

# PAPER 1

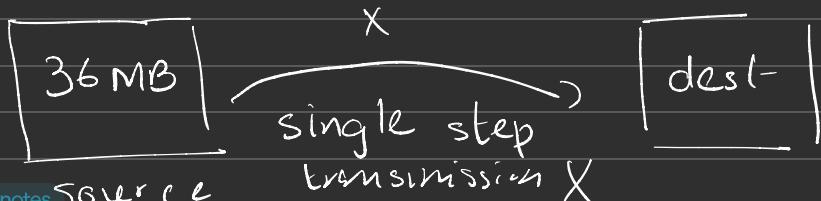
## DATA TRANSMISSION

Intro: This topic deals with how data is transmitted from one device to another over the internet.

Device could be anything like a android phone, ios device, laptop or a PC.

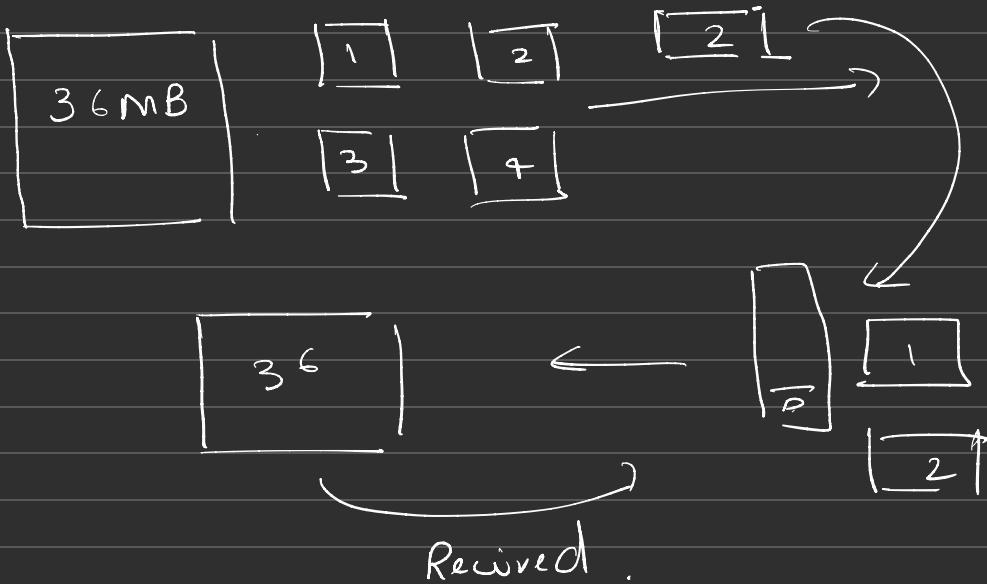
But in computer science we call the sender device as **source** and receiver device as **destination**.

Important Notes: Data is transferred from source device to destination device in form of **Packets**



Rather the data is always broken down into several small **chunks** which are called **Packets**

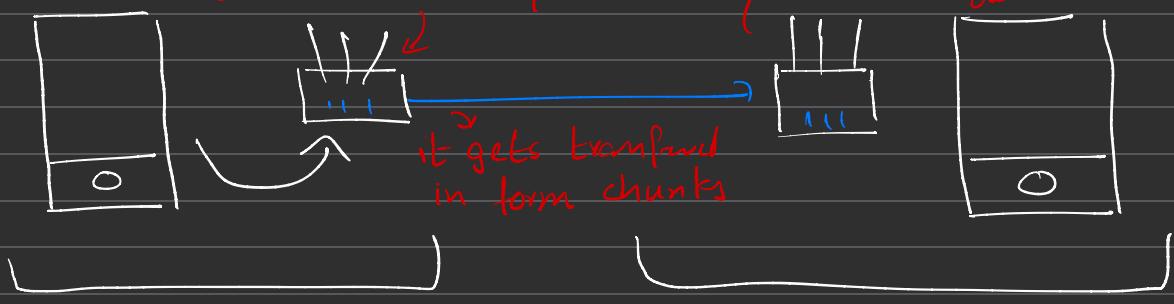
Broken into chunk



- These chunks are called **Packets**
- What divides the message/data into chunks?
- ROUTER → Routen -

At router the data gets broken into packet

At sender side the packets are assembled and sent to destination



Advantage :- The data transmission is much faster and quicker.

- There is no / low latency .

Disadvantage :- One issue is reassembling the packets sent into the same sequence

Source

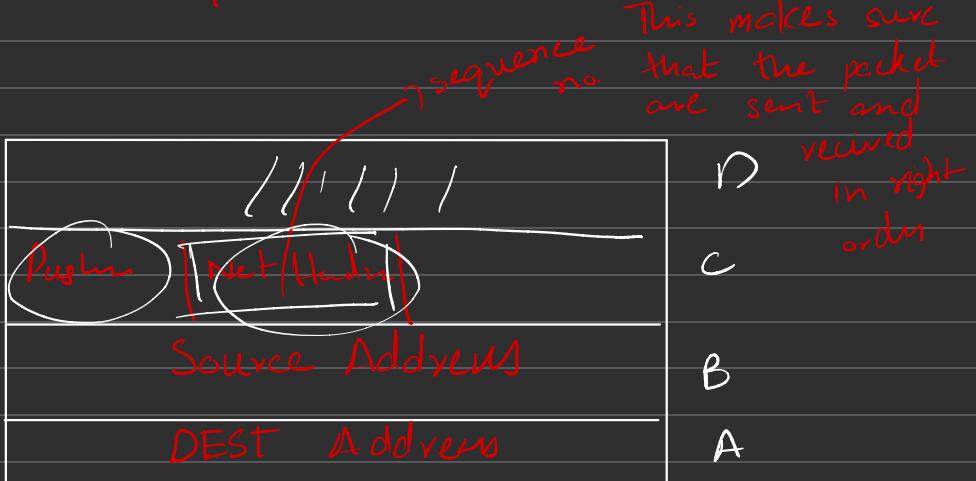


destination



Then how to keep the correct sequence .

For this purpose we need to understand the packet structure



## PACKET

- consist of a packet header, payload, trailer
- The packet header further consists of
  - ① IP Add of receiver
  - ② IP sender
  - ③ Packet Number
- The payload consist of Actual data.
- But if any error occurs we can still use a method to check and identify the errors in data transmission. Method name is CRC (cyclic Redundancy Check).

# Packet Transmission

Circuit  
Switching

Packet  
Switching

## Circuit Switching

- A dedicated communication path is established between sender and receiver before data transmission starts.
- Router checks if the path is empty to avoid collision.
- The entire path stays reserved until communication ends.
- All data follows the same path.
- It is like a telephone call. once connected, the line is yours until you hang up.

**Pros:** ① consistent and reliable connection.  
② No delay once connected.

**Cons:** ① wasted Resources.  
② slower setup.

## Packet Switching:

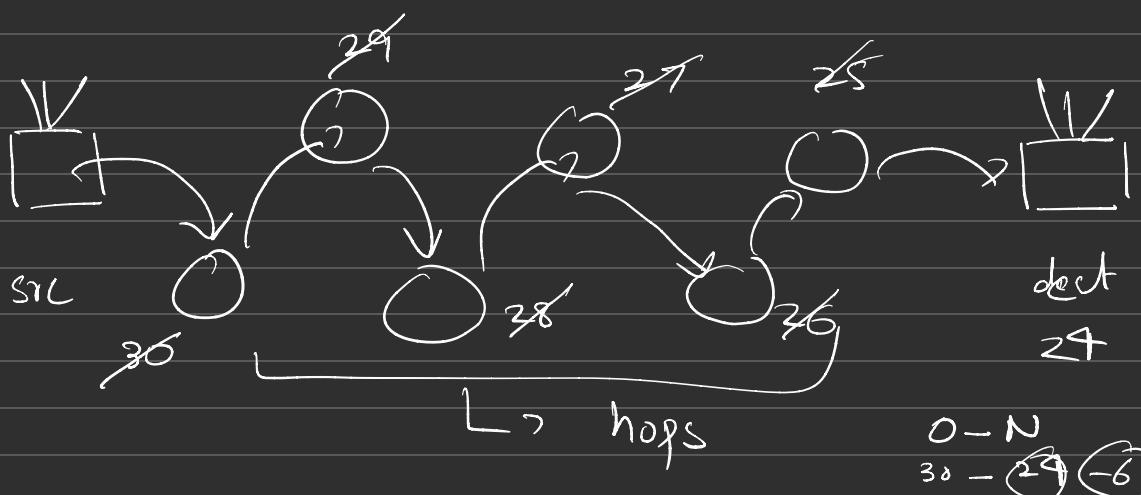
- Each packet can travel through different routes.
- At destination packet are assembled in correct order

**Pros:** ① very efficient use of network  
② No need to reserve a path.  
③ scales with large network.

**Cons:** ① Packets arrive out of order - can experience

## HOP NUMBER

when the packets are sent from a source to destination they keep on hopping from router to router until they arrive at the desired destination

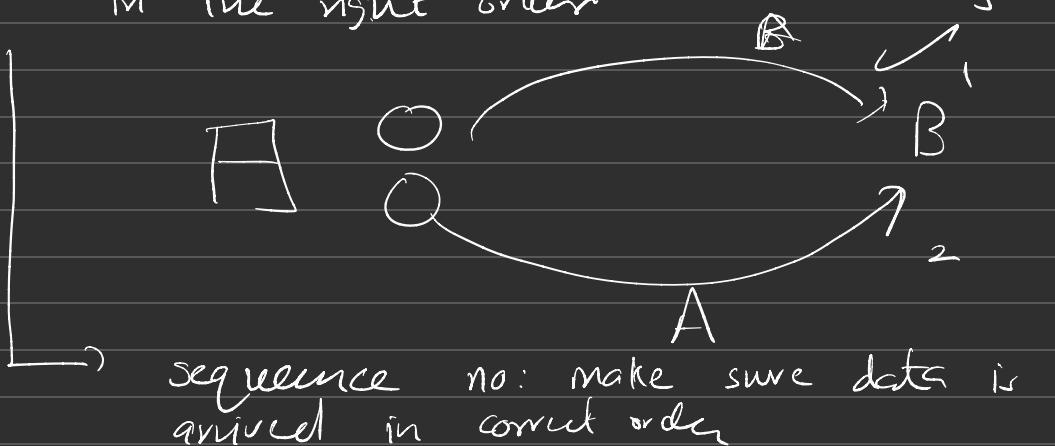


- Every packet has a hop number stored in its header which lowered by 1 by each hop

B A\*

# DATA TRANSMISSION :

- ① Data transmission direction
  - ↳ single direction
  - ↳ Multi direction
- ② Data Transmission
  - ↳ max no. of bits that can be transferred.
- ③ Methods for synchronization data.
  - ↳ How to ensure that data is received in the right order



# TYPES OF TRANSMISSION

- ① Simplex Transmission
- ② Half Duplex "
- ③ Full Duplex Transmission

## Simplex Transmission :

→ The data is only transferred in one direction

e.g. printer C → P ✓ P → C X  
speaker  
headphone

## Half Duplex :

→ Data can be sent in both directions along the same transmission line but NOT SIMULTANEOUSLY

walkie talkie

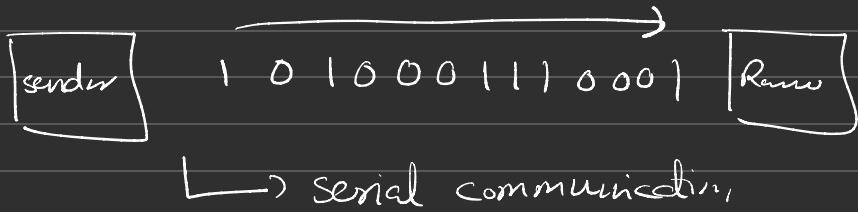
## FULL DUPLEX :

→ same as half duplex but data is sent and received simultaneously.

- telephone lines.
- google meet

## Serial Data Transmission :

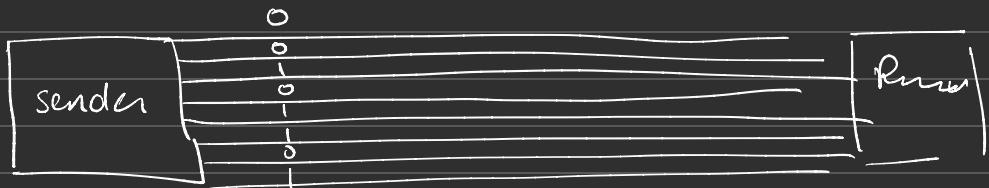
When data is transferred via single channel or single wire. One bit at a time.



## Parallel Data Transmission

Many data bits are sent using many parallel channels. usually 8 bits / 1 byte.

$$1 \text{ Byte} = 8 \text{ bits}$$



# Aynchronous Transmission

↳ The data is transferred in predefined bit pattern

# Synchronous

↳ No predefined 8/16 Bit sequence  
↳ The internal clock generates timing signals that are communicated with the data.

# Error Checking Techniques

- ① Parity check
- ② Check sum
- ③ Echo check

Parity Check: We determine the no of "1's" that make up the byte

- ① Odd Parity  $\rightarrow$  1's are odd than odd
- ② Even Parity  $\rightarrow$  1's are even than parity

$$\begin{array}{ccccccccc} & 64 & 32 & 16 & 8 & 4 & 2 & 1 \\ \downarrow & \downarrow \\ \text{even} & & & & & & & \end{array}$$
$$64 + 32 + 8 + 4$$
$$8 + 2 + 1$$

0 ① ① 0 ① ① 0 0

① ① ① 0 ① ① 0 0  
odd parity

Check Sum: Data is transferred with a supplementary value that is delivered at conclusion of each block

### Steps

- ① check is computed just before transmission
- ② Algorithm is used to perform calculation
- ③ transmit data along checksum
- ④ checksum is recalculated at receiver end.
- ⑤ compare the sent and received checksum
- ⑥ If donot match then retransmission occurs