AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)

FACULTY OF SCIENCE & TECHNOLOGY

Spring 2022-2023

Section: J, Group:6

LAB REPORT: 08

Supervised By: SADMAN SHAHRIAR ALAM

Date of Submission: 03.04.2023

Submitted by: **Abdullah Al Mahadi(ID:21-42044-1), Group 6**

Title: Study of Digital-to-Digital conversion (Line coding) using MATLAB

Abstract:

The objectives of this experiment aimed at achieving two main goals. The first goal is to gain an understanding of how MATLAB can be used to solve communication engineering problems. The second goal is to develop an understanding of Digital-to-Digital Conversion, specifically Line Coding, using MATLAB.

Apparatus:

MATLAB2016a

Performance Task:

My ID: 20-42044-1 = AB-CDEFG-H

So,
E = 0 in binary = 0 0 0 0

F = 4 in binary = 0 1 0 0

G = 4 in binary = 0 1 0 0

So, the binary stream = 0 0 0 0 0 1 0 0 0 1 0 0

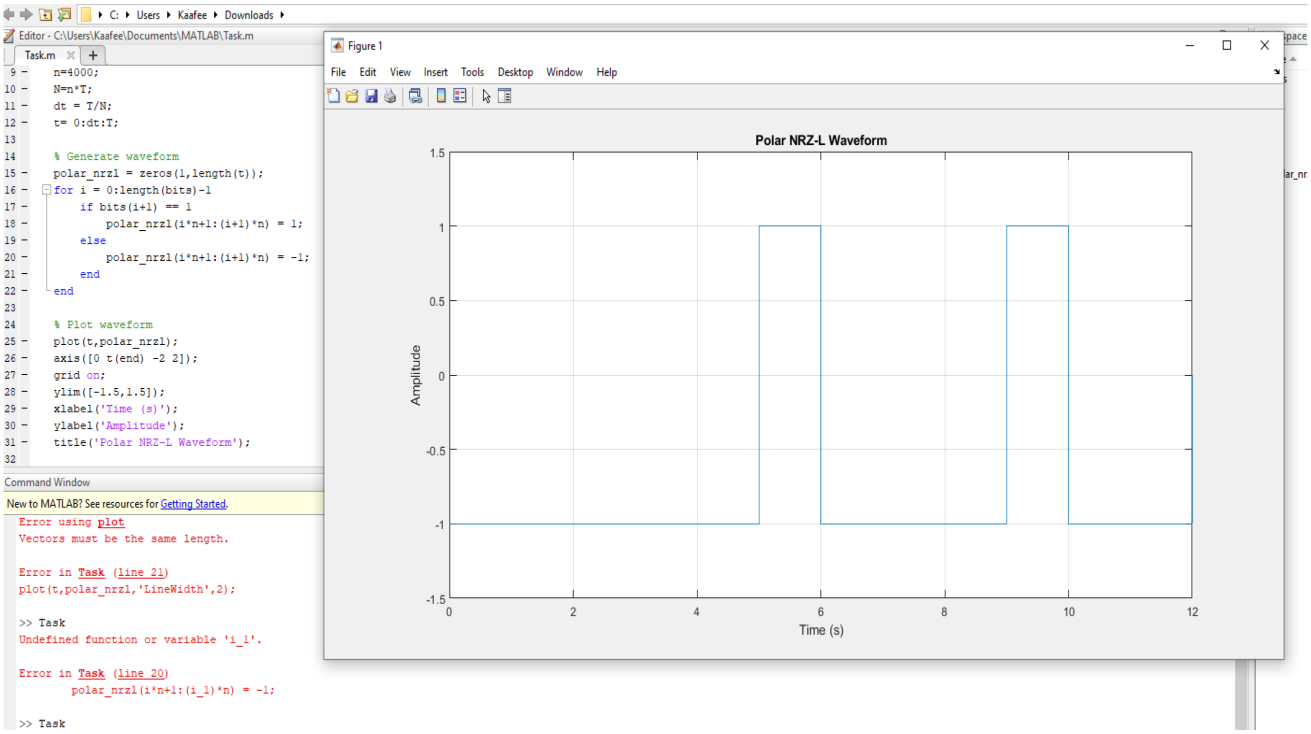
Task 1: Polar NRZ – L assuming bit rate is 4 kbps

Code:

```
% Polar NRZ-L waveform generation

% Bit rate = 4 kbps


% Define bit sequence

bits = [0 0 0 0 0 1 0 0 0 1 0 0];
```

```
% Set the time interval for each bit
T = length(bits);
n=4000;
N=n*T;
dt = T/N;
t= 0:dt:T;

% Generate waveform
polar_nrzl = zeros(1,length(t));
for i = 0:length(bits)-1
    if bits(i+1) == 1
        polar_nrzl(i*n+1:(i+1)*n) = 1;
    else
        polar_nrzl(i*n+1:(i+1)*n) = -1;
    end
end

% Plot waveform
plot(t,polar_nrzl);
axis([0 t(end) -2 2]);
grid on;
ylim([-1.5,1.5]);
xlabel('Time (s)');
ylabel('Amplitude');
title('Polar NRZ-L Waveform');
```

Simulation:



Task 2: Manchester assuming bit rate is 2 kbps

Code:

% Manchester waveform generation

% Bit rate = 2 kbps

% Define bit sequence

bits = [0 0 0 0 0 1 0 0 0 1 0 0];

% Set the time interval for each bit

T =length(bits);

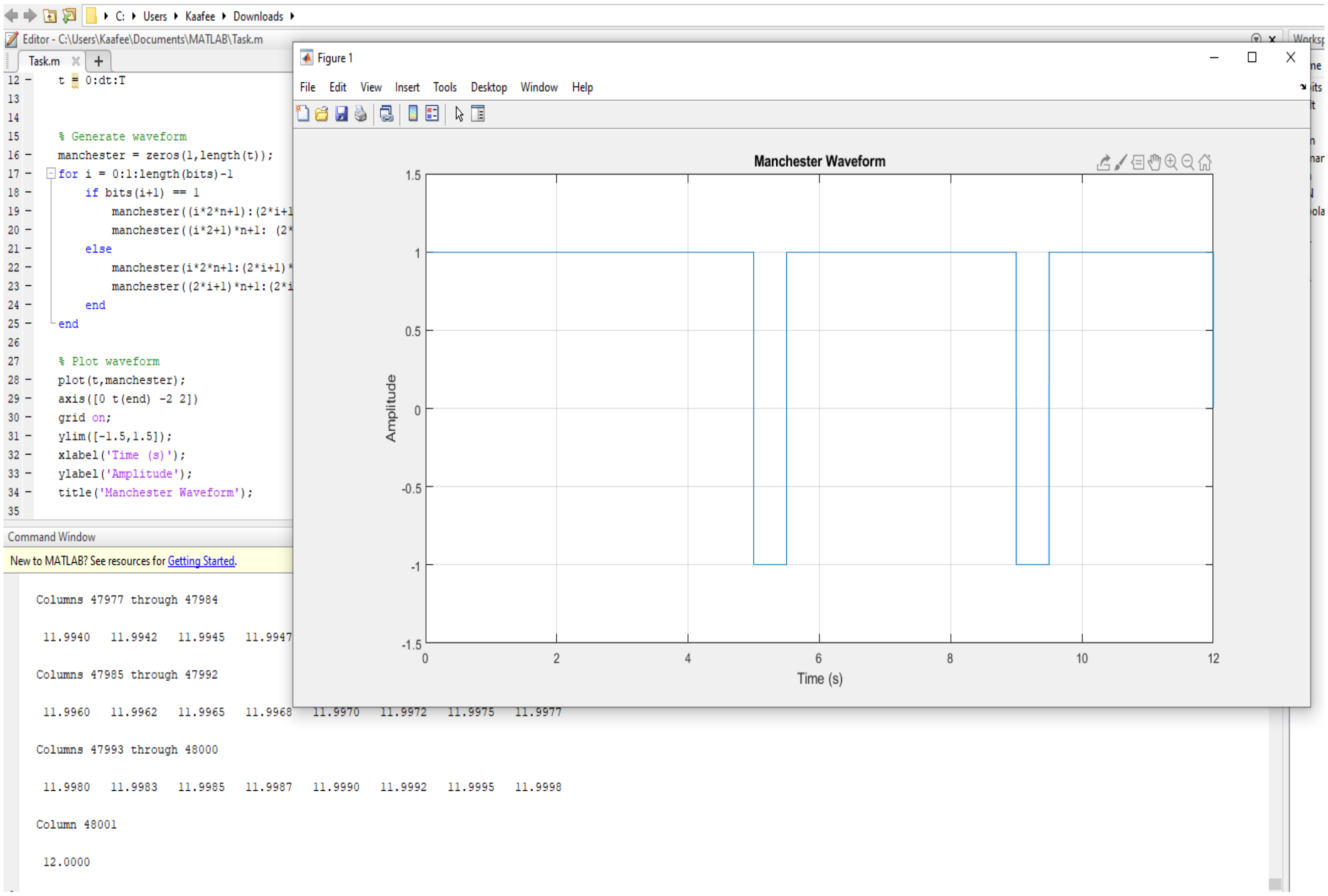
n = 2000;

N = 2*n*T

dt = T/N

```
t = 0:dt:T


% Generate waveform
manchester = zeros(1,length(t));
for i = 0:1:length(bits)-1
    if bits(i+1) == 1
        manchester((i*2*n+1):(2*i+1)*n) = -1;
        manchester((i*2+1)*n+1: (2*i+2)*n) = 1;
    else
        manchester(i*2*n+1:(2*i+1)*n) = 1;
        manchester((2*i+1)*n+1:(2*i+2)*n) = 1;
    end
end

% Plot waveform
plot(t,manchester);
axis([0 t(end) -2 2])
grid on;
ylim([-1.5,1.5]);
xlabel('Time (s)');
ylabel('Amplitude');
title('Manchester Waveform');
```

Simulation:



Task 3: AMI assuming bit rate is 5 kbps

Code:

% AMI waveform generation

% Bit rate = 5 kbps

% Define bit sequence

bits = [0 0 0 0 0 1 0 0 0 1 0 0];

% Set the time interval for each bit

T = length(bits);

```
n = 5000;
N= n*T;
dt = T/N;
pulse = -1;
t = 0:dt:T;

% Generate waveform
ami = zeros(1,length(t));

for i = 0:length(bits)-1
    if bits(i+1) == 1
        if pulse == -1
            ami(i*n+1: (i+1)*n) = -1;
        else
            pulse = 1;
            ami(i*n+1: (i+1)*n) = 1;
        end
    else
        ami(i*n+1:(i+1)*n) = 0;
        pulse = pulse*-1;
    end

end

% Plot waveform
plot(t,ami);
axis([0 t(end) -2 2]);
grid on;
```

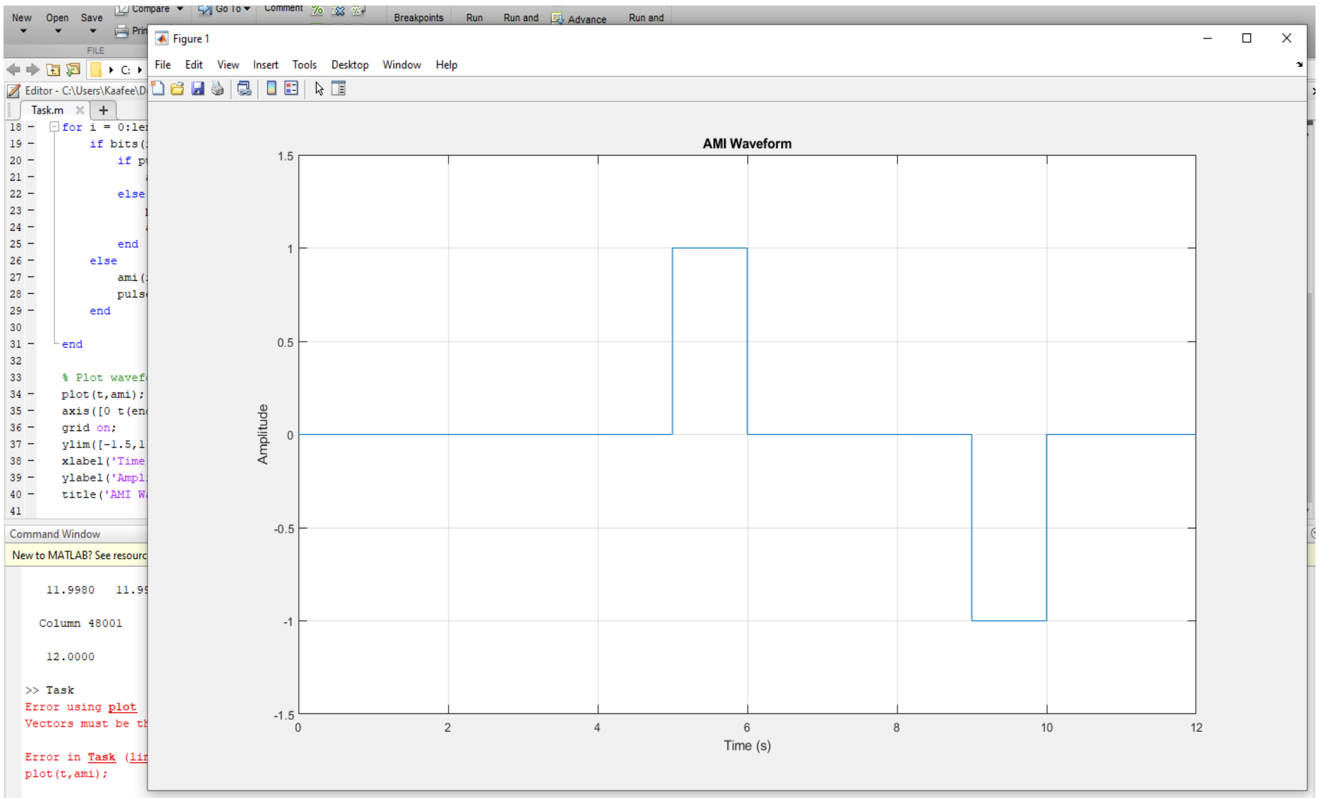
ylim([-1.5,1.5]);

xlabel('Time (s)');

ylabel('Amplitude');

title('AMI Waveform');

Simulation:



Task 4: MLT-3 assuming bit rate is 10 kbps

Code:

% MLT-3 waveform generation

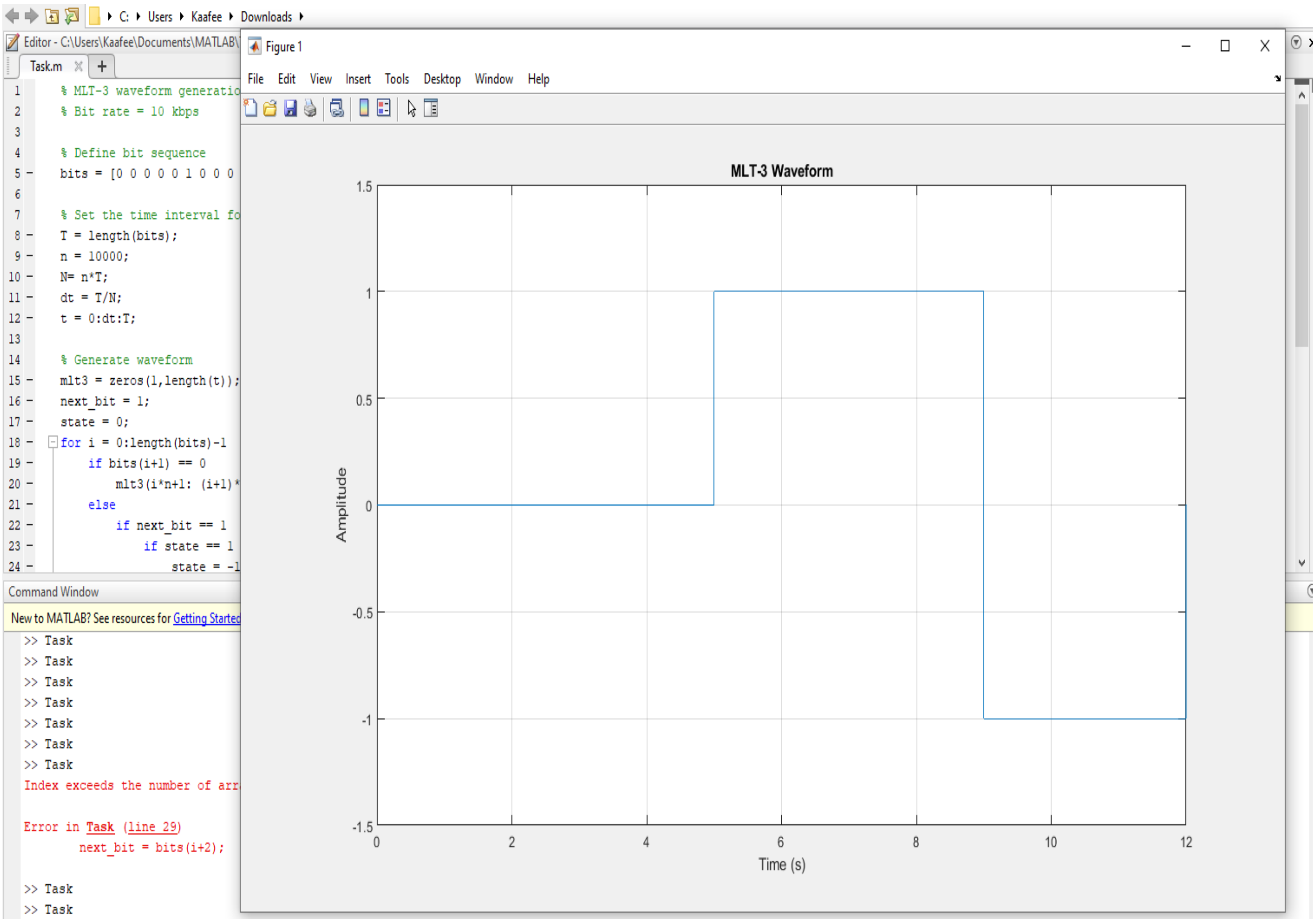
% Bit rate = 10 kbps

% Define bit sequence

```
bits = [0 0 0 0 0 1 0 0 0 1 0 0];

% Set the time interval for each bit
T = length(bits);
n = 10000;
N= n*T;
dt = T/N;
t = 0:dt:T;

% Generate waveform
mlt3 = zeros(1,length(t));
next_bit = 1;
state = 0;
for i = 0:length(bits)-1
    if bits(i+1) == 0
        mlt3(i*n+1: (i+1)*n) = state;
    else
        if next_bit == 1
            if state == 1
                state = -1;
            else
                state = 1;
            end
        end
        next_bit = bits(i+1);
        mlt3(i*n+1:(i+1)*n) = state;
```

```
    end
end

% Plot waveform
plot(t,mlt3);
axis([0 t(end) -2 2]);
grid on;
ylim([-1.5,1.5]);
xlabel('Time (s)');
ylabel('Amplitude');
title('MLT-3 Waveform');
```

Simulation:



Discussion:

This experiment aims to achieve two main goals. The first objective is to understand the application of MATLAB in solving communication engineering problems. MATLAB is a powerful tool that is widely used in engineering and scientific applications. It provides a user-friendly platform that enables the user to perform complex computations and simulations with ease. In this experiment, students will learn how to use MATLAB to solve communication engineering problems, which is an essential skill for any engineering student.

The second objective of the experiment is to develop an understanding of Digital-to-Digital Conversion, specifically Line Coding, using MATLAB. Digital-to-Digital Conversion is a crucial aspect of communication engineering, and Line Coding is one of the techniques used for this purpose. Line Coding involves the conversion of digital data into a format suitable for transmission over a communication channel. By understanding the principles of Line Coding and how to implement it using MATLAB, students will develop a deeper understanding of Digital-to-Digital Conversion.

Overall, this experiment is an excellent opportunity for students to learn valuable skills in communication engineering and gain hands-on experience using MATLAB. By achieving these objectives, students will be better prepared for future coursework and careers in the field of engineering.

Conclusion:

In conclusion, the experiment discussed in the abstract provides an excellent opportunity for students to gain valuable skills in communication engineering and MATLAB. By achieving the two main objectives of the experiment, students will be better equipped to solve communication engineering problems and implement Digital-to-Digital Conversion techniques using Line Coding. This knowledge is essential for any engineering student and will be beneficial for future coursework and career opportunities. Overall, this experiment is a practical and effective way to teach essential skills in the field of communication engineering.

REFERENCE:

1. MATLAB user guide.

2. Prof. Dr.-Ing. Andreas Czylwik, “MATLAB for Communications”