

**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 2



Bookmark

LAB 1 - TASK 2 (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 2

In this task, you will learn how to create the frame that is used at the link level to send the datagram

INSTRUCTIONS

The MATLAB code in the below window is similar to the code described in task 1 where we simulate performance of the slotted ALOHA protocol. The only difference in the code is that here we do not function **createFrame** to generate the frame structure but implement the function in detail.

Here, we consider a modified (simplified) frame structure, which consists four blocks, each with four total frame length is 16 bits. The four blocks contain the preamble, the node ID, the datagram, and checksum. An example of the frame is shown as follows:

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

The preamble, which is "1010" in this example, is a fixed sequence of bits utilized to indicate the start of a frame. The node ID, which is "0001" in this example, is the binary representation of the user's **id**. Here **id** is 1, indicating that the frame is from the first user. The conversion of the user id from decimal to binary is achieved by using the function **num2bin**. Given that the frame structure only uses four bits to represent the node ID, we can simulate a system with a maximum of 16 nodes. The third block, "1010", is the datagram of which is provided by the function **getNewDatagram**.

The final block contains the checksum bits, which are "0001" in this example. The checksum bits are used to check whether errors occurred during the transmission (recall the channel coding schemes we learned in task 1). In this simulation, the checksum bits are generated as the even parity bits for the first three blocks. Specifically, we first divide the first 12 bits in the frame into 4 groups, each with 3 bits, as shown by below:

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

Then, the 4 checksum bits are computed by performing the "exclusive or" operation over all three blocks.

Then, the 4 checksum bits are computed by performing the exclusive or operation over all three 1s in each group (one column in the above array). Equivalently, we can obtain the checksum bits by binary exclusive or of the bits in the first three (leftmost) columns are 0. The exclusive or of the bits in the fourth column is 1. As a result, the checksum bits for the above example are [0 0 0 1]. Another way to compute the checksum is counting the number of 1s in the words and set the bit of the checksum to 1 if this number is odd and to 0 otherwise. For example, for the leftmost bits we have two ones, which is even, so the checksum bit is 0.

Your task is to create the frame for the datagram and store the result inside the variable **frame**. In order to obtain the binary representation of the user's **id**, you can use the function **num2bin(id,4)**, where the second argument indicates the length of the binary representation. Set the preamble to be **[1 0 1 0]** for this task. 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 **createFrame**.

Your Solution

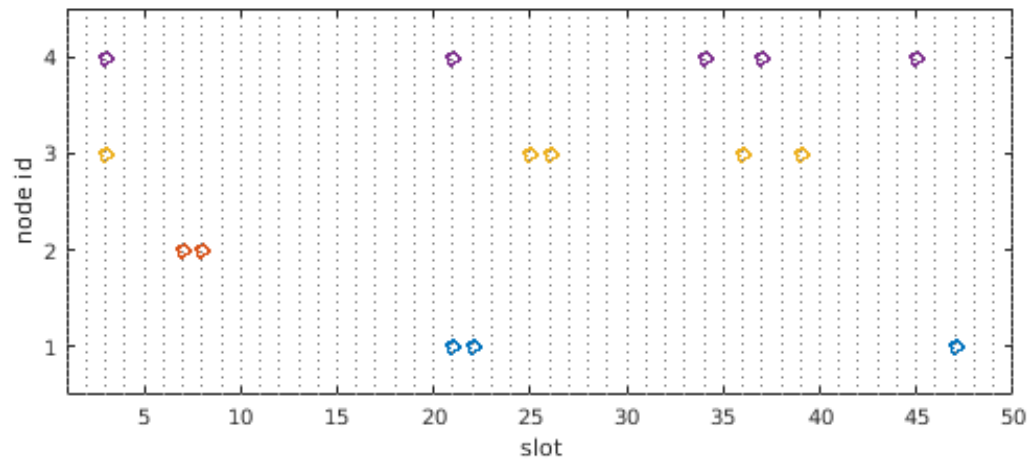
 Reset

 MATLAB Documentation (<https://www.mathworks.com>)

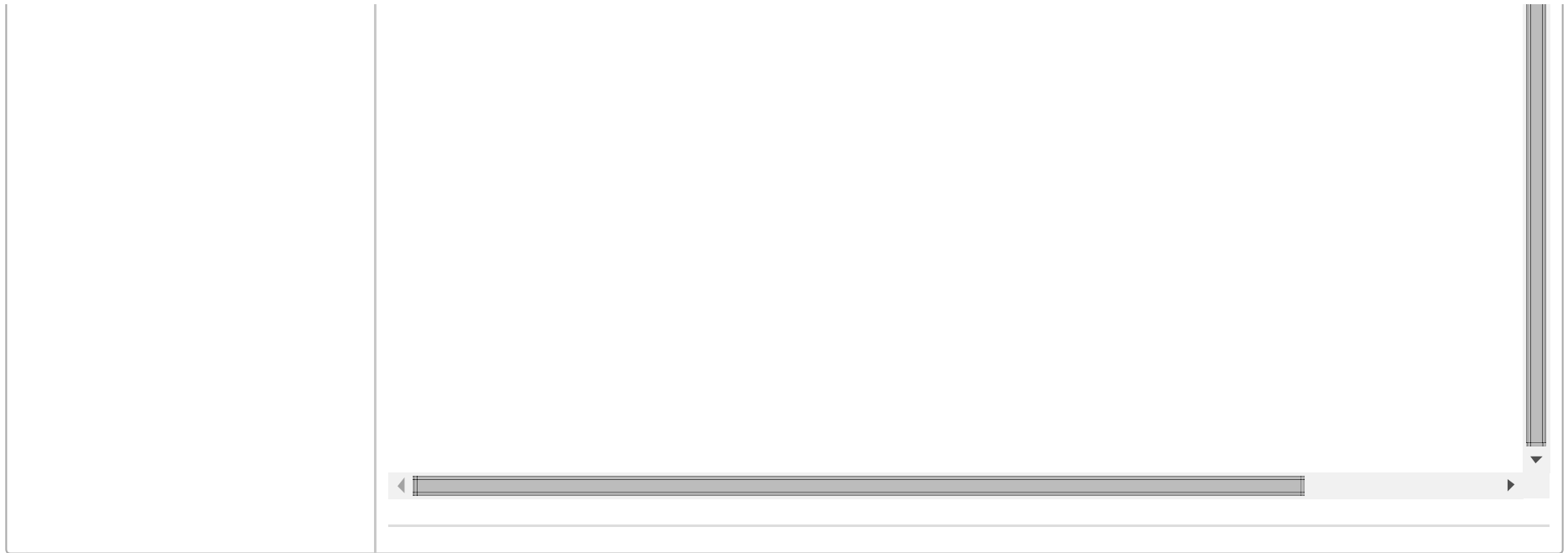
```
1 rng(0)
2
3 % parameters
4 n_slots = 1000; %
5 p = 0.1;
6 n_users = 4;
7
8
```

Output

Total number of slots: 1000
Empty slots: 664
Collisions: 335
Frame transmitted successfully: 1



Assessment Tests



© All Rights Reserved



© edX Inc. All rights reserved except where noted. EdX, Open edX and the edX and Open EdX logos are registered trademarks or trademarks of edX Inc.

POWERED BY
OPENedX



