3.1 Introduction and Transport-Layer Services

The Transport Layer is 4th Layer of the OSI model provides logical communication between application processes running on different hosts. It ensures that data is delivered efficiently, reliably, and securely from one end system to another.

functions of the transport Layer

1. process-to-process communication (~~host to host delivery~~)
2. Multiplexing and Demultiplexing
3. Connection-oriented (TCP) and connectionless (UDP) Connection management
4. Reliable Data transfer (TCP)
5. flow Control
6. Congestion Control

3.1.1 Relationship Between Transport and Network Layers

The Transport Layer (TCP/UDP) and Network Layer (IP) work together to deliver data across the internet but both have different jobs.

The Network Layer (IP) - The "postal service" Moves packets from one host to another (e.g. from your laptop to a youtube server). IP Addresses is used for host-to-host communication.
Network Layers works by

1. using IP Addresses to route packets (192.168.1.1)
2. Best-effort delivery but Not guarantees that packets

can be Reach the destination, arrive in the Correct o
Arrive without errors, Arrive on time.

↳ Network layer (IP) is unreliable.
↳ Host to host communication is the function of Netw Lay

Transport Layer :- uses port numbers (e.g 80 for HT provides process-to process

Communication. It is both reliable /unreliable Because It has both TCP and UDP concept.

The courier service
↳ Ensures data reaches the correct application (eg chrome browser, whatsApp) inside the host using port numbers.
Eg web browser (port 80) vs. email client (p

Transport Layers works using
(i) port numbers
(ii) TCP (Reliable courier) guarantees:
→ no lost/damaged data (retransmits if needed).
→ manages speed control (flow and congestion contro
→ Ensures in-order delivery → Retransmits Lost data
(iii) UDP (fast but unreliable courier):
No guarantees - just sends data quickly
Good for video calls, gaming.
No retransmits lost data (No recovery).

3.1.2 Overview of the Transport Layer in the Internet
↳ Transport Layer implements end-to-end communication (TCP & UDP manages)
↳ End to End error recovery is handled by TCP (transport layer) because it ensur

ensuring acknowledgements and retransmissions.

→ Port numbers are used to achieved process to process communication in Transport Layers.

↳ UDP process-to-process communication doesnot guarantee delivery because UDP is connection less and uses ports but doesnot ensure reliability.

① Difference between TCP and UDP.

TCP	UDP
① TCP stands for Transmission Control Protocol	① UDP stands for User Datagram protocol.
② Connection oriented	② Connection less
③ More Reliable (Acknowledgements and retransmissions).	③ Unreliable (no guarantees, no Acknowledgement
④ Data arrives in order.	④ No ordering (packets may arrive out of order).
⑤ Slower in speed (due to error checking).	⑤ faster (minimal overhead)
⑥ flow control (adjusts speed to avoid receiver overload).	⑥ No flow control
⑦ Congestion control	⑦ No congestion control
⑧ Error Recovery (Retransmits lost / corrupted packets)	⑧ No error recovery discards bad packets)
⑨ eg web browsing (HTTP), emails (SMTP), file transfer (FTP).	⑨ eg Video calls (zoom), gaming, DNS, Live Streaming

3.2 multiplexing and Demultiplexing

Multiplexing is a technique used to Combine multiple signals into one signal over a shared medium. It improves efficiency by allowing multiple data streams to be transmitted simultaneously.

If analog signals are multiplexed, it is Analog multiplexing and if digital signals are multiplexed that process is Digital multiplexing.

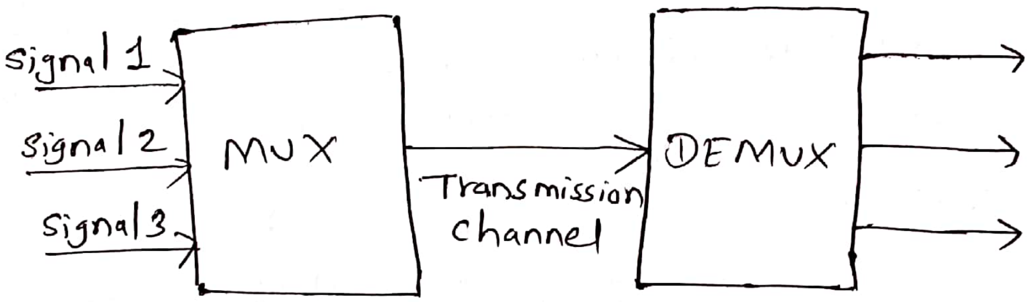


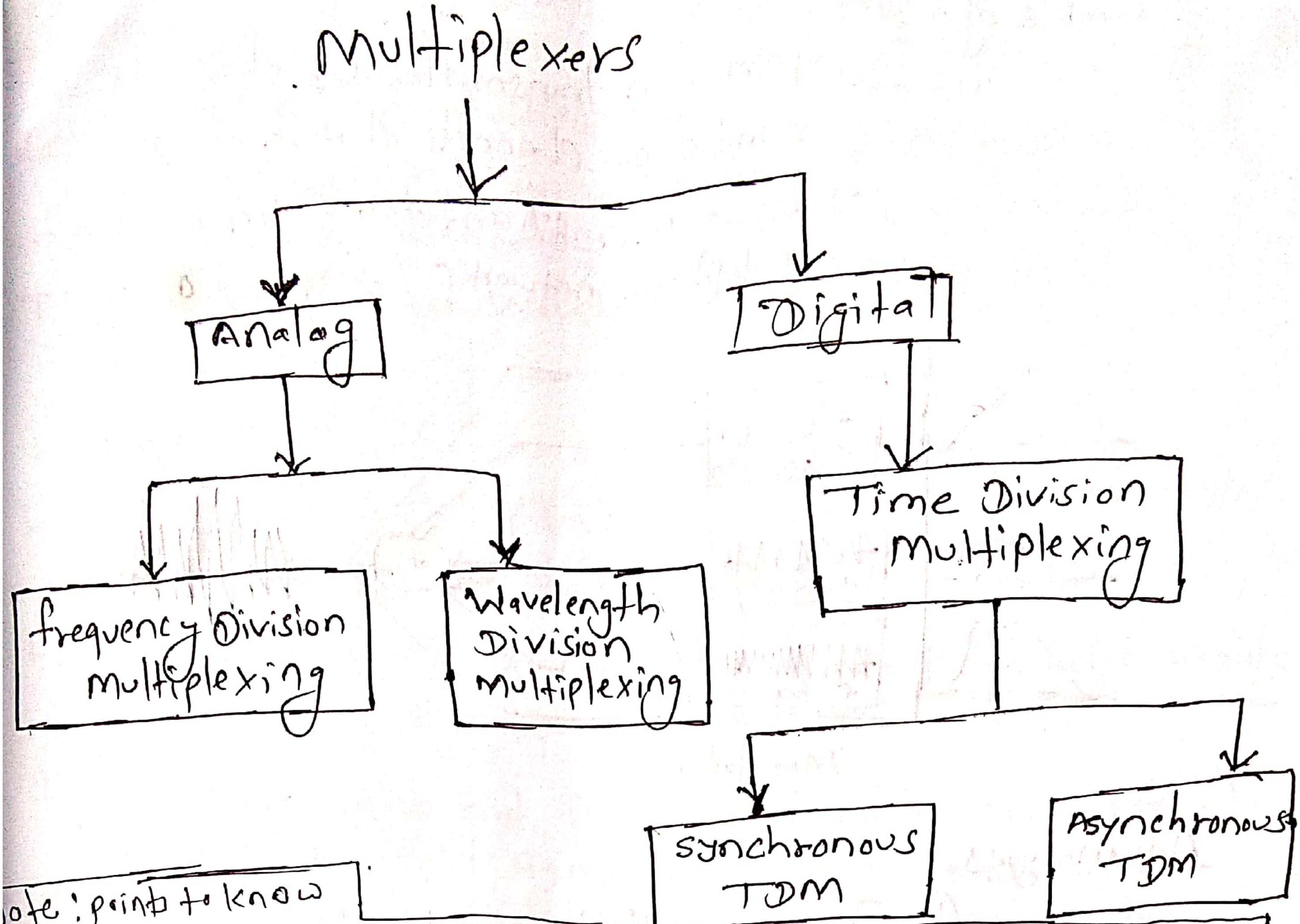
fig:- multiplexer and Demultiplexer

The device that does multiplexing can be simply called called as a MUX while the one that reverses the process which is demultiplexing is called as DEMUX.

Types of Multiplexers

There are mainly two types of multiplexers, namely analog and digital. They are further divided into FDM, WDM and TDM.

Scanned with OKEN Scanner



Note: points to know
"Bandwidth is maximum amount of ~~data~~ that can be transmitted over a Communication channel in a given amount of time"

1 frequency Division multiplexing :- frequency shared in FDM. In FDM, the transmission of multiple signals using different frequency slots transmit over a common link. FDM use only analog signals. Synchronization is not needed in FDM. The overall cost of an FDM is higher.

FDM divides the available bandwidth into multiple frequency bands, each carrying a separate signal.

<u>How FDM works:</u>

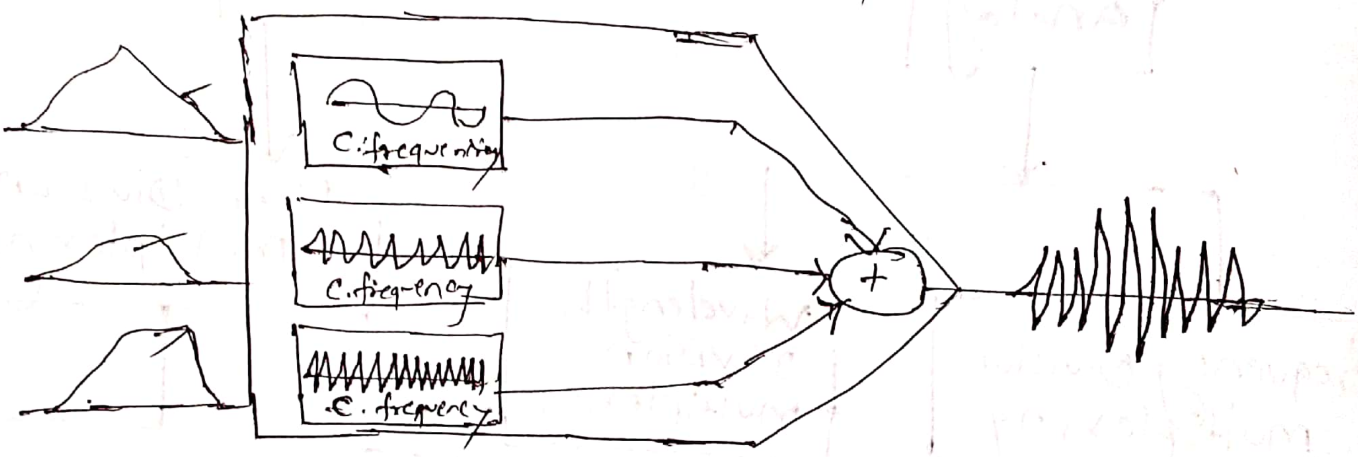
↳ Each signal is assigned a different frequency band.
↳ Signals are combined and transmitted simultaneously.
↳ A demultiplexer at the receiver extracts individual signal based on frequency.

Example of FDM

→ Television transmitter which sends a number of channels through a single cab[le]

→ Used in radio and television broadcasti[ng]

→ Telephone networks / FM radio / cable T[V]



Modulator

Advantages

(i) Enables multiple signals to be transmitted [at] the same time.

(ii) Efficient for Analog signals.

(iii) Less time delay as compared to TDM.

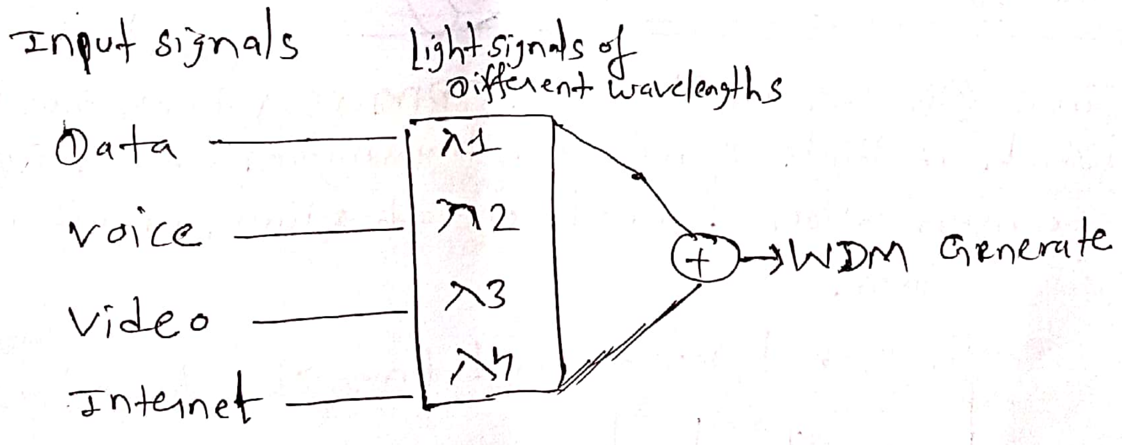
Wavelength Division Multiplexing

Wavelength Division Multiplexing is a technique used in fiber-optic communication where multiple signals are transmitted simultaneously over a single optical fiber by using different wavelengths (colors) of light.

How WDM works

- Multiple data signals (voice, video, internet) are converted into light signals of different wavelengths (colors).
- The WDM multiplexer (MUX) combines these light signals and transmits them over a single optical fiber.
- At the receiving end, the WDM demultiplexer (DEMUX) separates the light signals based on their wavelengths.
- Each separated signal is sent to its respective destination.

Diagram



WDM is a highway where each vehicle represents data data, voice, video, Internet signals. Each lane represents a different wavelength. All vehicles travel on the same road (fiber optic cable) but they stay in their assigned lanes (wavelengths) and do not interfere with each other.

Scanned with OKEN Scanner

Types of WDM

(i) Coarse wavelength Division multiplexing (CDW
(ii) Dense wavelength Division multiplexing (DWDM

(1) Coarse wavelength Division multiplexing (CW

→ CWDM uses fewer wavelengths with wide gaps between them. (Allowing congestion free

→ uses 8 to 18 wavelengths.

→ Best for shorter distances (up to 80 km) due to lower power.

(2) Dense wavelength Division multiplexing (DWDM)

→ DWDM uses many wavelengths to carry more d

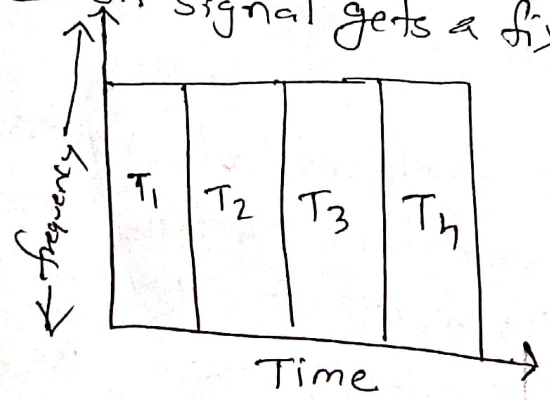
→ uses 40 to 160 wavelengths.

→ High Cost but supports long distances.

Time Division multiplexing

Time Division multiplexing TDM is a Digital technique in which multiple signals are transmitted over a single Communication channel, one at a time, in different time slots.

Each signal gets a fixed time slots to send its data



→ Radio systems use TDM.

→ Access to full length frequenc of channel.

→ No Gaurdband required for the wideband system.

Scanned with OKEN Scanner

Types of TDM (Time Division multiplexing)

1) Synchronous Time Division multiplexing

→ Each signal gets a fixed time slot, even if it has no data to send.

→ simple but can waste bandwidth.

→ Traditional telephone systems is the example of Synchronous Time Division multiplexing.

Example

Imagine 4 people each get 5 seconds to talk, even if they have nothing to say.

A	B	C (No data)	D	E
5s	5s	5s	5s	5s

C has not data but takes time.

2) Asynchronous Time Division multiplexing

→ Asynchronous TDM is also known as statistical TDM.

→ Time slots are assigned dynamically based on demand.

→ more efficient and saves bandwidth and time when some channels are idle.

Example

only those who have data get to use the mic to say.

A	B	C (No data)	D	E

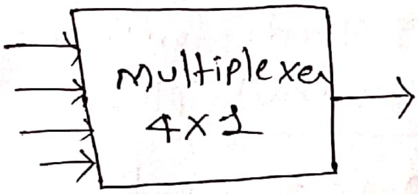
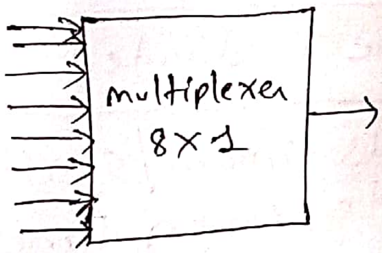
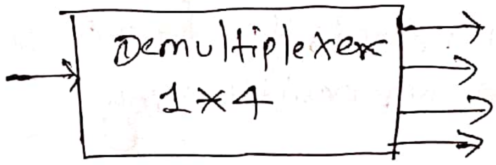
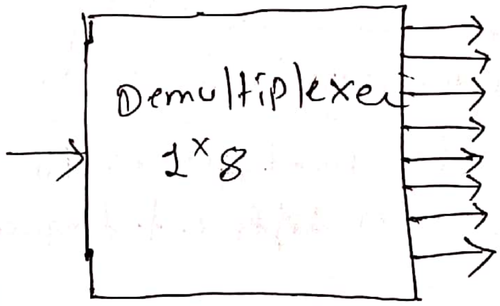
Time is not fixed so time taken by signals as per demand therefore, C has no data.

Difference between synchronous and Asynchronou Time Division multiplexing

Synchronous TDM	Asynchronous TDM
Time slots are fixed for each channel.	Time slots are given only when a channel has data.
May waste slots if a channel has no data (less efficiency).	No slots are wasted (more efficient).
simple and easy to implement	More complex due to dynamic slot assignment.
Needs strict synchronization	Needs synchronization but more flexible
Bandwidth may be under-utilized.	Bandwidth is better utilized
Eg Telephone systems	Example; computer networks, statistical multiplexing

Scanned with OKEN Scanner

Difference between Multiplexer and Demultiplexer

Multipler	Demultiplexer
Multiplexing is a process in which various data signals are integrated to give a single output.	Demultiplexing is a process in which one input data signal is divided into various output signals.
It takes various input signals and gives one output signal only.	It provides multiple output signals and takes one input signal only.
serial to parallel conversion is used in multiplexing	parallel to serial conversion is used in demultiplexers.
 	 
eg for 8 inputs there is one output, for 16 inputs there is 1 output, for 32 inputs there is 1 output.	eg for 1 input there are 8 outputs, for 1 input there are 16 outputs, for 1 input there are 32 outputs
Multiplexer is abbreviated as MUX.	The demultiplexer is abbreviated as DEMUX.
A multiplexer can be defined as a digital switch.	A Demultiplexer can be defined as a digital circuit.
The multiplexer works as a data selector	The demultiplexer works as a data distributor.

working of multiplexer is based on many to one principle.	working of demultiplexers is based on one too many principles.
TDM (Time Division Multi-plexer) uses a multiplexer at the sender's end.	TDM uses a demultiplexer at the receiver's end.
Data communication systems, Telephone Networks, Satellite systems to send many signals using one path	Used in receivers, switches, and routers to direct data.

Difference Between FDM and TDM

FDM	TDM
FDM stands for frequency Division multiplexing.	TDM stands for Time Division multiplexing.
FDM is Analog.	TDM is Digital.
frequency is divided.	Time is divided.
Needs modulators to assign different frequencies	Needs timers/synchronization to assign time slots.
Bandwidth is divided among users.	full bandwidth is given to each user in their time slot.
eg Radio & TV broadcasting	eg Computer Networks, Digital Telephony

3.3 Connectionless Transport : UDP

UDP stands for User Datagram protocol.
A way of sending data over the internet without checking for errors or delivery confirmation.

UDP is an alternative communications protocol to Transmission Control protocol (TCP) used primarily for establishing low-latency and loss-tolerating connections between applications on the internet.

Low Latency → fast data transimission
Loss Tolerance → No problem if a few packets are lost.

UDP is a transport layer protocol for client-server network applications. UDP uses simple transmission model but does not guarantee for relability, ordering and data integrity.
UDP is widely used for video calls or voice calls, online gaming, live streaming.

3.3.1 UDP Segment structure

UDP segment consists of 4 fields, each 2bytes (16 bits) long:

S.N.	field Name	size (bits)	Description
1	Source port	16	port number of the sender
2.	Destination port	16	port number of the receiver
3.	Length	16	Total length of the UDP segment including header & data.

Scanned with OKEN Scanner

2. Checksum 16 used for error-detecting of hea
and data.

←8 bytes→

Header	Data

Source port number 16 bits	Destination port number 16 bits
Total Length 16 bits	Checksum 16 bits

figure: User Datagram header format

Checksum is one of the error detecting technique. In checksum, data is divided into k segments each of m bits.

In sender's end, the segments are added using 1's complement arithmetic to get the sum. The sum is complemented to get the checksum.

· Now, checksum segment is sent along with data segment

At receiver's end, all received segment are added using 1's complement arithmetic to get sum. The sum is complemented. If the result is 0, then data is accepted. otherwise Rejected.

Scanned with OKEN Scanner

10011001	11100010	00100100	10000100

k = 4 segment
m = 8 bit

Sender side

```
   10011001
   11100010
 ----------
 1 01111011
         +1      carry add
 ----------
   01111100
 + 00100100   3rd seg
 ----------
   10100000
 + 10000100   4th seg
 ----------
 1 00100100
         +1
 ----------
   00100101   sum
   11011010   1's complement
              Checksum
```

Now sent checksum along with data segment

Receiver's end

```
   10011001
   11100010
 ----------
 1 01111011
         +1
 ----------
   01111100
   00100100
 ----------
   10100000
   10000100
 ----------
 1 00100100
```

00100101
11011010 Add checksum

11111111 1's complement

00000000 Now received data is accepted

Checksum Example of 2 data segments

[10101001 00111001]

Sender's side

10101001
+ 00111001

11100010 1's complement

00011101 Checksum

Receiver's End

10101001
00111001

11100010
+ 00011101 Add checksum

11111111 1's compleme

00000000 Now received data is accepted.

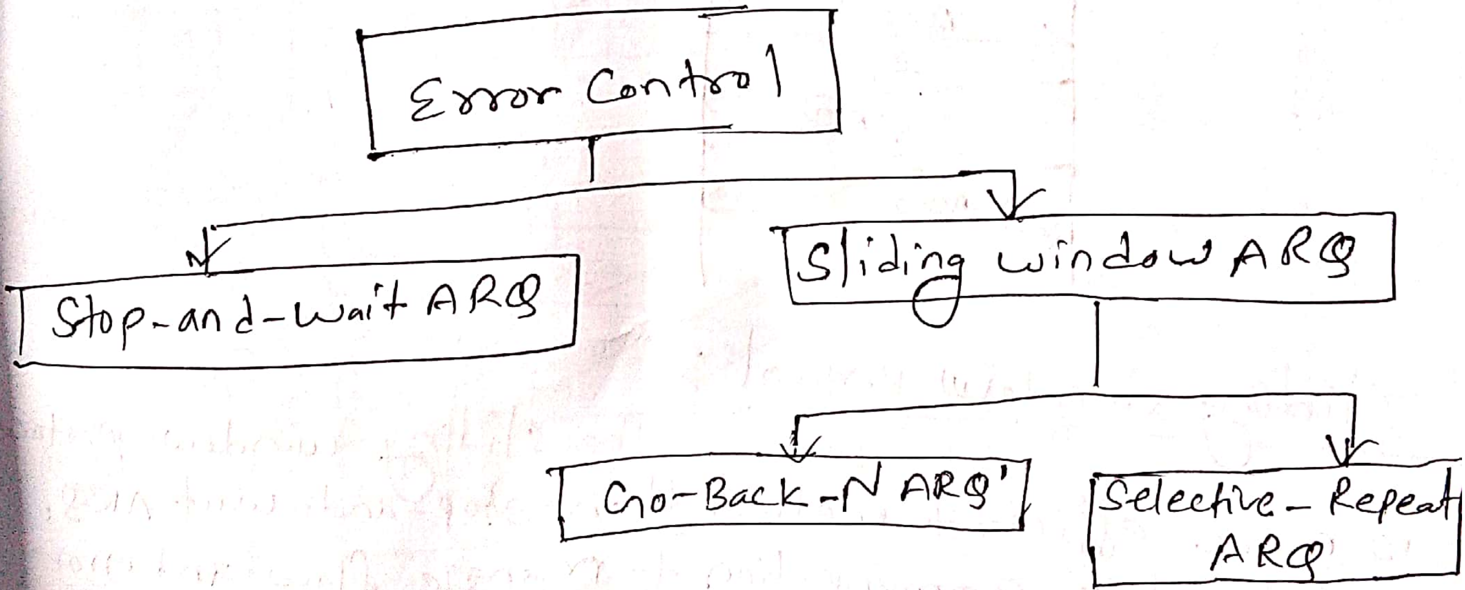
Scanned with OKEN Scanner

3.4 Principles of Reliable Data Transfer

Reliable data transfer ensures that data is delivered accurately, in order, and without loss or duplication over an unreliable network.

Error Control mechanism is followed such as

(i) Error detection (using checksum)
(ii) Positive Acknowledgement (ACKs)
(iii) Negative Acknowledgement (NACKs)
(iv) Retransmission
(v) Sequence maintaining or Reassembling

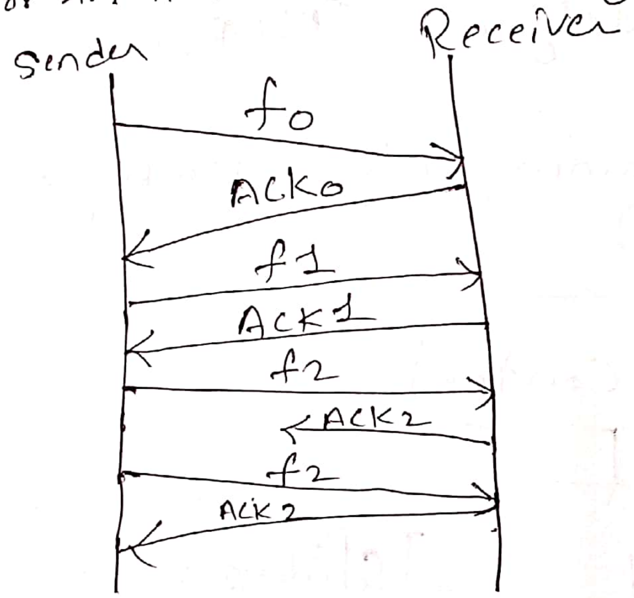


Stop-and-wait ARQ is one of the simplest error control protocols used in data communication. It ensures reliable data transfer by using acknowledgments (ACKs) and retransmissions.

Working principle of Stop and wait ARQ

1. Sender sends a data frame (packet) to the receiver.
2. Sender waits for an acknowledgment (ACK).
3. If ACK is received, it sends the next frame.
4. If ACK is not received within a certain time (timeou[t]) the sender resends the same frame.

"Stop & wait Send one frame at a time, and sender mu[st] wait for an ACK before sending the next frame."



Sliding window protocol :-

The sliding window proto[col] is more efficient method than stop-and-wait ARQ, used in data communication to manage flow and error control. It allows multiple frames to be sent before requiring an acknowledgment, making better use of network bandwidth.

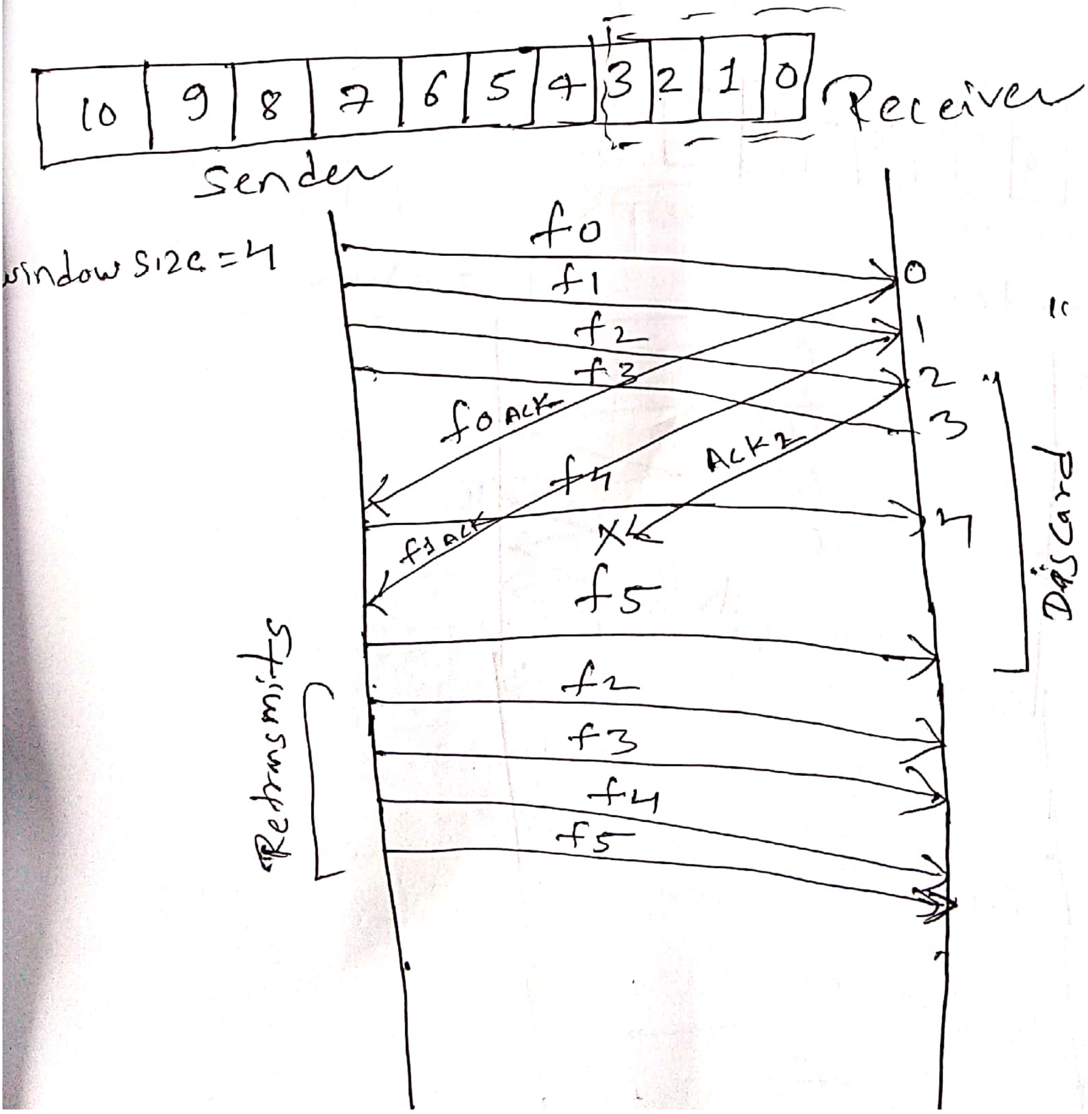
Working Mechanism :

→ Sender can send multiple frames up to a window size before needing an acknowledgment.

The receiver can receive multiple frames and send acknowledgments for them.
Both sender and receiver maintain a window (range of acceptable sequence numbers).

Types of sliding window protocols

Go-Back-N ARQ is an Automatic Repeat Request where sender can send N frames before needing an ACK. If any frame is lost or damaged, the receiver discards it and all subsequent frames. The sender then goes back and resends from that frame onward.



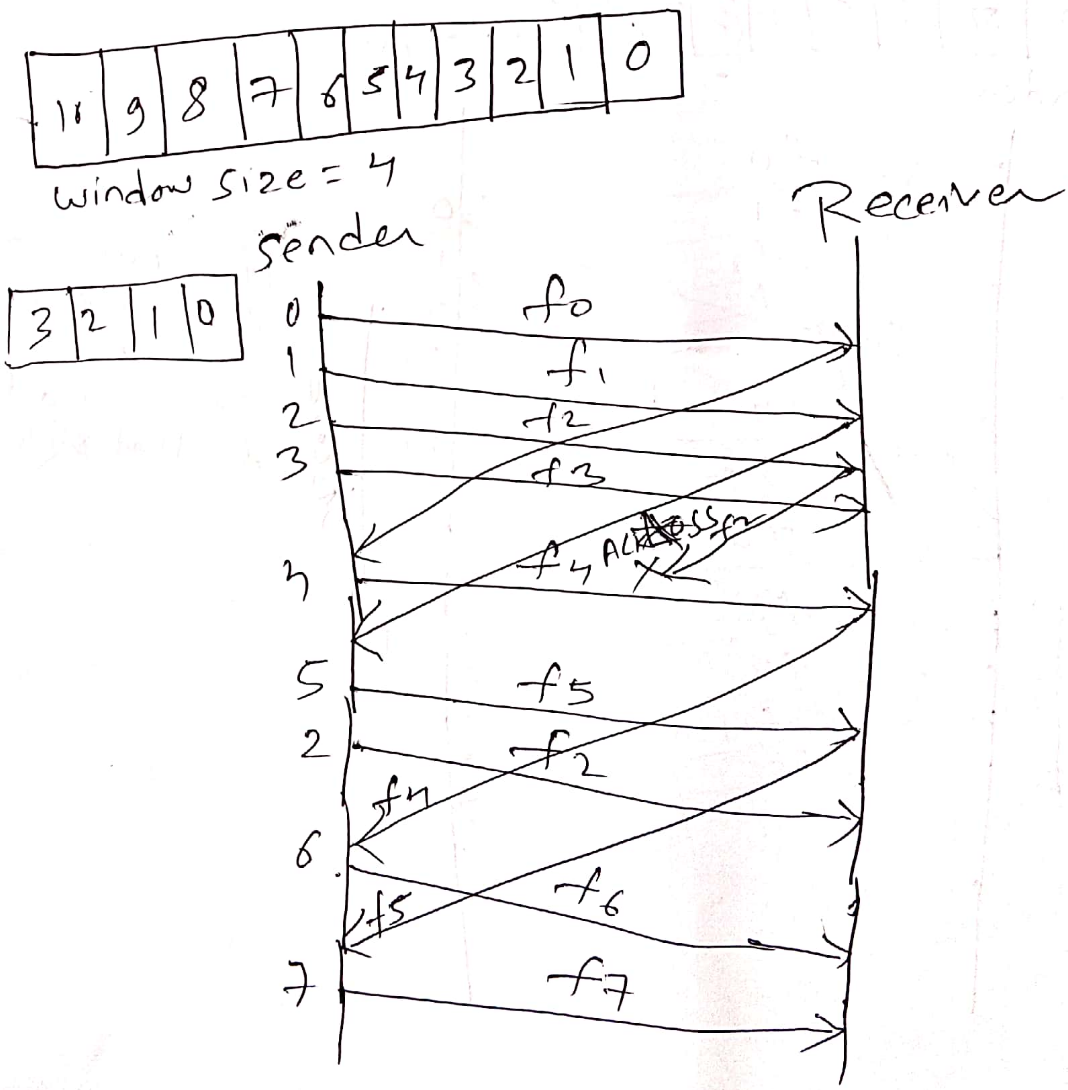
"if window size is eligible, then slide window & start send data for that slided window"

Selective Repeat ARQ :-

In selective Repeat ARQ, only the erroneous, lost frames are retransmitted, while correct frame are received and buffered.

The receiver while keeping track of sequence numbers, buffers the frames in memory and sends Negative AcknoWlegment for only frame which is missing or damaged.

The sender will send/retransmit packet for which NACK is received.



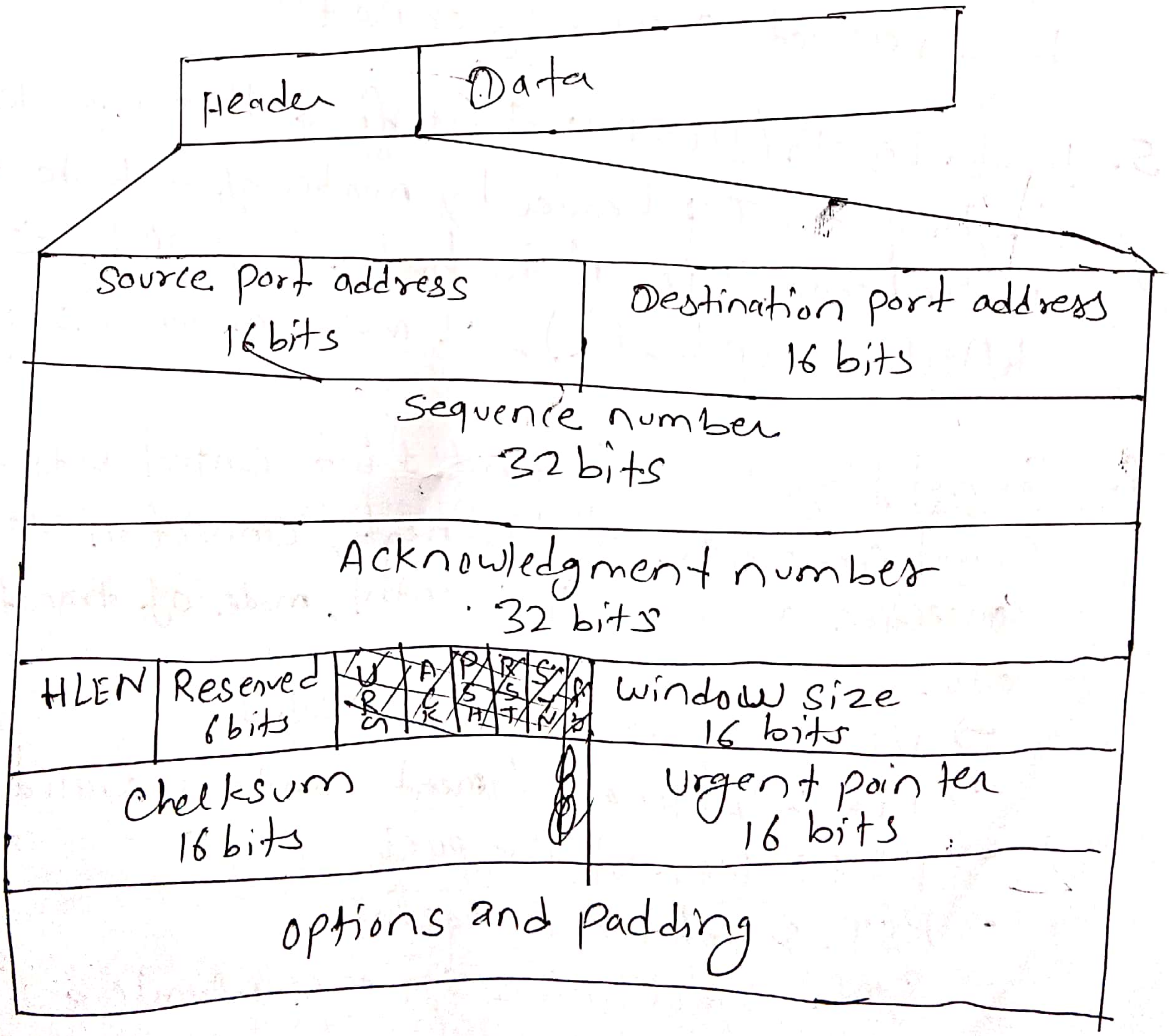
Scanned with OKEN Scanner

3.5 Connection - oriented Transport: TCP

TCP is a connection-oriented protocol, meaning that a connection must be established before data can be transmitted. This ensure reliable, ordered and error-checked delivery of data between applications.

TCP segment format

TCP segment consists of data bytes to be sent and a header that is added to the data by TCP as shown in figure.



Scanned with OKEN Scanner

1. Source port Address:- It is 16-bit field that holds the port address of the application that is sending the data segment.

2. Destination port Number: 16 bit field that holds the port address of the application in the host that is receiving the data segment.

3. Sequence Number: 32 bit field that holds the sequence number. It is used to reassemble the message at receiver end.

4. Acknowledgment Number: 32 bit field that holds the ack number. Information of of previously sent bytes being received successfully or not.

5. Header Length (HLEN): 4 bit field that indicate the length of the TCP header by number of 4-byte words in the header, i.e. if the header is of 20 bytes (min length of TCP header), and maximum length: 60 bytes.

6. Control flags: six different 1 bit control bits that control connection establishment, connection termination, connection abortion, flow control, mode of transfer rate etc.
   → URG:- Urgent pointer is valid
   → ACK :- Acknowledgment number is valid
   → PSH :- Request for push
   → RST: Reset the connection
   → SYN: Synchronize sequence Numbers
   → FIN: Terminate the connection.

7. Window size: window size of the sending TCP in bytes.
8. Checksum: checksum for error control.

Scanned with OKEN Scanner

9. urgent pointer: 16 bits field is used to point to data that is urgently required that needs to reach the receiving process at the earliest. The value of this field is added to the sequence number to get the byte number of the last urgent byte.

10. options: There can be up-to 40 bytes of optional information in the TCP header.

3.5.1 Round-Trip Time Estimation and Timeout

RTT means the total time taken for a message to go from the sender to receiver and then for the acknowledgment to come back.

TCP measure RTT to decide how long to wait before thinking a message is lost and needs to be resent (this waiting time is called timeout).

If the time out is too short, TCP may resend the packet unnecessarily.
If the timeout is too long, TCP will take too much time to recover when a packet is acutally lost.

"TCP estimates RTT to set an appropriate timeout for retransmissions).

3.5.2 Reliable Data Transfer

TCP guarantees reliable delivery using:

1. Acknowledgments (ACKS): receiver confirms data received.
2. Retransmissions: TCP resends the data, if no ACK is received.
3. Checksums: Error control to detect corrupted data.
4. Sequence numbers: To deliver packets in order.

Scanned with OKEN Scanner

3.5.3 flow Control

flow control ensures that sender does not send too much data too fast for the receiver to handle.

TCP uses a method Called Sliding window protocol:

- The receiver tells the sender how much free Space (window size) it has:
- The sender can send that much data without waiting for further acknowledgment.
- If the receiver's buffer is full, it can set the window size to zero, telling the sender to pause or stop.

Thus, flow control matches the sender's speed with the receiver's ability to process data.

3.6 Congestion Control in TCP

managing the amount of data sent into the network So that network doesn't get overloaded is called Congestion control.

Congestion control prevents from

(i) delays
(ii) packet loss.

Imagine - Many cars on a Road, if too many come at once, traffic jams happen. similarly, if too much data is sent over a network, it gets congested.

There are congestion control techniques in TCP.

1. slow start
2. Congestion Avoidance
3. fast Retransmit
4. fast recovery.

Scanned with OKEN Scanner

1. Slow Start:

At first, TCP sends a small amount of data. if it works well (no loss), it doubles the amount next time. This helps TCP quickly find how much the network can handle.

2. Congestion Avoidance:

when TCP feels that it's sending enough data, it slows down the growth. Instead of doubling, it increase window size slowly

3. fast Retransmit: If TCP receives 3 duplicate ACKs (packet lost), it retransmits the missing packet immediately. without waiting for time out.

4. fast Recovery:

After Resending, TCP does not start from the begining. It lowers the sending rate a little and keeps going.

Scanned with OKEN Scanner