

Ashesi University

Using a PRP robot manipulator for 3d  
printer print bed replacement

Capstone Project

B.Sc. Electrical and Electronics  
Engineering

Samuel Shamo Jnr. Abbey

Supervisor: Dr. Stephen Kofi Armah

## Table of Contents

Abstract.....	1
Chapter 1: Introduction .....	1
1.1 Background.....	2
1.2 Motivation.....	2
1.3 Problem Definition.....	3
1.4 Proposed Solution .....	3
1.5 Scope of Work .....	3
1.6 Objectives .....	4
Chapter 2: Literature Review .....	5
2.1 History of 3d printers .....	5
2.2 Paper Review .....	5
2.3 Related Work .....	7
Chapter 3: Methodology .....	8
3.1 User Requirements.....	8
3.2 Design Process.....	8
Chapter 4: Implementation .....	10
4.1 Assumptions.....	10
4.2 Hardware Design and Implementation .....	10
4.2.1 Power source.....	11
4.2.2 Microcontroller .....	11
4.2.3 Actuators .....	11
4.2.4 Sensor.....	11
4.2.5 Hardware Circuitry .....	12
4.3 Mechanical Design and Implementation.....	13
4.3.2 Load and Distance Calculations.....	13
4.3.3 Kinematic Modeling .....	18
4.3.4 CAD Model and Assembly drawings .....	22
Chapter 5: Results and Discussion.....	28
Chapter 6: Conclusion.....	41
6.1 Limitations .....	41
6.2 Future Works .....	41

References.....	42
-----------------	----

## Table of Figures

Figure 4.2.1 Schematic diagram for PRP robot manipulator.....	12
Figure 4.3.1 Maximum load Calculation from Solidworks .....	14
Figure 4.3.2 Maximum load calculation using slicer (Ultimaker Cura) .....	15
Figure 4.3.3 FEA Static test on Robot Manipulator .....	16
Figure 4.3.4 Summary of Static Test .....	17
Figure 4.3.5 2d view of Model.....	19
Figure 4.3.6 Kinematic model used for Denavit-Hartenberg table.....	20
Figure 4.3.7 Isometric view of CAD model .....	22
Figure 4.4.1 Simscape Multibody Simulation .....	24
Figure 4.4.2 Simulink Subsystems for CAD Model .....	25
Figure 4.4.3 Circuit Diagram for motor.....	25
Figure 5.1 Model with proportional control .....	29
Figure 5.2 Proportional control simulation.....	29
Figure 5.3 Performance values of proportional control .....	30
Figure 5.4 P control response of Output compared to Unit Step input.....	31
Figure 5.5 PI control System .....	32
Figure 5.6 Input and Output Response compared.....	33
Figure 5.7 Performance Characteristics of PI controller .....	34
Figure 5.8 PID controller implementation. ....	35
Figure 5.9 Input against Output response comparison after PID control is implemented.....	35
Figure 5.10 Performance characteristics of PID implementation .....	36
Figure 5.11 PID controller added to main system .....	37

Figure 5.12 Path Planning applied to PID controller into the system.....	37
Figure 5.13 Rotating Path of the PRP manipulator.....	38
Figure 5.14 Base path of the PRP manipulator.....	39
Figure 5.15 Vertical Path of the PRP Manipulator .....	40

## Table of Tables

Table 4.3.1 Pugh Chart for material selection .....	13
Table 4.3.2 Modified Denavit Hartenberg Table.....	20
Table 4.4.1 Stepper Motor Parameters from Datasheet.....	23
Table 14.4.2 Motor Parameters.....	26

## List of Abbreviations

BOM – Bill of Materials

PRP – Prismatic Revolute Prismatic

FEA – Finite Element Analysis

MATLAB – Matrix Laboratory

PID – Proportional Integral Derivative

## Abstract

Ashesi University, within the past few years of starting its Engineering programme has deemed it a necessity to have 3d printers available to students to encourage rapid prototyping and testing. However, these 3d printers are not explored to their maximum capabilities due to lab hour restrictions. In this paper, a constructive mechatronic process is undertaken to design the best solution for students to use the 3d printer as often as possible regardless of lab hour restrictions. A PRP robot manipulator, after a series of design iterations is designed and a suitable controller is selected for the manipulator to function efficiently. The PRP manipulator design was solely based on simulations using the appropriate software for each section of the design process. Based on the results from the controllers, the PID controller was the most suitable controller for the PRP manipulator as it met the user requirements.

## Chapter 1: Introduction

Robot manipulators are robots designed with linkages (commonly known as arms) joined together using actuators. The function of the manipulator determines the suitable actuator to be used; Examples of these actuators are servo motors, continuous rotation servo motors, stepper motors, linear actuators, and dc motors. Based on specific design requirements and expectations, the actuator used is selected accordingly. Some robot manipulators operate through telemetry (manual or direct control) while others are pre-programmed and autonomous. Autonomous robot manipulators can make decisions on their own, based on the range of capabilities determined by the controller and the degrees of freedom; for most cases, the decisions are event-driven (waiting for an incident and reacting to it).

## 1.1 Background

Robot manipulators are categorised based on different configurations: Gantry robots, also known as cartesian/rectilinear robots; they are robots with linear joints which are mounted overhead. Cylindrical robots have linear joints that connect to a rotating base joint. Polar robots have both linear and rotary joints. Jointed-arm robots, where the arms connect as joints with twisting capabilities and links, are connected using rotary joints. The evolution of autonomous Robot arm manipulators has had a significant impact on the modernisation of the manufacturing industry; thus, robots are present at every stage of the manufacturing process. The synthesis of two factors mobility and manipulation take capacity to much wider ranges of tasks than a fixed manipulator or mobile robot [1]. So, this combines both advantages from mobile robot and manipulator. 3d printers, also rapidly growing field has altered the additive manufacturing world. These machines, with the help of G-code, can extrude materials using a concept known as Fused Deposition Modelling(FDM) into irregular shapes that cannot be obtained through casting or moulding.

## 1.2 Motivation

After spending almost four years in Ashesi University and having the privilege of attending summer school designed for the engineering class, I have pondered over a few problems my fellow engineering students together with me, faced during that period. However, upon further analysis, there are still some which are in existence. One of the major problems is, the inability of students to access the 3d printers available in the school after specific hours of the day. This problem has a negligible impact when there is less work to be done with the printers. The problem, however, aggravates when different groups require a lot of printing done for project assignments. Students wait long periods to initiate the next print as they take turns, and unfortunately on some occasions get caught up with the closing time, hence are unable to print. Since these printers are not

networked, the remote control is impossible; therefore, we lose approximately eleven hours of possible print time in a day, and 55hours in a week (working days only). Developing a solution to solve this problem is my primary motivation; hence, this capstone.

### 1.3 Problem Definition

Users of 3d printers in Ashesi University, are unable to utilize the available 3d printers during certain hours of the day due to lab hour restrictions. This affects productivity and efficiency during project submission periods as teams are compelled to visit the lab during class hours and lose a lot of possible print hours. There is a need for an automated print bed replacement system to utilise the remaining hours when the lab is closed.

### 1.4 Proposed Solution

The proposed solution for this project an autonomous robot manipulator with three degrees of freedom (three linkages) with an end effector. From observation, the environment of the Ultimaker in the Ashesi Fabrication lab, there is limited space for operation and mobility. A variety of models, with different shapes, sizes and weights (based on infill) are printed using 3d printers. Hence, designing a manipulator to pick the printed object, would require a lot of image recognition, machine learning and a corresponding control system to properly function.

### 1.5 Scope of Work

To do away with this extremely large scope and arrive at the same results, replacing the entire print bed is the method to go with. The solution would consist of:

- Detecting when the 3d printed completes and print sequence.
- Removing the print bed together with the printed model from the 3d printer.
- Placing the print bed and the printed model on a safe platform.



- Picking an empty print bed and placing in the 3d printer workspace.

The end effector of the robot manipulator is a suction pad. The suction pad will be arranged in pairs to facilitate stability. The robot's operations are event-driven, and it will not have the remote-control capabilities as the manipulator will have more than one user (Ashesi students and faculty).

## 1.6 Objectives

The main objective of this project is to replace the print bed of a 3d printer. The specific objectives in this project are :

- Determining the maximum load possible to be lifted.
- Designing the circuit used to control the various electrical components in the manipulator.
- Testing the circuit using an appropriate simulation software.
- Designing the CAD model for the manipulator.
- Implementing the CAD model in MATLAB for further simulation and control.
- Testing various controllers on the manipulator and comparing the results.
- Selecting the suitable design based on requirements met and critical evaluation.

The user requirements to be achieved are:

- 0% Maximum overshoot
- Settling time less than 1 second
- Rise Time less than 1 second
- Stable System

## Chapter 2: Literature Review

In this chapter, a section of related work is discussed, and scientific papers are used to draw insights on the various design decisions and requirements.

### 2.1 History of 3d printers

The 3D printer is a three-dimensional prototyping machine which fabricates desired shape by layer over layer material deposition.[2]. Although it is not an entirely new technology, its exploration has increased within the past decade and research institutions, educational institutions and hobbyists are constantly exploring its capabilities to the full. Students are encouraged to have a lot of iterations during prototyping, and 3d facilitates this process of rapid prototyping. The cost of 3d printers covers a wide range depending on complexity, size and functionality. Some 3d printers output finished products for consumers and in no time, 3d printing will be an all-round manufacturing process as it is used a variety of fields. The two main distinctions between 3d printing and the other common manufacturing processes are the amount of waste produced and the complex geometry than it can form. Manufacturing methods which apply moulds require many moulds to produce different parts of a piece with complex geometry, whereas 3d printing can seamlessly produce that piece as one single entity.

### 2.2 Paper Review

This paper deals with designing event triggered nonlinear controllers for robot manipulators. To track the desired path trajectory of manipulator system, sliding mode control is studied and event triggering mechanism is developed. The objective of event triggered control is to ensure efficient utilization of resources by minimizing the control updates.

ET-SMC (Event Triggered Sliding Mode Controller) control of robust manipulator. ET-CT represents the (Even Triggered Computed torque) controller. Mechanical systems having complex non-linear dynamics with time varying uncertainties. Sliding mode control is used for systems with complex nonlinear dynamics with time carrying uncertainties. It drives the system states onto the sliding surface which is a defined surface in state space. It has been used as a robust approach structure of manipulator dynamics. The controller does not necessarily require the manipulator to be in the same vicinity. Hence the communication network, such as the network control system is required. Minimum data is sent to utilize the frequency bandwidth efficiently. This might come at a cost of the performance of the manipulator. Even triggered control can ensure trade-off between performance and bandwidth by sampling data only when the event occurs. Even triggered control is commonly used in wireless networks, manufacturing assembly, robotics and others. It is used specifically in the assembly of micro electric products, packing, welding and complex assembly.

A few uncertainties mentioned in this paper, which occurs in robot manipulators are:

- Structural – Payload variations and torque constants inaccuracies.
- Un-structural – non-linear friction and external disturbances.

It is for these few reasons why we need robust controllers. The paper made use of a 2-link planar robot arm. The use of these controllers ensures that the robot follows a prescribed desired trajectory. The two controllers were tested based on the inter-execution time and the control efforts. The dynamic modelling of the robot manipulator consists of a main system. Under the main system, there are two branches: the control section and the robot manipulator section. The control section consists of the event triggering mechanism, the communications network, Zero Order Hold and the controller.

Functionality: The position of the joints of the manipulator is obtained by the controller. Position information is sampled, and the error is calculated. The input is updated if the rule is violated, and it is sent back to the manipulator. When there is no violation, the input remains the same.

The simulation software used in this paper was MATLAB. From the simulations, the error in ET-SMC was lower than that of ET-CT [3].

The pros of this paper are the fact that they explored two controllers to select from based on the results from the simulation. However, I believe that they could have added a variety of controllers to have a larger sample size to work with. The goal of my capstone is to design an autonomous robot manipulator for print bed replacement. For this reason, remote control is not part of the iterations of designs I have considered for reasons such as unreliable network, network failure due to noise in the surrounding and other factors. Hence, a future work from this paper will incorporate an autonomous feature without network requirements. One thing I will adopt from the approach used is the type of controllers incorporated in the simulation.

## 2.3 Related Work

A few companies, outside Ghana who use 3d printers for manufacturing their products have implemented mechanisms for replacing print jobs from a 3d printer. This is to improve efficiency in the workplace as they have a print farm and replacing each print bed manually is not the best solution to the problem they face. For this reason, they have incorporated a cylindrical autonomous robot manipulator that replace print jobs in their print farm. The use of autonomous cylindrical robots was the best option for them since the printers they were using were without protective frames, hence the print beds could be accessed from any angle.

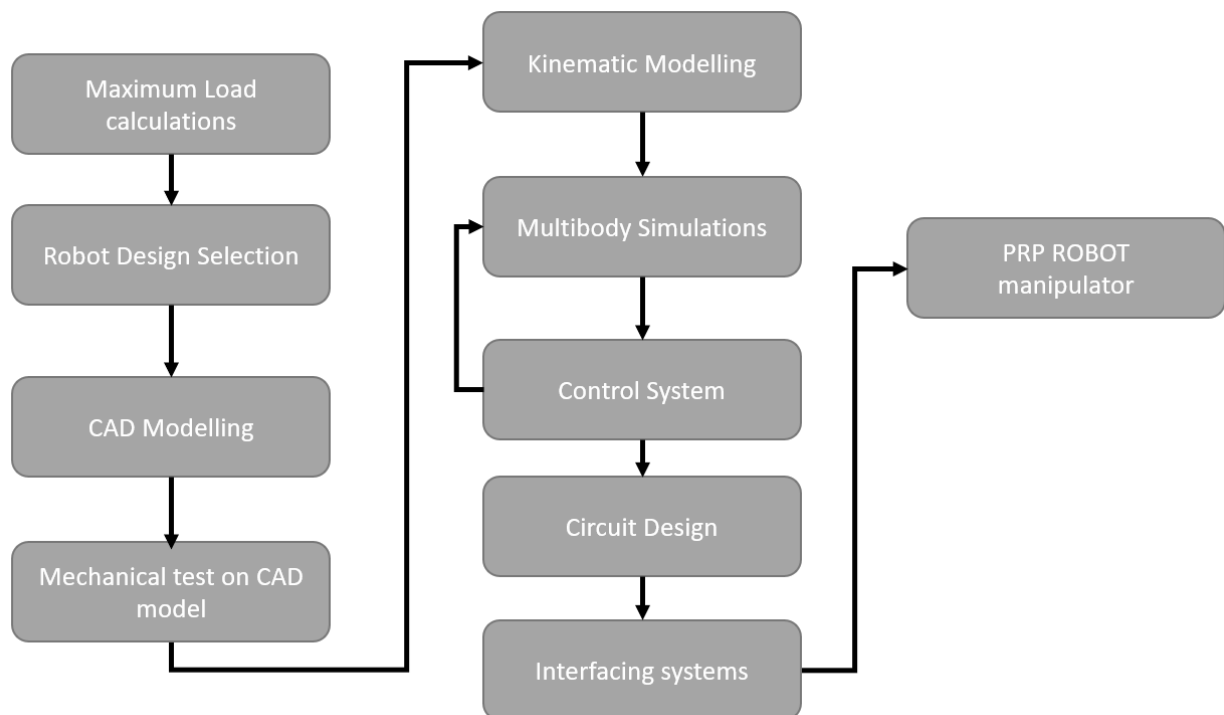
## Chapter 3: Methodology

This chapter describes the various steps that would be taken to design the manipulator.

### 3.1 User Requirements

As this solution is targeted at solving a problem faced by students in Ashesi University, I had to perform an informal research with students who frequently use the 3d printers to ensure that it was not a personal solution. From the few interviews conducted, it was evident that my mates had similar challenges with regards to this topic. I gathered some common requirements they expected from the manipulator; The manipulator should cover a wide section of the table it is mounted on; the manipulator should not inhibit free movement in the lab; It should not be capable of being remote controlled as students might take advantage of it. It should not require training of students.

### 3.2 Design Process



The first step to be undertaken is the maximum load calculations. The maximum load calculation refers to the process of determining the maximum load possible to be lifted by the manipulator. Every 3d printer has its maximum volume it can print hence, we can easily derive the maximum load. Then next step will be selecting the robot design. This is an iterative process of validating robot designs and justifying why a particular design suits the task at hand. After the robot selections process is over, the CAD modelling of the selected robot is done using a suitable CAD software. The mechanical test on the CAD model is then performed by applying the maximum load on the model to determine if it meets the mechanical requirements.

## Chapter 4: Implementation

The requirements for this project are that all three subsystems should have be underdamped and without overshoot. The software used to facilitate modelling, simulation and results analysis are Matrix Laboratory (MATLAB) Simulink and Simscape Multibody, Solidworks and Arduino. Solidworks was used to design the CAD model, perform necessary mechanical tests on it and link the CAD model with MATLAB for further simulation and analysis. MATLAB was used to implement the control system, simulate, and arrive at results and Arduino will be used to implement the tuned control system derived from MATLAB.

### 4.1 Assumptions

In this project, the focus is to replace the print bed of the 3d printer and as such, a few assumptions had to be made to mimic a controlled environment. These assumptions were that:

- the mechanism used in securing the print bed firmly on the heat plate is opened.
- the printer is faultless (completes initiated prints without any malfunction)
- the power supply for the printer and the manipulator are supported by UPS and Solar.
- The phase voltage from the house is 240V.

### 4.2 Hardware Design and Implementation

The proposed design for the manipulator required a circuit implementation to power and control the actuators and sensors used. Three factors considered in the selection of the components were cost, compatibility with Arduino and MATLAB and functionality. However, there had to be a fair balance between cost and functionality. The electrical components selected were ATMEGA 328P microcontroller, LM7805 and LM7812 voltage regulator, M1554A 230V to 12V transformer,

three NEMA 17 stepper motors, three a4988 stepper motor drivers, three 1uf capacitors and a Germanium diode.

#### 4.2.1 Power source

As part of the manipulator design, the main source of power for all electrical components will be the 230V 13A sockets available in the lab. As this voltage is too high for the selected components, it requires a voltage step-down to prevent damage of components. The M1554A step-down transformer was selected because it was the cheapest available step-down transformer online, at \$14 which was equivalent to Ghc 81.00.

#### 4.2.2 Microcontroller

The criteria used in selecting the microcontroller is the accessibility, cost, ease of use and the power consumption. The ATMEGA 328P is one of the most used microcontrollers as it installed on the Arduino Uno. For this reason, it is priced at Ghc 15.00.

#### 4.2.3 Actuators

Based on the functionality of the manipulator, one main criterion considered is a high torque at low speeds. Since the manipulator will be placed in an environment with a lot of human interaction, the speed during motion should be as low as possible to prevent avoidable accidents and still accomplish the task. The NEMA 17 stepper motor was selected for this reason.

#### 4.2.4 Sensor

The manipulator is an event driven manipulator, hence does not require all day power supply. For this reason, a magnetic reed switch is selected to power the manipulator only when the print sequence ends.



## 4.2.5 Hardware Circuitry

Figure 4.1 illustrates the complete circuit schematic for the proposed system.

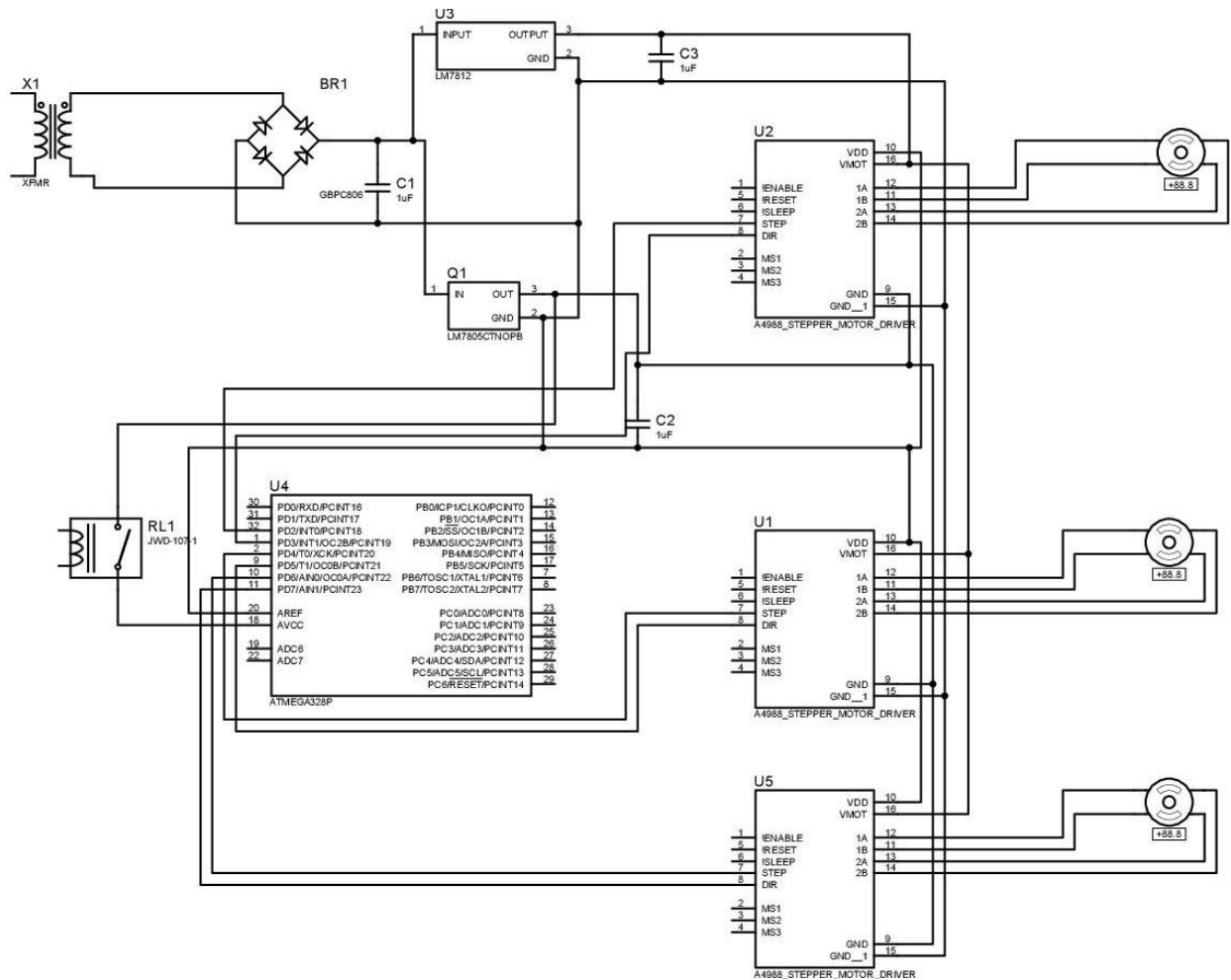


Figure 4.2.1 Schematic diagram for PRP robot manipulator

Moving from the top-left corner of the circuit diagram, the voltage received from the power source, 230V is stepped down to 12V AC using the M1554A step-down transformer. The 12V AC is passed through a full-wave rectification circuit to output a rippled 12 DC output. The ripples are smoothened using a 1uF capacitor connect parallel to the rectifier circuit. The LM7805 and LM7812 are connected to the terminals of the capacitor output for 5V and 12V DC supply, respectively. The 12 volts is used to power the NEMA 17 stepper motors, and the 5 volts is used

to power the stepper motor drivers, and the microcontroller. The ATMEGA 328P is boot-loaded to use the Arduino IDE to program the microcontroller.

### 4.3 Mechanical Design and Implementation

In this section, the various mechanical calculations and decisions are performed.

Table 4.3.1 Pugh Chart for material selection

	<b>Baseline</b>	<b>Weight</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>
<b>Criteria</b>	Steel		PLA/ABS	Wood	Carbon Fibre	Styrofoam
<b>Cost</b>	0	2	+2	+1	+3	+4
<b>Durability</b>	0	4	+2	+1	-3	+3
<b>Strength</b>	0	2	+2	+1	+3	-2
<b>Weight</b>	0	2	-2	-1	0	-3
<b>Appearance</b>	0	1	0	-1	+3	-1
<b>Total</b>			+12	+5	+3	+9

From the table above, the Pugh chart was used to select the material suitable to building the manipulator based on the cost, durability, strength, weight, and overall appearance. These materials are some of the most common materials available in the market.

#### 4.3.2 Load and Distance Calculations

The most important factor to consider mechanically, in the design of the manipulator is the maximum load it should carry. Since we are focusing on 3d printed model, Solidworks and

Ultimaker Cura were used to perform some analysis. The 3d Printer used for this analysis is the Ultimaker 2 extended +. It has a build volume of 223 x 223 x 305 mm and this corresponds to a maximum weight of 15.470 kg using the ABS material as indicated in Figure 4.2.

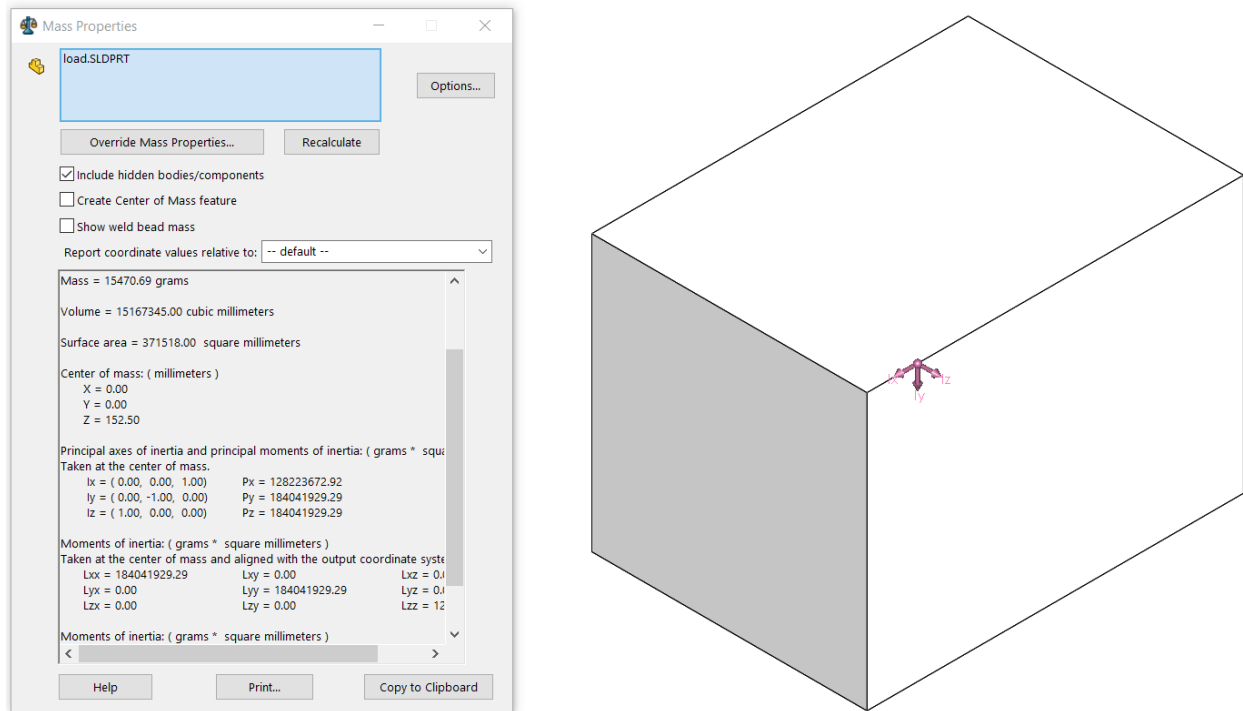


Figure 4.3.1 Maximum load Calculation from Solidworks

However, the slicing software determines the actual maximum build volume by the 3d printer. From analysis, the practical build volume is 190 x 188.8 x 290mm and this corresponds to a maximum weight of 11.442kg with 100% infill using the ABS material as indicated in Figure 4.3.

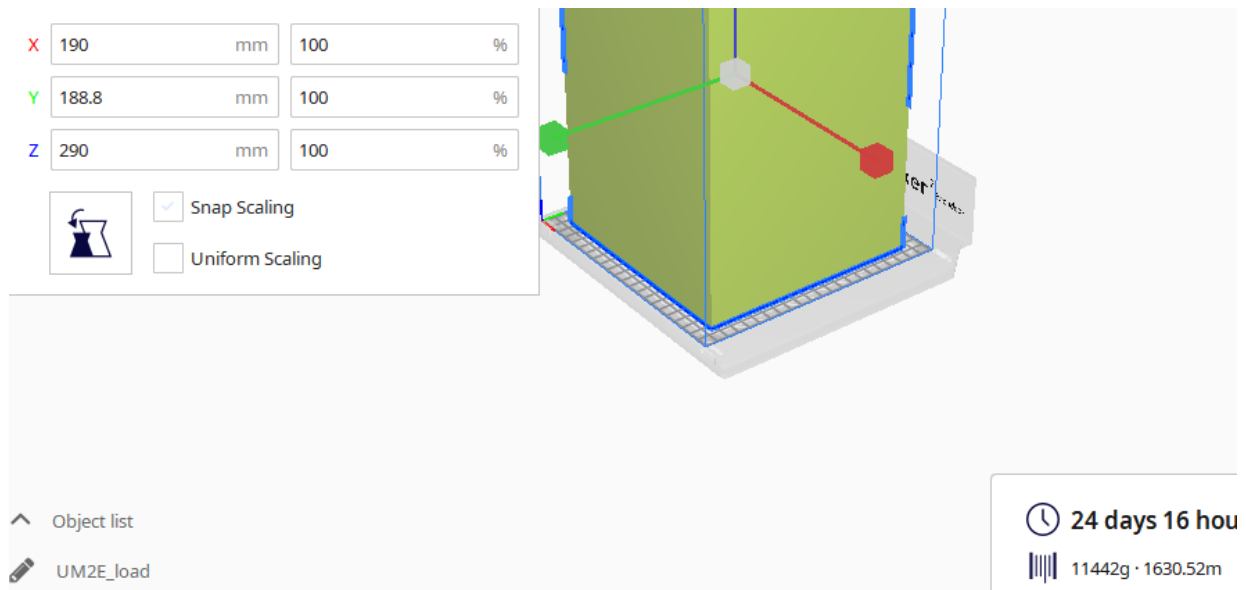


Figure 4.3.2 Maximum load calculation using slicer (Ultimaker Cura)

For safety reasons and taking the weight of the print bed into consideration, the maximum load to be carried is 12 kg. The four pneumatic suction caps will be used to pick and release the load. As such, static tests are performed on the manipulator, using Solidworks. The force applied on each suction cap is derived from the following calculations:

$$\text{Maximum load} = 12\text{kg}$$

$$\text{Gravitational force acting on body} = 10\text{ms}^{-2}$$

$$\text{Force applied per suction pad} = \frac{\text{Maximum Load}}{\text{number of suction pads}} * \text{Gravitational force} \text{ ---equation (4.1)}$$

$$= \frac{12\text{kg} * 10\text{ms}^{-2}}{4}$$

$$= 30\text{N per suction cup}$$

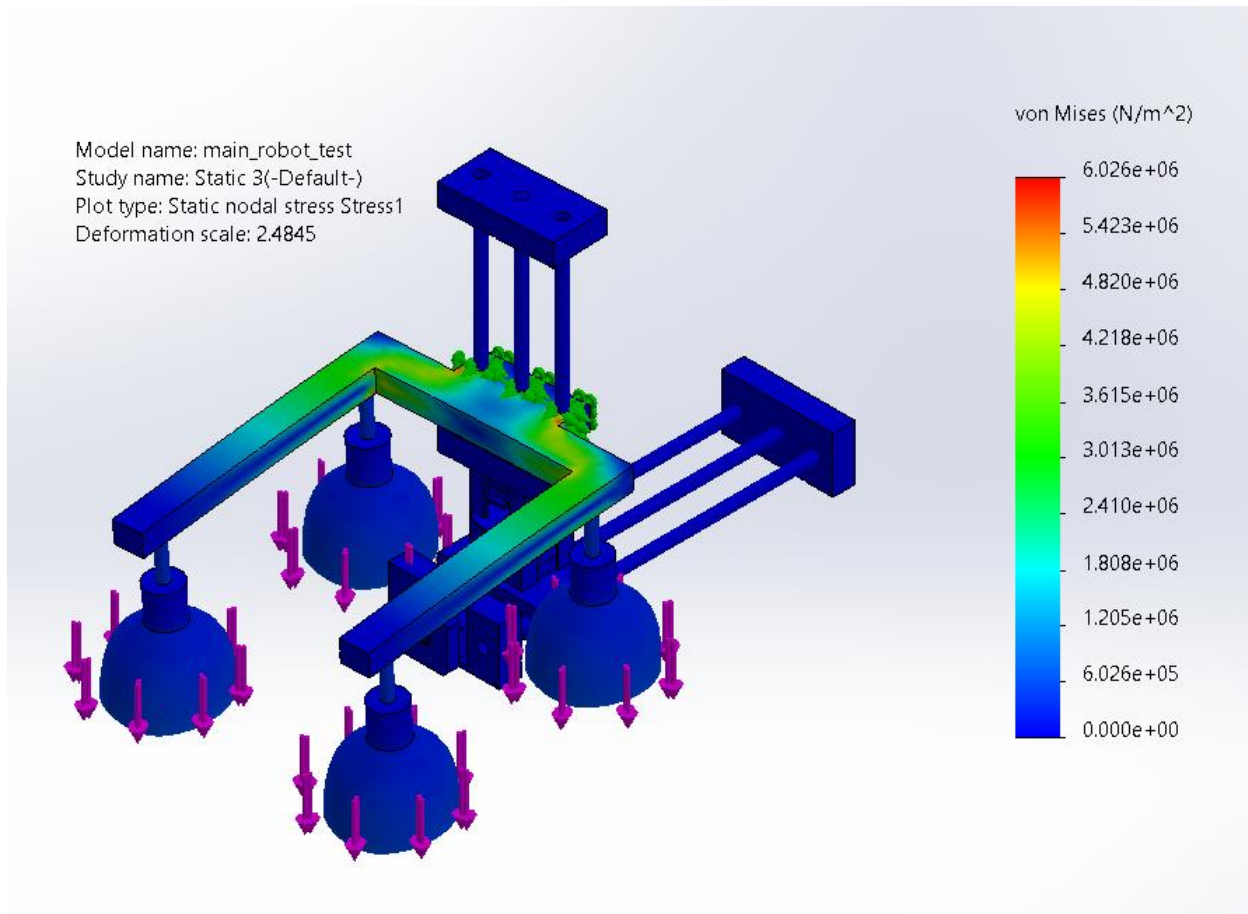


Figure 4.3.3 FEA Static test on Robot Manipulator

Figure 4.4 indicates the static test performed on the manipulator by applying a force of 30N derived from equation 4.1. From the figure, the maximum nodal stress indicated from the von misses plot is  $6.026 \times 10^6$ . This is smaller than the Tensile strength of each suction pad. When the static test is performed on a body and the Tensile strength is greater than the maximum stress on the body, it means that the body can withstand the load and is structurally sound. However, when the Yield strength is less than the maximum stress on the body, it means that the body is incapable of carrying the maximum load used for testing. From our results, the design of the PRP robot manipulator can lift the maximum 12kg load without any mechanical difficulties. A summary of the test performed is indicated in Figure 4.5.


	<b>Name:</b>	Rubber
	<b>Model type:</b>	Linear Elastic Isotropic
	<b>Default failure criterion:</b>	Unknown
	<b>Yield strength:</b>	9.23737e+06 N/m <sup>2</sup>
	<b>Tensile strength:</b>	1.37871e+07 N/m <sup>2</sup>
	<b>Elastic modulus:</b>	6.1e+06 N/m <sup>2</sup>
	<b>Poisson's ratio:</b>	0.49
	<b>Mass density:</b>	1,000 kg/m <sup>3</sup>
	<b>Shear modulus:</b>	2.9e+06 N/m <sup>2</sup>
	<b>Thermal expansion coefficient:</b>	0.00067 /Kelvin

Figure 4.3.4 Summary of Static Test

As part of the design, the lead screw is the mechanism used to convert the rotational motion derived from the motor into linear motion, and a rotational force into a linear force. Lead is the axial distance the screw travels in after a complete revolution. The linear distance travelled depends on the type of thread used. If the lead screw has one continuous thread, the lead would be the distance between two successive crests. For the PRP manipulator, a lead screw with a lead of 1.5 mm/rev was used. The linear distance can be derived using the Equation 4.1

$$\text{Linear distance} = \text{Lead Distance} \times \text{Rotation} \text{-----eqn(4.1)}$$

The base leadscrew and the end effector lead screw have different lengths hence, calculations must be performed for each lead screw. The length of the base lead screw is 280mm. From equation 4.1,

$$280\text{mm} = 1.5\text{mm/rev} \times x/360 \text{ rev}$$

$$x = 67200 \text{ revolutions}$$

The end effector lead screw has a length of 150mm. Using equation 1, the number of revolutions required to calculate a linear distance of 150mm will be:

$$150mm = 1.5mm/rev * x/360 rev$$

$$x = 36000 \text{ revolutions}$$

The NEMA 17 stepper motor has a step angle of  $1.8^0$ . This means that to complete one revolution the shaft of the motor must take  $\left(\frac{360}{1.8}\right) = 200 \text{ steps}$ . From this, the number of steps required of the robot to take to move a linear distance of 280mm and 150mm are  $(67200 * 200) = 13440000 \text{ steps}$  and  $(36000 * 200) = 7200000 \text{ steps}$ .

#### 4.3.3 Kinematic Modeling

A PRP robot manipulator is basically a manipulator with a prismatic joint (P), a revolute joint (R) and another prismatic joint(P). A prismatic joint is used to describe a joint where there is motion along an axis, thus translation. A revolute joint however is used to describe a joint where there is motion about an axis, thus rotation.

There are two methods for controlling robots in terms of positioning and orientation, and these are forward kinematics or inverse kinematics. Forward kinematics is used when we need to find the position and orientation of the end-effector from the given joint angles. On the other hand, inverse kinematics is used when we need to find the joint angles for a given position of the end-effector. Based on the functionality of the manipulator, inverse kinematics is the best method to use since the desired positions of the end effector are known. However, both equations were derived using the Denavit-Hartenberg theorem. This theorem summarizes the relationships between the various joints of your kinematic model and the reference frames based on four parameters. The relations are summarized in Table 4.1 .

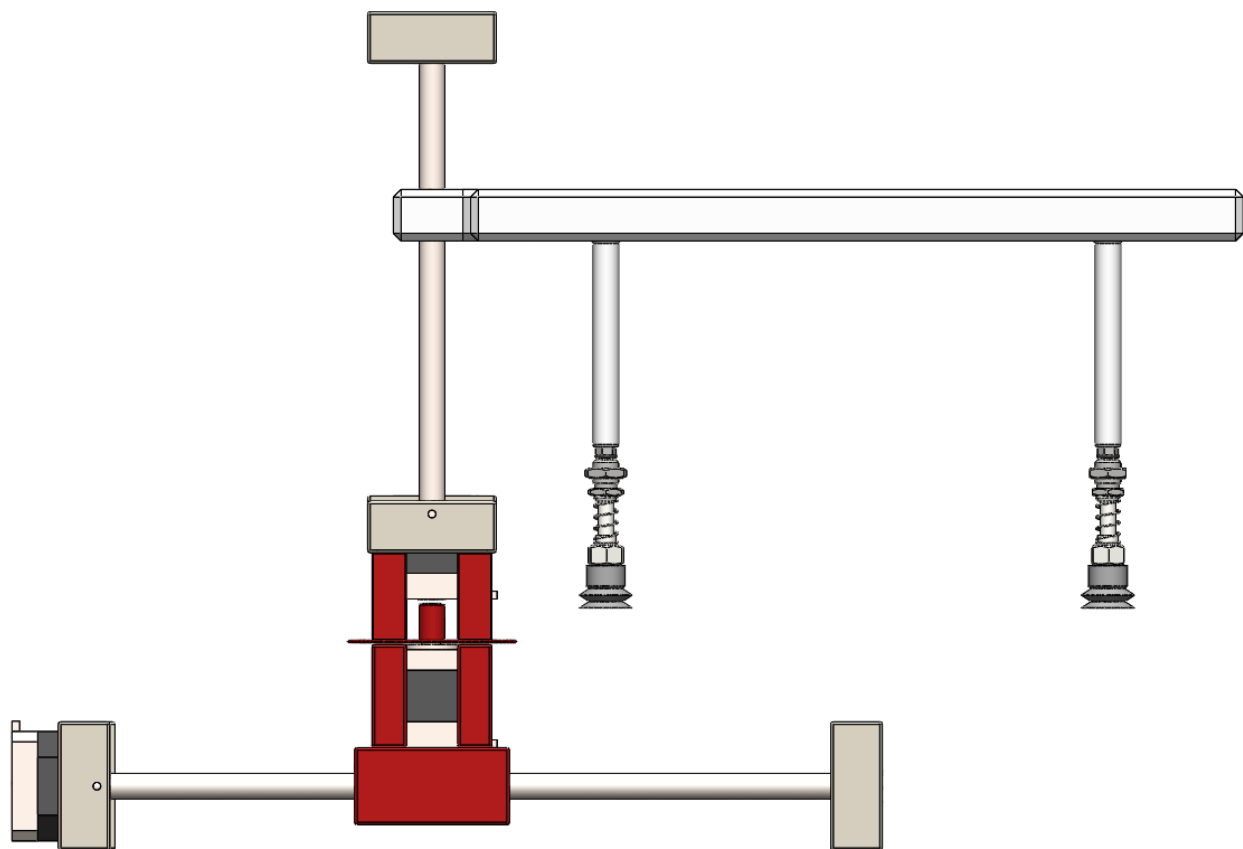


Figure 4.3.5 2d view of Model



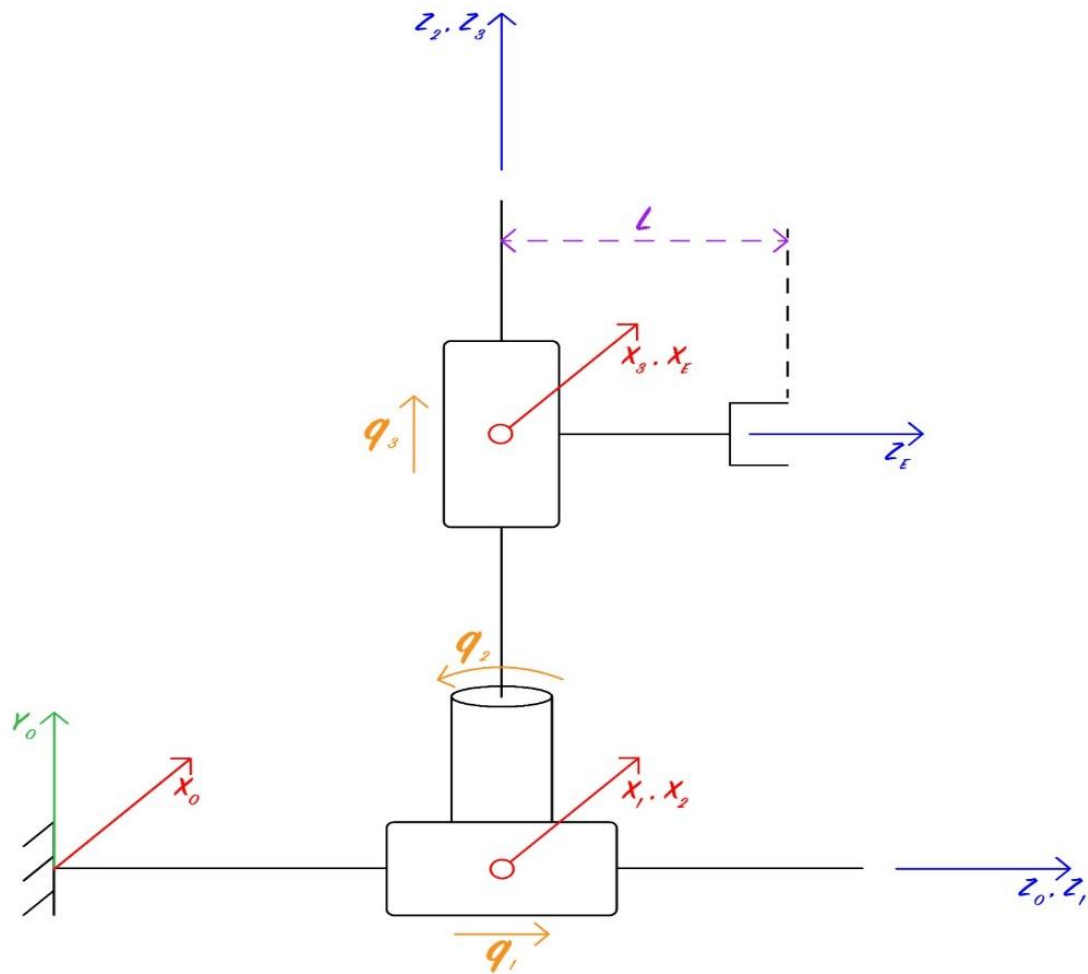


Figure 4.3.6 Kinematic model used for Denavit-Hartenberg table

Table 4.3.2 Modified Denavit Hartenberg Table

Modified Denavit Hartenberg Table				
Joint	$\Theta_i$	$\alpha_i$	$R_i$	$d_i$
0	0	0	0	0
1	0	0	0	$q_1$

2	$q_2$	$\frac{-\pi}{2}$	0	0
3	0	0	0	$q_3$
E	0	$\frac{\pi}{2}$	0	1

From the Denavit-Hartenberg table,  $\Theta$  represents the rotation about  $Z_{n-1}$  to get  $X_{n-1}$  to match  $X_n$ ,  $\alpha$  represents the rotation about  $X_n$  to get  $Z_{n-1}$  to match  $Z_n$ ,  $r$  represents the distance between the centre of the two frames along  $X_n$  and  $d$  represents the distance between the centres of the two frames along  $Z_{n-1}$ . Joint 0 represents the reference frame; joint 1 represents the prismatic joint connected to the reference frame; joint 2 represents the revolute joint on top of the base prismatic joint; joint 3 represents the prismatic joint attached to joint 2 and joint E represents the end effector frame attached to joint 3. The modified Denavit-Hartenberg table is used to derive the equations for both forward and inverse kinematics.

$$(q_1, q_2, q_3) \rightarrow (x, y, z)$$

$$x = l \sin(q_2) \quad (1)$$

$$y = q_3 \quad (2)$$

$$z = q_1 + l \cos(q_2) \quad (3)$$

Equations 1, 2 and 3 represent the forward kinematic equations of the PRP robot manipulator. The inverse kinematic equations can be derived from forward kinematics by making the joint distance/angles the subjects of the equations.

$$(x, y, z) \rightarrow (q_1, q_2, q_3)$$

$$q_1 = z + l \cos(q_2)$$

$$q_2 = \sin^{-1} \left( \frac{x}{l} \right)$$

$$q_3 = y$$

#### 4.3.4 CAD Model and Assembly drawings

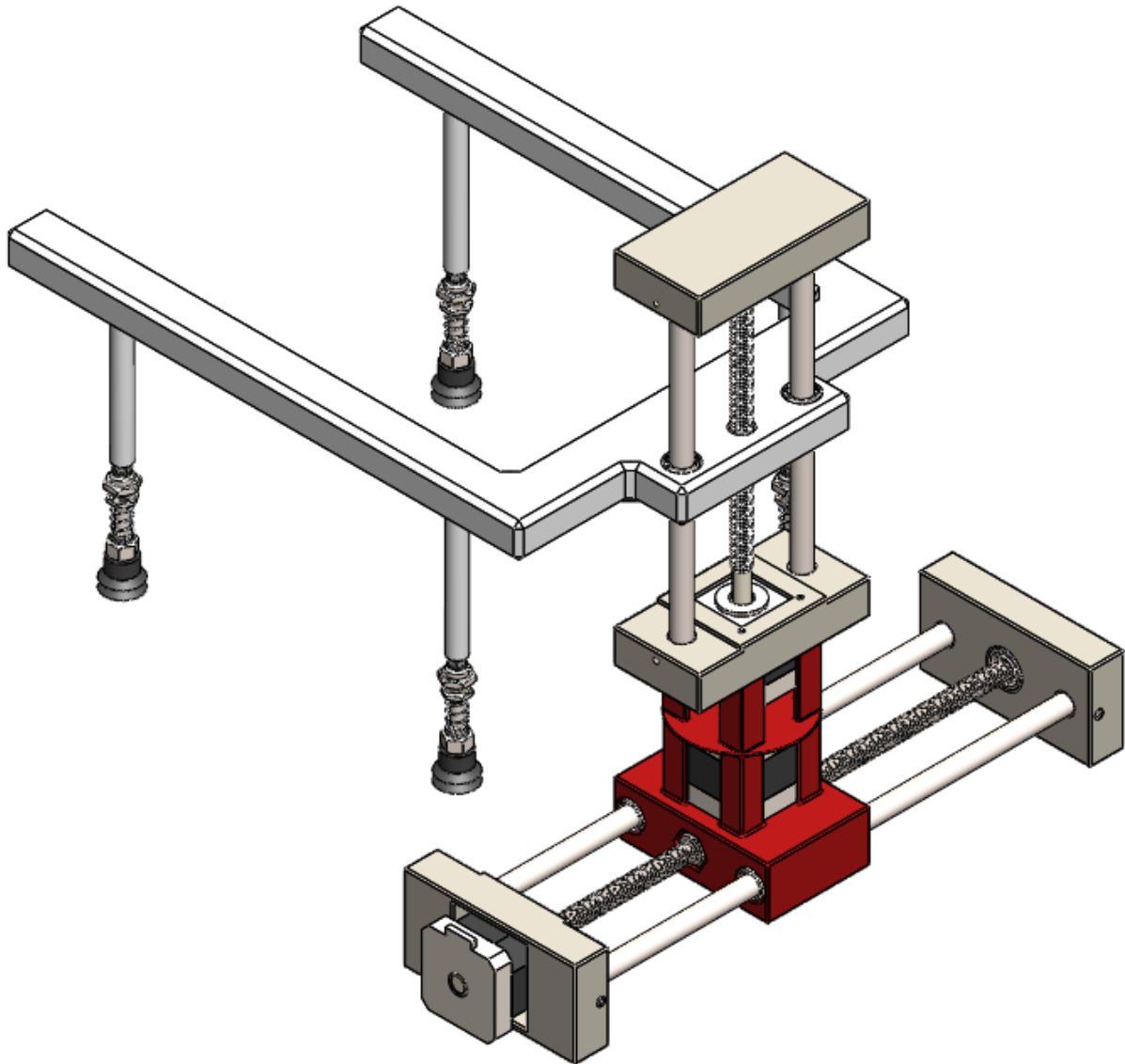


Figure 4.3.7 Isometric view of CAD model

#### 4.4 Software Design and Implementation

The software simulation was facilitated using MATLAB software. The various physical parameters of the model based on the data sheet were inputted in the software for simulations purposes. The Stepper Motor parameters summarized in Table 4.2 were entered into a MATLAB Script and registered in the workspace.

Table 4.4.1 Stepper Motor Parameters from Datasheet

Parameter	Variable Name	Value
Winding Inductance	Wi	48e-3
Winding Resistance	Wr	32.6
Step Angle	Sa	1.8
Maximum Flux Linkage	Mfl	1.8
Maximum Detent Torque	Mdt	0.016
Total Inertia	Ti	3.5
Total Friction	Tf	39.2266

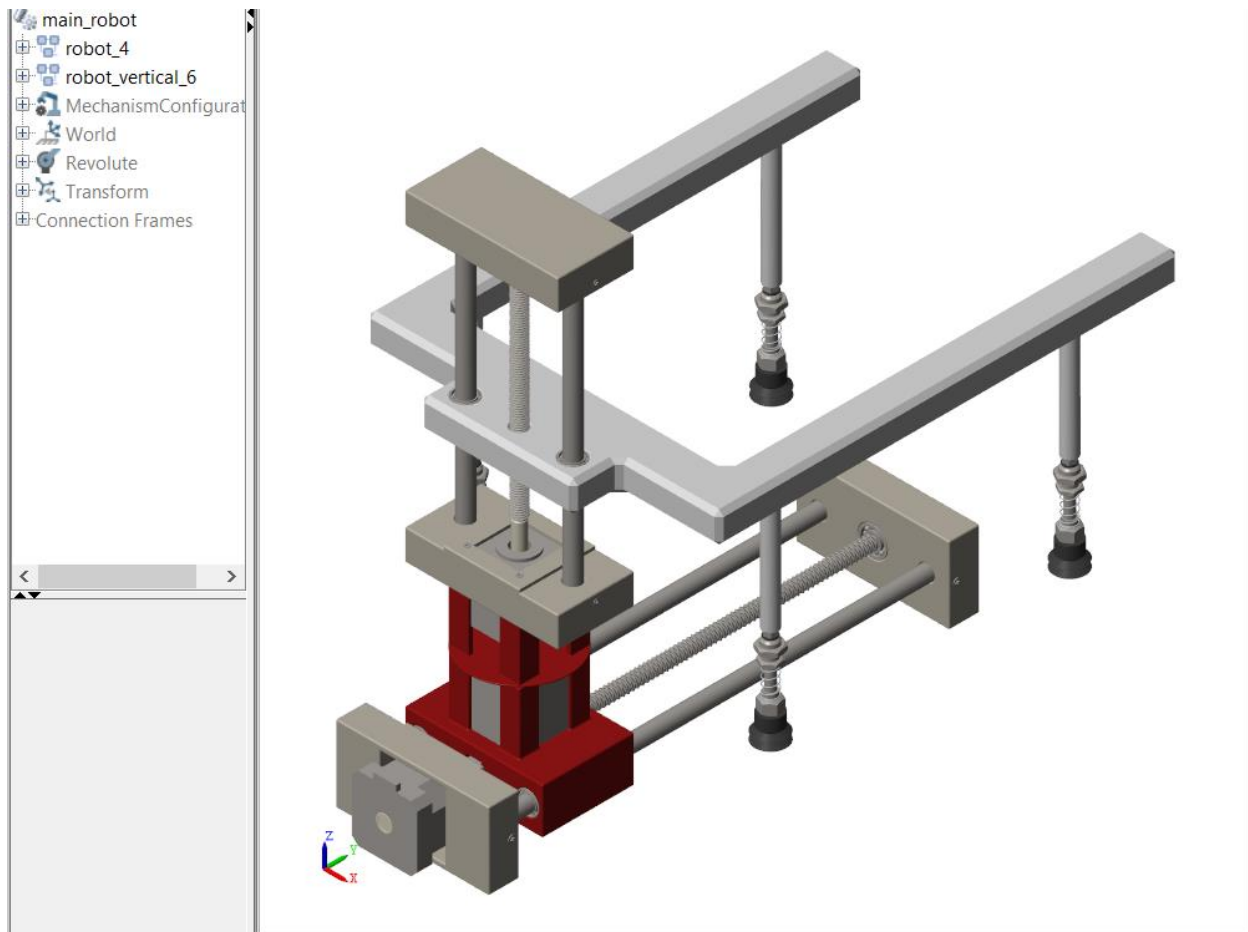


Figure 4.4.1 Simscape Multibody Simulation

After exporting the CAD model to MATLAB, an initial simulation was made on the model for a period of 10 seconds, to observe the natural response of the system in MATLAB. The gravitational force in MATLAB by default acts along the y-axis. Hence, in the simulation, all parts on the y-axis observed motion. The gravitational force was then changed in the mechanics configuration toolbox to act along the z- axis. The model was simulated again to observe the response of the assembly. From observation, the parts connected along the z-axis fell under gravity leaving the other parts connected to the x-axis or y-axis stationary.

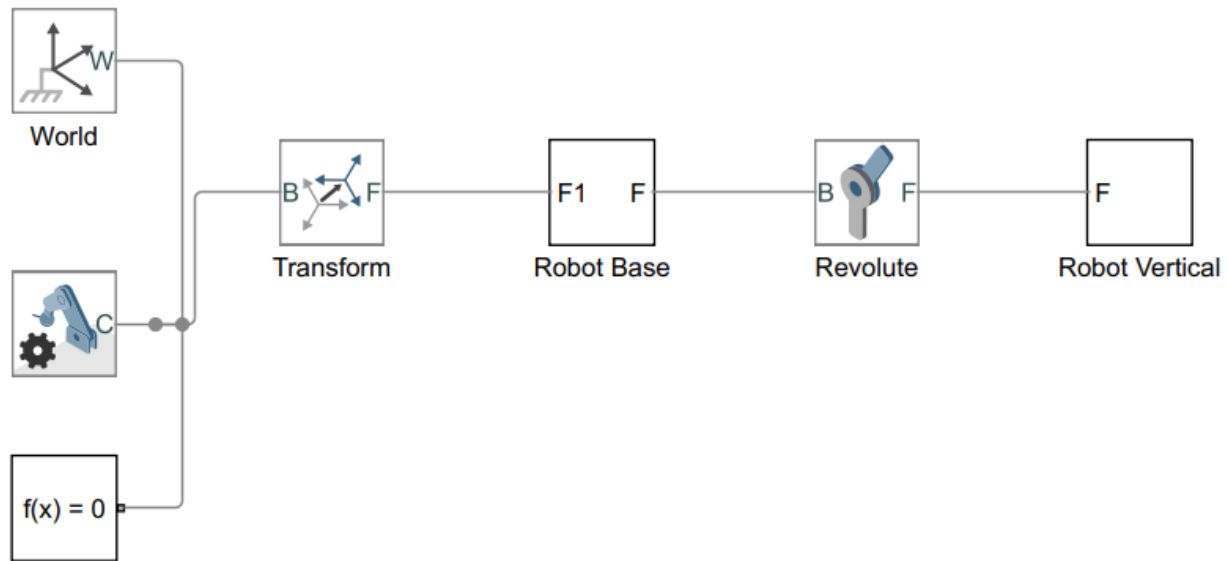


Figure 4.4.2 Simulink Subsystems for CAD Model

Before further simulation can continue the mathematical model of a motor must be derived.

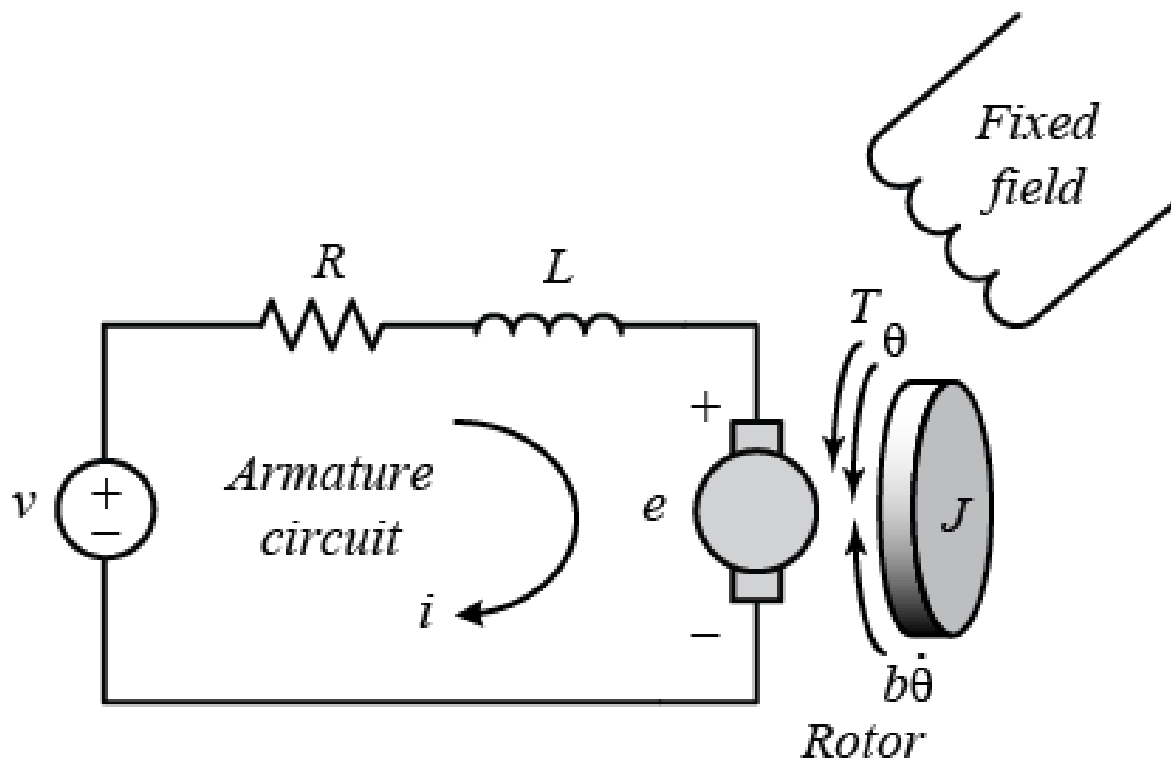


Figure 4.4.3 Circuit Diagram for motor

The electrical model of the motor consists of a voltage source, a resistor, and inductor, a magnetic field since it is brushless, and the motor shaft that converts the electrical signal into rotations. This means that the input for of model is voltage which is applied to the armature of the motor and the output is the equivalent rotational speed. The physical parameters were derived from a datasheet and are listed in Table 4.4.2

*Table 14.4.2 Motor Parameters*

Parameter	Value
Moment of Inertia (J)	0.01 kg m <sup>2</sup>
Motor viscous friction constant (b)	0.01 Nms
Electromotive force constant (Ke)	0.01 V/rad/sec
Motor Torque constant (Kt)	0.01 N.m/ Amp
Electric resistance (R)	1 Ohm
Electric Inductance (L)	0.5 H

To derive the transfer function of the motor, a few assumptions must be made; We assume that the magnetic field present in the motor during operation is constant. Since the motor torque is directly proportional to the magnetic field present in the motor and the current flowing through the armature of the motor, the assumption made will translate to the torque of the motor being proportional to the current flowing through the armature of the circuit.

$$T = K_t i$$

Where K is the constant of proportionality and i, is the armature current.

The back emf, which is an electromagnetic force appearing in an inductive circuit in such a direction as to oppose any change of current in the circuit[4] is directly proportional to the angular velocity of the shaft.

$e = K_e \dot{\theta}$  where  $e$  is the back emf,  $K_e$  is the constant of proportionality. The units of the constants  $K_e$  and  $K_t$  are equal, hence we assume them to be equal to  $K$ .

Newtons second law states that the force acting on an object is equal to the mass of that object times its acceleration [5] and Kirchoff's voltage law states that the algebraic sum of all voltages in a loop must equal zero [6]. Based on these two laws, we can derive two mathematical equations from Figure 4.4.3.

$$J\ddot{\theta} + b\dot{\theta} = K i$$

$$L \frac{di}{dt} + R i = V - K \dot{\theta}$$

Applying laplace transform to both equations,

$$s(Js + b)\Theta(s) = K I(s)$$

$$(Ls + R)I(s) = V(s) - Ks\Theta(s)$$

$$G(s) = \frac{\Theta(s)}{V(s)} = \frac{K}{(Js + b)(Ls + R) + K^2} \frac{\text{rad/sec}}{V}$$

Taking the angular velocity and electric current as state variables, the State-Space model can also be derived. Thus, taking the voltage applied to the motor as the input and the angular velocity as the output,



$$\frac{d}{dt} \begin{bmatrix} \dot{\theta} \\ i \end{bmatrix} = \begin{bmatrix} -\frac{b}{J} & \frac{K}{J} \\ -\frac{K}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ i \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ -\frac{1}{L} \end{bmatrix} V$$

$$y = [1 \ 0] \begin{bmatrix} \dot{\theta} \\ i \end{bmatrix}$$

From these mathematical equations, the various simulations can now be performed in MATLAB to determine which controller will suite the user requirements.

```
tf_model.m x +
1 - J = 0.01; %moment of inertia of the rotor
2 - b = 0.1; %motor viscous friction constant
3 - K = 0.01; %electromotive force constant/motor torque constant
4 - R = 1; %electric resistance
5 - L = 0.5; %electric inductance
6 - motot_sys = tf('s');
7 - Motor_tf = K/((J*motot_sys+b)*(L*motot_sys+R)+K^2);
```

## Chapter 5: Results and Discussion

In this chapter, the results of the simulations are illustrated and described.

Proceeding with the simulations, the P controller was first used on the transfer function of the motor. This is the most basic controller you can with regards to PID controllers. In this case, the only active parameter is the proportional control. The plant in this case is the transfer function of the motor. The basic idea of a control system is to figure out how to generate the appropriate actuating signal (input) so the system can produce the desired controlled variable (Output).

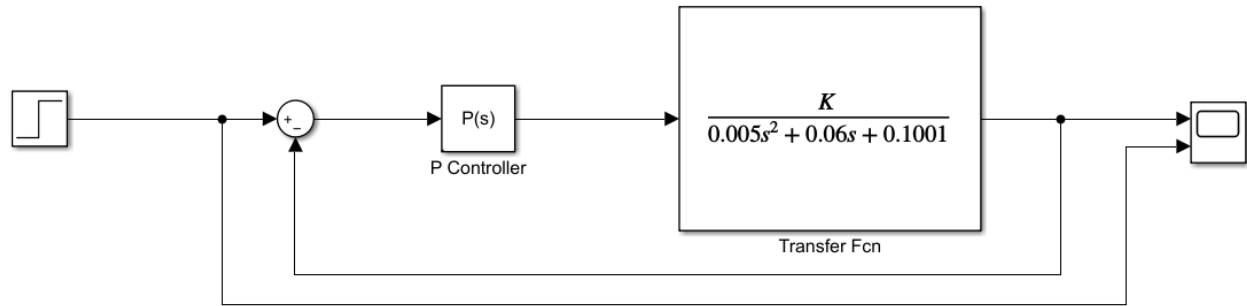


Figure 5.1 Model with proportional control

The set point was created using a unit step block in MATLAB. The essence of using a controller is to drive the error to zero. The error is basically the difference between the output and the input. If the input signal is equal to the output signal, a controller would be of no use. The proportional controller is designed to gradually reduce the error derived. As time increases, the error gradually reduces to zero hence. In figure 4.4.4 the feedback loop using proportional control is illustrated.

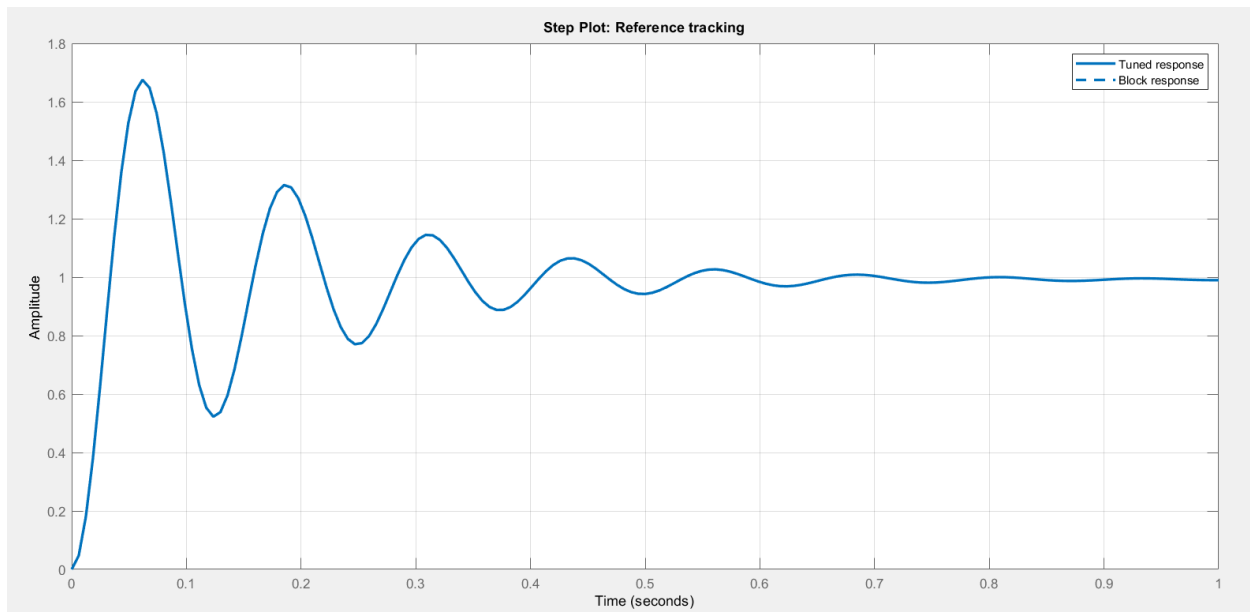


Figure 5.2 Proportional control simulation

The P controller is tuned using the PID tuner to arrive at a performance meeting the user requirements. The requirements were not met in this simulation however with the final response performance values indicated in Figure 5.3.

Controller Parameters		
	Tuned	Block
P	1282.6557	1282.6557
I	n/a	n/a
D	n/a	n/a
N	n/a	n/a
Performance and Robustness		
	Tuned	Block
Rise time	0.0225 seconds	0.0225 seconds
Settling time	0.634 seconds	0.634 seconds
Overshoot	68.8 %	68.8 %
Peak	1.68	1.68
Gain margin	Inf dB @ Inf rad/s	Inf dB @ Inf rad/s
Phase margin	13.6 deg @ 50.1 rad/s	13.6 deg @ 50.1 rad/s
Closed-loop stability	Stable	Stable

Figure 5.3 Performance values of proportional control

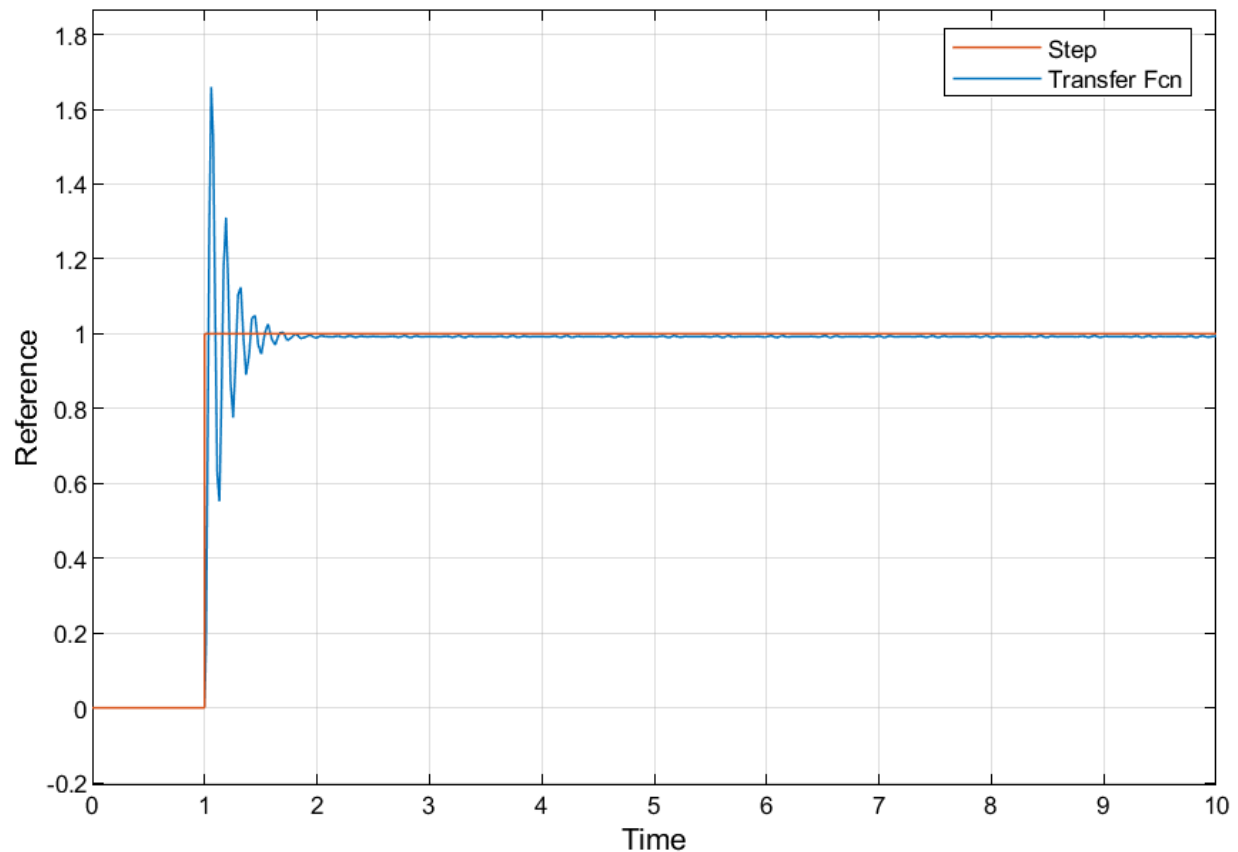


Figure 5.4 P control response of Output compared to Unit Step input

In tuning the response of the system, the focus was on the speed of the response and the settling time. In this case the settling time achieved was 0.634 seconds and the rise time was 0.0225 seconds. There was a lot of oscillations in this system before it finally reached Steady State. This is not a suitable response for the manipulator since it will be carrying loads. Thus, continuous oscillations might cause the suction to be ineffective and eventually lead to the dropping of the load while on its path. This cannot be solved regardless of the amount of gain included.

For this reason, the PI controller was explored. The PI controller is easy to stabilize faster compared to the P controller. With the PI controller, it acts simply as a memory controller. It keeps record of past input signals applied to the plant, by adding all signals inputted. When the steady

state is below the desired output, the error term is a non-zero. When a non-zero error is integrated, the output will increase. The integral output will continuously change as long as there is an error in the system. This will increase the speed of the motor as the P and I work hand in hand.

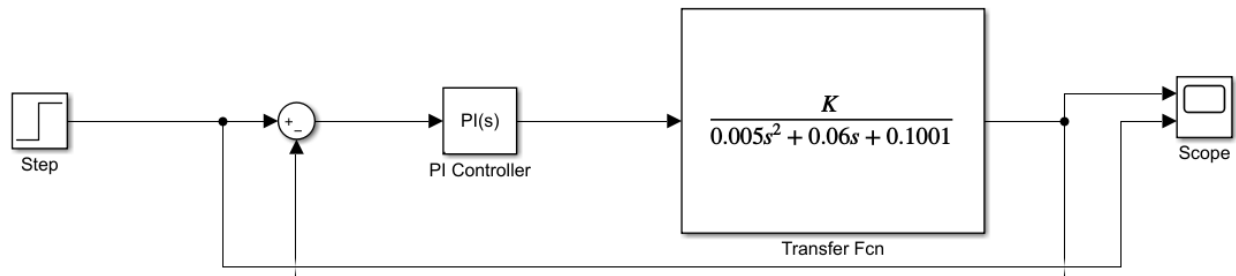


Figure 5.5 PI control System

The PI controller in Figure 5.4 was tuned and applied to the plant. This resulted in a 1.82 % overshoot in the system. Which is higher than the desired overshoot set initially (which was 0%) . The system had a fast rise time of 0.335 seconds and a very rapid settling time of 0.504 seconds as indicated in Figure 5.5.

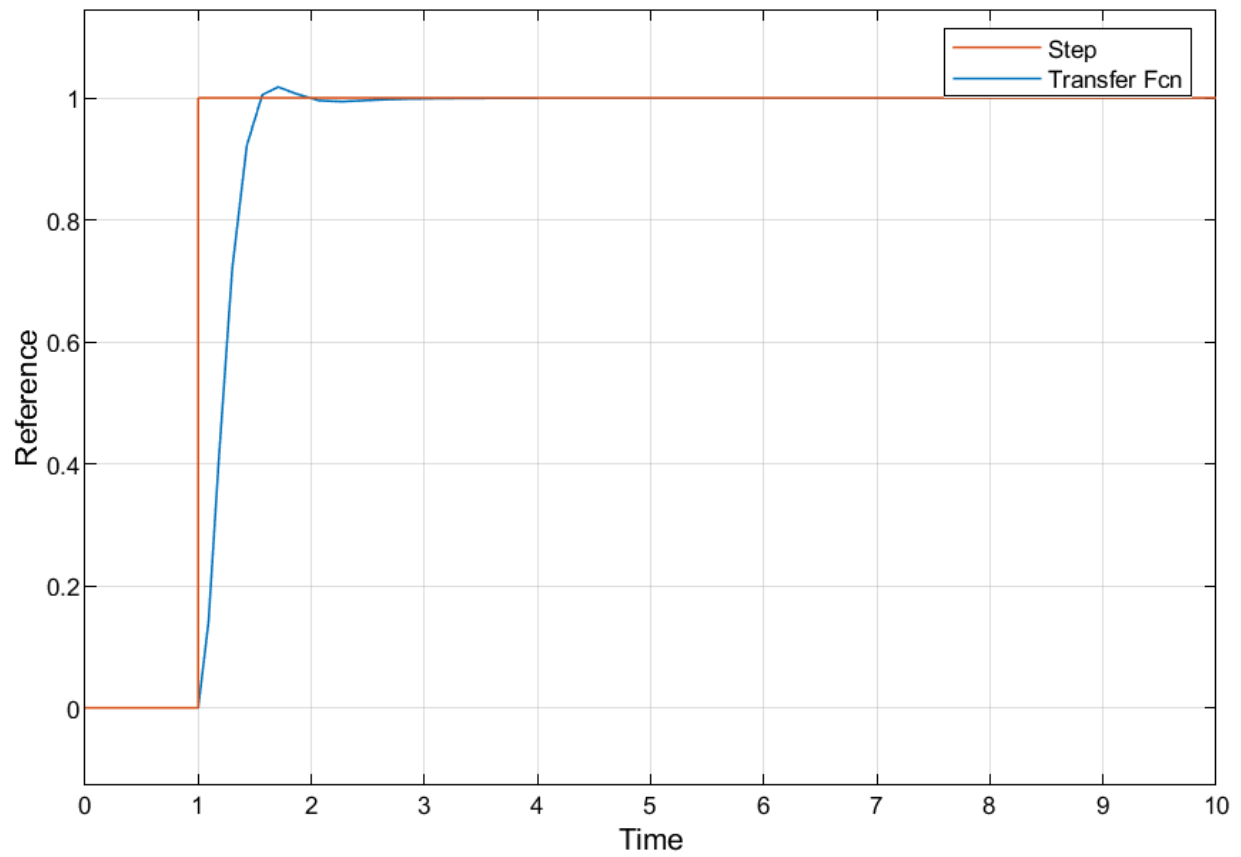


Figure 5.6 Input and Output Response compared

Controller Parameters		
	Tuned	Block
P	23.4622	23.4622
I	43.8079	43.8079
D	n/a	n/a
N	n/a	n/a

Performance and Robustness		
	Tuned	Block
Rise time	0.335 seconds	0.335 seconds
Settling time	0.504 seconds	0.504 seconds
Overshoot	1.82 %	1.82 %
Peak	1.02	1.02
Gain margin	Inf dB @ Inf rad/s	Inf dB @ Inf rad/s
Phase margin	68.4 deg @ 4.27 rad/s	68.4 deg @ 4.27 rad/s
Closed-loop stability	Stable	Stable

Figure 5.7 Performance Characteristics of PI controller

The PID controller was then explored to test if the 0% overshoot was possible. Adding the derivative to the PI controller helps predict the future of the response in order to respond to the prediction. The derivative produces a measure of the rate of change of the error. Thus, how fast the error is increasing or decreasing. In this case, if the plant is approaching the goal quickly, it means that the error is quickly decreasing. When the error is quickly decreasing, it means the error has a negative rate of change and this produces a negative value through the derivative. The derivative allows the controller to determine if the plant is closing in on the target very quickly. Hence it prematurely slows the speed of the motor when approaching the target. By this, the overshoot is completely reduced to 0% as indicated in figure 5.10.

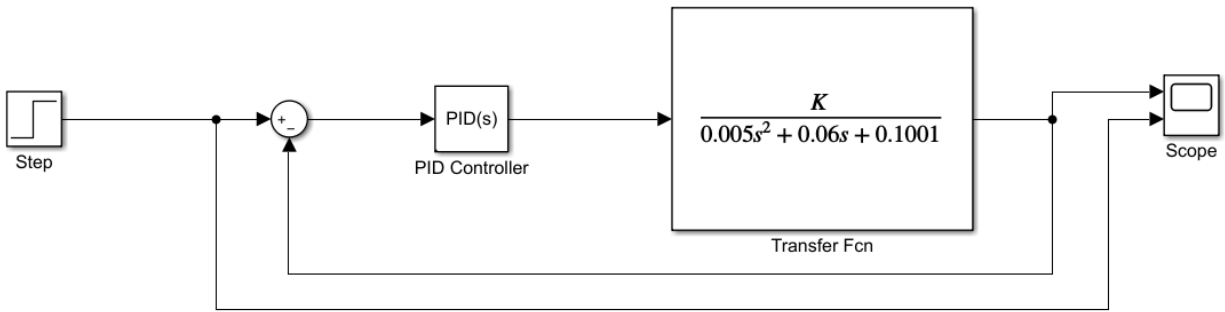


Figure 5.8 PID controller implementation.

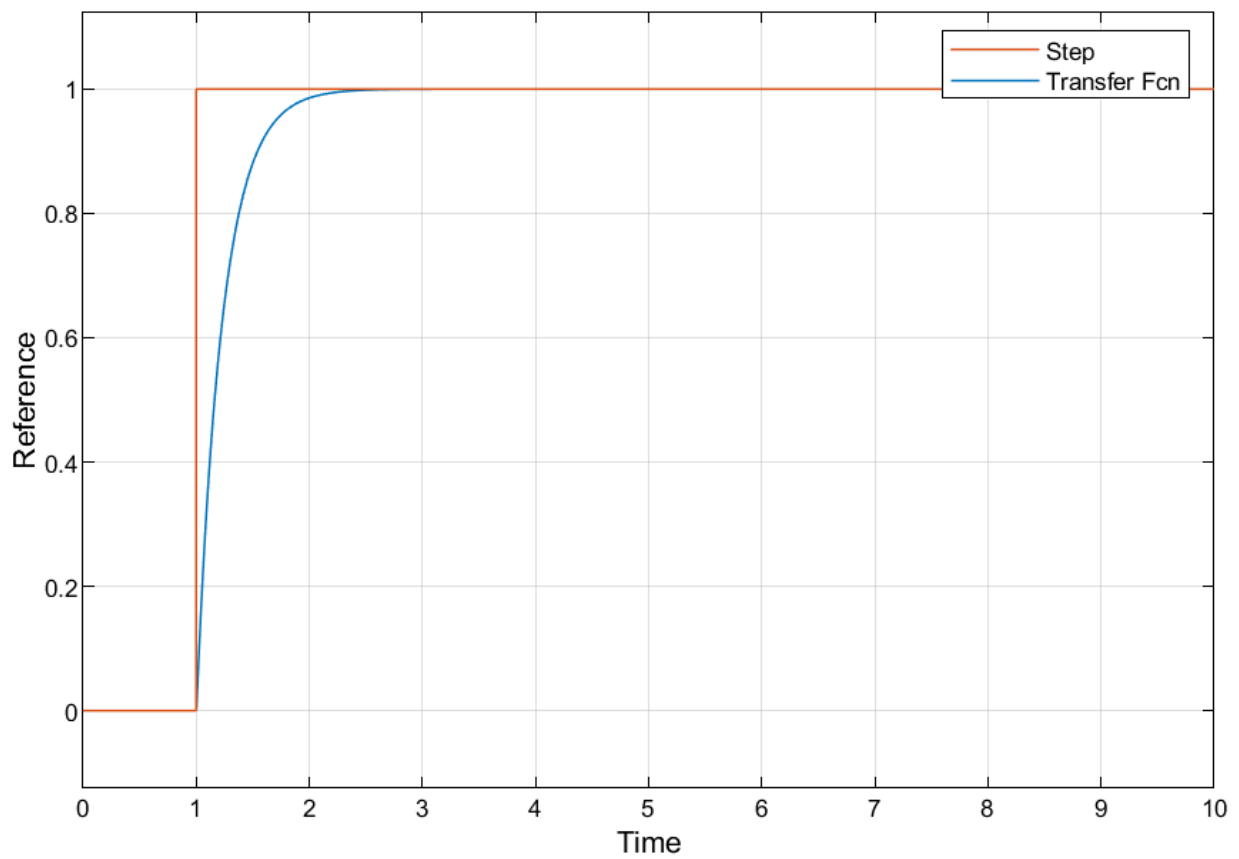


Figure 5.9 Input against Output response comparison after PID control is implemented



Controller Parameters		
	Tuned	Block
P	25.5178	25.5178
I	42.6434	42.6434
D	2.1299	2.1299
N	487.5546	487.5546
Performance and Robustness		
	Tuned	Block
Rise time	0.515 seconds	0.515 seconds
Settling time	0.92 seconds	0.92 seconds
Overshoot	0 %	0 %
Peak	0.999	0.999
Gain margin	Inf dB @ Inf rad/s	Inf dB @ Inf rad/s
Phase margin	90 deg @ 4.27 rad/s	90 deg @ 4.27 rad/s
Closed-loop stability	Stable	Stable

Figure 5.10 Performance characteristics of PID implementation

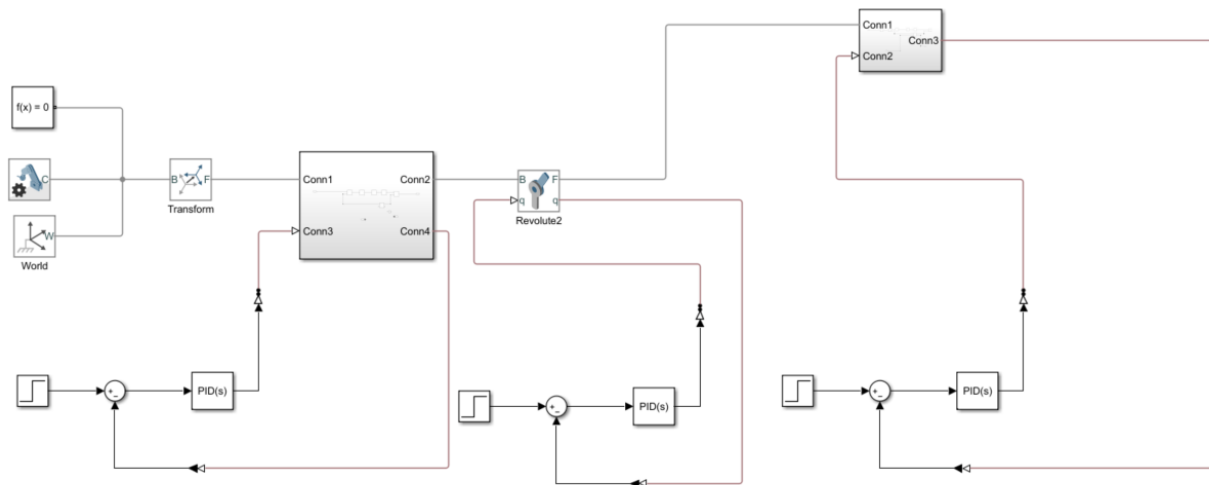


Figure 5.11 PID controller added to main system

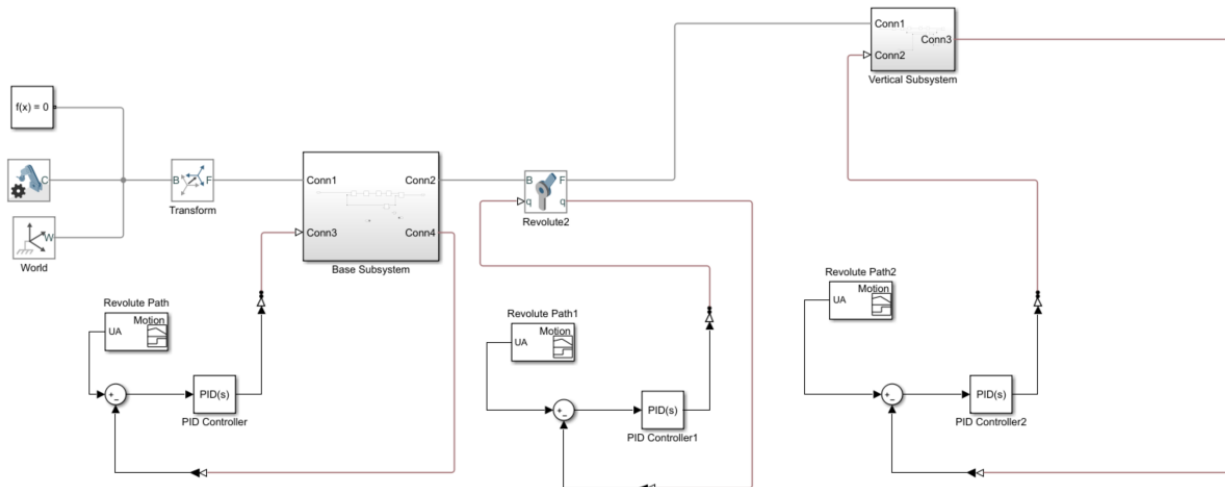


Figure 5.12 Path Planning applied to PID controller into the system

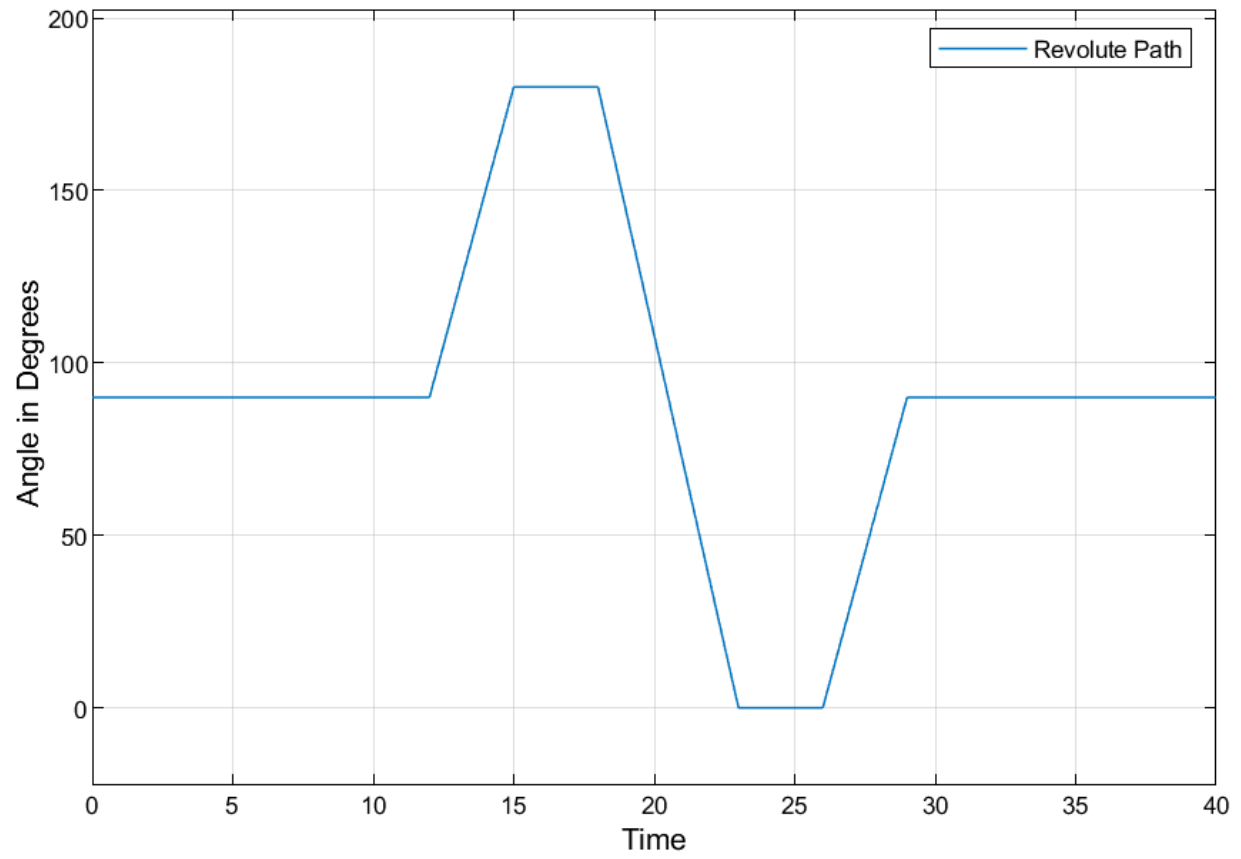
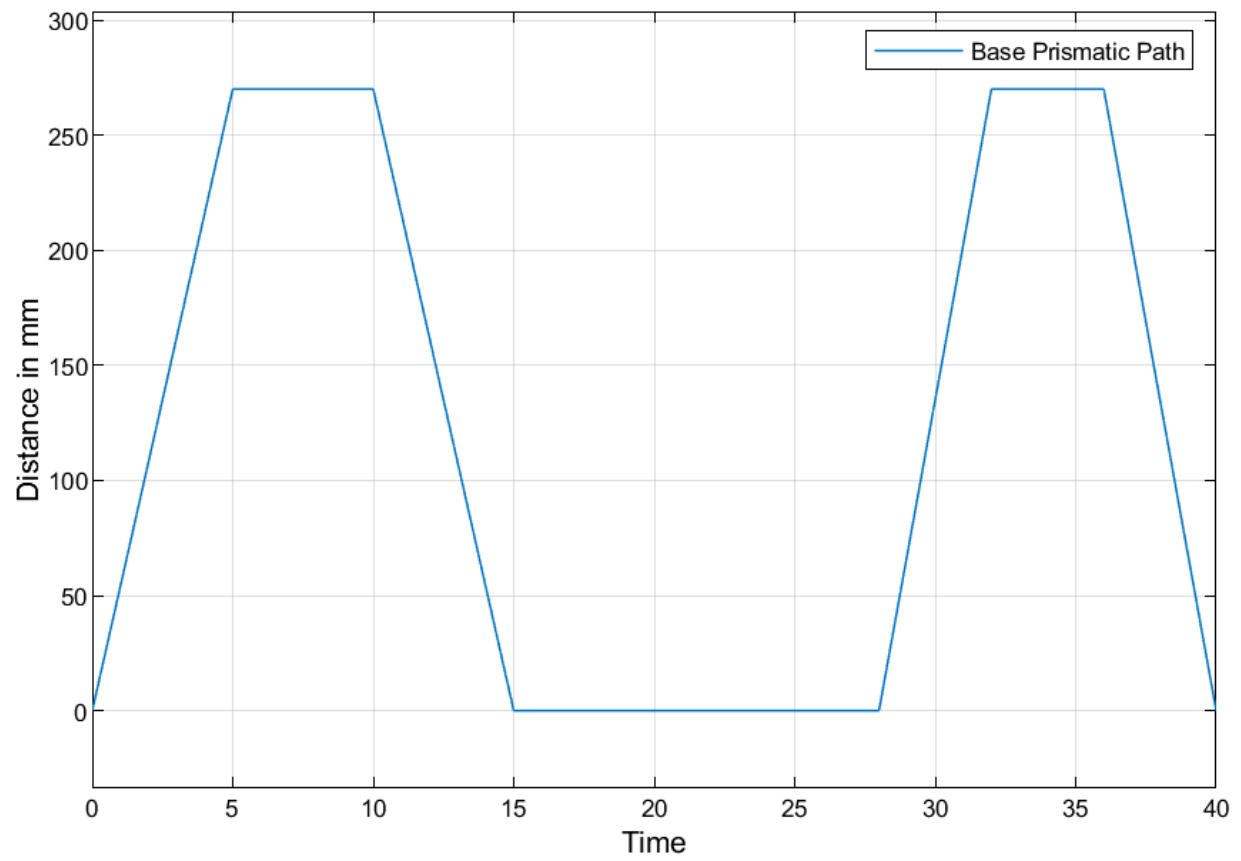


Figure 5.13 Rotating Path of the PRP manipulator



*Figure 5.14 Base path of the PRP manipulator*

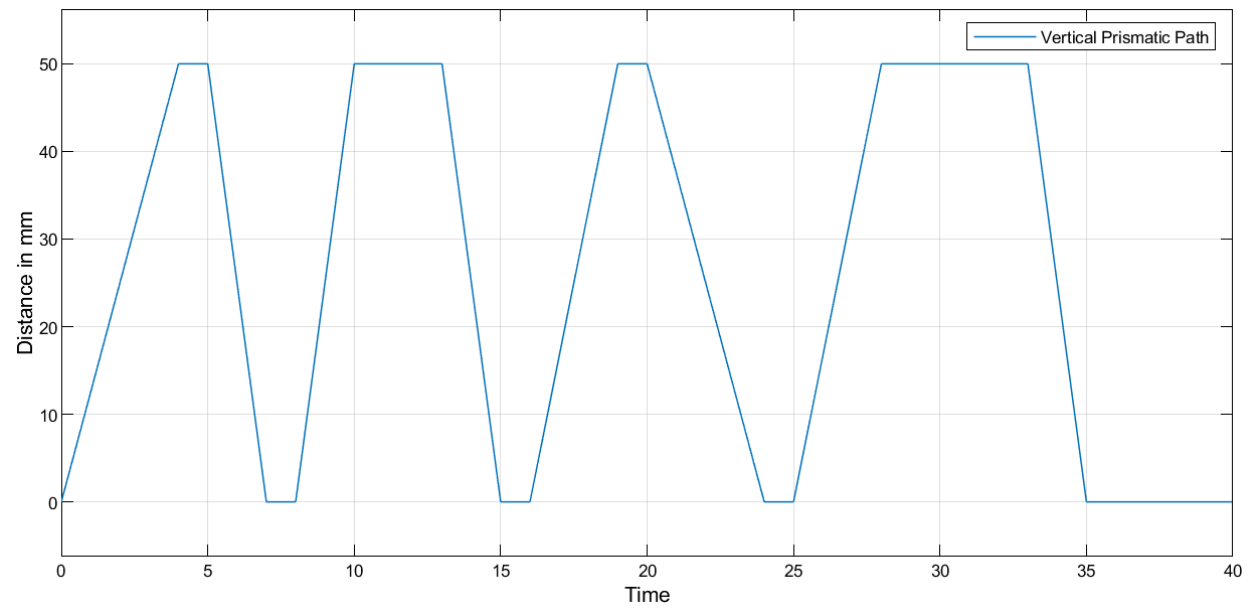


Figure 5.15 Vertical Path of the PRP Manipulator

## Chapter 6: Conclusion

The PID controller was able to produce the best results to meet the system requirements as compared to the P and PI controllers. However, the P and PI controllers could be used when the focus is not put on the overshoot and other specifications.

Indeed, the PRP robot manipulator could revolutionise the print process in Ashesi university and beyond. Its existence would be appreciated mostly during final project submission periods and at night when no one is allowed to visit the lab.

### 6.1 Limitations

The availability of parts in the country was one of my main limitations. Shipping, even though possible outrun the cost of the items unavailable in the country. The documentations in matlab with regards to stepper motors was also a huge limitation in my project. For these reasons, I was unable to build the robot manipulator to perform more tests and tweak the design if needed.

### 6.2 Future Works

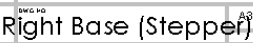
Remote controlling the PRP manipulator could be one of the major focuses to improve the functionality of the PRP manipulator during emergency periods.

## References

- [1] T. H. Luu and T. H. Tran, "3D vision for mobile robot manipulator on detecting and tracking target," 2015 15th International Conference on Control, Automation and Systems (ICCAS), Busan, 2015, pp. 1560-1565, doi: 10.1109/ICCAS.2015.7364605.
- [2] T. Subhasankari, A. Sharvin Infant, A. Viswasundar, M. Venkatesan and N. Mithran, "Integration of Hall Sensor in a 3D Printer as a Limit Switch," 2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 2017, pp. 1-3, doi: 10.1109/ICCIC.2017.8524539.
- [3] N. Kumar, A. Porwal, A. R. Singh, R. Naskar and S. Purwar, "Event Triggered Control of Robot Manipulator," 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 2019, pp. 362-366, doi: 10.1109/SPIN.2019.8711653.







44

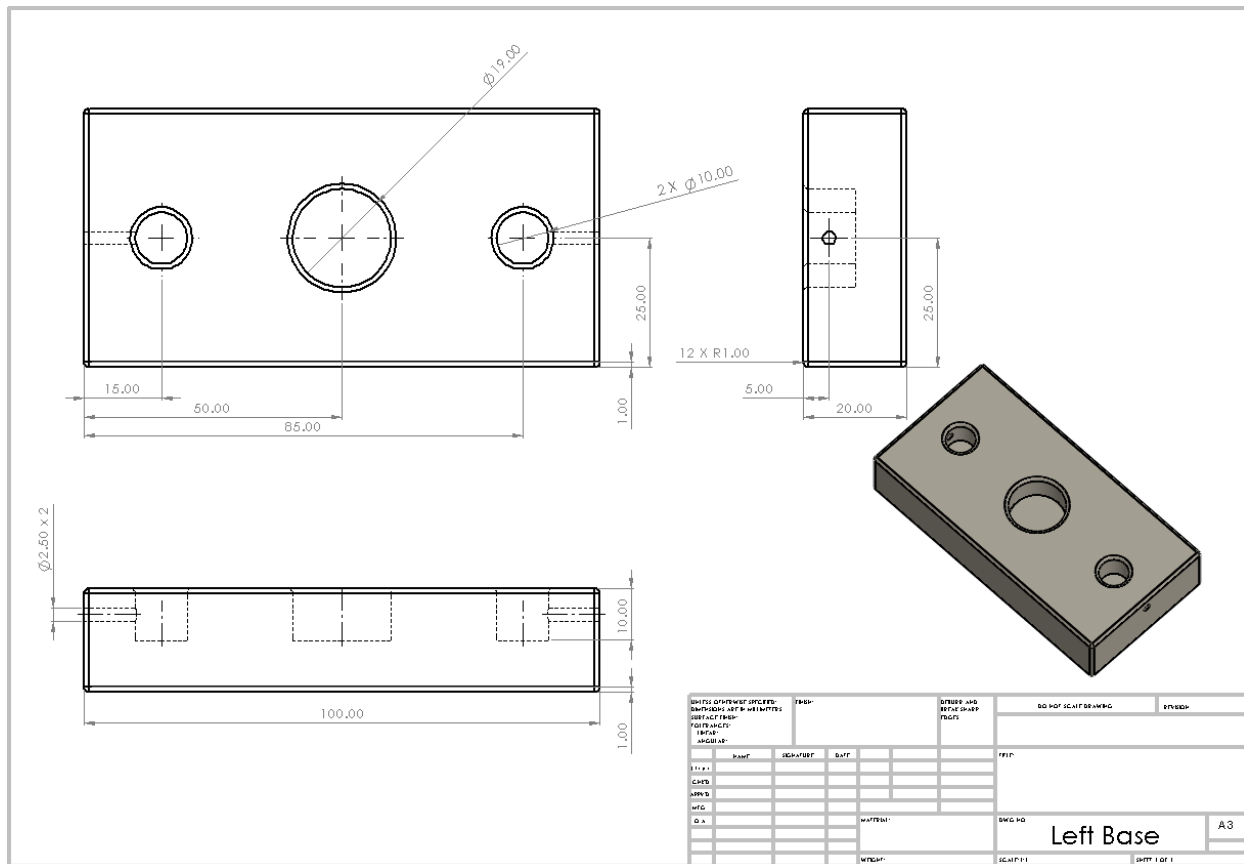
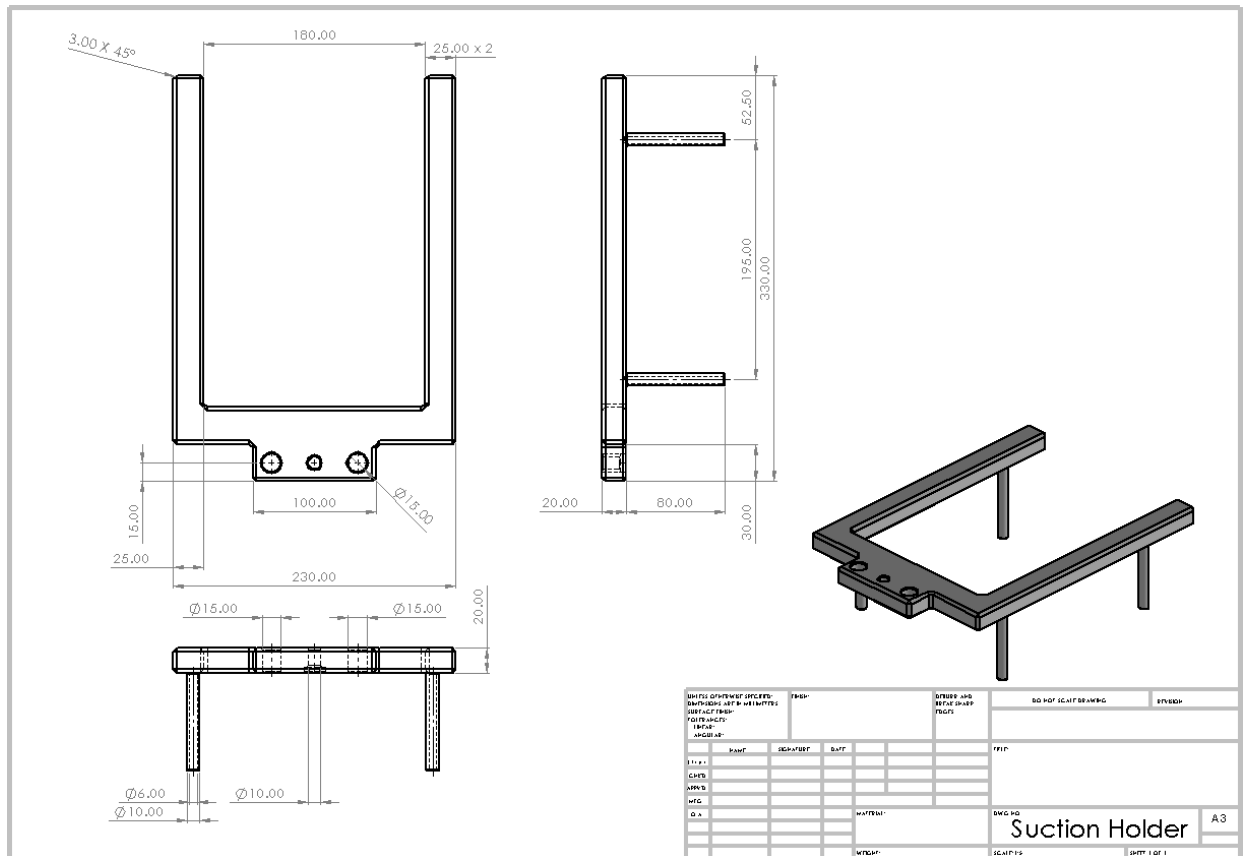


Figure A- 2 Drawing Dimensions of Left Base



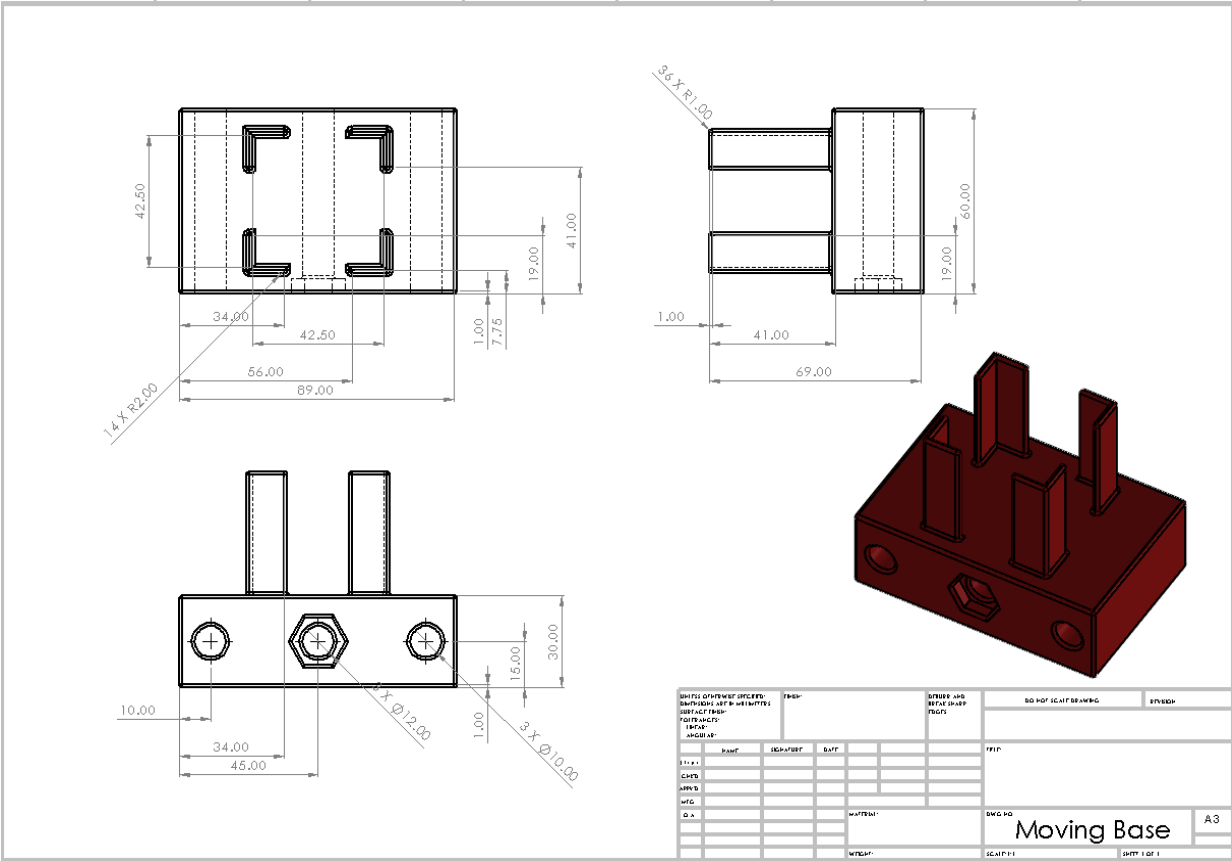


Figure A-4 Drawing Dimensions of Moving Base



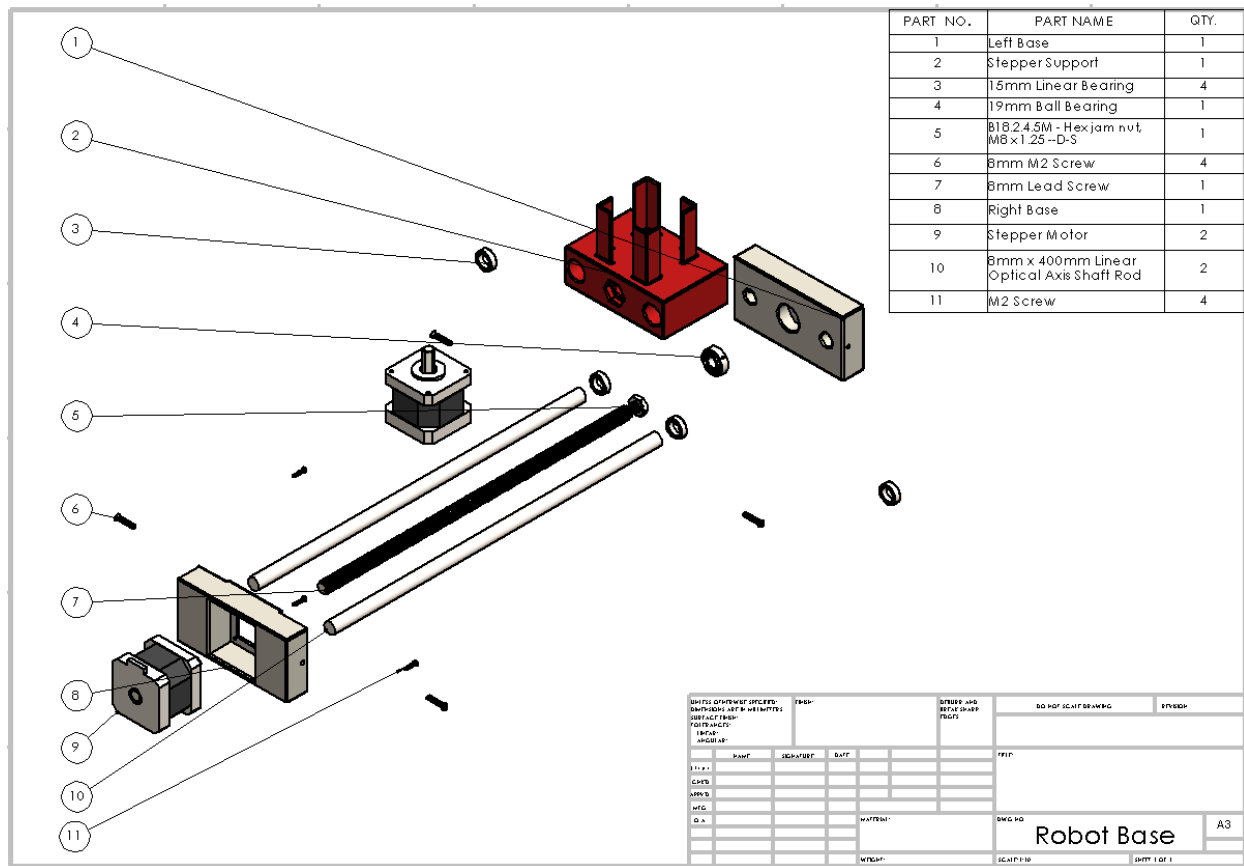


Figure A-6 Assembly Drawing of Robot Base Subsystem with BOM

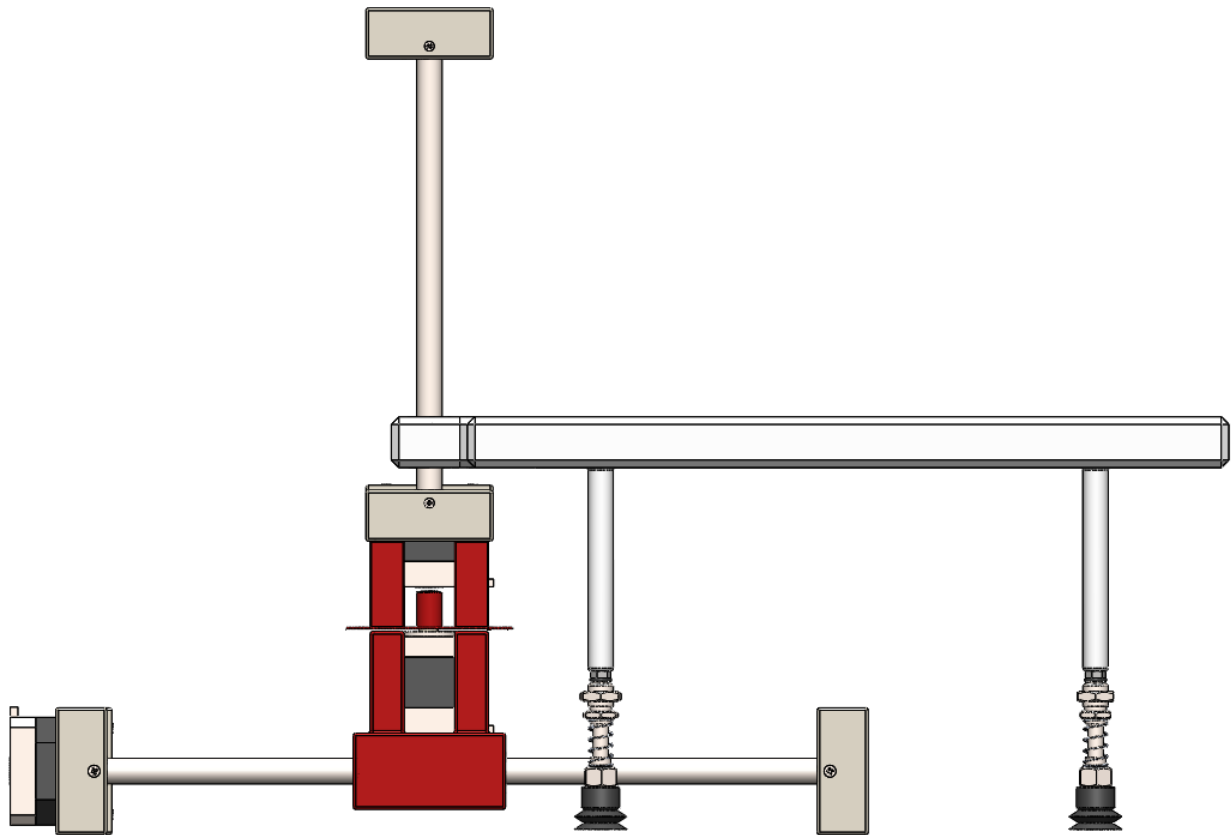


Figure A-7 Side View of PRP Robot Manipulator