Design and Analysis of Controller for a Robotic Arm Manipulator via State Space Approach

by

Nurfatihah Khalid

Dissertation submitted in partial fulfillment of
the requirements for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

DECEMBER 2010

UniversitiTeknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Design and Analysis of Controller for a Robotic Arm Manipulator via State Space Approach

by

Nurfatihah Khalid

A project dissertation submitted to the
Electrical and Electronic Engineering Programme
UniversitiTeknologi PETRONAS
in partial fulfilment of the requirement for the
BACHELOR OF ENGINEERING (Hons)
(ELECTRICAL AND ELECTRONIC ENGINEERING)

Approved by,

____
(Assoc. Prof Dr. Nordin Saad)

UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

December 2010

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

NURFATIHAH KHALID

ABSTRACT

The purpose of this study is to analyze and design a controller for robotic arm manipulator via state space approach. The dynamic of the system is modeled in state space representation as it provides a convenient and systematic way to model and analyze any systems. Further analysis of the system is performed by Matlab/Simulink with Control Tool Box. Based on the most suitable model, controller is first designed to modify the behavior of the system through feedback.Since most of current controller design techniques require the knowledge of the full system state for their implementation, observer is designed to computes the system states from the knowledge of the inputs and outputs of the system to be observed. Optimal control was also designed to minimize certain performance index. In short, this project is looking at modern control approach for the controller of a robotic arm manipulator which is expected to be better in terms of the system controllability and stability.

ACKNOWLEDGEMENT

Assalamualaikum w.b.t and Alhamdulillah,
First and foremost, I would like to praise and thanked the Lord above for giving me his guidance, assistance and strength in completing this Final Year Project successfully on time and making it a success. Without His assistance, I would not be able to finish and complete the project.

I would like to thank all people who have contributed directly and indirectly towards the accomplishment of this project, especially my dedicated supervisor, Associate Professor DrNordinSaad, for the valuable guidance, advice, and for keeping me on the right track and offering encouragement when I needed it. He inspired me greatly and his willingness to teach and sharing his knowledge contributed tremendously to this project.

Also, I want to express my gratitude to MsSitiHawa for her assistances in the documentation process. Her assistance was really appreciated and treasured, for without her help, sure I would have faced many difficulties throughout all the report completion.

Last but not least, my utmost gratitude also goes to my family especially my parents for their support and motivation given to me. Not forgetting also, all my friends who have helped me through thick and thin in accomplishing this project.

Thank you and best regards,
Nurfatihah Khalid

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

PID	Propotional,Integral,Derivative
DOF	Degree of Freedom
DC	Direct Current
emf	electromagnetic force
LQR	Linear Quadratic Regulator

CHAPTER 1

INTRODUCTION

This chapter aims to introduce and explain the project entitled “Design and Analysis of Controller for a Robotic Arm Manipulator via State Space Approach”. A background about this project is given, followed by statement of the problems to be addressed and lastly the objectives and scope of the work.

1.1. Background of Study

State space is becoming more and more popular and their use is growing day by day. The main reasons for this are due to its stability and adaptability. In addition, the state space representation provides a convenient and compact way to model and analyze system with multiple inputs and multiple outputs.

State space is suitable in designing controller for robotic arm manipulator where it is generally invented to perform a desired series of movement. The arm are frequently jointed or articulated so that it can be moved within the working volume of the arm. Motors or actuator are used to affect the motion of jointed arm sections. The operation of this motor and actuators are accomplished via computational modeling whereby controller is being used to guide arm manipulator to various position. In practice, analytical model of a robotic arm manipulator is first build up to develop state space models of the system and to be used for control design exercise.

1.2. Problem Statement

In the manufacturing industry, conventional control approaches are widely used for controlling a robot manipulator. Conventional control possesses several drawbacks, for example PID controllers are not adaptive and not robust. This project is looking at modern control approach for plant control which is expected to be better in terms of the system's controllability and stability.

1.3. Objectives

The objectives of the study are as below:

- To apply knowledge in state space for designing and analyzing of robotic arm controller.
- To do simulation and performance evaluation of robotic arm manipulator controller.

1.4. Scope of Study

The scopes of study in this project are as below:

- Applying the concepts of modern control to robotic control.
- Understanding on the system dynamic of robotic arm manipulator and equation associated to it.
- Design, simulation and analysis of a controller and observer for robotic arm manipulator via state space.
- Design and analysis of optimal control for robotic arm manipulator.

CHAPTER 2

LITERATURE REVIEW

This chapter explains the concepts and theories involved in this project. It also justifies some of the decision that has been made in executing this project.

2.1 Robotic Arm Manipulator

2.1.1 Need of a Controller

Figure 1: 6 DOF arm manipulator

The most common type of robotic arm manipulator used in the industry is the serial six degrees of freedom as shown in Figure 1 (6 DOF).The dynamic of a serial 6 DOF arm manipulator is moving rapidly as the robot arm is moving fast within its working range. Besides, the structure of this arm manipulator itself is elastic and mostly the gears have nonlinearities in form of backlash. Therefore, it is quite difficult to have a very precise control for the robotic arm manipulator [1].In order to have better control, the controller need to be designed such that the actual position is closed to the desired reference even with disturbances like motor torque and tool disturbances acting on the tool occurred.

2.1.2 Control of Robotic Arm Manipulator

The use of actuator is important to control the motion of robotic manipulator where it will drive the joints of the manipulator. Several types of actuator found in industry namely electrical, hydraulic and pneumatic actuator. Hydraulic and pneumatic are less widely used compare to electrical actuator due to some of their shortcomings.

In hydraulic actuators, incompressible fluids are used to get resulting pressure to drive the joints of the manipulator. The main disadvantage of this actuator is that they exhibit highly nonlinear behaviour due to the compressibility of the fluid and due to the leakage losses. [2].Pneumatic actuators use a compressible fluid to drive the piston. Hence, pneumatic actuators tend to have time delay due to the compressibility of the fluid [2].

The most widely used actuator is electrical actuator where it utilizes stepper motors and DC motors. The DC motor consists of two wire winding, one wrapped around the rotating armature (armature circuit) and the other wrapped around a fixed rotor (field circuit) that later produce a steady magnetic field.[2]. Instead of field control, many favour armature control since it allows the speed to vary in wider range than in case for field control [2, 5].

2.1.3 Summary of Controller Designed for Robotic Arm Manipulator

There exist many trends in robotic research whereby some suggest using neural network as the controller to substitute the conventional controller [2]. Neural network will be trained as time goes by and will eventually become a controller. After the training is completed, the previous controller can be removed and neural network will be in charge. However, neural network possess several drawbacks as:

1. It requires large diversity of training for real world application.
2. Should be avoided from over train.
3. For a complex plant, it is difficult to obtain the training set.

Also, many have suggested the application of fuzzy logic as the controller. [3] has studied a fuzzy logic controller to control wheeled mobile robot in a robot soccer game.

However, this control method requires predefined and fixed fuzzy rules which later reduce the flexibility and numerical processing capability of the controller [4].[4] also has come out with controller based on state feedback theory and applied a model free robust control approach for the trajectory tracking of PUMA 560 robot.

2.2 State Space Controller

2.2.1 Concepts/Theories

The state space model represents a physical system as *n* first order coupled differential equations. This form is better suited for computer simulation than an n^{th} order input-output differential equation**.**

2.2.2 State Space Representation

A system is represented in state space by the following equations:

$$\begin{aligned} \dot{x} &= Ax + Bu \quad (state\ equation) \\ y &= Cx + Du \quad (output\ equation) \end{aligned}$$

State equation can be solved for the state variables, *x* whereas output equation is used to calculate any other system variables [5] .The choice of variable for a given system is not unique where its requirement is that they must be linearly independent and minimum number of them is chosen [5].Here, a set of variable is said to be linearly independent if none of the variables can be written as a linear combination of the others.

2.2.3 Analysing System Representation

Analysing the system representation is important as to demonstrate the system application and evaluate the system response prior to inserting it into closed-loop system. Several technique found to perform analysis is state space representation with the first one is evaluating the poles and zeroes. The poles of a system are the values of 's' which make the denominator of the transfer function equal to zero or equivalently they are the eigenvalues of the 'A' matrix of the state space model. They determine the stability of the system. The zeros of a system are the values of’s’ which make the numerator of the transfer function equal to zero [6].System response also can be used to illustrate how the system responds when an input is suddenly applied. Other parameters related to this system response are settling time and rise time. Rise time and settling time yield information about the speed of the transient response. This information shall help in determining the speed and the nature of the system as do or do not degrade the performance of the system. [6].

2.2.4 Modelling of Robotic Arm Manipulator

Robotic arm manipulators are composed of link connected by joints to form a kinematics chain. Each joint represents the interconnection between two links [7].A useful robotic arm manipulator is one that is able to control its movement and the interactive force and torque between the robot and its environment. To control the robotic arm manipulator, first a mathematical model is required. The mathematical model of a robot is obtained from the basic physical law governing its movement [8].

Here, a simple model of one link robot manipulator as shown in Figure 2 is to be considered [9].

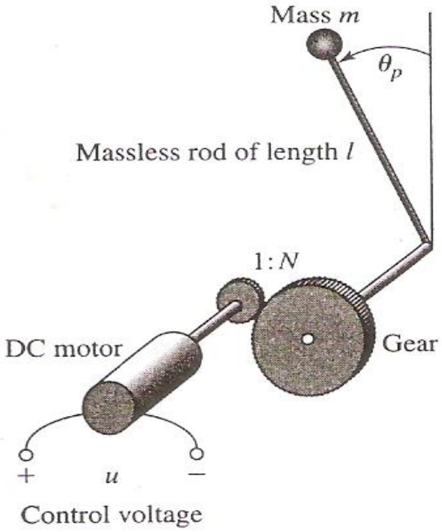


Figure 2: One link manipulator controlled by a DC motor via gear

The motion of the robot arm is controlled by a DC motor via a gear. A point of mass *m* attached to the end of a massless rod of length *l*.is modelled as the arm. Thus, the arm inertia is $I_a \equiv ml^2$ .Few assumptions were made whereby we said that:

1. The gear train has no backlash, and all connecting shaft are rigid.
2. The motor moment of inertia is negligible compared with that of the robot arm.

Based on Figure 1 also, it can be seen that counter clockwise rotation of the arm is defined as positive and clockwise rotation of the arm is defined as negative. Meanwhile, counter clockwise rotation of the motor shaft is defined as negative and clockwise rotation of the shaft is defined as positive. The schematic of DC motor which is mainly armature controlled is shown in Figure 3.

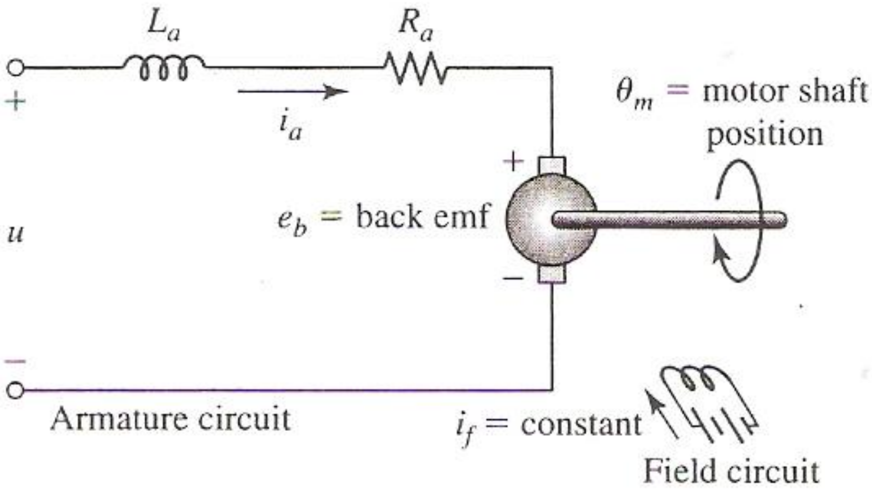


Figure 3: Schematic of an armature controlled DC motor

Based on Figure 3 shown, noticed that the torque delivered by the motor is:

$$T_m = K_m i_a \tag{1}$$

$K_m =$ motor torque constant
$i_a =$ armature current

Next, to indicate N gear ratio, the designation is said to be,

$$\frac{\theta_p}{\theta_m} = \frac{radius\, of\ motor\ gear}{radius\, of\ arm\, gear} = \frac{number\, of\ teeth\ of\ motor\ gear}{number\, of\ teeth\ of\ arm\ gear} = \frac{1}{N} \tag{2}$$

This is mainly because the gears are in contact and therefore,

$$\theta_p \times radius\, of\ \ arm\, gear = \theta_m \times radius\, of\ motor\ gear \tag{3}$$

Meanwhile, the radius of the gears is proportional to their number of teeth. Thus, the work done by the gears must be equal. Took T_p as the torque applied to the robot arm and this resulted to,

$$T_p \theta_p = T_m \theta_m \tag{4}$$

Therefore, the torque applied to the pendulum is,

$$T_p = NT_m = NK_m i_a \tag{5}$$

By using Newton’s second law, the equation to model the arm dynamics can be wrote as,

$$I_a \frac{d^2\theta_p}{dt^2} = mgl\sin\theta_p + T_p \tag{6}$$

Further substitution into above expression for $I_a = ml^2$ and equation (5) yields,

$$ml^2 \frac{d^2\theta_p}{dt^2} = mgl\sin\theta_p + NK_m i_a \tag{7}$$

The gravitational constant is $9.8\,m/s^2$.Kirchhoff's Voltage Law is applied to the armature circuit thus produce,

$$L_a \frac{di_a}{dt} + R_a i_a + K_b N \frac{d\theta_p}{dt} = u \tag{8}$$

where K_b is the back emf constant. Next, we assume that $L_a \approx 0$, thus yields,

$$R_a i_a + K_b N \frac{d\theta_p}{dt} = u \tag{9}$$

Computing i_a from above equation and substituting into equation (7) produce,

$$ml^2 \frac{d^2\theta_p}{dt^2} = mgl\sin\theta_p + NK_m \left(\frac{u}{R_a} - \frac{K_b N \frac{d\theta_p}{dt}}{R_a} \right) \tag{10}$$

Finally, a state space model of one link robotic arm manipulator can be constructed. State and output variables were chosen as,

$$x_1 = \theta_p \quad (arm\ joint\ position)\ , x_2 = \frac{d\theta_p}{dt} = \omega_p\ (joint\ arm\ velocity)\ , y = x_1 \tag{11}$$

Following simple state space model of the robotic arm manipulator is obtained using previous equation which later produce,

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{g}{l}\sin x_1 - \frac{K_b K_m N^2}{ml^2 R_a} x_2 + \frac{NK_m}{ml^2 R_a} u \end{bmatrix}$$
$$y = x_1 \tag{12}$$

Parameters for robotic arm manipulator were chosen as:

$$l = 1m \ , \ m = 1kg \ , \ N = 10 \ , \ K_m = 0.1 Nm/A \ , \ K_b = 0.1 V\sec/rad \ , \ R_a = 1\Omega \tag{13}$$

With those parameters chosen above, the robotic arm manipulator model took the form;

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ 9.8\sin x_1 - x_2 + u \end{bmatrix}$$
$$y = x_1 \tag{14}$$

The lineraized model has the form:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 9.8 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$
$$y = x_1 \tag{15}$$

where the output of the system is the arm joint position. The nomenclature of the system modeling is as indicated in Table 1.

Table 1: Nomenclature of the System Modeling

NOMENCLATURE			
I_a	Arm inertia	R_a	Armature resistance
m	mass	L_a	Armature inductance
l	Mass less rod length	ω_p	Arm joint velocity
T_m	Motor torque	i_a	Armature current
K_m	Motor torque constant	i_f	Field current
i_a	Armature current	K_b	Back emf constant
θ_p	Arm joint position	e_b	Back emf
θ_m	Motor shaft position		
T_p	Applied torque		
g	Gravitational constant		
N	Gear ratio		

2.2.5 Feedback and Feed forward Controller

In control system, there are basically two types of control to be designed namely feed forward and feedback. . The input to a feedback controller is the same as what it is trying to control - the controlled variable is "fed back" into the controller [5].It will measure the controlled variable and adjust the output based on its desired set points. However, feedback controller usually results in intermediate period where the controlled variable is not at desired set points.Therefore, then come the role of the feed forward controller which is to avoid the slowness of feedback control. Using feed forward control, the disturbances are measured and counted for before they have time to affect the system.

2.3 Observer

2.3.1 Concept/Theories

In reality, for state feedback design, all the states are seldom available for measurements. Therefore, given only measurements of some specified outputs of a dynamical system, all the states can be reconstructed using an observer if the system satisfies a property known as observability.Observability means that there are enough independent outputs to be able to determine what is going on with the full internal state of the system. It indicates that the chosen measurement scheme is a suitable one [10]. [11] cited out the unreliability of having a disturbance observer for robotic arm manipulator. The main function of this observer is to reduce external unknown or uncertain disturbance torques without the use of an additional sensor. Robotic manipulators work in a dynamic highly uncertain environment. For this application, rather than providing control, the disturbance observer shall focus more on trajectory planning and monitoring. The problem with this type of observer is such that a multilink robotic manipulator is a highly nonlinear and coupled system. Thus, the validity of using linear analysis and synthesis techniques may be doubtful [11].

2.3.2 Observer Design

The choice of observer is definitely independent on the choice of controller. There are two possibilities in implementing an observer. We can choose to design either a full order state observer or a reduced order observer. In full order state observer, we can observe all state variables of the system regardless of whether some state variables are already available for direct measurement. Meanwhile, the as opposed to full state observer, reduced order observer estimates fewer than *n* state variables where *n* is the dimension of the state vector [9].The disadvantage of this solution is that the measured state can be affected by the measured noise even though we surely can have less calculation in our observer. Overall, in both design, the main idea is to have an estimator with a dynamic quicker than the controlled plant [12].This can be achieved by placing the poles of the observer two to five times faster than the controller poles [9].However, if the sensor noise is considerable, we may choose the observer poles to be slower so that the bandwidth will become lower and smoothen the noise [9].

2.3.3 Summary of Observers Designed

[13] introduced an acceleration-based state observer for robot manipulators with elastic joints. They presented an observer which uses only motor position sensing, together with accelerometers suitably mounted on the links of the robot arm. The main advantage of this system is said that the error dynamics on the estimated state is independent from the dynamic parameters of the robot links and can be tuned with decentralized linear technique. The hardly available sensor to provide measurement for motor velocity had motivated the design of this state observer to replace the missing sensors.

Some other use of observer aside from robotic isapproximately to recover the state from its partial observation for ecological monitoring. Monitoring of ecological system is one of the major issues in ecosystem research. In many cases, state monitoring of a complex ecological system consists in observation (measurement) of certain state variables, and the whole state process has to be determined from the observed data. The solution proposed in [14] is the design of an observer system, which shall makes it possible to approximately recover the state from its partial observation.

2.4 Quadratic Optimal Design

The basic use of optimal design is to choose an input control so that the performance of the system is optimum with respect to some performance criterion. To optimize particular system, performance measure is needed where it is mathematically expressed in term of cost function. Therefore, the basic goal is to find a control function, u that will minimize the cost function. The system that is able to minimize the selected cost function or the performance index

$$J = \int_0^\infty (x^T\,Qx + u^T Ru)dt \tag{16}$$

is by definition, optimal. The most important point is that the design based on this quadratic performance index yields a stable control system.

[15] had presented a study on the development of optimal control for input tracking and vibration suppression of a flexible joint manipulator.A single-link flexible joint manipulator is considered and to study the effectiveness of the controllers, LQR controller is developed for its tip angular position control.The performances of the control schemes have been evaluated in terms of input tracking capability, level of vibration reduction, time response specifications and robustness. For their study, acceptable performance in input tracking control and vibration suppression has been achieved.

CHAPTER 3

METHODOLOGY

This chapter discussed the process involved in carrying out the study and also material used for its completion.

3.1 Procedure Identification

The project flow chart depicted in Figure 4 summarized the steps undertaken during the execution of this study.

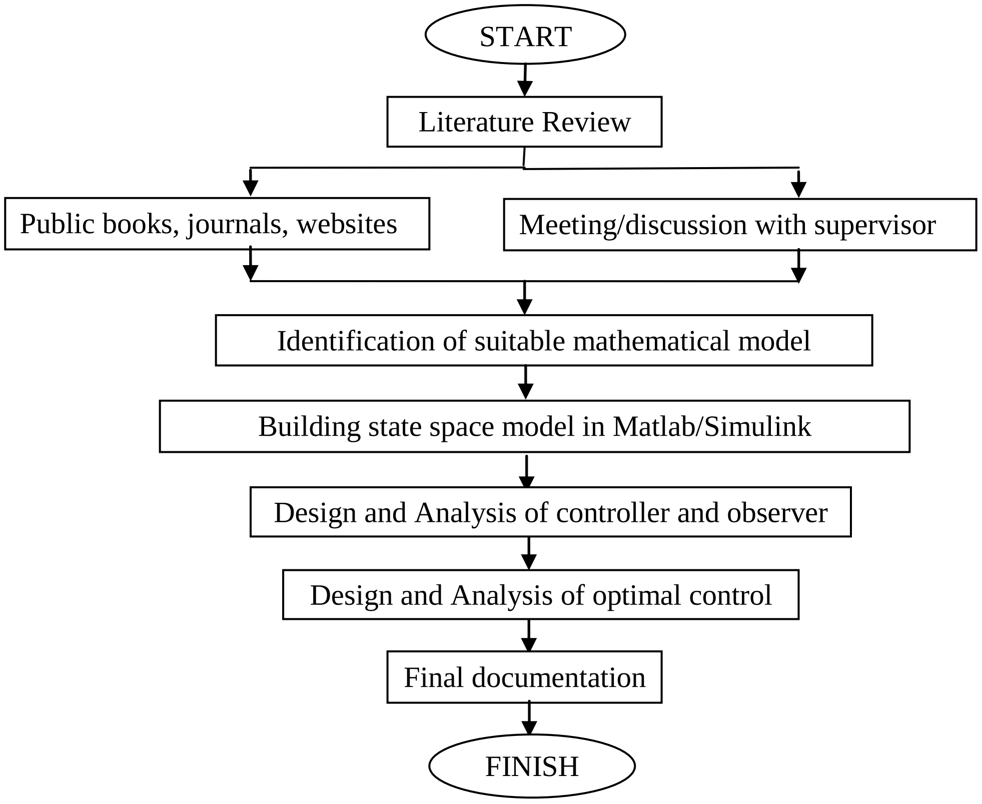


Figure 4: Project Flow Chart

3.2 Project Works

Based on Figure 4, in order to achieve the objectives of this study, research had been done on dynamic of robotic arm manipulator and state space representation. Thorough researches were done through internet, public books and journals to collect all available information.

All the accumulated information is analyzed to determine the most accurate equation to be used for robotic arm manipulator modeling via state space approach. Suitable equation is needed to make sure the best controller is produced.

By using the pre-determined equation, a state space model is built in MATLAB/Simulink with Control System tool box. Simulation and analysis of the system is made to analyze the system performance of the equation.

Later, the controller which consisted of state feedback and feed forward was designed. To complete the control system, observer such as full state order observer and reduced order observer is designed and evaluated for its performance. Also, the study includes the optimal control of the system. The end result is expected to be better than previous conventional control.

A Gantt chart is prepared for the completion and time management of this study based on the academic schedule and FYP guidelines (Please refer to APPENDIX 1).

3.3 Tools and Equipment Used

The tool required in this project is Matlab/SIMULINK with Control System Toolbox. This tool is used to perform simulation of the model designed throughout this project.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Modelling of the System

The equation (15) stated in Chapter 2 is to be considered as the robotic arm manipulator system. The linearized version of the equation and to be further evaluated is as shown in equation (17).

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 9.8 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

$$y = x_1 \text{ (arm joint position)}$$

$$x_2 = \dot{x}_1 \text{ (joint arm velocity)} \qquad (17)$$

The eigenvalues of the system is at -3.6702 and 2.6702. (Please refer to APPENDIX 2 for the calculations)

Also, several advantages of this model were acknowledged where:

1. The motion of the robot arm is controlled by a DC motor via gear. Thus, being electrically actuated, the model is certainly cleaner as it does not contribute to any fluid leaking and mostly cheaper.

2. The model used is also an armature controlled instead of field control. Thus, the field control current i_f is kept constant and T_m is controlled by varying the armature current i_a.Therefore, this allow the speed to vary in a wider range than in the case of field control.

4.2. Building State Space Model in MATLAB

Based on the equation predetermined, the state space model is built in Simulink as in Figure 5:

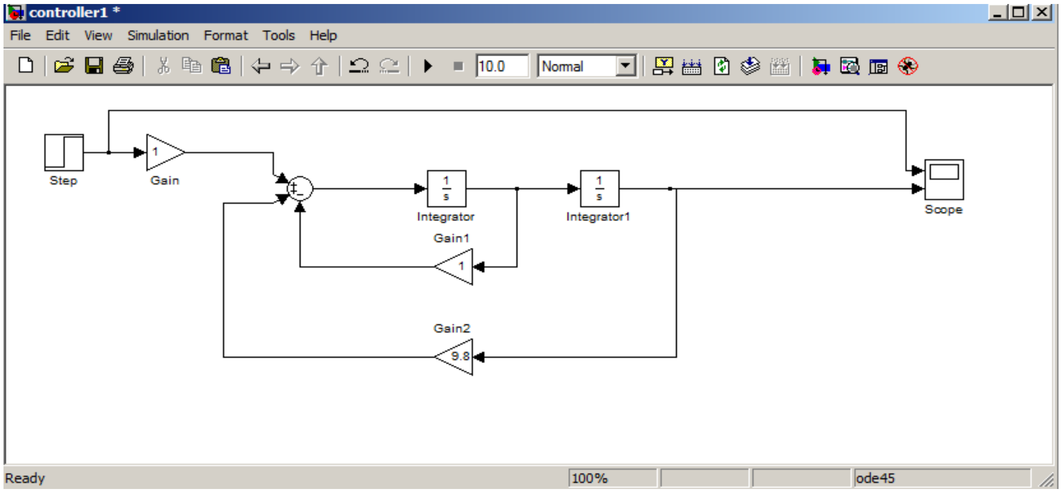


Figure 5: Model built in Simulink

The output of state space representation is fed to a scope for monitoring purpose. A unit step input as in Figure 6 is applied to the system and output is monitored:

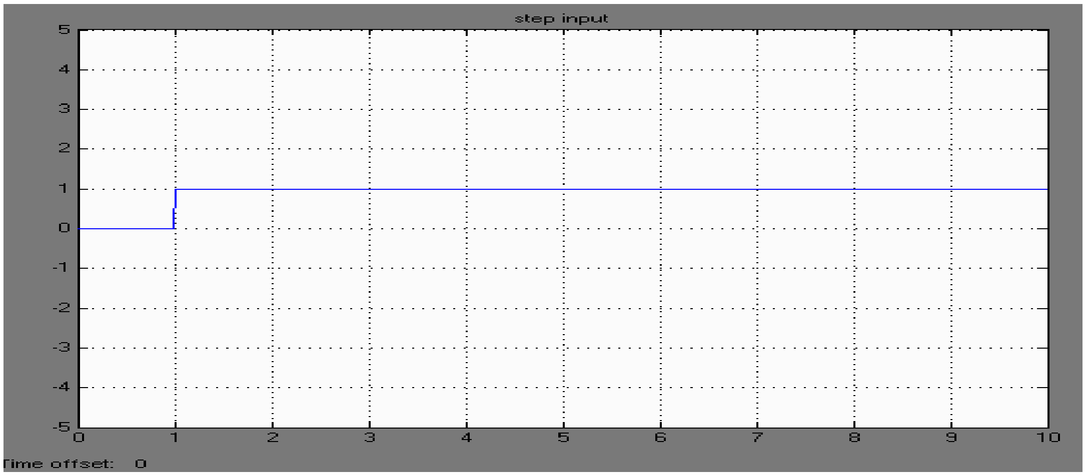


Figure 6: A unit step input applied to system

4.3. Analysis of the System

The output response of the state space representation when a unit step input is applied is as shown in Figure 7:

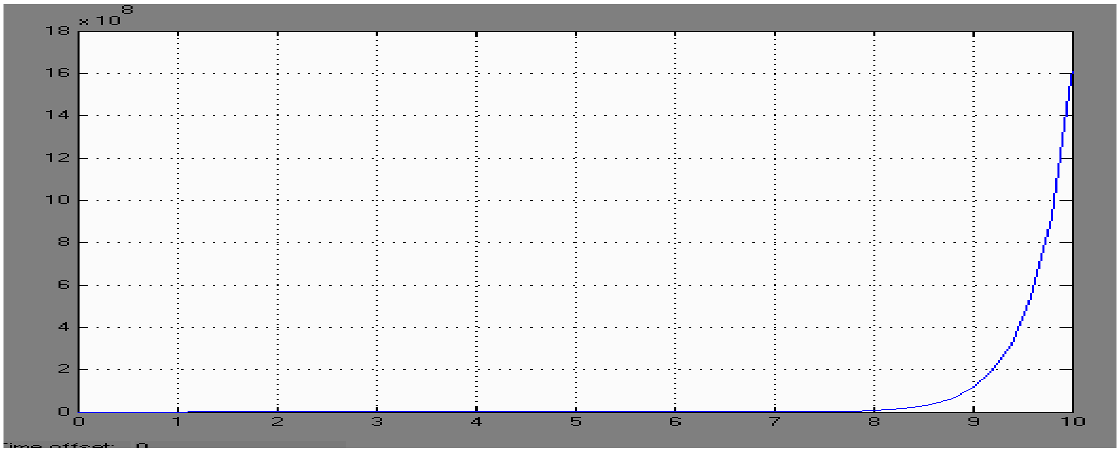
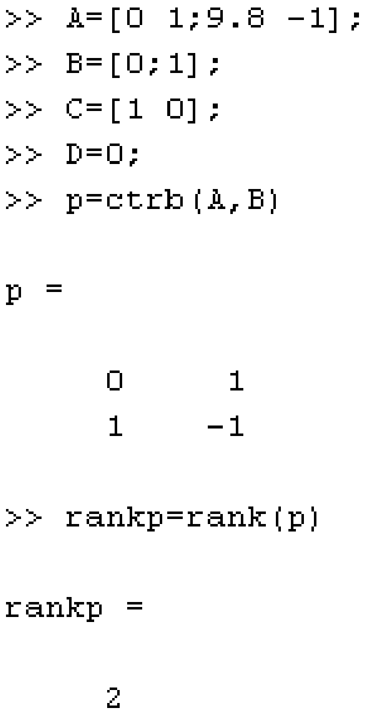


Figure 7: The output response of the system

As can be seen in Figure 7, the system is unstable in open loop. It is obvious from this plot that some sort of control will have to be designed to improve the dynamics of the system. However, to make sure that the state feedback can be designed, the controllability and observability of the system need to be examined using Matlab.

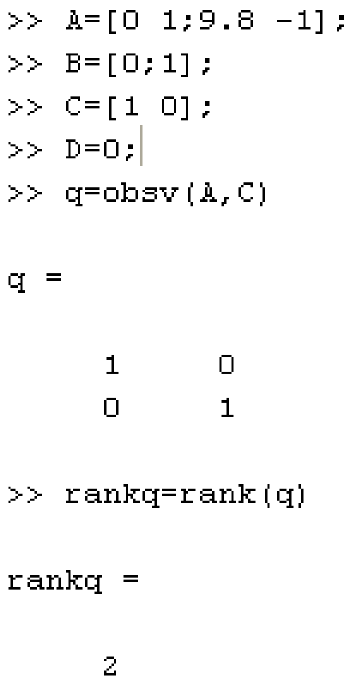
The controllability of the system is important as it provides that an input is available and brings any initial state to any desired final state. A system is found to be controllable if and only if the state representation has n x m matrix of $M_c = \left[B\,AB\,A^2B....A^{n-1}B\right]$ with rank n. Check on system controllability is performed as:



```
>> A=[0 1;9.8 -1];
>> B=[0;1];
>> C=[1 0];
>> D=0;
>> p=ctrb(A,B)

p =

     0     1
     1    -1

>> rankp=rank(p)

rankp =

     2
```

The system is of rank 2 and is found to be controllable.

On the other hand, the observability is crucial as knowing an output trajectory provide enough information to predict the initial state of the system. A system is found to be observable if and only if the state representation has n x m matrix of $M_o = \left[C^T\ A^T C^T\ A^{2T} C^T \ldots . A^{(n-1)T} C^T\right]$ with rank n.Check on the system observability is performed as :



```
>> A=[0 1;9.8 -1];
>> B=[0;1];
>> C=[1 0];
>> D=0;
>> q=obsv(A,C)

q =

     1     0
     0     1

>> rankq=rank(q)

rankq =

     2
```

The system is of rank 2 and is found to be observable.

4.4. Controller Design

There are basically two types of controller designed for the system which is feedback and feed forward controller. The first step to design the controller is to build the whole robotic arm manipulator system in Simulink.

4.4.1 Regulator System

Next, is to solve for regulator problem by designing state feedback gain K. For that, a step input is applied to the system as shown in Figure 8.

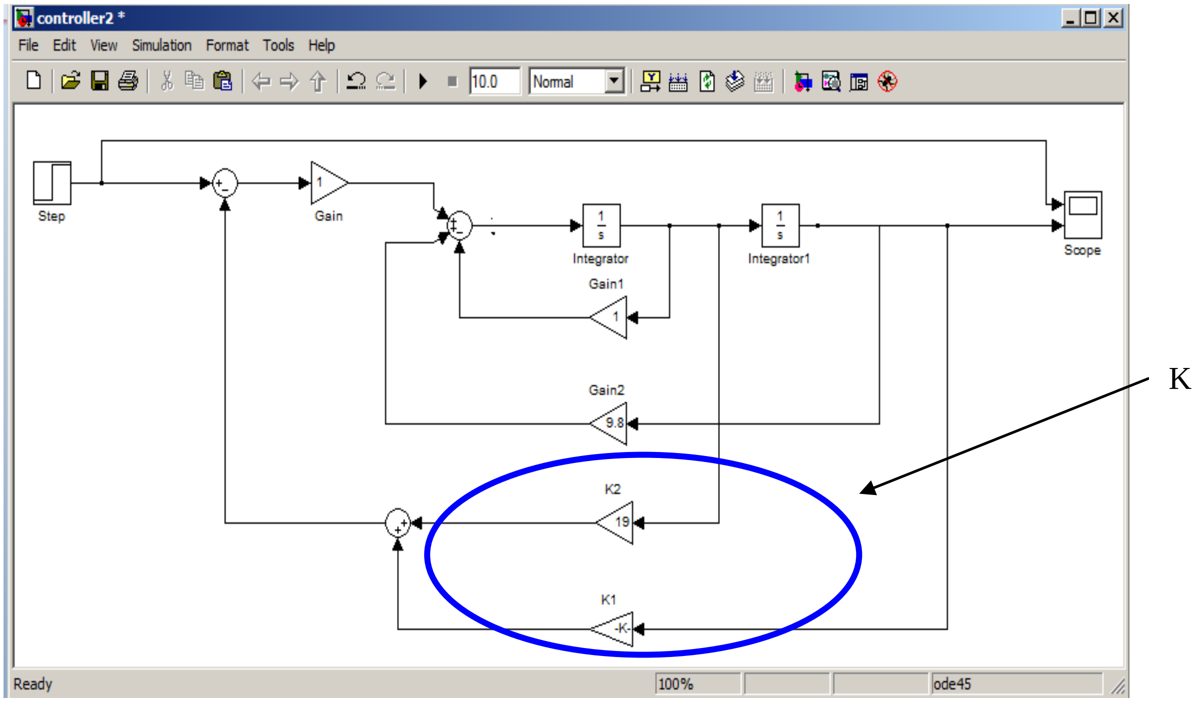
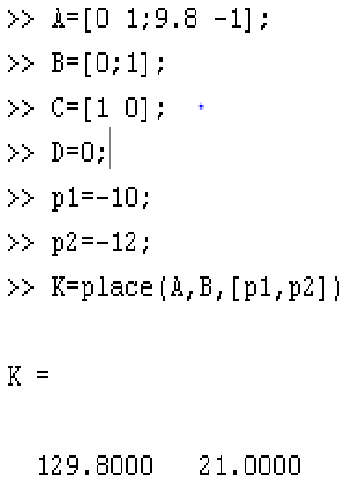


Figure 8: Robotic Arm Manipulator System using Subsystem Block with state feedback

The value of feedback gain K is found as below by using pole placement method where two cases of poles location are considered:

i. Case 1: p1= -10 , p2= -12 (Real Location)



```
>> A=[0 1;9.8 -1];
>> B=[0;1];
>> C=[1 0];
>> D=0;
>> p1=-10;
>> p2=-12;
>> K=place(A,B,[p1,p2])

K =

  129.8000   21.0000
```

The applied input and simulated response for state feedback is shown in Figure 9.

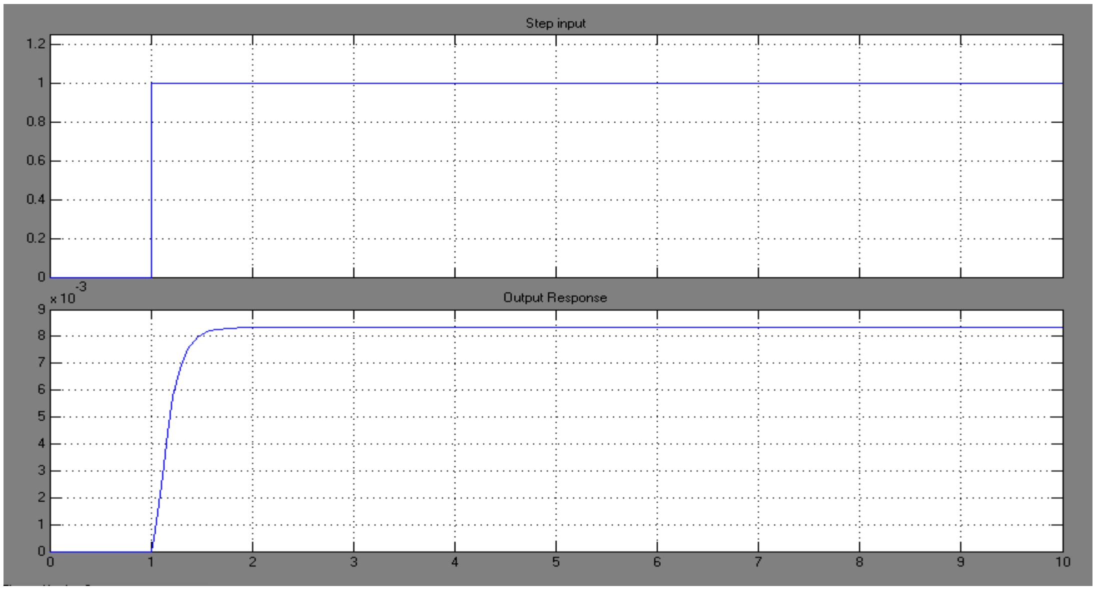
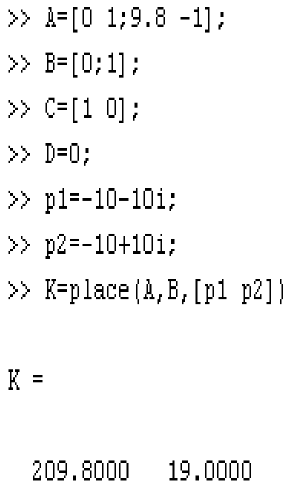


Figure 9: Step Input (top) and its Simulated Response for State Feedback (bottom).

ii. Case 2: p1=-10+10i , p2=-10-10i (Real and Imaginary Location)



```
>> A=[0 1;9.8 -1];
>> B=[0;1];
>> C=[1 0];
>> D=0;
>> p1=-10-10i;
>> p2=-10+10i;
>> K=place(A,B,[p1 p2])

K =

  209.8000   19.0000
```

The applied input and simulated response for state feedback is shown in Figure 10.

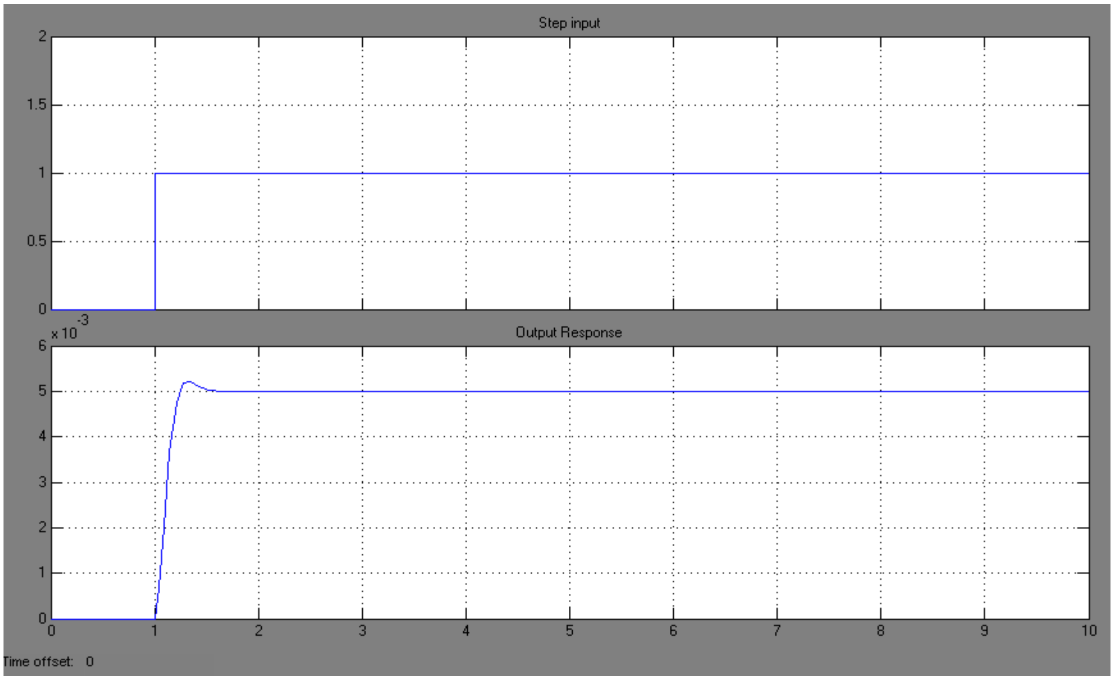


Figure 10: Step Input (top) and its Simulated Response for State Feedback (bottom)

From the result, we can see that overshoot occurred for Case 2 as shown in Figure 10.This is basically due to the imaginary pole introduced into the system .However, the significant similarities between output response in Figure 9 and Figure 10 is that both manage to achieve stability. Hence, we can say that by providing feedback to the system had improved its system characteristic and hence achieved stability despite the steady state error present. Basically, the error is due to the behavior of state feedback system which do not compare the output to the reference; instead, it compares all states multiplied by the control matrix (K*x) to the reference. Thus, to remove the steady state error, feed forward controller, N is designed and shall give zero steady state error for any input as shown in Figure 11.

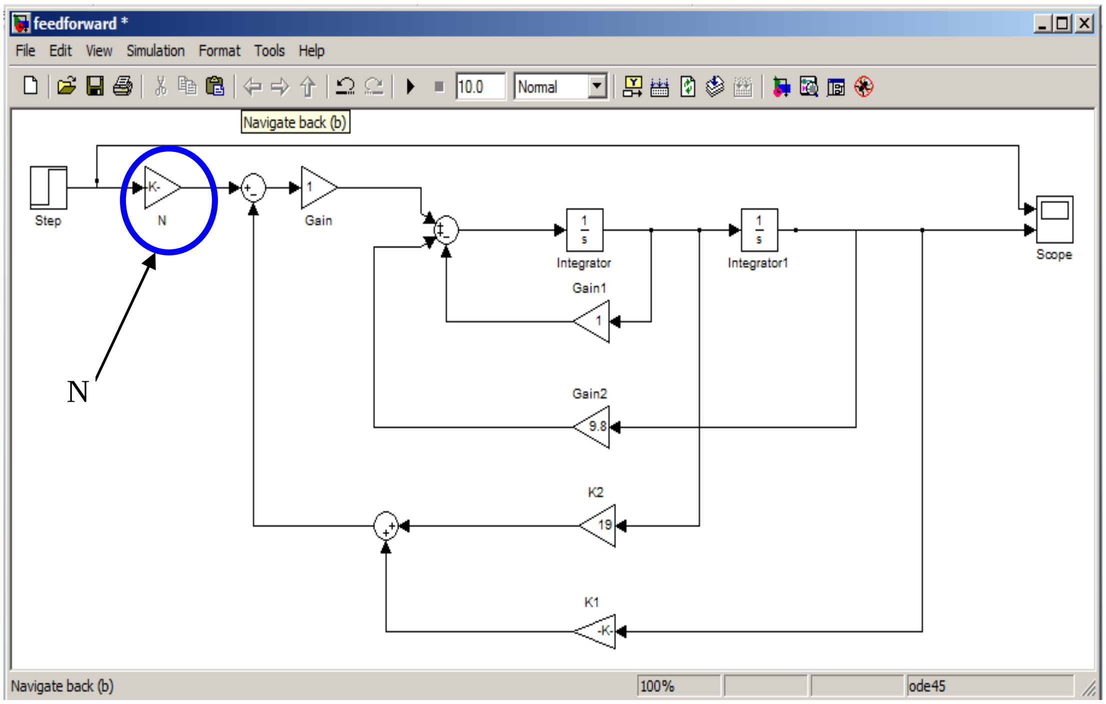


Figure 11: Robotic Arm Manipulator with State Feedback, K and Feed forward, N Gain

The value of N is then calculated to act as a feed forward gain. Based on Equation (18), we determined the value of N by first calculating the value of N_x and Nu .

$$N = N_u + KNx \tag{18}$$

The value of N_x and Nu is calculated as:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} N_x \\ N_u \end{bmatrix} = \begin{bmatrix} 0 \\ I \end{bmatrix}$$

$$\begin{bmatrix} N_x \\ N_u \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} N_x \\ N_u \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 9.8 & -1 & 1 \\ 1 & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & -9.8 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} N_x \\ N_u \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ -9.8 \end{bmatrix} \tag{19}$$

Substitute value obtain in (19) into equation (18) to find the value of feed forward, N for both cases;

i. Case 1: p1= -10 , p2= -12 (Real Location)

$$
\begin{aligned}
N &= N_u + KN_x \\
&= -9.8 + \begin{bmatrix} 129.8 & 21 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
&= -9.8 + 129.8 \\
&= 120
\end{aligned}
$$

Step input is applied to the system to observe the response as shown in Figure 12.

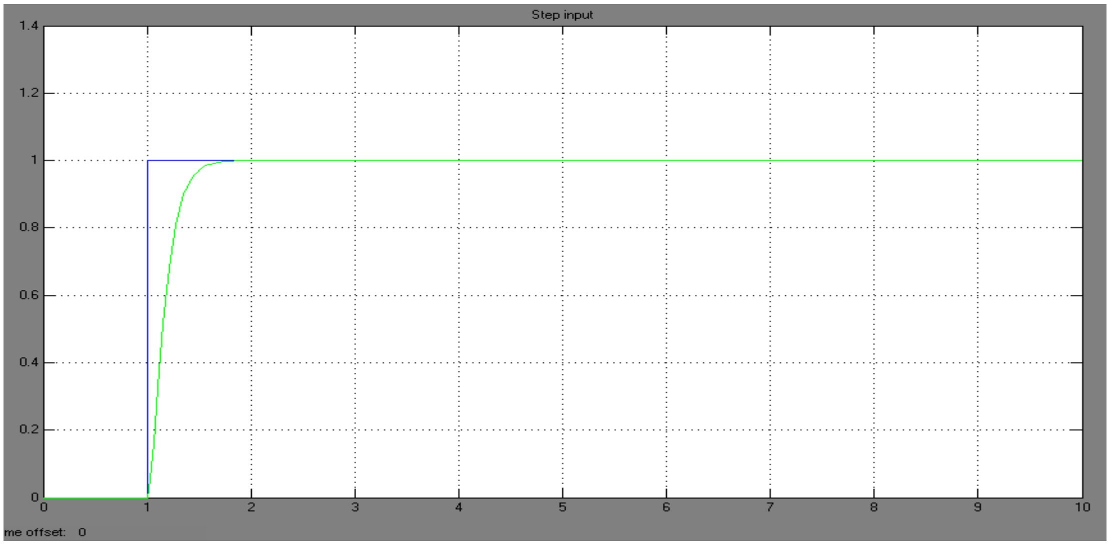


Figure 12: Step Input applied and Its Simulated Response for State Feedback and Feed forward

ii. Case 2: p1=-10+10i , p2=-10-10i (Real and Imaginary Location)

$$
\begin{aligned}
N &= N_u + KN_x \\
&= -9.8 + \begin{bmatrix} 209.8 & 19 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
&= -9.8 + 209.8 \\
&= 200
\end{aligned}
$$

Again, step input is applied to the system to observe the response as shown in Figure 13.

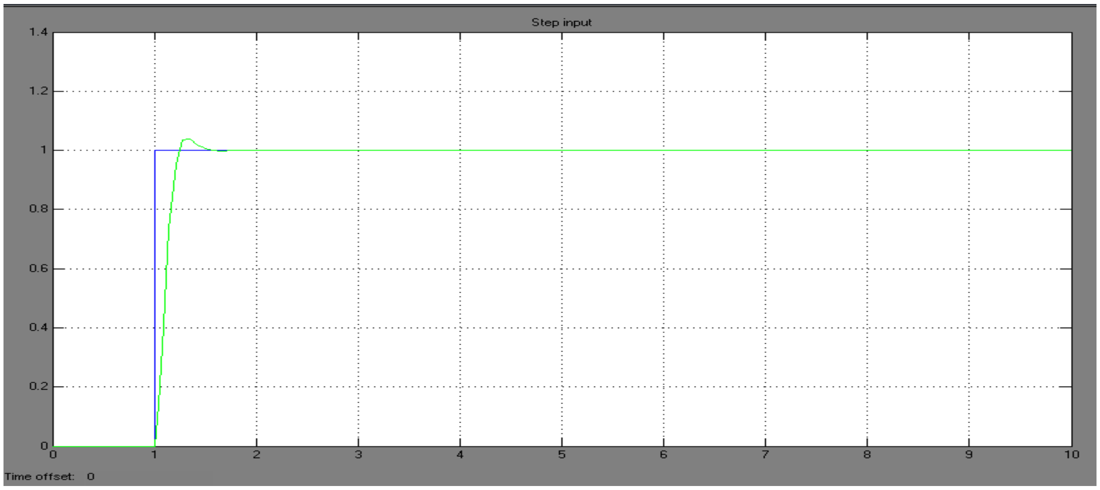


Figure 13: Step Input applied and Its Simulated Response for State Feedback and Feed forward

The simulation result in Figure 12 and 13 has demonstrated that for both cases, the presence of feed forward N had eliminated the system's offset as the system settled at 1 which is the same value applied to the step input. This indicated that feed forward N managed to scale the reference input so that the output response is equal to the reference input applied. Again, the only difference between both cases is the overshoot experienced by Case 2 as can be seen in Figure 13 mainly due to imaginary poles.

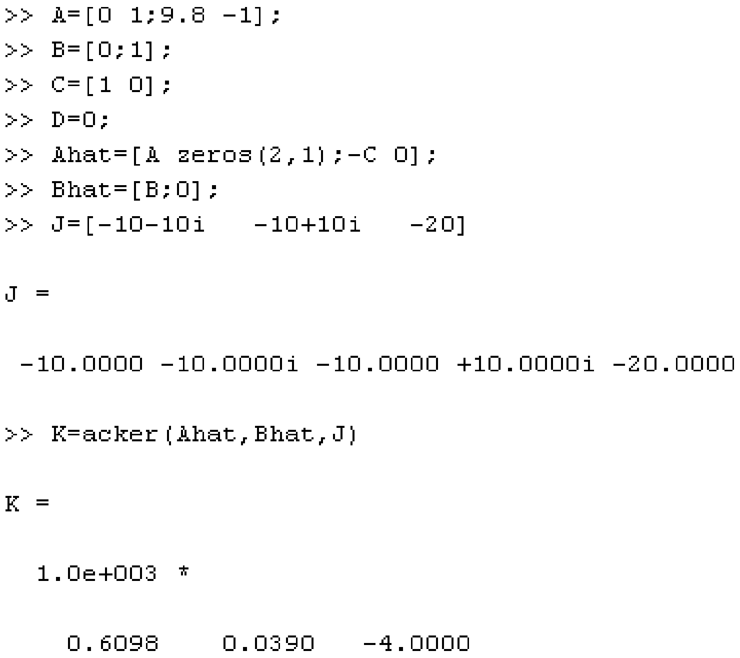
4.4.2. Servo/Tracking System

It is desired to control the positioning and movement of the robotic arm manipulator, for instance, moving the arm in a sinusoidal fashion. Therefore, tracking system is built by inserting an integrator in the feed forward path between the error comparator and the plant. Also, the output is fed back to the input as shown in Figure 14,Figure 16,Figure 18 and Figure 20.The reference input such as sinusoidal, stair, ramp up and ramp down is applied respectively to observe the robotic arm manipulator response.

` Before building the system in Simulink, the value of feedback gain and feed forward is calculated as integrator action had altered the controller form to be:

$$\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} A - BK & -BN \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} r \tag{20}$$

Then, the value of feed forward gain and feedback gain is found by using *acker* function in MATLAB:



```
>> A=[0 1;9.8 -1];
>> B=[0;1];
>> C=[1 0];
>> D=0;
>> Ahat=[A zeros(2,1);-C 0];
>> Bhat=[B;0];
>> J=[-10-10i   -10+10i   -20]

J =

 -10.0000 -10.0000i -10.0000 +10.0000i -20.0000

>> K=acker(Ahat,Bhat,J)

K =

  1.0e+003 *

    0.6098    0.0390   -4.0000
```

Thus, the value of feed forward gain is 4000 while the value of feedback gain is 609.8 and 39.0 are inserted in the system as in Figure 14,Figure 16,Figure 18 and Figure 20.

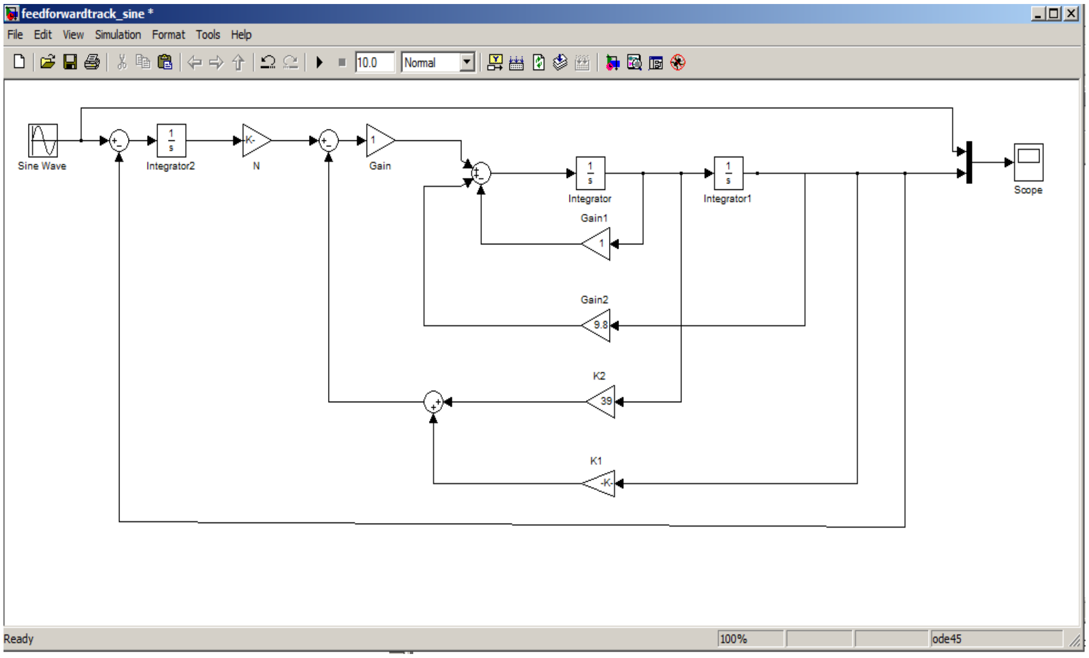


Figure 14: Tracking System with Sinusoidal Input

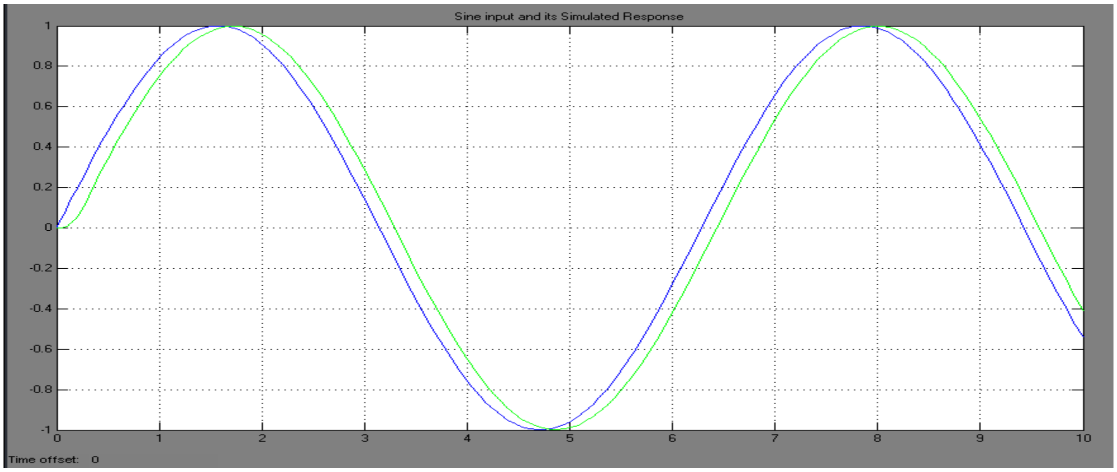


Figure 15: Sinusoidal Input applied and Its Simulated Response

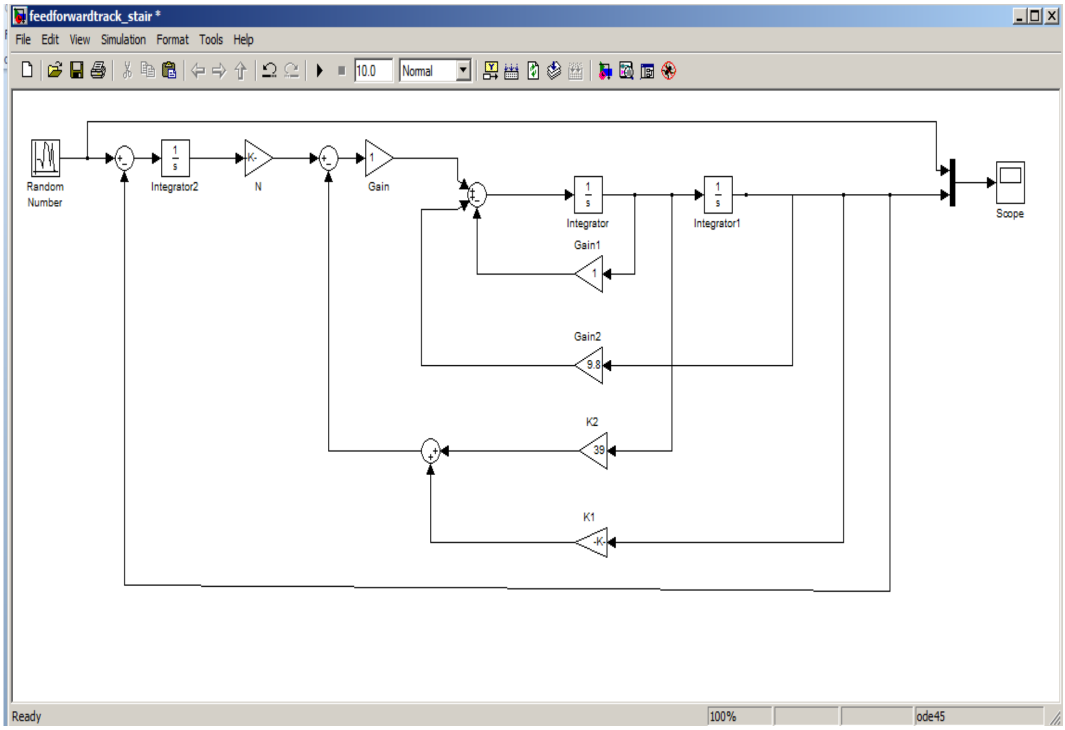


Figure 16: Tracking System with Stair Input

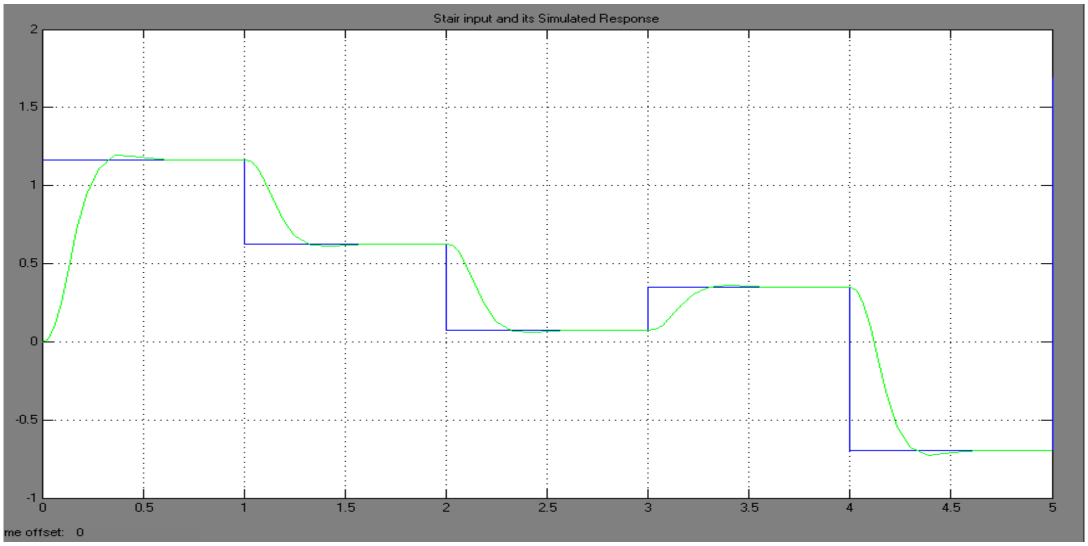


Figure 17: Stair Input Applied and Its Simulated Response

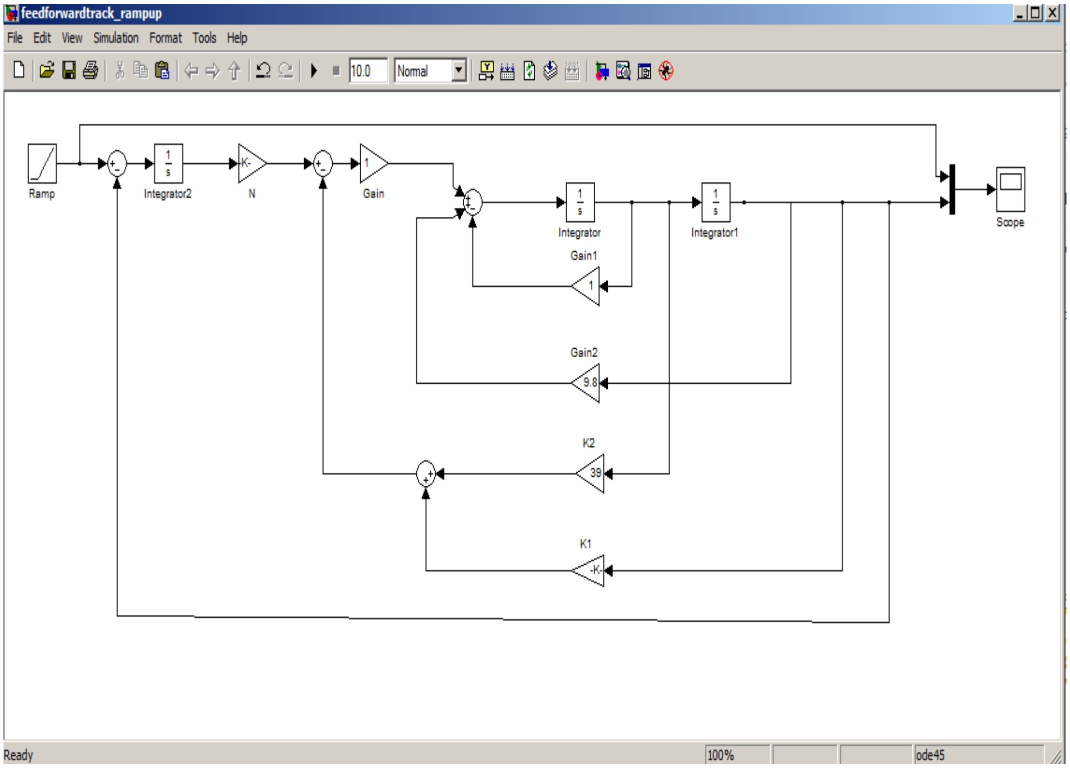


Figure 18: Tracking System with Ramp up Input

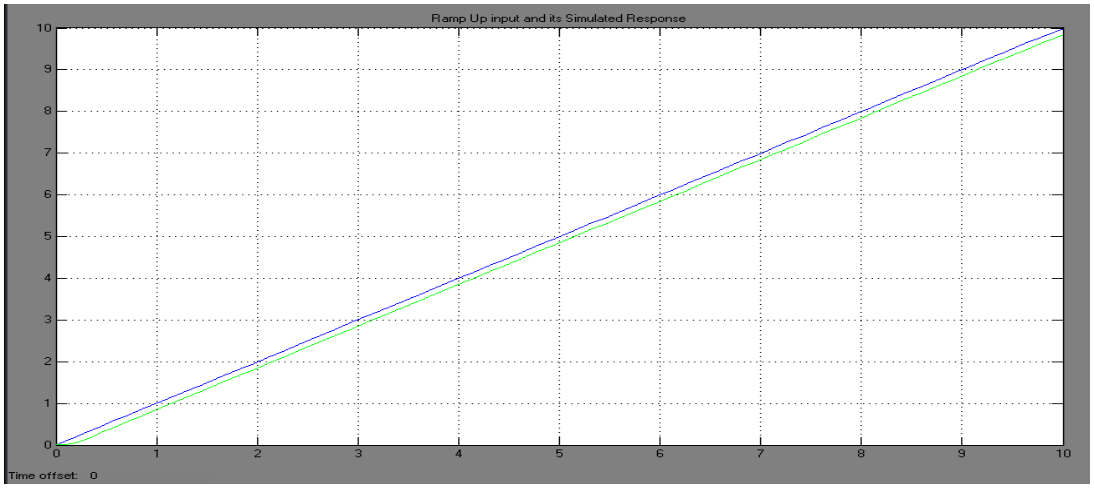


Figure 19: Ramp up Input Applied and Its Simulated Response

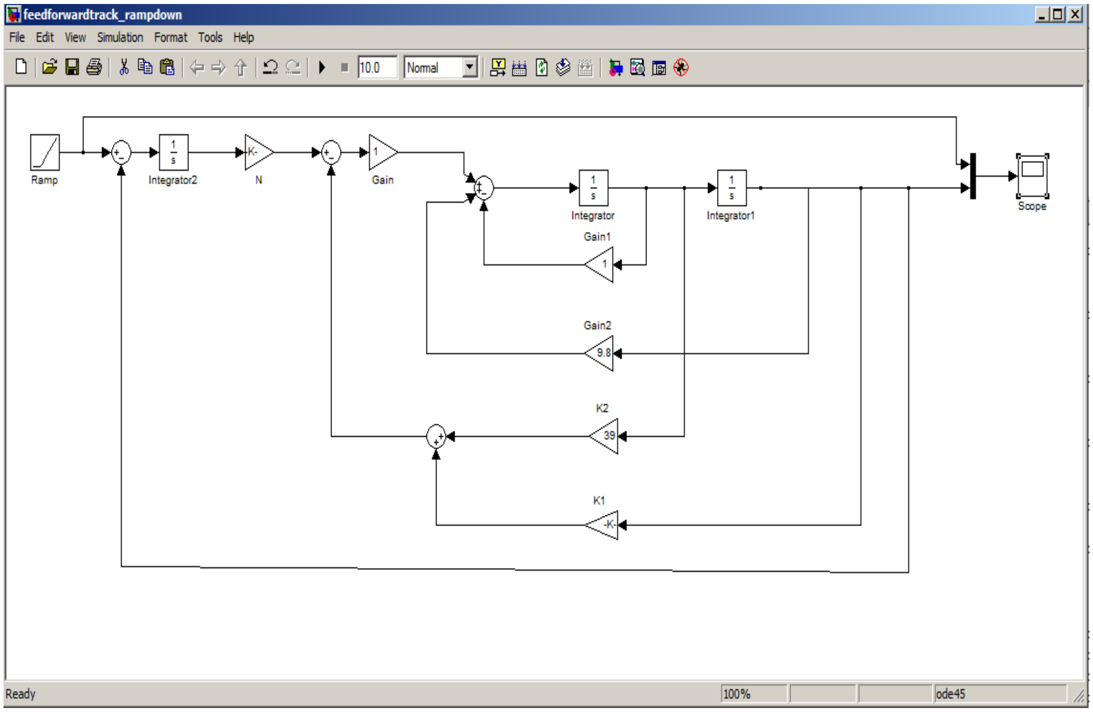


Figure 20: Tracking System with Ramp down Input

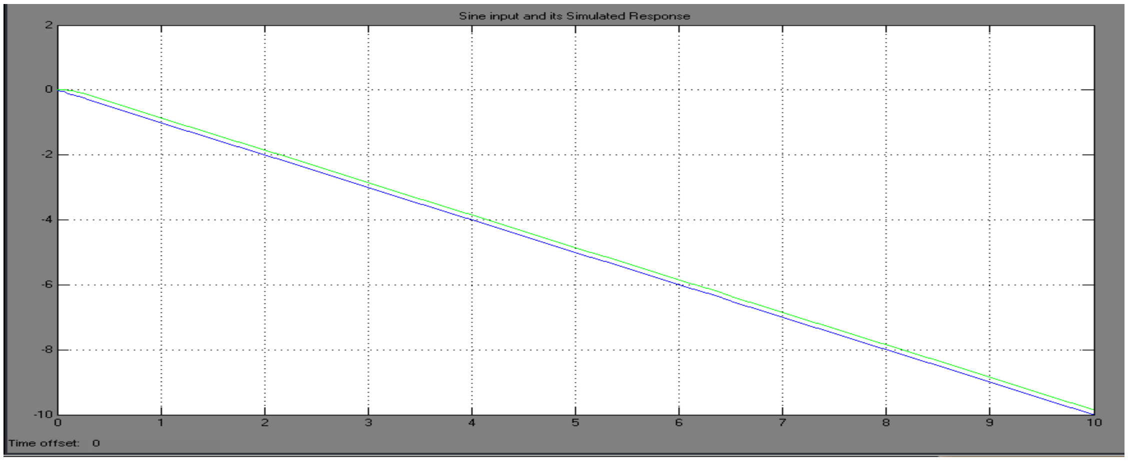


Figure 21: Ramp down Input Applied and Its Simulated Response

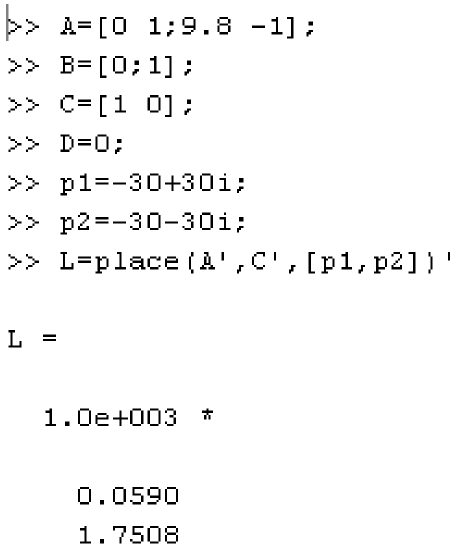
Referring to all the tracking system output response in Figure 15,Figure17,Figure 19 and Figure 21, it can be seen that it almost resemble its applied input. Therefore, we can say that by having the integrator and the fed back output would give an almost identical output response to any input applied.

4.5 State Observer

State observer is designed to estimate the unavailable state variables in the robotic arm manipulator system as in real practice, not all state is measurable. A state observer estimates the state variable based on the measurement of the output and control variables.

4.5.1 Full Order State Observer

The design process of observer started with the determination of observer gain matrix for full state observer.The poles (-30+30i and -30-30i) is chosen to obtain the observer characteristic lied on the left half plane and further from dominant pole.The observer gain L is found using pole placement method as indicated bellow:



```
>> A=[0 1;9.8 -1];
>> B=[0;1];
>> C=[1 0];
>> D=0;
>> p1=-30+30i;
>> p2=-30-30i;
>> L=place(A',C',[p1,p2])'

L =

   1.0e+003 *

    0.0590
    1.7508
```

Thus, the observer gain is found to be L= [59; 1750.8].

The observer gains, L= [59;1750.8] was inserted in the full state observer model constructed in Simulink as in Figure 22.

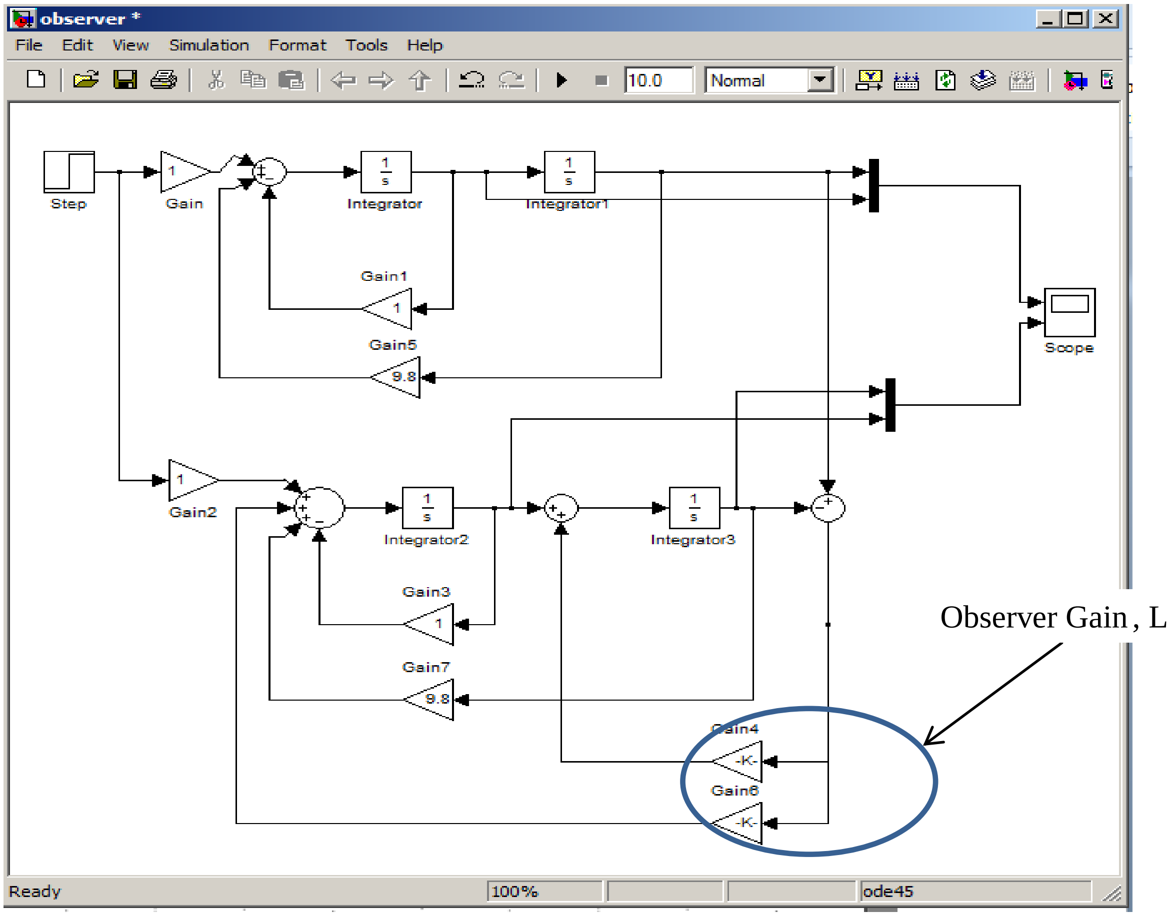


Figure 22: Block Diagram of robot arm manipulator system with full order state observer

The result of the simulated block diagram with step input is as shown in Figure 23.

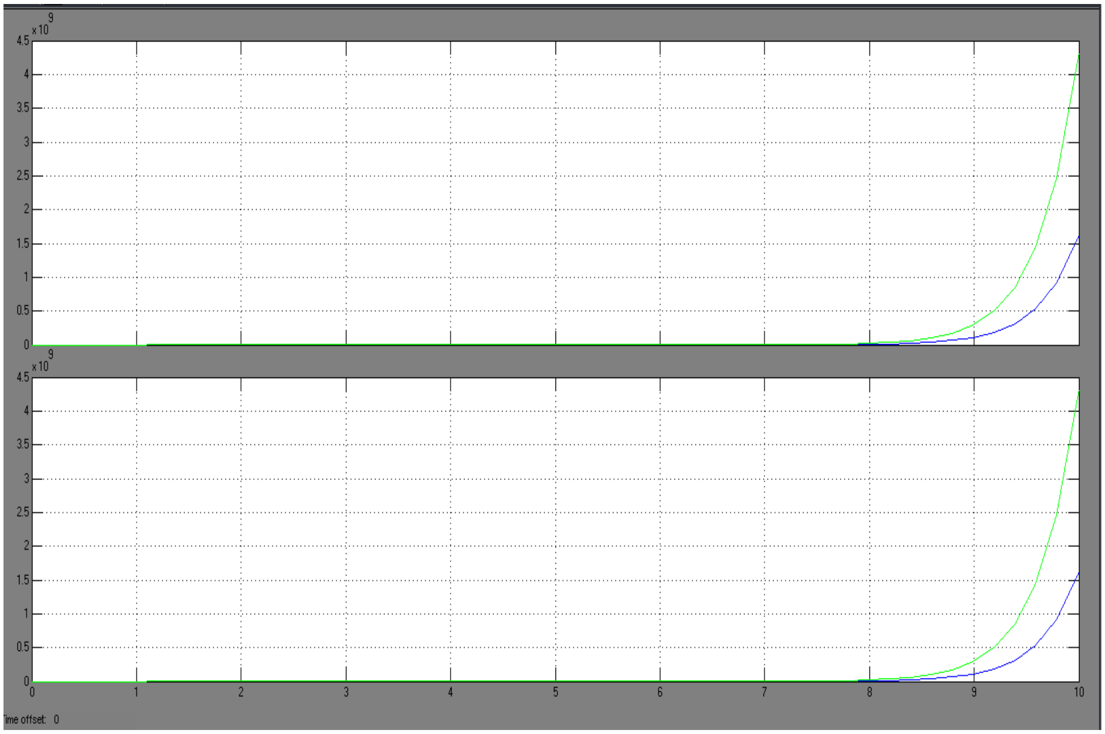


Figure 23:State of Original System(top) and Observer Measured State(bottom)

Here, we could say that for pole location of three times further than the controller poles, the observer output ($\hat{x}_1$and $\hat{x}_2$) exhibits the same behavior as the original system state (x_1 andx_2).

4.5.2 Reduced Order State Observer

The problems of having a full state observer despite behaving similarly to the original system state is, the cost of sensors used in the system.Therefore, reduced order observer is used to eliminate redundancy in full state observer as some of the state actually had already been measured,thus reducing the cost of sensors.For our case, we only need to measure x_2 as x_1 is already available.

The first stage in designing a reduced order observer is determining the new value of observer gain,L.In the design of reduced state observer , it is desirable to determine several observer gain matrices based on several different desired poles location by using pole placement method. To obtain the response of reduced order observer,observer gain was inserted in the robotic manipulator system modelled in Simulink as in Figure 24.

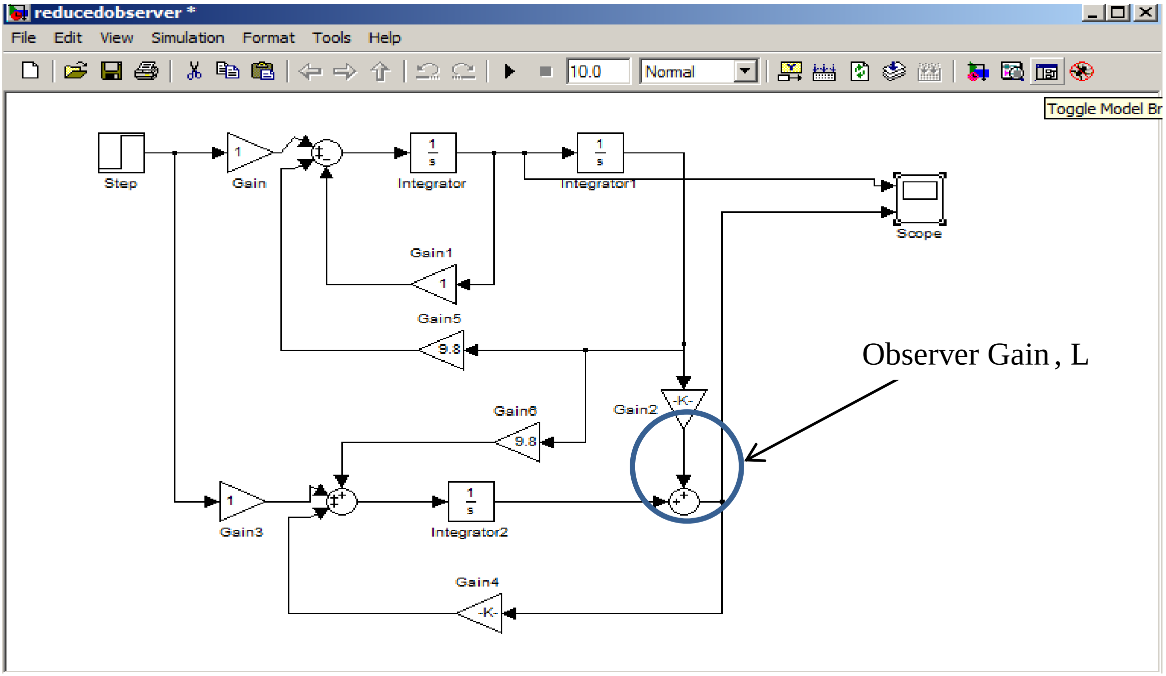
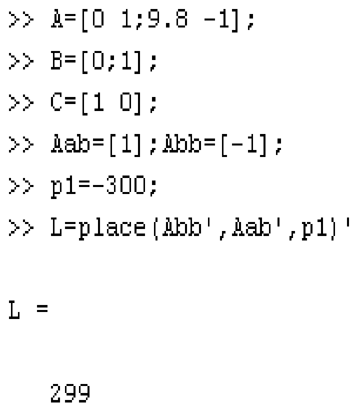


Figure 24: Block Diagram of robot arm manipulator system with a reduced order observer

i. Case 1: The first pole chosen for this current condition is -300 and pole placement method was used to find the reduced observer controller gain:



```
>> A=[0 1;9.8 -1];
>> B=[0;1];
>> C=[1 0];
>> Aab=[1];Abb=[-1];
>> p1=-300;
>> L=place(Abb',Aab',p1)'

L =

   299
```

The result of the simulated block diagram with step input is as shown in Figure 25.

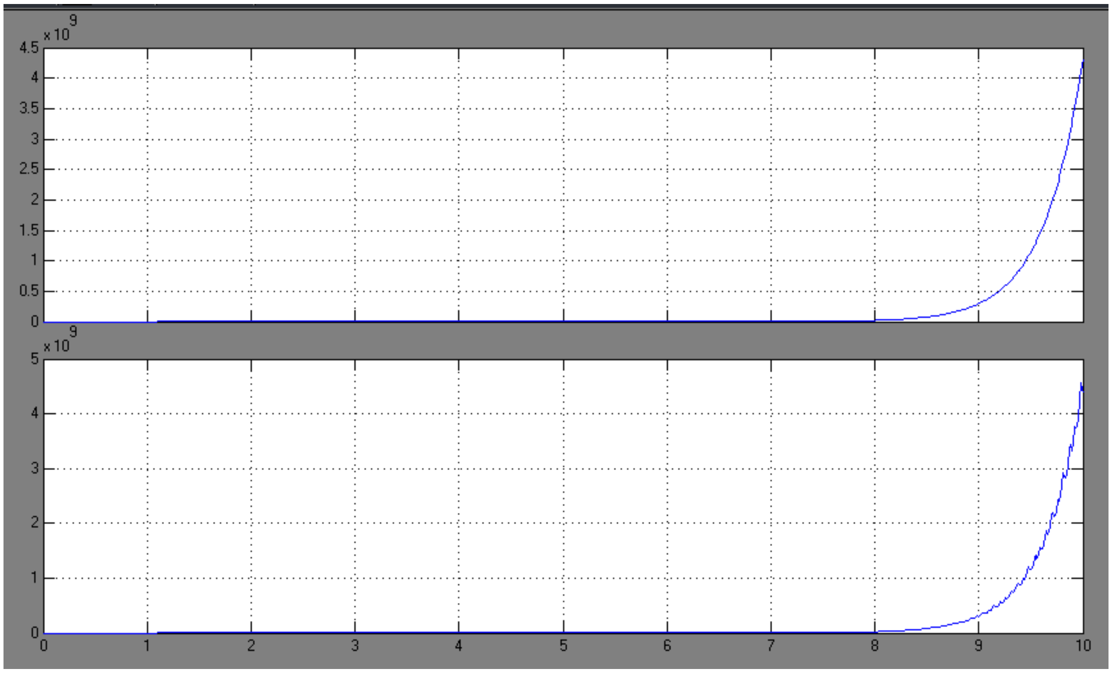
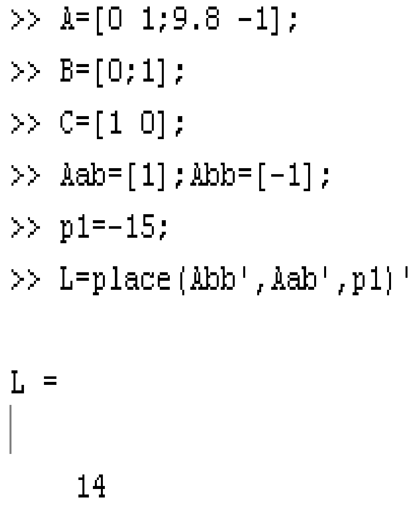


Figure 25: State of Original System,x_2 (top) and Observer Measured State, $\hat{x}_2$ (bottom)

Even though the measured state of reduced observer in Figure 25 seems to follow the state of original system, the fact that there is some noise occurred in the measured state could not be neglected. Hence, we can say that placing pole too further away shall introduce some noise to the reduced order observer system. Therefore, as the noise is considerable, we selected the observer poles to be slower in order to smooth the noise.

ii. Case 2: The second observer pole chosen is at -15 .We then solve for the observer gain as follows:



```
>> A=[0 1;9.8 -1];
>> B=[0;1];
>> C=[1 0];
>> Aab=[1];Abb=[-1];
>> p1=-15;
>> L=place(Abb',Aab',p1)'

L =

    14
```

The result of the simulated block diagram with step input is as shown in Figure 26.

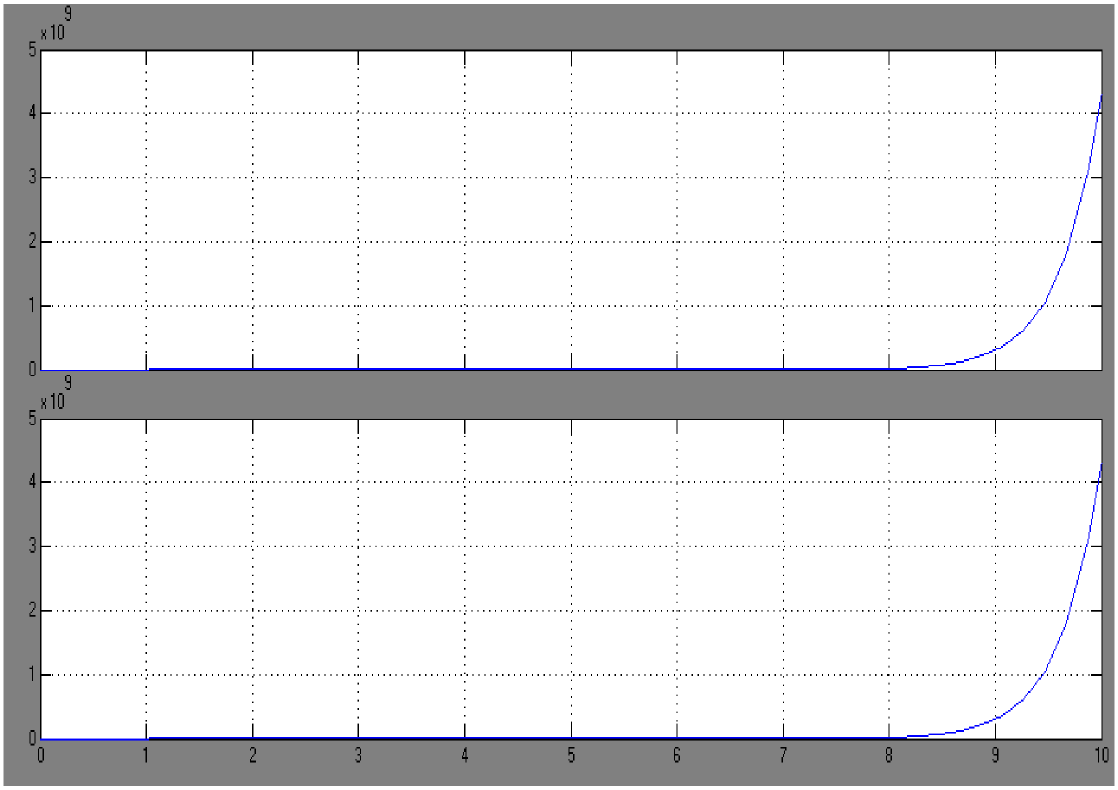


Figure 26: State of Original System,x_2 (top) and Observer Measured State, $\hat{x}_2$ (bottom)

Based on the result in Figure 26, it was confirmed that the pole located at –15 is quite desirable as the reduced observer showed behaviors similarities as the state of the system. However, to choose the most appropriate pole, we shall consider the error between the state of original system,x_2 and observer measured state, $\hat{x}_2$ before moving on to investigate other poles.The difference between the actual state and the observed state has been defined as :

$$e(t) = x(t) - \hat{x}(\mathrm{t}) \tag{21}$$

Therefore,the error is modelled in Simulink as in Figure 27.

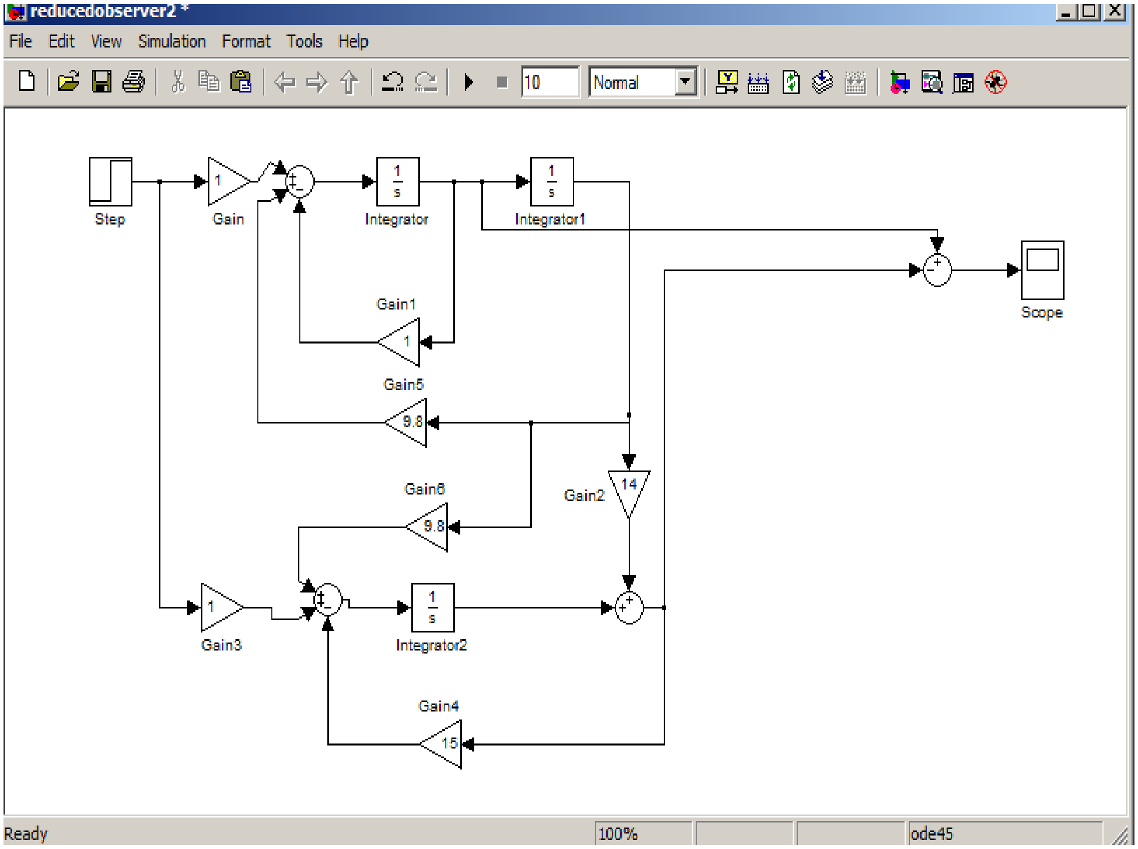


Figure 27: Block Diagram of difference between actual state and observed state

Hence,the difference between the actual state x_2 and $\hat{x}_2$is found as in Figure 28.

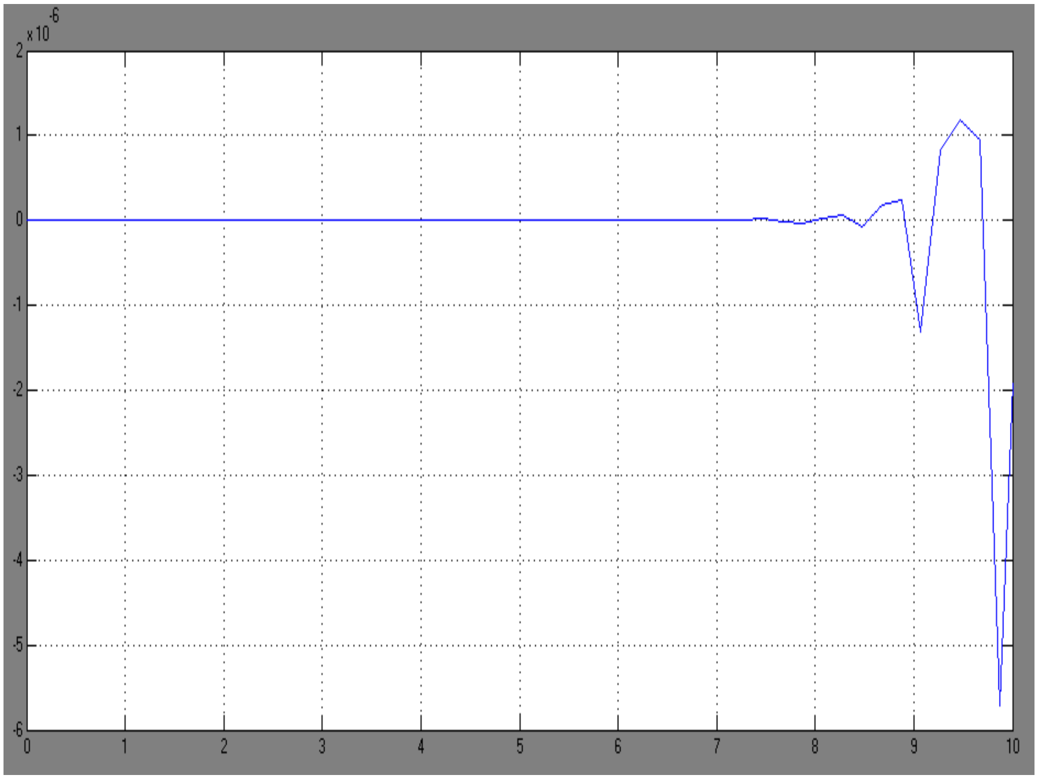
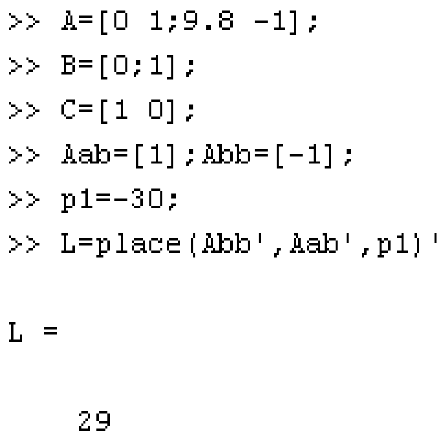


Figure 28: Difference Between Actual State and Observed State

There we can see that the error is very small which is in 10^{-6} .However,we still need to look at other pole as well.

iii. Case 3: The third poles is at -30 which is about four times further than the controller's pole .The observer's gain is found as:



```
>> A=[0 1;9.8 -1];
>> B=[0;1];
>> C=[1 0];
>> Aab=[1];Abb=[-1];
>> p1=-30;
>> L=place(Abb',Aab',p1)'

L =

    29
```

The result of the simulated block diagram with step input is as shown in Figure 29.

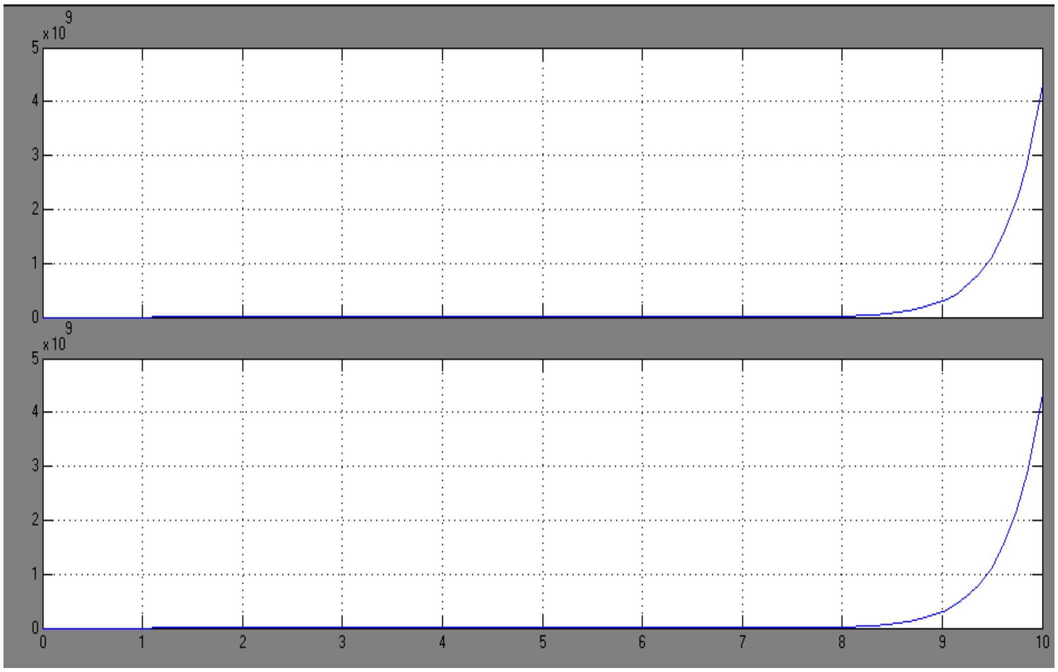


Figure 29: State of Original System,x_2 (top) and Observer Measured State, $\hat{x}_2$ (bottom)

As can be seen in Figure 29, the original system’s state exhibit the same behavior as observer measured state.However, we had to look into its error differences using equation (21).Thus, the error is found as in Figure 30.

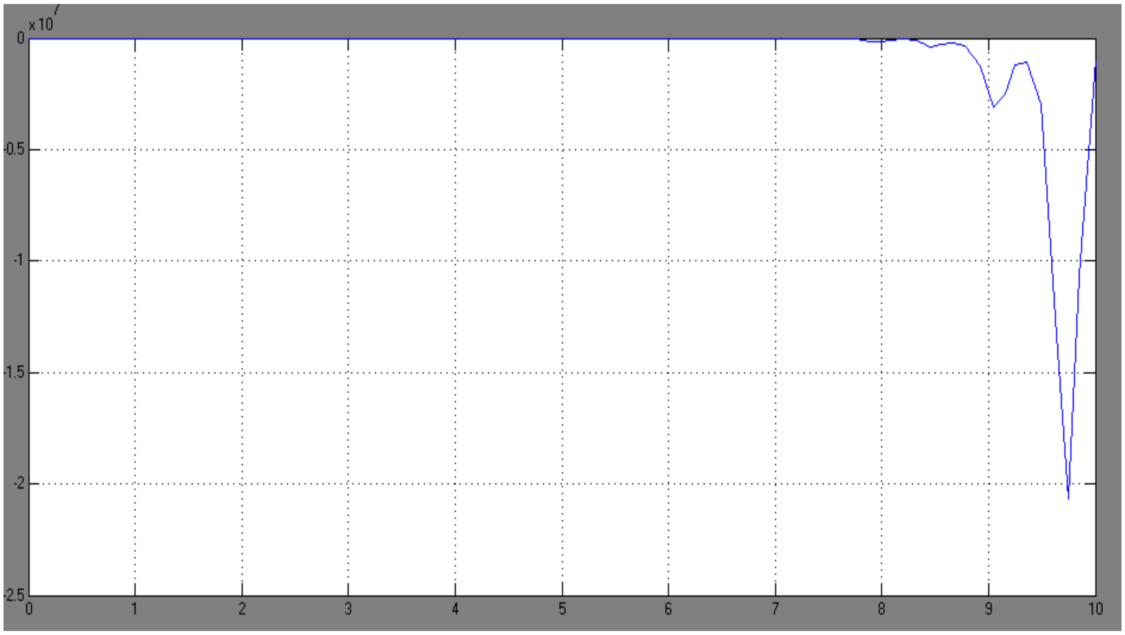


Figure 30: Difference Between Actual State and Observed State

There, we can see in Figure 30 that the error is quite large for Case 3 which is in 10^7.Hence, the pole for Case 3 is undesirable asnd we decided on Case 2 as our observer due to smaller error differences.

4.6 Quadratic Optimal Control

The basic goal is to find a control function, u that will minimize the cost function. Here, we assumed that x_2 which is the joint arm velocity is related to the applied signal (in voltage) and is given by the cost function:

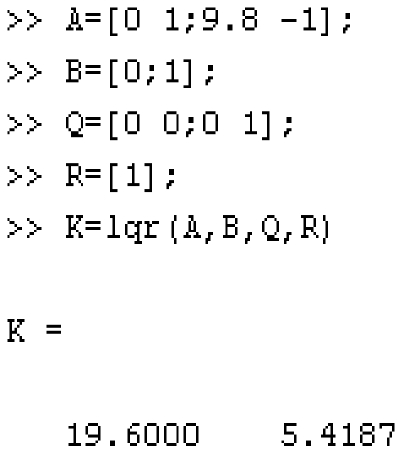
$$J = \int_0^\infty (x^2 + u^2)dt \tag{22}$$

Using the cost function, we found the value of Q (Please Refer to Appendix 3 for calculation) to be

$$Q = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{23}$$

In determining an optimal control law, we assume the value of R= [1].Q is a positive definite, or positive semi definite or real symmetric matrix and R is a positive-definite or a real symmetric matrix.

Based on the value of Q and R, we then found the optimal state feedback gain matrix, K such that the performance index or cost function is minimized using LQR method as below:



```
>> A=[0 1;9.8 -1];
>> B=[0;1];
>> Q=[0 0;0 1];
>> R=[1];
>> K=lqr(A,B,Q,R)

K =

   19.6000    5.4187
```

With the control signal u is given by

$$u = k_1(r - x_1) - (k_2 x_2) = k_1 r - (k_1 x_1 + k_2 x_2) \tag{24}$$

the optimal control of the plant can be constructed as shown in Figure 31.

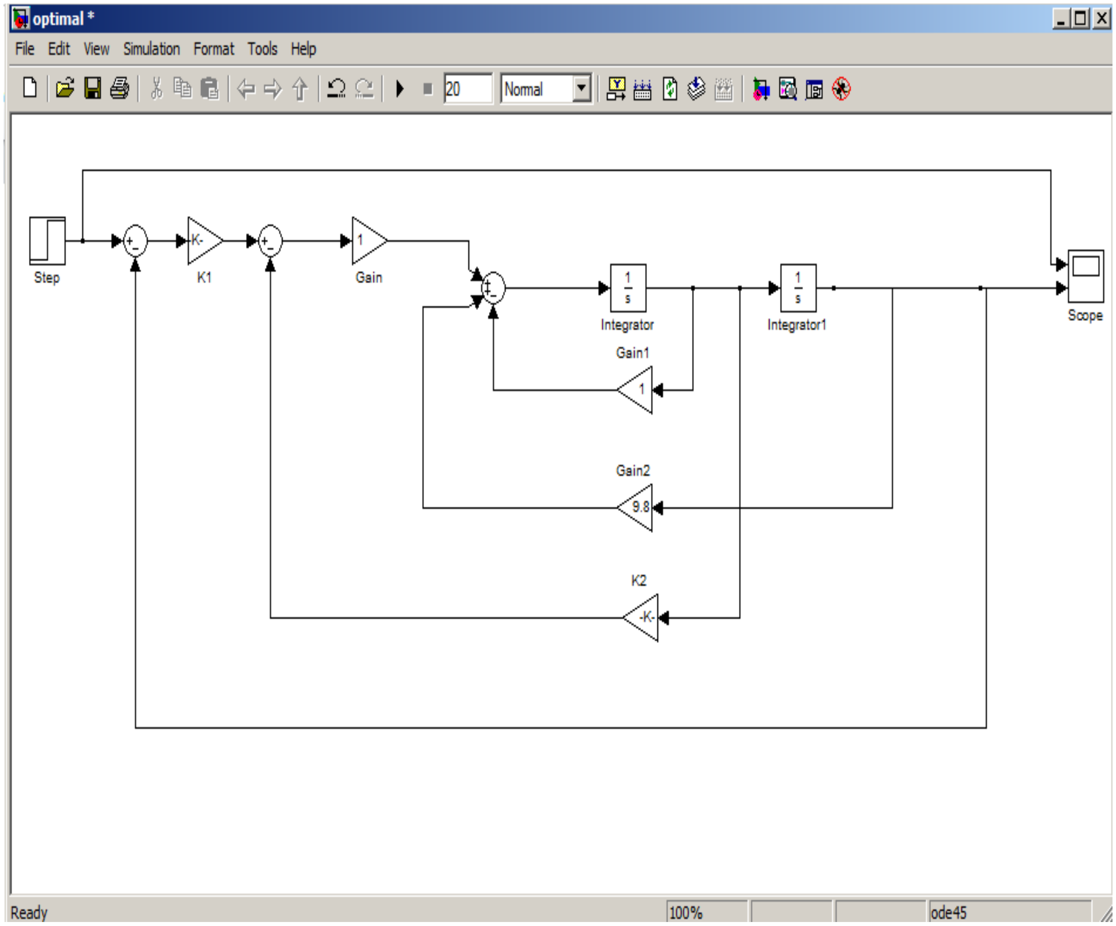


Figure 31: Block Diagram of optimal control of the plant

The output response of the optimal control system is as shown in Figure 32.

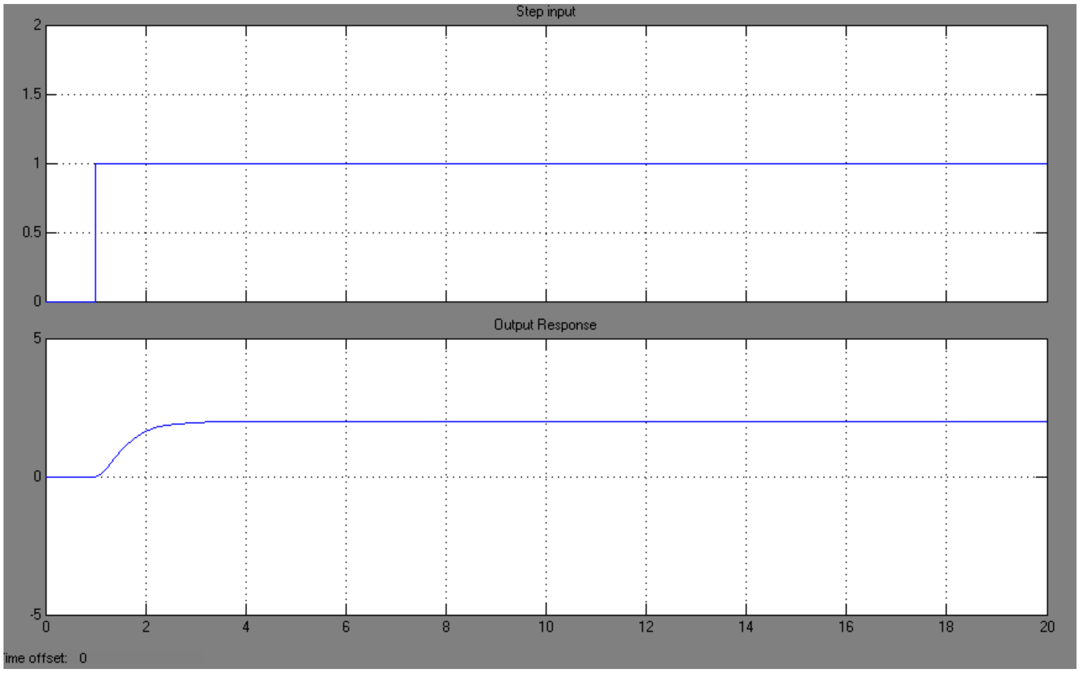


Figure 32: Unit Step input applied to the system (top) and its output response (bottom)

As shown in Figure 32, the design based on the quadratic performance yields a stable control system.

CHAPTER 5

CONCLUSIONAND RECOMMENDATIONS

5.1 Conclusion

In conclusion, from the knowledge of state space, design and analysis of a controller could be performed. Also, state feedback and feed forward controller for regulator is designed by using pole placement method while for the tracker problem, the design is based on Ackerman. Full state observer and reduced state observer were also designed to measure the internal state. For reduced order observer, we determine several observer gain matrices based on several different desired poles location by using pole placement method to have the best possible outcomes.Lastly, a quadratic optimal control system was designed using LQR method with minimization of performance index in mind.

5.2 Recommendations

There are several improvements that could be done on this study, for instance, more detailed analysis on the system's reduced order observer as it might be possible to have the observer's measured state to be exactly the same as observer's state by varying the poles location.. Also, a real implementation could be built based on the model.

REFERENCES

[1] M.Stig, O.Jonas, "*Robust Control of A Flexible Manipulator Arm: A Benchmark Problem*",Linkoping Technical Report,LiTH-ISY-R-2685,2006

[2] R.R. Szabo, P.Szabo, A.S.Pandya, "*Neural Network as Robot Arm Manipulator Controller*",IEEE Xplore,1994

[3] T.H. Lee,H.K. Lam,F.H.F. Leung,P.K.S. Tam, "*4 A practical fuzzy logic controller for the path tracking of wheeled mobile robots*", IEEE Control System Magazine,pp60-65,April 2003

[4] A.Izadbakhsh,M.M.Fateh, "*A Model Free Robust Control Approach for Robot Manipulator*", World Academy of Science, Engineering and Technology,2007

[5] Nise, Norman S., "*State Equation for Physical System*", Control Systems Engineering, 4th ed., John Wiley & Sons. Inc., pp112-142, 2004

[6] http://www.me.cmu.edu/ctm/ ,"*Modelling Tutorial for Matlab/Simulink*"

[7] Mark W. Spong Seth Hutchinson, M Vidyasagar, "S*ymbolic Representation of Robots*", Robot Modelling and Control, Wiley, 2006

[8] Frank L. Lewis,Darren M Dawson, Chaouki T. Abdullah, "*Common Kinematics Configuration*", Robot Manipulator Control , Marcel Dekker, 2004

[9] Katsuhiko Ogata, "Optimal Control", Modern Control Engineering, Prentice Hall,2002

[10] F.L.Lewis, "*State Observer and Regulator Design",2001*

[11] Wen-Hua Chen, D.J. Balance, Peter J. Gawthrop,J. O'Reilly*, "A Nonlinear Disturbance Observer for Robotic Manipulators*", IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 47, NO. 4, August 2000

[12] ETH Roberto Bucher, "*State Feedback Controller Design*", April 18, 2010

[13] Alessandro De Luca, DierkSchr¨oder, Michael Th¨ummel, "*An Acceleration-based State Observer for Robot Manipulators with Elastic Joints"* , 2007 IEEE International Conference on Robotics and Automation ,Roma, Italy, 10-14 April 2007

[14] I. López , J. Garay, R. Carreño, and Z. Varga, " *Observer Design for Ecological Monitoring*"

[15] Mohd Ashraf Ahmad, Raja Mohd Taufika Raja Ismail, Mohd Syakirin Ramli, "*Optimal Control with Input Shaping for Input Tracking and Vibration Suppression of a Flexible Joint Manipulator*", European Journal of Scientific Research, ISSN 1450-216X Vol.28 ,No.4 (2009), pp.583-599

APPENDICES

APPENDIX 1

FINAL YEAR PROJECT 1

Detail/ Week	1	2	3	4	5	6	7		8	9	10	11	12	13	14
Literature Review-Understand issue on Robotic	🟩	🟩	🟩												
Arm Manipukator			●												
Mathematical Model-Differential Equation				🟩	🟩	🟩	🟩								
							●								
Built State Space Model in Matlab/Simulink									🟩	🟩	🟩				
												●			
Analyze system performance												🟩	🟩	🟩	🟩
															●

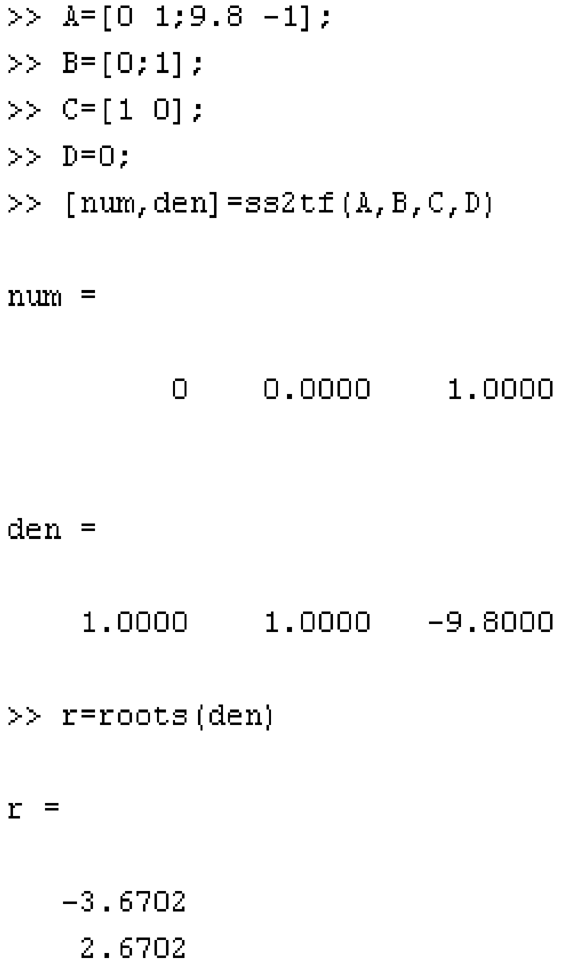
FINAL YEAR PROJECT 2

Detail/ Week	1	2	3	4	5	6	7		8	9	10	11	12	13	14
Designing Controller	🟩	🟩	🟩	🟩											
			●												
Designing Observer					🟩	🟩	🟩		🟩						
							●								
Evaluate System Perforamnace										🟩	🟩	🟩			
												●			
Documentation															
															●

🟩	Progress
●	Suggested Milestone

APPENDIX 2

The eigenvalues is found to be:



```
>> A=[0 1;9.8 -1];
>> B=[0;1];
>> C=[1 0];
>> D=0;
>> [num,den]=ss2tf(A,B,C,D)

num =

         0    0.0000    1.0000



den =

    1.0000    1.0000   -9.8000

>> r=roots(den)

r =

   -3.6702
    2.6702
```

APPENDIX 3

Given $\begin{bmatrix} \overset{\bullet}{x}_1 \\ \overset{\bullet}{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 9.8 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$ and $J = \int_0^\infty (x_2^2 + u^2) dt$.

Where P=solution of algebraic Riccati Equation.$A^T P + PA - PBB^T P + \in^T \in = 0$

To determine $\in$:

$J = \int_0^\infty x^T \in^T \in x + u^T u dt = \int_0^\infty (x^2 + u^2) dt$

Basically,$\in^T = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}$ and $\in = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}$

$\in^T \in = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} = \begin{bmatrix} a^2 & 0 \\ 0 & b^2 \end{bmatrix}$

But, $[x_1 \quad x_2] \begin{bmatrix} a^2 & 0 \\ 0 & b^2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = [x_1^2 a^2 \quad x_2^2 b^2]$

Or $x_1^2 a^2 + x_2^2 b^2 = x_2^2$

$\therefore b^2 = 1 \; or \; b = 1 \; and \; a = 0$

$\therefore \in = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \; and \; \in^T \in = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$

$\therefore Q = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$