Green University of Bangladesh

Department of Computer Science and Engineering (CSE)

Faculty of Sciences and Engineering

Semester: (Spring, Year: 2021), B.Sc. in CSE (Day)

Lab Report No: 04

Course Title: Data Communication Lab

Course Code: CSE 308 **Section:** PC-DD

Lab Experiment Name: Implementing AMI for bit sequence 11.

Student Details

Name	Id
Md. Romzan Alom	201902144

Lab Date: 19.08. 2022

Submission Date: 27.08.2022

Course Teacher's Name: Md. Nazmus Shakib

[For Teachers use only: Don't Write Anything inside this box]

Lab Report Status	
Marks: ……………………………………	Signature:....................
Comments:..	Date:..............................

Title of the Lab Experiment: Implementing AMI for bit sequence 11.

Objectives / Aim:

We learn about AMI from this experiment. We can take user input as a string and we can perform various operations on this input and show the result as signal output.

Introduction:

In telecommunication and data storage, Bipolar is a line coding scheme in which there are three voltage levels positive, negative, and zero. The voltage level for one data element is at zero, while the voltage level for the other element alternates between positive and negative. Bipolar line coding is two types AMI and Pseudotenary.

• Alternate Mark Inversion (AMI) A neutral zero voltage represents binary 0. Binary 1's are represented by alternating positive and negative voltages.

• Pseudoternary Bit 1 is encoded as a zero voltage and the bit 0 is encoded as alternating positive and negative voltages i.e., opposite of AMI scheme.

Problem:

By implementing AMI for bit sequence 11.

Problem analysis:

In this lab we will only implement Alternate Mark Inversion (AMI) because concept of Pseudoternary is revers of AMI line coding. Where, 11 are positive voltage level and negative voltage and single 1 and 0 mean no voltage level.

Code:

```
bits = [0 0 1 0 1 1 0 1 1 0];
bitrate = 1;
n = 1000;
T = length(bits)/bitrate;
N = n*length(bits);
dt = T/N;
t = 0:dt:T;
x = zeros(1,length(t));
lastbit = 1;
j=2;
for i=1:length(bits)
  if bits(i)==1 && bits(j) == 1
    x((i-1)*n+1:i*n) = lastbit;
    lastbit = lastbit;
   i=j;
    x((i-1)*n+1:i*n) = -lastbit;
  end
  j=j+1;

end

plot(t, x, 'Linewidth', 3);
```

Output:

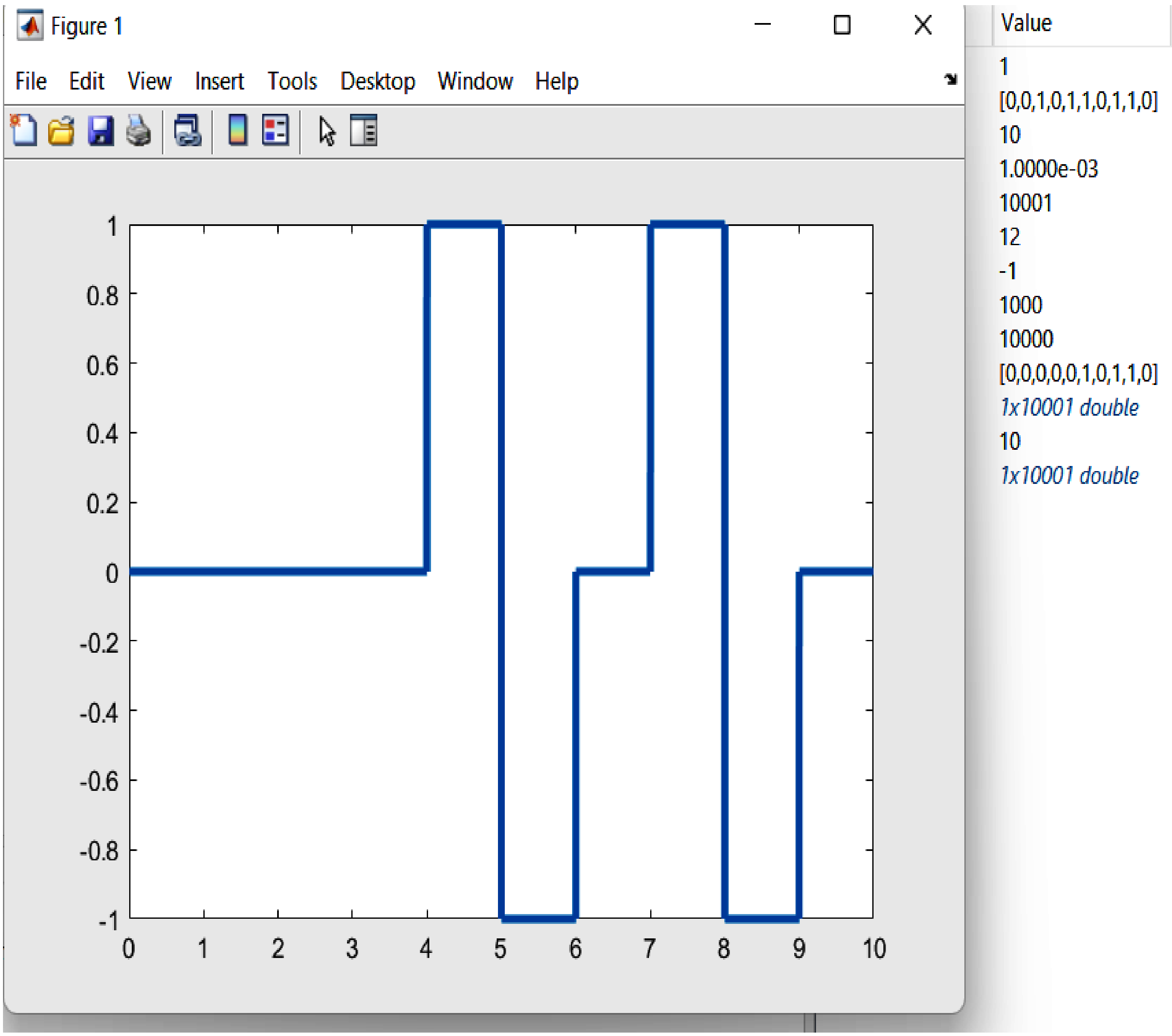
Figure 1
File Edit View Insert Tools Desktop Window Help
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0 1 2 3 4 5 6 7 8 9 10
Value
1
[0,0,1,0,1,1,0,1,1,0]
10
1.0000e-03
10001
12
-1
1000
10000
[0,0,0,0,0,1,0,1,1,0]
1x10001 double
10
1x10001 double

Analysis and Discussion:

1. From this lab, we knew AMI. And the working system of AMI. From this experiment, we saw some signal across the input. Those are very important for analysis signal.

2. This lab is completely based on software. So it may have some Software and Mechanical errors.

3. From this problem, we calculated in bit. When we operate with it, we facing some problem to understanding it.

4. In our lab, we use only one bit to AMI but lab report has two bit that’s why we face some problem. Cause of unknown software system.

5. Based on the focused objective(s) to understand about the AMI, the additional lab exercise made me more confident towards the fulfillment of the objectives(s).

6. Compile error.