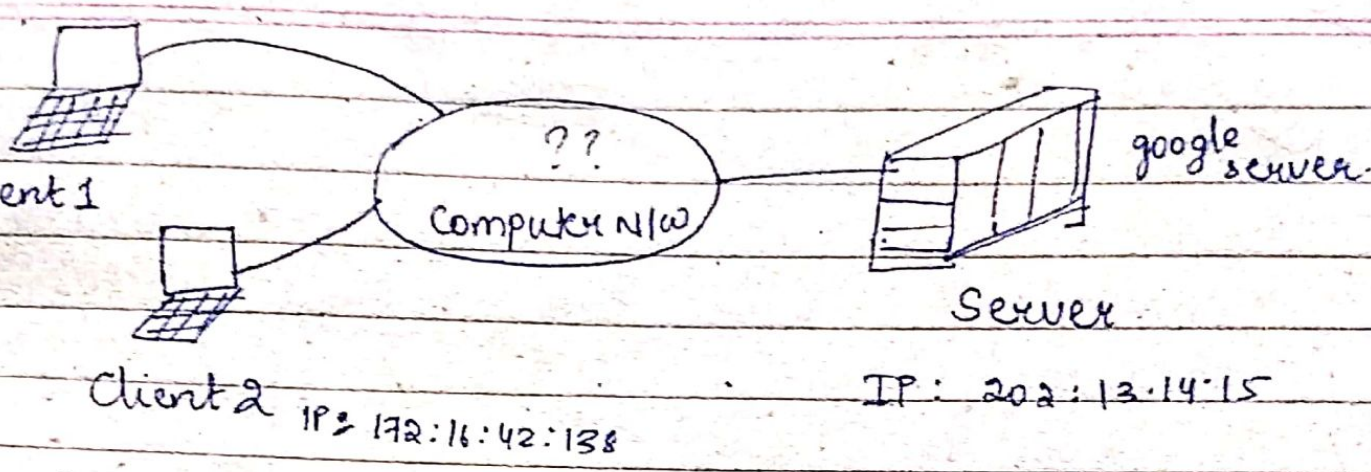
10/01/19.



Intranet → Client and server are in the same network
→ Small network.

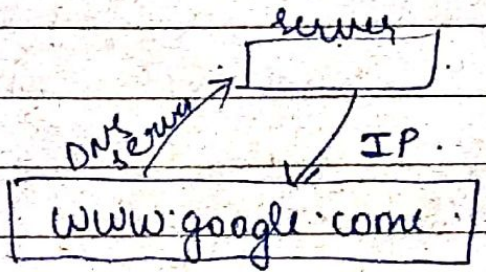
Internet → Big network
→ Client and server are not in the same geographical network.

- TCP/IP (TCP → Transmission control protocol).
Networks / model

→ We will cover this mostly in our course.

→ DNS → Domain Name Server.
Every client in a network has a unique address (IP address) which will allow it to identify in the network.

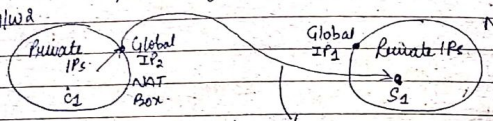
Suppose we have a google server, if a client type www.google.come in its browser then DNS convert/translate the url to its IP address, (here it is 202:13:14:15)



HTTP:- Hyper text transfer protocol.

IPV4 address is of 32 bits.

To overcome the problem of IPV4 exhaustion, we have NAT.



If we want to want to communicate in the same network (N/w 1) then we can use private IPs.

If a client C1 wants to communicate with server S1 in another network N/w 1 then, we cant use private IPs so the NAT Box gives the client of N/w 2 a global IP address, which is then received by the server in another network.

Range of private IPs.
- 10.1.2.x×2
- 172.
- 168.
- 202.

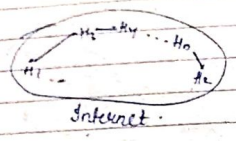
→ We can also have a NAT Box at the host side (server side). similarly if host wants to make request to client on another network, then NAT boxes will be used.

Why learn Internet?

① Job prospects
② Curosity
③ Impact on World (Political, Economical, Social/Societal)
- Political parties uses internet (Arab news)
- Google Adv. Sponsored search
- Online Search - Amazon
- Online Market place - ebay, olx.
- Encyclopedia → Wikipedia/Google Search
- Electronic commerce (PayPal)
- Discussion on sponsorship (TORR

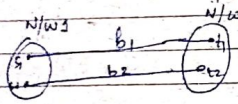
Key Problems	Example Solution
① Reliability despite failures	(i) Codes for error detection/correction (ii) Routing around failures.
② Network growth and evolution.	(i) Addressing and Naming (Subnetting schemes). (ii) Protocol layering
③ Allocation of resources like bandwidth.	(i) Multiple Access protocols (ii) Congestion control.
④ Security against threats.	(i) Confidentiality of Message (ii) Authentication b/w communication parties

Since, internet is very vast there can be failures.



Host H_1 wants to communicate with H_2
There are many intermediate nodes

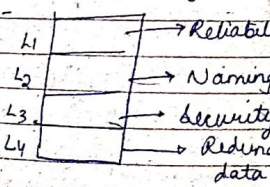
If H_4 goes down, there should be no failure in communication b/w H_1 and H_2 (that is network should be designed in a way that is reliable).

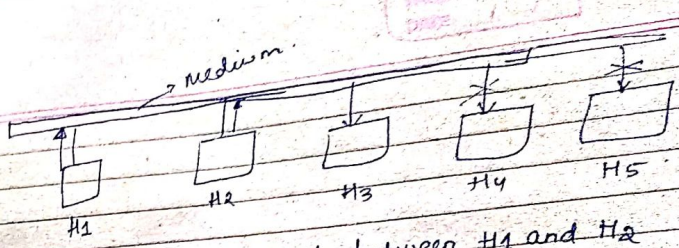


They are assigned a particular bandwidth so how to allocate this B/W is one of the key problem.

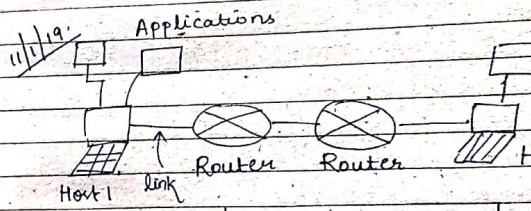
Instead of having one layer, we can divide our ~~protocols~~ network protocol into multiple layer.

for eg:-





There is a collision of packets between H_1 and H_2 and the packets will not reach H_4 and H_5 (as there is a collision and data may be lost).



Component	Function	Example
① Applications/Apps	Use the N/W	Skype, itunes
② Host/Note/Source/Destination	Support- User App	laptops, Mobile, Desktop.
③ Routers/Switches/Hub	Relays Msgs b/w links	Access point, cable/DSL, Modem.
④ link or channel	Connect Nodes	Wires, Wireless.

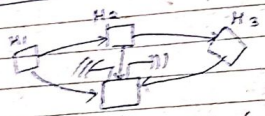
Types of links:

① Full Duplex — Bidirectional ⟷ (Simultaneously both host can talk to each other)

② Half- Duplex — Bidirectional ⇄ (At a single time, only one can communicate, but link is bid

③ Simplex
- Unidirectional ⟶ (wireless)
(The link is unidirectional, only one host will send msg and other will receive at any time)

④ Broadcast



Access point (wifi)

If any host or access point is using a wireless access link, then all the components that in the range of the wireless access link can communicate or will be connected.

Types of Network:

① Wired N/w
② Wireless (WiFi, 802.11)
③ ISP (have their own routing methods/mechanisms)
④ Cellular N/w (3G, 4G, 5G)
⑤ Satellite
⑥ Bluetooth
⑦ Telephone N/ws.

Only 1-4 we will be studying in this course.

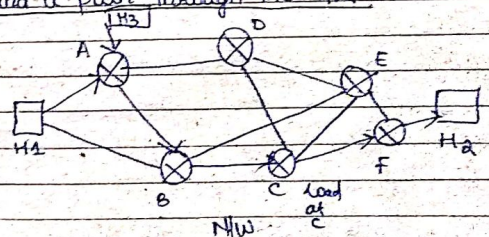
Network by Scale:

Scale	Type	Example
① Vicinity	PAN (Personal Area N/w)	Bluetooth
② Building	LAN (Local Area N/w)	Wifi, Ethernet
③ City	MAN (Metropolitan " ")	cable, DSL
④ Country	WAN (Wide area N/w)	Large ISP
⑤ Planet	Internet (N/w of N/ws)	Internet

(we will later cover all this in detail in slides)

Modular Networks:
→ the network being divided into multiple layer.

N/w does following operations of App:
① Make or break a connection (To start the communication, we have to establish a connection which is done by the N/w)
② Find a path through the N/w.



N/w

Initial connection b/w H1 and H2 is through ABCF.

If link BC is broken, then the N/w has to determine or make a decision for the alternative route. It establish a new route. ABEF for communication b/w H1 and H2.

③ Transfer Information Reliably

01110 → 11010
Transmitting (Actual m/sg) → receiving (m/sg received)

There should be a mechanism that N/w should ensure that the received m/sg is incorrect and send back the m/sg to sender for correction.

④ Send as fast as the N/w allows.

If node C or router is getting data from B and D also. So the router becomes overloaded and it become slow.
So the N/w should do something such that D sends the data to E instead of C for load balancing.

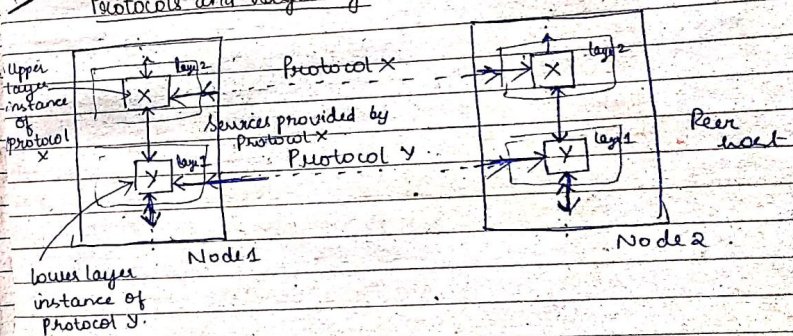
⑥ Secure info in transmitting.

⑦ Let many new hosts to be added to the N/w.

There are lot of functionalities, if we put this one layer, it becomes complicated. And if one functionality fails then the whole N/w fails.
If we have multiple layers, then we divide the functionality to different layers, even if one layer fails then the functionality on the other layers won't be affected. Thus this is the advantage of the modular network.

11/2/19

Protocols and Layering

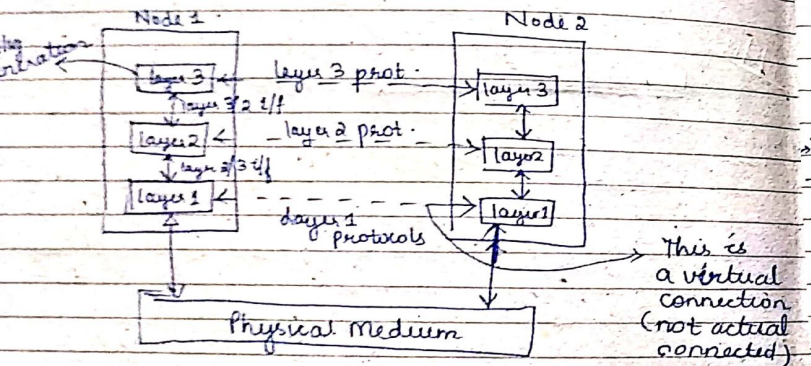


→ Instance of protocol X operating in Node 1 at layer 2.
→ Instance of protocol Y running at layer 1 in node 1.
→ Communication will take place between the respective layers in the node 1 and node 2.
i.e) layer 1 and layer 2.

→ Here L2 is running a protocol X.
It will use the instance of protocol X for communication.

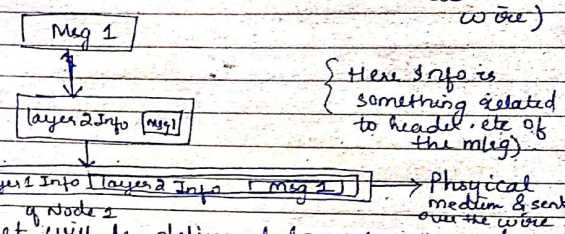
→ Services provided at the lower level will be used by the protocols at the upper layer.
i.e, lower layer protocols provides support to the upper layer protocols.

→ Each layer in a node will always communicate to the respective node in the other layer.



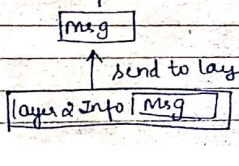
→ If node 1 wants to send msg node 2, it is generated by layer 3.

→ Message generated by layer is known as packet here.
→ layer 3 will send the M/sg packet to layer 2.
It will create a new packet, it will encapsulate the m/sg received by layer 3 and will include some information.
Then again, layer 2 will send the new modified packet to layer 1. It will do same as layer 2. Then the message is put to some physical medium. (is the actual wire)



→ The layer 1 packet will be delivered layer 1 of the node 2.
→ Your m/sg will be interacted to the topmost layer. as most of the apps are running at the topmost layer.

→ The packet is then with layer 1 of node 2, it will extract the layer 2 Info and remove layer 1 info.
→ The m/sg is then delivered to layer 2 in node 2.
It will remove layer 2 Info and extract the m/sg and will send it to the layer 3 of node 2.



Protocols

① HTTP → Application layer netwo protocol.
② FTP → "
③ TCP/UDP → Network layer protocol.
↳ layer oriented service.

If we are using a TCP protocol for transmitting msg and if somewhere packet is lost eg it ensures or tells the sender host that your packet is lost and please send again.

But in UDP if your packet is lost, you will not be able to know about the loss of packet.

④ SMTP → Application layer protocol.
③ Internet protocol → N/w protocol.
⑥ 802.11; Ethernet (802.3) (link layer protocol)
Layers
wireless wired

Application layer → (Interacts with the user)
Transport layer
Network layer
Link layer
Physical Medium

1st Fig

Browser

L4	HTTP
L3	TCP
L2	IP
L1	802.11

App.
Transport
N/w
Link

→ data

www.google.com → http://ww....

2nd Fig

Browser

HTTP

Tcp info | HTTP

IP info | Tcp info | HTTP

802.11 info | IP info | Tcp info | HTTP

↓ sent over the wireless medium

→ HTTP is one of the protocol that works at the application layer.

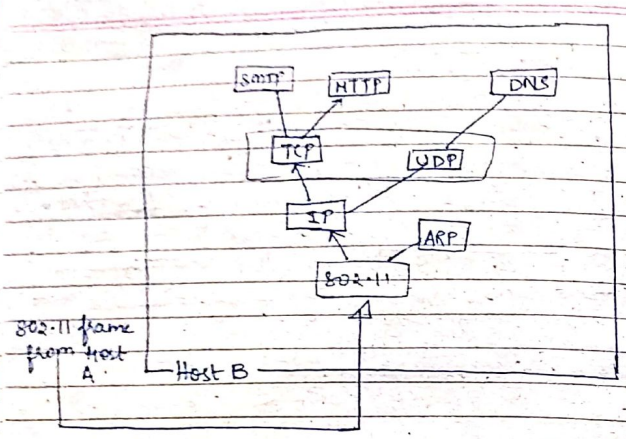
→ When this info is received by host 2, then it will do the reverse and will finally extract the browser information.

— x —

14/11/19

Representing as 2nd fig. is difficult so we represent it as:-

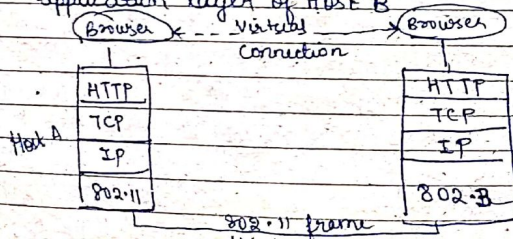
802.11	IP	TCP	HTTP



→ ARP → address resolution protocol.

→ How will the link layer decide to which protocol in the N/W layer it has to send the message? With the help of the info in the packet.

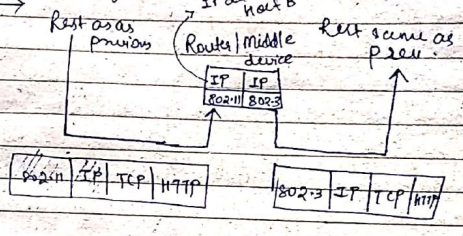
→ Then finally like this our packet reaches the application layer of Host B



→ Sender is in a wireless N/W and receiver is in a wired N/W.

→ Somehow 802.11 frame have to be converted to 802.3 frame.

→



→ Using the IP of first, it will create a new IP or copy the same IP and create a new packet.

→ Router will remove 802.11 link layer frame & replace it with 802.3 link layer frame and will create a new ethernet packet.

This is how we can make communication between two heterogeneous N/W.

Advantages of layering:

① Scalability.
② Reliability & Robustness.
③ Security.
④ Error control.
⑤ Congestion control.

Disadvantages of layering

① Additional Overhead
→ Add. overhead which increase the time of communication.
→ First we send the package down the layers of Host 1 and the all the way up to the application layer of Host 2.

② Information Hiding
→ The application will not be knowing whether it is operating over a wired or wireless N/w.
→ There are set of contraints in wireless N/w, constraint of bandwidth & resources.
If it knows whether it is operating in wireless N/w it knows resources are constraint, so it will allocate resources efficiently.
But if operating on wired N/w → it can allocate resources without much thinking as there are not much constraint.

OSI Reference Model

→ One of the earliest layer architecture model
→ Not used in current scenario

Application
Presentation
Session
Transport
Network
Data Link
Physical

Functionalities we need in every layered model are :—
① Routing
② Encryption
③ Retransmission
④ Congestion control

Physical Layer :-
① Transmit Raw bits of Data
② Determines the type of communication (1 or 2 way)
③ Deals with mechanical, electrical & physical aspects of data transmission.

Data link layer
① Ensures error free delivery of data
② Traffic Regulation Mechanism
③ Control Access to shared medium

→ We have very fast sender which sends data over a very high speed & receiver is not that fast the data link layer have mechanism to tell sender that you are sending data at a very high speed

which sender is unable to receiver & your data may get lost.

→ If there are more no. of host in one shared medium, then there may be collision of packets of different hosts.
So it is the work of data link layer, which controls that only one sender should send the data at a time to prevent collision and prevent the packets of getting lost.

18/1/19

Network layer:

→ Determine how packets are routed from source to destination.
→ Routing decision can be static (using static table entry)

Router A

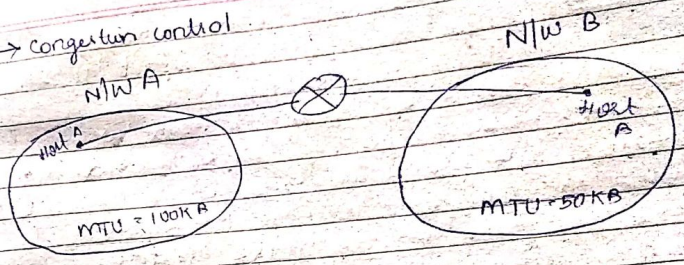
Host A (IP add)	Forward to Router/Node 1
Host B (IP add)	Forward to Router/Node 2

The entries in the table is fixed

Router table Entries

→ The table can be dynamic changing entries as per the traffic in the N/W.

→ congestion control



MTU :- Maximum Transmission Unit
(maxiim packet size it will support).

→ Router knows that if it will send it packet from Host A to Host B directly, then info will be lost as Host B can support only packet of 50KB.
→ so it will split the 100KB into 2 50KB packet and will then send to Host B.
This is one more functionality of N/W layer.

Transport layer:

→ When it receives msg from the session layer, it will breack msg into smaller units and then pass it to N/W layer.

→ TCP → reliable
UDP → unreliable
} Two major protocols in transport layer

→ We need router in between two hosts for 3 layers → Physical, Data link & Network because we get IP address at the Network layer so after that router knows the IP address and have to make decision upon this.
For the above layer, it need not need any routing device as it can directly communicate b/w Host A and Host B

Session layer:

↳ Establish session b/w 2 machines
→ Ensures recovery from crash

Presentation layer:

↳ concerned with the syntax & semantics of the information transmitted

→ It decides whether to send MSB first or LSB first.

Application layer:

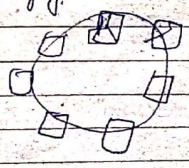
→ Widely used application protocol is HTTP
→ Widely used by users

LAN (Local Area Network)

→ Operating speed → 10Mbps to 10Gbps

→ One cable will be used to transmit the data → Bus Based LAN topology
→ At one instant only one machine is allowed to transmit the data

→ Ring based LAN topology
→ First need to acquire token
→ Comparitively slow wrt bus based topology



MAN
→ Eg - cable TV N/w.

WAN
→ Comparatively bigger than LAN & MAN
→ Subnet is operated by the Internet Service Provider.

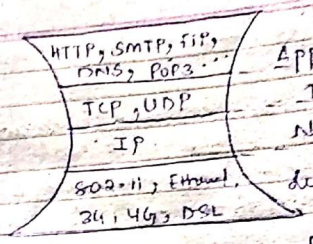
Wireless N/ws
→ Previous all n/w were wired N/ws
→ All the wireless N/w are eventually connected to wired N/w at some point. This is done at base station.
→ Eg - cellular N/w telephones.

TCP/IP Model.

7	Application
4	Transport
3	Network
2,1	Link

→ In this model we have combined the 1 and 2 layer OSI model to link layer.

→ The session & presentation layer of OSI model is combined in Transport & App. layer of this model.



Hour-glass model Protocols in TCP/IP model

Layer Based Names for data units.

Layer	Unit of Data
Application	Message
Transport	Segments
Network	Packets
Link	Frame
Physical	Bits/Bytes

Names of devices at different layer.

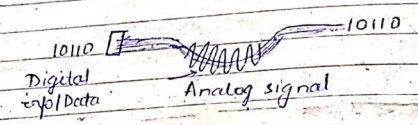
ⓐ Repeater/Hub. — Physical | Physical

ⓑ Switch (or bridge) → MAC address — Link | Link

ⓒ Router ⇒ IP address — N/w | N/w; link | link

Physical Layer

→ Concerns how signals are used to transfer message bits over a link
- wires carry analog signal
- we want to send digital bits.



Challenges we might face in converting digital to analog or analog to digital.

① Properties of wire
↳ wire, fibre, wireless

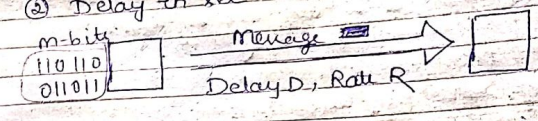
② Signal Propagation
↳ Bandwidth, Attenuations.

③ Modulation
↳ Representing bits.

④ Fundamental Limits (at what rate you can send your data across the link)
↳ Nyquist,
→ Shannon

Properties of Physical channel

① Rate (Bandwidth, capacity) in bits/sec
② Delay in sec



Rate → Unit of data a particular link will send at one time
D → How much time a link will take to send data from one end to another end.

→ Delay is a combination of 2 message latency factors:-
① Transmission Delay
② Propagation Delay → (time taken to transmit from one end to other)

How much time a particular host will take to put all the transmission bits / info in the link channel.

Transmission Delay = M/R

Propagation Delay = length of Phy wire (D) / C = speed of light

$\geq 2/3\,C \geq D$

Latency (L) : $\frac{M}{R} + D$

↓ Total delay.

M = m bits of data you want to transmit.

Eg Dialup with telephone modem

Prop. Delay = 5ms, Data Rate = 56Kbps.
Message size = 1250 bytes.
Find the latency ?

1 Kbs = 10^3 bits/sec

1250 bytes = 1250 × 8 bits.

M = 1250 × 8

$L = \frac{1250 \times 8}{56 \times 10^3} + 5 \times 10^{-3}$ sec

$= \frac{125 \times 8}{56 \times 100} + 5 \times 10^{-3}$

$= \frac{5}{28} + 5 \times 10^{-3}$ $= 0.17857 + 0.005$

$= 0.18357$ sec

→ Here, Transmission Delay is contributing the most in the delay.

Q Broadband Connection.

Prop. delay = 50ms, Data Rate = 10Mbps.
Message size = 1250 bytes.
Calculate L ?

$L = \frac{M}{R} + D$

$= \frac{1250 \times 8}{10 \times 10^6} + 50\text{ms}$ $= 0.005$ s.

$= \frac{125 \times 8 \times 2}{100 \times 100 \times 100} + 0.005$

$= \frac{1}{1000} + 0.005$

$= 0.001 + 0.05$

$= 0.051$ sec

$= 51$ ms

Here Propagation delay is the major contributor in delay.

21/1/19

Some of the n/w that was present befor TCP/IP model
① ARPANET

↳ If any of the switching office is destroyed then whole N/w will be fragmented or fail.

→ IMP → Interface message processor.
→ ARPANET later evolved into the internet.
↳ DNS convert host name into IP address
(Domain Name system)
↳ Each ISP has a backbone N/W.
If the host is not in ISP then ISP will handover the packet to its backbone N/W.
→ Point of presence.
↳ POP again converts the analog s/g to digital s/g.
↳ NAP (Network accesspoint).
If Backbone N/W doesn't have or recognise receiver it will send the packet to NAP (which connects many backbone N/W) and it will further send it to another backbone N/W (according to the Router table entry in NAP)

How the Internet works

↳ Backbone N/W is a collection of many regional ISP.

↳ Selecting a spectrum range that is not harmful for human health is an issue in wireless N/W.

↳ Telecommunication service provider should have a particular standardization as there are many service provider, so it is difficult for a user in one service provider who want to communicate with user in another service provider, to remember all the names.

Two bandwidth in which wifi works:-
① 802.11 a|b|c = 2.4GHz B.W.
② 802.11 n|an = 5GHz B.W.

IEEE → Institute of Electrical & Electronics Engineers.

802.15 → Bluetooth (Personal area N/Ws)

↓ → means not used anymore.

pico -10^{-12}
femto -10^{-15}
atto -10^{-18}
zepto -10^{-21}
yocto -10^{-24}

—— x ——

24/1/19

Physical Layer

→ Deals with physical aspects of the network i.e., Wires, fibres, etc.

Wires

→ Needs to be twisted so that the signal of one wire do not interfere with the other wire (or each don't cancel out each other)

↪ In twisted wire case, we need amplifier to amplify the weak signals.

→ Reliability & low cost → (continuely to be used in future)

↪ More thick wire and more bits or data will be transmitted

→ Category 3 → less twisting
→ less distance

Category 5 → more twisting
→ for larger distance

Coaxial cables

- Wires are insulated from outside.
→ 50 ohm cable → digital transmission.
75 " " → analog "

Coaxial cables have been replaced by optical fibre.

Optical fibres

→ Total internal reflection.
→ High data rate transmission 50 Gbps.
→ Without any amplification

Wireless Transmission

→ If we are using x-ray or gamma ray for wireless transmission then it is harmful for people.

Radio Transmission

→ Omnidirectional.
→ Bounce off by obstacles.
→ Signals may get absorbed (in rain for eg).

Some licence bands and are own by government
→ To use this band, we need permissions from the government

Some bands are open-source and are known as Industrial, scientific and Medical (ISM) band.

Our device works in bandwidth 83.5 MHz.

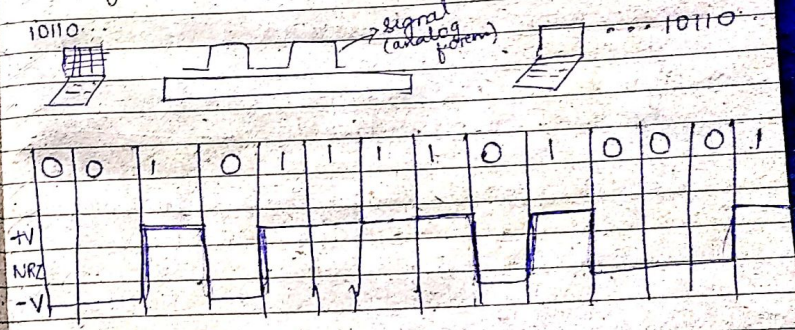
Routers can operate in two frequency bands → 83.5 MHz and 125 MHz

Latency → Once you send the signal to the satellite, the time elapsed from sending to receiving back is known as latency
→ Eg:- of GEO is GPS, which are using 10 satellites.

LEO - 50
MEO - 20
GEO - 3
} No. of satellites

25/1/19

Signals and Modulation

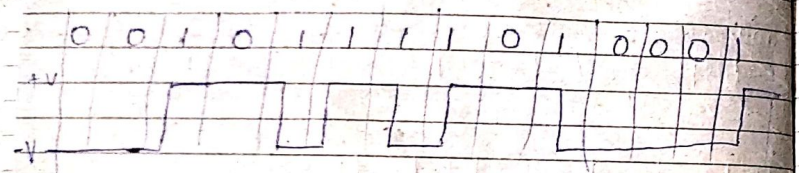


0 - will be represented by -V
1 - will be " " +V
NRZ = Not return to zero.

There are some problem with this scheme (NRZ).
① If we have some signal like.
1 0 0 0 0 0

If we have long runs of 0 or 1, it is very difficult to identify one bit from another (if we have same bit consecutively).

If you use a scheme where there will be transition in bit itself then we will be able to solve the above problem. This scheme is known as NRZI.



→ If we will only have transition if we encounter a bit 1 (And that transition will be in between)

This solves the problem when we have consecutive multiple 1's like (01111011111). But for consecutive 0's we will have to follow some other scheme.

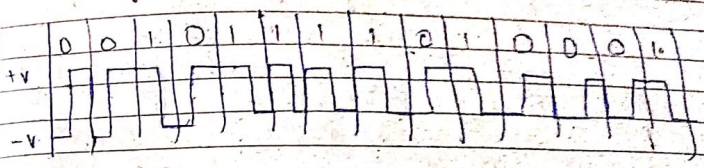
NRZI (Non return to zero Invert).

Manchester Encoding:

Here we will have transition for every bit.

0 → transition from low to high.

1 → transition from high to low.



Problem with this scheme:

* Manchester Encoding needs more bandwidth (because having more no. of transition) than other encoding schemes such as NRZ and NRZI, since it has more frequent transition among the bits.

4B/5B encoding:

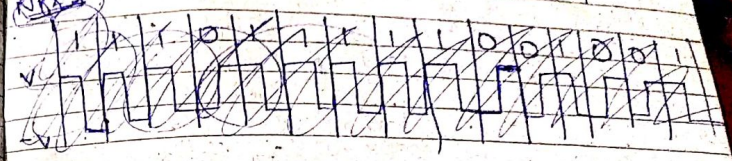
→ This is another encoding scheme.

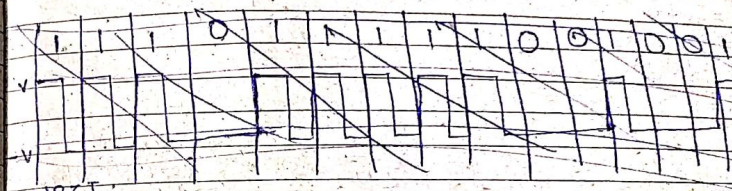
4-bit data signal	5-bit code
0000	11110
0001	01001
0010	10100
0011	10101
0100	01010
0101	01011
0110	01110
0111	01111
1000	10010
1001	10011
1010	10110
1011	10111
1100	11010
1101	11011
1110	11100
1111	11101

→ In this scheme 4-bit signal is encoded as 5 bit code.
→ This is unique code.
→ At max you can have two consequetive zeros in this scheme (not more than that)
→ We can generate over 5-bit code (but remembering the constraint of consecutive zeros).

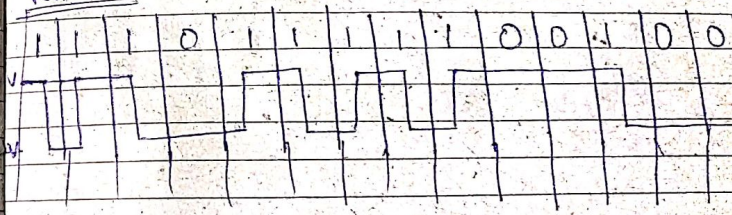
message bit :- 1111 0000 0001
11101 11110 01001

NRZ-I





NRZ-I



Method :-

(*) Invert signal on level 1 and keeps same on level 0 (after encoding 4 bit signal to 5-bit code)

→ In 5-bit code we have 32 combinations, but we have used only 16 of them as (24-16) so other 16 codes are used for specific purpose.

11111 ⇒ Used for idle line.

00000 ⇒ Line is dead

00100 ⇒ Halt transmission.

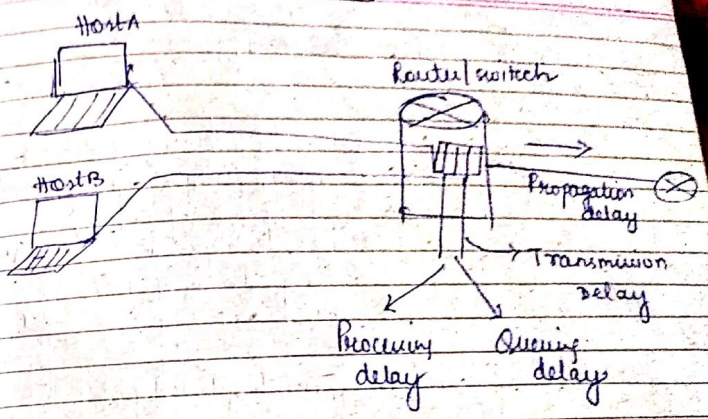
We still have 16-3 = 13 codes; (Out of these 13, 7 are invalid codes (because in those 7 codes we have more than 2 consecutive zeros for zeros).

→ Remaining 6 codes are various control symbols.

— x —

25/1/19

Tutorial - 3



→ The routers when received packets from hosts, it check whether the packets in correct format or not; so it consumes some time known as processing delay

→ When there are multiple hosts, then routers receives packets from multiple host and the routers put these packets in queue and this packet have to wait for some time till it is put to channel. It is Queuing delay.

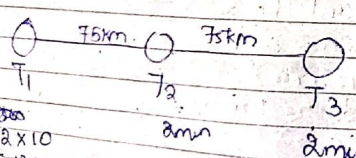
Transmission delay
Size of packet = L bits.
Transmission delay = L/R {R = data rate}

End to end delay = propg. delay + transmission + processing + queuing delay.

Q A caravan is travelling towards destination. Assume a propg. speed of 100 km/hr. There are 10 cars in total.

@ Suppose the caravan travels 150 km beginning in front of one toll booth, passing through second toll booth and finishing just after 3rd toll booth. Assuming that the TB can service one car every 12 seconds, what is end to end delay.

Sol L = 10



Processing delay at each toll booth
= 2 min.

Propg. delay between T_1 and T_2
$= \frac{45}{60} \times 60 = 45$ min.

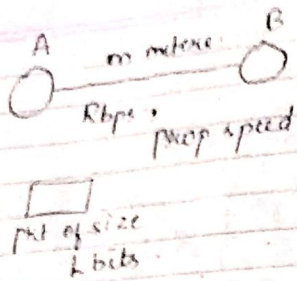
Prop delay b/w T_2 and T_3 = 45 min.

Total delay = 2 + 2 + 2 + 45 + 45
= 90 + 6
= 96 minutes.

(ii) For 8 cars

$T_1 = 8 \times 12 = 96$ sec
= 1.5 min. and 6 sec.
T_2 = 11
T_3 = 11

Total delay = 45 + 45 + 1.5 + 1.5 + 1.5 + 18 sec
= 94 min + 30 + 18 sec
= 94 min 48 sec



① Express the prop delay, d_{prop} in terms of m and s.

$d_{prop} = m/s$ sec.

② Determine the transmission time of pkt, d_{trans} in terms of L and R.

$d_{trans} = L/R$ sec.

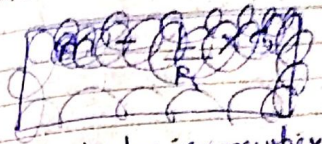
③ Ignoring processing & Queuing delay, obtain exp. for end to end delay.

$= m/s + L/R$.

④ Host A starts transmission at $t = 0$ sec. At time d_{trans}, where is the last bit of pkt?

It will have just finished putting into the link

⑤ Suppose $d_{prop} > d_{trans}$. At time $t = d_{trans}$ where is the first bit of the pkt?



→ It will be somewhere in the link and have not reached host B.

⑥ $S = 2.5 \times 10^8$, $L = 120$ bits, $R = 56$ kbps.

Find the distance (m) so that $d_{trans} = d_{prop}$

$\Rightarrow L/R = m/s$.

$\Rightarrow \frac{120}{56 \times 10^3} = \frac{m}{2.5 \times 10^8}$

$\Rightarrow m = \frac{120 \times 2.5 \times 10^8}{56 \times 10^3} = \frac{75 \times 10^5}{14}$ m.

$= 535.7$ km

≈ 536 km.

— x —

21/1/19

Q1 Consider sending real time voice from Host A to B over a packet switched network. Host A converts analog voice to digital 64 kbps bit streams on the fly. Host A then groups the bits into 56-byte packets. There is one link between Host A & B; its transmission rate is 2Mbps and its propagation delay is 10msec. As soon as Host B receives an entire packet, it converts the packets bit to an analog signal. How much time elapses from the time a bit is created until the bit is decoded at Host B. (as part of analog signal at Host B)

56 byte

$\frac{56 \times 8}{64 \times 10^3} = \frac{7}{8} = 1$

$d_{prop} = 10msec$

$d_{trans} = \frac{1143 \times 28}{2 \times 10^6} = \frac{1143 \times 28}{10^3} msec$

$= 32.004$

Processing time $= \frac{56 \times 8}{64 \times 10^3} = 7msec$.

Time required to transmit packet is

$d_{trans} = \frac{56 \times 8}{2 \times 10^6} = 0.224 msec$

$d_{prop} = 10msec$

Total delay $= 7msec + 10msec + 0.224msec$

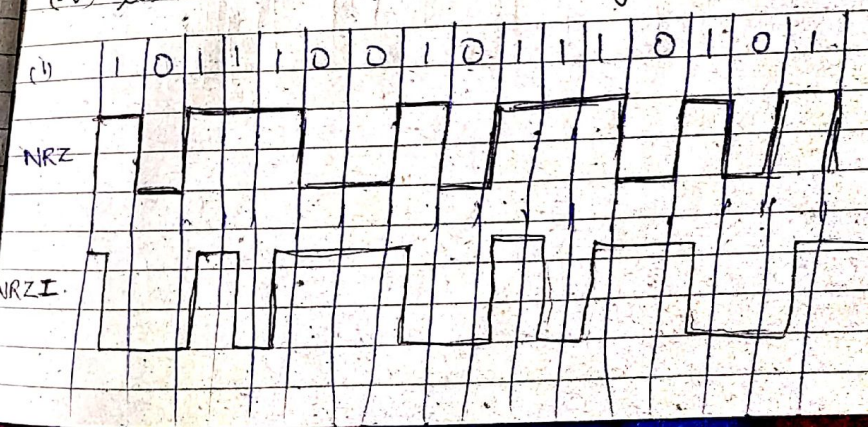
$= 17.224 msec$ → Ans

Q. Consider the following digital signal to be transmitted across an analog transmission media:-

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

(i) Encode this signal using NRZ.
(ii) " " " using NRZI.
(iii) Use the 4B/5B to encode.
(iv) Use the manchester encoding.

(i)

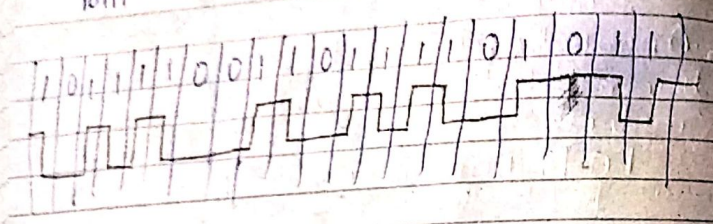


(iii) 4B/5B code

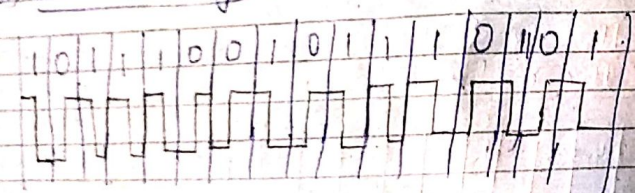
4B 1011 1001 0111 0101

5B ↓

10111 10011 01111 01011



(iv) Manchester coding



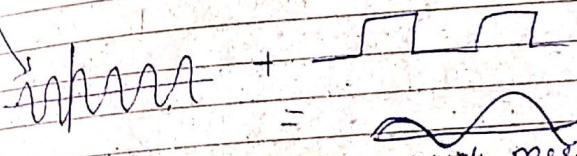
31/1/19

Base Band Modulation

↳ Signal is sent directly to wire
↳ These signals can't propagate well on fibre/wireless

Pass Band Modulation

↳ Signals are modulated with carrier signals
↳ carrier is simply a signal oscillating at a desired frequency. (2.4GHz for 802.11).



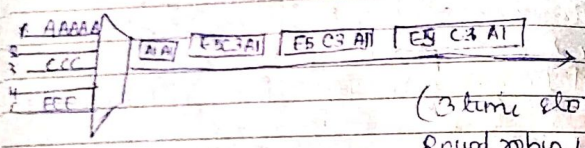
→ We superimpose carrier with message signal to prevent loss during transmission.

Time division multiplexing

→ Each user is alloted a time slot in the frame.
→ No:- of users = no:- of time slots (in each frame)

→ In synchronous TDM, each particular user is assigned a particular time slot. He can't use the time slot of another user. So we move on to asynchronous TDM.

→ In asynchronous TDM if we have n senders then we can have m time slot frame where m < n.
→ Here, servicing is done in round robin fashion.
→ Here we have delimeter which is the user not to recognise whose data is this.



(3 time slots frame)
Round robin fashion.

Frequency Division multiplexing

→ Here the bandwidth is limited to each user.
The whole channel is divided into multiple sub-channel.
Each data is send simultaneously, at same time.

Circuit switching & Packet switching

Nyquist Limit (1924)

The maximum data rate of a link (noiseless) is 2B where B is the bandwidth of the channel.

① $R = 2B$ bits/sec

② $R = 2B \log_2 V$ bits/sec, where V is the signal level.
where $V > 2$.

When $V = 2$
$R = 2B \log_2 2$
$= 2B$ (same as previous one)

Q Assuming $B = 3KHz$ and binary signal the maximum data rate?
$R = 2B$
$= 2 \times 3KHz$
$= 6KHz$ or 6000 bits/sec

Q Noiseless channel (bandwidth) = 5KHz
Signal level = 8.
$R = 2B \log_2 V$
$= 2 \times 5 \log_2 8$
$= 2 \times 5 \times 3$
$= 30KHz = 30000$ bits/sec.

Shannon Capacity (For Noisy channel)

- How many levels we can distinguish depends on signal to Noise ratio (SNR).
- SNR is represented on log scale in decibels.

$$SNR = 10 \log_{10}(S/N)$$

- $S/N = 100$

  then $SNR = 10 \log_{10}(100)$
  $= 10 \times 2$
  $= 20$ dB.

11/8/19

- It specifies the max. info carrying capacity of the given channel (with Noise) with a bandwidth of B.

$$C = B \log_2 (1 + S/N) \text{ bits/sec.}$$

C < Actual data rate.

Q. Channel of 3000 Hz
SNR = 30 dB
Calculate C?

$3000 \log_2 (1+30)$

~~3000~~ X

$3.0 = 10 \log_{10}(S/N)$
$\Rightarrow 3 = \log_{10} S/N$
$\Rightarrow S/N = 1000.$

$C = 3000 \log_2 (1+1000)$
$= 3000 \times 9.9672$
$= 29901.7$ bits/sec.
≈ 30000 bps. ≈ 30 kbps.

∴ The highest data rate with which we can transmit is 30 kbps (not higher than this).

④ Wire / Fibre

- Engineer to have requisite SNR and B.
- Can fix data rate.

B ↙ Here we can fix it.

⑤ Wireless.

- Given B, SNR varies greatly
  eg. upto 60 dB ⇒ $S/N = 10^6$.

→ Cannot design to have fix data rate for channel.

Q. ~~If the bandwidth~~
Assume the following characteristics of a communication link:
bandwidth = 600MHz
Channel capacity = 1Gbps.
What is the required SNR at the receiver to achieve this channel capacity. Express in dB.

B = 600MHz
ADR = 1000 Mbps.

$1000 = 600 \log_2(1 + S/N)$

$\Rightarrow \log_2(1 + S/N) = \frac{10}{6}$

$\Rightarrow$ ~~$\log_2(1 + S/N) = 1.175 - 1$~~

$\Rightarrow$ ~~$S/N = \text{anti}\log_2 \frac{10}{6} - 1$~~

$S/N = 2.175$

$SNR = 10 \log_{10}(2.175)$

$= 10 \times 0.337$

$= 3.37$ dB. → Ans.

Q. CAT-5 twisted pair cable has a bandwidth of 100MHz. We would like to transmit at a bit rate of 500Mbps. Is a signal to noise ratio of 30 dB enough to reliably transmit this information? Why or why not?

B = 100MHz.
Bit rate = 500Mbps.
we want to transmit.

$C = 100 \log_2(1 + 1000)$

$\Rightarrow C = 100 \log_2(1001)$

$= 100 \times 9.9672$

$= 996.72$ Mbps.

$30 = 10 \log_{10}(S/N)$
$\Rightarrow 3 = \log_{10}(S/N)$
$\Rightarrow S/N = 10^3$.

Yes, as $B < C$. then we can transmit it.

Q. What is the minimum signal-to-noise ratio in dB, that must be maintained in order to transmit a 600Kbps signal over a medium with a bandwidth of 20,000 Hz?

Sol $600 \times 10^3 = 20 \times 10^3 \log_2(1 + S/N)$

$\Rightarrow \frac{600}{20} = \log_2(1 + S/N)$

$\Rightarrow 30 = \log_2(1 + S/N)$

$\Rightarrow 2^{30} \Rightarrow 1 + S/N$

$\Rightarrow 10^9 - 1 \Rightarrow S/N$

$\approx 10^9$

$SNR = 10 \log_{10}(10^9)$

$= 9 \times 10 = 90$ dB.

↳ Ans

12/17

Tutorial

Q. We are given a medium that can reliably transmit frequencies b/w 0 & 25,000 Hz. Is it possible to transmit 200kbps of info along this line? If so, then describe a method and any conditions that must be satisfied. If not, explain why?

Assuming it to be a noiseless channel.

$R = 2B$ (binary signal).

$= 2 \times 25000$

$= 50{,}000$ bits/sec.

$= 50$ kbps.

No, it is not possible to transmit 200kbps as max rate with binary signal is 50kbps.

$R = 2B \log_2 16$. for 16 levels.

$= 8B$.

$= 8 \times 25000$

$= 200$ kbps.

For levels ≥ 16, we can transmit 200kbps across this channel.

With multi level s/g.

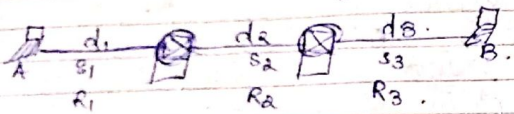
$C = 2B \log_2 V$

$\Rightarrow 200 = 2B \log_2 V$

$\Rightarrow 200 = \log_2 V$ $\Rightarrow \log_2 V = 4$

$V = 2^4 = 16$

Length = d_i
propagation speed = s_i
transmission rate of link i = R_i



Processing delay = d_{proc}.
Packet length = L
Prop. delay = $\frac{d_1}{R_1}, \frac{d_2}{s_2}, \frac{d_3}{s_3}$

Total end to end

$$= \left(\frac{d_1}{s_1} + \frac{L}{R_1}\right) + \left(\frac{d_2}{s_2} + d_{proc} + \frac{L}{R_2}\right)$$

$$+ \left(\frac{d_2}{s_3} + d_{proc} + \frac{L}{R_3}\right)$$

$$= \left(\frac{d_1}{s_1} + \frac{d_2}{s_2} + \frac{d_3}{s_3}\right) + 2d_{proc} + L\left(\frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3}\right)$$

↓ Total propagation delay ↓ Processing delay. ↓ Transmission delay

L = 1500 bytes = 1500 x 8 bits.
$s = 2.5 \times 10^8$ m/s.
R = 2Mbps
d_{proc} = 3msec.
$l_1 = 5000$ km $l_2 = 4000$ km, $l_3 = 1000$ km

Total end to end delay = Propagation delay + d_{proc} + Trans. delay

Prop. delay $= \frac{5000}{2.5 \times 10^5} + \frac{4000}{2.5 \times 10^5} + \frac{1000}{2.5 \times 10^5}$

$= \frac{5}{250} + \frac{4}{250} + \frac{1}{250} = \frac{10}{250} = 0.04$ sec

Processing delay
$= 2 \times 3 = 6$ msec
$= 0.006$ sec.

Transmission delay $= \frac{3 \times 1500 \times 8.4}{8 \times 10^6}$

$\frac{60 \times 100 \times 3}{10^6} = \frac{18}{10^3} = 0.0$ se

Total end to end delay

$= 0.04 \text{ sec} + 0.006 \text{ sec} + 0.018 \text{ sec}$

$= 0.064 \text{ sec}$

$= 64 \text{ msec}$ → Ans

Q. given $d_{proc} = 0$.

Supposing packet switch does not store & forward packets but instead immediately transmit each bit it receives,

So we will only have transmission delay at Host A, not at the packets.

$$\text{end to end} = \frac{L}{R} + d_1/s_1 + d_2/s_2 + d_3/s_3$$

$= 0.04 \text{ sec} + 0.006$

$= 40 + 6 \text{ msec}$

$= 46 \text{ msec}$ → Ans.

Queuing delay =

Q. 4-transmission delay + ½ of 1 transmission delay

$= 4 \times \frac{1500 \times 8}{2 \times 10^6} + \frac{1}{2} \times \frac{1500 \times 8}{2 \times 10^6}$

$= \frac{120 \times 2 \times 100}{10^6} + \frac{30 \times 100}{10^6}$

$= 0.024 + 0.003 \text{ sec}$

$= 27 \text{ msec}$ → Ans

Generalised,

Queuing delay

$= n \times \frac{L}{R} + \frac{L - x}{R}$

$= \frac{nL + (L - x)}{R}$ → Ans

$= \frac{(n+1)L - x}{R}$

4/2/19

Two techniques to build a N/w:
① Circuit switching (Telephone network)
② Packet switching (Computer)

In circuit switch N/w → Path is fixed → data arrive in order.
In packet switch → the receiver at the receiver end have to assemble the packet as data does not arrive in order.
→ Path is not fixed.

→ In packet switch is more fault tolerant as path is not fixed so if one path is faulty then we can use another path to communicate here.

→ Congestion control is good in circuit switch N/w as path is assigned previously But if more traffic is there then all the bandwidth is reserved, then congestion can happen in circuit switching also.

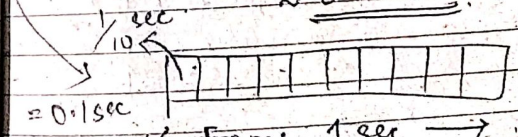
→ In circuit switch, bandwidth is fixed for particular sender and receiver. So if some one peer arrives it can't use it and bandwidth will be wasted
→ Service wise → circuit switching is better than packet switch as path is fixed so the packet reaching receiver will be reliable i.e. reliable link.

Ckt Switch V/S Packet Switching

* Assume a 1 Mbps link
* Users alternate b/w idle periods and data transmission at constant rate of 100 kbps.
* Further, user is active only 10% of time.
* With circuit switching, 100 kbps must be reserved for each user at all time.
* ckt switch with TDM, with frame of 1 sec.
* For pkt switching, if there are 35 users, with each user transmitting with probability (0.1), then the prob. more than 10 users will be active :-

$$= \binom{35}{10} (0.1)^{10} (1-0.1)^{35-10}$$

$$\approx 0.0004$$



→ Here in circuit switch N/W we can max. have 10 users
because each user need data rate 100Kbps to transfer
and maximum capacity of link = 1Mbps

For 10 users = 10 × 100 kbps
= 1 Mbps.

so if we have more than 10 users, then it will exceed the capacity of link which is 1Mbps.

But in packet switch the prob. of having active users is very low 0.0004
so it can accomodate more than 10 users at a time.

---

④ Suppose there are 10 users and only one user generates a one thousand 1,000 bits packets while other are silent.

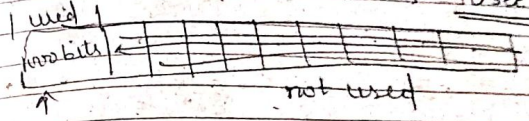
⑤ Under TDM ckt switching with 10 slots per frame and each slot consisting of 1000 bits.

↳ What is the time req. to transmit the packet?

sol With a data rate of 100 kbps, time req. to transmit 1000 bit pkt is

$$= \frac{1000}{100 \times 10^3} = 10\,ms.$$

Total time = $10^3 \times 10\,ms = 10\,sec$



by 1 user only.

So, 10 sec is very large number in case of Networking. As 9 slots are always remain unused. So we are waiting 9 slots every time in packet switching N/W.

In pkt switch, the user has access to entire bandwidth = 1Mbps.
Time required to transmit 1000 bit paket is

$$= \frac{1000}{10^6} = 1\,ms.$$

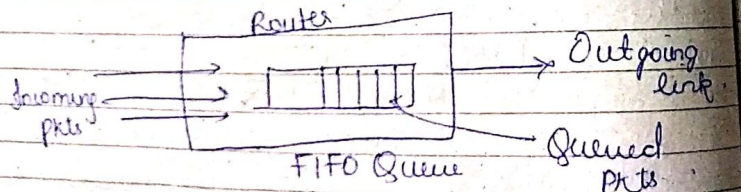
Total time = 1000 × 1 msec
= 1 sec.

→ In pkt switch time required is very less than ckt switch because we are using the entire bandwidth as the number of active users at a time in pkt switch is very less.
In real time, pkt switch consumes less time than circuit switch and hence it is better for transmission.

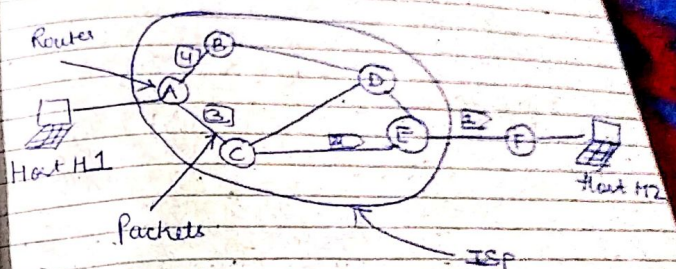
Packet Switched Network

1. Datagram Model :- Forwards pkts based on host destination address.
2. Virtual circuit model :- forwards pkts based on virtual circuit Nos.

Both uses Store and forward pkt switching mechanism



Datagram Model



A's forwarding Table

A	
B	B
C	C
D	B
E	C
F	C

T1

C's forwarding table

A	A
B	A
C	
D	E
E	E
F	E

→ Forwarding Table is not static. It may change with time.

→ The change can be for multiple reasons for congestion in the N/w, or failure of some link, etc.

At time T_1 the forwarding table is shown previously
At time T_0, the forwarding table might be changed like -

A's forwarding table

A	
B	B
C	C
D	B
E	B
F	B

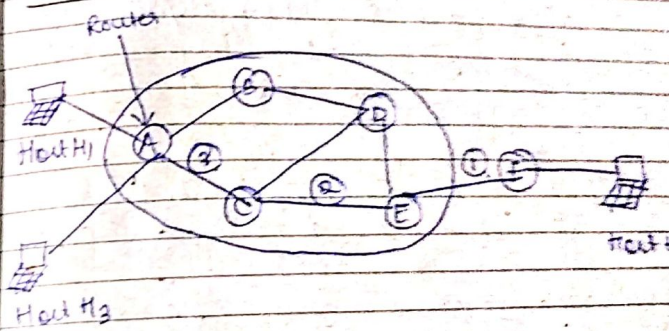
→ changed wrt to time T_1

The router will use the forwarding table to make the routing decision in datagram model.

→ forwarding table tells if one router wants to send the pkt to another table, then through which router it will send the packet

For A's forwarding table, if A wants to send the packet to B, then it can send directly to B (In right column it is superseded)

Virtual Circuit Model



① Connection Establishment
↳ Path is chosen, circuit info stored in routers.

② Data transfer, circuit is used
↳ Pkts are forwarded along the path

③ Connection teardown; circuit is deleted
↳ Circuit info is removed from routers

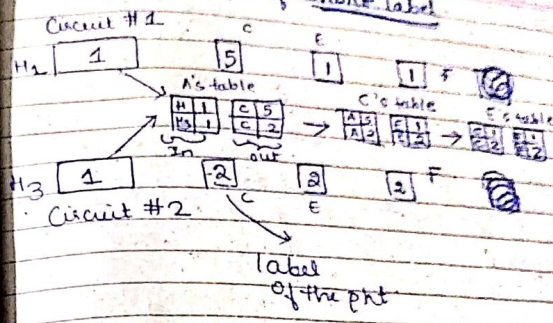
In circuit switching, a link is exclusively reserved for only a single pair of sender and receiver.
But in Virtual circuit switching, the link or the path is exclusively defined or remain fixed but there is statistical multiplexing (link can be shared by more than one pair of users, but path is fixed same as in circuit switching).

→ In virtual CS, packet contain short labels to identify the circuit.

Circuit #1
① H_1 — A — C — E — F — H_2
② H_3 — A — C — E — F — H_2
Circuit #2

(Multiple ckts sharing same link).

Now how C will identify whether pkt is coming from H_1 or H_3. Here we will make use of the concept of short label



The IN table represent, from where the pkt has arrived and what it is label when reached.
The OUT table represent, when that particular router sends the pkt to succeeding router, it change the label of the packet to send it further. To differentiate b/w the two packets, we give them different label.

This model is mainly used by MPLS.
↓
Multi-protocol labels Switching (used by ISPs).

The datagram model is used by Internet.

With this, our discussion on physical layer is completed.

Link layer

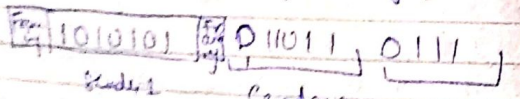
N/w layer / Link layer

When the packet comes from the network layer it is of the form 1010100- - - - - -

→ It is a continuous bit stream sending by many senders.

So the link layer delimits the stream of one sender from another sender.

→ In link layer we do framing.



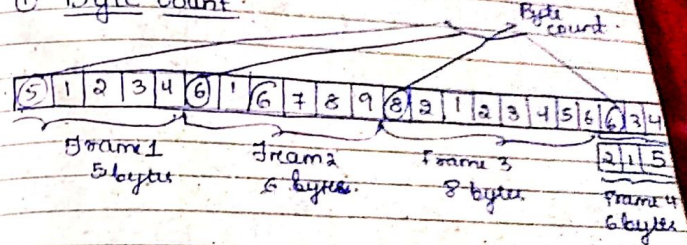
Framing :- Delimiting the start and end of frame.

Framing is a division of stream of bits received from n/w layer into a manageable unit called frame.

3/08/19

Different methods of framing are :-

① Byte count :



Starting byte always indicate how many bytes are there in the frame.

Drawback :-

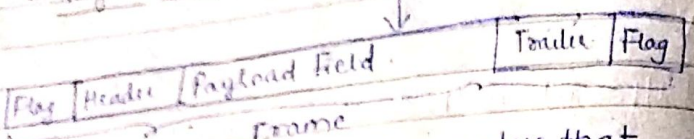
→ If you loose the byte count somewhere then the whole thing can go wrong. The synchronization will go wrong.

Suppose instead of ⑥ if there will be ③ → wrongly counted.

Then further all the count goes wrong.

⑧ Byte stuffing

N/w layer



① Have a special byte flag value that means start/end of the frame.

② Stuff the flag inside the frame with an escape code.

③ Escape the Escape code inside the frame (data payload).

→ Flag is some 1 byte value (Eg:- 10000001).

→ In Trailer value, we have some error detection and correction method.

→ If flag is in the payload field then it is difficult to mark the start and end of the field.

→ To overcome this, we have a flag byte inside the payload field we will write Esc infront of the flag bit value.

→ This differentiate the flag byte from the same payload field having flag value.

→ If we have Esc also in the payload field, then we will write another Esc infront of that Esc field.

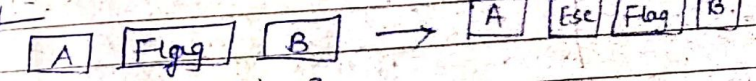
Esc Esc flag

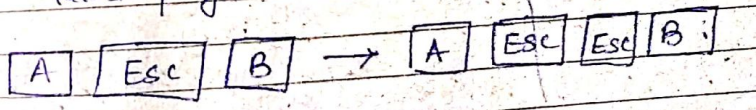
→ In general we can say, the unescape flag value, marks the start and end of the frame.

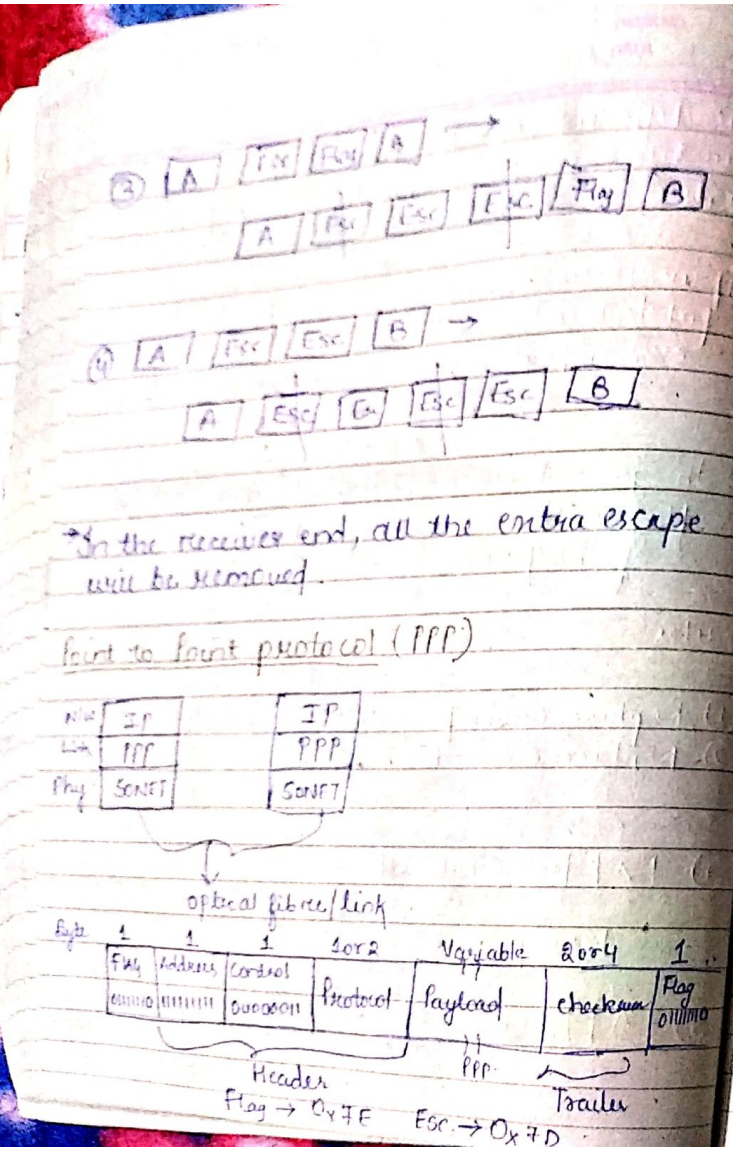
Rules:

① Replace each flag in data with Esc.
② Replace each Esc in data with Esc.

Eg:-

① 

② 



→ In the receiver end, all the extra escape will be removed.

Point to Point protocol (PPP)

optical fibre/link

Flag → 0x7E Esc → 0x7D

→ This Link layer protocol (PPP) use the byte stuffing

→ It uses SONET Network for channel.

→ The Header info is inserted by PPP protocol.

* The byte stuffing is done by stuffing bytes.

Now we will see bit stuffing.

② Bit stuffing

(*) On transmit after five 1s in a data insert a 0.

(*) On receive, a 0 after 5 1's is deleted

Databit :	0110	1111	1111	1111	1111	0011
Transmitted bit with bit stuffy	0110	1111	10111	11011	11101	0011

For bit stuffing the flag byte is :- 01111110.

→ This is done because in flag byte we have six consecutive 1's. So if in data we have 5 consecutive 1's we insert a 0 to ensure that it is not a flag byte.

000111111100 111110 1000

↓ Bit stuffing

0001111101100 111110 01000

→ output

stuffed bits.

At the receiver end,

0001111101100 111110 01000

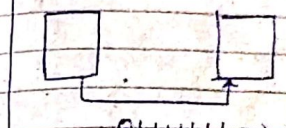
These two bits will be removed at the receiver end.

08/08/19

Error Coding:

- Error in some bits while reception at the receiver's end.

What can we do?



→ Error detection.
→ Error correction
→ Retransmit.

→ Very simple scheme to detect the error is sending two copy of the same data. But the problem with this scheme is if both copy are wrong and same, then the receiver will think that the data received is correct.

Other detection techniques are:-

① Parity Bit

↳ Take D data bits add 1 check bit that is sum modulo 2 of all data bits

Eg:

(a) 1C 1100 → parity bit
data bits

sum = 3

3 Modulo 2 = 1 - parity bit

At receiver end, always sum % 2 = 0 Then the data bits received are correct.

Drawback:-

→ This method does not work if there is a 2 bit error.

For eg:- If for the above data bits

changed bits = 1011001
changed

Still sum % 2 = 0

6 % 2 = 0 at receiver end

② Checksum:-

Sum up data in N-bit words

- Widely used in TCP/IP/UDP

For Internet, N = 16.

| 1500 bytes | 16 bit |

Internet checksum

→ Error detection and correction is done in trailer part of the frame.

Eg:- Data = 0001 f203 f4f5 f6f7

① Organize data in n-bit words (for Internet 16)

0000 0000 0000 0001 ← 0001
0010 0000 0011 ← f203
0100 1111 0101 ← f4f5
0110 1111 0111 ← f6f7
0000 ← ② Adding 0000 to it

1101 1101 1111 0000

→ 0010 ③ Sum all the bits

④ Sum the bit again with the carry bit

1101 1101 1111 0010

⑤ Take the complement

0010 0010 0000 1101
2 2 0 D

↓ checksum value.

⑥ Replace the 0000 we have added, ~~with~~ in data bits with checksum.

⑦ Send the data bits along with checksum to the receiver.

receiver.

```
0001 + 0000 0000 0000 0001
f203 - 1111 0010 0000 0011
f4f5 - 1111 0100 1111 0101
f6f7 - 1111 0110 1111 0111
2200 - 0010 0010 0000 1101
       1111 1111 1111 1101
```

Again do all the same steps at the receiver end.

→ Sum all the data bits.

→ Sum the carry bits with the result.

```
  1101 1101 1111 0000
+ 0010 0010 0000 1101
  ---------------------
  1111 1111 1111 1101

  1111 1111 1111 1101
             0010
  ---------------------
  1111 1111 1111 1111
```

→ Again complement the bits.

```
0000 0000 0000 0000
 0    0    0    0
```

→ If we get all zeros, then our data bit received is correct.

→ Even though it is better than parity bit method but it have some drawbacks. (See yourself).

→ Best method is CRC.

② Cyclic Redundancy Check (CRC).

→ Given 'n' data bit generate 'k' check bits such that 'n+k' bits are evenly divisible by generator C.

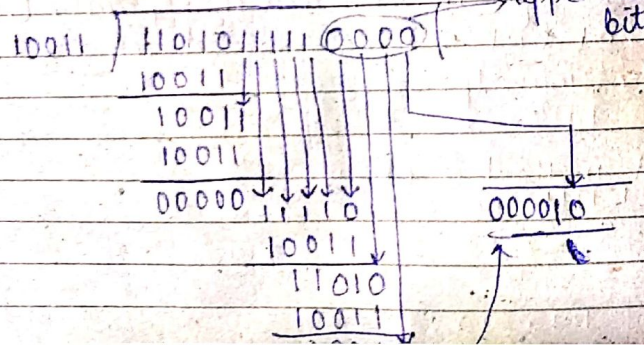
→ Based on mathematics of finite fields in which we represented a polynomial.
Eg:- 10011010 is $x^7+x^4+x^3+x^1$.

Eg:- Data bit ; 1101011111

(c) Divisor / Generator : 10011
↓
x^4+x^1+1

Find CRC:

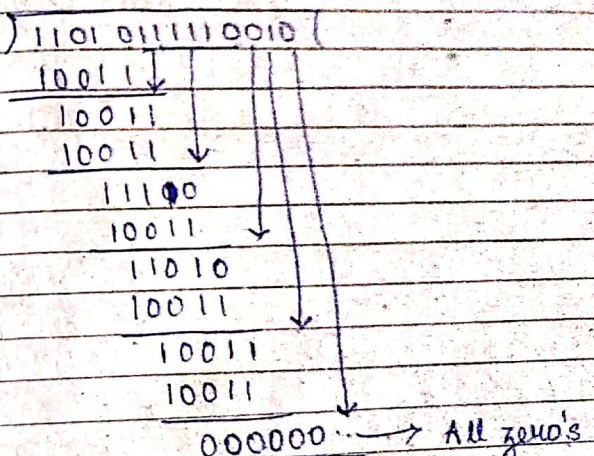
Appeld bits = C-1 (zeros).



Now our transmitted bit is :-

1101 0111110010

If at the receiver end, we do division with the same generated then we should get zero. Thus, our msg received is correct.



∴ Our received Msg is correct.

11/09/19

CRC :-

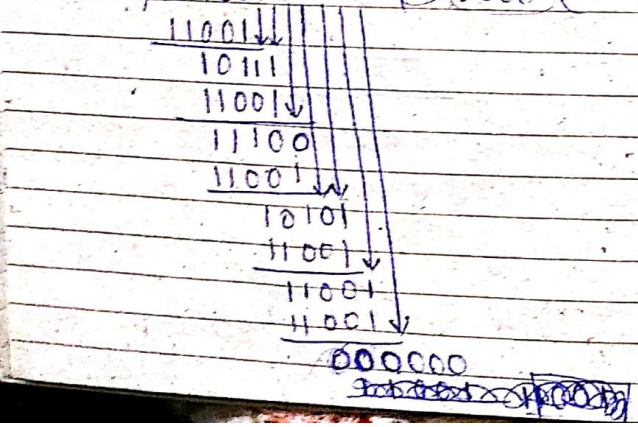
Send procedure :-

① Extend the n data bits with k zeros.
② Divide by the generator/divisor value, C.
③ Keep remainder, discard quotient.
④ Adjust k check bits by remainder.

Q Data bit $x^{11}+x^{10}+x^9+x^6+x^5+x^3+x$
= 111001100110

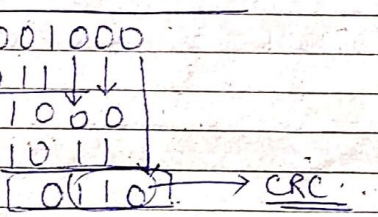
Generator = x^4+x^3+1
= 11001

Find the CRC and the data transmitted



Data transmitted = 111001100110 0000

Q Data bit :- 1001
Generator = 1011



Data transmitted = 1001110

Error Correction

① Hamming code :-
→ Uses extra parity bits to identify and correct single bit error.
→ Only applicable for single bit error code.
→ Parity bits are in power of 2 position.

0	1	1	1	0	0	1	0	1	0	1	0
P_1	P_2	2_3	P_4	d_5	d_6	d_7	P_8	d_9	d_{10}	d_{11}	d_{12}

→ All the power of 2 positions are reserved for parity bit

→ Rest all positions is fixed for data bits.

Now we have to find the parity bits.

P1: checks 1 bit, skips 1 bit, check 1 bit, skips 1 bit

(1, 3, 5, 7, 9, 11, 13 ----)

Rule:- ① Set a parity bit to 1 if the total NO of 1s in the positions that a parity bit checks is odd.

② If No of 1s checked by parity bit is even, set parity bit to 0.

P1: ? 1 0 1 1 1 = Even.

So set p1 = 0.

P2: Check 2 bits, skip 2 bits, check 2, skip 2

P2: ? 1 0 1 0 1 = Odd.

P2 = 1

P4: check 4, skip 4 ---

P4: ? 0 0 1 0 = odd

Set P4 = 1

P8: check 8, skip 8 ---

P8 = ? 1 0 1 0 = even.

Set P8 = 0.

⇒ Suppose at receiver end, if d_{10} is changed to 1 from 0.

Then how to detect this error?

At receiver: Perform P1 parity check, (don't take P1 into consideration)

P1 = 3, 5, 7, 9, 11

1 0 1 1 1 = Even

So P1 = 0 (✓)

P2 - (Don't consider 2).

3, 6, 7, 10, 11

1 0 1 1 1 = Even

P2 = 0 (X)

P_4 = (5, 6, 7, 12)
= 0 0 1 0 : odd
P_4 = 1 (✓)

P_8 = (9, 10, 11, 12)
= 1 1 1 0 : odd
P_8 = 1 (✗)

To get the position where error has occurred,
Error position = 2 + 8
= 10
This gives that at 10th position error has occurred.

Alternative:

P_8 = 0 1 1 1 0 = 1 (odd 1s)
P_4 = 1 0 0 1 0 = 0 (even 1s)
P_2 = 1 1 0 1 1 1 = 1 (odd 1s)
P_1 = 0 1 0 1 1 1 = 0 (even 1s)

Writing it as $P_8 P_4 P_2 P_1$
= 1010
decimal value = 10
This is the position where error has occurred

15/2/21

Error correction:-

① Hamming code (Not used in practice)
② Convolution code
③ Low density parity check (LDPC) → 802.11, WiMAX, LTE ------
↓
Soft error correction methods.

② and ③ are used in practice and are also widely used.

Error detection	Error Correction
~~① When the N/W is expected~~	①
① More efficient when errors are not expected.	① Needed when errors are expected.
② When errors are large, when they do occur.	② When there is no time for retransmission.

Retransmission:
Enables to recover from losses occuring during the transmission. Loss of frames is common in wireless links (802.11). Without

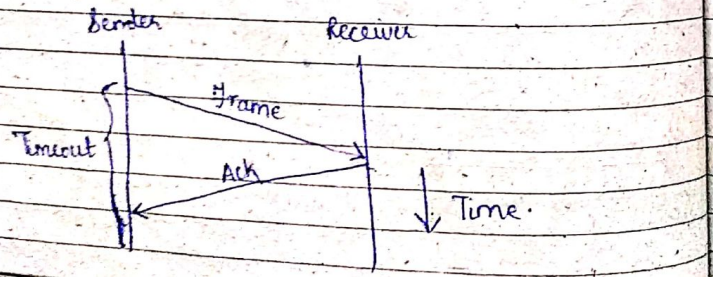
the mechanism for retransmission, many frames will be lost during transmission and Network as a whole would become inefficient.

Two strategies to handle Errors.

① Detect errors and retransmit frame. (Automatic Repeat Request (ARQ)).
② Correct errors with error correction codes (Discussed earlier)
* Hamming code.

↳ ARQ is often used when errors are common or must be corrected.
↳ Used in WiFi & TCP (discussed later)
(It can be used in multiple layer also like link, N/W, Transport, etc.)

Normal operations.



* Sender and receiver are shown as vertical lines. Time runs down the page.

* Ack is received before the timeout. So sender comes to know that frame has successfully been delivered to the receiver.

Now suppose if the frame send by the sender is not received by the receiver, then it will wait for the timeout period to gets its ack. If not get within that period, the communication fails (packet ~~does~~ does not reach the destination).

Issues related to this scheme:-

① Timeout value (TV).

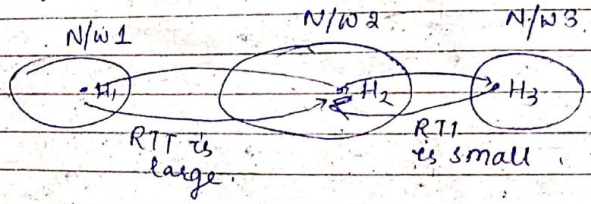
① If TV is small, then by the time Ack will be received by the sender, the timeout value will be expired.

② If TV is big, then if the frame was lost during transmission, then we have to wait all the way down to the timeout value and till then sender remains idle.

RTT is time taken by Ping to reach the receiver from sender and then the Ping ECHO from receiver to sender.
RTT = Round trip time.

The timeout value can be set as :- RTT + Small value

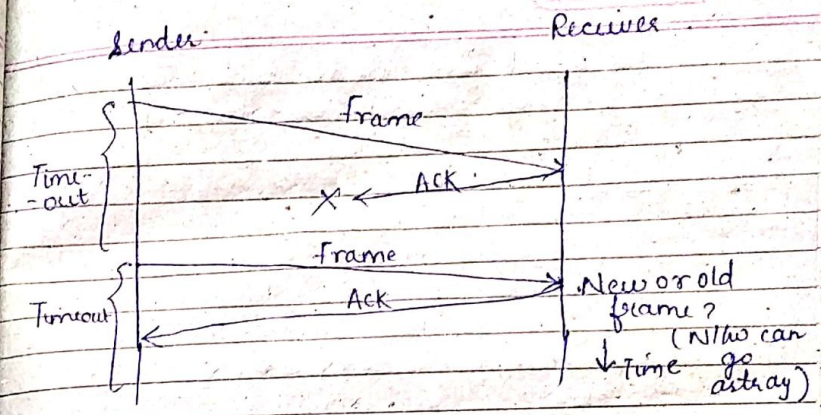
→ If we are in small N/w like LAN then we can set the RTT time fixed.
But for larger geographical area, we can't fix the RTT value.



Thus RTT is variable when we are dealing with N/ws in different geographical area.

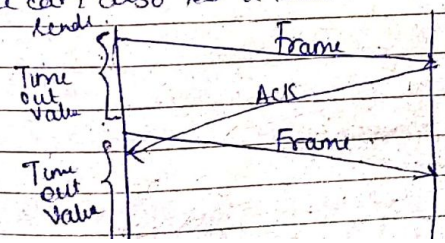
② Duplicate Frames.

→ This is the second issue which arise here.



If the sender send one frame and that frame's ACK was lost. Then send will wait till the time-out period.
After that it can send the frame again.
But now the receiver's will have confusion whether it is the old frame or a new frame.
Then the N/w can go astray.
That is the sender have send one frame but receiver have received two frames.

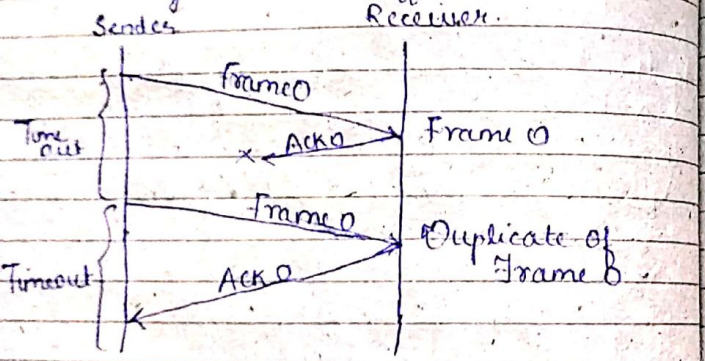
There can also be a situation like :-

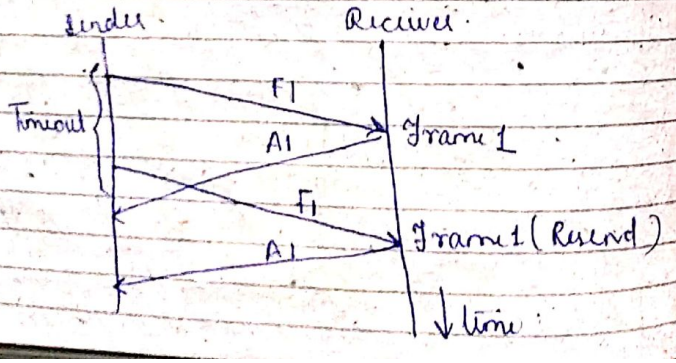


The issues which we have discussed just now can be solved using the following method.

Sequence Numbers

* Frames and ACKs must both carry sequence numbers for correctness.
* To distinguish one frame from the next one, a single bit is sufficient.





This is how stop and wait protocol work. This stop and wait protocol removes the problem of duplicate frames.

Limitation of Stop and Wait protocol

→ Only one single frame can be outstanding at a time. (Till we do not get ack of frame 1 we cannot send another frame)

Q. Given link; R = 1 Mbps; D = 50 ms.

$$RTT = 2D = 2 \times 50 = 100 \text{ ms}$$

⇒ How many frames per second can we send with this RTT value?

$$100 \text{ ms} \longrightarrow 1$$

$$1 \text{ s} \longrightarrow \frac{1 \times 10^3 \text{ bits or frame}}{100}$$

$$= \frac{10^3}{100} = 10 \text{ frame}$$

→ If R = 2 Mbps, then can we send more frame? No, the the no:- of frames doesn't depend on R value.

Q. If each frame is of 10,000 bits then what is the transmission rate?

Transmission rate = 10×10^4 bits/sec

= 100 kbps

∵ we have 1Mbps but we can only achieve 100 kbps in this stop and wait protocol.

To overcome the above the said problem, we move to next protocol (can be used in N/w, transport DLL, etc)

Sliding Window Protocols

① Generalization of stop and wait protocol.

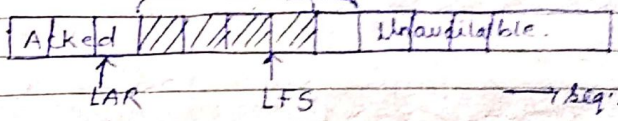
② Allows 'w' packets to be outstanding at any given time.

③ Can send w packets per RTT.

Sender view:

① Sender buffers upto w frames until they are ACK.

- LFS = last frame sent.
- LAR = last ACK Received



As the window size is 5, the sender can send 5 frames without getting any acknowledgement.

LAR → the sender has received the acknowledgement upto that frame.

LFS → The last frame sent by the sender.

When all the window size is filled, then sender cannot send any more frame, till it receives ACK of some other frame. Now, when sender receives Ack for any frame (1st of the window), then the window will slide to one frame. (now the sender have one available frame).

This is why this protocol is known as sliding window protocol.

The sender view is same for both Go-Back N and selective repeat protocol.

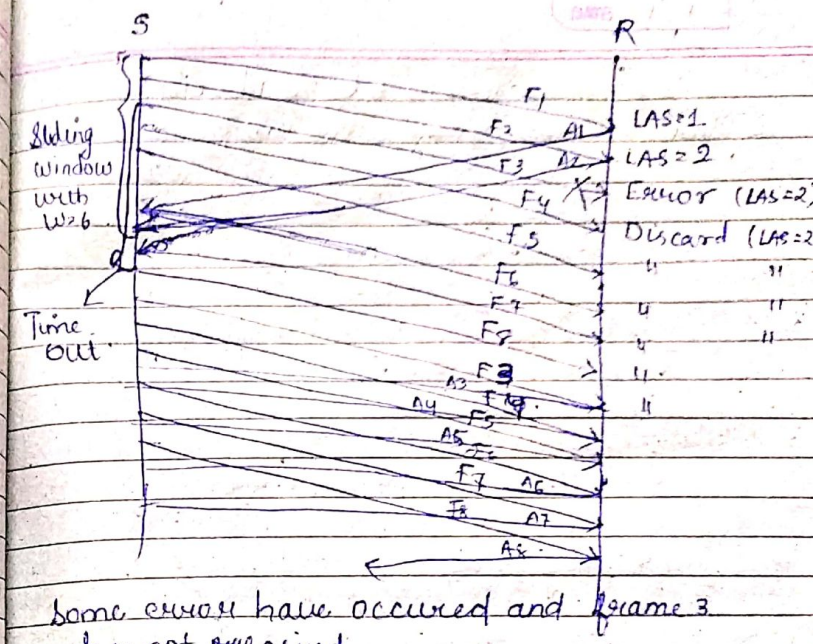
Receiver End :- (Go Back N).

① Receiver only keeps a single buffer for the next frame expected.
- State variable, LAS = Last Ack Sent.

② On receipt of frame :-
- If Seq. No is LAS+1, accept the frame, update LAS to LAS+1, and send ACK for frame with frame Seq. No = LAS+1.
- Otherwise, discard the frame. (No ACK is being sent).

→ It is called Go Back N protocol because all the packets or frame, are retransmitted (or are discarded by the receiver), for one single frame error.

→ Sender has to maintain the buffer, unless the ack has been received for



Some error have occured and frame 3 has not received.
When frame 4 will come at the receiver end, even though there is no error but still it will be discarded as 4 > LAS+1=3.
→ Now, all the frames till 6 will the discarded in the similar manner. But two extra frames 7 and 8 have also been discarded because, when the frame 1 & 2 have been reached correctly (ACK received by sender), the window has been slided. So now the window can accomodate 2 more frame (F7 & F8).

→ Then again, frame 3 will be sent when the time out goes out of value for frame 3.

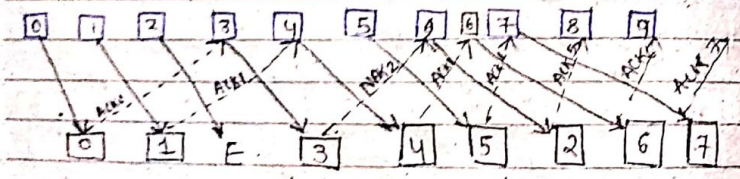
Receiver end:- (Selective Repeat).

① Receiver buffers out of order frames to reduce retransmission.

② ACK conveys highest in-order frame. hints about out of order frames.

③ TCP uses a selective Repeat design.

Sender



In selective repeat receiver have a buffer, and sender have no buffer.

→ ~~Receiver~~ Selective repeat is efficient because there is less retransmission.

When 2nd frame arrives, it encounters an error.

Then when it received 3rd frame, it send NAK2 and store it in its buffer.

When it received 4th & 5th frame it send ACK 1 (Highest acknowledged frame) and store 4th & 5th frame in the buffer of receiver.

→ When it receives 2, then it send Ack 5, ie, all the frames till 5 is acknowledged, as till 5th frame, it is stored in buffer.

→ Then further, transmission occurs normally.

—×—

18/8/19

How do you decide this value of w?

→ We have a link whose data rate is either B or R.

D = propagation delay

2D = RTT.

If size of the frame = S.

and the max. data which can be in a link is 2BD

So, total frames we can have = $\frac{2BD}{S}$

= total window size

Q. B = 1Mbps, D = 50ms
frame size = 10kb.

$$W = \frac{2 \times 1 \times 10^6 \times 50 \times 10^{-3}}{10 \times 1000}$$

$$= 10$$