FACULTY OF ELECTRICAL ENGINEERING

DEVELOPMENT AND CONTROL OF EXOSKELETON (UPPER LIMB) FOR MOBILITY SUPPORT

MUHAMMAD IZZ BIN ZAMLUS

BACHELOR OF ELECTRICAL ENGINEERING
(CONTROL, INSTRUMENTATION AND AUTOMATION)

2017

DEVELOPMENT AND CONTROL OF EXOSKELETON (UPPER LIMB) FOR MOBILITY SUPPORT

MUHAMMAD IZZ BIN ZAMLUS

A report submitted in partial fulfillment of the requirements for the degree of Bachelor in Electrical Engineering (Control, Instrumentation and Automation) with Honours

Faculty of Electrical Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2017

APPROVAL

I hereby declare that I have read this dissertation/report and in my opinion this dissertation/report is sufficient in terms of scope and quality as a partial fulfillment Bachelor of Electrical Engineering (Control, Instrumentation & Automation).

Signature	:	....
Supervisor Name	:	....
Date	:	....

DECLARATION

I declare that this thesis entitled "Development and Control of Exoskeleton (Upper Limb) for Mobility Support" is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : ....

Name : ....

Date : ....

Khas buat abah,mama dan adik-adik

ABSTRACT

Mobility support is the movement of an object to support itself**.** This project concerns about the mobility support of exoskeleton (upper limb).The development of upper limb focuses on the human arm which is important for the supportive movement. Human arm is the main mobility of the upper limb part of the body. This project is develop to control the assistive human arm robot that will be support for the elderly and disabilities people to move their hands. So, mobility mechanism will be develop to help these people to move their arm. Arm robot will be modelled in this paper by determining it using the physical modelling. The physical modelling will be create based on the mathematical modelling. A design of the arm model is developed using the computer aided design (CAD) software, Solidworks and simulated with MATLAB. MATLAB software is used to control the movement of human arm. A plug-in software called as Simmechanics will be export the model from the Solidworks and import to the MATLAB. The imported model is then direct to the extension file of MATLAB, Simulink. The arm model which is appears as the Simulink output will be run. A controller which is PID and PD controller is added to the Simulink output to control the movement of human arm. The human arm movement is analyzed based on the output performance of the controller. The best performance of controller will selected to put on the human arm model.

ABSTRAK

Sokongan mobiliti adalah pergerakkan sesuatu objek untuk memberi sokongan kepadanya. Projek ini membicangkan mengenai sokongan mobiliti daripada eksoskeleton (bahagian atas anggota badan manusia) .Bahagian anggota badan yang diberikan tumpuan ialah lengan manusia. Lengan manusia merupakan pergerakan utama anggota badan bahagian atas badan. Projek ini dijalankan bertujuan untuk mengawal lengan manusia. Lengan manusia dibantu oleh sebuah model robot yang akan menjadi sokongan kepada warga kurang upaya untuk menggerakkan tangan mereka. Jadi, model robot tersebut ini akan digunakan untuk membantu orang kurang upaya supaya lengan mereka dapat bergerak. Sebuah model robot yang menyerupai lengan manusia akan direka di dalam projek ini dengan menggunakan model fizikal. Model fizikal direka terlebih dahulu melalui model matematik. Satu reka bentuk model lengan akan direka dengan menggunakan perisian reka bentuk bantuan komputer (CAD) iaitu Solidworks. Satu perisian komputer simulasi iaitu MATLAB akan digunakan dalam projek ini. Simulasi MATLAB digunakan untuk mengawal pergerakan lengan manusia. Satu perisian plug-in yang dikenali sebagai Simmechanics digunakan untuk mengeksport model lengan dari Solidworks dan diimport kepada MATLAB. Model lengan yang diimport kemudiannya akan terus dihantar ke fail sambungan MATLAB Simulink. Model lengan yang muncul sebagai keluaran Simulink akan dijalankan. Dua buah alat kawalan iaitu pengawal PID (*Proportional-Integrator-Derivative*) dan pengawal PD (*Proportional-Derivative*) ditambah kepada keluaran Simulink yang berbeza untuk dijadikan sebagai sebuah sistem bagi mengawal pergerakan lengan manusia. Pergerakan lengan manusia dianalisis berdasarkan prestasi pengawal PID dan

pengawal PD tersebut. Presetasi alat kawalan yang lebih baik dipilih untuk diletak pada model lengan manusia.

ACKNOWLEDGEMENT

Alhamdulillah, first of all I would like to thank God for His love and blessing that I manage to complete my Final Year Project. With this opportunity, I would like to express my gratitude to my supervisor, Dr. Hyreil Anuar Bin Kasdirin for his guidance, support and valuable advices throughout the progress of this project. It would have been difficult to complete this project without his support and encouragement. Moreover, I am also very thankful to my classmates especially Ammar and Gowdie for their kind support and advices that are very useful throughout the project implementation. My appreciation also goes to my family especially to my abah, Zamlus Bin Abdul Latif and my mom, Musalwa Binti Ma'arof who has giving very valuable moral support until I could finish my project. Thanks for their encouragement, love and emotional supports that they had given to me. Not forgotten to my siblings, Iffah, Widad and Icha who always given full support to their big brother until he can complete his project.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
	APPROVAL	
	DECLARATION	
	ABSTRACT	**i**
	ABSTRAK	**ii**
	ACKNOWLEDGEMENT	**iv**
	TABLE OF CONTENTS	**v**
	LIST OF TABLE	**vii**
	LIST OF FIGURE	**viii**
	LIST OF APPENDICES	**x**
1	**INTRODUCTION**	
	1.1 Project Background	1
	1.2 Motivation	2
	1.3 Problem Statement	3
	1.4 Objective	4
	1.5 Scope	5
	1.6 Organization of Thesis	6
2	**LITERATURE REVIEW**	
	2.1 Mobility Assistive Robot Based Human Arm	7
	2.2 Degree of Freedom (DOF) of Human Arm	8
	2.3 Modelling of Human Arm	10
	2.4 Automation Model using Solidworks Software	12
	2.5 Automation Model using MATLAB Software (Simmechanics)	14
	2.6 Proportional-Integral-Derivative (PID) Controller	16
	2.7 Proportional –Derivative (PD) Controller	19
	2.8 Ziegler-Nichols Tuning Technique	19
	2.9 Summary	21
3	**METHODOLOGY**	
	3.1 Project Methodology	22
	3.2 Design Human Arm Model	25
	3.3 Transfer Human Arm Model from Solidworks to MATLAB Simulink via Simmechanics	32
	3.4 Design and Develop Controller of Human Arm Model	34
	3.4.1 Obtaining Human Arm Model Transfer Function	40
	3.4.2 Ziegler-Nichols Tuning Techniques	41
	3.5 Summary	51

CHAPTER	TITLE	PAGE
4	**RESULT AND DISCUSSION**	
	4.1 Human Arm Model Design	52
	4.2 Importing Human Arm Model via Simmechanics	54
	4.3 Develop and Design Control System of Human Arm Model	61
	4.3.1 The performance of controller and controller error on human arm model joints	65
	4.3.2 The comparison of performance controller trajectory and error on the human arm joints for $K_u = 1$	75
	4.3.3 The comparison of performance controller trajectory and error on the human arm joints for $K_u = 2$	78
	4.4 Discussion	80
	4.5 Summary	81
5	**CONCLUSION**	
	5.1 Conclusion	83
	5.2 Future Works	85
REFERENCES		**86**
APPENDIX		**92**

LIST OF TABLE

TABLE	TITLE	PAGE
2.1	Human Arm Joint Movement	10
2.2	Effects of PID controller	18
2.3	PID controller parameters by Ziegler-Nichols Method	20
3.1	Parameter of Human Arm Model	31
3.2	Parameter of Open-Loop Ziegler Nichols Tuning method	43
3.3	Parameter of Closed-Loop Ziegler-Nichols Tuning methods	45
3.4	Trajectory of human arm joint	46
3.5	Tuning parameters of PID controller and PD controller	47
4.1	Parameter of Ziegler-Nichols (Z-N) Tuning Method	63
4.2	The (Z-N) tuning value on shoulder joint for $K_u = 1$	65
4.3	The (Z-N) tuning value on elbow joint for $K_u = 1$	68
4.4	The (Z-N) tuning value on shoulder joint for $K_u = 2$	70
4.5	The (Z-N) tuning value on elbow joint $K_u = 2$	72
4.6	Data of PID and PD controller for $K_u = 1$	75
4.7	Data of output error of PID and PD controller for $K_u = 1$	77
4.8	Data of PID and PD controller for $K_u = 2$	78
4.9	Data of output error of PID and PD controller for $K_u = 2$	79

LIST OF FIGURE

FIGURE	TITLE	PAGE
1.1	Scope of the Project	5
2.1	7 DOF in a human arm	9
2.2	Four basic degrees of freedom of the arm	9
2.3	Geometric parameters of human artificial arm	11
2.4	A CAD model of articulated robot in Solidworks	14
2.5	Simmechanics diagram created in Simulink	15
2.6	Block diagram of PID controller	17
3.1	Flow chart of the project	24
3.2	Solidworks 2015 software	25
3.3	Example of body part design	26
3.4	Complete design of body part	26
3.5	Design of the box part	27
3.6	Development of the shoulder joint	28
3.7	Design of the knuckle pin part	29
3.8	Design of knuckle joint	30
3.9	Human arm model	30
3.10	MATLAB and MATLAB Simulink	32
3.11	Model is import to MATLAB	33
3.12	Simulink output of human arm model	33
3.13	Model is run in Simulink	34
3.14	Subsystem of the controller	36
3.15	PID Controller in Subsystem	37
3.16	Human Arm Model System	38
3.17	Steps to design the controller	39
3.18	Open-loop output linear points on shoulder joint	40
3.19	Closed-loop system on shoulder joint	41
3.20	Reaction Curve	42
3.21	Initial response of elbow joint	46
3.22	An example response of PID controller on elbow joint for $K_u = 1$	47
3.23	An example response of PD controller on elbow for $K_u = 1$	48
3.24	An example of elbow trajectory of both controllers for $K_u = 1$	48
3.25	Response error of PID controller on elbow for $K_u = 1$	49
3.26	Response error of PD controller on elbow 1 for $K_u = 1$	50
3.27	An example of elbow error for $K_u = 1$	50
4.1	Exploded view of human arm model parts	53
4.2	Human arm model	54
4.3	Imported of one arm model	56
4.4	Imported of two arm model	57

FIGURE	TITLE	PAGE
4.5	Arm model Simulink output an additional blocks (PD Controller)	59
4.6	Arm model Simulink Output and additional blocks (PID Controller)	60
4.7	Controller in Subsystem	61
4.8	Initial response of the joint	62
4.9	PID controller division	63
4.10	The trajectory of shoulder joint for $K_u = 1$	65
4.11	Trajectory error of shoulder joint for $K_u = 1$	66
4.12	Trajectory of elbow joint for $K_u = 1$	68
4.13	Trajectory error of elbow joint for $K_u = 1$	69
4.14	Trajectory of shoulder joint for $K_u = 2$	71
4.15	Trajectory error of shoulder joint for $K_u = 2$	72
4.16	Trajectory of elbow joint for $K_u = 2$	73
4.17	Trajectory error of elbow joint for $K_u = 2$	74

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
A	Gantt chart for Final Year Project I	92
B	Gantt chart for Final Year Project II	93

CHAPTER 1

INTRODUCTION

This chapter describes the mobility support of the exoskeleton (upper limb). The project emphasizes on the development and control of assistive exoskeleton which is upper limb part for mobility support. The project brief's explanation will be explain in the project background. The problem statement and motivation relating to this project will be explain in this chapter .Finally, objective and scope are listed accordingly.

1.1 Project Background

Exoskeleton used to support and defends human or animal's body [1]. The region in an animal extending from the deltoid region to the hand, including the arm, axilla and shoulder called as upper extremity [2]. The upper limb enables to grip, write, lift and throw among many other movements. The upper limb has been shaped by evolution, into a highly mobile part of the human body. The upper limb has developed for stable mobility support. This project focuses on the development of assistive human robot arm which is suitable for mobility support. Human arm refers to the structures part extended from shoulder to the elbow.

An arm is an internal mechanical structure which has joints that correspond to the human arm. By wearing the exoskeleton arm, mechanical power can be transmitted to the human arm as a result of the physical contact between the two [3]. People are unable to do their daily activities without an arm. This is the reason why arm is very useful to people.

The movement of the upper limb or upper extremity will be control using Proportional-Integral -Derivative (PID) and Proportional-Derivative (PD) controller. PID controller are common technologies used nowadays in order to design a controller. PD is a controller type which is from PID division. The other controller type from PID is Proportional-Integral (PI) controller. The PID controller only able to control low torque and speed which is can be define as simple system and it using manual panel of controller interface. PID controller are design to eliminate the need for continues operator attention. The integral term is most effective at low frequencies, the proportional term are at moderate frequencies and integral term are at high frequencies. The profits from the integral term is the reduction of steady state error and the differential term improve the receptiveness stability.

The project presents about the development and control of exoskeleton (upper limb) for mobility support. An assistive robot arm is created to test and develop a mobility support. The project focuses on the movement of human arm. The aim is to help injured and disabilities people to move their arm. People with these problems are the main subject for this project.

1.2 Motivation

Upper limb and arm movement disorders had given the motivation to develop a system for the movement of upper extremities to train its normal movement and give more control freedom in strength and speed. The controller which is attach on the joint of shoulder and elbow is a sophisticated system to study. Thus, more knowledge will be enhanced in engineering aspects. In general, this project uses the advance techniques in designing the model, creating the 3D model in Computer Aided Design (CAD) software. The development of model in related software enable engineers to investigate the movement of the model through the simulation in the controller software. Complete control system will be develop. It will benefit a lot since it use in the term of instrument and control engineering perspective.

1.3 Problem Statement

Strokes and arm injuries are the most factor that affects the movement of arm nowadays. Approximately 15 million people worldwide have a stroke each year. In a worldwide population, around 1 in 6 people in this world have a stroke in their lifetime [4]. Other than stroke, arm injuries are caused by cuts and laceration injuries. U.S. Bureau of Labor Statistics (BLS) reports that cuts and laceration injuries are the most common work injuries to the hands. The victims to the incidents accounted for 4,120 job transfer or restriction cases in 2012 [5].

In order to solve above problem, a mechanism was created to help affected people to move. So, the development and control of exoskeleton (upper limb) for mobility support has chosen in this project. This project is use to help people with minor stroke and arm injuries to move their affected body part to normal movement. So, the project is done by designing a wearable assistive arm model to activate disabilities arm. Next, if the patient is suitable to use the assistive model, it can train the arm to move in the good posture.

Many upgrading and modification have to be considered in order to achieve the target. So, the problem of the minor stroke and arm injuries is use as an approach to develop and control of human arm. This project will implement on how to investigate the development of an assistive upper limb exoskeleton component for specific mobility support.

1.4 Objective

The project to be developed is to fulfill the configuration and specification of development and control of upper limb or human arm. An assistive human arm model should be develop in the beginning part of the project. Due to shortage of budget, the human arm are able to design using the software tools The software tools that are used in this project is CAD software, Solidworks and simulation software, MATLAB.

The objectives of this project are used as a benchmark to make sure the progress can be carried properly. The "**Development and Control of Exoskeleton (Upper Limb) for Mobility Support**" project objectives can be defined as listed below:

1. Design human arm model using CAD software, Solidworks and transfer to MATLAB via Simmechanics.
2. Develop and design control system of human arm model using Proportional-Derivative (PD) controller and Proportional-Integral-Derivative (PID) controller using simulation software, MATLAB.
3. Validate PID and PD controller performance on human arm model.

1.5 Scope

In order to accomplish the objective of this project and enhanced the effectiveness of this system, few scopes has been considered. The scope of this project is to develop and control the upper extremities exoskeleton that can interfaced using Solidworks and MATLAB software. The scopes are shown as the flowchart in the figure 1.1:

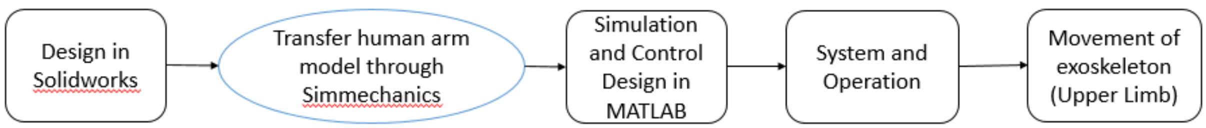


Figure 1.1: Scope of the project

Based on the figure 1.1, the development of exoskeleton (upper limb) model which is human arm begins by the designing a model in Solidworks .The designed of human arm model is transferred to the MATLAB by using the Simmechanics plug-in software.The simulation and control of human arm model happens in the MATLAB software. The system and operation of the model becomes complete as the controller in MATLAB able to control the movement of the model designed in Solidworks.The movement of exoskeleton (upper limb) is completed. The control design in this project is focusing on the PID and PD controller. The scope of the project also involve on the controlling variable value on PID and PD controller. This project focusing on the controlling of human arm movement on sholuder joint and elbow joint.

1.6 Organization of Thesis

The organization of this report is arrange as below:

- **Chapter 2,Literature Review:** This chapter discuss the article of the current and previous research to enhance an idea to do the project
- **Chapter 3, Methodology:** Methodology is one of the chapter discuss the process path of the project until the objectives is achieved. In this chapter, the development of this project is written from designing of human arm in Solidworks until the analysis of human arm movement in the MATLAB.
- **Chapter 4, Result and Discussion:** This chapter shows result of software development and simulation of human arm. In this chapter, the result are also based on objectives stated in the chapter 1.This project will show the result from designing of human arm model, transferring the model via Simmechanics. Finally, testing and simulation of human arm model in MATLAB. The analysis of human arm model is discuss in this chapter as well. The model is analyze based on its trajectory value and error. The value of the arm model trajectory will be discussed as well in this chapter.
- **Chapter 5, Conclusion:** This chapter summarize all the project flow from beginning to end. The result obtained also discussed in this chapter .Further project development will be tell in this chapter.

CHAPTER 2

LITERATURE REVIEW

In this chapter, the mobility assistive robot based human arm will be first topic to review. In order to develop the upper extremities, degree of freedom and modelling of human arm are also review in this chapter. Phases to develop human arm by using Solidworks and MATLAB will be clarify in the development of automation modelling part. Finally, the basic concept of PID controller is presented as well in this chapter.

2.1 Mobility Assistive Robot based Human Arm

Falling among the elderly or patients can be avoid by providing a mobility assistive robots which has the ability of physical and equilibrium support [6]. To overcome this situation, a Mobility Assistive Robots (MAR) is developed by joining features. An example of features are robot also derived with challenges in terms of safety. They have to operate in close interaction with humans especially for elderly or patients with cognitive and or physical disorder [7]. This project use to develop an assistive robot of exoskeleton (upper limb) for mobility support. A controller is design in controlling an arm robot. The movement of arm robot refers to the human arm. An example can be made based on the above statement is the development of the proposed rehabilitation robot called as NTUH-ARM. The rehabilitation robot provides 7 degree-of- freedom (DOF) motion. The robot runs as a subject to an inherent mapping between the 7 DOFs of the robot arm and the 4 DOFs of the human arm [8].

The mobility assistive robot is focuses on the development of arm robot. The movement of the arm robot are referring to the human arm.

2.2 Degree of Freedom of Human Arm

Human arm are the body parts of human body that able to move in seven degrees of freedom (7 DOF).The seven degrees freedom of human arm includes 3 in the shoulder joint, 1 in the elbow joint and 2 in the wrist. The shoulder joint consists of shoulder adduction/abduction (add/abd), extension/flexion (ext/flx) and lateral/medial (lat/med) rotation. In elbow joint, the flexion and supination/pronation (sup/pron) occurs in the forearm. Next, wrist extension/flexion (flx/ext) and adduction/abduction (add/abd) occurs at wrist joint while supination/pronation (sup/pron) of the forearm is formed by ulna and radii bones in the forearm. Elbow joint and wrist joint are located in the forearm [9]. Figure 2.1 illustrate the seven degrees of freedom (7 DOF) of human arm movement.

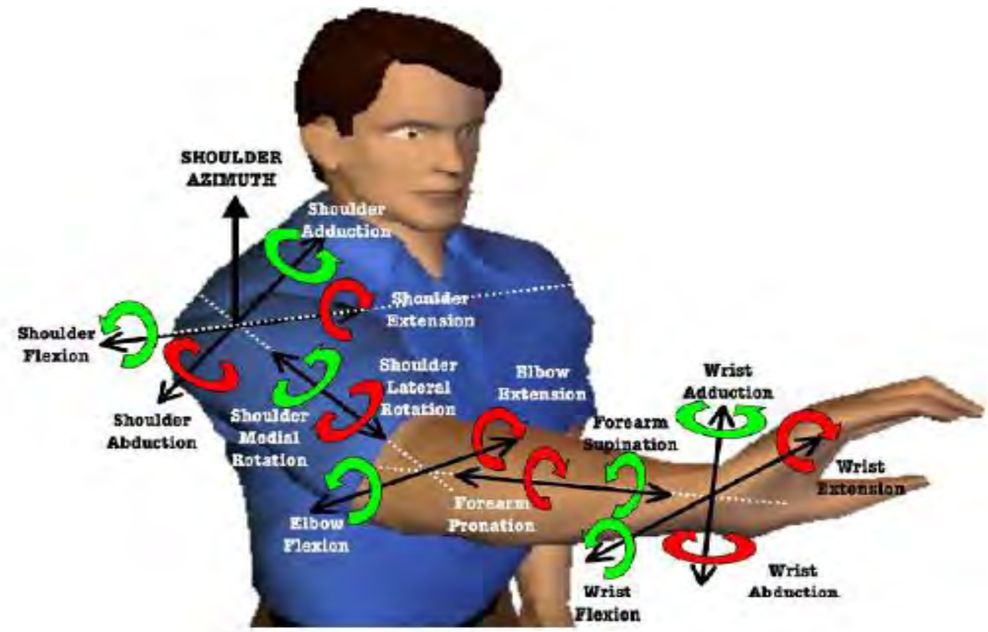


Figure 2.1: 7 DOF in a human arm [9].

The human arm have an abilities to move into basic four degrees of freedom (4 DOF).Four Degrees of freedom includes the shoulder abduction / adduction. Shoulder flexion / extension and internal / external rotates as well as the elbow flexion / extension motion [10]. Four basic degrees of human arm are shown in figure 2.2.

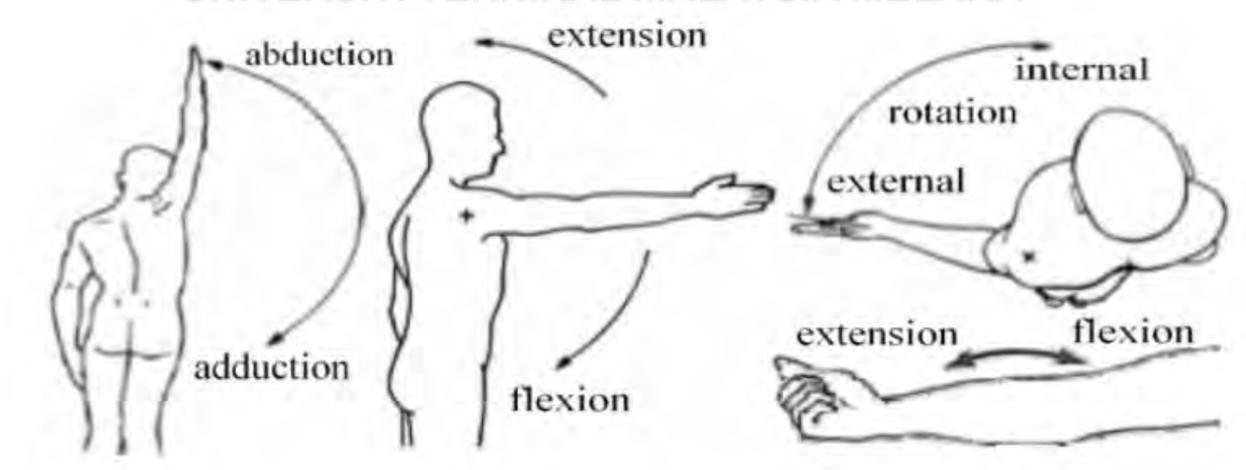


Figure 2.2: Four basic degrees of freedom of the arm .a) Adduction/Abduction b) Flexion/Extension c) External/Rotation/Internal d) Extension/Flexion [10]

The maximum degrees of human joint movement are shown in table 2.1.

Table 2.1: Human Arm Joint Movement.

Part	Flexion	Extension	Abduction	Adduction	Supination	Pronation
Shoulder	-180^0	$+80^0$	-180^0	$+50^0$	$+90^0$	-90^0
Elbow	-10^0	$+145^0$			$+90^0$	-90^0
Wrist	-90^0	$+70^0$	-15^0	$+40^0$		

This project focuses on the development of human arm in 2-DOF.The 1-DOF occurs in the shoulder joint which is extension/flexion (ext/flx) with angle of extension, ext is $+80^0$ and flexion is -180^0.The another 1-DOF occurs in the elbow joint which is also extension/flexion (ext/flx) with the angle of extension,ext is -10^0 and angle of flexion,flx is $+145^0$.

2.3 Modelling of Human Arm

A simple representation of a system at some particular point defined as a model. A modelling is use to promote understanding of the real system [11]. The main point of modelling is to enable the analyst to predict the effect of changes to the system.

Human arm are formed in physical modelling. To produce a physical model, a mathematical modelling are the basic requirements to produce human arm model. The mathematical modelling includes kinematics modelling and dynamics modelling.

The human artificial arm considered as shoulder and elbow with the linkage of two degree of freedom kinematic and dynamic structure. The structure having one degree of freedom for both shoulder and elbow individually. The structure have the link lengths of arm, ai. Link masses, mi and inertia, Ii. Proximal end of joints are taken as the center of gravity locations, Ri. Wrist is considered as an integral part of forearm. The mathematical modeling of all links are considered rigid and friction's joint is neglected [12]. Figure 2.3 shows the geometric parameters of human artificial arm.

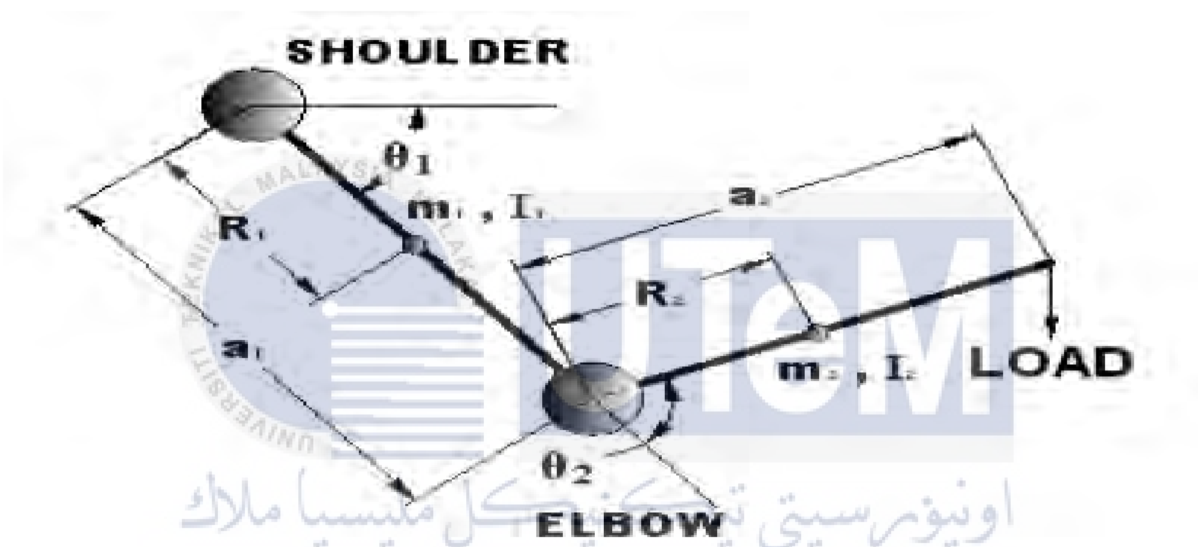


Figure 2.3: Geometric parameters of human artificial arm [12].

Two types of modelling that can be represented by mathematical modelling are kinematics modelling and dynamic modelling. Research paper [13] uses two types of mathematical modelling which are kinematics modelling and dynamics modelling. Both mathematical modelling are relate to each other to produce the movement of human arm. In kinematics modelling, Denavit-Hartenberg (DH) method is used for the positional analysis to determine the end position in 3D space. In dynamics modelling, Lagrange-Euler method is used for the dynamic analysis [13].

However, human arm also can be done by using one mathematical modelling, dynamics modelling. This statement can be proved by referring to the research [14] and research [15]. In research [14], the article presents the analysis of mathematical model on human arm. The human arm is dynamics model is analyze in terms of observable. The research is carried out to determine the possibility of arm tracking. The arm model is a two link manipulator which are able to move in two planes which is vertical plane and horizontal plane. After the linearization and neglecting the representing part,the model is analyzed. In general terms the motion of a rigid body can be described by vector nonlinear equation resulting from Euler-Lagrange formula of dynamics modelling.

Next, the Lagrangian method is applied in dynamics modelling for research .In research [15], the approach is used to develop the mathematical model arm. According to this approach, free-body diagram is drawn according to the hand movement. The degree of freedom is considered two since two parts of arm are actively moving. The Lagrangian is calculated from both potential and kinetic energy.

In this project, the modelling of human arm are done by referring both kinematics modelling and dynamics modelling. DH method will be used in kinematics modelling while Euler-Lagrange method will be used in dynamics modelling.

2.4 Automation Model using Solidworks Software.

Solidworks is a computer-aided design software which is consist of mouse-driven graphical user interface to help engineers and designers to visualize and communicate 3D models of manufactured objects [16].

An example can be made by using Solidworks is designing the articulated robot coordinate system which is in research [17]. In order to prepare proper kinematic relation to a simulation model, each element of physical model should connected to articulate robot coordinate systems. The individual elements of physical robot model is synthesized by using kinematic constraints. The motion of two neighborhood elements of kinematic constraints is defined in the number of degrees of freedom (DOF).In articulated robot case, each robot joint have one DOF of rotation. Each robot joint are assembled together in the Solidworks. The assembly of the articulated robot joints are done as follows:

a. One concentric mate between the cylindrical joint hinges of the two robot elements is applied. The joint DOF are reduced from six DOF to two DOF by the translation and rotation along rotation axis.
b. An appropriate coincident mate between two planes normal to the cylindrical axis is used. Thus, the joint DOF are finally reduced to only one DOF by rotation around the hinge axis of the joint.
c. The complete model of articulated robot development is visualized on the figure 2.4.

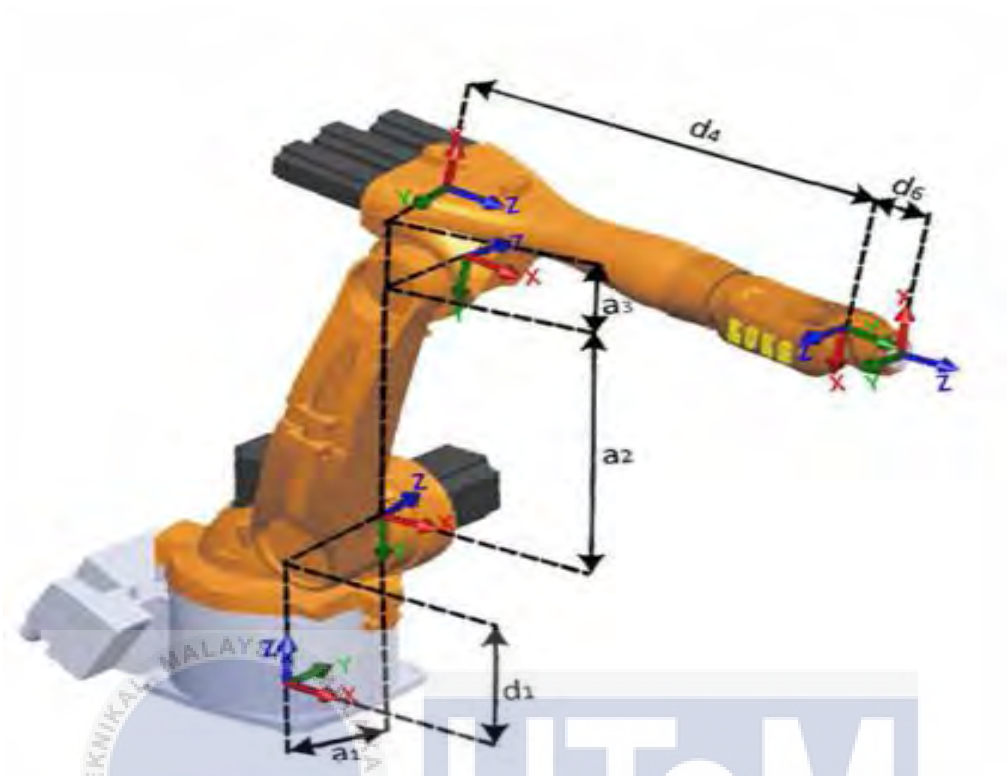


Figure 2.4: A CAD model of articulated robot in Solidworks [17].

The human arm model is designed in Solidworks. Two human arm model is designed in Solidworks which one arm model (stage 1 model) and two arm model (stage 2 model). Completed design arm is exported out to MATLAB.

2.5 Automation Model using MATLAB Software (Simmechanics).

The automation model is imported to the MATLAB software using the Simmechanics. Simmechanics is use to construct a block diagram for modeling and simulating mechanical system by raising the standard Newtonian forces and torques. A graphical way by connected block can be presented in mechanical system. This method can save time and effort the model. The block libraries consist the block set of bodies, joints, sensors, actuators, constraints and drivers, and force elements. The models in Simmechanics can be interfaced using Simulink block diagrams. All physical, geometrical and kinematics components relationships can be represent in physical modeling directly [18].

The model is exported from the Solidworks model by saving in the XML file. The other way is through dragging and connecting the corresponding module in the Simmechanics library according to the simplified model.

To actuate the control system of the closed-loop simulation, several features need to consider to modify the transfer block diagram. The features are joint actuator, joint sensor and several controllers. The controllers could be PID controller, PI controller, PD controller or Fuzzy controller. The example diagram are shown in figure 2.5.

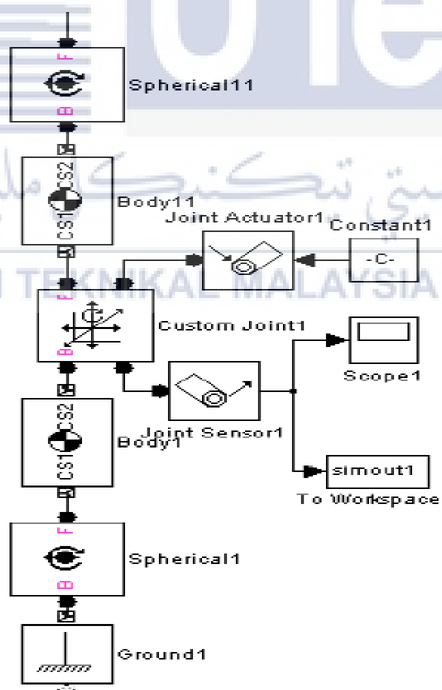


Figure 2.5: Simmechanics diagram
created in Simulink [18].

In this project, the model of human arm is imported to MATLAB by typing a code in the command window. The model is then transferred to the Simulink by helping from SimMechanics link and appears as the block diagram. The block diagram in the MATLAB is represented as the human arm model in the Solidworks software. SimMechanics is plug-in software that enables the human arm model to transfer the developed model from Solidworks to MATLAB.

2.6 Proportional-Integral-Derivative (PID) Controller

In control theory, a controller defined as a device that reduces the difference between the desired value (called as set point or reference) and measurement (process variable) by manipulation of an actuator. Sophisticated controllers are used in several measurement actuators.

Proportional-integral-derivative (PID) controller has been widely used in the industry because of its simple structure and strong performance in a wide range of operating environments. The parameters, denoted as *Kp, Ki,*and *Kd* correspond to the proportional, integral and derivative values respectively.

Interpreted in terms of time, $K_p$depends on the present error,K_i on the addition of past errors, and K_d is an expectation of future errors, based on existing of rate of change [19]. The diagram of general PID system is composed of proportional parameter, integral parameter and differential parameter.

The expression PID controller is

$$U(t) = K_p \left[e(t) + \frac{1}{T} \int_0^t e(t)dt + T_D \frac{de(t)}{dt} \right] \quad (2.1)$$

where K_P, I_T and D_T are proportional parameter, integration time constant and differential time constant of the controller [20]. The arrangement of the above equation is showed in the block diagram 2.6.

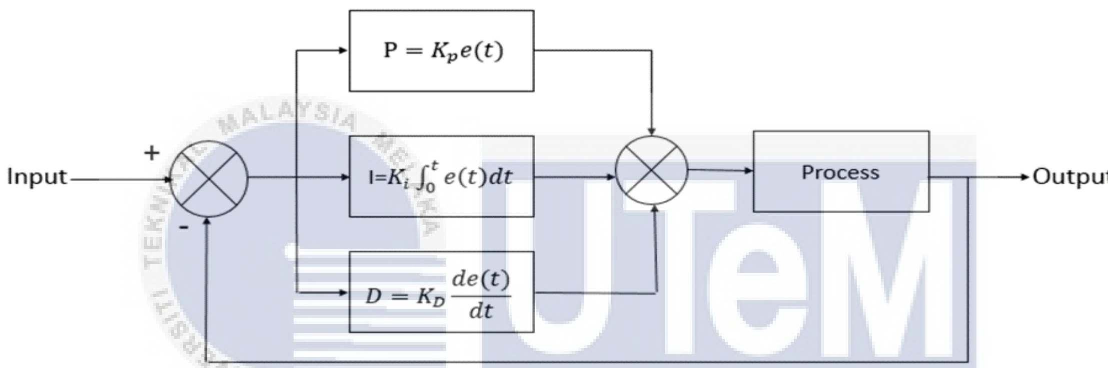


Figure 2.6: Block diagram of PID controller

In a research, PID controller is used to design the human arm-robot manipulator coordination problem in unstructured workspace. The PID controller is use to analyze the stability of the closed-loop system by taking the result from the Hennite- Biehler theorem. By using this technique, the effort of arm can be reduce during simple manipulator tasks [21].

In research [22], PID controller use to develop an asymptotic stability. This class known as robotic manipulators. A simulation was used to evaluate the system performance given the theoretical results of the controller's parameters with a unique exoskeleton system (EXO-UL7). The simulation also verify the semi-global asymptotic stability of the system.

A proportional controller (K_p) will have the effect of reducing the rise time and will reduce steady-state error. An integral control (K_i) will have the effect of eliminating the steady-state error. However, it may make the transient response worse. A derivative control (K_d) will have the result of increasing the stability of the system. K_d also result in reducing the overshoot and improving the transient response. Effects of each of controllers K_p, K_d and K_i on a system are briefed in the table shown 2.2.

Table 2.2: Effects of PID controller [23]

Response	Rise Time, T_r	Overshoot,OS	Settling Time, T_s	Steady-State Error,e_{ss}
K_p	Decrease	Increase	Small Change	Decrease
K_i	Decrease	Increase	Increase	Eliminate
K_d	Small Change	Decrease	Decrease	Small Change

The relationships should not accurate. This is because the response of K_P, K_i and K_d are different to each other. The changing one of these variables can change the effect of the system. The table 2.2 only used as a reference when determining the values for K_P, K_i and K_d for this reason [23].

PID controller was selected as the controller in the simulation of human arm. In this project, the PID controller will be placed in shoulder-joint and elbow-joint. The attachment of arm model and PID controller will creates a system. This will allow human arm to move in controllable conditions.

2.7 Proportional – Derivative (PD) Controller

PD controller could be classified as a special class of controller. This is because PD controller can provides lower overshoot for integrating processes with delay compared PID controller [24]. A simple PD controller suits to stabilize any kind of rigid manipulator about a reference position, provided that the gravitational forces are compensated [25]. In spite of numerous applications of PD controller in robotics, the number tuning relations are reported for PD controller is very little in conventional control [24]. PD control had better tracking error response with lower gains [26].

In this project, the PD controller will be placed in shoulder-joint and elbow-joint. The position of the controller are set as the PID controller. The combination of controller and arm model will creates a system. This will allow human arm to actuate with the desired movement.

2.8 Ziegler-Nichols Tuning Technique

In the present analysis, the Ziegler Nichols method of controller tuning and describing function method is applied. This technique is very influential. The Ziegler Nichols is developed using different tuning rules by simulation of different process and correlating parameters of controller with step response features [27]. In this work, the tuning techniques for PID controllers based on Ziegler-Nichols method. The tuning of the two PID controllers was conducted separately, being primarily tuned the multivariable PID controller responsible for the stability around the horizontal axis [28].

The transfer function of a PID control is given as

$$C(s) = K_p + \frac{K_i}{s} + K_d s \quad (2.2)$$

where K_p , K_i and K_d are controller parameters, which can tuned by designers. One approach to tune the parameters is Ziegler-Nichols method. The parameters are listed in Table 2.3, where K_u, the gain margin for loop stability and T_u is the oscillation frequency at the stability limit [29].

Table 2.3: PID controller parameters by the Ziegler-Nichols method [29]

Controller Type	K_p	K_i	K_d
P	0.5 K_u		
PI	0.45 K_u	$2\frac{K_p}{T_u}$	
PD	0.8 K_u		$\frac{K_P T_u}{8}$
PID	0.6 K_u	$2\frac{K_p}{T_u}$	$\frac{K_P T_u}{8}$

Ziegler Nichols tuning method was used to tune the simulation of human arm. PID and PD controller was selected as the main controller in the simulation of human arm. In this project, the Ziegler –Nichols tuning method will be used on PID and PD controller which is placed in shoulder-joint and elbow-joint. The combination of two controller will creates a system. This will allow human arm to actuate with the desired trajectory.

2.9 Summary

To develop a human arm assistive robot, modelling of human arm and degrees of freedom is considered in order to design the model. This subtopic is important in order to design the model using Solidworks and transfer the model to the MATLAB. The software development .The research article regarding this topic have been reviewed. The benefits of controller helps to the selection of the controller. Based on article reviewed, the PID and PD controller have several advantages in the industry. A tuning method was used in order to stabilize the response of the system. Ziegler-Nichols tuning method is an example to tune the response of the system in order to stabilize the response thorough the set point. The literature review is a proof of the project development.

CHAPTER 3

METHODOLOGY

This chapter describes about the project methodology. Methodology is created to show the path of this project. Methodology helps to make sure that the project is developed in systematically, smoothly and well organized in order to obtain a result. When the switching is performed, the network needs to be maintained in radial form.

3.1 Project Methodology

The project methodology begins by study the upper limb part of human body. The upper limb part that is focused for this project is the movement of human arm. In general, upper limb are the main mobility support that generates the movement of human arm. The upper limb area includes from shoulder to hand. The upper limb part that is focused in this project are shoulder, upper arm, elbow and forearm. Two joints of human arm that consist in this project are shoulder joint and elbow joint. All of this part of human arm are study based on the aspect of degree of freedom (DOF).

The flow of project begins by designing human arm model. The 2-DOF of human arm model is exist in physical modelling. The physical modelling basically was created based on mathematical modelling. The mathematical modelling includes by determining the parameters of the model and derivation of the formula for specific model. Mathematical modelling determined from the kinematic modelling and dynamic modelling. The project does don't carry on the calculation or determination of the mathematical modelling. This is because all of the human arm parameters have discovered from the literature review of research [10] in the previous chapter.

The model was design based on the mechanical joint which is called as revolute joint and knuckle joint. The revolute and knuckle joint are designed by using the Solidworks software (version 2015). Completed designed human arm based revolute and knuckle joint is then imported to the MATLAB (version 2016b) Simulink using Simmechanics plug-ins. The usage of Simmechanics is used to transfer the human arm model from Solidworks to MATLAB. The Solidworks software is used for designing the arm model while MATLAB Simulink is used for simulation of the designed model. The transferred human arm model is then existed in the MATLAB Simulink as Simulink output of arm model. Next, the model is test for its stability. If the model is unstable or static (not move), some modification must be made. Modification are not required if the model are stable. Next, the linear analysis is done in order to find the model transfer function. The reason in finding transfer function is to describe the input and output of the system.

Finally, a controller are designed in order to control the movement of arm model. This part is included in simulation in MATLAB. In this project, two types of controller was used to actuate the movement of human arm. There are Proportional-Integral-Derivative (PID) and Proportional-Derivative (PD) controller. The design are made in the MATLAB Simulink. Control of human arm has its own problem because it doesn't have limit of degree of freedom. An example can be made by referring to the movement of shoulder. The shoulder rotation has maximum flexion / extension is $-180^0 /+80^0$ .In this problem, the model is rotate full 360^0. So, this is the main reason why a controller design is important in this project. After the controller design, the system is analysis in terms of trajectory and error. Some declaration on the controller will be done which controller is suitable for the joints. The validation of the controller will be validate either PID or PD controller is better for shoulder or elbow joint. The PID and PD controller is expected to provide better system performance and transient response with specific system requirements which are small steady-state error, small overshoot, faster rise and settling time. The project methodology is summarized in the flow chart shown in figure 3.1.

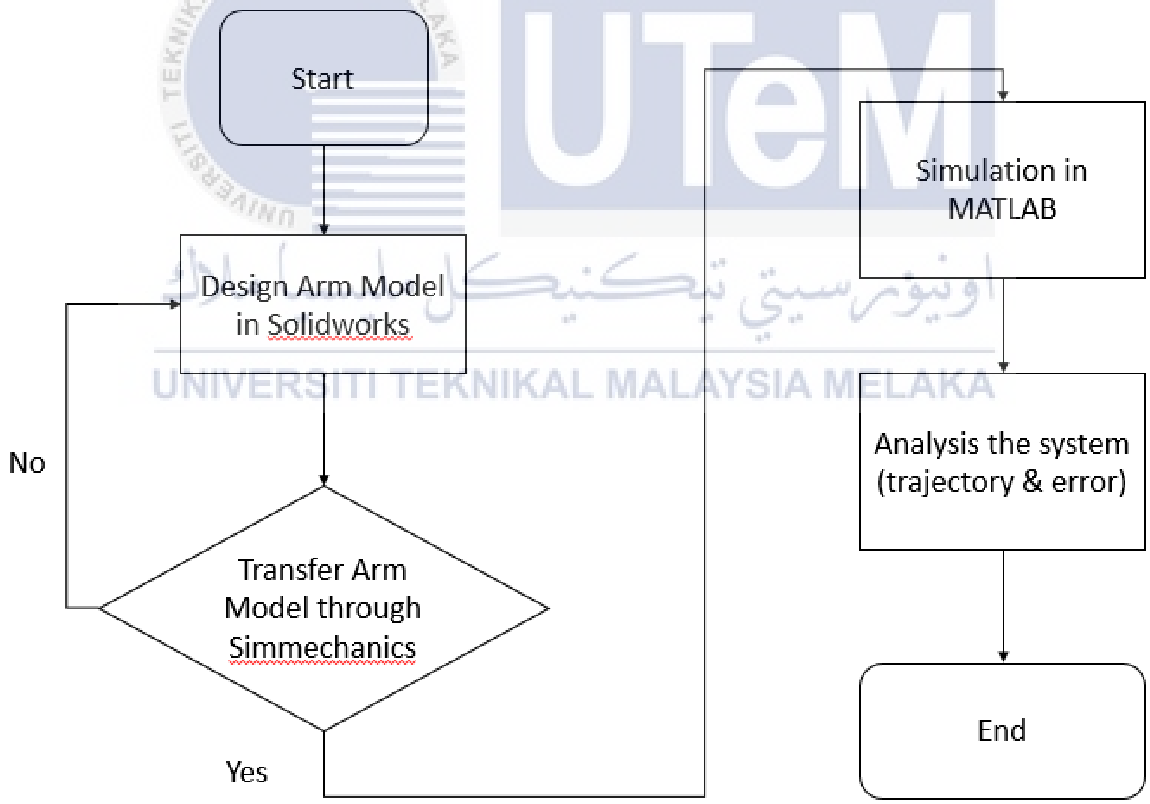


Figure 3.1: Flow chart of the project.

3.2 Design Human Arm Model

The human arm model are design using Solidworks software. The version of Solidworks used in this software is version 2015. SolidWorks is a computer-aided design (CAD) software which runs on the Microsoft Windows platform. Solidworks uses a parametric structures to create models and assemblies. Figure 3.2 shows an example of the Solidworks software.

Figure 3.2: Solidworks 2015 software

The project begin by design the human arm which consist of two joints. The first joint is located in the joint. The second joint is located at the elbow joint. The shoulder joint is constructed using the revolute joint and the elbow joint is constructed using knuckle joint.

1. Design of Shoulder Joint.

Two parts of component required to design the shoulder joint is body and box part. The shoulder joint are design using revolute joint. Revolute joint or known as hinge joint is one-degree-of-freedom kinematic pair used in mechanisms. Revolute joints provide single-axis rotation function used in many places such as door hinges, folding mechanisms, and other uni-axial rotation devices. Body parts are the first step the revolute joint is designed. Figure 3.3 shows the design of the body part.

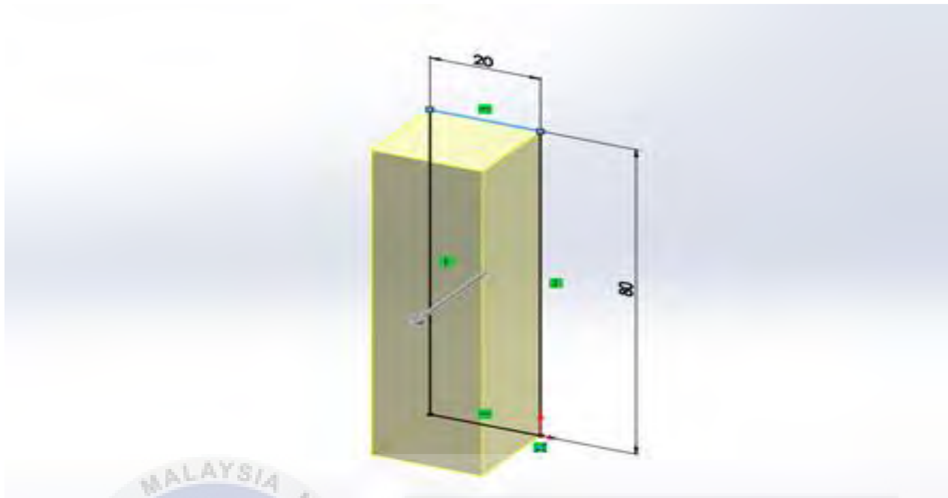


Figure 3.3: Example of body part design

By using all of the instructions in the Solidworks software, the body part is completed as shown in the figure 3.4.

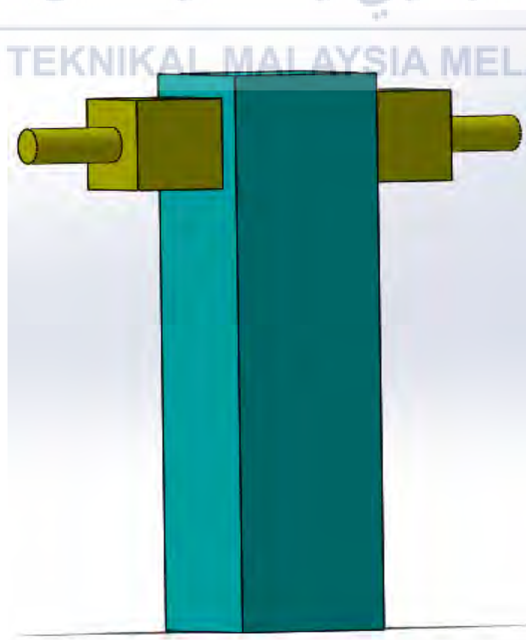


Figure 3.4: Complete design of body part

The second part design is the box part. The box part are the extension of the fork end part. Figure 3.5 shows the development of the second part of shoulder joint.

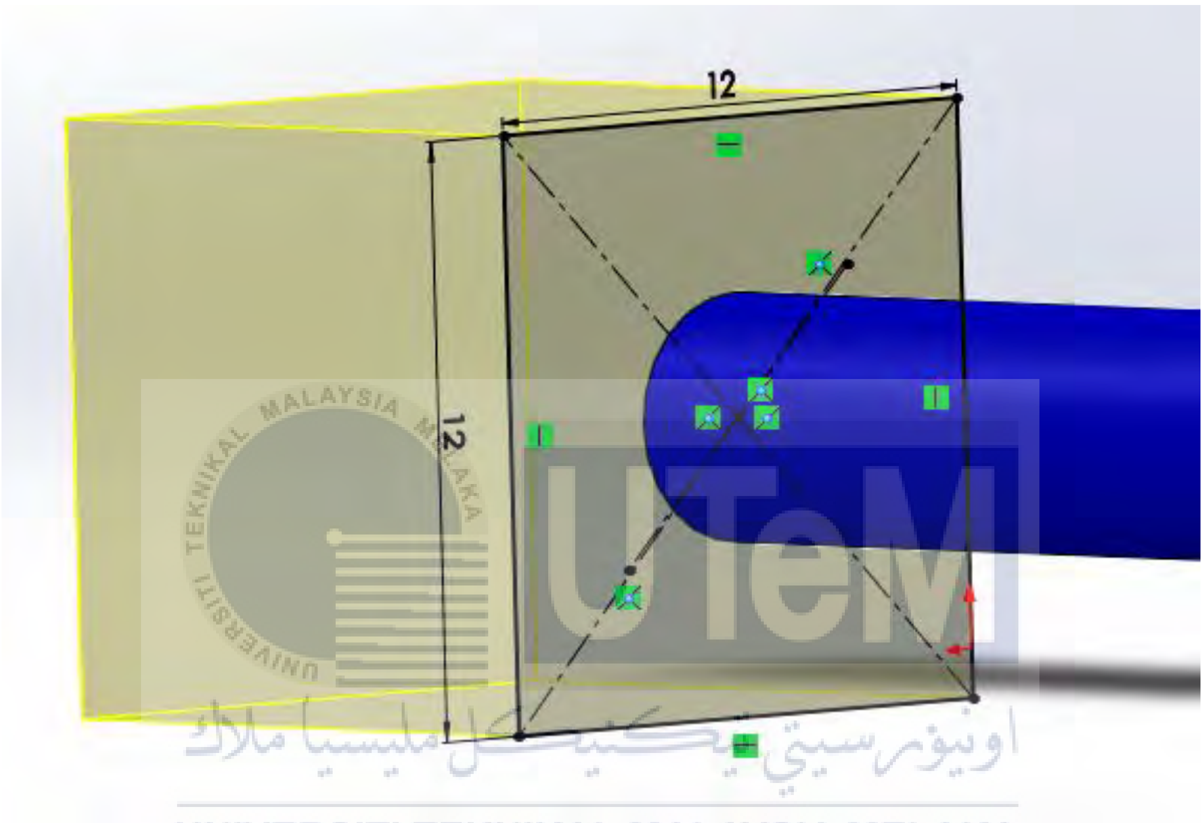


Figure 3.5: Design of the box part

Finally, the assembly between box part and the body part helps the completion of the shoulder joint based revolute joint. The assembly part of the shoulder joint are uses two mating process, concentric and coincidence. Concentric mate is a mate between two concentric features. Second step is coincident mate. Coincident mate is a mate between two features that the parts to coincide with each other. The concentric mate occurs at the inner part of box part and body part. The coincident mate occurs at the surface of the box part and body part. Figure 3.6 the development of the shoulder joint.

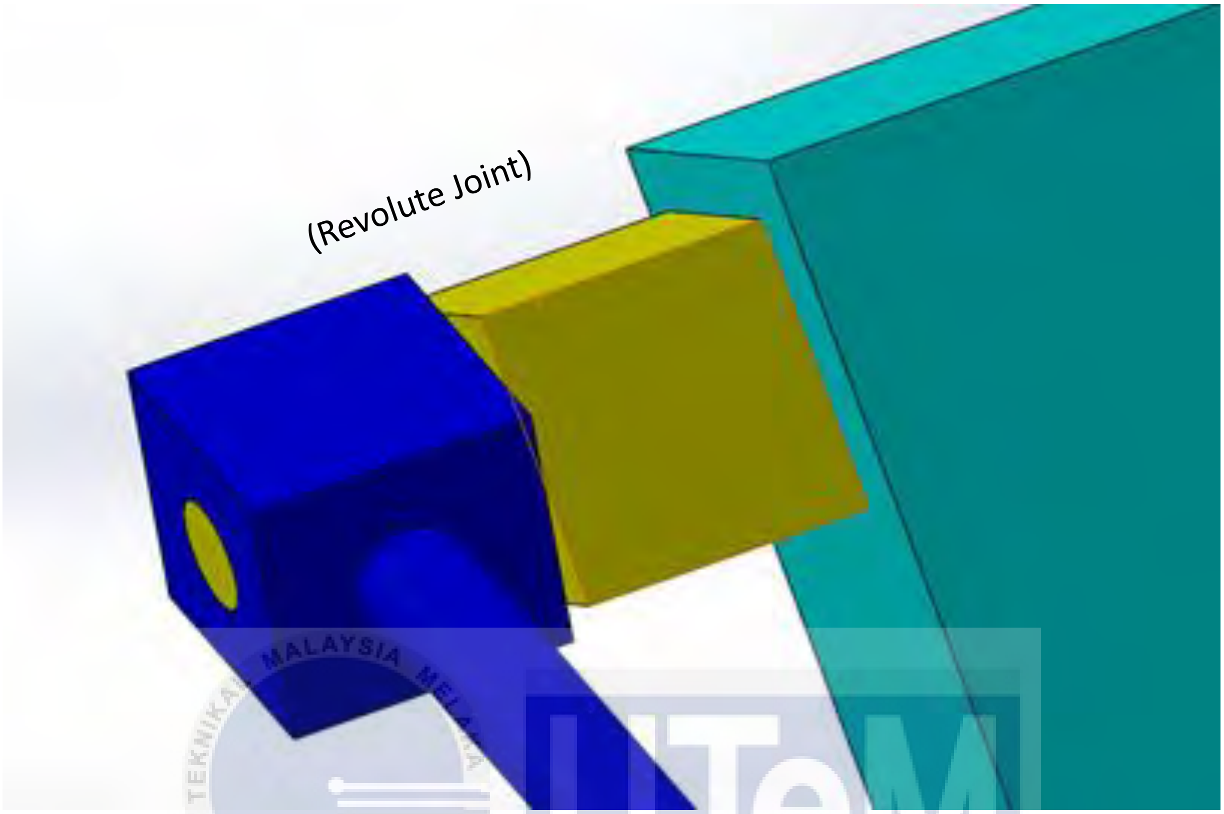


Figure 3.6: Development of the shoulder joint

2. Design of the elbow joint.

The design of the elbow joint are based on the mechanical joint which called as knuckle joint. Knuckle joint is a type of mechanical joint used in structures, to connect two intersecting cylindrical rods, whose axes lie on the same plane. It permits some angular movement between the cylindrical rods (in their plane).

It is specially designed to withstand tensile loads. There is a requirement of small amount of flexibility or angular moment when it necessary. In this project, knuckle joint represented as the elbow joint which connect between upper arm and forearm.

Figure 3.8 shows an example of knuckle joint design.There are 4 parts of component need to be consider in order to form the joint .There are:

1. Fork End
2. Eye End
3. Knuckle Pin
4. Collar

The fork end part is represent as the upper arm and eye end is represent as the forearm. Knuckle pin and collar are represent as the main ligament that holds both fork end and eye end. The design one of the part of knuckle joint is shown in the figure 3.7.

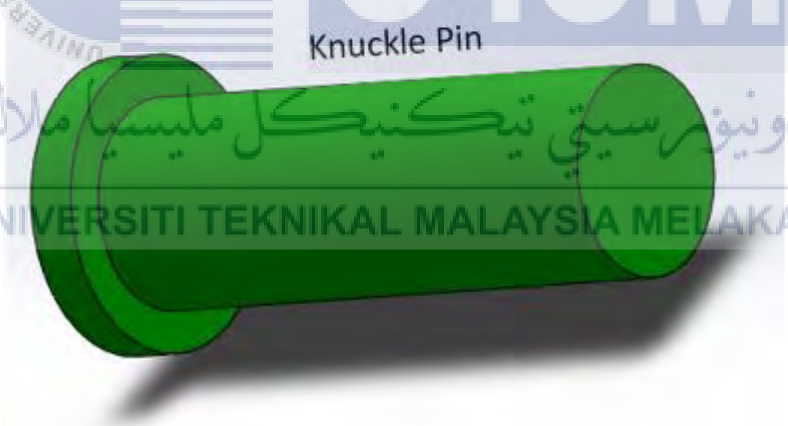


Figure 3.7: Design of the knuckle pin part

The assembly of the all four parts produces a knuckle joint. The assembly part of the elbow joint are uses two mating process (concentric and coincident) which combine together all four parts (fork end, eye end, knuckle pin and collar) to produce the knuckle joint. Figure 3.8 shows the design of the knuckle joint.

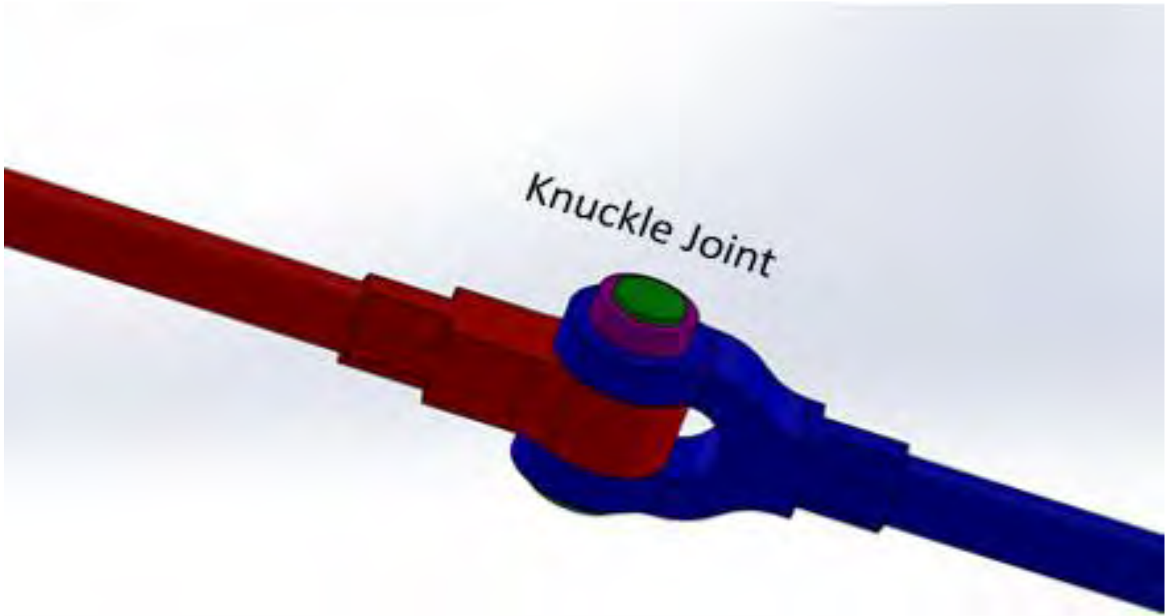


Figure 3.8: Design of Knuckle Joint

The combination between the shoulder joint and elbow joint produces a human arm model. Figure 3.9 shows the development of human arm design in Solidworks software.

Figure 3.9: Human arm model

In summary, the parameter of each of the model are determine. Table 3.1 shows the parameter of the human arm model.

Table 3.1: The parameter of Human Arm model [10]

Model Part	Length (cm)
Body	80.00
Eye End	28.08
Fork End	31.25

The eye end reflecting to the forearm, fork end reflecting to upper arm and body is similar reflects to normal human body. The complete model of human arm are finally is saved. The model is then is transfer to the MATLAB via Simmechanics tool.

3.3 Transfer Human Arm Model from Solidworks to MATLAB Simulink via Simmechanics

The model is export out from the Solidworks and the assembly file is saved as xml file format. The model is saved first as the XML is extension for an Extensible Markup Language (XML).XML file format used to create common information formats and share both the format and the data on the using standard ASCII text.

Figure 3.10: MATLAB and MATLAB Simulink

MATLAB is open in order to import the human arm model to the software. A code is typed in command window as mech_import('Upper Limb Model.xml'). The mech_import function either generates a new simscape multibody model or updates a previously generated specified model. It also generates or updates the associated body geometry files. The model is build and appears in the MATLAB Simulink as block diagram. The block diagram represents the human arm model that is originally designed and exported from Solidworks software. Figure 3.11 shows the process of importing human arm model into the MATLAB Simulink.

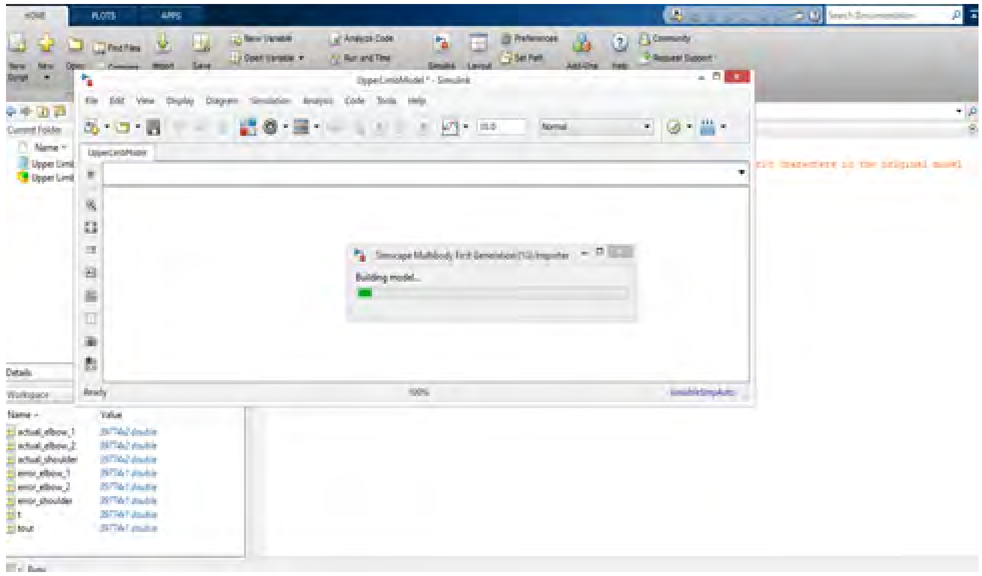


Figure 3.11: Model is import to MATLAB

The Simulink output represents the human arm model that is originally designed and exported from Solidworks software. The Simulink output is not too complex because only one human arm model is designed. The Simulink output in figure 3.12 shows the representation of imported human arm model.

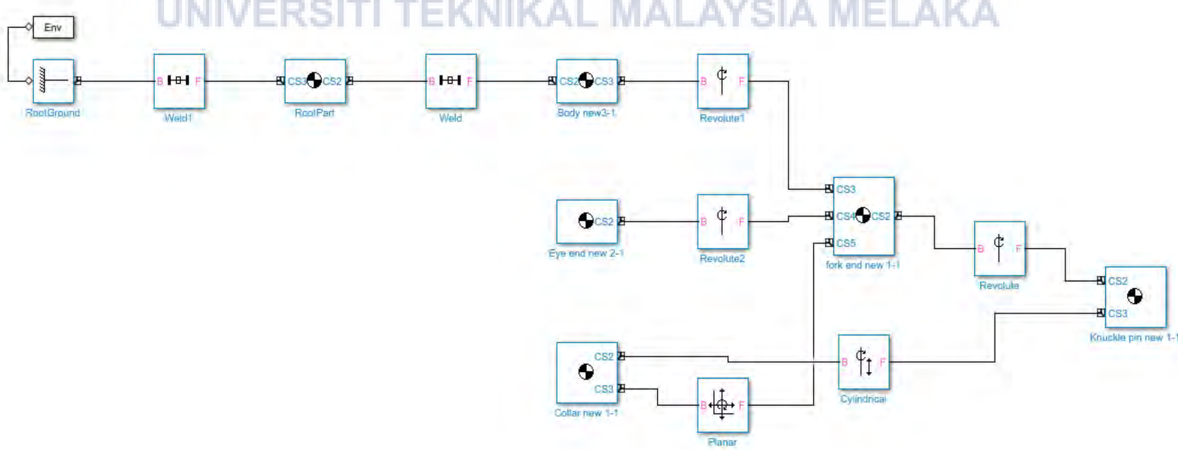


Figure 3.12: Simulink output of human arm model

The human model is test in order to determine whether it move or did not move (static). The preliminary result shows that the model moves in uncontrollable condition. Figure 3.13 shows the reaction of the human arm model as it run in the MATLAB Simulink.

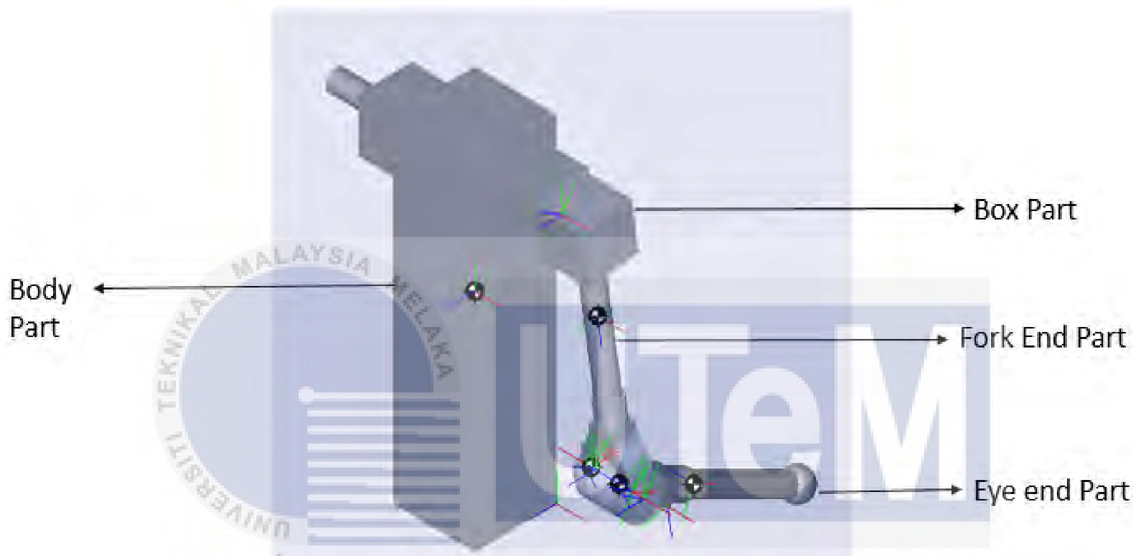


Figure 3.13: Model is run in Simulink

The model are able to move since the revolute block existed in the block diagram. Based on the model shown, one revolute block are existed in shoulder joint while other two revolute block are existed in the elbow joint. So, no modification are required since the system have the revolute blocks. The further steps is adding the additional blocks to the human arm Simulink output. The additional blocks are:

1. Step
2. Sum

3. Workspace
4. Clock
5. Joint Actuator
6. PID & PD controller.
7. Joint Sensor
8. Scope

The function of the step block is to provide a step between two definable levels at a specified time. The step block used to test the mechanism track. The use of the step block in this project is to determine the trajectory tracking of the model system. The function of sum block is to 1 to perform addition or subtraction on its inputs. Sum block is useful in this project because it is able to make the determination whether the system is open-loop system or closed-loop system.

The workspace block is included in the system because the workspace block helps for data collection. The desired response collected in the block will collected and appears in the MATLAB command window in excel form. In order to tally or balance the system with the simulation time, the clock added to the system. The clock block outputs the current simulation time at each simulation step.

The joint actuator blocks is added to the system. Joint actuator is used to actuate a joint primitive with linear/angular position, velocity and acceleration motion signals. Base-follower sequence and joint axis determines sign of forward motion. Inputs are Simulink signals. The joint sensor blocks in this system functions to measures linear/angular position of human arm by referring to the joint primitive.

PID and PD controller are included in the human arm model simulation. The controller are put separately in the MATLAB Simulink software. Two simulation has done in this project.

The first simulation is the simulation of human arm using PID controller and second simulation is the simulation of human arm using PD controller. Both of the PID and PD controller are arrange in the subsystem. Figure 3.14 shows an example of the subsystem of the controller.

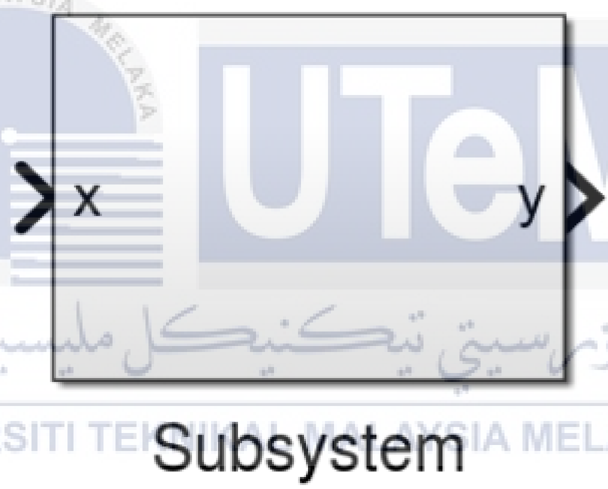


Figure 3.14: Subsystem of the controller

In the subsystem, The PID and PD controller are arrange in the separate blocks. The blocks in the subsystem:

1. PID Controller:
    a. Gain Block
    b. Integrator Block
    c. Derivative Block

2. PD Controller:

   a. Gain Block

   b. Derivative Block

All of the blocks in each subsystem are combined together to form a controller. Figure in 3.15 shows the arrangement of the blocks in the subsystem.

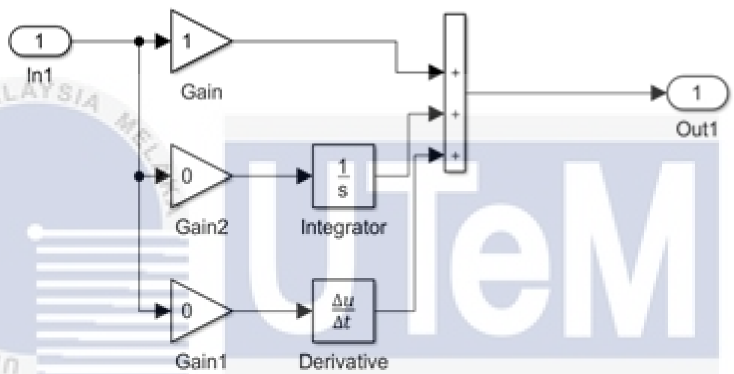


Figure 3.15: PID Controller in Subsystem

The function of scope block is to displays time domain signals with to respect simulation time. The usage of scope in this project is to display the trajectory and error signals of human arm with respect to simulation time. The time is set between 7 seconds. The response of the arm trajectory and error are analyses from scope block.

Finally, all of the additional blocks are combined with the arm block diagram transferred from Solidworks via Simmechanics. Figure 3.16 shows the forming of the system which consist of imported arm block diagram and additional blocks.

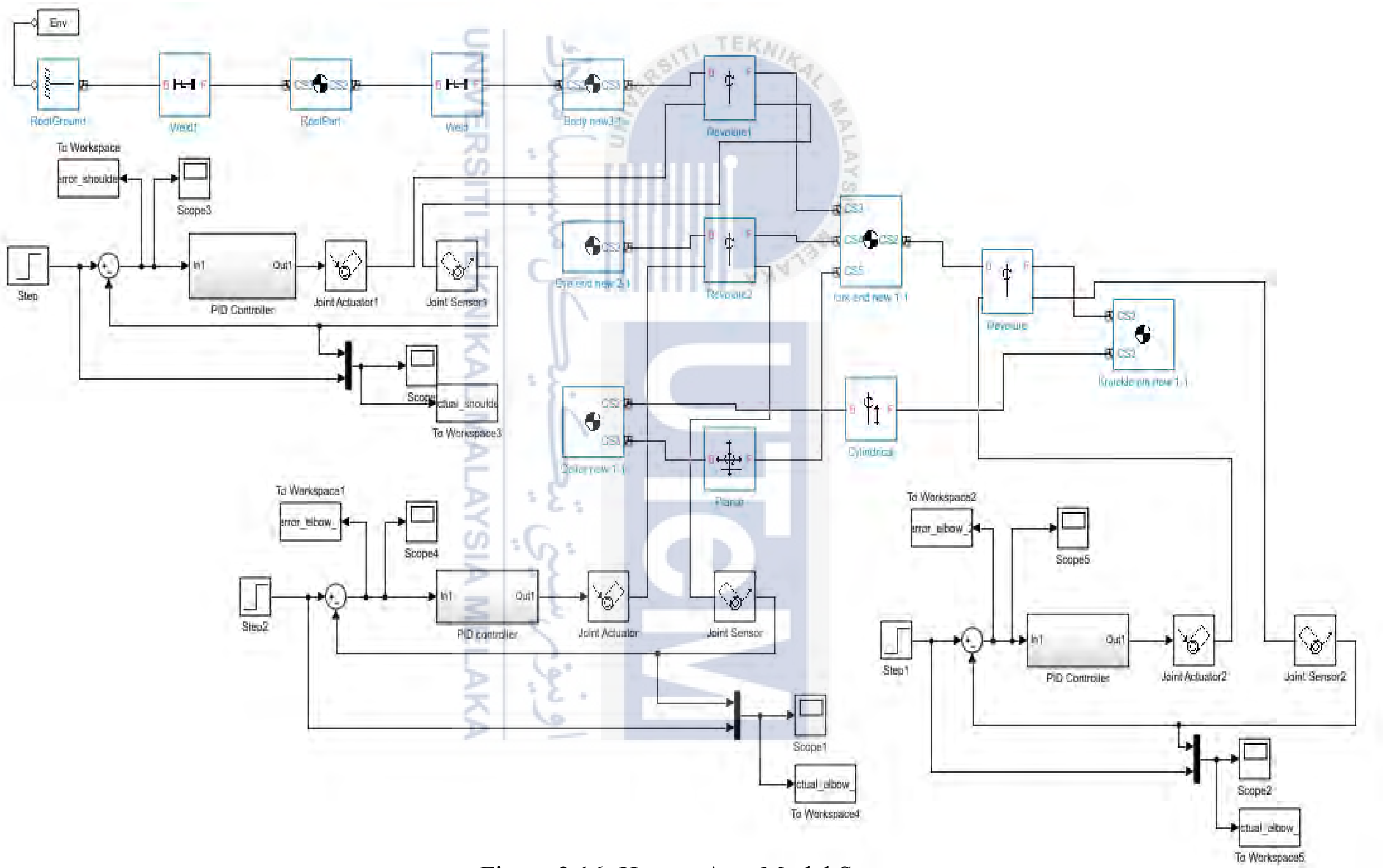


Figure 3.16: Human Arm Model System

3.4 Design and Develop Controller of Human Arm Model

Controller are widely used in industries. Basic controller can be divided into two which is feedback controller and feed-forward controller. In this project, the basic controller used is feedback controller. Flow chart shown in figure 3.17 shows the path of the controller design during the simulation.

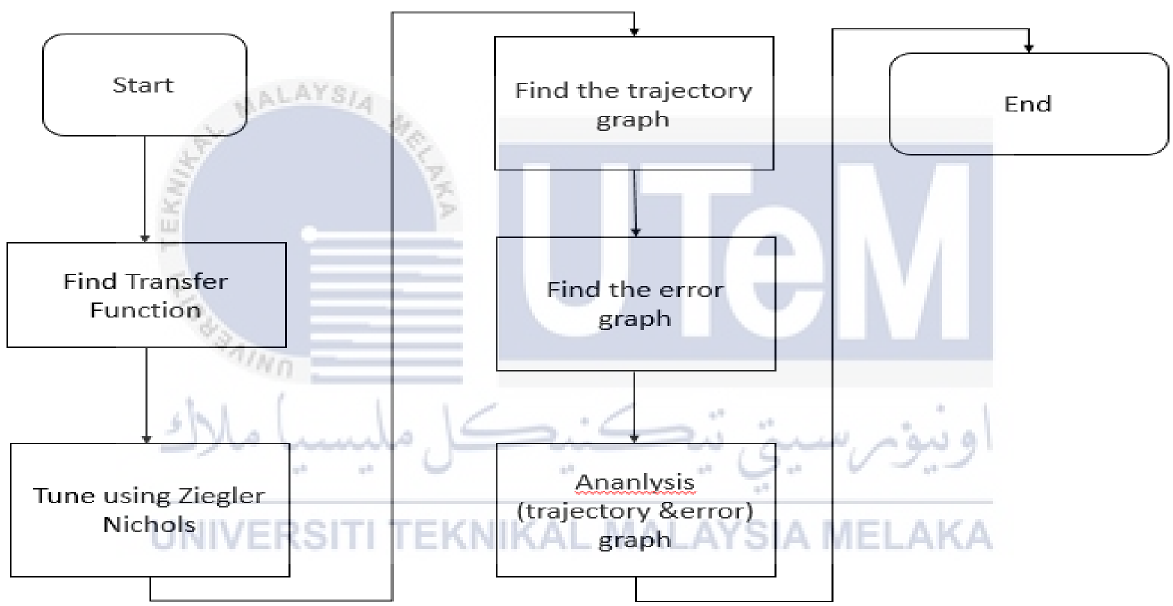


Figure 3.17: Steps to design the controller

The first step to design a controller is by determining the transfer function. The second step after completed finding transfer function is tune the unstable controller. In this project PID and PD controller are tune using Ziegler-Nichols tuning method. The third step is find the trajectory graph. The fourth step is finding error graph. The last step is analyze the response of trajectory and error on the graph. From this, a controller performance are validate whether it is suitable for the joints or otherwise.

3.4.1 Obtaining Human Arm Model Transfer Function

The first step to design controller is by finding the transfer function in the system. The transfer function of the controller are found by using linear analysis. The system are arranged in open-loop form in order to find the transfer function. The linear point have to determine from the block diagram. The linear points of the human arm system are found from the three section. The first section is in shoulder joint, the other two section are from elbow joint. This project consist of three controllers in order to control the human arm movement. Figure 3.18 shows the step tracing the linear points on the shoulder joint.

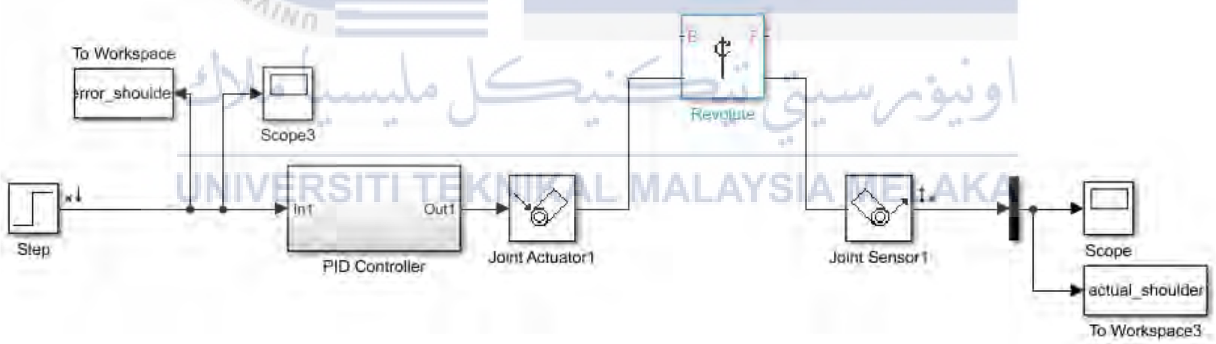


Figure 3.18: Open-loop output linear points on shoulder joint

The further step after finding linear points is finding the transfer function through linear analysis. The transfer function is obtained after linear analysis is done. The equation 3.1 shows the transfer function of the shoulder joint.

$$T(\text{Shoulder}) = \frac{85.49s^2+1563}{s^4+41.05s^2+339.9} \tag{3.1}$$

After finding the transfer function in each joint, the system is redesigned back to the closed-loop system each joint. Figure 3.19 shows the closed loop system design.

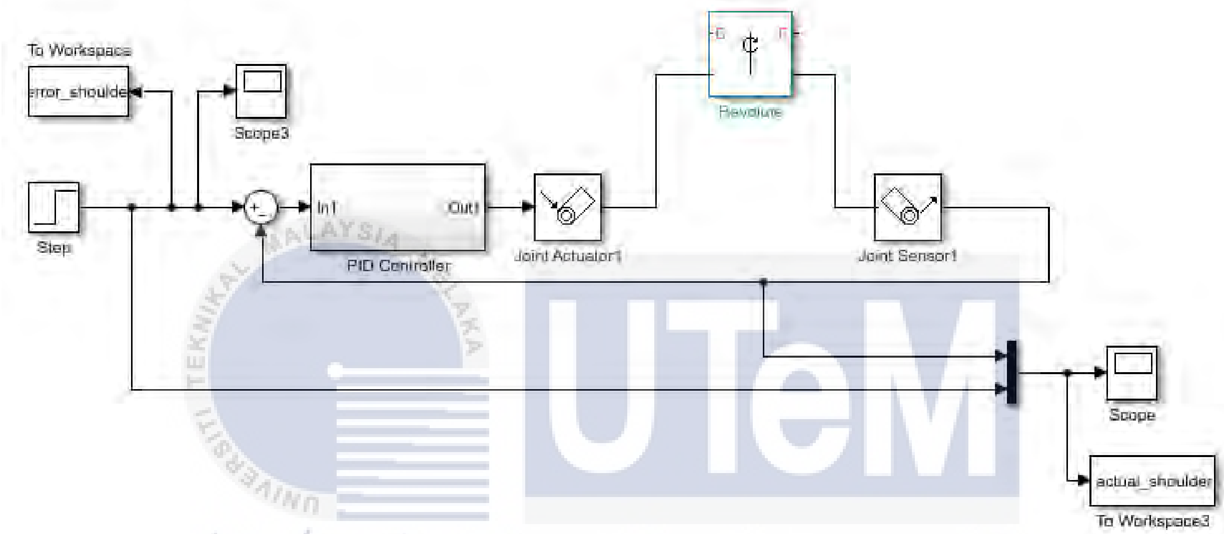


Figure 3.19: Closed-loop system on shoulder joint

3.4.2 Ziegler-Nichols Tuning Techniques

The suitable methods to make the systems response to be stable is by tuning the controller. Previously, the system is designed using open-loop system in order to find the system transfer function. Next, the system is redesign back to closed loop system. The reason is the controller of the system are design based on the feedback controller. The system is set to desired value. The system is run and the response is analyzed. The arm are moves freely and moves in uncontrollable condition. To solve the problem, tuning techniques are required in order to make the system follow the set point. The best tuning methods for this project are by using Ziegler-Nichols tuning techniques. There are two methods in determining Ziegler-Nichols Tuning Techniques:

a. Open-Loop Method

This method avoids the forced oscillations that are found in the continuous cycle tuning method. The Cycling should be avoided if the process is hazardous or critical. This method uses step changes and the rate at which the process reacts is recorded. The graph produces three different values used in mathematical calculations to determine the proper controller settings. To use the Ziegler-Nichols open-loop tuning method, the following steps should be follow:

i. Make an open loop step test

ii. From the process reaction curve determine the transportation dead time,τ_{dead} and the time constant, τ. Figure 3.20 shows an example of the reaction curve in order to determine the τ_{dead} and τ.

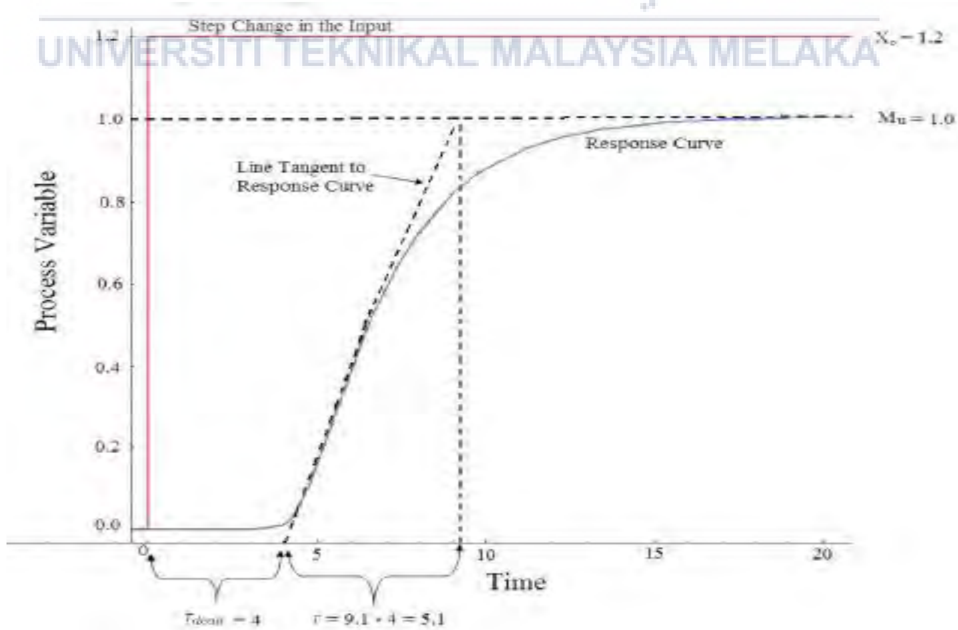


Figure 3.20: Reaction curve

iii. Determine ultimate value that the response reaches at steady-state, M_u, for a step change of X_0.

iv. Calculate the K_0 constant from the dead time and steady-state. The value of K_0 can determine from the formula below.

$$K_0 = \frac{X_0}{M_u} \frac{\tau}{\tau_{dead}} \tag{3.2}$$

where: K_0 = Constant

X_0 = Step change

M_u = Steady − state

τ = Time constant

τ_{dead} = Dead time

v. Determine the loop tuning constants. Plug in the reaction rate and lag time values to the. Ziegler-Nichols open-loop tuning equations for the appropriate controller P, PI, or PID in order to calculate the controller constants. Refer table 3.2 to determine the suitable equations for the equations.

Table 3.2: Parameter of Open-Loop Ziegler Nichols Tuning method [29]

	K_C	T_i	T_d
P	K_0		
PI	$0.9\ K_0$	$3.3\ \tau_{dead}$	
PID	$1.2\ K_0$	$2\ \tau_{dead}$	**$0.5\ \tau_{dead}$**

b. Closed-Loop Method

The continuous cycling method analyzes the process by forcing the controlled variable to oscillate in continuous cycles. The time duration of one cycle is called an ultimate period. The proportional setting that causes the cycling is called the ultimate proportional value. These two values are then used in mathematical formulas to calculate the controller settings.

The tuning method are shown in the steps below:

i. Bring the process to (or as close to as possible) the specified operating point of the control system to ensure that the controller during the tuning is “feeling” representative process dynamic and to minimize the chance that variables during the tuning reach limits.

ii. Turn the PID controller into a P controller by setting set $K_i = 0$ and $K_d = 0$. Initially set gain $K_p = 0$. Close the control loop by setting the controller in automatic mode.

iii. Increase K_p until there are sustained oscillations in the signals in the control system.

iv. Measure the oscillation frequency at the stability limit, T_u .

v. Calculate the controller parameter values according to table 3.3. Use these parameter values in the controller.

Table 3.3: Parameter of Closed-Loop Ziegler-Nichols Tuning method [29]

Controller Type	K_p	K_i	K_d
P	$0.5\ K_u$		
PI	$0.45\ K_u$	$2\frac{K_p}{T_u}$	
PD	$0.8\ K_u$		$\frac{K_P T_u}{8}$
PID	$0.6\ K_u$	$2\frac{K_p}{T_u}$	$\frac{K_P T_u}{8}$

The Ziegler-Nichols closed loop tuning method are chosen to control the trajectory of human arm. This is because the initial response of human arm shows the graph are continuously oscillate in the period. PID controller and PD controller are chosen for the simulation of human arm.

The controller design of human arm are proceed after having the transfer function and tuning techniques. In this project, PD controller and PID controller are selected to control the human arm movement. The control design of human arm occurs at shoulder joint and elbow joint. One controller are located at shoulder joint. Two other controllers are located at elbow joint. The joint at elbow are named as elbow. In this project, the controller design are studied based on trajectory and error. Each controller design at each joint have trajectory and error. The input of the trajectory are tabulated in table 3.4.

Table 3.4: Trajectory of human arm joint

Joints	Trajectory
Shoulder Joint	60^0
Elbow Joint	30^0

The simulation time are set to 7 seconds. The system is test in order to determine the initial response of the model. Figure 3.21 shows the initial response of the human arm system at the shoulder joint.

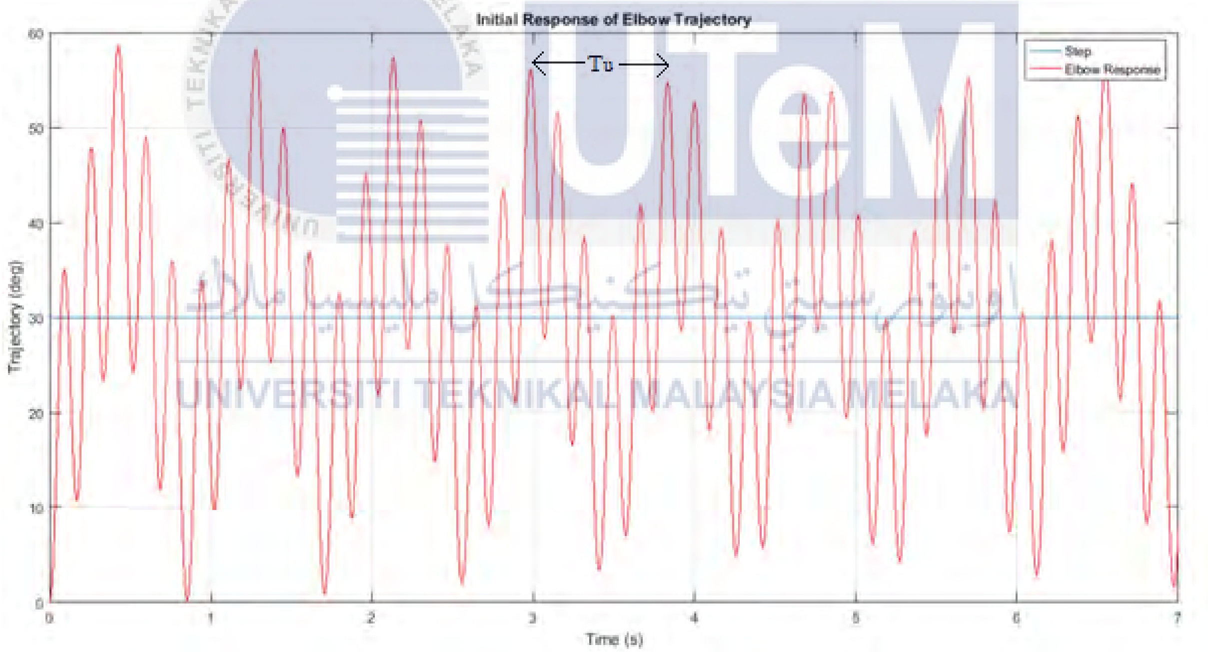


Figure 3.21: Initial response of elbow joint

Based on the figure 3.21, the human arm model are oscillates along the period. In order to stabilize the response, a tuning technique are used. Ziegler- Nichols (Z-N) tuning technique is used in this project in order to make the response of the system to follow the input. The (Z-

N) closed loop method is suitable to tune the response of human arm. The initial gain of the controller is set as $K_u = 1$. Next step is finding the oscillation frequency, T_u. T_u is determine from peak to peak oscillation period. Insert the value on the tabulated formula of PD controller and PID controller. The tuning parameters are shown in figure 3.5.

Table 3.5: Tuning parameters of PID controller and PD controller [26].

Type of Controller	K_p	K_i	K_d
PD	$0.8K_u$		$\frac{K_P T_u}{8}$
PID	$0.6K_u$	$2\frac{K_P}{T_u}$	$\frac{K_P T_u}{8}$

After obtaining the needed value, insert the value to the controller. Next, re-run the simulation. Figure 3.22 shows the new response of the model system in PID controller.

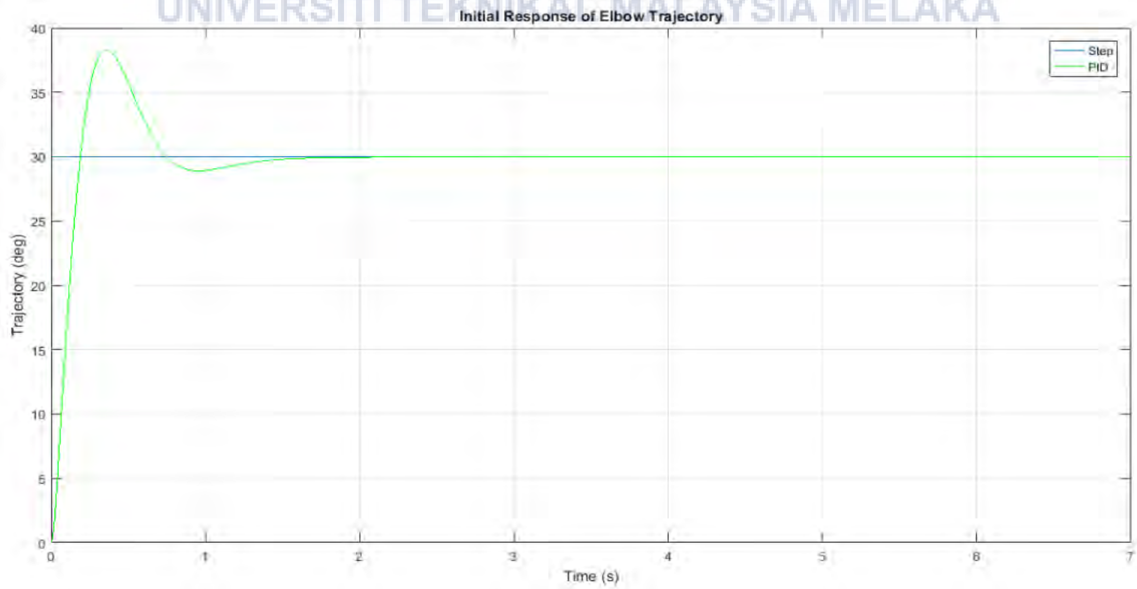


Figure 3.22: An example response of PID controller on elbow joint for $K_u = 1$

Next, the simulation is re-run for PD controller. Figure 3.23 shows the response of PD controller.

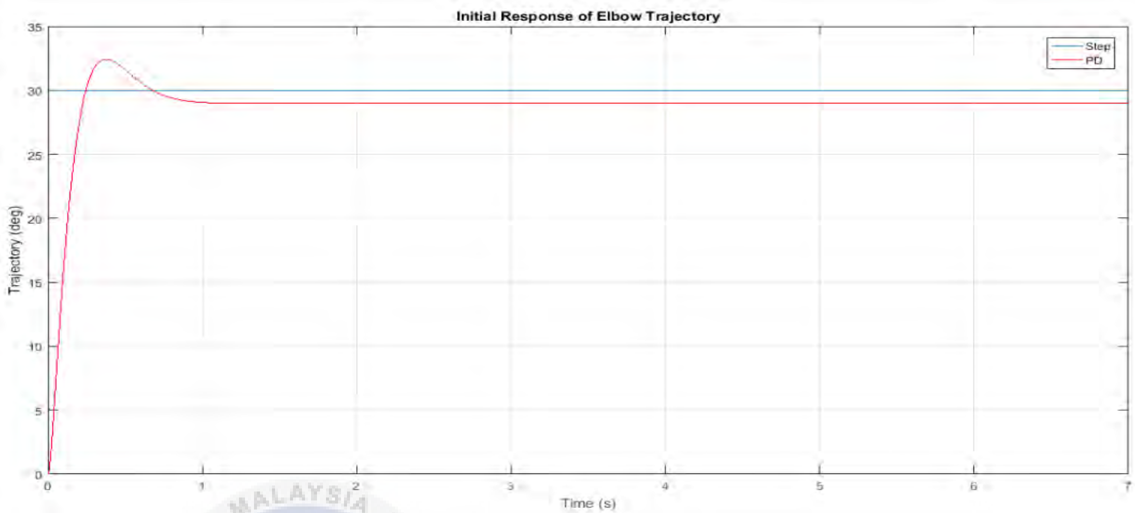


Figure 3.23: Response of PD controller on elbow for $K_u = 1$

In order to simplify the analysis of the response, the graph of PID controller and PD controller are combined. Figure 3.24 shows the combination of PID controller and PD controller in one graph.

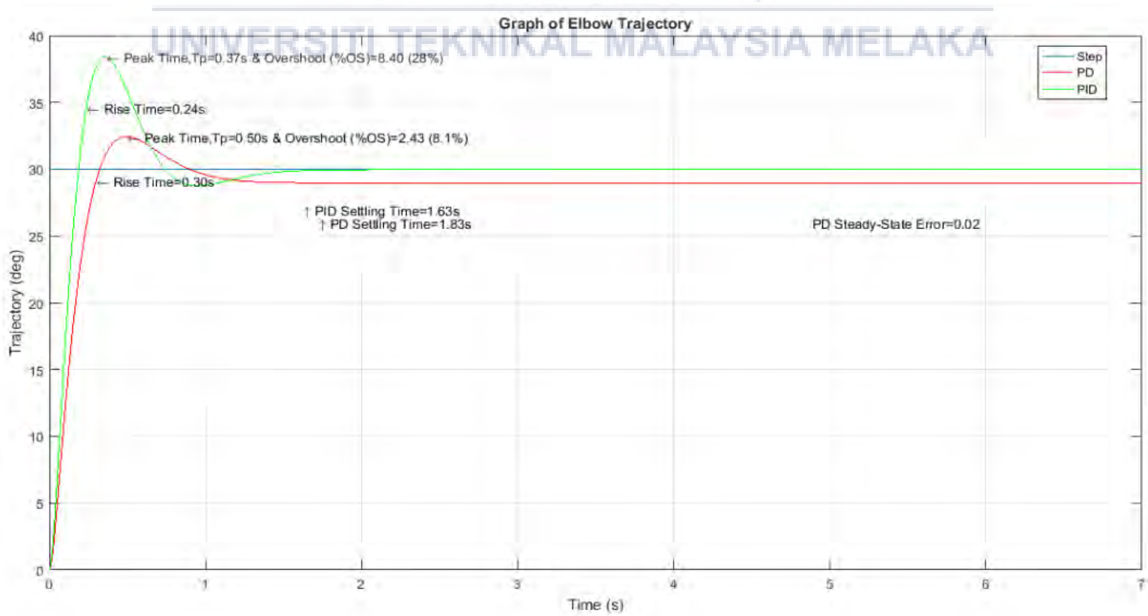


Figure 3.24: An example of elbow trajectory of both controllers for $K_u = 1$

Each response has its error. The error are measured on the trajectory. The set point in the graph error are defined as reference. The reference value are 0. If the transient behavior doesn't attach the reference, the response have the error. The range of error are measured by subtract the maximum error (e_{max}) and minimum error (e_{min}). The mathematical formula to calculating error:

$$\text{Range} = (e_{max}) - (e_{min}) \tag{3.2}$$

where :

e_{max} = Maximum error produced by system

e_{min} = Minimum error produced by system

Figure 3.25 shows the response of the error of PID controller in the model system in PID controller.

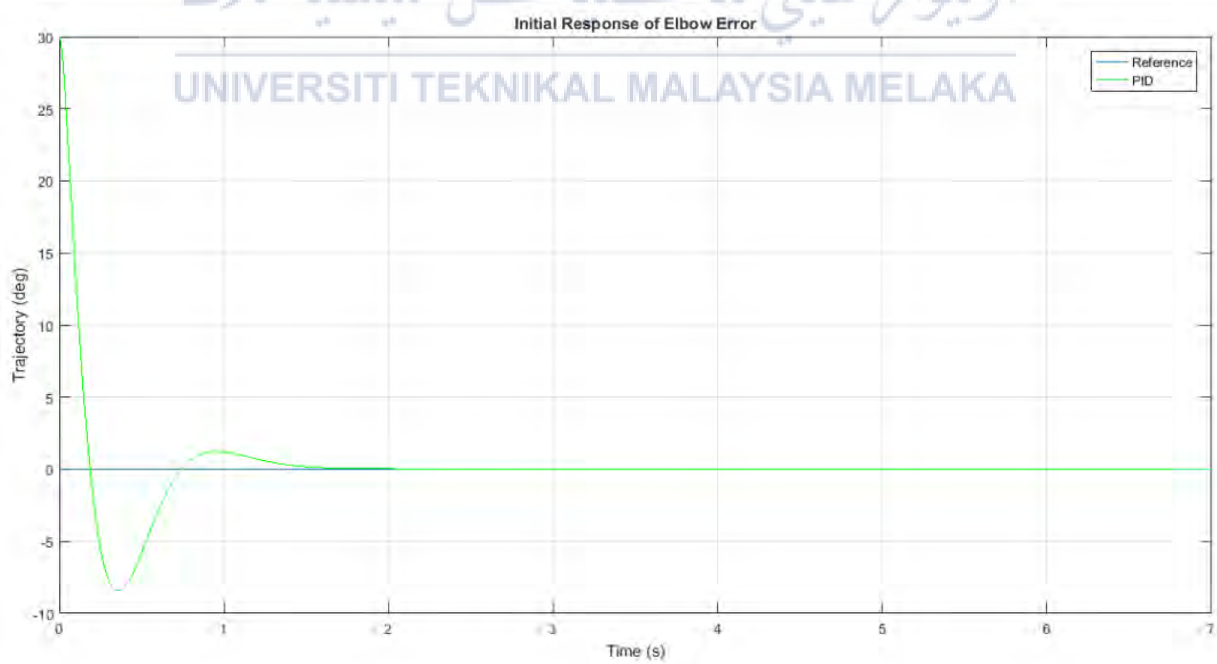


Figure 3.25: Response error of PID controller on elbow for $K_u = 1$

Figure 3.26 shows the response of the error of PID controller in the model system in PD controller.

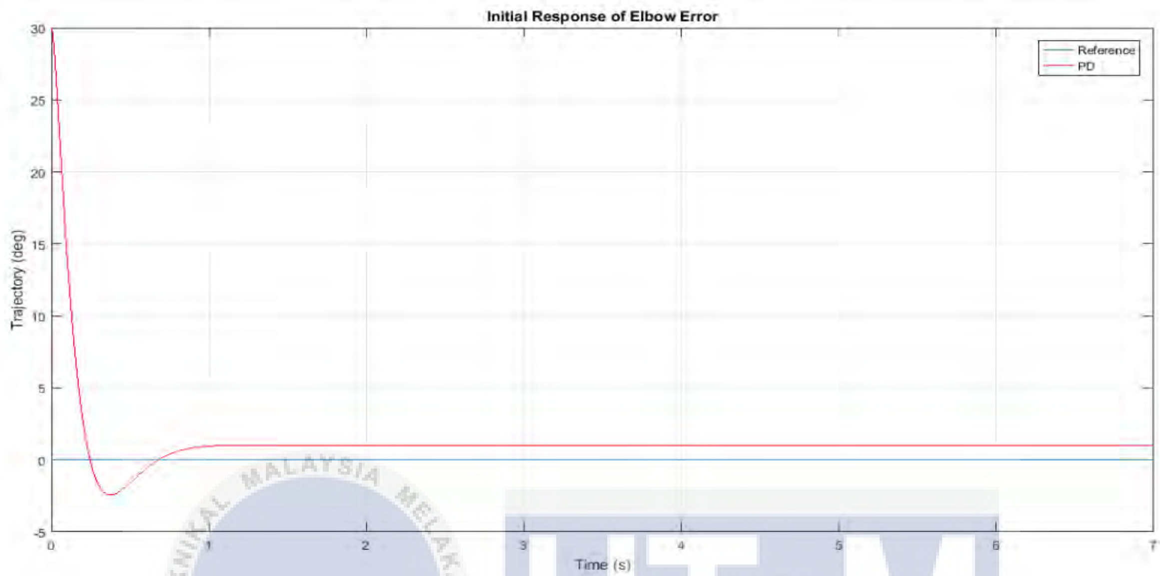


Figure 3.26: Response error of PD controller on elbow 1 for $K_u = 1$

Finally, after typing the data set coding and text description coding, the graph are presented in figure 3.27.

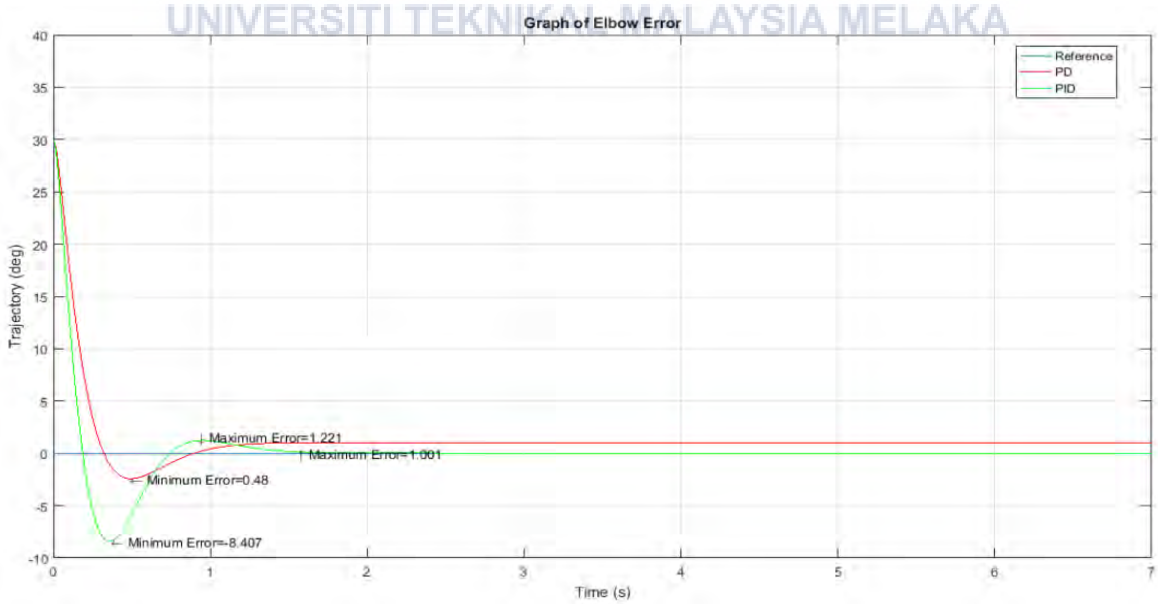


Figure 3.27: An example of elbow error for $K_u = 1$

After collecting all data required from the graph, the performance and error of the joints are analyzed. The performance of the joints are analyses based on the transient and steady-state responses. The error of the joints are analyses based on the range of error. The best controller performance (transient & steady-state responses) and minimum range of error will be chosen for the suitability of the joints. The analysis part are discuss in the chapter 4.

3.5 Summary

The methodology of this project begin by modelling of human arm. The model then is design in Solidworks and transfer to MATLAB. This section called as software development. The model is test by adding some of the blocks to the block diagram. The controller is attach to the elbow joint and test by using the PID and PD controller. The PID and PD controller value is tune on the plant model by using the Ziegler-Nichols tuning method. The initial gain of $K_u = 1$ and $K_u = 2$ are test on the system. The value of K_p, K_i and K_d are obtained by using the parameters of Ziegler-Nichols formula. The tuned response is run through the PID controller and PD controller. The system transient performance which is trajectory and error performance are analyzed. The best system will be selected either it is suitable to use in the shoulder joint and elbow joint. The valid controller will be used in the suitable joints.

CHAPTER 4

RESULT AND DISCUSSION

This chapter describes about the result and discussion of the project. Result is created to show the outcomes of the project after referring to the methodology. In this Final Year Project 2 (FYP 2), all result are displayed in this chapter. The result shown is the modelling of human arm in Solidworks. Transferring human arm model to MATLAB, and the controller design of human arm.

4.1 Human Arm Model Design

The human arm model is designed by using the Solidworks software. Six component parts was created. They are

1. Body part
2. Box part (combined with fork end)
3. Fork end
4. Eye end
5. Knuckle pin
6. Collar.

All of this part was created in separate file. Figure 4.1 shows the model part are shown in the exploded view.

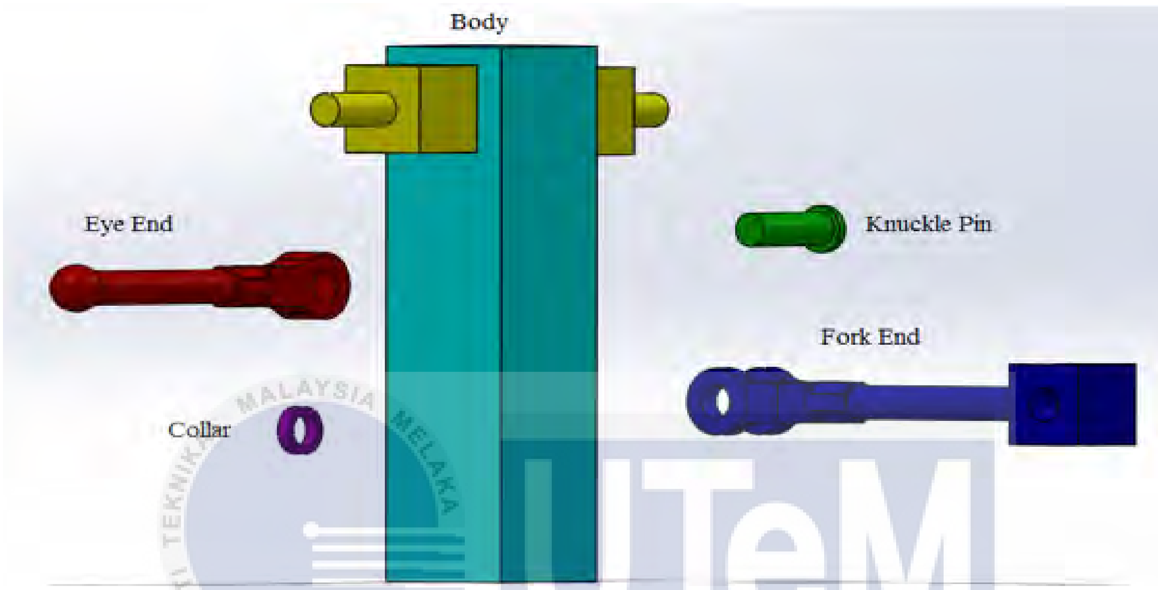


Figure 4.1: Exploded view of human arm model parts

The model's part are assembled in one file. The assembly of human arm occurs on the two joints. The first joint occurs in the shoulder joint. In shoulder joint, two parts was assembly which is body part and box part. The second joint occurs in the elbow joint. The assembly in elbow joint consist of the combination fork end, eye end knuckle pin and collar. The assembly of the model shows two degree of freedom (2-DOF) arm model was created in this project. The 1-DOF of human arm occurs at the shoulder joint with the extension/flexion of $-180^0/+80^0$.The another 1-DOF occurs at the elbow joint with the extension/flexion of $-10^0/+145^0$.

The model created using the mating techniques. The mating techniques includes coincidence and concentric of the model part. The model are created into two stages. The first stage is the modelling of one arm. The second stage is the modelling of two arm. Figure 4.2 shows the first stage modelling which one arm model and second stage model which is two arm model.

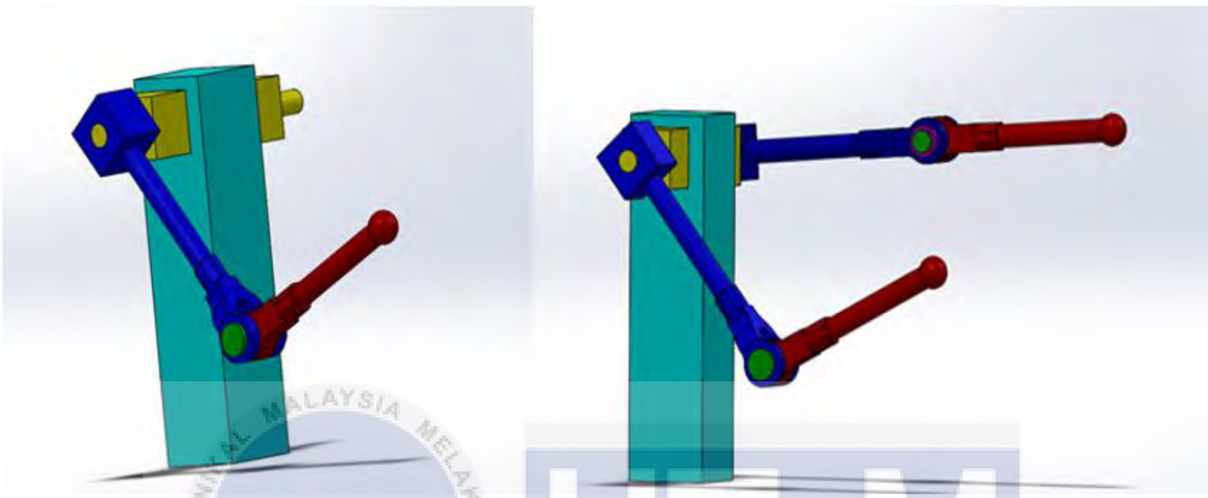


Figure 4.2: Human arm model; a) One arm model (stage one model)
b) Two arm model (stage two model)

The completed design of the two stages of human arm model is exported out from the Solidworks. Both model is saved in xml file as it exported out from the Solidworks software. The file xml is created by helping from the Simscape multibody / Simmechanics link. This plug-in helps the file in Solidworks to save in the xml files.

4.2 Importing Human Arm Model via Simmechanics

The model are imported to the MATLAB software from Solidworks software. The purpose of importing the file to MATLAB is to do the simulation and do the analysis on the model. The beginning step to import the model are by type a code 'mech_import ('*name of file.xml*') on the command widow, the modelling is transferred into the MATLAB and block diagram is created in the MATLAB Simulink. The stage one model (one arm model) and stage two model (two arm model) are imported to the MATLAB and appears as the block diagram in MATLAB Simulink.

Figure 4.3 shows the imported of human arm model (one arm model) represented as the block diagram MATLAB Simulink.

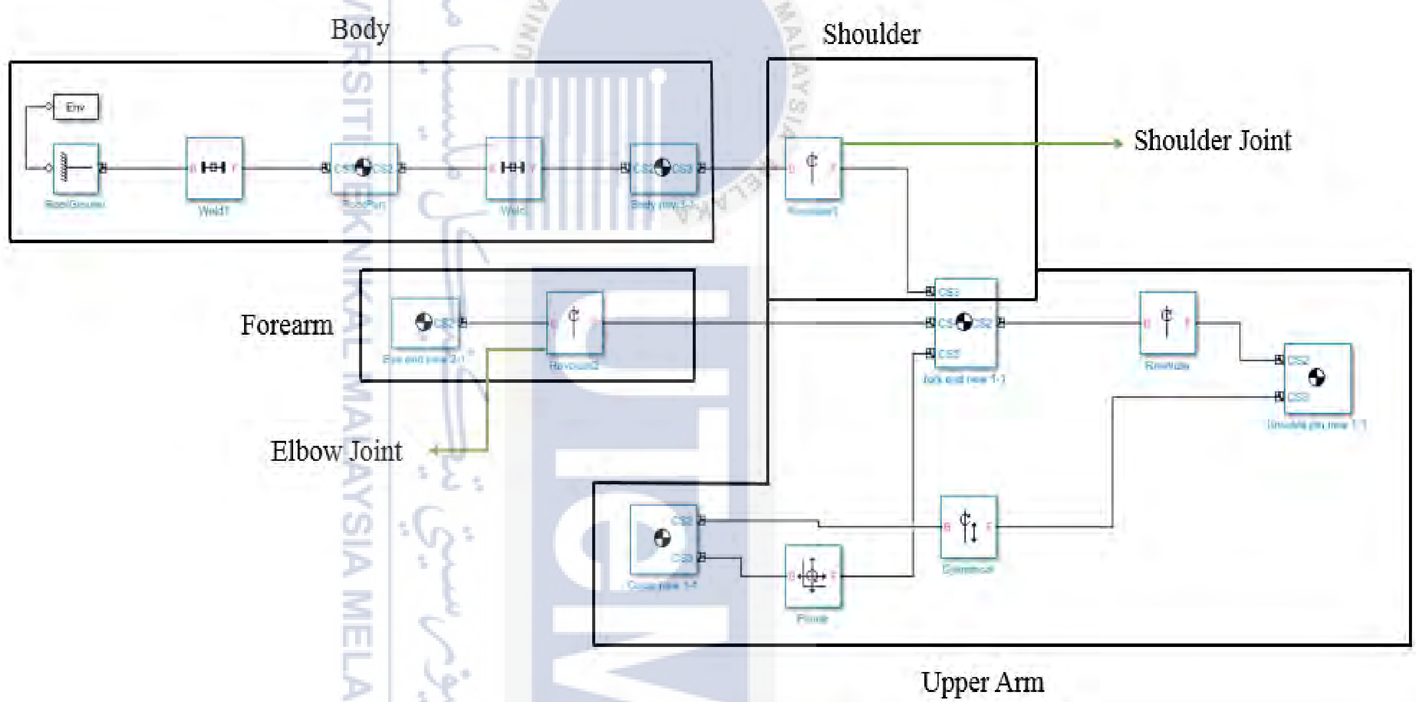


Figure 4.3: Imported of one arm model

Figure 4.4 shows the imported of human arm model (two arm model) represented as Simulink Output of MATLAB Simulink.

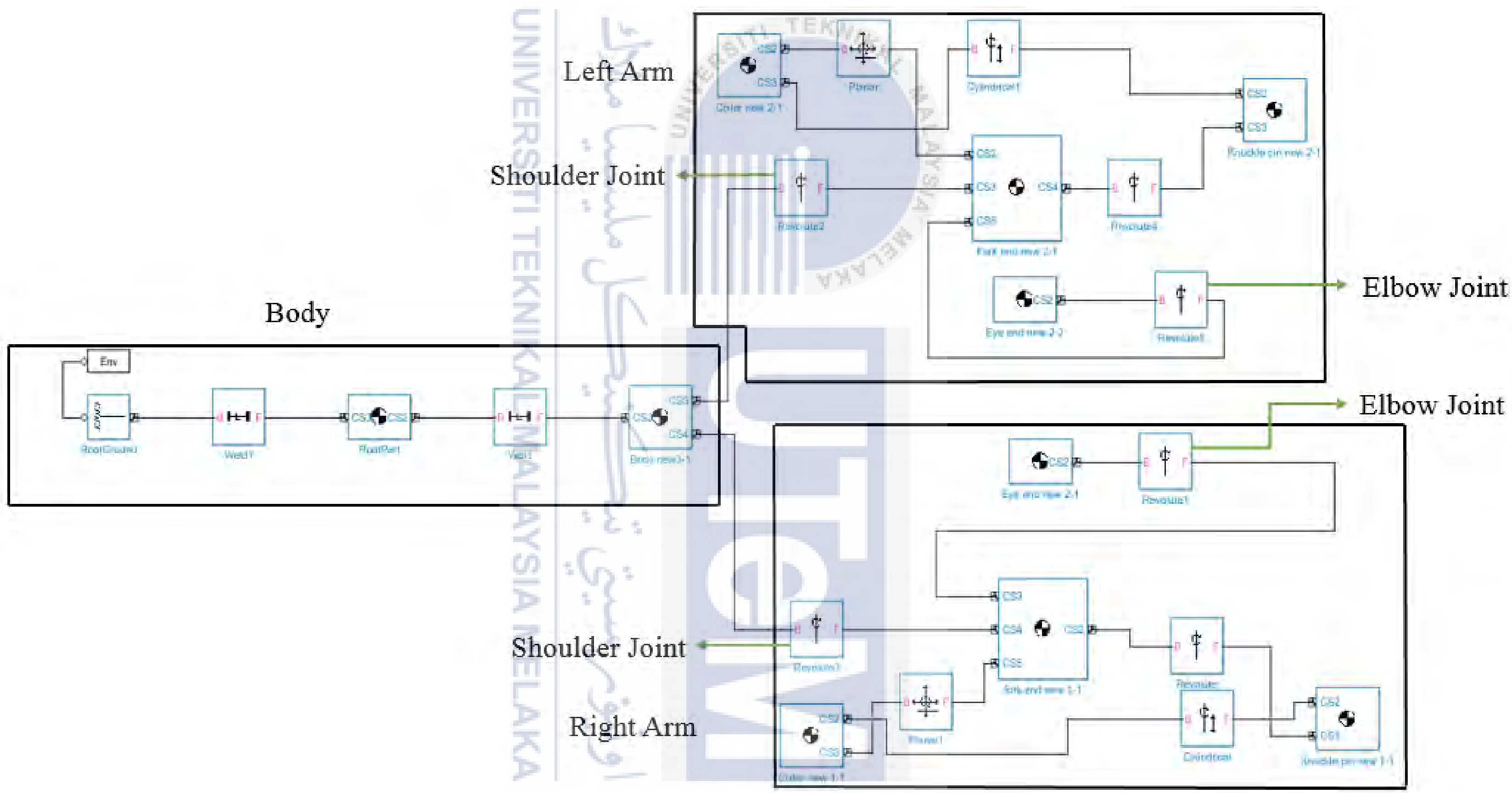


Figure 4.4: Imported two arm model

The first stage (one arm model) was selected in this project. This is because one arm model is simpler after it is being transferred to Simulink Output. The main objectives of this project is to design a controller that able to control the human arm. The controller design is focuses on the human arm movement by attaches the controller on the shoulder joint and elbow joint. The additional blocks are added and attached to Simulink Output .The added block are added and attached with controller in order to make the human arm model to become a control system. The controller are consist of Proportional-Derivative (PD) controller Proportional-Integral-Derivative (PID) controller. Both of the controller are designed in separated blocks and put in the subsystem. Figure 4.5 and figure 4.6 shows the configuration of arm model Simulink output and additional blocks which is consist of PID controller and PD controller in MATLAB Simulink.

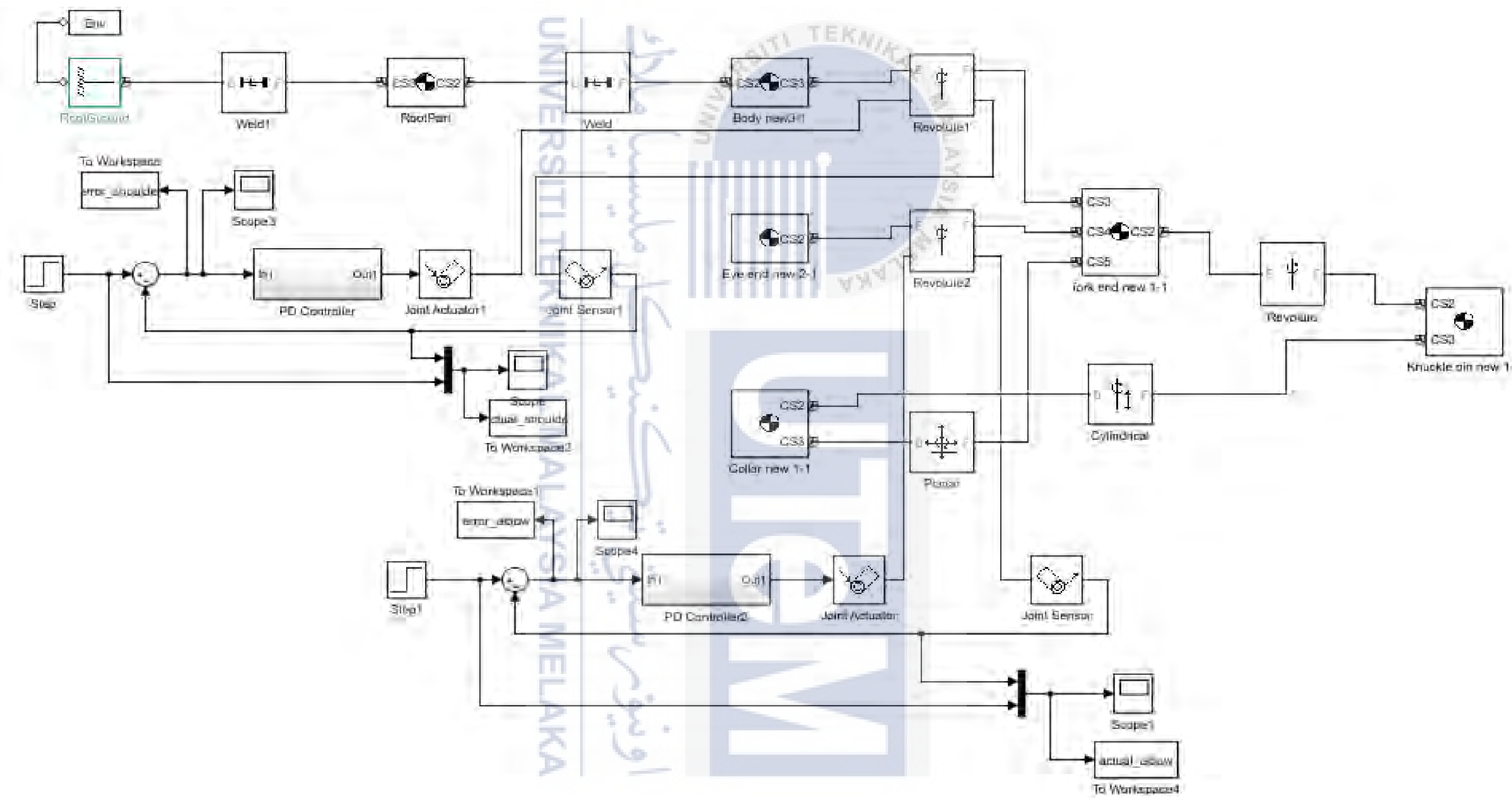


Figure 4.5: Arm model Simulink output an additional blocks

(PD Controller)

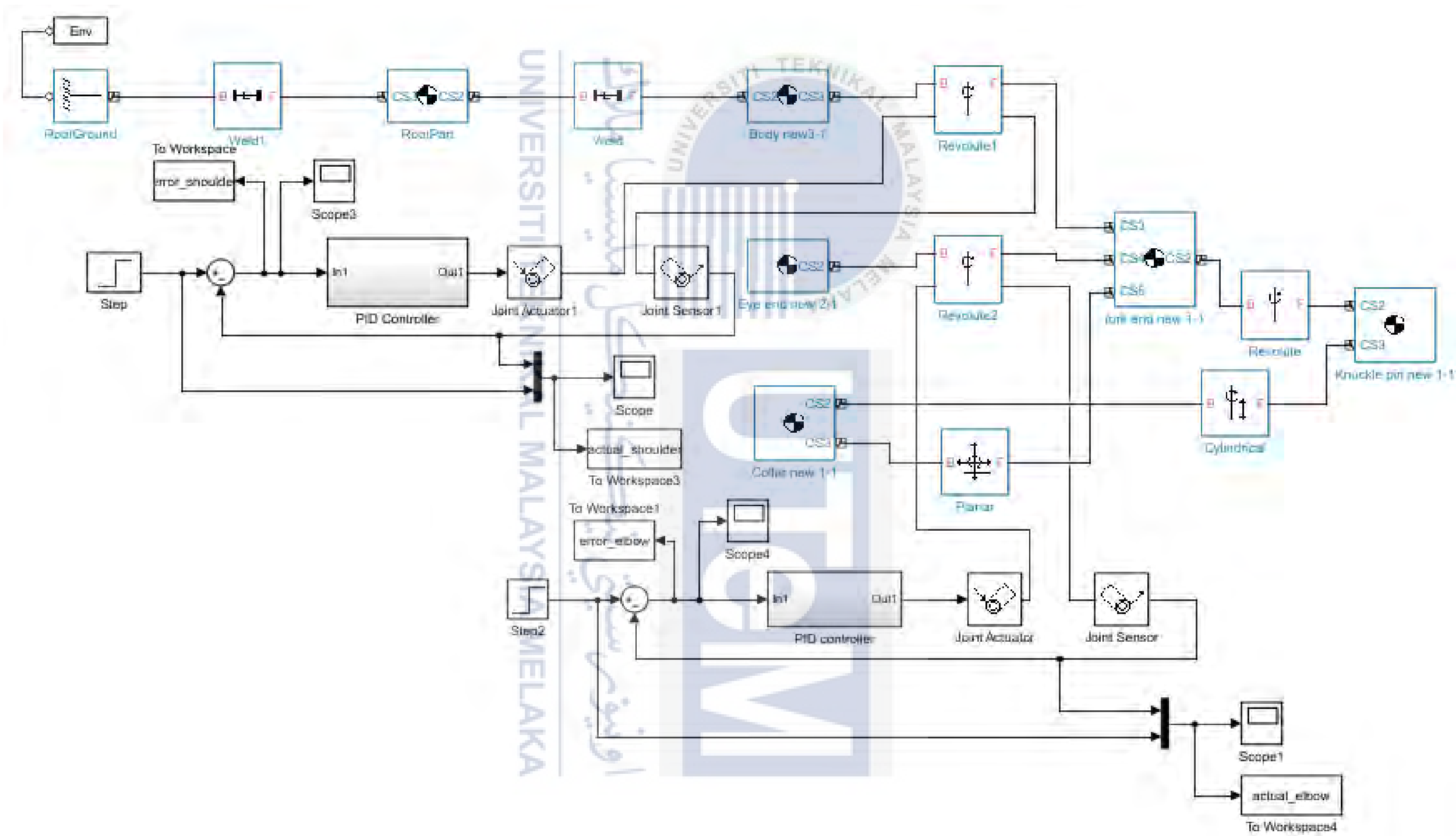
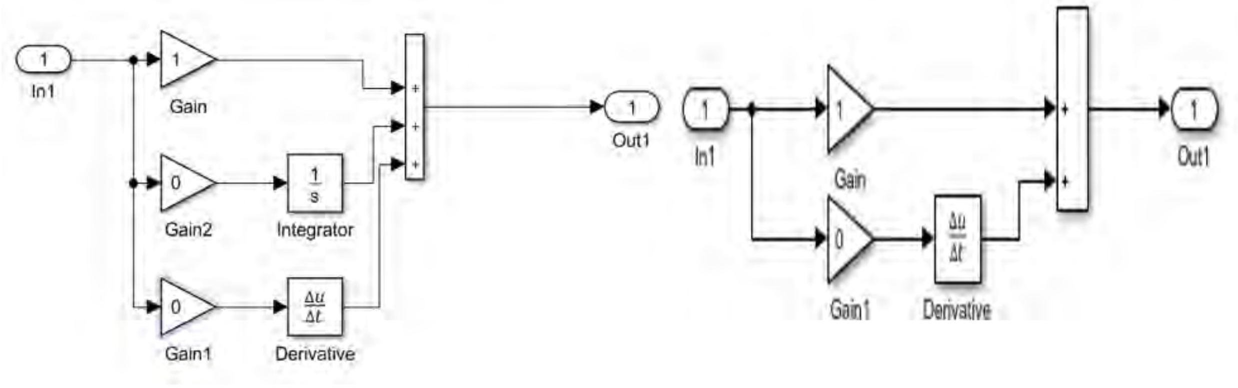


Figure 4.6: Arm model Simulink output an additional blocks

(PID Controller)

Figure 4.7 shows PID and PD controller configuration in the subsystem block.



a b

Figure 4.7: Controller in Subsystem; a) PID controller b) PD controller

Based on the figure 4.7, the additional blocks which is consist of controller (PD and PID controller) has attached to two joints on the human arm model. Three controller has attached on the human arm. One controller attached on the shoulder joint. The rest of the two controller attached at the elbow joint.

4.3 Develop and Design Control System of Human Arm Model

The first stage of human arm (one arm) was selected for the development and design of controller. The selecting of the controller have been done on the sub chapter of 4.2 (*Importing the Human Arm Model via Simmechanics*). The first step is finding the transfer function. The system was create in open-loop system in order to find the transfer function. The transfer function are determined by using the linear analysis on the system. The transfer function are determine at shoulder joint and elbow joint. The transfer function as shown in equation 4.1 and 4.2 :

a. Shoulder Joint :

$$\text{T (Shoulder)} = \frac{85.49\, s^2 + 1563}{s^4 + 41.05 s^2 + 339.9} \quad (4.1)$$

b. Elbow Joint :

$$T\text{ (Elbow)} = \frac{1029\, s^2 + 62.4}{s^4 + 41.05 s^2 + 339.9} \tag{4.2}$$

The system is redesigned back to the closed-loop system after obtaining the transfer function in each joint. The human arm model is run without the controller value. Figure 4.8 shows the initial response of shoulder joint and elbow joint controller.

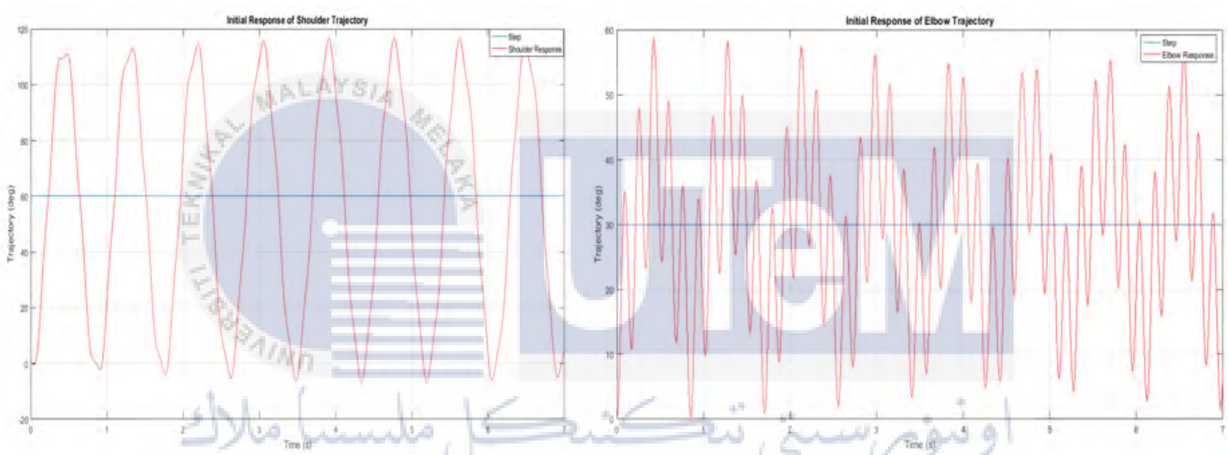


Figure 4.8: Initial response of the joints; a) Shoulder joint b) Elbow joint

The best controller to stabilize the above response are by using PID controller. PID controller can be divided into two parts which are PI controller and PD controller.

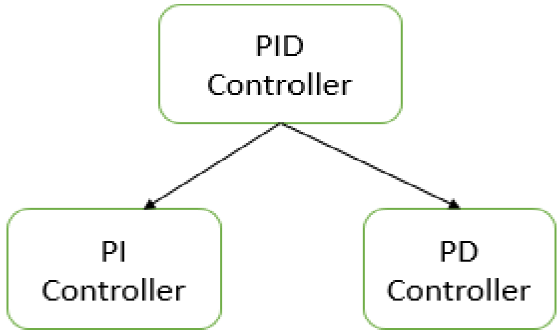


Figure 4.9: PID controller division

In this project, two controller has been designed. They are PID controller and PD controller. The movement of the selected human arm model are rate based on PID controller and PD controller. An analysis of human arm model are discuss at the end of this chapter.

A tuning method was used in this project. Ziegler-Nichols is the suitable tuning method for PID controller and PD controller in this project. Table 4.1 shows the formula of the PID controller and PD controller that used along the simulation of the project.

Table 4.1: Parameter of Ziegler-Nichols (Z-N) Tuning Method [29]

Type of Controller	K_p	K_i	K_d
PD	$0.8K_u$		$\frac{K_P T_u}{8}$
PID	$0.6K_u$	$2\frac{K_P}{T_u}$	$\frac{K_P T_u}{8}$

Initial value of the proportional gain was set on shoulder joint controller and elbow joint controller. The initial gain has determined as K_u .The value of K_u are 1 and 2. The simulation begins with the value of $K_u = 1$. The analysis of the graph are done on the trajectory graph and the error graph. The trajectory are analyzed based on the transient behavior. The transient behavior consist of rise time,T_r, Overshoot, OS, Settling Time,T_s and Peak Time , T_P .Next, this trajectory also analyzed based on the steady-state error, e_{ss} between the controllers. The error graph are analyzed based on the error. The range of error are analyzed on this subtopics.

4.3.1 The performance of controller on human arm model joints

Shoulder Joint for $K_u = 1$

Table 4.2: The (Z-N) tuning value on shoulder joint for $K_u = 1$

Type of Controller	K_p	K_i	K_d
PD	0.8		0.302
PID	0.6	0.397	0.227

In shoulder joint, the value of K_u was set to 1 and sustained oscillation time, T_u is 3.02 second .The trajectory on the elbow joint was set to 60^0. Figure 4.10 shows the movement of the shoulder joint on PID controller and PD controller. The value on the table 4.2 is inserted in both controller.

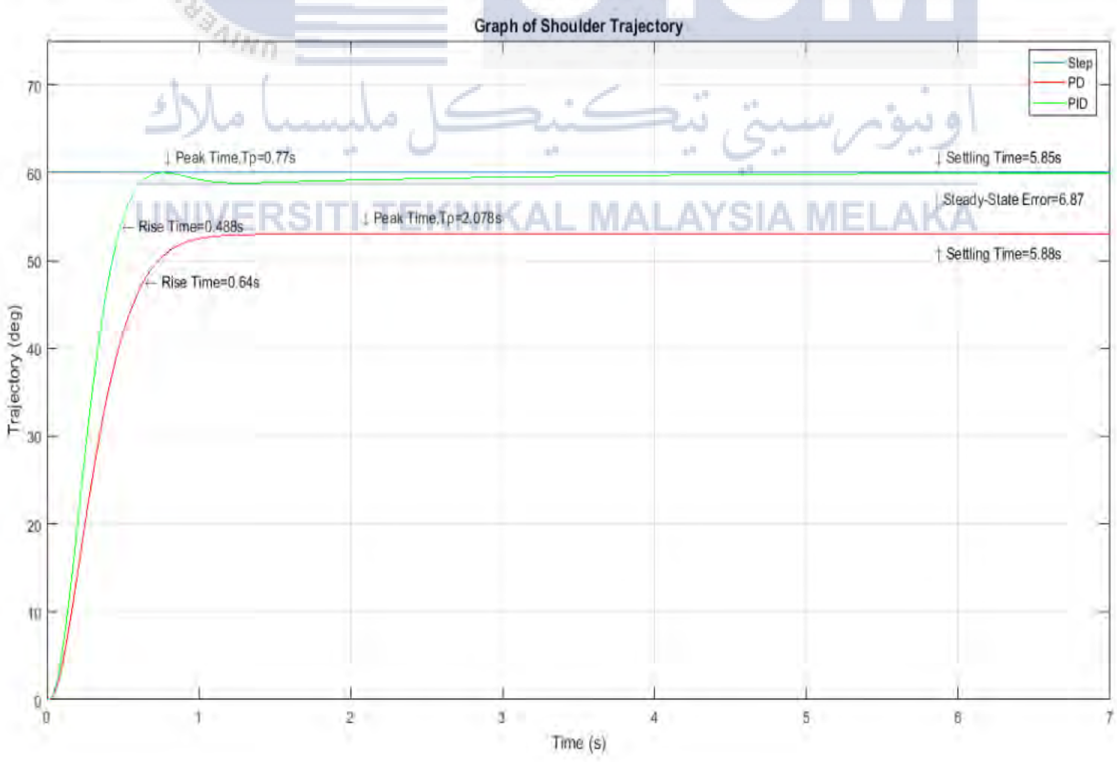


Figure 4.10: The trajectory of shoulder joint for $K_u = 1$

Figure 4.10 shows the transient and steady-state behavior of PID controller and PD controller on the trajectory of shoulder joint in human arm model. In PID controller, the rise time of human arm model trajectory is 0.488 seconds. The peak time of the trajectory is 0.77 seconds and the settling time of human arm model trajectory is 5.854 seconds. The human arm model trajectory does not have overshoot. No steady-state error occurs at human arm model since the model are stable at the set point.

In PD controller, the rise time of the human arm trajectory is 0.64 seconds. The peak time produced is 0.78 seconds. The settling time of the human arm trajectory is 2.078 seconds which the trajectory of human arm model are constant at 56.598 degrees. No overshoot can be analyze for the output performance of model. The human arm model has the steady-state error of 6.87.

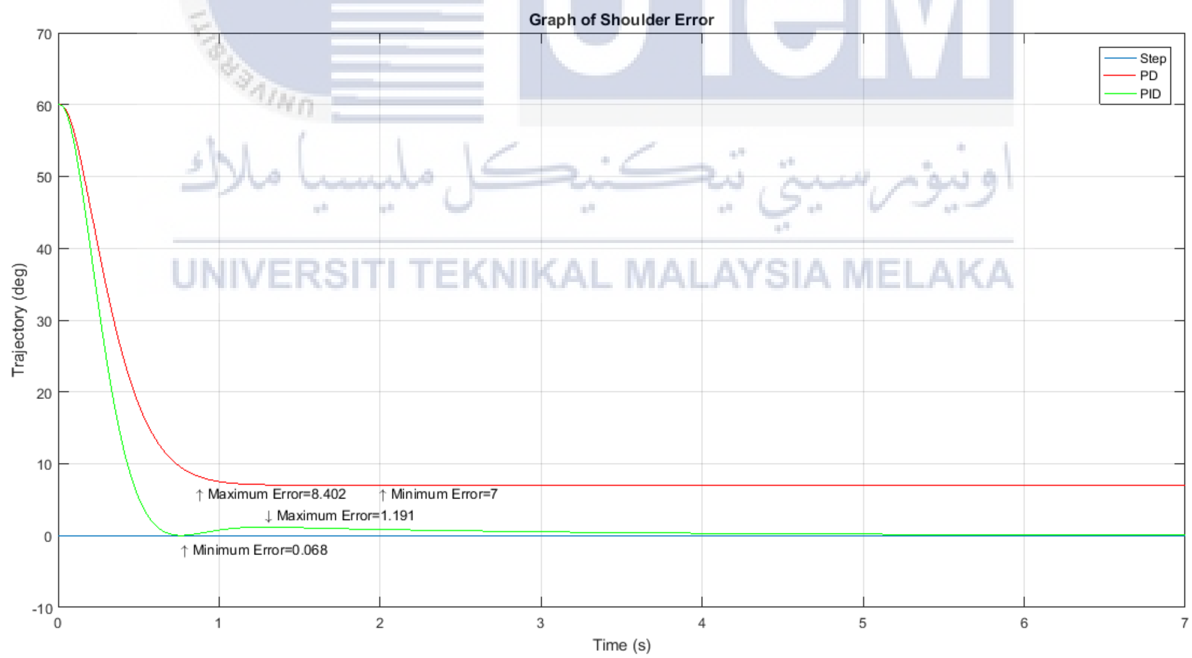


Figure 4.11: Trajectory error of shoulder joint for $K_u = 1$

Figure 4.11 shows the output error of PID controller and PD controller. In PID controller the minimum error is 0.068 degrees. The maximum error is 1.191 degrees. The range of error produced by the controller is 1.123 degrees.

In PD controller, the minimum error produced by system is 7 degrees. The maximum error in the system is 8.402 degrees. The range of error produced by the controller is 1.402 degrees.

Elbow Joint for $K_u = 1$

Table 4.3: The (Z-N) tuning value on elbow joint for $K_u = 1$

Type of Controller	K_p	K_i	K_d
PD	0.8		0.074
PID	0.6	1.613	0.056

In elbow joint, the value of K_u was set to 1.The sustained oscillation time, T_u of elbow joint is 0.744 seconds .The trajectory on the elbow joint was set to 30^0. Figure 4.12 shows the movement of the shoulder joint on PID controller and PD controller. The value on the table 4.3 is inserted in both controller.

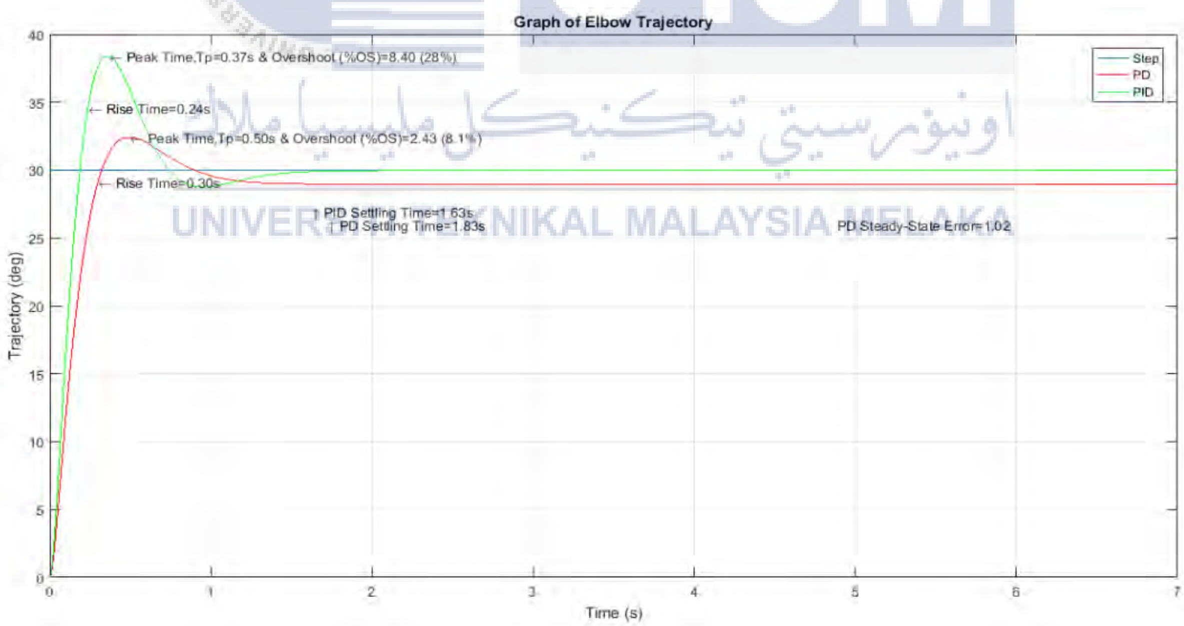


Figure 4.12: Trajectory of elbow joint for $K_u = 1$

Figure 4.12 shows the transient behavior and steady-state behavior of PID controller and PD controller on the trajectory of elbow joint in human arm model. In PID controller the rise time of human arm model trajectory is 0.24 seconds. The peak time of the trajectory is 0.37 seconds and the settling time of human arm model trajectory is 1.63 seconds. The human arm model trajectory have overshoot of 8.40 degrees with the percentage of overshoot (%OS) is 28%. No steady-state error occurs at human arm model since the model are stable at the set point.

In PD controller, the rise time of the human arm trajectory is 0.30 seconds. The peak time produced is 0.50 seconds. The settling time of the human arm trajectory is 1.83 seconds. The overshoot of the human arm model is 2.43 degrees with the percentage of overshoot (%OS) is 8.1%. The human arm model has the steady-state error of 1.02 which the system does not reach set point using PD controller.

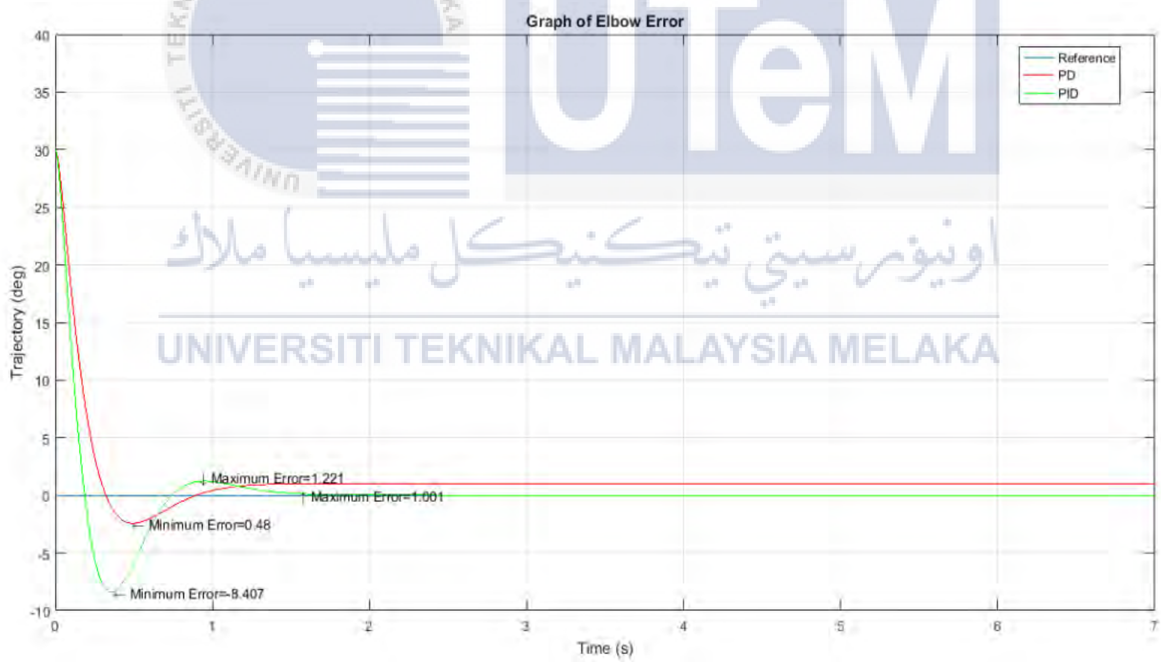


Figure 4.13: Trajectory error of elbow joint for $K_u = 1$

Figure 4.13 shows the output error of PID controller and PD controller. In PID controller the minimum error is -8.407 degrees. The maximum error is 1.221 degrees. The range of error produced by the controller is 9.628 degrees.

In PD controller, the minimum error produced by system is 0.48 degrees. The maximum error in the system is 1.001 degrees. The range of error produced by the controller is 0.521 degrees.

The value of initial gain is continue with the initial gain, K_u = 2. The trajectory joints of the human arm model are analyzed based on the transient behavior and error response.

Shoulder Joint for $K_u = 2$

Table 4.4: The (Z-N) tuning value on shoulder joint for $K_u = 2$

Type of Controller	K_p	K_i	K_d
PD	1.6		0.198
PID	1.2	2.429	0.148

In shoulder joint, the value of K_u was set to 2 and sustained oscillation time, T_u is 0.988 second .The trajectory on the elbow joint was set to 60^0. Figure 4.16 shows the movement of the shoulder joint on PID controller and PD controller. The value on the table 4.4 is inserted in both controller.

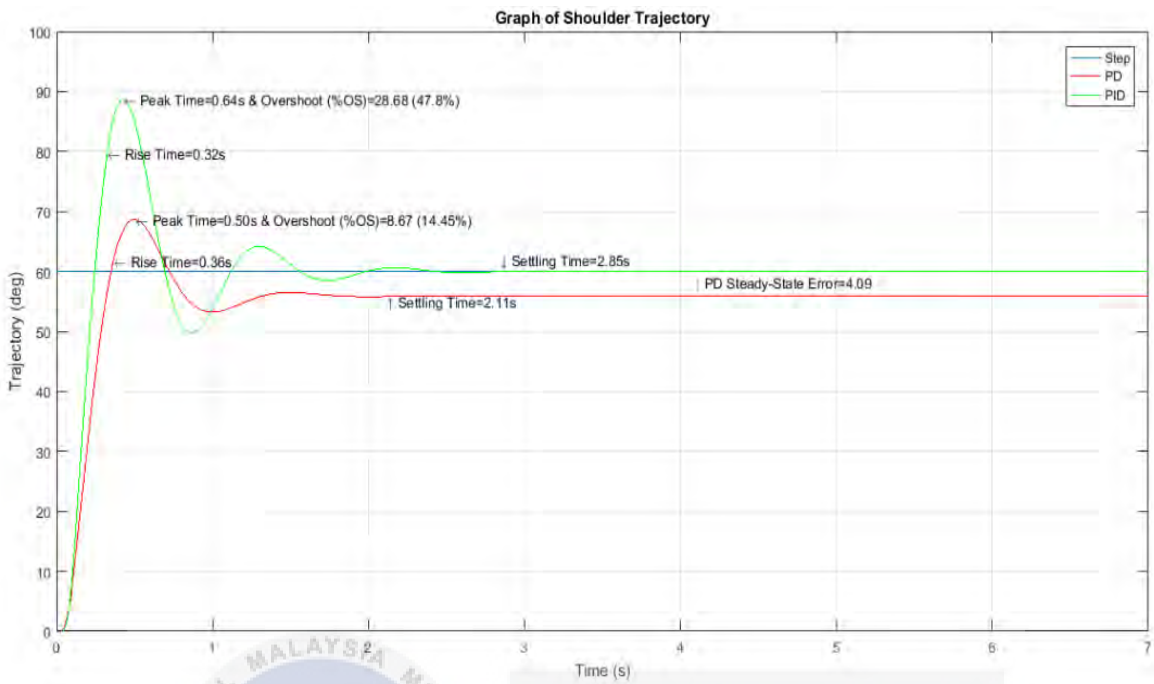


Figure 4.14: Trajectory of shoulder joint for $K_u = 2$

Figure 4.14 shows the transient and steady-state behavior of PID controller and PD controller on the trajectory of shoulder joint in human arm model. In PID controller the rise time of human arm model trajectory is 0.32 seconds. The peak time of the trajectory is 0.64 seconds and the settling time of human arm model trajectory is 2.85 seconds. The human arm model trajectory have overshoot of 28.68 degrees with the percentage of overshoot (%OS) is 47.8%. No steady-state error occurs at human arm model since the model are stable at the set point.

In PD controller, the rise time of the human arm trajectory is 0.36 seconds. The peak time produced is 0.50 seconds. The settling time of the human arm trajectory is 2.11 seconds. The overshoot of the model is 8.67 degrees with the percentage of overshoot (%OS) is 14.45%. The human arm model has the steady-state error of 4.09.

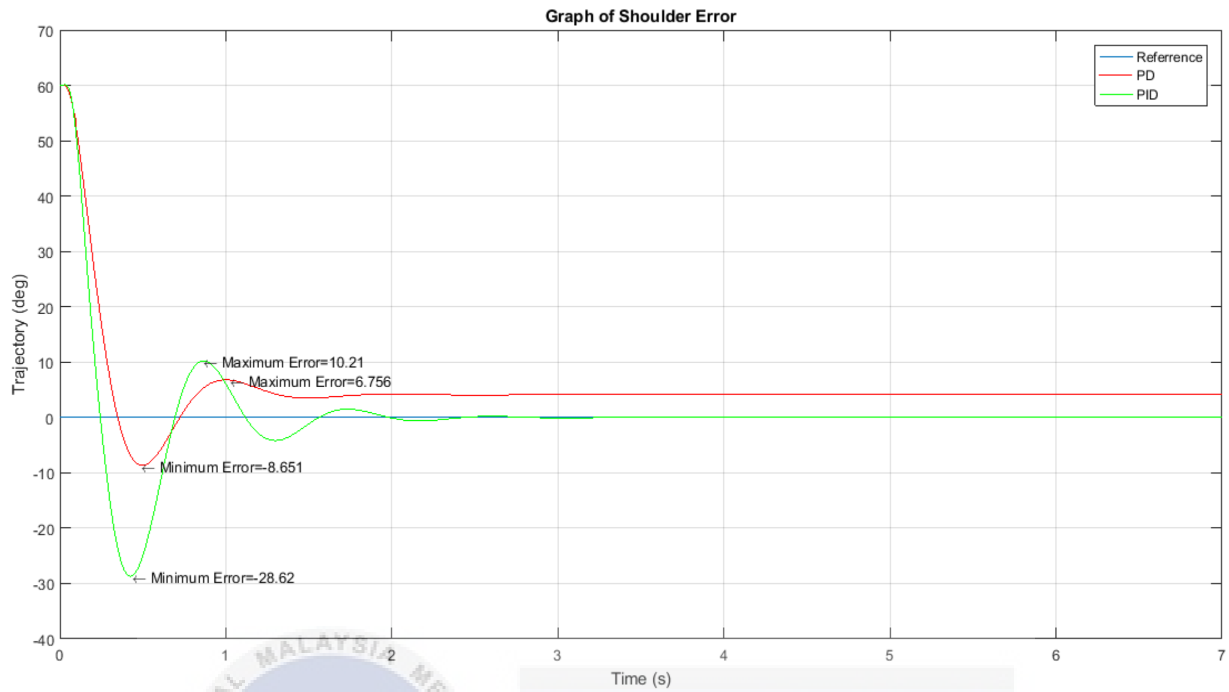


Figure 4.15: Trajectory error of shoulder joint for $K_u = 2$

Figure 4.15 shows the output error of PID controller and PD controller. In PID controller the minimum error is -28.62 degrees. The maximum error is 10.21 degrees. The range of error produced by the controller is 38.33 degrees.

In PD controller, the minimum error produced by system is -8.651 degrees. The maximum error in the system is 6.756 degrees. The range of error produced by the controller is 15.41 degrees.

Elbow Joint for $K_u = 2$

Table 4.5: The (Z-N) tuning value on elbow joint for $K_u = 2$

Type of Controller	K_p	K_i	K_d
PD	1.6		0.032
PID	1.2	15	0.024

In elbow joint, the value of K_u was set to 2 and sustained oscillation time, T_u is 0.16 seconds .The trajectory on the elbow joint was set to 30^0. Figure 4.18 shows the movement of the elbow joint on PID controller and PD controller. The value on the table 4.5 is inserted in both controller.

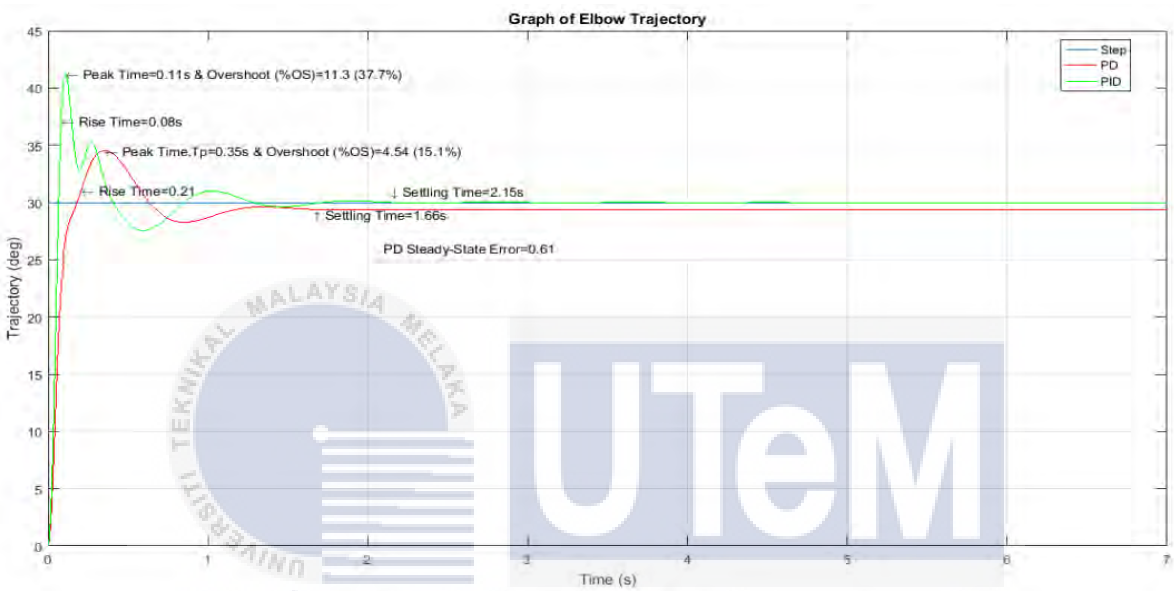


Figure 4.16: Trajectory of elbow joint for $K_u = 2$

Figure 4.16 shows the transient and steady-state behavior of PID controller and PD controller on the trajectory of elbow joint in human arm model. In PID controller the rise time of human arm model trajectory is 0.08 seconds. The peak time of the trajectory is 0.11 seconds and the settling time of human arm model trajectory is 2.15 seconds. The human arm model trajectory have overshoot of 11.3 degrees. No steady state error occurs at human arm model since the model are stable at the set point.

In PD controller, the rise time of the human arm trajectory is 0.21 seconds. The peak time produced is 0.35 seconds. The settling time of the human arm trajectory is 1.66 seconds. The overshoot of the human arm model is 4.54 degrees. The human arm model has the steady state error of 0.61 which the system does not reach set point using PD controller.

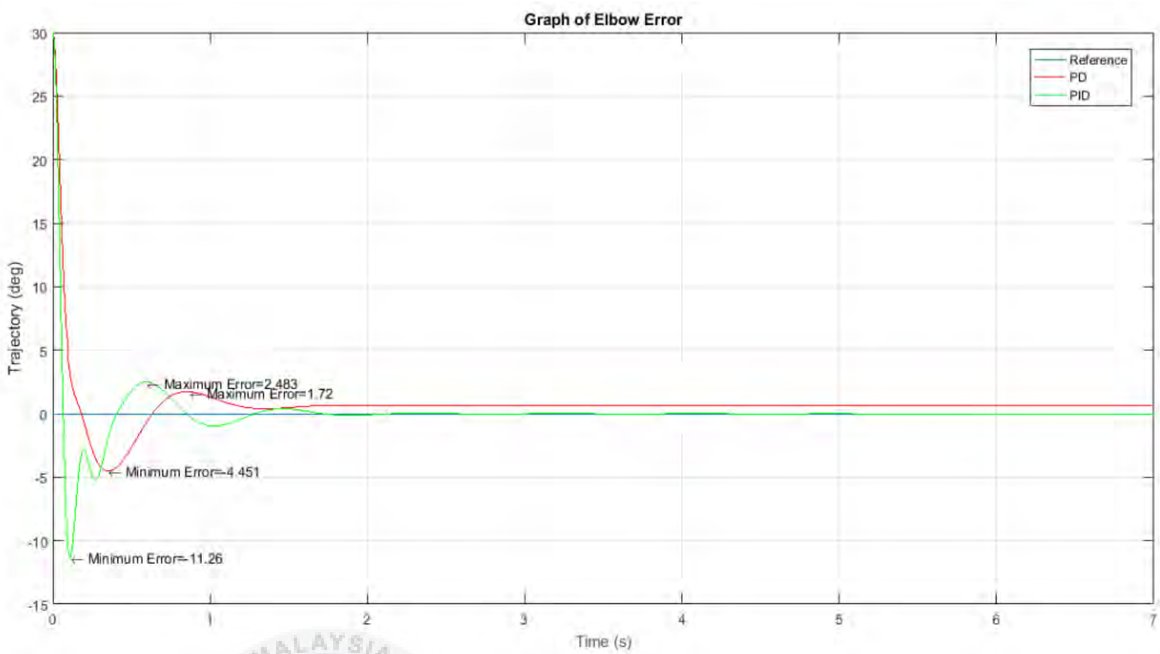


Figure 4.17: Trajectory error of elbow joint for $K_u = 2$

Figure 4.17 shows the output error of PID controller and PD controller. Table 4.16 shows the data of output error produced by both controller. In PID controller the minimum error is -11.25 degrees. The maximum error is 2.483 degrees. The range of error produced by the controller is 13.73 degrees.

In PD controller, the minimum error produced by system is -4.451 degrees. The maximum error in the system is 1.72 degrees. The range error produced by the controller is 6.17 degrees.

4.3.2 The comparison of performance trajectory and error on human arm joints for $K_u = 1$

The best performance of trajectory controller of the joints are taken from the requirement of transient response and steady-state response. The best controller performance are taking from the controller which have minimum overshoot (*%OS*), minimum settling time (T_s) and minimum steady-state error (e_{ss}). Table 4.20 shows the comparison of PID and PD controller of shoulder joint and elbow joint for initial gain of $K_u = 1$.

Table 4.6: Data of PID and PD controller for $K_u = 1$

$K_u = 1$	PID Controller	PD controller
Shoulder Joint	• **T_r : 0.488 seconds** • **T_p : 0.77 seconds** • T_s : 5.85 seconds • Overshoot (%OS): 0 (0%) • **e_{ss}: 0**	• T_r : 0.64 seconds • T_p : 0.78 seconds • **T_s : 2.078 seconds** • Overshoot (%OS) : 0 (0%) • e_{ss}: 6.87
Elbow Joint	• **T_r : 0.24 seconds** • **T_p : 0.37 seconds** • **T_s : 1.63 seconds** • **Overshoot (%OS): 0 (0%)** • **e_{ss}=0**	• T_r : 0.30 seconds • T_p :0.50 seconds • T_s :1.83 seconds • Overshoot (%OS): 2.43 (8.10%) • e_{ss}: 1.02

Based on table 4.6, PID controller in $K_u = 1$ is suitable for the shoulder joint. This is because PID controller minimum value of rise time (T_p) of 0.77 seconds and overshoot of 0 (%OS= 0%) of transient response compared to transient response of PD controller. In steady state response, the settling time (T_s) is5.85 seconds for PID controller This show that PID controller settling time is late compared to PD controller. However, PID controller didn't have steady-state error ($e_{ss} = 0$) with the minimum value of 0. System with minimum steady state error is important because it is to proof that the system is follow the set point. In this analysis, we found that the shoulder joint moves constant in the set point of 60^0 trajectory.

Next, PID controller also gives the better performance in elbow joint for $\mathrm{K_u} = 1$. This is because PID controller have follow the requirements that the controller have minimum overshoot (%OS) of 0, minimum settling time (T_s) which 1.63 seconds and minimum steady-state error (e_{ss}) of 0. This show that PID controller have better performances in terms of transient response and steady-state response compared to PD controller.

Table 4.7: Data of output error of PID and PD controller for $K_u = 1$

$K_u = 1$	PID Controller	PD controller
Shoulder Joint	• e_{max}: 1.191 degrees • e_{min} : 0.068 degrees • **Range of error :1.123 degrees**	• e_{max}:8.402 degrees • e_{min}: 7 degrees • Range of error :1.402 degrees
Elbow Joint	• e_{max}:1.221 degrees • e_{min}: -8.407 degrees • Range of error : 9.628 degrees	• e_{max}: 1.001 degrees • e_{min}: 0.48 degrees • **Range of error : 0.521 degrees**

Based on the table 4.7, the data of output error are analyze on the shoulder joint and elbow joint. The PID controller and PD controller error are analyzed. PID controller have minimum range of error of 1.123 degrees on shoulder joint compared to PD controller. Thus, PID controller on shoulder joint have its ability to eliminate the error compared to PD controller. Thus the graph in figure 4.10 shows that PID controller is good controller in shoulder joint as it can eliminate the error.

In elbow joint, PD controller have the minimum range of error compared PID controller. The minimum range of error for PD controller is 0.521 degrees. However, based on the graph shown in figure 4.13, PID controller are able to eliminate the error compared to PD controller during the simulation time of 1 to 2 seconds. PD controller couldn't eliminate the error during the simulation. So, PID is the good controller of both joints which can eliminate the error compared to PD controller.

4.3.3 The comparison of performance controller trajectory and error on the human arm joints for $K_u = 2$

The best performance of trajectory controller of the joints are taken from the requirement of transient response and steady-state response. The best controller performance are taking from the controller which have minimum overshoot (%OS), minimum settling time (T_s) and minimum steady-state error (e_{ss}). Table 4.21 shows the comparison of PID and PD controller of shoulder joint and elbow joint for initial gain of $K_u = 2$.

Table 4.8: Data of PID and PD controller for $K_u = 2$

$K_u = 2$	PID Controller	PD controller
Shoulder Joint	• **T_r: 0.32 seconds** • T_p :0.64 seconds • T_s :2.85 seconds • Overshoot: 28.68 (47.8%) • **e_{ss}: 0**	• T_r: 0.36 seconds • **T_p :0.50 seconds** • **T_s :2.11 seconds** • **Overshoot: 8.67 (14.45%)** • e_{ss}: 4.09
Elbow Joint	• **T_r: 0.08 seconds** • **T_p :0.11 seconds** • T_s :2.15 seconds • Overshoot = 11.3 (37.7%) • **e_{ss}: 0**	• T_r: 0.21 seconds • T_p :0.35 seconds • **T_s :1.66 seconds** • **Overshoot: 4.54 (15.1%)** • e_{ss}: 0.61

Based on table 4.8, PD controller in $K_u = 2$ is suitable for the shoulder joint. This is because PD controller have minimum value peak time (T_p) of 0.50 seconds. The overshoot (%OS) which is 8.67 (14.45%) of PD have less value in transient response compared to transient response of PID controller. In steady state response, the settling time for PD controller is faster compared to PID controller. However, PD controller have steady-state error (e_{ss}) with the minimum value of 4.09. This doesn't affect the system as long as the system doesn't the 20% of steady-state error (e_{ss}). So, PD controller is chosen for the movement of shoulder $K_u = 2$.

Next, PD controller also gives the better performance in elbow joint for $K_u = 2$. This is because PD controller have follow the requirements that the controller have minimum overshoot (%OS) which is 4.54 (15.1%) and minimum settling time (T_s) of 1.66 seconds. This show that PD controller have better performances in terms of transient response and steady-state response compared to PD controller.

Table 4.9: Data of output error of PID and PD controller for $K_u = 2$

$K_u = 2$	PID Controller	PD controller
Shoulder Joint	• e_{max}: 10.21 degrees • e_{min}: -26.82 degrees • Range of error : 38.83 degrees	• e_{max}: 6.756 degrees • e_{min}: -8.651 degrees • ***Range of error* : 15.41 degrees**
Elbow Joint	• e_{max}: 2.483 degrees • e_{min}: -11.25 degrees • *Range of error* : 13.73 degrees	• e_{max}: 1.72 degrees • e_{min}: -4.451 degrees • ***Range of error* : 6.17 degrees**

Based on table 4.9, the data of output error are analyze on the shoulder joint and elbow joint for $K_u = 2$. The PID and PD controller error are analyzed. PD controller have minimum range of error on shoulder joint compared to PID controller which the value is 15.41 degrees. Thus, PD controller on shoulder joint have its ability to eliminate the error compared to PID controller. However, based on the graph shown in figure 4.15, the PID controller are able to eliminate error although it has higher range of error compared to PD controller. In elbow joint, PD controller have the minimum error range compared PID controller. However, based on the graph shown in figure 4.17, PID controller are able to eliminate the error compared to PD controller during the simulation. PD controller couldn't eliminate the error during the simulation. So, PID is the good controller which can eliminate the error.

4.4 Discussion

The model are designed in Solidworks. In Solidworks model are designed with normal joint in shoulder joint and knuckle joint at elbow joint. The human arm model are designed with two stages model. The stage 1 model consist 1 arm human model and stage 2 model is two arm model. Completed design of human arm is transfer from the Solidworks to MATLAB through Simmechanics. The proof of the human arm model is imported to the MATLAB are shown as it appears as Simulink output in the MATLAB Simulink. Stage 1 model which is 1 arm model are selected for the simulation. The model is run by adding the blocks to the block diagram. Combination between Simulink output and added blocks produce the system where the human arm model are to simulate. In the beginning the model is design in open-loop system in order to actuate the model. Open-loop system also are used in order to discover the transfer function. Transfer function are discovered as representation functions of the plant. Next, the model is test in closed-loop system by using the PID controller and PD controller which is attached on 3 parts. 1 controller attached to shoulder joint and another two controller attached at elbow. Set point of the model is set as the trajectory and the value of the controllers are tune using Ziegler-Nichols tuning technique. Ziegler-Nichols have two tuning methods which is Ziegler-Nichols open-loop tuning method and Ziegler-Nichols closed-loop tuning method. This project use Ziegler-Nichols closed loop tuning method. This is because the controller are designed in feedback design and initial transient behavior which is continuously oscillate when initial gain is set as $K_u = 1$ and $K_u = 2$. PID controller and PD controller are run in separate system. 1 system run using PID controller and 1 system run using PD controller. The output model performance are analyzed in the transient behavior performance and error performance. The PID controller and PD controller are analyzed from this criteria. The response of the system are able to observe in the scope block. The experimental results shows that PID controller have the good transient response compared to PD controller for initial gain of $K_u = 1$. This shows that PID controller is suitable for shoulder joint and elbow joint. Next, the experimental results for $K_u = 2$ shows that PD controller have the good transient and steady-state response compared to PID controller. This is because, the system of both joint has achieved the highest gain. So, PD controller is suitable for the control system on shoulder and elbow joint for $K_u = 2$ although the system which have steady-state error on both joints. In the error analysis, PID controller are able to eliminate the error in both shoulder and elbow joints for initial gain of $K_u = 1$ and $K_u = 2$ during the simulation although have larger error compared to PD controller. PID controller are able to sustain at the reference point which is equal to 0 compared to PD controller.

4.5 Summary

The result is obtain by following the instruction from the methodology. Two stages model (stage 1 model: 1 arm model and stage 2 model: 2 arm model) are designed in

Solidworks .The model are able to transfer from the Solidworks to MATLAB. The proof of the human arm model is imported to the MATLAB are shown as it appears as the Simulink output in the Simulink. Combination between Simulink output and added blocks which controller design produce a system. Transfer function are determined from open-loop control system and the system is analyzed from in closed-loop control system. The system is analyses as the shoulder joint is attach with one controller and elbow joint is attached with two controllers. The controllers that is used in this project is PID controller PD controller which using the Ziegler-Nichols closed loop tuning methods. The performance of the controller are analyses in terms of transient response and error response. PID controller and PD controller are compared where both of the controller gives the good benefits to the system.

CHAPTER 5

CONCLUSION

This chapter is discussing about the conclusion. Conclusion is importance for the future development of the result. Next, the future works is discuss as the sub chapter of this topic in order for further development of the project.

5.1 Conclusion

The Development and Control of Exoskeleton (Upper Limb) for Mobility Support focuses on the human arm development. The human arm model is formed in physical modelling. The development of human arm model is using the 2 degree of freedom (2-DOF) .1-DOF includes the extension/flexion of shoulder joint .The other 1-DOF is the extension/flexion of elbow joint. A model is developed by using two software. The software are Solidworks (version 2015) and MATLAB (version 2016b). 2 stage model was designed in Solidworks. Stage 1 model uses 1 arm and stage 2 model consists of two arms. Designed human arm model in Solidworks is transferred to MATLAB. The model is transfer to MATLAB by using the Simmechanics. Simmechanics is the plug-in software based Simscape that enables human arm model to transfer between both software. Only stage 1 model is selected for simulation. The MATLAB is used to analyses the human arm model performance. In MATLAB, the human arm model are represented as Simulink output in MATLAB Simulink. Transfer function is determined and the model performance is analyzed. Some modification has been made to the model by adding some additional blocks (controller set) to the human arm model Simulink output. The combination produces a control system that is able to actuate the movement of arm. Next, the human arm model is tested by using adding a controller to Simulink output. The combination produces a system. The controller used in this project is Proportional-Integral-Derivative (PID) controller and Proportional-Derivative (PD) controller. PID controller and PD controller are selected in this project in order to compare which the controller is better performance and suitable for the stroke and arm injuries people. PID controller and PD controller is tested by putting on two joints of human arm. One controller is put on shoulder joint while other one controller are put in elbow joint. The simulation, the performance of human arm model is analyzed by injecting the step input which is trajectory to the model. The trajectory was set to 60^0 on shoulder joint and 30^0 on elbow joint. The human arm model are run in different system. One system is run using PID controller and the second system is run using PD controller. The performance of the both controller are analyses based on transient response and error response. It can be concluded that PID controller have better performance at initial gain $K_u = 1$ while PD controller have better performance at initial gain of $K_u = 2$. PID controller needed in $K_u = 1$ to actuate the movement. So, PID controller is suitable for the shoulder joint and elbow joint for $K_u = 1$ .PD controller suitable for gain of $K_u = 2$ as the system have achieve highest gain .So, PD controller is suitable for shoulder and elbow joint for the initial gain of $K_u = 2$. In error analysis, PID controller considered as robust controller because the error produced is bigger than error produced by PD controller for the shoulder and elbow joint in $K_u = 1$ and $K_u = 2$. Although the PID controller produces large error, it can sustain it desired response by follow the reference point.

5.2 Future Works

The future works of this project is to upgrade the design of the model in the Solidworks software. The model design in Solidworks is using mechanical joint type, knuckle joint. The arm model stage 2 which is 2 arm model will be used in the next project. A complex model will be design might be design beyond the current human arm model. The human arm model also will be design into 3 degree of freedom (3- DOF) instead of two degree of freedom (2-DOF). 2-DOF of human arm is develop in this project.

A fuzzy controller or fuzzy-PID controller will be used in the next project. The tuning method such as Cohen-Coon or Tyreus- Luyben tuning method. These controllers are suitable in tuning the PID controller. The controller will be compare and determine which one of the controller is better for the development of the assistive robot arm for this project.

REFERENCES

[1] D. Harper, "Online Ethymology Dictionary," 2001. [Online]. Available: http://www.etymonline.com/index.php?term=exoskeleton. [Accessed 2016 October 28].

[2] N. L. o. Medicine, "MeSH Descriptor Data," MeSH, 2011. [Online]. Available: https://www.nlm.nih.gov/cgi/mesh/2011/MB_cgi?mode=&term=Upper+Extremity&field=entry#TreeA01.378.800. [Accessed 2016 October 29].

[3] Panich, S. Kinematic analysis of exoskeleton suit for human arm. Journal of Computer Science, 6(11), p.1272, 2010.

[4] Stroke Statistics [Online].
Available:http://www.thepossibilities.co.uk/assets/downloads/stroke-statistics.pdf
[Accessed 2016 December 13].

[5] Hand Injury Statistics.Available:http://www.superiorglove.com/pages/blog/hand-injury-statistics [Accessed 2016 December 14].

[6] Geravand, M., Rampeltshammer, W. and Peer, A., 2015, August. Control of mobility assistive robot for human fall prevention. In 2015 IEEE International Conference on Rehabilitation Robotics (ICORR) (pp. 882-887). IEEE.

[7] Geravand, M. and Peer, A., 2014, August. Safety constrained motion control of mobility assistive robots. In 5th IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics (pp. 1073-1078). IEEE.

[8] Chen, S.H., Lien, W.M., Wang, W.W., Lee, G.D., Hsu, L.C., Lee, K.W., Lin, S.Y., Lin, C.H., Fu, L.C., Lai, J.S. and Luh, J.J., 2016. Assistive Control System for Upper Limb Rehabilitation Robot.

[9] Naidu, D., Stopforth, R., Bright, G. and Davrajh, S., 2011, September. A 7 DOF exoskeleton arm: Shoulder, elbow, wrist and hand mechanism for assistance to upper limb disabled individuals. In AFRICON, 2011 (pp. 1-6). IEEE.

[10] Moubarak, S., Pham, M.T., Moreau, R. and Redarce, T., 2010, August. Gravity compensation of an upper extremity exoskeleton. In 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology (pp. 4489-4493). IEEE.

[11] Modelling and Simulation.
Available at http://www.systems-thinking.org/modsim/modsim.htm [Accessed 2016 December 15].

[12] Nagarsheth, H.J., Savsani, P.V. and Patel, M.A., 2008, August. Modeling and dynamics

of human arm. In 2008 IEEE International Conference on Automation Science and Engineering (pp. 924-928). IEEE.

[13] Parasuraman, S., Kee, C.Y. and Oyong, A., 2009. Human upper limb and arm kinematics for robot based rehabilitation. In IEEE/ASME International Conference on Advanced Intelligent Mechatronics (I-Meng Chen 14 July 2009 to 17 July 2009) (Vol. 1, pp. 845-850). IEEE.

[14] Babiarz, A., Klamka, J., Zawiski, R. and Niezabitowski, M., 2014, June. An approach to observability analysis and estimation of human arm model. In 11th IEEE International Conference on Control & Automation (ICCA) (pp. 947-952). IEEE.

[15] Choudhury, T.T., Rahman, M.M., Khorshidtalab, A. and Khan, M.R., 2013, September. Modeling of Human Arm Movement: A Study on Daily Movement. In 2013 Fifth International Conference on Computational Intelligence, Modelling and Simulation (pp. 63-68). IEEE.

[16] Poyi, G.T., Wu, M.H., Bousbaine, A. and Wiggins, B., 2013, June. Validation of a quad-rotor helicopter Matlab/Simulink and solidworks models. In Control and Automation 2013: Uniting Problems and Solutions, IET Conference (pp. 1-6). IET.

[17] Othman, A., Belda, K. and Burget, P., 2015, October. Physical modelling of energy

consumption of industrial articulated robots. In 15th International Conference Control, Automation and Systems (ICCAS), 2015 (pp. 784-789). IEEE.

[18] Yun, C., Rong, C. and Jian, S., 2012, June. Simulation design on the 6-dof parallel vibration platform Based on SimMechanics and VirtualReality. In World Automation Congress (WAC), 2012 (pp. 1-4). IEEE.

[19] Ming, L.Z., 2015, June. Agricultural Product Quality Control System Design Based on PID Control. In 2015 Seventh International Conference on Measuring Technology and Mechatronics Automation (pp. 76-80). IEEE.

[20] Liang, G., Ye, W. and Xie, Q., 2012, August. PID control for the robotic exoskeleton: Application to lower extremity rehabilitation. In 2012 IEEE International Conference on Mechatronics and Automation (pp. 2345-2350). IEEE.

[21] Iqbal, K. and Roy, A., 2002. PID controller design for the human arm-robot manipulator coordination problem. In Intelligent Control, 2002. Proceedings of the 2002 IEEE International Symposium on (pp. 121-124). IEEE.

[22] Yu, W. and Rosen, J., 2010, December. A novel linear PID controller for an upper limb exoskeleton. In 49th IEEE Conference on Decision and Control (CDC) (pp. 3548-3553). IEEE.

[23] Pooneh Gholami and Samaneh Sadeghi, "Properties of PID Controllers," [Online]. Available: http://saba.kntu.ac.ir/eecd/ecourses/Indust%2084/Selected%20Projects/Sadeghi%20&%20Gholami/information/properties.htm. [Accessed 29 November 2016].

[24] Dey, C., Mudi, R.K. and Paul, P.K., 2012, December. A non-linear PD controller for pure integrating process with delay. In Electrical & Computer Engineering (ICECE), 2012 7th International Conference on (pp. 783-786). IEEE.

[25] Tomei, P., 1991. A simple PD controller for robots with elastic joints. IEEE Transactions on automatic control, 36(10), pp.1208-1213.

[26] Jazeh, H.E.S., Samavati, F.C., Vaezi, M. and Moosavian, S.A.A., 2011, August. The robustness analysis of two motion control algorithms for Stäubli TX40 robot in object manipulation task. In Mechatronics and Automation (ICMA), 2011 International Conference on (pp. 2165-2170). IEEE.

[27] Majid, M.A., Ibrahim, W.S.W., Mohamad, S. and Bakar, Z.A., 2013, December. A comparison of PID and PD controller with input shaping technique for 3D gantry crane. In Systems, Process & Control (ICSPC), 2013 IEEE Conference on (pp. 144-148). IEEE.

[28] e Silva, F.O. and de Carvalho Ferreira, L.H., 2013, May. Design and implementation of a PID control system for a coaxial two-wheeled mobile robot. In Industrial Electronics (ISIE), 2013 IEEE International Symposium on (pp. 1-6). IEEE.

[29] Kuo, Y.L. and Liu, S.M., 2015. Position Control of a Serial Manipulator Using Fuzzy-PID Controllers. International Journal of Automation and Smart Technology, 5(1), pp.18-26.

APPENDIX

Appendix A: Gantt chart for Final Year Project I

Item \ Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14
FYP TITLE														
Meet Supervisor														
Title Confirmation														
Topic Familiarization														
Log Book Preparation														
CHAPTER 1														
Objective & Scope														
Problem Statement														
Report Outline														
CHAPTER 2														
Basic Principle & Theory														
Source Findings														
Literature Review														
CHAPTER 3														
Methodology														
CHAPTER 4														
Result & Discussion														
CHAPTER 5														
Conclusion														
Panel Feedback Form														
Submission of FYP Report														
FYP Presentation														

Appendix B: Gantt chart for Final Year Project II

Item / Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14
FYP TITLE														
Meet Supervisor														
Title Confirmation														
Topic Familiarization														
Log Book Preparation														
CHAPTER 1														
Objective & Scope														
Problem Statement														
Report Outline														
CHAPTER 2														
Basic Principle & Theory														
Source Findings														
Literature Review														
CHAPTER 3														
Methodology														
CHAPTER 4														
Result & Discussion														
CHAPTER 5														
Conclusion														
Panel Feedback Form														
Submission of FYP Report														
FYP Presentation														