classmate
Date
Page

Data link layer.

Sublayers : LLC : Logical Link Control
MAC

N/W
↓

LLC
MAC

DLL

LLC – Framing
LLC – Error control.
LLC – Flow Control

↓
Physical

Virtual path ⇒ dotted lines ⇒ Protocols.

FRAMING

8 Character count : no of character in the frame.

(I)

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

5-1 = 4 characters — Frame 1

Frame 2 → 7 character (8-1)

Error in this 5 or 8 can cause problems.

(II) B

(II) Byte Stuffing

↳ 8 bits

frame →

FLAG	Header	Payload field	Trailer	FLAG

01111110 ← FLAG ⇒ A frame delimeter of 1 byte data ⇒ 8 bits.

↳ denotes start and end of the frame.

Sender (Before stuffing) → (After stuffing) Reciever

A	FLAG	B

→

A	ESC	FLAG	B

In this case, 'FLAG' is part of the data in the frame also. So error occurs since FLAG is a delimeter. So to avoid this, a byte 'ESC' is stuffed in the frame.

The reciever identifies ESC as the byte stuffed so it removes it & FLAG remains part of the frame.

A	ESC	B

A	ESC	ESC	B

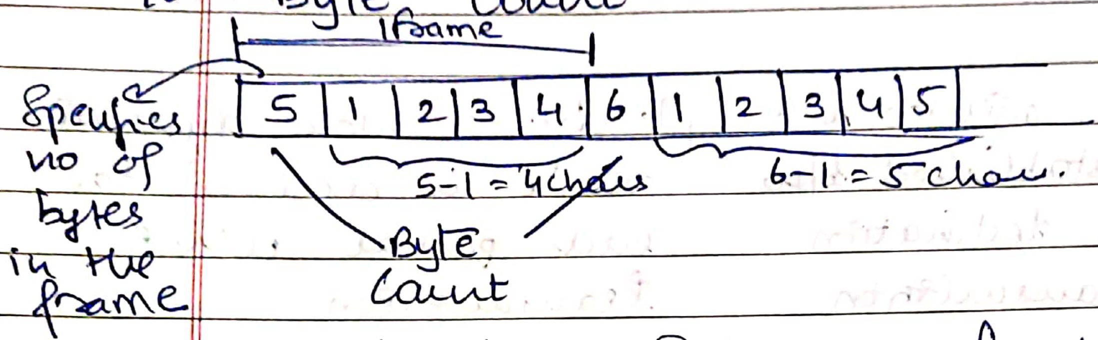
Scanned with CamScanner

1 frame:

Header	Payload field	Trailer

Framing mechanisms

1. Byte Count



- Drawback : ~~Error can lead to~~ Count can be garbled due to transmission error : 5 becomes 7 on single bit flip
No synchronization, unable to locate start of next frame.
Even if checksum is incorrect & destination is able to detect it, retransmission won't help since dest does not know how many bits to skip over to get to start of retransmission

2. Byte stuffing

FLAG	Header	Payload	Trailer	FLAG	FLAG

(end ↓ over first trailing FLAG, start ↓ over next FLAG)

Original bytes		After stuffing
A FLAG B	→	A ESC FLAG B
A ESC B	→	A ESC ESC B
A ESC FLAG B	→	A ESC ESC ESC FLAG B
A ESC ESC B	→	A ESC ESC ESC ESC B

3. Bit stuffing

framing is done at bit level instead of being tied ~~the~~ to ~~the~~ one 8 bit bytes.
frames can contain arbitrary no of bits made up of units of any size.
Each frame begins and ends with a special bit pattern 01111110 or 0x7E. This pattern is a flag byte.

original data

on the line — stuff '0' when you encounter 5 1's.
in the memory → stored in memory.
after des —

Flow control: Sender ~~can~~ ~~keep~~ systematically transmit frames faster than the receiver can accept them. This occurs when sender is running on a fast powerful computer and the receiver is running on a slow low end machine.
Approach to prevent ~~flow control~~: this Sender overwhelms the receiver, rec loses frames.

1. feedback based flow control.
→ Simplex stop and wait for noisy channel
   - Normal operation
   - Time out

$\boxed{ARQ}$

Only positive ACK.

→ ACK lost
→ ~~Packet lost~~ frame lost.

Timer times out
sender ~~then~~ retransmits the frame

to avoid duplication of frames: seq no in header of each frame is added by sender

Scanned with CamScanner

Date
Page

(II) Bit stuffing. [Better than byte stuffing ie adding 8 bits].

Data stream is of 0's and 1's, so bit stuffing is done by identifying 011111 → zero and five ones and then making the sixth bit 0 ie stuffed bit is '0'.

Reciever identifies 011111<u>0</u> as stuffed bit and removes it

| FLAG | - - - - - - | FLAG |

Errors

- Single Bit
- Multiple ~~Burst~~ bit
- Burst

• Hamming Code
• CRC

Error info is added in Trailer part so as to find out if there is any error in the data.

Modulo-2 arithmetic → XOR

Redundancy.

Sender: Recieves

Data M(x)

↓

Generating function
G(x)

↓

Redundancy check.

Redundancy check → Accept / Reject

G(x) ⇒ 0 1 1 0 1 — 3 0's added to end of data bits.

G(x) cannot start with '0'.

```
          100
1101 ) 1001000000
       1101↓↓
       0000000
```

4 bits so add 3 0's to the end of the data bits.

No of redundancy bits = No of data bits − 1
or
degree of polynomial.

classmate
Date
Page

```
          111101
     ___________
1101 ) 100100000
       1101
       -----
       01000
        1101
       -----
        01010
         1101
        -----
         01110
          1101
         -----
          00110
           0000
          -----
          01100
           1101
          -----
           0001  → Remainder
```

1101

0001$\boxed{00}$ → Remainder.

$M(x) + R(x)$

remainder

The reciever already has the GF. So it has 1101

100100001 is sent to the reciever.

∴ Reciever divides $M(x) + R(x)$ with 1101 and checks the remainder.

if remainder $= 0$, no error.

if " $\neq 0$, error is there.

Scanned with CamScanner

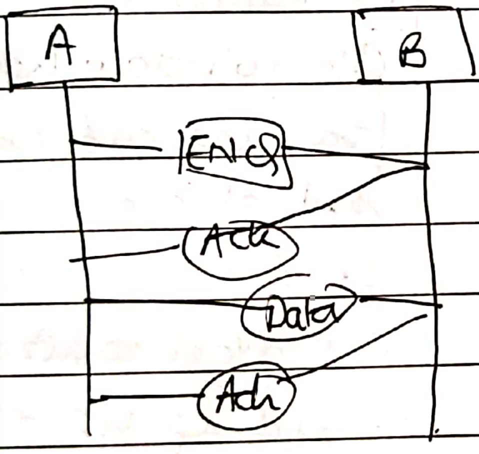
FLOW CONTROL.

Rate at which sender should send data such that the reciever can cope up.

Line discipline

Enquiry | Acknowledgement

+ve Ack -ve NAK



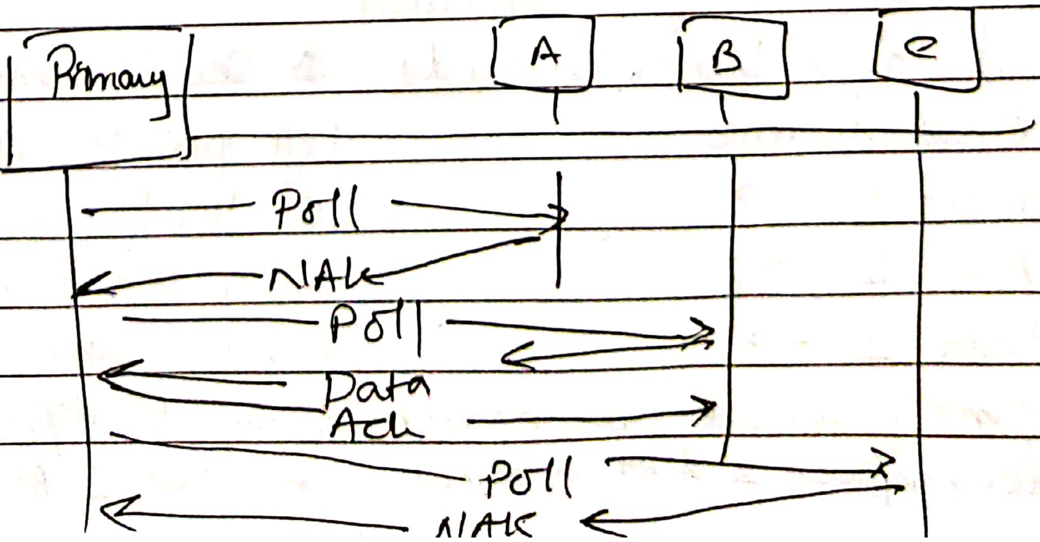
Enquiry : A enquires if B is ready for connection

Ack : B acknowledges when it is ready.

Multiline discipline

Poll

secondary.



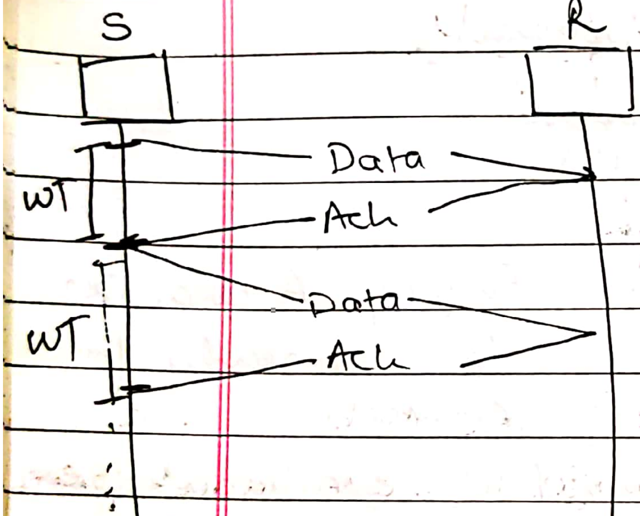
Scanned with CamScanner

Flow Control

Stop and wait ARQ mechanism

Send one frame at a time

- NO NACK.



- Sender sends data, waits for Ack & then sends next frame with different frame id.
- If frame is with errors, receiver identifies duplicated data & hence discards that frame.

Data 0 →

Ack 1 ←

Data 1 →

- Frame recieved has errors: timer expires & data is resent

Sliding window

Several frames at a time.

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

~~In this case~~ window

window size = 8 − 1 = 7

One window has many frames and can be sent at once.

Sender and Reciever are synchronized regarding no of frames in a window.

- After Ack, Sender and Reciever window slides.

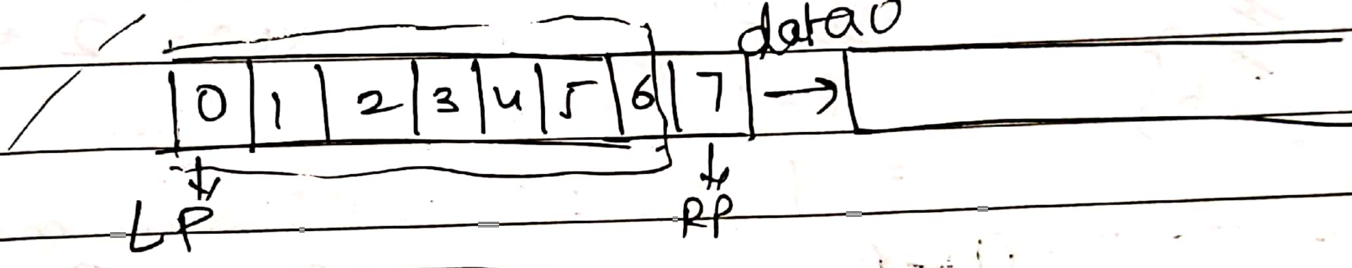
Sender window

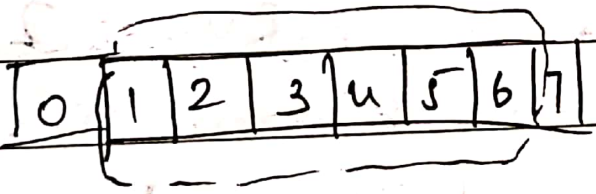
left pointer : moves 1 bit / frame to right when one frame is sent

Right Pointer : moves 1 frame to right when it recieves 1 ACK.

Drawback:
Time is wasted since only one frame is sent over physical medium.

Reciever window
left pointer : moves 1 frame right when one frame recieved
Rig Po : moves 1 frame right when one Ack is sent.





Go-back-n : if one frame is also lost, the whole process starts from the again.

Selective reject

Window size calculation.

1. Go back n : The no of bits reserved for frame identifier ⇒ m

if $m = 3$

then $2^3 = 8$ combinations ie 0–~~8~~ 6

-1

frame is one window

window size $= 8-1 = 7 = \{0 \ldots 6\}$

$SW = 2^m - 1$

$= 2^3 - 1$

$= 8 - 1$

$= 7$

0–6

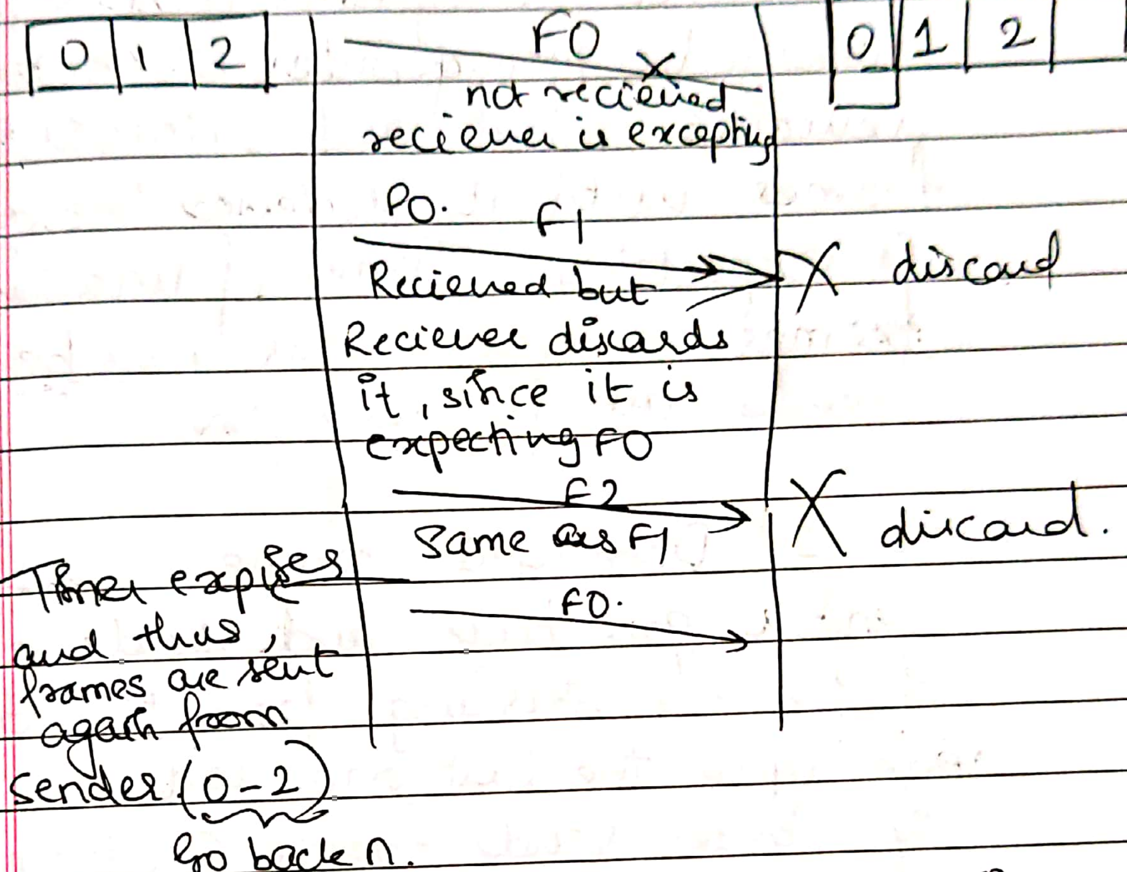
$SW = 2^{m-1}$

$RW = 2^{m-1}$

$RW = 1$

Cases for Go back n

i) Data lost



If acknowledgement is sent ie ACK 1 then go back n will start from first frame.

data + acknowledgement sent together

⇒ Go back n ARQ : In this m frames are sent before waiting for the acknowledgement. Copy of all frames are stored until the acknowledgement arrives. Frame no's have the range of 0 to $2^m - 1$. Sender sliding window slides to include new unsent frames when correct acknowledgement is recieved. Reciever sliding window, Size is always 1, follows a specific order to arrive at a specific frame. If the frame is lost, damaged out of order → Receiver remains silent and discards all the frames until it recieves the one it is expecting. Timer of unacknowledged frames expires. Sender goes back and resends all frames.

Case 1 : Damaged frame.

Sender goes back and sends a set of frames starting from the damaged one upto the last one sent.

eg Sender sends frame 6, timer for frame 3 expires and acknowledgement is lost in transit. Sender goes back to 3rd frame and sends 3, 4, 5, 6 again

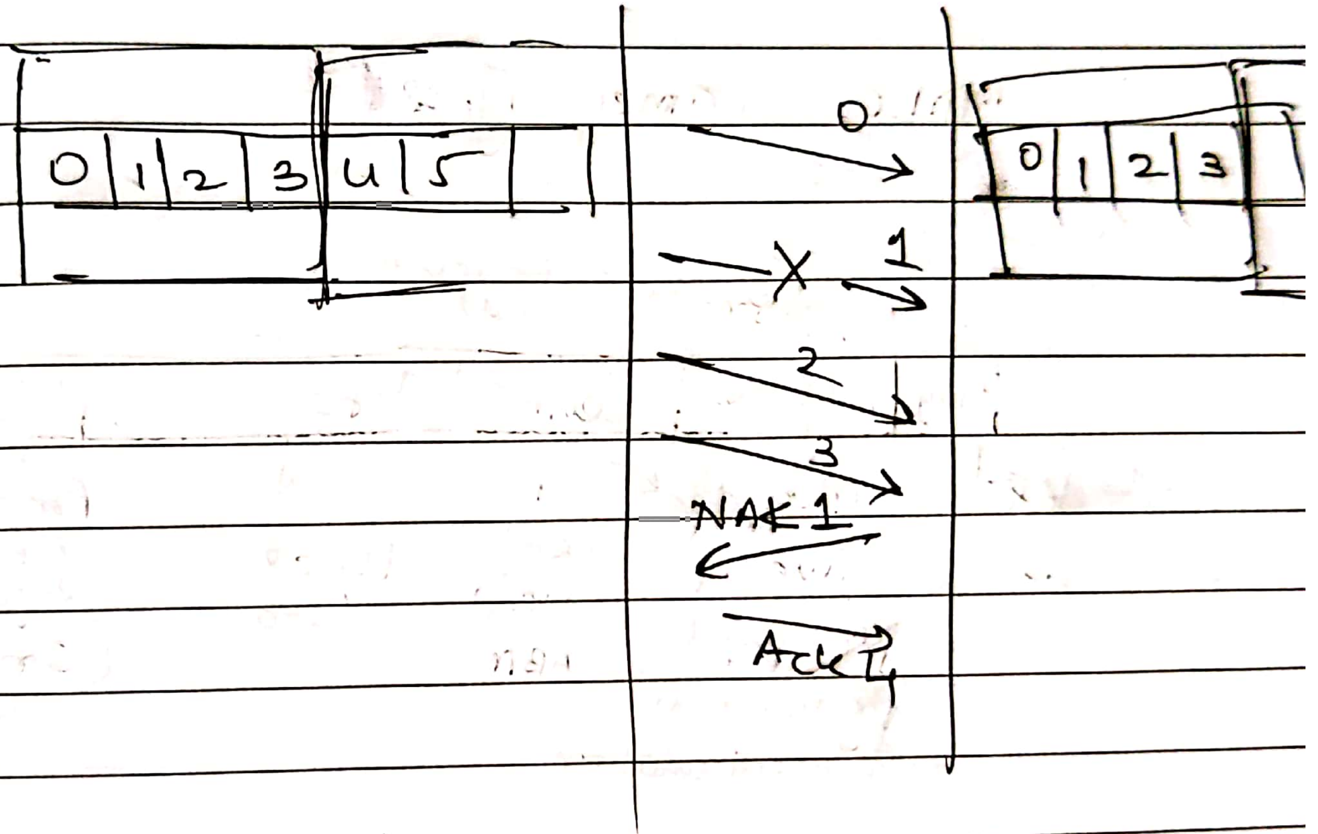
Scanned with CamScanner

* Selective Reject → support Negative Ack.

$S = 2^{m-1}$ $\qquad$ $R = 2^{m-1}$

$m = 3$

$S = 2^{3-1} = 2^2 = 4$ //



Selective reject : Same window size of sender & reciever

Date
Page

HDLC

High level Data Link Control Protocol.

HDLC Configuration : Unbalanced
Symmetrical
Balanced.

HDLC frame Types

1) I-frame

Sender & Reciever add.

Flag	Address	Control	Info	FCS	Flag

- Flag: delimeter 01111110
- Address: Sender & Reciever add.
- Control: gives the value of 'm', used for window size calculation
- ARM / NRM / ABM
- Info: uses data
- FCS: frame Check sequence (Error control)

2) S-frame supervisory frame. [managing frame ie no information]

Flag	Add	Control	FCS	Flag

3) U-frame [management of data frames]
unnumbered

Scanned with CamScanner

classmate
Date
Page

HDLC Control Field.

	Control	

N(S) N(R)

I frame

0				PF			

3 → window size

P | F

Poll bit → final bit

S frame

1	0			PF			

code N(R)

00 : Receive Ready (RR)
01 : Reject (REJ)
10 : Receive not ready (RNR)
11 : Selective reject (SREJ)

U frame

MAC : Medium Access Control.

- Sublayer of DLL.
- MAC : there is a single shared medium or channel for multiple machines

The allocation of these channels to the machines can be :
→ Static or
→ Dynamic → Practically used.

Dynamic Channel allocation has several protocols and thus there are certain assumptions or possibilities. ie take care of different cases.

Assumptions :

1. Station model : N independent station, at any instant one machine is up and working.

2. Single channel Assumption.

3. Collision Ass : when 2 stations at same time transmit data / signals and hence there is a possibility of collision.

Protocols to solve collision
→ Continuous time : N machines can

transmit simultaneously

→ Slotted Time : Time is divided in discrete slots. Each machine will wait for beginning of its own slot, ie it won't wait for channel to be free

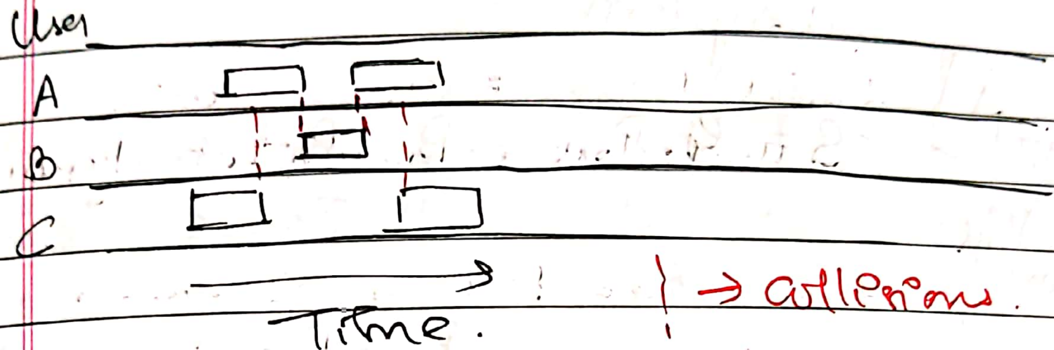
5. Carrier sense : check for the carrier status before transmitting frames.

Non carrier sense : machines dont check the status of the carrier before sending data.
or sense

Multiple Access Protocols.

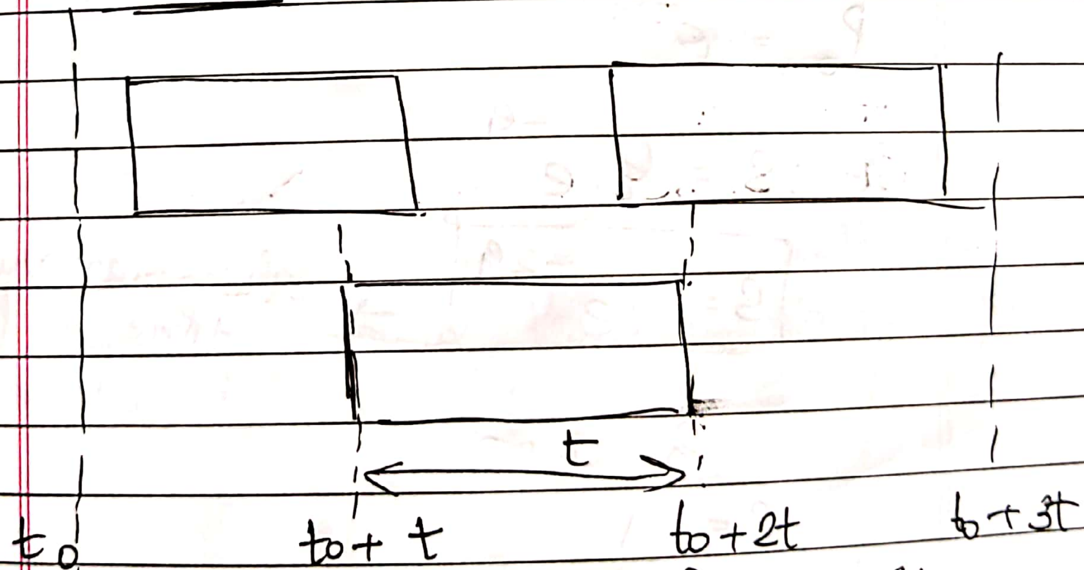
- ALOHA
- Carrier Sense MA protocol.
-

→ ALOHA
- Pure ALOHA
- Slotted ALOHA
- efficiency of the system (Channel utilization)



Since there is a collision there has to be retransmission

Derivation



Max vulnerable time = 2t
(2 frame)

Assuming 'N' frames, "G" is the throughput of the system

N = Total no of frames

G = Throughput (Total load) (old frames + new transmission)

$G > N$

S = Throughput of the system

$\boxed{G > N}$ always

$S = G \cdot P_0$ P_0 = Probability of zero collisions

P_k = Probability of k transmissions.

$$P_k = \frac{G^k e^{-G}}{k!}$$ → Poisson Distribution

$$P_0 = e^{-G}$$

$$\therefore S = G \cdot e^{-G}$$

$$\boxed{S = Ge^{-2G}}$$ → since max vulnerable time is 2 frames i.e. total time.

$G = 0.5$

$$S = \frac{1}{2e}$$

⇒ 18.4% → Total utilization of the channel.

Scanned with CamScanner

For Pure Aloha, 'N' frames per unit time are generated. At higher rate $N > 1$, considering new frames + retransmission of the collision frames.
Total 'G' frames can be generated
So $\boxed{G \geq N}$. At Low load, $\boxed{G = N}$
because of no collisions.
At higher load $\boxed{G > N}$
Throughput 'S' for the system with no collisions is $\boxed{S = G P_0}$

P_0 = no collision
Probability of 'k' frames generated is

$$P_r[k] = \frac{G^k e^{-G}}{k!}$$ by Poisson distribution.

If $k = 0$ ie no collisions
$\boxed{P_0 = e^{-G}}$

Considering Pure ALOHA, vulnerable period is of 2 frame times long
Throughput $\boxed{(S) = G . e^{-2G}}$

At $G = 0.5$

$$S = \frac{1}{2e} \Rightarrow 0.184 \Rightarrow 18.4\%$$

maximum throughput of the system which gives 18.4 % Channel utilization rate.

If I send 100 frames, only 18 frames will be sent

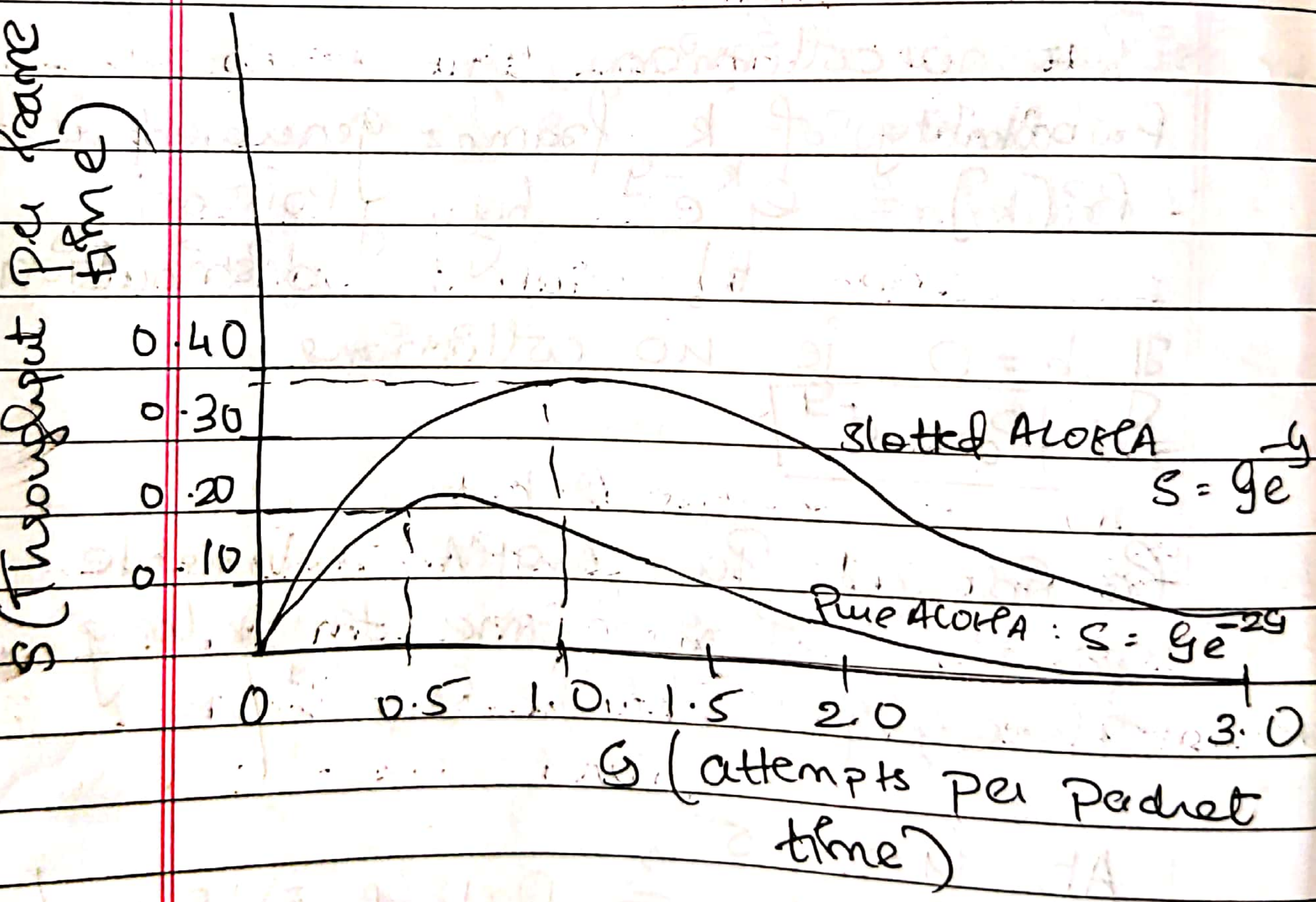
⇒ Slotted ALOHA

- Time divided into discrete intervals.
- Sender must wait for beginning of next time slot

∴ Collission is possible during start of slotted interval only

∴ Reduces vulnerability period by 50% ⇒ 1 frame time

∴ 36% Channel utilization.



Collision Sense Multiple Access Protocols (CSMA)

- Listen for carrier ("Carrier sense protocols")

1 persistent CSMA: Sender continuously keeps checking the channel status.
If busy they will wait, sends as soon as channel is free.
If collision, waits for random time, tries again.
Algo for random time ⇒ Exponential back off algo.
→ 100% checking continuously & starts transmission immediately when channel free.

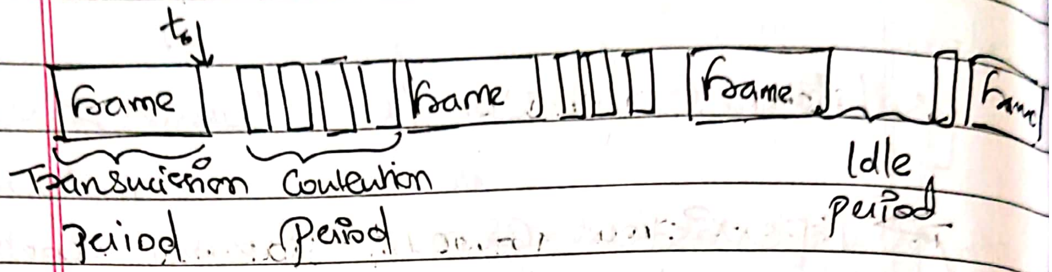
- Non-persistent CSMA.
Sender will not continuously check.
Checks in some random intervals.
Probability of Sender A & B checking at the same time will be less, hence less P( ) of collision.

- P-Persistent: Probability.
If idle, transmit with prop P.
If busy, waits random time.

classmate
Date
Page

✡✡ CSMA with Collision Detection.

Channel time is divided in slots / states.

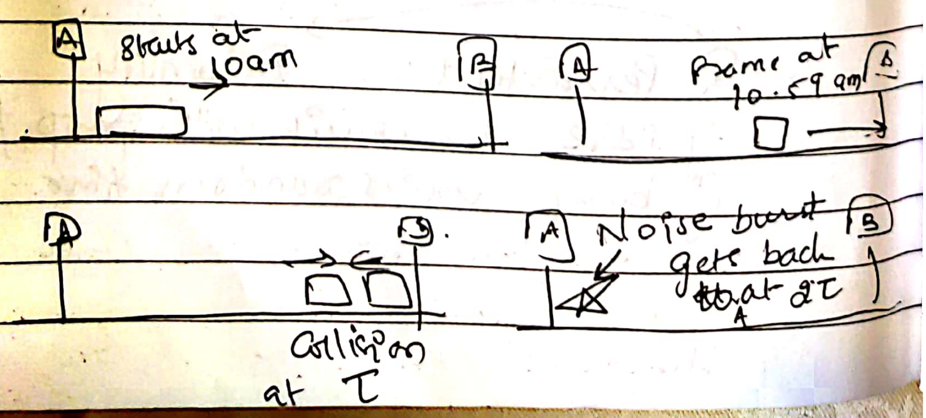


- If collision detected, immediately abort transmission.

Contention: At a time, multiple frames are competing for channel.

Time to detect Collision.

Line is of length l.
Propagation speed is v (m/s)
Then it takes $\tau = l/v$ (s) to get signals from one end to the other
And it takes worst case 2τ seconds to detect a collision



Scanned with CamScanner

2) Send data
3) Release connection

Chp6 : Transport Layer.

* Transport layer takes care of error control for outside the network.

Transport service
Upper Layer Service
⇒ Transport Service Primitives - Port address
⇒ Berkeley Sockets
⇒ Example of Socket Programming
⇒ Internet File Server

NSAP : IP address
↳ Network Service ~~Add~~ Access Point

TSAP : Transport service Access Point.

port no's at server machine for diff services at transport layer eg FTP, SMTP.

Socket : Communication end point
IP add Port add

Primitives ⇒
1. LISTEN
2. CONNECT
3. SEND
4 RECEIVE
5. DISCONNECT

Scanned with CamScanner

T.L ⇒ main is connection establishment.

Elements of Transport Protocols

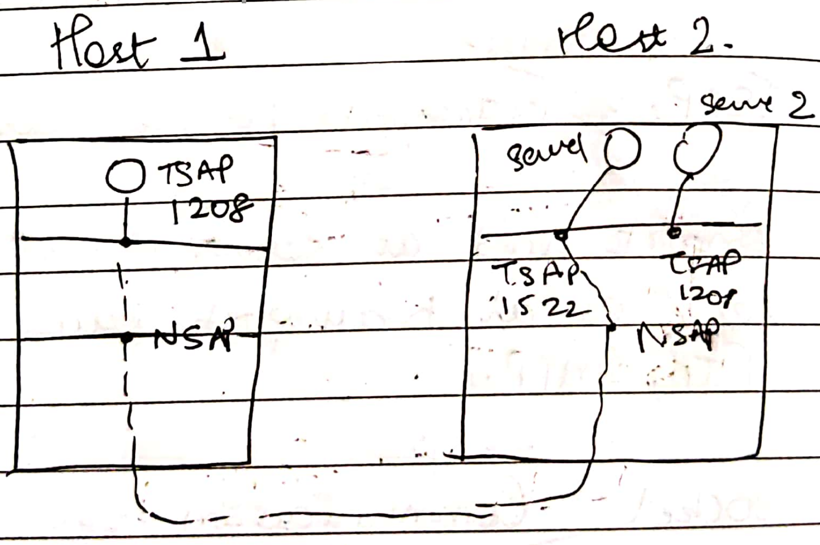
1. Addressing → Port addresses.

In Transport layer, there are 2 different machines not have a physical medium channel b/w them.

Both machines are connected to same router, sub net etc.

So we learn abt TCP connection outside network.

Port addresses are memory locations. Eg 1522, 1836 etc.



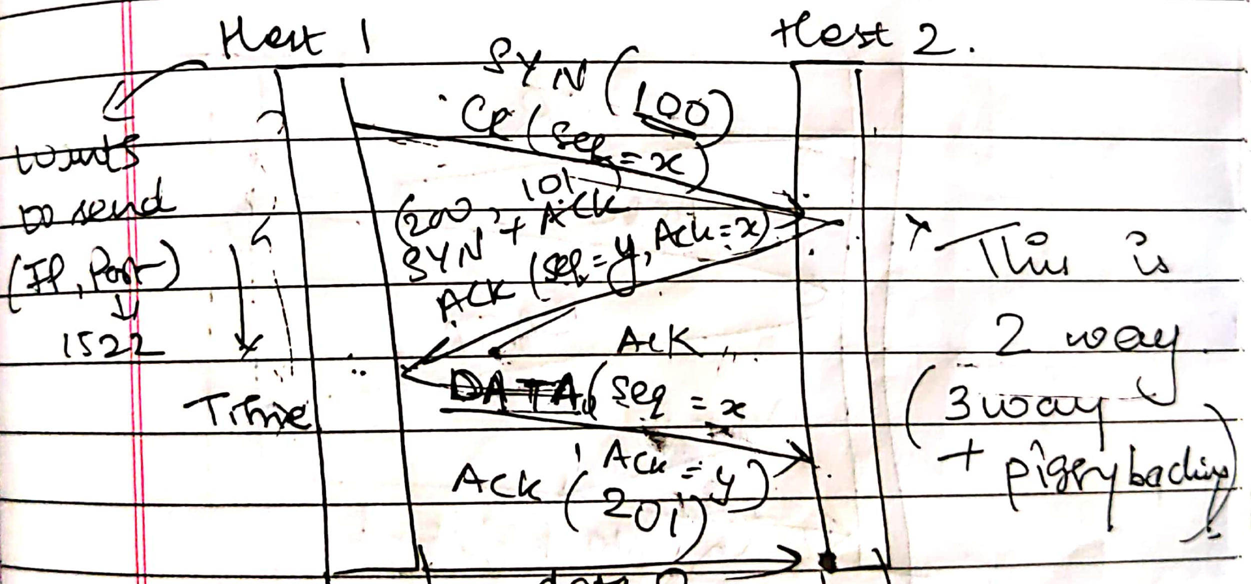
How will Host 1 come to know the port address for the service it wants to request from server.

Scanned with CamScanner

Process Server has a port address 100
All machines in that network will have that address.
So Host 1 only makes a request to port 100.
Process Server has a mapping table to give each service a unique port no. PS will have multiple services.

2. Connection Establishment (3 way Handshake)

3 messages are sent and received just to establish a connection.

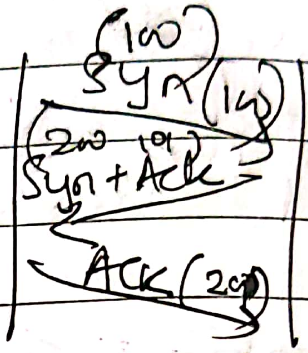


TCP header will have 2 parameters:
Connection Request for seq x is sent to host 2.
Ack is sent by Host 2 for seq x by sending a seq = y, y is random.
The data is sent for seq = x and Ack ~~seq~~ = (seq = y)

There is a problem with this called <u>Blue army problem</u>

Transport Service Primitives (2)

3 way :



CR = SYN

Connection Release