→ Introduction to OSI ✓

→ Flow control, Error control ****** ✓

→ Ethernet **** ✓

→ IPV4, Basics of IPV6 ****** ✓

→ Networking devices (Bridge, Switch, repeater, Hub) * ✓

→ TCP/IP, Sockets ****** ✓

→ Congestion control and Routing Algorithms *** ✓

→ Application Layer Protocols & Security ** ✓

⇒ ****** ≈ 60%–70% M in Gate (VI)

**** – 10%–15% ≈ 2M – 4M.

*** – 10%–15%

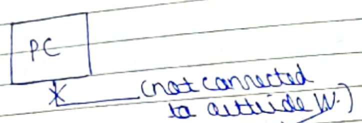
$\left.\begin{array}{r} ** \\ * \end{array}\right\}$ – remaining. ✓

1

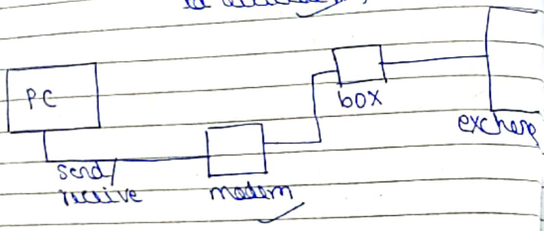
On avg: 8M – 10M from CN

** INTRODUCTION TO OSI

⇒ Closed System



⇒ Open System

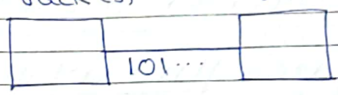


- Comp which is able to send/receive data: open System. (connected to outs. world)
- Closed System: can't access internet.
  → Autonomous System.

* Computer Networking

A group of Autonomous systems connected to same transmission medium is called as concept of computer Networking.

1970s



→ just connect. want exchange info.

1) (interface b/w comp & wire) (send data onto wire)
→ NIC (Network interface card) (MAC add)

2) we need a terminal to type (GUI) – Graphical User Interface
→ interface for sender to send a message.

S →(1LU) 0110... R
(but due to Noise on medium)
1 → 0, 0 → 1
∴ (msg prone to errors).

3) We require an error control Mechanism. (∵ Noise can't be avoided).

Scanned with CamScanner

⇒ Jack's internet can send 10^3 BPS.
but Jill recieves, 10^2 BPS.

1s : 1000 bytes can be flown into wire
1s : 100 bytes will be receiv.

∴ 900 Bytes lost. (technically).

4) But, due to some flow control mechanism, data won't be lost.

5) Security is needed.

6) Data conversion

7) Compression is required.

** To have a proper computer n/w between computers, just putting a medium is not enough.

All funct^n are required.
⇒ these functionalities are compulsory needed.

⇒ [International Telecom Standard lists down all major functionalities.]
divided into Layers.

* Layer : Set of Functionalities.

- The Standard is OSI (Open Source Interconnection)
  → theoritical model. [7 Layers].
- But, TCP is actually implemented Model [5 Layers].

* OSI

OSI	TCP
Applicn Layer	Applcn Layer.
Presentatn Layer	Tranpart Layer
Session Layer.	N/W Layer
Transport Layer	Data Link Layer
Network Layer	Physical Layer
Data Link Layer	
Physical Layer	

* Comp 1 (msg)

Comp 1 (msg)	Comp 2
AL	AL
Pres. L	Pres. L
Session L	Session. L
Transport L	Transport. L
N/W L	N/W L
Datalink L.	Datalink L.
Physical L.	Physical L.

[Medium → PHYSICAL medium, ... casting]

2

* When installing OS, layers also installed, along with TCP Protocol communicⁿ with NIC through OS.

⇒ Msg goes through all layers.

overhead

Comp 1	(Sender) Msg		(Receiver) Msg	Comp 2
AL	AH | msg	APDU	AH | msg	AL
PL	PH | APDU	PPDU	~~PH~~ | APDU	PL
SL	SH | PPDU	SPDU	~~SH~~ | PPDU	SL
TL	TH | SPDU	TPDU	~~TH~~ | SPDU	TL
N/W L	NH | TPDU	NPDU	~~NH~~ | TPDU	NL
DLL	DH | NPDU	DPDU	~~DH~~ | NPDU	DLL
PL	PH | DPDU	PDPU	~~PH~~ | DPDU	PL

PDPU (wire)

[APDU: Application L. Protocol Data Unit]

(Pg 1)

Q.1 Suppose end user has 200 bytes data to send to destination. Each layer adds 20 bytes overhead.

Sender	Layer	Receiver
200		200
↓ 220	AL	↑ 220
↓ 240	PL	↑ 240
↓ 260	SL	↑ 260
↓ 280	TL	↑ 280
↓ 300	NL	↑ 300
↓ 320	DLL	↑ 320
↓ 340	PL	↑ 340

medium

⇒ Pack. size at each layer shown. (a)

(b) AL : Size = 220 bytes

- 20 bytes (OH)
- 200 bytes (act. data) [payload]

Actual msg : Payload

$\text{overhead \% (AL)} = \frac{\text{required}}{\text{total}} \times 100$

$\Rightarrow \frac{20}{220} \times 100$

$\text{Payload \% (AL)} = \frac{200}{220} \times 100$

* Present Layer : 240 Bytes
  - 40 (overhead)
  - 200 (Payload)

$\text{overhead (\%)} = \frac{40}{240} \times 100 \Rightarrow 16\%$

$\text{Payload (\%)} = \frac{200}{240} = 83\%$

* Session layer : 260

$\text{overhead \%} = \frac{60}{260} \times 100 = 23\%$

$\text{Payload \%} = \frac{200}{260} \times 100 = 76\%$

* Transport layer : 280

$\text{overhead \%} = \frac{80}{280} = 28\%$

$\text{Payload \%} = \frac{200}{280} = 71.4\%$

* Network Layer : 300

$\text{Overhead \%} = \frac{100}{300}$

~~Q2~~ Pg 1

(Added & Subtracted & added)

* Protocol : Set of rules req for compatibility b/w 2 Systems.
  - 1. Syntax
  - 2. Semantics
  - 3. Timing

1. Syntax

| Header | Payload → actual msg |

| Header | Payload | trailer |

error control algorithms

error control info

2. Semantics

meaning of data path

= meaning of header : functionality added at each Layer.

Page No. 10

P

Payload: Actual data to be Sent

Trailer: error control information

3. Timing: what to be done at what time & at what layer?

- We have discussed timing till now

Now → what is happening at each layer.

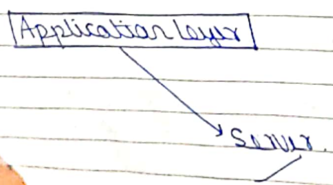
* Functionalities of Each layer

1. Application layer provides user interface to Network

• eg: makes us accessible to N/W.
→ access flipkart Server.

2. Server Side Synchronisation

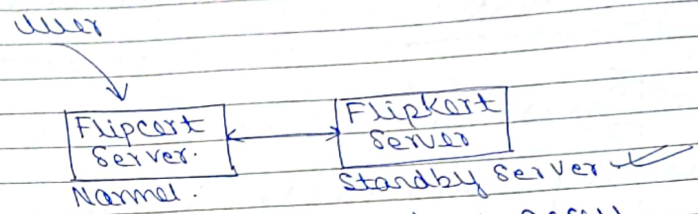
3. Checks if the application is ready for transmission of data.



[NL → ip add
TL → Port no]

Page No. 11

2 Modes → Normal.
→ standby.



(when Normal goes down, access moves to standby Server).

→ if we want to access server → Set of rules handled by AL Protocols.

** [AH | Message] APDU

Appln Headers
(adds only length of data)

2. Presentation Layer

(i) Responsible for data conversions → not req. in all cases.
(ii) data encryption & decryption
(iii) data compression & decompression.

[PH | APDU]

• data conversion algo.
• key in encryp/dec.
• algo for compression

VI

→ PH → information related to data conversions, encryption, compression.
(will be added in presentation layer Header).

5. Session layer

(i) It provides session to end users (unique values for each user)
(ii) Will be used for establishing, manage, terminate sessions of end users.
(iii) Synchronization of the data flow. (Check Pointing)

- from torrent, if light off, download starts from that point
- flipkart doesnt go out after logging out.

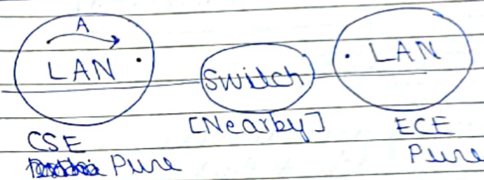
→ If connectivity losses b/w sender & receiver, start sending from past point.
(we need to maint. sessions at user level).
[Session info used for check pointing].

till user convenience purpose (these 3 layers).

[Actual Sending of Data]

4. Transport layer

(i) It provides end to end communication between Sender and receiver.
(ii) Port no. required. (TCP Layer)
(iii) make you reach to actual Process.



→ But if commⁿ b/w cities → MAN. (Router)
→ But if commⁿ b/w countries → (Routers + Gateway (WAN))

A: only DLL enough

N/W → Host → Ports (Process)
[N/W layer].

Both identified with ip Addr.

TCP & IP both useful ∴ TCP-IP Protocol.

N/W → house no.
TL → house no. → member A.

5. Network Layer

(ii) Source to Destination communication b/w sender & receiver

Router = identifies which n/w to send & drops msg to host (N/W Layer).

TRANSPORT LAYER

TCP
- Client server communication
- Socket (ip Add + Port No.)
  created at both client side & server side)

end to end communicatⁿ

connection oriented → connection less.

connection oriented:
1. establish connection
2. data transfer
3. close connection.

[Min Header Size = 20 B at any Layer]. 15

* Connection-oriented (more complex, time↑)

1. establish connectⁿ with receiver to underst. capacity of reciever.

C1 (5000B) C2 (1000B).

if no establish: 4000B Lost.
but if connectⁿ oriented:

send:

980	20
980	20
980	20
980	20
980	20
100	20

→ reciever [reassembly is done at receiver side] (with help of sequence nos)

6 divisions needed. [Segmentation at Sender needed] for proper communicⁿ. → to prevent data loss

TL functionalities also

- Segmentation & reassembly
- flow control
- reliability
- error control

} connectn oriented

* At receiver side, reassembly

5. Network Layer

(i) Source to destination communication.
(ii) Logical Addressing
- divided accord. to IP, subnets ...
(IPV4 & IPV6).

(iii) Congestion Control [Traffic control]

S1 3, S2 4, S3 2 → 9MBps → Router (5MBps) → Getting more data than capacity.

(iv). Routing (Routing Protocols)

(v) Fragmentation [Most imp]

TL 2000 bytes
NL
DLL

MTU → Max Transmis unit (Gen: 1500 B) [Can't send more than 1500B]

present in DLL

∴ reject (no loss) >1500.

2000 (after Adding Header) [every packet req. Header]

NPDU: $\underbrace{1480 \mid 20}_{1500}$ 520 | 20 : Fragments.

* Fragment: divide data in order to send data to the down layer.

* Connection-less Protocol (TL)

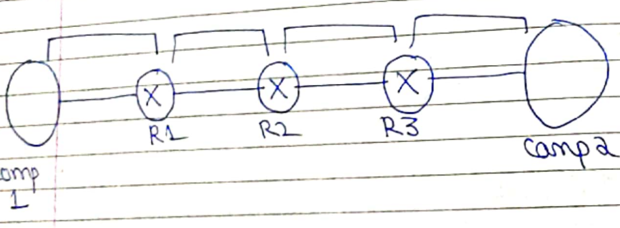
- won't be able to see the capabilities of receiver
- There is loss.
- No reliability.
- establishment time reduced ↓
- Fast transmission, Loss → no Prob.
- Multimedia Protocol
  UDP Protocol
  VOIP Protocol → Voice over IP.

N/W Layer.

Message Switching used by connection less whereas packet Switching used by connection oriented.

6. Data link layer

(i) Node to node communication.
(ii) Physical Addressing (MAC)
(iii) framing
(iv) flow control
(v) error control



* Flow Control (TL) > complex Flow Control (DLL)

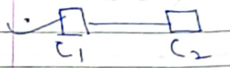
(TL) → integration Testing. (can be thought of)

(DLL) → unit testing.

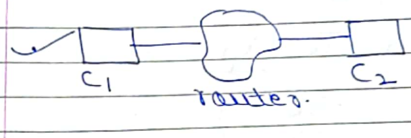
* Commⁿ on Wire: Physical Layer.

[Minimum req.]

- Wire — PL.



: DLL (within LAN — OK)



: NL. (within MAN) [cities]

WAN → uses Gateway / Port info.
→ Transport Layer [countries]

TL:	WAN
NL:	MAN
DLL:	within LAN

: Min req. (layer) for Sending data.

* in LAB, DLL enough.

- 192.75.68.35 → i/p of CPU

LAN (inside comp., anyone can access)

* Tell admin, make IP static, & allow access to specific router (exhibition).

p1

7. Physical Layer

(i) Conversion of Bit to Signal
(ii) Multiplexing / Demultiplexing.
(iii) Encoding / Decoding.

(Material) → Pg1

Q3 internet Layer: TCP/ip.

- AL
- TL
- NL
- DLL
- PL

} 5 Layer.

Q4 ✓

[D4.]

Q5

OSI Model	TCP/IP Model
(i) Theoritical Model	(i) Practically implemented.
(ii) 7 layers	(ii) 5 layers.

Simmilarities

OSI Model	TCP/IP Model
(i) Both uses layered Approach.	
(ii) Functionalities are Same in OSI & TCP/IP.	

6. Flow control / error can be done by TL & DLL.

7. Answers → in Material.

8. $M+(n7) \Rightarrow M+7n$

$\Rightarrow M+n^2$

$\therefore M+nh \qquad \therefore \frac{nh}{M+nh}$

9. Payload = 500

$500+(60+50+40+18)$

$\Rightarrow$ 668 B → sent to wires.

610 Bytes (size of packet at TCP Layer)

8. Layers → n

1 $M+h$
2 $M+2h$
⋮
n $M+nh$ → sent to wires.

$\therefore$ overhead % = $M+nh$ → M (Payload), nh

$\therefore$ fraction $\rightarrow \frac{nh}{M+nh}$

9. [Protocols are given here]

FTP $\rightarrow$ Applicn L
TCP $\rightarrow$ TL
IP $\rightarrow$ NL
Ethernet $\rightarrow$ DLL

	Sender	Receiver
	500	500
AL	560	560
TL	610	610 Ans.
IP	650	650
DLL	668	668

10.

	Header	Trailer	x
AL	10	0	$x+10$
TL	20	0	$x+30$
NL	20	0	$x+50$
DLL (MTU)	14	4	1500

(B4 entering into DLL : should be less than 1500)

$x+50 \leq 1500$
$x \leq 1450$

* BASICS OF ERROR CONTROL

→ Types of Errors
→ Bits in error
→ Finding no of transmissions for error Packets / No error Packets
→ Code Word, Hamming Distance

Techniques :

→ Error detection techniques
→ Error correction techniques.

* TYPES OF ERRORS

1. Damage
2. Loss
3. Duplicate

Damage → Single bit error, Burst error.

Single bit error:

Sender: 101010
Rec: 101011

· Burst error.
- Sequential → noise occurs sequentially
  - Sender: 10101010
  - Receiver: 10010110
  - Burst error length = 4 bits
- Non-sequential → noise occurs not sequentially
  - Sender: 10101010
  - Receiver: 01101001
  - burst error length = 8 bits (from where error starts to where error ends).

2. loss → [due to loss in the signal of transmission medium].

Sender → receiver

P1 → ~~lost~~ lost.

3. duplicate → due to loss of acknowle. & delayed acknow.

Sender → receiver

P1, P1 } duplicate

(will be explained further in error control mechanisms)

* No. of bits in Error:

The no. of bits in error will depend upon:

1. Bandwidth of Medium
2. Noise duration.

1 msec = 10^{-3} s
1 μs = 10^{-6} s
1 ns = 10^{-9} s
1 KBps = 10^{3} Bps
1 MBps = 10^{6} Bps
1 GBps = 10^{9} Bps

(in OS: 2 powers)

Q. Bandwidth of Medium is 1 Kbps.
Noise duration: 1 sec

No. of bits in error?

1s ⟶ 1 Kb (bits flowing)

∴ 1s = 1 Kb error.

Q. BW = 10 Kbps
Noise duration = 2 msec.

1s ⟶ $10^3 \times 10$.
2×10^{-3} ⟶ x.

$x = 2 \times 10^{-3} \times 10^{3} \times 10$ = 20 bits

$1s — 10^4$
500×10^{-6} — x

$5 \times 10^2 \times 10^{-6} \times 10^4$

$\Rightarrow$ 5 bits

Lecture 2

8. $\longrightarrow$ Pg 4

(a) 1500 bps

1500 b — 1 s
x — 2×10^{-3} s

$x = 1.5 \times 10^3 \times 2 \times 10^{-3}$ s

$\therefore \boxed{x = 3b} \rightarrow$ 3 bit error.

(b) 12 Kbps

12×10^3 b — 1 s
x — 2×10^{-3} s

$\therefore x = 2 \times 10^{-3} \times 1.2 \times 10^4$
$\Rightarrow 2.4 \times 10$
$\Rightarrow$
$\therefore \boxed{x = 24 b}$

(c) 96 Kbps

96×10^3 b — 1 s
x — 2×10^{-3} s

$\therefore x = 2 \times 10^{-3} \times 96 \times 10^3$

$\therefore \boxed{x = 192 b}$

units.

** Digital Signal → 101.... - bits/s

Analog Signal → ∿∿ - Hz

bit rate = 2000bps (BandWidth)

1 sec —— 2000 bits
x —— 1 bit

1 bit delay / 1 bit duration

$\therefore x = 1/2000$

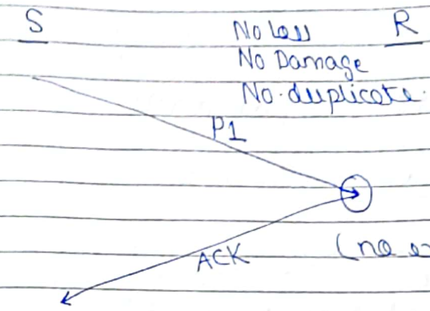
$\Rightarrow \frac{1}{2} \times 10^{-3} \Rightarrow 0.5$ msec.

* $\boxed{\text{1 bit delay} = \frac{1}{\text{bandwidth}}}$

Q Finding no. of transmissions for the packet with errors & without errors

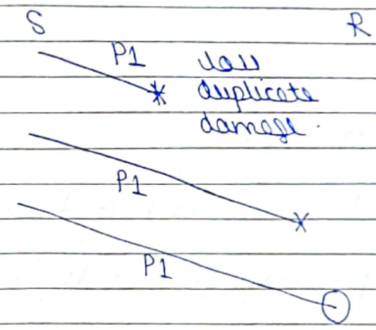
{ loss, damage, duplicate }

I)



1 transmiss (per packet)

II)



(per packet multiple transmission)

VI

* $\boxed{\text{Average no. of transmissions per packet} = \frac{1}{\text{Prob. of success rate}}}$

Q. SR = 40%.

Avg. no. of transm = $\frac{1}{0.4} = 2.5$

Pg 4

Q11

60% error → failure rate

∴ Success rate = 40%

Avg = $\frac{1}{0.4}$ = 2.5

(Avg no of frames → 2.5)

Q10 error rate = p

Success rate = 1 − p

Mean no. of transmission = $\frac{1}{1-p}$

(if ack lost, packets will be duplicate)

Q12 $\frac{1}{0.8} * 10$ =

Success rate → 80%

⇒ $\frac{1}{0.8}$ = 1.25 per packet. → *10

∴ 12.5

∴ 12.5 transmissions on avg for 10 frames

* Probabilities in Error Control

1) P_b = Probability of one bit error (Bit error Rate)

BW = 2Kbps
What is BER?

BW = 2×10^3 b — 1 s.

∴ BER = $\frac{1}{2 \times 10^3}$ = 0.5×10^{-3}

⇒ 0.5 ms

⇒ $\boxed{5 \times 10^{-4}}$ (no unit)

2) $1 - P_b$ = Probability of one bit without error

frame = combination of bits

$(1-P_b)(1-P_b)(1-P_b)$ . . . . L bits

3) ⇒ Probability of frame without error ⇒ $(1-P_b)^L$

Probability of frame with error

$$1-(1-P_b)^L$$

Q. Bandwidth = 10Mbps
what is Bit error Rate?

$$BER = \frac{1}{10 \times 10^6} = 10^{-7}$$

Q. A length of frame is 8 bits.
BW = 1 Kbps.
what is the Probability that the ust frame contains errors?

$$P_b = \frac{1}{10^3} \Rightarrow 10^{-3} \rightarrow ber$$

$$1-(1-10^{-3})^8 \text{ Ans}$$

Q. The frame is divided into two chunks for applying parity where each chunk is added with 1 parity bit. what is the probability of frame without error?

Soln. 8 bits → 4b (Chunk 1), 4b (Chunk 2)

4b + 1 → Parity bit

4b + 1 → Parity bit

∴ Total 10 bits (after adding parity)

$$(1-10^{-3})^{10}$$

* HAMMING DISTANCE

* CODE WORD.

→ Message + Checkbits

S → R ← error control algo on packets.

- CB for error control algo.

(no. of CB depend upon error control algo)

- Codeword rec. by receiver

S → R : Codeword

Header	Payload	Trailer

Payload: Msg + CB; Trailer: error algo

- Trailer is taken, applied on CW to check received msg is correct or not

* Hamming Distance

The no of bits changed between 2 code words is called as Hamming Distance.

eg: CW1 = 1010 ⊕ → comp. does EXOR op.
CW2 = 1111
0101
→ No. of 1s = HD → 2

$\boxed{\text{Hamming Distance } (d) = \text{no. of 1s}}$

iii) 3 CWs, {CW1, CW2, CW3}

$\boxed{d = \min\{CW1 \oplus CW2,\ CW1 \oplus CW3,\ CW2 \oplus CW3\}}$

* Advantages of Hamming Distance.

(i) Max no. of errors that can be detected $= d-1$

(ii) Max no. of errors that can be corrected $= \left\lfloor \frac{d-1}{2} \right\rfloor$ → step down.

Fast method.

13. CW1 = 10010011
CW2 = 11000011
CW3 = 01001001

$\left.\begin{matrix}CW1 = 10010011\\ CW2 = 11000011\end{matrix}\right\} 2$, $\left.\begin{matrix}CW2\\ CW3\end{matrix}\right\} 3$, CW1, CW3 } 5

CW1 ⊕ CW2 = 01010000
C

∴ min(2, 3, 5) → 2

GATE PROBLEMS

Q. Choose the best matching b/w Grp1 & Grp2.

Group 1	Group 2
P: DLL	1. Ensures reliable transm. of data over point-to point link layer. [DLL]
Q: NL	2. Encode & decode data for physical transmission. [PL]
R: TL	3. Allows end to end communication b/w 2 computers.
	4. Routes data from one network to another.

Q. In a communication network, a packet of length (L bits) takes link L1 with a probability of P1 or link L2 with a probability of P2. Link L1 and L2 has bit error probability of B1 and B2 respectively. The probability that the packet will be received without error via link L1 or L2:

P

$(1-b_1)^L$ $(1-b_2)^L$

L1 (P1) b1 L2 (P2) b2

(a) $(1-b1)^L P1 + (1-b2)^L P2$

(b) $[1-(b1+b2)^L] P1 P2$

(c) $(1-b1)^L (1-b2)^L P1 P2$

(d) $1-(b_1^L P1 + b_2^L P2)$

without error $(1-P_b)^L$

$\therefore (1-b1)^L \cdot P1 + (1-b2)^L \cdot P2$

or

Q. An error correcting code has following code words.

CW1: 0000 0000
CW2: 0000 1111
CW3: 0101 0101
CW4: 1010 1010
CW5: 1111 0000

What is the max no. of bit errors that can be corrected?

$d=4;\ \left\lfloor \frac{4-1}{2} \right\rfloor = \left\lfloor \frac{3}{2} \right\rfloor = \lfloor 1.5 \rfloor = 1$

Step down

FR

Q. On a wireless link, the probability of packet error is 0.2. A Stop & wait protocol is used to transfer the data across the link. What is avg no. of transmission attempts that are required to transfer 100 data packets.

$\frac{1}{0.8} \times 100 = \frac{100 \times 10}{8}$ 5/4 4

⇒ Prob. of success rate = 0.8

∴ 125 transmissions (avg).

* ERROR CONTROL MECHANISMS (Detection)

1. Parity ⟶ 1980s (not used now)
2. CRC ⟶ DLL — (CB added at end of msg)
3. CheckSum ⟶ TL.

(Aim to form CW)

* Parity — $\boxed{\text{Msg + Check bits}}$ ← generated by correspond. algo.

Msg → Parity

- one-dimensional — 1 parity bit.
- two-dimensional

(1 parity bit will be added)

even	odd
$\underbrace{1010}_{m}\ \underbrace{0}_{P}$	$\underbrace{1010}_{m}\ \underbrace{1}_{P}$
(even 1s)	(odd 1s)

* S → R

10100 → ⊙

noise

10111 → ⊙

Sender said he send even parity so rec. checks.

∴ will work for 1/odd bit error.

∴ they introduced two-dimensional parity

(i) The message breaks into multiple blocks according to block size which will be further broken into rows & columns.

(ii) Find row parity & column parity.

(iii) Design the code word in row order.

Q msg: 10110010
block size: 4 bits (given)
parity: even

msg → 10110010

c1	c2	c3	c4		
1	0	1	1	1	→ row 1
0	0	1	0	1	→ row 2
1	0	0	1	0	

→ even parity (row)

(column parity)

∴ CW = 10111 00101 10010

designed in row order.

* Size of CW in one dimensional parity

$= m + 1$ → size of check bits

* Size of CW in two dimensional parity (block size → K)

$\Rightarrow m + \underbrace{m/K + K + 1}_{\text{size of check bits}}$

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

→ m (the two message rows), → m/K (the row-parity bits 1, 1), → K (the parity row 1 0 0 1), → 1 (the last parity bit 0)

* CRC [Cyclic Redundancy Check]

(i) CRC will be used in data link layer error control mechanism.

In DLL

N/W Layer ↓

DH	NPDU	DT

→ trailer will also be added

(Here, ~~crc~~ GP will be included).

(ii) To compute CRC, sender uses polynomial (which is also called as generator polynomial)

* Msg + $\underbrace{\text{Check bits}}$ (CRC)

generated using generator polynomial.

x^4+x^3+1 / x^3+1 eg.

* Certain GP used to generate CB

The Valid Generator Poly will be:

(i) It shall be divisible by $x+1$. } cond^n.
(ii) It shouldn't be divisible by x.

$x^3+1 \longrightarrow$ GP.

Can we use this to gen. CB?

· x^2+x X (not valid)

$\hookrightarrow x(x+1)$ ∵ divisible by x.

x^3+1 → we use modulo-2 arithmatic division.

↓ binary

· $1x^3+0x^2+0x^1+1\cdot x^0$

↳ 1001

· $x+1 \longrightarrow$ 11 $(1\cdot x^1+1\cdot x^0)$

· $x \longrightarrow$ 10

⇒ Modulo 2 arithm. Div

11)1001(✓ (shud be)

10)1001((not)

$$
\begin{array}{r}
11\overline{)1001}(111 \\
\oplus 11 \\
\hline
\cancel{0}10 \\
\oplus 11 \\
\hline
\cancel{0}11 \\
\oplus 11 \\
\hline
\cancel{0}0
\end{array}
$$

rem = 0.

∴ (x^3+1) div. by $(x+1)$ ✓

$$
\begin{array}{r}
10\overline{)1001}(100 \\
10 \\
\hline
\cancel{0}00 \\
00 \\
\hline
\cancel{0}01 \\
00 \\
\hline
\cancel{0}1
\end{array}
$$

∴ rem ≠ 0.

∴ (x^3+1) not div. by x.

∴ $x^3+1 \longrightarrow$ valid Gener. Polynomial.

Key → GP

* Advantages of CRC :

(i) It can detect all single bit errors in the code word.

(ii) It can detect burst errors of length in odd no. of bits of codeWord.

(iii) It can detect burst errors of length less than or equal to degree of generator polynomial. $<= d$

(iv) It can detect burst errors of length greater than degree of generator polynomial with high probability. $> d$

eg: msg = 101010

GP: → to add CB

→ $x^3 + 1$

No. of CB depend upon degree of poly.

∴ 3 Check bits

$$\boxed{\text{Size of CW in CRC} = m + \text{degree(GP)}}$$

$\overbrace{101010}^{m}\overbrace{\_\,\_\,\_}^{cb}$: codeWord

$\underbrace{101010\_\_\_}_{9 \text{ bits in CW}}$

1. can detect error in any bit position (1st, 3rd ..., 4th)

2. It can detect 1 bit, 3 bit, 5 bit .. 9 bit error.
   1 bit: X (already detected)

3. It can detect :
   1 bit, 2 bit, 3 bit error:
   1 bit: X ; 2 bit: new ; 3 bit: X [detected in 2].

4. 4 bit, 5 bit, 6 bit, 7 bit, 8 bit, 9 bit.
   5 bit: X ; 7 bit: X ; 9 bit: X

⇒ Prob. of detecting these errors is 99%. but not 100%.

Q Which of this is valid?

(i) x

- not valid

(ii) $x^2 + x \quad = x(x+1)$

- not valid

(iii) $x+1$

- Yes, div. by $x+1$, not div by x

(iv) x^2+1

- Yes, valid.

11)101(11
11
011
11
0

10)101(10
10
001
00
1

2. Pg 3

4. x^7+x^5+1 $\quad x^3+1$ 1001

1001)10100001(10110
1001
00110
0000
01100
1001
01010
1001
00111
0000
111 → remainder
→ CRC

Msg + Checkbits ← CRC
↑ gener. by GP.

(Checkbits = CRC) [in CRC techn]
[CB = Parity] (in Parity T.)

Q. Message = 101010
GP = x^3+1

$x^3+x^2+x+1=0$
1 0 0 1

(i) Calculate CRC
(ii) Calculate Code Word using CRC Technique.

m = 101010 $\quad$ no. of CB = 3

1001)101010000(101
1001
00110
0000
1100 X.
1001
0101

1001)101010000(1011
11
1001
00111
0000
01110
1001
01110
1001
01110
1001
01110
1001
0111

rem = 111 → CRC

$$\boxed{CW = \underbrace{101010}_{m}\ \underbrace{111}_{CB}}$$

VI — trailer.

* DH | 101010111 | x^3+1

receiver rec this.

rec divides: 101010111 by x^3+1
(if rem → 0)
[correct data is reciv.]

** [If error present in trailer → we can't do nothing] → chance is 0.001%.

⇒ receiver does: GP) code word recieved (
if 0 → correct
else not correct

Everyone use a static Gen P. ← Standard.
x^6 <1500b
x^4 <6000b]

Current Networking Systems [Hardcoding req.]

decided based on length of msg.

* CHECK SUM

Key = Block Size

(i) It is used in Transport layer Error detection & Network layer if required. (some cases)

(ii) To find CheckSum:

(1) Sender breaks the msg into multiple Blocks according to the block size.
(2) Then it performs block wise addition, adds carry if any.
(3) Apply one's complement on the final value.

eg: data: $\underbrace{1111}_{b1}\underbrace{0011}_{b2}\underbrace{1100}_{b3}\underbrace{0000}_{b4}$

block size: 4 bits. (Key)

NOTE: If m not exactly div; add 0s in starting. (imp)

b1+b2

```
     1111
 ①   0011
     0010
        1
     0011  ← b1+b2
     1100  ← b3
     1111  ← b1+b2+b3
     0000  ← b4
     1111  ← b1+b2+b3+b4
```

1s

$\boxed{0000}$ ← Check Sum

Check bits

* CodeWord :

1111 0011 1100 0000 0000

$\underbrace{1111\ 0011\ 1100\ 0000}_{msg}\ \underbrace{0000}_{CheckSum}$

· Receiver Knows BlockSize (given in trailer)

PH	CW	TT

TT → block Size .

→ · Receiver receives the code Word & block Size and performs same Steps as by Sender.

(if sort → get 000 ...)

→ If the final result is 0s, then the received message is correct, else its not correct.

Q6 → Pg 4 (BS = 16 bits).

1001 0011 1001 0011

1001 1000 0100 1101

0010 1011 1110 0000

1

0010 1011 1110 0001

1101 0100 0001 1110 ← CheckSum

BS = 4

7) msg

1010	0001	0001	0001	1000
b1	b2	b3	b4	b5

1010
0001
1011 ← b1+b2
0001
1100 ← b1+b2+b3 .
0001
1101 ← b1+b2+b3+b4 .
1000
0101
\+ 1
0110

1s

⇒ 1001 ← CheckSum .

** CRC more comp. than CheckSum

- CRC:
  - node to node
  - more time
  - more complex
- CheckSum:
  - less time
  - less comp.
  - far end to end.

$\boxed{\text{CodeWord Size} = m + \text{BlockSize}}$

m ← CheckSum .

(not so imp in Gate).

* ERROR CORRECTION TECHNIQUES

(till now, check if correct/not)

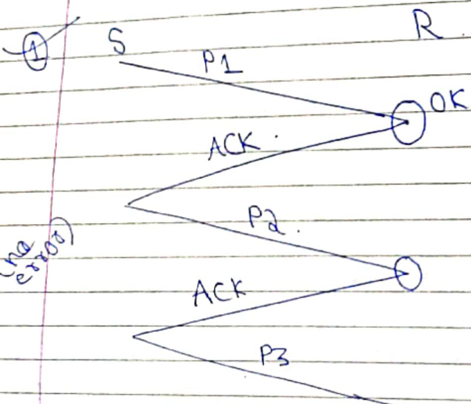
if not correct:

rec. sends negative acknowledgement, & ask for retransmission.

1. By retransmission → 6 cases
2. Hamming Code (Error detection & correction technique)

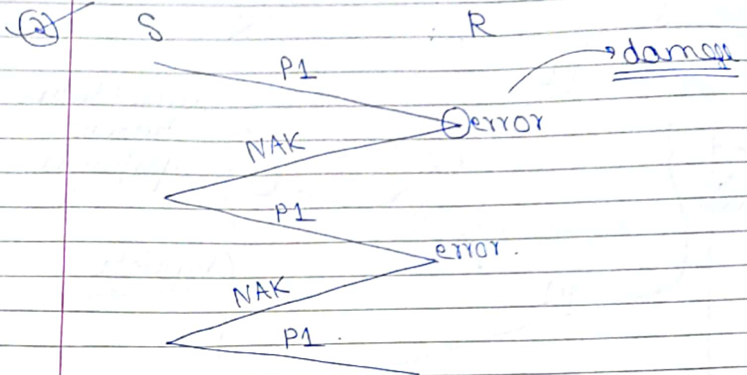
Basic Stop & Wait Protocol

1



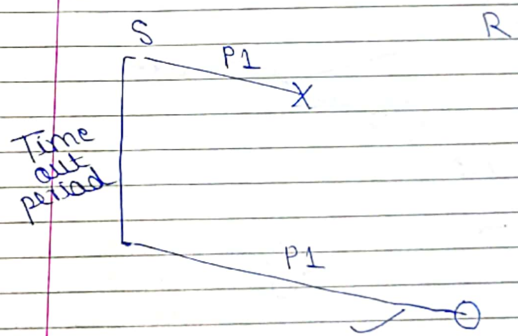
(no retrans. here)

2



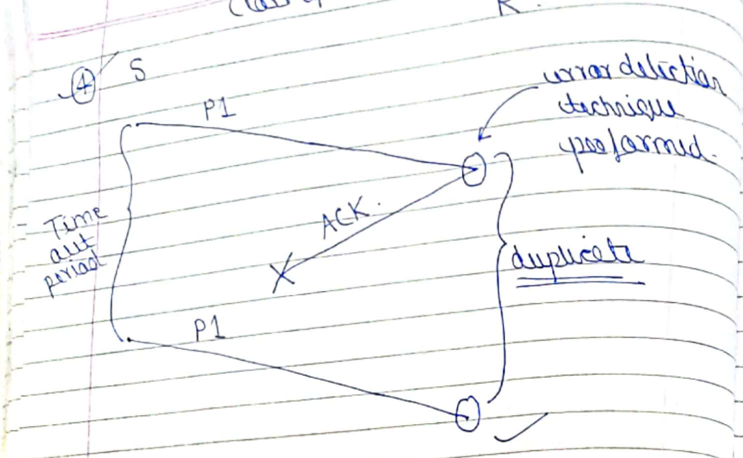
⇒ P1 retransmission will be done till P1 is success (+ve ACK). [we receive ACK]

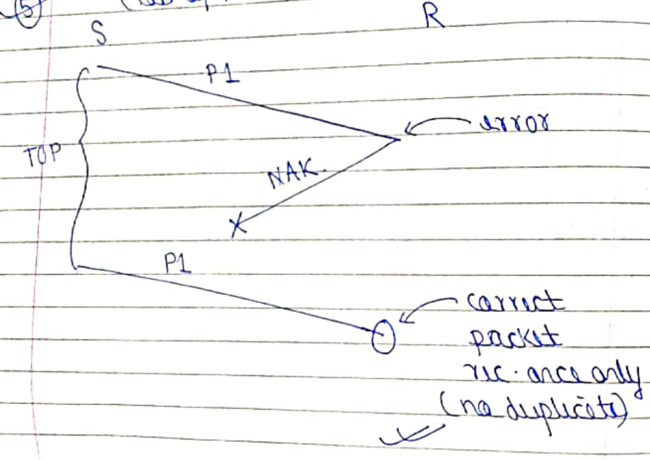
3 Loss (Sender waits till TOP, if don't rec ACK, retransmit)

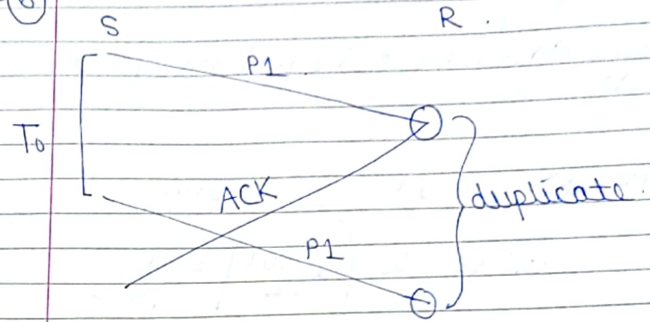


*(can't be damage... (contd...)

(loss of ACK)

4.


5. (loss of NACK)


6. (delayed ACK)


∴ duplicacy in:
- loss of acknowledge
- delay ACK

} V.I.

* HAMMING CODE (error detectn & correctn technique)

$$\boxed{m + K + 1 \leq 2^r}$$

r = no. of check bits

$K = \lceil \log_2 m + 1 \rceil$ → step up

Q. message size = 100 bits
using Hamming code find no. of check bits.

Sol. $m+k+1 \le 2^r$

step up.

where, $k = \lceil \log_2 101 \rceil = 7$.

$100+7+1 \le 2^r$

$108 \le 2^r$

$\boxed{r=7}$

⇒ 7 check bits to be added [msg + check bits]

code word size = 107 bits.

Q. msg size = 5 bits

$k = \log_2 6 \Rightarrow 3$ $k=3$

$5+3+1 \le 2^r$

$9 \le 2^r$. $r=4$

CW = 9 bits

⇒ $\underset{2^0}{\underset{1}{\_}}$ $\underset{2^1}{\underset{2}{\_}}$ $\underset{2^2}{\underset{3}{1}}$ $\underset{2^2}{\underset{4}{\_}}$ $\underset{5}{0}$ $\underset{6}{1}$ $\underset{7}{0}$ $\underset{2^3}{\underset{8}{\_}}$ $\underset{9}{1}$

(in 2^P positions ⟶ check bits)

Q. message = 10101010.

$m = 8$

$k = \lceil \log_2 9 \rceil = 4$

$8+4+1 \le 2^r$.

$13 \le 2^r$

$\boxed{r=4}$ $\boxed{CW = 12 \text{ bits}}$

CB.

⇒

1	1		1				0				
*	*	1	*	0	1	0	*	1	0	1	0
1	2	3	4	5	6	7	8	9	10	11	12
2^0 (r_0)	2^1 (r_1)		2^2 (r_2)				2^3 (r_3)				

	r_3	r_2	r_1	r_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
~~12~~	~~1~~	~~1~~	~~0~~	~~0~~

12 Values.

Check bit calculn

- r_0 : {1, 3, 5, 7, 9, 11}
  * 1 0 0 1 1

  * = 1 (even parity)

- r_1 = {2, 3, 6, 7, 10, 11}
  * 1 1 0 0 1

  * = 1

- r_2 = {4, 5, 6, 7}
  * 0 1 0

  * = 1

- r_3 = {8, 9, 10, 11, ~~12~~}
  * 1 0 1 0

  * = 0

Q. Find out code Word for the sender data

111000

using Hamming code?

Soln:

$m = 6$

$k = \log_2 7 \Rightarrow \underline{3}$

$6 + 3 + 1 \leq 2^r$

$10 \leq 2^r$

$\boxed{r = 4}$, $\boxed{CW = 10}$

CW

0	0	1	0	1	1	0	0	0	0
1	2	3	4	5	6	7	8	9	10
r_0	r_1		r_2				r_3		

	r_3	r_2	r_1	r_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0

r_3 r_2

$r_0 = \{1, 3, 5, 7, 9, 11\}$
$\Rightarrow \{*, 1, 1, 0, 0\}$

$* = 0$

$r_1 = \{2, 3, 6, 7, 10\}$
$* \; 1 \; 1 \; 0 \; 0$

$* = 0$

$r_2 = \{4, 5, 6, 7\}$ $\quad * = 0$
$r_3 = \{8, 9, 10\}$, $\quad * = 0$

* How error being detected & corrected?

Q. Assume the code word for 0010100000 for the msg 111000 is transmitted to the receiver.
How receiver will find position of the error? What is the code of correct code word?

1 2 3 4 5 6 7 8 9 10
0010100000 ← CW rec.
2^0 2^1 2^2 X 2^3

r_0: 1, 3, 5, 7, 9
$\Rightarrow$ 01100 — even

- $r_1 = \{2, 3, 6, 7, 10\}$
$\Rightarrow \{0\,1\,0\,0\,0\}$ — X (odd)
- $r_2 = \{4, 5, 6, 7\}$
$\Rightarrow \{0\,1\,0\,0\}$ — X (odd)
- $r_3 = \{8, 9, 10\}$
$\Rightarrow \{0\,0\,0\}$ — even

$\left\{ \begin{array}{l} r_1 = 2^{nd} \text{ pos} \\ r_2 = 4^{th} \text{ pos} \end{array} \right\}$ (~~add~~) → detection

$\therefore$ 6th Position → error

- We could only find single bit error after so much computation
- $\therefore$ retransmission is best till Now.

$\therefore$ Problem with Hamming code

$r_0 = \{1, 3, 5, 7, 9, \cancel{10}\}$

$\Rightarrow \{*, 1, 1, 0, 0\}$

$\underline{* = 0}$

$r_1 = \{2, 3, 6, 7, \cancel{10}\}$

$* \; 1 \; 1 \; 0 \; 0$

$\underline{* = 0}$

$r_2 = \{4, 5, 6, 7\}$ $\quad * = 0$

$r_3 = \{8, 9, \cancel{10}\}$, $\quad * = 0$

* Has error being detected & corrected?

~~Q~~ Assume the code word for the msg 0010100000 (positions 1 2 3 4 5 6 7 8 9 10) for the msg 111000 is transmitted to the receiver.

How receiver will find position of the error? What is the ~~code~~ of correct code word?

1 2 3 4 5 6 7 8 9 10

0010100000 ⟵ CW rec.

$2^0 \; 2^1 \; 2^2 \; X \; 2^3$

r_0: 1, 3, 5, 7, 9

$\Rightarrow$ 01100 — even

$r_1 = \{2, 3, 6, 7, 10\}$

$\Rightarrow$ {0 1 0 0 0} — X (odd)

$r_2 = \{4, 5, 6, 7\}$

$\Rightarrow$ {0 1 0 0} — X (odd)

$r_3 = \{8, 9, 10\}$

$\Rightarrow$ {0 0 0} — even

$\left\{ \begin{array}{l} r_1 = 2^{nd} \text{ pos} \\ r_2 = 4^{th} \text{ pos} \end{array} \right\}$ (~~sum~~) ⟶ detection

$\therefore$ 6th Position ⟶ error

- We could only find single bit error after so much computation.
- $\therefore$ retransmission is best till Now.

$\therefore$ Problem with Hamming code

* FLOW CONTROL :

If Sender capacity > receiver capacity

1. loss in data.
2. No reliability in data.

∴ To avoid this, we need flow control mechanisms.

Without flow control

Sender	receiver
BW = 1Mbps Packet length = 1000 bits	BW = 1 Kbps Packet length = 1000 bits.

No. of packets sent by sender / sec

$1 \times 10^6 b$ — 1 s

$1000 b$ — x

$10^6 / 10^3$

$x = 10^3$ packets sent/sec

No. of packets rec. rec. / s

$1P = 10^3 b$ — 1 s

∴ 1 P in 1 s

$\boxed{\text{Channel utilization} = 100\%}$

$\boxed{\text{Throughput} = 1 \text{ Mbps}}$ → used entire BW

$\boxed{\frac{BW}{PL} = \text{no. of Packets sent}}$

⇒ Sender sends 1000 P/S
receiver receives 1 P/S.

∴ 999 packets will be lost.

* Adding Flow Control :

Sender (1Mbps) — receiver (1Kbps)

1 s = 1000 Packets (segment) — 1 s = 1 PCK

send only 1 PCKT.

1. transmission time (delay) → PL/BW

$\frac{1 Kb}{1 Mbps} = \frac{10^3 \text{ bits}}{10^6 b/s}$

$\Rightarrow 10^{-3} s = 1 \text{ msec}$

remaining 999 msec ⇒ sender is idle.

Send time = 1ms
Idle/WT = 999 ms

$\boxed{\text{Channel utilization} = \frac{1 \text{ msec}}{1 s} \approx 0.1\%}$

→ how much utiliz. from total time.

Maximum Data Rate

$\boxed{\text{Throughput} = \text{Channel utilization} * BW}$

$\Rightarrow$ 0.1% of 1Mbps

$\Rightarrow$ 1 Kbps

* Sender has to wait:
Idle time ↑
Throughput ↓
Channel utilization ↓
} (with flow control)

* DELAYS

1. transmission delay } Most imp
2. propagation delay
3. processing delay } Networking devices
4. queue delay

1. Transmission Delay

Amount of time required for transmission of data is called as transmission delay

1Kb of data

1Mbps channel

$$\boxed{TD = \frac{\text{length of Packet}}{BW}}$$

$$= \frac{1\,Kb}{1\,Mbps} = \frac{10^3}{10^6} = 10^{-3}\,s = 1ms$$

2. Propagation Delay

Signal travelling time from one point to another.

d = 200m

speed = 200m/ μsec

$$PD = \frac{d}{s} = \frac{200 \times 10^{-6}}{200} \Rightarrow 1\,\mu sec$$

eg. Comp → data onto wire. (BW matters) [TD]

speed matters [PD].

length of packet doesn't matter here.

NOTE:

1. if Q asks transmission time / transmission delay:

time taken for transmission = TD

2. If Q asks:

time taken for source to dest / beginning to end of n/w / from 1 PC to another:

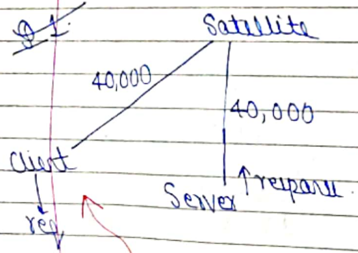
= PD

(on the Wire)

3. total time taken to reach destination.

TD + PD

~~Q1.~~ [Pg 5 → 1]



response to request =

Time taken from client to server via satellite

+

Processing delay at Server (Here, its 0)

+

return from server to client

(No TD → wireless) only PD.

$$\therefore PD = \frac{d}{s} = \frac{16{,}000\,Km}{3\times10^8\,m/s}$$

$$\Rightarrow \frac{16\times10^3\times10^3}{3\times10^8}$$

$$\Rightarrow 10^{-2}\times\frac{16}{3}$$

$$\boxed{PD = 533\ msec}$$

Q3 (Pg 5)

1024 × 768 pixels.

1 pixel → 3 bytes

(a) $\therefore \frac{2236416\ \text{bytes}\times 8}{56\times10^3} \Rightarrow 337\ s$

$$\boxed{TD = \frac{\text{length of Packet}}{BW}}$$

(b) $\frac{1024 \times 768 \times 3 \times 8}{10^6} = 18.874\,s$

$\Rightarrow 18.8 \approx 19\,s$

(c) $\frac{1024 \times 768 \times 3 \times 8}{10^7 \times 10^6}$

$\Rightarrow 1.887\,s$

(d) $0.18874\,s$

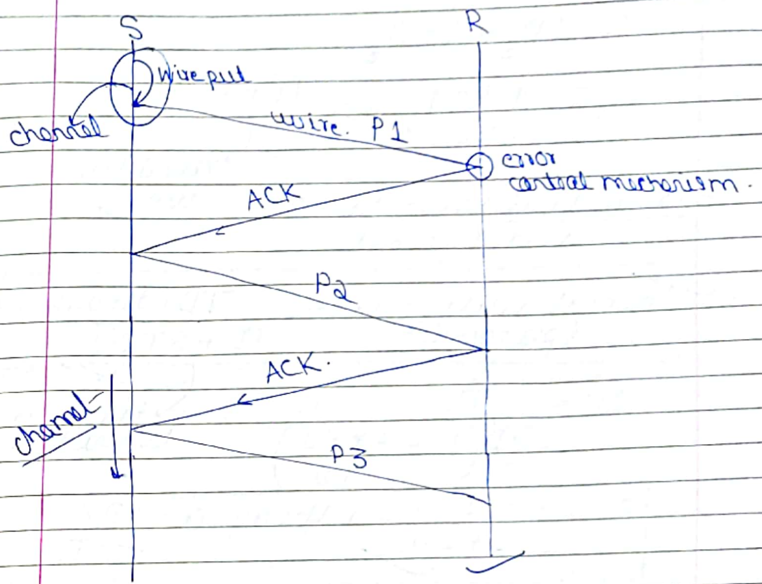
* Flow Control Mechanisms

Noiseless Channel (Theoritically) — no error

Noisy channel (Practical)

1. Stop & Wait ← related → 1. Stop & Wait ARQ
2. Sliding Window Protocol ← related → 2. Go back N Protocol
   3. Selective repeat Protocol

1. Stop AND WAIT PROTOCOL [noiseless]



Total time: (small delays ignored)

$S_{TD} + S \rightarrow R_{PD}$ / $PD_{S \rightarrow R}$

$\Rightarrow \{ TD_S + PD_{S \rightarrow R}$ (P1) $+$ Rcvr proc. delay (error cont) $+ TD_{Recv.}$ (1 byte ACK) $+ PD_{R \rightarrow S}$ (ACK travel) $\}$

[RTT → Round trip time]

$\Rightarrow$ Total time $= TD_S + PD_{S \to R} + PD_{R \to S}$

$\therefore TD_S + 2PD_S$

$\Rightarrow \underbrace{TD + RTT}_{\text{total time taken to send 1 packet}}$ ✓✓

$\boxed{RTT = 2PD}$ round trip time

$$\boxed{\Rightarrow \text{Channel utilization (per packet)} = \frac{TD \text{ (dep on BW)}}{TD + RTT}}$$

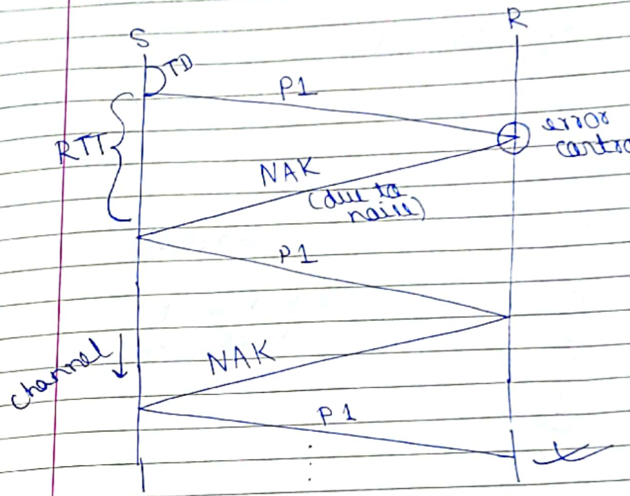
$TD + RTT$ → time taken to send 1 packet

$$CU = \frac{TD}{TD\left(1 + \frac{2 * PD}{TD}\right)}$$

$$\boxed{CU \Rightarrow \frac{1}{1+2a} \quad \text{where, } a = \frac{PD}{TD}}$$

$$* \boxed{\text{Throughput} = \text{Channel Utilizn} * BW}$$

* STOP & WAIT ARQ [Noisy channel]



$$CU = \frac{TD}{\frac{1}{1-P} * (TD + RTT)}$$

TD → always taken for Success rate.

P = error Rate.

∵ num doesn't change. [TD when success]

$$\boxed{CU = \frac{(1-P) * TD}{TD + RTT}}$$

Avg no. of transmissions per packet $*$ total time taken for transmission of 1 packet

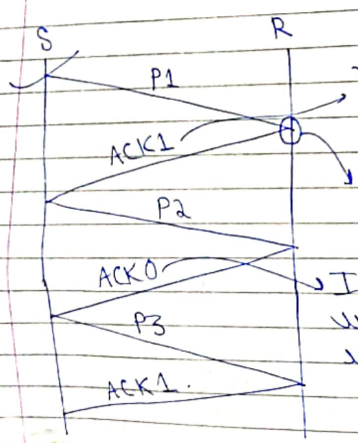
= total time

- Stop & Wait ARQ uses 2 sequence numbers (0,1) to control the errors.

→ 10 packets to be sent:

P1 - 0
P2 - 1
P3 - 0
P4 - 1
P5 - 0
P6 - 1
P7 - 0
P8 - 1
P9 - 0
P10 - 1

(Seq no to keep track) → {0,1}



→ rec rec. P (seq 0) & send (seq 1)

→ Here expecting 0.

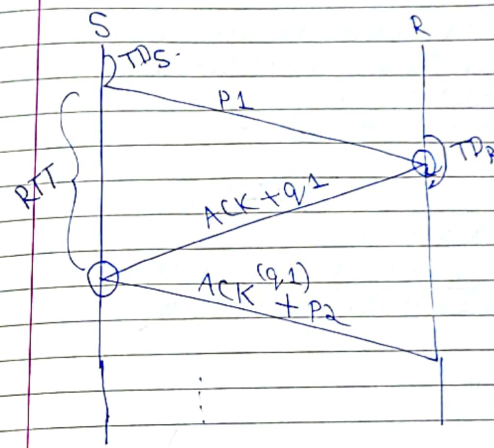
→ I have received packet with Sno 1, please send packet with Sno 0.

[Ack bit → 1 bit, seq no. is padded on it]

[NAK is enough in -ve ACK, Sno not required]
[Noiseless]

* Piggy Backing in Stop & Wait Protocol

- receiver sends some data along with ACK.



$$\boxed{CU = \frac{S_{TD} + R_{TD}}{R_{TD} + S_{TD} + RTT}}$$ → BW used in both sides.

$$\boxed{CU \Rightarrow \frac{2 * T_D}{2T_D + RTT}}$$

if, length of data is same at sender & receiver & BW also.

$$\frac{(2 * T_D)}{R_{TD} + S_{TD} + RTT}$$

Piggy Backing.
Stop & Wait ARQ

Noisy channel

* Damage Case

[PB in ARQ]

$$\boxed{CU \Rightarrow \frac{(1-P)(S_{TD}+R_{TD})}{S_{TD}+R_{TD}+RTT}}$$

How much fraction of time the channel is utilized from total time

Pg 5

VI

4. Nagraul ace req.
(when ack is received, only at that time we conclude packet is sent)

5. BW = 4Kbps.
PD = 20 msec.

$$CU = \frac{TD}{TD + RTT}$$

VI

$$50\% = \frac{L/BW}{L/BW + (2 * PD)}$$

$$0.5 = \frac{L/4Kbps}{L/4Kbps + 40\,msec}$$

$$0.5 = \frac{l/4}{l+160/4}$$

$$\frac{l}{4} \Big/ \frac{l}{4} + 40$$

$$\frac{l}{4} \Big/ \frac{l+160}{4}$$

$$\frac{l}{l+160}$$

$$\Rightarrow \frac{l}{l+80} = 0.5$$

$$\therefore l = 0.5l + 80$$

$$0.5l = 80$$

$$\boxed{l = 160\,bits}$$

6.

$$\boxed{\text{Bandwidth Delay Product} = RTT * BW}$$

VI

BW = 1 Mbps

RTT = 20 ms (1 bit takes 20 msec)

$$BDP = 20 \times 10^{-3} \times 10^6$$

$$\Rightarrow 20 \times 10^3 = 20000 \Rightarrow 2 \times 10^4 \text{ bits}$$

$$\Rightarrow 20Kb.$$

(b) Utilization percentage

$$\boxed{CU = \frac{TD}{TD + RTT}}$$

$$TD = \frac{10^3}{10^6} \Rightarrow 10^{-3} \Rightarrow 1\,msec.$$

$$CU = \cancel{10} \quad \frac{1}{21} = 4.7\%$$

if. 1 bit — 20ms.
1000 bit — 20ms (frame)
(PD doesn't depend on Length)

(c) Throughput = $\frac{4.76 \times 10^6}{100}$
$\Rightarrow 4.76 \times 10^4$

7. Piggybacked [Pg 6].

(b) CU = $\frac{2 * T_D}{2 * T_D + RTT}$

$\Rightarrow \frac{2}{2+20} = \underline{9.09\%}$

(c) Throughput = $\frac{9.09 \times 10^6}{100}$
$\Rightarrow 9.09 \times 10^4$

4. [Pg6]

Size of frame = 10^4 bits.
BW = $10 * 10^6$ bps

CU = $\frac{(1-p) * T_D}{T_D + RTT}$

$T_D = \frac{10^4}{10^7} \Rightarrow 10^{-3} = \underline{1\ msec}$

[Total delay = TD + PD]

RTT = 2 * 270 msec = 540 msec.

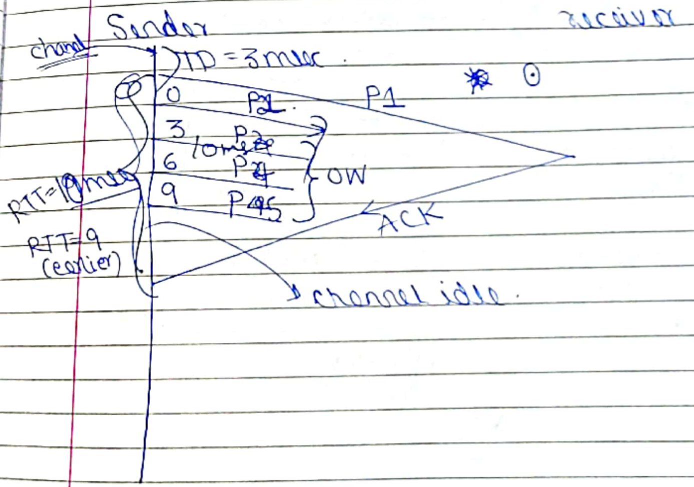
$\Rightarrow CU = \frac{(1 - 10^{-3}) * 1}{1 + 540}$

$\Rightarrow \underline{0.18\%}$

in Q: explicitly mentioned "total". → TD + PD.

TD
wire
PD.

* SLIDING WINDOW PROTOCOL



less channel utilization.

channel used for 3 msec & not used till acknwd.

Drawback of Stop & Wait

* Window: No. of packets that can be kept in the buffer before getting acknowledgement.

Optimal Window $= \left\lceil \frac{RTT}{TD} \right\rceil = \left\lceil 9/3 \right\rceil = 3$

approx 3 pack in buffer

If, RTT → 10

$\Rightarrow \left\lceil 10/3 \right\rceil = 4$ packets can be kept in buffer.

$\Rightarrow$ $\boxed{\text{Sender Window Size} = 1 + \text{Optimal Window}}$ (1)

$\boxed{\text{Sender WS} \Rightarrow 1 + \left\lceil \frac{RTT}{TD} \right\rceil}$ (2)

already sent.

→ 5

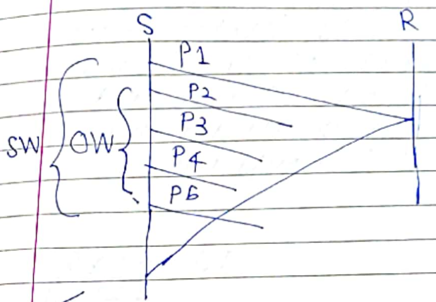
* SWS = 5 → 5 P can be kept in buffer b4 ack.

[1 2 3 4 5] 6 7 8 9 10

1 [2 3 4 5 6] 7 8 9 10

→ SW

after rec. ACK of 1



(Sender) buffer size = 5

* $\boxed{\text{no. of bits required to represent a sequence no. in SW} = \lceil \log_2 SWS \rceil}$ → Step up.

∵ 5 → min 3 bits.

$\lceil \log_2 5 \rceil = 3$

(3)

* Sender window size given =

no. of bits for seq no = $2^{\text{no. of bits for seq}}$

max utilization

Page No.: 81

$$\text{Channel Utilization in Sliding Window} = \frac{SWS * TD}{\underbrace{TD + RTT}_{\text{total time same}}}$$

VI

Q. Station A uses 32 byte packets to transmit messages to Station B using Sliding window protocol. The Round trip time b/w A & B is 80 msec and Bandwidth = 128 Kbps. What is the optimal WS that A should use?

(a) 20
(b) 40
(c) 160
(d) 320

Soln:

$TD = \frac{32 * 8}{128 * 10^3}$

$\Rightarrow \frac{2^5 * 2^3}{2^7 * 10^3}$

$\Rightarrow 2\,msec.$

$$\left[\frac{RTT}{TD}\right] = \frac{80}{32*8 / 128\times10^3}$$

2 msec

$$\Rightarrow \frac{80 \times 128 \times 10^3}{32 * 8}$$

$\therefore \underline{\underline{40}}$.

40. $L = 53$ bytes
$RTT = 60$ msec
$BW = 155$ Mbps

$$TD = \frac{53 \times 8}{155 \times 10^6}$$

$\Rightarrow 2.73 \times 10^{-6} \Rightarrow \underline{\underline{2.73\,\mu s}}$

$$\Rightarrow OW = \left[\frac{RTT}{TD}\right] = \frac{60 \times 10^{-3}}{2.73 \times 10^{-6}}$$

21978

$$\left[\frac{60\ msec}{2.73\ \mu s}\right] = \frac{60\ msec \times 10^3}{2.73\ \mu sec}$$

$\Rightarrow \underline{21{,}978}$

$\Rightarrow \underline{\underline{21K}}$ ~~bits~~ [Optimal Window]

11. $SWS = 1 + \text{Optimal } W$

$$\Rightarrow 1 + \left[\frac{RTT}{TD}\right]$$

$\Rightarrow$

$TD = l / BW$

$= \frac{64 \times 8}{1.536 \times 10^6}$

$\boxed{TD \Rightarrow 333.33\,\mu s}$

$RTT = 2 * PD$

$$\Rightarrow 2 * \frac{3000}{6\,\mu s}$$

$\Rightarrow 1000 \times 10^6\,s$.

$RTT = 2 * PD$

$\Rightarrow 2 * \frac{d}{S}$

$d = 3000 Km$

$1Km = 6 \mu sec$

$30000 — x$

$1 Km — 6$

$\therefore 3 \times 10^3 Km \times 6 \times 10^{-6}$

$\Rightarrow 18000 \mu s = PD$

$\boxed{RTT = 36000 \mu sec}$

$1 + \left[\frac{RTT}{PD}\right] = 1 + \left[\frac{36000}{333.33}\right]$

$\Rightarrow \boxed{SWS = 109}$

Sequence = $\log_2[109] = 7 bits$

Q. Length = 1000 bits

BW = 1 Mbps

3bit seq no = 8 (SWS)

$\frac{SWS * (TD + TD)}{TD + TD + RTT}$ RTT = 540 msec

$TD = \frac{1000}{10^6} \Rightarrow 10^{-3} = 1 msec$

$\Rightarrow \frac{8 * 2}{2 + 540} \Rightarrow 2.95\%$

Stop & Wait: $\frac{2TD}{2TD + RTT}$

$\Rightarrow \frac{2}{2 + 540} = 0.36\%$

Receiver window needed : Now (PTO)

noisy channel

* GO BACK-N & SELECTIVE REPEAT

Go-Back N	Selective Repeat
(i) In Go-Back-N, receiver window size is 1.	(i) In Selective repeat, SWS = RWS.
(ii) It is suitable for less Noisy channel.	(ii) It is suitable for more noisy channel (more errors coming)
(iii) It is a simple Protocol.	(iii) It is a complex Protocol
(iv) For each error, receiver discards N Packets and sender retransmits N Packets. when, N=SWS.	(iv) For each error, receiver discards only error packets and sender retransmits only error packets.
(v) No additional sub algorithm are required.	(v) Sender uses searching algorithms & receiver uses sorting algorithms

* No. of bits required for sequence no. = $\lceil \log_2 (SWS + RWS) \rceil$

* The no. of bits for seq = 2

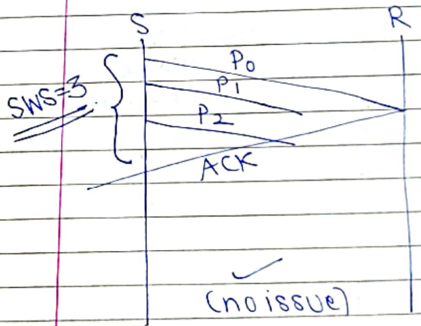
Find SWS & RWS using Go-Back N?

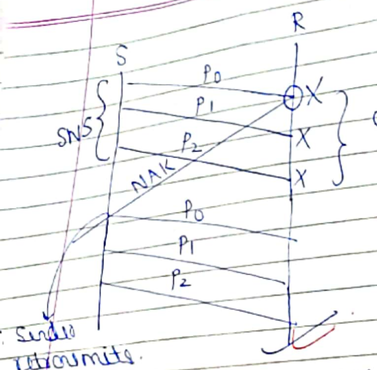
Soln.

$2^2 = 4$

SWS = 3. RWS = 1

b4, we get ACK; we can send 3 packets.

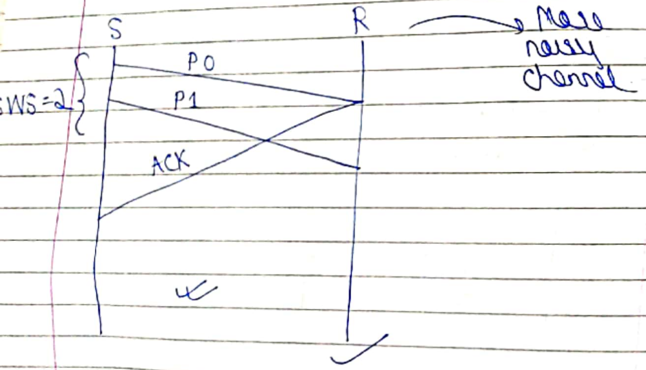


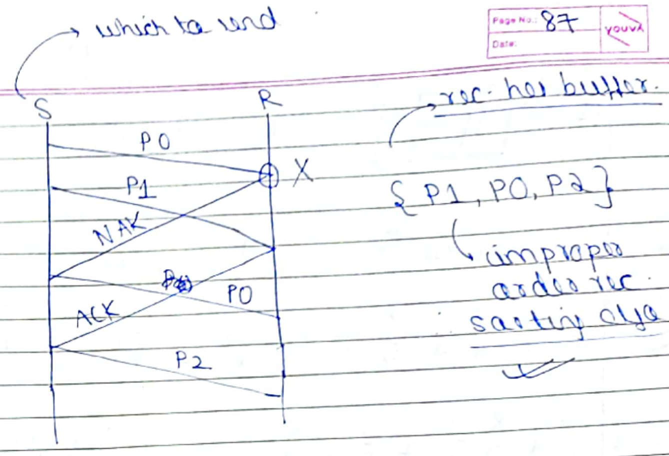


* Selective repeat

SWS = 2, RWS = 2

(Receiver has its own buffer)





→ More retransmissions done in Go Back N ∴ (less noisy)

* (Combn of GBN & SR is used in practical World) → VI

Q. Sender Window Size = 3

If every 5th Packet in the transmission is an error, using Go Back N Protocol, find out total no. of transmission for sending 10 packet.

Soln.

| 1 | 2 | 3 | 4 | 5 | 6 7 8 9 10. |

3 + 3 + (3 +   3 + 1   X

Page No. 89

1 2 3 4 5 6 7 8 9 10

1) 1 2 3 4 [5 6 7] 8 9 10.

2) 1 2 3 4 [5 6 7] 8 9 [7 10 . 8 9] 10

3) 1 2 3 4 5 6 7 5 6 7 8 9 7 8 [9 10]

4) 1 2 3 4 5 6 7 5 6 7 8 9 7 8 9 10 9 10

⇒ 18 transmissions

* Selective repeat

1 2 3 4 [5 6 7] 8 9 10

1 2 3 4 5 5 6 7 8 [9 10]

1 2 3 4 5 5 6 7 8 9 9 10

⇒ 12

1) 1 2 3 4 [5 6 7] 8 9 10

1 2 3 4 5 6 7 5 8 [9 10]

1 2 3 4 5 6 7 5 8 9 10 8 9

⇒ 12 transmissions

Q. SWS = 4

Every 6th Packet is an error. Find out total no. of transmission using both (10 packets)

1 2 3 4 5 [6 7 8 9] 10

1 2 3 4 5 6 7 8 9 6 7 [8 9 10]

1 2 3 4 5 6 7 8 9 6 7 8 9 10 8 9 10

⇒ 17 transmissions.

⇒ Selective repeat

1 2 3 4 5 [6 7 8 9] 10 .

1 2 3 4 5 6 7 8 9 6 10

11 transmissions

Pg 7

VI

15-frame seq.

13. BW = 1 Mbps
RRT = 20 msec
L = 1000

Soln.

(15 f per window)

- 15-frame seq → $\boxed{SWS = 15}$
- 15-bit seq → 15 bit seq every packet $\boxed{SWS = 2^{15}}$

∴ $\boxed{SWS = 15}$

same as SWP

$CU = SWS * \frac{TD}{TD + RTT}$

$TD = \frac{10^3}{10^6} \Rightarrow 1ms$

$\Rightarrow 15 * 1000/10^6$

$\Rightarrow 15 * \frac{1}{1+20}$

$\Rightarrow \frac{15}{21} \times 100 = 71.4\%$

efficiency (Channel Utilizn)

14. $\lceil \log_2 16 \rceil = 4$.

∵ SWS = 15 ; RWS = 1

$\log_2(15+1) \Rightarrow$ 4 bits

∴ It's a Go Back-N ARQ Protocol

15.

(a) Total (SWS + RWS) = ~~64~~ 256

SWS = 255, RWS = 1

(b) Selective repeat.

$2^7 = 128$.

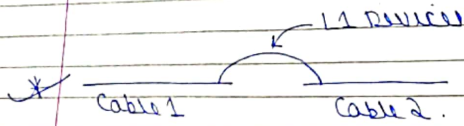
SWS = 64, RWS = 64

** BASICS OF NETWORKING DEVICES

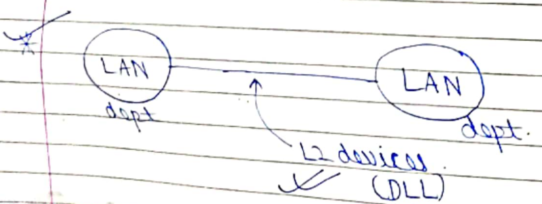
(Repeater, Hub, Switch, Bridge)

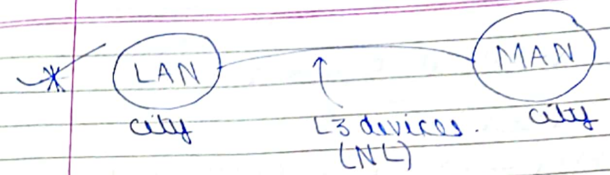
* TCP Layers [5 Layers]

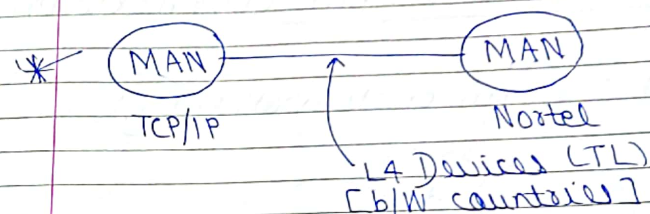
- AL
- TL → Gateways – Layer 4 devices
- NL → Routers – Layer 3 Devices
- DLL → Switch/Bridge – Layer 2 Devices
- PL → repeater/Hub – Layer 1 Device



b/w cables : L1 (PL)



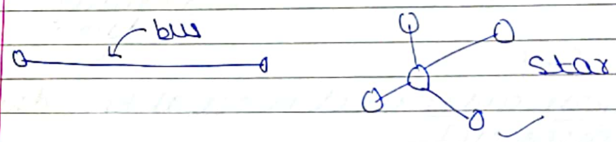


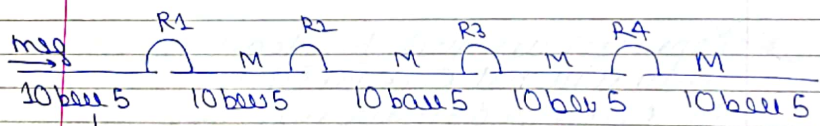


L-1 Repeater / Hub

(i) Both are Layer 1 Devices operating at Physical Layer

(ii) Repeater can be used in bus topology & Hub can be used in Star topology





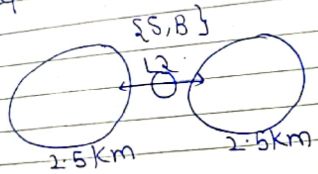
- 500 m length
- 10 mbps of BW
- Uses base band signaling.

(iii) follows a rule of 5-4-3-2-1.

∵ 1 cable → 500m.

5 cables — 2.5 Km (LAN max length)

if > 2.5 Km, create another LAN



Standard.

* 5 – 4 – 3 – 2 – 1

- 5 → 5 Cables.
- 4 → repeaters.
- 3 → Active
- 2 → Passive (Spare)
- 1 → collision domain.

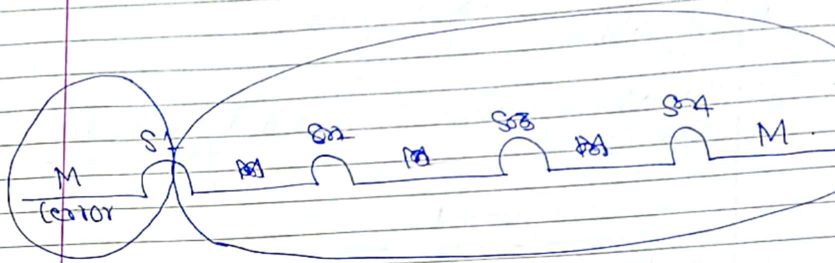
(2)

[Spare cables used in case of breakdown/ emergency]

* Repeaters are just broadcast devices. They will not check for errors. So, collision domain will be 1.
  - boost up the data & send to next.

[L2 device]

Switch → checks for errors & won't be allow to go further.



∴ 2 collision domain

∵ [Switches stop error msgs].

Q if (every repeater is replaced with Switch) → 5 CD.

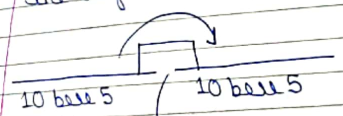
[Till how much, error will exist → CD]

8/6/19
Saturday

Lecture 3

* Repeaters → L1 Device

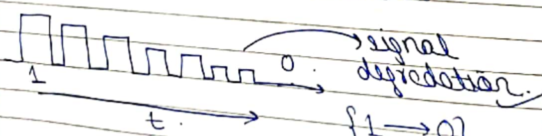
(i) Repeaters receives signals and regenerates the signals.



why regenerate?

∵ strength of signal reduces with time,

⇒ As the signal travels for some distance, the strength of the signal gets degraded and it may become zero. To avoid this scenario, repeaters will be incorporated in between cables.



(ii) Delay in repeaters:

Functionality of repeaters:

* Receive + regenerate + forward (to other ports).

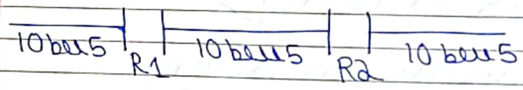
Receive → No delay

Regenerate → Processing time (generally ignore)

Forward → 1 bit delay $\left[\frac{1}{BW}\right]$

Q $\boxed{\text{Repeater delay} = \frac{1}{BW} + \text{Processing time (if mentioned in Q)}}$

Q Assume 3 10 base 5 cables connected using 2 repeaters.



BW = 10 Mbps

Speed of the signal = 200 m/µsec

(i) How long does it take for bit travelling from starting to end of network. (PD + 2 RD)

(ii) How many cables are involved in the collision?

(iii) Length of collision domain (ms).

Sol^n

(ii) Starting to end: [PD] + Repeater delay * 2

$$\left[\frac{dist}{speed} + 2 * \frac{1}{BW}\right]$$

$$\frac{1500m}{200m/\mu s} + 2 * \frac{1}{10Mbps}$$

$\Rightarrow 7.5\,\mu sec + 0.2\,\mu sec$

$\therefore$ 7.7 μsec

(iii) 3 cables

(if switch → 1 cable)

∴ 3 cables involved in collision

CD = 1

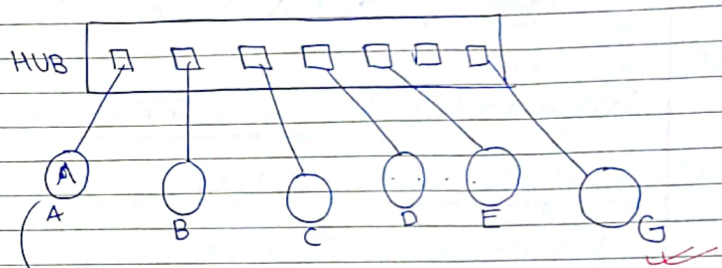
[In repeater, all cables will be involved in collision] propag. of error

(iii) Length of CD = 1500 m. (3×500) ∵ 1 Collision Domain

* HUB → L1 device (not an intelligent device) (PL)

(i) The functionality of HUB is same as the repeater only the difference in structural way.

(ii) Its a multiport Repeater.

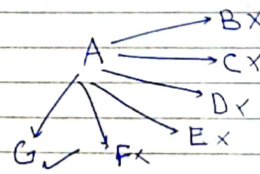


→ computer

[Each port connected to a comp.]

→ Its a broadcast device.

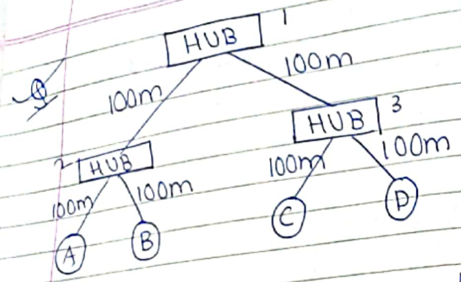
eg: A → G



→ B will discard since, its not interested to rec. the msg

(broadcasted to all).

Q.



If Data transfer happens from A → C

(i) No. of collision domains

(ii) Length of collision domains.

Soln (Hub won't look for any error control)

(i) No of CD = 1 (All Hubs)

If 1 → Switch : No. of CD = 2

(ii) Length of CD = 600m [max error can propagate to 600m]

∴ Bradcast to all. (not selective)

* Types of Hubs : (Theory) [Not so req.] (Gate)

1. Passive Hub
2. Active Hub
3. Intelligent Hub.

1. Passive Hub : just receive the signal & forward it.
2. Active Hub : will also regenerate the signal. [rec, regen, forw]
3. Intelligent Hub : It will amplify & also monitor traffic control.

NOTE : No error control mechanism in L1 Device

MAC addr

L2] BRIDGES/ SWITCH [DLL]

(i) Both are layer 2 Devices. They are Store and forward devices.

(ii) They can decrease collision domain, by separating every segment. (Length (CD))

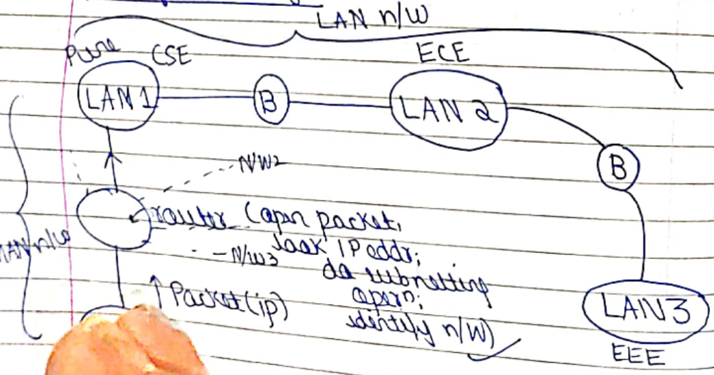
N/W ⟶ ip Address (Router)

DLL ⟶ MAC address.

∴ Bridge/Switch use MAC

(iii) These devices will be using MAC Address for communication b/w devices.

* Delay in Bridge (i) >2.5 Kms, create 1 mass LAN).



* i/p ⟶ MAC
ARP/ RARP prot.

- MAC add. used to identify comp. (48 bit)
- then Port no (TCP header) identify process.
- i/p Header identify n/w.

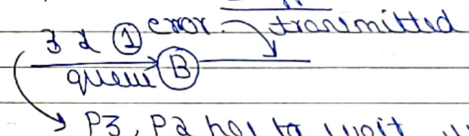
* Delay in Bridge/ Switch:

Delay

(i) It receives complete Packet ← [No delay
(ii) It will check for errors. ← [Process. d
(iii) Packet wait in the queue ← [Queue delay (Avg Q Delay

(Hub/ repeater ⟶ No memory) (buffer)

But, L2 has buffer.



→ P3, P2 has to wait, while P1 is undergoing error control.

(iv) Forward the Packet onto the wire. [Trans Delay

V.I.

$\therefore$ Delay of Bridge/Switch = {Processing delay + Avg Queue Delay + Transmission delay}

B1 B2

LAN 1	LAN 2	LAN 3
1 Km	1.5 Km	2 Km
1 Mbps	10 Mbps	100 Mbps
200m/µsec	200m/µsec	200m/µsec

Packet length = 10,000 bits.

(i) What is end to end delay? [from starting to end of network]

(ii) No. of collision domains.

(iii) Length of the collision domain in ms.

Soln

Transmission delay $\because$ PD & QD X

(i) $\frac{10^4}{10 \times 10^6} \Rightarrow 10^{-3} = 1$ msec (B1)

$\frac{}{10^6} \Rightarrow 10^{-4} \Rightarrow 0.1$ msec (B2)

$\Rightarrow \frac{4.5 \times 10^3}{200}$

$\Rightarrow \frac{4.5 \times 10^3}{2 \times 10^2}$

$2.25 \Rightarrow 22.5$ µsec (PD)

$\Rightarrow (22.5 \times 10^{-6}) + (1.1 \times 10^{-3})$

$\Rightarrow 0.0225 \times 10^3 \times 10^{-6}$

$\Rightarrow 1.1225$ ms $\times 10^{-3} \Rightarrow 1122.5$ µsec

(ii) $\therefore$ [PD + Delay at B1 + Delay at B2]

B1 B2

d/s d/s

$= \frac{10^4}{200}$ $\quad \frac{10^4}{10^7}$ $\quad \Rightarrow \frac{1500}{200}$ $\quad$ 0.1 msec

$\Rightarrow 5$ µsec $\quad \Rightarrow 1000$ µsec $\quad \Rightarrow 7.5$ µsec

$\frac{2000}{200}$

$\Rightarrow 10$ µs

(ii) No. of CD = 3

(iii) Length of CD $\Rightarrow$ max{L1, L2, L3}

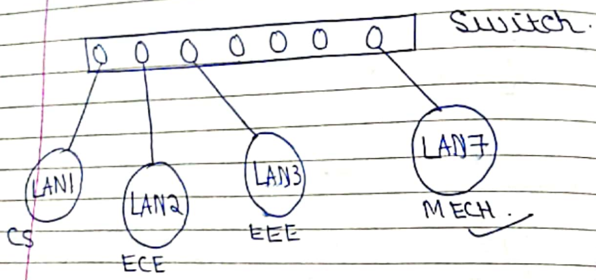
$\Rightarrow$ 2 Km

$\therefore$ max{CD1, CD2, CD3} ⇒ 2 Km

[max length where error can spread]

* SWITCH : → L2 Device

(i) Switch is a multiPort bridge.



Router → Switch → Hub
(subnet) (recognise LAN using MAC) [ranges] (broadcast to all PCs)

(ii) Operation of Switch is same as Bridge.
It is useful in Star Topology

* There are 2 types of Switch :
(i) Store & Forward Switch → [by default]
(ii) Cut through Switch.

Store, error check & forward

(i) ⇒ Stores the data in the buffer [queue], check for errors & forward the data → [Avg Q delay + Proc. D + Transm. D]

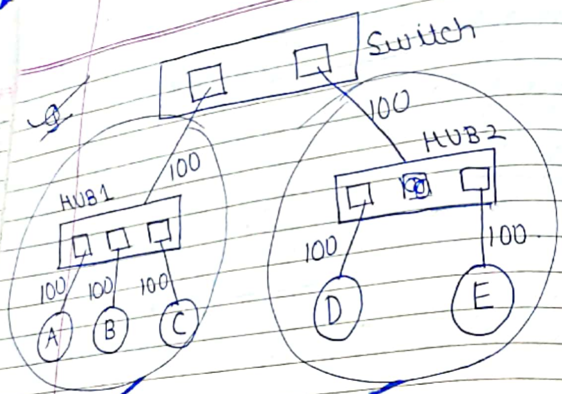
(ii) Cut through Switch.
→ look only for header (only dest. add)
Trailer → GP. (CRC) → DLL
$\therefore$ [No Process delay].
→ only Transmission Delay.

Trailer	Payload	Header

→ Cut through

[will not look into trailer].
$\therefore$ No error control mechanism

→ want be only noise in my channel.
→ & I want my data fast.



if Data transfer: A → D.

(i) No of collision domains? → 2

(ii) Length of collision domain?

max {400, 300}

⇒ 400 m

******

* SOCKETS, TCP/IP, UDP

AL —— AL Protocols
TL —— TCP Prot / UDP prot.
NL —— IP Prot.
DLL —— ethernet Prot.
PL → encoding/multiplexing/encryption.

M
↓
M + AH = APDU
↓
APDU + TH = TPDU
↓
TPDU + NH = NPDU

⇒ TCP, IP quite similar (Headers)

* IP Header

then will go to TCP Header

[N/W funct^n]

1. routing
2. frag.
3. congestion cont.
4. logical add.
5. Source to Dest.

all in IPH.

(ii) IP Header is 20 bytes fixed.
+ Same variable [padding]

1W ——— 4 bytes
2W ——— 4 bytes.
3W ——— 4 bytes
4W ——— 4 bytes
5W ——— 4 bytes

$\boxed{1W = 4 \text{ bytes}}$

∴ 1W ⟶ 32 bits (each bit has a sign)

IP Header

4	4	8	16				
00-03	04-07	08-15	16	17	18	19-31	
Version (4)	IHL (4)	TOS (8)	Total Length (16)				W1
IP identification (IP id) (16)			R (1)	DF (1)	MF (1)	fragment offset (13)	W2
TTL (8)	Protocol (8)	Header checksum (16)					W3
Source IP Address							W4
Destination IP Address							W5
Options & Padding (if required)							

IHL: Internet Header Length.
TOS: Type of Service.
DF: Dont fragment
MF: More fragments to follow.
R: reserved
TTL: time to live

⇒ NL ⟶ ip Header

(ii) Version: Version of IP, in 4 bits represented

* IETF
- Intern. Eng. Task force.
- they release RFCs (res. paper).
- they set up IP Header thinking about future

Prev: IPV4, Now ⟶ IPV6.
only 10% systems converted to IPV6.

0100 (32 bit IP)

if received by IPV6 comp: [128 bit IP]
(IPV4 ⟶ IPV6)
[remain incompatib.]

· identify the version of packet sent by Source, so as to remove incompatibility and convert the current IP Address to required IP Address.

(ii) IHL → VI [TCP also]

→ The length of the Header in Words. (Internet Header Length)

by default 5 Words → 0101. (5+)

Q IHL is represented by 0110. The no of bytes in option / padding

0110 → 6 words.

24 bytes in IHL.

Actual Header: 20 bytes (Normal)

⇒ ∴ 4 Bytes added in Options / Padding.

* IHL = 1000

No. of bytes / words in Options / Padding

∴ 3 words → ~~20~~ 12 bytes in Padding

(iii) Type of Service (TOS)

· Consist of 8 bits

- first 3 bits (ignored)
- 4 bits
- last 1 bit (reserved)

4 bits:

8 { 1000 : Minimize delay
4 { 0100 : Maximize throughput
2 { 0010 : Maximize reliability
1 { 0001 : Minimize monetary cost.

· TOS field indicates the service to be used by down Layer (DLL).

1000 → Upper Layer is asking to minimize the delay to lower Layer.

0010 → No error shud be there, (may take time).

Page No. 115

Q. A Physical layer can transmit maximum of 660 bytes of data. Suppose, IP layer wants to send 1460 bytes of data. Indicate:

- Total length
- IP id
- DF
- MF
- Fragment offset.

Soln.

1460 bytes to be sent.

660 [ ] PL X. (down layer)

FO=0

1460 (datagram)

(FO) $640/8 = 80$

(FO) $1280/8 = 160$

TL=660	TL=660	TL=200
640 | 20	20 | 640	20 | 180
660	660	200
MF = 1	1	0
DF = 0	0	

: fragments.

* IP identification

How all frag. from same Datagram
→ All these frag from same Datagram will be having same IP id.

* MF

- More fragments.
- (MF=0) → Last fragment.

* DF

(DF=1) → when we aren't fragmenting [rec. full DG]

* Total length (including Header)
[in Binary]

* Fragment offset

- How to identify order of fragments?
- FO of Starting fragment = 0.

FO of 2nd F = $640/8 = 80$

data completed in prev frag.

∴ 16 bit rep. in 13 bit so divid. by 8

* (Header checksum) = 16 bit
  optional here (in N/W Layer)
  [in TL ⟶ must]

* Protocol
- this field tells which protocol has been used (TCP/UDP/ICMP...)
- (open packet, find UDP, no error)
- warm to Down Layer

* TTL ⟶ will be discussed in routing
  - to avoid looping of packets in Network.
  - time to live.

* Source ip Add } logical Add.
  Dest ip Add } ⟶ 32 bit.

Q. An IP Packet has arrived with first 8 bits as shown below.

0100 0010.

The destination discards the packet.

Soln.
IPV4. IHL = 2 words (not 5)
⟹ 8 bytes only.
∴ Discard

$\boxed{\text{min 5 Words}}$

if: 0100 0100
4 W
[discard]

Q. Suppose an IP Datagram containing 256 bytes of data is divided into 2 Fragments; each containing 128 bytes of data.
Fill the following table for F1 & F2:

Header field	Datagram	frag 1	frag 2
Header length	5W	5 W	5 W
Total length	276	148	148
ID	3	3	3
MF	0	1	0
Frag offset	0	0	128/8 = 16

[20 | 256 bytes]

[20 | 128] F1 → 148

[20 | 128] F2 → 148

Q. Suppose a TCP message that contains 2048 bytes of data and 20 bytes of TCP Header is sent to IP layer for delivery across 2 Networks of internet. The first Network uses MTU of 1024 bytes & Second Network uses MTU of 512 bytes. For each network, fill the above Table consisting of fragments.

N1

Header field	Datagram	F1	F2	F3
Header length	5	5	5	5
TL	2088	1024	1024	88
ID	10	10	10	10
MF	0	1	1	0
FO	0	0	125	250
DF	0	0	0	0

N1

TPDU = 2068 bytes

[20 | 2068]

MTU = 1024 bytes

F1 [20 | 1004] → 1024

F2 [20 | 1004] → 1024

F3 [20 | 60] → 80

---

* Frag. happening in N/W layer.

TPDU = 2068 bytes

coming to IP

[20 | 2068] ip Header.

[20 | 1000] → 1020

[20 | 1000] → 1020

[20 | 68] → 88

TC

FO = 0   FO = 1004/8 ∵ decimal

⇒ 1000/8   125.5 → 125.

MF	1	1	0
FO	0	1000/8 = 125	2000/8 = 250
IP id	10	10	10

N2 TPDU = 2068

2068 → F1 [20 | 488], F2 [20 | 488], F3 [20 | 488], F4 [20 | 488], F5 [20 | 116]

	F1	F2	F3	F4	F5
TL =	508	508	508	508	136
ID =	10	10	10	10	10
MF =	1	1	1	1	0
FO =	0	61	122	183	244

Q. Consider a 2400 byte datagram into a link that has MTU of 700 bytes. IP identification = 422.

(i) How many fragments are generated? → 4 Fragments

(ii) What are the values of FO & MF for each fragment.

Data = 2380

Data → F1 [20 | 680], F2 [20 | 680], F3 [20 | 680], F4 [20 | 3~~8~~40]

	F1	F2	F3	F4
TL =	700	700	700	360
ID =	422	422	422	422
MF =	1	1	1	0
FO =	0	85	170	

360 byte msg

Q. In an IPV4 DataGram, the MF = 0.
The value HLen = 10 (HL).
The value of TL = 400 → payload covered
The FO value = 300

(i) The position of the datagram
(ii) The sequence no. of 1st & last bytes of Payload are:

(a) last fragment, 2400 & 2789.
(b) first fragment, 2400 & 2759
(c) last fragment, 2400 & 2759.
(d) Middle fragment, 300 & 689.

V.I.

Soln.

TL = 380/~~200~~

10W = 40 Bytes (H) [20 B - option].

⇒ [360 | 40]

360 → data.

payload = 360.

300 × 8 = 2400 (data will start from)

2400 + 360 = 2760 − 1 = 2759 (starts from 0).

* TCP PROTOCOL

(i) TCP is a connection oriented, byte oriented Protocol which gives maximum reliability of data.

(ii) It is a point to point protocol.

(iii) TCP uses cumulative acknowledgements for representing reliability of data.

(iv) TCP uses 3 way Handshake which further uses full duplex communication and piggy backing.

(v) TCP uses flow control & error control mechanisms for maximising reliability of data.

(vi) TCP Protocol was created so that different computers / computer N/WS can have a common language to communicate with each other.

9/6/19 Sunday

Lecture 4

Pg 5 → Q2 : (Flow Control).

Q) Speed = $\frac{2}{3} \times 3 \times 10^8$ m/s ⇒ 2×10^8 m/s

⇒ 200 m/ μs

← 100 m → Cable (100m)

100 bit.

∴ Length of each bit = 1m

(ii) BW = 10 Mbps

1 s —— 10 Mb ~~flowing on wire~~

1 μs —— 200 m (travel)

1 bit delay = $\frac{1}{10\ Mbps}$ ⇒ 0.1 μs

How much time 1 bit takes to move further?

1 μs = 200 m.

(1 bit) 0.1 μs = x.

x = 20 m

Length of 1 bit in n/w.

b) $1\,\mu s = 200\,m$
$0.65\,\mu s = x$

1 bit delay $= \frac{1}{1.536 \times 10^6}$
$\Rightarrow 0.65\,\mu s$

$x = 130\,m$ — length of 1 bit

$\therefore$ No of bits $= 768$

$\because$ 10,000 m of cable
$\therefore$ No of bits $= \frac{10{,}000}{130} \approx 768$

$\boxed{\text{No of bits} = \frac{\text{cable length}}{\text{bit length}}}$ VI

* TCP HEADER → 20 bytes.

|— 1 byte —||— 1 byte —||— 1 byte —||— 1 byte —|

HLEN = 5 words → IHL (IP header)

$\boxed{\text{Each W} \longrightarrow 4\text{ Bytes}}$

TCP Header :

HRCRCU

|— 1 byte —||— 1 byte —||— 1 byte —||— 1 byte —|

W1:	Source Port no (16)			Dest Port no (16)
W2:	(1500) Sequence no (32)			
W3:	Acknowledgement no (32) [1600]			
W4:	HLEN (4)	res (3)	Control Bits (9)	receiver WS (16)
W5:	checksum (16)			Urgent Pointer (16)
	Options & Padding (0–40 bytes)			

i) 2 bit seq no → 4 combⁿ
3 bit seq no → 8 combⁿ

$\therefore$ 16 bit Port no : → 00 ··· 0 … 11 ···· 1 (16) } 2^{16} Ports.

$\therefore$ [0 to 65,535 Ports] ..
choose any port out of this range.

VI — divided

1) 0 to 1023 port : Special purpose (AL protocols)
HTTP → P. 80.
FTP → P. 21.

2) 1024 to 49151 port : Unused. (in IPv6 may be used)

3) 49152 to 65535 ports : Normal uses.

M
AL AH+M → APDU
TL TH+APDU → TPDU.
NL NH+TPDU → NPDU.
DLL

in TL, we add Port no.(TH)
+
in NL, we add IP address (IP H)

Port no + IP addr → Socket.

* Source   Dest

Pno + IP   Pno + IP.

Virtual circuit/ connection (b/w S & D)
↓
Socket

⇒ 80+ 172.41.72.50. (Src)
↕ TCP connection
80+ 183 41.72.50 (Dest).

if 1 TCP conn is already established on the same set of par. another connect can't be established. Close conn first.

Http to Http req.

Page No: 127

⇒ comm to diff. (Virtual circuit)

80+172.41.72.50 —TCP conn→ 80+193.41.72.50.

TCP conn.

50120 + 172.41.72.50 (new port)

Q Pg 9 → Q1.

1. B/W same set of ports, 1 TCP conn is possible.

* SEQUENCE NUMBER → W-2

(1) TCP is a byte oriented Protocol.

Byte oriented

S
1000 bytes.

R receives 120 bytes of data (capacity).

1, 2, 3, 10

S.A.

| 20 | 100 | | 20 | 100 | .. | 20 | 100 | ... | 20 | 100 |

1599 1500 (SA of Payload)   1699 1600   1799 1700   2499 2400

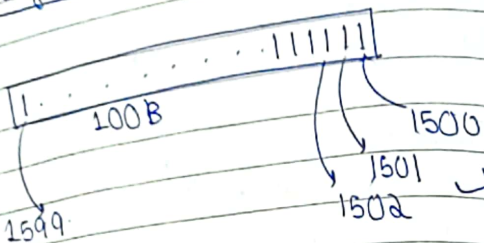
120

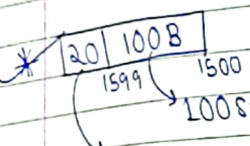
→ 1500 stored in Header [Seq no]

Segmentation

∵ TCP is byte oriented.
(Every byte → Seq no)

Seq A. (100 B of data)



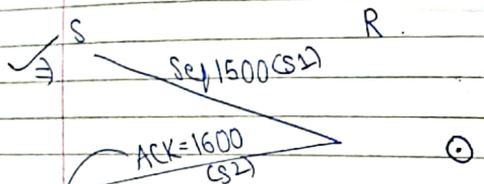


100 seq no.

seq no (H) ⟶ 1500

TL = 140 B — NL
120 B — Payload (NL) = TPDU
100 B — Payload (TL)

Q



at Down layer calc. 100 B (calculate) [TL].

next byte to be transmitted by the sender.

Q NL Taken length = 260 Bytes. → NPDU = 26[...]

AL
TL
NL

Payload of AL = 2[...] (msg)

* HLen : Length of the Header in Words (Same as IHL (NL)).

Q HLEN = 1100 ⇒ 12 W
Size of option in Words & Bytes

Soln:
7 W ⇒ 28 Bytes

* Control Bits:

(i) control bits are of 9 bits.

9 bits are:

1) SYN
2) FIN
3) ACK
4) PSH
5) RST
6) URG
7) }
8) } unused.
9) }

each byte in sync

Page No. 130

* TCP is conn. oriented & byte oriented Pr.

{ 1. establish connection
{ 2. Transfer the data
{ 3. Close the connection

The data will be present in TCP Header.

P.Q How to estab. connection?

Client — cloud (init) — Server.

Client | Server.

SYN=1, ACK=0 — at start zero — establish connect.

SYN=1, ACK=1 — ACK control Bit (not No.)

ACK=1, SYN=0

connection establishment Phase

established at client

3 Way Handshaking.

conn established at server

[SYN=1; I want to establish a conn].

Page No. 131

is data sent? ✗ — send data from seq no x (own)

P.Q client | Server

dummy

seq no = x
SYN=1, ACK=0.

[Syn RCVD State]

[Syn Sent State]

ACK=1, SYN=1,
ack no ~~seq no~~ = x+1, → dummy (send x+1)
seq no = y → ~~you send~~ seq no y [send its own seq no y].

ACK=1, SYN=0,
Seq no = x+1.
ACK no = y+1

[conn established at server]

[conn established at client]

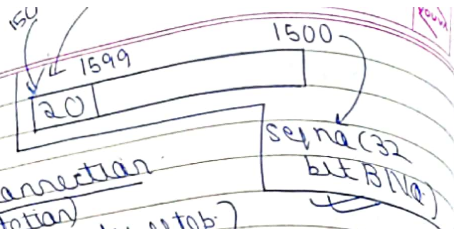
no data exchange

SYN=1, → establ. connection.

4 States.

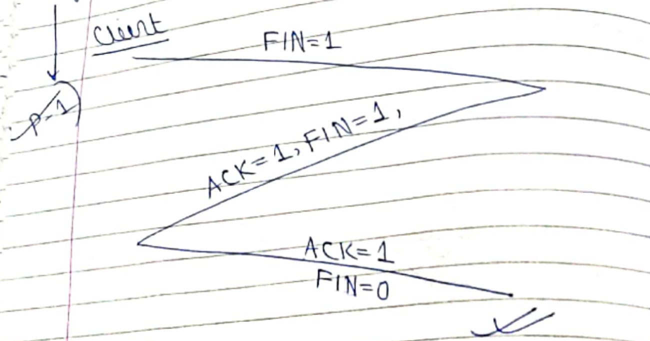
ACK → 1 bit [control bit]

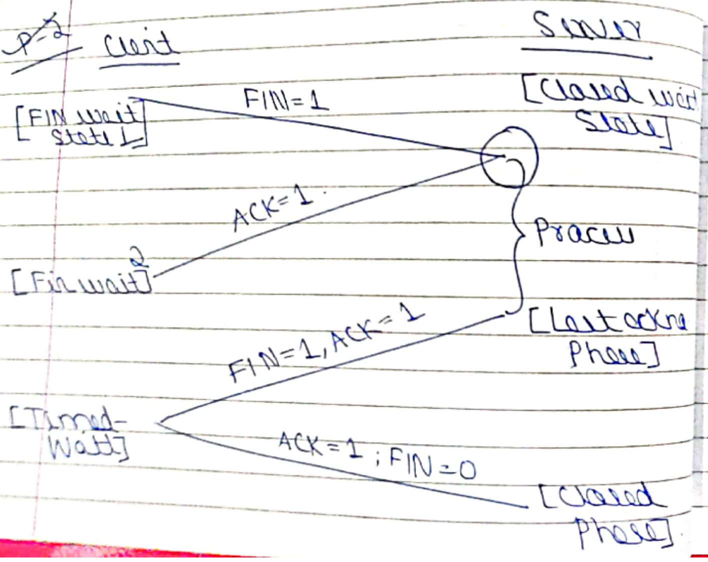
ACK no → 32 bit. [32 binary digit]



2) FIN

→ closing the connection
(types of implementation)
(When already estab.)





[If Server is busy proc. some data & receives FIN, → ~~FIN=1~~ . ACK=1 .
then finish process
& FIN=1]

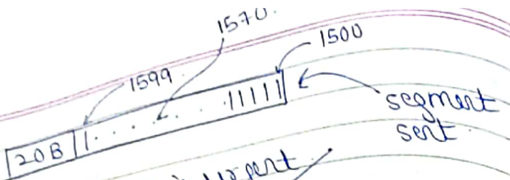
States are updated to see the actual path of the closing process → identify where it's going wrong.

Q Why we req States?

S/W → H/W.
divided into module.
(for every module, H/W node noti)

* RST → Reset the connection.
· when its an abnormal state.
· fiber has cut, alarm rings.
· Noise ↑∞

* PSH
URG } → URG = 1 .
(there is some urgent data to be processed).



data from 1570 is urgent

* URG-1, from where? → look into urgent pointer. [1570]

If U → 1550 - 1570 [range]

Q Data has to be urgently processed from 1550th byte to 1570th byte.

S1: URG = 1

S2: URG Pointer = 1550

S3: Options = Add and Urgent Pointer ⇒ 1570

S4: HLEN would be changed acc. to size of UP.

* PSH

I want to send 1 byte of data. [Payload]

AL	21B
TL	41B
NL	61B
DLL	81B
PL	101B

Wire (101B) → 1B Payload, 100B overhead

$$\text{Overhead \%} = \frac{100}{101} \times 100 = 99.1\%$$

$$\text{Payload \%} = \frac{1}{101} \Rightarrow 0.9\%$$

Silly Window Syndrome.

- To avoid this.

[Nagle's Algorithm] is used. → to reduce the overhead.

→ If the payload is less than expected percentage of data, stop transmission.

→ Wait till the Physical layer gets sufficient percentage of Payload.

→ Then, start transferring the data.

[expected → decided (50%)]

0.999% ↓ too less.

wait till some more data has come

1B + 1 + 10 + 10 + 40 + 60

no transmit

now send.

Q. "Hi" too less,

Down Layers won't allow.

(PSH=1) → push this imp data to appl^n (surpassing Nagle's Algorithm).

NOTE:

65,535 ... 0

| 20 | | → S1

max Segment Size = 65,536

65,536

| 20 | | → S2.

In 1 Segment, Max ~~data~~ Size = 65,536 Bytes.

(2) → Pg 9.

Q. The maximum Payload of TCP

65,495.

Max Segment Size = 65,536.

Payload = 65,495

$\underset{TL}{65{,}536 - 20} - \underset{AL}{20} = 65{,}496$ B.

$\underbrace{0 \text{ to } 65{,}495}_{\text{Payload}}$

Q5 → Pg 10.

	URG	ACK	PSH	RST	SYN	FIN
a)	0	0	0	0	0	0
b)	0	0	0	0	0	1
c)	0	1	0	0	0	1
d)	0	0	0	0	1	0
e)	0	1	0	0	1	0.

b) → connection closure request from client.

c) → ACK from connect^n closure from Server. (last ack phase).

(d) connect^n establishment req from client.

1097
1000

(c) Ack for conn establishment from server.

(a) client — Server

seq no = x
SYN = 1, ACK = 0

SYN = 1, ACK = 1,
ack no = $x+1$
seq no = y

ACK = 1, SYN = 0
ACK no = $y+1$, Seq no = $x+1$ → dummy data

ACK no = $x+2$
Seq no = $y+1$, ACK = 1,

ACK = $y+N-1$
Seq no = $x+2$

N = Payload

if seq no = 1001
ack = 1101
$y+N-1$

$y+N-1 = 1100$
$y+N = 1101$
$y = 1101-100$
$= 1001$

every where, same, ACK = 1
∵ [full duplex conn.]

If all 0s → invalid Packet
(a)

Q6 Host A = 3 Prog
Host B = 2 prog.

Host A — Host B

p, q, r — a, b

6 connections

Q7 efficiency at TCP Level.

40B → Payload

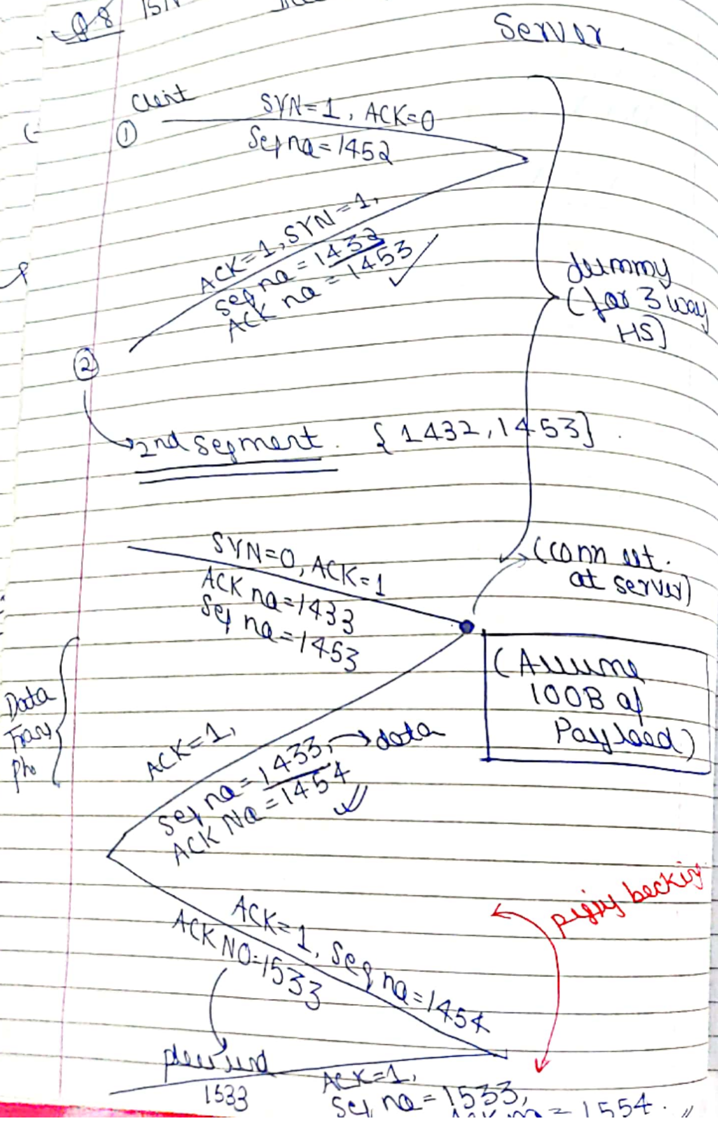
| 20 | 40 |

TCP Header

$e = \frac{40}{60} \times 100$
$\Rightarrow 66.6\%$

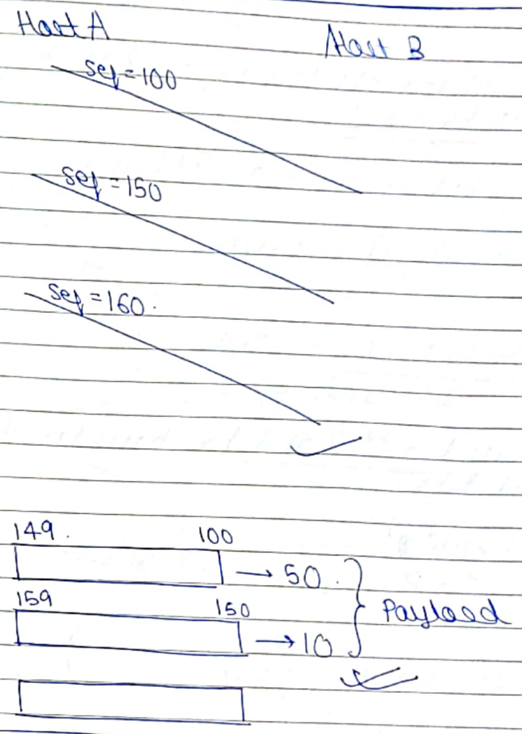
Payload %

Q8 ISN → Initial Segment no. / Init. Sequence no.



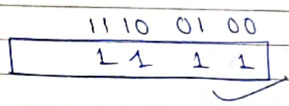
2nd segment {1432, 1453}

Q9.



Q10 → VVIP

2 bit seq no → {00, 01, 10, 11} → 4 bytes ending



for 3 bit seq no → 8 bytes will be send

[Completed all the seq no & starting from 1st seq no.]

for 32 bit, → 2^{32} bytes

000…0 – 1 Byte
⋮
111…1 – 1 Byte

2^{32} Bytes.

$\boxed{\text{VI}}$

To send 2^{32} B of data, how long does it take?

⇒ 45×10^6 b —— 1 s
2^{32} B —— x

1 bit delay = $1/BW$

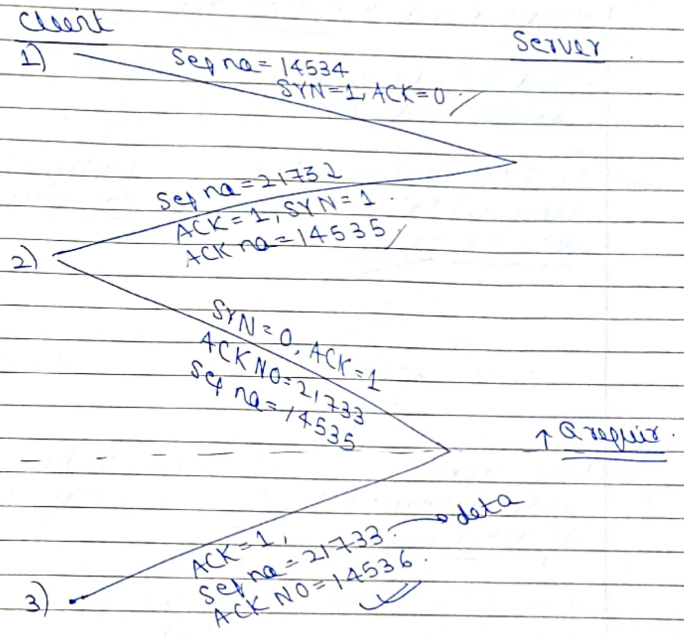
1 Byte delay = 1 Byte$/BW$

2^{32} Byte delay = 2^{32} Byte$/BW$.

$$\Rightarrow \frac{2^{32} * 8}{45\text{Mbps}} \Rightarrow 763.55 \approx 12\text{ min}$$

Q. 8 bit seq no.
If I have completely used 8 bit seq no.
→ We have send 256 Bytes.

Q.11. Pg 11



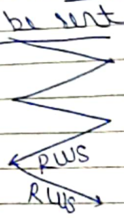
* Flow Control in TCP

Receiver window Size

1. Parties (client & Server) will come to know about their capacities.

(send RWS in header, then Sender will send accor to rec capacity)

2. In RWS explains the other party (sender/receiver) to understand the capacity of data to be sent.



3. Few Parameters may be required for exactly calculating the ~~size~~ capacity of other party.

∴ [It's a full duplex comm with piggy backed].

* Formulas:

[error & flow control mechanisms are same]

Receiver will use two control variables for calculating window size:

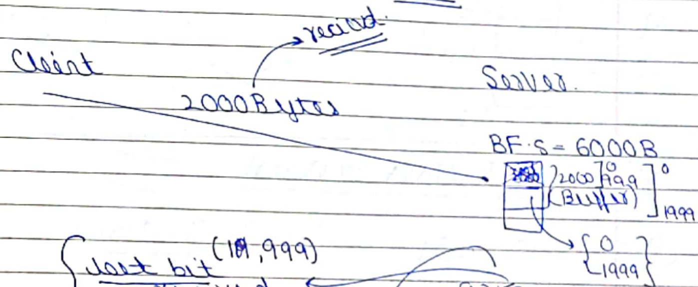
(1) last byte received (A)

(2) last byte read. (B)

when $A \neq B$.

$$\boxed{\text{Receiver Window Size} = \text{Buffer Size} - [(A) - (B)]}$$

* In most of the cases, $A = B$.



last bit received (19,999)

last bit read (999)

(A) LBr = 19,99.

But, read only till 999 (B)

(processed till 999)

$BS - [(A) - (B)]$

$\therefore BS = 6000 - 1999 - 999$.

$\Rightarrow 5000$

(send only 5000 max.)

(If got ack for 1000 bytes)

(∴ Buffer reduced by 1000.

⇒ SWS = At Sender side

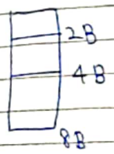
Last Byte sent = 1999.] 1000 kept in buffer rec.

Last Byte ACK = 999.

$$\boxed{SWS = RWS - (\text{Last Byte Sent} - \text{Last Byte ACK})}$$

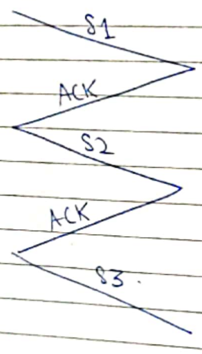
if NAK, maybe we need to retransmit

4.



Q4 Stop & Wait Protocol:

a)



SWS = 1 segment → always in Stop & Wait.

SWS = 6KB.

b) Last Byte Sent = 2048

Last Byte ack = 1024.

SWS = Ad.W − (2048 − 1024)

⇒ 6KB − 1024B

⇒ 6KB − 1KB

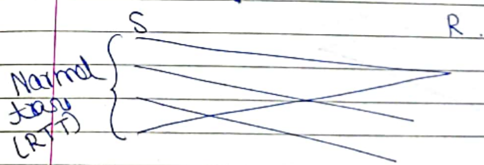
$\boxed{SWS \Rightarrow 5KB}$ → Sender can max transmit 5KB.

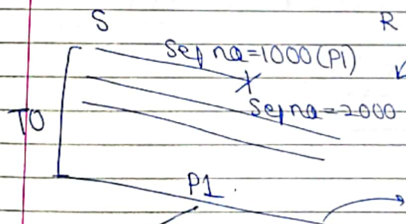
c) Capacity of Sender = 5KB.

But, sender has a req. of sending only 3KB of data

So, entire 3KB of data can be transmitted.

* CALCULATION OF TIME OUT IN TCP





[if retransmitting only P1 → Select repeat]

→ retransmit if not get ack till TO.

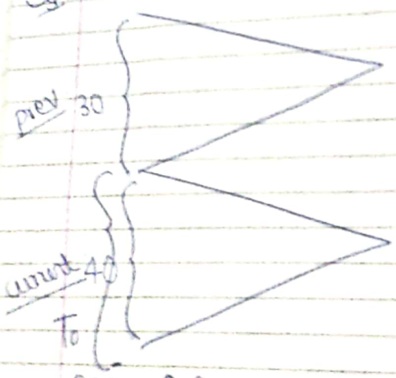
calculation of TO will depend upon previous RTTs

$\boxed{\text{Time out} = 2 * RTT}$

where, $\boxed{RTT = \alpha * \text{Previous } RTT + (1-\alpha) * \text{Current } RTT}$

where, α = Smoothening factor

Q13



$RTT = 0.9*30 + 0.1*40 = \underline{31 \text{ msec}}$

$\boxed{TO = 62 \text{ msec}}$

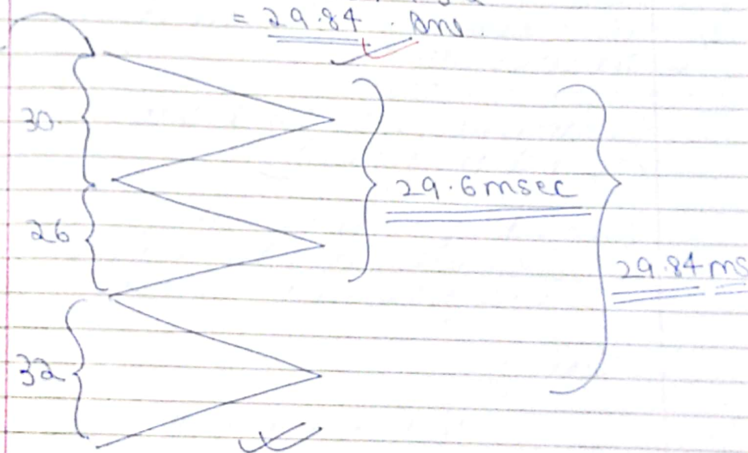
14. prev rtt = 30 → multiple RTT Q.
current rtt = 26.

$RTT = 0.9*30 + 0.1*26$
$= 27 + 2.6$
$\Rightarrow \underline{29.6 \text{ msec}}$ prev

prev RTT = 29.6
current RTT = 32

$RTT = 0.9*29.6 + 0.1*32$
$= 26.64 + 3.2$
$= \underline{29.84} \text{ ms.}$

Q



connection less.

* UDP → connection less.
→ I dont use capacity of rec, no error control, no reliability, no ACK, SYN.

UDP Header:

→ 8 byte HLen.

	00 to 15	16 to 31
W1	Source Port no	Destination Port no
W2	Length	Checksum.
	Data	

1. ⇒ UDP = connection less protocol
2. No reliability, No ACKs.

(Sender will cal CS, & send, but is optional for receiver);
→ used by rec. to check whether data is correct/not. [depends upon rec.]
— But no NAK/ACK involved

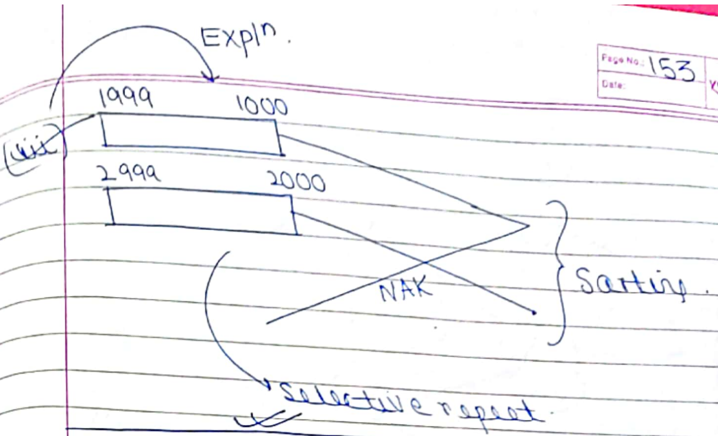
Checksum
→ [To let receiver know abt the status of msg. It is upto receiver to use the rec. data/discard the data.

TCP	UDP
(i) connection oriented Protocol	(i) connection less Protocol
(ii) If a message is sent by sender, the data will be completely sent with proper ack. mechanism. If there is error/loss of msg; the msg will be retransmitted. ∴ [It is reliable]	(ii) UDP is unreliable protocol. If a message has been sent by sender, the receiver cannot track lost packets but receiver can understand the authenticity of the data by using checksum mechanism.
(iii) TCP Protocol is ordered by using Sequence no.s. → using seq no.s, even receiver receives unorder packets, receiver can order the data using seq no.s (SW) → reassembly. Byte orient.	(iii) No order in receiving of the messages.
(iv) TCP is having more overhead.	(iv) UDP is having less overhead.
(v) TCP is slower.	(v) UDP is faster

VI

TCP	UDP
(vi) TCP uses flow control mechanisms like Sliding window, Go back N & Selective repeat.	(vi) No flow control Mechanism are used in UDP.
(vii) HTTP, SMTP, FTP, SSH (secure shell transfer) uses TCP.	(vii) DNS Protocol, Streaming Applications such as IPTV, VOIP (Voice over IP), TFTP (trivial FTP) uses UDP.

(viii)

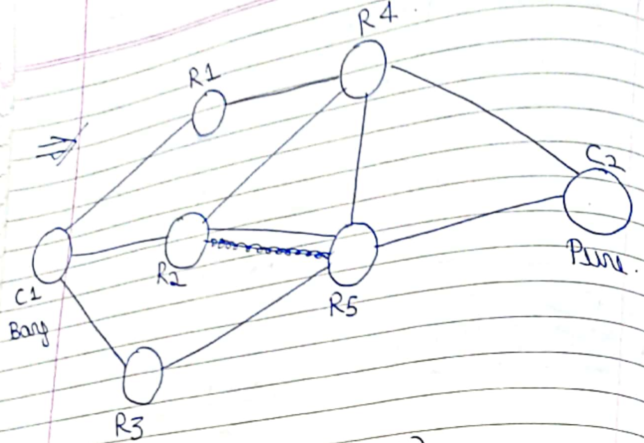


* ROUTING PROTOCOL AND CONGESTION CONTROL

⇒ Routing Protocols
- → Distance Vector routing.
- → Link Space Routing

⇒ Routing protocols
- Static routing algo
- dynamic routing algo.

$R_i \longrightarrow$ Routers
$C_i \longrightarrow$ Computers



Multi Paths b/w C1 & C2.

Path 1: C1 → R1 → R4 → C2.
Path 2: C1 → R2 → R5 → C2.
Path 3: C1 → R3 → R5 → C2
Path 4: C1 → R1 → R4 → R5 → C2
Path 5: C1 → R2 → R5 → R4 → C2.

which path to be selected?
⇓ decided by
Routing algorithms.

Shortest Path

* Static Routing Algorithm → 1980s. (Now none these times).

(i) In SRP, the path to particular source and destination is fixed.
(ii) Each router maintains the data of the neighbours based upon the ip address in the packet.
(iii) It will decide the path using static routing Table.

Each Router ⟶ Routing Table.

R1 ⟶ RT. (info of R4)

R2 ⟶ RT (info of R4, R5).

if 192, R2 ⟶ R5. (static).

⇒ Static Routing Table

R1 ⟶ R4.
R2 ⟶ R4 — 192
R2 ⟶ R5. — 182

* Disadvantage of SRA.

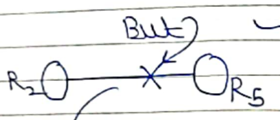
(i) If the link is cut, the router is not programmed to send through any other alternate path.
(ii) If the path to the next router is

computed, it won't be able to forward the packet to any other path because router is programmed for fixed paths.

dest. iP add = 192.72.5.50.

R2 ⟶ R5. → go through R5 (static).

But



∴ don't know how to proceed further

In Dynamic: can take altern. Paths.

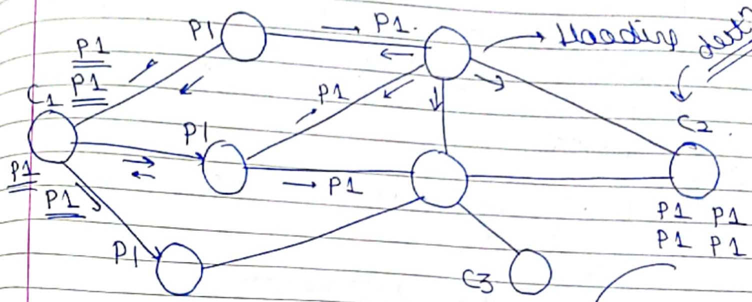
∴ From 1990s – 1995:
→ flooding.

(i) In Flooding Technique, the packet will be forwarded to all the connected neighbours.

(ii) In Flooding technique, lots of duplicate packets will be existing at each device.

(iii) The corresponding device will receive the packet & all other devices will discard the packet.

→ not valid in current situation.



it will rec. many duplicates. but will take one.

C_3 will rec. also packets (but discard) ∵ its not dest ip.

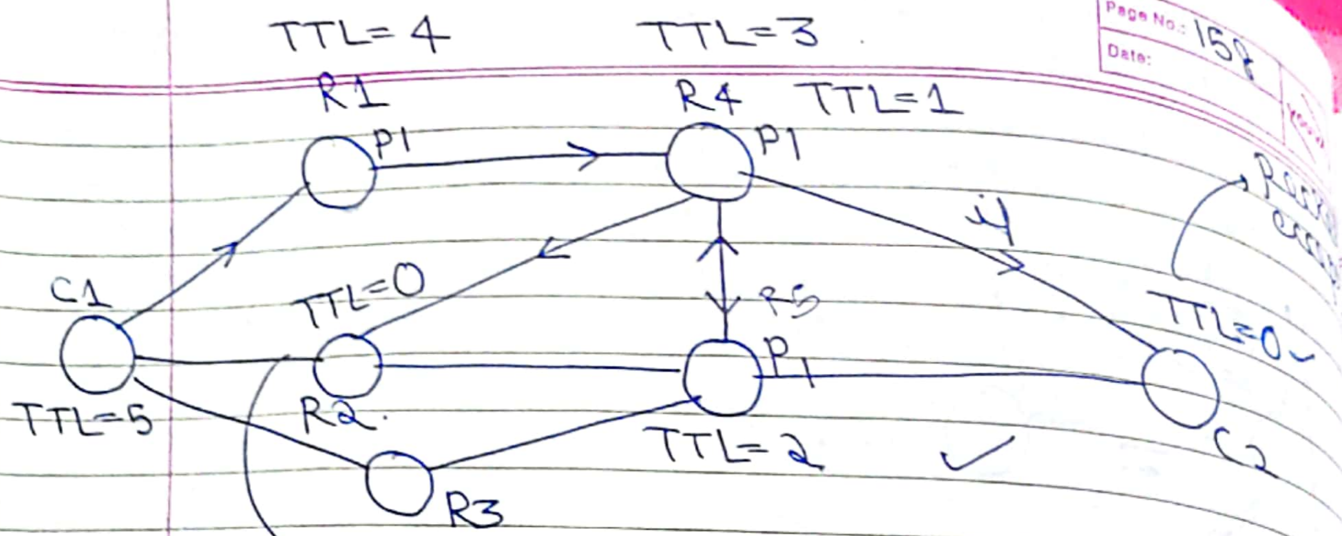
* Time To Live (TTL)

- no. of hops expected to reach destination (Maximum)

[Max no. of hops = C1 – R1 – R4 – R2 – R5 – C2]
→ 5 Hops

∴ $\boxed{TTL = 5}$

→ To avoid looping of packets



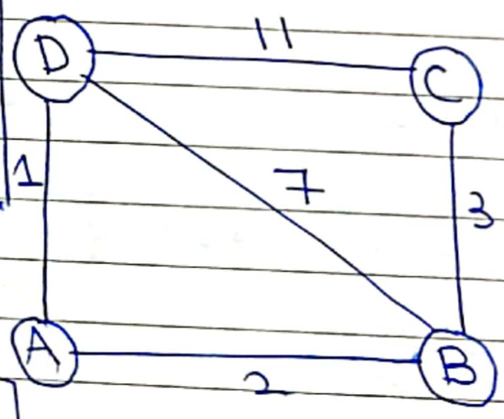
∵ hops over (completed) → 0.
⇒ not reached dest.
[entered loop]. (not in correct dirn)
∴ discard Packet].

* DYNAMIC ROUTING ALGORITHMS:

1. DISTANCE VECTOR ROUTING ALGO.

eg:

A	1
B	7
C	11
D	0



A	∞
B	3
C	0
D	11

DV of A

A	0
B	2
C	~~∞~~ 5
D	1

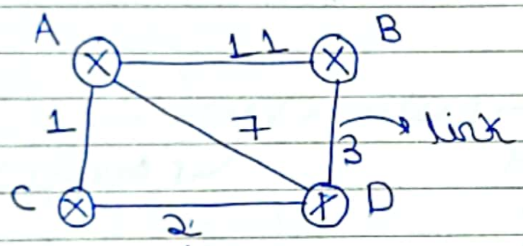
A	2
B	0
C	3
D	~~7~~ 3

DV of B

depending upon situations, the route may change.

DISTANCE VECTOR ROUTING

- Shortest dist. from Router to Router.
- every n/w is connected from routers



no. on edges ⇒ cost

dist/cost/PD/QD.

- minimize dist b/w every pair of V

RT (Routing table).

[every Node only knows abt nbrs.]

Dest	Dist.	Next hop

1: (Path = 1 edge)

RT at B :

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

prep. using local know.

Page No.: 159

3

RT of A

Dest	Dis	NH
A	0	A
B	2	B
C	∞	—
D	1	D

C1

TTL=5

not nbr.

RT of D

Dest	Dis	NH
A	1	A
B	7	B
C	11	C
D	0	D

RT of C

Distance Vector → [Array]

Dest	Dist	NH
A	∞	—
B	3	B
C	0	C
D	11	D

Itr^n 2: (Path may cont. at most 2 edges)

- every router will exchange their DV with nbrs.

eg: A ← D, B
C ← B, D

DV of A

* At A:

DV received from B, D

At B: DV rec. from A, C, D

At C: DV rec. from B, D

happening parallel

At D: DV rec. from A, B, C.

* At A: DV from B, D → nbrs.

update RT from DV(nbrs).

B	D
2	1
0	7
3	11
7 ✓	0 ✓

A	0	A
B	2	B
C	5	B
D	1	D

new RT from Knowl. from nbrs.

$A \rightsquigarrow B = \min \begin{cases} A \xrightarrow{1} D + D \rightsquigarrow B \ (7) \\ A \xrightarrow{2} B + B \rightsquigarrow B \end{cases}$

$A \rightsquigarrow C = \min \begin{cases} A \xrightarrow{1} D + D \rightsquigarrow C \ (11) \\ A \xrightarrow{2} B + B \rightsquigarrow C \ (3) \end{cases} \Rightarrow \underline{\underline{5}}$ ✓

$A \rightsquigarrow D = \min \begin{cases} A \xrightarrow{2} B + B \rightsquigarrow D \ (7) \\ A \xrightarrow{1} D + D \rightsquigarrow D \ (0) \end{cases} \Rightarrow \underline{\underline{1}}$ ✓

By Graph

Shortcut : → in Gate :

rom B, D.

n B . from D
1
7
11
0
AD=1 (Graph)

Updated RT at

A	0	A
B	2	B ← min at
C	5	B
D	1	D ← min at

from A, C, D.

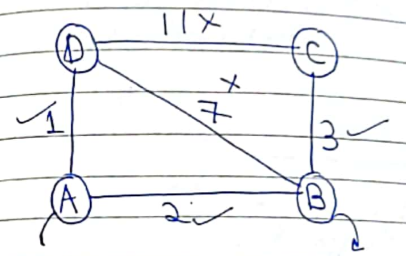
n A from C from D
∞ 1
3 7
0 11
11 0
CB=3 BD=7

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

SP decr.

SP → max 3 edges. ← 4 Nodes
max n−1 iterations

*



Final RT

A		
A	0	A
B	2	B
C	5	B
D	1	D

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

C		
A	5	B
B	3	B
C	0	C
D	6	B

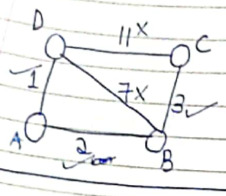
D		
A	1	A
B	3	A
C	6	A
D	0	D

Final RT after 3 ite

* Some Possible Qs :

which not used ?
→ DC, DB.

⇒ Directly are without RTS :



→ if alter^n better Path, nvr be used.

1. Given DV, what RT?
2. Given Graph, what is final RT?

C | ∞ | 5
D | 1

upon
...ors, the
may

D | 7 | 13

Dis of B

How this works?

Every Node constructs a 1D Array (Distance Vector) consisting of cost to all other nodes

DV. at A :

A	0
B	2
C	∞
D	1

Similarly, all other nodes

(ii) Each Node will distribute its distance vector to the connected neighbours.

eg: $\left\{ \begin{array}{l} A \xrightarrow{\text{send}} B, D \\ A \xleftarrow[\text{rec. DV.}]{} B, D \end{array} \right\}$

B ⟶ A, C, D.

(iii) If any of the Node which has received the distance vectors from its neighbours finds that its neighbour is advertising a path less than what it currently knows, then it will update its Distance Vector.

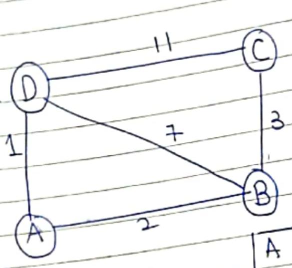
→ There, is no if :

I am at A   A ← B, D

A	5
B	3
C	0
D	~~0~~

10. 6.

A	1
B	3
C	10
D	0



A	0
B	2
C	~~∞~~ 5
D	1

min

A	2
B	0
C	3
D	~~7~~ 3

update

A → C = ∞
A → B → C = 5.
A → D → C = 12

B → D = 7.
B → A → D = 3
B → C → D = 14

[any node via B, D].

⇒ A to B : min { A → D → ~~B~~ D → B = 1 + 7 = 8. ; A → B + B → B = 2 } → 2

A to C : min { A → B + B → C = 5 ; A → D + D → C = 12 } → 5

A to D : min { A → B + B → D = 9 ; A → D + D → D = 1 } → 1

(All will happen parallely, take old V.)
[not updated value]

Shortcut:

at A

	B	D	
A	2	1	— 0
B	0	7	— min(2+0, 1+7) = 2
C	3	11	— min(5, 1+11) = 5
D	7	0	— min(2+7, 1+0) = 1

AB = 2   AD = 1

Update DV (A): 0, 2, 5, 1

B

	A	C	D	
A	0	~~3~~ ∞	1	- min{2, ∞, 8} = 2
B	2	3	7	— 0
C	∞	0	11	min{∞, 3, 18} = 3
D	1	11	0	min{3, 14, 7} = 3
	BA = 2	BC = 3	BD = 7	

(B): 2, 0, 3, 3

C

	B	D	
A	2	1	min{5, 12} = 5
B	0	7	min{3, 18} = 3
C	3	11	min{6, 22} = ~~0~~ 0
D	7	0	min{10, 11} = 10
	BC = 3	CD = 11	

D

	A	C	
A	0	∞	min{1, ∞} = 1
B	2	3	min(3, 14) = 3
C	∞	0	min(∞, 11) = 11
D	1	11	~~min(2, 22) = 2~~ 0.
	DA=1	DC=11	

* Max no. of iterations to be done =

[nodes − 1] = 4 − 1 = 3 → to get optimal value.

* 2nd itern: on updated values
↳ neighb's neighb's vals.

A

	B	D	
A	2	1	0.
B	0	3	min(2, 4) = 2
C	3	10	min(5, 11) = 5
D	3	0	min(5, 1) = 1
	AB=2	AD=1	

D

A	B	C	
0	2	∞	1
2	0	3	3
∞	3	0	10
1	7	11	0
1	7	11	

B

	A	C	D	
A	0	5	1	min(2, 10, 4) = 2
B	2	3	3	0
C	5	0	10	min(7, 5, 13) = ~~5~~ 3.
D.	1	10	0.	min(3, 15, 3) = 3
	BA=2	BC=~~5~~ 3	BD=3	

C

	~~A~~	B	D
A		2	1
B		0	3
C		3	10
D.			

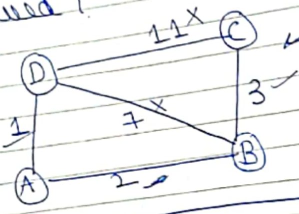
D

	~~A~~	B	D	
A		2	1	min(5, 11) = 5
B		0	3	min(3, 13) = 3
C		3	10	~~min(6, 20) = 6~~ 0
D		3	0	min(6, 10) = 6.
		CB=3	CD=10	

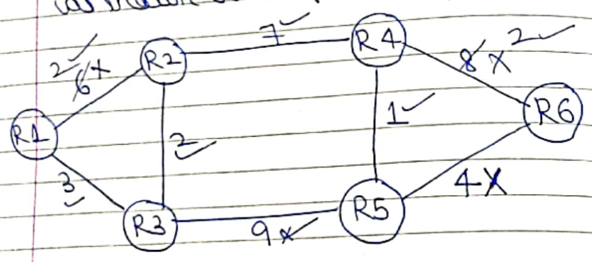
* [If no. of nodes − 1] → DV stabilise

optimal → just by looking.

Q Which of the edges are never been used? — imp Qs.



Q Consider a network with 6 routers R1 to R6 with links having weights as shown in the following diag:



After all RTs stabilize, how many links in the network will never be used for carrying any data?

(a) 4
(b) 3
(c) 2
(d) 1

from $R_1 - R_2$: $R_1 - R_3 - R_2$

from $R_4 - R_6$: $R_4 - R_5 - R_6$

∴ R_1 to R_2 & $R_4 R_6$ not used.

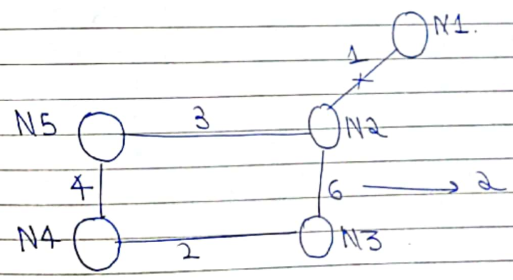
Suppose the weights of all unused links in the previous Q is changed to 2 & DV is used again, how many links will now remain unused

(a) 0
(b) 1
(c) 2
(d) 3

$R_5 - R_4 - R_6 = 3$.

$R_5 - R_6$

Q Consider a Network with 5 Nodes, N1 to N5 as shown below.



Once the routers have been stabilized, the distance vector is shown below:

N1(0, 1, 7, 8, 4)
N2(1, 0, 6, 7, 3)
N3(7, 6, 0, 2, 6)
N4(8, 7, 2, 0, 4)
N5(4, 3, 6, 4, 0)

(i) The cost of the link N_2 to N_3 reduces to 2 (in both the dirns).
(ii) After next round of updates, what will be the new distance V at N3?

(a) (3, 2, 0, 2, 5)
(b) (3, 2, 0, 2, 6)
(c) (7, 2, 0, 2, 5)
(d) (7, 2, 0, 2, 6)

N3

N3 → neighb.

	N2	N4	
N1	1	8	{3,10}=3
N2	0	7	{2,9}=2
N3	6	2	0
N4	7	0	{9,2}=2
N5	3	4	{5,6}=5

$N_3N_2=2$ $N_3N_4=2$

Q. After updating the prev. Qs, link N_1-N_2 goes down, N_2 will reflect this change immediately in its distance vector at a cost infinity (∞).
After the next round of update, what will be the cost to N1 in the DV of N3?

(a) 3
(b) 9
(c) 10
(d) ∞

N3

	N2	N4
N1	∞	8

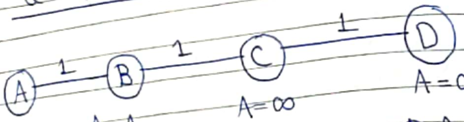
to reach N4 = 2
∴ 2+8 = 10

N3 − N4 = 2 ∴ 2+8 = 10

DV of N3.

from N_2	from N4	
∞	8	10
0	7	
6	2	
7	0	
3	4	
2	2	

* Distance Vector suffers from count-2-infinity problem:



	B	C	D
init 1:	A=1	A=∞	A=∞
itr^n 2:	A=1	A=2	A=∞
itr^n 3:	A=1	A=2	A=3

→ After 3 itern: optimal Value

Suppose, A —×— B (link ↓).

A=∞ ——— A=4 ——— A=3

A=3

⋮

∞

[Count to infinity problem]

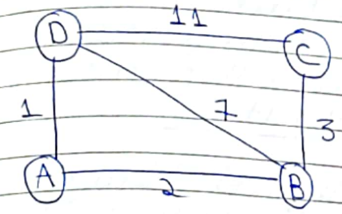
* To avoid this: Link State Vector.

→ after 95.

* LINK STATE ROUTING ALGORITHM.

D

A	1
B	7
C	11



C

B	3
D	11

A

B	2
D	1

link state packet → distance to its NBs.

B

A	2
C	3
D	7

Starting Pack = hello Pack

⇒ In link State routing Algorithm, each node Sends HELLO Packet to all the nodes in the n/W.

But DVR sends packets only to its Neighbours. → multi itr^ns.

→ send packet to all syst. in n/W

{ A ⟶ B, C, D }
{ A ⟵ B, C, D }

⇒ By looking at the link State Packets, any Node (in this eg: A), will construct the optimal value by using Shortest Path Algorithm (Dijkstra Algorithm).

⇒ The corresponding Node, will construct routing table after applying Shortest path algorithm.

⇒ Only 1 iteration is required in case of link State routing algorithm.

⇒ Whenever there is a change in the n/W, again the link state packet will be constructed & sends this packet to all the nodes connected in the n/W.

⇒ If there are frequent changes in the n/W, the link State packets will be changing very frequently.

⇒ To find out the latest packet, a seq. no is attached to each link State Packet.

⇒ Whenever a node receives multiple link State packets at the same time, due to change in the n/W frequently, the receiving node will accept the link State packet with Highest

Sequence no.

⇒ All the Packets with sequence no. except highest will be discarded.

A ⟵ BCD

RT

} optimal path to all other nodes

own ~~LSP~~ RT (by looking at others LSP)

If change: AB ⟶ ~~2~~ ~~3~~ ~~4~~ 5

Cno 1, 2, 3

(with each change, LSP created)

1st change ⟶ Sno 1
:
nth change ⟶ Sno n.

∴ Accept packet with highest seq no. (LSP) & discard rest.

Q. Among the popular routing algorithms distance vector & link state, which of the following are true?

S-1] Count to infinity problem is only with DV but not LSR. ✓

S-2] In LSR, Shortest Path Algo is run only at one node. ×

S-3] In DVR, the shortest path algo is run only at one node. ×

S-4] DVR requires less number of network messages than link state. ✓

(a) S1, S2, S4
(b) S1, S3, S4
(c) S2, S3
(d) S1, S4

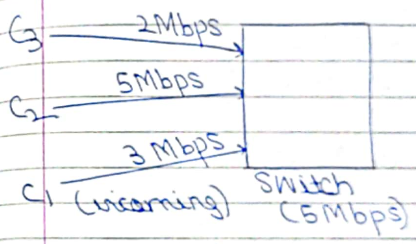
S-2] at all nodes (S-3, S-4).

* CONGESTION CONTROL

→ function of NL

* Congestion

Whenever more data is coming into network, routers/Switches having fixed length buffer, some of the issues will occur. This is called as congestion.



1S —— 5Mb [outgoing]

Switch can withold = 5Mb/S
Incoming data = 10Mb/S
→ ∴ congestion.

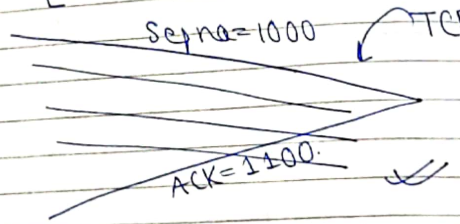
> fixed capacity.
→ router/switch ↓

* There are multiple ways to avoid this Issue:

⇒ Precautions for congestion control:

(1) By using Cumulative acknowledgement instead of Normal acknowledgement.

[CA → in TCP]



⇒ for all segments, 1 ack.

(2) By using Selective repeat Protocol, instead of Go Back N Protocol.
[more traffic]
less noisy channel.

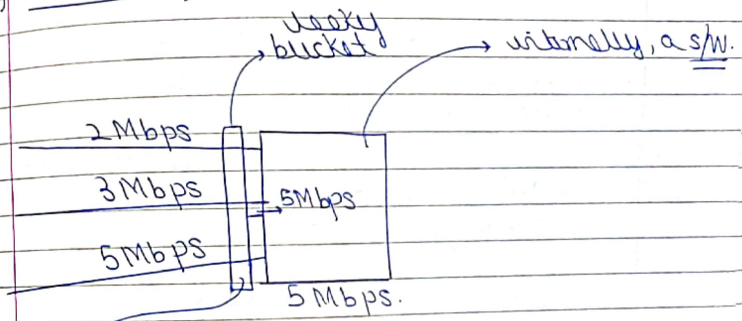
(3)

* Algorithms used for Congestion Control

(1) Leaky Bucket Algorithm
(2) Token Bucket Algorithm
(3) Congestion Control Algorithm
→ Slow Start Phase.
→ Congestion avoidance Phase
→ Congestion control Phase.

AIMD algo. [Add incr Mult. dec.]

(1) LEAKY BUCKET ALGORITHM:



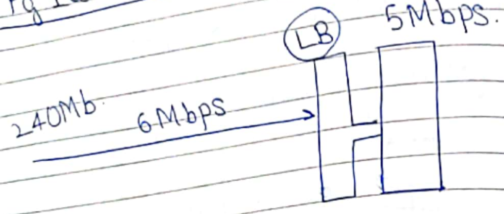
Incoming data = 10Mbps.
outgoing capacity = 5Mbps

→ take all data & provide to router with constant rate.

6 Mb — 1s.
240Mb — x.

Pg 12 → Q 1 (Congestion Control)

1.



Time taken to send in

⇒ No. of 6Mbps bursts towards router

⇒ $\frac{240Mb}{6Mbs} = \underline{\underline{40 s}}$

In 1 s — 6Mb coming —— 5Mb going.

∴ 1Mb stored in LB

$\boxed{\text{Min Size} = 40Mb}$

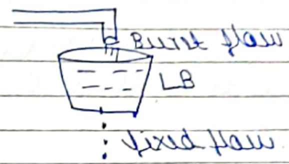
2. $\frac{100Mb}{5Mb\,ps} \Rightarrow \underline{\underline{20s}}$ ← time taken to send data

5Mb − 2 = 3Mb → store.

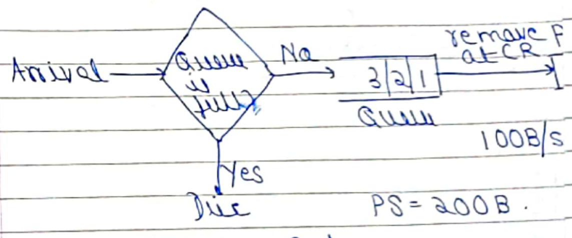
∴ $\boxed{60\ Mb}$ → for 20 s.

---

* LEAKY BUCKET → Congestion c



⇒ Bucket implem. in form of queue



PS = 200B.

Q. What is BS? → Gate:

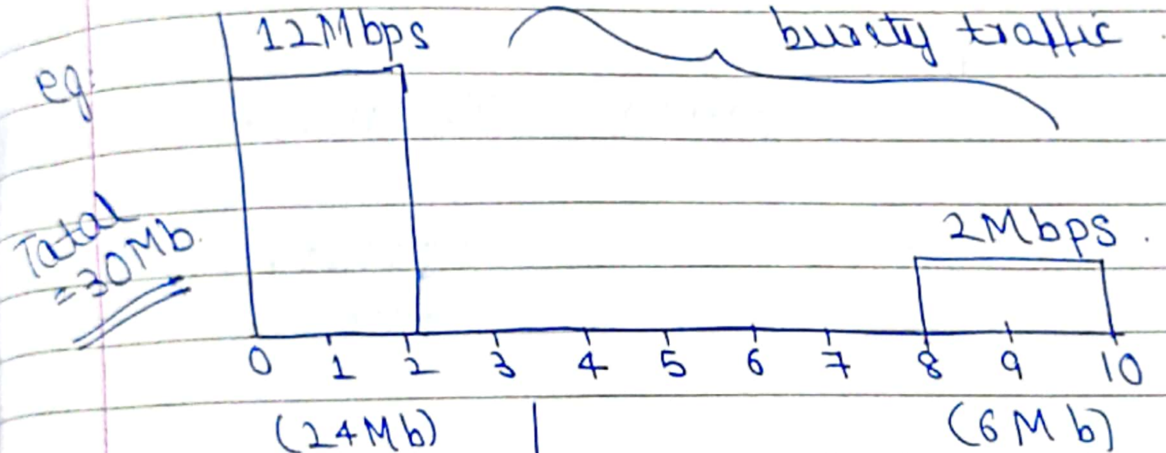
⇒

for first 10 s — idle

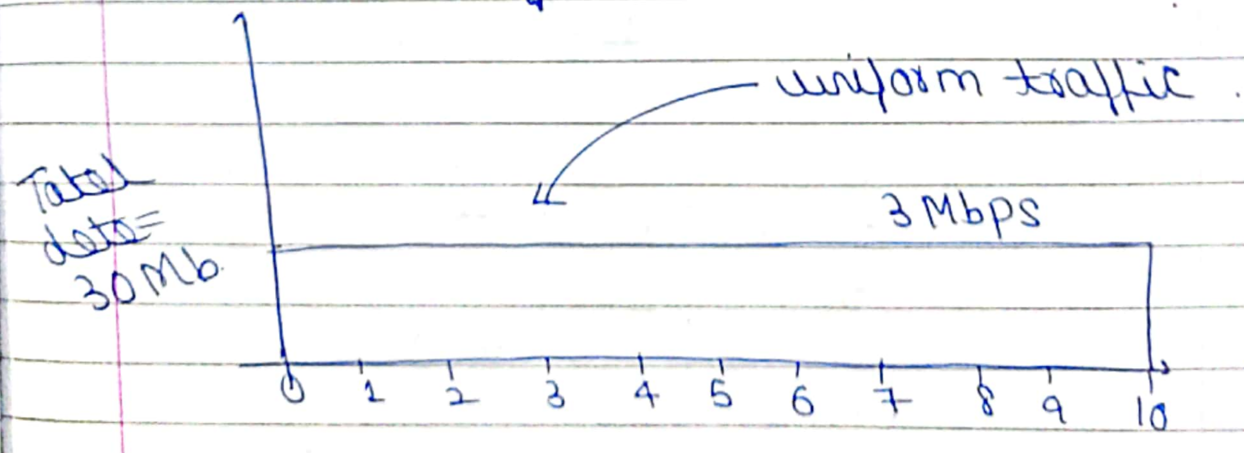
Leaky Bucket Algo:

1) When host has to send a packet, the packet is thrown in the bucket.
2) Leaky bucket leaks at constant rate.
3) Bursty traffic is converted to uniform traffic by using leaky bucket. (sometimes full / no traffic)

eg:



using LB (3Mbps).



3) 1st packet — 1st ms → sent
2nd packet — 4th ms
3rd packet — 7th ms.

P4-1000
P4-1000
~~P3-400~~
~~P2-400~~

Capacity = 3000 Bytes.

~~SNo. Time Packets in Bucket~~

Sno.	time (msec)	Pcks in Bucket	sent	lost	Buc. Space remaining
1	1	–	1	–	3000.
2	2	2	1	–	2600
3	3	2,3	1	–	2200
4	4	3,4	1,2	–	1600.
5.	5.	3,4,5	1,2	–	600.
6.	6.	3,4,5,6	1,2	400	0
7.	7	4,5,6,7	1,2,3	600	0
8	8	4,5,6,7,~~8~~	1,2,3	1100	0.
9	9	4,5,6,7,~~8~~	1,2,3	2200	0

X

Correct Table

Sno.	(msec) time	Packet in Bucket	Sent	Lost	Bucket Space rem (B)
1.	1	–	1	–	3000
2.	2	2	1	–	2600
3	3	2,3	1	–	2200.
4	4	3, 4	1,2	–	1600.
5	5	3,4,5	1,2	–	600
6.	6.	3,4,5,P(6) 600.	1,2	P(6) 400	0
7	7.	4,5,P(6) 600, P(7) 400	1,2,3	P(6) 400, P(7) 600	0
8.	8.	4,5,P(6) 600, P(7) 400,	1,2,3.	P(6) 400, P(7) 600, P(8)	0
9.	9	4,5, P(6) 600 P(7) 400	1,2,3	P(6) 400, P(7) 600. P(8), P(9)	0.
10	10.	5, P(6) 600, P(7) 400	1,2,3,4	P(6) 400, P(7) 600, P(8), P(9)	1000+– depends on data

Sno	time	Pcks in Bucket	sent	lost	Buck Sp rem.
1	1	–	1	–	3000
2	2	2	1	–	2600
3	3	2,3	1	–	2200
4	4	3,4	1,2	–	1600
5	5	3,4,5	1,2	–	600
6	6	3,4,5, P(6) 600	1,2	P(6)= 400	0
7	7	3,4,5, P(6) 600 P(7) 400	1,2,3	P(6) 400 P(7) 600	0.
8	8	4,5, P(6) 600 P(7) 400	1,2,3.	P(6), P(7) 400 600	0.
9	9				
	"	X	"	"	P(6) P(7) 8, 9	0.

4) Output = 5 Mb/min
input = 100 Mb/min ⟶ 12 sec
rem: 48 sec ✗

⟶ Bursty traffic.

Soln.

60 s —— 100 Mb of data
12 s —— x .
} i/p rate

⇒ 20 Mb of data → i

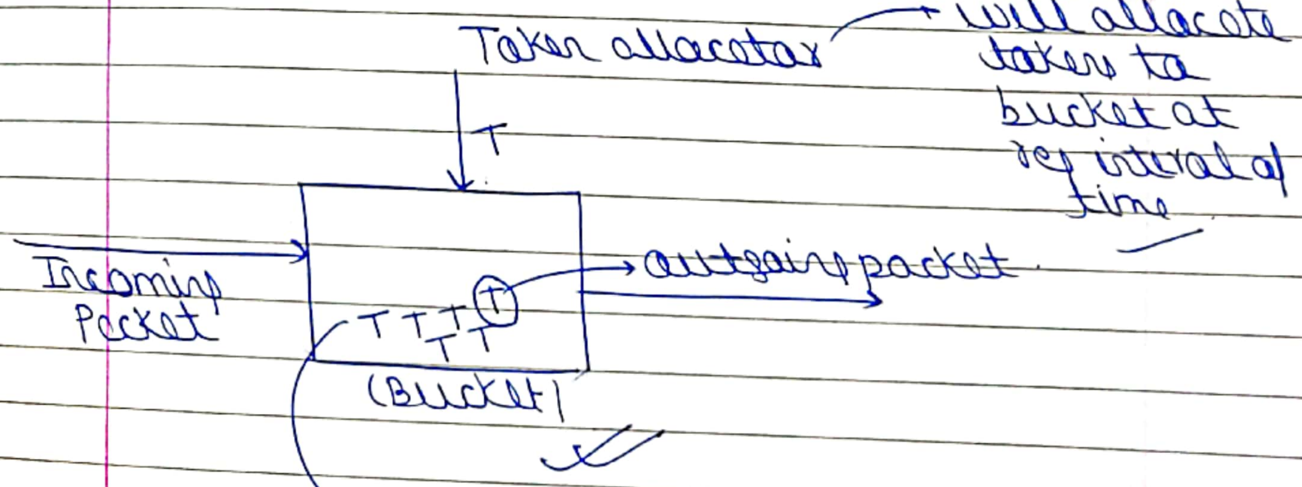
5 60 s —— 5 Mb
12 s —— x .
⇒ 1 Mb
} o/p rate

∴ $\boxed{\text{Min Bucket Size required} = 19\text{ Mb}}$

⟶ To avoid this scenario ⟶ Token Bucket

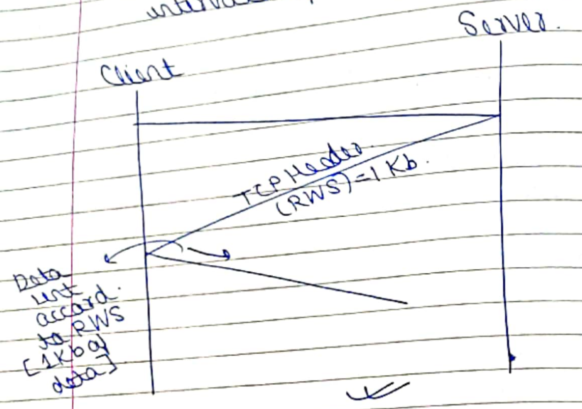
* TOKEN BUCKET ALGORITHM:

(i) In Token Bucket algorithm, token will be generated at regular intervals of time.
(ii) Token bucket has a fixed capacity, which has been calculated according to network size.
(iii) Token is authorisation to send data.
(iv) If more no. of tokens are available in the bucket, more data can be sent.
(v) If there is a ready packet, a token is removed from the bucket & packet is sent to the network.
(vi) If there is no token in the bucket, the packet cannot be sent.



{Compare it with Ranita Paratha mess Coupon}

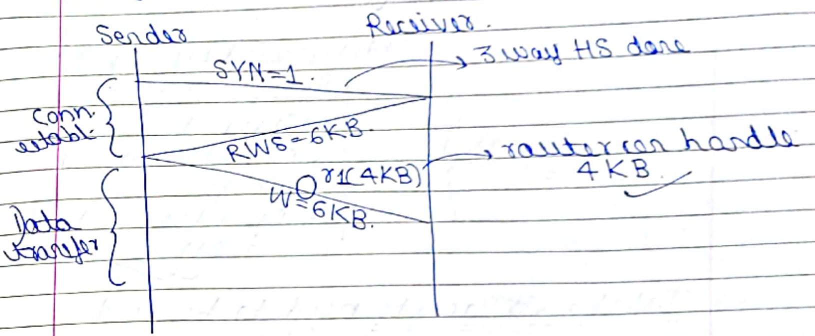
* Congestion Window:
  has to be calculated at regular intervals of time.



15/6/19
Saturday

Lecture 5

* Congestion Window



till now, SWS = min{recvr window, congestion window}

$\Rightarrow$ min{6KB, 4KB}

$\Rightarrow$ 4KB

* How to cal. Congestion Window?

Q. Let us say, Sender has to send total 27 Packets. In these 27 Packets, 10th Packet & 24th Packet was lost due to congestion. Provide the sequence in which the congest. window will be calculated & Packets will be sent.

Scanned with CamScanner

* Each packet wasn't able to exactly cal the CW.
(i.e. packets one by one, till some of the packets is lost, after that dec. ↓
↳ determined by congest. control algo.)

Phases
- → Slow start phase
- → congestion avoid phase
- → congestion control Phase.

⇒ Total: ~~27~~ Packets need to be send.
(P.size = 1 KB)

Start with SSP.
start with: ~~Max Segment Size (MSS)~~ → 1 KB (given)
↓ (Start with MSS)

RTT
1. Start with 1 Packet) ↑exp.
2. Send 2,3 Pack. (if ACK rec)
3. Send 4,5,6,7 ...
4. Send 8,9,10,11,12,13,14,15 → getting cong.

[Time Out] (✗ (cong WS = 8).
Threshold $= \frac{1}{2}$ CW = 4 Pac.
→ entering into CAP.

[(go to MSS)]

SS till Thresh
5. 10 — 1
6. 11, 12. — 2
7. 13, 14, 15, 16. — 4

8. ~~17, 18, 19, 20, 21, 22, 23~~

8. 17, 18, 19, 20, 21, → addit. increase algo. (add by 1 MSS)
9. 22, 23, 24✗, 25, 26, 27
→ (CW = 6, TW = 3).

10. 24
11. ~~26~~ 25, 26.
12. 27

∴ [12 RTTs used. each RTT → 10 ms. Total time = 120 ms]

* Instead of Slow Start, same comp uses AIMD algorithm
(Addit. increase, multiplicat. decr.)

1. 1 Packet → after getting ack, ③.
2. 2, 3) (not exp.) +1
3. 4, 5, 6.
4. 7, 8, 9, 10✗ (CWS = 4, TW = 2)

[go to TW] (2)

5. 10, 11
6. 12, 13, 14
7. 15, 16, 17, 18
8. 19, 20, 21, 22, 23 (5)
9. 24✗, 25, 26, 27 (CWS = 6, TW = 3)

[go TW]

(More traffic)

10. 24, 25, 26

11. 27

[11 RTTS]

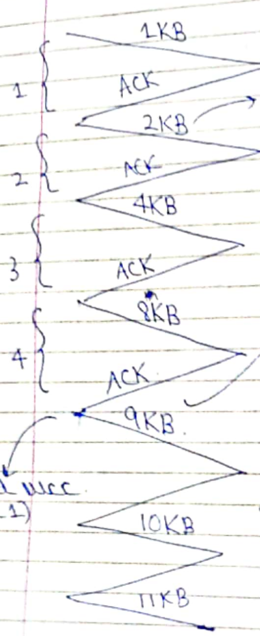
[default → SSP]

Pg 13

Q.6 TCP CW = 18 K Bytes → loss.

MSS = 1 KB.

$\boxed{\text{Threshold} = 9K}$



→ next I can send 2 KB, S2, S3

→ if 16 KB, error

$\therefore \boxed{\text{Window} = 9KB}$

not succ Q(1)

→ if 6 Burst succ. Q(2) → 11 KB

7. MSS = 2 KB.
RWS = 24 KB.
RTT = 10 msec.

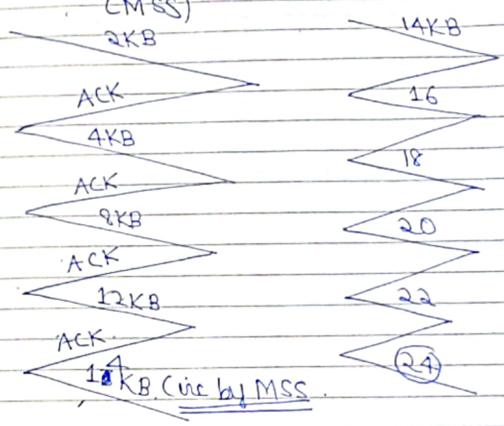
→ ∵ no TimeOut.

* SWS = min(RWS, CWS)

RW = 24 KB.

$\therefore \boxed{\text{No loss}}$ (CWS)

M-1 TW = 12 KB (MSS)



⇒ 9 RTTS → 90 msec.

M2

$\frac{24KB}{2KB} = 12$ segments

Threshold = 6 segments

1, 2, 4, 6 / 7 / 8 / 9 / 10, 11, 12

9 RTTS

* SWS depends on RWS & CWS both (CC Algo used)

Pg 13 → Q5

* Application Layer Protocols And Security

Gate → (DNS, SMTP, POP, FTP, HTTP) and Security

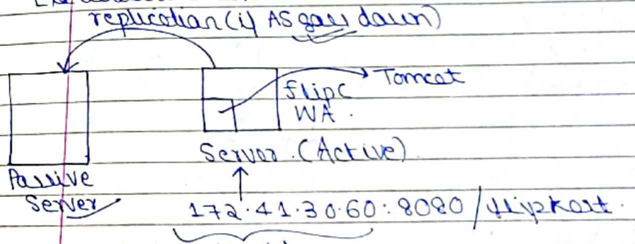
1. Basics of Public & Private Key Cryptograp.
2. Digital Signatures & Security.
3. Fire Walls.

* DNS Protocol.

(Domain Name Service).

→ Humans can rem names not No-s.

[to avoid disaster].



domain address ⟷ 172.41.30.60	

1. url
2. then look in DNS Server
3. fetch IP

→ if DNS ≈ table

if (sig. table → lot of time)
[400 crore of data]

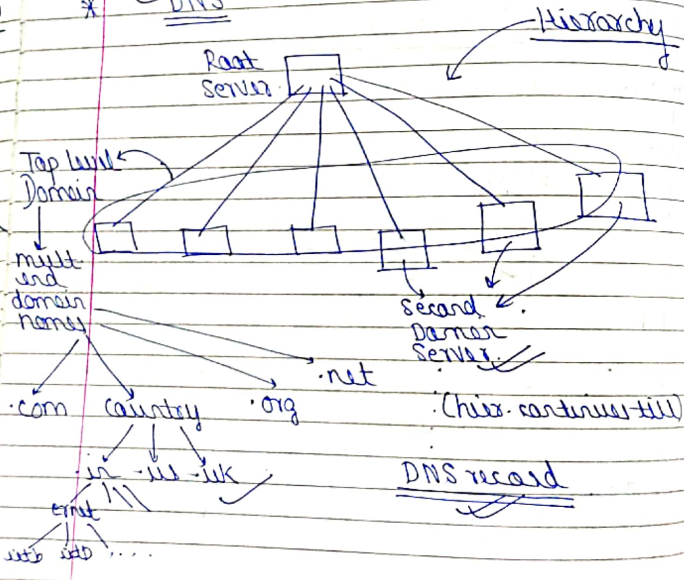
using 256 Gb ram → create 1 lakh threads to run parallely on Server

mapping

domain Addr/name ⟷ ip → DNS functn.

divide into mult. loc'n.

* DNS



Domain name ⟷ ip

⇒ Our aim is to get DNS record.

→ DNS record consist of :

NI

ip Address. – Domain name – Validity

→ some ip add are valid for some period of time.

Steps: Steps to get ip add.

1. Domain name (URL)
   eg: cse. iitm. ernet. in → domain name in url
2. will look into cache. →
3. if not avail. in cache:
   ↓
   contact root Server.
4. Root Server returns the top Level Domain Server's ip Address.
5. request will go to Top Level Domain Server.
   I don't have this ip, but will return .in
6. TDS will return .in Server's ip Address
   ⋮
   till USR gets DNS record.
7. update Cache.

* DNS uses UDP Protocol for further transmission without having any overhead.

(∵ User invoking will press enter if not received.

⇒ Application layer Protocols are set of rules in which these protocols indicate the conditions for sending and receiving between client and Server.

* FTP

- File Transfer Protocol

· FTP uses TCP. (Trivial) · TFTP uses UDP.

(i) FTP protocol is used to transfer the data between 2 computers on the same local Network.

(ii) FTP has secure algorithms to transfer the file between Sender and receiver.

(iii) In order to transfer a file from client to Server, user must authenticate on FTP Server.

(iv) By authentication, a session will be created between client & Server for transferring the files.

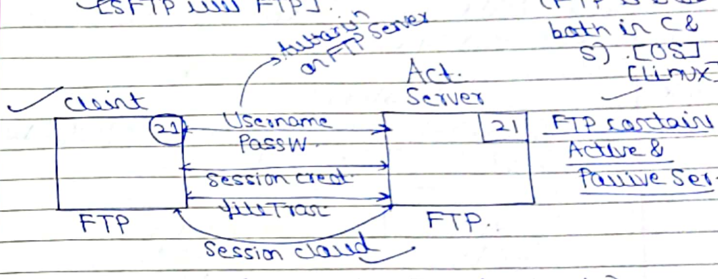
FTP has many. also.

∴ [In Windows, you need to install FTP]

(v) The authentication Process b/w client & Server, allows to set proper file permission which they don't have access and allowing client read, write & set executions / permissions on file.

(vi) Once, the transfer is complete, the session will be disconnected b/w client & Server.

[SFTP uses FTP]. (FTP is cross. both in C & S) [OS] [LINUX]



FTP contains Active & Passive Ser.

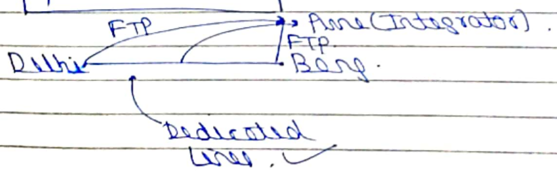
Q. Why telnet? (Diff b/w FTP & telnet)

can copy data from a remote server — can actually access & modify prog on remote server.

US In

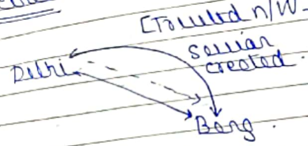
1. multiple acc.
2. fast.
3. more secure
4.

FTP > PD



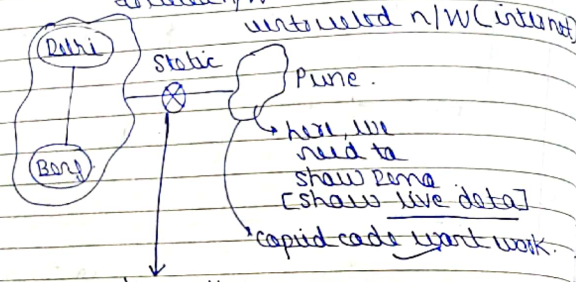
[FTP → copy.
Telnet → access, modify & see the remote system.]

Telnet:



- copying is not a soln. (FTP)
- from delhi, we can access remote server & can modify.

* trusted n/W.
untrusted n/W (internet)



Firewalls have an eye on each packet going from TN to UTN.

* SMTP

(i) Simple Mail Transfer Protocol is a TCP/IP Protocol for sending & receiving mail.
(ii) SMTP has a set of rules that allows software to send e-mail over internet.
(iii) SMTP has 2 parts:
{ 1. Messager → which will prepare messages
2. Transmitter. ↳ which will transmit the messages. }
(iv) The SMTP server act like post offices that will handle routing of the messages.
(v) SMTP Provides a set of codes that simplify communication of e-mail messages between servers.
(vi) Any email message has same parameters like sender, recipient, message body etc.
(vii) The text is separated by code words or members that identify the purpose of each section.

For eg: code — purpose of each section — parameter

MAIL FROM : Sender.
RCPT TO : recipient.
DATA : message.
QUIT : quit.
VRFY : verify an address.

* POP Protocol (IMAP)

(i) POP Protocol is useful for checking e-mails from a computer that is in specific locations.
(ii) For multiple locations, use IMAP protocol.
(iii) POP is used to download the mails when user comes online. (into cloud space)

Q Diff b/w DNS & SMTP.

a@ibm.com

ibm — Microsoft — gmail

a@ibm.com
b@ibm.com

Kept in mail server. [serv which cont all mails]

(mail from n no of users)

Sender: __
Recipient: __
— GUI

→ entire thing will go as txt [coll'n of st].
(SMTP attaches CW).

I
GUI
Sender: a@ibm.com
Recip : b@ibm.com

message (1)

① make it into a msg (container)
② Now, transmitter (set of msgs)

look into
(DNS Mail Server)
(look Rec. add.)
(get IP address)

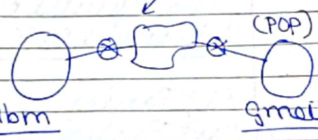
routed

server. [map funct^n. (POP) [map user to msg]]

II
Sender : a@ibm.in
Rec : b@gmail.co

message
~~text box~~
[gmail in other loc]

(look into DNS).



*POP maps user to msg & alloc. into cloud space.

go into corr. storage

eg: Gmail gives 15Gb storage.

198

* HTTP PROTOCOL

(i) Hyper text transfer protocol.
(ii) HTTP function as request, response protocol in client server computing model.
(iii) A web browser : eg: maybe a client and application running on the computer hosting a website may be server.
(iv) The client submits http request to the server.
(v) The server provides resources such as html files and other content and performs the functions on behalf of the client and returns a response message to the client.
(vi) The response contains completion status about the request & may also contain the requested content in its message body.
(vii) Http works in 3 steps :

(i) Passing the URL → ⊙
(ii) Sending the request.
(iii) Server Response.

Protocol : ip address : Port no / URL

(ii) Sending the req.
- GET — get the URL data.
- POST — Post on the server

199

ICRTC.
Post : src, dest.

→ i/p given as post request. (src/dest to IRCTC)

(iii) Server takes the i/p, process i/p, prepare the o/p & send it to client.

* HTTPS

- its a $\boxed{\text{secure version}}$ of http that allows secure e-commerce transactions such as online banking.

(every website, puts cookies in browser, knows our browsing history) — adds

eg: TOR. bypass all things & won't allow cookies to enter into website.

Q. Consider different activities related to email.

M1: Send an email from mail client to mail server.

M2: Download an email from mail box to mail client.

M3: Checking e-mail in a web browser.

Which is the application level protocol used in each activity.

Soln.

M1: SMTP ✓

M2: POP ✓

M3: HTTP ✓

Q. Which of the following transport layer protocol is used to support electronic mail.

(a) SMTP → Application layer Protocol.

(b) IP

(c) TCP

(d) UDP

Q. The protocol data unit for the application layer in internet stack is:

(a) Segment → TL.

(b) datagram → NL.

(c) Message.

(d) Frame. → DLL

AL: Message

↓

TL — Segment

IP — Datagram

(fragments) [MTU]

DLL - Frame → bits → Analog (PL)

---

* SECURITY

Symmetric Algorithms → The Key used by client & Server are same.

Asymmetric Algorithms. → The Key used by client & server are different.

---

* Message (plain text) ∵ (not secure).

Key → msg (encrypt)

↓

Cipher text (Sender)

medium → ← won't be understood by attacker.

↓

Cipher text ← Receiver

→ CT ← Key (decrypt)

CT → Message (Plain Text)

Not used now

* Symmetric Algo

1980-90s

Message → Shoot

Key
S - 1
h - 2
o - 3
t - 4

Cryptography

12334 → cipher Txt [Sender]

12334 ← Key (same as sender)

Shoot

* Asymmetric Algorithms

Public Key, Private Key

- Public Key: Known to everyone
- Private Key: Known only to authorised persons

P_b = Public Key
P_r = Private Key

encrypt.

$(P_b)_A, (P_b)_B, (P_r)_A$
Client A

$(P_b)_B, (P_b)_A, (P_r)_B$
Server B.

message

(encry) $(P_b)_B$ → message ← A

$((P_b)_B$ msg$)$ (Cypher Txt)

Cipher Txt → Cipher Txt

public Key of B

$((P_r)_B A)$ ← $(P_r)_B$ [decry.]

private Key of B

msg → Server B

* DIGITAL SIGNATURE

In DS, signing with priv. Key, but all people who have Public Key can read it. [of A].

Ⓐ Msg

Ⓑ

$(P_r)_A$ → msg

msg with DS: $((P_r)_A$ msg$)$ ← B

Signed Msg → signed msg

private key of A

$((P_b)_A (B))$ ← public Key of A

msg

Q. Using Public Key cryptography, X adds a digital signature σ to msg M. encrypts <M, σ> and sends it to Y. where, it is decrypted. which one of the following sequence of keys is used for the operation?

(a) Encryption: X Private Key followed by Y Public Key.
Decryption: Y Private Key followed by X Public Key

* to generate these keys (pair), → RSA Algo.

* RSA ALGORITHM

(i) It is used to find out pairs of private & public keys.
(ii) Select 2 large prime numbers

1. $P=3, Q=11$
2. find $n = P * Q \Rightarrow 33$
3. find out $\phi(n) = (P-1) * (Q-1)$
$\Rightarrow 2 * 10 \Rightarrow 20$
4. Choose e such that:
$1 \leq e \leq \phi(n)$ (<)
where, e is co-prime to $\phi(n)$

$\therefore \gcd(e, \phi(n)) = 1$

$\gcd(e, 20) = 1$.

eg: $e=3$

5. Calculate d such that $d * e = 1 \bmod \phi(n)$

$d * e = 1 + \underbrace{K * \phi(n)}_{\text{mult of } \phi(n)}$

eg: $\underset{d}{7} * \underset{e}{3} = 1 + (1 * 20)$

* Public Key = (e, n)
Private Key = (d, n)

Page No.: 206

16/6/19
Sunday

Lecture 6

Q (GATE) Using RSA Algorithm, 2 prime no.s are selected as : 13, 17.

public Key → 35 (e,n)

Find out private Key? (d,n)

Soln:

1. $n = 13 * 17 = 221$
2. $\phi(n) = 12 * 16 = 192$.
3. $e = 35$. $1 < 35 < 192$ ← (satisfied). → valid.
4. $\boxed{d * e = 1 \bmod \phi(n)}$

∴ $d * e = 1 + K * \phi(n)$

$d * 35 = 1 \bmod 192$. → rem.

↳ $(d*35) \% 192 = 1$

eg: $d * 2 = 1 \bmod 7$.
→ 4

$8 \% 7 = 1$

Page No. 207

(a) 11 (c) 15
(b) 13 (d) 17

(11 x 35) − 1 is it div by 192.

(a) $385 \% 192$

Q (GATE): The value of $3^{51} \bmod 5$ is ____.

Fermat's theorem $\boxed{a^{p-1} \bmod p = 1}$

$\boxed{a^{p-1} \cong 1 (\bmod p)}$

2^{16}

$a^{17-1} = 1 (\bmod p)$. [p = 17]

$2^{16} = 1 (\bmod 17)$.

$2^{16} = 65,536 \% 17 = 1$.

⇒ $3^{51} \bmod 5$

$(3^{5-1}) = 1 (\bmod 5)$

⇒ $3^{5-1} \bmod 5 \cong 1$.

⇒ $81 \% 5 = 1$

− $\boxed{2^{27} \bmod 7. \quad 2^6 \bmod 7 = 1}$ (Fermat's th.)

$\boxed{(3^4) \bmod 5 = 1}$

$3^{51} \bmod 5$

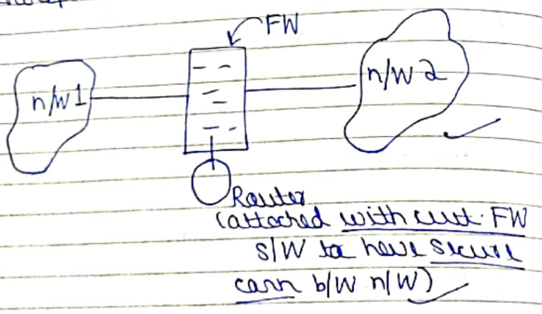
$(3^4)^{12} * 3^3 \bmod 5$

$1 * 27 \bmod 5$

$\Rightarrow \underline{\underline{2}}$

* FIREWALLS

(i) Firewall is a security software that will be used to have a secure communication/transfer between 2 networks



(ii) There are 2 types of implementing S/W :

(1) Packet filters
(2) Application gateways

(iii) Both types of firewalls will have the following characteristics :

(1) All the traffic b/w the n/w, will be passing thru the firewall.
(2) Only authorised traffic shall be allowed to pass to the network.
(3) Firewall shall be using strong internal algorithms.

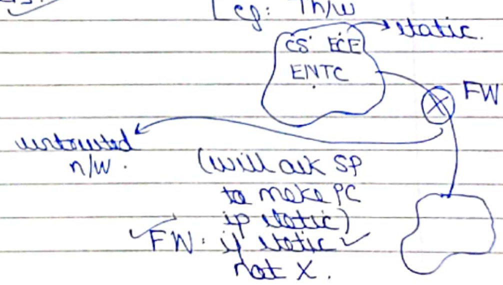
* PACKET FILTERS

(i) PF uses certain criteria to filter out the packet. (selectn criteria)

eg:

Rule : · based on ip address (range)
· protocol
· port no. (80 accept eg:)
etc...

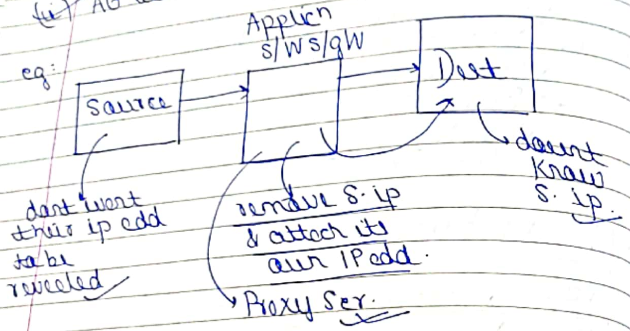
Action fields [Based on rules]. : if/else [allow/not allow]
eg: (in IP, if else condn)
[eg: Th/w



Application Gateways VI

(i) AG works as $\boxed{\text{proxy Server}}$

eg:



(ii) AG remove the IP addr of the source, attach its own IP address & send it to destn. Destination will not know, where the packet has been coming from; [creates a new connection b/w the AG & destination and terminates the connectn b/w source & dest.

[C4 – 6M min]

IPV4 ADDRESSING, BASICS OF IPV6

⇒ Basics:

$\overset{32}{1}\,\overset{16}{0}\,\overset{8}{1}\,\overset{4}{1}\,\overset{2}{0}\,\overset{1}{1} \longrightarrow 45$

$10110100 \longrightarrow 180$

Q. Convert 192

128	64	32	16	8	4	2	1
1	1	0	0	0	0	0	0

224: $(1\ 1\ 1\ 0\ 0\ 0\ 0\ 0)_2$ (128 64 32 16 8 4 2 1)

238: $(1\ 1\ 1\ 0\ 1\ 1\ 1\ 0)_2$

158: $(1\ 0\ 0\ 1\ 0\ 1\ 0\ 0)_2$ X

→ $(1\ 0\ 0\ 1\ 1\ 1\ 1\ 0)_2$

198: $(1\ 1\ 0\ 0\ 0\ 1\ 1\ 0)_2$

⇒ if 2 bit Bno → 00, 01, 10, 11
$\underbrace{\quad}_{2^2 \text{ combn}}$

if 3 bit → 8. (0 to 7) → $2^3 - 1$

if 8 bit → 256 combinations
[0 – 255] : Value range.

* IP Address

(i) IP address is a logical address at network layer level.
(ii) It is also called as Network Service Access Point.
(iii) IP addr. is used to identify System in the n/w.
(iv) IP addr is having 32 bit binary no. divided into 4 octates; where each octate consists of 8 bits (1 byte).

– 32 bit [Sip/Dip add → iP Header]

8 bit	8 bit	8 bit	8 bit
(0–255)	(0–255)	(0–255)	(0–255)

(v) if val > 255 ⇒ invalid V. (in any of the octate)

* Length of iP addr: 32 bit.

$\boxed{2^{32} \text{ ip Addresses is possible}}$

(vi) For easy operation, this is divided into multiple classes. Each class having its own range of iP addresses.

IANA – Internet Acc. N/W Authority.

Prev: diff to manage all IP add. [classful add]
→ divided into classes.
{Class A, Class B, Class C, Class D, Class E}

(vii) The range of the iP addresses of diff classes are:

Class A : 0.0.0.0 to 127.255.255.255

Class B : 128.0.0.0 to 191.255.255.255

Class C : 192.0.0.0 to 223.255.255.255

Class D : 224.0.0.0 to 239.255.255.255

Class E : 240.0.0.0 to 255.255.255.255

⇒ 2^{32} ip addr are possible.

exactly half of ip addr. ⟶ Class A.

Class A = 2^{31} $(2^{32}/2)$		
Class B = 2^{30} $(2^{31}/2)$	Class C = 2^{29} $(2^{30}/2)$	D / E

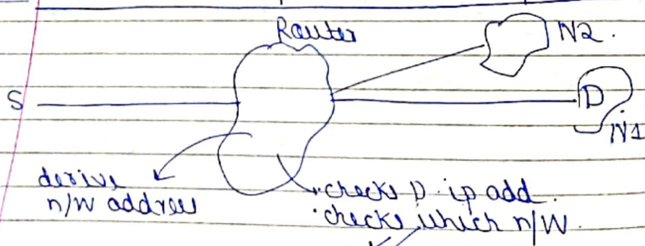
2^{32} ip Addr.

Class A = 2^{31}
Class B = 2^{30} } ⟶ VI
Class C = 2^{29}

if 3 bit b/w ⟶ 8 comb.

⟶ 0 0 0 ⟶ 1 bit kept constn. ($2^2 = 4$ comb.)
0 0 1
0 1 0
0 1 1

NOTE: if 32 bits, 1 kept constant, no. of combn = 2^{30}



* CLASS A

Given an ip Add ⟶ derive n/W address
derive host address.

* Class A.

⟶ rem 7 bits revolv.

[0] ___ . ___ . ___ . ___

⟵ N/W ⟶ | ⟵ Hosts ⟶

1st bit is const.

2^7 = 128 n/w possible

⟶ 2^{24} Hosts in each network.

[0 to 127].

Class A:

$\boxed{2^7 * 2^{24} \Rightarrow 2^{31} \text{ hosts/ip}}$

No. of n/W possib:

NA ⟶ 0.0.0.0   1.0.0.0   127.0.0.0

2^{24} 2^{24} . . . . . . 2^{24} ⟶ 127.255.255.255

0th   1st   127th n/W.

BA ⟶ 0.255.255.255   1.255.255.255

1st ip Address: 0.0.0.0.

last ip Addr: 127.255.255.255.

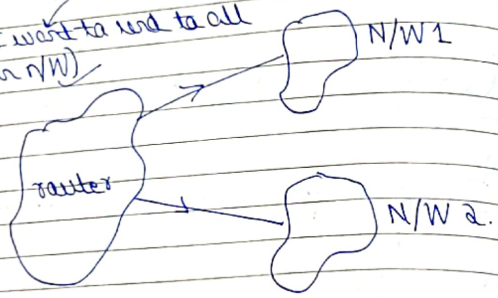
(any 127 Series → local ...

Page No.: 216

⇒ The first ip add in any of the n/w = N/W address

⇒ The last ip add in any of the n/w = broadcast address

(if I want to send to all ip add in n/W)



1.255.255.255 (rec. ip add.)

clss A, and router will know that it needs to braudcast.

→ ~~First octate & last octate~~

⇒ first n/w & last n/w in Class A are special → not used

⇒ first ip address & last ip address in each network is special (n/a to router) → given to local host.

→ NA
→ BA

Page No.: 217

Special ip Address → no. of rem. n/WS.

$2^{24} + 2^{24} + (126 * 2)$

Useful: (given to end user)

$(2^{24} - 2) * 126$

Q) What is the n/w add and Broadcast add for 100.127.224.5

∵ 100 = Class A.

⇒ N/W Addr: 100.0.0.0

Broadcast Addr: 100.255.255.255

* Class B.

Class B: 128.0.0.0 to 191.255.255.255

|—— N/W ——||—— Host ——|

10 ____ . ____ . ____ . ____

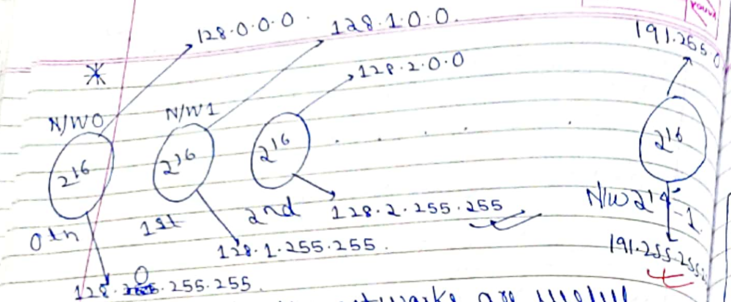
1st 2 bits const.

14 bit rev. ⇒ 2^{14} n/w possible

16 bits → 2^{16} hosts in each n/W.

2 octates repr. n/w

⇒ $\boxed{2^{14} * 2^{16} = 2^{30} \text{ hosts}}$ → Class B.



⇒ In these all the networks are useful

(In each n/w, 1st & last spcl)

Special

$2 * 2^{14} \Rightarrow 2^{15}$ ip

Useful

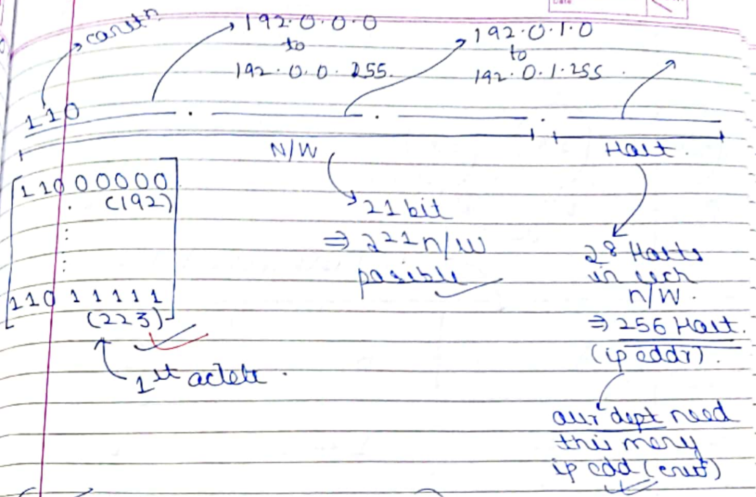
$(2^{16}-2) * 2^{14}$

* CLASS C

- first 3 octets → n/w
  last octet → host

192.0.0.0 to 223.255.255.255.

[Lab starts from 192]



Special

$2 * 2^{21} \Rightarrow 2^{22}$ n/w.

Useful:

$(2^8-2) * 2^{21}$

Class C

2^8 2^8 2^8 .... 2^8

0 1st 2nd $2^{21}-1$

IMP Table [Classful mechanism]

Class / Statistic	A	B	C
No. of n/w bits	7	14	21
No. of host bits	24	16.	8
No. of n/ws	$2^7=128$	2^{14}	2^{21}
No. of hosts in each n/w	2^{24}	2^{16}.	$2^8=256$
Total no. of Spcl ip	$2^{24}+2^{24}+(126*2)$	$(2*2^{14})$	$(2*2^{21})$
Total no. of useful ip	$(2^{24}-2)*126$	$(2^{16}-2)*2^{14}$	$(2^8-2)*2^{21}$

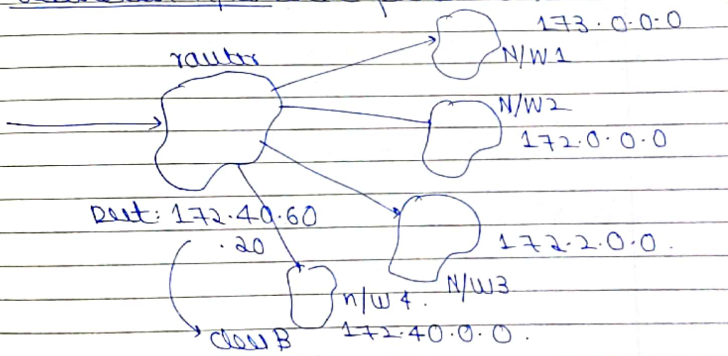
* Special Addresses:

1. N/W Address.
2. Directed Broadcast Address → Broadcast Add.
3. Limited Broadcast Address.
4. Loop Back address.

① N/W Address

(i) When packet is coming to the router, router conn. to mult. n/ws.
(this Packet sent to which n/w →) n/w

(ii) The n/w address will be derived from IP addr.
(iii) N/W address is used by router to find the destination n/w where packet has to be sent.



$\underbrace{172.40}_{\text{n/w Addr.}}$ 1st add: 172.40.0.0 (n/W Add)

∴ router needs to be programmed.

how to get n/w address?

∴ Router uses Subnet mask

(to divide IP add → n/w add, → host add)

⇒ Router uses Subnet mask to determine n/w address and host address within the n/W.

⇒ A Subnet Mask is a 32 bit binary no. which divides an IP Address into :

VI

n/W Add.   Host add.

* ⇒ In Subnet Mask, the octets which represents the network bits will be 1s and host bits will be rep. by 0s.

*Default Subnet Mask :

Class A : 11111111.0.0.0 (255.0.0.0)

Class B) : 255.255.0.0.

Class C : 255.255.255.0.

* What router do?

1) Router takes dest ip : 172.40.60.20.

look into subnet mask

∴ not there, look into d. SM.

Class B

2) will do and operⁿ with Subnet Mask :

255.255.0.0

VI

Router IP : 172.40.60.20. (AND)

Subnet Mask : 255.255.0.0

172.40.0.0 ⇒ n/W Address.

② Directed Broadcast Address → to all hosts in the other n/w.

(i) If a msg has to be sent to all the Systems in the corresponding network, then directed broadcast Address shall be used.

(ii) In General, the last ip address of the n/W will be Directed Broadcast Address.

(iii) for eg :

Router ip : 172.40.60.20 → Class B

The n/W Address will be : 172.40.0.0.

& broadcast Add. will be (which is DBA) : 172.40.255.255.

③ Limited Broadcast Address

(to all hosts in the same n/w.)

(i) If a msg has to be sent/broadcasted to all the systems in the same network, then it is called limited Broadcast address.

(ii) The default value of LBA is:

$\boxed{255.255.255.255}$

ip header.

150.20.40.60	255.255.255.255	Payload

Source IP.

→ 150.20.255.255

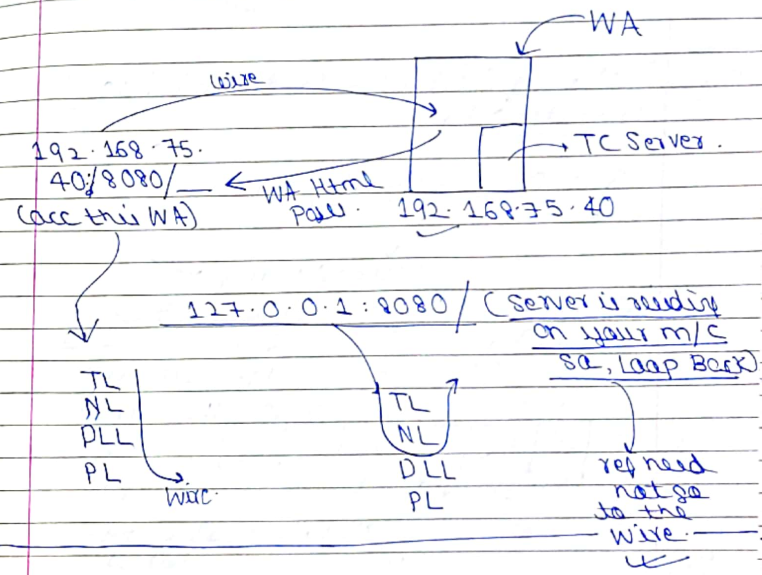
(we can do DBA of source ip too).

Q diff b/w DBA & LBA

④ Loop Back Address

(last n/w of class A)

(ii) Loop Back Add is a special IP address: (127.0.0.1) that is designated for software loopback interface of corresponding machine.



* Subnetting

→ The process of dividing a larger n/W into smaller n/W is called subnetting.

→ Advantage: more easy operation.

Disadvantage: wastage of address will be more.

eg. A class C network is divided into 8 subnetworks.

Class C → 2^8 hosts/n/W
↓
8 subnetworks

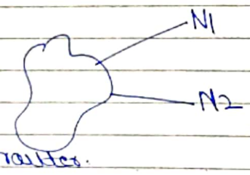
∴ No. of hosts in each subnetwork $= \frac{2^8}{2^3} \Rightarrow 2^5 \Rightarrow \underline{\underline{32}}$.

→ Sub n/W Add (1st Add)

	000	001	010	011	100	101	110	111
class C	~~32~~	32	32	32	32	32	32	~~32~~
	0th	1st	2nd	3rd	4th	5th	6th	7th

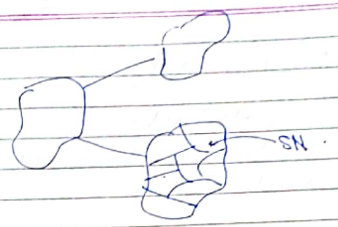
→ Broadcast add. within SN. (last Add).

Prev.



- Identify n/w add.
- host add. within n/w

Now



- N/W address
- find out Sub n/W address.
- within Sub n/W, find host address.

Stand: $\boxed{\text{When bigger n/W is divided: 1st SN/W, \& last SN/W not used}}$ → Convert?

Useful $= 30 * 6 = 180$.
Special $= 32 + 32 + 2 * 6 \Rightarrow 76$.

* Max no. of Hosts possible ⇒ 180 } Useful
Max no. of Subnets possible ⇒ 6
Max no. of hosts possible in each Sub n/W ⇒ 30.

* Any class divided into any no. of Subnets.

(SM used to get N/W Add.)

Here, SM to get S n/W Add.

Date:

Page No. 229

* Design Subnet Mask

(Divide hosts)

S-1: Find out no. of bits required for subnetting. (for repres. a subnet)

$\log_2(\text{no. of Subnetworks})$

$\Rightarrow \log_2 8 \Rightarrow \underline{\underline{3}}$

Each octets → 3 given to SN. → 5 rem. ⇒ 32

S-2: Borrow Subnet id bits from host bits of the n/W.

Class C: 256 hosts divided (2^8)

Take host bits: 8. (Prev all 8 bits we used)

→ 3 bits (Subnet) [1S] → 8 Subnets

→ 5 bits (Host in each [0s] subnet) → 2^5 → 32 Hosts / SN/W.

S-3: Take the default Subnet mask of the corr. class.

Class C: 255.255.255.0 ← Default SM.

- Subnet bits are represented by 1s, & host bits are rep by 0 in SM.

What means:

255.255.255.$\overbrace{11100000}^{\text{Host}}$

⇓

255.255.255.224

Q. Class B is divided into 256 subnetworks

(i) What is the subnet Mask?

(ii) Max possible hosts in each Sub n/W.

(i) Class B: 255.255.$\overbrace{0.0}^{\text{Host bit}}$

16 (host bits in C-B).

→ 8 bit (Subnet)

→ 8 bit (Host)

255.255.11111111.0

$\boxed{255.255.255.0}$

1. $\log_2(256) = 8$ → SN bits.

Class B: out of 16 bits (host)

8 bits [IS] (SN) — 8 bits [RS] (Host).

255.255.0.0 → default SM.

∴ 255.255.255.0

2. $\frac{2^{16}}{2^8} \Rightarrow 256 - 2 = 254$ max usable → hosts in each SN

∴ (254 * 254) → No. of SN usable hosts.

Q. Class A is divided into 4096 hosts.

1. 24 → $\log_2(4096)$

12 bit (SN) — 12 bit (Host)

2^{12} subn/w

255.0.0.0

255.255.11110000.0

255.255.240.0

2. $(2^{24} - 2)$: No. of SN ✗

$\frac{2^{24}}{2^{12}} \Rightarrow (2^{12} - 2)$: Useful SN

$(2^{12}-2) * (2^{12}-2)$ → Max no. of possible IPs.

$\Rightarrow (2^{12} - 2)$ → (Max. Hosts in each subn/w)

Q. Class C divided into 16 SNs.

1. 4 bit (SN)

8 bit (Host)

4 bit (SN) — 4 bit (Host)

255.255.255.0

128 64 32 16 8 4 2 1

255.255.255.11110000

↓

$\boxed{255.255.255.240}$ → SM (default)

Which of the following is valid SNM?

(a) 255.255.255.15 ✓

(b) 255.255.255.195 ✓

(c) " . " . " . 160 ✗

(d) " . " . " . 71 ✓

any order → Valid.

$\boxed{41s, 40s}$ ✓

(a) 255.255.255.00001111 ✓

(b) 255.255.255.11000011 ✓

(c) 255.255.255.160 X

(d) 255.255.255.71 ✓
→ 01000111

Q. If a class B network on the internet has a subnet Mask of 255.255.248.0 what is the max no. of hosts/subnet?

Class B.

(a) 1022
(b) 1023
(c) 2046
(d) 2048.

VI

Soln.

Class B

255.255.0.0. (SM of Class B)

255.255.11111000.0. (Given)

16 bits (Host bit of C-B)

5 bits (SN)

11 bits (Host)

2^{11} possible

$\Rightarrow (2^{11}-2) \Rightarrow 2046$.

Q. A address of class B is divided into subnets with 6 bits Subnet No. What is the max no. of subnets and max no. of hosts within each subnet?

Soln.

(c) 62, 1022

16

6 (6 1s) 10 (10 0s)

· Max no. of subnets = $2^6 - 2 \Rightarrow 62$ ✓

· Max no. of hosts / subnets = $2^{10} - 2 \Rightarrow 1022$.

Q. A organization has Class B network and divided into 64 Subnetworks each for 1 department. The SM would be:

(a) 255.255.0.0

(b) 255.255.64.0

(c) 255.255.128.0

(d) 255.255.252.0.

255.255.0.0.

Soln.

16

$\log_2(64) = 6$.

6 (6 1s) 10 (10 0s)

255.255.0.0

→ 255.255.11111100.0 ⇒ 255.255.252.

class C

* ANALYSIS OF SUBNET MASK

Subnet Mask	(Total Subnets) Useful	(Total Subnets) ~~Useless~~ (Special)	(Total hosts in each SN) Useful	(Total hosts in each SN) Spcl	Total no. of Hosts in all Subnets
255.255.255.0					(not divid)
255.255.255.128	0	2	126	2	0
255.255.255.192	2	2	62	2	124
255.255.255.224	6	2	30	2	180
255.255.255.240	14	2	14	2	196
255.255.255.248	30	2	6	2	180
255.255.255.252	62	2	2	2	124
255.255.255.254	126	2	0	2	0
255.255.255.255			(not divid)		

Limited Broadcast Add.

255.255.255.128.

255.255.255.10000000

8 bits → 1 bit (SN), 7 bit (Host)

1 bit (SN) → $2^1 = 2$ SNs

7 bit (Host) → 2^7

8b → 2b, 6b

⇒ 1 1

~~(b)~~ 2 1s, 6 0s

~~(c)~~ 1 1 1

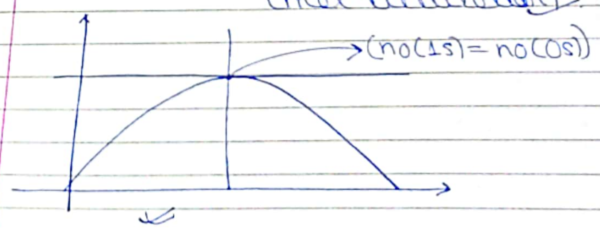
1 1 1 1 1 1

⇒ 255.255.255.240 → class C

240 → 11110000

$\boxed{\text{VI}}$

when no. of 1s = no. of 0s (max utilisation)



⇒ 255.255.255.0 → class B.

(At this, SM; Max. utilisation)

24 → 12, 12

⇒ 255.255.240.0 → Class A

* 192.168.75.0 → class B
$\underbrace{192.168.75}_{N/W}$

divided into Subnetworks, each having 8 hosts

1st SN

192.168.75.0
1
2
⋮
7

Q Each Subn/w → Range of ip addresses (What's the range?)

Q A ISP has given you class C network range having network address as : 209.50.1.0

(i) Find out Subnet Mask if it is divided into 20 Subnetwork.

(ii) Find out the range of IP address in each Subnetwork.

(i) 8 bit (Host) $\lceil \log_2 20 \rceil = 5$

5 bit [1s] (SN) 3 bit [0s] (Host).

→ 209.50.1. 11111000 ~~X~~

[209.50.1.248] ~~X~~ SM.

[255.255.255.248] . → SM (default)

255.255.255.0 (default SM of class C)

(ii) Range : 255.255.255.11111000 (SM)

2^3 right most 1.

8 hosts / S n/w (Trick).

Ranges

0th SN → 209.50.1.0 → (209.50.1.0 – 209.50.1.7)
1st SN 209.50.1.8 → (209.50.1.8 – 209.50.1.15)
1.16
1.24
1.32
1.40

19th SN 20 Subnets

Q Class B having network add of 172.16.0.0 with 60 Subnetworks.

Soln. $\log_2 60 \approx 6$

(i) 16 → 6 (1S) (SN), 10 (0S) (Host)

⇒ 255.255.0.0 → default SM of C-B

255.255.111111 00.0 → 1024

$\boxed{255.255.252.0}$ → SM. Trick.

(ii) Range

4×3 $4\times(n-1)$ 4×9

172.16.0.0 → (172.16.0.0 to 172.16.4.3

172.16.0.0 → (172.16.0.0 to 172.16.3.255)
172.16.4.0 → (172.16.4.0 to 172.16.7.255)
172.16.8.0 → (172.16.8.0 to 172.16.11.255)
172.16.12.0

172.16.236.0
(60 subn/ws)

VI

0.0 to 0.255 : 256
1.0 to 1.255 : 256
2.0 to 2.255 – 256
3.0 to 3.255 – 256

1024 host/n/w → 2^{10}

Q Subnet Mask of a class C Address is 255.255.255.240. The ip add. 192.168.1.58 is in which SN?

Soln. N/W Add → 192.168.1.0

8 bit → 4 bit (SN), 4 bit (Host) → 16 (Trick)

255.255.255.11110000

n/w address given

0	192.168.1.0	(192.168.1.0 – 192.168.1.15)
1	192.168.1.16	(16 – 31)
2	192.168.1.32	(32 – 47)
3	192.168.1.48	(192.168.1.48 – 192.168.1.63)
	192.168.1.64	

Subnetwork No = 3

Q. We need 300 ip addresses

If I take class C ⟶ 256 (not sufficient)

So,
I take class B → 2^{16} hosts
↓
65536 hosts
− 300 ⟶ used
65236 ⟶ wasted

(V.I)

* Drawback of Classful Addressing
(many are wasted)

→ Solution:
→ To combine two classes

256 — 256
Class C   Class ~~B~~ C

[Supernetting]

* Supernetting

(a) The process of combining two / more subnetworks is called supernetting

eg. If a user req 1000 Hosts
↓
Class B n/w shall be taken if we use classful addressing which leads to wastage of 64536 hosts

65536 − 1000 = 64536 ⟶ X [waste]

⇒ The alternative Method is :

(To combine 4 Class C networks and create a separate network) ⟶ supernet

[256 Class C — 256 Class C — 256 Class C — 256 Class C] ⟶ Supernet

Subnet mask ⟶ to be able to find out n/w address
└→ 2 Ways

N − 1

1st Method

1. $\log_2$ combined n/W $= \log_2 4 \Rightarrow$ 2 bits
2. Class C $\longrightarrow$ 8 bit

8 bit → 2 (Sub), 6 (Host)

$\underbrace{\text{2 (Sub) \quad 6 (Host)}}_{\text{divided into sub n/W}}$

8 bit (Host) $\xrightarrow{+2 \text{ bits}}$ 10 bits (Host)

3. 32 bits → 22 bit, 10 bit

2nd Method (preferable) [VI]

1. $\log_2$ no. of hosts $= \log_2 256 * 4 = \underline{10}$
2. 32 bits → 22 bits (n/W), 10 bit (host)

[Classless Inter Domain Routing]

* CIDR → classless

ip address / no. of n/W bits

* 172.48.7.5 / 22 [VI]

→ ip add is in a n/w which is consis. of 2^{10} Hosts

· What is subnet mask when comb. 4 Class C n/ws

↓

255.255.111111 00.0

255.255.252.0

Q. A Host has been assigned an address 140.16.79.19/21 → CIDR

(i) No. of hosts in this n/W

(ii) What is the subnet mask for this n/W

Soln

32 bits

(i) 21 bits (n/W) [21 1s]; 11 bit (host) [11 0s] → 2^{11} hosts

∴ 2048 hosts → available in n/W

32 → 23 (n/W), 9 bits (host)

∴ $\boxed{/21 \longrightarrow /23}$ n/W
(after div into 4 SN)

Range:

1) 128.119.152.0/23 – 0th

SM ⟶ 255.255.111110|1|0.0 — ② in 3rd octet
⟶ (255.255.254.0)

2) 128.119.154.0/23 – 1st

3) 128.119.156.0/23.

4) 128.119.158.0/23. → till 29 hosts.

SN add space.

1) 152.0 – 152.255 }
153.0 – 153.255 } 256 + 256 ⇒ 512

∴ 2^9 (512) hosts per sub n/W.

Q. Consider a subnet with ip address:
128.119.40.~~128~~ 64 /26
This network is divided into 4 Sub Networks
(ii) What is the subnet mask for each Subnet
(iii) Subnet add space for each subnet.

Soln
(255.255.255.11000000)
[255.255.255.192] → SM (n/W)

32 ⟵ CIDR → 26 1s n/W, 6 0s Host.
1) (This n/W consists of 26 hosts)

⇒ $\frac{26}{22}$ ⇒ 2 4 ⇒ 16 Hosts / subnet

32 → 28 (n/W) 1s, 4 (Host) 0s.

/26 ⟶ /28. n/W

⑯ [in 4th octet]
255.255.255.11110000

⇒ (255.255.255.240) → SM [for each SN]

128.119.40.~~128~~ 64 /28
128.119.40.80/28
128.119.40.96/28
128.119.40.112/28
} SN add space.

Divide → Increase (↑)
Combine → Decrease (↓)

CIDR Prefix

Q. /26 $\xrightarrow{\text{divide into 4 SNs}}$ /28.

/26 $\xrightarrow{\text{combine 4 SNs (aggregate)}}$ /24 ✓

VI.

Q. Suppose the following 4 SubNetworks are aggregated into single network consisting of:

139.179.192.0/20 → 2^{12}
139.179.208.0/20 → 2^{12}
139.179.224.0/20 → 2^{12}
139.179.240.0/20 → 2^{12}

Find CIDR prefix that should be used in order to advertise aggregated n/W?

Soln:

/20 → /18.

139.179.192.0/18 ← Aggregated n/W address.

VI

$2^{12} \times 2^2 \Rightarrow 2^{14}$ hosts ✓

32 → 18 (n/W), 14 (host)

Q. I have /22 n/W divided into 2^{12} Subnet.

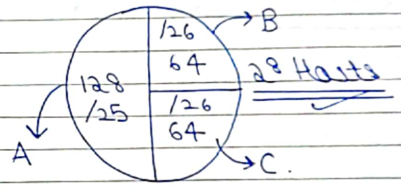
$\frac{16 \text{ hosts}}{2^{12}}$ ✗ (not poss.)

∴ /40 → not poss in IPV4.

✱ VLSM [Variable Length Subnet Mask]

(Q) I have a /24 n/W.

[In each SN, diff no. of IP add.]



32 bit → 24, 8 (host) = $\frac{2^8 \text{ Host}}{2}$ → 2^7 Hosts

A: 32 → 25 (n/W), 7 (host) ✓

$2^7 = 128$
10000000

Subnet Mask for A : 25

255.255.255.128/25

2^7
10000000

Subnet Mask for B : 26 ; C

2^6
11000000

255.255.255.192/26

→ **New Address range:**

N/W Add: 192.168.75.0 [given]

Start from A: (128)

A 192.168.75.0/25

B 192.168.75.128/26

C 192.168.75.192/26

Start from B:

B 192.168.75.0/26

C 192.168.75.64/26

A 192.168.75.128/25

(Order can Vary).

Q An ISP has following chunk of CIDR based IP addresses available in it, in the form of
245.248.128.0/20

The ISP has given half of the IP add to org. A, quarter to org B & quarter kept for itself.

Which of the following is valid allocations to A and B?

(a) 245.248.136.0/21 & 245.248.128.0/22

(b) 245.248.128.0/21 & 245.248.128.0/22

× (c) 245.248.128.0/22 & 245.248.132.0/21

× (d) 245.248.136.0/24 & 245.248.132.0/21

Soln:

2^{12} Hosts

$\boxed{A : 2^{11} \text{ Hosts}}$ $\boxed{A : /21}$

32 → 21, 11

SM: 255.255.248.0 (A)
11111000 → (B)

$\boxed{B : 2^{10} \text{ Hosts}}$ $\boxed{B : /22}$

32 → 22, 10

SM: (B) 11111100 (2^{10}) → (A)

(c) 255.255.11111100.0

V-I

Start from A:

A: 245.248.128.0/21
B: 245.248.136.0/22
C: 245.248.140.0/22

Start from B

B: 245.248.128.0/22
C: 245.248.132.0/22
A: 245.248.136.0/21

4. BASICS OF IPV6

(i) Using IPV4 addressing System, only 2^{32} ip Addresses can be possible. In the growing internet World, the range of ip Addresses in IPV4 is ~~also~~ not sufficient.

(ii) To avoid this, IPV6 has been introduced consisting of:
128 bit Address generating 2^{128} IP Address.

⇒ Features of IPV6:

(1) Faster Forwarding / Routing.
(2) Security.
(3) Mobility.
(4) No broadcast.
(5) Vast Address Space

(only 5% of the System has been moved to IPV6)

⇒ How is add. translation done? → Basics

Binary	Hex	Decimal
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

⇒ IPV6 address is 128 bit Address consisting of 8 double octates (16) separated by : (colon)

7CF1 : ___ : ___ : ___ : ___ : ___ : ___ : ___
16 bit, 16 b, 16 b, 16 b, 16 b, 16 b, 16 b, 16b

0111 1100 1111 0001
4b 4b 4b 4b

↓ 16b (1) double octate

(7CF1)

* To avoid the length of IPV6 address, some of the rules have been proposed.

eg:

2001 : 2A53 : 0000 : 0000 : 4372 : 0000 : 78EF : 0000 . (length ↑)

IPV6 address

replace null → 0.

2001 : 2A53 : 0 : 0 : 4372 : 0 : 78EF : 0

2001 : 2A53 : 0 : 4372 : 0 : 78EF : 0 .

eg2: 2001 : 2A53 : 0000 : 0000 : 0000 : 0000 : 0000 : 78EF .

↓

2001 : 2A53 : : 78EF .

5 : missing .
∴ contn 5 double octate 0s .

R3 eg3 :

0783 : 038E : : → rest are 0s .

↓ (del 0 at start)

783 : 38E : :

* Conversion of IPV4 to IPV6 or viceVersa :

for commn b/w IPV6 dev. & IPV4 dev.
compatibility ↑

IPV4 ⟶ 32 bits . IPV6 ⟶ 128 bits .

2 double octates .

____ : ______ : : → IPV6 add .

all 0s .

VI

* SUBNETTING IN IPV6

IPV6 address / ___
↳ No. of n/W bits

Suppose [2853: 7EF2: /48]

128 → 48 (n/W), 80 (host)

→ This ip add. is in n/W which consists of 2^{80} Hosts

[this big n/W not imp. today]

⇒ [2853: 7EF2:: /48]

↳ divided into 2^{12} subnets

→ /48 ⟶ /60 n/W.

* IPV6 Header contains Source & destination IPV6 address which is consisting of 128 bits each.
- It consists of Version field (4 bits).

Version: It indicates the Version of IP Packet

- Priority field: it indicates the priority of the packet
- Flow Label Field (20 bits)
  → this indicates the Quality of Service, and to specify sequence of Packets
  (FO & MF) → IPV4
- Payload Length: The length of the data which is being sent
- Hop Limit (8b): (Instead of TTL)
  → Hop limit is used to avoid looping of the packets in network
- Opr^n of HL same as TTL field.

ETHERNET

- Thick ← 10 Base-5 → 10 mbps BW, 500m
- Thin ← 10 Base-2 → 10 mbps BW, 185 m of Cable

} Bus/Star Topology

- Twisted Pair ethernet ← 10 Base-T → Star Topology

* Ethernet uses Manchester Encoding at Physical Layer.

Normal

(i) In General, each bit is indicated by 1 signal Level

Bit 1 → High Signal Level.
Bit 0 → Low SL.

digital

Analog.

(ii) In Manchester Encoding:

each bit is represented by 2 Signal levels. that means:

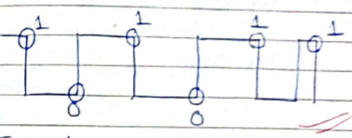
Bit 0 : (low → high)

Bit 1 : (high → low)

eg. 1 0 1 0 1 1

Normal.

1 0 1 0 1 1 Manchester.



⇒ Encoding mechanism used at PL.

(iii) The Signal levels will be represented in bauds/s. ∵ 1 bit = 2 SLs

* In Ethernet, $\boxed{\text{baud rate} = 2 * \text{bit rate}}$.

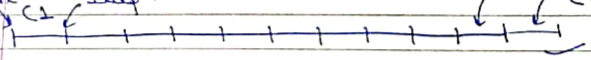
* Ethernet Operations:

- multiple Protocols

Pure aloha	Slotted aloha
(i) In Pure aloha, the transmission of the data doesnot depend upon the time	(i) In Slotted aloha, the transmission of the data depends upon the time.
(ii) Frame can be sent at any point of time.	(ii) Frame can be sent only at the beginning of time slot which uses time division multiplexing.
(iii) Pure aloha has possibility of collision.	(iii) Slotted aloha will be sending data in the corresponding time slots so no possibility of collision
(iv) Continuous time frame	(iv) Discrete time frames

*~~4~~ Slotted aloha [Pert. tym to send data]
→ time div. multiplexing.

10 sec → 10 comp.
start, stop, C1, C10, C1



- Only 1 System will send data at a time.
- Syst. need to wait for all other Comp

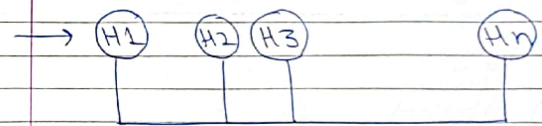
Slots, till it can again send data.
→ Discrete Time frames.

*Pure aloha → cont. time frames

⇒ In ethernet, n hosts are connected only 1 Host has to transfer data at a time

- come to LAN, within LAN many comp. how to send to a pert. comp → Ethernet

⇒ Remaining n-1 Host should be idle.



(1) If H1 wants to send data, it will prepare Ethernet Frame → Step 1

TL
NL
DLL → LLC (Log Link Control) } Sublayers.
PL. → MAC

It will multiplex the data

AL → | AH | Msg | APDU

TL | TH | APDU | TPDU

NL | NH | TPDU | NPDU

① DLL
subL
| LLC | NPDU | LLCPDU → frame (min 46B – 1500B)
| MAC | LLCPDU | MACT | → ethernet frame

Header, MAC trailer

| S·MA (6B) | D·MA (6B) | Length (2B) | : 14B of Header

46B–1500B (MTU)

4B (CRC)

Pg 7 → Q1

1. Smallest ethernet frame $= 14 + 46B + 4B = 64B$

Largest ethernet frame $= 14 + 1500B + 4B$
$\Rightarrow 1518B$

Payload % $= \frac{46}{64} \times 100$
$\Rightarrow 71.87\%$

$\frac{1500}{1518} \times 100$
$\Rightarrow 98.81\%$

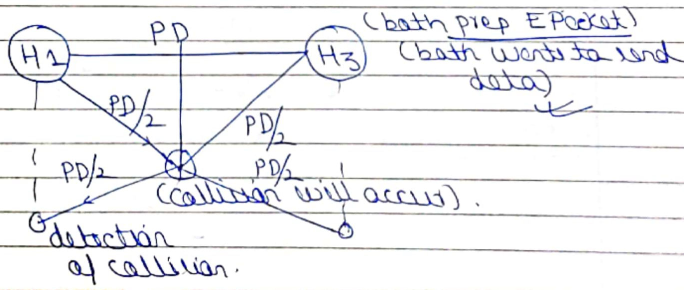
2. Min = 46B
42B recv.
4B → padding

② H1 prepared ethernet Packet ①
→ H1 checks the sense of carrier.
(whether data is flowing on wire/not).

③ If data is not flowing on the cable, H1 starts sending data

④ There is a possibility of collision, when both of the stations start sending data at the same time
∴ Its pure aloha.

⑤ When collision occurs
→ Both signals are collided.
→ Voltage increases.
→ It reverses the direction of propagatn.
→ collision signal will reach to one of the host (Source/Dest).



[within 1 PD time, collision will be detected]

(6) ⇒ Once the collision is detected, a jam signal will be sent on the n/w to alert other users. (alarm)

(7) Collision recovery Process will be started.

will be done by (Binary exponential back off Algorithm)

(BEBO Algorithm)

1. $K = \{0, 1, 2 \cdots 2^{i} - 1\}$

where,

$i = \min(10, \#\text{no. of collisions of the frame})$

2. $\boxed{\text{Waiting time} = K * \text{Slot time}}$

Q3 Pg 7.

1st collision: $i = \min(10, 1) \Rightarrow 1$

$K = \{0, \cdots 2^1 - 1\}$

$K = \{0, 1\}$ ← Hosts can select b/w 0, 1.

2nd collision: $i = 2$

$K = \{0, \quad 2^2 - 1\}$

$K = \{0, 1, 2, 3\}$

5th collision: $i = 5$

$K = \{0 \text{ to } 2^5 - 1\}$

$K = \{0, 1, 2 \cdots 31\}$

Probability (4) $\boxed{VI}$

$\boxed{1/32}$

---

eg. 2B, 2G Slot time = 5 min.
[Who will impress the girls?]

$\boxed{0}$ $\boxed{1}$

WT = 0 * 5 (won) WT = 1 * 5

---

4. Baud rate = 2 * bit rate

2 * 10 Mbps

$\boxed{\text{Baud rate} = 20\text{M bauds/S}}$

5. $PD = 2 \times 10^8$

$\Rightarrow \frac{D}{S} \Rightarrow \frac{2500}{2 \times 10^8} \Rightarrow$

$\Rightarrow 12.5 \times 10^{-6} = 12.5\ \mu sec$

Page No. 266

* CSMA/CD → Algo [Ethernet]

Collision/Carrier Sense multiple Access / Collision Detection

6. (A) (B)

1 bit delay $= \frac{1}{BW} = \frac{1}{10^7 bps} \Rightarrow \underline{0.1\,\mu sec}$

6. A ——10Mbps—— B

PD = 225 bit times
↓
225 bit delays

$\frac{225}{BW} = \frac{225}{10\,mbps} = \underline{22.5\,\mu s}$

collision detection will occur after 22.5 μs

(1) JAM Signal → dont wait till end (Send an Alert)

collision detection + TD of Jam Signal.

$1PD + \frac{l}{BW}$

$22.5\,\mu s + \frac{48}{10\,mbps} \Rightarrow \underline{27.3\,\mu sec}$

(b) K = {0, 1}

A	B	
0	0	– recollision.
0	1	– A won
1	0	– B won
1	1	– recollision.

(10) → B won

$27.3\,\mu s + (TD_B)\ \frac{l}{BW}$

$= 27.3\,\mu s + \frac{1000}{10^6}$

$\Rightarrow \underline{127.3\,\mu sec}$

[Collision detect? + Jam Transmit + Packet Transmit

7. BW = 10 Mbps.

PD = 225 bit times ⇒ 225 × 1 bit delay.

1 bit delay $= \frac{1}{BW} = 0.1\,\mu sec$

$22.5\,\mu sec \Rightarrow PD$ → Collision detect? time

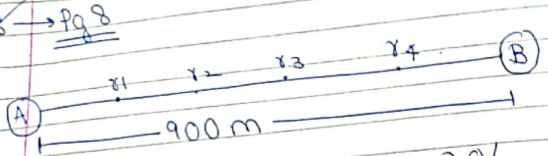
[Collision detection time + TD of JS + PD of JS]

22.5 + 22.5 + 1 bit Delay (Signal).

$45 + 0.1 \rightarrow \boxed{45.1\,\mu Sec}$

Scanned with CamScanner

8 → Pg 8



a) 1 repeater = 20 bit delay = $\frac{20}{BW}$

$\Rightarrow \frac{20}{10\,mbps} \Rightarrow 2\,\mu sec$

Repeaters → $8\,\mu sec$

$\Rightarrow PD = \frac{d}{s} + 4$ repeater delay

$= \frac{900}{200\,m/\mu sec} + 8\,\mu sec \Rightarrow 12.5\,\mu sec$

b) 1st collision = $12.5\,\mu sec$ (collision detect?)

$TD_s = \frac{10^3}{10 \times 10^6} \Rightarrow 0.1\,msec$

$\Rightarrow 100\,\mu sec$

collision detected

$\Rightarrow$ [PD + TD of Jam signal + ~~TD of Packet~~ + PD of Packet]

$\Rightarrow 12.5 + 100 + 12.5$

$\Rightarrow \boxed{125\,\mu sec}$

** Minimum Packet Size in Ethernet =

$\boxed{RTT * BW}$ → BW delay product.

why?

PD → collision occurred } [worst case]
+
PD } collision detected

$2PD * BW = RTT * BW$ → MPS

Q. Take 2 Km length broadcast LAN has 10Mbps of data rate. The signal travels along the wire at 200 m/μsec. What is the min Packet Size that can be used on the n/W?

$\Rightarrow \frac{2 \times 10^3}{2 \times 10^2} \Rightarrow 10\,\mu sec$ (PD)

$RTT = 20\,\mu sec$

$20 \times 10^{-6} \times 10 \times 10^6$

$\Rightarrow 200$ bits → Min. PS

6 6 6 7 7 7

Q. There are n Stations in Slotted ethernet LAN Each Station attempts to transmit with probability p in each time Slot. What is the probability that only 1 Station transmits in given times

(a) $nP(1-P)^{n-1}$ (c) $P * n(1-P)^{n-2}$
(b) $P(1-P)^{n-1}$ (d) $1-(1-p)^{n-1}$

Soln

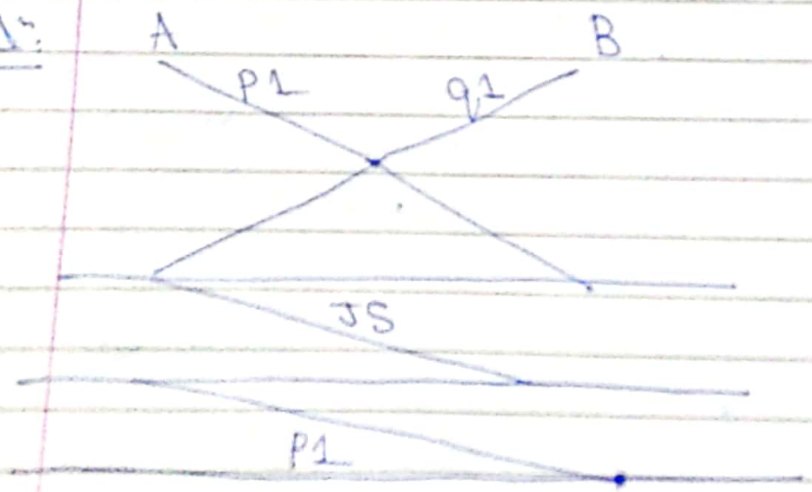
$S_1, S_2, S_3 \ldots S_n \rightarrow$ n stations
$p \quad (1-p) \quad (1-p) \quad (1-p)$

$nP(1-P)^{n-1}$

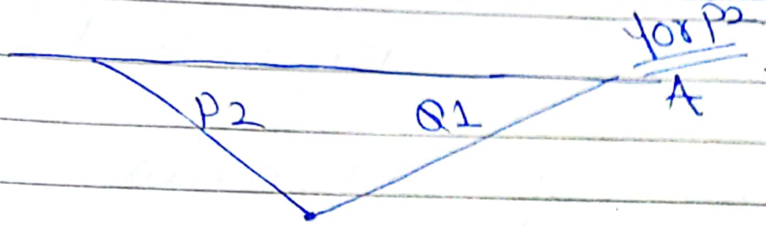
Q. A and B are 2 Stations on ethernet. Both A & B attempts to transfer a frame collide & A wins first back off race. At the end of successful transmission by A, again A and B collide. What is the probability that A wins the second back off race.

(a) 0.25 (c) 0.75
(b) 0.625 (d) 1.0

Soln:



1. Pg 198 (i) Passing URL



for P2 A: $i = \min(10, 1) \Rightarrow 1$

$K = \{0, 1\}$

for Q1 B: $i = \min(10, 2) = 2$

$K = \{0, 1, 2, 3\}$

A	B
0	0
1	1
	2
	3

(0,0) (0,1) (0,2) (0,3)

(1,0) (1,1) (1,2) (1,3)

$5/8 \Rightarrow \underline{\underline{0.625}}$

——— X ——— X ——— X ———

Scanned with CamScanner