CN HW4 資工三 b03902125 林映廷

1、

a. Note, wrap around if overflow

```
  11100000
+ 01011100
 100111100
  00111101

  00111101
  01010101
  10000010
```

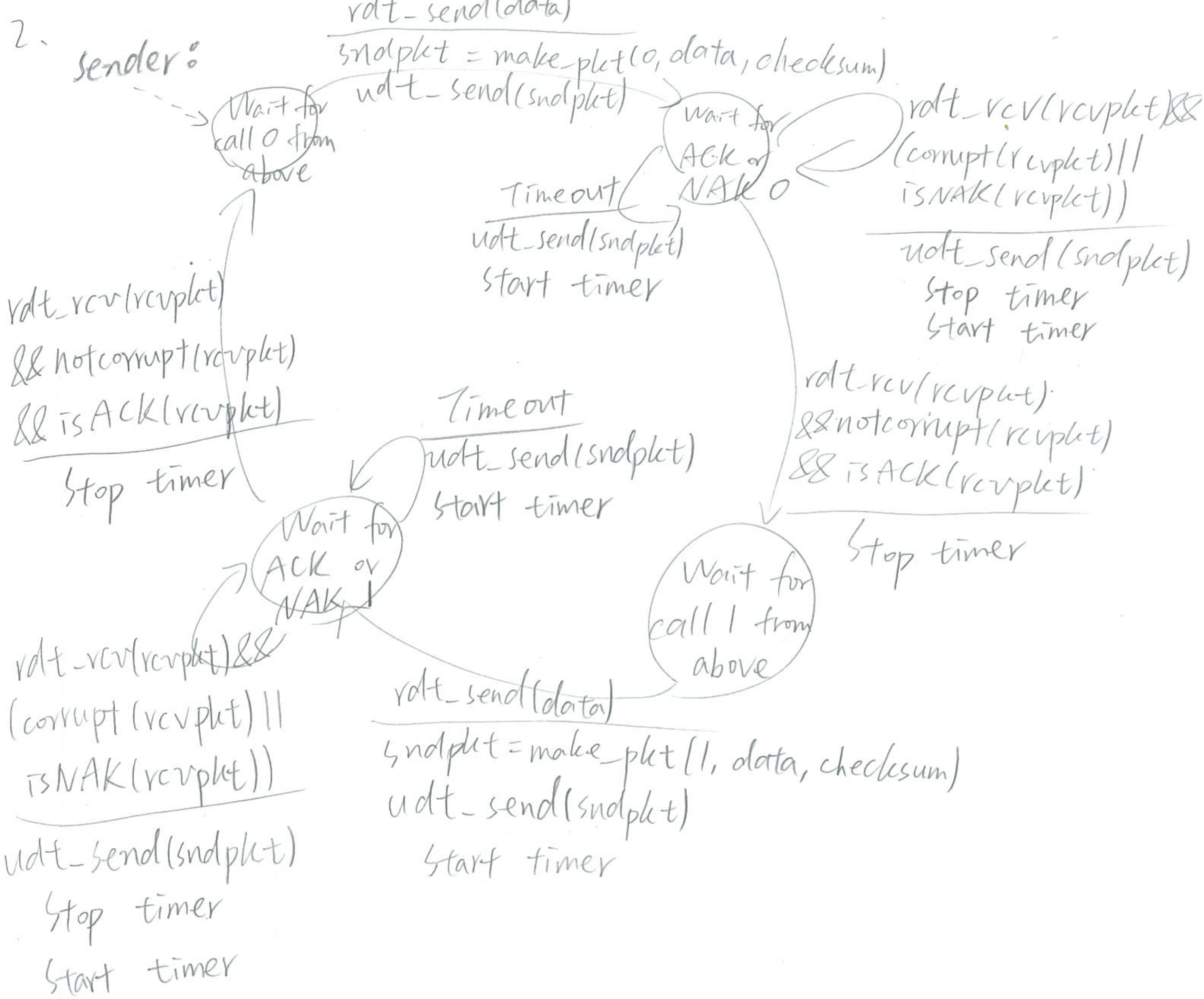
```
 10111001
 01110100
100101101
```

A: 10000010

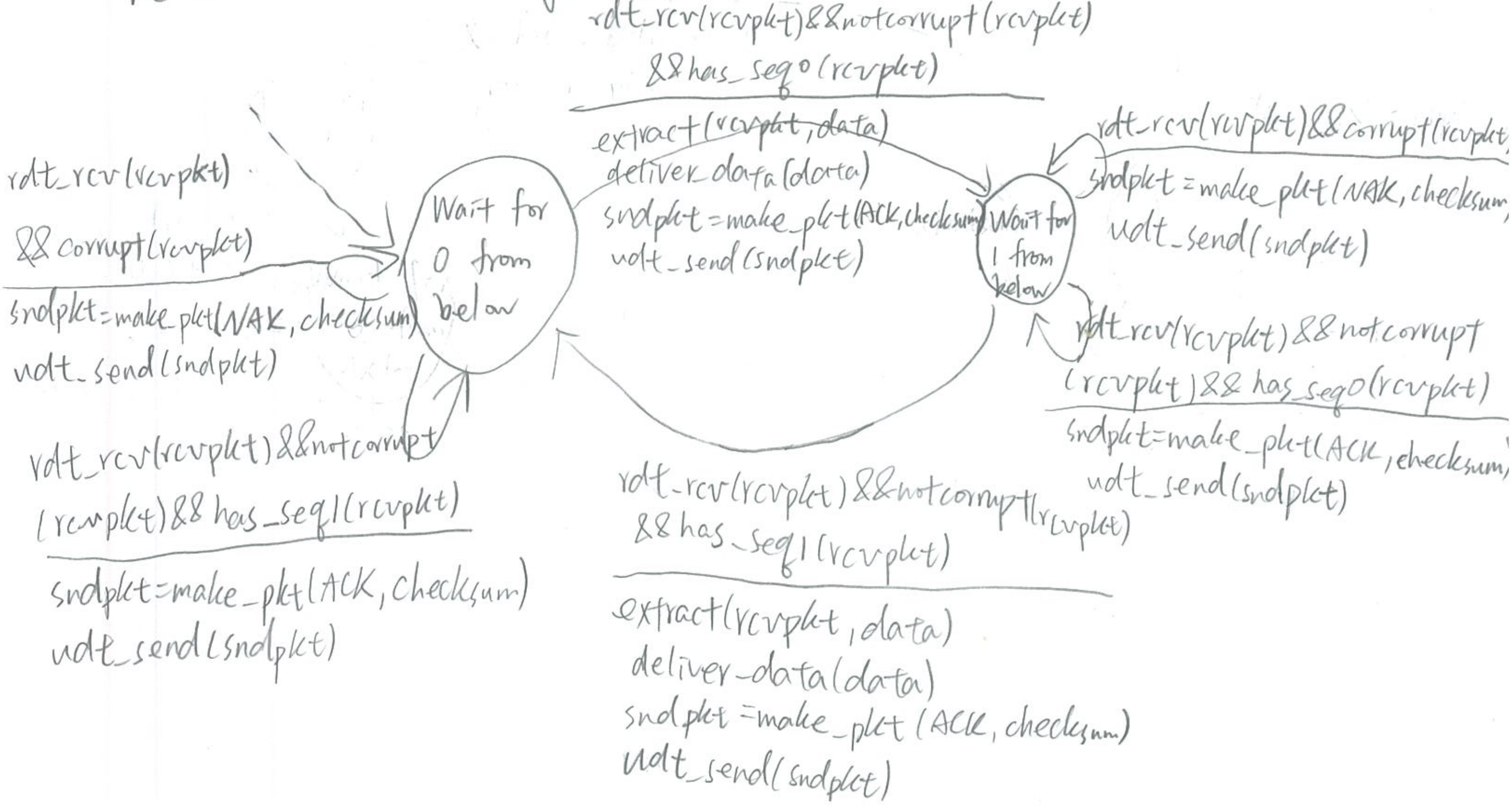
1.b. To detect errors, the receiver adds the four words (the three original words and the checksum). If the sum contains a zero, the receiver knows there has been an error. All one-bit errors will be detected, but two-bit errors can be undetected (e.g., if the last digit of the first word is converted to a 0 and the last digit of the second word is converted to a 1).

2.

sender:



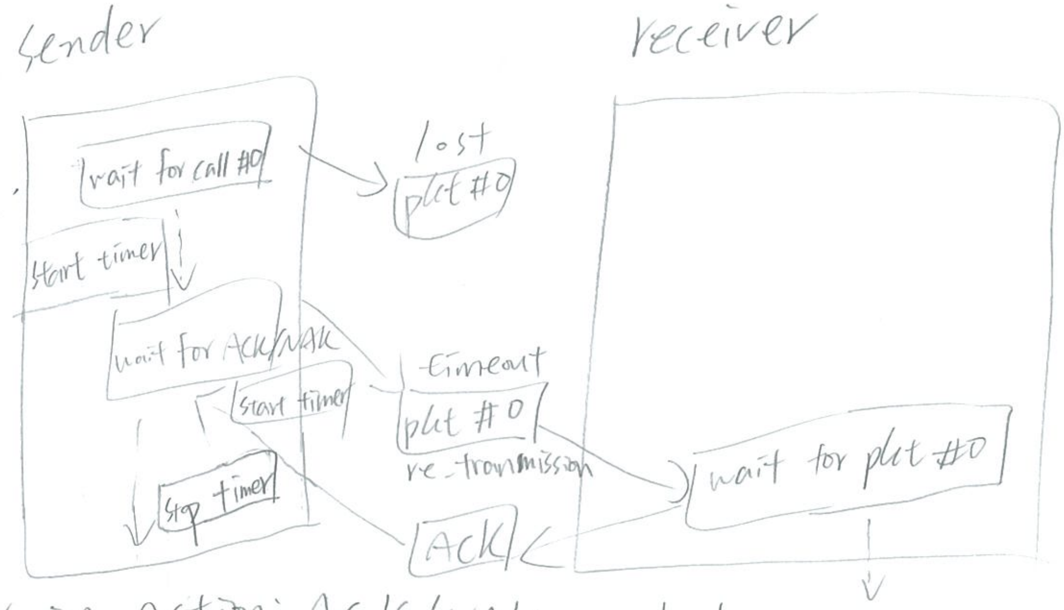
receiver: No changes are needed



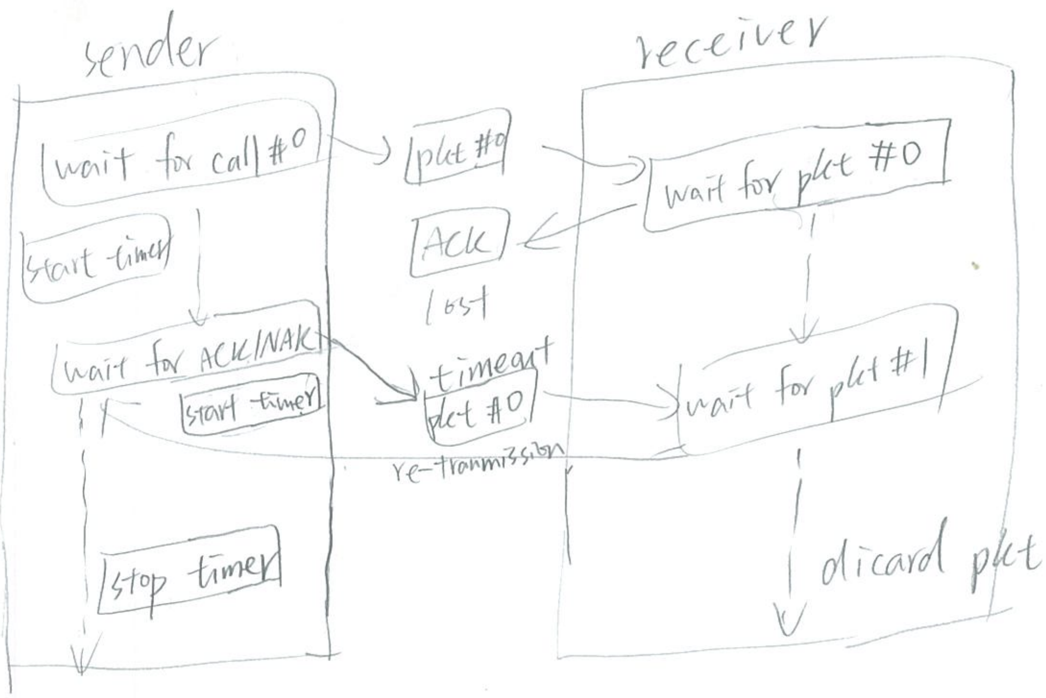
* Suppose the timeout is caused by a lost data packet, i.e. a packet on the sender-to-receiver channel. The receiver never received the previous transmission and, if the timeout retransmission is received, it look exactly the same as if the original transmission is being received

* Suppose an ACK is lost. The sender will eventually retransmit the packet on a timeout. But a retransmission is exactly the same action that is take if an ACK is garbled. Thus, the sender's reaction is the same with a loss, as with a garbled ACK.

* timer in action: data packet is lost.



* timer in action: ACK/NAK is lost



3. (a) Here we have a window size of $N=4$. Suppose the receiver has received packet $k-1$, and has ACKed that and all other preceding packets. If all of these ACK's have been received by sender, then sender's window is $[k, k+N-1]$. Suppose next that none of the ACKs have been received at the sender. In this second case, the sender's window contains $k-1$ and the N packets up to and including $k-1$. The sender's window is thus $[k-N, k-1]$. By these arguments, the senders window is of size 4 and begins somewhere in the range $[k-N, k]$.

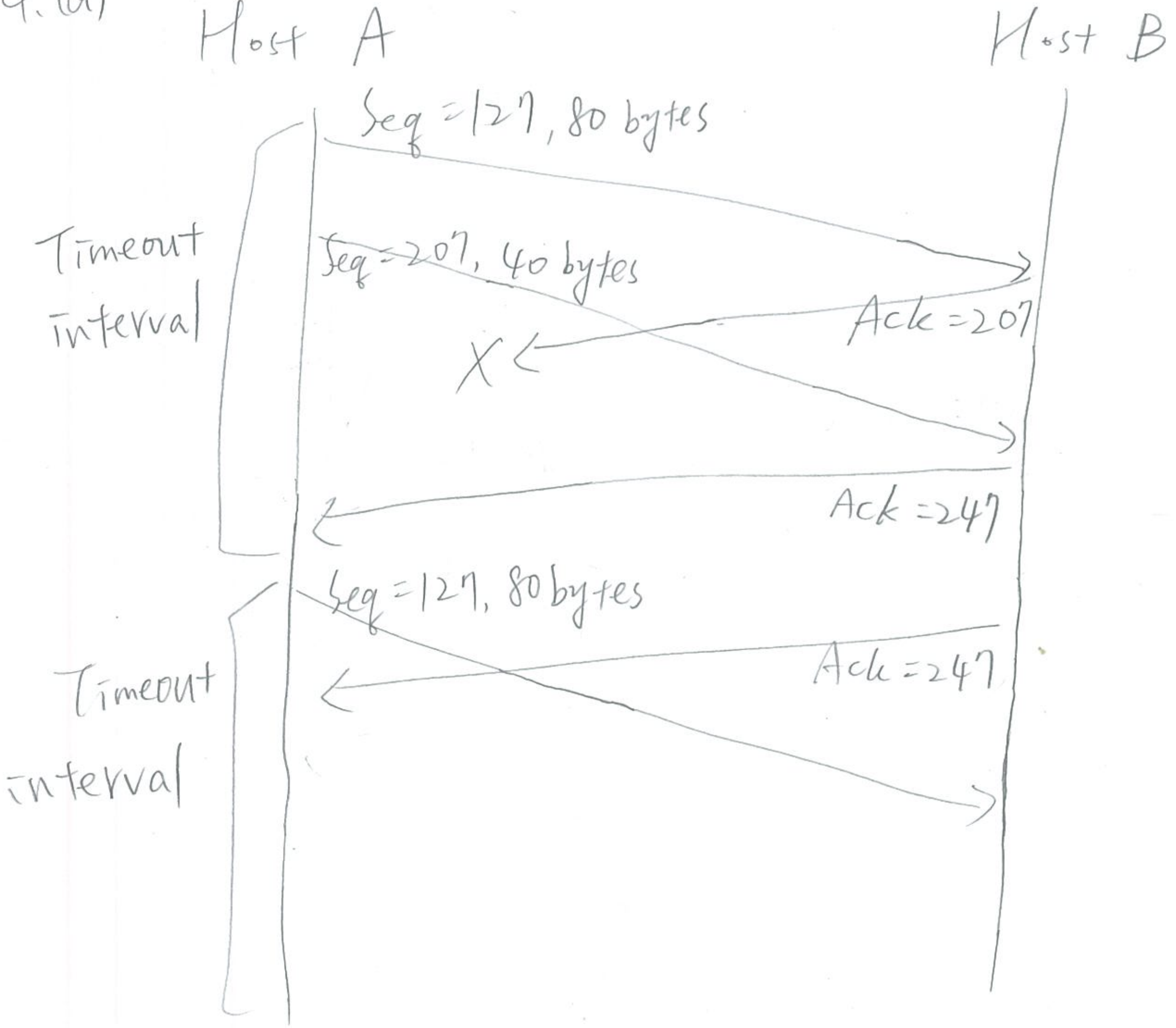
3. (b) If the receiver is waiting for packet k, then it has received (and ACKed) packet $k-1$ and the $N-1$ packets before that. If none of those N ACKs have been yet received by the sender, then ACK messages with values of $[k-N, k-1]$ may still be propagating back. Because the sender has sent packets $[k-N, k-1]$, it must be the case that the sender has already received an ACK for $k-N-1$. Once the receiver has sent an ACK for $k-N-1$, it will never send an ACK that is less that $k-N-1$. Thus, the range of in-flight ACK values can range from $k-N-1$ to $k-1$.

4. (a) In the second segment from Host A to B, the sequence number is 207; source port number is 302 and destination port number is 80.

4. (b) If the first segment arrives before the second, in the acknowledgement of the first arriving segment, the acknowledgement number is 207, the source port number is 80, and the destination port number is 302.

4. (c) If the second segment arrives before the first segment, in the acknowledgement of the first arriving segment, the acknowledge number is 127, indicating that it is still waiting for bytes 127 and onwords.

4. (d)



5.(a) TCP slow start is operating in the intervals [1,6] and [23, 26]

5.(b) TCP congestion avoidance is operating in the intervals [6, 16] and [17, 22]

5.(c) After the 16^{th} transmission round, packet loss is recognized by a triple duplicate ACK. If there was a timeout, the congestion window size would have dropped to 1.

5.(d) After the 22^{nd} transmission round, segment loss is detected due to timeout, and hence the congestion window size is set to 1.

5.(e) The threshold is initially 32, since it is at this window size that slow start stops and congestion avoidance begins.

5.(f) The threshold is set to half the value of the congestion window when packet loss is detected. When loss is detected during 22^{nd} transmission round, the congestion window size is 26. Hence, the threshold is 13 during the 24^{th} transmission round.

5.(g) The threshold is set to half the value of the congestion window when packet loss is detected. When loss is detected during 22^{nd} transmission round, the congestion windows size is 26. Hence the threshold is 13 during the 24^{th} transmission round.

5.(h) During the 1^{st} transmission round, packet 1 is sent; packet 2-3 are sent in the 2^{nd} transmission round; packets 4-7 are sent in the 3^{rd} transmission round; packets 8-15 are sent in the 4^{th} transmission round; packets 16-31 are sent in the 5^{th} transmission round; packets 32-63 are sent in the 6^{th} transmission round; packets 64-96 are sent in the 7^{th} transmission round. Thus packet 70 is sent in the 7th transmission round

5(i). The congestion window and threshold will be sent to half the current value of the congestion window (8) when the loss occurred. Thus, the new value of the threshold and window will be 4.

5(j). Threshold is 21. The congestion window size is first set to 1, and then grows to 4 in 19th round.