• **Flow Control** : Co-ordinates the amount of data that can be sent before receiving ACKNOWLEDEMENT

Flow control Protocols

Noiseless Channel	Noisy Channel
• No loss of packet, ACK, no duplication, etc.	• Can have loss of packet, ACK and can have duplication, etc.
•1 Simplest protocol	•1 Stop and wait ARQ
•2 Stop and wait protocol	•2 Go back-N ARQ
	•3 Selective Repeat ARQ

(ARQ : Automatic Repeat Request)

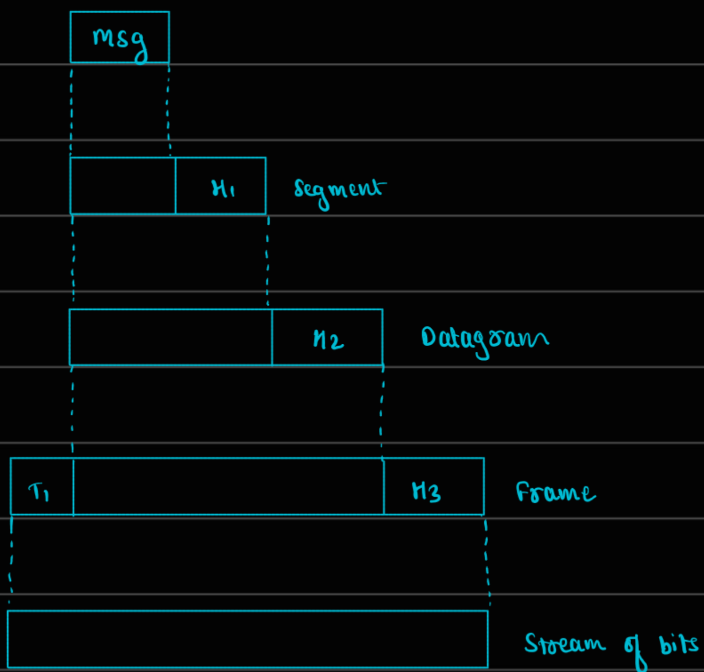
Application layer :

Transport layer :

Network layer :

Data link layer :

Physical layer :



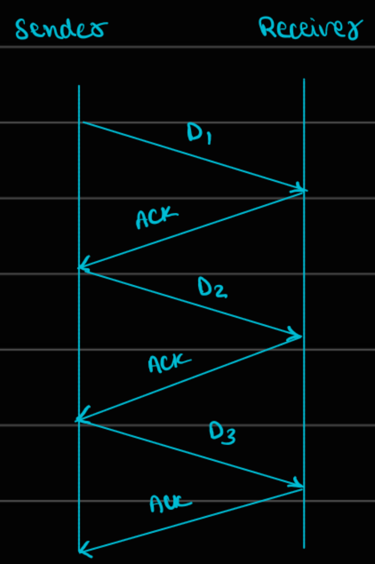
Stop and Wait Protocol

- **Stop and wait Protocol :**

  - Used in connection-oriented communication
  - It is a flow control for TRANSMISSION OF FRAMES over a noiseless channel
  - It provides UNIDIRECTIONAL data transmission with flow control facilities WITHOUT ERROR CONTROL

  - main idea : After transmitting one frame, the sender waits for an acknowledgement (ACK) before transmitting the next frame

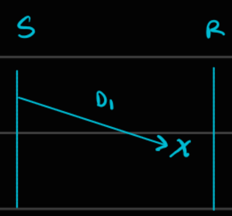
  - Sender Side :
    - 1 Send one data packet at a time
    - 2 Send next packet only after receiving ACK for the previous packet



  - Receiver Side :
    - 1 Receive and consume the data packet
    - 2 After CONSUMING packet, ACK needs to be sent

- **Problems of Stop and wait Protocol :**

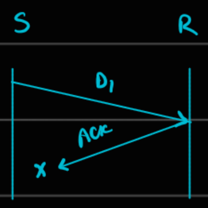
  - 1 lost data packet :



    - Sender waits for ACK for INFINITE amount of time
    - Receiver waits for data for INFINITE amount of time

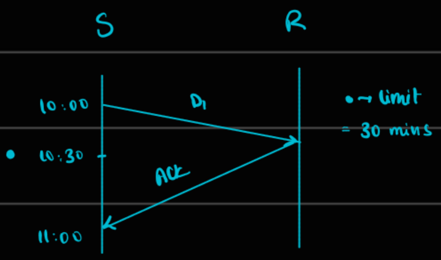
• 2 Lost Ack :

- Sender waits for Ack for INFINITE amount of time
- Receiver does not re-transmit Ack.



• 3 Delay Ack :

- Delay Ack might be considered as an Ack of some other packet. (Sender usually has a limit above which it won't wait for Ack for a given packet



All the above problems is resolved by Stop and wait ARQ

Stop and wait ARQ Protocol

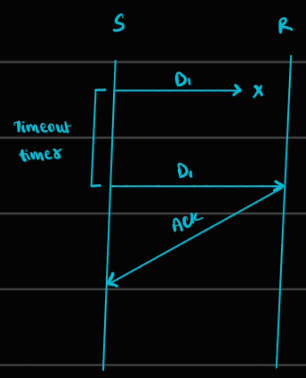
• Stop and wait ARQ Protocol :

- It provides both error control and frame control
- Error Control :
    - Is achieved by keeping a copy of sent frame until it receives an Ack
    - Sender starts a timer when it sends a frame. If Ack is not received within the allocated time period, the sender assumes that the frame was lost or damaged and RESENDS it
    - Receiver sends an Ack to the sender if it receives the frame correctly
    - Ack number always define the NO. OF NEXT EXPECTED FRAME
- Stop and Wait ARQ = Stop and Wait + Timeout Timer + Sequence Number (Data) + Sequence Number (Ack)

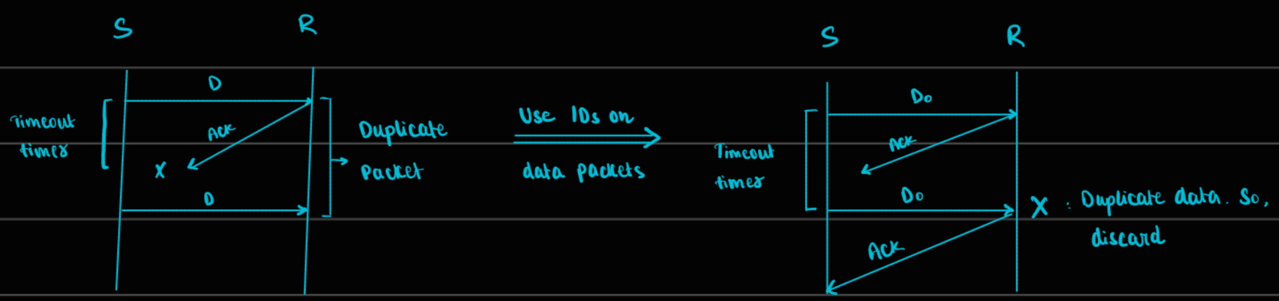
NOTE : Timeout timer is DIFFERENT FOR DIFFERENT FRAMES.

- Solution to above mentioned problems :

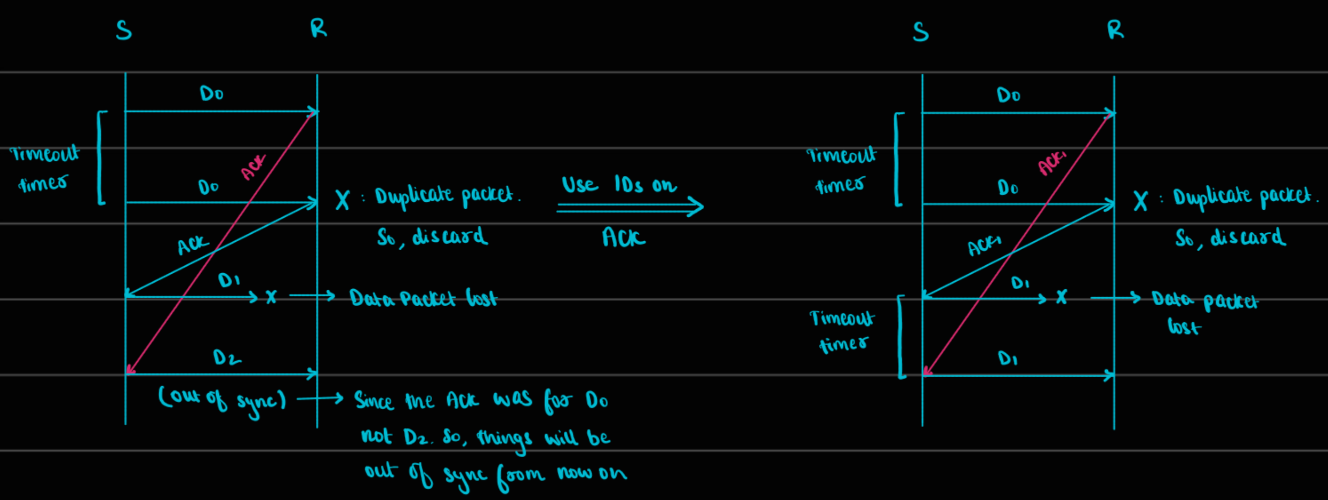
  - 1 Lost data packet : • By using the timeout timer



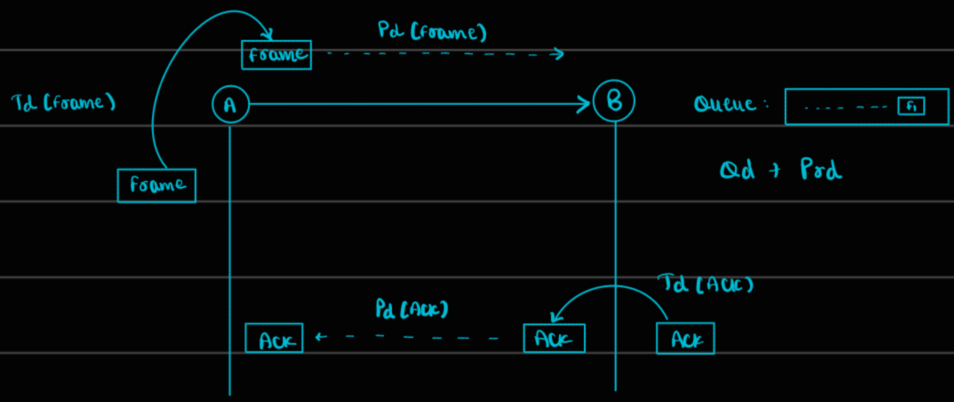
  - 2 Lost ACK : • By using IDs on data packet
    - Sender does not know whether the data packet is lost or the ACK is lost. So, sender RE-SENDS the data packet



- Delay ACK : • By using IDs on ACK



- Efficiency (OR Link Utilization OR Line Utilization OR Sender Utilization) :



Total time OR = Td (Frame) + Pd (Frame) + Qd + Prd + Td (Ack) + Pd (Ack)

Round trip time = Td (Frame) + 2 * Pd + Td (Ack) + Qd + Prd [Pd does not depend on packet size]

(RTT)

Usually , Qd and Prd are negligible and size packet size of Ack << packet size of data packet ,

Qd = 0 , Prd = 0 and Td (Ack) = 0

RTT = Td (Frame) + 2 * Pd + Td (Ack) + Qd + Prd

$$\text{Efficiency } (\eta) = \frac{\text{Useful time}}{\text{Total time}} = \frac{\text{Td (Frame)}}{\text{RTT}}$$

- Throughput (OR Effective Bandwidth OR Bandwidth Utilization OR Maximum Data Rate) :

$$\text{Throughput} = \frac{\text{length of Frame}}{\text{Total time}} = \frac{L}{\text{RTT}}$$

$$= \frac{(L/B) \times B}{\text{RTT}} = \frac{\text{Td (Frame)} \times B}{\text{RTT}} = \eta B$$

Eg. ① Sender wants to send 10 packets. Every 4^{th} packet is lost. Total no. of re-transmissions = ?

(using stop and wait protocol)

1	2	3	4	5	6	7	8	9	10
			x			x			x
			Re-transmit			Re-transmit			Re-transmit

Total no. of transmissions = 10 + 3 = 13

② Sender wants to send 500 packets on a link having an error probability of 0.2. Total no. of transmissions = ? (Using stop and wait protocol)

$$\text{Total transmissions} = 500 + 500(0.2) + 500(0.2)(0.2) + \dots$$
$$= 500/0.8 = 625$$

[need to include the error prob in re-transmissions as well]

$$\boxed{\text{Total transmissions} = \frac{np}{1-p} \quad \left[\begin{array}{l} n = \text{no. of packets and} \\ p = \text{error probability} \end{array}\right]}$$

③ Td (source) = a and Pd = b. Packet size of data = Packet size of frame. After what time the sender can send the second packet ? (Using stop and wait protocol)

$$RTT = Td\,(\text{source}) + 2 * Pd + \cancelto{0}{Qd} + \cancelto{0}{Prd} + Td\,(\text{ACK}) = a + 2b + a = 2a + 2b$$

④ Packet size = 1000 bits ; Data link distance = 1000 km ; Bandwidth = 20 Mbps ; Velocity of propagation = 2×10^8 m/s ; Efficiency of stop and wait protocol = ?

$$Td\,(\text{frame}) = L/B = 1000 / 50 \times 10^6 = 0.05\text{ ms} \quad ; \quad Pd = d/v = 100 / 2 \times 10^5 = 0.5\text{ m/s}$$

$$\text{Efficiency} = \frac{\text{useful time}}{RTT} = \frac{Td\,(\text{frame})}{Td\,(\text{frame}) + 2 * Pd + \cancelto{0}{Qd} + \cancelto{0}{Pgd} + \cancelto{0}{Td\,(\text{Ack})}} = \frac{5}{105} = 0.04761$$
$$= 4.761\ \%$$

⑤ B = 1.5mbps ; Frame size = 8192 bits ; RTT = 45ms ; η = ? (using stop and wait protocol)

$$\eta = T.B = \frac{L.B}{RTT} = \frac{8192 \times 1.5 \times 10^6}{45 \times 10^{-3}} = 0.1213 = 12.13\ \%$$

⑥ L = 1000 bytes ; B = 80kbps ; L_{Ack} = 100 bytes ; Transmitter rate at the receiver = 8 Kbps ; one-way propagation delay = 100ms. Assume stop and wait ARQ and no frame is lost ; sender throughput = ?

$$\text{Throughput (T)} = \frac{L}{RTT} = \frac{1000 \times 8}{Td\ (L_{frame}) + Td\ (Ack) + 2 * Pd}$$

$$Td\ (L_{frame}) = \frac{L_{frame}}{B_{frame}} = \frac{8 \times 1000}{80 \times 1000} = \frac{1}{10}\ s \quad \text{and} \quad Td\ (Ack) = \frac{L_{Ack}}{B_{Ack}} = \frac{8 \times 100}{80 \times 1000} = \frac{1}{10}\ s$$

$$\text{Throughput} = \frac{1000 \times 8}{\frac{1}{10} + \frac{2}{10} + \frac{1}{10}} = 20{,}000\ \text{bits/s} = 2500\ \text{bytes/s}$$

⑦ B = 1Mbps ; Pd = 0.75ms ; Prd = 0.25ms ; L = 1980 bytes ; L_{header} = 20 bytes ; L_{Ack} = 20 bytes. Assume no transmission errors and stop and wait ARQ protocol. Transmission efficiency = ?

$$\eta = \frac{\text{useful time}}{\text{Total time}} = \frac{Td\ (L_{frame})}{Td\ (L_{frame}) + Td\ (Ack) + 2 * Pd + Prd} = \frac{16}{17.91} = 0.8933 = 89.33\ \%$$

$$\boxed{\begin{aligned} Td\ (L_{frame}) &= L_{frame} / B = (1980 + 20) \times 8 / 10^6\ \text{bps} \\ Td\ (Ack) &= L_{Ack} / B = 20 \times 8 / 10^6\ \text{bps} \end{aligned}}$$

⑧ $B = 10^6$; $L = 1000$ bytes $= 8000$ bits ; $Td(Ack) \sim 0$; $Prd \sim 0$; $\eta = 25\%$; $(\text{prop. delay})^{-1} = ?$

$Td(\text{frame}) = L/B = 8\text{ ms}$

$\eta = Td(\text{frame}) \Big/ Td(\text{frame}) + \cancel{Td(Ack)}^{0} + 2 * Pd + \cancel{Qd}^{0} + \cancel{Prd}^{0}$

$\Rightarrow Pd = 12\text{ms}$

⑨ Error prob. of link $= 0.2$; No. of packets $= 100$; Avg. no. of transmission attempts $= ?$ (Assume stop and wait protocol.

$N = n/1-p = 100/0.8 = 125$

⑩ Stop and wait protocol. $B = 32$ kpps ; $Pd = 16$ ms ; $L = 32$ bytes $= 32 \times 8$ bits ; $\eta = ?$

$$\eta = \frac{Td(\text{frame})}{Td(\text{frame}) + 2 * Pd + \cancel{Qd}^{0} + \cancel{Prd}^{0} + \cancel{Td(Ack)}^{0}} = \frac{4B}{4B + 2 * 16 \times 10^{-3}} = \frac{1}{5} = 20\%$$

Go Back N Protocol

- **Go Back N Protocol :**

  - Issue with stop and wait protocol was that it was inefficient since a lot of time was wasted while waiting for Ack
  - It uses sliding window mechanism i.e. send K packets, wait for Ack, send K packets, wait for Ack and so on.
  - In Go-Back-N (GB-N) : • Sender Window Size (Sw) = N
    (N > 1) • Receiver Window Size (SR) = 1

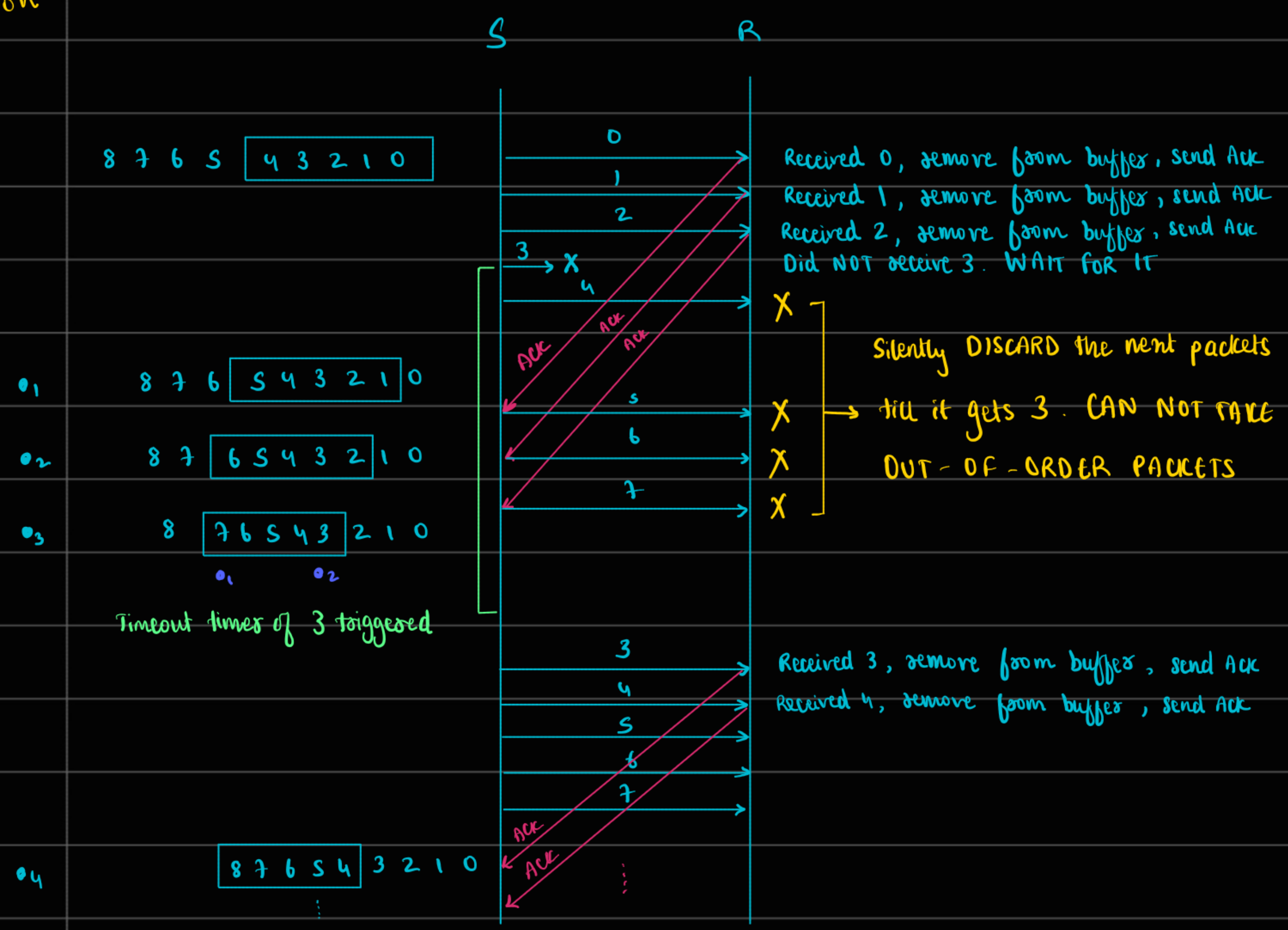
N = 1 will be the same as stop and wait

- Sender : • Send 'N' packets at once one-by-one. Wait for Ack.
  • Once Ack is received send the next 1 packet (sliding window)
- Receiver : • Send Ack one packet is received. DISCARD OUT OF ORDER packets

NOTE : • Timeout timer here is ONLY FOR THE FIRST ELEMENT OF THE WINDOW. Makes sense because if first packet is present in window it means that Ack for it has not been received and if that packet is lost, subsequent packets are anyway discarded. Also that's why other packets in the window don't have a timeout timer since it is unnecessary (even it did have it is pointless as it will be re-transmitted anyway. So, why wait for re-transmission ?)

more on that soon

- Working : Eg. GB-5 : kinda wrong. GBN uses Cumulative Ack and diff. seq. nos.



•1 : Got Ack for 0, Slide window, transmit 5

•2 : Got Ack for 1, Slide window, transmit 6

•3 : Got Ack for 2, Slide window, transmit 7

•4 : Got Ack for 3, Slide window, transmit 8

When packet 3 was missed:

•1 → last transmitted packet

•2 → first transmitted packet

NOTE:
- Silently discard means No Ack is sent to inform that that the packet has been discarded
- GO-BACK-N is ALWAYS FROM LAST TRANSMITTED PACKET

Eg. ① GB-3 ; Every 5^{th} packet is lost ; No. of packets to send = 10 ; No. of transmissions = ?

Without Shortcut:

1 2 3 4 5 6 7 8

1 2 3 4 5 6 7 8

1 2 3 4 5 6 7

1 2 3 4 5 6 7

1 2 3 4 5 6 7 (5 x)

1 2 3 4 5 6 7 5 6 7 8 9 (6 7 5 6 7 numbered 1 2 3 4 5; 7 x)

⋮

Shortcut: START the window from LOST packet and REPEAT the window. Make sure the size of the window is correct

With Shortcut: $\boxed{1\ 2\ 3}$ 4 $\boxed{5\ 6\ 7}$ 5 6 $\boxed{7\ 8\ 9}$ 7 8 $\boxed{9\ 10}$ 9 10

• → lost

Total no. of transmissions = 18

② GB-4 ; Every 6^{th} packet is lost ; No. of packets to send = 10 ; No. of transmissions = ?

$\boxed{1\ 2\ 3\ 4}$ 5 $\boxed{6\ 7\ 8\ 9}$ 6 7 $\boxed{8\ 9\ 10}$ 8 9 10

Total no. of transmissions = 17

③ Same as above but GB-3 and every 4^{th} packet is lost.

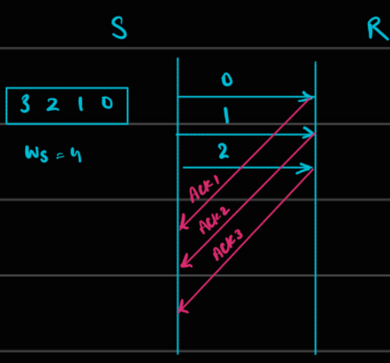
$\boxed{1\ 2\ 3}$ $\boxed{4\ 5\ 6}$ 4 $\boxed{5\ 6\ 7}$ 5 $\boxed{6\ 7\ 8}$ 6 $\boxed{7\ 8\ 9}$ 7 $\boxed{8\ 9\ 10}$ 8 $\boxed{9\ 10}$ 9 10

Total no. of transmissions = 27

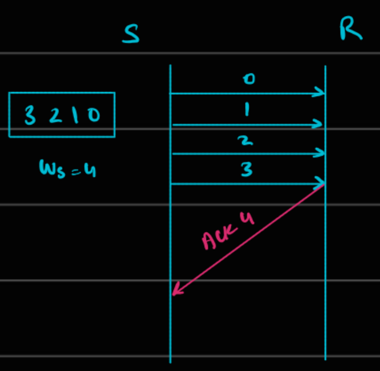
- <u>Short note on Ack and Ack Timers</u> :

  - There are 2 types of Ack : Independent and Cumulative.
  - Stop and wait uses independent and GB-N uses cumulative
  - Example :

Independent Ack



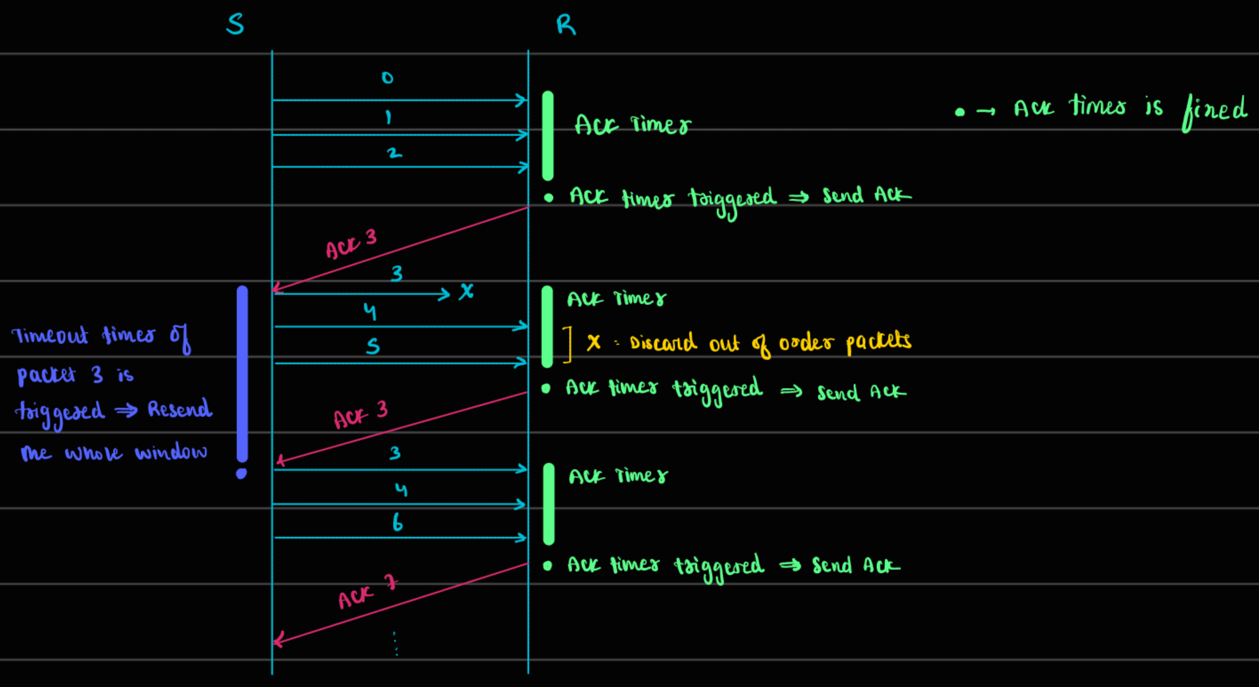
Cumulative Ack



- GB-N uses cumulative Ack and Ack no. defines the no. of next expected frame

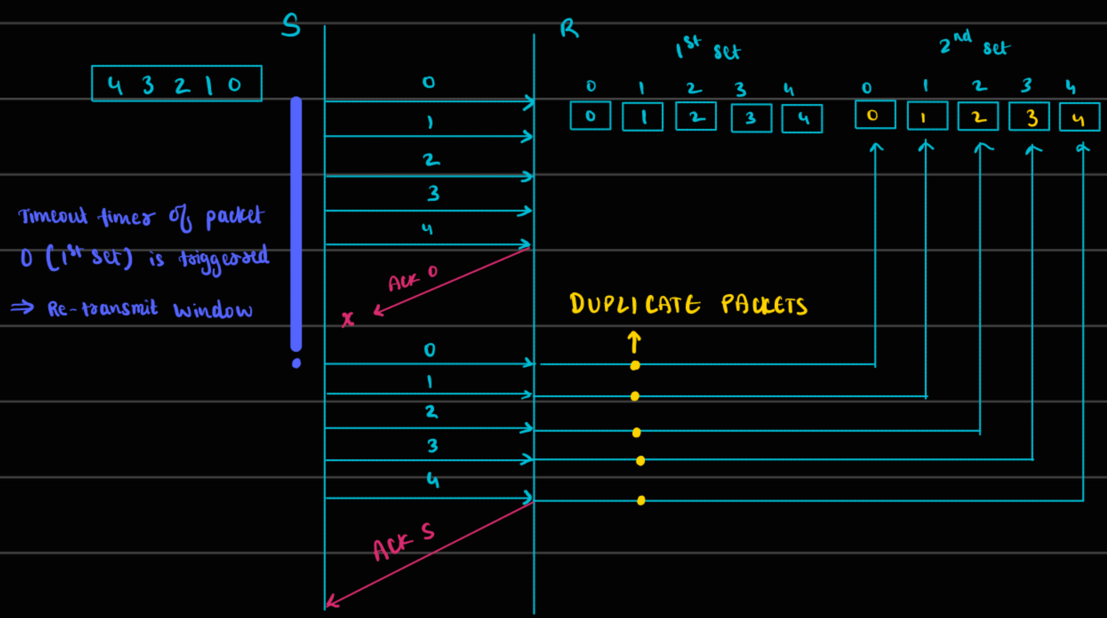
- Just like timeout timers, GB-N also uses a timer on the receiver side called the Ack timer. Receiver sends the Ack for the latest packet it received once the Ack timer is over. Note that the receiver MAY NOT get all the frames in the sender's window. See below for a better understanding :

(Eg. GB-3 ; 10 packets)



- **Short note on sequence numbers:**

- Instead of having increasing seq. nos. (0, 1, 2, ...), we can have cyclic seq. nos. (0, 1, 2, 0, 1, 2, ...)
- GB-N uses cyclic seq. nos. The window size and the seq. nos. are related. More on that soon.
- Idea : (eg. GB-5 ; seq. no = 5 (0 to 4))



But this is WRONG since these packets belong to 1^{st} set not the 2^{nd} one. The receiver can't tell which set packet 0 belongs to

ISSUE : DUPLICATE PACKETS RECEIVED AND ACCEPTED BY RECEIVER

- Solution to duplication problem:
  - Increase seq. no. OR
  - Decrease window size

we can also this formula to solve the duplication problem:

$$\boxed{W_S + W_R \leq ASN}$$

where ASN is the available seq. no., W_S and W_R are the window size of sender and receiver respectively.

In GB-N, $W_R = 1$ (always). So, for GB-N, $W_S + 1 \leq ASN$

Eg. (for GB-N) ① seq. no = 6 ⇒ $W_R = 5$; $W_S = 1$

② Seq. no = 18 ⇒ $W_R = 17$; $W_S = 1$

③ Seq. no. = 3 bits : Total seq. nos. = $2^3 = 8$ ⇒ $W_R = 7$; $W_S = 1$

④ Seq. no. = 4 bits : Total seq. nos. = $2^4 = 16$ ⇒ $W_R = 15$; $W_S = 1$

- Efficiency and Throughput: (for GB-N)

$$\eta = \frac{\text{Useful time}}{\text{Total time}} = \frac{N * Td\,(\text{frame})}{Td\,(\text{frame}) + Td\,(\text{ACK}) + 2 * Pd + Prd + Qd}$$

$$\text{Throughput} = \eta B = \frac{N * \text{length of frame}}{\text{Total time}}$$

Eg. ① Max. window size = 512. Range of seq. nos. = ? (Assume GB-N)

Seq. no. starts from 0 and wrt min. seq. no. required = $W_R + W_S$ = 513

So, range is 0 - 513

(2) B = 20 Kbps ; Pd = 400ms ; GB-10 ; L = 100 bytes ; Throughput = ?

$$Td\,(frame) = \frac{L}{B} = \frac{100 \times 8}{20 \times 10^3} = 40ms.$$

$$\eta = \frac{N * Td\,(frame)}{Td\,(frame) + \underbrace{Td\,(ack)}_{0} + \underbrace{P/d}_{0} + \underbrace{Qd}_{0} + 2 * Pd} = 0.4761$$

Throughput = 0.4761 x 20 = 9.52 Kbps ~ 10 Kbps

☆ (3) L = 1000 bits ; Dist. b/w 2 hosts = 5 km ; B = 1 mbps ; signal speed = 2ms / km ; GB-7 ; η = ?

Signal speed = 2ms per km ⇒ Propagation delay for 1km = 2ms

Total prop. delay (Pd) = Signal speed x Distance = 10ms

Td (frame) = L / B = 1ms

$$\eta = \frac{N * Td\,(frame)}{Td\,(frame) + \underbrace{Td\,(ack)}_{0} + \underbrace{P/d}_{0} + \underbrace{Qd}_{0} + 2 * Pd} = 1/3 = 0.3333 = 33.33\,\%$$

(4) L = 1000 bytes ; Td (one packet) = 1ms ; Dist. b/w 2 hosts = 10km ; Signal speed = 5ms / km ;

length of frame seq. no. = 6 bits ; Throughput in mbps = ? (Assume GB-N protocol)

$Td\,(frame) = L/B \Rightarrow B = 1000 \times 8 / 10^{-3} = 8\,mbps$

$$\eta = \frac{N * Td\,(frame)}{Td\,(frame) + \underbrace{Td\,(ack)}_{0} + \underbrace{P/d}_{0} + \underbrace{Qd}_{0} + 2 * Pd} = \frac{63}{101}$$

Seq. no = 6 bits ⇒ $W_s = 2^6 - 1$ and W_R = 1

⇒ $W_s = 2^6 - 1 = 63 = N$

Pd = 5ms / km x 10 km = 50ms

Throughput = ηB = 8 x 63 / 101 = 4.99 Mbps ~ 5 mbps

⑤ 2 hosts A and B ; Duplex link ; $L_{A \to B}$ = 1000 bytes ; Td (one packet) = 50 us ; Td (ACK) ~ 0 ;

Pd = 200 us ; Assume sliding window protocol where Ws = WR = 5 ; max. throughput = ?

$$\text{Throughput} = \frac{N * \text{length of frame}}{\text{Total time}} = \frac{5000\,B}{450\,us} = 11.11 \times 10^6\ B/s \quad \text{(Bytes per second)}$$

⑥ GB-N ARQ ; Frame size = 4000 bits ; B = 1mbps ; Pd = 18 ms ; min. value of Ws and min. no. of bits required for seq. no. field for max. utilization = ?

NOTE : If efficiency is not given, assume it to be 1

$$Td\,(\text{frame}) = L/B = 4000/10^6 = 4\,ms$$

$$\eta = \frac{N * Td\,(\text{frame})}{Td\,(\text{frame}) + \underbrace{Td\,(\text{ACK})}_{0} + \underbrace{Ppd}_{0} + \underbrace{Qd}_{0} + 2*Pd} \quad \text{max. utilization} \Rightarrow \eta = 1$$

$$1 = \frac{N * 4}{4 + 2*18} \Rightarrow N = 10$$

min. seq. no. = Ws + WR = 10 + 1 = 11 ⇒ Min. no. of bits required = $\lceil \log_2 11 \rceil = 4$

⑦ B = 512×10^3 bps ; Pd = 150 ms ; L_{frame} = 1KB ; L_{ACK} = 64 byte ; Prd = 5ms ; min. window size = ?

Assume GB-N protocol.

$Td\,(\text{frame}) = L_{frame}/B = 16\,ms$; $T_{ACK} = L_{ACK}/B = 1\,ms$

$$\eta = \frac{N * Td\,(\text{frame})}{Td\,(\text{frame}) + Td\,(\text{ACK}) + Prd + \underbrace{Qd}_{0} + 2*Pd} = 1 \quad (\eta \text{ is not given. So, } = 1)$$

$$\Rightarrow N = \lceil 322/16 \rceil = \lceil 20.125 \rceil = 21$$ (Always take ceil ($\lceil x \rceil$) of it)

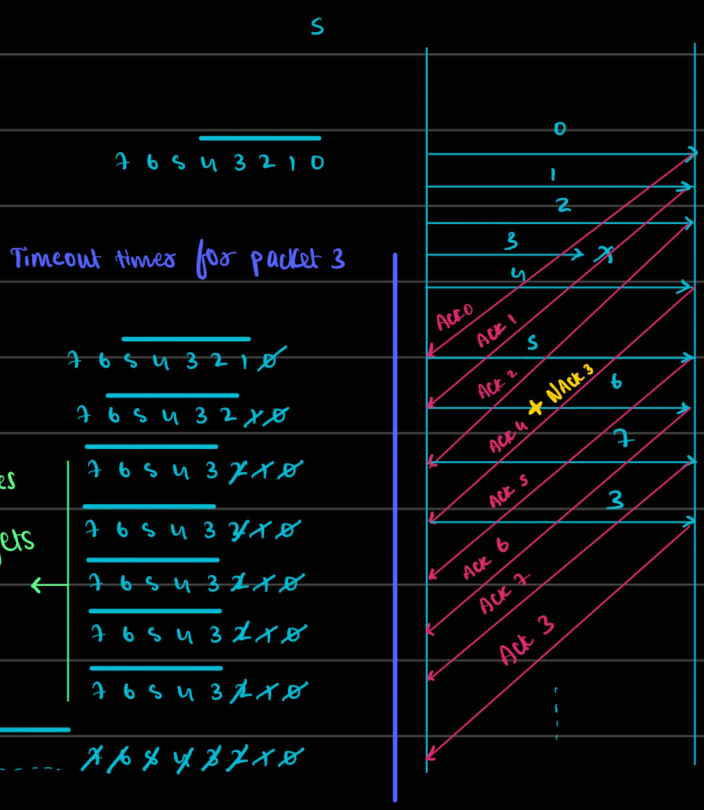
Selective Repeat / Selective Repeat ARQ Protocol

- ## Selective Repeat / Selective Repeat ARQ Protocol (SR)

  - In SR protocol, $W_S = W_R$
  - It uses INDIVIDUAL ACKNOWLEDGEMENT and ACK no. defines the no. of error free packets received.
    Meaning, for packet 0, ACK will be 0 NOT 1; for packet 1, ACK will be 1 NOT 2 and so on
    i.e. Packet no. is the same as its ACK no.
  - It CAN receive OUT-OF-ORDER packets but they are delivered to upper layer in order
  - SR protocol requires sorting and searching. Searching is done by sender and sorting is done by receiver
  - Timer is maintained for each and every frame in the window at the sender side

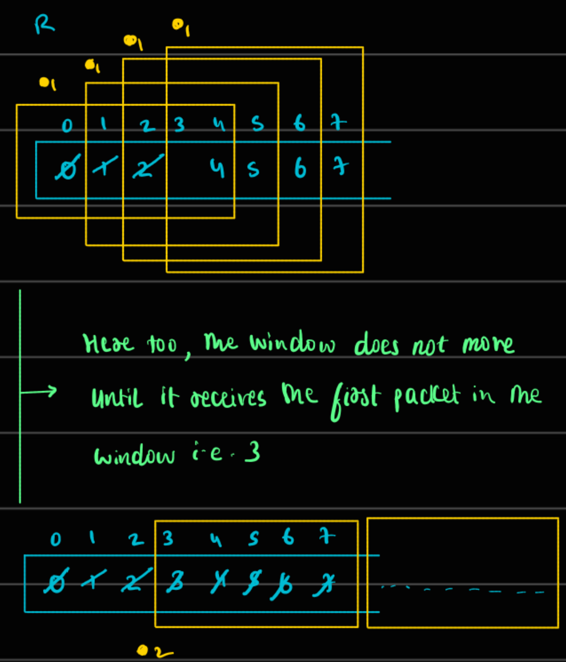
  - <u>Idea</u> : Re-transmit only the lost packet not the whole window

  - Working : $(W_S = W_R = S)$



Note that window does not move until it gets ACK for the FIRST packet in the window i.e. 3

Here too, the window does not move until it receives the first packet in the window i.e. 3



•1 → Sends to upper layer as is
•2 → Sends to upper layer after sorting

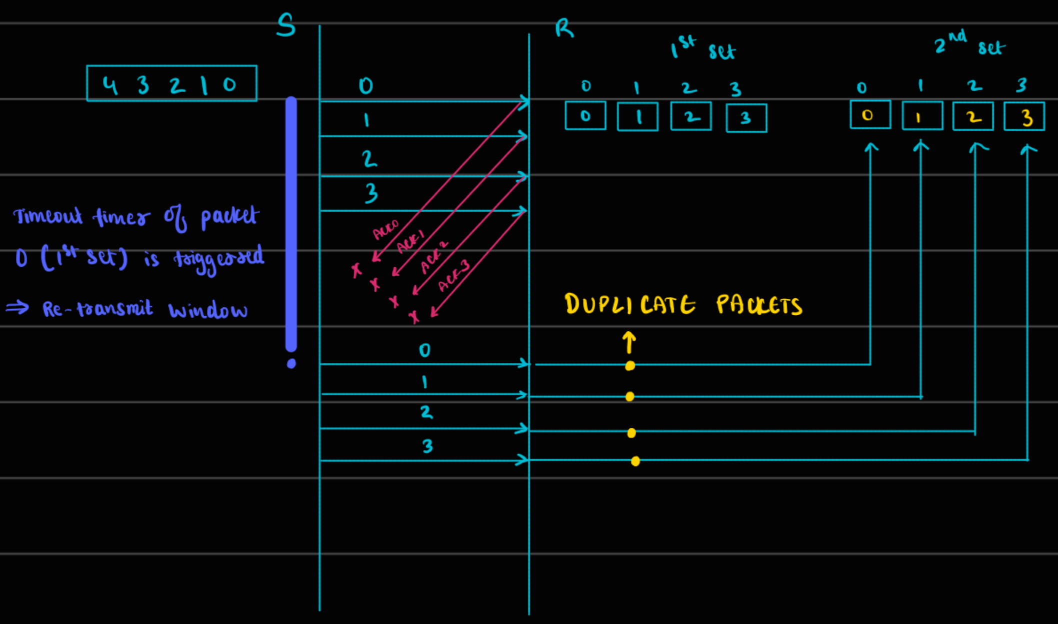
For 1st out of order delivery or corrupted packet, NACK (negative acknowledgement) is sent by receiver to sender

- When the sender receives NACK 3, it SEARCHES the window for packet 3 and IMMEDIATELY packet 3 is re-transmitted even though its TIMEOUT TIMER has not expired

NOTE : • Any case of say loss of NACK, etc. is handled by the TIMEOUT TIMER

- Short note on sequence numbers : (Read similar case to GB-N. This is a literal copy paste)

  - Instead of having increasing seq. nos. (0, 1, 2, ...), we can have cyclic seq. nos. (0, 1, 2, 0, 1, 2, ...)
  - GB-N uses cyclic seq. nos. The window size and the seq. nos. are related. More on that soon.
  - Idea : (eg. GB-S ; seq. no = 5 (0 to 4))



But this is WRONG since these packets belong to 1st set not the 2nd one. The receiver can't tell which set packet 0 belongs to

ISSUE : DUPLICATE PACKETS RECEIVED AND ACCEPTED BY RECEIVER

- Solution to duplication problem: • Increase seq. no. OR → GB-N uses this
• Decrease window size → SR uses this

WKT, $W_S + W_R \leq ASN$ and $W_S = W_R$. So, if $ASN = N$,

$W_S = W_R = \frac{N}{2}$ and if ASN has 'k' bits, $W_S = W_R = 2^{k-1}$

Eg. (for SR) ① Seq. no = 8 ⇒ $W_R = 4$; $W_S = 4$

② Seq. no = 16 ⇒ $W_R = 8$; $W_S = 8$

③ Seq. no. = 3 bits : Total seq. nos. = $2^3 = 8$ ⇒ $W_R = 4$; $W_S = 4$

④ Seq. no. = 4 bits : Total seq. nos. = $2^4 = 16$ ⇒ $W_R = 8$; $W_S = 8$

• Efficiency and Throughput: (for SR same as GB-N. In both cases, $N = W_S$)

$$\eta = \frac{\text{Useful time}}{\text{Total time}} = \frac{N * Td(\text{frame})}{Td(\text{frame}) + Td(\text{ACK}) + 2*Pd + Prd + Qd}$$

$$\text{Throughput} = \eta B = \frac{N * \text{length of frame}}{\text{Total time}}$$

Eg. ① $W_S = 75$, min. seq. nos. required in GB-N and SR = ?

GBN : $W_S + 1 = N$ ⇒ $N = 76$ ⇒ 0 to 75

SR : $W_S + W_R = N$ ⇒ $N = 150$ ⇒ 0 to 149

② N = max. seq. no ; W_S in GBN and SR = ?

GBN : N : 0, 1, 2, --, N $\Rightarrow$ Ws = N

SR : N : 0, 1, 2, ---, N $\Rightarrow$ Ws = (N+1)/2

[N is max. seq. no. NOT total no. of seq. no.]

③ B = 1mbps ; Pd = 1.25 s ; L = 1 KB ; min. no. of bits we need for seq. no. for GB-N and SR = ?

$$Td(frame) = L/B = 8 \times 1024 / 10^6 = 0.008192 \text{ sec}$$

$$\eta = \frac{Ws * Td(frame)}{Td(frame) + \underbrace{Td(Ack)}_{0} + 2*Pd + \underbrace{Qd}_{0} + \underbrace{Pod}_{0}}$$

η → 1 → since not given

So, Ws = ⌈306.17⌉ = 307

So, min. no. of bits required for

- GB-N : $307 = 2^k \Rightarrow k = 9$
- SR : $307 + 307 = 2^k \Rightarrow k = 10$