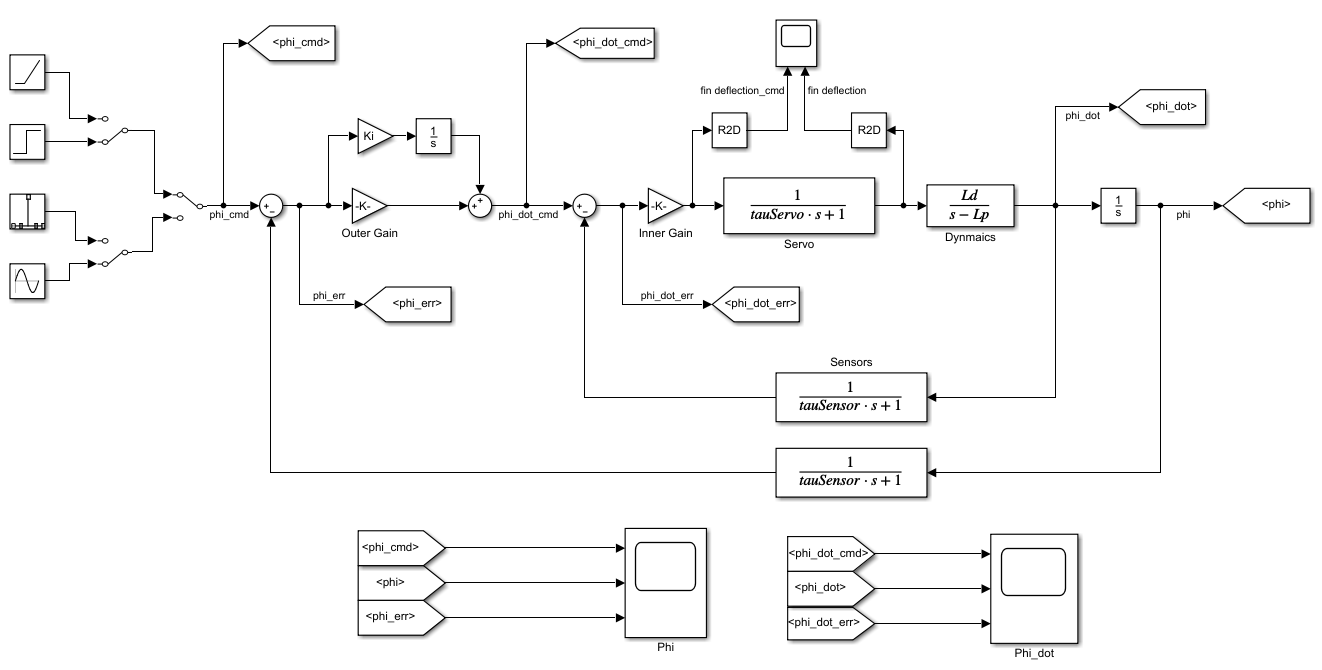
**Roll Control:**

Visual Representation of the entire control algorithm:

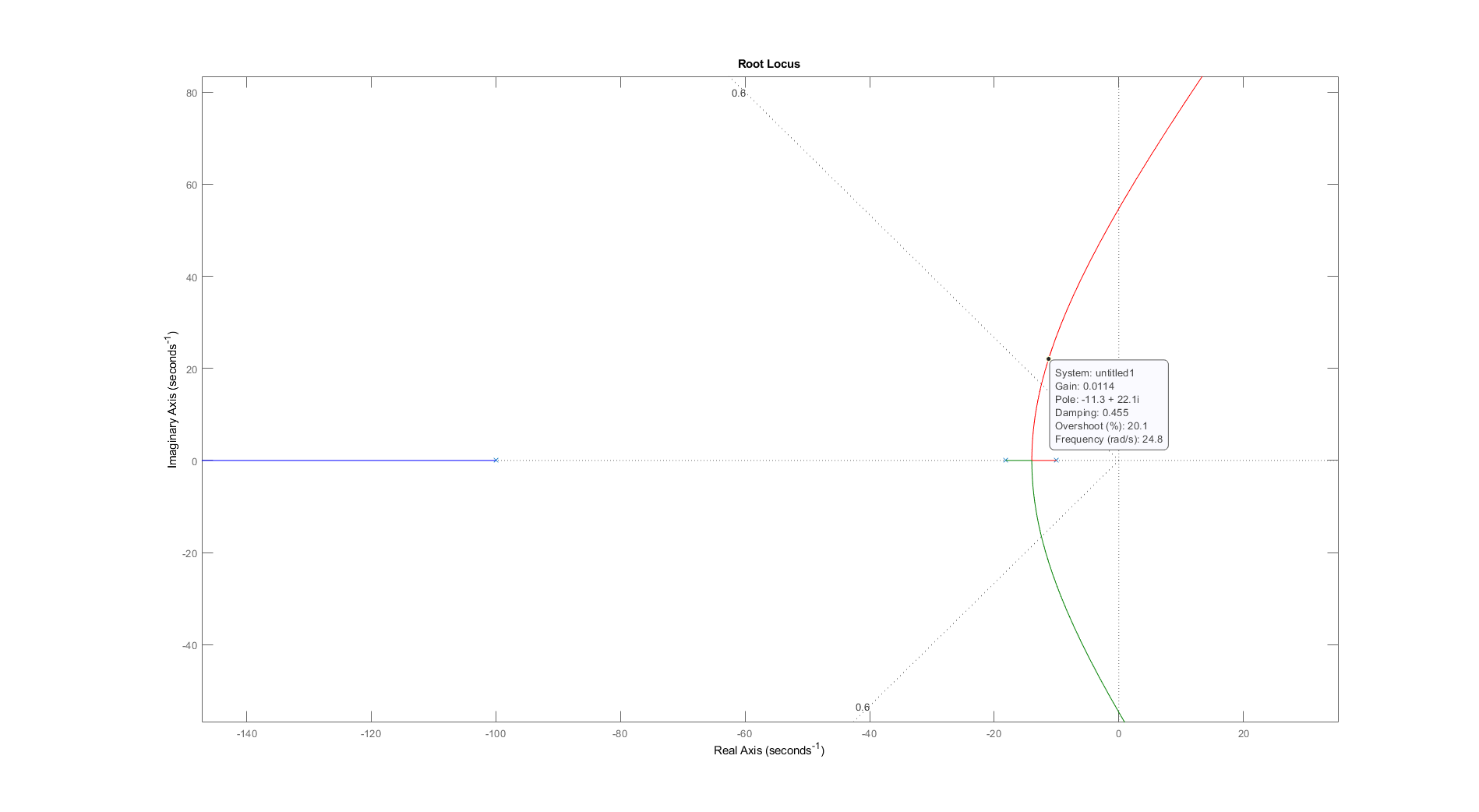


Time constants:

Flight conditions:

Other Parameters

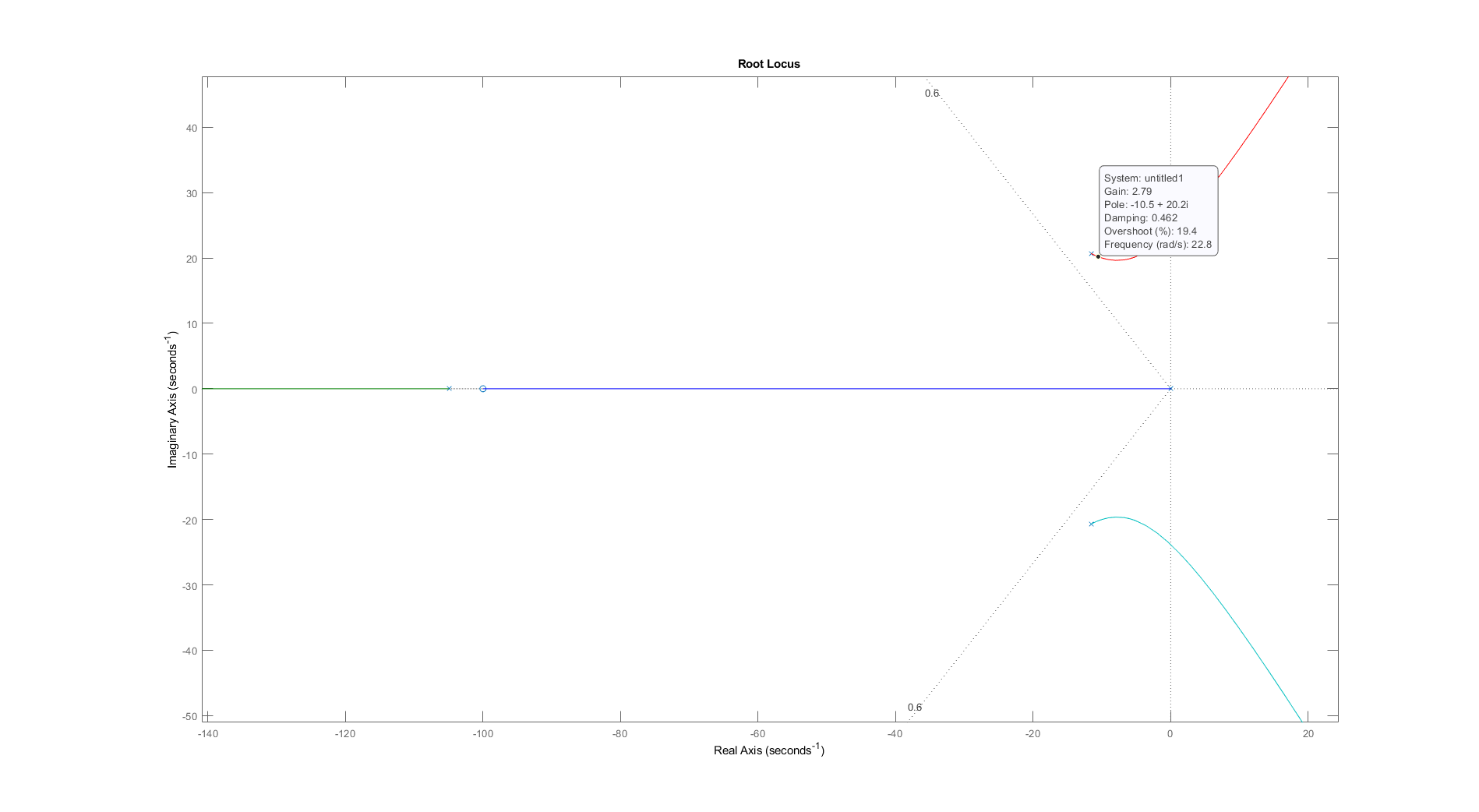
**Inner loop design:**

Root locus:  


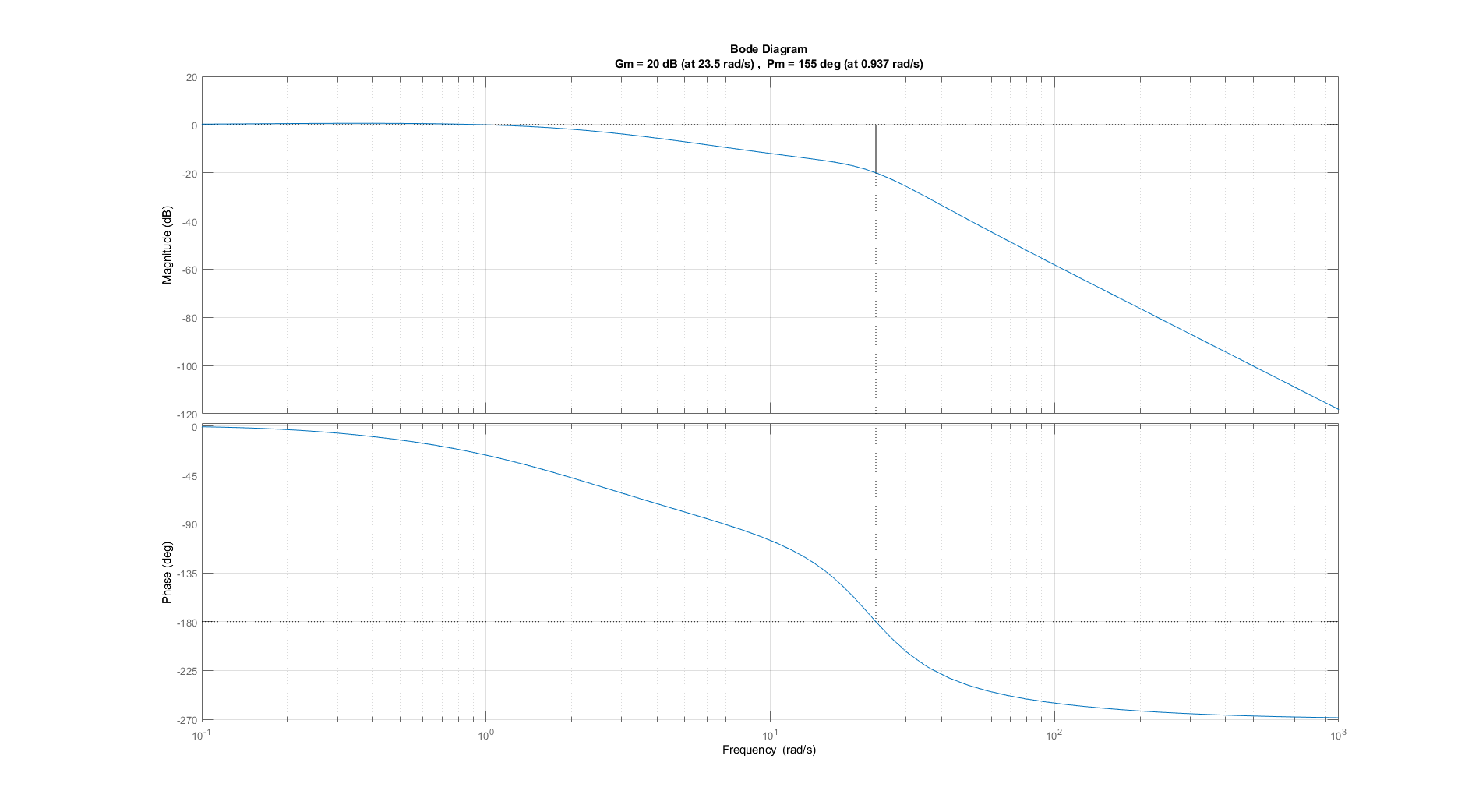
Choose an inner loop gain:

**Outer Loop Design:**

Root locus

  
Choose an outer loop gain:   
Add integral control for improved tracking of ramp input:

**Gain and Phase Plots of Outer Loop:**



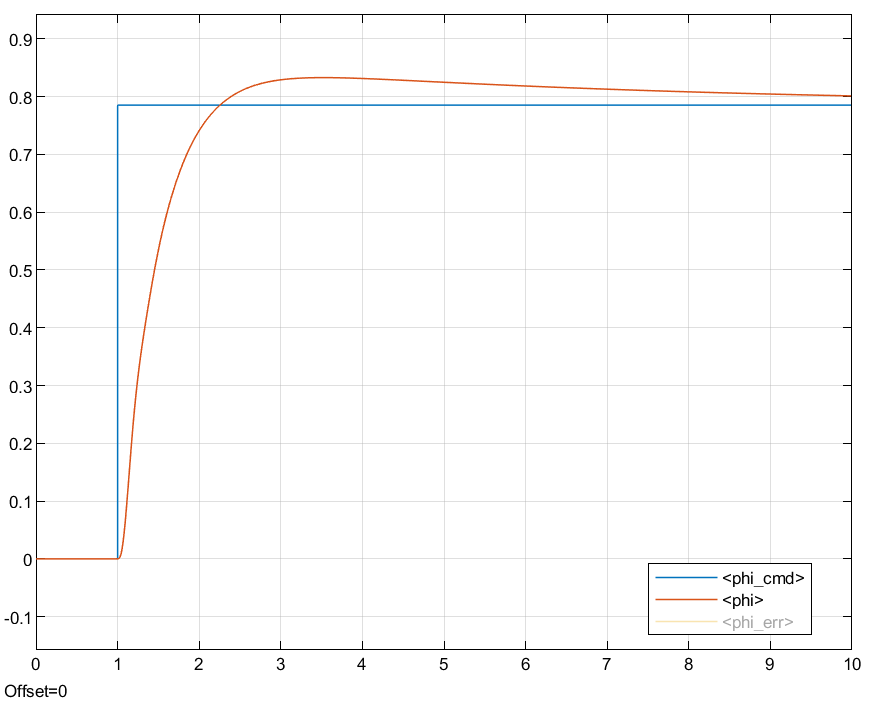
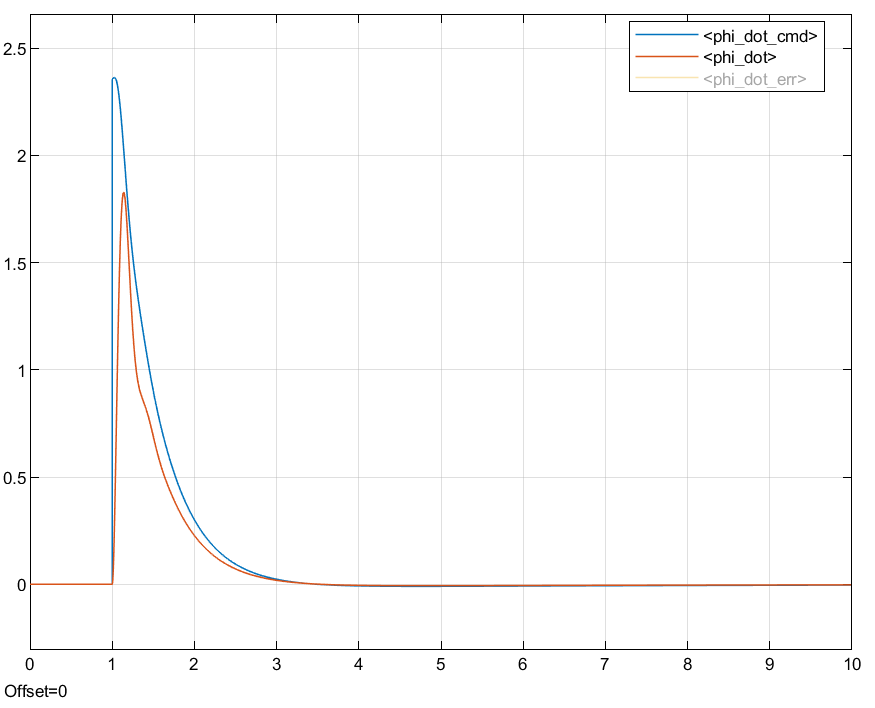
Gain margin: 20dB

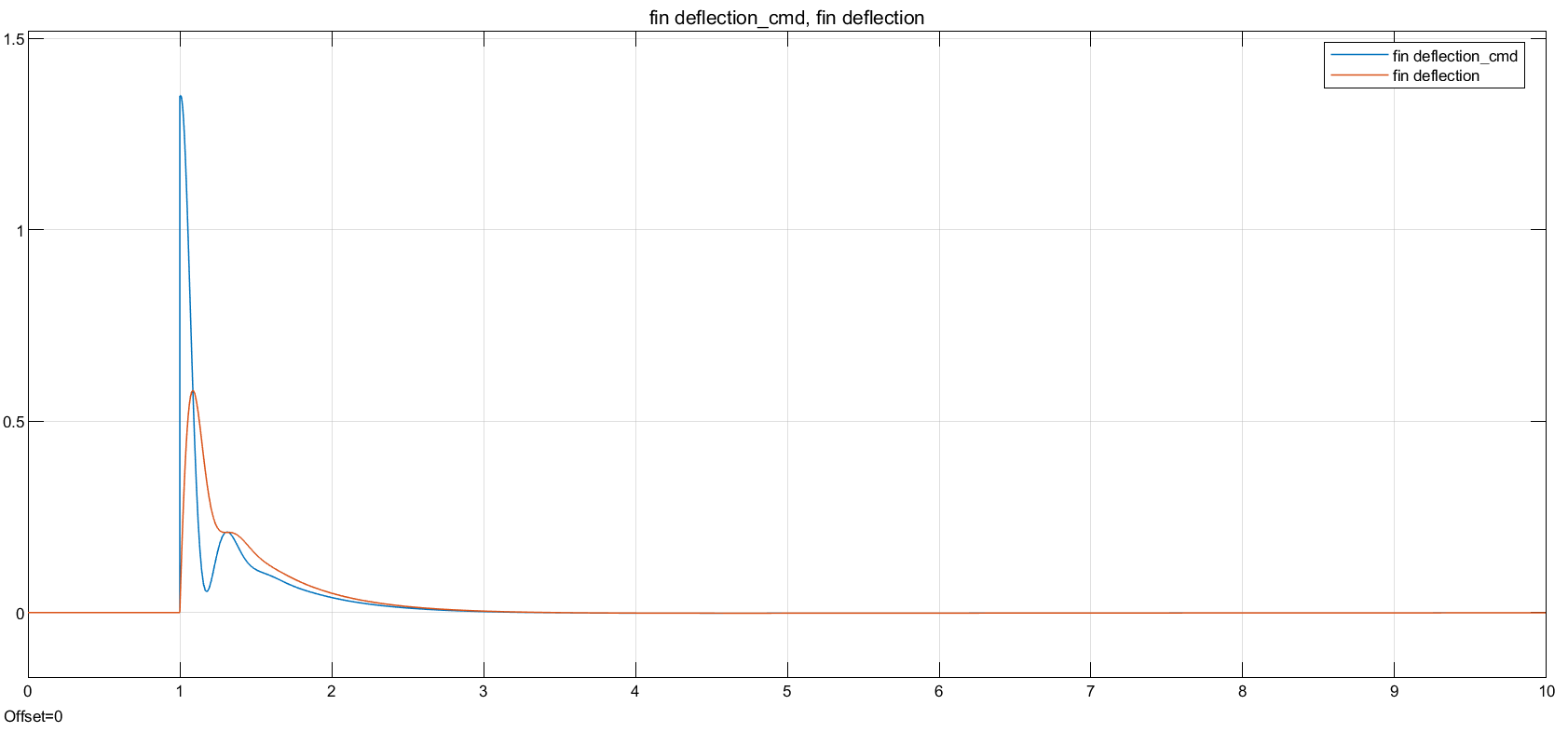
Phase margin: 155deg

Bandwidth (-3dB): 2.53 rad/s

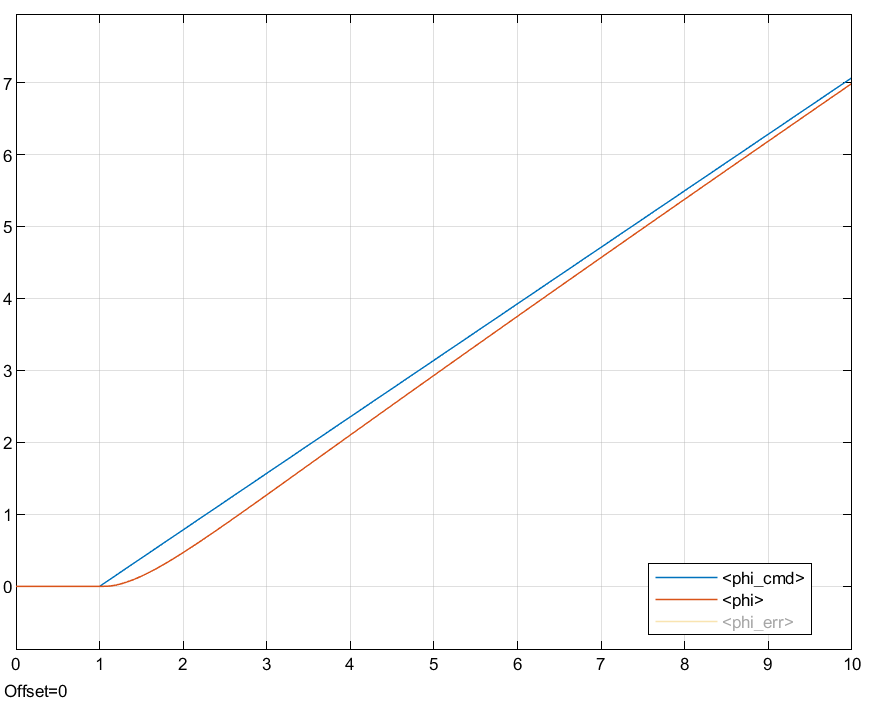
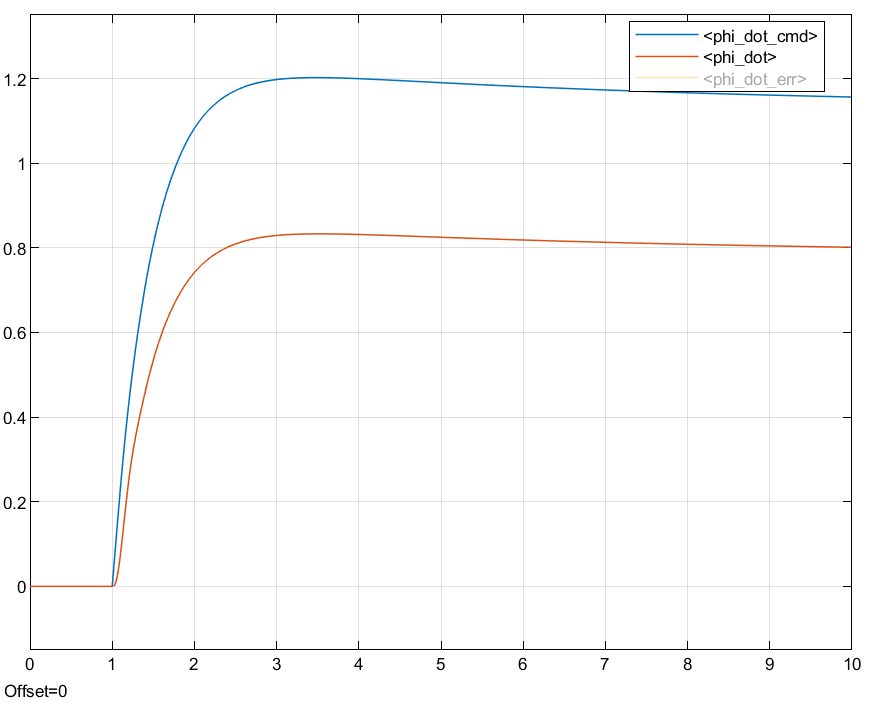
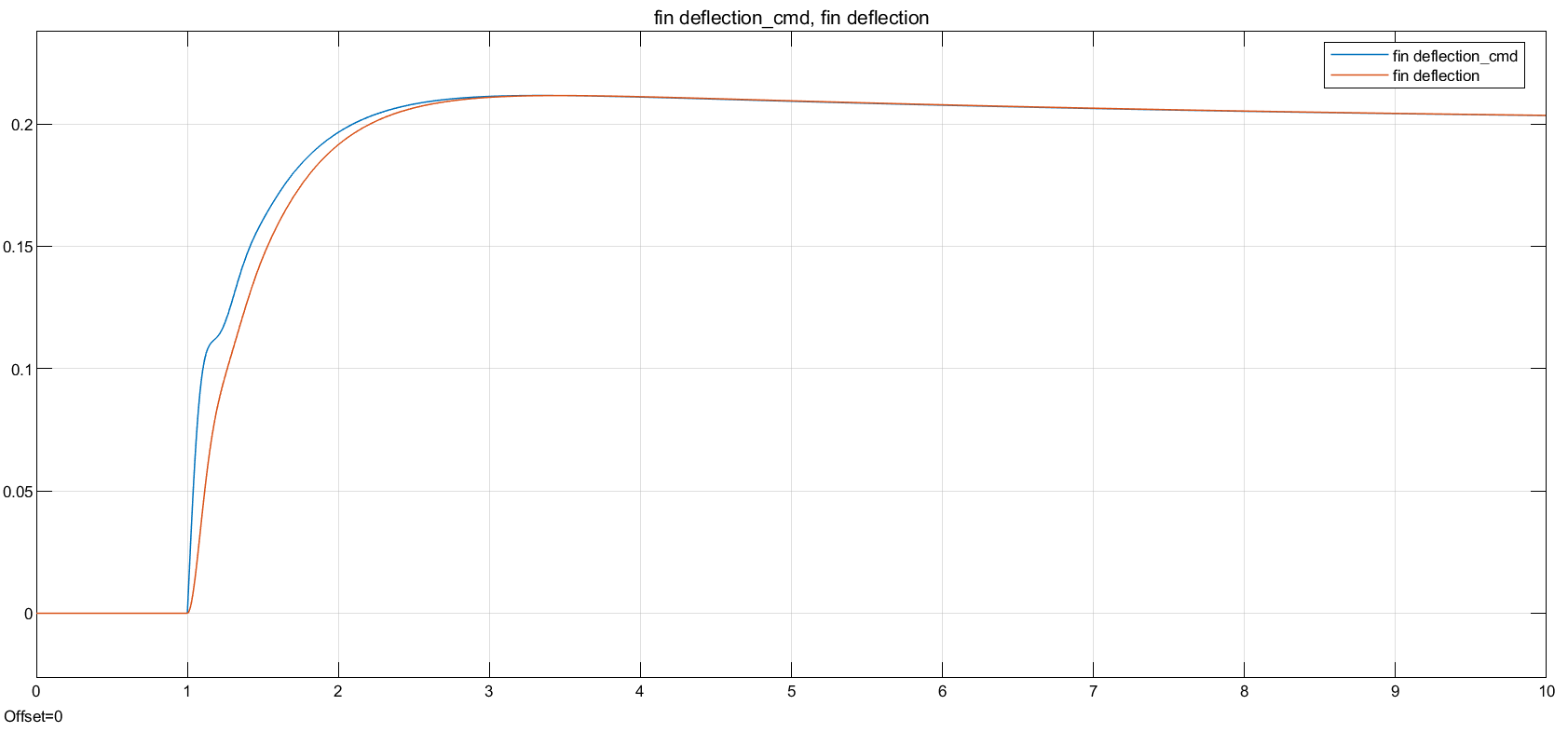
**Simulation:**

**Test 1:** Step input of 45 deg

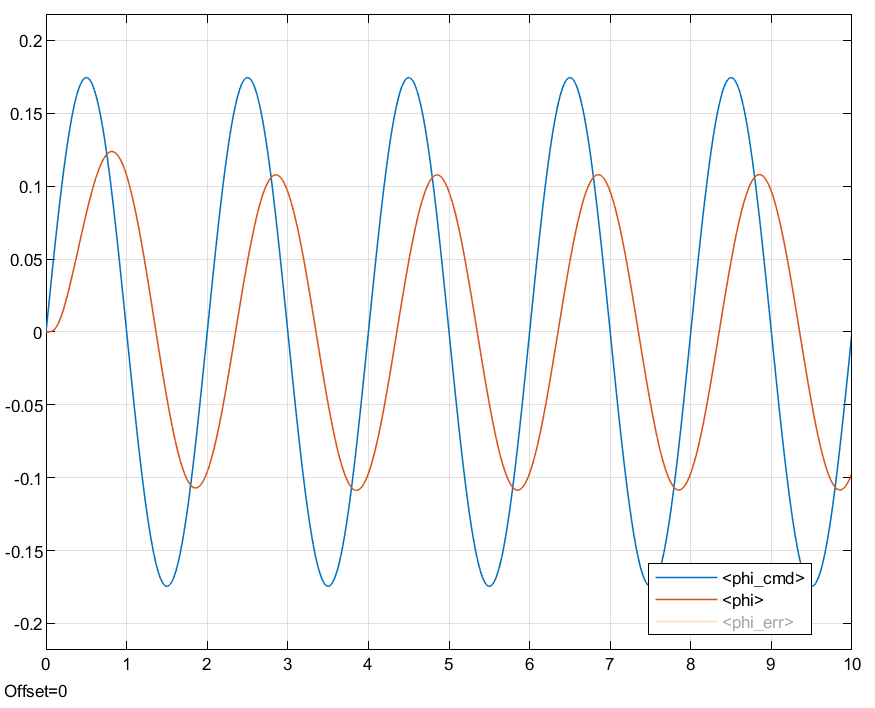
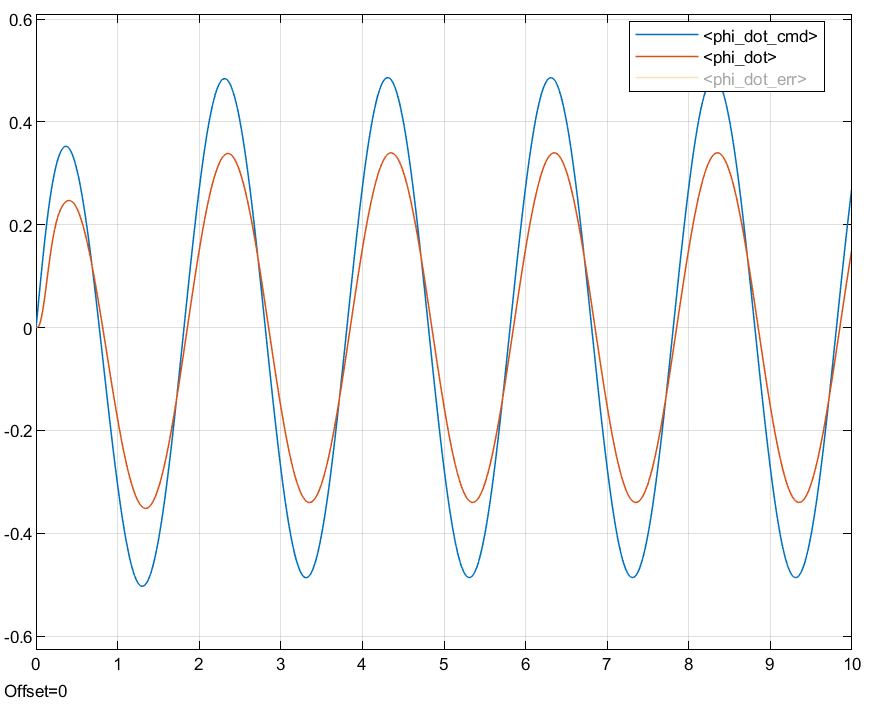
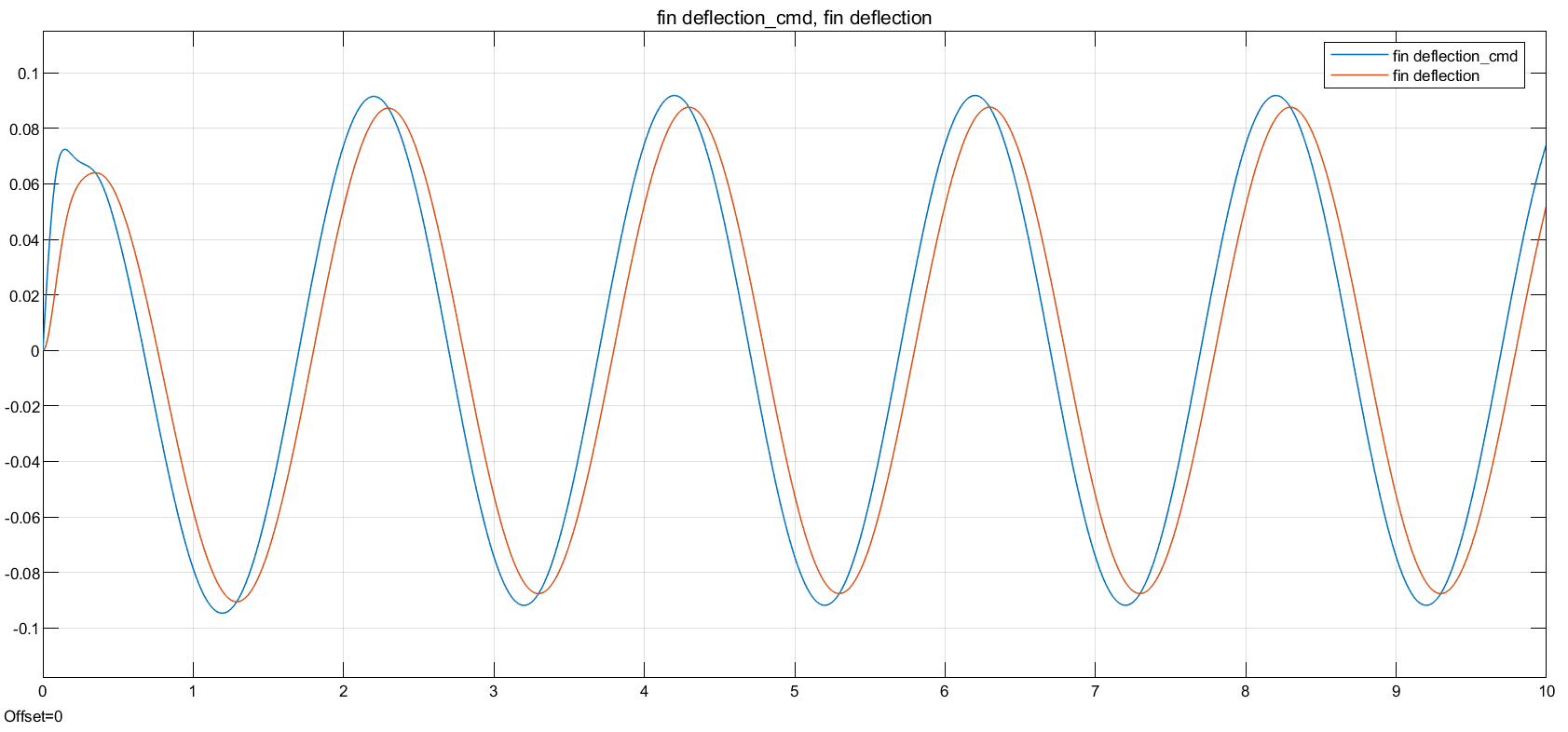
 



**Test 2:** Ramp input of 45 deg/s

**Test 3:** Sine input (amplitude: 45 deg, Frequency: Pi rad/s)

Matlab Code:

clear,clc

%% Parameteres

d = 0.08;

S = pi\*d^2/4;

U0 = 228.4; % ~~ 0.67Ma

Ix = 6.331e-3;

rho = 1.225;

TechnionConfig = 1;

if (TechnionConfig)

Clp = -51;

Cld = 4.03;

else

load('aero.mat','Clp','Cl\_delta\_l');

Clp = Clp(6,9,9);

Cld = -Cl\_delta\_l(6,9,9);

end

Lp =rho\*S\*U0\*d^2\*Clp/(4\*Ix);

Ld =rho\*S\*U0^2\*d\*Cld/(4\*Ix);

tauServo = 0.1;

tauSensor = 0.01;

%% Inner Loop Design

dyn = tf(Ld,[1 -Lp]);

servo = tf(1,[tauServo 1]);

sensor = tf(1,[tauSensor 1]);

rlocus(dyn\*servo\*sensor)

sgrid(0.6,1e10)

Kinner=0.01;

%% Outer Loop Design

innerLoop = feedback((Kinner\*dyn\*servo),sensor);

int = tf(1,[1 0]);

rlocus(innerLoop\*int)

sgrid(0.6,1e10)

Kouter = 3;

Ki = 0.5;

s = tf('s');

outerLoop = feedback((Kouter+Ki/s)\*innerLoop\*int,sensor);

%% Stability margins

margin(outerLoop)

systemBandwidth = bandwidth(outerLoop)

grid

sim('roll\_control\_loop');