Appendix - Matlab Code:

Task_5_2.m : [Calculations of Controllability and Reachability matrix]

```
clc;
Co = [B A*B A^2*B A^3*B ];
Ob = [C C*A C*A^2 C*A^3 ]';
```

LabB_ControllerOverSimulator_Continuous_Parameters.m:

```
close all;
clc;

afPoles =[-5.6 -843.1 -4 -4];
K = acker(A,B,afPoles);

% sys_test = tf([1],[-3 -843 -3 -3]);
% order_test = order(sys_test)
%K = [ -10.0000  -57.4908 -105.0371  -19.5009 ];
%[−17.6405 − 38.8205 − 52.5060 − 8.9724];
```

LabB_ObserverOverSimulator_Continuous_Parameters.m:

```
% load the PID
%kP = -46.6;
%kI = -260;
%kD = -0.1;

% parameters loaded from lab 1 data
C = [1 0 0 0;0 0 1 0];
L = (place(A',C',[-5.6*10 -843.1*10 -4.01*10 -4.02*10]))';
C_acc = C(1,:);
C_acc_bar = C(2,:);
T_inv = [C_acc;0 1 0 0;0 0 1 0 ;0 0 0 1];
T = inv(T_inv);
A_hat = T_inv*A*T;
B_hat = T_inv*B;
C_acc_hat= C_acc*T;
C_acc_bar_hat= C_acc_bar*T;
A_yy = A_hat(1,1);
A_yx = A_hat(1:1,2:4);
A_xy = A_hat(2:4,1:1);
A_xx = A_hat(2:4,2:4);
```

```
B_y = B_hat(1,1);
B_x = B_hat(2:4,1:1);
C_y = C_acc_bar_hat(1,1);
C_x = C_acc_bar_hat(1:1,2:4);
AA = A_xx;
CC = [1 0 0; 0 1 0];
L_new = (place(AA', CC' , [-5.6*10 -843.1*10 -4.01*10]))';
L_acc = L_new(1:3,1:1);
L_acc_bar = L_new(1:3,2:2);
M1 = A_xx - (L_acc* A_yx) - (L_acc_bar* C_x);
M2 = B_x - (L_acc*B_y);
M3 = A_xy - (L_acc* A_yy) - (L_acc_bar*C_y);
M4 = L_acc_bar;
M5 = L_acc;
M6 = T(1:4,1:1);
M7 = T(1:4,2:4);
figure;
subplot(3,1,1)
plot(x_w.time,x_w.signals.values(:,1))
title('x_w')
subplot(3,1,2)
plot(x_w.time,x_w.signals.values(:,2))
title('full order estimator x_w')
subplot(3,1,3)
plot(x_w.time,x_w.signals.values(:,3))
title('reduced order estimator x_w')

figure;
subplot(3,1,1)
plot(theta_b.time,theta_b.signals.values(:,1))
title('theta_b')
subplot(3,1,2)
plot(theta_b.time,theta_b.signals.values(:,2))
title('full order estimator theta_b')
subplot(3,1,3)
plot(theta_b.time,theta_b.signals.values(:,3))
title('reduced order estimator theta_b')


%%
x_w_system = x_w.signals.values(:,1);
x_w_full= x_w.signals.values(:,2);
x_w_reduced= x_w.signals.values(:,3);
```

```
full_x_error = abs(max(x_w_system - x_w_full))
reduced_x_error = abs(max(x_w_system - x_w_reduced))
theta_b_system = theta_b.signals.values(:,1);
theta_b_full= theta_b.signals.values(:,2);
theta_b_reduced= theta_b.signals.values(:,3);
full_theta_error = abs(max(theta_b_system - theta_b_full))
reduced_theta_error = abs(max(theta_b_system - theta_b_reduced))
```

LabB_ControllerOverSimulator_Discrete_Parameters.m:

```
close all;
clc;
% select the sampling time
%fSamplingPeriod = 0.005;
fSamplingPeriod = 0.01;
sys = ss(A,B,C,D);
%opt = c2dOptions('Method','tustin');
ed = eig(sysd.A);
oldpoles = [-843.1 -5.6 -4 -4];
polesd = exp(oldpoles*fSamplingPeriod);
Kd = acker(sysd.A,sysd.B,polesd);
%Kd = [-8.1792  -49.1223  -71.4928  -11.5909];
```

LabB_ObserverOverSimulator_Discrete_Parameters.m:

```
close all;
%clear all;
clc;
% select the sampling time
fSamplingPeriod = 0.01;
% load the PID
% kP = -46.6;
% kI = -260;
% kD = -0.1;

Cd = [1 0 0 0;0 0 1 0];
Dd= 0;
Ad = sysd.A;
Bd= sysd.B;
factor = 10;
Ld = (place(Ad',Cd',polesd))';
C_accd = Cd(1,:);
```

```
C_acc_bard = Cd(2,:);
T_invd = [C_accd;0 1 0 0;0 0 1 0 ;0 0 0 1];
Td = inv(T_invd);
A_hatd = T_invd*Ad*Td;
B_hatd = T_invd*Bd;
C_acc_hatd= C_accd*Td;
C_acc_bar_hatd= C_acc_bard*Td;
A_yyd = A_hatd(1,1);
A_yxd = A_hatd(1:1,2:4);
A_xyd = A_hatd(2:4,1:1);
A_xxd = A_hatd(2:4,2:4);
B_yd = B_hatd(1,1);
B_xd = B_hatd(2:4,1:1);
C_yd = C_acc_bar_hatd(1,1);
C_xd = C_acc_bar_hatd(1:1,2:4);
AAd = A_xxd;
CCd = [1 0 0; 0 1 0];
L_newd = (place(AAd', CCd' , [polesd(1) polesd(2) polesd(3)]))';
L_accd = L_newd(1:3,1:1);
L_acc_bard = L_newd(1:3,2:2);
Md1 = A_xxd - (L_accd* A_yxd) - (L_acc_bard* C_xd);
Md2 = B_xd - (L_accd*B_yd);
Md3 = A_xyd - (L_accd* A_yyd) - (L_acc_bard*C_yd);
Md4 = L_acc_bard;
Md5 = L_accd;
Md6 = Td(1:4,1:1);
Md7 = Td(1:4,2:4);
figure;
plot(x_w.time,x_w.signals.values(:,1))
title('x_w for complete system')
figure;
plot(theta_b.time,theta_b.signals.values(:,1))
title('theta_b for complete system')
figure ;
plot(u.time,u.signals.values);
title('u input')
```