



北京理工大学  
Beijing Institute of Technology

## 本科实验报告

实验名称： 信号与信息处理实验 I

课程名称：	信号与信息处理实验 I	实验时间：	10/19 10/26 11/2 11/9
任课教师：	范哲意	实验地点：	理 B-404
实验教师：	何冰松, 林艳飞	实验类型：	<input checked="" type="checkbox"/> 原理验证 <input type="checkbox"/> 综合设计 <input type="checkbox"/> 自主创新
学生姓名：	施念		
学号/班级：	05011609/1120161302	组 号：	
学 院：	信息与电子学院	同组搭档：	
专 业：	电子信息类	成 绩：	



信息与电子学院

SCHOOL OF INFORMATION AND ELECTRONICS

目录

实验 1 信号的时域描述与运算 .....1

    一 实验目的 .....3

    二 实验原理 .....3

    三 实验内容 .....6

    四 收获和体会 ..... 19

实验 2 LTI 系统的时域分析 ..... 20

    一 实验目的 ..... 20

    二 实验原理 ..... 20

    三 实验内容 ..... 22

    四 收获和体会 ..... 33

实验 3 信号的频域分析 ..... 34

    一 实验目的 ..... 34

    二 实验原理 ..... 34

    三 实验内容 ..... 37

    四 收获和体会 ..... 47

实验 4 LTI 系统的频域分析 ..... 48

    一 实验目的 ..... 48

    二 实验原理 ..... 48

    三 实验内容 ..... 50

    四 收获和体会 ..... 58

实验 5 连续系统的复频域分析 ..... 59

    一 实验目的 ..... 59

    二 实验原理 ..... 59

    三 实验内容 ..... 61

    四 收获和体会 ..... 67

实验 6 离散系统的 Z 域分析 ..... 68

    一 实验目的 ..... 68

    二 实验原理 ..... 68

    三 实验内容 ..... 69

    四 收获和体会 ..... 75

# 实验 1 信号的时域描述与运算

## 一 实验目的

1. 掌握信号的 MATLAB 表示及其可视化方法。
2. 掌握信号基本时域运算的 MATLAB 实现方法。
3. 利用 MATLAB 分析常用信号，加深对信号时域特性的理解。

## 二 实验原理

### 1. 连续时间信号的 MATLAB 表示

连续时间信号指的是在连续时间范围内有定义的信号，即除了若干个不连续点外，在任何时刻信号都有定义。在 MATLAB 中连续时间信号可以用两种方法来表示，即向量表示法和符号对象表示法。

从严格意义上来说，MATLAB 并不能处理连续时间信号，在 MATLAB 中连续时间信号是用等时间间隔采样后的采样值来近似表示的，当采样间隔足够小时，这些采样值就可以很好地近似表示出连续时间信号，这种表示方法称为向量表示法。表示一个连续时间信号需要使用两个向量，其中一个向量用于表示信号的时间范围，另一个向量表示连续时间信号在该时间范围内的采样值。

例如一个正弦信号可以表示如下：

```
>> t=0:0.01:10;  
>> x=sin(t);
```

利用 `plot(t,x)` 命令可以绘制上述信号的时域波形，如图 1 所示。如果连续时间信号可以用表达式来描述，则还可以采用符号表达式来表示信号。例如对于上述正弦信号，可以用符号对象表示如下：

```
>> x=sin(t);  
>> ezplot(X);
```

利用 `ezplot(x)` 命令可以绘制上述信号的时域波形

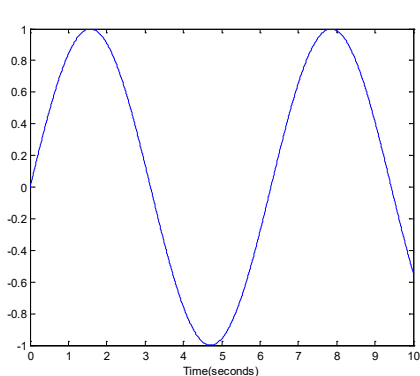


图1 利用向量表示连续时间信号

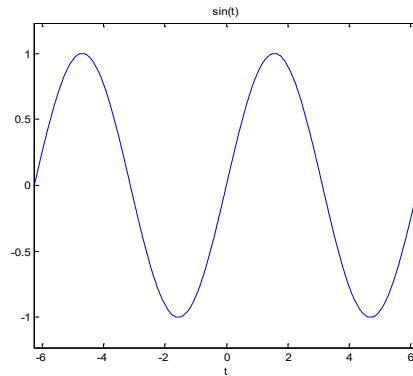


图2 利用符号对象表示连续时间信号

### 常用的信号产生函数

函数名	功能	函数名	功能
heaviside	单位阶跃函数	rectpuls	门函数
sin	正弦函数	tripuls	三角脉冲函数
cos	余弦函数	square	周期方波
sinc	sinc 函数	sawtooth	周期锯齿波或三角波
exp	指数函数		

## 2. 连续时间信号的时域运算

### 1) 相加和相乘

信号相加和相乘指两信号对应时刻的值相加和相乘，对于两个采用向量表示的可以直接使用算术运算的运算符“+”和“\*”来计算，此时要求表示两信号的向量时间范围和采样间隔相同。采用符号对象表示的两个信号，可以直接根据符号对象的运算规则运算。

### 2) 微分和积分

对于向量表示法表示的连续时间信号，可以通过数值计算的方法计算信号的微分和积分。这里微分使用差分来近似求取的，由时间向量 $[t_1, t_2, \dots, t_N]$ 和采样值向量

$[x_1, x_2, \dots, x_N]$ 表示的连续时间信号，其微分可以通过下式求得：

$$x'(t)|_{t=t_k} \approx \frac{x_{k+1} - x_k}{\Delta t}, k = 1, 2, \dots, N-1$$

连续时间信号的定积分可以由 MATLAB 的 quad 函数实现，调用格式为：

quad ('function\_name', a, b)

对于符号对象表示的连续时间信号，MATLAB 提供了 diff 函数和 quad 函数分别用于求微分和积分。

### 3. 离散时间信号的 MATLAB 表示

离散时间信号仅在一些离散时刻有定义。在 MATLAB 中离散时间信号需要使用两个向量来表示，其中一个向量用于表示离散的时间点，另一个向量表示在这些时间点上的值。例如对于如下时间信号

$$x(n) = \{-3, 2, -1, 2, 1, -1, 2, 3\}$$

采用 MATLAB 可以表示如下：

```
>> n=-3:4;  
>> x=[-3 2 -1 2 1 -1 2 3];  
>> stem(n,x,'filled');  
>> xlabel('n');  
>> title('x(n)');
```

Stem 函数用于绘制离散时间信号波形，为了与我们表示离散时间信号的习惯相同，在绘图时一般需要添加‘filled’选项，以绘制实心的杆状图形。上述命令绘制的信号时域波形如图 3 所示。

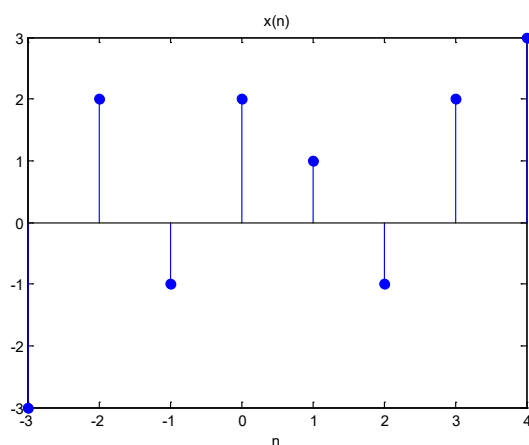


图3 离散时间信号示例

### 4. 离散时间信号的时域运算

离散时间信号的相加相乘是将两个信号对应的时间点上的值相加或相乘，可以直接使用算术运算的运算符“+”和“\*”来计算。

离散时间信号的位移，则可看作是将表示时间的向量平移，而表示对应时间点上的值的向量不变。

离散时间信号的反转，则可以看作是将表示时间的向量和表示对应时间点上的值的向量以零点为基准点，以纵轴为对称轴反折，向量的反折可以利用 MATLAB 的 flipr 函数实现。

### 三 实验内容

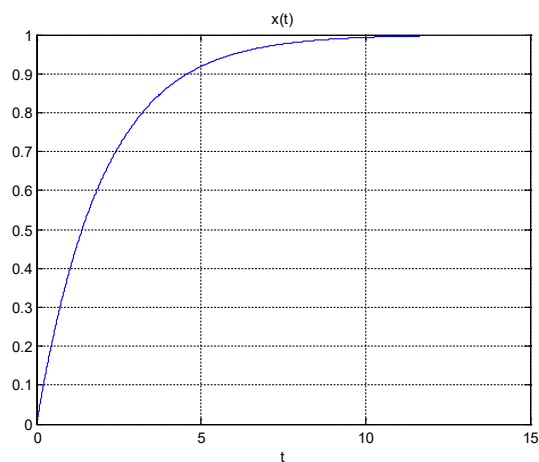
#### 1. 利用 MATLAB 绘制下列连续时间信号波形：

1.1  $x(t) = (1 - e^{-0.5t})u(t)$

代码：

```
clear;                                plot(t,x);
syms t;                               grid on;
t = 0:0.01:15;                        xlabel('t');
x = (1-exp(-0.5.*t)).*heaviside(t);   title('x(t)');
```

图像：

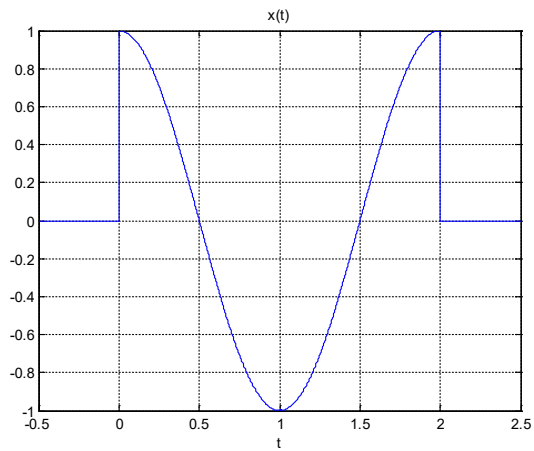


1.2  $x(t) = \cos(\pi t)[u(t) - u(t-2)]$

代码：

```
clear;                                plot(t,x);
syms t;                               grid on;
t = -0.5:0.002:2.5;                  xlabel('t');
%x = (1-exp(-0.5.*t)).*heaviside(t);  title('x(t)');
x = cos(pi.*t).*(heaviside(t)-heaviside(t-2));
```

图像：



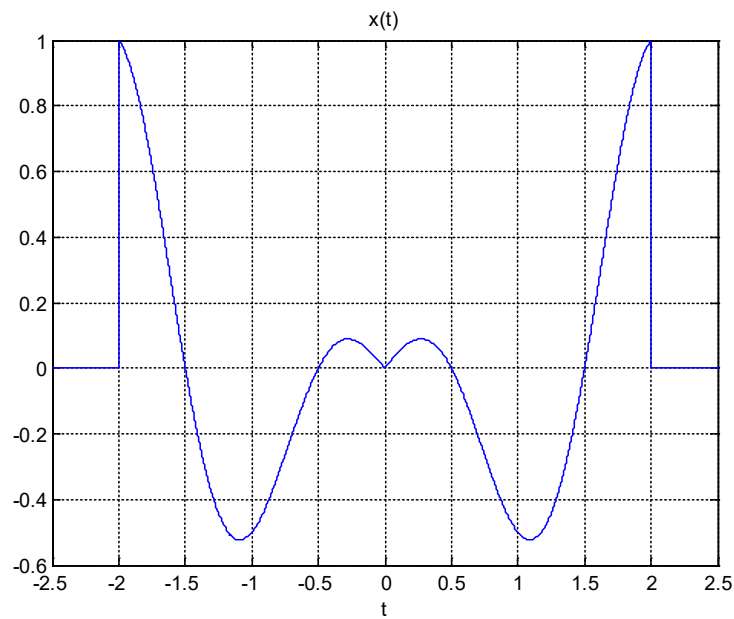
$$1.3 \quad x(t) = \frac{|t|}{2} \cos(\pi t) [u(t+2) - u(t-2)]$$

代码：

```
clear;
syms t;
t = -2.5:0.002:2.5;
%x = (1-exp(-0.5.*t)).*heaviside(t);
%x = cos(pi.*t).*(heaviside(t)-heaviside(t-2));
x = abs(t)/2.*cos(pi.*t).*(heaviside(t+2)-heaviside(t-2));

heaviside(t-2);
plot(t,x);
grid on;
xlabel('t');
title('x(t)');
```

图像：



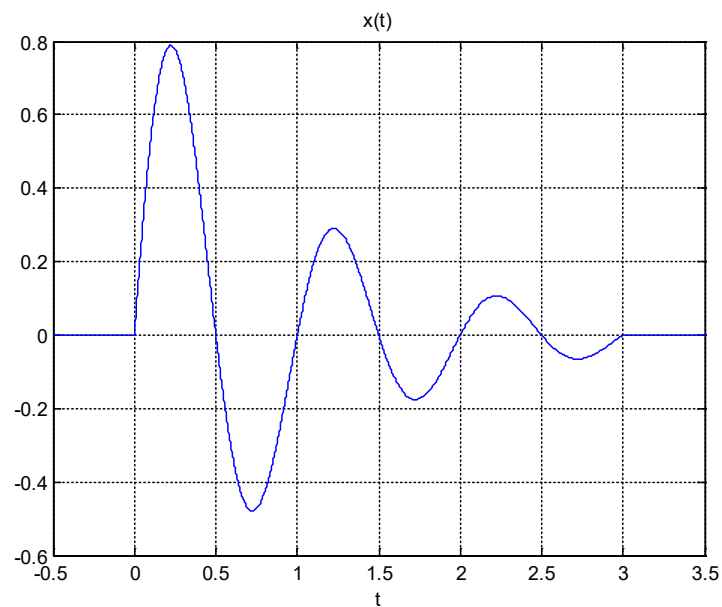
1.4  $x(t) = e^{-t} \sin(2\pi t)[u(t) - u(t-3)]$

代码：

```
clear;
syms t;
t = -0.5:0.002:3.5;
%x = (1-exp(-0.5*t)).*heaviside(t);
%x = cos(pi.*t).*(heaviside(t)-heaviside(t-2));
%x = abs(t)/2.*cos(pi.*t).*(heaviside(t+2)-heaviside(t-2));

x = exp(-t).*sin(2*pi.*t).*(heaviside(t)-heaviside(t-3));
plot(t,x);
grid on;
xlabel('t');
title('x(t)');
```

图像：



## 2. 利用 MATLAB 绘制下列离散时间信号波形：

2.1  $x(n) = u(n-3)$

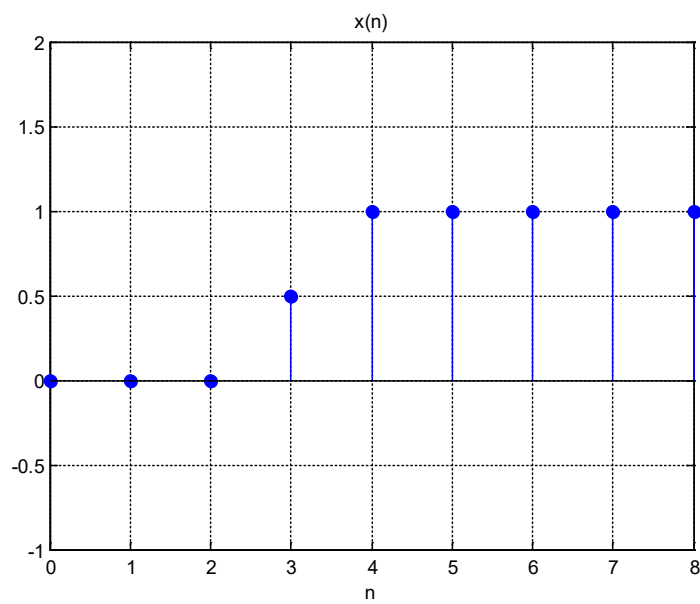
代码：

```
clear
syms x;
n = 0:8;
x = heaviside(n-3);
stem(n,x,'filled')

grid on;
axis([0,8,-1,2])
xlabel('n')
title('x(n)')
```



图像：



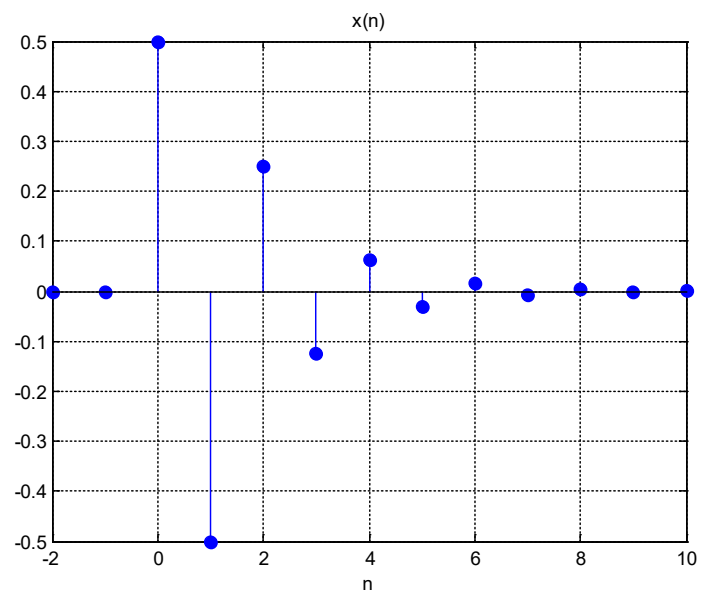
2.2  $x(n] = (-1/2)^n u(n]$

代码：

```
clear
syms x;
n = -2:10;
x = (-1/2).^n.*heaviside(n);
stem(n,x,'filled')

grid on;
axis([-2,10,-0.5,0.5])
xlabel('n')
title('x(n)')
```

图像：



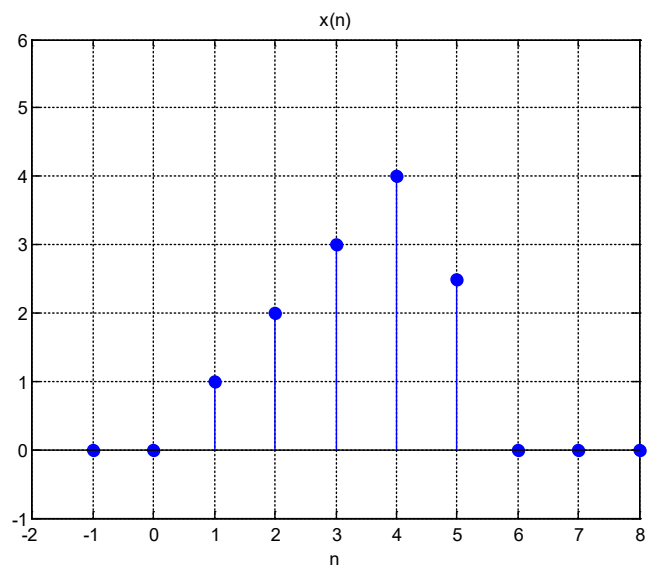
2.3  $x(n) = n[u(n) - u(n-5)]$

代码：

```
clear
syms x;
n = -2:8;
x = n.*(heaviside(n)-heaviside(n-5));
stem(n,x,'filled')

grid on;
axis([-2,8,-1,6])
xlabel('n')
title('x(n)')
```

图像：



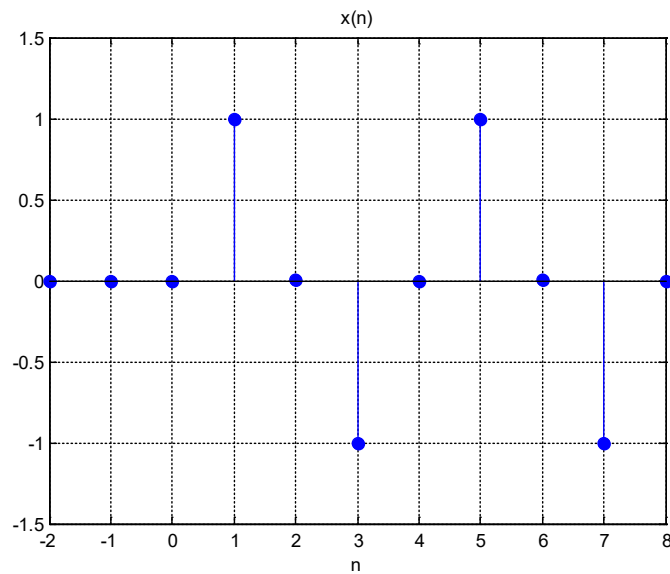
2.4  $x(n) = \sin(n\pi/2)u(n)$

代码：

```
clear
syms x;
n = -2:8;
%x = n.*(heaviside(n)-heaviside(n-5));
x = sin(n.*(pi/2)).*heaviside(n);

stem(n,x,'filled')
grid on;
axis([-2,8,-1.5,1.5])
xlabel('n')
title('x(n)')
```

图像：



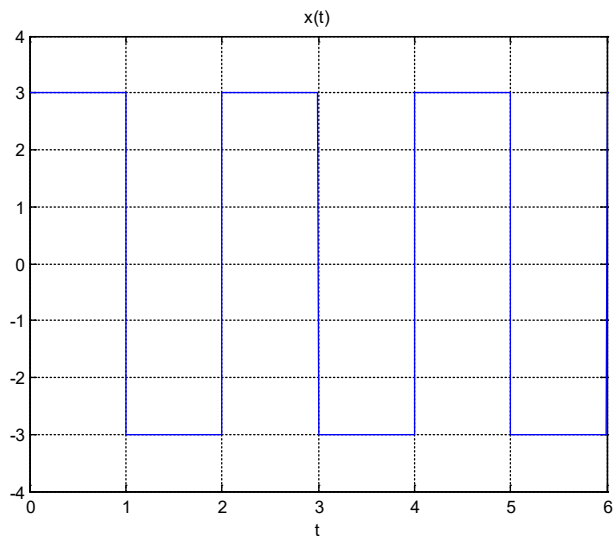
3. 利用 MATLAB 生成并绘制连续周期矩形波信号，要求周期为 2，峰值为 3，显示三个周期的波形：

代码：

```
clear
n = -2:0.0001:8;
% x = n.*(heaviside(n)-heaviside(n-5));
x = 3*square(pi.*n);
plot(n,x)

grid on;
axis([0 6 -4 4])
xlabel('t')
title('x(t)')
```

图像：

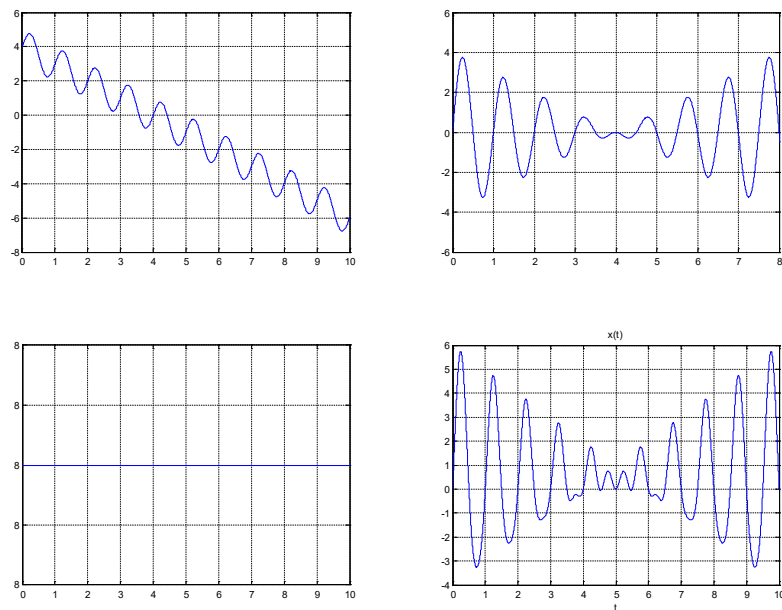


4. 已知信号  $x_1(t)$  (一个三角波的右半部分)，及信号  $x_2(t) = \sin(2\pi t)$ ，  
用 MATLAB 绘出  $x_3(t) = x_1(t) + x_2(t)$ 、 $x_4(t) = x_1(t) \times x_2(t)$ 、 $x_5 = x_1(-t) + x_1(t)$   
和  $x_6(t) = x_2(t) \times x_3(t-1)$  信号的波形 (在一个图像里面)：

代码：

```
clear
n = 0:0.0001:10;
x1 = 4-n;
x1_n = 4 + n;
x1n_1 = 5 - n;
x2n_1 = sin(2*pi.*(n-1));
x2 = sin(2*pi.*n);
subplot(221)
x3 = x1 + x2;
plot(n,x3);
grid on;
x3n_1 = x1n_1 + x2n_1;
subplot(222)
x4 = x1 .* x2;
plot(n,x4);
axis([0 8 -6 6])
grid on;
subplot(223)
x5 = x1 + x1_n;
plot(n,x5);
grid on;
subplot(224)
x6 = x2 .* x3n_1;
plot(n,x6);
xlabel('t')
title('x(t)')
```

图像：

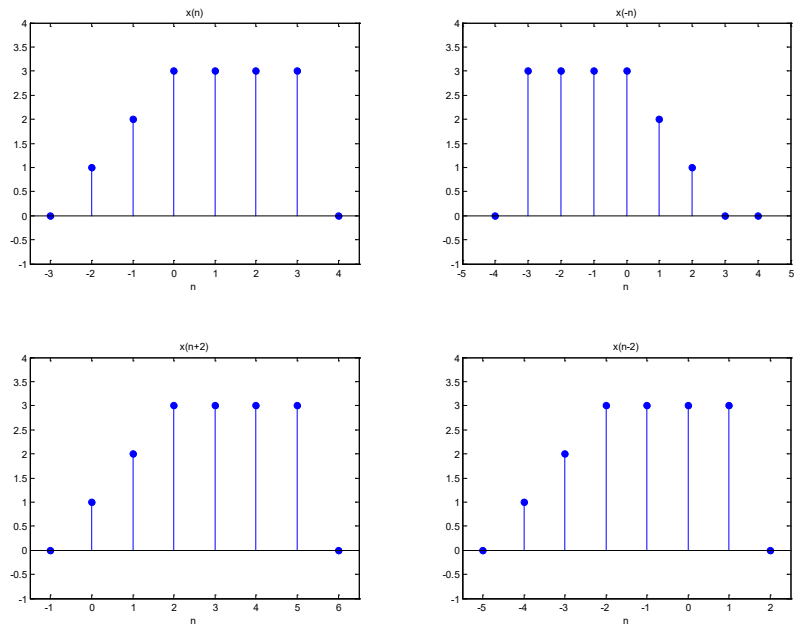


5. 已知离散时间信号  $x(n] = \{1, 2, 3, 3, 3, 3\}$ ，用 MATLAB 绘出  $x(n), x(-n), x(n+2)$  和  $x(n-2)$  的波形：

代码：

```
n = -4:4;
n1 = -n;
n2 = n+2;
n3 = n-2;
x = [0 0 1 2 3 3 3 3 0];
subplot(221)
stem(n,x,'filled');
axis([-3.5 4.5 -1 4]);
xlabel('n');
title('x(n)')
subplot(222)
stem(n1, x, 'filled')
axis([-5 5 -1 4]);
xlabel('n');
title('x(-n)')
subplot(223)
stem(n2, x, 'filled')
axis([-1.5 6.5 -1 4]);
xlabel('n');
title('x(n+2)')
subplot(224)
stem(n3, x, 'filled')
axis([-5.5 2.5 -1 4]);
xlabel('n');
title('x(n-2)')
```

图像：



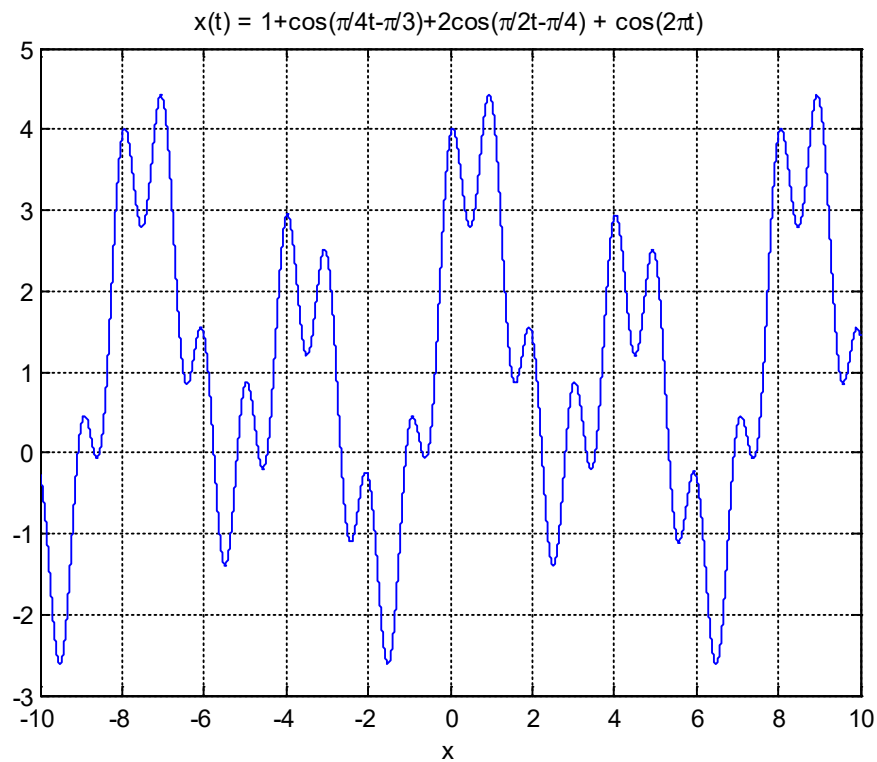
6. 用 MATLAB 编程绘制下列信号的时域波形，观察信号是否为周期信号？若是周期信号，周期是多少？若不是周期信号，请说明原因。

6.1 
$$x(t) = 1 + \cos\left(\frac{\pi}{4}t - \frac{\pi}{3}\right) + 2\cos\left(\frac{\pi}{2}t - \frac{\pi}{4}\right) + \cos(2\pi t)$$

代码：

```
t = -10:0.001:10;
x = 1 + cos(pi/4.*t-pi/3) + 2.*cos(pi/2.*t-pi/4) + cos(2*pi.*t)
plot(t,x)
xlabel('x');
title('x(t) = 1+cos(\pi/4t-\pi/3)+2cos(\pi/2t-\pi/4) + cos(2\pit)')
grid on;
```

图像：



结论：

该信号是周期信号，周期为 8

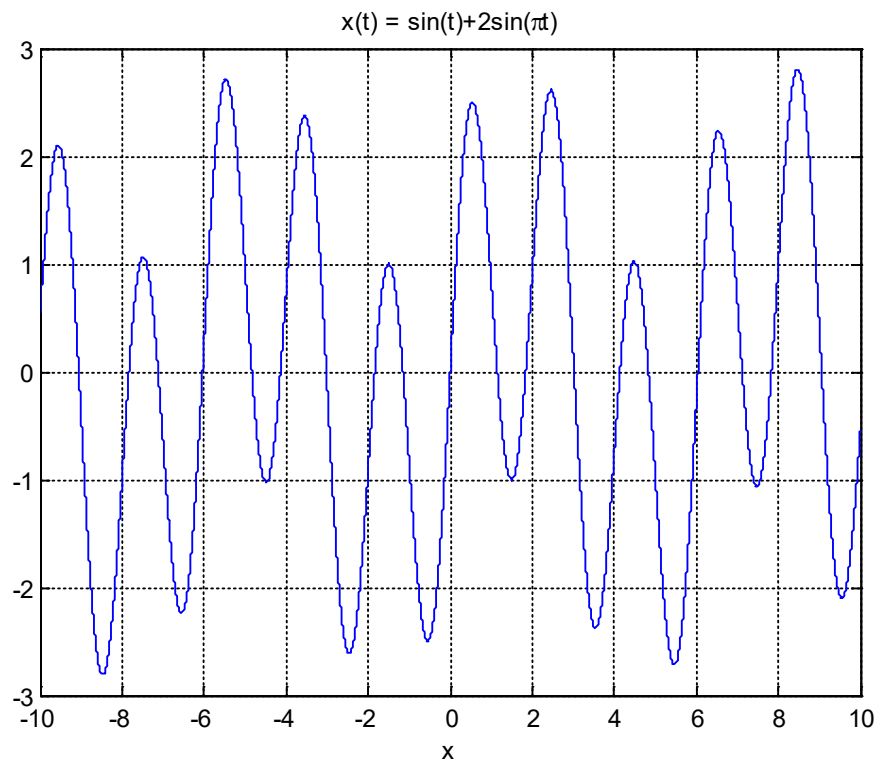
6.2  $x(t) = \sin(t) + 2\sin(\pi t)$

代码：

```
t = -10:0.001:10;
%x = 1 + cos(pi/4.*t-pi/3) + 2.*cos(pi/2.*t-pi/4) + cos(2*pi.*t)
x = sin(t)+2.*sin(pi.*t);
plot(t,x)
xlabel('x');
title('x(t) = sin(t)+2sin(\pit)')
grid on;
```



图像：



结论：

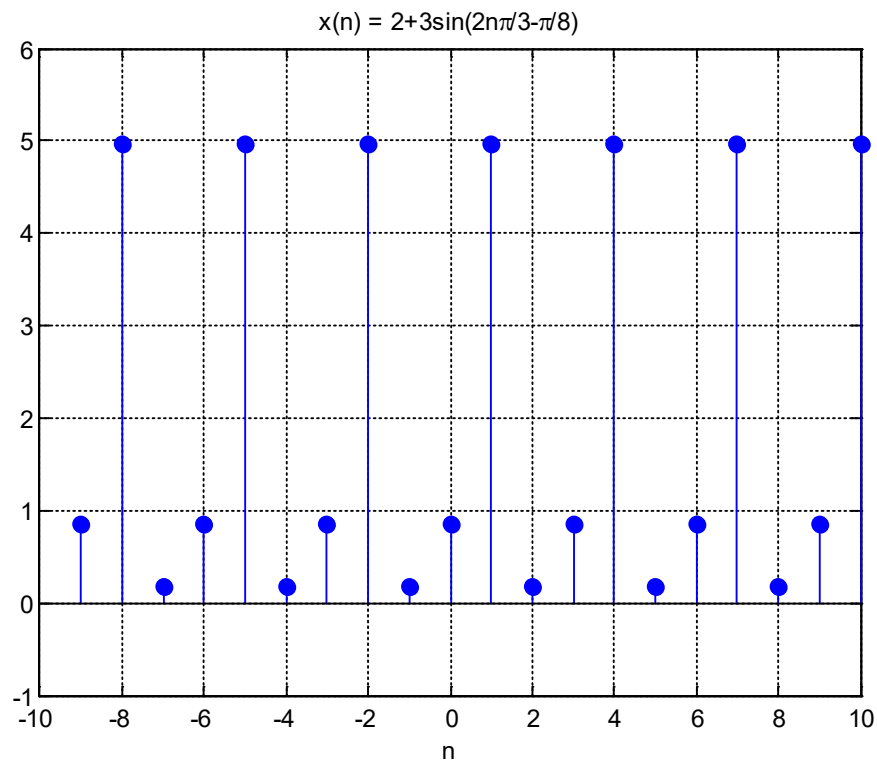
不是周期信号，此函数由一个周期为  $2\pi$  和一个周期为 2 的函数组成， $2\pi$  和 2 没有最小公倍数，所以没有周期。

$$6.3 \quad x(n) = 2 + 3 \sin\left(\frac{2n\pi}{3} - \frac{\pi}{8}\right)$$

代码：

```
t = -10:10;
%x = 1 + cos(pi/4.*t-pi/3) + 2.*cos(pi/2.*t-pi/4) + cos(2*pi.*t)
x = 2+3.*sin(2/3.*t.*pi-pi/8);
stem(t,x,'filled')
xlabel('n');
title('x(n) = 2+3sin(2n\pi/3-\pi/8)')
axis([-10 10 -1 6])
grid on;
```

图像：



结论：

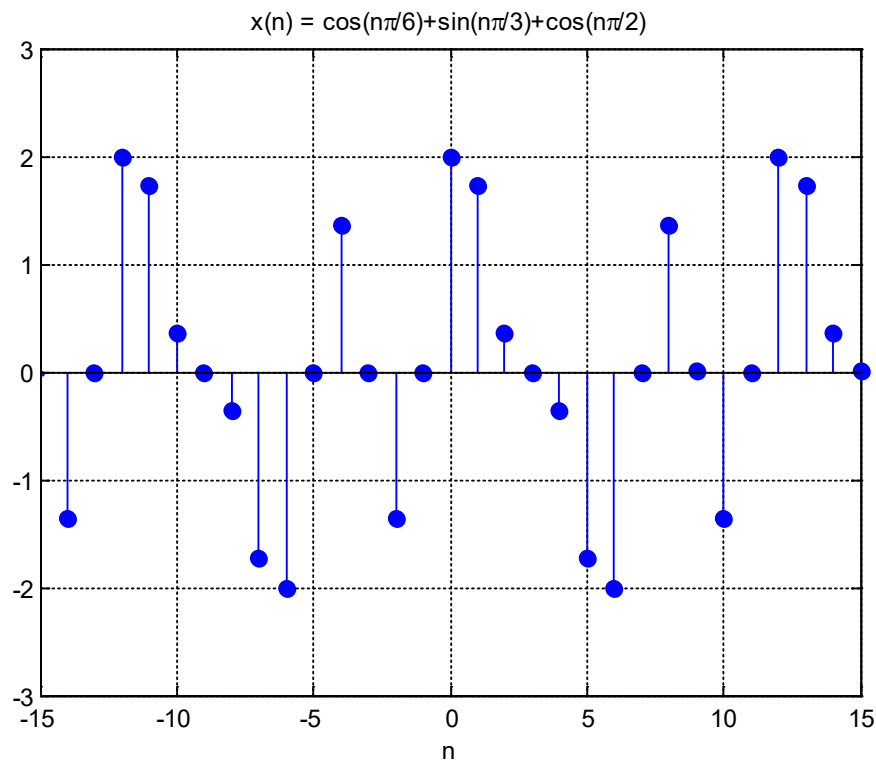
该信号是周期信号，周期为 3

6.4

代码：

```
t = -20:20;  
%x = 1 + cos(pi/4.*t-pi/3) + 2.*cos(pi/2.*t-pi/4) + cos(2*pi.*t)  
x = cos(t.*pi/6)+sin(t.*pi/3)+cos(t.*pi/2);  
stem(t,x,'filled')  
xlabel('n');  
title('x(n) = cos(n\pi/6)+sin(n\pi/3)+cos(n\pi/2)')  
axis([-15 15 -3 3])  
grid on;
```

图像：



结论：

该信号是周期信号，周期为 12。

## 四 收获和体会

第一次实验让我了解了 MATLAB 在信号与系统实验中的应用，知道了一些常见函数（如方波信号）的使用方法。不仅如此，我还学会了如何用 MATLAB 来绘制平常遇到的函数图像并进行一定的分析。

# 实验 2 LTI 系统的时域分析

## 一 实验目的

1. 掌握利用 MATLAB 对系统进行时域分析的方法。
2. 掌握连续时间系统零状态响应、冲激响应和阶跃响应的求解方法。
3. 掌握求解离散时间系统响应、单位抽样响应的方法。
4. 加深对卷积积分和卷积和的理解。掌握利用计算机进行卷积积分和卷积和计算的方法。

## 二 实验原理

### 1. 连续时间系统时域分析的 MATLAB 实现

#### 1) 连续时间系统的 MATLAB 表示

设 LTI 因果系统的微分方程一般式为：

$$a_n y^{(n)}(t) + a_{n-1} y^{(n-1)}(t) + \cdots + a_1 y'(t) + a_0 y(t) = b_m x^{(m)}(t) + b_{m-1} x^{(m-1)}(t) + \cdots + b_1 x'(t) + b_0 x(t)$$

则在 MATLAB 里，可以建立系统模型如下：

b=[];

a=[];

sys=tf(b,a);

#### 2) 连续时间系统的零状态响应

用 lsim(sys,x,t)表示求解零状态响应。

#### 3) 连续时间系统的冲激响应与阶跃响应。

用 impulse 函数来调用。

## 2. 离散时间系统时域分析的 MATLAB 实现

### 1) 离散时间系统的 MATLAB 表示

LTI 离散系统通常可以由系统差分方程描述;

则在 MATLAB 里, 可以建立系统模型如下:

$$b = [b_0, b_1, \dots, b_M];$$
$$a = [a_0, a_1, \dots, a_N];$$

### 2) 离散时间系统对任意输入的响应

用 filter (b,a,x) 函数调用。

### 3) 离散时间系统的单位抽样响应

用 impz 函数来调用。

## 3. 卷积和与卷积积分

### 离散时间序列的卷积和:

调用格式为 x=conv(x1,x2);

### 连续时间信号的卷积积分

连续时间信号  $x_1(t)$  和  $x_2(t)$  的卷积积分  $x(t)$  定义如下:

$$x(t) = x_1(t) * x_2(t) = \int_{-\infty}^{+\infty} x_1(\tau) \cdot x_2(t - \tau) d\tau$$

### 三 实验内容

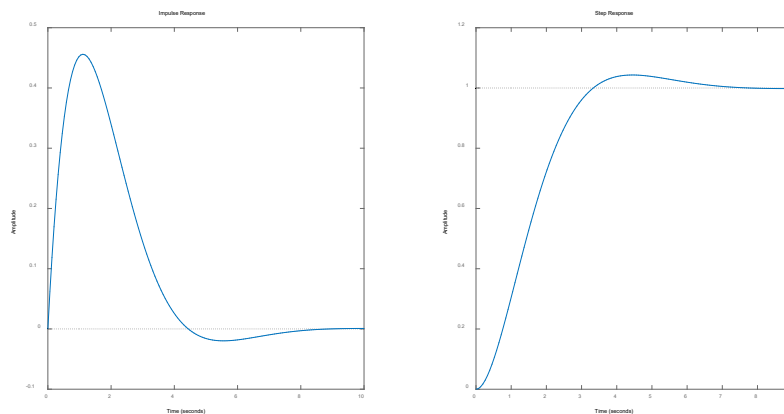
#### 1. 采用 MATLAB 绘出各系统的单位冲激响应和单位阶跃响应波形

1.1  $y''(t) + \sqrt{2}y'(t) + y(t) = x(t)$

代码：

```
out = [1 sqrt(2) 1];  
in = [1];  
sys = tf(in,out);  
subplot(121);  
impz(sys);  
subplot(122);  
step(sys);
```

图像：



1.2  $y''(t) + \sqrt{2}y'(t) + y(t) = x''(t)$

代码：

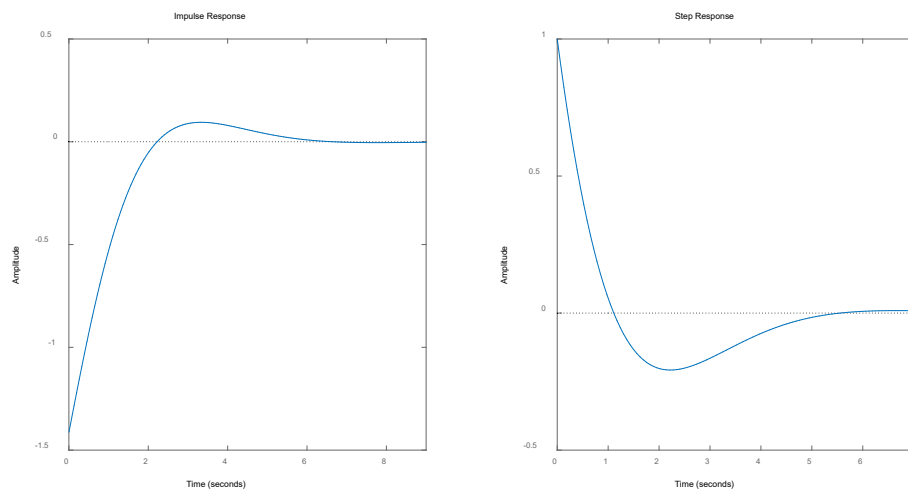
```
out = [1 sqrt(2) 1];  
in = [1 0 0];  
sys = tf(in,out);  
subplot(121);
```

```

impulse(sys);
subplot(122);
step(sys);

```

图像：



1.3  $y''(t) + y'(t) + y(t) = x'(t)$

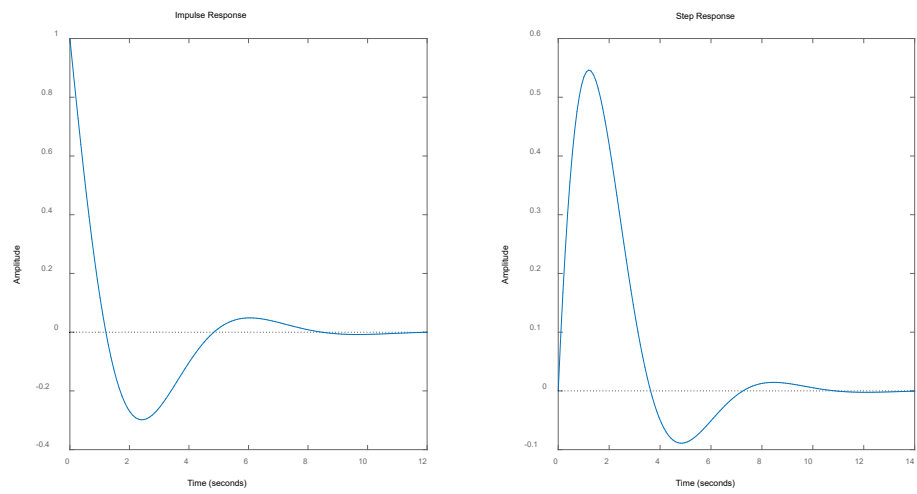
代码：

```

out = [1 1 1];
in = [1 0];
sys = tf(in,out);
subplot(121);
impulse(sys);
subplot(122);
step(sys);

```

图像：



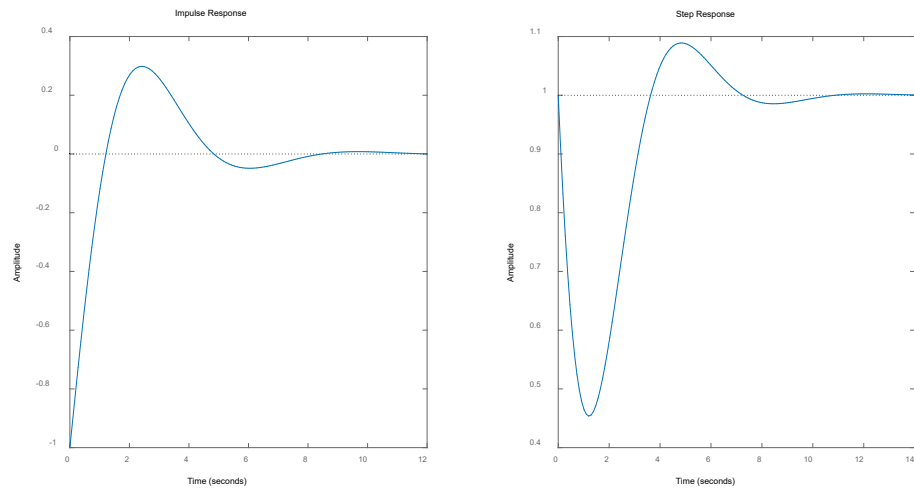
1.4  $y''(t) + y'(t) + y(t) = x''(t) + x(t)$

代码：

```
out = [1 1 1];  
in = [1 0 1];  
sys = tf(in,out);  
subplot(121);  
impz(sys);  
subplot(122);  
step(sys);
```



图像：



## 2. 已知某系统可以由如下微分方程描述

$$y''(t) + y'(t) + 6y(t) = x(t)$$

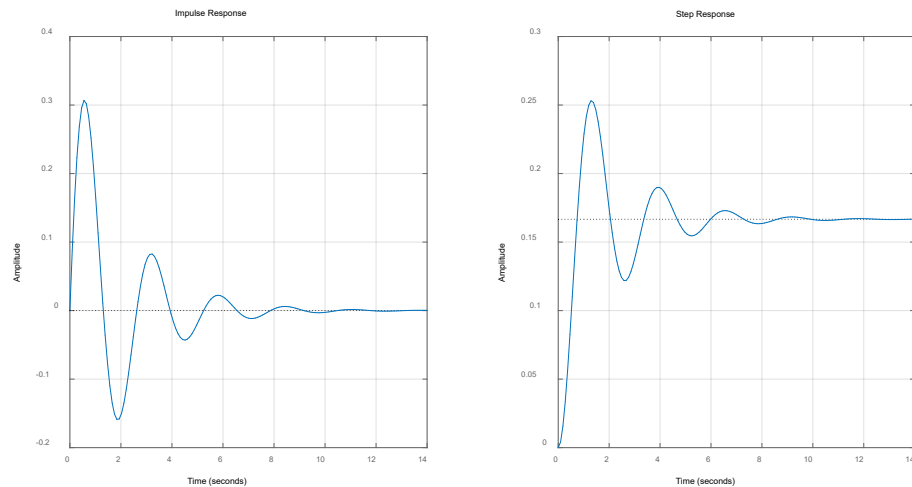
### 2.1 单位冲激响应和阶跃响应

代码：

```
out = [1 1 6];  
in = [1];  
sys = tf(in,out);  
subplot(121);  
impz(sys);
```

```
grid on;  
subplot(122);  
step(sys);  
grid on;
```

图像：



## 2.2 判断稳定性

观察输出的波形，当  $t$  趋向于无穷大时， $y(t)$  趋于 0，所以该系统是稳定的

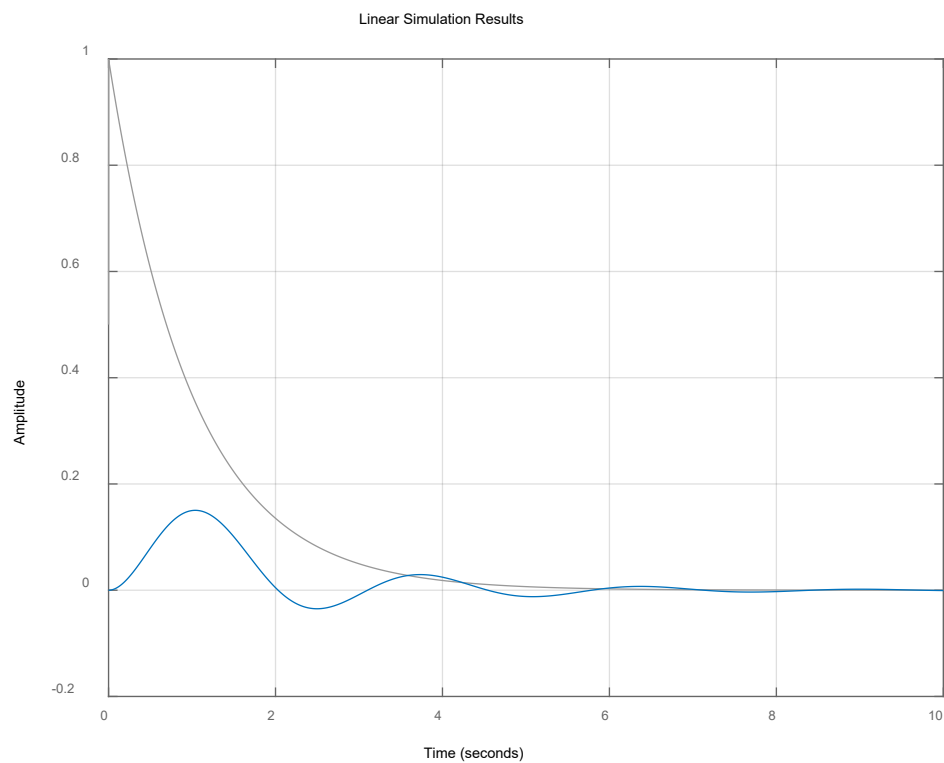
## 2.3 如果系统的输入为 $x(t) = e^{-t}u(t)$ ，求系统的零状态响应

代码：

```
out = [1 1 6];
in = [1];
t = 0:0.001:10;
x = exp(-t).*heaviside(t);
%sys = tf(in,out);
%subplot(121);
%impulse(sys);

%grid on;
%subplot(122);
%step(sys);
lsim(sys,x,t);%零状态响应
grid on;
```

图像：



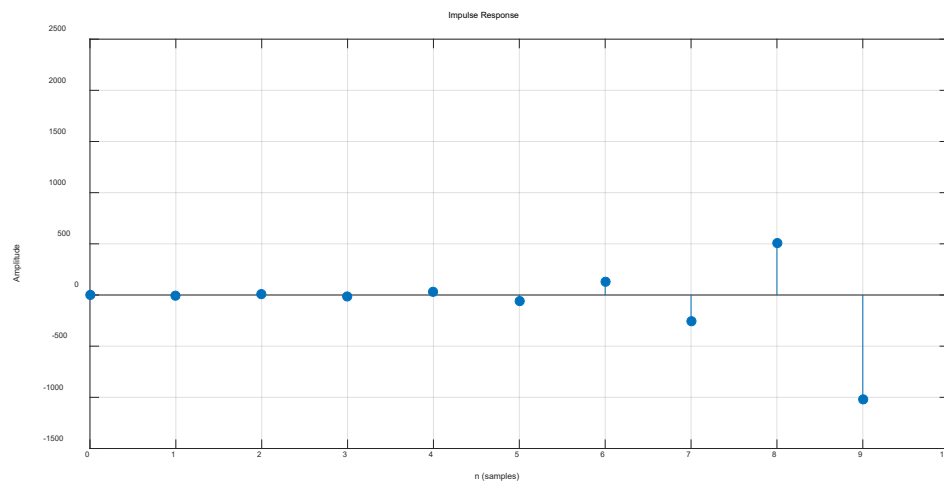
3. 已知描述离散系统的微分方程如下，用 MATLAB 绘出各系统的单位抽样响应，根据单位抽样响应的时域波形分析系统的稳定性。

$$3.1 \quad y(n) + 3y(n-1) + 2y(n-2) = x(n)$$

代码：

```
out = [1 3 2];  
in = [1];  
t = 0:0.001:10;  
%x = exp(-t).*heaviside(t);  
sys = tf(in,out);  
impz(in,out,0:10);  
grid on;
```

图像：

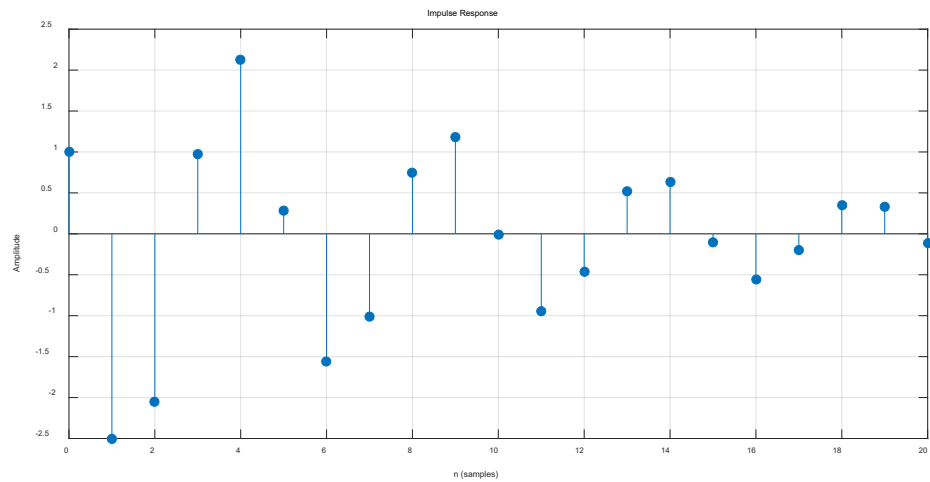


$$3.2 \quad y(n) + 3y(n-1) + 2y(n-2) = x(n)$$

代码：

```
out = [1 -0.5 0.8];  
in = [1 -3];  
t = 0:0.001:10;  
%x = exp(-t).*heaviside(t);  
sys = tf(in,out);  
impz(in,out,0:20);  
grid on;
```

图像：



4. 已知系统可以由如下差分方程描述  $y(n] + y(n-1) + 0.25y(n-2) = x(n)$  ,  
绘出单位抽样响应和单位阶跃响应。

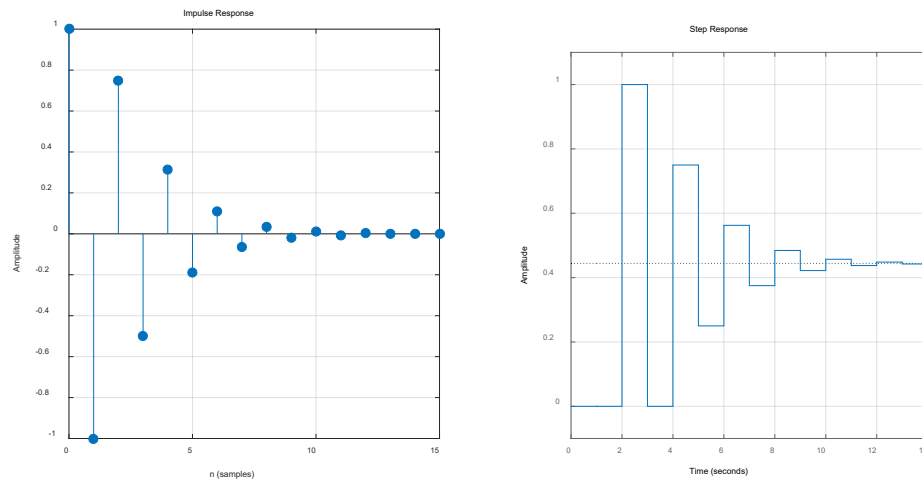
## 4.1

代码：

```
out = [1 1 0.25];
in = [1];
t = 0:0.001:10;
%x = exp(-t).*heaviside(t);
sys = tf(in,out);
subplot(121);
impz(in,out,0:15);

grid on;
%impulse(sys);
%grid on;
subplot(122);
dstep(in,out,0:15);
grid on;
%step(sys);
```

图像：



## 5. 用 MATLAB 计算如下两个序列的卷积，并绘出图形

$$x_1(n) = \{1, 2, 1, 1\}$$

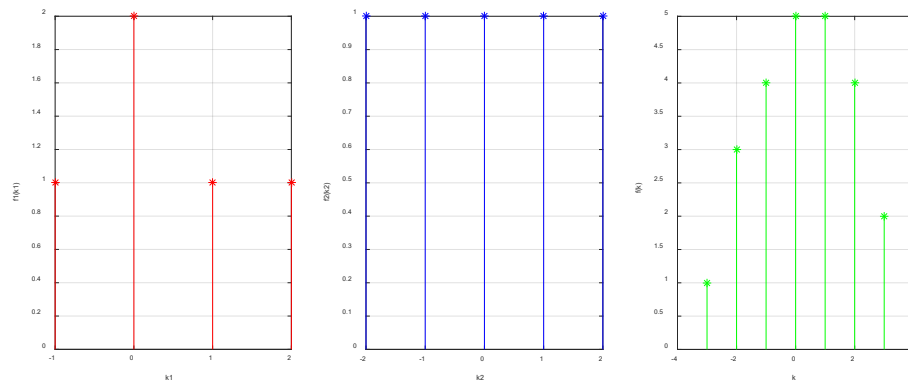
$$x_2(n) = \begin{cases} 1, & -2 \leq n \leq 2 \\ 0, & \text{其他} \end{cases}$$

代码：

```
f1 = [1 2 1 1];  
f2 = [1 1 1 1 1];  
k1 = [-1 0 1 2];  
k2 = [-2 -1 0 1 2]  
f=conv(f1,f2);  
k(1)=k1(1)+k2(1);  
k(length(f))=k1(length(f1))+k2(length(f2));  
k=k(1):k(length(f));  
subplot(131);  
stem(k1,f1,'*r');  
xlabel('k1');  
ylabel('f1(k1)');
```

```
grid on  
subplot(132);  
stem(k2,f2,'*b');  
xlabel('k2');  
ylabel('f2(k2)');  
grid on  
subplot(133);  
stem(k,f,'*g');  
xlabel('k');  
ylabel('f(k)');  
grid on
```

图像：

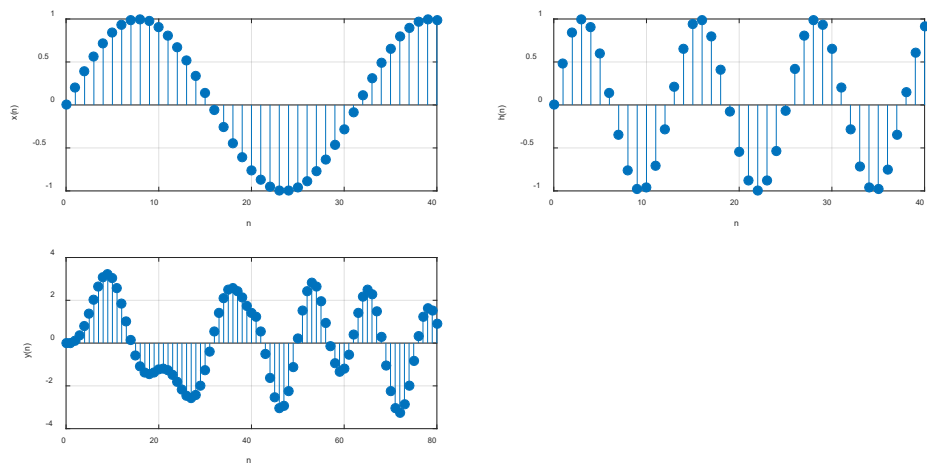


6. 已知某 LTI 离散系统，其单位抽样响应  $h(n) = \sin(0.5n), n \geq 0$ ，系统的输入为  $x(n) = \sin(0.2n), n \geq 0$ ，计算当  $n=0,1,2, \dots, 40$  时系统的零状态响应  $y(n)$ ，绘出  $x(n), h(n)$  和  $y(n)$  时域波形

代码：

```
n = 0:40;
h = sin(0.5.*n);
x = sin(0.2.*n);
[y t] = sconv(h,x,n,n,1);
subplot(221);
stem(n,x,'filled');
xlabel('n');
ylabel('x(n)');
grid on;
subplot(222);
stem(n,h,'filled');
xlabel('n');
ylabel('h(n)');
grid on;
subplot(223);
stem(t,y,'filled');
xlabel('n');
ylabel('y(n)');
```

图像：



## 7. 已知两个连续时间信号，求两个信号的卷积

代码：

```
function y = u(t)%函数
y = (t>=0);
```

```
clear;%主函数：
```

```
close all;
```

```
t0 = -3;
```

```
t1 = 3;
```

```
dt = 0.01;
```

```
t = t0:dt:t1;
```

```
x1 = 2*u(t+1)-2*u(t-1);
```

```
x2 = u(t+2)-u(t-2);
```

```
y = dt*conv(x1,x2);
```

```
subplot(221);
```

```
plot(t,x1);
```

```
grid on;
```

```
title('f1(t)');
```

```
xlabel('t')
```

```
ylabel('f1(t)')
```

```
axis([-1.5 1.5 0 2.5])
```

```
subplot(222);
```

```
plot(t,x2);
```

```
grid on;
```

```
title('f2(t)')
```

```
xlabel('t')
```

```
ylabel('f2(t)')
```

```
axis([-3 3 0 1.5])
```

```
subplot(212);
```

```
t = 2*t0:dt:2*t1;
```

```
plot(t,y);
```

```
title('f(t)=f1(t)*f2(t)')
```

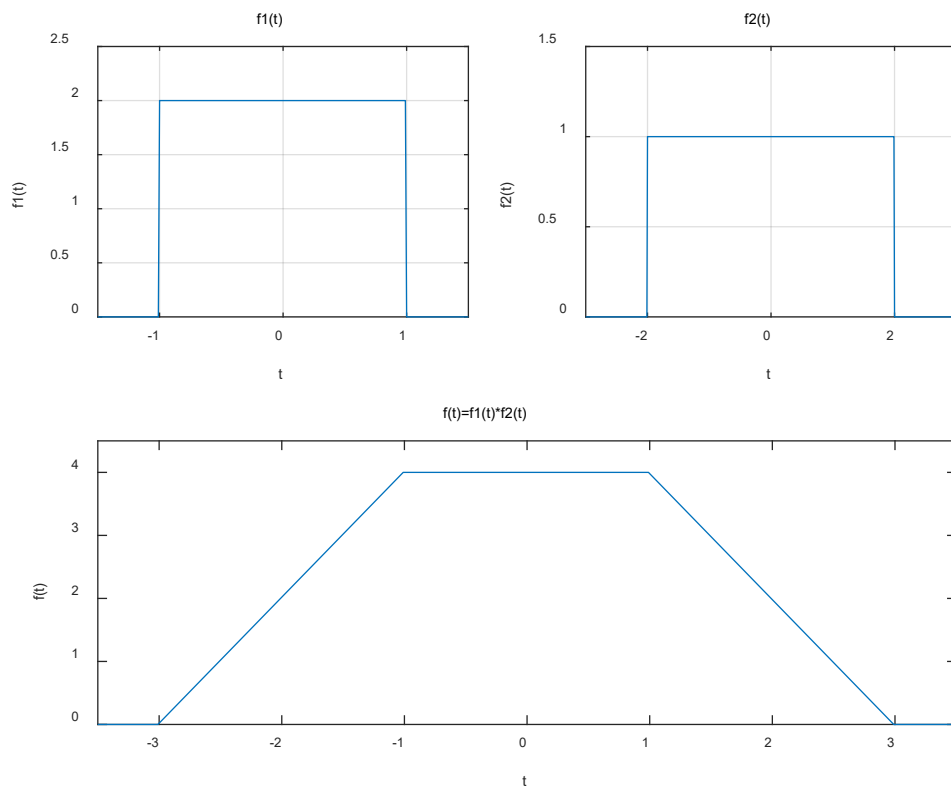
```
xlabel('t')
```

```
ylabel('f(t)')
```

```
axis([-3.5 3.5 0 4.5])
```



图像：



## 四 收获和体会

MATLAB 强大的运算功能深深的震撼到了我，信号课上烦人的卷积到了 MATLAB 里面可以轻易的画出并且可以随意增添图形属性。一些响应波形也可以轻松画出，十分感谢这次信号与系统软件实验，让我掌握了 MATLAB 除了画图和数据分析外的又一个新技能。

# 实验 3 信号的频域分析

## 一 实验目的

1. 深入理解信号频谱的概念，掌握信号的频域分析方法。
2. 观察典型周期信号和非周期信号的频谱，掌握其频谱特性。

## 二 实验原理

### 1. 连续周期信号的频谱分析

如果周期信号满足狄里赫利条件，就可以展开为傅里叶级数形式，即

$$x(t) = \sum_{k=-\infty}^{+\infty} c_k e^{jk\omega_0 t} \dots\dots\dots(1)$$

$$c_k = \frac{1}{T_0} \int_{T_0} x(t) e^{-jk\omega_0 t} dt \dots\dots\dots(2)$$

式中， $T_0$  表示基波周期， $\omega_0 = 2\pi / T_0$  为基波频率， $\int_{T_0} (\bullet)$  表示任一个基波周期内的积分。

式 (1) 和式 (2) 定义为周期信号复指数形式的傅里叶级数，系数  $c_k$  称为  $x(t)$  的傅里叶系数。周期信号的傅里叶级数还可以由三角函数的线性组合来表示，即

$$x(t) = a_0 + \sum_{k=1}^{+\infty} a_k \cos k\omega_0 t + \sum_{k=1}^{+\infty} b_k \sin k\omega_0 t \dots\dots\dots(3)$$

其中

$$a_0 = \frac{1}{T_0} \int_{T_0} x(t) dt, a_k = \frac{2}{T_0} \int_{T_0} x(t) \cos k\omega_0 t dt, b_k = \frac{2}{T_0} \int_{T_0} x(t) \sin k\omega_0 t dt \dots\dots\dots(4)$$

式 (3) 中同频率的正弦项和余弦项可以合并，从而得到三角函数形式的傅里叶级数，即

$$x(t) = A_0 + \sum_{k=1}^{+\infty} A_k \cos(k\omega_0 t + \theta_k) \dots\dots\dots(5)$$

其中

$$A_0 = a_0, A_k = \sqrt{a_k^2 + b_k^2}, \theta_k = -\arctan \frac{b_k}{a_k} \dots\dots\dots(6)$$

可见，任何满足狄里赫利条件的周期信号都可以表示成一组谐波关系的复指数函数或三角函数的叠加。一般来说周期信号表示为傅里叶级数时需要无限多项才能完全逼近原信号，但在实际应用中经常采用有限项级数来替代，所选项数越多就越逼近原信号。

## 2. 连续非周期信号的频谱分析

对于非周期连续时间信号，吸纳后的傅里叶变换和傅里叶逆变换定义为

$$X(\omega) = \int_{-\infty}^{+\infty} x(t)e^{-j\omega t} dt \dots\dots\dots(7)$$

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} X(\omega)e^{j\omega t} d\omega \dots\dots\dots(8)$$

式（7）和式（8）把信号的时域特性和频域特性联系起来，确立了非周期信号  $x(t)$  和频谱  $X(\omega)$  之间的关系。

采用 MATLAB 可以方便地求取非周期连续时间信号的傅里叶变换，这里我们介绍常用的集中方法。

### 1) 符号运算法

MATLAB 的符号数学工具箱提供了直接求解傅里叶变换和反变换的函数，fourier 函数和 ifourier 函数，基本调用格式为

$X = \text{fourier}(x)$

$X = \text{ifourier}(X)$

默认的时域变量为  $t$ ，频域变量为  $\omega$ 。

### 2) 数值积分法

除了采用符号运算的方法外，我们还可以利用 MATLAB 的 quad 函数，采用数值积分的方法来进行连续信号的频谱分析，quad 函数是一个用来计算数值积分的函数。利用 quad 函数可以计算非周期连续时间信号的频谱。Quad 函数的一般调用格式为：

$y = \text{quad}(\text{fun}, a, b)$

$y = \text{quad}(\text{fun}, a, b, \text{TOL}, \text{TRACE}, p1, p2, \dots)$

其中 fun 指定被积函数，可以采用 inline 命令来创建，也可以通过传递函数句柄的形式来指定，a、b 表示定积分的下限和上限，TOL 表示允许的相对或绝对积分误差，TRACE 表示以被积函数的点绘图形式来跟踪该函数的返回值，如果 TOL 和 TRACE 为空矩阵，则使用缺省值，“p1,p2, …”表示被积函数出时间  $t$  之外所需的其他额外输入参数。

### 3) 数值近似法

我们还可以利用 MATLAB 的数值计算的方法近似计算连续时间傅里叶变换。傅里叶变换  $X(\omega)$  可以由式（9）近似计算

$$X(\omega) = \int_{-\infty}^{+\infty} x(t) e^{-j\omega t} dt = \lim_{\Delta \rightarrow 0} \sum_{k=-\infty}^{+\infty} x(k\Delta) e^{-j\omega k\Delta} \Delta \dots\dots\dots (9)$$

当  $x(t)$  为时限信号，且  $\Delta$  足够小，式 (9) 可以演变为

$$X(\omega) = \Delta \sum_{k=a}^b x(k\Delta) e^{-j\omega k\Delta} \dots\dots\dots (10)$$

而式 (10) 中求和部分又可以表示成一个行向量和一个列向量的乘积

$$\sum_{k=a}^b x(k\Delta) e^{-j\omega k\Delta} = [x(a\Delta), x((a+1)\Delta), \dots, x(b\Delta)] \bullet \begin{bmatrix} e^{-j\omega a\Delta} \\ e^{-j\omega (a+1)\Delta} \\ \dots \\ e^{-j\omega b\Delta} \end{bmatrix} \dots\dots\dots (11)$$

式 (11) 可以很方便地利用 MATLAB 实现。

### 3. 离散周期时间信号的频域分析

基波周期为  $N$  的周期序列  $x(n)$  可以用  $N$  个成谐波关系的复指数序列的加权和表示，即

$$x(n) = \sum_{k=\langle N \rangle} c_k e^{jk(2\pi/N)n} \dots\dots\dots (12)$$

这里  $k=\langle N \rangle$  表示求和仅需包括一个周期内的  $N$  项，周期序列在一个周期内的求和与起点无关。将周期序列表示成式 (12) 的形式，成为离散傅里叶级数，而系数  $c_k$  则称为离散傅里叶系数。离散傅里叶系数  $c_k$  可以由式 (13) 确定。

$$c_k = \frac{1}{N} \sum_{k=\langle N \rangle} x(n) e^{-jk(2\pi/N)n} \dots\dots\dots (13)$$

傅里叶系数  $c_k$  也称为  $x(n)$  的频谱系数，而且可以证明  $c_k$  是以  $N$  为周期的离散频率序列。这说明了周期的离散时间函数对应于频域为周期的离散频率。

这里，我们用周期  $N$  与傅里叶系数  $c_k$  的乘积来表示周期离散时间信号的频谱，即

$$X(k) = N \bullet c_k = \sum_{k=\langle N \rangle} x(n) e^{-jk(2\pi/N)n} \dots\dots\dots (14)$$

$X(k)$  可以利用 MATLAB 提供的函数 `fft` 用来计算，调用格式为

$$X = \text{fft}(x)$$

该函数返回  $X(k)$  一个周期内的值，其中  $x$  表示  $x(n)$  一个周期内的样本值。

## 4. 离散非周期时间信号的频域分析

非周期序列  $x(n)$  可以表示成一组复指数序列的连续和

$$x(n) = \frac{1}{2\pi} \int_{2\pi} X(e^{j\Omega}) e^{j\Omega n} d\Omega \dots\dots\dots (15)$$

其中

$$X(e^{j\Omega}) = \sum_{n=-\infty}^{+\infty} x(n) e^{-j\Omega n} \dots\dots\dots (16)$$

式 (16) 称为  $x(n)$  的离散时间傅里叶变换, 式 (15) 和式 (16) 确立了非周期离散时间信号  $x(n)$  及其离散时间傅里叶变换  $X(e^{j\Omega})$  之间的关系。  $X(e^{j\Omega})$  是连续频率  $\Omega$  的函数, 称为频谱函数, 且  $X(e^{j\Omega})$  是周期的连续频率函数, 其周期为  $2\pi$ 。可见, 非周期离散时间函数对应于频域中是一个连续的周期的频率函数。

对于有限长的时间序列, 式 (16) 可以表示为

$$X(e^{j\Omega}) = \sum_{n=n_1}^{n_N} x(n) e^{-j\Omega n} = [x(n_1), x(n_2), \dots, x(n_N)] \bullet \begin{bmatrix} e^{-jn_1\Omega} \\ e^{-jn_2\Omega} \\ \dots \\ e^{-jn_N\Omega} \end{bmatrix} \dots\dots\dots (17)$$

式 (17) 可以方便地利用 MATLAB 实现。

## 三 实验内容

### 1. 已知周期矩形脉冲信号 $x(t)$

#### 1.1 计算其傅里叶级数

周期矩形方波的傅里叶级数:

$$x_N(t) = \frac{A}{2} + \frac{2A}{\pi} \left( \cos \omega_0 t - \frac{1}{3} \cos 3\omega_0 t + \frac{1}{5} \cos 5\omega_0 t - \dots \right)$$

## 1.2 利用 MATLAB 绘出由前 N 次谐波合成的信号波形，观察随着 N 的变化合成信号波形的变化规律

代码：

```
T = input('输入周期矩形脉冲信号的周期: ');
t1 = input('输入周期矩形脉冲信号的脉冲宽度: ');
N = input('输入 N: ');
```

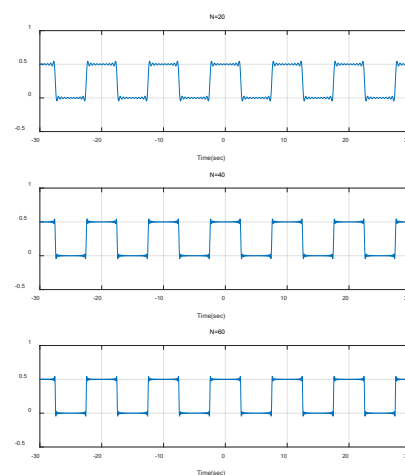
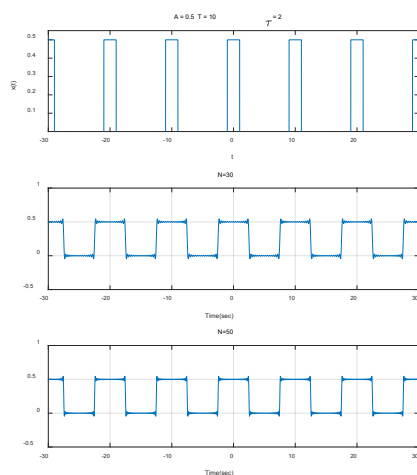
```
A = 1/t1;
t = -3*T:0.01:3*T;
y = A.*square(2*pi/T.*(t+t1/2),t1/T*100);
```

```
%绘周期方波
subplot(3,2,1)
plot(t,y);
axis([-3*T 3*T 0 A*1.1])
xlabel('t');
ylabel('x(t)');
```

```
title(['A = ' num2str(A) ' T = ' num2str(T) ' \tau = ' num2str(t1)]);
%绘周期方波
```

```
for i = 2:6
    x = zeros(size(t))+A/2;%初始化 x 变量
    w = 2*pi/T;%Omega
    for n = 1:2:N*i
        x = x + ((-1)^(n+3)/2)*(2*A/(pi*n)*cos(n*w.*t));
    end
    subplot(3,2,i)
    plot(t,x);
    grid on;
    title(['N=' num2str(N*i)]);
    xlabel('Time(sec)');
end
```

图像：



## 结论：

随着  $N$  的增大即选取的傅里叶级数的项数增加，合成波形越来越接近原来的矩形脉冲信号。

## 1.3 利用 MATLAB 绘出周期矩形脉冲信号的频谱，观察参数 $T$ 和 $\tau$ 变化时对频谱波形的影响。

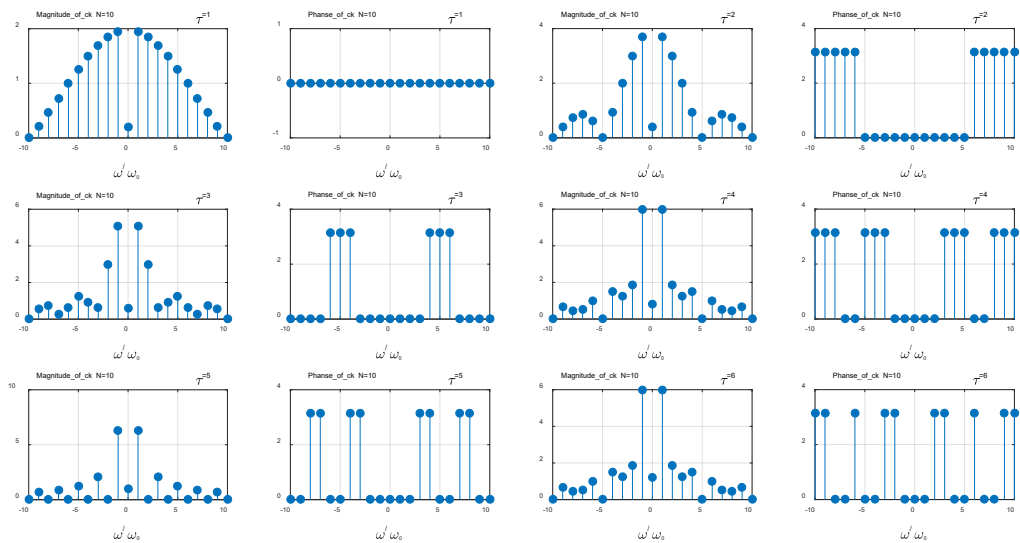
### 代码：

```
T = input('输入周期矩形脉冲信号的周期: ');%10
t0 = input('输入周期矩形脉冲信号的脉冲宽度: ');%1
A = input('输入 A: ');%2
N = input('输入 N: ');%10
%A = 1/t1;
for i = 1:6
    j = 2*i-1;
    t1 = t0*i;
    n1 = -N: -1;
    c1 = (A*pi./n1).*sin(n1.*pi.*t1./T);
    c0 = A * t1/T;
    n2 = 1:N;
    c2 = (A*pi./n2).*sin(n2.*pi.*t1./T);
    cn = [c1 c0 c2];

    n = -N:N;
    subplot(3,4,j);
    stem(n,abs(cn),'filled');
    xlabel('\omega/\omega_0');
    title(['Magnitude\_of\_ck   N=' num2str(N) ' \tau=' num2str(t1)]);
    grid on;

    subplot(3,4,j+1);
    stem(n,angle(cn),'filled');
    xlabel('\omega/\omega_0');
    title(['Phanse\_of\_ck   N=' num2str(N) ' \tau=' num2str(t1)]);
    grid on;
end
```

图像：



结论：

频谱波形与  $\frac{\tau}{T}$  的值有关，对于不同的  $T$  和  $\tau$ ，当比值相同时频谱波形图相同；当比值不同时，比值越小，频谱包络形状趋于收敛，过零点越少，谱线越密。

思考题

1) 什么是吉布斯现象，产生吉布斯现象的原因是什么？

将具有不连续点的周期函数（如矩形脉冲）进行傅立叶级数展开后，选取有限项进行合成。当选取的项数越多，在所合成的波形中出现的峰起越靠近原信号的不连续点。当选取的项数很大时，该峰值趋于一个常数。这种现象称为吉布斯现象。  
原因：当一个信号通过某一系统时，如果这个信号不是连续时间函数，则由于一般物理系统对信号高频分量都有衰减作用，从而产生吉布斯现象。

2) 以周期矩形脉冲信号为例，说明周期信号的频谱有什么特点。

周期信号的频谱是具有周期性的一系列脉冲信号，谱线间隔为  $\omega_0$ ，谱线的长度随着谐波次数的增高趋于收敛。

3) 周期矩形脉冲信号的有效频带宽度与信号的时域宽度有什么关系？

有效频宽与信号的时域宽度成反比



#### 4) 随着矩形脉冲信号参数 $\frac{\tau}{T}$ 的变化，其频谱结构如何变化？

比值越小，频谱包络形状趋于收敛，过零点越少，谱线越密

## 2. 已知矩形脉冲信号 $x(t)$

### 2.1 求该信号的傅里叶变换

傅里叶变换：

$$X(\omega) = AT_1 \frac{\sin(\omega T_1 / 2)}{\omega T_1 / 2} = AT_1 \text{sinc}(\omega T_1 / 2)$$

### 2.2 利用 MATLAB 绘出矩形脉冲信号的频谱，观察矩形脉冲宽度变化时对频谱波形的影响。

代码：

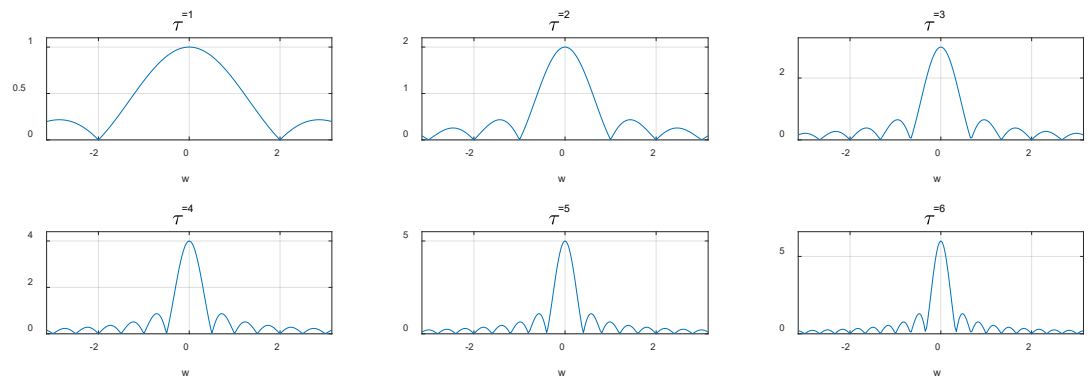
```
A = input('输入 A=');
subplot(2,3,T);
%A = 1/T;
ezplot(abs(y1),[-pi,pi]);
syms w;
title(['\tau=' num2str(T)])
xlabel('\omega');

for T = 1:6
axis([-pi pi 0 A*T*1.1]);

%振幅频谱
grid on;

y1 = A*T*abs(sinc(T/2.*w));
end
```

图像：



结论：

当矩形脉冲宽度  $\tau$  增大时，信号占有频带  $B_\omega$  减小，即信号的占有频带  $B_\omega$  与脉冲宽度  $\tau$  成反比。

**2.3 让矩形脉冲的面积始终等于 1，改变矩形脉冲宽度，观察矩形脉冲信号时域波形和频谱随矩形脉冲宽度的变化趋势。**

代码：

```
syms w;
```

```
for T = 1:3
```

```
%原图形
```

```
A = 1/T;
```

```
j = 2*T-1;
```

```
subplot(3,2,j)
```

```
t = -T:.001*T:T;
```

```
x=A*heaviside(t+T/2)-A*heaviside(t-T/2);
```

```
plot(t,x)
```

```
axis([-T T 0 A*1.1]);
```

```
grid on;
```

```
%振幅频谱
```

```
y1 = A*T*abs(sinc(T/2.*w));
```

```
subplot(3,2,j+1);
```

```
ezplot(abs(y1),[-pi,pi]);
```

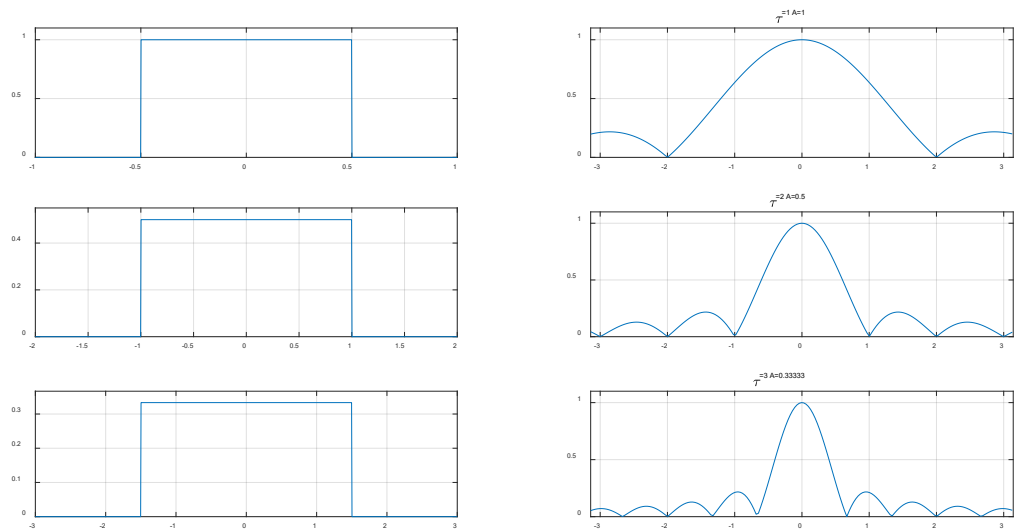
```
title(['\tau=' num2str(T) ' A=' num2str(A)])
```

```
xlabel('');
```

```
axis([-pi pi 0 A*T*1.1]);
end

grid on;
```

图像：



结论：

脉冲宽度越小，时域波形的幅值越大，信号占有的频带宽度越大。

## 思考题

1) 比较矩形脉冲信号和周期矩形脉冲信号的频谱，两者之间有什么异同。

相同点：它们的有效频带宽度都是与脉冲宽度成反比。

不同点：周期矩形脉冲信号的频谱是离散的，而矩形脉冲信号的频谱是连续的

2) 根据矩形脉冲宽度变化时频谱的变化规律，说明信号的有效频带宽度与其时域宽度之间有什么关系。当脉冲宽度趋向于 0 时，脉冲的面积始终等于 1，其频谱有什么特点？

矩形脉冲信号的有效频带宽度与时域宽度成反比，当  $\tau \rightarrow 0$ ，脉冲面积始终等于 1 时，其频谱会无限趋近于高度为 1 的一条直线。

### 3. 已知周期方波序列

$$x(n) = \sum_{k=-\infty}^{+\infty} [u(kN + n + N_1) - u(kN + n - N_1)]$$
，观察  $N$  和  $N_1$  变化时频谱波形的

变化趋势。

#### 3.1

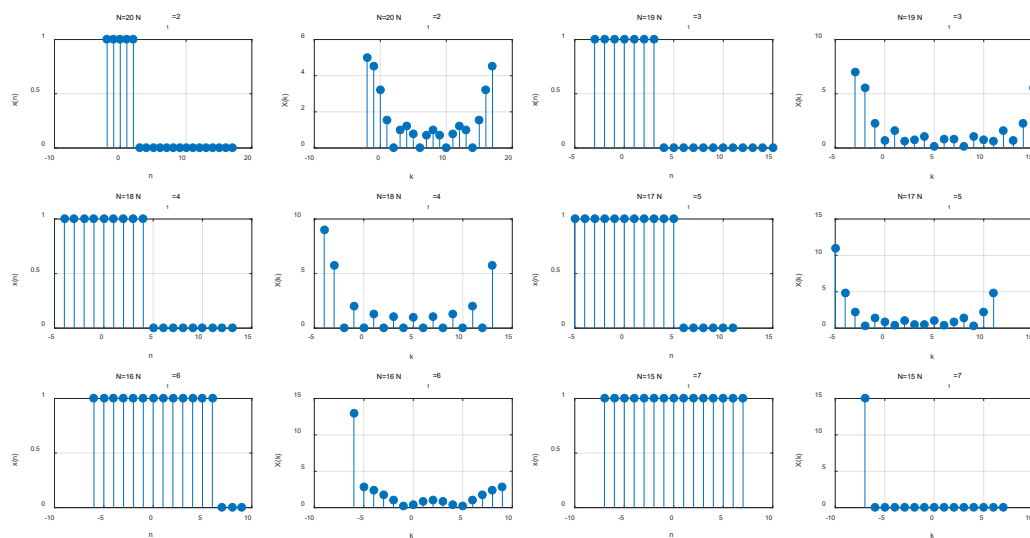
代码：

```
N_0=input('N=');
N1_0=input('N1=');
for i = 1:6
    j = 2*i-1;
    N = N_0-i;
    N1=N1_0+i;
    n=-N1:N1;
    x1=ones(size(n));
    n=N1+1:N-N1-1;
    x2=zeros(size(n));
    x=[x1,x2];
    n=-N1:N-N1-1;
    X=fft(x);

    subplot(3,4,j);
    stem(n,x,'filled');
    xlabel('n');
    ylabel('x(n)');
    title(['N=' num2str(N) ' N_{1}='
    num2str(N1)]);
    grid on;

    subplot(3,4,j+1);
    stem(n,abs(X),'filled');
    xlabel('k');
    ylabel('X(k)');
    title(['N=' num2str(N) ' N_{1}='
    num2str(N1)]);
    grid on;
end
```

图像：



思考题：

1) 以周期方波序列为例，说明周期序列与连续周期信号的频谱有何异同。

相同点：周期序列与连续周期信号的频谱都是离散的，且都有收敛性和谐波性。

不同点：连续周期信号在一个周期内要用无限多项级数来表示，而周期序列用有限项级数就可以表示。

2) 随着周期方波序列占空比的变化，其频谱如何随之变化？

方波序列占空比越小，频谱的谱线越密集，谱线高度越大。

4. 已知一矩形脉冲序列  $x(n) = \begin{cases} 1, & |n| \leq N_1 \\ 0, & |n| > N_1 \end{cases}$ ，利用 MATLAB 绘制周期方波序列的频谱波形，改变矩形脉冲序列的宽度，观察频谱波形的变化趋势。

4.1

代码：

```

N1_0=input('N1=');
for l=1:4
    j = 2*l-1;
    N1 = N1_0+j;
    n=-N1:N1;
    w=-pi:0.01:pi;
    x=ones(size(n));
    X=x*exp(-1i*n'*w);

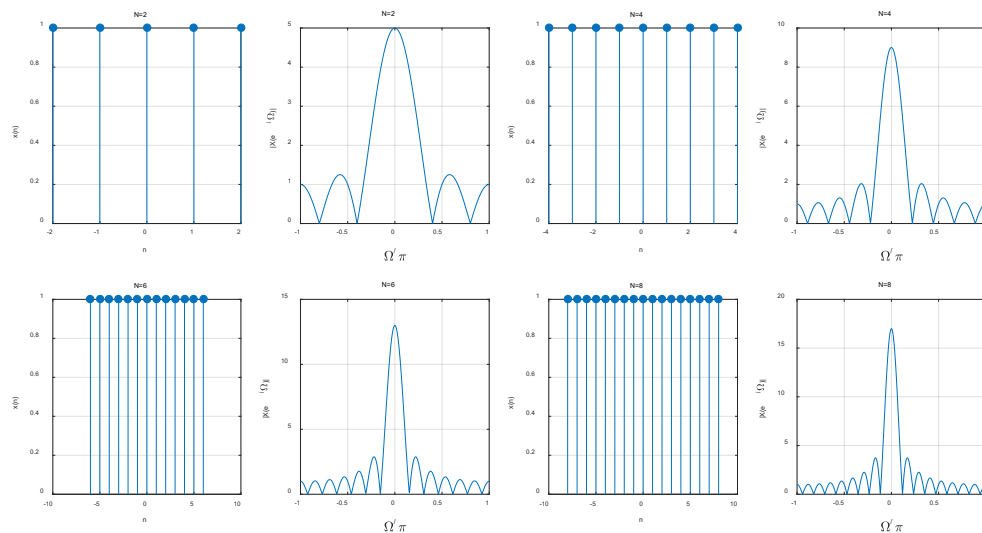
    subplot(2,4,j);
    stem(n,x,'filled');
    xlabel('n');

    ylabel('x(n)');
    title(['N=' num2str(N1)])
    grid on;

    subplot(2,4,j+1);
    plot(w/pi,abs(X));
    grid on;
    xlabel('\Omega/\pi');
    ylabel('|X(e^{j\Omega})|');
    title(['N=' num2str(N1)])
    grid on;
end

```

图像：



思考题：

1) 随着矩形脉冲信号宽度的变化，其频谱如何随着变化？其宽度与频谱的有效频带宽度有何关系？

随着矩形脉冲序列宽度的增加，其频谱的有效频带宽度减小。其宽度与频谱的有效频带宽度成反比。

## 四 收获和体会

在信号的频域分析这一章节，MATLAB 强大的函数处理能力很好地被展现了出来。当需要观察图像或者函数的某一性质随一个参数变换的时候，可以用一个简单的 for 循环，把所有的图像显示在一张图形里面，通过观察，实现目的

# 实验 4 LTI 系统的频域分析

## 一 实验目的

1. 加深对 LTI 系统频率响应基本概念的掌握和理解。
2. 学习和掌握 LTI 系统频率特性的分析方法。

## 二 实验原理

### 1. 连续时间系统的频率响应

系统的频率响应定义为系统单位冲激响应  $h(t)$  的傅里叶变换，即

$$H(\omega) = \int_{-\infty}^{+\infty} h(\tau) e^{-j\omega\tau} d\tau$$

若 LTI 连续时间系统的单位冲激响应为  $h(t)$ , 输入信号  $x(t)$ , 根据系统的时域分析可知系统的零状态响应为

$$y(t) = x(t) * h(t)$$

对等式两边分别求傅里叶变换，根据时域卷积定理得以得到：

$$Y(\omega) = X(\omega)H(\omega)$$

因此，系统的频率响应还可以由系统的零状态响应和输入的傅里叶变换之比得到

$$H(\omega) = Y(\omega) / X(\omega)$$

$H(\omega)$  反应了 LTI 连续时间系统对不同频率信号的响应特性，是系统内在固有的特性，与外部激励无关。 $H(\omega)$  又可以表示为

$$H(\omega) = |H(\omega)| e^{j\theta(\omega)}$$

其中  $|H(\omega)|$  称为系统的幅度响应， $\theta(\omega)$  称为系统的相位响应。

当虚指数信号  $e^{j\omega t}$  作用于 LTI 系统时，系统的零状态响应  $y(t)$  仍为同频率的虚指数信号，即

$$y(t) = e^{j\omega t} H(\omega)$$

由此还可以推导出正弦信号作用在系统上的响应。

对于由下述微分方程描述的 LTI 连续时间系统

$$\sum_{n=0}^N a_n y^{(n)}(t) = \sum_{m=0}^m b_m x^{(m)}(t)$$



其频率响应  $H(j\omega)$  可以表示为下面的  $j\omega$  的有理多项式

$$H(\omega) = \frac{Y(\omega)}{X(\omega)} = \frac{b_M(j\omega)^M + b_{M-1}(j\omega)^{M-1} + \cdots + b_1j\omega + b_0}{a_N(j\omega)^N + a_{N-1}(j\omega)^{N-1} + \cdots + a_1j\omega + a_0}$$

MATLAB 的信号处理工具箱提供了专门的函数 `freqs`，用来分析连续时间系统的频率响应，该函数有下列几种调用格式：

`[h,w]=freqs(b,a)` 计算默认频率范围内 200 个频率点上的频率响应的取样值，这 200 个频率点记录在 `w` 中。

`h=freqs(b,a,w)` `b`、`a` 分别为表示  $H(j\omega)$  的有理多项式中分子和分母多项式的系数向量，`w` 为频率取样点，返回值 `h` 就是频率响应在频率取样点上的数值向量。

`[h,w]=freqs(b,a,n)` 计算默认范围内 `n` 个频率点上的频率响应的取样值，这 `n` 个频率点记录在 `w` 中。

`Freqs(b,a,...)` 这种调用格式不返回频率响应的取样值，而是以对数坐标的方式绘出系统的幅频响应和相频响应。

## 2. 离散时间系统的频率响应

LTI 离散时间系统的频率响应定义为单位抽样响应  $h(n)$  的离散时间傅里叶变换。

$$H(e^{j\Omega}) = \sum_{N=-\infty}^{+\infty} h(n)e^{-j\Omega n}$$

对于任意的输入信号  $x(n)$ ，输入和输出信号的离散时间傅里叶变换有如下关系

$$Y(e^{j\Omega}) = H(e^{j\Omega}) X(e^{j\Omega})$$

因此，系统的频率响应还可以表示为

$$H(e^{j\Omega}) = Y(e^{j\Omega}) / X(e^{j\Omega})$$

当系统输入信号为  $x(n) = e^{j\Omega n}$  时，系统的输出

$$y(n) = e^{j\Omega n} * h(n) = \sum_{k=-\infty}^{+\infty} e^{j\Omega(n-k)} h(k) = e^{j\Omega n} H(e^{j\Omega})$$

由式可知，虚指数信号通过 LTI 离散时间系统后信号的频率不变，信号的幅度由系统频率响应的幅度值确定，所以  $H(e^{j\Omega})$  表示了系统对不同频率信号的衰减量。

一般情况下，离散系统的频率响应  $H(e^{j\Omega})$  是复值函数，可用幅度和相位表示。

$$H(e^{j\Omega}) = |H(e^{j\Omega})| e^{j\theta(\Omega)}$$

其中  $|H(e^{j\Omega})|$  称为系统的幅度响应， $\theta(\Omega)$  称为系统的相位响应。

若 LTI 离散时间系统可以由如下差分方程描述。

$$\sum_{i=0}^N a_i y(n-i) = \sum_{j=0}^M b_j y(n-j)$$

则由式子描述的离散时间系统的频率响应  $H(e^{j\Omega})$  可以表示为  $e^{j\Omega}$  的有理多项式。

$$H(e^{j\Omega}) = \frac{Y(e^{j\Omega})}{X(e^{j\Omega})} = \frac{b_0 + b_1 e^{-j\Omega} + \cdots + b_M e^{-jM\Omega}}{a_0 + a_1 e^{-j\Omega} + \cdots + a_N e^{-jN\Omega}}$$

MATLAB 的信号处理工具箱提供了专门的函数 `freqz`，用来分析连续时间系统的频率响应，该函数有下列几种调用格式：

`[H,w]=freqz(b,a,n)` `b`、`a` 分别为有理多项式中分子和分母多项式的系数向量，返回值 `H` 是频率响应在 0 到  $\pi$  范围内 `n` 个频率等分点上的数值向量，`w` 包含了这 `n` 个频率点。

`[H,w]=freqz(b,a,n,'whole')` 计算  $0 \sim 2\pi$  `n` 个频率点上的频率响应的取样值，这 `n` 个频率点记录在 `w` 中。

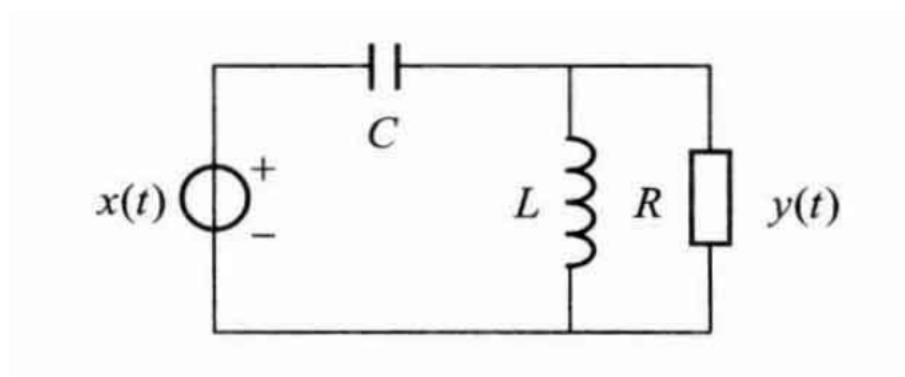
`H=freqz(b,a,w)` `w` 为取样频率点，计算这些频率点上的频率响应的取样值。

`Freqz(b,a,...)` 这种调用格式不返回频率响应的取样值，而是直接绘出系统的频幅响应和相频响应。

### 三 实验内容

#### 1. 已知一个 RLC 电路构造的二阶高通滤波器，已知其中

$$R = \sqrt{\frac{L}{2C}}, L = 0.4H, C = 0.05F$$



##### 1.1 计算该电路系统的频率响应及高通截止频率：

其频率响应为：

$$H(\omega) = \frac{(j\omega)^2}{(j\omega)^2 + 10j\omega + 50}$$

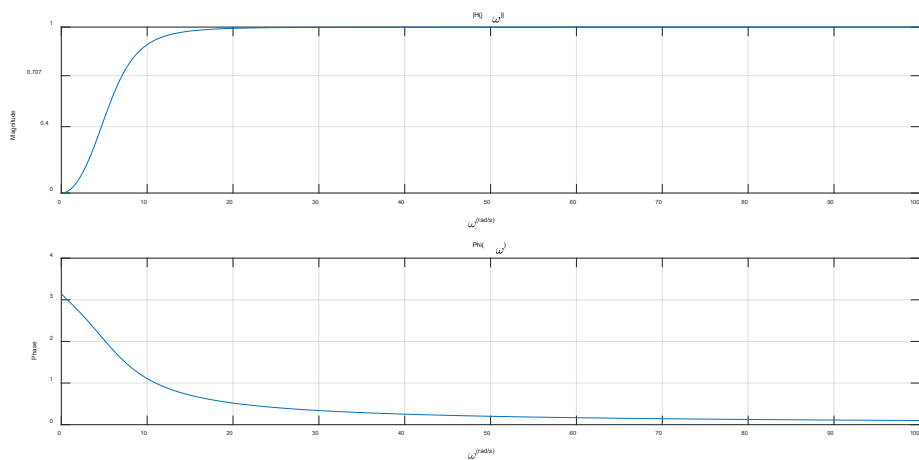
## 1.2 利用 MATLAB 绘制幅度响应和相位响应曲线，比较系统的频率特性与理论计算的结果是否一致。

代码：

```
b = [1 0 0];
a = [1 10 50];
[H w] = freqs(b,a);
subplot(211);
plot(w,abs(H));
%set(gca,'xtick',[0:10]);
set(gca,'ytick',[0 0.4 0.707 1]);
xlabel('\omega(rad/s)');
ylabel('Magnitude');
title('|H(j\omega)|');

grid on;
subplot(212);
plot(w,angle(H));
%set(gca,'ytick',[0 0.4 0.707 1]);
xlabel('\omega(rad/s)');
ylabel('Phase');
title('Phi(\omega)');
grid on;
```

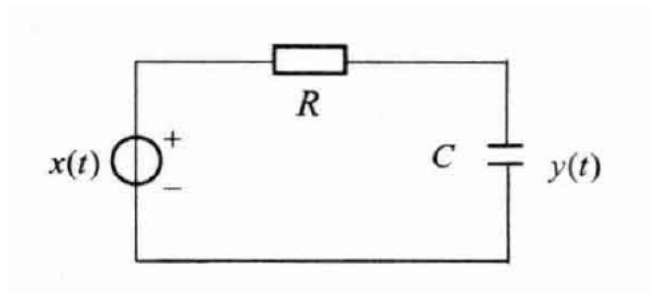
图像：



结论：

对比可知，理论和实际仿真结果一致。

## 2. 已知一个 RC 电路



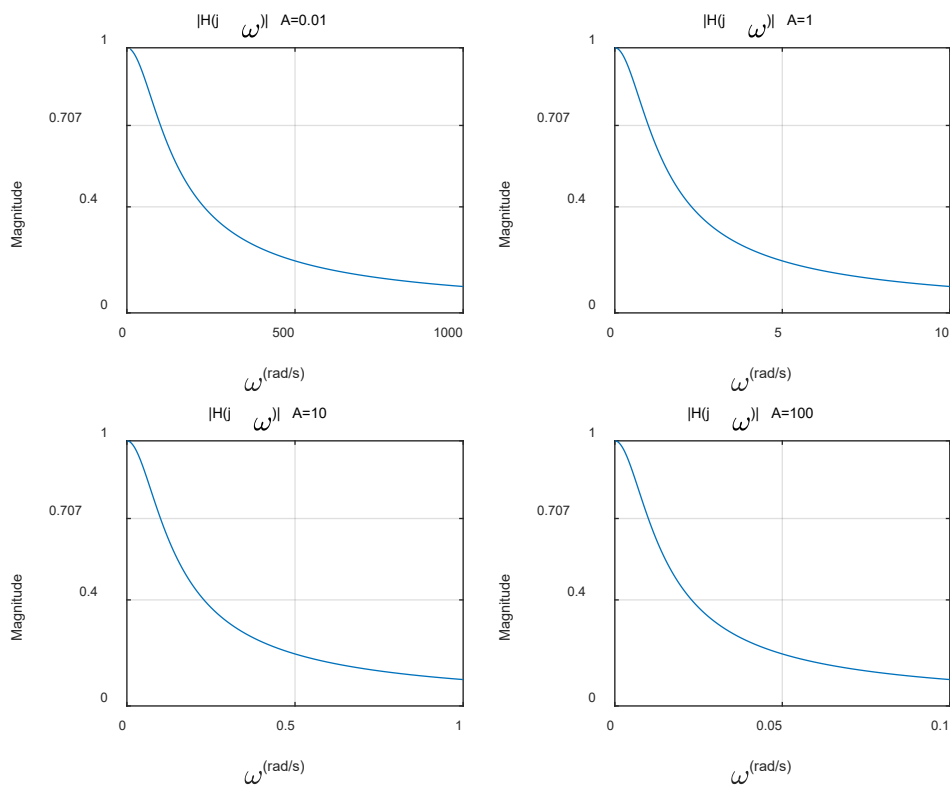
### 2.1 对不同的 RC 值，用 MATLAB 画出系统的幅度响应曲线

$|H(\omega)|$ ，观察实验结果，分析该 RC 实验电路具有什么样的频率特性（高通、低通、带通或带阻）？系统的频率特性随着 RC 值的改变，有何变化规律？

代码：

```
i = 1;
for A = [0.01 1 10 100]
    b=[1];
    a=[A 1];
    [H,w]=freqs(b,a);
    subplot(2,2,i);
    plot(w,abs(H));
    set(gca,'ytick',[0 0.4 0.707 1]);
    xlabel('\omega(rad/s)');
    ylabel('Magnitude');
    title(['|H(j\omega)|' A=' num2str(A)']);
    grid on;
    i = i + 1;
end
```

图像：



结论：

由图可知，注意观察横坐标，RC 电路具有带通特性，随着 RC 值的减小，带通频率增加。

**2.2 系统输入信号  $x(t)=\cos(100t)+\cos(3000t)$ ,  $t=0\sim 0.2\text{s}$ . 该信号包含了一个低频分量和一个高频分量。试确定适当的 RC 值，滤除信号中的高频分量。并绘出滤波前后的时域信号波形及系统的频率响应曲线。**

代码：

```
A = 0.004
b=[1];
a=[A 1]
[H,w]=freqs(b,a);
```

```
subplot(221)
plot(w,abs(H));
set(gca,'ytick',[0 0.4 0.707 1]);
xlabel("\omega(rad/s)");
```

```

ylabel('Magnitude');
title('|H(j\omega)|');
grid on;

t=0:0.0001:0.2
x=cos(100*t)+cos(3000*t);
subplot(222)
plot(t,x);
xlabel('t');

```

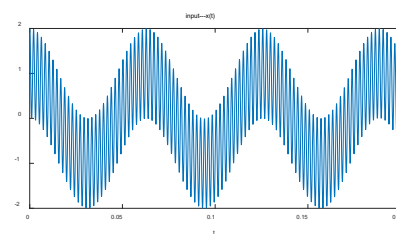
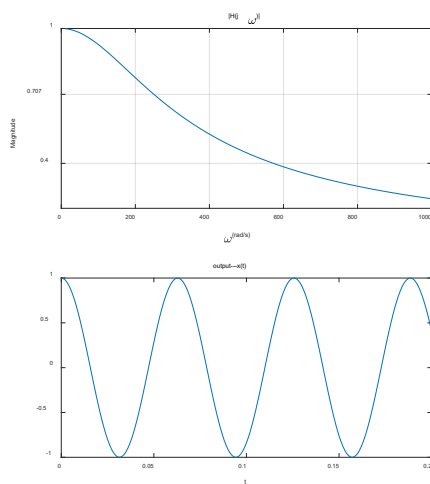
```

title('input---x(t)');

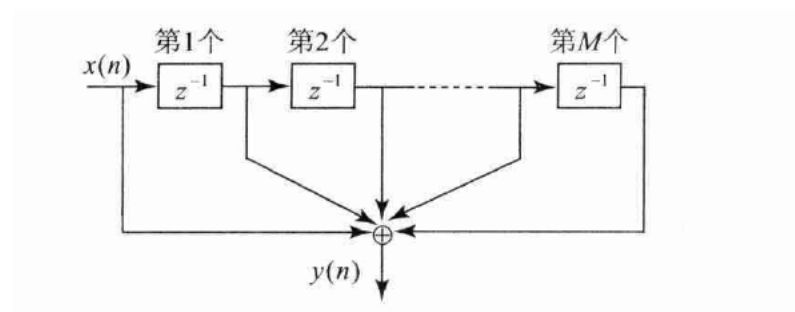
t=0:0.001:0.2;
x=cos(100*t);
subplot(223)
plot(t,x);
xlabel('t');
title('output---x(t)');

```

图像：



### 3. 已知离散系统的系统框图



### 3.1 写出 M=8 时系统的差分方程和系统函数

$$x[n] + x[n-1] + x[n-2] + \dots + x[n-8] = y[n]$$

$$H(z) = 1 + z^{-1} + z^{-2} + \dots + z^{-8}$$

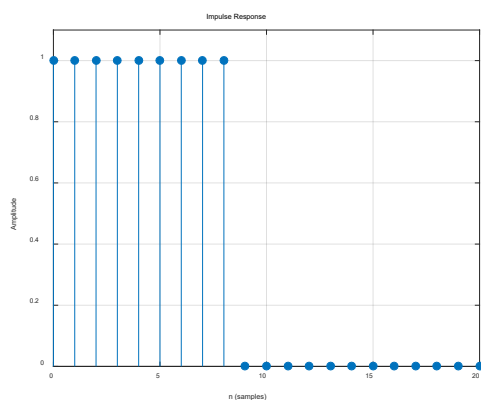
$$h[n] = [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1]$$

### 3.2 利用 MATLAB 计算系统的单位抽样响应

代码：

```
b=[1 1 1 1 1 1 1 1];  
a=[1];  
impz(b,a,0:20);  
axis([0 20 0 1.1]);  
grid on;
```

图像：



### 3.3 试利用 MATLAB 绘出其系统零极点分布图、幅频和相频特性曲线，并分析该系统具有怎样的频率特性。

代码：

%第一个脚本

```
b=[1 1 1 1 1 1 1 1];  
a=[1];  
[H,w]=freqz(b,a);  
subplot(211);
```

```
plot(w/pi,abs(H));  
xlabel('\omega(\pi)');  
ylabel('Magnititude');  
title('|H(e^j\omega)|');
```

```

grid on;
subplot(212);
plot(w/pi,angle(H)/pi);
xlabel('\omega(\pi)');
ylabel('Phase(\pi)');
title('\theta (\omega)');
grid on;

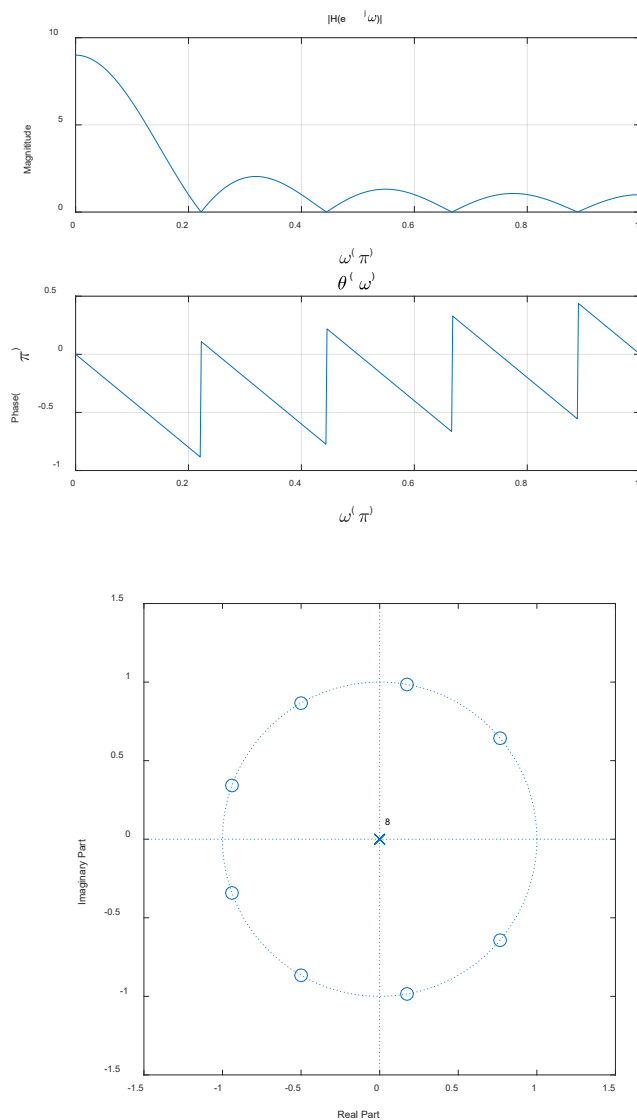
```

```

%第二个脚本
b=[1 1 1 1 1 1 1 1];
a=[1];
zplane(b,a);
axis([-1.5 1.5 -1.5 1.5]);

```

图像：



结论：

由图可知，该系统具有高通特性，即通高频，阻低频。



4. 已知一离散时间 LTI 系统的频率响应  $H(e^{j\Omega})$  已知，输入信号为  $x(n)=\cos(0.3\pi n)+0.5\cos(0.8\pi n)$ 。试根据式子分析正弦信号  $\sin(\Omega_0 t)$  通过频率响应为  $H(\quad)$  的离散时间系统的响应，并根据分析结果计算系统对于  $x(n)$  的响应  $y(n)$ ，用 MATLAB 绘出系统的输入与输出波形。观察实验结果，分析该系统具有什么样的频率特性（高通，低通，带通和带阻）？从输入输出信号上怎么反映出系统的频率特性？

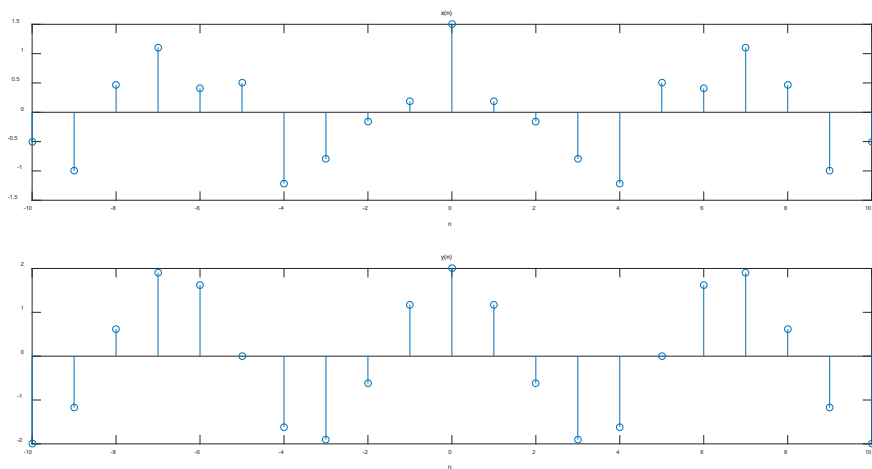
#### 4.1

代码：

```
subplot(211);
n=-10:10;
x=cos(0.3*pi*n)+0.5*cos(0.8*pi*n);
stem(n,x);
xlabel('n');
title('x(n)');

subplot(212);
x=cos(0.3*pi*n)+0.5*cos(0.8*pi*n);
y=2*cos(0.3*pi*n);
stem(n,y);
xlabel('n');
title('y(n)');
```

图像：



**结论：**

由图可知，该系统具有带通的频率特性

## **四 收获和体会**

在信号的 LTI 系统的频域分析这一章节，MATLAB 绘制零极点图等手绘十分复杂的图像十分方便。无论是连续时间系统还是离散时间系统，MATLAB 都能很容易绘制出理想的图形。虽然在做实验时，零极点分布图还没有很好地掌握，但是我相信提前掌握用 MATLAB 进行仿真实验很利于以后的学习。

# 实验 5 连续系统的复频域分析

## 一 实验目的

1. 掌握拉普拉斯变换及其反变换的定义，并掌握 MATLAB 实现方法。
2. 学习和掌握连续时间系统函数的定义及复频域分析方法。
3. 掌握系统零极点的定义，加深理解系统零极点分布与系统特性的关系。

## 二 实验原理

### 1. 拉普拉斯变换

连续时间信号  $x(t)$  的拉普拉斯变换定义为

$$X(s) = \int_{-\infty}^{+\infty} x(t)e^{-st} dt \dots\dots\dots(1)$$

拉普拉斯反变换定义为

$$x(t) = \frac{1}{2\pi j} \int_{\sigma-j\infty}^{\sigma+j\infty} X(s)e^{st} ds \dots\dots\dots(2)$$

在 MATLAB 中，可以采用符号数学工具箱的 laplace 函数和 ilaplace 函数进行拉氏变换和反拉氏变换。

L=laplace(F)符号表达式 F 的拉氏变换，F 中时间变量为 t，返回变量为 s 的结果表达式。

L=laplace(F,t)用 t 替换结果中的变量 s。

F=ilaplace(L)以 s 为变量的符号表达式 L 的拉氏反变换，返回时间变量为 t 的结果表达式。

F=ilaplace(L,x)用 x 替换结果中的变量 t。

除了上述 ilaplace 函数，还可以采用部分分式法，求解拉普拉斯逆变换，具体原理如下：

当  $X(s)$  为有理分式时，它可以表示为两个多项式之比：

$$X(s) = \frac{N(s)}{D(s)} = \frac{b_M s^M + b_{M-1} s^{M-1} + \dots + b_0}{a_N s^N + a_{N-1} s^{N-1} + \dots + a_0} \dots\dots\dots(3)$$

式 (3) 可以用部分分式法展成一下形式

$$X(s) = \frac{r_1}{s - p_1} + \frac{r_2}{s - p_2} + \dots + \frac{r_N}{s - p_N} \dots\dots\dots(4)$$

通过查常用拉普拉斯变换对，可以由式 (1-2) 求得拉普拉斯逆变换。

利用 MATLAB 的 residue 函数可以将  $X(s)$  展成式 (1-2) 所示的部分分式展开式, 该函数的调用格式为:  $[r,p,k] = \text{residue}(b,a)$  其中  $b$ 、 $a$  为分子和分母多项式系数向量,  $r$ 、 $p$ 、 $k$  分别为上述展开式中的部分分式系数、极点和直项多项式系数。

## 2. 连续时间系统的系统函数

连续时间系统的系统函数是系统单位冲激响应的拉氏变换

$$H(s) = \int_{-\infty}^{+\infty} h(t)e^{-st} dt \dots\dots\dots (5)$$

此外, 连续时间系统的系统函数还可以由系统输入和系统输出信号的拉氏变换之比得到

$$H(s) = Y(s) / X(s) \dots\dots\dots (6)$$

单位冲激响应  $h(t)$  反映了系统的固有性质, 而  $H(s)$  从复频域反映了系统的固有性质。由式 (6) 描述的连续时间系统, 其系统函数为  $s$  的有理函数

$$H(s) = \frac{b_M s^M + b_{M-1} s^{M-1} + \dots + b_0}{a_N s^N + a_{N-1} s^{N-1} + \dots + a_0} \dots\dots\dots (7)$$

## 3. 连续时间系统的零极点分析

系统的零点是指式 (7) 的分子多项式为零的点, 极点指使分母多项式为零的点, 零点使系统的值为零, 极点使系统函数的值无穷大。通常将系统函数的零极点绘在  $s$  平面上, 零点用  $\circ$  表示, 极点用  $\times$  表示, 这样得到的图形称为零极点的分布图。

由零极点的定义可知, 零点和极点分别指式 (7) 的分子多项式和分母多项式的根。利用 MATLAB 求多项式的根可以通过函数 roots 来实现, 该函数的调用格式为:

$r = \text{roots}(c)$   $c$  为多项式的系数向量, 返回值  $r$  为多项式的根向量。

分别对式 (7) 的分子多项式和分母多项式求根即可得到零极点。

此外, 在 MATLAB 中还提供了更简便的方法来求取零极点和绘制系统函数的零极点分布图, 即利用 pzmap 函数, 该函数的调用格式为:

$\text{pzmap}(\text{sys})$  绘出由系统模型  $\text{sys}$  描述的系统的零极点分布图。

$[p,z] = \text{pzmap}(\text{sys})$  这种调用方法返回极点和零点, 而不绘出零极点分布图。其中  $\text{sys}$  为系统传函模型, 由  $t$  命令  $\text{sys} = \text{tf}(b,a)$  实现,  $b$ 、 $a$  为传递函数的分子多项式和分母多项式的系数向量。

MATLAB 还为用户提供了两个专用函数  $\text{tf2zp}$  和  $\text{zp2tf}$  来实现系统传递函数模型和零极点增益模型的转换, 其调用格式为:

$$[z,p,k] = \text{tf2zp}(b,a)$$

$$[b,a] = \text{zp2tf}(z,p,k)$$

其中  $b$ 、 $a$  为传递函数的分子多项式和分母多项式的系数向量, 返回值  $z$  为零点列向量,  $p$  为极点列向量,  $k$  为系统函数零极点形式的增益。

### 三 实验内容

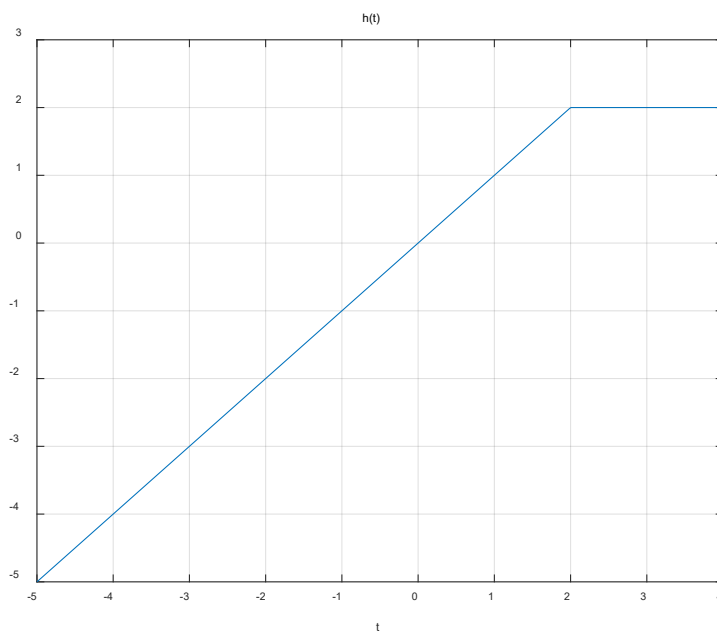
1. 已知系统的冲激响应  $h(t) = u(t) - u(t-2)$ , 输入信号  $x(t) = u(t)$ , 是采用复频域的方法求解系统的响应, 编写 MATLAB 程序实现。

代码:

```
syms t h x y H X Y;
h = heaviside(t) - heaviside(t - 2);
x = heaviside(t);
H = laplace(h);
X = laplace(x);
Y = X*H;

y = ilaplace(Y);
disp(y);
ezplot(y,[-5,4]);
title('h(t)');
grid on;
axis([-5 4 -5 3]);
```

图像:



结论:

由 MATLAB 显示及验证可知,  $y(t) = t - (t-2)u(t-2)$

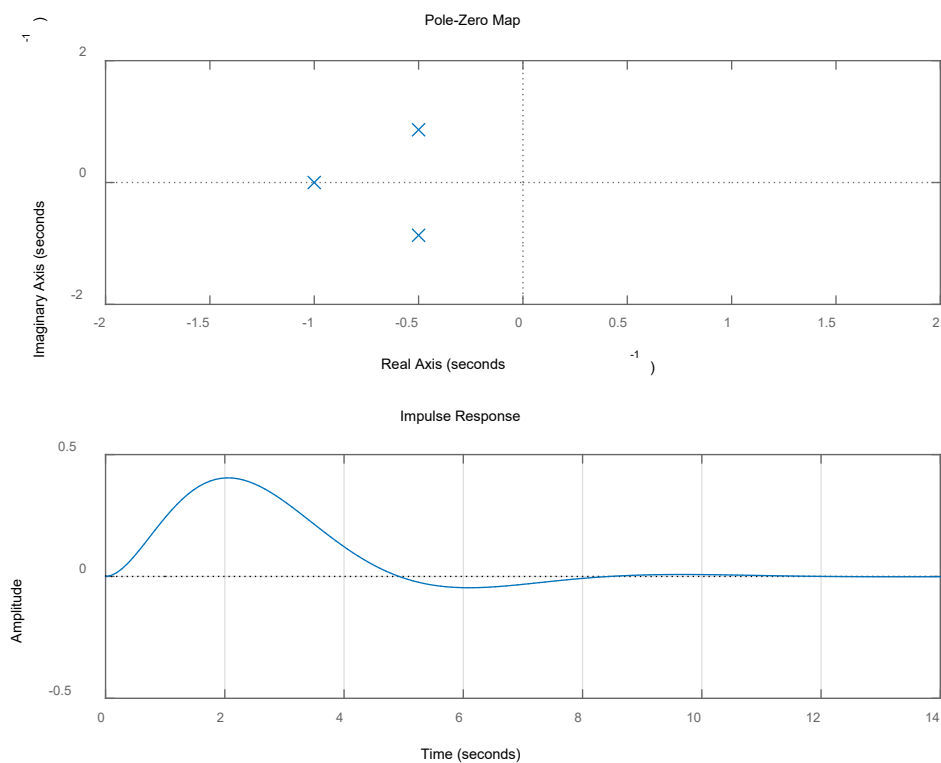
2. 已知因果连续时间系统的系统函数分别如下，试采用 MATLAB 画出其零极点分布图，求解系统的冲激响应  $h(t)$  和频率响应  $H(\omega)$  并判断系统是否稳定

$$2.1 \quad H(s) = \frac{1}{s^3 + 2s^2 + 2s + 1}$$

代码：

```
syms H s;                                axis([-2,2,-2,2]);
b = 1;
a = [1,2,2,1];                            subplot(212);
H = tf(b,a);                             impulse(H);
subplot(211);                             grid on;
pzmap(H);
```

图像：



结论：

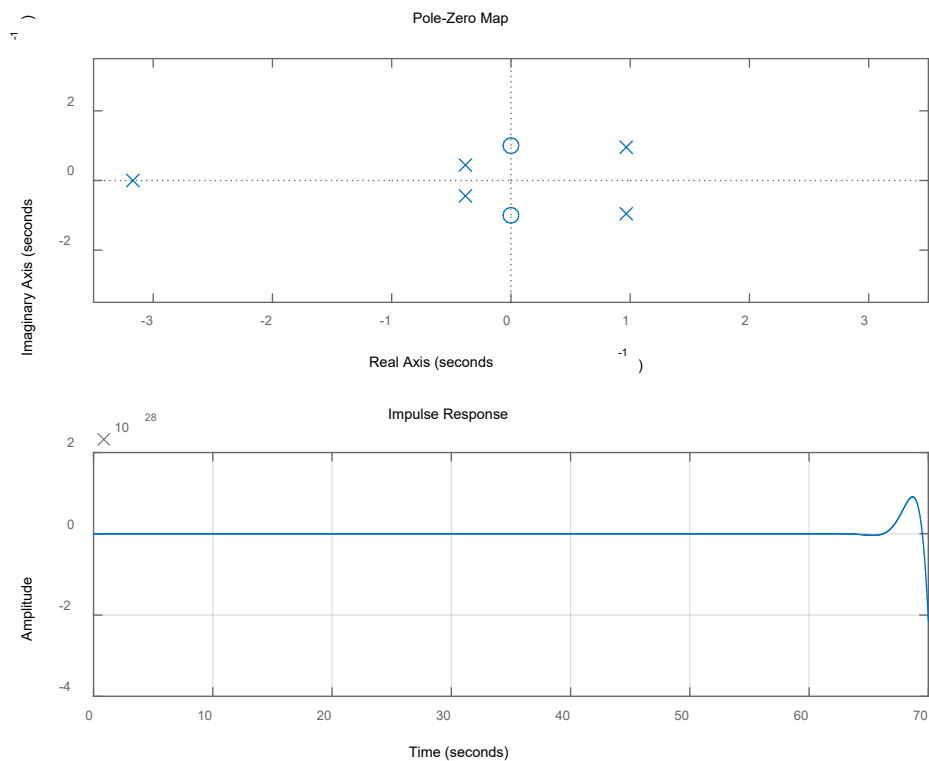
该因果系统的所有极点都位于 S 平面的左半平面，所以系统是稳定的。

$$2.2 \quad H(s) = \frac{s^2 + 1}{s^5 + 2s^4 - 3s^3 + 3s^2 + 3s + 2}$$

代码：

```
b = [1,0,1];  
a = [1,2,-3,3,3,2];  
H = tf(b,a);  
  
subplot(211);  
pzmap(H);  
  
axis([-3.5,3.5,-3.5,3.5]);  
  
subplot(212);  
impz(H);  
grid on;
```

图像：



结论：

该因果系统的所有极点不全位于 S 平面的左半平面，所以系统是不稳定的。

3. 已知连续时间系统函数的极点位置分别如下所示（设系统无零点），试用 MATLAB 绘制 6 中不同情况下，系统函数的零极点分布图，并绘制相应冲激响应的时域波形，观察并分析系统函数极点位置对冲激响应时域特性的影响

### 3.1-3.3

代码：

```
clear all;
z = []; %根据条件系统无零点
p = [0,-2,2]; %极点
%p = p(2)
p_1 = 0
```

```
for i = 1:3
j = 2*i-1;
z=[];
```

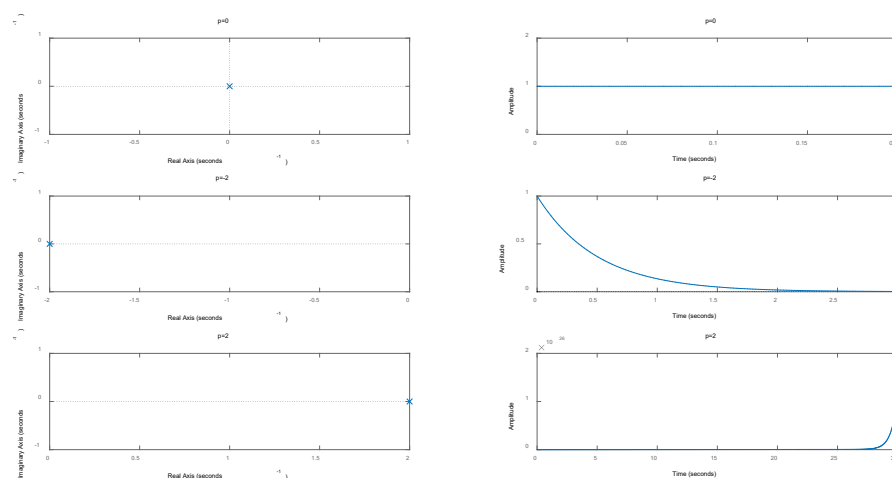
```
p_1 = p(i);
k = [1];
```

```
[b,a] = zp2tf(z,p_1,k);
sys = tf(b,a);
subplot(3,2,j);
pzmap(sys); %绘制零极点图
title(['p=' num2str(p_1)])
```

```
subplot(3,2,j+1);
impz(sys); %绘制单位冲激响应
title(['p=' num2str(p_1)])
```

```
end
```

图像：





### 3.4-3.6

代码：

```
clear all;  
z = []; %根据条件系统无零点  
p = [2j,-2j,-1+4j,-1-4j,1+4j,1-4j]; %极点  
%p = p(2)  
p_1 = 0
```

```
for i = 1:3
```

```
    j = 2*i-1;
```

```
    z=[];
```

```
    p_1 = p(j:j+1)
```

```
    k = [1];
```

```
    [b,a] = zp2tf(z,p_1,k);
```

```
    sys = tf(b,a);
```

```
    subplot(3,2,j);
```

```
    pzmap(sys); %绘制零极点图
```

```
    title(['p=' num2str(p_1)])
```

```
    subplot(3,2,j+1);
```

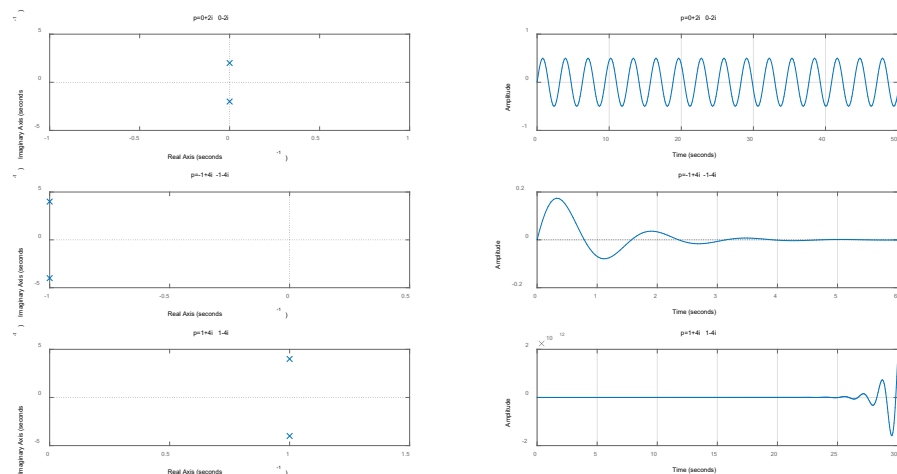
```
    impulse(sys); %绘制单位冲激响应
```

```
    title(['p=' num2str(p_1)])
```

```
    grid on;
```

```
end
```

图像：



结论：

当极点唯一且在原点时， $h(t)$ 为常数；

当极点唯一且是负实数时， $h(t)$ 为递减的指数函数；

当极点唯一且是正实数时， $h(t)$ 为递增的指数函数；

当 $H(s)$ 有两个互为共轭的极点时， $h(t)$ 有一  $\sin t$  因子；

当 $H(s)$ 有两个互为共轭的极点且他们位于右半平面时， $h(t)$ 还有一  $e^t$  因子；

当  $H(s)$  有两个互为共轭的极点且他们位于左半平面时,  $h(t)$  还有一  $e^{-t}$  因子;

4. 用 MATLAB 分别绘制三个系统的零极点分布图及相应冲激响应的时域波形, 观察并分析系统函数零点位置对冲激响应时域特性的影响。

$$(1)H(s)=\frac{1}{s^2+s+17}$$

$$(2)H(s)=\frac{s+8}{s^2+s+17}$$

$$(3)H(s)=\frac{s-8}{s^2+s+17}$$

代码:

```
b_0=[1,0,1,8,1,-8];
```

```
a=[1 2 17];
```

```
for i = 1:3
```

```
    j = 2*i-1;
```

```
    b = b_0(j:j+1)
```

```
    sys=tf(b,a);
```

```
    subplot(3,2,j);
```

```
    pzmap(sys);%求零极点图
```

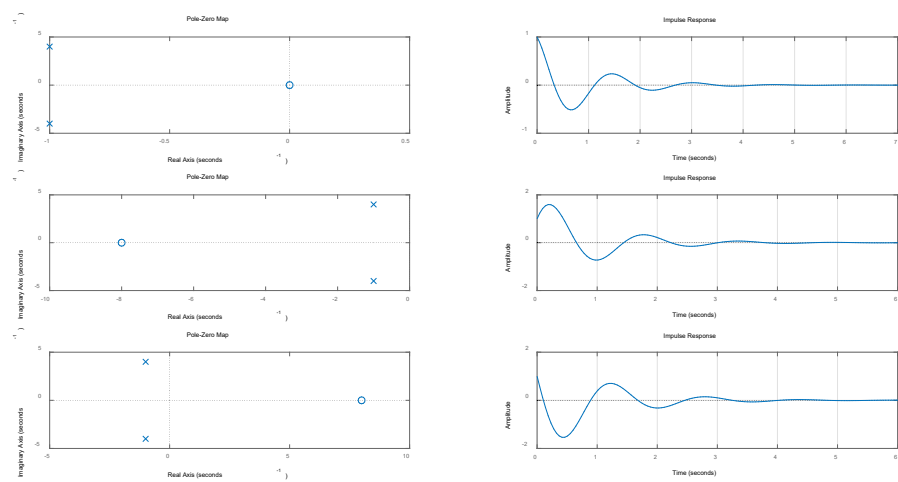
```
    subplot(3,2,j+1);
```

```
    impulse(b,a);%求冲击响应
```

```
    grid on;
```

```
end
```

图像：



结论：

当极点不变时，零点分布只影响系统时域响应的幅度和相位，对时域响应模式没有影响。即不会改变是衰减振荡还是增长振荡

## 四 收获和体会

每次遇到信号不会的题目就会拿 MATLAB 把题目解一遍，然后解着解着就知道怎么做原来的题目了，在我看来 MATLAB 是信号处理的一大利器。熟练掌握 MATLAB 有利于从另一个层面来思考问题。

# 实验 6 离散系统的 Z 域分析

## 一 实验目的

1. 掌握 z 变换及其反变换的定义，并掌握 MATLAB 实现方法。
2. 学习和掌握离散时间系统系统函数的定义及 z 域分析方法。
3. 掌握系统零极点的定义，加深理解系统零极点分布与系统特性的关系。

## 二 实验原理

### 1. Z 变换

序列 $x(n)$ 的z变换定义为

$$X(z) = \sum_{n=-\infty}^{+\infty} x(n)z^{-n}$$

Z反变换定义为

$$x(n) = \frac{1}{2\pi j} \oint_r X(z)z^{n-1}dz$$

在MATLAB中，可以采用符号数学工具箱的ztrans函数和iztrans函数计算z变换和z反变换：

$Z=ztrans(F)$  求符号表达式F的z变换。

$F=ilaplace(Z)$  求符号表达式 Z 的 z 反变换。

### 2. 离散时间系统的系统函数

离散时间系统的系统函数 $H(z)$ 定义为单位抽样响应 $h(n)$ 的z变换

$$H(z) = \sum_{n=-\infty}^{+\infty} h(n)z^{-n}$$

此外，连续时间系统的系统函数还可以由系统输入和输出信号的z变换之比得到

$$H(z) = Y(z) / X(z)$$

由上式描述的离散时间系统的系统函数可以表示为

$$H(z) = \frac{b_0 + b_1z^{-1} + \cdots + b_Mz^{-M}}{a_0 + a_1z^{-1} + \cdots + a_Nz^{-N}}$$

### 3. 离散时间系统的零极点分析

离散时间系统的零点和极点分别指使系统函数分子多项式和分母多项式为零的点。在MATLAB中可以通过函数roots来求系统函数分子多项式和分母多项式的根，从而得到系统的零极点。

此外，还可以利用MATLAB的zplane函数来求解和绘制离散系统的零极点分布图，zplane函数调用格式为：

zplane(b,a) b,a为系统函数的分子、分母多项式的系数向量（行向量）。

zplane(z,p) z,p为零极点序列（列向量）。

系统函数是描述系统的重要物理量，研究系统函数的零极点分布不仅可以了解系统单位抽样响应的变化，还可以了解系统的频率特性响应以及判断系统的稳定性：

- 1) 系统函数的极点位置决定了系统单位抽样响应  $h(n)$  的波形，系统函数零点位置只影响冲激响应的幅度和相位，不影响波形。
- 2) 系统的频率响应取决于系统的零极点，根据系统的零极点分布情况，可以通过向量分析系统的频率响应。
- 3) 因果的离散时间系统稳定的充要条件是  $H(z)$  的全部极点都位于单位圆内。

## 三 实验内容

### 1. 已知因果离散时间系统的系统函数分别为：

$$H(z) = \frac{z^2 + 2z + 1}{z^3 - 0.5z^2 - 0.005z + 0.3}$$

$$H(z) = \frac{z^3 - z^2 + 2}{3z^4 + 3z^3 - z^2 + 3z - 1}$$

试采用 MATLAB 画出其零极点分布图，求解系统的冲激响应  $h(n)$  和频率响应

$H(e^{j\Omega})$ ，并判断系统是否稳定

代码：

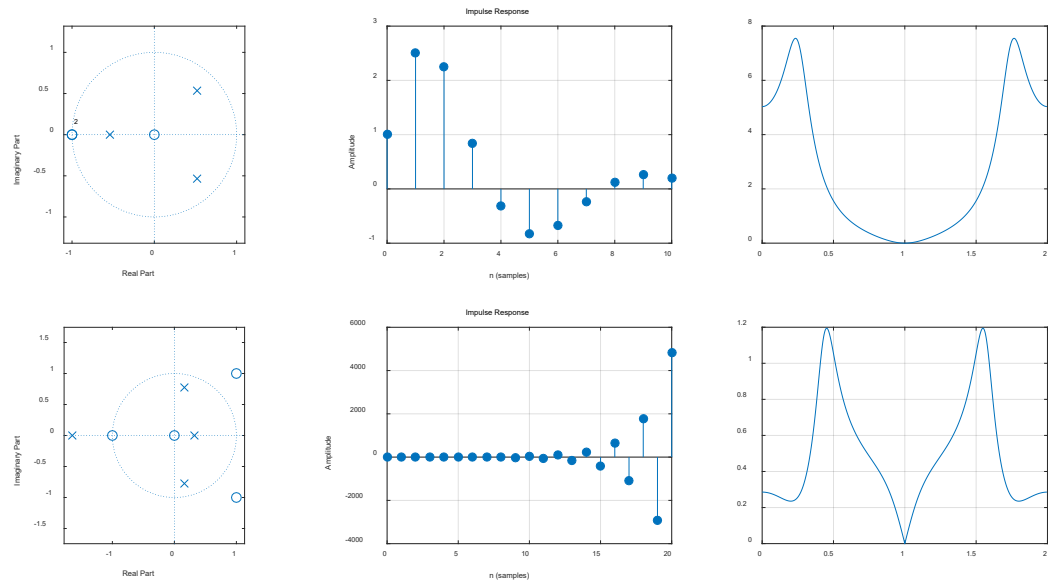
```
b=[1 2 1];
a=[1 -0.5 -0.005 0.3];
subplot(231);
zplane(b,a);
subplot(232);
impz(b,a,0:10);
grid on;
subplot(233);
[H,w]=freqz(b,a,'whole');
plot(w/pi,abs(H));
grid on;

b=[1 -1 0 2];
a=[3 3 -1 3 -1];
subplot(234);
zplane(b,a);
subplot(235);
impz(b,a,0:20);
```

```
grid on;
subplot(236);
[H,w]=freqz(b,a,'whole');
```

```
plot(w/pi,abs(H));
grid on;
```

图像：



结论：

- 1) 对于第一个函数，当 t 趋近于无穷时，响应趋近于零，所以该系统是稳定的。
- 2) 对于第二个函数，当时间趋近无穷时，该系统的响应是发散的，故，该系统不是稳定系统

2. 已知因果连续时间系统的系统函数分别如下，试采用 MATLAB

画出其零极点分布图，求解系统的冲激响应  $h(t)$  和频率响应  $H(\omega)$  并

判断系统是否稳定

序号	零点(z)	极点(p)
1	0	0.25
2	0	1
3	0	-1.25
4	0	$0.8e^{j\frac{\pi}{6}}, 0.8e^{-j\frac{\pi}{6}}$
5	0	$e^{j\frac{\pi}{8}}, e^{-j\frac{\pi}{8}}$

6	0	$1.2e^{j\frac{3\pi}{4}} \setminus 0.8e^{-j\frac{3\pi}{4}}$
---	---	--

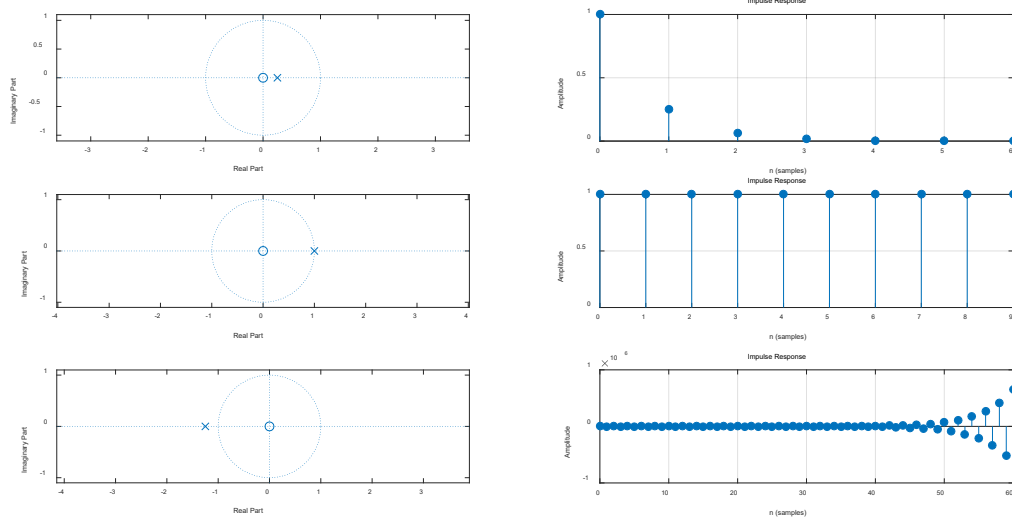
## 2.1 – 2.3

代码：

```
clear all;
p_1 = [0.25,1,-1.25];
b = [1 0];
for i = 1:3
j = 2*i-1;
p=p_1(i);
a=poly(p);
```

```
sys = tf(b,a)
subplot(3,2,j);
zplane(b,a);
subplot(3,2,j+1);
impz(b,a);
grid on;
end
```

图像：



## 2.4 – 2.6

代码：

```
clear all;
p_2 = [0.8*exp(j*pi/6) 0.8*exp(-j*pi/6)
exp(j*pi/8) exp(-j*pi/8) 1.2*exp(3*j*pi/4)
1.2*exp(-3*j*pi/4)]
```

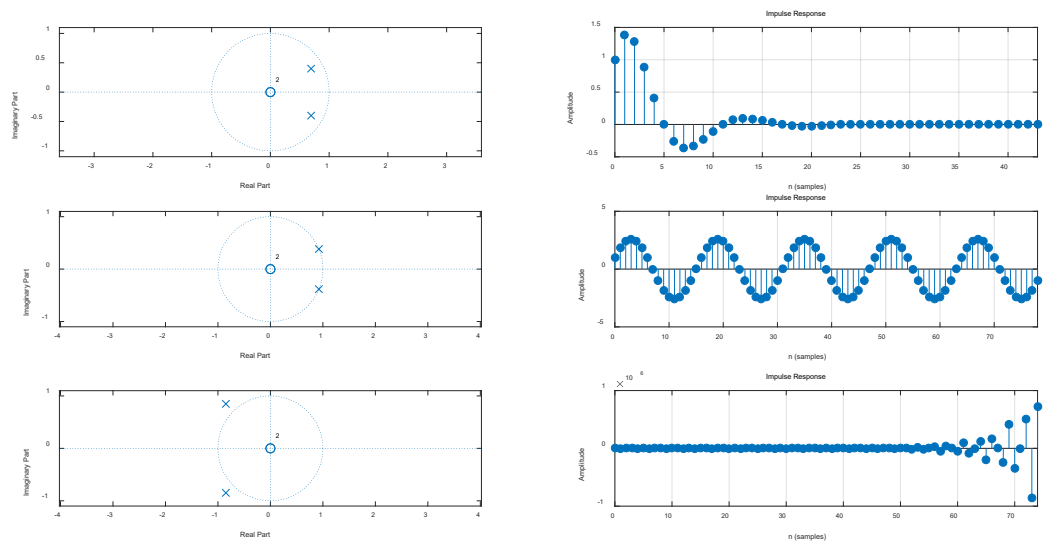
```
b = [1 0];
for i = 1:3
b = [1 0];
```

```
j = 2*i-1;
p=p_2(j:j+1);
```

```
a=poly(p);
sys = tf(b,a)
subplot(3,2,j);
```

```
zplane(b,a);
subplot(3,2,j+1);
impz(b,a);
grid on;
end
```

图像：



结论：

如果只有一个极点，响应波形为指数型。当极点在单位圆内时，呈指数衰减；当极点在单位圆上，为定值（单位阶跃响应）；当极点在单位圆外时，呈指数增长。如果有一对共轭极点，响应为振荡型。当极点在单位圆内，呈衰减振荡；当极点在单位圆上，为等幅振荡；当极点在单位圆外时，呈增幅振荡。



3. 以下两个系统具有相同的极点，但零点不同，用 MATLAB 分别绘制两个系统的零极点分布图及相应单位抽样响应的时域波形，观察分析系统函数零点位置对单位抽样响应时域特性的影响。

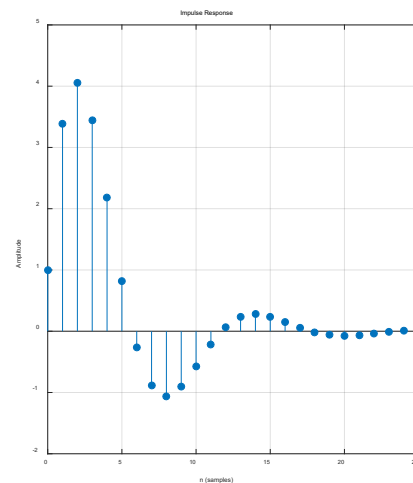
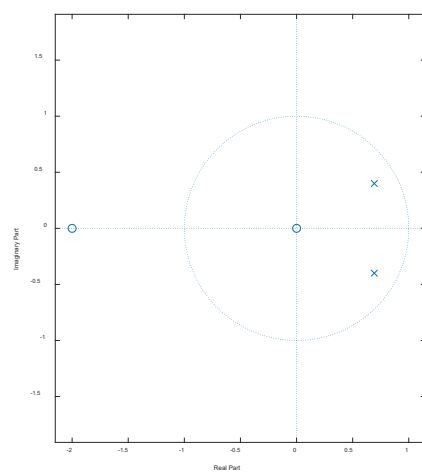
$$3.1 \quad H(z) = \frac{z(z+2)}{(z-0.8e^{j\frac{\pi}{6}})(z-0.8e^{-j\frac{\pi}{6}})}$$

代码：

```
clear all;
b=[1 2 0];
a=poly([0.8*exp(j*pi/6) 0.8*exp(-j*pi/6)]);
sys = tf(b,a);
subplot(121);

zplane(b,a);
subplot(122);
impz(b,a,0:25);
grid on;
```

图像：



$$3.2 \quad H(z) = \frac{z(z-2)}{(z-0.8e^{j\frac{\pi}{6}})(z-0.8e^{-j\frac{\pi}{6}})}$$

代码：

```
clear all;
b=[1 -2 0];
a=poly([0.8*exp(j*pi/6) 0.8*exp(-j*pi/6)]);
sys = tf(b,a);
subplot(121);

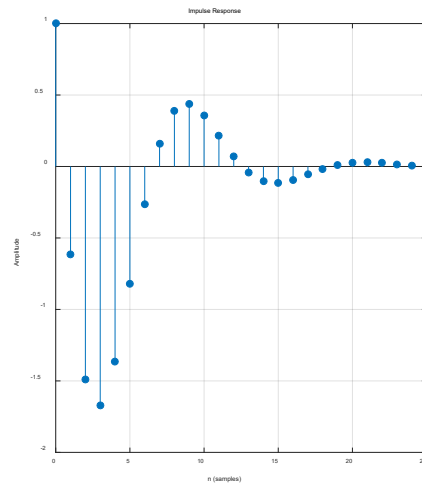
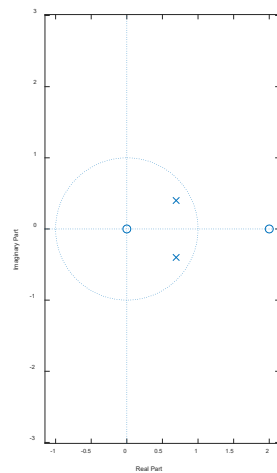
zplane(b,a);
subplot(122);
impz(b,a,0:25);
grid on;
```

```
sys = tf(b,a);
subplot(121);
zplane(b,a);
```

```
subplot(122);
impz(b,a,0:25);
```

```
grid on;
```

图像：



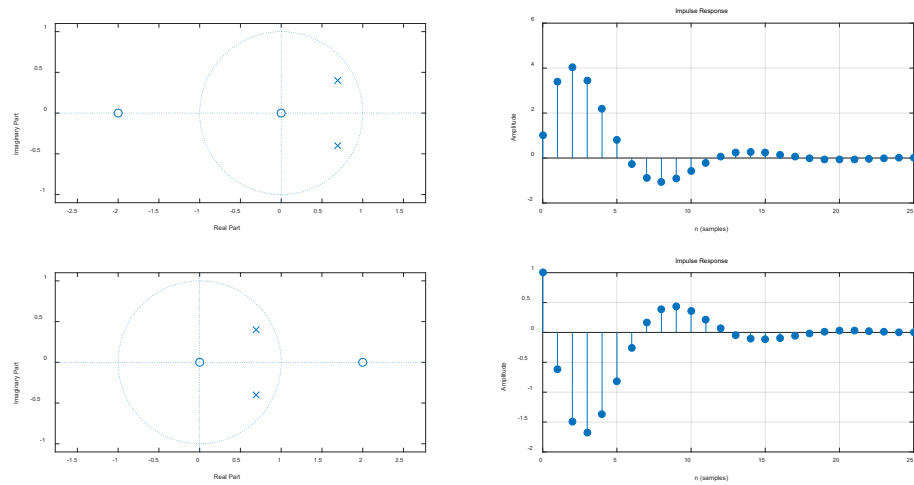
结论：

代码：

```
clear all;
b=[1 2 0];
a=poly([0.8*exp(j*pi/6) 0.8*exp(-j*pi/6)]);
sys = tf(b,a);
subplot(221);
zplane(b,a);
subplot(222);
impz(b,a,0:25);
```

```
grid on;
b=[1 -2 0];
sys = tf(b,a);
subplot(223);
zplane(b,a);
subplot(224);
impz(b,a,0:25);
grid on;
```

图像：



总结：

当零点沿纵轴对折时，时域波形图近似沿横轴对折。但只是波形的对称，各点的值并没有等幅改变。

## 四 收获和体会

本次实验加强了我对  $z$  变换的理解以及  $z$  域分析法的理解，同时不得不再一次感叹 MATLAB 的强大。在以后的学习过程中我还会进一步加强 MATLAB 的熟练使用，理论与实践相结合，学好信号与系统以及接下来的科目。