single error. Although there are some Hamming codes that can correct more than one error, our discussion focuses on the single-bit error-correcting code.
- First let us find the relationship between *n* and *k* in a Hamming code. We need to choose an integer $m >= 3$. The values of nand *k* are then calculated from m as $n = 2^m - 1$ and $k = n - m$. The number of check bits $r = m$.

- For example, if $m = 3$, then $n = 7$ and $k = 4$. This is a Hamming code $C(7, 4)$ with $d_{min} = 3$. Below table shows the datawords and codewords for this code.

Datawords	Codewords	Datawords	Codewords
0000	0000000	1000	1000110
0001	0001101	1001	1001011
0010	0010111	1010	1010001
0011	0011010	1011	10111 00
0100	0100011	1100	1100101
0101	01011 10	1101	1101000
0110	0110100	1110	1110010
0111	0111001	1111	1111111

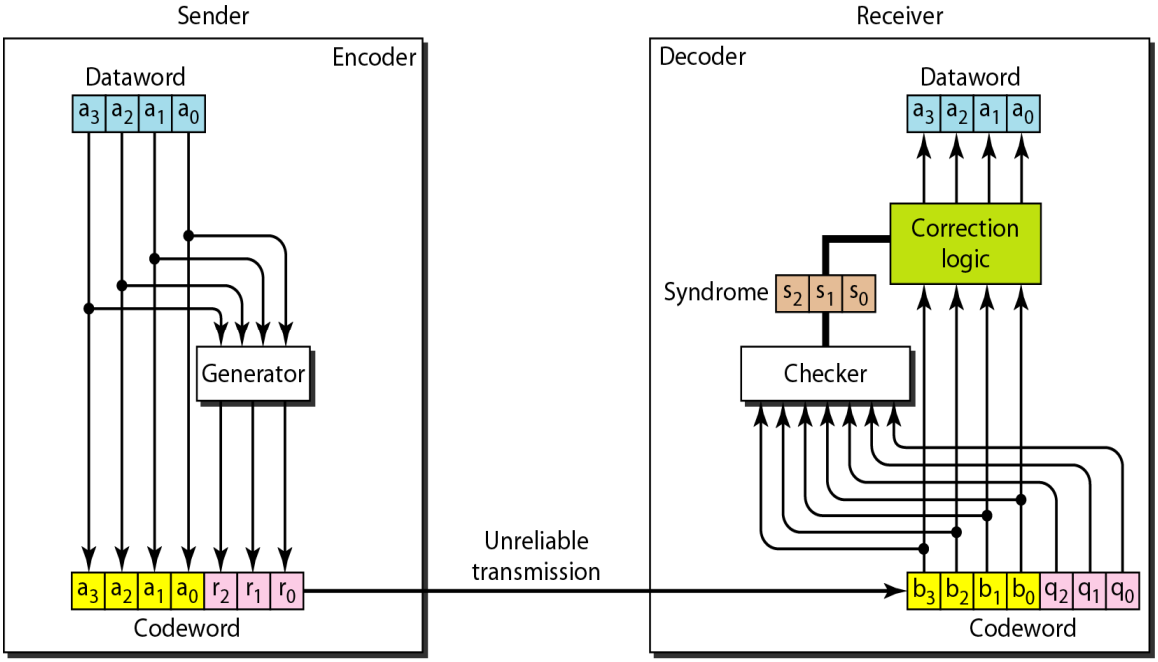


Figure: The structure of Encoder and Decoder for a Hamming code

- A copy of a 4-bit dataword is fed into the generator that creates three parity checks r_0, r_1 and r_2 as shown below:

r0=a2+a1+a0 modul02
r1=a3+a2+a1 modul02
r2=a1+a0+a3 modul02

- The checker in the decoder creates a 3-bit syndrome ($s_2s_1s_0$) in which each bit is the parity check for 4 out of the 7 bits in the received codeword:

S0=b2+b1+b0+q0 modulo2
S1=b3+b2+b1+q1 modulo2

S2=b1+b0+b3+q2 modulo2

Syndrome	000	001	010	011	100	101	110	111
Error	None	q_0	q_1	b_2	q_2	b_0	b_3	b_1

Table :*Logical decision made by the correction logic analyzer*

3.4.4Performance

- A Hamming code can only correct a single error or detect a double error. However, there is a way to make it detect a burst error, as shown in figure.
- The key is to split a burst error between several codewords, one error for each codeword. In data communications, we normally send a packet or a frame of data. To make the Hamming code respond to a burst error of size *N,* we need to make *N* codewords out of our frame. Then, instead of sending one codeword at a time, we arrange the codewords in a table and send the bits in the table a column at a time.
- In figure, the bits are sent column by column (from the left). In each column, the bits are sent from the bottom to the top. In this way, a frame is made out of the four codewords and sent to the receiver. Figure shows that when a burst error of size 4 corrupts the frame, only 1 bit from each codeword is corrupted. The corrupted bit in each codeword can then easily be corrected at the receiver.

3.5CYCLIC CODES

- Cyclic codes are special linear block codes with one extra property. In a cyclic code, if a codeword is cyclically shifted (rotated), the result is another codeword.
- For example, if 1011000 is a codeword and we cyclically left-shift, then 0110001 is also a codeword.
- In this case, if we call the bits in the first word a_0 to a_6 ,and the bits in the second word b_0 to b_6, we can shift the bits by using the following:
- In the rightmost equation, the last bit of the first word is wrapped around and becomes the first bit of the second word.

3.5.1Cyclic Redundancy Check

- We can create cyclic codes to correct errors. However, the theoretical background required is beyond the scope of this book. In this section, we simply discuss a category of cyclic codes called the cyclic redundancy check (CRC) that is used in networks such as LANs and WANs.

- Table shows an example of a CRC code. We can see both the linear and cyclic properties of this code.

A CRC code with C(7, 4)

Dataword	*Codeword*	*Dataword*	*Codeword*
0000	0000000	1000	1000101
0001	0001011	1001	1001110
0010	0010110	1010	1010011
0011	0011101	1011	1011000
0100	0100111	1100	1100010
0101	0101100	1101	1101001
0110	0110001	1110	1110100
0111	0111010	1111	1111111

- In the encoder, the dataword has *k* bits (4 here); the codeword has *n* bits (7 here). The size of the dataword is augmented by adding *n* - *k* (3 here) Os to the right-hand side of the word. The n-bit result is fed into the generator.

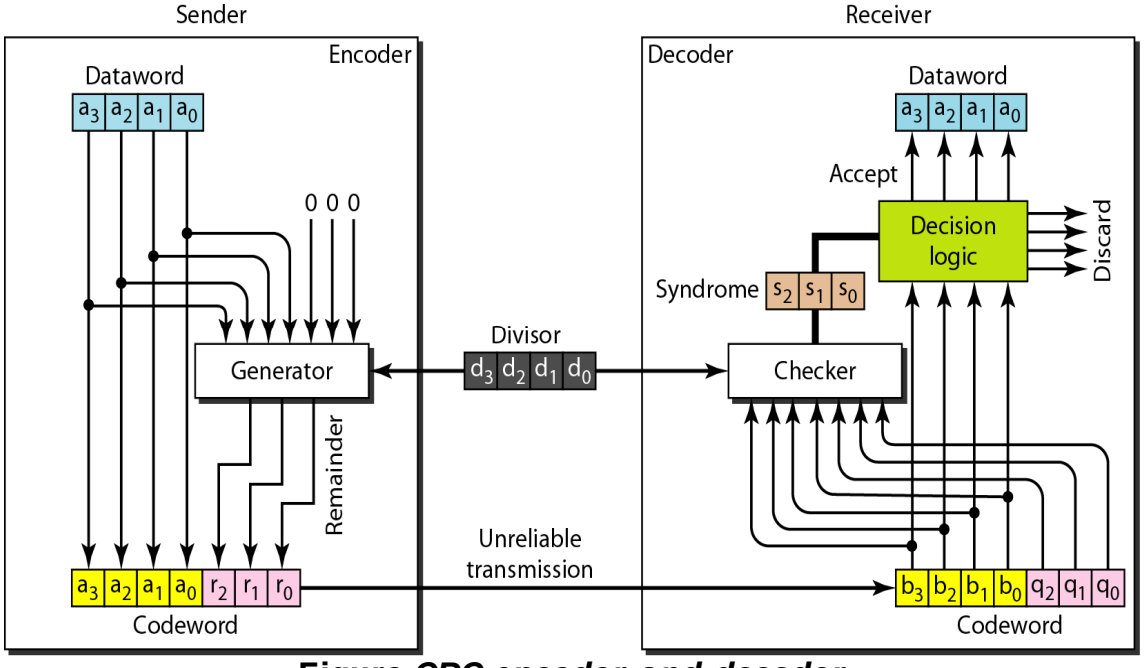


Figure:*CRC encoder and decoder*

- The generator uses a divisor of size *n* - *k* + 1(4 here), predefined and agreed upon. The generator divides the augmented dataword by the divisor (modulo-2 division). The quotient of the division is discarded; the remainder $(r_2r_1r_0)$ is appended to the dataword to create the codeword.

- The decoder receives the possibly corrupted codeword. A copy of all *n* bits is fed to the checker which is a replica of the generator. The remainder produced by the checker is a syndrome of *n* - *k* (3 here) bits, which is fed to the decision logic analyzer. The analyzer has a simple function. If the syndrome bits are all as, the 4 leftmost bits of the codeword are accepted as the dataword (interpreted as no error); otherwise, the 4 bits are discarded (error).

Encoder

- Let us take a closer look at the encoder. The encoder takes the dataword and augments it with *n* - *k* number of 0s. It then divides the augmented dataword by the divisor, as shown in figure.

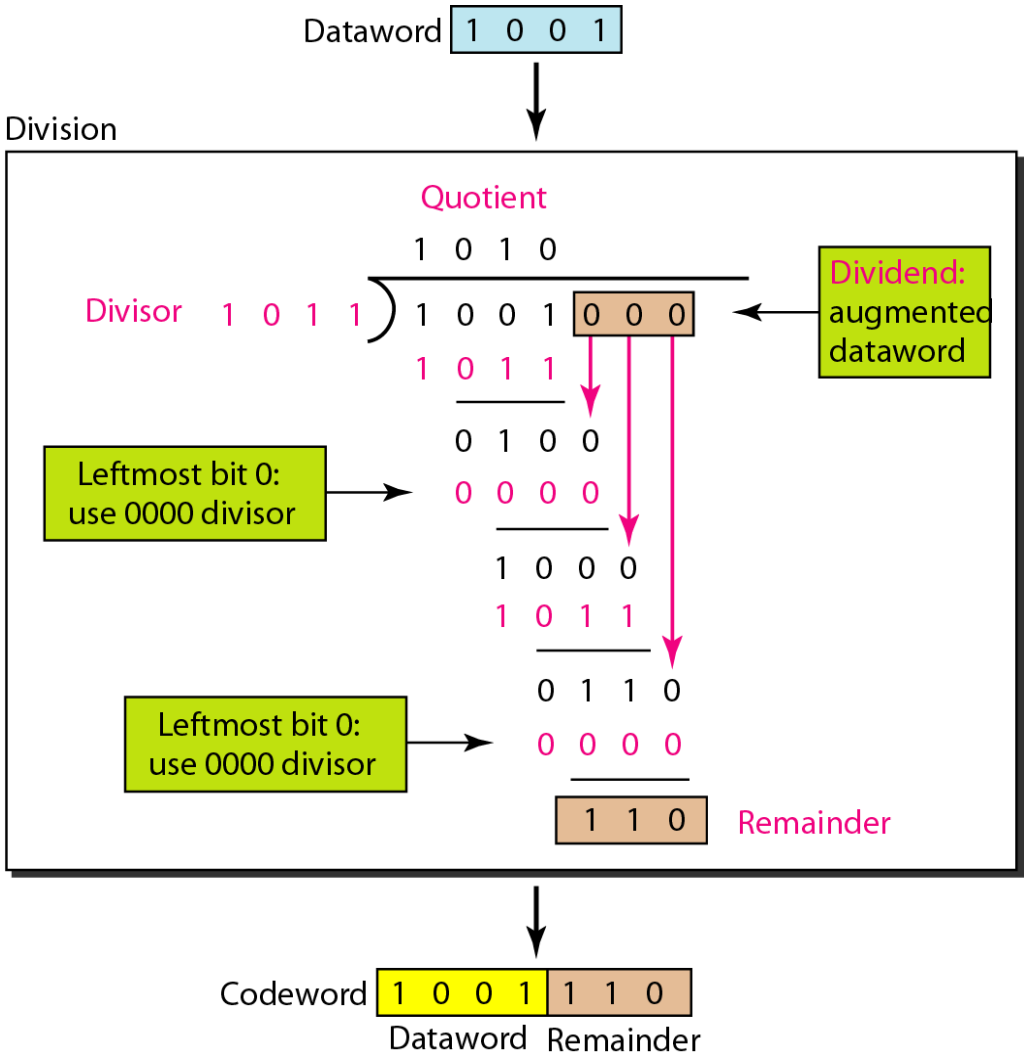


Figure: *Division in CRC encoder*

- The process of modulo-2 binary division is the same as the familiar division process

- As in decimal division, the process is done step by step. In each step, a copy of the divisor is XORed with the 4 bits of the dividend. The result of the XOR operation (remainder) is 3 bits (in this case), which is used for the next step after 1 extra bit is pulled down to make it 4 bits long.
- There is one important point we need to remember in this type of division. If the leftmost bit of the dividend (or the part used in each step) is 0, the step cannot use the regular divisor; we need to use an all-0s divisor. When there are no bits left to pull down, we have a result. The 3-bit remainder forms the check bits *(r2' rl'* and *ro).* They are appended to the dataword to create the codeword.

Decoder

- The codeword can change during transmission. The decoder does the same division process as the encoder. The remainder of the division is the syndrome. If the syndrome is all 0s, there is no error; the dataword is separated from the received codeword and accepted. Otherwise, everything is discarded. Figure shows two cases: The lefthand figure shows the value of syndrome when no error has occurred; the syndrome is 000. The right-hand part of the figure shows the case in which there is one single error.
  The syndrome is not all 0s (it is 011).

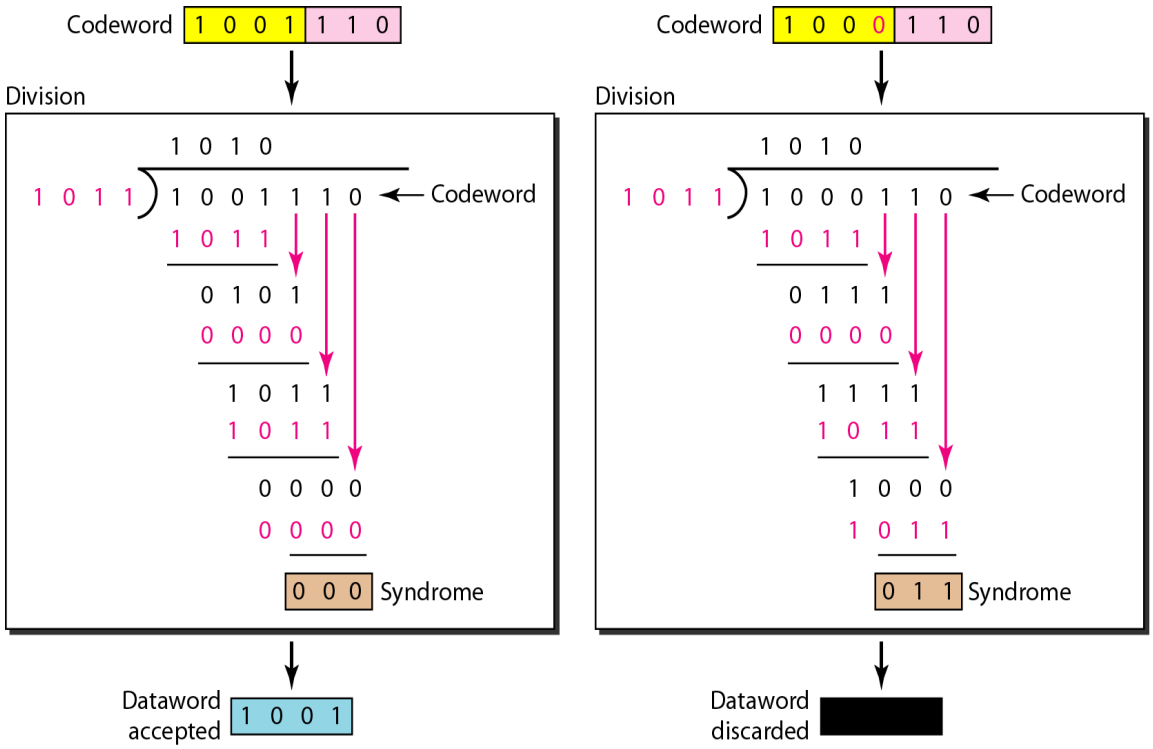


Figure: *Division in the CRC decoder for two cases*

3.5.2 Polynomials

- A better way to understand cyclic codes and how they can be analyzed is to represent them as polynomials.
- A pattern of 0s and 1s can be represented as a **polynomial** with coefficients of 0 and 1. The power of each term shows the position of the bit; the coefficient shows the value of the bit. Figure(a)a shows a binary pattern and its polynomial representation.
- In figure(b) can be shortened by removing all terms with zero coefficients and replacing x1 by *x* and *x0* by 1.

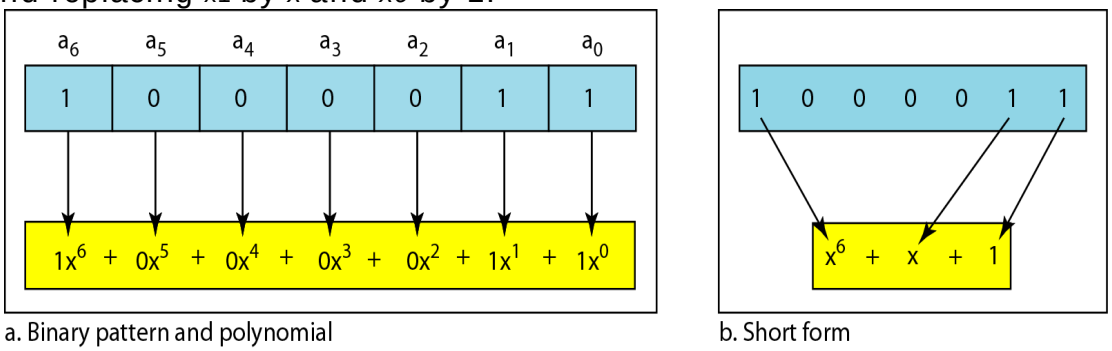


Figure :*A polynomial to represent a binary word*

Degree of a Polynomial

- The degree of a polynomial is the highest power in the polynomial. For example, the degree of the polynomial $x^6 + x + 1$ is 6. Note that the degree of a polynomial is 1 less that the number of bits in the pattern. The bit pattern in this case has 7 bits.

Adding and Subtracting Polynomials

- Adding and subtracting polynomials in mathematics are done by adding or subtracting the coefficients of terms with the same power. In our case, the coefficients are only 0 and 1, and adding is in modulo-2. This has two consequences. First, addition and subtraction are the same.
- Second, adding or subtracting is done by combining terms and deleting pairs of identical terms. For example, adding $x^5 + x^4 + x^2$ and $x^6 + x^4 + x^2$ gives just $x^6 + x^5$. The terms x^4 and x^2 are deleted. However, note that if we add, for example, three polynomials and we get x^2 three times, we delete a pair of them and keep the third.

Cyclic Code Encoder Using Polynomials

- The dataword 1001 is represented as $x^3 + 1$. The divisor 1011 is represented as $x^3 + x + 1$. To find the augmented dataword, we have left-shifted the dataword 3 bits (multiplying by x^3) The result is $x^6 + x^3$. Division is straightforward.
- We divide the first term of the dividend, x^6, by the first term of the divisor, x^3. The first term of the quotient is then x^6/x^3, or x^3. Then we multiply *x3* by the divisor and subtract, the result from the dividend. The result is x^4,

with a degree greater than the divisor's degree; we continue to divide until the degree of the remainder is less than the degree of the divisor.

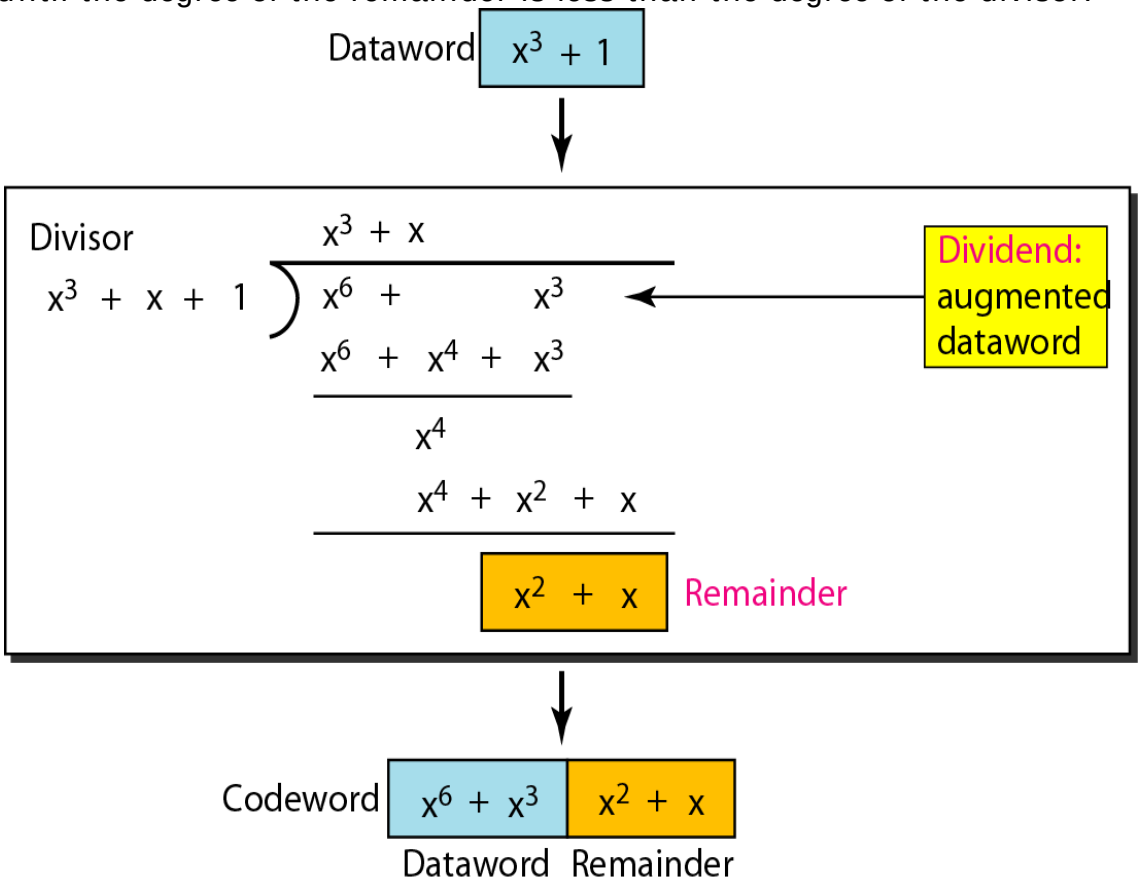


Figure: CRC division using polynomials

Cyclic Code Analysis

- We can analyze a cyclic code to find its capabilities by using polynomials. We define the following, *wheref(x)* is a polynomial with binary coefficients. Dataword: *d(x)* Syndrome: *s(x)* Codeword: *c(x)* Error: *e(x)* Generator: *g(x)*
- If *s(x)* is not zero, then one or more bits is corrupted. However, if *s(x)* is zero, either no bit is corrupted or the decoder failed to detect any errors.
- In our analysis we want to find the criteria that must be imposed on the generator, *g(x)* to detect the type of error we especially want to be detected. Let us first find the relationship among the sent codeword, error, received codeword, and the generator. We can say

$$\text{Received codeword} = c(x) + e(x)$$

- In other words, the received codeword is the sum of the sent codeword and the error. The receiver divides the received codeword by *g(x)* to get the syndrome. We can write this as

$$\frac{\text{Received codeword}}{g(x)} = \frac{c(x)}{g(x)} + \frac{e(x)}{g(x)}$$

- The first term at the right-hand side of the equality does not have a remainder (according to the definition of codeword). So the syndrome is actually the remainder of the second term on the right-hand side. If this term does not have a remainder (syndrome =0), either *e(x)* is 0 or *e(x)* is divisible by *g(x)*.

3.6 CHECKSUM

Like linear and cyclic codes, the checksum is based on the concept of redundancy. Several protocols still use the checksum for error detection as we will see in future chapters, although the tendency is to replace it with a CRC. This means that the CRC is also used in layers other than the data link layer.

Idea

The concept of the checksum is not difficult. Let us illustrate it with a few examples.

Case1:

Suppose our data is a list of five 4-bit numbers that we want to send to a destination. In addition to sending these numbers, we send the sum of the numbers. For example, if the set of numbers is (7, 11, 12, 0, 6), we send (7, 11, 12, 0, 6, 36), where 36 is the sum of the original numbers. The receiver adds the five numbers and compares the result with the sum. If the two are the same, the receiver assumes no error, accepts the five numbers, and discards the sum. Otherwise, there is an error somewhere and the data are not accepted.

Case2:

We can make the job of the receiver easier if we send the negative (complement) of the sum, called the checksum. In this case, we send (7, 11, 12, 0, 6, −36). The receiver can add all the numbers received (including the checksum). If the result is 0, it assumes no error; otherwise, there is an error.

One's Complement

- All of our data can be written as a 4-bit word (they are less than 15) except for the checksum. One solution is to use one's complement arithmetic. In this arithmetic, we can represent unsigned numbers between 0 and 2^n - 1 using only *n* bits.
- If the number has more than *n* bits, the extra leftmost bits need to be added to the *n* rightmost bits (wrapping). In one's complement arithmetic, a negative number can be represented by inverting all bits

(changing a 0 to a 1 and a 1 to a 0). This is the same as subtracting the number from 2^n - 1.

- **Example:** How can we represent the number 21 in one's complement arithmetic using only four bits?
  The number 21 in binary is 10101 (it needs five bits). We can wrap the leftmost bit and add it to
  the four rightmost bits. We have (0101 + 1) = 0110 or 6.
- How can we represent the number -6 in one's complement arithmetic using only four bits?

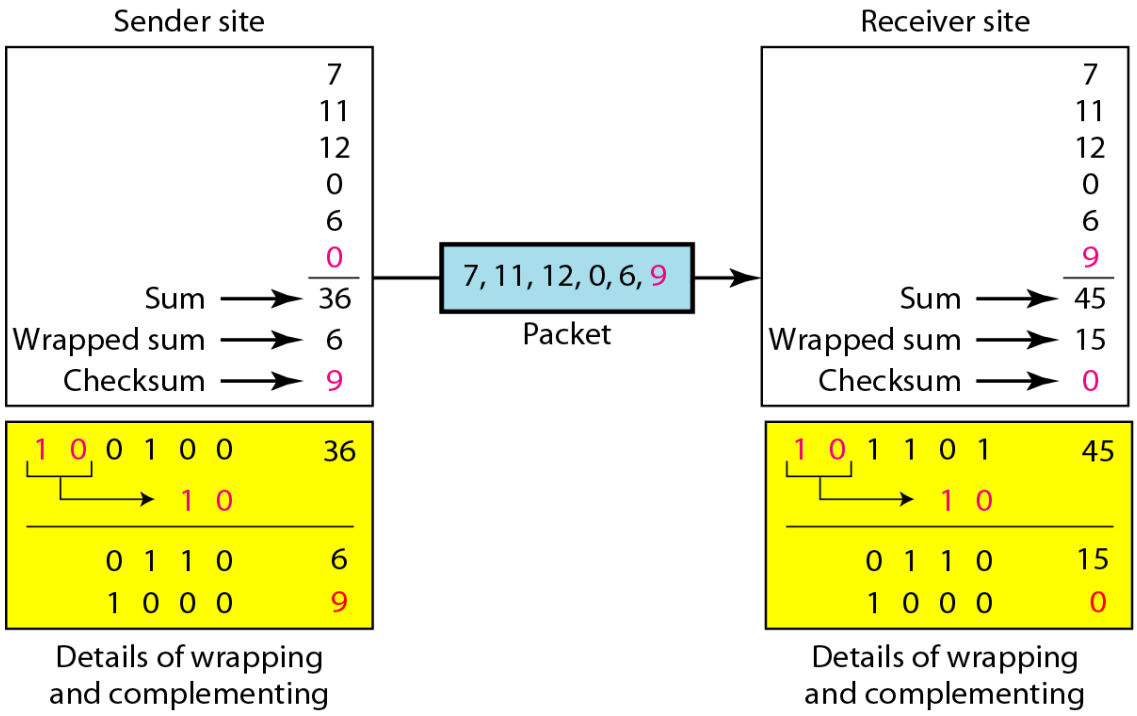
Solution

In one's complement arithmetic, the negative or complement of a number is found by inverting all bits. Positive 6 is 0110; negative 6 is 100I. If we consider only unsigned numbers, this is 9. In other words, the complement of 6 is 9. Another way to find the complement of a number in one's complement arithmetic is to subtract the number from 2^n - 1 (16 - 1 in this case).

Example:

Figure shows the process at the sender and at the receiver. The sender initializes the checksum to 0 and adds all data items and the checksum (the checksum is considered as one data item and is shown in color). The result is 36. However, 36 cannot be expressed in 4 bits. The extra two bits are wrapped and added with the sum to create the wrapped sum value 6. In the figure, we have shown the details in binary. The sum is then complemented, resulting in the checksum value 9 (15 − 6 = 9). The sender now sends six data items to the receiver including the checksum 9.

The receiver follows the same procedure as the sender. It adds all data items (including the checksum); the result is 45. The sum is wrapped and becomes 15. The wrapped sum is complemented and becomes 0. Since the value of the checksum is 0, this means that the data is not corrupted. The receiver drops the checksum and keeps the other data items. If the checksum is not zero, the entire packet is dropped.



Internet Checksum

Traditionally, the Internet has been using a 16-bit checksum. The sender calculates the
checksum by following these steps.

Sender site:

- The message is divided into 16-bit words.
- The value of the checksum word is set to O.
- All words including the checksum are added using one's complement addition.
- The sum is complemented and becomes the checksum.
- The checksum is sent with the data.

Receiver site:

- The message (including checksum) is divided into 16-bit words.
- All words are added using one's complement addition.
- The sum is complemented and becomes the new checksum.
- If the value of checksum is 0, the message is accepted; otherwise, it is rejected

---

3.7 TYPES OF SERVICES PROVIDED TO THE NETWORK LAYER:

- Unacknowledged Connectionless service
- Acknowledged Connectionless service
- Acknowledged Connection-Oriented service

Unacknowledged Connectionless service:

- no recovering of lost or corrupted frame
  - when the error rate is very low
  - real-time traffic, like speech or video
- Losses are taken care of at higher layers
- Used on reliable medium like coax cables or optical fiber, where the error rate is low.
- Appropriate for voice, where delay is worse than bad data.
- It consists of having the source machine send independent frames to the destination machine without having the destination machine acknowledge them.
- Example: Ethernet, Voice over IP, etc. in all the communication channel were real time operation is more important that quality of transmission.

Acknowledged Connectionless service:

- returns information a frame has safely arrived.
  - time-out, resend, frames received twice
  - unreliable channels, such as wireless systems. Useful on unreliable medium like wireless.
- Acknowledgements add delays.
- Adding ack in the DLL rather than in the NL is just an optimization and not a requirement. Leaving it for the NL is inefficient as a large message (packet) has to be resent in that case in contrast to small frames here.
- On reliable channels, like fiber, the overhead associated with the ack is not justified.
- Each frame send by the Data Link layer is acknowledged and the sender knows if a specific frame has been received or lost.
- Typically the protocol uses a specific time period that if has passed without getting acknowledgment it will re-send the frame.
- This service is useful for commutation when an unreliable channel is being utilized (e.g., 802.11 WiFi).
- Network layer does not know frame size of the packets and other restriction of the data link layer. Hence it becomes necessary for data link layer to have some mechanism to optimize the transmission.

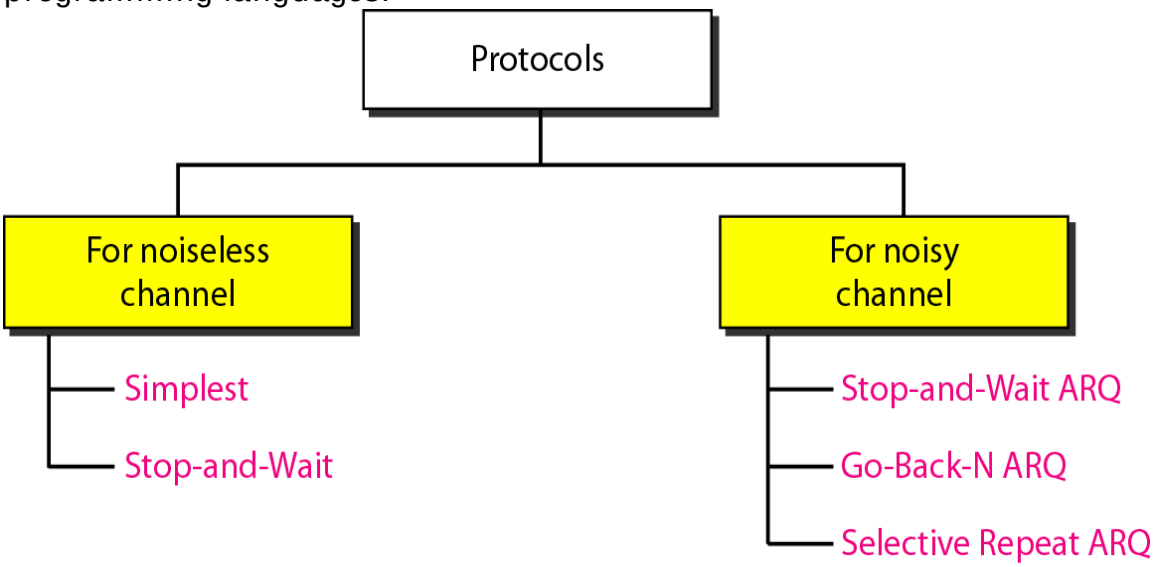
Acknowledged Connection-oriented service

- established connection before any data is sent.
- provides the network layer with a reliable bit stream.
- Most reliable, Guaranteed service –
  - Each frame sent is indeed received
  - Each frame is received exactly once
  - Frames are received in order
- Special care has to be taken to ensure this in connectionless services
- Source and Destination establish a connection first.
- Each frame sent is numbered
  - Data link layer guarantees that each frame sent is indeed received.

  - It guarantees that each frame is received only once and that all frames are received in the correct order.
- Exs: Satellite channel communication, Long-distance telephone communication, etc.

3.8 Elementary DLL protocols

- The data link layer can combine framing, flow control, and error control to achieve the delivery of data from one node to another. The protocols are normally implemented in software by using one of the common programming languages.



- Protocols are classified into various types depending on different channels.

  They are:

  1) noiseless channel
  2) noisy channel

3.8.1 Noiseless channel:

- Protocols for noiseless channel are further classified into two

  i) simplest protocol

  ii) stop-and-wait protocol

3.8.1.1Simplest Protocol

- It is a unidirectional protocol in which data frames are traveling in only one direction-from the sender to receiver. We assume that the receiver can immediately handle any frame it receives with a processing time that is small enough to be negligible.
- The data link layer of the receiver immediately removes the header from the frame and hands the data packet to its network layer, which can also accept the packet immediately.
- In other words, the receiver can never be overwhelmed with incoming frames.

Design

- There is no need for flow control in this scheme. The data link layer at the sender site gets data from its network layer, makes a frame out of the data, and sends it. The data link layer at the receiver site receives a frame from its physical layer, extracts data from the frame, and delivers the data to its network layer.
- The data link layers of the sender and receiver provide transmission services for their network layers. The data link layers use the services provided by their physical layers (such as signaling, multiplexing, and so on) for the physical transmission of bits.
- Figure shows a design.

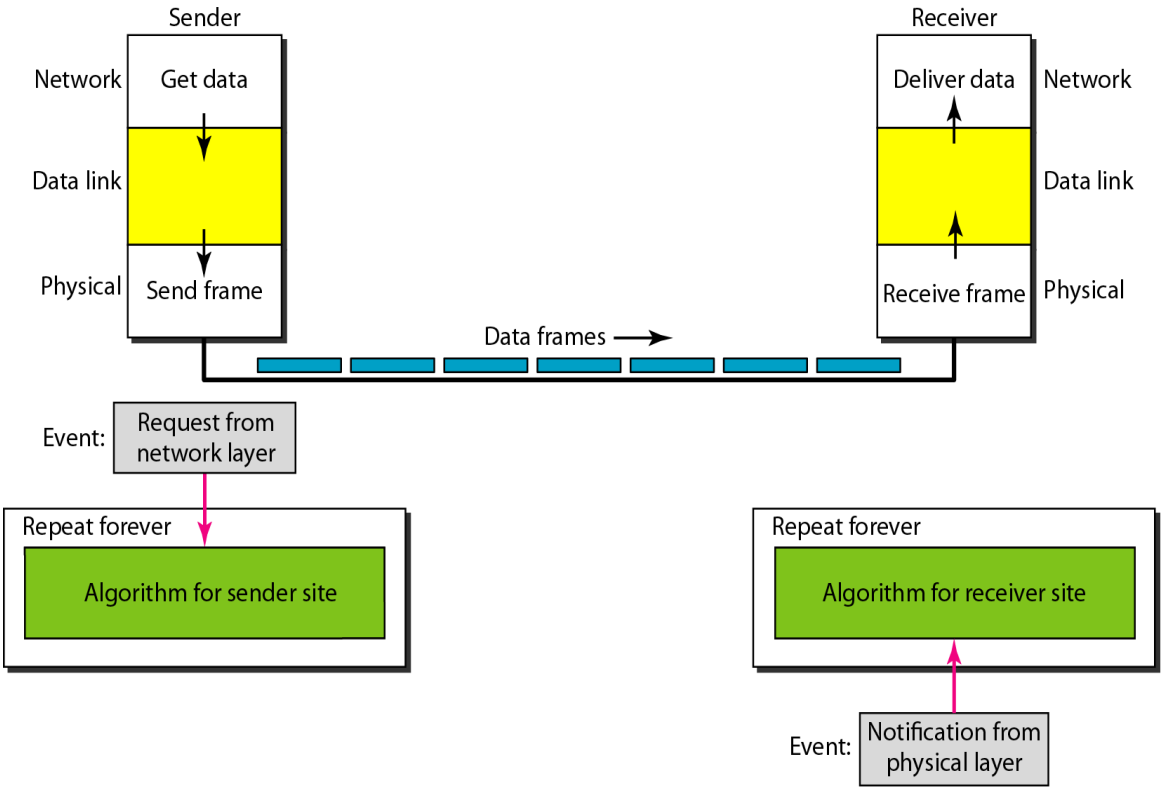


Figure: ***The design of the simplest protocol with no flow or error control***

- The sender site cannot send a frame until its network layer has a data packet to send. The receiver site cannot deliver a data packet to its network layer until a frame arrives.
- If the protocol is implemented as a procedure, we need to introduce the idea of events in the protocol. The procedure at the sender site is constantly running; there is no action until there is a request from the network layer.
- The procedure at the receiver site is also constantly running, but there is no action until notification from the physical layer arrives.

- Both procedures are constantly running because they do not know when the corresponding events will occur.
- Figure shows an example of communication using this protocol. It is very simple. The sender sends a sequence of frames without even thinking about the receiver. To send three frames, three events occur at the sender site and three events at the receiver site. Note that the data frames are shown by tilted boxes; the height of the box defines the transmission time difference between the first bit and the last bit in the frame.

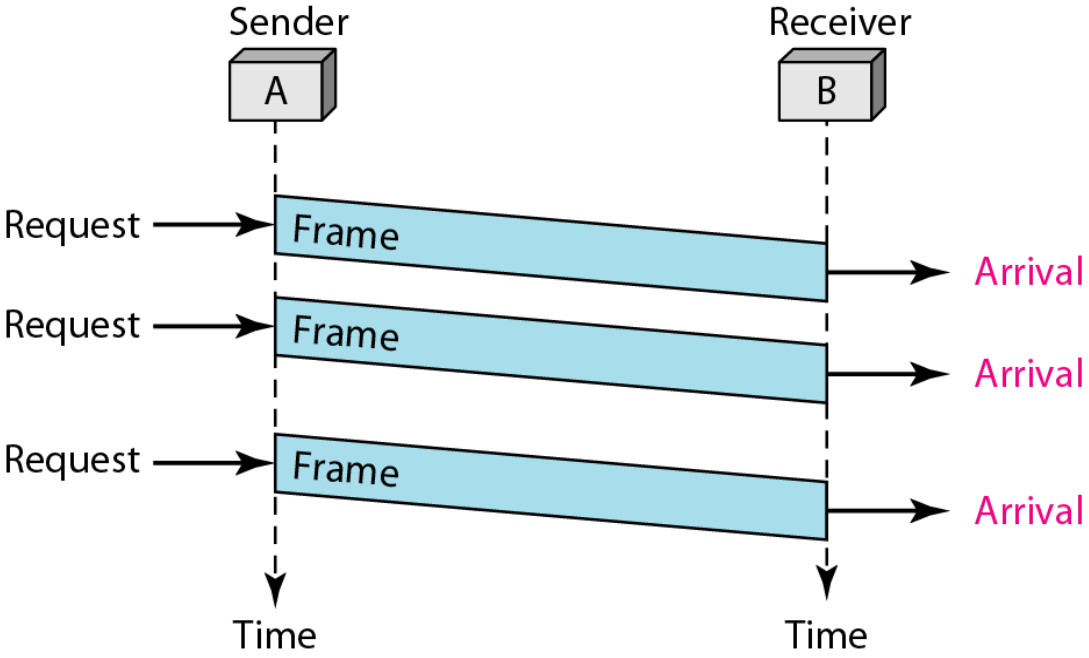


Figure: Flow diagram for above design

3.8.1.2 Stop-and-Wait Protocol

- If data frames arrive at the receiver site faster than they can be processed, the frames must be stored until their use.
- Normally, the receiver does not have enough storage space, especially if it is receiving data from many sources.
- This may result in either the discarding of frames or denial of service. To prevent the receiver from becoming overwhelmed with frames, we somehow need to tell the sender to slow down. There must be feedback from the receiver to the sender.
- The protocol we discuss now is called the Stop-and-Wait Protocol because the sender sends one frame, stops until it receives confirmation from the receiver (okay to go ahead), and then sends the next frame.
- We still have unidirectional communication for data frames, but auxiliary ACK frames (simple tokens of acknowledgment) travel from the other direction.

Design

- Figure illustrates the mechanism. Comparing this figure with above figure, we can see the traffic on the forward channel (from sender to receiver) and the reverse channel. At any time, there is either one data frame on the forward channel or one ACK frame on the reverse channel. We therefore need a half-duplex link.

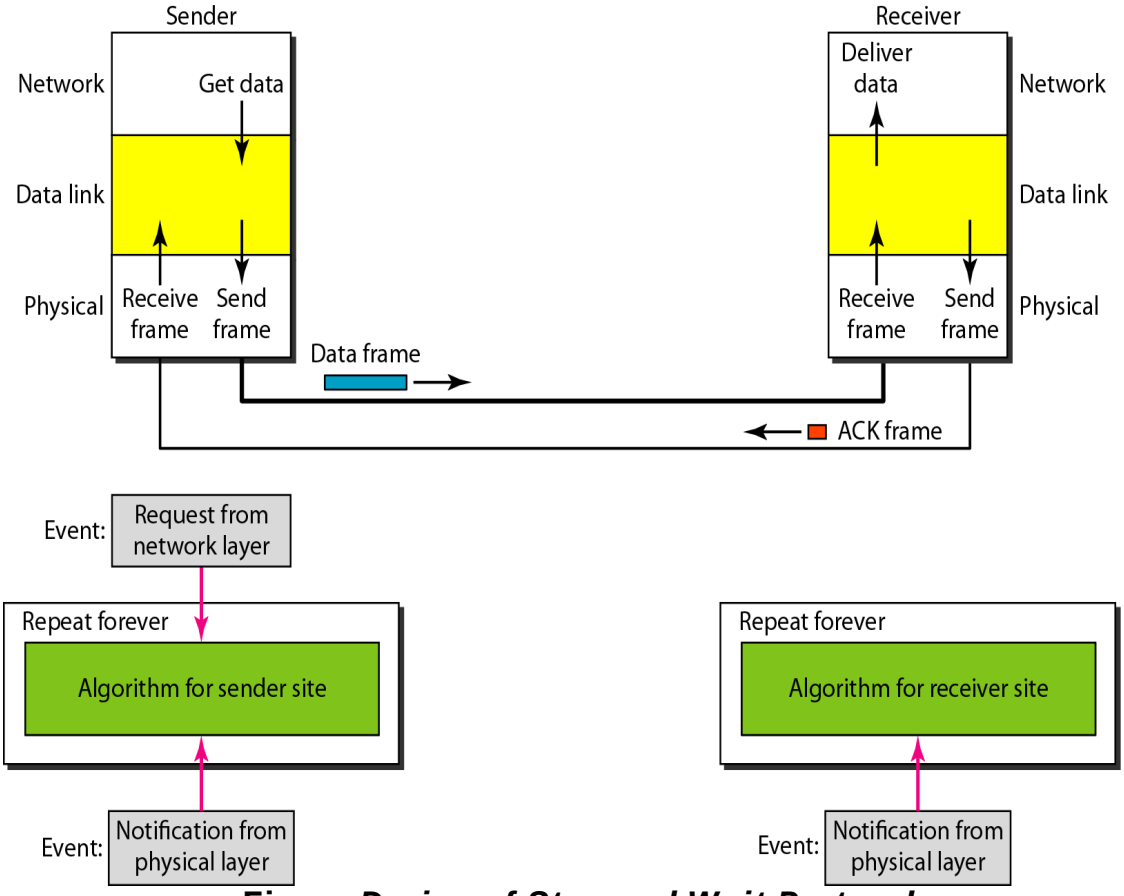


Figure:*Design of Stop-and-Wait Protocol*

Analysis

- Here two events can occur: a request from the network layer or an arrival notification from the physical layer. The responses to these events must alternate. In other words, after a frame is sent, the algorithm must ignore another network layer request until that frame is acknowledged.
- We know that two arrival events cannot happen one after another because the channel is error-free and does not duplicate the frames. The requests from the network layer, however, may happen one after another without an arrival event in between.
- We need somehow to prevent the immediate sending of the data frame. Although there are several methods, we have used a simple *canSend* variable that can either be true or false.

- When a frame is sent, the variable is set to false to indicate that a new network request cannot be sent until *canSend* is true. When an ACK is received, canSend is set to true to allow the sending of the next frame.
- Figure shows an example of communication using this protocol. It is still very simple. The sender sends one frame and waits for feedback from the receiver. When the ACK arrives, the sender sends the next frame. Note that sending two frames in the protocol involves the sender in four events and the receiver in two events.

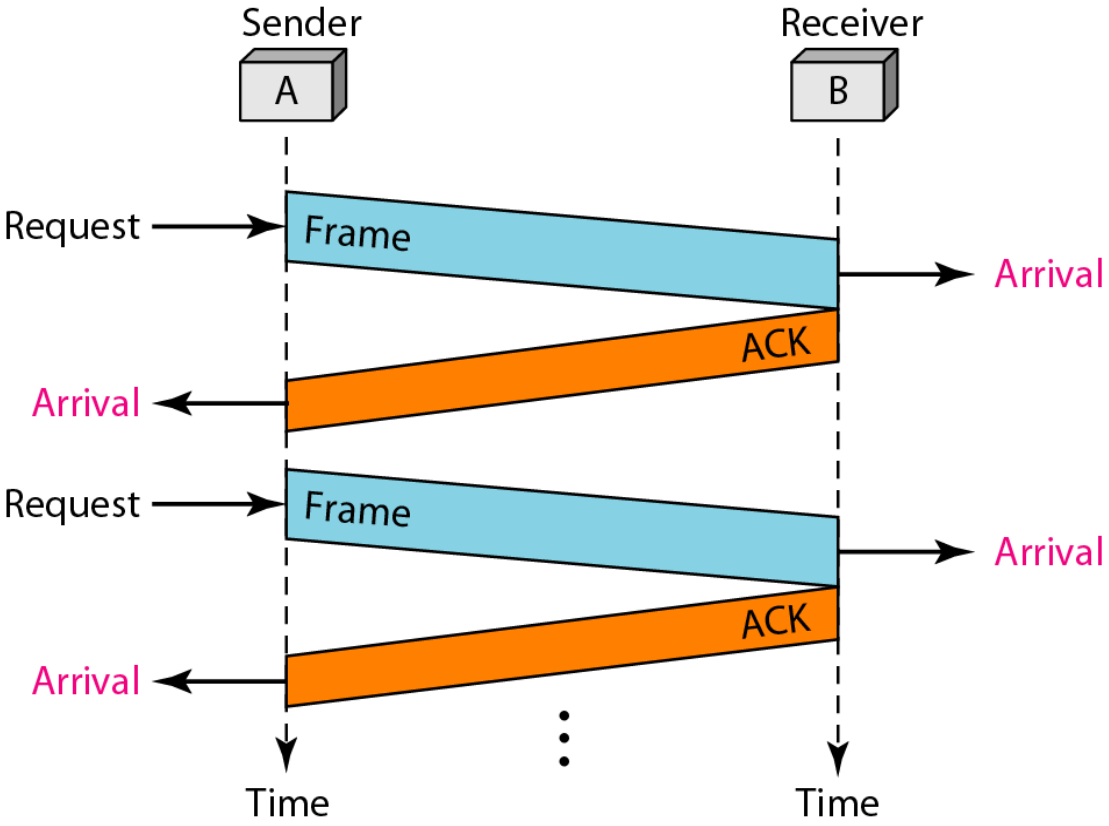


Figure:*Flow diagram for above design*

3.8.2 NOISY CHANNELS

Although the Stop-and-Wait Protocol gives us an idea of how to add flow control to its predecessor, noiseless channels are nonexistent.

3.8.2.1 Stop-and-Wait Automatic Repeat Request

- The Stop-and-Wait Automatic Repeat Request (Stop-andWait ARQ), adds a simple error control mechanism to the Stop-and-Wait Protocol.
- To detect and correct corrupted frames, we need to add redundancy bits to our data frame.When the frame arrives at the receiver site, it is checked and if it is corrupted, it is silently discarded. The detection of errors in this protocol is manifested by the silence of the receiver.

- Lost frames are more difficult to handle than corrupted ones. In our previous protocols,
  there was no way to identify a frame. The received frame could be the correct one, or a duplicate, or a frame out of order. The solution is to number the frames. When the receiver receives a data frame that is out of order, this means that frames were either
  lost or duplicated.
- The completed and lost frames need to be resent in this protocol. If the receiver does
  not respond when there is an error, how can the sender know which frame to resend? To remedy this problem, the sender keeps a copy of the sent frame. At the same time, it starts a timer. If the timer expires and there is no ACK for the sent frame, the frame is resent, the copy is held, and the timer is restarted. Since the protocol uses the stop-and-wait mechanism, there is only one specific frame that needs an ACK even though several copies of the same frame can be in the network.
- Since an ACK frame can also be corrupted and lost, it too needs redundancy bits and a sequence number. The ACK frame for this protocol has a sequence number field.
  In this protocol, the sender simply discards a corrupted ACK frame or ignores an out-of-order one.

Sequence Numbers

The protocol specifies that frames need to be numbered. This is done by using sequence numbers. A field is added to the data frame to hold the sequence number of that frame.

One important consideration is the range of the sequence numbers. Since we want to minimize the frame size, we look for the smallest range that provides unambiguous communication. The sequence numbers of course can wrap around. For example, if we decide that the field is *m* bits long, the sequence numbers start from 0, go to 2^m - 1, and then are repeated.

Let us reason out the range of sequence numbers we need. Assume we have used *x* as a sequence number; we only need to use *x* + 1 after that. There is no need for *x* + 2. To show this, assume that the sender has sent the frame numbered *x.* Three things can happen.

1. The frame arrives safe and sound at the receiver site; the receiver sends an acknowledgment.

The acknowledgment arrives at the sender site, causing the sender to send the next frame numbered *x* + 1.

2. The frame arrives safe and sound at the receiver site; the receiver sends an acknowledgment,

but the acknowledgment is corrupted or lost. The sender resends the frame (numbered *x)* after the time-out. Note that the frame here is a duplicate. The receiver can recognize this fact because it expects frame *x* + I but frame *x* was received.

3. The frame is corrupted or never arrives at the receiver site; the sender resends the frame (numbered *x)* after the time-out.

We can see that there is a need for sequence numbers *x* and *x* + I because the receiver needs to distinguish between case 1 and case 2. But there is no need for a frame to be numbered *x* + 2. In case 1, the frame can be numbered *x* again because frames *x* and *x* + 1 are acknowledged and there is no ambiguity at either site. In cases 2 and 3, the new frame is *x* + I, not *x* + 2. If only *x* and *x* + 1 are needed, we can let *x* = 0 and *x* +1 == 1. This means that the sequence is 0, 1, 0, 1, 0, and so on.

In Stop-and-Wait ARQ we use sequence numbers to number the frames. The sequence numbers are based on modul0-2 arithmetic.

Acknowledgment Numbers

Since the sequence numbers must be suitable for both data frames and ACK frames, we use this convention: The acknowledgment numbers always announce the sequence number of the next frame expected by the receiver. For example, if frame 0 has arrived safe and sound, the receiver sends an ACK frame with acknowledgment 1 (meaning frame 1 is expected next). If frame 1 has arrived safe and sound, the receiver sends an ACK frame with acknowledgment 0 (meaning frame 0 is expected).

In Stop-and-Wait ARQ the acknowledgment number always announces in modul0-2 arithmetic the sequence number of the next frame expected.

Design

Figure shows the design of the Stop-and-WaitARQ Protocol. The sending device keeps a copy of the last frame transmitted until it receives an acknowledgment for that frame. A data frames uses a seqNo (sequence number); an ACK frame uses an ackNo (acknowledgment number). The sender has a control variable, which we call *Sn* (sender, next frame to send), that holds the sequence number for the next frame to be sent (0 or 1).

The receiver has a control variable, which we call *Rn* (receiver, next frame expected), that holds the number of the next frame expected. When a frame is sent, the value of *Sn* is incremented (modulo-2), which means if it is 0, it becomes 1 and vice versa. When a frame is received, the value of *Rn* is incremented (modulo-2), which means if it is 0, it becomes 1 and vice versa. Three events can happen at the sender site; one event can happen at the receiver site. Variable *Sn* points to the slot that matches the sequence number of the frame that has been sent, but not acknowledged; *Rn* points to the slot that matches the sequence number of the expected frame.

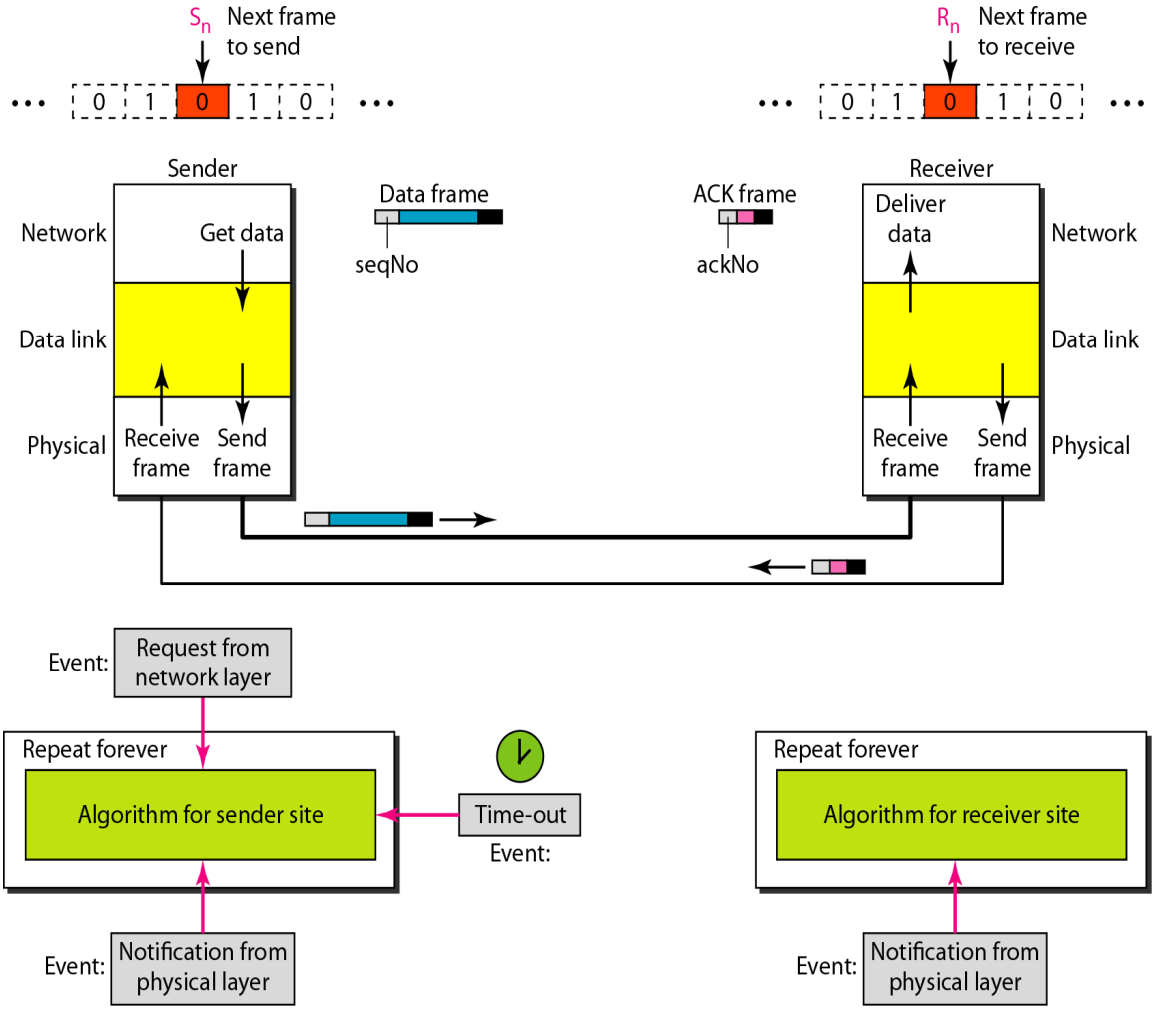


Figure :*Design of the Stop-and-Wait ARQ Protocol*

Below figure shows an example of Stop-and-Wait ARQ. Frame 0 is sent and acknowledged. Frame 1 is lost and resent after the time-out. The resent frame 1 is acknowledged and the timer stops. Frame 0 is sent and acknowledged, but the acknowledgment is lost. The sender has no idea if the frame or the acknowledgment is lost, so after the time-out, it resends frame 0, which is acknowledged.

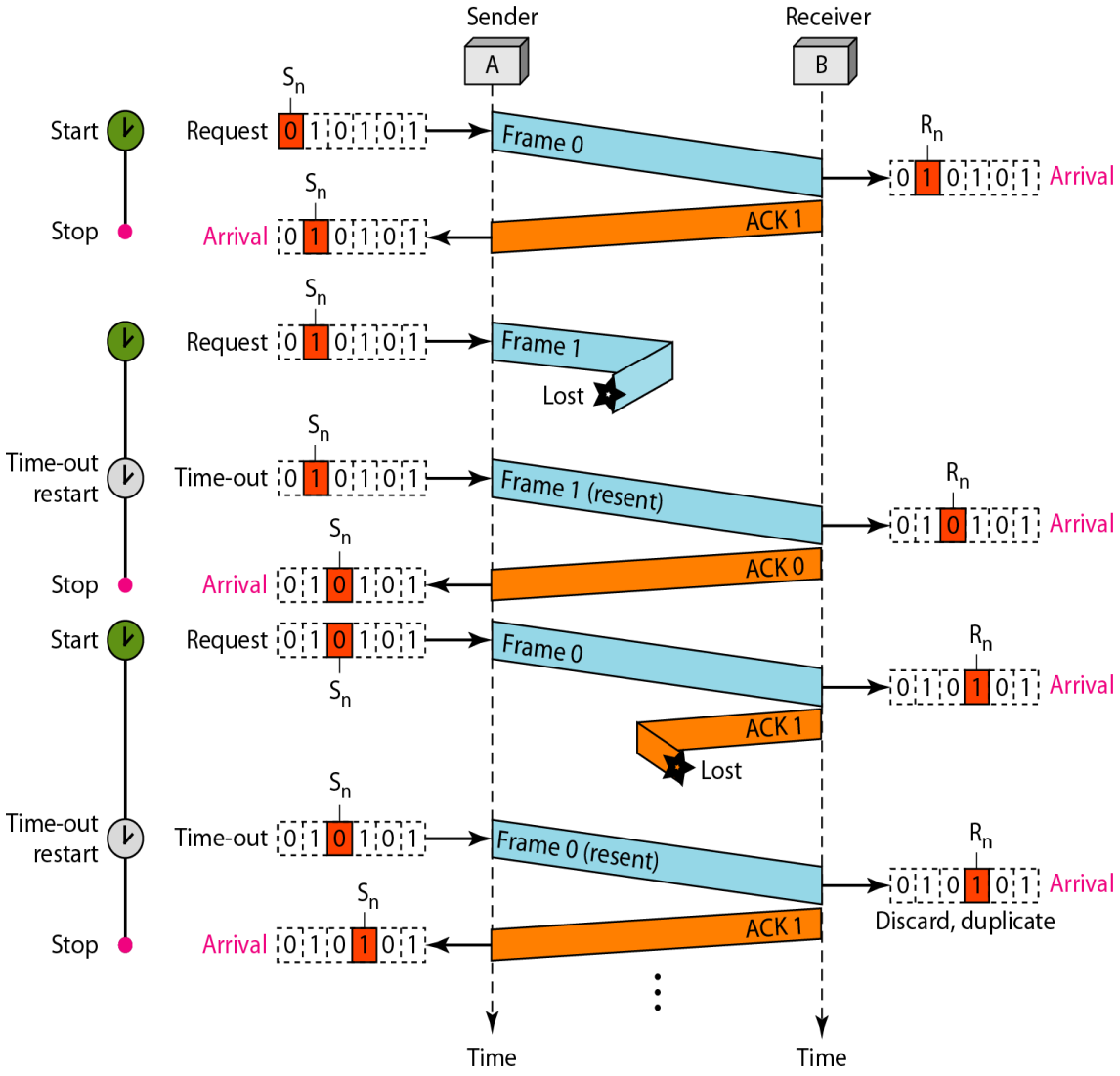


Figure :*Flow diagram for above design*

UNIT-III

Assignment-Cum-Tutorial Questions

SECTION-A

Objective Questions

1. When 2 or more bits in a data unit has been changed during the transmission, the error is called [ ]
   a. random error b. burst error
   c. inverted error d. no error
2. CRC stands for [ ]
   a. cyclic redundancy check b. code repeat check
   c. code redundancy check d. cyclic repeat check
3. The ..... layer provides a well defined service interface to the network layer, determining how the bits of the physical layer are grouped into frames. [ ]
   a. Data Link b. Physical
   c. Network d. Session
4. The different types of services provided by data link layer is/are .
   a. Unacknowledged connectionless service [ ]
   b. Acknowledged connectionless service
   c. Acknowledged connection oriented service
   d. All of the above.
5. ..... is used to indicate to the network layer that an event has happened, for example, establishment or release of a connection.
   a. Request b. Indication [ ]
   c. Response d. Confirm
6. In ..... we need to know the exact number of bits that are corrected and more importantly, their location in the message. [ ]
   a. error searching b. error detection
   c. error correction d. error transmission
7. ..... is the process in which the receiver tries to guess the message by using redundant bits. [ ]

a. Forward error correction b. Backward error correction
c. Transmission d. Retransmission

8. In block coding, we divide our message into blocks, each of k bits, called ..... [ ]
a. Dataword b. Generator
c. Codeword d. Checker

9. ..... in the data link layer separates a message from one source to a destination, or from other messages to other destinations, by adding a sender address and a destination address. [ ]
a. Transforming b. Framing
c. Separating d. Messaging

10. In ....., there is no need for defining the boundaries of the frames; the size itself can be used a delimiter. [ ]
a. Standard Size Framing b. Fixed Size Framing
c. Variable Size Framing d. Constant Size Framing

11. Which of the following is/are the methods used for carrying out framing. [ ]
a. Character count
b. Starting and ending characters, with character stuffing.
c. Starting and ending flags with bit stuffing.
d. All of the above

12. In ..... the source machine sends independent frames to the destination machine without having the destination machine acknowledge them. [ ]

a. Unacknowledged connectionless service

b. Acknowledged connectionless service

c. Acknowledged connection oriented service

d. Unacknowledged connection oriented service

13. ..... is the most sophisticated service provided by the data link layer to the network layer. The source and destination machines establish a connection before any data transfer takes place. [ ]
a. Unacknowledged connectionless service

b. Acknowledged connectionless service

c. Acknowledged connection oriented service

d. Unacknowledged connection oriented service

14..... is prevalent in LANs, we need a way to define the end of the frame and the beginning of the next. [ ]

a. Standard Size Framing b. Fixed Size Framing

c. Variable Size Framing d. Constant Size Framing

15. The checksum of 1111 and 1111 is _____. [ ]

a. 0000 b. 1111 c. 1110 d. 0111

16. In cyclic redundancy checking, the divisor is _____ the CRC. [ ]

a. one bit less than b. one bit more than

c. the same size as d . unequal size

17. For hamming distance d_{min} and number of errors D, the condition for receiving invalid codeword is [ ]

a. $D <= d_{min} + 1$ b.$D <= d_{min}-1$

c. $D <= 1- d_{min}$ d. $D <= d_{min}$

18. In modulo-2 arithmetic, _____ give the same results. [ ]

a. addition and subtraction b. addition and multiplication

c. addition and division d. multiplication and subtraction

19. In cyclic redundancy checking, what is the CRC? [ ]

a. The quotient b. The dividend

c. The divisor d. The remainder

20. Which error detection method consists of just one redundant bit per data unit? [ ]

a. CRC b. Checksum

c. Simple parity check d. Two-dimensional parity check

21. In _____ coding, we divide our message into blocks, each of k bits, called ___. [ ]

a. block; blockwords b. block; datawords

c. linear; datawords d. none of the above

22. An error-detecting code inserted as a field in a block of data to be transmitted is known as [ ]

a. Frame check sequence b. Error detecting code

c.Checksum d.flow control

SECTION-B

SUBJECTIVE QUESTIONS

1. What is the Hamming distance? What is the minimum Hamming distance?
2. Define CRC. Describe how CRC works for error detection with an example
3. Define framing and the reason for its need
4. Describe the two protocols in noiseless channels with their algorithms
5. Compare and contrast byte-stuffing and bit-stuffing. Which technique is used in byte-oriented protocols? which technique is used in bit-oriented protocols?
6. Briefly describe the services provided by the data link layer
7. Explain the working of checksum.
8. Explain the working of simple parity checksum with encoder and decoder.
9. What is the remainder obtained by dividing X^7+X^5+1 by the generator polynomial X^3+1?
10. Perform CRC on the data 11100011 using the generator polynomial X^4+x^3+1.

SECTION-C

QUESTIONS AT THE LEVEL OF GATE

The message 11001001 is to be transmitted using the CRC polynomial x^3 + 1 to protect it from errors. The message that should be transmitted is:

a. 11001001000 [ ]
b. 11001001011
c. 11001010
d. 110010010011 **[GATE 2007]**

2. A bit-stuffing based framing protocol uses an 8-bit delimiter pattern of 01111110. If the output bit-string after stuffing is 01111100101, then the input bit-string is

a. 0111110100 [ ]
b. 0111110101
c. 0111111101
d. 0111111111 **[GATE 2014]**

UNIT-IV

SLIDING WINDOW PROTOCOLS

The sliding window is an abstract concept that defines the range of sequence numbers that is the concern of the sender and receiver.

In other words, the sender and receiver need to deal with only part of the possible sequence numbers.

The range which is the concern of the sender is called the send sliding window; the range that is the concern of the receiver is called the receive sliding window.

The send window is an imaginary box covering the sequence numbers of the data frames which can be in transit.

In each window position, some of these sequence numbers define the frames that have been sent; others define those that can be sent.

The maximum size of the window is $2^m - 1$.

Figure shows a sliding window of size **15** ($m = 4$).

The window at any time divides the possible sequence numbers into four regions.

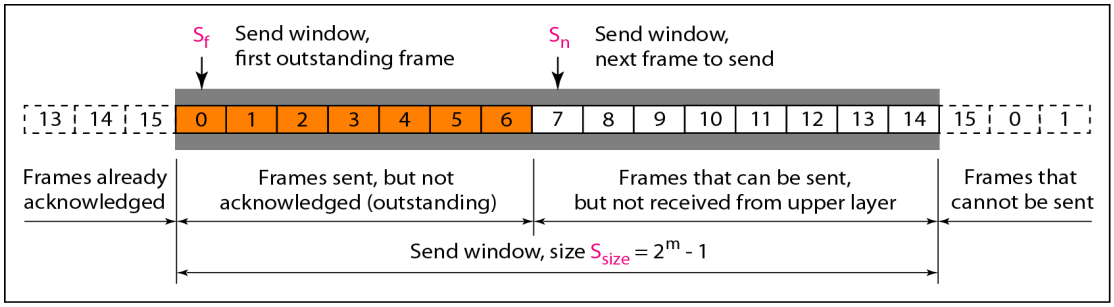
The first region, from the far left to the left wall of the window, defines the sequence numbers belonging to frames that are already acknowledged.

The sender does not worry about these frames and keeps no copies of them.

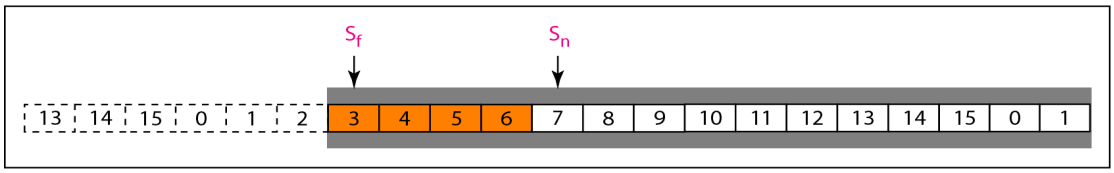
The second region, colored in Figure, defines the range of sequence numbers belonging to the frames that are sent and have an unknown status. The sender needs to wait to find out if these frames have been received or were lost. We call these outstanding frames.

The third range, white in the figure, defines the range of sequence numbers for frames that can be sent; however, the corresponding data packets have not yet been received from the network layer.

Finally, the fourth region defines sequence numbers that cannot be used until the window slides, as we see next.



a. Send window before sliding



b. Send window after sliding

The window itself is an abstraction; three variables define its size and location at any time. These variables are

Sf *(send* window, the first outstanding frame),
Sn (send window, the next frame to be sent), and
S_{size} (send window, size).

The variable ***Sf*** defines the sequence number of the first (oldest) outstanding frame. The variable ***Sn*** holds the sequence number that will be assigned to the next frame to be sent. Finally, the variable S_{size} defines the size of the window, which is fixed in our protocol.

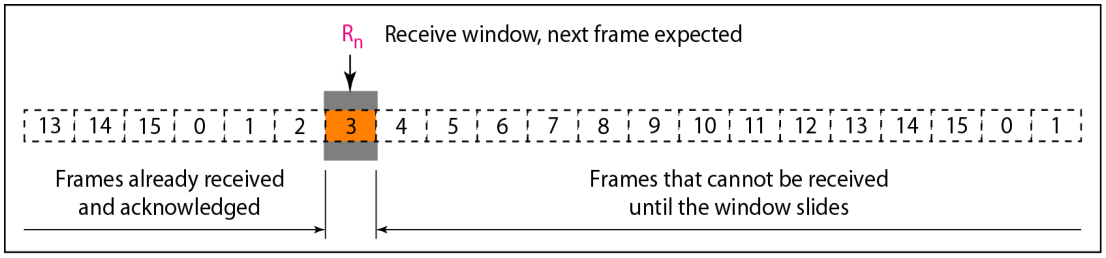
Figure (b) shows how a send window can slide one or more slots to the right when an acknowledgment arrives from the other end. As we will see shortly, the acknowledgments in this protocol are cumulative, meaning that more than one frame can be acknowledged by an ACK frame.

In Figure (b), frames 0, 1, and 2 are acknowledged so the window has slid to the right three slots. Note that the value of ***Sf*** is 3 because frame 3 is now the first outstanding frame.

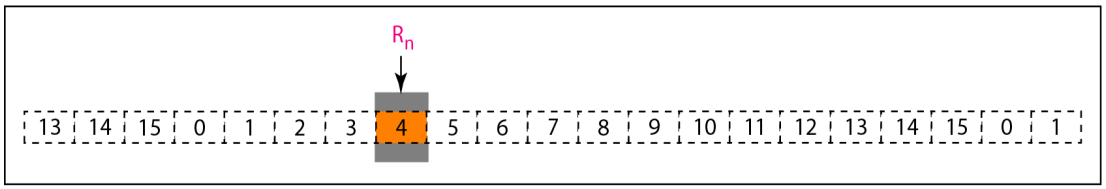
The send window can slide one or more slots when a valid acknowledgment arrives

The receive window makes sure that the correct data frames are received and that the correct acknowledgments are sent.

The size of the receive window is always 1(one). The receiver is always looking for the arrival of a specific frame. Any frame arriving out of order is discarded and needs to be resent. Figure shows the receive window.



a. Receive window



b. Window after sliding

We need only one variable ***Rn*** (receive window, next frame expected) to define this abstraction.

The sequence numbers to the left of the window belong to the frames already received and acknowledged; the sequence numbers to the right of this window define the frames that cannot be received.

Any received frame with a sequence number in these two regions is discarded. Only a frame with a sequence number matching the value of ***Rn*** is accepted and acknowledged.

4.1.1 A One-Bit Sliding Window Protocol

Before tackling the general case, let us examine a sliding window protocol with a window size of 1. Such a protocol uses stop-and-wait since the sender transmits a frame and waits for its acknowledgement before sending the next one.

- Under normal circumstances, one of the two data link layers goes first and transmits the first frame.
- The starting machine fetches the first packet from its network layer, builds a frame from it, and sends it. When this (or any) frame arrives, the receiving data link layer checks to see if it is a duplicate. If the frame is the one expected, it is passed to the network layer and the receiver's window is slid up.
- The acknowledgement field contains the number of the last frame received without error. If this number agrees with the sequence number of the frame the sender is trying to send, the sender knows it is done with the frame stored in *buffer* and can fetch the next packet from its network layer. If the sequence number disagrees, it must continue trying to send the same frame. Whenever a frame is received, a frame is also sent back.
- Now let us examine one-bit sliding protocol to see how resilient it is to pathological scenarios. Assume that computer *A* is trying to send its frame 0 to computer *B* and that *B* is trying to send its frame 0 to *A*. Suppose that *A* sends a frame to *B*, but *A*'s timeout interval is a little too short. Consequently, *A* may time out repeatedly, sending a series of identical frames, all with *seq* = 0 and *ack* = 1.
- When the first valid frame arrives at computer *B*, it will be accepted and *frame expected* will be set to a value of 1. All the subsequent frames received will be rejected because *B* is now expecting frames with sequence number 1, not 0. Furthermore, since all the duplicates will

have *ack* = 1 and *B* is still waiting for an acknowledgement of 0, *B* will not go and fetch a new packet from its network layer.

- After every rejected duplicate comes in, *B* will send *A* a frame containing *seq* = 0 and *ack* = 0. Eventually, one of these will arrive correctly at *A*, causing *A* to begin sending the next packet. No combination of lost frames or premature timeouts can cause the protocol to deliver duplicate packets to either network layer, to skip a packet, or to deadlock. The protocol is correct.
- However, to show how subtle protocol interactions can be, we note that a peculiar situation arises if both sides simultaneously send an initial packet. This synchronization difficulty is illustrated in below figure. In part (a), the normal operation of the protocol is shown. In (b) the peculiarity is illustrated. If *B* waits for *A*'s first frame before sending one of its own, the sequence is as shown in (a), and every frame is accepted.
- However, if *A* and *B* simultaneously initiate communication, their first frames cross, and the data link layers then get into situation (b). In (a) each frame arrival brings a new packet for the network layer; there are no duplicates. In (b) half of the frames contain duplicates, even though there are no transmission errors. Similar situations can occur as a result of premature timeouts, even when one side clearly starts first. In fact, if multiple premature timeouts occur, frames may be sent three or more times, wasting valuable bandwidth.

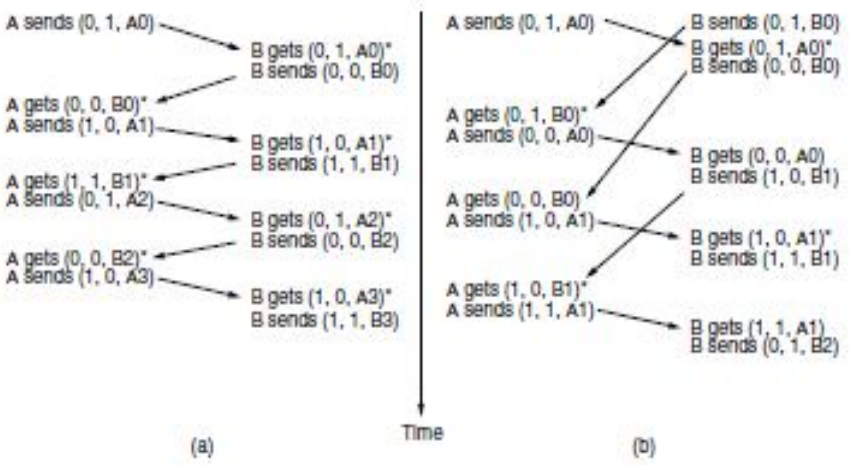


Fig: Two scenarios for above protocol (a) Normal case (b) Abnormal case
The notation is (seq, ack, packet number). An asterisk indicates where a network layer accepts a packet

Go-Back-N Automatic Repeat Request

To improve the efficiency of transmission (filling the pipe), multiple frames must be in transition while waiting for acknowledgment. In other words, we need to let more than one frame be outstanding to keep the channel busy while the sender is waiting for acknowledgment.

The first is called **Go-Back-N Automatic Repeat Request**. In this protocol we can send several frames before receiving acknowledgments. We keep a copy of these frames until the acknowledgments arrive.

Sequence Numbers

Frames from a sending station are numbered sequentially. However, because we need to include the sequence number of each frame in the header, we need to set a limit.
If the header of the frame allows *m* bits for the sequence number, the sequence numbers range from **0 to *2m* - 1**.
For example, if *m* is 4, the only sequence numbers are 0 through 15 inclusive. However, we can repeat the sequence. So the sequence numbers are

0, 1,2,3,4,5,6, 7,8,9, 10, 11, 12, 13, 14, 15,0, 1,2,3,4,5,6,7,8,9,10, 11, ...

Timers

Although there can be a timer for each frame that is sent, in our protocol we use only one. The reason is that the timer for the first outstanding frame always expires first; we send all outstanding frames when this timer expires.

Acknowledgment

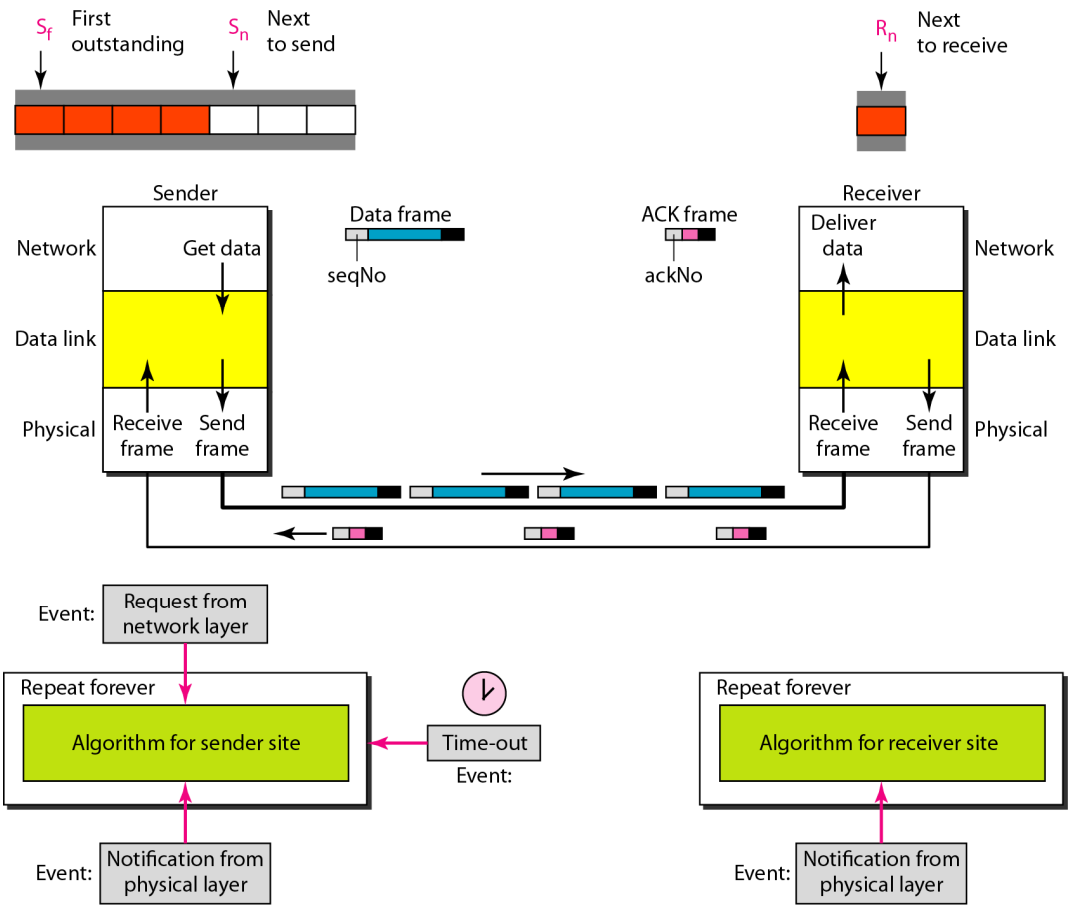
The receiver sends a positive acknowledgment if a frame has arrived safe and sound and in order. If a frame is damaged or is received out of order, the receiver is silent and will discard all subsequent frames until it receives the one it is expecting. The silence of the receiver causes the timer of the unacknowledged frame at the sender site to expire. This, in turn, causes the sender to go back and resend all frames, beginning with the one with the expired timer. The receiver does not have to acknowledge each frame received. It can send one cumulative acknowledgment for several frames.

Resending a Frame

When the timer expires, the sender resends all outstanding frames. For example, suppose the sender has already sent frame 6, but the timer for frame 3 expires. This means that frame 3 has not been acknowledged; the sender goes back and sends frames 3, 4, 5, and 6 again. That is why the protocol is called ***Go-Back-N* ARQ**.

Design

Figure shows the design for this protocol. As we can see, multiple frames can be in transit in the forward direction, and multiple acknowledgments in the reverse direction. The idea is similar to Stop-and-Wait ARQ; the difference is that the send window allows us to have as many frames in transition as there are slots in the send window.



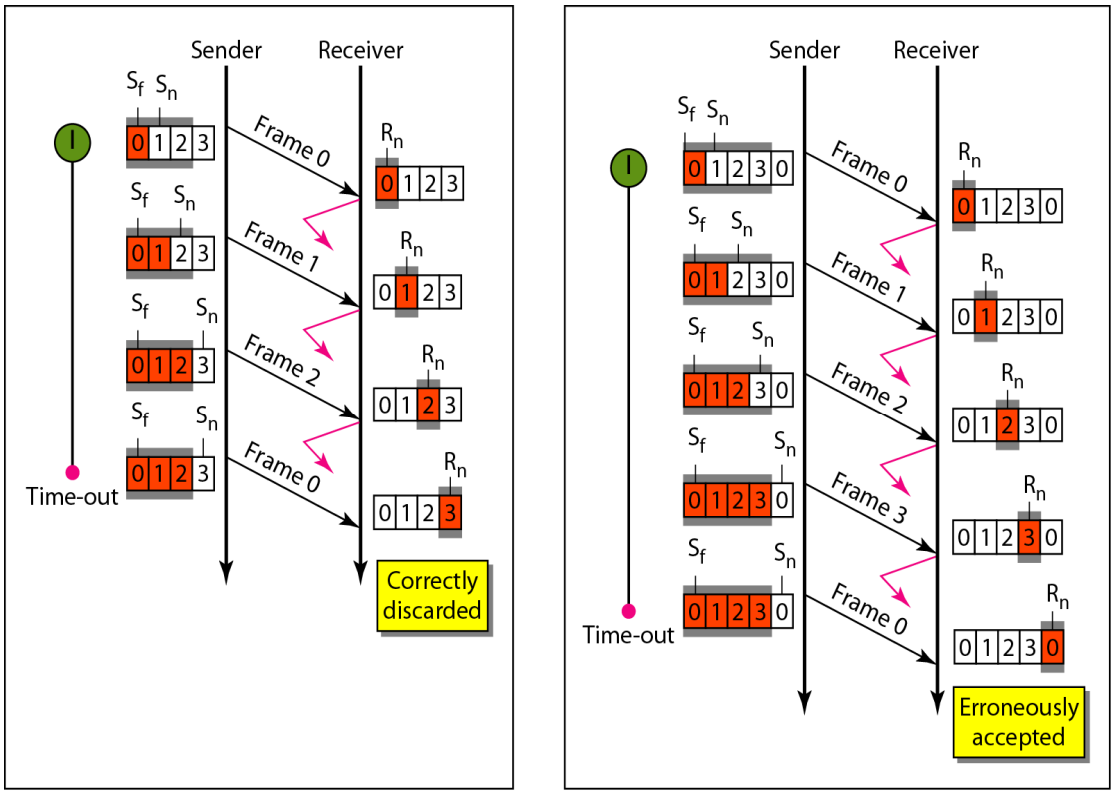
Send Window Size:

We can now show why the size of the send window must be less than 2^m.

As an example, we choose m =2, which means the size of the window can be 2^m - 1, or 3. Below mentioned Figure compares a window size of 3 against a window size of 4.

If the size of the window is 3 (less than 2^2) and all three acknowledgments are lost, the frame timer expires and all three frames are resent. The receiver is now expecting frame 3, not frame 0, so the duplicate frame is correctly discarded.

On the other hand, if the size of the window is 4 (equal to 2^2) and all acknowledgments are lost, the sender will send a duplicate of frame 0. However, this time the window of the receiver expects to receive frame 0, so it accepts frame 0, not as a duplicate, but as the first frame in the next cycle. This is an error.



a. Window size < 2^m

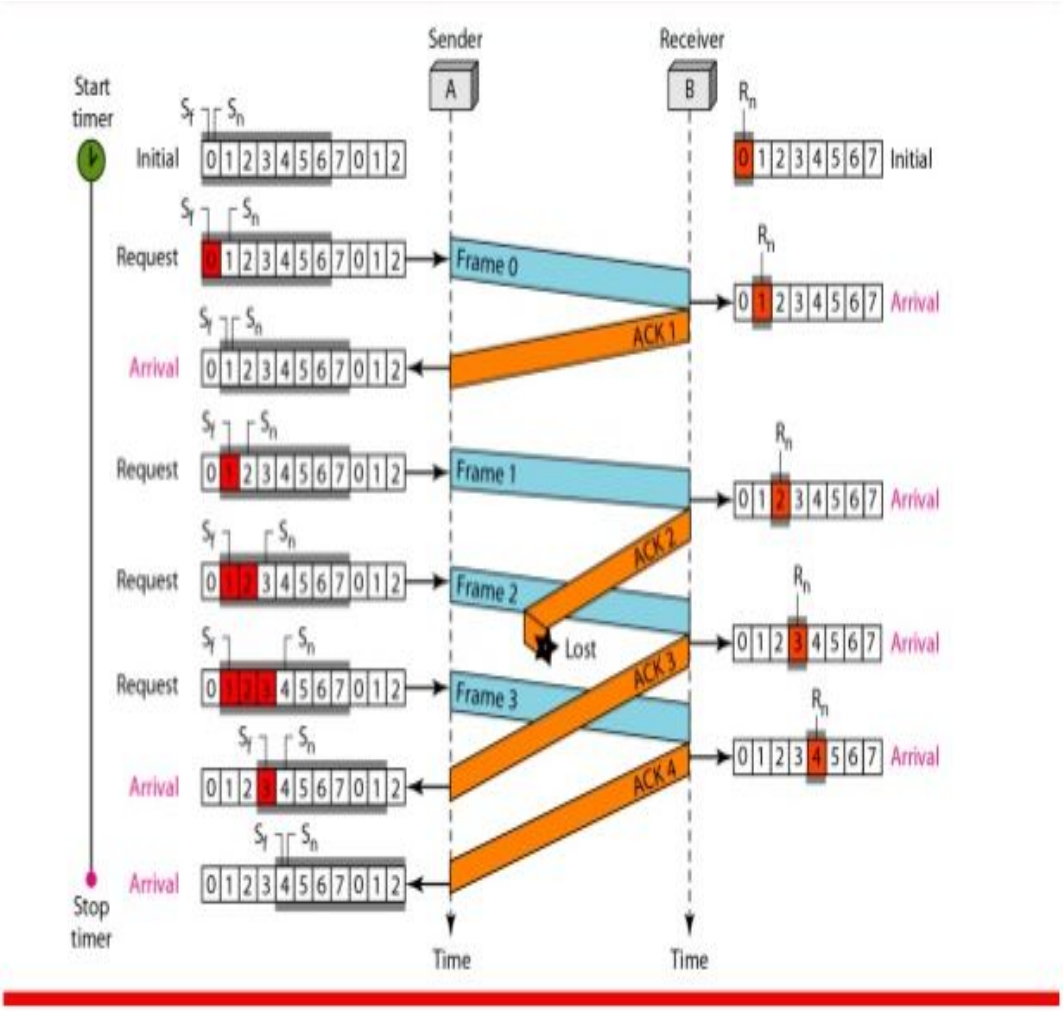
b. Window size = 2^m

Flow Diagram:

Figure shows an example of Go-Back-*N*. This is an example of a case where the forward channel is reliable, but the reverse is not. No data frames are lost, but some ACKs are delayed and one is lost. The example also shows how cumulative acknowledgments can help if acknowledgments are delayed or lost.

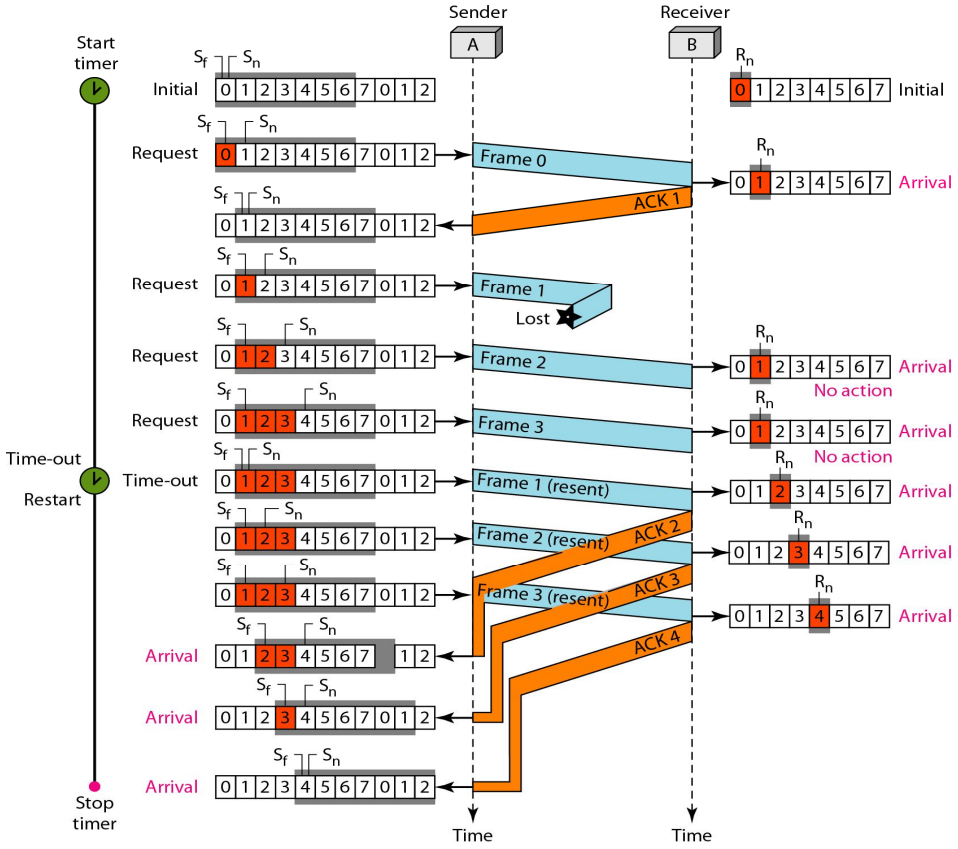
After initialization, there are seven sender events. Request events are triggered by data from the network layer; arrival events are triggered by acknowledgments from the physical layer. There is no time-out event here because all outstanding frames are acknowledged before the timer expires. Note that although ACK 2 is lost, ACK 3 serves as both ACK 2 and ACK3.

There are four receiver events, all triggered by the arrival of frames from the physical layer.



Below Figure shows what happens when a frame is lost. Frames 0, 1, 2, and 3 are sent. However, frame 1 is lost. The receiver receives frames 2 and 3, but they are discarded because they are received out of order (frame 1 is expected). The sender receives no acknowledgment about frames 1, 2, or 3. Its timer finally expires. The sender sends all outstanding frames (1, 2, and 3) because it does not know what is wrong. Note that the resending of frames l, 2, and 3 is the response to one single event.

When the sender is responding to this event, it cannot accept the triggering of other events. This means that when ACK 2 arrives, the sender is still busy with sending frame 3. The physica1layer must wait until this event is completed and the data link layer goes back to its sleeping state. We have shown a vertical line to indicate the delay. It is the same story with ACK 3; but when ACK 3 arrives, the sender is busy responding to ACK 2. It happens again when ACK 4 arrives. Note that before the second timer expires, all outstanding frames have been sent and the timer is stopped.



In Go-Back-N ARQ, the size of the send window must be less than 2m;
The size of the receiver window is always 1.

Selective Repeat ARQ

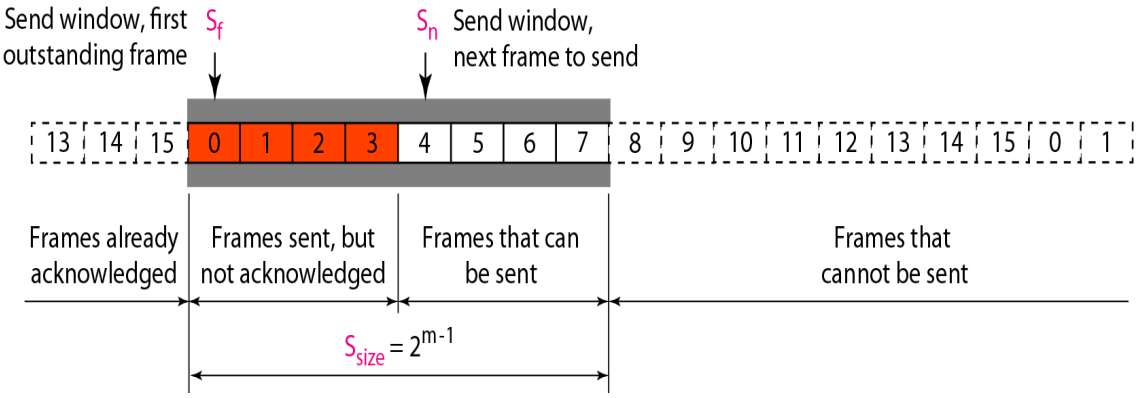
Go-Back-N ARQ simplifies the process at the receiver site. The receiver keeps track of only one variable, and there is no need to buffer out-of-order frames; they are simply discarded.
However, this protocol is very inefficient for a noisy link. In a noisy link a frame has a higher probability of damage, which means the resending of multiple frames. This resending uses up the bandwidth and slows down the transmission.
For noisy links, there is another mechanism that does not resend *N* frames when just one frame is damaged; only the damaged frame is resent. This mechanism is called **Selective Repeat ARQ**. It is more efficient for noisy links, but the processing at the receiver is more complex.

Windows

The Selective Repeat Protocol also uses two windows: a send window and a receive window. However, there are differences between the windows in this protocol and the ones in Go Back-N.

First, the size of the send window is much smaller; it is 2^{m-1}. Second, the receive window is the same size as the send window.

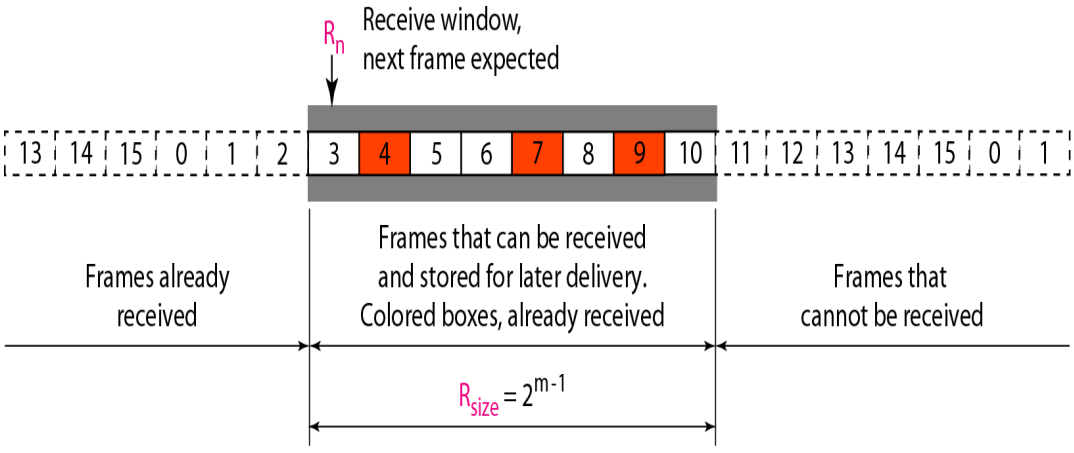
The send window maximum size can be 2^{m-1}. For example, if $m = 4$, the sequence numbers go from 0 to 15, but the size of the window is just 8 (it is 15 in the *Go-Back-N* Protocol). The smaller window size means less efficiency in filling the pipe, but the fact that there are fewer duplicate frames can compensate for this. The protocol uses the same variables as we discussed for Go-Back-N. We show the Selective Repeat send window in Figure to emphasize the size.



The receive window in Selective Repeat is totally different from the one in GoBack- N.

First, the size of the receive window is the same as the size of the send window (2^{m-1}). The Selective Repeat Protocol allows as many frames as the size of the receive window to arrive out of order and be kept until there is a set of in-order frames to be delivered to the network layer.

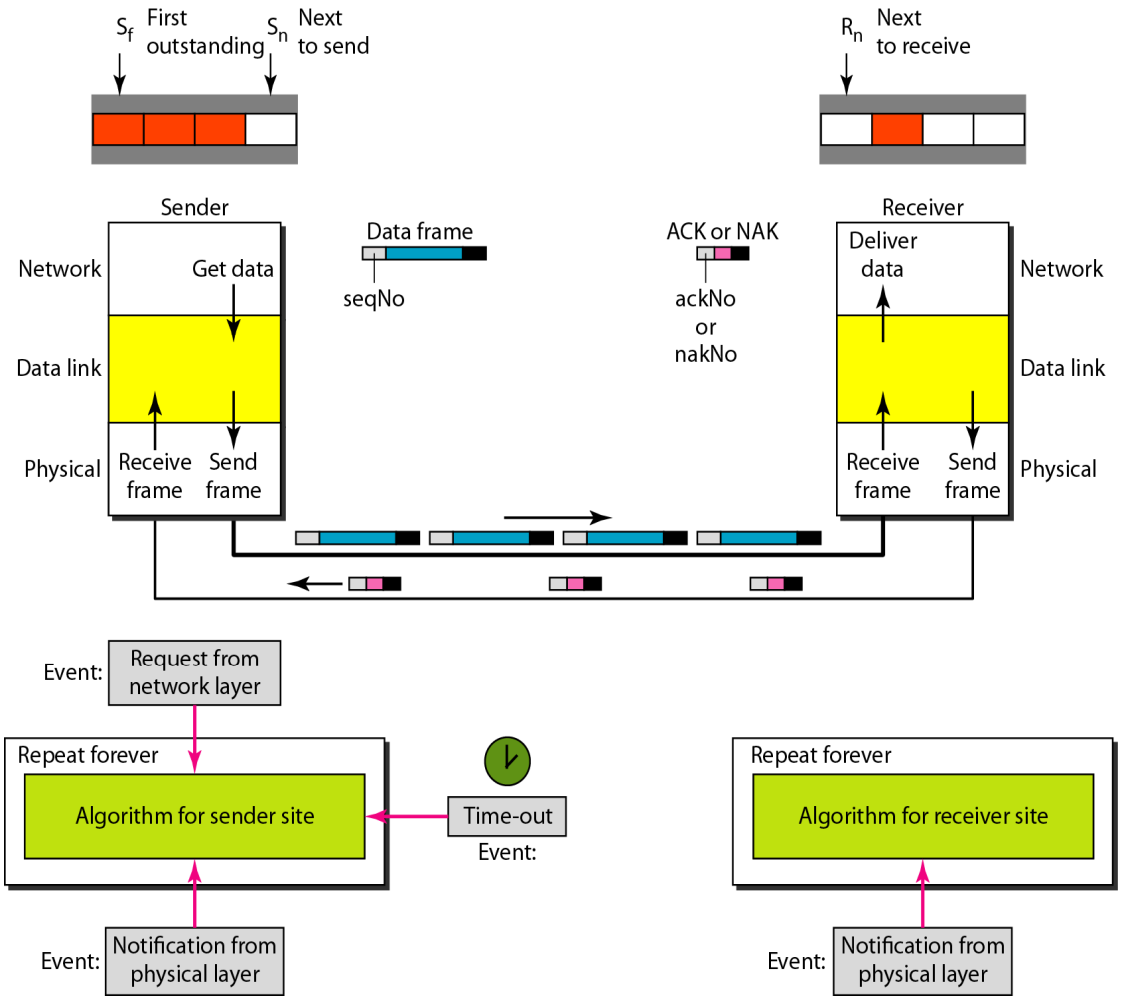
Because the sizes of the send window and receive window are the same, all the frames in the send frame can arrive out of order and be stored until they can be delivered. Figure shows the receive window.



Those slots inside the window that are colored define frames that have arrived out of order and are waiting for their neighbors to arrive before delivery to the network layer.

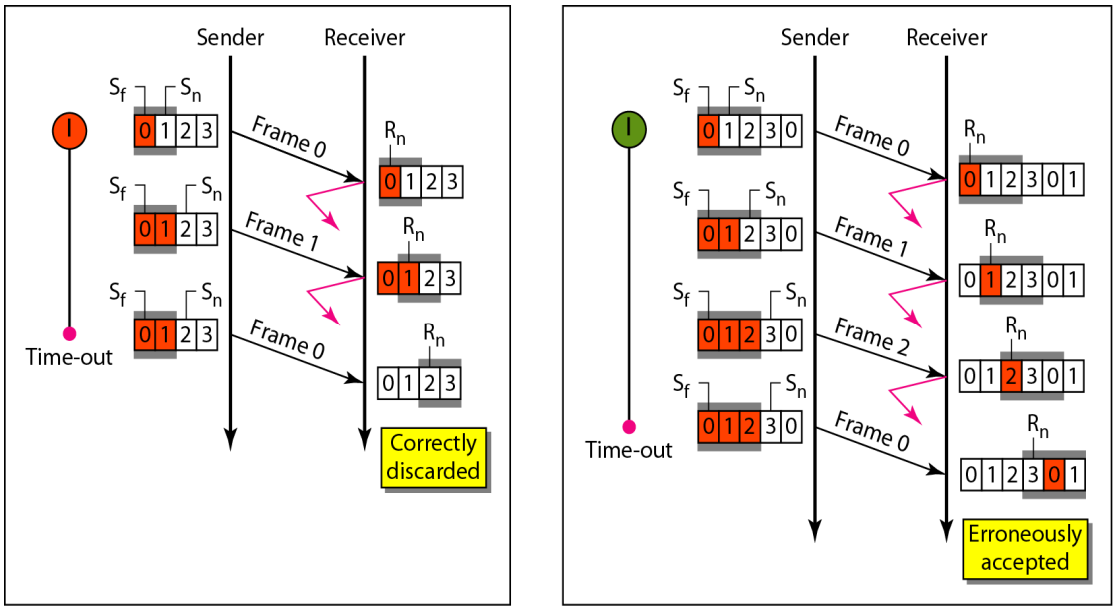
Design

The design in this case is to some extent similar to the one we described for the Go-Back-N, but more complicated, as shown in Figure.



Window Sizes

We can now show why the size of the sender and receiver windows must be at most one-half of 2^m. For an example, we choose $m = 2$, which means the size of the window is $2^m/2$, or 2. Figure shows compare a window size of 2 with a window size of 3.



a. Window size = 2^{m-1} b. Window size > 2^{m-1}

If the size of the window is 2 and all acknowledgments are lost, the timer for frame 0 expires and frame 0 is resent. However, the window of the receiver is now expecting frame 2, not frame 0, so this duplicate frame is correctly discarded.

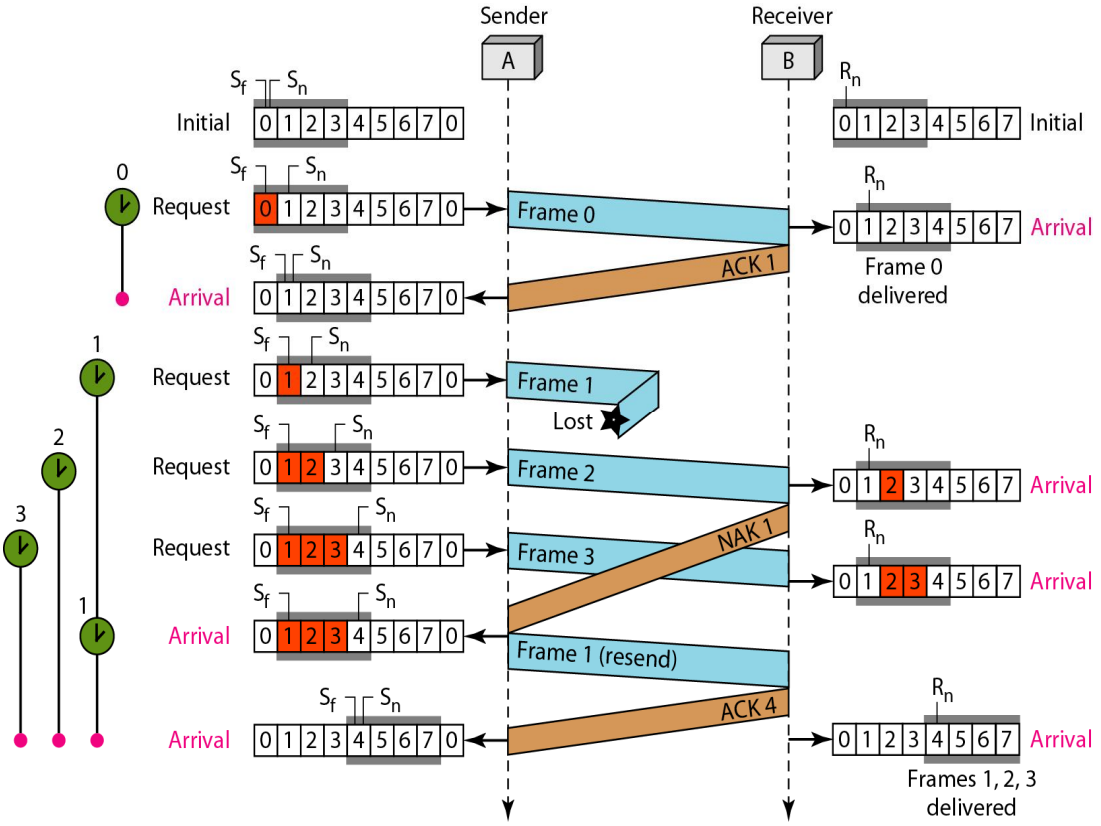
When the size of the window is 3 and all acknowledgments are lost, the sender sends a duplicate of frame 0.

However, this time, the window of the receiver expects to receive frame 0 (0 is part of the window), so it accepts frame 0, not as a duplicate, but as the first frame in the next cycle. This is clearly an error.

In Selective Repeat ARQ, the size of the sender and receiver window must be at most one-half of 2m.

Flow Diagram:

One main difference is the number of timers. Here, each frame sent or resent needs a timer, which means that the timers need to be numbered (0, 1,2, and 3). The timer for frame 0 starts at the first request, but stops when the ACK for this frame arrives. The timer for frame 1 starts at the second request, restarts when a NAK arrives, and finally stops when the last ACK arrives. The other two timers start when the corresponding frames are sent and stop at the last arrival event.



Piggybacking

The three protocols we discussed in this section are all unidirectional: data frames flow in only one direction although control information such as ACK and NAK frames can travel in the other direction. In real life, data frames are normally flowing in both directions: from node A to node B and from node B to node A. This means that the control information also needs to flow in both directions.

A technique called **piggybacking** is used to improve the efficiency of the bidirectional protocols. When a frame is carrying data from A to B, it can also carry control information about arrived (or lost) frames from B; when a frame is carrying data from B to A, it can also carry control information about the arrived (or lost) frames from A.

HDLC

High-level Data Link Control (HDLC) is a bit-oriented protocol for communication over point-to-point and multipoint links.

Configurations and Transfer Modes

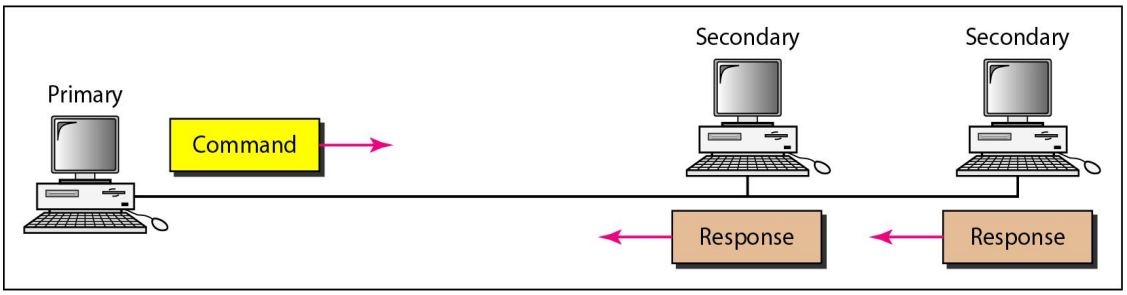
HDLC provides two common transfer modes that can be used in different configurations: normal response mode (NRM) and asynchronous balanced mode (ABM).

Normal Response Mode

In normal response mode (NRM), the station configuration is unbalanced. We have one primary station and multiple secondary stations. A primary station can send commands; a secondary station

can only respond. The NRM is used for both point-to-point and multiple-point links, as shown in figure.

a. Point-to-point



b. Multipoint

Asynchronous Balanced Mode

In asynchronous balanced mode (ABM), the configuration is balanced. The link is point-to-point, and each station can function as a primary and a secondary (acting as peers), as shown in figure. This is the common mode today.

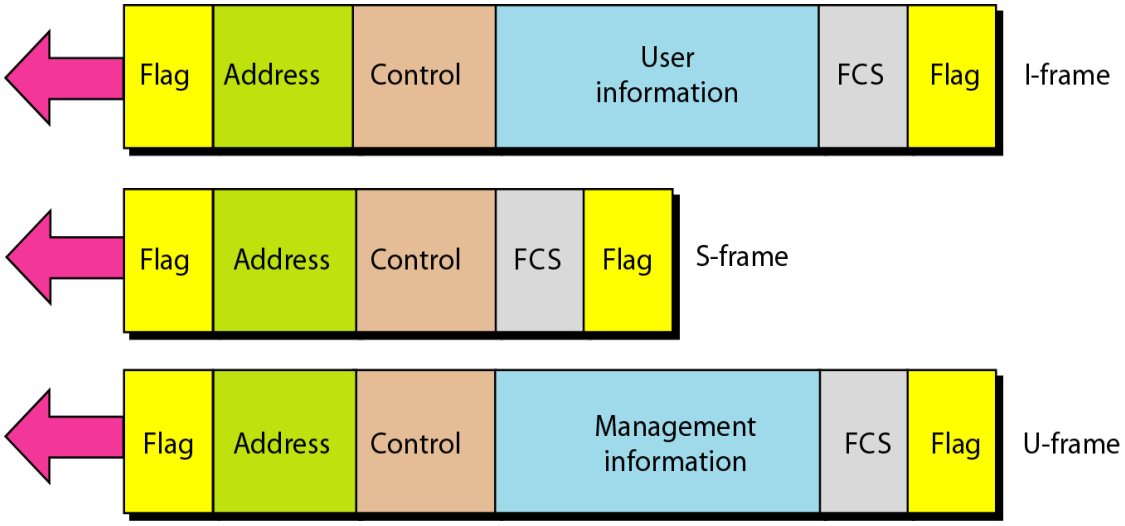
Frames

To provide the flexibility necessary to support all the options possible in the modes and configurations just described, HDLC defines three types of frames: information frames (I-frames), supervisory frames (S-frames), and unnumbered frames (U-frames).

- Each type of frame serves as an envelope for the transmission of a different type of message. I-frames are used to transport user data and control information relating to user data (piggybacking). S-frames are used only to transport control information. U-frames are reserved for system management. Information carried by U-frames is intended for managing the link itself.

Frame Format

Each frame in HDLC may contain up to six fields, as shown in figure : a beginning flag field, an address field, a control field, an information field, a frame check sequence (FCS) field, and an ending flag field. In multiple-frame transmissions, the ending flag of one frame can serve as the beginning flag of the next frame.



Fields

Let us now discuss the fields and their use in different frame types.

- **Flag field:** The flag field of an HDLC frame is an 8-bit sequence with the bit pattern 01111110 that identifies both the beginning and the end of a frame and serves as a synchronization pattern for the receiver.
- **Address field:** The second field of an HDLC frame contains the address of the secondary station. If a primary station created the frame, it contains a ***to*** address. If a secondary creates the frame, it contains a ***from*** address. An address field can be 1 byte or several bytes long, depending on the needs of the network. One byte can identify up to 128 stations (l bit is used for another purpose). Larger networks require multiple-byte address fields. If the address field is only 1 byte, the last bit is always a 1. If the address is more than 1 byte, all bytes but the last one will end with 0; only the last will end with 1. Ending each intermediate byte with 0 indicates to the receiver that there are more address bytes to come.
- **Control field:** The control field is a 1- or 2-byte segment of the frame used for flow and error control. The interpretation of bits in this field depends on the frame type.
- **Information field:** The information field contains the user's data from the network layer or management information. Its length can vary from one network to another.
- **FCS field:** The frame check sequence (FCS) is the HDLC error detection field. It can contain either a 2- or 4-byte ITU-T CRC.

Control Field

The control field determines the type of frame and defines its functionality. The format is specific for the type of frame, as shown in figure.

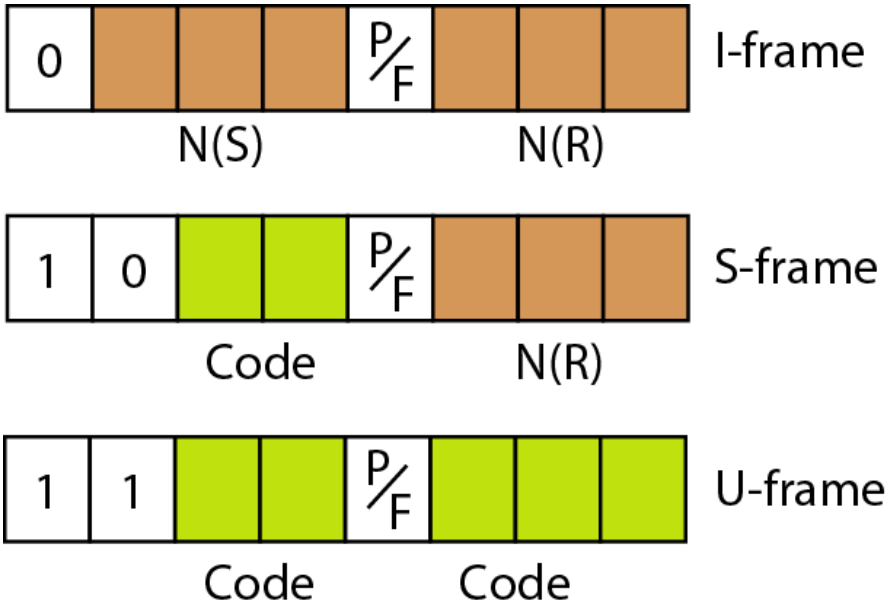


Fig: Control field format for the different frame types

Control Field for I-Frames

I-frames are designed to carry user data from the network layer. In addition, they can include flow and error control information (piggybacking).

- The subfields in the control field are used to define these functions. The first bit defines the type. If the first bit of the control field is 0, this means the frame is an I-frame. The next 3 bits, called *N(S)*, define the sequence number of the frame. Note that with 3 bits, we can define a sequence number between 0 and 7; but in the extension format, in which the control field is 2 bytes, this field is larger. The last 3 bits, called *N(R)*, correspond to the acknowledgment number when piggybacking is used.
- The single bit between *N(S)* and *N(R)* is called the *P/F* bit. The *P/F* field is a single bit with a dual purpose. It has meaning only when it is set (bit = 1) and can mean poll or final. It means *poll* when the frame is sent by a primary station to a secondary (when the address field contains the address of the receiver). It means *final* when the frame is sent by a secondary to a primary (when the address field contains the address of the sender).

Control Field for S-Frames

Supervisory frames are used for flow and error control whenever piggybacking is either impossible or inappropriate (e.g., when the station either has no data of its own to send or needs to send a command or response other than an acknowledgment). S-frames do not have information fields. If the first 2 bits of the control field is 10, this means the frame is an S-frame. The last 3 bits, called *N(R)*, corresponds to the acknowledgment number (ACK) or negative acknowledgment number (NAK) depending on the type of S-frame. The 2 bits called code is used to define the type of S-frame itself. With 2 bits, we can have four types of S-frames, as described below:

- **Receive ready (RR):** If the value of the code subfield is 00, it is an RR S-frame. This kind of frame acknowledges the receipt of a safe and sound frame or group of frames. In this case, the value *N(R)* field defines the acknowledgment number.
- **Receive not ready (RNR):** If the value of the code subfield is 10, it is an RNR S-frame. This kind of frame is an RR frame with additional functions. It acknowledges the receipt of a frame or group of frames, and it announces that the receiver is busy and cannot receive more frames. It acts as a kind of congestion control mechanism by asking the sender to slow down. The value of *NCR)* is the acknowledgment number.

- **Reject (REJ):** If the value of the code subfield is 01, it is a REJ S-frame. This is a NAK frame, but not like the one used for Selective Repeat ARQ. It is a NAK that can be used in *Go-Back-N* ARQ to improve the efficiency of the process by informing the sender, before the sender time expires, that the last frame is lost or damaged. The value of *N(R)* is the negative acknowledgment number.
- **Selective reject (SREJ):** If the value of the code subfield is 11, it is an SREJ S-frame. This is a NAK frame used in Selective Repeat ARQ. Note that the HDLC Protocol uses the term *selective reject* instead of *selective repeat.* The value of *N(R)* is the negative acknowledgment number.

Control Field for U-Frames

Unnumbered frames are used to exchange session management and control information between connected devices. Unlike S-frames, U-frames contain an information field, but one used for system management information, not user data.

- As with S-frames, however, much of the information carried by U-frames is contained in codes included in the control field. U-frame codes are divided into two sections: a 2-bit prefix before the P/F bit and a 3-bit suffix after the P/F bit. Together, these two segments (5 bits) can be used to create up to 32 different types of U-frames. Some of the more common types are shown in table

Code	*Command*	*Response*	*Meaning*
00 001	SNRM		Set normal response mode
11 011	SNRME		Set normal response mode, extended
11 100	SABM	**DM**	Set asynchronous balanced mode or **disconnect mode**
11 110	SABME		Set asynchronous balanced mode, extended
00 000	UI	**UI**	Unnumbered information
00 110		**UA**	**Unnumbered acknowledgment**
00 010	DISC	**RD**	Disconnect or **request disconnect**
10 000	SIM	**RIM**	Set initialization mode or **request information mode**
00 100	UP		Unnumbered poll
11 001	RSET		Reset
11 101	XID	**XID**	Exchange ID
10 001	FRMR	**FRMR**	Frame reject

POINT-TO-POINT PROTOCOL

Although HDLC is a general protocol that can be used for both point-to-point and multipoint configurations, one of the most common protocols for point-to-point access is the Point-to-Point Protocol (PPP).

- Today, millions of Internet users who need to connect their home computers to the server of an Internet service provider use PPP.
- The majority of these users have a traditional modem; they are connected to the Internet through a telephone line, which provides the services of the physical layer. But to control and manage the transfer of data, there is a need for a point-to-point protocol at the data link layer.

PPP provides several services:
1. PPP defines the format of the frame to be exchanged between devices.
2. PPP defines how two devices can negotiate the establishment of the link and the exchange of data.
3. PPP defines how network layer data are encapsulated in the data link frame.
4. PPP defines how two devices can authenticate each other.
5. PPP provides multiple network layer services supporting a variety of network layer protocols.
6. PPP provides connections over multiple links.
7. PPP provides network address configuration. This is particularly useful when a home user needs a temporary network address to connect to the Internet.

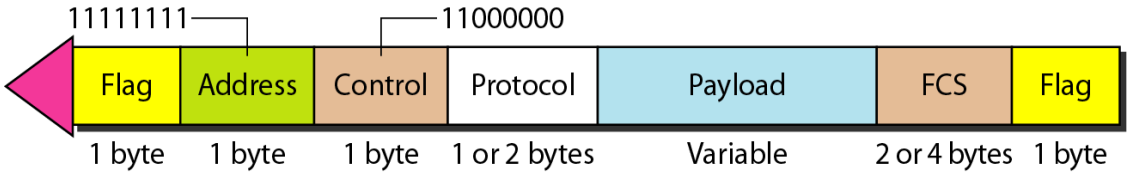
On the other hand, to keep PPP simple, several services are missing:
I. PPP does not provide flow control. A sender can send several frames one after another with no concern about overwhelming the receiver.
2. PPP has a very simple mechanism for error control. A CRC field is used to detect errors. If the frame is corrupted, it is silently discarded; the upper-layer protocol needs to take care of the problem. Lack of error control and sequence numbering may cause a packet to be received out of order.
3. PPP does not provide a sophisticated addressing mechanism to handle frames in a multipoint configuration.

Framing

PPP is a byte-oriented protocol. Framing is done according to the discussion of byte oriented protocols at the beginning of this chapter.

Frame Format

Figure shows the format of a PPP frame. The description of each field follows:



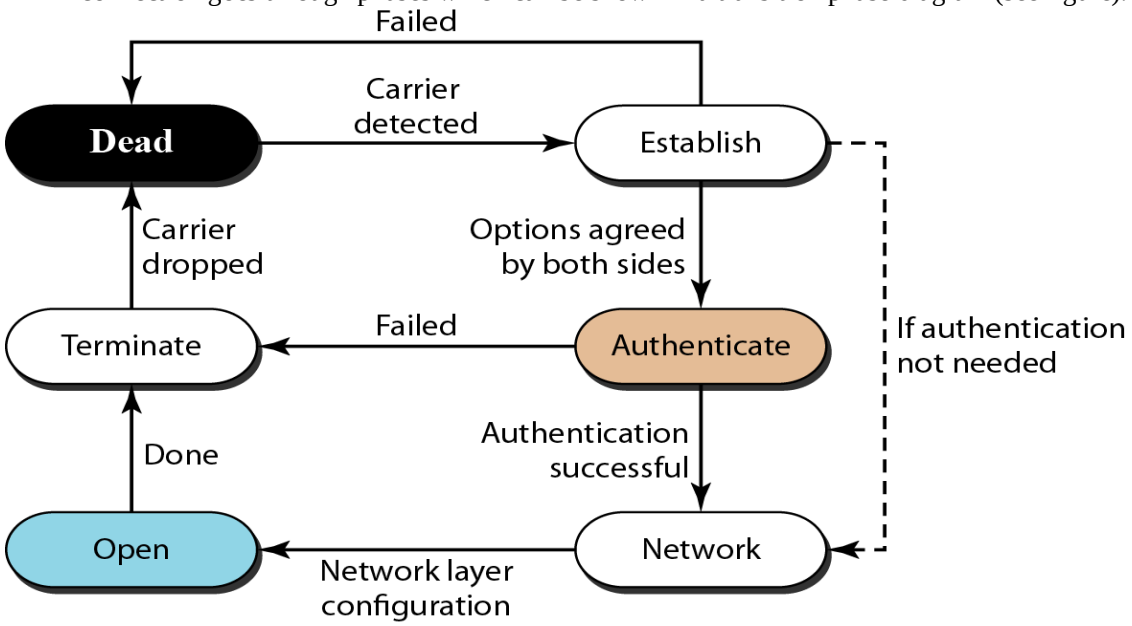
- **Flag**: A PPP frame starts and ends with a 1-byte flag with the bit pattern 01111110. Although this pattern is the same as that used in HDLC, there is a big difference. PPP is a byte-oriented protocol; HDLC is a bit-oriented protocol.
- **Address:** The address field in this protocol is a constant value and set to 11111111 (broadcast address).
- **Control:** This field is set to the constant value 11000000 (imitating unnumbered frames in HDLC). PPP does not provide any flow control. Error control is also limited to error detection. This means that this field is not needed at all, and again, the two parties can agree, during negotiation, to omit this byte.
- **Protocol:** The protocol field defines what is being carried in the data field: either user data or other information. This field is by default 2 bytes long, but the two parties can agree to use only 1 byte.
- **Payload field:** This field carries either the user data or other information. The data field is a sequence of bytes with the default of a maximum of 1500 bytes; but this can be changed during negotiation. The data field is bytestuffed if the flag byte pattern appears in this field. Because there is no field defining the size of the data field, padding is needed if the size is less than the maximum default value or the maximum negotiated value.
- **FCS.**: The frame check sequence (FCS) is simply a 2-byte or 4-byte standard CRC.

Byte Stuffing

The similarity between PPP and HDLC ends at the frame format. PPP, as we discussed before, is a byte-oriented protocol totally different from HDLC. As a byte-oriented protocol, the flag in PPP is a byte and needs to be escaped whenever it appears in the data section of the frame. The escape byte is 01111101, which means that every time the flag like pattern appears in the data, this extra byte is stuffed to tell the receiver that the next byte is not a flag.

Transition Phases

A PPP connection goes through phases which can be shown in a transition phase diagram (see figure).



- **Dead:** In the dead phase the link is not being used. There is no active carrier (at the physical layer) and the line is quiet.
- **Establish:** When one of the nodes starts the communication, the connection goes into this phase. In this phase, options are negotiated between the two parties. If the negotiation is successful, the system goes to the authentication phase (if authentication is required) or directly to the networking phase.
- **Authenticate:** The authentication phase is optional; the two nodes may decide, during the establishment phase, not to skip this phase. However, if they decide to proceed with authentication, they send several authentication packets. If the result is successful, the connection goes to the networking phase; otherwise, it goes to the termination phase.
- **Network:** In the network phase, negotiation for the network layer protocols takes place. PPP specifies that two nodes establish a network layer agreement before data at the network layer can be exchanged. The reason is that PPP supports multiple protocols at the network layer. If a node is running multiple protocols simultaneously at the network layer, the receiving node needs to know which protocol will receive the data.
- **Open:** In the open phase, data transfer takes place. When a connection reaches this phase, the exchange of data packets can be started. The connection remains in this phase until one of the endpoints wants to terminate the connection.
- **Terminate:** In the termination phase the connection is terminated. Several packets are exchanged between the two ends for house cleaning and closing the link.

UNIT-IV

Assignment-Cum-Tutorial Questions

SECTION-A

I. Objective Questions

1. Which feature of Go-Back-N ARQ mechanism possesses an ability to assign the sliding window in the forward direction? [ ]
a. Control Variables b. Sender Sliding Window
c. Receiver Sliding Window d. Resending of frames

2. Which ARQ mechanism deals with the transmission of only damaged or lost frames despite the other multiple frames by increasing the efficiency & its utility in noisy channels? [ ]
a. Go-Back-N ARQ b. Selective Repeat ARQ
c. Stop-and-Wait ARQ d. All of the above

3. What are the frames issued by the secondary station of HDLC ,known as?
a. Link b. Command c. Response d. both a and b [ ]

4. Which operational mode/s of HDLC support/s the balanced configuration by governing point-to-point link connectivity in addition to the primary as well as secondary functions performed by the station? [ ]
a. NRM b. ABM c. Both a & b d. neither a or b

5. Which category of HDLC frames undergoes error and flow control mechanisms by comprising send and receive sequence numbers? [ ]
a. U-frames b. I-frames c. S-frames d. All of the above

6. HDLC is an acronym for _____

7. In Go-Back-N ARQ, if 5 is the number of bits for the sequence number, then the maximum size of the send window must be ____

8. High-level Data Link Control (HDLC) is a _____ protocol for communication over point-to-point and multipoint links.

9. In Selective Repeat ARQ, the size of the sender and receiver window must be at most _____ of 2^m

10. _____line discipline where only some of the enquiries are acknowledged is more efficient then Stop-and-Wait

11. The general strategy for handling errors when frames are pipelined is called _____

12. Range of possible sequence numbers of Go-Back-N Protocol which is concern of receiver is called the [ ]

a. sender sliding window b. pipelining at sender
c. pipelining at receiver d. receiver sliding window

13. The maximum window size for data transmission using the selective reject protocol with n-bit frame sequence numbers is [ ]

a) 2^n b) 2^{n-1} c) 2^n-1 d)2^{n-2}

14. Frame type that refers to High-level Data Link Control error detection field is [ ]

a. Frame Check Sequence field b)Control field

b. flag field d) information field

15. In control field, first bit defines the [ ]

a. sequence number b) acknowledgement number
c. type d) information

16. In Normal Response Mode (NRM), station configuration is [ ]

a. balanced b) unbalanced

c. zero d) undefined

17. In transition phase, term that refers to start of communication is

a. authenticate b) establish [ ]

c. network d) terminate

18. Internet users use the [ ]

a. High-level Data Link Control

b. Password Authentication Protocol

c. Multipoint Protocol

d. Point to Point Protocol

19. The most common protocol for point-to-point access is the Point-to-Point Protocol (PPP), which is a _____protocol. [ ]

a. bit-oriented

b. byte-oriented

c. character-oriented

d. string-oriented

20. Both Go-Back-*N* and Selective-Repeat Protocols use a _____ [ ]

a. sliding frame

b. sliding window

c. sliding packet

d. none of the above

21. In the _____Protocol, if no acknowledgment for a frame has arrived, we resend all outstanding frames [ ]

a. Stop-and-Wait ARQ

b. Go-Back-N ARQ

c. Selective-Repeat ARQ

d. all of the above

22. The _____Protocol has both flow control and error control. [ ]

a. Stop-and-Wait

b. Go-Back-N ARQ

c. Selective-Repeat ARQ

d. both (b) and (c)

23. High-level Data Link Control (HDLC) is a _____ protocol for communication over point-to-point and multipoint links. [ ]

a. bit-oriented

b. byte-oriented

c. character-oriented

d. string-oriented

SECTION-B

SUBJECTIVE QUESTIONS

1. Define piggybacking.
2. Give the mechanism of Go-back N ARQ technique.
3. Briefly explain one bit sliding window protocol
4. Explain the working of stop- and- wait sliding window protocol
5. Discuss about the configuration and control fields of HDLC.
6. There are three types of data transfer modes defined by HDLC. What are they? Explain them.
7. Describe the services provided by PPP protocol. Also, list some services which does PPP does not provide.
8. Explain the frame format and transition phases of PPP.
9. A channel has a bit rate of 4 kbps and a propagation delay of 20 msec. For what range of frame sizes does stop-and-wait give an efficiency of at least 50 percent?
10. What different types of protocols are used in PPP to do Multiplexing? Describe them in detail.

SECTION-C

QUESTIONS AT THE LEVEL OF GATE

1. Host A is sending data to host B over a full duplex link. A and B are using the sliding window protocol for flow control. The send and receive window sizes are 5 packets each. Data packets (sent only from A to B) are all 1000 bytes long and the transmission time for such a packet is 50 µs. Acknowledgement packets (sent only from B to A) are very small and require negligible transmission time. The propagation delay over the link is 200 us. What is the maximum achievable throughput in this communication? **[GATE 2003]**

2. Station A uses 32 byte packets to transmit messages to Station B using a sliding window protocol. The round trip delay between A and B is 80 milliseconds and the bottleneck bandwidth on the path between A and B is 128 kbps. What is the optimal window size that A should use? [ ]

a. 20 b. 40 C.160 d. 320 **[GATE 2006]**

UNIT-V

Multiple Access

Objectives:
Familiarize the student with the basic taxonomy and terminology of multiple access protocols in computer networks

Syllabus:
Random Access: ALOHA, career sense multiple access (CSMA), career sense multiple access with collision detection, career sense multiple access with collision avoidance, **Controlled Access** - Reservation, Polling, Token Passing.

Outcomes:
Students will be able to

- specify and identify deficiencies in existing protocols, and then go onto formulate new and better protocols
- Understand fundamental underlying principles of MAC layer
- Identify the different types of random access and controlled access methods
- Comparision of different types of random access and controlled access methods

Learning Material

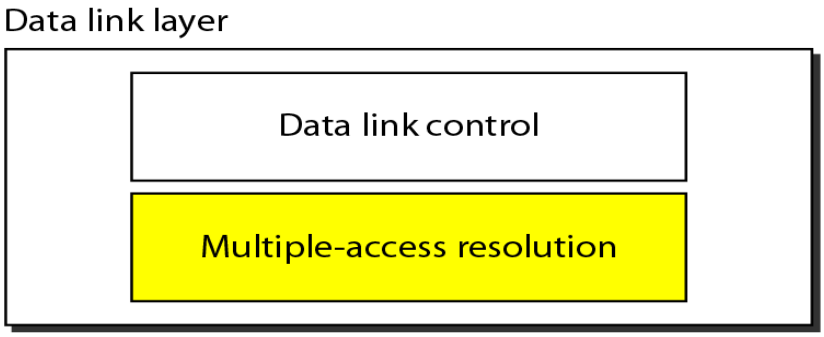
Introduction

The data link control, a mechanism which provides a link with reliable communication. In the protocols we described, we assumed that there is an available dedicated link (or channel) between the sender and the receiver. This assumption mayor may not be true. If, indeed, we have a dedicated link, as when we connect to the Internet using PPP as the data link control protocol, then the assumption is true and we do not need anything else.

- On the other hand, if we use our cellular phone to connect to another cellular phone, the channel (the band allocated to the vendor company) is not dedicated. A person a few feet away from us may be using the same channel to talk to her friend.

We can consider the data link layer as two sublayers.

- The upper sublayer is responsible for data link control, and the lower sublayer is responsible for resolving access to the shared media. If the channel is dedicated, we do not need the lower sublayer.
- Figure shows these two sublayers in the data link layer.

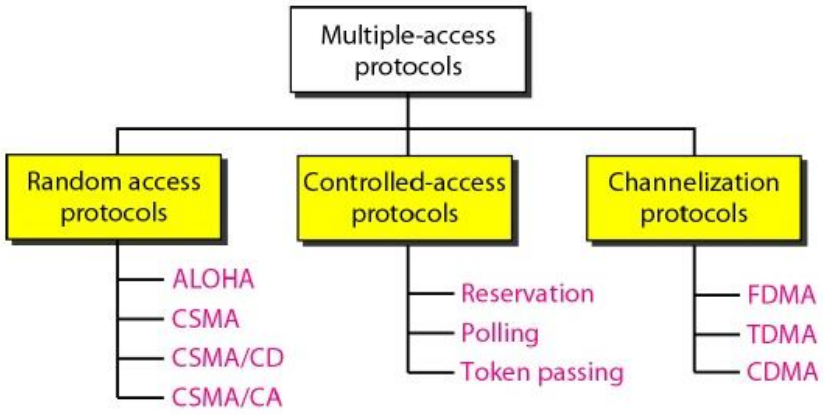


When nodes or stations are connected and use a common link, called a multipoint or broadcast link, we need a multiple-access protocol to coordinate access to the link.

- The problem of controlling the access to the medium is similar to the rules of speaking in an assembly. The procedures guarantee that the right to speak is upheld and ensure that two people do not speak at the same time, do not interrupt each other, do not monopolize the discussion, and so on.

The situation is similar for multipoint networks. Many formal protocols have been devised to handle access to a shared link. We categorize them into three groups. Protocols belonging to each group are shown in figure

Figure: Taxonomy of multiple-access protocols



RANDOM ACCESS

In random access or contention methods, no station is superior to another station and none is assigned the control over another.

- No station permits, or does not permit, another station to send. At each instance, a station that has data to send uses a procedure defined by the protocol to make a decision on whether or not to send.
- This decision depends on the state of the medium (idle or busy). In other words, each station can transmit when it desires on the condition that it follows the predefined procedure, including the testing of the state of the medium.

Two features give this method its name.

- First, there is no scheduled time for a station to transmit. Transmission is random among the stations. That is why these methods are called *random access.*
- Second, no rules specify which station should send next. Stations compete with one another to access the medium. That is why these methods are also called *contention* methods.

In a random access method, each station has the right to the medium without being controlled by any other station. However, if more than one station tries to send, there is an access conflict-collision-and the frames will be either destroyed or modified.

To avoid access conflict or to resolve it when it happens, each station follows a procedure that answers the following questions:

- When can the station access the medium?
- What can the station do if the medium is busy?
- How can the station determine the success or failure of the transmission?
- What can the station do if there is an access conflict?

The random access methods we study in this chapter have evolved from a very interesting protocol known as ALOHA, which used a very simple procedure called multiple access (MA). The method was improved with the addition of a procedure that forces the station to sense the medium before transmitting. This was called carrier sense multiple access. This method later evolved into two parallel methods: carrier sense multiple access with collision detection (CSMA/CD) and carrier sense multiple access with collision avoidance *(CSMA/CA)*. *CSMA/CD* tells the station what to do when a collision is detected. *CSMA/CA* tries to avoid the collision.

ALOHA

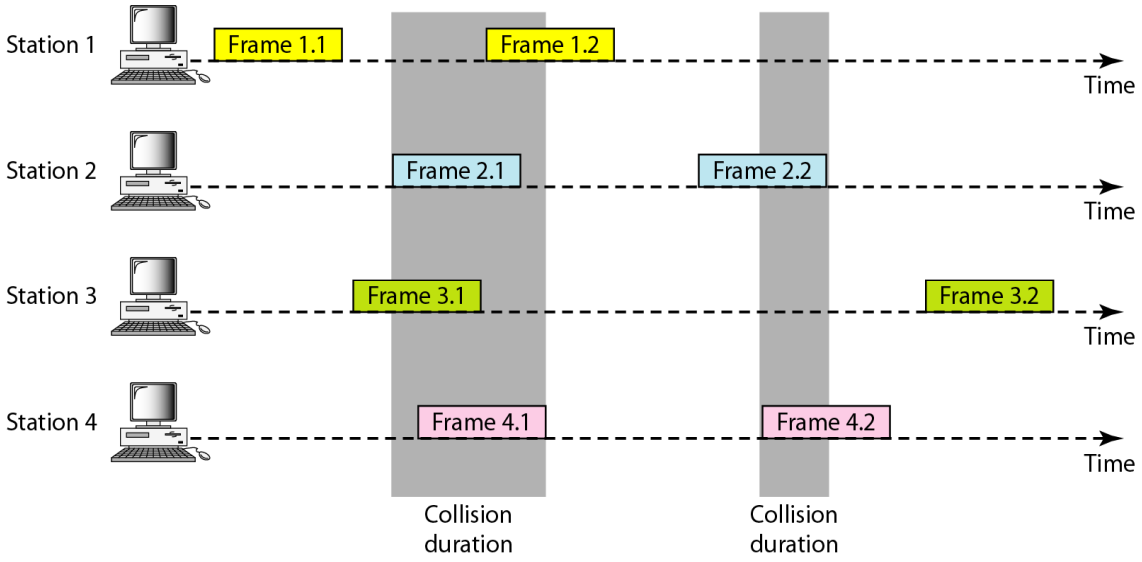
ALOHA, the earliest random access method, was developed at the University of Hawaii in early 1970. It was designed for a radio (wireless) LAN, but it can be used on any shared medium.

It is obvious that there are potential collisions in this arrangement. The medium is shared between the stations. When a station sends data, another station may attempt to do so at the same time. The data from the two stations collide and become garbled.

Pure ALOHA

The original ALOHA protocol is called pure ALOHA. This is a simple, but elegant protocol.

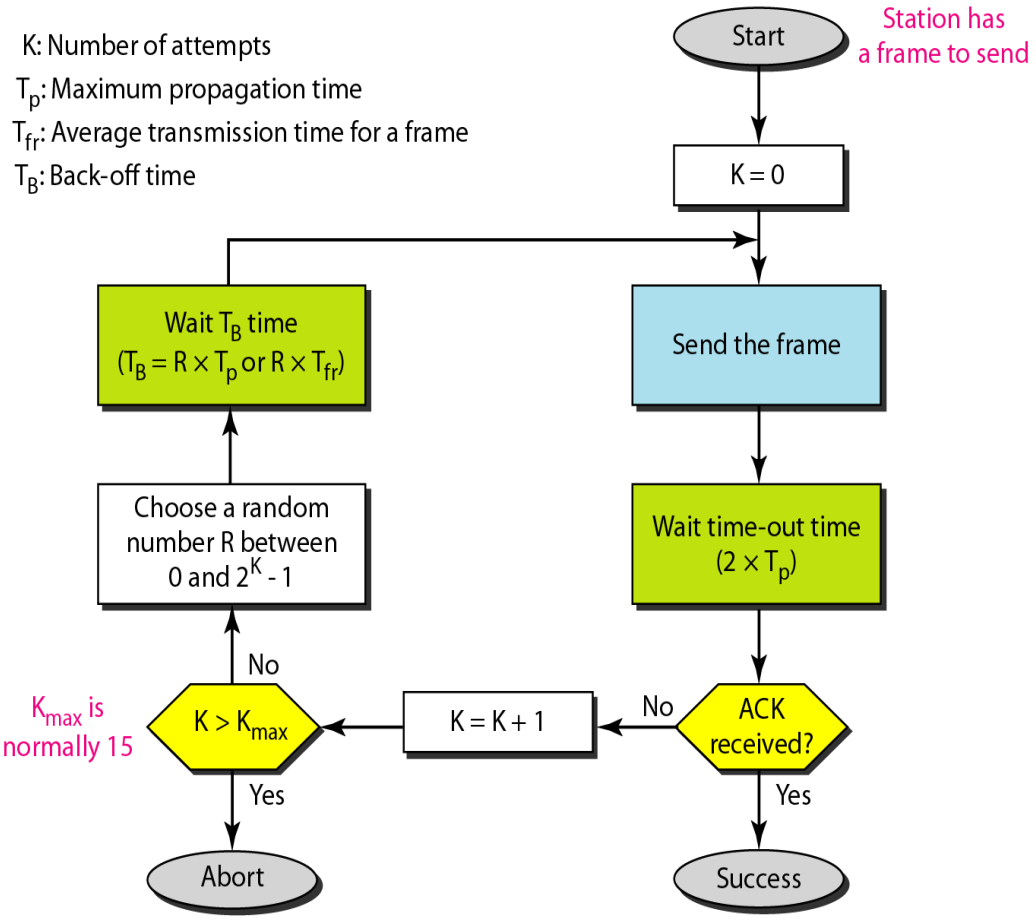
- The idea is that each station sends a frame whenever it has a frame to send. However, since there is only one channel to share, there is the possibility of collision between frames from different stations.
- Figure shows an example of frame collisions in pure ALOHA.



- There are four stations (unrealistic assumption) that contend with one another for access to the shared channel. The figure shows that each station sends two frames; there are a total of eight frames on the shared medium. Some of these frames collide because multiple frames are in contention for the shared channel.
- Figure shows that only two frames survive: frame 1.1 from station 1 and frame 3.2 from station 3. We need to mention that even if one bit of a frame coexists on the channel with one bit from another frame, there is a collision and both will be destroyed.
- It is obvious that we need to resend the frames that have been destroyed during transmission. The pure ALOHA protocol relies on acknowledgments from the receiver. When a station sends a frame, it expects the receiver to send an acknowledgment.
- If the acknowledgment does not arrive after a time-out period, the station assumes that the frame (or the acknowledgment) has been destroyed and resends the frame.

A collision involves two or more stations. If all these stations try to resend their frames after the time-out, the frames will collide again.

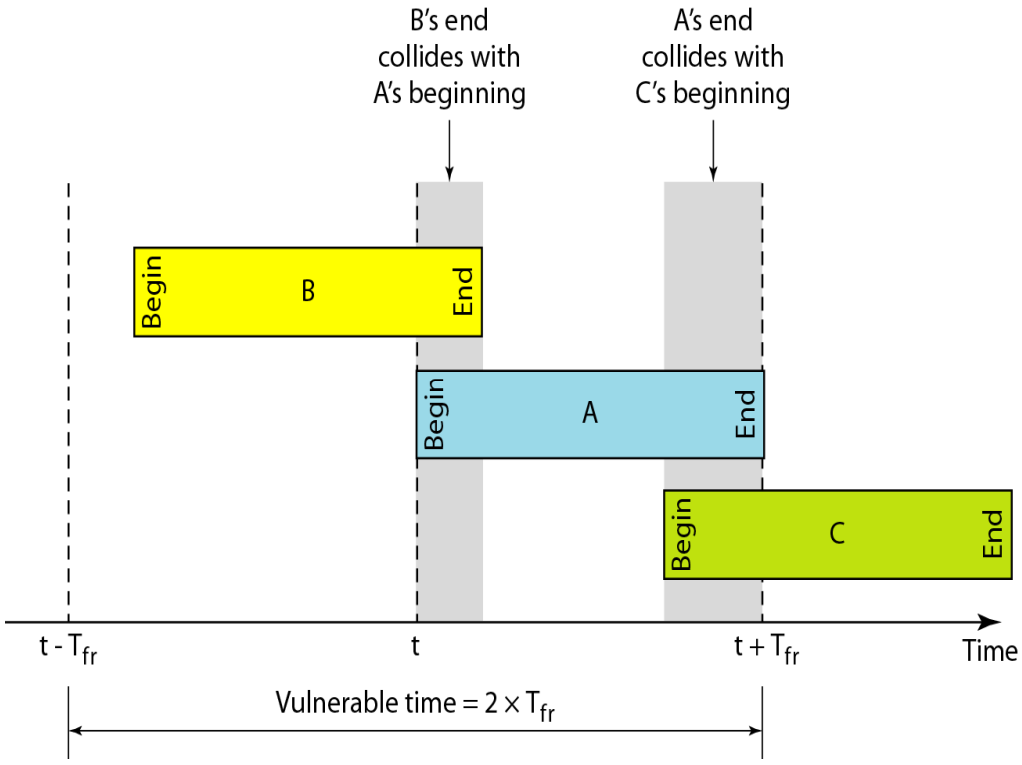
- Pure ALOHA dictates that when the time-out period passes, each station waits a random amount of time before resending its frame. The randomness will help avoid more collisions. We call this time the back-off time T_B.
- Pure ALOHA has a second method to prevent congesting the channel with retransmitted frames.
- After a maximum number of retransmission attempts K_{max}, a station must give up and try later. Figure shows the procedure for pure ALOHA based on the above strategy.



- The time-out period is equal to the maximum possible round-trip propagation delay, which is twice the amount of time required to send a frame between the two most widely separated stations (2 x T_p).
- The back-off time T_B is a random value that normally depends on K (the number of attempted unsuccessful transmissions).
- The formula for T_B depends on the implementation. One common formula is the **binary exponential back-off.** In this method, for each retransmission, a multiplier in the range 0 to 2^K - 1 is randomly chosen and multiplied by T_p (maximum propagation time) or T_{fr} (the average time required to send out a frame) to find T_B.
- Note that in this procedure, the range of the random numbers increases after each collision. The value of K_{max} is usually chosen as 15.

Vulnerable time Let us find the length of time, the **vulnerable time,** in which there is a possibility of collision. We assume that the stations send fixed-length frames with each frame taking T_{fr} s to send. Figure shows the vulnerable time for station A.

-



- Station A sends a frame at time *t*. Now imagine station B has already sent a frame between $t - T_{fr}$ and *t*. This leads to a collision between the frames from station A and station B. The end of B's frame collides with the beginning of A's frame.
- On the other hand, suppose that station C sends a frame between *t* and $t + T_{fr}$. Here, there is a collision between frames from station A and station C.
- The beginning of C's frame collides with the end of A's frame.
- Looking at figure , we see that the vulnerable time, during which a collision may occur in pure ALOHA, is 2 times the frame transmission time.

Pure ALOHA vulnerable time = 2 x T_{fr}

Throughput : Let us call G the average number of frames generated by the system during one frame transmission time.

- Then it can be proved that the average number of successful transmissions for pure ALOHA is $S = G \times e^{-2G}$.
- The maximum throughput S_{max} is 0.184, for G = 1. In other words, if one-half a frame is generated during one frame transmission time (in other words, one frame during two frame transmission times), then 18.4 percent of these frames reach their destination successfully. This is an expected result because the vulnerable time is 2 times the frame transmission time.
- Therefore, if a station generates only one frame in this vulnerable time (and no other stations generate a frame during this time), the frame will reach its destination successfully.

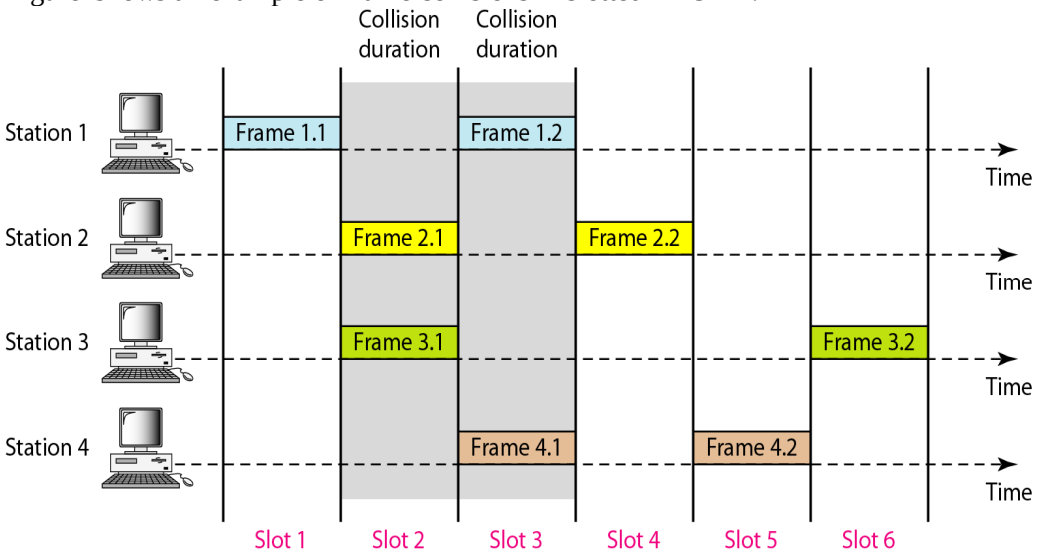
The throughput for pure ALOHA is $S = G \times e^{-2G}$.

The maximum throughput $S_{max} = 0.184$ when $G = (1/2)$.

Slotted ALOHA

Pure ALOHA has a vulnerable time of 2 x T_{fr} . This is so because there is no rule that defines when the station can send.

- A station may send soon after another station has started or soon before another station has finished. Slotted ALOHA was invented to improve the efficiency of pure ALOHA.
- In slotted ALOHA we divide the time into slots of T_{fr} and force the station to send only at the beginning of the time slot.
- Figure shows an example of frame collisions in slotted ALOHA.



- Because a station is allowed to send only at the beginning of the synchronized time slot, if a station misses this moment, it must wait until the beginning of the next time slot.
- This means that the station which started at the beginning of this slot has already finished sending its frame.
- Of course, there is still the possibility of collision if two stations try to send at the beginning of the same time slot. However, the vulnerable time is now reduced to one-half, equal to T_{fr}. Figure shows the situation.
- Figure shows that the vulnerable time for slotted ALOHA is one-half that of pure ALOHA.

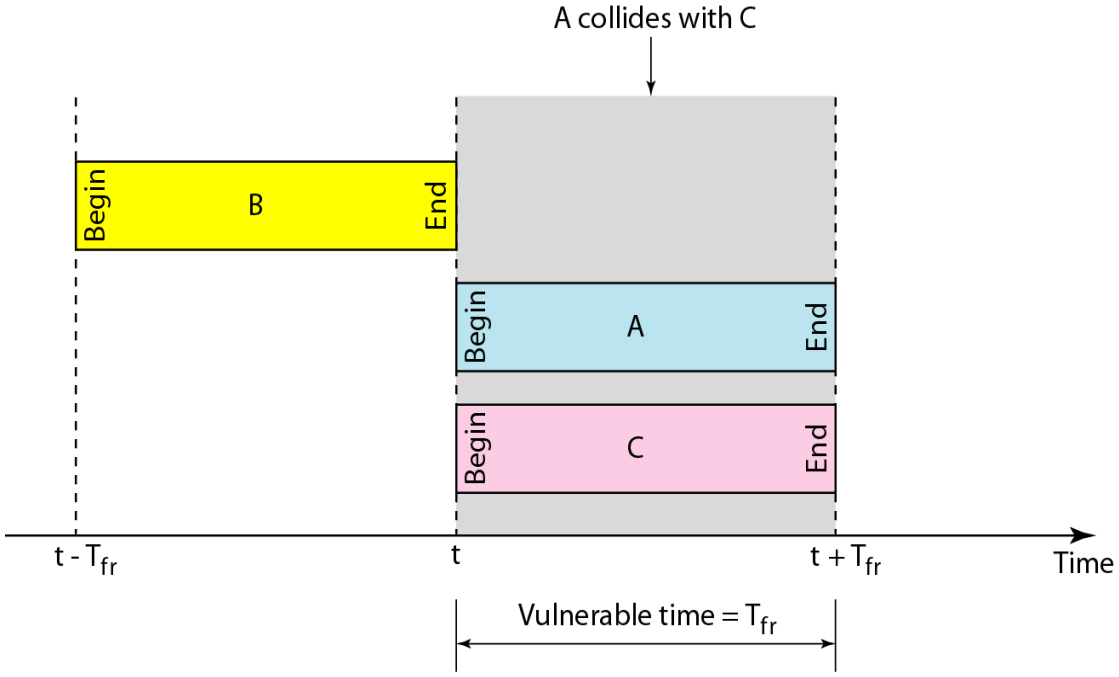
Slotted ALOHA vulnerable time = T_{fr}

Throughput: It can be proved that the average number of successful transmissions for slotted ALOHA is S = G x e^{-G}.

- The maximum throughput Smax is 0.368, when G = 1. In other words, if a frame is generated during one frame transmission time, then 36.8 percent of these frames reach their destination successfully.
- This result can be expected because the vulnerable time is equal to the frame transmission time. Therefore, if a station generates only one frame in this vulnerable time (and no other station generates a frame during this time), the frame will reach its destination successfully.

The throughput for slottedALOHA is S =: G x e^{-}G.

The maximum throughput *Smax* == 0.368 when G=1.



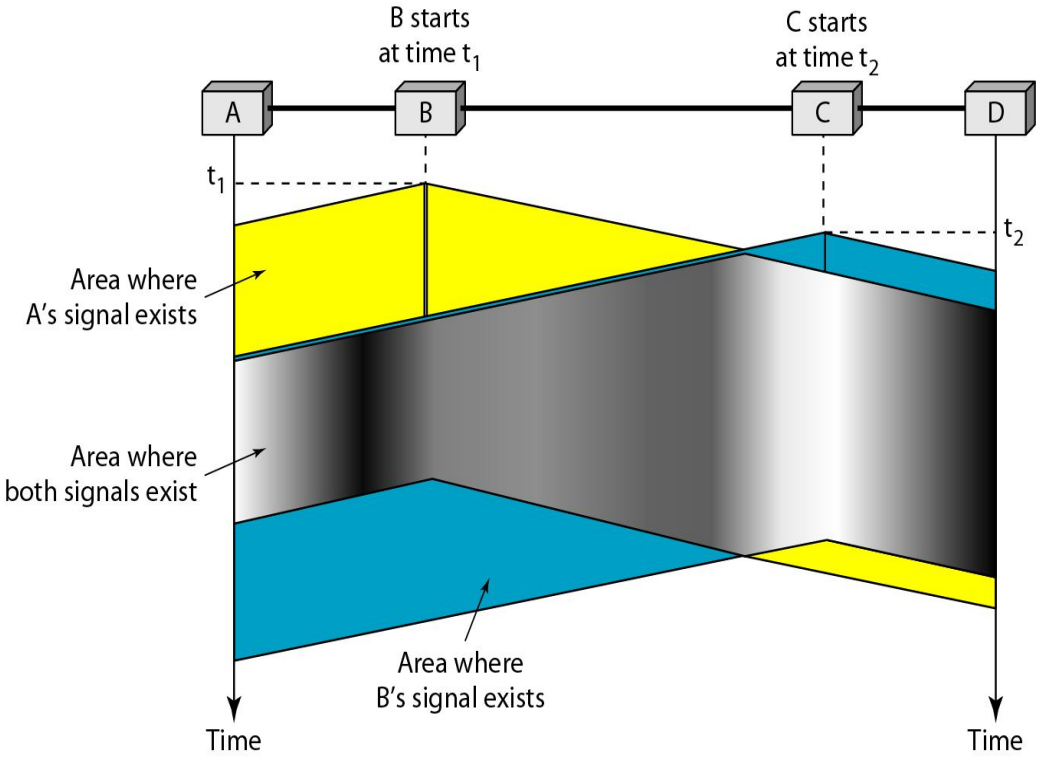
Carrier Sense Multiple Access (CSMA)

To minimize the chance of collision and, therefore, increase the performance, the CSMA method was developed.

- The chance of collision can be reduced if a station senses the medium before trying to use it. Carrier sense multiple access (CSMA) requires that each station first listen to the medium (or check the state of the medium) before sending.
- In other words, CSMA is based on the principle "sense before transmit" or "listen before talk."

CSMA can reduce the possibility of collision, but it cannot eliminate it. The reason for this is shown in figure, a space and time model of a CSMA network. Stations are connected to a shared channel.

- The possibility of collision still exists because of propagation delay; when a station sends a frame, it still takes time (although very short) for the first bit to reach every station and for every station to sense it.
- In other words, a station may sense the medium and find it idle, only because the first bit sent by another station has not yet been received.
- At time t_1, station B senses the medium and finds it idle, so it sends a frame. At time $t2$ $(t_2 > t_1)$, station C senses the medium and finds it idle because, at this time, the first bits from station B have not reached station C. Station C also sends a frame. The two signals collide and both frames are destroyed.



Vulnerable Time

The vulnerable time for CSMA is the propagation time T_p . This is the time needed for a signal to propagate from one end of the medium to the other.

- When a station sends a frame, and any other station tries to send a frame during this time, a collision will result. But if the first bit of the frame reaches the end of the medium, every station will already have heard the bit and will refrain from sending. Figure shows the worst case.
- The leftmost station A sends a frame at time t_1, which reaches the rightmost station D at time $t_1 + T_p$. The gray area shows the vulnerable area in time and space.

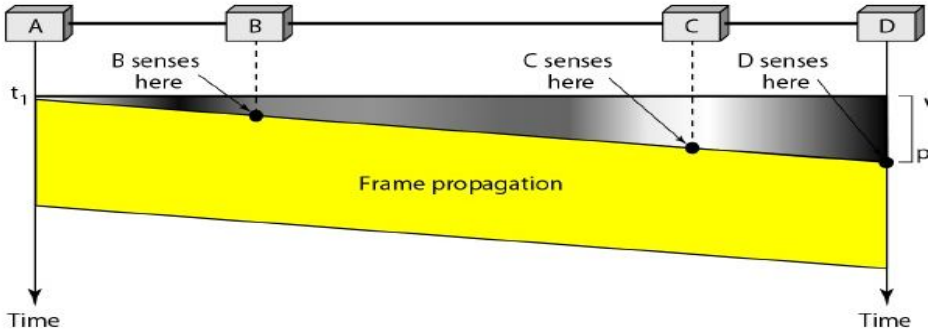
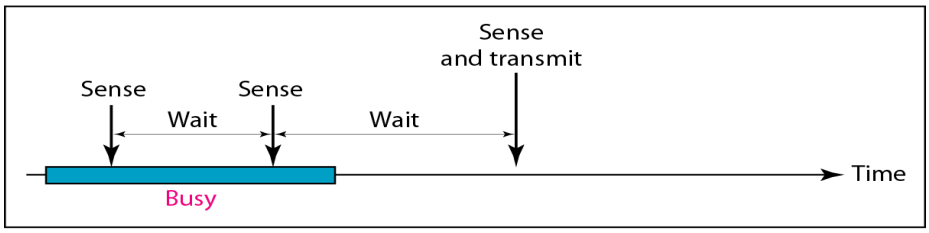


Figure: Vulnerable time in CSMA

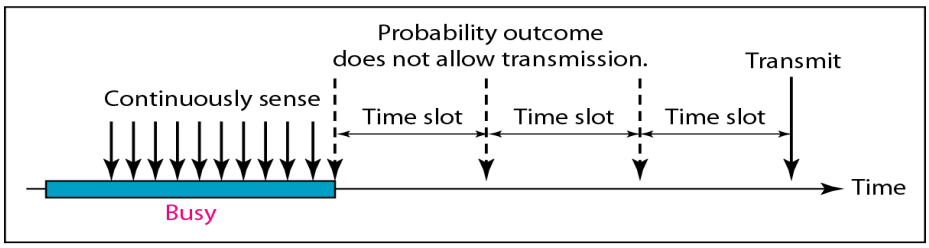
Persistence Methods

What should a station do if the channel is busy? What should a station do if the channel is idle? Three methods have been devised to answer these questions: the 1-persistent method, the nonpersistent method, and the p-persistent method. Figure shows the behavior of three persistence methods when a station finds a channel busy.

a. 1-persistent



b. Nonpersistent

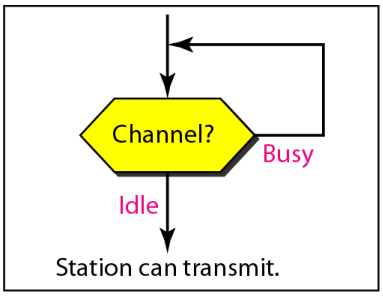


c. p-persistent

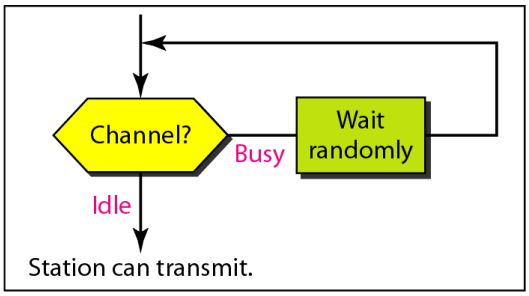
1-Persistent: The **1-persistent method** is simple and straightforward. **In** this method, after the station finds the line idle, it sends its frame immediately (with probability 1). This method has the highest chance of collision because two or more stations may find the line idle and send their frames immediately.

Nonpersistent : In the **nonpersistent method,** a station that has a frame to send senses the line. If the line is idle, it sends immediately. If the line is not idle, it waits a random amount of time and then senses the line again.

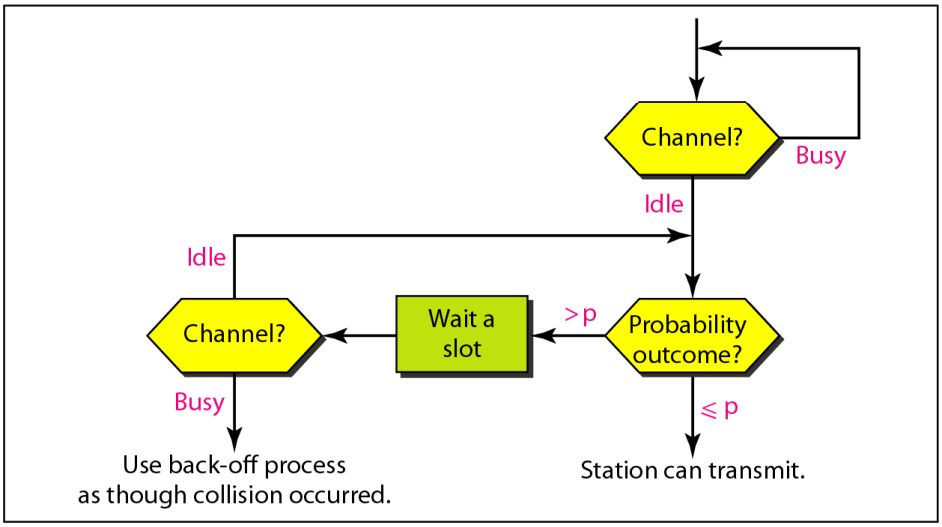
- The nonpersistent approach reduces the chance of collision because it is unlikely that two or more stations will wait the same amount of time and retry to send simultaneously. However, this method reduces the efficiency of the network because the medium remains idle when there may be stations with frames to send.



a. 1-persistent



b. Nonpersistent



c. p-persistent

p-Persistent : The **p-persistent method** is used if the channel has time slots with a slot duration equal to or greater than the maximum propagation time.

- The p-persistent approach combines the advantages of the other two strategies. It reduces the chance of collision and improves efficiency. In this method, after the station finds the line idle it follows these steps:

1. With probability *p*, the station sends its frame.

2. With probability $q = 1 - p$, the station waits for the beginning of the next time slot and checks the line again.

a. If the line is idle, it goes to step 1.

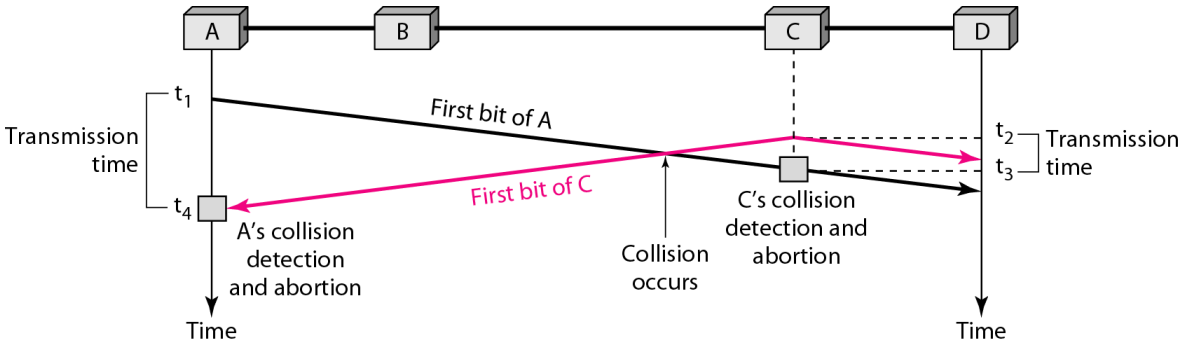
b. If the line is busy, it acts as though a collision has occurred and uses the back off procedure.

Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

The CSMA method does not specify the procedure following a collision. Carrier sense multiple access with collision detection (CSMA/CD) augments the algorithm to handle the collision.

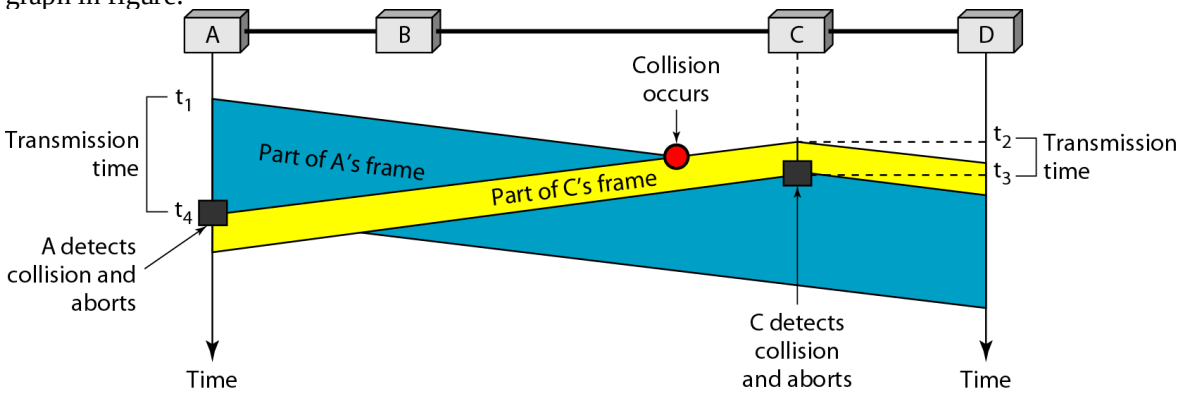
- In this method, a station monitors the medium after it sends a frame to see if the transmission was successful. If so, the station is finished. If, however, there is a collision, the frame is sent again.

- To better understand CSMA/CD, let us look at the first bits transmitted by the two stations involved in the collision. Although each station continues to send bits in the frame until it detects the collision, we show what happens as the first bits collide. In figure , stations A and C are involved in the collision.



- At time t_1, station A has executed its persistence procedure and starts sending the bits of its frame. At time t_2, station C has not yet sensed the first bit sent by A. Station C executes its persistence procedure and starts sending the bits in its frame, which propagate both to the left and to the right.
- The collision occurs sometime after time t_2, Station C detects a collision at time t_3 when it receives the first bit of A's frame. Station C immediately (or after a short time, but we assume immediately) aborts transmission. Station A detects collision at time t_4 when it receives the first bit of C's frame; it also immediately aborts transmission.
- Looking at the figure, we see that A transmits for the duration $t_4 - t_1$;C transmits for the duration $t_3 - t_2'$. Later we show that, for the protocol to work, the length of any frame divided by the bit rate in this protocol must be more than either of these durations.
- At time t_4, the transmission of *A:s* frame, though incomplete, is aborted;at time t_3, the transmission of B's frame, though incomplete, is aborted.

Now that we know the time durations for the two transmissions, we can show a more complete graph in figure.



Minimum Frame Size

For *CSMA/CD* to work, we need a restriction on the frame size. Before sending the last bit of the frame, the sending station must detect a collision, if any, and abort the transmission.

- This is so because the station, once the entire frame is sent, does not keep a copy of the frame and does not monitor the line for collision detection. Therefore, the frame

transmission time T_{fr} must be at least two times the maximum propagation time T_p. To understand the reason, let us think about the worst-case scenario.

- If the two stations involved in a collision are the maximum distance apart, the signal from the first takes time T_p to reach the second, and the effect of the collision takes another time T_p to reach the first. So the requirement is that the first station must still be transmitting after $2T_p$.

Procedure

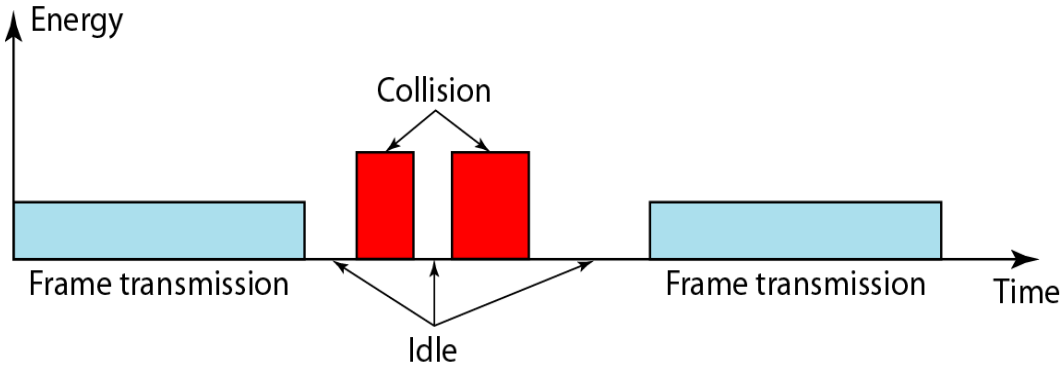
Now let us look at the flow diagram for *CSMA/CD* in figure . It is similar to the one for the ALOHA protocol, but there are differences.

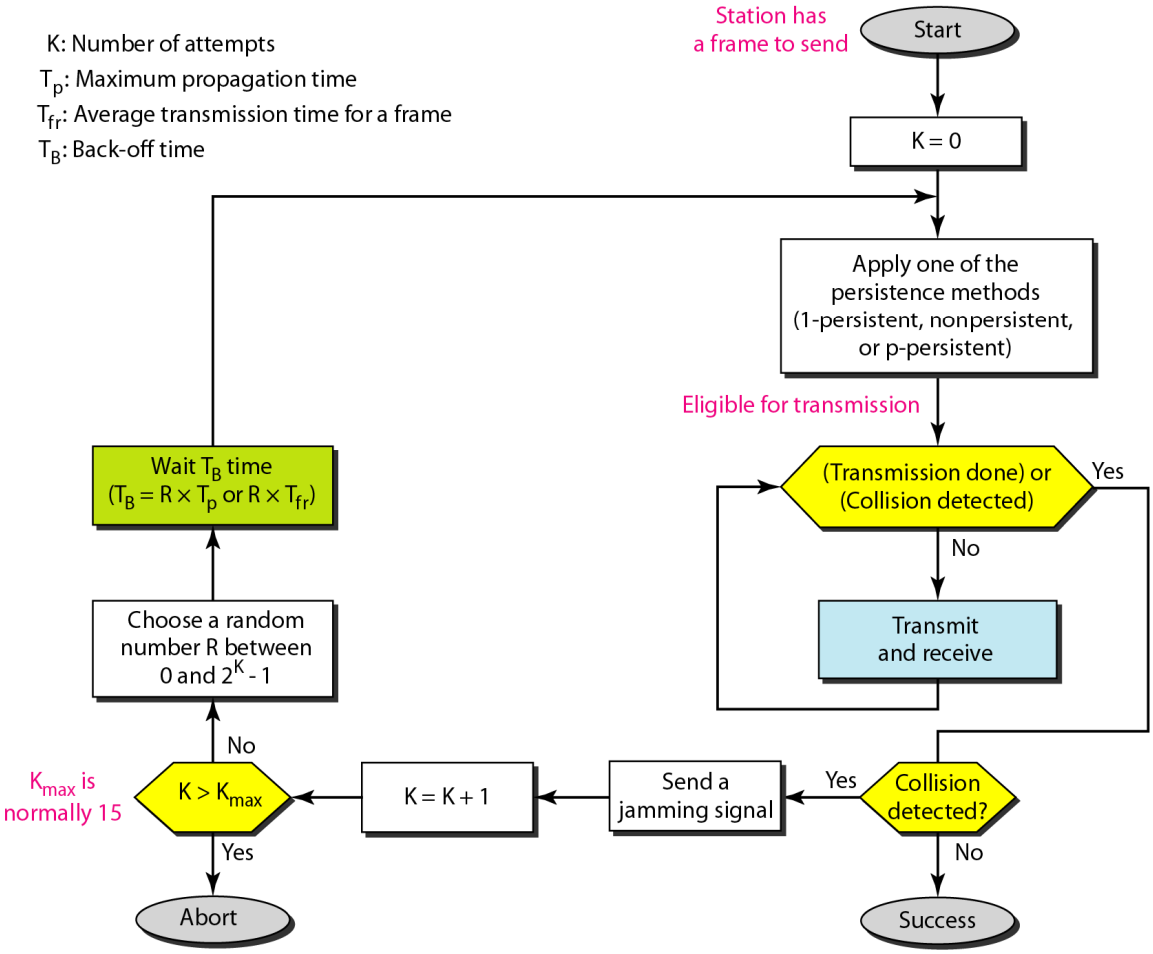
- The first difference is the addition of the persistence process. We need to sense the channel before we start sending the frame by using one of the persistence processes we discussed previously (nonpersistent, I-persistent, or p-persistent). The corresponding box can be replaced by one of the persistence processes shown in figure.
- The second difference is the frame transmission. In ALOHA, we first transmit the entire frame and then wait for an acknowledgment. In *CSMA/CD,* transmission and collision detection is a continuous process. We do not send the entire frame and then look for a collision.
- The station transmits and receives continuously and simultaneously (using two different ports). We use a loop to show that transmission is a continuous process. We constantly monitor in order to detect one of two conditions: either transmission is finished or a collision is detected. Either event stops transmission.
- When we come out of the loop, if a collision has not been detected, it means that transmission is complete; the entire frame is transmitted. Otherwise, a collision has occurred.
- The third difference is the sending of a short jamming signal that enforces the collision in case other stations have not yet sensed the collision.

Energy Level

We can say that the level of energy in a channel can have three values: zero, normal, and abnormal. At the zero level, the channel is idle.

- At the normal level, a station has successfully captured the channel and is sending its frame.
- At the abnormal level, there is a collision and the level of the energy is twice the normal level. A station that has a frame to send or is sending a frame needs to monitor the energy level to determine if the channel is idle, busy, or in collision mode.
- Figure shows the situation.





Throughput

The throughput of *CSMA/CD* is greater than that of pure or slotted ALOHA. The maximum throughput occurs at a different value of G and is based on the persistence method and the value of *p* in the p-persistent approach.

- For 1-persistent method the maximum throughput is around 50 percent when G =1. For nonpersistent method, the maximum throughput can go up to 90 percent when G is between 3 and 8.

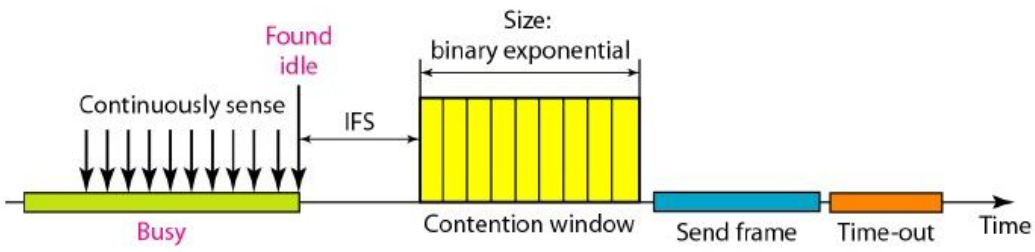
Carrier Sense Multiple Access withCollision Avoidance (CSMA/CA)

The basic idea behind *CSMA/CD* is that a station needs to be able to receive while transmitting to detect a collision.

- When there is no collision, the station receives one signal: its own signal. When there is a collision, the station receives two signals: its own signal and the signal transmitted by a second station.
- To distinguish between these two cases, the received signals in these two cases must be significantly different. In other words, the signal from the second station needs to add a significant amount of energy to the one created by the first station.

In a wired network, the received signal has almost the same energy as the sent signal because either the length of the cable is short or there are repeaters that amplify the energy between the sender and the receiver. This means that in a collision, the detected energy almost doubles.

- However, in a wireless network, much of the sent energy is lost in transmission. The received signal has very little energy. Therefore, a collision may add only 5 to 10 percent additional energy. This is not useful for effective collision detection.
- We need to avoid collisions on wireless networks because they cannot be detected. Carrier sense multiple access with collision avoidance *(CSMA/CA)* was invented for this network. Collisions are avoided through the use of CSMA/CA's three strategies: the interframe space, the contention window, and acknowledgments, as shown in figure.



In terframe Space (IFS)

First, collisions are avoided by deferring transmission even if the channel is found idle. When an idle channel is found, the station does not send immediately. It waits for a period of time called the interframe space or IFS.

- Even though the channel may appear idle when it is sensed, a distant station may have already started transmitting. The distant station's signal has not yet reached this station. The IFS time allows the front of the transmitted signal by the distant station to reach this station.

- If after the IFS time the channel is still idle, the station can send, but it still needs to wait a time equal to the contention time (described next). The IFS variable can also be used to prioritize stations or frame types. For example, a station that is assigned a shorter IFS has a higher priority.

Contention Window

The contention window is an amount of time divided into slots. A station that is ready to send chooses a random number of slots as its wait time.

- The number of slots in the window changes according to the binary exponential back-off strategy.
- This means that it is set to one slot the first time and then doubles each time the station cannot detect an idle channel after the IFS time.
- This is very similar to the p-persistent method except that a random outcome defines the number of slots taken by the waiting station. One interesting point about the contention window is that the station needs to sense the channel after each time slot. However, if the station finds the channel busy, it does not restart the process; it just stops the timer and restarts it when the channel is sensed as idle. This gives priority to the station with the longest waiting time.

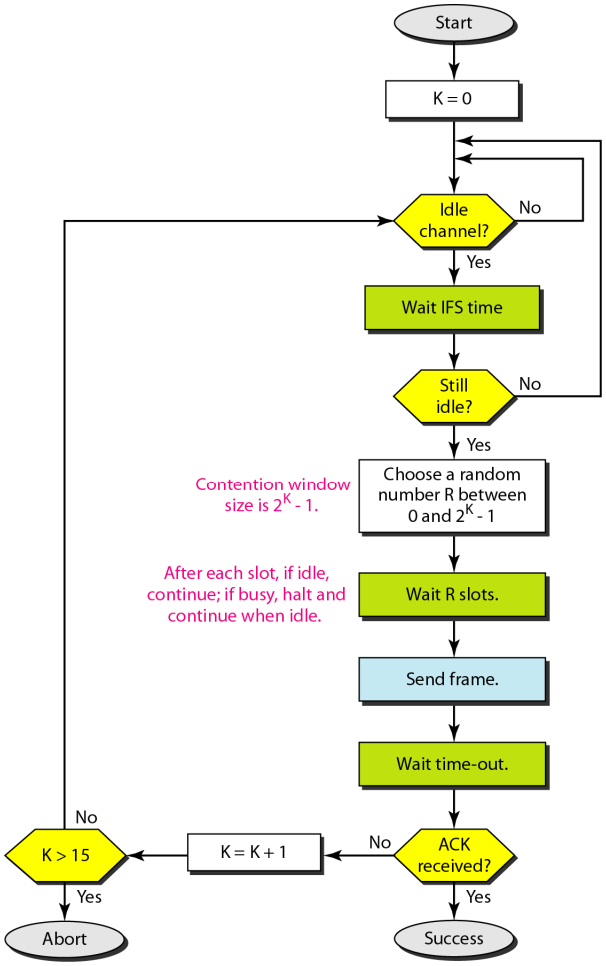
Acknowledgment

With all these precautions, there still may be a collision resulting in destroyed data. In addition, the data may be corrupted during the transmission. The positive acknowledgment and the time-out timer can help guarantee that the receiver has received the frame.

Procedure

Figure shows the procedure. Note that the channel needs to be sensed before and after the IFS. The channel also needs to be sensed during the contention time.

- For each time slot of the contention window, the channel is sensed. If it is found idle, the timer continues; if the channel is found busy, the timer is stopped and continues after the timer becomes idle again.



CSMA/CA and Wireless Networks

CSMA/CA was mostly intended for use in wireless networks. The procedure described above, however, is not sophisticated enough to handle some particular issues related to wireless networks, such as hidden terminals or exposed terminals.

CONTROLLED ACCESS

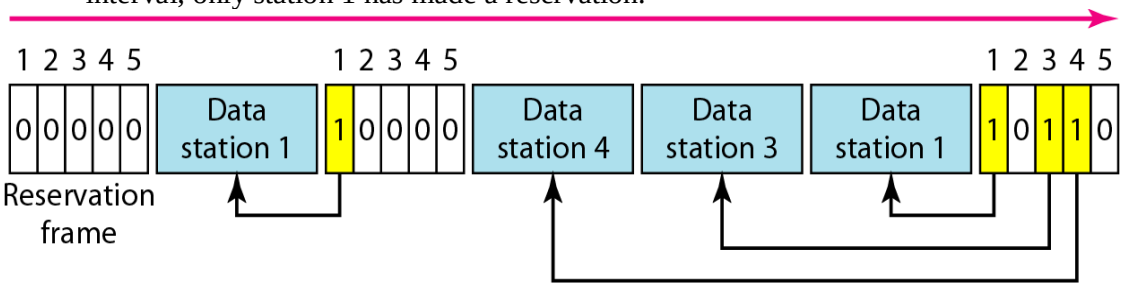
In controlled access, the stations consult one another to find which station has the right to send. A station cannot send unless it has been authorized by other stations. We discuss three popular controlled-access methods.

Reservation

In the reservation method, a station needs to make a reservation before sending data. Time is divided into intervals. In each interval, a reservation frame precedes the data frames sent in that interval.

- If there are *N* stations in the system, there are exactly *N* reservation minislots in the reservation frame. Each minislot belongs to a station. When a station needs to send a data frame, it makes a reservation in its own minislot. The stations that have made reservations can send their data frames after the reservation frame.

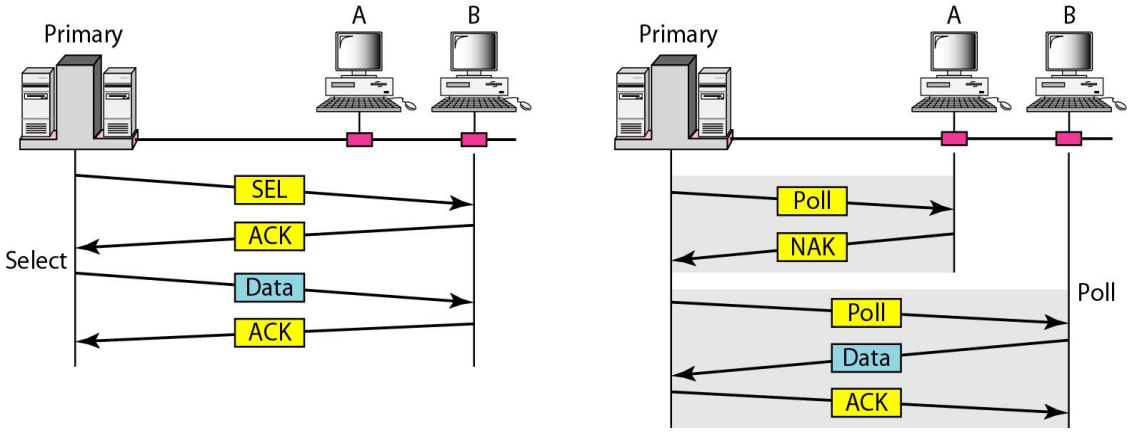
- Figure shows a situation with five stations and a five-minislot reservation frame. In the first interval, only stations 1, 3, and 4 have made reservations. In the second interval, only station 1 has made a reservation.



Polling

Polling works with topologies in which one device is designated as a primary station and the other devices are secondary stations.

- All data exchanges must be made through the primary device even when the ultimate destination is a secondary device.
- The primary device controls the link; the secondary devices follow its instructions. It is up to the primary device to determine which device is allowed to use the channel at a given time.
- The primary device, therefore, is always the initiator of a session



If the primary wants to receive data, it asks the secondaries if they have anything to send; this is called poll function. If the primary wants to send data, it tells the secondary to get ready to receive; this is called select function.

Select

The *select* function is used whenever the primary device has something to send. Remember that the primary controls the link. If the primary is neither sending nor receiving data, it knows the link is available.

- If it has something to send, the primary device sends it. What it does not know, however, is whether the target device is prepared to receive. So the primary must alert the secondary to the upcoming transmission and wait for an acknowledgment of the secondary's ready status.

- Before sending data, the primary creates and transmits a select (SEL) frame, one field of which includes the address of the intended secondary.

Poll

The *poll* function is used by the primary device to solicit transmissions from the secondary devices.

- When the primary is ready to receive data, it must ask (poll) each device in turn if it has anything to send.
- When the first secondary is approached, it responds either with a NAK frame if it has nothing to send or with data (in the form of a data frame) if it does.
- If the response is negative (a NAK frame), then the primary polls the next secondary in the same manner until it finds one with data to send.
- When the response is positive (a data frame), the primary reads the frame and returns an acknowledgment (ACK frame), verifying its receipt.

Token Passing

In the token-passing method, the stations in a network are organized in a logical ring. In other words, for each station, there is a *predecessor* and a *successor*.

- The predecessor is the station which is logically before the station in the ring; the successor is the station which is after the station in the ring.
- The current station is the one that is accessing the channel now. The right to this access has been passed from the predecessor to the current station.
- The right will be passed to the successor when the current station has no more data to send.

But how is the right to access the channel passed from one station to another? In this method, a special packet called a token circulates through the ring. The possession of the token gives the station the right to access the channel and send its data. When a station has some data to send, it waits until it receives the token from its predecessor.

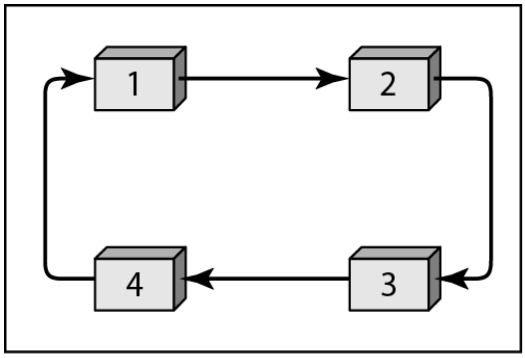
- It then holds the token and sends its data. When the station has no more data to send, it releases the token, passing it to the next logical station in the ring. The station cannot send data until it receives the token again in the next round. In this process, when a station receives the token and has no data to send, it just passes the data to the next station.

Token management is needed for this access method. Stations must be limited in the time they can have possession of the token. The token must be monitored to ensure it has not been lost or destroyed.

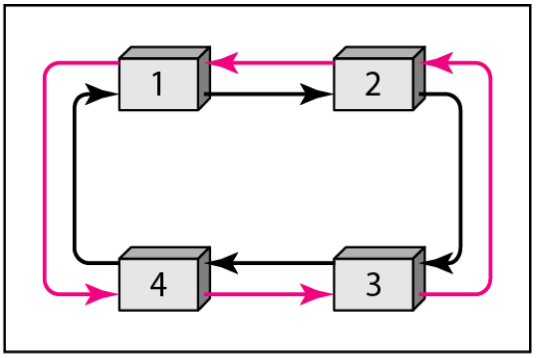
- For example, if a station that is holding the token fails, the token will disappear from the network. Another function of token management is to assign priorities to the stations and to the types of data being transmitted. And finally, token management is needed to make low-priority stations release the token to high priority stations.

Logical Ring

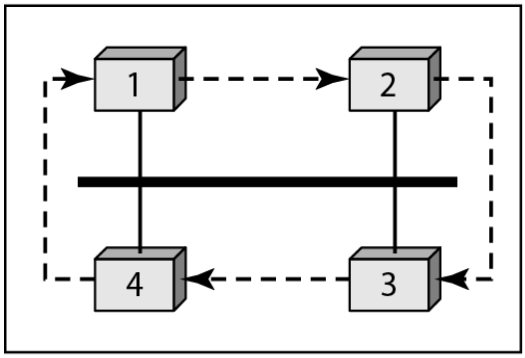
In a token-passing network, stations do not have to be physically connected in a ring; the ring can be a logical one. Figure show four different physical topologies that can create a logical ring.



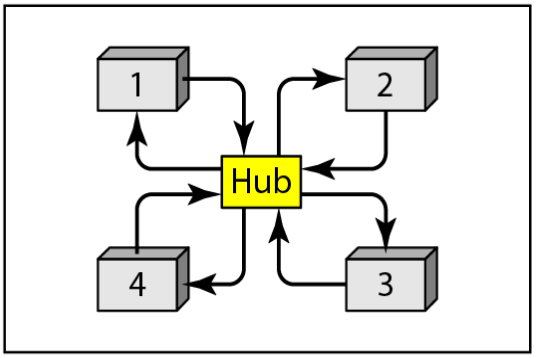
a. Physical ring



b. Dual ring



c. Bus ring



d. Star ring

In the physical ring topology, when a station sends the token to its successor, the token cannot be seen by other stations; the successor is the next one in line. This means that the token does not have to have the address of the next successor.

- The problem with this topology is that if one of the links-the medium between two adjacent stations fails, the whole system fails.

The dual ring topology uses a second (auxiliary) ring which operates in the reverse direction compared with the main ring. The second ring is for emergencies only (such as a spare tire for a car).

- If one of the links in the main ring fails, the system automatically combines the two rings to form a temporary ring.
- After the failed link is restored, the auxiliary ring becomes idle again. Note that for this topology to work, each station needs to have two transmitter ports and two receiver ports.
- The high-speed Token Ring networks called FDDI (Fiber Distributed Data Interface) and CDDI (Copper Distributed Data Interface) use this topology.

In the bus ring topology, also called a token bus, the stations are connected to a single cable called a bus.

- They, however, make a logical ring, because each station knows the address of its successor (and also predecessor for token management purposes).

When a station has finished sending its data, it releases the token and inserts the address of its successor in the token.

- Only the station with the address matching the destination address of the token gets the token to access the shared media. The Token Bus LAN, standardized by IEEE, uses this topology.

In a star ring topology, the physical topology is a star. There is a hub, however, that acts as the connector. The wiring inside the hub makes the ring; the stations are connected to this ring through the two wire connections.

- This topology makes the network less prone to failure because if a link goes down, it will be bypassed by the hub and the rest of the stations can operate. Also adding and removing stations from the ring is easier. This topology is still used in the Token Ring LAN designed by IBM.

UNIT-V

Assignment-Cum-Tutorial Questions

SECTION-A

I. Objective Questions

1. In _____ each station sends a frame whenever it has a frame to send.

a. pure ALOHA b. slotted ALOHA [ ]
c. both (a) and (b) d. neither (a) nor (b)

2. In pure ALOHA, the vulnerable time is _____ the frame transmission time.

a. the same as b. two times [ ]

c. three times d. three times

3. The maximum throughput for pure ALOHA is _____ per cent. [ ]

a. 12.2 b. 18.4 c. 36.8 d. 42.6

4. The vulnerable time for CSMA is the _____propagation time. [ ]

a. the same as b. two times

c. three times d. four times

5. In the _____ method, after the station finds the line idle, it sends its frame immediately. If the line is not idle, it continuously senses the line until it finds it idle. [ ]

a. non-persistent b. 1-persistent

c. *p*-persistent d. both a and b

6. In the _____method, a station that has a frame to send senses the line. If the line is idle, it sends immediately. If the line is not idle, it waits a random amount of time and then senses the line again. [ ]

a. nonpersistent b. 1-persistent

c. *p*-persistent d. both a and b

7. In the _____method, after the station finds the line idle it sends or refrain from sending based on the outcome of a random number generator. If the line is busy, it tries again. [ ]

a. nonpersistent b. 1-persistent

c. *p*-persistent d. both a and b

8. _____ augments the CSMA algorithm to detect collision. [ ]

a. CSMA/CA b. CSMA/CD

c. either (a) or (b) d. both (a) and (b)

9. . In _____ methods, a station cannot send unless it has been authorized by other stations. [ ]

a. random access b. controlled access

c. channelization d. both a and b

10. In the _____ method, a station needs to make a reservation before sending data. Time is divided into intervals. [ ]

a. reservation b. polling

c. token passing d. none of the above

11.In _____, each station is forced to send only at the beginning of the time slot. [ ]

a. pure ALOHA b. slotted ALOHA

c. both (a) and (b) d. neither (a) nor (b)

12. In _____ methods, no station is superior to another station and none is assigned the control over another. [ ]

a. random access b. controlled access

c. channelization d. none of the above

13. In _____, a station monitors the medium after it sends a frame to see if the transmission was successful. If so, the station is finished. If, however, there is a collision, the frame is sent again. [ ]

a. CSMA/CA b. CSMA/CD

c. either (a) or (b) d. both (a) and (b)

14. In the _____ method, time is divided into intervals. In each interval, a reservation frame precedes the data frames sent in that interval. [ ]

a. reservation b. polling

c. token passing d. none of the above

15. In _____ methods, the stations consult one another to find which station has the right to send. [ ]

a. random access b. controlled access

c. channelization d. none of the above

16. To avoid collisions on wireless networks, _____was invented.

a. CSMA/CA b. CSMA/CD [ ]

c. either (a) or (b) d. both (a) and (b)

17. In the _____ method, all data exchanges must be made through the primary device even when the ultimate destination is a secondary device.

a. reservation b. Polling [ ]

c. token passing d. none of the above

18. In _____, collisions are avoided through the use of three strategies: the interframe space, the contention window, and acknowledgments. [ ]

a. CSMA/CA b. CSMA/CD

c. either (a) or (b) d. both (a) and (b)

19. In the _____ method, the primary device controls the link; the secondary devices follow its instructions. [ ]

a. reservation b. polling

c. token passing d. none of the above

20. In the _____ method, a special packet called a _____ circulates through the ring. [ ]

a. reservation: control frame b. polling: poll request

c. token passing: token d. none of the above

SECTION-B

SUBJECTIVE QUESTIONS

1. What is pure ALOHA and slotted ALOHA? Mention the advantages of slotted ALOHA?
2. Explain the significance and usage of persistent methods in CSMA
3. Explain Carrier Sense Multiple Access with Collision avoidance(CSMA/CD)
4. Explain Carrier Sense Multiple Access with Collision Detection(CSMA/CD)
5. Explain Polling controlled access method
6. Discuss about token passing controlled access method
7. Compare and contrast a random access protocol with a controlled access protocol
8. What is contention window?
9. What is CSMA? Bring out the differences between 1-persistent, non-persistent, and p-persistent of CSMA?
10. What is meant by vulnerable period? Show that the vulnerable time period of slotted ALOHA is half of the pure ALOHA.

SECTION-C

QUESTIONS AT THE LEVEL OF GATE

1. A 2 km long broadcast LAN has 10^7 bps bandwidth and uses CSMA/CD. The signal travels along the wire at 2×10^8 m/s. What is the minimum packet size that can be used on this network? [GATE 2003]

UNIT-VI

Objectives:

To familiarize with the basics of wired LANs ,Wireless LANs and bridges

Syllabus:

IEEE Standards: Data link layer, physical layer, Manchester encoding, **Standard Ethernet**:MAC Sub Layer, physical layer, **IEEE - 802.11**: Architecture, MAC sub layer, frame structure. **Data Link Layer Switching**- Bridges, Local internet working Spanning tree bridges, remote bridges, switch virtual LANs.

Outcomes:

Students will be able to

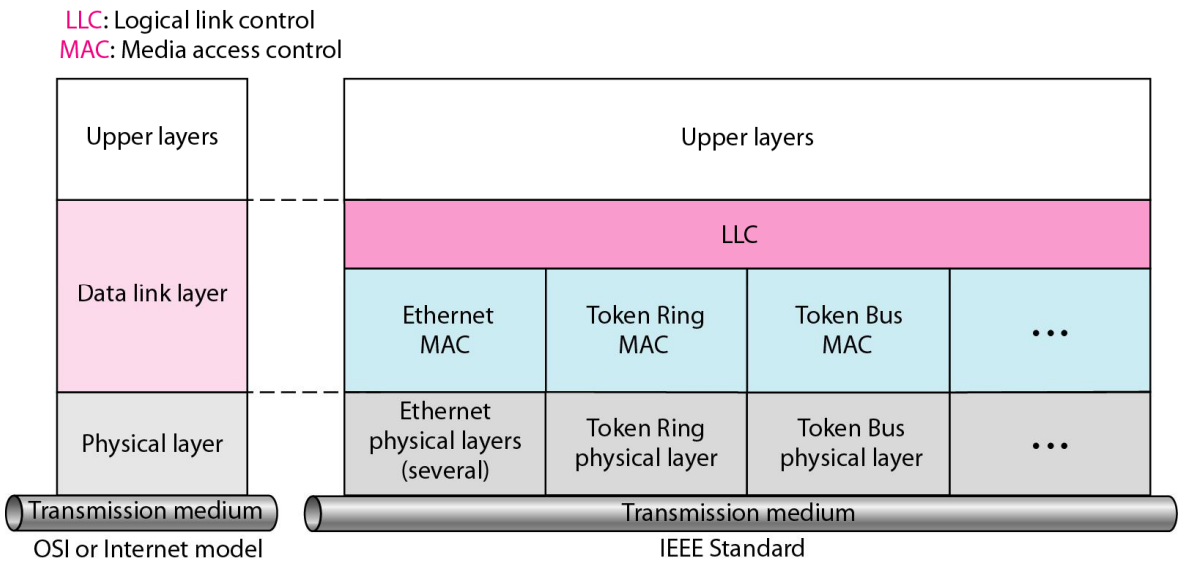
- Understand fundamental underlying principles of MAC sublayer in Standard Ethernet and IEEE 802.11
- Identify the different types of random access and controlled access methods
- To understand the architecture of the 802.11 and know the entities involved
- To understand the ethical issues in bridges
- Defining the concept of bridges, spanning tree bridges and virtual LANs

Learning Material

IEEE STANDARDS

In 1985, the Computer Society of the IEEE started a project, called Project 802, to set standards to enable intercommunication among equipment from a variety of manufacturers. Project 802 does not seek to replace any part of the OSI or the Internet model.

- Instead, it is a way of specifying functions of the physical layer and the data link layer of major LAN protocols.
- The standard was adopted by the American National Standards Institute (ANSI). In 1987, the International Organization for Standardization (ISO) also approved it as an international standard under the designation ISO 8802.
- The relationship of the 802 Standard to the traditional OSI model is shown in figure. The IEEE has subdivided the data link layer into two sublayers: logical link control (LLC) and media access control (MAC). IEEE has also created several physical layer standards for different LAN protocols.

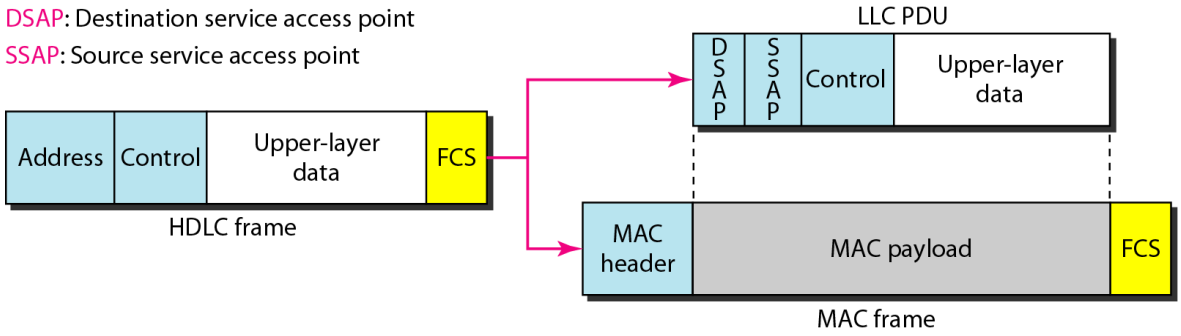


Data Link Layer

The data link layer in the IEEE standard is divided into two sublayers: LLC and MAC.

Logical Link Control (LLC): The data link control handles framing, flow control, and error control.

- In IEEE Project 802, flow control, error control, and part of the framing duties are collected into one sublayer called the logical link control.
- Framing is handled in both the LLC sublayer and the MAC sublayer.
- The LLC provides one single data link control protocol for all IEEE LANs. In this way, the LLC is different from the media access control sublayer, which provides different protocols for different LANs.
- A single LLC protocol can provide interconnectivity between different LANs because it makes the MAC sublayer transparent. Figure 13.1 shows one single LLC protocol serving several MAC protocols
- Framing LLC defines a protocol data unit (PDU) that is somewhat similar to that of HDLC. The header contains a control field like the one in HDLC; this field is used for flow and error control. The two other header fields define the upper-layer protocol at the source and destination that uses LLC. These fields are called the destination service access point (DSAP) and the source service access point (SSAP).
- The other fields defined in a typical data link control protocol such as HDLC are moved to the MAC sublayer. In other words, a frame defined in HDLC is divided into a PDU at the LLC sublayer and a frame at the MAC sublayer, as shown in figure



Media Access Control (MAC)

The multiple access methods including random access, controlled access, and channelization. IEEE Project 802 has created a sublayer called media access control that defines the specific access method for each LAN.

- For example, it defines CSMA/CD as the media access method for Ethernet LANs and the token passing method for Token Ring and Token Bus LANs.

Physical Layer:

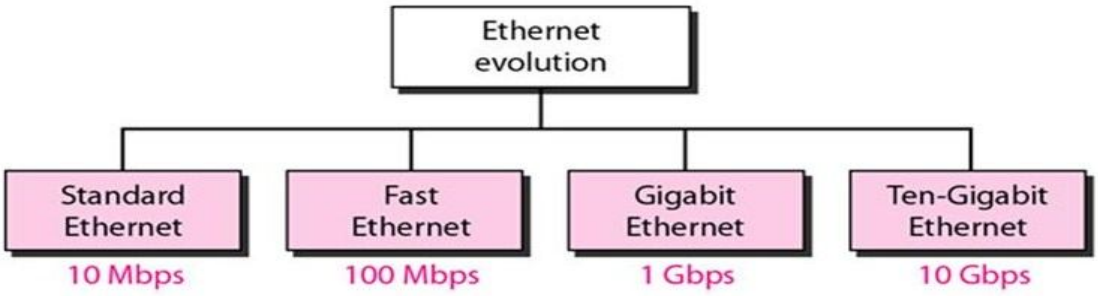
The physical layer is dependent on the implementation and type of physical media used. IEEE defines detailed specifications for each LAN implementation

---

STANDARD ETHERNET

The original Ethernet was created in 1976 at Xerox's Palo Alto Research Center (PARC). Since then, it has gone through four generations: Standard Ethernet (10 Mbps), Fast Ethernet (100 Mbps), Gigabit Ethernet (l Gbps), and Ten-Gigabit Ethernet (l0 Gbps), as shown in figure .

We briefly discuss all these generations starting with the first, Standard (or traditional) Ethernet



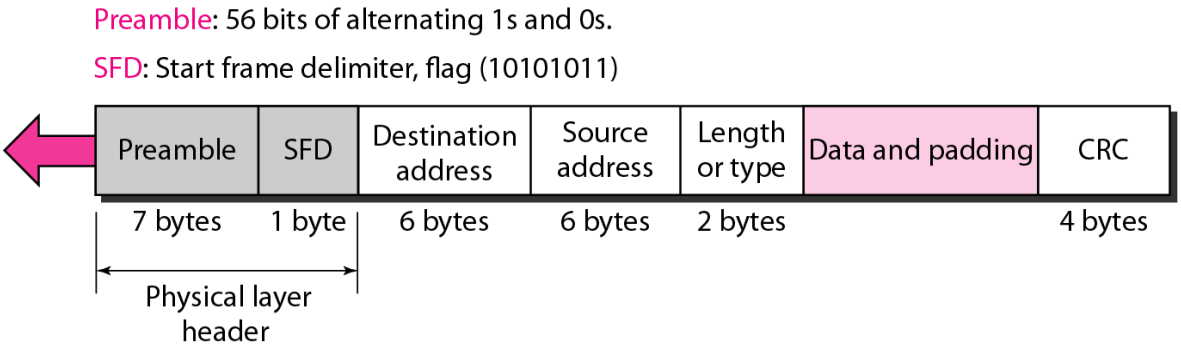
MAC Sublayer:

In Standard Ethernet, the MAC sublayer governs the operation of the access method. It also frames data received from the upper layer and passes them to the physical layer.

Frame Format

The Ethernet frame contains seven fields: preamble, SFD, DA, SA, length or type of protocol data unit (PDU), upper-layer data, and the CRC.

- Ethernet does not provide any mechanism for acknowledging received frames, making it what is known as an unreliable medium.
- Acknowledgments must be implemented at the higher layers.
- The format of the MAC frame is shown in figure



Preamble:The first field of the 802.3 frame contains 7 bytes (56 bits) of alternating 0s and 1s that alerts the receiving system to the coming frame and enables it to synchronize its input timing. The pattern provides only an alert and a timing pulse.

- The 56-bit pattern allows the stations to miss some bits at the beginning of the frame. The preamble is actually added at the physical layer and is not (formally) part of the frame.

Start frame delimiter (SFD): The second field (l byte: 10101011) signals the beginning of the frame. The SFD warns the station or stations that this is the last chance for synchronization. The last 2 bits is 11 and alerts the receiver that the next field is the destination address.

Destination address (DA): The DA field is 6 bytes and contains the physical address of the destination station or stations to receive the packet.

Source address (SA): The SA field is also 6 bytes and contains the physical address of the sender of the packet.

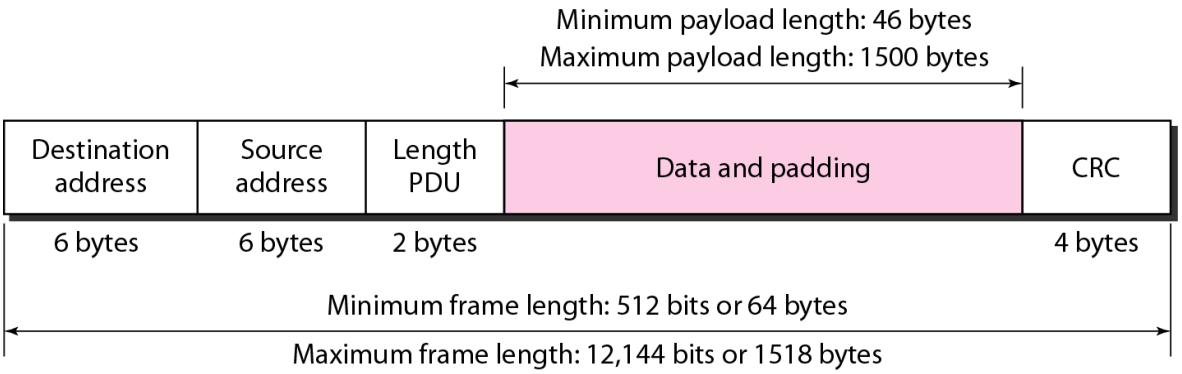
Length or type: This field is defined as a type field or length field. The original Ethernet used this field as the type field to define the upper-layer protocol using the MAC frame. The IEEE standard used it as the length field to define the number of bytes in the data field. Both uses are common today.

Data: This field carries data encapsulated from the upper-layer protocols. It is a minimum of 46 and a maximum of 1500 bytes.

CRC: The last field contains error detection information, in this case a CRC-32

Frame Length

Ethernet has imposed restrictions on both the minimum and maximum lengths of a frame,



The minimum length restriction is required for the correct operation of CSMA/CD as we will see shortly. An Ethernet frame needs to have a minimum length of 512 bits or 64 bytes. Part of this length is the header and the trailer.

- If we count 18 bytes of header and trailer (6 bytes of source address, 6 bytes of destination address, 2 bytes of length or type, and 4 bytes of CRC), then the minimum length of data from the upper layer is 64 - 18 = 46 bytes.
- If the upper-layer packet is less than 46 bytes, padding is added to make up the difference.

Addressing

Each station on an Ethernet network (such as a PC, workstation, or printer) has its own network interface card (NIC). The NIC fits inside the station and provides the station with a 6-byte physical address. As shown in Figure 13.6, the Ethernet address is 6 bytes (48 bits), normally written in hexadecimal notation, with a colon between the bytes.

Figure 13.6 ***Example of an Ethernet address in hexadecimal notation***

Unicast, Multicast, and Broadcast Addresses A source address is always a unicast address--the frame comes from only one station. The destination address, however, can be unicast, multicast, or broadcast. Figure 13.7 shows how to distinguish a unicast address from a multicast address. If the least significant bit of the first byte in a destination address is 0, the address is unicast; otherwise, it is multicast.

Figure 13.7 ***Unicast and multicast addresses***

13.7

Access Method: CSMA/CD

Standard Ethernet uses 1-persistent CSMA/CD

Slot Time In an Ethernet network, the round-trip time required for a frame to travel from one end of a maximum-length network to the other plus the time needed to send the jam sequence is called the slot time.

Slot time =round-trip time + time required to send the jam sequence

Slot Time and Maximum Network Length There is a relationship between the slot time and the maximum length of the network (collision domain). It is dependent on the propagation speed of the signal in the particular medium. In most transmission media, the signal propagates at 2 x 10^8 m/s (two-thirds of the rate for propagation in air). For traditional Ethernet, we calculate

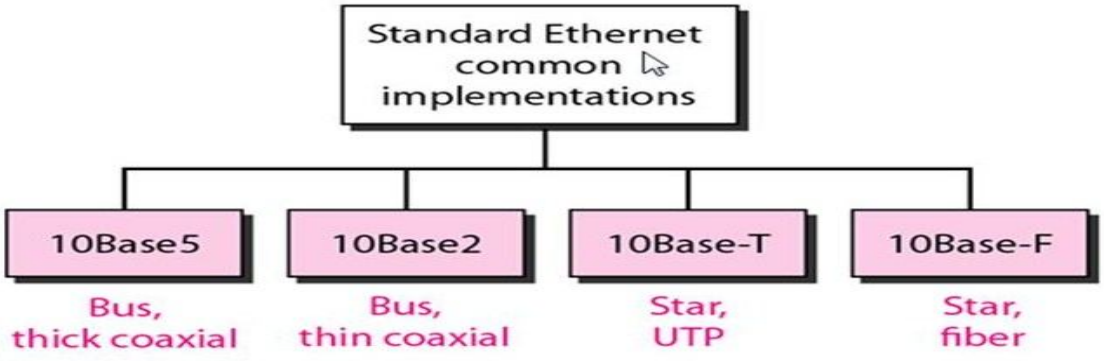
MaxLength = PropagationSpeed X SlotTime/ 2

Of course, we need to consider the delay times in repeaters and interfaces, and the time required to send the jam sequence. These reduce the maximum-length of a traditional Ethernet network to 2500 m, just 48 percent of the theoretical calculation.

MaxLength=2500 m

Physical Layer:

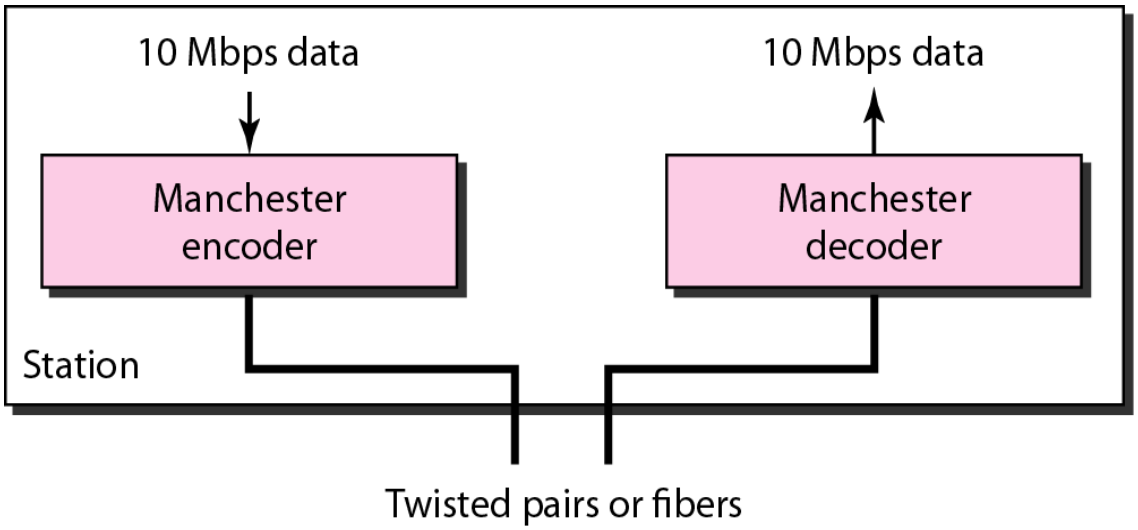
The Standard Ethernet defines several physical layer implementations; four of the most common, are shown in Figure



Encoding and Decoding

All standard implementations use digital signaling (baseband) at 10 Mbps. At the sender, data are converted to a digital signal using the Manchester scheme; at the receiver, the received signal is interpreted as Manchester and decoded into data Manchester encoding is self-synchronous, providing a transition at each bit interval.

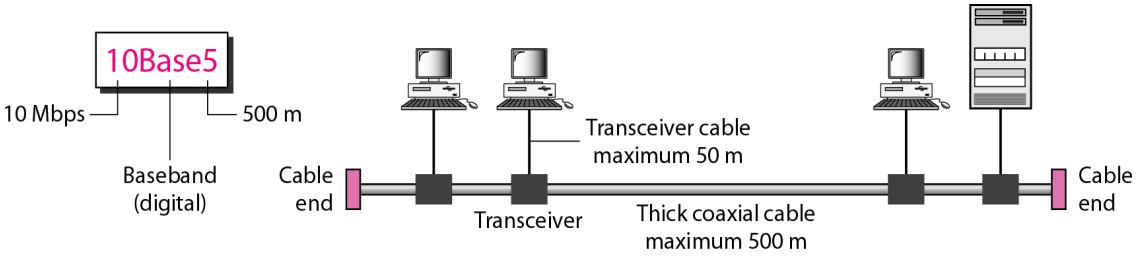
- Figure shows the encoding scheme for Standard Ethernet.



10Base5: Thick Ethernet

The first implementation is called 10Base5, thick Ethernet, or Thicknet. The nickname derives from the size of the cable, which is roughly the size of a garden hose and too stiff to bend with your hands.

- 10Base5 was the first Ethernet specification to use a bus topology with an external transceiver (transmitter/receiver) connected via a tap to a thick coaxial cable. Figure shows a schematic diagram of a 10Base5 implementation.



- The transceiver is responsible for transmitting, receiving, and detecting collisions. The transceiver is connected to the station via a transceiver cable that provides separate paths for sending and receiving. This means that collision can only happen in the coaxial cable.

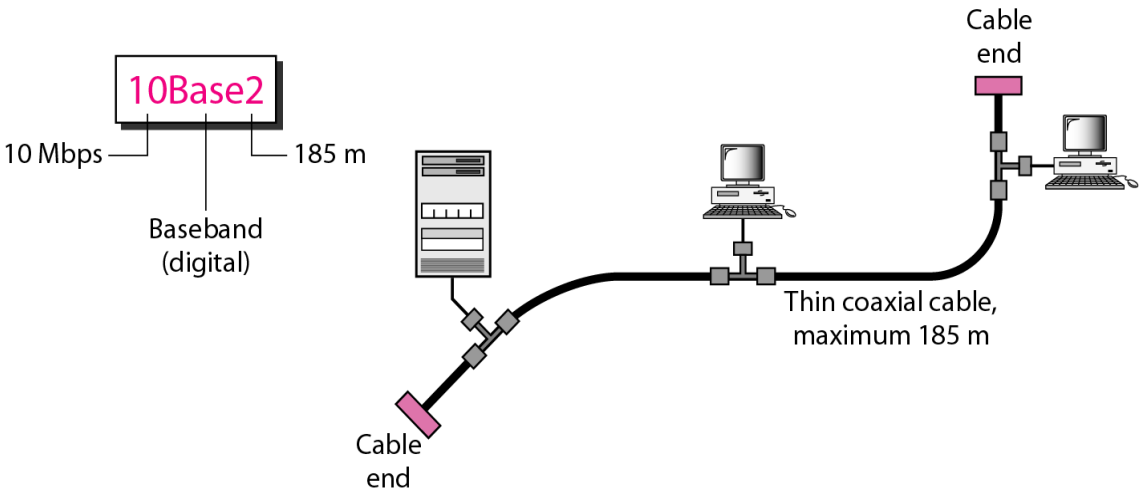
- The maximum length of the coaxial cable must not exceed 500 m, otherwise, there is excessive degradation of the signal. If a length of more than 500 m is needed, up to five segments, each a maximum of 500-meter.

10Base2: Thin Ethernet

The second implementation is called 10Base2, thin Ethernet, or Cheapernet.

- 10Base2 also uses a bus topology, but the cable is much thinner and more flexible. The cable can be bent to pass very close to the stations.

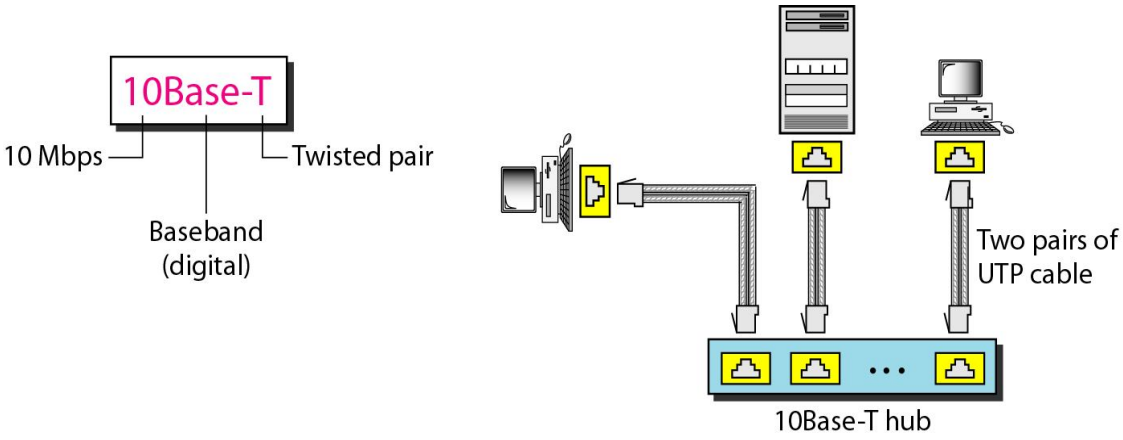
- In this case, the transceiver is normally part of the network interface card (NIC), which is installed inside the station. Figure shows the schematic diagram of a 10Base2 implementation.



10Base-T: Twisted-Pair Ethernet

The third implementation is called 10Base-T or twisted-pair Ethernet.

- 10Base-T uses a physical star topology. The stations are connected to a hub via two pairs of twisted cable, as shown in Figure

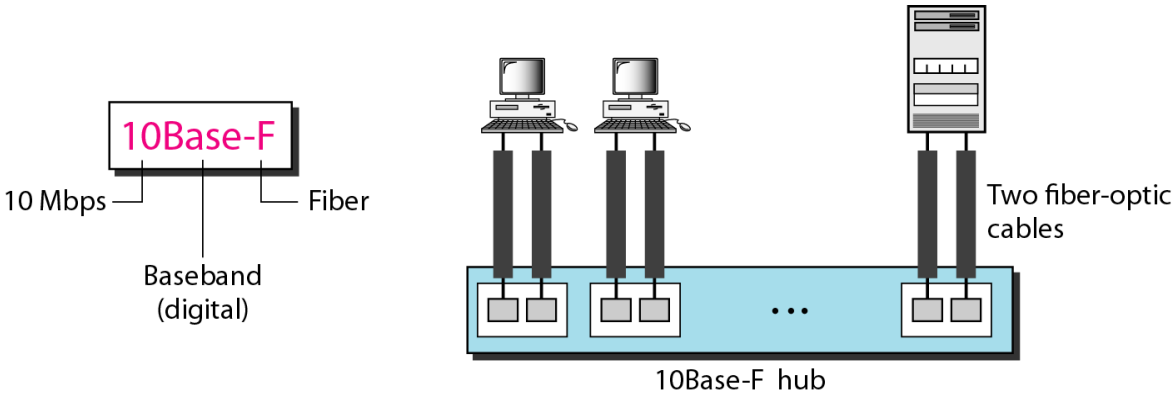


- Note that two pairs of twisted cable create two paths (one for sending and one for receiving) between the station and the hub. Any collision here happens in the hub. Compared to 10Base5 or 10Base2, we can see that the hub actually replaces the coaxial

10Base-F:Fiber Ethernet

Although there are several types of optical fiber l0-Mbps Ethernet, the most common is called 10Base-F.

- 10Base-F uses a star topology to connect stations to a hub. The stations are connected to the hub using two fiber-optic cables, as shown in Figure .



Summary of Standard Ethernet implementations

Characteristics	*10Base5*	*10Base2*	*10Base-T*	*10Base-F*
Media	Thick coaxial cable	Thin coaxial cable	2 UTP	2 Fiber
Maximum length	500 m	185 m	100 m	2000 m
Line encoding	Manchester	Manchester	Manchester	Manchester

IEEE 802.11

IEEE has defined the specifications for a wireless LAN, called IEEE 802.11, which covers the physical and data link layers.

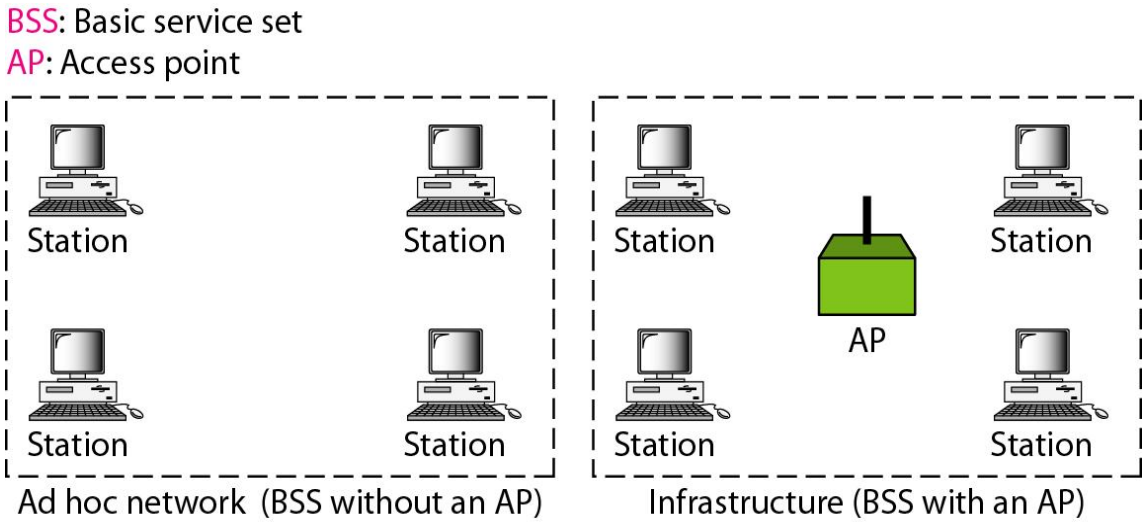
Architecture

The standard defines two kinds of services: the basic service set (BSS) and the extended service set (ESS).

Basic Service Set

IEEE 802.11 defines the basic service set (BSS) as the building block of a wireless LAN.

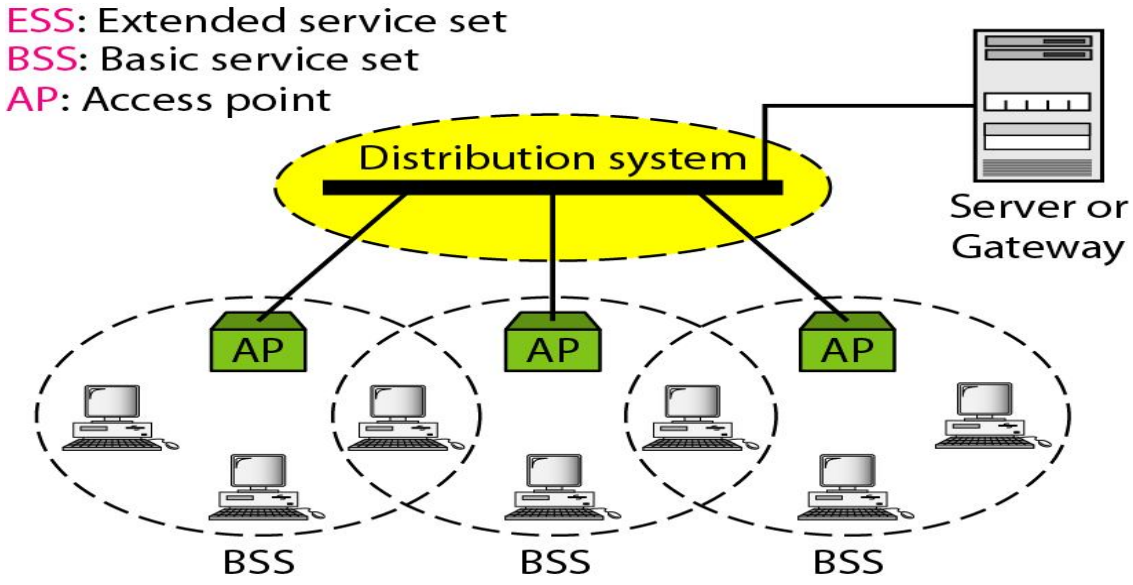
- A basic service set is made of stationary or mobile wireless stations and an optional central base station, known as the access point (AP). Figure shows two sets in this standard.



Extended Service Set

An extended service set (ESS) is made up of two or more BSSs with APs. In this case, the BSSs are connected through a distribution system, which is usually a wired LAN.

- The distribution system connects the APs in the BSSs. IEEE 802.11 does not restrict the distribution system; it can be any IEEE LAN such as an Ethernet.
- Note that the extended service set uses two types of stations: mobile and stationary. The mobile stations are normal stations inside a BSS.
- The stationary stations are AP stations that are part of a wired LAN.
- Figure shows an ESS.



Station Types

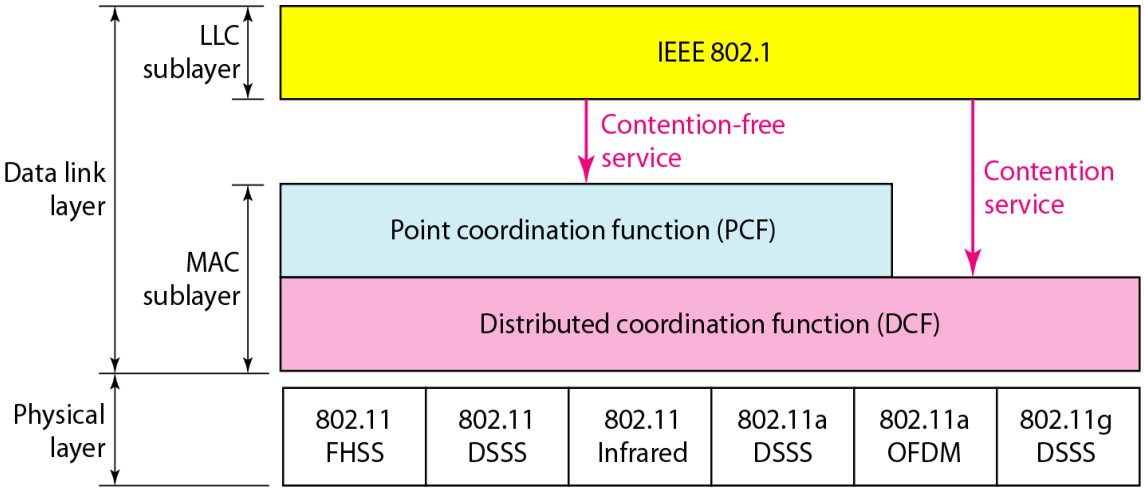
IEEE 802.11 defines three types of stations based on their mobility in a wireless LAN: no-transition, BSS·transition, and ESS-transition mobility.

- A station with no-transition mobility is either stationary (not moving) or moving only inside a BSS.
- A station with BSS-transition mobility can move from one BSS to another, but the movement is confined inside one ESS.
- A station with ESS-transition mobility can move from one ESS to another.
- However, IEEE 802.11 does not guarantee that communication is continuous during the move

MAC Sublayer

IEEE 802.11 defines two MAC sublayers: the distributed coordination function (DCF) and point coordination function (PCF).

- Figure shows the relationship between the two MAC sublayers, the LLC sublayer, and the physical layer.

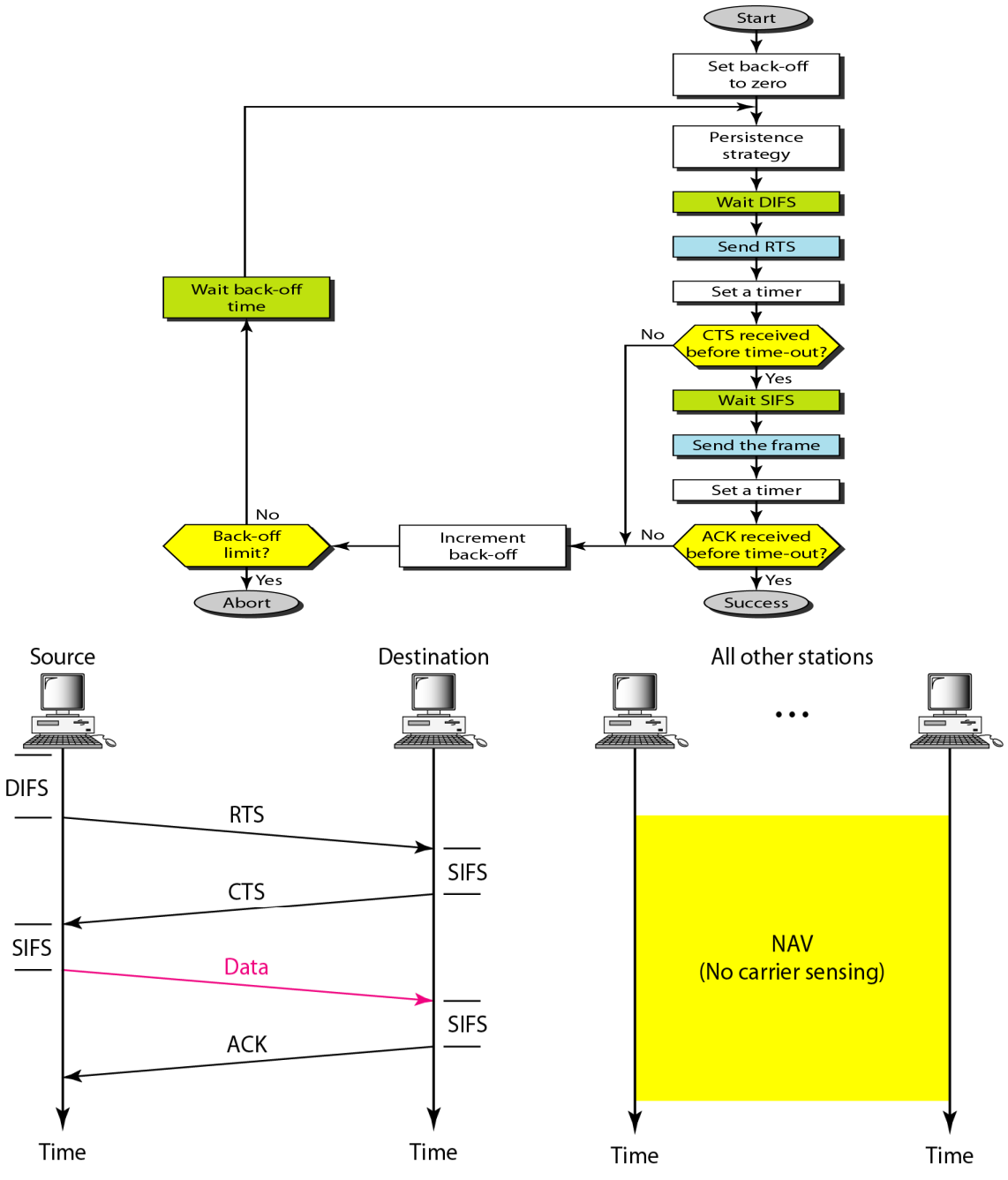


Distributed Coordination Function

One of the two protocols defined by IEEE at the MAC sublayer is called the distributed coordination function (DCF). DCF uses CSMAICA as the access method. Wireless LANs cannot implement CSMAfCD for three reasons:

- For collision detection a station must be able to send data and receive collision signals at the same time. This can mean costly stations and increased bandwidth requirements

- Collision may not be detected because of the hidden station problem. We will discuss this problem later in the chapter.
- The distance between stations can be great. Signal fading could prevent a station at one end from hearing a collision at the other end.



➢ Before sending a frame, the source station senses the medium by checking the energy level at the carrier frequency.

- The channel uses a persistence strategy with back-off until the channel is idle.

- After the station is found to be idle, the station waits for a period of time called the distributed interframe space (DIFS); then the station sends a control frame called the request to send (RTS).

➢ After receiving the RTS and waiting a period of time called the short interframe space (SIFS), the destination station sends a control frame, called the clear to send (CTS), to the source station. This control frame indicates that the destination station is ready to receive data.

➢ The source station sends data after waiting an amount of time equal to SIFS

➢ The destination station, after waiting an amount of time equal to SIFS, sends an acknowledgment to show that the frame has been received. Acknowledgment is needed in this protocol because the station does not have any means to check for the successful arrival of its data at the destination. On the other hand, the lack of collision in CSMA/CD is a kind of indication to the source that data have arrived.

When a station sends an RTS frame, it includes the duration of time that it needs to occupy the channel. The stations that are affected by this transmission create a timer called a network allocation vector (NAV) that shows how much time must pass before these stations are allowed to check the channel for idleness. Each time a station accesses the system and sends an RTS frame, other stations start their NAV.

In other words, each station, before sensing the physical medium to see if it is idle, first checks its NAV to see if it has expired. Figure shows the idea of NA

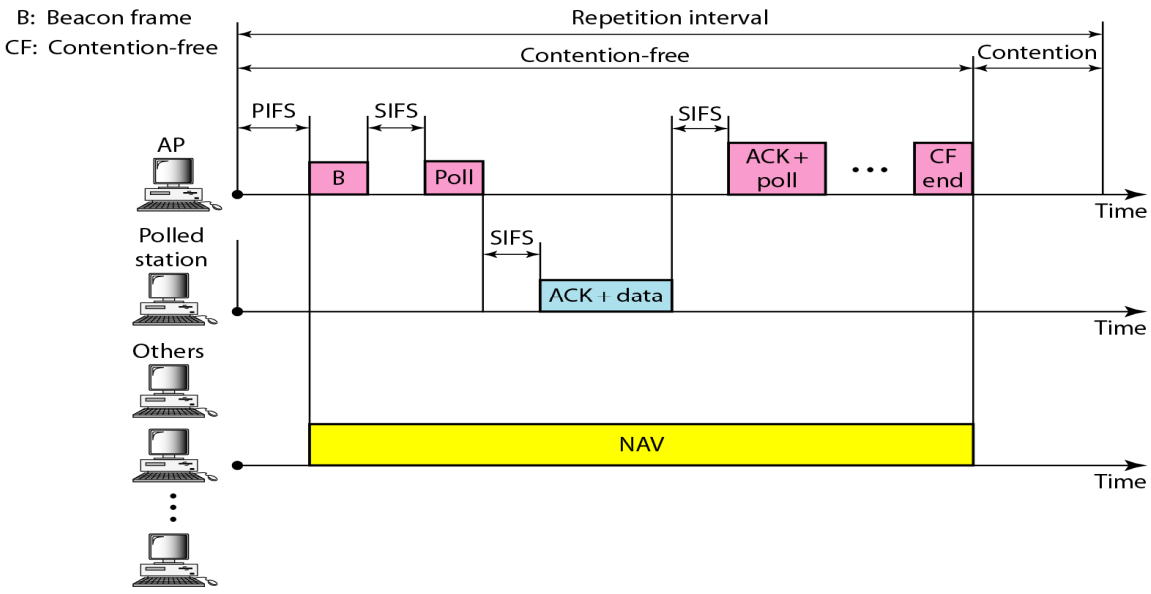
Point Coordination Function (PCP)

The point coordination function (PCF) is an optional access method that can be implemented in an infrastructure network (not in an ad hoc network). It is implemented on top of the DCF and is used mostly for time-sensitive transmission.

PCF has a centralized, contention-free polling access method. The AP performs polling for stations that are capable of being polled. The stations are polled one after another, sending any data they have to the AP.

Due to the priority of PCF over DCF, stations that only use DCF may not gain access to the medium. To prevent this, a repetition interval has been designed to cover both contention-free (PCF) and contention-based (DCF) traffic.

The repetition interval, which is repeated continuously, starts with a special control frame, called a beacon frame. When the stations hear the beacon frame, they start their NAV for the duration of the contention-free period of the repetition interval. Figure shows an example of a repetition interval.



Frame Format

The MAC layer frame consists of nine fields, as shown in Figure

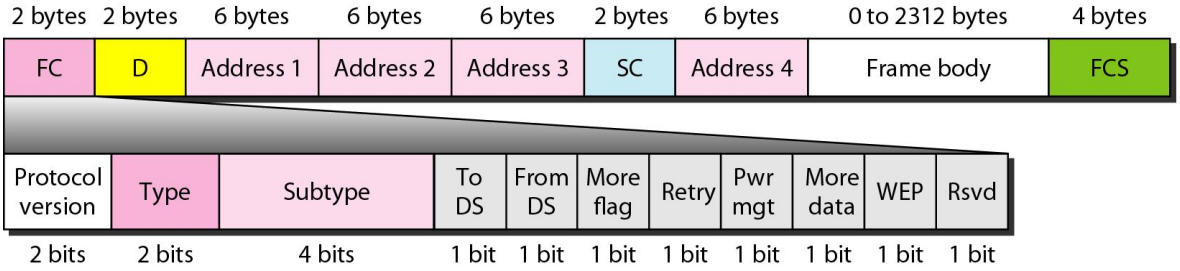


Table 14.1 *Subfields in FC field*

Field	*Explanation*
Version	Current version is 0
Type	Type of information: management (00), control (01), or data (10)
Subtype	Subtype of each type (see Table 14.2)
To DS	Defined later
From DS	Defined later
More flag	When set to 1, means more fragments
Retry	When set to 1, means retransmitted frame
Pwr mgt	When set to 1, means station is in power management mode
More data	When set to 1, means station has more data to send
WEP	Wired equivalent privacy (encryption implemented)
Rsvd	Reserved

14.29

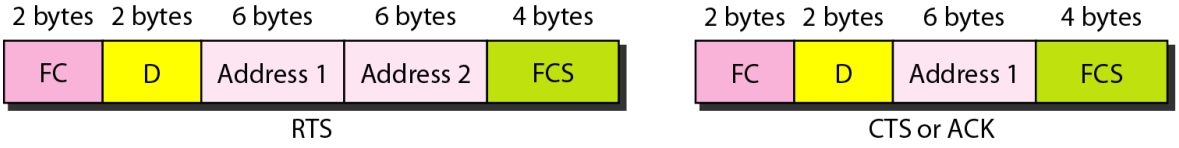
- **Frame control (FC):** The FC field is 2 bytes long and defines the type of frame and some control information. Table 14.1 describes the subfields.
- **D:**In all frame types except one, this field defines the duration of the transmission that is used to set the value of NAY. In one control frame, this field defines the ID of the frame
- **Addresses:** There are four address fields, each 6 bytes long. The meaning of each address field depends on the value of the To DS and From DS subfields and will be discussed later.
- **Sequence control:** This field defines the sequence number of the frame to be used in flow control.
- **Frame body:** This field, which can be between 0 and 2312 bytes, contains information based on the type and the subtype defined in the FC field.
- **FCS:** The FCS field is 4 bytes long and contains a CRC-32 error detection sequence.

Frame Types

A wireless LAN defined by IEEE 802.11 has three categories of frames: management frames, control frames, and data frames.

Management Frames: Management frames are used for the initial communication between stations and access points.

Control Frames: Control frames are used for accessing the channel and acknowledging frames. Figure shows the format.



For control frames the value of the type field is 0 I; the values of the subtype fields for frames we have discussed are shown in Table

Values of subfields in control frames

Subtype	*Meaning*
1011	Request to send (RTS)
1100	Clear to send (CTS)
1101	Acknowledgment (ACK)

BRIDGES

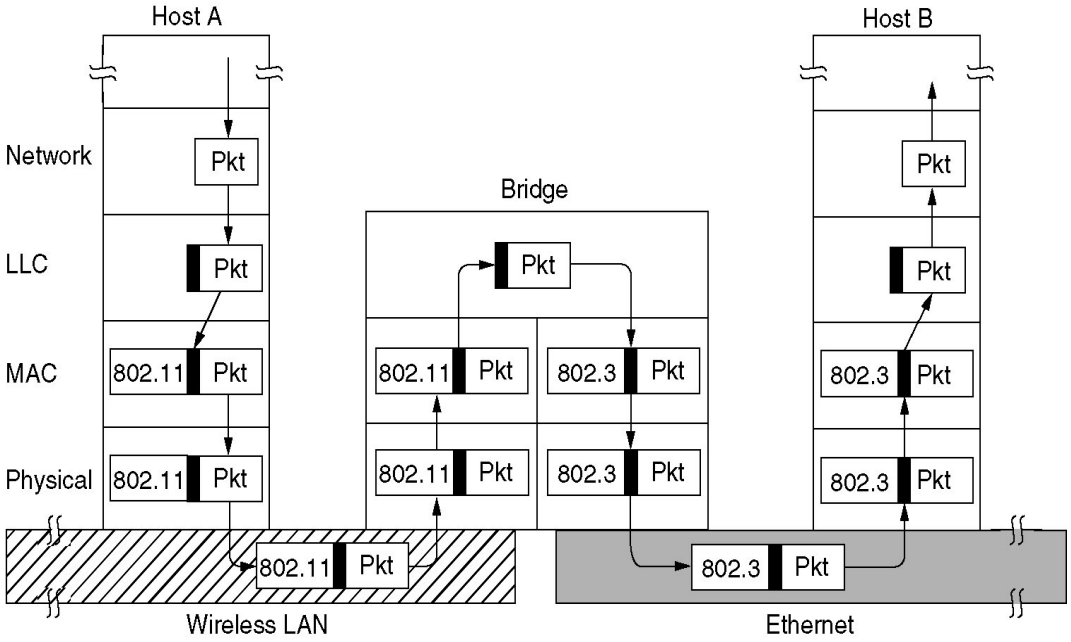
Many organizations have multiple LANs and wish to connect them. LANs can be connected by devices called bridges, We will mention six reasons why a single organization may end up with multiple LANs.

- First, many university and corporate departments have their own LANs, primarily to connect their own personal computers, workstations, and servers. Since the goals of the various departments differ, different departments choose different LANs, without regard to what other departments are doing. Sooner or later, there is a need for interaction, so bridges are needed. In this example, multiple LANs came into existence due to the autonomy of their owners.
- Second, the organization may be geographically spread over several buildings separated by considerable distances. It may be cheaper to have separate LANs in each building and connect them with bridges and laser links than to run a single cable over the entire site
- Third, it may be necessary to split what is logically a single LAN into separate LANs to accommodate the load. At many universities, for example, thousands of workstations are available for student and faculty computing.
- Fourth, in some situations, a single LAN would be adequate in terms of the load, but the physical distance between the most distant machines is too great (e.g., more than 2.5 km for Ethernet). Even if laying the cable is easy to do, the network would not work due to the excessively long round-trip delay. The only solution is to partition the LAN and install bridges between the segments. Using bridges, the total physical distance covered can be increased.
- Fifth, there is the matter of reliability. On a single LAN, a defective node that keeps outputting a continuous stream of garbage can cripple the LAN. Bridges can be inserted at critical places, like fire doors in a building, to prevent a single node that has gone berserk from bringing down the entire system. Unlike a repeater, which just copies whatever it sees, a bridge can be programmed to exercise some discretion about what it forwards and what it does not forward.
- Sixth, and last, bridges can contribute to the organization's security. Most LAN interfaces have a promiscuous mode, in which all frames are given to the computer, not just those addressed to it.

BRIDGES FROM 802.x TO 802.y

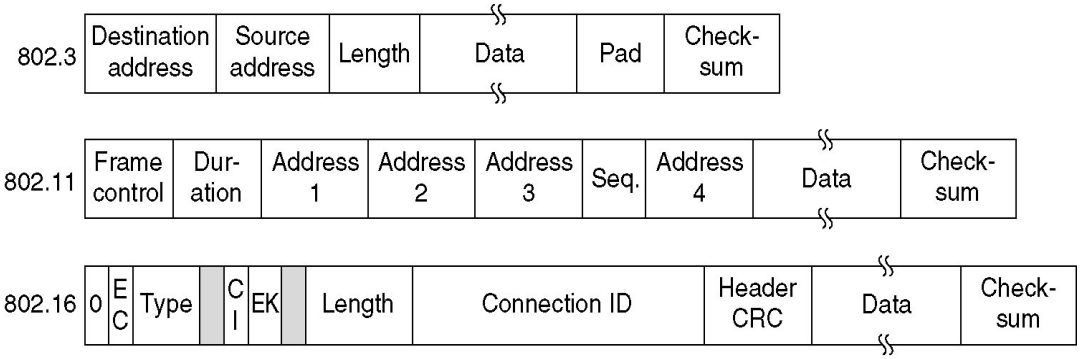
Figure illustrates the operation of a simple two-port bridge. Host A on a wireless(802.11) LAN has a packet to send to a fixed host, B, on an (802.3) Ethernet to which the wireless LAN is connected. The packet descends into the LLC sublayer and acquires an LLC header(Shown in black in the figure). Then it passes into the MAC sublayer and an 802.11 header is prepended to it. This unit goes out over the air and is picked up by the base station, which sees that it needs to go to the fixed Ethernet. When it hits the bridge connecting the 802.11 network to the 802.3 network;

it starts in the physical layer and works its way upward. In the MAC sublayer in the bridge, the 802.11 header is stripped off.The bare packet is then handed off to the LLC sublayer in the bridge.



Moving a frame from one LAN to another is easy. Such is not the case. Some difficulties are there when tryinmg to build a bridge between the various 802 LANs. Some of the problems are:

- Each LAN uses a different frame format. Unlike the differences between the Ethernet, token bus and token ring, which were due to history and big corporate egos, here the differences are to some extent legitimate. For example, the Duration field in 802.11 is there due to the MACAW protocol and makes no sense in the Ethernet. As a result, any copying between different LANs require reformatting, which takes CPU time, requires a new checksum calculation



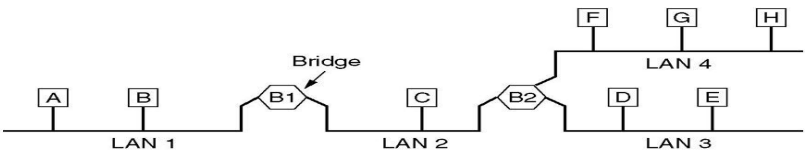
- A second problem is that interconnected LANs do not necessarily run at the same data rate. When forwarding a long run of back-to-back frames from a fast LAN to a slower one, the bridge will not be able to get rid of the frames as fast as they come in.

- A third problem, and the potentially the most serious of all, is that different 802 LANs have different maximum frame lengths.
- Another point is security. 802.11 support encryption in the data link layer. Ethernet does not. This means that the various encryption services available to the wireless networks are lost when traffic passes over an Ethernet.
- Another point is quality of service. 802.11 uses PCF mode. Ethernet has no concept of quality of service.

LOCAL INTERNETWORKING:

- In large organizations with many LANs, just interconnecting them all raises a variety of issues, even if they are all just Ethernet. Ideally, it should be possible to go out and buy bridges designed to the IEEE standard, plug the connectors into the bridges, and everything should work perfectly, instantly.
- There should be no hardware changes required, no software changes required, no setting of address switches, no downloading of routing tables or parameters, nothing. Just plug in the cables and walk away.
- Furthermore, the operation of the existing LANs should not be affected by the bridges at all. In other words, the bridges should be completely transparent (invisible to all the hardware and software). Surprisingly enough, this is actually possible.
- Let us now take a look at how this magic is accomplished. In its simplest form, a transparent bridge operates in promiscuous mode, accepting every frame transmitted on all the LANs to which it is attached. As an example, consider the configuration of Fig.. Bridge B1 is connected to LANs 1 and 2, and bridge B2 is connected to LANs 2, 3, and 4.
- A frame arriving at bridge B1 on LAN 1 destined for A can be discarded immediately, because it is already on the correct LAN, but a frame arriving on LAN 1 for C or F must be forwarded.

Local Internetworking



A configuration with four LANs and two bridges.

- When a frame arrives, a bridge must decide whether to discard or forward it, and if the latter, on which LAN to put the frame. This decision is made by looking up the destination address in a big (hash) table inside the bridge. The table can list each possible destination and tell which output line (LAN) it belongs on.
- For example, B2's table would list A as belonging to LAN 2, since all B2 has to know is which LAN to put frames for A on. That, in fact, more forwarding happens later is not of interest to it.
- When the bridges are first plugged in, all the hash tables are empty. None of the bridges know where any of the destinations are, so they use a flooding algorithm: every incoming frame for an unknown destination is output on all the LANs to which the bridge is connected except the one it arrived on.
- As time goes on, the bridges learn where destinations are, as described below. Once a destination is known, frames destined for it are put on only the proper LAN and are not flooded.
- The algorithm used by the transparent bridges is backward learning.As mentioned above, the bridges operate in promiscuous mode, so they see every frame sent on any of their LANs.
- By looking at the source address, they can tell which machine is accessible on which LAN. For example, if bridge B1 in figure sees a frame on LAN 2 coming from C, it knows that C must be reachable via LAN 2, so it makes an entry in its hash table noting that frames going to C should use LAN 2.
- Any subsequent frame addressed to C coming in on LAN 1 will be forwarded, but a frame for C coming in on LAN 2 will be discarded.

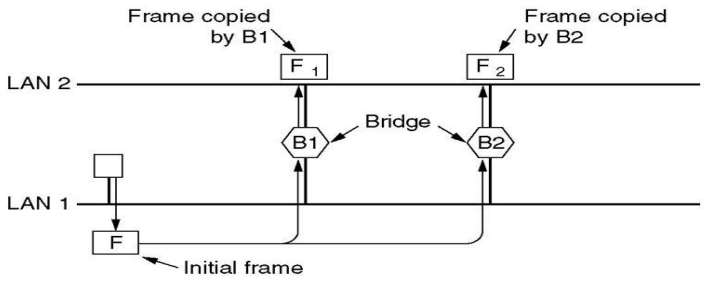
SPANNING TREE BRIDGES

- To increase reliability, some sites use two or more bridges in parallel between pairs of LANs, as shown in Fig.. This arrangement, however, also introduces some additional problems because it creates loops in the topology.

- A simple example of these problems can be seen by observing how a frame, F, with unknown destination is handled in Fig. Each bridge, following the normal rules for handling unknown destinations, uses flooding, which in this example just means copying it to LAN 2.

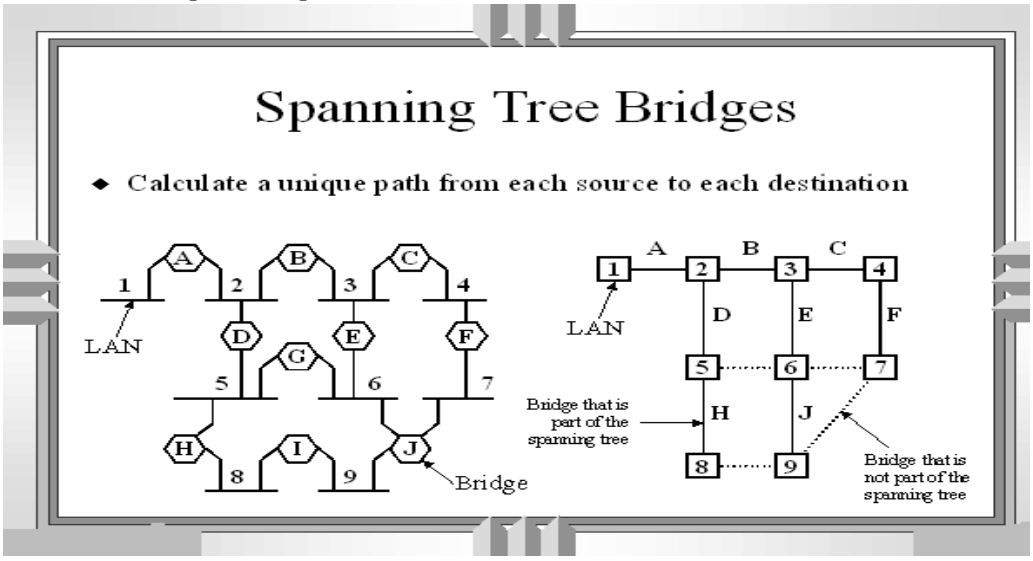
- Shortly thereafter, bridge 1 sees F2, a frame with an unknown destination, which it copies to LAN 1, generating F3 (not shown). Similarly, bridge 2 copies F1 to LAN 1 generating F4 (also not shown). Bridge 1 now forwards F4 and bridge 2 copies F3. This cycle goes on forever.

Spanning Tree Bridges



Two parallel transparent bridges.

- The solution to this difficulty is for the bridges to communicate with each other and overlay the actual topology with a spanning tree that reaches every LAN. In effect, some potential connections between LANs are ignored in the interest of constructing a fictitious loop-free topology.
- For example, in fig. (a) we see nine LANs interconnected by ten bridges. This configuration can be abstracted into a graph with the LANs as the nodes. An arc connects any two LANs that are connected by a bridge.
- The graph can be reduced to a spanning tree by dropping the arcs shown as dotted lines in fig. (b). Using this spanning tree, there is exactly one path from every LAN to every other LAN.
- Once the bridges have agreed on the spanning tree, all forwarding between LANs follows the spanning tree. Since there is a unique path from each source to each destination, loops are impossible.



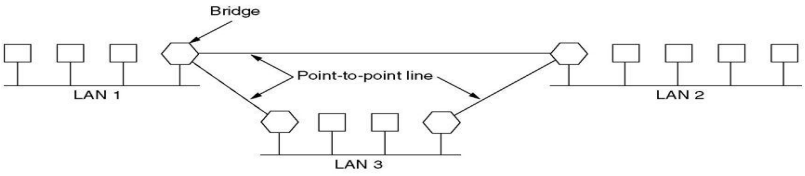
- To build the spanning tree, first the bridges have to choose one bridge to be the root of the tree. They make this choice by having each one broadcast its serial number, installed by the manufacturer and guaranteed to be unique worldwide. The bridge with the lowest serial number becomes the root. Next, a tree of shortest paths from the root to every bridge and LAN is constructed. This tree is the spanning tree. If a bridge or LAN fails, a new one is computed.

REMOTE BRIDGES

A common use of bridges is to connect two (or more) distant LANs. For example, a company might have plants in several cities, each with its own LAN.

- Ideally, all the LANs should be interconnected, so the complete system acts like one large LAN. This goal can be achieved by putting a bridge on each LAN and connecting the bridges pairwise with point-to-point lines (e.g., lines leased from a telephone company).
- A simple system, with three LANs, is illustrated in Fig. The usual routing algorithms apply here. The simplest way to see this is to regard the three point-to-point lines as hostless LANs. Then we have a normal system of six LANS interconnected by four bridges.

Remote Bridges

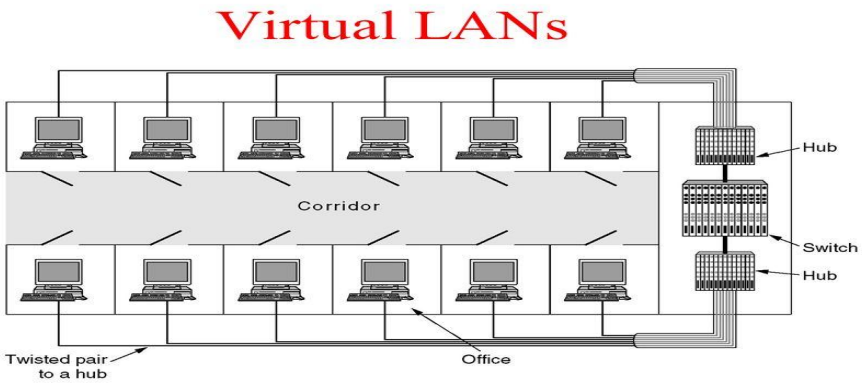


Remote bridges can be used to interconnect distant LANs.

- Various protocols can be used on the point-to-point lines. One possibility is to choose some standard point-to-point data link protocol such as PPP, putting complete MAC frames in the payload field.
- This strategy works best if all the LANs are identical, and the only problem is getting frames to the correct LAN. Another option is to strip off the MAC header and trailer at the source bridge and put what is left in the payload field of the point-to-point protocol.
- A new MAC header and trailer can then be generated at the destination bridge. A disadvantage of this approach is that the checksum that arrives at the destination host is not the one computed by the source host, so errors caused by bad bits in a bridge's memory may not be detected.

VIRTUAL LANS

- In the early days of local area networking, thick yellow cables snaked through the cable ducts of many office buildings. Every computer they passed was plugged in. Often there were many cables, which were connected to a central backbone (as in Fig. 4-39) or to a central hub. No thought was given to which computer belonged on which LAN. All the people in adjacent offices were put on the same LAN whether they belonged together or not. Geography trumped logic. 247 With the advent of 10Base-T and hubs in the 1990s, all that changed.
- Buildings were rewired (at considerable expense) to rip out all the yellow garden hoses and install twisted pairs from every office to central wiring closets at the end of each corridor or in a central machine room, as illustrated in fig.
- If the Vice President in Charge of Wiring was a visionary, category 5 twisted pairs were installed; if he was a bean counter, the existing (category 3) telephone wiring was used (only to be replaced a few years later when fast Ethernet emerged).



A building with centralized wiring using hubs and a switch.

- Does it matter who is on which LAN? After all, in virtually all organizations, all the LANs are interconnected. In short, yes, it often matters. Network administrators like to group users on LANs to reflect the organizational structure rather than the physical layout of the building for a variety of reasons.
- One issue is security. Any network interface can be put in promiscuous mode, copying all the traffic that comes down the pipe. Many departments, such as research, patents, and accounting, have information that they do not want passed outside their department.
- In such a situation, putting all the people in a department on a single LAN and not letting any of that traffic off the LAN makes sense. Management does not like hearing that such an arrangement is impossible unless all the people in each department are located in adjacent offices with no interlopers.

- A second issue is load. Some LANs are more heavily used than others and it may be desirable to separate them at times. For example, if the folks in research are running all kinds of nifty experiments that sometimes get out of hand and saturate their LAN, the folks in accounting may not be enthusiastic about donating some of their capacity to help out.

- A third issue is broadcasting. Most LANs support broadcasting, and many upper-layer protocols use this feature extensively. For example, when a user wants to send a packet to an IP address x, how does it know which MAC address to put in the frame? but briefly summarized, the answer is that it broadcasts a frame containing the question: Who owns IP address x? Then it waits for an answer. And there are 248 many more examples of where broadcasting is used.

- As more and more LANs get interconnected, the number of broadcasts passing each machine tends to increase linearly with the number of machines. Related to broadcasts is the problem that once in a while a network interface will break down and begin generating an endless stream of broadcast frames.

- The result of this broadcast storm is that (1) the entire LAN capacity is occupied by these frames, and (2) all the machines on all the interconnected LANs are crippled just processing and discarding all the frames being broadcast. At first it might appear that broadcast storms could be limited in scope by separating the LANs with bridges or switches, but if the goal is to achieve transparency (i.e., a machine can be moved to a different LAN across the bridge without anyone noticing it), then bridges have to forward broadcast frames. Having seen why companies might want multiple LANs with restricted scope, let us get back to the problem of decoupling the logical topology from the physical topology.

- Suppose that a user gets shifted within the company from one department to another without changing offices or changes offices without changing departments. With hubbed wiring, moving the user to the correct LAN means having the network administrator walk down to the wiring closet and pull the connector for the user's machine from one hub and put it into a new hub.

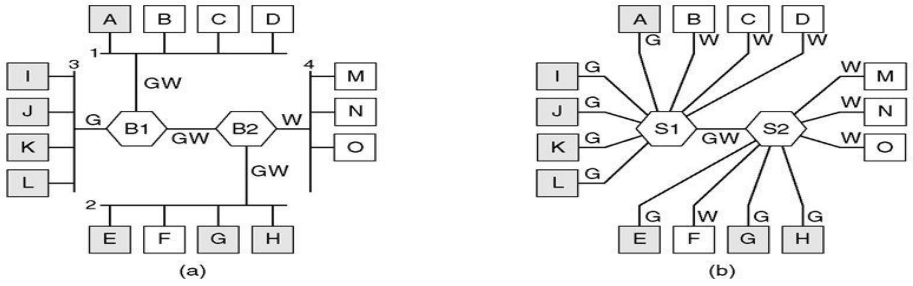
- In many companies, organizational changes occur all the time, meaning that system administrators spend a lot of time pulling out plugs and pushing them back in somewhere else. Also, in some cases, the change cannot be made at all because the twisted pair from the user's machine is too far from the correct hub (e.g., in the wrong building).

- In response to user requests for more flexibility, network vendors began working on a way to rewire buildings entirely in software. The resulting concept is called a VLAN (Virtual LAN) and has even been standardized by the 802 committee. It is now being deployed in many organizations. Let us now take a look at it. For additional information about VLANs, see (Breyer and Riley, 1999; and Seifert, 2000).

VLANs are based on specially-designed VLAN-aware switches, although they may also have some hubs on the periphery, as in fig. To set up a VLAN-based network, the network administrator decides how many VLANs there will be, which computers will be on which VLAN, and what the VLANs will be called.

- Often the VLANs are (informally) named by colors, since it is then possible to print color diagrams showing the physical layout of the machines, with the members of the red LAN in red, members of the green LAN in green, and so on. In this way, both the physical and logical layouts are visible in a single view. As an example, consider the four LANs of fig. (a), in which eight of the machines belong to the G (gray) VLAN and seven of them belong to the W (white) VLAN. The four physical LANs are connected by two bridges, B1 and B2. If centralized twisted pair wiring is used, there might also be four hubs (not shown), but logically a multidrop cable and a hub are the same thing. Drawing it this way just makes the figure a little less cluttered. Also, the term "bridge" tends to be used nowadays mostly when there are multiple machines on each port, as in this figure, but otherwise, "bridge" and "switch" are essentially interchangeable. fig. (b) shows the same machines and same VLANs using switches with a single computer on each port.

Virtual LANs (2)



(a) Four physical LANs organized into two VLANs, gray and white, by two bridges. (b) The same 15 machines organized into two VLANs by switches.

UNIT-VI

Assignment-Cum-Tutorial Questions

SECTION-A

Objective Questions

1. Ethernet frame consists of [ ]
a) MAC address b) IP address c) both (a) and (b) d) either a or b

2. What is start frame delimeter (SFD) in ethernet frame? [ ]
a) 10101010 b) 10101011
c) 00000000 d) 11111111

3. High speed ethernet works on [ ]
a) coaxial cable b) twisted pair cable
c) optical fiber d) none of the mentioned

4.Field of Media Access Control frame that is used to alternate 1s and 0s is called [ ]

a) SFD b) preamble

c) source address d) destination address

5. Protocol Data Unit (PDU) is similar to [ ]

a) LLC b)HDLC c) MAC d)DSAP

6. In 10Base2, cable is [ ]

a) Thick b)Thin c) Small d)Large

7. Maximum length of optical fiber in 10BaseF is [ ]

a) 100 meter b)185 meters c)500 meter d)2000 meter

8. In 10Base5, maximum length of coaxial cable is [ ]

a) 10 meter b)250 meter c)500 meter d)5000 meter

9. 10BaseT is used to connect stations by using [ ]

a) Star Topology b)Bus Topology

c) Ring Topology d)Mesh Topology

10. The advantage of using Manchester format of coding is [ ]
a) Power saving b) Polarity sense at the receiver
c)Noise immunity d) None of the above

11. Bridge must therefore discard any frames too large for its [ ]

a) Data b)Connection c)Bridge d)System

12. Basic service sets are connected through a distribution system in

a) Basic service set b)Extended service set [ ]
b) Cellular Set d)Distributed set

13.Distributed Coordination Function (DCF) is used mechanism of

a) CSMA/CD b)CSMA/CA [ ]

c)TDMA d)FDMA

14. After receiving the RTS and waiting a period of time called the…… [ ]

a) DIFS b)SIFS c)NAV d)ACK

15. Management frames, control frames, and data frames are used by

a) Wireless LANs b)Wireless MANs [ ]
b) Wireless WANs d)None

16. VLANs provides [ ]

a) time-efficiency and security b)reduce network traffic
b) both a and b d)None

17. VLANs create [ ]

a) broadcast domains b)limited domains
b) metropolitan area domains d)none

18. A _____ is a connecting device that operates in the physical and data link layers of the Internet model. [ ]

a) Repeater b)Bridge

c)Router d)None of the above

19. In transparent bridges, redundancy of bridges can create loops in system which is very [ ]

a) Easy b)Undesirable

c) Difficult d)Long

20. Bridge can operate on both layers those are [ ]

a) physical and data link layer
b) physical and sessional layer
c) application and data link layer
d) physical and presentation layer

21. A transparent bridge's duties include [ ]

a) filtering frames b)forwarding
b) blocking d)All of them

SECTION-B

SUBJECTIVE QUESTIONS

1. What is slot time in MAC sublayer?
2. What are the advantages of spanning tree bridges?
3. Explain 802.11 frame format with a neat sketch
4. Explain MAC sub layer in Standard Ethernet
5. Discuss about categories of standard Ethernet
6. Explain MAC sublayer in 802.11
7. How IEEE standards contribute to physical and data link layers?
8. Explain the functioning of bridge from 802.x to 802.y.
9. What is spanning tree? Describe the process of spanning tree bridge to find the spanning tree with an example.
10. Sketch the Manchester encoding and binary encoding for the bit stream: 0001110101(Assume the line is initially in low state.)

SECTION-C
QUESTIONS AT THE LEVEL OF GATE

1. Which of the following is NOT true with respect to a transparent bridge and a router? [ ]

a. Both bridge and router selectively forward data packets

b. A bridge uses IP addresses while a router uses MAC addresses

c. A bridge builds up its routing table by inspecting incoming packets

d. A router can connect between a LAN and a WAN **[GATE 2004]**

2. In a network of LANs connected by bridges, packets are sent from one LAN to another through intermediate bridges. Since more than one path may exist between two LANs, packets may have to be routed through multiple bridges. Why is the spanning tree algorithm used for bridge-routing? [ ]

a. For shortest path routing between LANs

b. For avoiding loops in the routing paths

c. For fault tolerance

d. For minimizing collisions **[GATE 2005]**