2020F_ESE 3014_1

SEMESTER: 3^{rd}

INSTRUCTOR: Linchen Wang

LAB: 1

DUE DATE: 1 Oct 2020 - 23:55

SUBMITTED DATE: 25 Sept 2020- 20:00

STUDENTS: Amandeep Singh (C0765434)

Lab-1

INTRODUCTION

This lab is meant to deal with basic Amplitude modulation using an input and a carrier wave, apart from that this lab is focused on the application of Matlab/Octave and how to perform AM on these softwares.

DESCRIPTION

For the purpose of this lab I'll be using Matlab R2019a, which was just released last year. A few of the predefined functions used for this lab are explained below.

- **zeros () :** Zeros (x, y) here x and y are used to represent the size of the matrix where x stands for rows and y stands for columns & as the name suggests zeros() is used to create a matrix of zeros.

- **randi (max, x, y):** Here max stands for the maximum limit of a random number that can be generated for eg: if max is 10 then all the random elements generated are going to be less than or equal to 10, x and y are used to represents rows and columns of the matrix.

- **plot (x, y):** Plot simply plots the graph between the x and y coordinates provided.

- **ammod (x, fc, fs, phase, carmag):** Ammod() is used to return an amplitude modulated signal where 'x' stands for the input signal, 'fc' is the carrier frequency, 'fs' is the sampling frequency, 'phase' specifies the initial phase of the modulated signal and 'carmag' is the magnitude of the carrier wave.

- **awgn (x,SNR):** Here awgn() is used to introduce white gaussian noise to any signal, also here x stands for the input signal and SNR stands signal to noise ratio.

QUESTION & ANSWERS:

Note: everything written with ">>" is a matlab command eg: **>>plot(x,t);** & the outputs are included in the appendix.

Q1. Create a 5x1 vector of zeros. Create a 2x5 matrix of random numbers.

a.

>>zeros(5,1);

```
ans =

    0
    0
    0
    0
    0
```

b.

```
>>randi(10,2,5);
ans =

    8    6    9    6    2
    3    7   10    2    3
```

Q2. Multiply a column of a matrix with an element of this same matrix

>>b=randi(10,2,5)

```
b =

    9    9   10    2    7
    3    3    4    3    5
```

>>b(2,3)

```
ans =
    4
```

>>b(2,3)*b

```
ans =

   36   36   40    8   28
   12   12   16   12   20
```

Q3. Create a plot of the sin function between 0 and 6π.

```
>>t = [0:0.1:6*pi];
>>a = sin(t);
>>plot(t,a)
```

Q4. Simulate an amplitude modulation (AM) system with all input, carrier and output signals. Say the input signal is a cosine wave with amplitude as 2V and frequency as 1000Hz. The carrier signal is also a cosine wave with amplitude as 5V and frequency as 10KHz. The modulation degree is 0.5, and the initial phases of all cosine wave are 0. (Recall Nyquist sampling theorem to avoid distortion i.e. under sampling).

```
>>f = 1000;
>>t = (0:1/f:100)';
>>fc = 10000;
>>fs = 44100;
>>x = 2*cos(2*pi*t);
>>y = 0.5 * ammod(x,fc,fs,0,5);
>>plot(y)
```

Q5. Use the signals above, consider an actual vivid simulation mode, and add random noise in output signal. In this simulation, we divide time domain into several duration, and call each duration as frame. The scanning cycle of an oscilloscope is equal to frame period, that means each time we simulate a frame of signal, and the display will be refreshed once. Therefore, we can get a constantly sliding input signal, a carrier signal with phase jitter, and output signal with noise.

```
>>f = 1000;
>>t = (0:1/f:100)';
>>fc = 10000;
>>fs = 44100;
>>x = 2*cos(2*pi*t);
>>y = 0.5 * ammod(x,fc,fs,0,5);  %till here everything is taken from 'Q4'

>>noise= awgn(y,0);              %now we add the noise
>>subplot(2,1,1)
>>plot (y,'green');
>>subplot(2,1,2)
>>plot (noise,'red');
>>plot(t,[y,x])
```

CONCLUSION

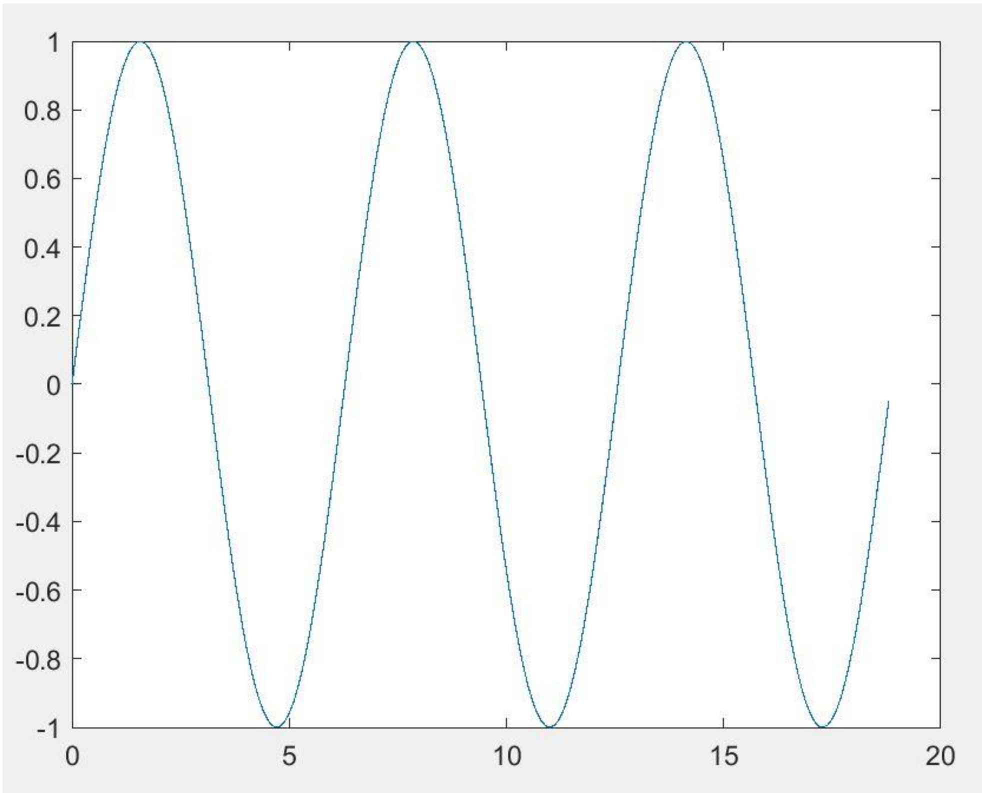
- Amplitude modulation is quite helpful when we want to transfer a signal from one spot to another as it’s multiplied with a carrier wave so in case of transmission if there is a loss of data the input signal can survive it.
- Matlab is very useful tool when it comes to visualizing signals and performing simple matrix calculation although the minimum system requirements mentioned on the website are bare minimum, according to me the minimum requirement should anything above an i3 8th gen with 8GB RAM and atleast 50GB SSD storage space.

APPENDIX

- Outputs for Q3, Q4 & Q5 are as follows:

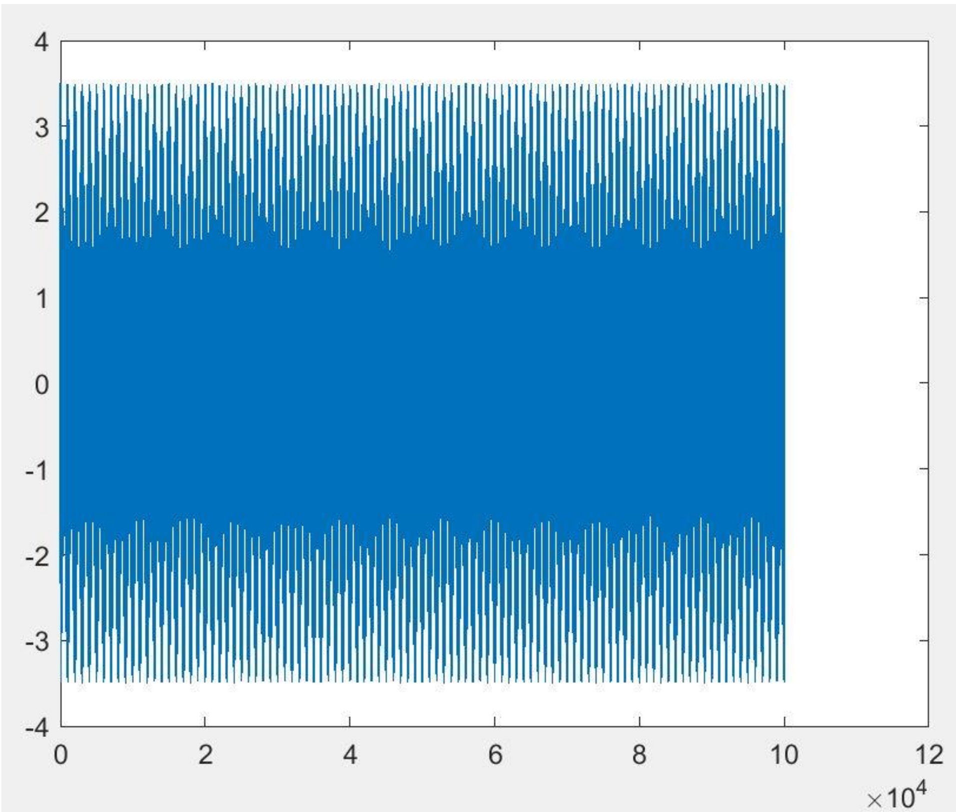
Q3.

```
>>t = [0:0.1:6*pi];
>>a = sin(t);
>>plot(t,a)
```



Q4.

```
>>f = 1000;
>>t = (0:1/f:100)';
>>fc = 10000;
>>fs = 44100;
>>x = 2*cos(2*pi*t);
>>y = 0.5 * ammod(x,fc,fs,0,5);
>>plot(y)
```



Q5.

```
>>f = 1000;
>>t = (0:1/f:100)';
>>fc = 10000;
>>fs = 44100;
>>x = 2*cos(2*pi*t);
>>y = 0.5 * ammod(x,fc,fs,0,5);  %till here everything is taken from 'Q4'

>>noise= awgn(y,0);              %now we add the noise
>>subplot(2,1,1)
>>plot (y,'green');              %fig-1 & represents the input signal(green) without noise
>>subplot(2,1,2)
>>plot (noise,'red');            %fig-1 & represents the noise(red)
>>plot(t,[y,x])                  %fig-2 & represents the input signal + noise
```

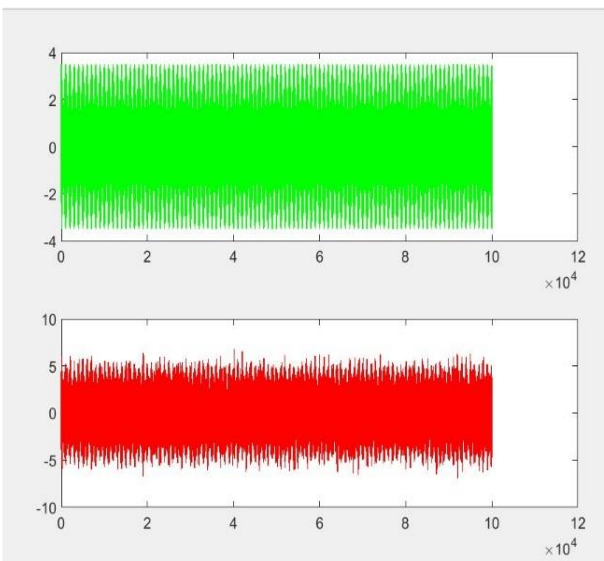


Fig-1

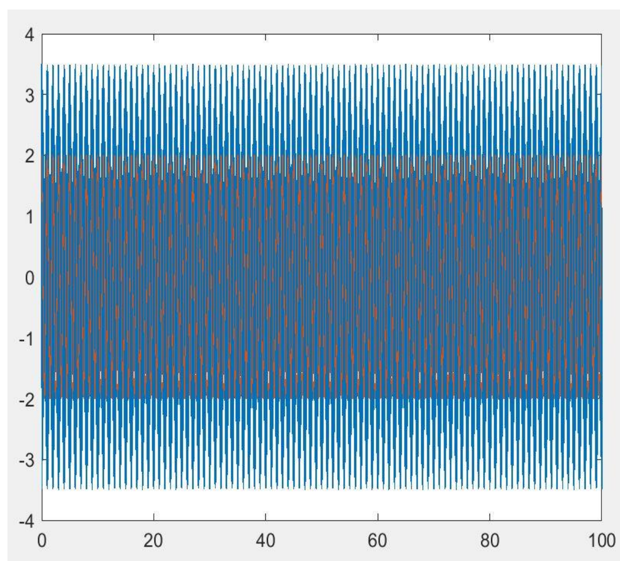


Fig-2