HKUSTx: ELEC1200.3x A System View of Communications: From Signals to Packets (Part 3)

Bookmarks

- Pre-course Materials
- Topic 1: Course Overview
- **Topic 2: The Link Layer**

2.1 Link Layer
Week 1 Quiz due Jan 25, 2016 at 15:30 UTC

2.2 Multiple Access Protocols
Week 1 Quiz due Jan 25, 2016 at 15:30 UTC

2.3 Aloha Protocol
Week 1 Quiz due Jan 25, 2016 at 15:30 UTC

2.4 Efficiency of Slotted Aloha
Week 1 Quiz due Jan 25, 2016 at 15:30 UTC

Topic 2: The Link Layer > 2.5 Lab 1: Link Layer > LAB 1 - TASK 3

Bookmark

LAB 1 - TASK 3 (EXTERNAL RESOURCE) (1.0 points possible)

2.5 Lab 1: Link Layer
Lab due Jan 25, 2016 at 15:30 UTC

MATLAB download and tutorials

LAB 1 - TASK 3

In this task, you will learn how to check the correctness of received frame.

INSTRUCTIONS

The MATLAB code in the below window is similar to the code described in Task 1 where we simulat performance of the slotted ALOHA protocol. The difference here is that we do not use the function **checkReceivedFrame**, but implement it in details. Also, instead of generating the frames on the f some pre-generated frames to the variable **"frame"** to simplfy simulation and checking the results

In this task, we detect errors in a given frame. Consider a small network where **n_users**=4 users sl same channel to access a receiving node, which relays their datagrams the rest of the network. On has been received by the receiving node, it can check three things: the preamble, the user id, and checksum. For this simulation, we set the preamble to be **[1 0 1 0]**. The user id of received frame s valid value, within 1 and **n_users**=4 inclusive. To check the id we use the function **id = bin2num(w** convert a word of four binary numbers into a decimal number. Finally, we need to check the correc message using the checksum.

To do that, we can divide the frame into four blocks of four bits each. Then, we can compute the ch these four blocks in the same way as described in the previous task (attention! Now we consider 4 instead of 3). If the frame is correct the computed checksum should consist of four zeros. For exam receive the frame:

```
[ 1 0 1 0 ] [ 0 0 0 1 ] [ 1 0 1 0 ] [ 0 0 0 1]
```

We can divide it into the four words:

```
[ 1 0 1 0 ]
[ 0 0 0 1 ]
[ 1 0 1 0 ]
[ 0 0 0 1 ]
```

and the obtained the checksum is **[0 0 0 0]**.

In the initial MATLAB code, we load the variable **frames**, which contains the list of frames that we w Then it loops over all the frames and your task is to validate the inspected frame (variable **frame**), the preamble, the **id** and the checksum. Set the variable **cs_ok** to 1 if the checksum is correct and otherwise. Do the same with the variables **pre_ok**, **id_ok** and **frame_ok** by checking the preamble frame, respectively. Please, revise the code between the lines

```
% % % % Revise the following code   % % % %
```

and

```
% % % % Do not change the code below % % % %
```

Do not change other parts of the code and do not use the function **checkReceivedFrame**. Note t "cs_ok", "pre_ok", "id_ok" and "frame_ok" to indicate whether the checksum, the preamble, the use whole frame is valid. Do not rename them.

Your Solution

Reset MATLAB Documentation (https://www.mathworks.

```

    % check the preamble
    %disp(frame(16:13));
    rx_preamble = frame(1:4);
    %disp(rx_preamble);
    pre_ok = isequal(rx_preamble, [1 0 1 0]);

    %check if the id is between 1 and n users, both inclusive
```

Assessment Tests: Passed

✓ **Is problem setup unmodified?**

✓ Have the messages been checked correctly?

Output

```
Correct preamble: 493/800
Correct id:       557/800
Correct checksum: 542/800
Correct frames: 472/800
```