پروژه ششم:

طراحی کنترل کننده با روش جایابی ساختار ویژه و ردیابی با ساختار ویژه:

در این پروژه سه مقدار ویژه ما منفی به دست می آیند ولی امکان دارد که یکی از مقادیر ویژه مثبت باشد و این بار جایابی ما مفید تر واقع خواهد شد که در این پروژه داریم:

Plant فضای حالت زیر را در نظر می گیریم.

$$\frac{dx}{dt}=\begin{pmatrix}-2 & 1 & 0\\ 0 & -3 & 0\\ 0 & 0 & -4\end{pmatrix}x+\begin{pmatrix}1 & 0\\ 1 & -0.5\\ 1 & 0.2\end{pmatrix}u$$

$$y=\begin{pmatrix}-17 & 4 & -10\\ 5 & 8 & -6\end{pmatrix}x$$

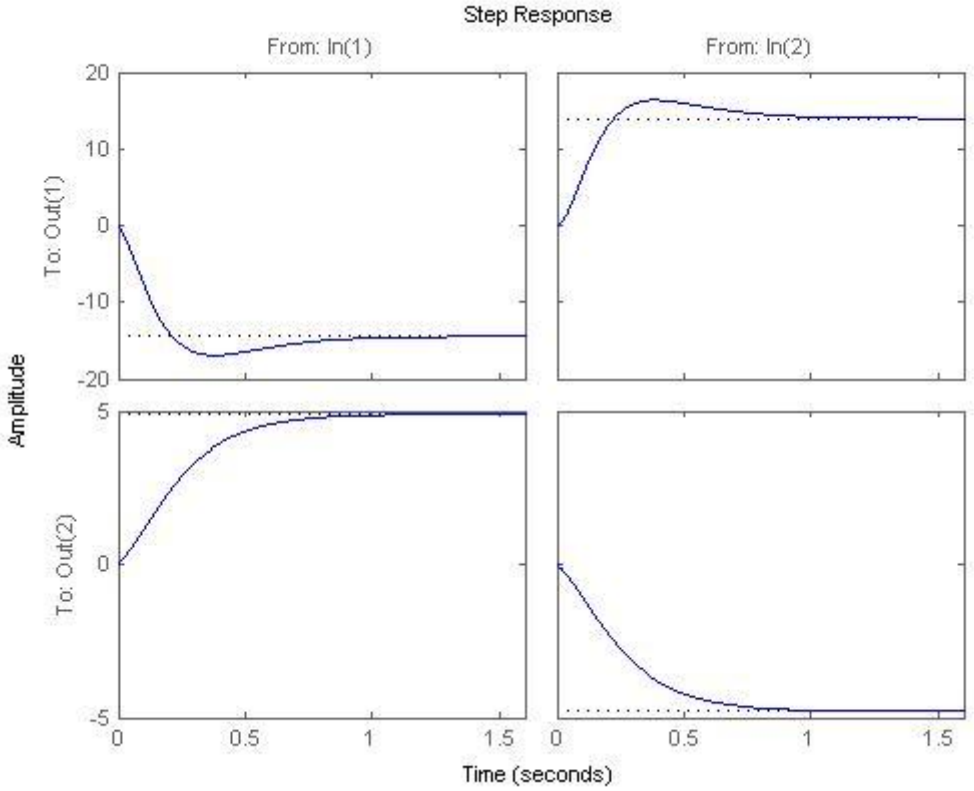
ابتدا، پایداری سیستم حلقه باز را بررسی می کنیم:

نتایج محاسبات مقادیر ویژه و پاسخ پله ی حلقه باز که در متلب انجام شده است، به شکل زیر است.

```
A=[-2 1 0;0 -3 0;0 0 -4];
B=[1 0;1 -0.5;1 0.2];
C=[-17 4 -10;5 8 -6];
eig(A)
```

ans = -2 -3 -4

بنابراین ، سیستم حلقه باز پایدار است و پاسخ پله ی آن به صورت زیر است.



تداخل در سیستم مشهود است، اما سیستم پایدار است.

برای کم کردن زمان نشست ، میتوانیم ، قطب ها را کمی دور تر قرار دهیم.

برای این منظور قطب های مطلوب را ، مقادیر $(-6,-9,-12)$ در نظر میگیریم و با استفاده از ساختار ویژه ، آن ها را جایابی می کنیم.

ابتدا برای مقادیر ویژه ی مطلوب ، ماتریس های S را به شکل نرمال هرمیتی تشکیل می دهیم، تا بتوانیم، فضای پوچی ماتریس S را به دست آوریم و در نهایت بتوانیم ، بردار های ویژه و بردار های $K.v_i$ را بیابیم.

K ماتریس فیدبک حالت است.

دستورات متلب و نتایج به شکل زیر است.

```
S1=[A+6*eye(3) B;zeros(2,5)];
S2=[A+9*eye(3) B;zeros(2,5)];
S3=[A+12*eye(3) B;zeros(2,5)];
a=rref(S1);
b=rref(S2);
c=rref(S3);
a(4,4)=-1;a(5,5)=-1;
b(4,4)=-1;b(5,5)=-1;
c(4,4)=-1;c(5,5)=-1;
```

حال با استفاده از فرم استاندارد هرمیتی، فضای پوچی ماتریس $\begin{pmatrix} A-\lambda_i I & B \end{pmatrix}$ را می یابیم.

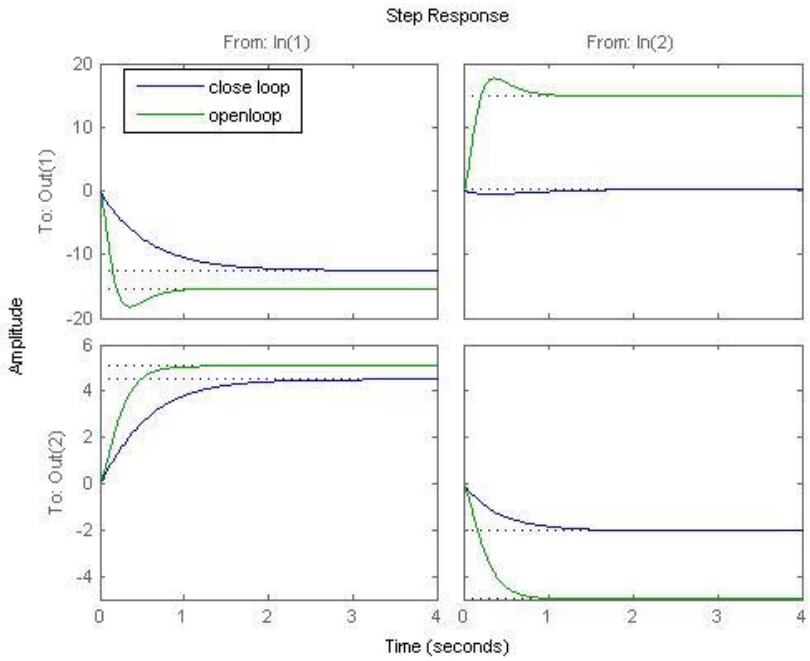
در ادامه نیز ماتریس K را میابیم.

```
a1=a(:,4)+a(:,5);
b1=b(:,4)+b(:,5);
c1=c(:,4)+c(:,5);
Q=[a1(4,1) b1(4,1) c1(4,1);a1(5,1) b1(5,1) c1(5,1)];
V=[a1(1,1) b1(1,1) c1(1,1);a1(2,1) b1(2,1) c1(2,1);a1(3,1) b1(3,1)
c1(3,1)];
K=Q*V^(-1)
Acl=A+B*K
eig(Acl)
```

ans = -6 -9 -12

ملاحظه می شود که قطب های حلقه بسته در نقاط مورد نظر جایابی شدند.

پاسخ پله ی سیستم جدیدحلقه بسته و همچنین سیستم اولیه در شکل زیر مشاهده می شود.



ملاحظه می شود که زمان نشست کم تر شده است و اثر تداخل بهبود یافته است.

مقایسه ی زمان نشست ، برای ورودی های دوم به شرح زیر است:

ورودی دوم: سیستم حلقه باز

SS1 =

RiseTime: 1.1664

SettlingTime: 1.5458

سیستم حلقه بسته :

SS2 =

RiseTime: 0.0063

SettlingTime: 0.4105

برای ورودی های اول هم به صورت مشابه ، قابل محاسبه است.

برنامه ی متلب برای محاسبه ی این مقادیر و همچنین رسم نمودارها(برنامه ی نهایی)

در زیر مشاهده می شود:

```
clear all;close all; clc;
A=[-2 1 0;0 -3 0;0 0 -4];
B=[1 0;1 -0.5;1 0.2];
C=[-17 4 -10;5 8 -6];
D=[0 0;0 0];
ghotb1=eig(A)
step(A,B,C,D); hold on
Transfarfunction1=ss2tf(A,B,C,D,2)
sys=tf([0 -4 -13.5 6],[0 -5.2 -32.5 -49.2]);
SS1=stepinfo(sys,'RiseTimeLimits',[0.05,0.95])
S1=[A+6*eye(3) B;zeros(2,5)];
S2=[A+9*eye(3) B;zeros(2,5)];
S3=[A+12*eye(3) B;zeros(2,5)];
a=rref(S1);
b=rref(S2);
c=rref(S3);
a(4,4)=-1;a(5,5)=-1;
b(4,4)=-1;b(5,5)=-1;
c(4,4)=-1;c(5,5)=-1;
a1=a(:,4)+a(:,5);
b1=b(:,4)+b(:,5);
c1=c(:,4)+c(:,5);
Q=[a1(4,1) b1(4,1) c1(4,1);a1(5,1) b1(5,1) c1(5,1)];
V=[a1(1,1) b1(1,1) c1(1,1);a1(2,1) b1(2,1) c1(2,1);a1(3,1) b1(3,1)
c1(3,1)];
K=Q*V^(-1)
Acl=A+B*K
ghotb2=eig(Acl)
Transfarfunction2=ss2tf(Acl,B,C,D,2)
sys=tf([0 -4 3208 9668],[0 -5.2 -345 -3229]);
SS2=stepinfo(sys,'RiseTimeLimits',[0.05,0.95])
step(Acl,B,C,D)
```

نتایج نهایی در command window Matlab

```
ghotb1 =

    -2
    -3
    -4


Transfarfunction1 =

         0   -4.0000  -13.5000    6.0000
         0   -5.2000  -32.5000  -49.2000


SS1 =

        RiseTime: 1.1664
    SettlingTime: 1.5458
     SettlingMin: -0.1213
     SettlingMax: -0.0787
       Overshoot: 0
      Undershoot: 630.7692
            Peak: 0.7692
        PeakTime: 0


K =

  -93.3333  230.6667  -33.3333
  -93.3333  230.6667  -33.3333


Acl =

  -95.3333  231.6667  -33.3333
  -46.6667  112.3333  -16.6667
 -112.0000  276.8000  -44.0000


ghotb2 =

  -12.0000
   -9.0000
   -6.0000


Transfarfunction2 =

   1.0e+03 *

         0   -0.0040    3.2085    9.6680
```

```
         0   -0.0052   -0.3457   -3.2296


SS2 =

        RiseTime: 0.0063
    SettlingTime: 0.4105
     SettlingMin: -8.1341
     SettlingMax: -2.8680
       Overshoot: 171.6705
      Undershoot: 25.6914
            Peak: 8.1341
        PeakTime: 0.0435

>>
```

ملاحظه می شود که پاسخ نهایی سیستم حلقه بسته، کاملا بهبود یافته است.

قسمت دوم پروژه(Tracking)

Plant زیر را در نظر می گیریم:

$$\dot{x} = Ax + Bu$$

$$y = cx$$

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 4 & 4 & -1 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

می خواهیم که خروجی، مقدار ورودی $r(t) = \begin{pmatrix} 1 \\ -2 \end{pmatrix}$ را دنبال کند.

در ابتدا شرایط لازم و کافی را برای این که بتوان قانون کنترل را پیدا کرد ، بررسی می کنیم.

شرط کنترل پذیری:

```
A=[0 1 0;0 0 1;4 4 -1];
B=[1 0;0 0;0 1];
C=[1 0 0;0 1 1];
R1=[B A];
```

ans =3

در ادامه رتبه ی ماتریس $\begin{pmatrix} B & A \\ 0 & -E \end{pmatrix}$ را بررسی می کنیم که باید برابر n+p یا جمع دو پارامتر مرتبه ی سیستم و مرتبه ی بردار ورودی ای که میخواهیم آن را ردیابی کنیم باشد.

در این پلنت:

```
D=[0 0;0 0];
R2=[B A;D -C];
rank(R2)
```

ans =5

بنابراین شرایط لازم و کافی برقرار است.

در ادامه مشابه قسمت الف همین پروژه، به ازای قطب های مطلوب، ماتریس های S را تشکیل می دهیم.

قطب های سیستم اولیه و حلقه باز در 1-,2 , 2- قرار دارند و قطب های مطلوب در 6-,5-,4-,3-,2-
قرار

دارند.

بنابراین ماتریس های S و سپس فرم HNF و درنهایت ماتریس فیدبک حالت K را مشابه قبل می یابیم.

این عملیات باید روی ماتریس های زیر انجام شود.

$$\bar{A}=\begin{pmatrix} A & 0 \\ -C & 0 \end{pmatrix}$$

$$\bar{B}=\begin{pmatrix} B \\ 0 \end{pmatrix}$$

$$\bar{C}=\begin{pmatrix} C & 0 \end{pmatrix}$$

برنامه ی متلب در ادامه آمده است:

```
Abaar=[A E1;-C D];
Bbaar=[B;D];
S1=[Abaar+2*eye(5) Bbaar];
S2=[Abaar+3*eye(5) Bbaar];
S3=[Abaar+4*eye(5) Bbaar];
S4=[Abaar+5*eye(5) Bbaar];
S5=[Abaar+6*eye(5) Bbaar];
a=rref(S1);
b=rref(S2);
c=rref(S3);
d=rref(S4);
e=rref(S5);
%HNF form of matrix a=a1
a1=[a(1:4,1:7);zeros(1,7);a(5,1:7);zeros(1,7)];
a1(5,5)=-1;
a1(7,7)=-1;
%HNF form of matrix b=b1
b1=[b(1:5,1:7);zeros(1,7);zeros(1,7)];
b1(6,6)=-1;
b1(7,7)=-1;
%HNF form of matrix c=c1
c1=[c(1:5,1:7);zeros(1,7);zeros(1,7)];
c1(6,6)=-1;
c1(7,7)=-1;
%HNF form of matrix d=d1
d1=[d(1:5,1:7);zeros(1,7);zeros(1,7)];
d1(6,6)=-1;
d1(7,7)=-1;
%HNF form of matrix e=e1
e1=[e(1:5,1:7);zeros(1,7);zeros(1,7)];
e1(6,6)=-1;
e1(7,7)=-1;
% Matrix V & Q:
```

```
V=zeros(5,5);
Q=zeros(2,5);
V(:,1)=a1(1:5,7);
V(:,2)=b1(1:5,7);
V(:,3)=c1(1:5,7);
V(:,4)=d1(1:5,7);
V(:,5)=e1(1:5,7);
Q(:,1)=a1(6:7,7);
Q(:,2)=b1(6:7,7);
Q(:,3)=c1(6:7,7);
Q(:,4)=d1(6:7,7);
Q(:,5)=e1(6:7,7);
%Matrix Kbaar is:
Kbaar=Q*V^(-1);
%state apace in closeloop is:
Acl=[Abaar+Bbaar*Kbaar]
Bcl=[0 0;0 0;0 0;1 0;0 1]*[1 0;0 -2]
Ccl=[1 0 0 0 0;0 1 1 0 0]
Dcl=[0 0;0 0]
%response to input 1
[tfn1,tfd1]=ss2tf(Acl,Bcl,Ccl,Dcl,1)
%response to input 2
[tfn2,tfd2]=ss2tf(Acl,Bcl,Ccl,Dcl,2)
step(Acl,Bcl,Ccl,Dcl,2)
```

ماتریس K به صورت زیر به دست آمد.

```
Kbaar =

  -2   -2    0    0   -2
-184   39  -17   36  162
```

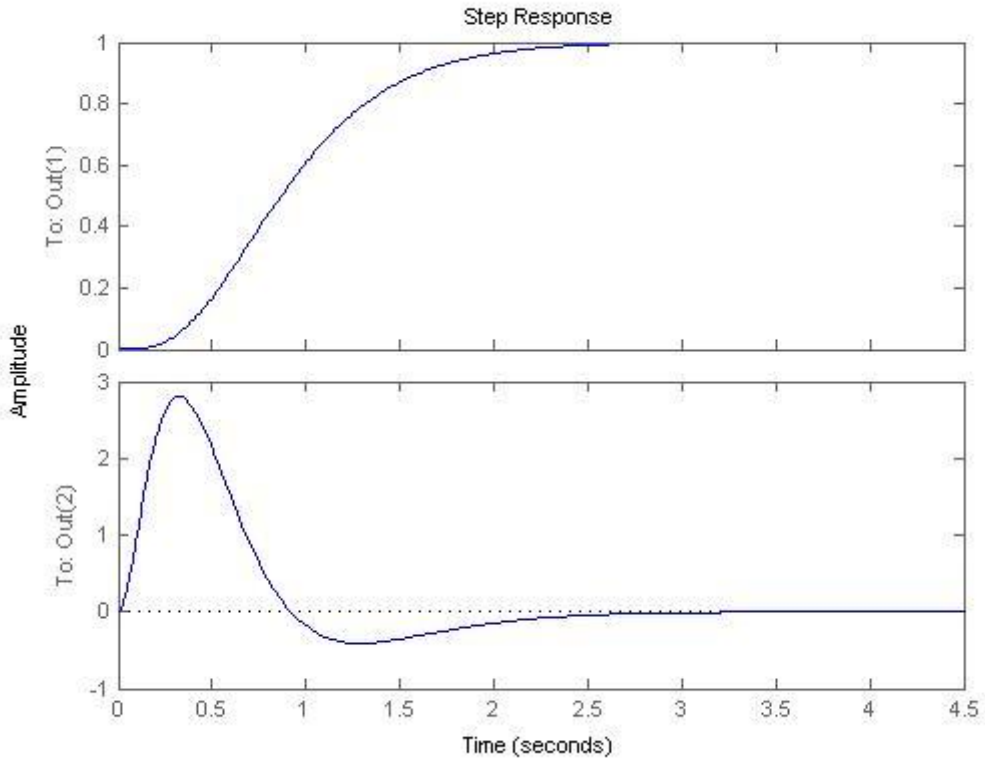
مقادیر ویژه ی Acl به صورت زیر حاصل می شود.

```
ans =

   -6.0000
   -5.0000
   -4.0000
   -3.0000
   -2.0000
```

در ادامه برای اطمینان از صحیح بودن طراحی، و اطمینان از ردیابی، پاسخ به ازای ورودی اول و ورودی دوم

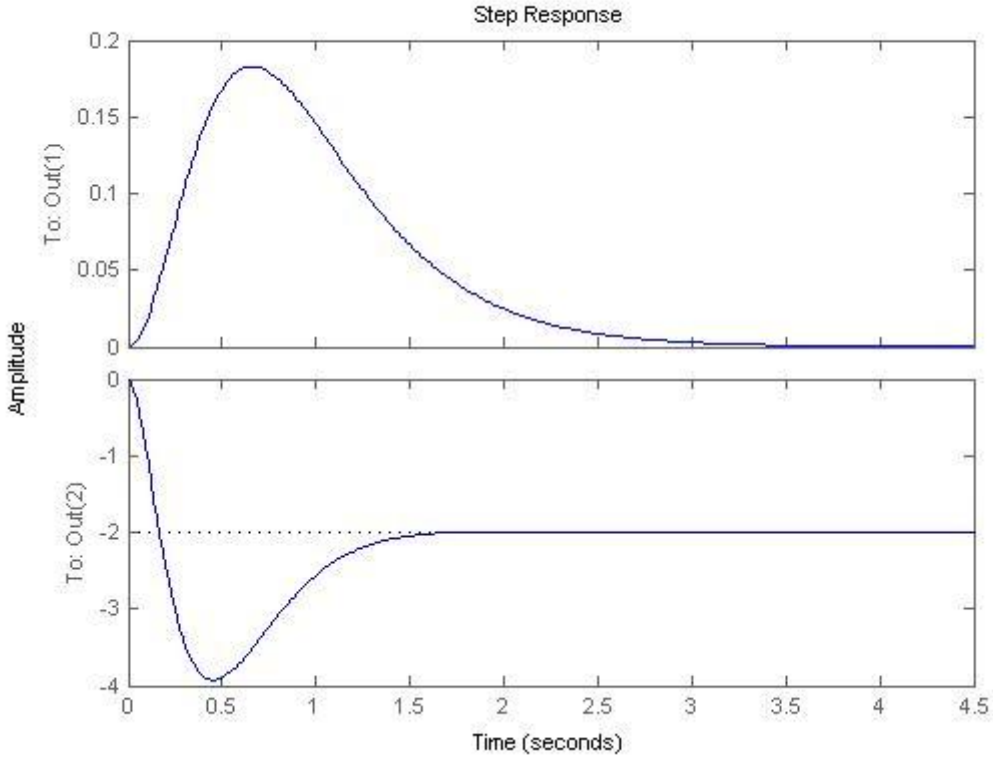
را رسم می کنیم.



در شکل بالا پاسخ به ورودی $r=\begin{pmatrix}1\\0\end{pmatrix}$ ، را مشاهده می کنیم.

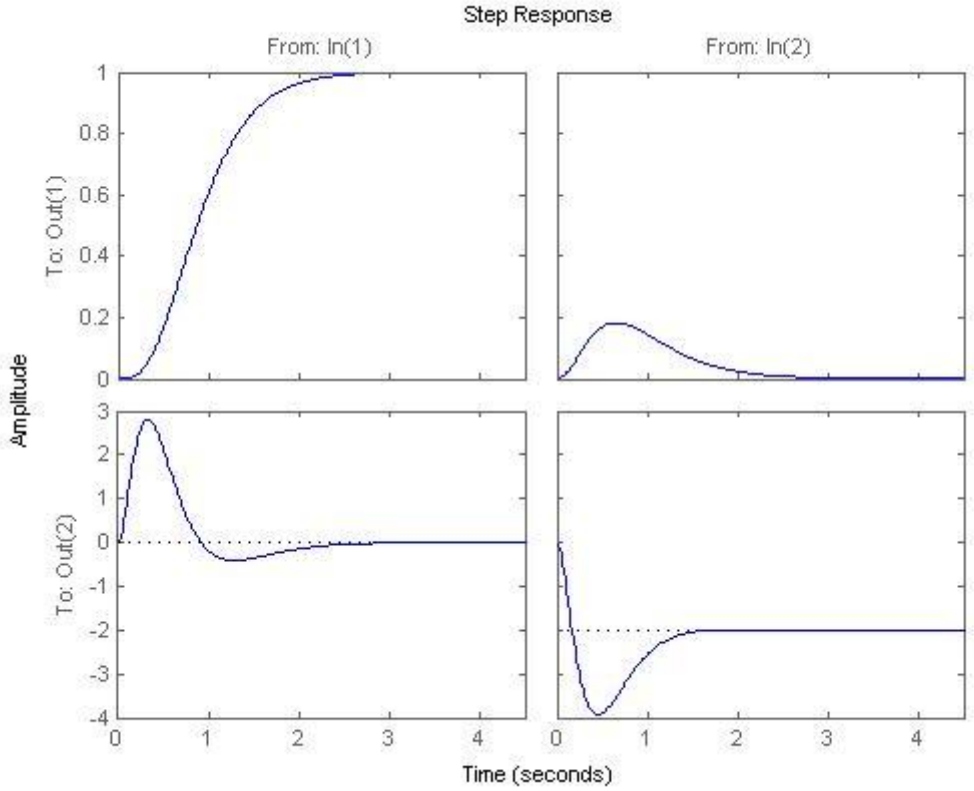
می بینیم که خروجی اول ، ورودی اول را ردیابی می کند.

حال پاسخ به ورودی $r=\begin{pmatrix}0\\-2\end{pmatrix}$ ، را مشاهده می کنیم.



مشخص است که خروجی دوم ، ورودی دوم را ردیابی می کند.

در نهایت نیز، پاسخ توأم دو ورودی و دو خروجی را رسم می کنیم.



می بینیم که در این حالت، تداخل ورودی اول در خروجی دوم، وجود دارد.