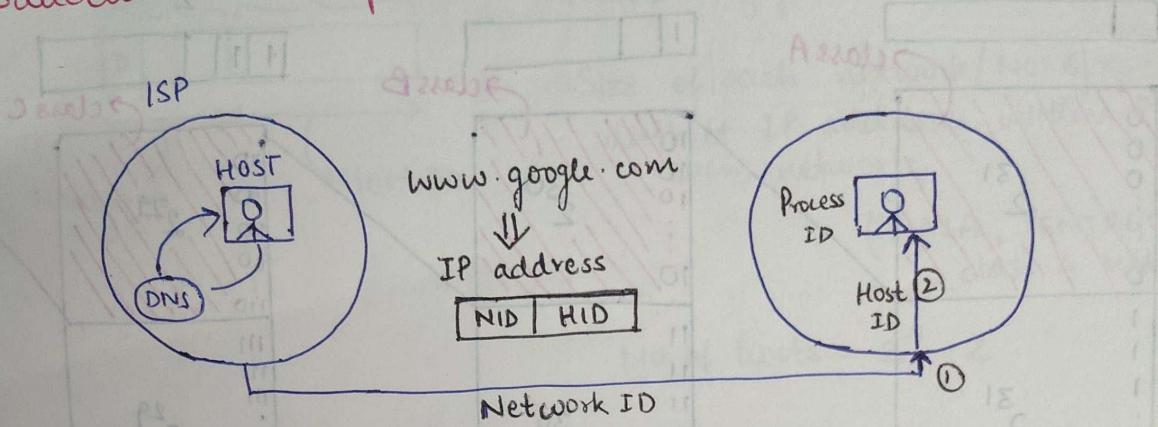


Introduction to Computer Networks and IP Address



① Service that is used to convert domain name into IP address is called domain name service.

② Port number is used to signify a particular process in the host. For well known services, port number is predefined & fixed.

http :- 80

SMTP :- 25

ftp :- 21

③ Even though we need to reach google.com home page, we need to visit DNS which gives the IP Address of google.com home page. This is known as **DNS overhead**.

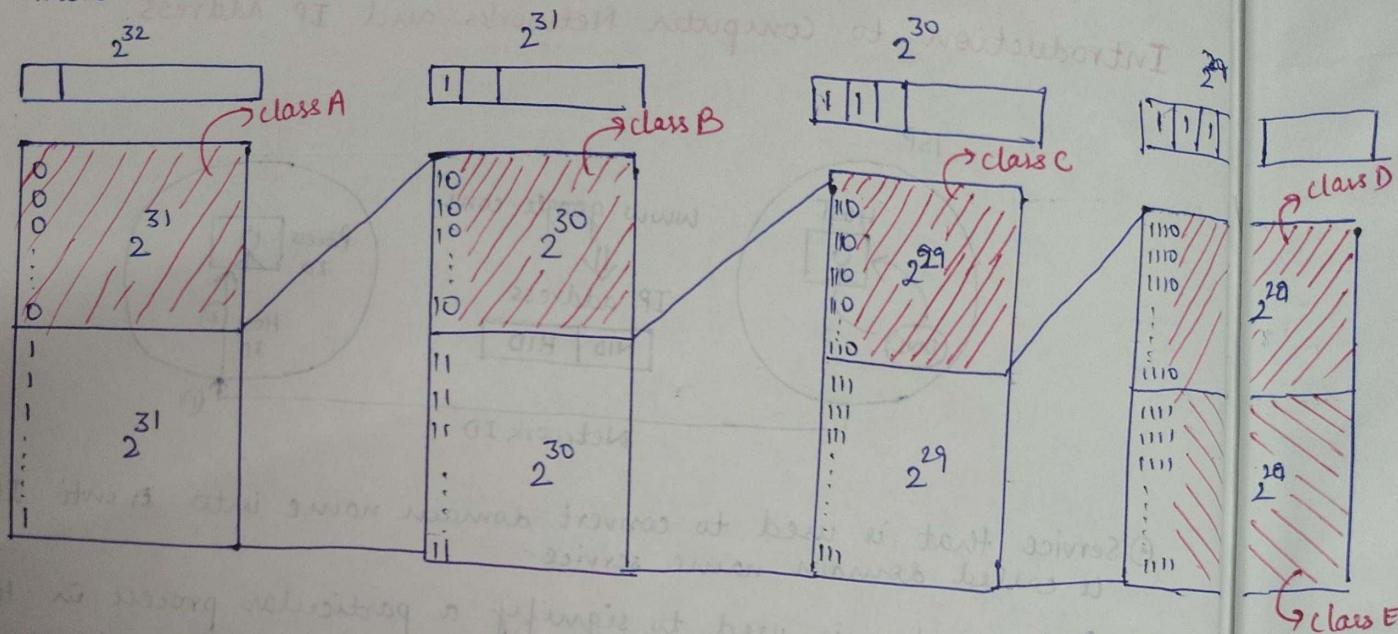
Rectification: The IP address of google.com is stored actually locally in the computer system for some time. If again we want to visit that page, we can directly get IP address.

If IP address in the system expires, we have to go to DNS (no alternative)

01 000 0000 0000 0000 0000 0000 0000
011 0000 0000 0000 0000 0000 0000 0000
0111 0000 0000 0000 0000 0000 0000 0000
1111 0000 0000 0000 0000 0000 0000 0000

Classful IP address classification

There are 32 bits in the IP address.



Number of IP addresses in a network of class A = 2^{31}

$$\text{class B} = 2^{30}$$

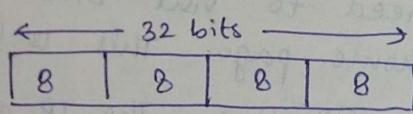
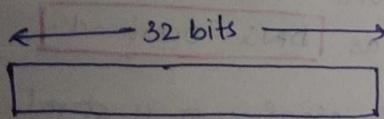
$$\text{class C} = 2^{29}$$

$$\text{class D} = 2^{28}$$

$$\text{class E} = 2^{27}$$

Popular representation of IP address —

→ Binary notation → Dotted Decimal Notation (4 octets)



(192.168.173.9)
8bits 8bits 8bits 8bits

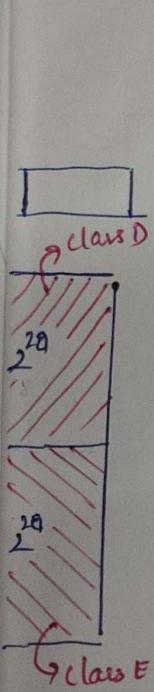
Class A → starts with 0

Class B → starts with 10

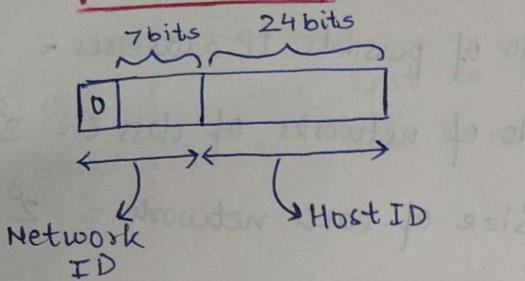
Class C → starts with 110

Class D → starts with 1110

Class E → starts with 1111



CLASS A



Number of possible networks of class A = $2^7 = 128$

Size of each network / No. of distinct IP addresses within each network = $2^{24} = 16M$

(NASA, PENTAGON use class A network)

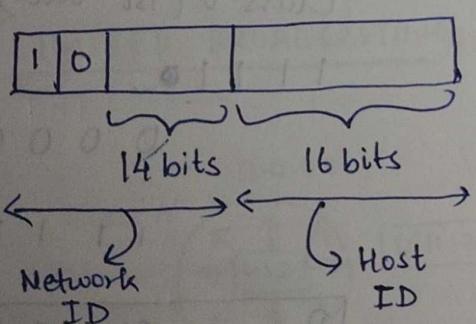
$$\text{No. of hosts} = 2^{24} - 2$$

- Starting address (All zeroes) and ending address (all ones) are not used

$$\text{Number of networks in class A} = 128 - 2 = \underline{\underline{126}}$$

$$\therefore \text{Range} = 0 - 127$$

CLASS B



$$\text{No. of IP addresses possible} = 2^{30}$$

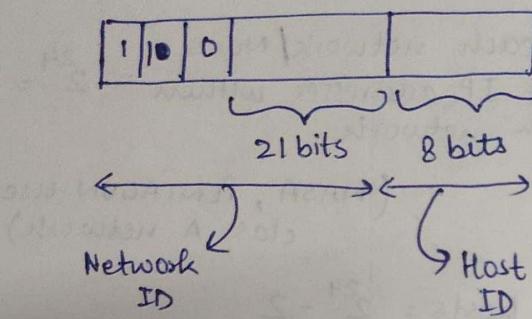
$$\text{No. of networks of class B} = 2^{14}$$

$$\text{Size of each network} = 2^{16}$$

$$\text{No. of hosts} = 2^{16} - 2$$

$$\text{Range} = 128 - 191$$

CLASS C



No. of possible IP addresses = 2^{29}

No. of networks of class C = 2^2

Size of each network = 2^8

No. of hosts in each network = $2^8 - 1$

$$\boxed{\text{Range} = 192 - 223}$$

CLASS D And CLASS E

There is nothing like network IDs and host IDs.

Class D (1st octet)

1110 _____

$$0000 = 224$$

$$1111 = 239$$

$$\boxed{\text{Range} = 224 - 239}$$

Class E (1st octet)

11110 _____

$$0000 = 240$$

$$1111 = 255$$

$$\boxed{\text{Range} = 240 - 255}$$

① used for multicasting

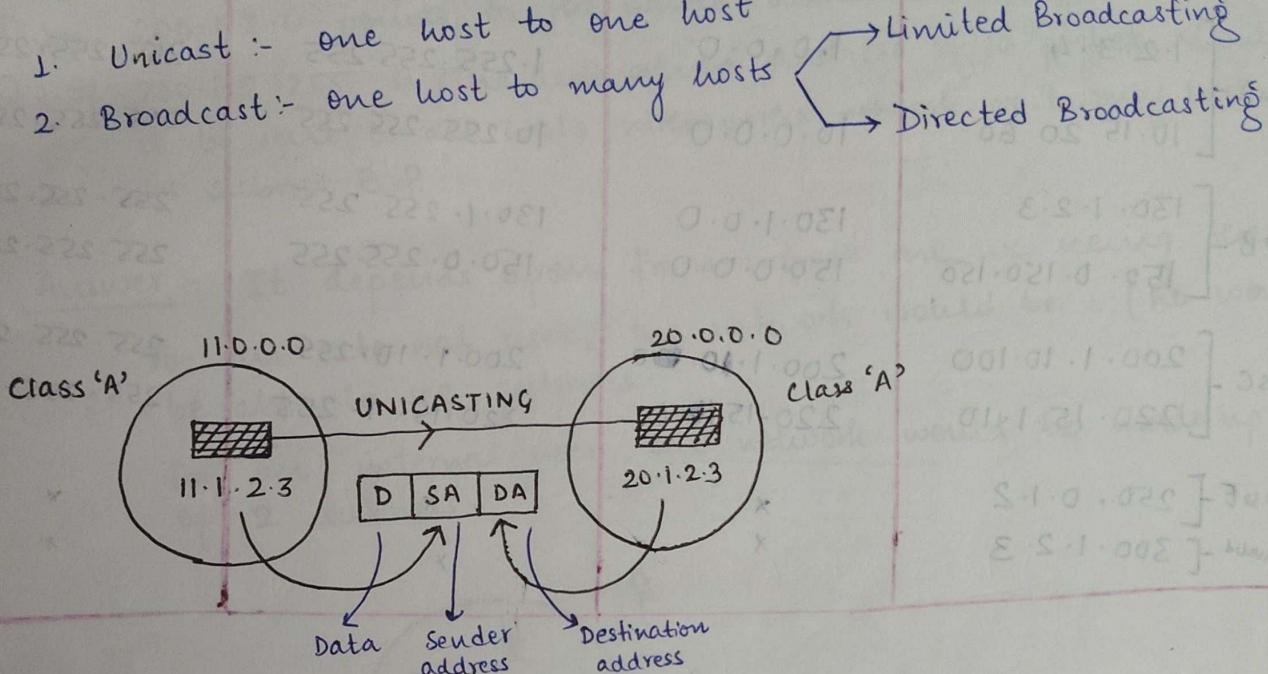
② used for group emailing,
group broadcasting.

① used for military
applications.

Types of casting

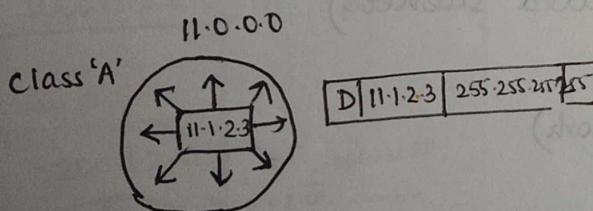
1. Unicast :- one host to one host
2. Broadcast :- one host to many hosts

Unicast
Limited Broadcast
Directed Broadcast



When 'Host ID' has all 0's then the IP address represents the network. This is the reason first IP address in a network is never used as a valid IP address of the host.

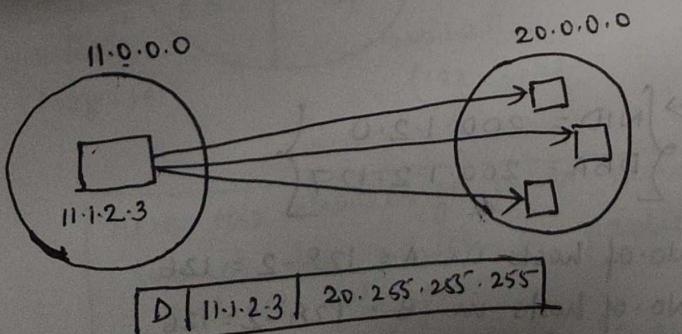
LIMITED BROADCASTING



If Destination address contains all 1's in IP address i.e. 255.255.255.255 then, the packet will be sent to all the hosts within that network.

$$LBA = 255.255.255.255$$

DIRECTED BROADCASTING



If Destination address contains all 1's in the Host ID part, i.e. X.255.255.255, then, the packet will be sent to all hosts in the network.

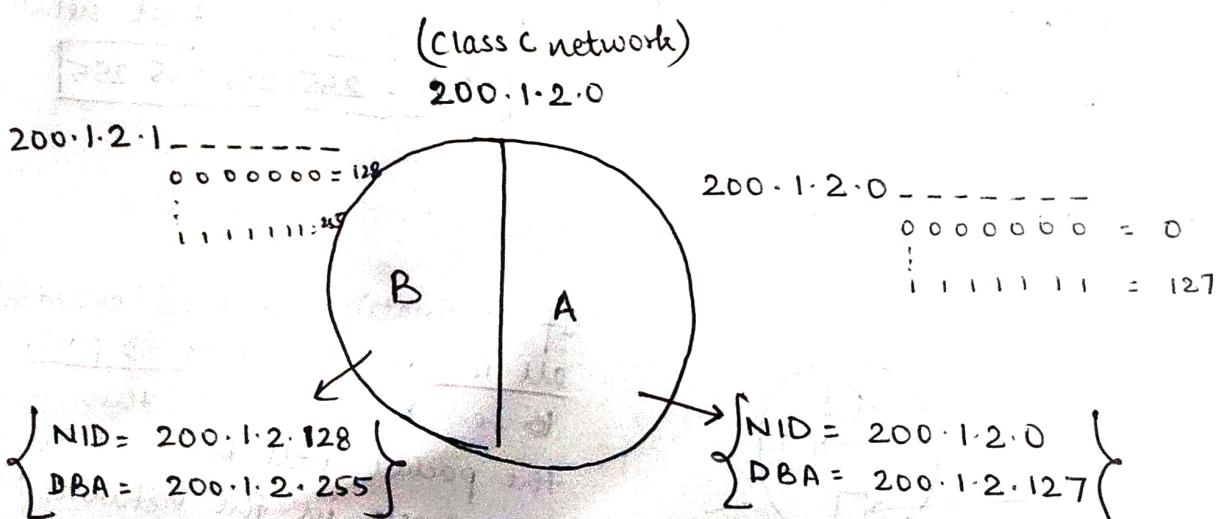
$$DBA: NID, HID = all 1's$$

I/P address	Network ID	Directed Broadcast address	Limited Broadcast Address
Class A [1.2.3.4 10.15.20.60]	1.0.0.0	1.255.255.255	255.255.255.255
	10.0.0.0	10.255.255.255	255.255.255.255
Class B [130.1.2.3 150.0.150.150]	130.1.0.0	130.1.255.255	255.255.255.255
	150.0.0.0	150.0.255.255	255.255.255.255
Class C [200.1.10.100 220.15.1.10]	200.1.10.0	200.1.10.255	255.255.255.255
	220.15.1.0	220.15.1.255	255.255.255.255
Class E [250.0.1.2]	x	x	x
Invalid [300.1.2.3]	x	x	x

SUBNETS , SUBNET MASK, ROUTING

- When the size of network is big, its maintenance will be difficult.
- Also there is a lack of security in a big network.

- Subnetting :-** It is the process of dividing a network into smaller networks (called subnets)



$$\text{No. of hosts in A} = 128 - 2 = 126$$

$$\text{No. of hosts in B} = 128 - 2 = 126$$

$$\text{Total hosts in } 200.1.2.0 = 126 + 126 = 252$$

∴ Subnetting results in decrease in no. of valid hosts.

"There will be loss of IP address due to SUBNETTING"

SUBNET MASK

Ambiguity -

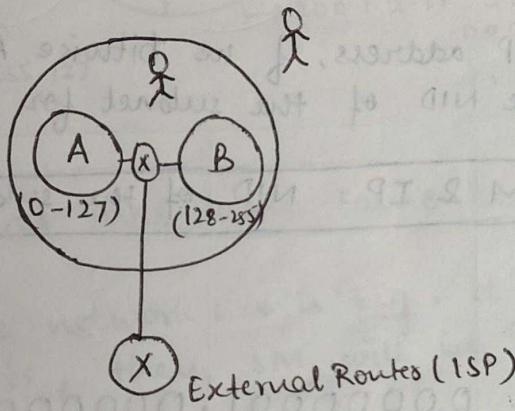
Q.1. 200.1.2.0 points to the whole network or only subnet A?

Q.2. If 200.1.2.255 is the destination address of a packet, should it be sent to all hosts within the network or to only hosts in subnet B?

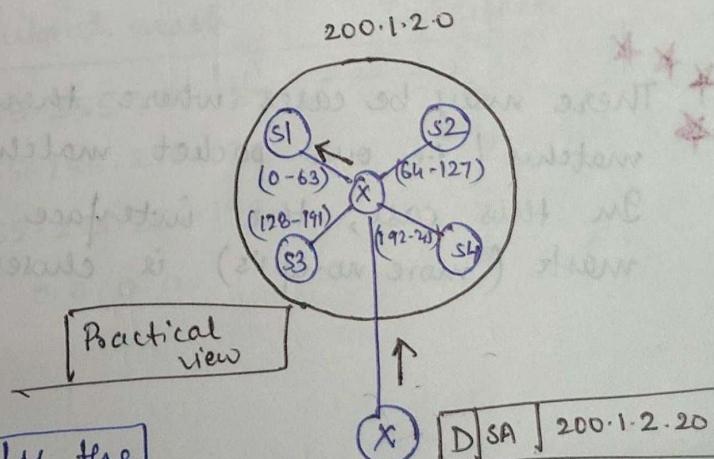
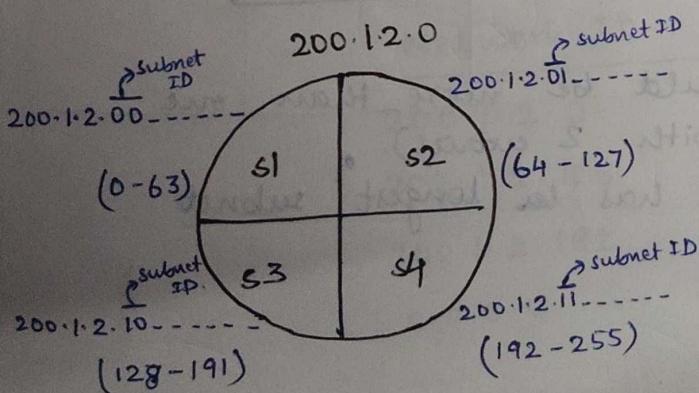
Answer:- It depends upon from where are we seeing the network.

For an external user, the network would be 1 (he won't be able to see subnets).

For an internal user, the network would be made up of 2 subnets



Four subnets -



Here the challenge is to identify the subnet for which a particular IP address belongs.

This is done by using 'SUBNET MASK'

SUBNET MASK

Subnet Mask = 32 bit number

Network ID, subnet ID part — 1's
Host ID part — 0's

Subnet mask for the 4 subnet network shown above is

$$\begin{array}{c} 200 \cdot 1 \cdot 2 \cdot \\ \hline \text{NID} \quad \text{SID} \end{array} - \text{HID}$$

$$\Rightarrow 11111111 \cdot 11111111 \cdot 11111111 \cdot 11000000$$

$$SM = 255 \cdot 255 \cdot 255 \cdot 192$$

Given an IP address, if we bitwise AND with SM, we will get the NID of the subnet for which the IP belongs to.

$$SM \& IP = \text{NID of the subnet}$$

$$IP: 11001000 \cdot 00000001 \cdot 00000010 \cdot 10000010 = 200 \cdot 1 \cdot 2 \cdot 10$$

$$SM: 11111111 \cdot 11111111 \cdot 11111111 \cdot 11000000 = 255 \cdot 255 \cdot 255 \cdot 192$$

$$NID: 11001000 \cdot 00000001 \cdot 00000010 \cdot 10000000 = 200 \cdot 1 \cdot 2 \cdot 10$$

∴ The IP address belongs to the subnet $200 \cdot 1 \cdot 2 \cdot 10$

There may be cases where there could be more than one matches (i.e. our packet matches with 2 entries).

In this case, the interface that has the longest subnet mask (more no. of 1's) is chosen.

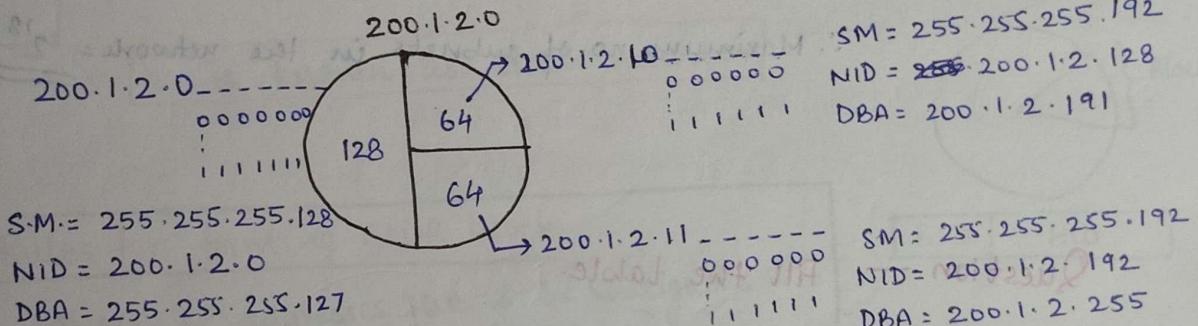
VARIABLE LENGTH SUBNET MASKING

Dividing a network into subnets of unequal sizes is called variable length subnet masking (VLSM)

When subnet size is different, then each subnet will get a different subnet mask.



Three subnets



- The networks that have same sizes have same subnet mask
- When the network size is big, it means there are more hosts, then, SM will be small.
- When the network size is small i.e. less no. of hosts, then, SM will be big.

Routing table

Network ID	Subnet mask	Interface
200.1.2.0	255.255.255.127	a
200.1.2.128	255.255.255.192	b
200.1.2.192	255.255.255.192	c
0.0.0.0	0.0.0.0	d

SUBNET MASKING

Finding number of subnets from the subnet mask 2 class of network

Given subnet mask = 255.255.255.192

$$= 11111111.11111111.11111111.11000000$$

$$= NID + SID = \text{No. of } 1's$$

$$NID + SID = 26$$

If the network belongs to class A, then,

$$NID = 8$$

$$\therefore 8 + SID = 26 \Rightarrow SID = 18$$

$$\therefore \text{Maximum no. of subnets in the network} = \underline{\underline{2^8}}$$

Question — Fill the table.

Subnet mask	No. of hosts	Subnets in class A	Subnets in class B	Subnets in class C
255.0.0.0	$2^{24}-2$	1	—	—
255.128.0.0	$2^{23}-2$	2^1	—	—
255.192.0.0	$2^{22}-2$	2^2	—	—
255.240.0.0	$2^{20}-2$	2^4	—	—
255.255.0.0	$2^{16}-2$	2^8	1	—
255.255.254.0	2^9-2	2^{15}	2^7	—
255.255.255.0	2^8-2	2^{16}	2^8	1
255.255.255.224	2^5-2	2^{19}	2^{11}	2^3
255.255.255.240	2^4-2	2^{20}	2^{12}	2^4

00000000	0
10000000	128
11000000	192
11100000	224
11110000	240
11111000	248
11111100	252
11111110	254
11111111	255

Classless Inter Domain Routing (CIDR)

- ① No class A / class B / class C / class D / class E network is present.
- ② The number of bits in **BID** part is specified with the IP address in CIDR notation.

Instead of NID,
BID is used
(Block ID)

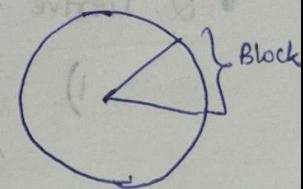
CIDR notation :- $a.b.c.d/n$

32 bit
IP address

No. of bits
in BID

If $n=20$, then,
BID = 20 bits
HID = 12 bits

n is known as slash number



Rules for forming CIDR block

- ① All IP addresses should be contiguous
- ② Block size should always be power of 2
 - ↳ No. of hosts in block
- ③ First IP address in the block should be evenly divisible by the size of block.

Q:- Tell whether the following IP addresses form a CIDR block?

1) 100.1.2.32
100.1.2.33
⋮
100.1.2.47

Rule 1: ✓

Rule 2: No. of IP addresses = $16 = 2^4$ ✓

Rule 3: First IP address —

$100.1.2.00100000$ ✓
↳ 0's : divisible by 2^4

∴ The IP addresses form CIDR Block

2) 20.10.30.32
20.10.30.33
⋮
20.10.30.63

Rule 1: ✓

Rule 2: No. of IP addresses = $32 = 2^5$ ✓

Rule 3: First IP address —

$20.10.30.00100000$ ✓
↳ 0's : divisible by 2^5

∴ The IP addresses form CIDR block

- 3) 150.10.20.64
 150.10.20.65
 :
 150.10.20.127

Rule 1: Rule 2: No. of IP addresses = $64 = 2^6$

Rule 3: First IP address -

150.10.20.0 0000000
 6 0's divisible by 2^6

∴ IP addresses form valid CIDR block

Q. Derive the range of CIDR block -

1) 20.10.30.35 / 27

BID = 27 bits HID = 5 bits

20.10.30.00100011
 BID HID
 00000 (32)
 00001 (33)
 :
 11111 (63)

∴ Range is

20.10.30.32
 20.10.30.33

20.10.30.63

2) 100.1.2.35 / 20

BID = 20 bits HID = 12 bits

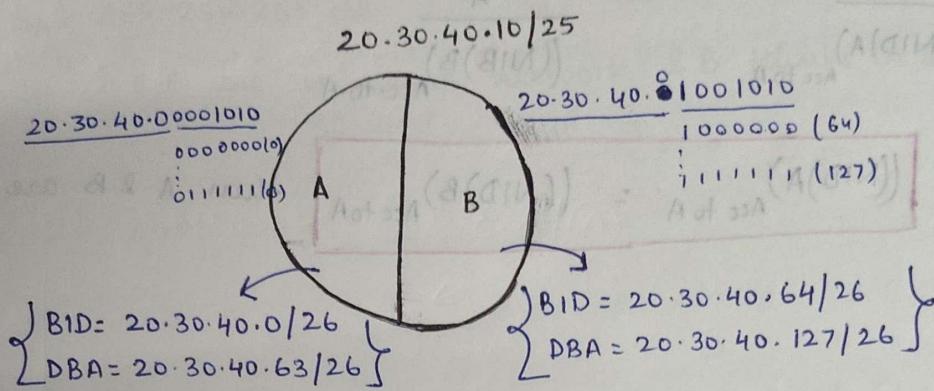
100.1.000000010.00010100
 BID HID
 0000.00000000 (0.0)
 0000.00000001 (0.1)
 :
 1111.11111111 (15.255)

∴ Range is

100.1.0.0
 100.1.0.1
 :
 100.1.15.255

Subnetting in CIDR

73



VLSM in CIDR blocks —

[Same as classful]

IMPORTANT QUESTIONS ON SUBNET MASK

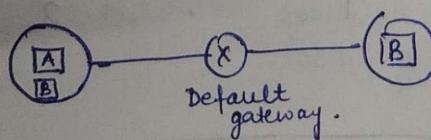
- ① Subnet mask is also known as 'Network mask'
- ② 'ipconfig' command gives the following information —
 - IPv4 address :- provided by ISP
 - Default gateway :- default router connected to the network
 - Subnet Mask :- subnet mask that should be used.
 - DNS :- conversion to IP address.

Let I_A = IP address of A

I_B = IP address of B

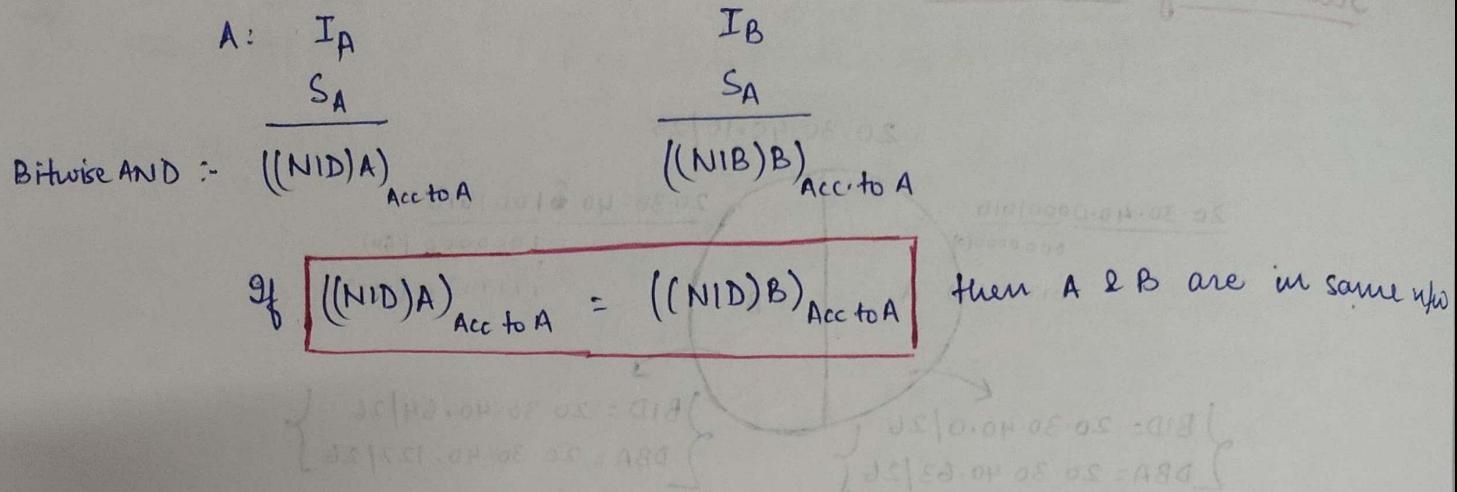
S_A = subnet mask of A

S_B = subnet mask of B.



On sending a packet from A to B, 2 cases are possible —

- ① If A & B are in same network, then, the packet can be transferred directly.
- ② If A & B are in different network, then, packet is sent from A to DGW and then from DGW to B.



Example 1:-

$$I_A = 200.1.2.10 \quad I_B = 200.1.2.130 \text{ in Mask}$$

$$S_A = 255.255.255.128$$

$$I_A : 11001000.00000001.0000010.00001010$$

$$S_A : 11111111.11111111.11111111.10000000$$

$$11001000.00000001.0000010.00000000$$

$$= 200 . 1 . 2 . 0 \text{ "pings"}$$

$$((NID)A)_{Acc\ to\ A} = 200.1.2.0$$

$$I_B = 11001000.00000001.00000010.10000010$$

$$S_A = 11111111.11111111.11111111.10000000$$

$$11001000.00000001.00000010.10000000$$

$$200 . 1 . 2 . 0$$

$$\therefore ((NID)A)_{Acc\ to\ A} \neq ((NID)B)_{Acc\ to\ A}$$

$\therefore A$ assumes A & B are in different network

Example 2: (Practice)

$$I_A = 200 \cdot 1 \cdot 2 \cdot 10$$

$$S_A = 255 \cdot 255 \cdot 255 \cdot 128$$

$$I_B = 200 \cdot 1 \cdot 2 \cdot 69$$

$$S_B = 255 \cdot 255 \cdot 255 \cdot 192$$

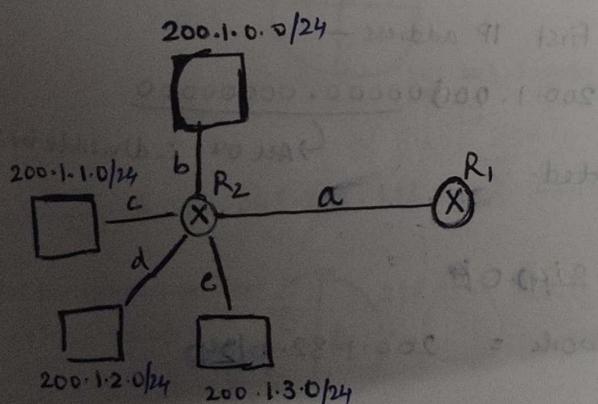
Ans:- [A thinks A & B are in same network & B thinks A is in different network]

Supernetting or Aggregation

- Routing table contains single entry for each and every network.
- If no. of networks is large, then, the size of routing table will grow exponentially and the router takes a lot of time to process it.
- Hence we need to aggregate / combine the networks.
- This process is known as supernetting / aggregation.

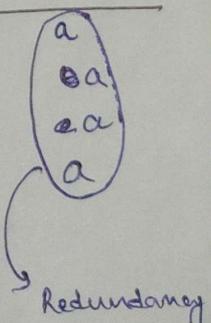
Rules for aggregation

- All the network IDs should be contiguous.
- Size of each network should be same and they should be power of 2.
- The 1st IP address should be evenly divisible by size of the block.



Routing table for R1 —

NID	Subnet mask	Interface
200.1.0.0	255.255.255.0	
200.1.1.0	255.255.255.0	
200.1.2.0	255.255.255.0	
200.1.3.0	255.255.255.0	



Redundancy

Q.1. Aggregate the following network IDs

- 1) 200.1.0.0/24
- 2) 200.1.1.0/24
- 3) 200.1.2.0/24
- 4) 200.1.3.0/24

Rule 1: ✓

Rule 2: Size of each n/w = 2^8 ✓

Rule 3: First IP address —

Total size of network = $4 \times 2^8 = 2^{10}$

200.1.00000000.00000000

All 0's are divisible by 2^{10}

∴ The networks can be aggregated.

SUPERNET MASK

32 bits

No. of 1's = Fixed part

No. of 0's = Variable part / no. of hosts

∴ Supernet mask = 255.255.252.0

N/w ID of the aggregated network is 200.1.0.0/24

Q.2. Aggregate the following network IDs

- 1) 200.1.32.0/24
- 2) 200.1.33.0/24
- ⋮
- 4) 200.1.47.0/24

Rule 1: ✓

Rule 2: Size of each n/w = 2^8

Total size of aggregated network = $2^4 \times 2^8 = 2^{12}$

Rule 3: First IP address —

200.1.00000000.00000000

All 0's are divisible by 2^{12}

∴ The networks can be aggregated.

SUPERNET MASK = 255.255.240.0

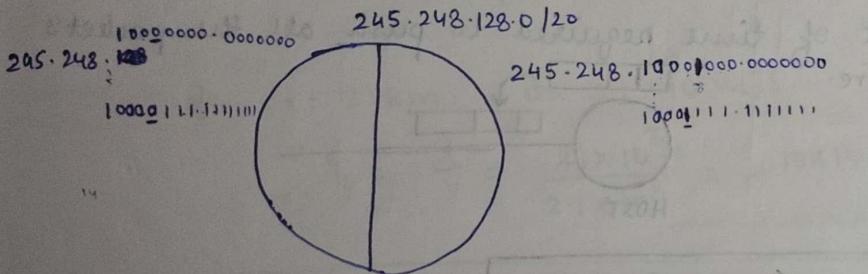
N/w IP for aggregated network = 200.1.32.0/20

Gate 2012

17

ISP has the following chunk of CIDR based IP addresses available with it : 245.248.128.0/20. The ISP wants to give half of the chunk of IP addresses to organization A and a quarter to organization B while retaining the remaining with itself. Which of the following is a valid allocation of addresses to A & B.

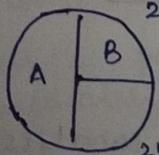
- a) ✓ 245.248.136.0/21 and 245.248.128.0/22
- b) 245.248.128.0/21 and 245.248.128.0/22
- c) 245.248.132.0/22 and 245.248.132.0/21
- d) 245.248.136.0/24 and 245.248.132.0/21



A can be 245.248.128.0/21 or 245.248.136.0/21

If A is 245.248.128.0/21 —

$$245.248.10000000.00000000 = 245.248.136.0/22$$



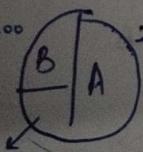
OR

$$245.248.10000100.00000000 = 245.248.240.0/22$$

{ Not present in option-

If A is 245.248.136.0/21 —

$$\begin{aligned} & 245.248.10000000.00000000 \\ & = 245.248.128.0/22 \end{aligned}$$



$$\begin{aligned} & 245.248.10001000.00000000 \\ & = 245.248.136.0 \end{aligned}$$

$$A = 245.248.136.0/21$$

$$B = 245.248.128.0/22$$

FLOW CONTROL METHODS

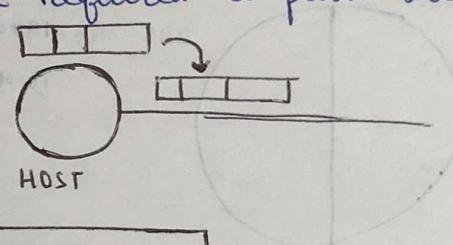
Delays in Computer Network:-

1) Transmission Delay - (T_t)

The time taken to transmit the packet from the host to the outgoing line is called transmission delay.

Also known as store-and-forward delay & packetization delay.

It is the amount of time required to push all the packet's bits into the wire.



$$\text{Transmission delay} = \frac{\text{Data size}}{\text{Bandwidth}}$$

Q:- Bandwidth = 1 bps [1 bit per second]

Data = 10 bits

. Transmission delay = $\frac{10}{1} = \underline{\underline{10s}}$

Q:- Bandwidth = 1 kbps , data size = 1000 bits

Transmission delay = $\frac{1000}{1000} s = \underline{\underline{1s}}$

Q:- Bandwidth = 1 Kbps , data size = 1KB

Transmission delay = $\frac{1024}{1000} = \underline{\underline{1.024s}}$

For data size,

bcoz data size is measured in binary

$$\left. \begin{array}{l} K = 1024 \\ M = 1024 \times 1024 \\ G = 1024 \times 1024 \times 1024 \end{array} \right\}$$

for Bandwidth

$$\left. \begin{array}{l} K = 1000 \\ M = 10^6 \\ G = 10^9 \end{array} \right\}$$

bcoz bandwidth is measured in decimal

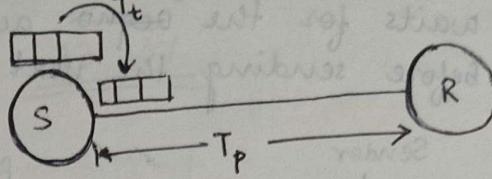
(T_p) 2) Propagation delay -

The time taken by a bit to reach from one end of the link to the other end of the link is called propagation delay.

Propagation delay depends upon -

1) Distance b/w sender & receiver

2) Velocity of the signal in the transmission medium



$$T_p = \frac{d}{v}$$

For optical fibres,

$$v = 70\% \text{ speed of light} = \frac{70}{100} \times 3 \times 10^8 = 2.1 \times 10^8 \text{ m/s}$$

Q:- $d = 21 \text{ km}$ $v = 2.1 \times 10^8 \text{ m/s}$ $T_p = ?$

$$T_p = \frac{d}{v} = \frac{21 \times 10^3}{2.1 \times 10^8} = 10 \times 10^{-5} \text{ s} = 10^{-4} \text{ s}$$

Total time taken to send packet from sender to receiver = $T_t + T_p$

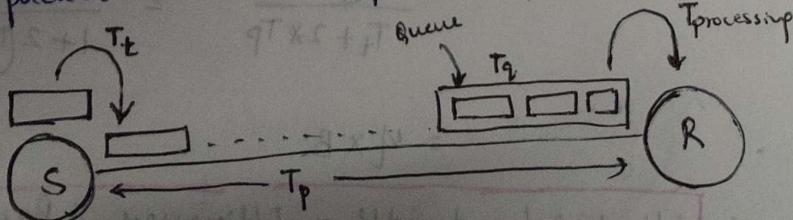
(T_q) 3) Queuing delay -

When a packet reaches the destination, it is not processed immediately. All the packets are stored in the queue before being processed.

The amount of time a packet sits in the queue before it gets processed is called queuing delay (T_q)

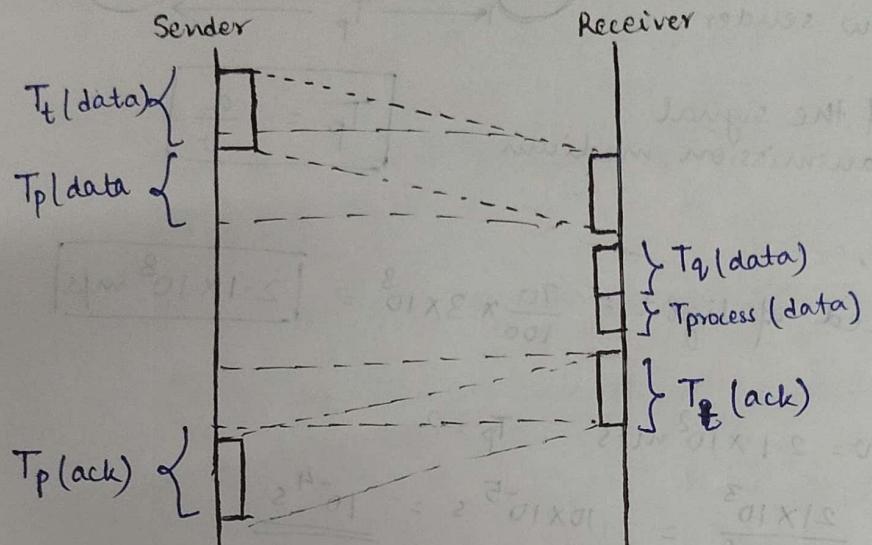
4) Processing delay

The time taken by the receiver to process all the bits of a packet is called processing delay.



Flow control methods stop and wait

Stop and wait is the simplest flow control mechanism in which the sender sends a packet and then stops and waits for the acknowledgement from the receiver before sending the next packet.



Total time -

$$\begin{aligned}
 & T_t(\text{data}) + T_p(\text{data}) + \\
 & I_q(\text{data}) + I_{\text{process}}(\text{data}) \\
 & + T_t(\text{ack}) + T_p(\text{ack}) \\
 = & T_t(\text{data}) + 2T_p + I_{\text{pack}} \\
 = & T_t + 2 * T_p
 \end{aligned}$$

[beacause propagation time for every packet is same]

$$\therefore \boxed{\text{Total time} = T_t + 2 * T_p} \quad \text{for 1 packet}$$

$$\text{Efficiency, } \gamma = \frac{\text{useful time}}{\text{Total time}} = \frac{T_t}{T_t + 2 * T_p} \cdot \frac{1}{1 + 2(T_p/T_t)} = \boxed{\frac{1}{1+2a}}$$

Throughput = No. of bits that can be sent in 1 second using this protocol.

Throughput)

$$\begin{aligned}
 \text{Effective bandwidth /} &= \frac{L}{T_t + 2 * T_p} = \frac{L \times B \times (\gamma_B)}{T_t + 2 * T_p} \\
 \text{bandwidth utilization /} &= \frac{T_p \times B}{T_t + 2 * T_p} = \frac{B}{1 + 2(T_p/T_t)} = \frac{1}{1+2a} \times B \\
 &\text{is utilization} \\
 &= \gamma \times B
 \end{aligned}$$

$$\therefore \boxed{\text{Effective bandwidth} = \text{Efficiency} \times \text{Bandwidth}}$$

Q:- Calculate efficiency -

1) $T_t = 1 \text{ msec}$ $T_p = 1 \text{ msec}$

$$\text{Efficiency, } \eta = \frac{1}{1+2a} \text{ where } a = \frac{T_p}{T_t} = \frac{1}{1} = 1$$
$$= \frac{1}{1+2} = \frac{1}{3} = \underline{\underline{33.33\%}}$$

2) $T_t = 2 \text{ msec}$ $T_p = 1 \text{ msec}$

$$\text{Efficiency} = \frac{1}{1+2a} \text{ where } a = \frac{T_p}{T_t} = \frac{1}{2}$$
$$= \frac{1}{1+2 \times \frac{1}{2}} = \frac{1}{2} = \underline{\underline{50\%}}$$

Q. If $\eta = 50\%$, what is the relation b/w T_p and T_t ?

Given:- $\eta = 50\% = 0.5$

$$\therefore \frac{1}{1+2a} = 0.5 \Rightarrow 1+2a = \frac{1}{0.5} = 2 \Rightarrow 2a = 1 \Rightarrow a = \frac{1}{2}$$

$$\therefore \frac{T_p}{T_t} = \frac{1}{2} \Rightarrow \boxed{T_t = 2T_p}$$

Q. If $BW = 4 \text{ Mbps}$, $T_p = 1 \text{ msec}$ $L = ?$ so that $\eta = \text{atleast } 50\%$.

$$BW = 4 \text{ Mbps} = 4 \times 10^6 \text{ bits/sec.} \quad T_p = 1 \times 10^{-3} \text{ sec.}$$

$$\eta \geq 50\% \Rightarrow \eta \geq 0.5 \Rightarrow \frac{1}{1+2a} \geq 0.5$$

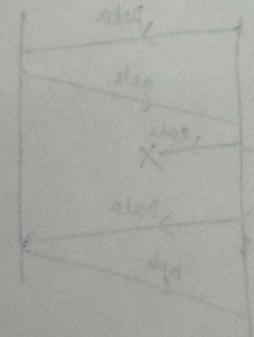
$$\Rightarrow 1+2a \leq 2 \Rightarrow 2a \leq 1 \Rightarrow a \leq 0.5 \Rightarrow \frac{T_p}{T_t} \leq 0.5$$

$$\Rightarrow 2T_p \leq T_t \Rightarrow \frac{L}{BW} \geq 2 \times 10^{-3} \text{ sec}$$

$$\Rightarrow L \geq 2 \times 10^{-3} \times BW$$

$$\Rightarrow L \geq 2 \times 10^{-3} \times 4 \times 10^6$$

$$\Rightarrow \boxed{L \geq 8 \times 10^3 \text{ bits}}$$



Factors which affect efficiency —

$$\eta = \frac{1}{1+2a} = \frac{1}{1+\frac{2T_p}{T_t}} = \frac{\frac{1}{d}}{1+2 \cdot \frac{d}{v} \cdot \frac{BW}{L}}$$

Variables —

d — inversely proportional

v — (fixed)

BW — (fixed)

L — ~~distance~~

If distance increases, η decreases

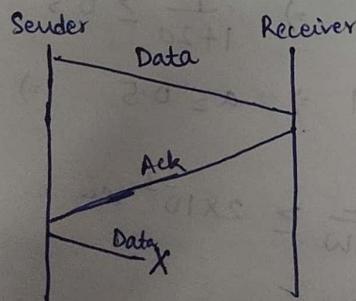
∴ Stop and wait is best for LANs

If data size increases, η ~~decreases~~ increases

∴ Stop and wait is good for big packets

Problems and their rectification in stop and wait protocol —

1) Data lost problem



If data_x is lost —

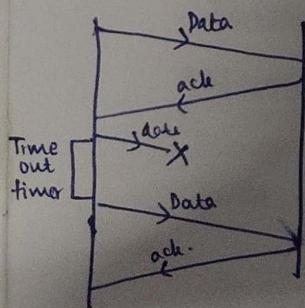
Sender will wait for acknowledgement signal before transmitting next packet.

Receiver will never send acknowledgement signal bcoz it did not receive packet.

∴ DEADLOCK.

Rectification:- Timeout timer is used.

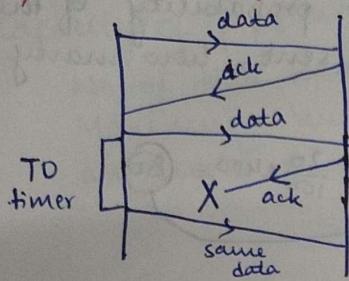
↳ If acknowledgement signal is not received within this time, same packet is sent again.



STOP and WAIT + TIMEOUT TIMER = Stop and wait ARQ

(Automatic Request Repeat Request)

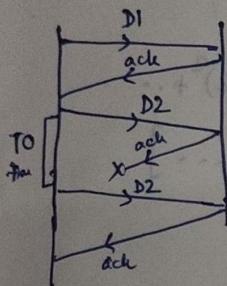
2) Acknowledgement lost problem



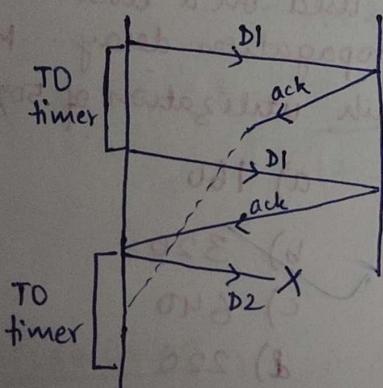
- ① Acc. to sender, maybe data packet was lost due to which it did not get acknowledgement signal.
- ② It sends same data packet again.
- ③ Acc. to receiver, different data packet is received and it will be processed again.

Duplicate packet problem.

Rectification: Have sequence number on each data packet / numbering of packets



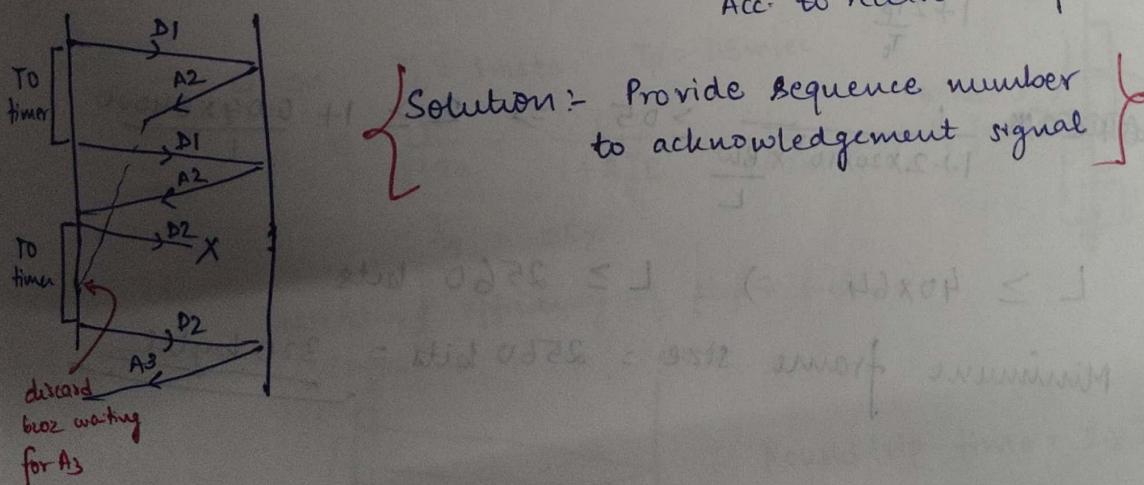
3) Delayed acknowledgement problem.



- ① Acknowledgement signal is delayed.
- ② After timeout, sender again sends the same packet and receives acknowledgement.
- ③ It sends 2nd packet but it is lost and the delayed acknowledgement signal is received.

Acc. to sender - 2 packets are sent.

Acc. to receiver - 1 packet is received.



Solution: Provide sequence number to acknowledgement signal

Q:- There is some problem in the communication channel due to which some of the bits are lost. Let error probability of the channel be $0.2 = 20\%$. If 400 packets are sent, how many packets are transmitted totally?

$$\text{No. of packets which get lost out of 400} = \frac{20}{100} \times 400 = 80$$

Again from these 80,
20% will be lost and so on.

$$\begin{aligned}\therefore \text{No. of packets transmitted} &= 400 + 400(0.2) + 400(0.2)^2 + \dots \\ &= 400 \left[1 + 0.2 + (0.2)^2 + (0.2)^3 + \dots \right] \\ &= 400 \cdot \frac{1}{1-0.2} = \frac{400}{0.8} \\ &= \frac{400 \times 10}{8} = \underline{\underline{500 \text{ packets}}}\end{aligned}$$

Gate 2015

Suppose that stop and wait protocol is used on a link with a bit rate of 64 kb/sec and 20 msec propagation delay. Minimum frame size in bytes to achieve a link utilization of 50%.

Given: $T_p = 20 \times 10^{-3} \text{ sec}$

Bandwidth, $BW = 64000 \text{ bits/sec}$

Link utilization = $\frac{BW \cdot T_p}{BW \cdot T_p + 2T_p} \geq 50\%$.

a) 160

b) 320

c) 640

d) 220

$$\Rightarrow \frac{1}{1+2\frac{T_p}{T_e}} \geq 0.5$$

$$\Rightarrow \frac{1}{1+2 \times 20 \times 10^{-3} \times \frac{BW}{L}} \geq 0.5 \Rightarrow 2 \geq 1 + \frac{0.04 \times 64000}{L}$$

$$\Rightarrow L \geq 40 \times 64 \Rightarrow L \geq 2560 \text{ bits}$$

\therefore Minimum frame size = 2560 bits = 320 bytes

CAPACITY OF PIPE AND PIPELINING

125

Capacity of pipe -

No. of bits that can

Maximum number of bits that can be present in a pipe at an instant is called capacity of pipe.

Capacity of pipe depends upon 'bandwidth' & 'propagation delay'.

$$\boxed{\text{capacity of pipe} = \text{BW} * T_p} \quad \text{for half duplex}$$

$$\boxed{\text{capacity of pipe} = 2 * \text{BW} * T_p} \quad \text{for full duplex}$$

If capacity is high, it is called thick pipe.

If capacity is low, it is called thin pipe.

For stop and wait,

$$\eta = \frac{1}{1+2a} = \frac{1}{1+2 * \frac{T_p}{T_t}} = \frac{1}{1+2 * \frac{T_p * BW}{L}}$$

$$(COST) = \frac{1}{1 + \frac{C}{L}} \quad \therefore \eta \propto \frac{1}{C}$$

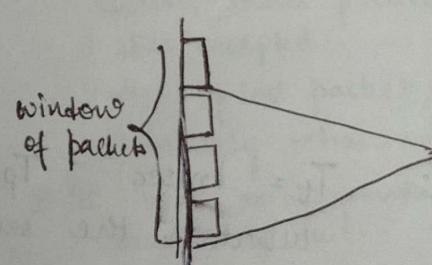
If capacity is more,
efficiency of stop & wait
is less.

Pipelining

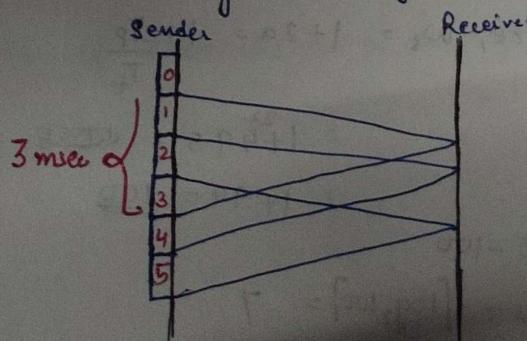
Time taken to transmit 1 packet in stop & wait protocol = $T_t + 2 * T_p$

Given:- $T_t = 1\text{ msec}$ $T_p = 1.5\text{ msec}$

$$\eta = \frac{1}{1+2a} = \frac{1}{1+2 * \frac{T_p}{T_t}} = \frac{1}{1+2 * 1.5} \\ = \frac{1}{4} = 25\%$$



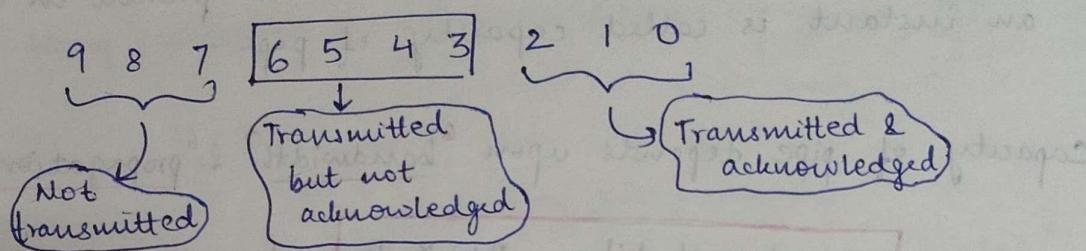
Increasing efficiency of stop & wait -



Round trip time = $2 * T_p = 3\text{ msec}$.

25

Buffer / Queue is maintained at the sender side so that if a packet is lost, it can be retransmitted. When acknowledgement signal for a packet is received, it is removed from the buffer.



Also known as sliding window protocol

The sender window size in sliding window $\stackrel{W_s}{=} 1+2a$

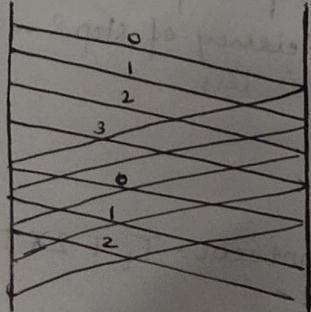
- ① The sequence nos of the packets have to be stored in the header field of the packet.

$$\text{Min. no. of sequence nos} = 1+2a$$

$$\text{Min. no. of bits in sequence no. fields} = \text{No. of sequence nos} = 1+2a$$

$$2^n = 1+2a \Rightarrow n \log 2 = \log(1+2a)$$

$$\Rightarrow n = \lceil \log_2(1+2a) \rceil$$



[Packet no. of packet transmitted & acknowledged can be used again.]

$$Q:- T_t = 1\text{ msec} \quad T_p = 49.5\text{ msec}$$

What is the sender window size for maximum efficiency?

For maximum efficiency,

$$\text{Sender window size, } W_s = 1+2a = 1+\frac{2T_p}{T_t}$$

$$= 1 + \frac{2 \times 49.5}{1} = \cancel{495} = \underline{\underline{99}}$$

$$= 1 + 99 = \underline{\underline{100}}$$

Sequence nos $\rightarrow 0-100$

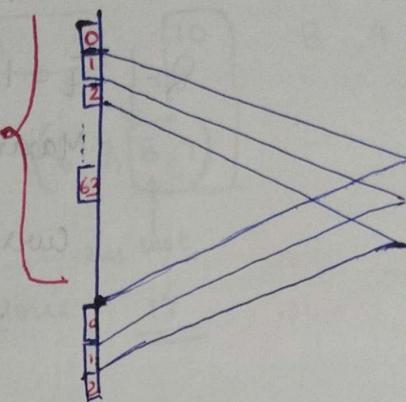
$$\text{Minimum no. of bits} = \lceil \log_2 100 \rceil = 7$$

27
 \Rightarrow If in the above question, the number of sequence field bits is given to be 6 then, 64 sequence numbers are possible ranging from (0-63)

$$\therefore y = \frac{64}{100} = 0.64$$

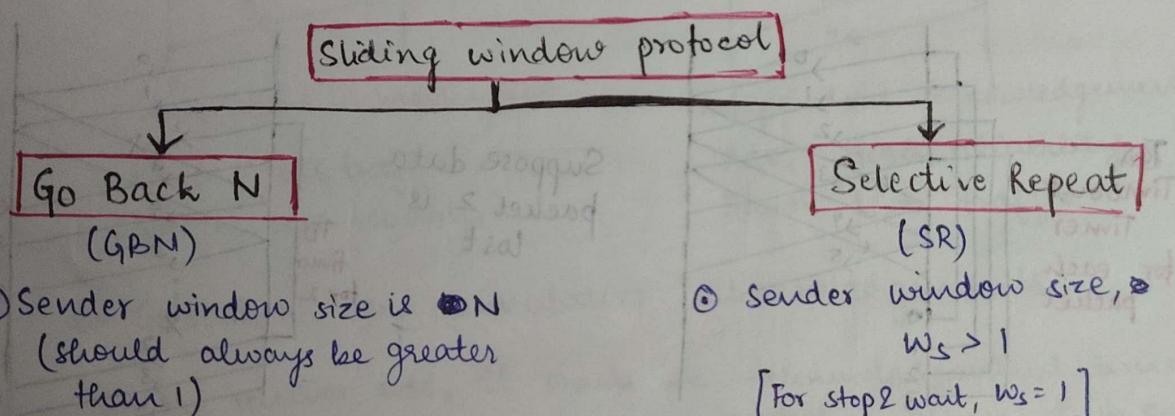
\therefore Window size in stop & wait is limited by the number of bits in the sequence number field.

$$W_s = \text{Math.min}(1+2a, 2^n)$$



① Sliding window protocol is a theoretical concept.

It is practically implemented by —



① Sender window size is W_s
(should always be greater than 1)

② Receiver window size is always 1

[Receiver will never accept out of order packets.
Whole window is re-transmitted in case of a packet loss]

③ It uses cumulative acknowledgement.

① Sender window size, $W_s > 1$
[For stop & wait, $W_s = 1$]

② Receiver window size,

$W_R = W_s$
[Even if one of the packets is lost, other packets are still accepted.]

Only 1 lost packet is selectively retransmitted]

③ It uses independent acknowledgement.

No. of retransmissions is high

No. of retransmissions is low

GO BACK N

① Sete Sender size in Go Back N is N

$$Q: T_t = 1\text{ msec} \quad T_p = 49.5\text{ msec}$$

GBN GO BACK N $a = ?$
BW = 40 Mbps

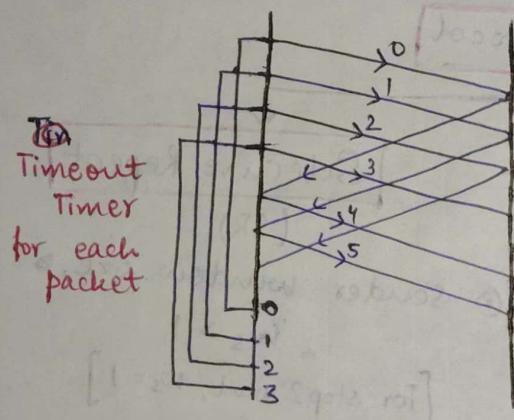
$$\text{Maximum window size} = 1 + 2a = 1 + 2 \frac{T_p}{T_t} = 1 + 99 = 100$$

Current window size = 10

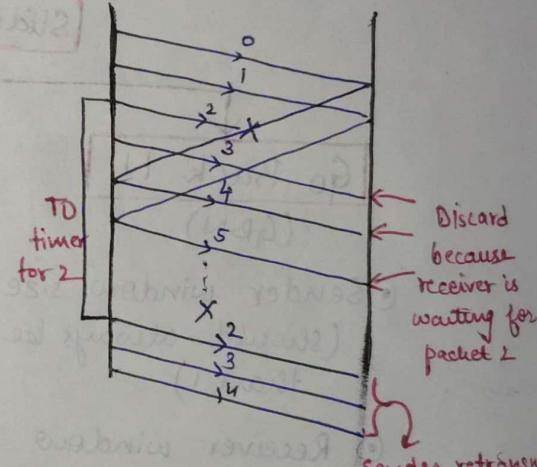
$$\therefore \eta = \frac{10}{100} = \underline{\underline{10\%}}$$

$$\text{Throughput} = \eta \times \text{BW} = \frac{10}{100} \times 40 \text{ Mbps} = \underline{\underline{4 \text{ Mbps}}}$$

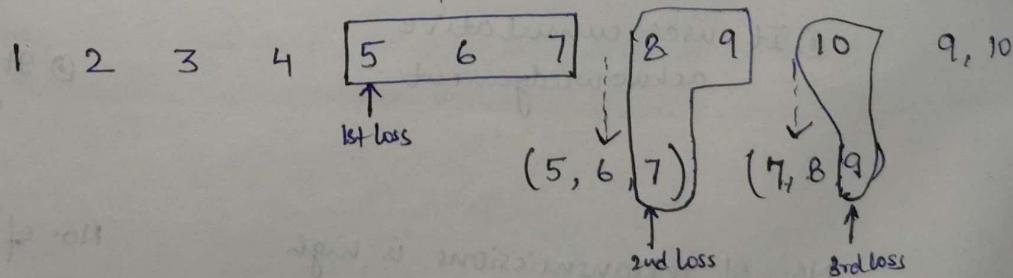
② Receiver window size is 1



Suppose data packet 2 is lost.

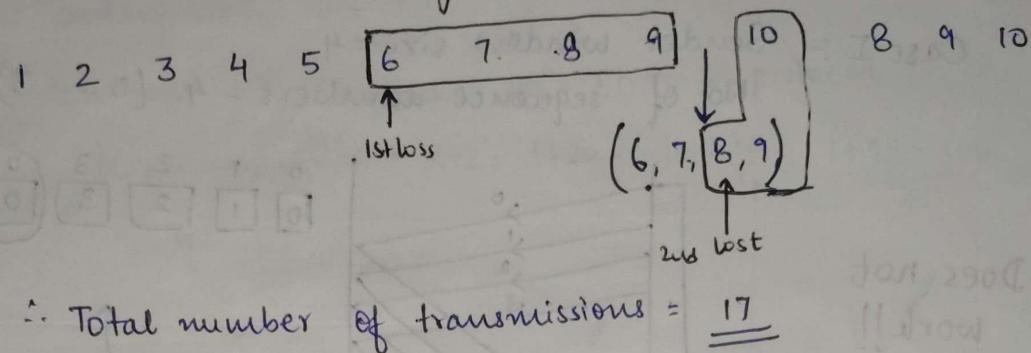


Q:- In Goback 3, if every 5th packet that is being transmitted is lost and 10 packets are to be sent, how many transmissions are required?

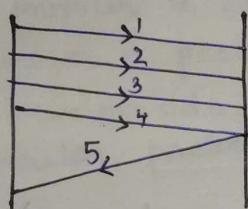


\therefore Total no. of transmissions = 18

Q:- In Go Back-N, every 6th packet is lost. 10 packets are to be sent. How many transmissions?



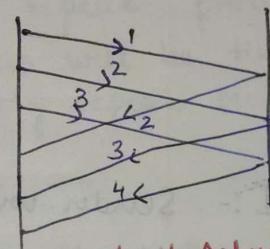
① Acknowledgements :- There are 2 kinds of acknowledgements



Cumulative Acknowledgment

Adv:- Less traffic

Disadv:- Less reliable



Independent Acknowledgment

Adv: High reliability

Disadv: More traffic

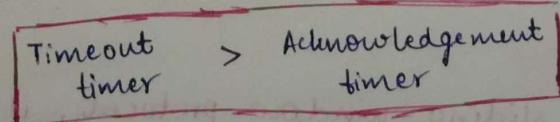
② Go Back-N uses cumulative acknowledgement.

At the receiver side, it starts an acknowledgement timer whenever receiver receives any packet. When it expires, it sends cumulative acknowledgement for the number of packets received in that interval.

If 'N' packets are received, $N+1$ acknowledgement no. will be sent.

!! Imp:- Acknowledgement timer will not start until the receiver has received a packet

!! Timeout timer at sender side should be greater than acknowledgement timer at receiver side.

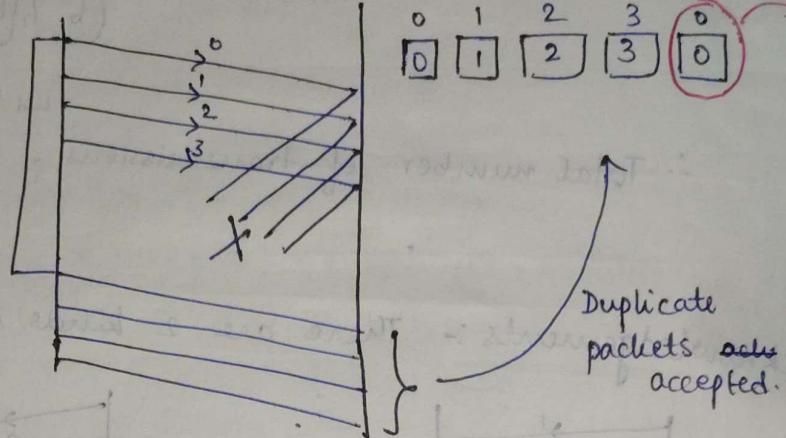


Relationship b/w window sizes and sequence numbers

Case I : Sender window size = 4

No. of sequence numbers = 4 (0...3)

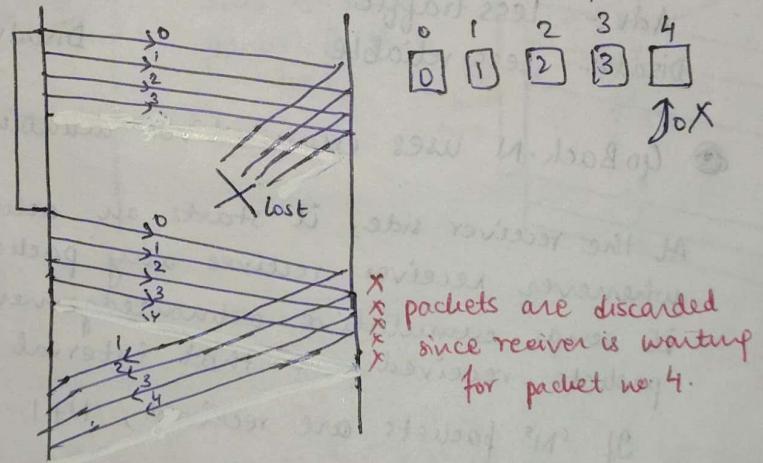
Does not work!!



Case II :- Sender window size = 4

No. of sequence numbers = 5 (0...4)

Works!



⇒ If sender window size = 'N' & receiver window size = '1', then to detect duplicate packets, ' $N+1$ ' sequence nos should be used.

For any sliding window protocol, if it has to work without any problem, the condition is

$$W_s + W_r \leq ASN$$

(Available Sequence Numbers)

SELECTIVE REPEAT

- ① Sender window size, $W_s > 1$

Q:- $T_t = 1\text{ms}$ $T_p = 49.5\text{ms}$ $W_s = 50$ SR protocol $\eta = ?$

Maximum ^{sender} window size = $1 + 2a = 1 + \frac{2T_p}{T_t} = 1 + 99 = 100$

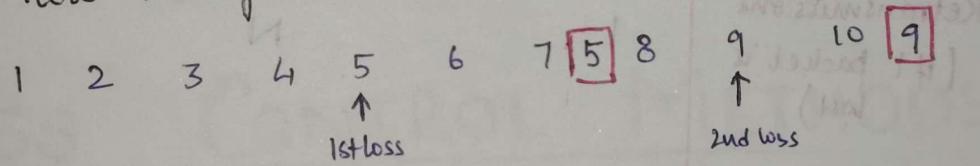
$$\therefore \eta = \frac{50}{100} = 50\%$$

- ② Receiver window size is equal to sender window size

$$W_R = W_s$$

This protocol is called 'selectively repeat' protocol because whenever a packet is lost, there will be timeout at the sender side & sender selectively repeats only that particular packet.

- Q:- $W_s = 3$ 10 packets are sent. Every 5th packet is lost. How many transmissions in SR protocol?



Total no. of transmissions = 12

- ③ The Acknowledgements are independent.

- !! In case of packet loss, GBN & SR behave in the same way.
- !! In case of corrupted data packet, GBN doesn't send acknowledgement signal while SR sends NACK (Negative acknowledgement)

The maximum window size should be half here

max seq number

Comparison of various flow control protocols

	Stop and wait (SAW)	Go Back N (GBN)	Selective Repeat (SR)
Efficiency	$\frac{1}{1+2a}$	$\frac{N}{1+2a}$	$\frac{N}{1+2a}$ → sender window size
Buffers (sender + receiver buffers)	1 + 1	$N + 1$	$N + N$
Sequence Numbers	2	$N + 1$	$2N$
Retransmissions (if 1 packet is lost)	1	N	1
Bandwidth Required	Low	High (because of high transmissions)	Moderate
CPU requirement [complexity]	Low	Moderate (No sorting at receiver No searching at sender)	High (out of order packets are also accepted) Sorting is then done at receiver searching of lost packet at sender.
Implementation	Low	Moderate	Complex
Acknowledgments	Independent	Cumulative	Independent

Gate 2016

133

Consider 128×10^3 b/s satellite communication one way propagation delay of 150 ms. Selective Retransmission protocol is used on the link to send data with frame size of 1 kB. Neglect transmission time of acknowledgement. The minimum number of bits required for sequence number fields to achieve 100% utilization is

a) 2

b) 4

c) 6

d) 8

$$\eta = \frac{W}{1+2\alpha} \quad \text{where } W = \text{sender window size}$$

$$W \leq 1+2\alpha$$

$$\Rightarrow W \leq 1 + 2 \frac{T_p}{T_t} \quad T_t = \frac{L}{BW} = \frac{1024 \times 8}{128 \times 10^3}$$

$$= \frac{2^{13-7}}{10^3} = \frac{2^6}{10^3} = 0.064$$

For SR protocol,

$$W_s = W_R$$

$$\leq 5.68 \Rightarrow 5$$

$$\therefore \text{No. of sequence nos required} = W_s + W_R \\ = 5 + 5 = 10$$

$$\text{No. of bits required} = \lceil \log_2 10 \rceil = 4$$

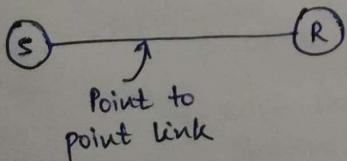
ACCESS CONTROL METHODS

Stations can communicate with one another using 2 types of links —

Types of links

Point to point links

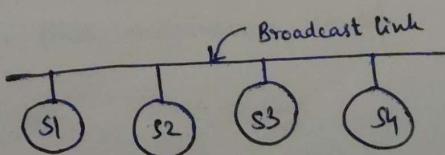
- ① dedicated link b/w 2 stations
- ② entire capacity of the link is used for communication b/w 2 stations only.



Point to point link

Broadcast links

- ① It is the common link to which multiple stations are connected.
- ② The capacity of the link is shared among the connected stations for communication.



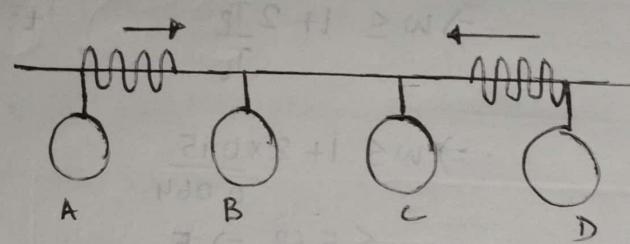
Access Control

Access Control is a mechanism that controls the access of stations to the transmission link.

Need:-

To prevent the ~~occurrence~~ occurrence of collision or if the collision occurs, deal with it.

Collision of data packets results in data corruption.



A & D start transmitting simultaneously.

- ④ Access control methods are the methods used for providing the access ~~code~~ control.
 - ④ ensure smooth flow of traffic on the network.
 - ④ implemented at the Data ~~bit~~ link layer of the OSI model.

METHODS

~~Access control methods~~

→ Time Division Multiplexing

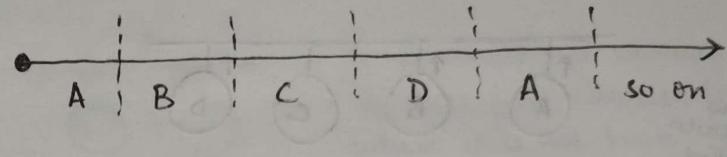
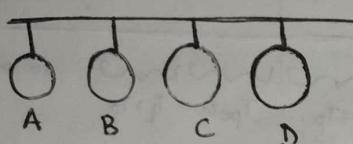
- Polling
 - CSMA/CD
 - Token Passing
 - Aloha

1) Time Division Multiplexing

of equal size

ON 11/09/18

- ① The link time is divided into slots and each slot is assigned/ allocated to the stations in Round Robin manner.
- ② Each station transmits its data during the time slot allocated to it.
- ③ If station does not have any data to transmit, the time slot goes waste.



Size of time slot

Size of each time slot is kept such that each station gets sufficient time to

↳ transmit the packet on the transmission line (T_t)

↳ last bit of the packet is able to get out of the link (T_p)

$$\therefore \text{Size of time slot} = T_p + T_t$$

$$\boxed{\text{Efficiency, } \eta = \frac{\text{useful time}}{\text{cycle time}} = \frac{T_t}{T_t + T_p} = \frac{1}{1+a}}$$

$$\boxed{\text{Effective Bandwidth/Bandwidth utilization/Throughput} = \eta \times \text{BW}}$$

Disadvantage

If any station does not have data to transmit, then, the time slot is wasted. This leads to decrease in efficiency.

Q:- If $T_p = 1\text{ms}$, $T_t = 1\text{ms}$, $\text{BW} = 4\text{Mbps}$ • find effective bandwidth, efficiency

$$\eta = \frac{1}{1+a} = \frac{1}{1+T_p/T_t} = \frac{1}{1+1} = 50\%.$$

$$\text{Effective Bandwidth} = \eta \times (\text{BW}) = 0.5 \times 4\text{Mbps} = 2\text{Mbps}$$

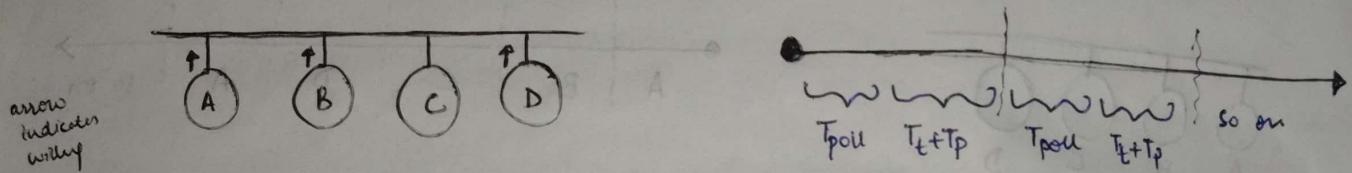
If each station requires 2Mbps BW, find maximum no. of stations

$$N \times 2\text{Mbps} = \text{Max available BW}$$

$$\Rightarrow N \times 2\text{Mbps} = 2\text{Mbps} \Rightarrow N = 1000$$

2) Polling

- ① One station is selected among a group of stations willing to transmit data by using a polling algorithm.
- ② The selected station then sends the data to the destination.
- ③ Then, the cycle repeats.



$$\boxed{\text{Efficiency} = \frac{\text{useful time}}{\text{cycle time}} = \frac{T_t}{T_{poll} + T_t + T_p}}$$

Advantages

- ① no time slot is wasted (unlike TDM)
- ② maximum efficiency & bandwidth utilization

Disadvantages

- ① Time is wasted during polling
- ② Each station has equal probability of being selected in each round. \therefore some stations may starve.

$$\boxed{\text{Maximum Available Effective Bandwidth} = \frac{\text{Total no. of stations}}{\text{Bandwidth Requirement of 1 station}}}$$

Throughput

$$P_{\text{eff}} = \frac{1}{T+1} = \frac{1}{N+1} = \frac{1}{n+1}$$

$$\text{Throughput} = P_{\text{eff}} \times \text{Bandwidth} = \frac{1}{n+1} \times \text{Bandwidth}$$

For example, if there are 10 stations, then the throughput will be $\frac{1}{11} \times \text{Bandwidth}$

$$\text{Throughput} = \frac{1}{n+1} \times \text{Bandwidth}$$

$$\text{Throughput} = \frac{1}{11} \times \text{Bandwidth}$$

3) CSMA/CD

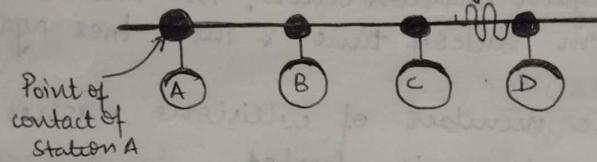
used in Ethernet

LANS not WANS

Carrier Sense Multiple Access / Collision Detection

Step 1:- Sensing the carrier

- Any station willing to transmit the data senses the carrier (checks if carrier is free or not).
- If it finds the carrier free, it starts transmitting the packet otherwise not.



If A senses the carrier, it will find carrier-free while it is not bcoz D is transmitting data.

Step 2:- Detecting the collision

- Detection of collision is done by the transmitting station.
- For detecting collision, CSMA/CD implements the following condition -

$$T_t \geq 2 * T_p$$

$$L \geq 2 * T_p * BW$$

2 cases are possible

① No collided signal comes back during transmission

② Collided signal comes back during transmission

(indicates data packet is transmitted successfully)

Minimum size of the packet.
If size of packet is less than the minimum size, extra bits are added to it.

- Collision Detection is not possible if $L < 2 * T_p * BW$

Step 3:- Releasing JAM signal

- JAM is a 48 bit signal

released by transmitting station as soon as it detects collision

- It alerts other stations not to transmit their data immediately after collision (otherwise there is a possibility of collision with same data packet again)

- Frequency of JAM signal is different from that of data signals. This ensures that JAM signal does not collide with data the data signals undergone collision.

Step 4: Waiting for Back Off Time

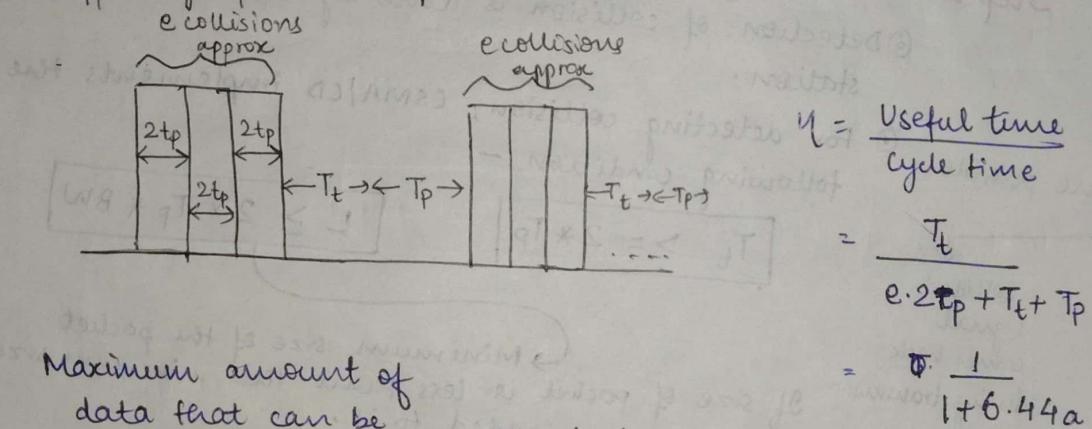
After the collision occurs, the transmitting station waits for a random amount of time called back-off time (determined by backoff algorithm).

After backoff time, it tries transmitting the data packet again.

If again collision occurs, the station again waits for some random backoff time & then tries again.

If the number of collisions reaches its limit, the transmission is aborted.

Efficiency of CSMA/CD / Ethernet



Maximum amount of data that can be sent through ethernet = 1500 bytes

Back off Algorithm

n = No. of collisions of a data packet.

Transmitting stations choose a no. from $(0$ to $2^n - 1)$

$$\boxed{\text{Waiting time} = T_{\text{slot}} \times K}$$

→ Applicable for only 2 stations

→ If A won after 1st collision its probability for 2nd collision is higher.

→ Collision probability decreases exponentially

If distance decreases, η increases

∴ suitable for LANs
not suitable for WANs

If size of data packet increases, η increases

∴ suitable for bigger data packets.

Gate 2015

Consider a CSMA/CD network that transmits data at a rate of 100 Mbps (10^8 b/s) over 1km cable with no repeaters. If the minimum frame size required for this network is 1250 bytes, what is the signal speed in the cable in km/sec?

- A) 8000
- B) 10000
- C) 16000
- D) ~~20000~~

$$BW = 100 \text{ Mbps} = 10^8 \text{ bps}$$

Length of cable = 1km

Minimum size of data packet = $1250 \times 8 \text{ bits}$

$$T_t = \frac{1250 \times 8}{10^8} \text{ s} = \frac{10000}{10^8} = 10^{-4} \text{ s}$$

For CSMA/CD network,

$$T_t = 2T_p \quad [\because \text{packet size is minimum}]$$

$$T_p = \frac{d}{v} = \frac{L}{v}$$

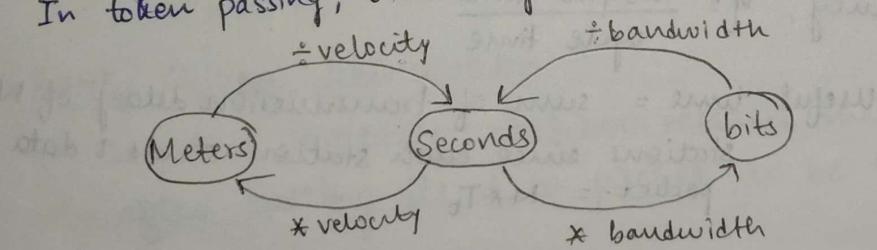
$$\Rightarrow 10^{-4} = 2 \times \frac{L}{v} \Rightarrow v = 2 \times 10^4 = \underline{\underline{20000 \text{ km/s}}}$$

$$[THT \times N + qT = THT \times N + \frac{L}{v} = \text{exit step}]$$

4) Token passing

Ring topology

In token passing, time may be expressed as meters/second/bits.



Terminology

1. Token:

- ① small message composed of special bit pattern.
- ② represents the permission to send a data packet.

③ A station is allowed to send data packet if & only if it possesses the token otherwise not.

2. Ring latency:

Time taken by a bit to complete one revolution of the ring is called ring latency.

$$\text{Ring latency} = \frac{d}{v} + N * b$$

Length of the ring = d

Speed of the bit = v

No. of stations = N

Bit delay at each station = b

Time taken by the bit to traverse the ring (T_p)

Time taken by the stations to hold the bit.

3) Token Holding time

Time for which a station holds the token before transmitting to the other side.

4) Cycle time

The time taken by the token to complete one revolution of the ring is called token time cycle time.

$$\text{Cycle time} = \frac{d}{v} + N * \text{THT} = T_p + N * \text{THT}$$

Efficiency -

$$\text{Efficiency, } \eta = \frac{\text{useful time}}{\text{cycle time}}$$

Useful time = sum of transmission delay of N stations since each station sends 1 data packet. = $N * T_t$

$$\therefore \eta = \frac{N * T_t}{T_p + N * \text{THT}}$$

!! *Token Holding Time' depends upon the strategy implemented

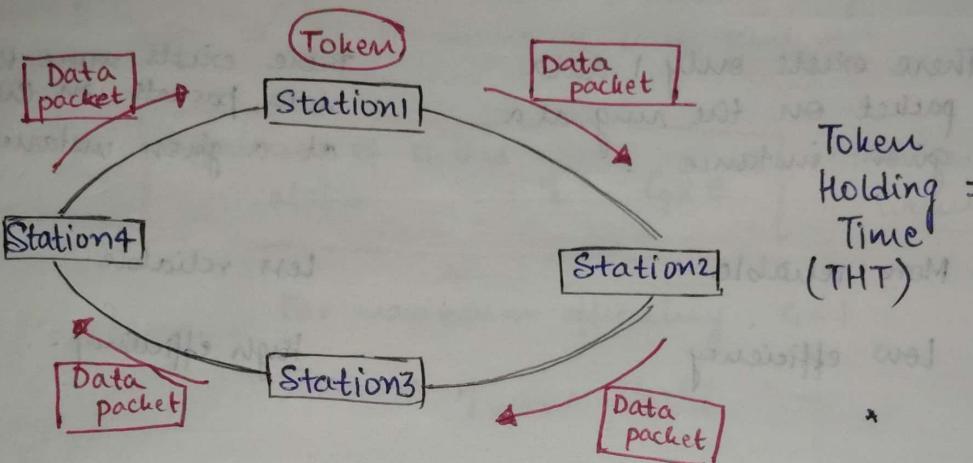
Token Passing strategies

Delayed Token Reinsertion

Early Token Reinsertion

Delayed Token Reinsertion

In this strategy, the station keeps holding the token until the last bit of data packet transmitted by it takes the complete revolution of the ring & comes back to it.



$$\text{Token Holding Time (THT)} = \text{Transmission delay} + \text{Ring Latency}$$

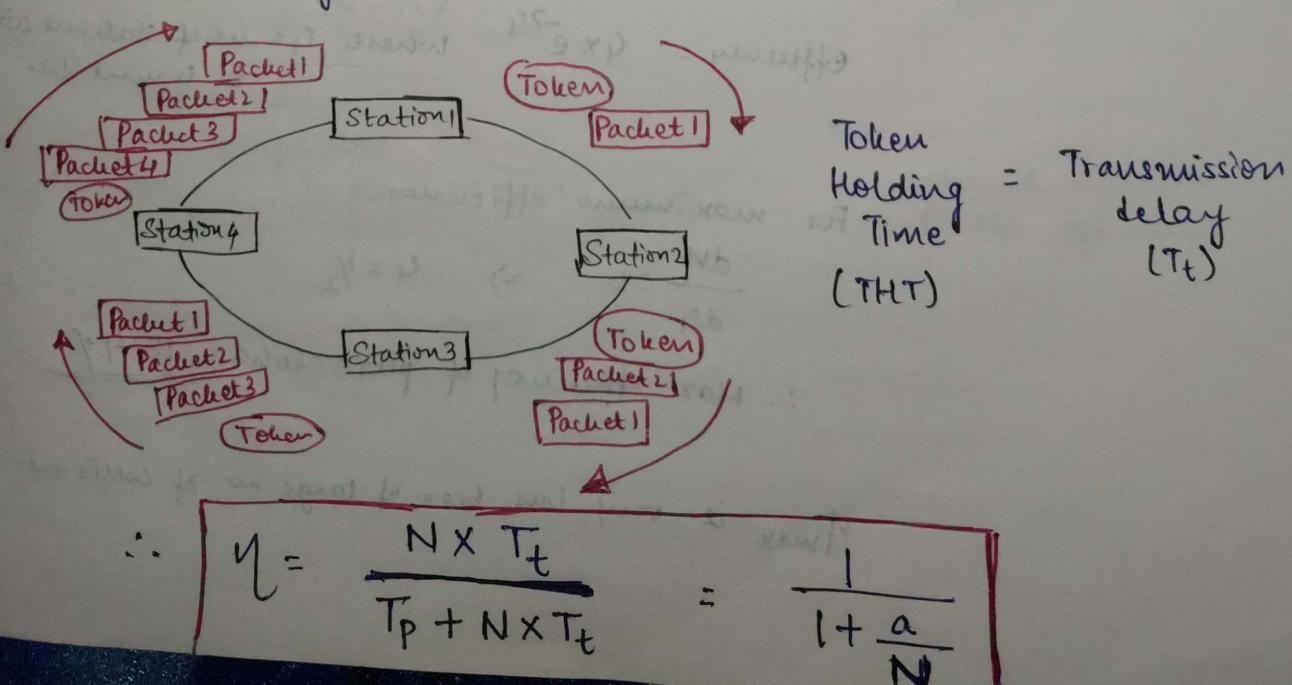
Assumption

- bit delay is 0 in most cases

$$\begin{aligned} \therefore \eta &= \frac{N \times T_t}{T_p + N \times THT} = \frac{N \times T_t}{T_p + N(T_t + T_p)} \\ &= \frac{1}{\frac{a}{N} + 1 + a} = \frac{1}{1 + (1 + \frac{1}{N})a} = \frac{1}{1 + (\frac{N+1}{N})a} \end{aligned}$$

Early Token Reinsertion

In this strategy, station releases the token immediately after putting its data packet to be transmitted on the ring.



Delayed Token Reinsertion (DTR) v/s Early Token Reinsertion (ETR)

Each station holds the packet until its data packet reaches back to it.

Each station releases the token immediately after putting its data packet on the ring.

There exists only 1 data packet on the ring at a given instance.

There exists more than one data packets on the ring at a given instance.

More reliable

Less reliable.

Low efficiency

High efficiency.

5. ALOHA

→ Pure Aloha

- ① stations can transmit data at any time whenever they want.
- ② After transmitting packet, station waits for some time.

If acknowledgement signal is received, then transmission is successful.

If not received, then station waits for random time 'back off time' & then transmits again.

After backoff limit is reached, it aborts the transmission.

$$\text{Efficiency} = \frac{q \times e^{-2q}}{q(q+1)} \quad \text{where } q = \text{no. of stations willing to transmit data.}$$

For maximum efficiency,

$$\frac{dq}{dq} = 0 \Rightarrow q = \frac{1}{2}$$

$$\therefore \text{Max. efficiency of pure aloha} = \underline{\underline{18.4\%}}$$

q_{\max} is very less bcoz of large no. of collisions.

→ Slotted Aloha

Time is divided into slots.

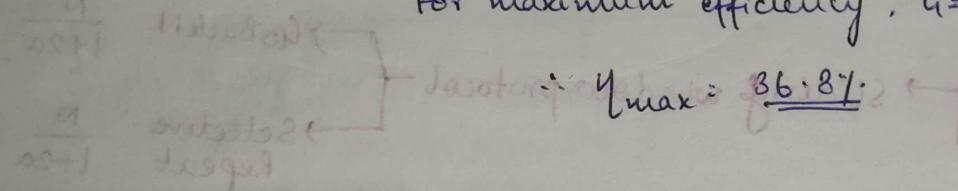
Every station can transmit only at the beginning of time slot.

[Collision only occurs when 2 stations try to transmit data at the beginning of same time slot]

Efficiency of slotted aloha, $\eta = G \times e^{-G}$

G = No. of stations willing to transmit.

For maximum efficiency, $G=1$



Pure Aloha

v/s

Slotted Aloha

Any station can transmit at any time. Any station can transmit at the beginning of time slot.

The time is continuous & the time is discrete and not globally synchronized. globally synchronized.

Vulnerable time = $2 \times T_t$

Vulnerable time = T_t

$$\eta = G \times e^{-2G}$$

$$\eta = G \times e^{-G}$$

$$\eta_{\max} = 18.4\%.$$

$$\eta_{\max} = 36.8\%.$$

(G=1/2)

(G=1)

Main adv -

simplicity in implementation

Main adv -

reduced collisions double efficiency.

Summary

Flow control methods :-

Make sure that the receiver is not overburdened with the packets.

→ Stop and wait protocol $\eta = \frac{1}{1+2a}$

→ Sliding window protocol

 → Go Back N $\frac{N}{1+2a}$

 → Selective Repeat $\frac{N}{1+2a}$

Access control methods

Methods that control the access of stations to the communication link.

→ Time Division Multiplexing

$$\eta = \frac{1}{1+a}$$

→ Polling

$$\eta = \frac{T_t}{T_{poll} + T_t + T_p}$$

→ CSMA / CD

$$\eta = \frac{1}{1+6.44a}$$

→ Token Passing

 → Delayed Token Reinsertion

$$\eta = \frac{1}{1+(N+1)a/N}$$

 → Early Token Reinsertion

$$\eta = \frac{1}{1+a_N}$$

→ Aloha

 → Pure Aloha

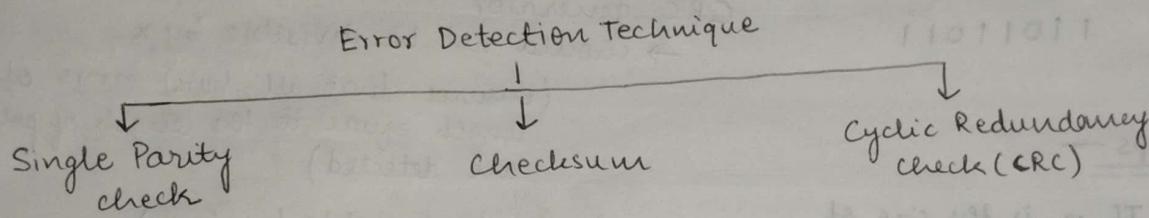
$$\eta = 4 \times e^{-2a} \quad \eta_{max} = 18.4\%$$

 → Slotted Aloha

$$\eta = 4 \times e^{-4} \quad \eta_{max} = 36.8\%$$

ERROR CONTROL METHODS

- ① Process of detecting and correcting data frames that have been corrupted or lost during transmission.
- Lost packets are handled by Flow control methods.
- ② Error detection is the technique used to check if some error occurred in the data during transmission.



Single Parity Check

In this technique, one extra bit called parity bit is sent along with original bits

Example —

- ① Data to be transferred = 100100
even parity is used.

$$\text{Parity bit} = 1$$

Code word received by receiver = 1001001

Receiver counts no. of 1's & determines if it is even or not

Advantages

- ① guarantees detection of odd number of bit errors.

Disadvantages

- ② cannot detect even no. of bit errors.

Steps — [Even parity]

- At sender side, total no. of 1's in data is counted.
Parity bit is set to 0/1 to make the total no. of 1's even.
→ Data + Parity bit is transmitted.
→ Receiver receives the packet & counts no. of 1's. If even, data is correct else it requests for retransmission.

- ② Data transferred = 110010
odd parity is used.

$$\text{Parity bit} = 0$$

$$\text{Code word generated} = 1100100$$

If no corruption takes place, receiver receives 1100100.

Cyclic Redundancy Check

based on binary division.

CRC Generator :- algebraic polynomial represented as bit pattern.
Bit pattern is obtained by the following rule -

Example -

CRC Generator is

$$x^7 + x^6 + x^4 + x^3 + x + 1$$

Bit Pattern is

$$11011011$$

CRC generator

→ should not be divisible by x

(ensures that all burst errors of length equal to the length of polynomial are detected)

→ should be divisible by $(x+1)$

(ensures that all the burst errors of odd no. of bits are detected)

Steps -

- ① If n is the size of CRC generator, $n-1$ bits are appended to the end of data.
- ② Divided by CRC generator.
- ③ Remainder obtained is called CRC.
- CRC consists of $(n-1)$ bits.

- ④ CRC is appended to the original data to be transmitted.

- ⑤ At Receiver side, code is again divided by CRC generator.

On division if remainder is 0, then, data is not corrupted else it is corrupted.

CRC generator

→ should not be divisible by x

(ensures that all burst errors of length equal to the length of polynomial are detected)

→ should be divisible by $(x+1)$

(ensures that all the burst errors of odd no. of bits are detected)

Example

Data to be transferred = 1101011011

CRC generator $\rightarrow x^4 + x + 1$

$$= 10011$$

$$\begin{array}{r}
 10011 \overline{)1101011011} \\
 10011 \\
 \hline
 0100111011 \\
 10011 \\
 \hline
 000001011
 \end{array}
 \quad \therefore \text{CRC} = 1011$$

~~प्राप्त रिसेप्टर~~
~~संकेत~~
~~ट्रॉट~~

Data transmitted $\rightarrow 11010110111011$

At receiver's end -

$$\begin{array}{r}
 10011 \overline{)11010110111011} \\
 10011 \\
 \hline
 01001110111011 \\
 10011 \\
 \hline
 0000010111011 \\
 10011 \\
 \hline
 00100011 \\
 10011 \\
 \hline
 000
 \end{array}$$

Checksum

~~KCATE 120 021~~

- ① If m bit checksum is used, data unit is divided into segments of m bit.
 - ② All the segments are added.
 - ③ 1's complement of the sum.
 - ④ The value obtained is called checksum.
-
- ⑤ Received data unit is divided into m bit segments.
 - ⑥ All the segments are added along with checksum value.
 - ⑦ Value obtained is complemented.
 - If $\text{result} = 0$, no corruption
If $\text{result} \neq 0$, corrupted data.

Example -

Data unit to be transmitted =

~~for error detection~~ = $100110011111000100010010010000100$

8 bit checksum is used

$$10011001 + 011100010 + 000100100 + 10000100$$

$$= 1000100011 \quad = 1200010010 \quad (8 \text{ bits})$$

(wrap up)

1's complement = 11011010

At receiver's end → sum of all segments + checksum value

$$= 00100101 + 11011010 = 11111111$$

1's complement = 00000000

∴ data is not corrupted

ISO OSI STACK

Career

Functions needed to
be implemented for
communication

Mandatory
functions

optional
functions

- ① Error control
- ② Flow control (receiver
should not be overflooded)
- ③ Access control
- ④ Multiplexing/Demultiplexing
- ⑤ Addressing

- ⑥ Encryption/Decryption
- ⑦ Checkpointing
- ⑧ Routing
- ⑨ Other

70 functions both optional
& mandatory

To implement the above functionalities, various reference models
are used

- in syllabus {
- ① ISO-OSI = 1100010001
 - ② TCP/IP = 0100010001
 - ③ ATM = 01011011
 - ④ X.25 = 01011011
 - ⑤ IEEE (deals with LANs) = 01011011 + 10011001
- } Reference
models.

ISO-OSI model

International Standard Organization - Open System Interconnection

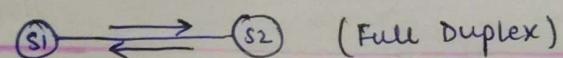
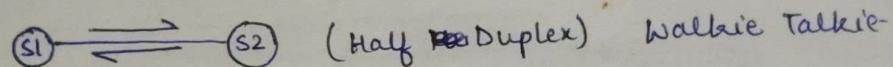
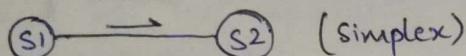
Layers in the OSI model -

- | | | |
|-------------------------|-----------------------|---|
| User
interactiveness | 1) Application layer | Advantages of layering |
| | 2) Presentation layer | |
| | 3) Session layer | |
| Thick layer | 4) Transport layer | ① Divide and conquer
② Encapsulation
③ Abstraction
④ Easy testing. |
| | 5) Network layer | |
| | 6) Data link layer | |
| | 7) Physical layer | |

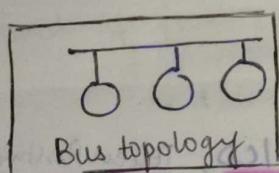
1) Physical layer

EMFP ① Physical layer deals with Electrical, Mechanical, Functional & Procedural characteristics

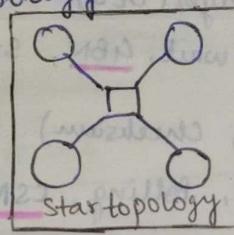
② Transmission modes:



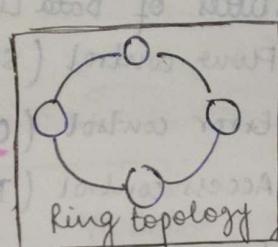
③ Deals with topology —



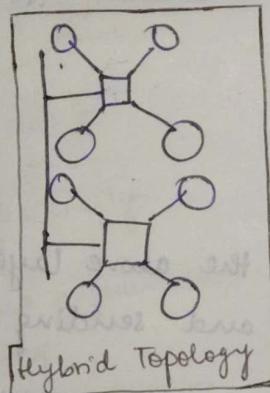
Bus topology



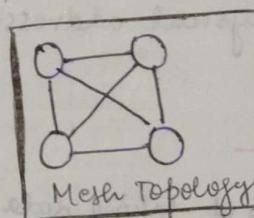
Star topology



Ring topology



Hybrid Topology



Mesh Topology

④ Deals with encoding (bits → signals/waves)

Manchester encoding

⑤ Represented by \overline{L}

1 Represented by L

Differential Manchester Encoding

⑥ Represented by \overline{L}/L
1 Represented by L/\overline{L}

In both the encodings,

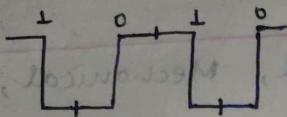
$$\text{Band Rate} = \frac{2 \times \text{Bit Rate}}{\text{No. of bits sent per second}}$$

No. of voltages being sent per second

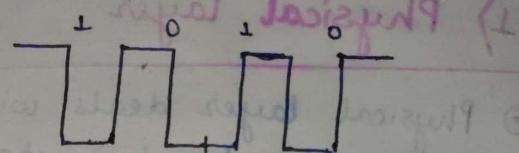
↳ No. of bits sent per second

Example of encoding -

10 10



Manchester
encoding



Differential Manchester
encoding

Packet header (excluding flow)

(excluding flow)

(excluding flow)

(excluding flow)

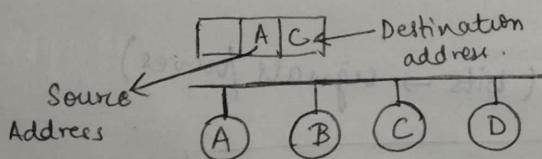
2) DATA LINK LAYER (limited to be used within a network)

Responsibilities of Data Link Layer (DL) (pocket view also)

- ① Flow control (Stop & wait, CBN, SR)
- ② Error control (CRC, checksum)
- ③ Access control (TDM, Polling, CSMA/CD, TOKEN PASSING, Aloha)
- ④ Framing
- ⑤ Physical addressing.

Framing -

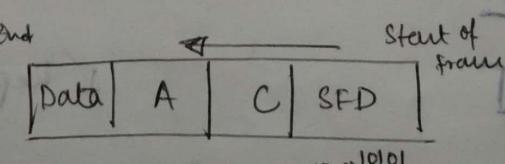
⑥ process of taking data from the above layer (network layer), putting it in a frame and sending it.



Starting of the packet is seen by B & it will discard it since DA is C.

Every ^{station} frame should know beginning of the frame -
∴ There is a special field in the beginning called

* Starting Frame Delimiter (SFD)

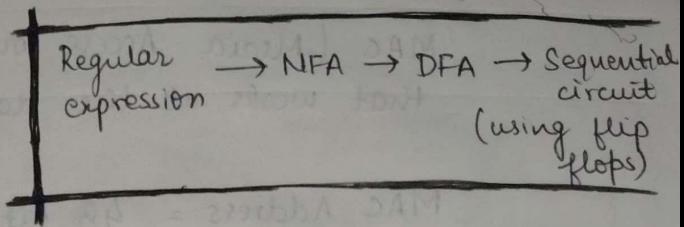


spurious art word NI
= art word

- ① SFD represents that a frame is coming and alerts all the stations.
- ② SFD → sequence of bits that is entirely different from data.
- ③ Bits in SFD →

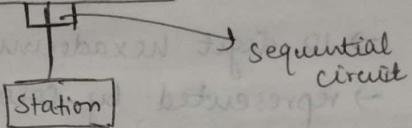
$$(10101010 \dots 11) (0+1)^*$$

SFD Data



Sequential circuit has the capability of recognising the bit pattern of SFD.

$D \mid A \mid C \mid SFD$



End of frame →

Frames

Fixed size/length frames

Since frame size is fixed, end of frame can be easily determined.

Problem:- Internal fragmentation
Solution:- Padding
↳ Adding dummy bits.

Variable size/length frame.

Ethernet (CSMA/CD)
Having a field saying length of the frame

Problem:- Corruption in length field.

Token passing
Having an ending delimiter

Problem:- ED can match with the data available

$ED \mid Data \mid A \mid C \mid SFD$

costly & obsolete
Character stuffing
used when only characters are used in the data.

If data contains ED, a byte is stuffed into data to differentiate it.

ED = \$ escaped using 10\$

If data contains 10\$, 101010\$ is used.

Imp for gate

Bit stuffing
Sender stuffs a bit to break the pattern

Receiver receives frame.

If data contains 011101, it removes the 0 & reads the data.

Data = 011100011110 ED = 0111

Data after bit stuffing -

011010001101100

Solution

Physical Addressing

MAC (Media Access Control) is the physical address that works at the data link layer.

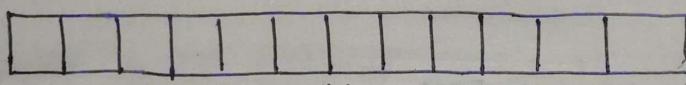
MAC Address = 48 bit no.

embedded into Network Interface Card (NIC) during manufacturing.

→ 12 digit hexadecimal number

→ represented by Colon Hexadecimal notation.

cisco	CC:46:D6 → XXXXXX
3C : 5A : B4 → Google	
3C : D9 : 2B - Hewlett Packard	

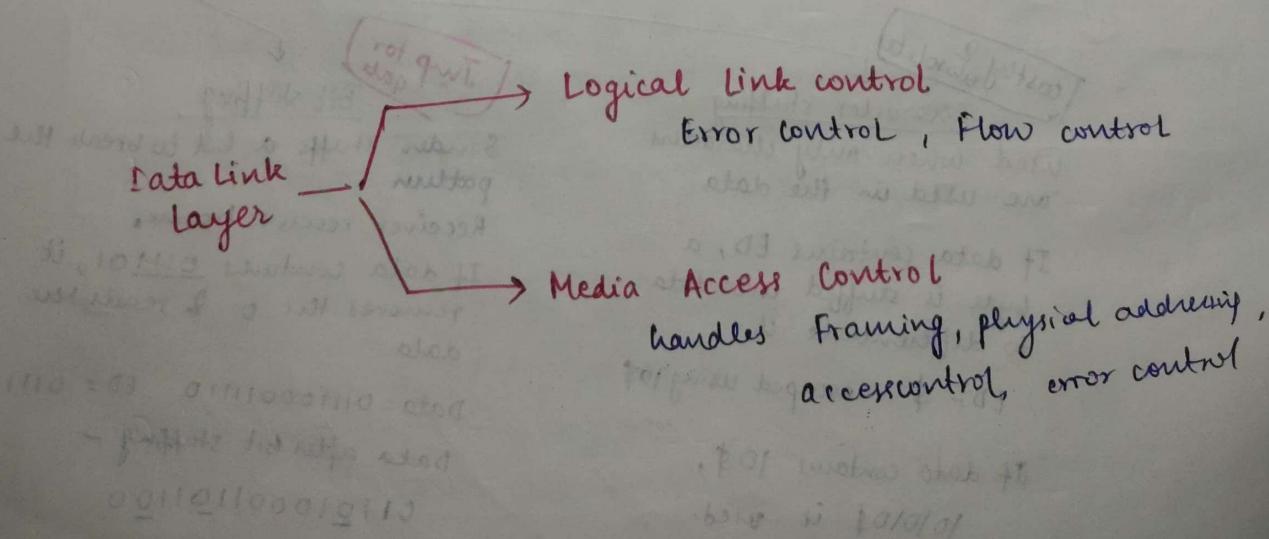
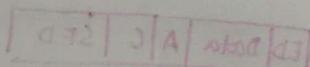


Identifies the manufacturer.
Called
Organizational Unique Identifier (OUI)

Network Interface controller assigned by manufacturer
(Data & Sl. No.)

Command to see MAC address → ipconfig/all

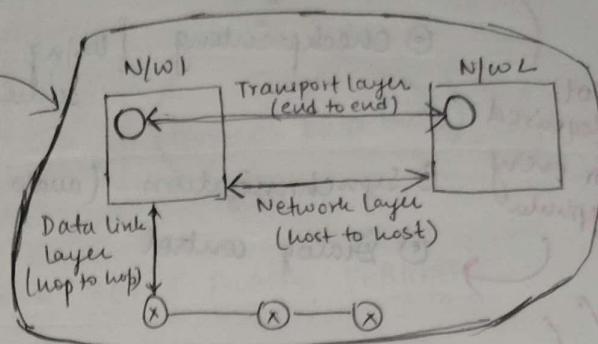
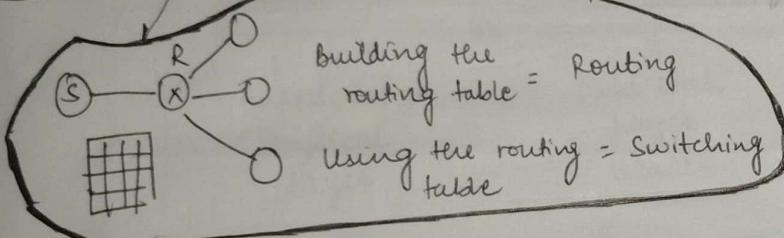
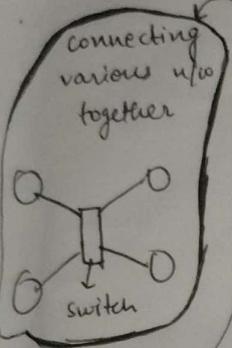
Token Ring, LAN Technologies like Ethernet use MAC address as physical address.
AppleTalk does not use MAC Address as physical address.



3) NETWORK LAYER

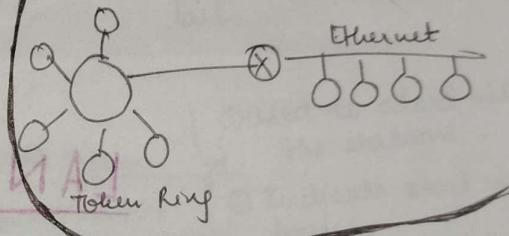
Main responsibilities —

- ① Host to Host connectivity
- ② Logical Addressing (at IP)
- ③ Switching
- ④ Fragmentation
- ⑤ Routing
- ⑥ Congestion Control (at TCP)



AIRTEL MASTERS 32389

connecting n/w of different topology.
Max. size of packet for ethernet = 1500 bytes
.. packet needs to be broken down into small parts.



ETHERNET (IEEE 802.3)

4) TRANSPORT LAYER

Main responsibilities are

- ① End to end communication using port numbers
(Port numbers = Service Point Addressing)
- ② Flow control (SR)
SR → software layer (TL)
GBN → hardware layer (DLL)
- ③ Error control (checksum)
- ④ Segmentation (Takes data from Application, Presentation, Session layer) and divides into segments.
- ⑤ Multiplexing & Demultiplexing
- ⑥ Congestion control
(How to deal with unpredictable N/S)

SESSION LAYER

- ① Authorization and authentication
- ② Checkpointing [losing interconnection, movie downloads from few same point (not beginning) if internet is available]
 Not required for every application
- ③ Synchronization (audio & video)
- ④ Dialog control

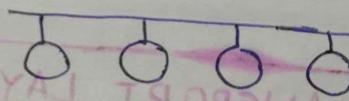
PRESENTATION LAYER

- ① Character translation
- ② Encryption & decryption
- ③ Compression (.zip)

LAN Technologies

ETHERNET (IEEE 802.3)

- ① Topology : Bus Topology



- ② Access Control Method : CSMA/CD

- ③ Encoding : Manchester encoding

0 represented by 
1 represented by 

- ④ No acknowledgement signal is sent via cable

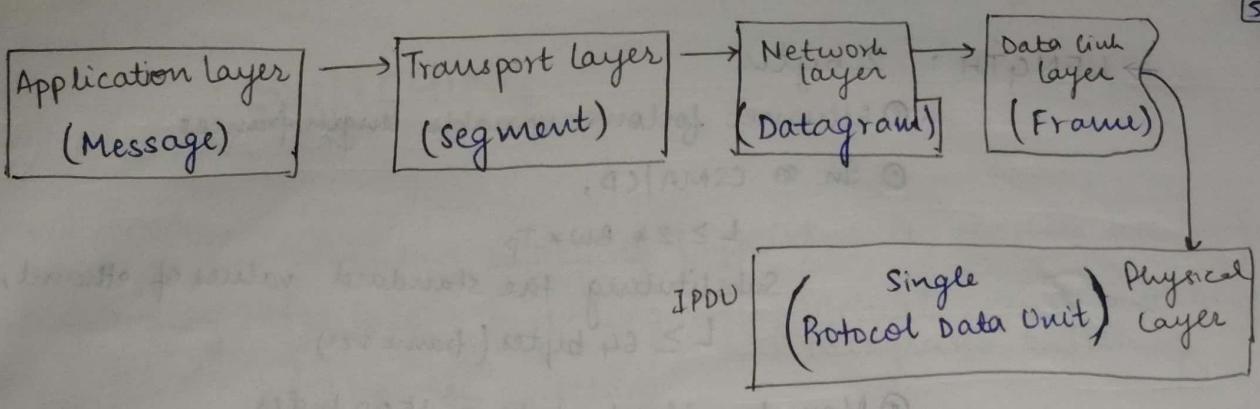
- ⑤ Data Rates (Bandwidth) 10mbps - 100mbps - 1gbps

fast
ethernet

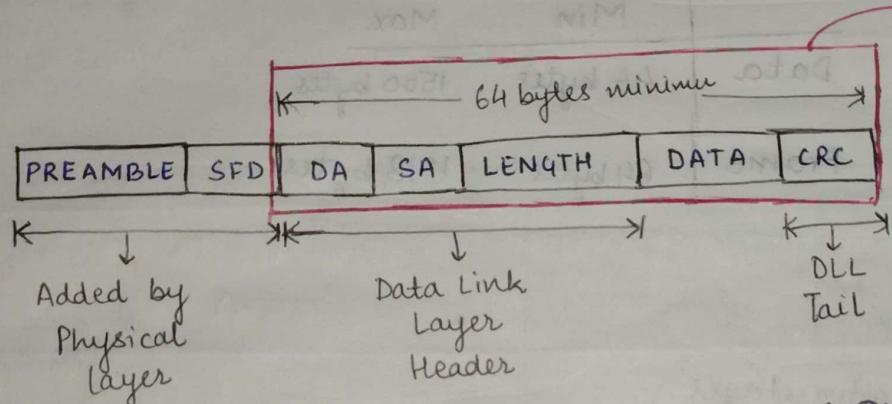
Gigabit
ethernet

- ⑥ Ethernet operates at DLL

(LAN Technologies are dealt in DLL)



FRAME FORMAT OF ETHERNET (IEEE 802.3 FRAME FORMAT)



- PREAMBLE → 7 bytes 101010....10 {the stations
 - SFD : [Start Frame Delimiter] : 1 byte = 10101011 ⑤ Indicate start of frame, synchronization
 - DA : Destination address } MAC addresses
 - SA : Source address } MAC addresses
 - CRC : Cyclic Redundancy check : 4 bytes (32 bit ~~checksum~~ is used) CRC generator

Types of MAC addresses

UNICAST

LSB of 1st bytes
is 0

1A:2B:3C:48:56:6F

Unicast DA means
packet is sent to
only 1 host.

LSB of first byte
is 1

LSB of first byte
is 1

11°23':30":48:56:6F

Multicast DA means that packet is sent to a group of hosts.

MAC address of group
= Multicast MAC.

All the bits are
1's in MAC address.

FF:FF;FF:FF:ff:ff

The packet is sent to

all the costs in

the network

→ LENGTH : 2 bytes

- ② Ethernet follows variable length frames.

- ③ In ② CSMA/CD,

$$L \geq 2 * BW * T_p$$

Substituting the standard values of ethernet,
 $L \geq 64$ bytes (frame size)

- ④ Max length of data = 1500 bytes

* Preamble &
SFD are not
included in
frame size bcoz
frame = DLL

PL adds
preamble & SFD

	Min	Max
Data	46 bytes	1500 bytes
Frame	64 bytes	1518 bytes

No. of fields in Disadvantages

- ① Not applicable for real time applications
- ② Not applicable for interactive applications
- ③ No priorities. So not suitable for client server applications

7 bytes	6 bytes	2 bytes	4 bytes
Preamble	SFD	DA	SA

1 byte

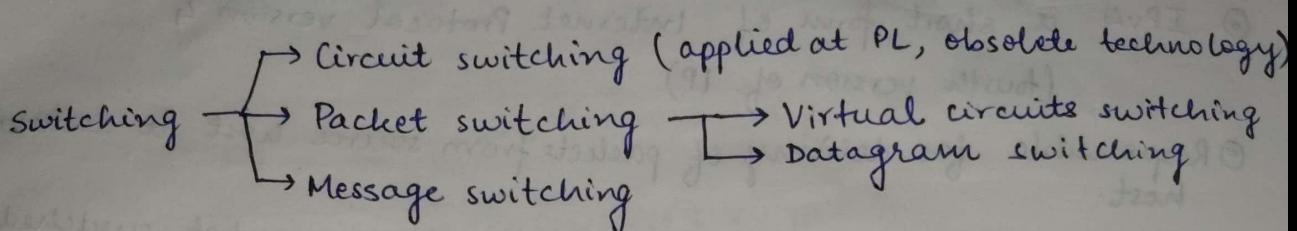
6 bytes

2 bytes

4 bytes

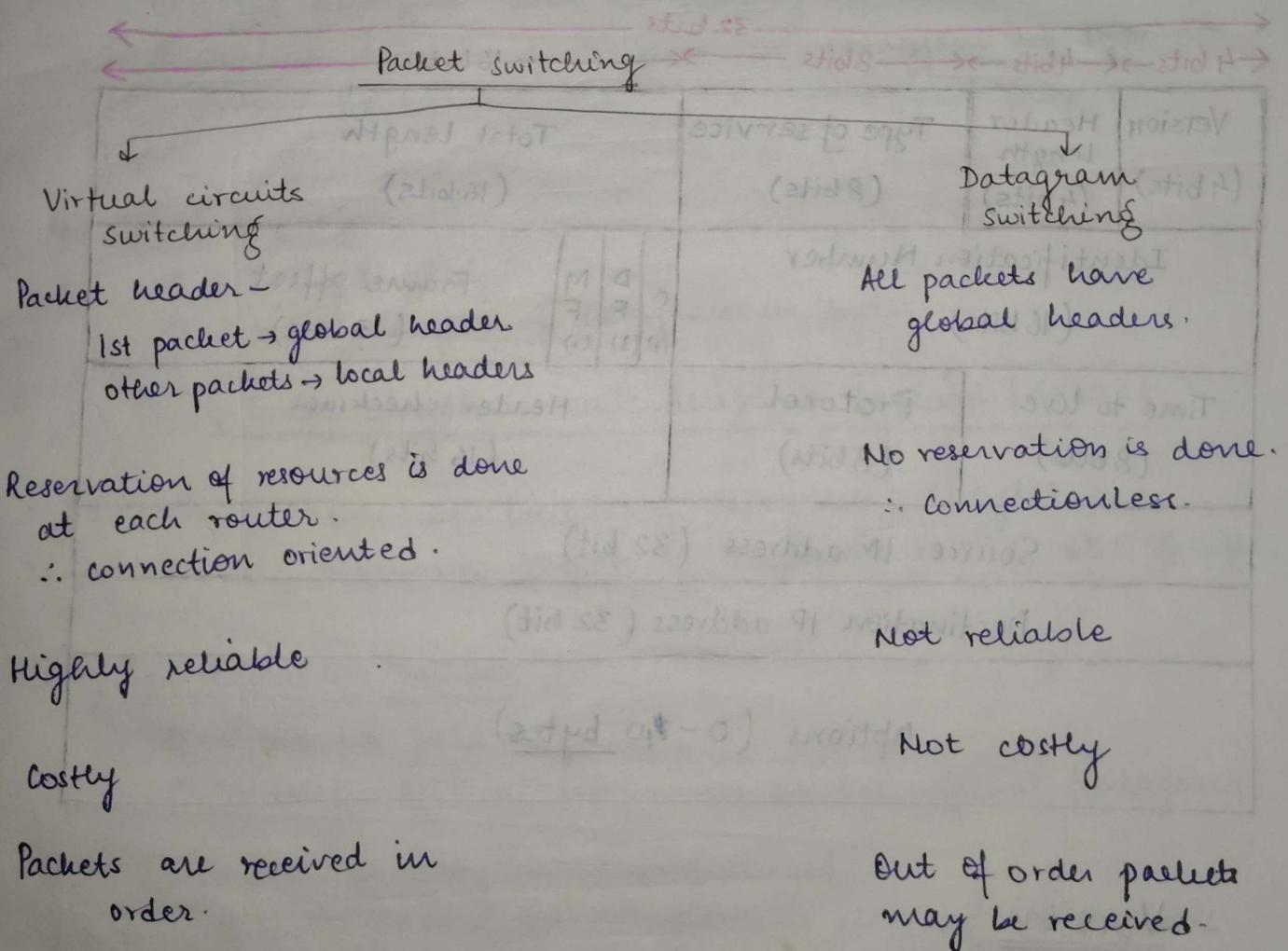
SWITCHING

Hv9I



→ Switching is done at network layer.

Hv9I



IPv4

- ① IPv4 is short form of Internet Protocol version 4
 - (fourth version of IP)
- ② Responsible for delivery of packets from source host to destination host.
- ③ IPv4 is a connectionless protocol for use on packet switched network

IPv4 Header

Version (4 bits)	Header Length (4 bits)	Type of service (8 bits)	Total length (16 bits)
Identification Number (16 bits)	O (1) D (1) M (1) F (13 bits)	Frame offset (13 bits)	
Time to live (8 bits)	Protocol (8 bits)	Header checksum (16 bits)	
Source IP address (32 bit)			
Destination IP address (32 bit)			
Options (0 - 40 bytes)			

1. Version

- ④ 4 bit field that indicates version of IP being used.
- ④ For IPv4, it is 0100
- ④ Datagrams with different versions are parsed differently.
 → IPv4 datagrams are parsed by version 4 parsers.
 → IPv6 datagrams are parsed by version 6 parsers.

2. Header length

- ① 4 bit word that tells the length of header in 4 byte words.
- Example:- If length of header = 44 bytes,
header length field contains $44/4 = 11$

$$20 \text{ bytes} \leq \text{Header size} \leq 60 \text{ bytes}$$

$$\textcircled{5} \quad 5 \leq \text{Header length field} \leq 15$$

Scaling factor = 4

3. Type of service

- ① 8 bit ~~bits~~ field used for Quality of service.
- ② Special treatment is given to datagram for ~~TCP~~ a particular service.

4. Total length

- ① 16 bit ~~bit~~ field.
- ② Total ~~be~~ length of datagram in bytes.

$$\text{Total length} = \text{Header length} + \text{Payload (data)}$$

Minimum length of datagram = 20 bytes

Maximum length of datagram = $2^{16} = 65536$ bytes

5. Identification Number

- ① 16 bit ~~bit~~ field.
- ② Identification of fragments in the original datagram.

When an IP datagram is fragmented,

- ③ Each fragmented datagram is assigned the same identification number.
- ④ The number is important for reassembly of ~~po~~ fragmented datagrams.

6. DF bit

Do Not Fragment bit.

0 → grants permission to intermediate devices to fragment datagram if needed.

1 → Datagram cannot be fragmented.

If now requires datagram to be fragmented to travel further but the settings do not allow, then, the datagram is discarded.

An error message is sent to the sender saying that the packet has been discarded due to its settings.

7. MF bit

More Fragments bit.

0 → indicates to the receiver that the datagram is either the last fragment in the set or it is the only fragment.

1 → indicates to the receiver that the current datagram is a fragment of some larger datagram.

More fragments are following.

MF bit is set to 1 on all fragments except the last one.

8. Fragment offset

① 13 bit field.

② Indicates the position of a fragmented datagram in the original unfragmented datagram.

③ First fragmented datagram has a fragment offset 0.

Fragment offset for a given fragmented datagram = $\frac{\text{No. of bytes ahead of it in the original unfragmented datagram}}{\text{data}}$

④ Scaling factor = 8

9. Time to live

- ① TTL is a 8 bit field.
- ② indicates maximum number of hops a datagram can take to reach the destination.
- ③ Main purpose of TTL is to prevent the IP datagrams from looping around forever in a routing loop.

Value of TTL is decremented by 1 when

- !! ① Datagram takes a hop to any intermediate device having network layer.
- !! ② Datagram takes a hop to the destination

Datagram is discarded if value of TTL is 0 before reaching destination. — *infinite loops*

short burst ↘

10. Protocol

- ① 8 bit field.
- ② Tells the new layer at the routers / destination to which protocol the IP datagram belongs.

③ Protocol no—

ICMP → 1 14MP - 2

TCP → 6 UDP - 17

Priority of protocols — TCP > UDP > 14MP > ICMP

11. Header checksum

① 16 bit field.

② checksum value of entire header.

At each hop —

- ③ Header set checksum is compared
- ④ If mismatched, datagram is discarded.
- ⑤ Router modifies the checksum field whenever it modifies IP header.

Fields that can be modified —

1. TTL
2. Options
3. Header length
4. Total length
5. Fragment offset.

Computation of header checksum includes IP header only.

Errors in the data field are handled by the encapsulated protocol.

12. Source IP Address

- ① 32 bit field
- ② logical address of sender of the datagram.

13. Destination IP Address

- ① 32 bit field
- ② logical address of destination of datagram.

14. Options

- ③ 0 - 40 bytes

purpose of options -

Record Route

This option is used to record the IP addresses of the routers through which datagram passes.

Max. no. of IPv4 router addresses that can be recorded = 9

Source Routing

This option is used to specify the ~~route~~ that the datagram must take to reach destination.

Loose routing

only some routers are given in the path.

Strict Routing

all the routers are given.

Padding

To make header length multiple of 4.

FRAGMENTATION

- Process of dividing a datagram into fragments during its transmission.
- Done by intermediary devices like routers at the network layer.

Need -

Each n/w has its MTU (Maximum Transmissible Transmission Unit)
 maximum size of datagram
 that can be sent to it.

Fragments

Datagrams of size greater than MTU cannot be transmitted through the network.

Datagram Fragmentation

When router receives a datagram to transmit further, it examines

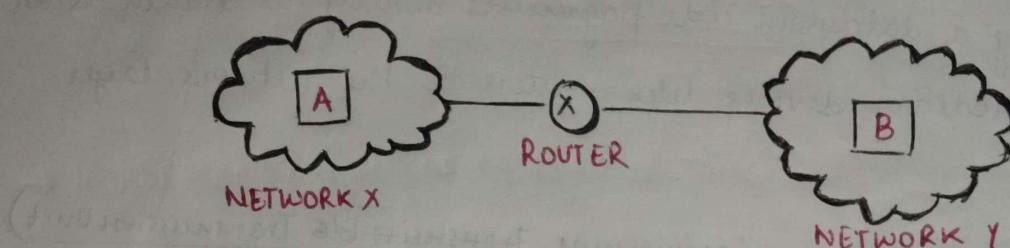
- Size of datagram
- MTU of destination network.
- DF bit in IP header.

<u>Cases</u>	<u>DF bit</u>	<u>Action</u>
Size \leq MTU	-	Transmit the datagram
Size $>$ MTU	0	Datagram is fragmented
Size $>$ MTU	1	Datagram is discarded

Changes made by the router during fragmentation

- Total length field is changed to the size of fragment.
- MF bit for every datagram other than last fragment is set to 1.
- It changes the fragment offset value
- Recalculates the header checksum.

IP Fragmentation example -



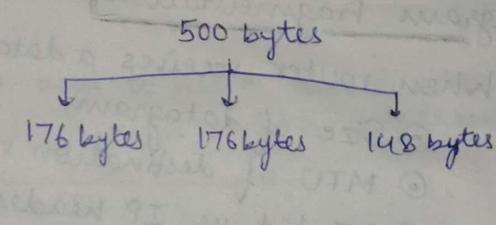
MTU of n/w X = 520 bytes

MTU of n/w Y = 200 bytes

Host A wants to send a message to host B.

Greatest no less than 180 which is the multiple of 8 = 176

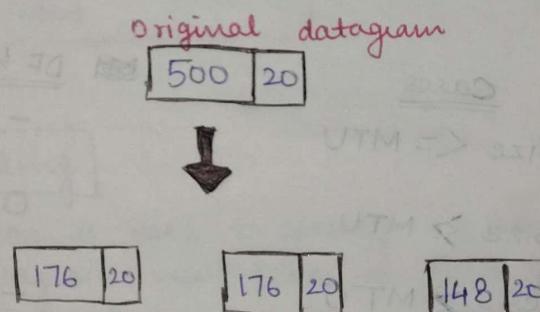
∴ The datagram is divided as follows —



Length of datagram (520 bytes)

< MTU of n/w Y.

∴ DF bit is set to 0, therefore, fragmentation is allowed.



Size of fragments should be
multiple of 8 (because
frame offset value is scaled
to 8).

Maximum value

• size of datagram header
in each fragment = 20 bytes

∴ Payload length maximum
that can be transmitted
 $= \frac{200}{8} - 20 = 180$ bytes

1st fragment —

① Header length field value = 5

② Total length field value = 196

③ MF = 1

④ Fragment offset = 0

⑤ Header checksum is recalculated

⑥ Identification no. is same as original

3rd fragment

Header length = 5

Total length = 168

MF = 0

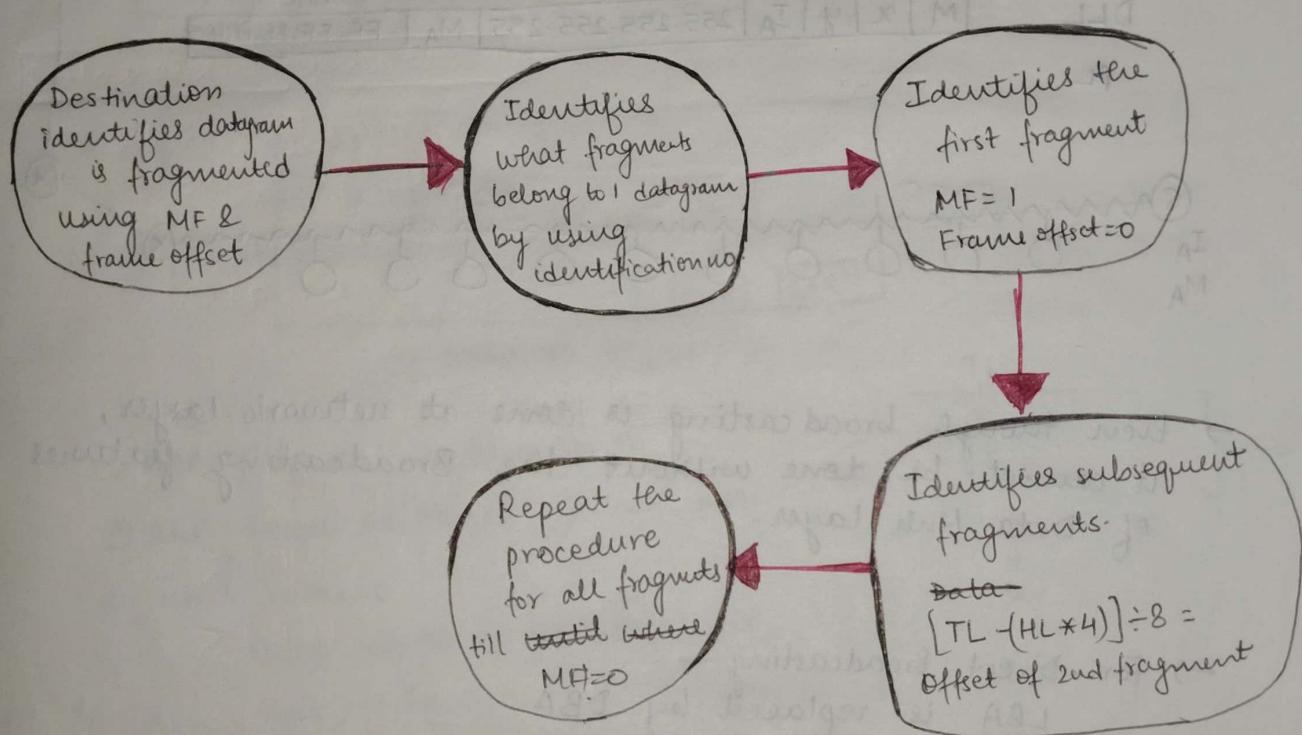
Fragment offset = $\frac{176 + 176 - 352}{8} = 44$

Reassembly Algorithm

165

Depending upon MF and frame offset, receiver identifies that the datagram is fragmented.

<u>MF</u>	<u>Frame offset</u>	
1	0	1st fragment
1	!0	Intermediate frame
0	0	only single datagram (No fragmentation)
0	!0	Last fragment



Important points -

② Different fragments from same datagram can take different routes to reach destination

③ Reassembly is always done at destination (never at routers)

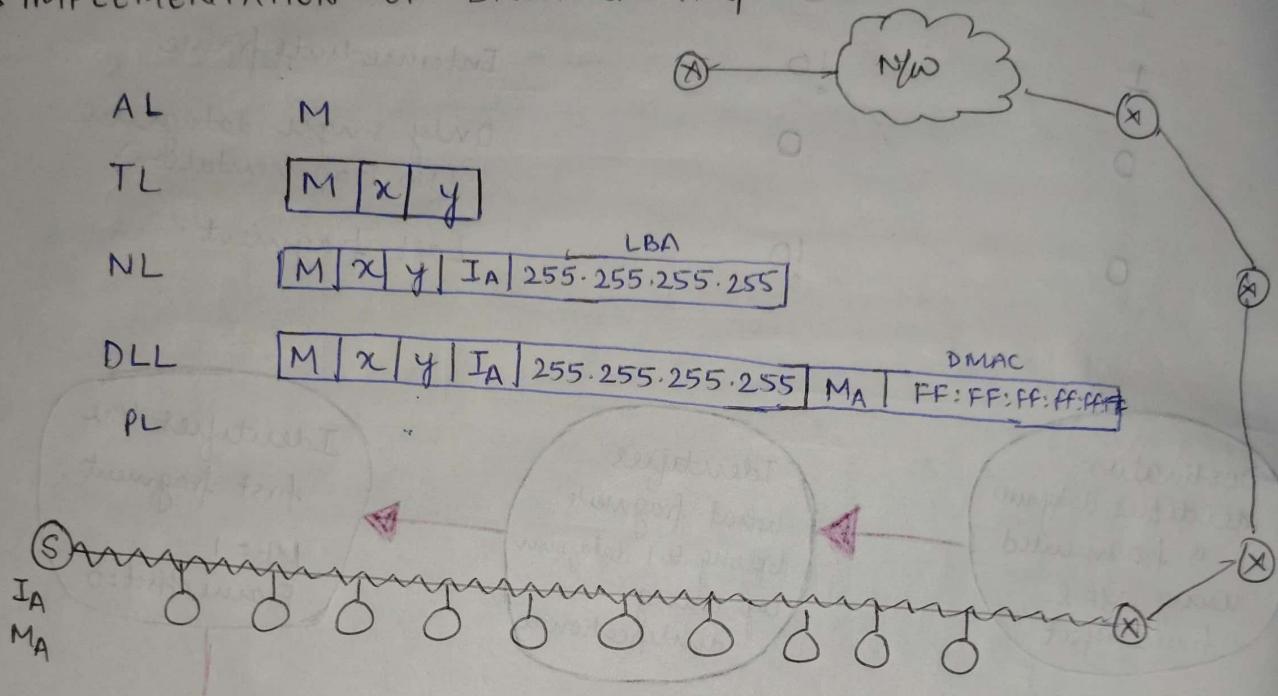
$$\text{Total overhead} = (\text{No. of fragments} - 1) \times \text{size of IP header.}$$

$$\text{Efficiency} = \frac{\text{Data without header}}{\text{Data with header}}$$

$$\frac{\text{Bandwidth utilization}}{\text{Throughput}} = \text{Efficiency} \times \text{Bandwidth.}$$

PROTOCOLS AND CONCEPTS AT NETWORK LAYER

→ IMPLEMENTATION OF BROADCASTING



→ Even though broadcasting is done at network layer, it cannot be done without the Broadcasting features of Data link layer.

→ For Direct Broadcasting -

LBA is replaced by DBA

DMAC replaced by next router MAC address

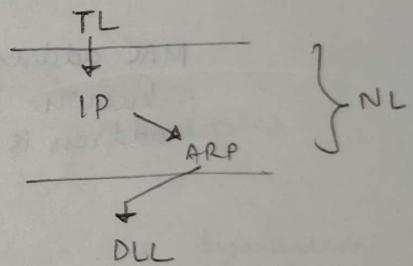
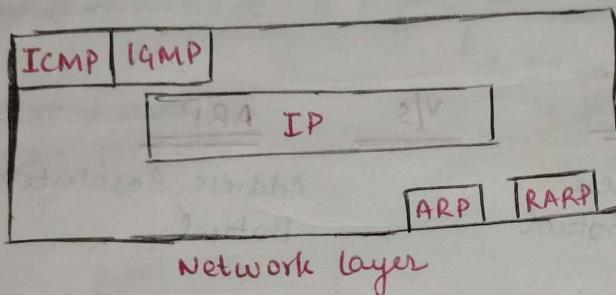
Router routes the message as unicast message & when the router forwards to the nfw, then, DBA is changed back to LBA & DMAC will be FF:FF:FF FF:FF:FF

ARP (Address Resolution Protocol)

Need: Most of the computer programs/applications use logical address (IP address) to send/receive messages. However, the actual communication happens over physical address (MAC address).

So, destination MAC address is required to communicate with other prot devices.

Function of ARP \Rightarrow IP address \rightarrow MAC address.



- ① ARP Request is broadcast
- ② ARP Request Response is unicast
- ③ used when

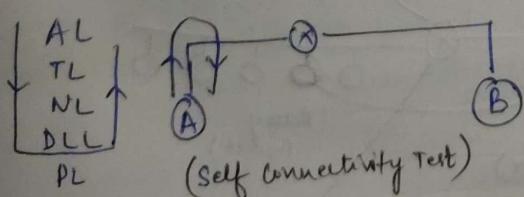
Host needs to find MAC of host

Host \rightarrow Router

Router \rightarrow Host

Router \rightarrow Router

Spec
Special Address 127 / Loop Back Address



If a packet is sent with destination address 127 then the packet is again going to come back to the same host.

The command to test is ping 127.0.0.1. We are supposed to see RTT as $\approx 15\text{ms}$.

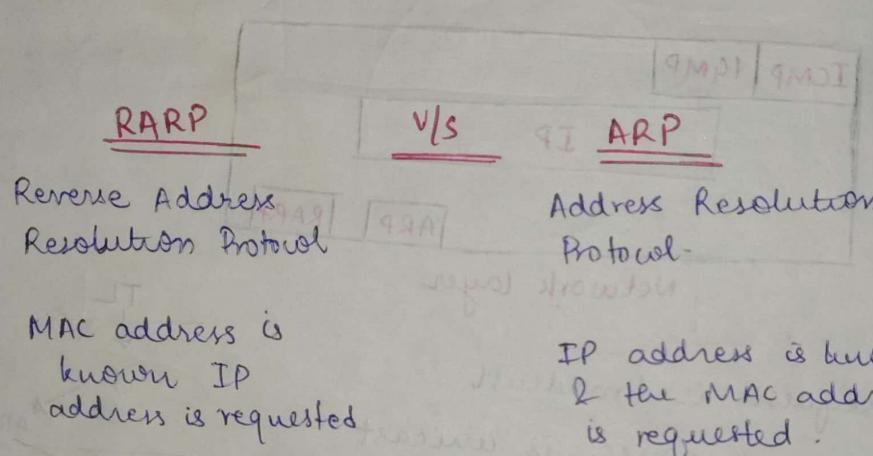
If something like Timeout appears, it is wrong in our NIC and we should troubleshoot.

~~Not in~~ RARP (Reverse Address Resolution Protocol)

- ① IP of a system is not constant. Therefore it isn't stored in the ROM, while MAC address is stored in ROM.
- ② So, RARP is used to get IP address from MAC address.
- ③ NFS → Network File Server
Contains a table of IP & MAC addresses.

Disadvantages

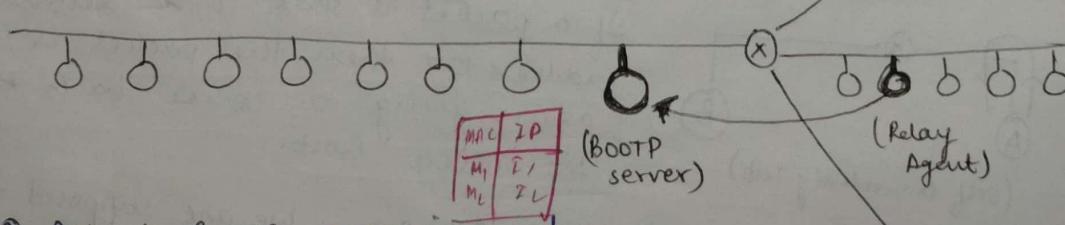
- ① Every n/w should have RARP server
- ② static mapping (no. of IP addresses > no. of hosts)



BOOTP (Bootstrap Protocol)

- ① Main difference b/w RARP & BOOTP — BOOTP works at AL, and RARP works at DLL.

AL	R
TL	R
NL	0.0.0.0 255.255.255.255
DLL	MAC FFFFFFFFFFFF
PL	



- ① Only 1 BootP server is used.
- ② Relay agents are present in those n/w which do not have BootP server.

Relay agent can communicate with the BOOTP server because it knows the IP address of BOOTP server. [9MCI] 16

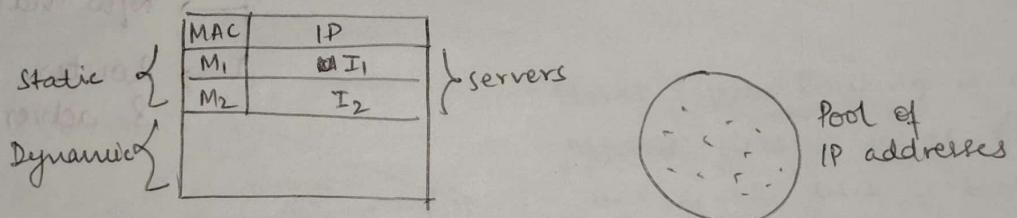
- Adv → Only 1 Bootp server is used
Disadv → Mapping table is static

DHCP (Dynamic Host configuration Protocol)

- ① Main difference b/w DHCP and BOOTP is that the table is not static in DHCP.
② DHCP has a table at DHCP server & the table has 2 parts — static part & dynamic part.

Servers are given permanent IP address

IP address is assigned when required valid upto lease time



If host A, M_A wants its IP address then the dynamic part will look like.

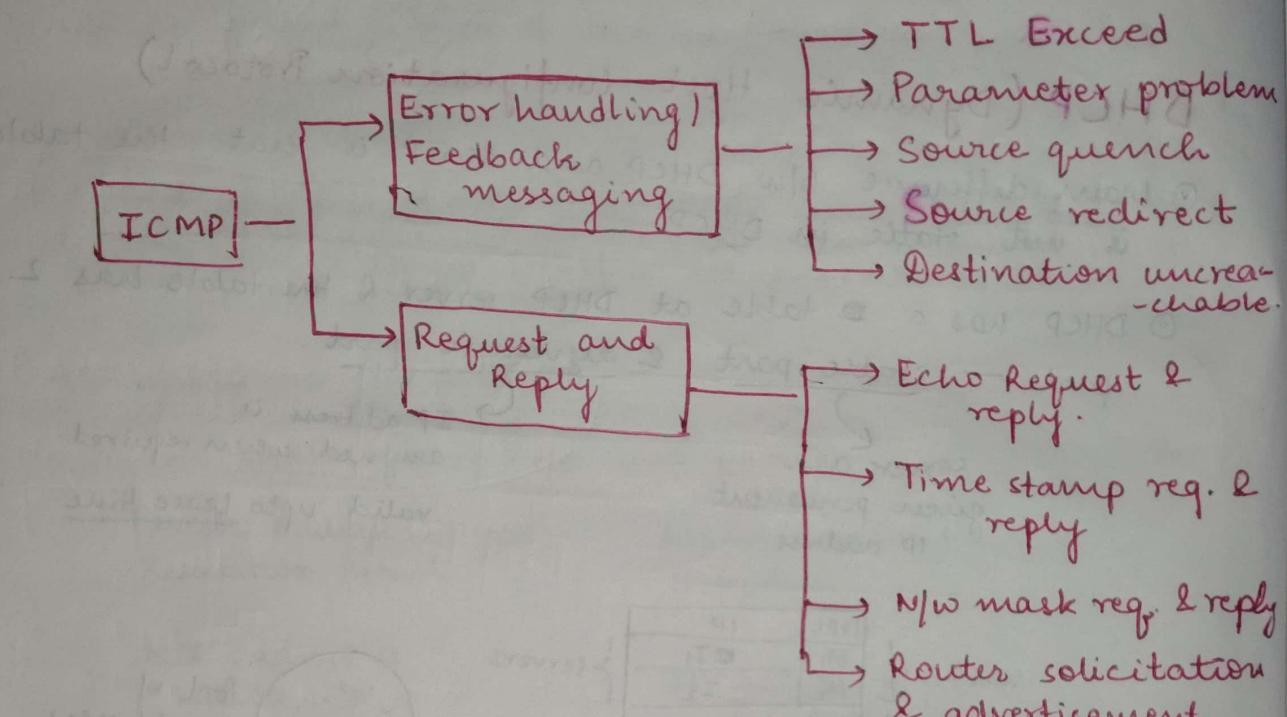
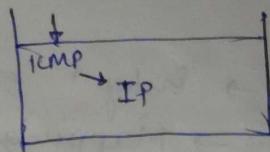
MAC	IP	Lease Time
M _A	I _A	10min
M _B	I _B	10min

- ① Only 1 DHCP server is enough
② Table is dynamic. (No. of IP addresses used = No. of hosts online)
③ To make DHCP compatible with BootP, both DHCP & BOOTP should have same port number.

→ DHCP is operated at application layer
∴ cannot be implemented on Router as Router has only 3 layers.

ICMP [Internet Control Message Protocol]

② works at Network layer



Error Handling / Feedback messaging

whenever there is an error/ anything goes wrong, ICMP protocol is used.

IP is unreliable.

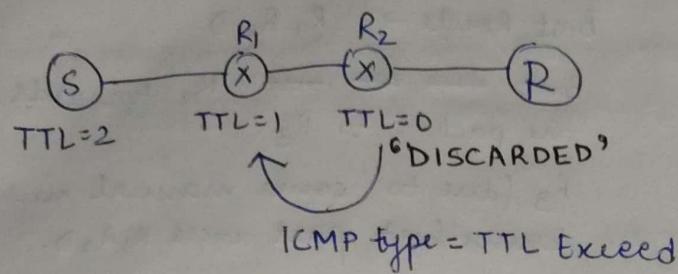
Packets are generally discarded at routers due to congestion

- If IP packet is lost, ICMP packet is generated
- If ICMP packet is lost, no ICMP packet should be generated

beoz it may fall into infinite loop.

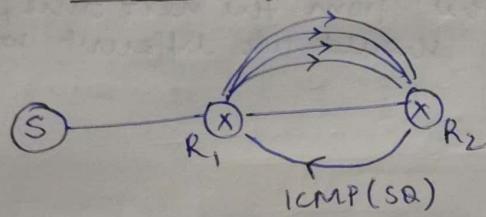
∴ (ICMP + IP) → unreliable.

1. TTL Exceed



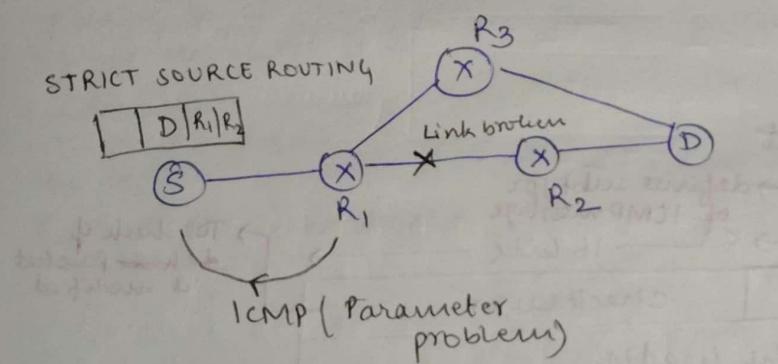
If TTL is 0 before the packet has reached its destination, then, router sends ICMP packet to the source.

2. Source quench



R₁ sends a lot of packets & R₂ cannot handle them. It sends ICMP packet of type 'source quench' asking R₁ to send packets via a different route.

3. Parameter problem



Strict Source Routing is used. Options field is set as [D, R₁, R₂] but R₁ → R₂ link is broken. Packet is discarded.

ICMP (Parameter Problem) is sent to the sender.

4. Destination unreachable

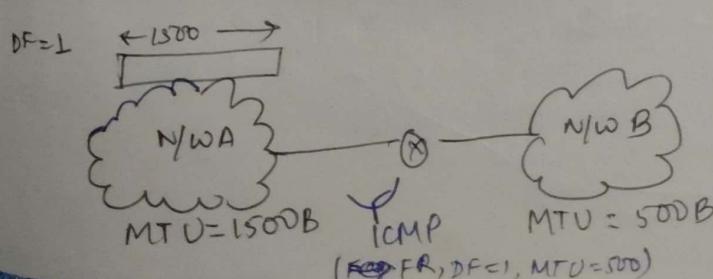
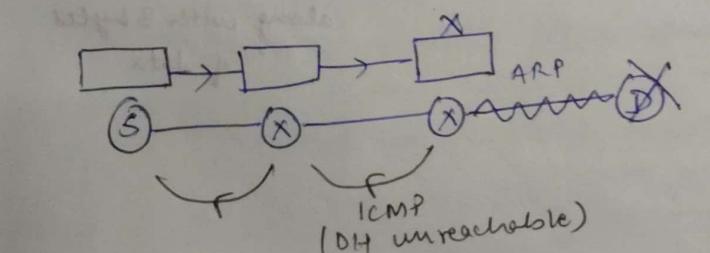
→ It is of 2 types
↳ Destination Host Unreachable
↳ Destination Port Unreachable

→ R₂ sends ARP request for getting MAC address of the host.

If host is down, R₂ sends ICMP(DHU) error message

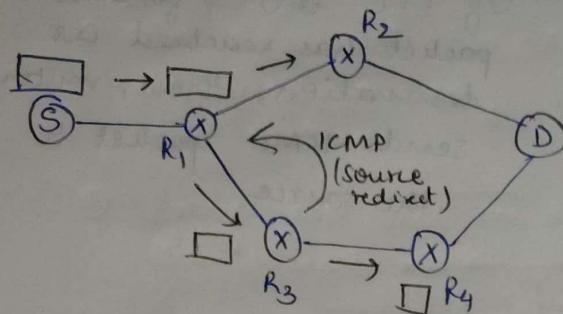
DF bit is set to 1 but MTU requires fragmentation

FR = Fragmentation Required.



5. Source Redirect

- ① packet is not discarded.
② it is warning not error message



If a packet is fragmented into 100 parts and it is discarded, then only 1 ICMP packet is sent.

Bcoz anyway the sender will have to send the ~~entire~~^{all} mess-100 fragments again.

→ If 3rd packet is lost, no ICMP is generated.

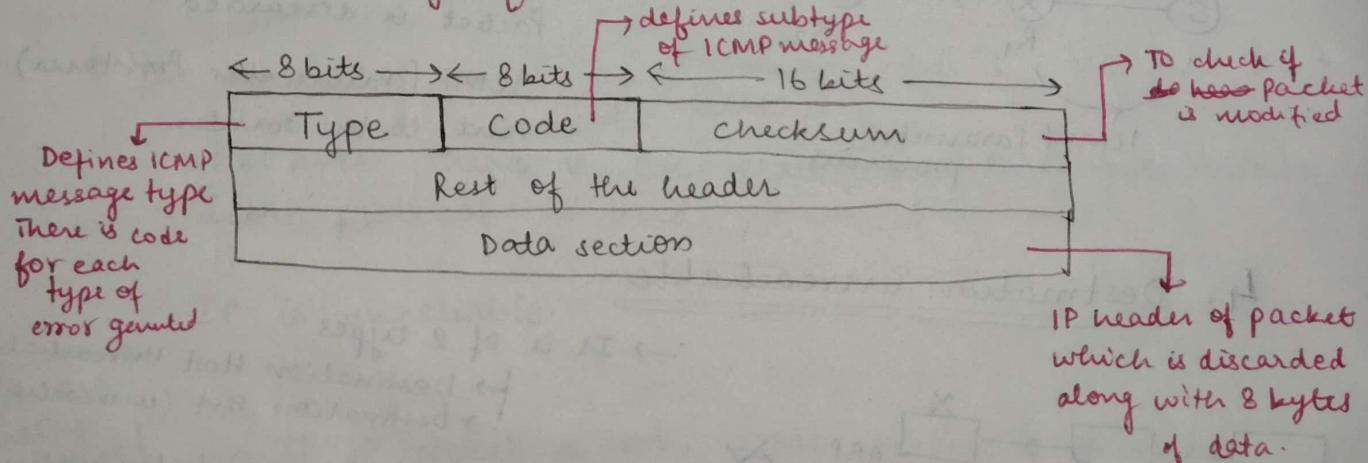
Best Route → R₁, R₂, D

But due to some R₁ forwards the packet to R₃.

R₃ (due to some manual mechanism) knew best part was R₁, R₂, D.

It will forward the packet to R₄ but at the same time, it will send ICMP packet to R₁ telling it that from the next time, it should use different route.

ICMP message format



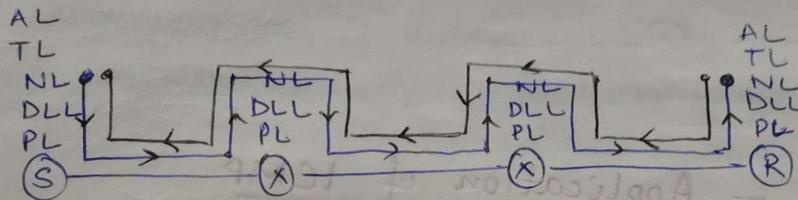
ICMP Request and Reply messaging

① Echo Request and Reply

② used to test whether the NL of destination and the intermediate routers are working or not.

③ PING command is used = Packet Internet Groper.

→ works at NL not AL
 ∵ not client-server



② Router solicitation and advertisement

station present in n/w
 sends broadcasting
 messages to all the
 routers connected to it
 router which is avail-
 able sends back
 the reply.

It is then made default
router/gateway.

→ Newly added router to a network sends message to all the hosts that it is available to be used as default router.

③ Time stamp Request and Reply

④ N/w devices are available in different parts of the world.

⑤ different time zones.

→ problem of synchronization

→ As a result flow of messages

For better synchronization, special type of ICMP message called Time stamp request & reply is used.

Network Mask Request and Reply

- ① N/W mask is used in routers.
- ② N/W Administrator is responsible for setting up the router in such a way that the n/w mask is present in the router.
- ③ Every host can send ICMP packet to the router to get the subnet mask.

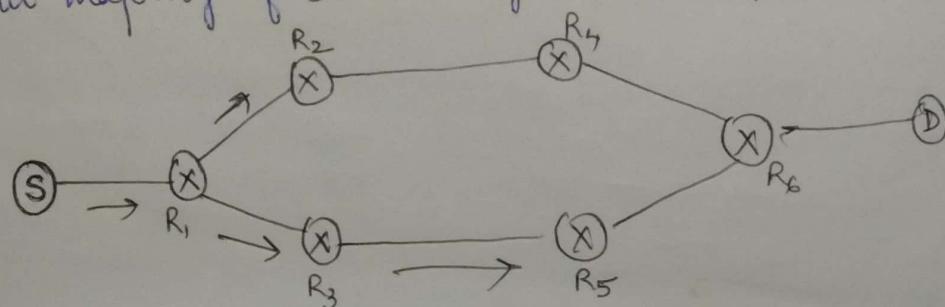
Traceroute - Application of ICMP

Traceroute provides a map of how data on the internet travels from the source to the destination.

Procedure -

- ① Send a packet with TTL=1. TTL = 0 at first router and it sends back IP of the router.
- ② Repeat the process with TTL=2, 3, 4 ...
- ③ Destination doesn't send ICMP packet when TTL=0 because packet isn't discarded.
 \therefore A dummy port no. that doesn't exist at the destination is used.
 In this case, destination ~~sends~~ sends ICMP packet of type Destination Port Unreachable.
- ④ The process is stopped when ICMP(DPU) pack is received.

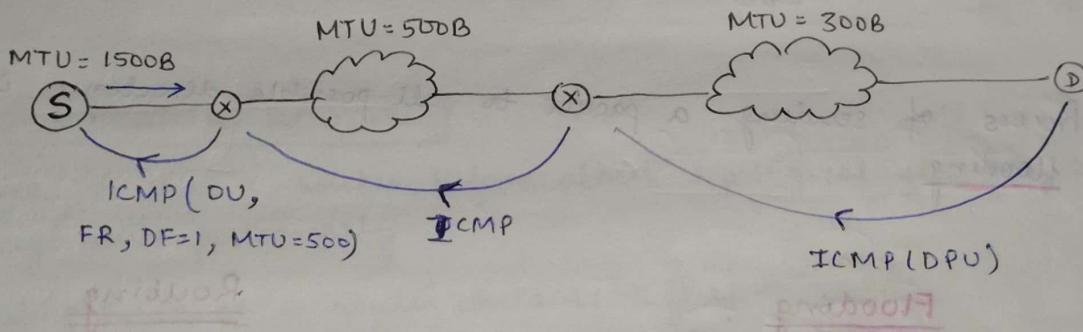
→ Traceroute might not ~~be~~ give the actual path.
 But in majority of cases we get actual path.



Traceroute = ~~R₁, R₂, R₅, R₆~~ = not valid path
bcz packet can move in any direction independently.

PMTUD - Path MTU Discovery - Application of ICMP

Finding ~~maximum~~ minimum size of datagram that can be sent from source to destination such that no fragmentation takes place.



Sender sends a packet of size 1500B

It receives ICMP (DU, FR, DF=1, MTU = 500)

Sender sends a packet of size 500B

It receives ICMP (DU, FR, DF=1, MTU = 300)

Sender sends a packet of size 300B

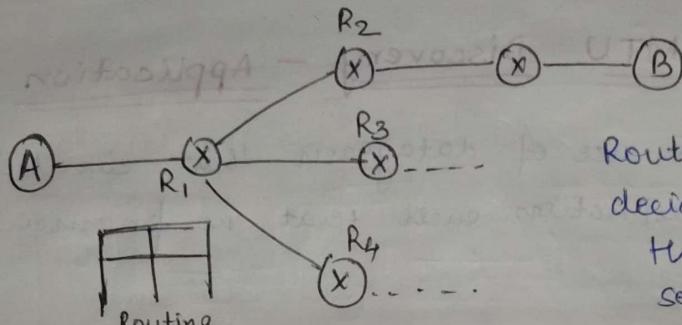
It receives ICMP (DPDU)

because dummy port no. is used.

ROUTING

at every router

- ① Process of preparing a routing table is called routing.



Routing table is used to decide to which router the packet should be sent next.

- ② Process of sending a packet to all possible directions is called flooding.

Flooding

- ① No routing tables are required
- ② Shortest path is guaranteed
- ③ Highly reliable

Disadv

- ④ Duplicate packets at destination
- ⑤ More traffic

Routing

- ① Routing table is required
- ② Shortest path is not guaranteed
- ③ Less reliable

Adv

- ④ No duplicate packet at destination
- ⑤ Less traffic

Routing algorithms

Static

→ Manually prepare and upload routing table offline.

→ If topology changes (link broken or new link added), or traffic changes (high traffic at a router), then, manual intervention is required.

∴ Not used

Dynamic

Topology changes & traffic changes are handled automatically by the algorithm.

Distance Vector Routing (DVR)

Link State Routing (LSR)

DISTANCE VECTOR ROUTING

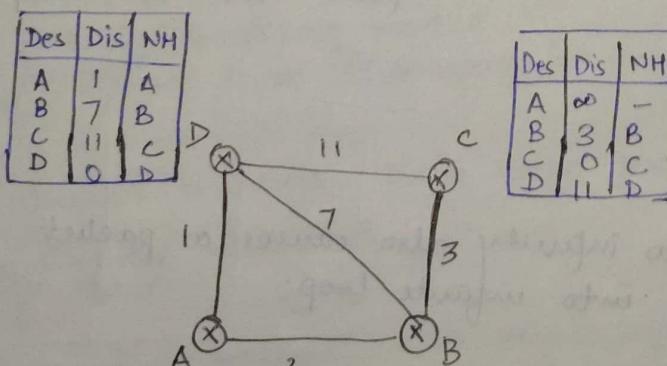
DVR requires that a router inform about topological changes to its neighbours periodically.

The routing table at a router contains 3 fields—

- 1) Destination
- 2) Distance
- 3) Next Hop.

Algorithm—

- ① Router transmits its distance vector to each of its neighbours.
- ② Each router receives & saves the most recently received distance vector from its neighbours.
- ③ Router then recalculates its distance vector.



Des	Dis	NH
A	0	A
B	2	B
C	∞	-
D	1	D

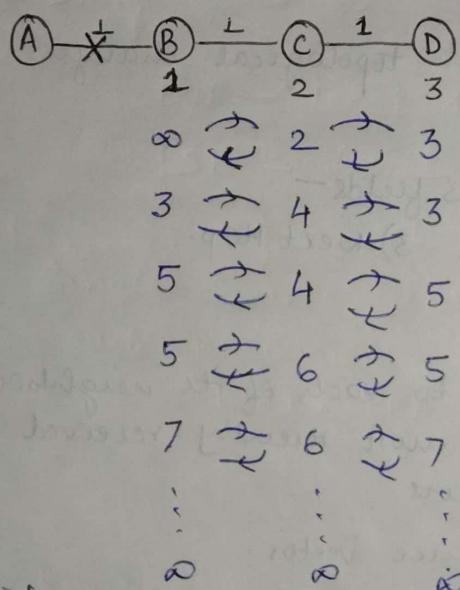
Des	Dis	NH
A	2	A
B	0	B
C	3	C
D	7	D

- Each router sends the distance vector to neighbouring routers.
 - Routing table is again designed by using the latest data.
- Distance = minimum of the values from neighbours

Bellman Ford Algorithm

Count to infinity - problem of DVR

If a link is down, then, it takes a lot of time for the routers to update their routing tables.



(distances to A)

Link b/w A & B is

down suddenly

The distances gradually tend to infinity.

Reason for the above problem -
Only distance vector is sent exchanged b/w the routers.
(not the next hop).

Count to infinity also causes a packet to fall into infinite loop.

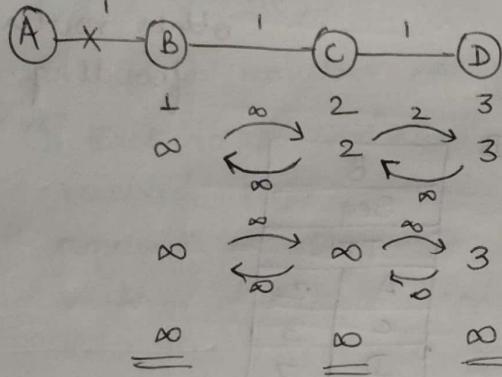
	A	B	C	D
A	0	1	2	3
B	1	0	1	3
C	2	1	0	2
D	3	3	2	0

	A	B	C	D
A	0	1	2	3
B	1	0	1	3
C	2	1	0	2
D	3	3	2	0

Split Horizon - Solution to Count to Infinity Problem.

Solution to count to infinity is called 'Split Horizon'

While sending distance vector from A to B, distances in A which depend upon B (ie. with next hop as B) are set to ∞ .

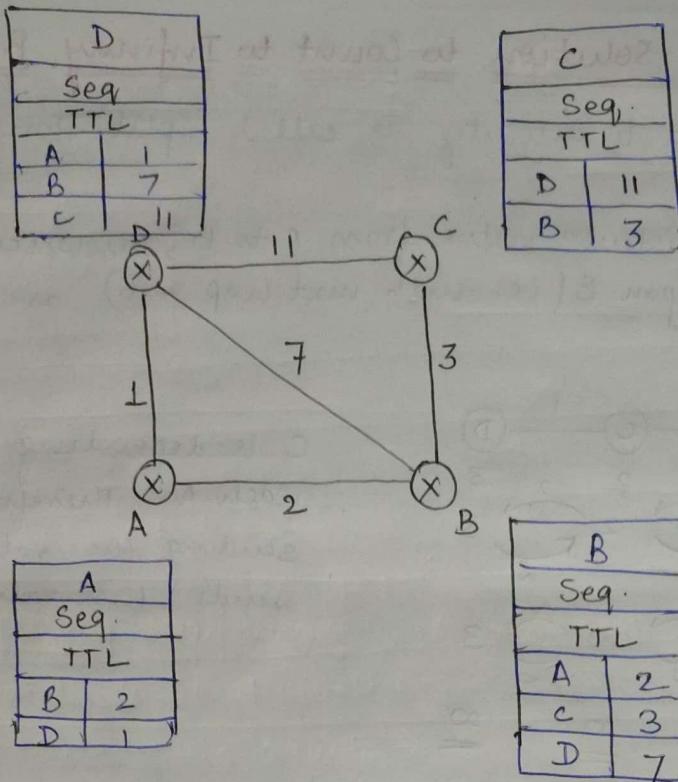


C is depending upon B to reach A. Therefore, instead of sending the value 2, it sends the value ∞ .

- ① By using split horizon, convergence is fast, no loops are formed.
- ② In case of only DVR, convergence is slow and loops will occur.

LINK STATE ROUTING (LSR)

- ① Every router creates something called Link State Packet
- ② Link state packet contains -
 - Neighbouring nodes
 - Distance to the neighbouring nodes.
- ③ Every router receives link state packet from every other router in the network.
- ④ The link state packets are sent by flooding.
- ⑤ By using the packet info in the packets received, each router creates its routing table using DJIKSTRA ALGORITHM



The Link state packets are sent to every other router by flooding.

* Every node has global database

DVR - local database (depends only upon info of neighbors)

LSR - global database (depends upon info of all routers).

Single source shortest Path algorithm is applied at each node

DIJKSTRA ALGORITHM **LINK STATE ROUTING**

∴ Routing table at A

Routing table at B

Des	Dis	NH
A	0	A
B	2	B
C	5	B
D	1	D

Des	Dis	NH
A	2	A
B	0	B
C	3	C
D	3	A

* LSR converges faster than DVR

~~LSR~~ - Problems

① Heavy traffic due to flooding.

⇒ Sequence number is present on each link state ~~Packet~~.

Each router stores the seq. no. of the latest packet received from a particular host. If a packet with sequence no. less than that already received reaches a router, (due to a different path which takes longer time), then, the packet is discarded.

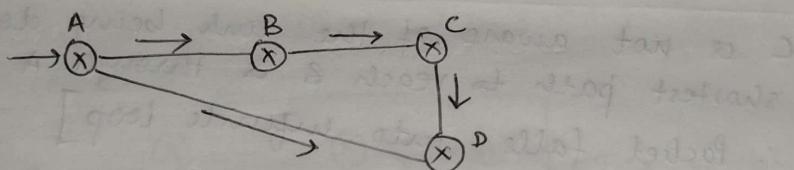
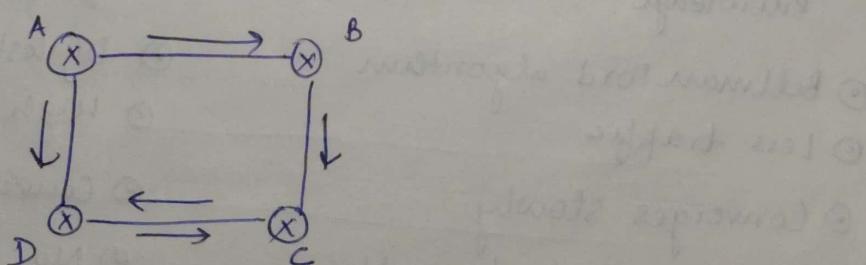


Table present at A -

Router	Latest	
B	15	(B, 8) - Discarded ✗
C	20	(B, 15) - Accepted ✓
D	30	(B, 11) - Discarded ✗

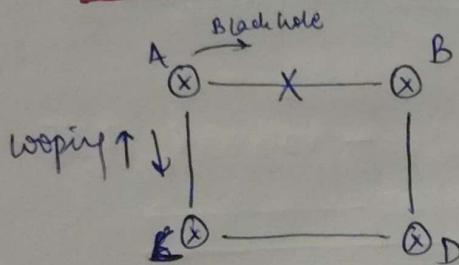
* Flooding can be controlled by using sequence numbers. If old packet is received at a router, it is not transmitted further.

⇒ Time to live field is used to prevent a packet from falling into infinite loop.



⇒ Sequence bits can get corrupted; therefore To solve this problem, validity / lifetime column is used. After a fixed amt. of time, the entry is discarded & the currently travelling packet seq. no is accepted.

Other problems with LSR



LSR has 2 persistent & transient problems (for short time)

These problems are present only for a short amount of time and get rectified automatically.

1) Black hole [If A-B link is down suddenly, it takes timeout time to discover that link is down. Upto that time, packets sent from A to B will fall into black hole]

2) Looping [After timeout, A redesigns its routing table and finds that shortest path to reach B is via C→D, B. It sends the packet to C.

C is not aware of the link being down. As to C, shortest path to reach B is through A.
∴ Packet falls into infinite loop]

Differences b/w DVR & LSR -

Distance Vector Routing

- ① Used in 1980s
- ② BW required is less
(no flooding, vector is sent only to neighbours)
- ③ Every router has local knowledge
- ④ Bellman Ford algorithm
- ⑤ Less traffic
- ⑥ Converges slowly
- ⑦ Count to infinity problem
- ⑧ Persistent looping
- ⑨ Implemented by using RIP protocol.

Link State Routing

- ① used in 1990s
- ② BW required is more
(flooding, sent to every router)
- ③ Every router has global knowledge
- ④ Dijkstra algorithm.
- ⑤ High traffic.
- ⑥ Converges faster
- ⑦ No count to infinity problem
- ⑧ Transient loop
- ⑨ Implemented by using OSPF

RIP (Routing Information Protocol)

- ① RIP is the implementation of DVR → simple to implement
- ② Metric = Hopcount (weights)
- ③ Infinity is represented by number 16 (i.e. if 16 hops are there, infinity is reached).

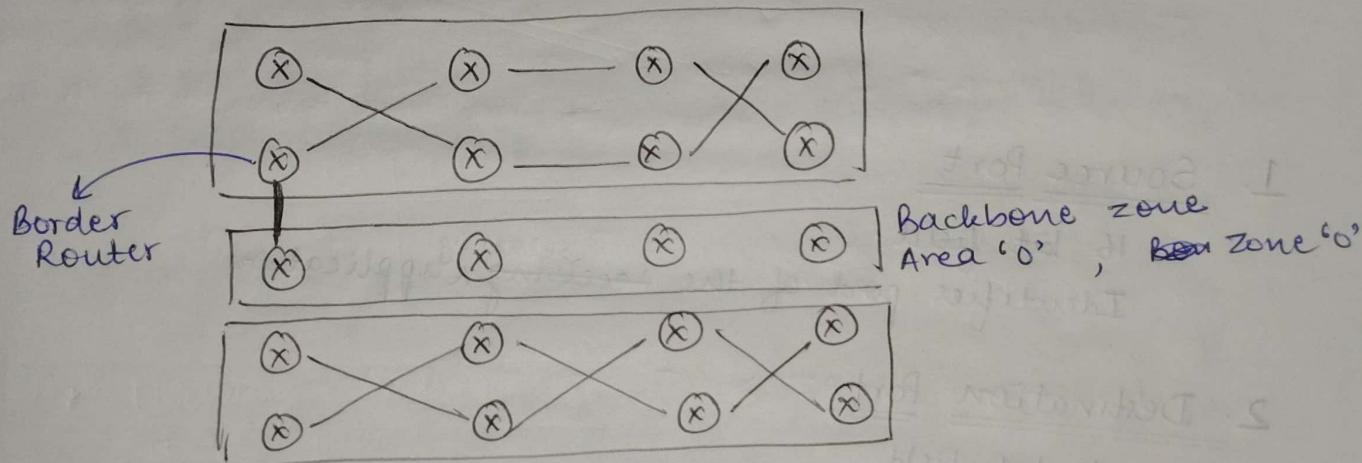
DSPF

Implementation of LSR → computation is complex.

DSPF divides all the routers into some regions and flooding is restricted in a particular region.

There is a router designated as 'Border Router' which receives all the flooded information, prepares summary.

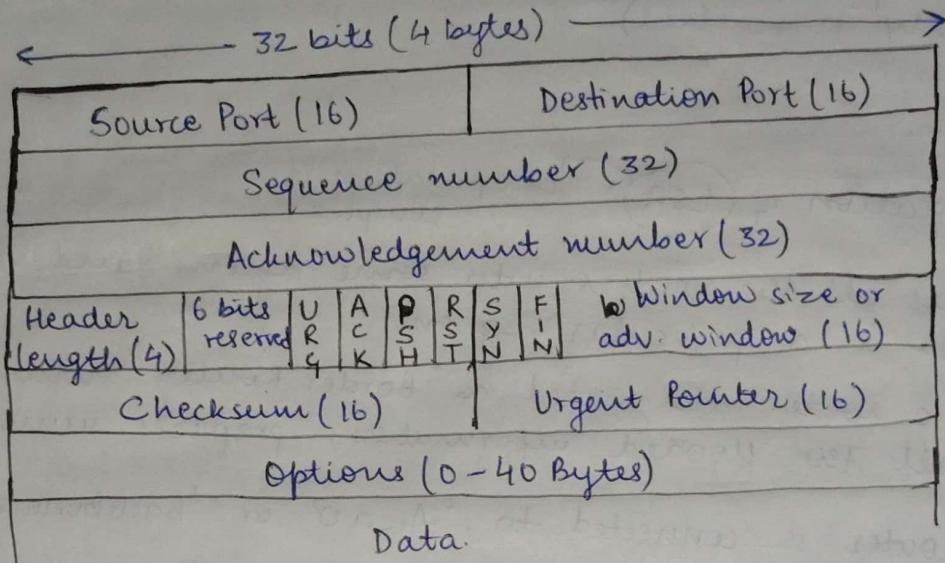
Border Router is connected to 'Area 0' or 'Backbone zone'



EIGRP protocol = RIP + OSPF

Transmission Control Protocol (TCP)

TCP header -



1. Source Port

16 bit field.

Identifies port of the ^{sending} application

2. Destination Port

16 bit field

Identifies port of the receiving application

$0-2^{16}-1$ port numbers can
be represented
(0-65535)

Http: 80

ftp: 21

Telnet: 23

SMTP: 25

0-1023 → well known
services
port no.

1024-49151 - Reserve

49152-65535 - General
public

Port no + IP address = identifies connection uniquely

TCP is connection oriented protocol (Resources are reserved)

Socket = Port no. + IP 48 bit no which identifies process
(16bit) (32bit)

3. Sequence Number

- ① 32 bit field (randomly chosen)
- ② TCP assigns a unique sequence number to each byte of data contained in the TCP segment.
- ③ This field contains the sequence number of first data byte

4. Acknowledgement Number

- ① 32 bit field.
- ② contains sequence number of the data byte that receiver expects to receive next from the sender.
- ③ Acknowledgement number = Sequence number of last received data byte + 1.

5. Header length

- ① 4 bit field.
- ② scaling factor = 4.
- ③ helps in knowing from where the actual data begins.

Length of TCP header
always lies in the range 20 - 60 bytes

Header length field
can contain values
in the range 5 - 15

Wrap Around Time

The time is the time required after which a particular sequence number will be used again.

∴ there are 32 bits in sequence no. field

∴ No. of sequence numbers possible = 2^{32}

Wrap Around Time (WAT) depends upon bandwidth.

Example - Bandwidth = 1 MBps

$$\Rightarrow 1 \text{ sec} \rightarrow 1 \text{ MB} \Rightarrow 10^6 \text{ bytes} \rightarrow 1 \text{ sec}$$

$$\Rightarrow 10^6 \text{ seq. no.} \rightarrow 1 \text{ sec} \Rightarrow 1 \text{ seq. no.} \rightarrow \frac{1}{10^6} \text{ sec}$$

$$\therefore 2^{32} \text{ seq. nos.} = \frac{1}{10^6} \times 2^{32} \text{ sec} = 4296.967 \text{ sec}$$

In today's internet, there is a concept of lifetime
 Lifetime → Maximum amount of time for which the packet is alive (or the time taken to reach the destination in worst case)

$$\boxed{\text{Lifetime} = 3 \text{ min} = 180 \text{ sec}}$$

As long as WAT > LT, no problem.

If WAT < LT, problem arises due to same sequence nos.

Solutions -

↳ Decreasing bandwidth (but not practical because we would DIE for bandwidth. Right?)

↳ Increasing the number of sequence numbers
 [This is done by utilizing options field]

Example- Bandwidth = 1GB

— 1 sec — 1GB \Rightarrow 1 sec \rightarrow 1G sequence nos.

\Rightarrow 180 sec \rightarrow $(180 \times 1G)$ seq. nos. required.

No. of bits in sequence number field = $\lceil \log_2(180 \times 1G) \rceil \approx 42$

∴ No. of additional bits required for seq. no. = $42 - 32 = 10$

These bits are used
in options and are
called time stamp.



No. of bits in seq no.
field required to
prevent the problem
of wrap around time
 $= \lceil \log_2(BW \times LT) \rceil$

Establishing TCP connection

Three Way Handshaking

[SYN] \rightarrow S- geht

Step 1:- [SYN]

For establishing a connection,

- ① Client sends a request segment ~~for~~ to the server.
- ② Request segment only consists of TCP header with an empty payload.
- ③ Then, it waits for the reply segment from the server

Request segment consists of

- Initial sequence number
- SYN bit set to 1
- Maximum segment size (MSS) of the network.
- Receiving window size

replies etc

Step-2 :- [SYN + ACK]

After receiving the request segment,

- ① Server responds to the client by sending the reply segment.
- ② It informs the client of the parameters at the server side

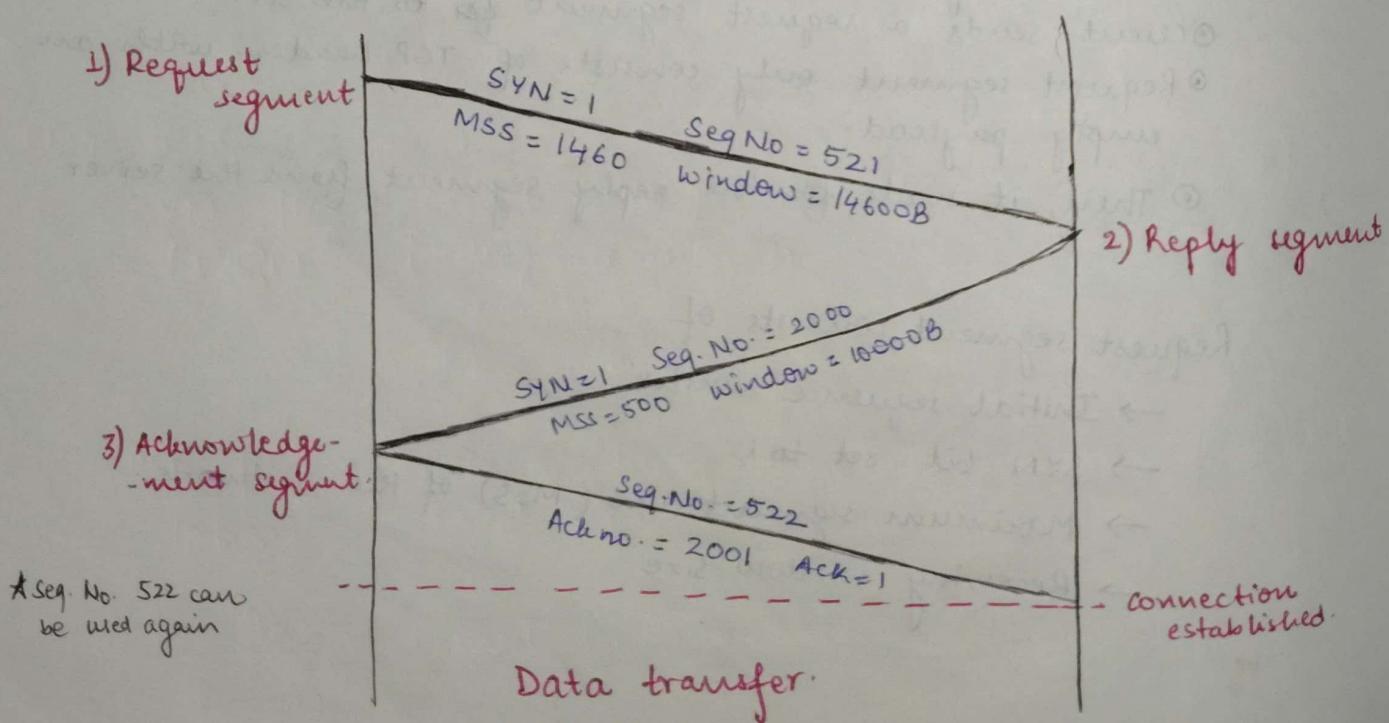
Reply segment consists of the following information -

- Initial sequence number
- SYN bit set to 1
- Maximum segment size (MSS) of the n/w.
- Receiving window size
- Acknowledgement number
- ACK bit set to 1.

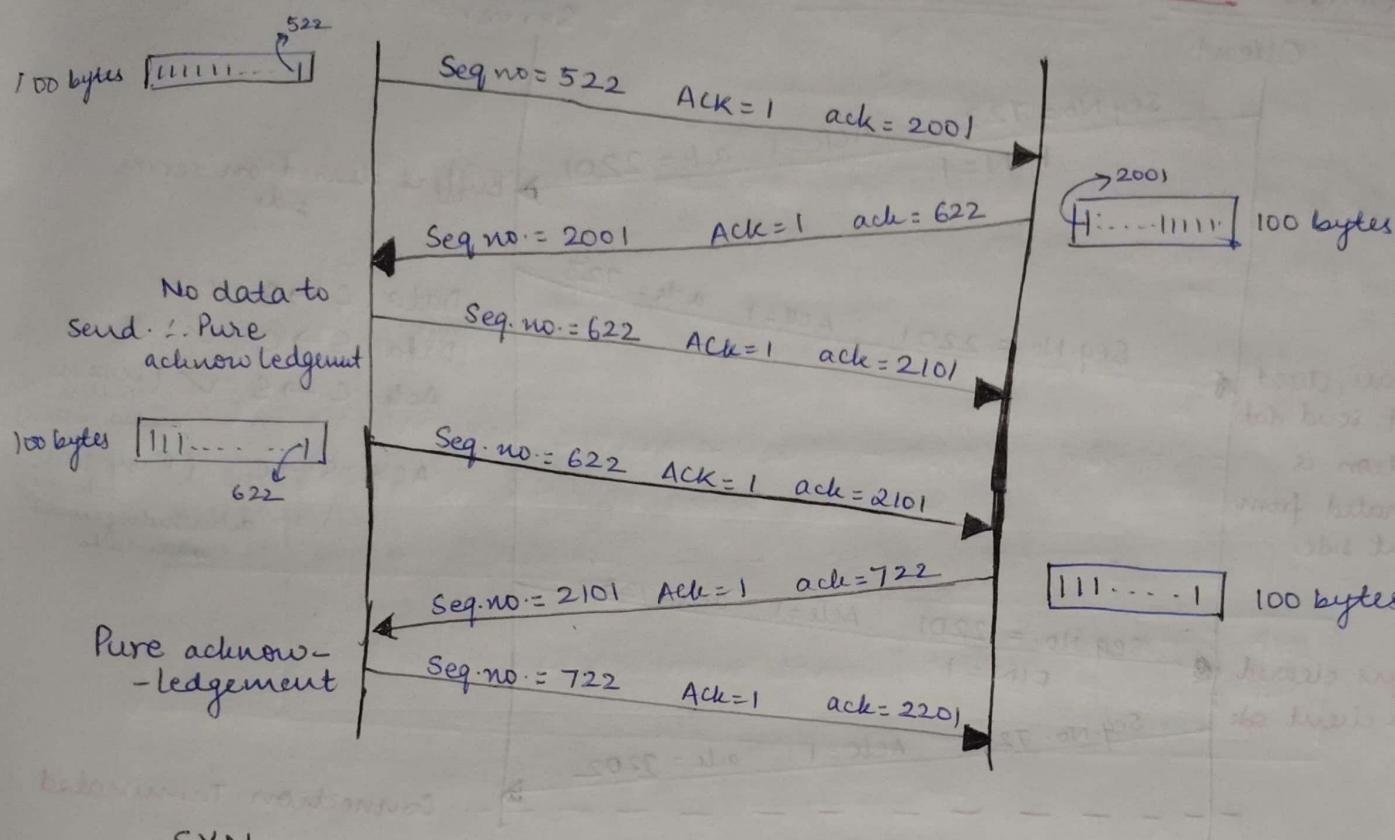
Step-3 :- [ACK]

After receiving the reply segment,

- ① Client acknowledges the response of the server.
- ② It acknowledges the server by sending a pure acknowledgement.



DATA TRANSFER AFTER CONNECTION ESTABLISHMENT



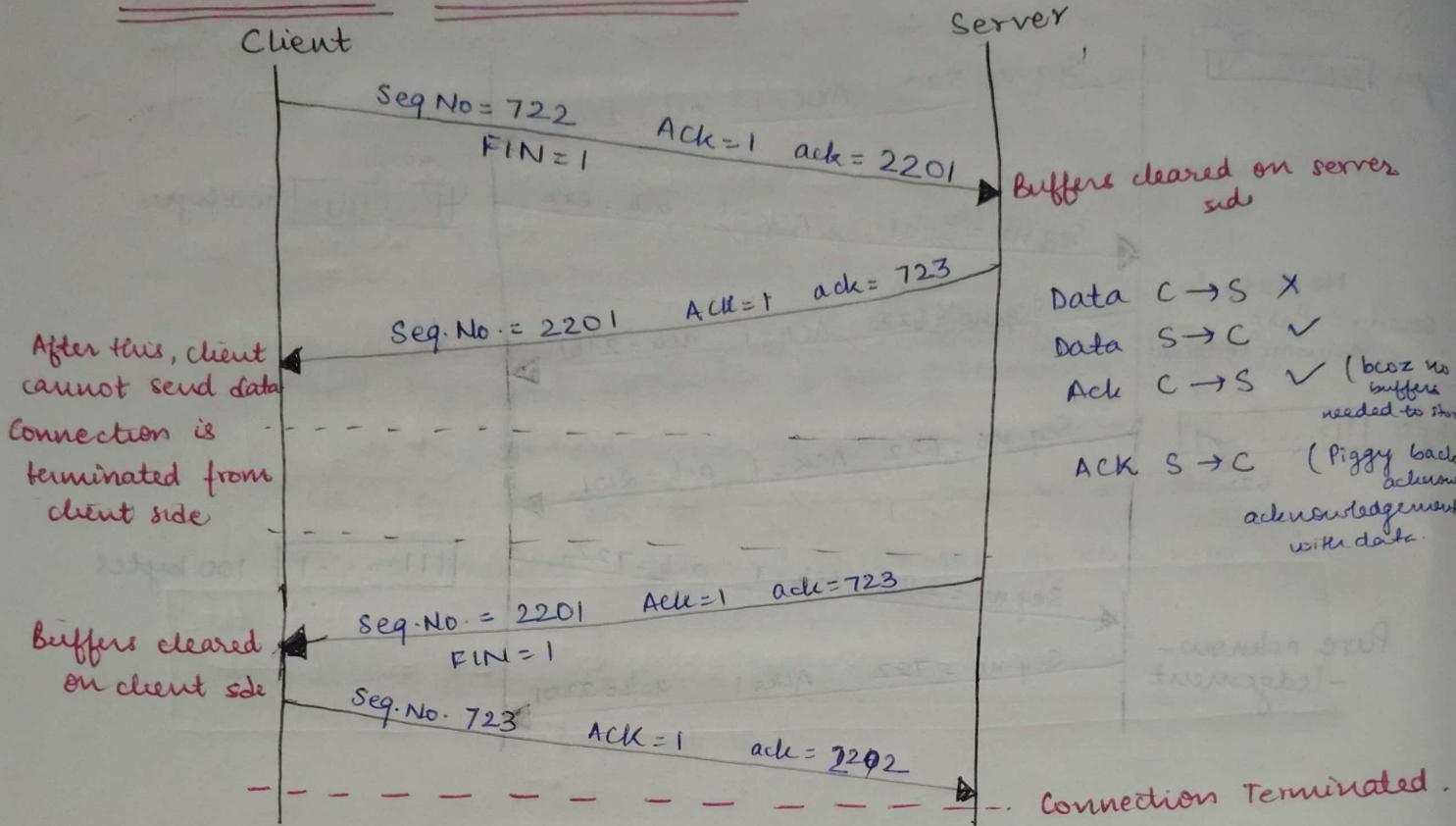
<u>SYN</u>	<u>ACK</u>
1	0
1	1
0	1
0	0

- 1st segment (Request segment)
- 2nd segment (Reply segment)
- Pure acknowledgement
- Not possible

① Sender: If there is no data to send immediately, then, instead of waiting, pure acknowledgement is sent.

No sequence number is consumed for pure acknowledgement.

CONNECTION TERMINATION



(Answer for question) Question no. 1

- ① 3 segments sent for data connection establishment.
- ② Data transfer
- ③ 4 segments (max) or 2 segments (min) for connection termination.

Flags used

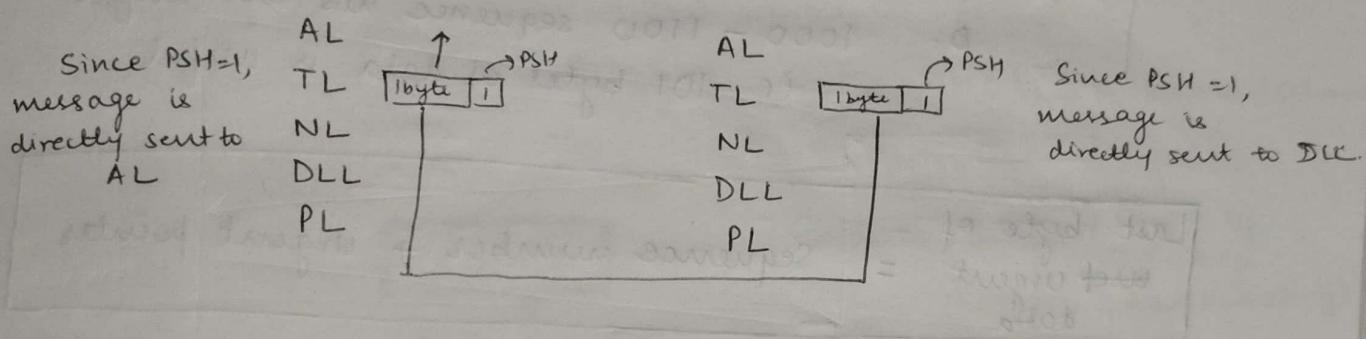
SYN flag :- Synchronizing sequence numbers
 (1 bit)
 [telling client/server which sequence no. is randomly chosen]
 takes 1 bit sequence number.

ACK flag :- Tells whether acknowledgement number field is valid or not.
 (0.5 bit)

FIN flag :- Request for connection termination
 (1 bit)

PSH (Push flag)

- Generally, TL takes data from AL ~~one~~^{some} bytes at a time and waits for more data until the size of data reaches MCS.
- PSH flag is used to tell TCP not to buffer the data and directly push the data (though of small size) to the DLL.
- Interactive applications [chat, Telnet, SSH, PuTTY, Rlogin] etc. have PSH=1.



URG (Urgent Flag)

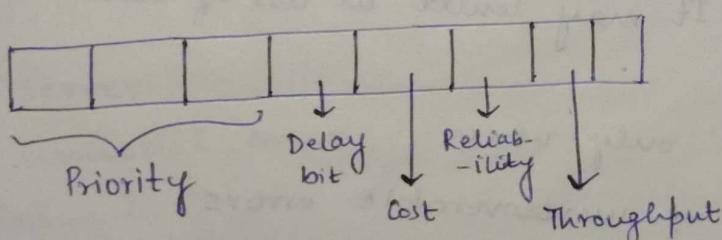
- used to tell that the given segment is urgent and should be executed first.
 - On the receiver side, a segment with URG flag set to 1 is processed first.
- Thus, this flag helps in out of order processing of data.

Routers do not have TL ∴ they cannot see the URG flag.

∴ Type of service field (IP datagram) is used.

8 bits.

For urgent ~~bit~~ flag,
priority bits = 111 (7)



URGENT POINTER (16 bits)

In a TCP segment, if some part of data is important, urgent pointer is used.

Urgent pointer tells the last seq sequence number upto which data is important.

If seq.no. = 1000, urgent pointer = 100. Then,

- o 1000 - 1100 sequence nos. are useful.
i.e. 101 bytes of data is useful.

$$\text{Last byte of } \cancel{\text{useful}} \text{ urgent data} = \text{Sequence number} + \text{urgent pointer}$$

RST flag (Reset)

used to reset the TCP connection

- o When RST bit is set to 1,
- o It indicates the receiver to terminate the connection immediately.
- o It causes both the sides to release the connection & all its resources abnormally.
- o The transfer of data ceases in both the directions.
- o It may result in loss of data that is in transit.

used only when

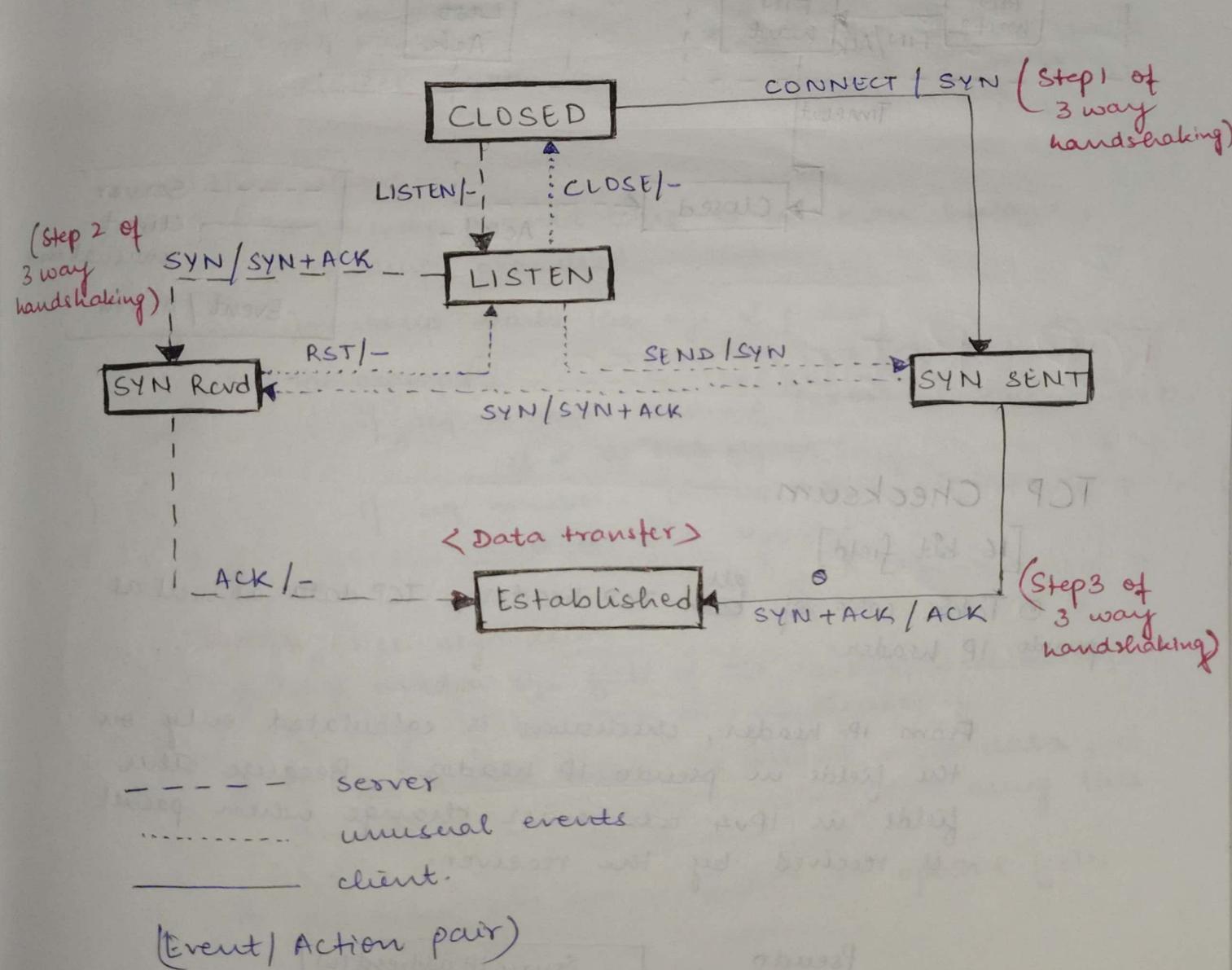
→ unrecoverable errors

→ no chance of terminating the TCP connection normally.

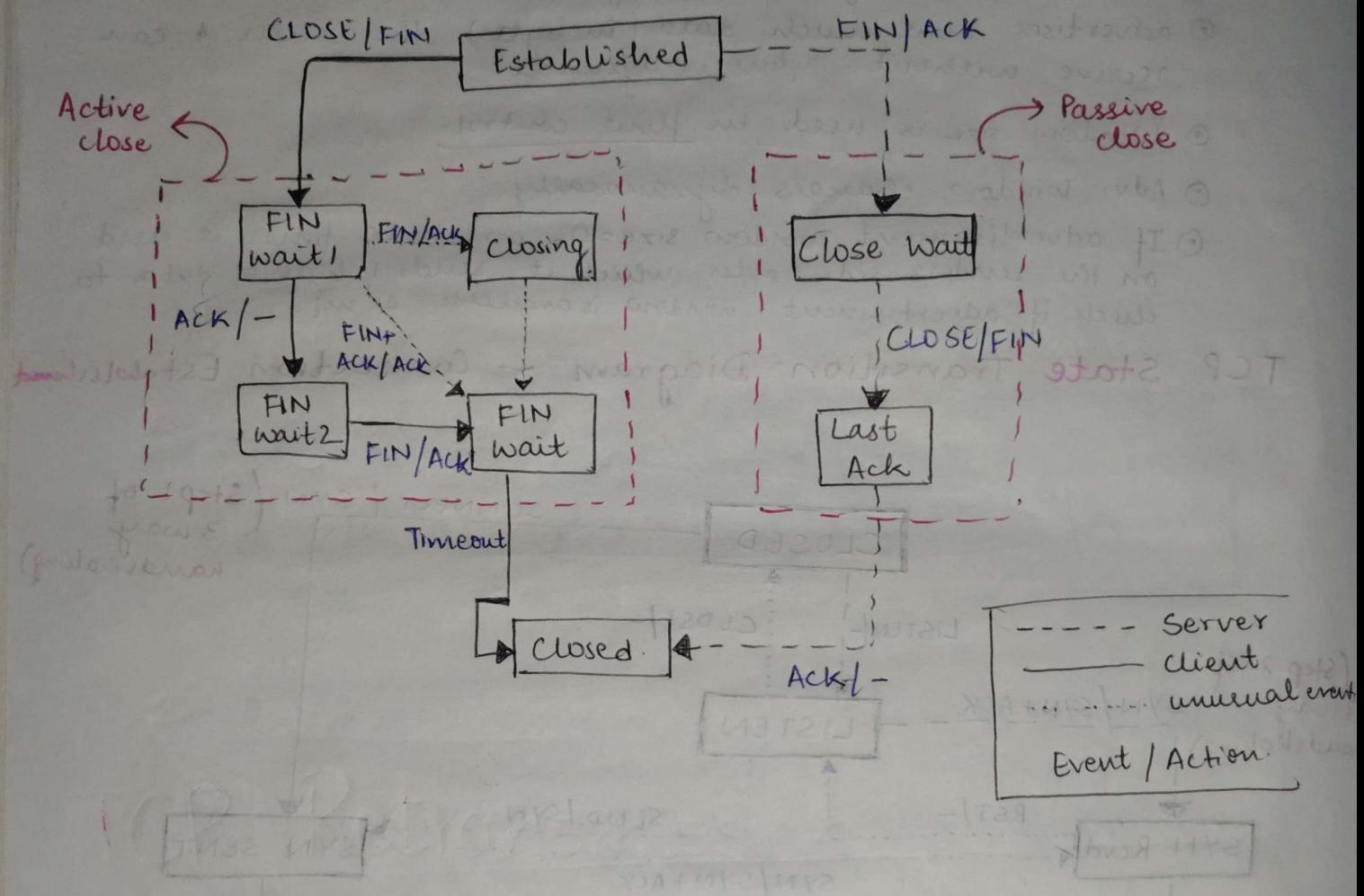
Window size

- ① 16 bit field.
- ② contains the size of the receiving window of the sender.
- ③ advertises how much data (in bytes) the sender can receive without acknowledgement.
- ④ Window size is used in flow control.
- ⑤ Adv. window changes dynamically.
- ⑥ If advertisement window size=0, persistent timer is used on the sender's side after which it sends 1 byte of data to check if advertisement window is available or not.

TCP State Transition Diagram — Connection Established



TCP State Transition Diagram - Connection Release.



TCP Checksum

[16 bit field]

<ref> network 0703 </ref>

- Takes care of all ~~TCP header~~ TCP header, TCP data as well as pseudo IP header.

From IP header, checksum is calculated only on the fields in pseudo IP header. Because other fields in IPv4 ~~are~~ header change when packet is received by the receiver.

Pseudo
IP
Header.

Source IP address(16)		
Destination IP address(16)		
00000000	Protocol (8)	TCP segment length (16)
(B)		

Why IP header check is included in calculated calculating TCP checksum?

In order to have double checking.

To make sure that the message always reaches correct destination.

Options

0 - 40 bytes.

Options field is used for following purposes -

1. Time stamp
2. Window size extension
3. Parameter negotiation
4. Padding.

Time stamp -

When wrap around time is less than lifetime, timestamps are used.

Timestamp marks the age of packet.

For example,

if seq. number = 000...0 (32 bits) & time stamp = 0,
it is the first segment.

if seq. number = 000...0 (32 bits) & time stamp = 1,
it is 2^{32} th segment.

Window size negotiation

① Using window size field of TCP header, window size of only 16 bits can be represented.

② If the receiver wants to receive more data, it can advertise its greater window size using this field.

③ The extra bits are appended in options field

Parameter negotiation

During connection establishment, both sender and receiver have to specify their maximum segment size (MSS).

There is no such field for MSS.

∴ It is specified in options field.

Padding

Addition of dummy data to fill up unused space in the transmission unit and make it conform to the standard size is called padding.

Header length field should be multiple of 4.

TCP Retransmission

① Retransmitting of a segment that is lost before reaching the receiver is called retransmission.

② When sender discovers that the segment sent by it is lost, it retransmits the same segment to the receiver.

TCP segment is lost when -

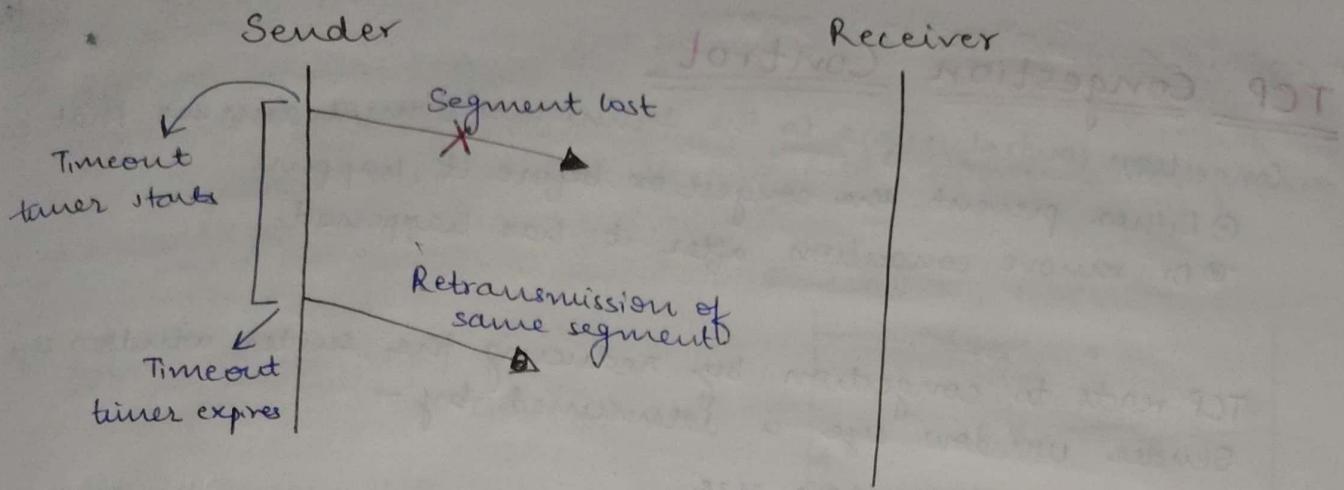
Either Time Out Timer expires

Or it receives 3 duplicate acknowledgements.

Retransmission after timeout expiry -

Each time sender transmits a TCP segment to the receiver, it starts Time Out Timer.

If sender does not receive acknowledgement for the sent segment before the timer goes off, it retransmits the same segment to the receiver & resets the timer.



Retransmission after receiving 3 duplicate acknowledgements.
sender assumes that a packet was been lost if it gets
3 duplicate acknowledgements for a TCP segment sent by
it.

This is known as Early retransmission or Fast Retransmission.

Sender sends 5 TCP segments.

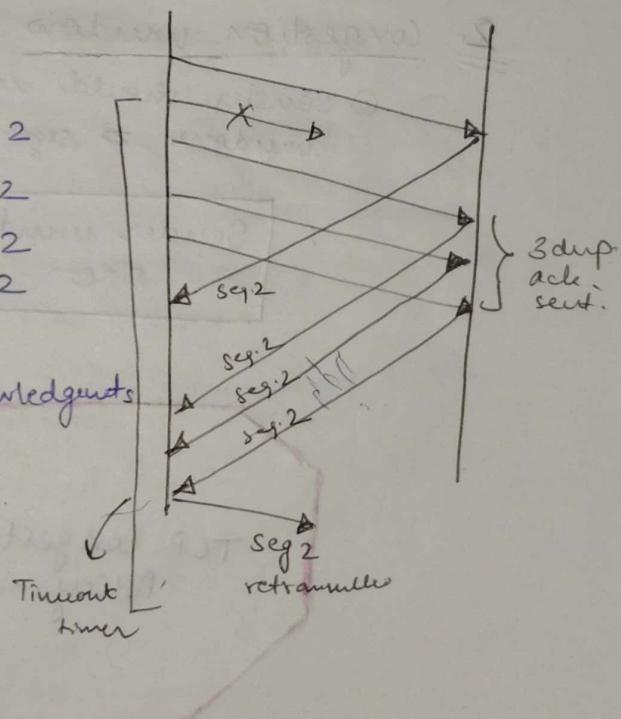
2nd segment gets lost.

Segment 1 sends ack for segment 2

Segment 3 sends ack for " 2

Segment 4 " " " 2

Segment 5 " " " 2



Sender receives 3 duplicate acknowledgments
in total.

∴ it assumes segment 2 is lost.

TCP Congestion Control

Congestion control refers to the techniques & mechanisms that can either prevent the congestion before it happens or remove congestion after it has happened.

TCP reacts to congestion by reducing the sender window size.
Sender window size is determined by -
→ Receiver window size
→ Congestion window size

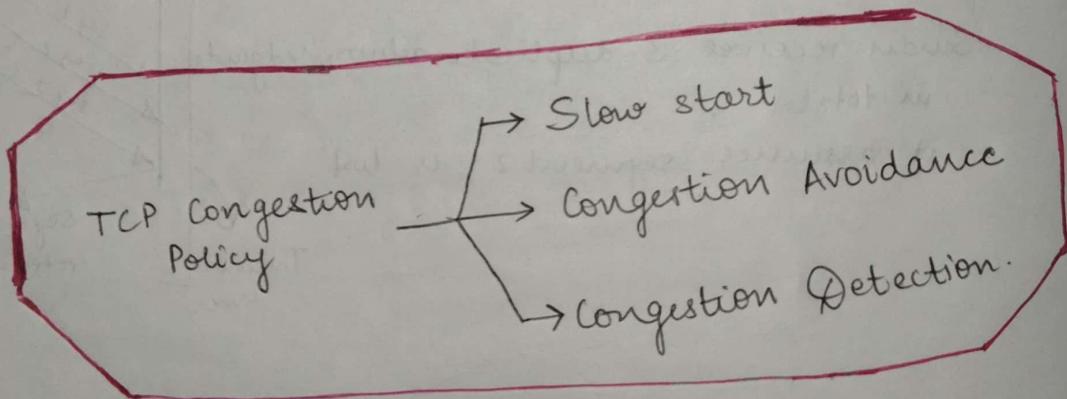
1. Receiver window size -

- ① Advertisement window.
- ② Sender should not send data greater than receiver window size.
- ③ Receiver dictates its window size through TCP header.

2. Congestion window -

- ④ Sender should not send data greater than congestion window size.

$$\therefore \text{Sender window size} = \text{Minimum}(\text{Receiver window size}, \text{Congestion window size})$$



Slow Start phase

Initially, sender sets congestion window size = Maximum Segment Size (MSS).

The congestion window size is increased exponentially when acknowledgements are received.

After 1 round trip,

$$\text{congestion window size} = (2)^1 = 2 \text{ MSS}$$

After 2 round trip,

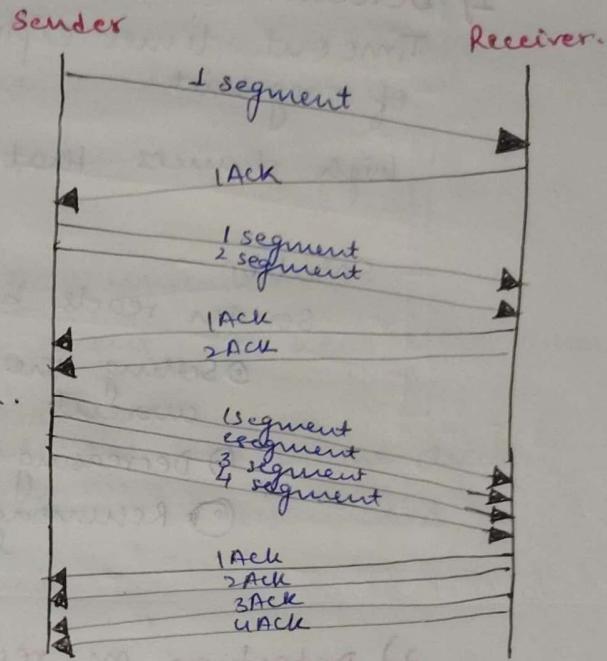
$$\text{congestion window size} = (2)^2 = 4 \text{ MSS}$$

After 3 round trips,

$$\text{congestion window size} = (2)^3 = 8 \text{ MSS} \dots$$

This phase continues until the congestion window size reaches slow start threshold.

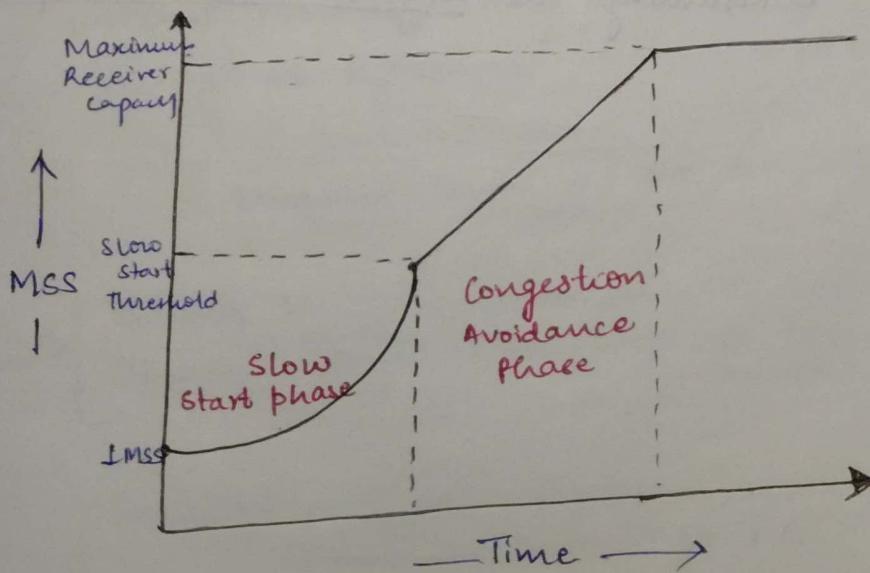
$$\text{Threshold} = \frac{\text{Receiver window size}}{\text{MSS} * 2}$$



Congestion Avoidance Phase

After reaching threshold,

- ① Sender increases congestion window size linearly to avoid the congestion.
- ② On receiving acknowledgement, sender increments the congestion window size by 1.



Congestion Detection phase

When sender detects the loss of segments, it reacts in different ways depending upon how the loss is detected.

1) Detection on Time out

Time out timer expires before receiving the acknowledgement of a segment.

high chances that it was due to congestion in n/w.

Solution -

Sender reacts by -

- ① Setting the slow start threshold to half the current window size.
- ② Decreasing CWS to 1MSS.
- ③ Resuming the slow start phase.

2) Detection on receiving 3 Duplicate Acknowledgements

- ① Sender receives 3 duplicate acknowledgements for a segment.
- ② weak possibility that it was due to congestion in the n/w

Solution -

Sender reacts by -

- ① Setting the slow start threshold to half the current congestion window size.
- ② Decreasing CWS to slow start threshold.
- ③ Resuming the congestion avoidance phase.

TCP Timer Management

(10)

1) Time wait timer -

- ① Time wait timer is used for connection termination.
- ② Time ~~wait~~ wait timer = $2 \times$ lifetime.
- ③ Started when sender sends the second FIN segment

2) Keep Alive Timer

- ① used to terminate long idle TCP connections.
- ② Each time server hears from client, ~~the~~ keep alive timer is reset.
- ③ After it expires, probe segments are sent to the client.
- ④ If no response is received, connection is terminated.

3) Acknowledgement timer

used for cumulative acknowledgement + piggyback acknowledgement.

4) Persistent timer

TCP uses persistent timer to deal with zero window size deadlock situation.

5) Timeout timer

Time for which sender should wait for acknowledgement without retransmitting the packet.
should be dynamic.

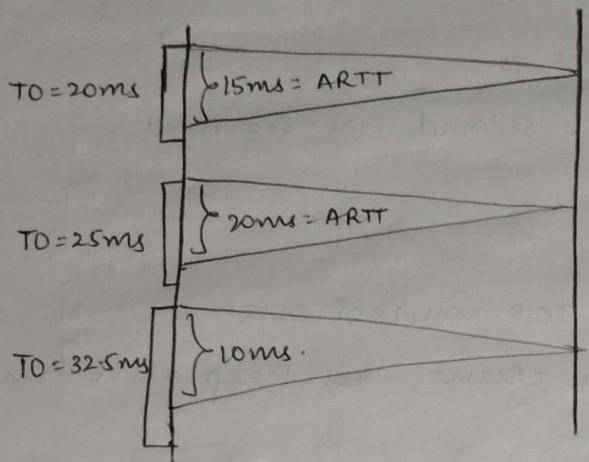
$$\text{Timeout timer} = 2 \times \text{Round Trip Time.}$$

Problem {
Timeout timer is too small \rightarrow unnecessary retransmission
Timeout timer is too large \rightarrow large waiting time if packet is lost.

Solutions \rightarrow Basic algorithm
 \hookrightarrow Jacobson's Algorithm

Basic Algorithm for Timeout Timer computation

① Timeout timer is calculated dynamically.



$$IRTT = 10 \text{ ms} \text{ (guessing)}$$

$$TO = 2 * IRTT = 20 \text{ ms}$$

$$ARTT = 15 \text{ ms}$$

$$NRTT = \alpha IRTT + (1-\alpha) ARTT$$

α = Smoothening factor
(will be given in que)

$$\begin{aligned} NRTT &= 0.5 \times 10 + 0.5 \times 15 \\ &\approx 12.5 \text{ ms} \end{aligned}$$

⇒ This algorithm is dynamic in nature.

⇒ Disadvantage - $TO = 2 * RTT$

No logic why 2 is used.

IRTT = Initial Round Trip Time

ARTT = Actual Round Trip Time

NRTT = Next Round Trip Time (expected)

For next packet,
IRTT = 12.5 ms

$$TO = 25 \text{ ms}$$

$$ARTT = 20 \text{ ms}$$

$$\begin{aligned} NRTT &= 0.5 \times 12.5 + 0.5 \times 20 \\ &= 16.25 \text{ ms} \end{aligned}$$

Jacobson's Algorithm for Timeout Timer Computation

Similar to basic.

$$TO = 4 * ID + IRTT$$

$$AD = |IRT - ARTT|$$

ID = Initial Deviation

AD = Actual Deviation

$$NRTT = \alpha IRTT + (1-\alpha) ARTT$$

$$ND = \alpha ID + (1-\alpha) AD$$

Karn's modification says that if we don't receive the acknowledgement of the packet within timeout, keep doubling the timeout timer for the next packets.

Need For UDP [User Datagram Protocol]

TCP is disadvantageous for some situations -

1) When the application needs just 1 request and 1 reply.

Ex:- DNS, BOOTP, DHCP, NTP (Network Time Protocol), NNP (Network News Protocol), TFTP, RIP, OSPF (Trivial File Transfer Protocol)

2) For broadcasting & multicasting, buffers are needed to be allocated in TCP for each host. This leads to inefficiency.

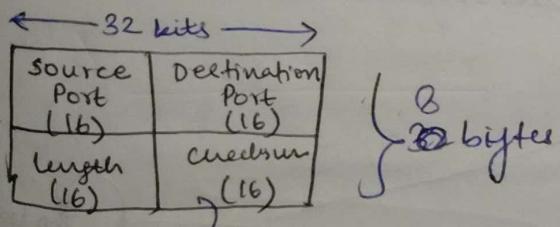
∴ UDP is used for broadcasting (multicasting).

3) When we require speed than reliability.

Ex → Watching a youtube video in HD format → TCP
[if 1 frame is lost, video pauses]

but when video is not in HD, some frames may be lost but video will not be clear. not pause. Also, quality of video may drop due to loss of frames.

UDP Header



computed on
UDP header,
data,
pseudo IP header.

8 bytes of header

① UDP should communicate the options to the IP Datagram. The various options are

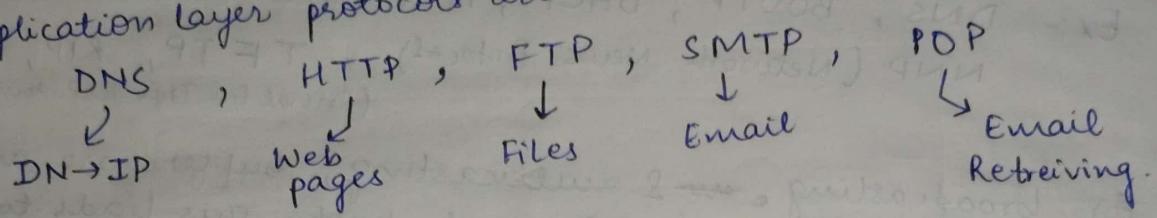
- 1) Trace Route
- 2) Record Route
- 3) Time stamp.

② Additional responsibility of UDP is to inform application layer about ICMP error packets.

Application Layer protocols

Application layer is responsible for all the services that internet provides.

Application layer protocols are



1) DNS (Domain Name Service)

(Port number : 53)

① DNS uses UDP at transport layer

② used to convert Domain name to IP address.

③ Types of domains —

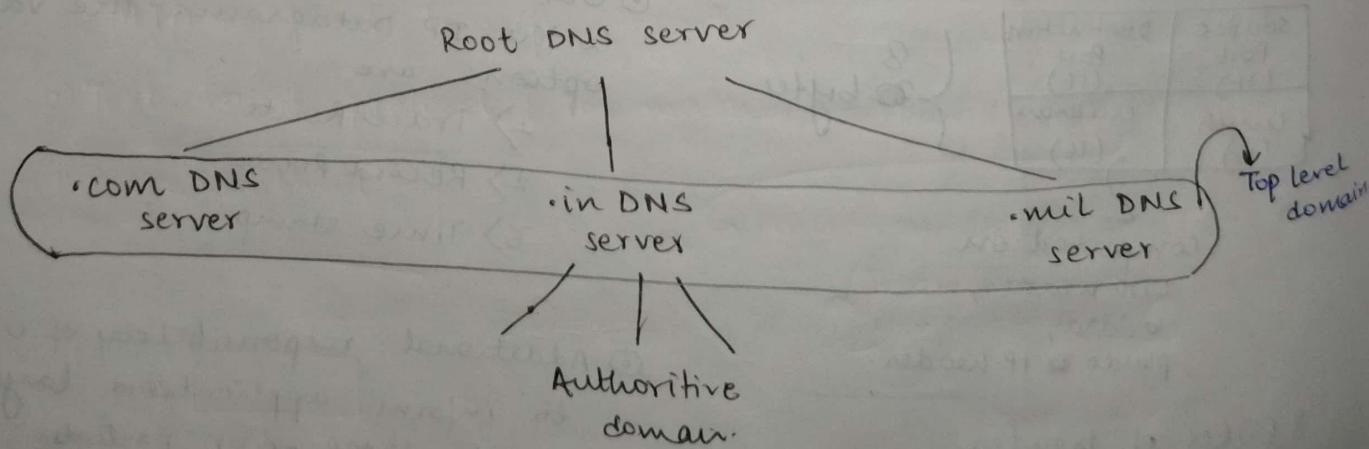
→ Generic domain (.com, .edu, .mil, .org, .net)

→ Country domain (.in, .us, .uk)

→ Inverse domain (Given IP address → Find domain name)

④ DNS is also used for load balancing

DNS database is organized in hierarchical manner.

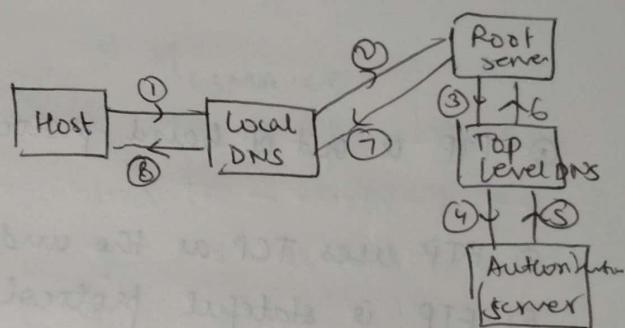
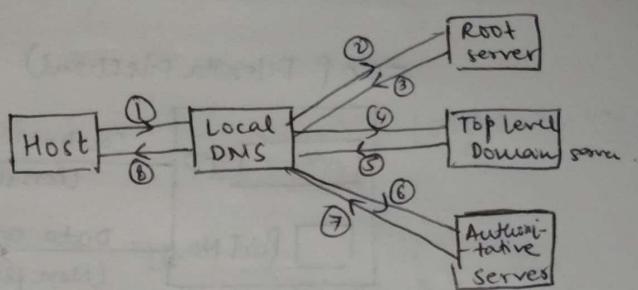


① Internet Engineers Task Force (IETF) manages 13 root servers 1105
across the world to deal with failures.

② Local DNS is used to prevent DNS overhead.
It stores freq. used domain names & IP addresses for a time duration.

Local DNS contacts Root server
in 2 ways -

- 1) Iterative way
- 2) Recursive way.



2) HTTP (Hyper Text Transfer Protocol)

- ① HTTP always needs reliability (it does not implement it by itself)
- ② HTTP uses TCP at transport layer.
- ③ HTTP is in band protocol (both commands & data go in 1 connection).
- ④ HTTP is stateless protocol (user activity is not tracked on server side)

2 versions — HTTP 1.0 (non persistent connection)

1 connection is used for each object

HTTP 2.1 (persistent connection)

1 connection for all objects

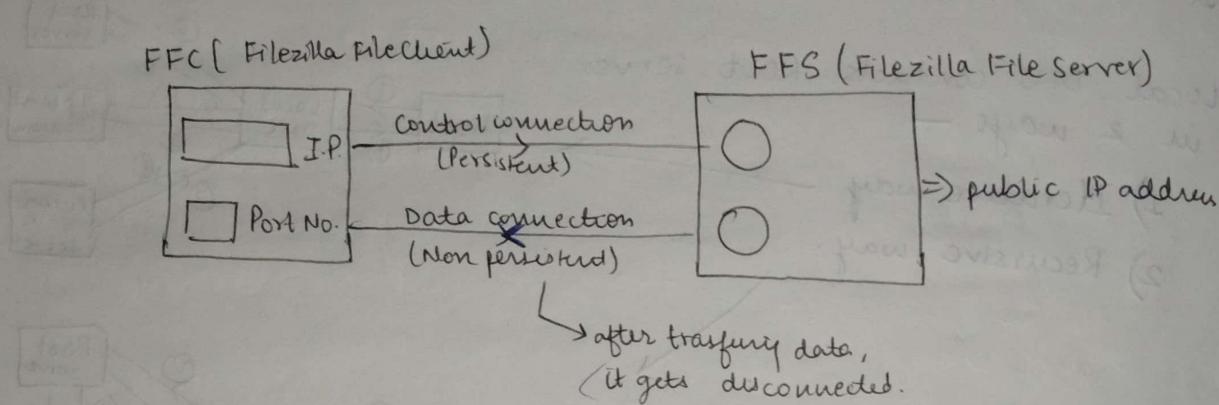
Methods supported by HTTP

1. Head → returns header (metadata) of a webpage
2. Get
3. Post } used with forms
4. Put → to upload an object.
5. Delete → to delete the object
6. Trace → Trace the servers which are sending data.
7. Options
8. Connect → used by https (for authentication)

3) File Transfer Protocol (FTP)

Port Number = 21

- ① used to transfer FTP.
- ② Tectia, Filezilla are popular FTPs.



- ④ FTP is out of band protocol (commands & data flow through different connections)
- ⑤ FTP uses TCP as the underlying protocol.
- ⑥ FTP is stateful protocol.

4) SMTP (Simple Mail Transfer Protocol)

- ① Emails are transferred using SMTP.

- ② FTP cannot be used for emails because to transfer file using FTP, both server and client should be online.

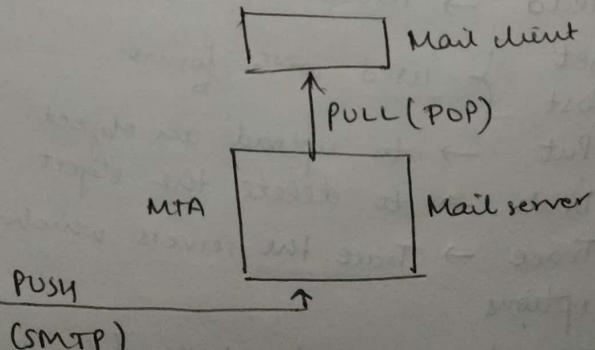
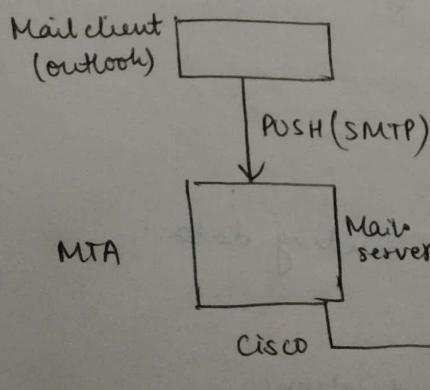
MTA = Mail Transfer Agent

- ③ Text based protocol (For multimedia transfer, MIME software used)

SMTP & POP are in band protocol

↳ Multipurpose Internet Mail Extension

{
Text → Non-text
Non-text → Text}



For any given situation, efficiency of Token Ring is always greater than CSMA/CD

$$\eta_{\text{CSMA/CD}} = \frac{1}{1+6.44A} \quad (N \rightarrow \infty)$$

$$\eta_{\text{TR-E}} = \frac{1}{1+\alpha/N}$$

$$\text{if } N \rightarrow \infty \quad \eta_{\text{TR-E}} = \frac{1}{N \rightarrow \infty} \frac{1}{1+\alpha/N} = 100\%$$

$$\eta_{\text{TR-D}} = \frac{1}{1+\alpha(1+1/N)}$$

$$\text{if } N \rightarrow \infty \quad \eta_{\text{TR-D}} = \frac{1}{N \rightarrow \infty} \frac{1}{1+\alpha(1+1/N)} = \frac{1}{1+\alpha}$$

No need
to solve numerical
if comparison is req'd

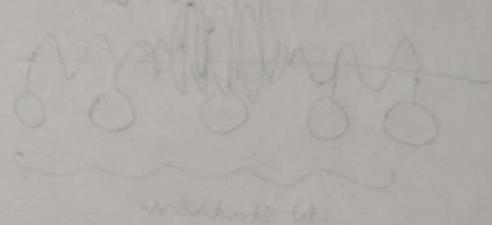
$$\eta_{\text{TR-E}} > \eta_{\text{TR-D}} > \eta_{\text{CSMA-CD}}$$

$$\frac{1}{1+(\frac{N+1}{N})\alpha}$$

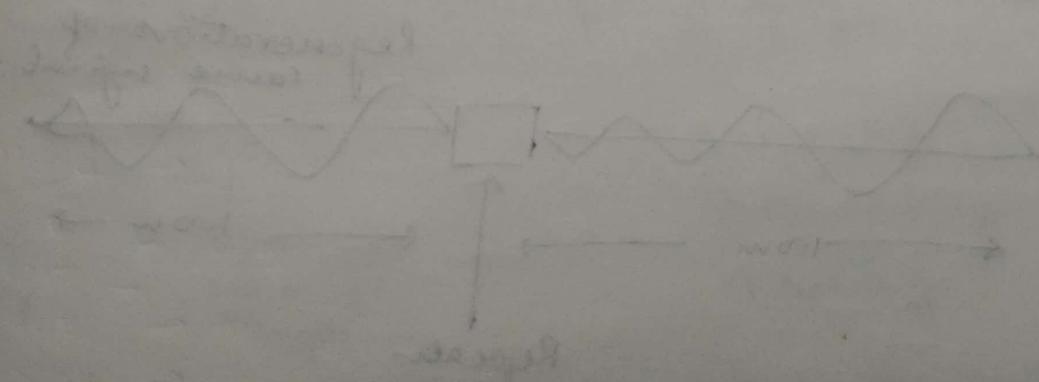
$$\frac{1}{1+\alpha/N}$$

$$\frac{1}{1+6.44\alpha}$$

Maximized utilization



Original idea was to prevent collisions at busy times. So idea is to divide the bus into segments. - algorithm of token passing between nodes. So when one node gets token, it can transmit data.



Hardwares in computer network

① Host has 5 layers - AL TL NL DLL PL.

② Cables can be

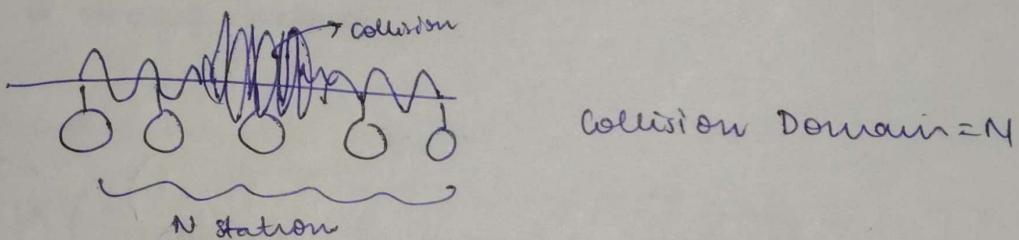
10 base T (10Mbps, No multiplexing, 100m)

10 base 2 (" " , 200m)

10 base 5 (" " , 500m)

↓
length of LAN segment.

- Operate at physical layer. (completely hardware)
- Problem - attenuation (signal strength reduces)
- Collisions are possible



Repeaters -

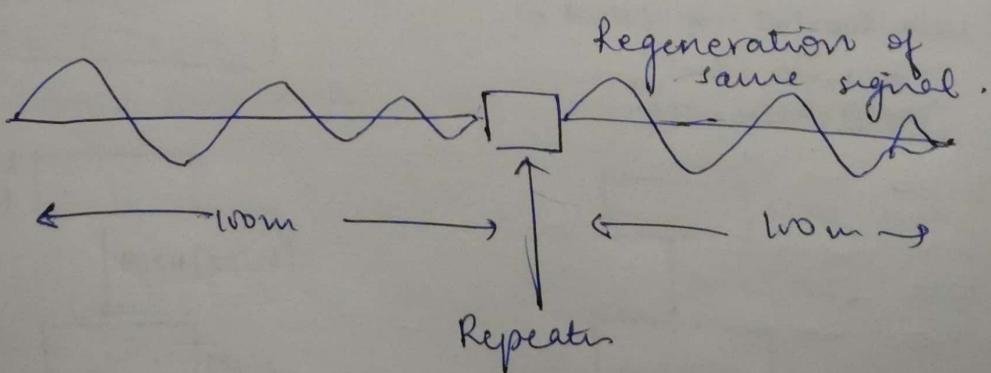
used to increase the length of LAN segment.

For example - 10 base T cables are used.

max distance upto which signal is alive is 100m.

∴ Repeaters are used to transmit signal to greater distances.

LAN spans for
200 m



→ used to connect 2 LAN segments of similar type

→ Physical layer.

→ Collisions are possible.

Collision Domain = n [Collision Domain is unaffected]

If we cut a wire and introduce repeater to connect them,
no change in collision domain.

→ Range of LAN is increased.

switched and bridged

at 2nd and 3rd layer

at transmission and switching

delay time is added

Hub

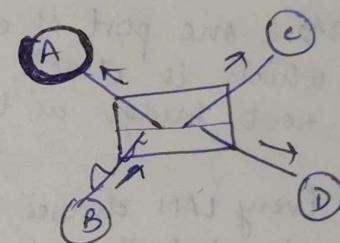
Multiport repeater is called hub.

→ Traffic is very high

→ works purely at physical layer.

→ Collisions are possible.

Collision domain = N

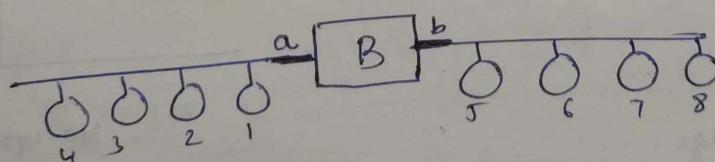


Cost is less -

Bridge

Used to connect LAN segments of different types.

PL, DLL



Native

2 types

Static
manually
filled

Dynamical
Learning/
Transparent.

a) Filtering

b) Forwarding

c) Flooding

able to store and forward
→ no collision inside
bridge.

Collision domain = n^2

	MAC port
1	a
2	a
3	a
4	a
5	b
6	b
7	b
8	b

when a station sends
message, its MAC
port are noted.
time taking but
dynamic.

Avoiding infinite loops in ethernet at IP → Time to live field is used.

Avoiding infinite loops in network containing multiple bridges

→ Minimum spanning tree algorithm

Spanning Tree Algorithm

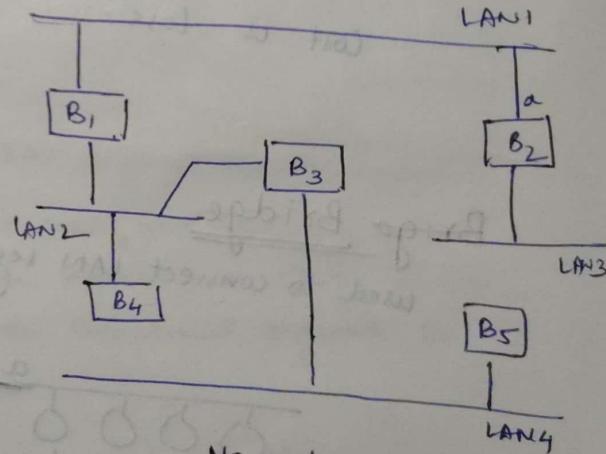
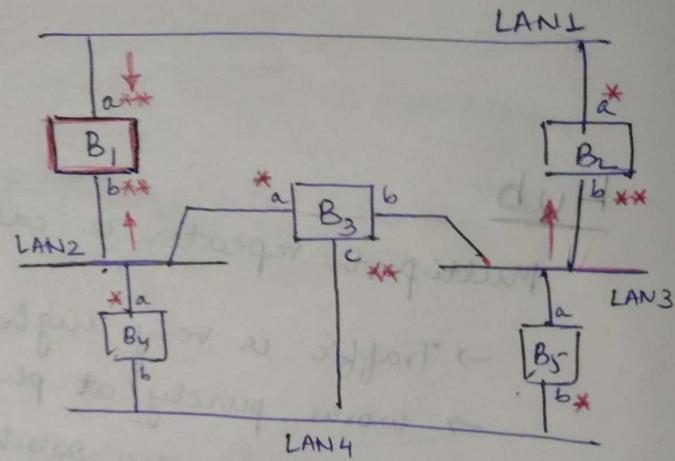
Every bridge has a built-in ID.

The one with smallest ID is taken as root bridge.

Mark one port of each bridge which is closest to the root bridge as the root port (**).

Every LAN chooses the bridge closest to it as the designated bridge for that LAN. Make the corresponding port as designated port (***).

Mark the root ports and designated ports as forwarding ports and block the remaining.

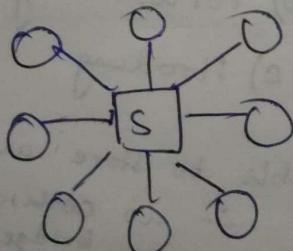


Switch

Multiport bridge.

It connects hosts instead of LAN segment.

Collision domain is 0.



Traffic is less.

It is costly.

Every one can communicate at the same time.

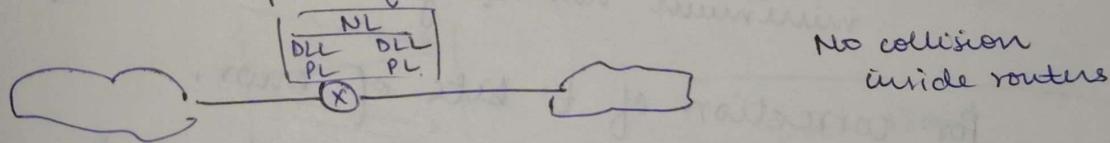
<u>Device</u>	<u>Broadcast Domain</u>	<u>Collision Domain</u>
Repeater	Same	Same
Hub	Same	Same
Bridge	Same	Reduce
Switch	Same	Reduces (o)
Routers	Reduces	Reduces
Gateways	Reduces	Reduces

Routers

Connecting 2 different networks

Every Router has NL, DLL and PL.

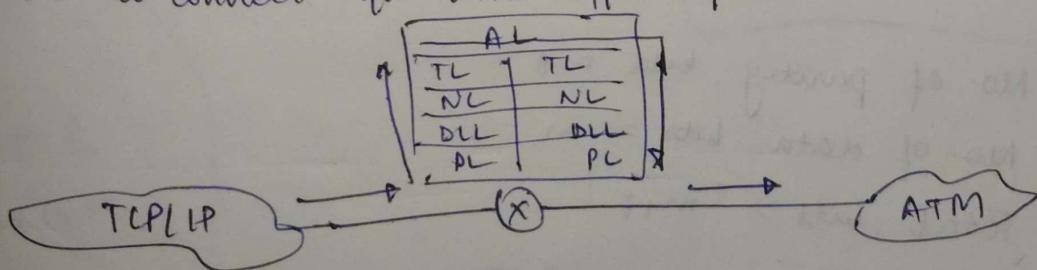
Each port of router has DLL & PL.



Gateways

Protocol converter.

Used to connect n/w with different protocols.



Used for proxy

Application layer	— Application & Gateway
Transport layer	— Transport gateway
N/w layer	— Router
DLL	— Bridge, Switch
Physical	— Repeaters

Hamming code

For detection of t bit error,

$$\text{minimum Hamming distance } d_{\min} = t+1$$

For correction of t bits of error,

$$\text{minimum Hamming distance } d_{\min} = 2t+1$$

$$\text{No. of parity bits} = r$$

$$\text{No. of data bits} = m$$

$$\text{Total bits} = m+r$$

$$\text{Relation b/w } m \& r$$

$$2^r \geq m+r+1$$

congestion detection phase

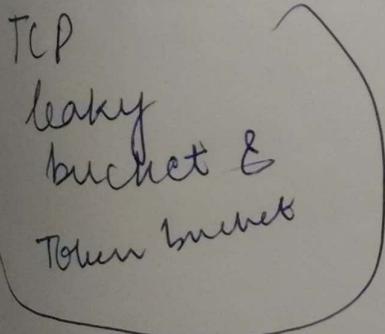
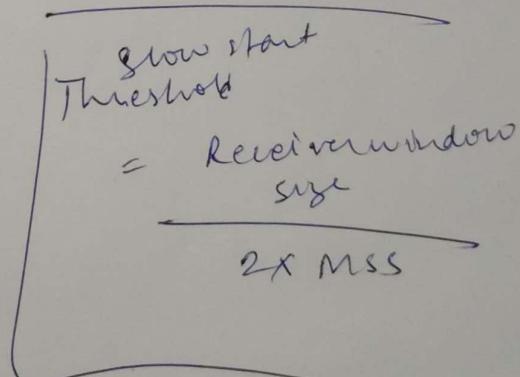
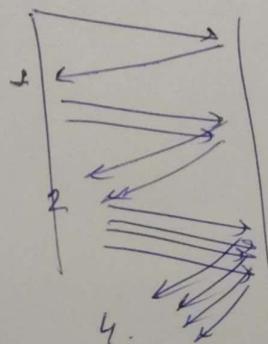
slow start phase

- ① cwnd increases by 1 MSS on every successful acknowledgement.
- ② cwnd doubles after every round trip (set of acknowledgements are reached).

congestion avoidance phase

- ③ cwnd increases by 1 MSS on every success after every RTT.
- ④ cwnd increases by $\frac{1}{2}$ cwnd after every successful acknowledgement.

If cwnd is window size, then cwnd acknowledgements will arrive in 1 RTT



"Life was never meant to turn into thiss!"

COMPUTER

NETWORKS

1. Forouzan Data Communications & Networking
2. Kurose Ross Computer Networking: A Top-Down Approach.
3. Tanenbaum Computer Networks -

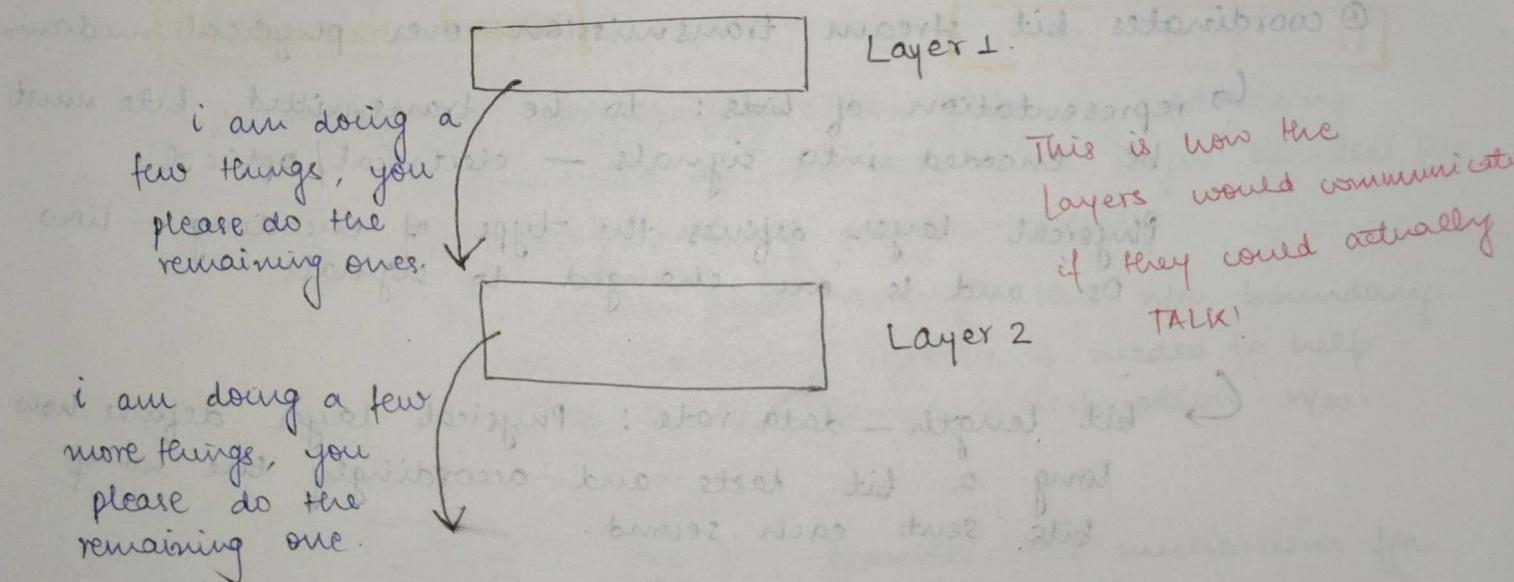
Why Layering?

Establishes principles

1. Modularity - one problem is decomposed into a number of smaller more manageable subproblems.
⇒ more flexibility in designing, modifying and evolving computer networks.
2. Functional Reuse - a common functionality of lower layer can be shared by many upper layers.

- ① Lower layer provides services to upper layers.

Protocol Layer



Protocol

Protocol definition

- ① An agreed upon convention for communication
 - ↳ both endpoints need to understand the protocol.
- ② Protocols must be formally defined and unambiguous.

Layering and Headers

- ↳ Each layer needs to add some control information to the data in order to do its job.
- Header [↳ This data is typically prepended to the data before being given to the lower layer.
- ↳ Once the lower layers deliver the data and control information — the peer layer uses the control information from the peer below to request a signal.

Physical Layer

- ① coordinates bit stream transmission over physical medium
 - ↳ representation of bits: to be transmitted, bits must be encoded into signals — electrical/optical.
 - Physical layer defines the type of encoding — how 0s and 1s are changed to signals.
- ↳ bit length - data rate: Physical layer defines how long a bit lasts and accordingly the no. of bits sent each second.

Data Link Layer

functions

- ② framing: The D.L.L. divides the stream of bits received from the N/W layer into manageable data units called frames.

MAC
address
(48 bits
hardcoded into NIC)

① physical addressing: The D.L.L. adds a header to the frame to specify the NIC address of appropriate receiver on the other side.

② error control: The D.L.L. adds reliability to the physical layer by adding a trailer of information necessary to detect/recover damaged/lost frames.

③ access control: When 2 or more devices are connected to the same link, the DLL determines which device has control over the link at any given time.

Network Layer

packet fragmentation

① logical addressing: The physical addressing implemented by the data link layer handles the addressing/delivery problem locally.

If a packet passes through the N.W. boundary another addressing system is needed to help distinguish b/w source & destination N.W.

② routing: The N.L. provides the mechanism for routing/switiching packets to their final destination.

③ fragmentation and reassembly:

The N.L. sends messages down to the D.L.L. for transmission. Some D.L.L. technologies have limits on the length of messages that can be sent. If the packet that the N.L. wants to send is too large, the N.L. must split the packet up, send each piece to D.L.L. and then have pieces reassembled once they arrive on destination machine.

Routing v/s Forwarding

Routing is the process of finding the routes.

- ↳ Routing protocols can run even if no packets need to be sent. Keeps the table ready to forward a packet.

Forwarding is the process of using a route to send a packet towards the destination.

- ↳ Done only when a packet arrives at the node whose final destination is not this node.

- ↳ Done in real time.

Transport layer

Transport layer

- ① port addressing - Computers often run several processes at the same time. For this reason, process to process delivery means delivery not only from one computer to other but also from specific process to other.

∴ Transport layer address includes a type of address called port address.

- ② segmentation and reassembly -

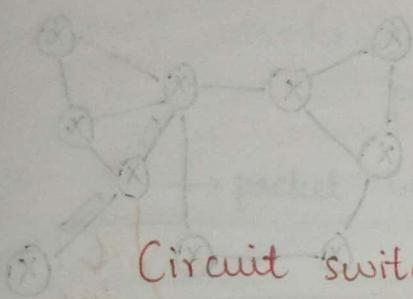
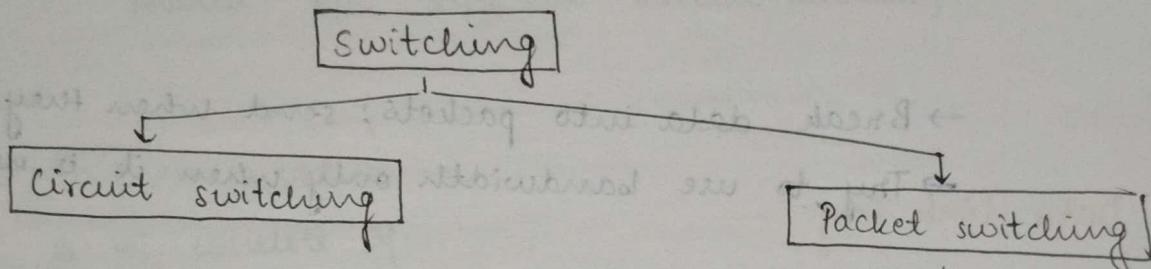
A message is divided into segments, each segment containing a sequence no. These nos enable the transport layer to reassemble msg correctly & to identify & replace packets that are lost.

- ③ flow and error control -

Flow & error control in this layer are performed end to end rather than across a single link.

Switching

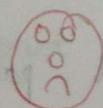
Long distance transmission is typically done over a few of switched nodes.



- Circuit switching -** data is not broken into parts.
- ① data sent continuously
 - ② creating a session reserves dedicated bandwidth in series of switches b/w caller and recip recipient.
 - ③ Guaranteed capacity (in both directions) so long as session up.

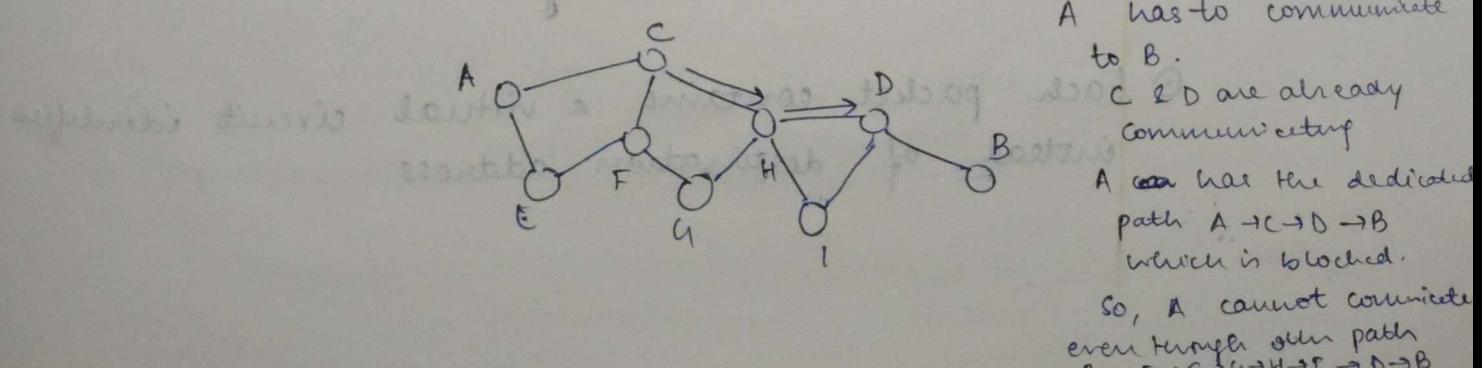
★ → Before sending, must reserve the capacity for session.

→ Success = bandwidth is guaranteed for life of session.



→ often the links may be busy.

(addresses) networks



Packet switching

principles

① Link is shared

② To each frame, a little bit of information (header) is added which tells the switch how to forward it

envelope

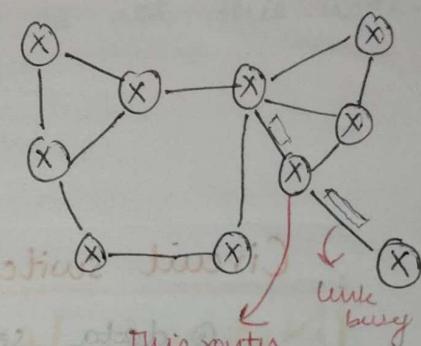
→ Break data into packets; send when they are ready

→ Try to use bandwidth only when it is needed.

transmission time

* Every intermediate device (router) has a queue.

If the link is busy & no alternate link is there, packets are stored in queue and transmitted when link is available.



→ If queue itself is full, drop the packet.

Virtual circuit approach

① Preplanned route established before any packet is sent.

② Call request and call accept packets establish connection (handshaking)

③ Each packet contains a virtual circuit identifier instead of destination address

- ① No routing decisions required for individual packets.
- ② Clear request packet is used to setup circuit.
- ③ Not a dedicated path (links in a path may be shared b/w different virtual circuits).

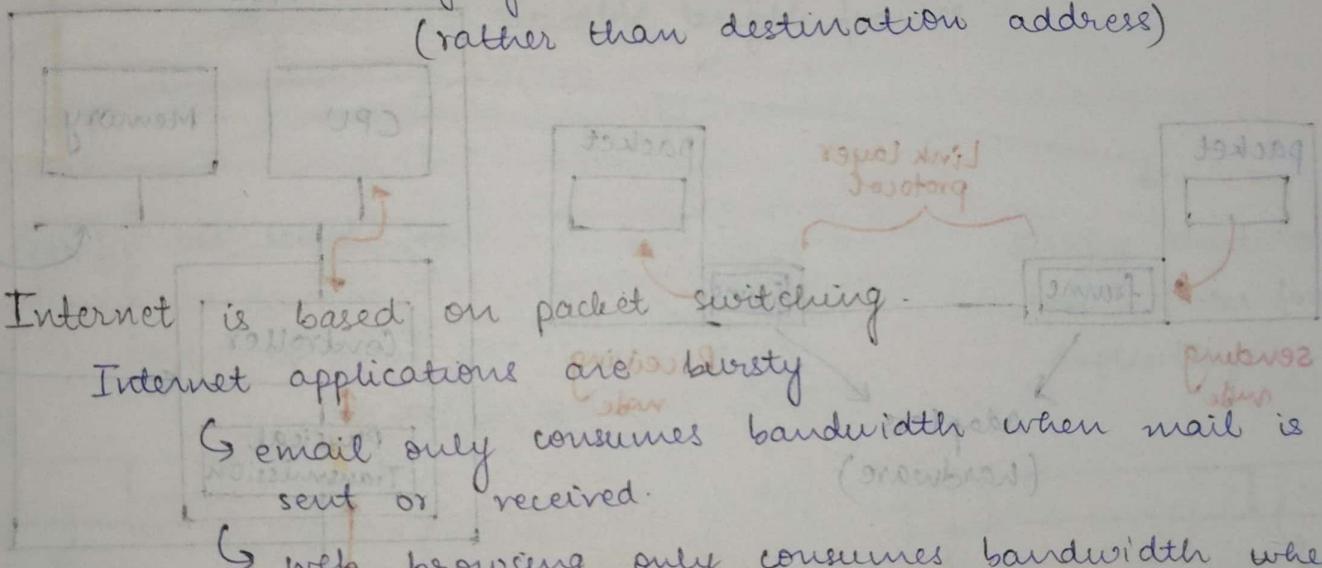
VC implementation -

a VC consists of

↳ a path from source to destination

(SIA) Setup (VC numbers, one number for each link along the path)

→ packet belonging to vc carries VC number
(rather than destination address)



Internet is based on packet switching.

Internet applications are bursty

↳ email only consumes bandwidth when mail is sent or received.

↳ web browsing only consumes bandwidth when you visit site.

④ Packet switching is much more efficient way to support bursty applications

Short burst & many small at short interval

long burst

↳ videoconferencing work emails download music etc.

existing limitations of higher layers prevent off

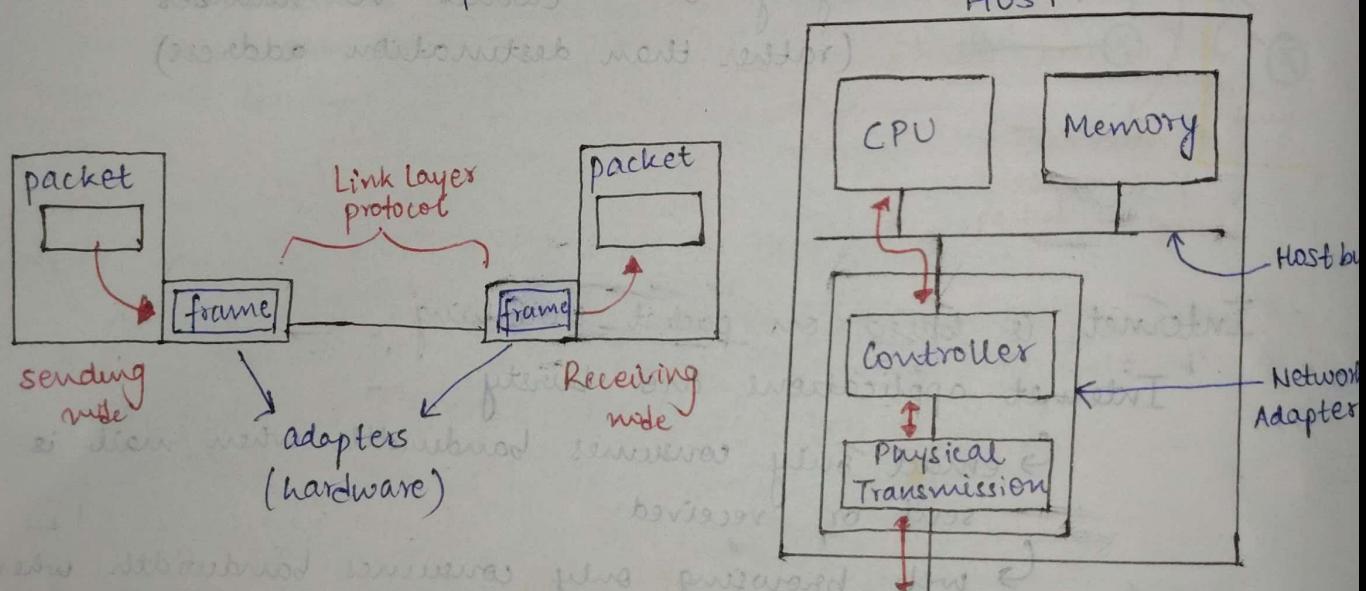
1. Physical Layer.

Scope → concerned with how signals are used to transfer message bits over a link.
(translating traffic w/o binary)

2. Data link layer—

Link layer is implemented by using hardware

- Link layer is implemented in 'adapter' (NIC)
- Adapter receives datagram from net layer and creates frames.



1. Framing

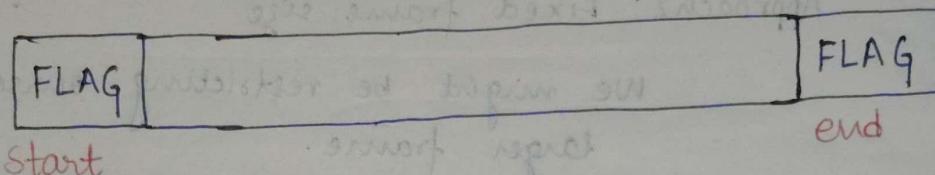
The data stream is broken up into discrete chunks.

Receiver needs to know when a frame starts and ends.

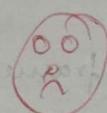
Otherwise errors from misinterpretation of data stream.

Byte stuffing

Add 1 byte of flag at the start and end of the frame.



Two consecutive flags indicate start and end of frame.



what if FLAG itself is there in the data

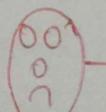
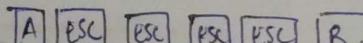
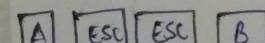
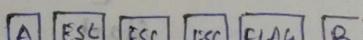
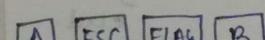
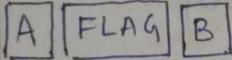
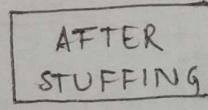
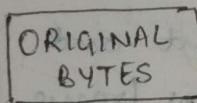
(afterall, it's just a sequence of bits. RIGHT?)

solution

use escape character if FLAG is present in data. #FLAG.

Rule at receiver end

whenever # is present, ignore escape character and consider next byte as part of data.



what if receiver contains escape character in its sequence?

solution

use one more escape character

→ treated as end FLAG - # FLAG - - - FLAG

→ - - # - - -

FLAG - - # # - - - FLAG

→ - - # - - -

ESC, FLAG → These take up 1 byte each time used.

Therefore byte stuffing is not optimal.

Bit stuffing

Simple algorithm -

01111110 used as flag

At transmitter → after every 5 consecutive ones, insert a zero.

At receiver → In a pattern of 111110 anywhere in the data, remove the trailing

TRANSMITTER

01111110

browns

RECEIVER
011111010

1111100 → 11111000
11111 → 111110

{ lost one bit
bottam }

2. Error Handling

Detect and/or correct errors in received frame.

Errors

Single bit error

only 1 bit gets corrupted.

0 0 0 0 0 0 1 0

0 0 0 0 1 0 1 0

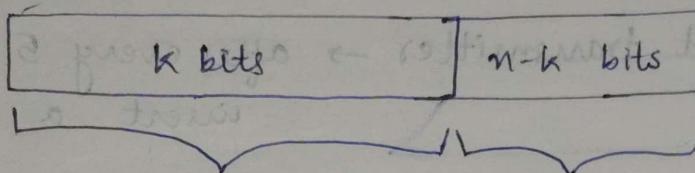
Burst error

multiple bits get corrupted.

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

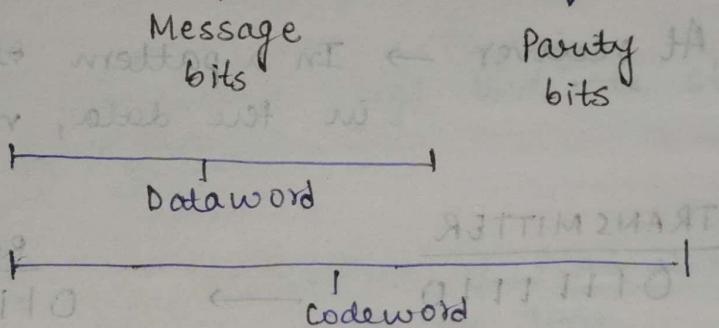
Block Coding

↳ a few bits are appended to the message
e.g. as now 0111110



The bits can be appended anywhere

- start ✓
- middle ✓
- end ✓



If there are k bits for dataword,
No. of datawords possible = 2^k .

Error control methods

Error Detection

- ① → Parity bits
- ② → Cyclic Redundancy Check (CRC)
- ③ → Checksums

Error Correction

- ① → Hamming codes.

Receiver always knows the list of all valid codewords.

Parity Checking

- ① simplest scheme
- ② add an extra bit to the data, such that the total number of 1 bits is even (even parity) or odd (odd parity)

→ Even parity :- make the no. of 1s in a bit string an even number.

commonly used

→ Odd parity :- make the no. of 1s in a bit string an odd number

→ Check at the receiver

Given a message of k data bits, D_1, D_2, \dots, D_k append a parity bit P to make a codeword

→ P is exclusive OR of the data bits

$$P = D_1 \oplus D_2 \oplus D_3 \oplus \dots \oplus D_k$$



→ Receiver can detect error only if the number of corrupted bits are odd

$S = m+k$

9 M

Hamming distance

The Hamming distance b/w 2 words (of same size) is the number of differences b/w the corresponding bits.

$$d(000, 011) = 2$$

$$d(10101, 11110) = 3.$$

$$d(01101010, 11011011) = 4$$

Minimum Hamming distance -

The minimum Hamming distance is the smallest Hamming distance b/w all possible pairs in a set of words.

$$\text{Minimum Hamming distance} = 2.$$

Minimum Hamming distance in case of single parity codes

$$d_{\min} = 2$$

IMP

If the minimum Hamming distance b/w any 2 valid codewords is represented as 'd'.

Maximum no. of bits of error that can be detected by the receiver = $d-1$.

To detect detection upto s errors, what should be the minimum Hamming distance in the set of codewords?

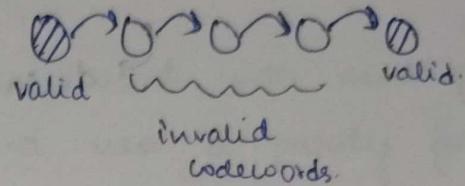
A. s

Suppose $s = 3$.

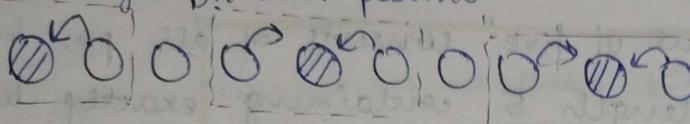
B. $s-1$

~~C. $s+1$~~

D. None of these.



- Only single bit error possible



This way, no set of s errors in a single bit could turn one valid codeword into some other valid codeword.

if invalids are non overlapping, then, we

can always correct errors.

④ We can correct errors if we are really sure about the original codeword that caused it.

d_{\min} = Minimum Hamming distance

Maximum bits of error that the

- receiver can correct = $\frac{d_{\min} - 1}{2}$

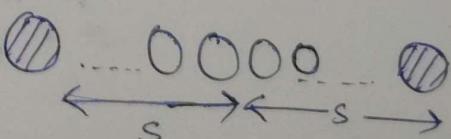
To guarantee CORRECTION upto s errors, what should be the minimum Hamming distance in the set of codewords?

A. $2s$

B. $2s-1$

~~C. $2s+1$~~

D. None of these.



$$(1+2s)$$

$$\therefore 2s+1 = d_{\min}$$

If minimum d invalid codewords in between,

↳ we can detect x bits of error

↳ we can correct $\lfloor \frac{x}{2} \rfloor$ bits of error

$$d_{\min} = x+1.$$

bits of error
are small

Ques

The 'two out of five' consists of all possible binary words of length 5 containing exactly two 1s.

No. of codewords $\rightarrow \underline{\underline{\underline{\underline{\underline{5C_2}}}}}$

2. Minimum Hamming distance $\rightarrow 2$

3. Errors that the code can detect $\rightarrow 1$

4. Errors that the code can correct $\rightarrow 0$

Summary of Hamming distances -

→ All errors of d or fewer bits in a codeword can be detected if and only if the minimum Hamming distance of the code is $(d+1)$.

→ All errors of c or fewer bits in a codeword can be corrected if and only if the minimum Hamming distance in the code is $(2c+1)$.

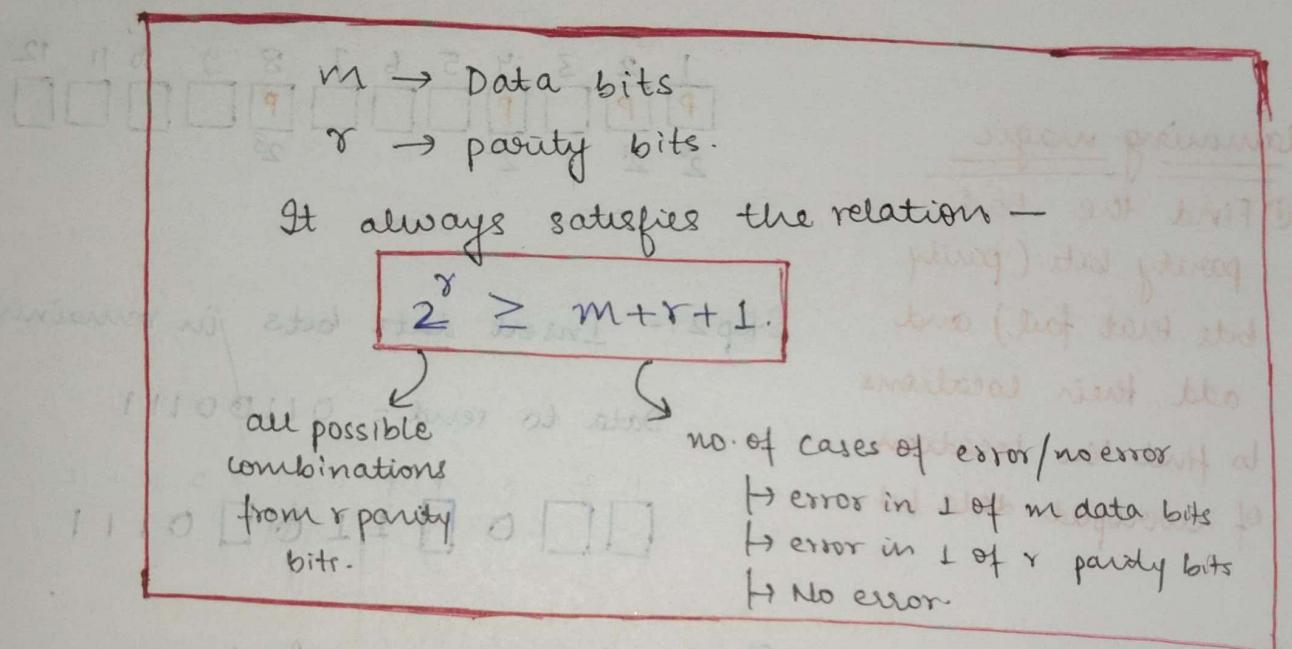
→ A code is a error correcting AND error detecting ($d \geq c$) code if and only if its minimum Hamming distance is $(d+c+1)$.

Single Error correcting code (SEC)

Designing code which can correct single bit error

Ex → Hamming code

Basic idea → Divide the dataword into multiple overlapping groups and use a parity for each group separately.



Let k be the minimum number of bits needed to correct all possible t bits error then, which of the following conditions must satisfy?

$$\text{No. of parity bits} = \geq k$$

$$2^k \geq 1 + \binom{n}{1} + \binom{n}{2} + \binom{n}{3} + \dots + \binom{n}{t}$$

Hamming Code

→ Construction

① Number the bits starting from 1: bit 1, 2, 3, 4, 5, ...

Step 1:- Mark all bit positions that are powers of 2 as parity bits.

(positions 1, 2, 4, 8, 16, 32, ...)

1	2	3	4	5	6	7	8	9	10	11	12
P	P		P				P				

$2^0 \quad 2^1 \quad 2^2 \quad 2^3$

Hamming magic

② Find the bad parity bit (parity bits that fail) and add their locations to find the location of corrupted data bit.

Step 2:- Insert data bits in remaining positions

Data to send = 01100111

1	2	3	4	5	6	7	8	9	10	11	12
		0		1	1	0		0	1	1	

Step 3:- Calculate the Parity bits

Each parity bit calculates the parity of some of the bits in the codeword. The position of the parity bit determines the sequence of bits that it alternately checks & skips.

Position 1:- 1, 3, 5, 7, 9, ...

Position 2:- 2, 3, 6, 7, 10, 11, ...

∴ Codeword is

$$\left(\begin{array}{c} 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{array}\right) + \left(\begin{array}{c} 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{array}\right) + \left(\begin{array}{c} 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{array}\right)$$

~~0100111~~

010111010111

$$P_1 = 3^{\text{rd}} \oplus 5^{\text{th}} \oplus 7^{\text{th}} + 9^{\text{th}} + 11^{\text{th}} \\ = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 = 0$$

$$P_2 = 2^{\text{nd}} \oplus 3^{\text{rd}} \oplus 6^{\text{th}} \oplus 7^{\text{th}} \oplus 10^{\text{th}} \oplus 11^{\text{th}} \\ = 0 \oplus 1 \oplus 0 \oplus 1 \oplus 1 = 1$$

$$P_4 = 5^{\text{th}} \oplus 6^{\text{th}} \oplus 7^{\text{th}} \oplus 12^{\text{th}} \\ = 1 \oplus 1 \oplus 0 \oplus 1 = 1$$