Sh3

UTOPIA

① No need to initialize the headers?
* Bec. we don't need to add a Seq no ; as no frames are lost or damaged

② No need to set a timer at the Sender or the reciever?
* No frames are lost ; since the channel is error-free
* the Sender will always Send, and the receiver will always receive since the network has ∞ no of Packets to Send [UTOPIA Assump.]
* No timer needed at the Sender as it receives nothing!

Stop & wait

③ No need for a Seq no? (Buffer Size = 1)
* Bec. only one frame is Sent at a time, no Confusion can happen!

④ Wh. we Send a dummy empty ACK? ie No ACK no
* Bec. we ack one frame at a time, no Confusion can happen!

⑤⑥ No need to Set a timer at the Sender Nor the receiver?
* Bec the Channel is error free; No frames or ACKs are lost or damaged

Simplex Protocol for Noisy Channel

⑦ why we need to set a Seq no to the sent frame? why one bit only

* Since the channel is noisy, while sending a frame, a late frame could reach the receiver [out of order]. we need the Seq no to ensure the in-order delivery of the frames.
* we need 1 bit only, bec the receiver needs to check in a window of two frames [i.e one bit] for the correct frame

⑧ why we need to set an ack no to the sent ACKs?

* Same as ⑦, but for ACKs

⑨ why we need to set a timer at the sender?

* Since the channel is noisy, the frame or its ACK can be lost, the timer [i.e timeout mechanism] is set to prevent deadlocks → [Sender: waiting for an ACK, Rec: Not getting the frame it should ACK]

⑩ why we need to set a timer at the rec?

* Same as ⑨ but, for ACKs. [Sender: waiting for an ACK, Rec: thought it has sent one] was lost

In general, we need only one timer [at the sender or the rec]

$$\text{Transfer time } (T_t) = \frac{\text{Frame length } (L)}{\text{Bandwidth } (B)}, \quad \text{Throughput } (T_h) = \frac{L}{T_t + 2T_p}$$

$$\eta = \frac{\text{useful time } (T_t)}{\text{Total time } (T_t + 2T_p)}$$

$$\downarrow \quad \eta = \frac{B * T_t}{T_t + 2T_p} = \eta * B$$

LinkCapacity

(11) $2TP = 250 * 10^{-6}$ sec, $BW = 1500 * 10^3$ B/sec, $L = 512$ Bytes

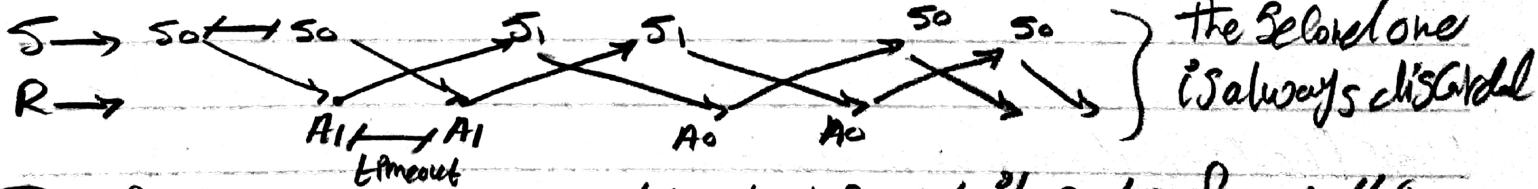
$$Tt = \frac{L}{B} = \frac{512}{1500 * 10^3} = 0.34133 * 10^{-3} \text{ sec}$$

$$\eta = \frac{Tt}{Tt + 2TP} = \frac{\quad}{\quad + 250 * 10^{-6}} = 0.58$$

$$Th = \eta * B = 0.58 * 1500 * 10^3 \approx 870 \text{ KBPS}$$

(12) In Simplex Noisy Channel, Sender's timeout < round trip delay (2TP), discuss what will happen, even it only happened for one packet!

* Frames & ACKs [at least a pair of them] will be Send twice, they will live inefficiently even after!



(13) why does the sender Send the last Packet it Sent When getting duplicate ACKs? [what Assumps are made?]

* It assumes
  - Frame was lost and the rec timed out [so, sender sends S0, again] — rec Send A1 again
  - ACK was delayed [the second one, IS just an del ack]
  - ACK was modified by the Channel [rec A1 instead of A0]
  - ACK was duplicated by the Channel [2A1 instead of 1A1]

(14) Packet Duplication at the reverse Channel in UTOPIA, How to Solve?

* No changes are needed [No Problem], as there is No data Sent through the reverse Channel [from rec to Sender]