A project report submitted to Institute for Automation of Complex Power Systems RWTH Aachen.

Controller Design for an Inverter Pendulum System controlled by a Motor

Banerjee, Dipankar	417640	MATLAB programming, Conceptual design and Implementation.	Dipankar Banerjee.
De, Nilava	426577	MATLAB programming testing and verification, Report.	Nilava De

Submission Date: 25.08.2021

1. Introduction

1.1)

Linearization:

$$\left\{\left(\frac{mr}{3} + md\right) l^2 + \frac{m_d r_d 2}{2}\right\}\ddot{\theta} = -\left(\frac{m_r}{2} + m_d\right) gl \sin\theta + Tu - c\frac{d\theta}{dt} \qquad (1)$$

$\sin\theta$ can be written as below as per Taylor's Expansion:

$$\sin\theta = [\theta - \frac{\theta^3}{3!} - \frac{\theta^5}{5!} \ldots]$$

At $\theta = \pi$,

$$\theta - \pi = t \;\rightarrow\; \theta = \pi + t$$

$$\sin\theta = \sin(\pi + t) = -sint$$

$$\sin\theta \approx sint = -[t - \frac{t^3}{3!} - \frac{t^5}{5!} \ldots]$$

$$= -t + \frac{t^3}{3!} + \frac{t^5}{5!} \ldots$$

$$= -(\theta - \pi) + \frac{(\theta - \pi)^3}{3!} + \frac{(\theta - \pi)^5}{5!} \ldots$$

Approximating $\sin\theta$ as per Taylor 1st order expansion, to obtain the linearized system at the given equilibrium point $\theta = \pi$,

$$\sin\theta = -(\theta - \pi) \qquad (a)$$

Using (a) in eq (1) we get,

$$\ddot{\theta} = \frac{1}{J}(\frac{mr}{2} + \ddot{m}d)gl(\theta - \pi) + \frac{1}{J}T_u - \frac{C}{J}\ddot{\theta}$$

$$x_1 = \theta$$

$$x_2 = \dot{\theta} = \dot{x_1} = \omega$$

$$y = \theta = x_1$$

$$u = T_u$$

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ \frac{1}{J}\left(\frac{mr}{2} + md\right)gl & -\frac{c}{z} \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{1}{J} \end{bmatrix} u + \begin{bmatrix} 0 \\ \frac{-\pi}{J} \end{bmatrix}$$

$$y = [1 \quad 0]x$$

1.2.)

$$m_T l^2 \ddot{\theta} = -m_T g l \sin\theta + T_u - c\dot{\theta}$$

Using θ = π,

$$J\ddot{\theta} = m_T g l(\theta - \pi) + T_u - c\dot{\theta}$$

$$x_1 = \theta$$

$$x_2 = \dot{\theta} = \dot{x_1}$$

$$y = \theta = x_1$$

$$u = T_u$$

$$\dot{x_2} = \frac{m_T g l}{J} x_1 + \frac{1}{J} u + \frac{c}{J} x_2 - \frac{m_T g l}{J} \pi$$

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ \frac{m_T g l}{J} & -\frac{c}{z} \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{1}{J} \end{bmatrix} u + \begin{bmatrix} 0 \\ \frac{-m_T g l}{J} \pi \end{bmatrix}$$

$$y = [1 \quad 0]x$$

Type equation here.

Task 1

In Task 1 we are ignoring the closed loop motor dynamics and assuming no noise is present.

Task 1.1

In Task 1.1 firstly, we have set the **Inertial_model** parameter to **1** and defined the plant model as well as created the state space of the model. After that we have plotted the pole-zero map and the following values were obtained as per result;

$$P_1 = -5.1662, P_2 = 4.9789$$

The system is unstable as we have one positive pole. (Pole on the RHP)

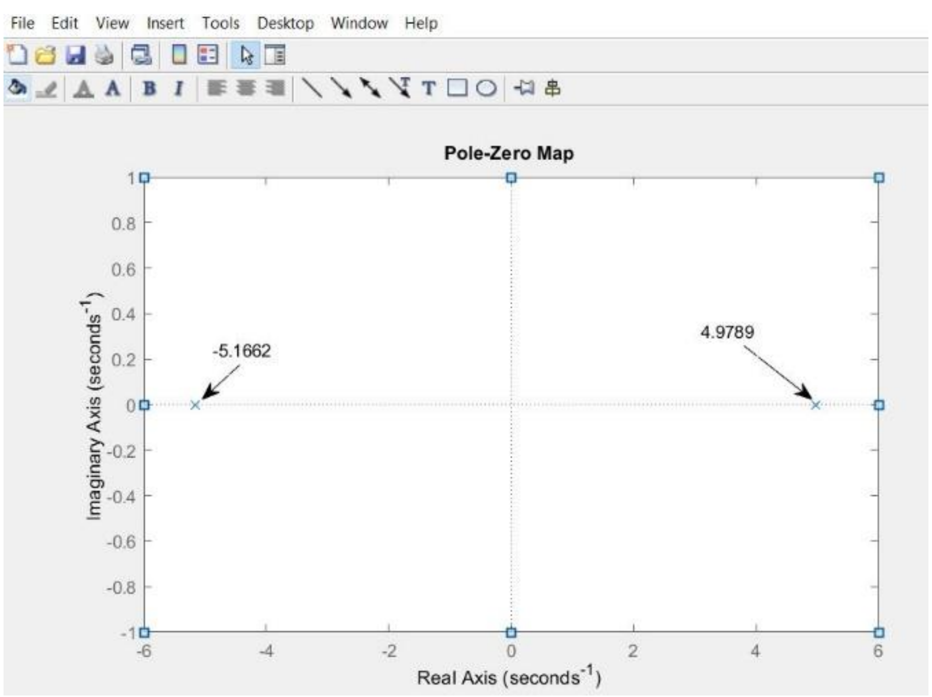

Figure 1. PZ map

Subsequently, if we decrease the length of the rod, the system still remains unstable, but the system becomes significantly faster and we obtain the new pair of poles which comes to be;

$$P1 = -7.5435, P2 = 6.7968$$

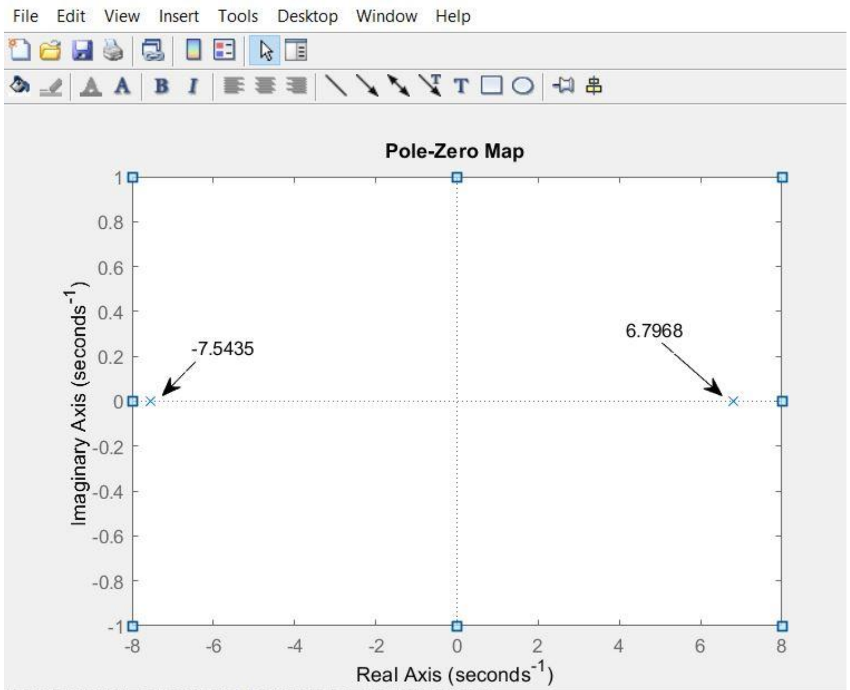

Figure 2. PZ plot decreasing rod Length

Furthermore, if we assume the length of the rod to be fixed and decrease the radius of the disc, the system still remains unstable, the values of the poles move away from the origin very slightly. So, it can be assumed that the radius of the disc in the given system has a very slight effect on the system dynamics. The new poles are;

$$P_1 = -5.1684,\ P_2 = 4.9810$$

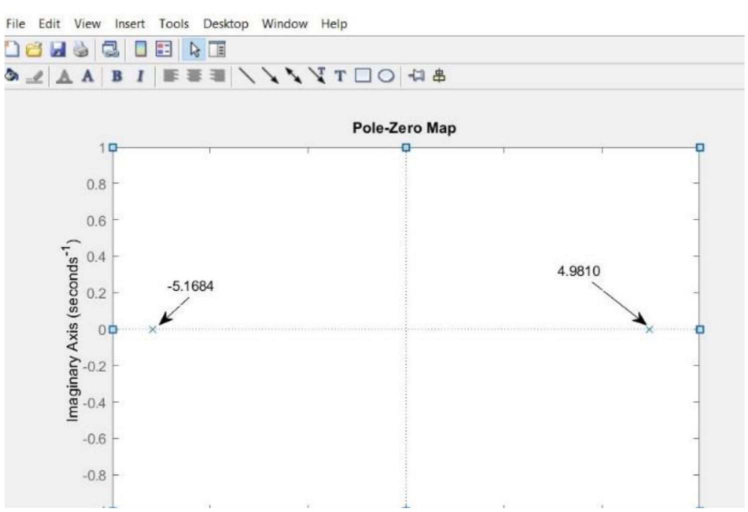

Figure 3. PZ plot decreasing disc radius

Similarly, if we increase the radius of the disc the system still remains unstable, but the values of the poles move closer to the origin very slightly. So, it can be assumed that the radius of the disc in the given system has a very slight effect on the system dynamics.

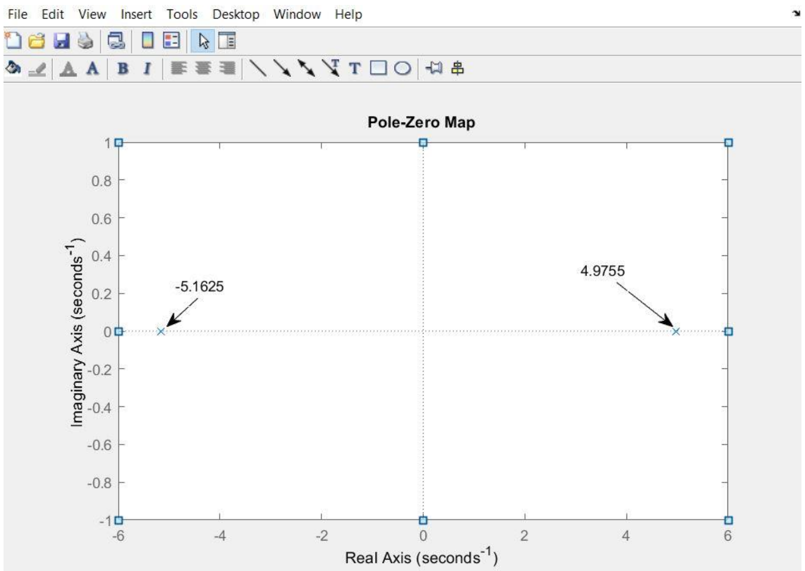

Figure 4. PZ plot increasing disc radius

Task 1.2

In Task 1.2 we only set the **Inertial_model parameter** to **2** and rest everything remains unchanged. Hence, the total mass is assumed to be concentrated as a point mass through a massless rod.

Hence, we can observe that the Left hand poles of the system are shifted more towards the origin, therefore the system becomes slower.

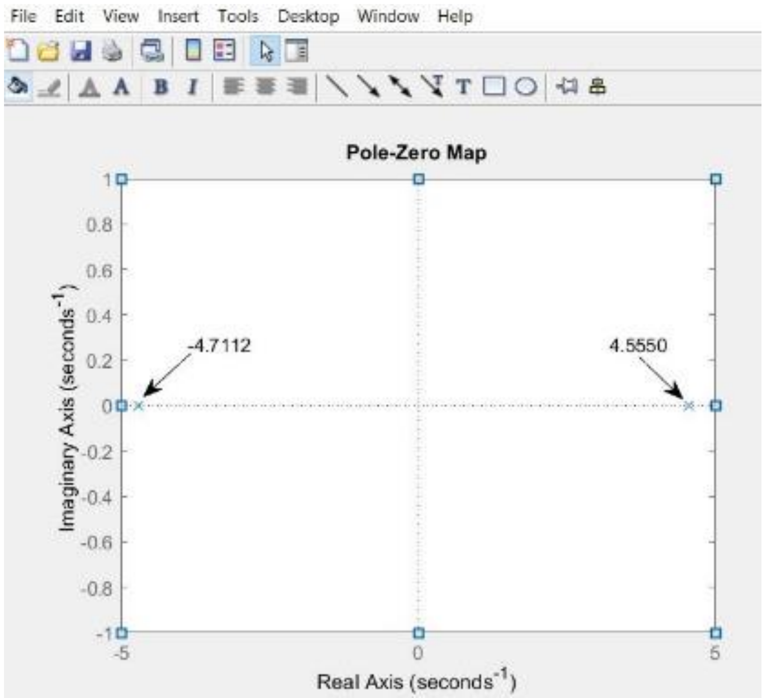

Figure 5. PZ plot pointless mass

Now if we decrease the mass of the point mass, the poles of the system shift further towards the origin making the system even slower than it was previously.

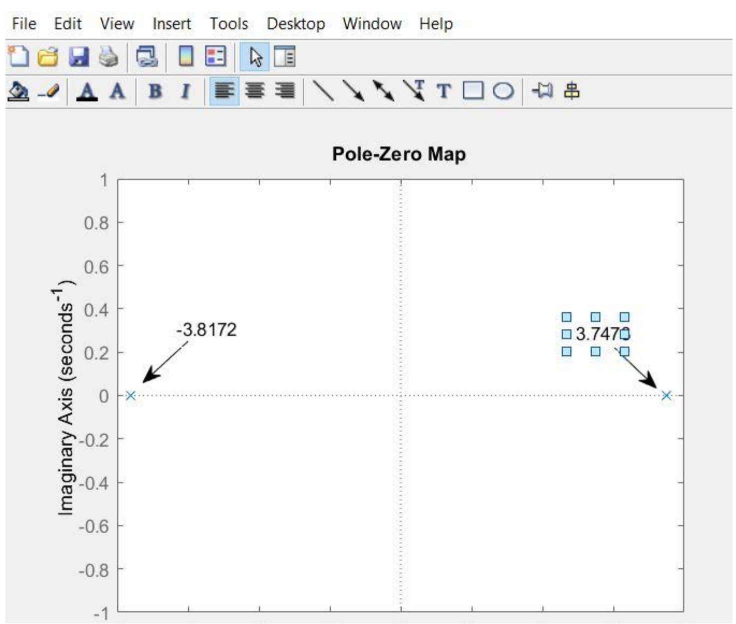

Figure 6. PZ plot pointless reduced mass

Finally, if we increase the mass of the point mass, the poles shift towards the left and makes the overall system faster.

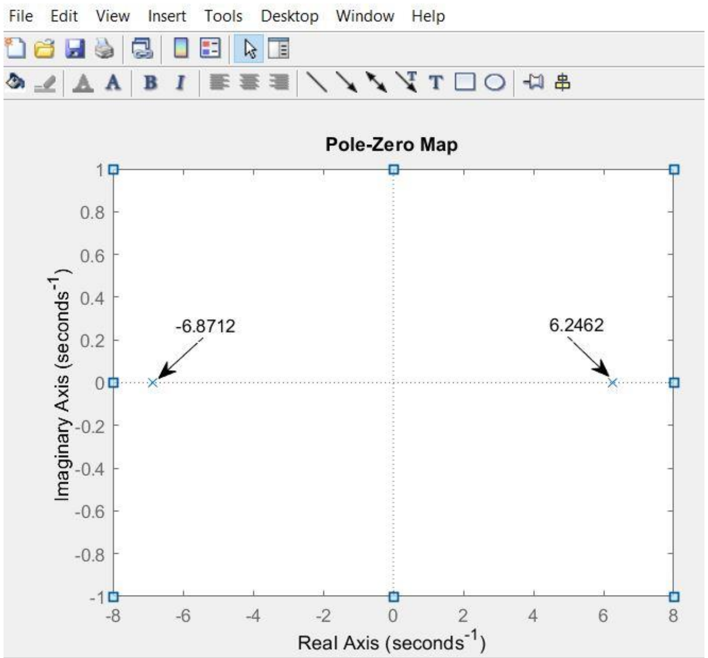

Figure 7. PZ plot pointless increased mass

Task 1.3

In this task we discretise the plant model using the zero-order hold method and then we calculate the poles of the system. From here we can see that the plant is unstable. We consider that there is no actuator dynamics of the motor for this task. Now we design a Linear Quadratic Regulator control to control the position of the pendulum and stabilize it in a vertical position. Firstly, we check the controllability of the system and then we design the controller. A disturbance torque of $\pm 75\text{Nm}$ is also applied to the plant. (As we are including the actuator dynamics of the motor for the next task, an extra state needs to be considered as well, therefore we will have 3 separate gains so we are assuming the 3rd gain to be zero, so that we have a universal dimension for the gain of the controller.)

$\mathrm{Q} = [2000\ 0;\ 0\ 50];$

$\mathrm{R} = 1;$

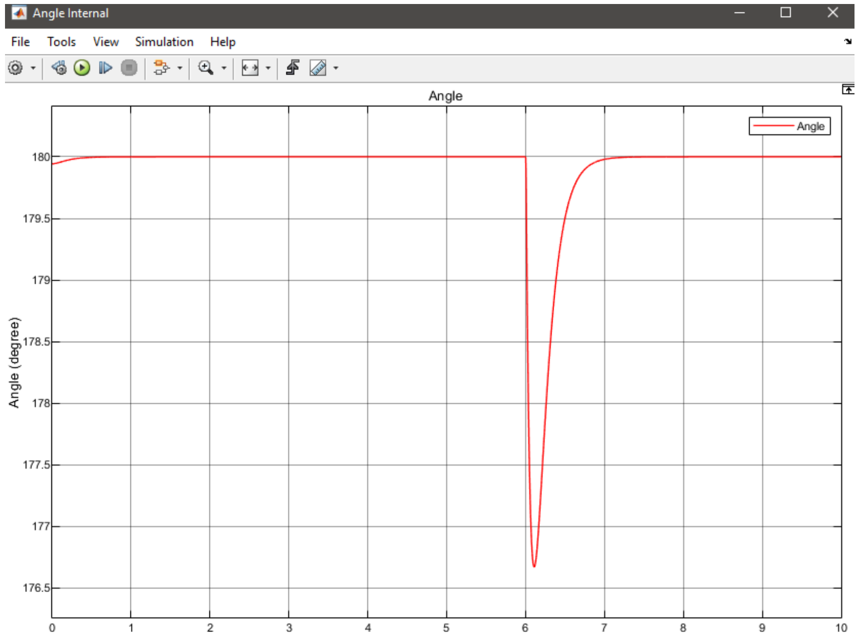

Figure 8. Internal angle

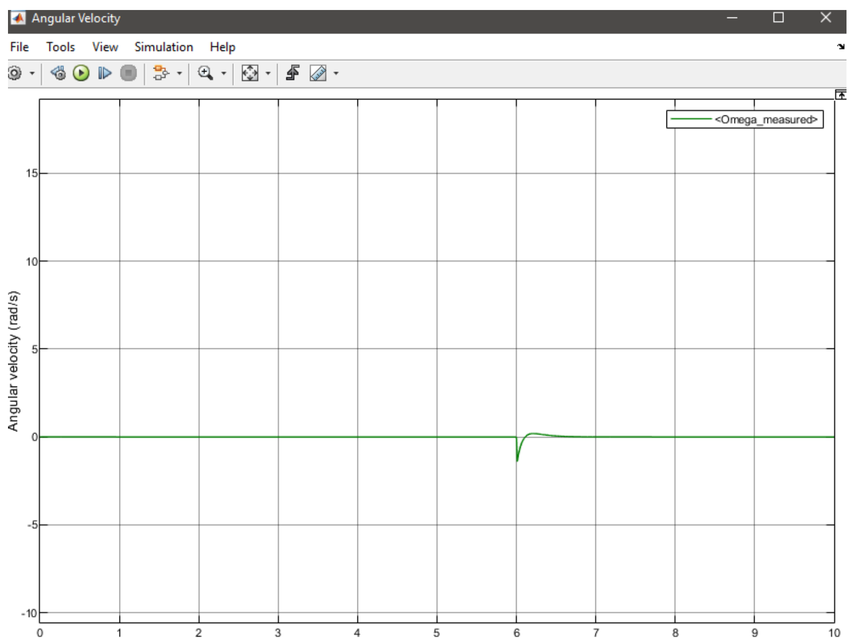

Figure 9. Angular velocity

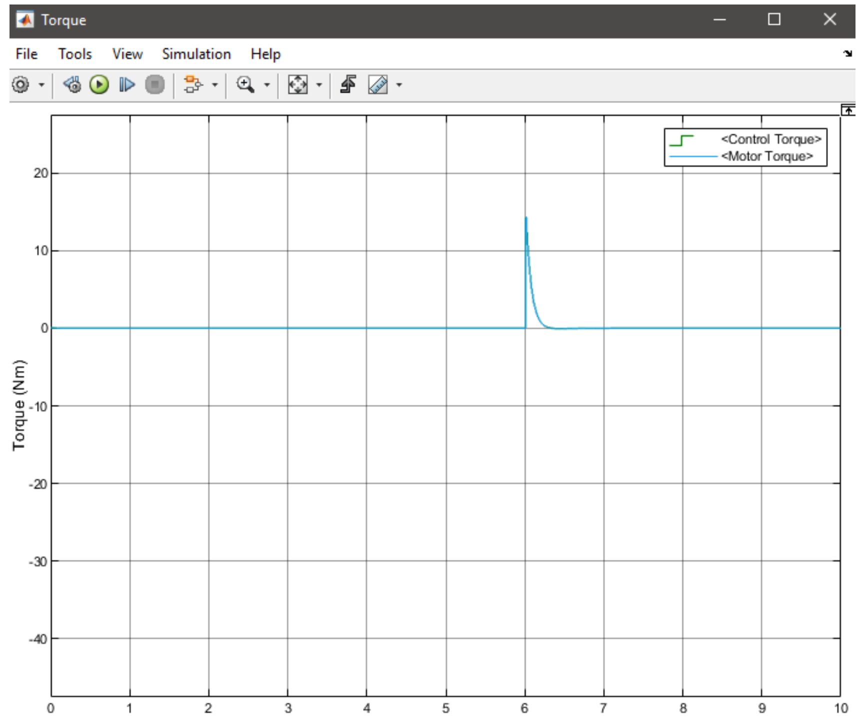

Figure 11. Torque

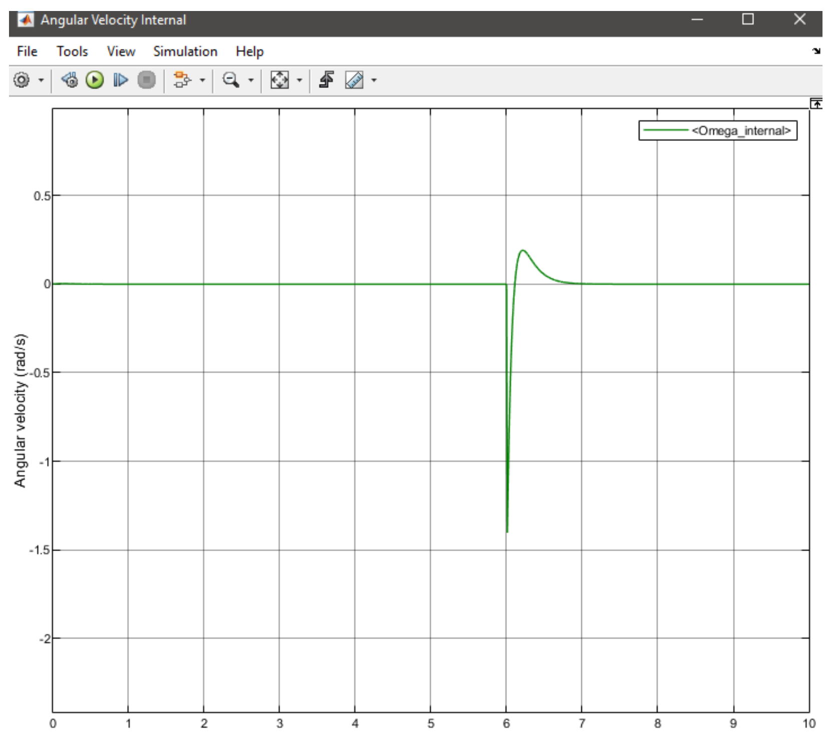

Figure 10. Internal angular velocity

From the above plots we can see that the motor torque remains well within the limit of ± 20Nm and the angle reaches its original state within 1 sec after the disturbance is added.

Task 2

For task 2 we now introduce the motor dynamics and rest remains same as the previous task.

Task 2.1

In this task we have an additional state that is being added by the motor, hence now there are 3 states that we have to measure. Therefore, we are adding an observer which will provide an estimated observed state.

$$TF = \frac{[-Plant1.T_{\_mn} 1]}{[Plant1.T_{\_md} 1]}$$

Task 2.2

In this task we again discretise the plant model and design the LQR. We first check the controllability then calculate the relative gain. We obtain the values of Q and R after running the LQR algorithm. We maximize the 1st state of Q matrix as we want to control the angle, hence we punish error in the angle which is relatively higher than the other states. We now perform an observability check and add an observer to the system. Then we are check the poles of the system which makes the system unstable.

Q = [2000 0 0;0 50 0;0 0 50];
R = 1;

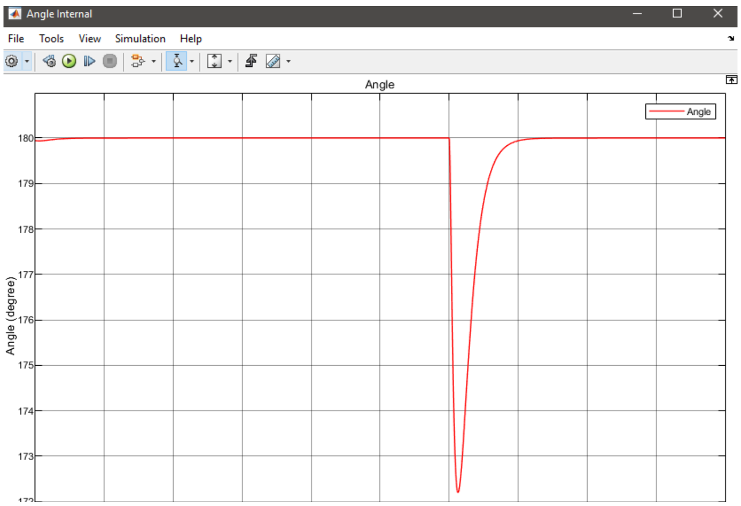

Figure 12. Angular velocity

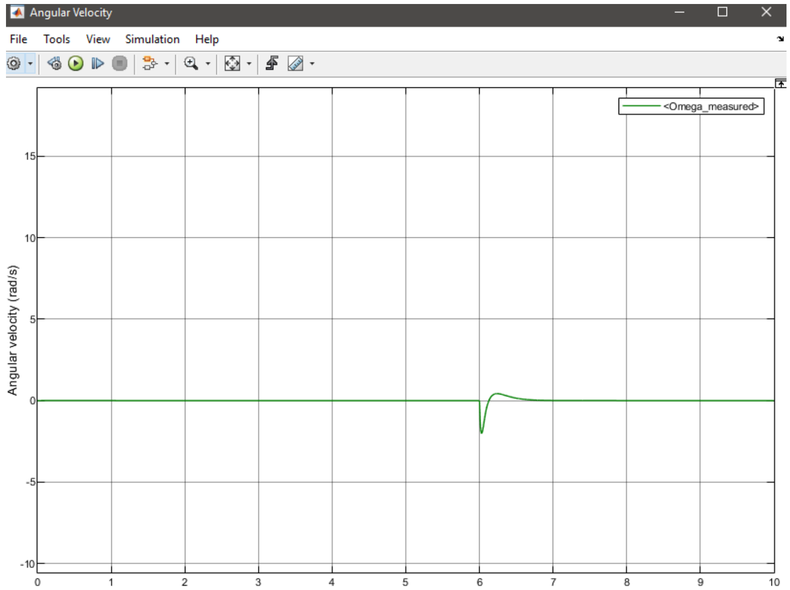

Figure 13. Internal angle

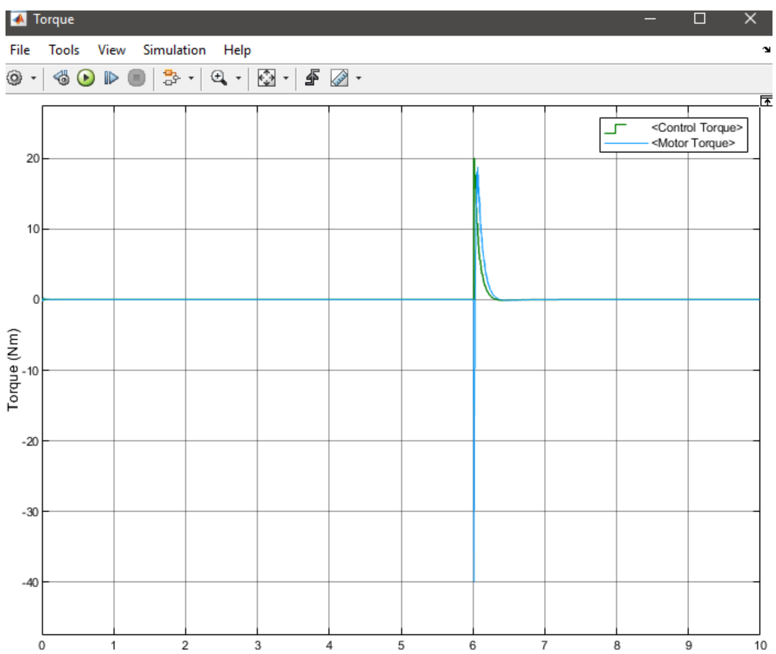

Figure 14. Torque

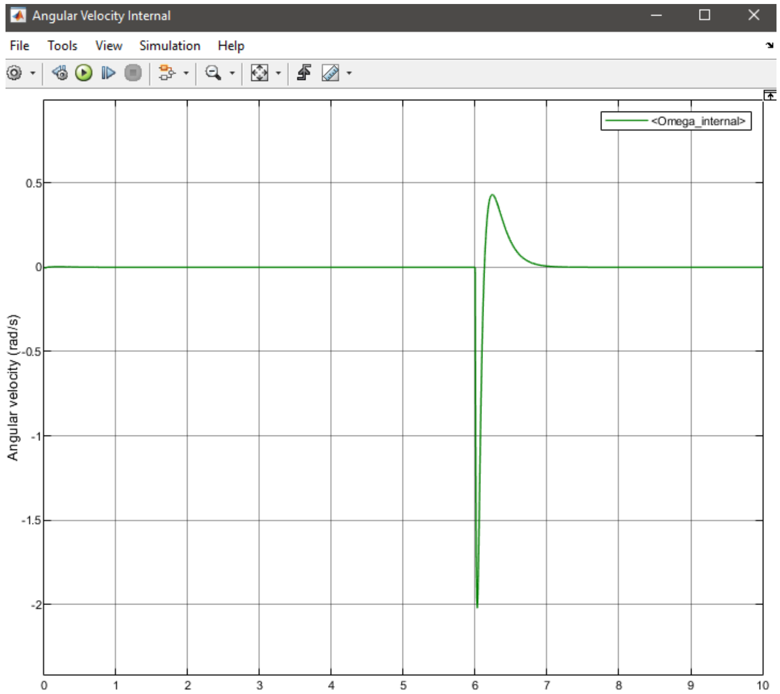

Figure 15. Internal angular velocity

Task 3

In this task we introduce measurement noise to the system.

Task 3.1

In this task we discretise the system and design the Kalman filter to estimate the states.

Since we have only 1 output to measure, we would consider single element for the noise, i.e., a scalar.

There is no process noise in the system. So, we select Q as 0. We have white noise as measurement noise. Since white noise has a 0 mean and 1 covariance distribution, we assume Q as 1. The correlation between process noise and measurement noise is zero and we have a zero-mean covariance. Hence the value N is zero.

Task 3.2

Finally, we have designed LQG regulator and accordingly the optimal gain is calculated. The state-feedback is computed and then we use a Kalman Filter block in Simulink to model the state estimate from the measure noisy states.

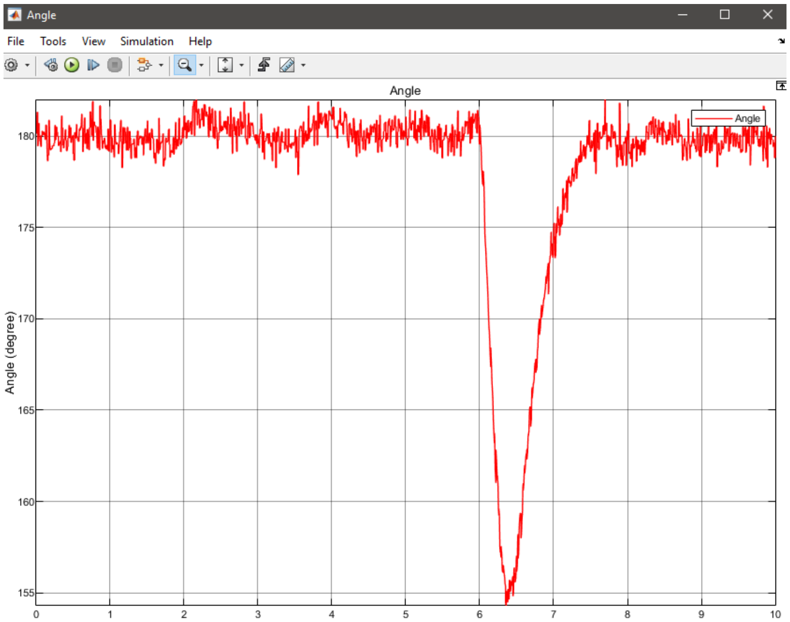

Figure 16. Angle

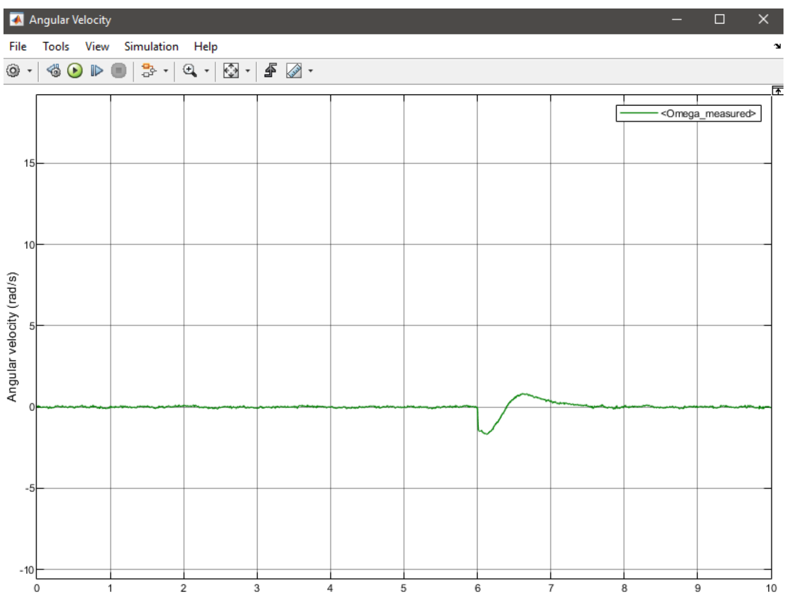

Figure 17. Angular velocity

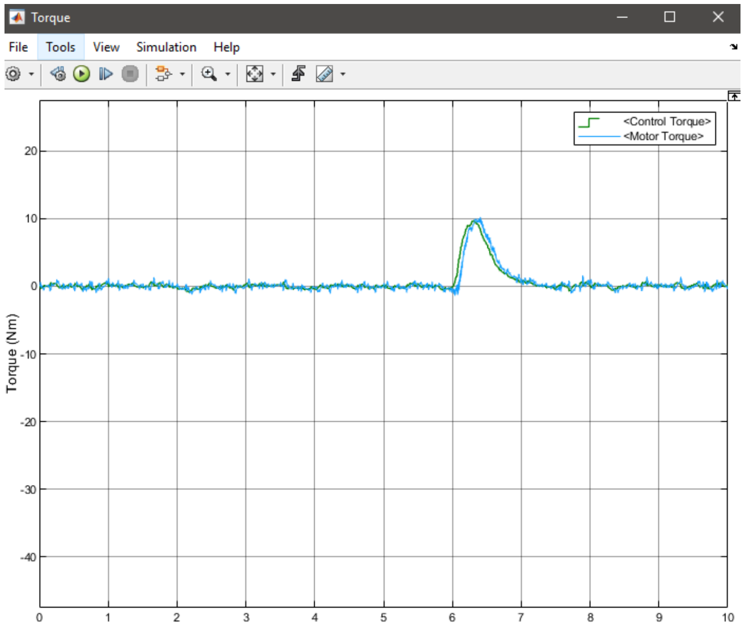

Figure 18. Torque

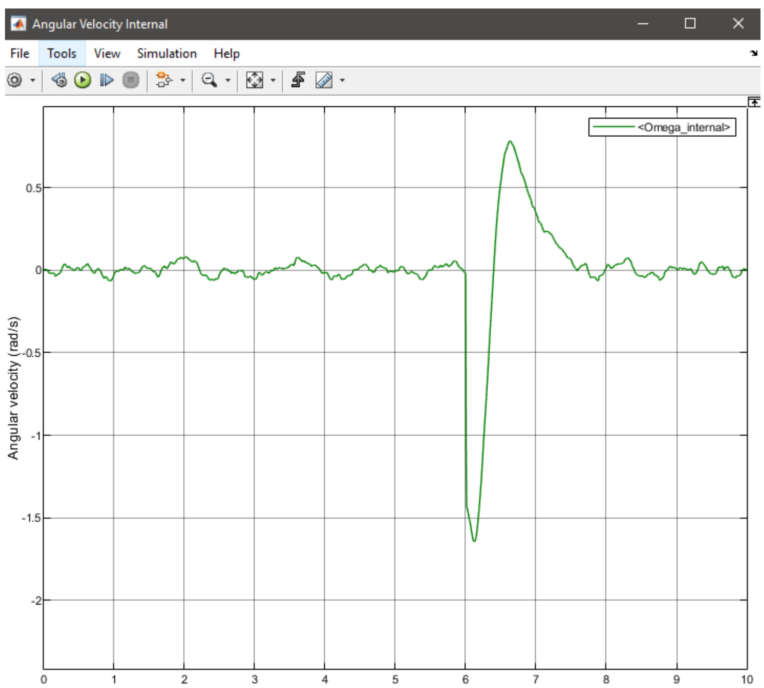

Figure 19. Internal angular velocity