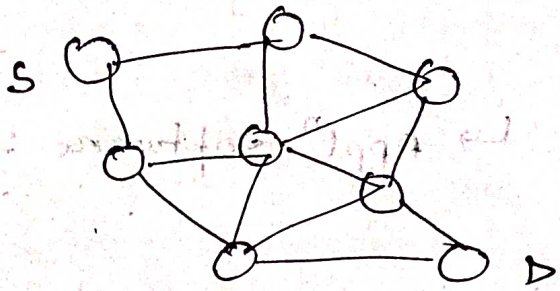
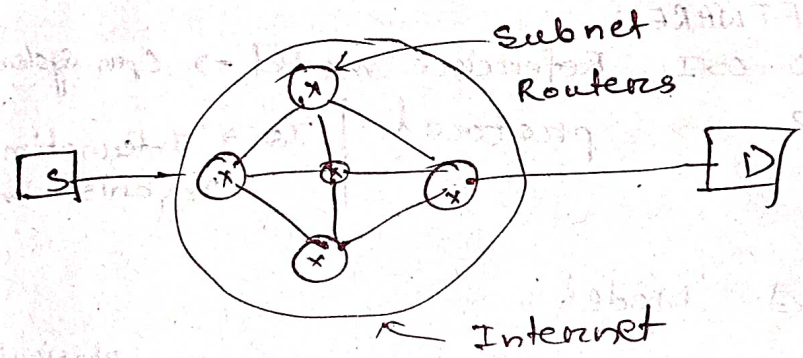


- **Routing**

  IP Address

↳ Next node becomes as source and decides to which neighbour node it should forward the message, so as to reach the destination with shortest path.



↳ We do not have any control over the internet.

- **Internet Components :-**

(i) Physical medium (STP, Coex, FO) — wired

Wifi, WiMax, BT — wireless

(ii) Interconnecting devices :
  → for long distance
  ↳ connecting different type of wires.

↳ Hub, switch, Router, Gateway,

- Fireworld → block unintended packets.
- Gateways → allows / blocks some packets

③ Computers: (PC, Laptop.
↳ connecting devices

(4) Network Software: (TCP/IP)

↳ App(n software: (email, Browser)
VOIP, Video conferencing

MAC → Medium access control protocol.

Cizen's Cipher
letter = letter + 3
A → D
B → E

NETWORK SOFTWARE:

1. ISO. OSI Reference model → Open system Interconnection
TCP/IP → protocol | ISO → International Standardisation Organisation.

- ISO. OSI Model:

Application	interface bet" user & network protocols
Presentation	convert format
Session	
Transport	process to process communication
Network	
Data link	flow control, error "
Physical	→ handle arr of bit

physical
data link
Network

Application

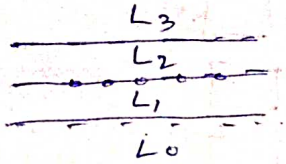
Big-endian
Little-endian

00001111 ← last bit

00001111 → BE
11110000 → LE

- ISO-OSI Reference Model → 7 layers
- TCP-IP Referance Model → 4,5 layers

- Layeredd Design:-



Send - frame to N/w layer

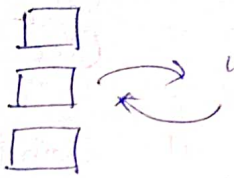
Send - packet to DLL (Packet)

Receive - packet from Net. Layer

- Why Layered Approach:-

⇒ It becomes modular

↳ Modular design → non-monolithic layer



we can take out
we can replace it with another one.

↳ It reduces complexity

↳ It will help in speed of development and correction.

⇒ In monlithic it become really tiredsome.

↳ Standardization becomes easier.

AL → interface between user
↑ ↓ → Interfaces

PL → compression/encryption → presenting data

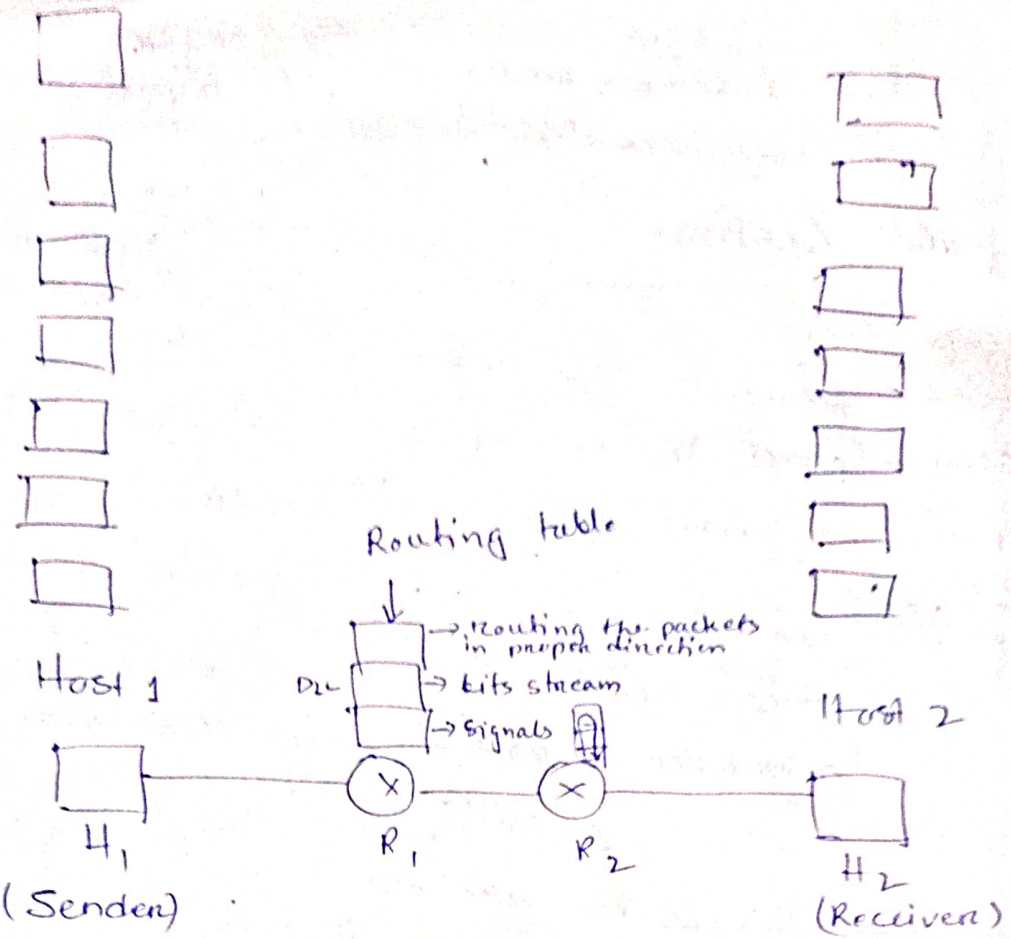
SL → session management

TL → process to processes communication.

NL → moving packets from source to destination

DLL → hop to hop communication

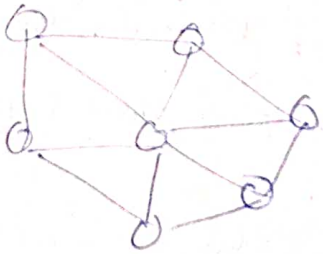
PhyL



- DLL can send bits stream but it cannot specify whom to send. → thus network layer helps.

- If router sends to all → flooding → duplicacy, havoc

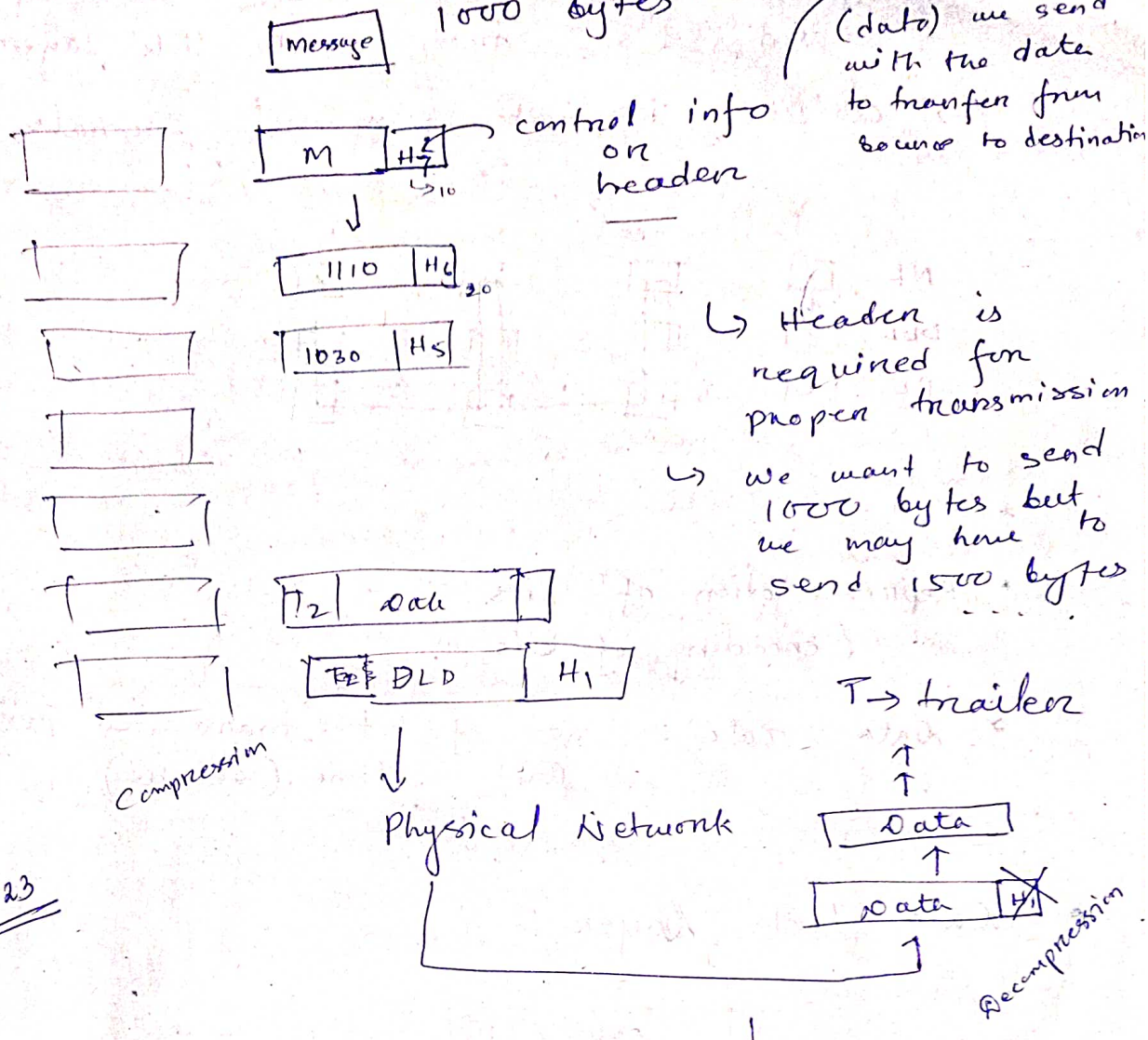
Topology → interconnection of diff. routers.



→ Routing table keeps on changing.

↳ Routing table should be always up-to-date.

↳ 6 DLL implementation is required (for 2 Routers) from sending from Sender ~~Host~~ to receiver



↳ Header is required for proper transmission

↳ We want to send 1000 bytes but we may have to send 1500 bytes ...

9/10/23

M
H7
AL
M_p → message processed
PL
M_p
H6
SL
Data
H5
SIP
TL
T. Data
H4
→ Segmentation → based on the size of packet it got.
Logical shielding
NL
N. Data
H3
IP. address
DLL
D. Data
H2
Data of Phy layer
H1

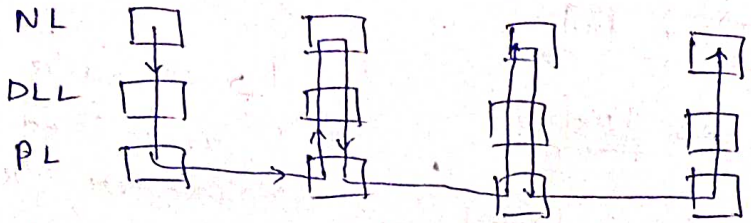
whether the data size is manageable by the network or not

↳ Reassembly ↓ in receiver or intermediate node.

MTU (Max. transport unit)

= Min (MTU_1, MTU_2, MTU_3 .. etc_3)

- Phy. layer: receivers bits in terms of 0's and 1's. from data link layer.
  ↳ convert it into physical signals

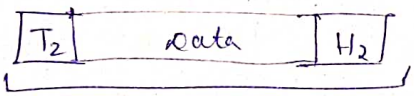


*

* Representation of bits
  (encoding of bits

* Data rate : no. of bits transmitted per unit time. (seconds).

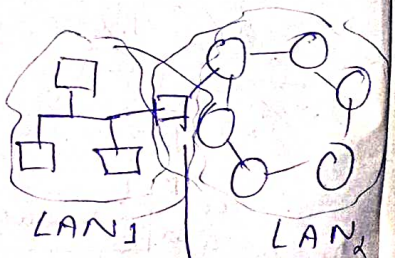
- Data Link Layer :-



Frame → collection of bits

* Framing :- Frame delimitan
  ↳ start and end delimeter is use
  ⇒ to determine the start & end of a frame.

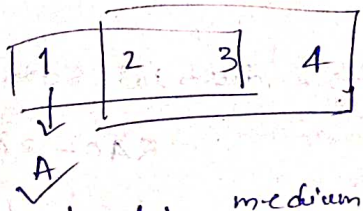
- Addressing :
  MAC Address
  ↳ for communication within a network.
  ?? How ^ internetwork communication is done?



Multi protocol devices
↓
it is a part of two LAN

LLC - Logical link control
↳ Framing
↳ Flow control
↳ Error control

* Flow control :-
SWP (sliding window Protocol)
↳ without acknowledgement, only minimum specified no. of frames can be send and not more than that.



* Error control :- medium is not perfect

Note:
(i) NL → helps to decide which outgoing line should be choosed.
(ii) DLL → finds the MAC address of the next node.
(iii) PL → transmits the data.

segment/datagram → packet
PL DLL

⇒ Hop to hop communication → DLL to DLL
⇒ Source to destination → NL to NL

• Network Layer :-
* Directing of packets from S to D.
* Logical addressing / IP addressing
* Routing of packets

(ARP → Address resolution protocal)

DLL:

- MAC (Medium access control)

Frequency Range:

800 - 950 MHz
1800 - 1900 MHz

ISM → Industrial, scientific & Medical band
80 MHz
240 - 248 GHz
IEEE 802.11 b/g/n

↓

Freq. left free by govt.

↳ At a time only one channel can be used..

If there are multiple devices, trying to access a shared resource, how we can do that without conflict.

⇒ Switch → it broadcasts → it doesn't need MAC address

- Network Layer:

source to destination
end to end connection → not a physical connection

- Transport Layer :->

Service point addressing or process address
↳ process to process communication.

↳ Segmentation and Reassembling
↓ in sender side ↓ in receiver side

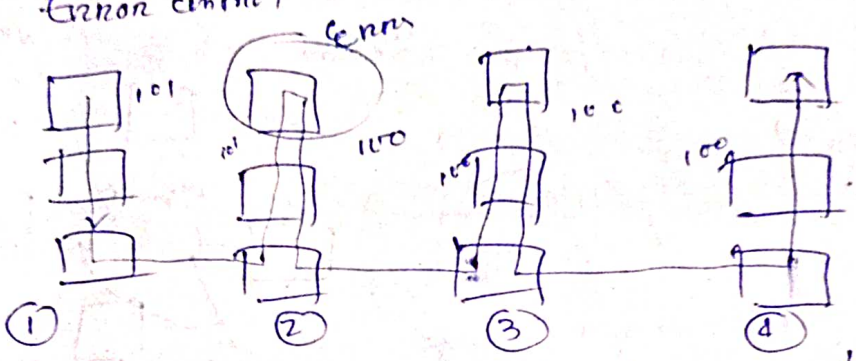
* If application layer sends a very large data, small networks cannot ~~se~~ send it.

↳ So, if is broken and send and corresponding reassembling is done in receiver's side.

• ↳ Why we need error control in transport layer ?

↳ There might be malfunction of data in bet^n.

Error control



↳ seems correct, because the error in network layer won't be caught.

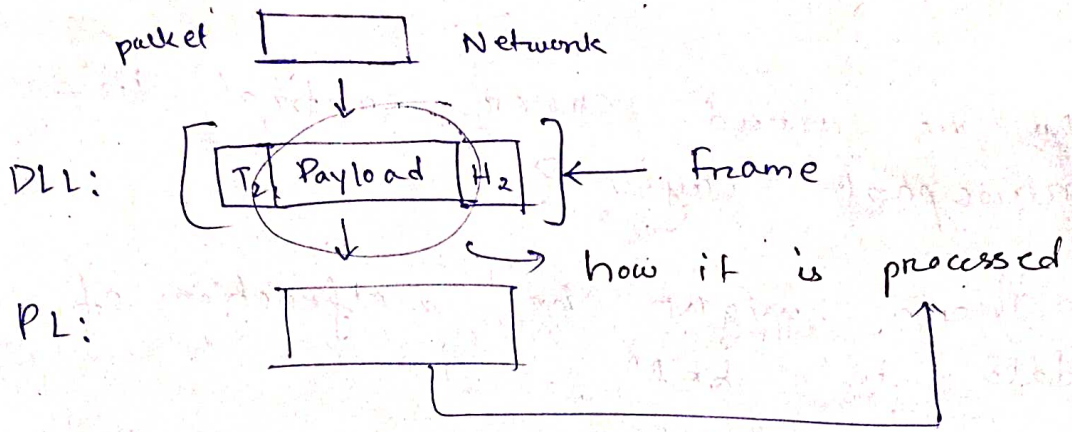
→ ① It sends 101 and ② Receives 101. ✓

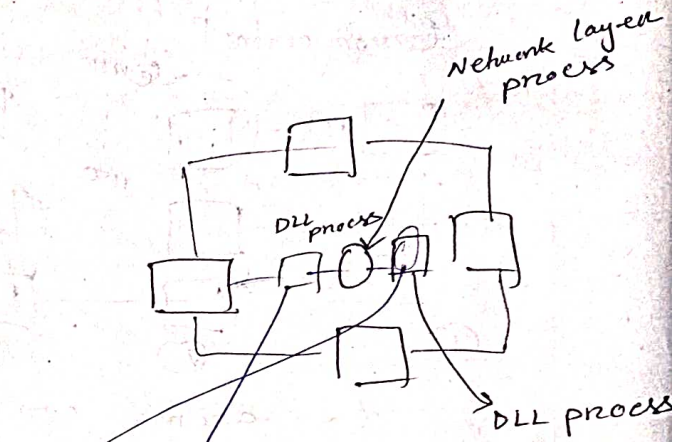
→ Network layer error in ② → 100
↳ converted.

→ ② sends 100, ③ receives 100.
↓
③ it seems fine but it is wrong.

13/10/23

Operating System.





DLL process receives frame and converts it into packet.

→ After Network layer processing, again DLL process is done to convert into frame

↳ frame delimeters → start & end delimeters of a frame.

⇒ While transmitting it will have packet from network layer + header + trailers and start & end delimeters.

- Header: ↳ Addresses → MAC address
  ↳ unique in all type of devices.

* Type of frame :- → Control Frame
→ Data Frame
→ Management Frame

↳ sequence numbering

- Trailers:-

Error correction & error detection.

↳ Detection & retransmission.

↳ Forward error correction.

* Byte Oriented Protocols

01101010

byte oriented → 8 bit

* Bit Oriented Protocols

0 11 0

↳ bit by bit it will transmit

- Delimeters:-

① Character Count Method:

↓
character means byte

eg: 1 2 3 4 5 7 8 9 3 0 8

| 5 | 1 | 2 | 3 | 4 |

| 4 | 5 | 7 | 8 |

| 5 | 9 | 3 | 0 | 8 |

↳ no. of character to be send including itself.

5 = 101
↓
100

↳ only a change of one bit and whole transmission will be affected.

| 4 | 1 | 2 | 3 | 4 |

| 4 | 5 | 7 | 8 |

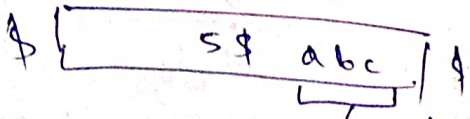
| 5 | 9 | 3 | 0 | 8 |

↳ 8 character is not present

* Using special characters:

→ Using $ at start and in the end.

→ But if $ is used in the body, then it will discard everything after that.



⇓

we use another dollar to solve this.

↳ Let's say we use ∝ before $, which will say that the $ following ∝ is not an start and end delimeter but part of it.

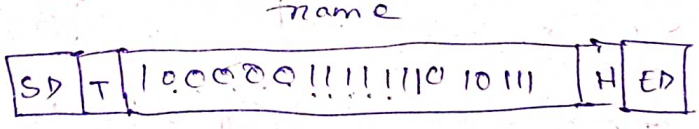
* what if ∝ is also a part?

Let's say ∝$ is part of it

we use ∝ ∝ $

↳ it is a part

③ Bit Stuffing Technique :-



0 1 1 1 1 1 1 0

↓

used as start and end delimeter (0 - Six 1's - 0)

↓

If there are more than six ones are there, after five ones we will add one 0, as extra bit. (append a zero).

1 0 0 0 0 0 1 1 1 1 1 0 1 1 0 1 0 1 1 1

X 0 1 2 3 4 X 0 1 X 0 X 0 1 2 3

Destuffing in the Receiver's side:

↳ after five 1's, the 0 will be deleted.

* Worst case:

↳ for all ones, after every five ones, one zero will be added.

20% extra bit will be added.

- Error Control:-

↳ Error Detection

↳ Error Correction (FEC)

↳ Forward error correction

* Transmission → receiver

↓

error detection

↳ retransmission mechanism

* If two devices are sending message for one another, and they are going apart, connection is lost, so how long the sender will wait?

↳ there will be timeout.

* Extra bits are send, called redundant bits.

$$\left[\frac{0111101}{m}\ \frac{101}{R}\right] \rightarrow \text{codeword.}$$

$n = m + r$

n → codeword, m → message

R → redundant bits.

valid code word / invalid code word

Hamming distance: → if distance is more → coding is good.
(d)

```
 10010
 01010
-------
```

d = 2

↳ XOR on both the codeword

```
 10010
 01010
-------
 11000
```

↳ Then count no. of 1's.

∴ d = 2.

[7 4] hamming code
↓ ↓
n m

$r = 7 - 4 = 3$

codeword $[x_1 x_2 x_3 x_4 x_5 x_6 x_7]$

↳ anything power of 2 is taken by r bits.

$2^0 = 1$
$2^1 = 2$
$2^2 = 4$
$2^3 = 8$

∴ Position 1, 2 and 4 will be taken by r-bits and
position 3, 5, 6, 7 are taken by message

$m = x_3 x_5 x_6 x_7$
$r = x_1 x_2 x_4$

$X_4 = X_5 + X_6 + X_7$

$X_2 = X_5 + X_6 + X_7$

$X_1 = X_3 + X_5 + X_7$

[Any no. can be expressed as sum of power of 2]

$7 = 4 + 2 + 1$

$6 = 4 + 2$

$5 = 4 + 1$

$3 = 2 + 1$

Eg:

1010

1	2	3	4	5	6	7
1	0	1	1	0	1	0

parity bit checking

↳ even parity → total no. of one including parity even should be even.

↳ it can only detect bit change of one bit only.

$a = X_4 + X_5 + X_6 + X_7$

$b = X_2 + X_3 + X_6 + X_7$

$c = X_1 + X_3 + X_5 + X_7$

→ a, b, c must be zero → no error

→ if a, b, c is not zero, there is error.

1	2	3	4	5	6	7
~~1~~	~~0~~	1	~~1~~	0	1	~~0~~
0	1	1	0	0	1	1

if 7th bit is garble (error)

it effect the 4th, 2nd and 1st n bits.

~~0 11 0~~

eg 2: 0 1 1 0

1	2	3	4	5	6	7
1	1	~~0~~ 1	0	1	1	0

a b c

0 1 1 = 3 rd position is error.

Cyclic Redundancy Check

x^4 x^3 x^2 x^1 $\rightarrow x^0$

1 0 0 1 0 0 1 1 1 0 0 1

Additional bits

⇒ Multiply term with 0 or 1 acc. to the input bit.

1 0 1 1 1 0 1 $\rightarrow x^0$

$1 \times x^6 + 0 \times x^5 + 1 \times x^4 + 1 \times x^3 + 1 \times x^2 + 0 \times x^1 + 1 \times x^0$

$x^6 + x^4 + x^3 + x^2 + x^0$

$M(x)$ =

$x^n P(x)$.

→ [P(x)] [n no. of zeros]

→ append with n no. of zeros.

① Let n be the degree of generator polynomial $g(x)$. Then append n zero bit to the lower order of the frame (right hand side)

$T(x)$ = $M(x)$ + n bits

② Divide the bits stream corresponding to $x^n M(x)$ by bit stream corresponding $g(x)$ using Mod 2 operation.

$$\begin{array}{r} 0 \\ 0 \overline{)} \\ \hline 0 \end{array} \qquad \begin{array}{r} 1 \\ 0 \overline{)} \\ \hline 1 \end{array} \qquad \begin{array}{r} 0 \\ 1 \overline{)} \\ \hline 1 \end{array} \qquad \begin{array}{r} 1 \\ 1 \overline{)} \\ \hline 0 \end{array}$$

Subtract the remainder from the bit stream corresponding to $x^n M(x)$ using Mod 2 subtraction. The result is the checksumed frame to be transmitted on if is known as T(x) (Transmitted frame).

$M(x) = 11010110110$ $\quad G = 10011$

$x^4 M(x) = 11010110110000$ $\quad$ degree of $G(x) = 4$
$\therefore n = 4$

↳ same message appended by 4 zeroes.

```
          11000101
      ┌──────────────────
10011 │ 11010110110000
      │ 10011
      │  10011
      │  10011
      │   10110
      │   10011
      │    10100
      │    10011
      │     1110
```

Transmitted message: 11010110111110

Message: 11001001 is to be transmitted using CRC polynomial $x^3 + 1$, to protect it from errors. What should be the transmitted message.

M(x) = 11001001

• How receiver will detect the errors?

(a) ↳ Divide receive bits stream by G(x)
↳ If it is divisible → no error.

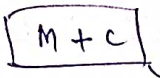
(b) Divide and compare the remainder.

problem (strength)

↳ If the message is divisible by g(x) and after adding extra bits still it is divisible. After adding the remainder, if the T(x) becomes not divisible then receiver can detect it easily.

But if due to transmission error, the message becomes divisible by g(x), so it won't detect the error.

M = n-bit message
C = r-bit checksum.



→ Transmitted bit stream.

Assume error E(x)

Receiver receiver: T(x) + E(x)

Receiver $T(x) + E(x) / G(x) = 0$

↳ correct frame

If it is not 0 → error

30/10/23

Receiver acknowledges/informs about lost or duplicate frame / anonomys frame.

↳ Retransmission is required (by senden)

⇓

ARQ → Automatic Repeat request

- Flow control: set of procedure to inform a fast senden about the capabilities of a receiver.

Protocols:

- Noiseless channel
  - Simplest Protocol
  - Stop and Wait Protocol
- Noisy channel
  - Stop and wait ARQ
  - Go back-N ARQ
  - Selective Repeat

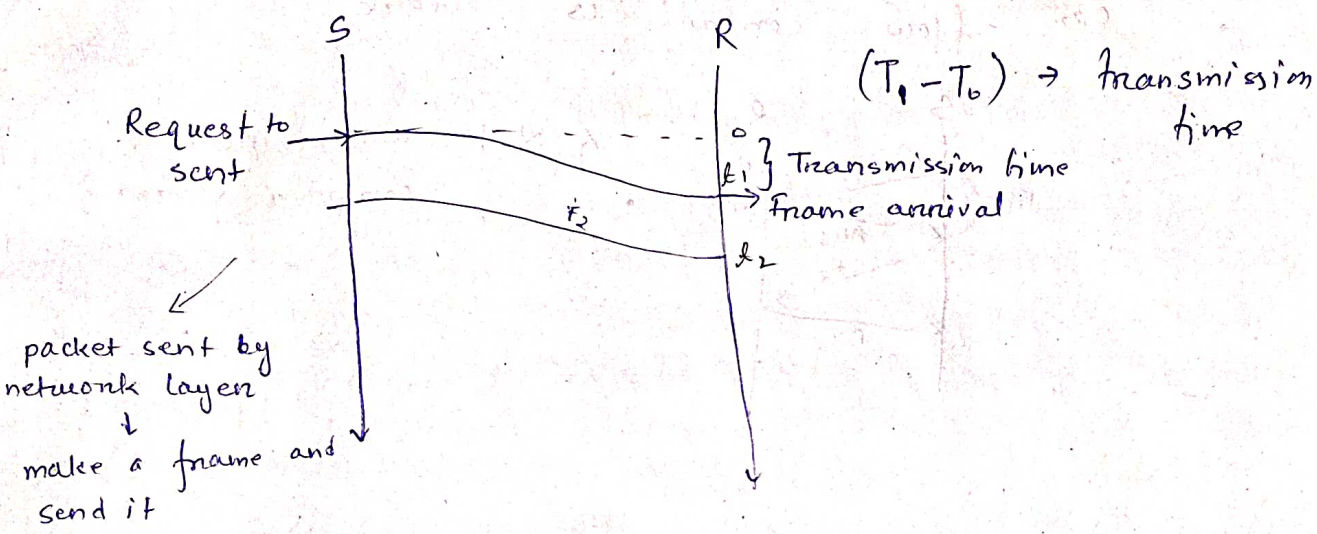
- Simplest Protocol :-

↳ No restriction
↳ channel is perfect
↳ no noise ⇒ no error assumption.

↳ it is not a realistic

↳ sender can send anytime (no restriction), it means receiver has unlimited buffer, (speed of receiver is unlimited)

↳ no flow control, no error control required.



$(T_1 - T_0)$ → transmission time

Sender side

while (1) / while (true)

NS2 / NS3 code

```
{ WaitforEvent();
  if Event(RequesttoSend) //→ packet coming from network layer
  {
      Getdata();
      Makeframe();
      Sendframe(); // send the frame to the receiver site.
  }
```

↳ infinite loop

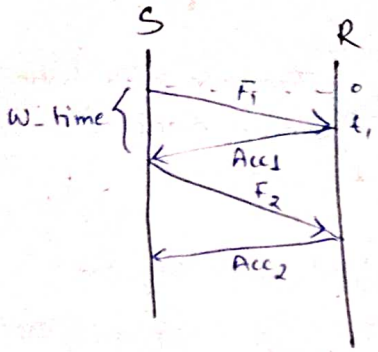
Receiver's side

```
while(1) { Waitfor Event();
           if Event(Arrivalnotification)
           { ReceiverFrame();
             Extract Frame();
             Deliverdata(); // delivering packet to N.L
           }
}
```

- Stop and Wait Protocol

  sender send 1 frame and wait for acknowledge to come. Then only it will send next frame.

  ↳ Receiver has limited buffer.
  ↳ flow control is there.



{ Note: we do not consider error }

```
Cansend = True
{
while (1)
Cansend = True;
{
  wait_forEvent();
  if Event( Requesttosent && Cansend == True)
  {
      Getdata();
      Makeframe();
      Senddata();
      Cansend = False;
  }
}
wait for Event
```

```
WaitforEvent();
if Event( Arrival_notification)
{
    ReceiveFrame();
    Cansend = True;
}
```

- ARQ
  - ↳ for noisy channel

(1) Stop and wait ARQ
(2) Go-back N ARQ
(3) Selective Repeat

Gaurav Sir

1/11/

Components of a Network:

1. NIC (Network Interface Card)
2. Servers
3. Workstation
4. Cabling System
5. Shared resources & Peripherals.

- Simplex, half-duplex, full-duplex
- ISO - OSI Layer

International Standardisation Organisation
Open System Interconnection

7 - layers

1. Application Layer
2. Presentation layer
3. Session Layer
4. Transport Layer
5. Network "
6. Data-link layer
7. Physical Layer

[Aaj Phir Se Test Nhi Dena Pareega].

TCP/IP

HTTP; SNMP, SMTP, FTP, TELNET

UDP: ICMP (Internet Control Message Protocol)
user data gram protocol.

• IP Layer:

TCP/IP

↳ Dev. by ARPANET (adv. Reseach Project Ajency Network)
↳ It consist 4 layer
↳ It is a horizontal approach
↳

OSI

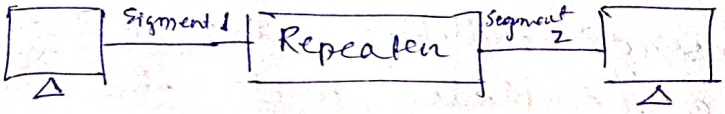
↳ It is dev. by ISO (Int. Standardisation Organisation)
↳ It consist 7 layer
↳ It is vertical approach.

9/11

- Hub :-

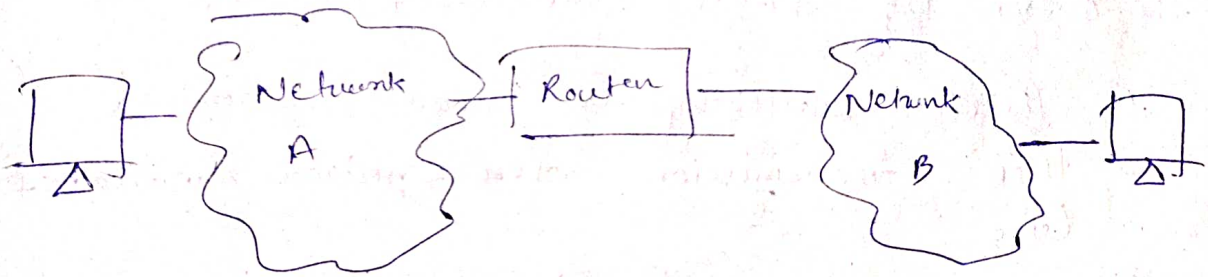
  The general meaning of the word hub is any connecting device.

- Repeater: Repeater is a connecting device which can operate in data link layer.



The major diff. bet^n bridge and repeater is that bridge has a filtering capability

- Router : It is a connecting device that can connect two or more network



- Diff bet^n hub, switch, router

HUB	SWITCH	ROUTER
→ Simple networking device works under Physical layer of OSI Model that connects bunch of computers. → Lots of wastage of bandwidth.	→ Switch is a multicast networking device under datalink layer that connects computers.	→ It is also a device under network layer that connects two or more different networks.

* Hub and Switch connects PC on the same network but router connects in different network.

Switching Techniques :-

- Service Switching: It establish a dedicated path bet^{n} sender and receiver.
  Eg: It is used in public telephone network also in voice transmission.

- Message switching: A message is transferred as a complete unit and routed through intermediate nodes at which it is stored and forwarded.

- Packet Switching: When we send a message, then the whole message is divided into smaller pieces of packets.

Adv. of Packet Switching in Network :-

(1) Packet switching is more economical then transmission over private/dedicated lines.

(2) Diff packets can travel along diff routes till they reach their desdination. Thus, a failure in one switch or a particular link does not effect.

(3) Packets are accepted even when network is busy.

Network Layers Protocol

① Address Resolution Protocol

↳ It is used to convert a network address eg (IPv4) to a physical address ~~eg (IP, 4)~~ such as an ethernet address also known as a MAC address.

② DHCP (Dynamic host configuration Protocol)

↳ The DHCP has been ~~derived~~ deviced to provide static and dynamic address.

③ ICMP: (Internet Control Message Protocol)

↳ The ICMP supports the unreliable and connectionless internet protocol.
There are two categories of ICMP messages:

(i) ~~Error~~ Reporting
(ii) Query messages

④ IGMP (Internet Group Message Protocol)

↳ It can be used for one to many networking application such as online streaming, video and gaming.
Also it is used in (IPv4) networks

- IPV4

Class-full Address:

A → 0-127 IP addresses
B → 128-191 "
C → 192-223 "
D → 224-239 "
E → 240-255

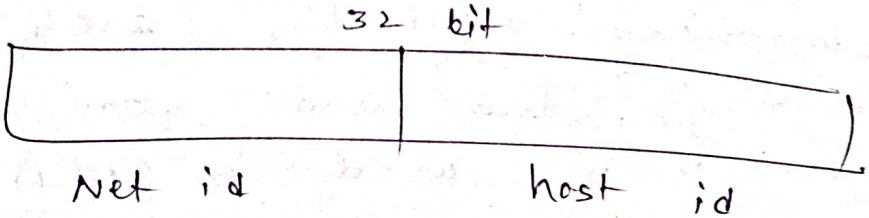
An IPV4

- IPV4 address are unique. They are unique in the sense that each address defines one, and one connection to the internet
- IPV4 addresses are universal in the sense that the addressing system must be accepted by any host that ~~connects~~ wants to be connected to the internet.

Classfull Addressing

IPV4 addressing, at its inception, use the concept of classes. This architecture is called classfull addressing.

In classful addressing, the address phase is divided into 5 classes.



CIDR (Classless Inter Domain Routing)

The notation slash n is also called CIDR notation where slash n can be 8, 16 or 24 in classful addressing.

Classless Addressing:

To overcome address depletion, and give more organization access to the internet, classless organization was designed

During the era of classful addressing, subnetting was introduced. A large network to divide the address into several groups. And each group is assigned to smaller networks, all subnets.

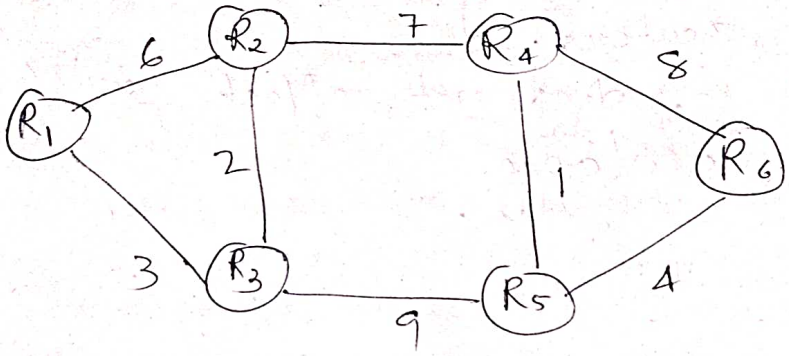
- **Supernetting:**

  In supernetting several networks are combined to create a supernetwork or supernet.

  A must is a 32-bit no. where n leftmost bits 1's and the 32-bit

- Linked-state Routing

↳ In Linked-state routing, if each node in the domain has the entire topology of the domain, the list of nodes and links, we use Dijkstra's algorithm to build a routing table.



R_1	R_2	R_3	R_4	R_5	R_6
	6	3	∞	∞	∞
R_1R_3	5	3	∞	12	∞
$R_1R_3R_2$	5	3	12	12	∞
$R_1R_3R_2R_4$	5	3	12	12	21
$R_1R_3R_2R_4R_5$	5	3	12	12	16
$R_1R_3R_2R_4R_5R_6$					

	V
R_1 0	R_1
R_2 5	R_1
R_3 3	R_1
R_4 12	R_3R_2
R_5 12	R_3
R_6 16	R_3R_5

The Dijkstra's algorithm creates a shortest path tree from a graph. The algo. divides the nodes into two sets : tentative & permanent.

- Distance Vector Routing :

In distance vector routing, the least cost route betⁿ any two nodes is the route with min distance. In this protocol, as the name implies, each node maintains a vector of min. distance to every node.

NAT (Network Address Translation)

NAT is a router like function where IP address of IP datagrams are replaced at the boundry of a private network.

NAT is a method that enables host on private network to communicate with host on the internet

- Basic Operation of NAT

① NAT device has address translation table.
② One - to - one address translation.
③ Mapping betⁿ private IP to public IP

10.0.0.0 → eg of private IP.

STOP and WAIT Algo for senders side and receivers side

1. Start the program.
2. Generate a random no. that gives the total no. of frames to be transmitted.
3. Transmit the 1st frame.
4. Receive the ack. for the 1st frame.
5. Transmit the next frame.
6. Find the remaining frames to be sent.
7. If an ack is not received, for a particular frame, retransmit that frame alone again.
8. Repeat the step 5 to 7 till the no. of remaining frames to be sent becomes 0.
9. END