电子科技大学 格拉斯哥 学院

UOG-UESTC Joint School of UESTC

标 准 实 验 报 告

Lab Report

（实验）课程名称： 信号与系统

(LAB) Course Name： Signals and Systems

电子科技大学教务处制表

UoG-UESTC Joint School

Lab-3 Report

Student Name：蓝心悦
Student No. :2017200601018
Instructor：许渤
Location：
Date：5/14/2019

一、 Laboratory name：

Signals and Systems

二、 Project name：

Representing and studying about the properties and applications of FS, FT and sampling using MATLAB

三、 Lab hours：4

四、 Theoretical background：

1. The basic concepts and properties of Fourier series, Fourier transform, sampling and partial fraction expansion of a signal. In this lab, I learnt how to represent, manipulate and analyze the properties of Fourier series, Fourier transform and sample of a signal in MATLAB.
2. Some basic MATLAB commands for representing the FS, FT and sampling including fft, ifft, fftshift, residue, dash, dot, linspace, and freqs.

五、 Objective：

1. Perform the Fourier serises using MATLAB
2. Perform the Fourier transform using MATLAB. Plot the spectrum of signals.
3. Further understand the properties of Fourier transform.
4. Apply Fourier transform in modulation.

六、 Description：

1. Compute the Discrete-time frequency response with freqz. 3.2
2. Eigen functions of Discrete-Time LTI Systems. 3.4(a)(b)(c).
3. Synthesizing Signals with the Discrete-Time Fourier Series. 3.5(d)(e)(f)(h)
4. Properties of the Continuous-Time Fourier Transform. 4.3(a)-(f)
5. Time- and Frequency-Domain Characterizations of Systems. 4.4(a)-(f).
6. Perform Partial Fraction Expansion. 4.5(b)
7. Amplitude Modulation with Morse code. 4.6(a)(b)(c)(d)

七、 Required instruments：

Computer, MATLAB

八、 Procedures, Analysis of Lab data & result and Conclusion：

```
%3.2(a)(b) I used the function of last lab session, and also
%used the freqz function of textbook to complete this tutorial
a1=[1 -0.8 0];
b1=[2 0 -1];
[H1 omega1]=freqz(b1,a1,4);
H1
omega1
%(c)
a1=[1,0.8 0];
b1=[2,0,-1];
[H2 omega2]=freqz(b1,a1,4,'whole')
```

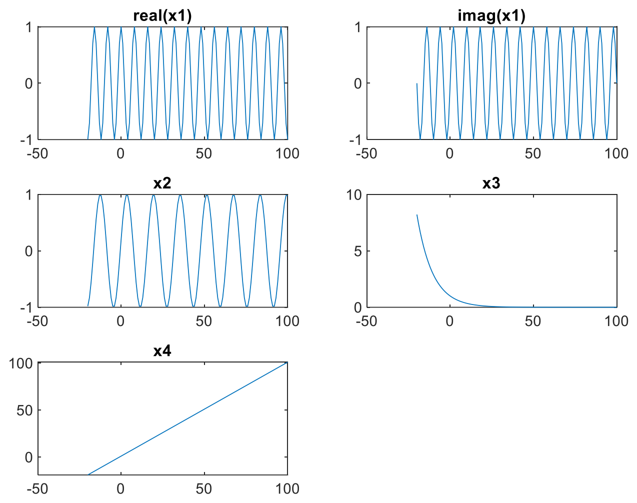
```
H1 =

   5.0000 + 0.0000i
   2.8200 - 1.3705i
   1.8293 - 1.4634i
   0.9258 - 0.9732i


omega1 =

         0
    0.7854
    1.5708
    2.3562


H2 =

   0.5556 + 0.0000i
   1.8293 + 1.4634i
   5.0000 + 0.0000i
   1.8293 - 1.4634i


omega2 =

         0
    1.5708
    3.1416
    4.7124
```

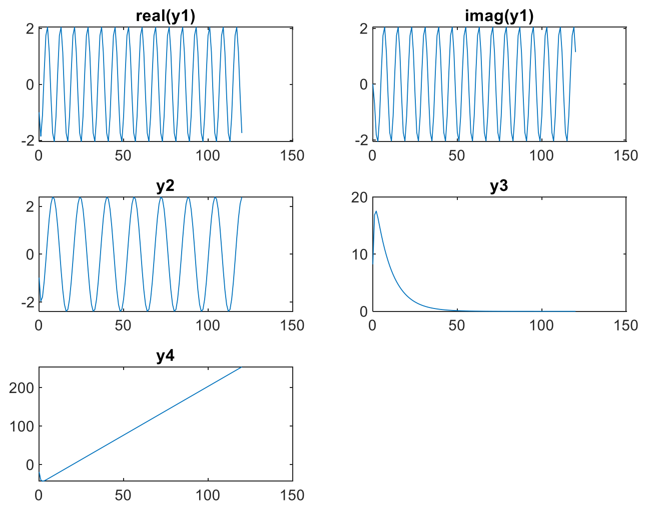
.

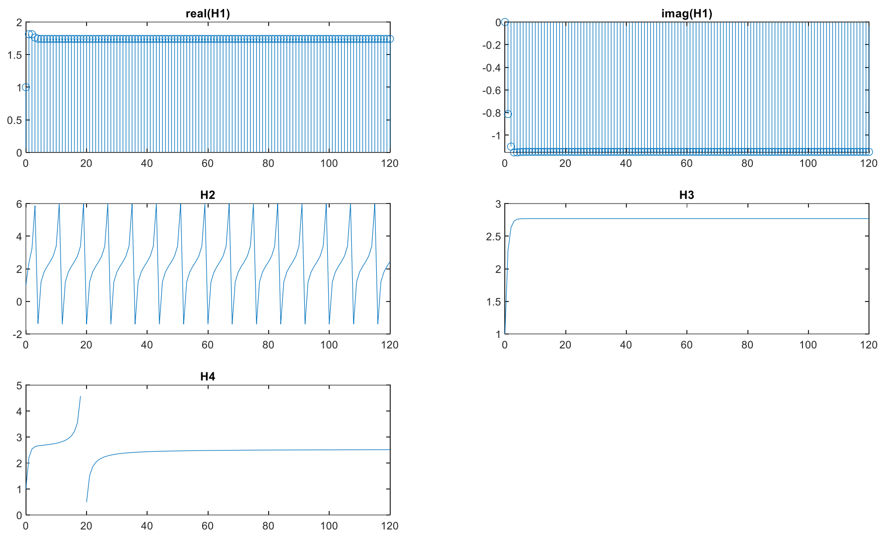
```
%3.4 this exercise examines the eigenfunction property of
%discrete-time LTI systems. When the input sequence is a
```

```
%complex exponential, the output is the same complex
%exponential only scaled in amplitude by a complex constant
n=[-20:100];
x1=exp(j*pi*n/4);
x2=sin(pi*n/8+pi/16);
x3=(9/10).^n;
x4=n+1;
subplot(3,2,1)
plot(n,real(x1))
title('real(x1)')
subplot(3,2,2)
plot(n,imag(x1))
title('imag(x1)')
subplot(3,2,3)
plot(n,x2)
title('x2')
subplot(3,2,4)
plot(n,x3);
title('x3')
subplot(3,2,5)
plot(n,x4)
title('x4')
pause

a=[1 -0.25];
b=[1 0.9];
y1=filter(b,a,x1);
y2=filter(b,a,x2);
y3=filter(b,a,x3);
y4=filter(b,a,x4);
n=[0:120];
subplot(3,2,1)
plot(n,real(y1))
title('real(y1)')
subplot(3,2,2)
plot(n,imag(y1))
title('imag(y1)')
subplot(3,2,3)
plot(n,y2)
title('y2')
subplot(3,2,4)
plot(n,y3)
title('y3')
subplot(3,2,5)
plot(n,y4)
title('y4')
pause

H1=y1./x1
H2=y2./x2
H3=y3./x3
H4=y4./x4
subplot(3,2,1)
stem(n,real(H1))
title('real(H1)')
subplot(3,2,2)
```

```
stem(n,imag(H1))
title('imag(H1)')
subplot(3,2,3)
plot(n,H2)
title('H2')
subplot(3,2,4)
plot(n,H3)
title('H3')
subplot(3,2,5)
plot(n,H4)
title('H4')
```

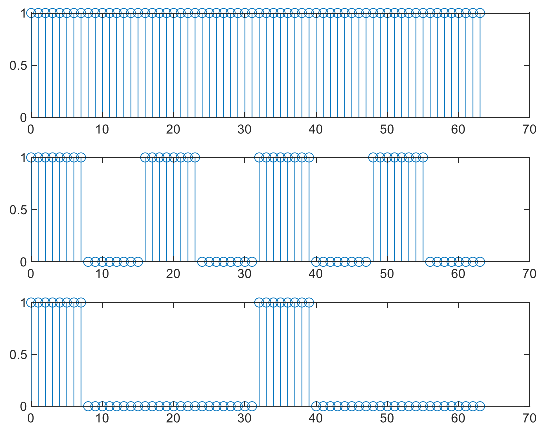


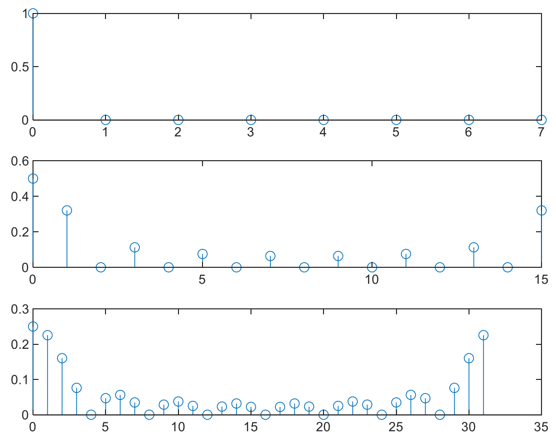


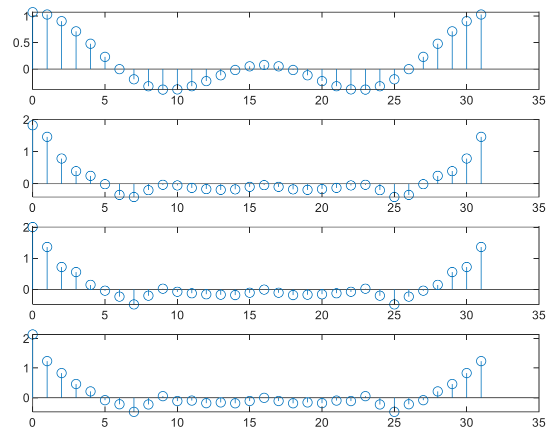


```
%3.5(d)I analyzed a set of periodic discrete-time signals to
%obtain the DTS coefficients and constructed one of these
%signals by adding in a few coefficients at a time
n=[0:63];
x1=ones(1,8);
x2=[ones(1,8) zeros(1,8)];
x3=[ones(1,8) zeros(1,24)];
subplot(3,1,1)
stem(n,[x1 x1 x1 x1 x1 x1 x1 x1])
subplot(3,1,2)
stem(n,[x2 x2 x2 x2])
subplot(3,1,3)
stem(n,[x3 x3])
pause

a1=(1/8)*fft(x1)
a2=(1/16)*fft(x2)
a3=(1/32)*fft(x3)
subplot(3,1,1)
stem([0:7],abs(a1))
subplot(3,1,2)
stem([0:15],abs(a2))
subplot(3,1,3)
stem([0:31],abs(a3))
pause
%3.5(e)
a1=(1/8)*fft(x1);
a2=(1/16)*fft(x2);
a3=(1/32)*fft(x3);
subplot(3,1,1)
stem(0:7,abs(a1))
subplot(3,1,2)
stem(0:15,abs(a2))
subplot(3,1,3)
stem(0:31,abs(a3))
pause
```

```
%3.5(f)&(h)
n=0:31;
x3=[ones(1,8),zeros(1,24)];
a3=(1/32)*fft(x3);
a3m=abs(a3);
x3_2=1/8+2*a3m(1)*cos(pi/16*n)+2*a3m(2)*cos(2*pi/16*n);
subplot(4,1,1);
stem(n,x3_2);
x3_8=1/8+2*a3m(1)*cos(pi/16*n)+2*a3m(2)*cos(2*pi/16*n)+2*a3m(3
)*cos(3*pi/16*n)+2*a3m(4)*cos(4*pi/16*n)+2*a3m(5)*cos(5*pi/16*
n)+2*a3m(6)*cos(6*pi/16*n)+2*a3m(7)*cos(7*pi/16*n)+2*a3m(8)*co
s(8*pi/16*n);
subplot(4,1,2);
stem(n,x3_8);
x3_12=1/8+2*a3m(1)*cos(pi/16*n)+2*a3m(2)*cos(2*pi/16*n)+2*a3m(
3)*cos(3*pi/16*n)+2*a3m(4)*cos(4*pi/16*n)+2*a3m(5)*cos(5*pi/16
*n)+2*a3m(6)*cos(6*pi/16*n)+2*a3m(7)*cos(7*pi/16*n)+2*a3m(8)*c
os(8*pi/16*n)+2*a3m(9)*cos(9*pi/16*n)+2*a3m(10)*cos(10*pi/16*n
)+2*a3m(11)*cos(11*pi/16*n)+2*a3m(12)*cos(12*pi/16*n);
subplot(4,1,3);
stem(n,x3_12);
x3_all=1/8+2*a3m(1)*cos(pi/16*n)+2*a3m(2)*cos(2*pi/16*n)+2*a3m
(3)*cos(3*pi/16*n)+2*a3m(4)*cos(4*pi/16*n)+2*a3m(5)*cos(5*pi/1
6*n)+2*a3m(6)*cos(6*pi/16*n)+2*a3m(7)*cos(7*pi/16*n)+2*a3m(8)*
cos(8*pi/16*n)+2*a3m(9)*cos(9*pi/16*n)+2*a3m(10)*cos(10*pi/16*
n)+2*a3m(11)*cos(11*pi/16*n)+2*a3m(12)*cos(12*pi/16*n)+2*a3m(1
3)*cos(13*pi/16*n)+2*a3m(14)*cos(14*pi/16*n)+2*a3m(15)*cos(15*
pi/16*n)+a3m(16)*exp(i*pi*n);
subplot(4,1,4);
stem(n,x3_all);
```

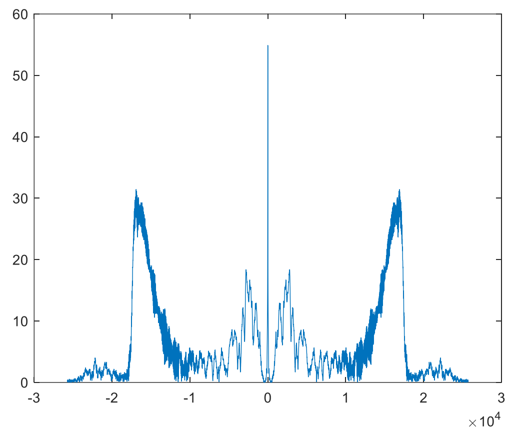






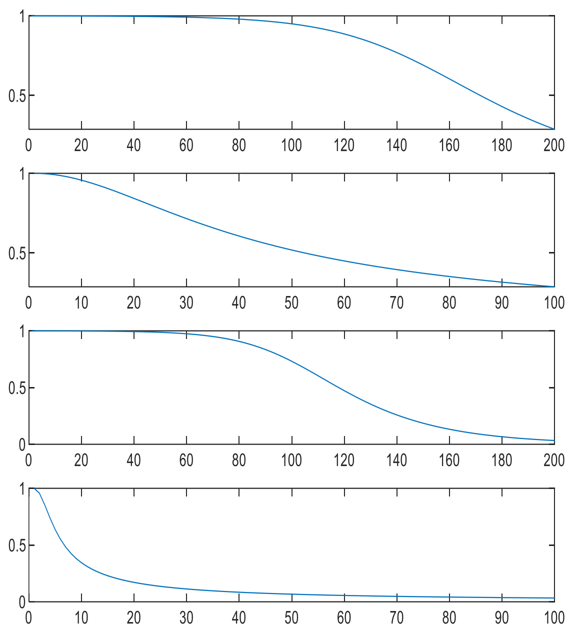
```
%4.3
load splat
y=y(1:8192);
N=8192;
fs=8192;
Y=fftshift(fft(y));
w=[-pi:2*pi/N:pi-pi/N]*fs;
plot(w,abs(Y))
pause

Y1=conj(Y);
y1=ifft(fftshift(Y1));
y1=real(y1);
plot(w,abs(y1))
pause
sound(y1,fs)
pause

Y2=abs(Y);
y2=ifft(fftshift(Y2));
y2=real(y2)
plot(w,abs(y2))
pause
sound(y2,fs)
pause
```

```
Y3=exp(i.*angle(Y))
y3=ifft(fftshift(Y3));
plot(w,abs(y3))
pause
sound(y3,fs)
```

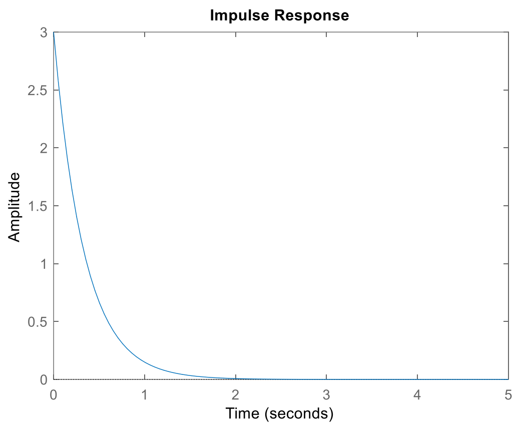


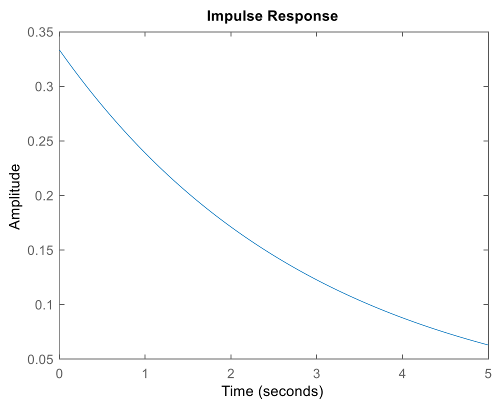
(b) the voices are time-inverse to each other.
(c) As y(t) is real, its magnitude and phase are both real.
(f) The phase is more crucial .

```
%4.4(a)(b)
w=linspace(0,10);
%system1
a1=[1 3];
b1=3;
subplot(4,1,1)
H1_1=freqs(b1,a1);
plot(abs(H1_1))
subplot(4,1,2)
H1_2=freqs(b1,a1,w);
plot(abs(H1_2))
%system2
a2=[1 1/3];
b2=1/3;
subplot(4,1,3)
H2_1=freqs(b2,a2);
plot(abs(H2_1))
subplot(4,1,4)
H2_2=freqs(b2,a2,w);
plot(abs(H2_2))
```

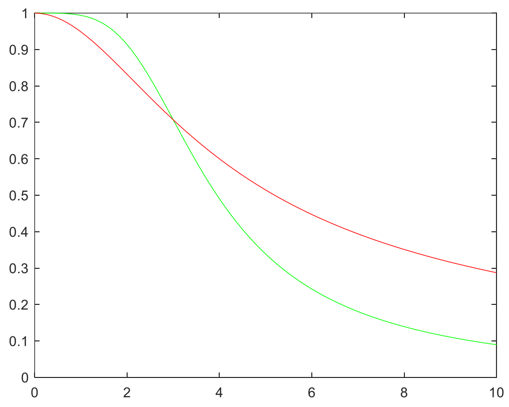


```
%4.4(c)
a1=[1 3];
b1=3;
a2=[1 1/3];
b2=1/3;
t=linspace(0,5);
impulse(b1,a1,t)
pause
impulse(b2,a2,t)
```





```
%4.4(e)(f)
wc=3;
w=linspace(0,10);
[b2,a2]=butter(2,wc,'s');
H2=freqs(b2,a2,w);
plot(w,abs(H2),'g')
hold on
a1=[1 3];
b1=3;
H1=freqs(b1,a1,w)
plot(w,abs(H1),'r')
```

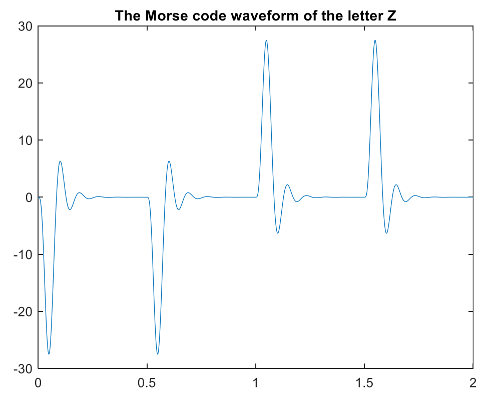


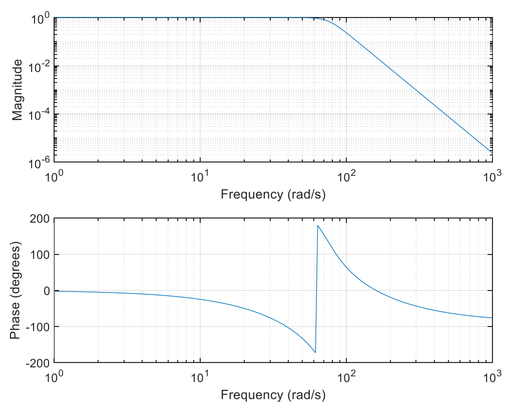
(b)agree
(d)They are similar to each other. **Symmetry of scaling in time and frequency domain.**

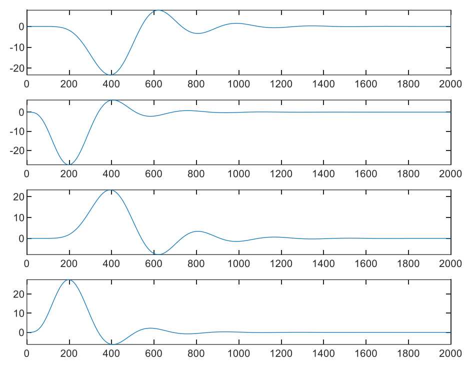
```
%4.5(b)
a1=[1 1.5 0.5];
b1=[0 1 -2];
[r1,p1]=residue(b1,a1)
```

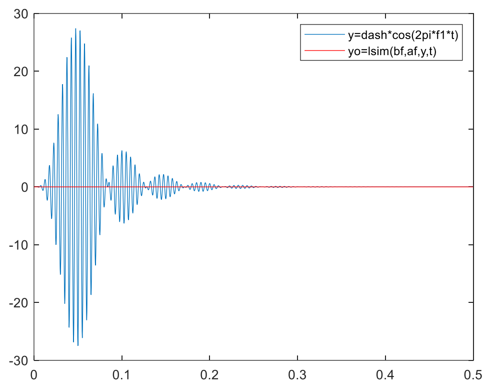
```
r1 =

     6
    -5


p1 =

   -1.0000
   -0.5000
```

```
%4.6(a)
load ctftmod.mat
z=[dash dash dot dot];
plot(t,z);
title('The Morse code waveform of the letter Z')
pause
%4.6(b)
freqs(bf,af)
pause
%4.6(c)
load ctftmod.mat
ydash=lsim(bf,af,dash,t(1:length(dash)));
ydot=lsim(bf,af,dot,t(1:length(dot)));
subplot(4,1,1)
plot(ydash)
subplot(4,1,2)
plot(dash)
subplot(4,1,3)
plot(ydot)
subplot(4,1,4)
plot(dot)
pause
%4.6(d)
figure
y=dash.*cos(2*pi*f1*t(1:length(dash)));
y0=lsim(bf,af,y,t(1:length(dash)));
plot(t(1:length(dash)),y,t(1:length(dash)),y0,'r');
legend('y=dash*cos(2pi*f1*t)','yo=lsim(bf,af,y,t)')
```









```
(d) yes.
```

九、 Summary and comments:

From this lab, we studied the eigen function properties and synthesis of discrete time signals. Then the performance of CTFT is proved by audio signal. The relationship between the impulse response and the frequency response is discussed. Function residue is useful for calculating the impulse response of complex system functions.

十、 Suggestion for this lab:

I suggest to provide the solutions for the questions in the lab session after submission, so that we can correct our code and have further comprehension of knowledge.

Score:

Instructor: