



دانشگاه علم و صنعت ایران



گزارش پروژه پایانی کنترل صنعتی

استاد: دکتر سهیل گنجه فر

نویسنده: سید مهدی موسویون - ۹۹۴۱۳۱۳۶

فهرست مطالب

Tasks	۳
شرح مسئله	۴
شناسایی سیستم و وارد کردن آن در متلب و سیمولینک	۵
شناسایی مدل سه جزئی و چهار جزئی	۷
شناسایی مدل سه جزئی	۷
شناسایی مدل چهار جزئی	۱۰
مقایسه پاسخ های بدست آمده از شناسایی	۱۲
اطلاعات نقطه نهایی سیستم	۱۵
اعمال فیدبک رله	۱۵
تعیین پارامترهای نقطه نهایی	۱۷
PID طراحی	۱۹
طراحی به روش زیگلر - نیکولز (حوزه زمان)	۱۹
طراحی به روش زیگلر - نیکولز (حوزه فرکانس)	۲۵
طراحی به روش زیگلر - نیکولز (تعیین یافته)	۲۹
طراحی به روش تنظیم	۳۴
مقایسه تمامی کنترل کننده ها	۳۸
طراحی کنترل کننده PID به همراه تخمین گر اسمیت	۴۰
با اعمال سیستم اصلی	۴۱
طراحی با مدل های شناسایی شده	۴۴
طراحی با مدل سه جزئی	۴۴
طراحی با مدل چهار جزئی	۴۶
شبیه سازی ها	۴۹

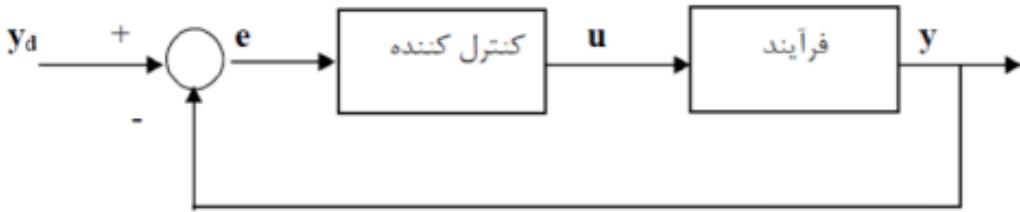


Tasks

سیستم زیرا در نظر بگیرید:

$$\frac{131.8(S+1)e^{-2.5S}}{(S+2.5)^4(S+1.5)^3}$$

- ۱- برای سیستم فوق مدل ۳ جزئی و ۴ جزئی را شناسائی نمایید.
- ۲- پاسخ پله‌ی مدل‌های شناسائی شده را همراه با سیستم اصلی رسم نموده و مقایسه نمایید.
- ۳- اطلاعات نقطه نهایی سیستم را به روش فیدبک رله بدست آورده، پاسخ ورودی خروجی سیستم را بطور هم زمان رسم و نتایج را تحلیل نمایید.
- ۴- مطابق با لوگ دیاگرام زیر، برای سیستم کنترل کننده‌ی حلقه بسته کنترل کننده PID را به روش‌های زیر طراحی نموده و تمامی حالت‌ها را با یکدیگر مقایسه نمایید.



- ۱-۴) به روش زیگلر نیکولز در حوزه‌ی زمان یک کنترل کننده PID برای سیستم طراحی نموده و پاسخ سیستم را به ورودی پله واحد رسم نمایید. ضرایب کنترل کننده را برای پاسخ مناسب تنظیم نمایید. (تصویر سعی و خطای)
- ۲-۴) به روش زیگلر نیکولز در حوزه فرکانس یک کنترل کننده PID برای سیستم طراحی نموده و پاسخ سیستم را به ورودی پله رسم نمایید. ضرایب کنترل کننده را برای پاسخ مناسب تنظیم نمایید. (تصویر سعی و خطای)
- ۳-۴) به روش زیگلر نیکولز تعمیم یافته برای نقطه 61° ، $r_b = 0.41$ ، $\varphi_b = 61^\circ$ یک کنترل کننده PID برای سیستم طراحی نموده و پاسخ سیستم را به ورودی پله رسم نمایید. ضرایب کنترل کننده را برای پاسخ مناسب تنظیم نمایید. (تصویر سعی و خطای)
- ۴-۴) به روش تنظیم λ یک کنترل کننده PID مناسب برای سیستم طراحی نموده و پاسخ سیستم را به ورودی پله رسم نمایید. ضرایب کنترل کننده را برای پاسخ مناسب تنظیم نمایید. (تصویر سعی و خطای)

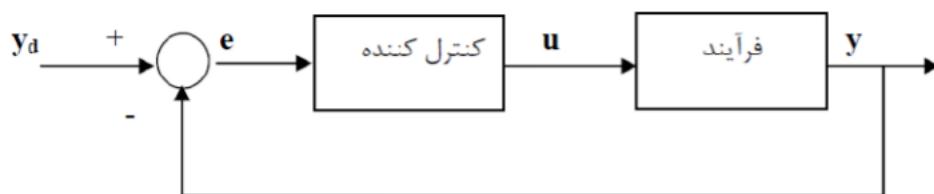
- ۵- کنترل کننده PID طراحی شده در گام ۴-۴ را به همراه پیش‌بینی کننده اسمیت بکار گرفته، پاسخ سیستم را به ورودی پله رسم نمایید. این مساله را در دو حالت حل نمایید. (الف) دینامیک سیستم بطور کامل معلوم است. (ب) اطلاعات ما از مدل سیستم همان مدل ۳ جزئی شناسائی شده می‌باشد.

سیستم زیرا در نظر بگیرید:

$$\frac{131.8(s+1)e^{-2.5s}}{(s+2.5)^4(s+1.5)^3}$$

۱- شرح مسئله

همانطور که در معادله بالا پیداست این سیستم یک سیستم است که در مخرج مرتبه ۷ و در صورت مرتبه ۱ و نیز به صورت تاخیر دار $e^{-2.5s}$ میباشد. از ما در این مسئله خواسته شده با مدل های سه جزئی و چهار جزئی تخمین ها و نیز شناسایی مناسب را برای این سیستم انجام دهیم. پاسخ پله ای مدل های شناسایی شده را همراه با سیستم اصلی رسم نموده و مقایسه نماییم. ما این کار را به روش تئوری آموزش داده شده انجام خواهیم داد که در طول گزارش بصورت کامل شرح شده است. در ادامه خواسته شده تا با استفاده از روش فیدبک رله سیستم را به نقطه نهایی برد و پارامتر های مورد نظر را بدست آوریم و تحلیل نماییم. سپس با استفاده از روش لاندا که روش زیگلر نیکولز در حوزه زمان و فرکانس و نیز زیگلر نیکولز تعیین یافته و همچنین با استفاده از روش گر اسمنیت مطابق شکل زیر کنترل کننده طراحی کنیم. در نهایت خواسته شده تا سیستم را با استفاده از تخمین گر اسمنیت تخمین زده و در دو حالت دینامیک سیستم بطور کامل معلوم است و اطلاعات ما از مدل سیستم همان مدل جزئی شناسایی شده می باشد بررسی نماییم.

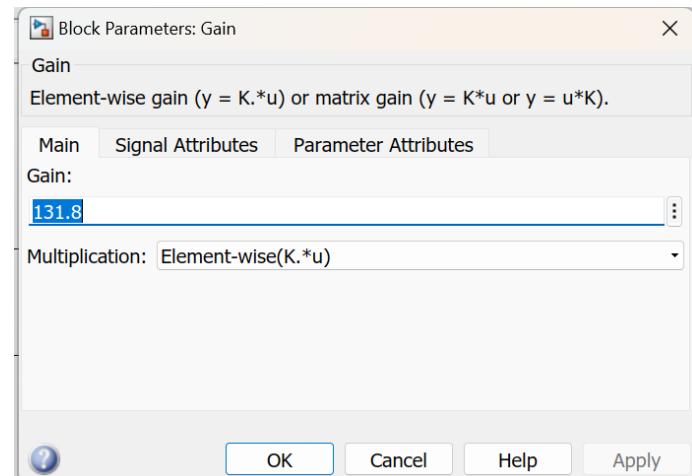
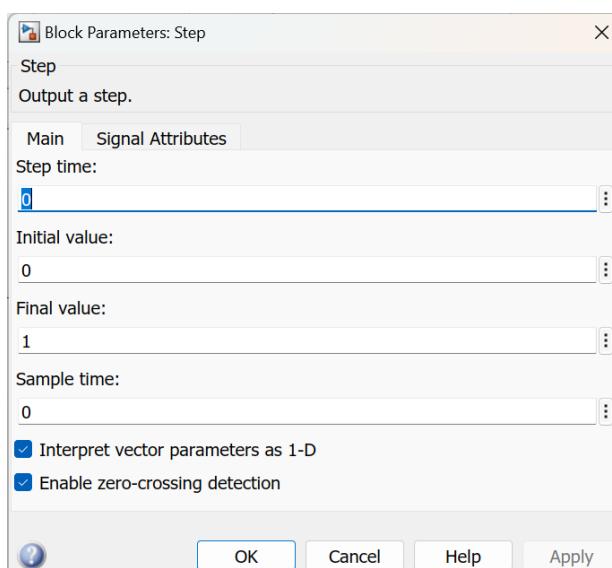
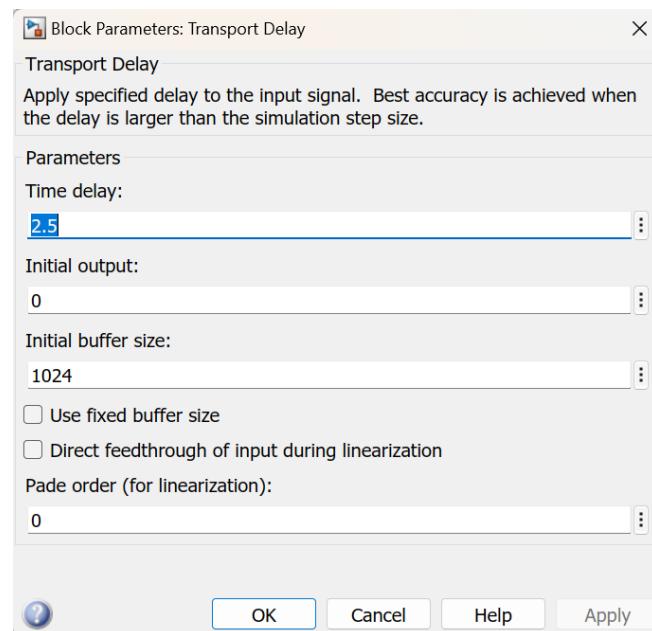
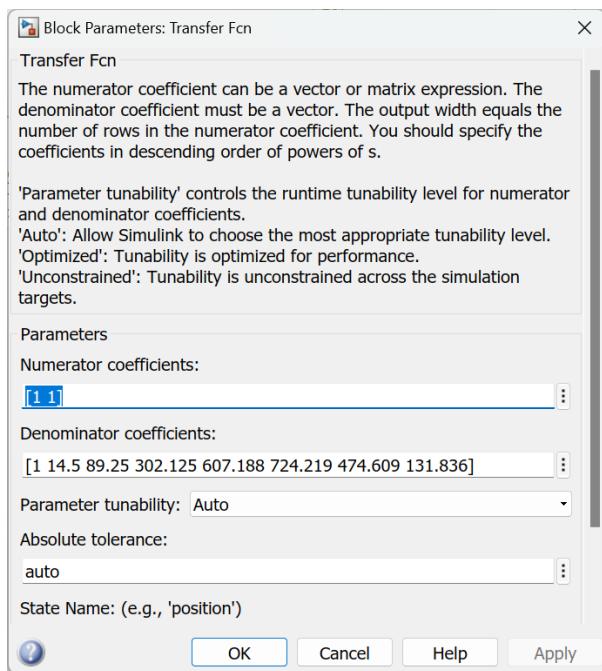


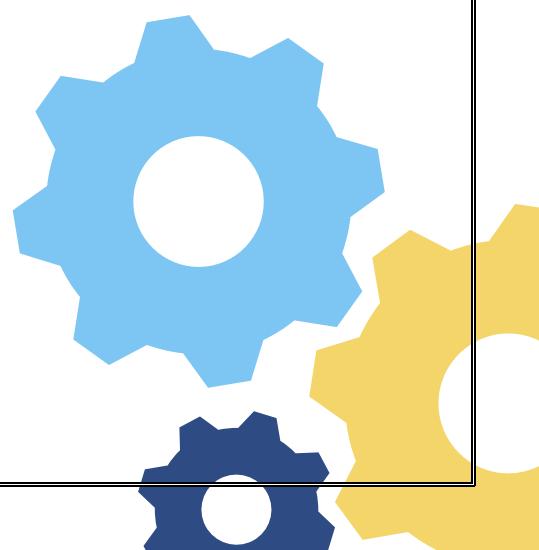
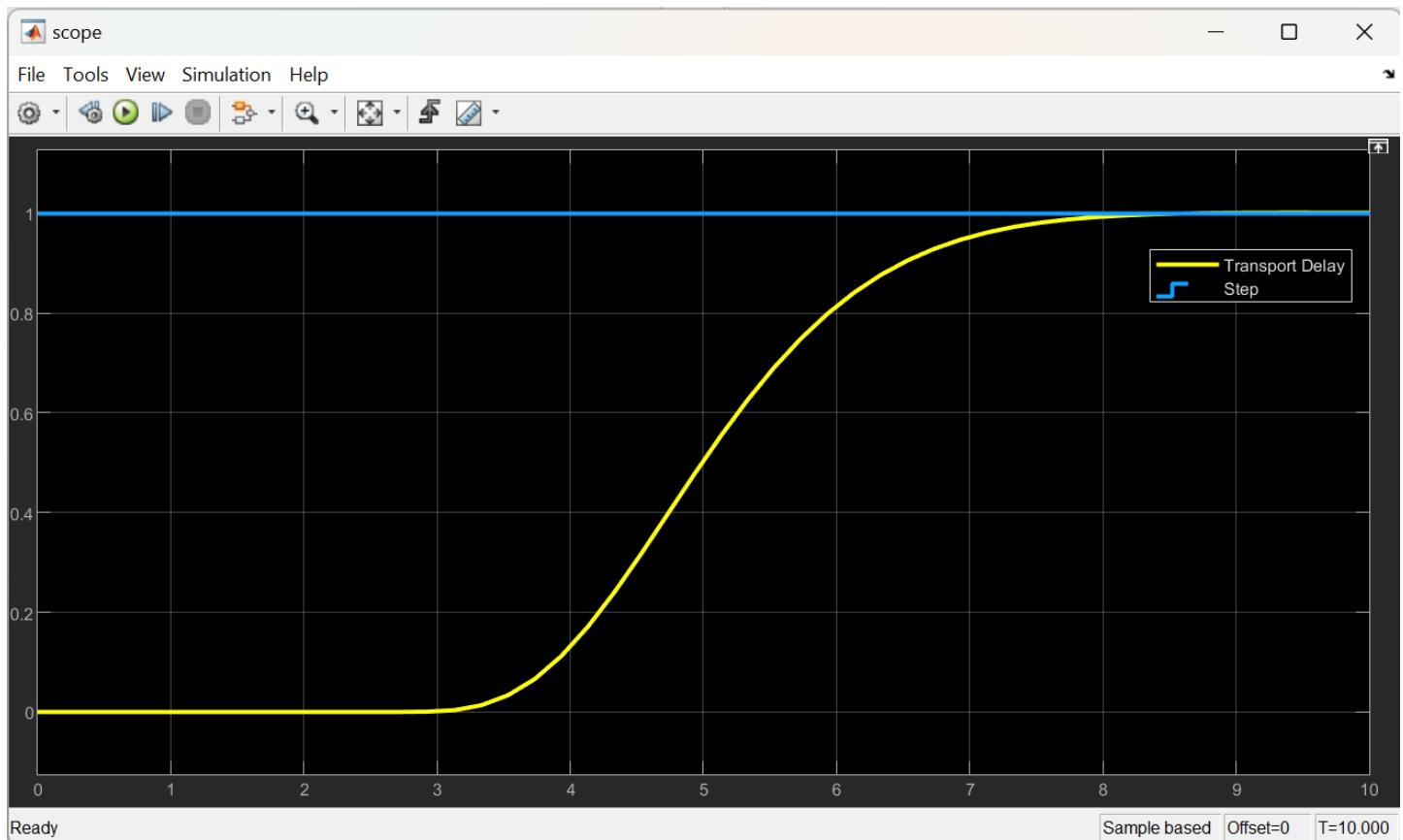
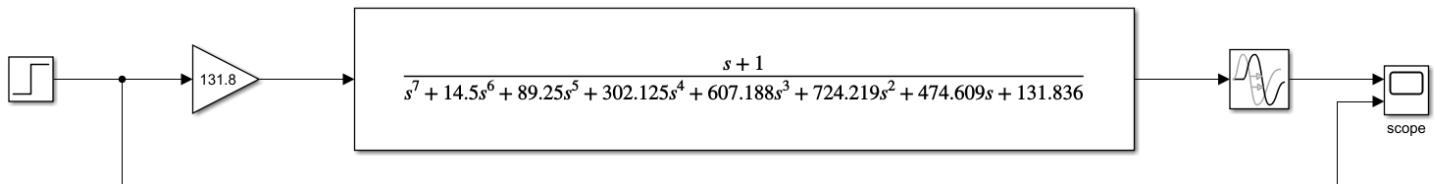
ابتدا مخرج معادله را باز میکنیم:

$$\frac{131.8s + 131.8}{s^7 + 14.5s^6 + 89.25s^5 + 302.125s^4 + 607.188s^3 + 724.219s^2 + 474.609s + 131.836}$$

۲- شناسایی سیستم و وارد کردن آن در مطلب و سیمولینک

ابتدا نرم افزار مطلب را باز کرده و سپس وارد بخش سیمولینک مطلب میشویم. مدل گسترده شده مسئله را در قسمت Transfer Fcn وارد میکنیم. سپس با بلوک Gain عدد ۱۳۱.۸ که ضریب ما هست را اضافه میکنیم. از آنجا که میخواهیم مدل های فوق را شناسایی نماییم نیاز به رسم پاسخ پله سیستم خواهیم داشت پس یک بلوک پله به آن اضافه میکنیم که Step Time آن ۰ است. نهایتا با یک بلوک Scope تاخیر تابع را اعمال میکنیم و به وسیله یک آن را مشاهده میکنیم.





۳- شناسایی مدل سه جزئی و چهار جزئی

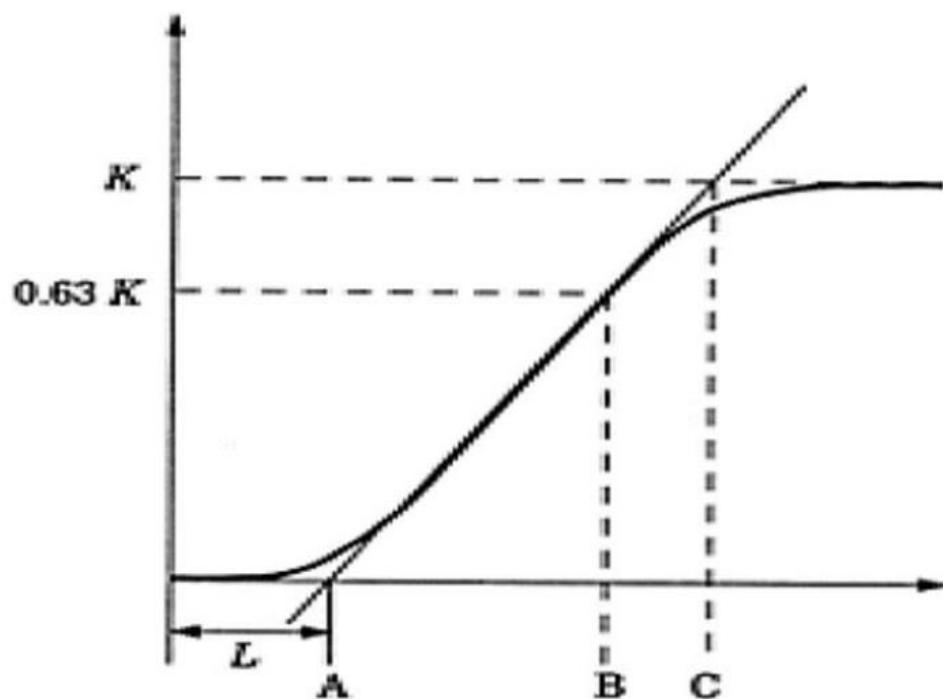
۱-۳ شناسایی مدل سه جزئی

میدانیم که فرم کلی مدل سه جزئی بصورت زیر است:

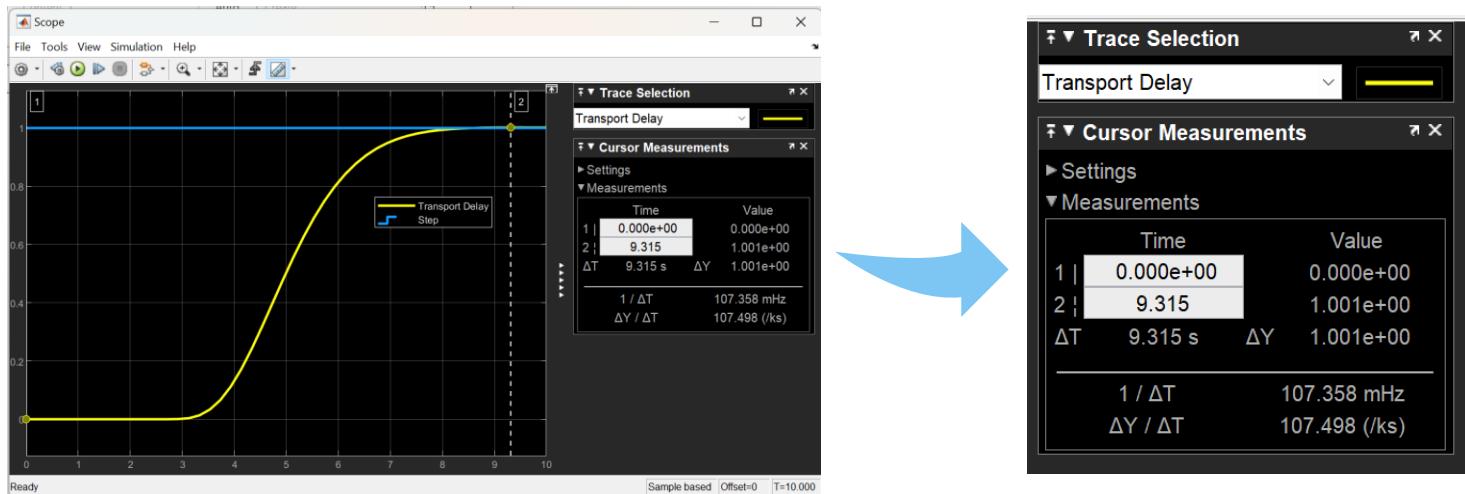
$$G(s) = \frac{K}{1+sT} e^{-sL}$$

حال ما باید با توجه به دانش تئوری و همچنین با استفاده از نرم افزار متلب مقادیر $K - T - L$ را بدست بیاوریم.

مطابق شکل زیر در جایی که نمودار به سمت ثابت شدن میرود K بدست میاید.



پس با استفاده از نرم افزار مطلب و قسمت Measuring سیمولینک اندازه K را بدست می آوریم.

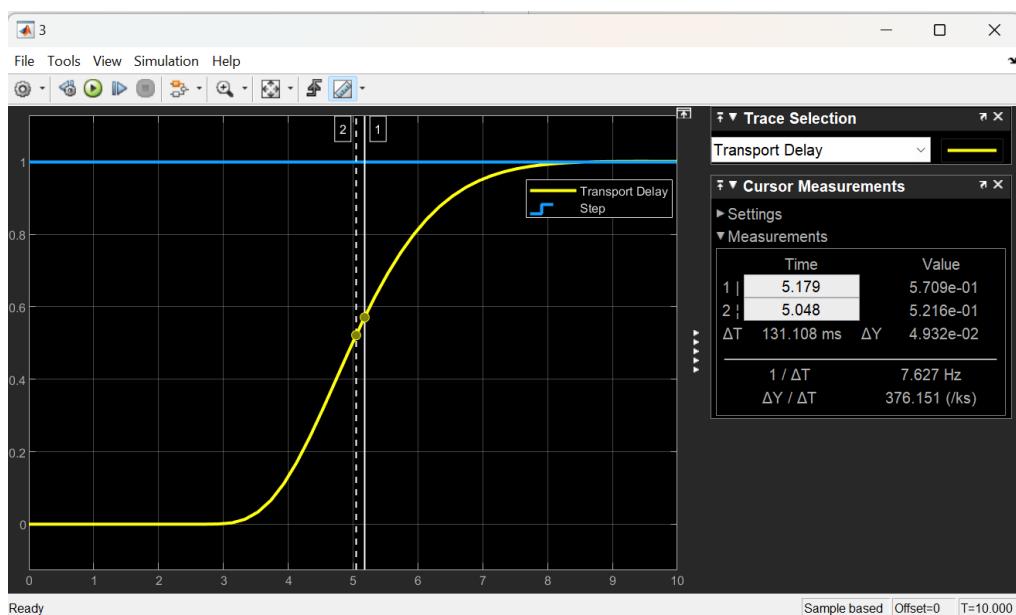


همانگونه که مشاهده میشود عدد ما 1.001 میباشد

$$K = 1.001$$

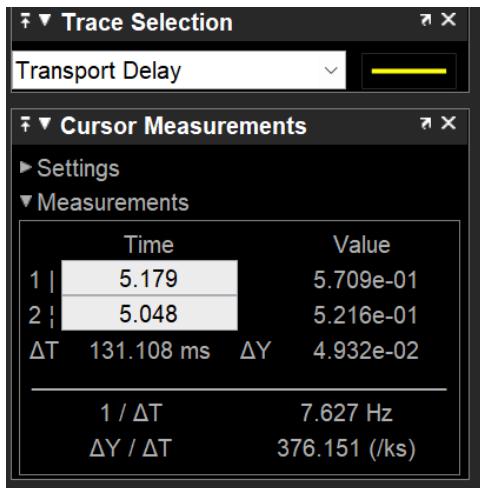
حال با انتخاب دو نقطه بسیار نزدیک به هم (حالت مشتقی) در جایی که بیشترین شیب نمودار دیده می

شود سعی می شود که شیب خط بدست بیاید:



● محاسبه شیب خط:

با نتایج بدست آمده زیر سعی به تخمین معادله خط مماس (با بیشترین شیب) می شود:



$$\frac{\Delta y}{\Delta T} = 0.376$$

$$\text{Line eq : } y = mt + a$$

$$\text{Line eq based point 2 : } 0.521 = 0.376 * 5.05 + a$$

$$a = -1.378$$

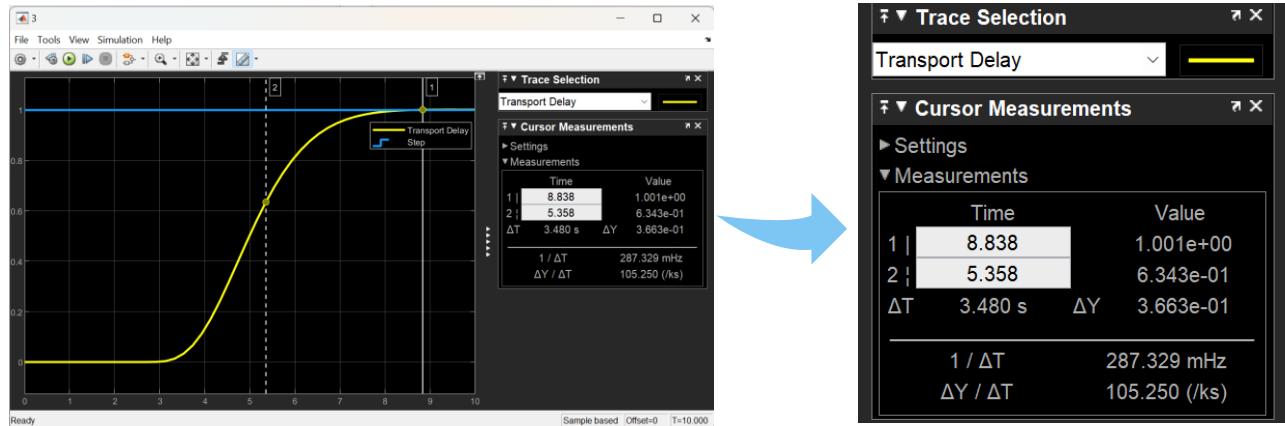
$$\underline{\text{eq : } y = 0.376*t - 1.378}$$

همانگونه که میدانیم: $T = A - B$ or $T = A - C$

پس برای محاسبه $G_{(s)} = \frac{K}{1+sT} e^{-sL}$ باید سه نقطه A B C را بیابیم.

A: when $Y = 0$ we can find A, so: $0 = 0.376*t - 1.378$ then : $A = 3.66 = L$ in eq

B: we know 0.63K is the value of plot in B and $K = 1.001$ so in scope we can find B :



So B is 5.358

C: C is the time of plot when we have value K so C is : $C = 8.814$

بر اساس انتخاب مدل AC یا AB به دو جواب میرسیم:

$$T_1 = AB = 1.698$$

$$T_2 = AC = 5.154$$

$$L = A$$

پس مدل های ما بصورت زیر بدست می آیند:

$$G_{(s)} = \frac{1.001}{1 + 1.698s} e^{-3.66s}$$

$$G_{(s)} = \frac{1.001}{1 + 5.154s} e^{-3.66s}$$

تحلیل:

این دو مدل دارای تاخیر ثابت هستند اما سیستمی که با نقطه C نوشته شده است ثابت زمانی بیشتری دارد و کنترل از B است. در شبیه سازی پاسخ خواهیم داد که مدل B تقریب بهتری است.

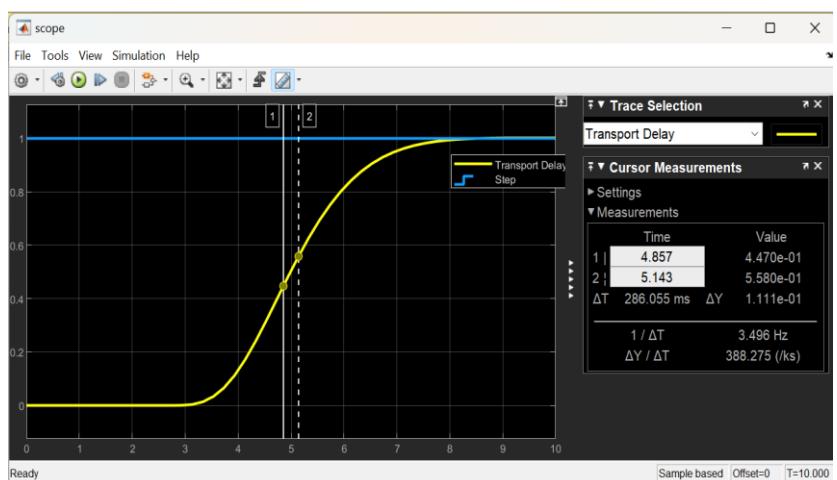
۲-۳ شناسایی مدل چهار جزئی

میدانیم که فرم کلی مدل سه جزئی بصورت زیر است:

$$G(s) = \frac{Ke^{-sL}}{(1+sT_1)(1+sT_2)}$$

همانطور که میدانیم مدل چهار جزئی با استفاده از دانش ما و محاسبات دستی قابل شناسایی نیست و نیاز به محاسبات کامپیوتری دارد. جهت انجام این محاسبات و حل معادله غیر خطی دو نقطه رندوم برای این نقاط پیدا کرده و اقدام به حل معادلات میکنیم. مقدار K و L به مانند قبل محاسبه می شود. در انتخاب آن دو نقطه بهتر است به این صورت عمل می کنیم که در دامنه های T = ۰.۳۳ و T = ۰.۶۷ مقدار زمان را لحاظ کنیم.

$$S(t) = K \left(1 + \frac{\left(T_2 e^{-\frac{(t-L)}{T_2}} - T_1 e^{-\frac{(t-L)}{T_1}} \right)}{T_1 - T_2} \right) \quad T_1 \neq T_2$$



حال ما دو نقطه را بدست آورديم پس برای حل معادله از کد زیر استفاده ميکنيم:

```
t1=4.857;
t2=5.143;
d1=0.447;
d2=0.558;
K=1.001;
L=3.66;
```

```
func = @(t_array)root(t_array,t1,t2,L,d1,d2,K);
t0 = [0.01,0.5];
t_array = fsolve(func,t0)
```

```
function F = root(t_array,t1,t2,L,d1,d2,K)
F(1) = 1+(t_array(2)*exp(-(t1-L)/t_array(2))-t_array(1)*exp(-(t1-L)/t_array(1)))/(t_array(1)-t_array(2)) - d1/K;
F(2) = 1+(t_array(2)*exp(-(t2-L)/t_array(2))-t_array(1)*exp(-(t2-L)/t_array(2)))/(t_array(1)-t_array(2)) - d2/K;
end
```

كه در نتيجه به ما جواب نهايی را ميدهد:

Equation solved.

fsolve completed because the vector of function values is near zero
as measured by the value of the **function tolerance**, and
the **problem appears regular** as measured by the gradient.

<stopping criteria details>
t_array = 1x2
0.1169 1.8192

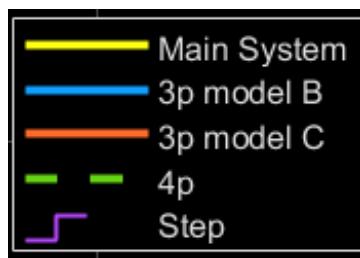
حال که تمام پارامتر های لازم را داريم در معادله جايگذاري ميکنيم و مدل چهار جزئی را بدست می آوريم:

$$G_{(s)} = \frac{1.001}{(1 + 0.1169s)(1 + 1.8192s)} e^{-3.66s}$$

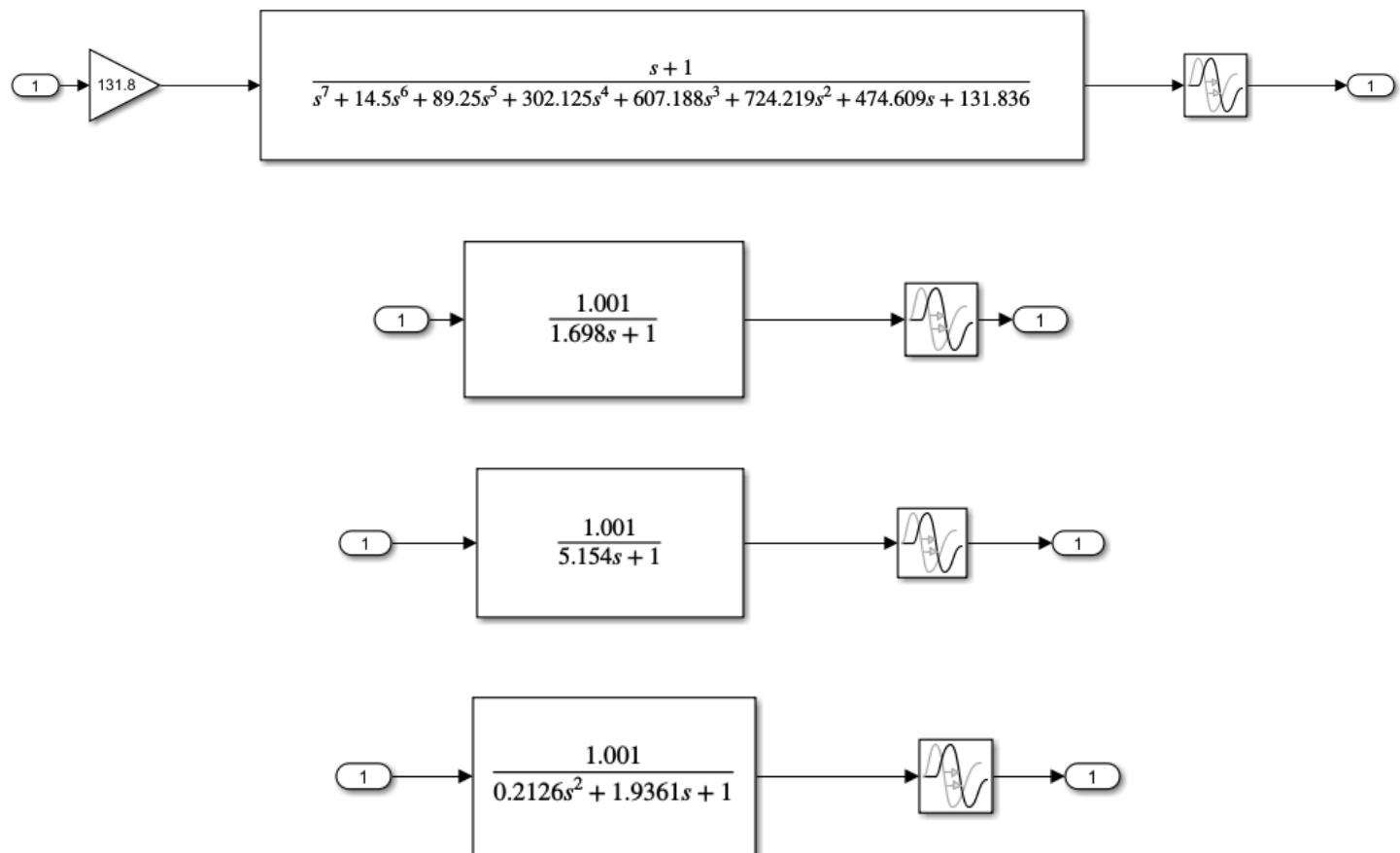
۴- مقایسه پاسخ های بدست آمده از شناسایی

برای مقایسه نهایی مدل های بدست آمده ابتدا باید آنها را کنار هم رسم کنیم. با استفاده سیمولینک مطلب

مشابه آنچه برای قسمت قبل انجام دادیم با استفاده از بلوک های پله و فانکشن و اسکوپ پاسخ ها را شبیه سازی میکنیم و با اسکوپ مشاهده میکنیم. مطابق شکل زیر تمامی سیستم ها را به صورت زیر سیستم

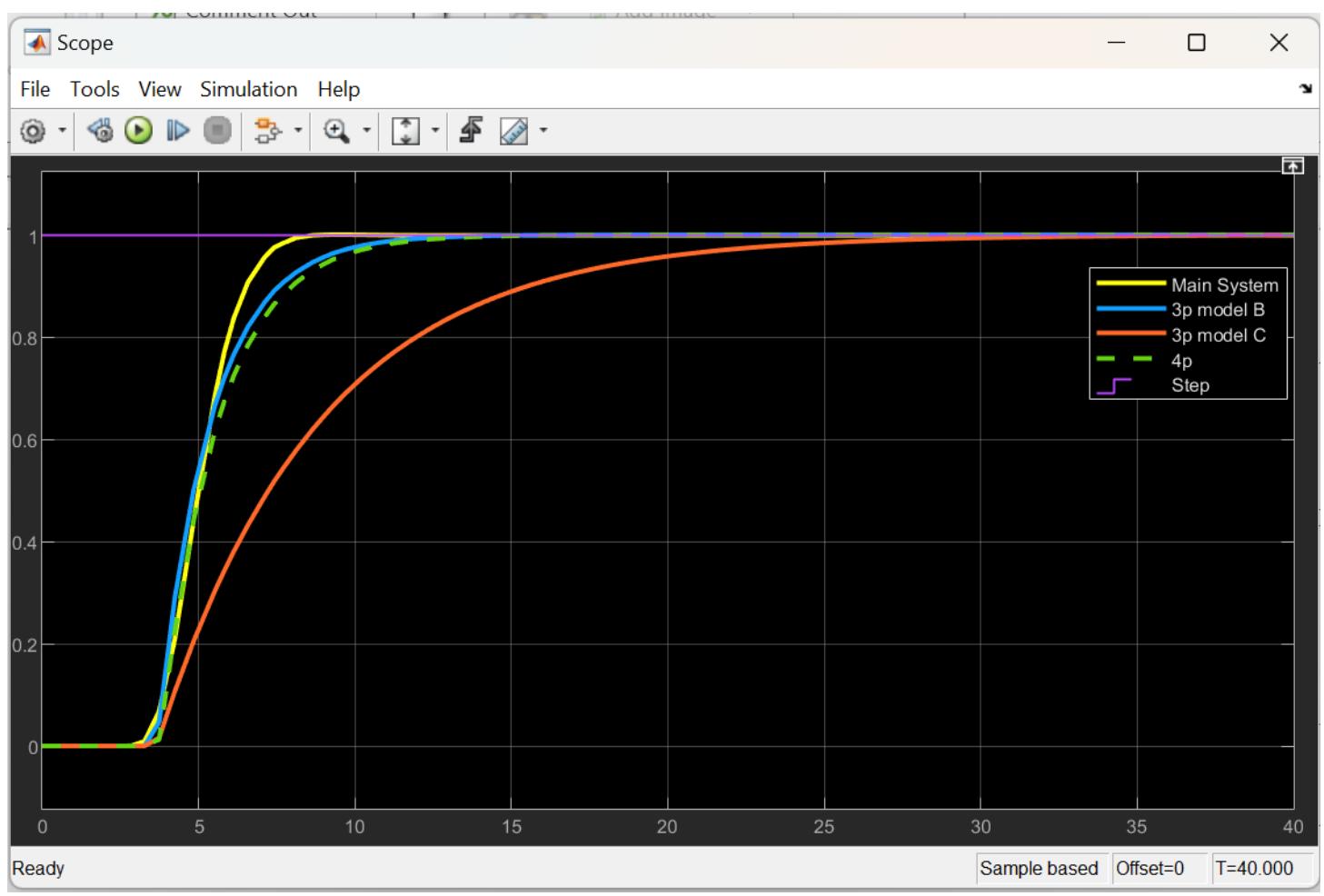
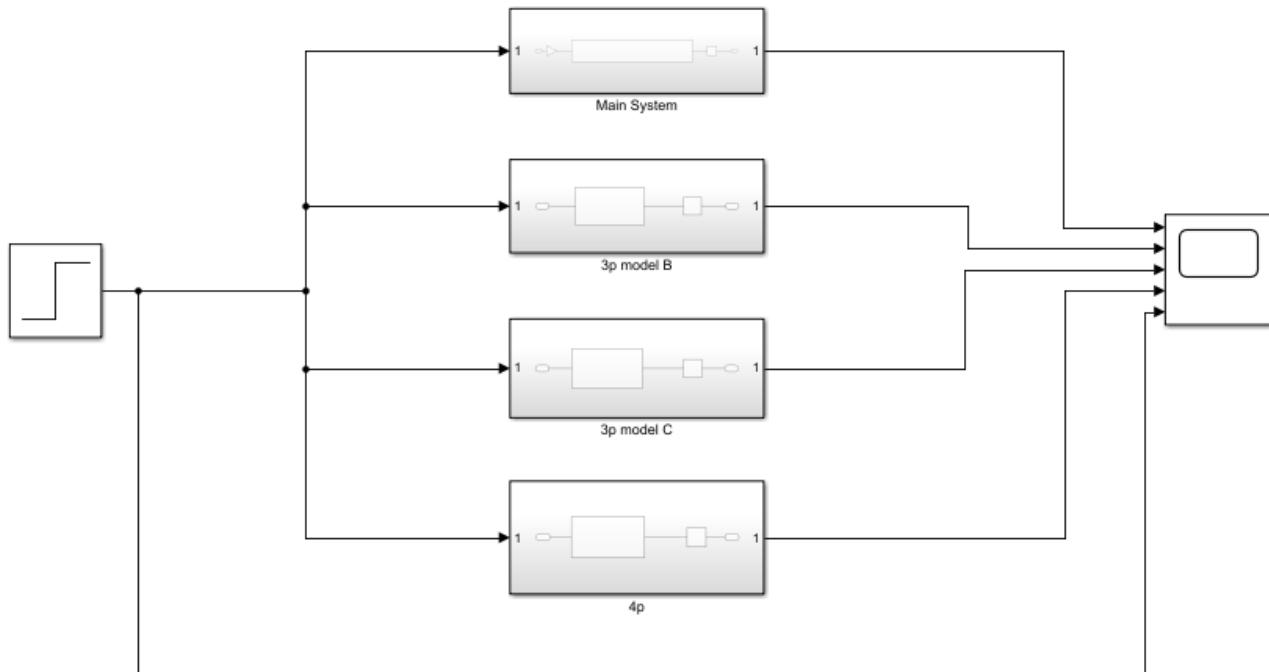


پیاده سازی کرده و خروجی آنها را به یک اسکوپ جهت مقایسه میدهیم.



- همانگونه که مشاهده میشود در مدل چهار جزئی مخرج باز شده است و عبارات در هم ضرب شده اند.

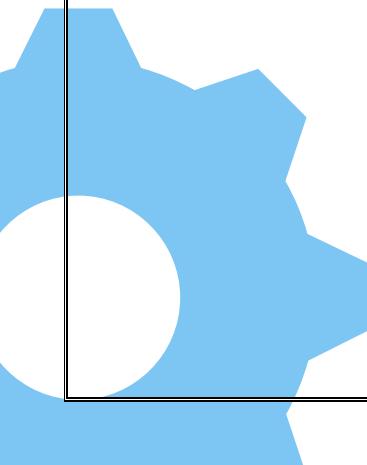
به صورت همزمان ورودی پله واحد به سیستم ها داده و پاسخ آنها را به همراه ورودی در اسکوپ مشاهده می کنیم:



مقایسه : همانطور که از نمودارهای فوق پیداست مدل چهارجزئی رفتاری شبیه تر به سیستم اصلی دارد و

دقیق تر از مدل سه جزئی می باشد . البته هیچ کدام از دو سیستم نتوانستند اورشوت سیستم را دقیق شناسایی کنند . اگر دقت شود منحنی پاسخ سیستم باتابع انتقال اصلی مقدار بسیار کمی در حد ۱۰۰۰ اورشوت دارد . اما از لحاظ نزدیک بودن پاسخ به سیستم اصلی و شبیب و سرعت رشد مدل چهارجزئی عملکرد بهتری از خود به نمایش گذاشته است . علت این امر بر آن است که برای محاسبه مدل چهارجزئی از پارامتر های بیشتری استفاده شده است و به مراتب هم محاسبه و شناسایی آن زمان بر و دشوار تر است . همچنین سیستم را با مرتبه دوم به جای اول (در شناسایی سه جزئی) تخمین زده ایم . طبیعتاً انتظار هم داشتیم که پاسخ سیستم با مدل چهارجزئی رفتار بهتری داشته باشد . همچنین می توان گفت که مدل چهارجزئی حالت ماندگار و حالت گذرای سیستم را نسبت به مدل سه جزئی با تقریب بهتر و دقیق تری مدل می کند . از بین مدل های سه جزئی ، مدلی که با انتخاب نقطه B بدست آمده دارای پاسخ سریع تری است و نتیجه بهتری داده است . پس میتوان گفت مدل سه جزئی با اندازه $63K$ دقت بهتری به نسبت مدل C با اندازه k دارد . مدل چهارجزئی با توجه به دقت بیشتر دارای نتیجه عالی در تعقیب پاسخ پله سیستم اصلی دارد . تاخیر سیستم و مقدار نهایی پاسخ ها نیز رعایت شده است .

یس در نتیجه مدل چهار جزئی < مدل سه جزئی با $63K$ > مدل سه جزئی با k

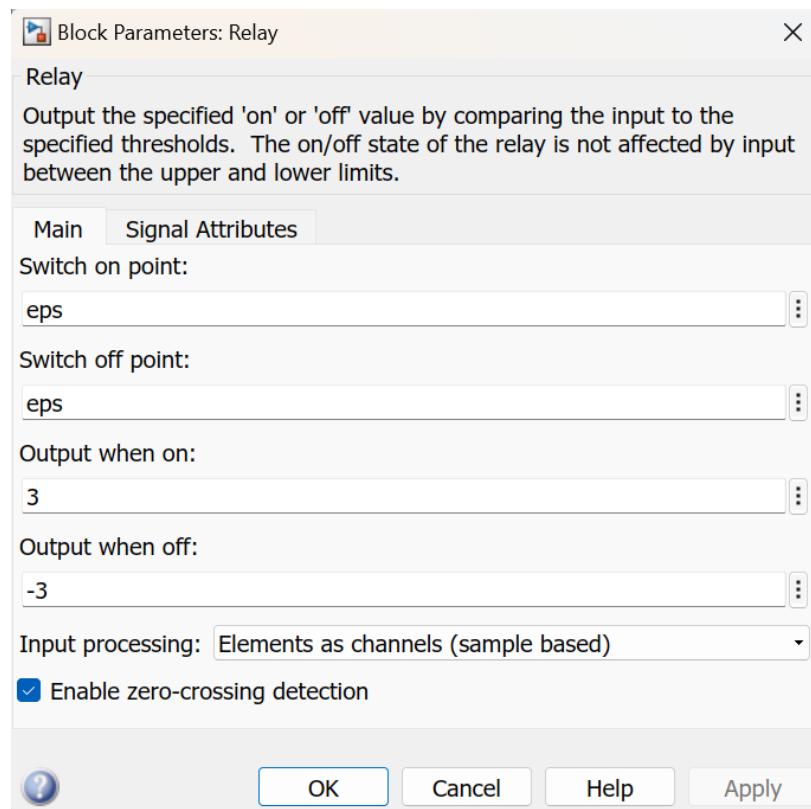
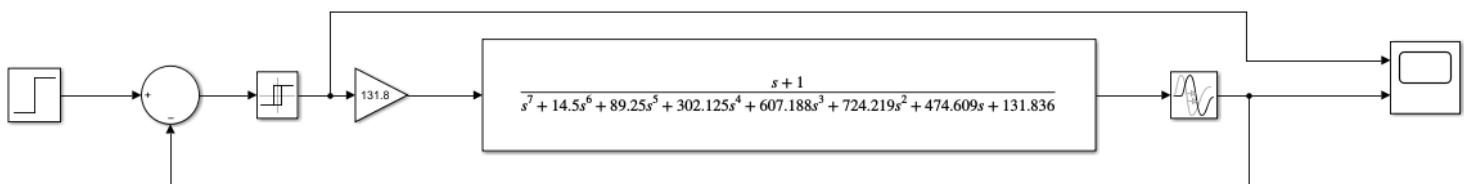


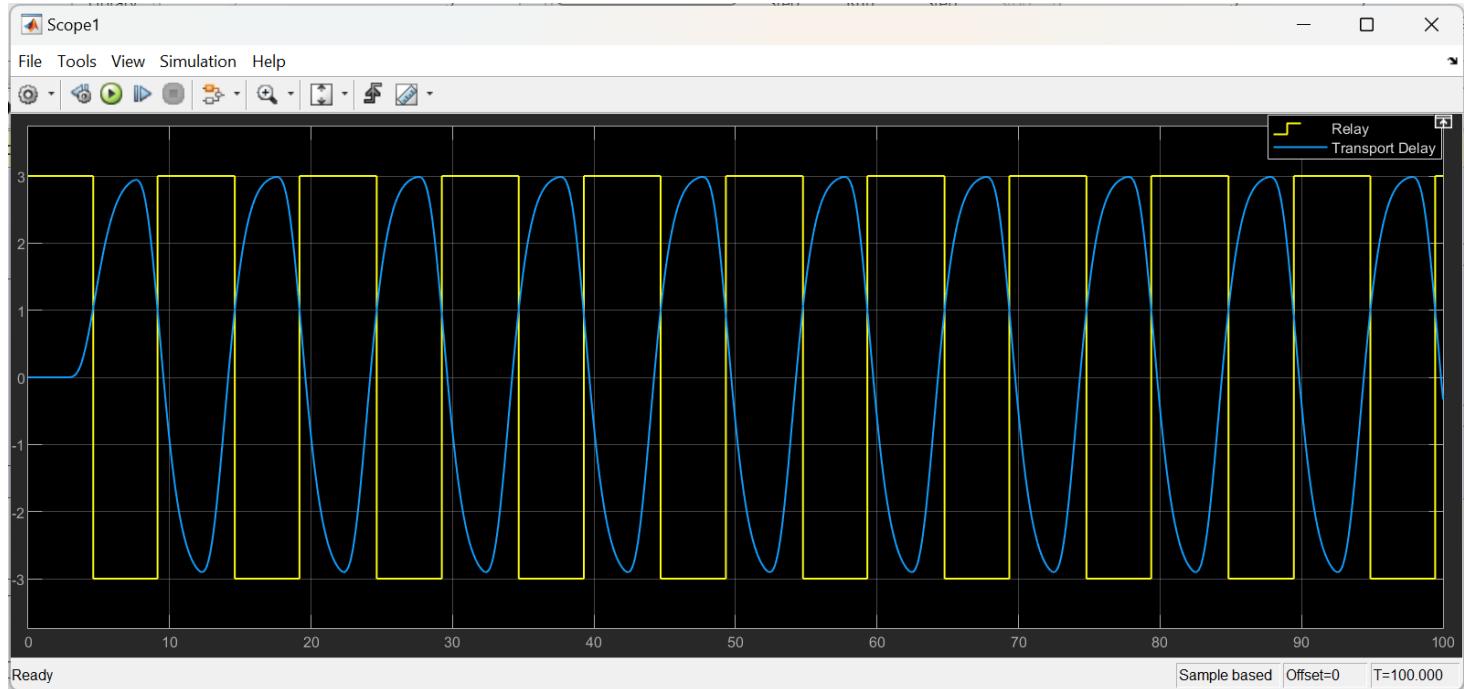
۵- اطلاعات نقطه نهایی سیستم

محاسبه نقطه نهایی از آن جهت اهمیت دارد که اطلاعات مهمی از پاسخ فرکانسی سیستم و تعیین پایداری را دراست و همچنین به کمک آن میتوان کنترل کننده PID نیز طراحی نمود. یک روش پایه برای محاسبه برای این نقطه، صفر قرار دادن بخش موهومی در پاسخ فرکانسی $G(j\omega)$ یا استفاده از دیاگرام بودی است. اما در درس کنترل صنعتی با دو روش دیگر آشنا شده ایم. روش اول بالا بردن بهره کلی تا جایی است که سیستم پاسخ نوسانی کامل بدهد که انجام این کار محدودیت دارد.

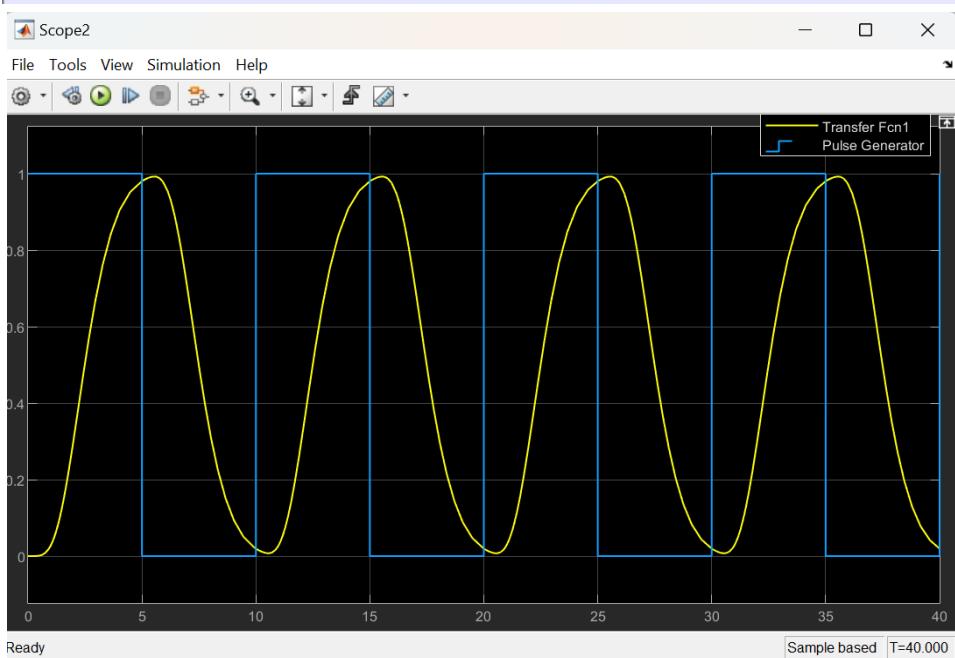
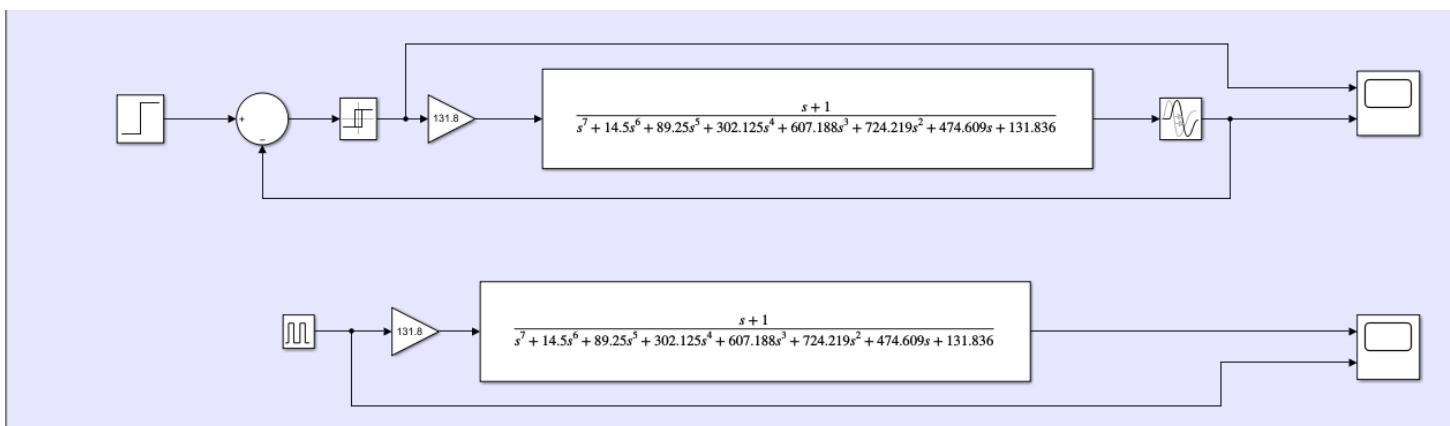
۱-۵ اعمال فیدبک رله

روش استفاده شده در این گزارش استفاده از فیدبک رله خواهد بود تا پاسخ نوسانی بدون میرا شود.





البته میتوان به جای رله با استفاده از pulse Generator نیز یک پالس سینوسی مطابق شکل زیر ساخت:



همانطور که مشاهده میشود

نتیجه رله و پالس ساز بصورت

یکسان است و حتی با پالس ساز

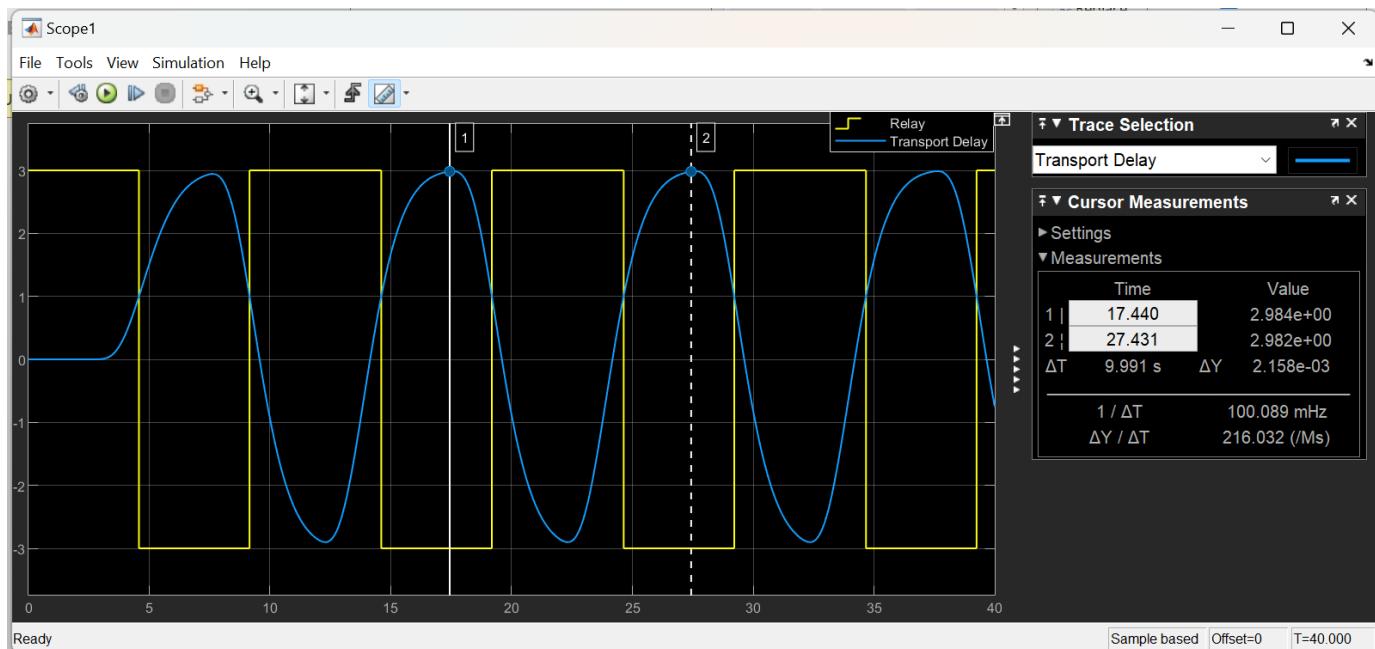
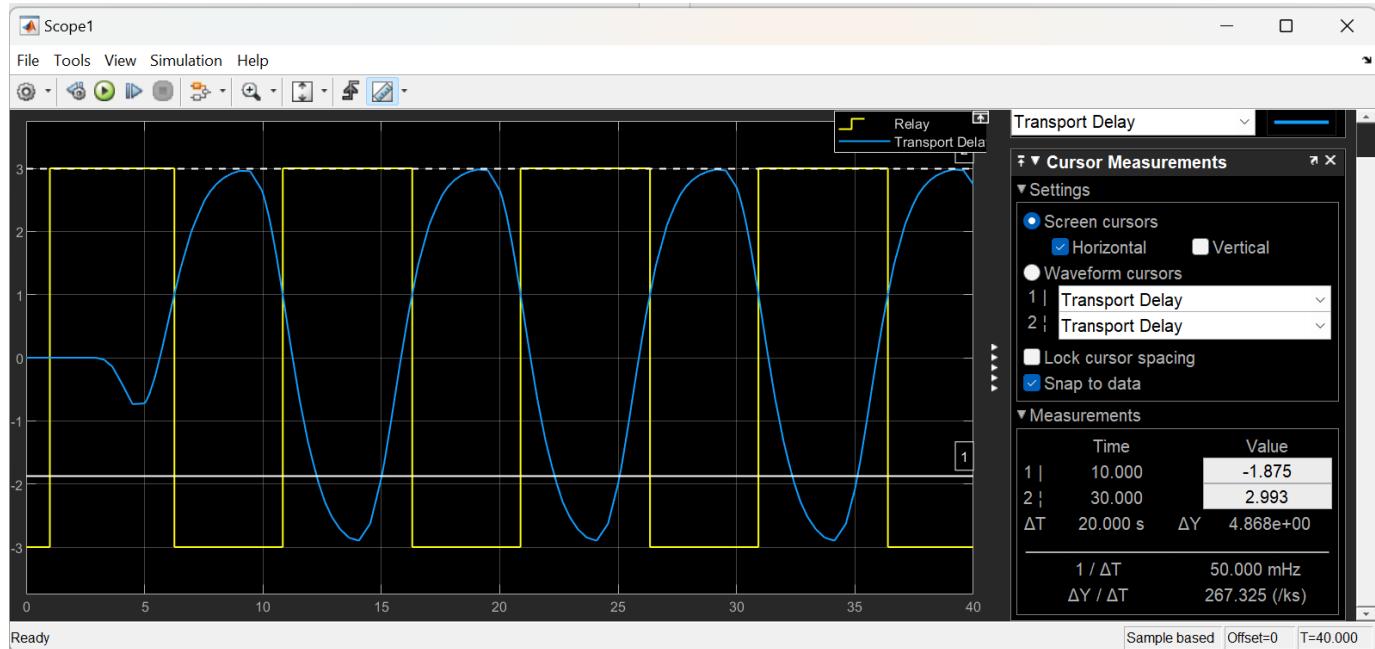
بهتر است.

۲-۵ تعیین پارامترهای نقطه نهایی

با انجام آزمایش حال نوبت به تعیین پارامترهای نقطه نهایی سیستم میباشد. با استفاده از رابطه حال از روی نمودار بایستی مقدادیر مجهول پیدا شود.

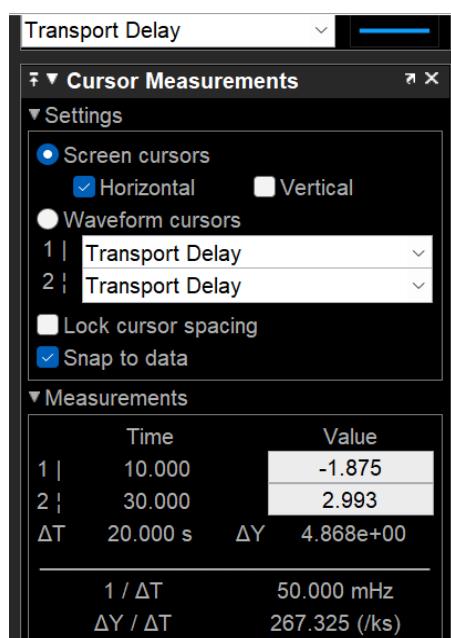
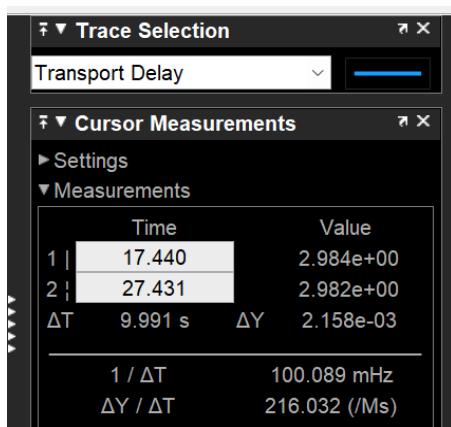
$$G(i\omega_n) = -\frac{\pi a}{4d}$$

مقدار $2a$ مربوط به پیک تا پیک پاسخ نوسانی و مقدار $2d$ برای پیک تا پیک دامنه رله است. اطلاعات مربوط به نقطه نهایی از روابط زیر بدست خواهد آمد:



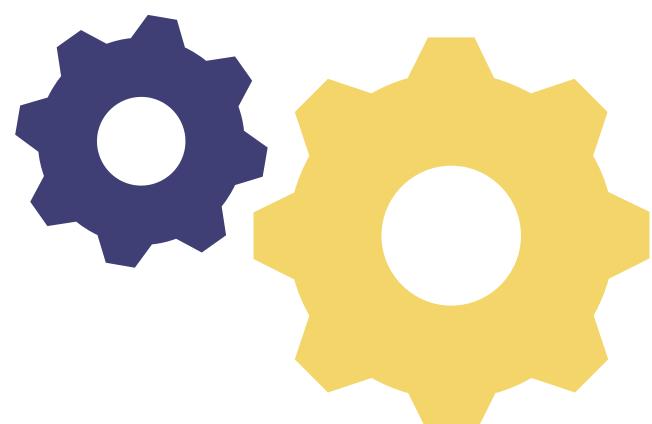
با توجه به شکل های زیر دوره تناوب برابر با ۹.۹۹۱ ثانیه است، همچنین دامنه نوسانات برابر ۲.۹۹۳ و دامنه رله مقدار ۳ که در قسمت تنظیمات آن تنظیم کردیم بدست می‌آیند. حال با داشتن این مقادیر و جایگذاری آن در فرمول میتوان

نوشت:



$$G_{(iwu)} = -\frac{\pi \times 2.993}{4 \times 3} = -0.784$$

$$Ku \cdot G_{(iwu)} = -1 \quad \dots \quad Ku = \frac{1}{0.784} = 1.275$$



۶- طراحی PID

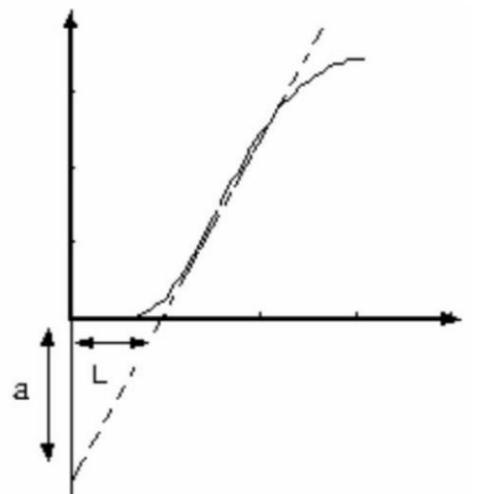
۱-۶ طراحی به روش زیگلر - نیکولز (حوزه زمان)

در این بخش ما میخواهیم به طراحی کنترل کننده به روش زیگلر - نیکولز در حوزه زمان بپردازیم. باید توجه داشت که برای یافتن شیب حداکثری و رسم آن پیش تر پارامتر های مورد نیاز برای این قسمت پیدا شده و در این بخش صرفا به یادآوری آن ها میپردازیم.

مطابق شکل رو به رو ما برای ساخت کنترل کننده PID

به دو فاکتور L - a نیاز داریم که این دو فاکتور در بخش

۳-۱ بدست آمده است:



$$L = 3.66 \quad , \quad a = 1.378$$

حال با استفاده از پارامترهای بدست آمده، از جدول زیر استفاده می کنیم تا ضرائب کنترل کننده PID بدست بیاید.

T_p	T_d	T_i	K	کنترل کننده
$4L$	0	0	$1/a$	P
$5.7L$	0	$3L$	$0.9/a$	PI
$3.4L$	$L/2$	$2L$	$1.2/a$	PID

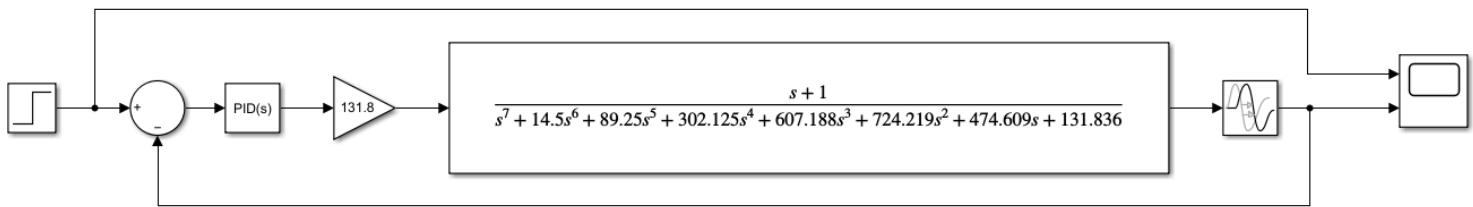
با انجام محاسبات بصورت دستی پارامترهای ما به شکل زیر به دست می آید:

$$K = 0.781 \quad , \quad T_i = 7.32 \quad , \quad T_d = 1.83 \quad , \quad T_p = 12.44$$

حال مطابق آنچه از کنترل کننده PID میدانیم میتوانیمتابع انتقال کنترل کننده را بنویسیم:

$$C(S) = K \left(1 + \frac{1}{T_i s} + T_d s\right) \rightarrow C(s) = 0.781 \left(1 + \frac{1}{7.32s} + 1.83\right)$$

حال در محیط سیمولینک شروع به شبیه سازی میکنیم:



حال به سراغ تنظیمات بلوک PID میرویم. در تنظیمات مربوط به بلوک آن را بصورت Ideal تنظیم میکنیم.

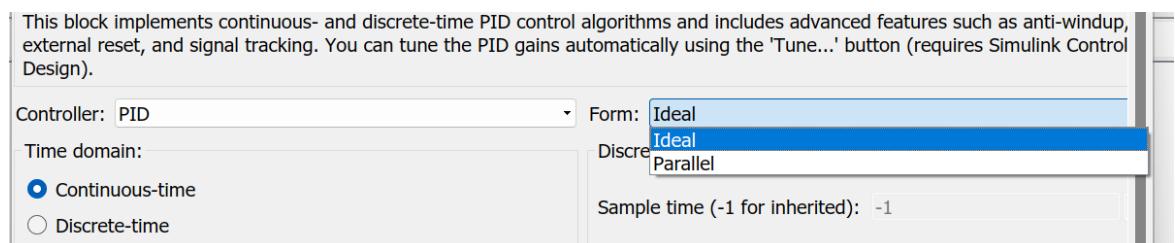
با توجه به فرمول مربوط به کنترلر PID که در بخش تنظیمات این بلوک نوشته شده است

متغیرهای $K - P - I - D$ مینویسیم.

$$K = P = 0.781$$

$$I = 1/T_i = 0.137$$

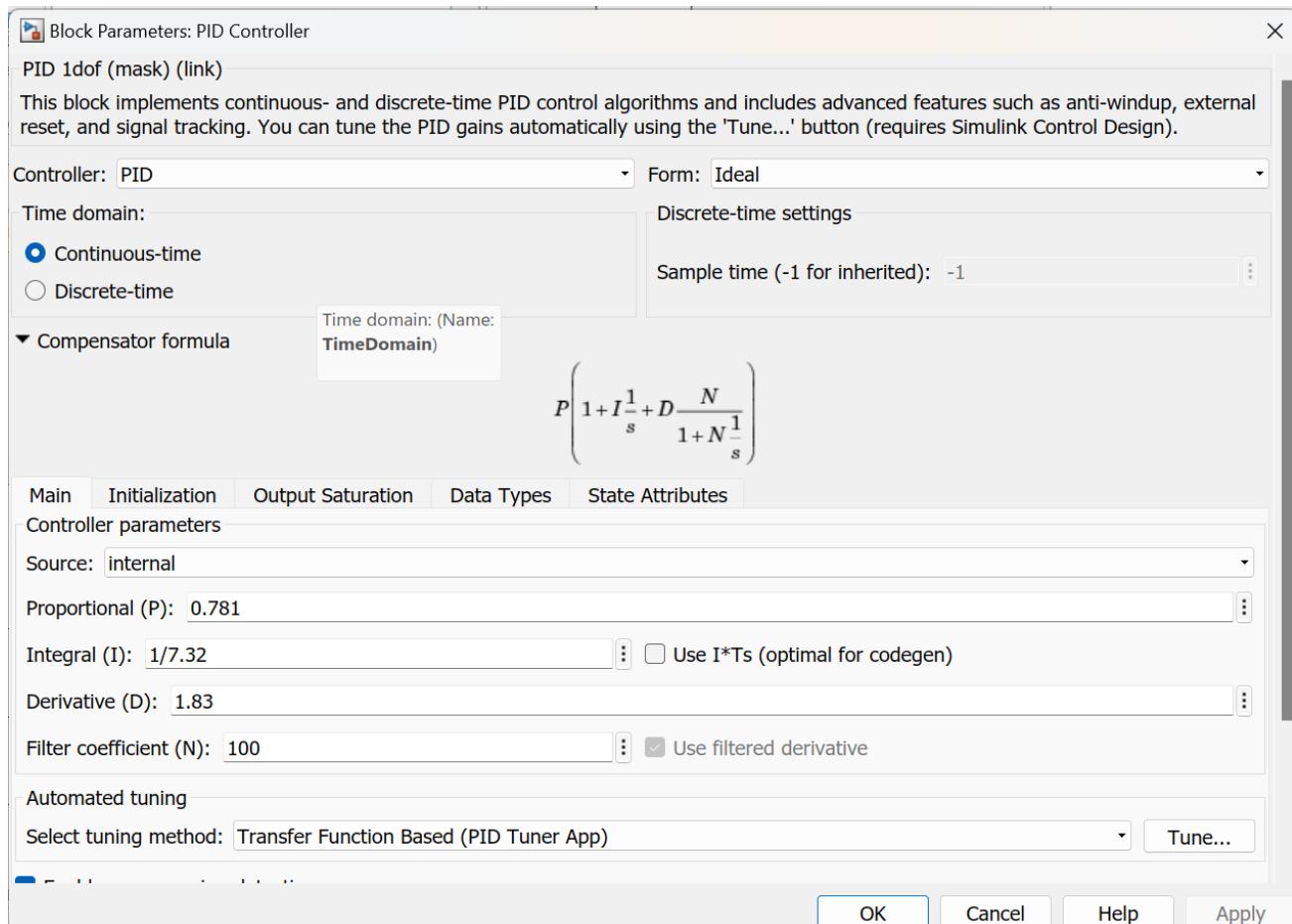
$$D = T_d = 1.83$$



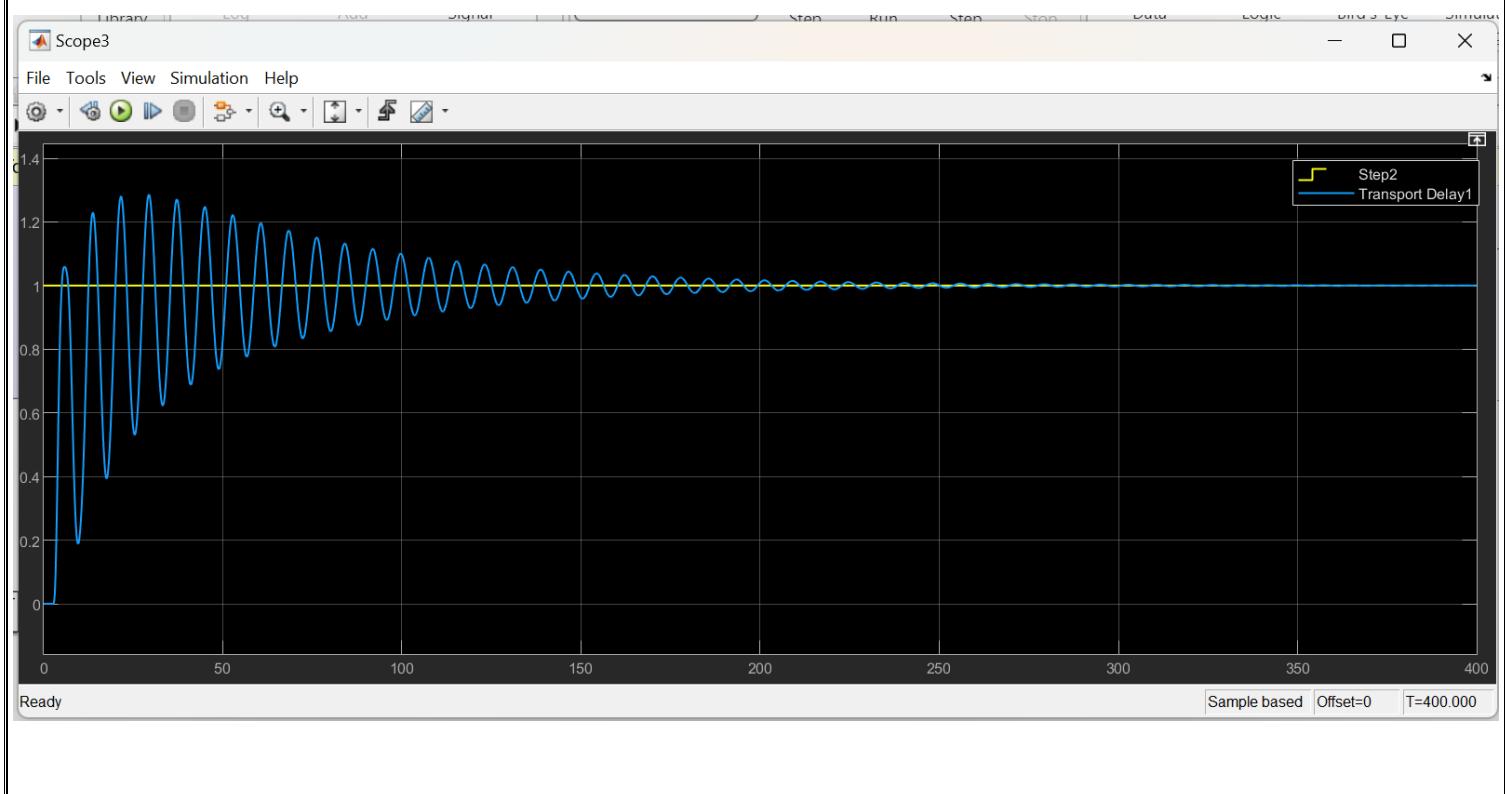
▼ Compensator formula

$$P \left(1 + I \frac{1}{s} + D \frac{N}{1 + N \frac{1}{s}} \right)$$

فرم کلی تنظیمات به شکل زیر است:

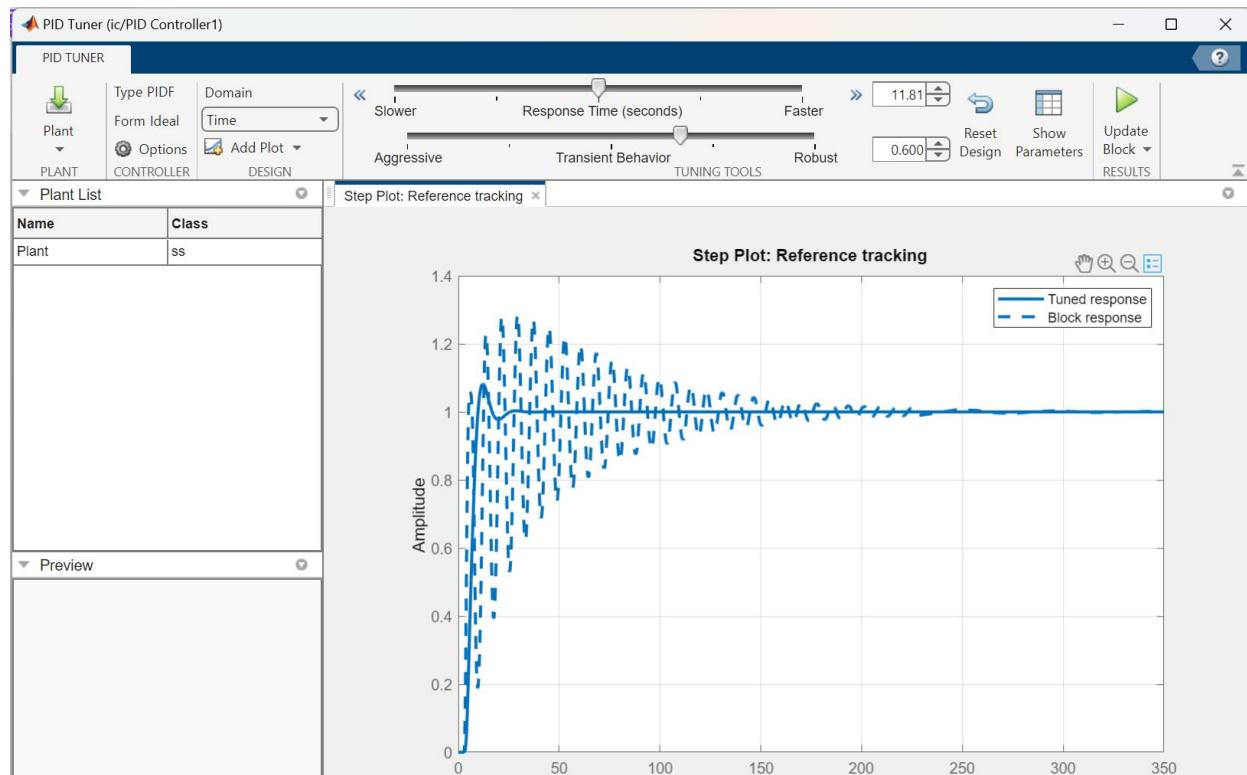
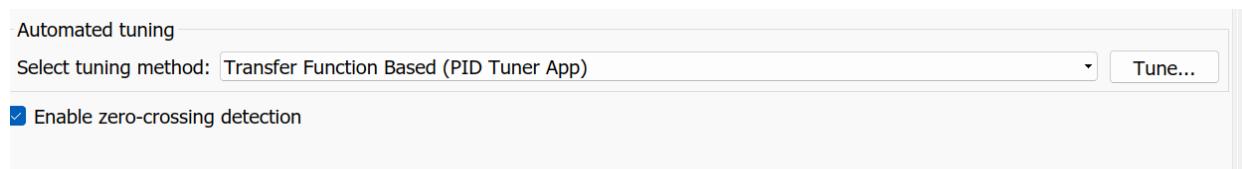


نتیجه شبیه سازی به شکل زیر است:

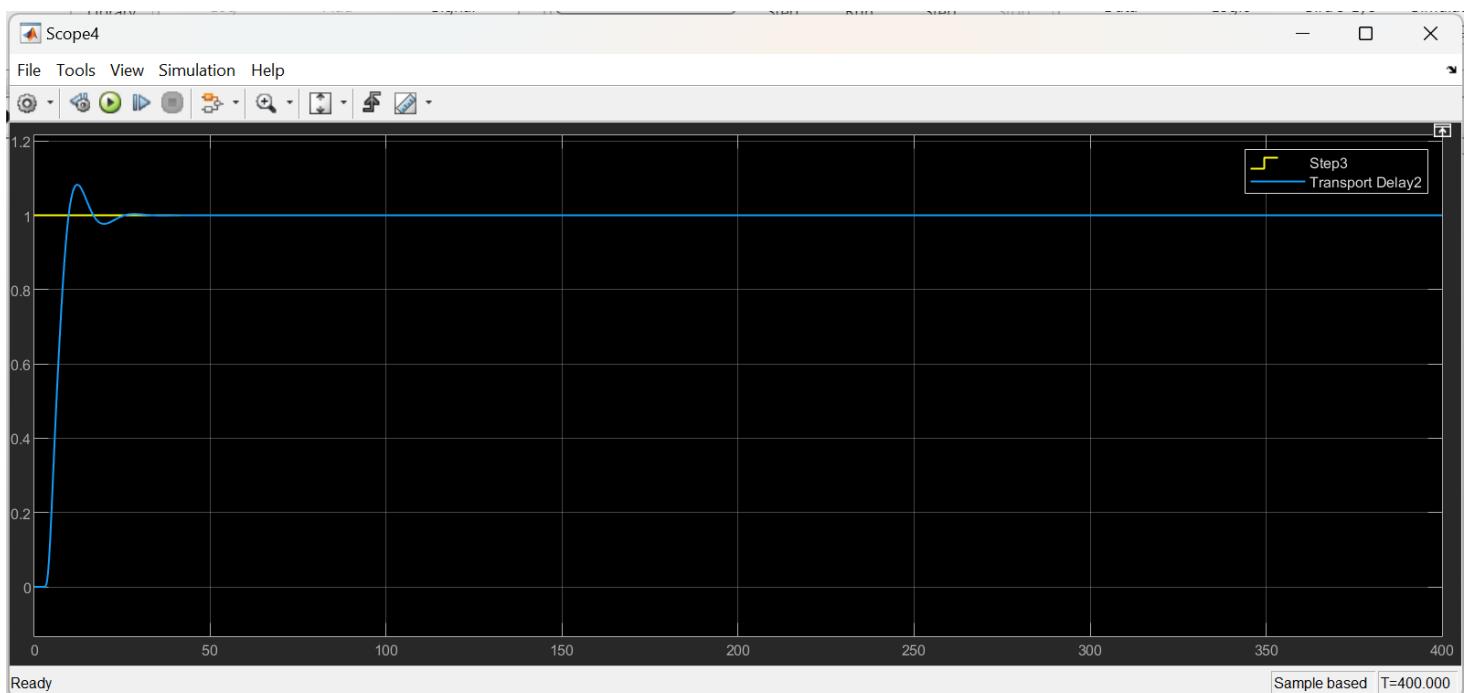
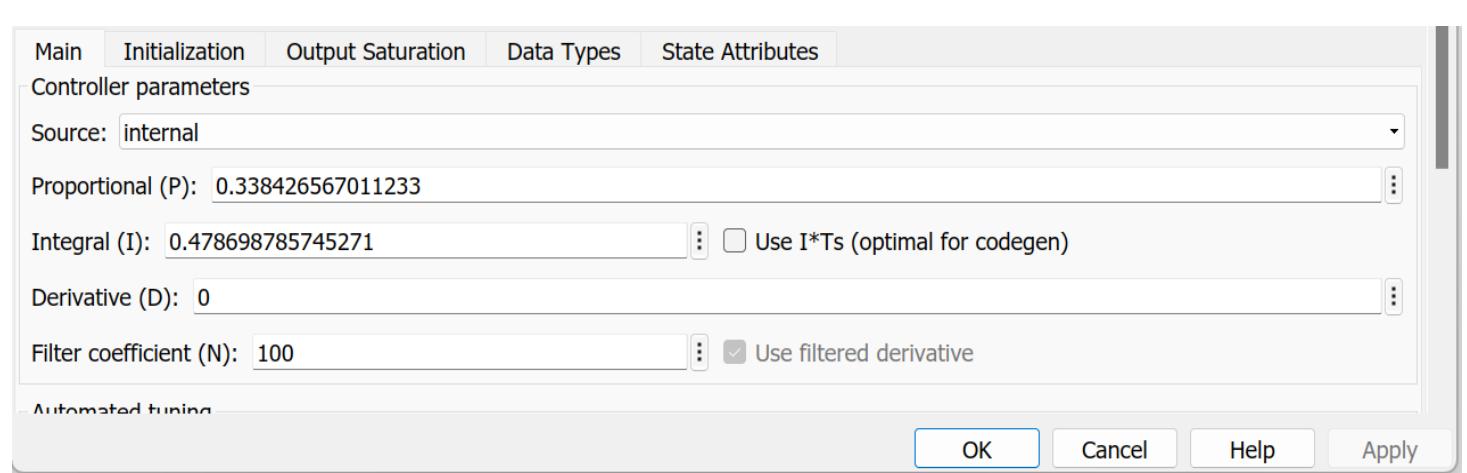


همانطور که مشاهده میشود این پاسخ مناسب نیست زیرا زمان نشت آن بسیار طولانی است و در حدود ۳۰۰ ثانیه است پس این کنترل کننده مناسب نخواهد بود. در ادامه سوال این بخش از ما خواسته شده جهت مقایسه با استفاده از tuner نرم افزار طراحی کنترل کننده را به روش آزمون و خطأ انجام دهیم. برای انجام ابتدا با ابزاری که نرم افزار در اختیار ما قرار داده کنترل کننده مناسبی طراحی میکنیم پس از آن با انجام تحلیل روی پاسخ قبلی سعی میکنیم ضرایب را به گونه ای تغییر دهیم که پاسخ مناسب حاصل گردد.

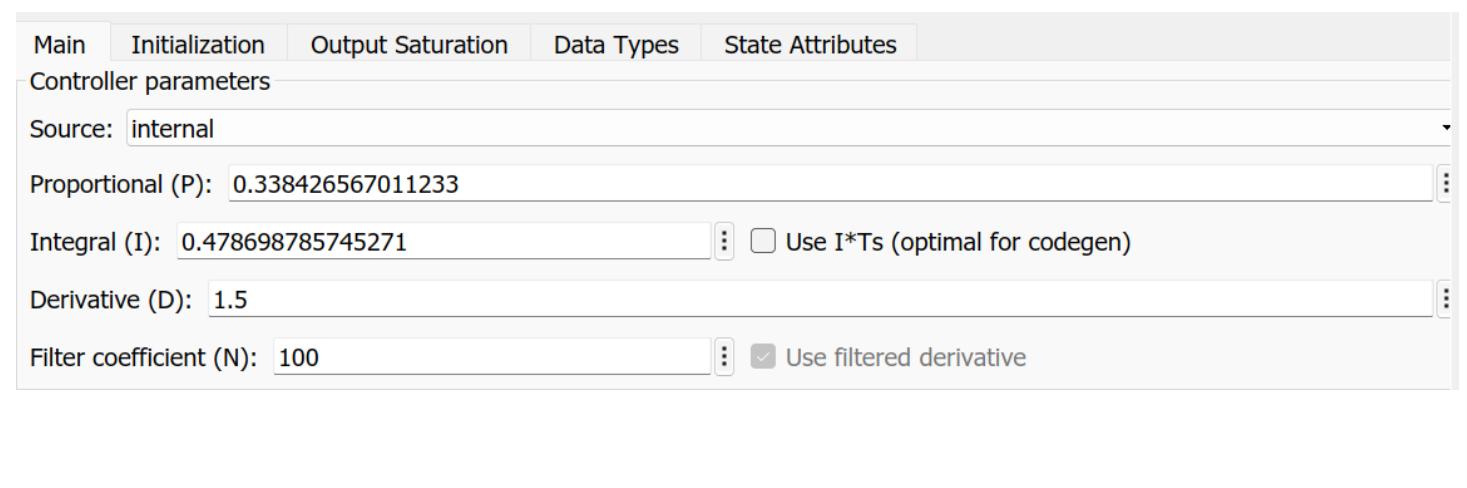
پس ابتدا ما در بخش تنظیمات کنترل کننده قسمت tune را انتخاب میکنیم.

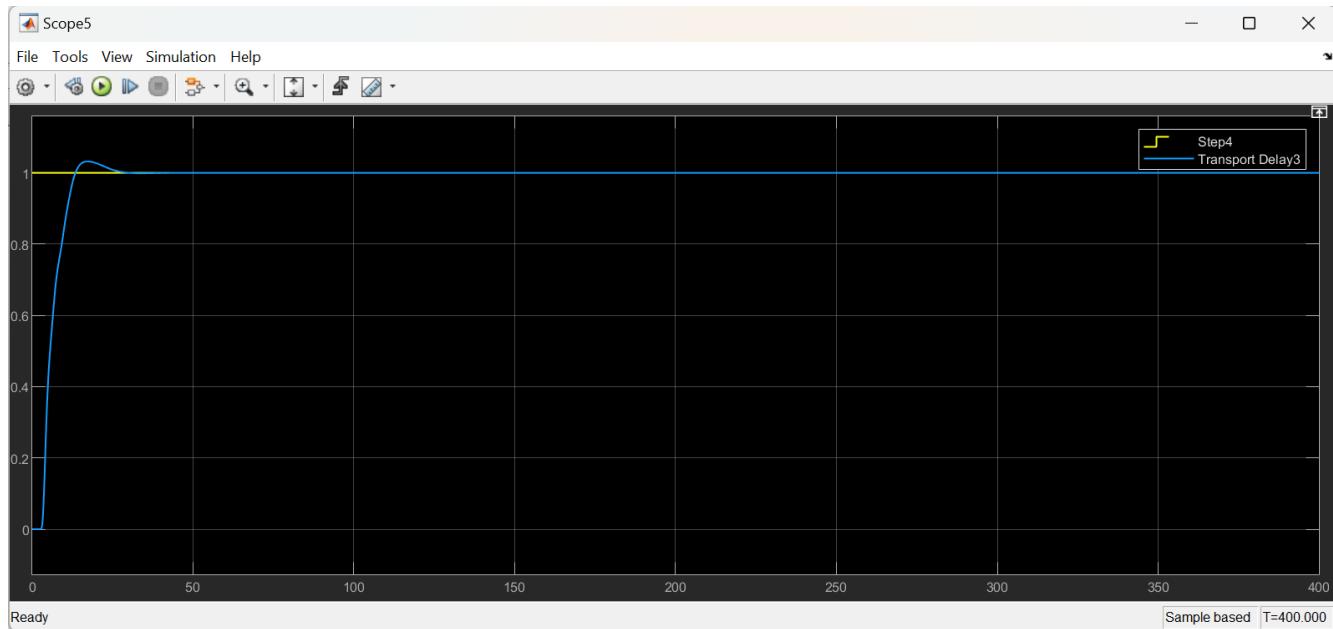


از قسمت بالا گزینه Update Block را انتخاب میکنیم تا ضرایب ما را تنظیم کند و کنترل کننده ای بهتری داشته باشیم.



حال این نمودار تیون شده را با نمودار قبلی با ضرایب مقایسه میکنیم و همانطور که مشخص است نمودار بعد از تیون شده زمان نشست آن بسیار کمتر شده و در حدود ۲۵ ثانیه شده است. اما با تغییر در ضرایب D کنترلر میتوان همین نمودار را هم بهتر و مناسب کرد.

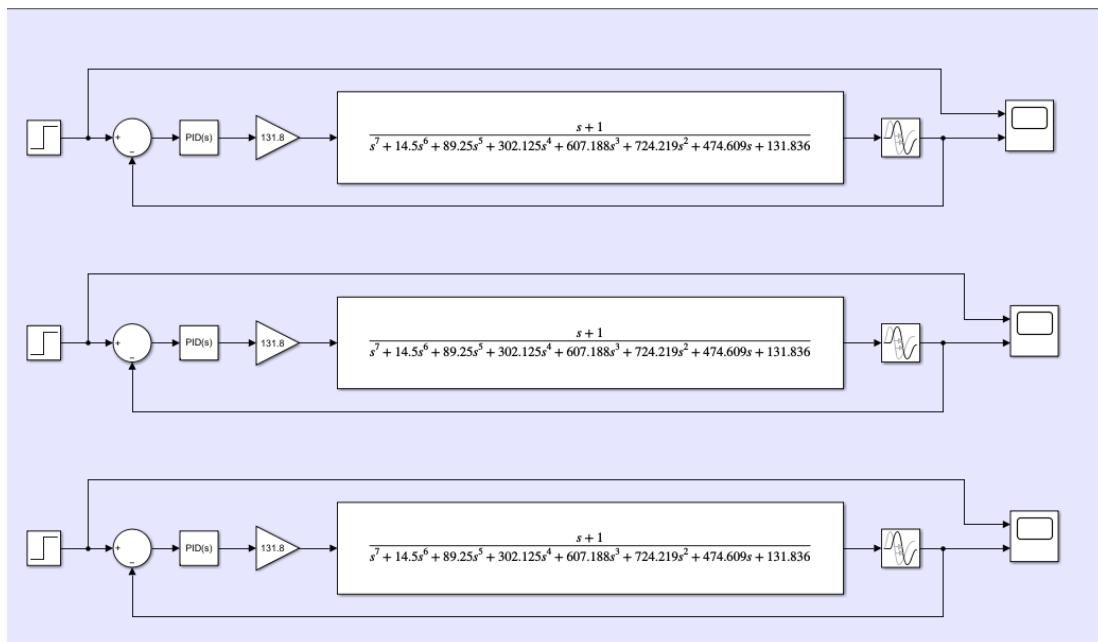




تحلیل:

سعی کردیم با تحلیل خروجی قبلی مقادیر زیگر - نیکولز سیستم را بهبود ببخشیم. همانطور که پیداست سیستم حاصل دچار نوسانات بود. این امر ناشی از ضریب مشتق گیر سیستم بود که آن را در بلوك تیون شده با صفر کردن ضریب آن از مدار خارج کرده بود.

در فرآیند طراحی به صورت دستی، با اندک افزایش ضریب مشتق گیر که این ضریب اورشوت و اندرشوت را ملایم میکرد توانستیم به یک کنترل کننده مناسب با زمان نشست مناسب دست پیدا کنیم.

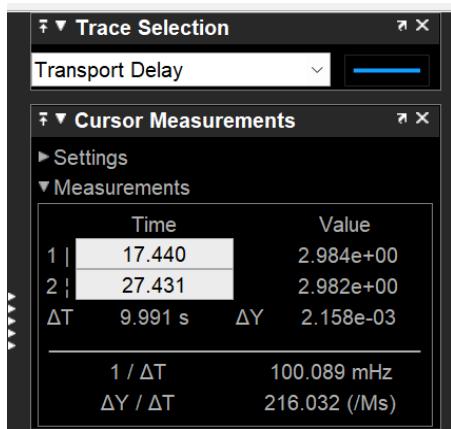


۲-۶ طراحی به روش زیگلر - نیکولز (حوزه فرکانس)

در این بخش نیز ما میخواهیم به طراحی کنترل کننده به روش زیگلر - نیکولز در حوزه فرکانس بپردازیم.

باید توجه داشت که برای یافتن T_u و K_u پیش تر پارامتر های مورد نیاز برای این قسمت پیدا شده و در

این بخش صرفا به یادآوری آن ها میپردازیم. پس مجدداً پارامترهای بخش ۴ را در اینجا می آوریم:



$$T_u = \frac{2\pi}{\omega_u} \quad K_u \cdot G(\omega_u) = -1$$

$$K_u = 1.275$$

$$T_u = 9.991$$

حال مطابق جدول زیر و همانند قسمت قبل ضرایب را بدست می آوریم.

T_p	T_d	T_i	K	کنترل کننده
T_u			$0.5K_u$	P
$1.4T_u$		$0.8T_u$	$0.4K_u$	PI
$0.85T_u$	$0.125T_u$	$0.5T_u$	$0.6K_u$	PID

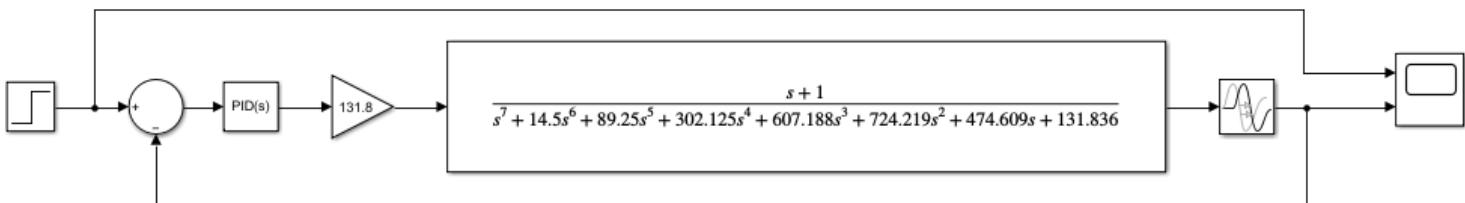
با انجام محاسبات بصورت دستی پارامترهای ما به شکل زیر به دست می آید:

$$K = 0.765 , \quad T_i = 4.996 , \quad T_d = 1.249 , \quad T_p = 8.492$$

حال مطابق آنچه از کنترل کننده PID میدانیم میتوانیمتابع انتقال کنترل کننده را بنویسیم:

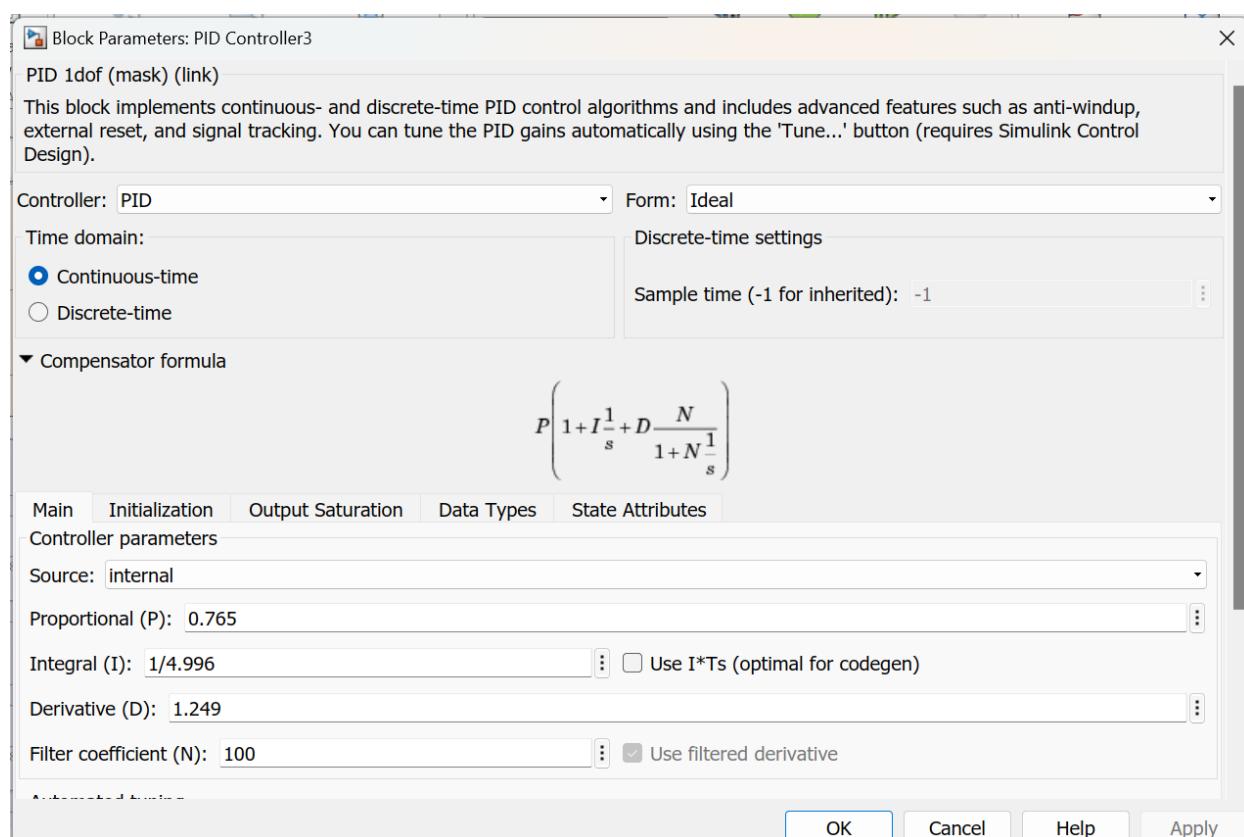
$$C(s) = k \left(1 + \frac{1}{T_i s} + T_d s \right) \rightarrow C(s) = 0.765 \left(1 + \frac{1}{4.996 s} + 1.249 \right)$$

حال در محیط سیمولینک شروع به شبیه سازی میکنیم:

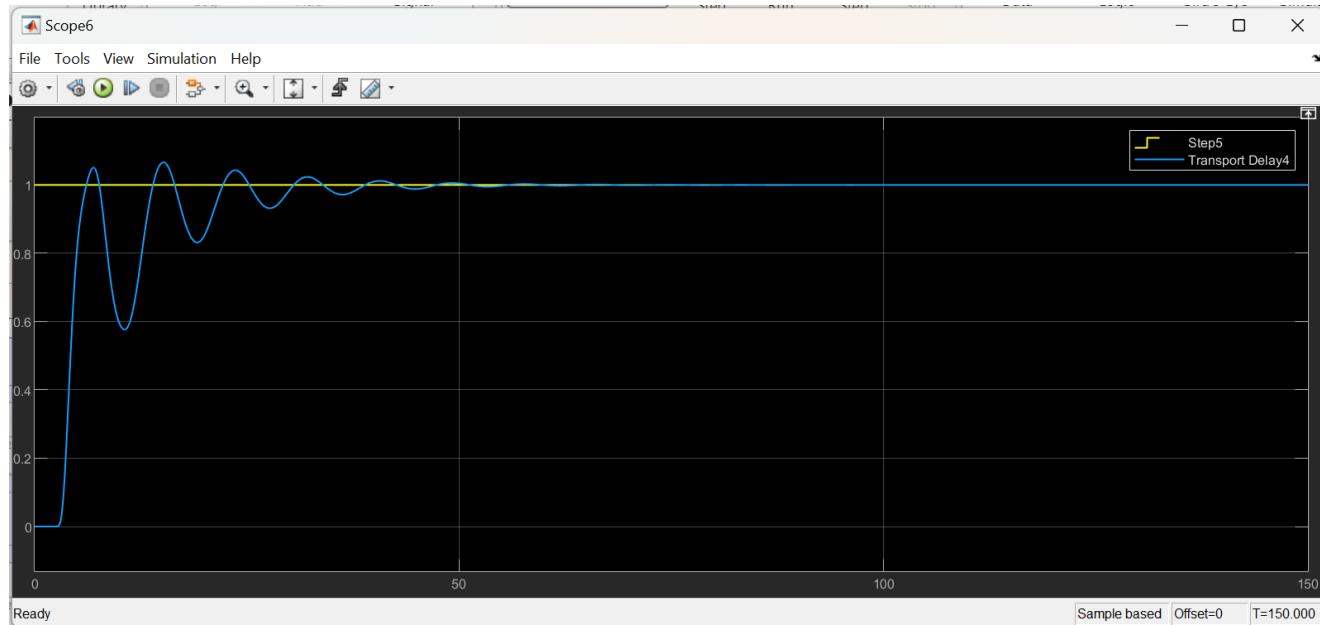


وارد تنظیمات بلوک PID شده و مطابق آنچه برای قسمت قبل انجام داده برای این قسمت نیز انجام میدهیم.

در نهایت بخش تنظیمات بلوک ما به شکل زیر خواهد شد.

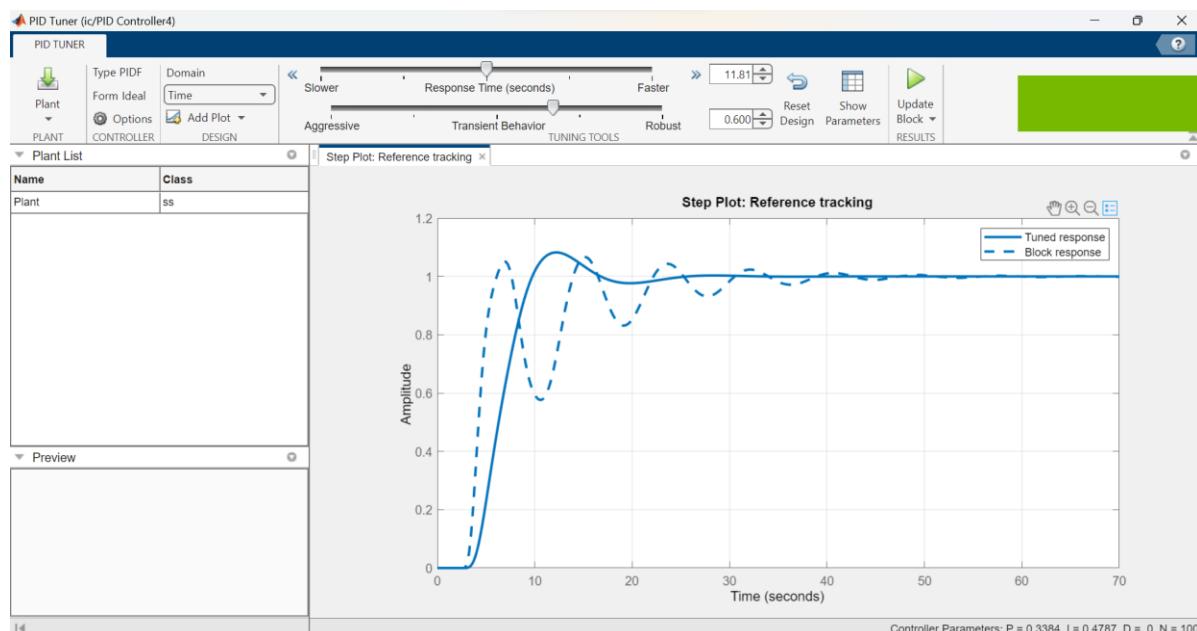


نتیجه شبیه سازی به شکل زیر است:

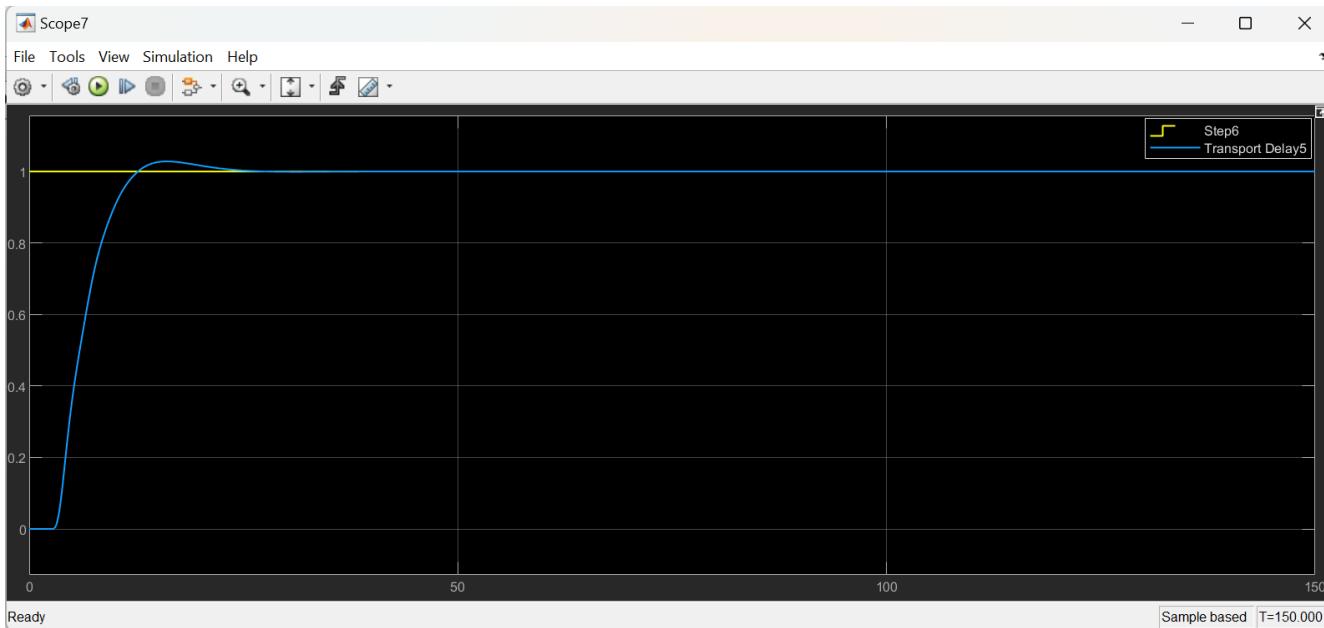


همانطور که مشاهده میشود این پاسخ به نسبت قسمت قبل مناسب تر است زیرا زمان نشست آن کوتاه تر شده است و در حدود ۶۵ ثانیه است اما همچنان توانایی بهتر شدن را دارد. در ادامه سوال این بخش همانند بخش قبل از ما خواسته شده جهت مقایسه با استفاده از tuner نرم افزار طراحی کنترل کننده را به روش آزمون و خطأ انجام دهیم. برای انجام ابتدا با ابزاری که نرم افزار در اختیار ما قرار داده کنترل کننده مناسبی طراحی میکنیم پس از آن با انجام تحلیل روی پاسخ قبلی سعی میکنیم ضرایب را به گونه ای تغییر دهیم که پاسخ مناسب حاصل گردد.

پس ابتدا ما در بخش تنظیمات کنترل کننده قسمت tune را انتخاب میکنیم.

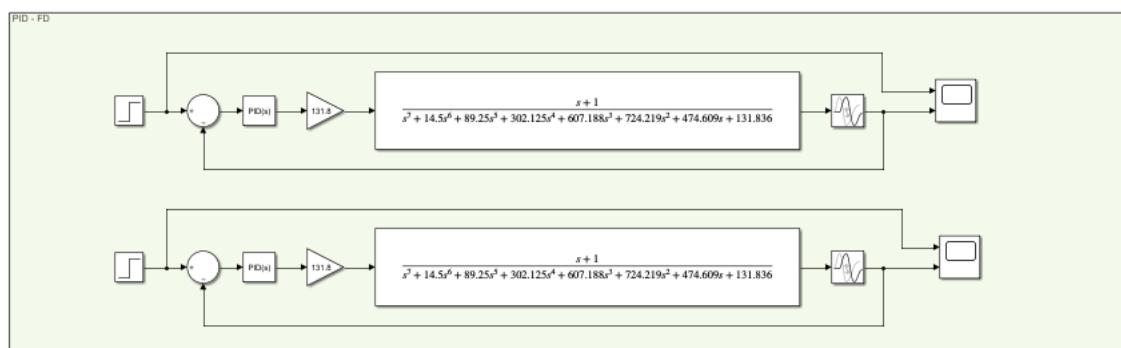


Main	Initialization	Output Saturation	Data Types	State Attributes
Controller parameters				
Source: internal				
Proportional (P): 0.338426567011233				
Integral (I): 0.478698785745271 <input type="checkbox"/> Use I*T _s (optimal for codegen)				
Derivative (D): 1				
Filter coefficient (N): 100 <input type="checkbox"/> Use filtered derivative				



تحلیل:

در این بخش نیز شبیه قبل سعی کردیم با تحلیل خروجی قبلی مقادیر زیگر - نیکولز سیستم را بهبود بخشیم. همانطور که پیداست سیستم حاصل دچار نوسانات بود اما نوسانات آن از قسمت قبل کمتر بود. سپس ما با تیون کردن و بدست آوردن ضرایب جدید و همچنین تغییر دستی ضریب مشتق گیر بعد از تیون کردن به یک کنترل کننده مناسب با زمان نشست مناسب رسیدیم.



۳-۶ طراحی به روش زیگلر- نیکولز (تعمیم یافته)

در این بخش نیز ما میخواهیم به طراحی کنترل کننده به روش زیگلر- نیکولز تعمیم یافته بپردازیم. باید توجه داشت که برای یافتن ضرایب موردنظر ما مطابق فرمول های زیر که در کلاس تئوری درس به آنها اشاره شد و آموختیم عمل میکنیم و آنها را به دست می آوریم.

با رجوع به مطالب کلاس اگر در انتقال نقطه نهایی، نقطه ابتدایی را نقطه نهایی اصلی با اندازه $1/K_u$ و زاویه

۱۸۰ در نظر بگیریم، نقطه دوم b خواهد بود. اطلاعات ضرائب کنترل کننده به مانند زیر محاسبه می شود.

$$PI : \begin{cases} K = K_u r_b \cos \varphi_b \\ T_i = \frac{-T_u}{2\pi \operatorname{tg}(\varphi_b)} \end{cases} \quad PID : \begin{cases} K = K_u r_b \cos \varphi_b \\ Ti = \frac{T_u}{\pi} \left(\frac{1 + \sin \varphi_b}{\cos \varphi_b} \right) \\ T_d = \frac{\alpha T_u}{\pi} \left(1 + \frac{\sin \varphi_b}{\cos \varphi_b} \right) \end{cases}$$

طراح کنترل کننده نقاط مربوط به b را پیشنهاد می دهد. مقادیر زیر در این مساله مورد استفاده قرار گرفته است.

$$\vartheta_b = 61 \quad r_b = 0.41$$

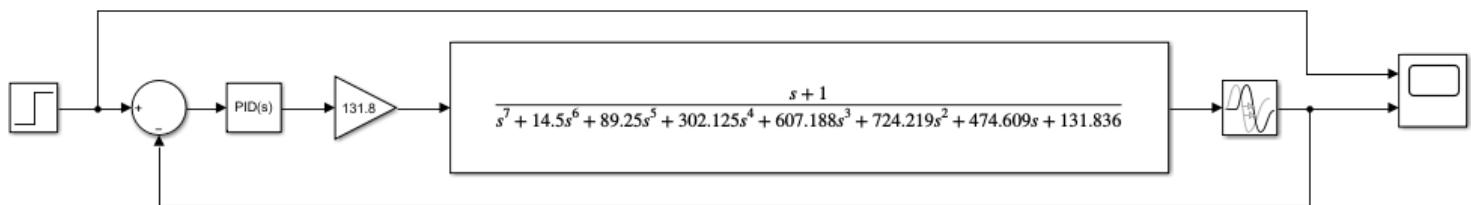
مقادیر مربوط به اطلاعات نقطه نهایی نیز از قسمت های قبل بدست آورده بودیم:

$$K_u = 1.275 \quad T_u = 9.991$$

پس ضرائب با یک محاسبه ساده بدست خواهد آمد:

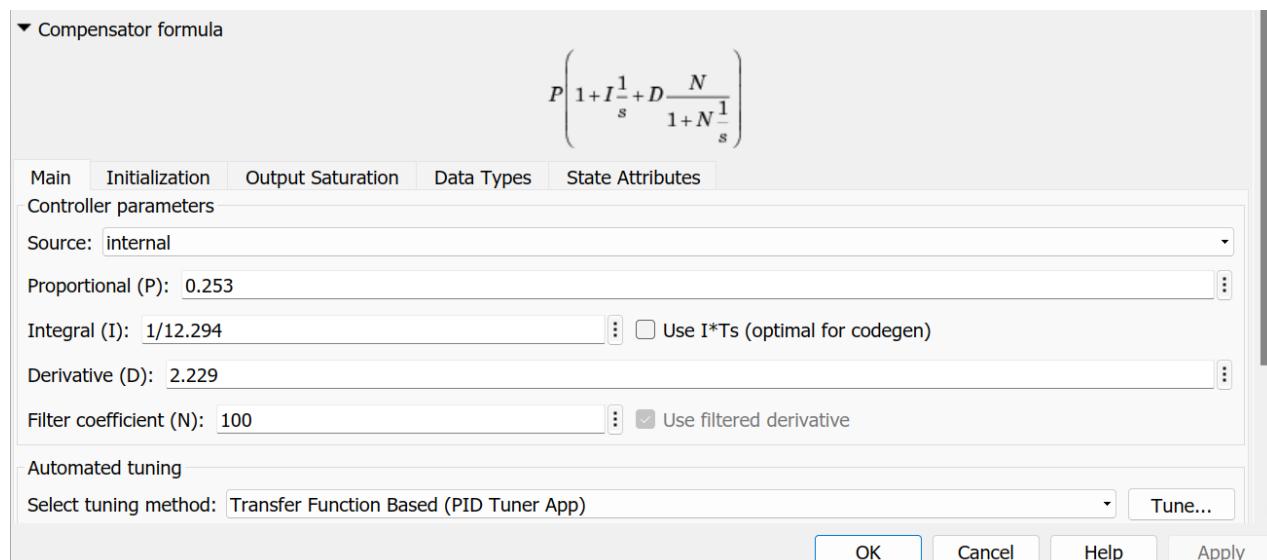
$$K = 0.253 , \quad T_i = 12.294 , \quad T_d = 2.229$$

حال در محیط سیمولینک شروع به شبیه سازی می کنیم:

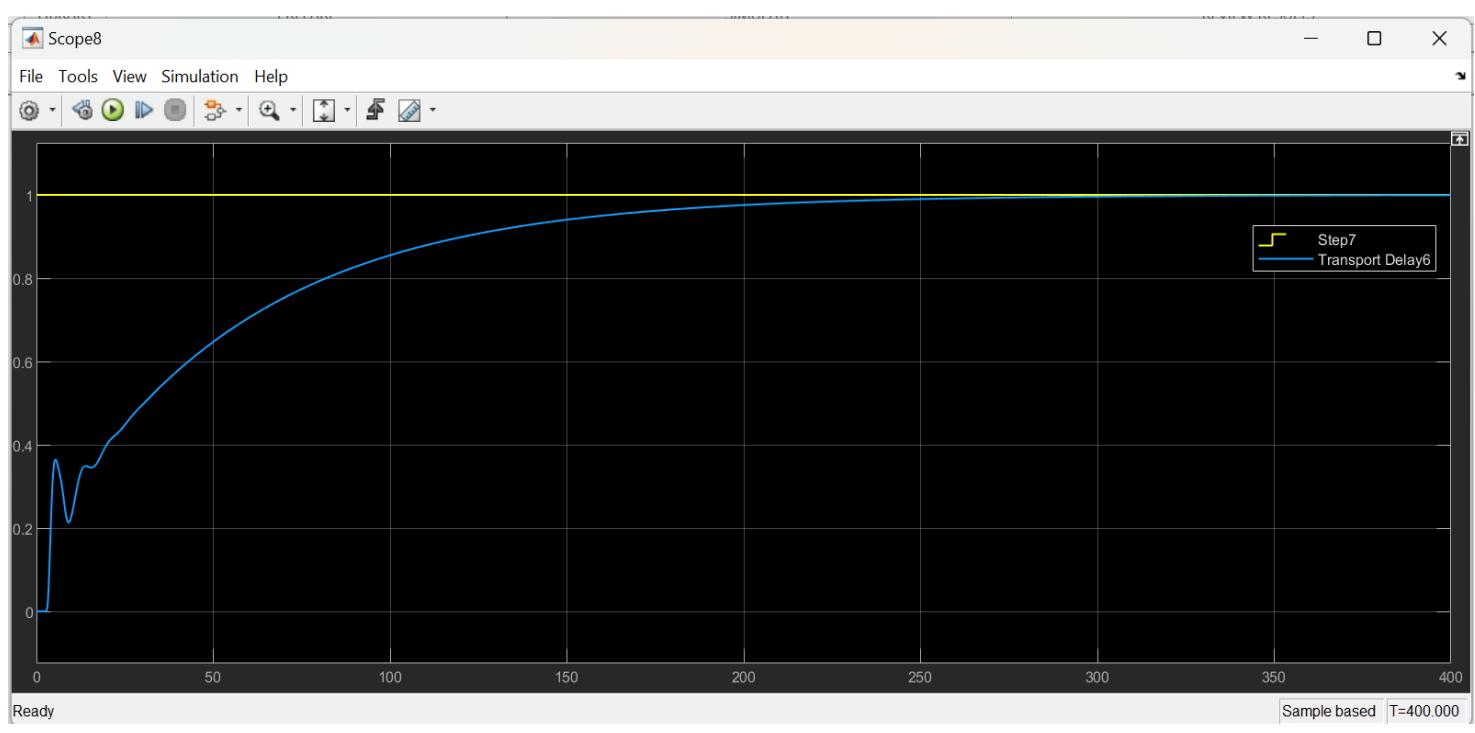


وارد تنظیمات بلوک PID شده و مطابق آنچه برای قسمت قبل انجام داده برای این قسمت نیز انجام میدهیم.

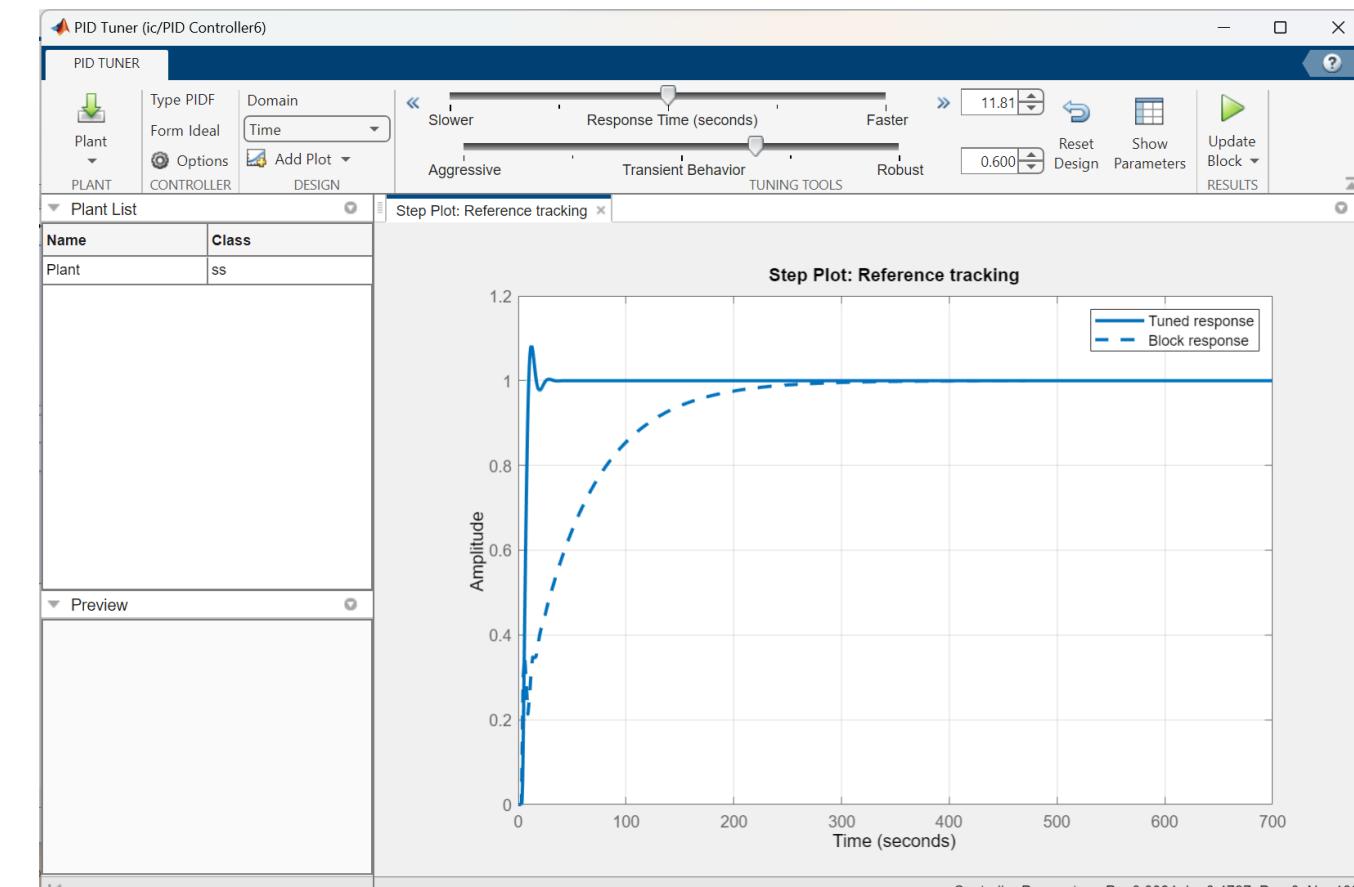
در نهایت بخش تنظیمات بلوک ما به شکل زیر خواهد شد.



شبیه سازی به شکل زیر است:



همانطور که از نمودار مشخص است و با جایگذاری ۰.۲۵ به جای آلفا یک پاسخ با زمان نشست تقریبا طولانی ولی بدون اورشوت داریم. برای بهبود عملکرد آن میتوانیم با تغییر متغیر های داده شده در مسئله پاسخ خود را بهبود بخشیم اما در روش دوم همانند قسمت های قبلی با تیون کردن این کار را انجام میدهیم که نتایج آن در جلوتر آمده است.



Compensator formula

$$P \left(1 + I \frac{1}{s} + D \frac{N}{1 + N \frac{1}{s}} \right)$$

Main Initialization Output Saturation Data Types State Attributes

Controller parameters

Source: internal

Proportional (P): 0.338426567011233

Integral (I): 0.478698785745271 Use I*Ts (optimal for codegen)

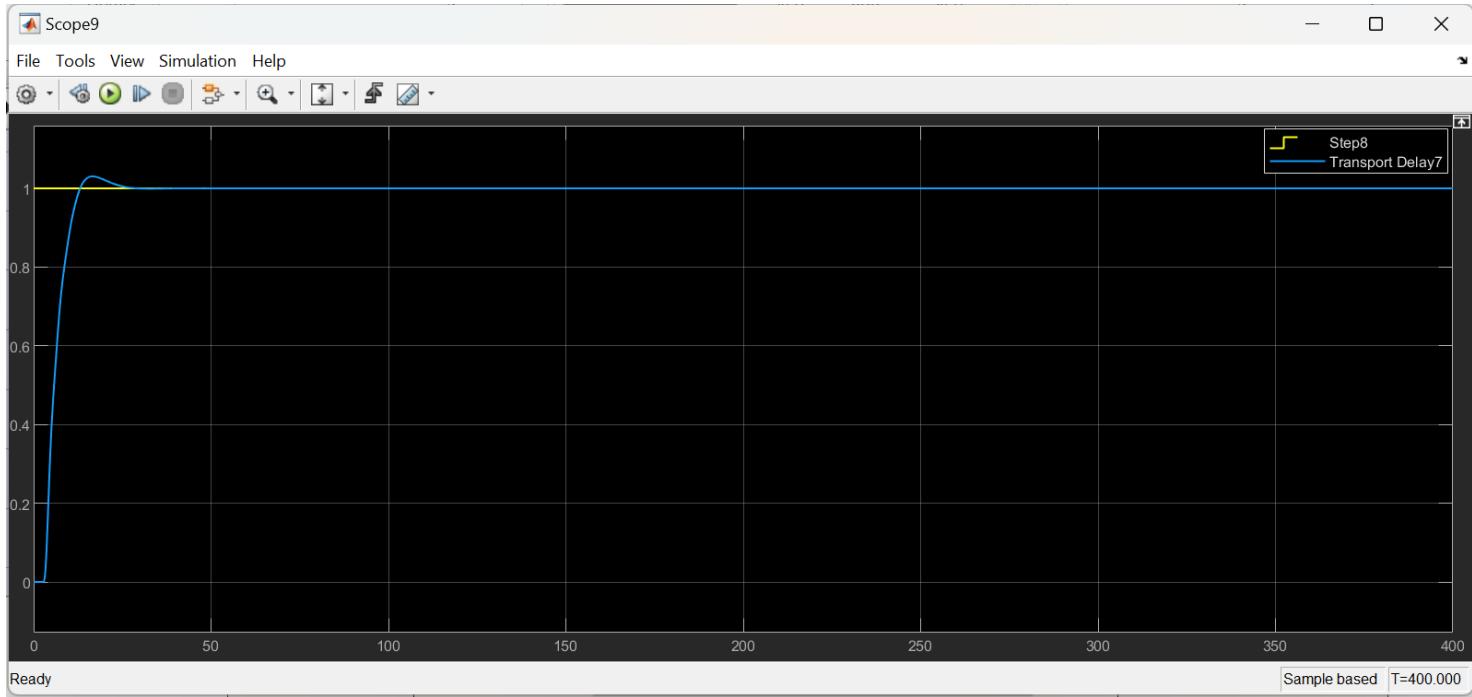
Derivative (D): 1.2

Filter coefficient (N): 100 Use filtered derivative

Automated tuning

Select tuning method: Transfer Function Based (PID Tuner App)

Enable zero-crossing detection



تحلیل:

همانگونه که در قسمت های قبل دیده شد با تیون کردن و تغییر آن و دادن مقدار به ضریب مشتق گیر ما پاسخ بهتری خواهیم داشت . همانگونه که گفته شد میتوانستیم با تغییر اندازه r_1 و r_2 به این مهم دست پیدا کنیم اما با روش تیون کردن نیز به این مهم دست پیدا کردیم و پاسخ ما بهتر و زمان نشست آن بهتر شد.

در ادامه با تغییر r_1 و r_2 مجدد شبیه سازی را برای دریافت پاسخ بهتر امتحان میکنیم.

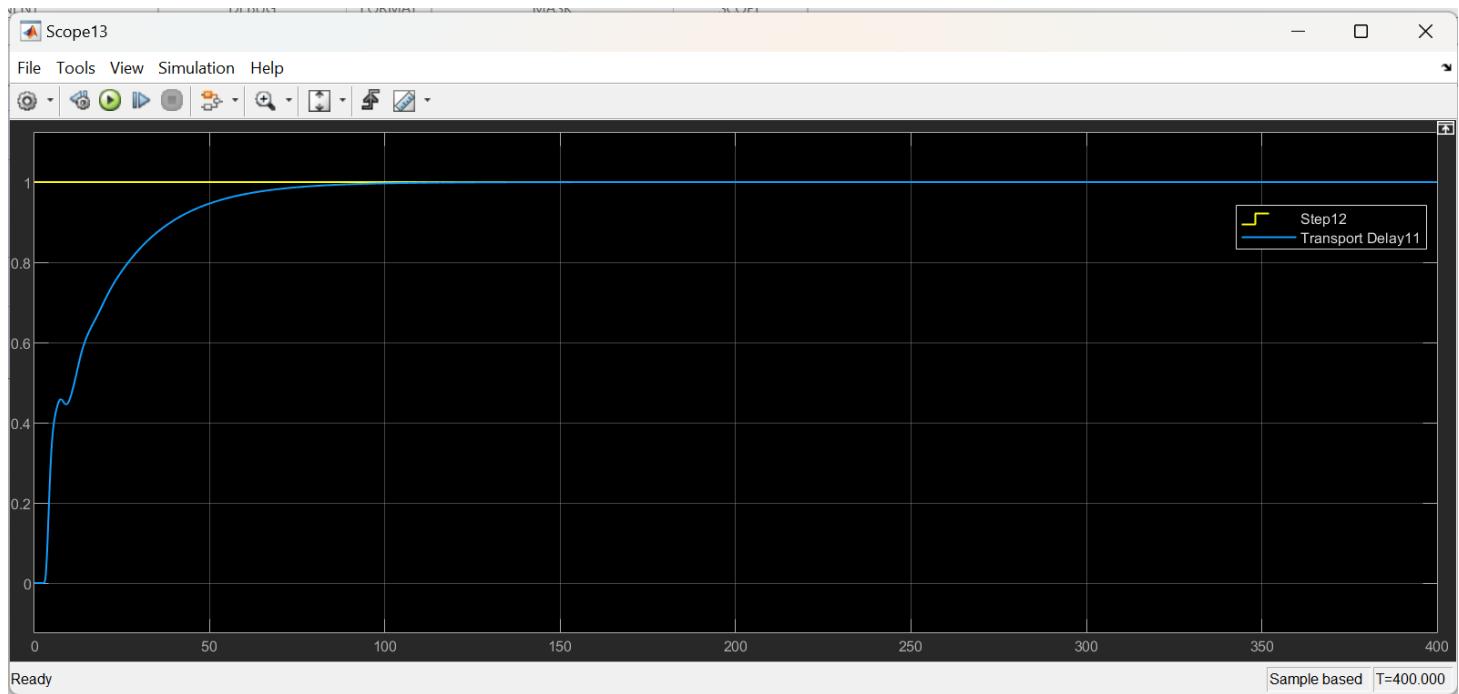
$$\vartheta_b = 30 \quad r_b = 0.4$$

مقادیر مربوط به اطلاعات نقطه نهایی نیز از قسمت های قبل بدست آورده بودیم:

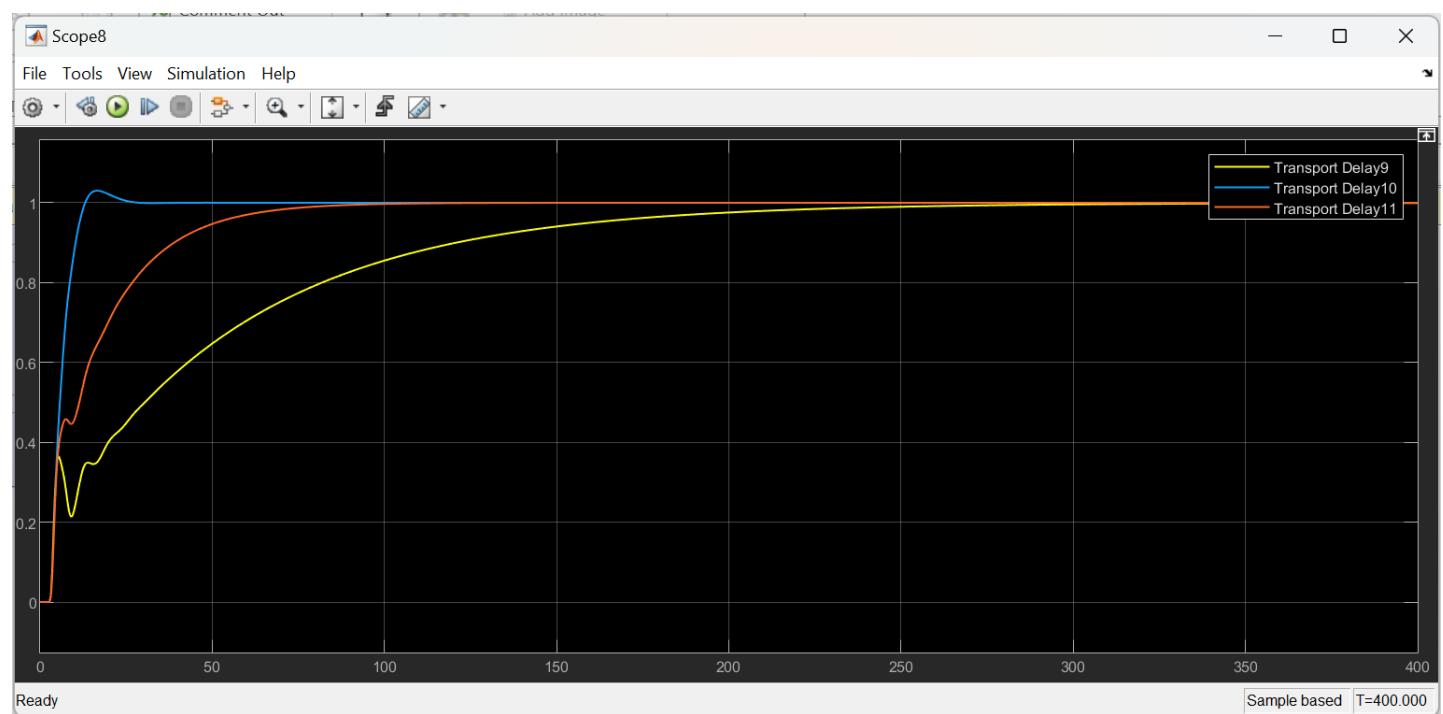
$$K_u = 1.275 \quad T_u = 9.991$$

پس ضرائب با یک محاسبه ساده بدست خواهد آمد:

$$K = 0.331, \quad T_i = 5.508, \quad T_d = 1.254$$



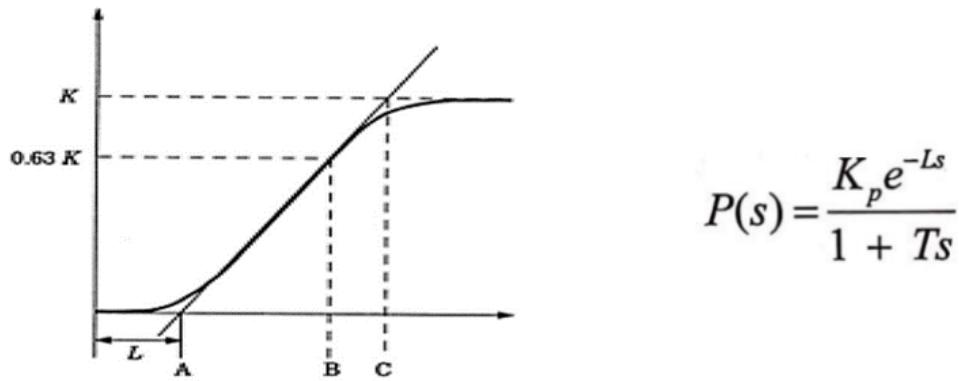
مقاييس:



۴-۶ طراحی به روش تنظیم λ

این روش مبتنی بر جایابی قطب های حلقه بسته سیستم است. سیستم متشکل از دو جزء $P(s)$ و $C(s)$ است. با نوشتن معادله مشخصه حلقه بسته، قطب ها در مکان دلخواه که وابسته به مقدار λ است، قرار می گیرد.

فرض می شود که پلنت به صورت مدل سه جزئی زیر تعریف شده است:



که در بخش ۳ گزارش این مدل ها بدست آمده اند. دو مدل سه جزئی با استفاده از نقطه C و B در اختیار بود که در شبیه سازی مشاهده شده مدل B تقریب بهتری از سیستم اصلی است. پس از آن استفاده خواهیم کرد.

$$G(s) = \frac{1.001}{1 + 1.698s} e^{-3.66s}$$

میدانیم:

$$C(s) = K \left(1 + \frac{1}{T_i s} + T_d s \right) = K \left(\frac{1 + T_i s + T_i T_d s^2}{T_i s} \right)$$

سپس آن را با تبدیل هایی به فرم زیر در می آوریم:

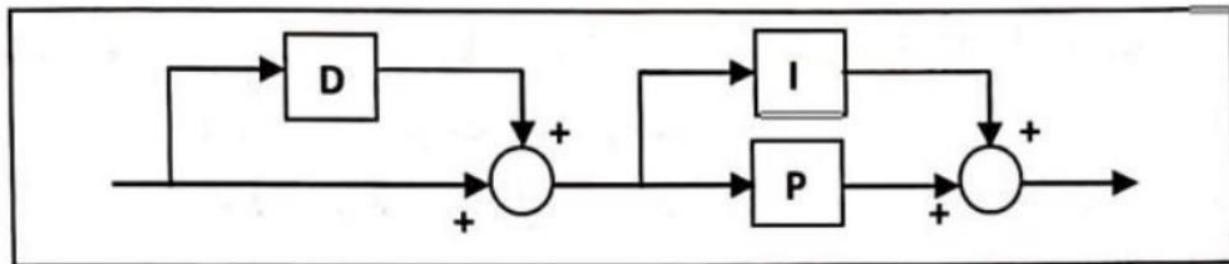
$$K' = K' \frac{T'_i + T'_d}{T'_i}; \quad T'_i = T'_i + T'_d; \quad T'_d = \frac{T_i T'_d}{T'_i + T'_d}$$

با تقریب ترم تاخیر به چندجمله ای (تقریب پده)، نوشتن معادله مشخصه

سیستم حلقه بسته تسهیل می شود.

$$C'(s) = K' \frac{(1 + T'_i s)(1 + T'_d s)}{T'_i s}$$

فرم تداخلی $C'(S)$ به صورت زیر بسته می شود:



سپس با قرار دادن قطب ساده معادله مشخصه در $s = -1/\lambda$ ضرایب کنترل بدست می آید.

در این مسئله ما $\lambda = 3T$ و یک بار $\lambda = T$ در نظر گرفته و آنها را باهم مقایسه میکنیم.

$$T'_i = T$$

$$T_i = T + \frac{L}{2}$$

$$T'_d = \frac{L}{2}$$

$$T_d = \frac{TL}{L + 2T}$$

$$K' = \frac{1}{Kp} \frac{T}{\frac{L}{2} + \lambda}$$

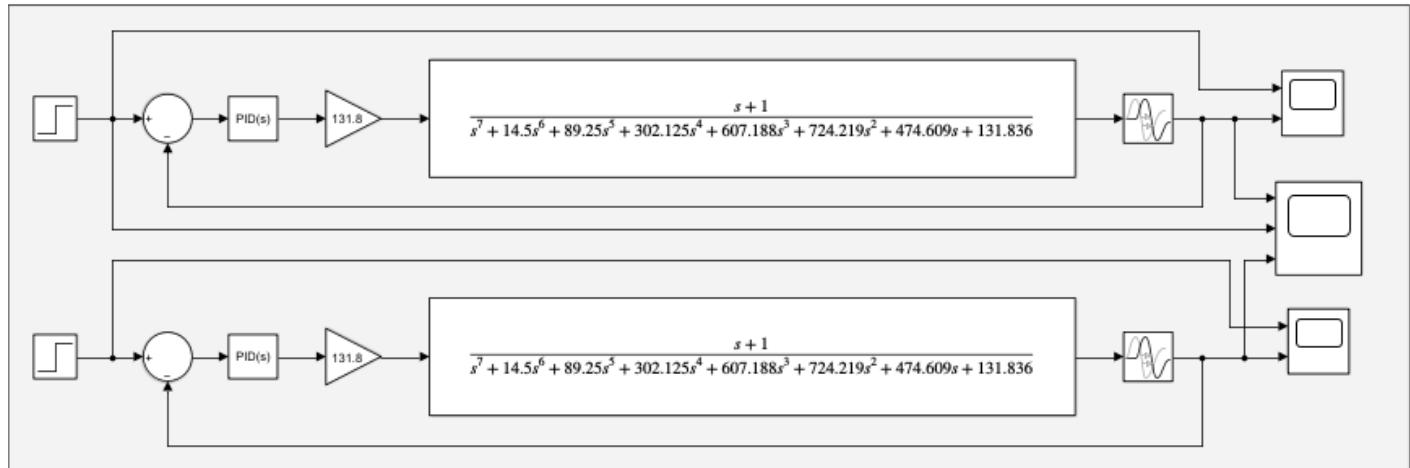
$$K = \frac{1}{Kp} \frac{\frac{L}{2} + T}{\frac{L}{2} + \lambda}$$

با انجام محاسبات ضرایب بصورت زیر بدست می آیند.

$$K = 0.51, T_i = 3.525, T_d = 0.881 \quad \text{When } \lambda = 3T$$

$$K = 0.99, T_i = 3.525, T_d = 0.881 \quad \text{When } \lambda = T$$

حال در محیط سیمولینک شروع به شبیه سازی میکنیم:



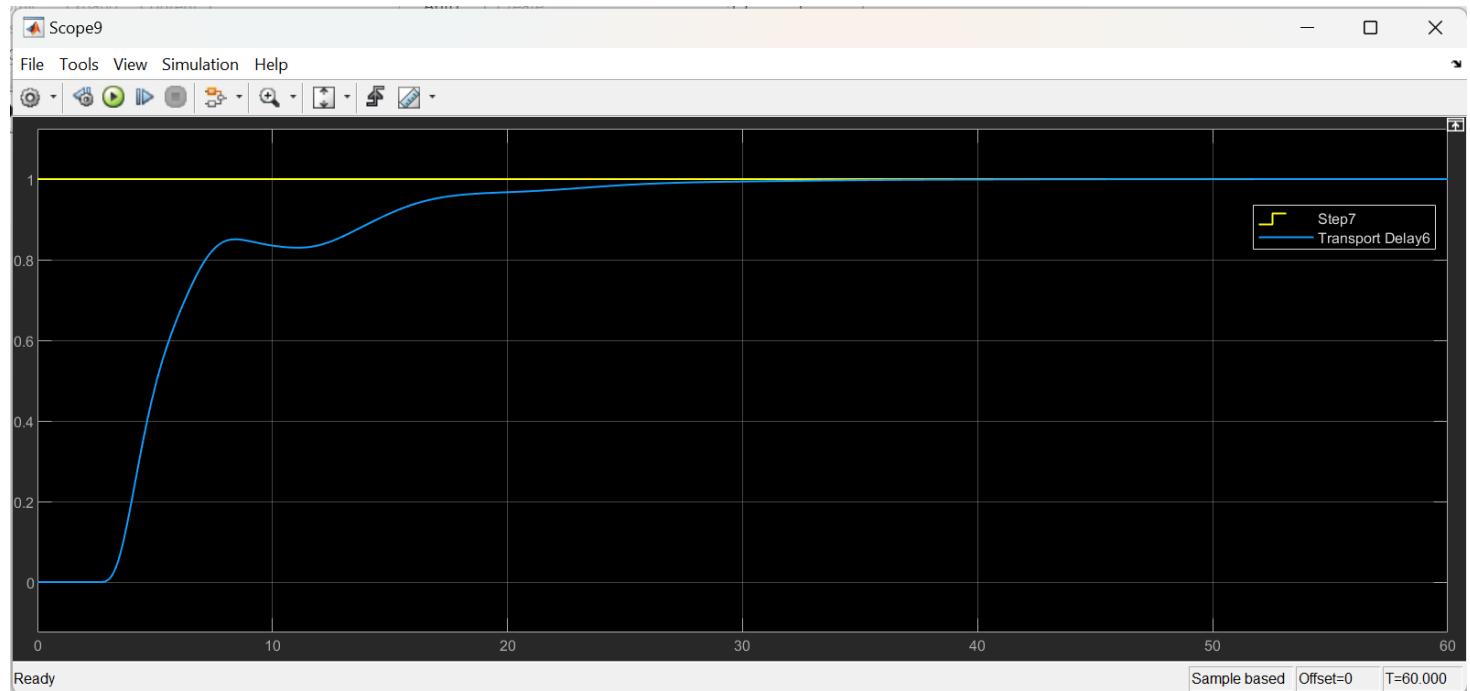
• پاسخ پله در حالت $\lambda = 3T$

Proportional (P): 0.51

Integral (I): 1/3.525

Derivative (D): 0.881

Use I*T_s (optimal for codegen)



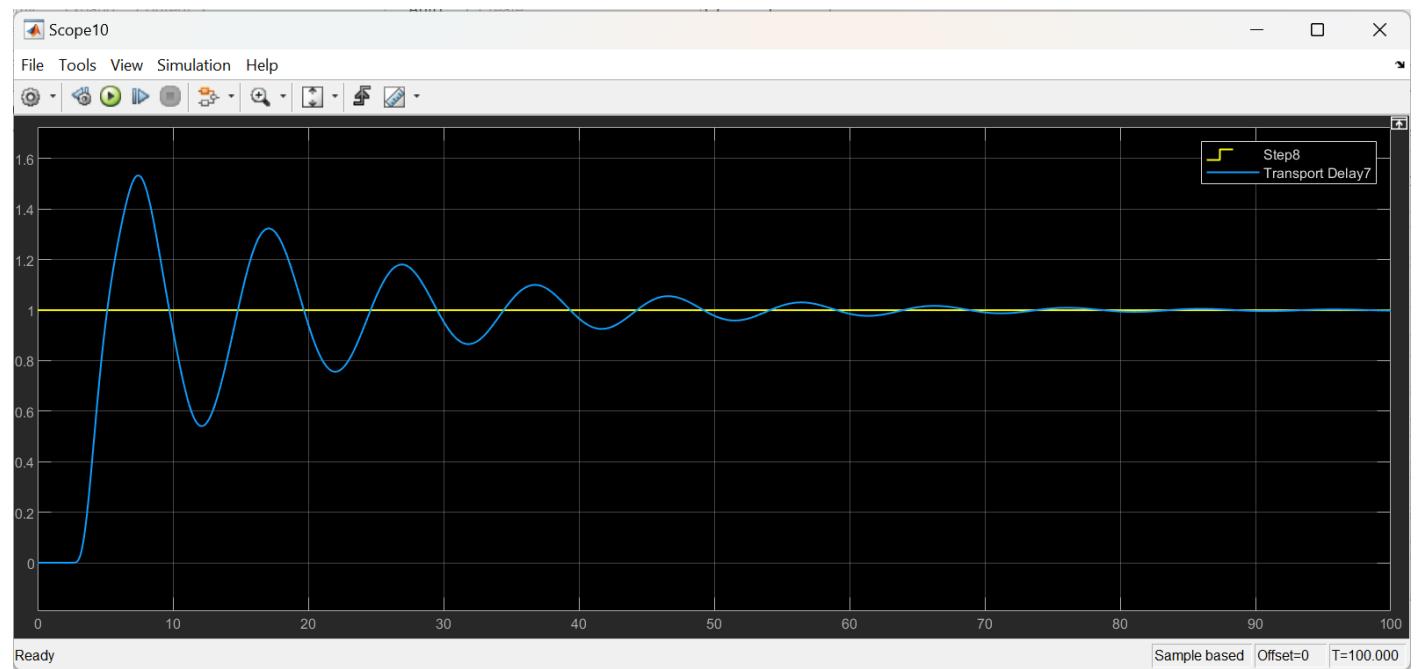
• پاسخ پله در حالت $\lambda = T$

Proportional (P): 0.99

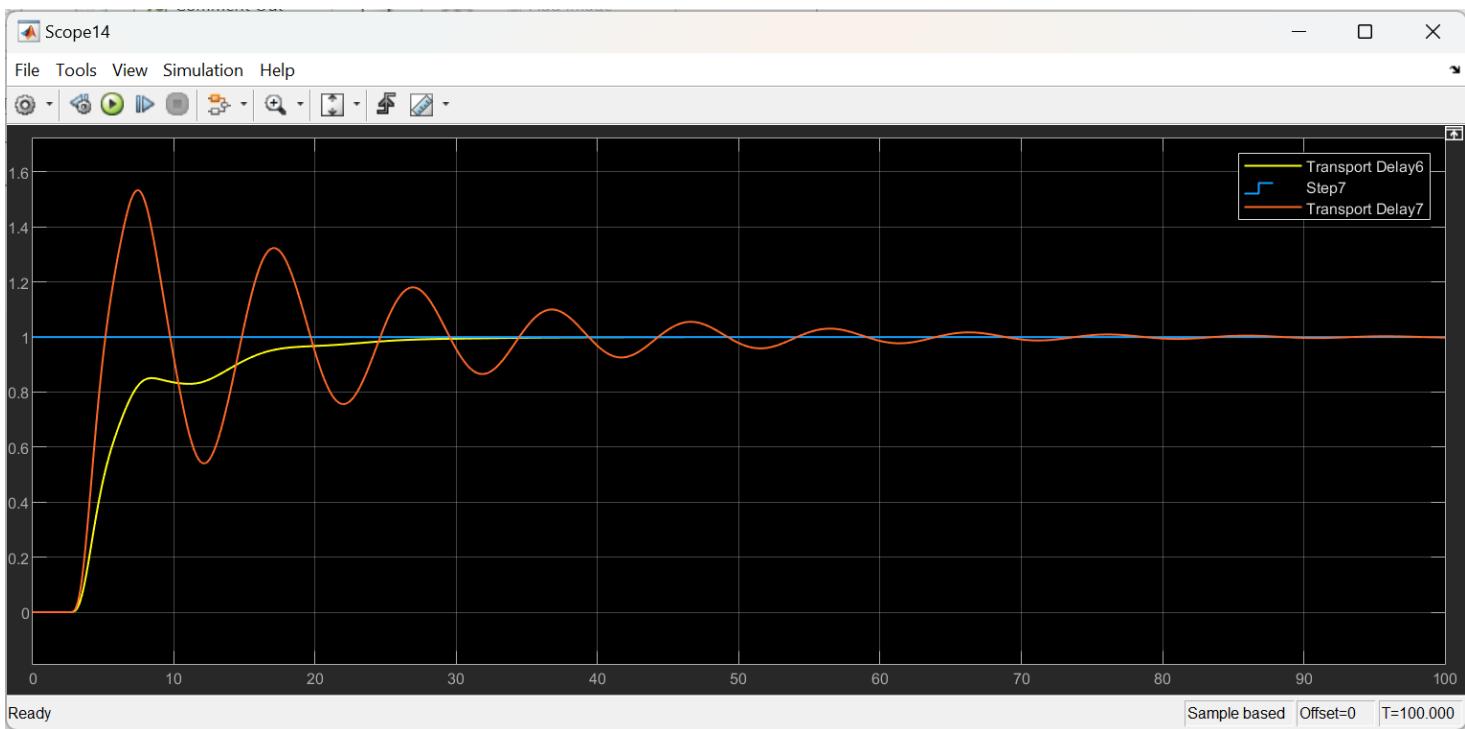
Integral (I): 1/3.525

Derivative (D): 0.881

Use I*T_s (optimal for codegen)



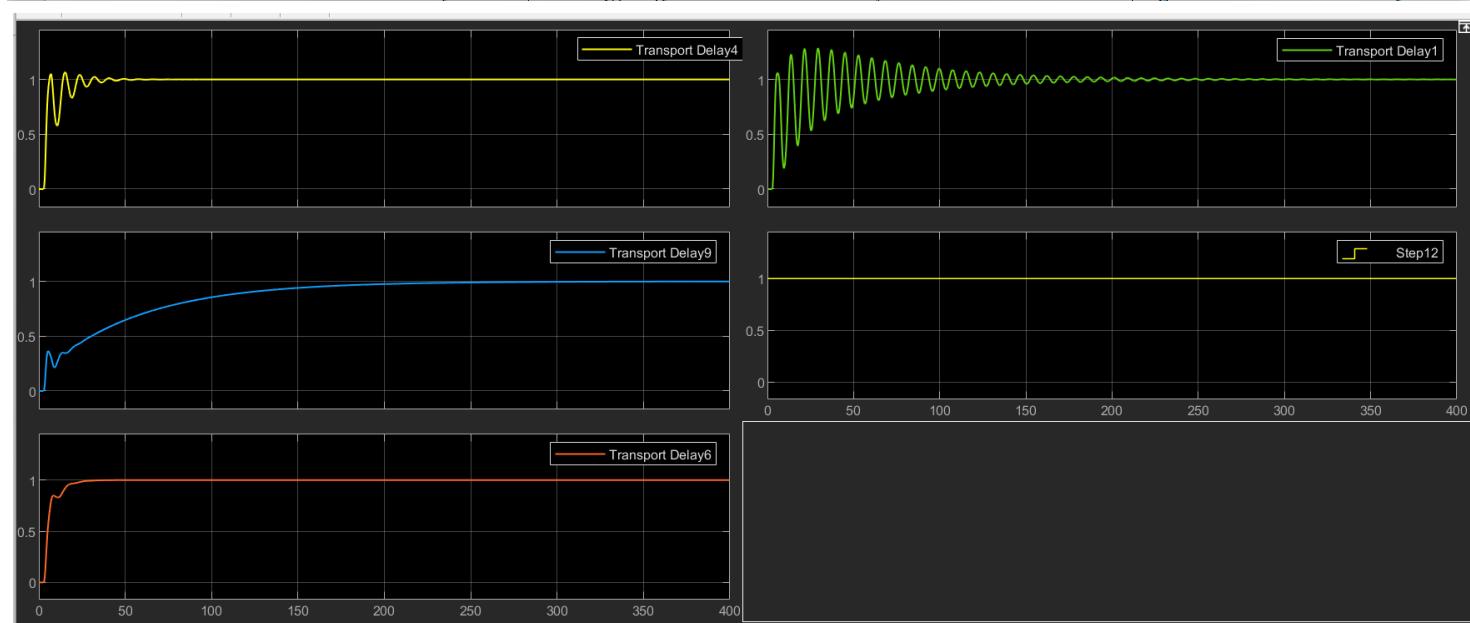
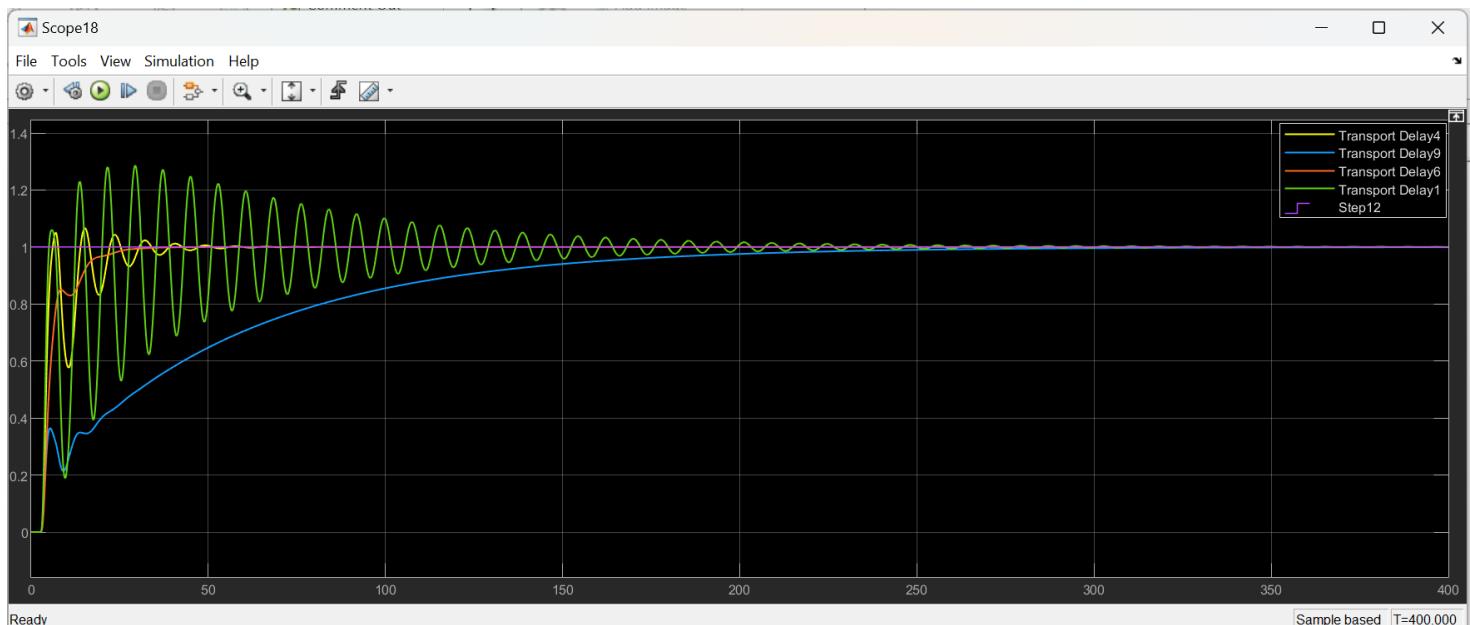
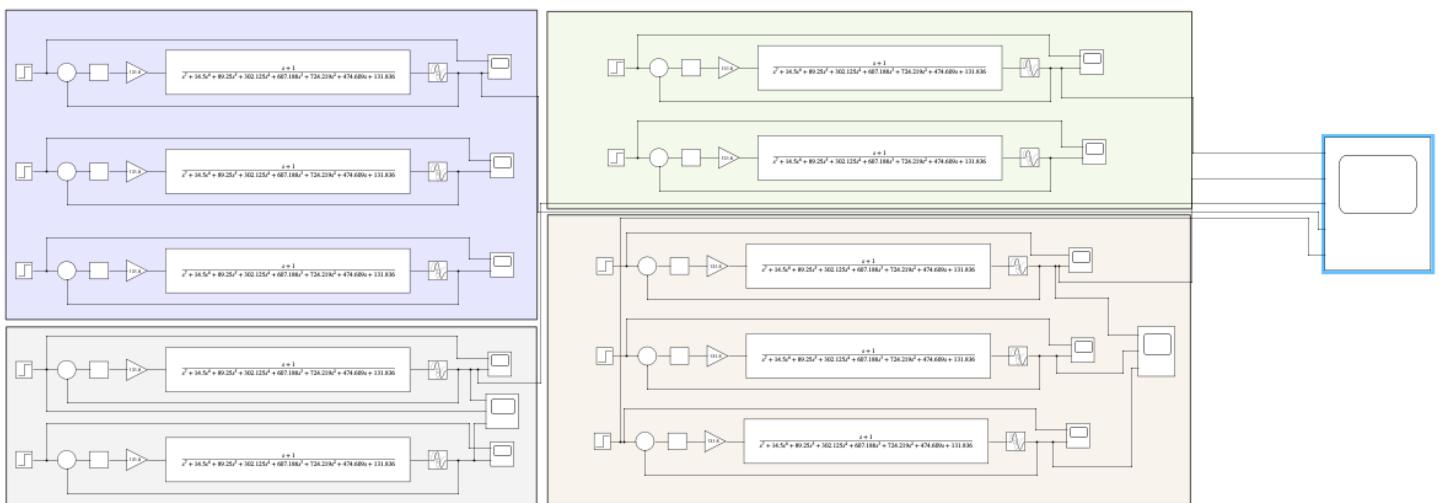
مقایسه و تحلیل:



این دو کنترل کننده در بخش های مشتق و انتگرال گیر دارای عبارت های یکسان هستند و تفاوت تنها در بخش تناسبی K است. با توجه به اینکه در طراحی با $3T$ بهره تناسبی کمتر است، پاسخ بدون نوسان است و به صورت نرم و کند به سمت پاسخ ماندگار حرکت کرده است. اما در طراحی با T با افزایش بهره تناسبی، سرعت افزایش یافته است اما پاسخ دارای فراجهش های زیادی است. در نتیجه به همین دلیل است که معمولاً $\lambda = 3T$ را در نظر میگیرند و با توجه به آن مسائل را حل میکنند. همانگونه که در تئوری درس توضیح داده شد به عنوان یک قاعده کلی میتوان برای داشتن یک کنترلر مقاوم $\lambda = 3T$ را در نظر گرفت اما در حالتی که پارامتر های مدل را به دقت میشناسیم میتوانیم آنرا با اختیاط برابر T در نظر بگیریم.

یکی از ضعف های این روش حذف قطب پایدار سیستم توسط صفر کنترل کننده است که برای فرآیند های با تاخیر غالب پاسخهای خوبی را تولید نخواهد کرد.

۷- مقایسه تمامی کنترل کننده ها



نمودار بالا مقایسه ای از تمامی پاسخ های به دست آمده در مدت زمان شبیه سازی ۴۰۰ ثانیه است. همانگونه که از نمودار های بالا مشخص است تمامی پاسخ ها تا قبل از اعمال سعی و خطا و تیون کردن آنها برای بهبود پاسخ آورده شده اند.

در روش ابتدایی ما یعنی روش زیگلر – نیکولز در حوزه زمان همانگونه که مشاهده میشود دارای زمان نشست بسیار زیادی بوده و همچنین دارای نوسانات شدیدی میباشد، سرعت پایداری مناسبی ندارد و در کل پاسخ خوبی برای پلت موردنظر ما نمی باشد.

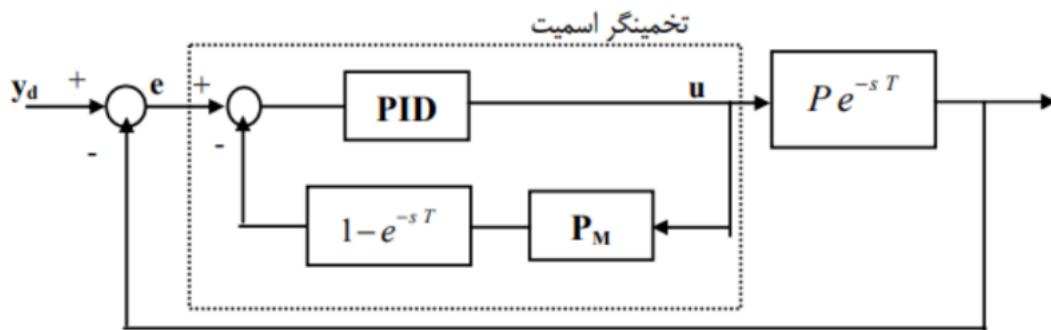
در روش زیگلر – نیکولز در حوزه فرکانس مشاهده میکنیم که این نوسانات و زمان پایداری ما به نسبت روش قبل بهبود نسبی پیدا کرده است اما همچنان دارای اورشوت و اندرشوت هایی است و از سرعت مطلوبی برخوردار است.

در روش زیگلر – نیکولز تعمیم یافته مشاهده میشود که نوسانات بسیار کنترل شده اما سرعت آن پایین بوده و سیستم در زمان نسبتاً زیادی به پایداری میرسد.

و در نهایت در روش λ مشاهده میکنیم که سیستم در حداقل نوسانات است و همچنین از سرعت مطلوبی برخوردار است و به پایداری میرسد. در نتیجه در شبیه سازی همانگونه که در آموزش تئوری نیز شرح داده شده بود در بین تمامی روش ها روش λ بهتر از همه عمل کرده و دارای سرعت مناسب و نوسانات پایین تری به نسبت بقیه روش های موجود میباشد.

-۸- طراحی کنترل کننده PID به همراه تخمین گر اسمنیت

تخمین گر اسمنیت یا Predictor Smith برای طراحی بهتر سیستم های کنترلی با تاخیر ثابت است. این روش، یک مسیر فیدبک در کنترل کننده طراحی شده از قبل قرار می دهد تا بتواند اثر تاخیر را در قطب های تابع تبدیل حلقه بسته را حذف کند یا بکاهد و بدین روش تاخیر را در خروجی کنترل نماید. زیرا که حضور تاخیر گاهای باعث ناپایداری سیستم و کنترل آن بسیار دشوار خواهد شد. در این روش تابع تبدیل سیستم حلقه بسته سیستم همانند سیستم بدون تاخیر است که یک ترم تاخیر در خروجی آن ظاهر شده است.



تابع تبدیل حلقه بسته این سیستم به صورت زیر است:

$$\frac{y}{y_d} = \frac{C(s)P(s)e^{-TS}}{1 + P_M(s)C(s)(1 - e^{-TS}) + C(s)P(s)e^{-TS}}$$

$$\frac{y}{y_d} = \dots = \frac{CPe^{-ST}}{1 + CP_M - CP_{Me}^{-ST} + CPe^{-ST}}$$

اگر بتوانیم تخمین خوبی از تابع تبدیل سیستم بدست آوریم، اثر عبارت تاخیر در مخرج از بین می رود.

$$\text{If } P = P_M \Rightarrow \frac{y}{y_d} = \frac{CP}{1 + CP} \cdot e^{-ST}$$

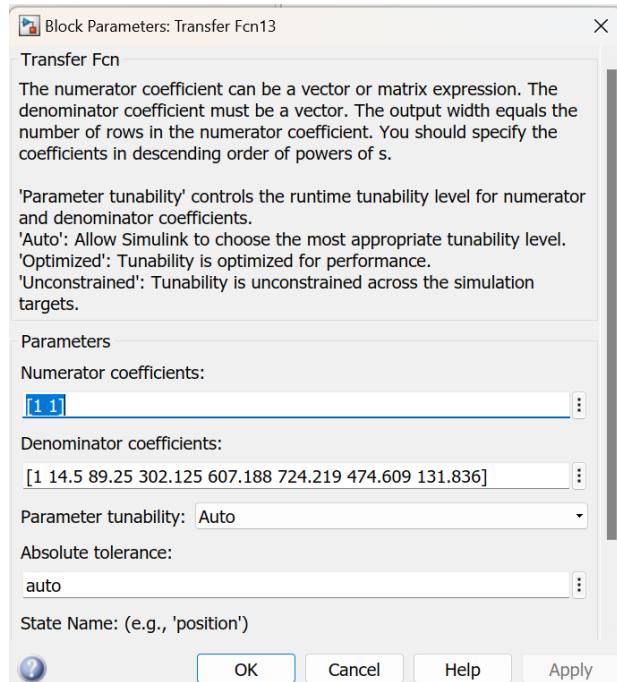
۱-۸ با اعمال سیستم اصلی

در این حالت بلوک دیاگرام به صورت زیر خواهد بود، مطابق دیاگرام بالا در سیمولینک شبیه سازی

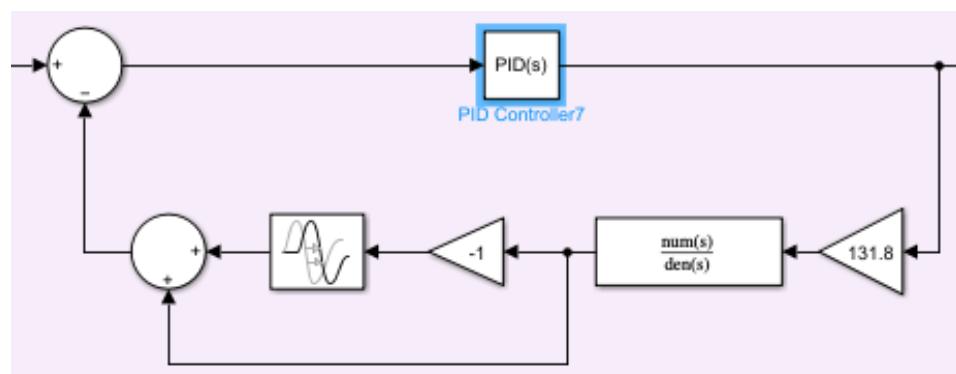
میکنیم و باید توجه داشت که چون در این بخش سیستم اصلی را اعمال میکنیم پس از خود سیستم در فیدبک (P_M) استفاده میکنیم.

مطابق آنچه در صورت سوال گفته شده برای بلوک کنترل کننده خود باید از ضرایب λ استفاده کنیم.

Proportional (P):	0.51	
Integral (I):	1/3.525	<input type="checkbox"/> Use I*T _s (optimal for codegen)
Derivative (D):	0.881	

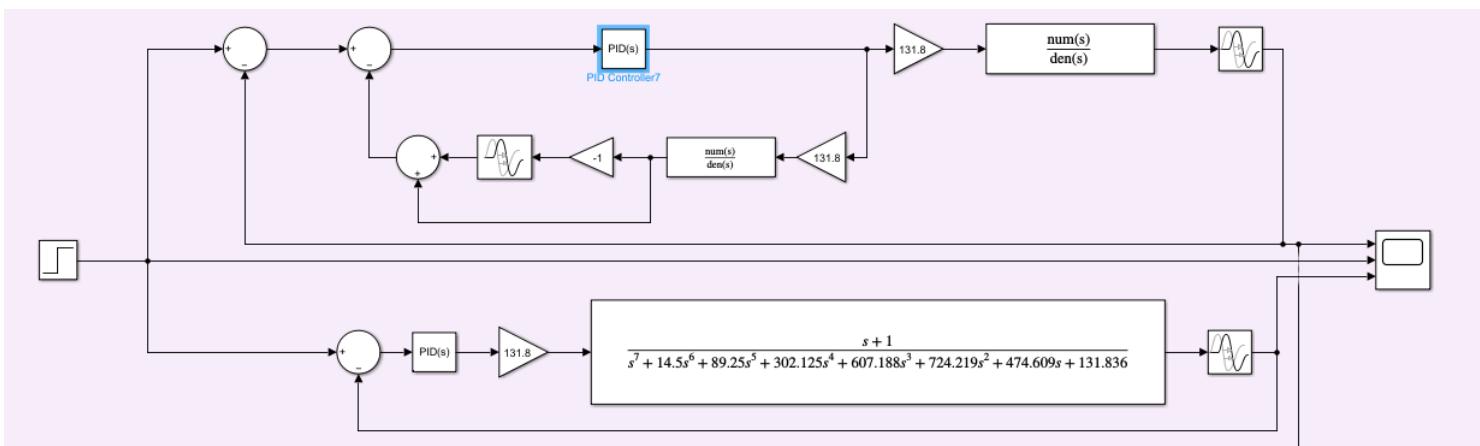


تخمینگر اسمیت ما به شکل زیر است:

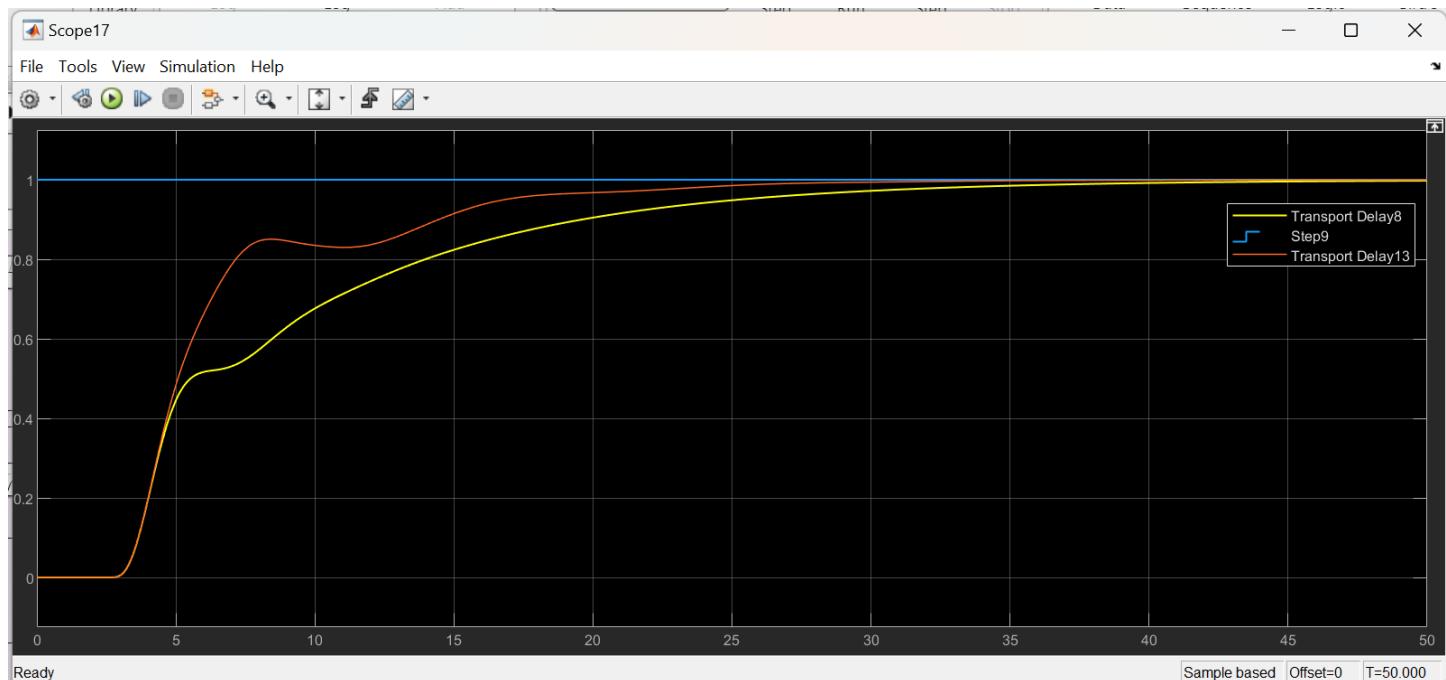


در شکل زیر طراحی کنترل کننده PID به روش اسمیت به همراه پلت اصلی ما برای اعمال مقایسه نشان

داده شده است.

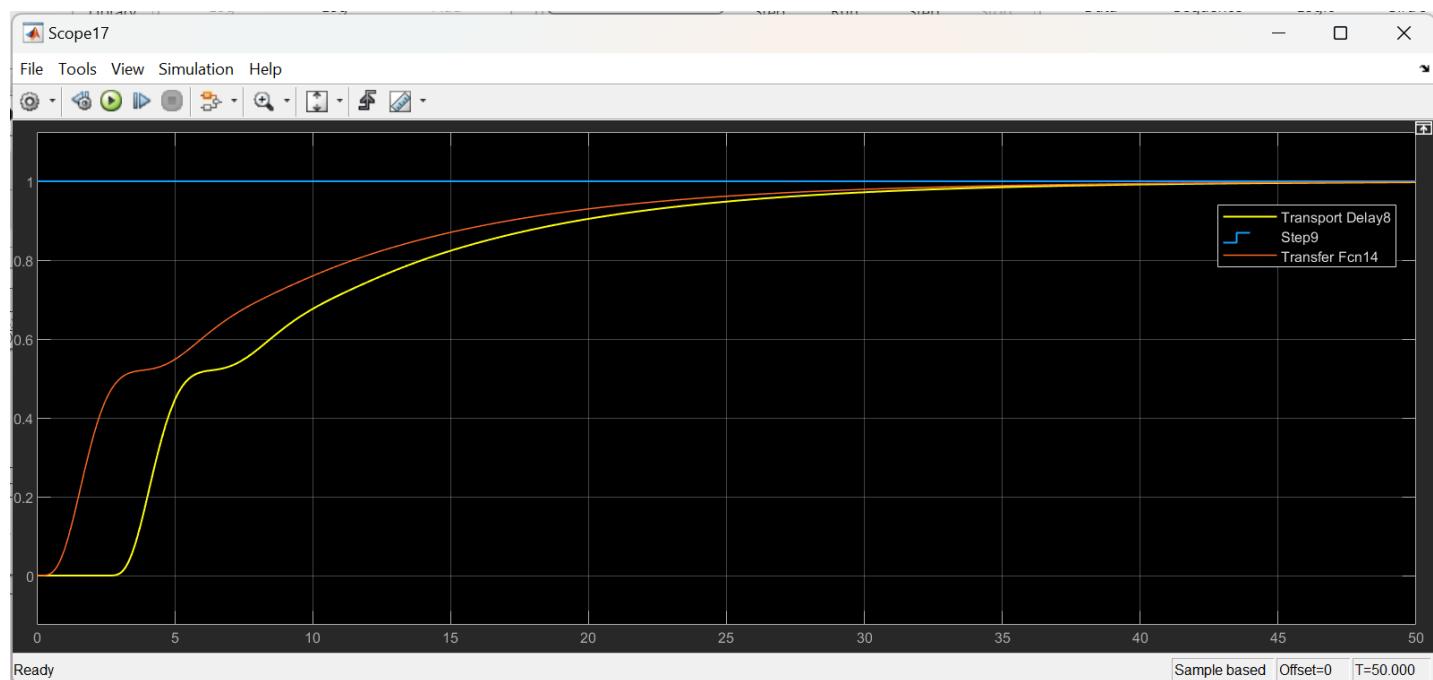
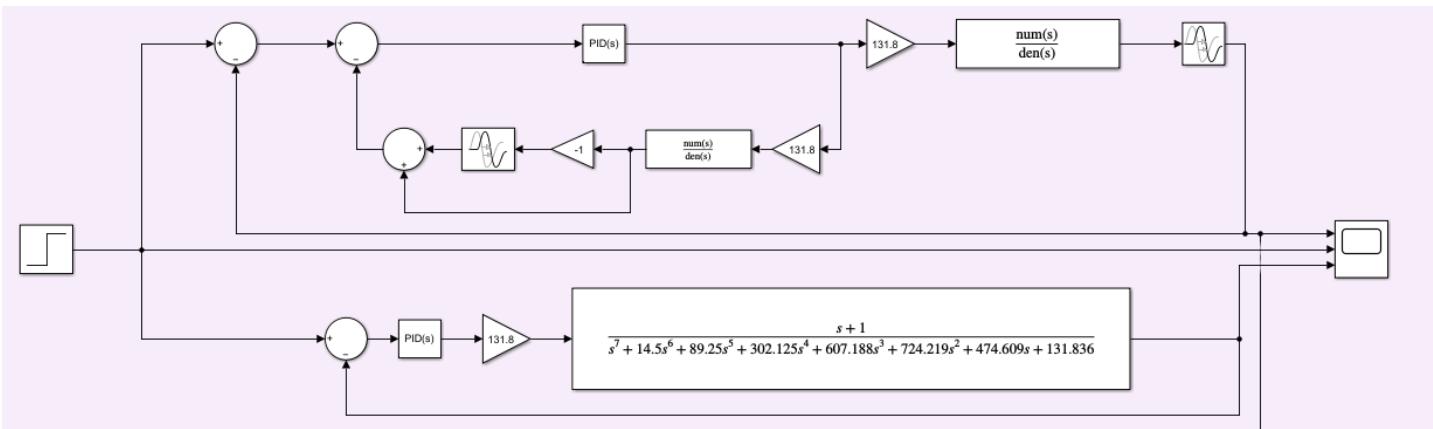


خروجی ما بر روی اسکوپ در شکل زیر آمده است.



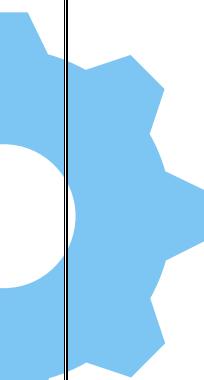
همانگونه که از نمودار مشاهده میشود که اثباتی بر ادعای تئوری ما مبنی بر حذف اثر تاخیر در تابع سیستم حلقه بسته است، تخمینگر اسمیت به خوبی توانسته است این تاخیر را از سیستم تابع حلقه بسته ما حذف کند اما همانگونه که مشاهده میشود باعث کاهش سرعت شده است.

برای درک تصویری این موضوع و اثبات ادعای ما ترم تاخیر را در سیستم اصلی حذف کرده و مجدد شبیه سازی میکنیم.



از نمودار مشاهده میشود که طبق محاسبات ما و دادن تاخیر ۲.۵ ثانیه ای دقیقاً این تاخیر در تخمین گر اسمیت ما

اعمال شده است که اثباتی بر ادعای تئوری ما مبنی بر حذف ترم تاخیر از سیستم حلقه بسته ما است.



۲-۸ طراحی با مدل های شناسایی شده

۱-۲-۸ طراحی با مدل سه جزئی

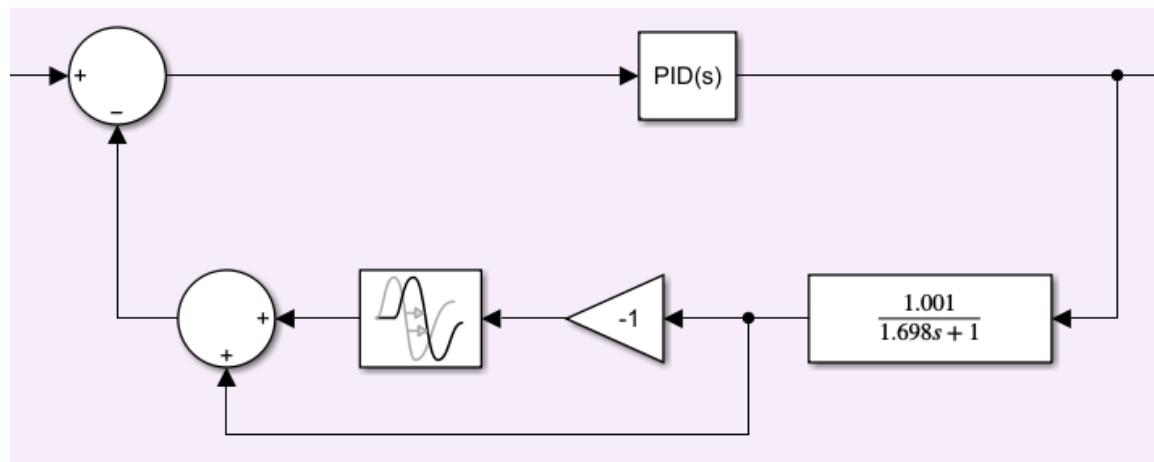
در این بخش به جای استفاده از سیستم اصلی از مدل سه جزئی شناسایی شده در قسمت های

قبل استفاده میکنیم. مدل سه جزئی ما به این صورت بود:

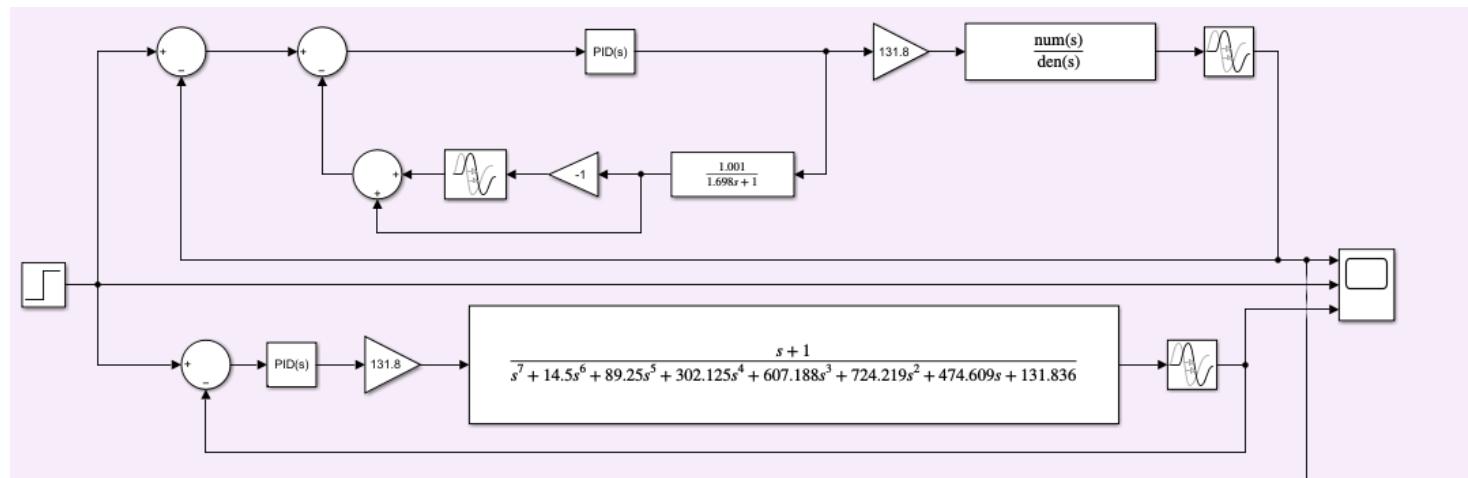
$$G(s) = \frac{1.001}{1 + 1.698s} e^{-3.66s}$$

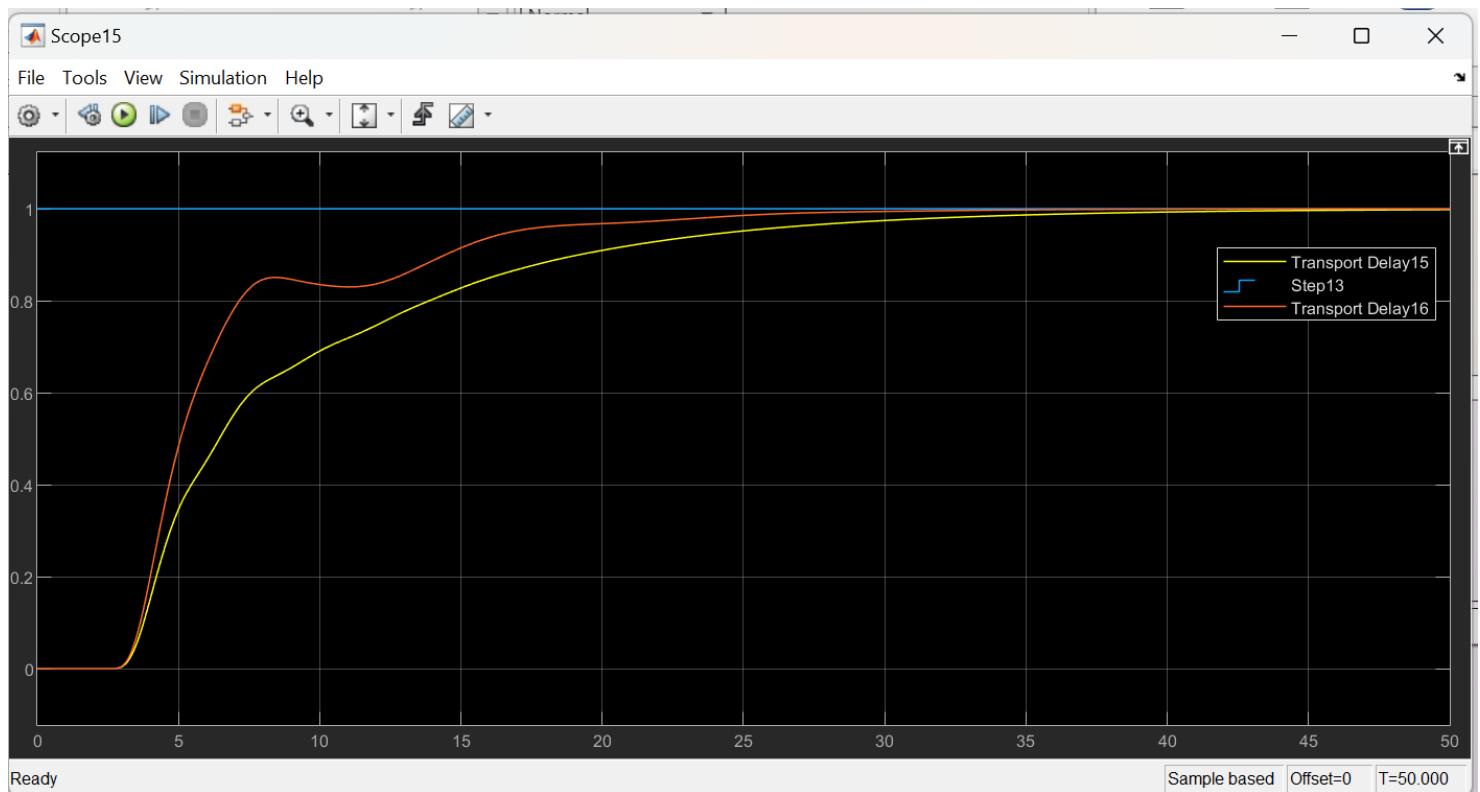
با اعمال این مدل به جای P_M در تخمینگر اسمنیت مدل را در سیمولینک شبیه سازی کرده و نتایج

را مشاهده میکنیم.

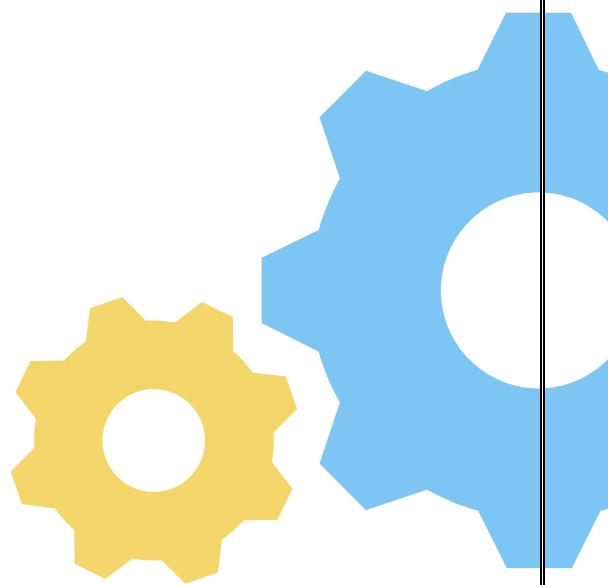


همانند بخش قبل این شبیه سازی را همراه با سیستم اصلی برای اعمال مقایسه انجام میدهیم.





مشاهده میشود که این تخمین به خوبی تخمین قبل نیست و نمیتوان با اطمینان گفت تاخیر را از تابع سیستم حلقه بسته ما حذف کرده است.



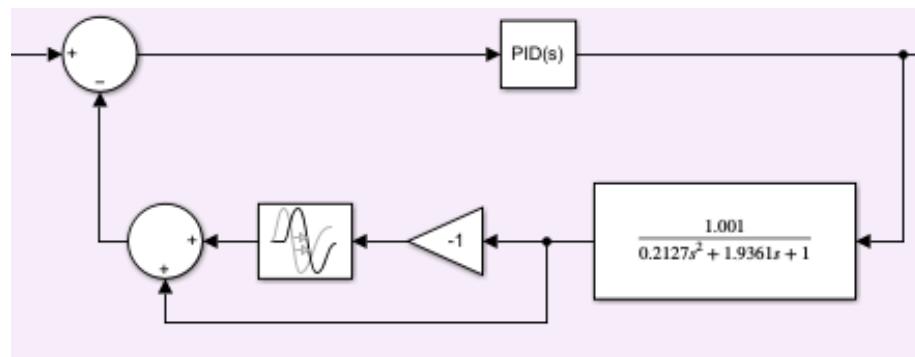
۱-۲-۸ طراحی با مدل چهار جزئی

مشابه بخش قبل در این بخش به جای استفاده از سیستم اصلی از مدل چهار جزئی شناسایی

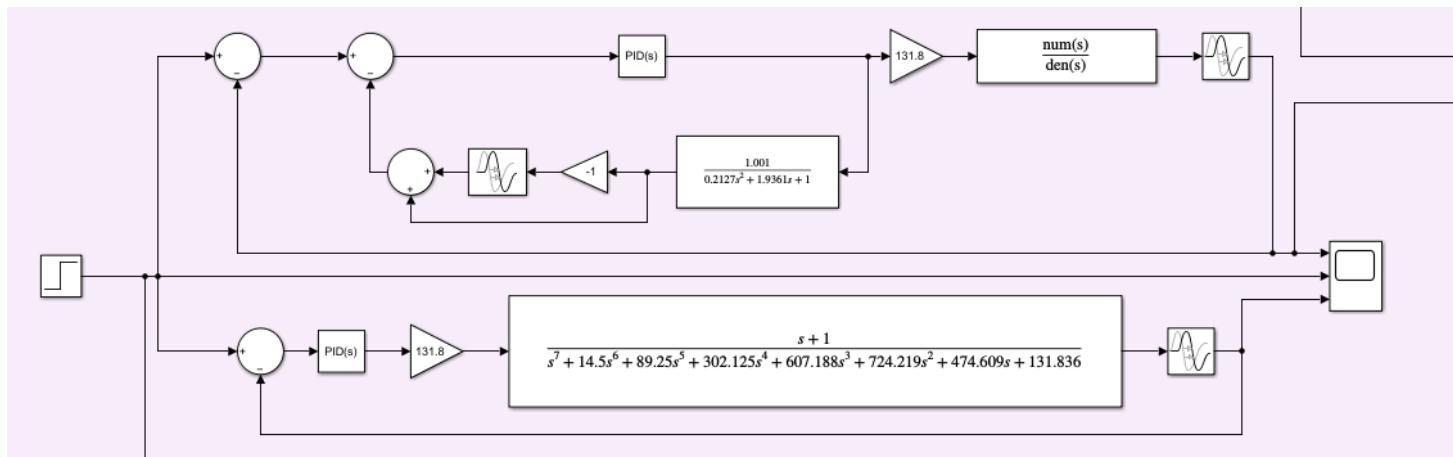
شده در قسمت های قبل استفاده میکنیم. مدل چهار جزئی ما به این صورت بود:

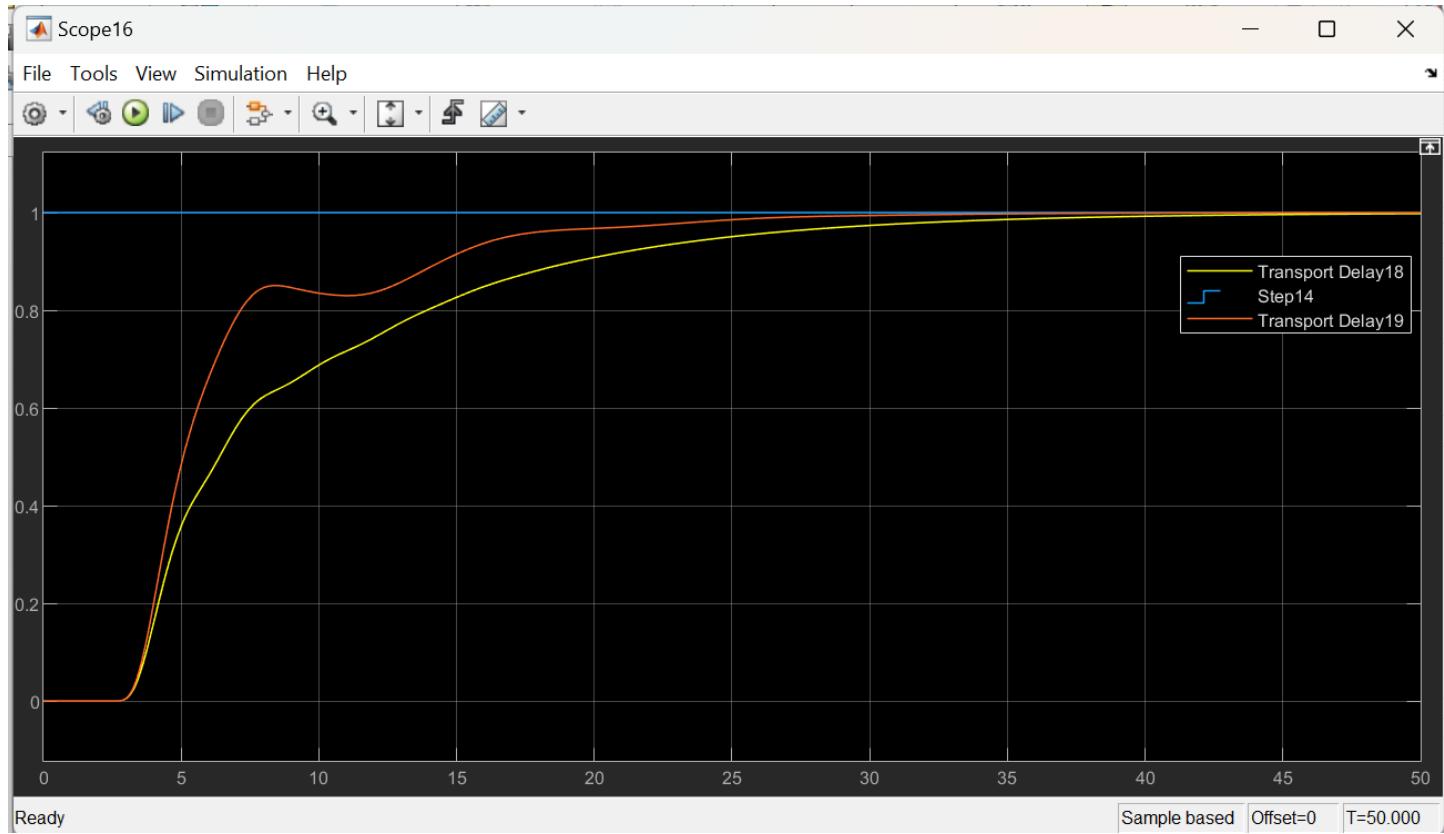
$$G_{(s)} = \frac{1.001}{(1 + 0.1169s)(1 + 1.8192s)} e^{-3.66s}$$

با اعمال این مدل به جای P_M در تخمینگر اسミت مدل را در سیمولینک شبیه سازی کرده و نتایج را مشاهده میکنیم.

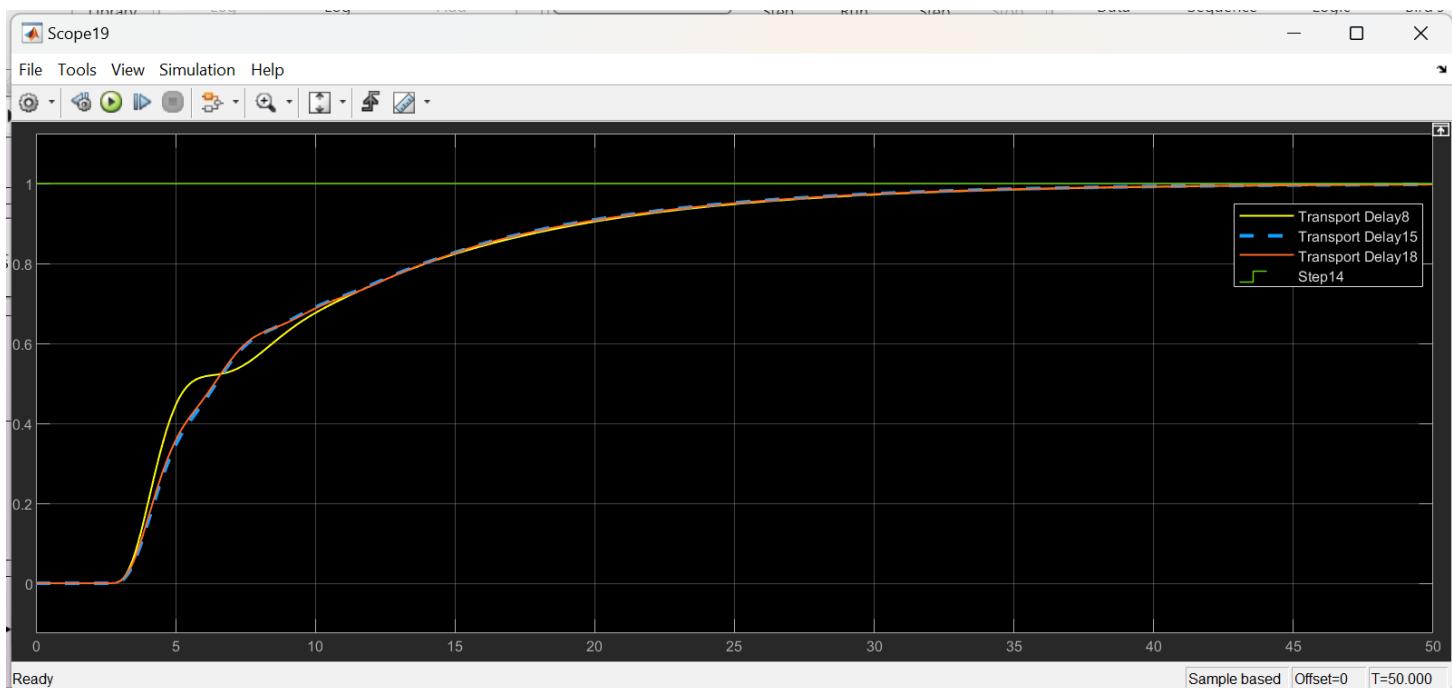


همانند بخش قبل این شبیه سازی را همراه با سیستم اصلی مقایسه انجام میدهیم.





مقایسه:

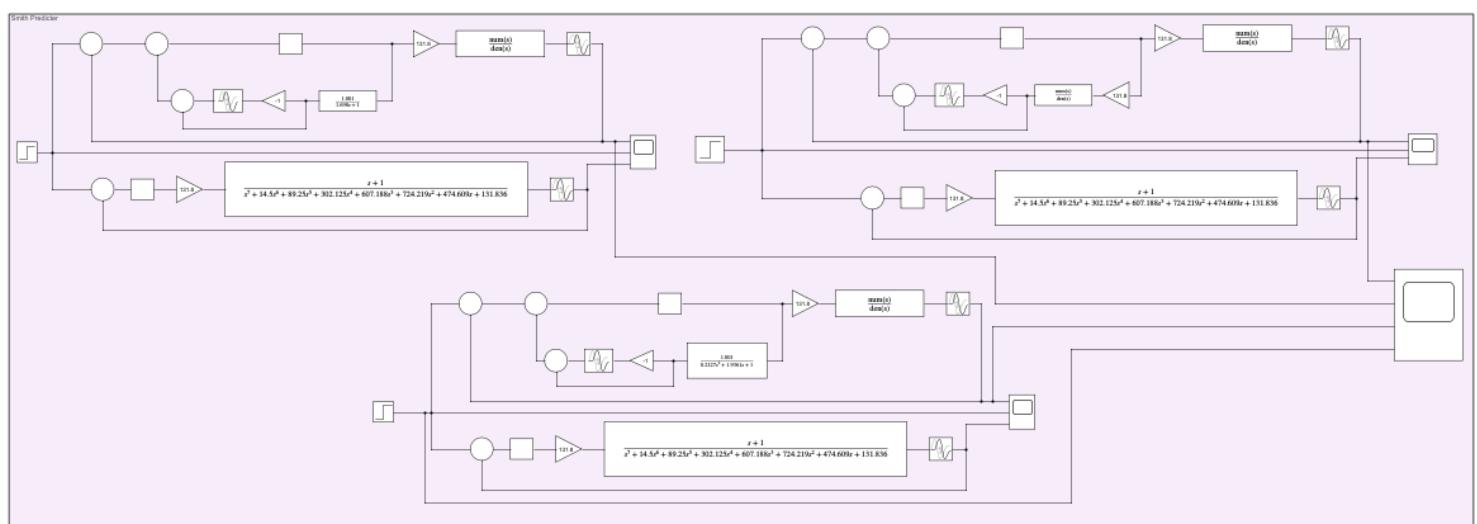
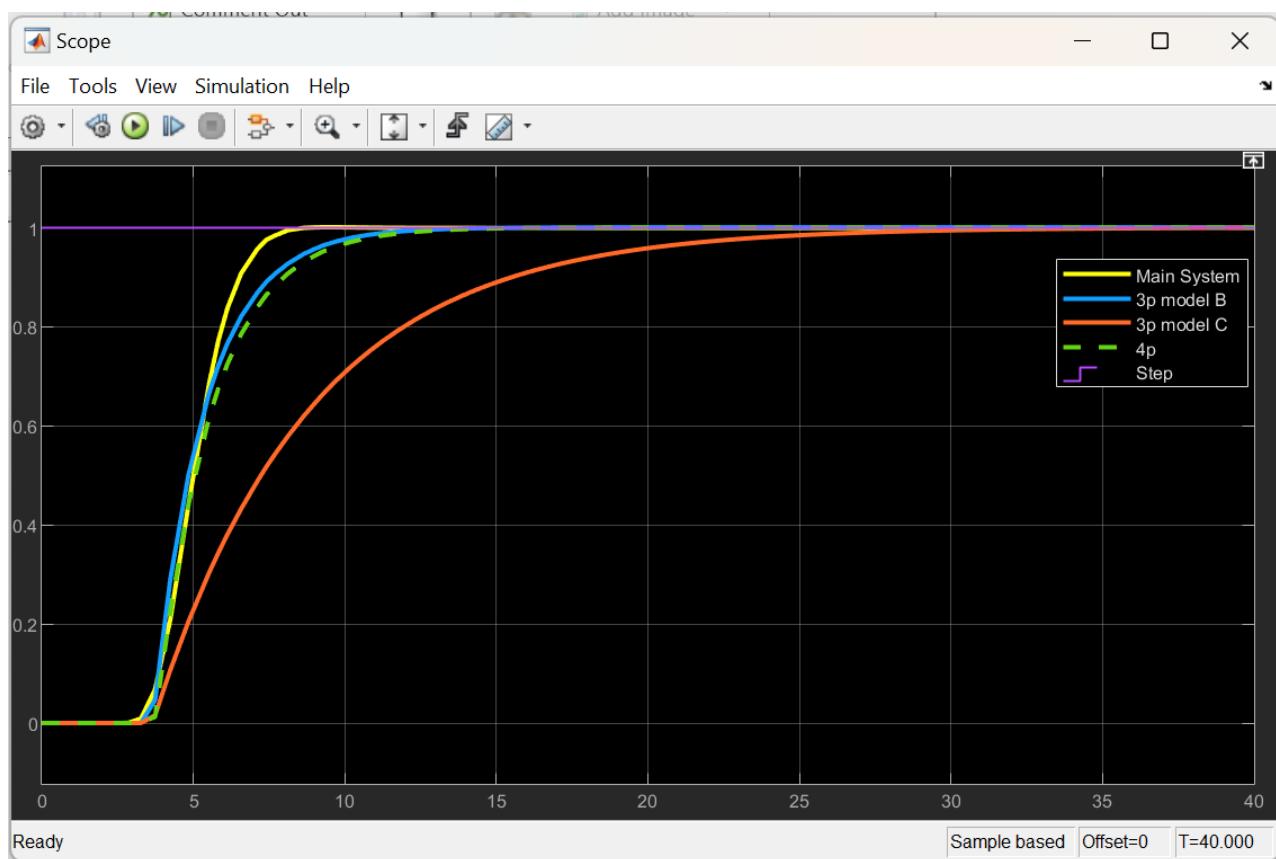


همانگونه که در بخش های قبل گفته شد و از نمودار رو به رو مشخص است تخمینگر اسمیت هنگامی که سیستم

اصلی را به آن اعمال کرده باشیم پاسخ بسیار بهتری دارد اما وقتی مدل سه جزئی و چهار جزئی را اعمال کردیم

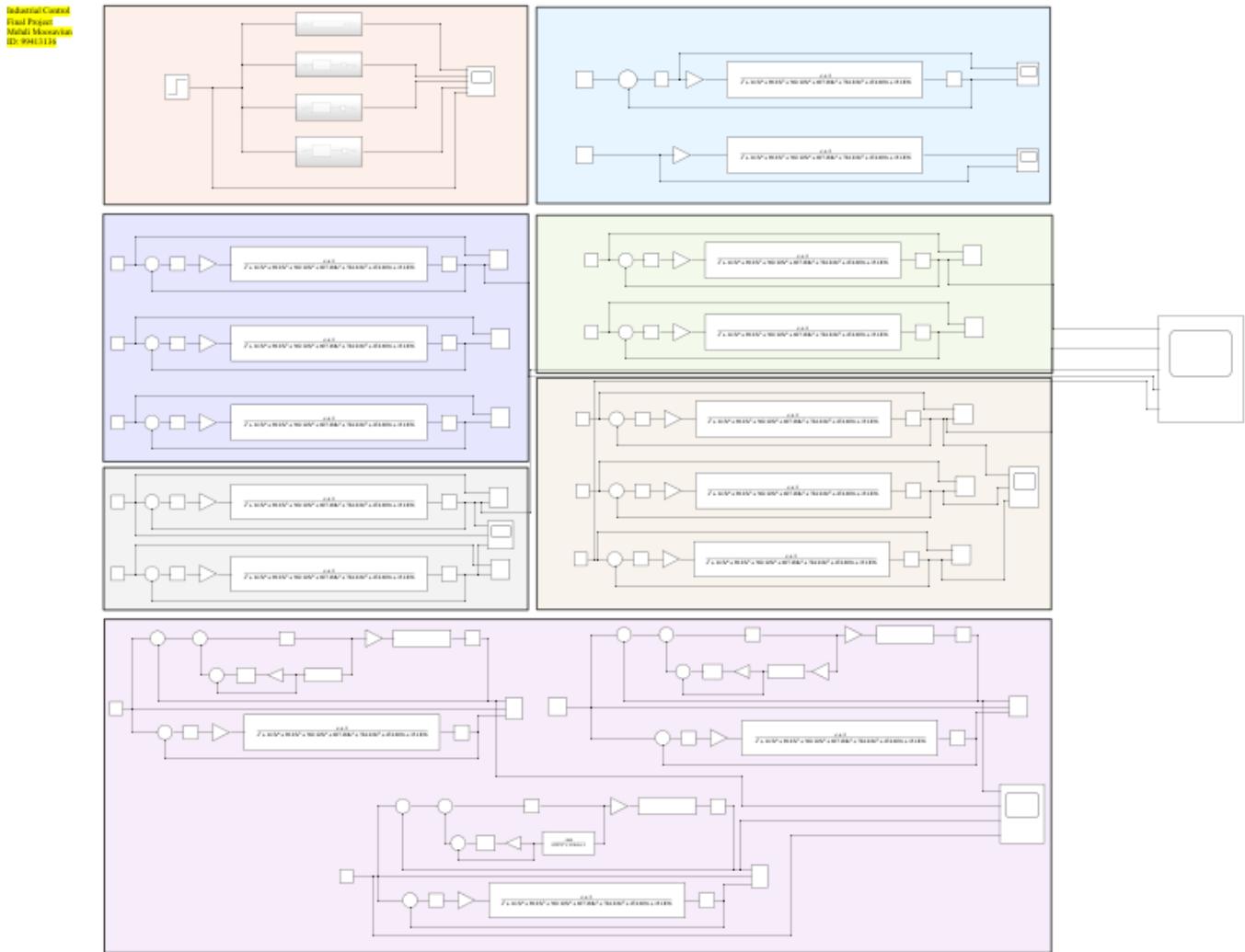
نتوانست به خوبی تاخیر را از سیستم اصلی حذف کند. در تئوری ما میدانیم هرچه که از مدل های دو جزئی به

چهار جزئی میرویم پاسخ ما بهبود پیدا میکند که این یک اصل اثبات شده و پذیرفته شده است، اما در این سیستم در تخمین گر اسمیت این موضوع رخ نداد و پاسخ سیستم سه جزئی و چهار جزئی تقریبا همپوشانی داشت که این میتواند دلیلی بر کارایی و دقیقی مدل سه جزئی بدست آمده ما باشد. در بخش ۲ نیز در قسمت مقایسه مشاهده شد که پاسخ مدل سه جزئی ما بسیار به مدل چهار جزئی نزدیک بود که برای یادآوری شکل آن در زیر آورده شده است.



۹- شبیه سازی ها

در انتهای یک اسکرین شات از شبیه سازی های انجام شده در محیط سیمولینک مطلب برای این پروژه آورده شده است.



با تشکر از توجه شما

سید مهدی موسویون

دی ماه ۱۴۰۲