Lecture 1:

Introduction to Data Communication

What is Data Communication?

Data Communication refers to the process of exchanging data (which can be in digital or analog form) between two or more devices through a transmission medium. This medium can be wired (such as cables, optical fibers) or wireless (such as radio waves, microwaves, or infrared).

- At least two devices are required for communication:
    1. A **Sender:** The device or system that generates and sends data.
    2. A **Receiver:** The device or system that receives the data.
- Data Communication is only meaningful if the data sent is accurately received and understood by the receiving device.

Data communication is not just about sending data; it ensures the correct delivery, accuracy, and timeliness of the data.

Purpose

The primary purpose of data communication is to enable devices to share information and resources efficiently.

- **Information Sharing:** Devices exchange messages, files, audio, video, and sensor readings.
- **Resource Sharing:** Devices can share hardware (like printers, storage devices) and software (like applications or databases) over a network.
- **Collaboration:** Communication allows multiple users or systems to work together in real-time or asynchronously.

Example of Purpose:

- A computer sending a print job to a shared printer (resource sharing).
- A server providing access to an online database to hundreds of users simultaneously (information sharing).

Examples

1. **Sending an Email:**
   - When you send an email from your laptop, it is broken into packets and transmitted through the Internet (a combination of wired and wireless media) to reach the recipient's mail server.
2. **Streaming Videos:**
   - Platforms like YouTube or Netflix deliver large video files over the Internet. Efficient data communication ensures minimal buffering and clear playback.
3. **IoT Devices Transmitting Data:**
   - A smart home thermostat constantly sends temperature readings to a cloud server. This allows the user to monitor and control the temperature from a smartphone app, even when away from home.
4. **Online Banking Transactions:**
   - When you transfer money online, your data is securely transmitted between your bank's system and the central banking server to complete the transaction.

- Data communication is the backbone of modern computing systems. Without it, devices would function in isolation.
- It relies on both hardware (like cables, routers, wireless transmitters) and software protocols (like TCP/IP, HTTP) to work correctly.
- The goal is reliable, fast, and secure exchange of data between devices.

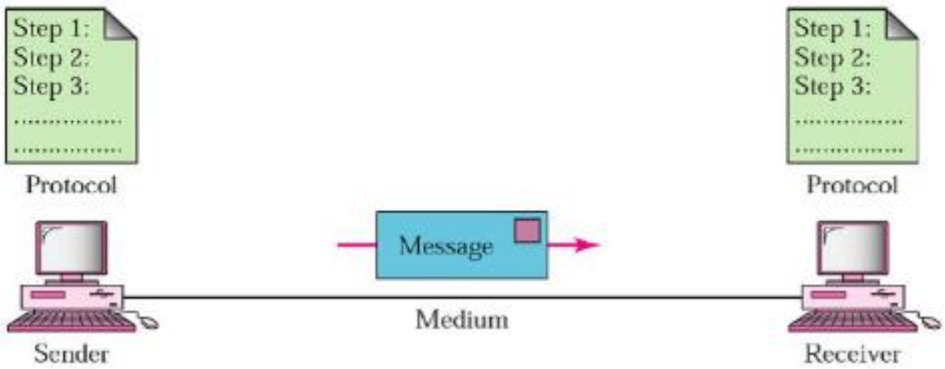
Example Diagram for Notes:

Sender (Laptop) ---> Transmission Medium (Wi-Fi) ---> Receiver (Web Server)

This illustrates how data (like a webpage request) flows from your device to a server through a medium.

Basic Components of Data Communication

For any data communication system to work effectively, it must have five essential components. These components ensure that the data is sent, transmitted, and received correctly.



1. **Message**

The message is the actual data or information that needs to be communicated from the sender to the receiver.

- Types of data:
  - Text (e.g., documents, emails)
  - Numbers (e.g., transaction data)
  - Images (e.g., photos, graphics)
  - Audio (e.g., voice messages, music)
  - Video (e.g., video calls, streaming movies)

Example:

- In a WhatsApp voice message, the message is the audio recording itself.

2. **Sender (Transmitter)**

The sender is the device or person that generates, sends, or transmits the data.

- Function:
    - It encodes the message into signals (digital or analog) suitable for transmission over a medium.
- Examples:
    - Computers, smartphones, IoT sensors, security cameras

Example:

- When you send an email, your computer or smartphone acts as the sender.

3. Receiver

The receiver is the device or person that receives the transmitted data from the medium.

- Function:
    - It decodes the message from the signals and presents it in a usable format to the user.
- Examples:
    - Computers, web servers, printers, cloud storage servers

Example:

- When your friend receives your email, their email server and computer act as the receiver.

4. Medium (Channel)

The medium is the physical path or channel through which the data travels from the sender to the receiver.

- Types:
    - **Wired:** Twisted-pair cables, coaxial cables, optical fiber
    - **Wireless:** Radio waves, microwaves, infrared signals, satellite links

Example:

- If you are using Wi-Fi, the medium is wireless radio signals.
- If you are using a LAN cable, the medium is the copper or fiber cable.

5. Protocol

A protocol is a set of rules and conventions that define how data is formatted, transmitted, and received.

- Purpose:
  - Ensures both sender and receiver understand each other.
  - Handles issues like error checking, data compression, and timing.
- Examples:
  - TCP/IP (Transmission Control Protocol/Internet Protocol)
  - HTTP/HTTPS (Hypertext Transfer Protocol)
  - FTP (File Transfer Protocol)

Example:

- When browsing a website, your browser and the web server follow HTTP or HTTPS protocols to communicate correctly.

Example Scenario: Browsing a Website

1. Sender: Your computer or smartphone
2. Message: HTTP request for the webpage
3. Medium: Wi-Fi network or Ethernet cable
4. Receiver: Web server hosting the website
5. Protocol: TCP/IP and HTTP

Flow:

Your Device (Sender) ---> Wi-Fi/Ethernet (Medium) ---> Web Server (Receiver)

- Your device sends an HTTP request to the server using TCP/IP protocols through the Internet.
- The server receives the request, processes it, and sends back the webpage data.

Protocol used in Communication(Sender end) :

Protocol	**Layer**	**Role**
DNS	Application	Resolves domain name to IP address
UDP	Transport	Used by DNS (though DNS can also use TCP for larger responses)
HTTP	Application	Web request after IP is known
TCP	Transport	Used for reliable data delivery (e.g., HTTP, HTTPS)
IP	Network	Handles routing across networks
Ethernet/Wi-Fi	Data Link	Local delivery via MAC address
PHY	Physical	Transmits raw bits over physical medium

Characteristics of Effective Communication

For data communication to be successful and reliable, it must meet certain essential characteristics. If any of these characteristics fail, the communication may be useless or incomplete.

1. **Delivery**
   Data must be delivered to the correct destination or recipient.
   - The system must ensure that the data reaches the intended device or user and not someone else.
   - Delivery involves proper addressing and routing of the data.

- Example:
  - When you send an email to your teacher, it should be delivered to their email address and not to another student's inbox.

2. Accuracy

Data should be received exactly as it was sent, without any alteration or loss.

- Explanation:
  - Errors during transmission can corrupt data, making it unusable.
  - Communication systems often use error detection and correction techniques to ensure accuracy.
- Example:
  - If you download a software update and a few bits get corrupted, the file may not install or may cause system errors.

3. Timeliness

Data must arrive at the right time for it to be useful.

  - Delays can render the data useless, especially in real-time applications.
  - Timeliness is also called latency—low latency is critical for some applications.
- Example:
  - Online stock trading platforms must display price updates instantly. A delay of even a few seconds can result in financial losses.

4. Jitter

Jitter refers to the variation in packet arrival times during data transmission.

- Explanation:
  - Inconsistent packet arrival can distort audio or video quality, causing uneven playback or buffering.
  - Minimizing jitter is essential for applications like video conferencing and VoIP calls.
- Example:

- In a video call, if packets arrive irregularly, you might hear distorted voices or see frozen video frames.

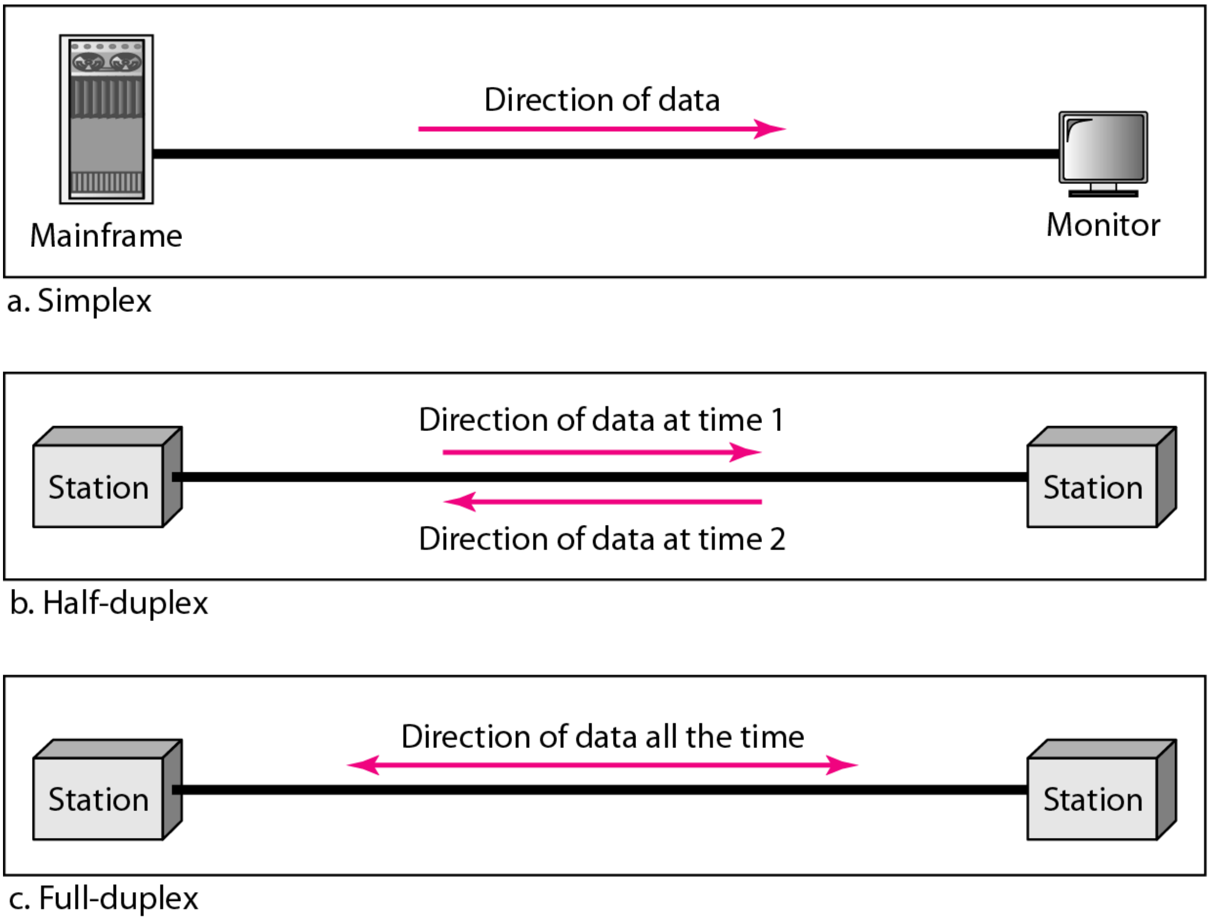
Real-World Example

Video Call (e.g., Zoom, Google Meet):

- If delivery fails, the call may connect to the wrong user.
- If accuracy fails, the audio/video may be garbled or incomplete.
- If timeliness fails, there will be noticeable lags and delays.
- If jitter is high, you may experience buffering or frozen screens.

Types of Data Communication

Data communication between devices can be classified based on the direction of data flow. There are three main types: Simplex, Half-Duplex, and Full-Duplex.



1. Simplex Communication

In simplex communication, data flows only in one direction. The sender can send data, but the receiver can only receive it; it cannot reply.

- It is a one-way communication system.
- The receiver has no way to acknowledge or respond.
- Usually used in applications where feedback is not required.

- Example:
  - Keyboard → CPU: The keyboard sends keypress signals to the CPU, but the CPU does not send data back to the keyboard.
  - Broadcasting systems like radio or TV transmissions are also simplex; the station sends signals, and the audience only receives them.
- Visual Representation:

Sender ------------> Receiver

2. Half-Duplex Communication

In half-duplex communication, data can flow in both directions, but only one direction at a time.

- Sender and receiver take turns to communicate.
- When one device is sending, the other must wait until the channel is free.
- Less efficient than full-duplex, but simpler and cheaper.

- Example:
  - Walkie-Talkies: One person presses the "push-to-talk" button and speaks; the other person can only respond after the channel is released.
  - Two-way intercom systems also work in half-duplex mode.
- Visual Representation:

Sender <----------- Receiver

OR

Sender -----------> Receiver

3. Full-Duplex Communication

In full-duplex communication, data flows in both directions simultaneously. Both devices can send and receive data at the same time.

- Requires two separate communication paths or sophisticated channel-sharing methods.
- Highly efficient, as there is no waiting for the channel to be free.
- Used in applications where real-time communication is necessary.

- Example:

  - Telephone Conversations: Both people can speak and listen at the same time.
  - Internet connections, video calls, and mobile communication are also full-duplex.
- Visual Representation:

Sender <-------------> Receiver

Comparison Table

Type	Direction	Efficiency	Example
Simplex	One-way only	Low	Keyboard → CPU, TV
Half-Duplex	Two-way (one at a time)	Moderate	Walkie-talkies
Full-Duplex	Two-way (simultaneously)	High	Telephones, Internet

Key Takeaways

- Simplex: Best for devices that only need to send data.
- Half-Duplex: Useful for devices that take turns sending and receiving.
- Full-Duplex: Ideal for real-time, interactive communication.

Short Q&A

Q: Why is full-duplex more efficient than half-duplex?

A: Because data can flow in both directions at the same time without waiting.

1. Q: Is television broadcast simplex or full-duplex?

A: Simplex, because the signal is only sent from the TV station to the viewers.

Data Flow Modes

Data flow modes describe how bits are transmitted from the sender to the receiver in a communication system. There are two main types: Serial Transmission and Parallel Transmission.

1. Serial Transmission

In serial transmission, bits of data are sent one after another, sequentially, over a single communication channel or wire.

- Only one bit is transmitted at a time.
- Requires less wiring, making it cost-effective.
- Data transfer speed is slower compared to parallel, but it is more reliable for long distances.
- Less chances to signal interference and synchronization problems.

- Example:

- USB (Universal Serial Bus): Transfers data serially between computer and peripheral devices.
- Communication over the Internet or telecommunication lines also uses serial transmission.

- Visual Representation:

Bit 1 → Bit 2 → Bit 3 → Bit 4 → Bit 5 ...

(One after another on a single wire)

2. Parallel Transmission

In parallel transmission, multiple bits are sent simultaneously, with each bit traveling on a separate channel or wire.

- Several bits (usually 8, 16, 32, or more) are transmitted at the same time.
- Requires multiple wires or channels.
- Faster than serial transmission for short distances.
- Can be affected by synchronization problems and crosstalk (interference between wires) at longer distances.

- Example:
  - Communication between CPU and RAM (system bus) uses parallel transmission.
  - Printers connected with old parallel ports (e.g., Centronics connector) use parallel data transfer.
- Visual Representation:

Bit 1   Bit 2   Bit 3   Bit 4   Bit 5 ...

(Each bit on a separate wire at the same time)

Comparison Table

Aspect	**Serial Transmission**	**Parallel Transmission**
Bits Transferred	One bit at a time	Multiple bits simultaneously
Wires Needed	Single wire or channel	Multiple wires/channels
Distance	Best for long distances	Best for short distances
Speed	Slower than parallel	Faster (but only for short distances)
Examples	USB, Internet communication	CPU ↔ RAM bus, old printer ports

- Serial transmission is cost-effective and reliable for long-distance communication because it uses fewer wires and reduces signal interference.
- Parallel transmission is preferred for short-distance communication where speed is more important, but it becomes impractical for long distances due to synchronization issues.

Short Q&A

1. Q: Why is parallel transmission not used for Internet communication?

   A: It requires multiple wires, which is impractical over long distances and is more susceptible to crosstalk.

Data Representation in Communication

In data communication, all types of information (text, numbers, images, audio, and video) must be **converted into binary form (0s and 1s)** because digital devices understand only binary signals. Different data types use different representation techniques.

1. Text

Text is represented by assigning each character (letters, digits, symbols) a unique binary code using **character encoding standards**.

- **Encoding Methods:**
  - **ASCII (American Standard Code for Information Interchange):** Uses **7 or 8 bits** to represent characters.
  - **Unicode:** Developed to support multiple languages and symbols, uses **16 bits or more**.
    - Example: The character अ (Devanagari) has a unique Unicode code point.
- **Example:**
  - Word **"HELLO"** in ASCII is stored as a sequence of binary codes for H, E, L, L, O.

2. Numbers

Numbers are represented in **numeric systems**, most commonly **binary**, but decimal and hexadecimal are also used.

- **Representation Methods:**
  - **Binary (Base 2):** Uses 0 and 1. (Used internally by computers)
  - **Decimal (Base 10):** Uses digits 0–9. (Human-readable format)
  - **Hexadecimal (Base 16):** Uses digits 0–9 and letters A–F. (Used for compact representation of binary)
- **Example:**
  - Decimal 5 = Binary 101 = Hexadecimal 5
  - Decimal 255 = Binary 11111111 = Hexadecimal FF

3. Images

Images are represented as a collection of **pixels (picture elements)**, where each pixel is assigned a color code in binary.

- **Techniques:**
  - Each pixel's color is stored using codes like **RGB (Red, Green, Blue)** values.
  - Images are compressed and stored using formats such as:
    - **JPEG (Joint Photographic Experts Group):** Compressed image format.
    - **PNG (Portable Network Graphics):** Supports transparency and lossless compression.
- **Example:**
  - A 1920×1080 image has 2,073,600 pixels. Each pixel’s color is represented in binary.

4. Audio/Video

Audio and video signals are continuous (analog), so they must be converted into digital (binary) form using sampling and quantization.

- **Techniques:**
  - **Sampling:** Taking periodic measurements of the signal's amplitude.
  - **File Formats:**
    - **Audio:** MP3, WAV, AAC
    - **Video:** MP4, AVI, MKV
- **Example:**
  - In an MP3 audio file, the sound waves are sampled thousands of times per second and stored as binary values.

Key Takeaways

- **Binary representation (0 and 1)** is the foundation for all types of data in communication.
- Different encoding standards and formats are used depending on the data type.
- Proper representation ensures that data is transmitted, stored, and decoded correctly.

Short Q&A

Q: Why do computers use binary representation for data?

A: Because digital devices operate using two voltage levels (on/off) that map directly to binary (1/0).

Q: Which standard should be used if you want to represent characters from multiple languages?

A: Unicode, because it supports thousands of characters worldwide.

Key Terms

1. **Bandwidth:** Maximum data-carrying capacity of a medium
   - Example: 100 Mbps broadband
2. **Latency:** Time taken for data to travel
   - Example: Satellite Internet has ~600 ms latency
3. **Throughput:** Actual data successfully transmitted per second
   - Example: On a 100 Mbps plan, actual throughput may be 80 Mbps

Q. Give one **real-world example** for each communication mode (simplex, half-duplex, full-duplex).

Possible Answers:

1. Simplex – Fire alarm system
2. Half-Duplex – Walkie-talkies
3. Full-Duplex – Video calls

Q: Can a device be both a sender and a receiver?

A: Yes. Most devices, like computers and smartphones, can both send and receive data.

Q: Which characteristic is most important for a real-time application like online gaming?

A: Timeliness and low jitter.

Q: Is your classroom projector an example of Simplex, Half-Duplex, or Full-Duplex?

A: Simplex – it only receives data from the computer.

Q: Why is serial communication preferred over parallel for long distances?

A: Parallel lines can suffer from signal interference and are more expensive over long distances.

Q: Why is binary used for data representation?

A: Because digital devices use two voltage levels (on/off) which map naturally to binary (1/0).

Q: Name two industries that heavily rely on data communication.

A: Banking and healthcare (among many others).

Q: Why is throughput always less than or equal to bandwidth?

A: Due to network overhead, congestion, and errors.

Q: What is the difference between a protocol and a medium?

A: Protocols are rules for communication; medium is the physical channel.

Q: Which is faster: serial or parallel communication?

A: Parallel is faster for short distances, but serial is more reliable for long distances.

Q: What happens if jitter is too high during a live video stream?

A: The stream will freeze, buffer, or lose synchronization.

Q: Why do we need error detection in communication systems?

A: To ensure accuracy, as data can get corrupted during transmission.