ZAQ

126/200

Midterm # 1, due by noon, Tuesday 07.19.16 (This is an individual test, worth 200 points. You may consult any printed/web-based resource you want, but you may not discuss this test with anybody.)

July 18, 2016

1. **[this problem is worth 40 points total]** Let D be the bitstream received by the DLC layer from the IP layer. The bitstream D maps to the equivalent polynomial $D(x)$ and *vice versa*; $D \leftrightarrow D(x)$. Also, let $G(x)$ be the generator polynomial of a CRC code and G its equivalent bit-domain representation; $G \leftrightarrow G(x)$.

$\boxed{\text{Suppose } G(x) = x^4 + x + 1 \text{ and } D = 1001101}$

(a) Find G. [2 points]

10011

(b) Find $D(x)$. [2 points]

$x^6 + x^3 + x^2 + 1$

(c) Let β be the degree of $G(x)$. What is β? [2 points]

4

(d) Let $A(x) = D(x)\,x^{\beta}$. Find $A(x)$. [2 points]

$x^{10} + x^7 + x^6 + x^4$

(e) Let A be the bit-domain representation of $A(x)$. What is A? [2 points]

10011010000

(f) Let $Q(x)$ be the quotient when $A(x)$ is divided by $G(x)$ modulo 2 and $R(x)$ the remainder. Find $Q(x)$ and $R(x)$. [5 + 5 = 10 points]

$Q(x) = x^6 + 1$ $\qquad$ $R(x) = x^1 + 1$

(g) Let Q and R be the equivalent bit-domain representations of $Q(x)$ and $R(x)$ respectively. Find Q and R. [2 + 2 = 4 points]

Q = 1000001   R = 0011

(h) Let $P(x) = A(x) \ominus R(x)$, where '$\ominus$' denotes subtraction using modulo 2 arithmetic. Find $P(x)$. [2 points]

$x^{10} + x^7 + x^6 + x^4 + x^1 + 1$

(i) Let P be the bit domain representation of $P(x)$. This is the transmitted frame after error control coding. What is P? [2 points]

10011010011

(j) Find an example of an error pattern (i.e., instead of receiving P, the receiver receives the bitstream P') which would go undetected. Also explain in detail why that error pattern will go undetected. [12 points]

"undetected" p(x) could be a multiple to G(x)

$$[x^4 + x + 1]\,[x^6 + 1 + \overbrace{q'(x)}^{=1}] = R'(x)$$

$$= x^{10} + x^7 + x^6$$

10011000000

$2^5 - 1 = 31$

$51 - 31 = 20$

-20 modulo $31 = 11 = 01011$

27 3 0 21

*

2. [**this problem is worth** $10 \times 4 = 40$ **points total**] A 5-bit checksum is to be appended to the following sequence of four 5-bit words [11011, 00011, 00000, 10101, checksum].

(a) Write down the checksum (in binary form).

01011

(b) Suppose the channel flips the 11^{th} and 15^{th} bits of the transmitted frame (i.e., the first and last bits of the third 5-bit word). Will the receiver be able to detect this error? Explain your answer.

11011, 00011, 10001, 10101

(27 + 3 + 17 + 21)

$b_0 + b_1 + b_2 + b_3$ modulo $31 \neq 0$

Yes, the answer should equal to 0

what's the answer you get??

(c) Suppose the channel flips the 1^{st}, 2^{nd}, 6^{th} and 7^{th} bits of the transmitted frame (i.e., the first two bits of the first and second words). Will the receiver be able to detect this error? Explain your answer.

No, basically you switched b_0 and b_1 and check sum can't detect such errors since $b_1 + b_0 + b_2 + b_3$ modulo $31 = 0$

(d) Suppose the channel flips the 11^{th} bit of the transmitted frame. Will the receiver be able to **correct** this error? Provide proper explanation if your answer is yes. If your answer is no, you can provide a counterexample to prove your point. Of course, you can assume that the receiver does not know how many errors the channel made.

-6

No, The checksum is only for error detection, not correction.

There might be more than an error, Which means there will be more than a soloution to satisfy $b_0 + b_1 + b_2 + b_3$ modulo $31 = 0$

when??

3. **[this problem is worth 20 points total]** Suppose the English alphabet only contained the characters a, e, i, o, and u. All possible messages can use only these 5 characters. You want to map each of these characters to 8-bit codewords (there's no parity check bit) **such that the Hamming Distance of the dictionary is maximized** (i.e, you want to separate out these three characters in bit space as much as possible).

   (a) How would you map the letters to codewords? [5 points]

   0000 0000
   0000 1111
   1111 0000
   1010 1010
   1011 1111

   −3

   (b) What is the Hamming Distance of your dictionary? [5 points]

   4 with your dictionary, you should get 3.

   if you change this to '1', you will get a '4'.

   (c) What is the maximum number of characters you can use from the alphabet so that you could **correct** up to all 2-bit errors, assuming that each character is still represented by 8 bits? How would you map those characters to codewords? [10 points]

   the dictionary should be $2 \times 2 + 1 = 5$

   saying

   $$\begin{array}{cc} a & \\ e & \end{array}\begin{bmatrix} 0000 & 0000 \\ 1000 & 1111 \end{bmatrix}$$

   but you are using a dictionary of '2'.

   $$\left\lfloor \frac{5-1}{2} \right\rfloor = 2 \text{ errors}$$

4. **[this problem is worth 35 points total]** Consider a Go back N protocol **without any errors**. The transmitter is at liberty to dump up to N frames, without waiting for any acknowledgement from the receiver. Suppose it starts the transmission process at time $t = t_0$. The first ACK from the receiver (in response to the first frame sent by the transmitter) arrives back at the transmitter at time $t = t_1$, where $t_1 > t_0$. Suppose the time interval $t_1 - t_0$ is adequate for the transmitter to transmit a batch of N frames to the receiver (i.e., it has finished transmitting N frames within that interval). Then, the **effective frame transmission rate** is $\frac{N}{t_1 - t_0}$ frames/sec. This formula can also be used for the Stop-and-Wait protocol since it is a special case of the Go Back N protocol with $N = 1$.

   Now consider the network in Fig. 1. Node A is using node B as a relay to send frames to node C. The Go Back N protocol is used on the link (A, B) with $N = 3$ and the Stop-and-Wait protocol is used on the link (B, C). The following parameters are given:

   - data frames sent by node A are 1000 bits each
   - all ACK frames sent by either B or C are 100 bits each
   - the propagation delay on both links is 5×10^{-6} **second/km**. Note the unit.
   - processing delays at all nodes are negligible.
   - the data rate of the link (A, B) is 100 Kbps (kilobits per second).
   - the data rate of the link (B, C) is unknown, say x bps.

   (a) What should x be so that there is no buffer overflow at B in the long run? (**Hint**: For the input buffer at B not to overflow, the effective frame transmission rate on the outgoing link (B, C) must be greater than or equal to the effective frame transmission rate on the incoming link (A, B). In order to solve this problem, you would need to find the effective frame transmission rate on the link (A, B) and the effective frame transmission rate on the link (B, C) which will be in terms of x. Equate the two effective frame transmission rates and solve for x.) [25 points]

   (b) As you know, setting proper timeout periods is critical to preventing unnecessary transmissions. How should nodes A and B set their timers? You can't find a proper timer value for node B unless you have solved part (a) above. [10 points]

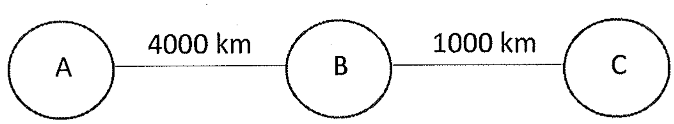


Figure 1: Network for Problem 4.

A

$\boxed{L_D = 1000 \qquad R = 100 \times 10^3}$ $\quad N = 3$

see soln

$$EITR_{A-B} = \frac{3 \times 1000}{2(4 \times 10^6 \times 5 \times 10^{-9}) + \left(\frac{3000 + 100}{100 \times 10^3}\right)}$$

25

I can't figure out your logic here

B_C

$$= \frac{1000}{2(1 \times 10^6 \times 5 \times 10^{-9}) + \left(\frac{1000 + 100}{x}\right)}$$

$$x = \frac{3300000}{41} = 80487.8 \text{ bps}$$

??

B

$$T_{A-B} = 2(4 \times 10^6 \times 5 \times 10^{-9}) + \frac{3000 + 100}{100 \times 10^3}$$

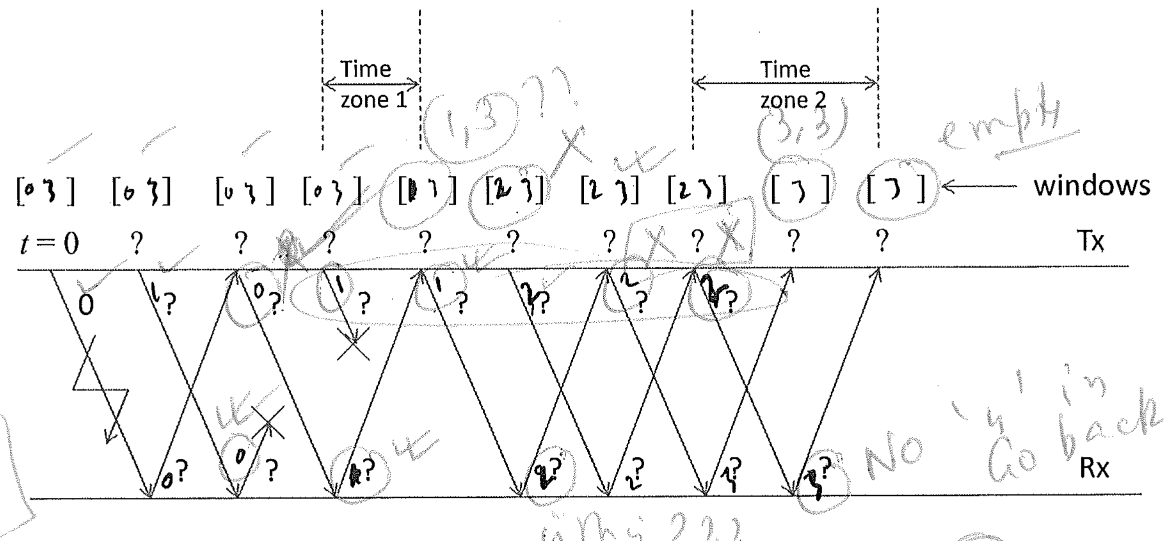
$= 0.071$ s $\leftarrow$ Node A

$$T_{BC} = 2(1 \times 10^6 \times 5 \times 10^{-9}) + \frac{1000 + 100}{80487.8} = 0.0237 \text{ s}$$

setting the timer at 0.024 seconds

on node B

5. **[this problem is worth 25 points total]** A (tx, rx) pair is using Go-back N protocol at the DLC layer with $N = 4$. The transmitter has only four data frames to send to the receiver, starting with frame number 0. The figure below shows certain events. The lightning bolt symbol denotes a frame error and the cross symbol denotes a lost frame. ACK's are cumulative and implicit.



-10

(a) Fill in the data frame numbers (i.e., from tx to rx). [5 points] -1

(b) Fill in the ACK numbers (i.e., from rx to tx) [5 points] -0.5

(c) The timeline starts from $t = 0$, as shown. Fill in the missing time instant values at the transmitter (indicated by '?') assuming that each data frame is 1 Kb and the link data rate is 2 Kbps. [5 points] -5

(d) For each of the 't' values, write down the window at the transmitter (indicated by [ ]) [5 points] -1

(e) Explain what's going on at the tx in time zones 1 and 2. Why is the tx behaving as it is during these time zones? [5 points]

-2.5

Time Zone 1, the data frame 1 was lost, the resent again and received correctly.

Time Zone 2, frame 2 was receive and receiver keeps asking for fram 3

no, it's timing out

6. **[this problem is worth 40 points total]** Suppose node A sends a message which is split up into five frames to node B, where each frame is five **bytes** long. Assume that the probability that the channel flips any **bit**, **independently of others**, is 0.01.

(a) Compute the probability that any frame is received correctly by node B. [5 points]

$[1-(1-0.01)^5]^5 = 2.8276 \times 10^{-7}$ prop ~~of all frame in error~~

$1-(2.8276\times 10^{-7}) = 0.99999$ any frame recieved correctly

(b) Compute the probability that the message is received correctly by node B. [5 points]

$(1-0.01)^5]^5 = 0.7778$

(c) What is the probability that the first three bits in the first frame sent by node A are in error? [5 points]

$(0.01)^3$

-30

(d) What is the probability that the first three bits in the last frame sent by node A are in error? [2.5 points]

$(0.01)^3$

(e) What is the probability that any three bits in the first frame sent by node A are in error? [5 points]

$\binom{25}{3}(1-0.01)^4(0.01)^3 = 0.00184$

(f) What is the probability that any three bits in the last frame sent by node A are in error? [2.5 points]

$\binom{25}{3}(1-0.01)^2(0.01)^3 = 0.0000098$

There's no 0.49 no here which you used next pg

(g) Suppose node B has no error correction capability. For even a single bit error, it requests a retransmit from node A. Based on your answer in part (b), how many frames do you think node A will need to transmit, **on an average**, before the message gets through correctly? [5 points]

$$\frac{1}{0.7779}$$

X incorrect, but conceptually O.K.

(h) Find the probability that the first two frames sent by node A are in error. [5 points]

$$0.049^2$$

X where did you get 0.049 from??

(i) Find the probability that any two frames sent by node A are in error. [5 points]

$$\binom{5}{2}(0.049)^2(1-0.049)^3 = 0.0207$$

X ??