PROJECTING HYDRAULIC SYSTEM

Teodora Spasojević

2019/0427

Mentor : professor Kosta Jovanović

1. ANALYSIS OF HYDRAULIC AND MECHANICAL SYSTEM

SCHEME OF MECHANICAL SYSTEM

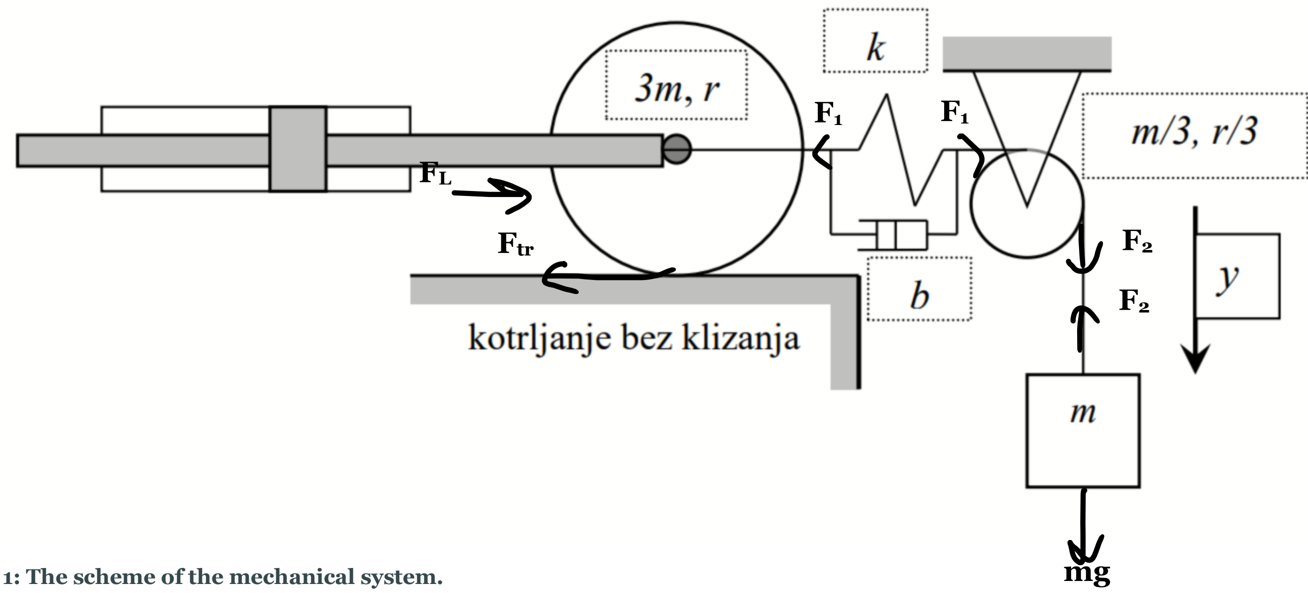


Figure 1: The scheme of the mechanical system.

PARAMETERS HYDRAULIC SYSTEM

Constant	Value	Unit
K	0	$\frac{N}{m}$
Bp	30	$\frac{N}{m/s}$
Mt	11	kg
Ap	$1.4\cdot10^{-2}$	m^2
Vt	$4\cdot10^{-3}$	m^3

Kc	$1.8 \cdot 10^{-12}$	$\frac{m^5}{Ns}$
Ctp	$3 \cdot 10^{-13}$	$\frac{m^5}{Ns}$
Kq	1	$\frac{m^2}{s}$
Kt	$6.8 \cdot 10^{-2}$	$\frac{Nm}{A}$
Kf	$2.5 \cdot 10^{3}$	$\frac{N}{m}$
r	0.02	m
β_e	$13 \cdot 10^{8}$	$\frac{N}{m^2}$
R	10	Ω
U_{max}	24	V

PARAMETERS OF THE MECHANICAL SYSTEM

Constant	Value	Unit
L	1,2	m
T	1,5	s
T_{acc}	0,3	s
m	180	kg
r	0,3	m
k	$2 \cdot 10^{6}$	$\frac{N}{m}$
b	150	$\frac{N}{m/s}$

ACTUATOR:

$A_p P_L = M_t \ddot{x}_p + B_p \dot{x}_p + K x_p + F_L$

$Q_L = A_p \dot{x}_p + C_{tp} P_L + \frac{Vt}{4\beta e} \dot{P}_L$

$Q_L = K_q x_v - K_c P_L$

$F_i = K_f x_v$

$K_t i = T_L$

$U = Ri$

веза : $F_i r = T_L$

LOAD:

T1: $3m\ddot{x}_p = F_L - F_1 - F_{tr}$

P1: $\frac{1}{2} 3mr^2 \frac{\ddot{x}_p}{r} = rF_{tr}$

T2: /

P2: $\frac{1}{2} \frac{m}{3} \frac{r^2}{9} \frac{\ddot{y}}{r/3} = \frac{r}{3} (F_1 + F_2)$

T3: $m\ddot{y} = mg - F_2$

P3: /

$F_1 = k(x_p - y) + b(\dot{x_p} - \dot{y})$

$z = (z_1, z_2, z_3, z_4, z_5) = (x_p, \dot{x}_p, P_L, y, \dot{y})$, $u=U$, $y=x_p$

$\dot{z}_1 = \dot{x_p} = z_2$

$$\dot{z}_2 = \ddot{x}_p = -\frac{2(k+K)}{2M_t+9m} z_1 - \frac{2(B_p+b)}{2M_t+9m} z_2 + \frac{2A_p}{2M_t+9m} z_3 + \frac{2k}{2M_t+9m} z_4 + \frac{2b}{2M_t+9m} z_5$$

$$\dot{z}_3 = \dot{P_L} = -A_p \frac{4\beta_e}{V_t} z_2 - \left(K_c + C_{tp}\right) \frac{4\beta_e}{V_t} z_3 + K_q \frac{4\beta_e}{V_t} \frac{K_t}{rRK_f} u$$

$$\dot{z}_4 = \dot{y} = z_5$$

$$\dot{z}_5 = \ddot{y} = \frac{6k}{7m} z_1 + \frac{6b}{7m} z_2 + \frac{6k}{7m} z_3 - \frac{6k}{7m} z_4 - \frac{6k}{7m} z_5 + \frac{6}{7} g$$

EQUATIONS OF SYSTEMS IN STATE SPACE IN A MATRIX FORM

$$\dot{z} = \begin{bmatrix} \dot{x_p} \\ \ddot{x_p} \\ \dot{P_L} \\ \dot{x_2} \\ \ddot{x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -\dfrac{2(k+K)}{2M_t+9m} & -2\dfrac{B_p+b}{2M_t+9m} & \dfrac{2A_p}{2M_t+9m} & \dfrac{2k}{2M_t+9m} & \dfrac{2b}{2M_t+9m} \\ 0 & -A_p \dfrac{4\beta_e}{V_t} & -\left(K_c + C_{tp}\right)\dfrac{4\beta_e}{V_t} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ \dfrac{6k}{7m} & \dfrac{6b}{7m} & 0 & -\dfrac{6k}{7m} & -\dfrac{6b}{7m} \end{bmatrix} \begin{bmatrix} x_p \\ \dot{x_p} \\ P_L \\ y \\ \dot{y} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ K_q \dfrac{4\beta_e}{V_t} \dfrac{K_t}{rRK_f} \\ 0 \\ 0 \end{bmatrix} u$$

$$+ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \dfrac{6g}{7} \end{bmatrix}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_p \\ \dot{x_p} \\ P_L \\ y \\ \dot{y} \end{bmatrix}$$

Transfer function: $G(s) = C(sI - A)^{-1}B$

1. REALIZATION IN MATLAB

INITIALIZATION OF THE SYSTEM AND CREATION OF MODELS IN STATE SPACE

CODE:

```
%% Project

clear;
close all;
clc;

%% Parameter initialization

Bp = 30;
K = 0;
Ap = 1.4*(10^(-2));
Mt = 11;
Vt = 4*(10^(-3));
Kc = 1.8*(10^(-12));
Ctp = 3*(10^(-13));
Kq = 1;
Kt = 6.8*(10^(-2));
Kf = 2.5*(10^(3));
R = 10;
r = 0.02;
Be = 13*(10^(8));
Umax = 24;

m = 180;
k = 2*(10^(6));
b = 150;
L = 1.2;
T = 1.5;
r1 = 0.3;

a_max= 10/3;
v_max = a_max*0.2*T;

g = 9.81;
```

```
%% matrix A - state/system matrix

a21 = -2*(k+K)/(2*Mt+9*m);
a22 = -2*(b+Bp)/(2*Mt+9*m);
a23 = 2*Ap/(2*Mt+9*m);
a24 = 2*k/(2*Mt+9*m);
a25 = 2*b/(2*Mt+9*m);

a32 = -Ap*4*Be/Vt;
a33 = -4*Be*(Kc+Ctp)/Vt;

a51 = k*6/(7*m);
a52 = b*6/(7*m);
a54 = -k*6/(7*m);
a55 = -b*6/(7*m);

A = [0 1 0 0 0; a21 a22 a23 a24 a25; 0 a32 a33 0 0; 0 0 0 0 1; a51
a52 0 a54 a55];

%% matrix B - input matrix

b31 = Kq*Kt*4*Be/(Vt*R*r*Kf);

B = [0 0 b31 0 0]';

%% matrix C - output matrix

C = [0 0 0 1 0];

%% matrix H

h51 = (6/7)*g;

H = [0 0 0 0 h51]';

%% transfer function

s = tf('s');

G = C*(s*eye(5)-A)^(-1)*B;

%% drawing zeros and poles and step response
```

```
figure(1)
    pzmap(G);

figure(2)
    step(G);
    title('Step response of the system in open loop');
    grid on;

figure(3)
    step(G/(1+G));
    title('Step response of the system in closed loop');
    grid on;

%% running the simulation

sim('projekat_56.slx');
```

FIND K_{KR} AND T_{KR}

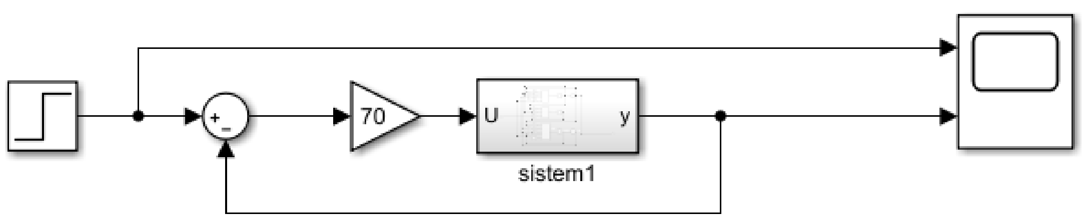


Figure 2: The scheme of the system we use to find Tkr and Kkr.

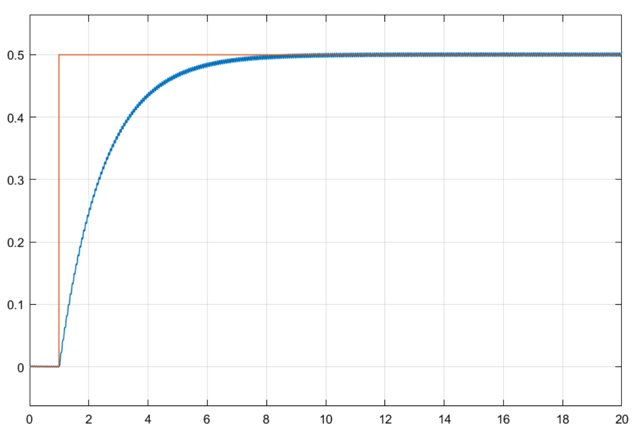


Figure 3: Searching for a suitable Kkr when we bring a step function to the input of the system.

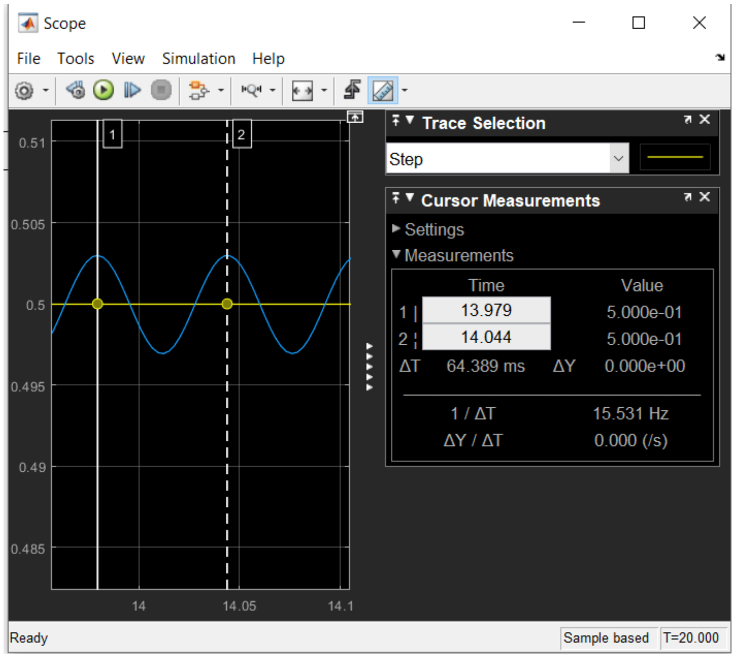


Figure 4: Determination of Kkr and Tkr.

$K_{kr} = 70$

$T_{kr} = 0.0644s$

SCHEME OF THE WHOLE SYSTEM

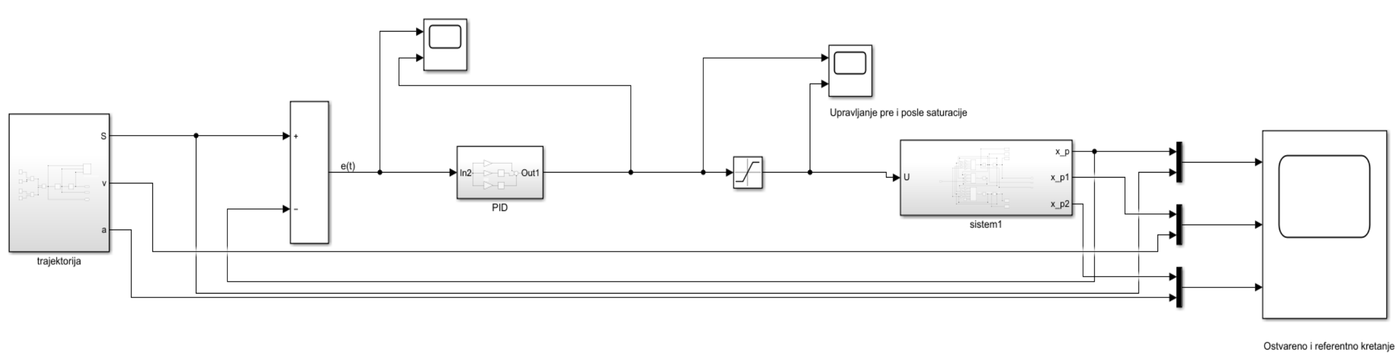


Figure 5: Realization of the entire system in Simulink.

SCHEMATIC OF THE CONTROL-GENERATING SUBSYSTEM

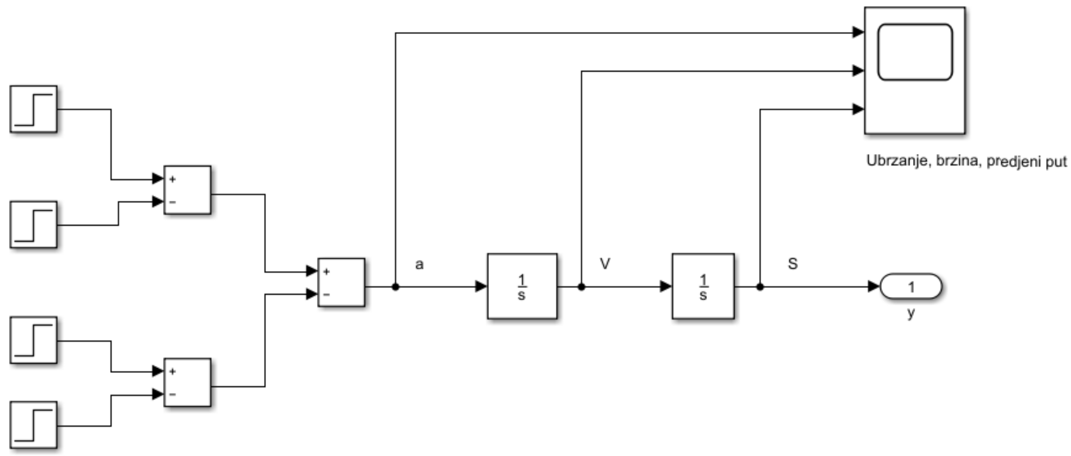


Figure 6: Control Generation Subsystem.

PID REGULATOR SCHEME

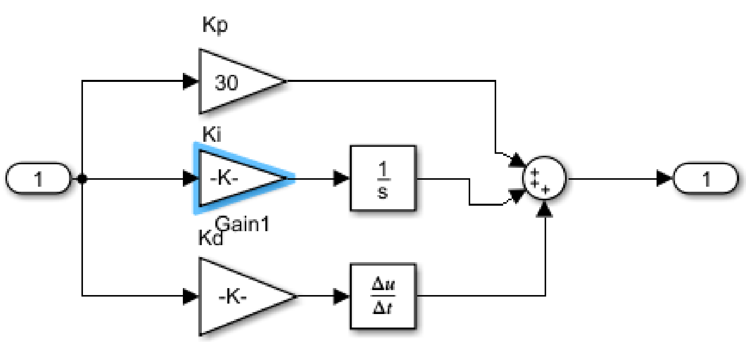


Figure 7: PID regulator scheme.

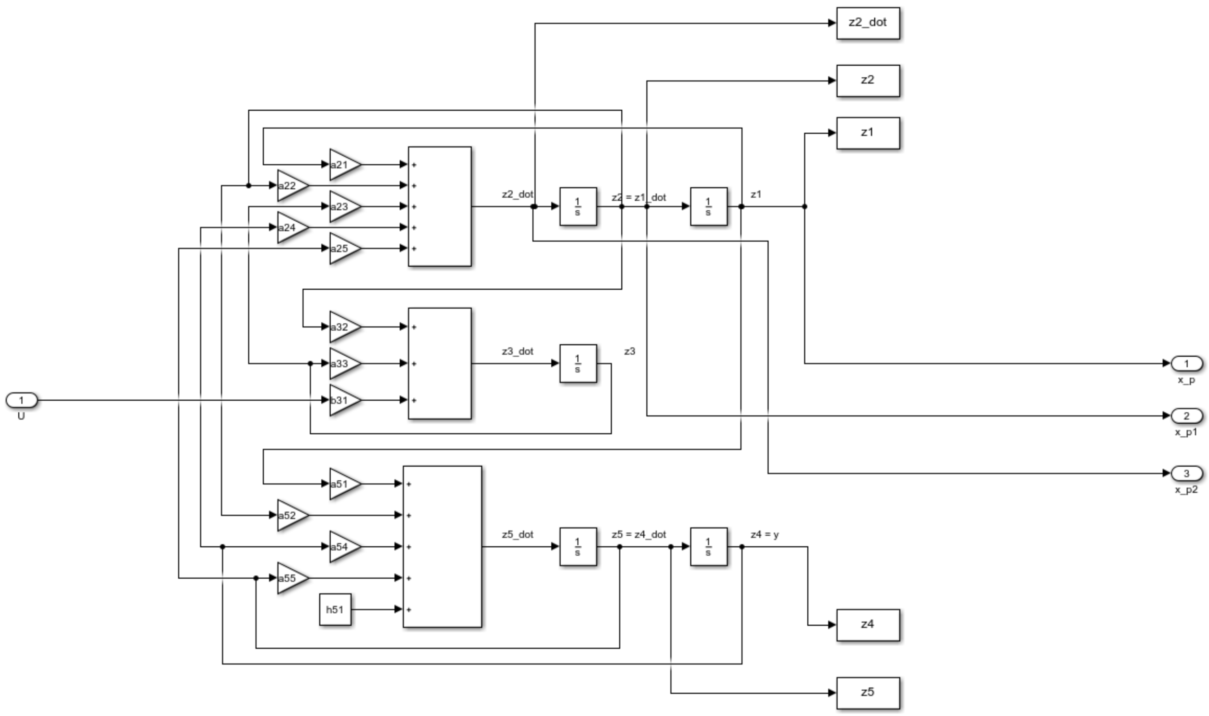


Figure 8: Realization of the load model and actuator.

If we take the movement behind the spring and the silencer in the load for the output of the system:

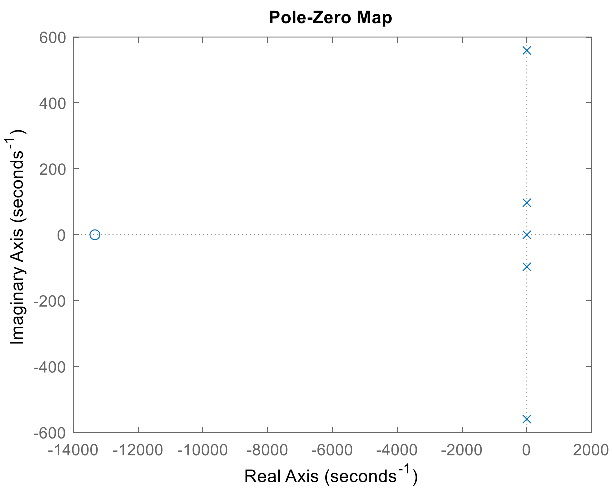


Figure 9: Graph of zeros and poles of the system transfer function.

The final zeros affect the acceleration of the system. The closer they are to the imaginary axis, the more they accelerate the system. Zero in our system, due to the distance from the imaginary axis, has no significant effect on the speed of the system.

On the other hand, the poles slow down the system and the closer they are to the imaginary axis, the more they slow down the system.

From there, we conclude that our system will be slow.

We can also note that the system has one pole located in the coordinate beginning. This means that the system has first order astatism.

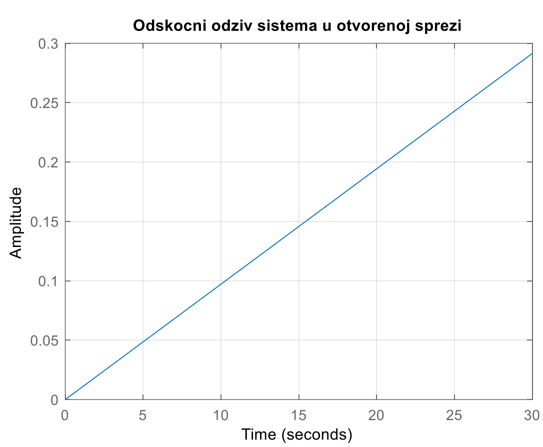


Figure 10: System step response in an open loop.

Fourier's transformation of the step signal is 1/s and knowing that our system has first order astatism, in the step response, dominant will be the pole in the coordinate beginning of the second order. Therefor we expect the step response of the system in an open loop to be ramp shaped.

Figure 11: Step response of the system in closed loop.

If for the output of our system, we take the movement of the piston rod.

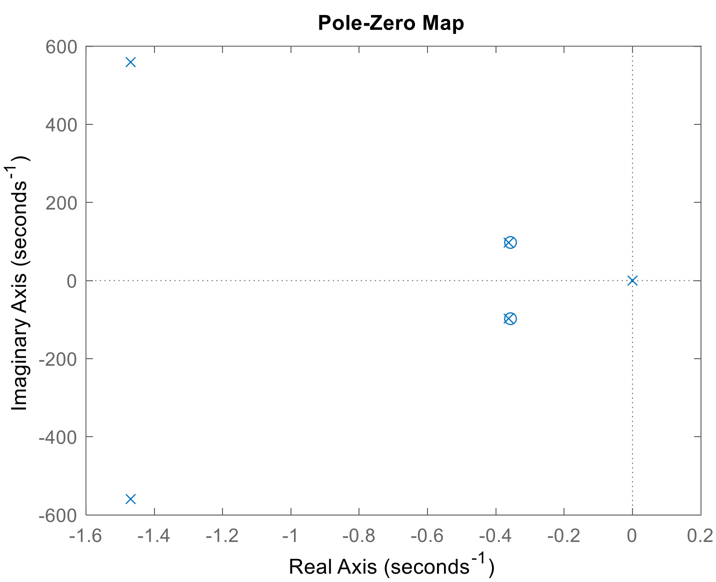


Figure 12: Zeros and poles of the system transfer function.

We have a pair of poles and zeros that make up a dipole and they have no effect on the speed of the system. In addition to them we have two poles, but again they are not far from the imaginary axis, so they slow down the system. We also have first order astatism again.

Since we have first order astatism, the error in the stationary state will be equal to zero in response, if the step function is brought to the input, we will not have the overshot either.

Figure 13: Step response of the system in open loop.

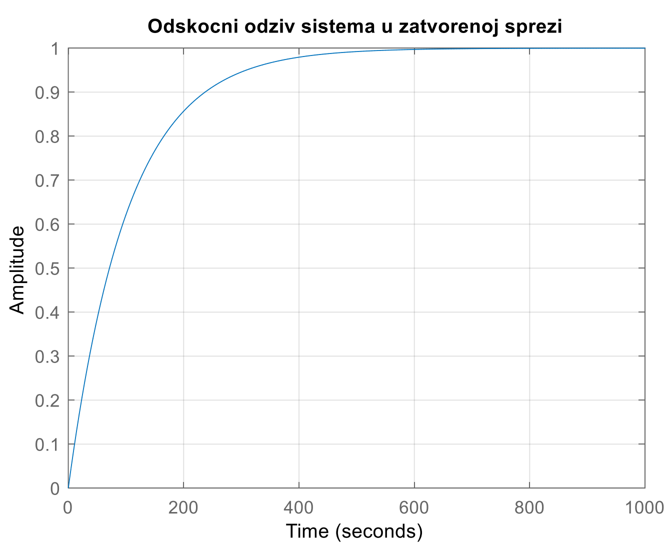


Figure 14: Step response of the system in closed loop.

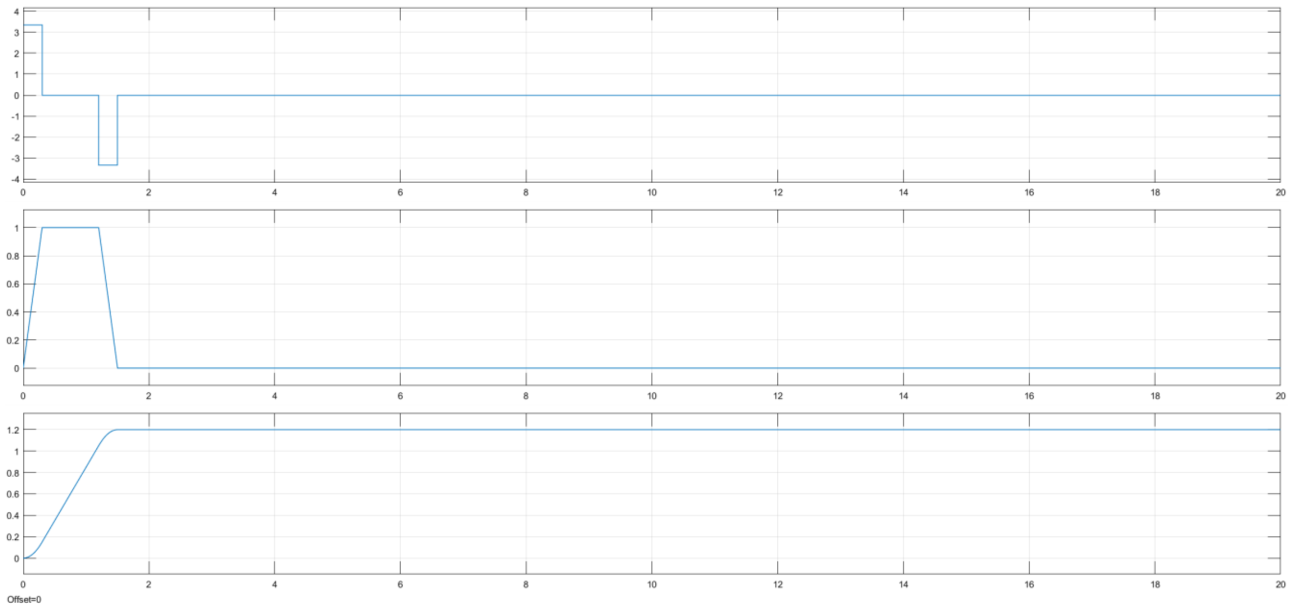


Figure 15: Graphics of reference trajectory.

In the setting of the PID controller, the Ziegler-Nichols method was first used.

According to this method, the parameters of the PID regulator are calculated as follows:

Kp = 0.6*Kkr

Ki = 0.6*Kkr*2/Tkr

Kd = 0.6*Kkr*Tkr/8

With a controller set up like this and without adding block that will add the physical limits of the system, we get the following graph as the output of the system:

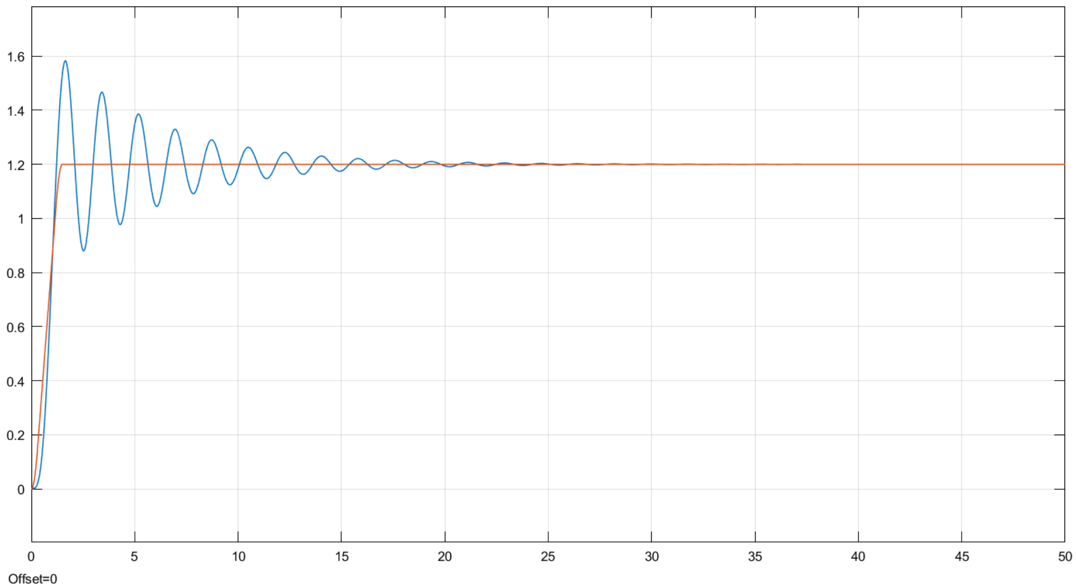


Figure 16: Reference and accomplished movement.

Then we introduce its physical limitations into the system by adding the saturation block. The graphs of control and output from the system with limits are as follows:

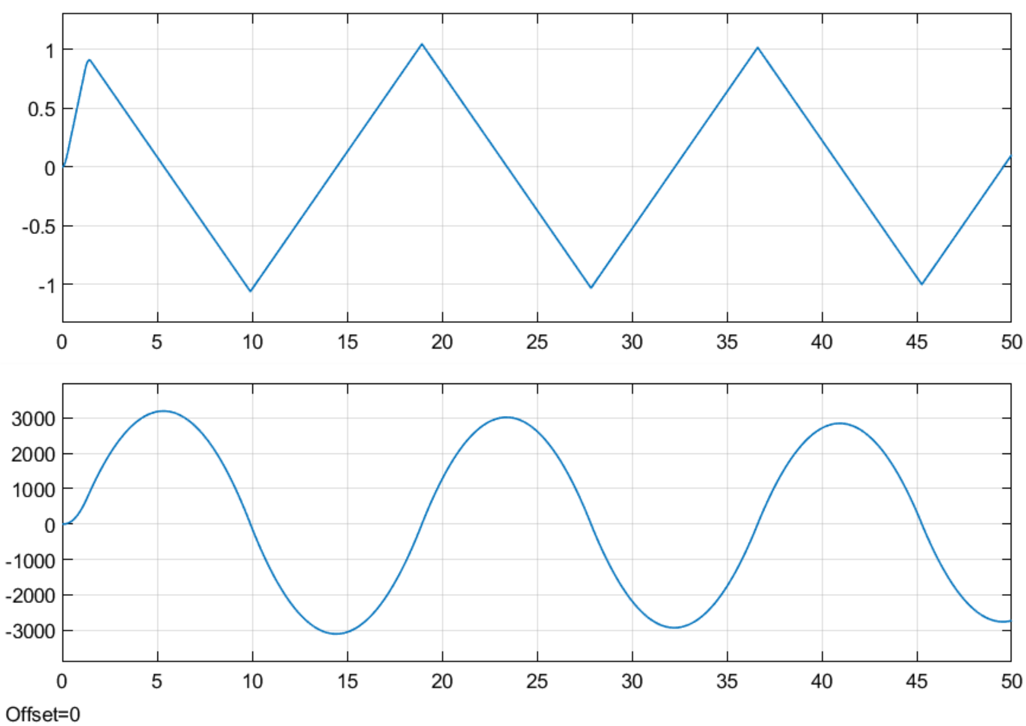


Figure 17: The influence of the PID regulator on control.

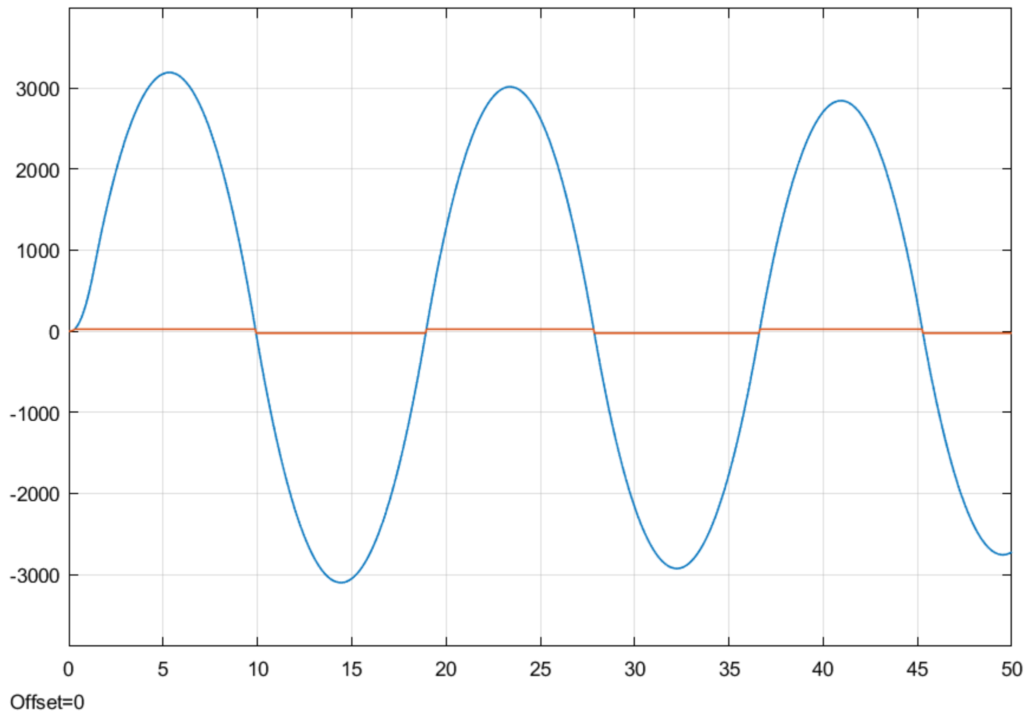


Figure 18: Control before and after saturation.

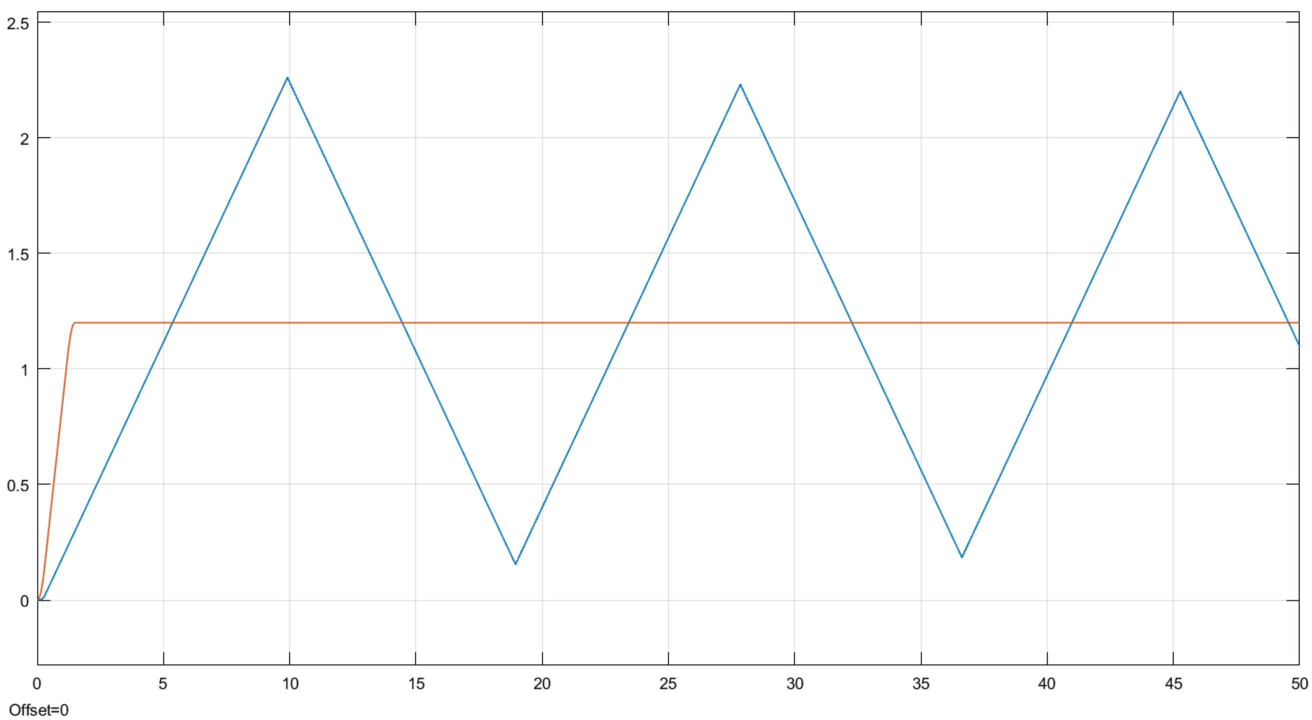


Figure 19: Reference and accomplished movement.

From the last graph we see that we got a ringing at the output of the system.

The theoretical values of control signal significantly exceed our physical limitations.

We see that the moment the system reaches the benchmark, it does not continue to track it, but significantly exceeds the reference input. This is due to the large accumulated integral action and at the moment when the reference value is reached, the output continues to grow until the integrator is discharged.

From there, we conclude that the key problem is in the adjustment of the PID regulator, specifically the integral action and that the Ziegler-Nichols method is not enough for us, but that we must subsequently reduce the integral effect so that the input, as much as possible, does not exceed the physical limits of the system.

When we reduce the integral effect 200 times, we get the following graphs:

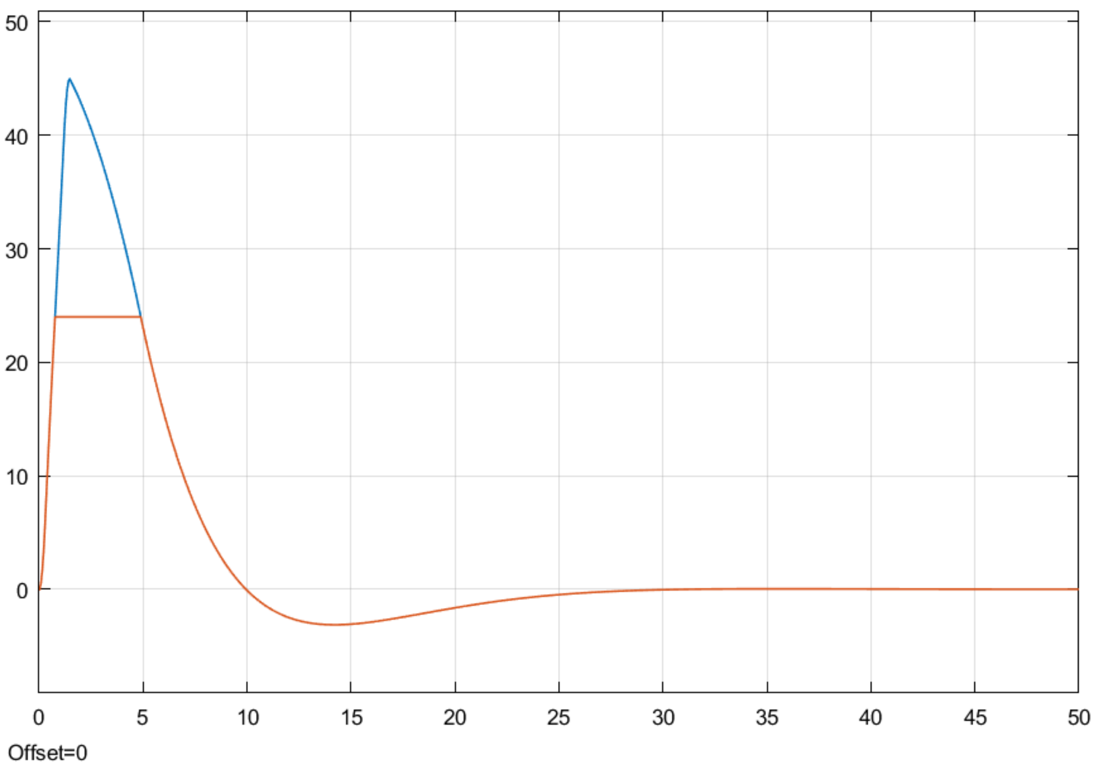


Figure 20: Control signal before and after saturation.

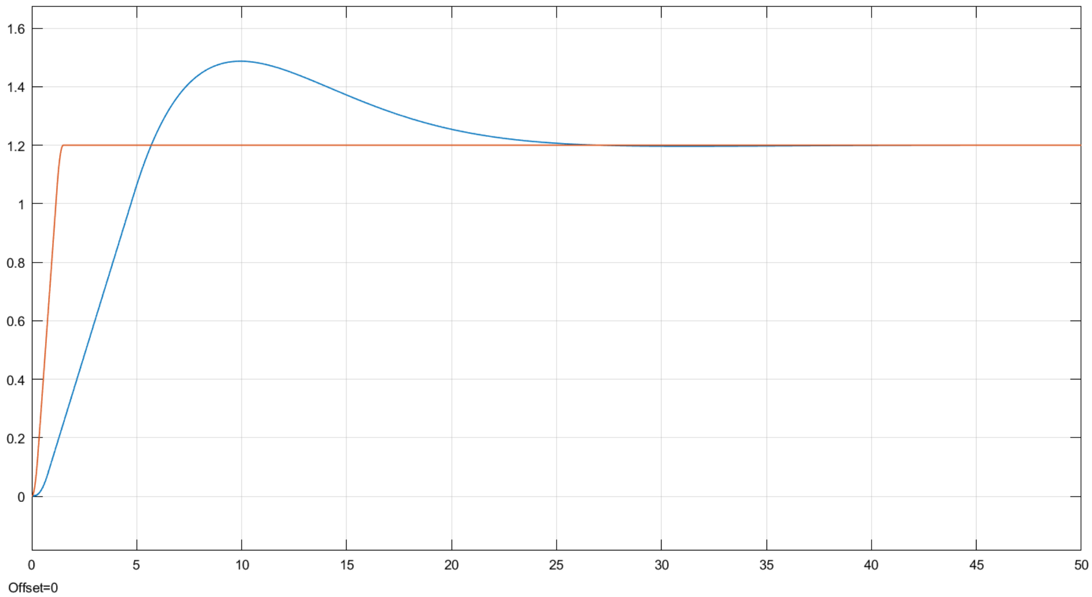


Figure 21: Reference and accomplished movement.

We see that we cut off a much smaller part when we reduce the integral effect and that we no longer have the ringing of the response.

If we were to eliminate the integral effect, we would get the following graphics:

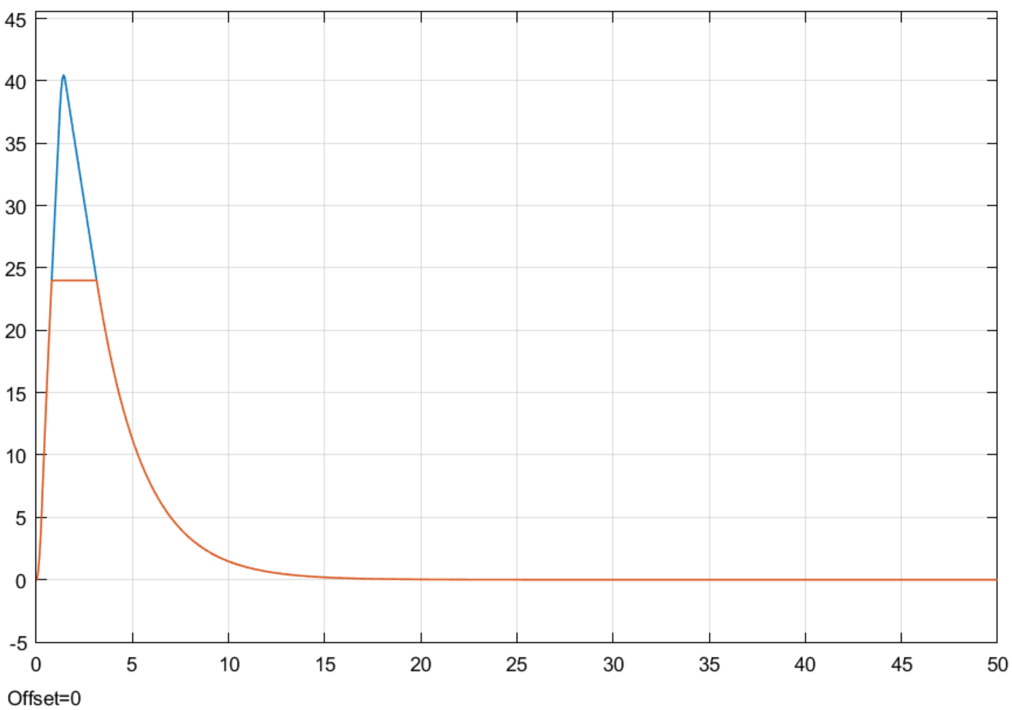


Figure 22: Control before and after saturation.

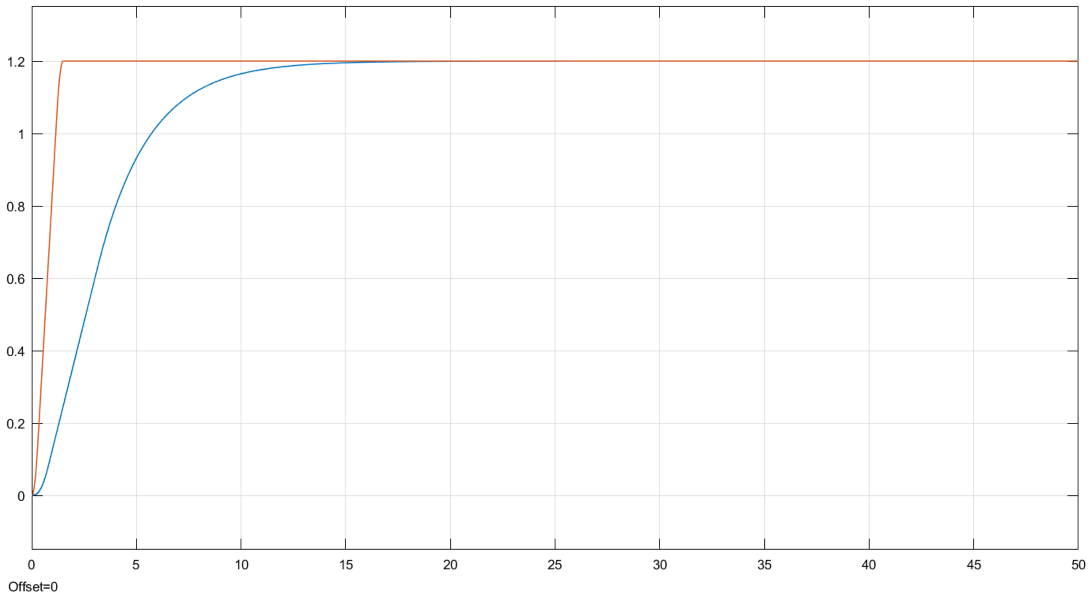
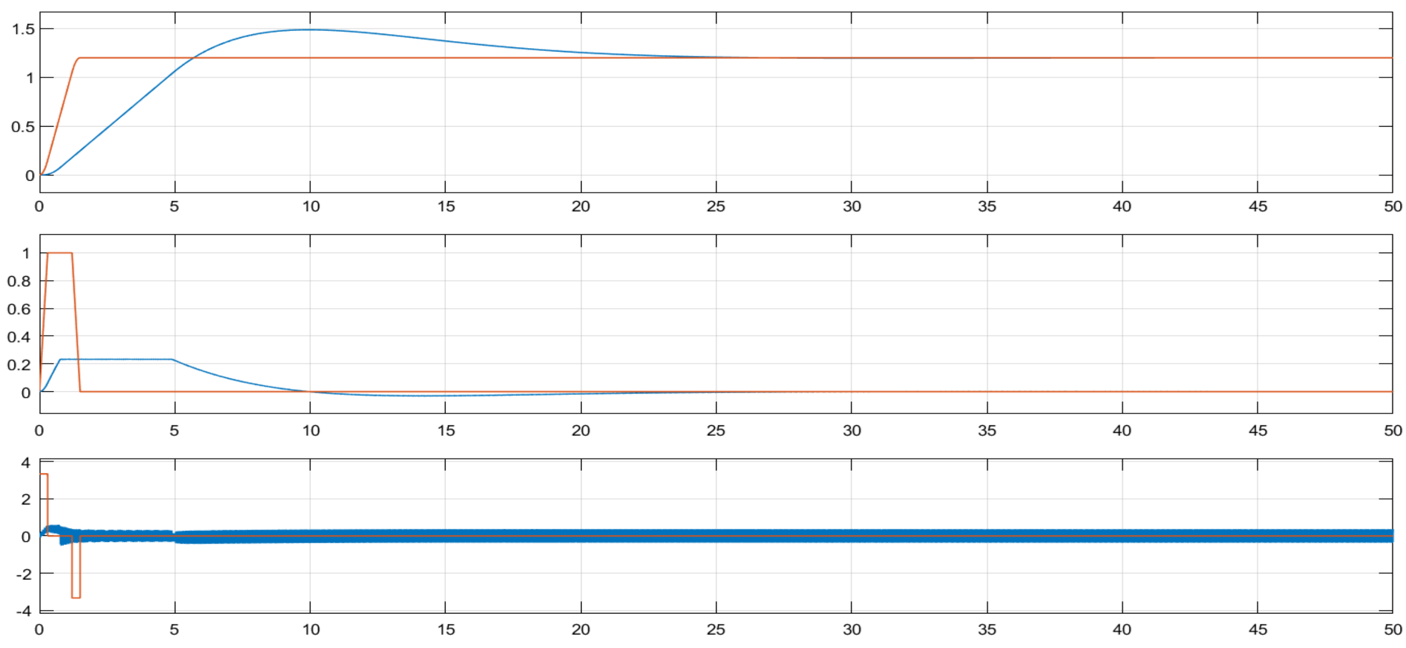


Figure 23: Reference and accomplished movement.

We know that by increasing the integral effect, the overshot increases, and the system accelerates. Therefore, when we completely remove it, we do not have an overshot, but the system is therefore slower.

The final graphics of reference and realized movement, speed and acceleration of the piston and with integral action of PID controller reduced by 200 times are:



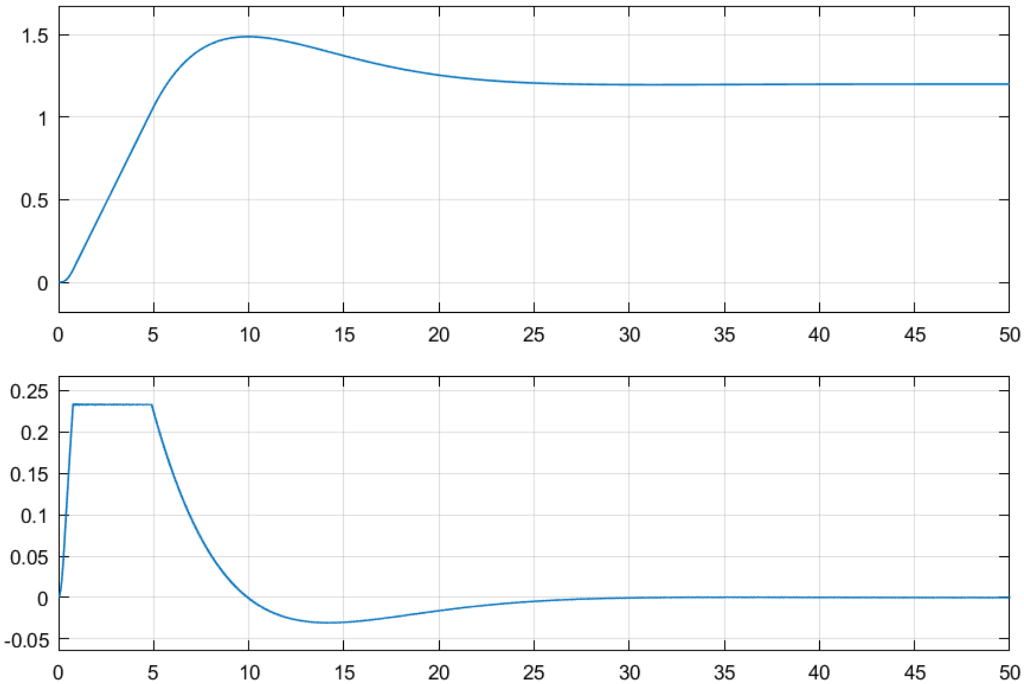


Figure 24: Graphics of movement of the piston and piston speed.

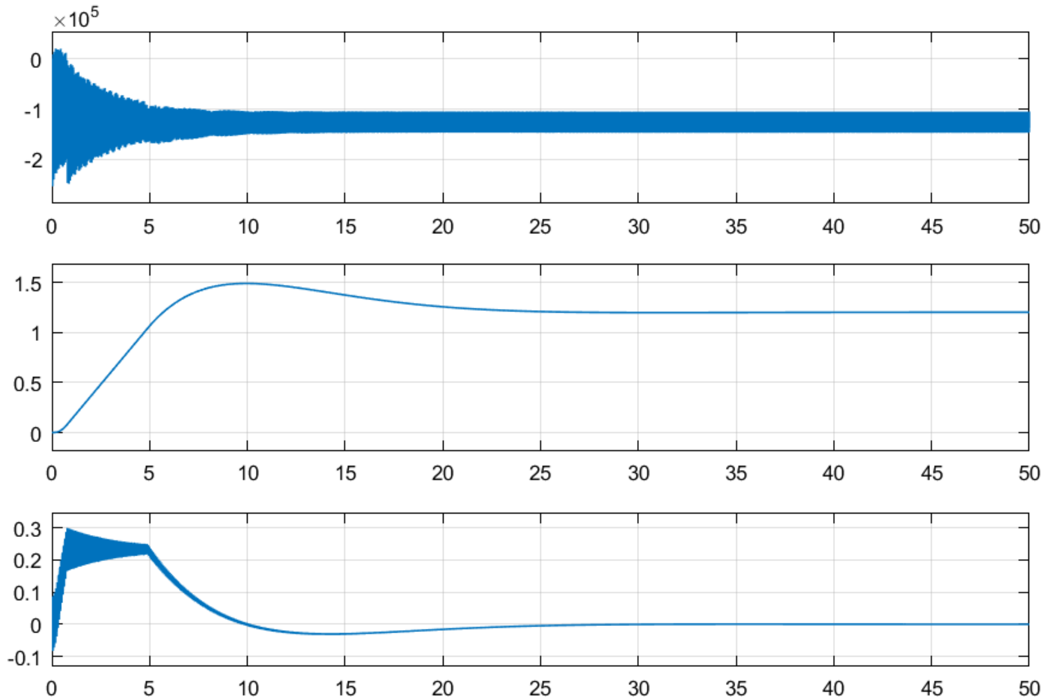


Figure 25: Differential pressure, movement of the valve and valve speed graphics.

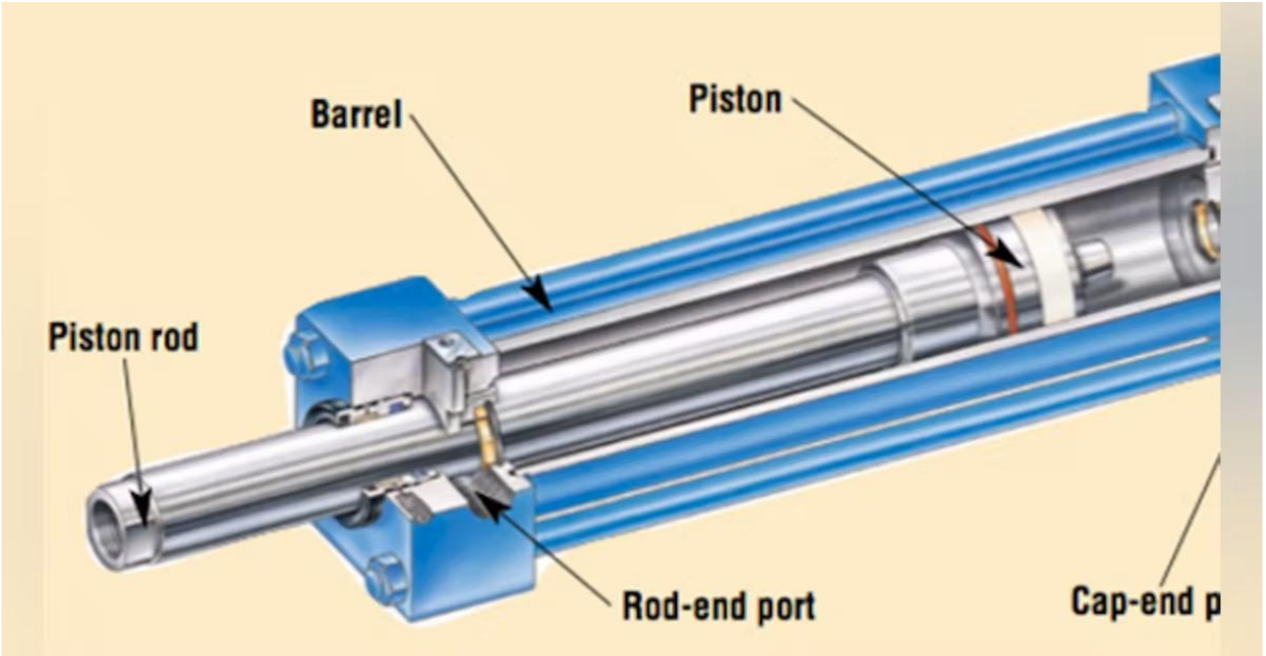


Figure 26: Parts of a hydraulic cylinder

CODE:

```
%% finding the maximum force value, stroke and speed of piston/piston rod

xp = z1.Data;
xp_dot = z2.Data;
xp_2dot = z2_dot.Data;
y = z4.Data;
y_dot = z5.Data;

F1 = k.*(xp - y) + b.*(xp_dot - y_dot);
F1_max = max(abs(F1));

Fl = (9/2).*m.*xp_2dot + F1;
Fl_max = max(abs(Fl));

h = max(z1.Data);
V = max(z2.Data);
```

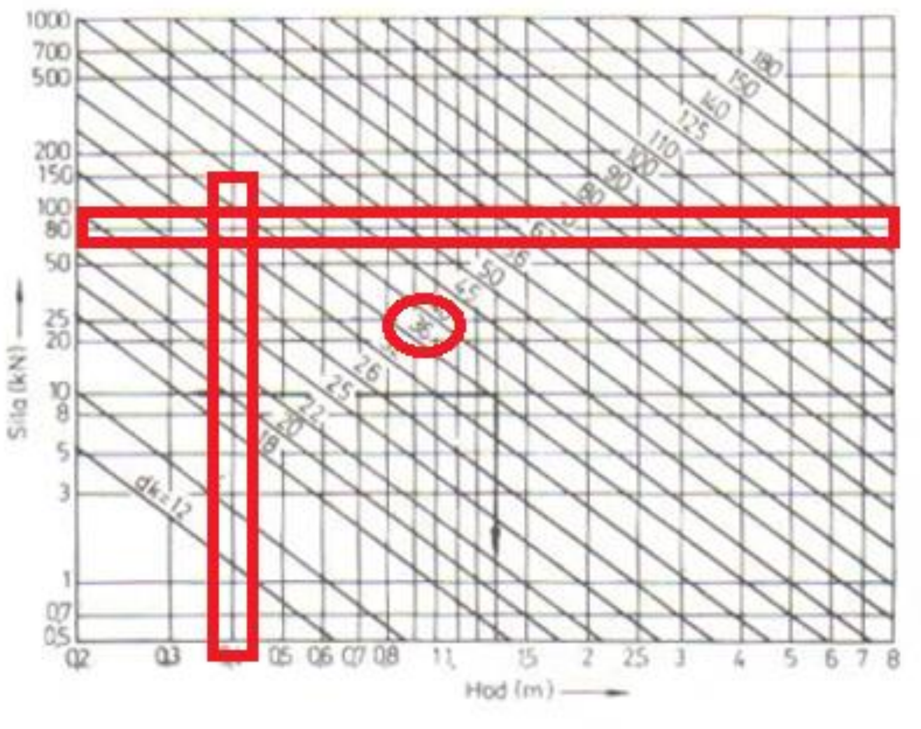
```
disp(F1_max);
disp(Fl_max);
disp(h);
disp(V);
```

Fi = 3710.2 N – the maximum force achieved by the actuator
h = 1.4861m – maximum clip stroke
V = 0.2345 m/s – maximum piston speed

When calculating the maximum force achieved by the actuator, we take absolute value, because for the force we get negative values. From that we conclude that it is a force when retracting, not pulling out the piston. Therefore, in the expression for force the active surface will depend on the diameter of the piston and the piston rod. In later equations the active surface depends only on the diameter of the piston.

1. DESIGN OF HYDRAULIC SYSTEM

Parameter	Value	Unit
Fi_{max}	3559.7	N
h	1.4874	m
v	0.2345	m/s
η_{CF}	0,9	/
η_{vol}	0,95	/
η_m	0,8	/
n	2000	obr/min
p	160	bar



d = 32mm based on a graphical method that directly uses Euler's pattern, for a force between 3 and 5 kN and at a stroke of 1.5m.

```
%% parameter determination

d = 32*10^(-3);
eta_cf = 0.9;
eta_vol = 0.95;
eta_m = 0.8;
n = 2000/60;
p = 160*(10^5);

Ap = Fl_max/p/eta_cf;
D = sqrt(4*Ap/pi + d^2);

disp(Ap);
disp(D);

Ap1 = D^2*pi/4;
Q = Ap1*v;
```

```
V = Q/n/eta_vol;

disp(Q);
disp(V);

P = p*Q/eta_m/eta_vol;

disp(P);
```

$Fi_{max} = p * A * \eta_{CF}$
$A = (D^2 - d^2) * \pi / 4$

$A = 2.472\ cm^2$

D = 36.6 mm – piston diameter

$Q = n * V * \eta_{vol}$

$Q = A * v$

$Q = 0.24661\ dm^3/s = 14.7966\ l/min$

$V = 7.7876\ cm^3$ – hydraulic pump volume

$P = (p * Q) / (\eta m * \eta vol)$

P = 5191.8 W – power of the hydraulic power unit that drives the pump