Planar 2-Degrees Of Freedom Underactuated Manipulator: Design And Control

By

Ayman Karaki

Maher Alsarsour

Mohammad Dwaik

Supervisor

Dr. Iyad Hashlamon

Submitted to the College of Engineering

in partial fulfillment of the requirements for the

Bachelor degree in Mechatronics Engineering

Palestine Polytechnic University

July 2021

Planar 2-Degrees Of Freedom Underactuated Manipulator: Design And Control

By

Ayman Karaki

Maher Alsarsour

Mohammad Dwaik

Supervisor

Dr. Iyad Hashlamon

Submitted to the College of Engineering

in partial fulfillment of the requirements for the

Bachelor degree in Mechatronics Engineering

Palestine Polytechnic University

July 2021

Palestine Polytechnic University

College of Engineering

Department of Mechanical Engineering

Mechatronics Engineering

Planar 2-Degrees Of Freedom Underactuated Manipulator: Design And Control
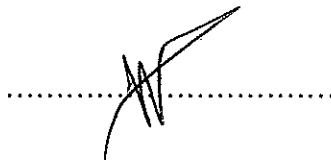
By

Ayman Karaki   Maher Alsarsour

Mohammad Dwaik

Submitted to the College of Engineering
in partial fulfillment of the requirements for the
Bachelor degree in Mechatronics Engineering

Supervisor Signature

....

Chair of the Department Signature

Abstract

This project aims to design, control and implement a planar two degrees of freedom (2DOF) underactuated manipulator. This robot is important and had been studied in the literature which reviewed and analyzed it based on the used control strategy. A full dynamic model of this type of robot is derived and analyzed, several control approaches based on this model have been implemented on an experimental setup that has been designed and implemented in this project.

The strategies of the control are divided into two parts, the first is by controlling the 2 DOF underactuated manipulator using the suggested control methods, and the other is finding a relation between the joint angles to convert the 2 DOF manipulator to 1 DOF manipulator then controlling it.

Keywords: underactuated, two link, planar, position control

الملخص

يهدف هذا المشروع الى تصميم وبناء و التحكم بذراع آلي افقي ثنائي درجات الحرية غير كامل الدفع(عدد المحركات اقل من عدد درجات الحرية) . هذا الذراع مهم و تمت دراسته و عرضه و تحليله حسب استراتيجيات التحكم المستخدمة في الدراسات السابقة .

تم اشتقاق و تحليل نموذج ديناميكي كامل للذراع ، و بنيت عدة تقنيات تحكم على اساس هذا النموذج على التطبيق العملي للذراع الذي أنشئ في هذا المشروع .

تقسم استراتيجية التحكم بالذراع الى قسمين ، الاول يكون بالتحكم بالذراع ثنائي درجات الحرية باستعمال التقنيات المقترحة ، والثاني يكون بإيجاد علاقة بين زوايا الذراع و بذلك يتحول الى احادي درجة الحرية ويتحكم به على هذا الاساس .

Dedication

We dedicate our project to our beloved parents for their continuous invaluable support and encouragement all through the years and to our dear siblings for providing us with a comfortable environment for study and research.

Acknowledgment

We would like to express our gratitude to our supervisor, Dr. Iyad Hashlamon, for his full support and guidance and remarkable suggestions. Also we would like to thank Dr.Khaled Tmaizi for his valuable help and advice. We would also like to thank our teachers for all the efforts they have exerted to make us qualified engineers who can assume, with confidence, our role in building our community. Thanks are also due to our classmates and friends for their cooperation and encouragement.

Table of Contents

List of Figures

List of Tables

Chapter 1 **Introduction**

1.1   Introduction

Mechanical systems can be classified into three major classes according to their degree of actuation. A mechanical system can be fully actuated when each degree of freedom can be individually controlled because the system has as many actuators as degrees of freedom. When the system has more actuators than degrees of freedom, the system is said to be overactuated.
Finally, underactuated mechanical systems are systems where the numbers of the actuators are less than system's degrees of freedom (DOF) to be controlled, in other words, they are robots with both passive and active joints. For example, when a joint motor in a fully actuated robot fails, that joint becomes passive. So it is important in such cases to design control techniques to still control the robot, if possible.

| Space robot used in NASA | Pick and Place Robotic Arm | Underwater Robots |

| heavy industrial manipulator | AutoNaut Surface vessel |
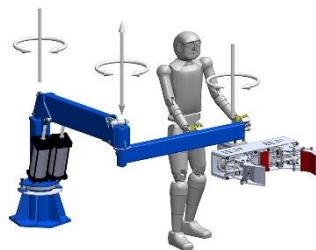
Figure 1.1: Some applications of underactuated systems

Underactuated systems have very important applications such as space robotics, underwater robots, surface vessels, pick-and-place mechanisms, industrial manipulators, and flexible and mobile robots.

Important examples of underactuated systems are when the goal of the designers is to design a robot with more natural and flexible movement such as Boston Dynamics Robot celebrating the new year 2021 [1] ,or in bio inspired robots such as worm inspired robot [2], they are also used in medical field [3]. Figure 1.1 shows some applications of Underactuated manipulators.

The control task of such systems is complicated by the characteristics such as complex nonlinear dynamics, nonholonomic behavior, and lack of linearity that are often exhibited by this class of nonlinear systems, however, underactuated robots are theoretically interesting and practically important class of nonlinear systems among the research community. Thus, controlling this class of robots is a challenging task and still remains as an open problem, and it doesn't have a general method to solve it.

The challenge of controlling these systems is of great meaning because it allows us to develop simpler systems with reduced weight, cost and energy consumption, and increased flexibility and less damage while hitting an object, which yields to a more efficient robots than fully actuated robots for specific purposes.

1.2 Recognition of the need

In the last few decades, robots have been used increasingly in different fields, such as industrial and medical fields to reduce time and efforts, they can be used also to do tasks which are difficult or dangerous instead of humans.

When a robot is designed, it should match the costumer's requirements, some of these requirements are the energy consumption, overall cost and weight.

In design phase, if these robots can be replaced by underactuated robots, their overall cost will be less since less number of motors are used, thus, energy consumption and overall weight will be less than fully actuated one.

This project aims to find a proper control method to control an underactuated robot prototype, this method can be used later to control the robots in real life.

1.3 Literature Review

This section provides an overview of some of previous efforts of planar 2-DOF underactuated manipulator shown in Figure 1.2. This robot is a two link serial planar robot, joint 1 is an active joint while joint 2 is passive, link 1 has length l_1 with center of gravity CG_1 at position r_1 and rotates at an angle θ_1. Link 2 has length l_2 with center of gravity CG_2 at position r_2 and rotates at an angle θ_2. The point E is the end effector.
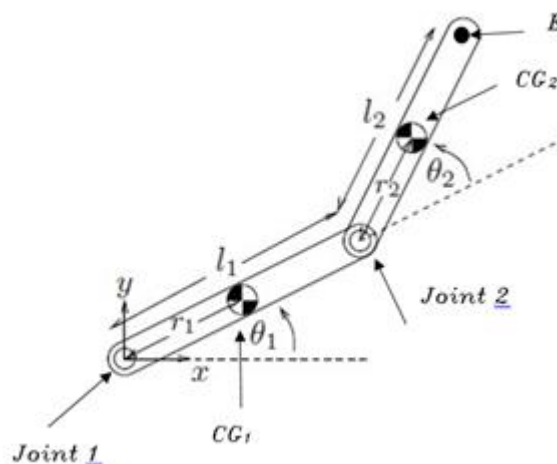
Figure 1.2: 2 DOF underactuated manipulator

First of all, controllability of the system should be checked to decide if the project is applicable or not.

In [4] ,the authors discussed the controllability of underactuated manipulators with one unactuated joint, and proved that if the first joint (the basic one joint 1) is actuated, then the system is

3

controllable. In other words, the system can move from a given initial point to a desired point through a trajectory.

To prove this, the authors used global stabilizing feedback controller which provides two trajectories, one starts from the initial point and the other starts from the final point, after having this information, the authors used a bidirectional approach to prove that the trajectory exists.

Finally, the authors used a numerical example to discuss the controllability of the general n-links underactuated manipulators when only the first joint is actuated.

The authors concluded that if the system has the first joint actuated, and only one of the other joints is unactuated, then the system is controllable, otherwise, the system is partially controllable.

Although our system is controllable, the control of it is a challenge since the system doesn't satisfy Brockett's theorem[1], which means that the system cannot be controlled using continuous control laws, thus the system can be controlled by time varying control or discontinuous control laws.

In [5] ,the authors discussed a strategy based on time varying control laws to control planar 2R underactuated manipulator by making the system tracking appropriate trajectories to reach the desired points. The trajectories depend on parameters which depend on the initial and final desired points, and these parameters are selected by a trajectory optimization algorithm.

To ensure that the system reaches the desired points, a feedback control is used.

Using numerical example, the authors concluded that if the initial values are selected appropriately and the system tracked the planned trajectories, it will reach the desired final point without error.

In [6] , the authors discussed another strategy to control the system by taking friction forces into consideration, the control consist of two stages

- The first is to use a controller to move the passive link (the second link) to its desired point.
- The second controller aim is to move the active link (the first link) to its desired point without disturbing the second link, this can be applied by ensuring that actuating torque is less than the friction force.

Using simulation, the authors concluded that this strategy is effective.

In [7] , the authors reviewed the underactuated systems shown in Figure 1.3 and their applications, such as inverted wheel pendulum and Acrubot, classified them, and reviewed their control methods and strategies and their dynamic models and the advantages and disadvantages of each control strategy and the challenges of controlling them.

In [3] the authors introduced an underactuated gripper of human limbs used in rescue operations, this gripper aims to assists laying patients and rehabilitation exoskeleton or prosthesis patients.

The interaction between robots and human must be stable and safe especially when physical contact occurs.

The researchers controlled the system by an impedance Model Predictive Control (MPC) to stabilize the system and done experiments with volunteers people to obtained the results and used the machine learning to get the estimation model by three methods: Gaussian Process (GPR),

---

[1] Brockett's theorem states that if f is continuously differentiable a necessary condition for the existence of a continuously differentiable feedback control u that renders 0 locally asymptotically stable is that the image of f contain an open neighborhood of 0.

Regression Tree (RT) and Bagging Regression Tree (BRT), and very precise sensors to locate the gripper to the desired location to perform safe and autonomous grasps with enough robustness and reliability to manipulate human limps.

(a) The Acrobot

(b) The Pendubot

(a) The cart-pole system

(b) The crane system

The rotational pendulum.

The Inverted wheel pendulum.

The beam and ball.
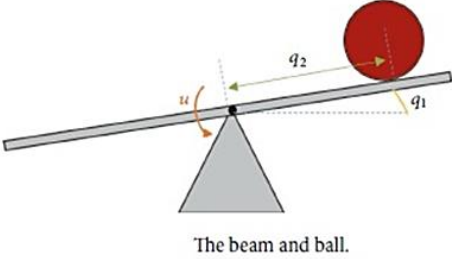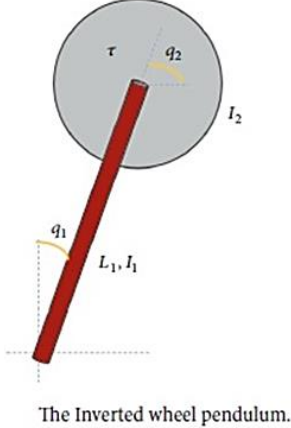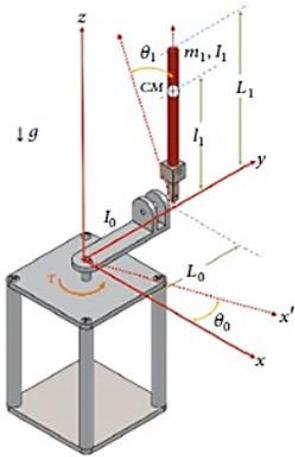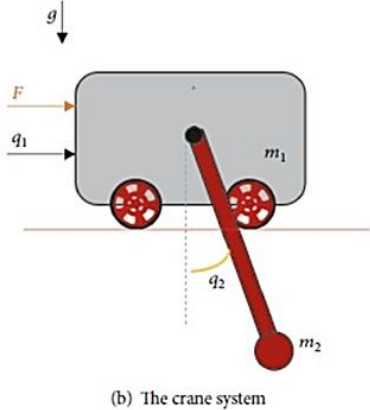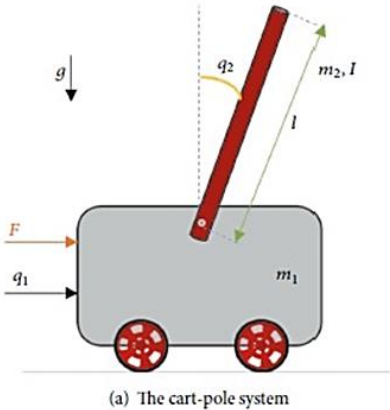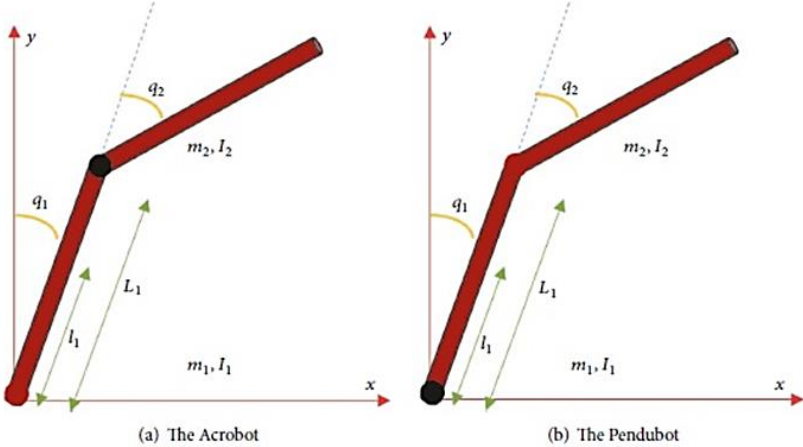
Figure 1.3: Some types of underactuated systems

1.4 Methodology

This project contains several stages

- **Theoretical calculations:** which contain the dynamic model and the design of the mechanical structure for the manipulator.
- **Software modeling:** the manipulator will be modeled in SOLIDWORKS software to check some values such as maximum deflection and safety factor.
- **Simulation of fully actuated robot:** the manipulator will be simulated using MATLAB and SIMULINK software then the response of the fully-actuated model will be studied.
- **Control simulation of underactuated robot:** the suitable controller will be designed to control the underactuated robot.

1.5 Time table

Table 1.1 shows the time table for the first semester distributed on 16 weeks.

Table 1.1: Time table of the first semester

Tasks \ Weeks	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Identification of Project Idea	▓	▓	▓	▓												
Literature review				▓	▓	▓	▓									
Introduction Chapter 1								▓	▓							
Mathematical model Chapter 2					▓	▓	▓	▓	▓							
Mechanical design Chapter 3										▓	▓	▓	▓	▓		
Begin controller study Chapter 5															▓	▓

Table 1.2 shows the time table for the second semester distributed on 16 weeks.

Table 1.2: Time table of the second semester

Weeks / Tasks	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Electrical design	■	■	■													
Project requirements and collecting data	■	■	■													
Controller study Chapter 5 And disturbance estimation Chapter 4		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Model Simulation						■	■									
Parts manufacturing	■	■														
Manipulator assembly	■	■														
Testing and experimental results								■	■	■	■	■	■	■	■	■
Results conclusion and discussion														■	■	■

1.6   Cost table

Table 1.3 shows the estimated cost of the project, where some devices such as NI DAQ card and DC motor are given by the university laboratories.

Table 1.3: Estimated cost table

Tools and device	Number	Piece price (NIS)	Price (NIS)
NI Daq card	1	4000	4000
Arduino microcontroller	1	100	100
DC motor	1	300	300
Motor driver	1	30	30
Encoder	2	250	500
Cables, Wires	2	50	100
Bearing	1	25	25
Bolt,nuts	4	5	20
Parts manufacturing			200
Raw materials	1	40	40
	Total (NIS)		5275

Chapter 2   Problem definition and mathematical model

2.1 Problem definition

Although the underactuated systems have difficulties of control, it have many advantages such as reduction of weight, reduction of tendency to breakdown, and reduction of energy cost. [7]

Reference [7] has reviewed the control strategies used to control the systems in Figure 1.3 from 2000 till now, the strategies are divided to linear, nonlinear, and intelligent .

The linear strategies consist of PID, Linear Quadratic Regulation (LQR), and pole placement, while the nonlinear strategies consist of feedback linearization, sliding mode, and backstepping, the intelligent strategies consist of fuzzy controllers and neural networks.

All the previous systems in Figure 1.3 are controlled using various methods mentioned in previous paragraph except PID control, which is not used at all or used in very narrow range. The objective of this project is to attempt to control the 2 link planar underactuated system using PID control with some modifications because of its ease of implementation and its good performance.

The objective of the project is to make the system to achieve the following goals

- Stop at desired joint angles.
- Draw a circle at any center, an example in Figure 2.1.
- Draw a circle its center is the origin (actuator), an example in Figure 2.2.

Where the first and the third targets are achieved by experimental approach, where the position of the end effector is known and a relationship between the joint angles is established, which converts the 2 DOF manipulator to 1 DOF manipulator , the second target is achieved by simulation based on the model of 2 DOF manipulator.
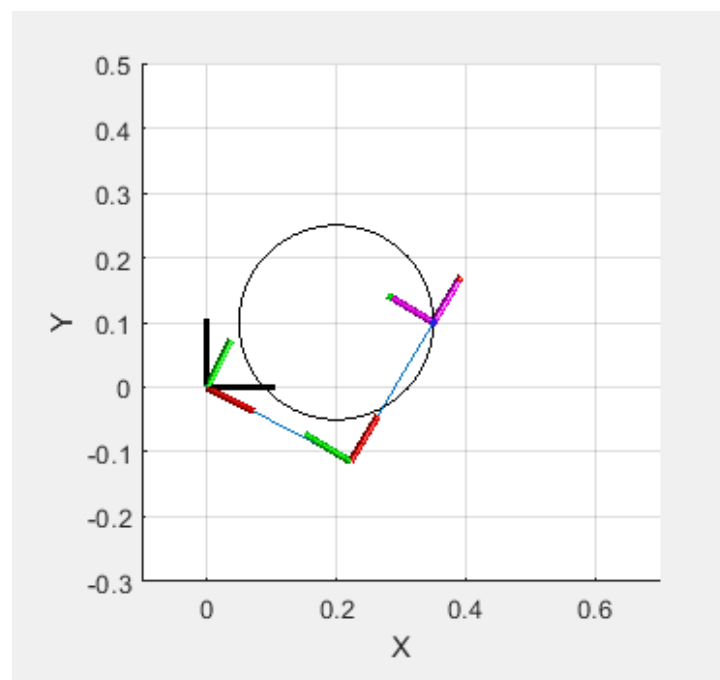
Figure 2.1: Circle at random center goal
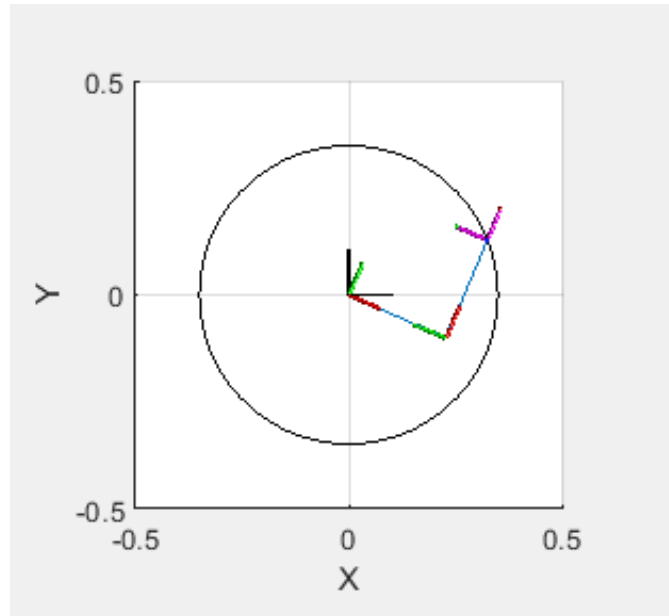
Figure 2.2: Circle with center at origin goal

2.2   Mathematical model

2.2.1   Direct (forward) kinematics

The mechanical structure of a manipulator is characterized by a number of degrees of freedom (DOFs) which uniquely determine its posture. Each DOF is typically associated with a joint articulation and constitutes a joint variable. [8]

The direct kinematics aim to compute the pose of the end effector from values of joint variables with respect to the base frame. [8]

The computation of the direct kinematics is offered by a geometric analysis of the manipulator.
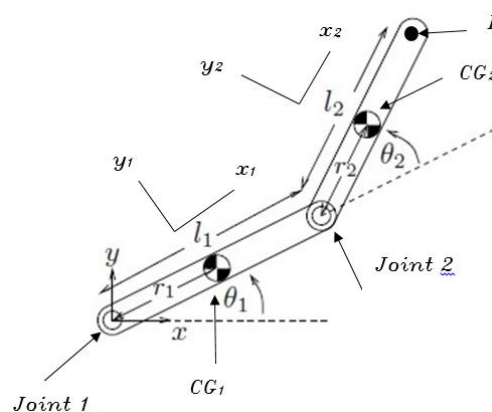
Figure 2.3: Two link manipulator with joints frames

As shown in Figure 2.3, the base frame at joint 1 with axes x, y and z which follows the right hand rule. An auxiliary frame will be attached to each link to describe its pose with respect to the base

11

frame.as follows: frame 1 for link 1 with axes x_1 and y_1, and frame 2 for link 2 with axes x_2 and y_2 [8].

The rotation matrices of the auxiliary frames with respect to the base frame shown in Figure 2.3 are

$$R_2^1 = \begin{pmatrix} c_1 & -s_1 & 0 \\ s_1 & c_1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$R_E^2 = \begin{pmatrix} c_2 & -s_2 & 0 \\ s_2 & c_2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{1}$$

$$R_E^1 = R_2^1 R_E^2 = \begin{pmatrix} c_{12} & -s_{12} & 0 \\ s_{12} & c_{12} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Where $c_1 = \cos(\theta_1), s_1 = \sin(\theta_1), c_2 = \cos(\theta_2), s_2 = \sin(\theta_2), c_{12} = \cos(\theta_1 + \theta_2), s_{12} = \sin(\theta_1 + \theta_2)$.

Using geometrical approach we find the position of the end effector with respect to the base, as

$$P_E^1 = \begin{pmatrix} P_x \\ P_y \\ P_z \end{pmatrix} = \begin{pmatrix} L_1 c_1 + L_2 c_{12} \\ L_1 s_1 + L_2 s_{12} \\ 0 \end{pmatrix} \tag{2}$$

Then the homogenous transformation matrix is [8]

$$T_E^1 = \begin{pmatrix} R_E^1 & P_E^1 \\ 0^T & 1 \end{pmatrix}$$

$$T_E^1 = \begin{pmatrix} c_{12} & -s_{12} & 0 & L_1 c_1 + L_2 c_{12} \\ s_{12} & c_{12} & 0 & L_1 s_1 + L_2 s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{3}$$

2.2.2   Inverse kinematics

The inverse kinematics goal is to determine of the joint variables corresponding to a given end effector position and orientation[8]. The importance of solving such a problem is to determine the joint variables to achieve desired missions based on specified end-effector poses. [9]

A geometrical approach can solve the inverse kinematics problem , this approach is suitable for simple structured robots, for complex structured robot, an analytical approach is used to solve the problem [8],since the manipulator shown in Figure 2.4 is simple, geometrical analysis is used.
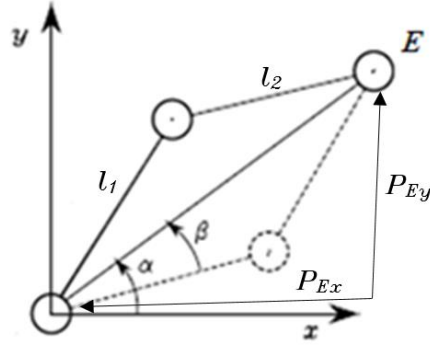
Figure 2.4: Analysis of the manipulator

Using geometrical techniques, and given the end effector position, the joint variables are found as follows

We apply cosine theorem for the triangle formed by L_1, L_2 and the segment between points 1 and E as shown in Figure 2.5 .
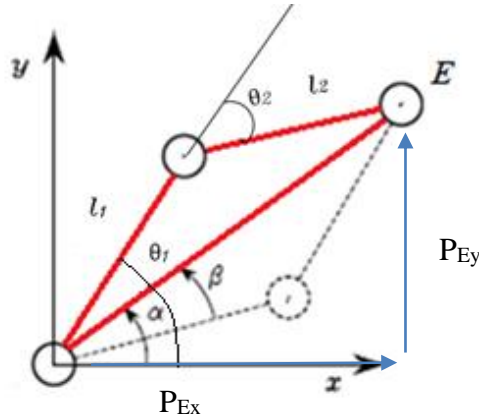
Figure 2.5: Triangular analysis of manipulator

$$P_{Ex}^2 + P_{Ey}^2 = L_1^2 + L_2^2 - 2L_1L_2\cos(180 - \theta_2) \tag{4}$$

Where P_{Ex} and P_{Ey} are the position of end-effector E.

It's known that

$$\cos(180 - \theta_2) = -\cos\theta_2 \tag{5}$$

Which leads to

$$\cos\theta_2 = \frac{P_{Ex}^2 + P_{Ey}^2 - L_1^2 - L_2^2}{2L_1L_2} \tag{6}$$

$$\theta_2 = \pm\cos^{-1}(\cos\theta_2)$$

The elbow-up posture is obtained for $\theta_2 \in (-\pi, 0)$ while the elbow-down posture is obtained for $\theta_2 \in (0, \pi)$.

θ_1 is found from computing the angles α and β by geometry in Figure 2.4 .

13

$$\alpha = \tan^{-1} \frac{P_{Ey}}{P_{Ex}} \qquad (7)$$

β is computed by using cosine theorem, which leads to

$$\sqrt{P_{Ex}^2 + P_{Ey}^2} \cos\beta = L_1 + L_2 \cos\theta_2 \qquad (8)$$

Substitution of $\cos\theta_2$ leads to

$$\beta = \cos^{-1}\left(\frac{P_{Ex}^2 + P_{Ey}^2 + L_1^2 - L_2^2}{2L_1\sqrt{P_{Ex}^2 + P_{Ey}^2}}\right) \qquad (9)$$

With $\beta \in (0, \pi)$ so as to preserve the existence of triangles. Then,

$$\theta_1 = \alpha \pm \beta \qquad (10)$$

Where the positive sign holds for $\theta_2 < 0$ and the negative sign for $\theta_2 > 0$.

2.2.3 Differential kinematics

The goal of the differential kinematics is to find the relationship between the joint velocities and the end-effector linear and angular velocities [9]. This mapping is described by a matrix, termed geometric Jacobian.

The Jacobian constitutes one of the most important tools for manipulator characterization, it is useful for finding singularities, analyzing redundancy and determining inverse kinematics algorithms [8].

The general equation for both linear and angular velocities is

$$\begin{pmatrix} \dot{P} \\ \omega \end{pmatrix} = J\dot{\theta}$$

$$\dot{\theta} = \begin{pmatrix} \dot{\theta}_1 & \dot{\theta}_2 \end{pmatrix}^T \qquad (11)$$

Where $\dot{P}$ and ω are the end-effector linear and angular velocities respectively, J is the Jacobian matrix and $\dot{\theta}$ is the joint velocities vector.

Since the orientation is of no concern, the equation reduces to

$$\dot{P} = J\dot{\theta} \qquad (12)$$

Where J here is the positional Jacobian matrix which is found from

$$P = P_E^1 = \begin{pmatrix} P_x \\ P_y \\ P_z \end{pmatrix} = \begin{pmatrix} L_1\, c_1 + L_2\, c_{12} \\ L_1\, s_1 + L_2\, s_{12} \\ 0 \end{pmatrix}$$

$$\dot{P} = \begin{pmatrix} \dot{P}_x \\ \dot{P}_y \\ \dot{P}_z \end{pmatrix} = \frac{\partial P}{\partial \theta}\frac{d\theta}{dt} \tag{13}$$

$$J = \frac{\partial P}{\partial \theta}$$

$$J = \begin{pmatrix} \dfrac{\partial P_x}{\partial \theta_1} & \dfrac{\partial P_x}{\partial \theta_2} \\[2mm] \dfrac{\partial P_y}{\partial \theta_1} & \dfrac{\partial P_y}{\partial \theta_2} \\[2mm] \dfrac{\partial P_z}{\partial \theta_1} & \dfrac{\partial P_z}{\partial \theta_2} \end{pmatrix} \tag{14}$$

After differentiation, Jacobian equals

$$J = \begin{pmatrix} -L_1 s_1 - L_2 s_{12} & -L_2\, s_{12} \\ L_1\, c_1 + L_2\, c_{12} & L_2\, c_{12} \\ 0 & 0 \end{pmatrix} \tag{15}$$

2.2.4 Dynamic model

Derivation of the dynamic model of a manipulator has an important role for motion simulation, analysis of manipulator structures, and design of controllers. Simulating manipulator motion allows testing controllers and trajectory planners without having a physical system. The analysis of the dynamic model can be helpful for mechanical design of prototype arms. Computation of the forces and torques required for the execution of desired motions is required for selecting joints, transmissions and actuators. [8]

The dynamic model of the manipulator provides a description of the relationship between the joint actuator torques and the motion of the structure. There are some methods to derive the equation of motion for such a robot, like Newton Euler's formulation and Lagrange formulation .[9] .The general equation of motion is

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + g(\theta) = \tau(\theta) \tag{16}$$

Where M is the mass matrix, C is the Coriolos and centrifugal matrix, g is the gravity vector and τ torque vector.

Noting that the mass matrix is configuration dependent (function of joints position θ). This mathematical dependency emphasizes the physical fact that the robot arm has different inertia based on its configuration [9].

Since the manipulator acts in planar plane, the gravitational contribution is absent, thus g(θ)=0.

The method used to derive the dynamic model of the two link planar manipulator here is Lagrange formulation.

Lagrange equation of motion is given by

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_i}\right) - \frac{\partial L}{\partial \theta_i} = \tau_i \tag{17}$$

$$i = 1, 2$$

Where L is the Lagrangian, which is given by [9]

$$L = K - P \tag{18}$$

Where K and P are the total kinetic and potential energy of the manipulator, respectively.

The calculation of the kinetic and potential energy of the links is a function of the joint angles and velocities. This requires having a model for the mass distribution of the links. Since each link is a rigid body, its kinetic and potential energy can be defined in terms of its total mass and its moments of inertia about the center of mass.
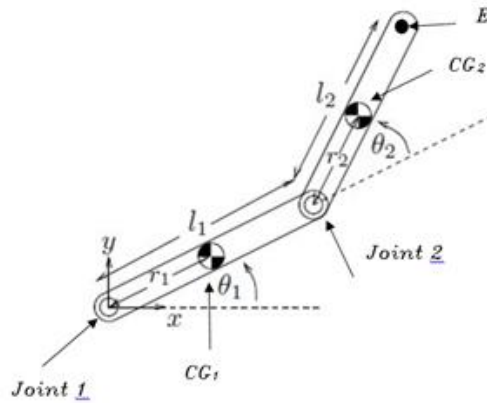
Figure 2.6: 2 link manipulator for dynamic model

For the system shown in Figure 2.6, where

- r_1 is the distance between first center of gravity and first joint.
- r_2 is the distance between second center of gravity and second joint.
- L_1 is the length of first link.
- L_2 is the length of second link.

The inertia tensor of the object expressed in the body frame is [10]

$$I = \begin{pmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{pmatrix} \tag{19}$$

Where I_{xx}, I_{yy}, I_{zz}, are called the Principal Moments of Inertia about the x,y,z axis, respectively, and I_{xy}, I_{xz}, etc., are called the Cross Products of Inertia.

Since the mass distribution of the body is symmetric with respect to the body attached frame then the cross products of inertia are identically zero, which leads to

$$I = \begin{pmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{pmatrix} \qquad (20)$$

The general kinetic energy equation of the rigid body is [10]

$$K = \frac{1}{2}mv^2 + \frac{1}{2}I\omega^2 \qquad (21)$$

Where the first term is the translational kinetic energy and the second term is the rotational kinetic energy.

Letting $v_i \in R^3$ be the translational velocity of the center of mass for the ith link and $\omega_i \in R^3$ be the angular velocity, the kinetic energy of the manipulator is [11]

$$K(\theta,\dot{\theta}) = \frac{1}{2}m_1 v_1^2 + \frac{1}{2}I_1\dot{\theta}_1^2 + \frac{1}{2}m_2 v_2^2 + \frac{1}{2}I_2\dot{\theta}_2^2 \qquad (22)$$

Since the links rotate about z-axis ,the inertia about z-axis is only required and Since the motion of the manipulator is restricted to the xy plane, v_i is the magnitude of the xy velocity of the center of mass and ω_i is a vector in the direction of the z-axis, with $\omega_1 = \dot{\theta}_1$ and $\omega_2 = \dot{\theta}_1 + \dot{\theta}_2$.

Kinetic energy becomes [11]

$$K(\theta,\dot{\theta}) = \frac{1}{2}m_1(\dot{x}_1^2 + \dot{y}_1^2) + \frac{1}{2}I_{1zz}\dot{\theta}_1^2 + \frac{1}{2}m_2(\dot{x}_2^2 + \dot{y}_2^2) + \frac{1}{2}I_{2zz}(\dot{\theta}_1 + \dot{\theta}_2)^2 \qquad (23)$$

Where I_{1zz} is the moment of inertia of the first link around the z axis and I_{2zz} is the moment of inertia of the second link around the z axis.

Let $p_i = (x_i, y_i, 0)$ denote the position of the ith center of mass. Letting r_1 and r_2 be the distance from the joints to the center of mass for each link, as shown in Figure 2.6, we have [11]

$$
\begin{aligned}
x_1 &= r_1 c_1 \\
\dot{x}_1 &= -r_1 s_1 \dot{\theta}_1 \\
y_1 &= r_1 s_1 \\
\dot{y}_1 &= r_1 c_1 \dot{\theta}_1 \\
x_2 &= l_1 c_1 + r_2 c_{12} \\
\dot{x}_2 &= -(l_1 s_1 + r_2 s_{12})\dot{\theta}_1 - r_2 s_{12}\dot{\theta}_2 \\
y_2 &= l_1 s_1 + r_2 s_{12} \\
\dot{y}_2 &= (l_1 c_1 + r_2 c_{12})\dot{\theta}_1 + r_2 c_{12}\dot{\theta}_2
\end{aligned}
\qquad (24)
$$

The kinetic energy becomes [11]

$$K(\theta,\dot{\theta}) = \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix}^T \begin{pmatrix} \alpha + 2\beta c_2 & \delta + \beta c_2 \\ \delta + \beta c_2 & \delta \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \qquad (25)$$

17

Where

$$\alpha = I_{1zz} + I_{2zz} + m_1 r_1^2 + m_2(l_1^2 + r_2^2)$$
$$\beta = m_2 l_1 r_2 \tag{26}$$
$$\delta = I_{2zz} + m_2 r_2^2$$

Kinetic and potential energy are substituted in Lagrangian [11]

$$L = K - P \tag{27}$$

Since the manipulator acts in planar plane, the potential energy equals zero, which leads to

$$L = K \tag{28}$$

To find torques, Lagrange equation of motion is applied

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_i}\right) - \frac{\partial L}{\partial \theta_i} = \tau_i \tag{29}$$
$$i = 1, 2$$

After calculations of the previous equation, the equation of motion is [11]

$$\begin{pmatrix} \alpha + 2\beta c_2 & \delta + \beta c_2 \\ \delta + \beta c_2 & \delta \end{pmatrix} \begin{pmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{pmatrix} + \begin{pmatrix} -\beta s_2 \dot{\theta}_2 & -\beta s_2 (\dot{\theta}_1 + \dot{\theta}_2) \\ \beta s_2 \dot{\theta}_1 & 0 \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} = \begin{pmatrix} \tau_1 \\ \tau_2 \end{pmatrix} \tag{30}$$

Since the second joint is passive, $\tau_2 = 0$

$$\begin{pmatrix} \alpha + 2\beta c_2 & \delta + \beta c_2 \\ \delta + \beta c_2 & \delta \end{pmatrix} \begin{pmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{pmatrix} + \begin{pmatrix} -\beta s_2 \dot{\theta}_2 & -\beta s_2 (\dot{\theta}_1 + \dot{\theta}_2) \\ \beta s_2 \dot{\theta}_1 & 0 \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} = \begin{pmatrix} \tau_1 \\ 0 \end{pmatrix} \tag{31}$$

The state space representation of the system is required for later work. Assume the states of the system are

$$x = \begin{pmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \tag{32}$$

Using feedback linearization assuming a well known mode, every nonlinear term in (30) is eliminated, therefore, the dynamic model becomes

$$\begin{pmatrix} I_{1zz} + I_{2zz} & I_{2zz} \\ I_{2zz} & I_{2zz} \end{pmatrix} \begin{pmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{pmatrix} = \begin{pmatrix} \tau_1 \\ 0 \end{pmatrix}$$

$$I \begin{pmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{pmatrix} = \begin{pmatrix} \tau_1 \\ 0 \end{pmatrix} \tag{33}$$

The mathematical model in (32) is considered to be fully linearized in state space representation form, which is

18

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \qquad (34)$$

$$B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ I^{-1} \end{pmatrix} \qquad (35)$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \qquad (36)$$

$$D = 0 \qquad (37)$$

Where $A \in \mathbb{R}^{4x4}$ and $B \in \mathbb{R}^{4x2}$ and $C \in \mathbb{R}^{2x4}$.

2.2.5 Autolev software code

Autolev software is used to get some dynamic information of some mechanical systems.

In this project, Autolev software is used to generate

- Rotation matrices.
- Position vectors of selected points.
- Position and velocity equations of end effector.
- Jacobian matrix.
- Jacobian dot matrix.
- Equation of motion.

Where the lines beginning with an arrow are the results of the code.

The used notations in Autolev software are shown in Figure 2.7 and Table 2.1.
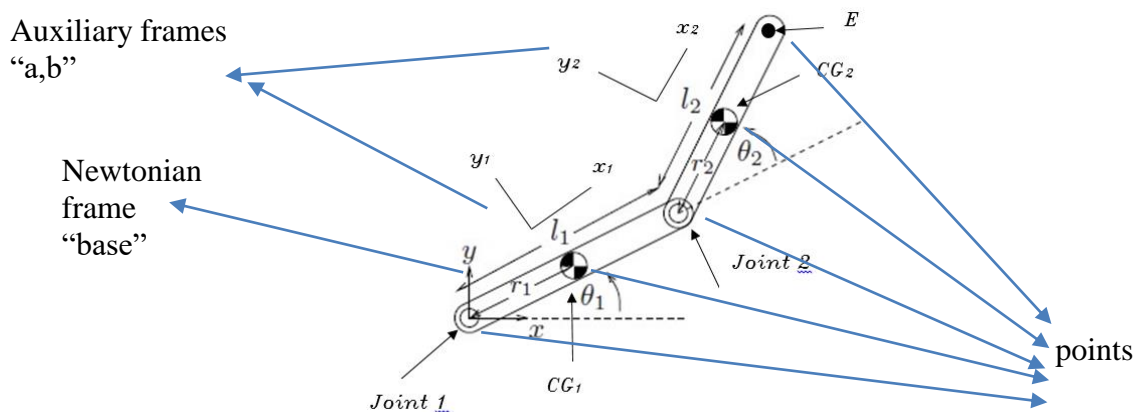
Figure 2.7: Used notations in autolev

Table 2.1: Used notations in Autolev

Constant values	L_1, L_2, r_1, r_2
Variables	- End effectors position (x,y,z) and their derivatives - Joint angles (θ_1, θ_2) and their derivatives

Figure 2.8 shows the declaration of the frames, points, variables and constants.

```
(1) newtonian base
  (2) frames a,b
  (3) points p1,p2,cg1,cg2,e
  (4) variables theta1'',theta2'',x'',y'',z''
  (5) constants L1,r1,L2,r2
```

Figure 2.8: Autolev code initializing

2.2.5.1 Rotation matrices

Figure 2.9 shows the rotation matrices $R_2{}^1$ and $R_E{}^2$.

```
 (6) simprot(base,a,3,theta1)
-> (7) base_a = [COS(theta1), -SIN(theta1), 0; SIN(theta1), COS(theta1), 0; 0, 0, 1]

  (8) simprot(a,b,3,theta2)
-> (9) a_b = [COS(theta2), -SIN(theta2), 0; SIN(theta2), COS(theta2), 0; 0, 0, 1]
```

Figure 2.9: Autolev rotation matrices

2.2.5.2 Position vectors of selected points

Figure 2.10 shows the position vectors,where a_1 is the x axis of auxiliary frame a and a_2 is the y axis of auxiliary axis a and so on.

```
 (10) p_p1_cg1>=r1*a1>
-> (11) p_p1_cg1> = r1*a1>

  (12) p_cg1_p2>=(l1-r1)*a1>
-> (13) p_cg1_p2> = (L1-r1)*a1>

  (14) p_p2_cg2>=r2*b1>
-> (15) p_p2_cg2> = r2*b1>

  (16) p_cg2_e>=(l2-r2)*b1>
-> (17) p_cg2_e> = (L2-r2)*b1>
```

Figure 2.10: Autolev position vectors

2.2.5.3   Position and velocity equations of end effector

Figure 2.11 shows the position and velocity equations of end effector.

```
    (18) x=dot(p_p1_e>,base1>)
-> (19) x = L1*COS(theta1) + L2*COS(theta1+theta2)

    (20) y=dot(p_p1_e>,base2>)
-> (21) y = L1*SIN(theta1) + L2*SIN(theta1+theta2)

    (22) z=dot(p_p1_e>,base3>)
-> (23) z = 0

    (24) x'=dt(x)
-> (25) x' = -L1*SIN(theta1)*theta1' - L2*SIN(theta1+theta2)*(theta1'+theta2')

    (26) y'=dt(y)
-> (27) y' = L1*COS(theta1)*theta1' + L2*COS(theta1+theta2)*(theta1'+theta2')

    (28) z'=dt(z)
-> (29) z' = 0
```

Figure 2.11: Autolev end effector position and derivatives

2.2.5.4   Jacobian matrix

Figure 2.12 shows Jacobian matrix.

```
    (30) j=[d(x,theta1),d(x,theta2);d(y,theta1),d(y,theta2);d(z,theta1),d(z,theta2)]
-> (31) j[1,1] = -L1*SIN(theta1) - L2*SIN(theta1+theta2)
-> (32) j[1,2] = -L2*SIN(theta1+theta2)
-> (33) j[2,1] = L1*COS(theta1) + L2*COS(theta1+theta2)
-> (34) j[2,2] = L2*COS(theta1+theta2)
-> (35) j[3,1] = 0
-> (36) j[3,2] = 0
```

Figure 2.12: Autolev jacobian matrix

2.2.5.5   Jacobian dot matrix

Figure 2.13 shows Jacobian dot matrix.

```
    (37) j_dot=dt(j)
-> (38) j_dot[1,1] = -L1*COS(theta1)*theta1' - L2*COS(theta1+theta2)*(theta1'+theta2')
-> (39) j_dot[1,2] = -L2*COS(theta1+theta2)*(theta1'+theta2')
-> (40) j_dot[2,1] = -L1*SIN(theta1)*theta1' - L2*SIN(theta1+theta2)*(theta1'+theta2')
-> (41) j_dot[2,2] = -L2*SIN(theta1+theta2)*(theta1'+theta2')
-> (42) j_dot[3,1] = 0
-> (43) j_dot[3,2] = 0
```

Figure 2.13: Autolev jacobian dot matrix

2.2.5.6   Equation of motion

Figure 2.14 shows equation of motion by value of torques.

```
   (1) variables theta1'',theta2''
   (2) constants iz1,iz2,m1,m2,l1,l2,r1,r2
   (3) L=0.5*[theta1',theta2']*[iz1+iz2+m1*(r1^2)+m2*(l1^2)+m2*(r2^2)+2*m2*l1*r2*cos(theta2),iz2+m2*(r2^2)+m2*l1*r2*cos(theta2);iz2+m2*(r2^2)+m2*l1*r2*cos(theta2),iz2+m2*(r2^2)]*[theta1';theta2']
-> (4) L[1] = 0.5*(iz1+iz2+m1*r1^2+m2*l1^2+m2*r2^2+2*l1*m2*r2*COS(theta2))*theta1'^2 + 0.5*theta2'*((iz2+m2*r2^2)*theta2'+2*(iz2+m2*r2^2+l1*m2*r2*COS(theta2))*theta1')

   (5) L_theta1_dot=d(L,theta1')
-> (6) L_theta1_dot[1] = (iz2+m2*r2^2+l1*m2*r2*COS(theta2))*theta2' + (iz1+iz2+m1*r1^2+m2*l1^2+m2*r2^2+2*l1*m2*r2*COS(theta2))*theta1'

   (7) L_theta1=d(L,theta1)
-> (8) L_theta1 = [0]

   (9) tau1=dt(L_theta1_dot)
-> (10) tau1[1] = (iz2+m2*r2^2+l1*m2*r2*COS(theta2))*theta2'' + (iz1+iz2+m1*r1^2+m2*l1^2+m2*r2^2+2*l1*m2*r2*COS(theta2))*theta1'' - 2*l1*m2*r2*SIN(theta2)*theta1'*theta2' - l1*m2*r2*SIN(theta2)*theta2'^2

   (11) L_theta2_dot=d(L,theta2')
-> (12) L_theta2_dot = [(iz2+m2*r2^2)*theta2' + (iz2+m2*r2^2+l1*m2*r2*COS(theta2))*theta1']

   (13) L_theta2=d(L,theta2)
-> (14) L_theta2 = [-l1*m2*r2*SIN(theta2)*theta1'*(theta1'+theta2')]

   (15) tau2=dt(L_theta2_dot)-L_theta2
-> (16) tau2[1] = l1*m2*r2*SIN(theta2)*theta1'^2 + (iz2+m2*r2^2)*theta2'' + (iz2+m2*r2^2+l1*m2*r2*COS(theta2))*theta1''
-------------
```

Figure 2.14: Autolev equation of motion

Chapter 3 **Mechatronics approach**

3.1 Mechanical design

3.1.1 The proposed mechanical design

This subsection introduces the proposed mechanical design of underactuated two link manipulator. The manipulator is designed using SOLIDWORKS software. [12]

Multiple views are shown in the following figures (Figure 3.1 to Figure 3.5).
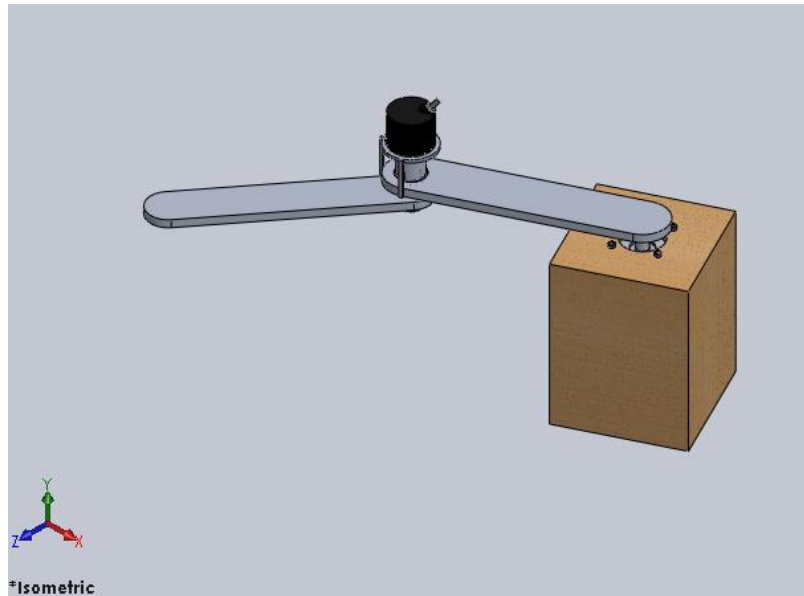
Figure 3.1: Isometric view of the design
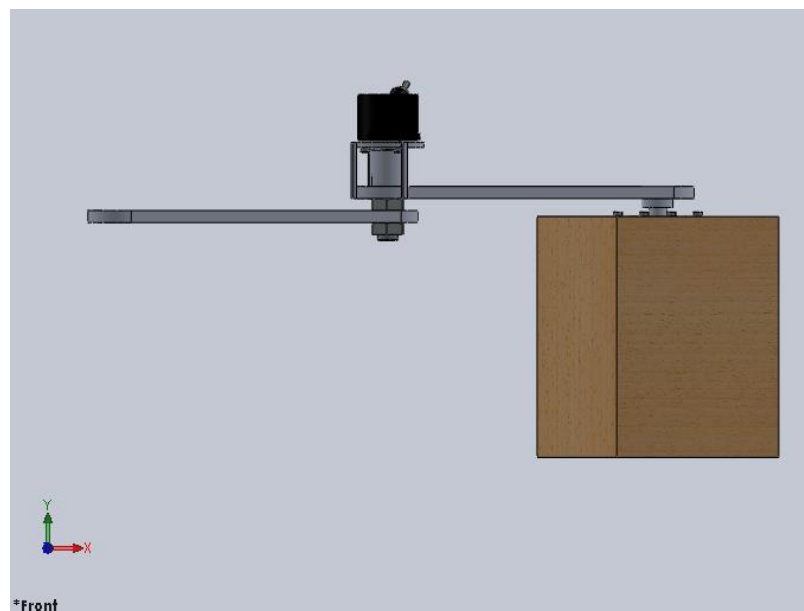
Figure 3.2: Front view of the design

Figure 3.3: Top view of the design

Figure 3.4: Exploded view of the design
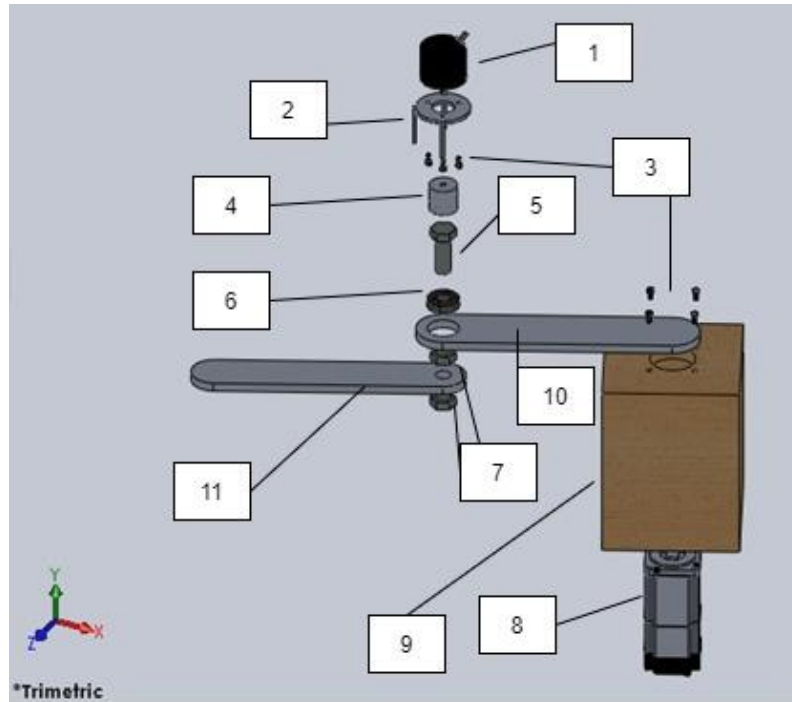
Figure 3.5: Exploded view of the design with numbers

Table 3.1 shows the names of the numbered parts in Figure 3.5.

Table 3.1: Names of the numbered parts

Part number	Part name	Additional information
1	Rotary encoder	Nemicon 0SS-03-2
2	Rotary encoder support	Aluminum 1060 alloy
3	Fasteners of servo motor and encoder	M3 size
4	Coupler of bolt and encoder	Bronze alloy
5	Coupling bolt	M6 size
6	Deep groove ball bearing	NSK company 6003 ZZ VV DDU
7	Fastening nuts	M6 size
8	AC servo motor	Pittman GM9413 DC gearmotor
9	AC servo motor housing	Wood
10	First link	Aluminum 1060 alloy
11	Second link	Aluminum 1060 alloy

3.1.2   Maximum deflection, stress and factor of safety analysis

The two links are made of aluminum 1060 alloy, and their thickness is 10 mm. That yields a first link mass of 0.462 Kg and second link mass of 0.384 Kg, these masses are considered as gravity force.

The encoder holder has a mass of 0.03 Kg, the encoder has a mass of 0.082 Kg, the coupler of bolt and encoder has a mass of 0.05 Kg, and the bolt and its nuts have a mass of 0.177 Kg ,that yields to a force equals

$$W = 3.31N \downarrow$$

Von Mises stress is performed in SOLIDWORKS by applying W at the end effector position, the result is shown in Figure 3.6.
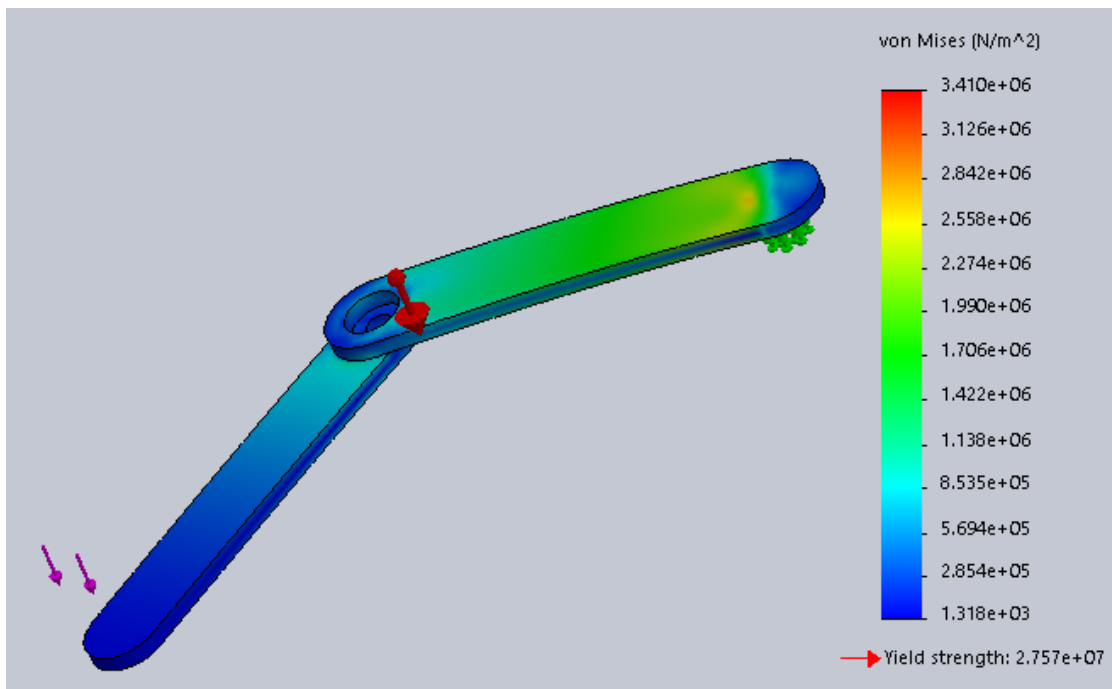
Figure 3.6: Results of Von Mises stress

The result show that the maximum Von Mises stress due to bending ≈ 2.2 MPa in the links.

The deflection along model due to bending is shown in Figure 3.7 .
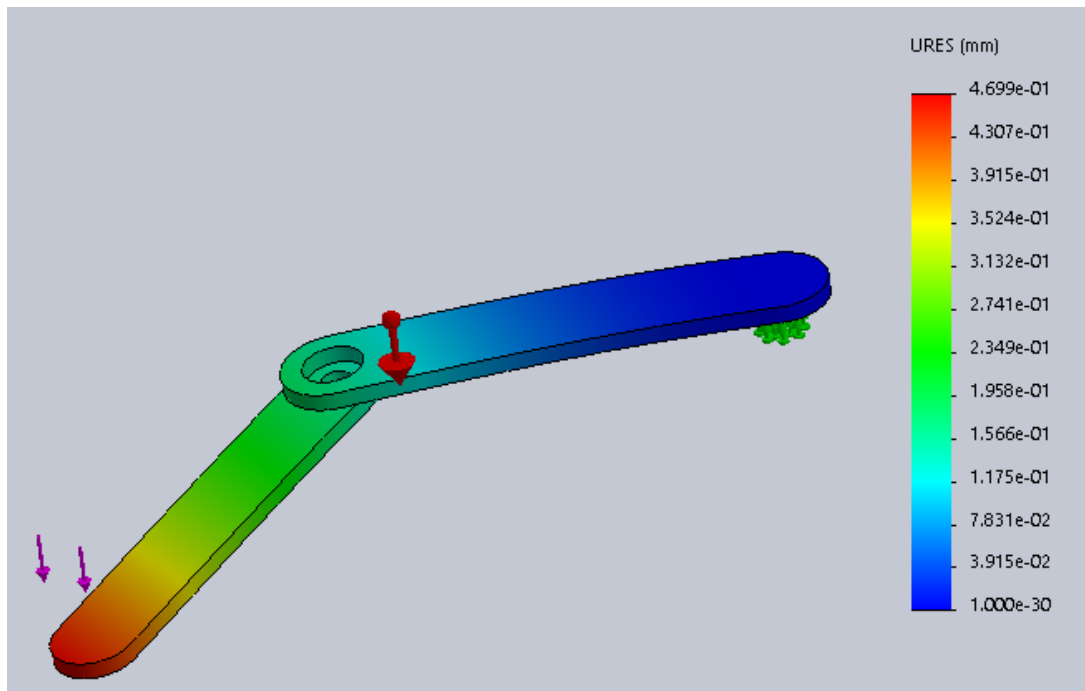
Figure 3.7: Results of deflection test

The maximum deflection due to bending is 0.47 mm, which is an acceptable value and doesn't affect the design.

The factor of safety results are shown in Figure 3.8, the results show that the minimum factor of safety is 8.1, this high number is due to the light weight of the parts pinned on the system and the strength of Aluminum used in the system, thus, the design is considered as safe design.
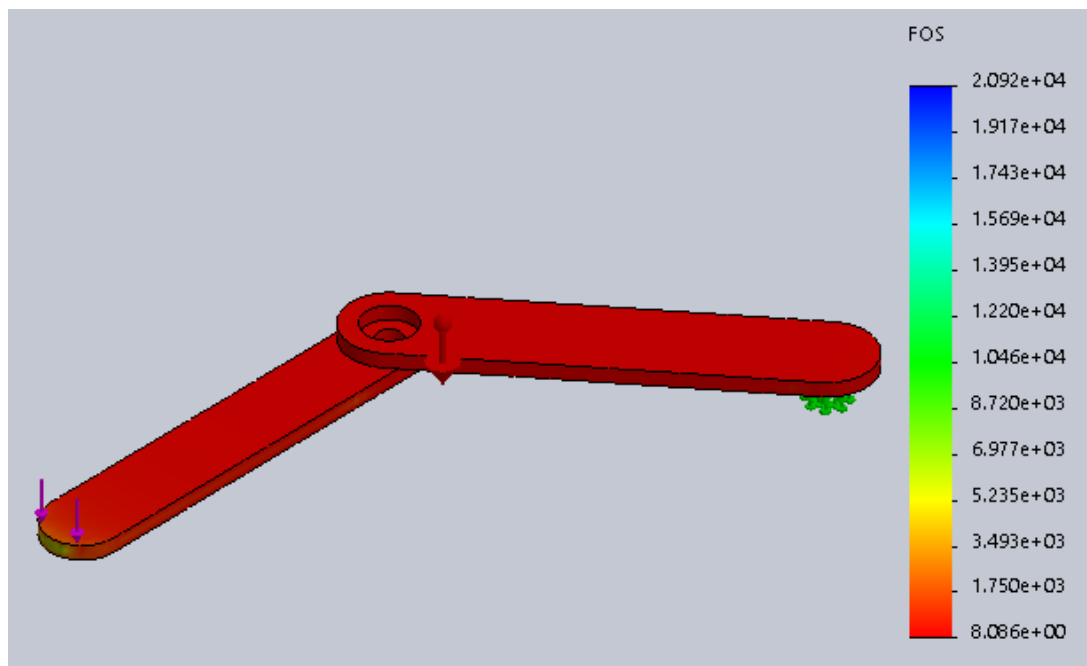
Figure 3.8: Results of factor of safety

3.1.3 Dimensions of the parts

The following figures (Figure 3.9 to Figure 3.12) are the detailed drawings of the first link, second link and the coupler of bolt and encoder, where all dimensions in millimeters (mm). [12]
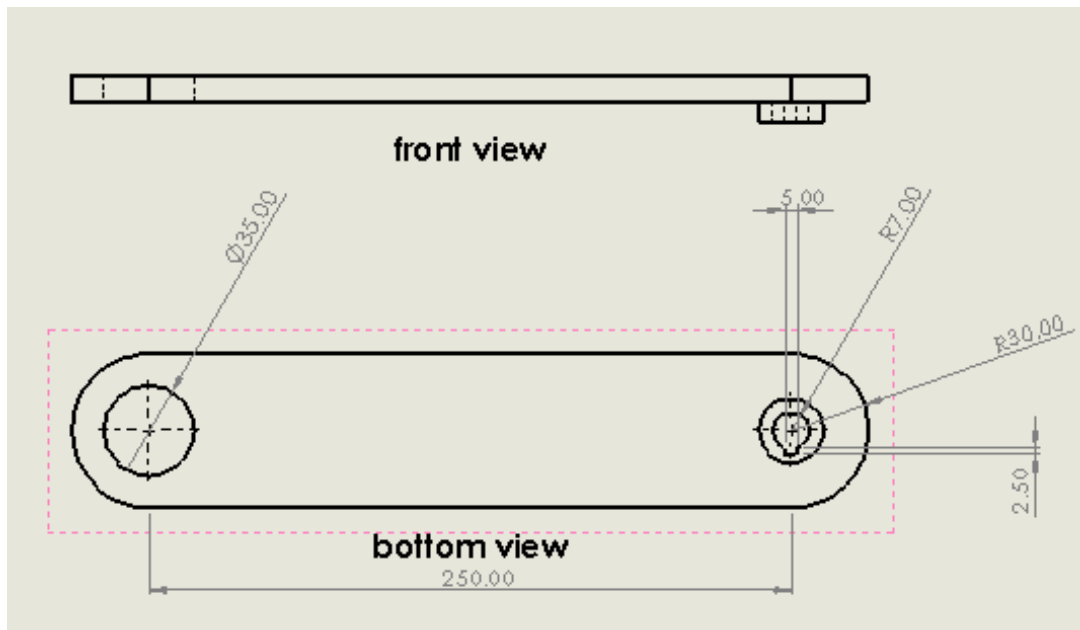
Figure 3.9: First part of first link dimensions
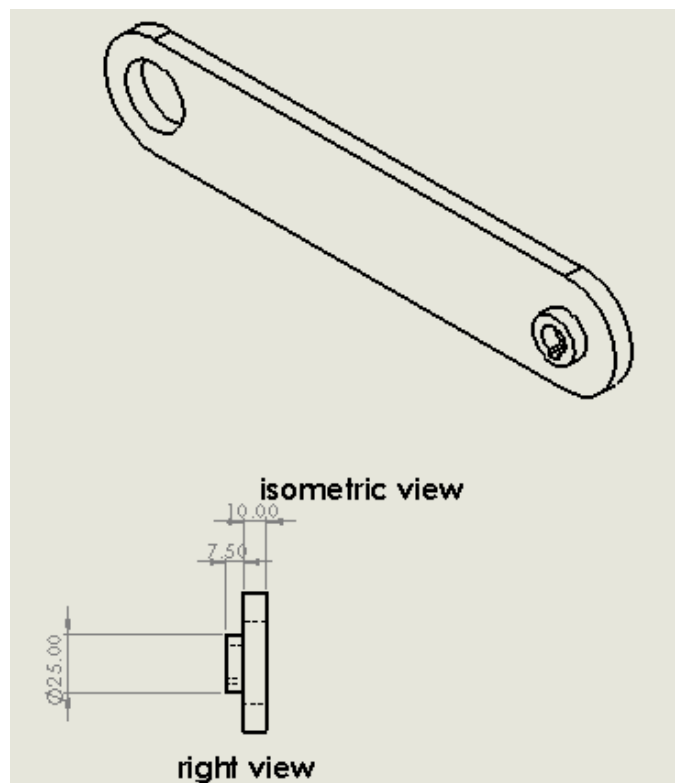
Figure 3.10: Second part of first link dimensions
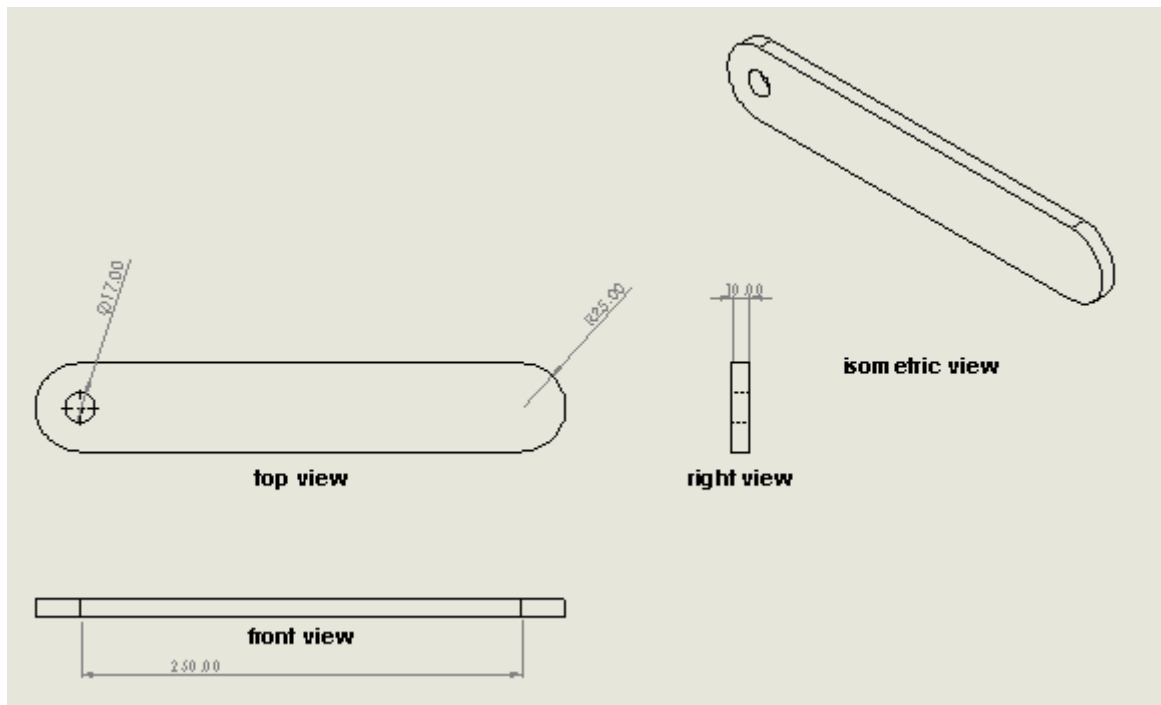
Figure 3.11: Second link dimensions
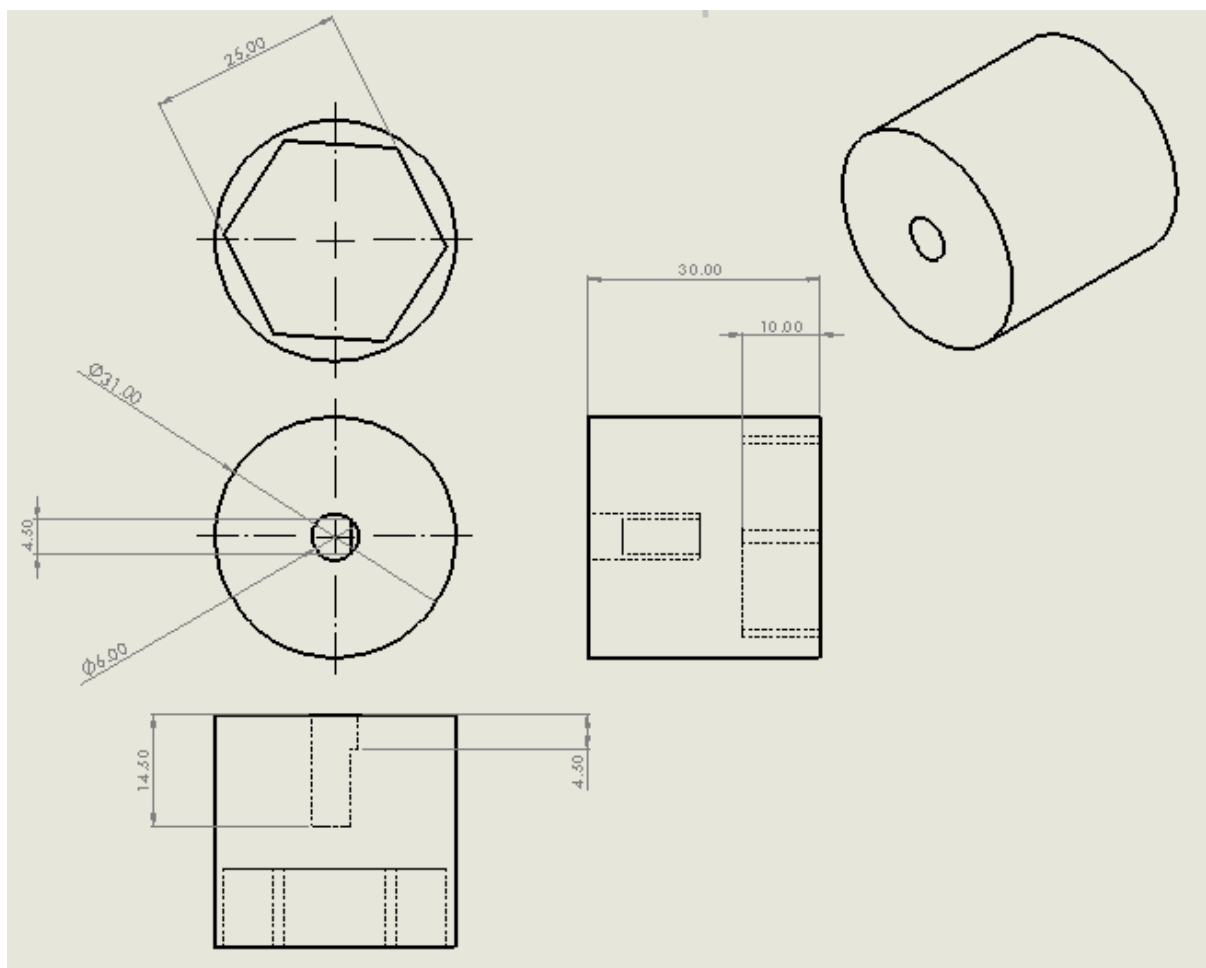
Figure 3.12: Dimensions of coupler of bolt and encoder

3.1.4   Motor selection

The trajectory of the end effector is shown in Figure 3.13, this trajectory requires joints trajectory shown in Figure 3.14.
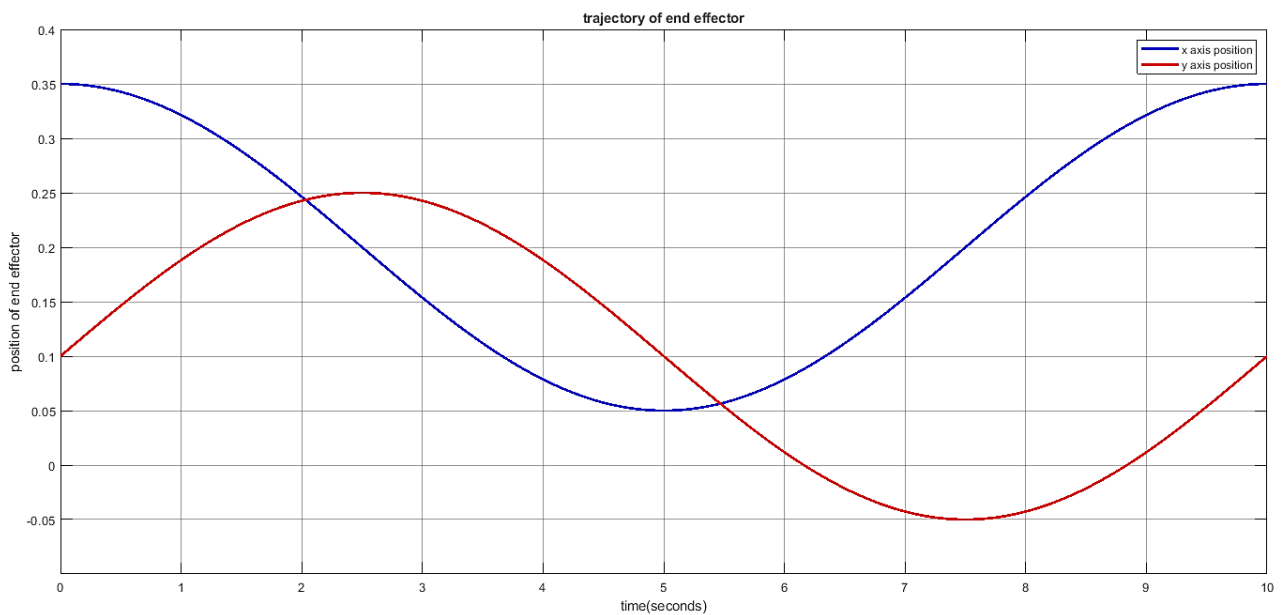
Figure 3.13: Trajectory of the end effector
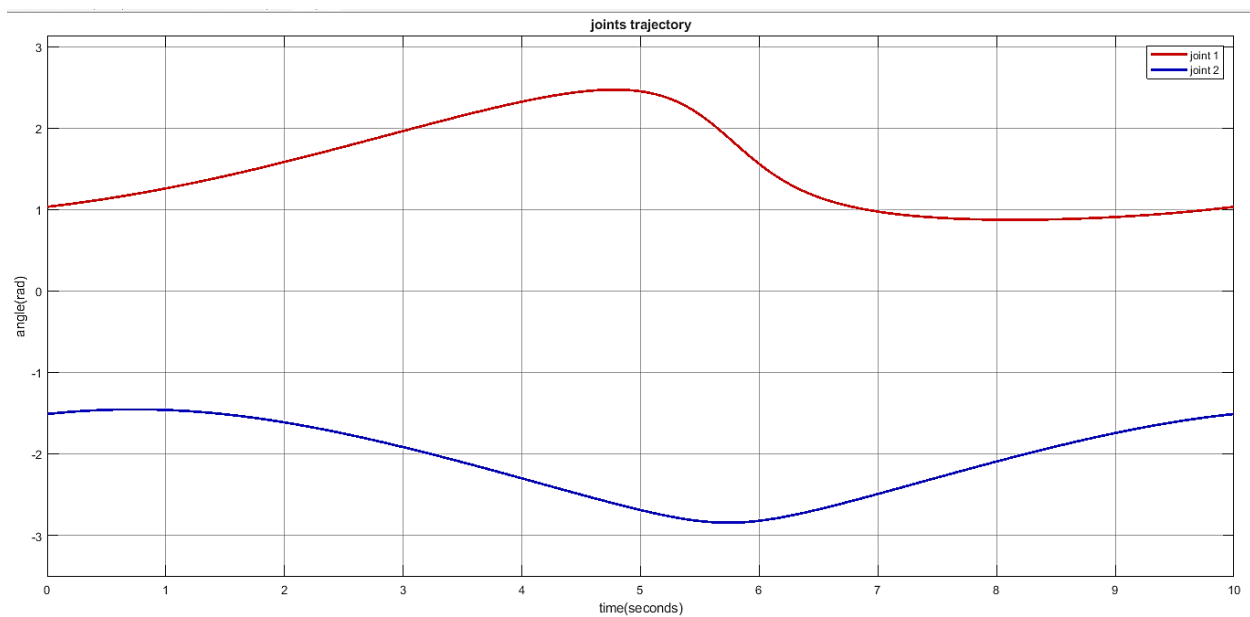
Figure 3.14: Joints trajectory

Then, the maximum angular accelerations of the joints are

$$\alpha_1 = 17\,rad\,/\sec^2$$
$$\alpha_2 = 15\,rad\,/\sec^2$$

31

And the mass moments of inertia about center of gravity based on the CAD model are

$$I_1 = 33.591x\,10^{-4}\,Kg\,.m^2$$
$$I_2 = 27.161x\,10^{-4}\,Kg\,.m^2$$

Then, the parallel axis theorem in// is used to obtain the mass moments of inertia of each link about joint 1 and 2 [13].

$$I = I + md^2 \qquad\qquad (38)$$

Where m is the mass of the link and d is the distance between the link's center of gravity and the position of the joint, accordingly, using the following CAD values of the masses and distances as $m_1 = 0.462Kg\,,m_2 = 0.384Kg\,,d_1 = d_2 = 0.125m$.

Then, the mass moments of inertia of each link about joint 1 and 2 are

$$I_1 = 105.778x\,10^{-4}\,Kg\,.m^2$$
$$I_2 = 87.161x\,10^{-4}\,Kg\,.m^2$$

According to previous results, the required torques are

$$\tau_1 = 0.18N\,.m$$
$$\tau_2 = 0.13N\,.m$$

One motor will be used, thus, the selected motor should deliver at least 0.32 N.m torque to actuate the manipulator properly.

3.2   Electrical design

3.2.1   Electrical components

The main factors for choosing the components are their price, availability of them in university laboratories and local market or members of the team (some parts could be purchased by a member for a previous course project), ability to meet requirements of the project, and ease of programming. Each component is presented in this section.

1. Motor

A Pittman GM9413 DC gearmotor is used as an actuator for the first active joint some specifications are shown in Table 3.2 the datasheet of the motor is shown in Appendix A.

Table 3.2: Specifications of the motor

Voltage	12 VDC
Maximum current	5.54 A
Rated speed	285 rpm
Rated torque	٠.45 N.m
Peak torque	2.17 N.m
Gear ratio	19.7:1
Weight	0.431 Kg

32

2. Encoders

Each joint of the manipulator is attached to an encoder to provide a feedback of the joint angles to the computer at real time.

The two encoders are 2 channels encoders, the channels are A and B.

The specifications and connection of first joint encoder are in Table 3.3.

Table 3.3: Specifications of the first joint encoder

Model	SAIET RIH44-36022
Power supply	5VDC/12VDC
Resolution	400 pulse/revolution
Connection	- Red: power supply - Black: ground - White: signal A - Green: signal B

The specifications and connection of second joint encoder are in Table 3.4.

Table 3.4: Specifications of the second encoder

Model	Nemicon 0SS-03-2
Power supply	4.5VDC-13.2VDC
Resolution	300 pulse/revolution
Connection	- Red: power supply - Black: ground - Green: signal A - White: signal B

3. Motor driver

A L298N Dual H-Bridge Motor Driver is used to control the motor up to 2A in both directions.

The commands of direction and speed of motor are fed to the driver using a microcontroller or a data acquisition card.

Some specifications of the driver are shown in Table 3.5 [14], the datasheet of the driver is shown in Appendix B.

Specifications of the motor driver

Input voltage	3.2VDC-40VDC
Power supply	5VDC-35VDC
Peak current	2A
Operating cuurent	0-36mA
Maximum power consumption	20W(at Temperature=75 Celsius)

4. Data Acquisition Card

The components of the system in this project are integrated with each other using high level programming language in Simulink software with NI PCI-6221 data acquisition card, this card is installed inside a desktop computer box and it has a screw terminal connector shown in Figure 3.15 to easily connect the wires of various devices with the card.

The pinout of the card is shown in Figure 3.16,the NI PCI-6221 provides many functions as digital I/O, analog I/O, Programmable Function Interface (PFI) which can be individually configured as a PFI terminal or a digital I/O terminal,two general purpose counters/timers,and a frequency generator[15], the datasheet of the NI PCI-6221 is shown in Appendix C.

Figure 3.15: NI-6221 screw terminal connector

AI 0	68	34	AI 8
AI GND	67	33	AI 1
AI 9	66	32	AI GND
AI 2	65	31	AI 10
AI GND	64	30	AI 3
AI 11	63	29	AI GND
AI SENSE	62	28	AI 4
AI 12	61	27	AI GND
AI 5	60	26	AI 13
AI GND	59	25	AI 6
AI 14	58	24	AI GND
AI 7	57	23	AI 15
AI GND	56	22	AO 0
AO GND	55	21	AO 1
AO GND	54	20	NC
D GND	53	19	P0.4
P0.0	52	18	D GND
P0.5	51	17	P0.1
D GND	50	16	P0.6
P0.2	49	15	D GND
P0.7	48	14	+5 V
P0.3	47	13	D GND
PFI 11/P2.3	46	12	D GND
PFI 10/P2.2	45	11	PFI 0/P1.0
D GND	44	10	PFI 1/P1.1
PFI 2/P1.2	43	9	D GND
PFI 3/P1.3	42	8	+5 V
PFI 4/P1.4	41	7	D GND
PFI 13/P2.5	40	6	PFI 5/P1.5
PFI 15/P2.7	39	5	PFI 6/P1.6
PFI 7/P1.7	38	4	D GND
PFI 8/P2.0	37	3	PFI 9/P2.1
D GND	36	2	PFI 12/P2.4
D GND	35	1	PFI 14/P2.6

CONNECTOR 0
(AI 0-15)

TERMINAL 68 — TERMINAL 34

TERMINAL 35 — TERMINAL 1

NC = No Connect

Figure 3.16: NI-6221pinout

5. Arduino microcontroller

An Arduino Mega 2560 is a microcontroller based on ATmega2560 processor, it has 54 pins which are divided into 15 pins for PWM output, 16 pins for analog input, 4 pins for hardware serial ports, and all 54 pins can be used as digital I/O [16]. The specifications of the microcontroller is shown in Table 3.6 and the pinout is shown in Figure 3.17.

35

Table 3.6: Specifications of Arduino microcontroller

Microcontroller	ATmega2560
Operating voltage	5VDC
Input voltage	6VDC-20VDC
Digital I/O pins	54(15 can provide PWM output)
Analog input pins	16
DC current per I/O pin	20mA
DC current for 3.3V pin	50mA
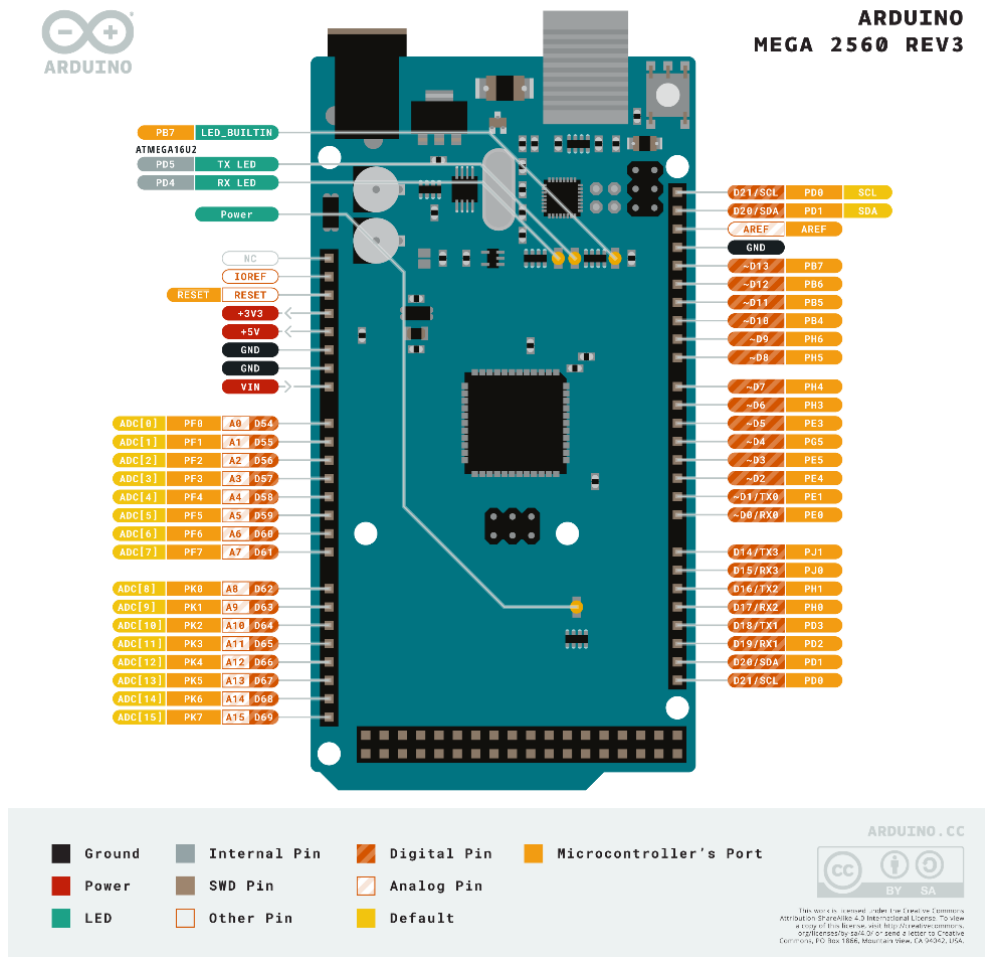
Figure 3.17: Arduino Mega 2560 pinout

Since the two counters in NI PCI-6221 are occupied for the two joints encoders, PWM generation is not applicable because it is based on the counters, therefore, the Arduino Mega 2560 is used as analog to PWM converter to convert the analog speed value from NI PCI-6221 to PWM speed value to meet the L298N driver's input.

3.2.2   Electrical connection

The electrical connection of the devices of the manipulator is shown in Figure 3.18.
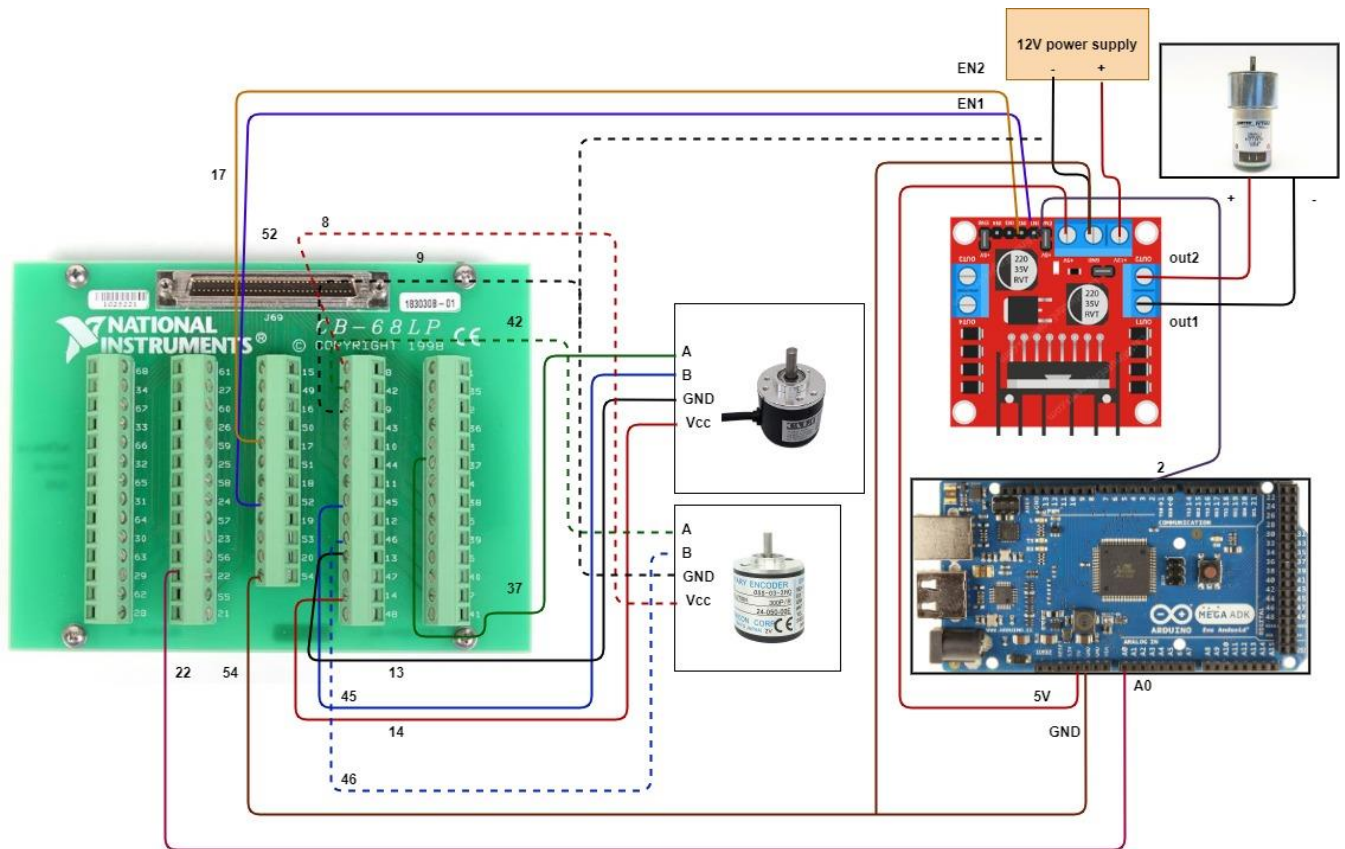
Figure 3.18: Electrical connection of the system

The NI-6221 Programmable Function Interface (PFI) ports can be programmed as counters for the joints encoders as shown in Table 3.7 [15].

Table 3.7: Default NI-6221 Counter/Timer Pins

Counter/Timer Signal	Default Connector 0 Pin Number (Name)
CTR 0 SRC	37 (PFI 8)
CTR 0 GATE	3 (PFI 9)
CTR 0 AUX	45 (PFI 10)
CTR 0 OUT	2 (PFI 12)
CTR 0 A	37 (PFI 8)
CTR 0 Z	3 (PFI 9)
CTR 0 B	45 (PFI 10)
CTR 1 SRC	42 (PFI 3)
CTR 1 GATE	41 (PFI 4)
CTR 1 AUX	46 (PFI 11)
CTR 1 OUT	40 (PFI 13)
CTR 1 A	42 (PFI 3)
CTR 1 Z	41 (PFI 4)
CTR 1 B	46 (PFI 11)
FREQ OUT	1 (PFI 14)

3.3   Implementation of the project

The project is implemented using the parts in this chapter, a mass of Iron is pinned to the motor to provide stability for the center of rotation of the links, Figure 3.19 to Figure 3.22 show the implemented project.
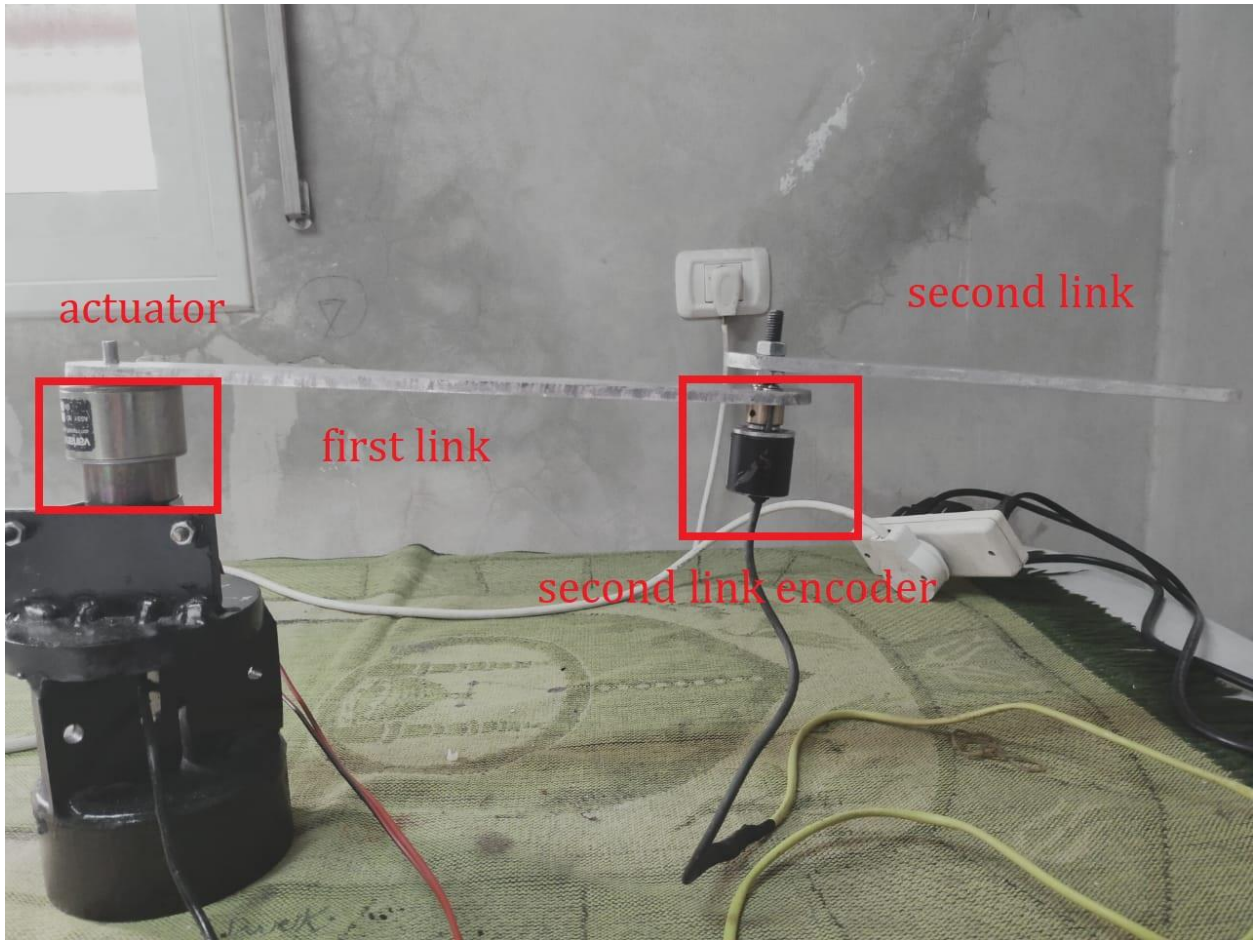
Figure 3.19: Side view of the project
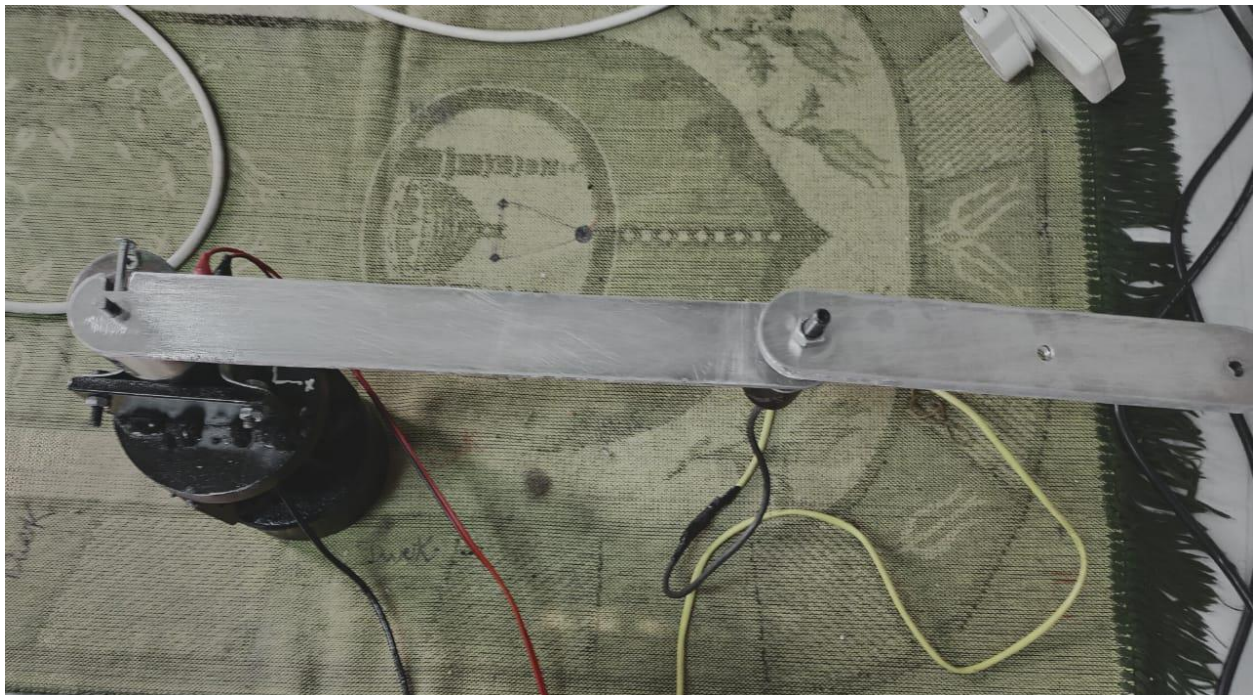
Figure 3.20: Upper view of the project

Figure 3.21: First link encoder in the project
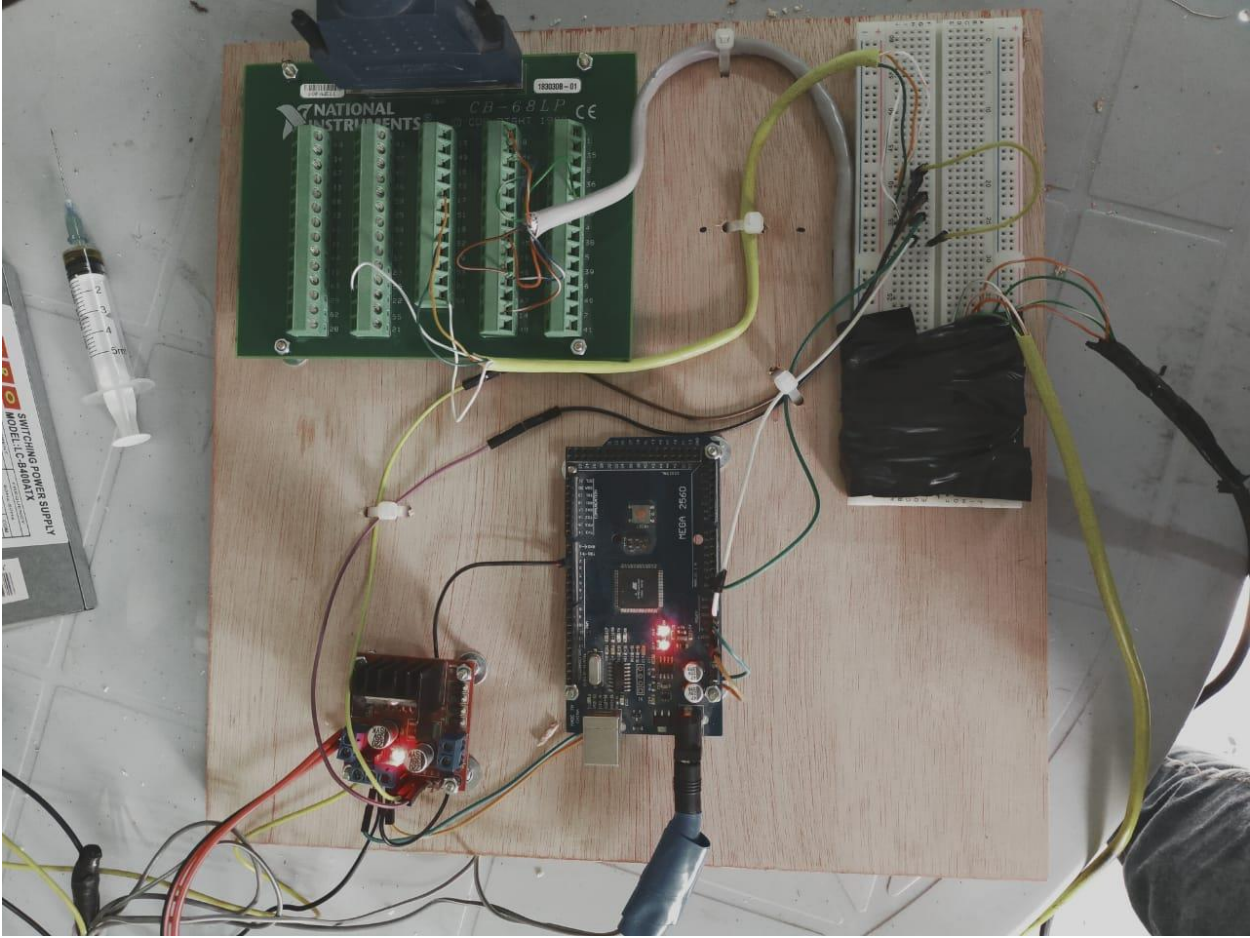
Figure 3.22: Electrical parts of the project

Chapter 4   Disturbance estimation

This chapter introduce a linear robot model of the dynamics. This can be achieved by representing the effects of all nonlinear terms as a lumped disturbance vector. This vector can be estimated using adaptive Kalman filter (AKF) algorithm and directly applied to the model. All terms in the matrices described in the dynamic model in (29) are nonlinear except the inertia matrix which consist of linear and nonlinear terms, the linear part represents the inertia matrix (I) which consists of the inertia of the two links [17], thus, so (29) can be written as

$$I\ddot{\theta} + \left(M\left(\theta \right) - I \right)\ddot{\theta} + C\left(\theta, \dot{\theta} \right)\dot{\theta} = \tau \tag{39}$$

The inertia matrix M is

$$M\left(\theta \right) = I + \tilde{M}\left(\theta \right) \tag{40}$$

We can call the terms with uncertainty or non-linear and un-modeled parts as lumped non-linear disturbance ζ, then (36) can be written as

$$I\ddot{\theta} + \zeta = \tau \tag{41}$$

Where ζ is given by

$$\zeta = \tilde{M}\left(\theta \right)\ddot{\theta} + C\left(\theta, \dot{\theta} \right)\dot{\theta} \tag{42}$$

(39) is estimated and used in control law for disturbance compensation .

(38) can be written in state space representation by choosing the joint angular displacements and velocities as a states.

$$x = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}^T = \begin{bmatrix} \theta_1 & \theta_2 & \dot{\theta}_1 & \dot{\theta}_2 \end{bmatrix}^T \tag{43}$$
$$\dot{x} = Ax + Bu + F_d\zeta$$

$$\dot{x} = \begin{bmatrix} 0_{2\times2} & I_2 \\ 0_{2\times2} & 0_{2\times2} \end{bmatrix} x + \begin{bmatrix} 0_{2\times2} \\ I^{-1} \end{bmatrix} u + \begin{bmatrix} 0_{2\times2} \\ -I^{-1} \end{bmatrix} \zeta \tag{44}$$

Where x is the state vector, $0_{q\times\beta}$ is zero matrix of size $q \times \beta$, I_σ is an identity matrix of size σ, and $u = \tau$. The measurement vector can be represented as

$$y = Cx$$
$$y = \begin{bmatrix} I_2 & 0_{2\times2} \end{bmatrix} x \tag{45}$$

For this system, the states x_1 and x_2 are measurable, the states x_3 and x_4 are pseudo measurable, while the vector ζ is not measurable. This vector will be estimated using AKF and added directly to the output of the controller to eliminate the effects of nonlinear terms in the system. [18]

The AKF is an adaptive observer that can be used to estimate the disturbance by augmenting the disturbance with the states as an extended state or system. It consists of two layers per each sample time, prediction step and updating layer. The main advantage of this algorithm that does not require the noise covariance matrices, instead, it needs just the initial values of them, and the algorithm will update it according to the behavior of error between the estimated states and the measured states. [17]

The estimator algorithm is active under the following assumptions

- The process and measurement noises are assumed to be independent and mutually uncorrelated with the given means and covariances.

- The inputs are considered to be piecewise constant over the sampling time interval.
- The noise covariances are considered to be constant.
- The process and measurements have the same sampling time.

To use the AKF estimator algorithm, the disturbance should be augmented with system states as an extended states, accordingly, (41) is rewritten in the form

$$\begin{bmatrix} \dot{x} \\ \dot{\zeta} \end{bmatrix} = A_e \begin{bmatrix} x \\ \zeta \end{bmatrix} + B_e u$$

$$\begin{bmatrix} \dot{x} \\ \dot{\zeta} \end{bmatrix} = \begin{bmatrix} A & F_d \\ 0_{2\times4} & 0_{2\times2} \end{bmatrix} \begin{bmatrix} x \\ \zeta \end{bmatrix} + \begin{bmatrix} B \\ 0_{2\times2} \end{bmatrix} u \qquad (46)$$

The measurement vector consists of the joint angles that are measured using joint encoders one of them attached to the active joint and the other one to the passive joint. In addition to that, differentiating the angular position numerically using backward Euler formula gives valuable information. Hence $\dot{\theta}$ is considered as pseudo measured. (43) is discretized and rewritten it using backward Euler formula and yields

$$\begin{bmatrix} x(k) \\ \zeta(k) \end{bmatrix} = A_d \begin{bmatrix} x(k-1) \\ \zeta(k-1) \end{bmatrix} + B_d u(k) + v(k-1) \qquad (47)$$

Where

$$A_d = \begin{bmatrix} I_2 & TI_2 & 0_{2\times2} \\ 0_{2\times2} & I_2 & -TI^{-1}I_2 \\ 0_{2\times2} & 0_{2\times2} & I_2 \end{bmatrix}, B_d = \begin{bmatrix} 0_{2\times2} \\ TI^{-1} \\ 0_{2\times2} \end{bmatrix} \qquad (48)$$

Where $H_{new} = [I_4 \quad 0_{4\times2}]$ is the output matrix, T is the sampling time and k is the time index. $\upsilon \in \mathbb{R}^9$ and $v \in \mathbb{R}^6$ are the zero mean Gaussian process and measurement noises with covariance matrices Q and R respectively, i.e. $\upsilon \sim N(0, Q)$ and $v \sim N(0, R)$. The covariance matrices Q and R are unknown and have an important effect on Kalman filter estimates. If the given value of Q is much smaller than the true value, then the result is biased estimated states $\hat{x}$ and $\hat{\zeta}$. On the other hand, if the given value of Q is much larger than the true value, then the estimated states $\hat{x}$ and $\hat{\zeta}$ will oscillate around the true value. The advantage of the AKF is that it does not need the values of the noise covariance matrices, just initial values of them are required. Then by its recursive structure, it updates the covariance matrices based on the innovation e between the predicted states and the measured vector. This error is used to update and correct the predicted states through Kalman gain which takes into the consideration the uncertainty in the model through the covariance matrices. The output of this filter are the estimated states $\hat{x}$ and $\hat{\zeta}$.

The AKF requires positive constant N_R and N_Q, initial values of matrices R_0 and Q_0, and an initial value of the estimation error covariance matrix P_0. The AKF algorithm is shown between Equations (46) and (61) respectively. Initial values $\bar{\omega}_0, \bar{e}_0, \hat{x}_0, P_0, N_R, N_Q, Q_0 > 0, R_0 > 0$.

$$\begin{bmatrix} \hat{x}^-(k) \\ \hat{\zeta}^-(k) \end{bmatrix} = A_d \begin{bmatrix} \hat{x}(k-1) \\ \hat{\zeta}(k-1) \end{bmatrix} + B_d u(k) \qquad (49)$$

$$P^-(k) = A_d P(k-1) A_d^T + Q(k-1) \qquad (50)$$

$$e(k) = z(k) - H_{new}\begin{bmatrix} \hat{x}^-(k) \\ \hat{\zeta}^-(k) \end{bmatrix} \qquad (51)$$

$$\alpha_1 = \frac{N_R - 1}{N_R}$$

$$\alpha_2 = \frac{N_Q - 1}{N_Q} \qquad (52)$$

$$\bar{e}(k) = \alpha_1\bar{e}(k-1) + \frac{1}{N_R}e(k) \qquad (53)$$

$$\Delta R(k) = \frac{1}{N_R - 1}(e(k) - \bar{e}(k))(e(k) - \bar{e}(k))^T - \frac{1}{N_R}\left(H_{new}P^-H_{new}^T\right)_{(k)} \qquad (54)$$

$$R(k) = \alpha_1 R(k-1) + \Delta R(k) \qquad (55)$$

$$K(k) = P^-(k)H_{new}^T(H_{new}P^-(k)H_{new}^T + R(k))^{-1} \qquad (56)$$

$$\begin{bmatrix} \hat{x}(k) \\ \hat{\zeta}(k) \end{bmatrix} = \begin{bmatrix} \hat{x}^-(k) \\ \hat{\zeta}^-(k) \end{bmatrix} + K(k)e(k) \qquad (57)$$

$$P(k) = (I - K(k)H_{new})P^-(k) \qquad (58)$$

$$\hat{\omega}(k) = \begin{bmatrix} \hat{x}(k) \\ \hat{\zeta}(k) \end{bmatrix} - \begin{bmatrix} \hat{x}^-(k) \\ \hat{\zeta}^-(k) \end{bmatrix} \qquad (59)$$

$$\bar{\omega}(k) = \alpha_2\bar{\omega}(k-1) + \frac{1}{N_Q}\hat{\omega}(k) \qquad (60)$$

$$\Delta Q(k) = \frac{1}{N_Q}\left(P(k) - A_d P(k)A_d^T\right) + \frac{1}{N_Q - 1}(\hat{\omega}(k) - \bar{\omega}(k))(\hat{\omega}(k) - \bar{\omega}(k))^T \qquad (61)$$

$$Q(k) = \alpha_2 Q(k-1) + \Delta Q(k) \qquad (62)$$

$$Q(k-1) = |diag(Q(k))| \qquad (63)$$

$$R(k-1) = |diag(R(k))| \qquad (64)$$

Where $(.)^-$ and $(.)$ stand for the prior and posterior estimates, respectively. K is the Kalman gain. $z \in \mathbb{R}^6$ is the measurement vector and $\hat{\omega}$ is the state error.

Implementation note: for a noisy system, it is much better to give more weight to the previous known values $(R(k-1),\ Q(k-1))$ than the current noisy reading, and this is achieved by selecting big N_R and/or N_Q. In the same context, small N_R and/or N_Q give more weight to the current reading $(\Delta R_k, \Delta Q_k)$ for less noisy readings.

The control block diagram with Kalman Filter is shown in Figure 4.1 and a full detailed block diagram is shown in Figure 4.2 and the MATLAB m files of the functions are in Appendix D.
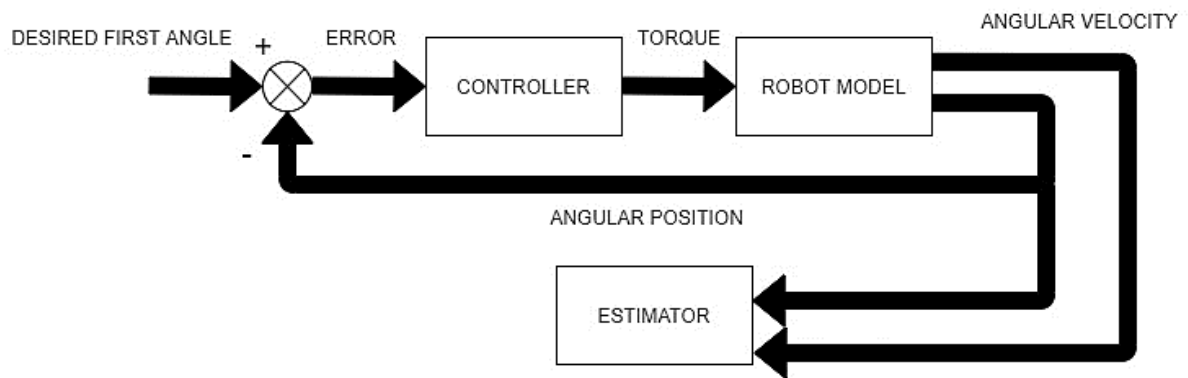
Figure 4.1: Block diagram of the system with estimator
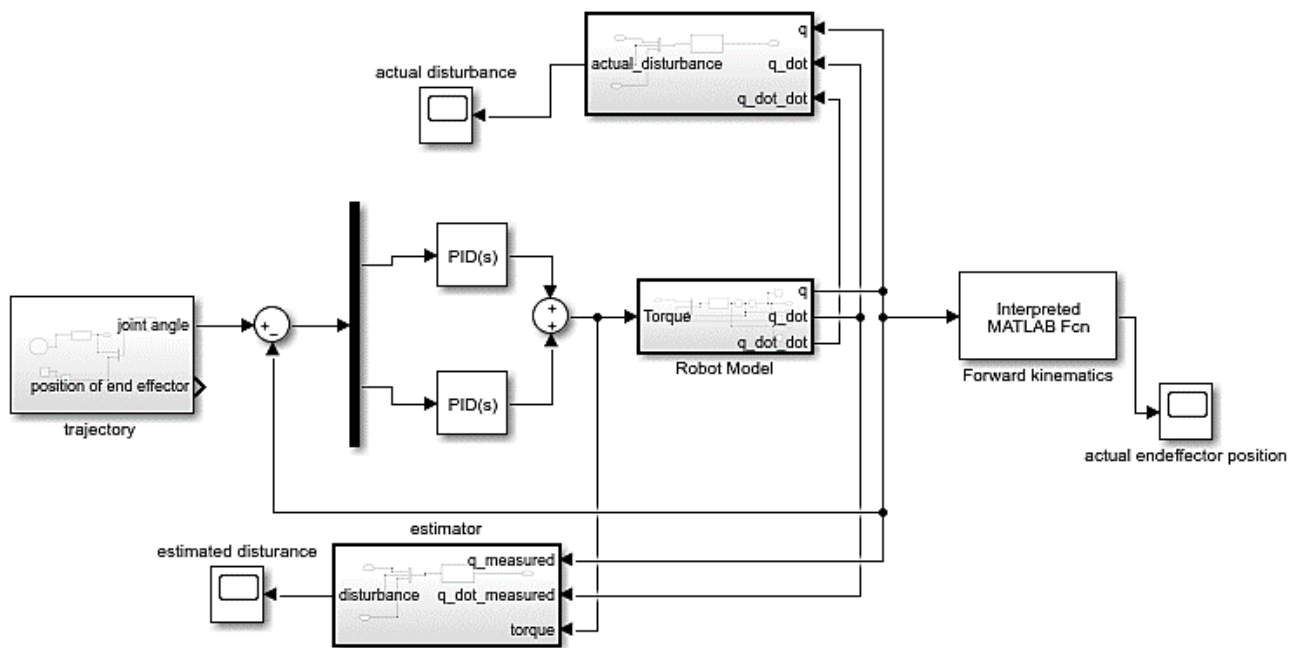
Figure 4.2: Block diagram of the system with estimator in simulink

The validity of the estimator is determined by how accurate it estimates the disturbance, when the difference (error) between the actual and estimated disturbances is zero, then estimator calculates accurately, Figure 4.3 shows the error between the disturbances of system in Figure 4.2.
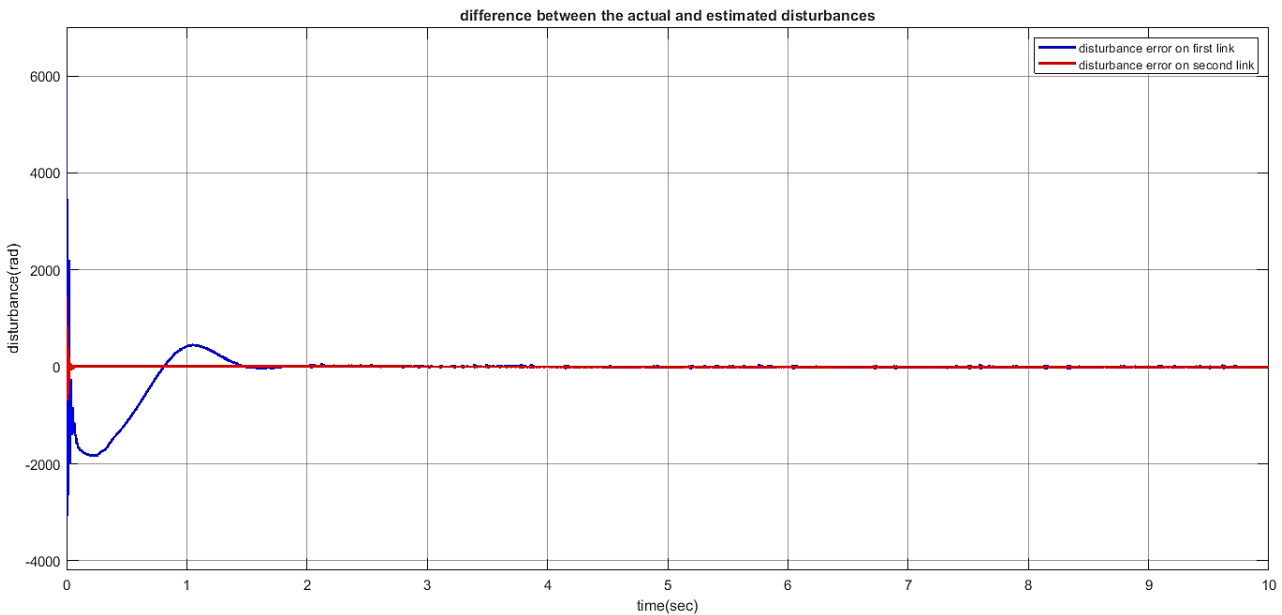
Figure 4.3: Difference between actual and estimated disturbances

Chapter 5   Controller design and implementation

5.1 Introduction

This chapter introduces the control method of planar 2-DOF underactuated manipulator. Manipulator's controllers decide the joint torques/forces that are needed to make the mechanical structure track a desired trajectory with time as shown in Figure 3.13 with a desired dynamic performance. The design process of these controllers will be based on the dynamic model of the manipulator in Chapter 2 [8]. The control scheme will be in the joint space, where the desired end-effector pose will be mapped to the joint variables, theses joint variables will act as the reference for the controller. The general scheme of joint space control is shown in Figure 5.1.
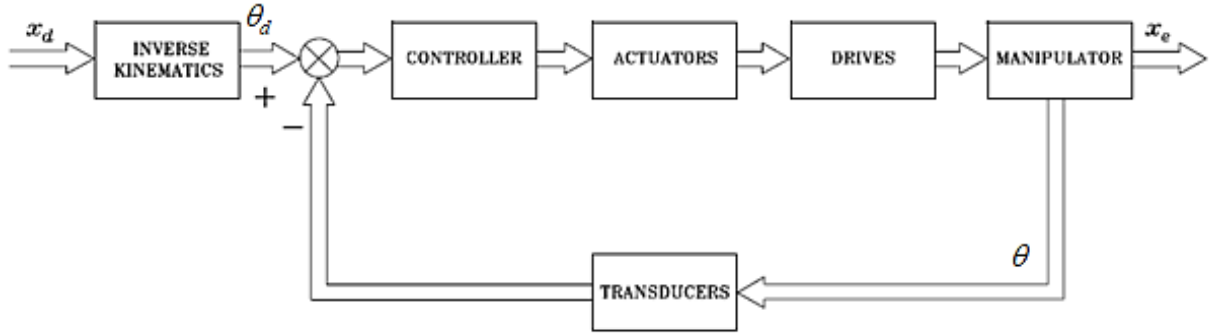
Figure 5.1: Block diagram of joint angles control

5.2 Controllability of the system

This section introduces the check of controllability of the system, the controllability indicates the ability of the input of the system to take the state variables of the system from initial state to a desired state.

5.2.1 Scientific background

The scientific background of the system's controllability is recalled from [19]. Note that the angle θ_i mentioned in this section is the angle between i^{th} link and the positive x axis (base coordinate frame).

The authors of [4] studied the linear (local) controllability of a robot with a single passive joint around a trajectory S_ω described by

$$S_\omega = \left\{ (\theta, \dot{\theta}) \mid \theta_1 = \theta_2, \dot{\theta}_1 = \dot{\theta}_2 = \omega, \omega \neq 0 \right\} \tag{65}$$

Assume the base coordinate frame rotates at constant angular velocity ω and let

$$\psi = \begin{pmatrix} \psi_1 & \psi_2 \end{pmatrix}^T \in \mathbb{R}^2 \tag{66}$$

Be the vector of angles with respect to this frame and satisfy

$$\psi_i = \theta_i - \omega t \tag{67}$$

Then (65) can be expressed as

$$S_\omega = \left\{ (\psi, \dot{\psi}) \mid \psi_1 = \psi_2, \dot{\psi}_1 = \dot{\psi}_2 = 0 \right\} \tag{68}$$

Expressing the motion equation as

$$\sum_{j=1}^{n} d_{ij} \cos(\theta_i - \theta_j)\ddot{\theta}_j + \sum_{j=1}^{n} d_{ij} \sin(\theta_i - \theta_j)\dot{\theta}_j^{\,2} = \tau_i - \tau_{i+1} \tag{69}$$

$$i = 1, 2$$

With $n = 2$, $\tau_{n+1} = 0$ and $\tau_2 = 0$, (69) yields

$$\begin{pmatrix} d_{11} & d_{12}\cos(\theta_1 - \theta_2) \\ d_{21}\cos(\theta_2 - \theta_1) & d_{22} \end{pmatrix}\begin{pmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{pmatrix} + \begin{pmatrix} 0 & d_{12}\sin(\theta_1 - \theta_2)\dot{\theta}_2 \\ d_{21}\sin(\theta_2 - \theta_1)\dot{\theta}_1 & 0 \end{pmatrix}\begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} = \begin{pmatrix} \tau_1 \\ 0 \end{pmatrix} \tag{70}$$

Where

$$d_{ii} = I_i + m_i r_i^{\,2} + l_i^{\,2} \sum_{k=i+1}^{n} m_k$$

$$d_{ij} = d_{ji} = m_j l_i r_j + l_i l_j \sum_{k=j+1}^{n} m_k \tag{71}$$

And using (67), (69) becomes

$$\sum_{j=1}^{n} d_{ij} \cos(\psi_i - \psi_j)\ddot{\psi}_j + \sum_{j=1}^{n} d_{ij} \sin(\psi_i - \psi_j)(\dot{\psi}_j + \omega)^2 = \tau_i - \tau_{i+1} \tag{72}$$

Consider the linear (local) controllability around S_ω. That is, one assumes that $(\psi, \dot{\psi})$ are close to S_ω in (68). Linearization of (69) around any point in S_ω yields

$$\sum_{j=1}^{n} d_{ij} \ddot{\tilde{\psi}}_j + \sum_{j=1}^{n} d_{ij} (\tilde{\psi}_i - \tilde{\psi}_j)\omega^2 = \tau_i - \tau_{i+1} \tag{73}$$

Where $\tilde{\psi} = \left(\tilde{\psi}_1, \tilde{\psi}_2 \right)^T \in \mathbb{R}^2$ is the configuration of the linearized system. (73) Can be expressed in the following matrix form as

$$\begin{pmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{pmatrix}\begin{pmatrix} \ddot{\tilde{\psi}}_1 \\ \ddot{\tilde{\psi}}_2 \end{pmatrix} + \omega^2 \begin{pmatrix} d_{12} & -d_{12} \\ -d_{21} & d_{21} \end{pmatrix}\begin{pmatrix} \tilde{\psi}_1 \\ \tilde{\psi}_2 \end{pmatrix} = \begin{pmatrix} \tau_1 \\ 0 \end{pmatrix} \tag{74}$$

Let

$$M_0 = \begin{pmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{pmatrix}$$

$$U = \begin{pmatrix} d_{11} + d_{12} & 0 \\ 0 & d_{21} + d_{22} \end{pmatrix} \tag{75}$$

$$F = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Then using (75) in (74) yields

$$M_0 \ddot{\tilde{\psi}} + \omega^2 (U - M_0)\tilde{\psi} = F\tau_1 \tag{76}$$

Define

$$q = M_0 \tilde{\psi} \tag{77}$$

From (76), it yields

$$\ddot{q} = \omega^2 (I_n - \tilde{M}_0)q + F\tau_1 \tag{78}$$

Where

$$\tilde{M}_0 = UM_0^{-1} \tag{79}$$

And let

$$N = \omega^2 (I_n - \tilde{M}_0) \tag{80}$$

Then (78) becomes

$$\ddot{q} = Nq + F\tau_1 \tag{81}$$

It is mentioned in [4] that (81) is controllable if and only if

$$rank \begin{pmatrix} F & NF \end{pmatrix} = 2 \tag{82}$$

And the previous equation holds if and only if

$$rank \begin{pmatrix} F & \tilde{M}_0 F \end{pmatrix} = 2 \tag{83}$$

The following statements are recalled from [4]: Although condition (83) may be true in many cases, it is difficult to determine analytically whether (83) holds or not. However, in the case of the 2-link system with the first joint active and the second joint passive, (83) obviously holds.

5.2.2 Numerical check of controllability

Values in equations from (85) to (91) are the values obtained according to the previous section to determine the controllability of the system, where values in (84) are the parameters of the system.

$$\begin{aligned}
m_1 &= 0.462Kg \\
m_2 &= 0.384Kg \\
l_1 &= l_2 = 0.25m \\
r_1 &= r_2 = 0.125m \\
I_1 &= 0.0105Kg.m^2 \\
I_2 &= 0.0087Kg.m^2
\end{aligned} \tag{84}$$

$$M_0 = \begin{pmatrix} 0.0418 & 0.0120 \\ 0.0120 & 0.0147 \end{pmatrix} \tag{85}$$

$$M_0^{-1} = \begin{pmatrix} 31.23 & -25.47 \\ -25.47 & 88.72 \end{pmatrix} \tag{86}$$

$$U = \begin{pmatrix} 0.0538 & 0 \\ 0 & 0.0267 \end{pmatrix} \tag{87}$$

$$\tilde{M}_0 = \begin{pmatrix} 1.68 & -1.37 \\ -0.68 & 2.37 \end{pmatrix} \tag{88}$$

$$F = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \tag{89}$$

$$C_m = [F \quad \tilde{M}_0 F] = \begin{pmatrix} 1 & 1.68 \\ 0 & -0.68 \end{pmatrix} \tag{90}$$

51

$$rank\ (C_m) = 2 = n \qquad (91)$$

Since the rank of the previous controllability matrix equals 2 which equals the number of joints (n) , the system is controllable.

5.3   Joint angles control methods

5.3.1   PID approach

Proportional Integral Derivative controller (PID) is a classical control technique, which is a combination of PD and PI controllers, where the PD controller improves the transient response dynamics such as rise time, settling time and peak time, while PI controller improves steady-state error. A disadvantage of this controller is the gradual decrease in the speed of the response when the steady state error is improved. [20]

PID has two zeros and a pole at the origin. One zero and the pole at the origin can be designed as the ideal integral compensator, and the other zero can be designed as the ideal derivative compensator, the PID controller has the following transfer function

$$G_{PID} = K_p + \frac{K_I}{s} + K_D s \qquad (92)$$

Where

K_p is the proportional gain

K_I is the integral gain

K_D is the derivative gain

5.3.1.1   Regular PID controlling

This control approach aims to control the two joint angles simultaneously using PID controllers, since the system has no transfer function due to nonlinearity, The values of K_p , K_I and K_D are chosen based on trial and error. The block diagram of PID control of joint angles in Simulink is shown in Figure 5.2, since the underactuated manipulator has one actuator, then the input torque signal size must be [1x1] , this signal is the result of summing the two controllers torque signals.
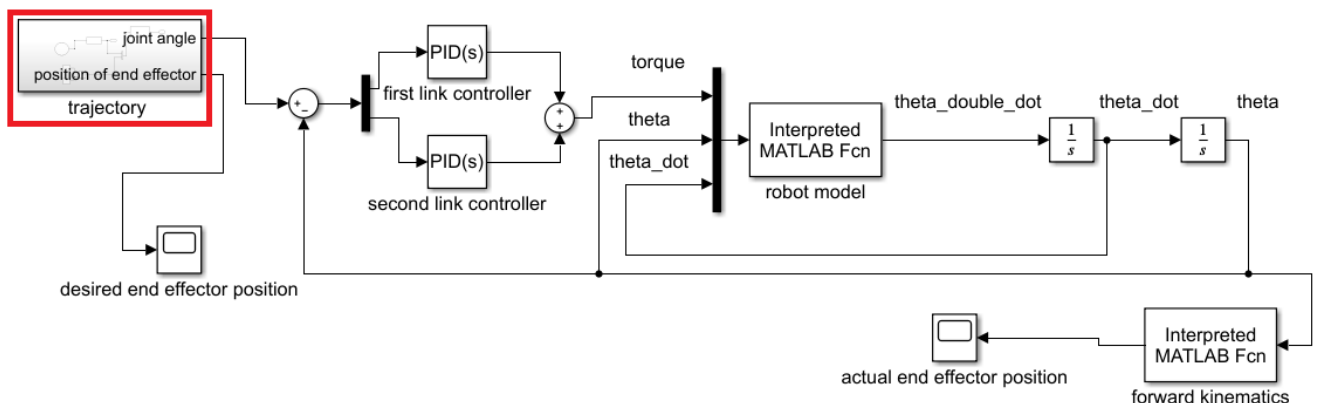
Figure 5.2: Control block diagram of PID controlled system

The components of 'trajectory' block is shown in Figure 5.3, while the MATLAB m.files are in Appendix D.
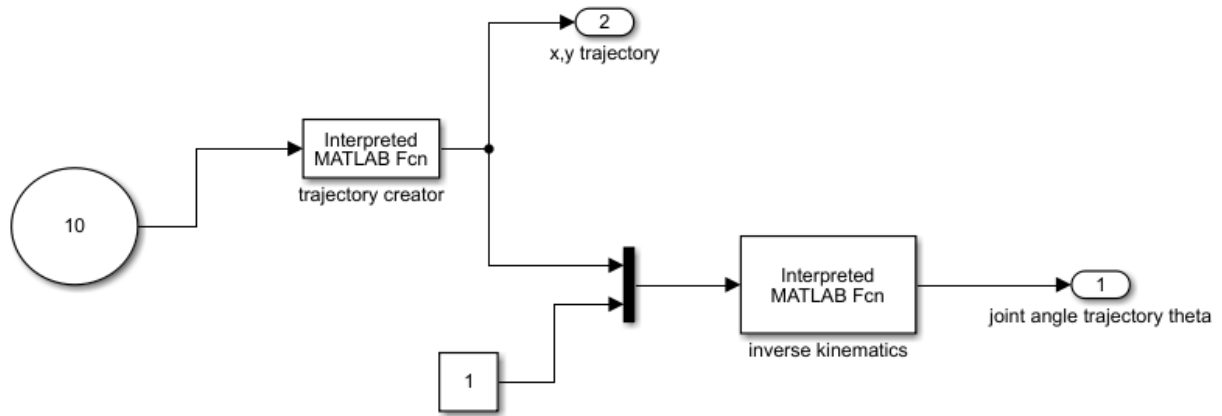
Figure 5.3: Components of trajectory creator block

The PID gains which chosen by trial and error are

$$K_{P1} = 20$$
$$K_{I1} = 0$$
$$K_{D1} = 20$$
$$K_{P2} = 0$$
$$K_{I2} = 0$$
$$K_{D2} = -30$$

(93)

The resulting joint angles compared with the desired angles using PID control are shown in Figure 5.4.
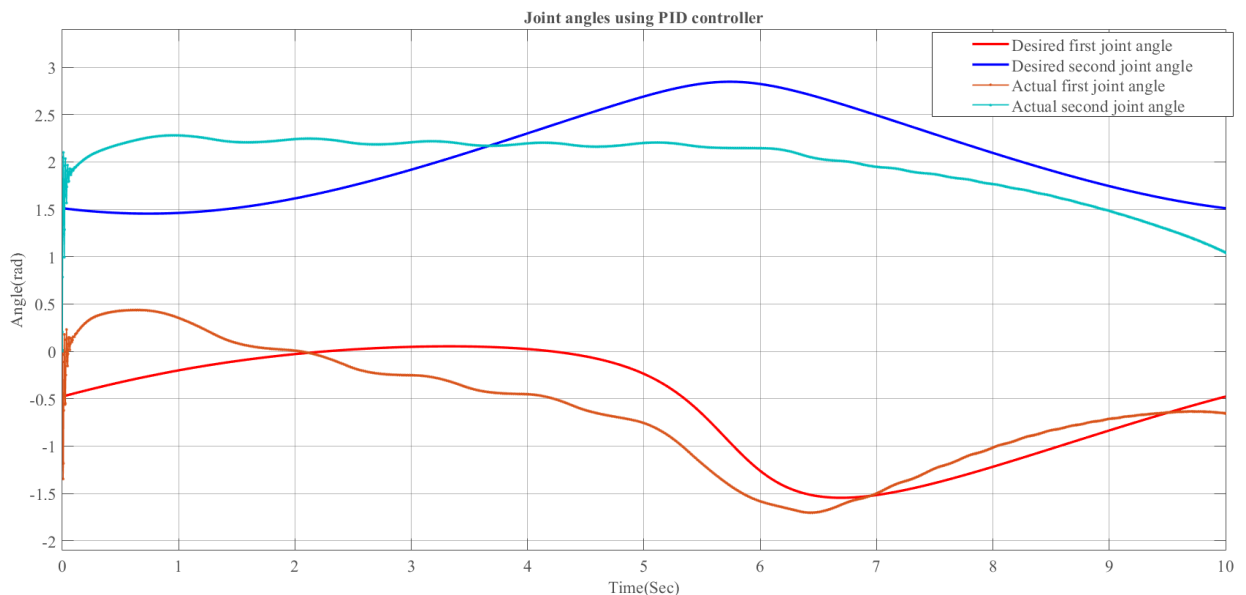
Figure 5.4: Resulting joint angles using PID control

The trajectory isn't completely applicable. The second angle reaches 2.25 rad due to large jump of the system in the beginning then it doesn't change because the rotating torque of the first link is less than the friction opposition of the second link.

53

The resulting end effector position of the resulting path by the manipulator compared with the desired position is shown in Figure 5.5.
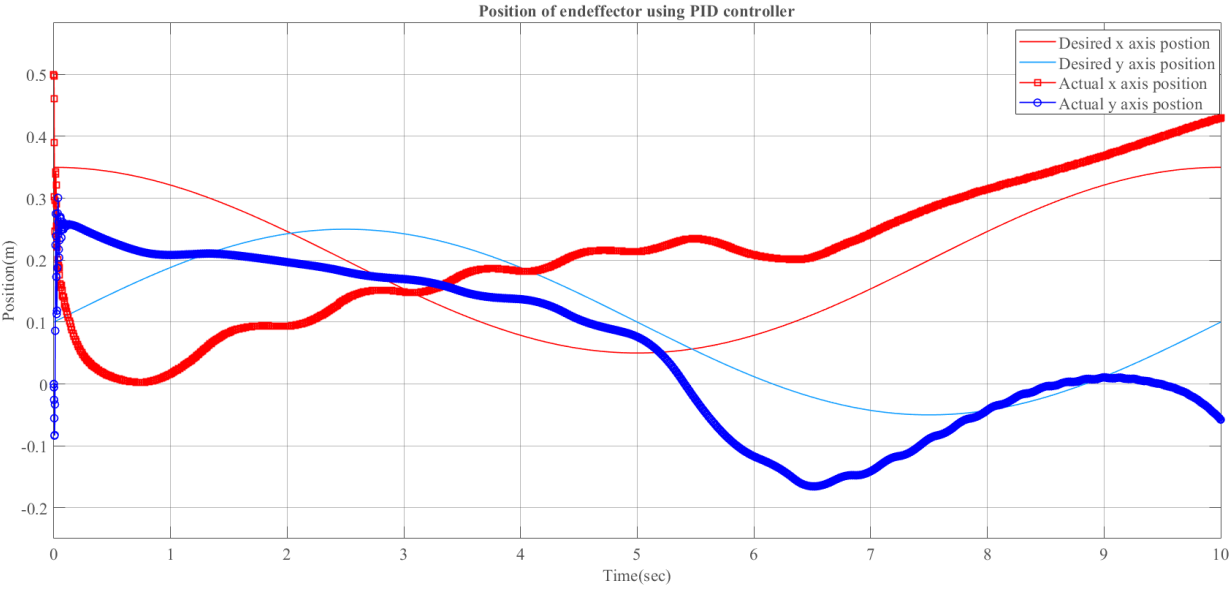
Figure 5.5: Resulting endeffector position using PID control

The resulting end effector position graph using xy graph compared with the desired position is shown in Figure 5.6.
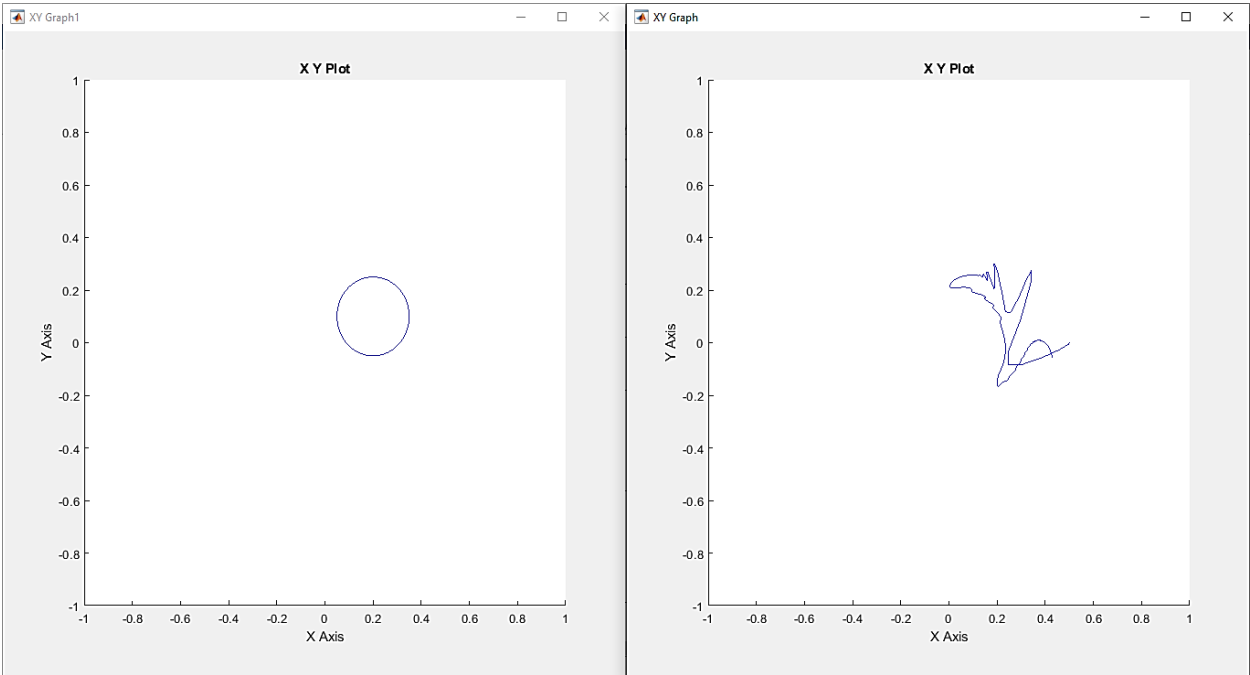
Figure 5.6: Resulting graph by endeffector using PID control

Figure 5.7 shows the control effect of each controller and the input torque to the model, which is the summation of the controllers contributions. The high jump in torque value is due to the high acceleration to match the desired trajectory at the beginning, and it can be neglected or solved by

adding homing function to the model. Except the beginning, the input torque is zero along the simulation time.
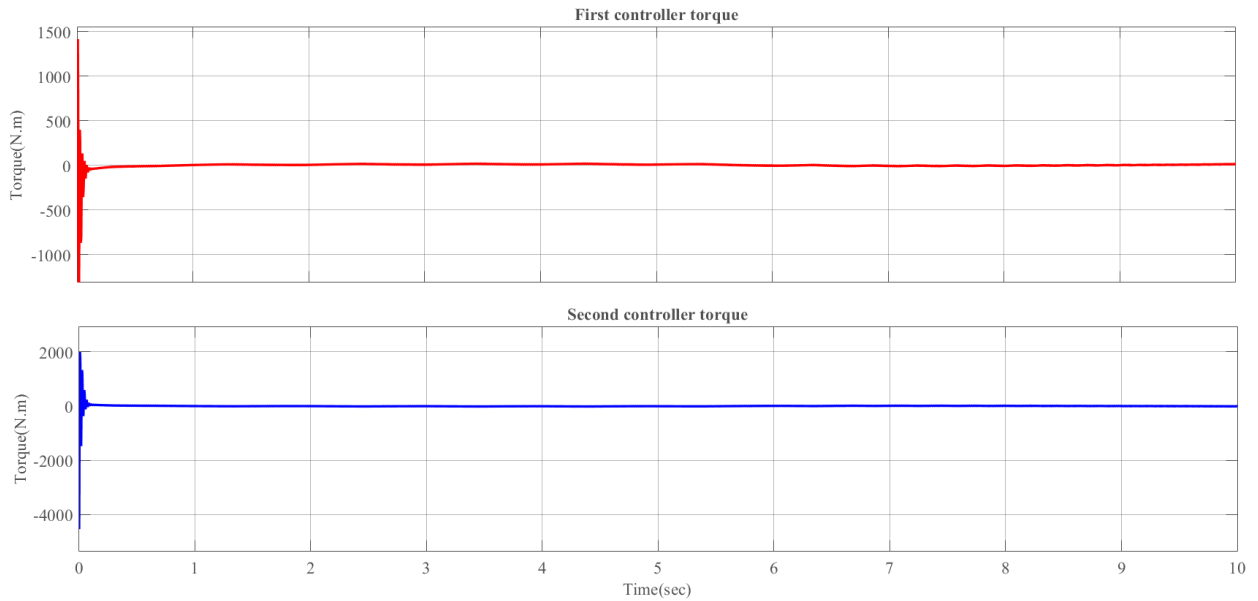
Figure 5.7: Required torque for the system using PID control

This strategy failed to achieve the desired response, in addition, it has two problems, the high error of system response and zero input torque, to attempt to reduce these problems, an error switching function is suggested.

5.3.1.2   PID controller with switching controllers

The overall response of the system is compensated and the problem of equal and opposite link torques and zero input torque is solved by using a programmed switch, The block diagram of PID control with switching is shown in Figure 5.8,while the switcher function code is shown in appendix D.
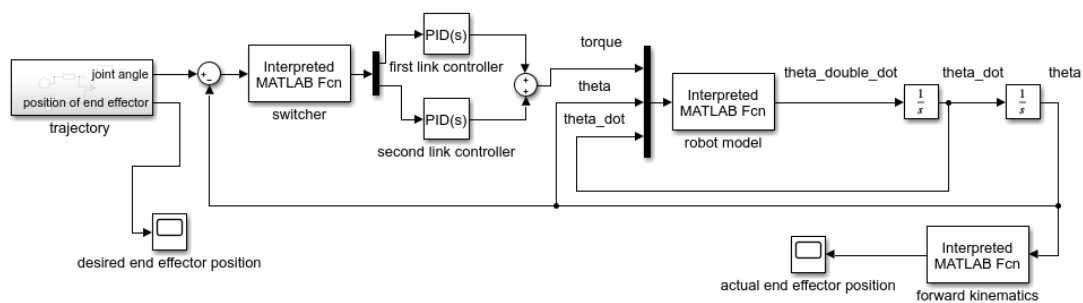
Figure 5.8: Control block diagram of PID with switching controllers system

The programmed switch is based on eliminating the control effect of a controller among a specific range of angle error.

The resulting joint angles compared with the desired angles are shown in Figure 5.9.The contribution of the switch is concluded from the better response of the system in joint plane.

55
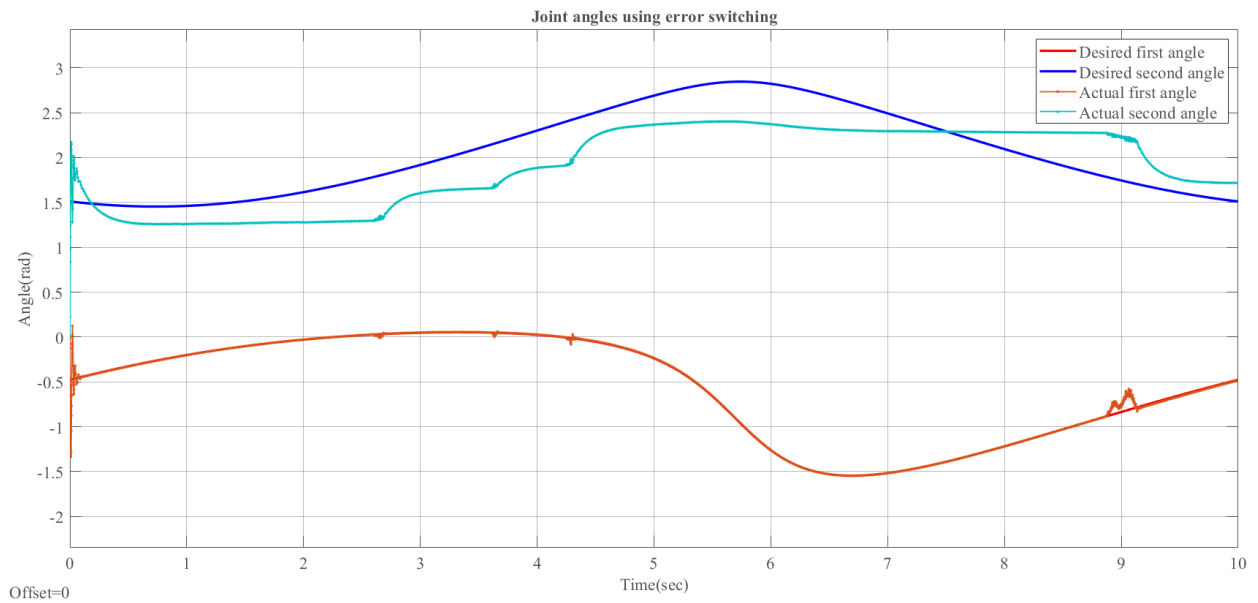
Figure 5.9: Resulting joint angles using PID with switching controllers

The resulting end effector position of the resulting path by the manipulator compared with the desired position is shown in Figure 5.10.
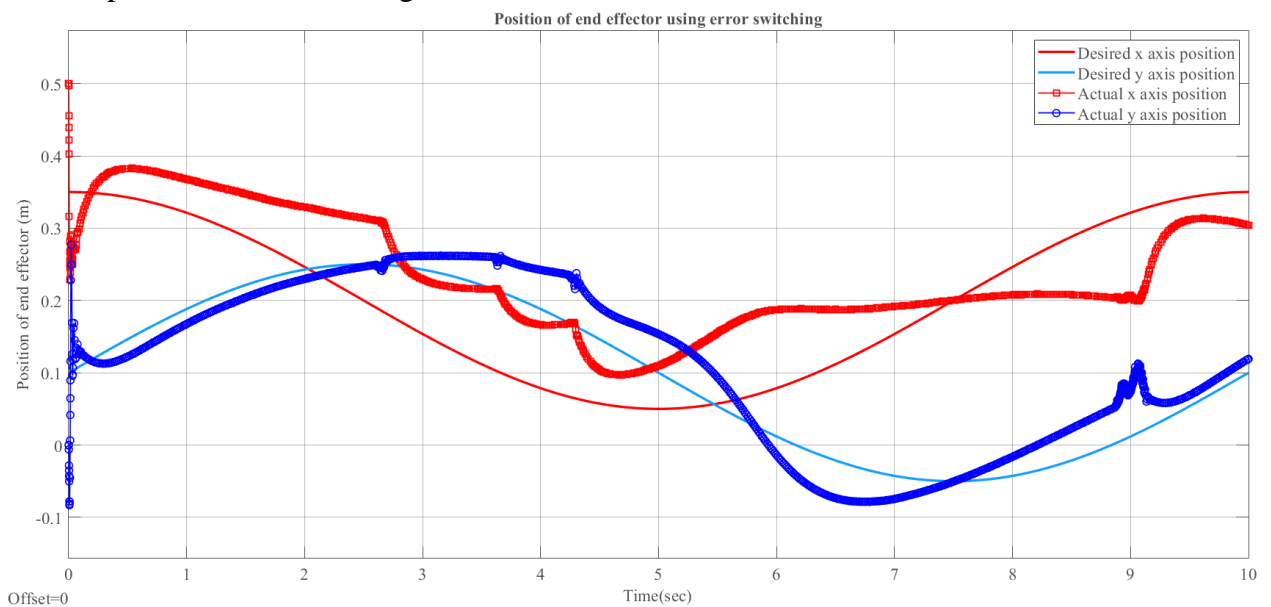
Figure 5.10: Resulting endeffector position using PID with switching controllers

The resulting end effector position graph using xy graph compared with the desired position is shown in Figure 5.11.
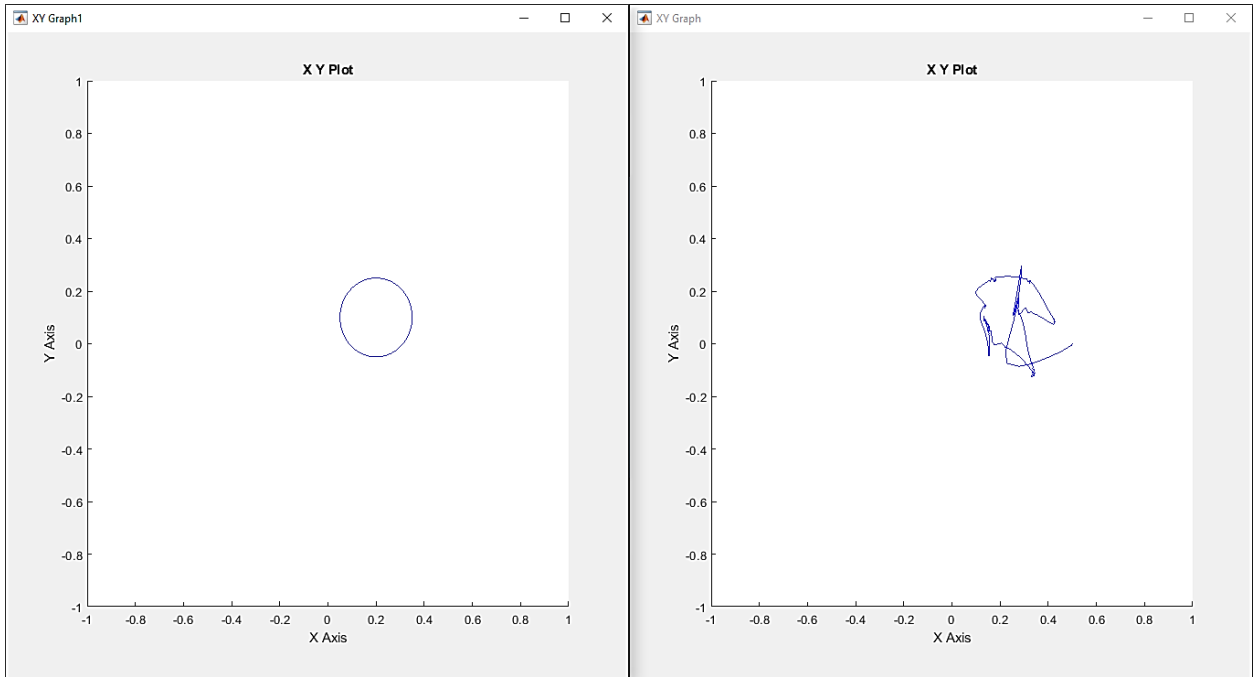
Figure 5.11: Resulting graph by endeffector using PID with switching controllers

Figure 5.12 and Figure 5.13 show the control effect of each controller and the input torque to the model, which is the summation of the controllers contributions. Although the problem of zero torque is solved, it's obvious that the required torque to drive the system to the desired trajectory is much higher. The required torque is about 1300 N.m in both directions, this high torque is referred to the loss of actuation of second link and the continuous change of the second angle.
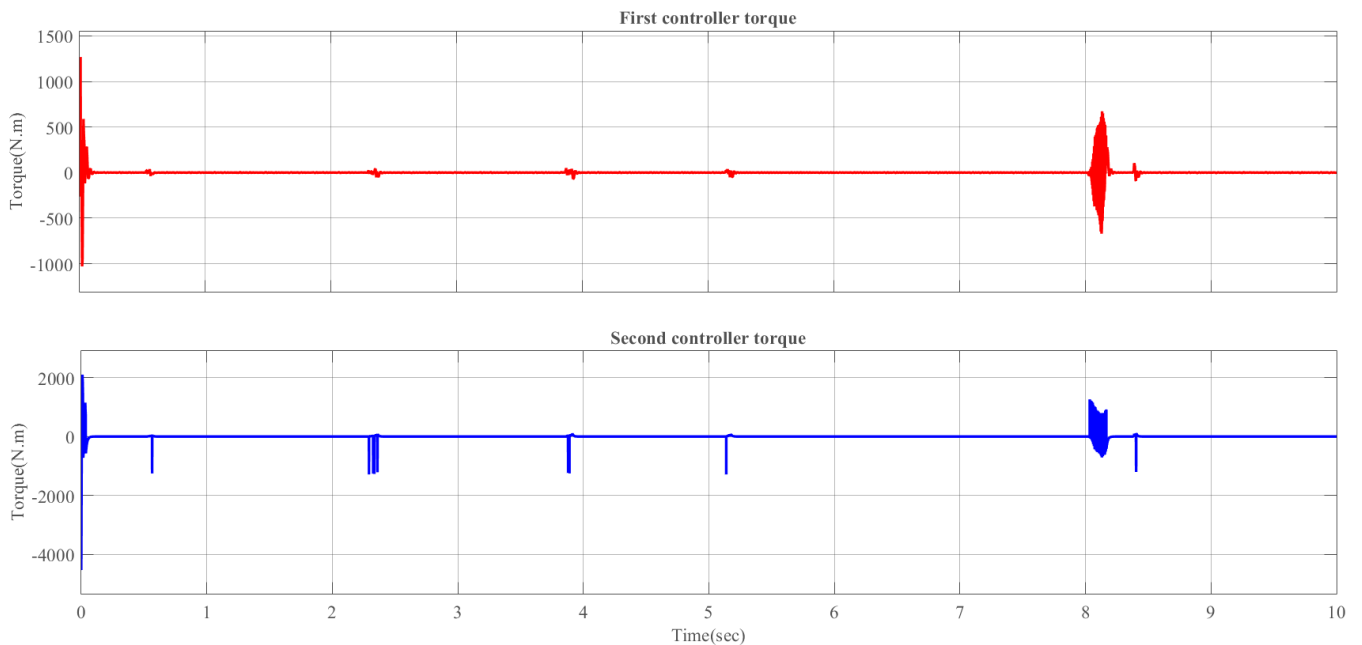
Figure 5.12: The required torque of each link controller using PID and switching

controllers
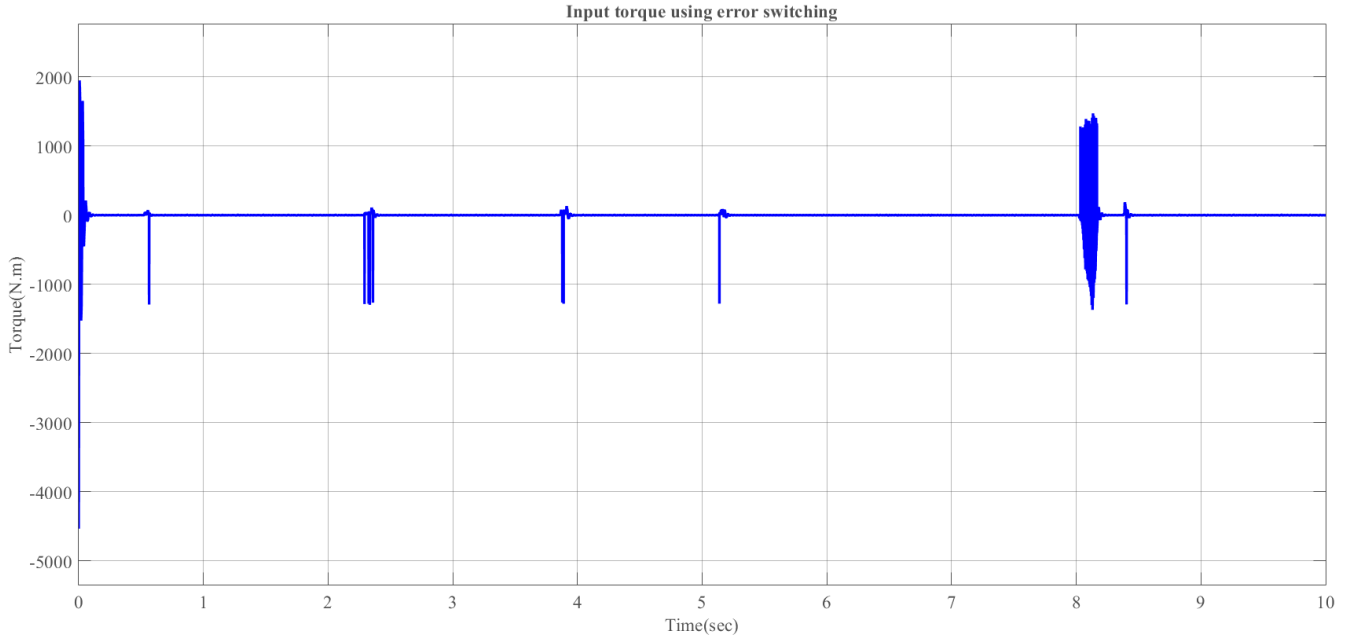
Figure 5.13: The input torque to the model using PID with switching controllers

Despite the results of this strategy after are better than using PID alone due to the contribution of the switch, this strategy can't be practically applied due to the need of high actuating torque.

5.3.2   LQR approach

LQR controller is a linear controller that provides optimally controlled feedback gains to enable the closed-loop poles stable and high performance design of systems.

The purpose of optimal control system design is to achieve a desired performance or specifications of each individual state to reach its final value with minimum input energy or control effort (minimum actuators efforts).

The performance index (J_p) for regulator problem can be described as the following

$$J_p = \int_0^\infty (\vec{x}^T Q \vec{x} + \vec{x}^T K^T R K \vec{x})dt$$
$$= \int_0^\infty \vec{x}^T (Q + K^T R K)\vec{x}dt \tag{94}$$

Where $\vec{x}$ is the state vector of the system in joint space which consists of angular positions and speeds of the joints, Q is called performance weighting matrix (state weighting matrix), R is called effort weighting matrix (inputs weighting matrix) and K is the feedback gain matrix.

The selection of matrices Q and R is a trial and error process, but a practicable method as a first choice for these matrices can be derived using Bryson's rule which states that a simple and reasonable choice for this matrices is to be diagonal as the following [18]

$$Q = diag\left(\frac{1}{max.\ acceptable\ output\ error^2}\right) \tag{95}$$

$$R = diag \left(\frac{1}{max . \ acceptable \ input \ error^2} \right) \qquad (96)$$

The block diagram of the system with LQR controller is shown in Figure 5.15 and a full detailed block diagram is shown in Figure 5.15
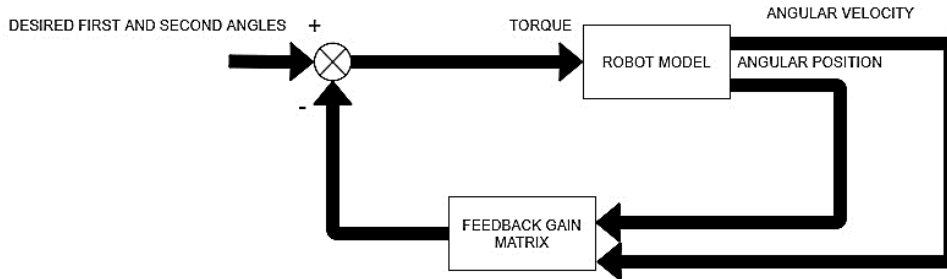
Figure 5.14: Control block diagram of LQR controlled system
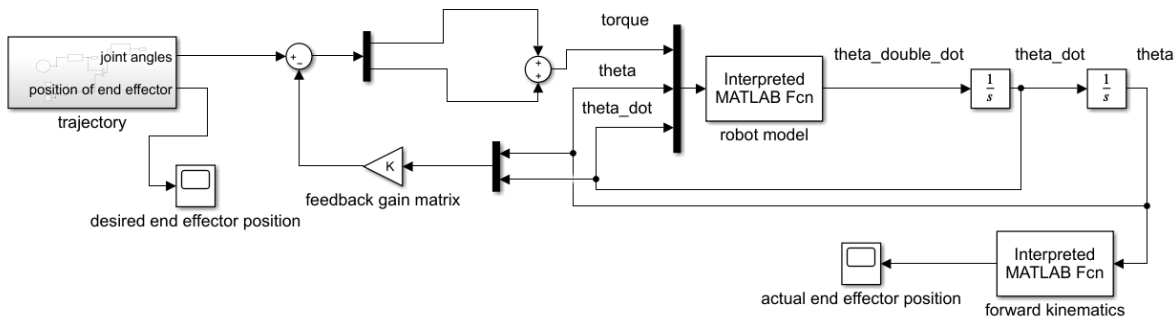
Figure 5.15: Control block diagram of LQR controlled system in simulink

The resulting joint angles compared with the desired angles using LQR control are shown in Figure 5.16.
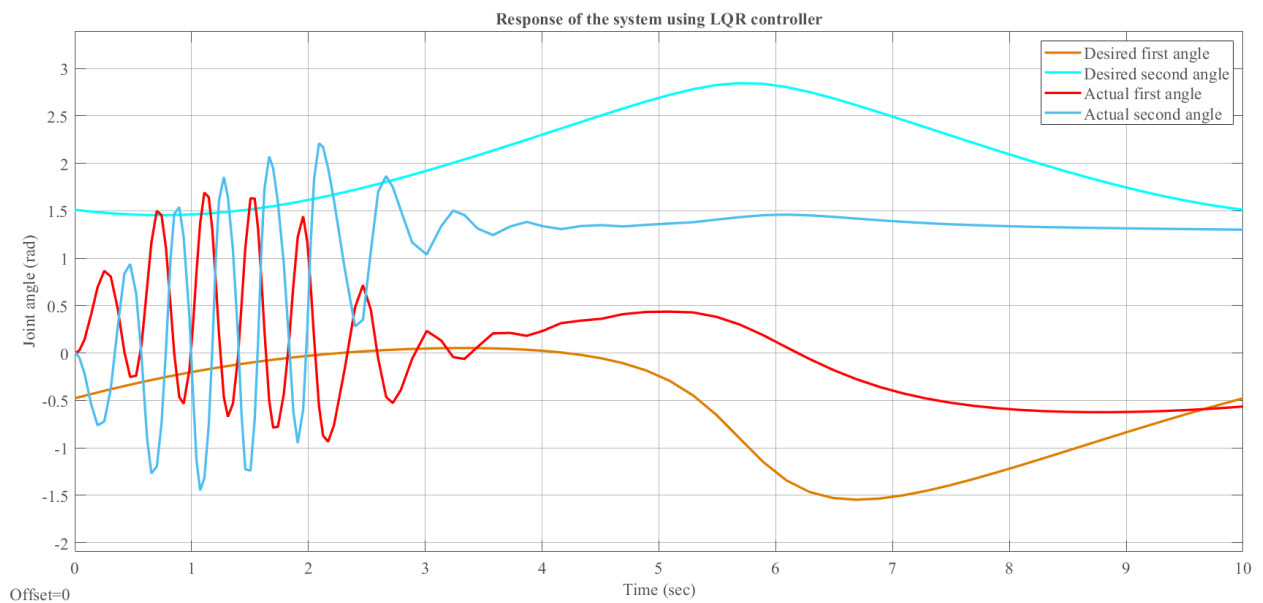
Figure 5.16: Resulting joint angles using LQR controller

Figure 5.17 shows the torque of the two links and the input torque to the robot model.
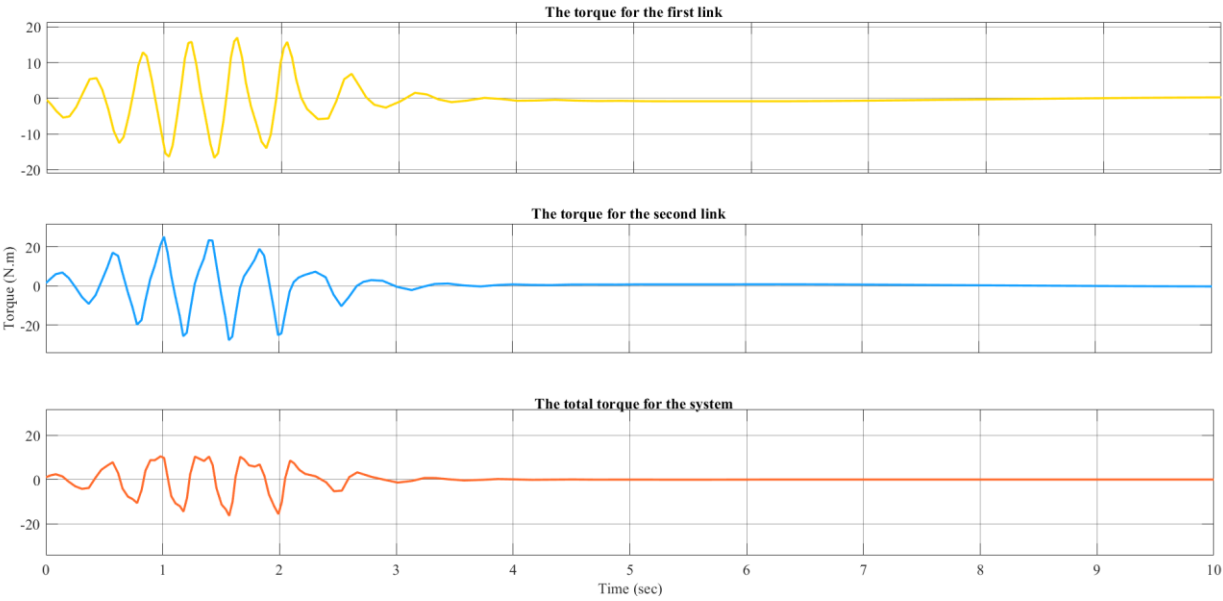
Figure 5.17: The required torque for each link and the input torque using LQR controller

The resulting end effector position graph using xy graph compared with the desired position is shown in Figure 5.18.
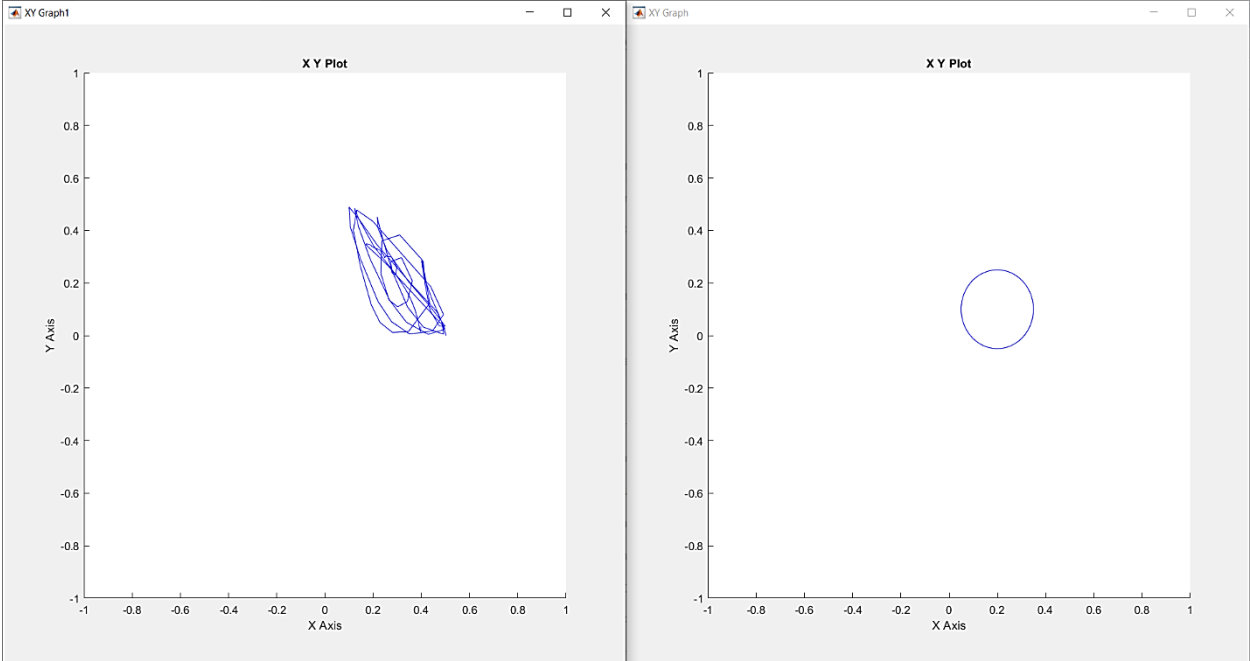
Figure 5.18: Resulting graph by endeffector using LQR controller

The LQR controller strategy failed to achieve the desired joint angles, an additional problem is the input torque which is zero.

5.3.3 Two stages approach

The methodology reviewed in [6] is used to control the joint angles, it consists of two stages, first stage is achieve the desired second joint then achieve the desired first joint by low torque values less than the friction force between the two links.

The robot is controlled via first Joint, the measurements are the position of the two joints using two independent encoders. The control environment is Matlab software, specifically Simulink environment for simulation, and for experimental part Real-time toolbox in Simulink is used, where the sampling time is *0.001 seconds* for both parts, the solver is *Ode45(Dormand-Prince)* in simulation and *Ode4(Runge-Kutta)* in experimental part, and the initial conditions are considered to be zero.

The results are recorded using Matlab S-function which shown later in this section.

The control block diagram is shown in Figure 5.19 and a full detailed block diagram is shown in Figure 5.20 and Figure 5.30 and the MATLAB m files of the functions are in Appendix D.
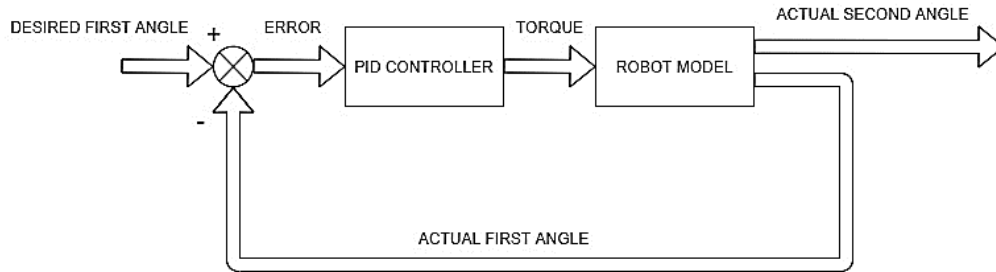
Figure 5.19: Control block diagram of first link control

5.3.3.1 Simulation results

A full detailed block diagram is shown in Figure 5.20.
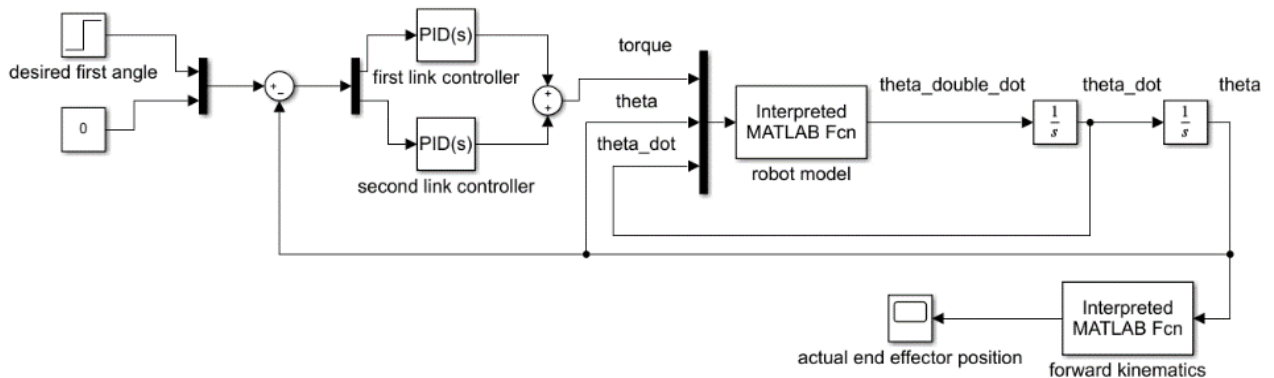
Figure 5.20: Control block diagram of first link control in simulink

Where the used controller is PID controller with the following transfer function

$$G_{PID} = K_p + \frac{K_I}{s} + K_D s \qquad (97)$$

Where

61

K_p is the proportional gain

K_I is the integral gain

K_D is the derivative gain

Since the PID control in joint space is for first joint only, the second link controller is off and its gains equal zero. The PID controller gains are chosen based on trial and error, which are

$$
\begin{aligned}
K_{p1} &= 15 \\
K_{I1} &= 0 \\
K_{D1} &= 4 \\
K_{p2} &= 0 \\
K_{I2} &= 0 \\
K_{D2} &= 0
\end{aligned} \tag{98}
$$

The result of the first angle using PID controller with parameters in (98) for a step input of 40 degrees is shown in Figure 5.21
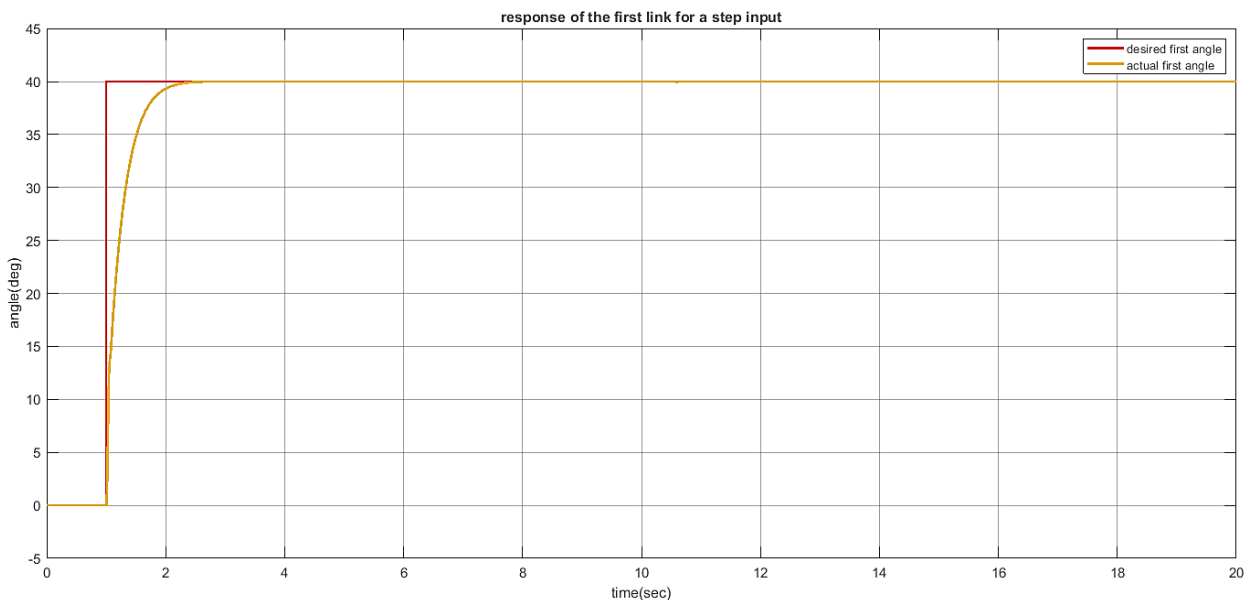
Figure 5.21: Response of the first link for a step input in simulation

The specifications of the response are

$$
\begin{aligned}
\%OS &= 0\% \\
e_{ss} &= 0\% \\
T_s &= 0.91\text{sec}
\end{aligned} \tag{99}
$$

5.3.3.1.1. Reach desired joint angles

In order to attempt to find a relationship between the first angle and second angle, second link angle is recorded while giving the system shown in Figure 5.20 a number of desired first angles, the results are shown in Table 5.1

Table 5.1: Results of second link angle for a group of first link angles in simulation

First angle(rad)	Second angle(rad)
1	0
1.5	0.17
2	0.5
2.5	1
3	1.7
3.5	2.5
4	3.3
4.5	4
5	4.7
5.5	5.25
6	5.72

The values of second link angle against first link angle is plotted using EXCEL software where the plot is shown in Figure 5.22.
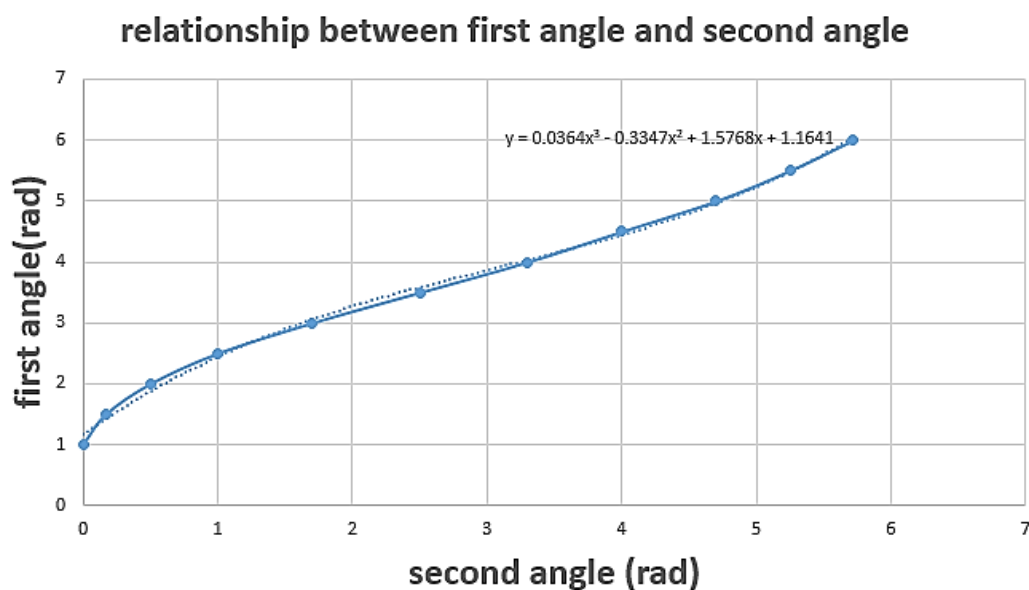
Figure 5.22: Relationship between the first angle and second angle in simulation

The equation of the curve is determined using EXCEL, which is

$$\theta_1 = 0.0364\theta_2^3 - 0.3347\theta_2^2 + 1.5768\theta_2 + 1.1641 \qquad (100)$$

Using (100), a desired second angle can be achieved by means of first angle, the block diagram in Simulink environment is shown in Figure 5.23 and the MATLAB m files of the functions are in Appendix D.
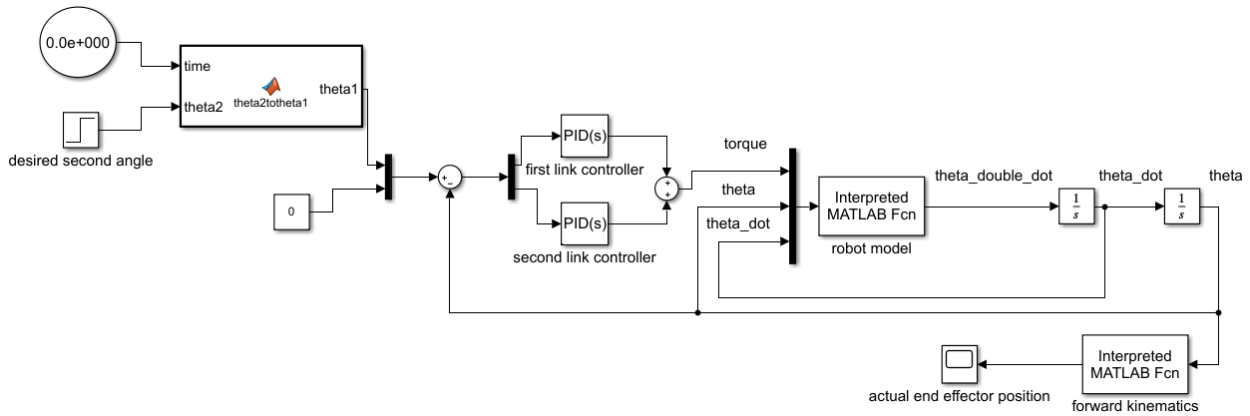
Figure 5.23: Control block diagram of second link control in simulink

Following figures are the resulting responses of the system for some random second link angles.

Figure 5.24 shows the response of the system for $\theta_2 = 60$ degrees.
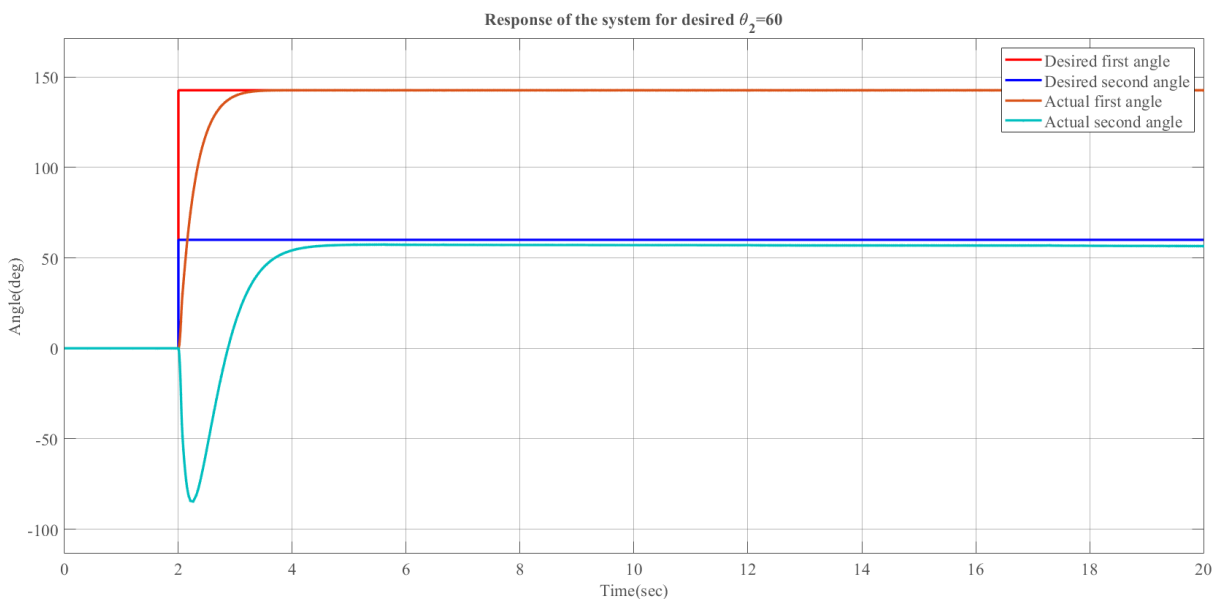
Figure 5.24: The response of the system at desired θ2=60 deg in simulation

Figure 5.25 shows the response of the system for $\theta_2 = 120$ degrees.
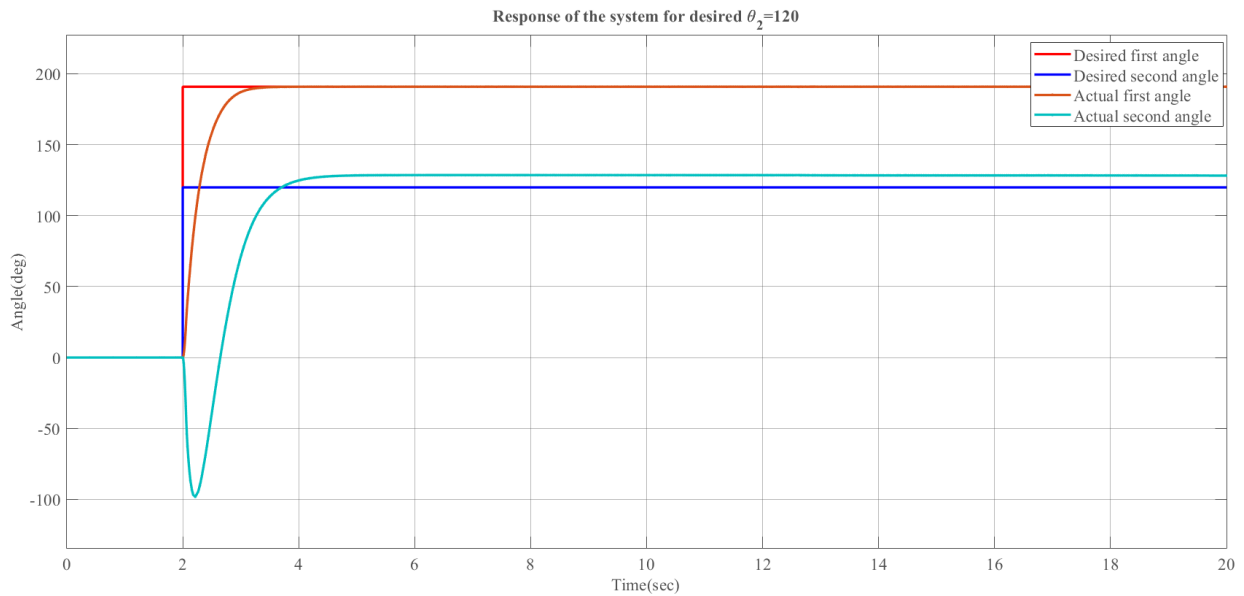
Figure 5.25: The response of the system at desired θ2=120 deg in simulation

5.3.3.1.2. Actuator(origin) centered circles

The methodology of the previous section is extended to follow a trajectory that leads to draw a circle where its center is the actuator by driving the system at low rotational speeds to prevent torques higher than the friction force between the two links, this condition is essential to maintain the angle of the second link.

To determine the required angles by means of a required radius, cosine theorem is used by the geometry of the manipulator .The geometry of the system is shown in Figure 5.26.
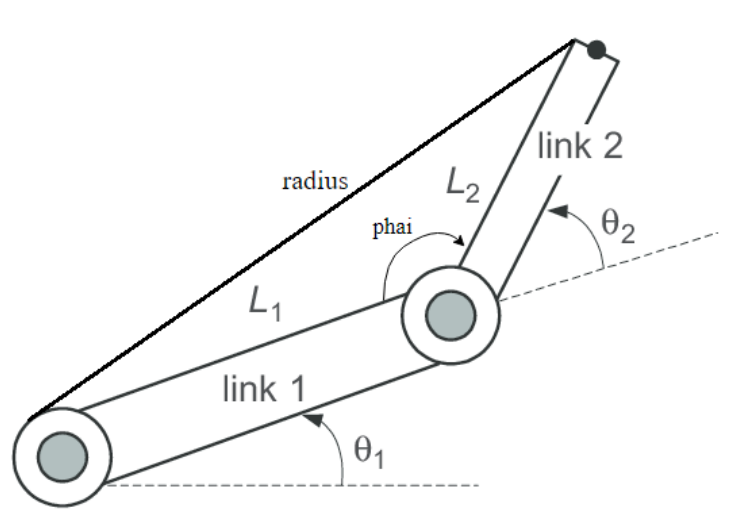
Figure 5.26: Geometry of the system

And the equations of the triangle made by the two links and the radius are

$$\cos\phi = \frac{L_1^{\,2} + L_2^{\,2} - radius^2}{2L_1L_2}$$

$$\phi = \cos^{-1}(\frac{L_1^{\,2} + L_2^{\,2} - radius^2}{2L_1L_2})$$ (101)

$$\phi = 180 - \theta_2$$

$$\theta_2 = 180 - \phi$$ (102)

The block diagram of this method is shown in Figure 5.27 and a cosine theorem-based Matlab function is added where the MATLAB m files of the functions are in Appendix D.
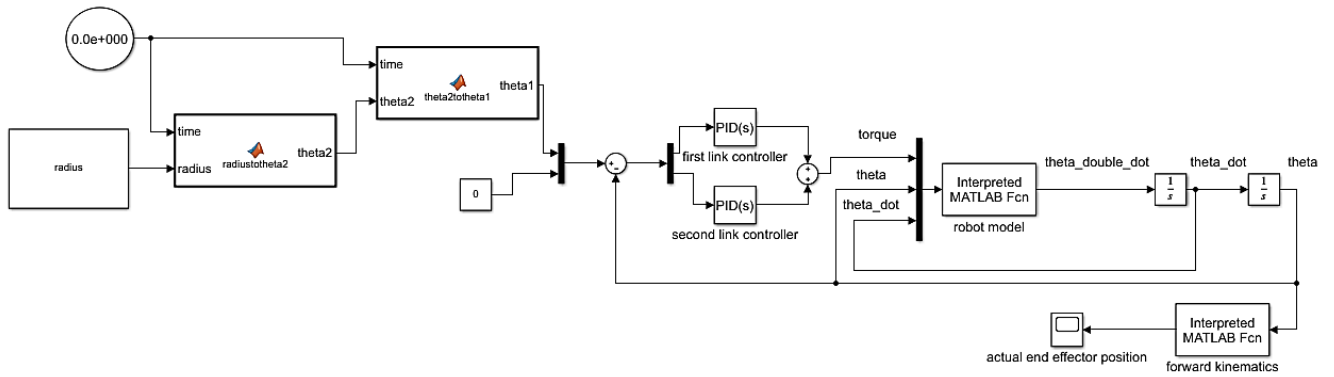
Figure 5.27: Circular trajectory control block diagram in Simulink

Figure 5.28 show the response of the system for radius=20cm.
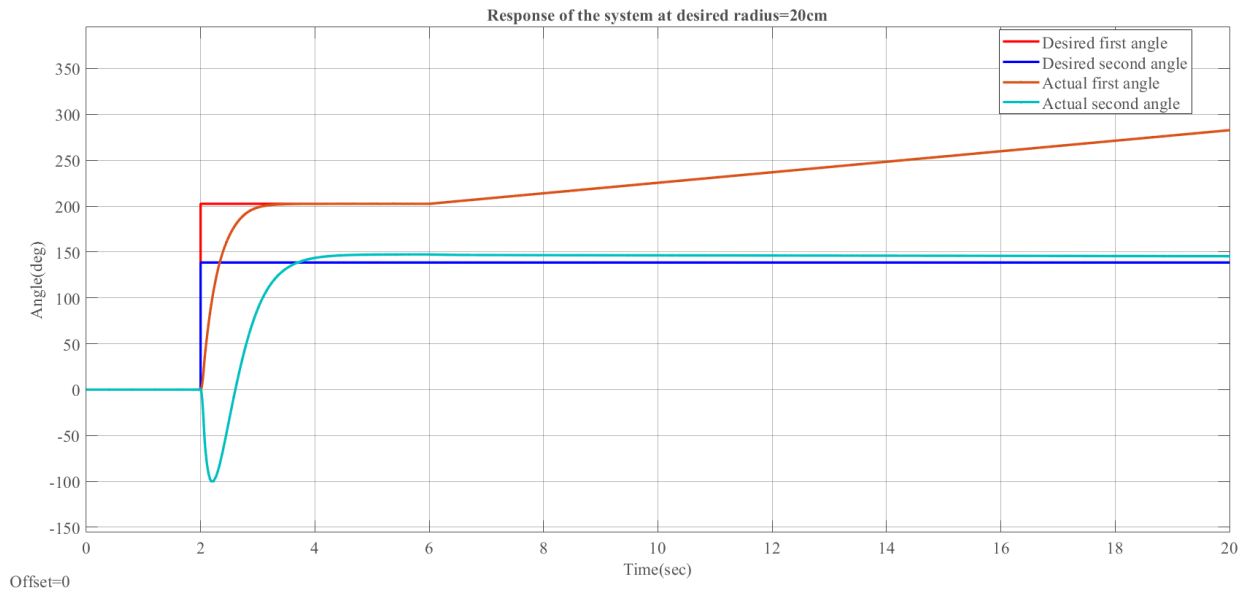
Figure 5.28: Response of the system for radius=20cm in simulation

Figure 5.29 show the response of the system for radius=30cm.
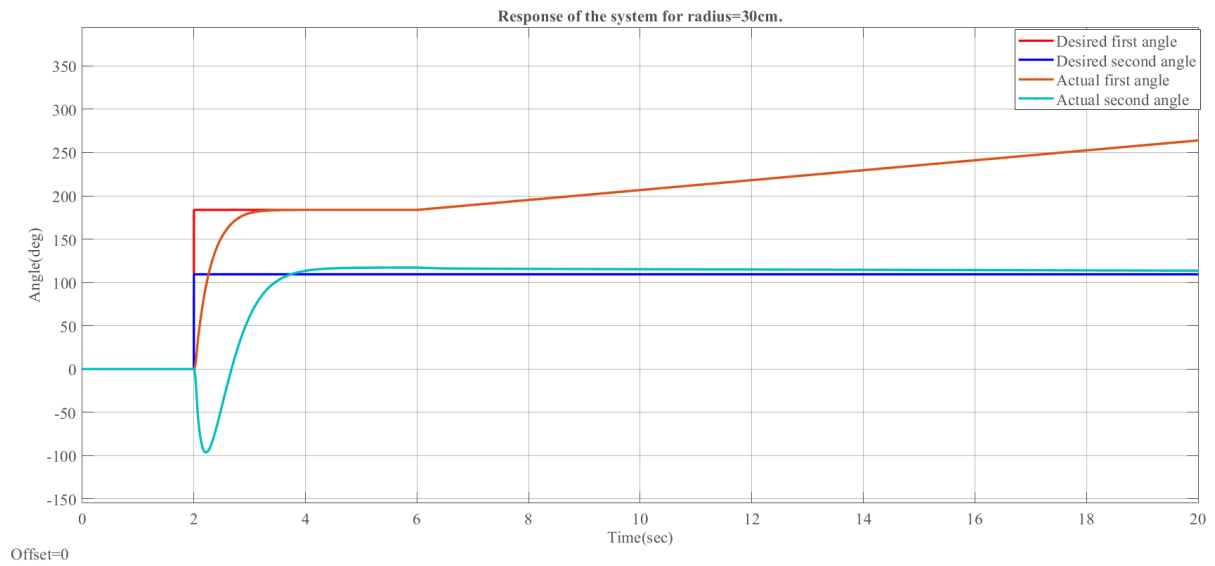
Figure 5.29: Response of the system for radius=30cm in simulation

5.3.3.2   Experimental results

A full detailed block diagram for control of first link is shown in Figure 5.30
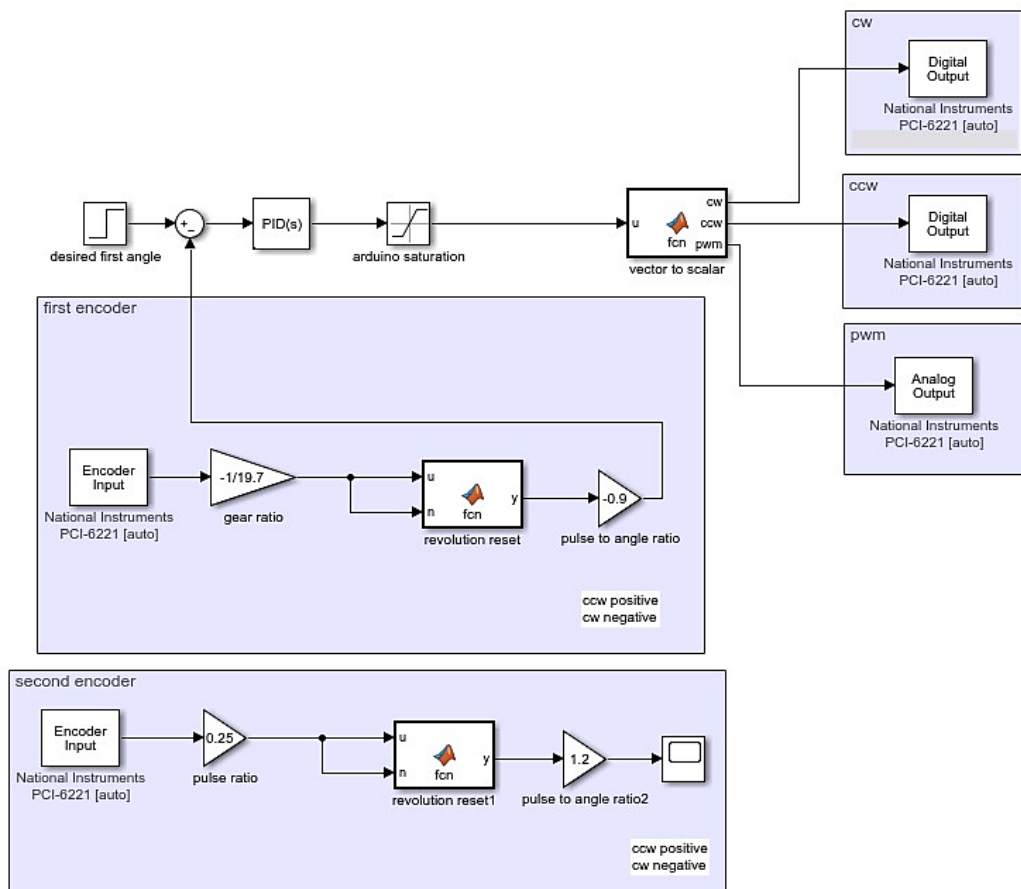
Figure 5.30: Control block diagram of first link control in real time simulink

Since an Arduino microcontroller is used as mentioned in Chapter 3, saturation block is used to prevent the damage of the Arduino microcontroller, the saturation limits are from -5 to 5 which are the limits of the input voltage to Arduino.

The PID controller gains are chosen based on trial and error, which are

$$K_p = 5$$
$$K_I = 0 \tag{103}$$
$$K_D = 0$$

The experimental result of the first angle using PID controller with parameters in (103) for a step input of 40 degrees is shown in Figure 5.31.
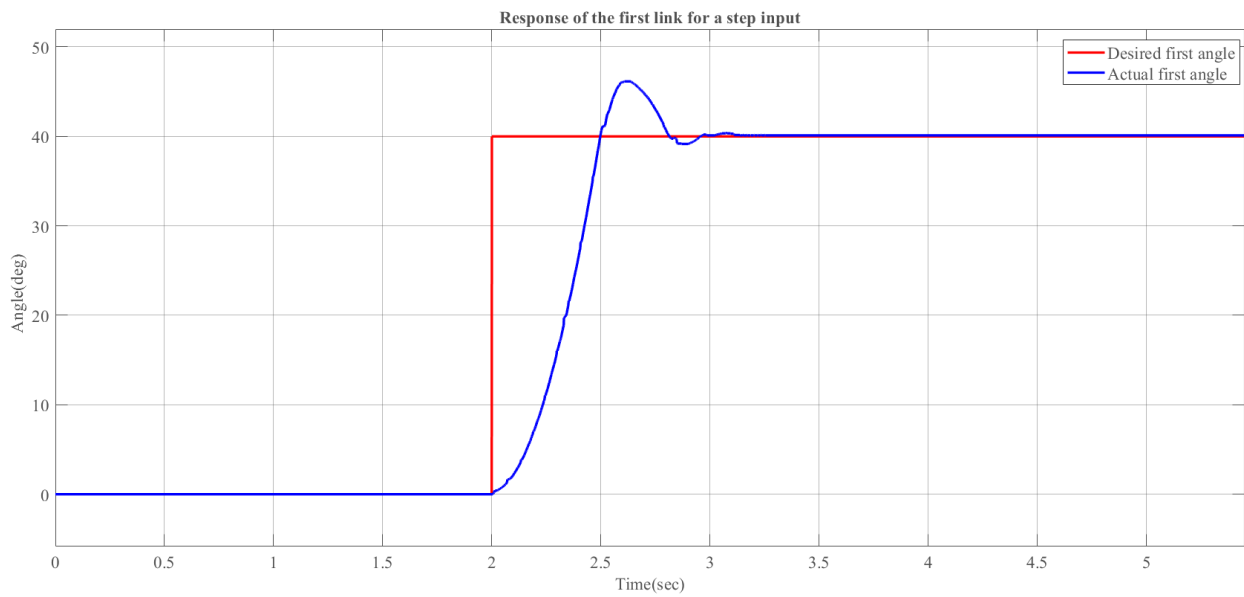
Figure 5.31: Response of the first link for a step input in experimental part

The specifications of the response are

$$\%OS = 15.3\%$$
$$e_{ss} = 0.27\%$$
$$T_s = 0.91 \text{sec} \tag{104}$$
$$T_p = 0.61 \text{sec}$$

5.3.3.2.1.　Reach desired joint angles

In order to attempt to find a relationship between the first angle and second angle, a sample of second link angles are recorded while giving the system shown in Figure 5.30 a number of desired first angles, the results are shown in Table 5.2 .

Table 5.2: Results of second link angle for a group of first link angles in experimental

First link angle (deg)	Second link angle #1	Second link angle #2	Second link angle #3	Second link angle #4	Second link angle #5	Mean value (deg)
10	29.4	31.5	27.6	29.7	30.9	29.82
15	44.4	45	47.1	43.8	44.1	44.9
20	55.2	57.3	53.4	49.8	55.5	54.24
30	84	83.4	79.2	83.7	85.8	83.22
45	131.7	126.3	133.5	133.2	126.3	130.2
60	160.2	161.1	161.7	163.8	160.8	161.52
75	173.4	167.4	164.7	173.4	171.6	170.1
90	176.4	175.2	175.2	175.2	177.3	175.86

The mean value of second link angle against first link angle is plotted using EXCEL software where the plot is shown in Figure 5.32.
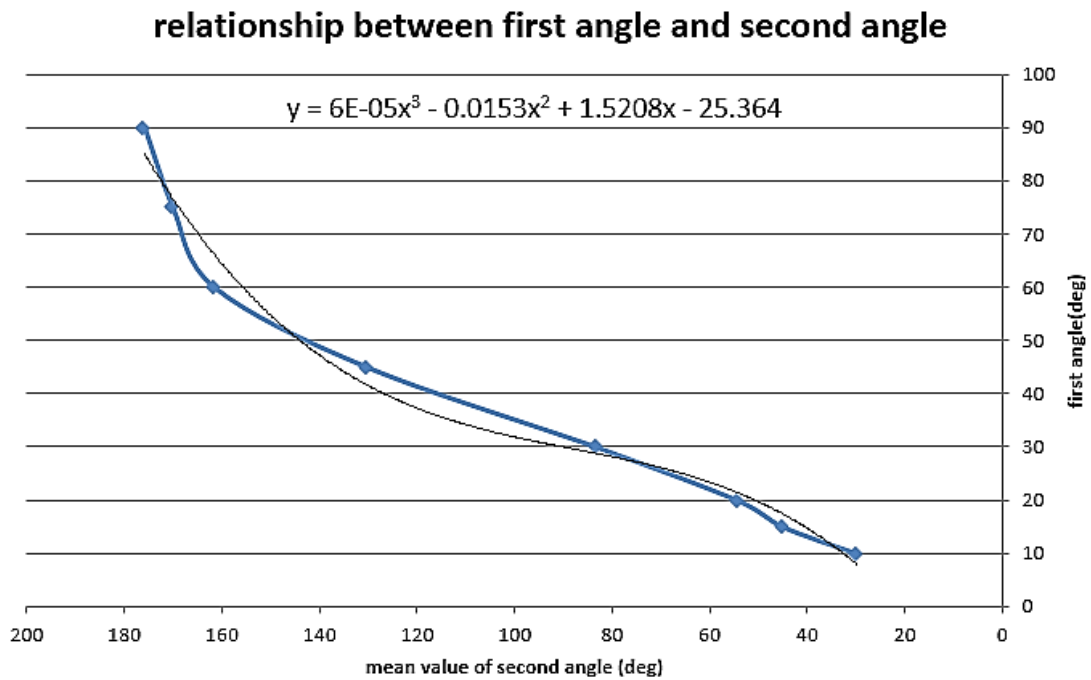
Figure 5.32: Relationship between the first angle and second angle

The equation of the curve is determined using EXCEL, which is

$$\theta_1 = 6*10^{-5}\theta_2^{\,3} - 0.0153\theta_2^{\,2} + 1.5208\theta_2 + 25.364 \qquad (105)$$

Using (105), a desired second angle can be achieved by means of first angle, the block diagram in Simulink environment is shown in Figure 5.33 and the MATLAB m files of the functions are in Appendix D.
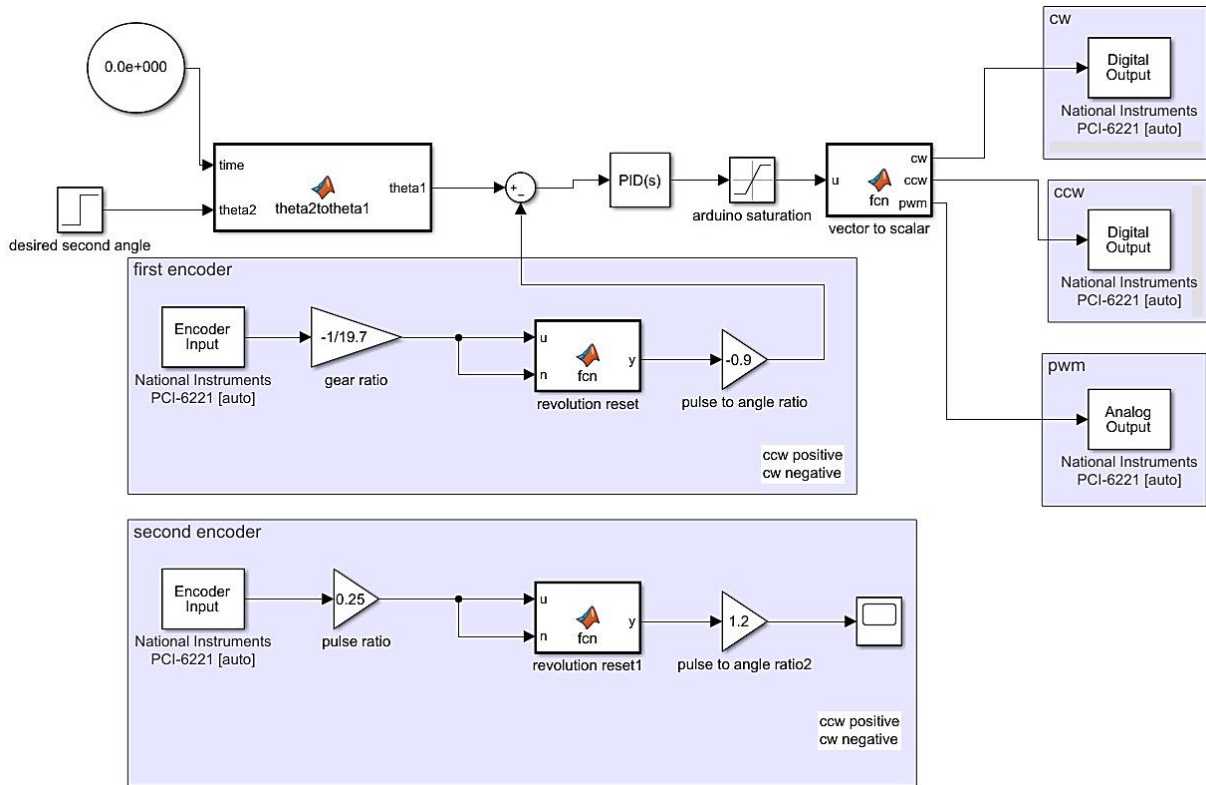
Figure 5.33: Control block diagram of second link control in real time simulink

Following figures are the resulting responses of the system for some random second link angles.

Figure 5.34 shows the response of the system for $\theta_2 = 50$ degrees.
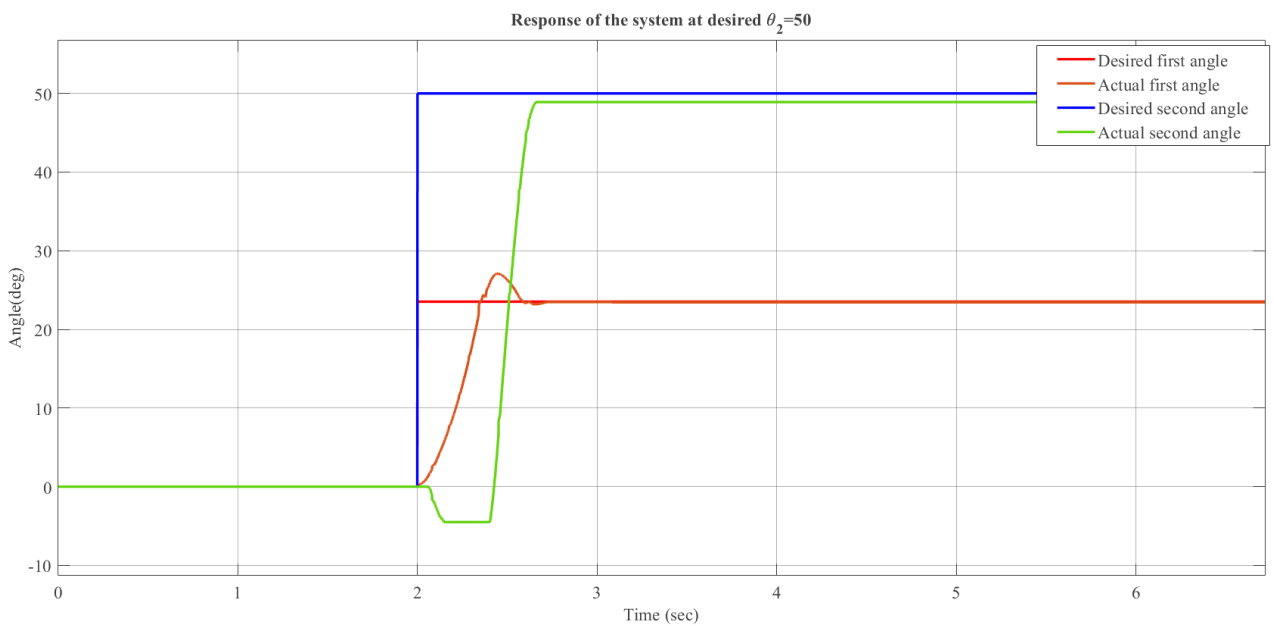
Figure 5.34: The response of the system at desired θ2=50 deg in experimental

The response has $e_{ss} = 2.2\%$ for the second angle.

Figure 5.35 shows the response of the system for $\theta_2 = 90$ degrees.
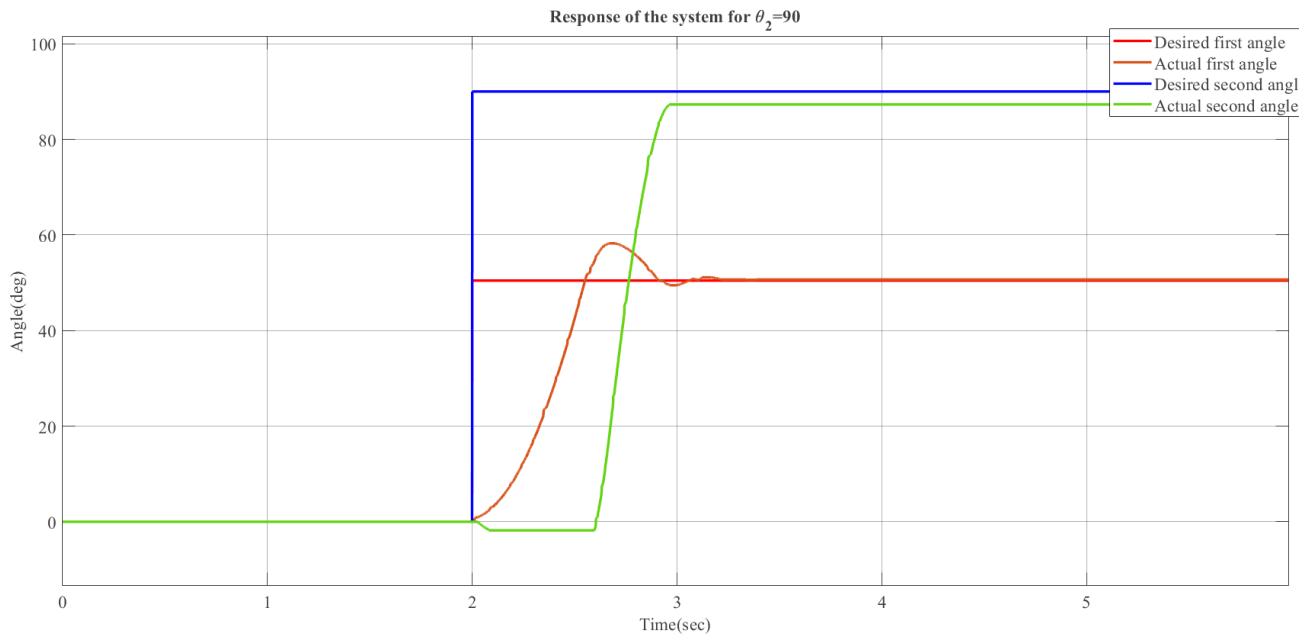
Figure 5.35: The response of the system at desired θ2=90 deg in experimental

The response has $e_{ss} = 3\%$ for the second angle.

Figure 5.36 shows the response of the system for $\theta_2 = 130$ degrees.
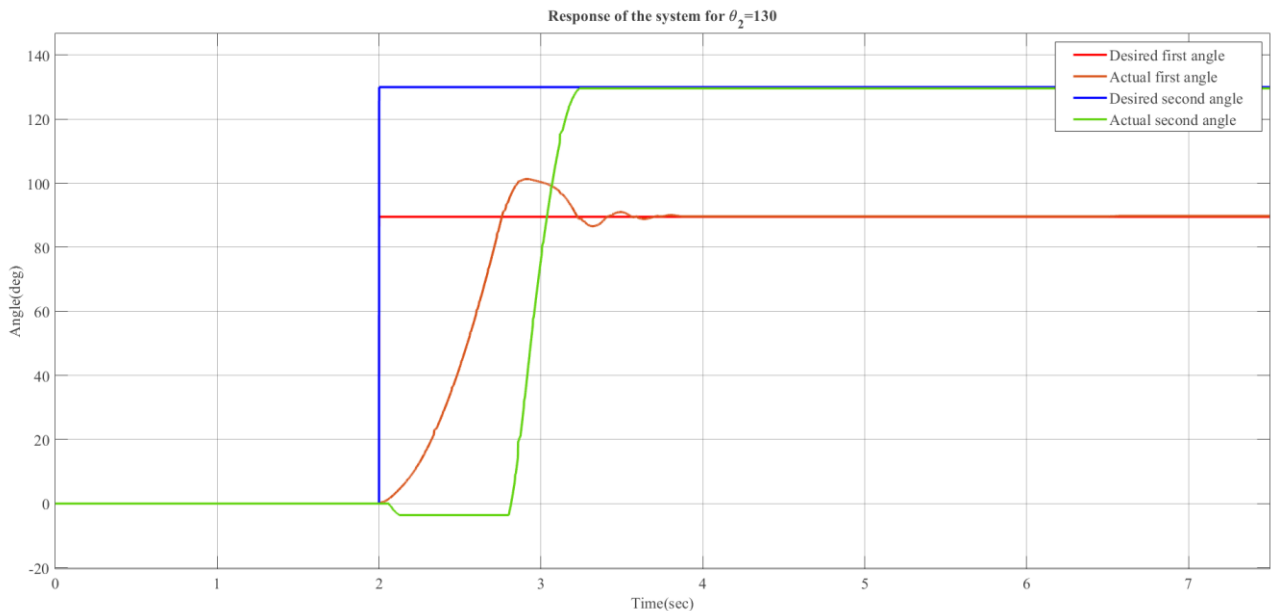
Figure 5.36: The response of the system at desired θ2=130 deg in experimental

The response has $e_{ss} = 0.3\%$ for the second angle.

71

Figure 5.37 shows the response of the system for $\theta_2 = 150$ degrees.
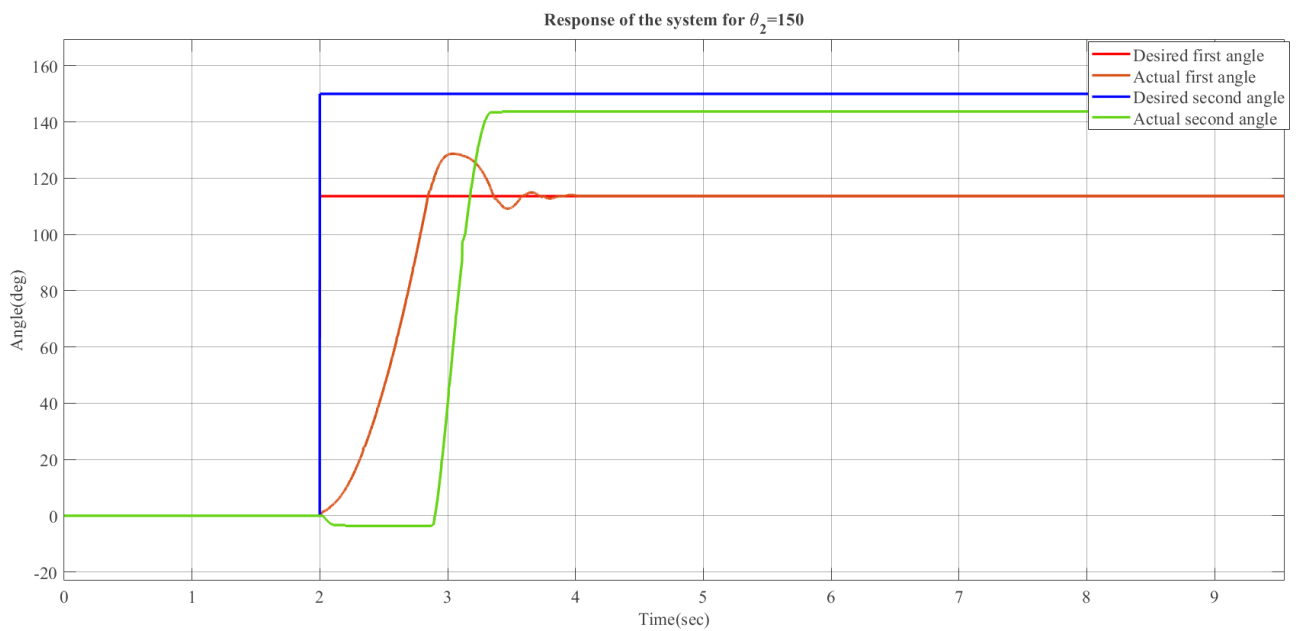
Figure 5.37: The response of the system at desired θ2=150 deg in experimental

The response has $e_{ss} = 4.2\%$ for the second angle.

The responses of the second link have a steady state error about 5% where the steady state error of first link response goes to zero, the fluctuation in steady state error is due to difference in friction values between the two links.

Previous figures show that the high deceleration in clockwise direction of the first link to return to the steady state value is the reason of having a positive second link angle in counterclockwise direction, therefore, the overshoot has a positive impact on the response.
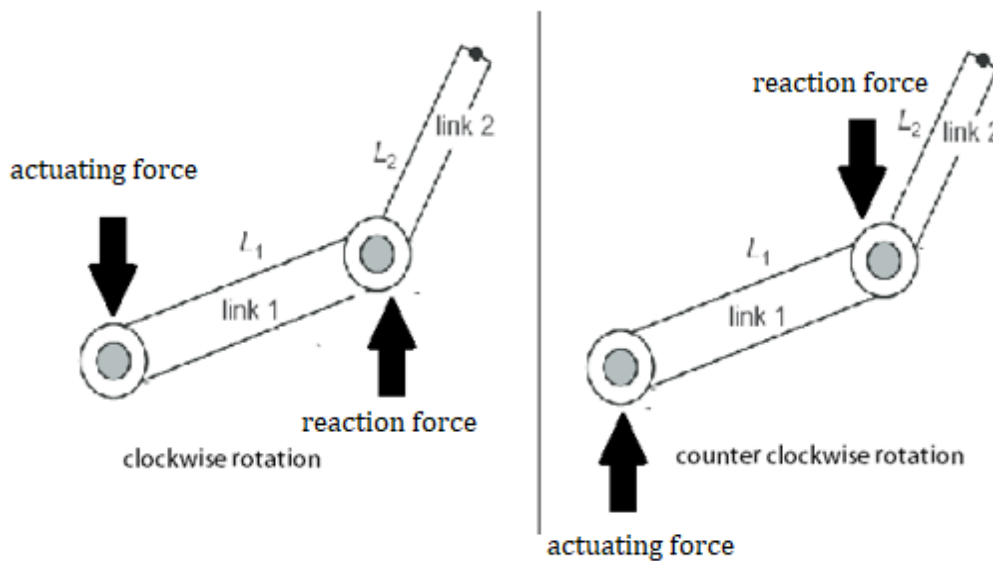
Figure 5.38: Free body diagram of the system

Figure 5.38 shows the free body diagram of the system, when the acceleration of the first link is in clockwise direction, a reaction force becomes on the other end of the link, this force makes the second link to rotate in counterclockwise direction, and vise versa is true.

5.3.3.2.2. Actuator (origin) centered circles

The block diagram of this method is shown in Figure 5.39 and a cosine theorem-based Matlab function is added where the MATLAB m files of the functions are in Appendix D.
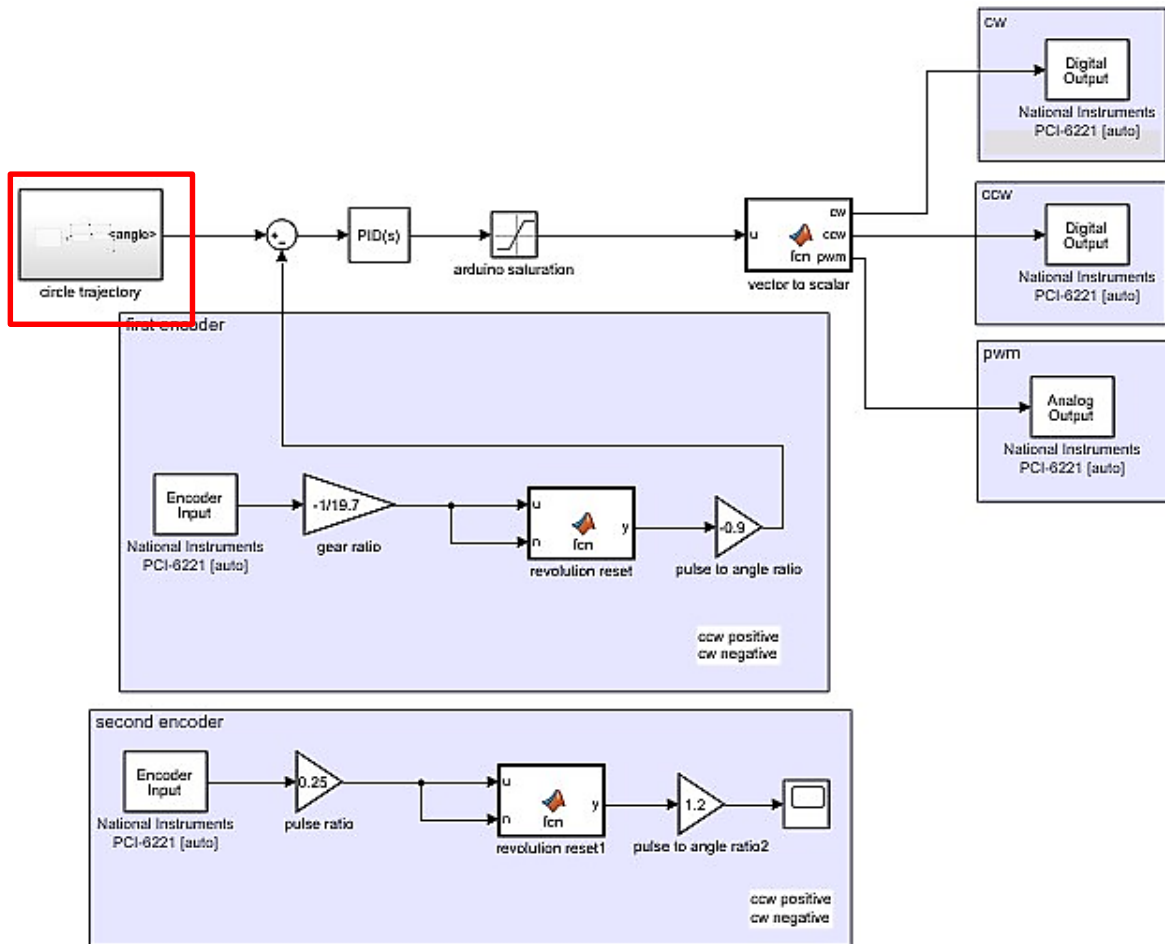
Figure 5.39: Circular trajectory control block diagram in real time simulink

The circle trajectory functions in Figure 5.39 is shown in Figure 5.40.
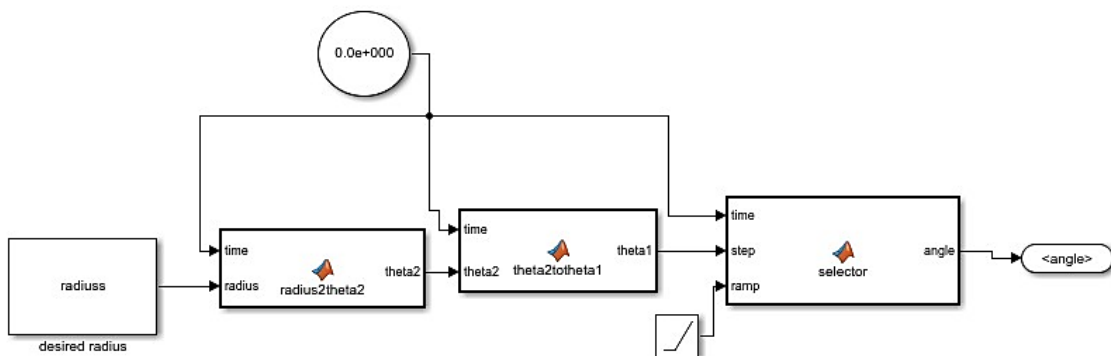
Figure 5.40: Circular trajectory block in experimental approach

73

Figure 5.41 and Figure 5.42 show the response of the system for radius=20cm. The resulting end effector position graph using xy graph is shown in Figure 5.43
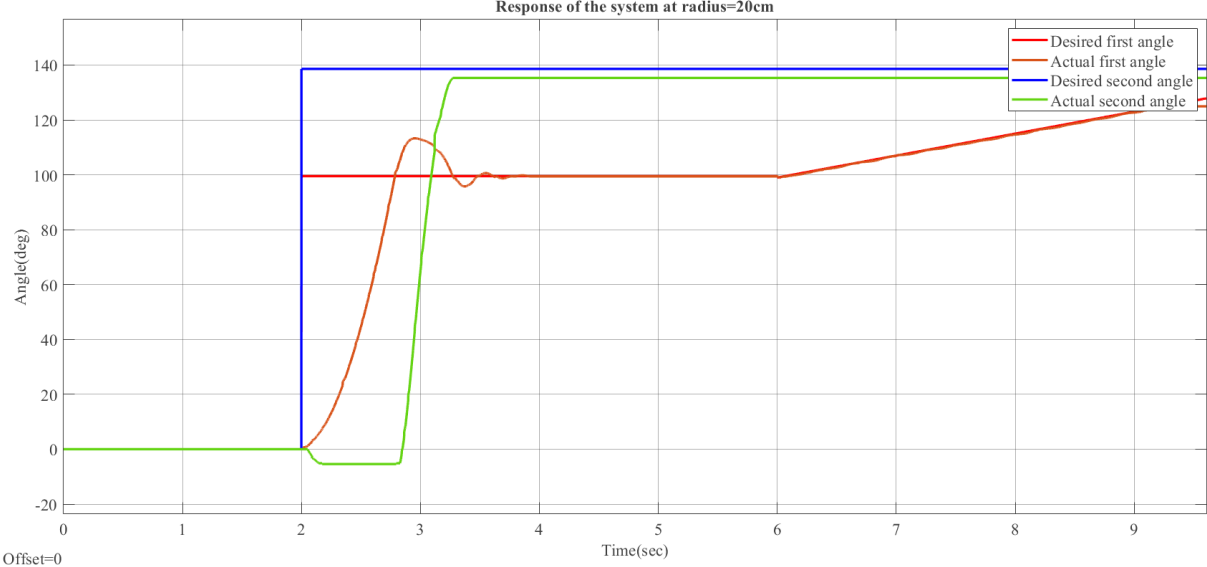
Figure 5.41: Response of the system at desired radius=20cm
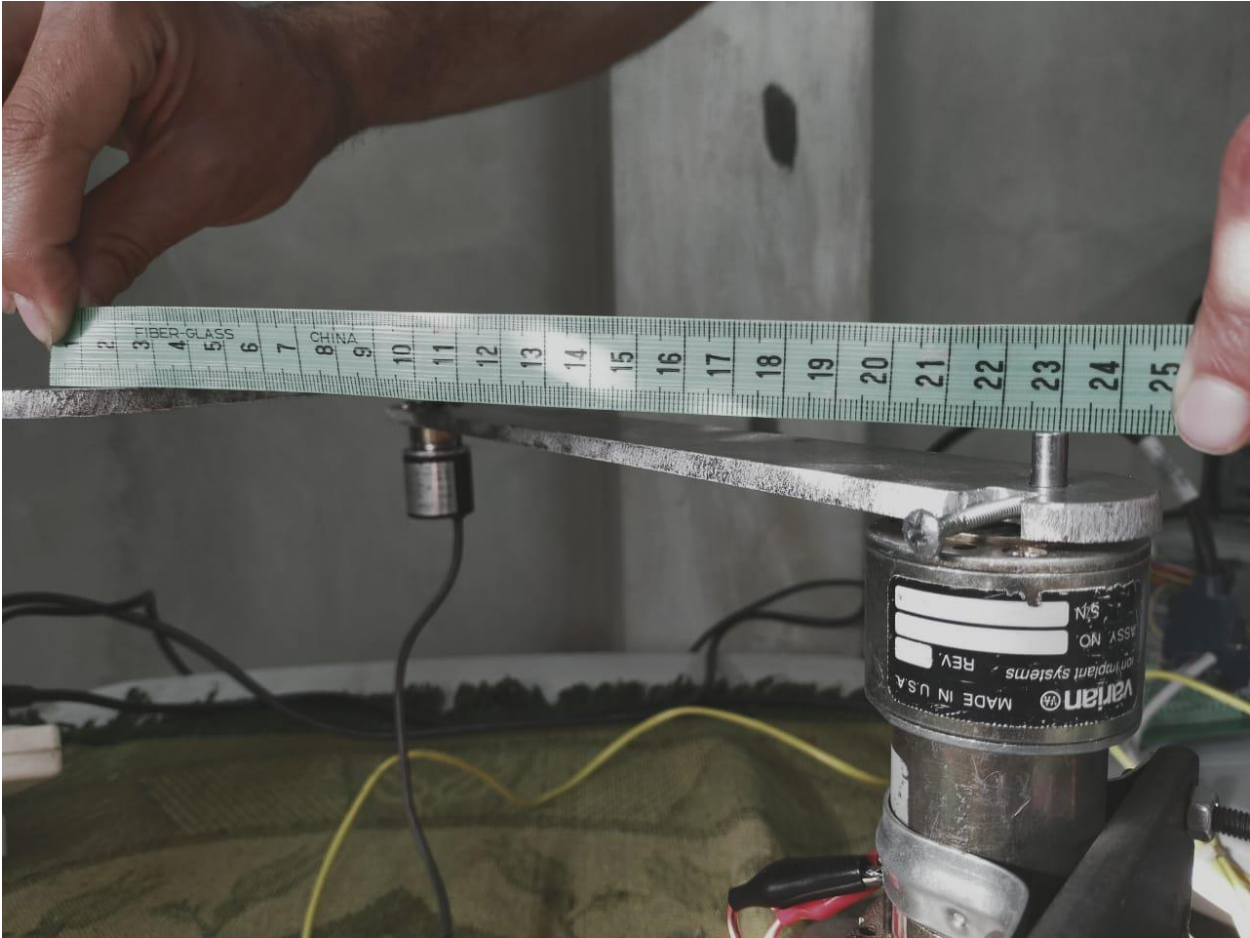
Figure 5.42: Measurement of the radius at desired radius=20cm
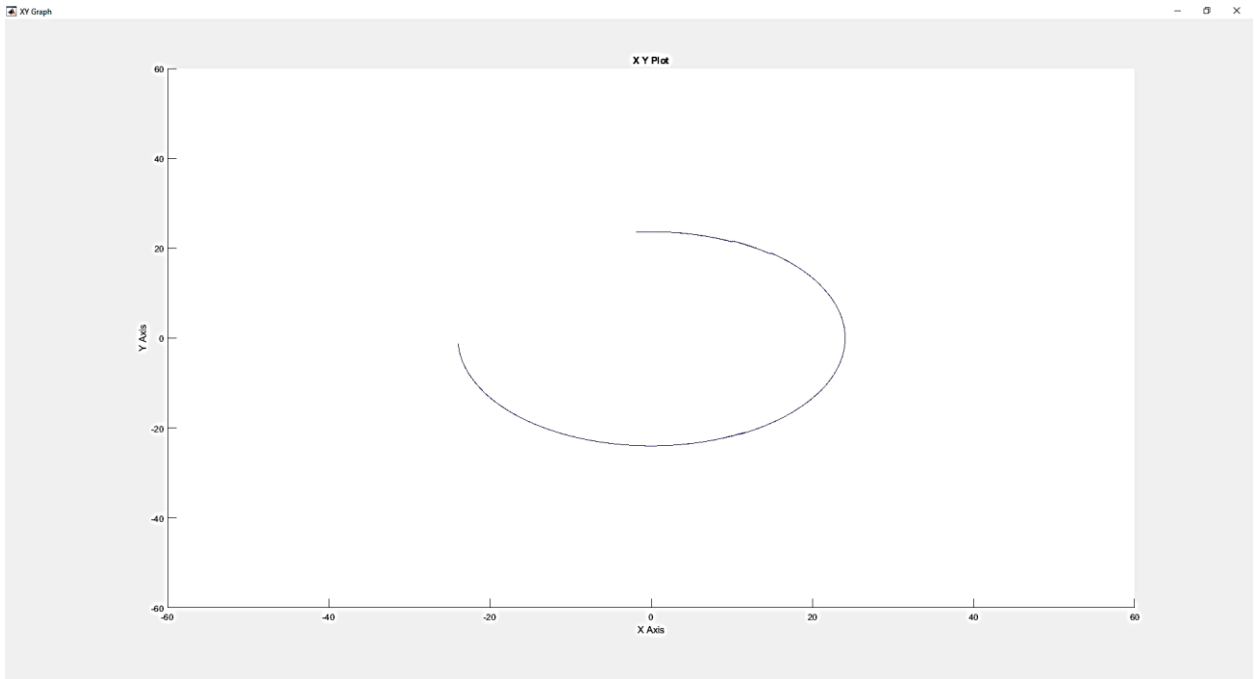
Figure 5.43: End effector position graph at radius=20cm in experimental

Figure 5.44 and Figure 5.45 show the response of the system for radius=35cm. The resulting end effector position graph using xy graph is shown in Figure 5.46
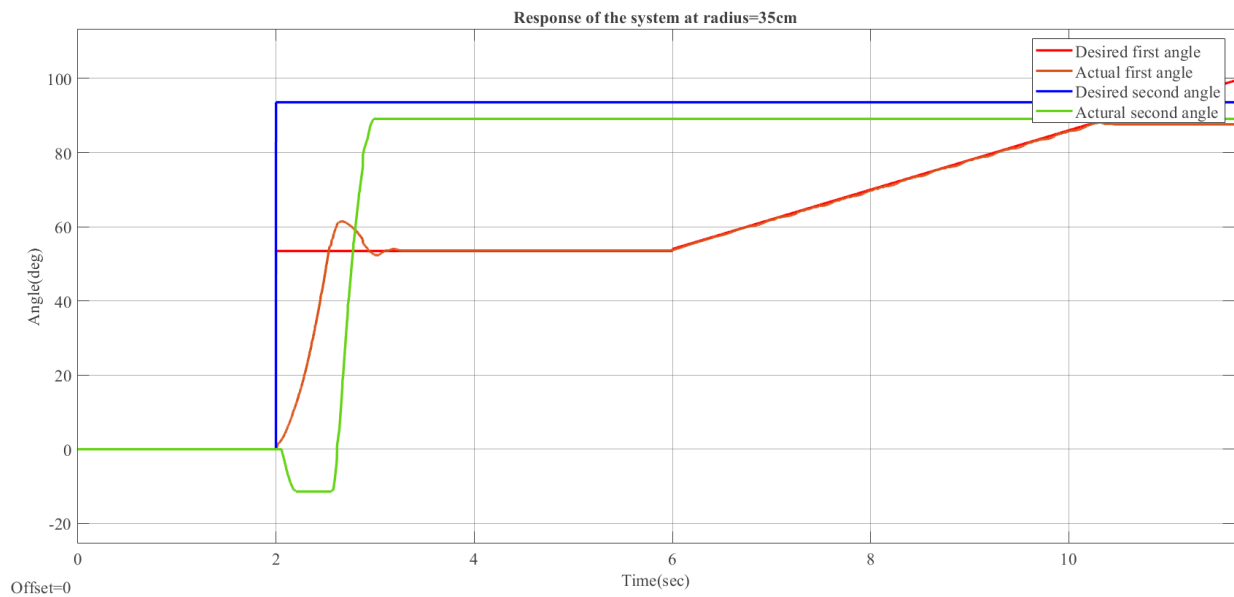
Figure 5.44: Response of the system for radius=35cm
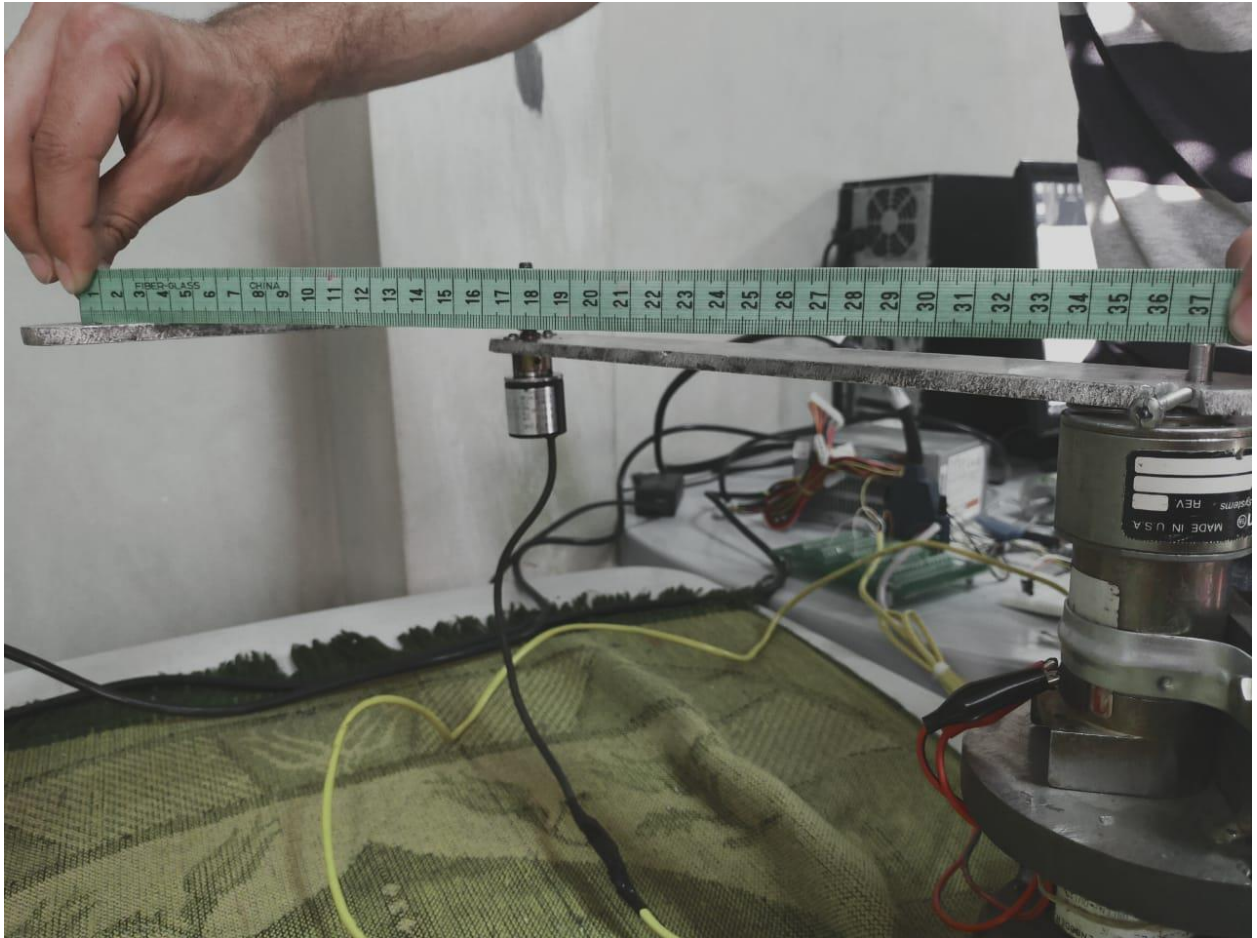
Figure 5.45: Measurement of the radius at desired radius=35cm
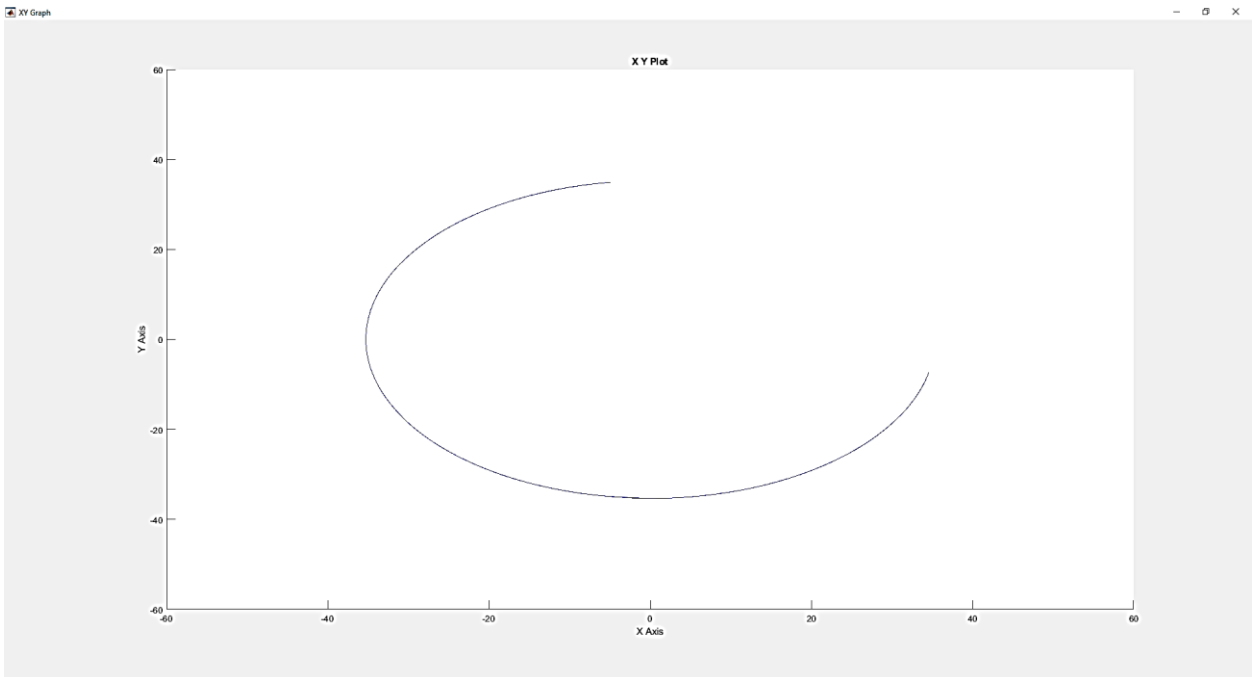
Figure 5.46: End effector position graph at radius=35cm in experimental

Figure 5.47 and Figure 5.48 show the response of the system for radius=45cm. The resulting end effector position graph using xy graph is shown in Figure 5.49
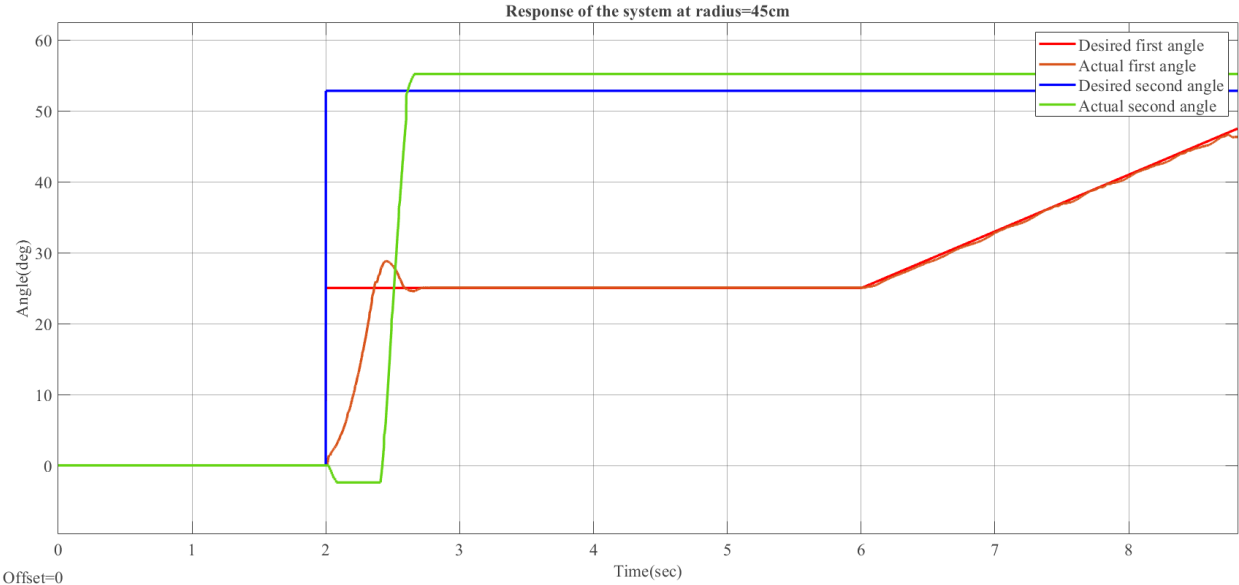
Figure 5.47: Response of the system for radius=45

Figure 5.48: Measurement of the radius at desired radius=45cm
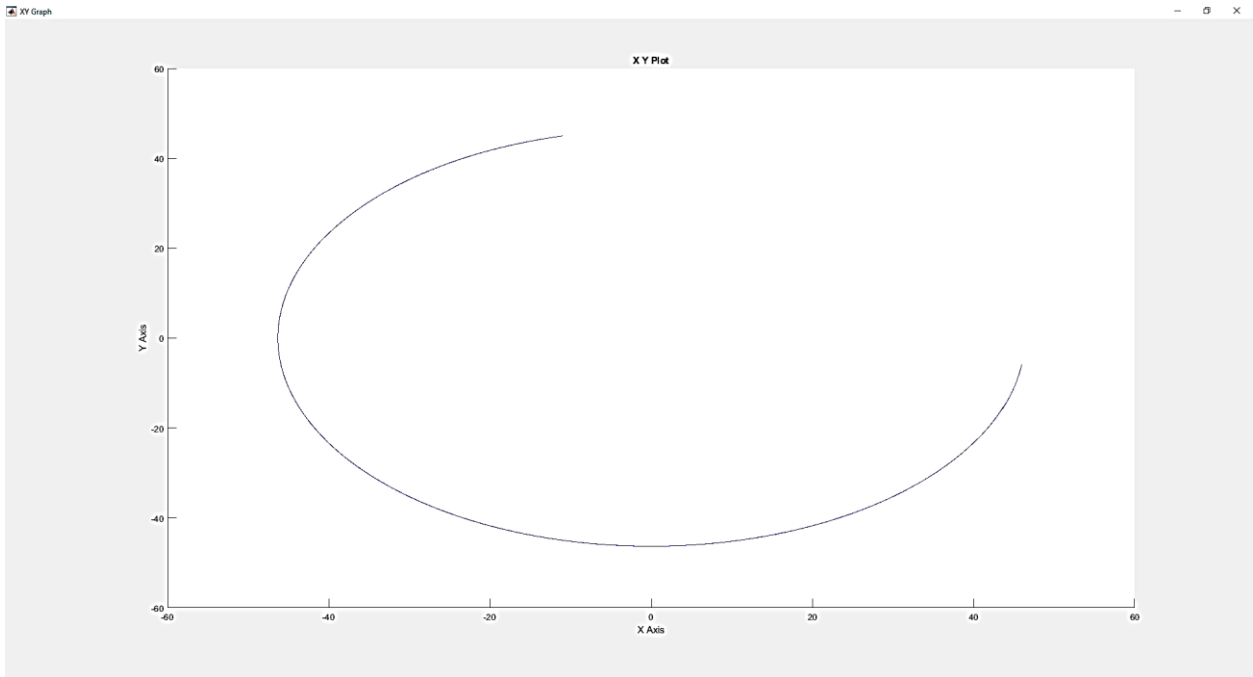
Figure 5.49: End effector position graph at radius=45cm in experimental

The resulting radius of the circles have tolerance $\pm 2cm$ based on the steady state error of the response of the second link which may be referred to the low number of the pulses of the second link encoder and the variation of the friction between the two links.

5.4   Results and recommendations

Results

The first two goals are achieved successfully using PID control of first link and by finding a relationship with the second link.

Various control strategies failed to achieve the third goal although it compensated the resulting response of the system, this result is due to the continuous change in the second link angle which could not be trackable at all times.

Recommendations

Recommendations for the control strategies of the system

Intelligent control strategies such as MPC control and fuzzy PID control are suggested to control the system.

Machine learning can be used for advanced control of the system.

Recommendations for the implementation of the system

A mobile microcontroller such as PIC microcontroller or DAQ for laptops is suggested instead of the classic DAQ because of the need of Desktop computer.

A module could be designed to replace the many electrical parts to save size.

References

1. DYNAMICS, B. *Do you love me?* 2020 6/7/2021; Available from: https://www.youtube.com/watch?v=fn3KWM1kuAw.
2. Liu, P., et al., *A survey on underactuated robotic systems: Bio-inspiration, trajectory planning and control.* Mechatronics, 2020. **72**: p. 102443.
3. Gandarias, J.M., et al. *Underactuated Gripper with Forearm Roll Estimation for Human Limbs Manipulation in Rescue Robotics*. in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2019.
4. Kobayashi, K. and T. Yoshikawa, *Controllability of Under-Actuated Planar Manipulators with One Unactuated Joint.* The International Journal of Robotics Research, 2002. **21**(5-6): p. 555-561.
5. Rosas-Flores, J.A., J. Alvarez-Gallegos, and R. Castro-Linares, *CONTROL OF AN UNDERACTUATED PLANAR 2R MANIPULATOR: EXPERIMENTAL RESULTS.* IFAC Proceedings Volumes, 2002. **35**(1): p. 265-270.
6. Mahindrakar, A.D., S. Rao, and R.N. Banavar, *Point-to-point control of a 2R planar horizontal underactuated manipulator.* Mechanism and Machine Theory, 2006. **41**(7): p. 838-844.
7. Krafes, S., C. Zakaria, and A. Saka, *A Review on the Control of Second Order Underactuated Mechanical Systems.* Complexity, 2018. **2018**: p. 1-17.
8. Siciliano, B., et al., *Robotics: Modelling, planning and control*. 2009. p. 1-623.
9. Ahmad Manasra, A.A.S., Ala Salaymeh,Yousef Natsheh, *Design, control and implementation of SCARA robot for sorting missions with machine vision*, in *Mechanical Engineering*. 2018, Palestine Polytechnic University: Hebron.
10. Spong, M.W., *Robot Dynamics and Control*. 1989: John Wiley & Sons, Inc.
11. Murray, R., S. Sastry, and L. Ze-xiang. *A Mathematical Introduction to Robotic Manipulation*. 1994.
12. Systèmes, D. *SOLIDWORKS software*. 2020 [cited 2020; Available from: https://www.solidworks.com/.
13. Beer, F.P., et al., *Mechanics of materials*. 2015.
14. components101. *L298N Motor Driver Module*. 2020 [cited 2021; Available from: https://components101.com/modules/l293n-motor-driver-module.
15. Instruments, N. *NI 6221 68-Pin Device Specifications*. 2020 [cited 2021; Available from: https://www.ni.com/pdf/manuals/375303c.pdf.
16. ARDUINO.CC. *ARDUINO MEGA 2560 REV3*. 2021 [cited 2021; Available from: https://store.arduino.cc/usa/mega-2560-r3.
17. Hashlamon, I., *Adaptive Disturbance Estimation and Compensation for Delta Robots.* 2020. **14**: p. 413-422.
18. Sharida, A., *Real time adaptive control for Delta Robots*. 2020, Palestine Polytechnic University: Hebron.
19. Xin, X., et al. *Linear Controllability and Observability of n-Link Underactuated Planar Revolute Robot Moving in Constantly Rotating Frame in Horizontal Plane*. in *2019 IEEE 58th Conference on Decision and Control (CDC)*. 2019.
20. Nise, N.S., *Control Systems Engineering, Sixth*. 2011: John Wiley & Sons, Incorporated.

Appendices

Appendix A

Pittman GM9000 motors

Gearmotor Data

Line No.	Parameter	Symbol	Units	Reduction Ratios											
				5.9:1	11.5:1	19.7:1	38.3:1	65.5:1	127.8:1	218.4:1	425.9:1	728.1:1	1419.8:1	2426.9:1	4732.5:1
MECHANICAL SPECIFICATIONS (Standard and High-Torque Gears)															
1	Max. Load Standard Gears[1]	T_L	oz·in (N·m)	175 (1.24)	175 (1.24)	175 (1.24)	175 (1.24)	175 (1.24)	175 (1.24)	175 (1.24)	175 (1.24)	175 (1.24)	175 (1.24)	175 (1.24)	175 (1.24)
2	Max. Load High-Torque Gears[1]	T_L	oz·in (N·m)	300 (2.12)	300 (2.12)	300 (2.12)	300 (2.12)	300 (2.12)	300 (2.12)	300 (2.12)	300 (2.12)	300 (2.12)	300 (2.12)	300 (2.12)	300 (2.12)
3	Gearbox Shaft Rotation[2]	—	—	CW	CW	CCW	CCW	CW	CW	CCW	CCW	CW	CW	CCW	CCW
4	Gearbox Efficiency	—	%	81	81	73	73	66	66	59	59	53	53	48	48
5	Gearbox Weight	W_G	oz (gm)	5.90 (167.3)	5.90 (167.3)	6.26 (177.5)	6.26 (177.5)	6.62 (187.7)	6.62 (187.7)	6.98 (197.9)	6.98 (197.9)	7.34 (208.1)	7.34 (208.1)	8.18 (231.9)	8.18 (231.9)
6	Gearbox Length	L_2	in max (mm max)	1.373 (34.9)	1.373 (34.9)	1.373 (34.9)	1.373 (34.9)	1.373 (34.9)	1.373 (34.9)	1.373 (34.9)	1.373 (34.9)	1.373 (34.9)	1.373 (34.9)	1.528 (38.8)	1.528 (38.8)
7	Length, GM92X2/GM94X2	L_3	in max (mm max)	3.101 (78.8)	3.101 (78.8)	3.101 (78.8)	3.101 (78.8)	3.101 (78.8)	3.101 (78.8)	3.101 (78.8)	3.101 (78.8)	3.101 (78.8)	3.101 (78.8)	3.256 (82.7)	3.256 (82.7)
8	Length, GM92X3/GM94X3	L_3	in max (mm max)	3.476 (88.3)	3.476 (88.3)	3.476 (88.3)	3.476 (88.3)	3.476 (88.3)	3.476 (88.3)	3.476 (88.3)	3.476 (88.3)	3.476 (88.3)	3.476 (88.3)	3.631 (92.2)	3.63 (92.2)
9	Length, GM92X4/GM94X4	L_3	in max (mm max)	3.676 (93.4)	3.676 (93.4)	3.676 (93.4)	3.676 (93.4)	3.676 (93.4)	3.676 (93.4)	3.676 (93.4)	3.676 (93.4)	3.676 (93.4)	3.676 (93.4)	3.831 (97.3)	3.831 (97.3)
10	Length, GM9235/GM9435	L_3	in max (mm max)	3.976 (101.0)	3.976 (101.0)	3.976 (101.0)	3.976 (101.0)	3.976 (101.0)	3.976 (101.0)	3.976 (101.0)	3.976 (101.0)	3.976 (101.0)	3.976 (101.0)	4.131 (104.9)	4.131 (104.9)
11	Length, GM9236/GM9436	L_3	in max (mm max)	4.326 (109.9)	4.326 (109.9)	4.326 (109.9)	4.326 (109.9)	4.326 (109.9)	4.326 (109.9)	4.326 (109.9)	4.326 (109.9)	4.326 (109.9)	4.326 (109.9)	4.481 (113.8)	4.481 (113.8)
MECHANICAL SPECIFICATIONS (High-Torque Wide Face Gears)															
12	Max. Load[1]	T_L	oz·in (N·m)	NOT AVAILABLE		500 (3.53)	500 (3.53)	500 (3.53)	500 (3.53)	500 (3.53)	500 (3.53)	500 (3.53)	500 (3.53)	NOT AVAILABLE	
13	Gearbox Shaft Rotation[2]	—	—			CCW	CCW	CW	CW	CCW	CCW	CW	CW		
14	Gearbox Efficiency	—	%			73	73	66	66	59	59	53	53		
15	Gearbox Weight	W_G	oz (gm)			6.52 (184.8)	6.52 (184.8)	6.88 (195.0)	6.88 (195.0)	7.24 (205.3)	7.24 (205.3)	8.08 (229.1)	8.08 (229.1)		
16	Gearbox Length	L_2	in max (mm max)			1.373 (34.9)	1.373 (34.9)	1.373 (34.9)	1.373 (34.9)	1.373 (34.9)	1.373 (34.9)	1.528 (38.8)	1.528 (38.8)		
17	Length, GM92X2/GM94X2	L_3	in max (mm max)			3.101 (78.8)	3.101 (78.8)	3.101 (78.8)	3.101 (78.8)	3.101 (78.8)	3.101 (78.8)	3.256 (82.7)	3.256 (82.7)		
18	Length, GM92X3/GM94X3	L_3	in max (mm max)			3.476 (88.3)	3.476 (88.3)	3.476 (88.3)	3.476 (88.3)	3.476 (88.3)	3.476 (88.3)	3.631 (92.2)	3.631 (92.2)		
19	Length, GM92X4/GM94X4	L_3	in max (mm max)			3.676 (93.4)	3.676 (93.4)	3.676 (93.4)	3.676 (93.4)	3.676 (93.4)	3.676 (93.4)	3.831 (97.3)	3.831 (97.3)		
20	Length, GM9235/GM9435	L_3	in max (mm max)			3.976 (101.1)	3.976 (101.1)	3.976 (101.1)	3.976 (101.1)	3.976 (101.1)	3.976 (101.1)	4.131 (104.9)	4.131 (104.9)		
21	Length, GM9236/GM9436	L_3	in max (mm max)			4.326 (109.9)	4.326 (109.9)	4.326 (109.9)	4.326 (109.9)	4.326 (109.9)	4.326 (109.9)	4.481 (113.8)	4.481 (113.8)		
NO-LOAD SPEED (All Gears)															
22	GM9X12	S_{NL}	rpm (rad/s)	1399 (147)	717 (75.1)	420 (44.0)	215 (22.5)	126 (13.2)	64.6 (6.76)	37.8 (3.96)	19.4 (2.03)	11.3 (1.18)	5.8 (.607)	3.4 (.356)	1.7 (.178)
23	GM9X32	S_{NL}	rpm (rad/s)	1189 (125)	610 (63.9)	357 (37.4)	183 (19.2)	107 (11.2)	54.9 (5.75)	32.1 (3.36)	16.5 (1.73)	9.6 (.513)	4.9 (.513)	2.9 (.304)	1.5 (.157)
24	GM9X13	S_{NL}	rpm (rad/s)	948 (99.3)	486 (50.9)	284 (29.7)	146 (15.3)	85.3 (8.93)	43.8 (4.59)	25.6 (2.68)	13.1 (1.37)	7.7 (.806)	3.9 (.408)	2.3 (.241)	1.2 (.126)
25	GM9X33	S_{NL}	rpm (rad/s)	1016 (106)	521 (54.6)	305 (31.9)	156 (16.3)	91.5 (9.58)	46.9 (4.91)	27.4 (2.87)	14.1 (1.48)	8.2 (.859)	4.2 (.440)	2.5 (.262)	1.3 (.136)
26	GM9X14	S_{NL}	rpm (rad/s)	1300 (136)	667 (69.8)	390 (40.8)	200 (20.9)	117 (12.3)	60.0 (6.28)	35.1 (3.68)	18.0 (1.88)	10.5 (1.10)	5.4 (.565)	3.2 (.335)	1.6 (.168)
27	GM9X34	S_{NL}	rpm (rad/s)	1043 (109)	535 (56.0)	313 (32.8)	160 (16.8)	93.9 (9.83)	48.1 (5.04)	28.2 (2.95)	14.4 (1.51)	8.5 (.890)	4.3 (.450)	2.5 (.262)	1.3 (.136)

[1]Represents gearbox capability only. Continuous load torque capability will vary with gear ratio, motor selection, and operating conditions.
[2]Shaft rotation is designated while looking at output shaft with motor operating in a clockwise direction.

PITTMAN®

Gearmotor Data, continued

Line No.	Parameter	Symbol	Units	Reduction Ratios											
				5.9:1	11.5:1	19.7:1	38.3:1	65.5:1	127.8:1	218.4:1	425.9:1	728.1:1	1419.8:1	2426.9:1	4732.5:1
NO-LOAD SPEED (All Gears), continued															
28	GM9X35	S_{NL}	rpm (rad/s)	1075 (112.5)	552 (57.8)	322 (33.7)	166 (17.4)	96.9 (10.1)	49.7 (5.2)	29.1 (3.0)	14.9 (1.6)	8.7 (.913)	4.4 (.468)	2.6 (.274)	1.3 (.140)
29	GM9X36	S_{NL}	rpm (rad/s)	834 (87.3)	427 (44.7)	250 (26.2)	128 (13.4)	75 (7.85)	38.5 (4.03)	22.5 (2.36)	11.5 (1.20)	6.8 (.712)	3.5 (.367)	2.0 (.209)	1.0 (.105)

Motor Data

Line No.	Parameter	Symbol	Units	GM9X12	GM9X32	GM9X13	GM9X33	GM9X14	GM9X34	GM9X35	GM9X36
30	Continuous Torque (Max.)[3]	T_C	oz·in (N·m)	1.6 (11.2×10^{-3})	2.3 (16.2×10^{-3})	3.2 (22.6×10^{-3})	4.7 (33.2×10^{-3})	3.9 (27.8×10^{-3})	6.1 (43.1×10^{-3})	6.9 (48.7×10^{-3})	9.5 (67.1×10^{-3})
31	Peak Torque (Stall)[4]	T_{PK}	oz·in (N·m)	8.4 (59.0×10^{-3})	13.8 (97.5×10^{-3})	15.6 (110.2×10^{-3})	31.6 (223.2×10^{-3})	23.9 (168.8×10^{-3})	41.3 (291.7×10^{-3})	49.4 (348.9×10^{-3})	61.8 (436.4×10^{-3})
32	Motor Constant	K_M	oz·in/√W (N·m/√W)	1.16 (8.2×10^{-3})	1.62 (11.4×10^{-3})	1.94 (13.7×10^{-3})	2.66 (18.8×10^{-3})	2.05 (14.5×10^{-3})	3.01 (21.3×10^{-3})	3.21 (22.7×10^{-3})	4.11 (29.0×10^{-3})
33	No-Load Speed	S_0	rpm (rad/s)	8251 (864.1)	7015 (734.6)	5592 (585.6)	5993 (627.6)	7666 (802.8)	6151 (644.2)	6348 (664.8)	4916 (514.8)
34	Friction Torque	T_F	oz·in (N·m)	0.4 (2.8×10^{-3})	0.5 (3.5×10^{-3})	0.5 (3.5×10^{-3})	0.6 (4.2×10^{-3})	0.5 (3.5×10^{-3})	0.6 (4.2×10^{-3})	0.65 (4.6×10^{-3})	0.8 (5.6×10^{-3})
35	Rotor Inertia	J_M	oz·in·s² (kg·m²)	2.2×10^{-4} (1.55×10^{-6})	2.7×10^{-4} (1.91×10^{-6})	3.9×10^{-4} (2.75×10^{-6})	4.6×10^{-4} (3.25×10^{-6})	5.4×10^{-4} (3.81×10^{-6})	5.9×10^{-4} (4.17×10^{-6})	7.9×10^{-4} (5.58×10^{-6})	1.0×10^{-3} (7.06×10^{-6})
36	Electrical Time Constant	τ_E	ms	0.53	0.63	0.74	0.84	0.80	0.85	0.89	1.06
37	Mechanical Time Constant	τ_M	ms	22.8	14.4	14.7	9.29	18.1	9.25	10.9	8.5
38	Viscous Damping—Infinite Source Impedance	D	oz·in/krpm (N·m/rad/s)	0.0086 (5.79×10^{-7})	0.0272 (1.83×10^{-6})	0.0113 (7.62×10^{-7})	0.0335 (2.25×10^{-6})	0.0125 (8.43×10^{-7})	0.0387 (2.60×10^{-6})	0.0450 (3.03×10^{-6})	0.0525 (3.54×10^{-6})
39	Viscous Damping—Zero Source Impedance	K_D	oz·in/krpm (N·m/rad/s)	1.00 (6.74×10^{-5})	1.94 (1.31×10^{-4})	2.78 (1.87×10^{-4})	5.23 (3.52×10^{-4})	3.11 (2.09×10^{-4})	6.68 (4.50×10^{-4})	7.6 (5.12×10^{-4})	12.5 (8.42×10^{-4})
40	Maximum Winding Temp.	θ_{MAX}	°F (°C)	311 (155)	311 (155)	311 (155)	311 (155)	311 (155)	311 (155)	311 (155)	311 (155)
41	Thermal Impedance	R_{TH}	°F/watt °C/watt	72.9 (22.7)	72.9 (22.7)	66.4 (19.1)	66.4 (19.1)	62.8 (17.1)	62.8 (17.1)	58.5 (14.7)	56.3 (13.5)
42	Thermal Time Constant	τ_{TH}	min	7.21	7.21	11.1	11.1	12.0	12.0	12.9	13.5
43	Motor Weight	W_M	oz (gm)	6.96 (197.3)	6.98 (197.9)	8.98 (254.6)	8.90 (252.3)	10.1 (286.3)	10.1 (286.3)	0.0 (TBD)	13.8 (391.2)

Model GM9XX2 Winding Data (Other windings available upon request)

Line No.	Parameter	Symbol	Units	9X12				9X32			
44	Reference Voltage	E	V	12.0	19.1	24.0	30.3	12.0	19.1	24.0	30.3
45	Torque Constant	K_T	oz·in/A (N·m/A)	1.86 (13.2×10^{-3})	2.95 (20.8×10^{-3})	3.72 (26.3×10^{-3})	4.68 (33.1×10^{-3})	2.20 (15.6×10^{-3})	3.50 (24.7×10^{-3})	4.40 (31.1×10^{-3})	5.53 (39.1×10^{-3})
46	Back-EMF Constant	K_E	V/krpm (V/rad/s)	1.38 (13.2×10^{-3})	2.18 (20.8×10^{-3})	2.75 (26.3×10^{-3})	3.46 (33.1×10^{-3})	1.63 (15.6×10^{-3})	2.59 (24.7×10^{-3})	3.25 (31.1×10^{-3})	4.09 (39.1×10^{-3})
47	Resistance	R_T	Ω	2.63	6.45	10.2	16.1	1.93	4.70	7.38	11.6
48	Inductance	L	mH	1.35	3.40	5.42	8.56	1.16	2.94	4.64	7.34
49	No-Load Current	I_{NL}	A	0.26	0.16	0.13	0.10	0.32	0.20	0.16	0.13
50	Peak Current (Stall)	I_P	A	4.56	2.96	2.35	1.88	6.22	4.06	3.25	2.60

[3]Continuous torque specified at 25°C ambient temperature and without additional heat sink.

PITTMAN®

Model GM9XX3 Winding Data (Other windings available upon request)

Line No.	Parameter	Symbol	Units	9X13				9X33			
51	Reference Voltage	E	V	12.0	19.1	24.0	30.3	12.0	19.1	24.0	30.3
52	Torque Constant	K_T	oz·in/A (N·m/A)	2.80 (19.8×10^{-3})	4.47 (31.6×10^{-3})	5.60 (39.5×10^{-3})	7.07 (49.9×10^{-3})	2.67 (18.9×10^{-3})	4.20 (29.7×10^{-3})	5.28 (37.3×10^{-3})	6.68 (47.2×10^{-3})
53	Back-EMF Constant	K_E	V/krpm (V/rad/s)	2.07 (19.8×10^{-3})	3.31 (31.6×10^{-3})	4.14 (39.5×10^{-3})	5.23 (49.9×10^{-3})	1.98 (18.9×10^{-3})	3.10 (29.7×10^{-3})	3.90 (37.3×10^{-3})	4.94 (47.2×10^{-3})
54	Resistance	R_T	Ω	2.17	5.32	8.33	13.2	1.08	2.53	3.94	6.21
55	Inductance	L	mH	1.54	3.93	6.17	9.84	0.84	2.08	3.29	5.27
56	No-Load Current	I_{NL}	A	0.20	0.13	0.10	0.08	0.30	0.19	0.15	0.12
57	Peak Current (Stall)	I_P	A	5.54	3.59	2.88	2.30	11.1	7.55	6.09	4.88

Model GM9XX4 Winding Data (Other windings available upon request)

Line No.	Parameter	Symbol	Units	9X14				9X34			
58	Reference Voltage	E	V	12.0	19.1	24.0	30.3	12.0	19.1	24.0	30.3
59	Torque Constant	K_T	oz·in/A (oz·in/A)	2.06 (14.5×10^{-3})	3.27 (23.1×10^{-3})	4.13 (29.2×10^{-3})	5.22 (36.9×10^{-3})	2.58 (18.2×10^{-3})	4.07 (28.7×10^{-3})	5.17 (36.5×10^{-3})	6.50 (45.9×10^{-3})
60	Back-EMF Constant	K_E	V/krpm (V/krpm)	1.53 (14.5×10^{-3})	2.42 (23.1×10^{-3})	3.05 (29.2×10^{-3})	3.86 (36.9×10^{-3})	1.91 (18.2×10^{-3})	3.01 (28.7×10^{-3})	3.82 (36.5×10^{-3})	4.81 (45.9×10^{-3})
61	Resistance	R_T	Ω	1.10	2.59	4.06	6.40	0.83	1.89	2.96	4.62
62	Inductance	L	mH	0.81	2.04	3.25	5.19	0.63	1.56	2.51	3.97
63	No-Load Current	I_{NL}	A	0.29	0.18	0.14	0.11	0.33	0.21	0.16	0.13
64	Peak Current (Stall)	I_P	A	10.9	7.36	5.91	4.73	14.5	10.1	8.11	6.55

Model GM9X35/9X36 Winding Data (Other windings available upon request)

Line No.	Parameter	Symbol	Units	9X35				9X36			
65	Reference Voltage	E	V	12.0	19.1	24.0	30.3	12.0	19.1	24.0	30.3
66	Torque Constant	K_T	oz·in/A (oz·in/A)	2.47 (17.4×10^{-3})	3.99 (28.2×10^{-3})	4.94 (34.9×10^{-3})	6.27 (44.3×10^{-3})	3.25 (23.0×10^{-3})	5.24 (37.0×10^{-3})	6.49 (45.8×10^{-3})	8.24 (58.2×10^{-3})
67	Back-EMF Constant	K_E	V/krpm (V/krpm)	1.83 (17.4×10^{-3})	2.95 (28.2×10^{-3})	3.65 (34.9×10^{-3})	3.65 (44.3×10^{-3})	2.4 (23.0×10^{-3})	3.88 (37.0×10^{-3})	4.8 (45.8×10^{-3})	6.09 (58.2×10^{-3})
68	Resistance	R_T	Ω	.68	1.56	2.37	3.72	0.71	1.64	2.49	3.91
69	Inductance	L	mH	.51	1.34	2.05	3.30	0.66	1.72	2.63	4.24
70	No-Load Current	I_{NL}	A	0.38	0.24	0.19	0.16	0.33	0.20	0.16	0.13
71	Peak Current (Stall)	I_P	A	17.6	12.2	10.1	8.14	16.9	11.7	9.64	7.74

PITTMAN®

Model GM9X12

Motor Speed & Current vs. Torque
(24V Winding)
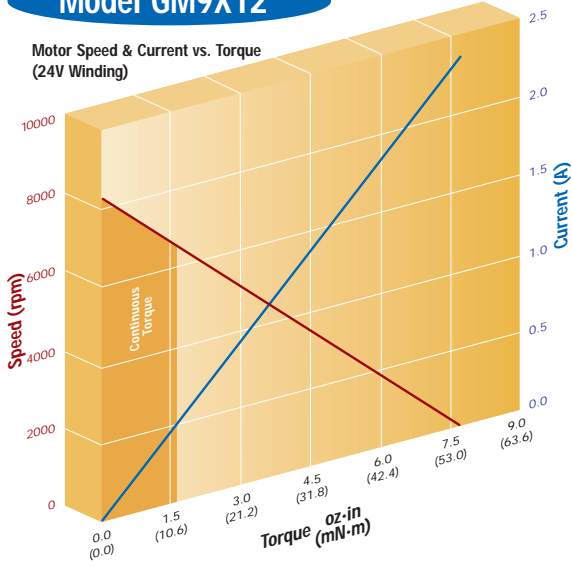
Model GM9X14

Motor Speed & Current vs. Torque
(24V Winding)
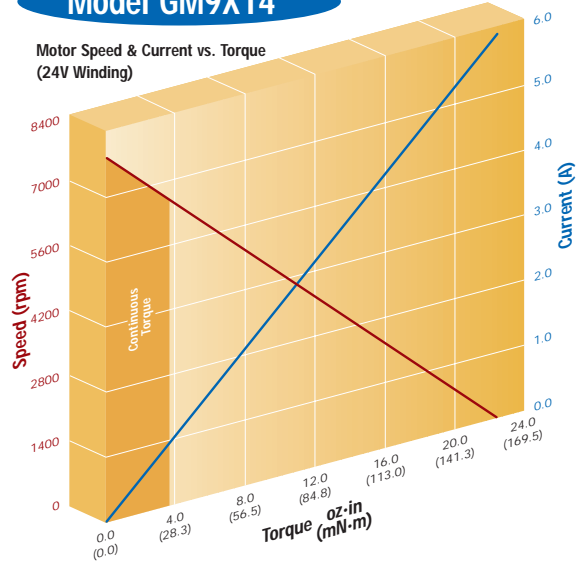
Model GM9X13

Motor Speed & Current vs. Torque
(24V Winding)
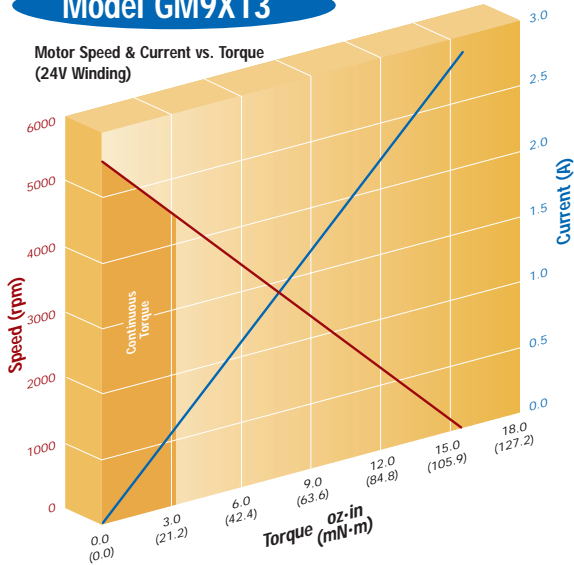
Model GM9X32

Motor Speed & Current vs. Torque
(24V Winding)

PITTMAN®

LCG-11
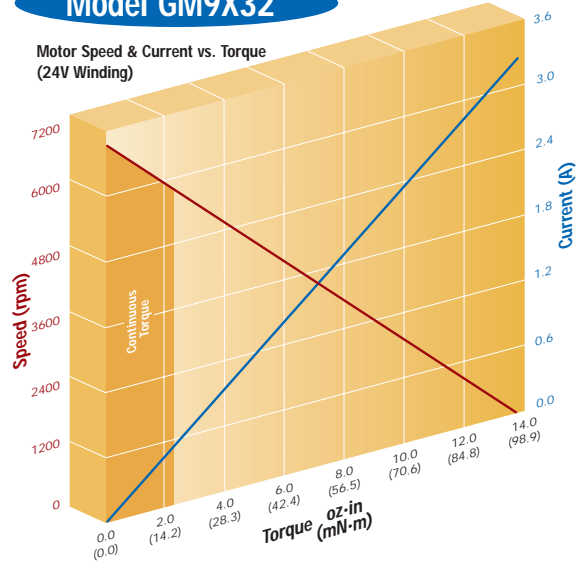
Model GM9X33

Motor Speed & Current vs. Torque
(24V Winding)
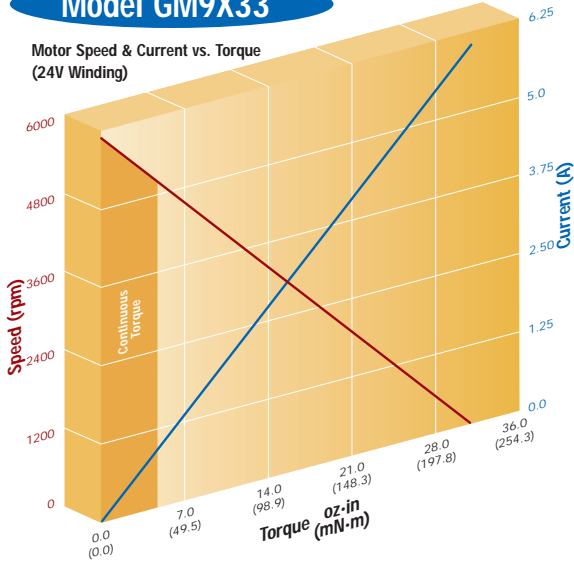
Model GM9X35

Motor Speed & Current vs. Torque
(24V Winding)
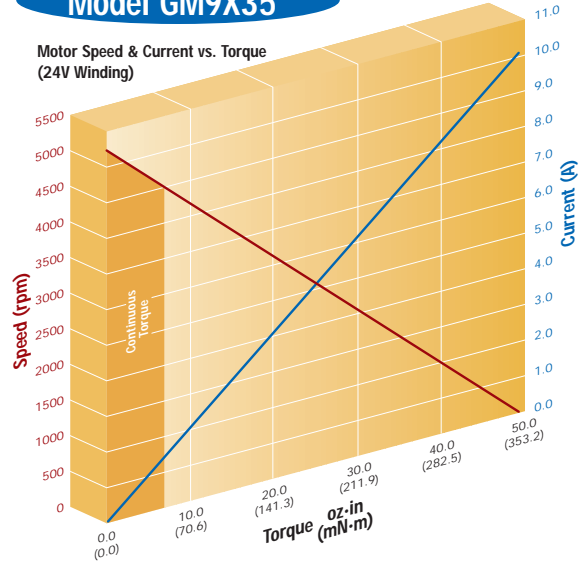
Model GM9X34

Motor Speed & Current vs. Torque
(24V Winding)
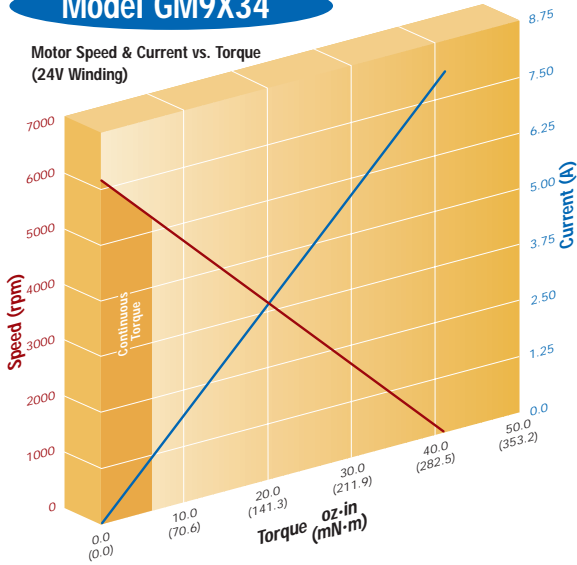
Model GM9X36

Motor Speed & Current vs. Torque
(24V Winding)

PITTMAN®
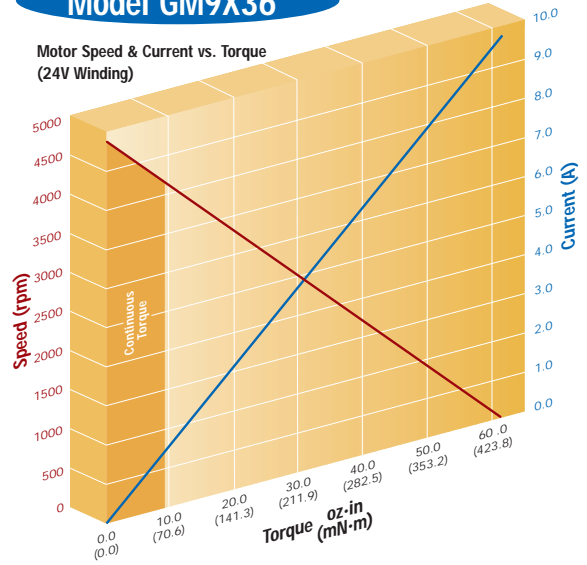
Appendix B

L298N motor driver

<u>**User Guide**</u>

L298N Dual H-Bridge Motor Driver

This dual bidirectional motor driver, is based on the very popular L298 Dual H-Bridge Motor Driver Integrated Circuit. The circuit will allow you to easily and independently control two motors of up to 2A each in both directions.It is ideal for robotic applications and well suited for connection to a microcontroller requiring just a couple of control lines per motor. It can also be interfaced with simple manual switches, TTL logic gates, relays, etc. This board equipped with power LED indicators, on-board +5V regulator and protection diodes.
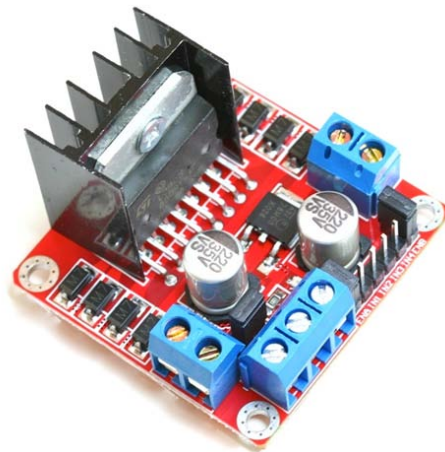
SKU: MDU-1049

<u>Brief Data:</u>

- Input Voltage: 3.2V~40Vdc.
- Driver: L298N Dual H Bridge DC Motor Driver
- Power Supply: DC 5 V - 35 V
- Peak current: 2 Amp
- Operating current range: 0 ~ 36mA
- Control signal input voltage range :
- Low: -0.3V $\leqslant$ Vin $\leqslant$ 1.5V.
- High: 2.3V $\leqslant$ Vin $\leqslant$ Vss.
- Enable signal input voltage range :
  - o   Low: -0.3 $\leqslant$ Vin $\leqslant$ 1.5V (control signal is invalid).
  - o   High: 2.3V $\leqslant$ Vin $\leqslant$ Vss (control signal active).
- Maximum power consumption: 20W (when the temperature T = 75 ℃).
- Storage temperature: -25 ℃ ~ +130 ℃.
- On-board +5V regulated Output supply (supply to controller board i.e. Arduino).
- Size: 3.4cm x 4.3cm x 2.7cm

Schematic Diagram:
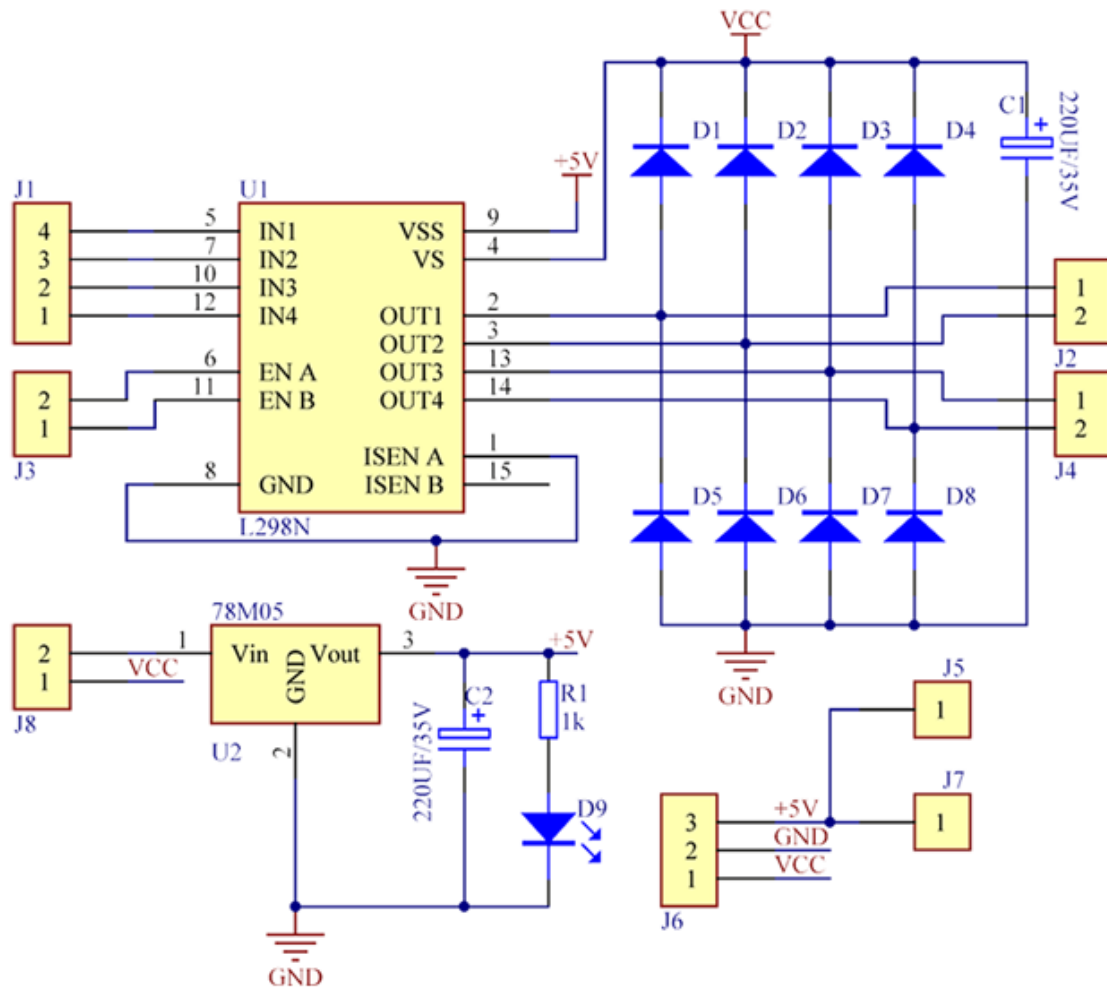
Board Dimension & Pins Function:

Output A

5V Enable

+5V Power

Power GND

+12V Power

A Enable

Logic Input

B Enable

Output B

43mm

43mm
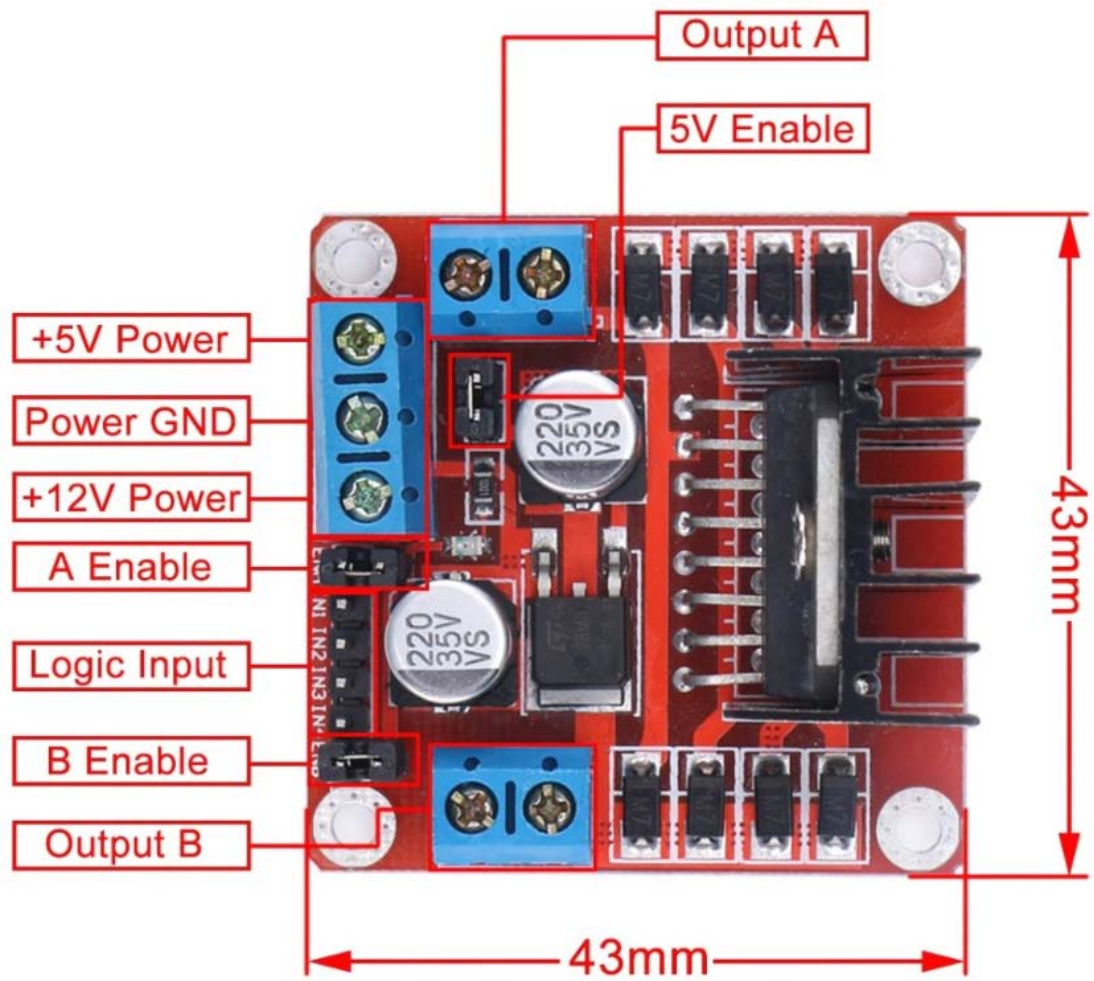
Appendix C

NI PCI-6221 data acquisition card

NI 6221

M Series Data Acquisition: 16-Bit, 250 kS/s, 16 AI, 24 DIO, 2 AO

The following specifications are typical at 25 °C, unless otherwise noted. For more information about the NI 6221, refer to the *M Series User Manual* available at *ni.com/manuals*.

Analog Input

Number of channels	8 differential or 16 single ended
ADC resolution	16 bits
DNL	No missing codes guaranteed
INL	Refer to the *AI Absolute Accuracy* section
Sample rate	
Single channel maximum	250 kS/s
Multichannel maximum (aggregate)	250 kS/s
Minimum	No minimum
Timing accuracy	50 ppm of sample rate
Timing resolution	50 ns
Input coupling	DC
Input range	±0.2 V, ±1 V, ±5 V, ±10 V
Maximum working voltage for analog inputs (signal + common mode)	±11 V of AI GND
CMRR (DC to 60 Hz)	92 dB
Input impedance	
Device on	
AI+ to AI GND	>10 GΩ in parallel with 100 pF
AI- to AI GND	>10 GΩ in parallel with 100 pF

NATIONAL INSTRUMENTS

Device off	
AI+ to AI GND	820 Ω
AI- to AI GND	820 Ω
Input bias current	±100 pA
Crosstalk (at 100 kHz)	
Adjacent channels	-75 dB
Non-adjacent channels	-90 dB
Small signal bandwidth (-3 dB)	700 kHz
Input FIFO size	4,095 samples
Scan list memory	4,095 entries
Data Transfers	
PCI/PXI	DMA (scatter-gather), interrupts, programmed I/O
USB	USB Signal Stream, programmed I/O
Overvoltage protection for all analog input and sense channels	
Device on	±25 V for up to two AI pins
Device off	±15 V for up to two AI pins
Input current during overvoltage condition	±20 mA maximum/AI pin

Settling Time for Multichannel Measurements

Accuracy, full-scale step, all ranges	
±90 ppm of step (±6 LSB)	4 μs convert interval
±30 ppm of step (±2 LSB)	5 μs convert interval
±15 ppm of step (±1 LSB)	7 μs convert interval

Typical Performance Graphs

Figure 1. Settling Error versus Time for Different Source Impedances
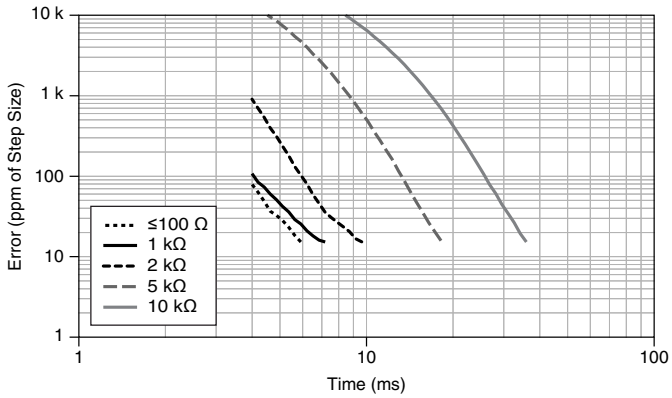
Figure 2. AI Small Signal Bandwidth
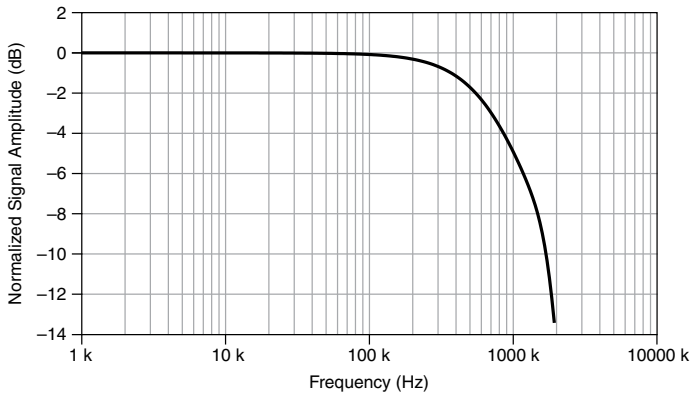
Figure 3. AI CMRR
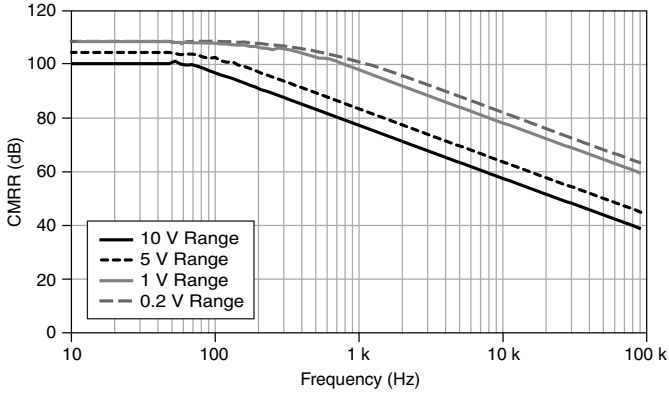
AI Absolute Accuracy

Note Accuracies listed are valid for up to one year from the device external calibration.

Table 1. AI Absolute Accuracy

Nominal Range Positive Full Scale	Nominal Range Negative Full Scale	Residual Gain Error (ppm of Reading)	Residual Offset Error (ppm of Range)	Offset Tempco (ppm of Range/°C)	Random Noise, σ (µVrms)	Absolute Accuracy at Full Scale (µV)	Sensitivity (µV)
10	-10	75	20	57	244	3,100	97.6
5	-5	85	20	60	122	1,620	48.8
1	-1	95	25	79	30	360	12.0
0.2	-0.2	135	80	175	13	112	5.2

Note Sensitivity is the smallest voltage change that can be detected. It is a function of noise.

Gain tempco	25 ppm/°C
Reference tempco	5 ppm/°C
INL error	76 ppm of range

AI Absolute Accuracy Equation

AbsoluteAccuracy = Reading · (GainError) + Range · (OffsetError) + NoiseUncertainty

> *GainError = ResidualAIGainError + GainTempco · (TempChangeFromLastInternalCal) + ReferenceTempco · (TempChangeFromLastExternalCal)*
>
> *OffsetError = ResidualAIOffsetError + OffsetTempco · (TempChangeFromLastInternalCal) + INLError*
>
> $NoiseUncertainty = \dfrac{\text{Random Noise} \cdot 3}{\sqrt{100}}$ for a coverage factor of 3 σ and averaging 100 points.

AI Absolute Accuracy Example

Absolute accuracy at full scale on the analog input channels is determined using the following assumptions:

- TempChangeFromLastExternalCal = 10 °C
- TempChangeFromLastInternalCal = 1 °C
- number_of_readings = 100
- CoverageFactor = 3 σ

For example, on the 10 V range, the absolute accuracy at full scale is as follows:

> GainError = 75 ppm + 25 ppm · 1 + 5 ppm · 10 = 150 ppm
>
> OffsetError = 20 ppm + 57 ppm · 1 + 76 ppm = 153 ppm
>
> $NoiseUncertainty = \dfrac{244\ \mu V \cdot 3}{\sqrt{100}} = 73\ \mu V$
>
> AbsoluteAccuracy = 10 V · (GainError) + 10 V · (OffsetError) + NoiseUncertainty = 3,100 µV

Analog Output

Number of channels	2
DAC resolution	16 bits
DNL	±1 LSB
Monotonicity	16 bit guaranteed
Maximum update rate	
1 channel	833 kS/s
2 channels	740 kS/s per channel
Timing accuracy	50 ppm of sample rate
Timing resolution	50 ns
Output range	±10 V

Output coupling	DC
Output impedance	0.2 Ω
Output current drive	±5 mA
Overdrive protection	±25 V
Overdrive current	10 mA
Power-on state	±20 mV[1]
Power-off glitch	400 mV for 200 ms
Output FIFO size	8,191 samples shared among channels used
Data transfers	
PCI/PXI	DMA (scatter-gather), interrupts, programmed I/O
USB	USB Signal Stream, programmed I/O
AO waveform modes	Non-periodic waveform, periodic waveform regeneration mode from onboard FIFO, periodic waveform regeneration from host buffer including dynamic update
Settling time, full-scale step, 15 ppm (1 LSB)	6 μs
Slew rate	15 V/μs
Glitch energy	
Magnitude	100 mV
Duration	2.6 μs

AO Absolute Accuracy

Absolute accuracy at full-scale numbers is valid immediately following internal calibration and assumes the device is operating within 10 °C of the last external calibration.

✎ **Note** Accuracies listed are valid for up to one year from the device external calibration.

---

[1] When the USB Screw Terminal device is powered on, the analog output signal is not defined until after USB configuration is complete.

Table 2. AO Absolute Accuracy

Nominal Range Positive Full Scale	Nominal Range Negative Full Scale	Residual Gain Error (ppm of Reading)	Gain Tempco (ppm/°C)	Residual Offset Error (ppm of Range)	Offset Tempco (ppm of Range/°C)	Absolute Accuracy at Full Scale (µV)
10	-10	90	10	40	5	3,230

Reference tempco	5 ppm/°C
INL error	128 ppm of range

AO Absolute Accuracy Equation

AbsoluteAccuracy = OutputValue · (GainError) + Range · (OffsetError)

> *GainError = ResidualGainError + GainTempco · (TempChangeFromLastInternalCal) + ReferenceTempco · (TempChangeFromLastExternalCal)*
>
> *OffsetError = ResidualOffsetError + AOOffsetTempco · (TempChangeFromLastInternalCal) + INLError*

Digital I/O/PFI

Static Characteristics

Number of channels	24 total, 8 (P0.<0..7>), 16 (PFI <0..7>/P1, PFI <8..15>/P2)
Ground reference	D GND
Direction control	Each terminal individually programmable as input or output
Pull-down resistor	50 kΩ typical, 20 kΩ minimum
Input voltage protection	±20 V on up to two pins[2]

Waveform Characteristics (Port 0 Only)

Terminals used	Port 0 (P0.<0..7>)
Port/sample size	Up to 8 bits
Waveform generation (DO) FIFO	2,047 samples
Waveform acquisition (DI) FIFO	2,047 samples

---

[2] Stresses beyond those listed under *Input voltage protection* may cause permanent damage to the device.

DI or DO Sample Clock frequency	0 MHz to 1 MHz, system and bus activity dependent
Data transfers	
PCI/PXI	DMA (scatter-gather), interrupts, programmed I/O
USB	USB Signal Stream, programmed I/O
DI or DO Sample Clock source[3]	Any PFI, RTSI, AI Sample or Convert Clock, AO Sample Clock, Ctr *n* Internal Output, and many other signals

PFI/Port 1/Port 2 Functionality

Functionality	Static digital input, static digital output, timing input, timing output
Timing output sources	Many AI, AO, counter, DI, DO timing signals
Debounce filter settings	125 ns, 6.425 µs, 2.56 ms, disable; high and low transitions; selectable per input

Recommended Operating Conditions

Level	Minimum	Maximum
Input high voltage (V_{IH})	2.2 V	5.25 V
Input low voltage (V_{IL})	0 V	0.8 V
Output high current (I_{OH}) P0.<0..7>	—	-24 mA
Output high current (I_{OH}) PFI <0..15>/P1/P2	—	-16 mA
Output low current (I_{OL}) P0.<0..7>	—	24 mA
Output low current (I_{OL}) PFI <0..15>/P1/P2	—	16 mA

Electrical Characteristics

Level	Minimum	Maximum
Positive-going threshold (VT+)	—	2.2 V
Negative-going threshold (VT-)	0.8 V	—

---

[3] The digital subsystem does not have its own dedicated internal timing engine. Therefore, a sample clock must be provided from another subsystem on the device or an external source.

Level	Minimum	Maximum
Delta VT hystersis (VT+ - VT-)	0.2 V	—
I_{IL} input low current ($V_{in} = 0$ V)	—	-10 µA
I_{IH} input high current ($V_{in} = 5$ V)	—	250 µA

Digital I/O Characteristics

Figure 4. DIO Port 0: I_{oh} versus V_{oh}
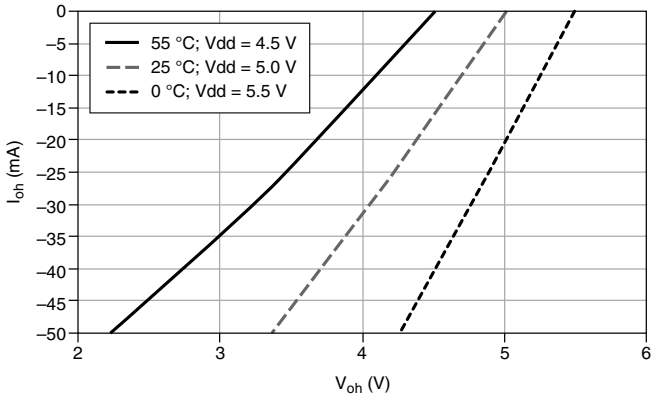
Figure 5. DIO PFI/Port 1/Port 2: I_{oh} versus V_{oh}
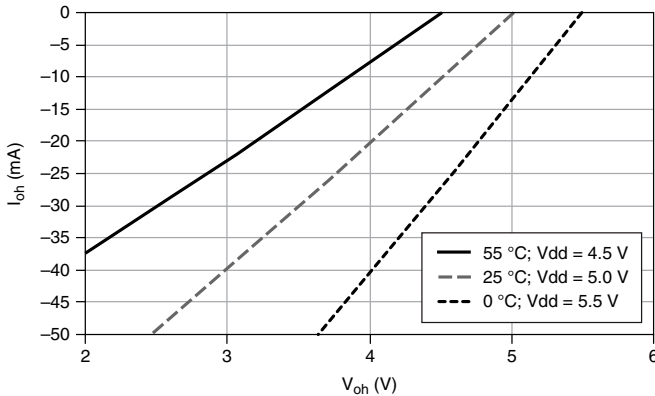
Figure 6. DIO Port 0: I_{ol} versus V_{ol}
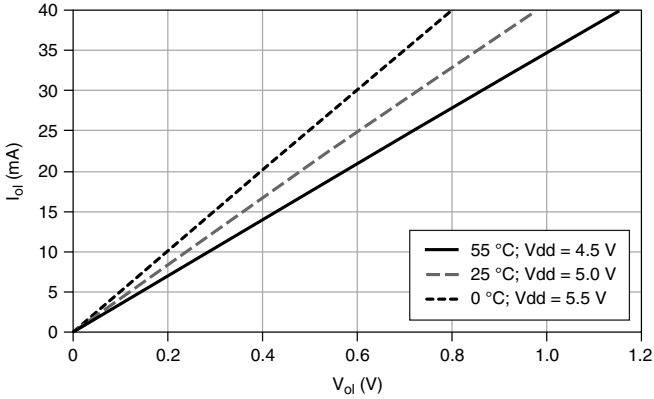
Figure 7. DIO PFI/Port 1/Port 2: I_{ol} versus V_{ol}
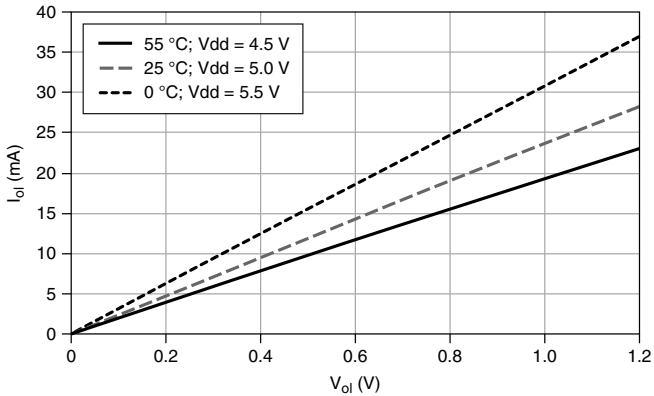
General-Purpose Counters/Timers

Number of counter/timers	2
Resolution	32 bits
Counter measurements	Edge counting, pulse, semi-period, period, two-edge separation
Position measurements	X1, X2, X4 quadrature encoding with Channel Z reloading; two-pulse encoding
Output applications	Pulse, pulse train with dynamic updates, frequency division, equivalent time sampling

Internal base clocks	80 MHz, 20 MHz, 0.1 MHz
External base clock frequency	0 MHz to 20 MHz
Base clock accuracy	50 ppm
Inputs	Gate, Source, HW_Arm, Aux, A, B, Z, Up_Down
Routing options for inputs	Any PFI, RTSI, PXI_TRIG, PXI_STAR, analog trigger, many internal signals
FIFO	2 samples
Data transfers	
PCI/PXI	Dedicated scatter-gather DMA controller for each counter/timer; interrupts, programmed I/O
USB	USB Signal Stream, programmed I/O

Frequency Generator

Number of channels	1
Base clocks	10 MHz, 100 kHz
Divisors	1 to 16
Base clock accuracy	50 ppm

Output can be available on any output PFI or RTSI terminal.

Phase-Locked Loop (PLL)

Note PCI/PXI devices only.

Number of PLLs	1
Reference signal	PXI_STAR, PXI_CLK10, RTSI <0..7>
Output of PLL	80 MHz Timebase; other signals derived from 80 MHz Timebase including 20 MHz and 100 kHz Timebases

External Digital Triggers

Source	Any PFI, RTSI, PXI_TRIG, PXI_STAR
Polarity	Software-selectable for most signals

Analog input function	Start Trigger, Reference Trigger, Pause Trigger, Sample Clock, Convert Clock, Sample Clock Timebase
Analog output function	Start Trigger, Pause Trigger, Sample Clock, Sample Clock Timebase
Counter/timer function	Gate, Source, HW_Arm, Aux, A, B, Z, Up_Down
Digital waveform generation (DO) function	Sample Clock
Digital waveform acquisition (DI) function	Sample Clock

Device-to-Device Trigger Bus

PCI	RTSI <0..7>[4]
PXI	PXI_TRIG <0..7>, PXI_STAR
USB source	None
Output selections	10 MHz Clock, frequency generator output, many internal signals
Debounce filter settings	125 ns, 6.425 μs, 2.56 ms, disable; high and low transitions; selectable per input

Bus Interface

PCI/PXI	3.3 V or 5 V signal environment
USB	USB 2.0 Hi-Speed or full-speed[5, 6]
DMA channels (PCI/PXI)	6, can be used for analog input, analog output, digital input, digital output, counter/timer 0, counter/timer 1
USB Signal Stream	4, can be used for analog input, analog output, counter/timer 0, counter/timer 1

The PXI device supports one of the following features:

- May be installed in PXI Express hybrid slots
- Or, may be used to control SCXI in PXI/SCXI combo chassis

---

[4] In other sections of this document, RTSI refers to RTSI <0..7> for the PCI devices or PXI_TRIG <0..7> for PXI devices.

[5] If you are using an USB M Series device in full-speed mode, device performance will be lower and you will not be able to achieve maximum sample/update rates.

[6] Operating on a full-speed bus may result in lower performance.

Table 3. PXI/SCXI Combo and PXI Express Chassis Compatibility

M Series Part Number	SCXI Control in PXI/SCXI Combo Chassis	PXI Express Hybrid Slot Compatible
191332B-03	No	Yes
191332B-13	Yes	No
191322A-0x	Yes	No

Power Requirements

Current draw from bus during no-load condition[7]

+5 V	0.02 A
+3.3 V	0.25 A
+12 V	0.15 A

Current draw from bus during AI and AO overvoltage condition[7]

+5 V	0.02 A
+3.3 V	0.25 A
+12 V	0.25 A

⚠ **Caution** USB devices must be powered with an NI offered AC adapter or a National Electric Code (NEC) Class 2 DC source that meets the power requirements for the device and has appropriate safety certification marks for country of use.

USB power supply requirements	11 to 30 VDC, 20 W, locking or non-locking power jack with 0.080 in. diameter center pin, 5/16-32 thread for locking collars

Current Limits

⚠ **Caution** Exceeding the current limits may cause unpredictable behavior by the device and/or PC/chassis.

PCI, +5 V terminal	1 A maximum[8]

---

[7] Does not include P0/PFI/P1/P2 and +5 V terminals.

[8] Older revisions have a self-resetting fuse that opens when current exceeds this specification. Newer revisions have a traditional fuse that opens when current exceeds this specification. This fuse is not customer-replaceable; if the fuse permanently opens, return the device to NI for repair.

PXI

+5 V terminal	1 A maximum
P0/PFI/P1/P2 and +5 V terminals combined	2 A maximum[8]

USB

+5 V terminal	1 A maximum[8]
P0/PFI/P1/P2 and +5 V terminals combined	2 A maximum
Power supply fuse	2 A, 250 V

Physical Characteristics

Dimensions

PCI printed circuit board	10.6 cm × 15.5 cm (4.2 in. × 6.1 in.)
PXI printed circuit board	Standard 3U PXI
USB Screw Terminal enclosure (includes connectors)	26.67 cm × 17.09 cm × 4.45 cm (10.5 in. × 6.73 in. × 1.75 in.)
USB BNC enclosure (includes connectors)	28.6 cm × 17 cm × 6.9 cm (11.25 in. × 6.7 in. × 2.7 in.)
USB OEM	Refer to the *NI USB-622x/625x/628x OEM User Guide*

Weight

PCI	92 g (3.2 oz)
PXI	162 g (5.7 oz)
USB Screw Terminal	1.2 kg (2 lb 10 oz)
USB OEM	131 g (4.6 oz)

I/O connector

PCI/PXI	1 68-pin VHDCI
USB Screw Terminal	64 screw terminals
USB BNC	20 BNCs and 30 screw terminals
USB Screw Terminal/BNC screw terminal wiring	16 to 28 AWG

Calibration

Recommended warm-up time	
PCI/PXI	15 minutes
USB	30 minutes
Calibration interval	1 year

Maximum Working Voltage

Maximum working voltage refers to the signal voltage plus the common-mode voltage.

Channel-to-earth	11 V, Measurement Category I

Measurement Category I is for measurements performed on circuits not directly connected to the electrical distribution system referred to as MAINS voltage. MAINS is a hazardous live electrical supply system that powers equipment. This category is for measurements of voltages from specially protected secondary circuits. Such voltage measurements include signal levels, special equipment, limited-energy parts of equipment, circuits powered by regulated low-voltage sources, and electronics.

⚠ **Caution** Do not use for measurements within Categories II, III, or IV.

🖎 **Note** Measurement Categories CAT I and CAT O (Other) are equivalent. These test and measurement circuits are not intended for direct connection to the MAINS building installations of Measurement Categories CAT II, CAT III, or CAT IV.

Environmental

Operating temperature	
PCI/PXI	0 ºC to 55 ºC
USB	0 ºC to 45 ºC
Storage temperature	-20 ºC to 70 ºC
Humidity	10% RH to 90% RH, noncondensing
Maximum altitude	2,000 m
Pollution Degree (indoor use only)	2

Indoor use only.

Shock and Vibration (PXI Only)

Operational shock	30 g peak, half-sine, 11 ms pulse (Tested in accordance with IEC 60068-2-27. Test profile developed in accordance with MIL-PRF-28800F.)
Random vibration	
Operating	5 Hz to 500 Hz, 0.3 g_{rms}
Nonoperating	5 Hz to 500 Hz, 2.4 g_{rms} (Tested in accordance with IEC 60068-2-64. Nonoperating test profile exceeds the requirements of MIL-PRF-28800F, Class 3.)

Safety

This product is designed to meet the requirements of the following electrical equipment safety standards for measurement, control, and laboratory use:

- IEC 61010-1, EN 61010-1
- UL 61010-1, CSA 61010-1

> **Note** For UL and other safety certifications, refer to the product label or the *Online Product Certification* section.

Electromagnetic Compatibility

This product meets the requirements of the following EMC standards for electrical equipment for measurement, control, and laboratory use:

- EN 61326-1 (IEC 61326-1): Class A emissions; Basic immunity
- EN 55011 (CISPR 11): Group 1, Class A emissions
- EN 55022 (CISPR 22): Class A emissions
- EN 55024 (CISPR 24): Immunity
- AS/NZS CISPR 11: Group 1, Class A emissions
- AS/NZS CISPR 22: Class A emissions
- FCC 47 CFR Part 15B: Class A emissions
- ICES-001: Class A emissions

> **Note** In the United States (per FCC 47 CFR), Class A equipment is intended for use in commercial, light-industrial, and heavy-industrial locations. In Europe, Canada, Australia and New Zealand (per CISPR 11) Class A equipment is intended for use only in heavy-industrial locations.

> **Note** Group 1 equipment (per CISPR 11) is any industrial, scientific, or medical equipment that does not intentionally generate radio frequency energy for the treatment of material or inspection/analysis purposes.

> **Note** For EMC declarations and certifications, and additional information, refer to the *Online Product Certification* section.

CE Compliance $C \in$

This product meets the essential requirements of applicable European Directives, as follows:

- 2014/35/EU; Low-Voltage Directive (safety)
- 2014/30/EU; Electromagnetic Compatibility Directive (EMC)
- 2011/65/EU; Restriction of Hazardous Substances (RoHS)

Online Product Certification

Refer to the product Declaration of Conformity (DoC) for additional regulatory compliance information. To obtain product certifications and the DoC for this product, visit *ni.com/certification*, search by model number or product line, and click the appropriate link in the Certification column.

Environmental Management

NI is committed to designing and manufacturing products in an environmentally responsible manner. NI recognizes that eliminating certain hazardous substances from our products is beneficial to the environment and to NI customers.

For additional environmental information, refer to the *Minimize Our Environmental Impact* web page at *ni.com/environment*. This page contains the environmental regulations and directives with which NI complies, as well as other environmental information not included in this document.

Waste Electrical and Electronic Equipment (WEEE)

> **EU Customers** At the end of the product life cycle, all NI products must be disposed of according to local laws and regulations. For more information about how to recycle NI products in your region, visit *ni.com/environment/weee*.

电子信息产品污染控制管理办法（中国 **RoHS**）

> **中国客户** National Instruments 符合中国电子信息产品中限制使用某些有害物质指令（RoHS）。关于 National Instruments 中国 RoHS 合规性信息，请登录 ni.com/environment/rohs_china。（For information about China RoHS compliance, go to ni.com/environment/rohs_china.）

Device Pinouts

Figure 8. NI PCI/PXI-6221 Pinout

AI 0	68	34	AI 8
AI GND	67	33	AI 1
AI 9	66	32	AI GND
AI 2	65	31	AI 10
AI GND	64	30	AI 3
AI 11	63	29	AI GND
AI SENSE	62	28	AI 4
AI 12	61	27	AI GND
AI 5	60	26	AI 13
AI GND	59	25	AI 6
AI 14	58	24	AI GND
AI 7	57	23	AI 15
AI GND	56	22	AO 0
AO GND	55	21	AO 1
AO GND	54	20	NC
D GND	53	19	P0.4
P0.0	52	18	D GND
P0.5	51	17	P0.1
D GND	50	16	P0.6
P0.2	49	15	D GND
P0.7	48	14	+5 V
P0.3	47	13	D GND
PFI 11/P2.3	46	12	D GND
PFI 10/P2.2	45	11	PFI 0/P1.0
D GND	44	10	PFI 1/P1.1
PFI 2/P1.2	43	9	D GND
PFI 3/P1.3	42	8	+5 V
PFI 4/P1.4	41	7	D GND
PFI 13/P2.5	40	6	PFI 5/P1.5
PFI 15/P2.7	39	5	PFI 6/P1.6
PFI 7/P1.7	38	4	D GND
PFI 8/P2.0	37	3	PFI 9/P2.1
D GND	36	2	PFI 12/P2.4
D GND	35	1	PFI 14/P2.6

CONNECTOR 0
(AI 0-15)

TERMINAL 68 — TERMINAL 34

TERMINAL 35 — TERMINAL 1

NC = No Connect

Figure 9. NI USB-6221 Screw Terminal Pinout

AI 0	1		17	AI 4	
AI 8	2		18	AI 12	
AI GND	3		19	AI GND	
AI 1	4		20	AI 5	
AI 9	5		21	AI 13	
AI GND	6		22	AI GND	
AI 2	7		23	AI 6	
AI 10	8		24	AI 14	
AI GND	9		25	AI GND	
AI 3	10		26	AI 7	
AI 11	11		27	AI 15	
AI GND	12		28	AI GND	
AI SENSE	13		29	NC	
AI GND	14		30	AI GND	
AO 0	15		31	AO 1	
AO GND	16		32	AO GND	

P0.0	65		81	PFI 8/P2.0	
P0.1	66		82	D GND	
P0.2	67		83	PFI 9/P2.1	
P0.3	68		84	D GND	
P0.4	69		85	PFI 10/P2.2	
P0.5	70		86	D GND	
P0.6	71		87	PFI 11/P2.3	
P0.7	72		88	D GND	
PFI 0/P1.0	73		89	PFI 12/P2.4	
PFI 1/P1.1	74		90	D GND	
PFI 2/P1.2	75		91	PFI 13/P2.5	
PFI 3/P1.3	76		92	D GND	
PFI 4/P1.4	77		93	PFI 14/P2.6	
PFI 5/P1.5	78		94	D GND	
PFI 6/P1.6	79		95	PFI 15/P2.7	
PFI 7/P1.7	80		96	+5 V	

NC = No Connect

Figure 10. NI USB-6221 BNC Top Panel and Pinout

Connector and LED Information

The *I/O Connector Signal Descriptions*, *+5 V Power Source*, and *USER 1 and USER 2* sections contain information about M Series connector signals, power, and user-defined terminals. The *LED Patterns* section contains information about M Series USB device LEDs.

Note Refer to Appendix A, *Module/Device-Specific Information*, for device I/O connector pinouts.

I/O Connector Signal Descriptions

Table 3-1 describes the signals found on the I/O connectors. Not all signals are available on all devices.

Table 3-1. I/O Connector Signals

Signal Name	Reference	Direction	Description
AI GND	—	—	**Analog Input Ground**—These terminals are the reference point for single-ended AI measurements in RSE mode and the bias current return point for DIFF measurements. All three ground references—AI GND, AO GND, and D GND—are connected on the device. Refer to the *Connecting Analog Input Signals* section of Chapter 4, *Analog Input*.
AI <0..79>	Varies	Input	**Analog Input Channels**—For single-ended measurements, each signal is an analog input voltage channel. In RSE mode, AI GND is the reference for these signals. In NRSE mode, the reference for each AI <0..15> signal is AI SENSE; the reference for each AI <16..63> and AI <64..79> signal is AI SENSE 2*. For differential measurements, AI 0 and AI 8 are the positive and negative inputs of differential analog input channel 0. Similarly, the following signal pairs also form differential input channels: <AI 1, AI 9>, <AI 2, AI 10>, <AI 3, AI 11>, and so on. Refer to the *Connecting Analog Input Signals* section of Chapter 4, *Analog Input*.
AI SENSE, AI SENSE 2	—	Input	**Analog Input Sense**—In NRSE mode, the reference for each AI <0..15> signal is AI SENSE; the reference for each AI <16..63> and AI <64..79> signal is AI SENSE 2*. Refer to the *Connecting Analog Input Signals* section of Chapter 4, *Analog Input*.
AO <0..3>	AO GND	Output	**Analog Output Channels**—These terminals supply the voltage output. Refer to the *Connecting Analog Output Signals* section of Chapter 5, *Analog Output*.
AO GND	—	—	**Analog Output Ground**—AO GND is the reference for AO <0..3>. All three ground references—AI GND, AO GND, and D GND—are connected on the device. Refer to the *Connecting Analog Output Signals* section of Chapter 5, *Analog Output*.

Table 3-1. I/O Connector Signals (Continued)

Signal Name	Reference	Direction	Description
D GND	—	—	**Digital Ground**—D GND supplies the reference for P0.<0..31>, PFI <0..15>/P1/P2, and +5 V. All three ground references—AI GND, AO GND, and D GND—are connected on the device. Refer to the *Connecting Digital I/O Signals* section of Chapter 6, *Digital I/O*.
P0.<0..31>	D GND	Input or Output	**Port 0 Digital I/O Channels**—You can individually configure each signal as an input or output. Refer to the *Connecting Digital I/O Signals* section of Chapter 6, *Digital I/O*.
APFI <0,1>	AO GND or AI GND	Input	**Analog Programmable Function Interface Channels**—Each APFI signal can be used as AO external reference inputs for AO <0..3>, AO external offset input, or as an analog trigger input. APFI <0,1> are referenced to AI GND when they are used as analog trigger inputs. APFI <0,1> are referenced to AO GND when they are used as AO external offset or reference inputs. These functions are not available on all devices; refer to the specifications for your device. Refer to the *APFI <0,1> Terminals* section of Chapter 11, *Triggering*.
+5 V	D GND	Output	**+5 V Power Source**—These terminals provide a fused +5 V power source. Refer to the *+5 V Power Source* section for more information.
PFI <0..7>/ P1.<0..7>	D GND	Input or Output	**Programmable Function Interface or Port 1 Digital I/O Channels**—Each of these terminals can be individually configured as a PFI terminal or a digital I/O terminal. As an input, each PFI terminal can be used to supply an external source for AI, AO, DI, and DO timing signals or counter/timer inputs. As a PFI output, you can route many different internal AI, AO, DI, or DO timing signals to each PFI terminal. You also can route the counter/timer outputs to each PFI terminal. As a Port 1 digital I/O signal, you can individually configure each signal as an input or output. Refer to the *Connecting Digital I/O Signals* section of Chapter 6, *Digital I/O*, or to Chapter 8, *PFI*.

Table 3-1. I/O Connector Signals (Continued)

Signal Name	Reference	Direction	Description
PFI <8..15>/ P2.<0..7>	D GND	Input or Output	**Programmable Function Interface or Port 2 Digital I/O Channels**—Each of these terminals can be individually configured as a PFI terminal or a digital I/O terminal. As an input, each PFI terminal can be used to supply an external source for AI, AO, DI, and DO timing signals or counter/timer inputs. As a PFI output, you can route many different internal AI, AO, DI, or DO timing signals to each PFI terminal. You also can route the counter/timer outputs to each PFI terminal. As a Port 2 digital I/O signal, you can individually configure each signal as an input or output. Refer to the *Connecting Digital I/O Signals* section of Chapter 6, *Digital I/O*, or to Chapter 8, *PFI*. Refer to Table 7-6, *68-Pin Device Default NI-DAQmx Counter/Timer Pins*, to find the default NI-DAQmx counter/timer pins for most M Series devices.
USER <1,2>	—	—	**User-Defined Channels**—On USB-62xx BNC devices, the USER <1,2> BNC connectors allow you to use a BNC connector for a digital or timing I/O signal of your choice. The USER <1,2> BNC connectors are internally routed to the USER <1,2> screw terminals. Refer to the *USER 1 and USER 2* section for more information.
CHS GND	—	—	**Chassis Ground**[†]—This terminal connects to the USB-62xx BNC device metal enclosure. You can connect your cable's shield wire to CHS GND for a ground connection. Refer to the *USB Device Chassis Ground* section of Chapter 1, *Getting Started*.
NC	—	—	**No connect**—Do *not* connect signals to these terminals.

[*] On NI 6225 devices, the reference for each AI <16..63> signal is AI SENSE 2, and each AI <64..79> signal is AI SENSE in NRSE mode.

[†] USB-62xx Screw Terminal users can connect the shield of a shielded cable to the chassis ground lug for a ground connection. The chassis ground lug is not available on all device versions.

Appendix D

Matlab m.file code

```matlab
%PID control 5.3.1 and LQR control 5.3.2
%parameters of the model
global m1 m2 L1 L2 r1 r2 iz1 iz2
m1=0.462; %mass of the first link
m2=0.384; %mass of the second link
L1=0.25;  %length of the first link
L2=0.25;  %length of the second link
r1=0.125; %distance between first joint and center of gravity of first link
r2=0.125; %distance between second joint and center of gravity of second link
iz1=0.0105778; %mass moment of inertia of first link
iz2=0.0087161; %mass moment of inertia of second link
%--------------------------
%manipulator model function
function result = robot_model_underactuated(inputs)
global m1 m2 L1 r1 r2 iz1 iz2; % the parameters of the system
%alpha,beta,delta constants
alpha=iz1+iz2+m1*(r1^2)+m2*(L1^2+r2^2);
beta=m2*L1*r2;
delta=iz2+m2*(r2^2);
%inputs to the model
torque=[inputs(1);0];
q=inputs(2:3);
q_d=inputs(4:5);
friction=[0.0;0.1].*q_d;

M=[alpha+2*beta*cos(q(2)),delta+beta*cos(q(2));delta+beta*cos(q(2)),delta]; %mass ↙
matrix
C=[-beta*sin(q(2))*q_d(2),-beta*sin(q(2))*(q_d(1)+q_d(2));beta*sin(q(2))*q_d(1),0]; %↙
Coriolos and centrifugal matrix
q_dd=inv(M)*(torque-C*q_d-friction); %calculation of angular acceleration of joints
result=q_dd;
end
%--------------------------
%forward kinematics function
%this function calculates the endeffector position for a given joint angles
function result = forward(inputs)
theta1=inputs(1);
theta2=inputs(2);
global L1 L2;
x=L1*cos(theta1)+L2*cos(theta1+theta2);
y=L1*sin(theta1)+L2*sin(theta1+theta2);
result=[x,y];
end
%--------------------------
%inverse kinematics function
%this function calculates the required joint angles for a required
%endeffector position in x,y axes
function result = inverse(inputs)
x=inputs(1);
y=inputs(2);
n=inputs(3);
global L1 L2;
alpha=atan(y/x);
beta=acos(((x^2)+(y^2)+(L1^2)-(L2^2))/(2*L1*sqrt((x^2)+(y^2))));
eta=((x^2)+(y^2)-(L1^2)-(L2^2))/(2*L1*L2);
```

```matlab
%decide if elbow up or down
if n==0
    theta1=alpha-beta;
    theta2=acos(eta);
end
    if n==1
        theta1=alpha+beta;
        theta2=-acos(eta);
    end
result=[theta1,theta2]';
end
%--------------------------
%trajectory creator function
function result=traj2(t)
l1=0.25; %first link length
l2=0.25; %second link length
count = length(t);
center = [0.2 0.1 0]; %center of the circle in x,y,z axes
radius = 0.15; %radius of the circle
theta = t*(2*pi/10); %the joint angles
points = center + radius*[cos(theta) sin(theta) zeros(size(theta))]; %position of↙
endeffector in x,y axes
result=points(1:2);
end
%--------------------------
%switcher
%when second link error is bigger than 0.43 rad the second controller is on
%only,otherwise,the first controller is on only
function result = switcher(inputs)
e1=inputs(1); %first link error
e2=inputs(2); %second link error
if abs(e2)>0.43
    o1=0;
    o2=e2;
else
    o2=0;
    o1=e1;
end
result=[o1;o2];
end
%--------------------------
%LQR control 5.3.2
global iz1 iz2
%state space representation of the system
A=[0,0,1,0;0,0,0,1;0,0,0,0;0,0,0,0];
in=[iz1+iz2,iz2;iz2,iz2];
B=[0,0;0,0;inv(in)];
C=[1,0,0,0;0,1,0,0];
D=0;
%Q and R matrices
Q=[1385,0,0,0;0,1485,0,0;0,0,485,0;0,0,0,585];
R=800;
%gain matrix
K=lqr(A,B,Q,R)
```

```matlab
%---------------------------
%actual disturbance estimator
function result = actual_disturbance(inputs)
global m1 m2 L1 r1 r2 ;
%inputs to the model
q=inputs(1:2);
q_dot=inputs(3:4);
q_dot_dot=inputs(5:6);
beta=m2*L1*r2;
%I=[iz1+iz2,iz2;iz2,iz2];
%nonlinear terms of mass matrix
M_prime=[m1*(r1^2)+m2*(((L1)^2)+r2^2)+2*beta*cos(q(2)),m2*(r2^2)+beta*cos(q(2));m2*↙
(r2^2)+beta*cos(q(2)),m2*(r2^2)];
%M=M_prime+I;
%nonlinear terms of Coriolos and centrifugal matrix
C=[-beta*sin(q(2))*q_dot(2),-beta*sin(q(2))*(q_dot(1)+q_dot(2));beta*sin(q(2))*q_dot↙
(1),0];
%calculation of actual disturbance
disturbance_nonlinear=M_prime*q_dot_dot+C*q_dot;
result=disturbance_nonlinear;
end
%---------------------------
%disturbance estimator(kalman filter)
I=[iz1+iz2 iz2;iz2 iz2];
xk_1_hat=zeros(6,1);
uk=zeros(2,1);
Pk_1=100*eye(6);
Qk_1=zeros(6,6);
zk=zeros(4,1);
NR=1000;
NQ=1000;
ek_new_1=zeros(4,1);
Rk_1=zeros(4);
Wk_1_bar=zeros(6,1);
function result=AKF(inputs)
global iz1 iz2 T xk_1_hat Pk_1 Qk_1  NR NQ ek_new_1 Rk_1 Wk_1_bar;
q=inputs(1:2);%
q_dot=inputs(3:4);%
zk=[q;q_dot];
%zk=q'
uk=[inputs(5);0];%
I=[iz1+iz2 iz2;iz2 iz2];%
I_inv=inv(I);%
A=[zeros(2) eye(2); zeros(2) zeros(2)];%
B=[zeros(2) ;I_inv ];%
Fd=[zeros(2); -1*I_inv];%
C=[eye(2) zeros(2)];%
Ae=[A Fd; zeros(2,4) zeros(2)];%
Be=[B;zeros(2)];%
Ce=[C zeros(2)];%
Ad=[eye(2) T*eye(2) zeros(2);zeros(2) eye(2) -T*I_inv*eye(2);zeros(2) zeros(2) eye↙
(2)];%
Bd=[zeros(2);T*I_inv;zeros(2)];%
H_new=[eye(4) zeros(4,2)];%
```

```matlab
Xk_hat_bar=Ad*xk_1_hat+Bd*uk;
Pk_bar=Ad*Pk_1*Ad'+Qk_1;
ek=zk-(H_new*Xk_hat_bar);
alfa1=(NR-1)/NR;
alfa2=(NQ-1)/NQ;
ek_new=alfa1*ek_new_1+(ek/NR);
Rk_dalta=1/(NR-1)*(ek-ek_new)*(ek-ek_new)'-(1/NR*(H_new*Pk_bar*H_new'));
Rk=alfa1*Rk_1+Rk_dalta;
K_k=Pk_bar*H_new'*inv(H_new*Pk_bar*H_new'+Rk);
xk_hat=Xk_hat_bar+K_k*ek;
%%xk_hat=xk_hat(6,:)'
Pk=(eye(6)-(K_k*H_new))*Pk_bar;
wk_hat=xk_hat-Xk_hat_bar;
Wk_bar=alfa2*Wk_1_bar+(wk_hat/NQ);
Qk_dalta=(1/NQ)*(Pk-(Ad*Pk*Ad'))+(1/(NQ-1))*(wk_hat-Wk_bar)*(wk_hat-Wk_bar)';
Qk=alfa2*Qk_1+Qk_dalta;
Qk_1=abs(diag(diag(Qk)));
Rk_1=abs(diag(diag(Rk)));
xk_1_hat=xk_hat;
ek_new_1=ek_new;
Pk_1=Pk;
Wk_1_bar=Wk_bar;
result=xk_hat(5:6);
%-------------------------
%-------------------------
%experimental approach 5.3.3
%-------------------------
%revolution reset(encoder 1 reset)
%this function resets the encoder's reading after exceeding the number of
%pulses(instead of channel Z in 3 channels encoders)
function y = fcn(u,n)
k=fix(n/400)
if (u>=400||u<=-400)
    y=u-400*k;

else
        y=u

    end;
%-------------------------
%revolution reset1(encoder 2 reset)
%this function resets the encoder's reading after exceeding the number of
%pulses(instead of channel Z in 3 channels encoders)
function y = fcn(u,n)
k=fix(n/300)
if (u>=300||u<=-300)
    y=u-300*k;

else
        y=u

    end;
%-------------------------
%vector to scaler
%this function converts the value to magnitude and direction
```

```matlab
function [cw,ccw,pwm] = fcn(u)
if u>=0
    ccw=1;
    cw=0;
else
     cw=1;
    ccw=0;
end
pwm=abs(u);
%-------------------------
%theta2 to theta1 function
%this function calculates the required first link angle step input by means
%of second link angle
function theta1 = theta2totheta1(time,theta2)
if time<2
    theta1=0; %the first two seconds the system is off
else
theta1 = 0.0038*(theta2^2) + 0.141*theta2 + 6.9758;
end
%-------------------------
%radius to theta2 function
% this function calculates the second link angle using cosine theorem
function theta2 = radius2theta2(time,radius)
link1=30;
link2=20;
if time<2
  theta2=0; %the first two seconds the system is off
else
      phai=acosd(((link1^2)+(link2^2)-(radius^2))/(2*link1*link2));
    theta2=180-phai;
end
%-------------------------
%selector function
%the first 6 seconds are for step input then it switches to ramp input
function angle = selector(time,step,ramp)
if time<6
    angle=step;
else
angle=ramp;
end
```