Data Communication & Computer Network

NOTE

Although every effort has been made to avoid errors and omissions, there is still a possibility that some mistakes may be missed due to invisibility.

This E - book is issued with the Understanding that the author is not responsible in any way for any errors/omissions.

BCA 303: Data Communication & Computer Networks

Question Paper pattern for Main University Examination **Max Marks: 100**

Part-1 (very short answer) consists 10 questions of two marks each with two questions from each unit. Maximum limit for each question is up to 40 words.

Part-II (short answer) consists 5 questions of four marks each with one question from each unit. Maximum limit for each question is up to 80 words.

Part-III (Long answer) consists 5 questions of twelve marks each with one question from each unit with internal choice.

UNIT-I

Introduction: Network definition, Network topologies, Types of Network, Layered network architecture, Categories of Network, protocol, Standards and interface.

Network Models: OSI reference model, OSI model architecture and functions of layers.TCP/IP protocol suite.

UNIT- II

Data Communication Fundamentals and Techniques: Analog and digital signal. Data-rate limits, Digital to digital line encoding schemes, Pulse code modulation, Digital to analog modulation- ASK, FSK, PSK, QAM, multiplexing techniques- FDM, TDM, WDM, transmission modes

Transmission Media : Guided media (Twisted Pair Cable, Coaxial Cable & Fiber-Optic Cable) and Unguided media: Radio wave, Infrared, Microwave Communication. Satellite, Geosynchronous Satellites Communication.

UNIT-III

Networks Switching Techniques: Circuit switching; Packet switching- Connectionless datagram switching, Connection-oriented virtual circuit switching.

Data Link Layer Functions and Protocol: Error detection and error correction techniques, Data-link control-framing and flow control, Error recovery protocols- Stop and wait ARQ, Go-back-n ARQ, Selective repeat ARQ, Point to Point Protocol on Internet.

UNIT-IV

Access mechanisms Multiple Access Protocol and Networks: ALOHA,CSMA/CD protocols, Ethernet LANS, connecting LAN and back-bone networks- Repeaters. Hubs, Switches, Bridges, Router and Gateways.

Networks Layer Functions and Protocols: Routing, Routing algorithms. Network layer protocol of Internet- IP protocol. Internet control protocols.

UNIT-V

Transport Layer Functions and Protocols: Transport services, Berkeley socket interface overview, Transport layer protocol of Internet- UDP and TCP. Overview of Application layer protocol, DNS protocol, WWW & HTTP protocols.

Unit I: Introduction to Networks

Introduction: Network definition, Network topologies, Types of Network, Layered network architecture, Categories of Network, protocol, Standards and interface.

Network Models: OSI reference model, OSI model architecture and functions of layers.TCP/IP protocol suite.

Introduction to Networking

A **network** is a system where multiple devices, such as computers, servers, routers, and other network devices, are connected to share resources, exchange information, or communicate. Networks allow the transmission of data and facilitate various services like file sharing, email, and web browsing.

Key components of a network include:

- **Nodes**: Devices such as computers, printers, and routers that are part of the network.
- **Links**: Physical or wireless connections that interconnect devices.
- **Protocols**: The set of rules that dictate how devices communicate and transfer data within a network.

Networking enables efficient resource sharing and communication over local or global distances, with numerous applications across different sectors, such as business, education, and entertainment.

Network Topologies

Network topology refers to the arrangement of devices and how they are interconnected. Different topologies offer different benefits and are suited to varying needs. Common types of network topologies include:

- **Bus Topology**: In this layout, all devices share a common central communication medium (the bus), and data is transmitted in one direction. It is simple but may be prone to traffic congestion and failure if the central bus is damaged.
- **Ring Topology**: Each device is connected to two other devices, forming a closed loop. Data travels in one direction around the ring. While efficient in terms of managing data flow, failure in one device or connection can disrupt the entire network.
- **Star Topology**: In a star network, devices are connected to a central hub or switch, which directs data traffic between them. This setup is easy to manage and expand, but the central hub is a critical point of failure.
- **Mesh Topology**: Every device in the network is directly connected to every other device. This provides high redundancy and resilience but requires many connections, making it expensive and complex.
- **Tree Topology**: A hybrid of star and bus topologies, this arrangement uses a central backbone to which various star-configured branches are attached. It offers scalability but can become difficult to manage as the network grows.
- **Hybrid Topology**: A combination of two or more different types of topologies, designed to leverage the benefits of each. For instance, a network could have star-topology branches connected to a bus topology backbone.

Each topology comes with its pros and cons, and the choice depends on factors like scalability, cost, fault tolerance, and ease of management.

Types of Networks

Networks can be classified based on their size, geographic reach, and purpose. Some of the common types include:

- **LAN (Local Area Network)**: A network limited to a small geographic area, such as a single building, office, or campus. It typically provides high-speed connections for devices like computers and printers. Examples of LANs include office networks and home Wi-Fi networks.
- **WAN (Wide Area Network)**: A network that spans a large geographical area, often connecting multiple LANs. The internet is the largest example of a WAN. WANs are typically more expensive to set up and maintain, but they provide connectivity over vast distances.
- **MAN (Metropolitan Area Network)**: A network that covers a city or a large campus. It is larger than a LAN but smaller than a WAN. MANs are often used to interconnect different buildings or offices in a metropolitan area, providing high-speed data transfer within that region.
- **PAN (Personal Area Network)**: This type of network is intended for personal devices, typically within a range of up to 10 meters. Examples of PANs include Bluetooth connections for devices like smartphones, laptops, or wireless keyboards.
- **VPN (Virtual Private Network)**: A VPN is used to provide secure remote access to a network over the internet. It uses encryption and tunneling protocols to ensure that data transferred between a user and the network remains private and secure.

Layered Network Architecture

A **layered network architecture** simplifies complex network tasks by breaking them down into distinct layers, each of which has a specific function in data communication. This approach enables easier troubleshooting, modularity, and standardization.

The **OSI Model** and **TCP/IP model** are two key frameworks used in networking to describe the layers of network communication.

Protocols, Standards, and Interfaces

- **Protocol**: A protocol is a set of rules governing how data is transmitted across a network. It ensures devices can communicate effectively and that data is transferred reliably and securely. Common protocols include:
    - **HTTP (Hypertext Transfer Protocol)**: Used for transferring web pages.
    - **FTP (File Transfer Protocol)**: Used for transferring files over a network.
    - **TCP/IP (Transmission Control Protocol/Internet Protocol)**: A suite of protocols that underpins most network communication on the internet.
    - **SMTP (Simple Mail Transfer Protocol)**: A protocol for sending emails.
    - **IP (Internet Protocol)**: A protocol for addressing and routing data packets across the network.
- **Standards**: Standards ensure consistency across devices and software, allowing them to work together. These are set by organizations such as the IEEE and IETF. Examples include:
    - **IEEE 802.3**: Ethernet standard for local area networks.

- **IEEE 802.11**: Wi-Fi standard for wireless networking.

- **Interfaces**: Interfaces are the physical or logical points where devices communicate. Examples include:
  - **Ethernet Ports**: The physical interface through which devices connect via cables.
  - **Wi-Fi**: A wireless interface that allows devices to communicate over the airwaves without needing physical connections.

Network Models:

A network model is a conceptual framework used to understand and describe the functions of a network in terms of layers or components, ensuring interoperability, scalability, and reliability in communication.

OSI Reference Model

The **OSI (Open Systems Interconnection)** Model is a conceptual framework that standardizes the communication functions of a network into seven distinct layers. This model helps to understand and troubleshoot networking issues and simplifies the design and development of network protocols.

1. **Physical Layer**: This is the lowest layer, responsible for transmitting raw data over physical media, such as cables, radio waves, or fiber optics. It defines electrical signals, physical connections, and how data is sent and received.
   - Examples: Ethernet cables, fiber-optic cables, wireless transmission.
2. **Data Link Layer**: This layer ensures reliable data transfer by detecting and correcting errors in the physical layer. It packages bits into frames and handles access to the shared network medium.
   - Examples: Ethernet, Wi-Fi (802.11), MAC (Media Access Control) addressing.
3. **Network Layer**: The network layer is responsible for routing data packets between different devices across networks, ensuring data reaches its destination. It handles logical addressing, routing, and packet forwarding.
   - Examples: IP (Internet Protocol), routers.
4. **Transport Layer**: This layer manages end-to-end communication between devices and ensures reliable delivery of data by segmenting and reassembling data. It provides error correction and flow control.
   - Examples: TCP (Transmission Control Protocol), UDP (User Datagram Protocol).
5. **Session Layer**: The session layer manages sessions or connections between applications. It ensures that communication between applications is continuous and organized.
   - Examples: RPC (Remote Procedure Call), NetBIOS.
6. **Presentation Layer**: The presentation layer is responsible for data translation, encryption, and compression. It ensures that the data sent from the application layer is in a format that the receiving system can understand.
   - Examples: SSL/TLS (for encryption), JPEG (image formatting).
7. **Application Layer**: This is the topmost layer, where network services and applications operate. It directly interacts with end-users, providing them with services like web browsing, file transfer, and email.
   - Examples: HTTP (Hypertext Transfer Protocol), FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol).

TCP/IP Protocol Suite

The **TCP/IP model** is the foundation of most internet and network communications today. It is a more simplified version of the OSI model, typically described with four layers:

1. **Link Layer**: Corresponds to the OSI model's Physical and Data Link layers, responsible for physical transmission and data framing.
    - Examples: Ethernet, Wi-Fi.
2. **Internet Layer**: This layer is responsible for logical addressing, routing, and forwarding data packets across networks. It uses IP addresses to identify devices on the network.
    - Examples: IP (Internet Protocol), ARP (Address Resolution Protocol), ICMP (Internet Control Message Protocol).
3. **Transport Layer**: Similar to the OSI model's Transport layer, the transport layer ensures reliable data transmission between devices, providing error handling, flow control, and data segmentation.
    - Examples: TCP, UDP.
4. **Application Layer**: This layer provides application-level services, such as file transfer, email, and web browsing, much like the top layers of the OSI model.
    - Examples: HTTP, FTP, DNS (Domain Name System).

The TCP/IP protocol suite is designed to support communication over large-scale networks such as the internet, ensuring flexibility, scalability, and interoperability.

Conclusion

Understanding networking concepts, protocols, and models is essential for comprehending how devices communicate over local or wide-area networks. The OSI and TCP/IP models provide structured frameworks for managing network communication, with each layer handling distinct functions. Protocols, standards, and interfaces enable interoperability between different devices and technologies, ensuring efficient and secure data transmission.

Unit II: Data Communication

Data Communication Fundamentals and Techniques: Analog and digital signal. Data-rate limits, Digital to digital line encoding schemes, Pulse code modulation, Digital to analog modulation- ASK, FSK, PSK, QAM, multiplexing techniques- FDM, TDM, WDM, transmission modes

Transmission Media : Guided media (Twisted Pair Cable, Coaxial Cable & Fiber-Optic Cable) and Unguided media: Radio wave, Infrared, Microwave Communication. Satellite, Geosynchronous Satellites Communication.

Data Communication Fundamentals and Techniques

Data communication refers to the process of transmitting data between devices over a communication medium. In modern networking, it plays a critical role in ensuring that data is transferred efficiently and accurately between different systems. This unit delves into the fundamental techniques and principles behind data communication, exploring signal types, encoding schemes, modulation techniques, multiplexing, and transmission media.

Analog and Digital Signals

- **Analog Signals**: Analog signals are continuous signals that represent data in a wave-like form. These signals vary in amplitude, frequency, or phase, depending on the information being transmitted. Analog signals are often used in traditional telecommunications systems and older technologies.
    - **Characteristics**: Analog signals are continuous, which means they can represent a range of values. However, they are more susceptible to noise and degradation over long distances.
    - **Examples**: Telephone calls over traditional landlines, AM/FM radio signals.
- **Digital Signals**: Digital signals represent data in discrete binary form (0s and 1s), making them less prone to noise and more reliable over long distances. These signals are composed of a sequence of pulses that represent binary data.
    - **Characteristics**: Digital signals are less affected by noise and interference compared to analog signals, and they provide a more efficient means of data transmission, especially in modern communication systems.
    - **Examples**: Internet communication, modern telephony, digital television.

Data-Rate Limits

The **data rate** is the speed at which data can be transmitted over a communication channel, typically measured in **bits per second (bps)**. Several factors influence the data rate, including the bandwidth of the transmission medium and the quality of the signal.

- **Bandwidth**: The range of frequencies that a communication channel can carry. A higher bandwidth allows more data to be transmitted simultaneously.
- **Signal Quality**: The signal-to-noise ratio (SNR) is crucial for determining the maximum data rate. Higher SNR results in fewer errors and more reliable data transmission.
- **Transmission Medium**: Different types of cables and media support varying data rates. For example, fiber-optic cables support higher data rates compared to copper cables.

Shannon-HartleTheorem:
This theorem provides a formula to calculate the maximum theoretical data rate C of a communication channel based on its bandwidth B and signal-to-noise ratio S/N:

$$C = B \log_2(1 + \frac{S}{N})$$

Where:

- C is the channel capacity (maximum data rate),
- B is the bandwidth,
- S is the signal power,
- N is the noise power.

Digital to Digital Line Encoding Schemes

Line encoding involves converting digital data into a specific format suitable for transmission over a medium. The goal is to make the transmission more reliable by preventing errors, improving synchronization, and ensuring signal integrity. Common digital-to-digital encoding techniques include:

- **NRZ (Non-Return-to-Zero)**: In this method, a bit is represented by a high voltage for one binary state and a low voltage for the other state. It does not return to zero between bits.
    - **Advantages**: Simple and efficient.
    - **Disadvantages**: Difficulty in synchronization over long transmissions.
- **RZ (Return-to-Zero)**: In this scheme, each bit is represented by a positive voltage for half of the bit period and then returns to zero for the other half.
    - **Advantages**: Better synchronization than NRZ.
    - **Disadvantages**: Higher bandwidth usage.
- **Manchester Encoding**: A method where each bit is represented by a transition at the middle of the bit period. A logical 1 might be represented by a transition from low to high, and a logical 0 by a high-to-low transition.
    - **Advantages**: Ensures synchronization as each bit has a transition.
    - **Disadvantages**: Requires more bandwidth.
- **Differential Manchester Encoding**: Similar to Manchester encoding, but the direction of the transition (low-to-high or high-to-low) indicates the data value.

Pulse Code Modulation (PCM)

Pulse Code Modulation (PCM) is a method used to convert analog signals into digital form. It is widely used in digital audio and telecommunication systems. PCM works by sampling an analog signal at regular intervals and quantizing the amplitude of each sample to the nearest value in a finite set of possible values.

- **Process**:
    - **Sampling**: The analog signal is sampled at a constant rate.

    - **Quantization**: Each sample is approximated to the nearest value within a specific range of digital values.
    - **Encoding**: The quantized values are then encoded as binary numbers.
- **Applications**: PCM is used in digital telephony (e.g., VoIP), audio recording, and digital audio formats like CD and DVD.

Digital to Analog Modulation Techniques

When transmitting digital data over analog channels, **modulation** is used to convert the digital signal into an analog signal that can travel over the physical medium. Common digital-to-analog modulation techniques include:

- **ASK (Amplitude Shift Keying)**: In ASK, the amplitude of the carrier signal is varied according to the binary data. A high amplitude represents one bit, and a low amplitude represents another bit.
    - **Advantages**: Simple to implement.
    - **Disadvantages**: Susceptible to noise and interference.
- **FSK (Frequency Shift Keying)**: FSK modulates the frequency of the carrier signal. Different frequencies represent different binary values (0 or 1).
    - **Advantages**: More robust against noise than ASK.
    - **Disadvantages**: Requires more bandwidth.
- **PSK (Phase Shift Keying)**: PSK modulates the phase of the carrier signal to represent data. A phase shift represents binary data.
    - **Advantages**: Efficient and resistant to noise.
    - **Disadvantages**: Requires accurate phase synchronization.
- **QAM (Quadrature Amplitude Modulation)**: QAM combines both amplitude and phase modulation. Multiple bits are encoded per symbol, enabling high data rates.
    - **Advantages**: High data rates and efficient use of bandwidth.
    - **Disadvantages**: More complex and susceptible to noise.

Multiplexing Techniques

Multiplexing is a technique used to combine multiple data streams into one signal over a shared medium, optimizing the use of available bandwidth. There are various types of multiplexing techniques:

- **FDM (Frequency Division Multiplexing)**: In FDM, the available bandwidth is divided into multiple non-overlapping frequency bands, and each data stream is transmitted at a different frequency.
    - **Applications**: Analog radio and television broadcasting, satellite communication.
- **TDM (Time Division Multiplexing)**: In TDM, the transmission time is divided into time slots. Each data stream is allocated a specific time slot to transmit its data.
    - **Applications**: Digital telephony, satellite communication.

- **WDM (Wavelength Division Multiplexing)**: WDM is similar to FDM, but it is used in optical fiber communication. Different wavelengths (colors) of light carry separate data streams over the same fiber-optic cable.
    - **Applications**: Fiber-optic communication systems.

Transmission Modes

The **transmission mode** refers to the direction of data flow between two devices. It can be classified into the following categories:

- **Simplex Mode**: Data flows in one direction only, from the sender to the receiver, without any return communication.
    - **Example**: Television broadcasts.
- **Half-Duplex Mode**: Data can flow in both directions, but not at the same time. Each device takes turns transmitting and receiving.
    - **Example**: Walkie-talkies.
- **Full-Duplex Mode**: Data flows in both directions simultaneously. Both devices can send and receive data at the same time.
    - **Example**: Telephone conversations.

Transmission Media

Transmission media refers to the physical or wireless medium used to transmit data from one device to another. It can be categorized into **guided media** (wired) and **unguided media** (wireless).

- **Guided Media**: In guided media, data is transmitted over physical cables or fibers.
    - **Twisted Pair Cable**: Made of pairs of copper wires twisted together. It is commonly used for telecommunication and Ethernet networks.
    - **Coaxial Cable**: Consists of a central conductor, an insulating layer, a metallic shield, and an outer jacket. Coaxial cables are used for cable television and broadband internet.
    - **Fiber-Optic Cable**: Uses light signals to transmit data, offering high bandwidth and resistance to noise and interference. Fiber-optic cables are widely used in modern high-speed internet connections.
- **Unguided Media**: Data is transmitted over the air without the need for physical cables.
    - **Radio Waves**: Used for wireless communication, including AM/FM radio and Wi-Fi.
    - **Microwave Communication**: Involves high-frequency radio waves to transmit data over long distances, often used in satellite communication and point-to-point communication.
    - **Infrared**: Infrared signals are used for short-range communication, such as remote controls and some wireless devices.
- **Satellite Communication**: Communication that involves transmitting signals via satellites orbiting Earth. Satellites are used for long-distance communication, including television broadcasting, weather monitoring, and GPS services.

- **Geosynchronous Satellites**: These satellites orbit the Earth at a fixed position relative to the Earth's surface. This allows for continuous communication with specific regions.

Conclusion

This unit covers the essential techniques and methods in data communication, such as analog and digital signals, encoding schemes, modulation techniques, and multiplexing. Transmission media, both guided and unguided, play a key role in ensuring the successful transfer of data across networks. Understanding these concepts is fundamental for the design and operation of modern communication systems.

Unit III: Network Switching & Data Link Layer

Networks Switching Techniques: Circuit switching; Packet switching- Connectionless datagram switching, Connection-oriented virtual circuit switching.

Data Link Layer Functions and Protocol: Error detection and error correction techniques, Data-link control-framing and flow control, Error recovery protocols- Stop and wait ARQ, Go-back-n ARQ, Selective repeat ARQ, Point to Point Protocol on Internet.

Networks Switching Techniques

Network switching refers to the methods used to manage the routing of data in a network. Switching is essential in determining the path a packet takes to travel across different networks and ensuring efficient communication between devices. The two primary types of switching techniques are **circuit switching** and **packet switching**.

Circuit Switching

Circuit switching is a method where a dedicated communication path or circuit is established between two devices for the duration of their conversation or data transmission. Once the circuit is established, the communication takes place over this dedicated path, and the circuit is released once the communication ends.

- **How it Works**: A connection is established between the sender and the receiver before any data can be transmitted. This path is exclusive, and no other data can be sent along it until the conversation ends. This type of switching is often used in traditional telephone networks.
- **Advantages**:
    - Constant data rate and reliable communication.
    - No delays once the circuit is established.
- **Disadvantages**:
    - Inefficient use of resources: Even during idle times, the communication path remains reserved, which can lead to underutilization.
    - Setup delays: Establishing the circuit takes time.

Example: Traditional telephone networks where a dedicated circuit is established for each phone call.

Packet Switching

In **packet switching**, data is broken into small packets and each packet is sent independently over the network. Each packet may take a different route to the destination, and the packets are reassembled at the receiver. Packet switching is more flexible and efficient than circuit switching, particularly for data networks like the internet.

- **How it Works**: Data is divided into smaller packets, each with a header containing routing information. These packets are then transmitted across the network, and routers or switches determine the best path for each packet. Once all packets reach the destination, they are reassembled in the correct order.
- **Advantages**:
    - More efficient use of network resources.
    - No need for a dedicated path, so the network can handle multiple communications simultaneously.

- **Disadvantages**:
    - Variable data rates: Since packets may take different paths, the arrival time of packets can vary, leading to potential delays.
    - Packet loss or congestion can occur if the network is overloaded.

Types of Packet Switching:

- **Connectionless Datagram Switching**: In this method, each packet is treated independently. There is no need for a pre-established connection. Each packet is routed independently, and packets may take different paths to reach the destination. This type of switching is used in the Internet's IP-based communication.
    - **Advantages**: Flexible and scalable, as there is no need for a dedicated connection.
    - **Disadvantages**: Potential for out-of-order packets and higher overhead.
- **Connection-Oriented Virtual Circuit Switching**: This method establishes a virtual connection before the data is transmitted. Even though the connection is virtual, the packets follow the same predefined path through the network, ensuring that they are delivered in the correct order.
    - **Advantages**: Ensures reliable packet delivery and in-order reception.
    - **Disadvantages**: Slight overhead for maintaining the virtual connection and slower to set up.

Examples: The Internet uses connectionless datagram switching via IP, whereas protocols like ATM (Asynchronous Transfer Mode) use connection-oriented virtual circuit switching.

Data Link Layer Functions and Protocol

The **Data Link Layer (DLL)** is the second layer in the OSI model, responsible for transferring data between adjacent network nodes in a wide area or local area network. The key functions of the data link layer include error detection and correction, flow control, and the proper framing of data to ensure reliable communication.

Error Detection and Error Correction Techniques

Error detection and correction are critical to maintaining data integrity in communication systems. The data link layer uses specific techniques to identify and correct errors that may occur during data transmission.

- **Error Detection**: Techniques like **parity bits**, **checksums**, and **Cyclic Redundancy Check (CRC)** are used to detect errors in transmitted data.
    - **Parity Bit**: A single bit added to a data unit to make the number of 1s either even (even parity) or odd (odd parity). This helps detect errors in data but cannot correct them.
    - **Checksums**: A mathematical value derived from the data being transmitted, and the receiver uses it to check if the received data is correct.
    - **CRC (Cyclic Redundancy Check)**: A more advanced error detection technique that uses polynomial division to check for errors in data transmission. It is highly reliable and widely used in networks.

- **Error Correction**: When errors are detected, error correction techniques are used to recover the original data. This can involve requesting retransmissions of the corrupted data or using algorithms like **Hamming Code** to correct errors automatically.
    - **Forward Error Correction (FEC)**: Techniques where redundant bits are added to data, allowing the receiver to detect and correct errors without needing a retransmission.

Data-Link Control: Framing and Flow Control

- **Framing**: Framing refers to the process of dividing the data into manageable chunks (frames) for transmission. Each frame consists of a header, data, and a trailer, where the header contains control information like addresses and sequencing information, and the trailer contains error-checking information.
    - **Types of Framing**:
        - **Character-Oriented Framing**: A frame is marked by special characters (e.g., start and stop flags).
        - **Bit-Oriented Framing**: Framing is done using a sequence of bits, and bit-stuffing techniques are used to ensure the start and end delimiters do not appear in the data.
- **Flow Control**: Flow control is a technique used to manage the rate of data transmission between two devices, preventing buffer overflow or congestion. It ensures that the sender does not overwhelm the receiver with too much data too quickly.
    - **Types of Flow Control**:
        - **Stop-and-Wait**: The sender sends one frame and waits for an acknowledgment (ACK) from the receiver before sending the next frame. This ensures that the receiver is not overloaded but can lead to inefficient use of the channel.
        - **Sliding Window Protocol**: This allows multiple frames to be sent before receiving an acknowledgment, increasing the efficiency of data transmission.

Error Recovery Protocols

To maintain reliable communication, error recovery protocols are used to manage errors, ensuring that lost or corrupted data can be retransmitted. Some of the widely used protocols for error recovery include:

- **Stop-and-Wait ARQ (Automatic Repeat Request)**: In this protocol, the sender transmits one frame and waits for an acknowledgment from the receiver. If no acknowledgment is received within a timeout period, the frame is retransmitted.
    - **Advantages**: Simple to implement and ensures reliability.
    - **Disadvantages**: It is inefficient, especially in high-latency networks, as it requires the sender to wait for an acknowledgment before sending the next frame.
- **Go-Back-N ARQ**: In this method, the sender can send multiple frames before receiving acknowledgments. If an error occurs in any frame, the sender must retransmit the erroneous frame and all subsequent frames.
    - **Advantages**: Better utilization of network resources than Stop-and-Wait.

- **Disadvantages**: Retransmitting multiple frames can lead to inefficiency, especially in high-error-rate networks.

- **Selective Repeat ARQ**: This protocol allows the sender to retransmit only the specific frames that were lost or corrupted, rather than all subsequent frames as in Go-Back-N ARQ.
  - **Advantages**: More efficient than Go-Back-N as only the lost or corrupted frames are retransmitted.
  - **Disadvantages**: More complex to implement than Go-Back-N.

Point-to-Point Protocol (PPP) on the Internet

The **Point-to-Point Protocol (PPP)** is a data link layer protocol used to establish a direct connection between two nodes, typically for communication over serial links such as dial-up connections or VPNs. PPP provides features like:

- **Framing**: PPP encapsulates data into frames for transmission.
- **Error Detection**: It includes mechanisms for error detection (via a checksum).
- **Authentication**: PPP can use authentication methods (e.g., PAP or CHAP) to verify the identity of the nodes.

PPP is widely used for Internet connections, especially in older dial-up modems and virtual private networks (VPNs).

Conclusion

This unit provides a comprehensive understanding of **network switching techniques** and the **data link layer functions**. We explored circuit switching and packet switching, with a focus on connectionless and connection-oriented methods. The data link layer plays a crucial role in ensuring reliable data transfer by using error detection, error correction, and flow control techniques. Protocols like ARQ and PPP are essential for maintaining data integrity and ensuring efficient communication across networks.

Unit IV: Access Mechanisms & Network Layer

Access mechanisms Multiple Access Protocol and Networks: ALOHA,CSMA/CD protocols, Ethernet LANS, connecting LAN and back-bone networks- Repeaters. Hubs, Switches, Bridges, Router and Gateways.

Networks Layer Functions and Protocols: Routing, Routing algorithms. Network layer protocol of Internet- IP protocol. Internet control protocols.

Access Mechanisms, Multiple Access Protocols, and Networks

In network communication, **access mechanisms** are the methods by which multiple devices share and access the same communication medium. Multiple access protocols are essential in ensuring that devices in a network can transmit and receive data without interference. This section will explore various access mechanisms and protocols, along with the components involved in connecting local area networks (LANs) and backbone networks.

ALOHA Protocol

ALOHA (Additive Links On-line Hawaii Area) is one of the simplest multiple access protocols used for communication. It was initially developed for satellite communication but has also been used for local area networks.

- **Pure ALOHA**: In Pure ALOHA, a station sends data packets whenever they have data to transmit. If the transmission does not collide with another station's transmission, it will be successfully received by the receiver. If a collision occurs, the station will retransmit the data after a random amount of time.
    - **Advantages**: Simple to implement.
    - **Disadvantages**: Inefficient, as collisions are common in busy networks, resulting in many retransmissions.
- **Slotted ALOHA**: In Slotted ALOHA, time is divided into slots, and each station waits for the beginning of a slot to send its packet. This reduces the chances of collision because all stations now attempt transmission at synchronized time intervals.
    - **Advantages**: More efficient than Pure ALOHA.
    - **Disadvantages**: Still prone to some collisions, especially under high network loads.

CSMA/CD Protocol

Carrier Sense Multiple Access with Collision Detection (CSMA/CD) is a protocol used in Ethernet networks to control access to the shared communication medium. The protocol aims to reduce collisions by sensing whether the channel is busy before transmitting data.

- **How it Works**:
    - **Carrier Sensing**: A station listens to the channel before transmitting to check if another station is already transmitting.
    - **Transmission**: If the channel is idle, the station transmits. If the channel is busy, the station waits until it is clear.
    - **Collision Detection**: While transmitting, the station continues to listen for any collision. If a collision occurs, the station stops transmitting, waits for a random backoff time, and then retries.

- **Advantages**: It minimizes the number of collisions and reduces network congestion.
- **Disadvantages**: CSMA/CD is only efficient in low-traffic conditions. It becomes inefficient in heavily loaded networks due to frequent collisions.

Ethernet LANs

Ethernet is a widely used **LAN (Local Area Network)** technology based on the CSMA/CD protocol. It enables devices within a local area to communicate over a shared medium, typically using twisted pair cables or fiber optics. Ethernet standards (e.g., IEEE 802.3) define the rules for data transmission in a network.

- **Ethernet Frames**: Ethernet frames consist of a header, data payload, and a checksum (for error detection). Ethernet uses MAC addresses for addressing devices.
- **Ethernet Speeds**: Ethernet speeds have evolved from 10 Mbps to 100 Mbps (Fast Ethernet), 1 Gbps (Gigabit Ethernet), and 10 Gbps (10-Gigabit Ethernet).
- **Switching**: In modern Ethernet networks, **switches** replace hubs for efficient data transmission, reducing collisions by providing dedicated communication channels between devices.

Connecting LANs and Backbone Networks

In large networks, multiple LANs are connected to form a **backbone network** that handles the data traffic between them. Various devices are used to connect and manage the flow of data between these networks.

- **Repeaters**: A repeater is a device that amplifies or regenerates signals to extend the transmission distance. It is used in cases where the signal weakens over long distances (e.g., in long Ethernet cables).
- **Hubs**: A hub is a basic network device that broadcasts the data it receives to all connected devices. While it connects devices in a network, it is less efficient than switches because it does not direct data to specific devices.
- **Switches**: A **switch** is a more intelligent version of a hub. It forwards data only to the device with the correct MAC address, reducing network congestion and improving overall network performance.
- **Bridges**: A **bridge** connects two or more network segments, typically within the same LAN, to allow communication between them. Bridges filter traffic by examining MAC addresses and forwarding only relevant data between segments.
- **Routers**: A **router** connects multiple networks (e.g., connecting a LAN to the internet or multiple LANs). Routers make forwarding decisions based on IP addresses and determine the best path for data to travel across different networks.
- **Gateways**: A **gateway** is a device that connects different types of networks, often with different protocols. It acts as a translator, converting one protocol to another to enable communication between different network architectures.

Network Layer Functions and Protocols

The **Network Layer (Layer 3)** of the OSI model is responsible for routing data from the source to the destination across different networks. This layer performs important tasks like addressing, routing, and packet forwarding.

Routing and Routing Algorithms

Routing is the process of selecting the best path for data to travel across a network. Routers perform this task by using various routing algorithms, which determine the most efficient route based on network topology and other factors such as traffic load, latency, and path reliability.

- **Routing Algorithms**: There are several types of routing algorithms:
    - **Distance Vector Routing**: Each router maintains a table of the distance to various destinations, typically using the Bellman-Ford algorithm. Routers exchange distance tables to update their knowledge of the network.
    - **Link-State Routing**: Routers send information about their directly connected neighbors and their link status. This allows each router to construct a complete network topology and determine the best route using algorithms like Dijkstra's algorithm.
    - **Hybrid Routing**: Combines aspects of both distance vector and link-state routing (e.g., Enhanced Interior Gateway Routing Protocol, or EIGRP).
- **Static vs. Dynamic Routing**: Static routing requires manual configuration of routing tables, while dynamic routing allows routers to automatically update their routes in response to changes in the network.

Network Layer Protocols of the Internet

The **Internet Protocol (IP)** is the fundamental protocol of the Internet. It is a connectionless protocol that defines how data packets are addressed and routed across networks.

- **IPv4**: The most commonly used version of IP, which uses a 32-bit address format to identify devices on a network (e.g., 192.168.1.1). IPv4 provides about 4.3 billion unique addresses, which are currently being exhausted.
- **IPv6**: IPv6 is the successor to IPv4, providing a much larger address space (128-bit address format) to accommodate the growing number of devices connected to the Internet. It also includes improved security features and better routing efficiency.
- **IP Addressing**: An IP address consists of two main parts:
    - **Network Part**: Identifies the network.
    - **Host Part**: Identifies the specific device on the network.

Internet Control Protocols

Internet Control Protocols are used to manage and control the behavior of networks and ensure reliable communication. Common Internet control protocols include:

- **ICMP (Internet Control Message Protocol)**: ICMP is used to send error messages and operational information about the network. For example, when a device cannot be reached, ICMP sends a "destination unreachable" message. It is also used by the **ping** command to test connectivity between devices.
- **ARP (Address Resolution Protocol)**: ARP is used to map an IP address to a MAC address within a local network. When a device knows an IP address but needs to find the corresponding MAC address, it sends an ARP request to the local network.
- **DNS (Domain Name System)**: DNS translates human-readable domain names (e.g., www.virendragoura.com) into IP addresses that computers use to identify each other on the network.

Conclusion

In this unit, we covered the key concepts related to **access mechanisms** and **multiple access protocols**, such as ALOHA, CSMA/CD, and Ethernet LANs. We also discussed the different devices used to connect and manage networks, such as repeaters, hubs, switches, routers, and gateways. Additionally, we explored the **network layer** functions and protocols, focusing on routing, IP, and Internet control protocols like ICMP, ARP, and DNS. Understanding these fundamental concepts is essential for designing and managing efficient network system

Unit V: Transport Layer & Application Protocols

Transport Layer Functions and Protocols: Transport services, Berkeley socket interface overview, Transport layer protocol of Internet- UDP and TCP. Overview of Application layer protocol, DNS protocol, WWW & HTTP protocols.

Transport Layer Functions and Protocols

The **Transport Layer** is the fourth layer in the OSI model and plays a crucial role in providing end-to-end communication services for applications. It ensures reliable data transfer between hosts by establishing, maintaining, and terminating communication sessions. This layer is responsible for flow control, error handling, and segmentation of data to make it suitable for transmission over the network.

Transport Services

The transport layer provides several key services that are critical for enabling reliable data communication between applications running on different devices.

- **Segmentation and Reassembly**: The transport layer takes large chunks of data from the application layer and divides them into smaller segments. These segments are sent over the network and reassembled into the original data at the receiving side.
- **Reliability**: The transport layer can ensure data reliability by providing mechanisms for error detection, acknowledgment, and retransmission of lost or corrupted data.
- **Flow Control**: Flow control ensures that the sender does not overwhelm the receiver with too much data. The transport layer can control the rate of data transmission to ensure efficient communication.
- **Multiplexing**: The transport layer allows multiple communication sessions between applications on the same device, each identified by unique port numbers. It multiplexes multiple streams of data into a single transmission channel.
- **Error Detection and Correction**: Transport protocols like TCP provide error detection and correction by adding checksums to data segments. If errors are detected, the transport layer ensures that the data is retransmitted.

Berkeley Socket Interface Overview

The **Berkeley Sockets API** is a programming interface that provides the means for communication between applications over a network. It is widely used for implementing network applications in the transport layer.

- **Socket**: A socket is an endpoint for communication. It is a combination of an IP address and a port number, and it allows the transport layer to provide communication between two applications on different devices.
- **Types of Sockets**:
    - **Stream Sockets (TCP)**: These provide reliable, connection-oriented communication. Stream sockets are used for protocols like TCP.
    - **Datagram Sockets (UDP)**: These provide connectionless, unreliable communication, and are used for protocols like UDP.
- **Socket Operations**: The socket interface allows applications to create, bind, listen, accept, connect, send, receive, and close sockets to facilitate network communication.

Transport Layer Protocols of the Internet

The two primary transport layer protocols used in Internet communication are **UDP (User Datagram Protocol)** and **TCP (Transmission Control Protocol)**. These protocols serve different purposes, and understanding their differences is key to choosing the right protocol for specific applications.

UDP (User Datagram Protocol)

UDP is a connectionless transport layer protocol that is designed for applications that require fast, low-latency communication with minimal overhead.

- **Key Features**:
    - **Connectionless**: UDP does not establish a connection before data transfer and does not guarantee that data will be received by the recipient.
    - **No Acknowledgments**: UDP does not perform acknowledgment or retransmission of lost data, making it a faster but less reliable protocol.
    - **No Flow Control**: UDP does not have any built-in mechanism for controlling the rate of data transmission.
    - **Lightweight**: UDP headers are smaller compared to TCP, which makes it more efficient for small, frequent transmissions.
- **Use Cases**: UDP is ideal for applications where speed is more critical than reliability, such as video streaming, real-time gaming, DNS queries, and VoIP.

TCP (Transmission Control Protocol)

TCP is a connection-oriented transport layer protocol that provides reliable, error-checked, and sequenced communication between devices.

- **Key Features**:
    - **Connection-Oriented**: TCP establishes a connection before data is sent, ensuring both the sender and receiver are ready to exchange data.
    - **Reliability**: TCP guarantees reliable delivery of data by using acknowledgments, retransmissions, and sequence numbers to ensure data is delivered in the correct order.
    - **Flow Control**: TCP uses flow control mechanisms such as **Sliding Window Protocol** to prevent congestion and ensure the receiver can handle the incoming data rate.
    - **Congestion Control**: TCP dynamically adjusts the rate of data transmission based on network congestion, ensuring the network is not overloaded.
- **Use Cases**: TCP is used in applications where reliability and accuracy are critical, such as web browsing (HTTP), email (SMTP), file transfer (FTP), and remote login (SSH).

Overview of Application Layer Protocols

The **Application Layer** is the topmost layer in the OSI model and interacts directly with end-user applications. It provides protocols and interfaces for communication between networked devices. Some widely used application layer protocols include **DNS**, **HTTP**, and **WWW**.

DNS Protocol (Domain Name System)

The **Domain Name System (DNS)** is an application layer protocol that translates human-readable domain names (e.g. www.virendragoura.com) into IP addresses (e.g., 192.168.1.1) that computers use to identify each other on the network.

- **DNS Functionality**:
    - **Name Resolution**: DNS resolves domain names to IP addresses, allowing users to access websites using human-readable names rather than numerical IP addresses.
    - **Distributed Database**: DNS is a hierarchical and distributed system, with DNS servers around the world managing parts of the domain name space.
    - **Caching**: To reduce latency, DNS queries are often cached, so repeated queries to the same domain do not need to be resolved again.
- **DNS Structure**: DNS queries can be recursive (where the DNS server queries other servers on behalf of the client) or iterative (where the client queries multiple servers directly).

WWW & HTTP Protocols

The **World Wide Web (WWW)** is an application layer service that allows users to access resources over the Internet, such as web pages, images, and videos. The primary protocol used for communication between web browsers and web servers is **HTTP** (Hypertext Transfer Protocol).

- **HTTP (Hypertext Transfer Protocol)**:
    - **Request-Response Model**: HTTP is a stateless protocol that uses a request-response model. A client (such as a web browser) sends an HTTP request to a server, and the server responds with the requested content.
    - **Methods**:
        - **GET**: Requests data from the server (e.g., requesting a web page).
        - **POST**: Sends data to the server (e.g., submitting a form).
        - **PUT**: Uploads data to the server.
        - **DELETE**: Deletes data from the server.
    - **Stateless**: HTTP is stateless, meaning each request is independent, and the server does not retain any information about previous requests.
- **HTTPS (Hypertext Transfer Protocol Secure)**: HTTPS is the secure version of HTTP, where communication is encrypted using SSL/TLS protocols. This ensures that sensitive data, such as login credentials and financial transactions, are transmitted securely over the Internet.

Conclusion

This unit covers the essential functions of the **Transport Layer** and the protocols that enable communication between applications over the Internet. We explored **UDP** and **TCP**, two fundamental transport protocols that cater to different needs in terms of speed and reliability. Additionally, we discussed **DNS** and **HTTP**, which are crucial **application layer protocols** enabling modern web browsing, domain name resolution, and Internet communication. Understanding the transport layer and application protocols is fundamental to building and managing networked applications effectively.