

实验 7 离散 LSI 系统的时域响应

1518 班 15352408 张镓伟

一、实验目的

- (1)加深对离散 LSI 系统时域特性的认识。
- (2)掌握 MATLAB 求解离散时间系统响应的基本方法。
- (3)了解 MATLAB 中求解系统响应的子函数及其应用方法。

二、实验涉及的 MATLAB 子函数

1.dlsim

功能：求解离散系统的响应。

调用格式：

y=dlsim(b, a, x); 求输入信号为 x 时系统的响应。

说明：b 和 a 分别表示系统函数 H(z)中，由对应的分子项和分母项系数所构成的数组。

三、实验原理

1.离散 LSI 系统时域响应的求解方法

一个线性移不变离散系统可以用线性常系数差分方程(式(5-1))表示，也可以用系统函数(式(5-2))表示。无论是差分方程还是系统函数，一旦式中的系数 b_m 和 a_k 的数据确定了，则系统的性质也就确定了。因此，在程序编写时，往往只要将系数 b_m 和 a_k 列写成数组，然后调用相应的处理子函数，就可以求出系统的响应。

$$\sum_{k=0}^N a_k y(n-k) = \sum_{m=0}^M b_m x(n-m) \quad (5-1)$$

$$\begin{aligned} H(z) &= \frac{Y(z)}{X(z)} = \frac{b(z)}{a(z)} = \frac{\sum_{m=0}^M b_m z^{-m}}{\sum_{k=0}^N a_k z^{-k}} \\ &= \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_m z^{-m}}{1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_k z^{-k}} \end{aligned} \quad (5-2)$$

对于离散 LSI 系统的响应，MATLAB 为我们提供了多种求解方法：

- (1)用 conv 子函数进行卷积积分，求任意输入的系统零状态响应。
- (2)用 dlsim 子函数求任意输入的系统零状态响应。
- (3)用 filter 和 filtic 子函数求任意输入的系统完全响应。

本实验重点介绍(2)、(3)两种方法。

2.用 dlsim 子函数求 LSI 系统对任意输入的响应

对于离散 LSI 系统任意输入信号的响应，可以用 MATLAB 提供的仿真 dlsim 子函数来求解。

例 6-1 已知一个 IIR 数字低通滤波器的系统函数公式为

$$H(z) = \frac{0.1321 + 0.3963 z^{-1} + 0.3963 z^{-2} + 0.1321 z^{-3}}{1 - 0.34319 z^{-1} + 0.60439 z^{-4} - 0.20407 z^{-3}}$$

输入两个正弦叠加的信号序列

$$x = \sin\left(\frac{n}{2}\right) + \frac{1}{3} \sin(10n)$$

求该系统的响应。

编写MATLAB程序如下：

```
nx=0: 8*pi;  
x=sin(nx/2)+sin(10*nx)/3; %产生输入信号序列  
subplot(3, 1, 1); stem(nx, x);  
a= [1, -0.34319, 0.60439, -0.20407] ; %输入系统函  
                                     %数的系数  
b= [0.1321, 0.3963, 0.3963, 0.1321] ;  
nh=0: 9;  
h=impz(b, a, nh); %求系统的单位冲激响应  
subplot(3, 1, 2); stem(nh, h);  
y=dlsim(b, a, x); %求系统的响应  
subplot(3, 1, 3); stem(y);
```

程序执行的结果如图6-1所示。

从系统的输出响应 $y(n)$ 可以看出，原输入序列中的高频信号部分通过低通滤波器后已被滤除，只剩下频率较低的 $\sin(n/2)$ 分量。

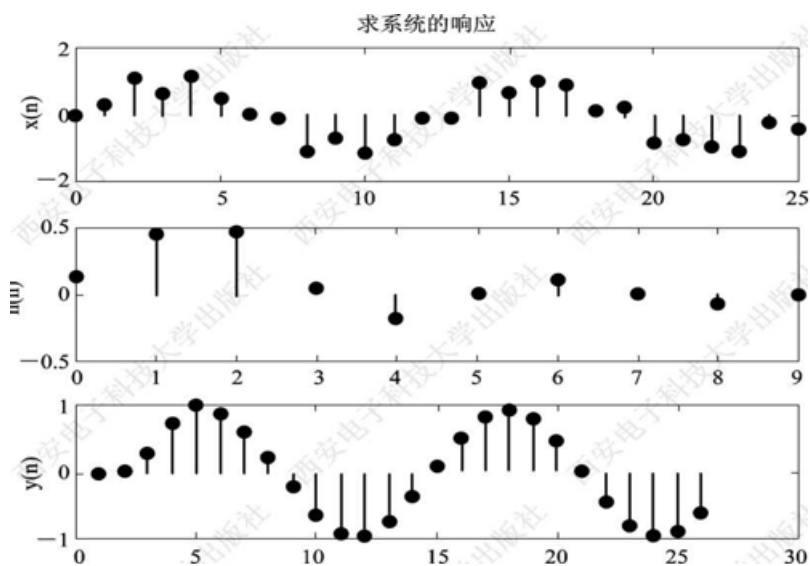


图6-1 例6-1 $x(n)$ 、 $h(n)$ 、 $y(n)$ 的波形

3.用 filter 和 filter 子函数求 LSI 系统对任意输入的响应

filter 和 filter 子函数采用递推法进行系统差分方程的求解，可以用于求解离散 LSI 系统对任意输入的完全响应。当输入信号为单位冲激信号或单位阶跃信号时，求得的响应即为系统的单位冲激响应或单位阶跃响应。

本实验则使用任意输入序列 $x(n)$ ，求系统的完全响应。

例 6-2 已知一个 LSI 系统的差分方程为

$$y(n]=0.9y(n-1)+x(n)+0.9x(n-1)$$

满足初始条件 $y(-1)=0$, $x(-1)=0$, 求系统输入为 $x(n)=e^{-0.05+j0.4n}u(n-2)$ 时的响应 $y(n)$ 。

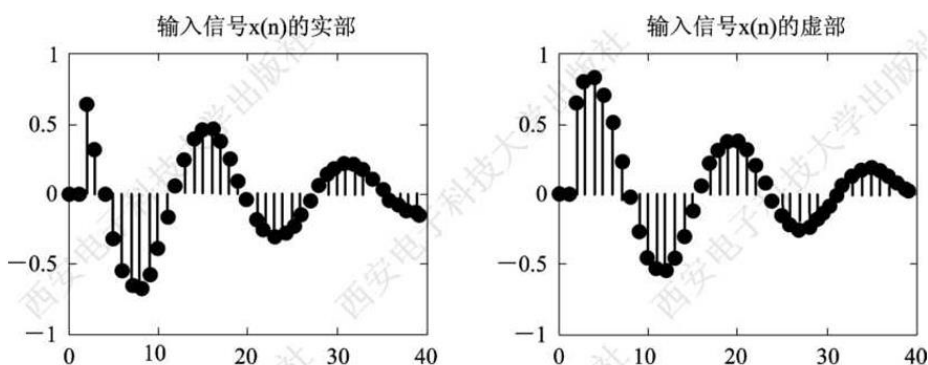
解 将上式整理后得到:

$$y(n]-0.9y(n-1)=x(n)+0.9x(n-1)$$

由上式可列写出其 b_m 和 a_k 系数。

编写 MATLAB 程序如下:

```
a=[1, -0.9]; %输入差分方程的系数
b=[1, 0.9];
x01=0; y01=0; %输入初始条件
xi=filter(b, a, x01, y01); %计算初始状态
N=40; n=0: N-1;
x=(exp((-0.05+j*0.4)*n)).* [n>=2];
%建立输入信号x(n)
y=filter(b, a, x, xi); %求系统的完全响应
subplot(2, 2, 1), stem(n, real(x));
title('输入信号x(n)的实部');
subplot(2, 2, 2), stem(n, imag(x));
title('输入信号x(n)的虚部');
subplot(2, 2, 3), stem(n, real(y));
title('系统响应y(n)的实部');
subplot(2, 2, 4), stem(n, imag(y));
title('系统响应y(n)的虚部');
结果如图6-2所示。注意: 由于输入信号是一个复指数信号, 作图时应分别表示。
```



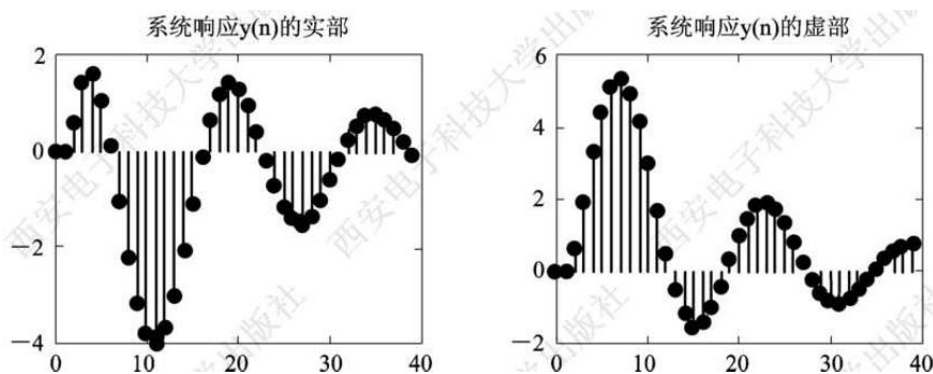


图 6-2 例 6-2x(n)和 y(n)的实部、虚部波形

例 6-3 已知一个系统的差分方程为

$$y(n) - 1.5y(n-1) + 0.5y(n-2) = x(n) \quad n \geq 0$$

满足初始条件 $y(-1)=4$, $y(-2)=10$, 用 `filtic` 和 `filter` 子函数求系统输入为 $x(n)=(0.25)^n u(n)$ 时的零输入、零状态以及完全响应。

解 为了更深入地理解 `filtic` 和 `filter` 子函数的用途, 我们对上述方程进行推导, 可得到完全响应的。

$$y(n) = \left[\left(\frac{1}{2} \right)^n + \frac{1}{3} \left(\frac{1}{4} \right)^n \right] u(n) + \frac{2}{3} u(n)$$

在使用 `filtic` 和 `filter` 子函数进行系统差分方程的求解时, 我们同时将上面推导出的公式也编入程序, 与 `MATLAB` 子函数计算的结果进行比较。编写 `MATLAB` 程序如下:

```
a = [1, -1.5, 0.5]; %输入系统a、b系数
b = [1];
N=20; n=0: N-1;
x=0.25.^n; %建立输入信号x(n)
x0=zeros(1, N); %建立零输入信号
y01 = [4, 10]; %输入初始条件
xi=filtic(b, a, y01); %计算初始状态
y0=filter(b, a, x0, xi); %求零输入响应
xi0=filtic(b, a, 0); %计算初始状态为零的情况
y1=filter(b, a, x, xi0); %求零状态响应
y=filter(b, a, x, xi); %求系统的完全响应
%用公式求完全响应
y2=((1/3)*(1/4).^n+(1/2).^n+(2/3)).*ones(1, N);
subplot(2, 3, 1), stem(n, x);
title('输入信号x(n)');
```

```

subplot(2, 3, 2), stem(n, y0);
title('系统的零输入响应');
subplot(2, 3, 3), stem(n, y1);
title('系统的零状态响应');
subplot(2, 2, 3), stem(n, y);
title('用filter求系统的完全响应y(n)');
subplot(2, 2, 4), stem(n, y2);
title('用公式求系统的完全响应y(n)');

```

程序执行的结果如图 6-3 所示

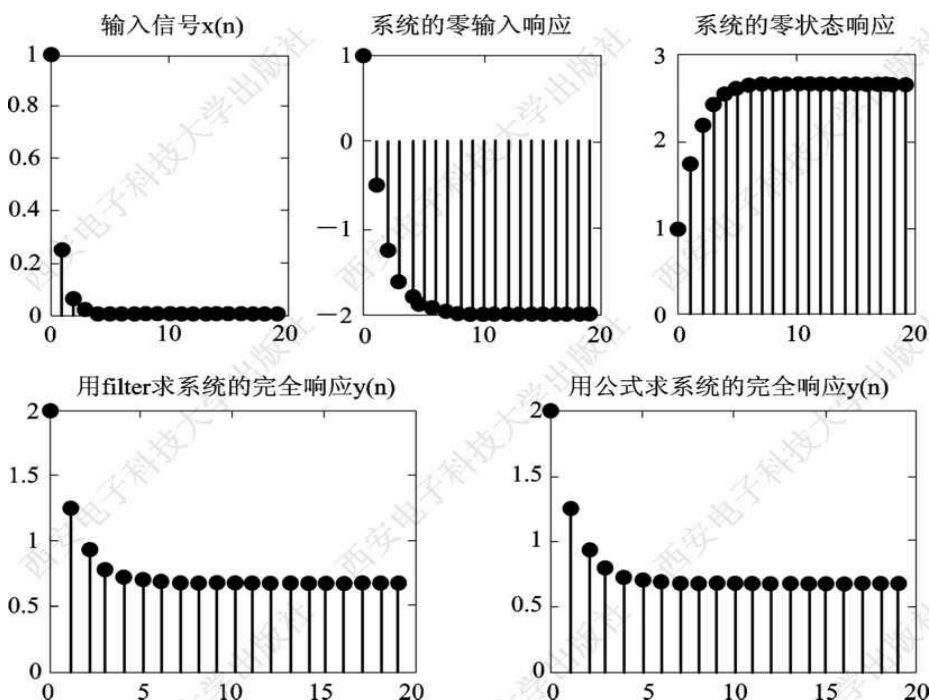


图 6-3 例 6-3x(n)和系统零输入、零状态和完全响应的波形

四、实验任务

(1) 一个 LSI 系统的差分方程表示式为:

$$y(n) - 0.5y(n-1) + y(n-6) - 0.5y(n-7) = x(n) - x(n-1) + x(n-2)$$

满足初始条件 $y(-1)=0$, $x(-1)=0$, 试用 dlsim 和 filter 两种方法求此系统的输入序列 $x(n)$ 为下列信号时的响应:

① $x(n) = u(n-3)$

② $x(n) = \delta(n) - \delta(n-5)$

③ $x(n) = e^{0.1n}u(n-3)$

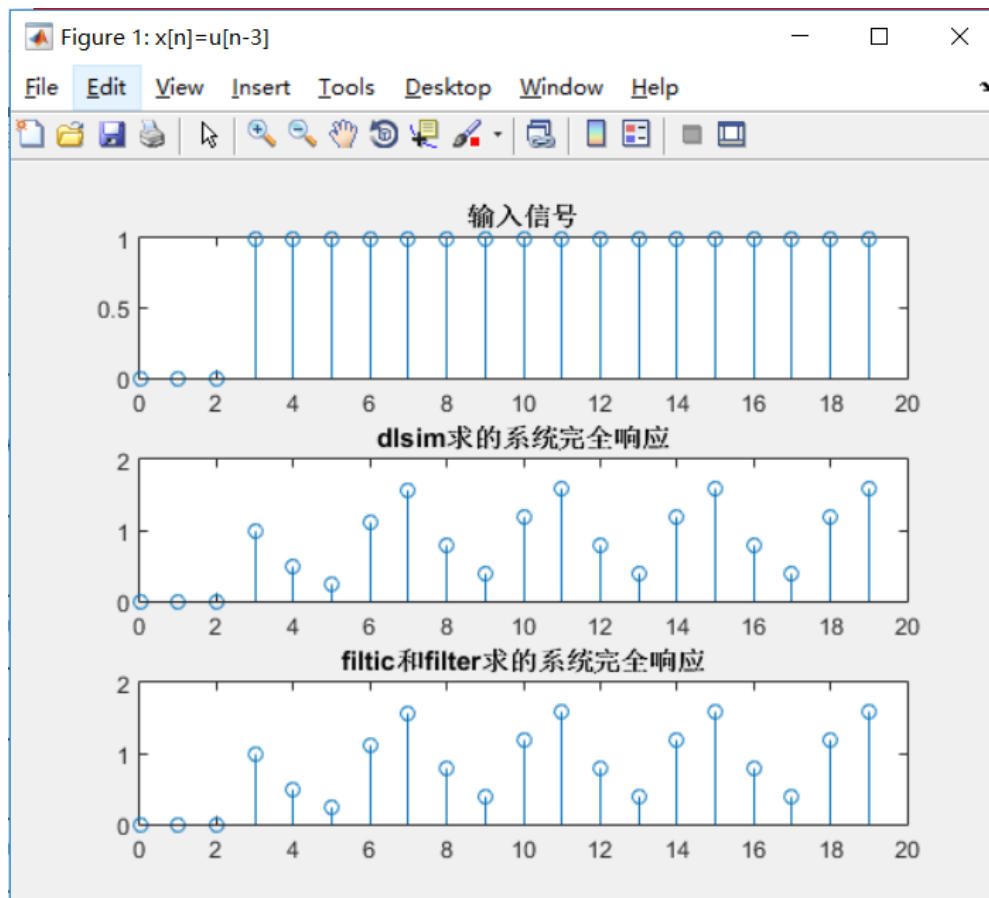
④ $x(n) = (0.5)^n u(n)$

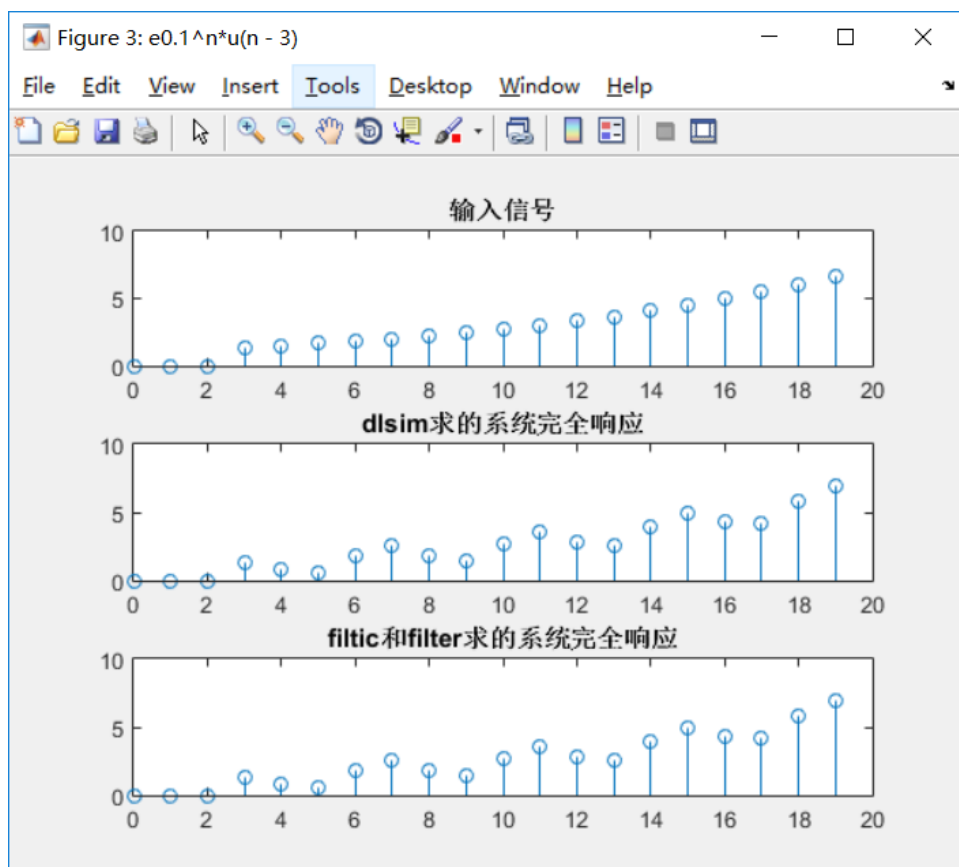
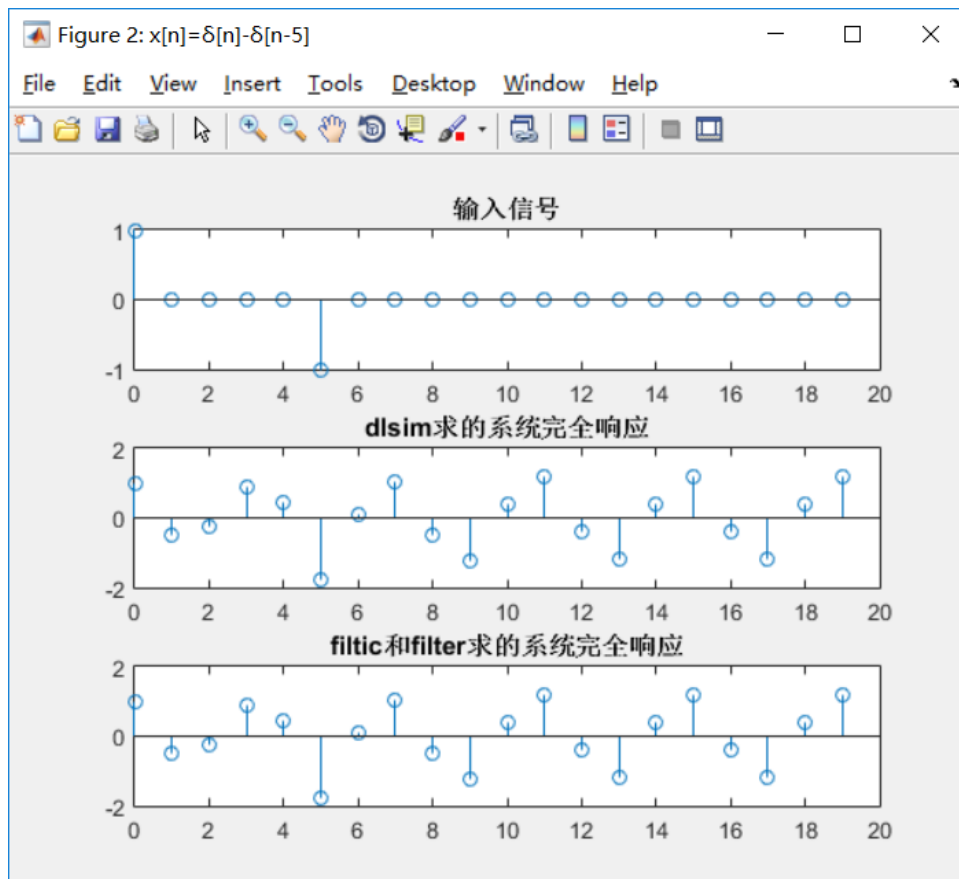
说明: 可以在同一段程序中将 4 个小题的 $x(n)$ 分别输入, 用 dlsim 和 filter 两种方法求输出, 同时显示采用不同子函数处理后的输出结果 $y(n)$ 。

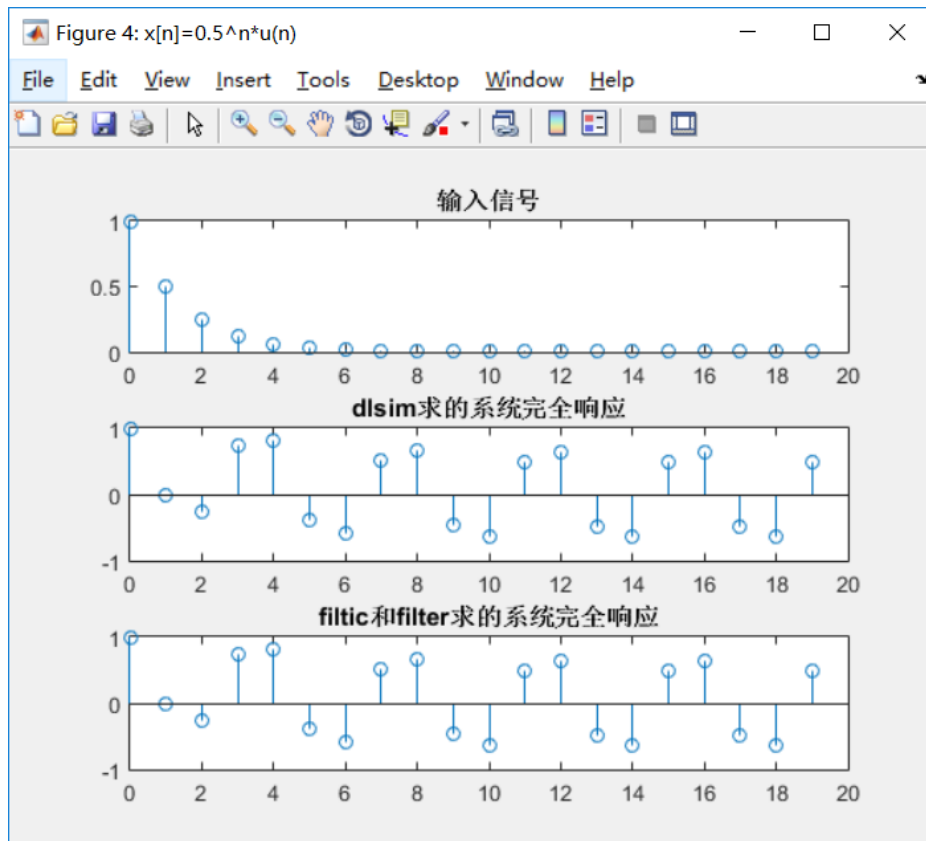
```

a=[1,-0.5,1,-0.5]; %输入系统a、b系数
b=[1,-1,1,0];
N=20;n=0:N-1;
for i=1:4
    if i==1 %建立输入信号x(n)
        x=0+[n>=3]; %u[n-3]
        figure('Name','x[n]=u[n-3]');
    elseif i==2
        x=0+[n==0]-[n==5]; % $\delta[n]-\delta[n-5]$ 
        figure('Name','x[n]= $\delta[n]-\delta[n-5]$ ');
    elseif i==3
        x=exp(0.1.*n).*[n>=3]; % $e^{0.1n}u(n-3)$ 
        figure('Name',' $e^{0.1n}u(n-3)$ ');
    else
        x=0.5.^n.*[n>=0]; % $0.5^n u(n)$ 
        figure('Name','x[n]= $0.5^n u(n)$ ');
    end
    subplot(3,1,1);stem(n,x);title('输入信号')
    %用dlsim求
    y=dlsim(b,a,x);
    subplot(3,1,2);stem(n,y);title('dlsim求的系统完全响应')
    %用filtic和filter求
    x01=0;y01=0;%输入初始条件
    xi=filtic(b,a,y01,x01);%计算初始状态
    y=filter(b,a,x,xi); %求系统完全响应
    subplot(3,1,3);stem(n,y);title('filtic和filter求的系统完全响应')
end

```







(2) 一个 LSI 系统的系统函数表示式为

$$H(z) = \frac{0.187632 + 0.241242z^{-1} + 0.241242z^{-2} + 0.187632z^{-3}}{1 - 0.602012z^{-1} + 0.495684z^{-2} - 0.0359244z^{-3}}$$

满足初始条件 $y(-1)=5$, $y(-2)=5$, 试用 `filtic` 和 `filter` 子函数求此系统的输入序列 $x(n)$ 为下列信号时的零输入、零状态以及完全响应:

① $x(n) = \delta(n-3)$

② $x(n) = R_5(n)$

③ $x(n) = \cos \frac{2\pi}{3}n + \sin \frac{3\pi}{10}n$

④ $x(n) = 0.6^n u(n-3)$

```

a=[1,-0.5,1,-0.5]; %输入系统a、b系数
b=[1,-1,1,0];
N=20;n=0:N-1;
for i=1:4
    if i==1 %建立输入信号x(n)
        x=0+[n>=3]; %u[n-3]
        figure('Name','x[n]=u[n-3]');
    elseif i==2
        x=0+[n==0]-[n==5]; %δ[n]-δ[n-5]
        figure('Name','x[n]=δ[n]-δ[n-5]');
    end
end

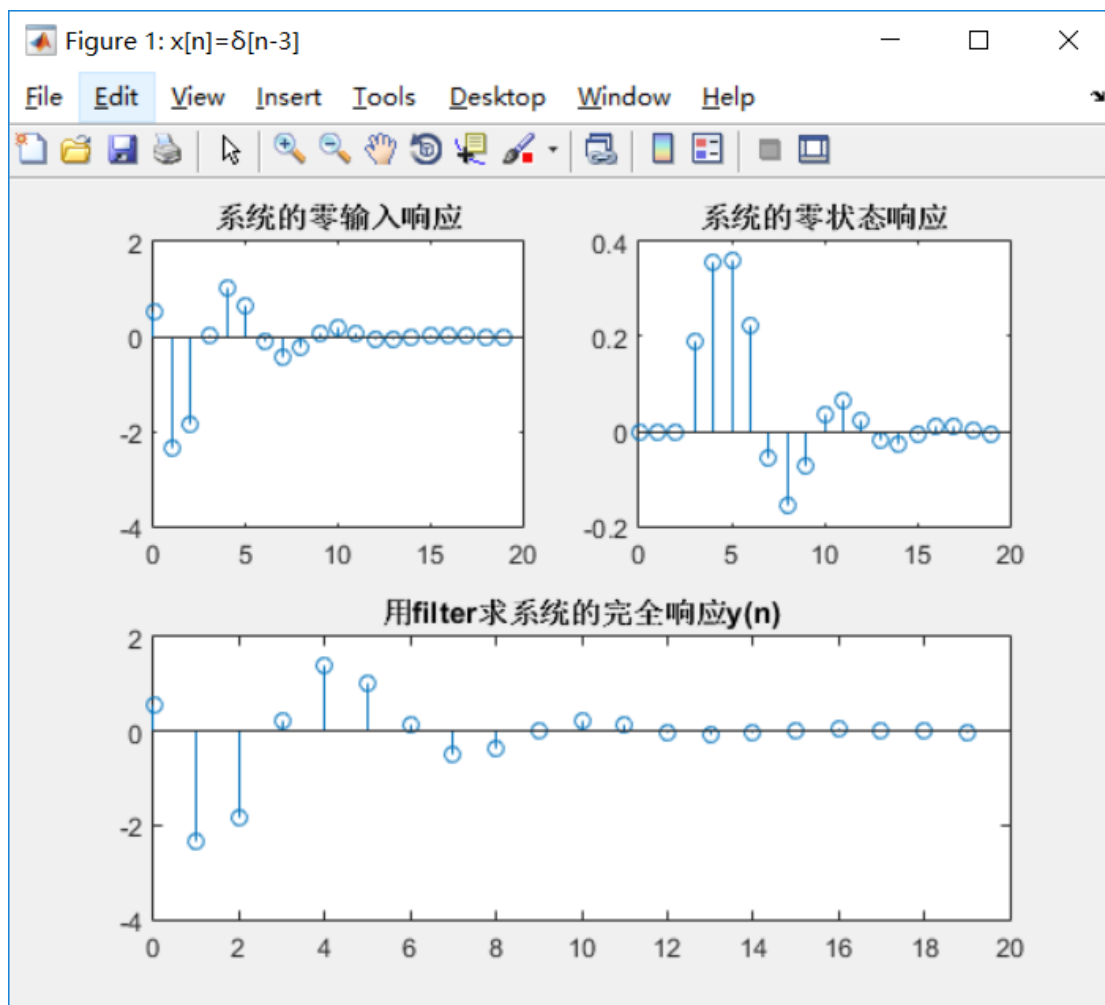
```

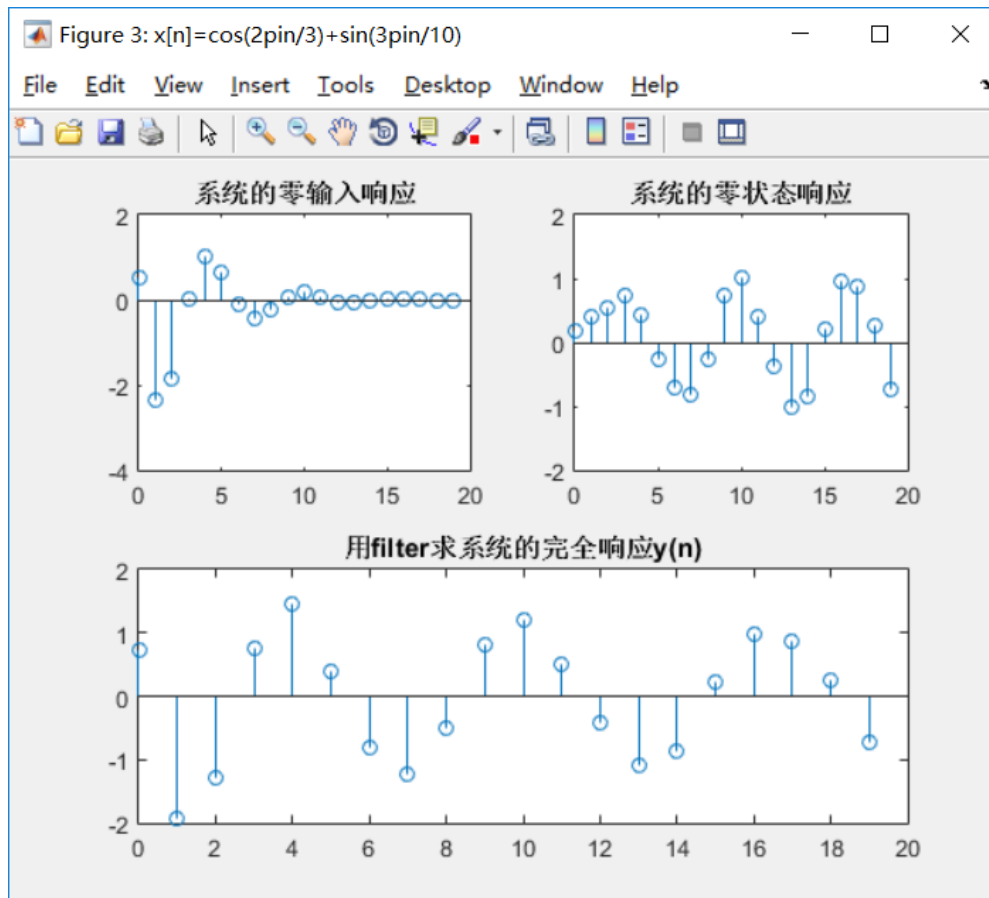
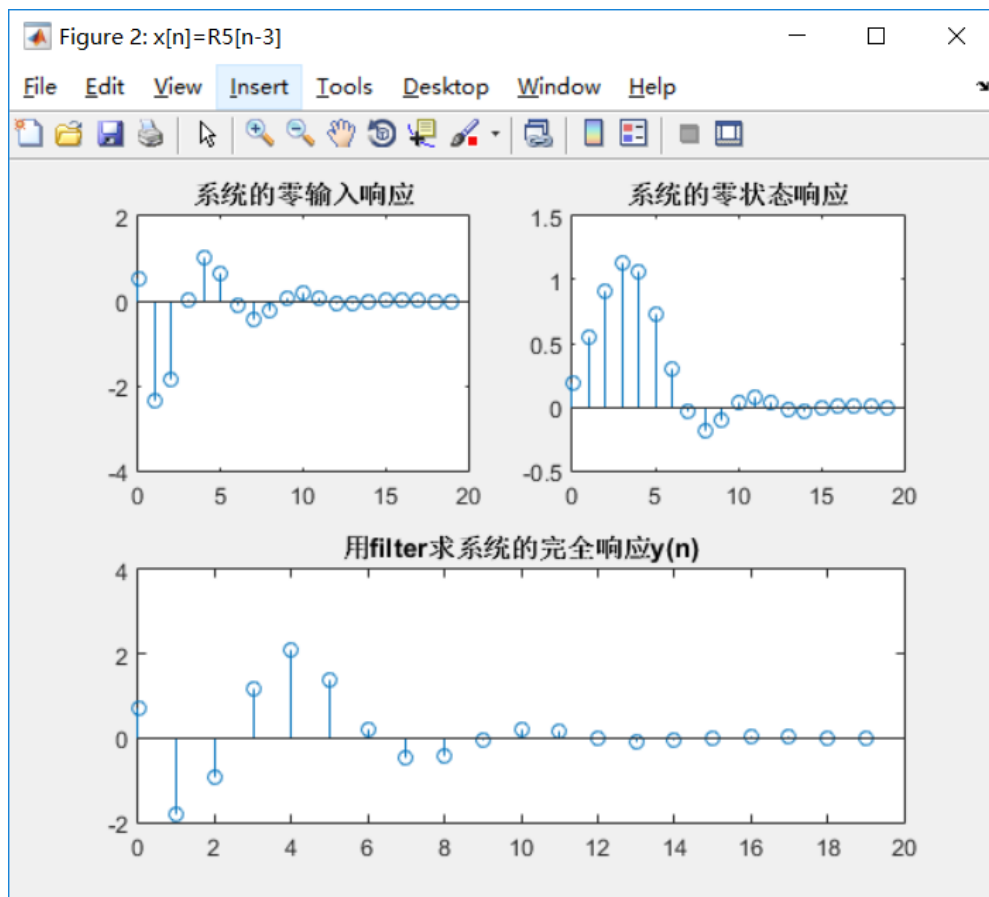


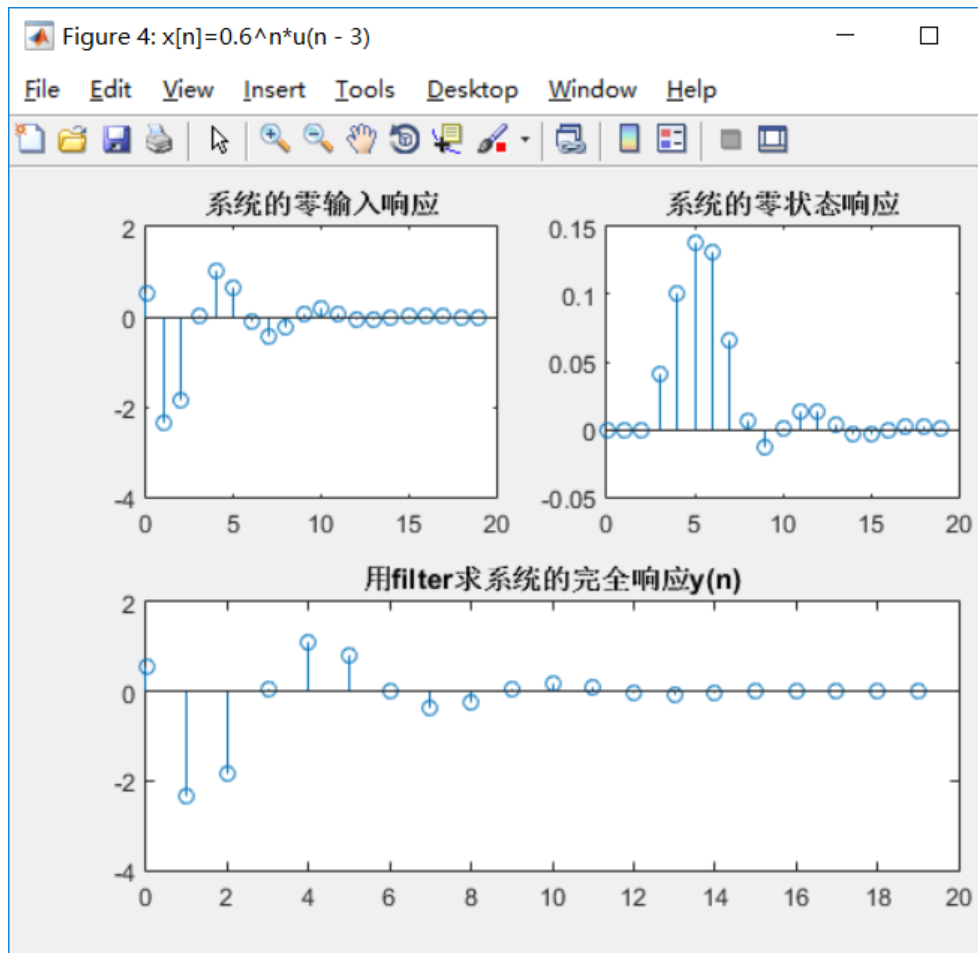
```

elseif i==3
    x=exp(0.1.*n).*[n>=3];%e^0.1n*u(n-3)
    figure('Name','e0.1^n*u(n-3)');
else
    x=0.5.^n.*[n>=0]; %0.5^n*u(n)
    figure('Name','x[n]=0.5^n*u(n)');
end
subplot(3,1,1);stem(n,x);title('输入信号')
%用dlsim求
y=dlsim(b,a,x);
subplot(3,1,2);stem(n,y);title('dlsim求的系统完全响应')
%用filtic和filter求
x01=0;y01=0;%输入初始条件
xi=filtic(b,a,y01,x01);%计算初始状态
y=filter(b,a,x,xi); % 求系统完全响应
subplot(3,1,3);stem(n,y);title('filtic和filter求的系统完全响应')
end

```







(3)思考题:

①回答预习思考题: MATLAB 中提供的 `dl Sims` 和 `filter` 两种方法, 使用中有何不同?

答: `dl Sims` 只要知道系统函数分子分母的系数序列以及输入信号就可以通过 `dl Sims(b,a,x)` 一步求出该离散 LSI 系统对任意输入信号 x 的响应。`filtic` 和 `filter` 子函数采用递推法进行系统差分方程的求解, 可以用于求解离散 LSI 系统对任意输入的完全响应。这个方法还要知道系统的初始条件并用 `filtic` 计算初始状态, 再用 `filter` 求响应。

②MATLAB 中提供了哪些求解离散 LSI 系统时域响应的方法及相关子函数?

答: (1)用 `conv` 子函数进行卷积积分, 求任意输入的系统零状态响应。

(2)用 `dl Sims` 子函数求任意输入的系统零状态响应。

(3)用 `filter` 和 `filtic` 子函数求任意输入的系统完全响应。

(4)用 `impz` 求时域单位冲激响应。

(5)用 `dstep` 求时域单位阶跃响应