CSL-255

Wednesday, January 5, 2022 10:12 AM

05|01|2022

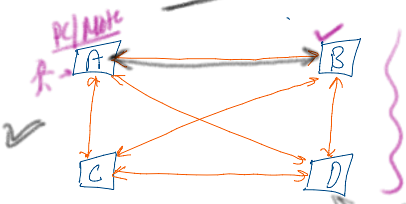
Computer Networks

communication

send|Receive

✓Topology (Mess)

PC|Mobile

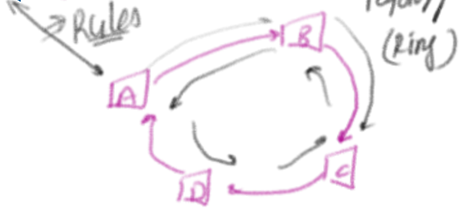


N/w,

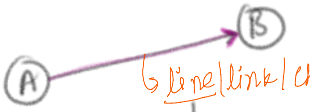
Network / Group

Rules

Topology (Ring)



N/w 2



↳ line|link|channel|medium

- wired
- wireless

Topology (Bus)

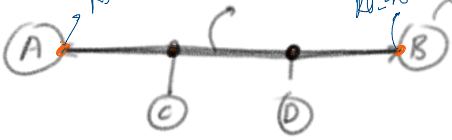
N/w 3

line-configuration :⇒

cross- CAT5|6 ↳ RJ45

- Point-to-Point ↳ CAT5|6
- multipoint/shared ✓ any one | CAT5-6 | optical | fiber | coaxial

shared medium|link



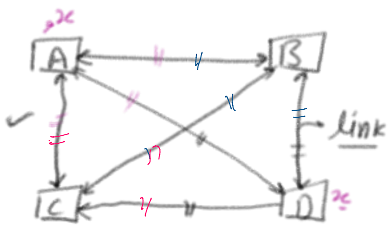
Point-to-Point

Note:- for communication, each device will use some pre-defined set of Rules called Protocol →
- OSI- Reference model → (7-layers)
- TCP/UDP protocols → 4/5 layers

Topology :⇒ Topologies defines the Physical or logical arrangement of links in a N/w.

- Mess ✓
- Star ✓
- Tree ✓
- Bus ✓
- Ring ✓
- Hybrid

* Mess Topology :⇒ in a mesh Topology, every device has a dedicated Point-to-Point link to every other device.



n = Devices ⇒ 04 ✓

l = links → 06 ✓

$l = \frac{n(n-1)}{}$

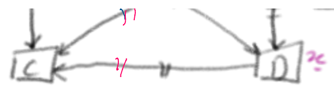
l = links = 06

$n = 100$

$l = \frac{100 \times 99}{2} \approx 5000$

$$\boxed{l = \frac{n(n-1)}{2}}$$

$$\boxed{l = \frac{4 \times 3}{2} = 06}$$



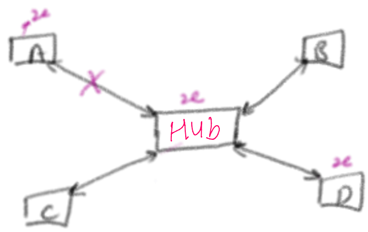
Small networks

$n = 1000$, $l = 50,000$

$n = 10000$, $l = 5000000$

* it is good for small N/ws but not for larger n/ws due to link failure/handle.

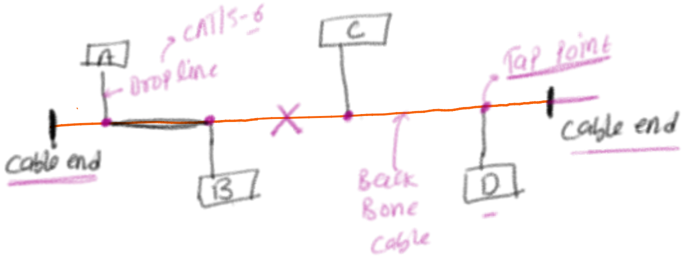
Star Topology :⇒ in star topology, each device has a dedicated point-to-point link only to a central controller, usually called a HUB.



Note:- ① Single point of failure (HUB)
② Bottleneck problem (HUB)

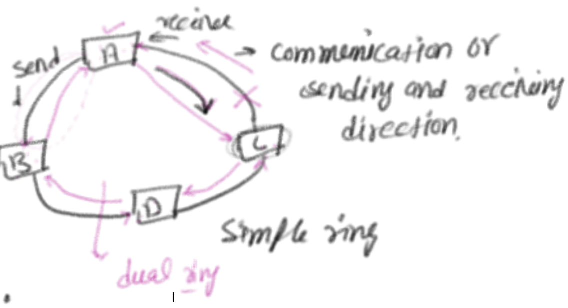
07/01/2022

Bus Topology :⇒ all the devices are in point-to-point configuration through multipoint or shared link/line.



Note: ① if the number of devices are increased then configuration becomes difficult.
② Back bone cable failures

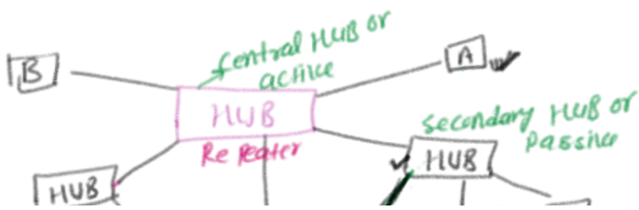
Ring Topology :⇒ in ring topology, each device has a dedicated point-to-point line configuration only with two devices on either side of it.



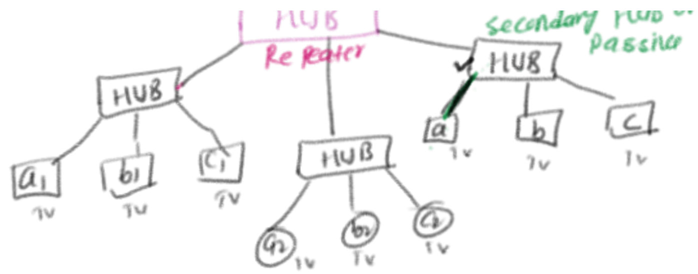
Note:- ① unidirectional traffic can be a disadvantage.
② in simple ring, a break in the ring (such as a disabled devices) can disable the entire N/w.

Tree Topology :⇒ it is a variation of star topology and nodes (devices) are linked to a central HUB that controls the traffic to the n/w.

* active HUB, contains a repeater, which is a H/w device that



* active HUB, contains a repeater, which is a H/w device that regenerates the received bit pattern or signal before sending them to the device or passive HUB.



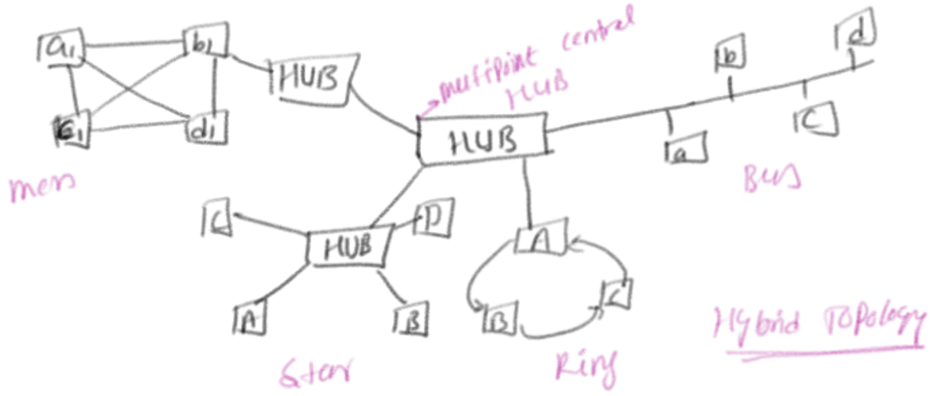
* Passive HUB, it provides a simple physical (point-to-point or dedicated link) connection b/w the attached devices.

* the advantage and disadvantages are generally the same as star topology.

* Example ⇒ TV cable connection.

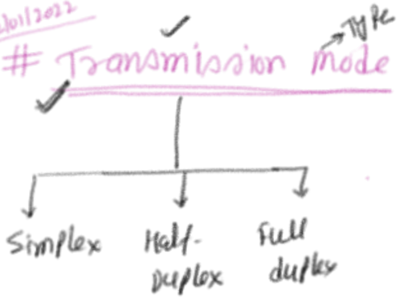
Hybrid Topology :⇒ Combination of more than one topology is called Hybrid Topology.

(MAN)



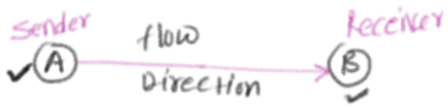
12/01/2022

Transmission mode :⇒



* simplex :→ in the simplex transmission mode, the communication is unidirectional.

Ex. one-way street

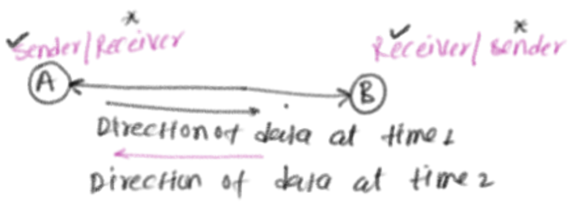


Mainframe — monitor, Printer

KeyBoard — mainframe / CPU

* Half-Duplex :⇒ in Half-duplex, each station can both transmit and Receive, But Not at the same time.

Ex. walkie-talkies

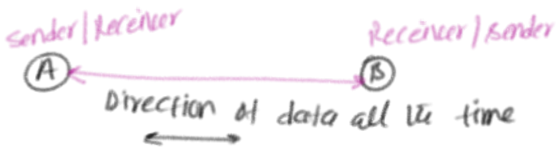


$\Delta t = 1$ sec.
A → sender
B → Receiver
$\Delta t = 1.5/02$ sec.

Direction of data at time 2

At = 1.5/02 sec.
A → Receiver
B → Sender

* **Full-Duplex** :⇒ it is also called duplex. Both stations can transmit and receive simultaneously.

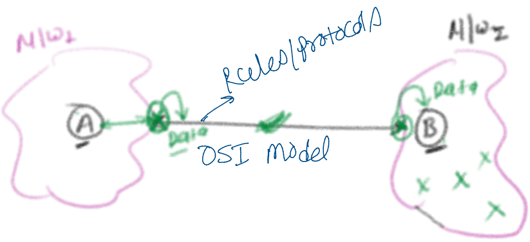


Ex: Full-duplex communication in the Telephone N/w.

Networks Type :⇒

- LAN (local Area N/w) → single Building
- MAN (Metropolitan Area N/w) multiple Building
- WAN (Wide Area N/w) ⇒ public city N/w

The OSI Reference model :⇒ Open Systems interconnection (OSI)

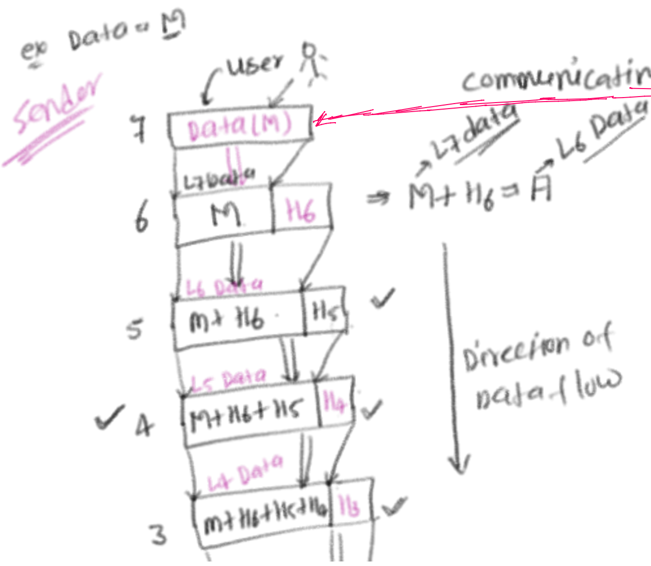


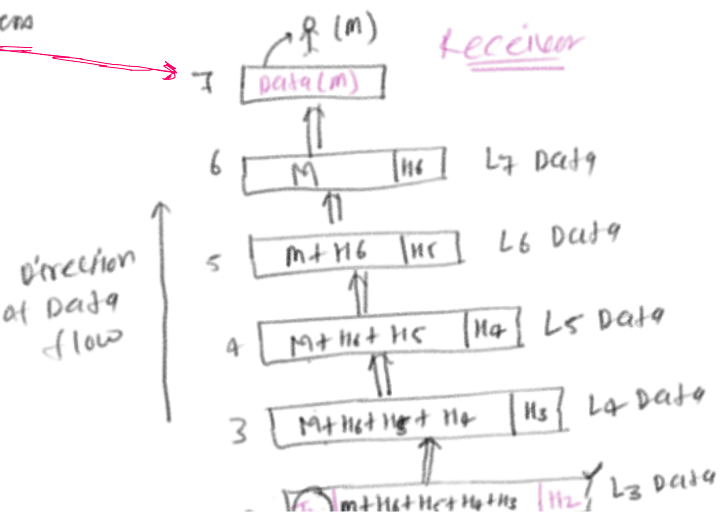
* OSI Model has layered architecture

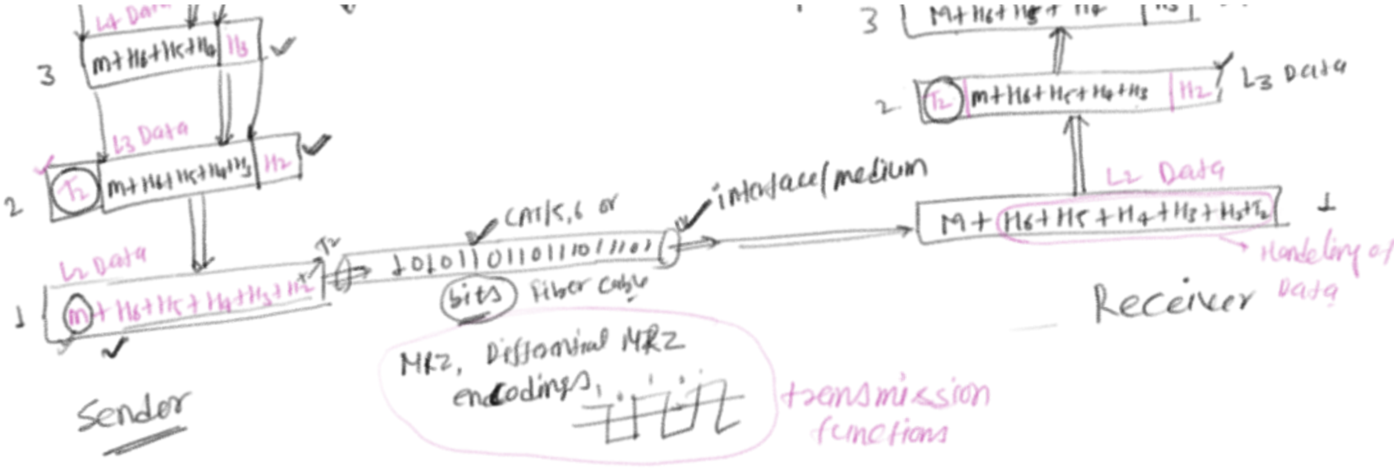
Physical layer :⇒

* Data exchange using OSI model ✓

ex Data = M







Physical layer(1) :⇒ the physical layer coordinates the functions required to transmit a bit stream over a physical medium.

* it deals with mechanical and electrical specifications of the interface.

* functions of physical layer are as follows

⇒ Physical characteristics of interface and medium

⇒ Bit representation ✓

⇒ Data Rate ⇒ Transmission rate, No. of bits per second.

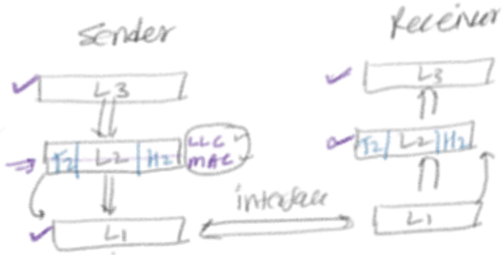
⇒ Synchronization of bits

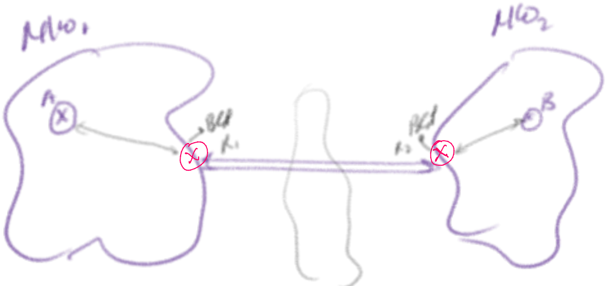
⇒ line configuration ⟶ Eg (A) ⟷ (B) point-to-point or shared/multipoint

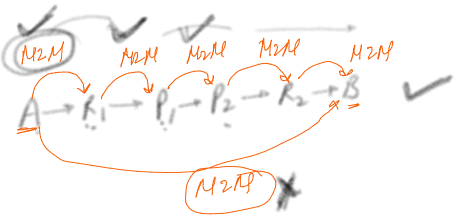
⇒ physical Topology

⇒ Transmission mode

Data link layer (2) :⇒ it provide Node-to-Node delivery.





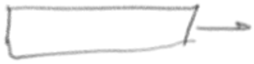


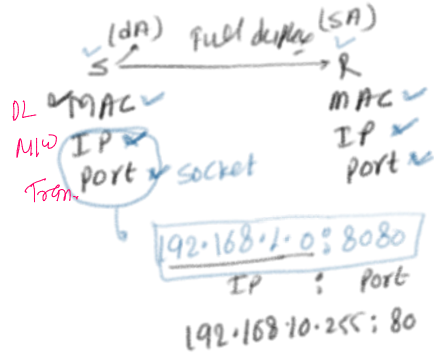
⇒ Framing :⇒ the data link layer divides the stream of bits received from N/W layer into manageable data unit called frames.

⇒ physical Addressing :⇒ NIC (Ethernet, 802, 802.3, 802.4, 802.5 . . .)
↑
RJ45
802.11, a, b, g, n . . .
(DA) . . . (SA)

⇒ Physical Addressing :⇒ MAC (Ethernet, 802, 802.3, ...)
RJ45
802.11, a, b, g, n ...

⇒ Flow control :⇒ Stop & wait Protocol, sliding window Protocol





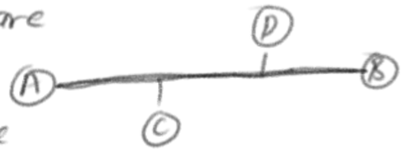
⇒ **Error control** :⇒ the data link layer adds reliability to the physical layer by adding mechanisms to detect and re-transmit damaged or lost frames.

* It also uses a mechanism to prevent duplication of frames.
* Error control is normally achieved by a trailer address added to the end of the frame.

⇒ Stop-&-wait ARQ, sliding window ARQ → Go-Back-n ARQ / selective-Reject ARQ

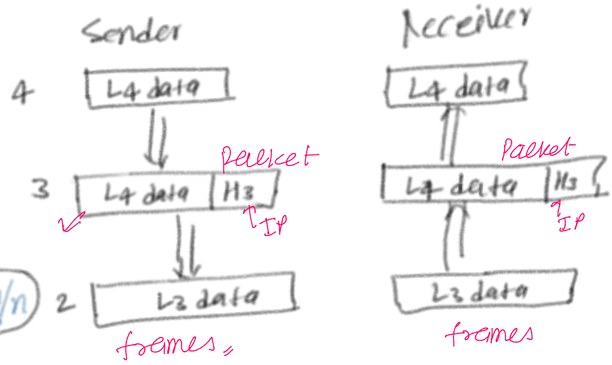
⇒ **Access control** :⇒ When two or more devices are connected to the same link, data link layer protocols are necessary to determine which device has control over the link at any give time.

shared link



Network layer :⇒

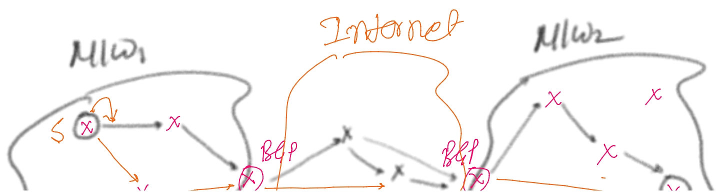
⇒ **Packets** :⇒ manageable unit of data by N/w layer called packet.



⇒ **logical Addressing (IP)** :⇒

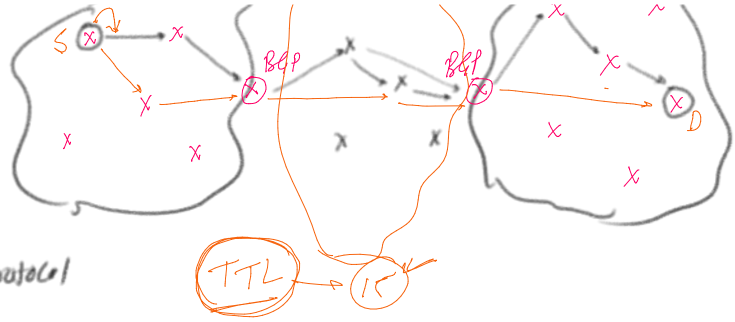
- classfull — 192.168.1.0 or 10.0.0.1
  - class A, class B — Public Addressing
  - class C — Private Addressing
  - class D — Broad/multicast Addressing
  - class E (Reserved)
- classless — 192.168.1.0/n

⇒ **Routing** :⇒ it is used to findout a smallest path



Findout a smallest path B/W source and Destination

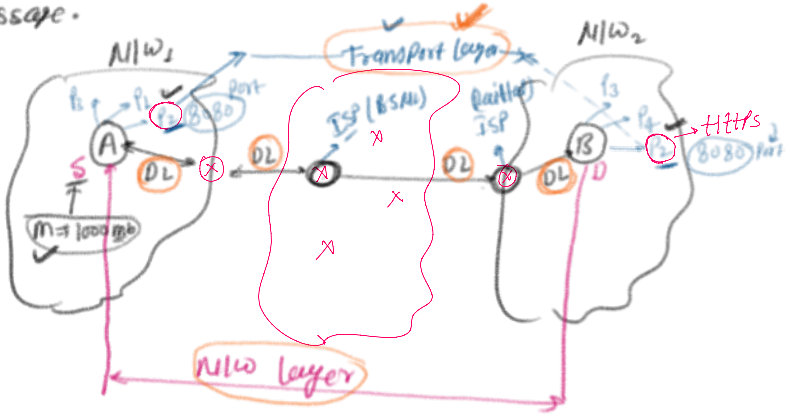
- DVR Algo → RIP Protocol
- OSPF Algo → Link-state Protocol



Transport layer :⇒ The Transport layer is responsible for source-to-destination (end-to-end) delivery of entire message.

P-2-P

Data link layer :	Node-to-Node	S → D
N/w layer	: Host-to-Host	S → D
Transport layer	: Process-to-Process	S/P → D/P

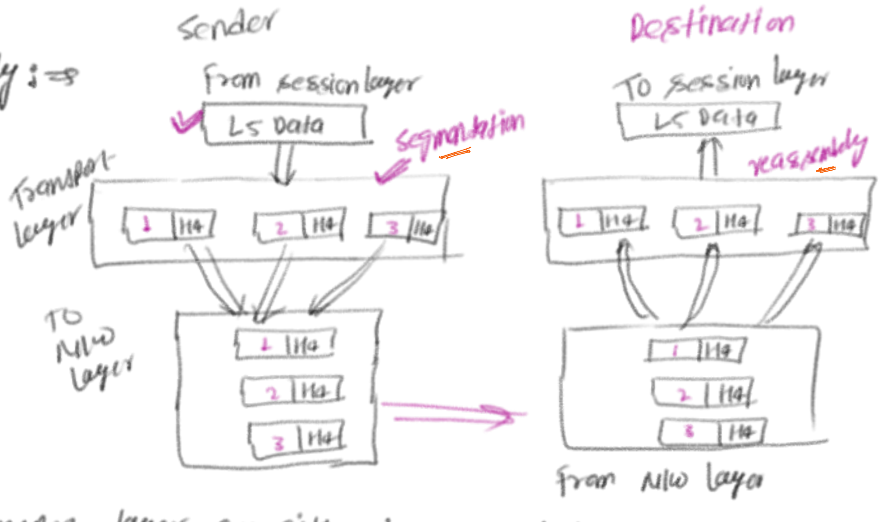


Service-point Addressing (Port Address) :⇒ Transport layer header must include a type of address called a service-point-Address (Port Address) for specific process of a node to the specific process of another node.

⇒ Segmentation and reassembly :⇒

Ex 10 ⇒ process No. or segment No.

1/10
2/10
3/10
4/10
⋮
10/10



⇒ Connection Control:⇒ The transport layer can either be connectionless or connection oriented. Connectionless transport layer treats each segment as an independent packet and delivers to the transport layer at destination machine. while,

- A connection oriented transport layer makes a connection with transport layer at destination machine first, before delivering the packets.

⇒ Flow Control:⇒ like data link layer, the transport layer is responsible for flow control. However, flow control at this layer is performed end-to-end rather than across a single link.

⇒ Error Control:⇒ In case of transport layer makes sure that the entire

⇒ Error control :⇒ The sending transport layer makes sure that the entire message arrives at the receiving transport layer without error (damage, loss, or duplication).

* Error correction is achieved through Re-transmission

Session layer :⇒

* The session layer is the Network dialog controller.
* It establishes, maintains, and synchronizes the interaction b/w communicating systems. (→ Terminates)

× Dialog control :⇒ The session layer allows two systems to enter into a dialog.

* It allows the communication b/w two processes to take place either in Half duplex or full duplex.

× connection establishment :⇒

Handshake Protocols → Needs for connection Establish or End / Terminate

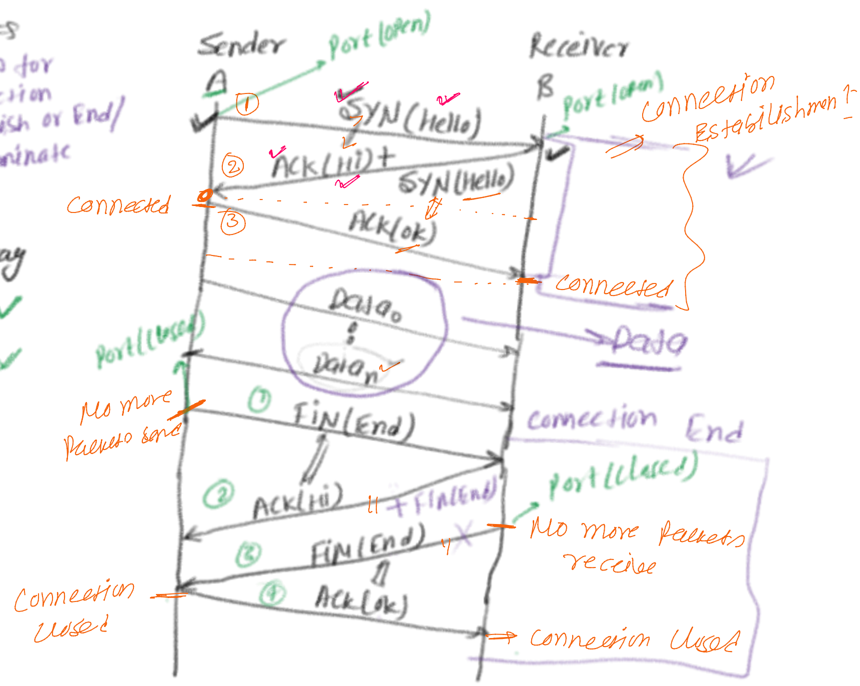
A —(Hi)→ B

2-way | 3-way | 4-way | n-way

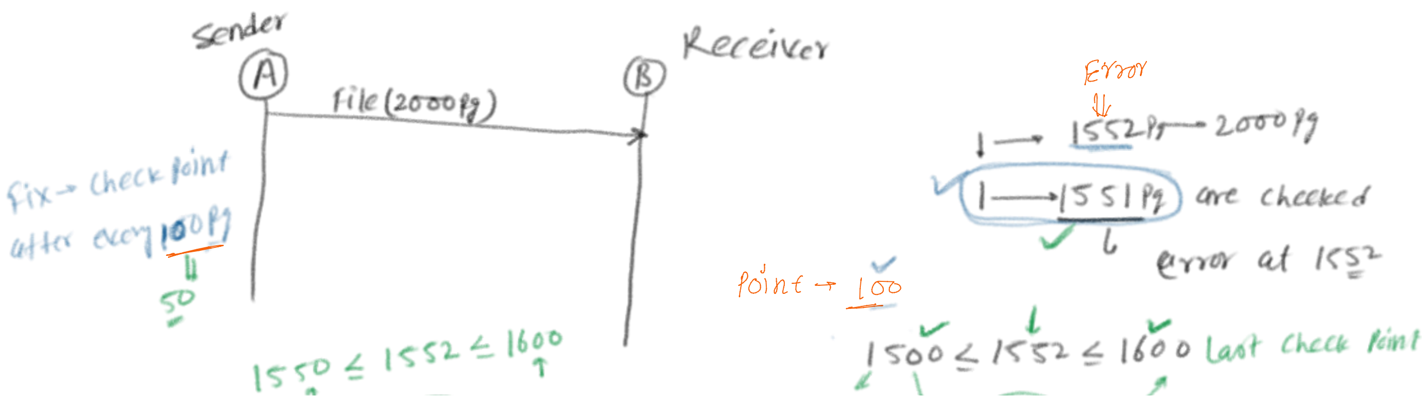
flow control mechanism

× Active open
× passive open

* Half - open
* Full - open
* Half - close
* Full - close



× Synchronization :⇒ The session layer allows a process to add check points (synchronization points) into a stream of data.



$1550 \le 1552 \le 1600$

50 pgs

$1500 \le 1552 \le 1600$ Last check point

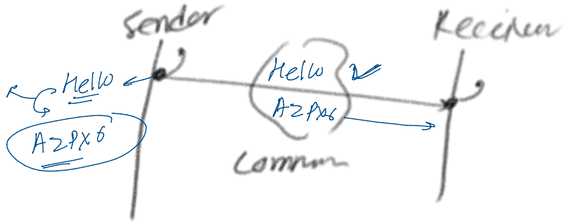
Pre. check point

100 only to check 100 pages.

Presentation layer :- it is concerned with the syntax and semantics of information exchanged b/w two communicating systems.

cryptography

- Encoding of Data / De-coding data
- Encryption / Decryption of Data
- Compressed data

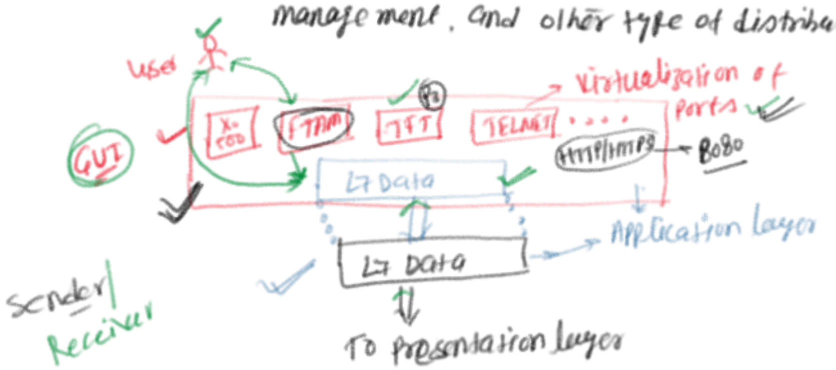


- Translation :-
- Encryption :-
- Compressed :-

Application layer :- The application layer enables the user, whether human or software, to access the Network.

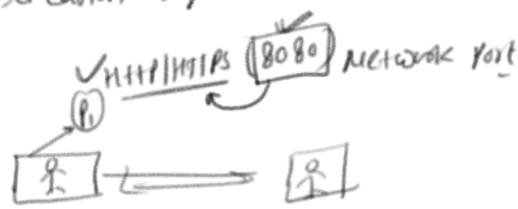
* it provides user interfaces and support for services such as electronic mail, remote file access and transfer, shared database management, and other type of distributed information services.

Ex

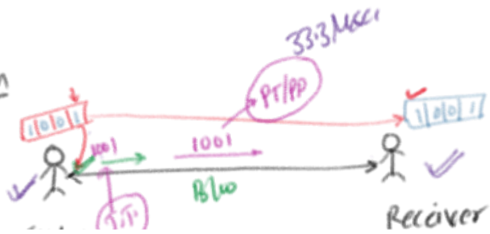


* Network virtual terminal :-
* mail service :- it provides the basis for email forwarding and storage. → POP3 | SMTP

and more services are offered by Application layer.



Examples



Propagation Time / delay = $\frac{\text{Distance}}{\text{Speed}}$

Transmission Time = $\frac{\text{message size}}{}$

Propagation Time / delay = $\frac{\text{Distance}}{\text{Speed}}$

Transmission Time = $\frac{\text{message size}}{\text{Bandwidth}}$

Ex① if the message size is 1 MB and it is passed in a B/W of 1 GbPs, then calculate the Transmission Time.

Soln.

$$\text{Transmission Time (TT)} = \frac{\text{message size}}{\text{Bandwidth}} \Rightarrow \frac{1 \times 8 \times 10^{6} \text{ bits}}{1 \times 10^{9} \text{ bits}} \text{ sec.}$$

B = Byte
b = bits

$$T.T. = 8 \times 10^{-3} \text{ sec. or}$$

$$\boxed{T.T. = 08 \text{ msec.}}$$

Ex② if the distance b/w sender and receiver is 05 km and data speed is 50% the speed of light; then calculate the propagation time.

Soln

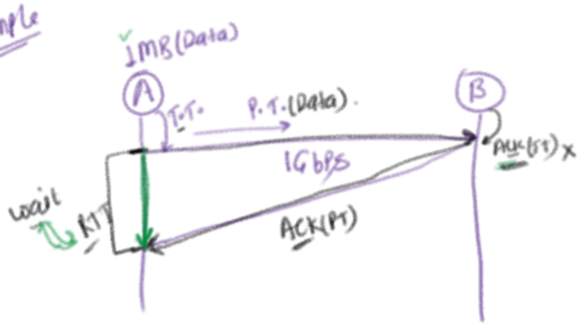
$$\text{Propagation time / delay} \Rightarrow \frac{\text{Distance}}{\text{Speed}}$$

$$PT = \frac{5 \times 10^{3} \text{ m}}{\frac{1}{2} \times 3 \times 10^{8} \text{ m}} \text{ sec}$$

$$PT = 3.33 \times 10^{-5} \text{ sec.} \quad \text{or} \quad \boxed{PT = 33.3 \ \mu\text{sec.}}$$

Total time = T.T. + P.T.
= 08 msec + 33.3 µsec.

*Example



$$\text{Total time} \Rightarrow TT_{(Data)} + PT_{(Data)} + T.T_{(ACK)} + PT_{(ACK)}$$

$$TT_{(ACK)} = \frac{ACK(\text{size})}{1 \text{ GbPs}} \Rightarrow \frac{0.0001 \text{ MB}}{1 \text{ GbPs}} \simeq 0 \text{ sec. or can be ignored}$$

× ACK message size is very very less

$$\boxed{\text{Total time} = TT_{(Data)} + PT_{(Data)} + PT_{(ACK)}}$$

or

$$\boxed{\text{Total time} = TT_{(Data)} + 2 \times PT_{(Data, ACK)}} \Rightarrow RTT$$

$$\boxed{\text{Round trip time} = 2 \times PT}$$

$$\boxed{\text{link utilization} = \frac{\text{Transmission time}}{\text{Total time}}} \quad \text{or} \quad \boxed{\mu = \frac{TT_{(Data)}}{TT_{(Data)} + 2 \times PT}}$$

Ex③ if the link utilization of the sender is 50% and PT is 50 msec. then calculate the transmission time?

Soln

$$\mu = \frac{TT_{(Data)}}{TT_{(Data)} + 2 \times PT} \Rightarrow \mu = \frac{x}{x + 2 \times PT} \Rightarrow \frac{1}{2} = \frac{x}{x + 2 \times 50}$$

Sol^n

$$U = \frac{TT_{(Data)}}{TT_{(Data)} + 2 \times PT} \Rightarrow U = \frac{x}{x + 2 \times PT} \Rightarrow \frac{1}{2} = \frac{x}{x + 2 \times 50}$$

$$\Rightarrow \frac{1}{2} = \frac{x}{x + 100}$$

Assume $TT_{(Data)} = x$

$$\boxed{TT_{(Data)} = 100\,msec.}$$

$$\Rightarrow 2x = x + 100$$

$$\Rightarrow \boxed{x = 100}$$

or $TT_{(Data)} = 2 \times PT$
$= 2 \times 50\,msec.$
$\boxed{TT_{(Data)} = 100\,msec.}$ } only for analysis

Ex④ If the Transmission time is 20 msec. and Round trip time is 40 msec., then calculate utilization of link by sender. If the link bandwidth is 1 Gbps then message size?

Sol^n

$$U = \frac{TT}{TT + \underbrace{2 \times PT}_{RTT}} \Rightarrow U = \frac{20\,msec}{20\,msec + 40\,msec.}$$

$$\Rightarrow U = \frac{20}{60} \rightarrow 1/3 \quad \text{and} \quad \boxed{2/3 \text{ is remain free}}$$

or $\boxed{U = 33.33\%}$

* $TT = \dfrac{\text{Message size}}{\text{Bandwidth}}$

$$20 \times 10^{-3}\,sec = \frac{x}{1 \times 10^{9}\,bits}\,sec. \Rightarrow x = 20 \times 10^{9} \times 10^{-3}\,bits$$

$\boxed{x = 20 \times 10^{6}\ bits}$ or $\boxed{\text{message size} = 20\,Mb}$

Error Detection and correction :⇒

Data can be corrupted during transmission. For reliable communication, error must be detected and corrected.

* The Error detection and correction are implemented either at the data link layer or the transport layer of the OSI reference model.

* Type of Errors :⇒
  - → Single-bit-Error
  - → Burst-bit-Error

bit form

Data ⇒ 10101101010101 → Error
1010001 10101 ✗

* Single-bit-Error :⇒ The term single-bit-Error means that only one bit of given data unit such as a byte, bits, char, packet, etc. is changed from 1 to 0 or from 0 to 1.

Ex.

10010101 (sent) $\xrightarrow{\text{Transmit}}$ 10011101 (received) — 0 to 1

$\boxed{\text{0 changed to 1}}$

* Burst-bit-Error :⇒ The term Burst-bit-Error means that Two or more bits

0 changed to 1

* **Burst-bit-Error:⇒** The term Burst-bit-Error means that Two or more bits in the data unit have changed from 1 to 0 or from 0 to 1.

Ex 11010101 $\xrightarrow{\text{comparing}}$ 10101110 / 10010001

sent — received — Two bits are changed

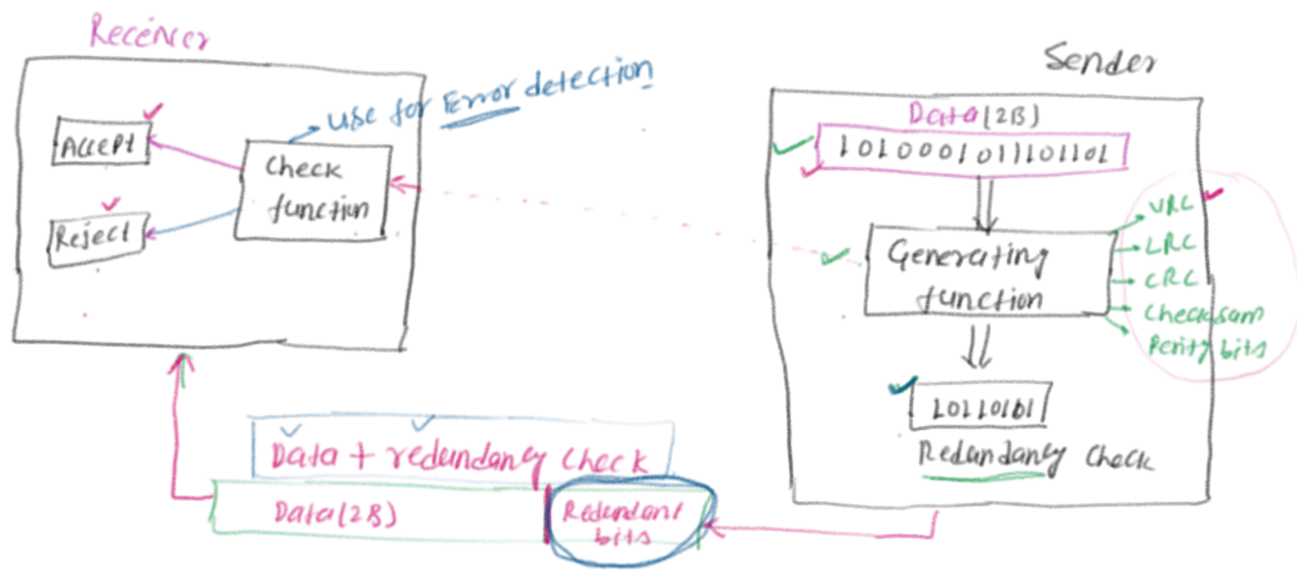
more bits are changed

Error Detection :⇒ one error detection mechanism that would satisfy these requirements would be to send every data unit twice. The receiving device would then be able to do a bit-for-bit comparison b/w two versions of the data.

Redundancy :⇒ A shorter group of bits may be appended to the end of each unit. This technique is called redundency because of the extra bits are redundant to the information.

* Error detection uses the concept of redundancy, which mean adding Extra bits for detecting errors at the destination, and these bits are called redundant bits.

Ex



* Four types of redundancy checks are used with in information (data).

Detection methods

- VRC
- LRC (1-D, 2-D)
- CRC
- checksum

VRC (Vertical Redundancy check) :⇒ it is also called a Parity check.

VRC (Vertical Redundancy check) :⇒ it is also called a Parity check.

Parity check → Even, ODD

in VRC, a parity bit is added to every data unit so that the total number of 1s becomes even.

Ex

Data ⇒ 1100001

Sender → Even Parity Generator → 1

1100001 | 1 — Data + Parity → Checking function: is the total No. of 1s even? — yes → Accept; NO → Reject

11000011

1100001(1) ×

1110001 1 ×

1000001 1 ×

11 0(1)0(1)1(1)

* VRC is a single bit parity check function.

Ex② Data: 1010101 ⟶ 1010101 | 0 ⟶ Receiver: 1010100 0 × ODD 1s

Note :⇒ So, if total No. of 1s in data unit are—

Even Parity check	ODD parity check
Even → 0 (Parity)	Even → 1 (Parity)
ODD → 1 (Parity)	ODD → 0 (Parity)

Ex Assume that sender wants to send the word "World". The chars are coded as ASCII code.

Soln

Sender: World →

w	o	r	l	d
1110111 0	1101111 0	1110010 0	1101100 0	1100100 1
1B	1B	1B	1B	1B

Receiver ← w, o, r, l, d } for each data unit (char), even parity check is generated and check function is used for acceptance of bits at destination.

Note: VRC can detect all single-bit errors. It can also detect burst errors only if the total number of errors in each data unit is odd.

LRC (Longitudinal Redundancy check) :⇒ in LRC, a block of bits is

LRC (longitudinal Redundancy check) :⇒ in LRC, a block of bits is organized in a tabular form (rows and columns).

1B

⇒ in LRC, a block of bits is divided into rows and a redundant row of bits is added to the whole Block.

Ex original Data ⇒ 11100111 11011101 00111001 10101001 + 10101010

1 Block, 2 Block, 3 Block, 4 Block, 5 Block (LRC)

1-Dimensional parity check function

Rows →	11100111	1B
colums ↓	11011101	1B
	00111001	1B
	10101001	1B
sender END →	10101010	LRC
Receiver END ←	00000000	All zero in check function, ⇒ No Error

if any bit in check function are Non-zero ⇒ Error

original data + LRC

Transmission data

2-Dimensional Parity check function :⇒ similar to LRC, but in 2-D

Ex original Data → 11100111 11011101 00110001 10101001

R-column → Redundant Column

R-Row → Redundant Row

		R-column	
Rows ←	11100111	0	1st Block
columns ↓	11011101	0	2nd Block
	00110001	1	3rd Block
	10101001	0	4th Block
2-D Parity check function	10100010	1	5th Block
	R-Row		

Rows → (Not columns)

New Data ⇒ for Transmission: 111001110 (1 Block), 110111010 (2 Block), 001100011 (3 Block), 101010010 (4 Block), 101000101 (5 Block)

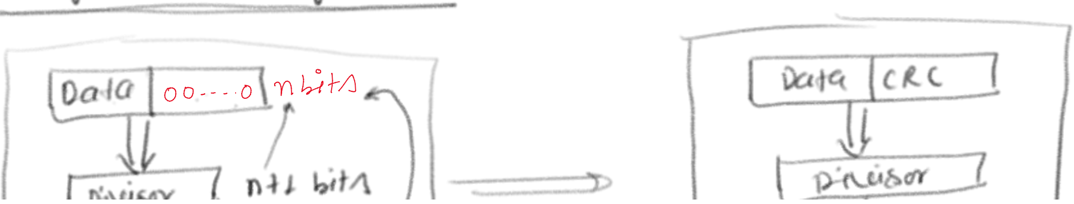
for Error Checking

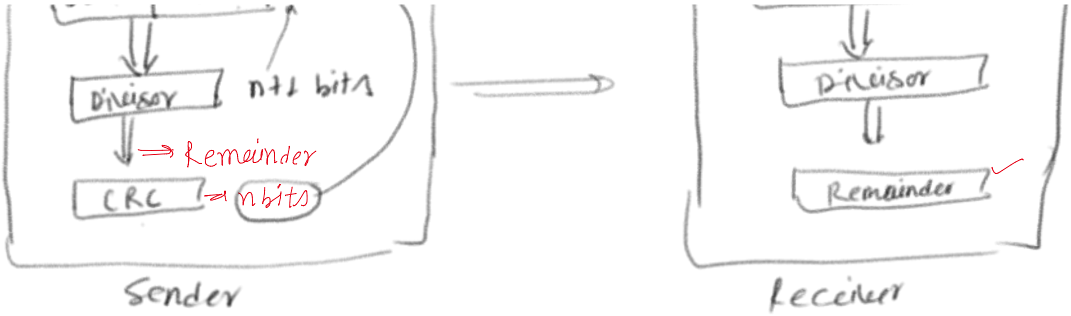
11100111	0
11011101	0
00110001	1
10101001	0
10100010	1
00000000	0

✓ check bits of 2D Parity function

⇒ in 2D Parity check function, a block of bits is divided in rows and the redundant row and column of bits are added to the whole block.

CRC (cyclic redundancy check) :⇒ CRC, is based on binary division.





Divisor
n+1 bits
Remainder
CRC
n bits
Sender
Divisor
Remainder
Receiver