

Exam in SSY305 Kommunikationssystem

Department of Electrical Engineering

Exam date: August 19, 2019, 14:00–18:00

Document updated: August 19, 2019

Teaching Staff

Erik Ström (examiner), 031-772 5182

Chouaib Bencheikh Lehocine, 073-742 6431

Material Allowed material is

- Chalmers-approved calculator
- L. Råde, B. Westergren. Beta, Mathematics Handbook, any edition.
- One A4 page with your own handwritten notes. Both sides of the page can be used. Photo copies, printouts, other students' notes, or any other material is not allowed.
- A paper-based dictionary, without added notes (electronic dictionaries are not allowed).

Grading A correct, clear and well-motivated solution gives a maximum of 12 points per problem.

An erroneous answer, unclear, incomplete or badly motivated solutions give point reductions down to a minimum of 0 points. No fractional points are awarded.

Answers in any other language than Swedish or English are ignored.

Solutions Are made available at the earliest at 19:00 on the course web page.

Results Exam results are posted on Ping-Pong no later than on August 26, 2019. The grading review is on August 28 and September 4, 2019, 13:00–14:00 in room 6436 in the ED-building (Chouaib's office).

Grades The final grade on the course will be decided by the projects (maximum score 46), quizzes (maximum score 6), and final exam (maximum score 48). Project and exam must be passed (see course-PM for rules). The sum of all scores will decide the grade according to the following table.

Total Score	0–39	40–69	70–79	≥ 80
Grade	Fail	3	4	5

**PLEASE NOTE THAT THE PROBLEMS ARE NOT NECESSARILY
ORDERED IN DIFFICULTY.**

Good luck!

1. Suppose we use a systematic block code for error detection. The codewords are formed as $\mathbf{c} = [\mathbf{d} \ \mathbf{p}]$ where $\mathbf{d} = [d_0 \ d_1 \ d_2 \ d_3]$ are the information bits and $\mathbf{p} = [p_0 \ p_1]$ are the parity bits. The parity bits are computed as

$$p_0 = d_0 \oplus d_1 \oplus d_2$$

$$p_1 = d_1 \oplus d_2 \oplus d_3.$$

Let $\mathbf{y} = \mathbf{c} \oplus \mathbf{e}$ be the received word, where $\mathbf{e}$ is the error pattern.

- (a) Suppose $\mathbf{y} = [0 \ 0 \ 0 \ 1 \ 1 \ 0]$. Will the receiver declare that an error has occurred? Motivate. (2p)
 - (b) Suppose $\mathbf{y} = [0 \ 0 \ 0 \ 1 \ 0 \ 1]$. Will the receiver declare that an error has occurred? Motivate. (2p)
 - (c) Suppose $\mathbf{d} = [0 \ 0 \ 0 \ 0]$. List all undetectable error patterns. (4p)
 - (d) Let $\mathcal{C} = \{\mathbf{c}_0, \mathbf{c}_1, \dots, \mathbf{c}_{N-1}\}$ denote the set of all codewords. Show that the code is linear, i.e., that $\mathbf{c}_i \oplus \mathbf{c}_j \in \mathcal{C}$ for all $i, j \in \{0, 1, \dots, N-1\}$. (2p)
 - (e) Show that, since the code is linear, the set of undetectable error patterns for any $\mathbf{d}$ is the same as for $\mathbf{d} = [0 \ 0 \ 0 \ 0]$. (2p)
2. Consider an Internet-of-Things (IoT) scenario in which a large number of transmitters (things) wants to report measurements to a central location (basestation). The transmitters are distributed uniformly on a disk with radius $r = 10$ km with the basestation in the center. The transmitted frames are 40 byte long.

The physical layer has data rate R bit/s and the propagation speed is $c = 3 \times 10^8$ m/s.

We will consider two medium access protocols: Aloha and slotted Aloha. In the slotted case, synchronization is assumed to be perfect.

- (a) Suppose use Aloha as medium access. What is the best system throughput we can achieve for $R = 5$ Mbit/s? Answer in the unit [frames/second]. (3p)
- (b) Now consider slotted Aloha. The slot duration is the frame duration plus a guard interval. What is the smallest guard interval needed such that transmissions from different slots cannot collide at the basestation? (3p)
- (c) Repeat Part (a) for slotted Aloha. Use the slot duration from Part (b). (3p)
- (d) Suppose we want to improve throughput by increase the data rate R . For which data rate does Aloha and slotted Aloha have the same throughput? (3p)

Hint: Suppose the network transmitters generate, on the average, G frames per frame duration. Under certain conditions, which we assume to be satisfied in this problem, it can be shown that Aloha has system throughput S frames/(frame duration), where $S = G \exp(-2G)$. We can think of G as a free variable that we can adjust to maximize throughput.

Similarly, the system throughput for slotted Aloha is $S' = G' \exp(-G')$, where the system throughput S' and offered traffic G' is measured in frames/(slot duration). Note that S and S' are measured in different units.

3. (a) Explain what the hidden terminal problem is in Wi-Fi and how it can be mitigated using the RTS/CTS handshake. (4p)
- (b) Which ARQ protocol is preferable on links with small delay-bandwidth products and relative high frame error probabilities? Choose from Stop-And-Wait, Go-Back- N , and Selective Repeat. Motivate. (2p)
- (c) Suppose Alice want to communicate confidentially with Bob using asymmetric cryptography. Which key does Alice use to encrypt her messages to Bob? Choose from Alice's private key, Alice's public key, Bob's private key, or Bob's public key. Motivate (2p)
- (d) Consider a protocol in layer n . Which data unit has the most number of bits, the n -PDU or the n -SDU? Motivate. (2p)
- (e) Explain the purpose of the demodulator in Shannon's communication model. (2p)

4. Consider a Go-Back- N ARQ protocol with transmitter state variables

S_f = sequence number of first (oldest) outstanding packet

S_n = sequence number of next packet to send (which is not yet received from upper layer)

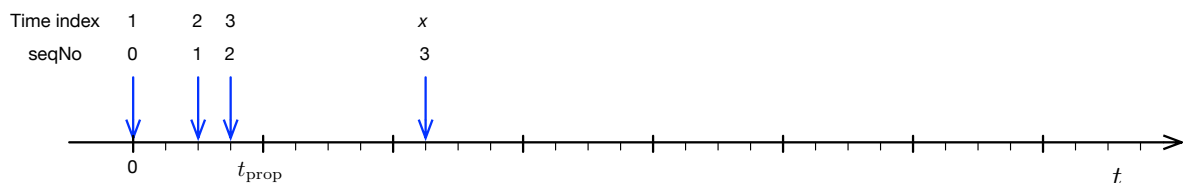
and receiver state variable

R_n = sequence number of next expected packet.

The protocol is initialized as $S_f = S_n = R_n = 0$ and an empty send window. If there are no outstanding packets, then by convention we set $S_n = S_f$. The send window size is 3. Cumulative ACKs are used, i.e., an ACK packet with acknowledgement number (ackNo) A implies that all packets with sequence numbers (seqNo) less than A have been received. The transmitter (TX) and receiver (RX) is connected via a full-duplex link.

Suppose processing times and packet durations are negligible compared to the propagation time t_{prop} . The timeout is set to $4t_{\text{prop}}$.

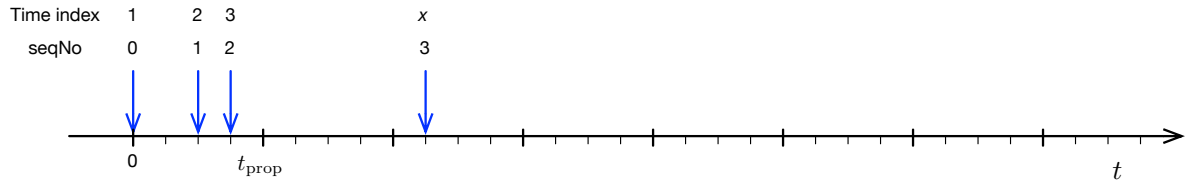
The packet arrivals from the higher layer to the ARQ protocol in the transmitter node is depicted in the figure below.



We recall that the state variables and send window content can change when events occur. The relevant events are arrivals of information packets from higher layer, transmissions of packets, receptions of packets, and timeouts.

- Assume that information and ACK packets are transmitted error-free. Copy the timeline above and indicate when information and ACK packets are transmitted and received. Label the time instances with the corresponding seqNo and ackNo of the transmitted and received packets. Moreover, enumerate the time instances starting from time index 1 (see timeline above). (2p)
- Explain how the state variables and send window changes at the time instances marked in the timeline from Part (a). Do this by filling out the table on one of the attached paper at the end of this exam. Note that several events can occur at a single time index, see the example on the next page. (3p)
- Suppose the ACK corresponding to the information packet with seqNo = 1 is lost. Repeat Part (b). (2p)
- Suppose that the transmission of the information packet with seqNo = 1 is lost, but that all other future transmissions of information and ACK packets are error-free. Repeat Part (a), i.e., copy the timeline above and add all events to it. (3p)
- Explain how the state variables and send window changes at the time instances marked in the timeline from Part (d). Do this by filling out the table on one of the attached paper at the end of this exam. (2p)

Each row in the event table defines a single event: a timeout, or a packet arrival, packet transmission, or packet reception. In the example below, we record the events that occur at time index 1 and 2 in the timeline below.



The arrival of the first packet to the ARQ protocol in the TX node implies two events: the arrival itself and the subsequent transmission of the packet. The same is true for the arrival of the second packet.

Note: only the shaded parts of the table is filled out. Hence, the values of the state variables and send window are not filled out (this is part of the exam problem).

Time Index when event occur	Node where event occurs		Type of event				Sequence number of the packet involved in event							
	TX	RX	<u>Arriv</u>	Trans	Rec	Time out	<u>seqNo</u>	<u>ackNo</u>	S_f	S_n	R_n	Send Window		
<i>Start</i>									0	0	0	—	—	—
1	X		X				0							
	X			X			0							
2	X		X				1							
	X			X			1							

Detailed explanation of the first three rows in the table:

Row *Start*: defines the initial values of the state variables (all zero) and the send window (empty).

Row 2: explains that the first event

- occurs at time index 1,
- occurs at the TX node,
- is an arrival event, and
- that the involved packet has seqNo = 0

Row 3: explains that the second event

- occurs at time index 1,
- occurs at the TX node,
- is a transmission event, and
- that the involved packet has seqNo = 0

Row 4 and 5 follows the same pattern, but with different time index and seqNo.

Answer by filling out blank (non-shaded) parts of the table.

[illegible]

Answer by filling out blank (non-shaded) parts of the table.

[illegible]

Answer by filling out blank (non-shaded) parts of the table.

[illegible]

Answer by filling out blank (non-shaded) parts of the table.

[illegible]

Exam in SSY305 Kommunikationssystem

Department of Electrical Engineering

Exam date: August 19, 2019, 14:00–18:00

Document updated: August 21, 2019

Teaching Staff

Erik Ström (examiner), 031-772 5182

Chouaib Bencheikh Lehocine, 073-742 6431

Material Allowed material is

- Chalmers-approved calculator
- L. Råde, B. Westergren. Beta, Mathematics Handbook, any edition.
- One A4 page with your own handwritten notes. Both sides of the page can be used. Photo copies, printouts, other students' notes, or any other material is not allowed.
- A paper-based dictionary, without added notes (electronic dictionaries are not allowed).

Grading A correct, clear and well-motivated solution gives a maximum of 12 points per problem.

An erroneous answer, unclear, incomplete or badly motivated solutions give point reductions down to a minimum of 0 points. No fractional points are awarded.

Answers in any other language than Swedish or English are ignored.

Solutions Are made available at the earliest at 19:00 on the course web page.

Results Exam results are posted on Ping-Pong no later than on August 26, 2019. The grading review is on August 28 and September 4, 2019, 13:00–14:00 in room 6436 in the ED-building (Chouaib's office).

Grades The final grade on the course will be decided by the projects (maximum score 46), quizzes (maximum score 6), and final exam (maximum score 48). Project and exam must be passed (see course-PM for rules). The sum of all scores will decide the grade according to the following table.

Total Score	0–39	40–69	70–79	≥ 80
Grade	Fail	3	4	5

**PLEASE NOTE THAT THE PROBLEMS ARE NOT NECESSARILY
ORDERED IN DIFFICULTY.**

Good luck!

1. Suppose we use a systematic block code for error detection. The codewords are formed as $\mathbf{c} = [\mathbf{d} \ \mathbf{p}]$ where $\mathbf{d} = [d_0 \ d_1 \ d_2 \ d_3]$ are the information bits and $\mathbf{p} = [p_0 \ p_1]$ are the parity bits. The parity bits are computed as

$$p_0 = d_0 \oplus d_1 \oplus d_2$$

$$p_1 = d_1 \oplus d_2 \oplus d_3.$$

Let $\mathbf{y} = \mathbf{c} \oplus \mathbf{e}$ be the received word, where $\mathbf{e}$ is the error pattern.

- (a) Suppose $\mathbf{y} = [0 \ 0 \ 0 \ 1 \ 1 \ 0]$. Will the receiver declare that an error has occurred? Motivate. (2p)
- (b) Suppose $\mathbf{y} = [0 \ 0 \ 0 \ 1 \ 0 \ 1]$. Will the receiver declare that an error has occurred? Motivate. (2p)
- (c) Suppose $\mathbf{d} = [0 \ 0 \ 0 \ 0]$. List all undetectable error patterns. (4p)
- (d) Let $\mathcal{C} = \{\mathbf{c}_0, \mathbf{c}_1, \dots, \mathbf{c}_{N-1}\}$ denote the set of all codewords. Show that the code is linear, i.e., that $\mathbf{c}_i \oplus \mathbf{c}_j \in \mathcal{C}$ for all $i, j \in \{0, 1, \dots, N-1\}$. (2p)
- (e) Show that, since the code is linear, the set of undetectable error patterns for any $\mathbf{d}$ is the same as for $\mathbf{d} = [0 \ 0 \ 0 \ 0]$. (2p)

Solution

- (a) The receiver will declare error because

$$\hat{d}_0 \oplus \hat{d}_1 \oplus \hat{d}_2 = 0 \neq \hat{p}_0$$

$$\hat{d}_1 \oplus \hat{d}_2 \oplus \hat{d}_3 = 1 \neq \hat{p}_1$$

and $\mathbf{y}$ is therefore not a valid codeword.

- (b) The receiver will not declare an error since

$$\hat{d}_0 \oplus \hat{d}_1 \oplus \hat{d}_2 = 0 = \hat{p}_0$$

$$\hat{d}_1 \oplus \hat{d}_2 \oplus \hat{d}_3 = 1 = \hat{p}_1$$

and $\mathbf{y}$ is therefore a valid codeword.

- (c) If the data vector is $\mathbf{d} = [0 \ 0 \ 0 \ 0]$, then the transmitted vector is $\mathbf{c}_0 = [0 \ 0 \ 0 \ 0 \ 0 \ 0]$. The receiver will not declare error only if the received vector $\mathbf{y} = \mathbf{c}_0 \oplus \mathbf{e} = \mathbf{e} \in \mathcal{C}$. Hence, if $\mathcal{E}$ is the set of all undetectable errors for the data vector $\mathbf{d} = [0 \ 0 \ 0 \ 0]$, then $\mathcal{E} = \mathcal{C} \setminus \{\mathbf{c}_0\}$. Following that, the undetectable error patterns are

$$\begin{array}{ll} [0 \ 0 \ 0 \ 1 \ 0 \ 1] & [0 \ 0 \ 1 \ 0 \ 1 \ 1] \\ [0 \ 0 \ 1 \ 1 \ 1 \ 0] & [0 \ 1 \ 0 \ 0 \ 1 \ 1] \\ [0 \ 1 \ 0 \ 1 \ 1 \ 0] & [0 \ 1 \ 1 \ 0 \ 0 \ 0] \\ [0 \ 1 \ 1 \ 1 \ 0 \ 1] & [1 \ 0 \ 0 \ 0 \ 1 \ 0] \\ [1 \ 0 \ 0 \ 1 \ 1 \ 1] & [1 \ 0 \ 1 \ 0 \ 0 \ 1] \\ [1 \ 0 \ 1 \ 1 \ 0 \ 0] & [1 \ 1 \ 0 \ 0 \ 0 \ 1] \\ [1 \ 1 \ 0 \ 1 \ 0 \ 0] & [1 \ 1 \ 1 \ 0 \ 1 \ 0] \\ [1 \ 1 \ 1 \ 1 \ 1 \ 1] & \end{array}$$

(d) Let $\mathbf{c}_i = [d_0 \ d_1 \ d_2 \ d_3 \ p_0 \ p_1]$ and $\mathbf{c}_j = [d'_0 \ d'_1 \ d'_2 \ d'_3 \ p'_0 \ p'_1]$, then we have

$$\begin{aligned}(p_0 \oplus p'_0) &= (d_0 \oplus d'_0) \oplus (d_1 \oplus d'_1) \oplus (d_2 \oplus d'_2) \\ (p_1 \oplus p'_1) &= (d_1 \oplus d'_1) \oplus (d_2 \oplus d'_2) \oplus (d_3 \oplus d'_3).\end{aligned}$$

hence we can conclude that $\mathbf{c}_i \oplus \mathbf{c}_j = \mathbf{c}_k \in \mathcal{C}$, where $\mathbf{c}_k$ is the codeword with information bits $d''_n = d_n \oplus d'_n$.

(e) From the answer to question (c) we know that the set of undetectable errors for the data vector $\mathbf{d} = [0 \ 0 \ 0 \ 0]$ is $\mathcal{E} = \mathcal{C} \setminus \{\mathbf{c}_0\}$. Let $\mathbf{c}_i$ be the transmitted code word. We have that $\mathbf{y} = \mathbf{c}_i \oplus \mathbf{e}$. By definition, $\mathbf{e}$ is undetectable if $\mathbf{e} \neq \mathbf{c}_0$ and $\mathbf{y} \in \mathcal{C}$. If $\mathbf{e}$ is a nonzero codeword, i.e., $\mathbf{e} \in \mathcal{E}$, then $\mathbf{y} = \mathbf{c}_i \oplus \mathbf{e}$ is also a codeword since the code is linear and $\mathbf{e}$ is therefore an undetectable error.

2. Consider an Internet-of-Things (IoT) scenario in which a large number of transmitters (things) wants to report measurements to a central location (basestation). The transmitters are distributed uniformly on a disk with radius $r = 10$ km with the basestation in the center. The transmitted frames are 40 byte long.

The physical layer has data rate R bit/s and the propagation speed is $c = 3 \times 10^8$ m/s.

We will consider two medium access protocols: Aloha and slotted Aloha. In the slotted case, synchronization is assumed to be perfect.

- Suppose use Aloha as medium access. What is the best system throughput we can achieve for $R = 5$ Mbit/s? Answer in the unit [frames/second]. (3p)
- Now consider slotted Aloha. The slot duration is the frame duration plus a guard interval. What is the smallest guard interval needed such that transmissions from different slots cannot collide at the basestation? (3p)
- Repeat Part (a) for slotted Aloha. Use the slot duration from Part (b). (3p)
- Suppose we want to improve throughput by increase the data rate R . For which data rate does Aloha and slotted Aloha have the same throughput? (3p)

Hint: Suppose the network transmitters generate, on the average, G frames per frame duration. Under certain conditions, which we assume to be satisfied in this problem, it can be shown that Aloha has system throughput S frames/(frame duration), where $S = G \exp(-2G)$. We can think of G as a free variable that we can adjust to maximize throughput.

Similarly, the system throughput for slotted Aloha is $S' = G' \exp(-G')$, where the system throughput S' and offered traffic G' is measured in frames/(slot duration). Note that S and S' are measured in different units.

Solution

- For Aloha, the maximum normalized throughput is achieved for $G = 1/2$. We have $\frac{dS}{dG} = 0 \rightarrow G = 1/2$, hence $S_{\max} = \frac{1}{2}e^{-1}$ [frames/frame duration]. We compute the frame duration T_f .

$$T_f = \frac{40 \times 8}{R} = 64 \mu s \quad (1)$$

The maximum achievable throughput is

$$R_{\text{eff}} = \frac{S_{\text{max}}}{T_f} = 2.874 \times 10^3 \text{ [frames/s]} \quad (2)$$

$$= 0.92 \text{ Mbit/s.} \quad (3)$$

- (b) Let τ be the maximum propagation time from any station to the base station, $\tau = \frac{10^4}{c} = 1/3 \times 10^{-4}$ s. Now let's assume that a station is located 10 km away from the base station, when the station starts transmitting at the beginning of a new slot, the frame will reach the base station after $T_f + \tau$. Hence, the guard interval should be $T_G = \tau$, to ensure that transmission from different slots will not collide at the basestation.

- (c) For slotted Aloha, the maximum normalized throughput is achieved for $G' = 1$. This can be obtained as follows

$\frac{dS'}{dG'} = 0 \rightarrow G' = 1$, hence $S'_{\text{max}} = e^{-1}$ [frames/slot duration]. The slot duration is

$$T_{\text{slot}} = T_f + T_G \quad (4)$$

$$= 97.333\mu. \quad (5)$$

The maximum achievable throughput for slotted aloha is

$$R'_{\text{eff}} = \frac{S'_{\text{max}}}{T_{\text{slot}}} = 3.78 \times 10^3 \text{ [frames/ s]} \quad (6)$$

$$= 1.21 \text{ Mbit/s.} \quad (7)$$

- (d) We write the maximum achievable throughput of Aloha and slotted Aloha as function of R

$$R_{\text{eff}} = R \times \frac{e^{-1}}{2 \times 320} \quad (8)$$

$$R'_{\text{eff}} = R \times \frac{e^{-1}}{320 + \tau R}. \quad (9)$$

Then, if we let $320 + \tau R = 2 \times 320$, we obtain

$$R = \frac{320}{\tau} = 9.6 \text{ Mbit/s.} \quad (10)$$

3. (a) Explain what the hidden terminal problem is in Wi-Fi and how it can be mitigated using the RTS/CTS handshake. (4p)
- (b) Which ARQ protocol is preferable on links with small delay-bandwidth products and relative high frame error probabilities? Choose from Stop-And-Wait, Go-Back- N , and Selective Repeat. Motivate. (2p)
- (c) Suppose Alice want to communicate confidentially with Bob using asymmetric cryptography. Which key does Alice use to encrypt her messages to Bob? Choose from Alice's private key, Alice's public key, Bob's private key, or Bob's public key. Motivate (2p)
- (d) Consider a protocol in layer n . Which data unit has the most number of bits, the n -PDU or the n -SDU? Motivate. (2p)
- (e) Explain the purpose of the demodulator in Shannon's communication model. (2p)

Solution

- (a) The hidden-terminal problem occurs in carrier-sense multiple access systems when an on-going transmission might not be detected by all nodes. This is a problem in practice for Wi-Fi networks. Suppose Node A and Node B are out of radio range of each other (i.e., they cannot detect each others transmissions). However, the nodes are in radio range of the access point (AP). Suppose that Node A transmits a frame to the AP. During the transmission, Node B senses the channel to determine if it also can transmit a frame. Node B will not hear Node A's transmission and therefore thinks (wrongly) that the channel is idle and starts a transmission. Node A and Node B's transmissions will then collide at the AP, which could lead to that one or both of the frames are lost. Here we say that Node A is a hidden terminal to Node B. To counteract this problem, we can use a request-to-send/clear-to-send (RTS/CTS) handshake procedure. The handshake work like this: if Node A wants to transmit a data frame to the AP, it first transmits a RTS frame to the AP. The AP answers the request by sending a CTS frame to Node A. The latter transmission, i.e, the CTS from the AP to Node A is also picked up by Node B. Hence, Node B now knows that the channel will be busy, in spite of the fact that Node B did not hear the RTS packet. (The CTS packet contains information on how long the channel will be busy in order to complete transmission of the subsequent information frame from Node A.) The RTS/CTS procedure is not perfect since simultaneously transmitted RTS frames from Node A and Node B can collide. However, since the RTS/CTS frames are much shorter than typical data frames, the probability for RTS collision is smaller than the probability for data frame collisions (when RTS/CTS is not used). However, RTS/CTS implies a significant overhead and is not used very often in practice.
- (b) For low delay-bandwidth products, the Stop-And-Wait efficiency is close to optimal and it has the lowest complexity among the considered ARQ protocols. Using Go-Back- N only improves on Stop-And-Wait when the frame error probability is low. Hence, there is very little, if any, gain in efficiency with Go-Back- N and it is not an attractive choice due to complexity reasons. Selective-Repeat is more robust against frame errors, but since the performance gain compared to Stop-And-Wait is marginal and complexity is high, Selective-Repeat is not attractive. In summary, Stop-And-Wait is probably the best choice.
- (c) Alice will use Bob's public key to encrypt messages to Bob and Bob will use his private key to decrypt the message. The public keys are not secret—hence if these

cannot be used for decrypting messages that are supposed to be confidential (since anyone can use the public key). Alice cannot use her own public key to encrypt, since no one, including Bob, should know her private key. Hence, the only possibility is for Alice to encrypt with Bob's public key (which is known to Alice). To summarize, everyone can encrypt messages to Bob, but since only Bob knows the Bob's private, only Bob can decrypt the message.

- (d) Since the the n -SDU is encapsulated into the n -PDU, the n -PDU will (in general) have more bits than the n -SDU.
- (e) The purpose of the demodulator is to convert the received analog signal to decisions on the transmitted bits.

4. Consider a Go-Back- N ARQ protocol with transmitter state variables

S_f = sequence number of first (oldest) outstanding packet

S_n = sequence number of next packet to send (which is not yet received from upper layer)

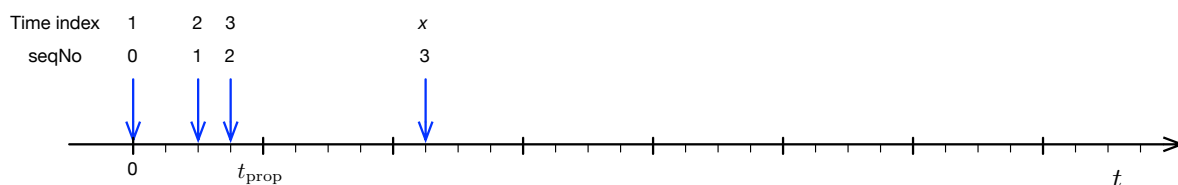
and receiver state variable

R_n = sequence number of next expected packet.

The protocol is initialized as $S_f = S_n = R_n = 0$ and an empty send window. If there are no outstanding packets, then by convention we set $S_n = S_f$. The send window size is 3. Cumulative ACKs are used, i.e., an ACK packet with acknowledgement number (ackNo) A implies that all packets with sequence numbers (seqNo) less than A have been received. The transmitter (TX) and receiver (RX) is connected via a full-duplex link.

Suppose processing times and packet durations are negligible compared to the propagation time t_{prop} . The timeout is set to $4t_{\text{prop}}$.

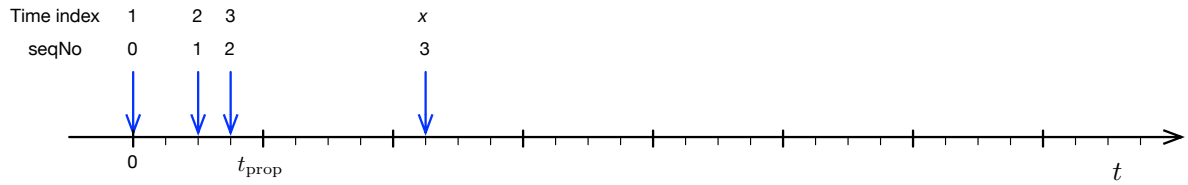
The packet arrivals from the higher layer to the ARQ protocol in the transmitter node is depicted in the figure below.



We recall that the state variables and send window content can change when events occur. The relevant events are arrivals of information packets from higher layer, transmissions of packets, receptions of packets, and timeouts.

- Assume that information and ACK packets are transmitted error-free. Copy the timeline above and indicate when information and ACK packets are transmitted and received. Label the time instances with the corresponding seqNo and ackNo of the transmitted and received packets. Moreover, enumerate the time instances starting from time index 1 (see timeline above). (2p)
- Explain how the state variables and send window changes at the time instances marked in the timeline from Part (a). Do this by filling out the table on one of the attached paper at the end of this exam. Note that several events can occur at a single time index, see the example on the next page. (3p)
- Suppose the ACK corresponding to the information packet with seqNo = 1 is lost. Repeat Part (b). (2p)
- Suppose that the transmission of the information packet with seqNo = 1 is lost, but that all other future transmissions of information and ACK packets are error-free. Repeat Part (a), i.e., copy the timeline above and add all events to it. (3p)
- Explain how the state variables and send window changes at the time instances marked in the timeline from Part (d). Do this by filling out the table on one of the attached paper at the end of this exam. (2p)

Each row in the event table defines a single event: a timeout, or a packet arrival, packet transmission, or packet reception. In the example below, we record the events that occur at time index 1 and 2 in the timeline below.



The arrival of the first packet to the ARQ protocol in the TX node implies two events: the arrival itself and the subsequent transmission of the packet. The same is true for the arrival of the second packet.

Note: only the shaded parts of the table is filled out. Hence, the values of the state variables and send window are not filled out (this is part of the exam problem).

Time Index when event occur	Node where event occurs		Type of event				Sequence number of the packet involved in event							
	TX	RX	Arriv	Trans	Rec	Time out	seqNo	ackNo	S_f	S_n	R_n	Send Window		
<i>Start</i>									0	0	0	—	—	—
1	X		X				0							
	X			X			0							
2	X		X				1							
	X			X			1							

Detailed explanation of the first three rows in the table:

Row *Start*: defines the initial values of the state variables (all zero) and the send window (empty).

Row 2: explains that the first event

- occurs at time index 1,
- occurs at the TX node,
- is an arrival event, and
- that the involved packet has seqNo = 0

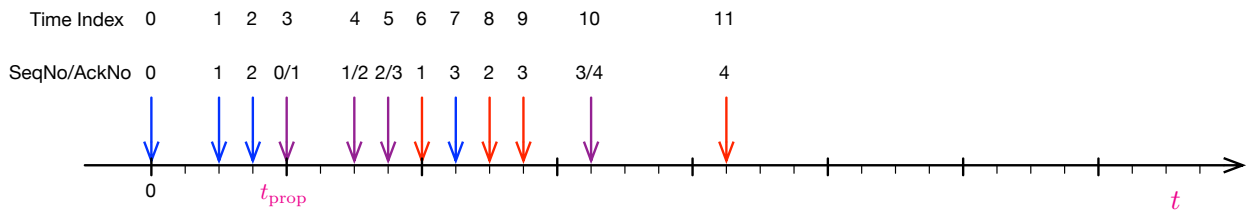
Row 3: explains that the second event

- occurs at time index 1,
- occurs at the TX node,
- is a transmission event, and
- that the involved packet has seqNo = 0

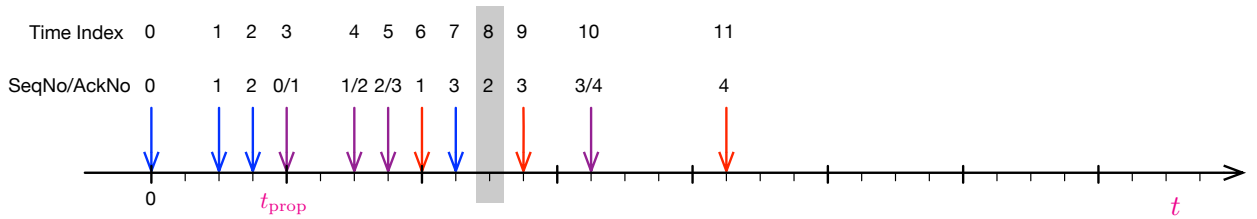
Row 4 and 5 follows the same pattern, but with different time index and seqNo.

The timelines and event tables are as follows.
Timelines for Part (a) and (d)

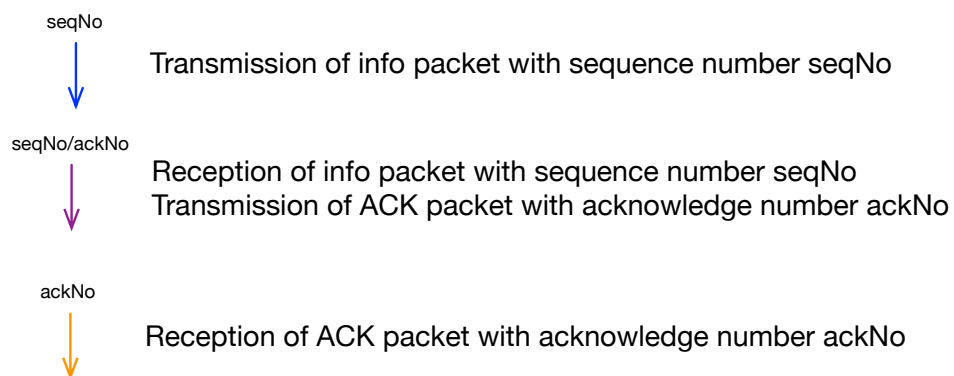
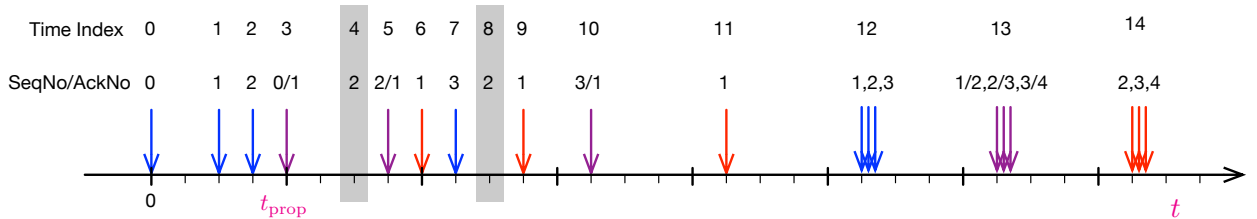
No info packet/ACK loss



ACK with ackNo = 2 lost



Packet with seqNo = 1 lost



Part (b)

No lost information or ACK packets

Time Index when event occur	Node where event occurs		Type of event			Sequence number of the packet involved in event							
	TX	RX	Arrival	Trans	Recep	seqNo	ackNo	S_f	S_n	R_n	Send Window		
Start								0	0	0	—	—	—
0	X		X			0			1		0	—	—
	X			X		0							
1	X		X			1			2		0	1	—
	X			X		1							
2	X		X			2			3		0	1	2
	X			X		2							
3		X			X	0				1			
		X		X			1						
4		X			X	1				2			
		X		X			2						
5		X			X	2				3			
		X		X			3						
6	X				X		1	1			1	2	—
7	X		X			3			4		1	2	3
	X			X		3							
8	X				X		2	2			2	3	—
9	X				X		3	3			3	—	—
10		X			X	3				4			
		X		X			4						
11	X				X		4	4			—	—	—

Part (c)

ACK of packet with seqNo = 1 lost, i.e., the ACK packet with ackNo = 2

Time Index when event occur	Node where event occurs		Type of event			Sequence number of the packet involved in event							
	TX	RX	Arrival	Trans	Recep	seqNo	ackNo	S_f	S_n	R_n	Send Window		
Start								0	0	0	—	—	—
0	X		X			0			1		0	—	—
	X			X		0							
1	X		X			1			2		0	1	—
	X			X		1							
2	X		X			2			3		0	1	2
	X			X		2							
3		X			X	0				1			
		X		X			1						
4		X			X	1				2			
		X		X			2						
5		X			X	2				3			
		X		X			3						
6	X				X		1	1			1	2	—
7	X		X			3			4		1	2	3
	X			X		3							
8	X						2	1			2	3	—
9	X				X		3	3			3	—	—
10		X			X	3				4			
		X		X			4						
11	X				X		4	4			—	—	—

Note that at time instant 8, the ACK with AckNo = 2 is never received. Hence, the TX node cannot update S_f . However, at time instant 9, the ACK with AckNo = 3 is received and the TX node sets $S_f = 3$. The state variables at the TX and RX nodes are then identical to the case when there are no lost ACK frames. Hence, the protocol has recovered without any extra transmissions.

Part (e)

Lost packet with SeqNo = 1

Time Index when event occur	Node where event occurs		Type of event			Sequence number of the packet involved in event							
	TX	RX	Arrival	Trans	Recep	seqNo	ackNo	S_f	S_n	R_n	Send Window		
Start								0	0	0	—	—	—
0	X		X			0			1		0	—	—
	X			X		0							
1	X		X			1			2		0	1	—
	X			X		1							
2	X		X			2			3		0	1	2
	X			X		2							
3		X			X	0				1			
		X		X			1						
4		X				1				1			
		X											
5		X			X	2				1			
		X		X			1						
6	X				X		1	1			1	2	—
7	X		X			3			4		1	2	3
	X			X		3							
8	X						1	1			1	2	3
9	X				X		1	1			1	2	3
10		X			X	3				1			
		X		X			1						
11	X				X		1	1			1	2	3
12	X			X		1		1	2		1	2	3
	X			X		2		1	3		1	2	3
	X			X		3		1	4		1	2	3
13		X			X	1				2			
		X		X			2						
		X			X	2				3			
		X		X			3						
		X			X	3				4			
		X		X		4							
14	X					2		2			2	3	—
	X					3		3			3	—	—
	X					4		4			—	—	—