

**TWO DEGREE OF FREEDOM (2DOF) MOTION CONTROL OF  
UPPER LIMB ROBOTIC ARM MECHANISM**

**ABDUL RAHMAN BIN KHAIRUDDIN**



**BACHELORS OF MECHATRONICS ENGINEERING WITH  
HONOURS  
UNIVERSITI TEKNIKAL MALAYSIA MELAKA**

**2019**



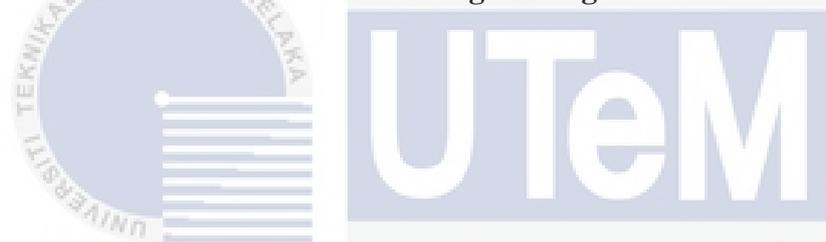
اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

**TWO DEGREE OF FREEDOM (2DOF) MOTION CONTROL OF UPPER LIMB  
ROBOTIC ARM MECHANISM**

**ABDUL RAHMAN BIN KHAIRUDDIN**

**A report submitted  
in partial fulfillment of the requirements for the degree of  
Bachelor of Mechatronics Engineering with Honours**

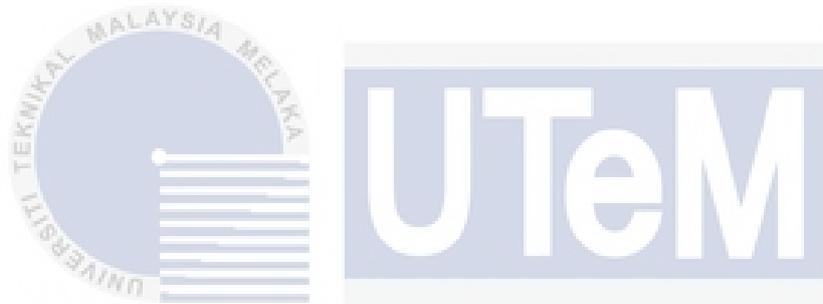


**Faculty of Electrical Engineering**  
اونيورسي تيكنيكل ماليسيا ملاك

**UNIVERSITI TEKNIKAL MALAYSIA MELAKA**

**UNIVERSITI TEKNIKAL MALAYSIA MELAKA**

**2019**

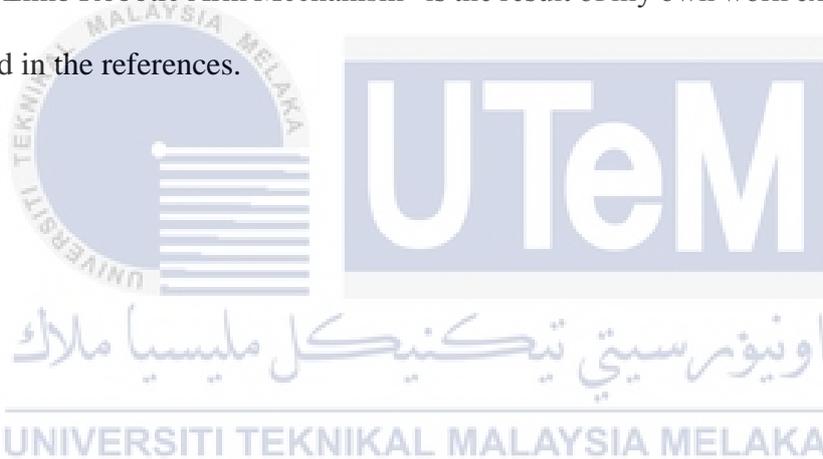


اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

## DECLARATION

I declare that this report entitled “Two Degree of Freedom (2DOF) Motion Control of Upper Limb Robotic Arm Mechanism” is the result of my own work except for quotes as cited in the references.



Signature : .....

Author : .....

Date : .....

## APPROVAL

I hereby declare that I have read this thesis and in my opinion, this thesis is sufficient in terms of scope and quality for the award of Bachelor of Mechatronic Engineering with Honours.



اونيورسيتي تيكنيكل مليسيا ملاك

Signature : .....

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Supervisor Name : .....

Date : .....

## DEDICATION

To my beloved mother and father.



## ABSTRACT

The project's main purpose is to plan a controller which can control the yield plot for an upper limb of robotic arm. A structure of mechanical arm of two degree of freedom (2-DOF) designed and optimized. Study is done to explore the controller to be connected on the mechanical arm. PID controller is picked and analysed in term of its execution, for example, rise time, settling time, steady-state error, and overshoot. The experimental setup is carried out. Open loop simulation are first done to acquire the transfer function of each of the motor. Simulation for an uncompensated framework is done to watch the closed loop system characteristics without utilizing the controllers. From that point onward, closed loop simulations are completed for compensated system by utilizing PID controller. Two kinds of trials are done, to be specific point to point direction control and tracking control tests. Investigation is made dependent on the outcomes acquired.

## ABSTRAK

Tujuan utama projek ini adalah untuk mereka bentuk pengawal yang dapat mengawal sudut keluaran untuk lengan robot. Struktur lengan robot dua darjah kebebasan (2-DOF) direka dan dioptimumkan. Kajian dijalankan untuk mengkaji jenis pengawal yang sesuai untuk digunakan pada lengan robot. Pengawal PID dipilih dan dikaji dari segi prestasinya seperti kesilapan keadaan mantap, masa penyelesaian, masa meningkat dan 'overshoot'. Persediaan eksperimen dijalankan. Simulasi 'open loop' mula-mula dijalankan untuk mendapatkan fungsi pemindahan setiap motor. Simulasi untuk sistem 'uncompensated' dijalankan untuk memerhatikan ciri sistem 'closed loop' tanpa menggunakan pengawal. Selepas itu, simulasi 'closed loop' dijalankan untuk sistem 'compensated' menggunakan pengawal PID. Dua jenis eksperimen dijalankan, iaitu titik ke arah kawalan trajektori dan eksperimen kawalan penjejakan. Analisis dibuat berdasarkan hasil yang diperoleh.

## ACKNOWLEDGEMENTS

I find the relief that my journey cannot be completed without the help of many people. I had accepted many useful assists and guideline from many different persons during the whole performance of my Final Year Project. I love to extent my deepest gratitude and appreciation to these people. Firstly, I felt very fortune to have Dr. Mariam Binti Md Ghazaly, my supervisor who guide me throughout this project. Thank you for expertise and consistent advised.

Furthermore, I must share my special thanks of gratitude to my parents for providing me the opportunity to study at university and giving me finance and mentally support throughout my study life and thee whole period of my FYP. This accomplishment would not have been done or completed without them.

Last but not least, I would resemble my indebtedness to all my course mates for willing to share and help me to complete my FYP with their knowledge and experiences. I perceive my FYP as a huge breakthrough in profession growth. I will make great effort to practice added talents and knowledge and I will endure to effort on the development, in direction to achieve preferred career goals.

# TABLE OF CONTENTS

<b>DECLARATION</b>	
<b>APPROVAL</b>	
<b>DEDICATION</b>	
<b>ABSTRACT</b>	<b>i</b>
<b>ABSTRAK</b>	<b>ii</b>
<b>ACKNOWLEDGEMENTS</b>	<b>iii</b>
<b>TABLE OF CONTENTS</b>	<b>iv</b>
<b>LIST OF FIGURES</b>	<b>viii</b>
<b>LIST OF TABLES</b>	<b>xii</b>
<b>LIST OF SYMBOLS AND ABBREVIATIONS</b>	<b>xiv</b>
<b>LIST OF APPENDICES</b>	<b>xv</b>
<b>CHAPTER 1 INTRODUCTION</b>	<b>1</b>
1.1 Motivation	1
1.2 Problem Statement	2
1.3 Objective	2
1.4 Scope and Limitation	3

<b>CHAPTER 2 LITERATURE REVIEW</b>	<b>4</b>
2.1 Introduction	4
2.2 Robotics	4
2.3 Upper limb robotic arm	5
2.4 Motor	8
2.5 Controllers	10
2.6 Summary	15
<b>CHAPTER 3 METHODOLOGY</b>	<b>16</b>
3.1 Introduction	16
3.2 Research Methodology	16
3.2.1 Project Methodology	17
3.2.2 Experiment Methodology	17
3.3 Structure of Robotic Arm	17
3.4 Equipment usage	20
3.4.1 12V DC Geared Motor with Hall Effect Encoder by Cytron	20
3.4.2 Micro-Box 2000/2000C (xPC Target Machine)	21
3.4.3 Motor Driver Circuit	22
3.5 System Overview	23
3.6 Calibration of Encoder for DC Geared Motor	24
3.7 Open Loop Control	24

3.8	System Identification Tools	27
3.9	Development of Closed Loop System	28
3.9.1	Development of Uncompensated Closed Loop System	28
3.9.2	Controller Design	30
3.9.3	Design and Development of Compensated Closed Loop System with Proportional-Integral-Derivative (PID) Controller	30
3.9.4	Tuning Methods	32
3.9.5	Trial and Error Method	32
3.9.6	Ziegler-Nichols Method	33
<b>CHAPTER 4 RESULTS AND DISCUSSION</b>		<b>34</b>
4.1	Introduction	34
4.2	Open Loop Test	35
4.2.1	Linearity	50
4.3	Uncompensated System	52
4.3.1	Point to Point Trajectory Control for Uncompensated System	52
4.3.2	Tracking Control for Uncompensated System	56
4.4	Compensated System with PID controller	60
4.4.1	Point to Point Trajectory Control with PID controller	60
4.4.1.1	Trial and Error	60
4.4.1.2	Ziegler Nichols	68
4.4.2	Tracking Control with PID controller	76

4.5	Summary	82
<b>CHAPTER 5 CONCLUSION AND RECOMMENDATION</b>		<b>83</b>
5.1	Conclusion	83
5.2	Recommendation	85
<b>REFERENCES</b>		<b>87</b>
<b>APPENDICES</b>		<b>91</b>



## LIST OF FIGURES

Figure 2.1: Robotic Mechanism.....	5
Figure 2.2: Illustration of an Upper Appendage of Automated Arm with (2DOF) [8] .....	6
Figure 2.3: Type of Control Methods for Robotic Arm.....	10
Figure 2.4: Different Arm Movement [23] .....	11
Figure 2.5: Comparison of Three Controllers for RMS Error. Bar Charts Represent Mean Values for Twenty Two Motions and Error Bars represent Maximum and Minimum Values [23].....	12
Figure 2.6: Comparison of Three Controllers for Correlation Factors [23].....	12
Figure 2.7: Comparison of Three Controllers for Mean Absolute Error [23].....	13
Figure 2.8: Experiment result in the upper displacement zone [24] .....	14
Figure 3.1: Drawing of the Robotic Arm Mechanism .....	18
Figure 3.2: Structure of first link (a) Front view, (b) Back view, (c) Side view, (d) Top view, (e) Bottom view .....	18
Figure 3.3: Structure of second link (a) Top view, (b) Bottom view, (c) Back view, (d) Side view .....	19
Figure 3.4: Structure of robotic after assemble all the parts .....	19
Figure 3.5: DC Geared Encoder Motor and its Detachable Cover .....	21
Figure 3.6: Components of Micro-Box Module .....	23

Figure 3.7: System Concept .....	24
Figure 3.8: Block Diagram of Open Loop System .....	25
Figure 3.9: The Open Loop System Model Block Diagram .....	26
Figure 3.10: Real Time Simulation and Experimental Block Diagram .....	27
Figure 3.11: System Identification Tools.....	28
Figure 3.12: The Block Diagram of the Uncompensated Closed Loop System for the Robotic Arm.....	29
Figure 3.13: The Uncompensated Closed System Block Diagram in Simulink .....	29
Figure 3.14: Block diagram of a typical PID controller .....	31
Figure 3.15: Compensated Closed Loop System with PID Controller using MATLAB Simulink.....	31
Figure 4.1: Simulation and experiment flow .....	35
Figure 4.2: Input voltage and Output Angle Versus Times Graph, (2V).....	40
Figure 4.3: Input voltage and Output Angle Versus Times Graph, (3V).....	41
Figure 4.4: Input voltage and Output Angle Versus Times Graph, (4V).....	42
Figure 4.5: Input voltage and Output Angle Versus Times Graph, (5V).....	43
Figure 4.6: Input voltage and Output Angle Versus Times Graph, (6V).....	44
Figure 4.7: Input voltage and Output Angle Versus Times Graph, (7V).....	45
Figure 4.8: Input voltage and Output Angle Versus Times Graph, (8V).....	46
Figure 4.9: Input voltage and Output Angle Versus Times Graph, (9V).....	47
Figure 4.10: Input voltage and Output Angle Versus Times Graph, (10V) .....	48
Figure 4.11: The Error of the Output Angle from the Voltages Applied (2V-10V)..	49
Figure 4.12: Error of the output angle when 5V is applied as input (lowest error) ...	50
Figure 4.13: Graph of output angles against input voltages .....	51

Figure 4.14: The performance of the robotic arm for 15° of reference angle.....	53
Figure 4.15: The performance of the robotic arm for 30° of reference angle.....	54
Figure 4.16: The performance of the robotic arm for 60° of reference angle.....	55
Figure 4.17: Results of tracking error experiment for an uncompensated system with input angle of 15° .....	57
Figure 4.18: Results of tracking error experiment for an uncompensated system with input angle of 30° .....	58
Figure 4.19: Results of tracking error experiment for an uncompensated system with input angle of 60° .....	59
Figure 4.20: Result of Point to Point Trajectory Control Experiment for PID Control with Input Angle of 15° and $K_p$ value of 30 .....	61
Figure 4.21: Result of Point to Point Trajectory Control Experiment for PID Control with Input Angle of 30° and $K_p$ value of 30 .....	62
Figure 4.22: Result of Point to Point Trajectory Control Experiment for PID Control with Input Angle of 60° and $K_p$ value of 30 .....	63
Figure 4.23: Result of Point to Point Trajectory Control Experiment for PID Control with Input Angle of 15° and $K_p$ value of 30 $K_d$ value is 1.....	65
Figure 4.24: Result of Point to Point Trajectory Control Experiment for PID Control with Input Angle of 30° and $K_p$ value of 30 $K_d$ value is 1 .....	66
Figure 4.25: Result of Point to Point Trajectory Control Experiment for PID Control with Input Angle of 60° and $K_p$ value of 30 $K_d$ value is 1 .....	67
Figure 4.26: Results of Point to Point Trajectory Control Experiment for a PID Control System with input angle 180° and $K_p$ value of 120. ....	70
Figure 4.27: Results of Point to Point Trajectory Control Experiment for a PID Control System with input angle 180° and $K_p$ value of 120 (reduced scale). ....	71
Figure 4.28: Point-to-Point Trajectory Control Experiment Results for a 15 ° Input Angle PID Control System, 72 $K_p$ and 0.88992 $K_d$ value.....	73
Figure 4.29: Point-to-point trajectory control experiment results for a 30 ° input-angle PID control system, 72 $K_p$ and 0.88992 $K_d$ value .....	74

Figure 4.30: Point-to-point trajectory control experiment results for a $60^\circ$ input-angle PID control system, $72 K_p$ and $0.88992 K_d$ value .....	75
Figure 4.31: Performance of PID controller designed by using Trial and Error method with Sine Wave Signal for $15^\circ$ at 0.1 Hz .....	78
Figure 4.32: Performance of PID controller designed by using Trial and Error method with Sine Wave Signal for $30^\circ$ at 0.1 Hz .....	79
Figure 4.33: Performance of PID controller designed by using Ziegler's Nichols method with Sine Wave Signal for $15^\circ$ at 0.1 Hz .....	80
Figure 4.34: Performance of PID controller designed by using Ziegler's Nichols method with Sine Wave Signal for $30^\circ$ at 0.1 Hz .....	81



## LIST OF TABLES

Table 2.1: The Type and Characteristic for Automated Arms [7]	7
Table 2.2: Comparison of Multiple Motor Types in Terms of Their Pros and Cons [26]	9
Table 3.1: Specification of Motor	20
Table 3.2: Micro-Box Components	22
Table 3.3: Open Loop Simulation Parameters	26
Table 3.4: Parameters of Transient Response and the Effects Caused by Manipulating (P), (I), and (D) values	32
Table 3.5: Controller Parameters of Ziegler-Nichols Step Response Method	33
Table 4.1: System Identification Results for DC Motor ( $V_{in} = 2V$ )	40
Table 4.2: System Identification Results for DC Motor ( $V_{in} = 3V$ )	41
Table 4.3 System Identification Results for DC Motor ( $V_{in} = 4V$ )	42
Table 4.4: System Identification Results for DC Motor ( $V_{in} = 5V$ )	43
Table 4.5: System Identification Results for DC Motor ( $V_{in} = 6V$ )	44
Table 4.6: System Identification Results for DC Motor ( $V_{in} = 7V$ )	45
Table 4.7: System Identification Results for DC Motor ( $V_{in} = 8V$ )	46
Table 4.8: System Identification Results for DC Motor ( $V_{in} = 9V$ )	47
Table 4.9 System Identification Results for DC Motor ( $V_{in} = 10V$ )	48
Table 4.10: Data of output angle obtained when voltage is applied to the motor.	51

Table 4.11: Parameters for Point to Point Trajectory Control Experiments	53
Table 4.12: Parameters for Tracking Control Experiment	56
Table 4.13: Parameters for Point to Point Experiments using PID Controller	60
Table 4.14: Performance of Robotic Arm with PID Controller for Experimental with $K_p$ value of 30	63
Table 4.15: Performance of Robotic Arm with PID Controller for Simulation with $K_p$ value of 30	64
Table 4.16: Performance of Robotic Arm with PID Controller for Experimental with $K_p$ value of 30 and $K_d$ value is 1	67
Table 4.17: Performance of Robotic Arm with PID Controller for Simulation with $K_p$ value 30 and $K_d$ value is 1	68
Table 4.18: Point to Point Experiments Parameters using PID controller	69
Table 4.19: Parameters of PID controller obtained from simulation results	72
Table 4.20: Parameters for Point-to-Point Experiments using PID controller	73
Table 4.21: Performance of Robotic Arm with PID Controller for Experimental with $K_p$ value of 72 and $K_d$ value of 0.88992	75
Table 4.22: Performance of Robotic Arm with PID Controller for Simulation with $K_p$ value of 72 and $K_d$ value of 0.88992	76

## LIST OF SYMBOLS AND ABBREVIATIONS

DOF - Degree of Freedom

$K_p$  - Proportional Gain

$K_i$  - Integral Gain

$K_d$  - Derivative Gain

$K_u$  - Ultimate Gain

$T_i$  - Integrator Time Constant

$T_u$  - Ultimate Period

$T_d$  - Derivative Time Constant

$T_r$  - Rise Time

$T_s$  - Settling Time

$E_{ss}$  - Steady State Error

$OS$  - Overshoot

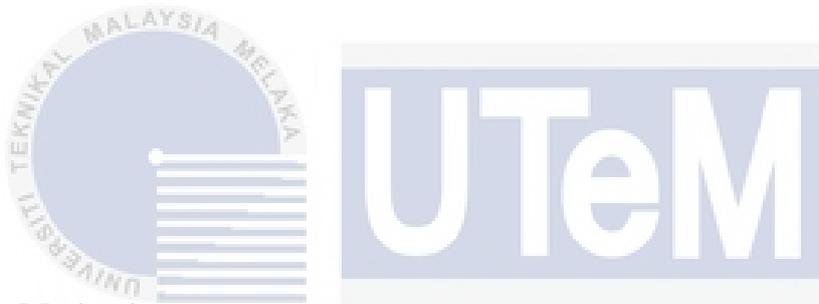
## LIST OF APPENDICES

APPENDIX A PROJECT RESEARCH METHODOLOGY IN FLOW CHART	91
APPENDIX B EXPERIMENT METHODOLOGY FLOW CHART	92
APPENDIX C PROJECT'S GANTT CHART	93



# CHAPTER 1

## INTRODUCTION



### 1.1 Motivation

Robots are progressively being incorporated into working undertakings to supplant people particularly to play out the monotonous assignment. These robots are right now utilized in numerous fields of uses including office, military errands, healing center tasks, hazardous condition and farming [13]. In this manner, the control of the robot ought to be planned so as to give fitting execution to a nonlinear, multivariable, nonstationary framework [14].

The motivation for this undertaking is to enhance the movement for a robotic arm utilizing position control and dissect the execution of the controllers as far settling time, rise time, and steady-state state error.

## 1.2 Problem Statement

Improper motion control may result in wounds and casualty. It is critical to improve the capability of a robotic arm along these lines. For movement control of automated arm, it is required to be in high precision, high efficiency, low in error for the output which empower it to decide the correct direction and the torque expected to accomplish a focused on result.

To achieve precise motion control, there are difficulties to obtain the desired output due to the sensitivity of the controller. For example, the parameters for PID controller are rather difficult to estimate in noisy environment while fuzzy logic does not required noise-free environment [1].

## 1.3 Objective

The main objectives of this project are:

1. To design and optimize the mechanism of 2DOF robotic arm
2. To derive each motor's transfer function by running the open loop test.
3. To design and develop controller to control the position for 2DOF upper limb robotic arm.
4. To analyze and compare the performance of the controller in terms of steady-state error, settling time and rise time.

#### 1.4 Scope and Limitation

The scope covered in this projects are:

1. Design and optimize the mechanism of 2DOF robotic arm using Solidworks.
2. Using MATLAB, develop and test controller to regulate the robotic arm position or output trajectory.
3. Study the PID controller performance.



## CHAPTER 2

### LITERATURE REVIEW



#### 2.1 Introduction

A human arm motion has been taken into consideration during the set-up designing stage. It was viewed as that it ought to take into consideration the best precise relocation conceivable, and that it ought to have the capacity to transport the best conceivable mass at the tip [24]. In this section, the structure and components which make up this whole project are investigated. Research is done in selecting the proper type of controllers to be used in the system.

#### 2.2 Robotics

In the assembling procedure, the greater part of the modern mechanized errands are done by specific machines which are intended to complete foreordained capacities. The resoluteness and for the most part surprising expense of these machines

have prompted an open enthusiasm for robots which are fit for playing out an assortment of assembling undertakings at lower creation costs and more prominent adaptability in works.

A terminology being used by Robot Institute of America provides a more accurate picture of modern robots: "A robot is a multi-purpose electromechanical manipulator designed to transfer equipment, components, tools or high-tech devices for a variety of functions through differential movements [25]." Figure 2.1 shows the mechanism of robots. It is closed loop system with feedback path.

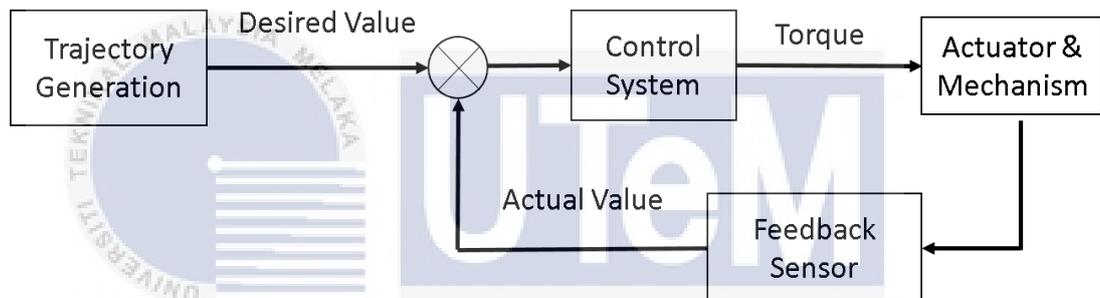


Figure 2.1: Robotic Mechanism

### 2.3 Upper limb robotic arm

A mechanical arm is a generally programmable robot with comparable capabilities to a human arm. The connections of such controllers are connected by joints that allow either rotational or translational relocation. The robot arms can be self-sufficient or physically controlled and can be used to perform a variety of errors with extraordinary accuracy [7]. The automated arm can also be installed or portable and can be used for home use. With respect to the automated arm, there 5 kind of mechanical arm that are utilized in modern today. Table 2.1 demonstrates the type and characteristic for automated arms.

In this segment, an upper appendage of mechanical arm with two level of opportunity is talked about as the task just spotlight on the upper appendage part. By and large the arm is associated with an engine and the revolution of engine prompts the movement of arm. Figure 2.2 represents an upper appendage of automated arm with (2DOF).

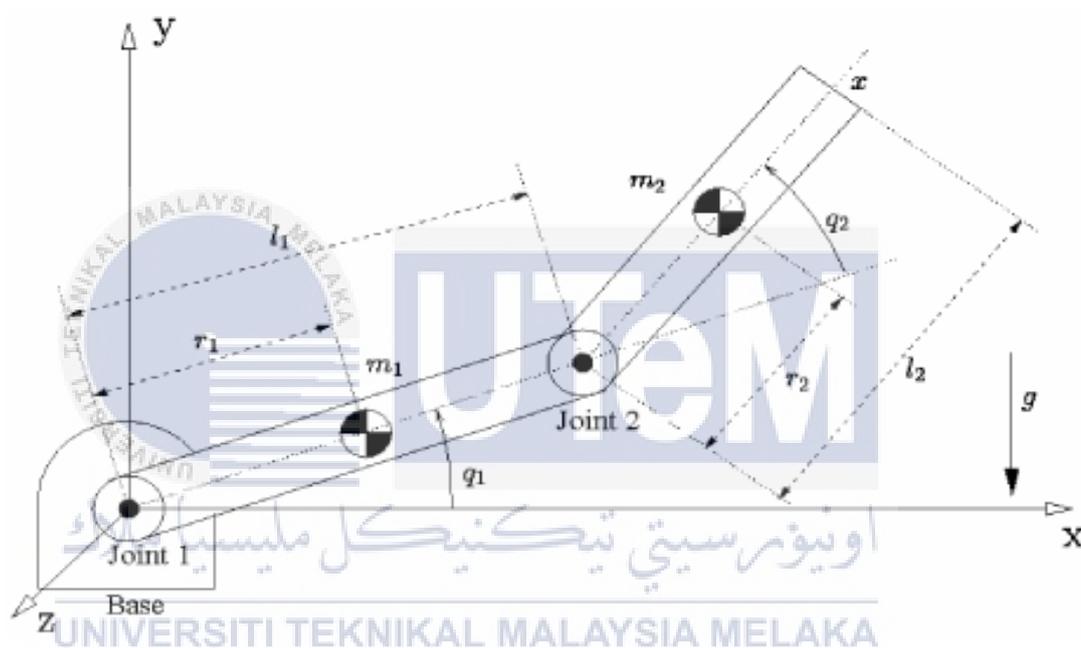
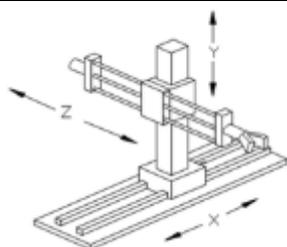
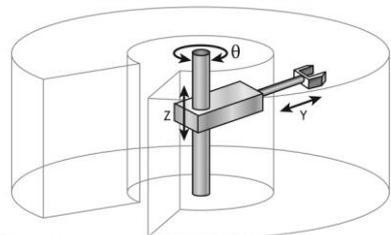
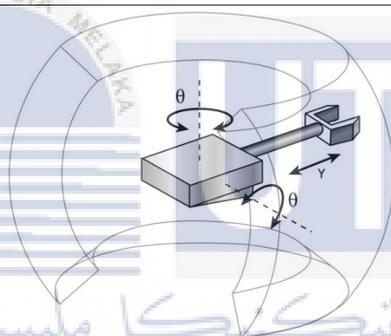
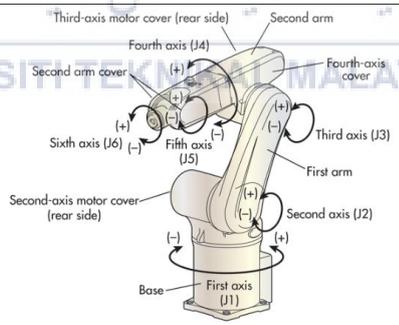
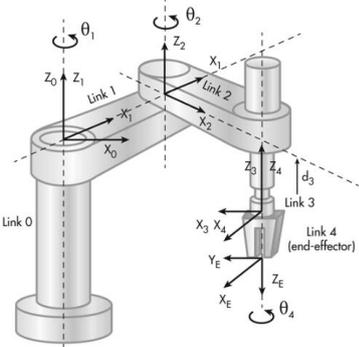


Figure 2.2: Illustration of an Upper Appendage of Automated Arm with (2DOF) [8]

Table 2.1: The Type and Characteristic for Automated Arms [7]

	Type	Diagram	Number of Joint	Characteristic
1	Cartesian Robot		Three perpendicular prismatic joints	Operate in X-Y-Z axis
2	Cylindrical Robot		Two prismatic joints and one revolute joint	Move cylindrically
3	Spherical Robot		Two revolute joints and one prismatic joint	Rotate on full rotation on spherical axis
4	Articulated Robot		All revolute joints	Used in complex workspace
5	SCARA		Two parallel revolute joints and one prismatic joint	Used for assembling parts on a plane

## 2.4 Motor

Motors convert electrical energy into mechanical energy and produce the torque required to move the desired target position. Motor selection and mechanical design is a critical part during the time spent structuring movement control framework [26]. The comparison of different motor types in terms of their strength and weakness is shown in Table 2.2.

A robotic arm activity requires high torque, relatively moderate speed and accurate positioning. Because of its excellent torque performance and apparently least disadvantages, brushless DC motor is selected to be used in this project after comparing these motors with each other.



Table 2.2: Comparison of Multiple Motor Types in Terms of Their Pros and Cons

[26]

	Stepper Motor	Brushed DC Servo Motor	Brushless DC Motor	Brushless Servo Motor
Pros	<ul style="list-style-type: none"> <li>• High torque at low speed</li> <li>• Inexpensive and widely available</li> <li>• High precision for motion control applications</li> </ul>	<ul style="list-style-type: none"> <li>• Easy installation</li> <li>• Low cost</li> <li>• High durability</li> </ul>	<ul style="list-style-type: none"> <li>• Low maintenance</li> <li>• High efficiency</li> <li>• Low noise production</li> </ul>	<ul style="list-style-type: none"> <li>• High acceleration</li> <li>• High torque</li> <li>• Low maintenance</li> </ul>
Cons	<ul style="list-style-type: none"> <li>• Lose up to 80% torque at high speed</li> <li>• High vibrations</li> <li>• High amount of heat</li> </ul>	<ul style="list-style-type: none"> <li>• Lower speed range</li> <li>• Poor heat dissipation</li> <li>• Moderately flat speed</li> </ul>	<ul style="list-style-type: none"> <li>• High cost</li> <li>• Complex to control</li> <li>• Required electric controller to operate motor</li> </ul>	<ul style="list-style-type: none"> <li>• Expensive</li> <li>• Complex system</li> </ul>
Fields	Positioning, micro-movement	Velocity control, high-speed position control	Position control	Robotics, pick-and-place, high-torque applications

## 2.5 Controllers

In recent decades, more attention has been paid to modeling and controlling flexible robot arms. To some extent, several basic research and control techniques for flexible manipulators have been investigated. [8].

There are numerous technique for control accessible for automated arm. Sorts of control of such non-linear frameworks can be partitioned into two noteworthy classes: (i) traditional control; and (ii) advance control. Traditional control includes: (1) adaptive control; (2) robust control, and (3) robust-adaptive ‘hybrid’ control. Advance control can be classified into: (1) learning control such as neural network-based control; (2) fuzzy-logic control; and (3) genetic control [1]. These methods of control are summarized in Figure 2.3.

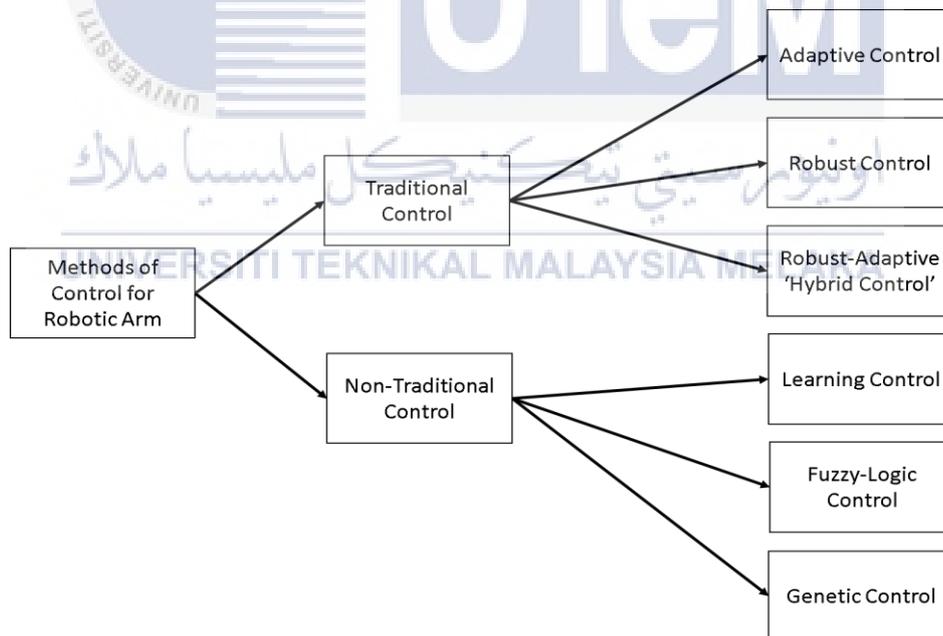
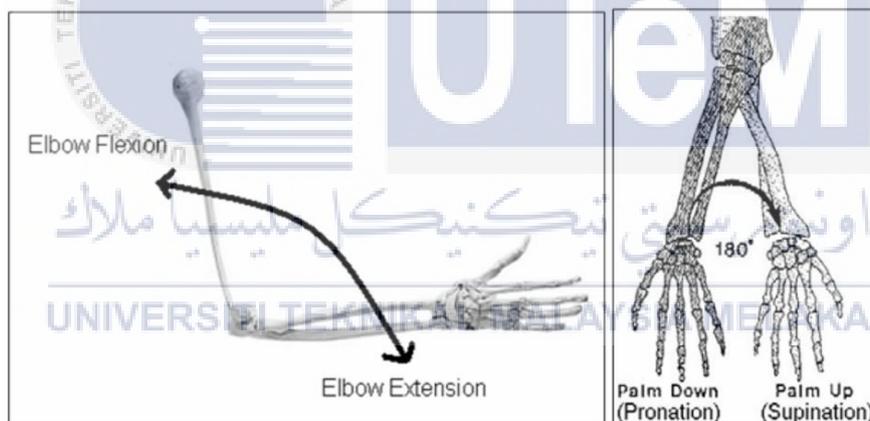


Figure 2.3: Type of Control Methods for Robotic Arm

C.S Lee and R.V. Gonzalez considered three different control strategies for a muscle-like actuated arm in two degrees of freedom for the flexion / extension of the elbow (f / e) and the pronation / supination of the forearm (p / s). Figure 2.4 shows the arm movement for elbow flexion and elbow extension pronation/supination [27].

Electromyograms (EMG) are used in their study to determine the control signal used to control the muscle cylinders. The first algorithm is a Fuzzy Controller with EMG signals and control input position error. The second algorithm is the Fuzzy-MA controller, which incorporated information on the moment arm into the existing Fuzzy logic control. The third algorithm is the conventional PID controller, which only worked on position and integration error [27].



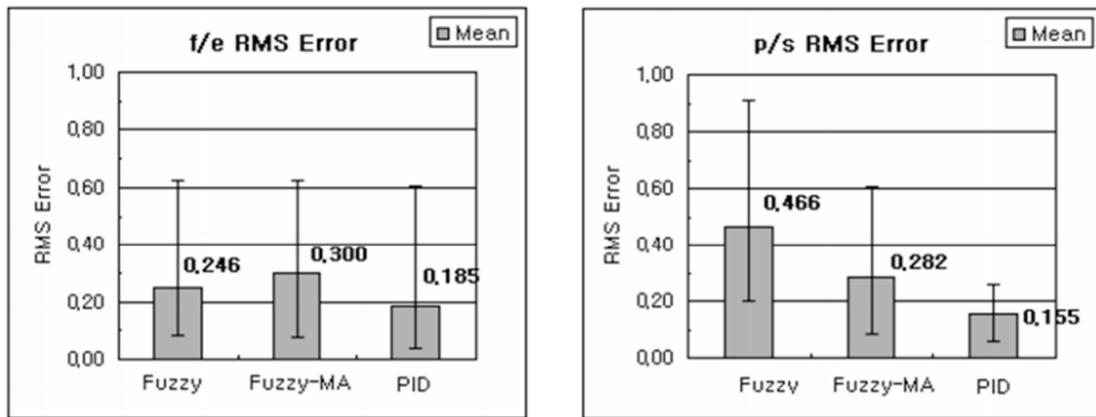
(a) Flexion/Extension

(b) Pronation (palm down)/  
Supination (Palm up)

Figure 2.4: Different Arm Movement [23]

The outcomes are examined as far as (a) Root-Mean-Square (RMS) error, (b) connection factor between the genuine and the coveted positions for every level of opportunity, and (c) mean absolute error (MAE) between the real and wanted positions for every level of opportunity. The consequences of investigation are appeared in Figure 2.5 which are about the RMS error utilizing three controllers. Figure 2.6 shows

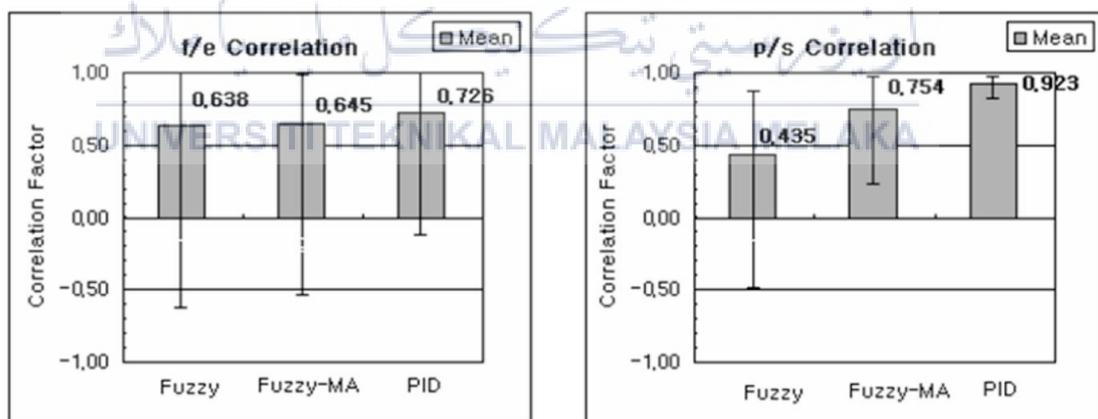
three examination of three controllers for relationship factor, though Figure 2.7 demonstrates the correlation of three controllers regarding mean absolute error.



(a) Flexion/Extension (f/e)  
RMS error

(b) Pronation/supination  
(p/s) RMS error

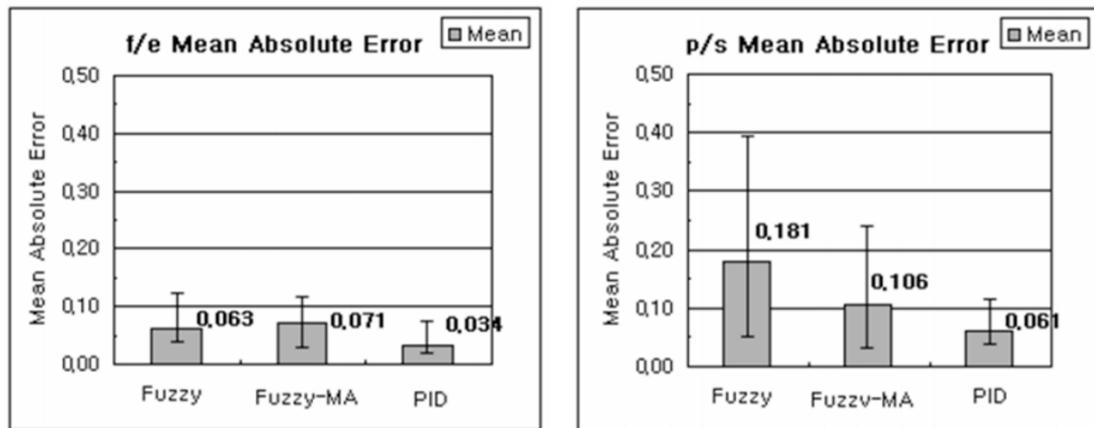
Figure 2.5: Comparison of Three Controllers for RMS Error. Bar Charts Represent Mean Values for Twenty Two Motions and Error Bars represent Maximum and Minimum Values [23].



(a) Flexion/Extension (f/e)  
correlation factor

(b) Pronation/supination  
(p/e) correlation factor

Figure 2.6: Comparison of Three Controllers for Correlation Factors [23]



(b) Flexion/Extension (f/e)  
mean absolute error

(a) Pronation/supination (p/s)  
mean absolute error

Figure 2.7: Comparison of Three Controllers for Mean Absolute Error [23]

The outcomes demonstrated that utilizing the moment arm data in the fuzzy controller recognizably enhanced p/s movement control. However, better control results were obtained in the PID controller compared to the fuzzy controllers. It is because PID controller quickly recovered the position error that was initially present. The fuzzy controller and Fuzzy-MA controller could not quickly recover the initial error.

This study showed that moment arms information incorporated into the fuzzy logic control technique improved the mechanical arm's response. However, a PID controller provides better accuracy than the EMG driven fuzzy based controllers.

Aron Pujana Arrese investigated the characteristics of pneumatic artificial muscles were investigated by the development of an experimental one-degree-of-freedom set - up based on Festo 's pneumatic muscles.. The experimental setup is non - linear and hard to properly control. An enhanced PID controller was designed as a reference. At the same time, a robust  $H_{\infty}$  controller and a sliding mode controller based on an observer have been developed and implemented. In addition, a position

controller was tuned to each pneumatic muscle based on an internal pressure loop. Comparisons are carried out using experimental results for each of the four position controllers [24].

Throughout the experiments, total three different areas of the displacement range are being tested. Figure 2.8 shows the experimental response to a ramp input of  $10^\circ$  and a slope of  $20^\circ/\text{s}$  applied in the upper displacement zone, where the mass at the tip is 3kg.

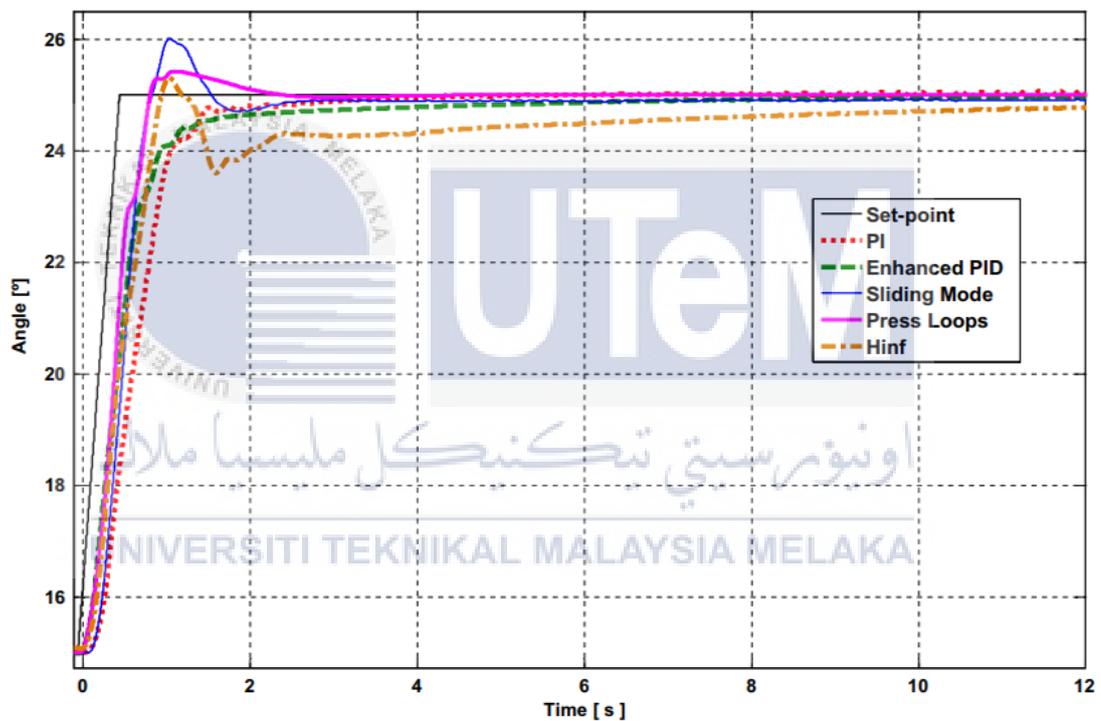


Figure 2.8: Experiment result in the upper displacement zone [24]

There is no overflow with the PI and the enhanced PID controls in the system. However, the time to settle is quite high. The sliding mode algorithm has the largest overflow. The  $H_\infty$  (Hinf) controller, on the other hand, takes a long time to eliminate the steady-state error. The pressure loop algorithm takes the arm to  $25^\circ$  before any other.

Based on the experimental analysis for four types of controllers, the results obtained with the classic PI, enhanced PID controller and  $H_\infty$  controller should be tuned in different operating areas. Despite the change in the pneumatic circuit, a control algorithm is designed based on independent pressure control of each muscle. The results achieved are best in terms of performance levels and in order to compensate for the prototype's non-linearity. The robustness study was satisfactory in comparison with the load.

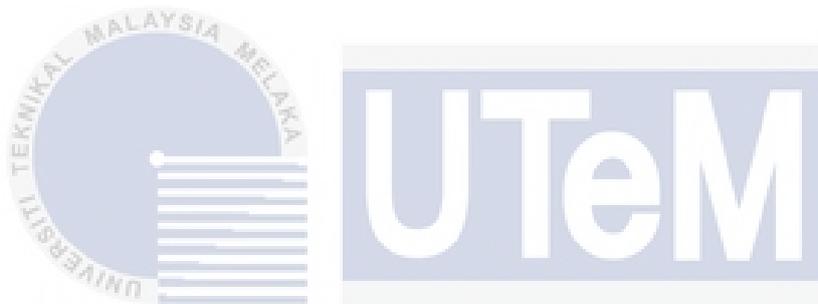
## 2.6 Summary

In this project, a robotic arm of two degree of freedom (2DOF) is constructed. After comparing multiple types of motor, DC geared motor is chosen to be used. Throughout the study on research papers, better understanding on the characteristics of each type of controller are obtained.

In order to implement this project, the PID controller is evaluated in terms of its steady-state error and settlement time. This means that more accurate motion control can be achieved by the developed method than the traditional robotic arm control method.

## CHAPTER 3

### METHODOLOGY



#### 3.1 Introduction

Research is carried out at the previous chapter. The types of controllers and components to be used are chosen. In this chapter, the procedures during research and experimental setup are introduced. Modeling of the system is done in this chapter.

#### 3.2 Research Methodology

Generally the research methodology is divided into two parts, namely project methodology and experiment methodology. Project methodology is the overall processes taken to complete this final year project, whereas experiment methodology is the procedures of the experimental setup. Appendix C shows the Gantt chart that summarized the research methodology of final year project

### 3.2.1 Project Methodology

The project research starts with literature review, which is the study on the other researcher's findings about this topic. This is helpful as it will give additional learning about the exploration theme. After the writing survey, kinds of segments to be utilized are picked. Experimental setup is conducted. At the same time, report is drafted whereas analysis is made based in the results obtained. Finally, the full report is written and the finalization of the overall project. Appendix A shows the flow chart methodology of the project research.

### 3.2.2 Experiment Methodology

The test setting started with the structure and production of the automated arm. At that point, the segments are determined and the type of controller to be used is chosen. Experimental installation is carried out inside the laboratory. Simulations are performed using the host computer simulation software (MATLAB). Analysis is made based on the results obtained. Appendix B shows the flow chart of the project experiment methodology.

### 3.3 Structure of Robotic Arm

The robotic arm is the biological arm's reference and is split into two parts, the first and second part portraying the arm and shoulder. The first and second connections are structured using Solidworks and the system drawing can be seen in Figure 3.1. The first link's length is 0.13 meter, whereas the second link's length is 0.158 meter. The automated arm is created utilizing 3D printing process and the material utilized is Acrylonitrile Butadiene Styrene (ABS). Figure 3.2 and Figure 3.3

shows the design of robotic arm from different perspectives including top view, bottom view and side view. Lastly, Figure 3.4 shows all the part after being assembled.

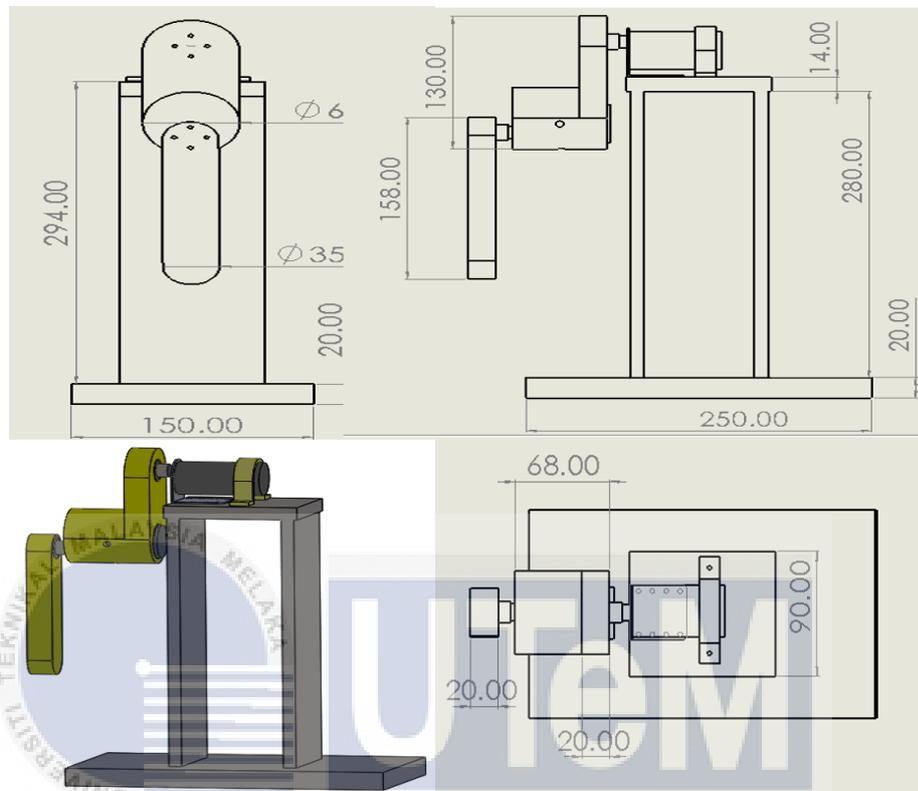


Figure 3.1: Drawing of the Robotic Arm Mechanism

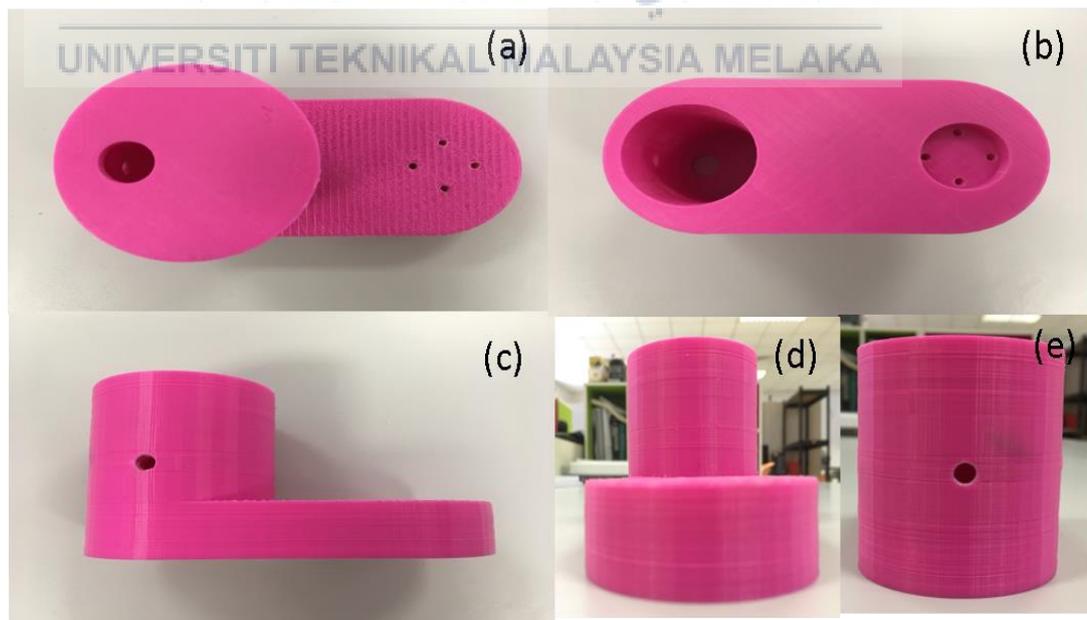


Figure 3.2: Structure of first link (a) Front view, (b) Back view, (c) Side view, (d) Top view, (e) Bottom view

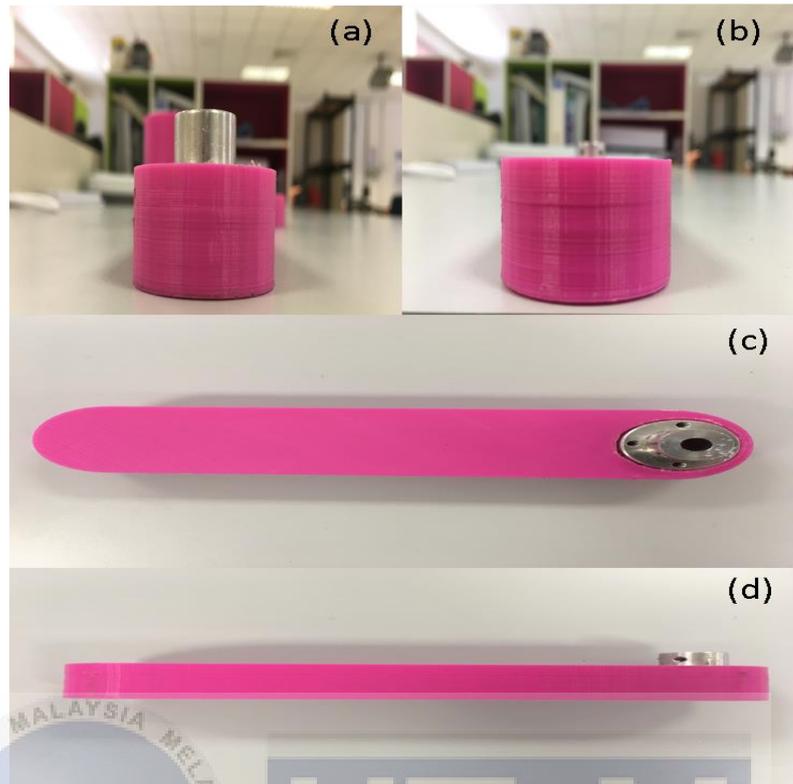


Figure 3.3: Structure of second link (a) Top view, (b) Bottom view, (c) Back view, (d) Side view

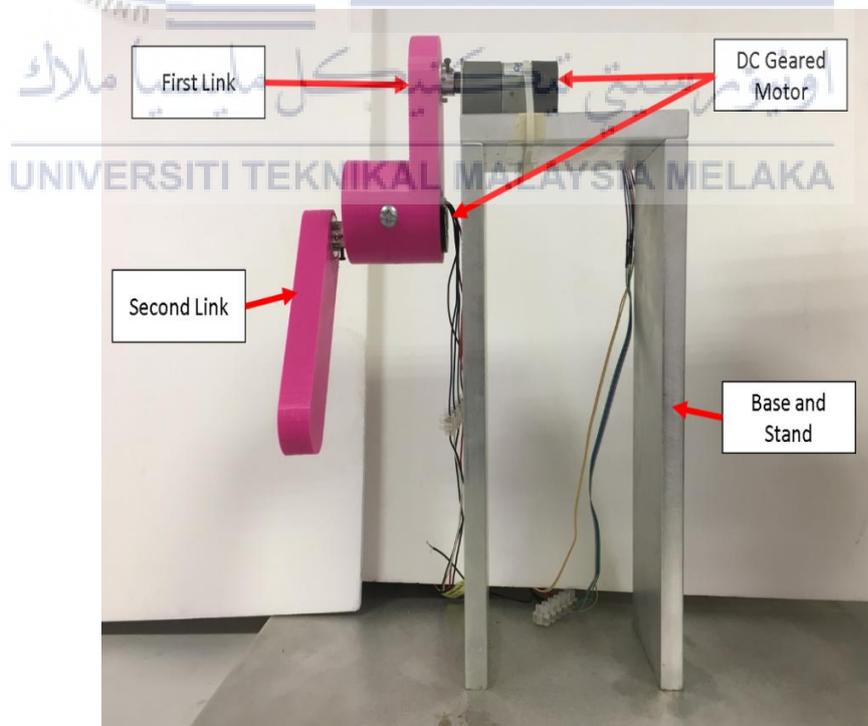


Figure 3.4: Structure of robotic after assemble all the parts

### 3.4 Equipment usage

The equipment segments utilized in this undertaking are appeared in this area. A DC geared motor is utilized to incite the automated arm. It has an appended Hall Effect encoder which is utilized to screen the development of arm.

#### 3.4.1 12V DC Geared Motor with Hall Effect Encoder by Cytron

Micro-Box will send a voltage output to the motor when the instruction is set inside the software. At that point, the motor will incite the robotic arm. The 5V Quadrature Hall Effect Encoder is built in to monitor the position and direction of rotation. Table 3.1 denotes the specification of this motor and the DC motor structure are shown in Figure 3.7.

Table 3.1: Specification of Motor

Parameter	Specification
Product Code	SPG30E - 300K
Operating Voltage	4.5V - 5.5V
Weight	160g
Output Power	1.1 Watt
Encoder Resolution	<ul style="list-style-type: none"> <li>• 3 pulses peer rear shaft revolution, single channel output</li> <li>• 810 counts peer main shaft revolution</li> </ul>
Gear Ratio	270:1
Rated Voltage	12VDC
Rated Speed	12 rpm
Rated Current	410 mA
Rated Torque	1.176 Nm



Figure 3.5: DC Geared Encoder Motor and its Detachable Cover

### 3.4.2 Micro-Box 2000/2000C (xPC Target Machine)

Micro-Box 2000/2000C used in this project is an element of Electro-Mechanical Engineering Control System (EMECS) by TeraSoft Inc. EMECS provide a platform for investigating a variety of control related problems such as system modeling, system identification, linear control, nonlinear control etc. In addition to hardware, Simulink blocks for the experiments are provided to help users in control design and simulation.

EMECS is made up of three components, the Micro-box 2000/2000C, driver circuit, and servo motor module, In this project, DC geared motor is used and thus, the servo motor provided in EMECS is not used. A Micro-Box module is associated between the motor and PC to go about as an interface between them. Table 3.2 lists out the specification of the Micro-Box module, whereas Figure 3.6 show its components.

Table 3.2: Micro-Box Components

Parameter	Specification
Power Supply	Min. 48W
Operating Voltage	9-36 VDC
Dimension	255(W) x 152(D) x 82(H) mm
Net Weight	2.0kg

### 3.4.3 Motor Driver Circuit

The driver circuit is used to drive the DC geared motor. Once the instructions from host computer are received, the driver circuit will actuate the movement of motor.

There are two cable connected to this circuit: the motor cable (yellow color) and the encoder cable (grey color). Both cables are required to connect to motor in order to actuate the robotic arm. Figure 3.6 shows the connection of each of cable that involved in the motor driver circuit.

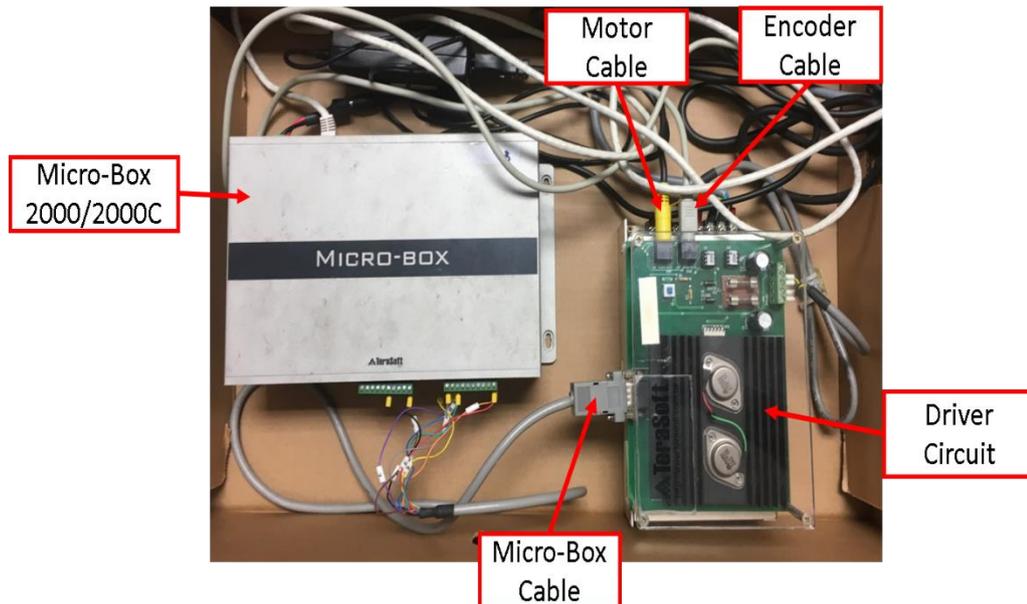


Figure 3.6: Components of Micro-Box Module

### 3.5 System Overview

The purpose of this system is to have the output angle of the motor follow the input angle which is being set by the user. First, the robotic arm is linked to the Micro-Box, while the Micro-Box device is the functionality between both the equipment (robotic arm) and the computer.

Micro-Box also operates as a unit for data collection that acquires data from the host computer and transfers the information to the motor driver circuit as a voltage output. The driver circuit then activates robotic arm's motion. Figure 3.7 indicates the relationship between the project elements.

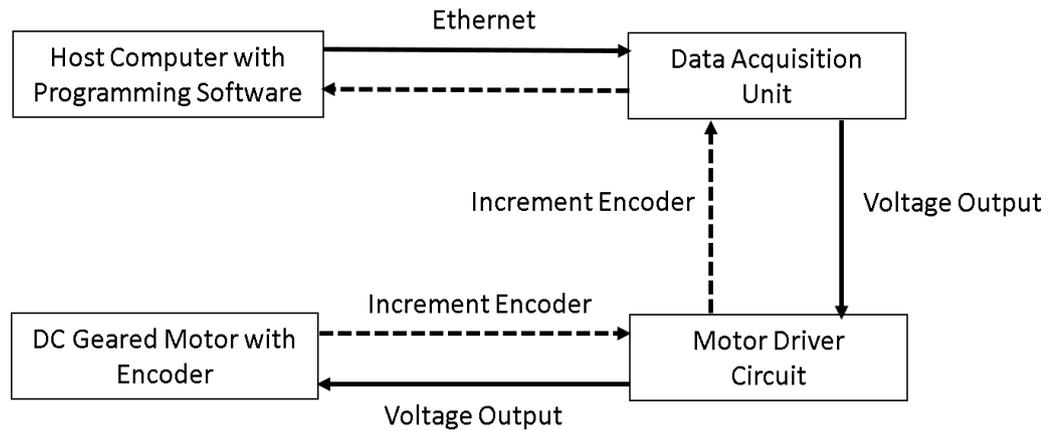


Figure 3.7: System Concept

### 3.6 Calibration of Encoder for DC Geared Motor

The calibration for the encoder of DC Geared Motor is needed so that the encoder can collect the reading as the desired value. The calibration of the encoder for DC Geared Motor should be done before the open loop test was conducted. The resolution of the encoder with DC Geared should be found where the conversion of the input voltage supply to angle of rotation together with the encoder reading is necessary. In order to determine the reliability and validity of the conversion, the value read by the encoder should be same as the arm robot position. Then, a gain was added into the Simulink block diagram and the experiment was run in order to observe the encoder reading and arm robot position which it will determine the reliability of the gain conversion.

### 3.7 Open Loop Control

Open loop system is a system in which the output does not affect the input due to lack of feedback system. The purpose of this open loop control is to identify the relationship between the reference input and output as well as to achieve the motor

transfer function. Open loop simulation is carried out using the system identification technique. System Identification is a MATLAB toolbox used to get mathematical models from input and output measured data. It helps identify the function of system transfer for further analysis. Typically, as shown in Equation 3.1, the simplified motor transfer function is a second order transfer function.

$$G(s) = \frac{As + B}{Cs^2 + Ds + E} \quad (3.1)$$

System identification process is performed in open loop conditions. The procedure is repeated 5 times for reproducibility testing. Mean and standard deviation values are obtained. The parameter value is chosen as the motor transmission feature closest to the average value and has the lowest standard deviation. This transfer function will then be substituted in the system for further evaluation. Figure 3.8 shows the block diagram for the open loop simulation.

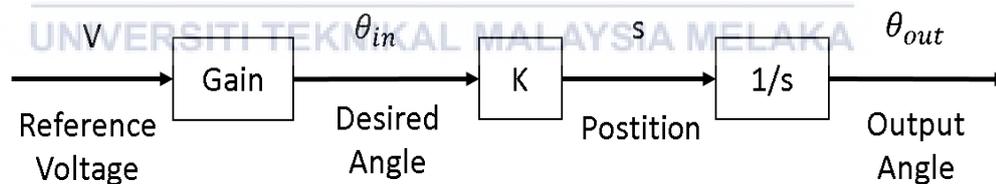


Figure 3.8: Block Diagram of Open Loop System

In this experiment setup, the experiment was conducted by drawing an open loop system model block diagram in the MATLAB Simulink as in Figure 3.9. The input of the model was a pulse generator which is able to rotate the motor in forward and backward within the sampling time by. A saturation block was added for safety purposes where it will saturate the input voltage when the input voltage exceeds the set

value. 9 samples of input supply with 2V until 10V were tested with the repeatability of at least three times for data collection.

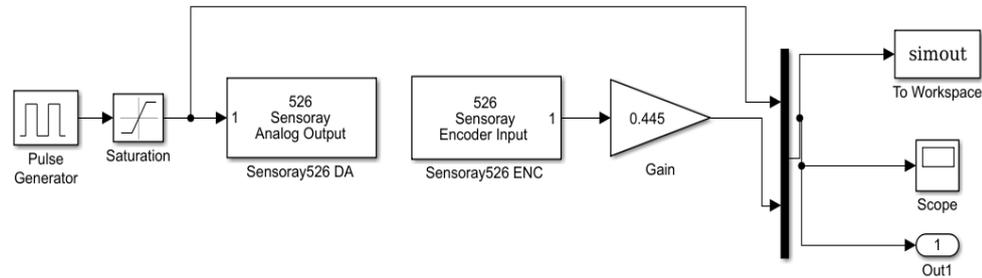


Figure 3.9: The Open Loop System Model Block Diagram

Sampling time for the open loop simulation is set to 0.001 seconds. This means that every millisecond is read. Input voltage is the only parameter that varies between 1 volt and 5 volt. The output data is in degree format. Table 3.3 shows in the open loop simulation parameters and each of their numeric values.

Table 3.3: Open Loop Simulation Parameters

Parameter	Numerical Value
Input Voltage (v)	1-5 Volt
Simulation time (t)	1s
Delay (t)	0s
Sampling time (t)	1ms
Input type	Step input

After simulated the open loop system, there will be five transfer function generated. Then each of the transfer function will be used to run the real time simulation with the experimental output to obtain the performance characteristic. Figure 3.10 is the block diagram to obtain the real time simulation and experimental performance characteristic of the robotic arm. The transfer function that provide the

characteristic from the simulation which is similar to the experimental result will be used for the further closed loop system and analysis.

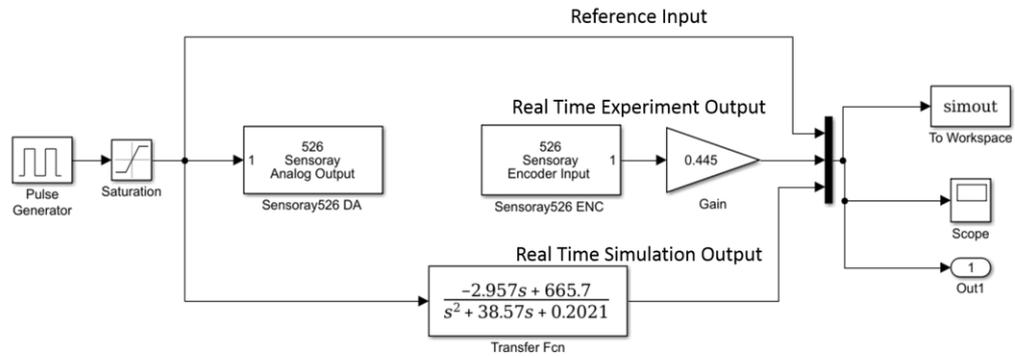


Figure 3.10: Real Time Simulation and Experimental Block Diagram

### 3.8 System Identification Tools

The system identification tools will compare both the input and output of the model and then generating a transfer function based on the relationship between the input and output data collected. Figure 3.11 shows the System Identification Tools configuration from MATLAB. After successfully running the open loop system, the output of the open loop system will then save data file. In order to convert and generating the matrices produced to obtain a transfer function the data file of the system need to be imported into the System Identification Tools and a transfer function related to the model will be generated.

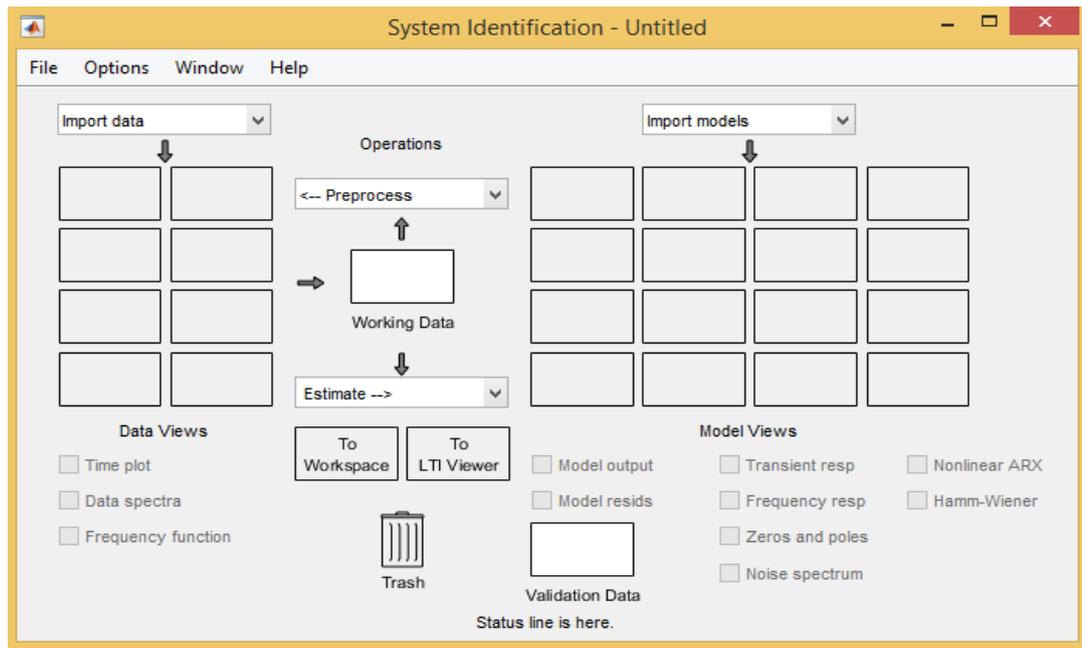


Figure 3.11: System Identification Tools.

### 3.9 Development of Closed Loop System

A closed loop system was developed to provide the feedback to compare both the output and input of the system. The feedback of the system will improve the system performance by reduce the error when there was difference between input and output. There were two types of closed loop system which were uncompensated closed loop system and compensated closed loop system.

#### 3.9.1 Development of Uncompensated Closed Loop System

An uncompensated closed loop system has a feedback system provided form the output and feed back to the summing and being compare with input reference. The difference between the output and the input reference is the error occurred and the feedback will reduce the error in the system. However, there will be no controller in the system. The block diagram of the uncompensated closed loop system for the robotic arm is shown in the Figure 3.12.

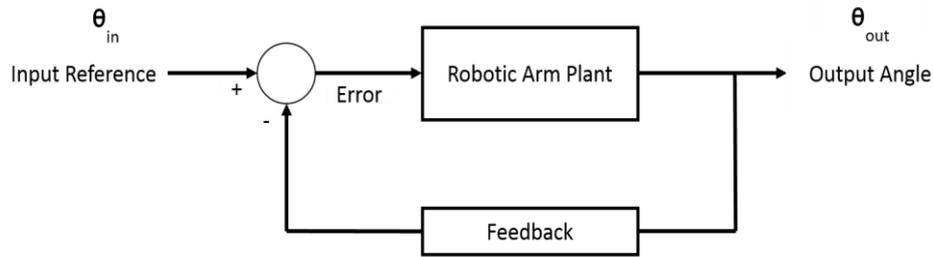


Figure 3.12: The Block Diagram of the Uncompensated Closed Loop System for the Robotic Arm

Uncompensated closed loop system is where a closed loop system is developed without any controller. A negative feedback is directly feedback to the input from the output. The block diagram of the uncompensated closed loop system is shown in Figure 3.13. The system performance of the robotic hand for both the real time simulation and experimental will then analyzed. Since it is an uncompensated closed loop system, there will be an error occurred in the system although the error had reduce implement the closed loop system.

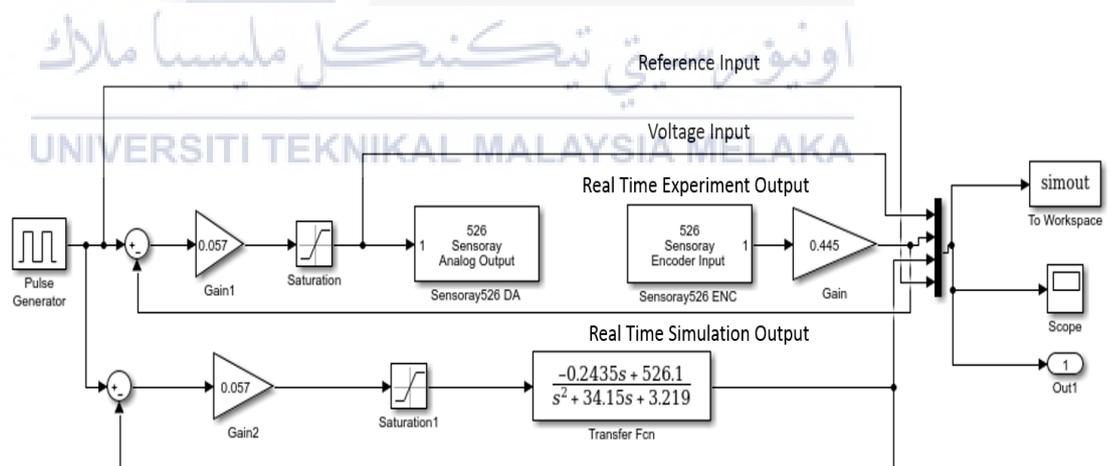


Figure 3.13: The Uncompensated Closed System Block Diagram in Simulink

### 3.9.2 Controller Design

Once the system is set up experimentally, it is crucial to design a controller to control the parameters in the system. In this project, the parameter that is being controlled is output angle of the motor. Thus, the controller must be able to achieve precise motion control and to minimize steady-state error.

### 3.9.3 Design and Development of Compensated Closed Loop System with Proportional-Integral-Derivative (PID) Controller

Compensated closed loop system is quite similar with the uncompensated closed loop system which also has the feedback in the system. However, the compensated closed loop system has a controller in the system where it will be further used for improving the output variables in order to reduce the error in the system. A compensated closed loop system provided a better performance in term of accuracy and precision due to the error is being reduce and the system performance can be controlled. A PID controller uses proportional, integral and derivative functions to control the input signal before it is sent to the plant. For driving the robotic arm, the angular position of the DC motor can be controlled. Due to its simplicity in architecture, the PID controller has remained the most commonly used controller in practically all industrial control applications. Figure 3.14 shows the PID controller block diagram and Figure 3.15 shows the whole compensated closed loop system designed by using Simulink.

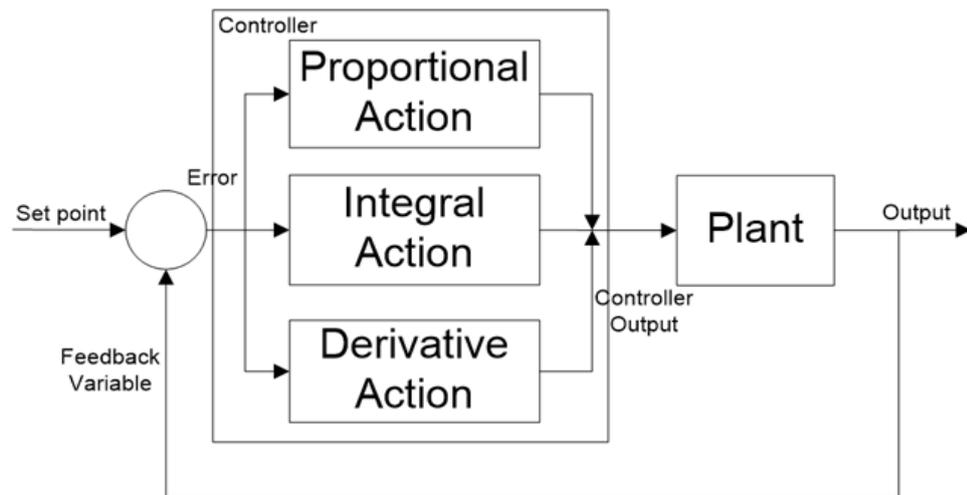


Figure 3.14: Block diagram of a typical PID controller

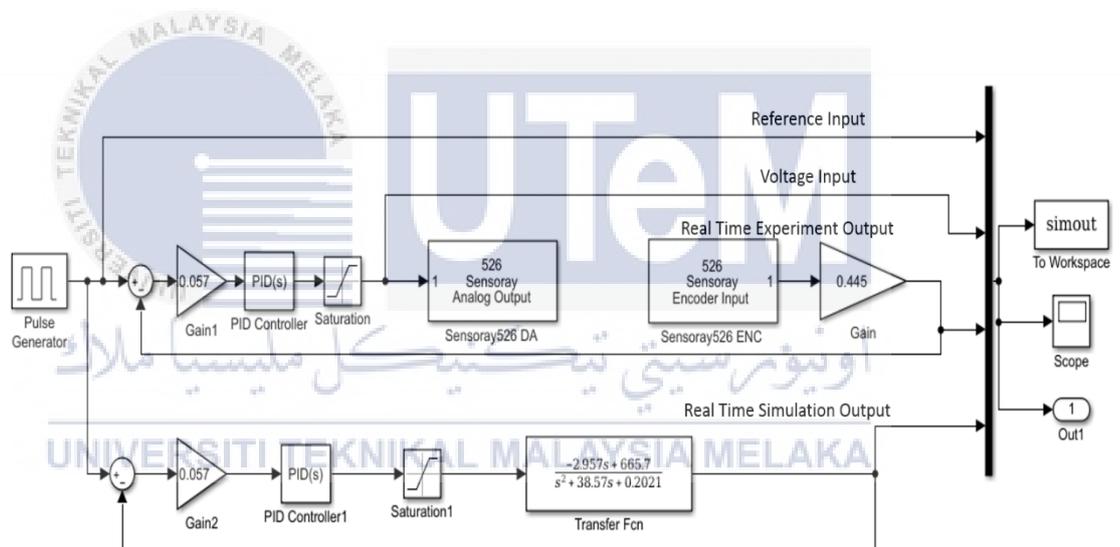


Figure 3.15: Compensated Closed Loop System with PID Controller using MATLAB Simulink

In order to design a PID controller that was suitable for the system, there were three parameters need to be adjust which were the Proportional parameter (P), Integral parameter (I) and Derivative parameter (D). The proportional parameter was the first parameter that need to be adjusted while letting the integral and derivative parameter with zero. The proportional parameter was adjusted until the rise time reach the desire value then the proportional parameter was known. Then the integral parameter will be

adjust while letting the proportional parameter same as the desired value and derivative parameter remain as zero. The adjusting of the integral parameter was adjusted until the error in the system is eliminated. Lastly, the derivative parameter will be adjusted until the transient response provides the best system performance and all the three parameters found will improve the point-to-point positioning of the robotic arm for this research.

### 3.9.4 Tuning Methods

In this project, both trial and error method and Ziegler-Nichols method are implemented to compare whether which method is useful in this case.

### 3.9.5 Trial and Error Method

In practice control, engineers often use trial and error for the tuning process. It is a relatively easier way to tune the controller. However, it can take a lot of time and does not ensure that the system performance is satisfactory. Table 3.4 shows the effects of manipulating the parameters of PID controller.

Table 3.4: Parameters of Transient Response and the Effects Caused by Manipulating (P), (I), and (D) values

Response	Overshoot	Settling Time	Rise Time	Stability	Transient Response	Steady-State Error
P	Increase	Small change	Decrease	Small change	Small change	Decrease
I	Increase	Increase	Decrease	Decrease	Degrade	Eliminate
D	Decrease	Decrease	Small change	Increase	Improve	No change

### 3.9.6 Ziegler-Nichols Method

One of the design methods is based on the process dynamics which can be acquired experimentally. Ziegler-Nichols method is one of the common methods used to determine the process dynamics based on the step response of the system.

In this project, Ziegler-Nichols frequency-domain method is implemented. This tuning method is to produce good values for three PID gain parameters which are Proportional Gain  $K_p$ , Integrator Time Constant  $T_i$  and Derivative Time Constant  $T_d$ . Frequency domain method is based on the frequencies response of the system. The parameters of the controller have been given formula in terms of Ultimate Gain  $K_u$  and Ultimate Period  $T_u$ . A controller is connected in the system. Parameters are set so that the control action is proportional. The gain is increased slowly until the system starts to oscillate. The gain when oscillation occurs is  $K_u$  whereas the period of oscillation is  $T_u$ . Table 3.5 shows the controller parameters obtained from Ziegler-Nichols frequency response method.

Table 3.5: Controller Parameters of Ziegler-Nichols Step Response Method

Controller	$K_p$	$T_i$	$T_d$
P	$0.5K_u$	-	-
PI	$0.4K_u$	$0.8T_u$	-
PID	$0.6K_u$	$0.5T_u$	$0.12T_u$

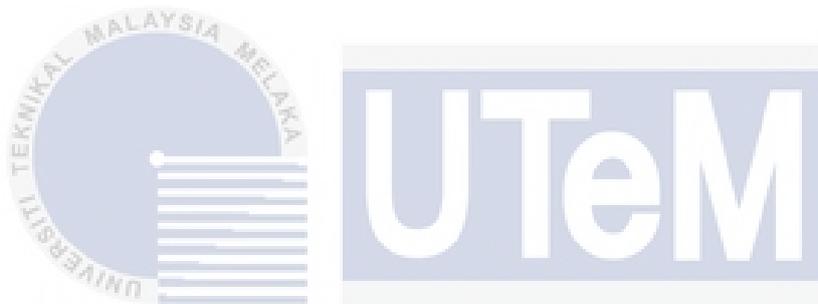
The equation of an ideal PID controller is as follows,

$$u(t) = K_p e(t) + K_i \int_0^t e(t) d\tau + K_d \frac{de}{dt} \quad (3.2)$$

$$u(t) = K_p e(t) + \frac{1}{T_i} \int_0^t e(t) d\tau + T_d \frac{de(t)}{dt} \quad (3.3)$$

## CHAPTER 4

### RESULTS AND DISCUSSION



#### 4.1 Introduction

The methodology of the project is discussed in the previous chapter. The expected results of the simulation are discussed in this section. Simulink diagrams in MATLAB are used to perform both the open-loop and closed-loop control systems procedure.

The open-loop simulation is the first will be carried out to observe the open-loop characteristics of the system. Closed-loop simulation is then will be carried out (with and without using controller) for different input angles. The angles that will be tested are  $15^\circ$ ,  $30^\circ$  and  $60^\circ$ . For the same batch of input angles, PID is then introduced to examine changes to the system. Figure 4.1 shows the structure of this chapter and the topics which are presented in this chapter.

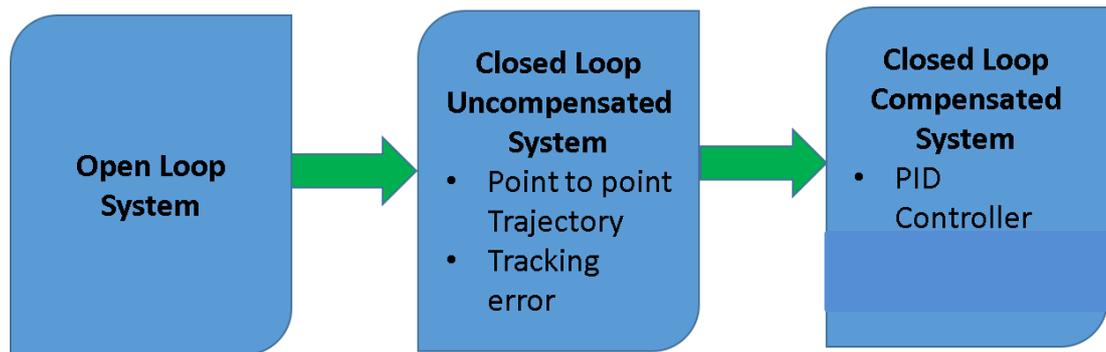


Figure 4.1: Simulation and experiment flow

## 4.2 Open-Loop Test

In this section, the results of the open-loop system will be discussed. The objectives of carrying out the open-loop system are:

1. To get the motor transfer function
2. To observe open-loop system characteristics.

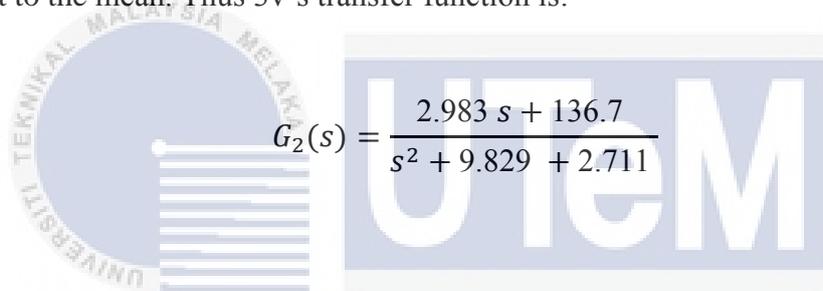
Several voltages (2V to 10V) will be applied to the system. One volt is not applied due to the torque produced was not enough to drive the DC Geared Motor. The aim of differing the voltage is to examine the value of the experiment and the simulation. The voltage value is then chosen where the experimental value is closest to the value of the simulation.

Table 4.1 indicates the DC motor model system identification variable once the voltage input is 2 volts. It will run up to 5 times and the closest value to the mean value will be selected to formulate the transfer function. After that, repeat the previous step by changing the input voltage to 3 volts until 10 volts. The transfer function chosen for 2V is:

$$G_1(s) = \frac{-4.459s + 901.5}{s^2 + 92.41 + 5.985e-09} \quad (4.1)$$

Figure 4.2 shows the graph of input voltage and output voltage of the open-loop test with the transfer function applied to the system  $G_1(s)$ . From the graph, it is observed that when 2 volts is applied to the system, the maximum output angle achieved is around 20°.

Table 4.2 indicates the DC motor model system identification variable once the voltage input is 3 Volt. The results show that the second transfer function is the nearest to the mean. Thus 3v's transfer function is:



$$G_2(s) = \frac{2.983s + 136.7}{s^2 + 9.829 + 2.711} \quad (4.2)$$

Figure 4.3 shows the graph of input voltage and output voltage of the open-loop test with the transfer function applied to the system  $G_2(s)$ . From the graph, it is observed that when 3 volts is applied to the system, the maximum output angle achieved is around 37°.

Table 4.3 indicates the DC motor model system identification variable once the voltage input is 4 Volts. The results show that the first transfer function is the nearest to the mean. Thus 4v's transfer function is:

$$G_3(s) = \frac{0.9251s + 314.9}{s^2 + 21.78 + 3.031} \quad (4.3)$$

Figure 4.4 shows the graph of input voltage and output voltage of the open loop test with the transfer function applied to the system  $G_3(s)$ . From the graph, it is observed that when 4 volt is applied to the system, the maximum output angle achieved is around  $54^\circ$ .

Table 4.4 indicates the DC motor model system identification variable once the voltage input is 5 volts. The results show that the fifth transfer function is the nearest to the mean. Thus 5v's transfer function is:

$$G_4(s) = \frac{-2.935s + 730.8}{s^2 + 39.34 + 0.2728} \quad (4.4)$$

Figure 4.5 shows the graph of input voltage and output voltage of the open loop test with the transfer function applied to the system  $G_4(s)$ . From the graph, it is observed that when 5-volt is applied to the system, the maximum output angle achieved is around  $71.5^\circ$ .

Table 4.5 indicates the DC motor model system identification variable once the voltage input is 6 Volt. The results show that the third transfer function is the nearest to the mean. Thus 6v's transfer function is:

$$G_5(s) = \frac{-0.4903s + 562.1}{s^2 + 34.62 + 3.369} \quad (4.5)$$

Figure 4.6 show the graph of input voltage and output voltage of the open loop test with the transfer function applied to the system  $G_5(s)$ . From the graph, it is

observed that when 6 volt is applied to the system, the maximum output angle achieved is around 90°.

Table 4.6 indicates the DC motor model system identification variable once the voltage input is 7 Volt. The results show that the fifth transfer function is the nearest to the mean. Thus 7v's transfer function is:

$$G_6(s) = \frac{1.008 s + 423.8}{s^2 + 29.16 s + 3.452} \quad (4.6)$$

Figure 4.7 show the graph of input voltage and output voltage of the open loop test with the transfer function applied to the system  $G_6(s)$ . From the graph, it is observed that when 7 volt is applied to the system, the maximum output angle achieved is around 93°. At this level, the angle slightly increase due to the motor has reached its saturation level. However, the value of voltage still needs to be increase until 10v in order to validate whether which voltage value can give the best result.

Table 4.7 indicates the DC motor model system identification variable once the voltage input is 8 Volt. The results show that the first transfer function is the nearest to the mean. Thus 8v's transfer function is:

$$G_7(s) = \frac{-0.2869 s + 406.1}{s^2 + 31.89 s + 3.535} \quad (4.7)$$

Figure 4.8 show the graph of input voltage and output voltage of the open loop test with the transfer function applied to the system  $G_7(s)$ . From the graph, it is

observed that when 8 volt is applied to the system, the maximum output angle achieved is 93°.

Table 4.8 indicates the DC motor model system identification variable once the voltage input is 9 Volt. The results show that the first transfer function is the nearest to the mean. Thus 9v's transfer function is:

$$G_8(s) = \frac{1.389s + 345.3}{s^2 + 30.67 + 3.354} \quad (4.8)$$

Figure 4.9 show the graph of input voltage and output voltage of the open loop test with the transfer function applied to the system  $G_8(s)$ . From the graph, it is observed that when 9 volt is applied to the system, the maximum output angle achieved is 93°.

Table 4.9 indicates the DC motor model system identification variable once the voltage input is 10 Volt. The results show that the first transfer function is the nearest to the mean. Thus 10v's transfer function is:

$$G_9(s) = \frac{-0.1239 s + 332.1}{s^2 + 32.87 + 3.649} \quad (4.9)$$

Figure 4.10 show the graph of input voltage and output voltage of the open loop test with the transfer function applied to the system  $G_9(s)$ . From the graph, it is observed that when 10 volt is applied to the system, the maximum output angle achieved is 94°. The errors between the real time signal and simulated signal of all the voltage applied are shown in the figure 4.11.

Table 4.1: System Identification Results for DC Motor ( $V_{in} = 2V$ )

Transfer Function	A	B	C	D	E
1	-4.459	901.5	1	92.41	5.985e-09
2	1.844	367.1	1	37.3	3.321e-10
3	-1.337	1248	1	135.7	4.579e-10
4	-10.83	603.8	1	65.44	1.118e-08
5	-10.4	605	1	65.66	5.079e-11
Mean	-5.18075	745.1	1	79.302	3.6e-09
Std. Dev.	5.560864618	339.06012	0	37.06407047	4.90646e-09

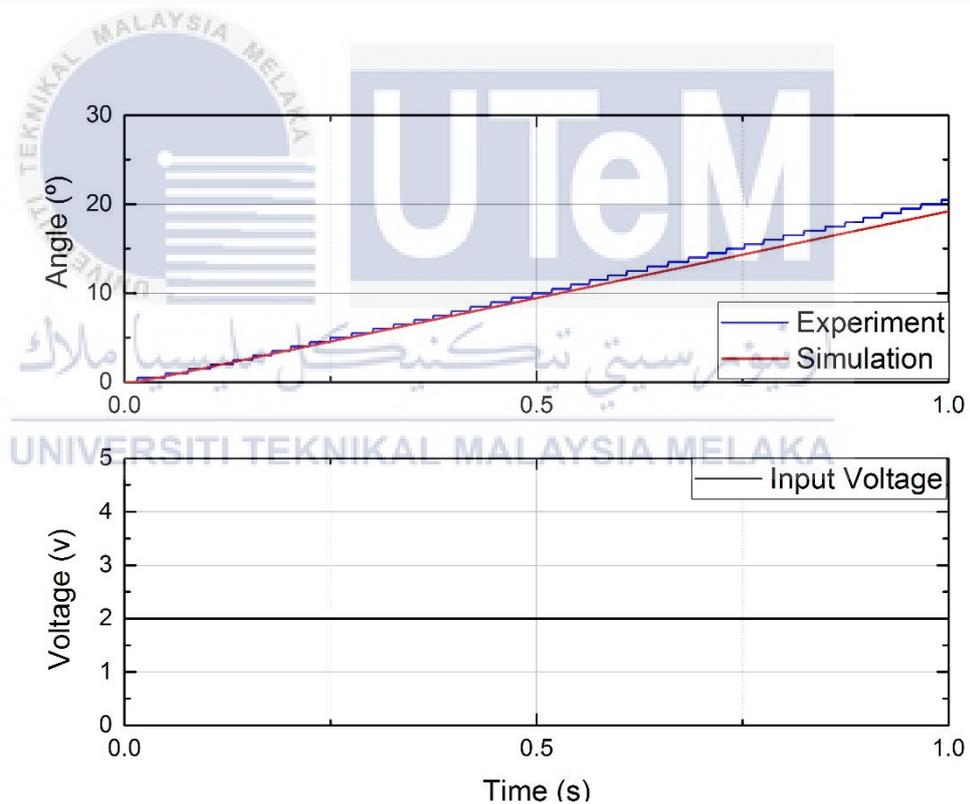


Figure 4.2: Input voltage and Output Angle Versus Times Graph, (2V)

Table 4.2: System Identification Results for DC Motor ( $V_{in} = 3V$ )

Transfer Function	A	B	C	D	E
1	4.333	47.62	1	3.422	1.156
2	2.983	136.7	1	9.829	2.711
3	7.981	63.33	1	4.385	1.408
4	1.097	251.2	1	18.16	4.493
5	7.123	94.59	1	6.661	1.949
Mean	4.796	118.7	1	8.4914	2.3434
Std. Dev.	2.8592053	81.52	0	5.94102	1.341301

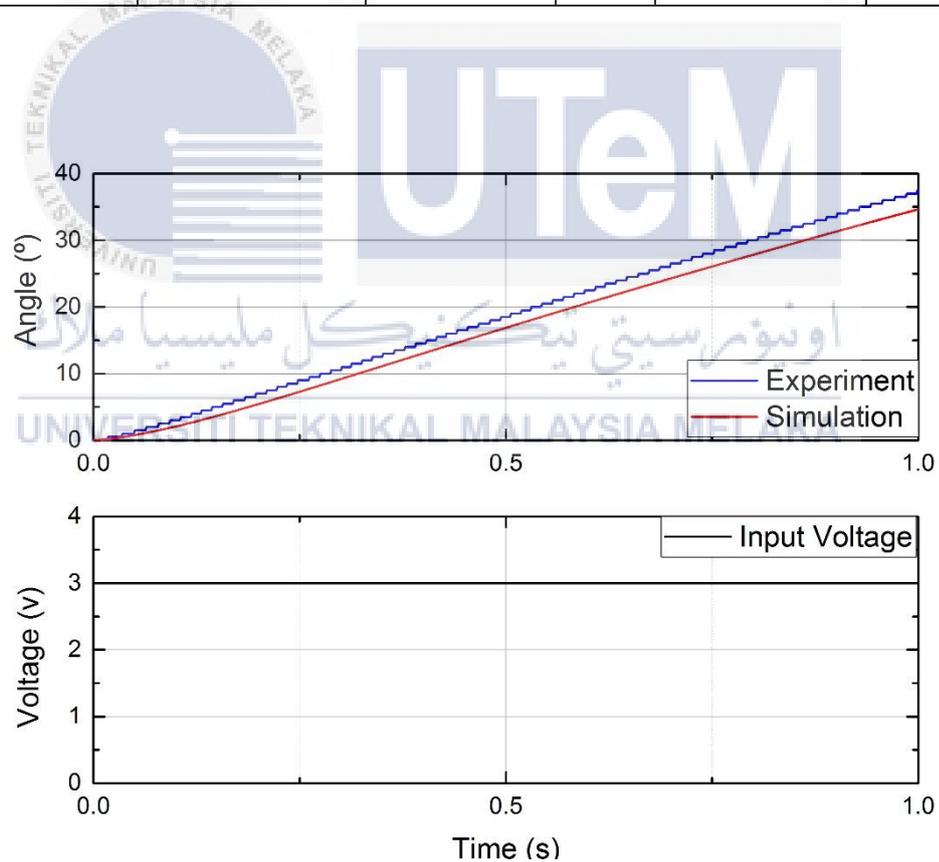


Figure 4.3: Input voltage and Output Angle Versus Times Graph, (3V)

Table 4.3 System Identification Results for DC Motor ( $V_{in} = 4V$ )

Transfer Function	A	B	C	D	E
1	0.9251	314.9	1	21.78	3.031
2	4.862	262.8	1	17.92	2.526
3	0.5189	319.8	1	21.76	3.027
4	0.8709	383.2	1	26.02	3.436
5	4.317	349.4	1	23.75	3.173
Mean	2.6422	326	1	22.246	3.0386
Std. Dev.	2.105773	44.65	0	2.98633	0.331263

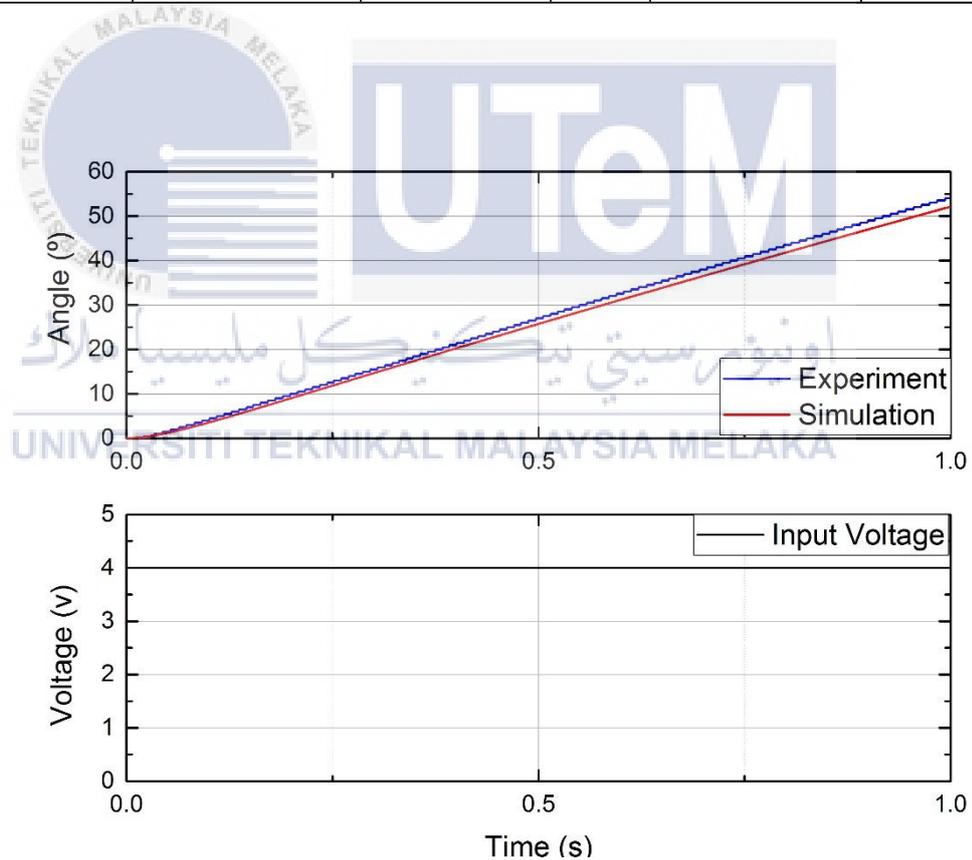


Figure 4.4: Input voltage and Output Angle Versus Times Graph, (4V)

Table 4.4: System Identification Results for DC Motor ( $V_{in} = 5V$ )

Transfer Function	A	B	C	D	E
1	-2.763	646.8	1	37.5	0.1581
2	-2.795	660.2	1	38.31	0.08069
3	-3.174	671.7	1	38.86	0.2141
4	-2.879	655	1	37.98	0.1448
5	-2.957	665.7	1	38.57	0.2021
Mean	-0.127625	511.1	1	33.276	3.2006
Std. Dev.	1.5561501	28.64	0	1.6888	0.366178

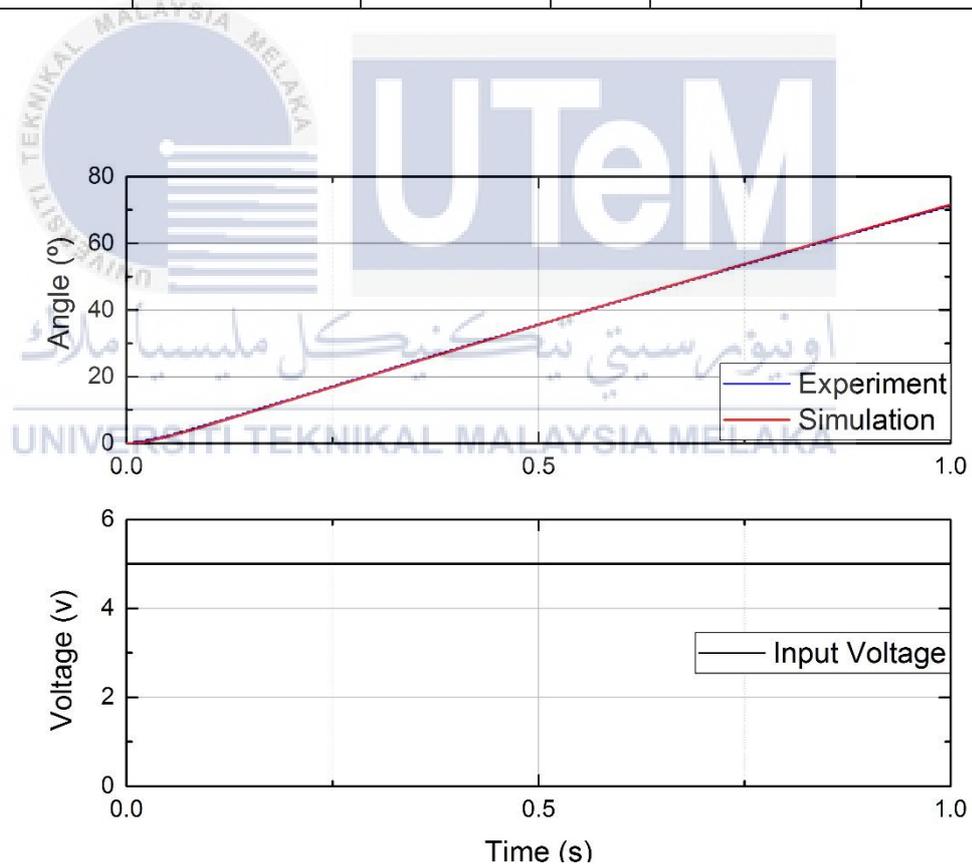


Figure 4.5: Input voltage and Output Angle Versus Times Graph, (5V)

Table 4.5: System Identification Results for DC Motor ( $V_{in} = 6V$ )

Transfer Function	A	B	C	D	E
1	-1.717	547.7	1	33.67	3.433
2	-0.8889	566	1	34.89	3.395
3	-0.4903	562.1	1	34.62	3.369
4	-1.023	559.6	1	34.55	3.281
5	1.674	517.1	1	31.9	3.076
Mean	-0.18205	550.5	1	33.926	3.3108
Std. Dev.	1.2875976	19.88	0	1.22173	0.142679

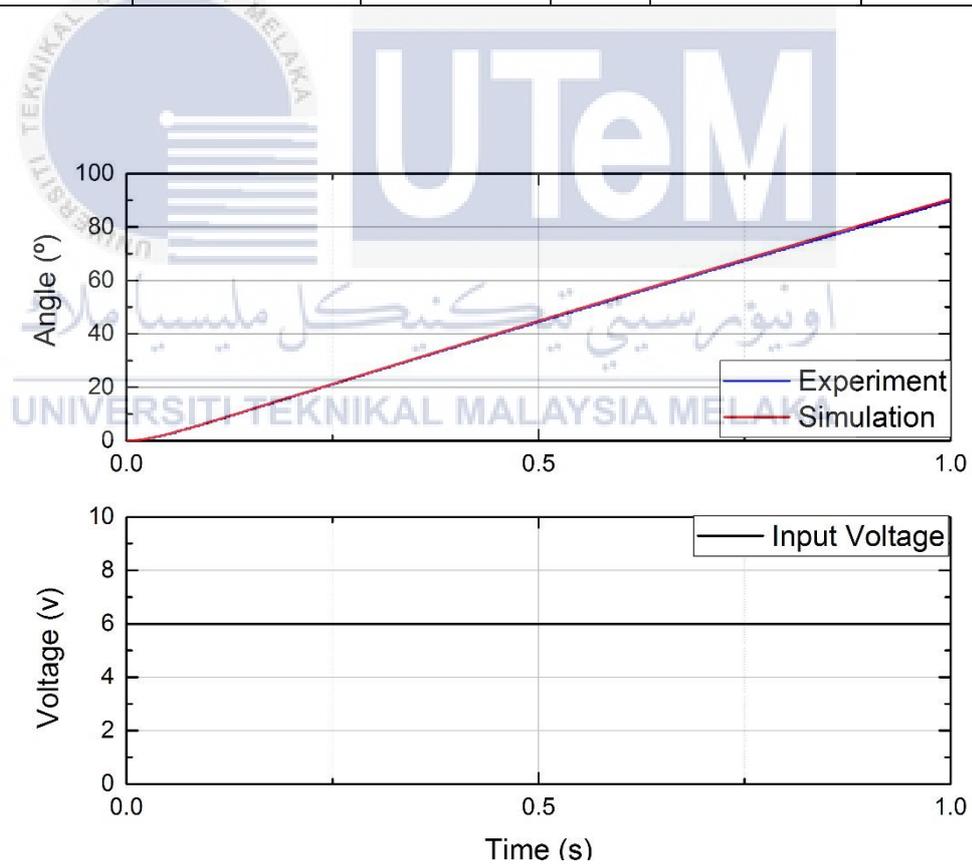


Figure 4.6: Input voltage and Output Angle Versus Times Graph, (6V)

Table 4.6: System Identification Results for DC Motor ( $V_{in} = 7V$ )

Transfer Function	A	B	C	D	E
1	0.751	442.4	1	30.27	3.77
2	3.265	423.5	1	28.99	3.622
3	-0.2122	479.7	1	33.11	3.879
4	0.9338	412.4	1	28.28	3.448
5	1.008	423.8	1	29.16	3.452
Mean	1.24865	436.4	1	29.962	3.6342
Std. Dev.	1.2801286	26.51	0	1.89881	0.191301

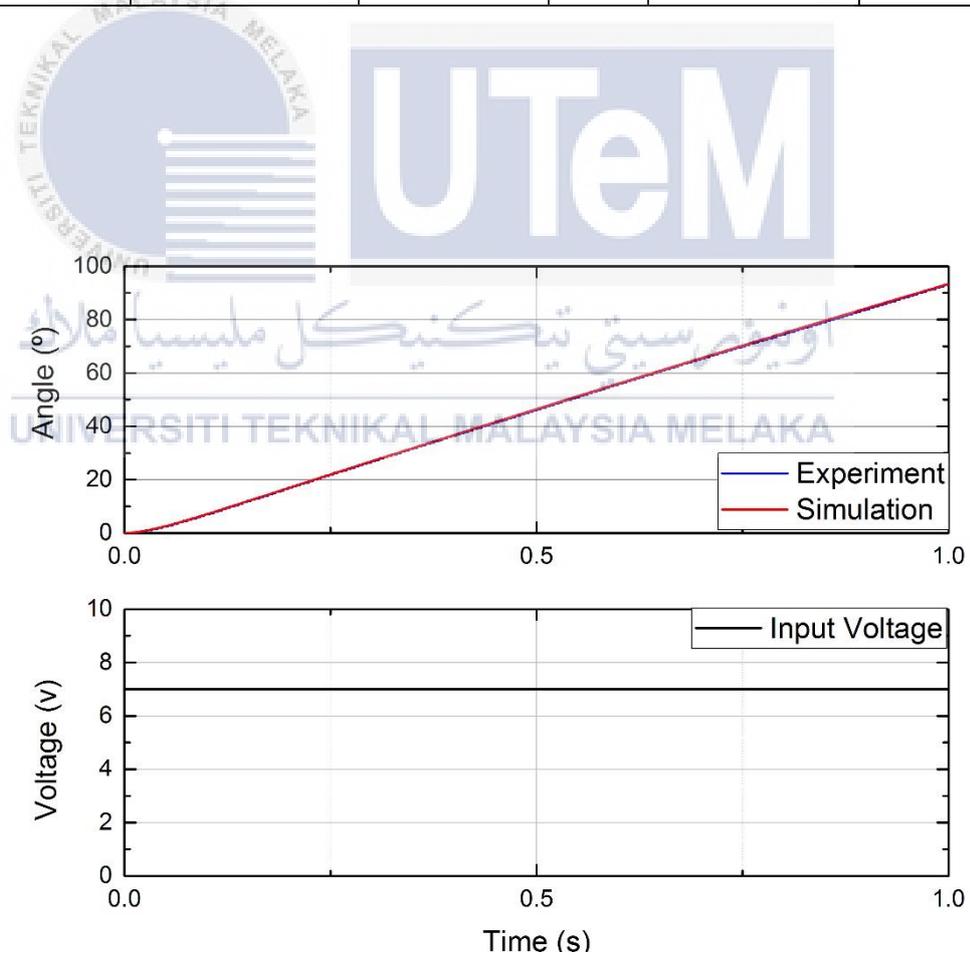


Figure 4.7: Input voltage and Output Angle Versus Times Graph, (7V)

Table 4.7: System Identification Results for DC Motor ( $V_{in} = 8V$ )

Transfer Function	A	B	C	D	E
1	-0.2869	406.1	1	31.89	3.535
2	0.7266	397.7	1	31.12	3.641
3	-1.014	429.7	1	33.73	3.686
4	-0.5585	399.8	1	31.09	3.869
5	1.76	400.9	1	31.44	3.552
Mean	0.228525	406.8	1	31.854	3.6566
Std. Dev.	1.1145965	13.15	0	1.0971	0.134117

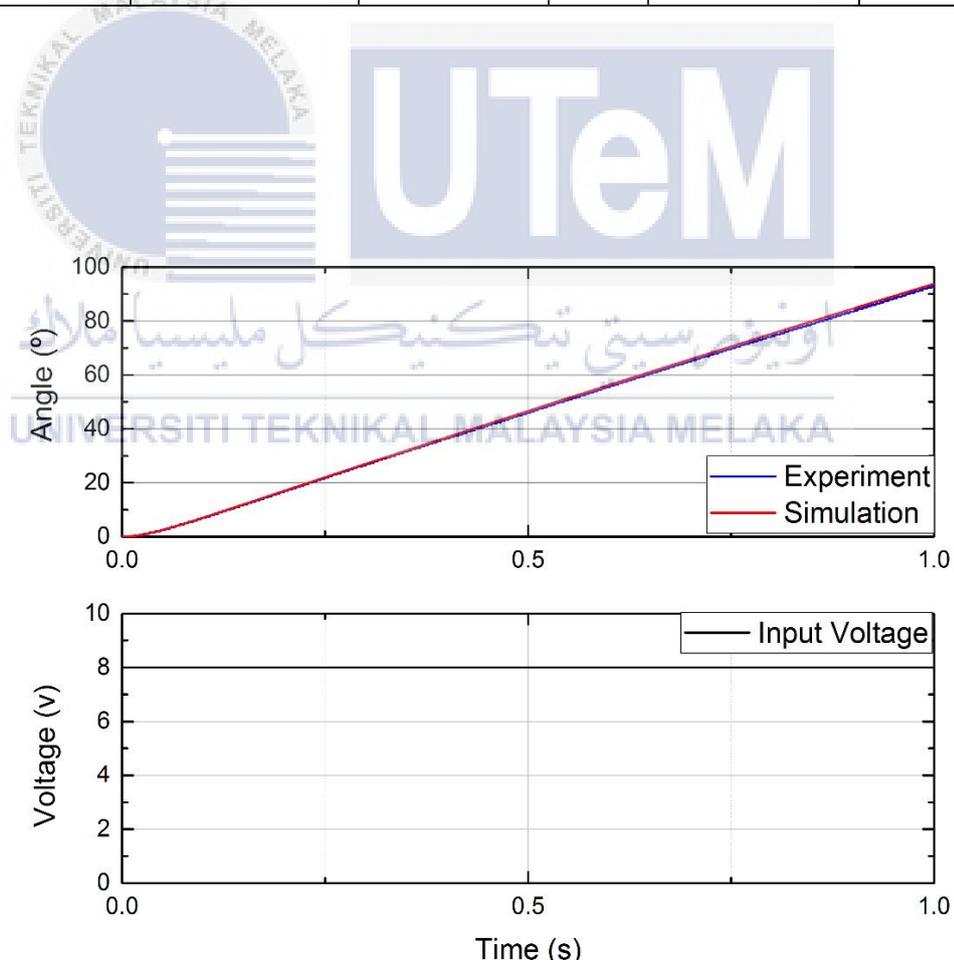


Figure 4.8: Input voltage and Output Angle Versus Times Graph, (8V)

Table 4.8: System Identification Results for DC Motor ( $V_{in} = 9V$ )

Transfer Function	A	B	C	D	E
1	1.389	345.3	1	30.67	3.354
2	-0.5023	363.1	1	32.05	3.756
3	1.797	330.5	1	29.23	3.487
4	-1.085	368.3	1	32.8	3.455
5	1.849	328	1	29.02	3.23
Mean	0.514675	347	1	30.754	3.4564
Std. Dev.	1.3810806	18.36	0	1.67348	0.195219

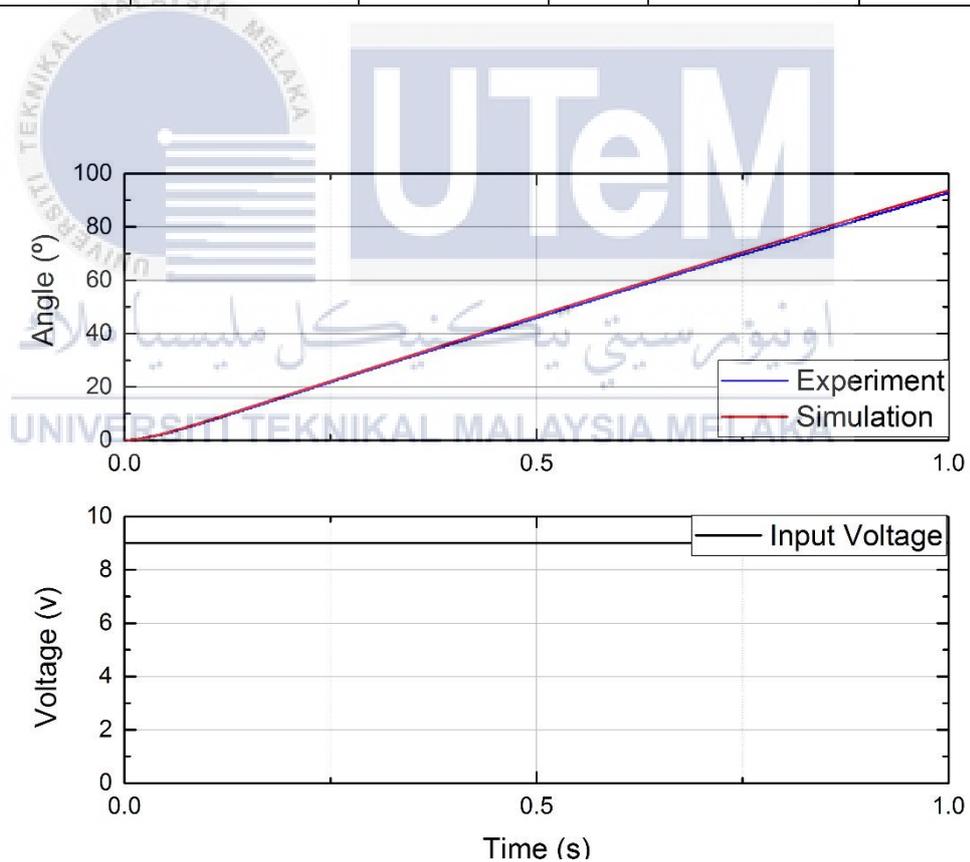


Figure 4.9: Input voltage and Output Angle Versus Times Graph, (9V)

Table 4.9 System Identification Results for DC Motor ( $V_{in} = 10V$ )

Transfer Function	A	B	C	D	E
1	-0.1239	332.1	1	32.87	3.649
2	-0.4614	340.4	1	33.59	3.784
3	1.356	312.6	1	30.8	3.437
4	-0.7183	351.5	1	34.89	3.658
5	-0.1493	296	1	29.02	3.7
Mean	0.00675	326.5	1	32.234	3.6456
Std. Dev.	0.8067442	22.21	0	2.32829	0.128251

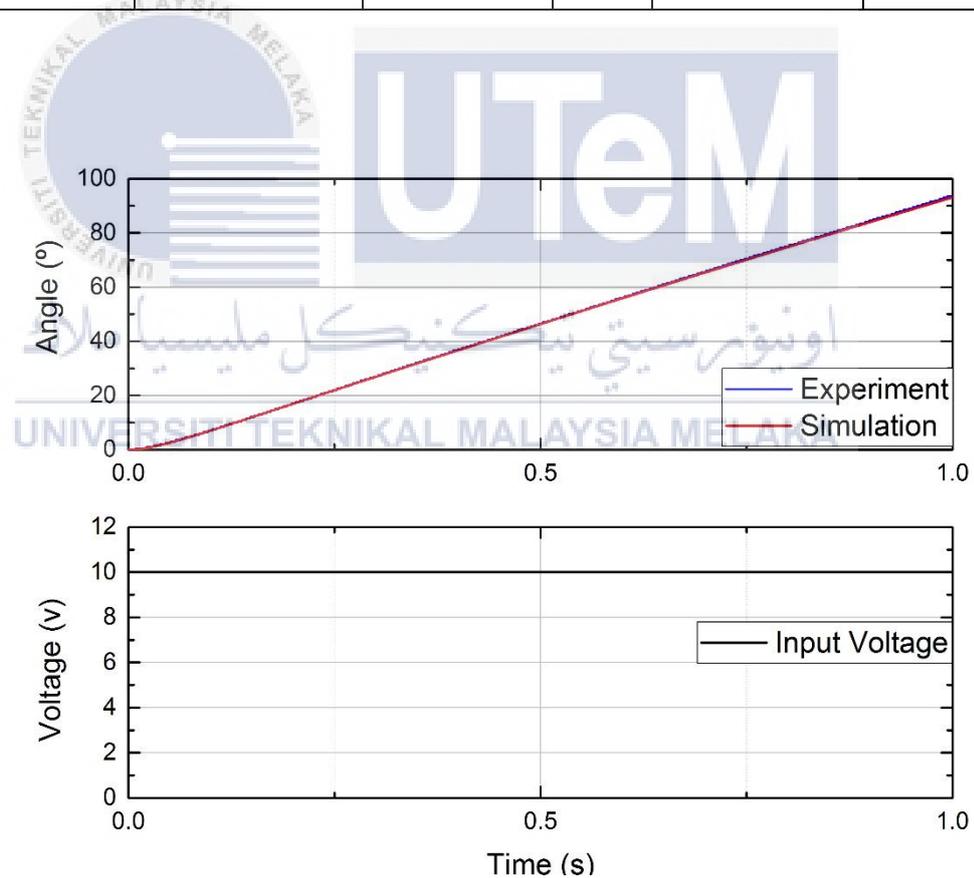


Figure 4.10: Input voltage and Output Angle Versus Times Graph, (10V)

After finished the simulation from 2v to 10v, all the results will be compared and the lowest error of the transfer function will be selected for further analysis and experiment during closed-loop simulation. The error each of the voltage applied are shown in the figure 4.11.

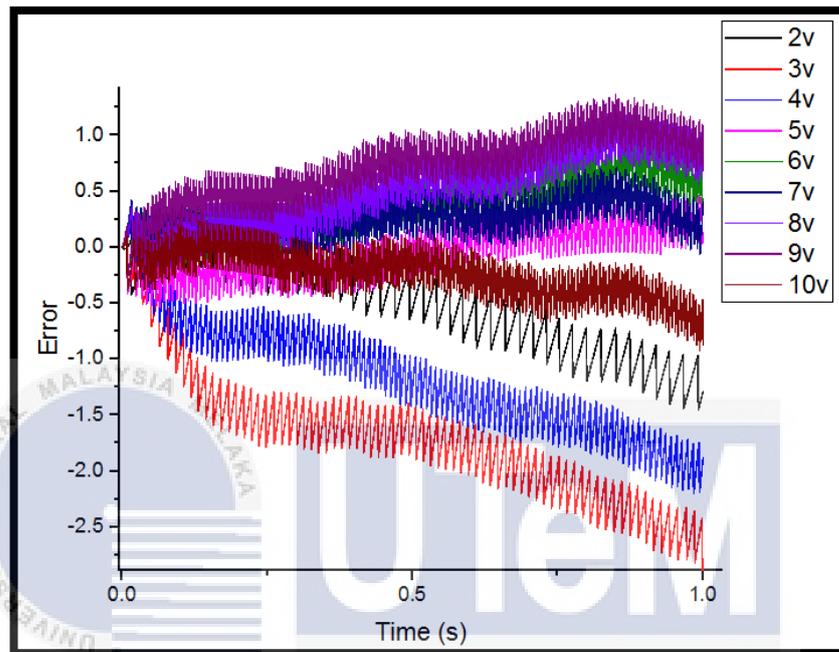


Figure 4.11: The Error of the Output Angle from the Voltages Applied (2V-10V)

As shown in the figure 4.12, 5 volt give the result that has the lowest error.

For that, 5v is chosen as the reference input voltage as the results shown are of less noise or disturbances compared to other results.

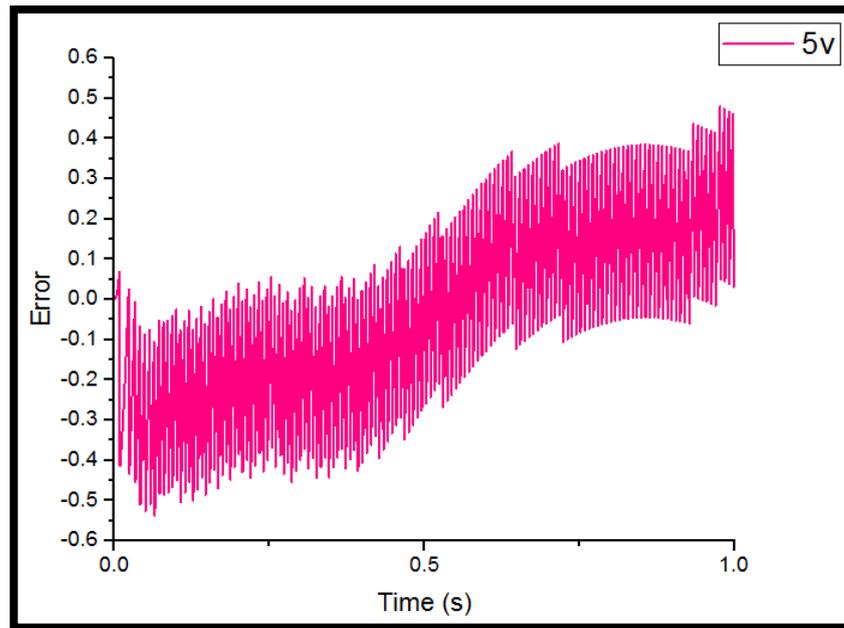


Figure 4.12: Error of the output angle when 5V is applied as input (lowest error)

Equation 4.4 is selected as the motor transfer function. Then it will be replaced in the system and used throughout closed-loop simulation for further evaluation.

#### 4.2.1 Linearity

The results of output angle (in degree) versus input voltage (in volt) are plotted. Table 4.10 shows the data obtained from open loop simulation. Figure 4.13 shows the graph of output angles against the input voltages. It is observed that the graph of output angle versus input voltage is linear at the first 6v. Thus, the voltage gain can be obtained from the graph. The voltage gain is 0.057. This gain is applied to the simulation to convert the reference angle to reference voltage.

Table 4.10: Data of output angle obtained when voltage is applied to the motor.

Input Voltage (v)	Output Angle (°)
1	0
2	20.5
3	37.5
4	54
5	71.5
6	90
7	93
8	93
9	93
10	94

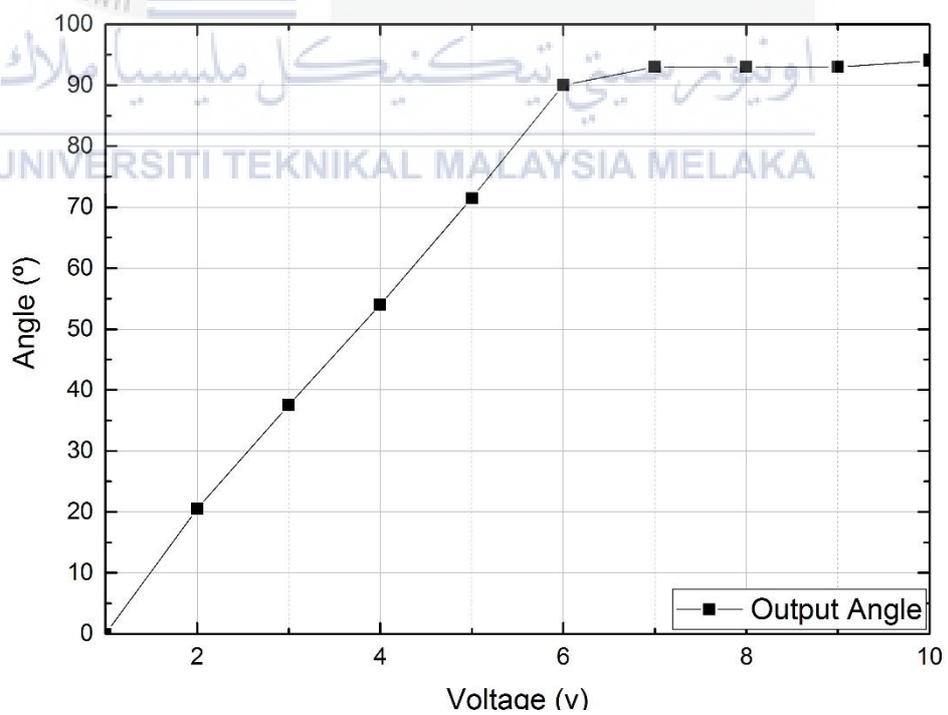


Figure 4.13: Graph of output angles against input voltages

### 4.3 Uncompensated System

In Uncompensated Closed-Loop System, a negative feedback is added into the system. The feedback in the system was required in order to reduce the error produced in the robotic arm system by comparing the feedback value with the reference value. An Uncompensated Closed-Loop System for the robotic arm is where a reference angle was used as the input and the DC Geared Motor will produce the output angle as a reference angle. An additional gain was added into the system where it was use to convert the reference angle to the reference voltage which will supply to the DC Geared Motor. The angle conversion gain is obtained through the linearity of the system produce during the open-loop test. Figure 4.14 to Figure 4.16 shows that the performance of the Uncompensated Closed-Loop System for the robotic arm.

#### 4.3.1 Point to Point Trajectory Control for Uncompensated System

Trajectory generation is the process of selecting a motion and the associated input control to provide a complete and precise description of the robot motion using a suitable model of robots.

In this section, a closed-loop uncompensated system (without using controller) is designed and simulated. Step signal is given as the input signal and the output graph is observed to verify whether the output signal follows the selected input signal. Table 4.11 shows the parameters being fixed as well as being varied. Figure 4.7 and 4.8 indicate the outcomes of the experiment of point-to-point trajectory control for input angles of 15°, 30° and 60°.

Table 4.11: Parameters for Point to Point Trajectory Control Experiments

Parameter	Numerical Value
Input Angle	15°,30°,60°
Simulation Time	5s
Sampling Time	1ms
Delay	0s
Input Type	Step Input
Controller	None

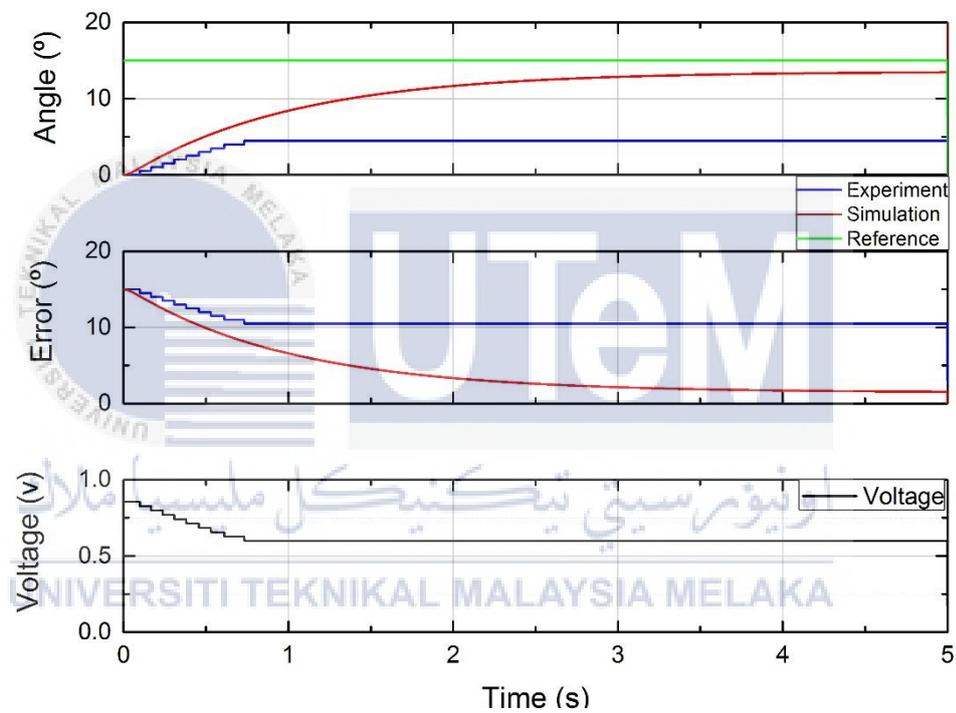


Figure 4.14: The performance of the robotic arm for 15° of reference angle

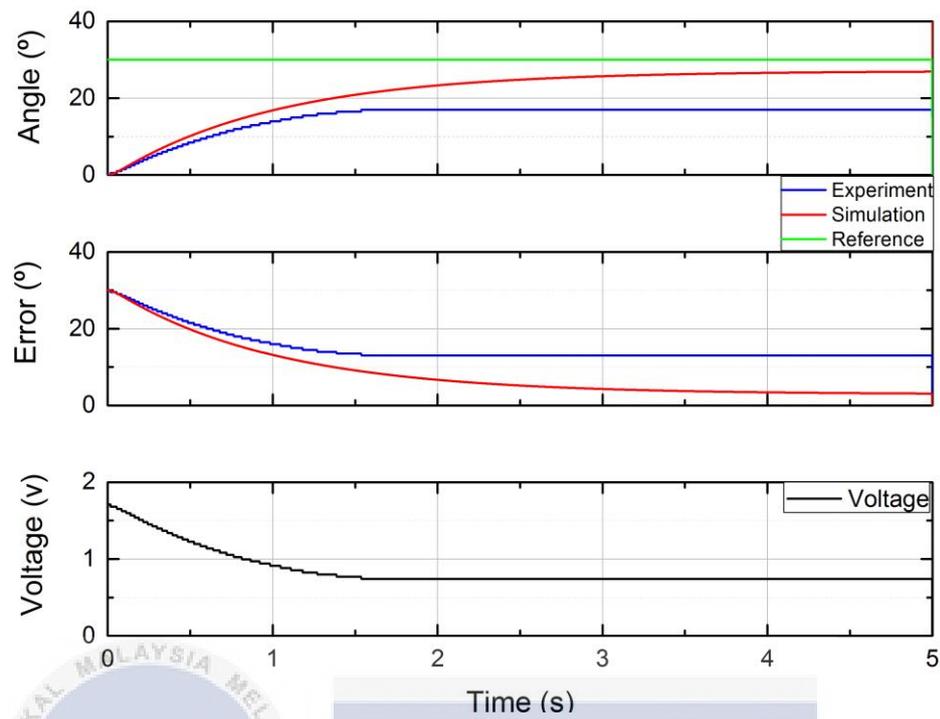


Figure 4.15: The performance of the robotic arm for 30° of reference angle

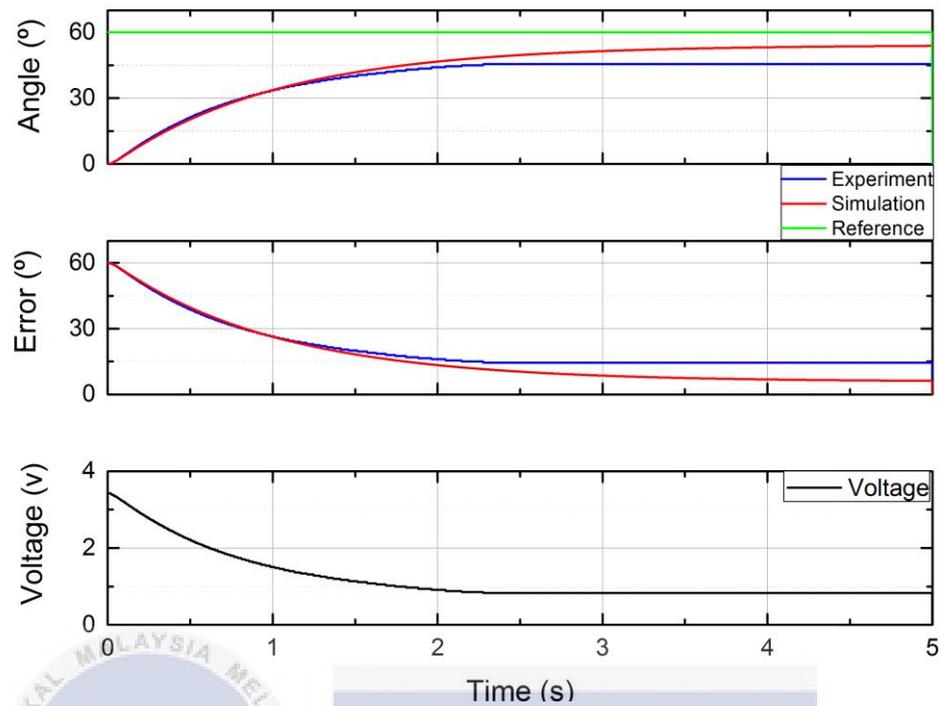


Figure 4.16: The performance of the robotic arm for  $60^\circ$  of reference angle

The reference input voltage supplied to the DC Geared Motor was based on the reference angle. This indicates that when the motor needs to rotate to the reference angle for  $1^\circ$ ,  $0.0571\text{V}$  was required by the DC Geared Motor in order for the motor to reach its desired angle. However, when the input voltage was less than  $0.8\text{V}$ , the torque produced was not enough to drive the DC Geared Motor for experimental results.

The samples that were used for the experiment and simulation were  $15^\circ$ ,  $30^\circ$ , and  $60^\circ$ . Then, the rise time and settling time for both the experimental and simulation were too long even though there was no overshoot in the system.

In other words, as the output signals do not obey the input signals, the uncompensated system throughout this experiment does not generate satisfactory results. In both cases, there are major errors.

### 4.3.2 Tracking Control for Uncompensated System

In this section, the tracking control experiments are carried out for different input angles. Sine wave is given as the input signal in each case. The frequency of the system is varied to observe the effects of different frequencies to the output signals. Table 4.12 shows the parameters being fixed as well as being varied. Figure 4.17 to 4.19 indicated the results of tracking error experiment when the input angle is  $15^\circ$ ,  $30^\circ$ ,  $60^\circ$ .

Table 4.12: Parameters for Tracking Control Experiment

Parameter	Numerical Value
Input Angle	$15^\circ, 30^\circ, 60^\circ$
Simulation Time	5s
Sampling Time	1ms
Delay	0s
Input Type	Sine wave
Controller	None

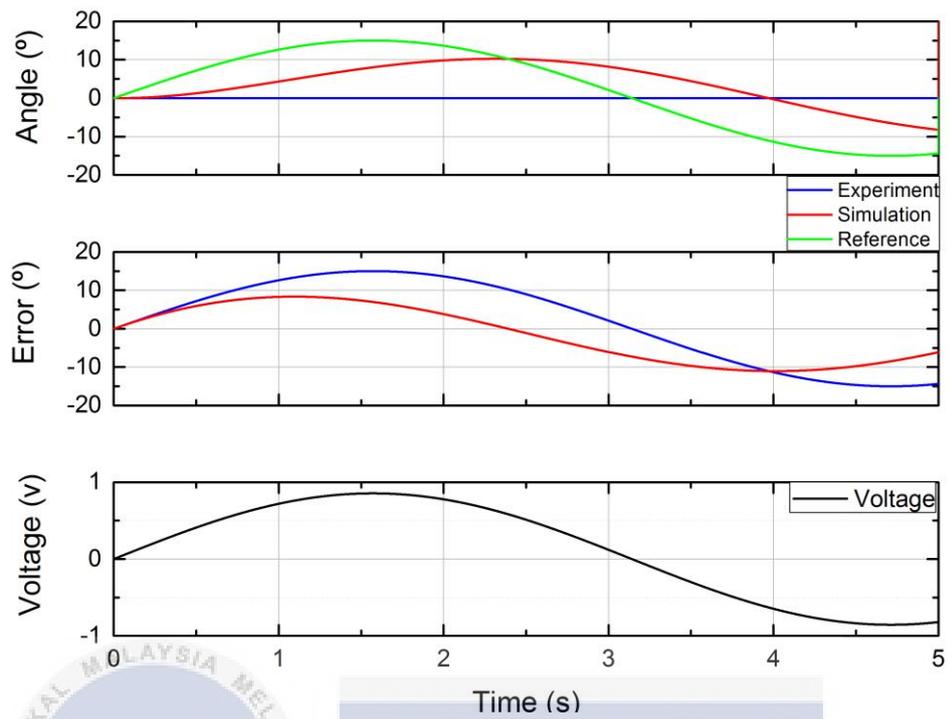
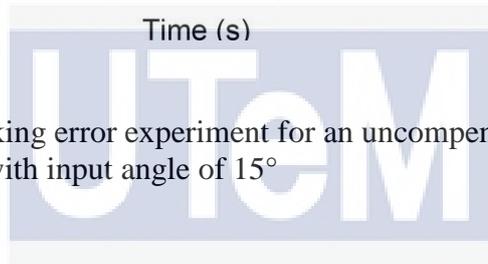
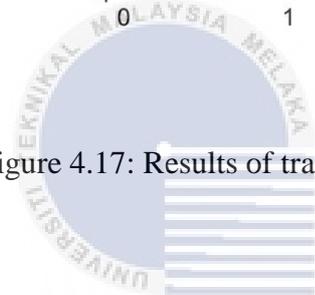


Figure 4.17: Results of tracking error experiment for an uncompensated system with input angle of  $15^\circ$



اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

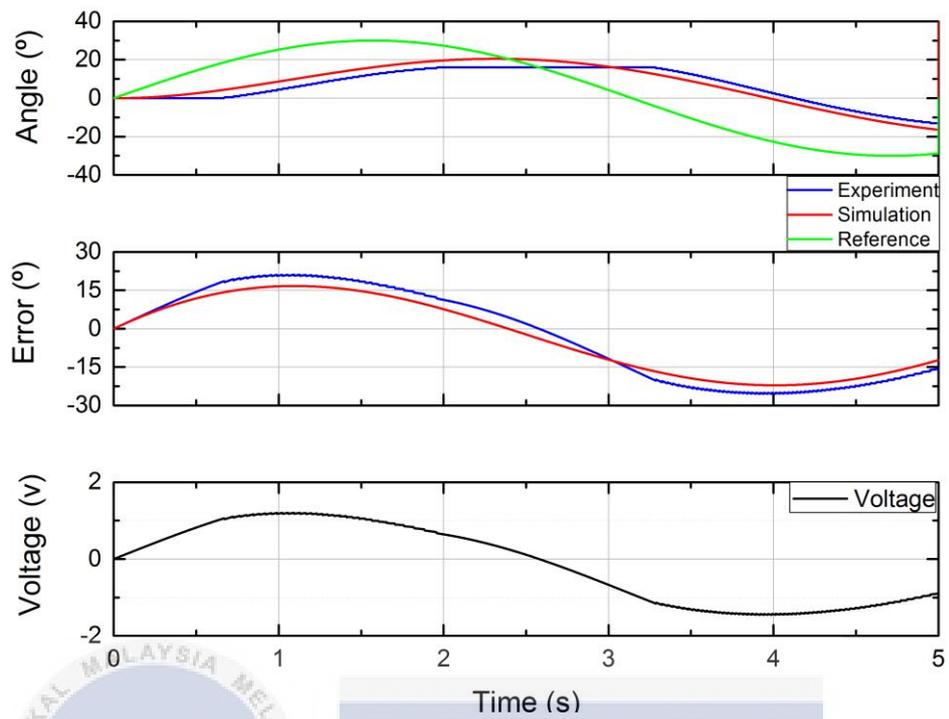


Figure 4.18: Results of tracking error experiment for an uncompensated system with input angle of  $30^\circ$

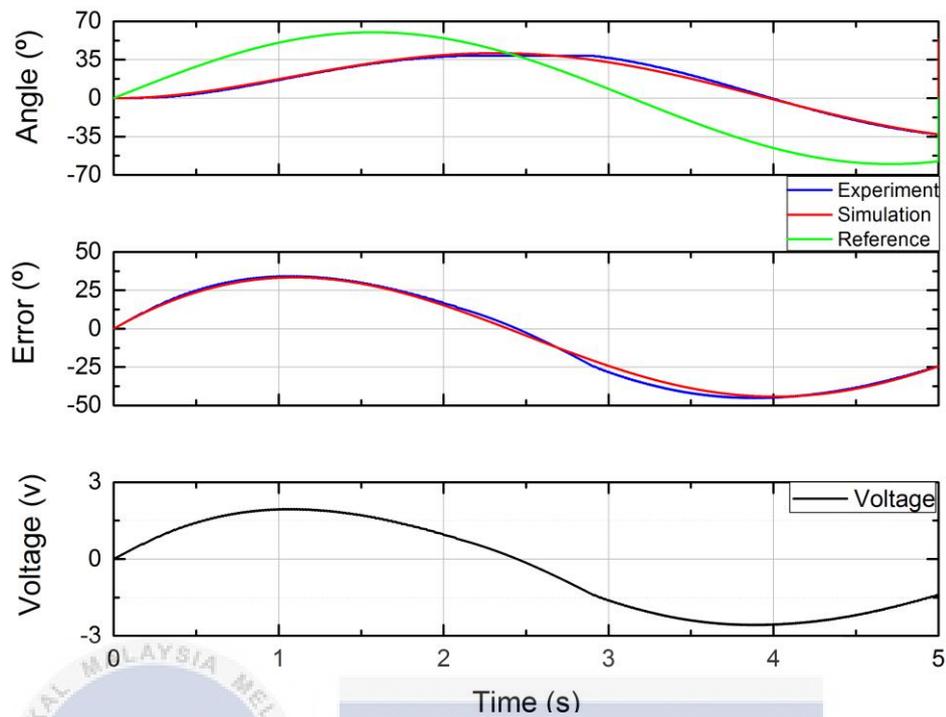


Figure 4.19: Results of tracking error experiment for an uncompensated system with input angle of  $60^\circ$

From the results of simulation for uncompensated system for both experiments, it is summarized that there are large steady-state errors occurred in each cases, in which the output angles are not proportional with the input angles. Hence, a controller is needed to improve the transient response of the system.

#### 4.4 Compensated System with PID controller

The system is introduced in this section with the PID controller. There are two kinds of studies, i.e. point-to-point trajectory control and tracking experiments. This is to evaluate the robotic arm potential and to accurately regulate its movement.

##### 4.4.1 Point to Point Trajectory Control with PID controller

In this experiment there two methods can be applied which are stated in Chapter 3 namely Trial and Error method and Ziegler's Nichols method. Both of this method used and analyzed based on its PID performance and compared which one the best method that gives the best result.

##### 4.4.1.1 Trial and Error

Simulation is conducted with  $K_i = 0$  and  $K_d = 0$  to determine the appropriate  $K_p$  value. The  $K_p$  gain is gradually increased until the system has the smallest steady-state error. Table 4.13 shows the parameters that are being fixed and varied with the PID controller in this compensated system. Figure 4.20 to Figure 4.22 shows the results of simulation with  $K_p$  value of 30. While Table and Table is the system response for the robotic arm for experiment and simulation with different input angles.

Table 4.13: Parameters for Point to Point Experiments using PID Controller

Parameter	Numerical Value
Input Angle	15°, 30°, 60°
Simulation Time	1 s
Sampling Time	1 ms
Delay	0 s
Input Type	Step input
Controller	PID

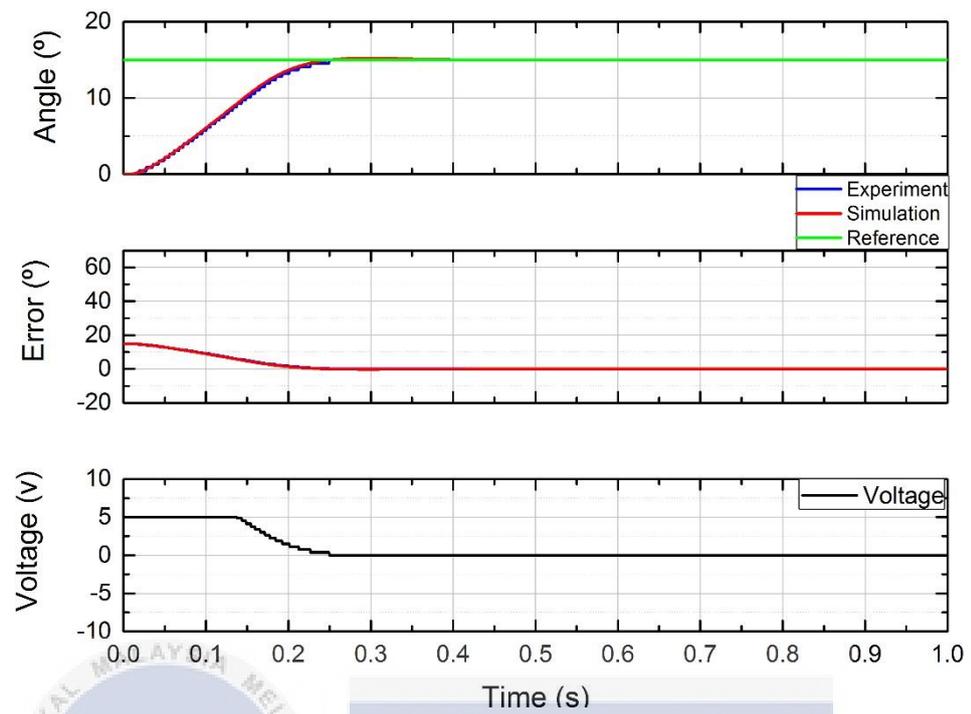


Figure 4.20: Result of Point to Point Trajectory Control Experiment for PID Control with Input Angle of  $15^\circ$  and  $K_p$  value of 30

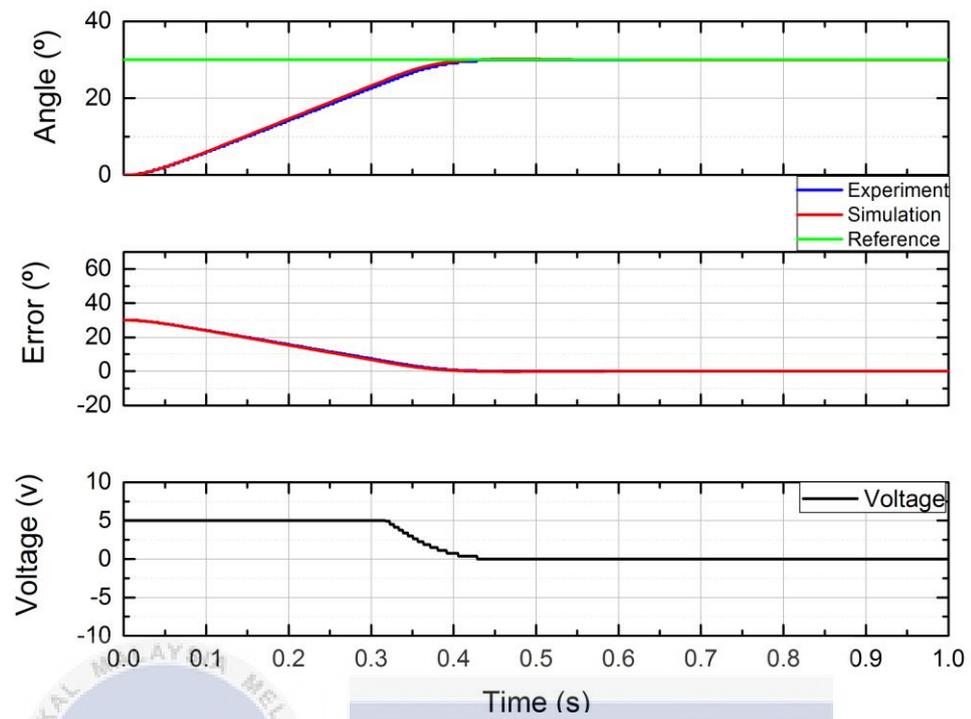


Figure 4.21: Result of Point to Point Trajectory Control Experiment for PID Control with Input Angle of  $30^\circ$  and  $K_p$  value of 30

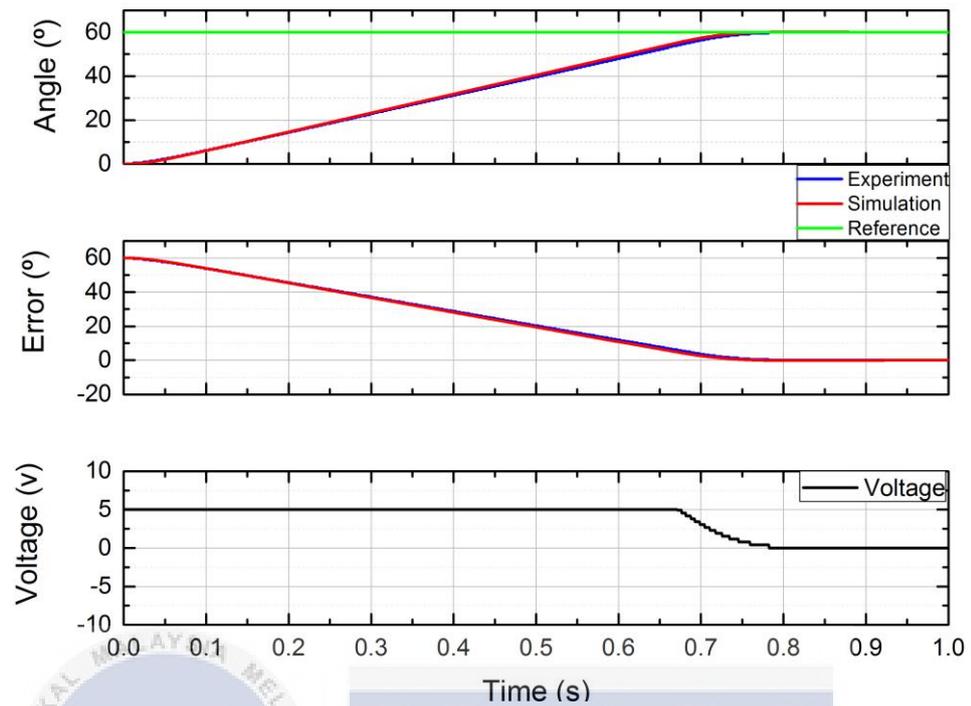


Figure 4.22: Result of Point to Point Trajectory Control Experiment for PID Control with Input Angle of  $60^\circ$  and  $K_p$  value of 30

Table 4.14: Performance of Robotic Arm with PID Controller for Experimental with  $K_p$  value of 30

System Response \ Angle, ( $^\circ$ )	Experimental		
	15	30	60
Rise Time, $T_r$ (s)	0.22	0.371	0.68
Settling Time, $T_s$ (s)	0.263	0.43	0.795
Overshoot, OS (%)	1.89	0.59	0.31
Steady State Error, $E_{ss}$ ( $^\circ$ )	0.05	0.07	0.09

Table 4.15: Performance of Robotic Arm with PID Controller for Simulation with  $K_p$  value of 30

Angle, (°)	Simulation		
	15	30	60
<b>System Response</b>			
Rise Time, $T_r$ (s)	0.215	0.368	0.67
Settling Time, $T_s$ (s)	0.33	0.422	0.787
Overshoot, OS (%)	2.02	0.69	0.33
Steady State Error, $E_{ss}$ (°)	0.008	0.016	0.031

It shows that the results of simulation and experiment have improved as compared to the uncompensated system. But it is still not the desired output because the result has overshoot. In order to decrease or eliminate the overshoot, the value of  $K_d$  need to be increased as stated in Table 3.4 in Chapter 3. Figure 4.23 to Figure 4.25 shows the results of simulation with the  $K_p$  value of 30 and  $K_d$  value of 1. While Table 4.16 and Table 4.17 is the system response for the robotic arm for experiment and simulation with different input angles.

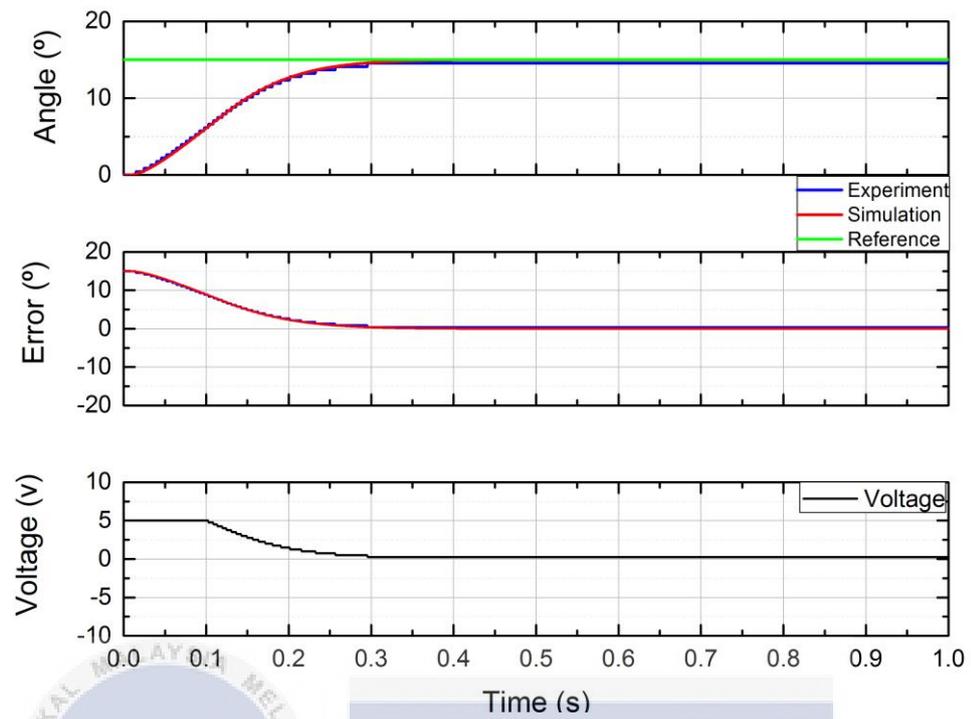


Figure 4.23: Result of Point to Point Trajectory Control Experiment for PID Control with Input Angle of  $15^\circ$  and  $K_p$  value of 30  $K_d$  value is 1

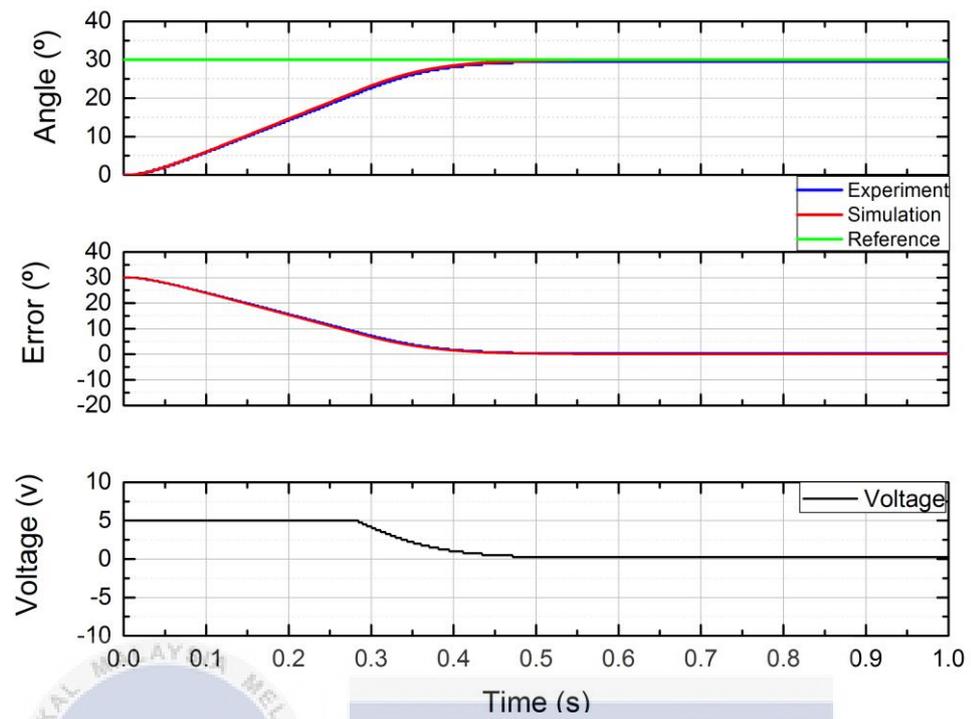


Figure 4.24: Result of Point to Point Trajectory Control Experiment for PID Control with Input Angle of  $30^\circ$  and  $K_p$  value of 30  $K_d$  value is 1

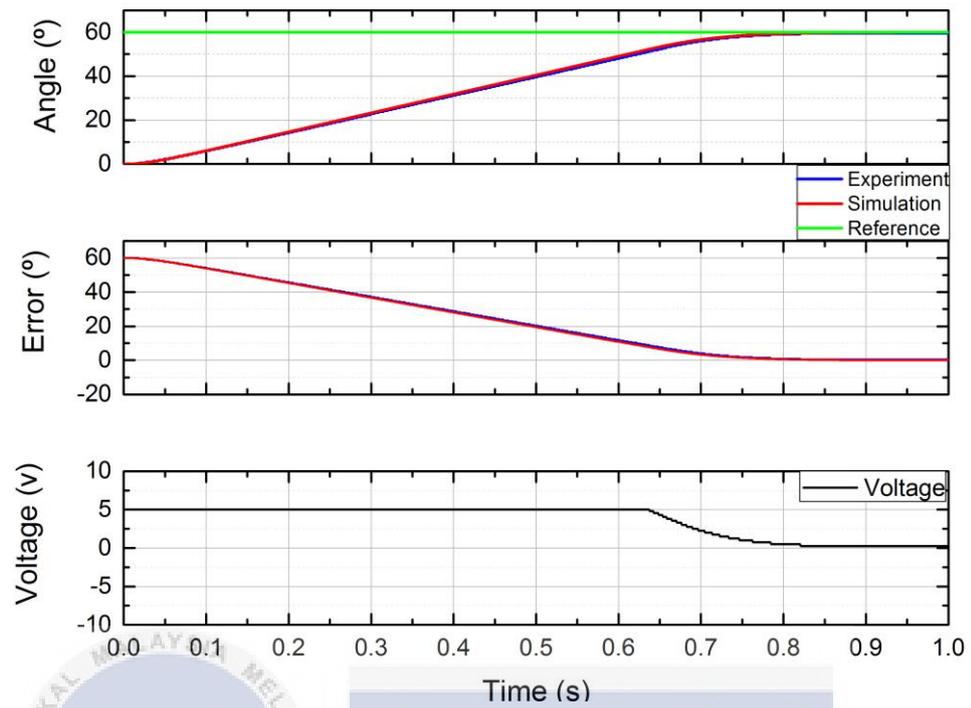


Figure 4.25: Result of Point to Point Trajectory Control Experiment for PID Control with Input Angle of  $60^\circ$  and  $K_p$  value of 30  $K_d$  value is 1

Table 4.16: Performance of Robotic Arm with PID Controller for Experimental with  $K_p$  value of 30 and  $K_d$  value is 1

System Response	Angle, ( $^\circ$ )	Experimental		
		15	30	60
Rise Time, $T_r$ (s)		0.27	0.364	0.652
Settling Time, $T_s$ (s)		0.3	0.482	0.764
Overshoot, OS (%)		0	0	0
Steady State Error, $E_{ss}$ ( $^\circ$ )		0.03	0.05	0.06

Table 4.17: Performance of Robotic Arm with PID Controller for Simulation with  $K_p$  value 30 and  $K_d$  value is 1

Angle, (°) System Response	Simulation		
	15	30	60
Rise Time, $T_r$ (s)	0.278	0.37	0.66
Settling Time, $T_s$ (s)	0.33	0.492	0.769
Overshoot, OS (%)	0	0	0
Steady State Error, $E_{ss}$ (°)	0.008	0.016	0.031

The PID controller designed has improved the system performance of the robotic arm where the rise time is less than 1s, settling time with less than 1s, overshoot of the system is zero and most importantly the PID controller should eliminate the steady-state error occurred in the robotic arm system in order to achieve the Point-to-Point Positioning Control of the robotic hand. Furthermore, the robustness of this controller designed by using Trial and Error will further be analyzed by using Tracking Control.

#### 4.4.1.2 Ziegler Nichols

The gain is increased for this method until the system begins to oscillate. Table 4.18 indicates the parameters that are fixed and varied with the PID controller in this compensated system.

Table 4.18: Point to Point Experiments Parameters using PID controller

Parameter	Numerical Value
Input Angle	15°,30°,60°
Simulation Time	1s
Sampling Time	1ms
Delay	0s
Input Type	Step input
Controller	PID

In this experiment, the system entered full oscillation when the  $K_p$  value rose to 120. The gain value during this condition is named as ultimate gain,  $K_u$  as discussed previously in Chapter 3, whereas the period of oscillation is named as  $T_u$ . The  $K_p$  value is then calculated from  $K_u$ ,  $K_i$  and  $K_d$  values are calculated on the basis of Equation 3.3.

Figure 4.26 shows the results complete oscillation of simulation with a constant input angle of 180° when the  $K_p$  value is 120. Figure 4.27 shows the closed up version of Figure 4.26 by reduce the scale of x and y-axis of the graph to see the oscillation more clearly.

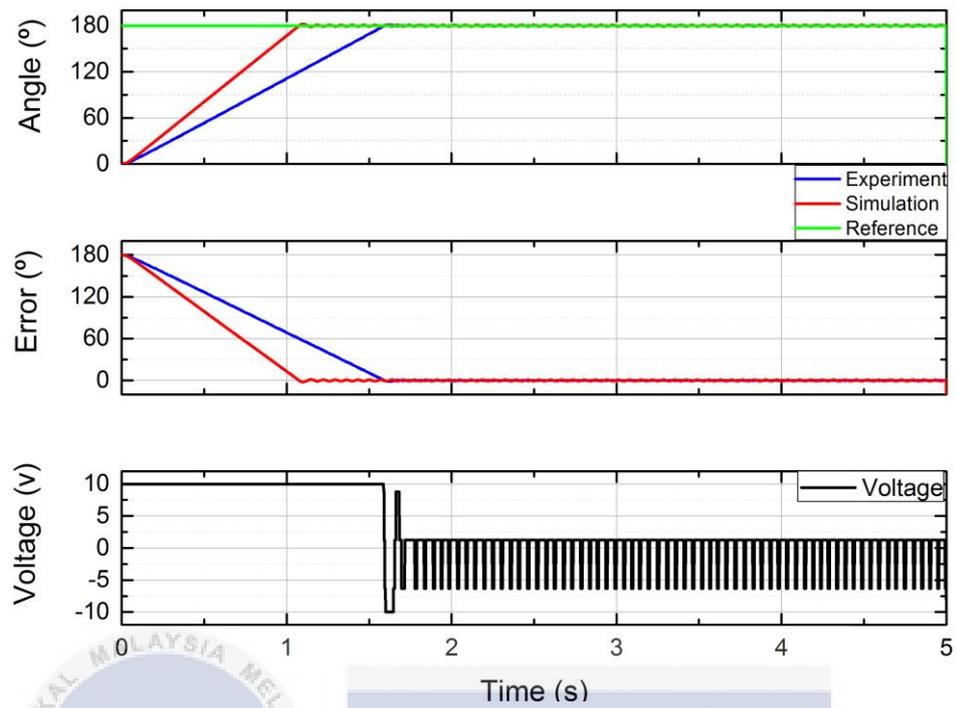


Figure 4.26: Results of Point to Point Trajectory Control Experiment for a PID Control System with input angle  $180^\circ$  and  $K_p$  value of 120.

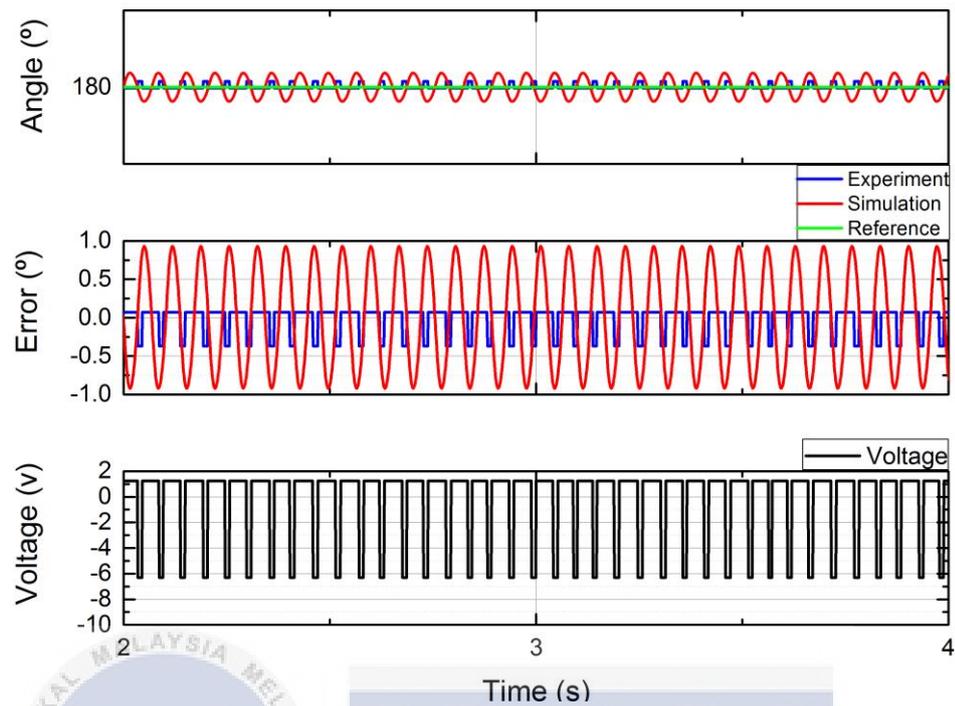


Figure 4.27: Results of Point to Point Trajectory Control Experiment for a PID Control System with input angle  $180^\circ$  and  $K_p$  value of 120 (reduced scale).

Since the system reached complete oscillation, the parameters of PID controller can be calculated,

Ultimate gain,  $K_u = 120$

From Table 3.5: Proportional gain value,  $K_p = 0.6T_u = (0.6)(120) = 72$

From the graph: Ultimate period,  $T_u = 1.180 - 1.077 = 0.103\text{s}$

From Table 3.5: Integral time,  $T_i = 0.5T_u = (0.5)(0.103) = 0.0515$

From Equation 3.3: Integral gain,  $K_i = \frac{K_p}{T_i} = \frac{72}{0.103} = 699.03$

From Table 3.5: Derivative gain  $T_d = 0.12T_u = (0.12)(0.103) = 0.01236$

From Equation 3.3: Derivative gain,  $Kd = TdKp = (0.01236) (72) = 0.88992$

Thus from the results of simulation of input angle  $180^\circ$ , the following data of PID can be obtained as shown in Table 4.19.

Table 4.19: Parameters of PID controller obtained from simulation results

Parameter	Numerical Value
Proportional gain, $Kp$	72
Integral gain, $Ki$	699.03
Derivative gain, $Kd$	0.88992

The calculation is just an approximation based on the experiment being carried out. The  $Kp$  value is set on the next experiment to validate the reliability of the results and the  $Ki$  value is varied to reduce the system's steady-state error.

As  $Ki$  values being applied to the system, there are vibrations of the robotic arm motion. There is no need for vibration. Thus, the  $Ki$  value is kept at zero. Table 4.20 demonstrates the parameters that are being fixed and varied with the PID controller in this compensated system. Figure 4.28 to Figure 4.30 shows the results when  $Kp$  and  $Kd$  value being applied to the PID controller. While Table 4.21 and Table 4.22 shows the system response of different input angles from experiment and simulation.

Table 4.20: Parameters for Point-to-Point Experiments using PID controller

Parameter	Numerical Value
Input Angle	15°,30°,60°
Simulation Time	1s
Sampling Time	1ms
Delay	0s
Input Type	Step input
Controller	PID

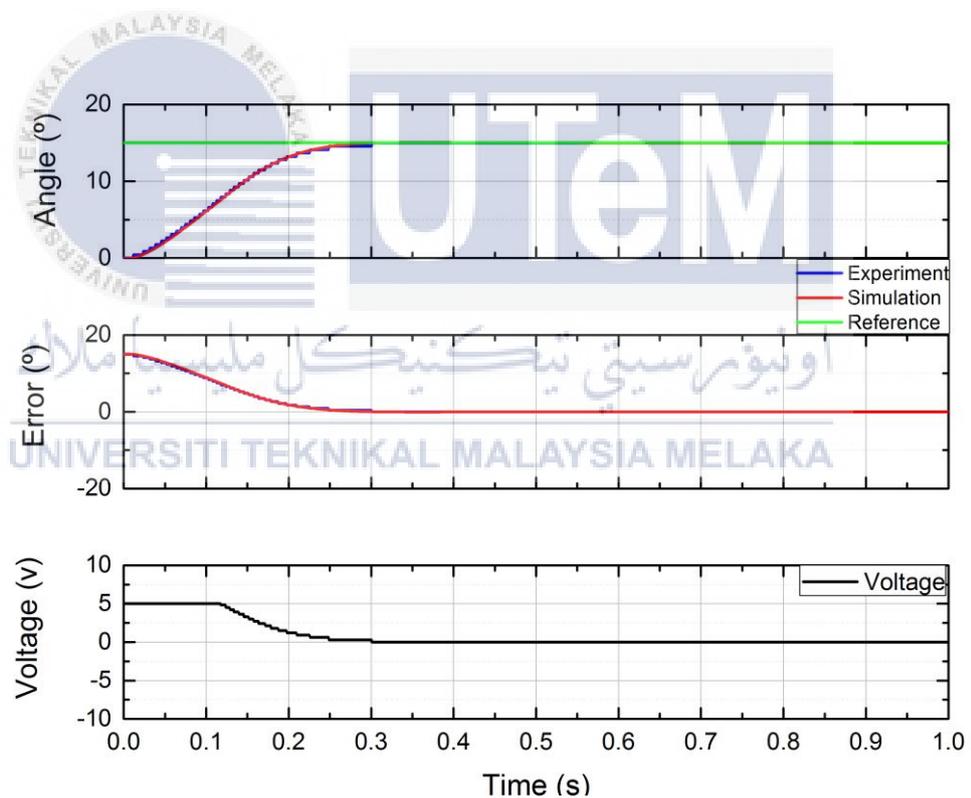


Figure 4.28: Point-to-Point Trajectory Control Experiment Results for a 15° Input Angle PID Control System, 72  $K_p$  and 0.88992  $K_d$  value

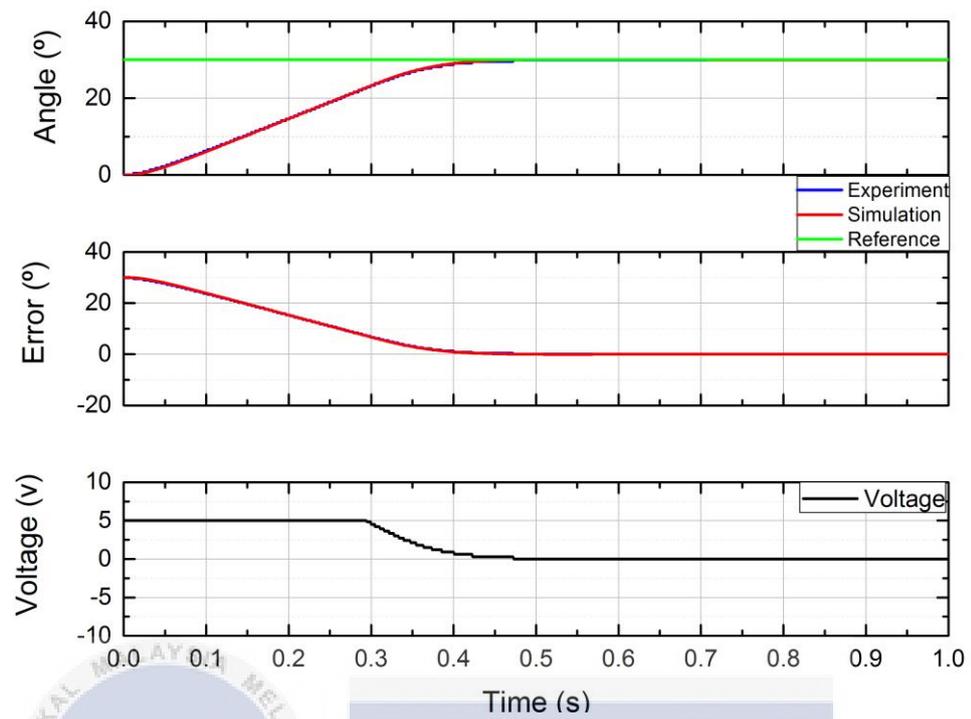


Figure 4.29: Point-to-point trajectory control experiment results for a  $30^\circ$  input-angle PID control system,  $72 K_p$  and  $0.88992 K_d$  value

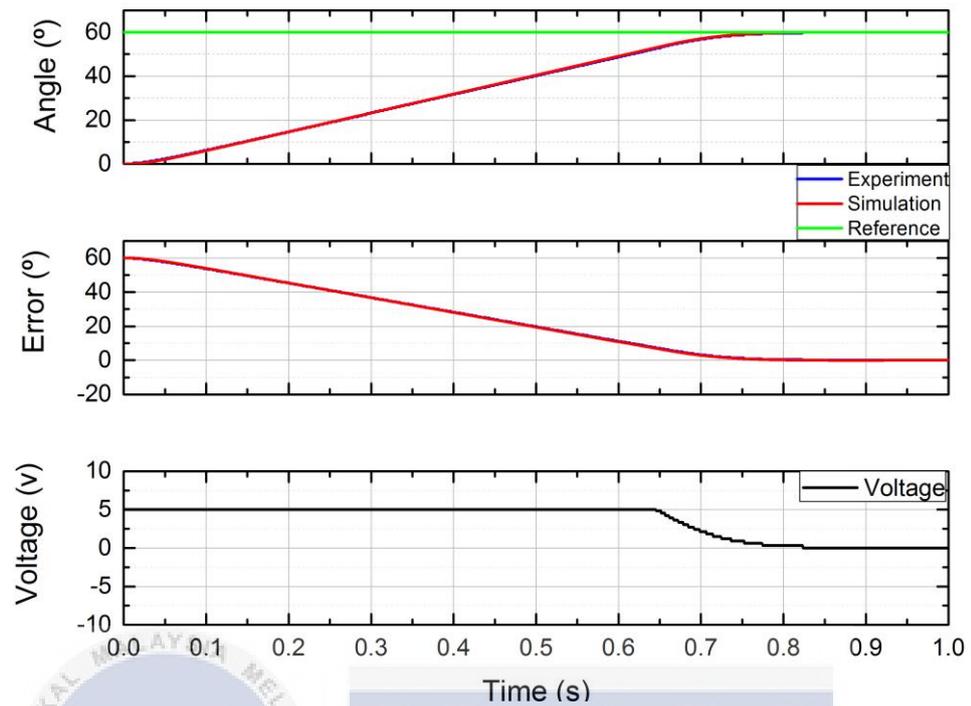


Figure 4.30: Point-to-point trajectory control experiment results for a  $60^\circ$  input-angle PID control system,  $72 K_p$  and  $0.88992 K_d$  value

Table 4.21: Performance of Robotic Arm with PID Controller for Experimental with  $K_p$  value of 72 and  $K_d$  value of 0.88992

System Response	Angle, ( $^\circ$ )	Experimental		
		15	30	60
Rise Time, $T_r$ (s)		0.232	0.372	0.693
Settling Time, $T_s$ (s)		0.322	0.481	0.842
Overshoot, OS (%)		0	0	0
Steady State Error, $E_{ss}$ ( $^\circ$ )		0.002	0.002	0.004

Table 4.22: Performance of Robotic Arm with PID Controller for Simulation with  $K_p$  value of 72 and  $K_d$  value of 0.88992

System Response	Angle, ( $^{\circ}$ )	Simulation		
		15	30	60
Rise Time, $T_r$ (s)		0.23	0.388	0.689
Settling Time, $T_s$ (s)		0.319	0.478	0.831
Overshoot, OS (%)		0.09	0	0
Steady State Error, $E_{ss}$ ( $^{\circ}$ )		0	0.001	0.002

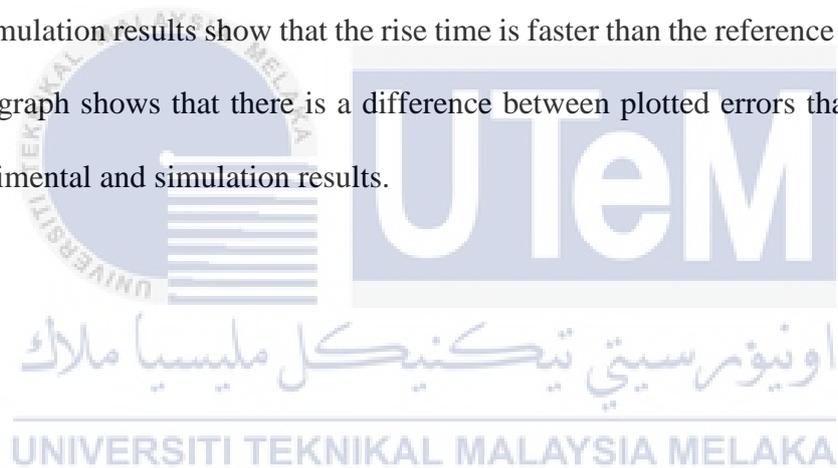
In this final result of Point-to-Point Trajectory Control experiment of the PID control system, Ziegler's Nichols method shows a slightly better result as compared to the Trial and Error method. The PID controller designed by two of this method will be applied to Tracking Control to test the robustness of the controller.

#### 4.4.2 Tracking Control with PID controller

The robustness of a controller is very important in order to make sure the ability of controller to manage and deal with the error that occurred in a system. In order to evaluate the robustness of the PID controller that has been designed using different form of input signal was given to the robotic arm system such which is sine wave signal. The comparison for the tracking control will be done for 15 and 30 with

frequency 0.1Hz. The performance of the tracking control with sine wave signal shows in Figure 4.31 to Figure 4.34.

The sine wave signal is used as the input reference signal with the frequency of 0.1Hz for 15 and 30 shows in Figure 4.31 to Figure 4.34. Figure 4.31 and Figure 4.32 are designed using the Trial and Error method while Figure 4.33 and Figure 4.34 using Ziegler's Nichols method. The period for completing one full cycle of oscillation is 10s. From Figure 4.31 to Figure 4.34, the error of the system is increasing as the input reference is increasing. The experimental results show that the rise time for the PID controller is able to track the signal is slower compared to the reference signal but the simulation results show that the rise time is faster than the reference signal. So, the error graph shows that there is a difference between plotted errors that occur in the experimental and simulation results.



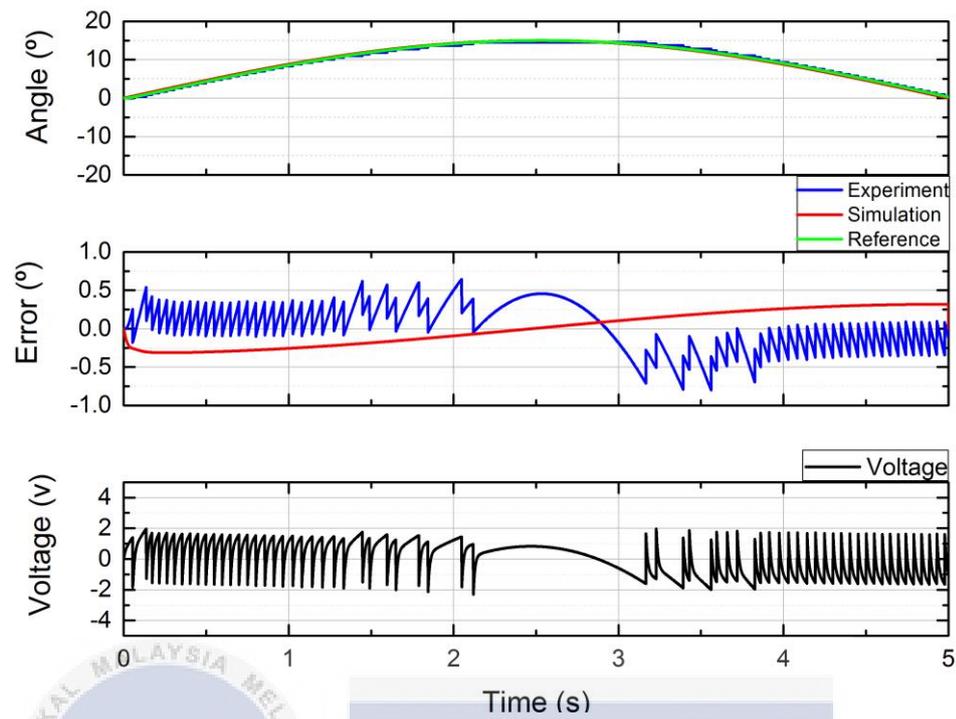


Figure 4.31: Performance of PID controller designed by using Trial and Error method with Sine Wave Signal for  $15^\circ$  at 0.1 Hz

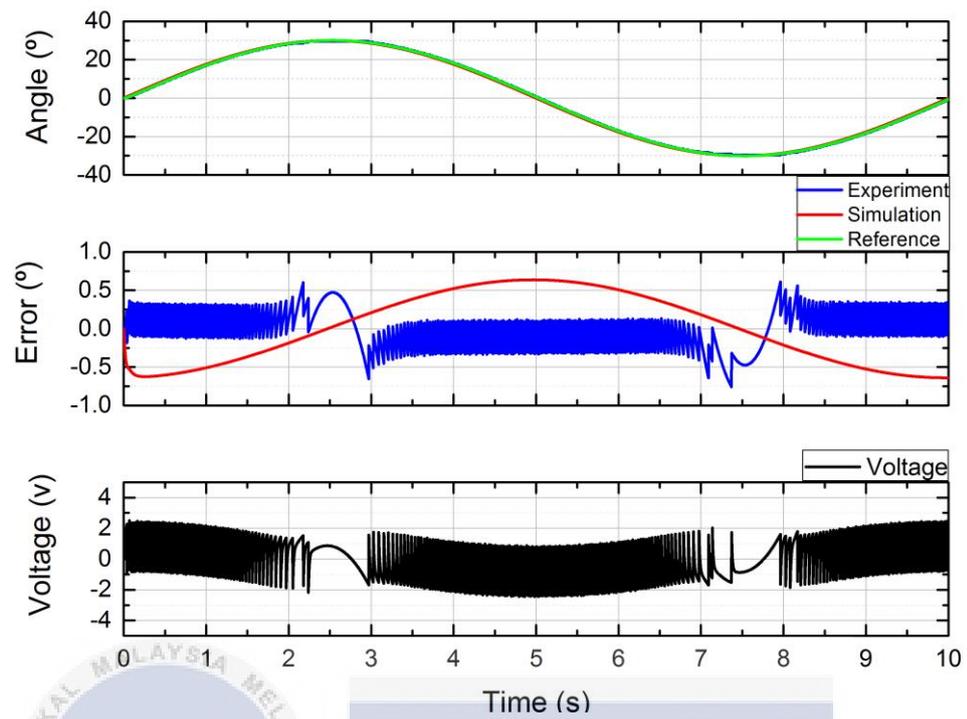


Figure 4.32: Performance of PID controller designed by using Trial and Error method with Sine Wave Signal for  $30^\circ$  at 0.1 Hz

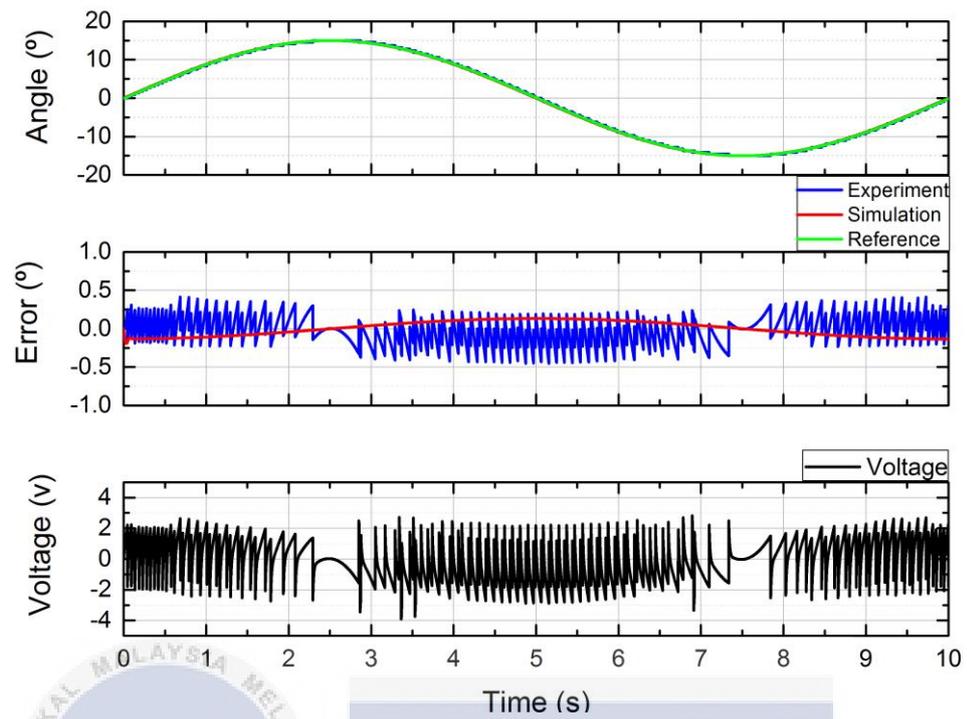


Figure 4.33: Performance of PID controller designed by using Ziegler's Nichols method with Sine Wave Signal for  $15^\circ$  at 0.1 Hz

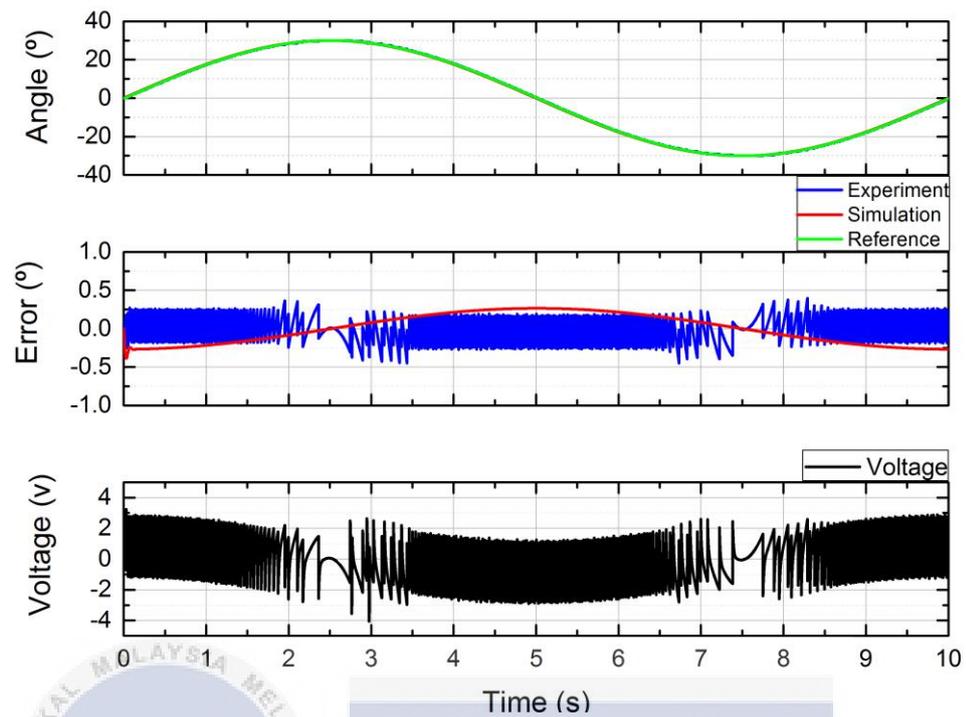


Figure 4.34: Performance of PID controller designed by using Ziegler's Nichols method with Sine Wave Signal for  $30^\circ$  at 0.1 Hz

From the error graph, it can be seen that the error of controller designed using Trial and Error method are exceed  $0.5^\circ$  while controller designed using Ziegler's Nichols is below  $0.5^\circ$ . This can be concluded that the PID controller that is designed using Ziegler's Nichols method is more robust compared to the Trial and Error method.

#### 4.5 Summary

In this project, the PID controller is designed to improve the system performance of the robotic arm. During the robotic arm performs the uncompensated closed-loop system, the rise time and settling time is slower meanwhile the steady-state error is large. However, the designed PID controller is introduced with Proportional gain of 72 and Integral gain of 0 and Derivative gain of 0.88992 and it successfully improves the system response in term of rise time, settling time, and steady-state error. When the reference angle is  $15^\circ$ , the PID controller improves the rise time from 0.642s to 0.232s. Then the settling time is improved from 0.754s to 0.322s. On the other hand, the steady-state error was eliminated. As the result, the designed PID controller is able to perform well in Point-to-Point Positioning Control as the robotic arm has the fast rise time and settling time with no steady-state error which fulfills the characteristics of a Point-to-Point Positioning Control. Although the PID controller performs well in Point-to-Point Positioning Control, the controller does not show the robustness characteristic when the PID controller is used for tracking control due to the PID controller cannot track the reference signal exactly.

## CHAPTER 5

### CONCLUSION AND RECOMMENDATION

#### 5.1 Conclusion

In conclusion, all the objectives for the Projek Sarjana Muda have been achieved where the open-loop performance of the robotic arm was evaluated. Other than that, the designed PID controller also successfully improved the Point-to-Point Positioning performance of the robotic hand. In addition, the experimental and simulation performance of the PID controller was analyzed and compared by conducting the Point-to-Point Positioning control and Tracking Control.

The calibration of the gain for the DC geared motor is very important in order to make sure the encoder able to provide the correct value base on the resolution of the DC geared motor. The gain obtained was 0.445. Then the open-loop test for the robotic arm was done in order to determine the characteristic of the robotic arm. The characteristic equation of mathematical model was then chosen by using the System Identification Tools where 9 transfer function will be generated for each input voltage supply. However, the transfer function which has the smallest error will be selected to

represent the characteristic equation of the robotic arm. So, the Transfer Function of 5V with the characteristic equation of  $G(s) = \frac{-2.935s+730.8}{s^2+39.34s+0.2728}$  was chosen as to represent the robotic arm model.

In order to perform Point-to-point Positioning control for the robotic arm, an uncompensated closed-loop system was developed where the feedback system is designed without the controller. The uncompensated closed-loop system shows the performance of the robotic arm does not perform well in Point-to-Point Positioning Control where the robotic arm has slow rise time, long settling time and most importantly is large steady-state error occurred in the system. So a PID controller was designed in order to improve the system performance of the robotic arm in terms of rising time, settling time and the steady-state error must be eliminated.

Then, the PID controller is designed based on two tuning method which was Ziegler Nichols tuning method of self-oscillation method and trial and error tuning method. The Ziegler Nichols tuning method was conducted until the constant oscillation occurred to obtain the Proportional gain for the PID controller. The trial and error method was done where the Integral gain and Derivative gain parameters were adjusted until the robotic finger shows better performance in Point-to-Point Positioning Control. So the parameters of the PID controller were the Proportional gain of 72, Integral gain of 0 and the Derivative gain of 0.88992. The designed PID controller has shown that the controller has improved the performance of the Point-to-Point Positioning control where the robotic arm has fast rise time, shorter settling time and no steady-state error occurred in the system as compared to the uncompensated closed-loop system.

In order to determine the robustness of the designed PID controller, tracking control has been done where the sine wave input signal was used. The PID controller that was designed using trial and error method and Ziegler Nichols method was compared to its performance. Ziegler Nichols method shows better performance as the error is less as compared to the trial and error method.

So, the designed PID controller has better performance in Point-to-point Positioning control where the robotic arm able to reach the desired position exactly. However, the PID controller performance of tracking control has shown the undesired result as the robotic arm unable to give performance exactly like the reference input.

## 5.2 Recommendation

The PID controller has successfully designed for Point-to-Point Positioning control system. However, the PID controller was not performed and show good performance for tracking control. On the other hand, the angle of each joint of the robotic arm cannot be controlled. So, the suggested future works are listed below,

1. Design the controller for the other part of the robotic arm

The project was done to evaluate based on the performance of the 2<sup>nd</sup> limb of the robotic arm. However, each of the robotic arms has characteristics in terms of length and weight. So a new controller can be designed to analyze the performance of the other joint of the robotic arm. The system performance for the controller for 2<sup>nd</sup> limb and 1<sup>st</sup> limb should be compared and evaluated.

2. Implement the designed PID controller for the robotic arm with load

The PID controller is able to perform the Point-to-Point Positioning Control well without load. However, the robotic finger should be added with the load and the performance of the Point-to-Point Positioning Control might be changed. So, the performance of the designed PID controller for the robotic arm will be done in order to evaluate the controller's performance and the robustness of the controller

### 3. Design the Artificial Intelligent Controller

The artificial intelligent controller is a controller related to human thinking and natural language. The artificial intelligent controller such as a fuzzy logic controller with a linguistic control strategy based on expert knowledge or control engineering knowledge might perform a better performance for both Point-to-Point Positioning Control and Tracking Control. This is because the controller is able to handle the system with non-linear characteristics. Then, the performance of the PID controller and Artificial Intelligent controller will be compared.

اونیورسیتی تکنیکل ملیسیا ملاک

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

## REFERENCES

- [1] J. Godjevac, "Comparison between PID and fuzzy control," p. 23, 2012.
- [2] S. Hari om Bansal, "PID Controller Tuning Techniques: A Review," *Journal of Control Engineering and Technology (JCET)*, p. 10, 2012.
- [3] H. Sathishkumar, "A novel fuzzy logic controller for vector controlled induction motor drive," p. 6, 2017.
- [4] N. Liouane, "Tuning PID Controller Using Multiobjective Ant Colony Optimization," *Applied Computational Intelligence and Soft Computing*, p. 8, 2012.
- [5] Shri Bhagwan, "A Review on: PID Controller," *International Journal on Recent Technologies in Mechanical and Electrical Engineering (IJRMEE)*, vol. 3, no. 2, p. 6, 2016.
- [6] Mohamed Slim Masmoudi, "Fuzzy logic controllers design for omnidirectional mobile robot," *Applied Soft Computing*, p. 19, 2016.

- [7] I. Mohd Ashiq Kamaril Yusoff, "Wireless Mobile Robotic Arm," *International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012)*, p. 7, 2012.
- [8] L. Adam Tan Mohd Amin, "Adaptive controller algorithm for 2-DOF humanoid robot arm," *2nd International Conference on System-Integrated Intelligence: Challenges for Product and*, p. 10, 2014.
- [9] J. R. Rogers, "Low-cost teleoperable robotic arm," p. 6, 2009.
- [10] I. Wan Muhamad Hanif Wan Kadir, "Internet Controlled Robotic Arm," *International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012)*, p. 7, 2012.
- [11] A. Shah R., "Concept for Automated Sorting Robotic Arm," *2nd International Conference on Materials Manufacturing and Design Engineering*, p. 6, 2018.
- [12] A. Simon A/L Luthsamy, "Design and Control of an Anthropomorphic," *Journal Of Industrial Engineering Research*, p. 9, 2016.
- [13] C. Luo, "A multi agent multi sensor based real-time sensory control system for intelligent security robot.," *IEEE International Conference on Robotics and Automation*, vol. 2, 2003.
- [14] J. Baltes, "A Cost Oriented Humanoid Robot Motion Control System," p. 6, 2014.
- [15] K. Petr Wagner. Jan Kordas, "Motion control for robots based on cubic Hermite splines in real-time," p. 6, 2012.
- [16] L. Yu-Sheng Lu, "Smooth motion control of rigid robotic manipulators with constraints on," p. 15, 2018.

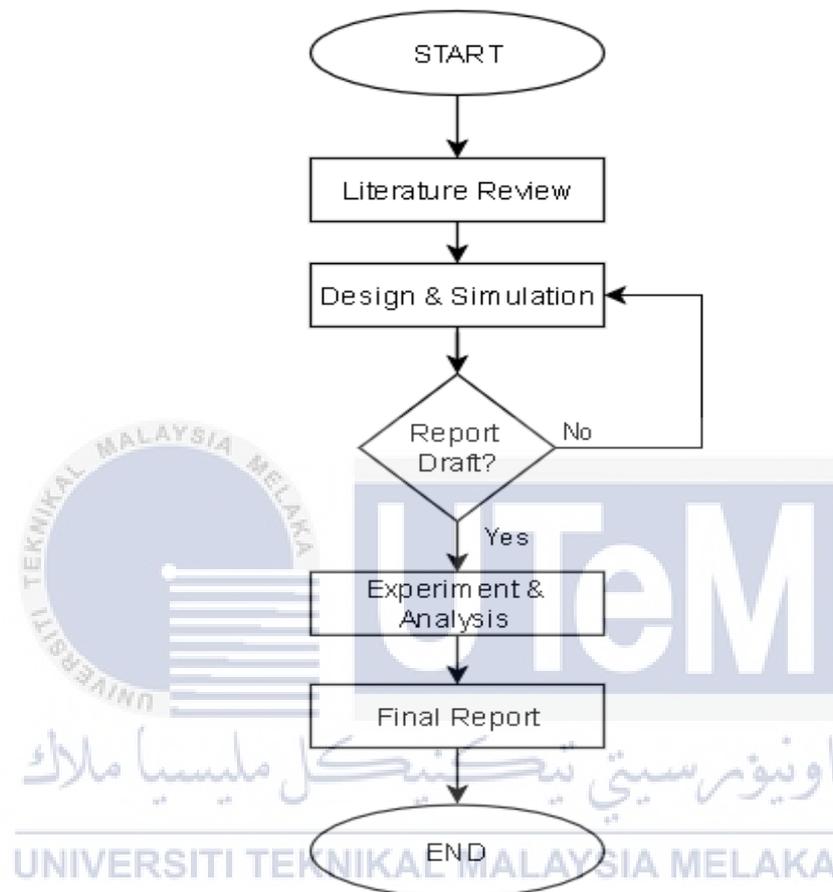
- [17] F. Loucif, "DC MOTOR SPEED CONTROL USING PID CONTROLLER," p. 6, 2005.
- [18] C. M. Nabanita Adhikary, "Sliding mode control of position commanded robot manipulators," p. 16, 2018.
- [19] D. Abbas Karamali Ravandi, "Hybrid force/position control of robotic arms manipulating in uncertain environments based on adaptive fuzzy sliding mode control," p. 11, 2018.
- [20] X. Xiang Luo, "Principle and method of speed control for dynamic walking biped robot," p. 16, 2015.
- [21] A. Rahib H. Abiyev, "Fuzzy control of omnidirectional robot," p. 9, 2017.
- [22] A. Morar, "DC MOTOR SPEED AND POSITION CONTROL SYSTEM," p. 6, 2007.
- [23] G. Chong-Sun Lee, "Fuzzy Logic versus a PID controller for position control of a muscle-like actuated arm," *Journal of Mechanical Science and Technology*, p. 9, 2008.
- [24] L. Aron Pujana-Arrese, "Modelling in Modelica and position control of a 1-DoF set-up powered," p. 18, 2010.
- [25] H. V. Daniel, *Industrial Robotics Handbook*, New York: Industrial Press Inc., 1983.
- [26] National Instruments, "Fundamentals of Motion Control," 19 March 2018.  
[Online]. Available: <http://www.ni.com/white-paper/3367/en/#toc6>.  
[Accessed 12 2018].

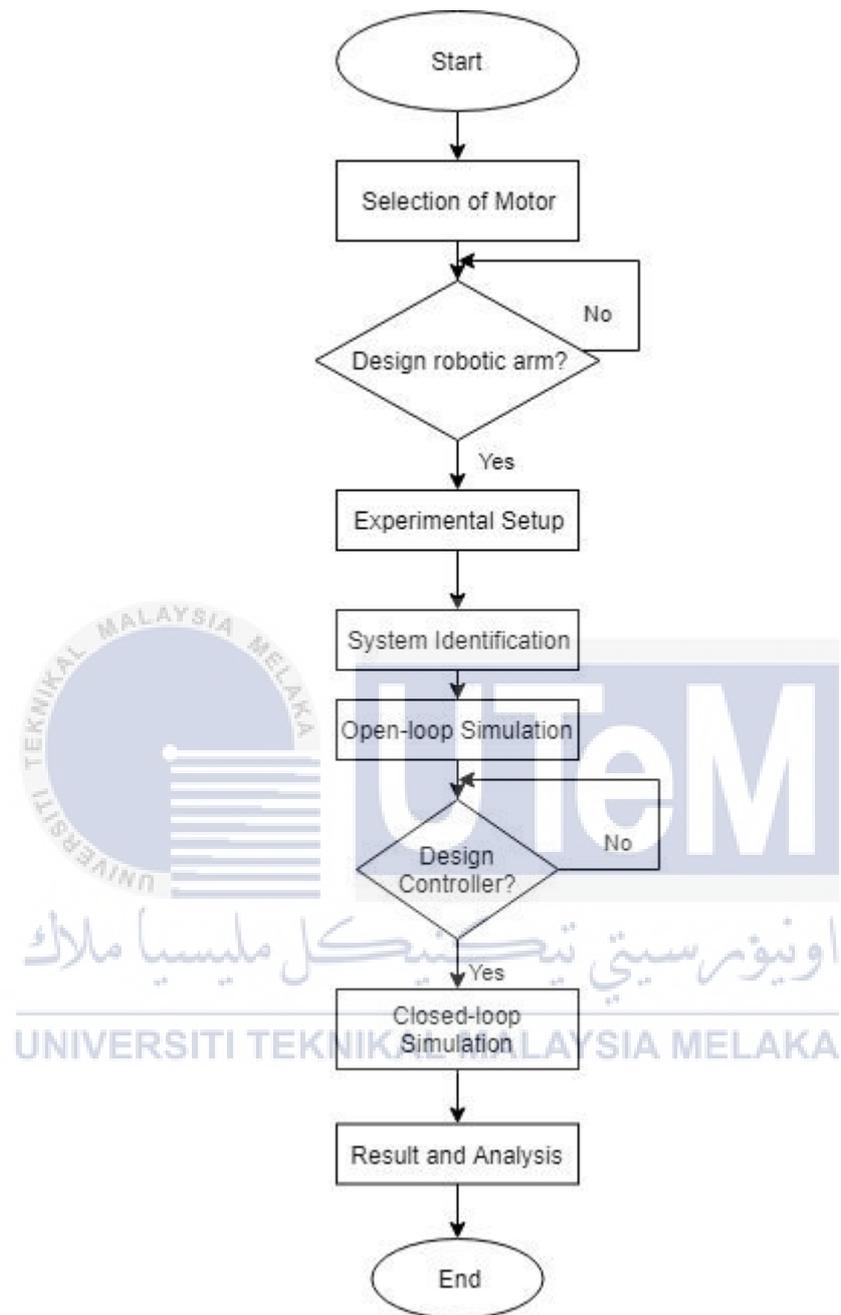
- [27] R. Gonzalez. C.S Lee, "Fuzzy logic versus a PID controller for position control of a muscle-like actuated arm," 2009.



## APPENDICES

### APPENDIX A PROJECT RESEARCH METHODOLOGY IN FLOW CHART



**APPENDIX B EXPERIMENT METHODOLOGY FLOW CHART**

**APPENDIX C PROJECT'S GANTT CHART**

Project Activities	SEM I																SEM BREAK						SEM II																			
	SEP				OCT				NOV				DIS				JAN			FEB			MAR				APR				MAY											
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	1	2	3	4	5	6	7	8	9	10	11	12	13	14						
<b>Introduction</b> <ul style="list-style-type: none"> <li>Background</li> <li>Problem Statement</li> <li>Objectives &amp; Scope</li> </ul>																																										
<b>Literature Review</b> <ul style="list-style-type: none"> <li>Review literature (Journal and textbook)</li> <li>Finding additional information and knowledge about project.</li> </ul>																																										
<b>Research Methodology</b> <ul style="list-style-type: none"> <li>Design Structure</li> <li>Experimental Setup</li> <li>Open-Loop Simulation</li> <li>Closed-Loop Simulation</li> </ul>																																										
<b>Experiment And Simulation Process</b>																																										
<b>Results And Discussion</b> <ul style="list-style-type: none"> <li>Analysis</li> <li>Open-Loop Simulation</li> <li>Closed-Loop Simulation</li> </ul>																																										
<b>Preparation Final Report</b>																																										
<b>Presentation FYP1</b>																																										
<b>Presentation FYP2</b>																																										

STUDY WEEK  
FINAL EXAMINATION