به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

گزارش پروژه سیستم های کنترل دیجیتال

نام و نام خانوادگی	پریسا اصغری توانا - نیکا امامی
شماره دانشجویی	810198356 - 810198352

فهرست

بخش ۱

معادلات دینامیک سیستم را مطابق زیر داریم:

$$\begin{cases} F_m = c\frac{I^2}{1-y} \\ m\ddot{y} = -mg - f_v\dot{y} + F_m \\ V = RI + L\dot{I} \end{cases}$$

متغیرهای حالت را مطابق صورت پروژه، به صورت زیر در نظر می‌گیریم و معادلات حالت را به صورت زیر می‌نویسیم:

y: ارتفاع گوی،

$\dot{y}$: سرعت گوی،

I: جریان گذرنده از سیم پیچ،

V: ولتاژ اعمالی به مدار (ورودی)،

$$F_m = \frac{cI^2}{1-y}$$

$$m\ddot{y} = -mg - f_v\dot{y} + F_m \rightarrow \ddot{y} = -g - \frac{f_v}{m}\dot{y} + \frac{F_m}{m}$$

$$V = RI + L\dot{I} \rightarrow \quad \dot{I} = \frac{V}{L} - \frac{RI}{L}$$

$$\begin{cases} x_1 = y \\ x_2 = \dot{y} \\ x_3 = I \end{cases} \qquad \begin{aligned} &\dot{x}_1 = \dot{y} = x_2 \\ &\dot{x}_2 = \ddot{y} = -g - \frac{f_v}{m}x_2 + \frac{1}{m}\frac{cx_3^2}{1-x_1} \\ &\dot{x}_3 = \dot{I} = \frac{V}{L} - \frac{R}{L}x_3 \\ &y = x_1 \end{aligned}$$

برای محاسبه نقطه تعادل، از رابطه داده شده برای yd استفاده کرده و مقادیر مشتق ها را برابر ۰ قرار میدهیم. روابط به صورت زیر خواهد بود:

$$\dot{x}_1 = x_2 \rightarrow f_1$$
$$\dot{x}_2 = -g - \frac{f_v}{m} x_2 + \frac{1}{m}\frac{c x_3^2}{1-x_1} \rightarrow f_2$$
$$\dot{x}_3 = \frac{v}{L} - \frac{R}{L} x_3 \rightarrow f_3$$
$$y = x_1$$

نقطه تعادل $\rightarrow \dot{x} = 0 \rightarrow$ $x_2 = 0$

$$v - R x_3 = 0 \rightarrow x_3 = \frac{v}{R}$$
$$-g + \frac{1}{m}\frac{c x_3^2}{1-x_1} = 0 \rightarrow gm(1-x_1) = c\left(\frac{v}{R}\right)^2$$
$$1-x_1 = \frac{c v^2}{g m R^2} \rightarrow x_1 = 1 - \frac{c v^2}{g m R^2}$$

با در نظر گرفتن مقدار پارامتر a برابر ۲ (رقم یکان شماره دانشجویی)، نقطه تعادل به صورت زیر بدست می‌آید:

$$x_1 = y \rightarrow y_d = 0.3 + 0.02 \rightarrow x_{1e} = 0.32$$
$$x_{2e} = 0$$
$$x_{3e} = v/R \rightarrow v :\ x_1 = 1 - \frac{c v^2}{g m R^2} \rightarrow v = \sqrt{\frac{(1-x_1) g m R^2}{c}}$$

پس : $v = \sqrt{\frac{(1-0.32)\times 9.8 \times 0.102 \times 25}{0.3}} = 7.526\ v$

پس : $x_{3e} = v/R = \frac{7.526}{5} = 1.5052$

پس : $\begin{bmatrix} x_1^* \\ x_2^* \\ x_3^* \end{bmatrix} = \begin{bmatrix} 0.32 \\ 0 \\ 1.5052 \end{bmatrix}$

اکنون مقادیر سایر پارامترها را حساب کرده و مطابق محاسبات زیر سیستم را خطی سازی میکنیم:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -9.8 - \left(\frac{0.02}{0.102}\right)x_2 + \frac{0.3}{0.102}\ \frac{x_3^2}{1-x_1} \\ \quad = -9.8 - 0.196\, x_2 + 2.9412\ \frac{x_3^2}{1-x_1} \\ \dot{x}_3 = \frac{1}{0.02}u - \frac{5}{0.02}x_3 = 50u - 250x_3 \end{cases}$$

$$\frac{\partial f_1}{\partial x_1} = \frac{\partial f_1}{\partial x_3} = 0 \qquad \frac{\partial f_1}{\partial x_2} = 1 \qquad \frac{\partial f_1}{\partial u} = 0$$

$$\frac{\partial f_2}{\partial x_1} = \frac{0.3(1.5052)^2}{0.102\ (1-0.32)} = 14.41$$

$$\frac{\partial f_2}{\partial x_2} = -0.196 \qquad \frac{\partial f_2}{\partial x_3} = \frac{0.3 \times 3 \times 1.5052^2}{1-0.32} = 13.02$$

$$\frac{\partial f_3}{\partial x_1} = \frac{\partial f_3}{\partial x_2} = 0 \ , \quad \frac{\partial f_3}{\partial x_3} = -250 \ ; \quad \frac{\partial f_2}{\partial u} = 0$$

$$\frac{\partial f_3}{\partial u} = 50$$

ماتریس‌های حالت مطابق زیر بدست می‌آیند:

$$X = \begin{pmatrix} x_1 - x_{1e} \\ x_2 - x_{2e} \\ x_3 - x_{3e} \end{pmatrix} , \ u = u - u_e$$

پس ماتریس ها : $A = \begin{bmatrix} 0 & 1 & 0 \\ 14.41 & -0.196 & 13.02 \\ 0 & 0 & -250 \end{bmatrix}$ و $B = \begin{bmatrix} 0 \\ 0 \\ 50 \end{bmatrix}$

$C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \qquad D = 0$

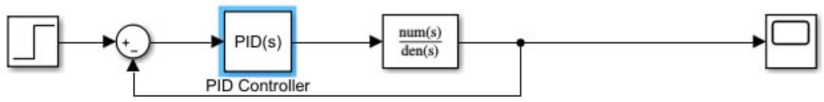
اکنون با استفاده از رابطه مربوطه، تابع تبدیل را در متلب بدست می‌آوریم:

```
A=[0 1 0;14.41 -0.1961 13.02; 0 0 -250];
B=[0;0;50];
C=[1 0 0];
D=0;
I=eye(3);
syms s
G=C*inv(s*I-A)*B+D
G=tf(6510000,[10000 2501961 346150 -36025000])
```

```
G =

                    6.51e06
  ---------------------------------------------
  10000 s^3 + 2.502e06 s^2 + 346150 s - 3.603e07

Continuous-time transfer function.
```

بخش ۲

در متلب و در بخش PID Tuner با در نظر گرفتن یک کنترلر PID، سیستم را پایدار میکنیم:



پاسخ سیستم به ورودی پله مطابق زیر است:

همان طور که میبینیم، خروجی سیستم در نهایت به مقدار عددی ۱ میل میکند، پس خطای حالت ماندگار سیستم ۰ است.

همچنین ویژگی های پاسخ گذرا و ماندگار در تصویر زیر قابل مشاهده هستند:

Controller Parameters

	Tuned	Block
P	84.5286	187.862
I	120.2832	349.54
D	11.4423	22.6285
N	3396.766	2000

Performance and Robustness

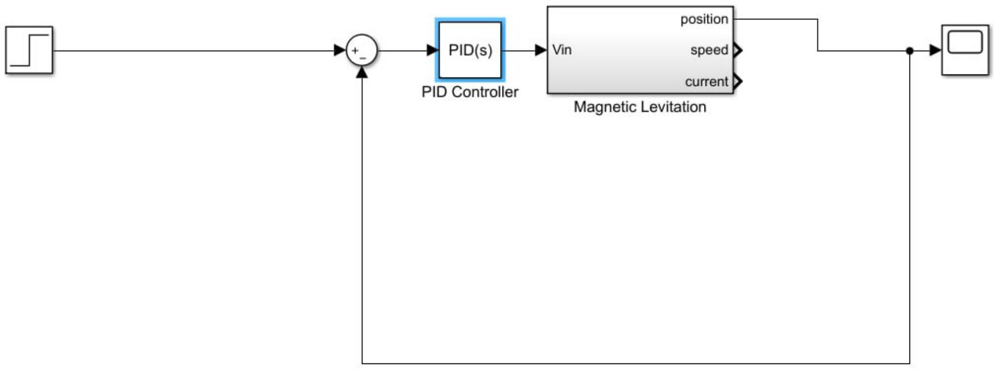
	Tuned	Block
Rise time	0.0427 seconds	0.022 seconds
Settling time	0.836 seconds	0.455 seconds
Overshoot	19.5 %	12.3 %
Peak	1.19	1.12
Gain margin	-19 dB @ 3.2 rad/s	-24.4 dB @ 3.89 rad/s
Phase margin	69 deg @ 29.7 rad/s	67.3 deg @ 57.7 rad/s
Closed-loop stability	Stable	Stable

همان طور که مشاهده می‌شود، زمان نشست، زمان صعود، مقدار فراجهش و حدفاز و حدبهره بدست آمده‌اند.

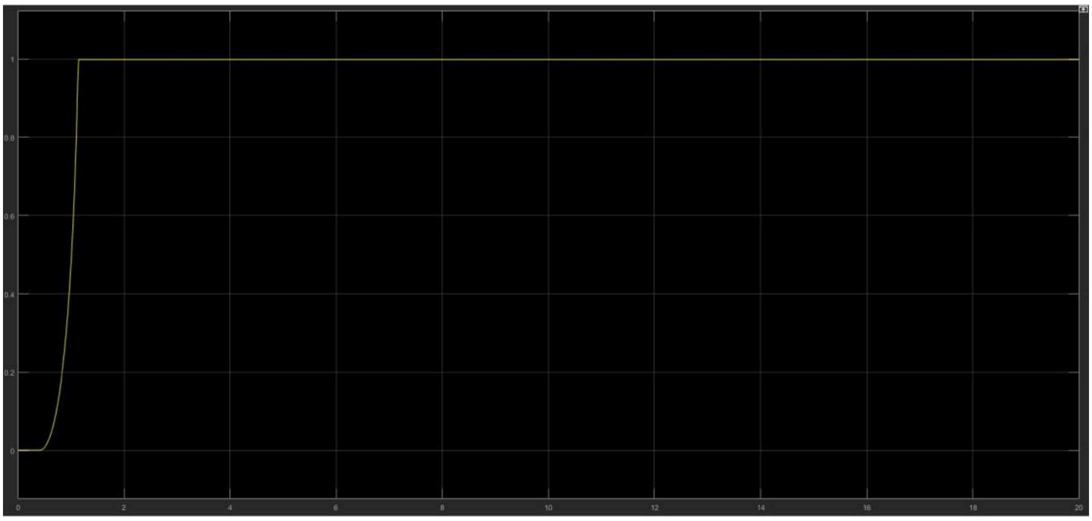
تابع تبدیل کنترل کننده که به خروجی‌های بالا منتهی شد، مطابق زیر است:

```
Continuous-time transfer function.
GC =

  4.544e04 s^2 + 3.761e05 s + 699080
  ----------------------------------
             s^2 + 2000 s
```

اکنون با قرار دادن کنترلر طراحی شده در این قسمت برای سیستم غیر خطی مطابق شکل زیر، پاسخ سیستم غیرخطی را نیز بررسی میکنیم.



پاسخ پله سیستم غیر خطی با کنترلر PID:



همان طور که میبینیم، سیستم غیرخطی نیز پایدار شد.

بخش ۳

گسسته سازی تابع تبدیل پیوسته به روش تبدیل ۲ خطی در محیط متلب، با استفاده از دستور c2d به شکل زیر قابل پیاده سازی است.

با استفاده از این روش، سیستم را به ازای ۳ نرخ نمونه برداری مختلف گسسته سازی میکنیم. مقادیر نرخ های نمونه برداری را با توجه به نامساوی مربوط به زمان صعود قرار میدهیم.

bilinear

T=0.004

```
G1dbl = c2d(GC,0.004,'tustin')
```

T=0.015

```
G2dbl = c2d(GC,0.015,'tustin')
```

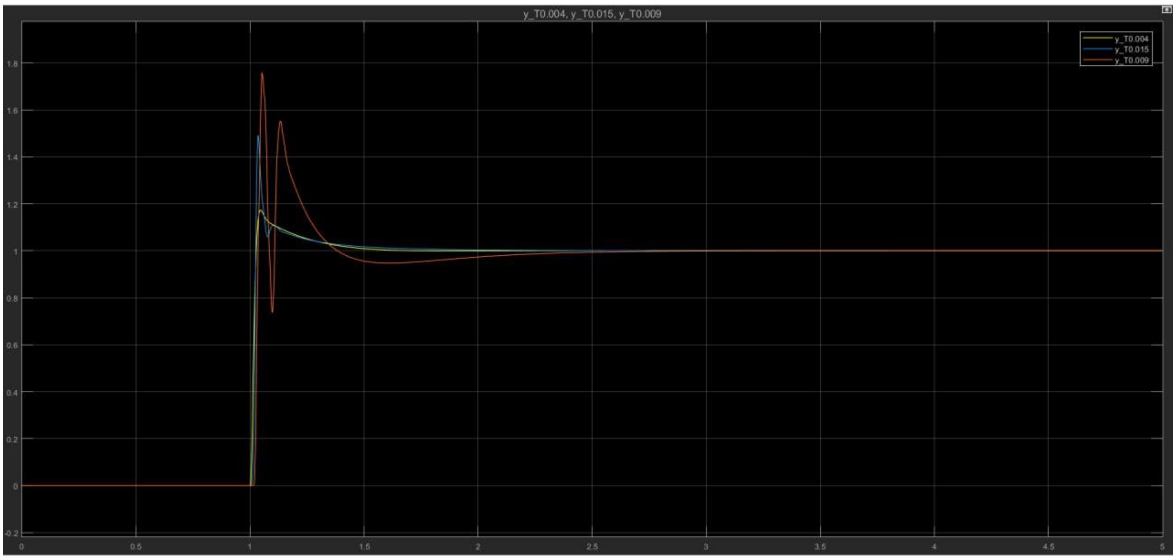
T=0.009

```
G3dbl = c2d(GC,0.009,'tustin')
```

توابع تبدیل مربوط به هرکدام از نرخ های نمونه برداری را به صورت زیر تشکیل میدهیم:

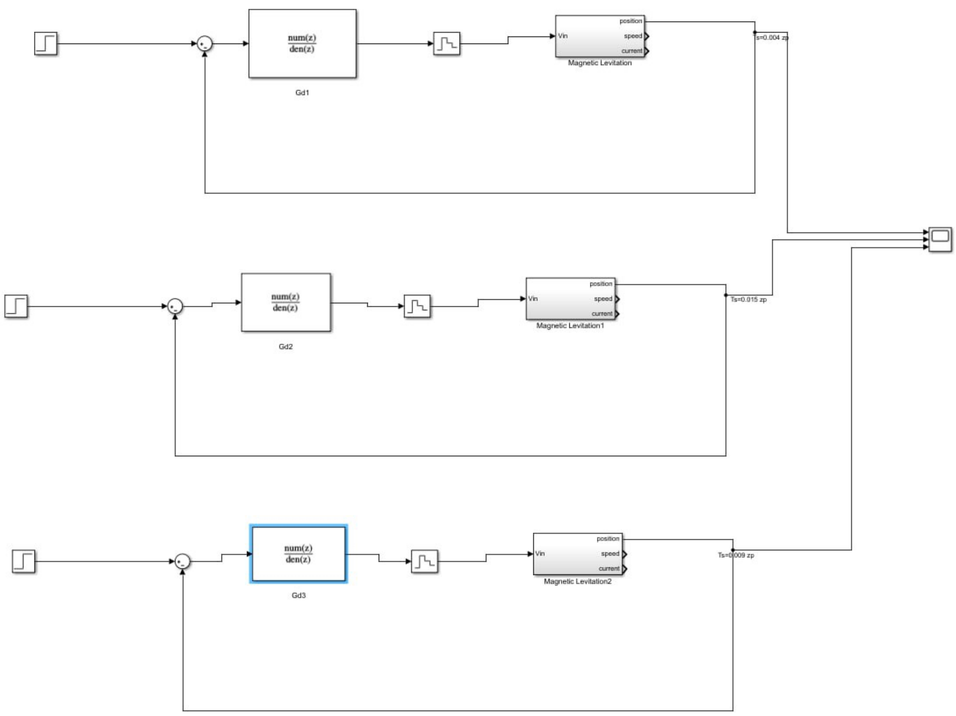
```
G1dbl =

  9240 z^2 - 1.818e04 z + 8939
  ----------------------------
       z^2 - 0.4 z - 0.6

Sample time: 0.004 seconds
Discrete-time transfer function.
G2dbl =

  3019 z^2 - 5676 z + 2666
  ------------------------
   z^2 - 0.125 z - 0.875

Sample time: 0.015 seconds
Discrete-time transfer function.
G3dbl =

  4715 z^2 - 9086 z + 4377
  ------------------------
     z^2 - 0.2 z - 0.8

Sample time: 0.009 seconds
Discrete-time transfer function.
```

اکنون برای سیستم گسسته سازی شده، پاسخ سیستم را به ازای سه نرخ نمونه برداری مختلف رسم میکنیم:

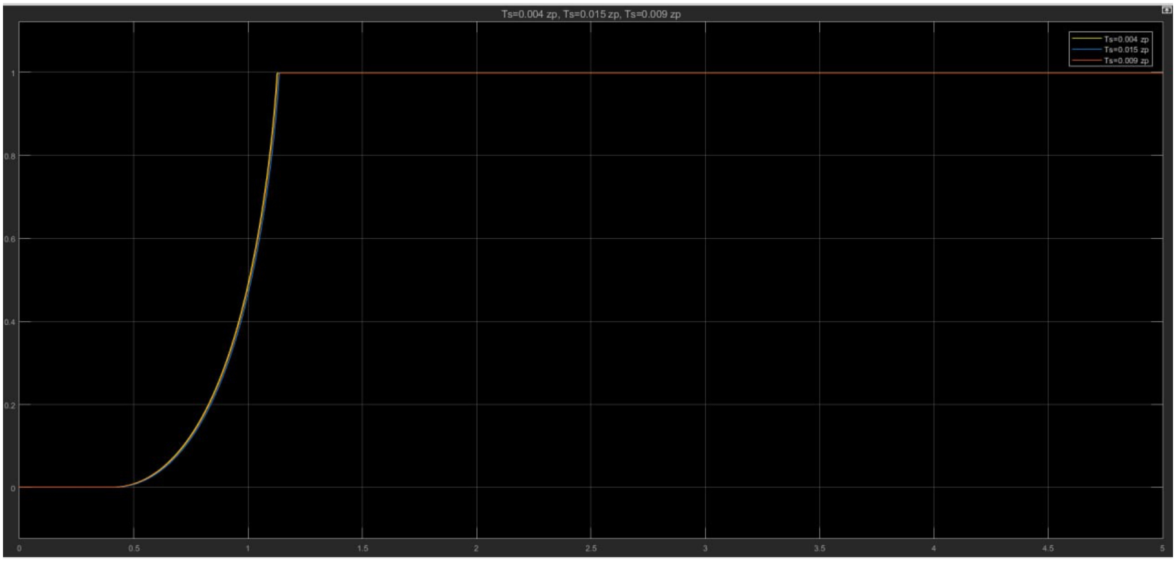


همانطور که میبینیم، بهترین پاسخ به ازای T = 0.004 رخ می دهد.

اکنون با قرار دادن کنترلرهای بدست آمده در سیستم غیر خطی، پاسخ این سیستم را نیز بررسی میکنیم.
مطابق زیر کنترلرها را در سیستم غیرخطی قرار میدهیم:



خروجی های سیستم به شکل زیر خواهند بود:



همانطور که میبینیم، سیستم غیرخطی گسسته سازی شده نیز پایدار است.

گسسته سازی تابع تبدیل پیوسته به روش صفر و قطب تطبیق یافته نیز در محیط متلب، با استفاده از دستور c2d به شکل زیر قابل پیاده سازی است:

3)

zero pole placement

T=0.004

```
G1dpz = c2d(GC,0.004,'matched')
```

T=0.015

```
G2dpz = c2d(GC,0.015,'matched')
```

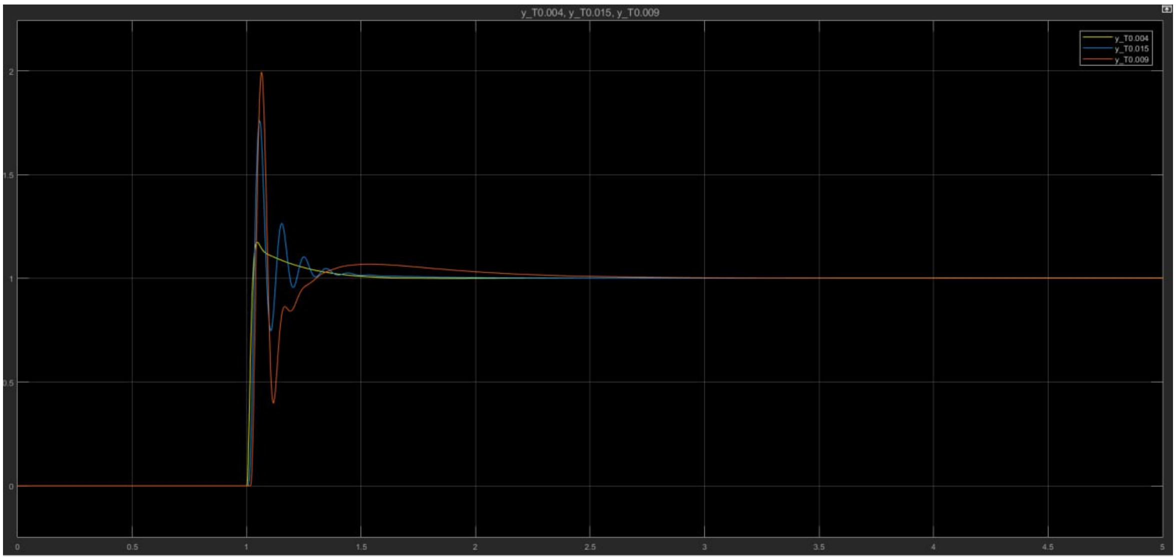
T=0.009

```
G3dpz = c2d(GC,0.009,'matched')
```

توابع تبدیل مربوط به هرکدام از نرخ های نمونه برداری را به صورت زیر تشکیل میدهیم:

```
G1dbl =

  9240 z^2 - 1.818e04 z + 8939
  ----------------------------
       z^2 - 0.4 z - 0.6

Sample time: 0.004 seconds
Discrete-time transfer function.
G2dbl =

  3019 z^2 - 5676 z + 2666
  ------------------------
   z^2 - 0.125 z - 0.875

Sample time: 0.015 seconds
Discrete-time transfer function.
G3dbl =

  4715 z^2 - 9086 z + 4377
  ------------------------
     z^2 - 0.2 z - 0.8

Sample time: 0.009 seconds
Discrete-time transfer function.
```

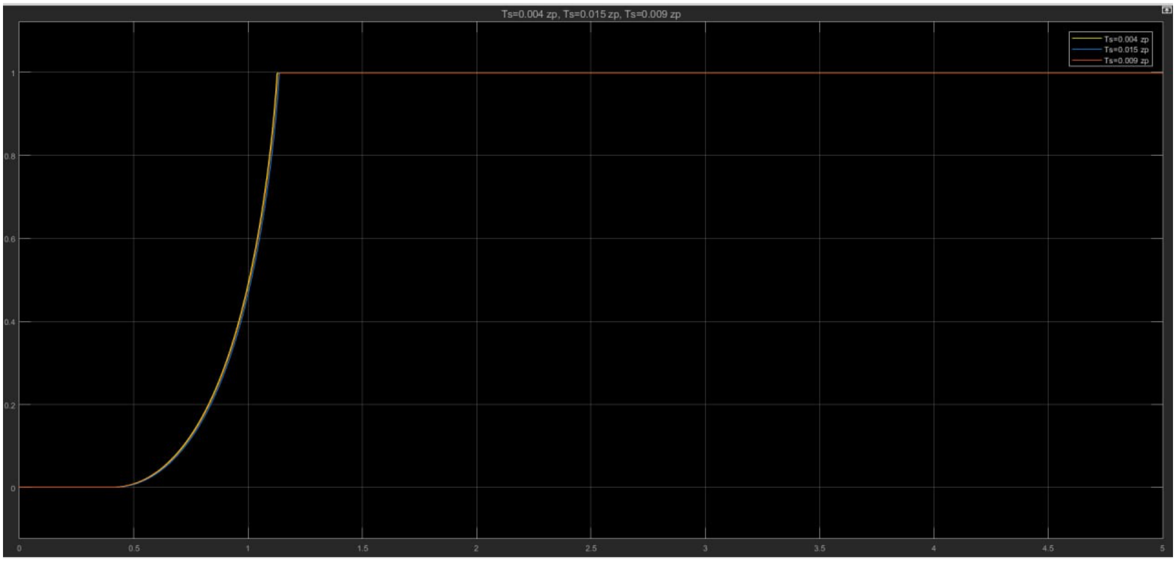
اکنون برای سیستم گسسته سازی شده، پاسخ سیستم را به ازای سه نرخ نمونه برداری مختلف رسم میکنیم:



همان طور که مشاهده می شود، پاسخ پله سیستم با نرخ نمونه برداری برابر ۰.۰۰۴ ثانیه، از سایر سیستم ها بهتر است.

اکنون با قرار دادن کنترلرهای بدست آمده در سیستم غیر خطی، پاسخ این سیستم را نیز بررسی میکنیم.

خروجی های سیستم به شکل زیر خواهند بود:



همانطور که میبینیم، سیستم غیرخطی گسسته سازی شده نیز پایدار است.

بخش ۴

با توجه به بخش قبل با در نظر گرفتن تابع تبدیل پیوسته سیستم و با استفاده از دستور c2d متلب، کدها را به صورت زیر وارد میکنیم تا نتایج مدنظر را در هر بخش بررسی کنیم:

4)

T=0.004

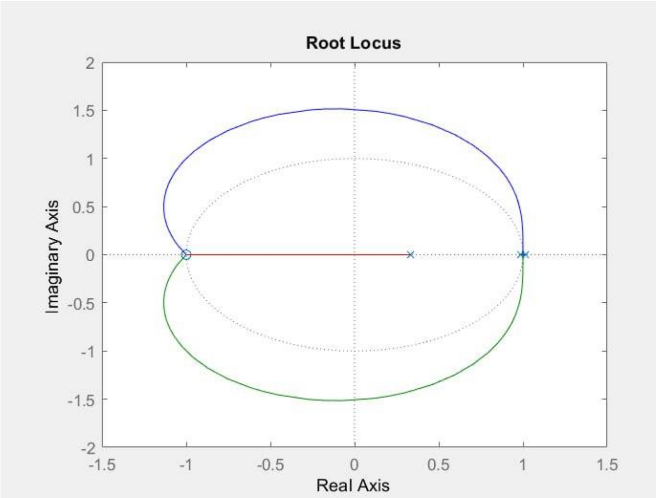
```
Gpz = c2d(G,0.004,'tustin')
rlocus(Gpz)
margin(Gpz)
grid on
fb = bandwidth( Gpz )
```

در این قسمت نرخ نمونه برداری را برابر ۰.۰۰۴ در نظر میگیریم که در بخش قبل بهترین نتیجه‌ها را گرفتیم.

تابع تبدیل گسسته سازی شده سیستم را محاسبه میکنیم که نتیجه به صورت زیر است:

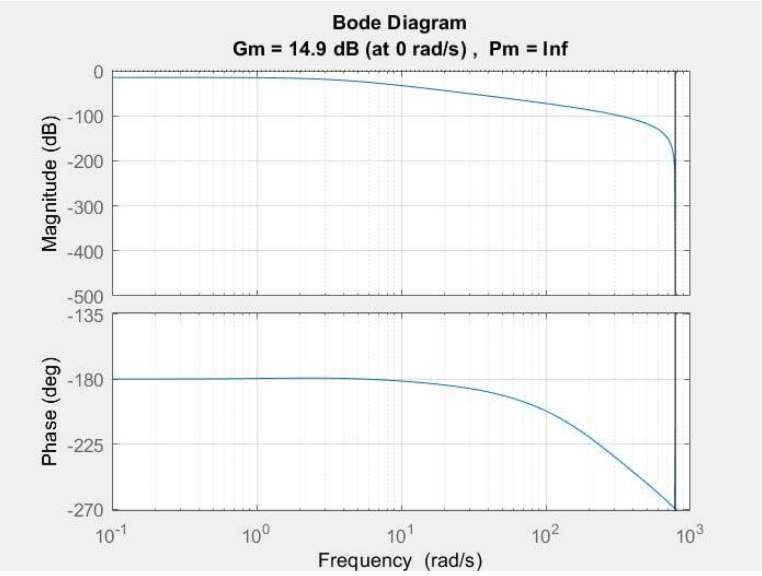
```
Gpz =

  3.471e-06 z^3 + 1.041e-05 z^2 + 1.041e-05 z
                                  + 3.471e-06
  ---------------------------------------------
       z^3 - 2.333 z^2 + 1.666 z - 0.3331

Sample time: 0.004 seconds
Discrete-time transfer function.
```

با استفاده از دستور root locus، نمودار مکان ریشه سیستم را رسم میکنیم که حاصل به صورت زیر است:



همانطور که میبینیم، با تغییرات همواره ۲ قطب خارج از دایره واحد می افتند، بنابراین به ازای هیچ مقدار k سیستم پایدار نمیشود.

نمودار بود سیستم را مطابق زیر رسم میکنیم. حد فاز (PM) و حد بهره (GM) در شکل مشخص شده اند:

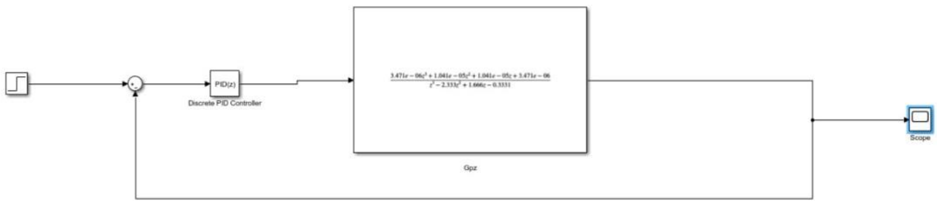


همچنین مقدار پهنای باند به صورت زیر بدست می آید:

```
fb = 2.4368
```

بخش ۵

سیستم را مطابق زیر می بندیم:



با استفاده از PID Tuner مقادیر پارامترهای کنترلر را به نحوی تعیین میکنیم که ویژگی های زمانی سیستم، با نتایج بخش ۲ مطابقت داشته باشد. در تصویر زیر، میتوان مقدار پارامترهای کنترل کننده و پارامترهای پاسخ گذرای سیستم را مشاهده کرد:

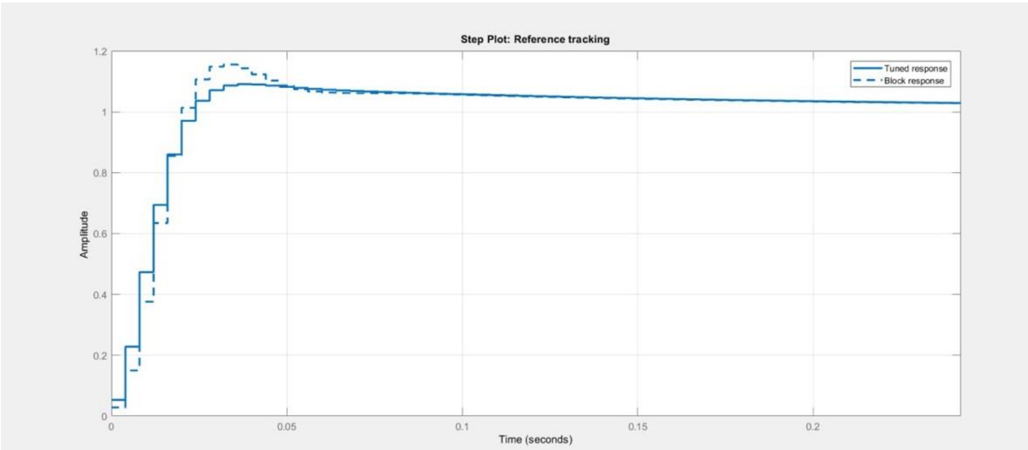
Controller Parameters

	Tuned	Block
P	252.3479	252.3479
I	165.9217	165.9217
D	33.6743	33.6743
N	475.0686	475.0686

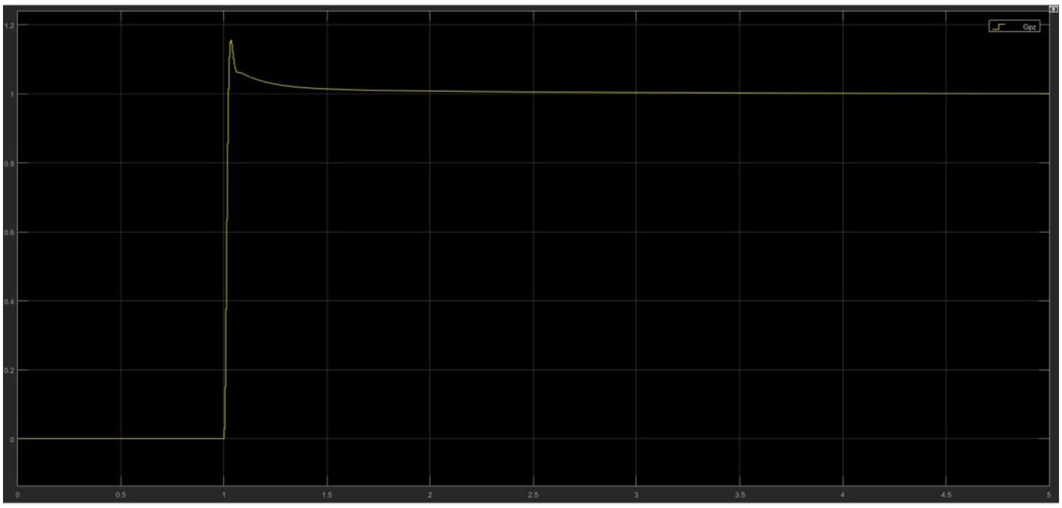
Performance and Robustness

	Tuned	Block
Rise time	0.016 seconds	0.016 seconds
Settling time	0.368 seconds	0.36 seconds
Overshoot	9.09 %	15.6 %
Peak	1.09	1.16
Gain margin	-33.3 dB @ 2.2 rad/s	17.8 dB @ 297 rad/s
Phase margin	66 deg @ 82.6 rad/s	56.6 deg @ 82.6 rad/s
Closed-loop stability	Stable	Stable

پاسخ پله کنترل کننده PID به صورت زیر است:

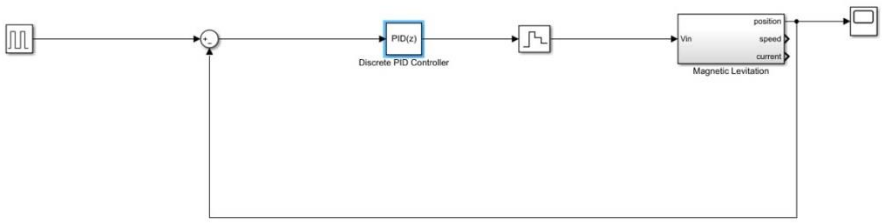


پس از update کردن پارامترها، پاسخ پله سیستم به صورت زیر بدست می آید:

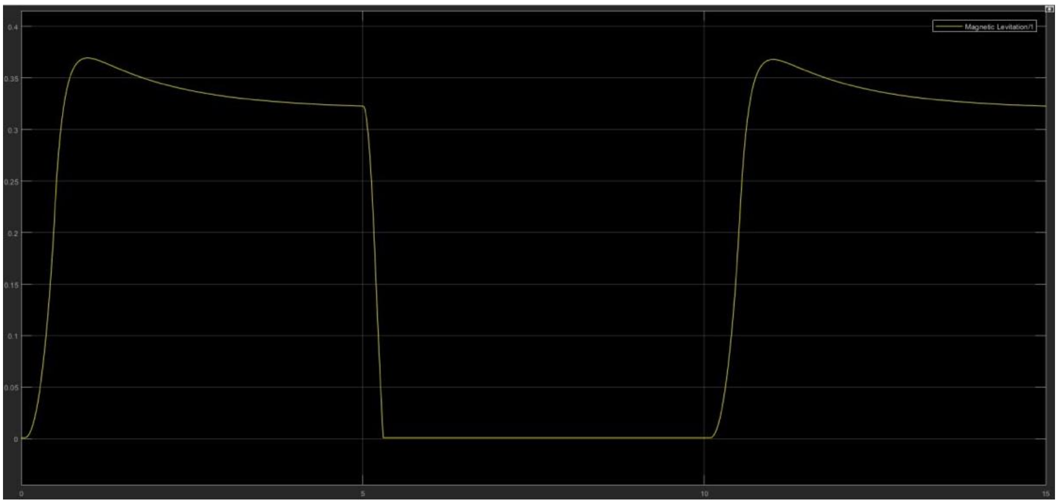


همان طور که میبینیم پاسخ سیستم خطی مطلوب است.

اکنون سیستم غیرخطی را مطابق زیر میبندیم:



پاسخ سیستم غیرخطی را نیز به ازای کنترل کننده بررسی میکنیم:



همانطور که میبینیم، سیستم غیرخطی نیز با استفاده از کنترلر طراحی شده، کنترل شد.

بخش ۶

با توجه به بخش قبل، تابع تبدیل کنترلر بدست آمده را از PID Tuner مطابق زیر استخراج میکنیم:

```
Gdz =

                Ts                  1
  Kp + Ki * ------ + Kd * -----------
               z-1          Tf+Ts/(z-1)

  with Kp = 252, Ki = 166, Kd = 33.7, Tf = 0.0021, Ts = 0.004

Sample time: 0.004 seconds
Discrete-time PIDF controller in parallel form.
```

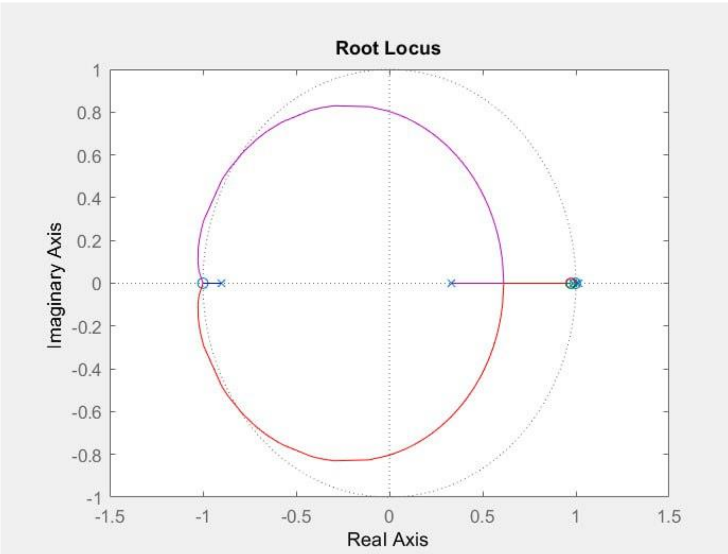
با ضرب تابع تبدیل کنترلر در تابع تبدیل پلنت که از بخش های قبل داشتیم، تابع تبدیل کلی سیستم را مطابق زیر بدست می آوریم:

6)

```
Gdz
Gtz=Gdz*Gpz
rlocus(Gtz)
margin(Gtz)
grid on
```

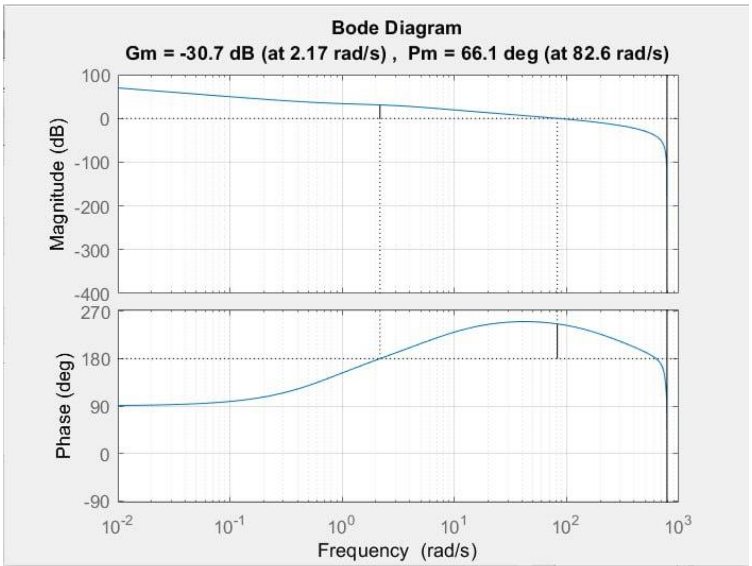
```
Gtz =

  0.0564 z^5 + 0.05807 z^4 - 0.1095 z^3

            - 0.1128 z^2 + 0.05308 z + 0.05474
  ----------------------------------------------

  z^5 - 2.433 z^4 + 0.9981 z^3 + 1.601 z^2

                         - 1.466 z + 0.2999

Sample time: 0.004 seconds
Discrete-time transfer function.
```

نمودار مکان ریشه سیستم کنترل شده:



همان طور که میبینیم، با تغییرات میتوان ناحیه‌ای را پیدا کرد که هر۳ قطب سیستم داخل دایره واحد قرار بگیرند.

نمودار بود سیستم کنترل شده:



همان طور که از روی نمودار بالا و با توجه به مقادیر مشخص شده در شکل میبینیم، حدفاز سیستم دیگر بی‌نهایت نیست و سیستم کنترل شده است، در حالی که حدفاز در سیستم کنترل نشده برابر با بینهایت بود.

بخش ۷

با داشتن تابع تبدیل گسسته سیستم از قسمت های قبل، صفرها و قطب های ناپایدار سیستم را بدست می آوریم.

سپس با در نظر گرفتن ورودی پله، تابع F(z) را مطابق زیر بدست آورده و در نهایت با جایگذاری رابطه در دسترس، تابع تبدیب کنترلر deadbeat را بدست می آوریم.

محاسبات انجام شده به صورت دستی و در متلب به شکل زیر هستند:

$$G_D(z) = \frac{F(z)}{(1-F(z))\,G(z)}$$

پاسخ پله : $R(z) = \dfrac{1}{1-z^{-1}}$

تابع تبدیل سیستم : $G_p(z) = 1.08 \times 10^{-5} z^3 + 3.241 \times 10^{-5} z^2$

$$G_P(z) = \frac{1.08 \times 10^{-5} z^3 + 3.241 \times 10^{-5} z^2 + 3.241 \times 10^{-5} z + 1.08 \times 10^{-5}}{z^3 - 2.282 z^2 + 1.564 z - 0.2821}$$

صفرها و قطب های ناپایدار : $(z - 1.0117)(z + 1.0309)$

پس : $1 - F(z) = (1 - z^{-1})(1 - 1.0117 z^{-1})(1 + 1.0309 z^{-1})$

پس : $F(z) = \dfrac{0.9808 z^2 + 1.0622 z - 1.043}{z^3}$

پس : $G_D(z) = [0.9808 z^5 - 1.176 z^4 - 1.933 z^3 + 3.765 z^2 - 1.931 z + 0.2942] \, / \, [1.08 \times 10^{-5} z^6 + 2.182 \times 10^{-5} z^5 - 1.085 \times 10^{-5} z^4 - 4.415 \times 10^{-5} z^3 - 1.121 \times 10^{-5} z^2 + 2.233 \times 10^{-5} z + 1.126 \times 10^{-5}]$

```
num = [0.0000108 0.00003241 0.00003241 0.0000108];
den = [1 -2.282 1.564 -0.2821];
sys = tf(num, den)

sys_poles = roots(den)
sys_zeros = zero(sys)
```

```
syms z
F = 1 - ((1-z^(-1))*(1-1.0117*z^(-1))*(1+1.0309*z^(-1)))
expanded_expr = expand(F)
```

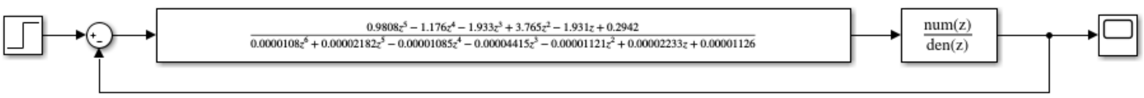
```
num_f = [0 0.9808 1.0622 -1.043]
den_f = [1 0 0 0]
Fz = tf(num_f, den_f)
```

```
Gd = Fz / ((1 - Fz) * sys)
```

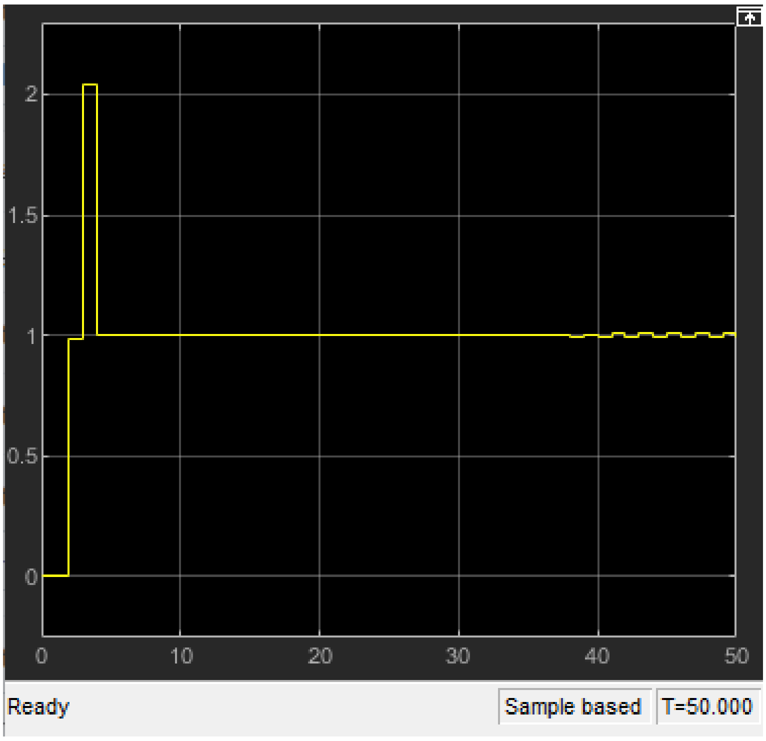
در نهایت کنترلر به شکل زیر بدست می آید:

```
Gd =

                0.9808 s^8 - 1.176 s^7 - 1.933 s^6 + 3.765 s^5 - 1.931 s^4 + 0.2942 s^3
  -------------------------------------------------------------------------------------------
  1.08e-05 s^9 + 2.182e-05 s^8 - 1.085e-05 s^7 - 4.415e-05 s^6 - 1.121e-05 s^5 + 2.233e-05 s^4
                                                                              + 1.126e-05 s^3
```

با جاگذاری در سیمولینک داریم:

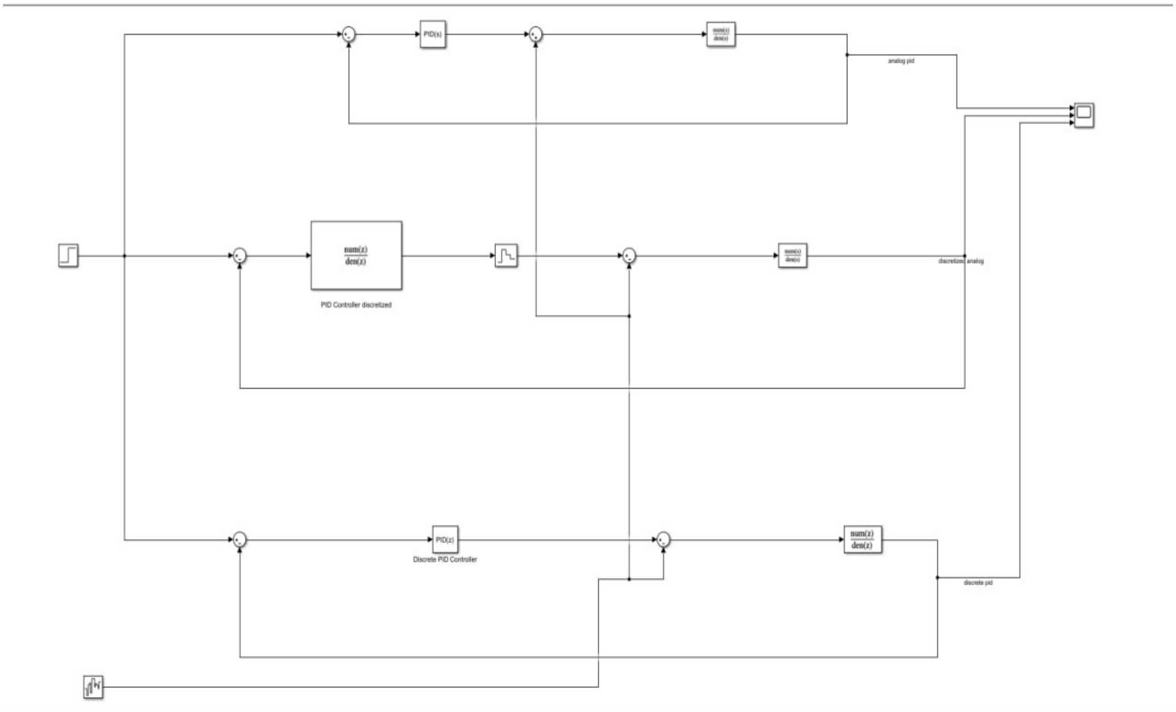


پاسخ سیستم کنترل شده به شکل زیر است که مطابق انتظار است:



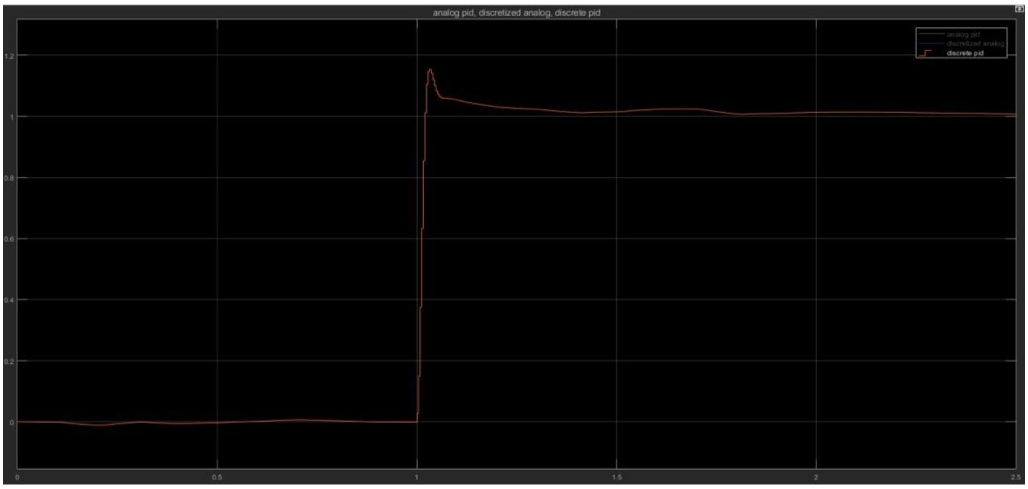
بخش ۸

مطابق شکل زیر، سه سیستم که اولی شامل پلنت و کنترل کننده انالوگ، دومی شامل کنترلر انالوگ گسسته شده و نمونه بردار و پلنت انالوگ، و سیستم سوم شامل پلنت گسسته سازی شده و کنترل کننده دیجیتال است را درنظر میگیریم و به پلنت هر سه سیستم، نویز سفید به عنوان اغتشاش وارد میکنیم.

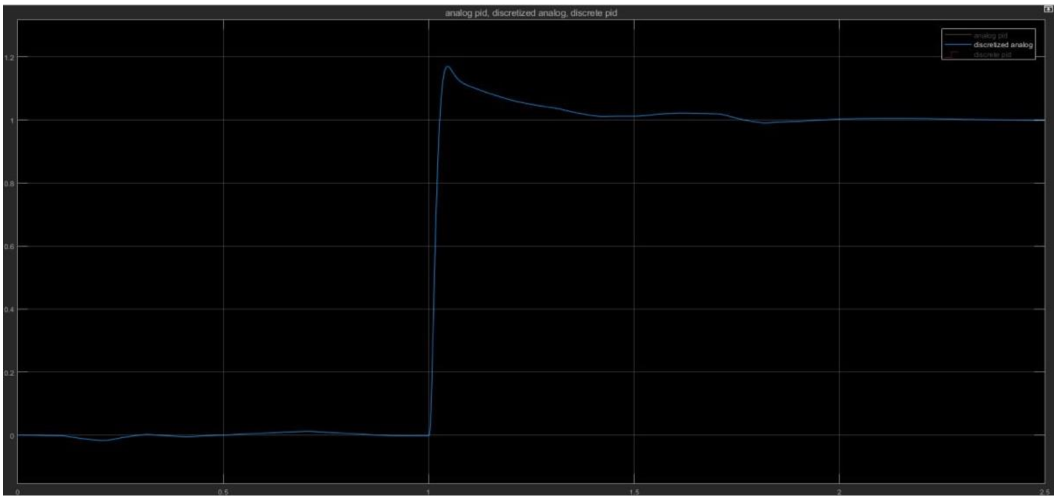


نتایج به صورت زیر هستند:

خروجی کنترل کننده و پلنت دیجیتال که همانطور که میبینیم به خوبی نویز را کنترل می کند.

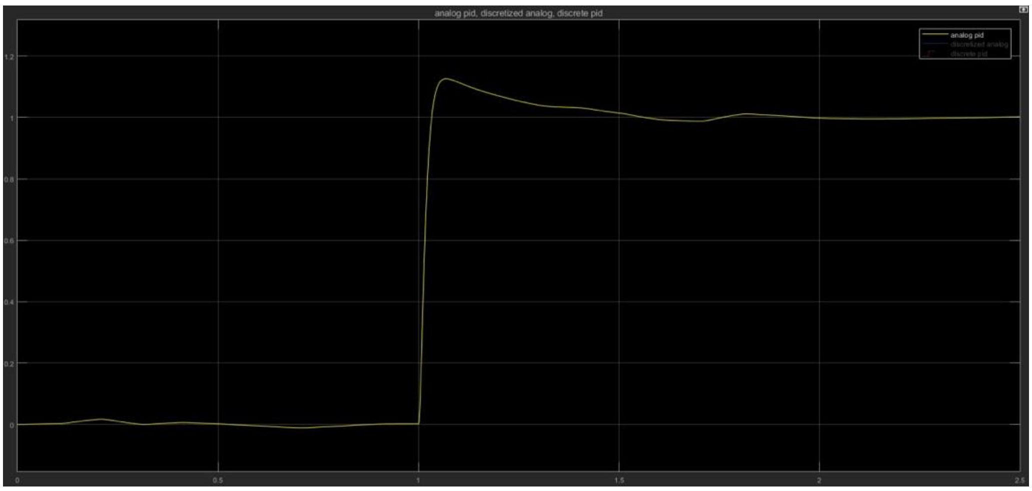


خروجی کنترل کننده گسسته شده، نمونه بردار و پلنت انالوگ:

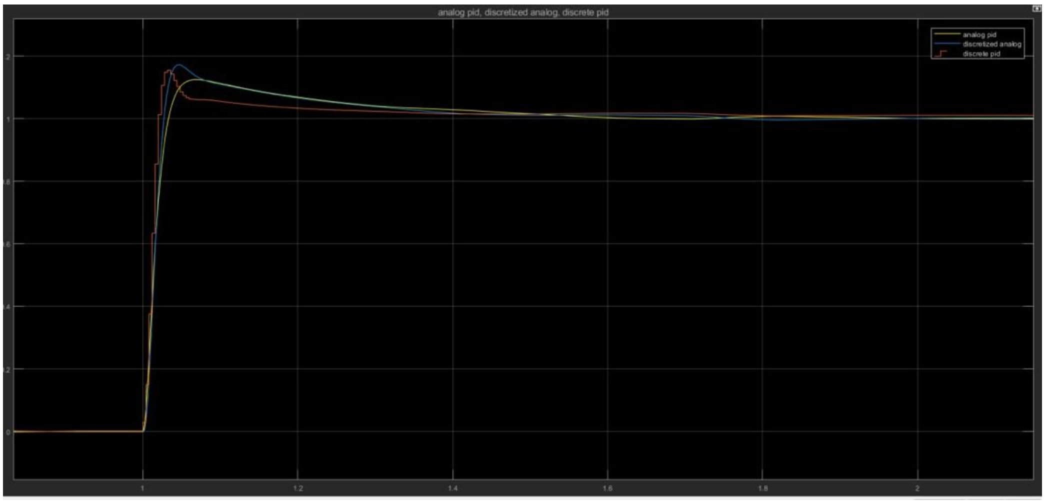


همانطور که میبینیم به خوبی نویز را کنترل می کند.

و در نهایت، خروجی پلنت و کنترلر انالوگ:



با قرار دادن نمودار هر سه سیستم کنار هم داریم:



همان طور که از روی نمودارهای خروجی مشاهده می شود، نتیجه سیستم آخر از سایر سیستم ها بهتر بود.

بخش ۹

فضای حالت سیستم را با استفاده از ماتریس‌های بدست آمده در بخش ۱ تشکیل داده و با استفاده از دستور زیر در متلب گسسته سازی میکنیم:

Part 9:

T = 0.004

```
sys = ss(A,B,C,D);
sysd=c2d(sys,0.004);
[Az,Bz,Cz,Dz] = dssdata(sysd)
```

فضای حالت سیستم گسسته سازی شده، به شکل زیر بدست می آید:

```
Az = 3×3
      1.0001    0.0040    0.0001
      0.0576    0.9993    0.0329
           0         0    0.3679

Bz = 3×1
      0.0000
      0.0038
      0.1264

Cz = 1×3
       1     0     0

Dz = 0
```

سپس، با استفاده از دستورهای مربوطه در متلب، ماتریس های کنترل پذیری و رویت پذیری سیستم گسسته شده را مطابق زیر تشکیل می‌دهیم:

Controllability

```
M= ctrb(Az,Bz)
rank_m=rank(M)
```

Observability

```
N= obsv(Az,Cz)
rank_n=rank(N)
```

ماتریس کنترل پذیری مطابق زیر بدست می آید:

```
M = 3×3
      0.0000    0.0000    0.0001
      0.0038    0.0080    0.0095
      0.1264    0.0465    0.0171

rank_m = 3
```

همان طور که میبینیم، ماتریس بالا مرتبه کامل سطری است، پس سیستم کنترل پذیر است.

ماتریس رویت پذیری مطابق زیر بدست می آید:

```
N = 3×3
      1.0000         0         0
      1.0001    0.0040    0.0001
      1.0005    0.0080    0.0002

rank_n = 3
```

همان طور که میبینیم، ماتریس بالا مرتبه کامل ستونی است، پس سیستم رویت پذیر است.

بخش ۱۰

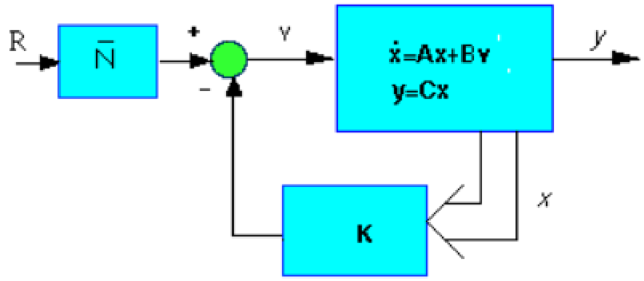
برای طراحی فیدبک حالت به نحوی که سیستم پاسخ مرده نوش داشته باشد، میدانیم که در کنترلر dead beat داریم: z^n=0

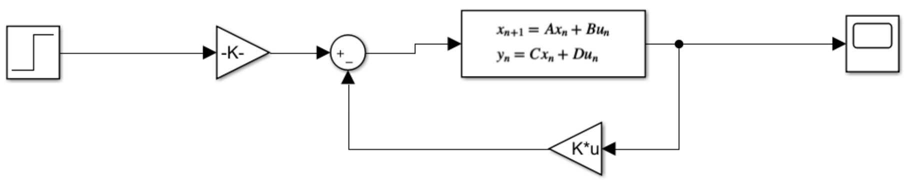
بنابراین قطب های مطلوب را برابر 0 میگذاریم. سپس ماتریس فیدبک حالت k را به روش Ackerman با استفاده از دستور زیر در متلب محاسبه می کنیم:

```
p = [0 0 0];
K = acker(Az,Bz,p)
```

فیدبک حالت k مطابق زیر بدست می آید:

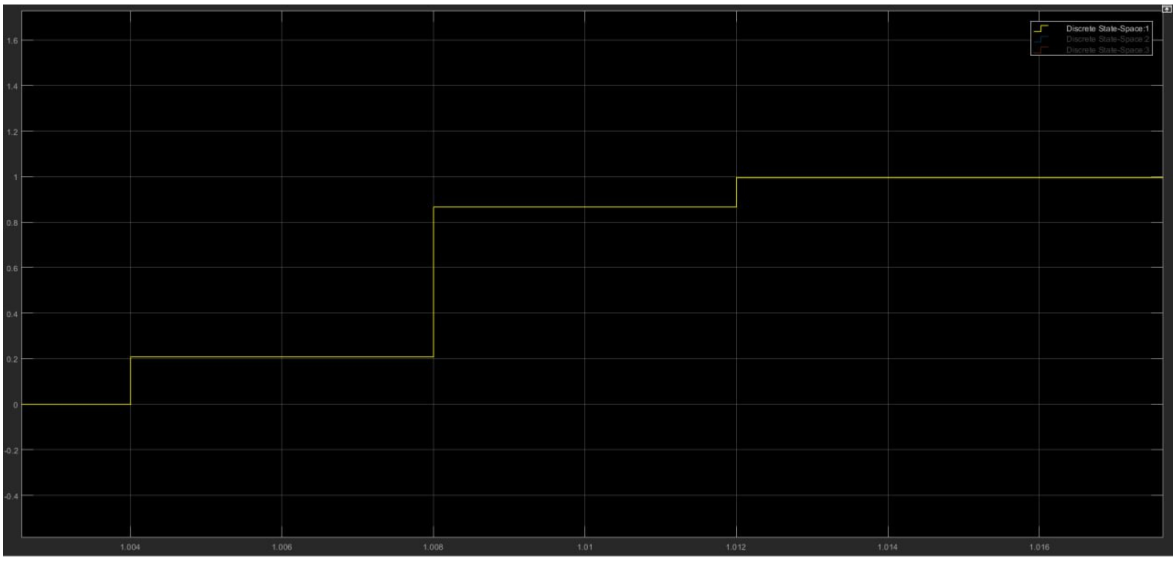
پس یک سیستم مطابق زیر داریم که آن را در محیط سیمولینک به صورت زیر شبیه سازی میکنیم:





هم چنین پیش از سیستم یک gain به مقدار ۳۷۷۳۶ قرار میدهیم تا مقدار نهایی سیستم را به مقدار پله برساند.

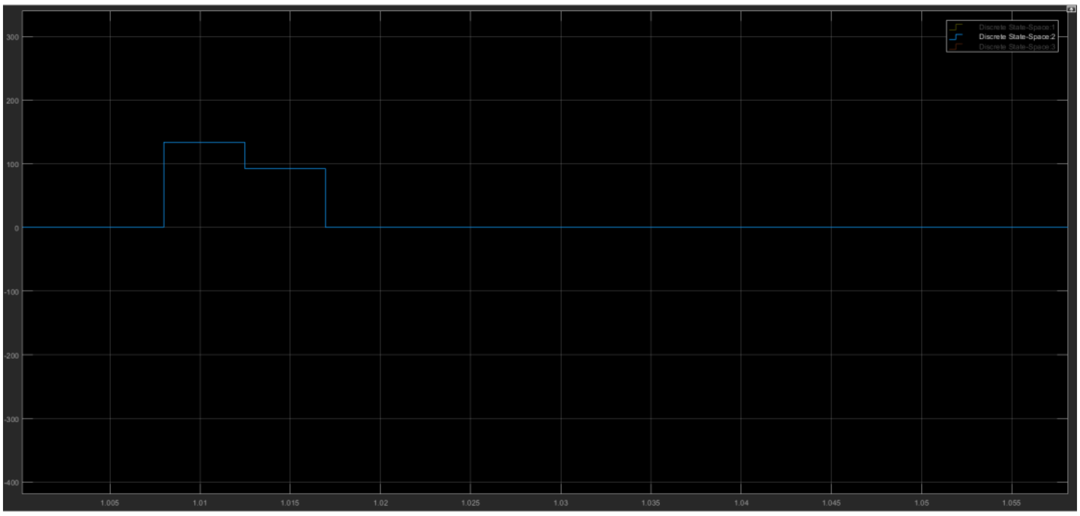
در نهایت، خروجی سیستم به صورت زیر در می آید:



همانطور که میبینیم پاسخ سیستم پس از 3 مرحله به مقدار نهایی خود میرسد.

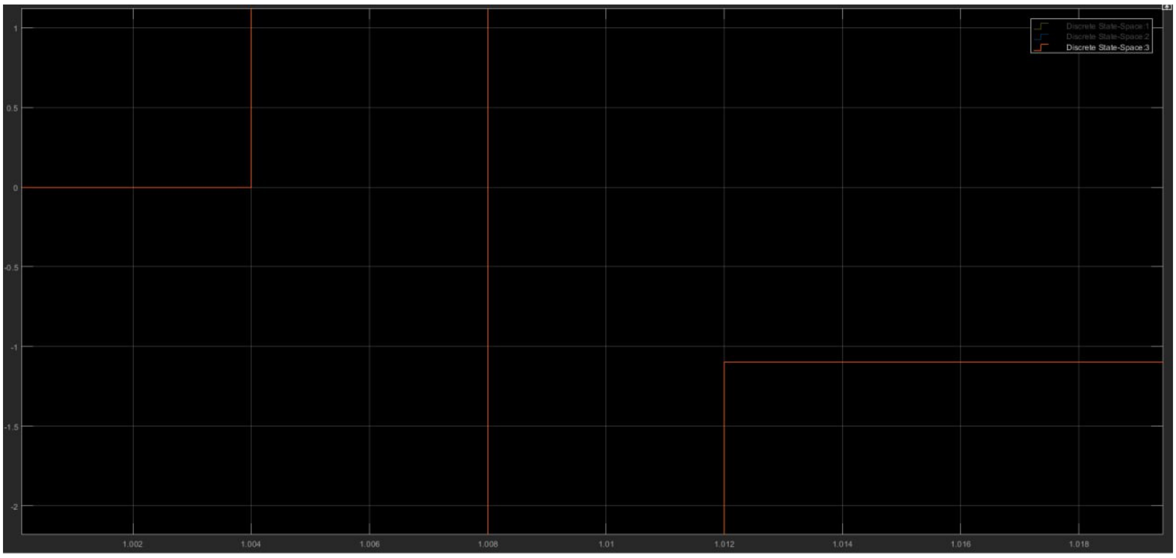
همچنین ارتفاع نهایی گوی با در نظر گرفتن مقداری خطای حالت ماندگار به حدود 1 میرسد.

همچنین نمودار حالت x2 (سرعت گوی) به صورت زیر است:



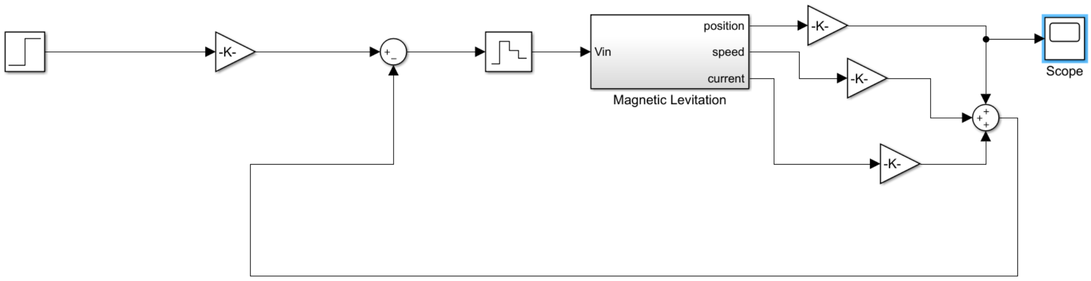
همانطور که میبینیم، ابتدا در یک گام سرعت افزایش یافته و سپس در دو گام به 0 میرسد.

برای حالت x3 (جریان سیم) مطابق نمودار زیر داریم:

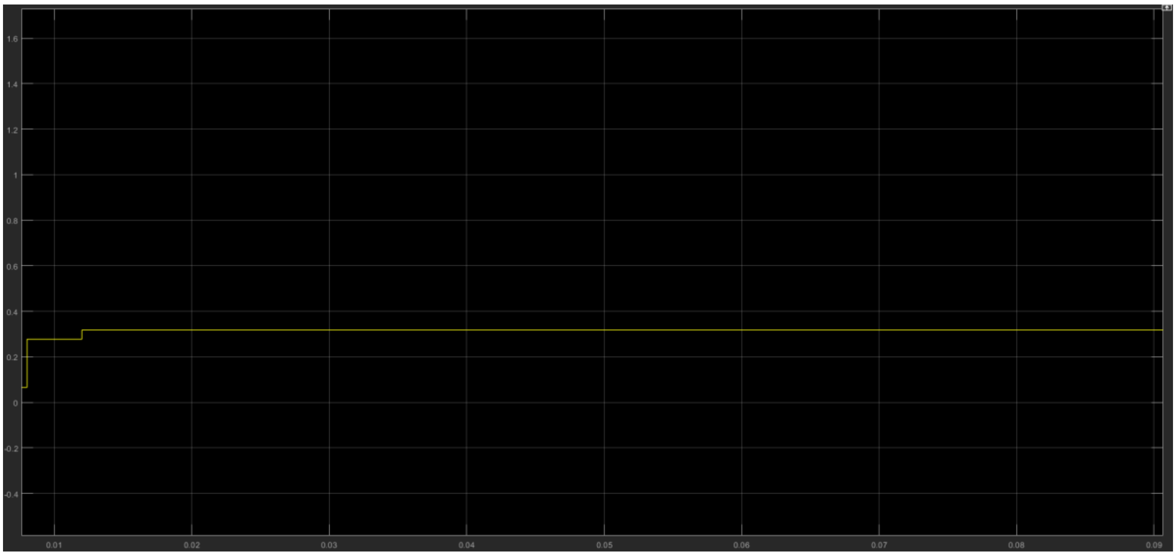


همانطور که مشاهده می‌شود، جریان سیم ابتدا در یک گام به مقدار مثبت زیادی رسیده و سپس مقدار منفی زیادی و درنهایت به مقدار ۱- میرسد.

حال کنترلر را برای سیستم غیرخطی شبیه سازی میکنیم:



ضرب ماتریس K را به صورت ترکیب ضرب elementwise و sum انجام میدهیم. همچنین دقت میکنیم که ضرایب به طور درست در متغیر حالت مربوط شان ضرب شوند.



در این حالت نیزی سیستم به مقدار مطلوب میرسد.

بخش ۱۱

ابتدا با استفاده از خاصیت دوگان، یک تخمین گر مرتبه کامل برای سیستم طراحی میکنیم. ابتدا مقادیر ویژه ماتریس ضرایب را بدست آورده و قطب های جدید را به نحوی قرار میدهیم که رویتگر از سیستم سریعتر باشد.

```
ans =
    1.0149
    0.9845
    0.3679
```

11)

```
eig(Az)
```

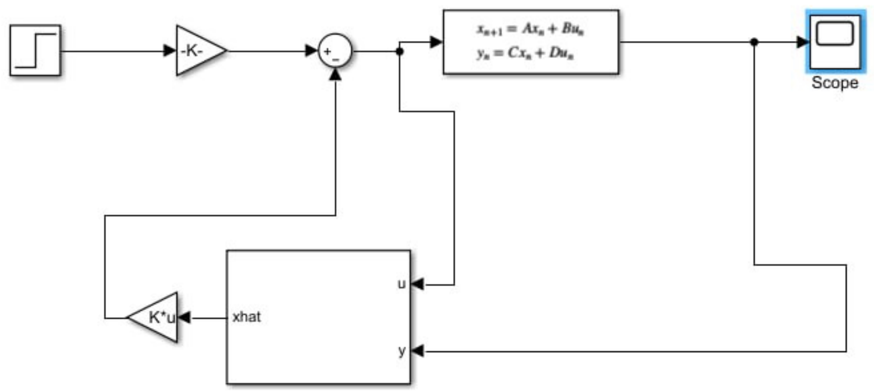
the poles of the observer should be smaller than the system (observer must be faster then the system)

```
po=[0.003 0.002 0.001];
L=place(Az',Cz',po);
L=L'
```

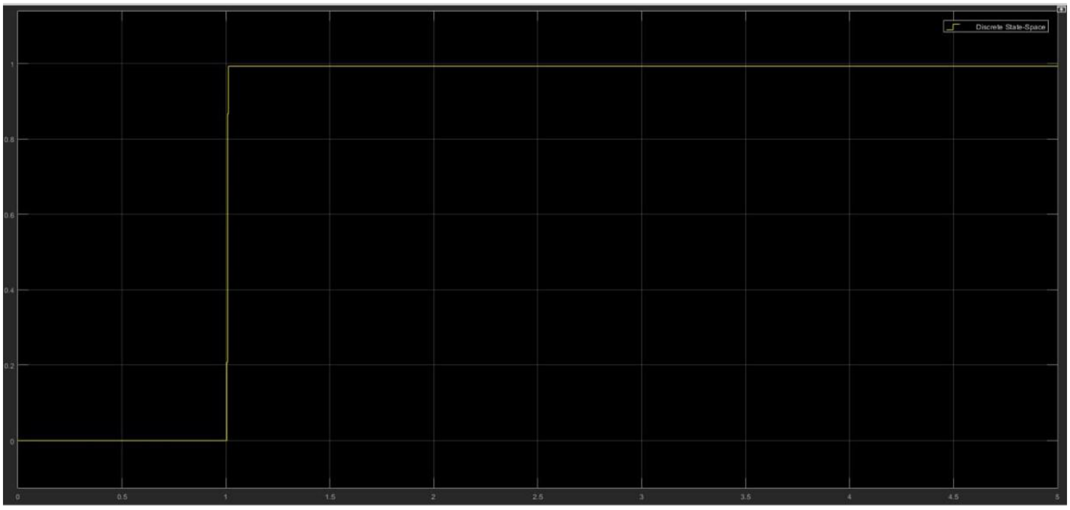
در سیمولینک از بلوک Luenberger Observer استفاده میکنیم که معادله زیر را پیاده سازی میکند:

$$\hat{x}(k+1) = A_d\hat{x}(k) + B_d u(k) + L_d(y(k) - \hat{y}(k)),$$

در نتیجه در سیستم نهایی داریم:



در نهایت برای خروجی سیستم داریم:



همان طور که میبینیم، با طراحی رویتگر و با فرض در دسترس نبودن تمام حالت های سیستم، نتیجه نهایی مطابق قسمت قبل و به خوبی کنترل شد که مطلوب ماست.