**گزارش پروژه اول**

**سیستم های کنترل خطی**

**نگار میرگتی ۸۱۰۱۹۴۴۱۳**

**بدست آوردن نقاط تعادل-1**

a = 4, b = 1, c = 3 =>

α = 4, β = 4/10, γ = 3/10, δ = 3

در نتیجه معادلات به شکل زیر در خواهند آمد‌ :

= 4x –(4/10)xy

= 3xy – 3/10y

شرط تعادل :0 = برای معادله اول و برای معادله دوم.

4x – (4/10)xy = 0 => 4x = (4/10)xy => 1 : x = 0 , 2 : y = 10

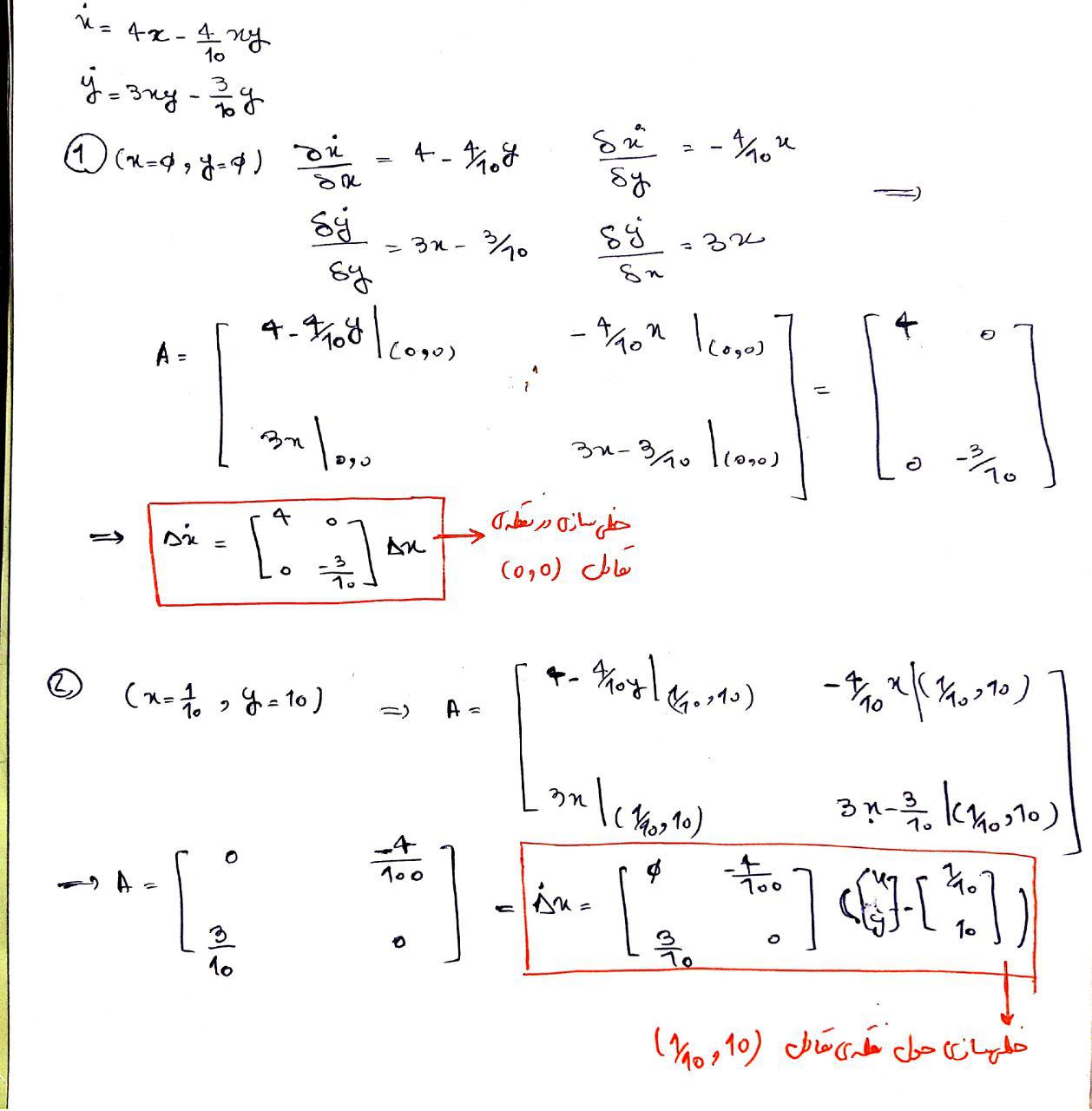
3xy – (3/10)y = 0 => 3xy = (3/10)y => 1 : y = 0, 2 : x = 1/10

در نتیجه دو نقطه ی تعادل برابر خواهند بود با :

x = 0, y = 0

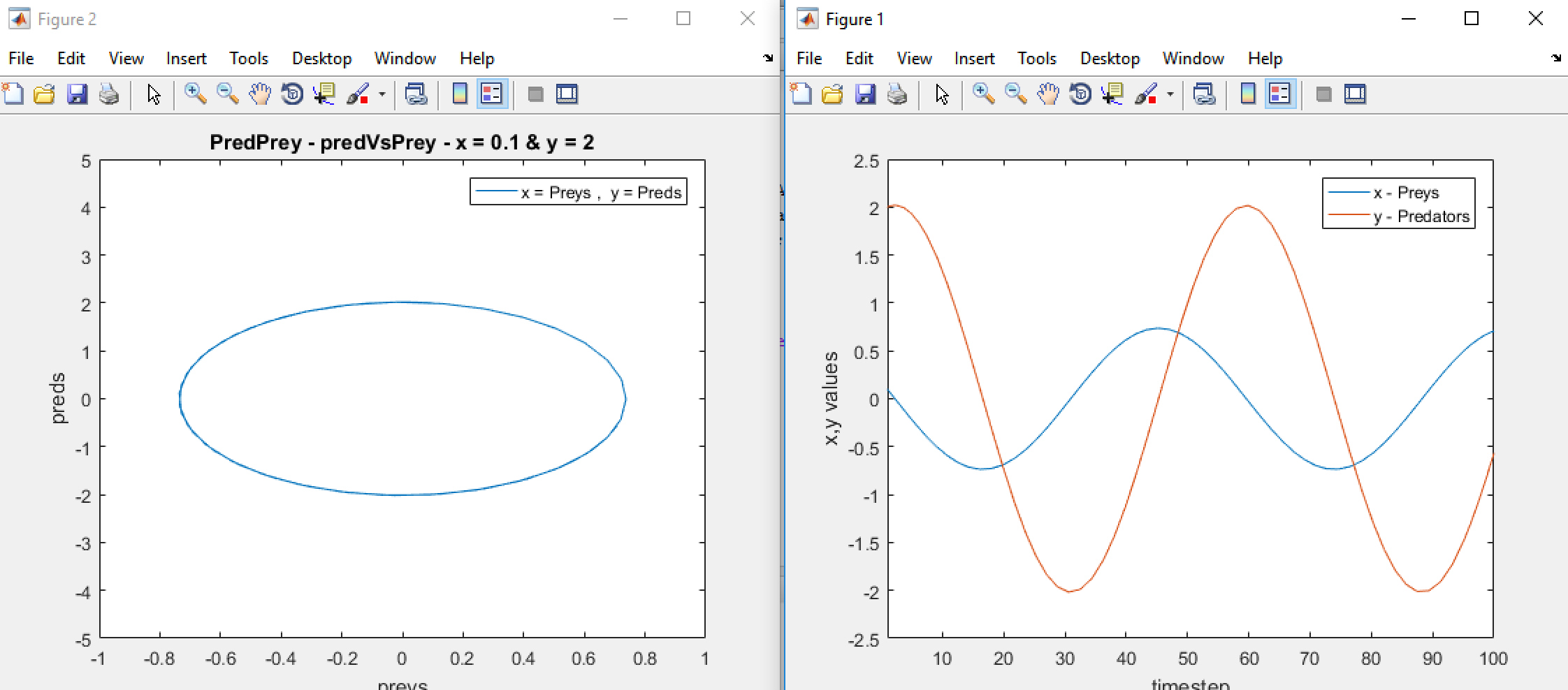
x = 1/10, y = 10

**-2 خطی سازی سیستم حول نقاط تعادل و بررسی پایداری**

****

**بدست آوردن فضای سیستم خطی شده و رسم پاسخ ها در متلب-3**

این بخش در فایل stateSpace.m پیاده سازی شده است. نمودار های خواسته شده در زیر قابل مشاهده است.



**سمت راست : شکار و شکارچی نسبت به زمان، سمت چپ شکار و شکارچی نسبت به هم**

**۴- بررسی سیستم در سیمولینک**

**۵- بدست آوردن تابع تبدیل و پاسخ پله و ضربه**

این بخش در فایل  **part5to11.m**  پیاده سازی شده است. تابع تبدیل فضای حالت به کمک دستورات زیر بدست آمد :

A = [0 -0.04; 0.3 0];

B = [1; 0];

C = [1 0];

D = 0;

SS=ss(A,B,C,D);

TFs=tf(SS);

TFs =

s

-----------

s^2 + 0.012

Continuous-time transfer function.

سپس پاسخ پله و پاسخ ضربه به کمک دستورات زیر بدست آمد :

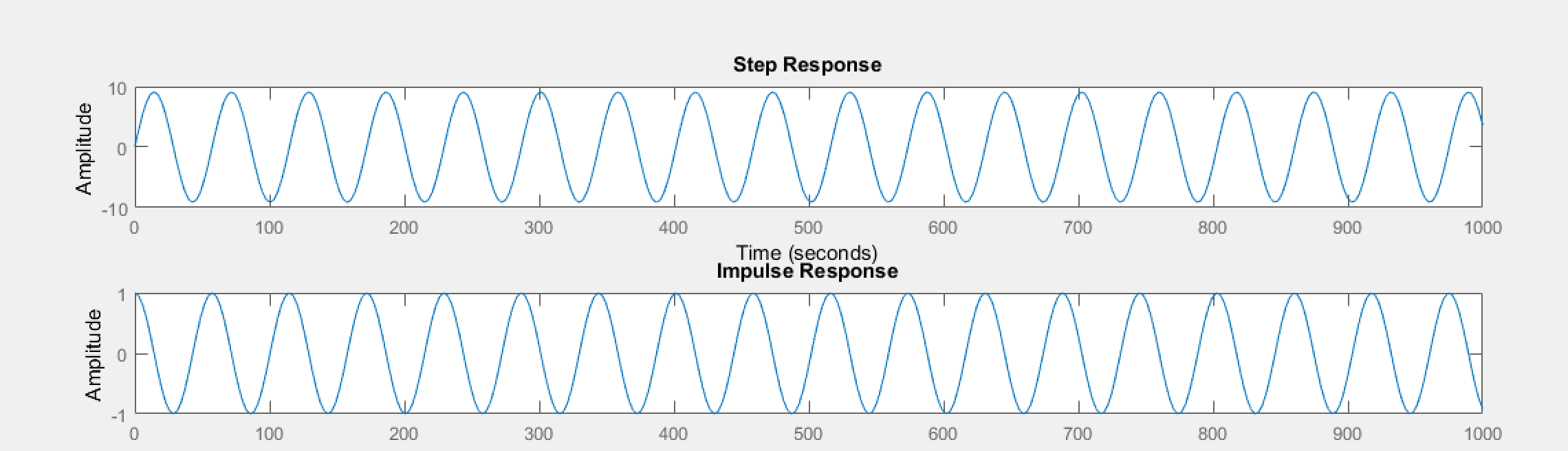
subplot(4,1,1)

step(TFs, 1000)

subplot(4,1,2)

impulse(TFs, 1000)

که نتایج بدست آمده در زیر قابل مشاهده است.



پاسخ پله و پاسخ ضربه ی تابع تبدیل

در ادامه به کمک دستورات زیر تابع تبدیل را به معادلات حالت تبدیل می کنیم.

[n,d]=ss2tf(A,B,C,D);

[Aprime, Bprime,Cprime, Dprime] = tf2ss(n, d);

معادله حالت بدست آمده به صورت زیر است :

tf2ss(n, d)

ans =

0 -0.0120

1.0000 0

که با معادله حالت ما متفاوت می باشد. علت این است که ...

**۷- ارتباط میان پاسخ ضربه سیستم و پاسخ به شرایط اولیه**

**۸- پاسخ پله تبدیل به همراه بهره ی ۰.۰۱ و فیدبک واحد**

پیاده سازی این بخش در فایل part5to11.m آورده شده است. به کمک دستورات زیر پاسخ پله سیستم مورد نظر را بدست می آوریم :

k = 0.01;

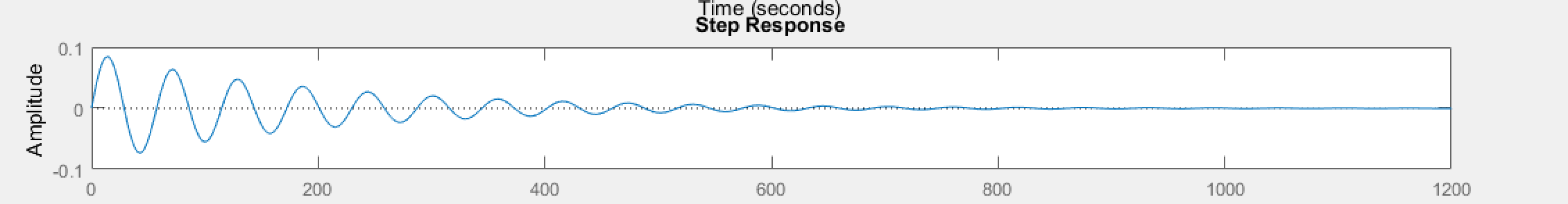
Tfeed = feedback(TFs\*k, 1);

subplot(4,1,3);

step(Tfeed);

S = stepinfo(Tfeed);

خروجی در زیر قابل مشاهده است.



**۹- شبیه سازی پاسخ شیب**

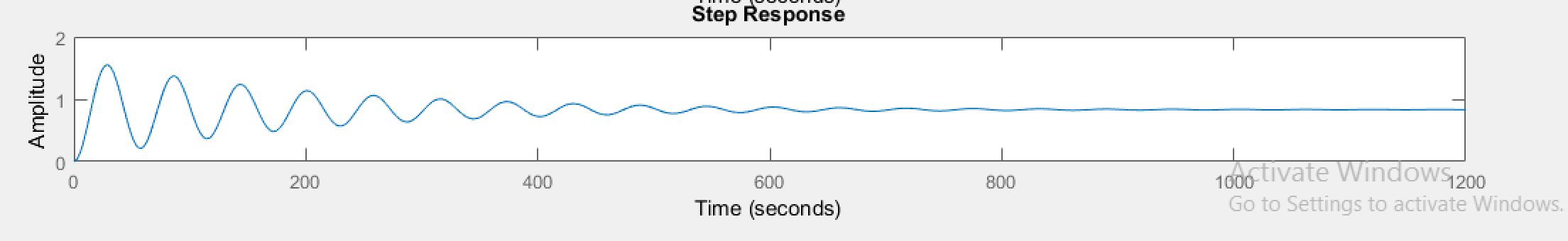
پیاده سازی این بخش در فایل part5to11.m آورده شده است. از آنجایی که در متلب دستوری برای بدست آوردن پاسخ شیب وجود ندارد، از دستور step و روش زیر استفاده می کنیم.

s = tf('s');

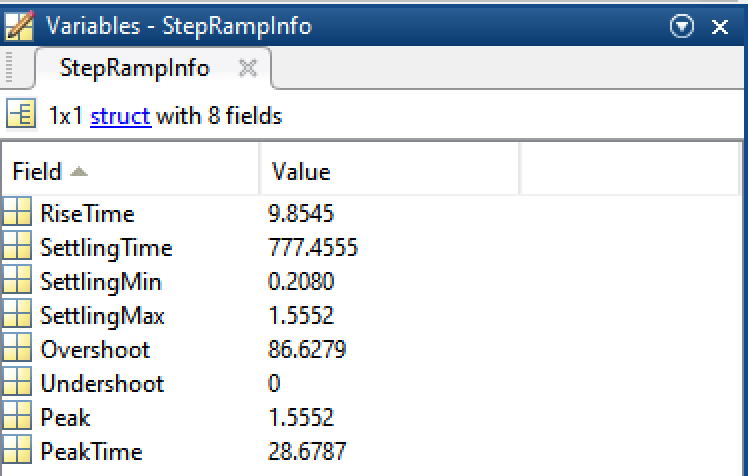
subplot(4,1,4), step(Tfeed / s); % Ramp response

StepRampInfo = stepinfo(Tfeed / s);

خروجی در زیر قابل مشاهده است.



همچنین اطلاعات این پاسخ در زیر آورده شده است.



همانطور که دیده می شود فراجهش برابر با 86.6279 و settling Time برابر با 777.4555 می باشد.

**۱۰- پاسخ سیستم حلقه بسته بالا به چند ورودی سینوسی با فرکانس و دامنه دلخواه**

پیاده سازی این بخش در فایل part5to11.m قرار داده شده است. چهار ورودی سینوسی زیر انتخاب شده است :

u = sin(130\*t);

u = 10 \* sin(10\*t);

u = 34 \* sin(0.1\*t);

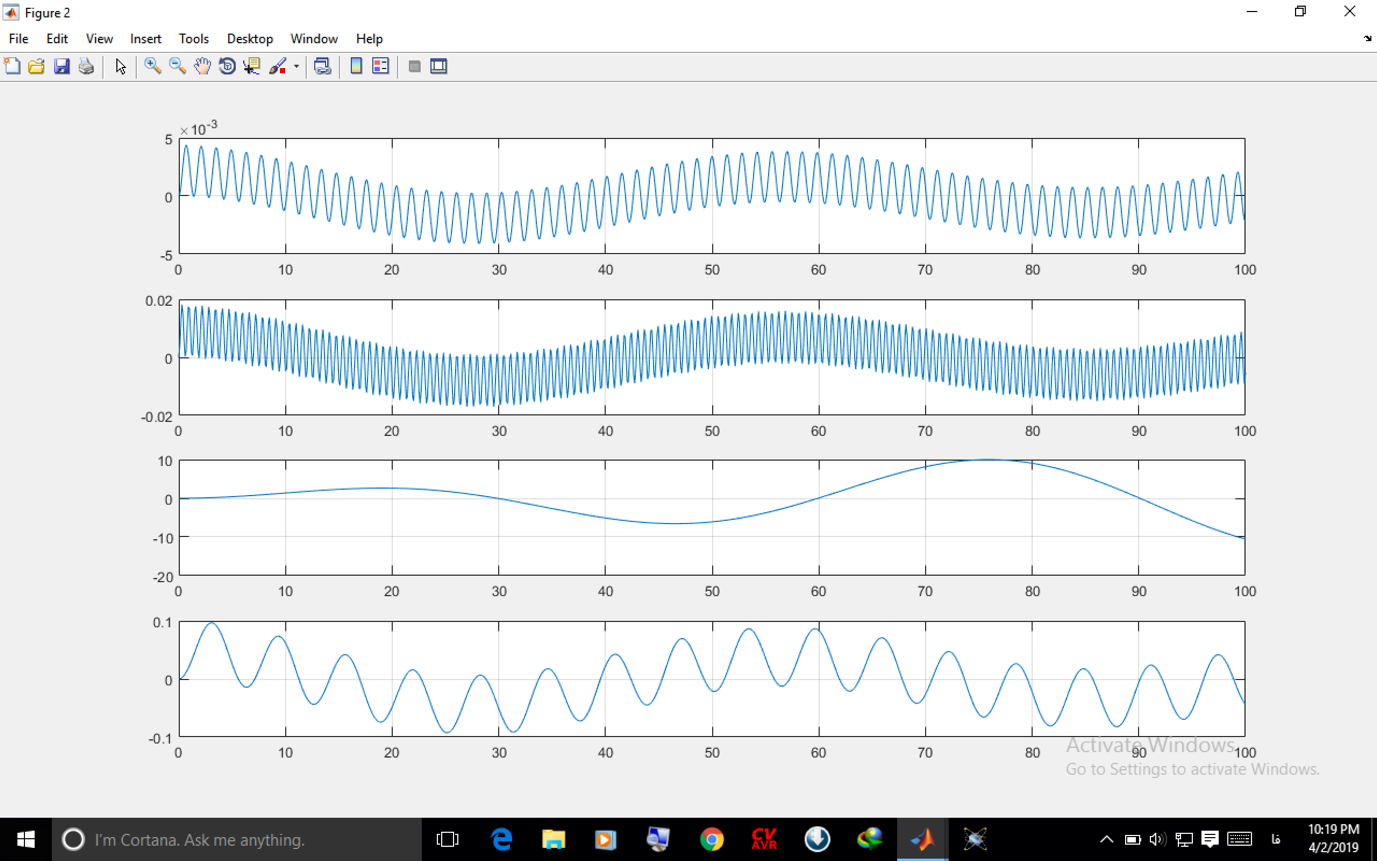
u = 5 \* sin(t);

به کمک دستور زیر پاسخ سیستم به هر یک از این ورودی ها بدست آمد :

y = lsim(Tfeed, u, t);

plot(t, y)

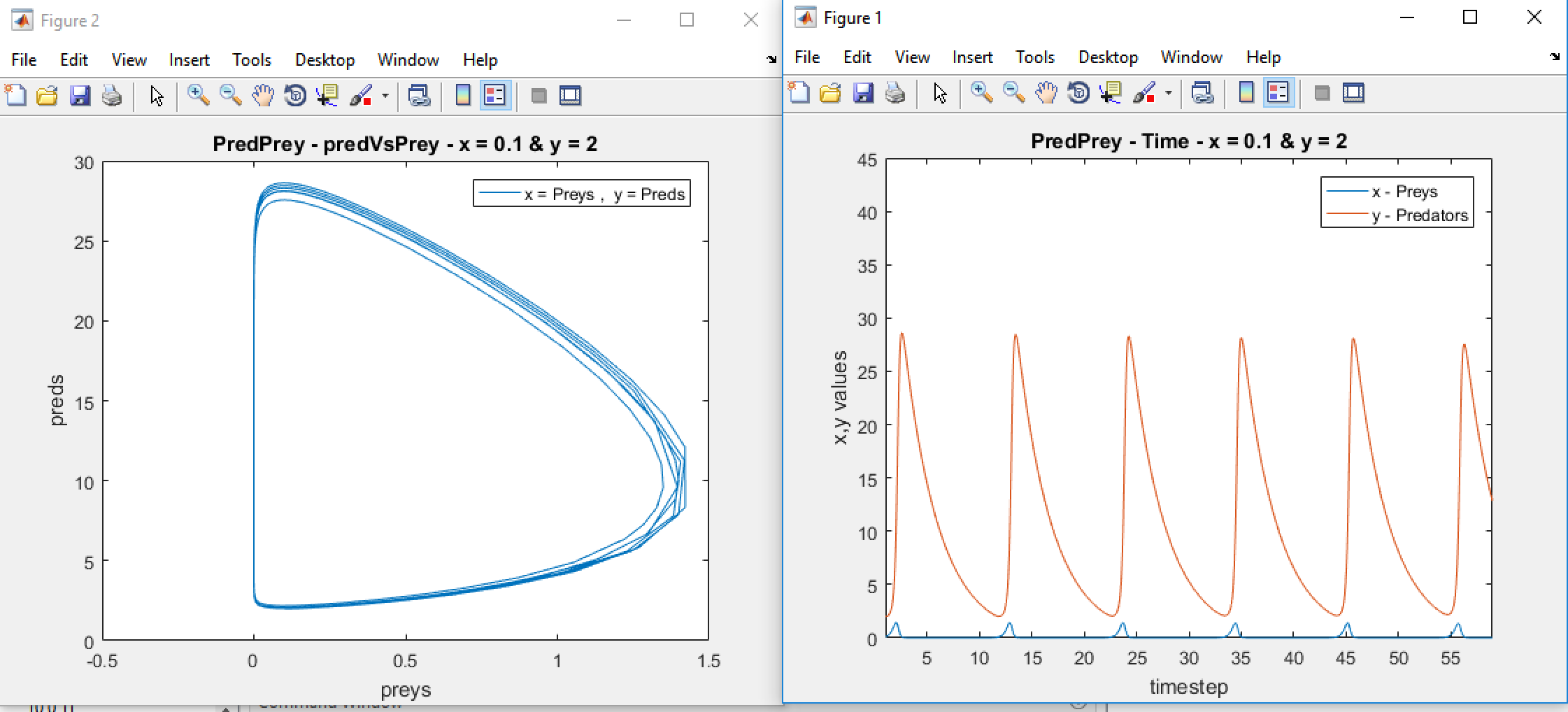
نتایج بدست آمده برای هر یک از ورودی های بالا به ترتیب در زیر قابل مشاهده است.



**۱۱- تابع تبدیل جدید**

**۱۲- شبیه سازی سیستم غیر خطی**

پیاده سازی این بخش در فایل predPrey\_nonlinear.m قرار داده شده است. خواسته های سوال ۳ برای این سیستم بدست آورده شده است که نتایج آن در زیر قابل مشاهده است.



**سمت راست : شکار و شکارچی نسبت به زمان، سمت چپ شکار و شکارچی نسبت به هم**